REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE YAHIA FARES DE MEDEA



Faculté de **T**echnologie

Mémoire de fin d'études de Master

Filière : Génie electrique
Spécialité : Automatiques et systèmes

Implémentation d'un réseau de neurone convolutionnel CNN pour le tri automatique de bouteilles en plastique

Présenté par :

- Mr Merrouki Ahmed
- Mr Hariche hasene

Devant le jury composé de :

Proposé par :
Promoteur : kara reda
Co_ promotrice : Sofi naima
Co-promoteur : Abudura salam

Promotion 2021-2022

Dédicaces

Je dédie ce modeste travail :
A ma chère mère,
Quoi que je fasse ou que je dise, je ne saurai point te remercier comme il se doit. Ton affectation me couvre, ta bienveillance me guide et ta présence été ma source de force pour affronter les différents obstacles.
A mon cher père,
Tu as toujours été à mes côtés pour me soutenir et m'encourager. Que ce travail traduit ma gratitude et mon affectation,
A ma sœur et mes deux frère qui m'ont encouragé sans cesse et cru enmoi,
A mon cher binôme hacen pour leurs aides dans les moments difficiles.
A tous mes amis.

AHMED

Dédicaces

Je dédie ce modeste travail :
A ma chère mère,
Quoi que je fasse ou que je dise, je ne saurai point te remercier comme il se doit. Ton affectation me couvre, ta bienveillance me guide et ta présence été ma source de force pour affronter les différents obstacles.
A mon cher père,
Tu as toujours été à mes côtés pour me soutenir et m'encourager. Que ce travail traduit ma gratitude et mon affectation,
A ma sœur qui m'a encouragé sans cesse et cru en moi,
A mon cher binôme Ahmed pour leurs aides dans les moments difficiles.
A tous mes amis.

HACENE

Remerciements et gratitude

On remercie Dieu de nous avoir donné la santé et la volonté d'achever le but de ce mémoire.

Tout d'abord, ce travail ne se saurait pas aussi riche et n'aurait pas pu voir le jour sans l'aide et l'encadrement. On les remercie pour la qualité de leurs encadrements, pour leurs patiences, leurs soutient moral, leurs encouragements.

On remercie aussi tous les membres jury d'avoir accepté jury ce travail.

ملخص

الهدف من مشروعنا هو إنشاء آلة تصنف الزجاجات البلاستيكية باستخدام شبكة عصبية ملتوية ، ومن أجل ذلك سوف تتقن أولاً تقنيات CNN وقياس لغة الثعبان والمكتبات (tansorflow و keras و numpy و os و cv2 وما إلى ذلك) وأنشأ مجموعة البيانات الخاصة بنا

الكلمات الدالة : شبكة عصبية ملتوية,لغة الثعبان

Abstract

The objective of our project is to create a machine that classifies plastic bottles using a convolutive neural network, and for this will first master the techniques of CNN and metser the python language and libraries (tansorflow, keras, numpy,os,cv2 etc.) and created our own dataset

Key words : Convolutive neural network ,CNN,Python,tansorflow,keras,numpy

---

Résumé

L’objective de notre projet est de réaliser une machine qui classifier les bouteilles en plastiques a l’aide d’un réseau de neurone convolutif, et pour cela en va d’abord maitriser les techniques de CNN et metser le langage python et les bibliothèques (tansorflow, keras, numpy,os,cv2…) et en a créé notre propre dataset

Mots clé : reseau de neurone convolutif,CNN,Python,tansorflow,keras,numpy

Introduction générale 1

Chapitre I. Apprentissage automatique, deep Learning et réseau de neurones

I.1**.** Apprentissage automatique 02

I.1.1 Introduction 02

I.1.2 Historique 02

I.1.3 principe 03

I.1.4 les types d'apprentissage 03

I.1.4.1 L'apprentissage supervisé 03
I.1.4.2 L'apprentissage non superviséI. 04
I.1.4.3 L'apprentissage semi-supervisé 05
I.1.4.4 L'apprentissage partiellement supervisé 05

I.1.4.5 L'apprentissage par renforcement 05

I.1.5 Types de problème 05
I.2 Deep Learning 06
I.2.1. Définition 06
I.2.2. Architectures de réseaux de neurones profonds 06
I.2.3. principe 06
I.2.4. Les applications du Deep Learning 06
I.2.4.1. La reconnaissance faciale 06
I.2.4.2. Le traitement automatique de langage naturel 07

I.2.4.3. Voitures autonomes 07
I.2.4.4. Recherche vocale et assistants à commande vocale 07
I.2.4.5. Traduction automatique 07
I.2.4.6. Reconnaissance d'image 07
I.2.4.7. La description automatique d'image 08
I.2.4.8. Colorisation automatique 08
I.2.4.9. La détection du cancer du cerveau 08
I.3. Réseaux de neurones 08
I.3.1. Définition 08
I.3.2. Les différents types de réseaux de neurones 09
I.3.2.1. Réseaux de neurones artificiels (ANN) 09
I.3.2.2. Réseaux de neurones récurrents (RNN) 10
I.3.2.3. Réseaux de neurones à convolution (CNN) 10
I.3.3. Avantages du réseau de neurones à convolution (CNN) 11
I.4. Conclusion 12

Chapitre II. la conception du système de classification et reconnaissance automatique

Introduction 14
L'influence de nombre de couche convolutive 14
L'influence du nombre de couche maxpooling 16
L'influence de la taille du filtre 18
Le meilleur model du CNN 21
Conclusion 24

Chapitre III. Implémentation du programme

Introduction 25
Recyclage 25
Définition 25
Type de recyclage 25
Avantage de recyclage 25
Le cycle de recyclage de bouteilles de pastique 26
La machine RVM 28
Fonctionnement de la machine RVM 28
Implémentation du système 29
Python 29
TansorFlow 29
Keras 29
La structure de notre programme 29
Conclusion 36

Conclusion générale et perspectives 37

Références bibliographiques 38

Symbole	Signification
IBM	International Business Machine
NLP	natural language processing
DL	Deep Leaning
ANN	Artificial Neural Network
CNN	Convollutional Neural Network
RNN	Recurrent Neural Network
RVM	Les Reverse Vending Machines
PET	polytéréphtalate d'éthylène
IA	I'intelligence Artificielle
HTML	HyperText Markup Language
ONEIROS	Open-ended Neuro-Electronic Intelligent Robot Operating System
API	application programming interface
CNTK	Cognitive Toolkit

Chapitre I. Apprentissage automatique, deep Learning et réseau de neurones

Tableau I.1.	Les types de problème	05

Chapitre II. La conception du système de classification et reconnaissance automatique

Tableau II.1.	Les structures des modèles d'entrainement pour L'influence de nombre de couche convolutive.	14
Tableau II.2.	La précision et temps entrainement des modèles pour L'influence de nombre de couche convolutive.	16
Tableau II.3.	Les structures des modèles d'entrainement pour l'influence de la couche maxpooling	17
Tableau II.4.	La précision et temps entrainement des modèles pour l'influence de la couche maxpooling	18
Tableau II.5.	Les structures des modèles d'entrainement pour l'influence de la taille du filtre	19
Tableau II.6.	La précision et temps entrainement des modèles pour l'influence de la taille du filtre	20
Tableau II.7.	La structure du modèle d'entrainement du meilleur modèle	21
Tableau II.8.	La précision et temps entrainement du meilleur modèle.	21
Tableau II.9.	La précision et temps d'entrainement du meilleur modèle pour différent epoches.	22
Tableau II.10	La structure du modèle d'entrainement du meilleur modèle modifié	23
Tableau II.11.	La précision et temps d'entrainement du meilleur modèle et meilleur modèle	23

Chapitre I. Apprentissage automatique, deep Learning et réseau de neurones

Figure I.1.	Archetectures du RNN	10
Figure I.2.	La carte caractéristique d'une image	11
Figure I.3.	Towards/data/science	11
Figure I.4.	Visage humain	12

Chapitre II. la conception du système de classification et reconnaissance automatique

Figure II.1.	La précision de validation d'une seule couche convolutive.	15
Figure II.2.	La précision de validation de deux couche convolutive.	15
Figure II.3.	La précision de validation de trois couche convolutive.	16
Figure II.4.	La précision avec une seule couche de maxpooling.	17
Figure II.5.	La précision avec deux couches de maxpooling.	18
Figure II.6.	précision avec une taille de 3*3.	19
Figure II.7	La précision avec un filtre de taille 5*5	20
Figure II.8	La précision avec un filtre de taille 7*7	20
Figure II.9	La précision du meilleur modèle	22
Figure II.10	La précision du meilleur modèle avec 8 epoches	23
Figure II.11	La précision du meilleur modèle modifié	23

Chapitre III. Implémentation du programme

Figure III.1.	bouteilles	26
Figure III.2.	BAC	27
Figure III.3.	machine de collection	27
Figure III.4.	schéma de recyclage des bouteilles	28
Figure III.5.	machine RVM	28
Figure III.6.	importation des bibliothèques	30
Figure III.7.	création des labels	30
Figure III.8.	Model	31
Figure III.9.	compilation et sommaire	32
Figure III.10.	transformer les catégories en numérations	33
Figure III.11.	division des images	33
Figure III.12.	les résultat de loss et de l'accuracy	33
Figure III.13.	la courbe des pertes en fonction des epoches	34
Figure III.14.	la courbe de la precision en fonction des epoches	35
Figure III.15.	résultats de prédictions	35
Figure III.16.	le code de camera	36
Figure III.17.	résultats de prédictions au tempe réel	36

Introduction générale

Au-delà de la pollution causée par sa production, de nombreux déchets plastiques ne sont pas jetés dans des poubelles et se retrouvent dans la nature. Ils mettent alors très longtemps à se dégrader. Le plus souvent, le plastique se désintègre sous forme de particules (micro ou nano plastiques). Ils polluent les sols, les fleuves et les océans et sont dangereux pour la faune qui vit dans ces milieux.

