

الجمهورية الجزائرية الديمقراطية الشعبية

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**

وزارة التعليم العالي والبحث العلمي

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**

– جامعة أبي بكر بلقايد – تلمسان

**Université Aboubakar Belkadi – Tlemcen –**

**Faculté de TECHNOLOGIE**



**MEMOIRE**

Présenté pour l'obtention du **diplôme** de **MASTER**

**En** : Génie industriel

**Spécialité** : Ingénierie de la production

**Par** : BENYETTOU Amina Souhila & HABIB Khadîdja

Thème

Contrôle qualité et détection de défauts pour les systèmes de triage  
industriel par l'utilisation de deep learning

Soutenu publiquement, le 18 / 06 / 2023 , devant le jury composé de :

Mme HOUBAD Yamina	MCB	Université de Tlemcen	Président
M.BESSENOUCI Hakim Nadir	MAA	Université de Tlemcen	Examineur
Mme. LARIBI Imane	MCB	Université de Tlemcen	Examineur
Mme TRIQUI-SARI Lamia	MCA	Université de Tlemcen	Encadreur
M. KADRAOUI Hicham	MCB	Université de Tlemcen	Co-Encadreur



### ***Remerciement***

*Nous remercions Dieu Tout-Puissant qui nous a aidés à compléter ce mémoire et a éclairé nos chemins pour nous et nous a permis de le compléter.*

*Nous adressons nos remerciements et notre appréciation à notre encadrante, le Mme TRIQUI-SARI Lamia, qui n'a pas lésiné sur nous avec toutes ses informations et références qui ont contribué à enrichir le sujet de notre étude dans ses différents aspects.*

*Nous adressons également nos remerciements notre Co-Encadrant M. KADRAOUI Hicham pour les conseils et les orientations qu'il nous a données et qui nous ont aidés à compléter ce mémoire.*

*Nos respectueux remerciements sont dédiés aux membres du jury, Melle HOUBAD Yamina, M.BESSENOUCI Hakim Nadir et Mme LARIBI Imane d'avoir accepté d'examiner et de porter leur jugement sur ce modeste travail, sans oublier le directeur universitaire de la faculté de technologie et tous les professeurs du département de génie électrique et électronique.*

*Nous adressons également nos sincères remerciements à nos parents et à tous ceux qui nous ont soutenus tout au long de notre parcours scolaire.*

*Et enfin, nous présentons nos vifs remerciements à tous ceux et celles qui nous ont aidées de près ou de loin à l'élaboration de ce travail.*

# Dédicaces

*La locomotive de mes recherches a traversé de nombreux obstacles, et pourtant j'ai essayé de les surmonter fermement avec la grâce de Dieu Allah.*

*Et maintenant, mon parcours universitaire s'est terminé après l'épuisement et les difficultés....*

*A celui qui la préfère à moi, et pourquoi pas, et n'a ménagé aucun effort pour toujours me rendre heureux. Chère mère.*

*A la personne avec un visage gentil et de bonnes actions, Ali n'a pas été avare tout au long de sa vie. Mon cher père.*

*A mes frères : **CHERIF, AYOUB, KHALED, KADA**, et à ma sœur **FATIMA ZOHRA** et le poussin **FARAH**.*

*A mes amis : **CHAIMA, WAHIBA, IKRAM, HOUDA, MERIEM, RANIA, NABILA, LAMIA**, et bien sur **SOUHILA AMINA**.*

*Je ne dois pas oublier mes professeurs qui ont eu le plus grand rôle au cours de ma carrière universitaire.*

*A toutes les personnes pour qui j'ai de l'amour et de la reconnaissance.*

*A tous ceux oubliés par la plume et sauvés par le cœur.*

**KHADIDJA HABIB**

# Dédicaces

*Je dédie ce mémoire à mes chers parents pour leur patience. A ma mère « **Fatma** », la lumière de ma vie, qui m'a soutenu et encouragé durant ces années d'études. A mon très cher père « **Adda** », ma gratitude ne suffit pas à exprimer ce qu'il mérite pour ses sacrifices depuis ma naissance, pendant mon enfance et même à l'âge adulte.*

*A mes sœurs **Keltoume, Fatiha, Keira**, À mes frères **Amine, Omar, Said, Tayeb** et **Ahmed**.*

*A mon cher mari **Soufiane** pour son soutien et son encouragement.*

*A tous mes amis qui m'ont soutenu dans l'accomplissement de cet humble travail : **Faiza, Rania, Khaoula, Farah, Chaimaa, Imane, Wissem, Amina, Karima, Zohra, Hadil** et **Lamis**.*

*A ma précieuse sœur **Rima** pour ses conseils, et sa merveilleuse aide à mes côtés et son soutien moral, sa compréhension en la souhaitant toute la joie et la prospérité pour elle et sa famille.*

*A tous mes chères étudiantes de **classe 148**.*

*A mon binôme **Khadidja** pour son partage de connaissances et ses efforts pour la réalisation de ce travail, je la souhaite mes meilleurs vœux de succès.*

*A ce qui nous ont été crédités de notre formation religieuse **Nbquran, Jil Tarjih** et **Ugel**.*

*A tous mes enseignants et à tous ceux qui ont contribué de près ou de loin à la réussite de ce projet.*

*A toute ma famille **BENYETTOU, OUAHBA** et **RABAH**.*

*Enfin, à moi-même, pour avoir travaillé dur et préservé malgré les obstacles et doutes. Ce projet représente pour moi une étape importante de ma vie et je suis fière de ce que j'ai accompli.*

**BENYETTOU Amina Souhila**

## Résumé

Ce mémoire se concentre sur la classification des images de boîtes d'emballage en utilisant le Deep Learning. Pour ce faire, un système de classification et de détection d'images a été créé en utilisant un réseau de neurones convolutifs (CNN). L'implémentation du programme a été réalisée sur Google Colab avec un accélérateur GPU pour accélérer les calculs mathématiques. Le programme utilise une série de cellules pour classer la nature de boîtes (défectueuse/non défectueuse). Pour que cette solution puisse être intégrée sur le terrain, des modifications doivent être apportées pour qu'elle puisse traiter un grand nombre de boîtes en temps réel à l'aide de caméras industrielles. Les résultats de la classification ont été satisfaisants avec une précision supérieure à 90%. Des techniques de prétraitement d'image ont également été appliquées pour améliorer la qualité des images avant l'entraînement du modèle. Enfin, les notions et définitions de base du domaine du Deep Learning ont été exposées, ce qui permet une meilleure compréhension de l'approche adoptée dans ce travail de recherche.

**Mot clés :** Classification, Deep Learning, système de détection, prétraitement d'image.

## Abstract

This thesis focuses on the classification of packaging box images using Deep Learning. To achieve this, a system for image classification and detection was created using Convolutional Neural Networks (CNNs). The program implementation was done on Google Colab with a GPU accelerator to speed up mathematical calculations. The program uses series of cells to classify the nature of the boxes (defective/non-defective). To integrate this solution in the field, modifications must be made to enable it to process a large number of boxes in real-time using industrial cameras. The classification results were satisfactory with an accuracy greater than 90%. Image preprocessing techniques were also applied to improve the quality of images before model training. Finally, the basic concepts and definitions of the Deep Learning domain were presented, which allows for a better understanding of the approach adopted in this research.

**Keywords :** Classification, Deep Learning, detection system, image preprocessing.

## ملخص

تركز هذه الأطروحة على تصنيف صور عب التغليف باستخدام التعلم العميق. لتحقيق ذلك، تم إنشاء نظام لتصنيف الصور واكتشافها باستخدام الشبكات العصبية التلافيفية (CNNs). تم تنفيذ البرنامج على Google Colab باستخدام مسرع GPU لتسريع العمليات الحسابية. يستخدم البرنامج سلسلة من الخلايا لتصنيف طبيعة الصناديق (معيبة / غير معيبة). لدمج هذا الحل في المجال، يجب إجراء تعديلات لتمكينه من معالجة عدد كبير من الصناديق في الوقت الفعلي باستخدام الكاميرات الصناعية. كانت نتائج التصنيف مرضية وبدقة تزيد عن 90%. كما تم تطبيق تقنيات المعالجة المسبقة للصور لتحسين جودة الصور قبل تدريب النموذج. أخيراً، تم تقديم المفاهيم والتعريفات الأساسية لمجال التعلم العميق، مما يسمح بفهم أفضل للنهج المتبع في هذا البحث.

**الكلمات المفتاحية:** التصنيف، التعلم العميق، نظام الكشف، المعالجة المسبقة للصور.

## Liste des acronymes

**ACP** Analyse en Composantes Principales.  
**ADAM** Adaptive Moment Estimation  
**ADC** Classification Descendante Hiérarchique.  
**AE** Auto-Encoder.  
**ANN** Artificial Neural Network.  
**BMP** Best Management Practice.  
**BSD** Berkeley Software Distribution.  
**CD** Contrôle Destructif.  
**CGAN** Conditional Generative Adversarial Network.  
**CND** Contrôle Non Destructif.  
**CNN** Convolutional Neural Network.  
**CRNN** Réseau de Neurones Convolutionnel Récurrent.  
**D** Dimension.  
**DCGAN** Deep Convolutional Generative Adversarial Network.  
**DNN** Deep Neural Networks.  
**ER** Entité-Relation.  
**FFNN** Feedforward Neural Network.  
**GAN** Generative Adversarial Network.  
**GIF** Graphics Interchange Format.  
**GIMP** GNU Image Manipulation Program.  
**GPU** Ground Power Unit.  
**GRU** Gated Recurrent Unit.  
**HCA** Analyse de Classification Hiérarchique.  
**HTML** Hyper Text Markup Language.  
**IA** Intelligence Artificiel.  
**IDE** Environnement de Développement Intégré.  
**JPEG** Joint Photographic Experts Group.  
**LSTM** Long Short-Term Memory.  
**MLP** Multilayer Perceptron.  
**PDF** Portable Document Format.  
**PNG** Portable Network Graphics.  
**RNN** Recurrent Neural Network.  
**SSD** Solid-State Drive.  
**SVM** Support Vector Machine.  
**TPU** Tensor Processing Unit.  
**UV** Expositions aux Rayons.  
**VCS** Visionnèrent de Code Source.  
**WGAN** Wasserstein Generative Adversarial Network.  
**XML** Extensible Markup Language.

## Table des figures

<b>Figure 1.1:</b> Les erreurs de coupe d'une boîte de ravier en polystyrène. ....	7
<b>Figure 1.2:</b> Les défauts sur le calcul numérique de l'importance du maillage. ....	8
<b>Figure 1.3:</b> La déformation d'une boîte de ravier en polystyrène. ....	8
<b>Figure 2.1:</b> Apprentissage supervisée.....	28
<b>Figure 2.2:</b> Classification et régression. ....	28
<b>Figure 2.3:</b> Régression linéaire et régression logistique. ....	29
<b>Figure 2.4:</b> Architecture d'un vecteur support machine.....	30
<b>Figure 2.5:</b> Architecture d'un arbre de décision.....	31
<b>Figure 2.6:</b> Neurone biologique et neurone artificiel (Werfelli, 2015). ....	33
<b>Figure 2.7:</b> Apprentissage non supervisé (Géron, 2017). ....	33
<b>Figure 2.8:</b> Apprentissage par renforcement (Géron, 2017). ....	36
<b>Figure 2.9:</b> Le choix de l'algorithme d'apprentissage selon certains facteurs (Li,2017). ....	37
<b>Figure 2.10:</b> Les sous-branches de l'intelligence artificiel. ....	38
<b>Figure 2.11:</b> Structure d'un réseau de neurone biologique et le cerveau humain. ....	38
<b>Figure 2.12:</b> Architecture d'un réseau de neurones à propagation avant. ....	39
<b>Figure 2.13:</b> Réseau neuronal récurrent simple. ....	40
<b>Figure 2.14:</b> Réseau de neurones convolutifs pour identifier l'image d'un oiseau. ....	42
<b>Figure 2.15:</b> Schéma simplifié du processus d'entraînement du GAN conditionnel. ....	44
<b>Figure 2.16:</b> Application de génération de visages dans une image avec un GAN. ....	45
<b>Figure 2.17:</b> Les graphes de chaque état (Géron, 2017). ....	47
<b>Figure 3.1:</b> Concept général de système proposé. ....	51
<b>Figure 3.2:</b> Schéma des étapes du traitement d'image.....	52
<b>Figure 3.3:</b> Image en défèrent couleur d'une boîte.....	53
<b>Figure 3.4:</b> Les dimensions d'une boite. ....	54
<b>Figure 3.5:</b> Contour sur la dimension de la boite.....	55
<b>Figure 3.6:</b> Diagramme de la phase de la création de modèle.....	57
<b>Figure 3.7:</b> Logo du Python. ....	62
<b>Figure 3.8:</b> La version de Python 3.9. ....	63
<b>Figure 3.9:</b> Logo de Google colab. ....	64
<b>Figure 3.10:</b> Logo de la bibliothèque OpenCV. ....	65
<b>Figure 3.11:</b> Logo de logiciel PyCharm. ....	66
<b>Figure 3.12:</b> Installation PyCharm.....	67
<b>Figure 3.13:</b> Interface PyCharm. ....	68
<b>Figure 3.14:</b> Création de l'environnement sous PyCharm. ....	69
<b>Figure 2.15:</b> Installation d'OpenCV par la commande pip. ....	69
<b>Figure 3.16:</b> Echantillon des images de classe A. ....	70
<b>Figure 3.17:</b> Echantillon des images de classe B.....	70
<b>Figure 3.18:</b> Partie de l'installation des bibliothèques dans Googl colab. ....	72
<b>Figure 3.19:</b> Code dans Google colab pour importer notre base de données.....	72
<b>Figure 3.20:</b> Taux de reconnaissance obtenu dans notre modèle. ....	73
<b>Figure 3.21:</b> Les résultats de l'évolution de taux de reconnaissance.....	74
<b>Figure 3.22:</b> Les résultats de l'évolution des pertes. ....	75

## Table des tableaux

<b>Tableau 2.1:</b> Comparaison entre l'apprentissage automatique et l'apprentissage en profondeur (Grossfeld, 2020).....	47
<b>Tableau 3.1:</b> les bibliothèques très populaire de Python. ....	62
<b>Tableau 3.2:</b> Partie de notre base de données en Excel.....	71

## Table des matières

Introduction générale .....	1
Chapitre 1 : Notion sur la détection des défauts et contrôle qualité .....	2
1 Introduction .....	3
2 Contrôle qualité .....	3
2.1 Définition de contrôle qualité .....	3
2.2 Importance de contrôle qualité .....	3
2.2.1 Satisfaction client .....	3
2.2.2 Réduction des couts .....	3
2.2.3 Amélioration de l'efficacité .....	4
2.2.4 Protection de la réputation .....	4
2.3 Les caractéristiques d'un contrôle qualité.....	4
2.4 Les types de contrôle .....	5
2.4.1 Contrôle destructif (CD).....	5
2.4.2 Contrôle non destructif (CND).....	5
3 Définition de la détection .....	5
4 Définition de défauts.....	5
5 La détection de défauts.....	5
5.1 Qu'est-ce qu'une détection de défauts ?.....	5
5.2 Méthodes de détections et localisation des défauts.....	6
5.2.1 Inspection visuelle .....	6
5.2.2 Méthodes de contrôle non destructif (CND).....	6
5.2.3 Méthodes d'analyse des vibrations .....	6
5.2.4 Analyse acoustique .....	6
5.2.5 Analyse d'images .....	6
5.3 Types de défauts .....	7
5.3.1 Les défauts de fabrication .....	7
5.3.2 Les d'écarts de matériaux .....	7
5.3.3 Les défauts de conception.....	7
5.3.4 Les défauts de performance .....	8
5.3.5 Les défauts de qualité.....	8
5.3.6 Les défauts de sécurité .....	8
5.3.7 Les défauts physiques .....	8
5.3.8 Les défauts électriques .....	9
5.3.9 Les défauts de logiciel.....	9
5.3.10 Les défauts de communication .....	9
5.3.11 Les défauts environnementaux .....	9
6 Système de triage industriel.....	9
6.1 Définition de Système de triage industriel.....	9
6.2 Types de systèmes de triage industriel .....	9
6.2.1 Systèmes de triage automatisés .....	9
6.2.2 Systèmes de triage par poids .....	10
6.2.3 Systèmes de triage par couleur .....	11
6.2.4 Systèmes de triage par forme .....	12
6.2.5 Systèmes de triage par caractéristiques spécifiques.....	12
6.2.6 Systèmes de triage par intelligence artificielle .....	13
6.3 L'impact de systèmes de triage industriel sur l'industrie .....	14
7 Les images.....	15
7.1 Définition d'image .....	15
7.2 Caractéristiques d'une image numérique .....	15

7.2.1	Résolution .....	15
7.2.2	Format de fichier .....	15
7.2.3	Profondeur de couleur .....	15
7.2.4	Taille du fichier .....	15
7.2.5	Compression.....	15
7.2.6	Métadonnées .....	16
7.2.7	Transparence.....	16
7.3	Acquisition d'une image.....	16
7.4	Outils d'acquisition des images .....	16
7.4.1	Appareils photo .....	16
7.4.2	Smartphones.....	16
7.4.3	Scanner.....	17
7.4.4	Caméras de surveillance .....	17
7.4.5	Logiciels de capture d'écran.....	17
7.4.6	Drones .....	17
7.4.7	Webcams .....	17
7.5	Types d'images numériques.....	17
7.5.1	Les images bitmap ou raster.....	17
7.5.2	Les images vectorielles .....	17
7.5.3	Les images en niveaux de gris.....	17
7.5.4	Les images en couleur indexée.....	17
7.5.5	Les images en couleurs réelles .....	17
7.6	Traitement d'image .....	18
7.7	Avantages des traitements d'images .....	18
7.7.1	L'amélioration de la qualité des images.....	18
7.7.2	La correction des défauts.....	18
7.7.3	L'extraction d'information .....	18
7.7.4	La création d'effets visuels .....	18
7.7.5	L'automatisation des tâches.....	19
7.8	Prétraitement d'images .....	19
7.8.1	La normalisation des couleurs et de la luminosité.....	19
7.8.2	Le redimensionnement .....	19
7.8.3	La suppression du bruit .....	19
7.8.4	La correction de la distorsion.....	19
7.8.5	La segmentation.....	19
8	Classification des images .....	19
8.1	Définition de la classification d'image.....	19
8.2	La motivation de la classification .....	20
8.3	Différentes méthodes de classification.....	20
8.3.1	La classification supervisée .....	20
8.3.2	La classification non supervisée.....	20
8.3.3	Les réseaux de neurones convolutifs (CNN).....	20
8.3.4	La classification basée sur les caractéristiques.....	20
8.3.5	La classification en cascade.....	20
8.3.6	La classification hybride.....	20
9	Etat de l'art.....	20
10	Conclusion .....	21
Chapitre 2 : Apprentissage automatique & Apprentissage profond .....		23
1	Introduction .....	24
2	Qu'est-ce qu'une intelligence artificielle ?.....	24
2.1	Avantages de l'intelligence artificielle .....	25

2.2	Inconvénients de l'intelligence artificielle .....	25
2.3	L'impact de l'intelligence artificielle dans l'industrie .....	26
3	L'apprentissage automatique .....	26
3.1	Définition d'apprentissage automatique .....	26
3.2	Types de systèmes d'apprentissage automatique .....	27
3.2.1	Apprentissage supervisé .....	27
3.2.2	Apprentissage non supervisé .....	33
3.2.3	Apprentissage semi-supervisé .....	35
3.2.4	Apprentissage par renforcement .....	35
3.2.5	Apprentissage profond .....	36
3.3	Le choix d'un type d'apprentissage automatique .....	36
4	De l'apprentissage automatique à l'apprentissage en profondeur .....	37
5	l'apprentissage profondeur .....	38
5.1	Définition d'apprentissage en profondeur .....	38
5.2	Réseau de neurones artificiels (ANN) .....	38
5.3	Réseaux de neurones à propagation avant (FFNN) .....	39
5.3.1	Pourquoi les réseaux de neurones à propagation avant (FFNN) ? .....	39
5.3.2	Types de réseaux de neurones à propagation avant (FFNN) .....	39
5.4	Réseaux de neurones récurrents (RNN) .....	40
5.4.1	Pourquoi les réseaux de neurones récurrents ? .....	41
5.4.2	Types de réseau de neurones récurrents .....	41
5.5	Réseaux de neurones convolutionnels (CNN) .....	42
5.5.1	Pourquoi les réseaux de neurones convolutionnels ? .....	42
5.5.2	Types de réseaux de neurones convolutionnels .....	43
5.6	Réseaux de neurones modèles adversaires génératifs (GAN) .....	43
5.6.1	Pourquoi les réseaux de neurones modèles adversaires génératifs ? .....	44
5.6.2	Types de réseaux de neurones modèles adversaires génératifs .....	44
6	le problème de sur-apprentissage et sous-apprentissage .....	45
6.1	Le sur-apprentissage .....	46
6.2	Le sous-apprentissage .....	46
7	Comparaison entre l'apprentissage automatique et l'apprentissage en profondeur .....	47
8	Conclusion .....	48
	Chapitre 3 : Conception et Implémentation .....	49
1	Introduction .....	50
2	Concept générale .....	50
2.1	Système d'acquisition .....	52
2.2	Prétraitement .....	52
2.2.1	Segmentation d'image .....	53
2.2.2	Analyse d'image .....	53
2.3	Entraînement .....	56
3	Les couches utilisées dans notre programme .....	58
4	Application et outils d'implémentation .....	58
4.1	Python .....	59
4.1.1	Quelques faits sur le langage de programmation python .....	62
4.1.2	Pourquoi en utilisant python ? .....	62
4.2	Google Colab .....	63
4.3	Open CV .....	64
4.4	PyCharm .....	65
5	Résultats et discussion .....	66
5.1	Base de données .....	66
5.2	Schéma de base de données .....	66

5.3	Configuration matériel et immatériel utilisé dans l'implémentation.....	67
5.3.1	Environnement de matériel : .....	67
5.3.2	Environnement immatériel .....	67
5.4	Les base des images .....	69
6	Préparation de l'environnement .....	71
6.1	Installation des bibliothèques .....	71
6.2	Importer les données .....	72
7	Visualisation des résultats .....	73
7.1	Taux de reconnaissance .....	73
7.2	Evolution de taux de reconnaissance.....	74
8	Conclusion .....	76
	Conclusion générale.....	77
9	Bibliographie .....	79

# Introduction générale

Le Deep Learning est une technique d'apprentissage automatique qui permet aux ordinateurs de reconnaître des modèles complexes et de résoudre des problèmes difficiles. Cette méthode est de plus en plus utilisée dans l'industrie pour améliorer les processus, réduire les coûts et améliorer la qualité des produits. Les applications du Deep Learning dans l'industrie incluent la reconnaissance d'image et de vidéo, le traitement du langage naturel, les prévisions et les prédictions, ainsi que l'automatisation de la production.

L'apprentissage en profondeur a un impact important sur l'industrie en offrant des avantages tels que l'amélioration de la qualité des produits, l'automatisation des tâches, l'optimisation de la chaîne d'approvisionnement et le développement de nouveaux produits. Grâce à cette technologie, les entreprises peuvent détecter rapidement les défauts dans les produits et les processus, automatiser les tâches répétitives et complexes, prédire la demande et développer des produits mieux adaptés aux besoins des clients. Tout cela contribue à améliorer l'efficacité, la performance et la compétitivité des entreprises dans divers domaines.

Notre travail consiste à utiliser l'apprentissage en profondeur (Deep Learning) pour classer des images d'emballages alimentaires en utilisant des réseaux de neurones artificiels convolutionnels. Il s'agit d'une technique d'apprentissage qui permet à un programme de reconnaître le contenu d'une image. Pour cela, nous allons utiliser différents modèles avec différentes architectures pour classer les images des boîtes d'emballages plastiques biodégradables alimentaires, certaines prises à partir d'une caméra. Nous allons appliquer ces modèles sur un dataset<sup>1</sup> qui est un ensemble d'images de boîtes d'emballages de différentes formes. Ce dataset<sup>1</sup> est divisé en deux classes : la classe A qui correspond aux images sans défauts et la classe B qui correspond aux images présentant des défauts. L'objectif est de créer un modèle qui soit capable de classer correctement les images de boîtes d'emballages en fonction de leur classe. Cela peut être utilisé pour des applications telles que le contrôle qualité dans l'industrie alimentaire.

Notre mémoire est organisé en 3 chapitres encadrer par une introduction générale et clôturer par une conclusion. Dans le premier chapitre, nous présentons les notions sur la détection des défauts et contrôle qualité, puis nous présentons les notion relative a la manipulation des images, plus précisément nous parlons de comment examiner les images, quelles sont leur caractéristique, leur acquisition et leur traitement et les quelle sont les méthodes de classification les plus appropriées , à la fin nous allons discuter quelques travaux connexes. À travers le deuxième chapitre, nous allons fournir un regard sur le domaine de l'intelligence artificielle et ses différentes branches « l'apprentissage automatique et l'apprentissage en profondeur », en définissant ce domaine, ces branches et ses différentes caractéristiques. Le troisième chapitre traite la conception et l'implémentation de notre système et ce chapitre se compose de trois parties : Nous allons d'abord, décrire le concept général de système proposé, par la suite, nous allons montrer les applications et les outils d'implémentation et en troisième point, nous expliquerons toutes les expériences que nous avons appliquées aux méthodes proposées et les résultats obtenus.

# **Chapitre 1**

## **Notion sur la détection des défauts et contrôle qualité**

### 1 Introduction

Le contrôle qualité et la détection de défauts sont des domaines en constante évolution, avec des avancées récentes dans l'utilisation de l'IA et de l'apprentissage automatique. L'état de l'art actuellement dans ce domaine est caractérisé par l'utilisation de techniques d'apprentissage profond pour améliorer la détection de défauts. Les réseaux de neurones convolutionnels (CNN) sont largement utilisés pour analyser les images et les vidéos, permettant aux systèmes de détecter des défauts tels que les fissures, les rayures, les bosses, etc. les systèmes de vision par ordinateur utilisant l'apprentissage profond ont montré des résultats prometteurs dans diverses industries. Il est important de noter que l'IA n'est pas la seule solution pour la détection de défauts, il existe d'autres techniques mais les résultats obtenus avec l'IA sont souvent plus précis et plus efficaces. En somme, l'IA et l'apprentissage automatique sont des outils clés pour le contrôle qualité et la détection de défauts. Les systèmes basés sur l'IA ont montré des résultats prometteurs et ces technologies continuent de s'améliorer.

### 2 Contrôle qualité

#### 2.1 Définition de contrôle qualité

Le contrôle de qualité est un ensemble de procédures et de méthodes qui visent à assurer que les produits ou services répondent aux spécifications établies et aux exigences des clients. Il vise également à identifier et corriger les écarts ou les défauts dans les produits ou les services avant qu'ils ne soient livrés aux clients. Le contrôle de qualité peut inclure des inspections, des tests, des mesures et des analyses pour évaluer la conformité des produits ou des services aux normes établies. Il peut également inclure des audits internes et des enquêtes pour évaluer les processus et les procédures de l'entreprise afin de s'assurer qu'ils sont efficaces et conformes aux normes de qualité. (Crosby, 1979).

#### 2.2 Importance de contrôle qualité

Le contrôle de qualité est crucial car il assure que les produits et services offerts sont conformes aux normes et aux exigences requises. Il permet également de repérer et corriger les erreurs avant qu'elles ne causent des dommages ou ne reçoivent les clients. En outre, il contribue à l'amélioration des processus de production et augmente la satisfaction des clients, ce qui peut avoir un impact positif sur les ventes et la réputation de l'entreprise. Voici quelques raisons pour lesquelles le contrôle qualité est si important :

##### 2.2.1 Satisfaction client

Le contrôle qualité aide à garantir que les produits ou services répondent aux attentes des clients. En fournissant des produits de haute qualité, les entreprises peuvent améliorer la satisfaction de leurs clients, ce qui peut se traduire par une meilleure fidélisation et une augmentation des ventes.

##### 2.2.2 Réduction des coûts

Le contrôle qualité permet de réduire les coûts en identifiant les problèmes de qualité plus tôt dans le processus de production. Cela peut aider à éviter les coûts liés aux retours de produits défectueux, aux réparations et aux rappels de produits.

### 2.2.3 Amélioration de l'efficacité

Le contrôle qualité permet d'améliorer l'efficacité en identifiant les problèmes de qualité et en fournissant des données précises pour orienter les efforts d'amélioration. Cela peut aider à optimiser les processus de production et à améliorer la qualité globale des produits.

