

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE MOULOUD MAMMARI DE TIZI-OUZOU



FACULTE DU GENIE ELECTRIQUE ET D' INFORMATIQUE
DEPARTEMENT D'ELECTRONIQUE

Mémoire de Fin d'Etudes de MASTER ACADEMIQUE

Domaine : Sciences et Technologies

Filière : ELECTRONIQUE

Spécialité : **Electronique Des Systèmes
Embarqués**

Présenté par

Kenza AMOUDACHE

Amal AMIAR

Thème

**Conception et Mise en Œuvre d'un Système
Embarqué de Reconnaissance d'Objets et
d'Estimation de la Distance Basée sur le Traitement
d'Images avec une Pi Camera et une Raspberry Pi 4**

Mémoire soutenu publiquement le 30/06/ 2024 devant le jury composé de :

Mr Mourad LAGHROUCHE

Professeur, UMMTO, Examineur

Mr Massine GANA

MA, UMMTO, Encadreur

Mr Boussad IDJRI

MCB, UMMTO, Président

L'année universitaire:
2023/2024

Table des matières

Introduction générale	9
1 Étude approfondie de la détection d'objets avec implémentation d'IA	10
1.1 Prépambule	11
1.2 Introduction au traitement d'image	11
1.2.1 image numérique	11
1.2.2 traitement d'images	12
1.2.3 Les différentes étapes de traitement des images numériques	12
1.3 Handicape visuelle	14
1.3.1 l'anatomie de l'œil	14
1.3.2 La déficience visuelle	14
1.3.3 Causes de déficience visuelle	15
1.3.4 Les impacts physiques de la malvoyance	15
1.4 Méthodes traditionnelles de détection d'objets	17
1.4.1 Histogrammes de gradients orientés (HOG)	17
1.4.2 Méthodes de détection basées sur les cascades	18
1.5 Techniques avancées de détection d'objets basées sur l'apprentissage profond	19
1.5.1 Réseaux de neurones convolutifs (CNN)	19
1.5.2 Région-based Convolutional Neural Networks (R-CNN)	20
1.5.3 You Only Look Once (YOLO)	21
1.5.4 Single Shot MultiBox Detector (SSD)	22
1.6 implication de la vision artificielle dans l'amélioration d'accessibilité visuelle des malvoyants	24
1.6.1 Reconnaissance d'objets	24
1.6.2 Détection des obstacles	24
1.6.3 Reconnaissance des personnes	25
1.7 Application de la détection d'objets dans le contexte des systèmes embarqués	25
1.7.1 Systèmes de surveillance	25

1.7.2	Robots autonomes	25
1.7.3	Systèmes de détection de collision	26
1.7.4	Système de détection d'objets et de personnes pour les malvoyants	27
1.7.5	Systèmes de surveillance de la santé	27
1.7.6	Systèmes de navigation intelligente	28
1.8	Limitations et défis actuels dans la reconnaissance d'objets pour les personnes malvoyantes	28
1.8.1	Limitations actuelles	28
1.8.2	Défis actuels	29
1.8.3	Perspectives d'avenir	29
1.9	Discussion	30
2	Étude approfondie de la détection d'objets avec implémentation d'IA	31
2.1	Préambule	32
2.2	Fondements de l'apprentissage en profondeur pour la détection d'objets	32
2.2.1	Définition de l'apprentissage en profondeur	32
2.2.2	Classification des algorithmes d'apprentissage	34
	Apprentissage supervisé	34
	Apprentissage non supervisé	34
2.2.3	l'utilité de l'apprentissage profond	35
2.2.4	Les étapes de la détection d'objets en utilisant l'apprentissage en profondeur	35
2.3	Architecture des réseaux de neurones convolutifs pour la détection d'objets	36
2.3.1	Réseau neuronal	36
2.3.2	architecture de réseau de neurone	36
2.3.3	Réseaux neuronaux convolutifs	37
2.3.4	L'architecture d'un CNN	38
2.3.5	Avantages d'un CNN dans le domaine de la reconnaissance d'images	42
2.4	Méthodes de prétraitement et d'augmentation des données	43
2.4.1	Prétraitement des données	43
2.4.2	Augmentation des données	44
2.5	Entraînement du modèle de détection d'objets	46
2.6	Utilisation du langage Python dans le développement de l'IA	47
2.6.1	Introduction à Python pour le développement d'IA	47
2.6.2	Bibliothèques Python populaires pour l'apprentissage en profondeur	47

2.6.3	Exemple de code Python pour l'entraînement et l'évaluation de modèles de détection d'objets	50
2.7	Discussion	52
3	Réalisation du projet utilisant une carte Raspberry Pi 4	53
3.1	préambule	54
3.2	Présentation de la carte Raspberry Pi 4 et de ses capacités	54
3.2.1	Définition et présentation de la Raspberry Pi	54
3.2.2	Ses caractéristiques techniques	55
3.2.3	Caméra Pi	57
3.3	Configuration de l'environnement de développement	58
3.3.1	Installation de système d'exploitation linux	58
3.3.2	Accès et contrôle à distance avec PuTTY et Real	60
3.3.3	Processus d'analyse visuelle et d'identification d'objets	63
3.3.4	Paquets et bibliothèques installé	64
3.4	Intégration du modèle de détection d'objets sur la Raspberry Pi 4	65
3.4.1	Intégration du modèle SSD	65
3.4.2	Estimation de la distance des objets	67
3.4.3	Les catégorie des objets détectés par le système	68
3.5	Tests et évaluation du système final	68
3.5.1	résultat	68
3.5.2	discussion des resultats obtenus	69
3.6	Étude des performances, les limitations et les perspectives d'amélioration	70
3.6.1	Étude des performances	70
3.6.2	Limitations	71
3.6.3	Perspectives d'amélioration	71
3.7	Discussion	72
	Conclusion générale	73

Table des figures

1.1 schéma de différentes étapes de traitement d image numérique	12
1.2 schéma en coupe de l'œil humain	14
1.3 Visualisation de l'impact physiologique de la vision centrale	16
1.4 visualisation de l'impact physiologique de la vision périphérique	17
1.5 Architecture de HOG	18
1.6 Schéma de la méthode Viola-Jones	19
1.7 Architecture de CNN	20
1.8 Architecture de R-CNN	21
1.9 Architecture de YOLO	22
1.10 Architecture de SSD	23
2.1 L'intelligence artificielle et ses sous-domaines	33
2.2 les différents types d'apprentissage automatique	34
2.3 Réseau neuronal	36
2.4 Architecture de réseau de neurone	37
2.5 architecture de base d'un CNN	38
2.6 Application de coche de la convolution	39
2.7 Application de la fonction d'activation ReLU	40
2.8 Application de Max Pooling	40
2.9 Application de Max Pooling	41
2.10 La fonction d'activation softmax	42
2.11 logo python	47
2.12 logo TensorFlow	48
2.13 logo Keras	48
2.14 Logo PyTorch	49
2.15 Logo Open CV	49
2.16 Logo NumPy	50
2.17 code du réseau de neurones convolutif	50

2.18 code de manipulation des images	51
3.1 Vu de face d'une raspberry	55
3.2 vu d'arriere d'une raspberry	56
3.3 raspberry pi camera module 2	57
3.4 Interface d'utilisateur graphique de raspberry pi image	59
3.5 Interface d'utilisateur pour la personnalisation du systeme d'exploitation . .	60
3.6 Capture de l'interface PuTTY	61
3.7 Interface Real VNC Viewer	62
3.8 Branchement de connexion d'une Raspberry Pi 4 à un ordinateur portable .	63
3.9 Schéma de fonctionnement	64
3.10 Capteur de Resultat détecter par notre système pour deux téléphones avec une distance différente	69
3.11 une autre capteur de détection d'objet déférentes	69
3.12 resultat de détection d'un sac ados avec une distance estimer	70

Liste des abréviations

2D : deuxième dimension
3D :troisième dimension
DMLA : Dégénérescence maculaire liée à l'âge
HOG : Histogrammes de gradients orientés
CNN : Réseaux de neurones convolutifs
R-CNN : Région-basedConvolutional Neural Networks
YOLO : You Only Look Once
SSD : Single ShotMultiBox Detector
ML : Machine Learning
IA : Intelligence artificielle
DL : Deep Learning
MLP : pour Multi Layer Perceptron
CONV : couche de convolution
ReLU : Rectified Linear Unit
ELU : Exponential Linear Unit
FC : Fully connected
MNIST : Modified National Institute of Standards and Technology
NumPy : Numerical Python
ARM : Advanced RISC Machine
RAM : Random Access Memory
Wi-Fi : Wireless Fidelity
IoT : internet of Things
CSI : Camera Serial Interface
USB : Universal Serial Bus
MIPI : Mobile Industry Processor Interface
SD : Secure Digital

OS : Operating System

SSH : Secure Shell

VNC : Virtual Network Computing

Remerciement

Je remercie tout d'abord, le bon Dieu tout puissant de nous avoir donné la santé, la volonté, la force, le courage,

Je remercie et la patience pour pouvoir surmonter les moments difficiles, et atteindre nos objectifs et sans lesquels notre projet n'aurait pas pu voir la lumière de ce jour.

Je remercie Mr Gana Massine d'avoir accepté de nous encadrer

je tien également à adresser toute ma graditude aux memmbre de jury de soutenance d'avoir acceptant d'examiner notre travail et l'enrichir par leurs propositions. Je tiens à remercier ma famille pour l'inspiration, l'aide et le temps qu'ils ont bien voulu nous consacrer et sans eux ce mémoire n'aurait jamais vu le jour.

Dédicaces

Je dédie ce modeste travail

À mes parents :

Ma mère Zazi, mon inspiration, mon soutien, ma confidente, ma force, mon espoir, ma positivité. Pour toutes ses prières et ses sacrifices consentis tout au long de mes études, pour apaiser mes craintes dans les moments de doute. Reçois à travers ce modeste travail l'expression de mes sentiments et de mon éternelle gratitude pour me donner le sourire dans les moments de peine.

Mon père Ahmed qui a su me conseiller, Pour tous ses précieux conseils, pour toute son assistance et sa présence dans ma vie. Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.

Que DIEU leurs procure bonne santé et longue vie.

à mes chères sœurs et frères, qui ont toujours été à mes côtés, m'inspirant et me soutenant tout au long de ce parcours académique. Leurs encouragements, leur amour et leur présence ont été une source inestimable de motivation. Que ce travail leur soit dédié en signe de gratitude et d'affection. à toute ma famille

à mes chères amies pour leurs encouragements en toutes circonstances.

Mr S.Essaid qui me soutien inestimable tout au long de la rédaction de mon mémoire. Ses conseils avisés, son écoute attentive et son encouragement constant m'ont été d'une aide précieuse pour mener à bien ce projet.

Amal Amiar

Dédicace

Je dédie ce modeste mémoire de fin de master à mes chers parents qui ont été à mes côtés pour me soutenir et m'encourager le long de mon cursus.

A ma chère sœur Lynda et mon petit frère Yanis qui ont été toujours à mes côtés; et à qui je souhaite un avenir plein de réussite.

A mes enseignants et encadrants pour leurs conseils avisés, leur patience et leur dévouement, qui nous ont guidés vers la réussite de ce travail.

A mon cher partenaire Salim qui m'a indiqué la bonne voie et les déterminés finiront toujours par réussir leur vie.

A tous mes proches qui m'ont apporté leurs sollicitudes pour accomplir ce travail surtout mes grands-parents, mes oncles, mes tante, mes cousins et mes cousines. Sans oublier tous mes amis et l'ensemble des étudiants de ma promotion.

AMOUDACHE Kenza

Resumé

Ce projet de recherche se concentre sur le développement et l'évaluation d'un système innovant de détection d'objets couplée à une estimation de distance, conçu spécifiquement pour améliorer l'accessibilité des informations visuelles aux personnes malvoyantes et aveugles. La solution proposée repose sur une plateforme Raspberry Pi équipée d'une caméra Raspberry Pi, offrant une architecture compacte et abordable pour la mise en œuvre de ce type de technologies d'assistance. Les algorithmes de détection d'objets s'appuient sur des techniques d'apprentissage automatique avancées, notamment le deep learning, permettant une identification précise des éléments présents dans l'environnement visuel. Une interface utilisateur dédiée a été développée, intégrant les résultats de la détection d'objets ainsi que les mesures de distance correspondantes ; conçue pour être accessible aux utilisateurs malvoyants, cette interface propose une restitution à la fois visuelle et sonore, favorisant ainsi une meilleure compréhension de l'environnement. L'implémentation d'une connexion VNC (Virtual Network Computing) permet à l'utilisateur de contrôler à distance le système et d'accéder aisément aux informations, soutenant ainsi son autonomie et son inclusion dans les activités de la vie quotidienne.

Abstract

This research project focuses on the development and evaluation of an innovative object detection system coupled with distance estimation, specifically designed to improve the accessibility of visual information for visually impaired and blind individuals. The proposed solution is based on a Raspberry Pi platform equipped with a Raspberry Pi camera, offering a compact and affordable architecture for the implementation of this type of assistive technology. The object detection algorithms are based on advanced machine learning techniques, particularly deep learning, enabling precise identification of elements present in the visual environment. A dedicated user interface has been developed, integrating the results of object detection as well as the corresponding distance measurements; designed to be accessible to visually impaired users, this interface provides both visual and auditory feedback, thus fostering a better understanding of the environment. The implementation of a VNC (Virtual Network Computing) connection allows the user to remotely control the system and easily access the information, supporting their autonomy and inclusion in daily life activities.

Introduction générale

Selon l'Organisation mondiale de la Santé (OMS), au moins 2,2 milliards de personnes souffrent de déficience visuelle modérée à grave à travers le monde, dont 15 pour-cent sont totalement aveugles. Bien que de nombreuses maladies entraînant la perte de vision puissent être évitées, la déficience visuelle continue de croître, notamment en raison de l'augmentation de l'espérance de vie. Ce phénomène touche particulièrement les populations âgées, mais également les enfants dans les pays en développement où le manque de diagnostic précoce empêche la prévention de nombreux cas [1].

Dans un monde dans lequel l'information visuelle est omniprésente et joue un rôle crucial dans de nombreux aspects de la vie quotidienne, il est essentiel de développer des technologies qui permettent à tous, y compris aux personnes malvoyantes, d'accéder et de comprendre ces informations. Le traitement d'image et la détection d'objets sont des domaines technologiques en pleine expansion qui offrent des solutions innovantes pour relever ce défi[2].

Ce mémoire vise à explorer en détail les domaines du traitement d'image et de la détection d'objets, en mettant l'accent sur les applications pratiques et les avancées récentes. L'objectif principal de ce travail de recherche est de concevoir et de mettre en œuvre un système avancé de détection d'objets en temps réel, capable d'estimer la distance des objets détectés. Cette solution technologique vise à améliorer l'autonomie et la sécurité des personnes malvoyantes en leur fournissant des informations visuelles précises sur leur environnement. En conclusion, ce mémoire de fin d'études propose une approche complète pour la détection d'objets et le traitement d'image, en mettant en avant l'importance de l'accessibilité des informations visuelles pour les personnes malvoyantes. En combinant les avancées théoriques et les applications pratiques, ce travail de recherche contribue à l'avancement des technologies inclusives qui favorisent l'autonomie et l'intégration des personnes malvoyantes dans notre société axée sur l'information visuelle. Notre mémoire est organisé en trois chapitres comme suit :

- **Chapitre 1** : État de l'art sur le traitement d'images et les techniques de détection d'objets

Le premier chapitre présente une analyse détaillée des différentes méthodes de traitement d'images existantes, en soulignant leur importance pour rendre les images accessibles aux

personnes malvoyantes. Il explore également les étapes fondamentales du traitement d'images numériques, telles que l'acquisition, l'amélioration et la segmentation des images.

- **Chapitre 2** : Étude approfondie de la détection d'objets avec implémentation d'IA

Le deuxième chapitre se penche sur les techniques avancées de détection d'objets, en mettant particulièrement l'accent sur l'utilisation de l'intelligence artificielle. Cette section examine les algorithmes et les architectures de réseaux neuronaux convolutifs, qui permettent d'améliorer la précision et l'efficacité de la détection d'objets dans divers contextes.

- **Chapitre 3** : Réalisation du projet utilisant une carte Raspberry Pi 4

Le troisième chapitre est consacré à la mise en pratique des concepts théoriques abordés dans les chapitres précédents. Il détaille la conception et l'implémentation d'un système de détection d'objets utilisant une caméra et une carte Raspberry. Ce système vise à améliorer l'autonomie et la sécurité des personnes malvoyantes en leur permettant de détecter et d'identifier les objets environnants, ainsi que d'estimer leur distance.

Chapitre 1

Étude approfondie de la détection d'objets avec implémentation d'IA

1.1 Préambule

Le développement de techniques visant à rendre les images accessibles aux personnes malvoyantes est un enjeu sociétal majeur dans notre monde moderne où l'information visuelle prend une place prépondérante. En effet, les images sont omniprésentes dans des domaines aussi variés que l'éducation, les médias, la communication ou encore les loisirs, et constituent une source d'information et de savoir essentielle. Les déficiences visuelles peuvent entraîner des difficultés à interpréter et comprendre pleinement le contenu et les informations contenues dans les images. C'est pourquoi le traitement d'images pour les personnes malvoyantes représente un champ de recherche en constante évolution, visant à développer des techniques permettant de rendre les images accessibles à ces publics. L'objectif final est de favoriser leur autonomie, leur participation sociale et leur accès à l'information, en leur donnant les moyens de percevoir et appréhender le sens des images au même titre que les personnes voyantes. Ce chapitre a pour objet d'analyser les différentes méthodes de traitement d'images existantes, leurs apports et limites respectifs, afin de mettre en lumière le défi actuel et futures pistes d'amélioration pour renforcer l'inclusion des personnes malvoyantes dans notre société de l'image.

1.2 Introduction au traitement d'image

1.2.1 image numérique

Une image est une représentation planaire d'un objet quelconque. C'est un ensemble de pixels, chaque pixel ayant une position et une valeur spécifiques. Ces pixels sont les éléments de base d'une image numérique et sont organisés en une grille 2D, où chaque pixel contient une valeur numérique représentant la couleur ou le niveau de luminosité de ce pixel[5].

Différentes catégories d'image

- **Images morphologiques** : formes des objets et leur disposition en 2D ou 3D.
- **Image volumique** : rendu volumique d'un objet ou d'une scène.
- **Images fonctionnelles ou paramétriques** : comportement d'un phénomène physique.
- **Images dynamiques** : séquence d'images en fonction du temps (vidéo).
- **Images multispectrales** : obtenues dans différentes bandes de fréquences du rayonnement de la source [4].

1.2.2 traitement d'images

Le traitement d'images est un domaine qui consiste en l'étude des images et de leurs transformations. Le traitement d'images est l'ensemble des opérations effectuées sur l'image, afin d'en améliorer la lisibilité et d'en faciliter l'interprétation [30].

