

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma –

Faculté des Mathématiques, d'Informatique et des Sciences de la matière

Département d'Informatique



Mémoire de Fin d'études Master

Filière : Informatique

Option : Science et technologie de l'information et de la communication

Thème

Poubelles Intelligentes pour la Reconnaissance des Bouteilles et la Récompense des Utilisateurs

Présenté par : M^{me} BENABID Manar

Membres du jury :

N	Nom et Prénom	Qualité
1	Dr FERKOUS Chokri	Encadrant
2	Dr FAREK Lazhar	Président
3	Dr BOURESSACE Hassina	Examineur
4	Dr BENAHCENE Nassiredine	Représentant Pole Pro

JUIN 2025

Remerciements

Je rends grâce à Allah, le Tout-Puissant, qui m'a accordé la force, la patience et la persévérance nécessaires pour mener à bien ce travail.

Je tiens à exprimer ma profonde gratitude à mon encadrant, Dr. Ferkous Chokri, pour son accompagnement précieux, ses conseils avisés et sa disponibilité tout au long de ce projet. Son soutien a été essentiel à l'aboutissement de ce mémoire.

Je remercie également tous les enseignants du département d'informatique, pour l'enseignement de qualité qu'ils nous ont dispensé durant ces années universitaires. Leurs savoirs et leur bienveillance ont largement contribué à ma formation.

Je ne saurais oublier de remercier mes collègues et amis, pour leur soutien constant, leur entraide et les moments d'échange fructueux tout au long de ce parcours.

Enfin, je souhaite adresser toute ma reconnaissance à mon époux bien-aimé, pour sa patience, son soutien indéfectible et son aide précieuse qui m'ont permis de poursuivre mes études dans les meilleures conditions. Sans lui, de nombreuses difficultés auraient été insurmontables. Je remercie également mes parents, pour leur amour inconditionnel, leurs sacrifices et leurs prières incessantes. Ils sont la véritable source de motivation et d'inspiration qui me pousse à réussir.

DEDICACE

Je dédie ce travail

À ma petite famille, mon cher époux et ma fille magnifique,

À ma grande famille, à ma mère, mon père

*À ma frère et sœur, ainsi qu'à mes neveux bien-aimés, Abd
elmouhaymen et Mohamed Hassan,*

Résumé

Face à l'augmentation massive des déchets plastiques et à leurs effets néfastes sur l'environnement, il devient essentiel de développer des solutions innovantes et efficaces pour y faire face. Dans ce cadre, ce projet s'appuie sur les technologies d'intelligence artificielle pour concevoir un système intelligent et portable capable de reconnaître automatiquement les bouteilles en plastique et de les trier. Le système repose sur une unité Raspberry Pi intégrée à une caméra numérique et à plusieurs composants électroniques tels qu'un moteur électrique, une imprimante thermique, des capteurs et des voyants lumineux. Une image est capturée pour chaque bouteille insérée, puis analysée par un modèle d'intelligence artificielle entraîné sur le réseau MobileNet, un modèle léger et performant conçu pour les appareils à ressources limitées. Une base de données volumineuse et équilibrée de plus de 70 000 images a été constituée et utilisée pour entraîner et optimiser le modèle. Après son intégration dans le système embarqué, les tests ont révélé une grande précision et une rapidité de traitement, faisant de cette solution une étape concrète vers l'amélioration du tri automatisé dans divers contextes tels que les espaces publics ou les établissements, tout en contribuant à sensibiliser les utilisateurs à l'importance du recyclage et à la protection de l'environnement.

Mots-clés : intelligence artificielle, réseaux de neurones, apprentissage profond, bouteilles en plastique, tri de bouteilles en plastique.

ملخص

في ظل التزايد الكبير للنفايات البلاستيكية وتأثيرها السلبي المتزايد على البيئة، أصبح من الضروري إيجاد حلول مبتكرة وفعالة لمواجهة هذا التحدي. في هذا السياق، يعتمد هذا المشروع على تقنيات الذكاء الاصطناعي لتصميم Raspberry Pi نظام ذكي ومحمول قادر على التعرف التلقائي على القنينات البلاستيكية وفرزها. يقوم النظام على وحدة مدمجة مع كاميرا رقمية ومجموعة من المكونات الإلكترونية مثل محرك كهربائي، طابعة حرارية، حساسات، وأضواء تنبيه، حيث يتم التقاط صورة لكل قارورة تدخل إلى النظام، ليتم تحليلها باستخدام نموذج ذكاء اصطناعي مدرب على شبكة ، وهي شبكة خفيفة وفعالة مخصصة للعمل على الأجهزة محدودة الموارد. تم تكوين قاعدة بيانات ضخمة MobileNet ومتوازنة تحتوي على أكثر من 70 ألف صورة، موزعة على فئات مختلفة، واستخدمت لتدريب وتحسين أداء النموذج. بعد دمج النموذج ضمن النظام، أظهرت الاختبارات دقة عالية وسرعة استجابة معتبرة، ما يجعل هذا الحل خطوة عملية نحو دعم عمليات الفرز الآلي في مختلف الأماكن مثل الفضاءات العمومية والمؤسسات، مع المساهمة في رفع مستوى الوعي... البيئي لدى المستخدمين وتحفيزهم على المساهمة في جهود إعادة التدوير.

الكلمات المفتاحية: الذكاء الاصطناعي، الشبكات العصبية، التعلم العميق، القارورات البلاستيكية، فرز القنينات البلاستيكية.

Abstract

Given the sharp increase in plastic waste and its negative impact on the environment, it has become essential to develop innovative and effective solutions to address this issue. In this context, this project leverages artificial intelligence technologies to design a smart and portable system capable of automatically detecting and sorting plastic bottles. The system is built around a Raspberry Pi unit integrated with a digital camera and various electronic components such as an electric motor, thermal printer, sensors, and indicator lights. For each bottle inserted, the system captures an image and analyzes it using an AI model trained on the MobileNet neural network — a lightweight and efficient model designed for low-resource embedded devices. A large and balanced dataset of over 70,000 images was collected and used to train and fine-tune the model. After integrating the model into the embedded system, experimental results demonstrated high accuracy and fast performance, making this solution a practical step toward enhancing automated sorting operations in various locations such as public areas or institutions, while also promoting environmental awareness and encouraging users to participate in recycling efforts.

Keywords: artificial intelligence, neural networks, deep learning, plastic bottles, plastic bottle sorting.

Table des matières

INTRODUCTION GENERALE.....	8
CHAP I : LES RESEAUX DE NEURONES ARTIFICIELS ET A CONVOLUTION	10
1.1 INTRODUCTION	11
1.2 NEURONE BIOLOGIQUE	11
1.3 NEURONE ARTIFICIEL.....	12
1.4 ARCHITECTURE DES RESEAUX DE NEURONES	12
1.4.1 Les fonctions d'activations	13
1.4.1.1 La fonction d'activation d'étape binaire.....	14
1.4.1.2 La fonction d'activation linéaire	14
1.4.1.3 La fonction d'activation sigmoïde	15
1.4.1.4 La fonction d'activation Tanh	15
1.4.1.5 La fonction d'activation ReLu	16
1.4.1.6 La fonction d'activation Leaky ReLU	17
1.4.1.7 La fonction d'activation PReLU : (Paramétric ReLU)	17
1.4.1.8 La fonction d'activation ELU.....	18
1.4.1.9 La fonction d'activation softmax	18
1.4.2 Les fonctions de perte	18
1.4.2.1 Erreur quadratique moyenne (MSE).....	19
1.4.2.2 Erreur absolue moyenne (MAE).....	19
1.4.2.3 Pert huber	20
1.4.3 Les algorithmes d'optimisation	20
1.4.3.1 La descente de gradient (GD)	21
1.4.3.2 La descente du gradient par batch	21
1.4.3.3 Descente du gradient stochastique	22
1.4.3.4 Descente du gradient avec mini-batch.....	22
1.4.3.5 Momentum	23
1.4.3.6 Gradient accéléré de Nesterov	23
1.4.3.7 Adagrad.....	24
1.4.3.8 RMSprop	24
1.4.3.9 Adadelta.....	25
1.4.3.10 Adam	25
1.4.3.11 Nadam.....	26
1.5 APPRENTISSAGE PROFOND (DEEP LEARNING OU DL).....	26
1.5.1 Les réseaux de neurones convolutifs.....	26
1.5.1.1 Les couches d'un CNN.....	27
1.5.1.2 La Couche de convolution	27
1.5.1.3 Couche de pooling.....	29
1.5.1.4 La couche fully connected (FC).....	29
1.6 QUELQUES ARCHITECTURE BASEES CNN	29
1.6.1 VGG	29
1.6.2 GoogLeNet.....	30
1.6.3 ResNet	31
1.6.4 EfficientNet.....	32
1.7 APPRENTISSAGE PAR TRANSFERT	33
1.8 CONCLUSION	34
CHAP II : LA GESTION DES DECHETS PLASTIQUES	35
1.9 INTRODUCTION	36
1.10 LA POLLUTION	36
1.10.1 Statistiques mondiales et locales (Algérie) sur les déchets plastiques	37
1.10.2 Les effets négatifs de l'élimination anarchique des bouteilles en plastique.....	37
1.11 INTRODUCTION SUR LE RECYCLAGE	38
1.12 DEFINITION DE LE RECYCLAGE	39
1.12.1 Les avantages du recyclage	39
1.12.2 Les difficultés ou défis du recyclage dans les pays en développement	39
1.13 LES METHODES TRADITIONNELLES DE COLLECTE ET DE TRI DU PLASTIQUE	40

1.13.1	La collecte manuelle :	40
1.13.2	Les centres de tri :	40
1.13.3	Les machines de tri industrielles :	41
1.14	PRESENTATION DU SYSTEME PROPOSE ET SON ROLE.....	41
1.14.1	Description du système	41
1.14.2	Avantages du système proposé par rapport aux méthodes traditionnelles.....	41
1.15	TRAVAUX ANTERIEURS	42
1.16	CONCLUSION	44
CHAP III :	CONCEPTION ET REALISATION.....	46
1.17	INTRODUCTION	47
1.18	PRESENTATION DES COMPOSANTS MATERIELS DU SYSTEME	47
1.18.1	Raspberry pi 4	47
1.18.2	Principales entrées et sorties – Raspberry Pi 4	48
1.18.3	Arduino MEGA.....	49
1.18.4	Moteur Nema 17 :	49
1.18.5	Microstep Driver.....	50
1.18.6	Caméra 5MP OV5647 avec Vision Nocturne	50
1.18.7	Imprimante thermique JP QR204 (58mm, intégrée)	51
1.18.8	Alimentation Double Canal 5V/6A – 24V/5A	52
1.18.9	Module Capteur D'obstacle Infrarouge KY-032	53
1.18.10	Breadboard 830 points.....	54
1.18.11	Flashing LED 5mm, Red, Green	54
1.18.12	Push Button Switch.....	54
1.18.13	Solderless Jumper Breadboard Wires M/M F/F M/F.....	55
1.19	ENVIRONNEMENT DE DEVELOPPEMENT LOGICIEL ET COMMUNICATION DANS LE SYSTEME ARDUINO	55
1.19.1	Arduino IDE	55
1.19.2	Les bibliothèques utilisées dans Arduino IDE.....	55
1.19.2.1	Bibliothèque Servo	56
1.19.2.2	Bibliothèque Adafruit_Nokia5110	56
1.19.2.3	Bibliothèque SoftwareSerial	56
1.19.2.4	Bibliothèque DHT	56
1.19.2.5	Communication série (Serial Communication) :	56
1.19.3	Langage de Programmation Python	57
1.19.4	Définitions détaillées des bibliothèques Python utilisées	57
1.19.4.1	Os	57
1.19.4.2	Shutil.....	58
1.19.4.3	Numpy	58
1.19.4.4	matplotlib.pyplot.....	58
1.19.4.5	Tensorflow	58
1.19.4.6	ImageDataGenerator	58
1.19.4.7	train_test_split.....	59
1.20	INTEGRATION DU SYSTEME ET CONNEXION DES COMPOSANTS	59
1.20.1	Principe de fonctionnement du système embarqué	60
1.21	ASSEMBLAGE DU SYSTEME ET INTEGRATION DES COMPOSANTS.....	61
1.21.1	Raspberry et le capteur d'obstacle	61
1.21.2	Raspberry et le Camera	62
1.21.3	Arduino et le driver	62
1.21.4	Driver et le moteur	62
1.21.5	Arduino et l'imprimante.....	63
1.21.6	L'alimentation et L'imprimante et le driver	63
1.21.7	Arduino et LED.....	63
1.22	METHODOLOGIE.....	63
1.22.1	Base de données et préparation	63
1.22.2	Choix du modèle et configuration de l'architecture	64
1.23	RESULTAT ET DISCUSSION	65
1.24	ANALYSE DES PERFORMANCES TYPQUES DU SYSTEME	66
1.25	CONCLUSION	67
CONCLUSION GENERALE		68
REFERENCES BIBLIOGRAPHIQUES		69

ANNEXES.....	74
Annexe 1 : Soumission du projet	74
Annexe 2 : Aspects innovants du projet	75
Annexe 3 : Analyse stratégique du marché.....	75
Annexe 4 : Plan de production et d'organisation.....	76
Annexe 5 : Plan financier	77
Annexe 6 : Prototype	81
Business model canevas	82

Liste des figures

Figure 1-1:Neurone biologique [4].....	11
Figure 1-2 : Représentation schématique d'un neurone artificiel [4]	12
Figure 1-3:Réseau neuronal avec 1 couches cachées [4]	13
Figure 1-4:Graph de la fonction d'étape binaire[7]	14
Figure 1-5: Graph de la fonction linéaire [8]	15
Figure 1-6: Graph de la fonction d'activation sigmoïde.....	15
Figure 1-7: Graph de la fonction d'activation tanh.....	16
Figure 1-8:Graph de la fonction d'activation ReLu	16
Figure 1-9:Graph de la fonction d'activation leaky ReLu	17
Figure 1-10:Graph de la fonction d'activation paramétrique ReLu	17
Figure 1-11:Graph de la fonction d'activation ELU	18
Figure 1-12:Une architecture typique d'un CNN[1]	27
Figure 1-13: Exemple d'opération de convolution avec un noyau de taille 3×3 sans padding.....	28
Figure 1-14 :Exemple d'opération de convolution avec padding, un tenseur d'entrée de taille 5×5, noyau de 3×3 et un stride de 1.....	28
Figure 1-15: Types d'opération de pooling	29
Figure 1-16:L'architecture VGG16.....	30
Figure 1-17: Architecture GoogLeNet.....	31
Figure 1-18:L'architecture ResNet.	32
Figure 1-19:Apprentissage par transfert	34
Figure 2-1 Architecture de mobile net utilisée [34]	43
Figure 2-2Network architecture of channel attention block (a) and spatial attention bloc.....	44
Figure 3-1: image de Raspberry pi 4.....	48
Figure 3-2: image de l'Arduino MEGA.....	49
Figure 3-3: image de Nema 17	49
Figure 3-4: Image de microster driver	50
Figure 3-5 : Camera 5MP OV5647	51
Figure 3-6 : Imprimante thermique JP QR204.....	52
Figure 3-7 : Alimentation Double Canal (5V/6A et 24V/5A)	53
Figure 3-8 : IR Obstacle Avoidance Sensor Module.....	53
Figure 3-9: Breadboard 830 points.....	54
Figure 3-10 : LED clignotante.....	54
Figure 3-11 : Bouton-poussoir.....	54
Figure 3-12: Les fils de connexion	55
Figure 3-13: L'architecture générale pour le développement du projet	59
Figure 3-14 : Train and Validation Accuracy and Loss.....	66

Introduction générale

Face aux défis environnementaux croissants auxquels notre monde est confronté aujourd'hui, la pollution par les déchets plastiques est devenue l'un des problèmes majeurs nécessitant des solutions efficaces et durables. Parmi ces déchets, les bouteilles en plastique sont parmi les plus répandues, consommées quotidiennement en quantités massives et jetées dans la nature ou dans des décharges sans tri rigoureux. Dans ce contexte, le recours aux technologies intelligentes, notamment l'intelligence artificielle, est devenue nécessaire pour améliorer les processus de tri et de recyclage.

Ce projet s'inscrit dans ce cadre et vise à développer un système intelligent basé sur des techniques d'apprentissage profond, notamment les réseaux de neurones convolutifs (CNN), pour identifier automatiquement les bouteilles acceptables et rejetées afin de faciliter le processus de tri. Ce travail se concentre sur la conception d'un modèle performant, son entraînement sur une base de données riche et diversifiée, puis son intégration dans un système embarqué capable de fonctionner dans des conditions quasi réalistes.

Ce travail se décompose en plusieurs étapes, incluant le traitement et la préparation des données, l'entraînement et les tests du modèle, ainsi que sa connexion à des composants physiques tels qu'une caméra, un moteur, un écran, etc. À travers ce projet, nous souhaitons contribuer au développement de solutions techniques qui soutiennent les efforts environnementaux et contribuent à la construction d'un système économique circulaire qui réutilise les ressources de manière intelligente et efficace.

Ce mémoire est structuré en trois chapitres principaux, suivis d'annexes, afin d'offrir une approche complète allant de la base théorique à la mise en œuvre pratique et à la valorisation du projet sous forme de startup.

Le premier chapitre présente les fondements théoriques liés aux réseaux neuronaux convolutifs (CNN), qui constituent le cœur du système de reconnaissance utilisé. Ce chapitre retrace l'évolution historique de ces réseaux, les défis rencontrés au fil du temps, ainsi que leurs applications actuelles, en mettant l'accent sur leur rôle dans le traitement d'image.

Le deuxième chapitre aborde la problématique environnementale des déchets plastiques. Il met en lumière les enjeux liés à leur accumulation, notamment en Algérie, et souligne l'importance du recyclage. Ce chapitre explore également les méthodes traditionnelles de gestion des déchets, les limites des infrastructures actuelles, et les solutions innovantes reposant

sur l'intelligence artificielle, tout en présentant des travaux de recherche antérieurs dans ce domaine.

Le troisième chapitre est consacré à la conception et à la réalisation du système de tri intelligent. Il décrit en détail les composants matériels et logiciels utilisés, les schémas de câblage, les interactions entre modules, ainsi que le fonctionnement global du prototype développé. L'objectif est de démontrer la faisabilité technique du système, sa robustesse et son potentiel d'intégration réelle.

Enfin, des annexes sont ajoutées pour présenter les aspects liés au développement du projet sous forme de startup technologique. Elles incluent notamment :

- Une étude de faisabilité économique et technique,
- Un aperçu du modèle économique proposé,
- Les objectifs à court et moyen terme,
- Les partenaires potentiels et la stratégie de déploiement,
- Ainsi que des éléments de communication et d'identité visuelle.

Cette structure progressive permet non seulement de comprendre les fondements scientifiques et techniques du projet, mais aussi de valoriser son impact environnemental et son potentiel entrepreneurial.

Chapitre

1

Les réseaux de neurones artificiels et à convolution

1.1 Introduction

Les réseaux neuronaux convolutifs (Convolutional Neural Networks, ou CNNs) sont aujourd'hui considérés comme le poulx de l'intelligence artificielle, inspirés par le fonctionnement des réseaux neuronaux biologiques dans le cerveau humain [1]. Ces réseaux possèdent une histoire riche, marquée par des périodes de développement alternant avec des phases de stagnation, en raison des nombreux défis auxquels ils ont été confrontés depuis leur émergence [1].

Parmi ces défis, on compte des restrictions et obstacles de nature conceptuelle, technique, matérielle et éthique. Cependant, l'introduction des réseaux multicouches, notamment avec la méthode de rétropropagation (backpropagation), a permis de surmonter une partie de ces limitations. Malgré ces avancées, les contraintes liées aux appareils à faible puissance et au coût élevé de la technologie ont persisté jusqu'à l'avènement de la révolution industrielle numérique [2].

C'est à ce moment que les réseaux neuronaux convolutifs ont commencé à offrir des résultats prometteurs, particulièrement dans les domaines de l'extraction de caractéristiques et de la reconnaissance d'objets. Aujourd'hui, ils sont utilisés dans de nombreux secteurs, notamment la reconnaissance d'images et de sons, le traitement automatique du langage naturel (NLP), la prédiction et l'analyse de données, les jeux, et bien d'autres encore [3].

1.2 Neurone biologique

Le système nerveux humain contient des cellules, appelées neurones. Les neurones sont connectés les uns aux autres à l'aide d'axons et de dendrites, et les régions de connexion entre les axones et les dendrites sont appelées synapses[4].

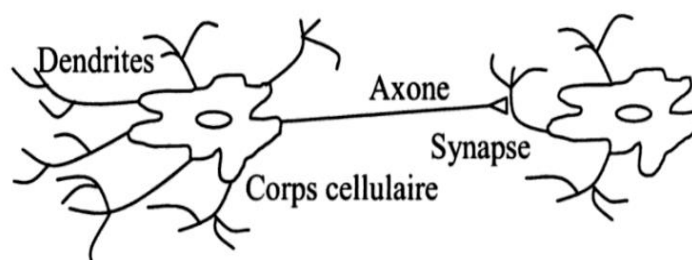


Figure 0-1: Neurone biologique [4]

Le modèle de neurone comprend (*Figure 0-1*) :

- **Des dendrites** : qui sont les parties principales de réception dans le neurone, captant les signaux qui lui parviennent
- **Un soma (corps cellulaire)** : corps de neurone compare la somme des signaux des dendrites à une certaine valeur seuil.
- **Un axone** : responsable a la transmission des signaux émis pare le corps cellulaire aux autres neurones.
- **Synapse** : qui correspond à la communication avec les autres parties via les axones et les dendrites.