### 2.2.4 Protection de la réputation

La réputation d'une entreprise est directement influencée par la qualité de ses produits et services. En offrant des produits de haute qualité, une entreprise peut renforcer sa réputation et sa crédibilité auprès de sa clientèle, ce qui est essentiel pour maintenir sa position concurrentielle sur le marché. Le contrôle qualité joue un rôle crucial dans la gestion efficace d'une entreprise, car il garantit la conformité et la fiabilité des produits et services. En assurant la qualité, les entreprises peuvent améliorer la satisfaction de leurs clients, réduire les coûts liés aux défauts et aux retours, accroître l'efficacité de leurs processus de production et protéger leur réputation à long terme.

## 2.3 Les caractéristiques d'un contrôle qualité

Le contrôle qualité est un processus essentiel pour garantir la conformité des produits ou services aux normes et aux attentes des clients. Les caractéristiques de contrôle qualité sont des éléments importants qui permettent d'évaluer la qualité d'un produit ou d'un service. Bien que ces caractéristiques puissent varier selon le secteur d'activité et les exigences des clients, il existe des caractéristiques courantes qui sont applicables dans de nombreux domaines. (Ishikawa, 1985).

Parmi les caractéristiques de contrôle qualité les plus fréquentes, on trouve :

- **La fiabilité** : mesure la capacité d'un produit ou d'un service à fonctionner de manière cohérente sans défaillance.
- **La durabilité** : mesure la capacité d'un produit à résister à l'usure et à l'usage quotidien sans se détériorer.
- **La conformité** : mesure la capacité d'un produit ou d'un service à respecter les normes et les réglementations en vigueur.
- **La sécurité** : mesure la capacité d'un produit ou d'un service à ne pas présenter de risques pour la santé ou la sécurité des utilisateurs.
- **La facilité d'utilisation** : mesure la capacité d'un produit ou d'un service à être utilisé facilement et efficacement par les utilisateurs.
- **La satisfaction du client** : mesure la capacité d'un produit ou d'un service à répondre aux attentes et aux besoins des clients.
- **La précision** : capacité à maintenir une précision stable dans la sortie.
- **La répétabilité** : capacité à produire des résultats constants pour des entrées données.
- **La régularité** : capacité à produire des résultats réguliers pour des entrées données.
- **La prévisibilité** : capacité à prévoir les résultats pour des entrées données.
- **Maintenance** : facilité à maintenir et à réparer le système.
- **La rapidité** : le système de contrôle de qualité doit être capable de détecter rapidement les écarts de qualité pour minimiser les retards et les coûts supplémentaires.

En veillant à la qualité de chacune de ces caractéristiques, les entreprises peuvent améliorer la satisfaction de leurs clients et renforcer leur réputation sur le marché. Les caractéristiques de contrôle qualité sont donc un élément clé de processus de contrôle qualité.

### **2.4 Les types de contrôle**

#### **2.4.1 Contrôle destructif (CD)**

Le contrôle destructif est une technique d'évaluation qui consiste à détériorer volontairement un produit ou un matériau afin de vérifier sa qualité ou sa conformité. Il est utilisé pour évaluer la solidité, la robustesse et la fiabilité d'un produit. Cependant, cette méthode est coûteuse et ne permet pas de réutiliser le produit testé. Il est principalement utilisé dans des situations où le produit peut causer des dommages corporels ou matériels en cas de défaillances.

#### **2.4.2 Contrôle non destructif (CND)**

Le contrôle non destructif est une méthode d'évaluation qui permet de vérifier la qualité ou la conformité d'un produit ou d'un matériau sans l'endommager. Il existe différentes techniques de contrôles non destructifs tels que l'inspection visuelle, la radiographie, l'ultrason, la magnétoscopie, et la tomographie. Ces techniques permettent de détecter des défauts ou des anomalies sans altérer le produit ou le matériau. Le contrôle non destructif est largement utilisé dans l'industrie pour la maintenance préventive, la sécurité des structures et la qualité des produits.

### **3 Définition de la détection**

La détection est la capacité de repérer la présence d'un signal, d'une information ou d'un élément spécifique parmi un ensemble de données. Cela peut inclure la détection de défauts dans les produits, la détection de la présence de substances dangereuses ou la détection de cibles dans les images ou les vidéos. La détection peut être effectuée de manière manuelle ou automatisée, en utilisant des techniques telles que la vision par ordinateur, l'analyse d'images, l'apprentissage automatique et la reconnaissance de formes. La détection est souvent utilisée dans des domaines tels que la sécurité, la robotique, la médecine, les sciences de la vie et les technologies de l'information.

### **4 Définition de défauts**

Un défaut est une imperfection ou une irrégularité dans un produit ou un système qui en réduit la qualité ou l'utilité. Les défauts peuvent être de différentes natures, comme des défauts de fabrication, des défauts de matériaux, des défauts de conception, des défauts de performance, etc. les défauts peuvent entraîner des problèmes de sécurité, des coûts supplémentaires pour les réparations ou les remplacements, ou des réclamations de la part des clients. Il est donc important de les détecter et de les corriger de manière efficace pour maintenir la qualité et la fiabilité des produits ou systèmes. (Cheng, Defect detection of glass surface based on deep learning, 2020)

### **5 La détection de défauts**

#### **5.1 Qu'est-ce qu'une détection de défauts ?**

La détection de défauts est un processus crucial qui permet d'identifier la présence de défauts ou d'anomalies dans un système ou une pièce. Ce processus est largement utilisé pour garantir

la fiabilité et la sécurité des équipements, notamment dans les domaines de la maintenance et de la réparation. Les défauts peuvent être de différentes natures, tels que des fissures, des désalignements, des fuites ou des dysfonctionnements, et peuvent être détectés en utilisant diverses techniques, comme l'inspection visuelle, les méthodes de contrôle non destructif, l'analyse des vibrations et acoustique, et l'analyse d'images. Une fois qu'un défaut est détecté, il est essentiel de le localiser avec précision afin de prendre les mesures appropriées pour corriger le problème. En résumé, la détection de défauts est un élément clé pour assurer la sécurité et la fiabilité des systèmes et des équipements.

### 5.2 Méthodes de détections et localisation des défauts

Il existe différentes méthodes de détection et de localisation des défauts, qui varient en fonction du type de défauts à détecter et des données disponibles. Voici quelques-unes des méthodes couramment utilisées :

#### 5.2.1 Inspection visuelle

Il s'agit de la méthode la plus simple et la plus courante, qui consiste à inspecter visuellement les pièces pour détecter tout défaut apparent.

#### 5.2.2 Méthodes de contrôle non destructif (CND)

Les méthodes de CND comprennent l'utilisation de techniques telles que la radiographie, la thermographie, l'électromagnétisme, la magnétoscopie et la liquéfaction pénétrante pour détecter les défauts internes.

#### 5.2.3 Méthodes d'analyse des vibrations

Cette méthode utilise des capteurs pour mesurer les vibrations d'une machine ou d'un système, ce qui permet de détecter les défauts tels que les fissures ou les désalignements.

#### 5.2.4 Analyse acoustique

Cette méthode utilise des microphones pour mesurer les sons émis par un système, ce qui permet de détecter les défauts tels que les fuites ou les blocages.

#### 5.2.5 Analyse d'images

Cette méthode utilise des algorithmes de traitement d'images pour détecter les défauts visuels, tels que les rayures ou les marques.

Une fois qu'un défaut est détecté, il est important de le localiser avec précision, pour ce faire, les méthodes suivantes sont souvent utilisées :

#### ✓ Inspection par ultrasons

Cette méthode utilise des ondes sonores à haute fréquence pour localiser la position exacte d'un défaut.

#### ✓ Méthodes de tomographie

Ces méthodes utilisent des techniques telles que la tomographie par rayon X ou la tomographie électrique pour créer des images tridimensionnelles du système, ce qui permet de localiser avec précision les défauts internes.

### ✓ Analyse thermique

Cette méthode utilise des caméras thermiques pour détecter les variations de température, ce qui peut aider à localiser les défauts tels que les fissures.

### ✓ Analyse de fréquence

Cette méthode utilise des capteurs pour mesurer les fréquences des vibrations et peut aider à localiser les défauts tels que les déséquilibres ou les vibrations anormales.

En utilisant ces méthodes de détection et de localisation, il est possible de détecter et de localiser rapidement et précisément les défauts dans un système.

## 5.3 Types de défauts

Il existe différents types de défauts qui peuvent être présents dans un produit ou un système, notamment :

### 5.3.1 Les défauts de fabrication

Ces défauts sont causés par des erreurs ou des problèmes lors de la fabrication d'un produit, tels que des erreurs de mesure, des erreurs de coupe, des erreurs de soudages, etc. (Voir figure 1.1).



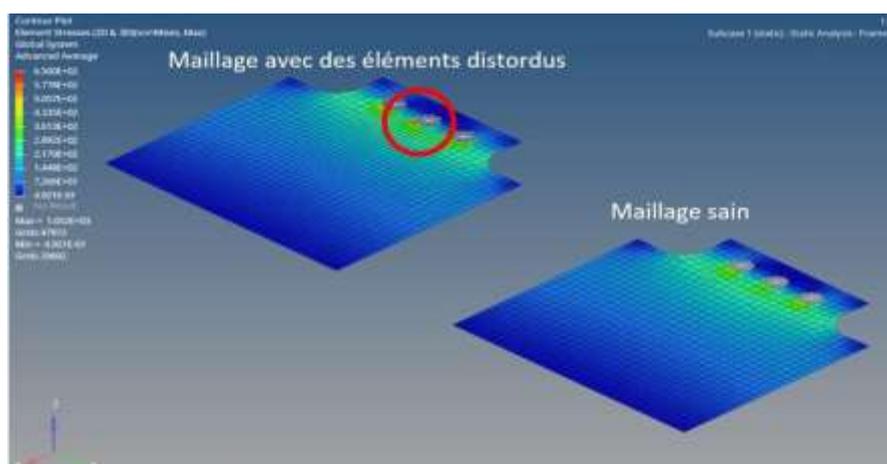
**Figure 1.1:** Les erreurs de coupe d'une boîte de ravier en polystyrène.

### 5.3.2 Les écarts de matériaux

Ces défauts sont causés par des problèmes avec les matériaux utilisés pour fabriquer un produit, tels que des matériaux de mauvaise qualité, des matériaux inadéquats ou des matériaux mal utilisés.

### 5.3.3 Les défauts de conception

Ces défauts sont causés par des erreurs ou des omissions dans la conception d'un produit ou d'un système, tels que des erreurs de calcul, des erreurs de spécifications, des erreurs de dimensionnement, etc. (Voir figure 1.2).



**Figure 1.2:** Les défauts sur le calcul numérique de l'importance du maillage.

### 5.3.4 Les défauts de performance

Ces défauts sont causés par des problèmes avec les performances d'un produit ou d'un système, tels que des problèmes de fiabilité, des problèmes de durabilité, des problèmes de sécurité, etc.

### 5.3.5 Les défauts de qualité

Ces défauts sont causés par des problèmes avec la qualité d'un produit ou d'un système, tels que des problèmes de conformité, des problèmes de fiabilité, des problèmes de durabilité, etc.

### 5.3.6 Les défauts de sécurité

Ces défauts sont causés par des problèmes avec la sécurité d'un produit ou d'un système, tels que des risques d'incendie, des risques d'électrocution, des risques de blessure, etc.

### 5.3.7 Les défauts physiques

Ce sont des altérations physiques de la matière, tels que des fissures, des éclats, des rayures, des bosses, des déformations, des plis, des trous, etc. (Voir figure 1.3).



**Figure 1.3:** La déformation d'une boîte de ravier en polystyrène.

### 5.3.8 Les défauts électriques

Ce sont des erreurs électriques, tels que des courts circuits, des surtensions, des sous tensions, des fuites de courant, des pannes de circuit, etc.

### 5.3.9 Les défauts de logiciel

Ce sont des erreurs dans le code informatique, tels que des bogues, des erreurs de syntaxe, des erreurs de logique, des erreurs de calcul, etc.

### 5.3.10 Les défauts de communication

Ce sont des erreurs dans la transmission des informations, tels que des erreurs de transmission, des erreurs de réception, des erreurs de compréhension, des erreurs d'interprétation, etc.

### 5.3.11 Les défauts environnementaux

Ce sont des erreurs causées par l'environnement, tels que des températures extrêmes, des vibrations, des chocs, des variations de pression, des variations de l'humidité, des expositions aux rayons UV, etc.

## 6 Système de triage industriel

### 6.1 Définition de Système de triage industriel

Un système de triage industriel est un ensemble de technologie et de machines utilisées pour classer et séparer différents types de matériaux ou de produits dans un environnement industriel. Il peut inclure des équipements tels que des convoyeurs, des séparateurs à air, des détecteurs de métaux, des caméras de vision par ordinateurs, des capteurs de poids et de dimensions, et des logiciels de contrôle automatisés. Il est utilisé dans divers secteurs tels que l'automobile, l'emballage alimentaire, la logistique, la récupération de matières et la fabrication électronique.

### 6.2 Types de systèmes de triage industriel

Il existe différents types de systèmes de triage industriel qui sont adaptés à des besoins spécifiques. Quelques-uns des systèmes les plus utilisés sont :

#### 6.2.1 Systèmes de triage automatisés

Les systèmes de triage automatisés sont des solutions technologiques qui permettent de trier rapidement et efficacement différents types d'objets, de matériaux ou d'informations sans intervention humaine directe. Ils utilisent des technologies avancées telles que la vision par ordinateur, les capteurs et les algorithmes de traitement des données pour réaliser le processus de tri de manière automatisée. Le fonctionnement des systèmes de triage automatisés se divise généralement en plusieurs étapes :

1. **Acquisition des données** : les objets à trier sont placés sur un tapis roulant ou un système de convoyage où des capteurs et des caméras recueillent des informations sur ces objets, comme leur taille, leur forme, leur couleur, etc.
2. **Analyse et reconnaissance** : Les données capturées sont traitées par des algorithmes de vision par ordinateur qui analysent les caractéristiques des objets. Ces algorithmes peuvent utiliser des techniques d'apprentissage automatique pour reconnaître et classer les objets en fonction de modèles préalablement définis.

3. **Prise de décision** : Une fois l'analyse effectuée, le système prend des décisions sur la destination appropriée pour chaque objet, en se basant sur des critères prédéfinis tels que le type d'objet, la qualité ou le poids
4. **Action de tri** : Après la prise de décision, le système met en œuvre l'action de tri en utilisant des mécanismes appropriés. Cela peut impliquer des bras robotisés, des jets d'air, des plaques basculantes, des courroies de tri, ou d'autres mécanismes similaires qui déplacent les objets vers les destinations appropriées en fonction des décisions prises précédemment.

Les systèmes de triage automatisés offrent plusieurs avantages :

- **Rapidité et efficacité** : Ils peuvent trier un grand nombre d'objets en peu de temps, ce qui permet d'augmenter la productivité et de réduire les délais de traitement.
- **Précision** : Les algorithmes de vision par ordinateur et les capteurs avancés permettent une reconnaissance précise des objets, minimisant ainsi les erreurs de tri.
- **Flexibilité** : Les systèmes de triage automatisés peuvent être configurés et adaptés pour trier différents types d'objets, ce qui les rend polyvalents et adaptés à différentes applications.
- **Réduction des coûts** : Ils permettent de réduire les coûts de main-d'œuvre liés au tri manuel, tout en augmentant la productivité globale du processus de tri.

Ils sont utilisés dans divers secteurs tels que la logistique, la production industrielle, le recyclage, la distribution de colis et l'industrie alimentaire, où un tri rapide et précis est nécessaire pour gérer efficacement les flux d'objets ou de matériaux.

### 6.2.2 Systèmes de triage par poids

Les systèmes de triage par poids sont des solutions automatisées utilisées pour classer les objets en fonction de leur poids. Ils sont largement employés dans divers secteurs tels que la logistique, la production industrielle, l'industrie alimentaire et le recyclage. Ces systèmes permettent un tri rapide et précis en se basant sur la mesure du poids des objets, simplifiant ainsi le processus de tri et de distribution. Le fonctionnement des systèmes de triage par poids implique plusieurs composants essentiels :

1. **Capteurs de poids** : Ces capteurs sont utilisés pour mesurer le poids des objets à trier. Ils peuvent utiliser des technologies telles que les cellules de charge, les balances ou les systèmes de pesée dynamique.
2. **Mécanismes de tri** : Une fois que le poids d'un objet est mesuré, le système utilise des mécanismes de tri pour diriger cet objet vers la destination appropriée en fonction de sa catégorie de poids prédéfinie. Ces mécanismes peuvent comprendre des bras robotisés, des systèmes de convoyage avec des pentes ajustables, des courroies de tri ou des plaques basculantes.
3. **Algorithme de décision** : Un algorithme est utilisé pour déterminer la catégorie de poids à laquelle chaque objet appartient et pour prendre la décision de tri correspondante. Cet algorithme est généralement configuré selon les spécifications du système de triage et peut être adapté en fonction des besoins spécifiques de l'application.

4. **Système de contrôle** : Un système de contrôle coordonne l'ensemble du processus de triage en recevant les informations des capteurs de poids, en effectuant les calculs nécessaires, en activant les mécanismes de tri et en assurant la communication entre les différentes parties du système.

Les systèmes de triage par poids offrent plusieurs avantages, tels qu'une classification précise des objets en fonction de leur poids, une exécution rapide, une réduction des erreurs de tri et une automatisation du processus, ce qui permet de gagner du temps et de réduire les coûts de main-d'œuvre.

Ces systèmes sont utilisés dans diverses applications, notamment le tri des colis en fonction de leur poids pour la livraison, le tri des produits alimentaires en fonction de leur poids pour l'emballage ou le tri des matériaux recyclables en fonction de leur poids pour un traitement adéquat.

### 6.2.3 Systèmes de triage par couleur

Les systèmes de triage par couleur sont des solutions technologiques utilisées pour classer et trier des objets en fonction de leur teinte ou de leur couleur. Ils sont largement utilisés dans différents domaines tels que l'industrie alimentaire, la fabrication, l'automatisation des processus et le tri des déchets.

Le fonctionnement des systèmes de triage par couleur comprend les éléments suivants :

1. **Capteurs de couleur** : Ces capteurs sont utilisés pour détecter et analyser les différentes nuances de couleur des objets à trier. Ils peuvent utiliser des technologies telles que la spectroscopie ou la vision par ordinateur pour mesurer avec précision les couleurs.
2. **Base de données de référence** : Les systèmes de triage par couleur utilisent une base de données préétablie contenant des informations sur les couleurs et les catégories associées. Cette base de données permet de comparer les couleurs détectées par les capteurs avec les références et de prendre des décisions de tri en conséquence.
3. **Algorithme de tri** : Les données capturées par les capteurs de couleur sont analysées à l'aide d'algorithmes spécifiques pour identifier la couleur de chaque objet. Ces algorithmes peuvent être configurés pour classer les objets en différentes catégories de couleur ou pour prendre des décisions plus complexes basées sur des combinaisons de couleurs.
4. **Mécanismes de tri** : Une fois que la couleur d'un objet est identifiée, des mécanismes de tri sont utilisés pour diriger l'objet vers la destination appropriée en fonction de sa catégorie de couleur prédéfinie. Ces mécanismes peuvent inclure des bras robotisés, des convoyeurs à voies multiples ou d'autres systèmes de tri mécaniques.

Les avantages des systèmes de triage par couleur incluent une classification précise des objets en fonction de leur couleur, une automatisation du processus de tri, une réduction des erreurs et une augmentation de la productivité. Ils permettent également de trier rapidement de grandes quantités d'objets en se basant uniquement sur la caractéristique de couleur.

Ces systèmes sont utilisés dans diverses applications, telles que le tri des aliments en fonction de leur maturité ou de leur qualité, le tri des produits manufacturés selon des normes de couleur spécifiques, le tri des déchets pour le recyclage, et bien d'autres.

### 6.2.4 Systèmes de triage par forme

Les systèmes de triage par forme sont des solutions technologiques utilisées pour classer et trier des objets en fonction de leur configuration géométrique. Ces systèmes automatisés sont couramment employés dans divers domaines tels que la logistique, la production industrielle, l'emballage et le tri des déchets.

Le fonctionnement des systèmes de triage par forme comprend les éléments suivants :

1. **Capteurs de forme** : Les capteurs de forme sont utilisés pour détecter et analyser la configuration géométrique des objets à trier. Ces capteurs peuvent utiliser différentes technologies, telles que la vision par ordinateur, les capteurs laser ou les capteurs tactiles, pour capturer les caractéristiques de forme des objets.
2. **Base de données de référence** : Les systèmes de triage par forme utilisent une base de données préétablie contenant des informations sur les formes et les catégories associées. Cette base de données permet de comparer les formes détectées par les capteurs avec les références et de prendre des décisions de tri en conséquence.
3. **Algorithme de tri** : Les données capturées par les capteurs de forme sont analysées à l'aide d'algorithmes spécifiques pour identifier la forme de chaque objet. Ces algorithmes peuvent être configurés pour classer les objets en différentes catégories de forme prédéfinies ou pour prendre des décisions plus complexes basées sur des caractéristiques de forme spécifiques.
4. **Mécanismes de tri** : Une fois que la forme d'un objet est identifiée, des mécanismes de tri sont utilisés pour diriger l'objet vers la destination appropriée en fonction de sa catégorie de forme. Ces mécanismes peuvent inclure des bras robotisés, des systèmes de convoyage avec des voies multiples ou des dispositifs de triage mécaniques spécifiques à la forme.

Les systèmes de triage par forme offrent plusieurs avantages, tels qu'une classification précise des objets en fonction de leur configuration géométrique, une automatisation du processus de tri, une réduction des erreurs et une augmentation de la productivité. Ils permettent également de trier rapidement et efficacement des objets de formes différentes sans nécessiter une intervention humaine directe. Ces systèmes sont utilisés dans diverses applications, comme le tri des pièces manufacturées, le tri des produits alimentaires basé sur leur forme, le tri des matériaux recyclables selon leur configuration spécifique, et bien d'autres.

### 6.2.5 Systèmes de triage par caractéristiques spécifiques

Les systèmes de triage par caractéristiques spécifiques sont des solutions technologiques utilisées pour classer et trier des objets en fonction de caractéristiques particulières autres que le poids, la couleur ou la forme. Ces systèmes automatisés sont largement utilisés dans divers domaines, tels que la fabrication, l'emballage, la logistique et le tri des matériaux.

Le fonctionnement des systèmes de triage par caractéristiques spécifiques implique les éléments suivants :

1. **Capteurs spécialisés** : Ces capteurs sont conçus pour détecter et mesurer des caractéristiques spécifiques des objets à trier, telles que la taille, la texture, la transparence, la conductivité électrique, la présence de marques ou de codes-barres, ou toute autre caractéristique pertinente pour le processus de tri.
2. **Base de données de référence** : Les systèmes de triage par caractéristiques spécifiques utilisent une base de données préétablie contenant des informations sur les caractéristiques spécifiques et les catégories associées. Cette base de données permet de comparer les données capturées par les capteurs avec les références et de prendre des décisions de tri en conséquence.
3. **Algorithme de tri** : Les données collectées par les capteurs spécialisés sont analysées à l'aide d'algorithmes spécifiques pour identifier les caractéristiques spécifiques de chaque objet. Ces algorithmes peuvent être configurés pour classer les objets en différentes catégories en fonction des caractéristiques détectées.
4. **Mécanismes de tri** : Une fois que les caractéristiques spécifiques d'un objet sont identifiées, des mécanismes de tri sont utilisés pour diriger l'objet vers la destination appropriée en fonction de sa catégorie de caractéristiques spécifiques. Ces mécanismes peuvent inclure des bras robotisés, des convoyeurs à voies multiples ou d'autres dispositifs de triage mécaniques adaptés aux caractéristiques spécifiques détectées.

Les systèmes de triage par caractéristiques spécifiques offrent plusieurs avantages, tels qu'une classification précise des objets en fonction de caractéristiques particulières, une automatisation du processus de tri, une réduction des erreurs et une amélioration de l'efficacité. Ils permettent également de trier rapidement et efficacement des objets en se basant sur des critères autres que le poids, la couleur ou la forme.

Ces systèmes sont utilisés dans une variété d'applications, telles que le tri des produits en fonction de leur taille ou de leur texture, le tri des objets en fonction de leur composition matérielle, le tri des produits en fonction de codes-barres ou d'étiquettes, et bien d'autres encore.

### 6.2.6 Systèmes de triage par intelligence artificielle

Les systèmes de triage par intelligence artificielle sont des solutions technologiques qui utilisent des algorithmes et des modèles d'apprentissage automatique pour classer et trier automatiquement des objets. Grâce à l'intelligence artificielle, ces systèmes sont capables d'effectuer des tâches de tri complexes et adaptables.

Le fonctionnement des systèmes de triage par intelligence artificielle se décompose en plusieurs étapes :

1. **Entraînement du modèle** : Les systèmes de triage par intelligence artificielle sont nourris avec de grandes quantités de données d'entraînement, qui comprennent des exemples d'objets déjà triés. Ces données sont utilisées pour entraîner un modèle d'apprentissage automatique qui peut apprendre à reconnaître les caractéristiques et les motifs des objets à trier.

2. **Extraction des caractéristiques** : Les modèles d'apprentissage automatique extraient automatiquement les caractéristiques significatives des objets à partir des données d'entrée, qu'il s'agisse d'images, de données sensorielles ou d'autres types de données. Ces caractéristiques permettent au modèle de comprendre les différences et les similitudes entre les objets.
3. **Classification et tri** : Une fois que le modèle est entraîné, il peut prendre des décisions de tri en fonction des caractéristiques détectées. Le modèle peut classifier les objets en différentes catégories prédéfinies ou créer ses propres catégories en fonction des données d'entraînement. Les objets sont ensuite dirigés vers les destinations appropriées en fonction de leur classe attribuée.
4. **Amélioration continue** : Les systèmes de triage par intelligence artificielle sont adaptatifs et peuvent être améliorés en continu. Ils peuvent être ré-entraînés avec de nouvelles données pour améliorer leur précision et leur capacité à trier des objets avec fiabilité.

Les systèmes de triage par intelligence artificielle offrent plusieurs avantages, notamment une grande précision dans la classification et le tri des objets, la capacité à traiter des données complexes et diverses, l'automatisation complète du processus de tri et la flexibilité pour s'adapter à différents types d'objets et de scénarios de tri.

Ces systèmes sont utilisés dans diverses applications, comme le tri des produits manufacturés, le tri des colis dans les centres de distribution, le tri des déchets pour le recyclage, le tri des documents, et bien d'autres. Ils permettent d'optimiser les opérations de tri, de réduire les erreurs et d'augmenter l'efficacité globale du processus de tri.

### 6.3 L'impact de systèmes de triage industriel sur l'industrie

Les systèmes de triage industriel ont des conséquences positives sur l'industrie en augmentant l'efficacité, la productivité et la qualité des produits finaux. Certains de ces impacts peuvent être :

- **Amélioration de l'efficacité** : les systèmes de triage automatisés peuvent trier les matériaux plus rapidement et plus précisément que les méthodes manuelles, réduisant les temps de production et les coûts.
- **Augmentation de la qualité des produits** : les systèmes de triage automatisés peuvent détecter et éliminer les matériaux défectueux ou non conformes, ce qui permet de garantir une meilleure qualité des produits finaux.
- **Réduction des coûts** : les systèmes de triage automatisés peuvent réduire les coûts en éliminant les erreurs humaines et en augmentant l'efficacité de la production.
- **Optimisation des ressources** : les systèmes de triage automatisés permettent une meilleure utilisation des matériaux en les triant efficacement et en réduisant les pertes.
- **Meilleure traçabilité** : les systèmes de triage automatisés peuvent aider à suivre les matériaux à chaque étape du processus de production, ce qui permet une meilleure traçabilité des produits.

En résumé, les systèmes de triage industriel ont un impact positif sur l'industrie en améliorant l'efficacité, la qualité et la rentabilité de la production moderne.

### 7 Les images

#### 7.1 Définition d'image

Une image peut être définie comme une représentation visuelle de quelque chose, que ce soit un objet, une personne, un paysage ou une abstraction. Elle peut être créée à partir de divers médias, comme la peinture, la photographie, la sculpture, la gravure, la vidéo, etc.

En informatique, une image est une représentation numérique d'une image physique ou virtuelle, qui peut être stockée sous forme de fichier image et être modifiée à l'aide de logiciels de traitement d'image. En d'autres termes, dans une image numérique, les pixels enregistrent la valeur moyenne des zones capturées, ce qui signifie qu'il n'y a pas de continuité mathématique entre les objets dans l'image. Les pixels représentent des échantillons discrets de l'image et ne sont pas reliés par des relations mathématiques continues. Par conséquent, chaque pixel est considéré comme une unité indépendante dans l'image, et les objets dans l'image sont définis par des regroupements de pixels qui partagent des caractéristiques similaires, tels que la couleur, la texture ou la forme.