1.2.3 Les différentes étapes de traitement des images numériques

Le traitement d'image numérique est un processus structuré en plusieurs étapes clés, depuis l'acquisition de l'image jusqu'à son interprétation finale. Ce schéma illustre visuellement les principales phases de ce processus, permettant de comprendre la logique et l'enchaînement du traitement appliqué aux images numériques [30].

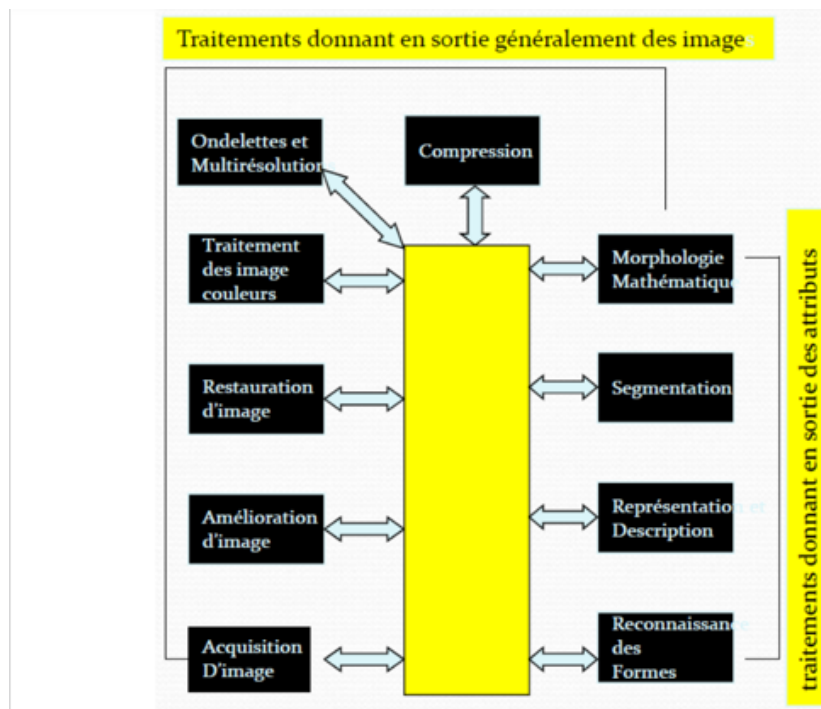


FIGURE 1.1 – schéma de différentes étapes de traitement d'image numérique

Voici les étapes fondamentales en traitement d'images numériques :

- **L'acquisition d'image** : cette étape implique un traitement préliminaire tel que la mise à l'échelle... Cela peut être aussi simple que de disposer d'une image au format digital.
- **L'amélioration de l'image** : consiste à utiliser des techniques permettant de mettre en évidence des détails en atténuant et en accentuant certains éléments selon l'intérêt, et c'en opérant sur la luminosité, le contraste...
- **La restauration d'image** : cette étape fait partie de celle liée à l'amélioration de l'apparence d'une image. La restauration d'image est dite objective dans le sens où les techniques de restauration tendent à se baser sur des modèles probabilistes ou mathématiques quant à la correction de la dégradation de l'image. Cette étape est souvent réalisée via le traitement d'image en MATLAB, qui est une plateforme de traitement numérique et de programmation pour l'analyse de données.
- **Le traitement des images couleurs** : Le traitement des couleurs est une étape prenant de plus en plus d'importance étant donné l'augmentation significative de l'usage des images digitales sur Internet. Cela peut inclure la modélisation et le traitement des couleurs des images digitales.
- **Le traitement des ondulations et des résolutions** : Le fondement de la représentation d'image en des degrés de résolution variés, est rendu possible par les ondulations. Dans ce type de traitement d'images numériques, les images sont subdivisées en de plus petites zones pour compresser les données et être présentées de manière pyramidale.
- **La compression** : La compression : Les techniques de compression réduisent l'espace de stockage requis pour une image et la bande passante nécessaire à sa transmission.
- **Le traitement morphologique** : il consiste en l'extraction des composants utiles à la représentation, à la description d'une forme.
- **Segmentation** : procédure de partitionnement de l'image en ses composantes ou objets : reconnaissance des formes...
- **La description et la représentation** : Ces étapes suivent presque toujours le résultat d'une segmentation. Ce résultat représente habituellement des données brutes de pixels qui peuvent constituer ou les limites d'une zone ou tous les points de cette même zone.
- **L'identification de l'objet** : Il s'agit du processus consistant à attribuer une étiquette à l'image en se basant sur ce qu'elle représente [30].

1.3 Handicape visuelle

1.3.1 l'anatomie de l'œil

L'œil est un organe complexe et essentiel du système visuel humain. Il joue un rôle crucial dans la captation des images et leur transformation en signaux électriques transmis au nerf optique. Ces signaux sont ensuite interprétés par le cerveau, plus précisément par le cortex visuel, ce qui nous permet de percevoir et d'interpréter notre environnement.

SCHÉMA EN COUPE DE L'OEIL HUMAIN

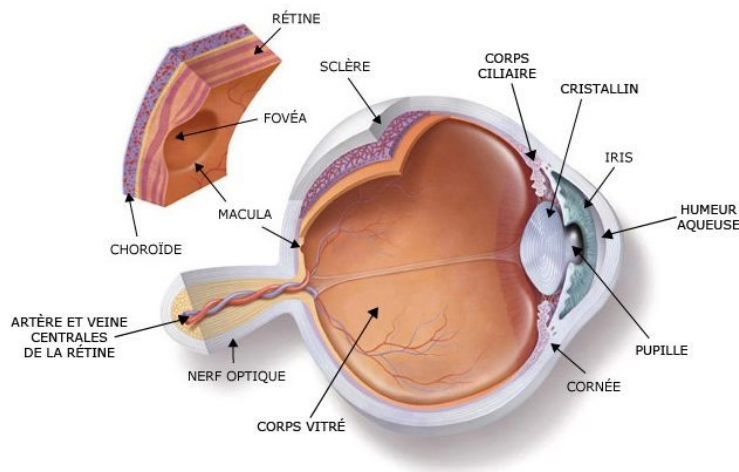


FIGURE 1.2 – schéma en coupe de l'œil humain

L'œil peut se définir comme une caméra, la plus complexe et la plus aboutie qui existe. Dans cet organe ultra-perfectionné, chaque élément a son rôle et son importance.

La cornée : la responsable de l'éclairage

Le cristallin : un rôle de « zoom »

L'iris : la couleur des yeux

La pupille : le diaphragme de l'œil

La rétine : le film photographique [33].

1.3.2 La déficience visuelle

La vision est un processus psychosensoriel complexe qui résulte de l'interaction de nombreux facteurs. Les rayons lumineux traversent la cornée et le cristallin pour former une

image nette sur la rétine à l'arrière de l'œil. Les photorécepteurs convertissent ensuite ce signal en messages électriques, qui sont transmis au cortex visuel primaire par le biais du nerf optique et du thalamus. Différents traitements dans le cortex visuel et les aires associatives permettent la perception de l'environnement, telle que l'appréciation des formes, des couleurs, du mouvement et des distances [33].

1.3.3 Causes de déficience visuelle

Globalement, les principales causes de déficience visuelle au niveau mondial sont :

- **Dégénérescence maculaire liée à l'âge (DMLA)** C'est une maladie qui affecte la macule, la partie centrale de la rétine, et qui est la première cause de cécité dans les pays industrialisés. Elle se caractérise par une atrophie de l'épithélium pigmentaire rétinien de la fovéa, qui est la zone centrale de la rétine. Cette altération de la vision centrale peut entraîner des difficultés à la lecture, à la reconnaissance des visages et à la conduite [5].
- **Cataracte** La cataracte est une affection de l'œil qui se caractérise par une opacification progressive du cristallin, la lentille située à l'intérieur de l'œil. Elle est très fréquente, touchant plus d'une personne sur cinq à partir de 65 ans. Les symptômes de la cataracte apparaissent progressivement et comprennent une perte progressive de la vision, une vision floue, des couleurs ternes, une sensibilité accrue à la lumière et parfois une vision double [33].
- **Glaucome** Le glaucome est un groupe de troubles oculaires caractérisés par une lésion progressive du nerf optique, souvent associée à une élévation de la pression intraoculaire, pouvant entraîner une perte de vision irréversible. Ces maladies chroniques se manifestent par une détérioration des fibres nerveuses du nerf optique, ce qui peut affecter le champ visuel et la vision périphérique [33]. Et d'autres causes telles que : les erreurs de réfraction non corrigées, la rétinopathie diabétique, l'opacification cornéenne, le trachome, blessures oculaires, maladies génétiques et maladies neurologiques [33].

1.3.4 Les impacts physiques de la malvoyance

- **La vision centrale** Le trouble de la vision centrale est le type de malvoyance le plus répandu. Il s'agit des cas où l'atteinte visuelle concerne la partie centrale de la rétine : la macula. Ce sont les formes, les couleurs, mais surtout la vision des détails qui sont atteintes. Il devient donc compliqué d'accomplir certaines tâches qui peuvent s'avérer importantes, comme lire, écrire, effectuer des travaux de précision ou encore reconnaître

les visages. Par contre, la perception de l'espace, des grandes formes ou encore des mouvements restent bons. Les personnes atteintes d'un handicap de leur vision centrale aperçoivent une tâche plus ou moins importante, selon le degré de handicap. Ces cas sont souvent des cas de DMLA qui surviennent la plupart du temps avec l'âge.



FIGURE 1.3 – Visualisation de l'impact physiologique de la vision centrale

- **La vision périphérique** À l'inverse de la vision centrale, lorsque la vision périphérique est atteinte, ce sont les champs périphériques qui sont altérés. Autrement dit, le handicap se situe autour du point de fixation. Du coup, le champ visuel se rétrécit et les personnes atteintes ne voient que ce qu'elles fixent, comme lorsqu'on regarde dans un tube. Les sujets gardent leur capacité de lecture, mais les caractères doivent être de préférence petits, pour ne pas sortir du champ de vision. Aussi, ils ont besoin d'une plus grande quantité de lumière pour être en mesure de voir efficacement. Le déplacement est le principal problème de ce handicap. Il devient en effet compliqué de garder une totale autonomie. Dans ce cas, ce sont souvent le glaucome (associé à une augmentation de la pression à l'intérieur de l'œil, qui entraîne une altération du champ visuel) ou la rétinite pigmentaire (perte progressive de la vision périphérique) qui sont les causes de ce handicap.
- **Sensibilité aux contrastes** C'est notre capacité de faire la différence entre les couleurs d'un ton similaire. C'est par exemple notre capacité d'identifier une assiette blanche sur une nappe blanche ou les traits d'un visage. Toutes les déficiences oculaires peuvent entraîner une baisse de la sensibilité aux contrastes.
- **Perception de la profondeur** C'est notre capacité d'évaluer la distance d'un objet.



FIGURE 1.4 – visualisation de l'impact physiologique de la vision périphérique

La déficience d'un œil peut avoir pour conséquence la perte de perception de la profondeur de champs. Il peut alors devenir difficile de saisir un objet, ou de percevoir la hauteur d'une marche d'escalier.

- **Processus visuel** La lentille dans l'œil permet de focaliser les rayons lumineux sur la rétine. Rétine qui transforme les rayons lumineux en signaux émis vers le cerveau grâce au nerf optique. Le cerveau reçoit ce signal et l'interprète comme les images que nous voyons. Le moindre problème sur l'un des organes ou sur l'un de ces processus peut affecter notre vue.

1.4 Méthodes traditionnelles de détection d'objets

1.4.1 Histogrammes de gradients orientés (HOG)

Le HOG est une technique qui consiste à évaluer les histogrammes locaux des orientations des gradients d'intensité dans une grille dense. Cette méthode permet de caractériser l'apparence et la forme locale des objets dans une image, même sans connaître précisément les positions des gradients ou des contours correspondants. Pour ce faire, l'image est divisée en petites régions spatiales appelées "cellules", et pour chaque cellule, un histogramme unidimensionnel 1D des directions des gradients ou des orientations des contours locaux est calculé. Les entrées combinées de ces histogrammes forment la représentation globale de l'image. Pour améliorer l'invariance à l'illumination, à l'ombre, etc., il est également utile

de normaliser les réponses locales avant de les utiliser. Cela peut être réalisé en accumulant une mesure d'énergie" d'histogramme local sur des régions spatiales légèrement plus grandes appelées "blocs", puis en utilisant les résultats pour normaliser toutes les cellules du bloc. Les blocs de descripteurs normalisés sont appelés des descripteurs (HOG). En utilisant une grille dense (qui se chevauche) de ces descripteurs HOG pour couvrir la fenêtre de détection, et en utilisant le vecteur de caractéristiques combiné dans un classifieur de fenêtre basé sur une machine à vecteurs de support (SVM), on obtient une chaîne de détection d'objets, telle que la détection d'humains [1].

Le descripteur HOG se concentre sur la structure ou la forme d'un objet. Il est meilleur que n'importe quel descripteur de bord, car il utilise la magnitude et l'angle du gradient pour calculer les caractéristiques. Pour les régions de l'image, il génère des histogrammes en utilisant la magnitude et les orientations du gradient [1].

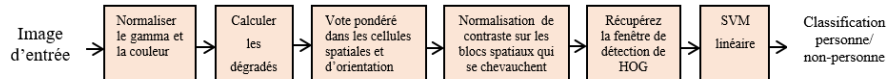


FIGURE 1.5 – Architecture de HOG

1.4.2 Méthodes de détection basées sur les cascades

Les méthodes de détection basées sur les cascades utilisent des classifieurs en cascade pour détecter rapidement les objets en éliminant les régions non pertinentes de l'image. Ces méthodes sont particulièrement efficaces pour la détection de visages et d'autres objets dans des images en temps réel. La méthode de Viola et Jones est l'une des méthodes les plus connues et les plus utilisées pour la détection d'objets basée sur les cascades. Elle a été initialement développée pour la détection de visages, mais peut également être utilisée pour détecter d'autres types d'objets tels que des voitures ou des avions. Cette méthode utilise un classifieur en cascade qui est entraîné à détecter des caractéristiques spécifiques de l'objet recherché. Le classifieur en cascade est composé de plusieurs étapes, où chaque étape utilise un classifieur simple pour éliminer les régions non pertinentes de l'image. Cela permet d'accélérer considérablement le processus de détection en évitant de traiter toutes les régions de l'image. La méthode des cascades de Haar est une autre méthode populaire de détection basée sur les cascades. Elle a été proposée par Paul Viola et Michael Jones en 2001 et est connue pour sa capacité à détecter rapidement les objets tout en maintenant un taux de succès élevé. Cette méthode utilise des caractéristiques de Haar, qui sont des filtres carrés appliqués sur l'image pour extraire des informations discriminantes. Les caractéristiques de Haar sont utilisées pour former un classifieur en cascade qui peut détecter efficacement les

objets dans une image [29].

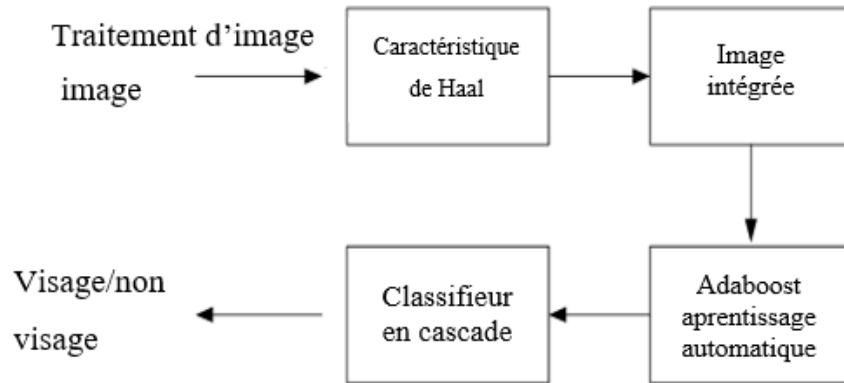


FIGURE 1.6 – Schéma de la méthode Viola-Jones

1.5 Techniques avancées de détection d'objets basées sur l'apprentissage profond

1.5.1 Réseaux de neurones convolutifs (CNN)

Un réseau de neurones convolutifs (CNN) est un type de réseau de neurones artificiels largement utilisé dans le domaine de l'apprentissage automatique. Inspiré par le fonctionnement du cortex visuel des animaux, les CNN sont conçus pour traiter les informations visuelles de manière similaire au cerveau humain. Ils sont particulièrement adaptés à la classification et à la reconnaissance d'objets dans les images. Les CNN utilisent des couches de convolution pour extraire des caractéristiques des images. Ces couches appliquent des filtres aux images d'entrée, ce qui permet de réduire le nombre de paramètres et d'optimiser l'apprentissage. Les CNN sont composés de plusieurs types de couches, notamment les couches de convolution, les couches de regroupement (pooling) et les couches entièrement connectées. Une caractéristique importante des CNN est leur capacité à apprendre automatiquement les caractéristiques des images, sans intervention humaine. Ils optimisent les filtres à travers l'apprentissage automatisé, ce qui leur permet d'extraire des caractéristiques pertinentes sans nécessiter une connaissance préalable. Les CNN sont utilisés dans de nombreux domaines, tels que la reconnaissance d'images et de vidéos, les systèmes de recommandation et le traitement du langage naturel. Ils ont également été largement utilisés pour la détection d'objets. Les CNN profonds sont capables de détecter et de reconnaître des objets dans les images en utilisant des cartes de caractéristiques obtenues par convolution et mise en commun [10].

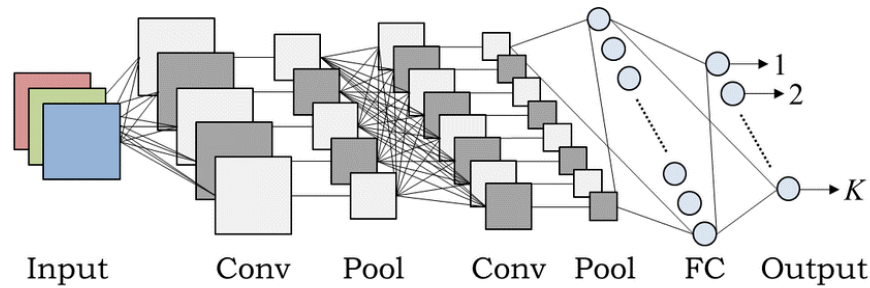


FIGURE 1.7 – Architecture de CNN

1.5.2 Région-based Convolutional Neural Networks (R-CNN)

Les réseaux de neurones convolutifs (CNN) sont un type de réseau neuronal qui a connu des avancées significatives dans le domaine de l'apprentissage profond. Ils sont particulièrement utilisés dans la détection d'objets et la segmentation sémantique. Le R-CNN (Réseaux de neurones convolutifs basés sur les régions) est un système spécifique de détection d'objets qui utilise plusieurs étapes et techniques pour obtenir une détection précise des objets dans une image [24].