1.3 Neurone Artificiel

Les neurones artificiels représentent le nœud principal, l'unité centrale de traitement ou l'unité informatique d'un réseau neuronal artificiel (réseau neuronal). Chaque neurone reçoit un ensemble d'entrées ($x_0, x_1, \dots, x_n$), chacune associée à un poids spécifique ($w_0, w_1, \dots, w_n$). Ces entrées sont traitées par une fonction de combinaison (F), souvent une somme pondérée ($\sum w_i x_i$), suivie d'une fonction d'activation pour produire une sortie (z) [4]. La figure ci-dessous montre un diagramme schématique du modèle mathématique d'un neurone artificiel

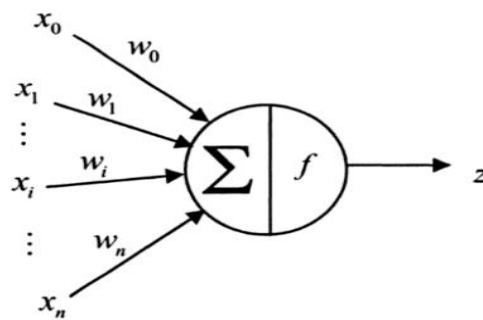


Figure 0-2 : Représentation schématique d'un neurone artificiel [4]

1.4 Architecture des réseaux de neurones

Les réseaux de neurones artificiels sont constitués de trois couches de base : la couche d'entrée, la couche cachée et la couche de sortie, chacune contenant un certain nombre de neurones[4], comme le montre la figure ci-dessous:

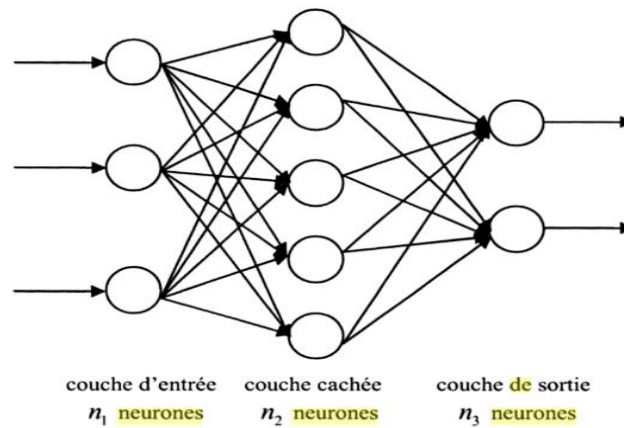


Figure 0-3: Réseau neuronal avec 1 couches cachées [4]

La couche d'entrée est la première couche constituée de n neurones élémentaires, qui reçoit les données d'entrée et transmet ses sorties vers l'extérieur. Cette couche ne réalise aucun calcul et ne contient pas de fonction d'activation.

La couche cachée est située entre la couche d'entrée et la couche de sortie. Elle peut inclure une ou plusieurs couches interconnectées de neurones. Ces couches sont responsables de la transformation et de l'extraction des caractéristiques pertinentes grâce à des calculs pondérés et des fonctions d'activation non linéaires. Les résultats intermédiaires produits par les couches cachées sont transmis aux couches suivantes [5].

La couche de sortie est la dernière couche du réseau et génère les prédictions finales ou les décisions en fonction des calculs effectués dans les couches précédentes. Chaque neurone dans cette couche représente une sortie spécifique, par exemple une classe dans une tâche de classification ou une valeur dans une tâche de régression.

1.4.1 Les fonctions d'activations

Les fonctions d'activation jouent un rôle très important dans les réseaux de neurones artificiels, car elles les rendent non linéaires et dynamiques. Sans ces fonctions, les réseaux de neurones ne seraient que de simples fonctions linéaires, réduites à des polynômes de premier degré, faciles à résoudre mais très limitées en complexité. Dans ce cas, ils ne pourraient ni apprendre ni reconnaître des relations complexes à partir des données [6].

Essentiellement, une fonction d'activation est une fonction mathématique qui transforme les sorties des neurones en fonction des entrées, puis transmet ces résultats à la couche suivante. C'est ce processus qui permet au réseau d'extraire et de modéliser des relations complexes pour

résoudre des tâches comme la classification, la régression et bien d'autres applications en apprentissage automatique. (2)

1.4.1.1 La fonction d'activation d'étape binaire

La fonction d'étape binaire est le plus simple, Elle est utilisée si la sortie de problème est binaire c'est-à-dire 0 ou 1. Cette fonction est peu utilisée dans les réseaux modernes en raison de ses limitations. Mathématiquement la fonction d'étape binaire peut être définie comme suit :

$$f(x)=1 \text{ si } x \geq 0 \text{ sinon } f(x)=0 \quad \text{Équation 0-1}$$

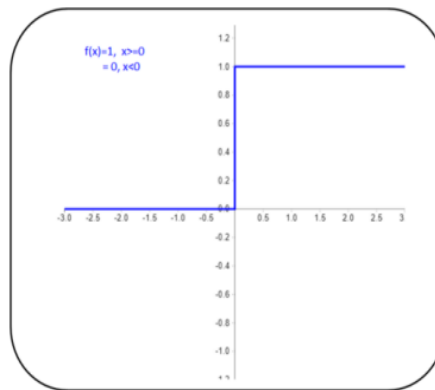


Figure 0-4: Graph de la fonction d'étape binaire[7]

1.4.1.2 La fonction d'activation linéaire

La fonction d'activation linéaire peut être définie par l'équation suivante :

$$f(x) = ax \quad \text{Équation 0-2}$$

- où a est une constante.

Cette fonction est principalement utilisée dans la couche de sortie des réseaux de neurones pour des tâches de régression, où la sortie est une valeur continue (par exemple, la prédiction du prix d'une maison). Dans ce cas, la valeur d'activation est identique à la valeur de pré-activation.

Cependant, cette fonction est rarement utilisée dans les couches cachées des réseaux modernes en raison de son manque de non-linéarité, ce qui limite la capacité du réseau à apprendre des relations complexes dans les données. Les fonctions linéaires sont idéales pour des tâches simples.

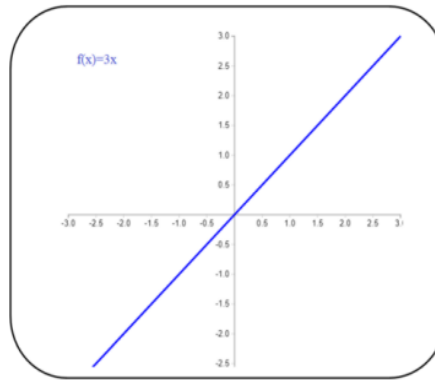


Figure 0-5: Graph de la fonction linéaire [8]

1.4.1.3 La fonction d'activation sigmoïde

La fonction sigmoïde est couramment utilisée dans les tâches de classification binaire, où la sortie est comprise entre 0 et 1. Elle est toujours positive et n'est pas centrée à l'origine, La valeur retournée par la sigmoïde représente une probabilité. Par exemple, une sortie de 0.8 peut être interprétée comme une probabilité de 80 % que la classe prédite soit 1. Cette fonction a longtemps été l'une des plus utilisées grâce à sa non-linéarité, mais elle présente des inconvénients, notamment en réseaux de neurones profonds, où elle souffre de problème de la longueur des calcule, elle souffre aussi également du problème du *vanishing gradient* (dérivée proche de zéro) lorsque les entrées sont très grandes ou très petites. Cela peut ralentir l'apprentissage, surtout dans les réseaux profonds, où les gradients peuvent devenir trop petits pour être efficaces lors de la mise à jour des poids [6]. Elle est définie par la formule :

$$f(x) = \frac{1}{e^{-x}}$$

Équation 0-3

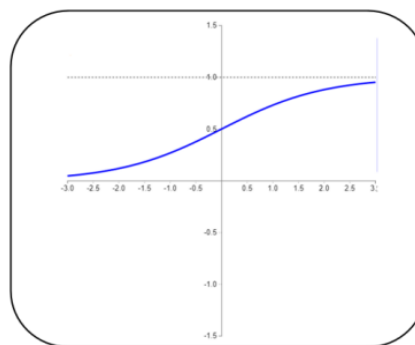


Figure 0-6: Graph de la fonction d'activation sigmoïde [6]

1.4.1.4 La fonction d'activation Tanh

La fonction Tanh (tangente hyperbolique) est une fonction d'activation non linéaire. Contrairement à la fonction sigmoïde, elle est symétrique par rapport à l'origine, ce qui signifie

que ses valeurs sont comprises entre -1 et 1. Cette symétrie permet aux sorties d'être centrées autour de zéro, ce qui peut faciliter l'apprentissage par rapport à la sigmoïde, car les activations peuvent être positives ou négatives mais elle souffre aussi comme le sigmoïde du problème du *vanishing gradient* (dérivée proche de zéro). Elle est définie par la formule :

$$f(x) = e^x - \frac{e^{-x}}{e^x} + e^x \quad \text{Équation 0-4}$$

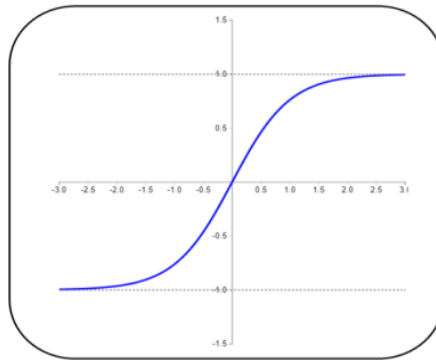


Figure 0-7: Graph de la fonction d'activation tanh [6]

1.4.1.5 La fonction d'activation ReLu

La fonction ReLU (Rectified Linear Unit) est la fonction la plus utilisée dans les réseaux de neurones convolutifs en raison de sa simplicité et de sa non-linéarité. Contrairement à la sigmoïde ou à tanh, la ReLU ne souffre pas du problème de vanishing gradient pour les entrées positives. Les dérivées ne deviennent pas proches de zéro pour des valeurs très petit ou très grand, ce qui permet un apprentissage plus rapide et plus efficace. D'autre part, la ReLU souffre du problème des neurones morts, car la sortie est zéro pour toutes les valeurs négatives [9]. Elle peut être écrite mathématiquement comme :

$$f(x) = \max(0, x) \quad \text{Équation 0-5}$$

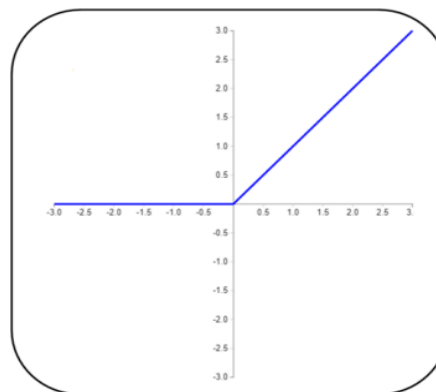


Figure 0-8: Graph de la fonction d'activation ReLu [6]

1.4.1.6 La fonction d'activation Leaky ReLU

Leaky ReLU (Rectified Linear Unit) est une variante de la fonction ReLU, conçue pour résoudre un problème connu sous le nom de "neurones morts" (ou "dead neurons") qui le relu standar souffre avec, la Leaky ReLU garde une petite pente (proportionnelle à α) pour les valeurs négatives, ce qui permet aux neurones de "réagir" même lorsque leurs entrées sont négatives [9]. Elle peut être exprimée mathématiquement par la fonction suivante :

$$f(x) = \begin{cases} x & \text{si } x \geq 0 \\ \alpha x & \text{sinon} \end{cases} \quad \text{Équation 0-6}$$

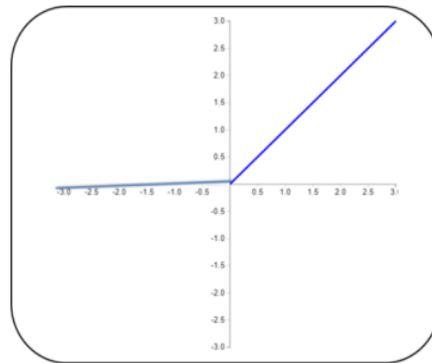


Figure 0-9: Graph de la fonction d'activation leaky ReLu [6]

1.4.1.7 La fonction d'activation PReLU : (Paramétric ReLU)

La fonction PReLU (Parametric ReLU) est une généralisation de la Leaky ReLU, qui introduit un paramètre d'activation pour les entrées négatives, permettant au modèle d'apprendre cette pente au lieu de la fixer à une valeur constante comme dans la Leaky ReLU. Elle peut être exprimé mathématiquement par la fonction suivante :

$$f(x) = \begin{cases} x & \text{si } x \geq 0 \\ \alpha x & \text{sinon} \end{cases} \quad \text{Équation 0-7}$$

Où α est un paramètre appris pendant l'entraînement du modèle pour les valeurs négatives de f .

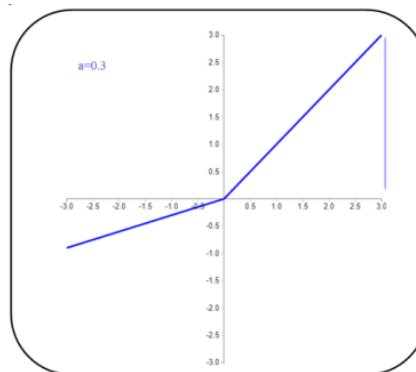


Figure 0-10: Graph de la fonction d'activation paramétrique ReLu [6]

1.4.1.8 La fonction d'activation ELU

La fonction ELU (Exponential Linear Unit) qui a été introduite pour résoudre certains problèmes liés à la ReLU et à la Leaky ReLU, comme la saturation et l'activation des neurones morts. L'ELU permet de rendre la sortie plus lisse et plus proche de zéro pour les valeurs négatives, ce qui peut améliorer la convergence et la performance du modèle. Elle est définie par :

$$f(x) = \begin{cases} x & \text{si } x \geq 0 \\ \alpha (e^x - 1) & \text{sinon} \end{cases} \quad \text{Équation 0-8}$$

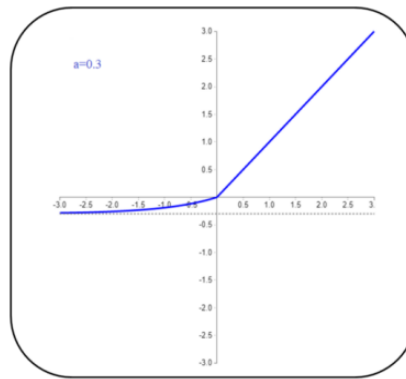


Figure 0-11: Graph de la fonction d'activation ELU [6]

1.4.1.9 La fonction d'activation softmax

Softmax est l'équivalent de la sigmoïde dans les problèmes de classification. Ainsi, la sigmoïde est utilisée dans la classification binaire et Softmax est utilisée dans la classification multiple. Cela signifie que le résultat est la probabilité de prédiction pour chaque classe[8]. Pour ce faire, elle est utilisée dans la couche de sortie dans les réseaux de neurones. Elle est définie par :

$$f(x)_i = \frac{e^{x_i}}{\sum_{k=1}^k e^{x_k}} \quad \text{avec } i=1, \dots, k \quad \text{Équation 0-9}$$

1.4.2 Les fonctions de perte

La fonction de perte, ou fonction de mesure de l'erreur, joue un rôle essentiel dans l'apprentissage automatique et profond. Elle permet de quantifier l'erreur commise par le modèle lors de ses prédictions, ce qui influence directement l'ajustement des poids au cours de l'entraînement. Le principe de ces fonctions repose sur le calcul de la différence entre la valeur réelle et la valeur prédite par le modèle[10]. Il existe plusieurs types de fonctions de perte, chacune adaptée à des situations spécifiques. Parmi les plus courantes, on trouve l'erreur quadratique moyenne (MSE),

l'erreur absolue moyenne (MAE) et d'autres variantes comme l'entropie croisée, Perte de charnière, Pert huber...

1.4.2.1 Erreur quadratique moyenne (MSE)

L'erreur quadratique moyenne (MSE) est l'une des fonctions de perte les plus couramment utilisées pour mesurer l'erreur d'un modèle. Elle calcule la moyenne des carrés des différences entre les valeurs prédites par le modèle et les valeurs réelles présentes dans la base de données. La mise au carré de ces différences permet de pénaliser davantage les grandes erreurs, rendant ainsi l'optimisation plus sensible aux écarts importants entre les prédictions et les valeurs cibles. Cette propriété fait du MSE un choix privilégié pour les problèmes de régression, bien qu'il soit plus sensible aux valeurs aberrantes en raison de l'amplification des erreurs élevées[11]. L'équation mathématique de l'erreur quadratique moyenne (EQM) est la suivante :

$$MSE = (1/n) * \sum (y_i - \bar{y})^2 \quad \text{Équation 0-10}$$

Où :

- n est le nombre d'échantillons dans l'ensemble de données
- y_i est la valeur prédite pour le i-ième échantillon
- $\bar{y}$ est la valeur cible pour le i-ième échantillon

1.4.2.2 Erreur absolue moyenne (MAE)

La fonction MAE (Mean Absolute Error) est une fonction de perte utilisée pour calculer l'erreur dans un algorithme d'apprentissage. Elle mesure l'écart moyen entre les valeurs prédites et les valeurs cibles en prenant leur différence absolue. Contrairement à la fonction MSE, qui met davantage en évidence les erreurs élevées en les élevant au carré, la MAE traite toutes les erreurs de la même manière, sans accorder plus de poids aux écarts importants. Cela en fait une option plus robuste aux valeurs aberrantes, bien qu'elle puisse être moins efficace lorsque des ajustements précis sont nécessaires[11]. L'équation mathématique de l'erreur absolue moyenne (MAE) est la suivante :

$$MAE = (1/n) * \sum |y_i - \bar{y}| \quad \text{Équation 0-11}$$

Où :

- n est le nombre d'échantillons dans l'ensemble de données
- y_i est la valeur prédite pour le i-ième échantillon
- $\bar{y}$ est la valeur cible pour le i-ième échantillon

1.4.2.3 Pert huber

La perte de Huber est une fonction de perte hybride qui combine les avantages de la MSE (Erreur Quadratique Moyenne) et de la MAE (Erreur Absolue Moyenne). Comme la MAE, elle est moins sensible aux valeurs aberrantes, ce qui la rend plus robuste dans les jeux de données contenant des outliers, Comme la MSE, elle pénalise également les petites erreurs, ce qui permet un ajustement plus précis des modèles. Cette nature hybride fait de la perte de Huber un bon compromis, particulièrement utile dans les problèmes de régression, où l'on souhaite réduire l'impact des valeurs extrêmes tout en maintenant une bonne précision pour les erreurs plus petites [10]. L'équation mathématique de la perte de huber est la suivante :

$$\begin{aligned} L(\delta, y, f(x)) &= (1/2) * (f(x) - y)^2 \quad \text{if } |f(x) - y| \leq \delta \\ &= \delta * |f(x) - y| - (1/2) * \delta^2 \quad \text{if } |f(x) - y| > \delta \end{aligned} \quad \text{Équation 0-12}$$

Où :

- L représente la fonction de perte de Huber
- δ est le paramètre delta, qui détermine le seuil de commutation entre les composantes quadratique et linéaire de la fonction de perte.
- y est la valeur réelle ou la valeur cible
- $f(x)$ est la valeur prédite

1.4.3 Les algorithmes d'optimisation

Les algorithmes d'optimisation sont des algorithmes très importants dans le domaine de l'intelligence artificielle, et tout ce qui est apprentissage automatique et profond « Deep and machine Learning », Cela se fait via l'algorithme de rétropropagation qui propagent les erreurs de la couche de sortie vers l'arrière et met à jour les paramètres couche par couche. Les algorithmes d'optimisation permettent de modifier les paramètres d'un réseau de neurone pour minimiser la fonction de perte comme Erreur quadratique moyenne (MSE), Erreur absolue moyenne (MAE), Perte de charnière, Perte d'entropie croisée, Perte Huber.

La descente de gradient et l'un des algorithmes d'optimisation qui joue un rôle crucial dans le processus d'apprentissage, une série de méthodes d'optimisation ont été proposées pour améliorer les performances et l'efficacité des modèles.

1.4.3.1 La descente de gradient (GD)

Le Gradient Descent est une méthode d'optimisation simple à implémenter et largement applicable, ce qui signifie qu'il peut être utilisé dans une variété de contextes et d'algorithmes d'apprentissage. Le Gradient Descent suit une approche itérative pour trouver les paramètres optimaux (par exemple, les poids d'un réseau de neurones) en se déplaçant progressivement dans la direction de la pente négative du gradient de la fonction de coût [12].

Le terme gradient désigne la dérivée. Ainsi, la méthode calcule la dérivée partielle de la fonction de coût par rapport à chaque paramètre. Le gradient indique comment chaque paramètre influence l'erreur.