Et là nous avant inspiré des payés développé une idée qui permettre de diminuer ses déchets de propager, en a pensé à convaincre les gens de recycle les bouteilles vides après les consomme tout ça en donnant une récompense pour chaque bouteille recyclée comme une source de motivation.

Notre machine est inspirée de la fameuse machine RVM qui classifier les bouteilles pour un prix acceptable, pour réaliser cette machine il faut d'abord savoir comment elle marche, cette dernière classifier les bouteilles et les identifier à l'aide d'un réseau de neurone convolutif.

D'après ce qu'on a dit le CNN et le cerveau de cette machine, et pour qu'on peut le programmer à nos conditions on a besoin d'un langage spécifier et de le métrise.

Chapitre I. Apprentissage automatique, deep Learning et réseau de neurones

Apprentissage automatique

Introduction

L'apprentissage automatique est une discipline scientifique et l'un des domaines de recherche de l'intelligence artificielle. **[1]**

L'apprentissage automatique fait référence au développement, à l'analyse et à la mise en œuvre de méthodes qui permettent (au sens large) aux machines d'évoluer à travers un processus d'apprentissage pour effectuer des tâches difficiles ou impossibles avec des moyens algorithmiques plus traditionnels. **[1]**

Des systèmes complexes peuvent être analysés, y compris des données liées à des valeurs symboliques (par exemple, pas des nombres, mais des probabilités ou des intervalles définis sur des attributs numériques) ou un ensemble de modalités possibles sur des attributs de valeur (nombres) ou des classifications. L'analyse peut même se concentrer sur des données présentées sous forme de graphiques ou d'arbres ou même de courbes (comme des courbes d'évolution dans le temps mesuré ; on parle alors de données continues, et non de données discrètes liées à des attributs - valeurs classiques). **[1]**

Historique

Il continuera à poser les bases de l'apprentissage automatique, avec son article sur « L'ordinateur et l'intelligence » en 1950, dans lequel il développe, entre autres, le test de Turing. En 1943, le neurophysiologiste Warren McCulloch et le mathématicien Walter Pitts publient un article décrivant le fonctionnement de neurones en les représentant à l'aide de circuits électriques Arthur Samuel, informaticien américain pionnier dans le secteur de l'intelligence artificielle, est le premier à faire usage de l'expression machine learning en 1959 à la suite de la création de son programme pour IBM en 1952 Une avancée majeure dans le secteur de l'intelligence machine est le succès de l'ordinateur développé par IBM, Deep Blue, qui est le premier à vaincre le champion mondial d'échecs Garry Kasparov en 1997. Le projet Deep Blue en inspirera nombre d'autres dans le cadre de l'intelligence artificielle, particulièrement un autre grand défi : IBM Watson, l'ordinateur dont le but est de gagner au jeu Jeopardy! En 2012, un réseau neuronal développé par Google parvient à reconnaître des visages humains ainsi que des chats dans des vidéos YouTube, **[4]**

En 2014, 64 ans après la prédiction d'Alan Turing, le dialogueur Eugene Goostman est le premier à réussir le test de Turing en parvenant à convaincre 33 % des juges humains au bout de

cinq minutes de conversation qu'il est non pas un ordinateur, mais un garçon ukrainien de 13 ans. En 2015, une nouvelle étape importante est atteinte lorsque l'ordinateur « AlphaGo » de Google gagne contre un des meilleurs joueurs au jeu de Go, jeu de plateau considéré comme le plus dur du monde. En 2016, un système d'intelligence artificielle à base d'apprentissage automatique nommé LipNet parvient à lire sur les lèvres avec un grand taux de succès. **[4]**

principe

La théorie de l'apprentissage utilise des outils mathématiques dérivés de la théorie des probabilités et de la théorie de l'information, Cela vous permet d'évaluer l'optimalité de certaines méthodes par rapport aux autres.

On peut citer trois types d'algorithme d'apprentissage automatique :

• Apprentissage supervisé.

• Apprentissage non supervisé.

• Apprentissage par renforcement.

les types d'apprentissage

L'apprentissage supervisé

L'apprentissage supervisé est la tâche d'apprentissage automatique la plus simple et la plus connue. Il est basé sur un certain nombre d'exemples pré classifiés, dans lesquels est connu à priori la catégorie à laquelle appartient chacune des entrées utilisées comme exemples. **[6]**

Dans ce cas, la question cruciale est le problème de généralisation, après l'analyse d'un échantillon d'exemples, le système devrait produire un modèle qui devrait fonctionner pour toutes les entrées possibles.

L'ensemble de données pour l'entrainement, est constitué de données étiquetées, c'est-à-dire d'objets et de leurs classes associées Afin de mieux comprendre ce concept, prenons un exemple : un utilisateur reçoit chaque jour un grand nombre d'e-mails, certains sont des e-mails d'entreprises importants et d'autres sont des e-mails indésirables non sollicités ou des spam.

Un algorithme supervisé sera présenté avec un grand nombre d'e-mails qui ont déjà été étiquetés par l'utilisateur comme spam ou non spam. L'algorithme fonctionnera sur toutes les données étiquetées, faire des prédictions sur l'e-mail et voir si c'est un spam ou non.

Cela signifie que l'algorithme examinera chaque exemple et fera une prédiction pour chacun pour savoir si l'e-mail est un spam ou pas. La première fois, l'algorithme fonctionne sur toutes les données non étiquetées, la plupart des e-mails seront mal étiquetés car il peut fonctionner assez mal au début,Cependant, après chaque exécution, l'algorithme compare sa prédiction au résultat souhaité.

Dans l'exemple que nous avons utilisé, nous avons décrit un processus dans lequel un algorithme apprend à partir de données étiquetées (emails qui ont été catégorisés comme spam ou non-spam).

Dans certains cas, le résultat n'est pas nécessairement discret et il se peut que nous n'ayons pas un nombre fini de classes dans lesquelles classer nos données.

Par exemple, nous essayons peut-être de prédire l'espérance de vie d'un groupe de personnes en fonction de paramètres de santé préétablis.

Dans un problème de régression, l'ensemble d'apprentissage est une paire formée par un objet et une valeur numérique associée.

Il existe plusieurs algorithmes d'apprentissage supervisé qui ont été développés pour la classification et la régression. Parmi tous, les arbres de décision, les règles de décision, les réseaux de neurones et les réseaux bayé siens.

L'apprentissage non supervisé

La deuxième classe d'algorithmes d'apprentissage automatique est appelée apprentissage non supervisé, dans ce cas, nous n'étiquetons pas les données au préalable, nous laissons plutôt l'algorithme arriver à sa conclusion. Ce type d'apprentissage est important car il est beaucoup plus commun dans le cerveau humain que l'apprentissage supervisé. Les algorithmes d'apprentissage non supervisé sont particulièrement utilisés dans les problèmes de clustering, dans lesquels, étant donné une collection d'objets, nous voulons être en mesure de comprendre et de montrer leurs relations. **[6]**

L'apprentissage semi-supervisé

Effectué de manière probabiliste ou non, il vise à faire apparaitre la distribution sous-jacente des « exemples » dans leur espace de description. Il est mis en oeuvre quand des données (ou « étiquettes ») manquent... Le modèle doit utiliser des exemples non-étiquetés pouvant néanmoins renseigner.**[6]**

Ex : En médecine : il peut constituer une aide au diagnostic ou au choix des moyens les moins onéreux de tests de diagnostics.

L'apprentissage partiellement supervisé

Quand l'étiquetage des données est partiel. C'est le cas quand un modèle énonce qu'une donnée n'appartient pas à une classe A, mais peut-être à une classe B ou C (A, B et C étant 3 maladies par exemple évoquées dans le cadre d'un diagnostic différentiel). **[6]**

L’apprentissage par renforcement

L'apprentissage par renforcement est une approche de l'intelligence artificielle qui met l'accent sur l'apprentissage du système à travers ses interactions avec l'environnement. Par exemple, un système qui modélise un joueur d'échecs qui utilise le résultat des étapes précédentes pour améliorer ses performances, est un système qui apprend avec le renforcement. **[6]**

Types de problème

Tableau I.1.les types de problème

Apprentissage supervisé	Apprentissage non supervisé	Apprentissage par renforcement
. Classification . Régression	. Clustering . Segmentation . Réduction de dimension	. Processus de décision . Systèmes de recommandation

Deep Learning

définition

L'apprentissage en profondeur (deep Learning) est un domaine de recherche sur l'apprentissage automatique basé sur un type particulier de mécanisme d'apprentissage. Il est caractérisé par l'effort de créer un modèle d'apprentissage à plusieurs niveaux, dans lequel les niveaux les plus profonds prennent en compte les résultats des niveaux précédents, les transformant et en faisant toujours plus d'abstraction. **[7]**

Cet aperçu des niveaux d'apprentissage est inspiré par la façon dont le cerveau traite l'information et apprend en réagissant aux stimuli externes.

Chaque niveau d'apprentissage correspond, par hyothèse, à l'une des différentes zones qui composent le cortex cérébral

Architectures de réseaux de neurones profonds :

Il existe un grand nombre de variables d'architectures profondes. La plupart d'entre eux sont dérivés de certaines architectures originales. Dans ces architectures mathématiques, chaque neurone effectue des calculs simples mais les données d'entrées passent à travers plusieurs couches de calcul avant de produire une sortie. Les résultats de la première couche de neurones servent d'entrée au calcul de la couche suivante et ainsi de suite. **[8]**

principe

Après avoir été préalablement entraîné sur une base d'exemples, un réseau de neurones artificiel pourra typiquement être utilisé dans la reconnaissance d'images, pour identifier un chat par exemple. Chaque couche du réseau sera dédiée à l'identification d'une caractéristique de l'animal : la silhouette, la tête, les deux oreilles, les quatre pattes, le pelage, etc. Les réseaux de neurones artificiels sont également très utilisés dans le traitement automatique du langage ou natural language processing (NLP). **[8]**

Les applications du Deep Learning

Le Deep Learning est utilisé pour une large variété d'applications telles que la détection de maladie, l'inspection d'équipement industriel, les véhicules autonomes, la découverte d'exoplanètes ou de médicaments, l'étude du génome ou encore la lutte contre le réchauffement climatique.