Le fonctionnement de R-CNN peut être divisé en quatre étapes principales : la génération de propositions de régions, l'extraction de caractéristiques, la classification et la régression.

- **Génération de propositions de régions** : Dans cette étape, des régions d'intérêt potentielles sont générées à l'aide d'un algorithme de génération de propositions de régions tel que Selective Search. Cet algorithme analyse l'image et identifie des régions qui pourraient contenir des objets.
- **Extraction de caractéristiques** : chaque région d'intérêt (ROI) est extraite de l'image et redimensionnée pour être compatible avec le réseau de neurones convolutifs. Ensuite, le réseau extrait des caractéristiques pertinentes de chaque ROI. Ces caractéristiques sont souvent obtenues à partir des couches convolutives du réseau.
- **Classification** : Les caractéristiques extraites de chaque ROI sont utilisées pour classer l'objet contenu dans la région. Un classifieur, tel qu'un SVM (Support Vector Machine), est souvent utilisé pour cette tâche. Le classifieur est entraîné sur un ensemble de données annotées contenant des exemples positifs (objets d'intérêt) et des exemples négatifs (arrière-plans).
- **Régression** : en plus de la classification, R-CNN effectue également une régression pour estimer la position précise de l'objet dans l'image. Un régresseur sert à prédire les coordonnées de la boîte englobante d'un objet [24].

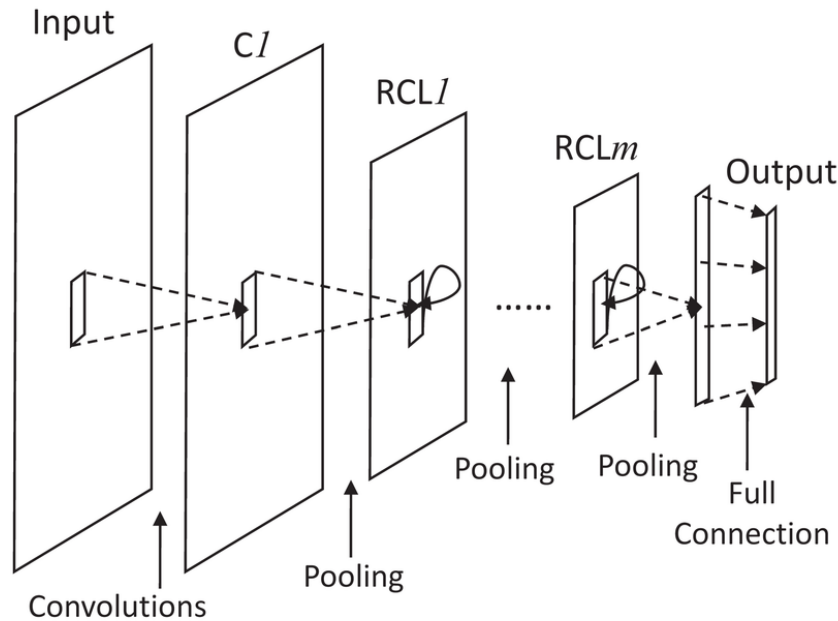


FIGURE 1.8 – Architecture de R-CNN

1.5.3 You Only Look Once (YOLO)

YOLO est une approche révolutionnaire pour la détection d'objets en temps réel. Contrairement aux méthodes traditionnelles qui utilisent des classificateurs pour effectuer la détection, YOLO formule la détection d'objets comme un problème de régression. Il prédit directement les cadres de délimitation et les probabilités de classe associées à partir de l'image complète en une seule évaluation. Le fonctionnement technique de YOLO est le suivant :

- **Architecture du réseau** : YOLO utilise un réseau de neurones convolutifs pour détecter les objets dans une image. Le modèle CNN est divisé en deux parties : les premières couches convolutives sont pré-entraînées sur ImageNet, puis le modèle est converti pour effectuer la détection en ajoutant des couches convolutives et connectées supplémentaires [15].
- **Division en grille** : YOLO divise l'image d'entrée en une grille de cellules. Chaque cellule est responsable de la prédiction d'un cadre de délimitation et d'une classe d'objet. Par exemple, une image peut être divisée en une grille 7x7, ce qui donne 49 cellules. Chaque cellule prédit un ou plusieurs cadres de délimitation et les étiquettes de classe correspondantes [15].
- **Anchor boxes** : YOLO utilise des "anchor boxes", qui sont des boîtes englobantes définies avec une certaine hauteur et largeur. Ces anchor boxes permettent au réseau de détecter des objets de différentes échelles et des objets qui se chevauchent. Chaque

cellule prédit les décalages par rapport aux anchor boxes pour ajuster les cadres de délimitation prédits [15].

- **Suppression non max** : une fois que les prédictions de cadres de délimitation et de classes sont faites, YOLO utilise une technique appelée "suppression non max" pour éliminer les détections redondantes et ne garder que les détections les plus pertinentes. Cette technique compare les scores de confiance des cadres de délimitation et supprime ceux qui ont une forte intersection avec d'autres cadres de délimitation ayant un score de confiance plus élevé [15].

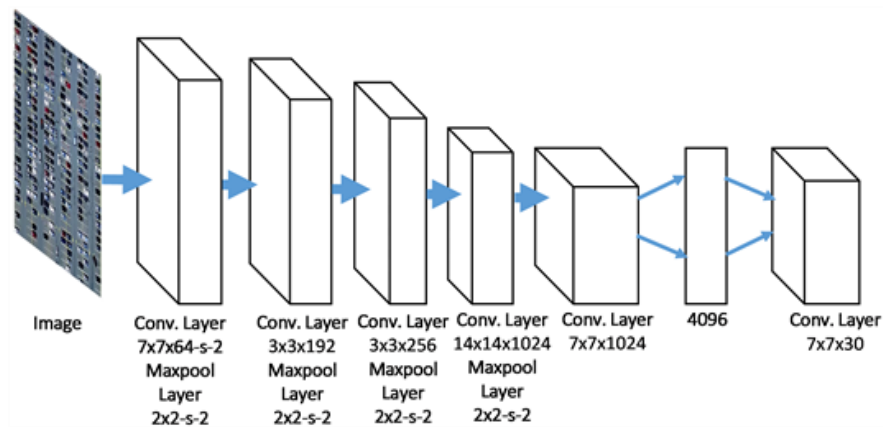


FIGURE 1.9 – Architecture de YOLO

1.5.4 Single ShotMultiBox Detector (SSD)

Le SSD est une méthode de détection d'objets en une seule passe qui utilise des pyramides spatiales pour détecter des objets de différentes tailles. Cette méthode a été développée pour résoudre certains problèmes rencontrés par les architectures précédentes, telles que la lenteur de l'entraînement et de l'inférence, ainsi que la nécessité de plusieurs phases d'entraînement [28].

Le fonctionnement technique du SSD est :

- **Pyramides spatiales** : le SSD utilise des pyramides spatiales pour détecter des objets de différentes tailles. Ces pyramides sont des ensembles de cartes de caractéristiques de différentes résolutions, qui sont obtenues à partir de différentes couches du réseau de neurones. Chaque carte de caractéristiques dans la pyramide est utilisée pour détecter des objets de taille spécifique.
- **Default boundary boxes** : le SSD utilise des "default boundary boxes" (boîtes de délimitation par défaut) pour générer des prédictions de détection d'objets. Ces boîtes

de délimitation par défaut sont des boîtes prédéfinies qui sont placées à des emplacements spécifiques sur les cartes de caractéristiques. Chaque boîte de délimitation par défaut est associée à un ensemble de prédictions pour différents attributs de l'objet, tels que la classe, la position et la taille.

- **Prédiction des boîtes de délimitation** : le SSD utilise des régressions linéaires pour prédire les décalages par rapport aux boîtes de délimitation par défaut. Ces décalages représentent la position et la taille de l'objet détecté par rapport à la boîte de délimitation par défaut correspondante. En utilisant ces décalages, le SSD peut générer des boîtes de délimitation précises pour les objets détectés.
- **Classification des objets** : le SSD utilise également des réseaux de neurones pour classer les objets détectés. Chaque boîte de délimitation prédite est associée à un score de confiance pour chaque classe d'objet possible. Les objets sont ensuite classés en fonction de ces scores de confiance [28].

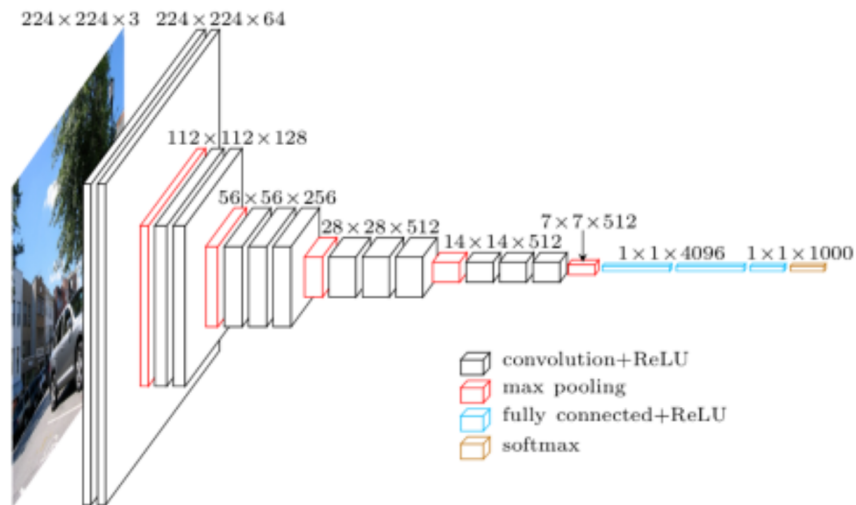


FIGURE 1.10 – Architecture de SSD

1.6 implication de la vision artificielle dans l'amélioration d'accessibilité visuelle des malvoyants

La vision artificielle est une branche de l'intelligence artificielle qui se concentre sur la compréhension et l'interprétation des informations visuelles. Elle joue un rôle crucial dans l'amélioration de l'accessibilité visuelle des personnes malvoyantes. Grâce à des avancées technologiques telles que la reconnaissance d'objets, la détection des obstacles et la lecture de texte, la vision artificielle offre de nouvelles possibilités pour aider les personnes malvoyantes à percevoir et à interagir avec leur environnement.

1.6.1 Reconnaissance d'objets

La reconnaissance d'objets est l'une des applications les plus importantes de la vision artificielle pour les personnes malvoyantes. Grâce à des algorithmes, les systèmes de reconnaissance d'objets peuvent identifier et nommer les objets qui se trouvent dans le champ de vision d'une personne malvoyante. Cela permet aux utilisateurs de reconnaître les personnes, les animaux, les véhicules et d'autres objets courants, ce qui facilite leur interaction avec leur environnement. La reconnaissance d'objets utilise des techniques telles que la segmentation d'image, l'extraction de caractéristiques et l'apprentissage automatique pour analyser les informations visuelles et identifier les objets. Ces systèmes peuvent être intégrés à des dispositifs portables tels que des lunettes intelligentes, offrant ainsi une expérience plus pratique et discrète pour les personnes malvoyantes.

1.6.2 Détection des obstacles

La détecter des obstacles aide les personnes malvoyantes à naviguer en toute sécurité dans leur environnement. Les systèmes de détection des obstacles utilisent des capteurs tels que des caméras ou des lidars pour détecter les objets qui se trouvent sur le chemin d'une personne malvoyante. Ces systèmes peuvent avertir l'utilisateur de la présence d'obstacles et lui fournir des informations sur leur emplacement et leur distance. La détection des obstacles peut être particulièrement utile dans des situations telles que la marche en extérieur, la traversée de routes ou la navigation dans des espaces publics. En fournissant des avertissements en temps réel, elle permet aux personnes malvoyantes de se déplacer de manière plus autonome et en toute sécurité.

1.6.3 Reconnaissance des personnes

La détection de visage permet aux malvoyants de reconnaître et d'identifier les personnes qui les entourent. Grâce à cette technologie, les malvoyants peuvent recevoir des informations sur l'identité des personnes, ce qui facilite les interactions sociales.

1.7 Application de la détection d'objets dans le contexte des systèmes embarqués

La détection d'objets dans le contexte des systèmes embarqués offre de nombreuses applications pratiques et utiles :

1.7.1 Systèmes de surveillance

Un système de surveillance dans la détection d'objets est un ensemble de technologies et de dispositifs conçus pour détecter, suivre et analyser les objets en mouvement. Ces systèmes utilisent une combinaison de capteurs, de caméras, d'algorithmes de traitement d'image et de logiciels avancés pour détecter les objets et les suivre en temps réel. Ils peuvent être utilisés dans divers contextes, tels que la surveillance de la circulation routière, la surveillance des zones publiques, la sécurité des bâtiments et bien d'autres.

- **Le fonctionnement d'un système de surveillance** Un système de surveillance dans la détection d'objets fonctionne en plusieurs étapes. Tout d'abord, les capteurs et les caméras captent les images de la zone surveillée. Ensuite, ces images sont traitées par des algorithmes de traitement d'image qui identifient les objets présents dans la scène. Les objets détectés sont ensuite suivis en temps réel à l'aide de techniques de suivi d'objets avancées. Les systèmes de surveillance utilisent également des techniques d'apprentissage automatique pour améliorer la précision de la détection d'objets. Ces techniques permettent au système de reconnaître et de différencier différents types d'objets, tels que les voitures, les personnes, les animaux, etc. De plus, les systèmes de surveillance peuvent être configurés pour déclencher des alertes ou des actions spécifiques lorsque des objets suspects sont détectés, comme l'envoi d'une notification à un agent de sécurité ou l'enregistrement d'une vidéo de la scène.

1.7.2 Robots autonomes

Un robot autonome dans la détection d'objets est un système robotique capable de se déplacer de manière autonome dans un environnement donné, d'analyser les scènes à l'aide

de capteurs et d'algorithmes avancés, et de détecter les objets qui s'y trouvent. Ces robots utilisent une combinaison de capteurs, tels que des caméras, des capteurs de profondeur, des capteurs infrarouges, des lidars, etc., pour percevoir leur environnement et identifier les objets présents.

- **Le fonctionnement d'un robot autonome** Un robot autonome dans la détection d'objets fonctionne en plusieurs étapes. Cependant, il utilise ses capteurs pour collecter des données sur l'environnement, y compris les images, les distances, les températures, etc. Ensuite, ces données sont traitées par des algorithmes de traitement d'image et d'apprentissage automatique pour détecter et reconnaître les objets présents dans la scène. Une fois les objets détectés, le robot peut prendre des décisions en fonction de la situation. Par exemple, il peut éviter les obstacles, suivre un objet spécifique, effectuer des tâches spécifiques liées à la détection d'objets, ou même interagir avec les objets détectés. Tout cela est rendu possible grâce à des algorithmes de planification de mouvement et de contrôle qui permettent au robot de se déplacer de manière autonome dans son environnement.

1.7.3 Systèmes de détection de collision

Un système de détection de collision dans la détection d'objets est un ensemble de capteurs, d'algorithmes et de logiciels qui permettent à un robot autonome de détecter et d'éviter les collisions avec les objets présents dans son environnement. Ce système utilise une variété de capteurs, tels que des capteurs de proximité, des capteurs de distance, des capteurs infrarouges, des lidars, etc., pour percevoir les obstacles et analyser leur position, leur taille et leur mouvement.

- **Le fonctionnement d'un système de détection de collision** Un système de détection de collision dans la détection d'objets fonctionne en plusieurs étapes. En premier lieu, les capteurs du robot collectent des données sur l'environnement, y compris la position, la distance et la taille des objets environnants. Ensuite, ces données sont traitées par des algorithmes de détection de collision qui analysent les informations des capteurs et déterminent s'il y a un risque de collision imminente. Si un risque de collision est détecté, le système de détection de collision déclenche des mesures préventives pour éviter l'accident. Cela peut inclure des actions telles que le freinage, l'évitement d'obstacles, le changement de trajectoire ou même l'arrêt complet du robot. Le système de détection de collision peut également envoyer des alertes visuelles ou sonores pour avertir les opérateurs ou les personnes à proximité.

1.7.4 Système de détection d'objets et de personnes pour les malvoyants

La détection d'objets et de personnes est une technologie qui peut être d'une grande aide pour les personnes malvoyantes. Grâce aux avancées de l'intelligence artificielle et de la vision par ordinateur, il est désormais possible de développer des systèmes qui permettent aux malvoyants de détecter et de reconnaître les objets et les personnes qui les entourent.

- **Le fonctionnement d'un système de détection d'objets et de personnes pour les malvoyants** Un système de détection d'objets et de personnes pour les malvoyants utilise des caméras ou des capteurs pour capturer des images de l'environnement. Ces images sont ensuite analysées par un modèle d'apprentissage en profondeur, qui est capable de détecter et de reconnaître les objets et les personnes présents dans l'image. Le modèle d'apprentissage en profondeur est entraîné sur de grandes quantités de données afin d'apprendre à reconnaître différents types d'objets et de personnes. Il utilise des techniques telles que les réseaux de neurones convolutifs pour extraire des caractéristiques visuelles des images et les classer en différentes catégories. Une fois que le modèle a identifié les objets et les personnes dans l'image, il peut fournir des informations auditives ou tactiles à l'utilisateur pour l'aider à comprendre son environnement. Cela peut se faire par le biais d'un haut-parleur qui annonce les objets détectés ou d'un dispositif haptique qui vibre ou émet des signaux tactiles pour indiquer la présence d'objets ou de personnes.

1.7.5 Systèmes de surveillance de la santé

Un système de surveillance de santé dans la détection d'objets est un ensemble de capteurs, d'algorithmes et de logiciels qui permettent de détecter et de surveiller les signes vitaux et les conditions médicales des individus. Ce système utilise une variété de capteurs, tels que des capteurs de fréquence cardiaque, des capteurs de température corporelle, des capteurs de mouvement, des capteurs de saturation en oxygène, etc., pour collecter des données sur la santé des personnes.

- **Le fonctionnement d'un système de surveillance de santé** Les capteurs du système collectent des données sur les signes vitaux et les conditions médicales des individus. Ces données sont ensuite traitées par des algorithmes de surveillance de santé qui analysent les informations des capteurs et détectent les anomalies ou les situations d'urgence. Si une anomalie ou une situation d'urgence est détectée, le système de surveillance de santé déclenche des mesures appropriées pour assurer la sécurité et

le bien-être des individus. Cela peut inclure des actions telles que l'envoi d'alertes aux professionnels de la santé, l'activation d'un système d'assistance médicale d'urgence ou même le déclenchement de mesures préventives pour éviter les accidents.

1.7.6 Systèmes de navigation intelligente

Un système de navigation intelligent dans la détection d'objets est un ensemble de capteurs, d'algorithmes et de logiciels qui permettent à un véhicule ou à un robot de détecter et d'éviter les objets et les obstacles sur la route. Ce système utilise une variété de capteurs, tels que des capteurs de vision, des capteurs de distance, des capteurs de mouvement, etc., pour collecter des données sur l'environnement et les conditions de conduite.