Différentes variantes de l'algorithme d'apprentissage par gradient sont disponibles et couramment utilisées, notamment les suivantes :

1.4.3.2 La descente du gradient par batch

Le Batch Gradient Descent (BGD) ajuste les poids d'un modèle en minimisant une fonction de perte. Il détecte l'erreur sur l'ensemble des données d'entraînement, mais ne met à jour les poids qu'après avoir évalué tous les exemples[13]. Ce processus, appelé époque d'entraînement, assure une convergence stable et un gradient d'erreur équilibré, offrant ainsi une meilleure précision. Cependant, BGD nécessite de charger toutes les données en mémoire, ce qui peut être coûteux pour de grands ensembles. De plus, il peut se coincer dans un minimum local [14]. Le BGD met à jour les paramètres du modèle θ selon la formule suivante :

$$\theta(\tau) = \theta(\tau-1) - \eta \cdot \nabla_{\theta} L(\theta(\tau-1); T) \quad \text{Équation 0-13}$$

Où:

- $\theta(\tau)$ = Position suivante
- $\theta(\tau-1)$ = Position actuelle
- η = Taux d'apprentissage (taille du pas)
- $\nabla_{\theta} L(\theta(\tau-1); T)$ = désigne la dérivée de la fonction de perte $L(\theta; T)$ par rapport à la variable θ basée sur l'ensemble d'apprentissage complet T et la valeur vectorielle variable $\theta(\tau-1)$ obtenue dans l'itération $\tau - 1$.
- T = the complete training set.

1.4.3.3 Descente du gradient stochastique

Le SGD (Stochastic Gradient Descent) est un algorithme d'optimisation caractérisé par son aspect stochastique, qui fait référence à l'utilisation d'échantillons aléatoires de données pour l'évaluation du modèle à chaque itération, plutôt que d'utiliser l'ensemble complet du jeu de données, cet échantillonnage aléatoire est réalisé en mélangeant les données, ce qui introduit une plus grande variabilité dans l'optimisation et permet d'explorer différentes directions dans l'espace des solutions, mais elle peut également provoquer de fortes fluctuations de la fonction de perte dans le processus de mise à jour [13]. Le SGD met à jour les paramètres du modèle θ selon la formule suivante :

$$\theta(\tau) = \theta(\tau-1) - \eta \cdot \nabla L(\theta(\tau-1); (x_i, y_i)) \quad \text{Équation 0-14}$$

Où :

- $\theta(\tau)$ = Position suivante
- $\theta(\tau-1)$ = Position actuelle
- η = Taux d'apprentissage (taille du pas)
- $L(\theta; (x_i, y_i)) = (F(x_i; \theta), y_i)$ désigne la perte introduite par le modèle sur l'instance de données (x_i, y_i) .

1.4.3.4 Descente du gradient avec mini-batch

Le Descente du gradient avec Mini-Batch (ou Mini-Batch Gradient Descent) est une méthode d'optimisation qui combine les avantages du Batch Gradient Descent et du Stochastic Gradient Descent (SGD). Elle utilise des sous-ensembles (ou *mini-batches*) d'exemples de données à chaque itération pour calculer le gradient et mettre à jour les paramètres du modèle [13].

Si la base de données contient T exemples, un mini-batch contient un sous-ensemble de B exemples (où $B \subset T$), La mise à jour des paramètres θ s'effectue comme suit :

$$\theta(\tau) = \theta(\tau-1) - \eta \cdot \nabla L(\theta(\tau-1); B) \quad \text{Équation 0-15}$$

Où :

- $\theta(\tau)$ = Position suivante
- $\theta(\tau-1)$ = Position actuelle
- η : le taux d'apprentissage.
- $L(\theta; B)$ désigne le terme de perte introduit par le modèle sur le mini-lot B .

La mise à jour consiste à calculer le gradient moyen des exemples du mini-batch et à ajuster les paramètres θ en conséquence.

Le Gradient Descent est très sensible au choix du taux d'apprentissage (η), un taux d'apprentissage mal choisi peut entraîner plusieurs problèmes qui affectent la convergence et la performance de l'algorithme [13]. Si le taux d'apprentissage est trop grand, les mises à jour des paramètres du modèle sont trop importantes à chaque itération, cela peut entraîner un dépassement du minimum global ou une oscillation autour de celui-ci sans jamais y converger et si un taux d'apprentissage trop faible entraîne des mises à jour des paramètres très petites, cela ralentit considérablement la convergence et peut bloquer le modèle dans un minimum local.

Pour résoudre les problèmes précédents, plusieurs algorithmes sont proposés :

1.4.3.5 Momentum

Le Momentum est une technique d'optimisation qui améliore les performances des algorithmes de descente de gradient en ajoutant une mémoire des gradients passés, ce qui permet des mises à jour plus efficaces et une convergence plus rapide.

Momentum met à jour les paramètres avec l'équation suivante :

$$\begin{aligned}\theta(\tau) &= \theta(\tau-1) - \eta \cdot \Delta v(\tau) \text{ avec} & \text{Équation 0-16} \\ \Delta v(\tau) &= \rho \cdot \Delta v(\tau-1) + (1 - \rho) \cdot \nabla \theta L(\theta(\tau-1))\end{aligned}$$

$v(\tau)$ désigne le terme d'impulsion introduit pour enregistrer les gradients historiques jusqu'à l'itération τ . Paramètre $\rho \in [0, 1]$ désigne le poids du terme de quantité de mouvement.

1.4.3.6 Gradient accéléré de Nesterov

Le gradient accéléré de Nesterov (Nesterov Accelerated Gradient, ou NAG) est une méthode d'optimisation avancée qui améliore la descente de gradient avec Momentum. Son objectif est d'accélérer la convergence vers un minimum global tout en réduisant les oscillations.

Le gradient accéléré de Nesterov anticipe la position future ($\theta_{\text{lookahead}}$) avant de calculer le gradient [13]. Cette anticipation permet de mieux guider la descente et d'accélérer la convergence.

$$\begin{aligned}\theta^{(t)} &= \theta^{(t-1)} - \eta \cdot \nabla_v^{(t)} & \text{Équation 0-17} \\ \text{Avec : } \theta_{\text{lookahead}} &= \theta^{(t-1)} - \eta \cdot \rho \nabla_v^{(t-1)} \\ \nabla_v^{(t)} &= \rho \nabla_v^{(t-1)} + (1 - \rho) \cdot \nabla_{\theta} L(\theta_{\text{lookahead}})\end{aligned}$$

1.4.3.7 Adagrad

Adagrad (Adaptive Gradient Algorithm) est un algorithme d'optimisation adapte dynamiquement le taux d'apprentissage pour chaque paramètre du modèle en fonction de l'historique des gradients. Cela le rend particulièrement utile pour des données ou des modèles où les caractéristiques ont des échelles différentes.

Adagrad ajuste le taux d'apprentissage de chaque paramètre en fonction de la somme cumulative des gradients au carré de ce paramètre[13]. L'équation d'apprentissage d'Adagrad peut être exprimée par :

$$\theta^{(t)} = \theta^{(t-1)} - \frac{\eta}{\sqrt{\text{diag}(G^t + \epsilon.I)}} \cdot g^{(t-1)} \text{ avec } \begin{cases} g^{(t-1)} = \nabla_{\theta} L(\theta^{(t-1)}) \\ G^t = \sum_{t=0}^{t-1} g^{(t)2} \end{cases} \quad \text{Équation 0-18}$$

$\text{diag}(G)$ définit une matrice diagonale de mêmes dimensions que G mais avec seulement les éléments sur la diagonale de G . $G(t)$ la matrice qui conserve les enregistrements des gradients historiques calculés depuis le début jusqu'à l'itération actuelle.

1.4.3.8 RMSprop

RMSprop (Root Mean Square Propagation) est un algorithme d'optimisation qui améliore les limitations de l'algorithme Adagrad. Il utilise une approche similaire pour adapter le taux d'apprentissage de chaque paramètre individuellement, mais au lieu d'accumuler les gradients au carré sur toutes les itérations (comme dans Adagrad), il utilise une moyenne mobile exponentielle des gradients au carré [14]. Cela permet d'éviter la diminution excessive du taux d'apprentissage, un problème rencontré avec Adagrad. Les équations de mise à jour des paramètres dans RMSprop peuvent être représentées par les équations suivant

$$\theta^{(t)} = \theta^{(t-1)} - \eta \frac{\eta}{\sqrt{\text{diag}(G^t + \epsilon.I)}} \cdot g^{(t-1)} \quad \text{avec } \begin{cases} g^{(t-1)} = \nabla_{\theta} L(\theta^{(t-1)}) \\ G^t = \rho G^{(t-1)} + (1 - \rho)g^{(t)}(g^{(t-1)})^T \end{cases} \quad \text{Équation 0-19}$$

Le paramètre ρ désigne le poids des gradients historiquement accumulés, ρ est généralement fixé à 0,9. Dans l'équation ci-dessus, le terme dénominateur calculé par :

$$\text{RMS}(g^{(t-1)}\rho) = \sqrt{\text{diag}(G^t + \epsilon.I)} \quad \text{Équation 0-20}$$

Par conséquent, l'équation de mise à jour s'écrit généralement comme suit :

$$\theta^{(t)} = \theta^{(t-1)} - \frac{\eta}{\text{RMS}(g^{(t-1)})} \cdot g^{(t-1)} \quad \text{Équation 0-21}$$

1.4.3.9 Adadelata

Adadelata est un algorithme d'optimisation qui améliore les limitations d'Adagrad, notamment la diminution excessive du taux d'apprentissage au fil des itérations. Proposé par Matthew D. Zeiler en 2012 [15], Adadelata ajuste dynamiquement les mises à jour des paramètres sans nécessiter de taux d'apprentissage global (η), ce qui simplifie son utilisation, Formellement, l'équation de mise à jour des paramètres dans Adadelata donnée par :

$$\theta(\tau) = \theta(\tau-1) - \frac{\eta}{\text{RMS}(g(\tau-1))} g(\tau-1) \quad \text{Équation 0-22}$$

RMS conserve les enregistrements des $\Delta\theta$ dans les itérations précédentes jusqu'à l'itération précédente $\tau - 1$.

Adadelata n'utilise pas le taux d'apprentissage dans l'équation de mise à jour, ce qui corrige efficacement deux faiblesses d'Adagrad précédemment introduites.

1.4.3.10 Adam

L'algorithme Adam (Adaptive Moment Estimation) est l'un des algorithmes d'optimisation les plus populaires en apprentissage automatique, particulièrement pour l'entraînement des réseaux de neurones profonds. Il combine les idées de Momentum et RMSprop pour obtenir une méthode adaptative et efficace. Adam est particulièrement apprécié pour sa robustesse et sa capacité à converger rapidement.

Adam utilise deux moyennes mobiles exponentielles pour suivre :

1. Le premier moment des gradients (moyenne des gradients, comme dans Momentum).

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \quad \text{Équation 0-23}$$

Où :

- m_t est une estimation du premier moment (moyenne des gradients).
- β_1 (≈ 0.9) est le facteur de décroissance pour la moyenne mobile.

2. Le second moment des gradients (variance des gradients, comme dans RMSprop).

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2 \quad \text{Équation 0-24}$$

Où :

- v_t Est une estimation du second moment (variance des gradients).
- β_2 (≈ 0.999) est le facteur de décroissance pour la moyenne mobile.

3. Pour corriger le biais initial (lorsque les estimations m_t et v_t sont biaisées vers 0), on applique les formules suivantes :

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t} \quad \text{Équation 0-25}$$

4. Finalement, les paramètres θ sont mis à jour selon :

$$\theta^{(t)} = \theta^{(t-1)} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad \text{Équation 0-26}$$

Où :

- η est le taux d'apprentissage global.
- ϵ est une petite constante pour éviter la division par zéro.

1.4.3.11 Nadam

Nadam (Nesterov-accelerated Adaptive Moment Estimation) est une variante de l'algorithme Adam qui intègre le principe de l'accélération de Nesterov. Il combine les avantages d'Adam (optimisation adaptative avec momentum et moyennes mobiles) et de l'accélération de Nesterov pour obtenir une convergence plus rapide et stable. Cet algorithme mettra à jour la variable de modèle avec l'équation :

$$\theta^{(t)} = \theta^{(t-1)} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad \text{avec :} \quad \hat{m}_t = \beta_1 \cdot \hat{m}_{t-1} + \frac{(1-\beta_1)}{(1-\beta_1^t)} \cdot g_t \quad \text{Équation 0-27}$$

1.5 Apprentissage profond (Deep learning ou DL)

L'apprentissage profond a été introduit en 2006, est une sous-discipline de l'intelligence artificielle (IA) et du machine learning (apprentissage automatique). Il est basé actuellement sur les réseaux de neurone artificiel (ANN) inspire du fonctionnement du cerveau humain [3].

Le Deep Learning s'est imposé comme une technologie révolutionnaire car il est capable de résoudre des problèmes difficiles que les approches classiques ne parviennent pas à aborder efficacement. Il excelle notamment dans des tâches nécessitant la reconnaissance de modèles dans de grandes quantités de données, comme la reconnaissance d'image, le traitement du langage naturel, les systèmes de recommandation, la conduite autonome, les jeux [16].

1.5.1 Les réseaux de neurones convolutifs

Un réseau de neurones convolutif (*Convolutional Neural Network* ou CNN), également connu sous le nom de ConvNet, est une technique d'apprentissage profond (Deep Learning). Il

est composé de plusieurs couches : une couche d'entrée, des couches cachées et une couche de sortie. Parmi les couches cachées, on trouve notamment les couches convolutives, qui sont responsables de l'extraction automatique des caractéristiques [17].

Les CNN sont exceptionnellement efficaces pour les tâches liées à la vision par ordinateur, comme la reconnaissance d'objets, la détection de visages ou encore la segmentation d'images

1.5.1.1 Les couches d'un CNN

L'architecture générale d'un réseau de neurones convolutif (CNN) est basée sur une structure hiérarchique de couches, organisées en séquence. Ces couches sont conçues pour transformer progressivement les données d'entrée brutes en une représentation de haut niveau, permettant de réaliser des tâches comme la classification ou la détection d'objets [18].

Un CNN se compose de trois types principaux de couches, chacune ayant un rôle spécifique où les couches convolutives et de regroupement extraient les caractéristiques locales des données d'entrée, tandis que les couches entièrement connectées réalisent les prédictions finales [18].

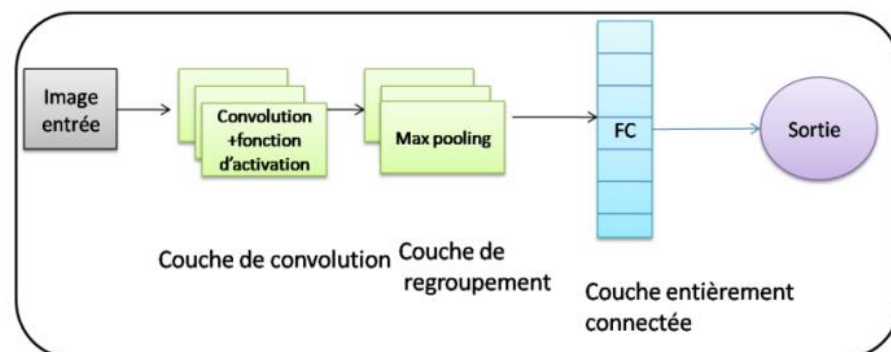


Figure 0-12: Une architecture typique d'un CNN[1]

1.5.1.2 La Couche de convolution

Ces couches constituent le cœur d'un CNN et sont responsables de l'extraction des caractéristiques à partir de l'application des filtres convolutifs sur l'entrée pour détecter des motifs spécifiques (bords, textures, formes, etc.). Ces filtres (ou noyaux) de petite taille, glissent sur les données d'entrée, comme une image ou les cartes de caractéristiques et appliquent une opération de convolution. Cette opération est une opération linéaire qui applique un tableau de nombres appelé noyau à un tableau d'entrée appelé tenseur. Le produit élément par élément entre chaque élément du noyau et le tenseur est calculé à chaque position du tenseur et additionné pour obtenir la valeur de sortie.

à la position correspondante du tenseur de sortie, appelée la carte de caractéristique. La convolution est répétée plusieurs fois sur plusieurs noyaux et possède deux hypermètres : la taille du noyau (3×3 , 5×5 ou 7×7) et le nombre de noyaux qui déterminent la profondeur des cartes de caractéristiques de sortie [16].

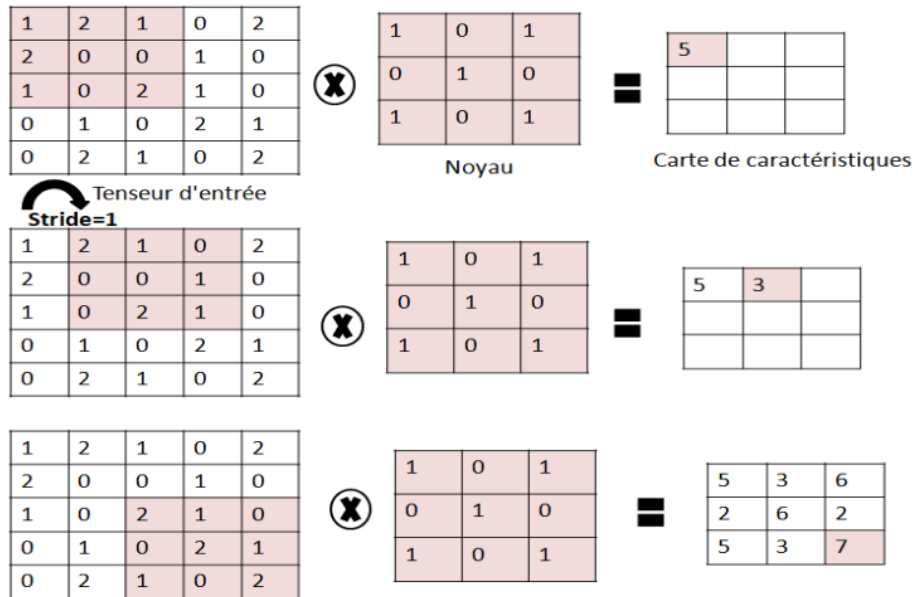


Figure 0-13: Exemple d'opération de convolution avec un noyau de taille 3×3 sans padding [16].

Le stride représente la distance entre la position de deux filtres consécutifs, Il choisit généralement à 1.

Le padding est une technique qui consiste à Ajoute des pixels vides (des zéros) autour de l'entrée pour conserver sa taille après convolution.

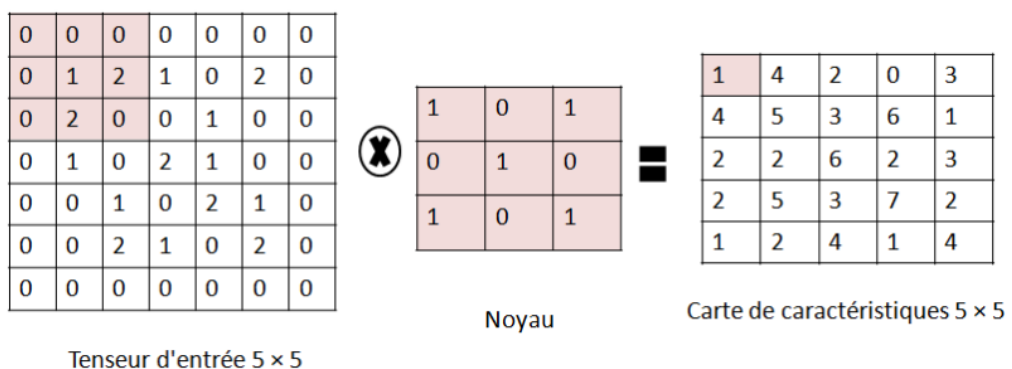


Figure 0-14 : Exemple d'opération de convolution avec padding, un tenseur d'entrée de taille 5×5 , noyau de 3×3 et un stride de 1 [16].

La sortie de la convolution est ensuite passée à travers une fonction d'activation non linéaire, la fonction d'activation la plus couramment utilisée aujourd'hui étant ReLU.

1.5.1.3 Couche de pooling

La tâche principale de la couche de pooling est le sous-échantillonnage des cartes de caractéristiques. Le pooling réduit les cartes de caractéristiques de grande taille pour créer des cartes plus petites, en conservant la majorité des informations (ou caractéristiques) dominantes à chaque étape. La taille du filtre, stride et le padding sont des hyperparamètres dans les opérations de pooling, similaires aux opérations de convolution. Plusieurs types de méthodes de pooling sont disponibles : tree pooling, gated pooling, average pooling, min pooling, max pooling, global average pooling (GAP), et global max pooling [1].

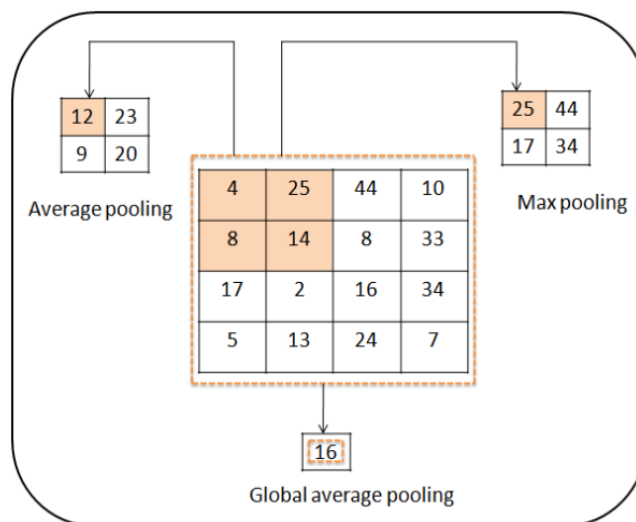


Figure 0-15: Types d'opération de pooling [16].