La reconnaissance faciale

Les yeux, le nez, la bouche, tout autant de caractéristiques qu'un algorithme de DL va apprendre à détecter sur une photo. Il va s'agir en premier lieu de donner un certain nombre d'images à l'algorithme, puis à force d'entraînement, l'algorithme va être en mesure de détecter un visage sur une image. **[8]**

Le traitement automatique de langage naturel

Le traitement automatique de langage naturel est une autre application du DL. L'algorithme va par exemple comprendre ce qui est dit dans un avis Google, ou va communiquer avec des personnes via des chatbots. **[8]**

Voitures autonomes

Les entreprises qui construisent de tels types de services d'aide à la conduite, ainsi que des voitures autonomes telles que Google, doivent apprendre à un ordinateur à maîtriser certaines parties essentielles de la conduite à l'aide de systèmes de capteurs numériques au lieu de l'esprit humain. Pour ce faire, les entreprises commencent généralement par entraîner des algorithmes utilisant une grande quantité de données. Vous pouvez imaginer comment un enfant apprend grâce à des expériences constantes et à la réplication. **[8]**

Recherche vocale et assistants à commande vocale

L'un des domaines d'utilisation les plus populaires de DL est la recherche vocale et les assistants intelligents à commande vocale. Le Siri d'Apple est sur le marché depuis octobre 2011. D'une année après Siri. Le plus récent des assistants intelligents à commande vocale est Microsoft Cortana. **[8]**

Traduction automatique

Il s'agit d'une tâche dans laquelle des mots, expressions ou phrases donnés dans une langue sont automatiquement traduits dans une autre langue. La traduction automatique existe depuis longtemps, mais DL permet d'obtenir les meilleurs résultats dans deux domaines spécifiques :

_ Traduction automatique de texte

_ Traduction automatique d'images

La traduction de texte peut être effectuée sans aucun traitement préalable de la séquence, ce qui permet à l'algorithme d'apprendre les dépendances entre les mots et leur correspondance avec une nouvelle langue. **[8]**

Reconnaissance d'image

Un autre domaine populaire en matière de DL est la reconnaissance d'image. Son objectif est de reconnaître et d'identifier les personnes et les objets dans les images, ainsi que de comprendre le contenu et le contexte. La reconnaissance d'image est déjà utilisée dans plusieurs secteurs tels que les jeux, les médias sociaux, la vente au détail, le tourisme, etc. Cette tâche nécessite la classification des objets d'une photo parmi un ensemble d'objets connus auparavant. Une variante plus complexe de cette tâche, appelée détection d'objet, consiste à identifier spécifiquement un ou plusieurs objets dans la scène de la photo et à dessiner un cadre autour d'eux. **[8]**

La description automatique d'image

Une fois que vous pouvez détecter des objets sur des photographies et générer des étiquettes pour ces objets, vous pouvez voir que l'étape suivante consiste à transformer ces étiquettes en description de phrase cohérente. Généralement, les systèmes impliquent l'utilisation de très grands réseaux de neurones convolutifs pour la détection d'objets sur les photographies, puis d'un réseau de neurones récurrent comme une mémoire à court terme à long terme pour transformer les étiquettes en une phrase cohérente. **[8]**

Colorisation automatique

La colorisation de l'image pose le problème de l'ajout de couleurs aux photographies noir et blanc. Le DL peut être utilisé pour utiliser les objets et leur contexte dans la photographie pour colorer l'image, un peu comme un opérateur humain pourrait aborder le problème. Cette capacité tire parti des réseaux de neurones de convolution de grande qualité et de très grande taille formés pour ImageNet et cooptés pour le problème de la colorisation de l'image. **[8]**

La détection du cancer du cerveau

Une équipe de chercheurs français a noté qu'il était difficile de détecter les cellules cancéreuses du cerveau invasives au cours d'une intervention chirurgicale, en partie à cause des effets de l'éclairage dans les salles l'opération. Ils ont découvert que l'utilisation de réseaux de neurones conjointement avec la spectroscopie Raman pendant les opérations leur permettaitde détecter les cellules cancéreuses plus facilement et de réduire le cancer résiduel après l'opération. **[8]**

Réseaux de neurones :

définition

Ensemble de neurones formels interconnectés permettant la résolution de problèmes complexes tels que la reconnaissance des formes ou le traitement du langage naturel, grâce à l'ajustement des coefficients de pondération dans une phase d'apprentissage.

Un réseau neuronal s'inspire du fonctionnement des neurones biologiques et prend corps dans un ordinateur sous forme d'un algorithme. Le réseau neuronal peut se modifier lui-même en fonction des résultats de ses actions, ce qui permet l'apprentissage et la résolution de problèmes sans algorithme, donc sans programmation classique. **[9]**

Les différents types de réseaux de neurones

Les différents types de réseaux de neurones en apprentissage profond, tels que les réseaux de neurones convolutifs (CNN), les réseaux de neurones récurrents (RNN), les réseaux de neurones artificiels (ANN), etc. changent notre façon d'interagir avec le monde. Ces différents types de réseaux de neurones sont au cœur de la révolution de l'apprentissage en profondeur, alimentant des applications telles que les véhicules aériens sans pilote, les voitures autonomes, la reconnaissance vocale, etc. Les trois types importants de réseaux de neurones qui constituent la base de la plupart des modèles pré-formés en apprentissage en profondeur sont :

- Réseaux de neurones artificiels (ANN)
- Réseaux de neurones récurrents (RNN)
- Réseaux de neurones à convolution (CNN)

Réseaux de neurones artificiels (ANN)

Un seul perceptron (ou neurone) peut être imaginé comme une régression logistique. Le réseau de neurones artificiels, ou ANN, est un groupe de plusieurs perceptrons/neurones à chaque couche,

L'ANN est également connu sous le nom de **« réseau neuronal Feed-Forward »** car les entrées ne sont traitées que dans le sens direct.

ANN se compose de 3 couches : Input, Hidden et Output.

La couche d'entrée accepte les entrées, la couche cachée traite les entrées et la couche desortie produit le résultat essentiellement, Chaque couche essaie d'apprendre certains poids. ANN peut être utilisé pour résoudre desproblèmes liés à :

- Données tabulaires
- Données d'image
- Données textuelles

Réseaux de neurones récurrents (RNN)

Essayons d'abord de comprendre la différence entre un RNN et un ANN du point de vue de l'architecture :

Une contrainte de bouclage sur la couche cachée d'ANN se transforme en RNN.

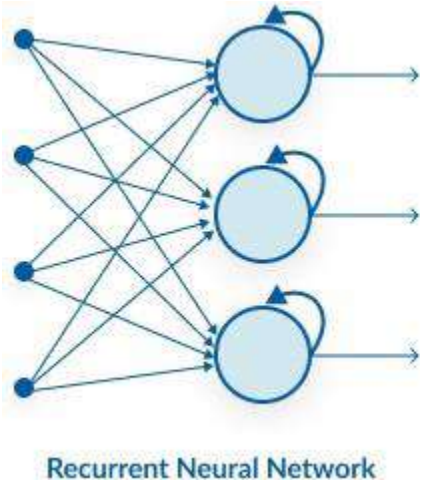


Figure I.1. Archetectures du RNN

RNN a une connexion récurrente sur l'état caché, Cette contrainte de bouclage garantit que les informations séquentielles sont capturées dans les données d'entrée, Nous pouvons utiliser des réseaux de neurones récurrents pour résoudre les problèmes liés à :

- Données de séries chronologiques
- Données textuelles
- Données audio

I.3.2.3 Réseaux de neurones à convolution (CNN) :

Les réseaux de neurones convolutifs (CNN) font fureur dans la communauté de l'apprentissage profond en ce moment. Ces modèles CNN sont utilisés dans différentes applications et domaines, et ils sont particulièrement répandus dans les projets de traitement d'images et de vidéos. Les éléments constitutifs des CNN sont des filtres, c'est-à-dire des noyaux**.** Les noyaux sont utilisés pour extraire les caractéristiques pertinentes de l'entrée à l'aide de l'opération de convolution. Essayons de saisir l'importance des filtres utilisant des images comme données d'entrée. La convolution d'une image avec des filtres donne une carte de caractéristiques. **[10]**

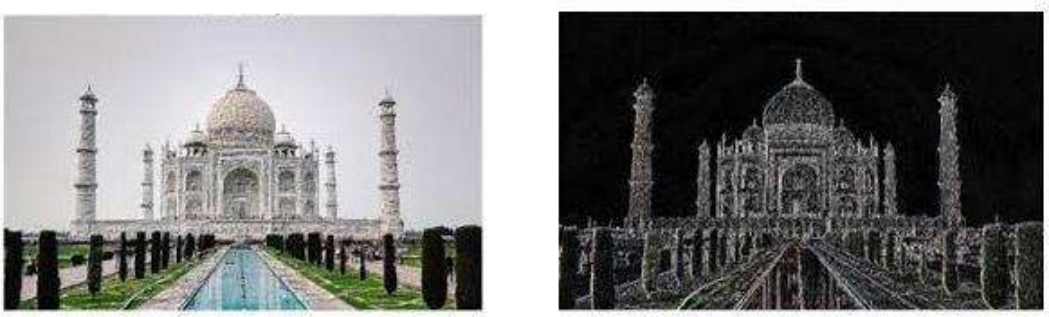


Figure I.2. La carte caractéristique d'une image

Avantages du réseau de neurones à convolution (CNN)

