

الجمهورية الجزائرية الديمقراطية الشعبية
République algérienne démocratique et populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique
جامعة عين تموشنت بلحاج بوشعيب
Université –Ain Temouchent- Belhadj Bouchaib
Faculté des SCIENCES et de TECHNOLOGIE
Département des Mathématiques et de l'Informatique



Projet de Fin d'Etudes
Pour l'obtention du Diplôme de Master en Informatique
Domaine : Mathématiques et Informatique
Filière : Informatique
Spécialité : Réseaux et Ingénierie des Données (RID)

Thème

Etude comparative sur les algorithmes de détection et de reconnaissance de véhicules dans une séquence d'images

Présenté Par :

- 1) Melle. BENZAZOU Aïcha
- 2) Melle. SAHRAOUI Yasmine Tekfa

Devant le jury composé de :

Dr. Mohammed Hakim BENDIABDALLAH M C A UAT.B.B (Ain Temouchent) Président
Mme. Fatima Zahra BERRAKEM M A A UAT.B.B (Ain Temouchent) Examinatrice
Mme. Samira SAIDI M A A UAT.B.B (Ain Temouchent) Encadrante

Année Universitaire 2021/2022



Remerciements



*Avant tout nous tenons à remercier **ALLAH** le tout puissant de nous avoir donné le courage, la patience, la santé et la volenté pour mener à bien ce modeste travail, qu'il soit béni et glorifié.*

*Nous exprimons nos profonds et sincères remerciements à notre encadrante **Mme SAIDI Samira**, pour avoir cru en nos capacités et de vouloir bien nous encadrer, pour le temps qu'elle nous a consacré, pour ses remarques et conseils ainsi que pour sa patience, sa compréhension et d'avoir éclairé nos chemins aussi d'avoir été là pour nous soutenir.*

*Également nous remercions les membres de **Jury** d'avoir accepté d'évaluer ce modeste travail.*

Nos vifs remerciements ainsi que nos gratitudes à nos très chers parents, qui nous ont accompagné et soutenu tout au long de notre cursus.

Enfin, sans oublier tous nos amis qui nous ont aidés et encouragés, même par leurs pensées, qui nous aiment et que nous aimons.

A tous on vous dit encore merci, c'est grâce à vous tous qu'on a pu réaliser ce mémoire.

Table des matières

TABLE DES MATIÈRES	I
LISTE DES FIGURES	III
LISTE DES TABLEAUX	IV
LISTE DES ABRÉVIATIONS	V
INTRODUCTION GÉNÉRALE	1
CHAPITRE I : DÉTECTION ET RECONNAISSANCE DE VÉHICULES DANS UNE SÉQUENCE D'IMAGES	3
I.1. INTRODUCTION.....	4
I.2. LE TRANSPORT INTELLIGENT.....	4
I.2.1. Définition	4
I.2.2. Objectifs des STI	5
I.2.3. Politique des STI en Algérie.....	5
I.3. LA DÉTECTION, CLASSIFICATION ET RECONNAISSANCE DES VÉHICULES.....	6
I.3.1. La détection des véhicules.....	6
I.3.2. La classification des véhicules.....	6
I.4. LA RECONNAISSANCE DE VÉHICULE DANS UNE VIDÉO.....	7
I.5. ETAT DE L'ART.....	8
I.5.1. Les méthodes de détection et classification des véhicules dans une vidéo.....	8
I.5.1.1. La détection de véhicules en mouvement.....	9
1. Les détections basées sur la différence inter-images.....	10
2. Les détections basées sur la modélisation du fond.....	10
3. Les détections basées sur la cohérence.....	11
I.5.1.2. Détection par les caractéristiques d'apparence.....	12
1. Extraction de caractéristiques.....	12
2. Apprentissage profond.....	13
I.5.2. Les algorithmes de détection et de reconnaissance de véhicules dans une vidéo.....	15
I.5.2.1. L'augmentation des données et RCNN plus rapide améliorent la détection et la reconnaissance des véhicules.....	15
I.5.2.2. Détection de véhicule et classification de type basée sur CNN-SVM.....	16
I.5.2.3. Une approche de détection de véhicule robuste basée sur un algorithme R-CNN plus rapide.....	17
I.5.2.4. Détection de véhicules à l'aide de différents algorithmes d'apprentissage profond à partir d'une séquence d'images.....	17
I.5.2.5. Détection de véhicules à l'aide de modèles d'apprentissage profond Alexnet et R-CNN plus rapide : Une étude comparative.....	18
I.6. CONCLUSION.....	19
CHAPITRE II : RÉSEAUX DE NEURONES ET RÉSEAUX DE NEURONES CONVOLUTIFS (CNN) 20	
II.1. INTRODUCTION.....	21
II.2. INTRODUCTION À L'INTELLIGENCE ARTIFICIELLE.....	21
II.2.1. <i>Qu'est-ce que l'intelligence artificielle.....</i>	21
II.2.2. <i>Brève histoire sur l'intelligence artificielle.....</i>	22
II.2.2.1. Canard de Vaucanson 1738.....	22
II.2.2.2. Le test de TURING 1950.....	23
II.2.2.3. La méthode de résolution 1965.....	23
II.2.2.4. Les années 1970.....	23
II.2.2.5. Les années 1980 et 1990.....	24
II.2.3. <i>Domaines d'application de l'IA.....</i>	24
II.3. APPRENTISSAGE PROFOND.....	25
II.3.1. <i>Apprentissage automatique (Machine Learning).....</i>	25
II.3.1.1. Apprentissage supervisé.....	26
II.3.1.2. Apprentissage non supervisé.....	26
II.3.1.3. Apprentissage par renforcement.....	27

II.3.2.	<i>Apprentissage automatique vs apprentissage profond</i>	27
II.4.	RÉSEAUX DE NEURONES ARTIFICIELS (RNA)	28
II.4.1.	<i>Qu'est-ce qu'un réseau de neurones artificiel</i>	28
II.4.2.	<i>Les couches des réseaux de neurones artificiels</i>	28
II.4.3.	<i>Les types des réseaux de neurones artificiels</i>	29
II.5.	RÉSEAUX DE NEURONES CONVOLUTIFS (CNN)	31
II.5.1.	<i>Architecture des CNN</i>	31
II.5.1.1.	Les couches de convolution (Convolution Layer CONV).....	31
II.5.1.2.	Les couches de Pooling	31
II.5.1.3.	Les couches de connexion complète (Full Connected Layer).....	32
II.5.1.4.	La fonction d'activation	32
II.5.2.	<i>Exemples d'architectures CNN</i>	33
II.5.2.1.	LeNet-5 (1998).....	33
II.5.2.2.	AlexNet (2012).....	33
II.5.2.3.	GoogleNet (2014).....	33
II.5.2.4.	VGGNet (2014).....	33
II.5.2.5.	ResNet (2015)	34
II.6.	CONCLUSION	34
CHAPITRE III : CONCEPTION ET IMPLÉMENTATION		35
III.1.	INTRODUCTION.....	36
III.2.	CONCEPTION	36
III.2.1.	<i>Choix des algorithmes</i>	36
III.2.1.1.	YOLOv5.....	36
III.2.1.2.	Faster R-CNN.....	36
III.2.2.	<i>Architecture YOLO</i>	37
III.2.3.	<i>Architecture R-CNN</i>	38
III.2.4.	<i>Architecture Fast R-CNN</i>	39
III.2.5.	<i>Architecture Faster R-CNN</i>	39
III.2.5.1.	ResNet-50.....	41
III.2.6.	<i>Mask R-CNN</i>	42
III.2.7.	<i>Architecture de détection de véhicule avec YOLOv5</i>	43
III.2.7.1.	Base d'entraînement :	43
III.2.7.2.	Paramètres utilisés	43
III.2.7.3.	Résultats sur des images fixes	44
III.2.8.	<i>Détection de véhicule avec Faster R-CNN</i>	44
III.2.8.1.	Base d'apprentissage	45
III.2.9.	<i>Evaluation des performances</i>	45
III.2.9.1.	Matrice de confusion	45
III.3.	IMPLÉMENTATION	46
III.3.1.	<i>Environnement et outils de développement</i>	46
III.3.1.1.	Google Colab.....	46
III.3.1.2.	Mode d'exécution GPU	47
III.3.1.3.	Langage Python.....	47
III.3.1.4.	Bibliothèques utilisées.....	47
III.3.2.	<i>Résultats d'entraînement</i>	48
III.4.	CONCLUSION	57
CONCLUSION GÉNÉRALE		58
BIBLIOGRAPHIE.....		59
RÉSUMÉ		