1.5.1.4 La couche fully connected (FC)

Cette couche est la dernière couche du CNN, elle est appelée fully connected car chaque neurone est connecté à chaque neurone de la couche précédente. Une couche fully connected prend la carte de caractéristique de la sortie et la convertit en un vecteur. Chaque couche fully connected est suivie d'une fonction d'activation [19].

1.6 Quelques Architecture basées CNN

Plusieurs architectures de réseaux ont été utilisées dans la littérature pour les tâches de classification d'images. VGG16, Inception Net et Resnet sont parmi les plus notables.

1.6.1 VGG

VGG (Visual Geometry Group) a été développé par le groupe de vision par ordinateur de l'Université d'Oxford (Simonyan et Zisserman). Il a été introduit en 2014 dans le cadre de la compétition ImageNet Large Scale Visual Recognition Challenge (ILSVRC). L'architecture de VGG est caractérisée par sa simplicité et se compose principalement de couches convolutionnelles, suivies de couches entièrement connectées (fully connected). Les couches de convolution de VGG utilisent systématiquement des filtres 3×3 $\times$ 3×3 , qui sont les plus petits filtres possibles, au lieu d'utiliser des filtres plus grands (par exemple, 5×5 $\times$ 5×5 ou 7×7 $\times$ 7×7), tout en permettant de capturer des motifs [20].

L'architecture de VGG se décline en plusieurs variantes en fonction de sa profondeur : VGG-11, VGG-13, VGG-16 et VGG-19. Le nombre correspond à l'ensemble du nombre de couches de convolution et de couches entièrement connectées, les VGG-16 et VGG-19 ayant obtenu des résultats exceptionnels [16].

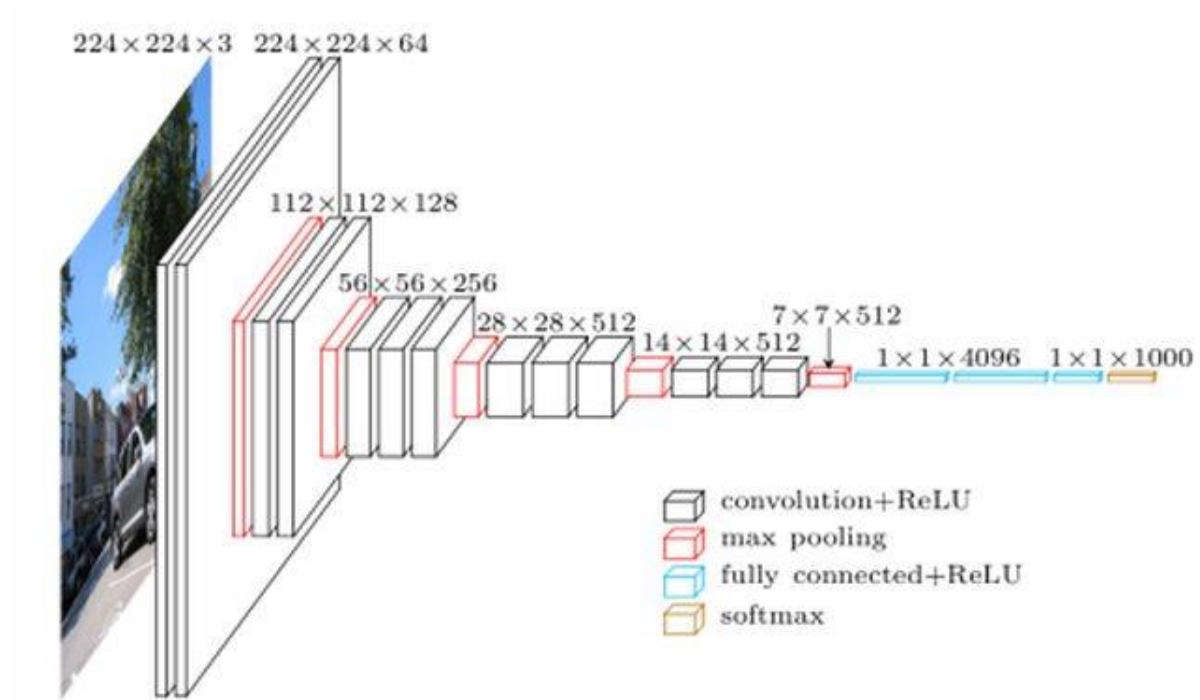


Figure 0-16: L'architecture VGG16

1.6.2 GoogLeNet

GoogLeNet a été conçu pour résoudre certaines des limitations des architectures précédentes, notamment la profondeur du réseau, le nombre de paramètres et la consommation de mémoire. Le modèle est particulièrement connu pour son innovatif Inception module, qui a amélioré l'efficacité du réseau en permettant d'appliquer plusieurs tailles de filtres (par exemple, 1×1 , 3×3 , 5×5) dans une même couche, puis de concaténer les résultats. L'objectif est de capturer

une gamme plus large de motifs dans les différentes échelles, tout en réduisant le nombre de paramètres [21].

L'Inception Module utilise également des couches de **réduction de dimension** comme des couches 1×1 convolutives pour réduire la complexité computationnelle et la mémoire nécessaire, ce qui rend l'architecture plus efficace. GoogLeNet est un modèle relativement profond, avec 22 couches (c'est-à-dire un réseau de 22 couches de poids) [21].

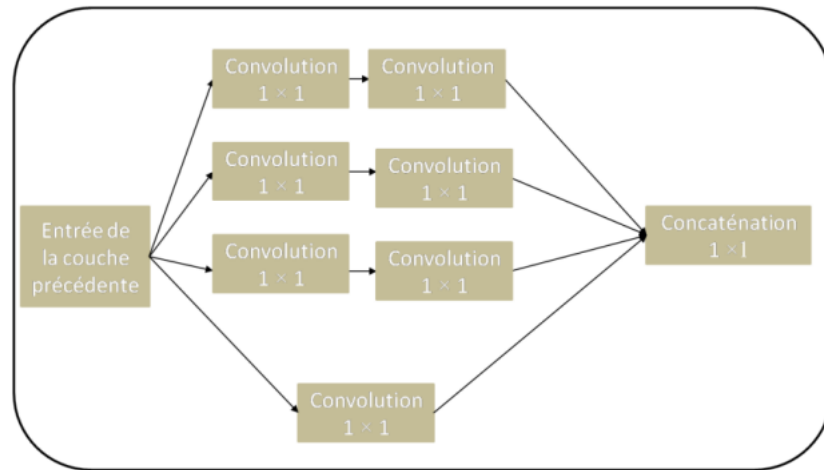


Figure 0-17: Architecture GoogLeNet [21]

1.6.3 ResNet

ResNet (Residual Network), introduit par Kaiming He et al. en 2015, est une architecture de réseau de neurones profonds conçue pour résoudre le problème de la dégradation des performances dans les réseaux très profonds. Cette dégradation se produit lorsque l'ajout de couches supplémentaires entraîne une baisse de précision. ResNet surmonte ce défi grâce aux connexions résiduelles (*skip connections*), qui permettent de contourner une ou plusieurs couches, facilitant ainsi la propagation des gradients et rendant possible l'entraînement de réseaux très profonds, avec jusqu'à 152 couches ou plus [22].

Les blocs fondamentaux de ResNet sont :

1. **Les blocs convolutifs**, qui reposent sur la fonction d'activation ReLU et la normalisation par lots (*Batch Normalization*) pour stabiliser l'entraînement et accélérer la convergence.
2. **Les blocs d'identité**, qui utilisent les connexions résiduelles pour transmettre directement les informations d'entrée sans modification ou avec une transformation minimale, renforçant la capacité d'apprentissage des couches profondes.

Grâce à cette structure, ResNet a redéfini l'approche des réseaux convolutifs profonds, établissant de nouveaux standards en termes de profondeur et de performances sur des tâches complexes comme la classification d'images.

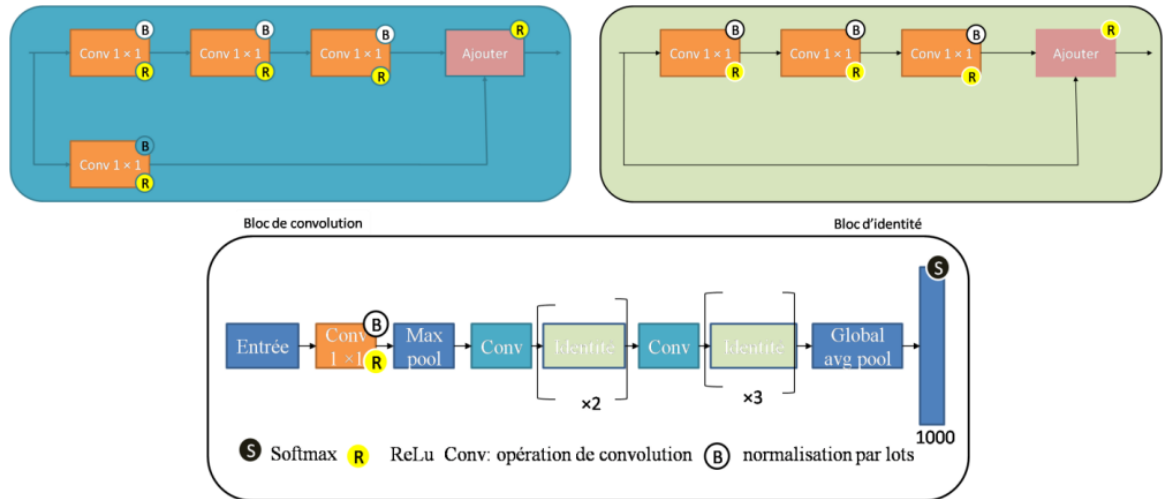


Figure 0-18: L'architecture ResNet [22].

1.6.4 EfficientNet

EfficientNet est une famille de modèles CNN (réseaux convolutifs) conçus pour offrir une grande précision tout en réduisant les coûts computationnels pour l'entraînement et l'inférence. Entraînés sur la base de données ImageNet, ces modèles sont particulièrement adaptés aux tâches de classification d'images. Classés de B0 à B7 en fonction de leur complexité et de leur précision, les modèles EfficientNet optimisent les performances sans relation linéaire directe avec le nombre de paramètres. Cette efficacité les rend adaptés à des tâches nécessitant un équilibre entre précision et charge de calcul [23].

Une des caractéristiques distinctives d'EfficientNet est l'utilisation de la fonction d'activation Swish, qui s'est avérée plus précise et plus efficace en termes de calcul que la fonction ReLU (Rectified Linear Unit), traditionnellement utilisée. Swish contribue à améliorer la précision tout en minimisant les ressources nécessaires.

EfficientNet repose sur une stratégie innovante appelée compound scaling, qui optimise simultanément trois dimensions fondamentales du réseau :

- Profondeur (d) : le nombre de couches dans le réseau.
- Largeur (w) : le nombre de filtres dans chaque couche convolutive.

- Résolution(r) : la hauteur et la largeur de l'image d'entrée.

Un coefficient d'échelle (noté ϕ) est utilisé pour ajuster ces trois dimensions de manière cohérente, permettant ainsi d'obtenir un équilibre optimal entre précision et efficacité. Cette méthode permet à EfficientNet d'atteindre des résultats de pointe tout en réduisant le nombre de paramètres par rapport à d'autres architectures [23]. L'équation principale du compound scaling est :

$$d = \alpha^\phi$$

$$w = \beta^\phi$$

$$r = \gamma^\phi$$

$$\text{De telle sorte que : } \alpha \geq 1 \quad \beta \geq 1 \quad \gamma \geq 1 \quad \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$

où :

- α , β , et γ sont des coefficients constants déterminés empiriquement pour contrôler respectivement l'échelle de profondeur, largeur, et résolution.
- ϕ est le coefficient global qui ajuste ces trois dimensions en fonction des ressources disponibles (plus ϕ est grand, plus le modèle est complexe).

EfficientNet est une architecture d'apprentissage profond qui combine efficacité computationnelle et précision, en exploitant une mise à l'échelle cohérente des dimensions du réseau et en intégrant des avancées comme la fonction Swish. Cela en fait un modèle idéal pour des applications nécessitant des performances élevées avec des ressources limitées.

1.7 Apprentissage par transfert

L'apprentissage par transfert est une technique d'apprentissage automatique dans laquelle un modèle préalablement formé sur une tâche ou un domaine spécifique (appelé domaine source) est réutilisé et ajusté (ou affiné) pour une autre tâche ou un autre domaine, souvent apparenté (appelé domaine cible), dans le but d'améliorer ses performances. Cette approche permet de réduire la quantité de données étiquetées nécessaires pour entraîner un modèle sur la tâche cible, tout en augmentant la généralisation [24].

L'objectif de l'apprentissage par transfert est d'appliquer les représentations et les informations apprises dans le domaine source à la tâche ou au domaine cible. La réussite de cette approche dépend en grande partie de la similarité entre les deux domaines. Plus les domaines source et cible sont similaires, plus il est probable que le transfert de connaissances sera efficace.

Cependant, même si les domaines ne sont pas identiques, il est possible que certaines informations importantes (comme des caractéristiques de bas niveau ou des relations générales) puissent être transférées pour améliorer les performances du modèle dans le domaine cible [24].

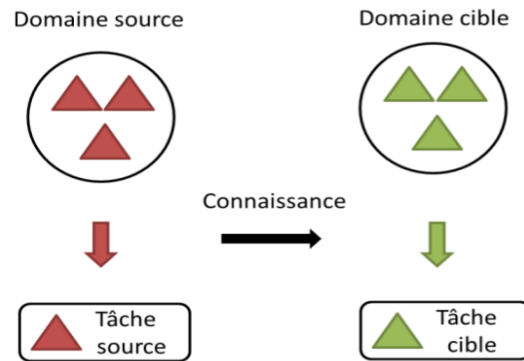


Figure 0-19: Apprentissage par transfert [25]

1.8 Conclusion

Dans ce chapitre, nous avons présenté les concepts fondamentaux des réseaux de neurones et de l'apprentissage profond. Nous avons commencé par présenter le neurone artificiel et biologique. Nous avons présenté les fonctions d'activation telles que Sigmoid, tanh, ReLU et ReLU ...etc., qui jouent un rôle crucial dans le processus de classification. Nous avons également abordé les algorithmes d'optimisation, notamment la descente de gradient, la descente de gradient stochastique élastique, ADAGRAD et ADAM...etc., qui permettent d'ajuster les poids du réseau neuronal au cours du processus de formation. Nous nous sommes ensuite concentrés sur les CNN. Nous avons expliqué le concept de l'opération de convolution, qui est un élément clé des réseaux neuronaux à convolution pour l'extraction de caractéristiques pertinentes. Nous avons exploré d'autres concepts tels que le padding, le pooling et la couche entièrement connectée, qui contribuent à l'efficacité des CNN. Enfin, nous avons mis en évidence les applications pratiques des CNN, telles que VGG16, Inception et ResNet. Ces modèles ont démontré la puissance et le potentiel des CNN dans les applications du monde réel.

Chapitre

2

La gestion des déchets plastiques

1.9 Introduction

Le problème de la gestion des déchets plastiques est devenu une préoccupation majeure à l'échelle mondiale, en particulier avec l'augmentation continue de la consommation de produits plastiques jetables. Ce chapitre vise à explorer les enjeux environnementaux liés à l'accumulation des déchets plastiques, en mettant en lumière l'importance de leur gestion efficace et le rôle crucial du recyclage dans cette démarche.

Nous commencerons par définir la pollution plastique et présenter des statistiques mondiales et locales, notamment en Algérie, pour mieux comprendre l'ampleur de ce phénomène. Ensuite, nous aborderons les effets négatifs de l'élimination anarchique des bouteilles en plastique, qui contribue à la dégradation de notre environnement et à la mise en danger des écosystèmes.

Le chapitre se penchera également sur les différentes méthodes de recyclage et les défis associés, en particulier dans les pays en développement où les infrastructures de gestion des déchets sont encore insuffisantes. Nous présenterons ensuite les méthodes traditionnelles de collecte et de tri du plastique, avant de présenter les solutions innovantes, comme les machines de collecte réversible, qui intègrent des technologies avancées telles que l'intelligence artificielle pour améliorer l'efficacité du processus de tri.

Enfin, nous discuterons des travaux antérieurs dans le domaine du recyclage des plastiques, en particulier ceux qui ont exploré l'utilisation de l'intelligence artificielle et du deep learning pour automatiser le processus de classification des déchets plastiques.

Le but de ce chapitre est de fournir un cadre théorique solide qui servira de base pour comprendre les défis et les solutions existants, tout en préparant le terrain pour la présentation du système innovant que nous proposons dans ce projet

1.10 La pollution

Au cours des dernières décennies, la pollution de l'environnement est devenue l'un des défis majeurs auxquels l'humanité est confrontée. Ses effets ne se limitent plus aux zones industrielles, mais touchent également les régions rurales, les océans et les forêts. Parmi les formes de pollution les plus préoccupantes figurent celle liée au plastique, en raison de sa non-biodégradabilité et de son utilisation massive dans la vie quotidienne.

Les bouteilles plastiques, en particulier, représentent une grande part des déchets plastiques jetés chaque jour. Utilisées principalement pour l'embouteillage de l'eau, des boissons et des jus, elles sont souvent abandonnées dans la nature après usage. Cela engendre une pollution visible des sols et des cours d'eau, et constitue un danger pour les animaux et les écosystèmes [26].

En Algérie, comme dans de nombreux pays en développement, ce problème est aggravé par l'absence de systèmes performants de tri et de recyclage, ainsi que par le manque de sensibilisation écologique chez une partie de la population. D'où l'importance de développer des solutions intelligentes, alliant technologie et engagement citoyen, pour freiner cette menace environnementale croissante.

La pollution plastique désigne l'accumulation des déchets plastiques dans l'environnement terrestre et marin, résultant d'une consommation excessive de plastiques non biodégradables. Ces déchets peuvent persister pendant des siècles, causant une détérioration durable des écosystèmes.

Parmi les principales sources de cette pollution : les sacs plastiques, les bouteilles, les emballages alimentaires, les produits à usage unique, ainsi que les déchets domestiques et industriels non triés [26].

1.10.1 Statistiques mondiales et locales (Algérie) sur les déchets plastiques

Au niveau mondial, plus de 400 millions de tonnes de plastique sont produites chaque année, dont environ 40% sont destinées à un usage unique. Moins de 10% de ces déchets sont recyclés, le reste finissant dans des décharges ou dans la nature[27][27].

En Algérie, on estime la production annuelle de déchets plastiques à environ 1,2 million de tonnes, un chiffre en constante augmentation en raison de la croissance démographique et de la consommation accrue de produits emballés [W13].

1.10.2 Les effets négatifs de l'élimination anarchique des bouteilles en plastique

L'élimination anarchique des bouteilles en plastique constitue l'une des principales sources de pollution environnementale de notre époque. En raison de leur nature non biodégradable, ces déchets persistent dans l'environnement pendant des centaines d'années. Lorsqu'elles sont jetées dans des espaces publics ou naturels, ces bouteilles se fragmentent sous l'effet des rayons UV et des conditions climatiques en microplastiques, de minuscules particules qui infiltrent facilement les sols, les nappes phréatiques et les cours d'eau, compromettant ainsi la qualité des ressources naturelles [28].

Ce type de pollution ne menace pas seulement les écosystèmes terrestres, mais présente également un danger majeur pour la faune, notamment marine. Les bouteilles abandonnées provoquent souvent l'étouffement ou l'empoisonnement d'animaux comme les poissons et les tortues qui les confondent avec de la nourriture. Par ailleurs, ces microplastiques ont été détectés dans l'eau potable, les produits de la mer et même dans le sel de consommation, soulevant des inquiétudes croissantes quant à leur impact potentiel sur la santé humaine [29].

Sur le plan économique, la gestion de ces déchets représente un coût élevé pour les collectivités locales, alors qu'ils pourraient être valorisés dans des filières de recyclage. De plus, l'accumulation de ces déchets dans les villes et les sites touristiques altère l'esthétique des espaces publics et nuit à l'image des territoires [30].

Ces effets combinés mettent en lumière l'urgence d'adopter des solutions durables reposant sur la sensibilisation citoyenne et l'utilisation de technologies modernes, afin de limiter la quantité de bouteilles jetées de manière sauvage et de promouvoir leur réintégration dans l'économie circulaire.

1.11 Introduction sur le recyclage

Dans un contexte mondial marqué par l'augmentation alarmante des déchets, notamment plastiques, le recyclage s'impose comme l'une des solutions les plus efficaces pour limiter l'impact environnemental de la surconsommation. Le recyclage, en tant que processus de traitement des déchets afin de leur donner une seconde vie, constitue un pilier fondamental de l'économie circulaire et du développement durable. Il permet non seulement de réduire la quantité de déchets envoyés en décharge ou incinérés, mais aussi de préserver les ressources naturelles, de diminuer les émissions de gaz à effet de serre et de créer des opportunités économiques, notamment dans les filières vertes [31].

Le recyclage des plastiques, en particulier, revêt une importance capitale face à la pollution croissante générée par ces matériaux à usage unique. Les bouteilles, emballages et sacs en plastique représentent une part importante des déchets ménagers et industriels, dont une faible proportion seulement est réellement recyclée, notamment dans les pays en développement. Cette situation révèle un besoin urgent de renforcer les systèmes de tri, de collecte sélective et de traitement, ainsi que d'intégrer des solutions technologiques innovantes pour améliorer l'efficacité du processus de recyclage.