Le terme « image » peut également faire référence à une représentation mentale d'un objet, d'un événement ou d'une idée formée à partir de l'expérience sensorielle ou de l'imagination, qui peut avoir un impact sur la perception, la pensée et le comportement des individus.

#### 7.2 Caractéristiques d'une image numérique

Une image numérique possède plusieurs caractéristiques qui la définissent par les paramètres suivants :

##### 7.2.1 Résolution

La résolution d'une image numérique dépend du nombre de pixels qui la composent, ce qui détermine sa netteté et son niveau de détail.

##### 7.2.2 Format de fichier

Le format de fichier d'une image numérique utilisé pour stocker l'image influence la manière dont les données sont organisées, avec des formats courants comme JPEG, PNG, GIF et BMP.

##### 7.2.3 Profondeur de couleur

La profondeur de couleur d'une image numérique est également une caractéristique importante, déterminant le nombre de bits utilisés pour chaque pixel, ce qui affecte la gamme de couleurs disponibles et la précision de l'image.

##### 7.2.4 Taille du fichier

La taille du fichier d'une image numérique est influencée par la résolution, le format de fichier et la profondeur de couleur, avec des images de haute qualité ayant tendance à être plus grandes.

##### 7.2.5 Compression

La compression de l'image numérique est souvent utilisée pour réduire la taille des fichiers d'images sans altérer significativement leur qualité. Les formats de compression les plus courants sont JPEG et PNG.

### 7.2.6 Métadonnées

Les métadonnées peuvent être incluses dans une image numérique, fournissant des informations telles que la date de création, les informations sur l'appareil photo, ou encore la localisation, etc.

### 7.2.7 Transparence

Certaines images numériques peuvent être partiellement ou totalement transparentes, ce qui permet de les superposer sur d'autres éléments sans affecter leur visibilité.

## 7.3 Acquisition d'une image

L'objectif principal de l'acquisition d'images est de créer des images numériques à partir de sources physiques ou virtuelles pour stocker et traiter les images à des fins variées, comme l'analyse scientifique, l'archivage, la production d'art numérique, la documentation de preuves, la surveillance de sécurité, la reconnaissance d'images, la cartographie, la visualisation de données, etc. Pour cela, divers dispositifs d'acquisition peuvent être utilisés, tels que des scanners, des caméras, des capteurs, des télescopes, etc. le processus d'acquisition dépend du dispositif utilisé, par exemple, pour scanner un objet, on le place sur la vitre du scanner qui le numérise en le balayant horizontalement et verticalement. Les images peuvent également être acquises à partir de sources virtuelles, comme des modèles 3D créés par ordinateur ou des images satellite, qui sont capturées à distance. Après l'acquisition, les images peuvent être modifiées et sauvegardées sous forme de fichiers numériques.

En créant des images numériques, on peut reproduire fidèlement des objets et des scènes du monde réel pour les étudier, les manipuler et les partager sous forme numérique. Les images numériques sont facilement stockables, transmissibles et modifiables sans affecter leur qualité originale.

En somme, l'acquisition d'images permet de convertir des données visuelles en informations numériques exploitables pour diverses applications dans différents domaines, tels que les sciences, les arts, le commerce et la technique.

## 7.4 Outils d'acquisition des images

Les outils d'acquisition d'images sont des équipements et des logiciels qui permettent de créer et de capturer des images. Il existe une variété d'outils couramment utilisés pour acquérir des images, notamment :

### 7.4.1 Appareils photo

Les appareils photo sont des outils d'acquisition d'images courants utilisés pour capturer des photos et des vidéos de haute qualité. Il existe une grande variété d'appareils photo, allant des appareils photo reflex numériques professionnels aux appareils photo numériques grand public.

### 7.4.2 Smartphones

Les smartphones modernes sont équipés d'appareils photo de haute qualité et peuvent être utilisés pour capturer des photos et des vidéos de qualité professionnelle.

### 7.4.3 Scanner

Les scanners sont des outils d'acquisition d'images qui permettent de numériser des documents ou des images existants pour les convertir en fichiers numériques.

### 7.4.4 Caméras de surveillance

Les caméras de surveillance sont des outils d'acquisition d'images utilisés pour surveiller les activités dans les lieux publics ou privés. Ces caméras sont souvent utilisées pour la sécurité et la surveillance.

### 7.4.5 Logiciels de capture d'écran

Les logiciels de capture d'écran permettent de capturer des images de l'écran de l'ordinateur ou du smartphone. Ces images peuvent ensuite être utilisées pour créer des présentations, des tutoriels ou des images pour les réseaux sociaux.

### 7.4.6 Drones

Les drones équipés de caméras sont de plus en plus utilisés pour capturer des images aériennes de haute qualité. Ces images peuvent être utilisées dans des secteurs tels que l'agriculture, l'immobilier et le cinéma.

### 7.4.7 Webcams

Les webcams sont des outils d'acquisition d'images qui permettent de capturer des images et des vidéos en temps réel. Ces outils sont couramment utilisés pour les appels vidéo, les conférences en ligne et les diffusions en direct sur les réseaux sociaux.

## 7.5 Types d'images numériques

Les images numériques peuvent être classées en différents types. Les principaux types sont :

### 7.5.1 Les images bitmap ou raster

Ces images sont composées de pixels individuels, chacun ayant une couleur et une luminosité définies. Les fichiers bitmap sont généralement volumineux et perdent de la qualité lorsqu'ils sont agrandis.

### 7.5.2 Les images vectorielles

Ces images sont composées de vecteurs, qui sont des formes géométriques définies par des lignes, des points et des courbes. Les fichiers vectoriels sont plus petits que les fichiers bitmap et peuvent être agrandis sans perte de qualité.

### 7.5.3 Les images en niveaux de gris

Ces images ne contiennent qu'une seule couleur, mais la luminosité de chaque pixel peut varier.

### 7.5.4 Les images en couleur indexée

Ces images utilisent une palette de couleurs limitée, généralement de 256 couleurs ou moins.

### 7.5.5 Les images en couleurs réelles

Ces images utilisent un espace de couleurs plus large, généralement de mode RGB ou CMJN, pour reproduire une gamme complète de couleurs.

### 7.6 Traitement d'image

Le traitement d'images numériques regroupe un ensemble de techniques utilisées pour manipuler, améliorer ou modifier des images. Ces techniques sont utilisées pour diverses applications telles que l'amélioration de la qualité des images, la correction de défauts, l'extraction d'informations utiles ou encore la création d'effets visuels.

Pour réaliser ces traitements, des logiciels spécialisés tels que Photoshop, GIMP ou des bibliothèques de traitement d'images en programmation peuvent être utilisés. Les principales techniques de traitement d'images comprennent :

#### ✓ Filtrage

Le filtrage permet de modifier les caractéristiques d'une image en appliquant des filtres tels que les filtres flous, les filtres de netteté ou les filtres de détection de contours.

#### ✓ Segmentation

La segmentation permet de diviser une image en parties distinctes, par exemple pour isoler un objet d'intérêt.

#### ✓ Détection d'objet

La détection d'objet permet de détecter des objets spécifiques dans une image, par exemple en utilisant des algorithmes de reconnaissance faciale ou de reconnaissance de texte.

#### ✓ Transformation d'image

La transformation d'image permet de modifier la forme, la taille ou l'orientation d'une image, par exemple en utilisant des techniques de rotation, d'agrandissement ou de réduction.

#### ✓ Reconnaissance de formes

La reconnaissance de formes permet de reconnaître des motifs spécifiques dans une image, par exemple en utilisant des algorithmes d'apprentissage automatique.

Le traitement d'images est utilisé dans de nombreux domaines, comme la médecine, la surveillance, la robotique, la conception graphique et l'industrie automobile, pour ne citer que quelques exemples.

### 7.7 Avantages des traitements d'images

Le traitement d'images numériques présente divers avantages, tels que :

#### 7.7.1 L'amélioration de la qualité des images

Les techniques de traitement d'images peuvent être utilisées pour améliorer la qualité des images en augmentant la netteté, en réduisant le bruit, en ajustant les couleurs et en améliorant le contraste.

#### 7.7.2 La correction des défauts

Les défauts tels que les yeux rouges, les rayures et les taches peuvent être corrigés à l'aide de techniques de traitement d'images.

#### 7.7.3 L'extraction d'information

Les images peuvent contenir des informations utiles qui peuvent être extraites en utilisant des techniques de traitement d'images telles que la segmentation et la reconnaissance de formes.

#### 7.7.4 La création d'effets visuels

Les techniques de traitement d'images peuvent être utilisées pour créer des effets visuels tels que la superposition d'images, l'ajout de filtres ou l'application de textures.

### 7.7.5 L'automatisation des tâches

Les techniques de traitement d'images peuvent être utilisées pour automatiser des tâches telles que la reconnaissance de texte, la détection d'objets ou la classification d'images. En somme, le traitement d'images numériques est un domaine important qui offre de nombreux avantages pour améliorer la qualité des images et automatiser des tâches.

### 7.8 Prétraitement d'images

Le prétraitement d'images est une étape importante dans le traitement d'images numériques. Il consiste à appliquer des techniques pour préparer les images avant d'utiliser des techniques plus avancées de traitement d'images. Les techniques de prétraitement sont utilisées pour :

- améliorer la qualité des images.
- Augmenter la netteté.
- Améliorer la résolution.
- Corriger les défauts et éliminer le bruit indésirable.

Les techniques courantes de prétraitement d'images comprennent :

#### 7.8.1 La normalisation des couleurs et de la luminosité

Cette technique permet d'ajuster les niveaux de luminosité et de couleurs des images pour les rendre uniformes.

#### 7.8.2 Le redimensionnement

Cette technique permet de modifier la taille des images pour les adapter à une taille spécifique.

#### 7.8.3 La suppression du bruit

Cette technique permet d'éliminer le bruit indésirable de l'image.

#### 7.8.4 La correction de la distorsion

Cette technique permet de corriger les distorsions géométriques de l'image.

#### 7.8.5 La segmentation

Cette technique permet de diviser l'image en régions plus petites pour faciliter l'analyse et l'extraction de caractéristiques.

Le prétraitement d'images est une étape cruciale pour obtenir des résultats précis et fiables dans le traitement d'images. Il permet de préparer les images en éliminant les défauts et en optimisant les images pour les techniques de traitement ultérieures.

## 8 Classification des images

### 8.1 Définition de la classification d'image

La classification d'image est un processus qui permet de classer des images numériques en différentes catégories préétablies. Cette classification peut être effectuée manuellement ou à l'aide d'algorithmes informatiques, tels que les réseaux de neurones convolutifs.

Le processus de classification implique la sélection d'un ensemble de caractéristiques ou de descripteurs de l'image qui permettent de la représenter de manière quantitative. Les

algorithmes de classification peuvent ensuite utiliser ces caractéristiques pour identifier les images et les classer en fonction de leur contenu.

La classification d'image est utilisée dans de nombreux domaines, comme la reconnaissance faciale, la reconnaissance de caractères, la surveillance de la sécurité et la reconnaissance de formes.

### 8.2 La motivation de la classification

La classification d'image est motivée par plusieurs facteurs. Tout d'abord, elle permet de gérer et d'analyser efficacement un grand nombre d'images numériques en les regroupant en catégories prédéfinies, facilitant ainsi leur traitement et leur organisation.

De plus, la classification d'image est importante pour la reconnaissance et la compréhension du contenu des images. Elle permet de détecter des motifs et des caractéristiques qui peuvent être utilisés pour des tâches de reconnaissance d'objet ou de reconnaissance de forme.

Enfin, la classification d'image est utile dans de nombreux domaines d'application tels que la sécurité, la reconnaissance faciale, la classification de documents et la reconnaissance de caractères. Elle permet d'automatiser des tâches fastidieuses et répétitives, et fournit des informations précieuses pour la prise de décisions.

### 8.3 Différentes méthodes de classification

La classification d'image peut être réalisée selon différentes méthodes. En voici quelques exemples :

#### 8.3.1 La classification supervisée

Cette méthode nécessite un ensemble de données d'apprentissage préétiqueté pour entraîner l'algorithme à classer de nouvelles images.

#### 8.3.2 La classification non supervisée

Cette méthode identifie des structures et des motifs dans les données d'images sans données préétiquetées.

#### 8.3.3 Les réseaux de neurones convolutifs (CNN)

CNN extraient des caractéristiques pertinentes de l'image et les utilisent pour classer les images en différentes catégories.

#### 8.3.4 La classification basée sur les caractéristiques

Cette méthode extrait des caractéristiques spécifiques de l'image, telles que la texture, la couleur ou la forme, pour classer les images.

#### 8.3.5 La classification en cascade

Cette méthode utilise plusieurs algorithmes de classification successifs pour affiner la classification des images.

#### 8.3.6 La classification hybride

Cette méthode combine plusieurs techniques de classification pour obtenir une classification plus précise et robuste des images.

## 9 Etat de l'art

### ➤ La détection d'objet avec Open CV et Deep Learning.

L'utilisation de méthodes d'apprentissage en profondeur pour la détection d'objets en vision par ordinateur, en se concentrant sur l'application de l'algorithme SSD a donné des résultats satisfaisants et une supériorité par rapport à la méthode Haar traditionnelle. (EDDINE, 2020)

- **Détection de défauts d'un processus industriel basée sur l'analyse en composantes principales des données.**

L'objectif du travail est de construire un modèle pour minimiser le nombre de défauts en surveillant les réglages des processus industriels grâce à l'analyse en composantes principales (ACP) et en utilisant des statistiques classiques de la littérature. Un cas d'étude de l'industrie papetière est présenté. (Conférence international)

- **Détection de défauts dans les procédés industriels par l'utilisation des méthodes d'identification paramétrique.**

La thèse étudie la détection de défauts dans une colonne à distiller en utilisant une combinaison de la modélisation de type boîte noire et des tests séquentiels d'hypothèses. Les résultats obtenus montrent que les méthodes proposées sont efficaces. (AGGOUNE, 2018)

- **Contribution à la Détection et au Diagnostic des Défauts dans un Système Machine à Induction-Convertisseur.**

« Deux approches pour la détection et le diagnostic des défauts dans les machines asynchrones ont été proposées, l'une basée sur skewness-corrélation-ondelette continue et l'autre basée sur ELM-DWT pour un classificateur de détection de défauts. » (BAZI, 2016)

- **Imagerie thermique pour le contrôle qualité.**

« Une méthode de contrôle qualité des dattes basée sur l'imagerie thermique et le traitement d'images a été présentée et testée empiriquement en utilisant des caractéristiques de couleur, texture et forme. Plusieurs méthodes de classification ont été utilisées pour obtenir des performances optimales. » (SLIMANE, Imagerie thermique pour le contrôle qualité., 2021).

- **Reconnaissance et classification des fruits à l'aide de techniques d'apprentissage en profondeur.**

Dans cette méthode ils ont utilisé l'apprentissage profond pour la reconnaissance et la classification des fruits, en surmontant les défis causés par les similarités de couleur, forme et taille. Les modèles utilisés ont obtenu une forte précision sur les données traitées et augmentées, mais l'augmentation de l'image n'a pas amélioré les performances. (KATHEPURI, 2020).

## 10 Conclusion

En conclusion, l'utilisation croissante de l'intelligence artificielle (IA) et de l'apprentissage automatique a considérablement amélioré le contrôle qualité et la détection de défauts. Les techniques d'apprentissage profond, notamment les réseaux de neurones convolutionnels (CNN), sont largement utilisés pour analyser les images et vidéos, offrant une détection de défauts plus précise et efficace. Les systèmes de vision par ordinateur basés sur l'IA ont démontré des résultats prometteurs dans diverses industries et continuent de s'améliorer. Bien que l'IA ne soit pas la seule solution pour la détection de défauts, elle offre souvent une précision et une efficacité supérieures. En résumé, l'IA et l'apprentissage automatique sont des outils

## **Chapitre1 : Notion sur la détection des défauts et contrôle qualité**

---

importants pour améliorer les processus de contrôle qualité et de détection de défauts dans l'industrie. Le prochain chapitre présentera et discutera plus en détail l'apprentissage automatique, en se concentrant sur l'étude de cas choisie.

**Chapitre 2 : Apprentissage  
automatique & Apprentissage  
profond**

### 1 Introduction

L'intelligence artificielle a connu des progrès importants récemment, notamment grâce à l'apprentissage automatique. Cette branche de l'IA permet aux systèmes d'apprendre à partir de données, sans programmation explicite ne soit nécessaire. Elle est utilisée dans divers domaines tels que la reconnaissance d'images, la recommandation de produits, la prédiction de la fraude et la détection de spam, etc. Les algorithmes d'apprentissage automatique peuvent être supervisés, non supervisés ou semi-supervisés, selon les données traitées. Cette discipline s'appuie sur les statistiques, les mathématiques et la programmation, et se caractérise par le développement continu de nouveaux modèles et algorithmes pour répondre aux besoins croissants de l'intelligence artificielle.

### 2 Qu'est-ce qu'une intelligence artificielle ?

L'intelligence artificielle (IA) est un domaine de l'informatique qui vise à développer des machines capables d'accomplir des tâches qui nécessitent normalement l'intelligence humaine. Ces tâches comprennent la compréhension du langage naturel, la reconnaissance d'images et de sons, la résolution de problèmes, la prise de décision et l'apprentissage. L'objectif est de créer des systèmes qui peuvent traiter, analyser et interpréter des données de manière similaire à un être humain, permettant ainsi de réaliser des activités complexes et de prendre des décisions intelligentes. L'IA repose sur des techniques telles que l'apprentissage automatique, les réseaux de neurones artificiels et le traitement du langage naturel pour atteindre ces objectifs.

L'intelligence artificielle (IA) peut être classée en deux catégories principales : l'IA faible et l'IA forte. L'IA faible, également connue sous le nom d'IA étroite, est conçue pour effectuer des tâches spécifiques et limitées. Elle est souvent spécialisée dans des domaines tels que la reconnaissance de la parole, la classification d'images, la traduction automatique, etc. L'IA faible excelle dans ces tâches spécifiques, mais elle n'a pas la capacité de généraliser ses connaissances ou d'apprendre de nouvelles tâches de manière autonome.

En revanche, l'IA forte, également appelée IA générale, vise à résoudre des problèmes intelligents de manière autonome. Elle est conçue pour posséder une compréhension générale du monde, être capable d'apprendre de manière autonome et de s'adapter à de nouvelles situations. L'IA forte est plus proche de l'intelligence humaine et cherche à reproduire les capacités cognitives globales. Cependant, le développement de l'IA forte reste un défi complexe et est encore l'objet de recherches et de discussions approfondies.

Les techniques d'IA reposent sur des algorithmes et des modèles mathématiques qui permettent aux machines d'analyser des données complexes, de détecter des modèles et de prendre des décisions en fonction de ces modèles. Les principales techniques d'IA comprennent :

### ➤ L'apprentissage automatique

Une méthode qui permet aux machines d'apprendre à partir de données en détectant des modèles dans ces données et en ajustant leurs paramètres pour mieux prédire les résultats futurs.

### ➤ Les réseaux de neurones

Une technique d'apprentissage automatique qui simule le fonctionnement du cerveau en utilisant des neurones artificiels pour analyser des données pour prendre des décisions.

### ➤ La logique floue

Une méthode qui permet aux machines de prendre des décisions en utilisant des règles floues et des degrés de certitude plutôt que des règles logiques strictes.

Effectivement, l'intelligence artificielle (IA) trouve de nombreuses applications dans divers domaines, tels que la santé, la finance, la sécurité, les transports et l'industrie. Cependant, l'utilisation de l'IA soulève également des préoccupations éthiques et de sécurité qui doivent être abordées. En ce qui concerne la protection de la vie privée, l'utilisation de l'IA implique souvent la collecte, le traitement et l'analyse de grandes quantités de données, y compris des données personnelles sensibles. Il est crucial de mettre en place des mesures de sécurité et des protocoles appropriés pour garantir la confidentialité et la sécurité des données, ainsi que le respect des réglementations sur la protection des données.

## 2.1 Avantages de l'intelligence artificielle

L'intelligence artificielle (IA) offre de multiples avantages, notamment :

- L'IA automatise les tâches répétitives, permettant aux humains de se consacrer à des tâches plus complexes et créatives.
- Grâce à sa capacité à traiter et analyser de grandes quantités de données en peu de temps, l'IA facilite la prise de décisions éclairées et rapides.
- L'IA trouve de nombreuses applications dans divers domaines, notamment la médecine, la finance, la sécurité et l'industrie manufacturière, en améliorant la précision et la qualité des résultats.
- L'IA contribue à l'amélioration de l'efficacité et de la productivité des entreprises en automatisant des processus tels que la gestion des stocks, la planification des ressources, la logistique et la maintenance.
- L'IA est également utilisée pour résoudre des problèmes complexes dans des domaines variés tels que la science, l'ingénierie, la recherche spatiale, la météorologie et bien d'autres encore.

Ces avantages ne sont que quelques exemples des nombreuses applications possibles de l'IA dans divers domaines de la vie.

## 2.2 Inconvénients de l'intelligence artificielle

Bien que l'intelligence artificielle (IA) présente de nombreux avantages et utilisations positives, elle comporte également certains inconvénients et risques potentiels, notamment :

- **Biais et discrimination** : l'IA peut reproduire et amplifier les biais et les discriminations présents dans les données utilisées pour l'entraîner. Par exemple, si un

algorithme est formé sur des données historiques qui ont des biais raciaux ou de genre, l'IA risque de perpétuer ces biais lorsqu'elle est utilisée pour prendre des décisions.

- **Perte d'emplois** : L'automatisation croissante grâce à l'IA peut effectivement entraîner une réduction d'emplois dans certaines industries, ce qui peut avoir des implications économiques et sociales négatives.
- **Vie privée** : L'IA a la capacité de collecter et d'analyser de grandes quantités de données personnelles, ce qui soulève des préoccupations en matière de vie privée si ces données sont mal utilisées ou divulguées.
- **Contrôle et transparence** : l'IA peut être difficile à comprendre et à contrôler, en particulier lorsqu'elle est utilisée pour prendre des décisions critiques qui ont un impact sur la vie des gens. Il peut être difficile de savoir comment l'IA prend des décisions et de détecter les erreurs ou les biais.
- **Coût élevé** : la mise en place et l'utilisation de l'IA peut être coûteuse en raison de la complexité de la technologie et des ressources requises pour son développement et sa maintenance.
- **Sécurité** : l'IA peut être vulnérable aux attaques et aux piratages, ce qui peut avoir des conséquences graves, notamment la manipulation des systèmes, le vol de données et la perturbation des opérations.

En fin de compte, il est important de prendre en compte ces inconvénients et de travailler à minimiser les risques associés à l'utilisation de l'IA.

### 2.3 L'impact de l'intelligence artificielle dans l'industrie

L'IA est en train de révolutionner de nombreux secteurs industriels en offrant des possibilités d'automatisation des tâches, d'amélioration des analyses de données, d'optimisation des décisions et de personnalisation des produits et services. Cependant, il est essentiel de prendre en compte les conséquences sociales et économiques découlant de son utilisation, notamment la suppression d'emplois et les inégalités. Il est donc crucial de surveiller attentivement les développements de l'IA et de mettre en place des réglementations appropriées pour gérer les impacts potentiels.

## 3 L'apprentissage automatique

### 3.1 Définition d'apprentissage automatique

Au cours de la dernière décennie, l'intérêt pour l'apprentissage automatique a considérablement augmenté. Cependant, il existe des conflits entre les capacités des machines et nos attentes (Patterson, 2017). Le Machine Learning ou apprentissage automatique, en tant que sous-domaine de l'intelligence artificielle, permet aux machines d'apprendre à partir de données et de les utiliser pour prendre des décisions ou effectuer des prédictions. Cela implique la création de modèles informatiques qui s'améliorent avec l'expérience grâce à l'acquisition de nouvelles données (Clayton, 2020). Il existe différents types d'apprentissage automatique, tels que supervisé, non supervisé et renforcé, utilisés dans des domaines tels que la reconnaissance vocale, la reconnaissance d'images et la prévision de séries chronologiques. Les algorithmes d'apprentissage automatique sont largement appliqués dans des industries telles que la finance, la santé, la défense, l'énergie et les technologies de l'information, afin d'améliorer les processus décisionnels et d'automatiser des tâches spécifiques.

### 3.2 Types de systèmes d'apprentissage automatique

Il existe effectivement différents types d'apprentissage automatique. Selon (Géron, 2017), la définition du type d'apprentissage repose sur les réponses aux deux questions suivantes :

- Est-ce que cet apprentissage peut trouver des structures ou des patterns dans les données sans étiquettes préalables ?

- Est-ce que cet apprentissage utilise une base de données fournie par des humains ?

1- Si la réponse est positive pour les deux questions, cela indique que nous avons affaire à un apprentissage supervisé.

2- Si la réponse est négative pour la première question et positive pour la deuxième question, cela indique que nous sommes en présence d'un apprentissage non supervisé.

3- Si la réponse est négative pour les deux questions, cela correspond à un type d'apprentissage par renforcement.

Voici les types de systèmes d'apprentissage automatique :

- **Apprentissage supervisé** : utilisé pour prédire une étiquette ou une valeur continue à partir de données étiquetées.
- **Apprentissage non supervisé** : utilisé pour explorer les structures dans les données sans étiquettes.
- **Apprentissage semi-supervisé** : utilisé lorsque seules certaines données sont étiquetées.
- **Apprentissage par renforcement** : ou un agent interagit avec son environnement pour maximiser une récompense.
- **Apprentissage profond** : Les réseaux de neurones multicouches sont utilisés pour traiter des données complexes.

Dans ce qui suit, nous définissons chacun de ces types : apprentissage supervisé, apprentissage non supervisé, apprentissage semi-supervisé, apprentissage par renforcement, et apprentissage profond.

#### 3.2.1 Apprentissage supervisé

Les algorithmes d'apprentissage supervisé sont formés sur des données labellisées, c'est-à-dire des données qui ont été étiquetées avec des résultats corrects. Ce type d'apprentissage automatique est utilisé pour des tâches telles que la classification et la régression.

Cette forme d'apprentissage implique l'utilisation d'un algorithme ( $F$  : modèle) pour prédire une variable de sortie ( $Y$  : output) en fonction d'un ensemble de variables d'entrée ( $X$  : input), grâce à l'analyse de données  $X$ .

De manière générale, une machine est capable d'apprendre une relation  $f : x \rightarrow y$  qui lie les variables d'entrée  $x$  à la variable de sortie  $y$  en étudiant des millions d'exemples d'associations  $x \rightarrow y$ .

L'objectif de l'apprentissage supervisé consiste à construire un modèle (classifieur) qui peut classer de nouvelles instances (qui ne sont pas dans  $X$ ) avec une marge d'erreur minimale. (Voir Figure 2.1).

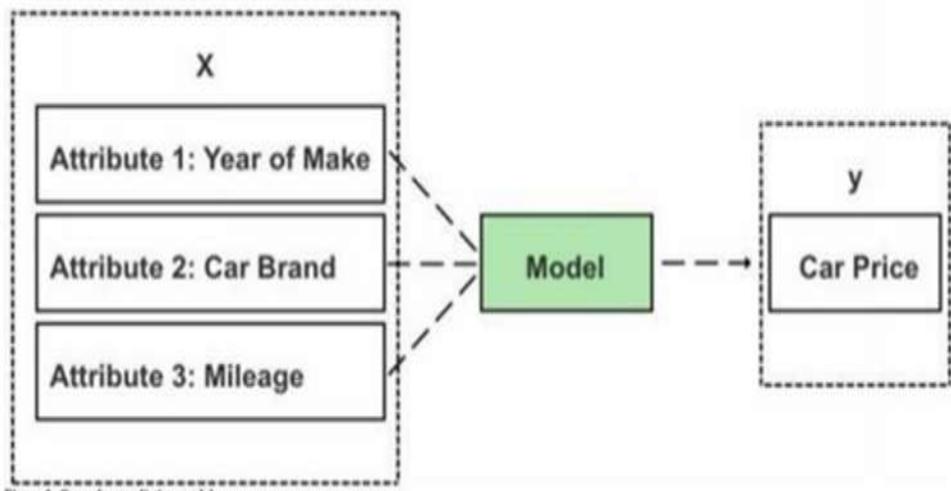


Figure 2.1: Apprentissage supervisée.

D'une manière générale, il existe deux algorithmes d'apprentissage supervisé couramment utilisés.

### 1. Régression

La régression est en effet un algorithme d'apprentissage supervisé qui cherche à trouver la relation entre une variable indépendante et une variable dépendante en utilisant une ligne droite ou une courbe pour représenter cette relation. Elle est couramment utilisée pour prédire une valeur continue à partir d'une ou plusieurs autres valeurs.