Liste des figures

Figure 1: Schéma de la classification d'un véhicule dans une image avec les réseaux de neurones. [Touahri, 2020]	7
Figure 2: Le schéma général des méthodes de reconnaissance de véhicules. [Rerbal et Foukia,2017]	8
Figure 3: Illustration du Background et Foreground (Autoroute Est-Ouest Algérie). [Fardjallah, 2014]	9
Figure 4: Liste des méthodes de détection et de classification des véhicules. [Harianto et al., 2021].....	9
Figure 5: La différence inter-image. [A. Bugeau, 2011].....	10
Figure 6: Soustraction du fond. [A. Bugeau, 2011]	10
Figure 7 : Illustration d'une opération de soustraction d'arrière-plan. [Fardjallah, 2014].....	11
Figure 8: (a) Image originale, (b) Régions en mouvement détectées. [Fardjallah, 2014].....	12
Figure 9: Etapes d'extraction des caractéristiques. [Chabot, 2015].....	13
Figure 10: Les principales catégories de détecteurs d'objets. [Atif et Misseraoui, 2020].....	14
Figure 11: Le Canard Digérateur 1738. [Benbrahim, 2020]	23
Figure 12: IA vs ML vs DP. [Medium, 2020].....	27
Figure 13: Organisation en couches d'un réseau de neurones artificiel. [Wikipédia]	29
Figure 14: Perceptron monocouche i. [Bisimwa, 2011].....	30
Figure 15: Perceptron multicouche. [Bisimwa, 2011]	30
Figure 16: Architecture standard d'un réseau à convolutions. [Fabien, 2018].....	31
Figure 17: (A) Max pooling avec un filtre 2*2 et un pas de 2, (B) Moy pooling.....	32
Figure 18: Allure de la fonction ReLU. [Santi, 2019]	32
Figure 19: Architecture de YOLO. [Redmon et al, 2016]	37
Figure 20: Principe de l'algorithme YOLO. [Redmon et al, 2016].....	37
Figure 21: Architecture R-CNN. [Girshick et al., 2014].....	38
Figure 22: L'architecture Fast R-CNN. [GIRSHICK, 2015]	39
Figure 23: Architecture de FasterR-CNN. [Redmon et al, 2016]	40
Figure 24: L'architecture du ResNet-50. [Aditi Rastogui, 2022]	41
Figure 25: L'architecture du Mask RCNN. [KAZI TANI L., 2020]	42
Figure 26: Quelques images de jeu de données vehicules-OpenImages.....	43
Figure 27: Résultats du modèle YOLOv5.....	44
Figure 28: Interface de Google Colab.	46
Figure 29: Paramètres du notebook sur Google Colab.	47
Figure 30: Graphes de métriques (YOLOv5).....	49
Figure 31: Matrice de confusion (YOLOv5).....	50
Figure 32: Estimateurs (metrics) (YOLOv5).	51
Figure 33: La fonction loss (YOLO v5).....	51
Figure 34: Détection de véhicules par YOLOv5 (1).	53
Figure 35: Détection véhicules par Faster R-CNN (1).....	53
Figure 36: Détection de véhicules par YOLO (2).	54
Figure 37: Détection de véhicules par Faster R-CNN (2).....	54
Figure 38: Détection de véhicules par YOLO (3).	55
Figure 39: Détection de véhicules par Faster R-CNN (3).....	55
Figure 40: Détection de véhicules par YOLO (4).	56
Figure 41: Détection de véhicules par Faster R-CNN (4).....	56

Liste des tableaux

Tableau 1: Résultats du test comparatif des trois méthodes à l'époque 10.	16
Tableau 2: Tableau récapitulatif des algorithmes étudiés pour la détection de véhicules.	19
Tableau 3: Comparaison de R-CNN, Fast R-CNN et Faster R-CNN.	40
Tableau 4: Matrice de confusion pour une classification binaire.....	45
Tableau 5: Rapport de classification.	48
Tableau 6: Comparaison entre les deux méthodes (séquence 1).....	53
Tableau 7: Comparaison entre les deux méthodes (séquence 2).....	54
Tableau 8: Comparaison entre les deux méthodes (séquence 3).....	55
Tableau 9: Comparaison entre les deux méthodes (séquence 4).....	56

Liste des abréviations

ANN : Artificiel Neural Network (réseau neuronal artificiel)

CNN : Convolutional Neural Network (réseaux de neurones convolutifs).

CPU : Central Processing Unit (unité centrale de traitement).

DL : Deep Learnig.

Faster R-CNN : Faster Region-CNN.

GPU : Graphics Processing Unit.

IA : Intelligence Artificielle.

ML : Machine Learning.

ResNet : RESidual NETwork (réseaux résiduels).

RNA : Réseaux de Neurones Artificiels.

STI : Systèmes de Transport Intelligents.

SVM : Support Vector Machine (machine à vecteurs de support).

TIC : Technologies de l'Information et de la Communication (Information and Communications Technologies).

YOLO : You Only Look Once.

Introduction Générale

L'intelligence artificielle a connu récemment de grandes avancées notamment grâce à l'apprentissage profond et la vision par ordinateur. Il est désormais possible de reconnaître un objet dans une image ou une séquence d'images ce qui est très pratique pour les véhicules autonomes et pour les systèmes de surveillance. Avec l'apprentissage profond, le développement de l'IA passe à la vitesse supérieure.

De nos jours, les vidéosurveillances constituent l'une des solutions de sécurité les plus anciennes et répandues. Afin de gérer le trafic routier, de nombreuses études sont menées sur ces systèmes de surveillance utilisant des techniques de traitement d'images.

La détection et la reconnaissance d'un véhicule à partir d'une séquence d'image est la première étape de la mise en œuvre d'un système avancé et sophistiqué. Elle est l'une des applications des systèmes de transport intelligent pour laquelle une solution fiable et efficace est toujours recherchée.

Notre projet entre dans le cadre de surveillance routière dont l'objectif principale est la détection et la reconnaissance des véhicules sur des vidéos du trafic routier ou autoroutier. A ce propos nous avons soulevé des questionnements qui nous ont interpellé et ont suscité notre intérêt face à la conjoncture actuelle en Algérie.

Nous présenterons une revue détaillée des techniques de détection et de classification des véhicules et nous discuterons également des différentes approches utilisées dans ce contexte.

Dans ce mémoire, nous nous interrogerons sur les nouvelles stratégies de l'intelligence artificielle et plus particulièrement l'apprentissage profond comment peut-il être utilisé pour la détection et la classification des véhicules (voiture, bus, moto, ...) dans une séquence vidéo. Pour cela, nous nous sommes basées sur deux outils disponibles proposés récemment exploitant les réseaux de neurones de convolution pour la détection d'objets : YOLO et Faster R-CNN. Nous avons choisi ces deux algorithmes, car YOLO permet d'obtenir les meilleurs résultats, il est adapté pour les systèmes de temps réel et Faster R-CNN est une des méthodes CNN les plus utilisées à ce jour. Nous utilisons pour l'implémentation le langage de programmation Python.

Cette étude que nous avons fait va nous permettre de subvenir aux besoins de détection et de classification des véhicules perçus dans une vidéo, ce qui va simplifier par la suite la question du trafic et l'identification des automobiles ou des véhicules suspects.

Le but de ce travail est de détecter et de reconnaître des véhicules dans une séquence d'images en se basant sur l'apprentissage profond. Puis faire une étude comparative sur des algorithmes de détection d'objets.