1.12 Définition de le recyclage

Le recyclage est le processus de transformation des déchets et des matériaux usagés en nouveaux produits pouvant être utilisés dans la fabrication d'autres articles. Ce concept repose sur la récupération de matériaux comme le plastique, le verre, le papier et les métaux, permettant ainsi de réduire l'épuisement des ressources naturelles tout en limitant la quantité de déchets envoyés dans les décharges [32].

1.12.1 Les avantages du recyclage

Le recyclage présente plusieurs avantages environnementaux, économiques et sociaux :

- **Réduction de la pollution** : En réduisant la quantité de déchets envoyés dans les décharges ou les océans, il permet de diminuer la pollution environnementale.
- **Conservation des ressources naturelles** : Le recyclage permet de réduire la dépendance à l'extraction de nouvelles ressources naturelles, comme les minéraux et les combustibles fossiles, préservant ainsi ces ressources pour les générations futures.
- **Création d'emplois** : Le secteur du recyclage génère un nombre considérable d'emplois dans les domaines de la collecte, du tri, du traitement et de la fabrication de produits recyclés. Selon les estimations, ce secteur pourrait créer des centaines de milliers d'emplois, notamment dans les pays en développement.
- **Durabilité économique** : Le recyclage aide à réduire les coûts de production et favorise la durabilité des industries locales en fournissant des matières premières à moindre coût [31].

1.12.2 Les difficultés ou défis du recyclage dans les pays en développement

Malgré ses nombreux avantages, le recyclage rencontre plusieurs défis dans les pays en développement, tels que :

- **Manque de sensibilisation** : De nombreux citoyens dans ces pays manquent de conscience quant à l'importance du recyclage et aux bonnes pratiques pour le mettre en œuvre.
- **Infrastructures insuffisantes** : Beaucoup de pays en développement souffrent d'un manque d'infrastructures modernes de gestion des déchets, comme des stations de tri et de traitement, rendant le recyclage inefficace.
- **Ressources financières limitées** : Le financement limité pour les activités de recyclage empêche le développement de technologies modernes et de systèmes intégrés de gestion des déchets.
- **Travail informel** : Une grande partie des travailleurs dans le secteur de la gestion des déchets dépend de méthodes informelles, ce qui entrave le développement et l'organisation de ce secteur [33].

1.13 Les méthodes traditionnelles de collecte et de tri du plastique

La collecte et le tri des déchets plastiques constituent des étapes essentielles dans le processus de recyclage. Cependant, dans de nombreux pays en développement, ces opérations reposent encore sur des méthodes traditionnelles qui manquent d'efficacité et de sécurité. Voici les principales approches utilisées :

1.13.1 La collecte manuelle :

Les déchets plastiques sont ramassés manuellement par des agents de propreté ou des récupérateurs informels, souvent appelés "chiffonniers", à partir des poubelles publiques, des rues ou des décharges. Bien que cette méthode soit peu coûteuse et facilement accessible, elle est pénible, expose les travailleurs à des risques sanitaires et sécuritaires, et reste limitée en termes de quantité et de qualité de tri.

1.13.2 Les centres de tri :

Ces centres reçoivent des déchets mélangés, qui sont triés manuellement ou à l'aide de lignes semi-automatiques équipées de tapis roulants et de dispositifs mécaniques. Ils améliorent

la qualité des matériaux recyclables, mais nécessitent une main-d'œuvre importante et des coûts opérationnels élevés. Leur efficacité dépend également du niveau de sensibilisation de la population au tri sélectif à la source.

1.13.3 Les machines de tri industrielles :

Utilisées dans des installations de traitement avancées, ces machines reposent sur des technologies sophistiquées, comme le tri par poids, taille, couleur, ou par spectrométrie infrarouge (NIR) pour identifier les types de plastiques. Bien qu'elles soient très précises et efficaces, leur coût élevé d'acquisition et de maintenance limite leur adoption dans les pays à faibles ressources.

1.14 Présentation du système proposé et son rôle

1.14.1 Description du système

Ce projet propose un système intelligent basé sur l'intelligence artificielle pour assurer le tri automatique des bouteilles plastiques. Le système est composé d'un capteur qui détecte la présence d'une bouteille et déclenche automatiquement la caméra pour en capturer une image. Cette image est ensuite analysée par l'intelligence artificielle afin de déterminer si la bouteille est recyclable ou non. En fonction du résultat, un moteur oriente la bouteille vers la direction appropriée (acceptation ou rejet).

Le système intègre également une interface numérique qui affiche l'état de chaque unité (en fonctionnement ou en panne), ainsi qu'une carte indiquant les emplacements où les systèmes sont installés. Ces fonctionnalités font du système un outil performant pour le tri intelligent à la source, tout en permettant un suivi à distance de son fonctionnement et de sa maintenance.

1.14.2 Avantages du système proposé par rapport aux méthodes traditionnelles

Comparé aux méthodes traditionnelles de collecte et de tri des déchets plastiques, le système intelligent proposé présente plusieurs avantages significatifs. Premièrement, il permet un tri à la source immédiat et autonome, ce qui réduit considérablement le besoin de main-d'œuvre humaine et limite les risques sanitaires associés à la manipulation des déchets.

Contrairement au tri manuel ou semi-automatisé qui dépend fortement de l'intervention humaine, ce système garantit une rapidité d'exécution, une précision accrue dans la classification des bouteilles et une réduction des erreurs.

Deuxièmement, l'utilisation de l'intelligence artificielle permet une amélioration continue des performances grâce à l'apprentissage automatique. Cela rend le système adaptable à différents types de bouteilles, même en cas de variations visuelles.

Troisièmement, l'intégration d'une interface numérique permettant le suivi en temps réel de l'état des machines et de leur localisation offre une gestion centralisée efficace, ce qui est totalement absent dans les solutions traditionnelles. De plus, ce système peut être déployé dans des lieux variés tels que les écoles, les centres commerciaux ou les gares, rapprochant ainsi le recyclage des citoyens et facilitant leur participation.

En résumé, le système proposé constitue une solution moderne, autonome et évolutive qui répond aux limites majeures des approches classiques, tout en soutenant les efforts de recyclage durable et intelligent.

1.15 Travaux Antérieurs

Dans ce chapitre, nous présentons un ensemble de travaux et projets antérieurs qui ont abordé des problématiques similaires ou proches de celles traitées dans ce mémoire, notamment l'utilisation de l'intelligence artificielle et de l'apprentissage profond pour la classification de bouteilles ou de déchets, ainsi que leur intégration dans des systèmes matériels intelligents.

L'objectif est de mettre en lumière les approches adoptées par différents chercheurs, en analysant les modèles utilisés (tels que les réseaux de neurones convolutifs, YOLO, Mobile Net...) ainsi que les architectures matérielles associées (Raspberry Pi, Arduino, caméras, moteurs, etc.).

Cette revue comparative permet également d'identifier les forces et les limites de ces travaux, et de souligner la valeur ajoutée du projet réalisé dans le cadre de ce mémoire, tant sur le plan méthodologique que sur le plan de l'implémentation réelle.

L'étude menée par **Deena Mariya**, intitulée “*Reverse Vending Machine for Plastic Bottle Recycling*” [33], présente le développement d'un système de machine de retour

automatique (Reverse Vending Machine) basé sur la vision par ordinateur et l'apprentissage profond. Le modèle a été entraîné à l'aide de TensorFlow et annoté manuellement avec l'outil LabelImg. Des modèles avancés comme Faster R-CNN et RetinaNet ont été testés pour améliorer la précision de détection. Le système comprend également une application Android et un capteur infrarouge (IR) pour la gestion de remplissage, montrant une intégration cohérente entre IA, IoT et interaction utilisateur [34].

Dans une autre étude, **Huma Zia** a développé une machine de recyclage inverse à faible coût pour la gestion des déchets plastiques, reposant sur un modèle MobileNet embarqué sur un dispositif Edge Computing avec un microcontrôleur ESP32 [34]. Le système intègre une caméra pour l'identification des objets, un convoyeur avec bras mécaniques, des capteurs, ainsi qu'un système de pesée pour calcul des points de récompense. L'étude a été appliquée dans des magasins de proximité à Séoul, en Corée du Sud, avec une approche de tri en trois étapes (lecture de code-barres, tri par couleur, et séparation spectrale via capteur NIR). L'efficacité globale du système a atteint jusqu'à 98 % pour certaines catégories [35].

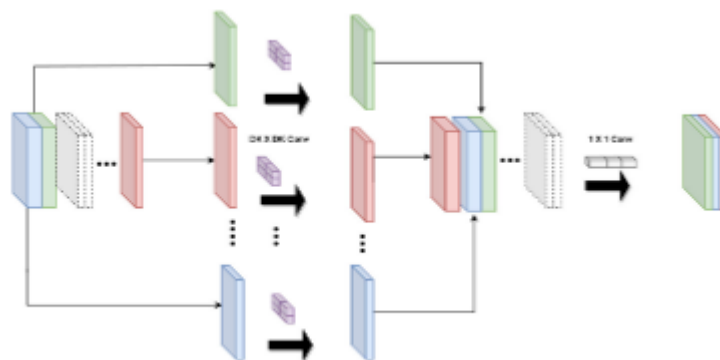


Figure 0-1 Architecture de mobile net utilisée [35]

Wahab et ses collaborateurs ont proposé un prototype de tri intelligent basé sur l'analyse d'image automatisée [36]. Le système utilise une webcam Logitech montée au-dessus d'un tapis roulant pour capturer des images RGB, lesquelles sont ensuite traitées par des filtres de Sobel, débruitées et analysées pour en extraire des caractéristiques géométriques. Plusieurs méthodes de classification ont été évaluées, dont des réseaux de neurones, qui ont donné des taux de reconnaissance proches de 100 % pour le plastique PET. Le traitement rapide (150 ms/image) rend ce système adapté aux applications industrielles [36].

Une autre contribution importante est l'étude *“Enhancement of Image Classification Using Transfer Learning and GAN-Based Synthetic Data Augmentation”* [25], dans laquelle

les chercheurs ont constitué un jeu de données industriel déséquilibré et restreint, qu'ils ont enrichi à l'aide d'un modèle GAN pour la génération d'images synthétiques. En complément, des techniques d'augmentation classiques ont été appliquées. Deux modèles, InceptionV3 et Xception, pré-entraînés sur ImageNet, ont été adaptés pour la tâche, atteignant respectivement 94,7 % et 95,3 % de précision. Ces résultats confirment l'efficacité de la combinaison transfert d'apprentissage et augmentation des données.

Enfin, **Xiao et al. (2022)** ont proposé un cadre nommé *Serial Attention Frame (SAF)* pour la classification multi-étiquettes de bouteilles usagées dans des environnements industriels[37]. Leur architecture repose sur trois modules principaux : un bloc résiduel basé sur ResNet101, un module d'attention mixte (spatiale et par canal), et un module d'auto-attention pour capturer les relations globales dans l'image. Le modèle a été évalué sur une base de données manuelle de 3902 images ainsi que sur les ensembles PASCAL VOC2007 et WIDER Attribute. Les résultats montrent une performance remarquable, avec un mAP de 97,79 % et un CF1 de 94,5 %, ce qui démontre l'efficacité du SAF pour le tri intelligent même dans des contextes de forte variabilité visuelle.

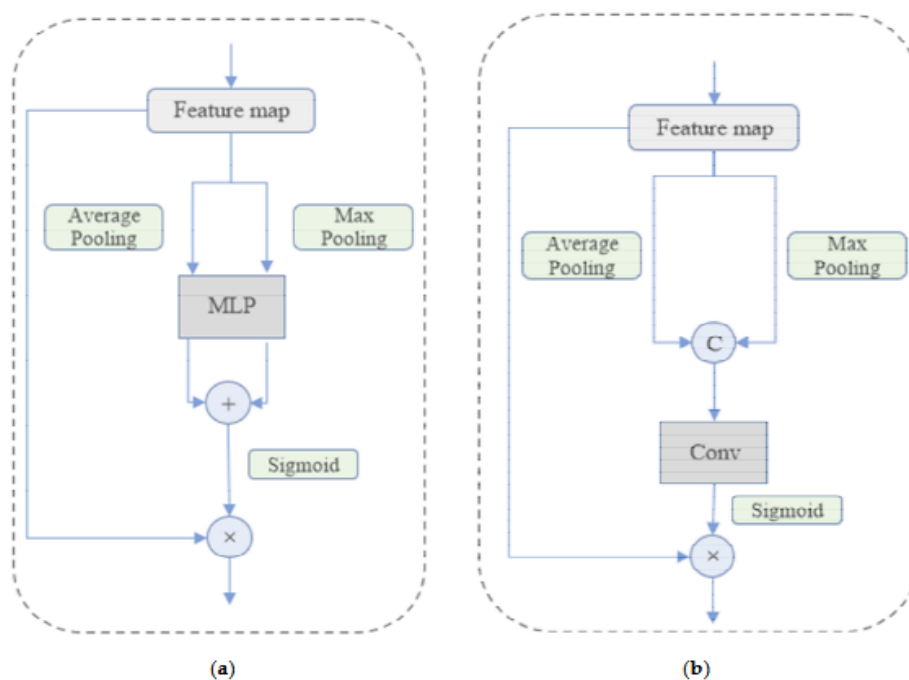


Figure 0-2 Network architecture of channel attention block (a) and spatial attention bloc[25]

1.16 Conclusion

Dans ce chapitre, nous avons présenté une revue complète des concepts relatifs à la pollution plastique et au recyclage, en mettant l'accent sur les méthodes traditionnelles de collecte et de tri des déchets plastiques. Nous avons également discuté des défis rencontrés par les pays en développement pour mettre en œuvre des solutions efficaces de gestion des déchets plastiques et de recyclage. L'impact négatif de l'élimination anarchique des bouteilles en plastique et ses conséquences environnementales ont également été abordés.

De plus, nous avons présenté le système proposé dans le cadre de ce projet, en soulignant ses avantages par rapport aux méthodes traditionnelles, et nous avons examiné les travaux antérieurs dans ce domaine qui ont contribué au développement de solutions technologiques pour résoudre les problèmes de gestion des déchets plastiques à l'aide de l'intelligence artificielle.

Dans le chapitre suivant, nous allons présenter le concept du système intelligent proposé, détailler ses composants et expliquer comment nous les avons intégrés pour créer une solution cohérente et efficace. Nous décrirons également la manière dont nous avons relié les différents éléments du système pour atteindre les objectifs du projet.

Chapitre

3

Conception et Réalisation

1.17 Introduction

Le chapitre de conception constitue une étape clé de ce travail, marquant la transition du cadre théorique vers la mise en œuvre pratique. Dans ce chapitre, nous présentons en détail l'architecture du système de tri intelligent des bouteilles plastiques, en abordant aussi bien les aspects matériels que logiciels.

Nous commencerons par l'identification et la description des composants électroniques et électromécaniques sélectionnés pour la réalisation du projet, tels que l'unité de traitement (Raspberry Pi ou Arduino), les capteurs (par exemple, un capteur infrarouge), les moteurs (comme le NEMA 17), les drivers, l'imprimante thermique, l'écran d'affichage, ainsi que les voyants lumineux. Chaque composant sera décrit en précisant son rôle, ses caractéristiques techniques, et les raisons de son intégration dans le système.

Par la suite, nous détaillerons le schéma de câblage et les connexions physiques entre les composants, en veillant à assurer la cohérence fonctionnelle et la fiabilité électrique du montage. Nous expliquerons également l'interaction entre la partie logicielle (modèle d'intelligence artificielle chargé de la classification) et la partie matérielle, à travers des protocoles de communication tels que le port série entre le Raspberry Pi et l'Arduino.

Enfin, un déroulement fonctionnel du système sera présenté : depuis la détection d'une bouteille par le capteur, en passant par la capture et l'analyse d'image via le modèle d'IA, jusqu'à l'exécution de l'action correspondante (acceptation ou rejet de la bouteille) à l'aide du moteur et l'affichage de la décision via l'écran ou l'imprimante. Ce chapitre permet donc de valider la faisabilité technique du système proposé et de démontrer sa robustesse, sa réactivité et son potentiel d'intégration dans un environnement réel.

1.18 Présentation des composants matériels du système

1.18.1 Raspberry pi 4

Raspberry Pi 4 est une ordinateuse monocarte (SBC – Single Board Computer) puissante, compacte et économique, largement utilisée dans les projets embarqués, notamment ceux intégrant l'intelligence artificielle, la vision par ordinateur, ou le traitement d'image [W1].

Dans le cadre de ce projet, le Raspberry Pi 4 est utilisé comme unité centrale de traitement, capable d'exécuter un modèle d'IA pour la classification des bouteilles, de capturer des images via une caméra, et de communiquer avec d'autres composants tels que l'Arduino, la caméra, l'écran et l'imprimante [W1].

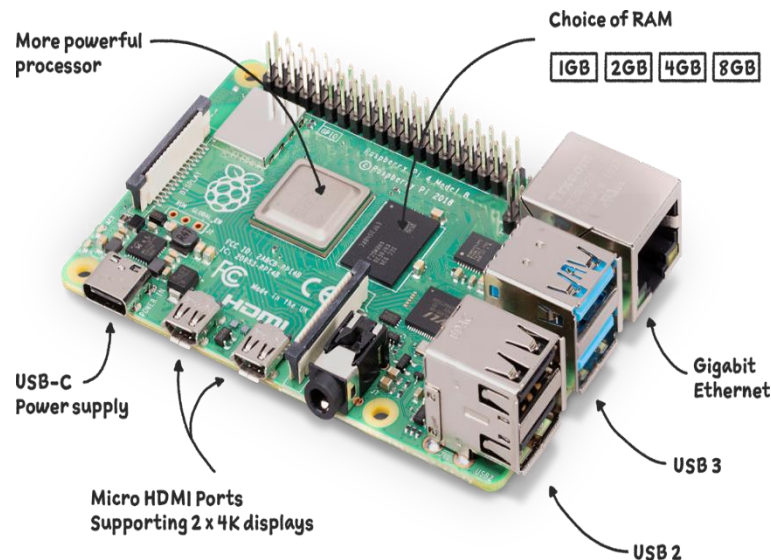


Figure 0-1: image de Raspberry pi 4 [W1]

1.18.2 Principales entrées et sorties – Raspberry Pi 4

Port CSI (Camera Serial Interface)

Utilisé pour connecter la caméra 5MP Night Vision OV5647. Permet la transmission directe de flux vidéo vers le Raspberry Pi pour traitement en temps réel via OpenCV.

Ports GPIO (General Purpose Input/Output)

Utilisés pour Recevoir un signal de déclenchement depuis le capteur infrarouge (IR).

Ports USB 3.0 / 2.0

Peuvent être utilisés pour le clavier ou souris durant le développement et la connexion à des périphériques comme une clé USB.

Port Ethernet (RJ45)

Utilisé pour la communication série avec la carte Arduino. Permet d'envoyer des chaînes de texte simples contenant la classification ("accepté 0" / "rejeté 1") directement au microcontrôleur.

Port HDMI (micro-HDMI)

Permet l'affichage des résultats ou du flux vidéo en direct sur écran externe durant les phases de test.

1.18.3 Arduino MEGA

La carte Arduino MEGA 2560 est une carte de développement basée sur le microcontrôleur ATmega2560, largement utilisée dans les projets nécessitant un grand nombre d'entrées/sorties. Elle dispose de 54 broches numériques dont 15 peuvent être utilisées comme sorties PWM, de 16 entrées analogiques, de 4 ports UART pour la communication série, ainsi que d'une mémoire Flash de 256 Ko, 8 Ko de SRAM et 4 Ko d'EEPROM. Elle se programme via l'environnement Arduino IDE et se connecte à un ordinateur par USB. Grâce à ses nombreuses ressources, elle est particulièrement adaptée aux systèmes complexes nécessitant le contrôle de plusieurs capteurs, moteurs ou modules, et représente une solution puissante et flexible par rapport à d'autres cartes comme l'Arduino Uno [W2].



Figure 0-2: image de l'Arduino MEGA [W2].

1.18.4 Moteur Nema 17 :

Le moteur NEMA 17 est un moteur pas à pas très utilisé dans les imprimantes 3D, les bras robotiques et les projets nécessitant un contrôle de position précis. "NEMA" signifie National Electrical Manufacturer Association, et le chiffre 17 désigne la taille de la face du moteur : 1.7 pouces $\times$ 1.7 pouces (~ 4.3 cm $\times$ 4.3 cm) [W3].



Figure 0-3: image de Nema 17 [W3].

1.18.5 Microstep Driver

Le Microstep Driver est un pilote de moteur pas à pas conçu pour contrôler des moteurs comme le NEMA 17. Il offre un contrôle précis grâce à la technologie de micro-pas (microstepping), qui permet de diviser chaque pas complet en plusieurs sous-pas, améliorant ainsi la précision du positionnement.

Ce pilote fonctionne avec une tension d'entrée comprise entre 9V et 42V DC, et permet un réglage du courant de sortie entre 0.5A et 3.5A, selon les besoins du moteur. Il dispose de commutateurs DIP pour configurer le courant et le mode de pas (pas entier, demi-pas, 1/4, 1/8, jusqu'à 1/32) [W4].