CNN apprend les filtres automatiquement sans le mentionner explicitement. Ces filtres aident à extraire les caractéristiques correctes et pertinentes des données d'entrée

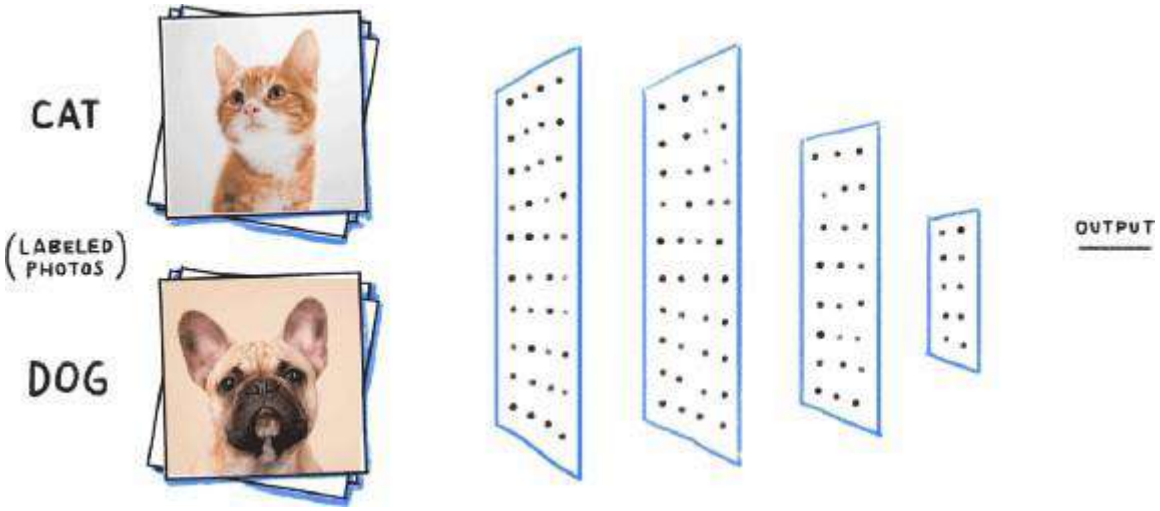


Figure I.3. Towards/data/science

CNN capture les caractéristiques spatiales d'une image. Les caractéristiques spatiales font référence à la disposition des pixels et à la relation entre eux dans une image. Ils nous aident à identifier l'objet avec précision, l'emplacement d'un objet, ainsi que sa relation avec d'autres objets dans une image. **[25]**



Figure I.4.visage humain

Dans la figure I.4, nous pouvons facilement identifier qu'il s'agit d'un visage humain en regardant des caractéristiques spécifiques comme les yeux, le nez, la bouche, etc. Nous pouvons également voir comment ces caractéristiques spécifiques sont disposées dans une image. C'est exactement ce que les CNN sont capables de capturer.

Conclusion :

Dans ce chapitre on a présenté les notions importantes qui sont en relation avec Apprentissage automatique, deep Learning, et réseau de neurones (définition, Architectures ,différente type ….etc).

Aussi qu'une vision générale sur l'apprentissage profond, toute on donnant en détail la méthode choisie dans notre travail de recherche qui est le CNNs

Chapitre II.

la conception du système de classification et reconnaissance automatique

Introduction

Un réseau neuronal convolutif (CNN) est un type de réseau neuronal artificiel utilisé principalement pour la reconnaissance et le traitement des images, en raison de sa capacité à reconnaître les motifs dans les images. Un CNN est un outil puissant, mais il nécessite des millions de points de données étiquetés pour la formation. Les CNN doivent être formés à l'aide de processeurs puissants, tels qu'un GPU ou un NPU, s'ils veulent produire des résultats assez rapidement pour être utiles.

Dans ce chapitre on présente la conception du système de classification et reconnaissance automatique des nombres basé sur les réseaux de neurones convolutionnels CNN L'implémentation de ce réseau de ce logiciel a été réalisée en langage Python.

L'influence de nombre de couche convolutive

Pour démontré l'influence du nombre de couche de convolution sur la qualité de Learning CNN on fait trois model :

Table II.1. Les structures des modèles d'entrainement pour L'influence de nombre de couche convolutive.

Model#1	Convolution filtre 7x7 32	Maxpooling	Dropout	Flatten	Dense	Dropout	Softmaax
Model#2	Convolution filtre 7x7 32 Convolution filtre 7x7 32	Maxpooling	Dropout	Flatten	Dense	Dropout	Softmaax
Model#3	Convolution filtre 7x7 32 Convolution filtre 7x7 32 Convolution filtre 7x7 32	Maxpooling	Dropout	Flatten	Dense	Dropout	Softmaax

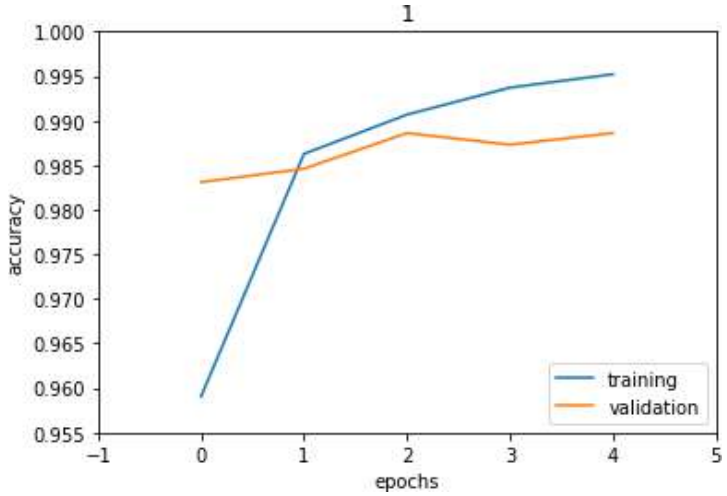


Figure II.1. La précision de validation d'une seule couche convolutive.

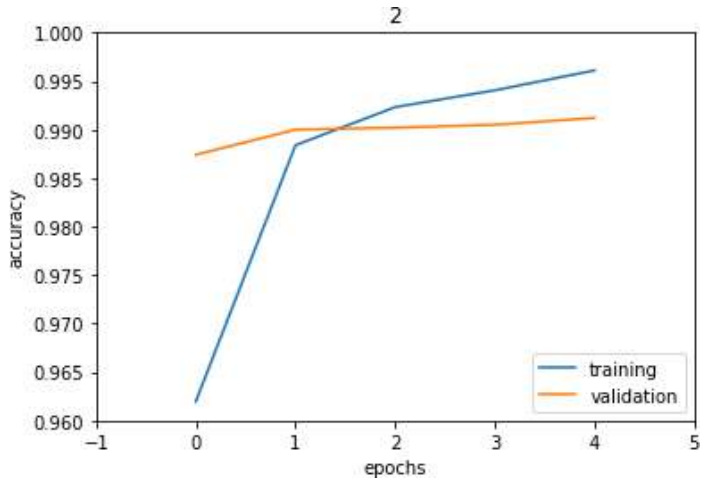


Figure II.2. La précision de validation de deux couche convolutive.

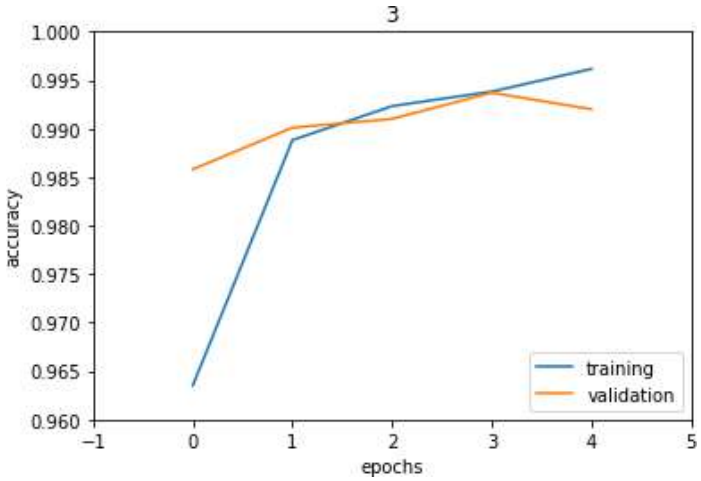


Figure II.3. La précision de validation de trois couche convolutive.

Tableau II.2. : La précision et temps entrainement des modèles pour L'influence de nombre de couche convolutive.

	Model#1	Model#2	Model#3
Précision	98.86%	99.12%	99.20%
Training time	6min 20 s	27min.11s	52min

Lorsque on a utilisé trois couches de convolution, on a observé une petite augmentation de la précision par rapport à la configuration à deux couches. De ces différents tests on a conclu qu'en arrivant à un certain nombre de couches de convolutions, les performances du réseau augmentent peu par rapport à l'augmentation de la hauteur des temps d'entraînement.

L'influence du nombre de couche maxpooling

Pour comprendre l'influence du nombre de couches maxpooling dans un réseau de neurones CNN, on a augmenté le nombre de couches. Cependant, on a fixé le nombre de couches de convolution à trois avec un filtre de taille 5 * 5. Les deux architectures ainsi que les résultats associés sont regroupés dans le tableau et les figures ci-dessous**.**

Table II.3. Les structures des modèles d'entrainement pour l'influence de la couche maxpooling.