Notre mémoire s'articule autour de trois chapitres tel que :

- **Chapitre 1** : fait l'objet d'étude sur les systèmes de transport intelligent, ensuite les méthodes de détection et reconnaissance de véhicules suivi d'un état de l'Art sur quelques études récentes sur la détection et la classification des véhicules dans une vidéo.
- **Chapitre 2** : est consacré à une étude sur l'intelligence artificielle suivi de quelques généralités sur les réseaux de neurones artificiels (RNA), les réseaux de neurones convolutifs (CNN) et le deep learning.
- **Chapitre 3** : est réservé à la conception, l'environnement et les outils de l'implémentation. Également nous décrivons les architectures utilisées pour mettre en œuvre le processus de détection et de classification des véhicules ainsi que les résultats obtenus et leurs interprétations.

En conclusion, nous synthétisons notre mémoire par les résultats obtenus et proposons des perspectives pour les travaux futurs qui seront élaborées par d'autres chercheurs.

*Chapitre I :
Détection et
reconnaissance de
véhicules dans une
séquence d'images*

I.1. Introduction

La détection de véhicules est une technique qui utilise des photographies pour identifier et étiqueter des véhicules.

Le problème de détection de véhicule est considéré comme un problème de segmentation consistant à séparer ou classer les pixels en 2 classes distinctes, l'arrière-plan et l'avant-plan.

Dans ce chapitre, nous allons tout d'abord donner un aperçu sur le système de transport intelligent, puis nous nous intéressons plus précisément à la détection et à la reconnaissance de véhicules dans une séquence d'images. Nous allons également passer en revue les approches et les méthodes les plus utilisées dans ce contexte.

I.2. Le transport intelligent

I.2.1. Définition

La ville intelligente désigne une ville utilisant les technologies de l'information et de la communication (TIC) pour améliorer la qualité des services urbains ou encore réduire ses coûts. Dans ces villes intelligentes on trouve les « systèmes de transport intelligents » (STI) qui sont des applications ou des services avancés associés à l'ingénierie des transports, les technologies de la communication, de l'information et du positionnement géographique. [Benahmed et Benyamina, 2017]

On appelle ces transports « intelligents » parce qu'ils sont liés à des capacités associées à l'intelligence humaine : mémoire, communication, traitement de l'information, capacités sensorielles et comportement adaptatif. On trouve les STI dans plusieurs champs d'activité : optimisation de l'utilisation des infrastructures de transport, amélioration de la sécurité et de la sûreté, développement des services ainsi que développement durable en favorisant le report de la route vers les autres modes plus respectueux de l'environnement. [Logistics, 2018]

On peut distinguer 4 grandes périodes dans le développement des STI :

- Années 1960-1970 : les prémices ;
- Années 1980-1995 : investissement dans l'information routière embarquée ;
- Années 1995-2000 : interopérabilité, billettique et autoroute automatisée ;
- Années 2000-2005 : mobilité durable, multi modalité et sécurité routière.

Aujourd'hui, les chercheurs essaient d'envisager le développement des véhicules du futur dans une logique globale, portant aussi bien sur la technologie que sur la conception des équipements et leur intégration dans les véhicules, afin de repenser la mobilité dans une perspective d'intermodalité faisant une place plus importante aux véhicules électriques. [M. Gérard VOISIN, 2009]

1.2.2. Objectifs des STI

L'objectif des STI est de résoudre les problèmes de sécurité et d'encombrement du trafic dans les systèmes de transport. Les systèmes de communication hertziens et filaires, qui permettent l'échange de différents types d'information entre les systèmes de transport, les systèmes de commande et les utilisateurs, jouent un rôle fondamental dans la réalisation de cet objectif. [José M. Costa, 2006]

Les STI permettent également :

- L'intégration de la chaîne de transport et de la logistique (suivi des véhicules, des marchandises) du client jusque dans les dépôts logistiques ;
- La prise en compte de la multimodalité et de l'intermodalité en assurant l'interopérabilité ;
- Le respect des réglementations imposant un suivi particulier (transport de matières dangereuses, produits pharmaceutiques, etc.). [Logistics, 2018]

Les technologies STI sont regroupées dans un ensemble de services utilisateur interdépendants visant à résoudre les problèmes liés au transport. [José M. Costa, 2006]

1.2.3. Politique des STI en Algérie

En raison de la crise du trafic routier et d'éclairage public en l'Algérie les pouvoirs publics ont décidé d'installer des systèmes de contrôle de la circulation des 25 carrefours au niveau de la wilaya d'Alger. En partenariat avec la société mixte algéro-espagnole doté d'une enveloppe financière de 19 milliards DA, ce projet dont la réalisation en 3 phases, est prévu sur 55 mois permettra de connaître le trafic routier en temps réel, d'améliorer les conditions de déplacement, réduire la durée des voyages et tenir informé les usagers de la route via Internet. [Trafic routier Alger, 2019]

I.3. La détection, classification et reconnaissance des véhicules

La vidéo est composée d'une série d'images fixes et animées qui changent à un taux de plus de 24 images par seconde. En conséquence, la vidéo bénéficie d'une image statique unique : les images continues sont interconnectées. En d'autres termes, une vidéo inclut les caractéristiques de mouvement du véhicule. [Harianto A et al., 2021]

Dans le domaine de la vision par ordinateur, la détection consiste à percevoir une scène statique ou dynamique, et de savoir le changement de mouvement approprié en déduisant ainsi la cinématique de tous les objets mobiles par rapport à la caméra. [Abidi et Slimani, 2016]

I.3.1. La détection des véhicules

La détection d'objets aide à localiser des instances d'objets dans des images ou des vidéos. Les algorithmes de détection d'objets exploitent généralement l'apprentissage automatique ou l'apprentissage profond pour produire des résultats significatifs. Lorsque les humains regardent des images ou des vidéos, nous pouvons reconnaître et localiser des objets d'intérêt en quelques instants.

La détection de véhicules est une technique qui utilise des photographies pour identifier et étiqueter des véhicules. Cette méthode est en plein essor dans les milieux universitaires et industriels depuis de nombreuses années. Jusqu'à présent, la plupart des méthodes de détection d'objets les plus récentes n'ont pas été en mesure de répondre aux critères de référence en matière de détection de véhicules. Les grandes variations de lumière, les occlusions denses et les grandes variations d'échelle des objets sont les principaux défis de la détection de véhicules. [Harianto A et al., 2021]

I.3.2. La classification des véhicules

La classification d'objets consiste à identifier ce que représente chacune des régions segmentées. En imagerie aérienne par exemple, on cherche à déterminer les zones urbaines, les forêts, zones d'eau, les montagnes, Il s'agit de regrouper les différents éléments (pixels) en thèmes correspondants à la vérité du terrain. Le résultat est bien entendu une image segmentée.

La classification des véhicules peut désormais être utilisée, entre autres, pour la programmation de l'entretien des routes, la surveillance du trafic et les alertes en cas d'infraction au code de la route. Comme les photos sont identiques à celles d'une voiture, l'identification peut être difficile. [Harianto A et al., 2021]

Les photos de véhicules peuvent être prises dans un certain nombre de conditions, notamment le point de vue, l'éclairage, etc. Diverses photos de l'identité « personnelle » du véhicule, telles que sa forme, sa face avant, son arrière, etc. peuvent être incluses. [Harianto A et al., 2021]

Pour différencier les classifications des véhicules, une amélioration de l'identification des véhicules peut être utilisée. Pour traiter les images prises afin de pouvoir visualiser le véhicule correctement, un système d'augmentation des données est utilisé. La correction des images et l'entraînement des images sur leur état actuel peuvent conduire à une meilleure précision. [Harianto A et al., 2021]

Récemment, le réseau neuronal convolutif est largement utilisé pour la classification de véhicules.