### 2. Classification

La classification est un algorithme d'apprentissage supervisé utilisé pour prédire une étiquette ou une classe en fonction de certaines caractéristiques ou variables d'entrée. Il s'agit de déterminer à quelle catégorie appartient un objet ou une observation en fonction de ses traits.

Par exemple, la classification peut être utilisée pour prédire la variété d'une pomme en fonction de ses caractéristiques telles que la forme, la couleur et la texture. (Voir figure 2.2).

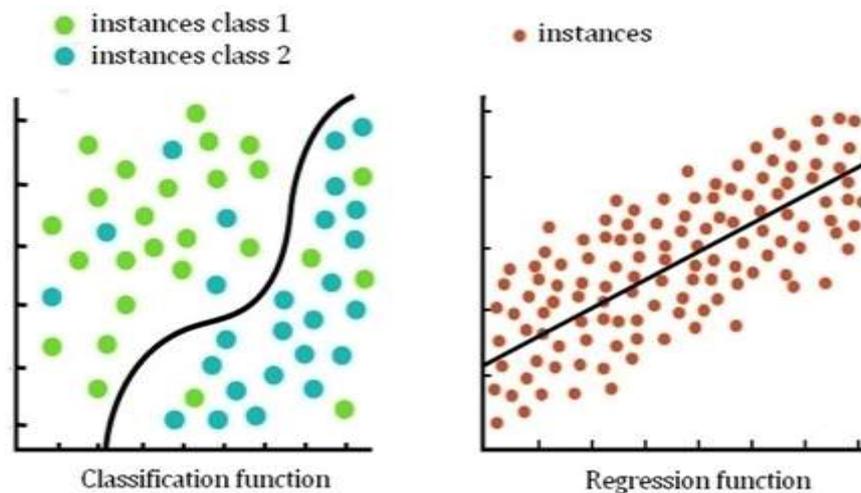


Figure 2.2: Classification et régression.

Les algorithmes les plus célèbres utilisés dans cette approche sont les suivants (GAËL, 2020) :

### ➤ La régression logistique

Effectivement, la régression logistique est un algorithme d'apprentissage supervisé utilisé pour prédire la probabilité d'un événement binaire. Il permet de modéliser la relation entre une variable binaire de sortie ( $Y$ ) et une ou plusieurs variables d'entrée ( $X_i$ ) en utilisant une fonction logistique. Cette fonction est utilisée pour estimer la probabilité qu' $Y$  prenne une certaine valeur en fonction des variables  $X_i$ .

L'objectif principal de la régression logistique est d'identifier les variables explicatives les plus pertinentes pour prédire l'événement binaire avec précision. Cette technique est couramment utilisée en classification, notamment pour la détection de spam ou la prédiction de la probabilité de défaut de crédit. (Voir figure 2.3).

### ➤ La régression linéaire

Effectivement, la régression linéaire est un type d'algorithme d'apprentissage supervisé qui permet de modéliser la relation entre une variable cible ( $Y$ ) et des variables prédictives ( $X_i$ ). Son objectif est de décrire la relation entre une variable dépendante ( $Y$ ) et une ou plusieurs variables indépendantes ( $X_i$ ) en utilisant une fonction linéaire. Cela permet de prédire la valeur de la variable cible en fonction des valeurs des variables prédictives. La régression linéaire est couramment utilisée pour des tâches de prédiction et d'estimation, notamment dans les domaines de l'économie, des sciences sociales et des statistiques. (Voir figure 2.3).

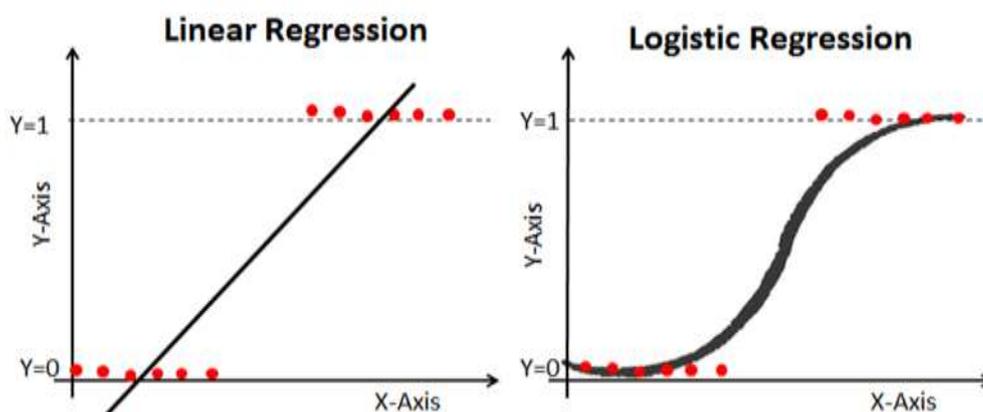


Figure 2.3: Régression linéaire et régression logistique.

### ➤ Machines à vecteurs de soutien (SVM)

Les machines à vecteurs de support (SVM) sont en effet des algorithmes d'apprentissage supervisé utilisés pour la classification et la régression. Ils sont appréciés pour leur capacité à traiter des données en haute dimension, leur efficacité en termes de mémoire et leur garantie théorique de trouver la frontière de décision optimale.

Pour séparer les données en différentes classes, les SVM utilisent un hyperplan qui maximise la marge, c'est-à-dire la distance entre l'hyperplan et les points de données les plus proches de

chaque classe. Ces points de données les plus proches de l'hyperplan sont appelés vecteurs de support, d'où le nom "machines à vecteurs de support".

De plus, les SVM peuvent utiliser des fonctions de noyau pour transformer les données d'entrée dans un espace de dimension supérieure, où il est plus facile de trouver un hyperplan séparateur. Cela permet de traiter des données non linéaires en les projetant dans un espace de dimension supérieure où elles peuvent être séparées de manière linéaire.

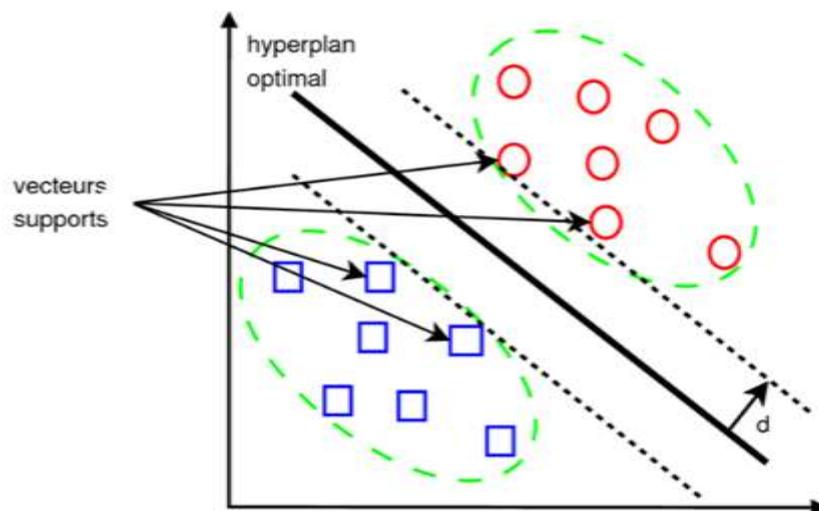


Figure 2.4: Architecture d'un vecteur support machine.

Les SVM sont largement utilisées dans de nombreux domaines, tels que la vision par ordinateur, la bio-informatique, la finance et le traitement du langage naturel, en raison de leur capacité à traiter des données complexes et à obtenir de bonnes performances de prédiction.

### ➤ La méthode des K plus proches voisins

La méthode des K plus proches voisins est un algorithme d'apprentissage automatique supervisé qui permet de classifier ou de prédire une valeur numérique en se basant sur les K échantillons les plus proches dans l'ensemble d'entraînement. Dans le cas de la classification, la méthode consiste à prédire la classe d'un nouvel échantillon en utilisant la classe majoritaire parmi les K échantillons les plus proches. Dans le cas de la régression, la méthode permet de prédire la valeur numérique en utilisant la moyenne des K échantillons les plus proches. Il est important de choisir la valeur de K de manière optimale pour éviter les effets de bruit dans les données ou la perte de précision.

La méthode des K plus proches voisins est utilisée dans divers domaines tels que la reconnaissance de caractères, la vision par ordinateur, la bio-informatique et la classification de documents. Toutefois, cette méthode peut être sensible à la dimensionnalité des données et nécessite une grande quantité de mémoire pour stocker l'ensemble d'entraînement.

### ➤ Naïve Bayes

Naïve Bayes est un algorithme de classification supervisée en apprentissage automatique qui repose sur le théorème de Bayes pour estimer la probabilité qu'un événement se produise. Il suppose que toutes les caractéristiques d'un ensemble de données sont indépendantes les unes

des autres, ce qui simplifie les calculs nécessaires pour estimer les probabilités. L'algorithme utilise des exemples d'apprentissage pour estimer les probabilités que chaque classe soit associée à une combinaison de caractéristiques, et utilise ces estimations pour déterminer la classe la plus probable pour chaque nouvel exemple.

Naïve Bayes est souvent utilisé pour la classification de textes, car il peut traiter un grand nombre de caractéristiques (par exemple, les mots dans un texte) de manière efficace. Cependant, il peut être moins performant pour les données complexes ou avec des caractéristiques fortement corrélées. En somme, Naïve Bayes est un algorithme simple et efficace pour la classification de données, mais il convient de prendre en compte ses limitations pour l'appliquer correctement.

### ➤ Les arbres de décision

Les arbres de décision sont une méthode de prédiction et de classification largement utilisée dans les domaines de l'apprentissage automatique et de l'analyse de données. En utilisant une structure d'arbre, l'ensemble de données est divisé en branches représentant chaque décision possible. Chaque branche est associée à un critère de partitionnement basé sur une variable d'entrée spécifique, et chaque feuille représente une prédiction ou une classification pour l'ensemble de données.

Les arbres de décision sont utilisés dans des domaines tels que la médecine, la finance et le marketing en raison de leur capacité à gérer les données manquantes et aberrantes, les données catégorielles et numériques, et leur capacité à fournir des résultats interprétables et visuellement attrayants.

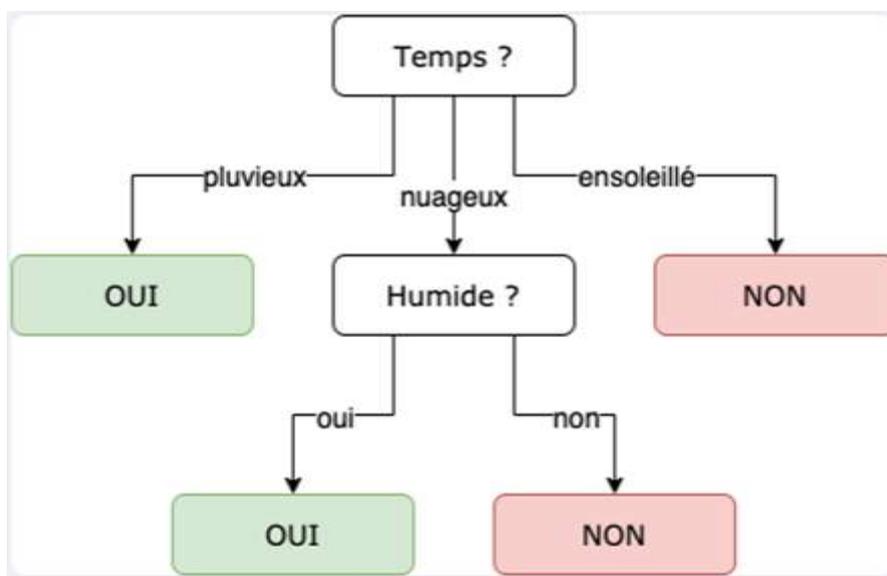


Figure 2.5: Architecture d'un arbre de décision.

Un arbre de décision est une représentation des règles de classification où chaque chemin de la racine à une feuille représente une règle. Il est composé de trois types de nœuds :

- Le nœud principal.
- Les nœuds de succursale.

- Les nœuds feuilles qui contiennent les étiquettes de classe.

Pour créer un modèle d'arbre de décision, on utilise des données d'apprentissage pour déterminer la variable à diviser à chaque nœud et la valeur de la division. Les décisions d'arrêt ou de division ultérieure conduisent à l'affectation des nœuds feuilles à une classe spécifique. Un avantage de l'arbre de décision est qu'il n'est pas nécessaire de créer exclusivement des variables fictives.

### ➤ Les Forêts Aléatoires

Les forêts Aléatoires sont une technique couramment utilisée en apprentissage automatique pour la classification, la régression et d'autres tâches prédictives. Elles sont constituées d'un ensemble d'arbres de décision, chacun construit à partir d'un sous ensemble aléatoire des données d'entraînement. A chaque itération, un nouvel arbre est construit en utilisant un sous ensemble différent des données d'entraînement, et les résultats de tous les arbres sont combinés pour produire une prédiction finale.

Les forêts Aléatoires sont appréciées pour leur capacité à gérer les données manquantes et les valeurs aberrantes, leur capacité à gérer de grandes quantités de données, leur capacité à détecter les interactions entre les variables et leur résilience face aux problèmes de sur apprentissage. Elles sont utilisées dans divers domaines tels que la reconnaissance de formes, la bio-informatique, l'imagerie médicale, la finance et le marketing.

### ➤ Les réseaux de neurones

Les réseaux de neurones sont un type d'algorithme d'apprentissage automatique inspiré de fonctionnement du cerveau humain. Ils sont composés d'unités de traitement appelées « neurones » interconnectées en couches, et chaque neurone effectue une opération mathématique sur les données d'entrée avant de les transmettre à d'autres neurones du réseau.

Le processus d'apprentissage d'un réseau de neurones se fait par ajustement des poids associés à chaque connexion entre les neurones, dans le but de minimiser l'erreur de prédiction entre la sortie attendue et la sortie réelle. Cette méthode, appelée « rétro propagation du gradient », permet au réseau de s'adapter aux données d'entraînement et de généraliser pour prédire de nouvelles données.

Les réseaux de neurones ont la capacité de traiter des données non structurées telles que des images, des sons et du texte, ce qui en fait une technologie précieuse pour la reconnaissance de la parole, la reconnaissance d'images, la traduction automatique, la prédiction de séries temporelles et d'autres applications. Ils sont de plus en plus utilisés dans de nombreux domaines en raison de leur efficacité pour traiter des données complexes. (Voir figure 2.6).

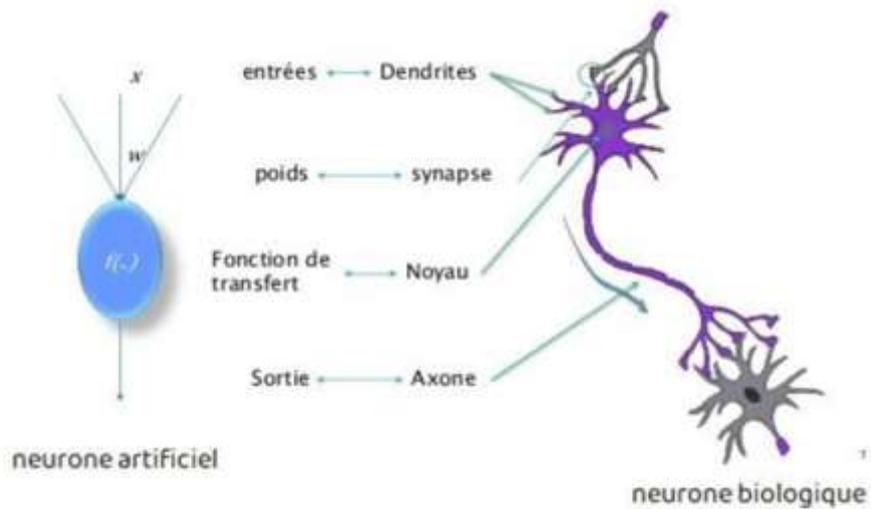


Figure 2.6: Neurone biologique et neurone artificiel (Werfel li, 2015).

### 3.2.2 Apprentissage non supervisé

Avec l'apprentissage non supervisé la machine n'a pas besoin de l'aide pour apprendre (voir figure 2.7). Les algorithmes d'apprentissage non supervisé sont formés sur des données non labellisées, ce qui signifie qu'ils doivent découvrir des relations et des modèles à partir des données elles-mêmes. Ce type d'apprentissage automatique n'utilise pas de label ou d'étiquette pour la classification des données. Il est utilisé pour trouver des structures cachées dans les données et même utilisé pour des tâches telles que la segmentation de clients et la réduction de dimensionnalité.



Figure 2.7: Apprentissage non supervisé (Géron, 2017) .

Lorsqu'on ne dispose que des données d'entrée  $X$  sans avoir de variable de sortie correspondante, on utilise l'apprentissage non supervisé. L'objectif de ce type d'apprentissage est de caractériser la distribution des données et les relations entre les variables, sans faire de distinction entre les variables observées et celles à prédire.

Selon (Géron, 2017), il existe deux types d'apprentissage non supervisé :

#### ➤ Regroupement (clustering)

Regroupement est un algorithme d'apprentissage non supervisé qui consiste à rassembler des observations similaires en groupes ou en clusters. Cet algorithme utilise des mesures de distance ou de similarité pour déterminer la similarité entre les observations et les affecter au même

cluster. Le regroupement est souvent utilisé pour la segmentation de données, la découverte de modèles et la compréhension de la structure sous-jacente de l'ensemble de données.

### ➤ Réduction de la dimensionnalité

La réduction de la dimensionnalité est un algorithme d'apprentissage non supervisé qui vise à réduire le nombre de dimensions d'un ensemble de données tout en conservant une représentation significative des données d'origine. L'objectif est de réduire la complexité du modèle tout en préservant les propriétés importantes des données, ce qui peut améliorer les performances des algorithmes d'apprentissages.

Les algorithmes les plus célèbres utilisés dans cette approche sont (Issarane, 2020) :

### ➤ K-Moyenne

La méthode des K-Moyenne est une technique de clustering non supervisé qui permet de diviser un ensemble de données en K groupes distincts. Elle cherche à minimiser la variance intra-cluster en itérativement assignant les données à K centres de cluster aléatoires, en réassignant les données à leur centre de cluster le plus proche, et en mettant à jour les centres de cluster. Les K centres de cluster finaux correspondent aux centres des K clusters formés.

Les K-moyennes sont couramment utilisées dans des domaines tels que la segmentation de clients, l'analyse de données géo spatiales et la classification de documents. Cependant, il convient de souligner que les résultats obtenus peuvent varier en fonction des choix initiaux des centres de cluster aléatoires.

### ➤ Analyse de classification hiérarchique (HCA)

L'analyse de classification hiérarchique est une méthode de clustering qui vise à regrouper des données similaires en formant une hiérarchie de clusters imbriqués les uns dans les autres. Contrairement à la méthode des K-moyennes qui nécessite de spécifier le nombre de clusters à former, l'HCA peut être divisée en deux types : la classification ascendante hiérarchique (HAC) et la classification descendante hiérarchique (ADC).

La HCA commence avec chaque observation formant son propre cluster, et itérativement fusionne les clusters les plus proches jusqu'à ce que toutes les observations soient dans un seul cluster. L'ADC commence avec toutes les observations dans un seul cluster, et divise itérativement les clusters les plus éloignés jusqu'à ce que chaque observation forme son propre cluster.

L'HCA est couramment utilisé dans des domaines tels que la biologie, l'écologie et le marketing pour classer des données. Elle permet également de visualiser graphiquement la hiérarchie des clusters grâce à un dendrogramme. Toutefois, il convient de noter que l'HCA peut être sensible à la présence d'oublier ou de données manquantes, et peut être coûteuse en temps de calcul pour des ensembles de données volumineux.

### ➤ Analyse en composantes principales (PCA)

L'analyse en composantes principales (PCA) est un algorithme d'apprentissage non supervisé utilisé en statistiques pour la réduction de dimension. Il cherche à trouver les directions (composantes principales) dans les données qui expliquent la plus grande partie de la variance. Cela peut être utile pour visualiser les données ou pour les préparer à une analyse plus poussée.

### ➤ Apriori

Apriori est un algorithme de fouille de données qui permet de découvrir des règles d'association entre des items dans un ensemble de données. L'algorithme est basé sur l'idée que si un

ensemble d'items apparaît fréquemment ensemble, il est probable qu'ils soient liés d'une manière ou d'une autre.

L'algorithme Apriori utilise une approche itérative pour trouver des ensembles d'items fréquents en utilisant le principe d'anti-monotonie, qui permet de ne générer que les sous-ensembles susceptibles d'être fréquents. Une fois que tous les ensembles d'items fréquents ont été trouvés, l'algorithme utilise ces ensembles pour générer des règles d'associations entre les items.

Apriori est couramment utilisé dans les domaines du marketing et du commerce électronique pour l'analyse des paniers d'achats des clients, la recommandation de produits et la segmentation de clients. Toutefois, il convient de noter que l'algorithme peut être coûteux en temps de calcul pour des ensembles de données volumineux.

### 3.2.3 Apprentissage semi-supervisé

Les algorithmes d'apprentissage semi-supervisé combinent les approches supervisé et non supervisé en utilisant des données à la fois labellisées et non labellisées. Il existe plusieurs exemples d'apprentissage semi-supervisé, tels que :

#### ➤ Classification de documents

Pour la classification de documents tels que les courriels ou les articles de presse en différentes catégories préétiquetées et en laissant le reste des catégories à déterminer.

#### ➤ Segmentation d'images

Pour segmenter des images en différents objets ou régions, en utilisant un petit nombre de régions préétiquetées et en laissant le reste des régions à déterminer.

#### ➤ Génération de textes

Pour générer du texte en utilisant un petit nombre de phrases d'entraînement et en laissant le reste du texte à déterminer.

### 3.2.4 Apprentissage par renforcement

Dans l'apprentissage par renforcement, la machine peut apprendre de manière autonome sans supervision humaine ni fourniture de données, en interagissant avec son environnement et en ajustant sa politique d'action pour maximiser les récompenses à long terme.

Les algorithmes d'apprentissage par renforcement sont formés en interagissant avec un environnement et en apprenant des récompenses ou des pénalités pour des actions spécifiques (Voir figure 2.8).

La machine est capable d'apprendre de manière autonome la meilleure stratégie, appelée politique, pour maximiser les récompenses au fil du temps. La politique détermine l'action que l'agent doit choisir dans une situation donnée. Cela permet à la machine de prendre des décisions en fonction des informations qu'elle reçoit de son environnement et d'adapter sa politique en fonction des récompenses obtenues. L'apprentissage par renforcement permet à la machine d'explorer différentes actions et d'ajuster sa politique en fonction des résultats obtenus pour atteindre un objectif spécifique. (Géron, 2017).

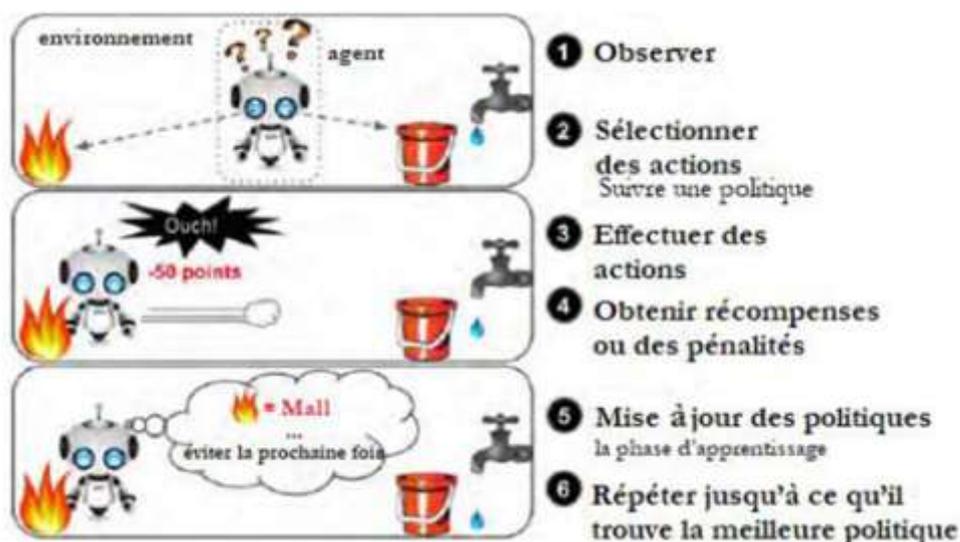


Figure 2.8: Apprentissage par renforcement (Géron, 2017).

Ce type d'apprentissage automatique est utilisé pour des tâches telles que la prise de décision dans les jeux vidéo et les systèmes robotiques.

Parmi les premiers algorithmes d'apprentissage par renforcement, on trouve l'algorithme « Temporal différence Learning », proposé par « Richard Sutton » en 1988 (Sutton, 1988). Un autre algorithme important est le « Q-Learning » développé dans le cadre d'une thèse par « Chris Watkins » en 1989 et publié réellement en 1992 (Watkins, 1992). Ces deux algorithmes ont été des contributions majeures dans le domaine de l'apprentissage par renforcement.

### 3.2.5 Apprentissage profond

L'apprentissage profond est une méthode de traitement de l'apprentissage automatique basée sur des réseaux de neurones profonds. Il implique la construction de modèles complexes qui sont formés sur des données d'entrée pour apprendre les relations sous-jacentes et les décrire à l'aide d'une représentation mathématique. L'apprentissage profond est souvent utilisé pour les tâches telles que la reconnaissance de la parole, la reconnaissance d'images et la prédiction de séquences.

### 3.3 Le choix d'un type d'apprentissage automatique

Le choix d'un type d'apprentissage automatique dépend de divers facteurs tels que la nature du problème à résoudre, la disponibilité des données et les objectifs de l'application. Face à la diversité des classificateurs disponibles, il est courant de se poser la question suivante : "Quel algorithme devrais-je utiliser ?".

Selon (Li, 2020), la réponse à cette question varie les facteurs suivants :

- La taille et la complexité des données.
- La disponibilité de la puissance de calcul et des ressources informatiques.
- L'urgence de la tâche.
- Le but d'utilisation de ces données.

La compétence et les connaissances de l'utilisateur en matière d'apprentissage automatique jouent également un rôle important dans le choix de la méthode appropriée. Il est essentiel de comprendre les avantages et les limites de chaque approche pour prendre une décision éclairée.

La ( Figure 2.9) fournit des recommandations sur les algorithmes à essayer en premier en fonction des facteurs mentionnés précédemment.

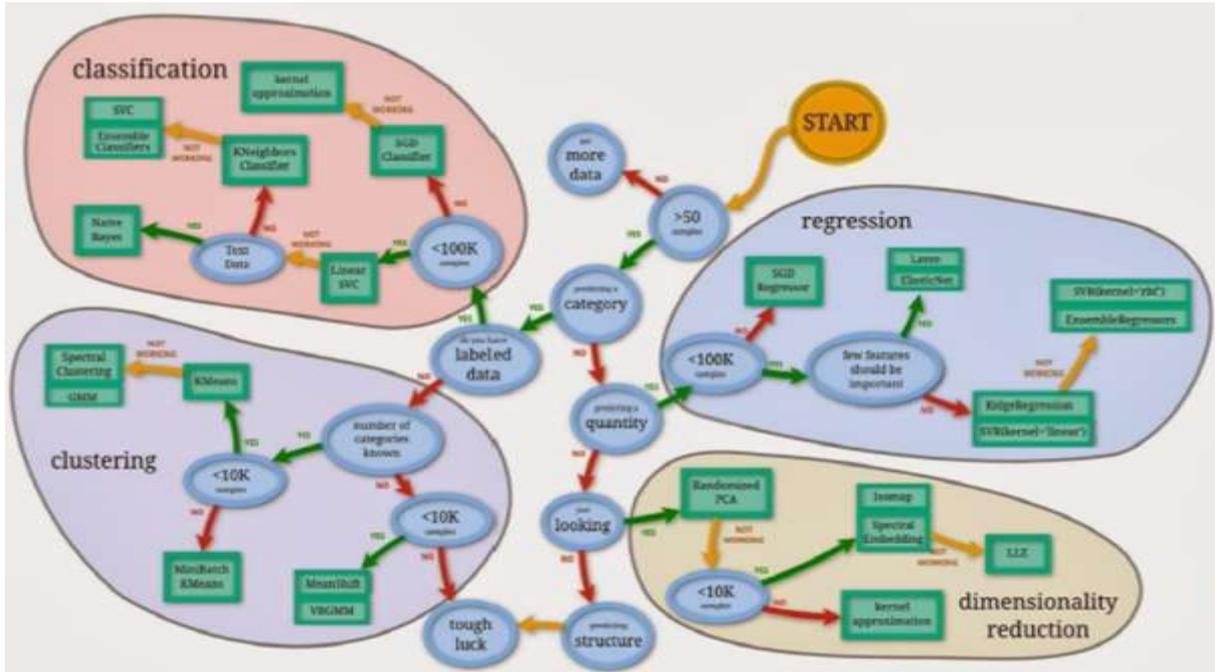


Figure 2.9: Le choix de l'algorithme d'apprentissage selon certains facteurs (Li, 2020).