Les signaux de commande sont transmis via trois broches principales :

- PUL (Impulsion) : contrôle du nombre de pas
- DIR (Direction) : sélection du sens de rotation
- ENA (Activation) : activation/désactivation du moteur

La connexion au moteur se fait via les bornes A+, A-, B+, B-, qui alimentent les bobines internes. Ce pilote est idéal pour les projets nécessitant une grande précision, tels que les imprimantes 3D, les traceurs, ou les systèmes de positionnement automatisé [W4].



Figure 0-4: Image de microster driver [W4].

1.18.6 Caméra 5MP OV5647 avec Vision Nocturne

Caméra OV5647 de 5 mégapixels, est une caméra compatible avec le Raspberry Pi. Elle est équipée de LEDs infrarouges qui lui permettent de fonctionner efficacement même dans des conditions de faible luminosité, voire dans l'obscurité totale. De plus, elle offre un champ de

vision large de 160°, ce qui est particulièrement utile pour capturer l'ensemble de la bouteille sans avoir besoin de déplacer la caméra [W5].



Figure 0-5 : Camera 5MP OV5647 [W5].

1.18.7 Imprimante thermique JP QR204 (58mm, intégrée)

L'imprimante utilisée dans notre système est une imprimante thermique compacte, adaptée pour l'impression de reçus ou de tickets. Elle fonctionne selon la méthode thermique directe, ce qui signifie qu'elle n'utilise ni encre ni ruban, mais imprime directement sur du papier thermique sensible à la chaleur [W6]. Ce type de technologie est particulièrement adapté aux systèmes embarqués et aux applications embarquées grâce à sa rapidité et sa fiabilité.

Les principales caractéristiques techniques de cette imprimante sont les suivantes :

- Résolution : 8 points par mm, soit 384 points par ligne
- Largeur d'impression effective : 48 mm
- Jeux de caractères pris en charge : ASCII, GB2312-80 (chinois)
- Polices d'impression :
 - ANK : 5 × 7
 - Caractères chinois : 12 × 24, 24 × 24
- Type de papier : papier thermique
- Largeur du papier : 57,5 ± 0,5 mm

- Diamètre du rouleau : jusqu'à 30 mm
- Interface de communication : série (RS-232 ou TTL) ou parallèle
- Alimentation : DC 5V à 9V
- Température de fonctionnement : de 5 °C à 50 °C
- Vitesse d'impression : entre 50 et 85 mm/s
- Durabilité : durée de vie moyenne de 500 millions de lignes (MCBF)
- Dimensions : 8,20 × 5,80 × 4,40 cm
- Poids : environ 179 g

Grâce à sa petite taille, sa faible consommation énergétique et sa compatibilité avec les microcontrôleurs via l'interface série, cette imprimante représente un choix idéal pour des applications embarquées comme les systèmes de tri intelligent ou de gestion automatisée de tickets.



Figure 0-6 : Imprimante thermique JP QR204 [W6].

1.18.8 Alimentation Double Canal 5V/6A – 24V/5A

L'alimentation double canal 5V/6A et 24V/5A est un dispositif qui fournit deux sources d'alimentation avec des tensions différentes pour alimenter des systèmes nécessitant des tensions et des courants distincts. Dans ce système, un canal fournit 5 volts avec un courant de 6 ampères, tandis que l'autre canal fournit 24 volts avec un courant de 5 ampères. Ce type d'alimentation est couramment utilisé dans les applications électroniques qui nécessitent des sources d'énergie multiples avec différentes tensions, telles que les circuits numériques (qui utilisent généralement 5V) et les systèmes nécessitant des tensions plus élevées, comme les

moteurs ou les équipements industriels, qui fonctionnent souvent à 24V. Ce système permet une alimentation simultanée avec différentes tensions, offrant ainsi une meilleure flexibilité et stabilité pour les systèmes complexes sans avoir besoin de plusieurs sources d'alimentation distinctes.



Figure 0-7 : Alimentation Double Canal (5V/6A et 24V/5A)

1.18.9 Module Capteur D'obstacle Infrarouge KY-032

Le module capteur d'obstacle infrarouge KY-032 est un capteur électronique conçu pour détecter la présence d'un objet à proximité à l'aide de la lumière infrarouge. Il émet un signal infrarouge à travers une LED émettrice, et détecte la réflexion de ce signal lorsqu'un objet se trouve devant le capteur. Lorsqu'un obstacle est détecté dans une certaine plage (habituellement entre 2 et 40 cm), le module génère un signal de sortie qui peut être lu par un microcontrôleur comme l'Arduino [W8].

Dans notre projet, le capteur infrarouge KY-032 est utilisé pour détecter l'entrée des bouteilles et pour indiquer quand la machine est pleine. Il joue ainsi un rôle essentiel dans l'automatisation du système

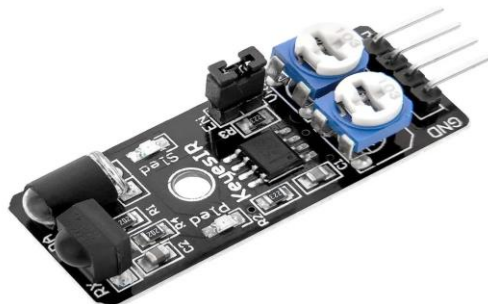


Figure 0-8 : IR Obstacle Avoidance Sensor Module [W8].

1.18.10 Breadboard 830 points

La MB102 Breadboard est une plaque de prototypage sans soudure très utilisée dans les projets électroniques. Elle permet de connecter facilement différents composants électroniques sans avoir à souder, ce qui facilite le développement, les tests et les modifications rapides du circuit.

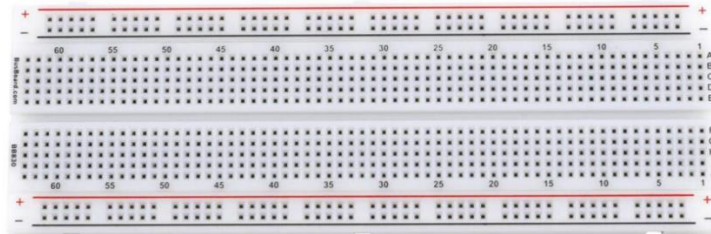


Figure 0-9: Breadboard 830 points

1.18.11 Flashing LED 5mm, Red, Green

La LED clignotante 5mm est une diode électroluminescente spéciale qui émet une lumière colorée (rouge, bleue ou verte) tout en clignotant automatiquement, sans besoin de circuit externe ou de signal de commande.



Figure 0-10 : LED clignotante

1.18.12 Push Button Switch

Le bouton-poussoir 6×6×4.3 mm est un petit interrupteur momentané utilisé dans les circuits électroniques pour déclencher une action spécifique lorsqu'il est pressé. Il revient automatiquement à son état initial après relâchement.



Figure 0-11 : Bouton-poussoir

1.18.13 Solderless Jumper Breadboard Wires M/M F/F M/F

Les fils de connexion jumper M/M F/F M/F sont des câbles souples utilisés pour établir des connexions temporaires entre différents modules électroniques, tels que les microcontrôleurs, capteurs, moteurs, breadboards, et autres composants.

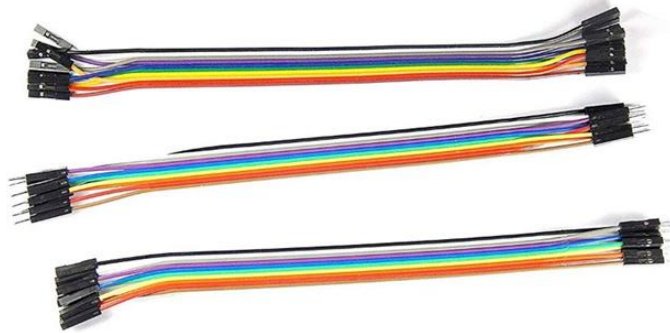


Figure 0-12: Les fils de connexion

1.19 Environnement de Développement Logiciel et Communication dans le Système Arduino

1.19.1 Arduino IDE

Arduino IDE est un environnement de développement intégré utilisé pour écrire et téléverser des codes sur les cartes Arduino. Cet environnement est caractérisé par sa facilité d'utilisation, ce qui le rend adapté aux débutants comme aux experts. Arduino IDE utilise le langage de programmation C/C++ et prend en charge un large éventail de cartes Arduino telles que Arduino Uno, Mega, et Nano. Il propose également des outils et des fonctionnalités pour simplifier la programmation et le test, y compris des bibliothèques intégrées permettant l'interaction avec divers composants tels que des moteurs, des capteurs, et des écrans [W14].

1.19.2 Les bibliothèques utilisées dans Arduino IDE

Dans le développement de notre système embarqué basé sur la carte Arduino, plusieurs bibliothèques logicielles ont été intégrées afin de faciliter la programmation et d'assurer une gestion efficace des composants électroniques. Ces bibliothèques offrent des fonctions prédéfinies permettant d'interagir avec les capteurs, les moteurs, les écrans, et d'autres périphériques, tout en réduisant considérablement la complexité du code.

1.19.2.1 Bibliothèque Servo

La bibliothèque Servo est utilisée pour contrôler les servomoteurs via des signaux PWM (modulation de largeur d'impulsion). Elle permet de positionner précisément le moteur à des angles allant généralement de 0 à 180 degrés. Cette fonctionnalité est essentielle dans les applications nécessitant un mouvement mécanique précis comme les bras robotiques ou les systèmes de tri. Grâce à ses fonctions simples, la bibliothèque Servo facilite la gestion du mouvement sans configuration complexe [W15].

1.19.2.2 Bibliothèque Adafruit_Nokia5110

Cette bibliothèque est conçue pour piloter l'écran LCD Nokia 5110. Elle permet d'afficher des textes, des icônes ou des graphiques simples sur l'écran, offrant ainsi une interface utilisateur visuelle dans le système. La communication avec cet écran se fait via le protocole SPI, et la bibliothèque simplifie grandement l'affichage sans nécessiter une gestion manuelle des signaux [W16].

1.19.2.3 Bibliothèque SoftwareSerial

La bibliothèque SoftwareSerial permet de créer des ports série supplémentaires sur les broches numériques de l'Arduino, en plus du port série matériel déjà présent. Cela est particulièrement utile lorsque plusieurs dispositifs doivent communiquer via UART (comme les modules Bluetooth ou GPS), surtout sur des cartes limitées comme l'Arduino MEGA. Elle offre une grande flexibilité pour les projets nécessitant plusieurs communications série simultanées [W17].

1.19.2.4 Bibliothèque DHT

La bibliothèque DHT est dédiée à la lecture de données provenant des capteurs de température et d'humidité comme les DHT11 ou DHT22. Elle fournit une interface simple pour récupérer ces informations environnementales avec précision. Utilisée fréquemment dans les projets de surveillance climatique ou les systèmes domotiques, elle permet d'intégrer des données ambiantes sans calculs complexes [W18].

1.19.2.5 Communication série (Serial Communication) :

Le moniteur série (Serial Monitor) est un outil clé dans Arduino IDE, permettant aux utilisateurs de surveiller les données envoyées depuis ou vers la carte Arduino. La

communication série est utilisée pour échanger des données entre Arduino et un ordinateur ou d'autres appareils, facilitant ainsi le débogage, la lecture des valeurs des capteurs, et l'envoi de commandes pour contrôler des appareils [W18].

1.19.3 Langage de Programmation Python

Python est un langage de programmation de haut niveau, interprété, orienté objet et multi-paradigme, conçu pour être simple à lire et facile à écrire. Il a été créé à la fin des années 1980 par Guido van Rossum et est devenu aujourd'hui l'un des langages les plus utilisés dans le monde, notamment dans les domaines de l'intelligence artificielle, du développement web, de l'automatisation, de la science des données et de l'informatique embarquée.

L'un des plus grands avantages de Python réside dans sa syntaxe claire et concise, qui permet aux développeurs d'écrire du code lisible et facilement maintenable. De plus, Python dispose d'un énorme écosystème de bibliothèques et de modules (comme NumPy, OpenCV, TensorFlow, Pandas...), qui rendent le développement plus rapide et plus puissant [W20].

Python est un langage multi-plateforme, ce qui signifie qu'il peut être exécuté sur différents systèmes d'exploitation (Windows, Linux, macOS, etc.) sans modification majeure du code. Il est également interprété, ce qui permet de tester et exécuter le code ligne par ligne sans compilation, rendant le débogage plus facile.

Enfin, sa communauté active et sa vaste documentation en font un excellent choix aussi bien pour les débutants que pour les experts. Grâce à toutes ces caractéristiques, Python est devenu un outil incontournable dans le développement de systèmes intelligents, notamment sur les plateformes embarquées comme le Raspberry Pi [W21].

1.19.4 Définitions détaillées des bibliothèques Python utilisées

1.19.4.1 Bibliothèque Os

La bibliothèque `os` est intégrée à Python et permet d'interagir avec le système d'exploitation. Elle offre des fonctions pour manipuler les fichiers, gérer les répertoires, accéder aux variables d'environnement, et exécuter des commandes du système. Elle est essentielle pour automatiser les tâches liées aux fichiers [W22].

1.19.4.2 Shutil

shutil fournit des fonctions avancées pour copier, déplacer, supprimer ou compresser des fichiers et dossiers. Elle est souvent utilisée pour organiser les jeux de données et préparer les fichiers avant le traitement ou l'entraînement [W23].

1.19.4.3 Numpy

NumPy est une bibliothèque fondamentale pour le calcul scientifique en Python. Elle permet la manipulation efficace de tableaux multidimensionnels (arrays) et offre de nombreuses fonctions mathématiques et statistiques. C'est une base indispensable en intelligence artificielle et en traitement de données [W24].

1.19.4.4 matplotlib.pyplot

Composant de la bibliothèque Matplotlib, pyplot est utilisée pour générer des visualisations graphiques (courbes, histogrammes, diagrammes). Elle permet de mieux comprendre les résultats par la visualisation des données et des performances des modèles [W25].

1.19.4.5 Tensorflow

TensorFlow est une bibliothèque open source développée par Google, dédiée à la création et l'entraînement de modèles d'apprentissage automatique et d'apprentissage profond. Elle prend en charge l'accélération GPU et permet de construire des réseaux neuronaux complexes de manière modulaire [W26].

1.19.4.6 Image Data Generator

Cet outil, inclus dans Keras via TensorFlow, permet de générer automatiquement des images modifiées à partir du jeu d'entraînement, en appliquant des transformations comme la rotation, le zoom, le renversement, etc. Cela permet d'augmenter la diversité des données d'entraînement et de réduire l'overfitting [W27].

1.19.4.7 train_test_split

Fonction de la bibliothèque scikit-learn, elle permet de diviser aléatoirement un jeu de données en sous-ensembles d'entraînement, de validation et de test. Cela garantit une évaluation fiable du modèle [W28].

1.20 Intégration du système et connexion des composants

Après avoir identifié l'ensemble des composants matériels et logiciels nécessaires au fonctionnement du système, la phase suivante consiste à les intégrer de manière cohérente afin d'assurer une communication fluide et une exécution harmonieuse des tâches. Cette étape vise à connecter les différents éléments — tels que la caméra, les capteurs, le Raspberry Pi, l'Arduino, l'écran d'affichage, le moteur et l'imprimante — en un système unifié. Chaque composant joue un rôle spécifique et leur intégration correcte garantit la fiabilité, la performance et la stabilité du dispositif global.

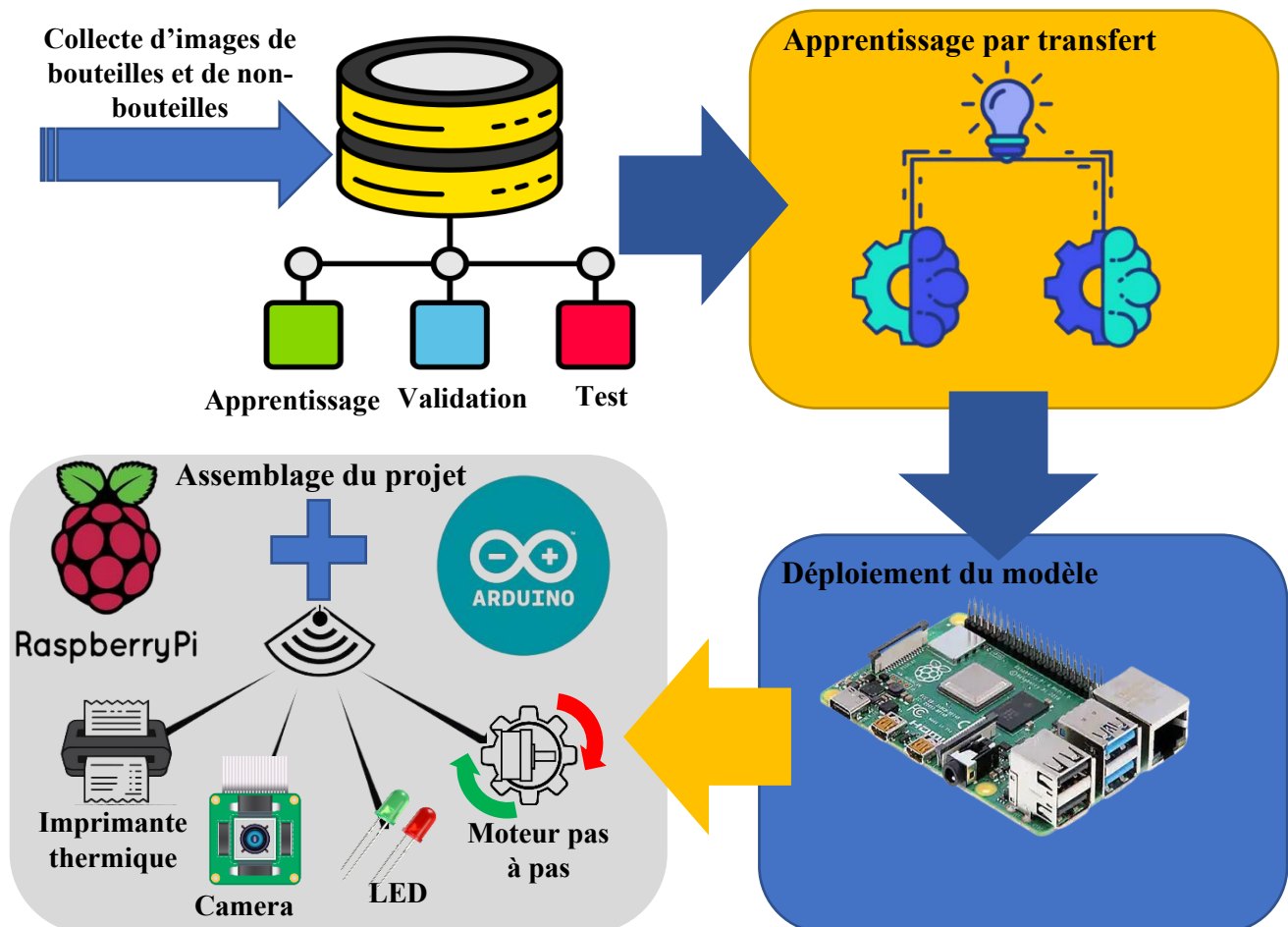


Figure 0-13: L'architecture générale pour le développement du projet

1.20.1 Principe de fonctionnement du système embarqué

Le système intelligent proposé repose sur une intégration harmonieuse de plusieurs composants électroniques, chacun jouant un rôle précis et complémentaire dans le processus de détection, de classification et de gestion automatique des bouteilles.

Au centre de ce système se trouve le Raspberry Pi, qui agit comme le cerveau principal du dispositif. Il héberge le modèle d'intelligence artificielle pré-entraîné, conçu pour reconnaître les bouteilles plastiques acceptables et non acceptables à partir d'images capturées en temps réel.

Le script Python est configuré pour s'exécuter automatiquement au démarrage du Raspberry Pi via un fichier crontab, assurant ainsi un fonctionnement autonome dès la mise sous tension.

Le Raspberry Pi est connecté à une caméra Pi via un câble CSI (Camera Serial Interface), assurant une transmission rapide et une qualité d'image optimale. Un capteur infrarouge (IR) est également installé à l'entrée du système, servant à détecter la présence d'un objet (bouteille) devant la caméra.

Lorsque le capteur IR détecte une bouteille, il envoie un signal au Raspberry Pi. En réponse, la caméra capture une image de la bouteille, qui est ensuite transmise **au** modèle de classification. Ce dernier analyse l'image et prend une décision sur l'acceptabilité de la bouteille.

Le résultat de cette classification est ensuite envoyé via une connexion série (Serial) à la carte Arduino, qui exécute l'action appropriée. Si la bouteille est acceptée, le moteur pas à pas (Stepper Motor) oriente la bouteille vers le bac de collecte. En revanche, si elle est rejetée, elle est dirigée vers une sortie différente.

De plus, un bouton-poussoir permet à l'utilisateur d'imprimer un ticket via une imprimante thermique, contenant des informations sur le nombre de bouteilles acceptées. Ces données sont également sauvegardées dans un fichier journal pour un suivi ultérieur. Elles sont aussi affichées en temps réel sur un écran (par exemple, un écran Nokia 5110). Le système est aussi capable d'alerter l'utilisateur lorsque le dispositif est plein, en affichant un message de saturation sur l'écran.

Par ailleurs, deux voyants lumineux (LED verte et LED rouge) sont intégrés au système afin de fournir un retour visuel instantané à l'utilisateur. Lorsque la bouteille est reconnue

comme acceptable, la LED **verte** s'allume brièvement pour indiquer que l'objet a été correctement classé et redirigé vers le bac de collecte. En revanche, si la bouteille est refusée, la LED rouge s'allume pour signaler que l'objet ne répond pas aux critères requis.