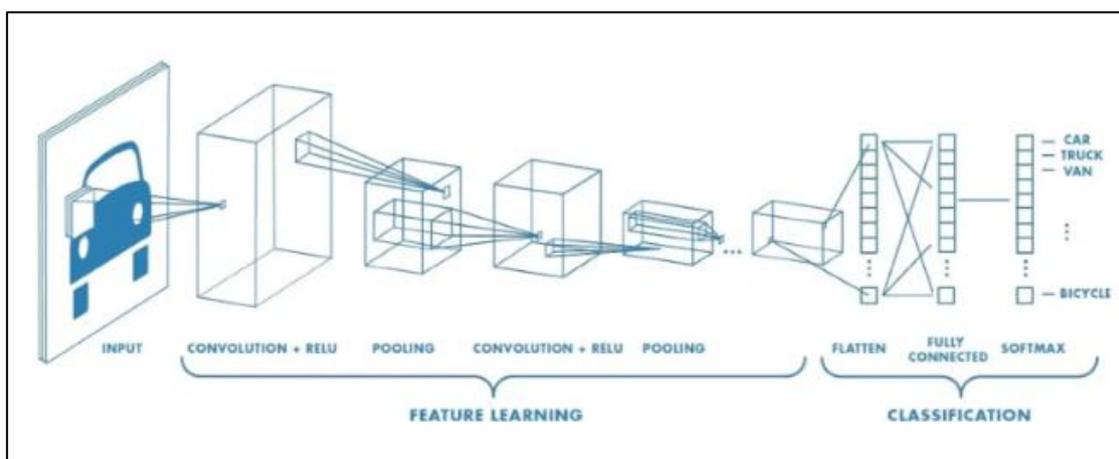


Figure 1: Schéma de la classification d'un véhicule dans une image avec les réseaux de neurones. [Touahri, 2020]

I.4. La reconnaissance de véhicule dans une vidéo

La reconnaissance des véhicules est un problème dans lequel un ordinateur doit reconnaître un véhicule de la même manière qu'une personne. L'identification des véhicules est généralement utilisée par le gouvernement à des fins telles que les contraventions électroniques et l'entretien des routes. [Harianto A et al., 2021]

L'image d'un véhicule peut inclure des détails tels que le type, la classe, etc. Un véhicule de la même classe peut avoir la même forme, mais sa forme, ou catégorie, peut être différente. En tant qu'êtres humains, nous pouvons instinctivement savoir quel type de véhicules nous regardons, et nous espérons qu'avec l'avènement de la technologie de reconnaissance des véhicules, les ordinateurs seront capables de faire de même. [Harianto A et al., 2021]

La reconnaissance de véhicules peut être améliorée en améliorant les photographies des véhicules la *figure 2* représente le schéma des méthodes de reconnaissance des véhicules.

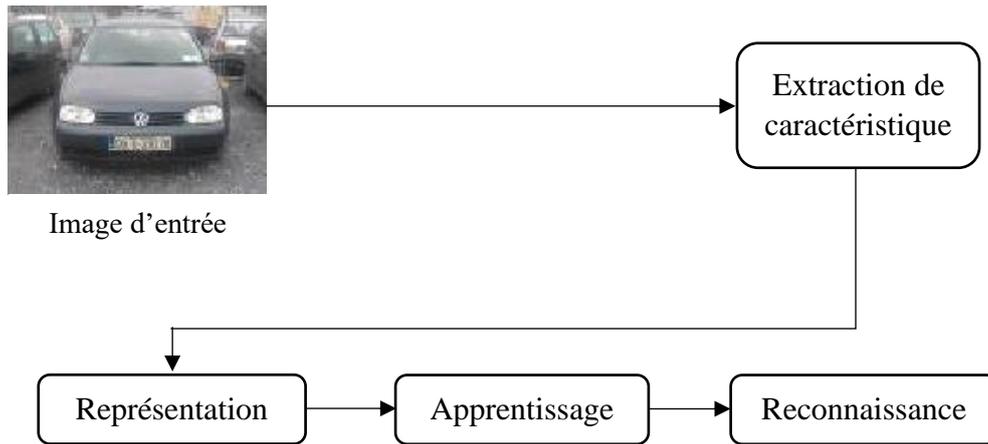


Figure 2: Le schéma général des méthodes de reconnaissance de véhicules. [Rerbal et Foukia,2017]

- **Extraction** : l'extraction d'un ensemble de descripteurs à partir de l'image, qui reflètent les informations pertinentes pour la reconnaissance du véhicule ;
- **Représentation** : la construction d'un modèle du véhicule pour représenter une catégorie ;
- **Apprentissage** : l'apprentissage des paramètres du modèle à partir des images d'apprentissage étiquetées ;
- **Reconnaissance** : l'appariement avec le modèle ou la classification en utilisant le modèle pour reconnaître le véhicule cible dans une image de test. [Rerbal et Foukia, 2017]

I.5. Etat de l'art

I.5.1. Les méthodes de détection et classification des véhicules dans une vidéo

La détection de véhicule basée sur la vidéo consiste à séparer les véhicules de l'arrière-plan (fond de l'image). Pour cela cette technologie se divise en deux sur deux catégories : **celles basées sur les caractéristiques du mouvement du véhicule** et **celles basées sur les caractéristiques statiques**. Le schéma dans la (*Figure 4*) résume ces deux catégories.

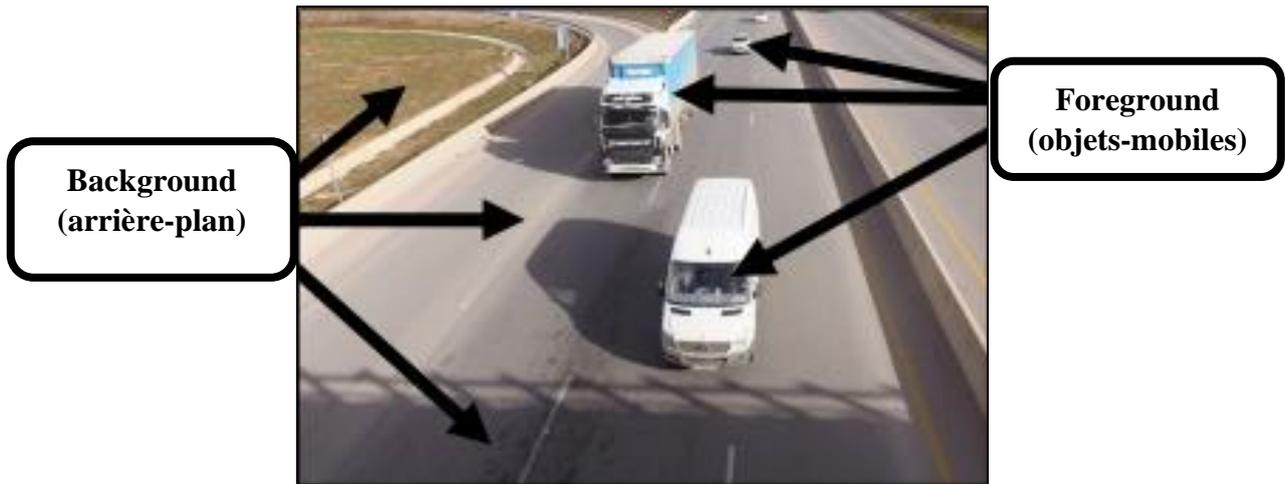


Figure 3: Illustration du Background et Foreground (Autoroute Est-Ouest Algérie). [Fardjallah, 2014]