#### 4 De l'apprentissage automatique à l'apprentissage en profondeur

Il est vrai que l'apprentissage automatique peut parfois rencontrer des problèmes de temps et l'efficacité, surtout lorsque les données sont volumineuses ou que les algorithmes sont complexes. Pour entraîner des modèles de qualité, il est souvent nécessaire de disposer de beaucoup de données et de temps de calcul. De plus, lorsque les algorithmes renvoient des prédictions inexactes, cela peut nécessiter l'intervention d'un ingénieur pour ajuster le modèle, ce qui peut prendre du temps et affecter l'efficacité.

Afin de prévenir ces problèmes, il est crucial de sélectionner avec soin les algorithmes d'apprentissage automatique en fonction des données et des problèmes à résoudre. Il est également essentiel de disposer de ressources adéquates, tant en termes de matériel que de personnel, pour entraîner et évaluer les modèles. En outre, il est recommandé de surveiller régulièrement les performances des modèles et d'effectuer des ajustements si nécessaire afin de maintenir leur efficacité. (Grossfeld, Le programme du Master en intelligence artificielle, 2020).

Les algorithmes simples d'apprentissage automatique sont efficaces dans de nombreux cas, mais ils sont limités pour résoudre des problèmes complexes de l'intelligence artificielle tels que la reconnaissance vocale ou la reconnaissance d'objets. Pour surmonter ces limites, l'apprentissage en profondeur a été développé. Il repose sur l'utilisation de réseaux de neurones profonds capables d'apprendre des modèles complexes en traitant les données à différents niveaux d'abstraction. Les réseaux de neurones profonds peuvent s'entraîner sur de grandes quantités de données et généraliser efficacement à de nouvelles données, ce qui est crucial pour des tâches telles que la reconnaissance vocale et la reconnaissance d'objets. En résumé, l'apprentissage en profondeur constitue un outil puissant pour résoudre des problèmes complexes en intelligence artificielle. (Goodfellow, 2016).

### 5 l'apprentissage profond

#### 5.1 Définition d'apprentissage en profondeur

Le Deep Learning, une branche de l'apprentissage automatique, utilise des réseaux de neurones artificiels profonds pour résoudre des problèmes complexes tels que la reconnaissance d'images et la traduction automatique. En utilisant des couches de neurones interconnectés, il apprend de manière autonome des caractéristiques abstraites à partir des données. Son succès grandissant est dû à ses performances impressionnantes et à la disponibilité accrue des données et des ressources de calcul. Le Deep Learning a révolutionné des domaines tels que la reconnaissance vocale et offre des opportunités pour de nouvelles avancées technologiques. (Voir figure 2.10).

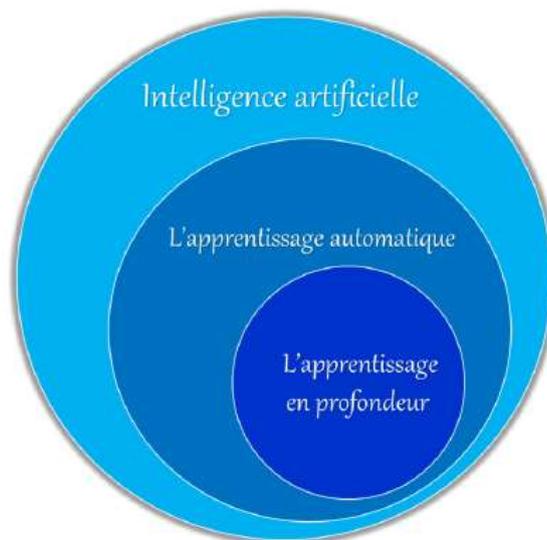


Figure 2.10: Les sous-branches de l'intelligence artificiel.

#### 5.2 Réseau de neurones artificiels (ANN)

Les réseaux de neurones artificiels sont des modèles de machine learning qui s'inspire du fonctionnement des réseaux de neurones biologiques. Ils sont utilisés dans divers domaines tels que la reconnaissance de formes, la classification des données, le traitement du langage naturel, la vision par ordinateur, etc. Un réseau de neurones artificiels est composé de neurones interconnectés, qui transmettent et traitent les informations à travers des couches. Chaque neurone effectue des opérations mathématiques sur les entrées reçues et transmet les résultats aux neurones suivants. La sortie de la dernière couche est utilisée pour les prédictions ou la classification des données. Les réseaux de neurones sont entraînés à l'aide d'algorithmes d'apprentissage supervisé, non supervisé ou semi-supervisé. (Voir figure 2.11).

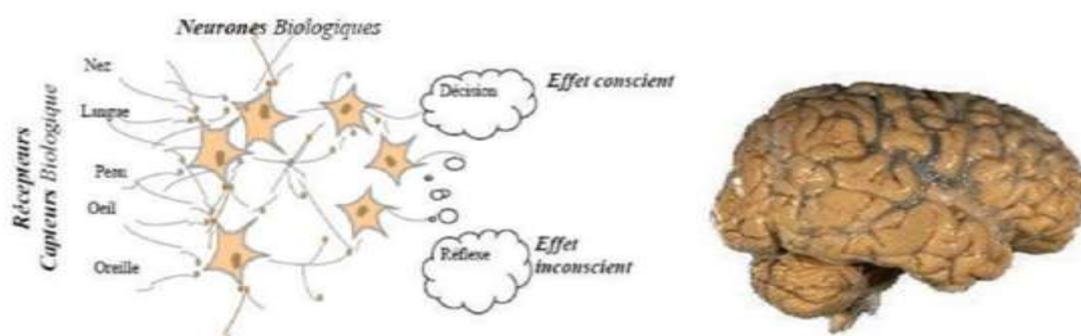


Figure 2.11: Structure d'un réseau de neurone biologique et le cerveau humain.

### 5.3 Réseaux de neurones à propagation avant (FFNN)

Un réseau de neurones à propagation avant est un type de réseau de neurones artificiels où l'information circule de manière unidirectionnelle, de l'entrée vers la sortie, sans boucles récurrentes. Il est composé de couches de neurones connectées de manière unidirectionnelle, avec une couche d'entrée, une couche de sortie et éventuellement des couches cachées. Ces réseaux sont couramment utilisés pour des tâches de classification et de régression en apprenant à partir de données d'entraînement pour prédire des résultats pour de nouvelles données. Ils sont efficaces pour modéliser des relations complexes et sont utilisés dans des domaines tels que la vision par ordinateur et le traitement du langage naturel. Comparés aux réseaux récurrents, ils sont plus simples à entraîner et à comprendre, mais peuvent être limités dans leur capacité à modéliser des séquences et des dépendances temporelles. (Voir figure 2.12).

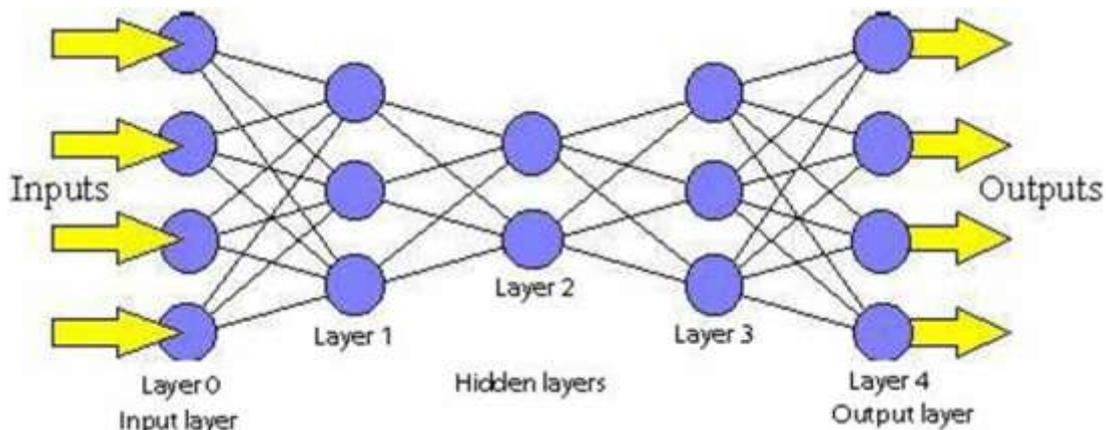


Figure 2.12: Architecture d'un réseau de neurones à propagation avant.

#### 5.3.1 Pourquoi les réseaux de neurones à propagation avant (FFNN) ?

Les FFNN sont une classe de modèles de réseaux de neurones artificiels qui sont largement utilisés dans le domaine de l'apprentissage automatique et de l'intelligence artificielle.

Ils ont plusieurs avantages par rapport à d'autres modèles de réseau de neurones notamment :

- Ils sont relativement simples à mettre en œuvre et à entraîner.
- Ils peuvent être utilisés pour une variété de tâches, notamment la classification, la régression et la prédiction de séquences.
- Ils ont une capacité de généralisation élevée, ce qui signifie qu'ils peuvent bien fonctionner sur des données qu'ils n'ont pas vues pendant l'entraînement.
- Ils sont faciles à interpréter, ce qui est important pour les applications où la transparence et la compréhensibilité des résultats sont importantes.

Dans l'ensemble, les FFNN sont une technique populaire pour la modélisation de données et sont souvent utilisés comme base pour des modèles plus complexes, tels que les réseaux de neurones convolutifs et les réseaux de neurones récurrents.

#### 5.3.2 Types de réseaux de neurones à propagation avant (FFNN)

Il existe plusieurs types de réseaux de neurones à propagation avant en fonction de leur architecture et de leur utilisation. En voici quelques exemples :

##### ✓ Les réseaux de neurones multicouches (Multilayer Perceptron, MLP)

Ce sont les réseaux de neurones à propagation avant les plus couramment utilisés. Ils sont constitués de plusieurs couches de neurones, chacune contenant un certain nombre de neurones

interconnectés. Les couches sont généralement une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie.

✓ **Les réseaux de neurones à convolution (Convolutional Neural Networks, CNN)**

Ce sont des réseaux de neurones utilisés pour le traitement d'images et de vidéos. Ils sont constitués de couches de convolution pour extraire des caractéristiques de l'image en entrée.

✓ **Les réseaux de neurones récurrents (Recurrent Neural Networks, RNN)**

Ce sont des réseaux de neurones utilisés pour traiter des données séquentielles telles que des séquences de texte ou de sons. Les RNN sont conçus pour conserver une mémoire des entrées précédentes afin de pouvoir prédire les sorties futures.

✓ **Les réseaux de neurones auto-encodeurs (Autoencoder Neural Networks, AE)**

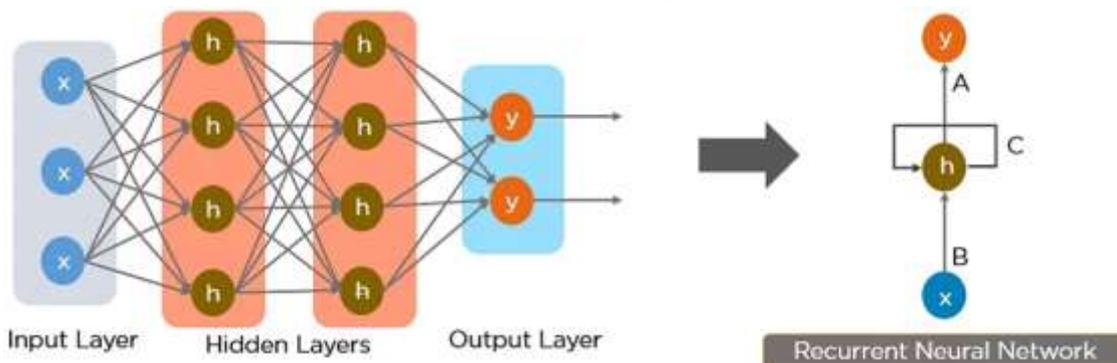
Ce sont des réseaux de neurones utilisés pour apprendre des représentations comprimées des données en entrée. Ils sont constitués d'une couche d'entrée, d'une ou plusieurs couches cachées et d'une couche de sortie qui reconstruit les données en entrée.

✓ **Les réseaux de neurones profonds (Deep Neural Networks, DNN)**

Ce sont des réseaux de neurones à propagation avant avec plusieurs couches cachées. Les DNN sont utilisés pour apprendre des représentations hautement abstraites des données en entrée et sont souvent utilisés dans des tâches de classification et de reconnaissance d'images.

### 5.4 Réseaux de neurones récurrents (RNN)

Les réseaux de neurones récurrents (RNN) sont spécialement conçus pour traiter des données séquentielles ou temporelles, telles que des séquences de mots ou des signaux sonores. Contrairement aux réseaux de neurones à propagation avant, les RNN utilisent des connexions récurrentes qui leur permettent de prendre en compte l'historique de l'entrée lors du calcul de la sortie. Cela les rend particulièrement adaptés à des tâches telles que la traduction automatique, la reconnaissance de la parole et la génération de texte. Le fonctionnement des RNN repose sur le principe de sauvegarder la sortie d'une couche spécifique et de la réinjecter en tant qu'entrée, ce qui leur permet de prédire la sortie de la couche suivante en tenant compte du contexte précédent. Cette capacité à prendre en compte la séquence des données est ce qui distingue les RNN des autres types de réseaux de neurones. (Voir figure 2.13).



**Figure 2.13:** Réseau neuronal récurrent simple.

Dans cette architecture, les nœuds des différentes couches du réseau neuronal sont regroupés pour former une seule couche de réseaux neuronaux récurrents. Les paramètres du réseau, représentés par A, B et C, sont utilisés pour améliorer la sortie du modèle.

Plus précisément, la couche d'entrée est représentée par "x", la couche cachée par "h" et la couche de sortie par "y". À chaque instant "t", l'entrée courante est une combinaison des entrées actuelles  $x(t)$  et des entrées précédentes  $x(t-1)$ . La sortie à chaque instant est rétro propagée à travers le réseau pour améliorer la sortie globale.

Cette architecture permet au réseau de prendre en compte le contexte temporel et les dépendances entre les instants précédents et actuels, ce qui est particulièrement utile pour des tâches impliquant des séquences temporelles. Les paramètres A, B et C sont appris pendant l'entraînement du réseau afin d'optimiser la performance du modèle.

### 5.4.1 Pourquoi les réseaux de neurones récurrents ?

Les RNN ont été créés parce qu'il y avait quelques problèmes dans le réseau de neurones à anticipation :

- Ne peut pas gérer les données séquentielles
- Ne considère que l'entrée actuelle
- Impossible de mémoriser les entrées précédentes

La solution à ces problèmes est le RNN. Un RNN peut :

- ✓ Gérer des données séquentielles.
- ✓ Accepter les données d'entrée actuelles et les entrées précédemment reçues.
- ✓ Les RNN peuvent mémoriser les entrées précédentes grâce à leur mémoire interne.

En fin de compte, les réseaux de neurones récurrents (RNN) sont des types de réseaux de neurones artificiels conçus pour traiter des données séquentielles, telles que du texte, des sons, des vidéos et des signaux temporels. Contrairement aux réseaux de neurones classiques, qui prennent une entrée et produisent une sortie, les RNN utilisent une architecture en boucle pour prendre en compte les informations contextuelles des données précédentes lors de la prise de décision.

Les RNN disposent d'une mémoire interne, appelée état caché, qui stocke les informations précédentes et les utilise pour guider le traitement de la prochaine entrée. Ils sont particulièrement utiles pour les tâches de traitement du langage naturel, comme la traduction automatique, la génération de texte et la reconnaissance de la parole. Les RNN ont également été utilisés pour la reconnaissance d'images, la prédiction de séries chronologiques et la modélisation de processus biologiques.

En somme, les réseaux de neurones récurrents sont une approche puissante pour traiter des données séquentielles en prenant en compte les informations contextuelles des données précédentes, ce qui les rend particulièrement adaptés pour les tâches de traitement du langage naturel et d'autres applications de reconnaissance de séquences.

### 5.4.2 Types de réseau de neurones récurrents

Les réseaux de neurones récurrents (RNN) sont classés en plusieurs types, chacun ayant des architectures et des caractéristiques distinctes pour traiter les données séquentielles. Voici quelques exemples :

- ✓ **Les réseaux de neurones récurrents standard (vanilla RNN)**

Il s'agit de la forme la plus simple de RNN. Dans cette architecture, chaque neurone prend l'entrée actuelle ainsi que l'état caché précédent et produit une sortie. Cependant, les RNN

standards souffrent souvent de problèmes de vanishing gradients, qui peuvent limiter leur capacité à capturer des informations à long terme.

### ✓ Les réseaux de neurones récurrents à portes (gated RNN)

Ce type de RNN utilise des portes pour contrôler le flux d'informations dans le réseau. Les réseaux de neurones récurrents les plus populaires de cette catégorie sont les réseaux LSTM (Long Short-Term Memory) et les réseaux GRU (Gated Recurrent Unit).

### ✓ Les réseaux de neurones récurrents bidirectionnels (bidirectionnel RNN)

Ces réseaux traitent les données séquentielles à la fois dans l'ordre direct et dans l'ordre inverse. Cette approche permet aux réseaux de capturer les informations contextuelles des données passées et futures.

### ✓ Les réseaux de neurones récurrents convolutifs (convolutional RNN)

Cette architecture hybride combine les convolutions, qui sont couramment utilisées dans le traitement d'images, avec les réseaux de neurones récurrents (RNN). Les RNN convolutifs sont particulièrement adaptés au traitement de données séquentielles à haute résolution, telles que les signaux audio ou vidéo.

Chaque type de RNN présente ses propres avantages et inconvénients, et le choix d'un type de RNN dépendra de la tâche à accomplir et des caractéristiques des données à traiter. Les RNN offrent une grande flexibilité dans le traitement de données séquentielles et peuvent s'adapter à une variété d'applications.

## 5.5 Réseaux de neurones convolutionnels (CNN)

Les réseaux de neurones convolutionnels sont une classe de réseaux de neurones artificiels qui sont particulièrement efficaces pour traiter des données d'entrée telles que des images. Ils fonctionnent en appliquant une série de filtres de convolution à l'image d'entrée, ce qui permet de capturer des caractéristiques de bas niveau telles que des bords et des coins, puis de combiner ces caractéristiques en des fonctionnalités plus abstraites à mesure que les données progressent dans le réseau. Les CNN sont largement utilisés pour des tâches telles que la classification d'images, la détection d'objets et segmentation sémantique. (Voir figure 2.14).

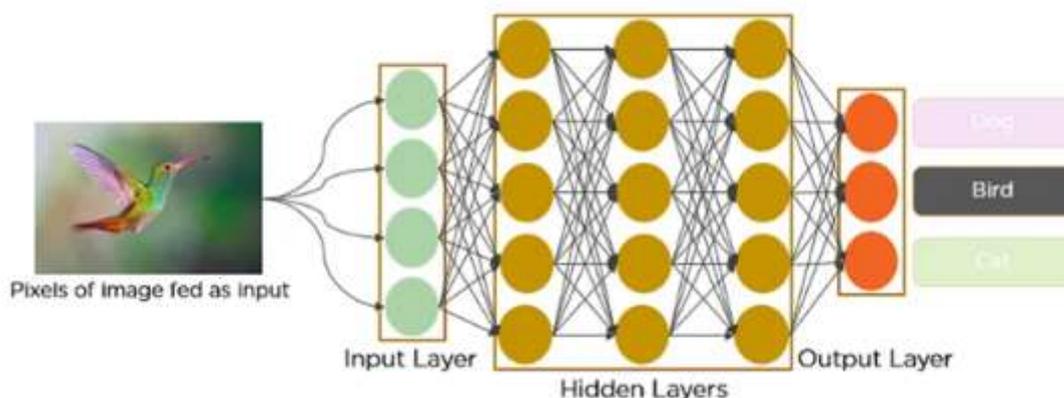


Figure 2.14: Réseau de neurones convolutifs pour identifier l'image d'un oiseau.

### 5.5.1 Pourquoi les réseaux de neurones convolutionnels ?

Les réseaux de neurones convolutionnels (CNN) ont été développés spécifiquement pour traiter les images, mais ils peuvent désormais être utilisés dans divers domaines, tels que la reconnaissance de séquences temporelles.

Leur efficacité dans la reconnaissance d'images réside dans leur capacité à détecter des motifs locaux tels que les contours, les textures et les motifs. Les CNN utilisent des filtres qui analysent chaque région de l'image pour en extraire les caractéristiques pertinentes.

En outre, les CNN peuvent apprendre des représentations hiérarchiques de l'image, en construisant des caractéristiques plus complexes à partir des caractéristiques plus simples détectées dans les couches de convolution précédentes. Cette capacité permet d'obtenir des informations abstraites et précises sur l'image.

En résumé, les CNN sont devenus un outil essentiel en vision par ordinateur et dans d'autres domaines en raison de leur capacité à détecter des motifs locaux, à apprendre des représentations hiérarchiques de l'image et à fournir des résultats précis pour diverses tâches liées à l'image.

### 5.5.2 Types de réseaux de neurones convolutionnels

Il existe plusieurs types de réseaux de neurones convolutionnels (CNN) qui ont été développés pour répondre à des besoins spécifiques en vision par ordinateur. Voici quelques-uns des types de CNN les plus couramment utilisés :

#### ✓ Le réseau de neurones convolutionnel standard

Il s'agit du type le plus simple de CNN, composé de couches de convolution, de pooling et de couche entièrement connectée.

#### ✓ Le réseau de neurones convolutionnel profond (Deep CNN)

Ce type de réseau est plus profond que le CNN standard et comporte généralement plusieurs couches convolutionnelles, suivies de couches de pooling et de couches entièrement connectées. Les CNN profonds ont été utilisés avec succès pour la reconnaissance d'images, la segmentation d'images, la détection d'objets et d'autres tâches.

#### ✓ Le réseau de neurones convolutionnel en cascade (Cascade CNN)

Ce type de réseau utilise une série de CNN qui sont entraînés de manière séquentielle pour améliorer les performances. Les résultats d'un CNN sont utilisés comme entrée pour le CNN suivant, ce qui permet d'améliorer la précision des prédictions.

#### ✓ Le réseau de neurones convolutionnel résiduel (ResNet)

Ce type de réseau a été développé pour résoudre le problème de la dégradation de la performance des réseaux de neurones plus profonds. Les résidus sont introduits dans les connexions entre les couches, ce qui permet aux informations de contourner certaines couches pour atteindre plus rapidement les couches de sortie.

#### ✓ Le réseau de neurones convolutionnel récurrent (CRNN)

Ce type de réseau utilise une combinaison de couches de convolution et de couches récurrentes pour traiter des données séquentielles telles que la parole ou les données de capteurs temporels. Ces différents types de CNN sont adaptés à des tâches spécifiques en vision par ordinateur, et chacun offre des avantages et des inconvénients selon la tâche à effectuer.

## 5.6 Réseaux de neurones modèles adversaires génératifs (GAN)

Les réseaux de neurones modèles adversaires génératifs est un type de réseau de neurones artificiels qui peut générer de nouvelles données en apprenant à partir d'un ensemble de données existant. Il se compose de deux réseaux de neurones, le générateur et le discriminateur, qui s'entraînent en s'opposant l'un à l'autre. Le générateur crée des données synthétiques, tandis que le discriminateur évalue si ces données sont réelles ou synthétiques. Au fil de temps, le

générateur apprend à créer des données de plus en plus réalistes en réponse aux commentaires du discriminateur, jusqu'à ce que le discriminateur ne puisse plus faire la différence entre les données synthétiques et réelles. (Voir figure 2.15).

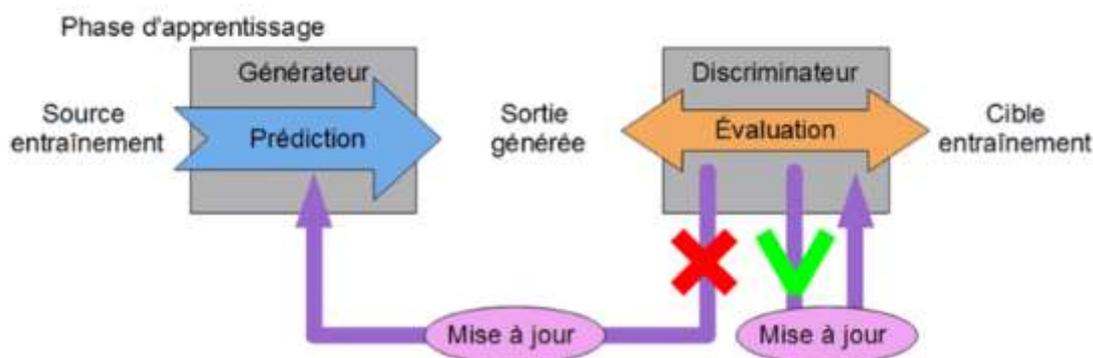


Figure 2.15: Schéma simplifié du processus d'entraînement du GAN conditionnel.

### 5.6.1 Pourquoi les réseaux de neurones modèles adversaires génératifs ?

Les réseaux de neurones GAN sont populaires pour plusieurs raisons :

- Ils peuvent générer des données synthétiques de haute qualité qui sont très similaires aux données réelles, ce qui peut aider à améliorer la précision des modèles de machine Learning.
- Les GAN permettent d'augmenter la quantité de données d'entraînement disponibles pour les modèles de machine Learning, ce qui peut être particulièrement utile lorsque les données sont rares ou coûteuses à collecter.
- Les GAN peuvent également être utilisés pour la création de contenu multimédia, comme des images, des vidéos et de la musique, ce qui peut avoir des implications dans des domaines tels que le cinéma, les jeux vidéo et l'art.
- Les GAN ont permis des avancées significatives dans le domaine de développement de la recherche, notamment en générant des images réalistes à partir de descriptions textuelles, ce qui peut avoir des implications importantes pour la recherche en intelligence artificielle et en informatique cognitive.

En résumé, les GAN sont devenus populaires en raison de leur capacité à générer des données synthétiques de haute qualité, à augmenter la quantité de données d'entraînement et à créer du contenu multimédia, tout en offrant des possibilités de recherche passionnantes.

### 5.6.2 Types de réseaux de neurones modèles adversaires génératifs

Il existe plusieurs types de GAN, chacun ayant des fonctionnalités et des applications spécifiques. Les types les plus courants sont :

#### ✓ GAN classique

C'est le premier type de GAN proposé par Ian Goodfellow en 2014. Il se compose de deux réseaux de neurones : un générateur et un discriminateur, qui sont entraînés en parallèle pour améliorer la qualité de la génération de données synthétiques.

#### ✓ DCGAN (Deep convolutional GAN)

C'est une variante du GAN classique qui utilise des couches de convolution profondes pour améliorer la qualité de la génération d'images. Les DCGAN sont souvent utilisés pour la synthèse d'images haute résolution.

### ✓ WGAN (Wasserstein GAN)

Contrairement aux GAN classiques qui utilisent la divergence de Kullback-Leibler pour mesurer la distance entre les distributions de probabilité, les WGAN utilisent la distance de Wasserstein. Cela permet d'obtenir des gradients plus stables et de meilleures performances de génération.

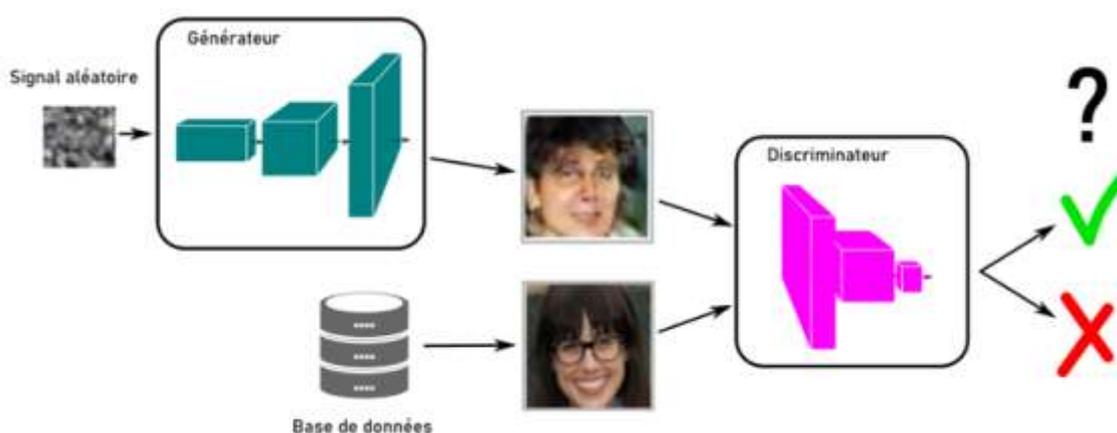
### ✓ CGAN (Conditional GAN)

Ce type de GAN permet de contrôler la génération de données synthétiques en ajoutant des informations conditionnelles. Par exemple, pour la synthèse d'images de visage, on peut fournir des informations sur l'âge, le sexe, la couleur des yeux, etc. pour générer des images correspondantes.