Model#3	Convolution filtre 5x5 32 Convolution filtre 5x5 32 Convolution filtre 5x5 32	Maxpooling	/	/	Dropout	Flatten
Model#4	Convolution filtre 5x5 32 Convolution filtre 5x5 32 Convolution filtre 5x5 32	Maxpooling	Convolution filtre 5x5 32 Convolution filtre 5x5 32 Convolution filtre 5x5 32	Maxpooling	Dropout	Flatten
Dense	Dropout	Softmax				
Dense	Dropout	Softmax				

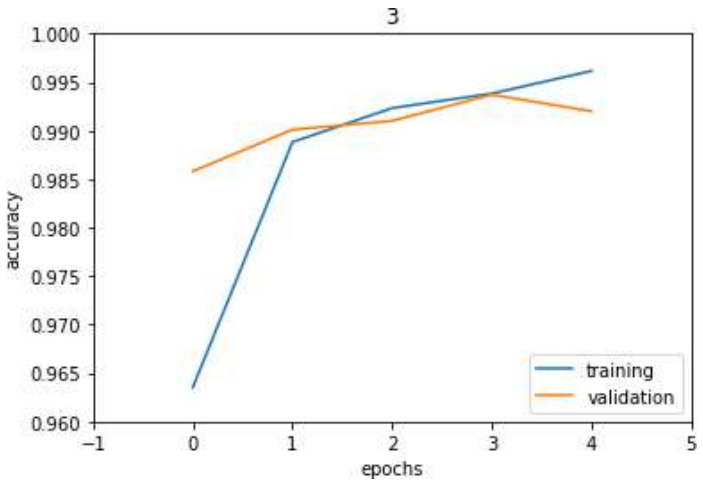


Figure II.4. La précision avec une seule couche de maxpooling.

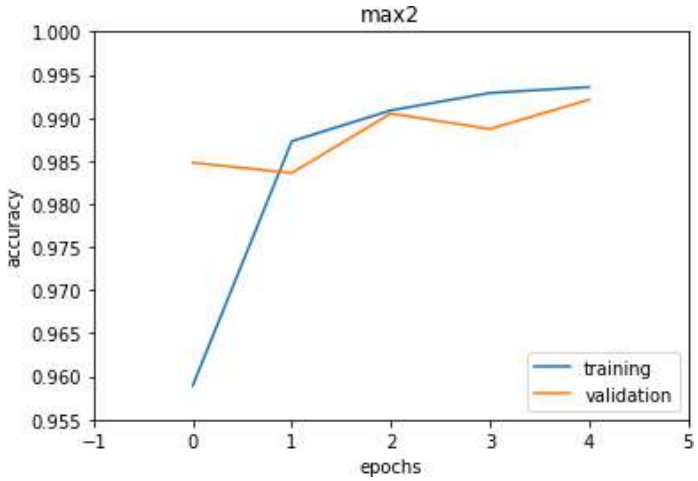


Figure II.5. La précision avec deux couches de maxpooling.

Table II.4. La précision et temps entrainement des modèles pour l'influence de la couche maxpooling.

	Mode#3	Mode#4
Précision	99.2%	99.21%
Training Time	52min	1h 0min.10s

De la Figure (II.3.1, II.3.2), on peut déduire: Lors de l'utilisation de deux couches de maxpooling, je constate une augmentation de 0.01 % de la précision de la validation par rapport à une seule couche.

L'influence de la taille du filtre

Dans cette partie, on veut voir l'effet de la taille du filtre sur les performances de CNN. Pour cela, on a choisi crée trois tailles de filtre : 3*3, 5*5 et 7*7, en plus de ça on a choisi le meilleur model des tests précèdent.

Table II.5. Les structures des modèles d'entrainement pour l'influence de la taille du filtre

Model#5	Convolution filtre 3x3 32 Convolution filtre 3x3 32	Maxpooling	Convolution filtre 3x3 64 Convolution filtre 3x3 64	Maxpooling	Dropout	Flatten
Model#6	Convolution filtre 5x5 32 Convolution filtre 5x5 32	Maxpooling	Convolution filtre 5x5 64 Convolution filtre 5x5 64	Maxpooling	Dropout	Flatten
Model#7	Convolution filtre 7x7 32 Convolution filtre 7x7 32	Maxpooling	Convolution filtre 7x7 64 Convolution filtre 7x7 64	Maxpooling	Dropout	Flatten
Dense	Dropout	Softmax				
Dense	Dropout	Softmax				
Dense	Dropout	Softmax				

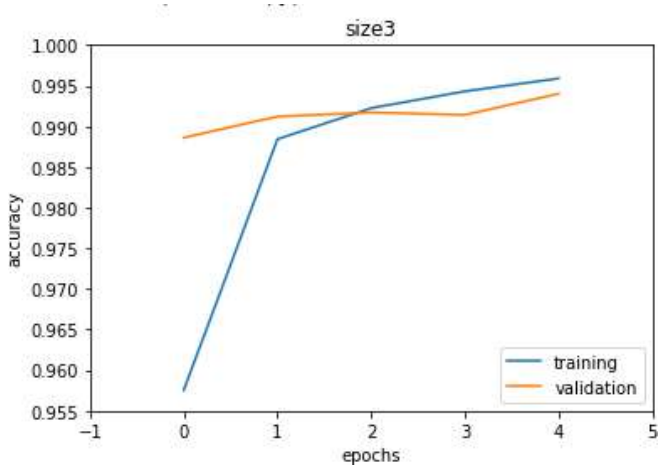


Figure II.6. précision avec une taille de 3*3.

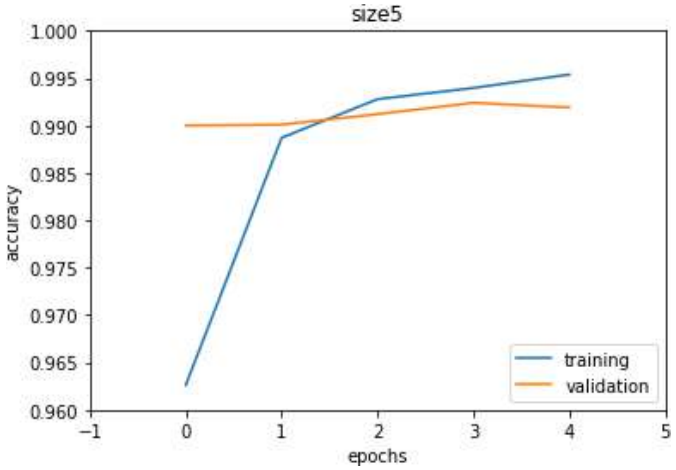


Figure II.7. La précision avec un filtre de taille 5*5.

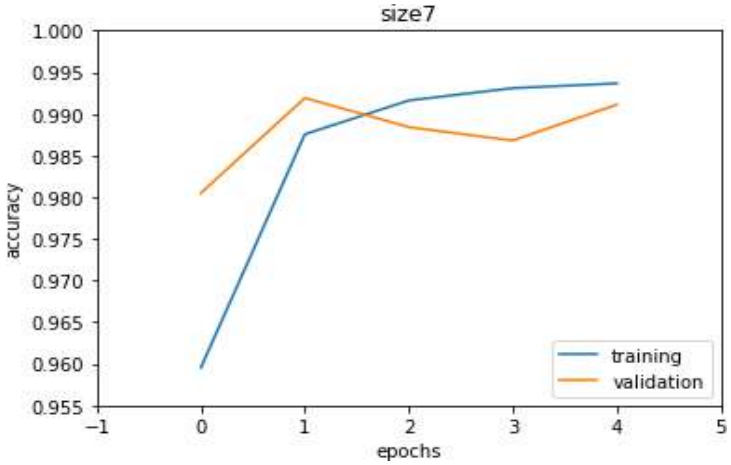


Figure II.8. La précision avec un filtre de taille 7*7.

Table II.6. La précision et temps entrainement des modèles pour l'influence de la taille du filtre.

	Model#5	Model#6	Model#7
Précision	99.40%	99.19%	99.11%
Training time	23min 29 s	52min.33s	1h 27min 28s

D'après la figure (II.4.1, II.4.2, II.4.3), je vois que la meilleure taille de filtre est 3*3. De ces tests, je peux logiquement conclure que le sous-dimensionnement des filtres augmente la précision de CNN.

II.5. Le meilleur model du CNN

Après avoir effectué les différents tests précédents et vu les résultats obtenus, on a sélectionné la configuration suivant :

Table II.7. La structure du modèle d'entrainement du meilleur modèle

Model #8	Convolution filtre 3x3 32 Convolution filtre 3x3 32 Convolution filtre 3x3 32	Maxpooling	Convolution filtre 3x3 64 Convolution filtre 3x3 64 Convolution filtre 3x3 64	Maxpooling	Dropout	Flatten
Dense	Dropout	Softmax				

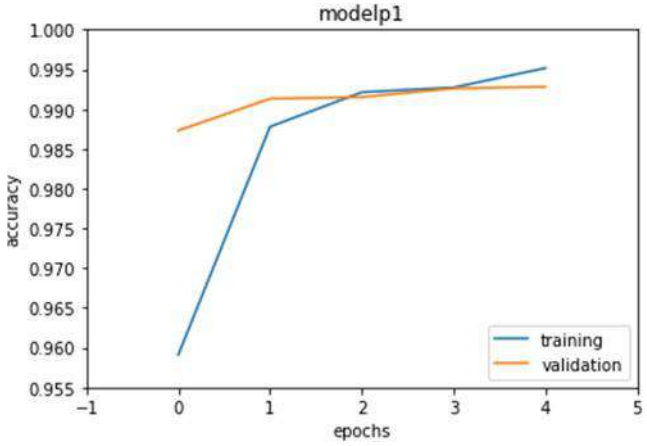


Figure II.9 . La précision du meilleur modèle

Table II.8. La précision et temps entrainement du meilleur modèle.

	Model#8
Précision	99.28%
Training Time	33min 37s

Donc on a vu que le modèle 8 est la meilleure option jusqu'à présent à cause de cela on refait le modèle 8 avec une époque de 8 au lieu de 5 et les résultats sont dans le tableau suivant.