Ce retour visuel simple mais efficace permet à l'utilisateur de comprendre immédiatement la décision du système sans avoir à consulter l'écran, renforçant ainsi l'ergonomie et l'interactivité du dispositif.

Cette interaction fluide entre les logiciels et les composants électroniques a permis de réaliser un système autonome, intelligent et réactif, capable de prendre des décisions en temps réel tout en assurant une interface conviviale avec l'utilisateur.

1.21 Assemblage du système et intégration des composants

Dans le cadre de l'architecture du système, une liaison directe a été établie entre le Raspberry Pi 4 et l'Arduino MEGA à l'aide d'un simple câble USB, sans recours à des connexions électriques supplémentaires.

La communication série (Serial Communication) a été choisie comme protocole d'échange entre les deux unités. Le Raspberry Pi agit comme unité centrale de traitement, responsable de l'analyse d'image et de l'exécution du modèle d'intelligence artificielle, tandis que l'Arduino MEGA assure le contrôle physique des composants en fonction des résultats reçus. La communication série a été configurée sur le Raspberry Pi via le port(USB), avec une vitesse de transmission fixée à 9600 bauds, garantissant un échange fluide et fiable des données.

1.21.1 Raspberry et le capteur d'obstacle

Un capteur infrarouge (IR) a été utilisé pour détecter la présence d'une bouteille à l'entrée du système. Ce capteur comporte trois broches :

- **VCC** connecté au port 3.3V du Raspberry Pi
- **GND** connecté au GND du Raspberry Pi
- **OUT** connecté au port GPIO 17 du Raspberry Pi

Lorsque la bouteille passe devant le capteur, celui-ci génère un signal numérique sur le port GPIO 17, déclenchant ainsi la capture de l'image et l'exécution du modèle d'intelligence artificielle.

1.21.2 Raspberry et le Camera

La caméra Raspberry Pi est directement connectée au port CSI du Raspberry Pi, permettant une transmission rapide des données sans latence. La caméra est ensuite configurée et utilisée dans le système via la bibliothèque picamera de Python, qui offre une interface simple pour contrôler la caméra, comme la capture d'images et leur affichage direct dans le système.

1.21.3 Arduino et le driver

L'Arduino Mega est relié au driver microstep pour contrôler le moteur NEMA 17 utilisé dans le système, Les connexions du driver sont les suivantes :

- ENA est reliée au port Pin 8 de l'Arduino pour activer ou désactiver le moteur.
- DIR est relié au port Pin 2 de l'Arduino pour contrôler la direction du moteur.
- PUL est relié au port Pin 5 de l'Arduino pour envoyer des impulsions de contrôle pour le mouvement du moteur.
- GND de Arduino est relié à GND du driver microstep, assurant une référence de masse commune entre les deux appareils.

Avec ces connexions, des signaux sont envoyés via PUL pour contrôler le mouvement précis du moteur, tandis que DIR détermine la direction. Le moteur est activé via ENA en fonction des résultats de classification de Raspberry Pi. Ces connexions permettent un contrôle précis du moteur en fonction de la décision du système concernant l'acceptation ou le rejet de la bouteille.

1.21.4 Driver et le moteur

Le moteur est connecté au driver de la manière suivante

- A+ et A- ainsi que B+ et B- du driver sont reliés aux bornes correspondantes du moteur NEMA 17 pour contrôler son mouvement.
- Les impulsions (PUL) font avancer le moteur pas à pas avec une grande précision, tandis que la direction du mouvement est déterminée par DIR. Le moteur est activé ou désactivé via ENA, ce qui permet de l'activer lorsque la bouteille est acceptée. Ces connexions permettent un contrôle précis du moteur en fonction des décisions reçues de la Raspberry Pi.

1.21.5 Arduino et l'imprimante

L'Arduino Mega est relié à l'imprimante thermique dans le système pour imprimer les résultats concernant les bouteilles acceptées ou rejetées.

Les connexions sont les suivantes :

- TX de l'imprimante thermique est relié au port Pin 11 de l'Arduino pour envoyer des données de l'Arduino vers l'imprimante.
- RX de l'imprimante thermique est relié au port Pin 12 de l'Arduino pour recevoir des données de l'imprimante.
- GND de l'imprimante thermique est reliée au GND de l'Arduino pour assurer une référence de masse commune.

1.21.6 L'alimentation et L'imprimante et le driver

Une alimentation externe a été utilisée pour fournir l'énergie nécessaire au fonctionnement du moteur pas à pas (NEMA 17) et de l'imprimante thermique. Le moteur a été connecté à la sortie 12V de l'alimentation, tandis que l'imprimante a été reliée à la sortie 5V. Les bornes GND des deux composants ont été connectées au GND de l'alimentation pour assurer une référence de masse commune.

1.21.7 Arduino et LED

Deux LEDs, une verte et un rouge, ont été connectées à l'Arduino pour indiquer l'état du système : la LED verte est reliée à la broche numérique D6 et la LED rouge à la broche D7, avec des résistances de limitation de courant pour protéger les composants.

1.22 Méthodologie

1.22.1 Base de données et préparation

Afin de développer un modèle de classification fiable pour la détection des bouteilles plastiques acceptables et non acceptables, nous avons initialement utilisé une base de données disponible sur Kaggle. Cependant, cette dernière présentait un déséquilibre considérable entre les classes : environ 40 000 images de bouteilles acceptables contre seulement 7 000 images de bouteilles non acceptables.

Pour remédier à ce déséquilibre et éviter tout biais d'apprentissage, nous avons collecté manuellement des images supplémentaires de bouteilles non acceptables à partir d'autres bases de données publiques. Cela nous a permis d'atteindre un ensemble global équilibré de plus de 70 000 images.

La base finale a été divisée comme suit :

- Ensemble d'entraînement (70 %) : 52 711 images
 - ✓ Bouteilles acceptables : 31 688
 - ✓ Bouteilles non acceptables : 21 023
- Ensemble de validation (15 %) : 11 296 images
 - ✓ Bouteilles acceptables : 6 791
 - ✓ Bouteilles non acceptables : 4 505
- Ensemble de test (15 %) : 11 297 images
 - ✓ Bouteilles acceptables : 6 791
 - ✓ Bouteilles non acceptables : 4 506

Ce processus de construction de base de données équilibrée constitue une étape cruciale pour assurer la performance et la robustesse du modèle d'apprentissage automatique.

Toutes les images ont été redimensionnées à une taille uniforme de 224×224 pixels afin d'assurer une cohérence dans l'entrée du modèle. Ensuite, une normalisation a été appliquée en divisant les valeurs des pixels par 255 pour les ramener dans l'intervalle [0, 1], ce qui facilite l'apprentissage du réseau de neurones.

Ce prétraitement a été réalisé à l'aide de la classe ImageDataGenerator de Keras, qui a permis de générer les lots d'images à la volée durant l'entraînement, la validation et le test.

1.22.2 Choix du modèle et configuration de l'architecture

Le modèle MobileNetV2 a été choisi comme modèle de base pour ce projet en raison de son efficacité élevée et de sa capacité à fonctionner de manière optimale dans des environnements informatiques limités tels que le Raspberry Pi. MobileNetV2 est un réseau neuronal convolutif léger et puissant, préalablement entraîné sur ImageNet. Nous avons utilisé ce modèle sans les couches supérieures, ce qui permet de personnaliser la tête du modèle pour répondre aux besoins d'une tâche de classification binaire entre les classes "Bouteille acceptée"

et "Bouteille non acceptée". Les couches de base du modèle ont été figées pour conserver les poids pré-entraînés sur ImageNet, ce qui accélère l'entraînement et améliore les performances même avec un ensemble de données limité. L'architecture finale du modèle comprend une couche de Global Average Pooling, suivie d'une couche dense Dense de 128 unités avec activation ReLU, une couche Dropout de 30% pour réduire le risque de sur-apprentissage (overfitting), et enfin une couche de sortie avec une seule unité et activation sigmoid. Le modèle a été compilé avec l'optimiseur Adam avec un taux d'apprentissage de 0.0001, et la fonction de perte utilisée est la `binary_crossentropy`, adaptée à une tâche de classification binaire. Les métriques utilisées pour évaluer la performance du modèle incluent la précision, la précision positive (precision) et le rappel (recall). La taille des images en entrée était de 224x224 pixels, une taille optimale pour le modèle MobileNetV2 afin d'obtenir une bonne performance tout en optimisant l'utilisation des ressources. La taille de lot choisie était de 32 images par étape d'entraînement, et le modèle a été entraîné pendant 6 époques, avec l'application d'un EarlyStopping pour arrêter l'entraînement si la perte de validation (`val_loss`) ne s'améliorait pas, tout en sauvegardant le meilleur modèle avec ModelCheckpoint en fonction de ses performances sur l'ensemble de validation.

1.23 Résultat et Discussion

Dans cette section, nous présentons les résultats obtenus à partir de l'entraînement du modèle sur notre ensemble de données. Nous avons évalué la performance du modèle en utilisant plusieurs métriques clés : Précision (Accuracy), Précision positive (Precision), et Rappel (Recall). Ces métriques ont été calculées sur l'ensemble de test après l'entraînement, en utilisant le meilleur modèle sauvegardé via l'utilisation du ModelCheckpoint.

Les résultats montrent une très bonne performance du modèle, avec une précision atteignant 99.98%, une précision positive de 99.98%, et un rappel également de 99.98%. Ces résultats indiquent que le modèle a bien appris à distinguer les bouteilles des objets non-bouteilles. La performance du modèle est particulièrement remarquable en raison de l'équilibre entre les classes, ce qui suggère que le modèle n'a pas été biaisé par une classe dominante.

	Prédiction positif	Prédiction négatif
Actuel positif	6790	1
Actuel négatif	1	4503

Table 0-1 matrice de confision

En analysant la matrice de confusion obtenue, nous pouvons voir que le modèle a correctement classifié presque toutes les images, avec seulement une petite quantité d'erreurs de classification. Il y a une prédiction erronée où une bouteille a été classée comme non-bouteille, et une autre erreur inverse où un objet non-bouteille a été classé comme une bouteille. Cependant, le nombre de ces erreurs est extrêmement faible par rapport au nombre total d'images testées, ce qui montre l'efficacité du modèle dans un scénario réel.

En termes de courbes d'apprentissage, la précision a continué d'augmenter à chaque époque, ce qui montre une bonne capacité de généralisation du modèle. De même, la perte a diminué au fil des époques, ce qui est un signe que le modèle a appris efficacement. Ces courbes suggèrent que le modèle a convergé après quelques époques d'entraînement et qu'il a évité le sur-apprentissage grâce à l'usage de Dropout et de EarlyStopping.

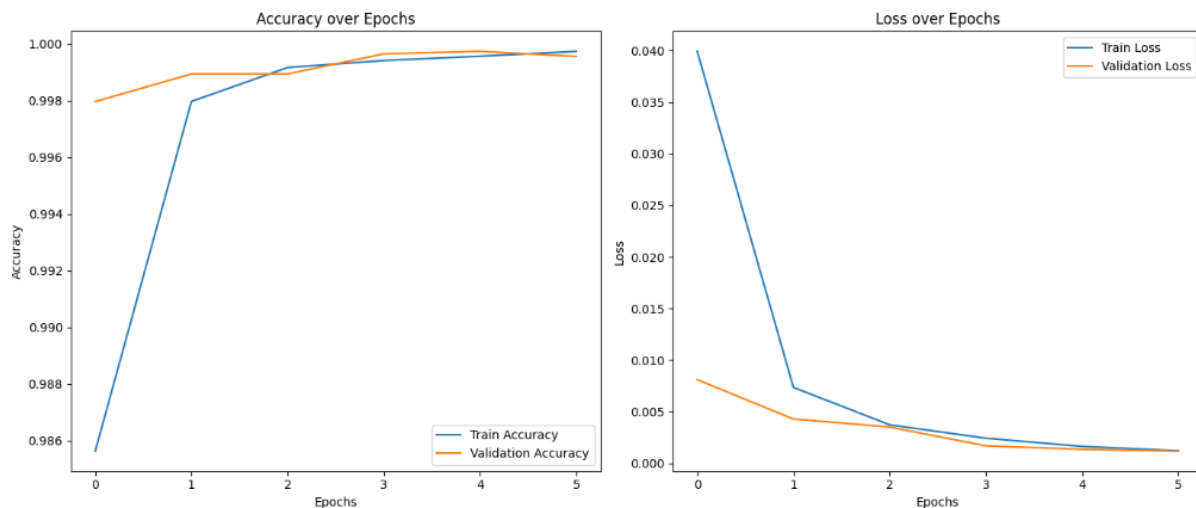


Figure 0-14 : Train and Validation Accuracy and Loss

1.24 Analyser des performances typiques du système

Les résultats obtenus lors de la phase d'évaluation confirment l'efficacité du modèle dans la classification des images selon les deux catégories ciblées. Cette performance élevée peut être expliquée par la disponibilité d'une base de données volumineuse, équilibrée et constituée d'images de bonne qualité et de grande diversité en termes d'angles, d'éclairage et d'arrière-plans, ce qui a favorisé un apprentissage optimal du modèle. Toutefois, certaines difficultés ont été rencontrées, notamment le problème de surapprentissage (Overfitting), ce qui nous a conduits à ajuster plusieurs hyperparamètres tels que la taille du lot (Batch Size), le nombre d'époques (Epochs), et à introduire une couche de régularisation (Dropout). Ces ajustements

ont permis de stabiliser progressivement les performances. Malgré ces résultats satisfaisants dans un environnement de test contrôlé, des limitations ont été observées lors de l'application du système en conditions réelles, notamment en présence d'images capturées par des caméras de faible qualité ou dans des environnements visuels différents de ceux utilisés durant l'entraînement. Cela souligne l'importance d'améliorer la capacité de généralisation du modèle et de garantir une performance robuste dans des contextes variés.

1.25 Conclusion

À travers ce chapitre, nous avons présenté et analysé les résultats obtenus par le modèle proposé, qui a démontré une performance élevée dans la classification des bouteilles, tant en termes de précision que de capacité à distinguer entre les deux classes ciblées. Nous avons également abordé les principales difficultés rencontrées, notamment le surapprentissage (Overfitting), que nous avons réussi à atténuer en ajustant le nombre d'époques, la taille des lots (batch size), et en intégrant une couche de régularisation (Dropout). Bien que les résultats aient été très satisfaisants dans un environnement contrôlé, les tests sur le terrain ont révélé que les performances du modèle peuvent être affectées par des conditions d'acquisition d'image variables, telles que la mauvaise qualité des caméras ou la complexité de l'arrière-plan. Ces constats ouvrent la voie à des améliorations futures pour renforcer la robustesse et la généralisation du système dans des contextes réels. Ce chapitre constitue ainsi une base solide pour envisager des évolutions plus poussées du modèle.

Conclusion générale

À l'issue de ces travaux, nous avons pu concevoir et développer un système intelligent capable d'identifier automatiquement les bouteilles en plastique grâce à un modèle d'apprentissage profond entraîné sur diverses images. Les résultats ont démontré que le système est capable d'offrir d'excellentes performances dans des conditions spécifiques, démontrant l'efficacité de la combinaison de l'IA et des systèmes embarqués pour résoudre des problèmes concrets tels que le tri automatique des déchets.

Malgré notre succès, le système présente des défis. Nous avons observé que les performances du modèle peuvent être affectées par des changements de conditions d'éclairage ou d'arrière-plan de l'image, et que la qualité de la caméra utilisée joue également un rôle clé dans la précision des résultats. Tous ces facteurs démontrent que le modèle nécessite des adaptations supplémentaires pour fonctionner efficacement dans des environnements réels et complexes.

Sur la base de ce qui précède, nous proposons d'étendre la base de données afin d'inclure différents types de déchets et des conditions d'imagerie plus diversifiées à l'avenir, améliorant ainsi la capacité de généralisation du modèle. L'architecture physique du système pourrait également être améliorée, en termes de type de caméra, de vitesse de traitement et de consommation d'énergie. De plus, le système pourrait être développé pour inclure des fonctions supplémentaires, telles que le tri d'autres déchets (canettes métalliques ou papier), ou intégré à des solutions industrielles plus larges. Ce projet représente une première étape vers une utilisation efficace de l'intelligence artificielle au service de l'environnement et ouvre de nouvelles perspectives pour de multiples applications dans le domaine du recyclage intelligent.

Références bibliographiques

- [1] M. M. Taye, « Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions », *Computation*, vol. 11, n° 3, p. 52, mars 2023, doi: 10.3390/computation11030052.
- [2] M. G. M. Abdolrasol *et al.*, « Artificial Neural Networks Based Optimization Techniques: A Review », *Electronics*, vol. 10, n° 21, p. 2689, nov. 2021, doi: 10.3390/electronics10212689.
- [3] T. Senjyu, P. N. Mahalle, T. Perumal, et A. Joshi, Éd., *Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2020, Volume 1*, vol. 195. in Smart Innovation, Systems and Technologies, vol. 195. Singapore: Springer Singapore, 2021. doi: 10.1007/978-981-15-7078-0.
- [4] P. Borne, M. Benrejeb, et J. Haggège, *Les réseaux de neurones: présentation et applications*. in Méthodes et pratiques de l'ingénieur, no. 15. Paris: Éd. Technip, 2007.
- [5] Roheen Qamar et Baqar Ali Zardari, « Artificial Neural Networks: An Overview », *Mesopotamian J. Comput. Sci.*, vol. 2023, p. 124-133, août 2023, doi: 10.58496/MJCSC/2023/015.
- [6] S. R. Dubey, S. K. Singh, et B. B. Chaudhuri, « Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark », 28 juin 2022, *arXiv*: arXiv:2109.14545. doi: 10.48550/arXiv.2109.14545.
- [7] S. Sharma, S. Sharma, et A. Athaiya, « ACTIVATION FUNCTIONS IN NEURAL NETWORKS », *Int. J. Eng. Appl. Sci. Technol.*, vol. 04, n° 12, p. 310-316, mai 2020, doi: 10.33564/IJEAST.2020.v04i12.054.
- [8] S. Siddharth, « ACTIVATION FUNCTIONS IN NEURAL NETWORKS », *International Journal of Engineering Applied Sciences and Technology*, IJEAST, p. 310-316, avril 2020.
- [9] A. Apicella, F. Donnarumma, F. Isgrò, et R. Prevete, « A survey on modern trainable activation functions », *Neural Netw.*, vol. 138, p. 14-32, juin 2021, doi: 10.1016/j.neunet.2021.01.026.
- [10] Q. Wang, Y. Ma, K. Zhao, et Y. Tian, « A Comprehensive Survey of Loss Functions in Machine Learning », *Ann. Data Sci.*, vol. 9, n° 2, p. 187-212, avr. 2022, doi: 10.1007/s40745-020-00253-5.
- [11] A. Jadon, A. Patil, et S. Jadon, « A Comprehensive Survey of Regression Based Loss Functions for Time Series Forecasting », 5 novembre 2022, *arXiv*: arXiv:2211.02989. doi: 10.48550/arXiv.2211.02989.
- [12] K. Blekos *et al.*, « A Review on Quantum Approximate Optimization Algorithm and its Variants », *Phys. Rep.*, vol. 1068, p. 1-66, juin 2024, doi: 10.1016/j.physrep.2024.03.002.
- [13] J. Zhang, « Gradient Descent based Optimization Algorithms for Deep Learning Models Training », *Information Fusion and Mining Laboratory*, p. 24, mars 2019.
- [14] Y. Tian, Y. Zhang, et H. Zhang, « Recent Advances in Stochastic Gradient Descent in Deep Learning », *Mathematics*, vol. 11, n° 3, p. 682, janv. 2023, doi: 10.3390/math11030682.
- [15] D. Z. Matthew, « ADADELTA: An Adaptive Learning Rate Method », New York University, USA.