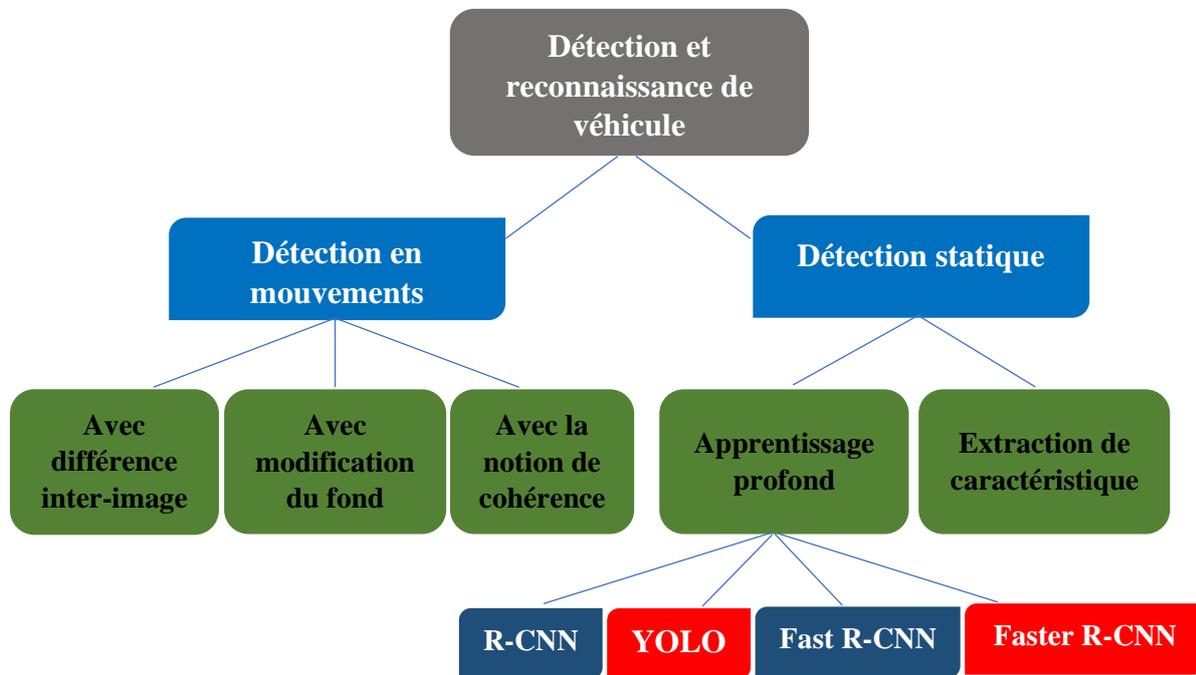


Figure 4: Liste des méthodes de détection et de classification des véhicules. [Harianto et al., 2021]

1.5.1.1. La détection de véhicules en mouvement

Parmi les méthodes de détection de véhicule en mouvement on peut citer :

- Les détections basées sur la différence inter-images.
- Les détections basées sur la modélisation du fond.
- Les détections utilisant la notion de cohérence

1. Les détections basées sur la différence inter-images

La méthode de détection de mouvement la plus simple consiste à faire la soustraction entre deux images successives. Elle porte aussi le nom de méthode de différence temporelle dans d'autres littératures. Les pixels dont l'intensité résultante est proche de zéro sont assimilés comme étant les pixels du fond. Les méthodes basées sur la différence inter-images s'adaptent très bien aux environnements dynamiques mais laisse des "trous" dans les véhicules en mouvements.

Par exemple, la différence temporelle souillée permet de détecter des véhicules en mouvement dans une séquence vidéo. Une amélioration de cette méthode consiste à extraire trois images de différence au lieu de deux permettant de décider si un pixel a bougé. [Abidi et Slimani, 2016]

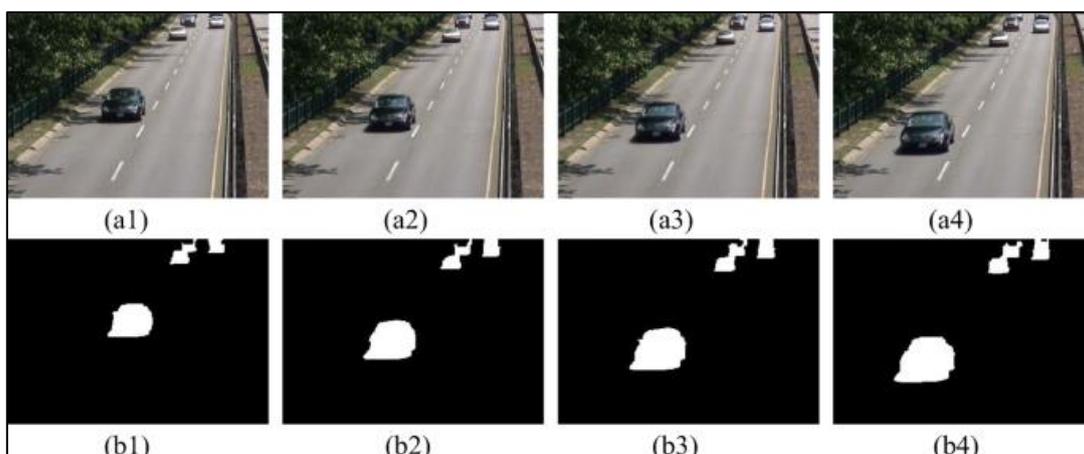


Figure 5: La différence inter-image. [A. Bugeau, 2011]

2. Les détections basées sur la modélisation du fond

Tout d'abord, la caméra doit être maintenue fixe et les occultations de parties du fond par des objets en mouvement doivent rester temporellement minoritaires. De plus, il est préférable d'avoir, au début de la séquence, un certain nombre d'images sans objets en mouvement afin de pouvoir apprendre correctement le fond. La modélisation du fond, tout comme les méthodes basées sur la détection inter-images, permet de décider si un pixel est en mouvement ou non. [A.Bugeau, 2011]

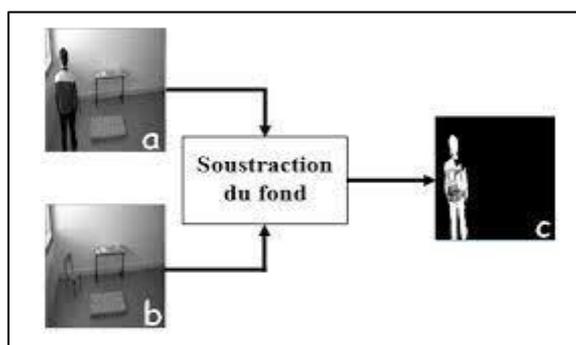


Figure 6: Soustraction du fond. [A. Bugeau, 2011]

L'inconvénient de cette approche est la nécessité de conserver en mémoire les images précédentes pour l'estimation. Une implémentation récursive a été proposée où la valeur médiane est incrémentée de 1 si le pixel est supérieur à sa valeur, et décrémente de 1 le cas échéant. L'inconvénient est une lente adaptation et demande par conséquent une longue période d'apprentissage. [Fardjallah et Boughazi, 2014]



Figure 7 : Illustration d'une opération de soustraction d'arrière-plan. [Fardjallah, 2014]

3. Les détections basées sur la cohérence

Une dernière approche consiste à définir un objet mobile comme une région ayant un mouvement cohérent. Dans ce contexte, un mouvement cohérent peut être défini comme un mouvement ayant de grande chance de provenir d'une "cible" classique (personne, véhicule). Cette définition a été utilisée pour la segmentation de mouvement en ajoutant l'hypothèse suivante : un objet avec un mouvement cohérent se déplace dans une direction approximativement constante pendant une courte période (en pratique quelques images).