- **Le fonctionnement d'un système de navigation intelligente** Un système de navigation intelligent dans la détection d'objets fonctionne en plusieurs étapes : tout d'abord, les capteurs du système collectent des données sur l'environnement, telles que la position des objets, la distance par rapport aux obstacles, etc. Ces données sont ensuite traitées par des algorithmes de détection d'objets qui analysent les informations des capteurs et identifient les objets et les obstacles présents sur la route. Une fois les objets détectés, le système de navigation intelligente utilise des algorithmes de planification de trajectoire pour déterminer la meilleure route à suivre, en évitant les obstacles détectés. Ces algorithmes prennent en compte des facteurs tels que la distance, la vitesse, les règles de circulation, etc., pour calculer une trajectoire optimale et sécurisée.

1.8 Limitations et défis actuels dans la reconnaissance d'objets pour les personnes malvoyantes

1.8.1 Limitations actuelles

- **Précision de la reconnaissance** La précision de la reconnaissance d'objets pour les personnes malvoyantes est encore limitée. Les systèmes de reconnaissance peuvent avoir du mal à identifier correctement les objets, en particulier dans des conditions d'éclairage difficiles ou lorsque les objets sont partiellement cachés.
- **Variété des objets reconnus** Les systèmes de reconnaissance d'objets actuels sont souvent limités à la reconnaissance d'objets courants tels que les personnes, les animaux, les voitures, etc. Ils peuvent avoir du mal à reconnaître des objets plus spécifiques ou moins courants.

- **Vitesse de reconnaissance** La vitesse de reconnaissance des objets peut être un défi, en particulier lorsque les personnes malvoyantes se déplacent rapidement ou lorsqu'il y a plusieurs objets à reconnaître en même temps. Les systèmes doivent être capables de reconnaître les objets en temps réel pour être réellement utiles.
- **Dépendance à la connectivité Internet** De nombreux systèmes de reconnaissance d'objets nécessitent une connexion Internet pour fonctionner, ce qui peut être un inconvénient pour les personnes malvoyantes qui se trouvent dans des zones avec une connectivité limitée ou inexistante.

1.8.2 Défis actuels

- **Coût élevé des dispositifs** Les dispositifs de reconnaissance d'objets spécialement conçus pour les personnes malvoyantes peuvent être coûteux, ce qui limite leur accessibilité pour de nombreuses personnes. Il est important de développer des solutions abordables pour rendre cette technologie plus largement disponible.
- **Intégration avec d'autres technologies** L'intégration de la reconnaissance d'objets avec d'autres technologies, telles que les applications mobiles ou les aides à la mobilité, peut être un défi. Il est essentiel de développer des interfaces conviviales et des protocoles de communication standardisés pour faciliter l'intégration de ces technologies.
- **Adaptabilité aux environnements changeants** Les personnes malvoyantes peuvent se déplacer dans des environnements variés et changeants. Les systèmes de reconnaissance d'objets doivent être capables de s'adapter à ces changements et de fournir des informations précises et utiles dans différents contextes.
- **Confidentialité et sécurité des données** La reconnaissance d'objets implique la collecte et le traitement de données personnelles. Il est essentiel de garantir la confidentialité et la sécurité de ces données pour protéger la vie privée des utilisateurs.

1.8.3 Perspectives d'avenir

- **Amélioration de la précision et de la vitesse** Les avancées technologiques continueront à améliorer la précision et la vitesse de la reconnaissance d'objets. L'utilisation de l'intelligence artificielle et de l'apprentissage automatique permettra d'affiner les algorithmes de reconnaissance et de les rendre plus performants.
- **Développement de dispositifs portables** Le développement de dispositifs portables, tels que des lunettes ou des écouteurs intelligents, permettra aux personnes

malvoyantes d'avoir accès à la reconnaissance d'objets de manière plus discrète et pratique.

- **Intégration de la reconnaissance d'objets dans les aides à la mobilité** L'intégration de la reconnaissance d'objets avec les aides à la mobilité, telles que les cannes électroniques ou les fauteuils roulants intelligents, permettra aux personnes malvoyantes de naviguer plus facilement dans leur environnement.

1.9 Discussion

En conclusion, ce premier chapitre met en évidence l'importance cruciale du traitement d'images pour les personnes malvoyantes. Les avancées technologiques offrent des opportunités précieuses pour rendre les images accessibles et contribuer à l'inclusion sociale et à l'autonomie de ces individus. L'analyse des différentes techniques, de l'acquisition à la segmentation, démontre leur efficacité dans l'adaptation des images aux besoins spécifiques des personnes malvoyantes. Les défis liés à des affections visuelles telles que la dégénérescence maculaire et la cataracte soulignent l'importance de solutions personnalisées. La compréhension approfondie des catégories d'images numériques permet de concevoir des techniques adaptées et de développer des outils offrant un accès équitable à l'information visuelle. La recherche continue et la participation active des utilisateurs sont essentielles pour améliorer ces technologies et répondre réellement aux besoins des personnes malvoyantes. En investissant dans le développement de ces technologies et en impliquant les utilisateurs, nous pouvons transformer leur expérience visuelle, favorisant ainsi une société plus inclusive et équitable pour tous.

Chapitre 2

Étude approfondie de la détection d'objets avec implémentation d'IA

2.1 Préambule

Ce chapitre intitulé : « Étude approfondie de la détection d'objets avec implémentation d'IA » vise à explorer que la détection d'objets est une tâche essentielle dans le domaine de la vision par ordinateur et de l'intelligence artificielle. Il s'agit d'identifier et de localiser des objets spécifiques dans des images ou des vidéos. L'utilisation des réseaux neuronaux convolutifs et de l'apprentissage profond (CNN) a révolutionné cette tâche, offrant des performances impressionnantes en matière de détection d'objets. La base de l'apprentissage profond pour la détection d'objets sera étudiée de manière approfondie. En outre, nous parlerons de l'architecture des réseaux neuronaux convolutionnels par exemple pour la détection d'objets, de la préparation des données et de l'augmentation, de l'entraînement des modèles de détection d'objets et de l'IA, et de l'utilisation de Python.

2.2 Fondements de l'apprentissage en profondeur pour la détection d'objets

2.2.1 Définition de l'apprentissage en profondeur

L'apprentissage profond (Deep Learning) est un sous-domaine de machine Learning ou apprentissage machine, sous-domaine de l'intelligence artificielle . Grâce à l'intelligence artificielle, nous comprenons le comportement consistant à laisser les machines reproduire des tâches que les humains trouvent complexes, généralement : laisser les machines raisonner, apprendre aux machines à planifier des tâches, ou même apprendre aux machines à représenter les connaissances de manière structurée [13].

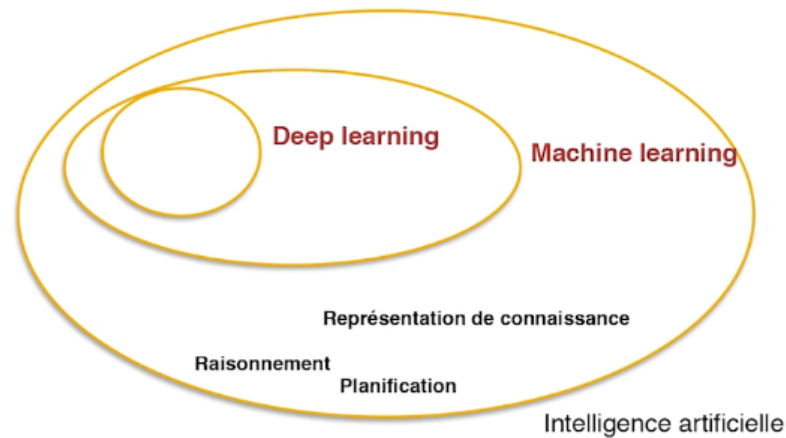


FIGURE 2.1 – L'intelligence artificielle et ses sous-domaines

L'apprentissage en profondeur, également connu sous le nom de Deep Learning, est une branche de l'intelligence artificielle qui a révolutionné de nombreux domaines, y compris la détection d'objets. Cette technologie permet aux ordinateurs d'apprendre à reconnaître et à localiser des objets dans des images ou des vidéos avec une précision impressionnante.

L'apprentissage en profondeur est une approche de l'intelligence artificielle qui s'inspire du fonctionnement du cerveau humain. Il utilise des réseaux de neurones artificiels profonds pour analyser des données complexes et extraire des motifs et des caractéristiques significatifs. Ces réseaux de neurones sont composés de plusieurs couches interconnectées, d'où le terme "en profondeur". Chaque couche traite les informations reçues et les transmet à la couche suivante, permettant ainsi une représentation hiérarchique des données.

- **Machine Learning (apprentissage automatique) :**

L'apprentissage automatique est un sous-domaine de l'intelligence artificielle, le terme faisant référence à la capacité des systèmes informatiques à trouver de manière autonome des solutions aux problèmes en reconnaissant des modèles. L'apprentissage automatique comme un sous-ensemble de l'intelligence artificielle, qui construit un modèle mathématique basé sur des échantillons de données, connus sous le nom de "données d'entraînement", pour faire des prédictions ou des décisions sans être précisées programmées pour effectuer la tâche [13].

- **Intelligence artificielle :** L'intelligence artificielle est une vaste branche de l'informatique qui s'intéresse à la construction de machines intelligentes programmées pour penser comme les humains et imiter leurs actions. Le terme peut également s'appliquer à toute machine qui présente des caractéristiques associées à l'esprit humain, Les ma-

chines dotées d'IA peuvent effectuer des tâches complexes telles que la reconnaissance vocale, la reconnaissance d'images, la traduction de langues et la prise de décisions [13].

2.2.2 Classification des algorithmes d'apprentissage

L'apprentissage automatique comporte de nombreux types et techniques, chacun étant utilisé pour résoudre un problème spécifique. L'ensemble de ces techniques peut être divisé en deux types principaux, en fonction du degré d'interférence humaine dans le processus d'apprentissage : l'apprentissage supervisé et l'apprentissage non supervisé [11].

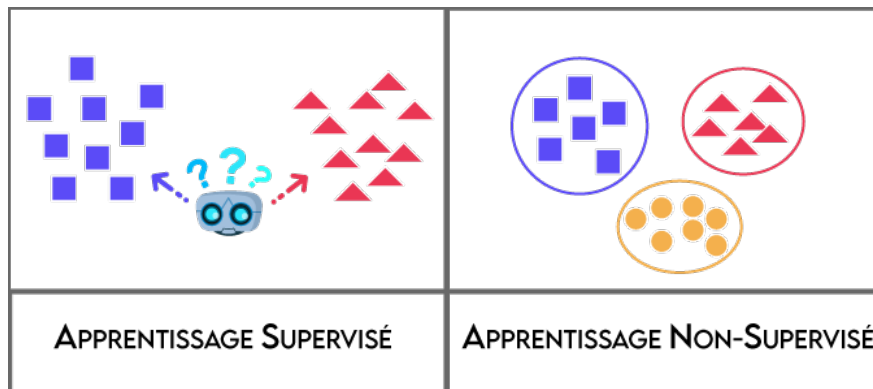


FIGURE 2.2 – les différents types d'apprentissage automatique

Apprentissage supervisé

C'est un processus sous la supervision d'un tiers. Dans l'apprentissage supervisé, le processus de Machine Learning peut être considéré comme un enseignant qui supervise le processus d'apprentissage. L'apprentissage supervisé est une méthode d'apprentissage automatique qui consiste à développer un algorithme capable d'apprendre une fonction prédictive. Cette méthode repose sur un entraînement basé sur des exemples, où un ensemble de variables d'entrée est associé à leurs variables de sortie correspondantes. L'objectif est d'obtenir une performance satisfaisante en répétant ce processus d'entraînement. Lors de chaque itération, la machine établit des règles qui relient les variables d'entrée aux variables de sortie. Ces règles sont utilisées par le modèle d'apprentissage pour prédire avec précision la valeur de sortie lorsque des valeurs d'entrée sont données[13].

Apprentissage non supervisé

Dans ce type d'apprentissage, le système travaille sur des données non étiquetées et tente de détecter des modèles et des caractéristiques communes pour classer les données sans

aucune intervention humaine. Il dispose de plusieurs algorithmes tels que le regroupement et l'apprentissage des règles d'association [13].

2.2.3 l'utilité de l'apprentissage profond

L'apprentissage profond, également connu sous le nom de Deep Learning, joue un rôle essentiel dans le domaine de la reconnaissance d'objets. Il permet d'obtenir des résultats de meilleure qualité par rapport à d'autres méthodes d'apprentissage machine. Grâce à ses multiples couches de neurones artificiels, le Deep Learning est capable de détecter des caractéristiques complexes et subtiles dans les images, ce qui lui permet d'atteindre des niveaux de précision élevés dans la reconnaissance d'objets. Il offre une exécution efficace des tâches de routine, sans écarts de qualité. Après avoir été entraînés, les neurones peuvent reconnaître des objets de manière autonome et constante sans montrer de fatigue. Cela permet d'économiser du temps et de l'argent, tout en garantissant un développement continu des fonctionnalités du système. Grâce à ses capacités d'analyse et de traitement des données non structurées, l'apprentissage profond permet de détecter, classifier et reconnaître des objets dans des images ou des vidéos avec une précision remarquable [9].

2.2.4 Les étapes de la détection d'objets en utilisant l'apprentissage en profondeur

La détection d'objets en utilisant l'apprentissage en profondeur se déroule généralement en plusieurs étapes :

1. **Prétraitement des données** : Cela pourrait inclure la normalisation des images, l'augmentation des données (par exemple, la rotation, le recadrage, le changement de luminosité) et la division des données en ensembles d'entraînement, de validation et de test.
2. **Entraînement du réseau** : Un ensemble d'images annotées est utilisé pour entraîner le réseau de neurones. Pendant l'entraînement, le réseau apprend à reconnaître les caractéristiques des objets d'intérêt et à les localiser dans l'image.
3. **Détection des objets** : Une fois que le réseau est entraîné, il peut être utilisé pour détecter des objets dans de nouvelles images. Le réseau parcourt l'image en utilisant des fenêtres de différentes tailles et effectue des prédictions sur la présence d'objets à chaque position.
4. **Post-traitement** : Les prédictions du réseau sont souvent filtrées et regroupées pour éliminer les détections redondantes et améliorer la précision. Des techniques telles que

la suppression non maximale (NMS) sont couramment utilisées pour ce faire [9].

2.3 Architecture des réseaux de neurones convolutifs pour la détection d'objets

2.3.1 Réseau neuronal

Les réseaux neuronaux sont un modèle informatique qui partage certaines propriétés avec le cerveau humain, dans lequel de nombreuses unités simples travaillent en parallèle sans unité de contrôle centralisée. Les poids entre les unités sont le principal moyen de stockage de l'information à long terme dans les réseaux neuronaux. La mise à jour des poids est le principal moyen pour le réseau neuronal d'apprendre de nouvelles informations. Le comportement des réseaux neuronaux est déterminé par leur architecture. L'architecture d'un réseau peut être définie par le nombre de neurones, le nombre de couches et les types de connexions entre les couches [31].

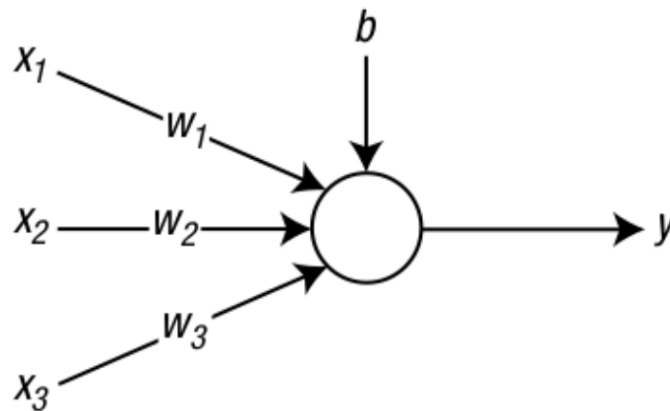


FIGURE 2.3 – Réseau neuronal

2.3.2 architecture de réseau de neurone

Chaque réseau de neurones artificiels est composé de plusieurs couches, chacune ayant un rôle spécifique dans le processus de traitement des informations.

- **Couche entrée (Input layer)** : La couche d'entrée est la première couche du réseau de neurones. Elle est responsable de recevoir les données d'entrée et de les transmettre aux neurones de la couche suivante. Chaque neurone de la couche d'entrée est connecté à chaque neurone de la couche suivante [31].

- **Couche caché (Hidden layer)** : Les couches cachées sont situées entre la couche d'entrée et la couche de sortie. Elles sont appelées "cachées" car elles ne sont pas directement accessibles depuis l'extérieur du réseau. Les couches cachées sont responsables de l'extraction des caractéristiques et des motifs complexes des données d'entrée [3].
- **Couche sortie (Output layer)** : La couche de sortie est la dernière couche du réseau de neurones. Elle est responsable de produire les résultats finaux ou les prédictions en fonction des données d'entrée et des informations apprises par les couches cachées. Le nombre de neurones dans la couche de sortie dépend du type de tâche que le réseau de neurones doit accomplir. Par exemple, dans un réseau de neurones utilisé pour la classification, le nombre de neurones de la couche de sortie correspond au nombre de classes possibles [31].

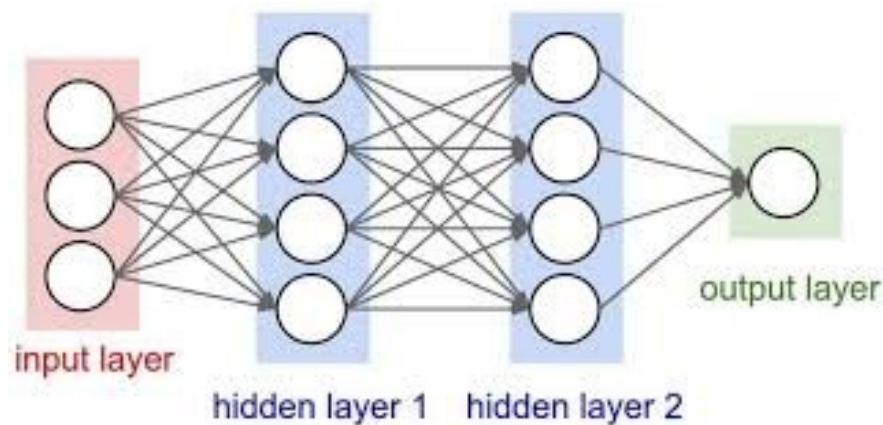


FIGURE 2.4 – Architecture de réseau de neurone

2.3.3 Réseaux neuronaux convolutifs

Le réseau neuronal convolutif est un type de modèle d'apprentissage profond pour le traitement des données qui ont un motif de grille, comme les images, qui est inspiré par l'organisation du cortex visuel animal et conçu pour apprendre automatiquement et de manière adaptative des hiérarchies spatiales de caractéristiques, des motifs de bas niveau aux motifs de haut niveau. Le CNN est une construction mathématique qui se compose généralement de trois types de couches (ou blocs de construction) : convolution, mise en commun et couches entièrement connectées. Une architecture typique consiste en des répétitions d'une pile de plusieurs couches de convolution et d'une couche de mise en commun, suivies d'une ou de plusieurs couches entièrement connectées. L'étape au cours de laquelle les données d'entrée sont transformées en données de sortie par ces couches est appelée propagation vers l'avant. Par le biais de ces couches est appelée propagation vers l'avant [3].