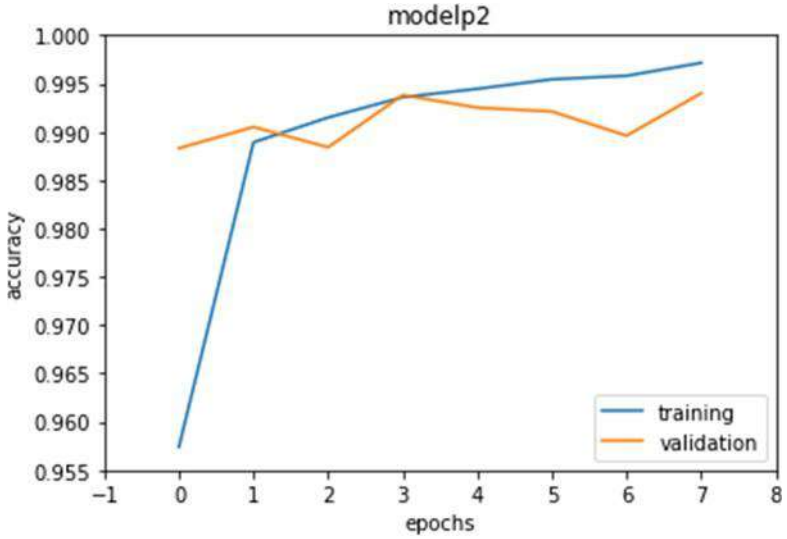


Figure II.10. La précision du meilleur modèle avec 8 epoches.

Tableau II.9. La précision et temps d'entrainement du meilleur modèle pour différent epoches.

	Model#8 avec 5 epoches	Model#8 avec 8 epoches
Précision	99.28%	99.40%
Training time	33min 37 s	53min.40s

Afin d'obtenir plus de précision, nous ajoutons une autre couche de convolution et nous faisons un autre test et voyons ce que nous obtenons.

Tableau II.10. La structure du modèle d'entrainement du meilleur modèle modifié

Model #8	Convolution filtre 3x3 32 Convolution filtre 3x3 32 Convolution filtre 3x3 32	Maxpooling	Convolution filtre 3x3 64 Convolution filtre 3x3 64 Convolution filtre 3x3 64	Maxpooling	Convolution filtre 3x3 128
Maxpooling	Dropout	Flatten	Dense	Dropout	Softmax

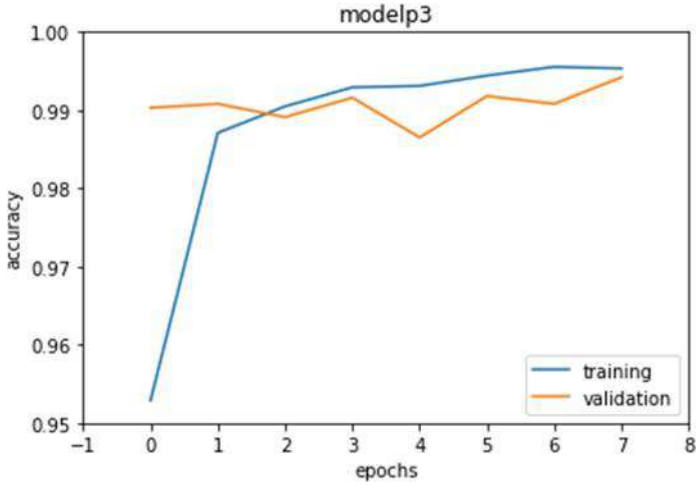


Figure II.11. La précision du meilleur modèle modifié

Tableau II.11. La précision et temps d'entrainement du meilleur modèle et meilleur modèle

	Model#8 avec 8 epoches	Model#8 modifié avec 8 epoches
Précision	99.40%	99.40%
Training time	53min.40s	1h 1min.26s

Lorsque en ajoutant une couche pour 128 le temps d'exécution augmenté de 8min 14s la précision n'a pas augmenté cette fois, et le meilleur résultat a été le modèle 5.2 avec une précision de 99.40%, temps d'entrainement de 53min 40s.

Conclusion

Les couches convolutionnelles sont les couches où les filtres sont appliqués à l'image originale, ou à d'autres cartes de caractéristiques dans un CNN profond. C'est là que se trouvent la plupart des paramètres spécifiés par l'utilisateur dans le réseau. Les paramètres les plus importants sont le nombre de noyaux et la taille des noyaux.

Maxpooling est une opération de mise en commun qui sélectionne l'élément maximal de la région de la carte de caractéristiques couverte par le filtre. Ainsi, la sortie après la couche de mise en commun maximale sera une carte de caractéristiques contenant les caractéristiques les plus importantes de la carte de caractéristiques précédente.

Dans les CNN, la taille des filtres de convolution détermine la taille du champ réceptif où l'information est extraite. Les méthodes basées sur les CNN utilisent des tailles de filtres prédéfinies et fixes dans chaque couche convolutionnelle, ce qui est appelé ci-après le CNN traditionnel.

Chapitre III. Implémentation du programme

INTRODUCTION

Si les matériaux utilisés ne sont pas recyclés, de nouveaux produits sont fabriqués en extrayant des matières premières fraîches de la Terre, par l'exploitation minière et forestière. Le recyclage aide à conserver les matières premières importantes et protège les habitats naturels pour l'avenir.

Dans ce chapitre nous avant parlé de la machine RVM, en à définir notre dataset, on a travaillé avec les bibliothèques Tensorflow et Keras pour l'apprentissage et la classification et en a appliqué le model que n'obtient au temps réel.

Recyclage

DEFINITION

Le recyclage est un processus de traitement des déchets et de réintroduction des matériaux résultants dans le cycle de production d'autres produits équivalents ou différents. **[11]**

Le recyclage permet de réduire les volumes de déchets, et donc leur pollution, et de préserver les ressources naturelles en réutilisant des matières premières déjà extraites.

Type de recyclage :

Il existe trois formes de recyclage : chimique : produire une réaction chimique pour, par exemple, séparer les composants ; mécanique : utiliser une machine pour transformer les déchets (broyer, par exemple) ; organique : produire un engrais ou un carburant.

Avantage de recyclage :

Recycler signifie recueillir des matériaux usagés et les transformer en nouveaux produits plutôt que de les jeter. Ce processus est bénéfique pour l'environnement et nos communautés locales pour les raisons suivantes :

- Réduction de la quantité de déchets qui finissent dans les centres d'enfouissement et les usines d'incinération.
- Utilisation d'une quantité beaucoup plus faible d'énergie.
- Réduction des émissions de carbone.
- Diminution de la pollution de l'eau et de l'air.
- Préservation des ressources naturelles précieuses, dont les arbres et l'eau.
- Protection de la faune et des habitats naturels.
- Création d'emplois et soutien de l'économie.

LE CYCLE DU RECYCLAGE DES BOUTEILLES PLASTIQUES

Ce cycle se compose de trois étape :

Avantage de recyclage :

Recycler signifie recueillir des matériaux usagés et les transformer en nouveaux produits plutôt que de les jeter. Ce processus est bénéfique pour l'environnement et nos communautés locales pour les raisons suivantes :

- Réduction de la quantité de déchets qui finissent dans les centres d'enfouissement et les usines d'incinération.
- Utilisation d'une quantité beaucoup plus faible d'énergie.
- Réduction des émissions de carbone.
- Diminution de la pollution de l'eau et de l'air.
- Préservation des ressources naturelles précieuses, dont les arbres et l'eau.
- Protection de la faune et des habitats naturels.
- Création d'emplois et soutien de l'économie.

❖ ETAPE 1 : LA DISTRIBUTION

Entre 6 et 8 milliards de bouteilles en plastiques sont achetées en Algérie chaque année. Elles sont disponibles dans tous les points de vente.



FIGURE III.1 :bouteilles

❖ ETAPE 2 : LA COLLECTE

Après l'utilisation, pensez à recycler vos bouteilles en plastiques ! Pour que la récupération des bouteilles plastiques soit faites, il suffit de les déposer dans des dispositifs spécifie (bac jaune, bleu, machine a collection sac, …etc.) .



FIGURE III.2 :BAC



FIGURE III.3 :machine de collection

❖ ETAPE 3 : LE TRI

Après la collecte, Les bouteilles en plastique en PET clair, issues de la récupération des bouteilles en plastiques, seront séparées du reste de la collecte et envoyées par un camion au usine Recyclage de Plastiques pour y être régénérées en granulés.

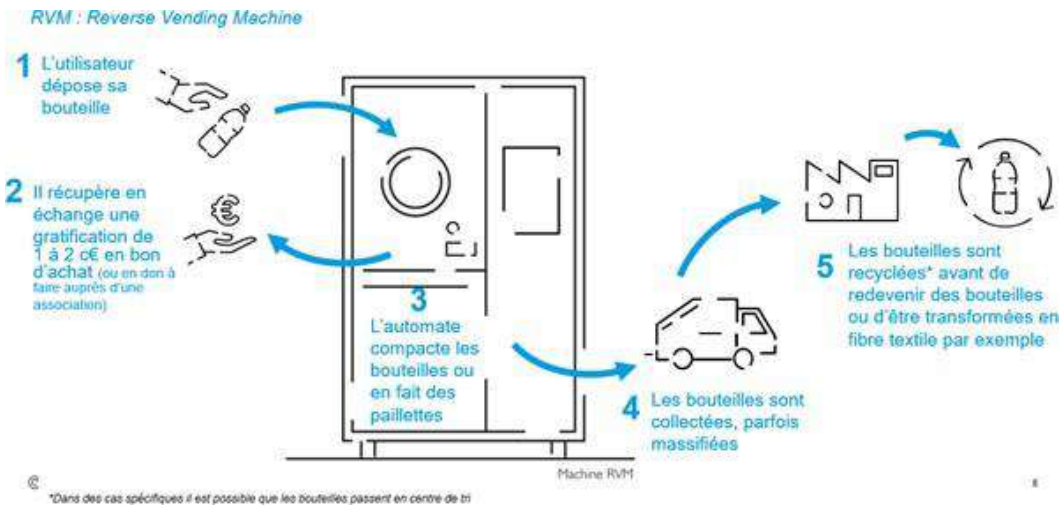


FIGURE III.4 :schéma de recyclage des bouteilles

La machine RVM

Les Reverse Vending Machines (RVM) automatisent le processus de collecte, de reconnaissance, de tri et de stockage des bouteilles en plastique participants dans un système de Deposit Return