- [16] L. Chen, S. Li, Q. Bai, J. Yang, S. Jiang, et Y. Miao, « Review of Image Classification Algorithms Based on Convolutional Neural Networks », *Remote Sens.*, vol. 13, n° 22, p. 4712, nov. 2021, doi: 10.3390/rs13224712.
- [17] L. Li, S. Zhang, et B. Wang, « Plant Disease Detection and Classification by Deep Learning—A Review », *IEEE Access*, vol. 9, p. 56683-56698, 2021, doi: 10.1109/ACCESS.2021.3069646.
- [18] Z. Li, F. Liu, W. Yang, S. Peng, et J. Zhou, « A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects », *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, n° 12, p. 6999-7019, déc. 2022, doi: 10.1109/TNNLS.2021.3084827.
- [19] K. O'Shea et R. Nash, « An Introduction to Convolutional Neural Networks », 2 décembre 2015, *arXiv*: arXiv:1511.08458. doi: 10.48550/arXiv.1511.08458.
- [20] K. Simonyan et A. Zisserman, « Very Deep Convolutional Networks for Large-Scale Image Recognition », 2014, *arXiv*. doi: 10.48550/ARXIV.1409.1556.
- [21] C. Szegedy *et al.*, « Going Deeper with Convolutions », 17 septembre 2014, *arXiv*: arXiv:1409.4842. doi: 10.48550/arXiv.1409.4842.
- [22] K. He, X. Zhang, S. Ren, et J. Sun, « Deep Residual Learning for Image Recognition », 10 décembre 2015, *arXiv*: arXiv:1512.03385. doi: 10.48550/arXiv.1512.03385.
- [23] M. Tan et Q. V. Le, « EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks », 11 septembre 2020, *arXiv*: arXiv:1905.11946. doi: 10.48550/arXiv.1905.11946.
- [24] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, et C. Liu, « A Survey on Deep Transfer Learning », 6 août 2018, *arXiv*: arXiv:1808.01974. doi: 10.48550/arXiv.1808.01974.
- [25] S. Chatterjee, D. Hazra, Y.-C. Byun, et Y.-W. Kim, « Enhancement of Image Classification Using Transfer Learning and GAN-Based Synthetic Data Augmentation », *Mathematics*, vol. 10, n° 9, p. 1541, mai 2022, doi: 10.3390/math10091541.
- [26] S. Liubartseva, G. Coppini, et R. Lecci, « Are Mediterranean Marine Protected Areas sheltered from plastic pollution? », *Mar. Pollut. Bull.*, vol. 140, p. 579-587, mars 2019, doi: 10.1016/j.marpolbul.2019.01.022.
- [27] R. Geyer, J. R. Jambeck, et K. L. Law, « Production, use, and fate of all plastics ever made », *Sci. Adv.*, vol. 3, n° 7, p. e1700782, juill. 2017, doi: 10.1126/sciadv.1700782.
- [28] A. L. Andrady, « Microplastics in the marine environment », *Mar. Pollut. Bull.*, vol. 62, n° 8, p. 1596-1605, août 2011, doi: 10.1016/j.marpolbul.2011.05.030.
- [29] M. Cole, P. Lindeque, C. Halsband, et T. S. Galloway, « Microplastics as contaminants in the marine environment: A review », *Mar. Pollut. Bull.*, vol. 62, n° 12, p. 2588-2597, déc. 2011, doi: 10.1016/j.marpolbul.2011.09.025.
- [30] A. Diggle et T. R. Walker, « Environmental and Economic Impacts of Mismanaged Plastics and Measures for Mitigation », *Environments*, vol. 9, n° 2, p. 15, janv. 2022, doi: 10.3390/environments9020015.
- [31] J. Kirchherr, D. Reike, et M. Hekkert, « Conceptualizing the circular economy: An analysis of 114 definitions », *Resour. Conserv. Recycl.*, vol. 127, p. 221-232, déc. 2017, doi: 10.1016/j.resconrec.2017.09.005.
- [32] R. Geyer, J. R. Jambeck, et K. L. Law, « Production, use, and fate of all plastics ever made », *Sci. Adv.*, vol. 3, n° 7, p. e1700782, juill. 2017, doi: 10.1126/sciadv.1700782.
- [33] D. C. Wilson, C. Velis, et C. Cheeseman, « Role of informal sector recycling in waste management in developing countries », *Habitat Int.*, vol. 30, n° 4, p. 797-808, déc. 2006, doi: 10.1016/j.habitatint.2005.09.005.
- [34] M. Deena, « Reverse Vending Machine for Plastic Bottle Recycling », Kerala - India.

- [35] H. Zia *et al.*, « Plastic Waste Management through the Development of a Low Cost and Light Weight Deep Learning Based Reverse Vending Machine », *Recycling*, vol. 7, n° 5, p. 70, sept. 2022, doi: 10.3390/recycling7050070.
- [36] E. Scavino, D. A. Wahab, A. Hussain, H. Basri, et M. M. Mustafa, « Application of automated image analysis to the identification and extraction of recyclable plastic bottles », *J. Zhejiang Univ.-Sci. A*, vol. 10, n° 6, p. 794-799, juin 2009, doi: 10.1631/jzus.A0820788.
- [37] J. Xiao, J. Xu, C. Tian, P. Han, L. You, et S. Zhang, « A Serial Attention Frame for Multi-Label Waste Bottle Classification », *Appl. Sci.*, vol. 12, n° 3, p. 1742, févr. 2022, doi: 10.3390/app12031742.

[W1] Raspberry Pi [Consulté 3\5\2025], à l'adresse <https://www.raspberrypi.com/>

[W2] Arduino [Consulté 13\5\2025], à l'adresse <https://www.arduino.cc/>

[W3] Stepperonline [Consulté 31\5\2025], à l'adresse <https://www.omc-stepperonline.com/fr/biaxial-nema-17-bipolaire-1-8deg-26ncm-36-8oz-in-0-4a-12v-42x34mm-4-fils-17hs13-0404d>

[W4] Haljia [Consulté 23\5\2025], à l'adresse <https://www.haljia.com/products/haljia-tb6600-4a-stepper-motor-driver-controller-20khz-cnc-single-axis-2-4-phase-hybrid-stepper-motor-driver-board-controller>

[W5] Raspberry Pi [Consulté 3\5\2025], à l'adresse <https://www.raspberrypi.com/documentation/accessories/camera.html>

[W6] Dzduino [Consulté 3\5\2025], à l'adresse <https://www.dzduino.com/jp-qr204-58mm-micro-r%C3%A9ception-embarqu%C3%A9-imprimante-thermique-fr?search=Imprimante%20thermique%20&description=true>

[W7] Dzduino [Consulté 3\5\2025], à l'adresse <https://www.dzduino.com/alimentation-double-canal-5v-6a-24v-5a-pour-bande-de-led-fr?search=Alimentation%20Double>

[W8] [Consulté 3\5\2025], à l'adresse <https://www.joy-it.net/en/products/SEN-KY032IR>

[W9] [Consulté 3\5\2025], à l'adresse https://www.websoog.com/fr/industriel-scientifique/1391-489-20cm-40pin-solderless-jumper-breadboard-wires-mm-ff-mf.html#/494-type-m_m

[W10] Agence Nationale des Déchets [Consulté 3\5\2025], à l'adresse <https://and.dz/>

- [W11] Arduino IDE [Consulté 3\5\2025], à l'adresse
<https://www.arduino.cc/en/software/>
- [W12] Bibliothèque de servomoteurs [Consulté 3\5\2025], à l'adresse
<https://www.arduino.cc/en/Reference/Servo>
- [W13] Bibliothèque de nokia-5110-3310-monochrome-lcd [Consulté 3\5\2025], à l'adresse <https://learn.adafruit.com/nokia-5110-3310-monochrome-lcd>
- [W14] Bibliothèque de softwareSerial [Consulté 3\5\2025], à l'adresse
<https://www.arduino.cc/en/Reference/softwareSerial>
- [W15] Bibliothèque de DHT-sensor-library [Consulté 3\5\2025], à l'adresse
<https://github.com/adafruit/DHT-sensor-library>
- [W16] Bibliothèque serial [Consulté 3\5\2025], à l'adresse
<https://www.arduino.cc/reference/en/language/functions/communication/serial/>
- [W17] Python [Consulté 3\5\2025], à l'adresse
<https://www.python.org/doc/essays/blurb>
- [W18] Raspberry Pi [Consulté 3\5\2025], à l'adresse
<https://www.raspberrypi.org/documentation/usage/python/>
- [W19] Python – OS module [Consulté 3\5\2025], à l'adresse
<https://docs.python.org/3/library/os.html>
- [W20] Shutil module [Consulté 3\5\2025], à l'adresse
<https://docs.python.org/3/library/shutil.html>
- [W21] NumPy – Documentation officielle [Consulté le 01/06/2025], à l'adresse :
<https://numpy.org/doc/>

[W22] TensorFlow – API Documentation [Consulté le 01/06/2025], à l’adresse : https://www.tensorflow.org/api_docs

[W23] Matplotlib.pyplot – Pyplot API [Consulté le 01/06/2025], à l’adresse : https://matplotlib.org/stable/api/pyplot_api.html

[W24] Scikit-learn – train_test_split [Consulté le 01/06/2025], à l’adresse : https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

Annexes

Annexe 1 : Soumission du projet

a. Idée du projet

L'idée de notre projet est née d'une observation de la réalité environnementale que nous vivons actuellement. Les bouteilles en plastique sont devenues omniprésentes dans les rues, sur les plages, et même dans les zones montagneuses. Cette prolifération incontrôlée des déchets plastiques représente un véritable danger pour notre environnement, surtout lorsqu'on la compare aux pays développés qui adoptent des systèmes intelligents et durables pour la gestion des déchets.

Conscients de cette problématique et animés par un sens de responsabilité écologique, nous avons décidé de concevoir un système intelligent capable de trier automatiquement les bouteilles en plastique, dans le but de réduire l'effort humain, gagner du temps, et diminuer les coûts liés aux méthodes de tri traditionnelles.

Ce projet vise à proposer une solution pratique, innovante et durable pour protéger notre environnement et contribuer à une gestion plus intelligente des déchets.

b. Valeur proposée

Valeur	Description
Réduction des efforts et des coûts	Le système permet d'automatiser le tri, réduisant considérablement les efforts humains ainsi que le temps et les coûts nécessaires au tri manuel.
Durabilité environnementale	Le projet contribue à la réduction de la pollution plastique et encourage le recyclage intelligent et moderne.
Autonomisation des porteurs de projets	Le système offre une base technologique exploitable pour initier des projets de recyclage réussis et durables.

Tableau 1 : les valeurs proposées

c. Objectifs du projet

- Contribuer à la réduction de la pollution causée par les bouteilles en plastique.

- Développer un système intelligent capable de reconnaître et trier automatiquement les bouteilles.
- Faciliter le processus de tri et limiter l'intervention humaine.
- Réduire le temps et les coûts liés à la collecte et au tri des déchets.

d. Calendrier du projet

Semaine	1	2	3	4	5	6	7	8	9	10
Étude et analyse du problème	*									
Recherche et sélection des composants du système		*								
Conception théorique et choix de la méthode de travail			*							
Développement et entraînement du modèle d'IA				*	*					
Intégration du modèle intelligent avec le système physique						*	*			
Test du système et correction des erreurs								*		
Évaluation finale du projet									*	
Préparation de la soutenance finale										*

Tableau 2 calendrier du projet

Annexe 2 : Aspects innovants du projet

- Utilisation de l'intelligence artificielle pour le tri des bouteilles plastiques.
- Intégration de plusieurs technologies dans un seul système intelligent.
- Simplification et accélération du processus de recyclage.
- Potentiel d'extension et de développement futur du système.

Annexe 3 : Analyse stratégique du marché

a. Segmentation du marché cible :

- **Fabricants d'équipements écologiques** : peuvent intégrer votre système intelligent dans leurs machines existantes.
- **Startups environnementales** : cherchent à développer des solutions intelligentes pour la gestion des déchets.
- **Institutions publiques et collectivités locales** : recherchent des solutions technologiques intégrables à leur infrastructure actuelle.
- **Organisations environnementales (locales et internationales)** : financent des projets de tri et de recyclage et peuvent acheter votre système comme composant d'un écosystème global.

b. Stratégies marketing

Le système que nous avons conçu représente une solution économique par excellence, basée sur des composants open source et des algorithmes développés localement, sans dépendance à des équipements importés coûteux.

Cela permet de réduire considérablement les coûts globaux du système comparé aux systèmes similaires importés, souvent onéreux à cause des frais de transport, des taxes, et des droits de licences logicielles.

Notre solution s'adresse donc idéalement aux collectivités locales (communes, écoles, associations), tout en garantissant qualité de performance et précision dans la classification.

c. Analyse de la concurrence

Concurrence	Forces	Faiblesses
Collecteurs de plastique Institutions gouvernementales	Présence établie sur le marché Soutien gouvernemental Réseaux de collecte informels	Méthodes inefficaces Dépendance à la main-d'œuvre humaine, perte de temps et d'argent Technologies obsolètes

Tableau 3 les points forte et faible des concurrences

Annexe 4 : Plan de production et d'organisation

a. Processus de production

1. Recherche et développement
2. Entraînement et amélioration du modèle d'intelligence artificielle
3. Collecte des composants électroniques nécessaires
4. Installation du modèle intelligent sur le Raspberry Pi
5. Assemblage de l'Arduino avec les autres composants électroniques

6. Connexion entre le Raspberry Pi et l'Arduino

b. Main-d'œuvre

Le développement de ce système intelligent repose sur une équipe technique réduite, dotée de compétences pluridisciplinaires.

- Dans la phase de conception et d'expérimentation, une seule personne peut assumer plusieurs rôles.
- En cas d'extension du projet, il est recommandé de répartir les tâches comme suit :

Rôle	Tâches principales
Développeur systèmes embarqués	Programmation, intégration Arduino et Raspberry Pi
Ingénieur en intelligence artificielle	Entraînement, amélioration et intégration du modèle IA

Tableau 4 la main d'œuvre de projet

c. Approvisionnement

Les composants électroniques nécessaires sont fournis localement par des distributeurs (fournisseurs) disponibles sur le marché algérien. Cela facilite leur acquisition, réduit les coûts d'importation et évite les délais d'attente prolongés.

Partenariats

Plusieurs parties ont contribué, ou peuvent contribuer, à la réalisation de ce projet :

- Les incubateurs universitaires et laboratoires de recherche peuvent apporter un soutien scientifique et technique.
- Les distributeurs et développeurs de composants électroniques jouent un rôle clé dans la fourniture du matériel (capteurs, ordinateurs, etc.).
- Les associations et organisations environnementales peuvent tester le système sur le terrain dans le cadre de campagnes de sensibilisation ou de projets pilotes.
- Les programmes de soutien à l'innovation et aux projets étudiants peuvent offrir un financement ou un accompagnement administratif durant les phases ultérieures.

Annexe 5 : Plan financier

a. Coûts et charges

Pour gérer et offrir un système robuste de reconnaissance des bouteilles, il est essentiel de définir avec précision tous les coûts et investissements nécessaires, y compris les coûts initiaux, opérationnels et récurrents. Les principaux éléments à considérer sont les suivants :

Coûts fixes

Le tableau ci-dessous présente les coûts fixes annuels estimés nécessaires pour l'exploitation du système intelligent proposé. Il inclut les salaires, les équipements techniques, les licences et les dépenses générales de gestion du projet.

Élément	Coût estimatif (DZD)	Fréquence
Bureau (achat ou location)	20 000	Mensuel
Aménagement du bureau (mobilier, design, infrastructure)	100 000	Une seule fois
Outils de développement des algorithmes d'intelligence artificielle	200 000	Une seule fois
Système de gestion de base de données	10 000	Annuel
Logiciels de sécurité informatique et protection des données	10 000	Annuel
Efforts de marketing digital (lancement du projet)	20 000	Une seule fois
Événements sectoriels et foires commerciales	50 000	Annuel / saisonnier
Assurance des équipements et responsabilité civile	50 000	Annuel
Licences légales et permis de logiciels	20 000	Annuel
Frais de déplacement des équipes d'installation	2 000	Selon projet
Maintenance des serveurs et équipements	50 000	Annuel
Mises à jour logicielles et amélioration du système	20 000	Tous les 6 mois
Outils d'assemblage et de soudure	100 000	Une seule fois
Serveurs pour le traitement et le stockage des données	20 000	Annuel
Salaires des développeurs, data scientists et designers (5 personnes)	1 200 000	Annuel
Total	1 872 000	

Tableau 5 les couts fixe

b. Coûts variables

Le tableau suivant détaille les coûts variables selon le nombre d'unités produites. Ces coûts fluctuent directement avec la quantité fabriquée du système.

Tableau 6 les couts variables

Élément	Coût estmatif (DZD)	Fréquence
Composants électroniques	130 000	Par unité
Emballage sécurisé	2 000	Par unité
Total par unité	132 000	

c. Provisions et amortissements

Le tableau ci-dessous présente les provisions et les amortissements prévus pour garantir la pérennité du projet face aux risques techniques, juridiques ou matériels imprévus.

Élément	Montant annuel (DZD)	Type
Amortissement de l'ordinateur	30 000	Amortissement
Provision pour maintenance imprévue	50 000	Provision
Provision pour risques juridiques et sinistres	50 000	Provision
Amortissement du mobilier	10 000	Amortissement
Amortissement des outils de soudure	35 000	Amortissement
Total annuel	175 000	

Tableau 7 Provisions et amortissements

d. Coûts annuels globaux selon les scénarios

Sur la base des coûts fixes, variables et des provisions, les scénarios suivants récapitulent les coûts annuels totaux selon trois hypothèses de production : pessimiste, moyenne et optimiste.

Scénario	Coûts variables (DZD)	Coûts fixes (DZD)	Provisions (DZD)	Coût total annuel (DZD)	Coût moyen par unité
Conservateur (10 u)	1 320 000	1 872 000	175 000	3 367 000	336 700
Moyen (20 u)	2 640 000	1 872 000	175 000	4 687 000	234 350
Optimiste (50 u)	6 600 000	1 872 000	175 000	8 674 000	173 480

Tableau 8 Coûts annuels globaux selon les scénarios

e. Chiffre d'affaires

Ce tableau présente le chiffre d'affaires prévisionnel estimé en fonction du volume de production dans chaque scénario, en supposant un prix de vente fixe par unité.

Scénario	Quantité vendue	Prix unitaire (DZD)	Chiffre d'affaires
Conservateur	10	450 000	4 500 000
Moyen	20	450 000	9 000 000
Optimiste	50	450 000	22 500 000

Tableau 9 chiffre d'affaires

f. Bénéfice net et seuil de rentabilité

Le tableau suivant permet de calculer la rentabilité du projet et le seuil de rentabilité, en se basant sur la marge sur coûts variables et la couverture des coûts fixes.

Scénario	Bénéfice net (DZD)	Marge sur coût variable	Taux de marge CV	Seuil de rentabilité (unités)
Conservateur (10 u)	1 133 000	3 180 000	0,7067	6
Moyen (20 u)	8 533 000	4 360 000	0,4844	9
Optimiste (50 u)	13 853 000	15 900 000	0,7067	7

Tableau 10 Bénéfice net et seuil de rentabilité

g. Méthodes et sources de financement

Dans le cadre du développement d'un projet innovant et naissant, il est essentiel d'identifier les différentes méthodes et sources de financement pour assurer un bon démarrage et une viabilité financière. Les principales options sont :

1. Autofinancement (fonds propres)
Par les apports personnels des membres de l'équipe ou des fondateurs.
2. Famille et amis
Apport financier de proche croyant au potentiel du projet.
3. Programmes de soutien aux startups
Incubateurs, universités ou initiatives publiques offrant des financements, accompagnements ou formations.
4. Concours et subventions
Participation à des concours d'innovation ou demande de subventions auprès d'organismes de soutien.
5. Investisseurs privés (Business Angels)
Entrepreneurs qui soutiennent des projets prometteurs en échange de parts dans l'entreprise.

6. Capital-risque (Venture Capital)
Pour les phases de croissance, faire appel à des fonds d'investissement spécialisés.
7. Prêts bancaires
Option classique de financement, nécessitant une bonne gestion des capacités de remboursement.
8. Financement participatif (Crowdfunding)
Utilisation de plateformes en ligne pour collecter des fonds auprès d'un large public intéressé.

Annexe 6 : Prototype

Le projet consiste en la réalisation d'un système intelligent de tri des bouteilles plastiques, combinant des technologies matérielles (capteurs, moteurs, affichage) et un modèle d'intelligence artificielle basé sur les réseaux neuronaux convolutifs (CNN). Ce système permet de détecter, classer et trier automatiquement les bouteilles recyclables. La vidéo suivante illustre le fonctionnement du prototype développé.

<https://drive.google.com/file/d/1-IIzuGOUT9l96iKvTkswP6i4ZL3Q-PyS/view>

Business model canevas

Partenaires clés ▪ Entreprises de collecte et de recyclage des déchets ▪ Collectivités locales (mairies, APC) ▪ Entreprises technologiques ou partenaires IA ▪ Fabricants de matériel (contenants intelligents...)	Activités clés ▪ Collecte de données et entraînement du modèle, ▪ Programmation du système embarqué (Raspberry Pi / Arduino) ▪ Conception et intégration du matériel (capteurs, caméra, moteur...) ▪ Tests et amélioration continue du système ▪ Maintenance et support technique	Proposition de valeur ▪ Automatiser et optimiser le tri des bouteilles plastiques à l'aide de l'intelligence artificielle. ▪ Réduire les coûts de main d'œuvre et les erreurs humaines dans les centres de recyclage. ▪ Contribuer à la protection de l'environnement et à l'économie circulaire. ▪ Fournir un tri rapide et précis avec visualisation en temps réel.	Relation client ▪ Accompagnement à l'installation et au démarrage ▪ Mises à jour régulières du système ▪ Relation directe personnalisée avec les clients institutionnels ▪ Canaux de feedback pour améliorer l'expérience utilisateur	Segments de clientèle ▪ Investisseurs individuels intéressés par des projets écologiques et durables. ▪ Startups et petites entreprises œuvrant dans le recyclage ou la gestion des déchets. ▪ Municipalités ou autorités gouvernementales cherchant à installer des machines de recyclage plastique dans les villes.
Structure des coûts ▪ Coût du matériel (caméra, capteurs, Raspberry Pi, moteur, imprimante...) ▪ Développement logiciel et IA ▪ Collecte et annotation des données ▪ Installation et tests sur le terrain ▪ Maintenance et assistance technique ▪ Marketing et communication ▪ Salaires ou rémunérations (si applicable)		Sources de revenus ▪ Vente du système (vente des appareils) ▪ Abonnements pour maintenance et support ▪ Frais pour améliorations et mises à jour supplémentaires		