### ✓ Cycle GAN

Ce type de GAN permet la traduction de données synthétiques d'un domaine à un autre. Par exemple, on peut utiliser un Cycle GAN pour traduire des images de chevaux en images de zèbres, sans avoir besoin d'un pair d'image correspondante pour l'apprentissage.

En résumé, chaque type de GAN a ses propres avantages et est adapté à des tâches de génération de données synthétiques spécifiques. Il est donc important de choisir le type de GAN le plus approprié en fonction des besoins de chaque projet. (Voir figure 2.16).



**Figure 2.16:** Application de génération de visages dans une image avec un GAN. (Clayton, 2020).

## 6 le problème de sur-apprentissage et sous-apprentissage

Le sur-apprentissage et le sous-apprentissage sont deux problèmes fréquents dans le domaine de l'apprentissage automatique. Le sur-apprentissage se produit lorsque le modèle est trop complexe par rapport aux données d'apprentissage, ce qui peut conduire à une très bonne performance sur les données d'apprentissage, mais une performance médiocre sur les données de test. En revanche, le sous-apprentissage se produit lorsque le modèle est trop simple par rapport aux données d'apprentissage et ne parvient pas à capturer les modèles importants dans les données, conduisant à une performance médiocre sur les données d'apprentissage et de test. Pour résoudre ces problèmes, il est important de trouver un équilibre entre la complexité du modèle et la quantité et la qualité des données d'apprentissage. Des techniques telles que la régularisation et l'augmentation des données peuvent être utilisées pour éviter le sur-apprentissage, tandis que l'utilisation de modèles plus complexes ou d'une meilleure qualité de données peut aider à résoudre le sous-apprentissage.

Les facteurs déterminant le rendement d'un algorithme d'apprentissage automatique ou profond sont sa capacité à :

1. Réduire l'erreur dans la phase d'apprentissage.
2. Réduire l'écart entre le résultat d'apprentissage et l'erreur de test.

Ces deux facteurs correspondent aux deux enjeux centraux d'apprentissage automatique qui sont :

- ✓ Sur-apprentissage « overfitting ».
- ✓ Sous-apprentissage « underfitting »

### 6.1 Le sur-apprentissage

A titre exemple, l'équipe de football illustre bien le problème de la généralisation en machine Learning. Le sur-apprentissage (overfitting) est un phénomène qui se produit lorsque le modèle d'apprentissage automatique est trop complexe par rapport aux données d'apprentissage. Cela peut conduire à une performance très élevée sur les données d'apprentissage, mais à une performance médiocre sur les données de test, car le modèle a appris à mémoriser les données d'apprentissage plutôt qu'à généraliser à de nouvelles données.

Dans l'exemple de l'équipe de football, cela reviendrait à juger tous les joueurs de l'équipe comme étant de mauvais joueurs, même si certains d'entre eux sont très compétents, simplement parce que l'équipe n'a pas bien performé lors des matchs de la ligue. De même, le modèle de machine Learning peut mal généraliser et sous-performer sur de nouvelles données, même si certaines parties du modèle sont très performantes. Pour résoudre le problème du sur-apprentissage, des techniques telles que la régularisation, la réduction de la complexité du modèle ou l'augmentation des données peuvent être utilisées pour aider le modèle à généraliser correctement (Géron, 2017).

### 6.2 Le sous-apprentissage

Cet exemple illustre également un autre problème courant en machine Learning appelé sous-apprentissage (underfitting). Lorsque le modèle est trop simple ou que le nombre de données est insuffisant, le modèle peut ne pas être en mesure de capturer la structure sous-jacente des données, ce qui peut conduire à des performances médiocres à la fois sur les données d'apprentissage et de test.

Dans le contexte de l'équipe de football, cela signifierait que l'équipe est jugée comme étant faible, même si nous ne disposons pas de suffisamment d'informations sur ses joueurs et ses performances pour en juger de manière fiable. De même, un modèle de machine Learning mal adapté peut ne pas être en mesure de capturer les caractéristiques clés des données, conduisant ainsi à des performances médiocres.

Pour résoudre le problème de sous-apprentissage, il est possible d'augmenter la complexité du modèle, d'ajouter des fonctionnalités supplémentaires ou de collecter davantage de données pour améliorer la performance du modèle. En résumé, le sous-apprentissage est un autre problème courant en machine Learning, qui peut être résolu en utilisant des techniques adaptées pour améliorer la capacité du modèle à capturer la structure sous-jacente des données (Géron, 2017).

Pour éviter ces deux problèmes, cités ci-dessus, plusieurs solutions ont été proposées dans la littérature (Géron, 2017). Concernant la résolution du problème de sur-apprentissage, nous pouvons :

1. Utiliser plus des données durant la phase d'apprentissage.
2. Réduire le bruit dans les données d'apprentissage (fixer les erreurs dans la base de données).

Pour résoudre le problème de sous-ajustement, nous pouvons :

1. Fournir une meilleure base de données à l'algorithme d'apprentissage.
2. Réduire les contraintes sur le modèle.

La figure suivante fournit des graphiques illustrant tous les états (Voir figure 2.17).

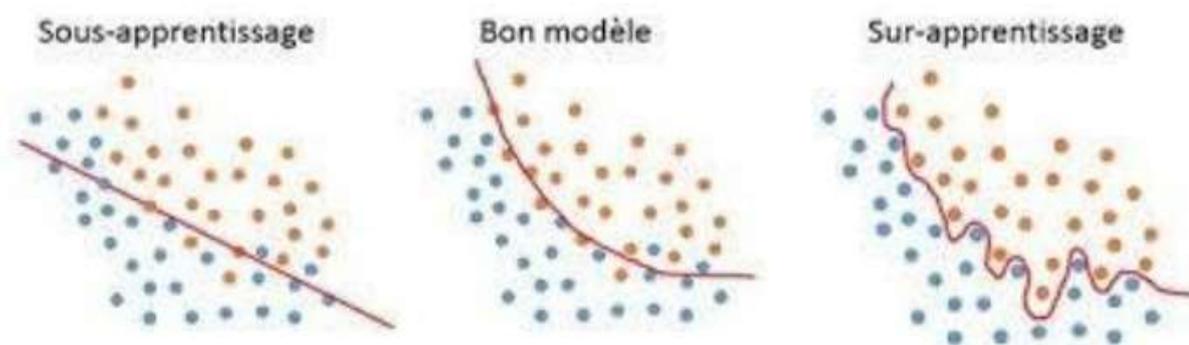


Figure 2.17: Les graphes de chaque état (Géron, 2017).

### 7 Comparaison entre l'apprentissage automatique et l'apprentissage en profondeur

	L'apprentissage automatique	L'apprentissage en profondeur
<b>Dépendances des données</b>	Une performance excellente avec des petites bases de données.	Une performance excellente avec les BIG-DATA <sup>1</sup> .
<b>Dépendances du matériel</b>	Travail sur une machine faible.	Besoin d'une machine fort avec a un CPU <sup>2</sup> fort aussi.
<b>Les caractéristiques des données</b>	Besoin de comprendre les caractéristiques des données.	Ne besoin pas de comprendre les caractéristiques des données.
<b>Le temps d'exécution</b>	Quelque minute à des heures.	Des semaines.
<b>Interprétabilité</b>	Certains algorithmes sont faciles (Arbre de disions) et d'autres sont impossibles (SVM, XGBoost <sup>3</sup> ).	Difficile à impossible.

Tableau 2.1: Comparaison entre l'apprentissage automatique et l'apprentissage en profondeur (Grossfeld, 2020).

### 8 Conclusion

L'apprentissage automatique est une discipline de l'intelligence artificielle qui permet aux ordinateurs d'apprendre à partir de données sans nécessiter une programmation explicite. Grâce à l'essor de la puissance de calcul, l'abondance de données disponibles et les avancées des techniques de modélisation, cette approche a connu une popularité croissante ces dernières années. Elle trouve des applications dans divers domaines tels que la prédiction des tendances du marché ou la reconnaissance vocale, pour n'en citer que quelques exemples.

Cependant, comme pour toute technologie, l'apprentissage automatique présente également des risques, tels que la discrimination algorithmique et la violation de la vie privée. Il est donc essentiel de l'utiliser de manière responsable et éthique, en prenant en compte des implications sociales, éthiques et juridiques.

En résumé, l'apprentissage automatique est une technologie en constante évolution qui offre de nombreuses opportunités pour améliorer notre vie quotidienne, mais qui nécessite également une réflexion approfondie sur ses implications éthiques et sociales.

Dans le prochain chapitre, nous aborderons la conception du système, en mettant en évidence les différentes phases proposées ainsi que l'ensemble de données utilisé. Nous explorerons les détails de chaque phase. De plus, nous mettrons en évidence les caractéristiques clés de l'ensemble de données utilisé et expliquerons comment il a été collecté et préparé pour l'entraînement des modèles.

---

<sup>1</sup> **BIG-DATA** : un ensemble très volumineux de données qu'aucun outil classique de gestion de base de données ou de gestion de l'information ne peut vraiment travailler (Bremme, 2015) ;

<sup>2</sup> **CPU** : Un processeur « central processing unit ».

<sup>3</sup> **XGBoost** : est une bibliothèque logicielle open source permettant de mettre en oeuvre des méthodes de Gradient boosting, Le Boosting de Gradient est un algorithme d'apprentissage supervisé dont le principe est de combiner les résultats d'un ensemble de modèles plus simple et plus faibles afin de fournir une meilleure prédiction (Cayla, 2018).

# **Chapitre 3 : Conception et Implémentation**

### 1 Introduction

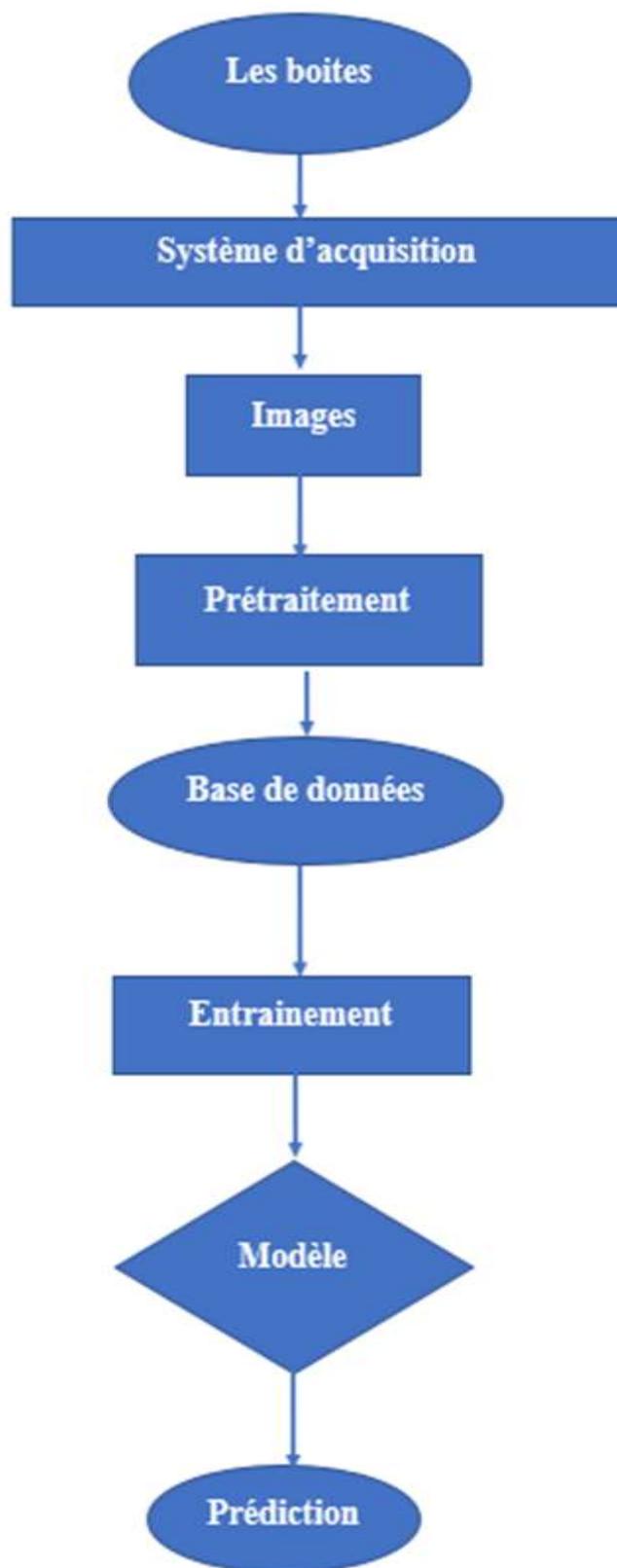
Les avancées dans l'évaluation non-destructive de la qualité des produits industriels ont été réalisées grâce aux technologies de l'IA. Au fil des décennies, plusieurs méthodes ont été développées pour évaluer la qualité des produits industriels sans les endommager. Dans les régions du Moyen-Orient et de l'Afrique du Nord, les boîtes d'emballage jouent un rôle économique crucial. Par conséquent, l'utilisation d'un système automatisé de classement non-destructif est essentielle pour améliorer l'efficacité, la précision et réduire les coûts de traitement.

Dans notre étude, nous avons adopté une approche innovante en utilisant des images pour évaluer la qualité des boîtes sans causer de dommages aux produits. Nous avons appliqué des techniques de traitement d'image et des méthodes d'apprentissage automatique profond pour obtenir des résultats optimaux dans la classification des boîtes et détecter efficacement les défauts. Cette approche permet d'améliorer la précision du processus d'évaluation et de faciliter la détection précoce des problèmes de qualité, contribuant ainsi à l'amélioration de l'efficacité globale de la chaîne de production.

### 2 Concept générale

En cas générale, notre système d'évaluation de la qualité des boîtes suivra quatre phases, telles qu'illustrées dans la figure 3.1.

Le système commence par la phase d'acquisition, où nous collectons des images pour chaque boîte, en leur attribuant une classe respective. Ensuite, nous appliquons un prétraitement sur les images, ce qui aboutit à la création d'une base de données. Dans la troisième phase, nous alimentons notre système avec la base de données précédente afin d'apprendre et de développer un nouveau modèle capable de prédire la classe des nouvelles images de boîtes lors de la phase de prédiction.



**Figure 3.1:** Concept général de système proposé.

### 2.1 Système d'acquisition

Le système que nous avons mis en place est conçu pour traiter des images couleur de boîtes. Plus précisément, notre étude se concentre sur le cultivar "boîte d'emballage alimentaire", une variété des boîtes qui est très populaire en raison de son utilisation. Les données collectées pour chaque boîte comprennent quatre images couleur (une pour chaque face de la boîte), les dimensions (largeur, longueur et hauteur) et la rigidité de chaque boîte.

L'éclairage revêt une importance primordiale lors de la capture d'images pour évaluer la qualité des aliments. Les conditions d'éclairage peuvent considérablement influencer la qualité de l'image obtenue. En utilisant un éclairage de haute qualité, nous pouvons simplifier et accélérer les étapes ultérieures de traitement d'image, ce qui permet de réduire les coûts du système de traitement d'image.

Nous avons également opté pour un fond de couleur uniforme sans texture, en choisissant un fond blanc. Aucune contrainte n'est imposée quant à la taille de l'image. Cette approche vise à garantir une visualisation claire et précise des aliments, favorisant ainsi une évaluation plus fiable de leur qualité.

### 2.2 Prétraitement

Le prétraitement joue un rôle essentiel dans les tâches de classification d'images car il permet de prendre en compte des aspects tels que la forme, la taille, les bruits et les pixels. Afin d'optimiser les performances des algorithmes de classification, nous appliquons des techniques de traitement d'image qui permettent de préparer les images pour les étapes ultérieures du traitement. Ces techniques incluent la normalisation des couleurs, la réduction du bruit, la mise à l'échelle et la normalisation de la taille des images. Le prétraitement permet ainsi d'améliorer la qualité des données d'entrée et de réduire les variations indésirables, facilitant ainsi la classification précise des images.

Dans le domaine industriel, le traitement d'image se compose généralement de trois étapes :



**Figure 3.2:** Schéma des étapes du traitement d'image.

L'amélioration de l'image vise à améliorer la qualité visuelle de l'image en réduisant le bruit indésirable et en augmentant le contraste pour une meilleure lisibilité. L'extraction des caractéristiques de l'image consiste à identifier et à isoler les informations importantes et discriminantes de l'image, telles que les contours, les textures ou les motifs, qui permettent de représenter de manière compacte les caractéristiques visuelles de l'image. Enfin, la classification des caractéristiques de l'image utilise ces informations pour attribuer une classe ou une étiquette à l'image en fonction de l'objectif spécifique de l'application, telle que la reconnaissance de formes, la classification de défauts. Ces étapes sont essentielles pour traiter les images de manière efficace et obtenir des résultats précis dans diverses tâches d'analyse d'images.

### 2.2.1 Segmentation d'image

La segmentation d'images est un processus essentiel qui vise à découper et analyser les caractéristiques d'une image afin de diviser celle-ci en régions pertinentes liées à des objets spécifiques ou des zones d'intérêt. Cette étape revêt une grande importance dans le domaine du traitement d'image, car la précision de la segmentation impacte directement la qualité des informations extraites par la suite. Une segmentation précise des objets présents dans l'image est nécessaire pour mesurer et analyser ces objets de manière adéquate, ce qui a une incidence directe sur les tâches de classification et de reconnaissance des objets. Une segmentation précise permet ainsi une meilleure interprétation et compréhension de l'image, et facilite l'extraction de caractéristiques pertinentes pour des applications ultérieures.

Le processus de segmentation d'une image couleur d'une boîte peut être divisé en plusieurs étapes. Tout d'abord, l'image couleur est convertie en une image en niveaux de gris pour simplifier le traitement ultérieur. Ensuite, le seuillage d'image (thresholding) est appliqué avec un intervalle de valeur de seuil allant de  $[0, 0, 0]$  à  $[125, 125, 125]$ . Ce processus permet de créer une image en noir et blanc où les pixels dont les valeurs sont supérieures au seuil sont considérés comme blancs, tandis que les pixels dont les valeurs sont inférieures au seuil sont considérés comme noirs. Cette étape permet de simplifier l'image en la réduisant à deux couleurs, ce qui facilite le processus de segmentation.

La figure suivante illustre les différentes étapes du processus de segmentation d'une image couleur d'une boîte.



**Figure 3.3:** Images-en différent couleur d'une boîte.

### 2.2.2 Analyse d'image

Dans le processus d'analyse d'image, notre système se divise en trois étapes clés pour extraire des informations quantitatives.

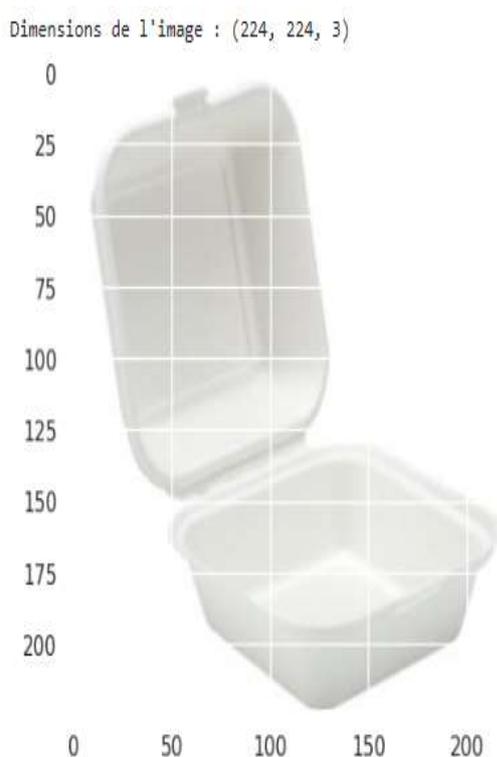
1. La détection des contours.

2. La mesure de taille des objets détectés.
3. La décomposition en composants de couleur.

La première étape consiste à utiliser une image binaire obtenue par la segmentation pour détecter les contours des objets d'intérêt. Cette étape est importante car elle permet d'isoler les objets de l'image pour une analyse plus approfondie.

Ensuite, la mesure de la taille est effectuée en utilisant les contours détectés pour calculer la surface, la circonférence et d'autres propriétés géométriques des objets. Cette étape est importante car elle permet de quantifier les caractéristiques des objets, ce qui peut être utile pour la classification et la reconnaissance.

Enfin, la décomposition en composants de couleur permet de séparer les objets en fonction de leur couleur. Cette étape est importante car elle permet de distinguer les objets qui ont des couleurs similaires mais des formes différentes, ce qui peut être utile pour la classification et la reconnaissance.



**Figure 3.4:** Les dimensions d'une boîte.

D'autre part, la dimension des boîtes est considérée comme un critère clé pour classer leur qualité. Pour mesurer la dimension, on utilise la hauteur de la boîte englobante de la zone de boîte segmentée. Cette hauteur est obtenue en dessinant un rectangle englobant la zone de contour détectée auparavant. Cela nous permet d'avoir une estimation précise de la taille de la boîte.



**Figure 3.5:** Contour sur la dimension de la boîte.

Le processus commence par parcourir l'image pixel par pixel pour chaque canal de couleur, et calculer trois valeurs :

### **1. la couleur minimale de l'objet**

Pour obtenir la couleur minimale dans notre approche, nous commençons par initialiser la valeur du premier pixel de l'objet en tant que valeur de référence. Ensuite, pour chaque pixel de l'objet, nous vérifions si sa valeur est inférieure à la valeur de référence actuelle. Si c'est le cas, nous mettons à jour la valeur de référence avec la valeur du pixel actuel. Ce processus se répète pour tous les pixels de l'objet afin de trouver la couleur minimale présente dans celui-ci. Cela nous permet d'identifier la teinte la plus sombre ou la plus basse dans l'objet analysé.

### **2. la couleur maximale de l'objet**

Pour obtenir la couleur maximale dans notre approche, nous commençons par initialiser la valeur du premier pixel de l'objet en tant que valeur de référence. Ensuite, pour chaque pixel de l'objet, nous vérifions si sa valeur est supérieure à la valeur de référence actuelle. Si c'est le cas, nous mettons à jour la valeur de référence avec la valeur du pixel actuel. Ce processus se répète pour tous les pixels de l'objet afin de trouver la couleur maximale présente dans celui-ci. Cela nous permet d'identifier la teinte la plus claire ou la plus élevée dans l'objet analysé.

### **3. la couleur moyenne de l'objet :**

Pour calculer la valeur de la couleur moyenne de l'objet, nous utilisons la formule suivante : la somme de toutes les valeurs de couleur de chaque pixel de l'objet, divisée par le nombre total de pixels présents dans l'objet. Cette formule nous permet d'obtenir la valeur moyenne de couleur de l'ensemble des pixels de l'objet. En effectuant cette opération, nous obtenons une estimation de la couleur centrale ou représentative de l'objet, qui peut être utilisée pour caractériser sa teinte globale.

### **4. Nombre des pixels spot :**

On calcul la valeur de "Variance" puis on calcul "l'écart type" d'objet à partir la "variance".

Ecart type = racine carré (Variance)

Puis on test si la valeur du pixel actuel vérifie l'un des 2 l'équation suivante on compte comme "pixel spot".

$Val > \text{couleur moyen} + 2 * \text{écart type}$

$Val < \text{couleur moyen} - 2 * \text{écart type}$

Enfin, la dernière étape de la phase du prétraitement c'est la concaténation des descripteurs.

Le vecteur descripteur des caractéristiques finales de la boîte est structuré comme suit :

La concaténation de 2 parties de caractéristiques :

### Partie 1 :

- Identificateur de la boîte
- Démentions
- La forme

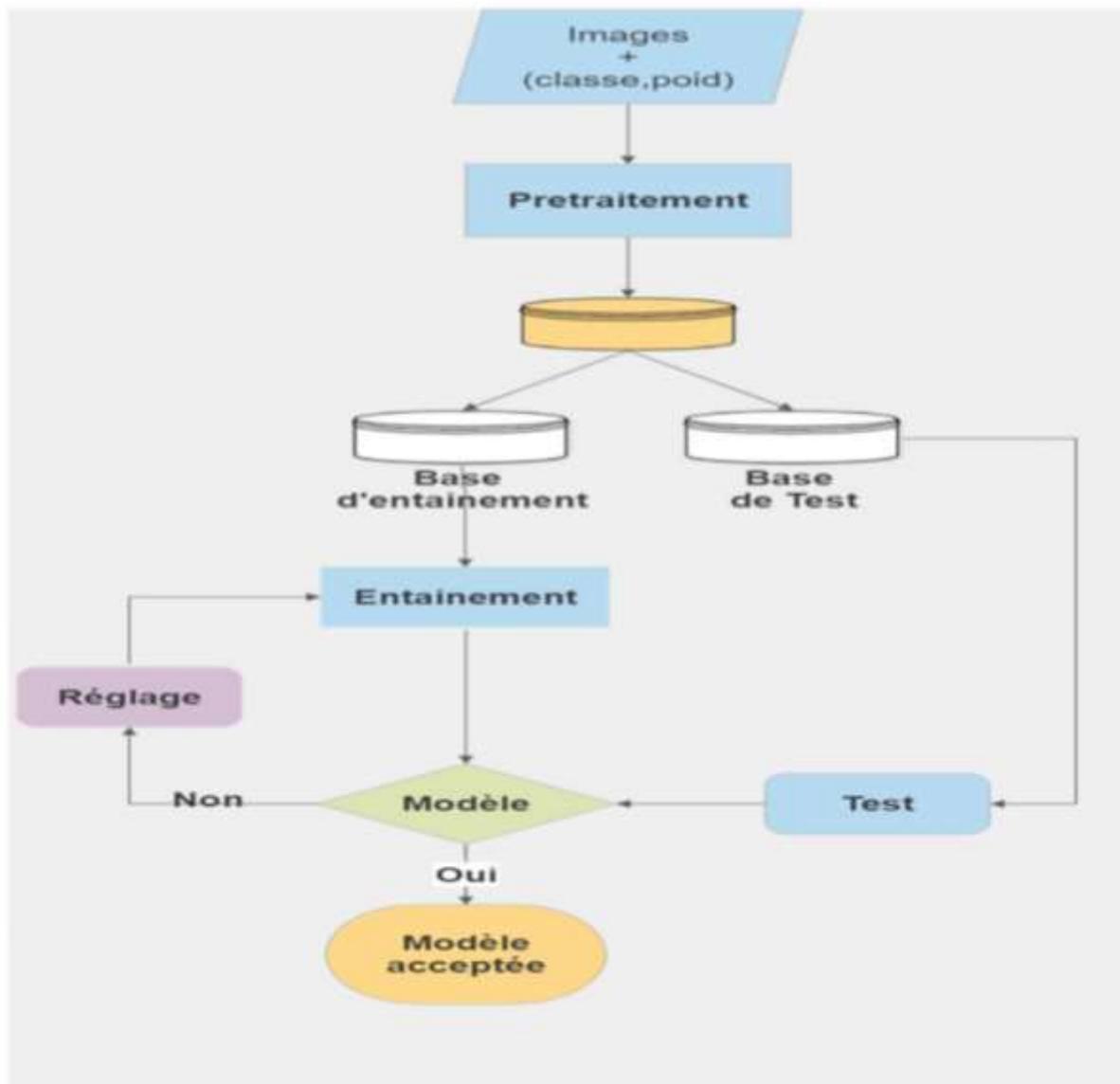
### Partie 2 :

Les caractéristiques des 4 faces de la boîte peuvent être combinées de la manière suivante :

- Pour la caractéristique de couleur minimale (pour les canaux R, G, B) : Prendre la valeur minimale de "Couleur min" parmi les 4 vecteurs.
- Pour la caractéristique de couleur maximale (pour les canaux R, G, B) : Prendre la valeur maximale de "Couleur max" parmi les 4 vecteurs.
- Pour la caractéristique de couleur moyenne (pour les canaux R, G, B) : Calculer la somme de "Couleur moyen" des 4 vecteurs et la diviser par 4.
- Pour le pourcentage de pixels spot : Calculer le total des pixels spot parmi les 4 vecteurs et le diviser par le total des pixels d'objet parmi les 4 vecteurs. Le total des pixels spot représente la somme des valeurs de "pixel spot" des 4 vecteurs, tandis que le total des pixels d'objet représente la somme des valeurs de "pixels d'objet" des 4 vecteurs.