FIGURE III.5 :machine RVM

Fonctionnement de la machine RVM

1. Le consommateur achète une boisson

2. Le conteneur vide est retourné à un Reverse Vending Machine

3. Le consommateur obtient une récompense instantanée

Implémentation du système

On a pris l'idée de notre machine du principe de fonctionnement de la machine RVM que ce trouve uniquement à l'Europe plus précisément en suisse, a fin de réalisée notre projet en a travaillé avec python et ses bibliothèques (tansorflow, keras, numpy, os, cv2…)

Python

Le langage Python est un langage de programmation open source multiplateformes et orienté objet. Grâce à des bibliothèques spécialisées, Python s'utilise pour de nombreuses situations comme machine learning, le développement logiciel, l'analyse de données, ou la gestion d'infrastructures. Il n'est donc pas, comme le langage HTML par exemple, uniquement dédié à la programmation web. **[12]**

TansorFlow

TensorFlow est une bibliothèque open source, permettant d'exécuter des applications de machine learning et de deep learning. Cet outil dédié à l'apprentissage automatique a été développé par Google, et est fortement utilisé dans le domaine de l'intelligence artificielle (IA). Ainsi, des professionnels comme des novices peuvent créer des modèles de machine learning ou de deep learning pour optimiser les capacités de leur matériel. **[13]**

Keras

Keras est une bibliothèque open source écrite en Python basée principalement sur les travaux du développeur de Google François Chollet dans le cadre du projet ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System). Une première version du logiciel multiplateforme a été publiée le 28 mars 2015. Le but de cette bibliothèque est de permettre la constitution rapide de réseaux neuronaux. Dans ce cadre, Keras ne fonctionne pas comme un framework propre mais comme une interface de programmation applicative (API) pour l'accès et la programmation de différents frameworks d'apprentissage automatique. Theano, Microsoft Cognitive Toolkit (anciennement CNTK) et TensorFlow font notamment partie des frameworks pris en charge par Keras. **[14]**

La structure de notre programme

- **Création de dataset**

En a créé trois classes de trois déférents types de bouteilles avec 20% des images pour la validation et 80% pour l'entrainement

- **L'implémentation des bibliothèques utilisé**

```
from tensorflow import keras
from tensorflow.keras.optimizers import Adam
from keras.models import Sequential
from keras.utils.np_utils import to_categorical
from keras.layers import Conv2D,MaxPooling2D,Flatten,Dense,Dropout
import matplotlib.pyplot as plt
import tensorflow as tf

from keras import models
import os

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder

from tensorflow.keras.models import load_model

from tensorflow.keras.utils import img_to_array
from keras.preprocessing import image
from keras.models import load_model

import numpy as np
import cv2
```

FIGURE III.6 :importation des bibliothèques

- **Création de labels**

Dans cette partie nous avant crée un dossier pour chaque image contient le même nom de classe

```
images_paths=[]
class_labels=[]
for folder in class_paths:
    folder_path=os.path.join(base_path, folder)
    for files in os.listdir(folder_path):
        images_paths.append(os.path.join(folder_path, files))
        class_labels.append(folder)
```

FIGURE III.7 :création des labels

- **Création du model**

Notre model ce compose de trois couche de convolution de taille 3x3
avec trois couche de maxpooling et une fonction d'activation relu et sigmoid

```
model=Sequential()
model.add(Conv2D(32,kernel_size=3,strides=1,padding="Same",activation="relu",input_shape=(image_size,image_size,3)))
model.add(Conv2D(32,kernel_size=3,strides=1,padding="Same",activation="relu"))
model.add(Conv2D(32,kernel_size=3,strides=1,padding="Same",activation="relu"))
model.add(MaxPooling2D(padding="same"))
model.add(Dropout(0.25))

model.add(Conv2D(64,kernel_size=3,strides=1,padding="Same",activation="relu"))
model.add(Conv2D(64,kernel_size=3,strides=1,padding="Same",activation="relu"))
model.add(Conv2D(64,kernel_size=3,strides=1,padding="Same",activation="relu"))
model.add(MaxPooling2D(padding="same"))


model.add(Conv2D(256,kernel_size=3,strides=1,padding="Same",activation="relu"))
model.add(MaxPooling2D(padding="same"))
model.add(Dropout(0.4))

model.add(Flatten())

model.add(Dense(64,activation="relu"))
model.add(Dropout(0.5))
model.add(Dense(n_classes,activation="sigmoid"))
```

FIGURE III.8 :Model

- **Compilation et sommaire**

  Nous avant utilise un optimisateur Adam et une loss pour un multi classification (categorical_crossentropy)

```
model.compile(optimizer="adam",loss="categorical_crossentropy",metrics=["acc"])
```

```
model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_7 (Conv2D)            (None, 128, 128, 32)      896

conv2d_8 (Conv2D)            (None, 128, 128, 32)      9248

conv2d_9 (Conv2D)            (None, 128, 128, 32)      9248

max_pooling2d_3 (MaxPooling  (None, 64, 64, 32)        0
2D)

dropout_3 (Dropout)          (None, 64, 64, 32)        0

conv2d_10 (Conv2D)           (None, 64, 64, 64)        18496

conv2d_11 (Conv2D)           (None, 64, 64, 64)        36928

conv2d_12 (Conv2D)           (None, 64, 64, 64)        36928

max_pooling2d_4 (MaxPooling  (None, 32, 32, 64)        0
2D)

conv2d_13 (Conv2D)           (None, 32, 32, 256)       147712

max_pooling2d_5 (MaxPooling  (None, 16, 16, 256)       0
2D)

dropout_4 (Dropout)          (None, 16, 16, 256)       0

flatten_1 (Flatten)          (None, 65536)             0

dense_2 (Dense)              (None, 64)                4194368

dropout_5 (Dropout)          (None, 64)                0

dense_3 (Dense)              (None, 3)                 195

=================================================================
Total params: 4,454,019
Trainable params: 4,454,019
Non-trainable params: 0
_________________________________________________________________
```

FIGURE III.9 :compilation et sommaire

- **En a créé une colonne binaire pour chaque catégorie et renvoie une matrice clairsemée ou un tableau dense**

```
ohe=OneHotEncoder()
class_labels=np.array(class_labels).reshape(-1, 1)
ohe.fit(class_labels)
class_labels_ohe=ohe.transform(class_labels).toarray()
```

```
class_labels_ohe
```

```
array([[1., 0., 0.],
       [1., 0., 0.],
       [1., 0., 0.],
       ...,
       [0., 0., 1.],
       [0., 0., 1.],
       [0., 0., 1.]])
```

FIGURE III.10 :transformer les catégories en numérations

- **Divisions des images on x train, y train, x test et y test**

  Nous avant divisé les images en 30% test et 70% train

```
x_train,x_test,y_train,y_test=train_test_split(images_paths,class_labels_ohe,test_size=0.3)
```

```
x_train_images=[]
for p in x_train:
    x_train_images.append(load_images(p))

x_test_images=[]
for p in x_test:
    x_test_images.append(load_images(p))

x_train_images=np.array(x_train_images)
x_test_images=np.array(x_test_images)
```

FIGURE III.11 :division des images

- **Entrainement du modèle**

  Nous avant entrainé notre model 10 fois (10 epochs)

  Et nous avant obtenez une accuracy de 94.17% et de loss de 26.67%

```
history=model.fit(x_train_images, y_train, epochs=10, validation_data=(x_test_images,y_test))
```

```
Epoch 1/10
27/27 [==============================] - 167s 6s/step - loss: 6.0026 - acc: 0.3488 - val_loss: 1.0889 - val_acc: 0.3583
Epoch 2/10
27/27 [==============================] - 166s 6s/step - loss: 1.0736 - acc: 0.3905 - val_loss: 0.8246 - val_acc: 0.6222
Epoch 3/10
27/27 [==============================] - 150s 6s/step - loss: 0.8143 - acc: 0.5714 - val_loss: 0.7204 - val_acc: 0.7167
Epoch 4/10
27/27 [==============================] - 151s 6s/step - loss: 0.7679 - acc: 0.6095 - val_loss: 0.5573 - val_acc: 0.8306
Epoch 5/10
27/27 [==============================] - 163s 6s/step - loss: 0.6624 - acc: 0.7107 - val_loss: 0.5413 - val_acc: 0.8000
Epoch 6/10
27/27 [==============================] - 151s 6s/step - loss: 0.4743 - acc: 0.8083 - val_loss: 0.3300 - val_acc: 0.9000
Epoch 7/10
27/27 [==============================] - 145s 5s/step - loss: 0.5096 - acc: 0.8321 - val_loss: 0.3101 - val_acc: 0.9278
Epoch 8/10
27/27 [==============================] - 167s 6s/step - loss: 0.4168 - acc: 0.8369 - val_loss: 0.1894 - val_acc: 0.9722
Epoch 9/10
27/27 [==============================] - 162s 6s/step - loss: 0.3768 - acc: 0.8750 - val_loss: 0.2063 - val_acc: 0.9667
Epoch 10/10
27/27 [==============================] - 187s 7s/step - loss: 0.3245 - acc: 0.8798 - val_loss: 0.2667 - val_acc: 0.9417
```

FIGURE III.12 : les résultat de loss et de l'accuracy

- **Les résultats d'entrainement**

  Nous avant obtenez les résultats des pertes de l'entrainement de notre model en fonction des epoches.

  Nous avant trouvé que la courbe de l'entrainement est décroissant par rapport aux epoches et ça s'est bien pour le model.