Les modèles de mouvement dérivés du flot optique sont utilisés comme primitives pour la détection ou le suivi. Le flot optique a pour rôle de décrire le mouvement cohérent des points entre des images successives, des mesures de flot optique cohérentes en direction sont accumulées pendant quelques pas de temps. Une estimation du déplacement de chaque pixel à travers une séquence d'images est ainsi obtenue. Il est alors possible de distinguer les objets mobiles, qui se déplacent avec une direction constante, des mouvements parasites. Une région cohérente peut être directement vue comme une couche. [Abidi et Slimani, 2016]

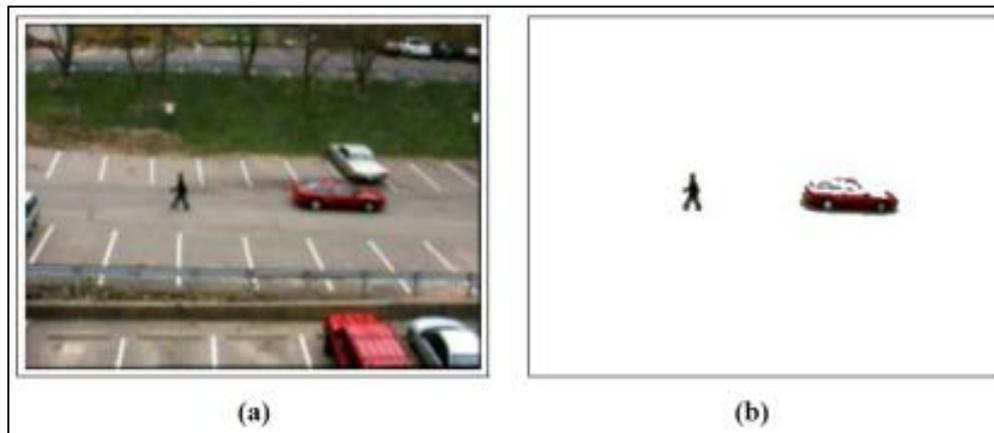


Figure 8: (a) Image originale, (b) Régions en mouvement détectées. [Fardjallah, 2014]

Bien que l'approche basée sur le mouvement soit rapide, elle est fortement gênée par les changements fréquents de contexte, bruit vidéo ou d'autres facteurs, ce qui entraîne des trous dans les véhicules et un bruit excessif. [Harianto et al., 2021]

1.5.1.2. Détection par les caractéristiques d'apparence

1. Extraction de caractéristiques

L'utilisation directe de l'image pour la détection et la reconnaissance n'est pas envisageable parce qu'elle contient une grande quantité d'information impertinente ou redondante. C'est pourquoi, la première étape de presque toutes les méthodes de reconnaissance est d'extraire un ensemble de caractéristique qui représentent les propriétés discriminantes de l'objet cible. [Chabot et al., 2015]

L'extraction des caractéristiques du véhicule dans les méthodes utilisées actuellement était généralement basée sur une ou plusieurs caractéristiques du véhicule, telles que le contour, la texture, le bord, la couleur ou des parties spécifiques du véhicule, telles que le pare-brise, les feux, les plaques d'immatriculation, etc. Cette approche est simple mais inefficace car une bonne extraction de caractéristiques à partir d'une définition simple est souvent difficile à réaliser. [Harianto et al., 2021]

Cette approche se fait sur l'ensemble de l'image, ce qui permet de calculer la suite des convolutions en une seule fois. A la fin de cette étape, une carte de caractéristiques est générée. C'est sur celle-ci que les patches à classifier sont extraits. [Chabot et al., 2015]

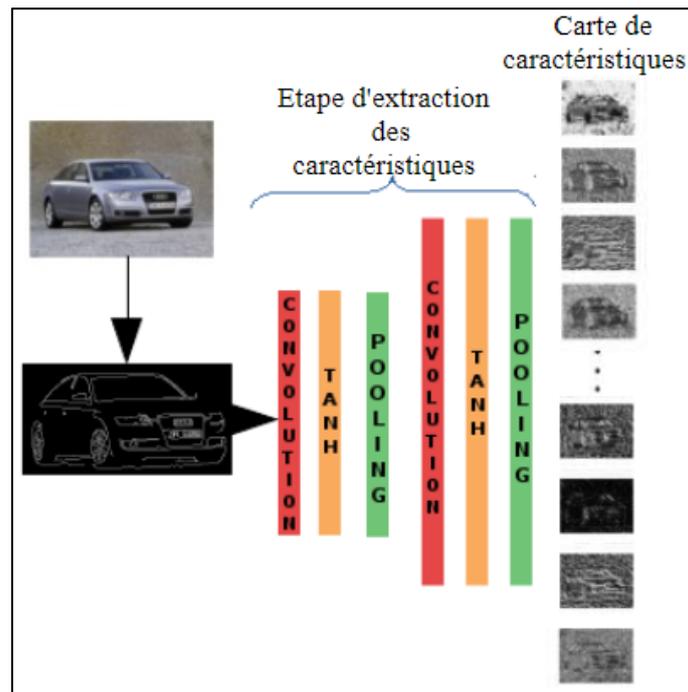


Figure 9: Etapes d'extraction des caractéristiques. [Chabot, 2015]

Récemment beaucoup de méthodes ont été proposé pour cette approche, mais les plus utilisées en raison de leur fiabilité et de la pertinence des informations porté sont : les caractéristiques ‘Pseudo-Haar’, ‘Haar-Like’, ‘Local Binary Pattern « LBP »’, ‘Transformé de Hough’, ‘Principal Component Analysis « PCA »’, ‘Histogram of oriented gradients « HOG »’. [Touahri, 2019]

2. Apprentissage profond

Plusieurs techniques d'apprentissage automatique pour la détection et la reconnaissance d'images ont été développées, telles que les réseaux de neurones et l'apprentissage profond. Ces dernière années, l'apprentissage profond a fait des progrès rapides dans le domaine de la détection de cibles, démontrant un niveau élevé d'extraction de caractéristiques et améliorant considérablement la détection. L'une des mises en œuvre les plus courantes est le réseau de neurones convolutifs (CNN). [Harianto et al., 2021]

Actuellement, les cadres de détection d'objets basés sur l'apprentissage profond peuvent être principalement divisés en deux types, notamment les méthodes basées sur la proposition de régions (**détecteurs à deux étages**) et les méthodes basées sur la régression (**détecteurs à un étage**) (figure 10). [Atif et Misseraoui, 2020]

- Détecteurs à deux étages :

Les détecteurs à deux étages ont donné une précision plus élevée avec de meilleures performances et rapportent des résultats idéaux mieux que les détecteurs à un étage dans la détection d'objet, mais ils sont généralement plus lents que les détecteurs à un car ils ont deux étages : d'abord l'extraction de proposition de régions, puis la classification de chaque proposition et la prédiction de la boîte englobante.

Même en limitant le nombre de régions à traiter, les améliorations de performance ne sont pas suffisantes pour faciliter le fonctionnement en temps réel. [Atif et Misseraoui, 2020]

- Détecteur à un étage :

Les détecteurs à un étage sont beaucoup plus rapides et plus recherchés pour les applications de détection d'objets en temps réel, mais ont des performances relativement médiocres par rapport aux détecteurs à deux étages. Ils peuvent réduire les calculs en supprimant l'étape de proposition de région et en formulant la détection d'objet comme un problème de régression dense. [Atif et Misseraoui, 2020]

Au lieu d'avoir deux réseaux pour deux tâches différentes (proposition et classification), un seul ConvNet est utilisé pour les deux tâches simultanément. Un seul réseau neuronal prédit les boîtes englobantes et les probabilités de classe directement à partir d'images complètes en une seule évaluation. [Atif et Misseraoui, 2020]

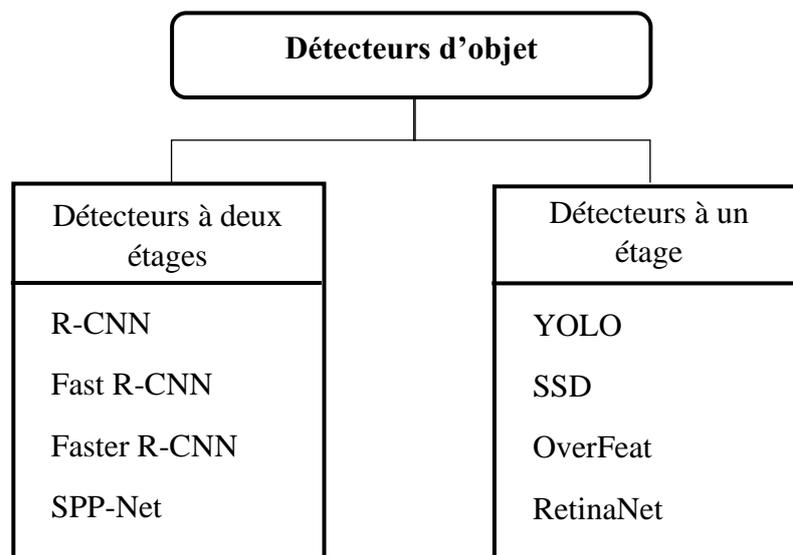


Figure 10: Les principales catégories de détecteurs d'objets. [Atif et Misseraoui, 2020]

Parmi les réseaux de neurones qui ont pu réaliser de la détection en temps réel avec des performances comparables à celles des autres réseaux, YOLO été le premier dans son époque.