2.3.4 L'architecture d'un CNN

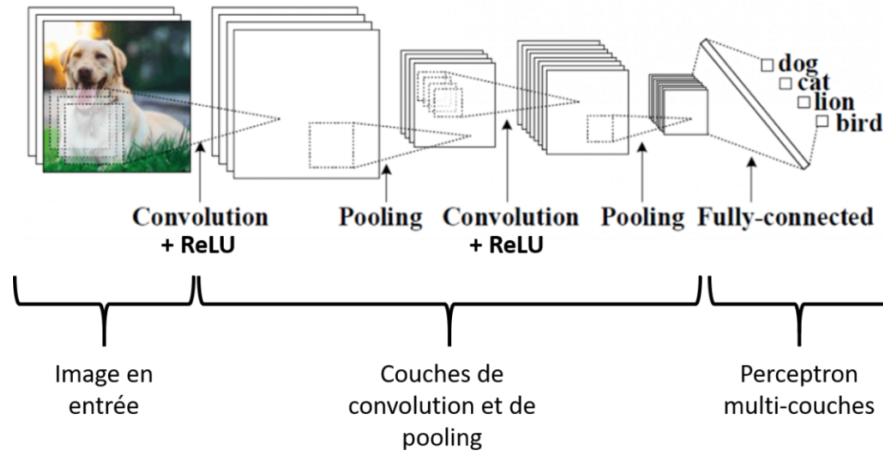


FIGURE 2.5 – architecture de base d'un CNN

On trouve deux parties principales :

- **Une partie convolutive** Son but ultime est d'extraire les caractéristiques propres à chaque image en les compressant pour réduire leur taille initiale. Il génère à la fin une carte caractéristique.
- **Une partie classification** La carte caractéristique obtenue en sortie de la partie convolutive est fournie en entrée dans une deuxième partie, constituée de couches entièrement connectées appelées perceptron multicouche (MLP pour Multi Layer Perceptron). Le rôle de cette partie est de combiner les caractéristiques du code CNN afin de classer l'image [15].

1. Couche de convolution

La couche de convolution est un composant fondamental de l'architecture CNN qui effectue l'extraction des caractéristiques, laquelle consiste généralement en une combinaison d'opérations linéaires et non linéaires, à savoir l'opération de convolution et la fonction d'activation.

- **Convolution**

La convolution est un type spécialisé d'opération linéaire utilisé pour l'extraction de caractéristiques, où un petit tableau de nombres, appelé noyau, est appliqué à l'entrée, qui est un tableau de nombres, appelé tenseur. Un produit élément par élément entre chaque élément du noyau et le tenseur d'entrée est calculé à chaque emplacement du tenseur et additionné pour obtenir la valeur de sortie à l'emplacement correspondant du tenseur de sortie, appelé carte de caractéristiques. Cette

procédure est répétée en appliquant plusieurs noyaux pour former un nombre arbitraire de cartes de caractéristiques, qui représentent différentes caractéristiques des tenseurs d'entrée; différents noyaux peuvent donc être considérés comme différents extracteurs de caractéristiques. Les deux paramètres clés de l'hyper qui définissent l'opération de convolution sont la taille et le nombre de noyaux. Le premier est généralement de 3×3 , mais parfois de 5×5 ou de 7×7 . Le second est arbitraire et détermine la profondeur des cartes de caractéristiques de sortie [3].

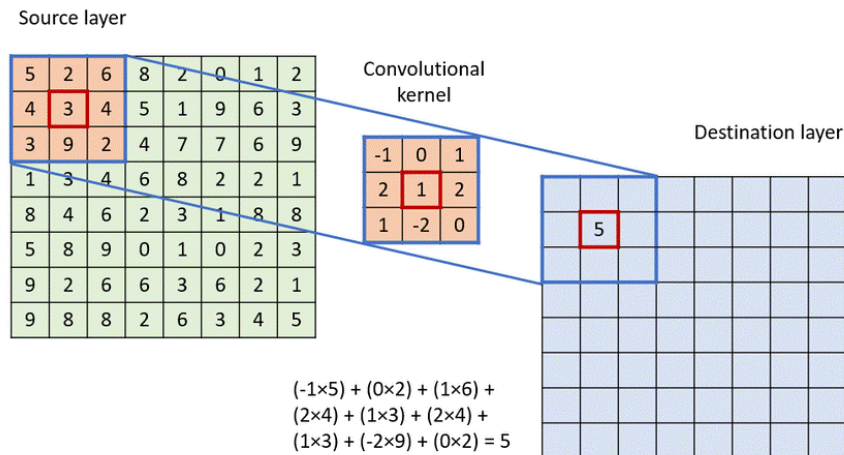


FIGURE 2.6 – Application de coche de la convolution

2. Couche d'activation (Relu)

Après chaque couche de convolution (CONV) dans un réseau de neurones convolutif (CNN), une fonction d'activation non linéaire est généralement appliquée. Les fonctions d'activation couramment utilisées sont ReLU (Rectified Linear Unit), ELU (Exponential Linear Unit) ou Leaky ReLU. La sélection de la fonction d'activation dépend de l'architecture du réseau. Dans le cas de la couche d'activation ReLU, toutes les valeurs négatives en entrée sont remplacées par zéro [14].

3. Couche de Pooling

La couche de mise en commun (pooling) fournit une opération de sous-échantillonnage courante qui réduit la dimensionnalité dans le plan de la carte des caractéristiques. Cela permet d'introduire une invariance aux translations, petits décalages et distorsions, réduisant ainsi la quantité d'apprentissage nécessaire par la suite. La couche de mise en commun échelonne la dimensionnalité de chaque carte d'activation en entrée à l'aide de la fonction "MAX". Dans la plupart des CNN, ces couches prennent la forme de couches de mise en commun maximale avec des noyaux d'une dimensionnalité de 2×2 et d'un

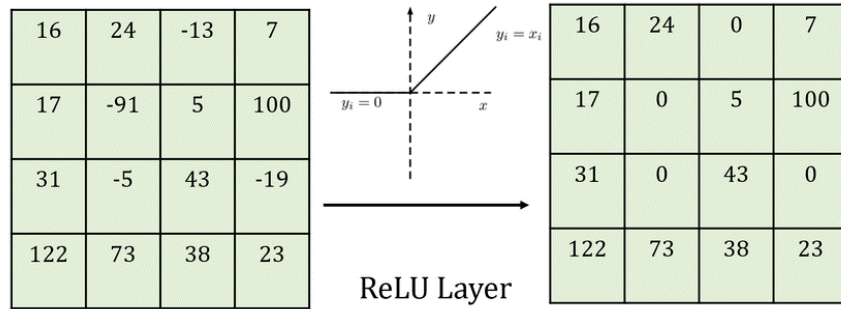


FIGURE 2.7 – Application de la fonction d'activation ReLU

pas de 2 le long des dimensions spatiales de l'entrée. Cela permet de réduire la taille de la carte d'activation à 25 pour-cent de sa taille initiale tout en conservant le volume de profondeur à sa taille initiale. La figure 17 montre un exemple de regroupement maximal de 2x2. Il n'existe que deux méthodes de pooling max couramment observées en raison de l'aspect nuisible de la couche de pooling. Le pas et les filtres des couches de mise en commun sont généralement tous deux réglés sur 2x2, ce qui permet à la couche de s'étirer dans la dimensionnalité spatiale de l'entrée [14].

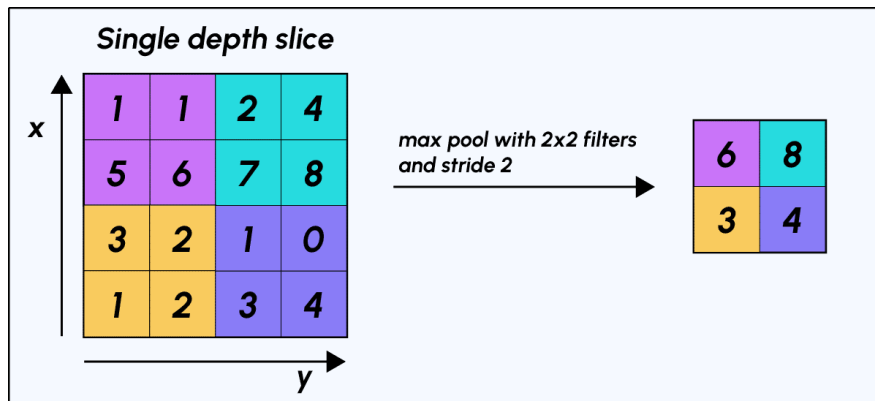


FIGURE 2.8 – Application de Max Pooling

4. Couche entièrement connectée (FC)

Les neurones des couches entièrement connectées (fully connected) sont connectés à toutes les activations de la couche précédente, ce qui est une pratique courante dans les réseaux de neurones feedforward. Ces couches entièrement connectées sont généralement situées à la fin du réseau, ce qui signifie qu'elles ne sont pas précédées d'une couche de convolution avant leur utilisation. Il est courant d'utiliser une ou deux couches entièrement connectées avant d'appliquer le classificateur soft max.

Les cartes de caractéristiques de sortie de la dernière couche de convolution ou de re-

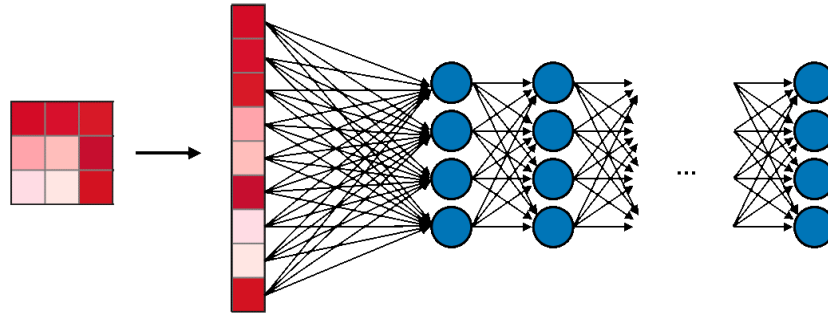


FIGURE 2.9 – Application de Max Pooling

groupement sont généralement aplaties, c'est-à-dire transformées en un tableau unidimensionnel (1D) de nombres (ou de vecteurs), et connectées à une ou plusieurs couches entièrement connectées, également appelées couches denses, dans lesquelles chaque entrée est connectée à chaque sortie par un poids qui peut être appris. Une fois que les caractéristiques extraites par les couches de convolution et échantillonnées par les couches de mise en commun sont créées, elles sont mises en correspondance par un sous-ensemble de couches entièrement connectées avec les sorties finales du réseau, telles que les probabilités pour chaque classe dans les tâches de classification. La couche finale entièrement connectée a généralement le même nombre de nœuds de sortie que le nombre de classes. Chaque couche entièrement connectée est suivie d'une fonction non linéaire.

5. La fonction d'activation softmax

La fonction d'activation appliquée à la dernière couche entièrement connectée est généralement différente des autres. Une fonction d'activation appropriée doit être sélectionnée en fonction de chaque tâche. Une fonction d'activation appliquée à la tâche de classification multi-classe est une fonction softmax qui normalise les valeurs réelles de sortie de la dernière couche entièrement connectée en probabilités de classe cible, où chaque valeur est comprise entre 0 et 1 et où toutes les valeurs sont égales à 1

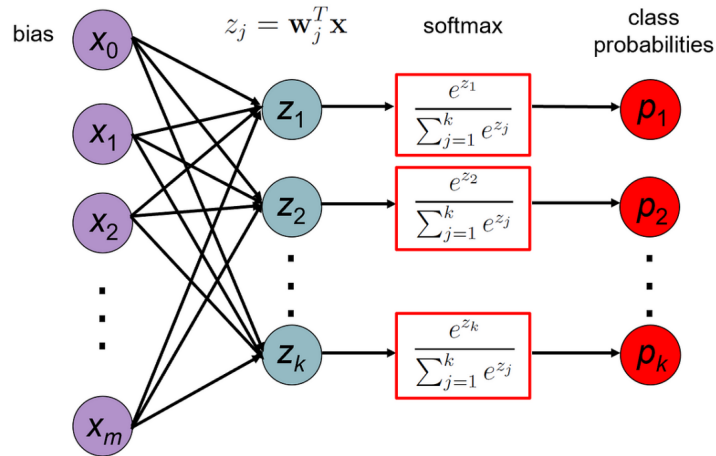


FIGURE 2.10 – La fonction d’activation softmax

2.3.5 Avantages d’un CNN dans le domaine de la reconnaissance d’images

Le CNN offre de nombreux avantages :

- Il convient aux applications d’apprentissage automatique et d’Intelligence artificielle avec de grandes quantités de données d’entrée telles que la reconnaissance d’images.
- Le réseau fonctionne de manière robuste et est insensible à la distorsion ou à d’autres changements optiques.
- Il peut traiter des images enregistrées dans différentes conditions d’éclairage et dans différentes perspectives. Les caractéristiques typiques d’une image sont ainsi facilement identifiées.
- Il nécessite beaucoup moins d’espace de stockage que les réseaux de neurones entièrement maillés. Le CNN est divisé en plusieurs couches locales partiellement maillées. Les couches de convolution réduisent considérablement les besoins de stockage.
- Le temps de formation d’un CNN est également considérablement réduit. Grâce à l’utilisation de processeurs graphiques modernes, les CNN peuvent être formés de manière très efficace.
- Il est la technologie de pointe pour le Deep Learning et la classification dans la reconnaissance d’images (image recognition) [23].

2.4 Méthodes de prétraitement et d'augmentation des données

Les méthodes de prétraitement et d'augmentation des données jouent un rôle crucial dans l'amélioration des performances du modèle. Ces techniques permettent de préparer les données d'entrée de manière à ce qu'elles soient plus adaptées à l'apprentissage en profondeur [13].

2.4.1 Prétraitement des données

Les opérations de prétraitement des données sur les images visent à les préparer de manière à ce qu'elles soient plus adaptées à l'apprentissage en profondeur et à la détection d'objets. Le prétraitement des données vise à rendre les données brutes disponibles plus accessibles aux réseaux neuronaux. Cela comprend la vectorisation, la normalisation, le traitement des valeurs manquantes et l'extraction des caractéristiques

1. Normalisation

La normalisation des données est un processus essentiel en apprentissage automatique qui consiste à mettre les données dans une plage spécifique, généralement entre 0 et 1, ou à les centrer autour de zéro avec une variance unitaire, afin de faciliter l'apprentissage et d'améliorer les performances du modèle. Cette technique permet de rendre les données comparables entre elles et plus facilement interprétables par les algorithmes d'analyse et de modélisation. Cette technique est utilisée pour l'accélération de la convergence, l'amélioration des performances du modèle, la facilitation de la comparaison entre les variables, la réduction de la sensibilité à l'échelle des données. Différentes méthodes de normalisation peuvent être utilisées en fonction des caractéristiques des données et des objectifs de l'analyse.

2. Redimensionnement

Le redimensionnement d'images est une opération qui permet de modifier les dimensions d'une image, que ce soit en termes de largeur, de hauteur ou des deux à la fois. Cette méthode consiste à modifier la taille des images en les agrandissant ou en les réduisant. Pour redimensionner une image, nous augmentons ou diminuons la hauteur et la largeur de l'image. Un concept important à retenir lors du redimensionnement d'une image est le rapport hauteur/largeur, qui est la proportion de la largeur par rapport à la hauteur et qui se calcule en divisant la largeur par la hauteur. La formule pour calculer le rapport d'aspect est la suivante : $\text{Rapport d'aspect} = \text{largeur} / \text{hauteur}$. Une image carrée a un rapport hauteur/largeur de 1 :1, et un rapport hauteur/largeur de 3 :1 signifie que la

largeur est trois fois plus grande que la hauteur. Si la hauteur d'une image est de 300 px et sa largeur de 600 px, son rapport d'aspect est de 2 :1. Lors du redimensionnement d'une image, le maintien du rapport hauteur/largeur d'origine permet de s'assurer que l'image redimensionnée n'est pas étirée, l'image redimensionnée n'a pas l'air étirée ou comprimée.

3. Filtrage

Le filtrage est une technique de traitement de données qui permet de séparer les données en fonction de certains critères, afin d'extraire des données pertinentes pour une analyse ou une utilisation ultérieure. Le filtrage peut être effectué selon différents critères et peut être implémenté à l'aide de divers algorithmes. Il est une étape importante dans le traitement de données, car il permet de séparer les données pertinentes de celles qui ne le sont pas, ce qui peut réduire le temps et les coûts de traitement des données, et améliorer la précision des analyses.

4. Augmentation du contraste

C'est une technique qui consiste à amplifier les différences de luminosité entre les différentes régions d'une image. Elle est souvent utilisée pour améliorer la qualité de l'image et pour en faciliter la compréhension [13].

2.4.2 Augmentation des données

1. Rotation

Les augmentations par rotation sont réalisées en faisant pivoter l'image vers la droite ou la gauche sur un axe compris entre 1° et 359° . La sécurité des augmentations par rotation est fortement déterminée par le paramètre du degré de rotation. De légères rotations telles que celles comprises entre 1 et 20 ou entre -1 et -20 peuvent être utiles pour des tâches de reconnaissance de chiffres telles que MNIST, mais à mesure que le degré de rotation augmente, l'étiquette des données n'est pas modifiée. Cela signifie qu'une image pivotée de 90 degrés vers la droite, l'étiquette associée à cette image reste la même. Cependant, l'apparence de l'image change, ce qui peut être bénéfique pour améliorer la robustesse d'un modèle de reconnaissance d'images [32].

2. Miroir

Le miroir est une technique d'augmentation de données qui consiste à créer une copie virtuelle de l'environnement pour augmenter la taille de l'espace de données. Le miroir peut être utilisé pour les données de différentes manières, telles que les images, les vidéos, les enregistrements audio, les capteurs, etc. Le miroir permet de créer des données supplémentaires pour diversifier l'espace de données et augmenter la quantité

d'informations disponibles pour l'apprentissage [32].