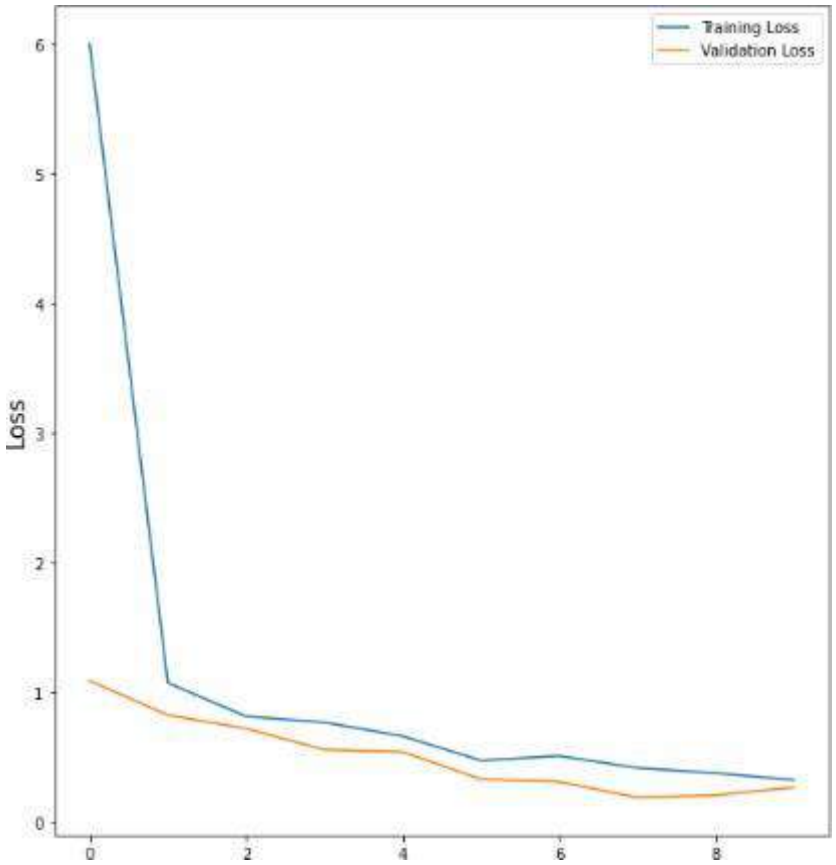


FIGURE III.13 :la courbe des pertes en fonction des epoches

Nous avant obtenez les résultats de la précision de l'entrainement de notre model en fonction des epoches.

Nous avant trouvé que la courbe de l'entrainement est croissant par rapport aux epoches et ça s'est bien pour le model

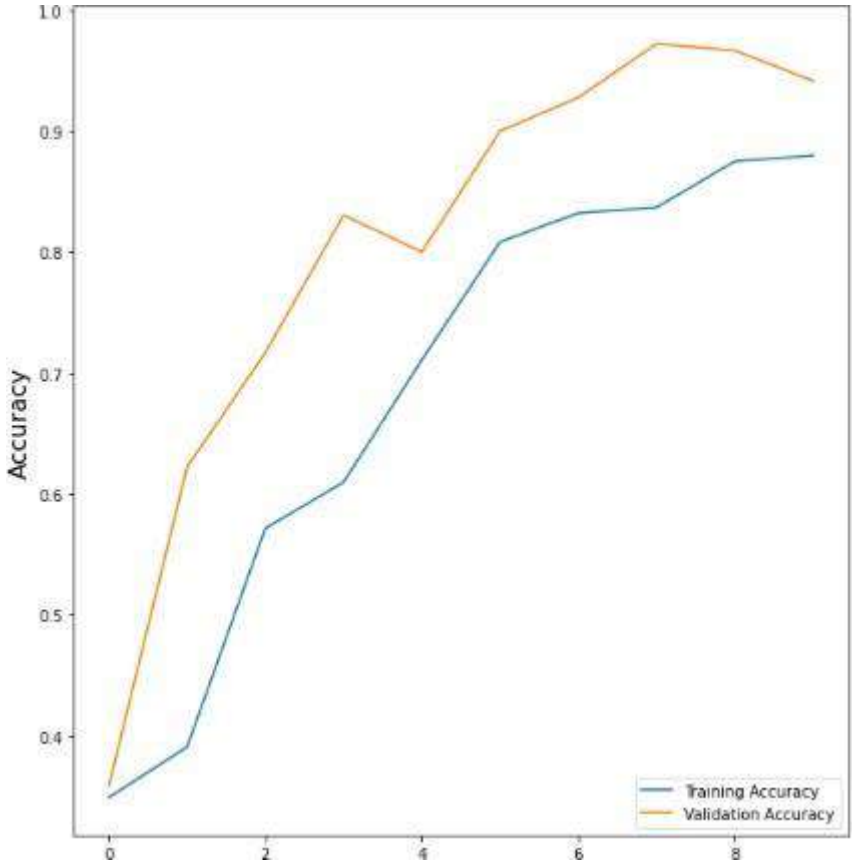


FIGURE III.14 : la courbe de la precision en fonction des epoches

- **Résultat de prédiction**
  Voici quelque résultat de la prédiction notre classification

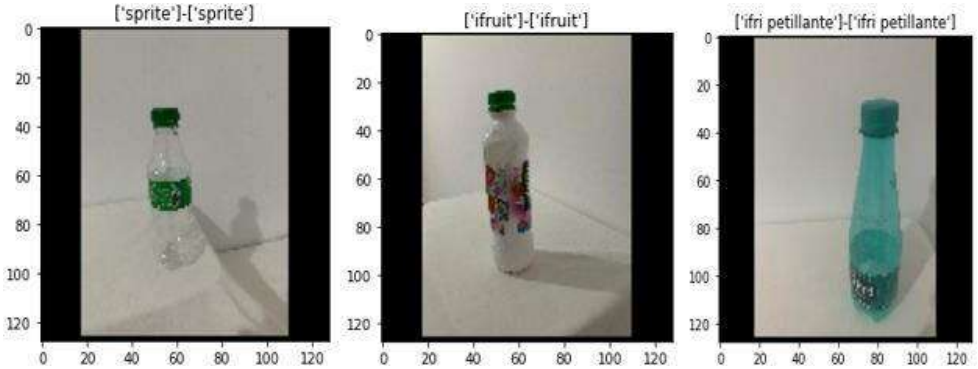


FIGURE III.15 : résultats de prédictions

- **Implémentation de camera au temps réel**

  Nous avant appliqué notre modèle au temps réel à travers une camera

```
cap = cv2.VideoCapture(0)

while(True):
    ret, frame = cap.read()
    #gray = cv2.cvtColor(frame, cv2.COLOR_2BGR)
    gray=cv2.resize(frame,(128,128),interpolation=cv2.INTER_AREA)
    R=gray
    #.astype('float32')/255
    R=img_to_array(R)
    R=np.expand_dims(R,axis=0)
    pred=classifier.predict(R)[0]
    label=class_labels[np.argmax(pred)]
    cv2.putText(frame,label,(10,50),cv2.FONT_HERSHEY_SIMPLEX,1,(255,5,116),2)
    cv2.imshow('gray',frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

FIGURE III.16 : le code de camera

- **Les résultats de classification au temps real**



FIGURE III.17 : résultats de prédictions au tempe réel

Conclusion :

Nous avant interpréter juste le côté de classification on parce que on n'a pas eu le temps pour réaliser le cotée de récompense et de réaliser complètement de la machine

A la fin de ce chapitre en a pue détecte nos bouteilles et les classifier au temps réel à l'aide d'un réseau de neurone convolutionelle et une data set spécifier qu'en a créé, tout ça est réalisé sous Python et ses bibliothèques.

Conclusion générale

Le recyclage est une habitude que chacun entre nous dois le respecter et le planter dans l'inconscience de l'être humain dès qu'il est nés et on n'arrive pas à trouver une meilleure solution qui accomplis cette mission que leur donner de l'argent a chaque fois qu'il recycle une bouteille C'est pour cette raison en a choisi ce projet.

On a pu savoir l'apprentissage automatique et le deeplearning et le réseau de neurone et leurs influences sur l'intelligence artificiel et le machine learning, et on a pu de maitriser le CNN
Qui grâce a lui on a réalisé là le modèle de classification des bouteilles.

Pour des raisons de temps et de manque de matérielle on n'a pas pu réaliser le modèle et la machine en générale, nous avant juste interpréter le cote de la classification et en négligeant le cote de récompense en espion que on y arrivera à finaliser ce projet et le mis en marche.

Références Bibliographiques

• [1] Trevor Hastie, Robert Tibshirani et Jerome Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2009, 2e éd.

• [4] Wesley Chai, A Timeline of Machine Learning History, 2020.

•[6] Huang T.-M., Kecman V., Kopriva I. (2006), Kernel Based Algorithms for Mining Huge Data Sets, Supervised, Semi-supervised, and Unsupervised Learning, Springer-Verlag, Berlin, Heidelberg.

•[7] Vincent Barra, Antoine Cornuéjols, Laurent Miclet, Apprentissage Artificiel : Concepts et algorithmes, Eyrolles, 2021

• [8] Antoine Krajnc, Les applications du Deep Learning, 2018.

•[9] By JAMES CHEN, neural network, 2021

•[10] Fabien Tschirhart (dir. Alain Lioret), Réseaux de neurones formels appliqués à l'Intelligence Artificielle et au jeu, ESGI (mémoire de master de recherche en multimédia et animation numérique).

•[11] Fédération professionnelle des entreprises du recyclage, Ademe, Évaluation environnementale du recyclage en France selon la méthodologie de l'analyse de cycle de vie, avril 2017.

•[12] « fazibear/tensorflow.cr » [archive], sur GitHub

•[13] « Module: tf.keras | TensorFlow » [archive], sur TensorFlow.org

•[14] « Keras Documentation » [archive], sur keras.io