La méthode de détection des véhicules qu'on a choisi pour notre implémentation est fondée sur cette technique qui est l'apprentissage profond, et on va la détaillée dans le chapitre 3.

1.5.2. Les algorithmes de détection et de reconnaissance de véhicules dans une vidéo

Nôtre étude bibliographique s'est concentrée sur une problématique qui est la détection et la reconnaissance de véhicules dans une séquence d'images. Pour cela notre recherche s'est fondée sur quelques travaux existants.

1.5.2.1. L'augmentation des données et RCNN plus rapide améliorent la détection et la reconnaissance des véhicules

La détection et la reconnaissance des véhicules sont améliorées par l'augmentation des données et un RCNN plus rapide. Cet article aborde l'idée comment un ordinateur peut reconnaître un véhicule de la même manière qu'une personne le ferait. Comparées aux méthodes traditionnelles, les technologies actuelles de la détection et la reconnaissance des véhicules basées sur les vidéos sont divisées en deux catégories. La première catégorie qui est basée sur les caractéristiques de mouvement de véhicule consiste à utiliser la corrélation d'image pour séparer le véhicule de l'arrière-plan, tandis que la deuxième catégorie est basée sur les caractéristiques d'apparence son ; concept est l'apprentissage profond pour classer les véhicules, pour cela il utilise le réseau de neurones de convolution et l'AlexNet.

Les auteurs ont adopté une architecture entièrement convolutive, pour leur modèle avancé, qui se fragmente en quatre différentes étapes :

- L'augmentation des données ;
- L'extraction de caractéristiques ;
- La proposition de zone réseau : production des RoIP en prenant comme entrée les cartes de caractéristiques ;
- Prédiction finales.

Les résultats des recherches montrent que l'augmentation des données améliore le résultat de 18 à 23 % par rapport à la référence. D'autre part elle a été utilisée pour reproduire les images qui pourraient être produites pour améliorer la précision. [Harianto et al., 2021]

- **La base de données utilisée :**

Google Open Image Dataset v6 a été utilisé pour ce processus, pour cela chaque véhicule est classé en quatre types (voiture, bus, camion et moto). 1000 photos de chaque classe ont été pris de la base, et chaque classe a été divisée en 80 % de données d'entraînement et 20 % de données de recherche. De plus chaque image est de dimension (685×1024) pixels. Pour l'environnement de travail ils ont utilisé Google Collab.

1.5.2.2. Détection de véhicule et classification de type basée sur CNN-SVM

Les auteurs de cet article se concentrent sur la détection et la classification des véhicules à l'aide d'un AlexNet amélioré et modifié pour cela deux méthodes différentes ont été utilisées. Tout d'abord, pour la détection des véhicules ils ont utilisé l'algorithme d'apprentissage en profondeur YOLO qui consiste à extraire les cibles avec Yolov2-tiny, son principe est d'améliorer considérablement la précision tout en la rendant plus rapide. Ils utilisent ensuite le réseau neuronal CNN en le combinant avec SVM pour la classification des objets. Le CNN est utilisé pour l'extraction des caractéristiques et il est basé sur AlexNet.

Afin d'améliorer la mobilité du réseau dans la détection de véhicules les auteurs prévoient d'augmenter le nombre d'ensembles de données, cette méthode sera également adoptée pour résoudre le problème de mauvaise précision dans la classification des véhicules. La méthode améliorée d'AlexNet utilisée par les auteurs est plus rapide que l'AlexNet original. En ajoutant SVM, la précision de la classification s'améliore et peut s'affiner dans une certaine mesure également le rappel de chaque catégorie augmente, comme illustré dans le *tableau 1*. [S.Karungaru et al. 2021]

- **La base de données utilisée :**

L'ensemble de données utilisé dans cet article est basé sur *BIT-Vehicle*, cette base de données se compose de 9850 images de véhicules. Les images contenues dans cette base sont extraites de vidéo obtenues par un appareil photo Canon EOS 550D à 24 endroits différents à Pékin et Tianjin en Chine, ces vidéos sont enregistrées à 25 images par seconde avec une performance de 960×540 pixels.

Methods	Car	Bus	Van	Others
AlexNet	98.17%	91.24%	68.44%	49.35%
Improved AlexNet	98.37%	93.81%	76.07%	71.43%
Improved AlexNet + SVM	98.87%	95.02%	76.50%	74.03%

Tableau 1: Résultats du test comparatif des trois méthodes à l'époque 10.

1.5.2.3. Une approche de détection de véhicule robuste basée sur un algorithme R-CNN plus rapide

Cet article propose une méthode robuste pour la détection des véhicules qui offre des résultats acceptables en termes de temps d'exécution et temps de précision. Les auteurs ont pu transformer l'architecture principale d'un réseau résiduel ResNet-50 pré-entraîné en un réseau neuronal convolutionnel basé sur le Faster R-CNN.

Les principales contributions de ce travail sont les suivantes :

- Evoluer l'impact d'un réglage correct des paramètres de l'algorithme Faster R-CNN ;
- Tester le modèle généré sur des données en conditions réelles avec différentes conditions météorologiques, temps et résolution.

Les résultats expérimentaux montrent que le facteur de sensibilité du système est très élevé et que dans quelques millisecondes les véhicules peuvent être détectés, ce qui permet d'obtenir des résultats plus précis. [Tourani et al, 2019]

- La base de données utilisée

L'ensemble de données *Cars Dataset* est sélectionné dans ce travail qui contient 16 185 images de 196 classes de véhicules. La largeur des images était redimensionnée à 300 tout en conservant le rapport d'aspect principal. Le modèle final a été entraîné sur 8 144 images d'entraînement de l'ensemble de données et les 8 041 images restantes ont été utilisées pour le test. La méthode a été mise en œuvre dans MATLAB.

1.5.2.4. Détection de véhicules à l'aide de différents algorithmes d'apprentissage profond à partir d'une séquence d'images

Pour cette approche, les auteurs se sont basés sur les modèles YOLO-v3, YOLO-v3-spp et YOLO-v3-tiny pour la détection de véhicules en temps réel. Ils ont fait une comparaison de ces modèles YOLO sur deux vidéos obtenues de manières différentes une terrestre et l'autre aérienne. Bien que la faible précision obtenue dans la vidéo aérienne, la vidéo terrestre a donné une bonne précision de **80.49%**. Pour cela les auteurs constatent que leur étude présente des valeurs de précision élevées par rapport à d'autres. [Atik et al, 2020]

- La base donnée utilisées

Pour réaliser la détection des véhicules deux vidéos ont été utilisées, une obtenue à partir d'un drone avec 1280×720 de résolution et une autre avec une résolution terrestre de 1080×1920. Le modèle YOLO utilise la base de données COCO, cet ensemble de données comporte 80 classes distinctes, 80 000 images d'entraînement et 40 000 images de vérification. L'étude a été réalisée sur le service Google Colab.

I.5.2.5. Détection de véhicules à l'aide de modèles d'apprentissage profond Alexnet et R-CNN plus rapide : Une étude comparative

Cet article vise à faire une étude comparative de deux modèles d'apprentissage profond utilisés pour la détection et la classification des véhicules, AlexNet et Faster R-CNN.

Pour le modèle AlexNet, la détection des objets a été effectuée sur la base de la soustraction de l'arrière-plan, tandis que la classification a été faite par un classifieur entraîné avec des caractéristiques d'images obtenues à partir d'un CNN. Le deuxième modèle utilisé est Faster R-CNN, ce dernier est un cadre de détection d'objet basé sur des réseaux de neurones convolutifs. Ce modèle a été utilisé sur la base de VGG16 qui donne des meilleurs résultats de performances.