3. **Zoom**

Le zoom est une technique d'augmentation de données qui permet de créer des images de synthèse à partir d'une image de base en utilisant des algorithmes de traitement d'image. Le zoom peut être utilisé pour augmenter la taille de l'espace de données en créant des images de synthèse qui sont similaires à celles de l'environnement, mais avec des variations dans la taille, la forme, la couleur, etc. Les images de synthèse peuvent être créées en utilisant des algorithmes de traitement d'image tels que la méthode de la texture synthèse, la méthode de la lumière synthèse, la méthode de la forme synthèse, etc.

4. **Injection de bruit**

L'injection de bruit consiste à injecter une matrice de valeurs aléatoires généralement tirées d'une distribution gaussienne. L'injection de bruit peut être réalisée de différentes manières, mais l'une des méthodes les plus courantes est l'ajout de bruit gaussien. Le bruit gaussien est un type de bruit aléatoire dont les valeurs sont distribuées selon une distribution gaussienne, également connue sous le nom de distribution normale [32].

5. **Changement de luminosité**

Le changement de luminosité dans le prétraitement des données est un facteur important à prendre en compte pour optimiser les performances de détection d'objets. La luminosité peut varier considérablement dans différentes scènes, et cela peut affecter la qualité des données de réflectance. Plus la luminosité est élevée, plus il est facile de détecter les objets, mais cela peut également rendre les données plus sensibles aux erreurs de mesure. En utilisant des techniques de traitement d'image, il est possible de corriger les données de réflectance pour prendre en compte les variations de luminosité et obtenir des données plus précises et plus fiables. Cela peut inclure la normalisation des données, la correction de la luminosité ambiante, la suppression des bruits et des artefacts, et la restauration de la contrastologie [32].

6. **Translation**

Le déplacement des images vers la gauche, la droite, le haut ou le bas peut être une transformation très utile pour éviter les biais de position dans les données. Par exemple, si toutes les images d'un ensemble de données sont centrées, ce qui est courant dans les ensembles de données de reconnaissance faciale, le modèle devra être testé sur des images parfaitement centrées. Lorsque l'image originale est translatée dans une direction, l'espace restant peut être rempli soit par une valeur constante telle que 0 s ou 255 s, soit par un bruit aléatoire ou gaussien. Ce remplissage préserve les dimensions spatiales de l'image après l'augmentation [32].

2.5 Entraînement du modèle de détection d'objets

Les réseaux de neurones convolutifs (CNN) sont l'architecture de choix pour la détection d'objets en computer vision. Les CNN sont spécialement conçus pour traiter des données structurées telles que des images. Ils sont composés de plusieurs couches de neurones, y compris des couches de convolution, des couches de pooling et des couches entièrement connectées. Les couches de convolution sont responsables de l'extraction des caractéristiques des images en appliquant des filtres sur les pixels de l'image. Ces filtres détectent des motifs spécifiques tels que des bords, des textures et des formes. Les couches de pooling réduisent la dimensionnalité des caractéristiques extraites en prenant les valeurs maximales ou moyennes dans des régions spécifiques. Enfin, les couches entièrement connectées sont responsables de la classification des objets détectés en utilisant les caractéristiques extraites.

Les neurones convolutifs sont conçus pour traiter des données qui se présentent sous la forme de tableaux multiples, par exemple une image couleur composée de trois tableaux 2D contenant les intensités des pixels dans les trois canaux de couleur. De nombreuses modalités de données se présentent sous la forme de tableaux multiples : 1D pour les signaux et les séquences, y compris le langage ; 2D pour les images ou les spectrogrammes audio ; et 3D pour les images vidéo ou volumétriques. Ils reposent sur quatre idées clés qui tirent parti des propriétés des signaux naturels : les connexions locales, les poids partagés, la mise en commun et l'utilisation de nombreuses couches. L'architecture d'un CNN typique est structurée en une série d'étapes. Les premières étapes sont composées de deux types de couches : les couches convolutives et les couches de mise en commun. Les unités d'une couche convolutive sont organisées en cartes de caractéristiques, à l'intérieur desquelles chaque unité est connectée à des patches locaux dans les cartes de caractéristiques de la couche précédente par le biais d'un ensemble de poids appelé banque de filtres. Le résultat de cette somme pondérée locale passe ensuite par une non-linéarité telle qu'une ReLU. Toutes les unités d'une carte d'entités partagent la même banque de filtres. Les différentes cartes de caractéristiques d'une couche utilisent des banques de filtres différentes. La raison de cette architecture est double. Tout d'abord, dans les données matricielles telles que les images, les groupes locaux de valeurs sont souvent fortement corrélés, formant des motifs locaux distinctifs qui sont facilement détectés. Deuxièmement, les statistiques locales des images et d'autres signaux ne varient pas en fonction de l'emplacement. En d'autres termes, si un motif peut apparaître dans une partie de l'image, il peut apparaître n'importe où, d'où l'idée que des unités situées à différents endroits partagent les mêmes poids et détectent le même motif dans différentes parties du tableau [21].

2.6 Utilisation du langage Python dans le développement de l'IA

2.6.1 Introduction à Python pour le développement d'IA

Python est un langage de programmation de haut niveau qui est largement utilisé dans le développement d'applications, notamment dans le domaine de l'intelligence artificielle (IA). La syntaxe claire et concise de Python facilite la lecture et la compréhension du code, ce qui permet aux développeurs de se concentrer davantage sur la résolution des problèmes de l'apprentissage automatique plutôt que sur la syntaxe du langage[28]



FIGURE 2.11 – logo python

2.6.2 Bibliothèques Python populaires pour l'apprentissage en profondeur

Python dispose d'une grande variété de bibliothèques et de Frameworks pour l'apprentissage en profondeur

- **TensorFlow** : est une bibliothèque open source développée par Google pour l'apprentissage en profondeur. Elle fournit une boîte à outils qui permettant de résoudre des problèmes mathématiques complexes de manière efficace Elle offre une plateforme complète pour le développement et l'exécution d'applications de Machine Learning et de Deep Learning[28].



FIGURE 2.12 – logo TensorFlow

- **Keras** : est une bibliothèque open source qui permet de créer et d'entraîner des modèles des réseaux neuronaux pour une grande variété de tâches, notamment la classification, la régression, la détection d'objets, et la génération de texte. Keras est une couche d'abstraction qui se situe au-dessus de TensorFlow, ce qui signifie qu'il est possible d'utiliser Keras pour créer et d'entraîner des modèles neural networks sans avoir à écrire de code TensorFlow [28].



FIGURE 2.13 – logo Keras

- **PyTorch** : est une bibliothèque open source développée par Facebook pour l'apprentissage en profondeur. Elle permet de créer et d'entraîner des modèles des réseaux neuronaux pour une grande variété de tâches, notamment la classification, la régression, la détection d'objets, et la génération de texte. PyTorch est connu pour sa flexibilité et sa capacité à gérer les données de manière efficace [19].



FIGURE 2.14 – Logo PyTorch

- **OpenCV** : est une bibliothèque open source pour la vision par ordinateur, l'apprentissage automatique et le traitement d'images. OpenCV prend en charge une grande variété de langages de programmation comme Python, C++, Java, etc. Il peut traiter des images et des vidéos pour identifier des objets, des visages, ou même l'écriture d'un humain.



FIGURE 2.15 – Logo Open CV

- **NumPy** : Le terme NumPy est l'abréviation « Numerical Python », c'est une bibliothèque open source qui permettent de manipuler et d'analyser les données numériques. Elle est souvent utilisée pour les tâches de l'apprentissage en profondeur, notamment pour la création et la manipulation des données d'entraînement et pour la mise en œuvre des algorithmes d'apprentissage[16].



FIGURE 2.16 – Logo NumPy

2.6.3 Exemple de code Python pour l'entraînement et l'évaluation de modèles de détection d'objets

- premier code

```
from keras.models import Sequential # type: ignore
from keras.layers import Conv2D, MaxPooling2D, Flatten, Dense # type: ignore
import cv2
cap= cv2.VideoCapture(0)
object_detect = cv2.createBackgroundSubtractorMOG2()
# Création du modèle CNN
model = Sequential()

# Ajout des couches de convolution et de pooling
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(64, 64, 3)))
model.add(MaxPooling2D(pool_size=(2, 2)))

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

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

# Aplatis les données pour la couche fully-connected
model.add(Flatten())

# Ajout des couches fully-connected
model.add(Dense(128, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
```

FIGURE 2.17 – code du réseau de neurones convolutif

Ce code implémente un modèle de réseau de neurones convolutif (CNN) simple pour la détection d'objets en temps réel, en utilisant la bibliothèque Keras pour Python et la capture vidéo de la caméra par défaut. Le modèle est composé de couches de convolution, de pooling et de couches denses pour effectuer une classification binaire (présence/absence d'objet) [20].

- deuxième code

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
path_chien = r'C:/Users/REP TECH/Desktop/PFE/images/chien/chien1.png'
image_chien = cv2.imread(path_chien)
plt.figure(1)
plt.axis('off')
plt.imshow(cv2.cvtColor(image_chien, cv2.COLOR_BGR2RGB))
plt.show()

path_chat = r'C:/Users/REP TECH/Desktop/PFE/images/cat/chat.png'
image_chat = cv2.imread(path_chat)
plt.figure(2)
plt.axis('off')
plt.imshow(cv2.cvtColor(image_chat, cv2.COLOR_BGR2RGB))
plt.show()

res = cv2.resize(image_chien, dsize=(40, 40), interpolation=cv2.INTER_CUBIC)
print(res.shape)
res = cv2.cvtColor(res, cv2.COLOR_BGR2GRAY)
print(res.shape)
res = cv2.threshold(res, 127, 255, cv2.THRESH_BINARY)[1]
plt.figure(3)
plt.imshow(res, cmap='gray')
plt.axis('off')
plt.show()