## 2.3 Entraînement

Au cœur du processus d'apprentissage profond se trouve l'entraînement du modèle, qui constitue l'étape clé pour permettre au modèle de "s'entraîner" à résoudre une tâche spécifique. Dans notre cas, nous utilisons la partie de l'ensemble de données réservée à l'entraînement pour apprendre à notre modèle à différencier les boîtes. Cette étape est essentielle pour que le modèle puisse identifier les caractéristiques importantes des données et les utiliser pour effectuer des prédictions précises sur de nouvelles données.



**Figure 3.6:** Diagramme de la phase de la création de modèle.

Cette figure illustre les différentes étapes de la phase d'entraînement du modèle proposé dans notre système.

La première étape consiste à diviser l'ensemble de données en deux sous-ensembles distincts, puis à sélectionner le type de modèle une fois que les étapes de préparation des données sont terminées. L'entraînement du modèle constitue l'étape clé de cette phase, car c'est là que l'apprentissage véritable a lieu. Une fois que le modèle est entraîné, il doit être testé pour déterminer s'il est capable de fournir des prédictions précises sur de nouvelles données. Si l'évaluation est réussie, nous pouvons passer à l'étape de réglage des hyperparamètres pour optimiser les performances du modèle.

Dans cette partie, nous mettons en pratique les étapes du système conçu. Nous utilisons des frameworks tels que TensorFlow, PyTorch et scikit-learn, ainsi que des outils tels que Jupyter Notebook, NumPy et Pandas pour le développement. Nous effectuons différentes expériences en utilisant les méthodes proposées, en ajustant les hyperparamètres et en évaluant les performances. Les résultats obtenus sont évalués à l'aide de métriques telles que la précision ainsi que des visualisations graphiques. Cette mise en œuvre nous permet d'analyser les performances du système dans diverses conditions et de prendre des décisions d'optimisation. En résumé, cette partie présente les détails de la mise en œuvre, les expériences réalisées et les résultats obtenus, permettant ainsi une évaluation complète de notre système.

### 3 Les couches utilisées dans notre programme

- 1) **La première couche** est une couche de convolution qui prend en entrée une image de taille  $(150, 150, 3)$  (3 canaux de couleur pour RGB) et applique 32 filtres de taille  $(3, 3)$  pour extraire des caractéristiques de bas niveau de l'image. La fonction d'activation utilisée est Relu.
- 2) **La deuxième couche** est une couche de max-pooling qui réduit la taille spatiale de l'image par un facteur de 2 en ne conservant que les valeurs maximales dans chaque fenêtre de taille  $(2, 2)$  de l'image.
- 3) **La troisième couche** est une autre couche de convolution qui prend en entrée l'image réduite en taille par la couche de max-pooling et applique 64 filtres de taille  $(3, 3)$  pour extraire des caractéristiques de plus haut niveau.
- 4) **La quatrième couche** est à nouveau une couche de max-pooling qui réduit la taille spatiale de l'image de moitié.
- 5) **La cinquième couche** est une couche de convolution avec 128 filtres de taille  $(3, 3)$  qui extrait des caractéristiques encore plus complexes de l'image.
- 6) **La sixième couche** est une couche de max-pooling qui réduit la taille spatiale de l'image de moitié.
- 7) **La septième couche** est une couche de convolution avec 128 filtres de taille  $(3, 3)$  qui extrait des caractéristiques encore plus complexes de l'image.
- 8) **La huitième couche** est une couche de max-pooling qui réduit la taille spatiale de l'image de moitié.
- 9) **La neuvième couche** est une couche de mise en forme qui transforme la sortie de la couche précédente en un vecteur 1D.
- 10) **La dixième couche** est une couche dense qui contient 512 neurones et utilise la fonction d'activation Relu.
- 11) **La onzième et dernière couche** est une couche dense de sortie qui contient un seul neurone et utilise la fonction d'activation sigmoïde pour prédire la classe binaire (0 ou 1) de l'image.

### 4 Application et outils d'implémentation

Les frameworks d'apprentissage automatique jouent un rôle essentiel dans la révolution de l'intelligence artificielle, en fournissant des outils pré-implémentés et des modèles qui simplifient le prétraitement, la classification et l'évaluation des données. Sans ces

frameworks, les scientifiques rencontreraient des difficultés considérables pour travailler sur des tâches d'apprentissage automatique complexes. Grâce à ces outils, les utilisateurs peuvent se concentrer sur la conception de modèles et l'analyse des résultats, plutôt que sur la mise en œuvre détaillée des algorithmes.

Il existe plusieurs options pour implémenter des programmes python, dont les plus populaires sont :

✓ **Python IDLE**

Un environnement de développement intégré (IDE) standard livré avec la distribution python, proposant des fonctionnalités telles que la coloration syntaxique, l'auto-complétions, le débogage, etc.

✓ **PyCharm**

Un IDE python, très populaire offrant des fonctionnalités avancées comme l'analyse de code, la complétion de code, la refonte de code, le support de Django, etc.

✓ **Anaconda**

Une distribution python complète avec de nombreux packages scientifiques et d'analyse de données préinstallés, tels que NumPy, Pandas, Matplotlib, Scipy, etc.

✓ **Jupyter Notebook**

Un environnement de développement interactif pour créer des documents de code, explorer les données, créer des graphiques, etc.

✓ **Visual Studio Code**

Un IDE léger prenant en charge plusieurs langages de programmation, y compris python, et disposant de nombreuses extensions pour une meilleure intégration avec les bibliothèques populaires comme NumPy, Pandas et TensorFlow.

✓ **Pylinstaller**

Un outil pour créer des fichiers exécutables autonomes pour les applications python, facilitant leur déploiement.

✓ **Flash**

Un micro-Framework web pour python permettant de créer des applications web légères et rapides.

✓ **Django**

Un Framework web complet pour python offrant des fonctionnalités comme la gestion de bases de données, l'authentification utilisateur, la gestion de formulaires, etc.

Il existe de nombreuses autres options d'outils et d'applications pour python, en fonction des besoins spécifiques de chaque développeur. Toutefois, ces choix sont parmi les plus populaires et les plus utiles pour les développeurs utilisant python.

### 4.1 Python

Python est un langage de programmation populaire et largement utilisé dans le monde entier. Il a été créé par Guido Van Rossum dans les années 1990 et se distingue par sa syntaxe claire et concise, qui rend le code facile à lire et à comprendre. Python est également connu pour sa grande portabilité, car il peut être utilisé sur différentes plateformes telles que Windows, Mac et Linux. En outre, Python dispose d'une bibliothèque standard étendue qui permet de réaliser de nombreuses tâches courantes, notamment la manipulation de

## Chapitre 3 : Conception et Implémentation

---

fichiers, les opérations mathématiques, la gestion des réseaux et des sockets, et la création d'interfaces graphiques. (Python Programming Language, 2021)

Python est largement utilisé dans de nombreux domaines, notamment dans le développement web, le traitement des données, la science des données, l'apprentissage automatique, l'intelligence artificielle, la visualisation de données, le développement de jeux vidéo et le traitement de l'information et bien plus encore. En raison de ses avantages en termes de simplicité, de portabilité et de richesse de son écosystème, Python est devenu un choix populaire pour les développeurs et les scientifiques des données dans de nombreux domaines.

La communauté des développeurs python est très active et travaille en collaboration pour améliorer le langage et développer de nouvelles bibliothèques et Framework.

Python dispose de nombreuses bibliothèques très populaires et largement utilisées pour différents domaines, notamment :

Bibliothèques	Définition
<b>NumPy</b>	Une bibliothèque pour le calcul numérique avec python, utilisée pour la manipulation de tableaux multidimensionnels, la transformation de Fourier, l'algèbre linéaire, des fonctions mathématiques avancées et des outils pour intégrer le code C/C++.
<b>Pandas</b>	Une bibliothèque pour la manipulation et l'analyse de données avec python, utilisée pour la manipulation de données en tables (lecture, écriture, fusion, agrégation, etc.), cette dernière fournit des structures de données pour la représentation de données tabulaires, des fonctions pour l'agrégation et le traitement de données, ainsi que des outils pour l'importation et l'exportation de données.
<b>Matplotlib</b>	Une bibliothèque pour la visualisation de données avec python, utilisée pour la création de graphiques et de visualisation (lignes, barres, histogrammes, nuages de points, etc.).
<b>Scikit-Learn</b>	Une bibliothèque pour l'apprentissage automatique avec Python, ce dernier offre des outils pour la classification, la

	régression, le clustering, la réduction de dimensionnalité, la sélection de modèle et d'autres tâches d'apprentissage automatique.
<b>TensorFlow</b>	Une bibliothèque pour l'apprentissage en profondeur (Deep Learning) avec Python, ce dernier offre des outils pour la création de réseaux de neurones, l'optimisation de modèles, l'entraînement de modèles et la détection d'anomalies. (Géron, 2017)
<b>PyTorch</b>	Une bibliothèque pour l'apprentissage en profondeur avec python, également utilisée pour la création de modèles de réseaux de neurones artificiels.
<b>Keras</b>	La bibliothèque Keras est une bibliothèque open source populaire pour l'apprentissage automatique et la construction de réseaux de neurones. Elle fournit une interface conviviale et intuitive pour créer, entraîner et déployer des modèles d'apprentissage profond.
<b>Django</b>	Un Framework web complet qui permet de développer des applications web complexes avec une architecture MVC (Modèle-Vue-Contrôleur) robuste.
<b>Beautiful Soup</b>	Une bibliothèque pour le web Scapin en Python qui permet d'extraire des données à partir de pages web en HTML ou XML.
<b>Requests</b>	Une bibliothèque pour l'envoi de requêtes HTTP en Python qui facilite la récupération de données à partir de sites web et d'APIs.
<b>Pygame</b>	Une bibliothèque pour la création de jeux vidéo qui offre des outils pour la gestion des graphiques, du son et de l'interaction utilisateur.
<b>Git</b>	Un système de gestion de version qui permet de suivre les modifications du code source, de travailler en collaboration et de restaurer les versions précédentes.
<b>NumPydoc</b>	Une bibliothèque pour la documentation de code Python, qui fournit des outils pour générer des documents de référence pour les fonctions, les classes et les modules.

<b>Sphinx</b>	Un générateur de documentation pour Python, qui permet de générer des documents HTML, PDF et d'autres formats à partir de documents de code source Python. (Goodfellow, 2016)
---------------	---

**Tableau 3.1:** les bibliothèques très populaires de Python.



**Figure 3.7:** Logo du Python.

### 4.1.1 Quelques faits sur le langage de programmation python

- Python est connu pour son utilisation de l'indentation pour marquer les blocs de code, ce qui est différent de nombreux autres langages de programmation qui utilisent des accolades ou des mots-clés pour marquer les blocs de code.
- Python est compatible avec tous les systèmes d'exploitation courants, y compris Windows, Mac OS X et les distributions Linux.
- Python est open source et sa licence est gérée par le python Software Fondation.
- Python est un langage de programmation polyvalent qui peut être utilisé pour une grande variété de tâches, telles que le développement web, le traitement de données, l'apprentissage automatique, la création de scripts, etc.
- Python est très populaire et est souvent classé parmi les langages de programmation les plus utilisés dans le monde, avec une grande communauté de développeurs.
- Python est un langage de programmation de haut niveau, ce qui signifie qu'il est facile à lire et à écrire, ce qui en fait un excellent choix pour les débutants en programmation.
- Python dispose d'une grande bibliothèque standard qui offre de nombreuses fonctionnalités et modules pour les développeurs.
- Python a une syntaxe claire et concise, qui rend le code facile à comprendre et à maintenir.

### 4.1.2 Pourquoi en utilisant python ?

La raison pour laquelle Python est souvent utilisée pour la classification des images est liée à son écosystème riche de bibliothèques et de Framework de traitement d'images, ainsi qu'à sa syntaxe simple et facile à comprendre qui permet une programmation rapide et efficace. Les bibliothèques les plus couramment utilisées pour la classification d'images en Python comprennent TensorFlow, PyTorch, Keras, scikit-image et Open CV. Ces bibliothèques fournissent des fonctionnalités telles que la manipulation et la transformation d'images,

l'entraînement de modèles de classification d'images et l'évaluation de la précision de ces modèles.

TensorFlow et PyTorch sont des Framework d'apprentissage automatique très populaire pour la classification d'images. Ils offrent des fonctionnalités avancées pour le traitement d'images, tels que la reconnaissance d'objet, la segmentation d'images et la classification d'images à l'aide de réseaux de neurones profonds.

Keras est une bibliothèque de haut niveau qui repose sur TensorFlow et permet de créer des modèles de classification d'images avec une syntaxe simple et intuitive. Elle est souvent utilisée pour des projets de classification d'images moins complexes.

Scikit-image et Open CV sont des bibliothèques de traitement d'images plus généralistes qui offrent des fonctionnalités pour la manipulation, la transformation et la reconnaissance de motifs dans les images. Elles sont souvent utilisées pour des projets de traitement d'images plus simples et moins axés sur l'apprentissage automatique.

En résumé, Python est couramment utilisé pour la classification d'images grâce à son écosystème riche de bibliothèques et de Framework de traitement d'images et à sa syntaxe facile à comprendre qui permet une programmation rapide et efficace.

Dans notre projet on utilise la version de Python 3.9.



**Figure 3.8:** La version de Python 3.9.

### 4.2 Google Colab

Google Colaboratory (ou Colab en abrégé) est un environnement de développement de projets en ligne basé sur les notebooks Jupyter. Il est proposé par Google et est entièrement gratuit.

Colab permet de travailler avec Python et les bibliothèques de données populaires tels que TensorFlow, Keras, PyTorch, Pandas et NumPy. Les utilisateurs peuvent écrire et exécuter du code dans un notebook, qui est un document interactif contenant des cellules de code et des cellules de texte formaté. Les notebooks sont sauvegardés sur Google Drive, ce qui permet de les partager facilement avec d'autres personnes.

Colab fournit également un accès gratuit à un GPU et à un TPU pour accélérer les tâches de traitement d'images et d'apprentissage automatique. Il est également possible de se connecter à des services de stockage cloud tels que Google Cloud Storage ou Amazon S3 pour accéder à des ensembles de données volumineux.

En plus de cela, Colab dispose d'une intégration avec les services Google tels que Google Sheets, Google Drive et Google Cloud Storage, ainsi que d'une intégration avec GitHub pour faciliter la collaboration.

Colab est très populaire pour l'enseignement, la recherche et le développement de projets personnels en raison de sa facilité d'utilisation, de sa flexibilité et de sa gratuité.



Figure 3.9: Logo de Google colab.

### 4.3 Open CV

Open CV (Open Source Computer Vision) est une bibliothèque open-source de traitement d'image et de vision par ordinateur. Elle fournit des outils et des algorithmes pour développer des applications de traitement d'images en temps réel, d'analyse vidéo et de reconnaissance d'objets, la segmentation d'image, l'estimation de mouvement, la vision stéréo, la calibration de caméra, entre autres fonctionnalités liées à la vision par ordinateur. Open CV est une bibliothèque très populaire dans l'industrie, la recherche et l'enseignement en vision par ordinateur, offrant une grande flexibilité et une compatibilité avec de nombreux langages de programmation tels que Python, Java et MATLAB, et prend en charge diverses plates formes, notamment Windows, Linux, MacOS, iOS et Android. Open CV développé en C++ et disponible sous licence BSD.

La bibliothèque est divisée en modules, chacun offrant des fonctionnalités spécifiques pour différents domaines de la vision par ordinateur. Les modules les plus couramment utilisés sont :

- **Core** : contient les fonctionnalités de base de la bibliothèque, telles que le traitement d'images, la conversion de formats d'image et les opérations arithmétiques et logiques.
- **Imgproc** : contient des fonctionnalités de traitement d'image telles que la détection de contours, la transformation de perspective, le flou et la dilation.
- **Vidéo** : contient des fonctionnalités d'analyse vidéo telles que la détection de mouvement, le suivi d'objets et l'estimation de flux optique.
- **Features2D** : contient des fonctionnalités de détection de points d'intérêt et de description de caractéristiques pour la reconnaissance d'objets.
- **Object Détection** : contient des fonctionnalités pour la détection d'objets et la reconnaissance faciale.
- **Machine Learning** : contient des outils pour l'apprentissage automatique, tels que les algorithmes de classification et de régression.

En constante évolution, la bibliothèque Open CV est mise à jour régulièrement pour inclure de nouvelles fonctionnalités et améliorer les performances et la stabilité. Elle est soutenue par une grande communauté active d'utilisateurs et de contributeurs.

La version utilisée dans ce projet est Open CV – 4.5.2.



Figure 3.10: Logo de la bibliothèque Open CV.

### 4.4 PyCharm

PyCharm est un environnement de développement intégré pour les développeurs Python, créé par JetBrains. Cet outil offre une large gamme de fonctionnalités pour aider les développeurs à écrire, déboguer et tester du code Python de manière efficace.

Il inclut notamment des fonctionnalités telles que l'auto complétion de code, la navigation de code, la refactoring de code, la gestion de versions, la prise en charge des tests unitaires, ainsi que la gestion de projets.

PyCharm offre de nombreuses fonctionnalités pour faciliter le développement de logiciels en Python, y compris :

- **Assistance au codage** : PyCharm offre une assistance au codage, notamment une complétion de code intelligente, des inspections de code et des suggestions de refactorisation.
- **Débogage** : PyCharm permet de déboguer facilement les programmes Python, avec des outils tels que des points d'arrêt, une console interactive et une évaluation d'expression.
- **Intégration de VCS** : PyCharm offre une intégration de versionnement de code source (VCS) avec Git, SVN et Mercurial.
- **Tests unitaires** : PyCharm prend en charge les tests unitaires en Python avec des outils pour exécuter des tests, générer des rapports et intégrer des Framework de test tels que pytest et unittest.
- **Gestionnaire de packages** : PyCharm permet de gérer les packages Python et les dépendances de projet avec des outils pour installer, désinstaller et mettre à jour les packages.
- **Interface utilisateur** : PyCharm offre une interface utilisateur personnalisable avec des thèmes, des icônes et des raccourcis claviers personnalisables.

En outre, PyCharm est très flexible et personnalisable grâce à son interface utilisateur qui permet aux développeurs de personnaliser l'IDE en fonction de leurs besoins. PyCharm est également doté de nombreux plugins et extensions pour étendre ses fonctionnalités de base.

En résumé, PyCharm est un outil indispensable pour tout développeur Python car il offre un environnement de développement professionnel, complet et confortable pour créer des projets Python complexes, tout en gagnant du temps et en évitant les erreurs. (Sebastian Raschka, 2019).



Figure 3.11: Logo de logiciel PyCharm.

## 5 Résultats et discussion

### 5.1 Base de données

Une base de données est un ensemble structuré et organisé de données qui sont stockées sur un ordinateur ou un système de stockage électronique. Elle permet de stocker, gérer et récupérer des informations de manière efficace et cohérente.

Les données sont généralement organisées en tables, qui sont composées de lignes et de colonnes. Chaque ligne représente un enregistrement individuel dans la base de données, et chaque colonne représente un attribut ou une caractéristique de cet enregistrement.

Les bases de données sont utilisées dans de nombreux domaines, tels que la finance, les ressources humaines, les soins de santé, l'éducation, la recherche scientifique et bien d'autres encore. Elles sont souvent utilisées pour stocker des informations importantes et sensibles, telles que des informations personnelles, des données financières ou des données scientifiques.

Notre système est de l'analyse des boîtes d'emballage collecte un ensemble d'images, composé de 5 images pour chaque boîte d'emballage. Ces 5 images sont constituées d'une image infrarouge et de 4 images couleur des 4 faces de la boîte. Cette étape de collecte des images constitue la phase d'acquisition des données.

Cependant, avant de pouvoir utiliser ces images pour entraîner notre système, nous devons effectuer une étape de prétraitement sur les images collectées. Cette étape consiste à transformer chaque image en un vecteur de caractéristiques, qui représente les propriétés clés de l'image. Le résultat de cette phase est une base de données contenant les vecteurs de caractéristiques correspondants à chaque image.

Ainsi, la base de données que nous allons utiliser pour entraîner et tester notre système est composée de ces vecteurs de caractéristiques, plutôt que des images originales. Cette approche de prétraitement nous permet de réduire considérablement la complexité du traitement des images et d'améliorer l'efficacité du système.

### 5.2 Schéma de base de données

Un schéma de base de données est un plan ou une représentation visuelle de la structure de la base de données. Il montre les tables, les relations entre les tables, les clés primaires et étrangères, les contraintes et les règles de validation.

Le schéma de la base de données est généralement créé avant la création de la base de données elle-même, et il est utilisé pour guider la conception de la base de données. Il peut être modifié au fil du temps pour refléter les changements dans les exigences de la base de données.

Le schéma de la base de données peut être représenté de différentes manières, telles que le modèle Entité-Relation (ER), le modèle Relationnel, le modèle Objet-Relationnel, etc. Le modèle ER est souvent utilisé pour la conception de base de données, car il est facile à comprendre et à visualiser.

En résumé, le schéma de la base de données est une représentation graphique de la structure et des relations des tables dans une base de données. Il est essentiel pour la conception et la maintenance efficaces de la base de données.

### 5.3 Configuration matériel et immatériel utilisé dans l'implémentation

#### 5.3.1 Environnement de matériel :

On a utilisé dans notre travail :

- Un PC portable HP Intel® Core™ i5.
- RAM de taille 16 Go.
- Disque SSD de taille 512 Go.
- Carte graphique Intel® UHD.
- Système d'exploitation Windows 10 professionnel (64 bits).

#### 5.3.2 Environnement immatériel

##### ➤ Installation de Python

Afin d'atteindre nos objectifs, il sera nécessaire de télécharger plusieurs dépendances (Python), pour lesquelles vous devriez déjà avoir effectué l'installation préalablement. De plus, nous aurons également besoin du logiciel Pycharm et de certaines bibliothèques.

##### ➤ Installation de Pycharm

PyCharm est un IDE (Integrated Développement Environnement) utilisé pour le développement de programmes en langage Python. Il offre des fonctionnalités avancées telles que l'analyse de code et un débogueur graphique. De plus, il permet la gestion des tests unitaires, l'intégration avec des outils de gestion de versions, et prend en charge le développement web avec Django. JetBrains, une entreprise tchèque, est responsable de son développement. PyCharm est un logiciel multi-plateforme qui fonctionne sur Windows, Mac OS X et Linux. Il existe une version professionnelle, qui est sous licence propriétaire, et une version communautaire, qui est diffusée sous licence Apache. (Voir Figure 3.12)



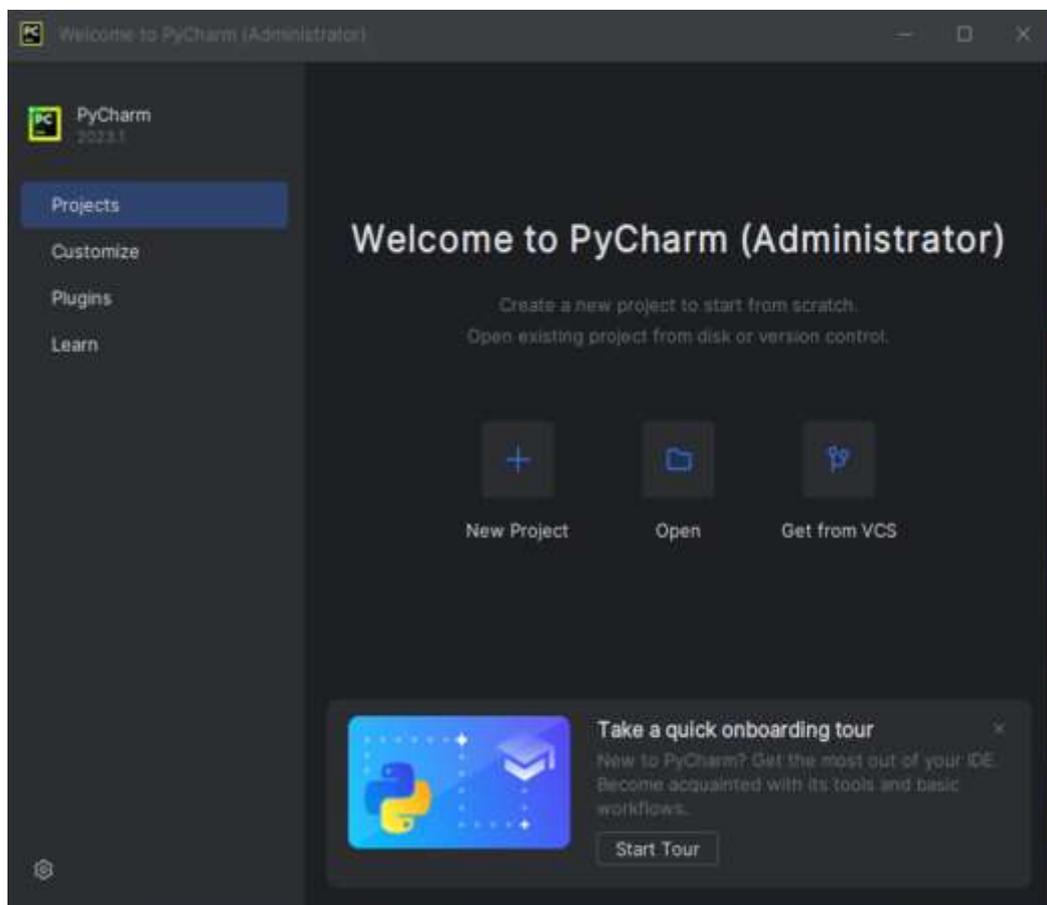
Figure 3.12: Installation PyCharm.

##### ➤ Création de l'environnement sous Pycharm

Pour créer un environnement de développement Python sous PyCharm, on a suivé les étapes suivantes :

- Ouvrir PyCharm et créer un nouveau projet en sélectionnant "File" > "New Project ». (Voir Figure 3.13).
- Sélectionner "Python" dans la liste des options de projet, puis choisir le répertoire où nous souhaitons stocker notre projet.
- Sélectionner la version de Python que nous souhaitons utiliser.
- Choisissez un nom pour notre environnement virtuel et sélectionnez l'emplacement où nous souhaitons l'enregistrer.
- Cliquer sur "Create" pour créer notre environnement de développement Python.

Une fois que nous avons créé notre environnement virtuel (Voir Figure 3.7), nous commençons installer des bibliothèques et des dépendances spécifiques à notre projet en utilisant le gestionnaire de packages Python (pip) à partir d'une console intégrée dans PyCharm.



**Figure 3.13:** Interface PyCharm. (Sebastian Raschka, 2019).

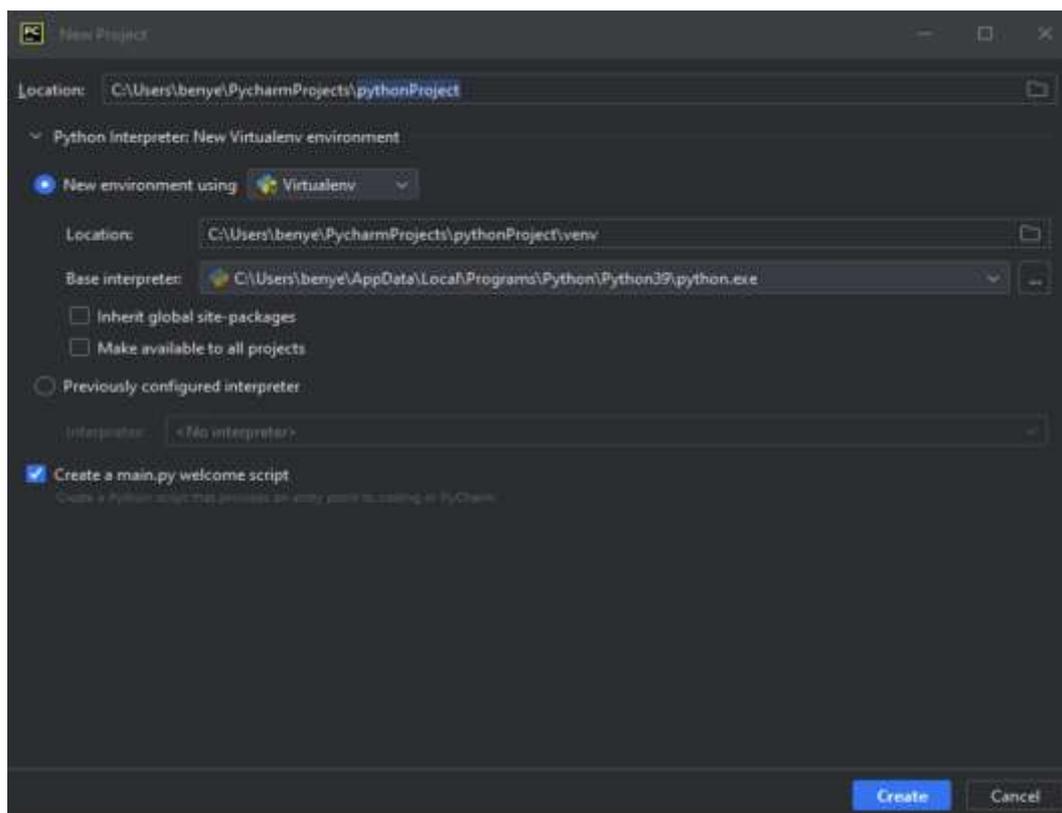


Figure 3.14: Création de l'environnement sous PyCharm.