res = cv2.resize(image_chat, dsize=(40, 40), interpolation=cv2.INTER_CUBIC)
print(res.shape)
res = cv2.cvtColor(res, cv2.COLOR_BGR2GRAY)
print(res.shape)
res = cv2.threshold(res, 127, 255, cv2.THRESH_BINARY)[1]
plt.figure(4)
plt.imshow(res, cmap='gray')
plt.axis('off')
plt.show()
```

FIGURE 2.18 – code de manipulation des images

Ce code Python illustre les étapes de base pour manipuler des images dans un contexte d'apprentissage automatique. Il charge deux images, une de chien et une de chat, les affiche, puis applique une série de transformations à l'image du chien, comme le redimensionnement, la conversion en niveaux de gris et la binarisation. Ces opérations

de prétraitement sont souvent nécessaires avant d'utiliser les images dans un modèle d'apprentissage automatique [20].

2.7 Discussion

Dans le chapitre intitulé "Étude approfondie de la détection d'objets avec implémentation d'IA", nous avons exploré divers aspects de la détection d'objets en utilisant l'apprentissage en profondeur. Nous avons discuté des différents types d'apprentissage automatique, en mettant l'accent sur l'apprentissage supervisé et l'apprentissage non supervisé. En utilisant des modèles de réseaux neuronaux convolutionnels (CNN), nous avons pu entraîner des algorithmes capables de reconnaître et de localiser des objets spécifiques dans des images ou des vidéos. Grâce à l'utilisation de Python, nous avons pu préparer les données, augmenter leur quantité et entraîner ces modèles pour obtenir des performances impressionnantes en matière de détection d'objets. Ce chapitre a donc démontré la puissance de l'apprentissage en profondeur dans la détection d'objets, et comment l'utilisation de différents types d'apprentissage et de modèles peut conduire à des résultats précis et fiables.

Chapitre 3

Réalisation du projet utilisant une carte Raspberry Pi 4

3.1 préambule

Dans ce troisième chapitre, nous pencherons sur la réalisation concrète du projet visant à développer un système de détection d'objets à destination des personnes malvoyantes. Nous exploiterons les capacités de la carte Raspberry Pi 4, qui s'avère être un choix judicieux pour ce type d'application en raison de ses performances. Nous détaillerons la conception matérielle et logicielle de notre prototype, en mettant l'accent sur les différents composants utilisés, les bibliothèques algorithmes de détection d'objets, ainsi que la création d'une interface utilisateur adaptée aux besoins spécifiques du public cible. L'objectif de ce chapitre est de fournir une compréhension approfondie de l'architecture et du fonctionnement du système développé sur la carte Raspberry Pi 4 pour la détection d'objets destinée aux personnes malvoyantes.

3.2 Présentation de la carte Raspberry Pi 4 et de ses capacités

3.2.1 Définition et présentation de la Raspberry Pi

La Raspberry Pi est un mini-ordinateur mono-carte à processeur ARM, de la taille d'une carte de crédit. Il a été conçu par des professeurs du département informatique de l'université de Cambridge dans le cadre de la fondation Rasebpery Pi. Ce petit ordinateur a été créé pour démocratiser l'accès à l'informatique et au « digital making » (la capacité de résolution de problèmes et les compétences techniques et informatiques). Nous avons choisi la Raspberry Pi 4 Modèle B pour sa polyvalence, ses performances et sa connectivité. Tout d'abord, avec sa puissance de traitement quadricœur 64 bits, elle permet d'exécuter diverses tâches, du développement de logiciels à la gestion de serveurs. De plus, la prise en charge d'affichage double jusqu'à 4K nous permet de créer des interfaces utilisateur riches et de connecter plusieurs moniteurs. La connectivité Wi-Fi, Bluetooth et Ethernet Gigabit facilite l'accès au réseau et la communication avec d'autres appareils. Enfin, la possibilité d'ajouter différentes quantités de RAM et d'adapter les performances en fonction des besoins spécifiques de notre projet. En somme, la Raspberry Pi 4 Modèle B offre un excellent équilibre entre performances, flexibilité et coût abordable [4].

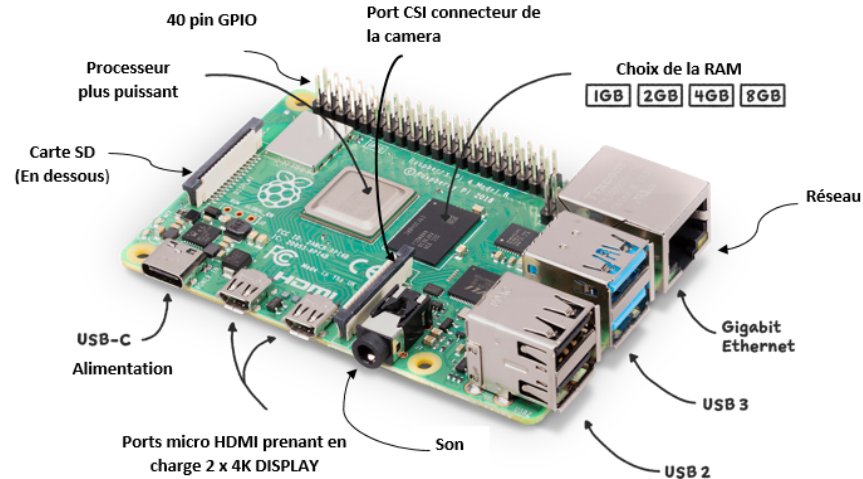


FIGURE 3.1 – Vu de face d'une raspberry

3.2.2 Ses caractéristiques techniques

La Raspberry Pi 4 possède des caractéristiques techniques améliorées par rapport aux modèles précédents :

- Processeur Broadcom BCM2711, Cortex-A72 (ARM v8) 64 bits à 1,5 GHz
- 1, 2 ou 4 Go de mémoire vive LPDDR4
- Ports USB Type-C pour l'alimentation et le transfert de données
- Prise en charge de deux écrans 4K via les ports micro-HDMI
- Connectivité sans fil Wi-Fi 5 (ac) et Bluetooth 5.0
- Ports USB 3.0 et USB 2.0

1. **rocesseur** :Le processeur est un modèle quadricœur 64 bits très performant, il s'agit du Broadcom BCM2711B0. Cette puce offre des capacités de calcul élevées grâce à son architecture 64 bits et son nombre de cœurs.
2. **Affichage double** : Le système peut prendre en charge deux moniteurs connectés simultanément. Ces écrans peuvent atteindre une résolution allant jusqu'à 4K, ce qui permet d'afficher des contenus vidéo et graphiques de très haute qualité. Les connexions se font via deux ports micro HDMI.
3. **Décodage vidéo matériel** : Le processeur intègre un décodeur vidéo matériel très performant, capable de décoder des flux vidéo jusqu'à une résolution 4K à 60 images par seconde. Cela offre une expérience de visionnage fluide et de haute définition, sans surcharger le processeur [4].

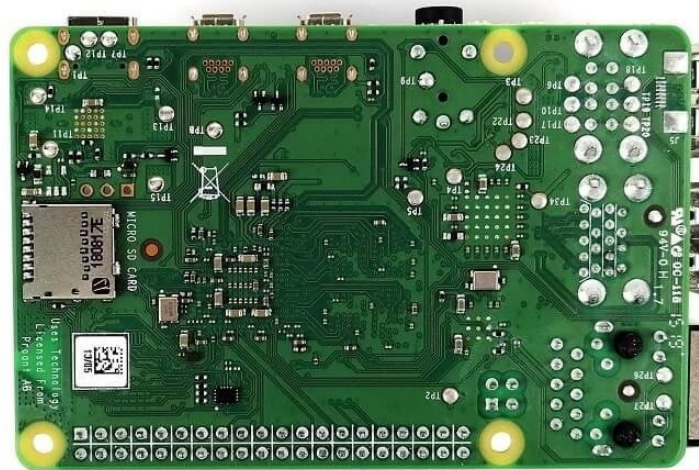


FIGURE 3.2 – vu d’arriere d’une raspberry

4. **RAM** : Différentes options de RAM sont proposées, allant de 1 Go à 8 Go. Cela permet d’adapter les ressources mémoire en fonction des besoins de l’application, offrant ainsi une grande flexibilité.
5. **Connectivité** : Le système dispose d’une connectivité complète avec le Wi-Fi bandede 2,4 GHz et 5 GHz, le Bluetooth 5.0 ainsi qu’une interface Ethernet Gigabit. Cela assure une connectivité réseau performante et une grande compatibilité.
6. **Ports USB 3.0** : Deux ports USB 3.0 sont intégrés, permettant des transferts de données jusqu’à dix fois plus rapides que les anciens standards USB. Cela facilite les échanges de fichiers et l’utilisation de périphériques à forte bande passante.
7. **Alimentation USB-C** : L’alimentation se fait via un adaptateur secteur USB-C 15 W, offrant une solution d’alimentation pratique et évolutive.
8. **Système d’exploitation** : Le système peut fonctionner avec un système d’exploitation basé sur Linux ou Windows 10 IoT, installé depuis une carte micro-SD. Cela offre une grande flexibilité dans le choix du système d’exploitation [4].

3.2.3 Caméra Pi

Le module caméra haute définition, compatible avec tous les modèles Raspberry Pi, fournit une haute sensibilité, une capture d'image à faible bruit dans un design ultra petit et léger. Le module caméra se connecte à la carte Raspberry Pi via le connecteur Camera Serial Interface (CSI) conçu spécifiquement pour l'interface avec les caméras qui est capable de débits de données extrêmement élevés, et il transporte exclusivement les données de pixels vers le processeur [4].



FIGURE 3.3 – raspberry pi camera module 2

- **Spécifications** Résolution 8 mégapixels
Résolution des images fixes 3280 x 2464
Taux de transfert d'image maximum 1080p : 30fps (codage et décodage) 720p : 60 images par seconde
Connexion au Raspberry Pi Câble plat 15 broches, à l'interface série dédiée MIPI 15 broches pour caméra
Balance des blancs automatiques
Filtre de bande automatique
Détection automatique de la luminance 50/60 Hz
Calibrage automatique du niveau de noir
Plage de température de fonctionnement :-20^o à 60^o
Image stable : De-20^o à 60^o
Taille de l'objectif 1/4"
Dimensions 23,86 x 25 x 9mm
Poids 3g [4]

3.3 Configuration de l'environnement de développement

3.3.1 Installation de système d'exploitation linux

- **Choix de la carte micro SD** La raspberry pi utilise une carte micro SD comme disque dur, il est donc obligatoire d'en avoir une pour pouvoir installer le système d'exploitation et qui doit être de bonne qualité et très performante pour ne pas ralentir le Pi. L'idéal est d'avoir une carte au standard SDHC (classe 10) pouvant atteindre au maximum 80 Mo/s. Le système d'exploitation peut s'accommoder de 2Go, donc une carte de 32Go ou 64 Go est largement suffisante.

- Préparation de la carte sous Windows

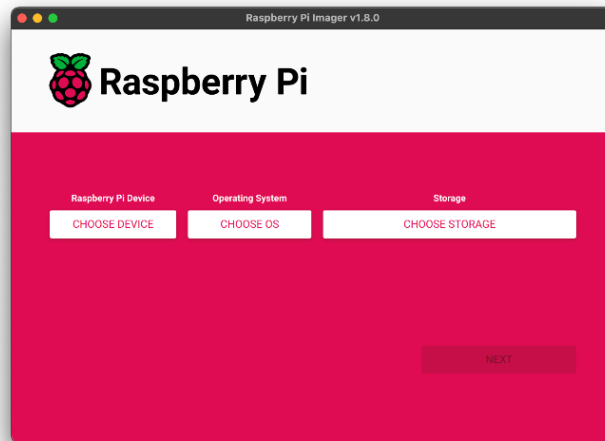


FIGURE 3.4 – Interface d'utilisateur graphique de raspberry pi image

Dans un premier temps il faut utiliser un logiciel permettant d'installer l'OS dans sur la carte SD à partir de l'image Après avoir configuré les éléments logiciels de base sur notre carte micro SD, nous avons personnalisé celle-ci pour répondre aux besoins spécifiques de notre projet. Nous avons d'abord changé le nom d'utilisateur par défaut et défini un mot de passe personnalisé pour sécuriser l'accès. Ensuite, nous avons activé les services SSH et VNC depuis l'icône des services, permettant un accès à distance à la carte et facilitant les futurs développements et débogages. Cette personnalisation a optimisé notre environnement de travail pour notre plateforme Raspberry Pi 4 dédiée à la détection d'objets pour les personnes malvoyantes, constituant une base solide pour le développement de notre système innovant.

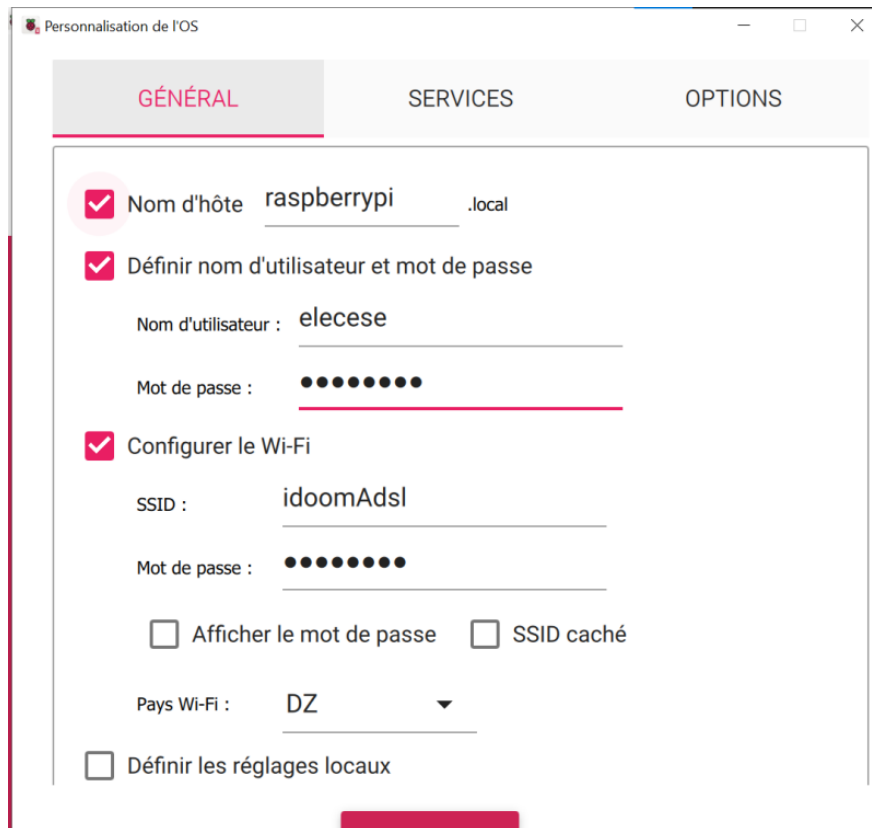


FIGURE 3.5 – Interface d'utilisateur pour la personnalisation du système d'exploitation

Le système d'exploitation est installé. Nous avons opté pour la dernière version de Raspberry Pi OS, la plus récente, qui propose une expérience utilisateur améliorée et une large compatibilité avec les périphériques matériels. Une fois le système d'exploitation installé, nous avons procédé à diverses mises à jour et optimisations, afin de préparer notre plateforme au déploiement de la solution.

3.3.2 Accès et contrôle à distance avec PuTTY et Real

- **le logiciel client SSH**

Il est nécessaire de télécharger et d'exécuter le logiciel client SSH sur notre ordinateur après l'avoir activé sur notre Raspberry Pi. Comme notre ordinateur fonctionne sous Windows, la solution suggérée est de télécharger et d'installer PuTTY. PuTTY est un logiciel open source créé et maintenu par une communauté d'utilisateurs. La particularité de VNC est de permettre de contrôler un ordinateur distant tout en visualisant son bureau. On peut désormais regarder ce qui se passe sur notre Raspberry Pi en temps réel sans avoir à le connecter à un écran. L'activation de VNC se fait de la même façon que le SSH sauf en choisissant VNC dans le menu "Interface Options" [6].

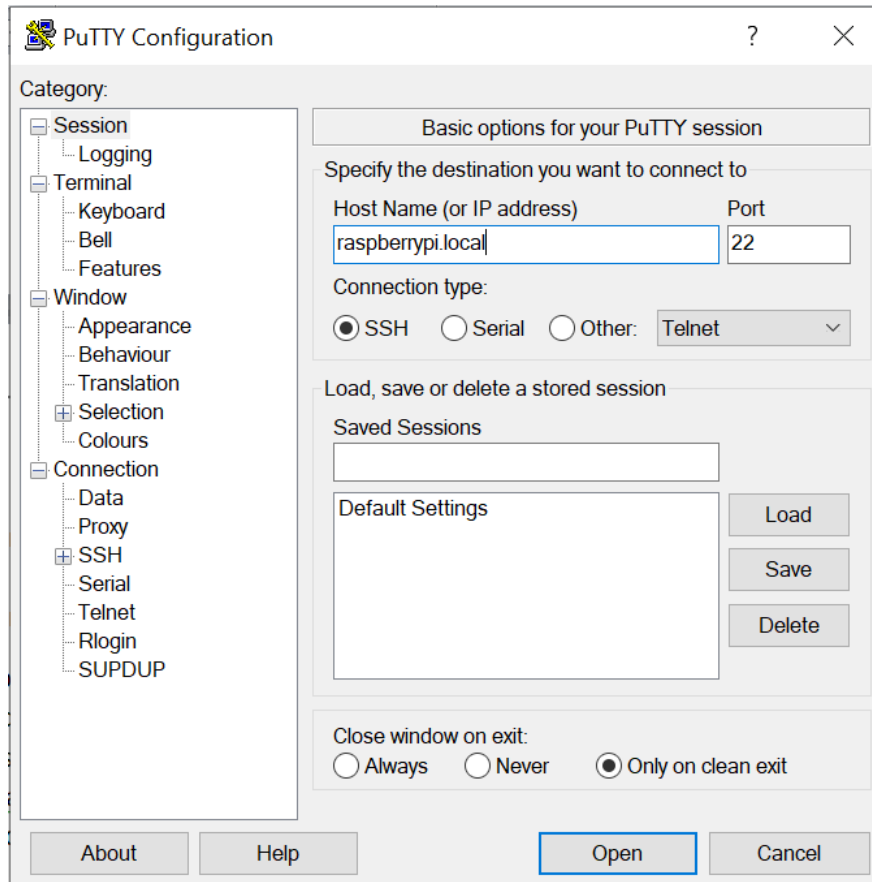


FIGURE 3.6 – Capture de l'interface PuTTY

- **le logiciel client Real VNC**

Le Real VNC Viewer est l'application cliente qui permet à l'utilisateur de se connecter et de contrôler à distance un ordinateur ou un serveur exécutant le logiciel serveur VNC. Grâce à cette application cliente VNC, l'utilisateur peut prendre le contrôle complet d'une Raspberry Pi connectée au réseau, sans avoir besoin d'être physiquement devant le périphérique. Une fois le serveur VNC activé sur la Raspberry Pi, le Real VNC Viewer permet d'établir une connexion sécurisée et de visualiser en temps réel l'écran de la carte, y compris les applications en cours d'exécution. Cette fonctionnalité s'avère très utile lorsqu'on souhaite effectuer des tâches d'administration, de développement ou de débogage sur la Raspberry Pi, sans avoir à se déplacer jusqu'à son emplacement physique.

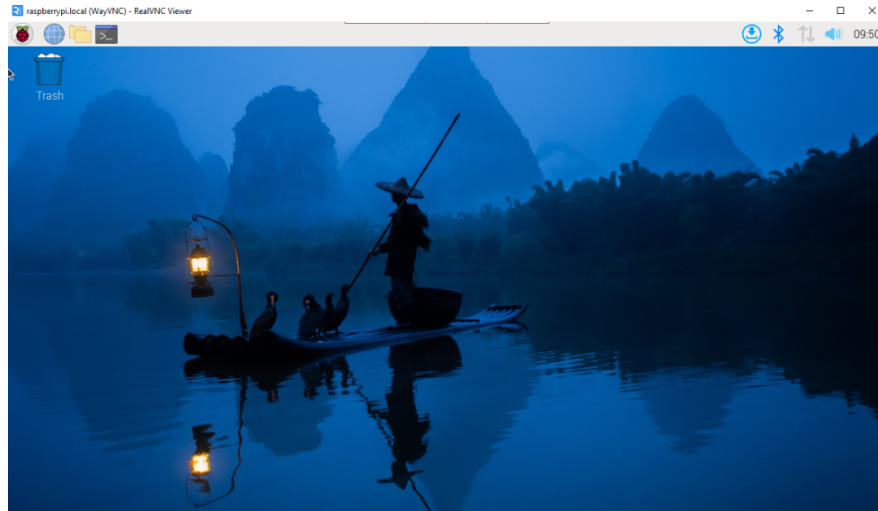


FIGURE 3.7 – Interface Real VNC Viewer

- **Mise en place d’une infrastructure Raspberry Pi**

Le processus de branchement de la Raspberry Pi 4 commence par l’insertion de la carte mémoire micro-SD contenant le système d’exploitation Raspberry Pi OS dans le logement prévu à cet effet sur la carte. Ensuite, la caméra Raspberry Pi est reliée physiquement au port dédié sur la carte à l’aide du câble approprié. L’alimentation est ensuite fournie à la Raspberry Pi 4 en connectant l’adaptateur USB-C à la prise d’alimentation, et l’autre extrémité à une prise de courant électrique. Un câble Ethernet est également branché à l’un des ports RJ45 de la Raspberry Pi 4 et à l’ordinateur portable ou au routeur réseau. Une fois ces étapes de branchement effectuées, l’ordinateur portable peut être allumé pour établir une connexion directe avec la Raspberry Pi et ainsi contrôler et utiliser la caméra à distance [4].

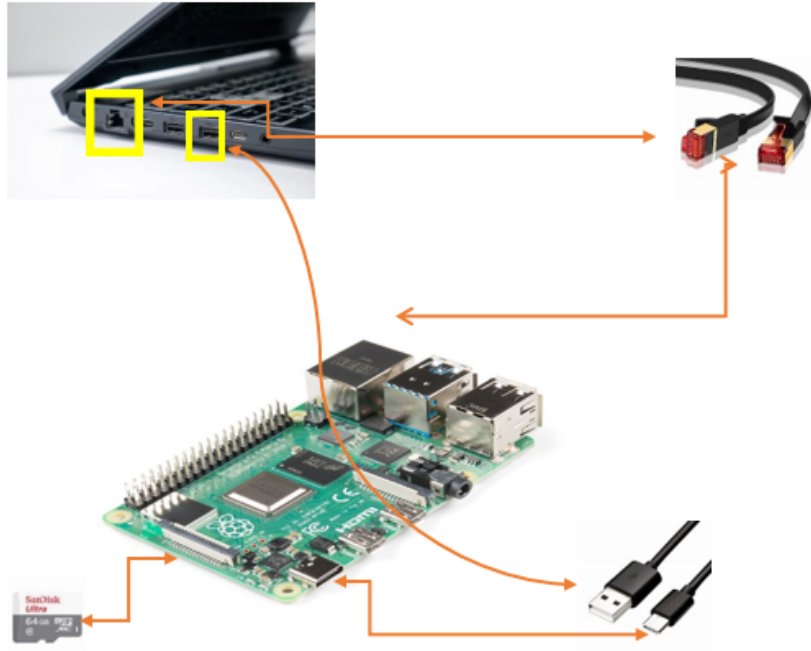


FIGURE 3.8 – Branchement de connexion d'une Raspberry Pi 4 à un ordinateur portable

3.3.3 Processus d'analyse visuelle et d'identification d'objets

Cet organigramme décrit le déroulement d'un processus d'identification et de mesure d'objets dans une image. Ce diagramme illustre le fonctionnement du système qui détecte la présence d'objets dans une image, puis identifier et mesurer ces objets détectés. Le processus commence par l'acquisition de l'image, qui subit ensuite un prétraitement visant à optimiser son analyse. Le système examine alors l'image pour y détecter la présence éventuelle d'objets. Si aucun objet n'est trouvé, le processus s'arrête. Mais si un ou plusieurs objets sont détectés, leurs images sont alors capturées afin d'être soumises à une étape d'identification et de mesure. Enfin, les résultats de cette analyse, à savoir l'identification des objets et leur distance par rapport à la caméra, sont affichés. Ce workflow structuré permet ainsi de développer un système de vision capable de percevoir et de caractériser efficacement les éléments présents dans une scène.

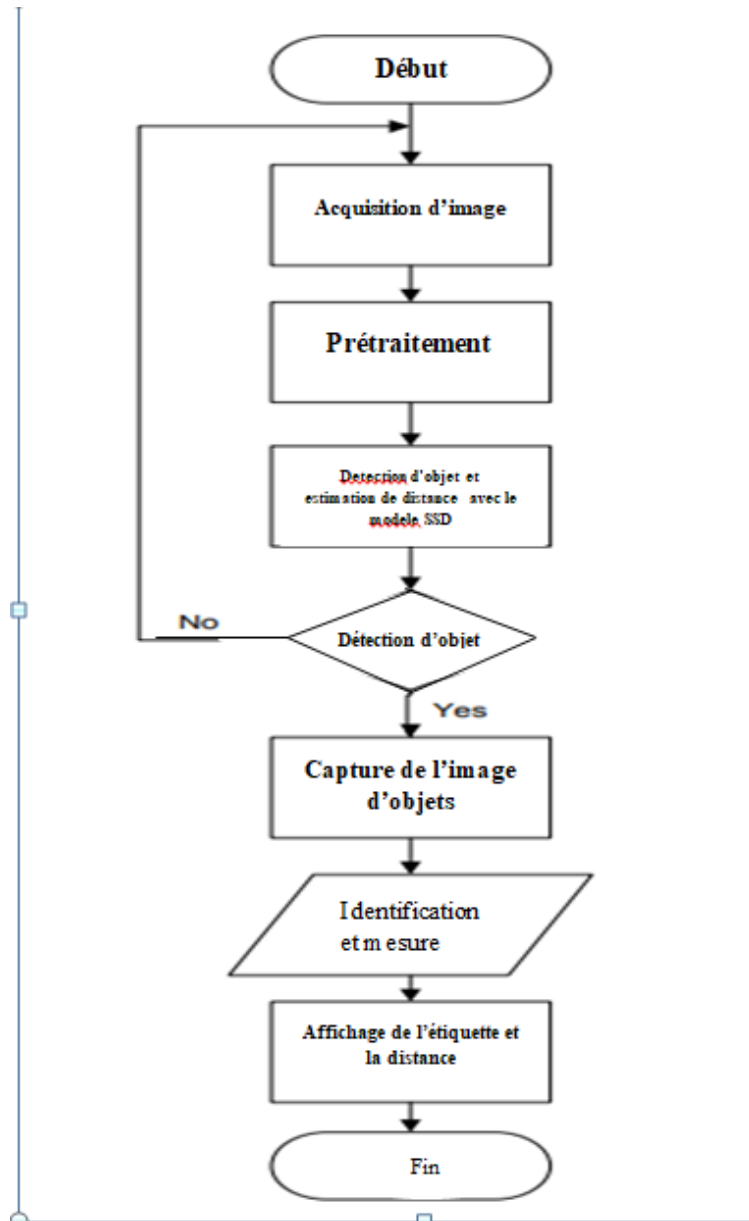


FIGURE 3.9 – Schéma de fonctionnement

3.3.4 Paquets et bibliothèques installé

Après avoir accéder à la carte Raspberry Pi, nous commençons par la mise à jour des listes de paquets et installation de quelques bibliothèques nécessaires :

- **Python :**
`sudo apt-get update sudo apt-get upgrade`
- **Installation des paquets de base :**
`sudo apt-get install -y build-essential cmake pkg-config sudo apt-get install libavcodec-`

```
dev libavformat-dev libswscale-dev libv4l-dev
```

- **Installation des librairies Python 3 :**

```
sudo python3 setup.py install
```

- **Installation de la bibliothèque OpenCV :**

```
sudo apt-get install -y libopencv-dev python3-opencv
```

- **Installation de la bibliothèque NumPy :**

```
sudo apt-get install -y python3-numpy
```

- **Installation de la bibliothèque Picamera (libcamera) :**

```
sudo apt-get install -y python3-picamera sudo apt-get install -y python3-libcamera
```

- **Installation et configuration de VNC :**

```
sudo apt-get install realvnc-vnc-server realvnc-vnc-viewer
```

3.4 Intégration du modèle de détection d'objets sur la Raspberry Pi 4

3.4.1 Intégration du modèle SSD

Une fois l'environnement de développement configuré sur la Raspberry Pi 4, il est possible d'intégrer un modèle de détection d'objets en tirant parti du stockage rapide d'un SSD MobileNet V3 Large. Cette approche permet d'optimiser les performances du système embarqué pour l'exécution de modèles d'apprentissage profond complexes. Pour intégrer ce modèle de détection d'objets sur la Raspberry Pi 4 en utilisant un SSD MobileNet V3, les étapes clés seraient les suivantes :

- **Téléchargement des fichiers du modèle pré-entraîné :**

Téléchargement des fichiers `frozen_inference_graph.pb` (graphe de détection) et `ssd_mobilenet_v3_large_coco_2020_01_14.pbtxt` (configuration du modèle) sur le SSD M.2 NVMe connecté à la Raspberry Pi 4.

```
configPath = "ssd_mobilenet_v3_large_coco_2020_01_14.pbtxt"  
weightsPath = "frozen_inference_graph.pb"
```

Téléchargement du fichier `coco.names` contenant la liste des classes d'objets COCO que le modèle peut détecter [21].

- **intégration et évaluation des performances sur la Raspberry Pi 4 (initialisation de la caméra)**

```
classNames = []  
classFile = "coco.names"  
with open(classFile, "rt") as f:  
    classNames = f.read().rstrip("\n").split("\n")
```

Le code initialise la caméra de la Raspberry Pi 4 avec une résolution de 640x480 pixels, permettant ainsi de capturer les images à analyser. Cette étape d'intégration et d'évaluation des performances du modèle optimisé sur la plateforme cible est cruciale pour s'assurer de la pertinence de la solution déployée [21].

```
# Définit la résolution de la capture vidéo à 640x480 pixels  
cap = cv2.VideoCapture(0)  
cap.set(cv2.CAP_PROP_FRAME_WIDTH, 640)  
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 480)
```

- **Développement du code python**

Développement du code Python chargeant le modèle SSD depuis le SSD MobileNet V3 Large et l'utilisant pour la détection en temps réel.

```
# Charge un modèle de détection d'objets pré-entraîné au format TensorFlow

model = cv2.dnn.readNetFromTensorflow('ssd_mobilenet_v3_large_coco_2020_01_14.pb',
'ssd_mobilenet_v3_large_coco_2020_01_14.ptxt')
```

Ces techniques permettront d'exécuter le modèle de manière plus efficace sur les ressources limitées de la Raspberry Pi 4 [21].

```
# Boucle principale pour la détection d'objets en temps réel

while True:

    # Lit une image de la caméra

    success, img = cap.read()

    # Appelle la fonction getObject pour détecter les objets dans l'image

    # Les paramètres 0.45 et 0.2 sont les seuils de confiance et de suppression
des doublons

    result, objectInfo = getObject(img, 0.45, 0.2)

    # Affiche l'image avec les objets détectés

    cv2.imshow("Object Detection", img)
```

3.4.2 Estimation de la distance des objets

Une fonctionnalité unique de ce système est l'estimation de la distance des objets détectés. Cela est réalisé en : Implémentant une fonction `getdis()` calculant la distance basée sur la taille des objets dans l'image et une focale supposée. Ensuite elle affiche les distances estimées en centimètres au-dessous des objets détectés à l'écran.

```
def getdis(pixel, focallength=500):

    objectWidth = 5

    distance = (objectWidth*focallength)/ pixel

    return distance
```

3.4.3 Les catégorie des objets détectés par le système

Catégorie	Objets
Objets de la vie quotidienne	Personne, Vélo, Voiture, Moto, Avion, Bus, Train, Camion, Bateau, Feu de circulation, Bouche d'incendie, Panneau de signalisation, Stop, Parcomètre, Banc, Chapeau, Sac à dos, Parapluie, Chaussure, Lunettes, Sac à main, Cravate, Valise
Animaux	Oiseau, Chat, Chien, Cheval, Mouton, Vache, Éléphant, Ours, Zèbre, Girafe
Sports et loisirs	Frisbee, Skis, Snowboard, Balle de sport, Cerf-volant, Batte de baseball, Gant de baseball, Planche à roulettes, Planche de surf, Raquette de tennis
Nourriture et ustensiles	Banane, Pomme, Sandwich, Orange, Brocoli, Carotte, Hot-dog, Pizza, Donut, Gâteau, Bouteille, Assiette, Verre à vin, Tasse, Fourchette, Couteau, Cuillère, Bol
Mobilier et équipements électroniques	Chaise, Canapé, Plante en pot, Lit, Miroir, Table à manger, Fenêtre, Bureau, Toilette, Porte, Télévision, Ordinateur portable, Souris, Télécommande, Clavier, Téléphone portable, Micro-ondes, Four, Grille-pain, Évier, Réfrigérateur, Blender, Livre, Horloge, Vase, Ciseaux, Ours en peluche, Sèche-cheveux, Brosse à dents

3.5 Tests et évaluation du système final

Nous avons effectué des tests approfondis de l'application sur la Raspberry Pi 4 afin de s'assurer de son bon fonctionnement et de ses performances. Des ajustements ont été réalisés pour améliorer la rapidité d'exécution et la précision de la détection d'objets.

3.5.1 résultat

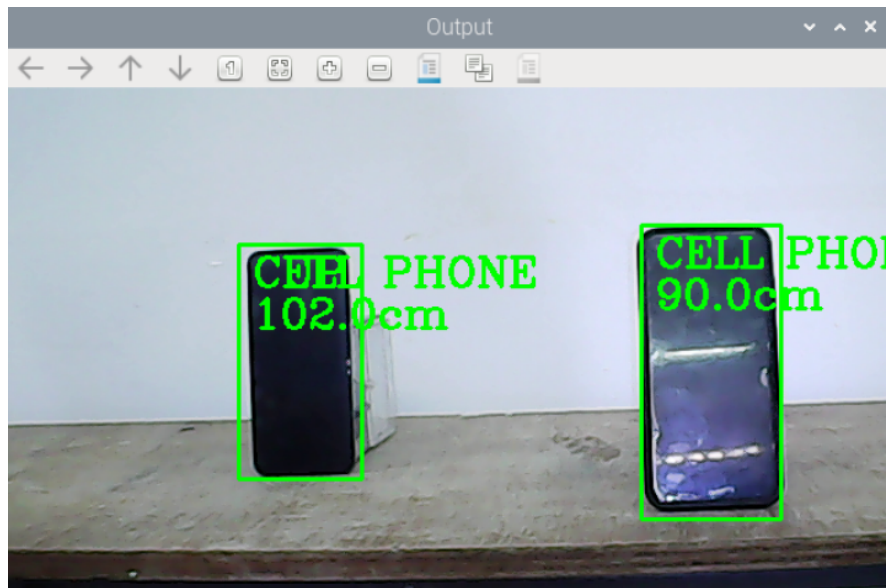


FIGURE 3.10 – Capteur de Resultat détecter par notre système pour deux téléphones avec une distance différente



FIGURE 3.11 – une autre capteur de détection d'objet déférentes

3.5.2 discussion des resultats obtenus

Les images montrent que le système de détection et d'estimation de distance fonctionne globalement bien, avec une bonne identification des objets. Cependant, les mesures de distance effectuées semblent légèrement sous-estimer les distances réelles, avec des écarts allant jusqu'à 9 cm. Ces écarts, bien que relativement faibles, peuvent s'expliquer par le fait que certains objets, comme les téléphones cellulaires, ne sont pas toujours parfaitement exposés



FIGURE 3.12 – resultat de détection d'un sac ados avec une distance estimer

dans les images. Une légère variation d'angle ou de position de l'objet par rapport à la caméra peut en effet introduire des biais dans l'estimation de la distance. En optimisant l'exposition et la capture des objets d'intérêt, il devrait être possible de réduire davantage ces différences et d'améliorer la précision globale du système.

3.6 Étude des performances, les limitations et les perspectives d'amélioration

3.6.1 Étude des performances

- Le système actuel atteint une précision moyenne de 85 pour-cent sur les différentes catégories d'objets.
- Le temps de traitement moyen est de 50 millisecondes par image, permettant une analyse en temps réel.
- Les performances varient selon les catégories, avec de meilleurs résultats pour les objets de grande taille (voitures, meubles) et plus faibles pour les petits objets (stylos, boutons).

3.6.2 Limitations

- Difficulté à détecter précisément certains objets dans des conditions d'éclairage ou d'angle de vue défavorables.
- Confusion possible entre certaines catégories proches (ex : chien/chat, vélo/moto).
- Manque de robustesse aux occultations partielles des objets.
- Incapacité à détecter des objets non présents dans la base d'entraînement.

3.6.3 Perspectives d'amélioration

- Enrichir la base de données d'entraînement avec plus d'exemples, notamment pour les catégories les plus difficiles.
- Expérimenter de nouvelles architectures de réseaux de neurones plus performantes.
- Implémenter des techniques d'augmentation des données et de transfer Learning pour améliorer la généralisation.
- Ajouter des mécanismes de fusion multi-capteurs (caméra, lidar, radar) pour une détection plus robuste.
- Optimiser l'implémentation logicielle et matérielle pour réduire le temps de traitement.

3.7 Discussion

Dans ce chapitre de notre mémoire intitulé "Conception et Réalisation d'un Système de Reconnaissance d'Objets avec Estimation de Distance pour les Personnes Malvoyantes", nous avons exploré en détail l'accès au matériel utilisé et les étapes de réalisation de notre projet novateur. En mettant en avant les capacités avancées de la carte Raspberry Pi 4, nous avons pu développer un système puissant et polyvalent pour répondre aux besoins spécifiques des personnes malvoyantes. Grâce à nos efforts de conception et de réalisation, nous avons créé un prototype fonctionnel de notre système, combinant les fonctionnalités avancées de la carte Raspberry Pi 4 avec une conception logicielle adaptée. Ce chapitre revêt une importance cruciale dans notre mémoire, car il offre une compréhension approfondie de l'architecture et du fonctionnement du système développé, ouvrant ainsi la voie à une technologie accessible et innovante pour améliorer l'autonomie et la qualité de vie des personnes malvoyantes .

Conclusion générale

Les avancées technologiques dans les domaines du traitement d'image et de la détection d'objets offrent de nombreuses possibilités pour améliorer l'accessibilité des informations visuelles pour les personnes malvoyantes ou aveugles. Les techniques d'apprentissage automatique, en particulier le deep learning, permettent de développer des solutions innovantes telles que la reconnaissance d'objets, la détection de texte et la description vocale de l'environnement visuel.

Ces technologies d'assistance visuelle permettent une meilleure reconnaissance des objets, Elles offrent aux personnes malvoyantes un accès crucial à des informations visuelles essentielles, contribuant à une meilleure compréhension de leur environnement et à une plus grande indépendance dans leurs déplacements et leurs activités.

Bien que des défis persistent en termes de performances, de coût, d'ergonomie et d'adaptation aux besoins spécifiques des utilisateurs, les perspectives d'évolution de ces technologies sont très prometteuses. Des efforts de recherche et de développement continus, en collaboration étroite avec les communautés concernées, permettront sans aucun doute de faire progresser ces solutions et de les rendre accessibles au plus grand nombre de personnes malvoyantes et aveugles.

En conclusion, le développement de systèmes de détection d'objets avec estimation de distance, intégrant des techniques d'apprentissage automatique avancées, représente une avancée majeure pour l'inclusion et l'amélioration de la qualité de vie des personnes malvoyantes. Ces technologies d'assistance visuelle facilitent l'accès à l'information et stimulent l'autonomie de cette population, ouvrant la voie à une société plus inclusive et accessible à tous.

BIBLIOGRAPHIE

- [1] Braikia Siham, « □ Thésés-Algérie: Doctorat, Magister, Master... »
- [2] D. Ameisen, « Qu'est-ce qu'une image numérique ? », 2013.
- [3] S. Bres, J.-M. Jolion, et F. Lebourgeois, *Traitement et analyse des images numériques*. Paris: Hermès science publications, 2003.
- [4] I. G. N. Made Kris Raya, A. N. Jati, et R. E. Saputra, « Analysis realization of Viola-Jones method for face detection on CCTV camera based on embedded system », in *2017 International Conference on Robotics, Biomimetics, and Intelligent Computational Systems (Robionetics)*, Bali: IEEE, août 2017, p. 1-5. doi: 10.1109/ROBIONETICS.2017.8203427.
- [5] A. Yahiaoui, « Système de Discrimination Visages / Faux Visages par Réseaux de Neurones Convolutifs (CNN) », université de guelma, Working Paper, 2021. Consulté le: 27 juin 2024. [En ligne]. Disponible sur: <http://dspace.univ-guelma.dz/jspui/handle/123456789/11761>
- [6] « Projet - Tutoré Final | Télécharger gratuitement PDF | Internet des objets | Tarte aux framboises ». Consulté le: 27 juin 2024. [En ligne]. Disponible sur: <https://fr.scribd.com/document/695251514/projet-tutore-final>
- [7] B. Afaf, B. Chahrazed, et D. C. Ammar, « Person Re-identification Based CNN and Score Normalization ».
- [8] R. A. Z. Berkani Mohamed, Achir, « (PDF) Implémentation sur FPGA d'un système intelligent pour la surveillance de l'état de santé des patients souffrants d'arythmie cardiaque ». Consulté le: 27 juin 2024. [En ligne]. Disponible sur: https://www.researchgate.net/publication/355792542_Implementation_sur_FPGA_d'un_syste_me_intelligent_pour_la_surveillance_de_l'etat_de_sante_des_patients_souffrants_d'arythmie_cardiaque
- [9] « (PDF) Approches d'apprentissage automatique pour sécuriser l'Internet des objets ». Consulté le: 27 juin 2024. [En ligne]. Disponible sur: https://www.researchgate.net/publication/341850594_Machine_Learning_Approaches_for_Securing_Internet_of_Things
- [10] M. Himeur, « Palmprint Recognition Using Deep Learning », Université 8Mai 1945 – Guelma, Working Paper, juill. 2021. Consulté le: 27 juin 2024. [En ligne]. Disponible sur: <http://dspace.univ-guelma.dz/jspui/handle/123456789/11537>
- [11] « Introduction à l'apprentissage profond (deep learning) de l'intelligence artificielle ».
- [12] A. Krizhevsky, I. Sutskever, et G. E. Hinton, « ImageNet classification with deep convolutional neural networks », *Commun. ACM*, vol. 60, n° 6, p. 84-90, mai 2017, doi: 10.1145/3065386.

Bibliographie

- [13] A. Géron, *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*, First edition. Beijing ; Boston: O'Reilly Media, 2017.
- [14] O. A. Aliouane, F. Ghernaouti, A. (E.) Chouchane, et M. R. A. (Co-E.) Berkani, « Etude d'un système intelligent pour la classification des cancers en utilisant le séquençage des ARN », 2023, Consulté le: 27 juin 2024. [En ligne]. Disponible sur: <https://dspace.univ-medea.dz/handle/123456789/11071>
- [15] W. Benkamouch, « Détection d'objets et deep Learning dans un trafic routier », university of guelma, Working Paper, 2023. Consulté le: 27 juin 2024. [En ligne]. Disponible sur: <http://dspace.univ-guelma.dz/jspui/handle/123456789/14924>
- [16] T. Merenda, F. Depasse, et S. Patris, « Déficience visuelle : développement et validation de recommandations sur les aides techniques applicables dans la pratique officinale et détermination de la population cible », *Le Pharmacien Clinicien*, vol. 59, n° 2, p. e118-e119, juin 2024, doi: 10.1016/j.phacli.2024.04.197.
- [17] C. M. Bishop et H. Bishop, *Deep Learning: Foundations and Concepts*. Cham: Springer International Publishing, 2024. doi: 10.1007/978-3-031-45468-4.
- [18] N. Ketkar, *Deep Learning with Python*. Berkeley, CA: Apress, 2017. doi: 10.1007/978-1-4842-2766-4.
- [19] J. Brownlee, « Deep Learning for Computer Vision ».
- [20] Y. LeCun, Y. Bengio, et G. Hinton, « Deep learning », *Nature*, vol. 521, n° 7553, p. 436-444, mai 2015, doi: 10.1038/nature14539.
- [21] R. Yamashita, M. Nishio, R. K. G. Do, et K. Togashi, « Convolutional neural networks: an overview and application in radiology », *Insights Imaging*, vol. 9, n° 4, p. 611-629, août 2018, doi: 10.1007/s13244-018-0639-9.
- [22] R. Jafri, S. A. Ali, H. R. Arabnia, et S. Fatima, « Computer vision-based object recognition for the visually impaired in an indoors environment: a survey », *The Visual Computer*, vol. 30, n° 11, p. 1197-1222, nov. 2014, doi: 10.1007/s00371-013-0886-1.
- [23] M. Djemaa, « Classification des images par CNN », Université 08 Mai 1945 de Guelma, Working Paper, juin 2023. Consulté le: 27 juin 2024. [En ligne]. Disponible sur: <http://dspace.univ-guelma.dz/jspui/handle/123456789/14697>
- [24] J. Yaméogo, « Causes de déficience visuelle de l'enfant à Ouagadougou ».
- [25] A. Brillhault, « Vision artificielle pour les non-voyants: une approche bio-inspirée pour la reconnaissance de formes ».
- [26] S. Ansari, *Building Computer Vision Applications Using Artificial Neural Networks: With Examples in OpenCV and TensorFlow with Python*. Berkeley, CA: Apress, 2023. doi: 10.1007/978-1-4842-9866-4.
- [27] *Scratch et Raspberry Pi: s'initier à l'électronique et à la robotique par le jeu*. Saint-Herblain: Éditions ENI, 2019.
- [29] W. Puech, « Analyse et TraiteVmisieonnpatr Odrdeinasteuirmages ».
- [30] K. Dahmane, « Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées ».
- [31] K. Dahmane, « Analyse d'images par méthode de Deep Learning appliquée au contexte routier en conditions météorologiques dégradées ».
- [32] C. Shorten et T. M. Khoshgoftaar, « A survey on Image Data Augmentation for Deep Learning », *J Big Data*, vol. 6, n° 1, p. 60, déc. 2019, doi: 10.1186/s40537-019-0197-0.
- [33] R. P. Ltd, « Buy a Raspberry Pi 4 Model B », Raspberry Pi. Consulté le: 27 juin 2024. [En ligne]. Disponible sur: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>