### ➤ Installation de bibliothèques

Nous allons utiliser plusieurs bibliothèques qui seront téléchargées via le gestionnaire de packages pip. Pour récupérer toutes ces bibliothèques, vous pouvez saisir les commandes suivantes dans une console Windows (Cmd) :

```
E:\python\opencv>pip install numpy-1.13.1+mk1-cp36-cp36m-win_amd64.whl
Processing e:\python\opencv\numpy-1.13.1+mk1-cp36-cp36m-win_amd64.whl
Installing collected packages: numpy
Successfully installed numpy-1.13.1+mk1

E:\python\opencv>pip install opencv_python-3.3.1-cp36-cp36m-win_amd64.whl
Processing e:\python\opencv\opencv_python-3.3.1-cp36-cp36m-win_amd64.whl
Installing collected packages: opencv-python
Successfully installed opencv-python-3.3.1
```

Figure 3.15: Installation d'Open CV par la commande pip.

## 5.4 Les base des images

Nous avons un ensemble de données principal. Cet ensemble de données (dataset1) contient des images industrielles réparties en deux classes (Voir Tableau 3.2) :

- **La classe A** : qui correspond aux images sans défauts. (Voir Figure 3.16).
- **la classe B** : qui correspond aux images présentant des défauts. (Voir Figure 3.17)

Les images sont de format PNG.

### Chapitre3 : Conception et Implémentation

---

Le nombre d'images dans notre base de données est de 730 et sont de format PNG de taille 2,10 Mo et de dimension 224 \*224 pixels.



**Figure 3.16:** Echantillon des images de classe A.



**Figure 3.17:** Echantillon des images de classe B.

	A	B	C	D	E	F	G	H
1	classe	Identifiant	largeur	longue	hauteur	solidité	caractéristiques	cas
2	A	1	222 mm	136 mm	020mm	souple	100%alim	non defect
3	B	2	223 mm	137 mm	020mm	souple	100%alim	defect
4	B	3	224 mm	138 mm	020mm	souple	100%alim	defect
5	B	4	225 mm	139 mm	020mm	souple	100%alim	defect
6	B	5	226 mm	140 mm	020mm	souple	100%alim	defect
7	B	6	227 mm	141 mm	020mm	souple	100%alim	defect
8	B	7	228 mm	142 mm	020mm	souple	100%alim	defect
9	B	8	229 mm	143 mm	020mm	souple	100%alim	defect
10	B	9	230 mm	144 mm	020mm	souple	100%alim	defect
11	B	10	231 mm	145 mm	020mm	souple	100%alim	defect
12	B	11	232 mm	146 mm	020mm	souple	100%alim	defect
13	B	12	233 mm	147 mm	020mm	souple	100%alim	defect
14	B	13	234 mm	148 mm	020mm	souple	100%alim	defect
15	B	14	235 mm	149 mm	020mm	souple	100%alim	defect
16	B	15	236 mm	150 mm	020mm	souple	100%alim	defect
17	B	16	230mm	165mm	020mm	souple	100%alim	defect
18	B	17	231mm	166mm	020mm	souple	100%alim	defect
19	B	18	232mm	167mm	020mm	souple	100%alim	defect
20	B	19	233mm	168mm	020mm	souple	100%alim	defect
21	B	20	234mm	169mm	020mm	souple	100%alim	defect
22	A	21	235mm	170mm	020mm	souple	100%alim	non defect
23	A	22	236mm	170mm	020mm	souple	100%alim	non defect
24	A	23	237mm	170mm	020mm	souple	100%alim	non defect
25	A	24	238mm	170mm	020mm	souple	100%alim	non defect
26	A	25	239mm	170mm	020mm	souple	100%alim	non defect
27	A	26	240mm	170mm	020mm	souple	100%alim	non defect

Tableau 3.2: Partie de notre base de données en Excel.

## 6 Préparation de l'environnement

### 6.1 Installation des bibliothèques

L'installation de bibliothèques Python dans Google Colab est essentielle pour préparer l'environnement de travail. Les bibliothèques Python sont souvent utilisées dans les projets d'apprentissage automatique et d'analyse de données pour offrir des fonctionnalités supplémentaires telles que la manipulation de tableaux, la visualisation de données et l'entraînement de modèles d'apprentissage automatique. Ainsi, l'installation de ces bibliothèques est cruciale pour mener à bien des projets dans Google Colab.

Pour installer les bibliothèques requises dans Google Colab, nous utilisons la commande `pip Install`. Cette commande permet d'installer des bibliothèques Python à partir de PyPI (Python Package Index) ou d'autres sources. Nous pouvons également installer des bibliothèques à partir de fichiers de distribution (fichiers `.whl` ou `.tar.gz`) en utilisant la commande `pip Install` et en fournissant l'URL du fichier de distribution. Après avoir installé une bibliothèque, nous pouvons l'importer dans notre code Python en utilisant la commande `import`. (Voire la Figure 3.18)

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import tensorflow as tf
from tensorflow import keras
from keras.preprocessing.image import ImageDataGenerator
```

Figure 3.18: Partie de l'installation des bibliothèques dans Google colab.

- **import numpy as np** : importe la bibliothèque NumPy et la renomme en np pour une utilisation plus simple dans le code.
- **import matplotlib.pyplot as plt** : importe la bibliothèque Matplotlib et spécifiquement le module pyplot, qui fournit une variété de fonctions pour créer et manipuler des graphiques.
- **import matplotlib.image as mpimg** : importe le module mpimg de Matplotlib, qui peut être utilisé pour lire et afficher des fichiers d'image.
- **import tensorflow as tf** : importe la bibliothèque TensorFlow pour la création et l'entraînement de modèles de Deep Learning.
- **from tensorflow import keras** : importe le module Keras de TensorFlow, qui fournit une API haut niveau pour la création et l'entraînement des modèles de Deep Learning.
- **from keras.preprocessing.image import ImageDataGenerator** : importe la classe **ImageDataGenerator** de Keras, qui peut être utilisée pour générer des versions augmentées de données d'image pour l'entraînement des modèles de Deep Learning.

### 6.2 Importer les données

On va importer les données depuis Google drive car il permet la fluidité de transfert sans endommager le fichier.

On a utilisé cette ligne de commande :(Voir Figure 3.19)

```
import os
dataset_dir =  '/content/drive/MyDrive/dataset'

train_dir = os.path.join(dataset_dir, 'train')
validation_dir = os.path.join(dataset_dir, 'validation')

#repertoires avec nos photos(d'entrainement) de defect et de non defect
train_defect_dir = os.path.join(train_dir, 'defect')
train_non_defect_dir = os.path.join(train_dir, 'non defect')

#repertoires avec nos photos(de validation) de defect et de non defect
validation_defect_dir = os.path.join(validation_dir, 'defect')
validation_non_defect_dir = os.path.join(validation_dir, 'non defect')
```

Figure 3.19: Code dans Google colab pour importer notre base de données.

### 7 Visualisation des résultats

#### 7.1 Taux de reconnaissance

Le taux de reconnaissance, ou l'accuracy, peut être calculé à partir des prédictions de notre modèle et des vraies étiquettes de nos données de test à l'aide de la formule suivante :

- **Théoriquement :**

Accuracy = (nombre de prédictions correctes) / (nombre total de prédictions).

- **Mathématiquement :**

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Où :

- ✓ TP (True Positive) est le nombre d'observations qui ont été correctement identifiées comme positives.
- ✓ TN (True Negative) est le nombre d'observations qui ont été correctement identifiées comme négatives.
- ✓ FP (False Positive) est le nombre d'observations qui ont été identifiées à tort comme positives.
- ✓ FN (False Negative) est le nombre d'observations qui ont été identifiées à tort comme négatives.

Dans notre programme on a obtenu ce résultat :

L'entraînement du modèle a pris 10 épochès, ce qui signifie que le modèle a été entraîné sur l'ensemble de données d'entraînement 10 fois. Les résultats d'entraînement pour chaque épochè sont affichés, y compris la perte d'entraînement, la précision d'entraînement, la perte de validation et la précision de validation.

Le modèle a commencé avec une perte d'entraînement élevée et une précision d'entraînement faible du premier épochè, mais ces valeurs ont progressivement amélioré au fil du temps. La précision de validation a également augmenté au fil des épochès, indiquant que le modèle a appris à généraliser sur de nouveaux exemples.

Au total, le modèle a atteint une précision d'entraînement de 0,8329 et une précision de validation de 0,9373 après 10 épochès. Cela indique que le modèle a bien appris à distinguer les images de défaut des images sans défaut dans l'ensemble de données. (Voir Figure 3.21).

```
Epoch 1/10
23/23 [=====] - 4s 60ms/step - loss: 0.6909 - accuracy: 0.5493 - val_loss: 0.6018 - val_accuracy: 0.6654
Epoch 2/10
23/23 [=====] - 1s 43ms/step - loss: 0.5704 - accuracy: 0.6397 - val_loss: 0.4860 - val_accuracy: 0.6977
Epoch 3/10
23/23 [=====] - 1s 41ms/step - loss: 0.4145 - accuracy: 0.7301 - val_loss: 0.2683 - val_accuracy: 0.9163
Epoch 4/10
23/23 [=====] - 1s 42ms/step - loss: 0.2420 - accuracy: 0.8904 - val_loss: 0.1289 - val_accuracy: 0.9582
Epoch 5/10
23/23 [=====] - 1s 42ms/step - loss: 0.1845 - accuracy: 0.9219 - val_loss: 0.1218 - val_accuracy: 0.9582
Epoch 6/10
23/23 [=====] - 1s 32ms/step - loss: 0.1704 - accuracy: 0.9151 - val_loss: 0.1172 - val_accuracy: 0.9582
Epoch 7/10
23/23 [=====] - 1s 40ms/step - loss: 0.1560 - accuracy: 0.9164 - val_loss: 0.1516 - val_accuracy: 0.9373
Epoch 8/10
23/23 [=====] - 1s 32ms/step - loss: 0.1889 - accuracy: 0.9288 - val_loss: 0.2180 - val_accuracy: 0.8954
Epoch 9/10
23/23 [=====] - 1s 39ms/step - loss: 0.1811 - accuracy: 0.9219 - val_loss: 0.3018 - val_accuracy: 0.8954
Epoch 10/10
23/23 [=====] - 1s 39ms/step - loss: 0.3190 - accuracy: 0.8329 - val_loss: 0.1688 - val_accuracy: 0.9373
```

Figure 3.20 : Taux de reconnaissance obtenu dans notre modèle.

### 7.2 Evolution de taux de reconnaissance

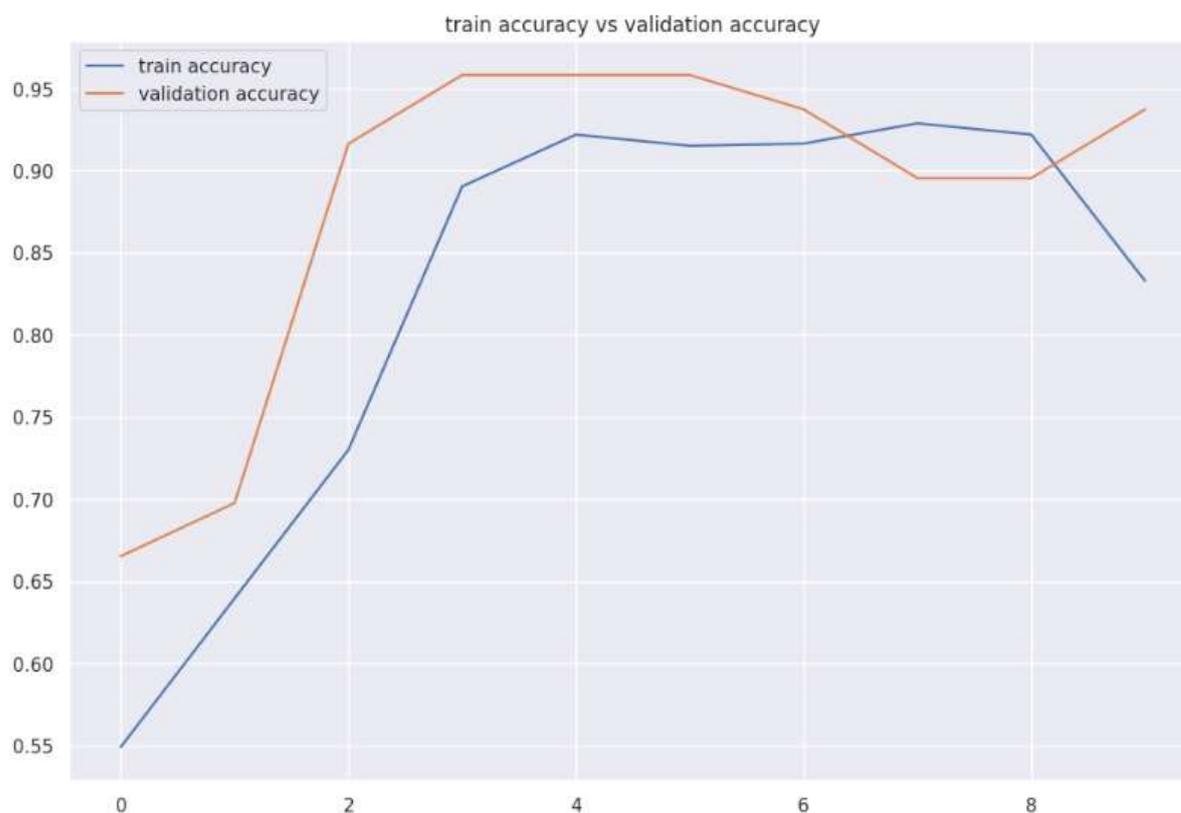


Figure 3.21: Les résultats de l'évolution de taux de reconnaissance.

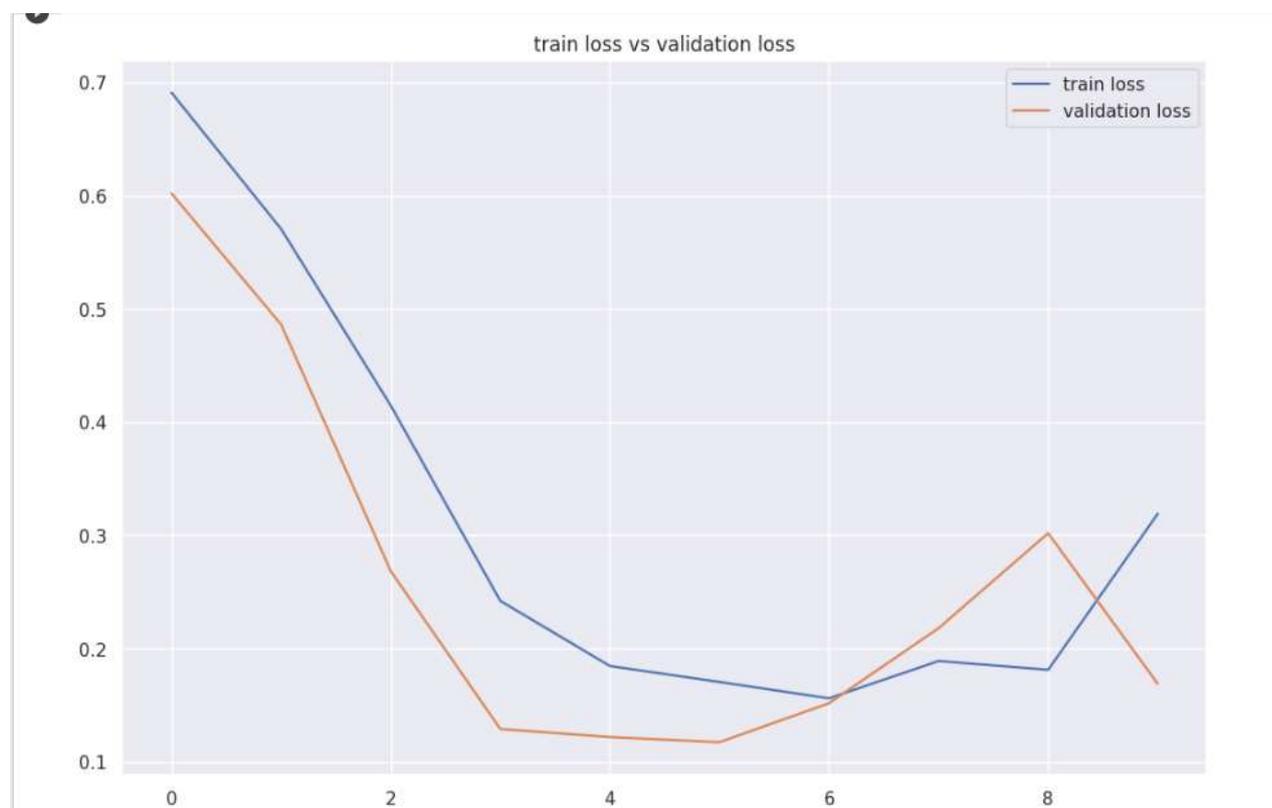
## Chapitre3 : Conception et Implémentation

La courbe tracée dans le graphique (Voir figure 3.21) représente l'évolution de l'exactitude (accuracy) du modèle d'apprentissage machine au fil des époques (epochs) d'entraînement.

La courbe bleue représente l'exactitude d'entraînement (train accuracy), c'est-à-dire la précision du modèle sur les données d'entraînement à chaque époque.

La courbe orange représente l'exactitude de validation (validation accuracy), c'est-à-dire la précision du modèle sur un ensemble de données de validation indépendant à chaque époque.

les courbes d'exactitude d'entraînement et de validation sont augmentent et ils se rapprochent l'une de l'autre au fil des époques, donc on conclut que le modèle apprend et généralise bien, et la convergence des courbes avec une exactitude élevée représente que la performance est solide de notre modèle.



**Figure 3.22:** Les résultats de l'évolution des pertes.

Ce graphe présente l'évolution de la perte (loss) du modèle d'apprentissage machine au fil des époques (epochs) d'entraînement (Voir Figure 3.22).

L'axe horizontal représente le nombre d'époques (epochs), indiqué par la plage de valeurs retournée par epochs. Chaque point sur l'axe horizontal correspond à une époque spécifique.

L'axe vertical représente la valeur de la perte (loss) du modèle. La perte d'entraînement (train loss) est représentée par la courbe bleue et la perte de validation (validation loss) est représentée par la courbe orange.

La courbe de perte d'entraînement (train loss) montre comment la perte du modèle évolue sur les données d'entraînement à chaque époque. La courbe diminue de manière constante, cela

indique une amélioration progressive de la performance du modèle sur les données d'entraînement.

La courbe de perte de validation (validation loss) montre comment la perte du modèle évolue sur un ensemble de données de validation indépendant à chaque époque. Elle permet d'évaluer la capacité du modèle à généraliser et à bien se comporter sur de nouvelles données. La courbe de perte de validation suit une tendance similaire à celle de la perte d'entraînement et diminue progressivement, cela suggère que le modèle généralise bien.

### **8 Conclusion**

Ce chapitre a présenté les résultats de manière visuelle à l'aide de figures et de tableaux, tout en expliquant et en comparant les résultats obtenus. L'accent a été mis sur l'explication détaillée de la méthodologie utilisée, ainsi que sur l'analyse approfondie des résultats. L'implémentation de l'étude a été réalisée en utilisant Python et les outils appropriés, ce qui a facilité le processus. L'ensemble du chapitre a fourni une vue complète de l'étude, y compris la méthodologie, les résultats et l'analyse, ainsi que les outils utilisés pour mener à bien l'étude.

## Conclusion générale

Dans le cadre de cette étude, notre objectif principal était de développer un système de classification et de détection de défauts en utilisant le Deep Learning. Nous avons commencé par fournir une explication détaillée des concepts fondamentaux du Deep Learning. Ensuite, nous avons mis en place un système d'apprentissage basé sur un réseau de neurones convolutifs (CNN) qui a démontré des performances satisfaisantes. Ce système a été entraîné sur un ensemble de données approprié et a été capable de classifier avec précision les différents types de défauts. Les résultats obtenus ont été prometteurs, ce qui souligne l'efficacité du Deep Learning dans le domaine de la détection de défauts.

Nous avons également décrit l'ensemble du processus de mise en œuvre de la méthode proposée en utilisant le langage de programmation Python et ses bibliothèques de Deep Learning telles que TensorFlow et Keras. Nous avons détaillé les différentes étapes du processus de classification, de la préparation des données à l'évaluation des résultats.

Nous avons choisi d'utiliser Google Colab pour implémenter notre réseau de neurones convolutif (CNN) en raison de la puissance de l'accélérateur GPU pour les calculs mathématiques. Le programme est composé de cellules de code dans un notebook Google Colab qui permettent de classifier les boîtes à partir des images, en utilisant le CNN.

Le CNN est un réseau de neurones multicouche qui commence par une couche de convolution et se termine par une couche entièrement connectée.

Dans le troisième chapitre ont montré que la méthode proposée a atteint une précision élevée de classification avec un taux de reconnaissance de plus de 90%. Les résultats obtenus montrent la faisabilité et l'efficacité de notre approche pour la classification des photos de boîtes d'emballage en utilisant le Deep Learning.

En conclusion, ce travail de mémoire a permis de montrer que l'utilisation du Deep Learning pour la classification des photos de boîtes d'emballage biodégradable peut être une solution performante et efficace. Les résultats obtenus ouvrent également la voie à de futures recherches dans ce domaine et à l'application de cette méthode dans divers secteurs industriels nécessitant une classification précise et rapide des images.

Cependant, pour intégrer cette solution sur le terrain et traiter un grand nombre de défaut en temps réel à l'aide de caméras industrielles, il est nécessaire de procéder à certaines modifications.

### Perspectives

Malgré les difficultés que nous avons eu durant la réalisation de notre travail, tel que le manque des bases de données, la difficulté de comprendre le fonctionnement des réseaux CNNs. Nous avons réussi à réaliser et implémenter notre modèle qui a pour rôle de classer correctement les images de boîtes d'emballages en fonction de leur classe et de leur présence ou non de défauts ,cela nous a permis de dégager plusieurs perspectives de travail :

**-Difficulté à généraliser aux nouvelles classes :** Les modèles de Deep Learning sont généralement conçus pour classer des catégories spécifiques d'images sur lesquelles ils ont été entraînés. Lorsqu'il est confronté à de nouvelles classes d'images pour lesquelles il n'a pas été entraîné, le modèle peut avoir du mal à effectuer des classifications précises.

**-Manque d'interprétabilité :** Les modèles de Deep Learning sont souvent considérés comme des boîtes noires, ce qui signifie qu'il peut être difficile de comprendre les raisonnements sous-jacents qui ont conduit à une classification particulière. Cela peut poser des problèmes dans les cas où une explication ou une justification est nécessaire.

-Il est important de prendre en compte ces limites lors de la conception et de l'utilisation d'un programme de classification des images basé sur le Deep Learning. Dans certains cas, d'autres approches ou méthodes complémentaires peuvent être nécessaires pour obtenir de meilleurs résultats.

### 9 Bibliographie

- AGGOUNE, L. (2018). Détection de défauts dans les procédés industriels par l'utilisation des méthodes d'identification paramétrique. *Thèse de doctorat*. Sétif.
- BAZI, S. (2016). Contribution à la Détection et au Diagnostic des Défauts dans un Système Machine à Induction-Convertisseur. *Mémoire de Master*. Batna .
- Brownlee, J. (2017). *"Deep Learning with Python"*. Machine Learning Mastery.
- Chen, M. Z. (2018). Saliency detection for defects in highly reflective surfaces using a deep convolutional neural network. *IEEE Transactions on Industrial Electronics*, 65(6), 5075-5083.
- Cheng, J. Y. (2020). Defect detection of glass surface based on deep learning. *IEEE Access*, 8, pp. 219465-219475.
- Cheng, J. Y. (2020). Defect detection of glass surface based on deep learning. *IEEE Access*, 8, 219465-219475.
- Chollet, F. (s.d.). *"Keras: Deep Learning for humans"*. Récupéré sur URL: <https://keras.io/>.
- Clayton, R. (2020, Retrieved Mars 08). *Qu'est-ce que le machine learning ?* (from Oracle Algeria) Récupéré sur <https://www.oracle.com/dz/artificial-intelligence/what-is-machine-learning.html>
- Crosby, P. B. (1979). *Quality is free: The art of making quality certain*. McGraw-Hill Education.
- EDDINE, T. H. (2020). La détection d'objet avec Open CV et deep Learning. *Mémoire de Master*. Biskra.
- GAËL. (2020, 05 Retrieved 04 ). *from Datakeen*. Récupéré sur Machine Learning: <https://datakeen.co/8-machine-learning-algorithms-explained-in-human-language/>
- Géron, A. (2017). *Hands-On Machine Learning with Scikit-learn and tensorflow*. USA: O'Reilly Media.
- Goodfellow, I. B. (2016). *Deep learning*. MIT Press.
- Grossfeld, B. (2020, 01 23). *Deep learning vs machine learning: a simple way to understand the difference*. . Récupéré sur Retrieved from zendesk blog: <https://www.zendesk.com/blog/machine-learning-and-deep-learning/>
- Grossfeld, B. (2020, 04 08). *Le programme du Master en intelligence artificielle*. Récupéré sur from Unidistance: : <https://unidistance.ch/intelligence-artificielle/master/modules/?gclid=Cj0KCQjwj7v0BRDOARIsAGh37io1ERDQK4WbBncrriozBfS6GtD7utp2VG>
- He, Y. Z. (2018). Defect detection and classification for steel surface based on deep learning. *In 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)* (pp. 530-534). IEEE.
- Ishikawa, K. (1985). *What is total quality control? (éd. Prentice-Hall, Englewood Cliffs, NJ, 1985)*. The Japanese Way.

- Issarane, H. (2020, 05 Retrieved 04 ). *Apprentissage Non Supervisé*. Récupéré sur from Le DataScientist: <https://le-datascientist.fr/apprentissage-non-supervise>
- KATHEPURI, S. (2020). Reconnaissance et classification des fruits à l'aide de techniques d'apprentissage en profondeur. *Thèse de doctorat* . Ireland.
- LeCun, Y. B. (2015). *Deep learning* (Vol. 521(7553)). Nature.
- Li, H. ( 2020, 03 Retrieved 20). *Which ML Algorithms to Use?* Récupéré sur from The Eponymous Pickle: <http://eponymouspickle.blogspot.com/2017/04/which-ml-algorithms-to-use.html>
- Patterson, J. &. (2017). *Deep Learning : A Practitioner's Approach*. Beijing: OReilly Media.
- Python Programming Language*. (2021). Récupéré sur Récupéré sur geeksforgeeks: <https://www.geeksforgeeks.org/python-programming-language/>
- Sebastian Raschka, V. M. (2019). *"Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2"*. Packt Publishing.
- Shao, Y. W. (2020). Deep learning based defect detection and classification in steel surfaces. *Measurement*, 167, 108270.
- Shi, C. Y. (2018). Weakly supervised defect detection via image and feature co-modulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1), 102-115.
- Shyam Narayan Jha, A. L. (2011). Measurement techniques and application of electrical properties for nondestructive quality evaluation of foods. *Journal of Food Science and Technology* .
- SLIMANE, M. A. (2021). Imagerie thermique pour le contrôle qualité. *Mémoire de Master*. Biskra.
- SLIMANE, M. A. (2021). Imagerie thermique pour le contrôle qualité. *Mémoire de Master*. Biskra.
- Sutton, R. S. (1988). Learning to predict by the method of temporal differences. *Machine Learning*. *TensorFlow website*,. (s.d.). Récupéré sur URL: <https://www.tensorflow.org/>.
- VINCENT, B. D. (2015). Détection de défauts d'un processus industriel basée sur l'analyse en composantes principales des données. Canada.
- Watkins, C. J. (1992). Q-learning. *Machine Learning*.
- whatisnumpy. (s.d.). *Récupéré sur numpy.org*. Récupéré sur (s.g): <https://numpy.org/doc/stable/user/whatisnumpy.html>
- Xu, C. X. ((2019)). A deep learning based defect detection and classification method for textile fabrics. *Measurement*, 139, 108-117.
- Zhang, H. W. (2019). Defect detection and classification in surface-mounted device based on deep learning. *IEEE Access*, 7, 95355-95364.