Les auteurs ont effectué plusieurs tests pour évaluer la qualité des détections, les taux d'échec et les temps employés pour effectuer la tâche de détection. Et les résultats obtenus montrent que Faster R-CNN surpasse le modèle AlexNet, non seulement dans le taux de détection mais aussi dans le temps d'analyse. [John W.Branch et al, 2017]

- La base de données utilisées

Pour cette recherche les auteurs utilisent une séquence vidéo d'environnement urbain. La séquence comporte 1812 images, ces images comprennent 36 voitures différentes, 7 motos et 1 bus. De plus, le modèle basé sur VGG16 est pré-entraîné avec le jeu de données *ImageNet*.

Nous pouvons résumer ces travaux existants sous le tableau suivant (**Tableau 2**) :

Article	Méthodes Utilisées	Base Utilisée	Taux de Précision	Perspectives ou Travaux Futures
Article 1 : Avril 2021	Faster R-CNN et VGG16	<i>Google open image v6</i>	79%	-L'augmentation des données ; -Combinaison avec une autre approche d'augmentation de donnée basée sur une classification différente.
Article 2 : Mars 2020	YOLOv2-tiny et AlexNet	<i>BIT-Vehicle</i>	94.45%	-Résoudre le problème de faible mobilité du réseau dans la détection de véhicules ; -Améliorer la précision de certaines catégories dans la classification des véhicules.
Article 3 : Mars 2019	Faster R-CNN	<i>Cars Dataset</i>	98%	-Augmentation de la vitesse de détection ; -Fournir une application de détection de véhicule basée sur YOLO ;
Article 4 : 2020	YOLOv3-supp	<i>COCO</i>	72.02%	
Article 5 : 2017	AlexNet et Faster R-CNN VGG16	<i>ImageNet</i>	97%	-Amélioration des résultats du modèle Faster R-CNN ; -Utiliser l'extraction de caractéristiques à base d'AlexNet ou VGG pour améliorer la classification.

Tableau 2: Tableau récapitulatif des algorithmes étudiés pour la détection de véhicules.

I.6. Conclusion

Dans ce chapitre, nous avons présenté les différentes techniques d'un système de détection et de reconnaissance de véhicules, en définissant quelques algorithmes de détection des véhicules tel que YOLO et Faster R-CNN que nous allons en discuter dans le dernier chapitre.

Nous avons vu aussi quelques travaux sur la détection de véhicules qui présentent des avantages et des inconvénients pour différentes situations. On peut tirer de là que les détecteurs à un étage sont rapides en détection d'objets.

Chapitre II :
Réseaux de neurones
et réseaux de
neurones convolutifs
(CNN)

II.1. Introduction

Depuis plus d'une dizaine d'année, l'intelligence artificielle vit une accélération dans son développement et son adoption. Aujourd'hui, cette technologie émergente est devenue applicable à tous les secteurs d'activité tels que le commerce, la médecine, l'énergie, l'industrie, la logistique, finance ou encore le transport.

Ce chapitre a pour but de donner une vision générale sur l'intelligence artificielle qui représente un domaine complexe. Également d'accroître les connaissances sur les réseaux de neurones artificiels, qui sont parmi les techniques récemment utilisées dans la reconnaissance et la classification des objets, ainsi que les réseaux de neurones convolutifs (CNN).

De plus, nous allons comprendre en quoi consiste l'apprentissage profond.

II.2. Introduction à l'intelligence artificielle

II.2.1. Qu'est-ce que l'intelligence artificielle

L'intelligence artificielle est un ensemble de techniques permettant de résoudre des problèmes complexes en s'inspirant de mécanismes cognitifs humains, agissant de manière rationnelle en fonction de faits, données et expériences, et capables d'atteindre de manière optimale un ou plusieurs objectifs donnés. [Ezratty, 2018]

Le but de l'intelligence artificielle est double. D'une part, l'IA s'attache à résoudre des problèmes qui relèvent d'activités humaines ou animales de nature variée : perception, planification, interprétation de données, diagnostic, prise de décision, compréhension du langage, conception. D'autre part, l'IA cherche à mieux comprendre et modéliser l'intelligence [Haton, 2013]. Elle se rapproche ainsi des sciences cognitives dont elle s'inspire par ailleurs pour la conception de modèles (mémoire, raisonnement, apprentissage).

L'intelligence artificielle est un mélange de mathématiques et d'informatique. Ce n'est pas de la magie, mais bien des calculs. Prenons l'exemple d'une voiture autonome, une intelligence artificielle est capable de conduire une voiture toute seule, à condition qu'un ingénieur programme les commandes utiles pour passer les vitesses, freiner, reculer et les règles du code de la route. [Ezratty, 2018]

Comme tout domaine scientifique complexe, l'IA n'a jamais été un terrain d'unanimité et cela risque de perdurer. A haut niveau, on peut la découper en trois grands domaines :

- **Le symbolisme** qui se focalise sur la pensée abstraite et la gestion des symboles, l'algorithmique et la logique. Le symbolisme modélise notamment les concepts sous la forme d'objets reliés entre eux par des prédicats logiques (appartient à, etc.). C'est dans cette catégorie que l'on peut ranger les systèmes experts et moteurs de règles qui les font fonctionner, et dans une certaine mesure, le web sémantique.
- **Le connexionnisme** qui se focalise sur la perception, dont la vision, la reconnaissance des formes et s'appuie notamment sur les réseaux neuronaux artificiels qui reproduisent à petite échelle et de manière approximative le fonctionnement générique du cerveau. C'est ici que l'on peut ranger le deep Learning utilisé dans la vision artificielle ou le traitement de la parole.
- **Le comportementalisme** qui s'intéresse aux pensées subjectives de la perception. C'est dans ce dernier domaine que l'on peut intégrer l'informatique affective qui étudie les moyens de reconnaître, exprimer, synthétiser et modéliser les émotions humaines. [Ezratty, 2018]

II.2.2. Brève histoire sur l'intelligence artificielle

L'IA a été définie de multiples manières et reste de ce fait un domaine dont il n'est pas facile de circonscrire précisément les contours. [Zouinar, 2020]

Les premières machines pensantes apparaissent dans les histoires de science-fiction dès les années 1920. Tandis que la véritable naissance de l'Intelligence Artificielle coïncide avec l'apparition des ordinateurs dans les années 1940-1950. [CEA, 2018]

Il faudra attendre les années 1970 pour que l'IA prenne véritablement son essor, avec l'émergence de nouveaux langages de programmation et logiciels tels que les systèmes experts. Aujourd'hui, ce sont les réseaux de neurones qui deviennent la norme et ce succès soulève des questions sur l'utilisation de l'IA, qui dépassent largement le cadre de la seule informatique. [Ezratty, 2018]

II.2.2.1. Canard de Vaucanson 1738

Jacques de Vaucanson célèbre par ses automates créé le Canard Digérateur en 1738. Ce canard artificiel de cuivre doré boit, mange, cancanne, barbote et digère comme un vrai canard, dont les ailes étaient représentées, os par os, d'un mécanisme identique à ceux des vrais oiseaux. Il était possible de programmer les mouvements de cet automate, grâce à des pignons placés sur un cylindre gravé, qui contrôlaient des baguettes traversant les pattes du canard.

Le mécanisme, placé dans l'imposant piédestal, visible par tous, dans le but de montrer la complexité du travail accompli. [Benbrahim, 2020]



Figure 11: Le Canard Digérateur 1738. [Benbrahim, 2020]

Toutefois, il fallait attendre les années 30 (1930) pour que certains mathématiciens fassent progresser la logique ; analyse des fondements mêmes des raisonnements mathématiques et les années 50 (1950) pour que les calculateurs rentrent dans le domaine civil. Ces deux outils rassemblés ont provoqué la naissance de l'IA, terme introduit en 1954 par John McCarthy.

[Benbrahim, 2020]

II.2.2.2. Le test de TURING 1950

En 1950, c'est au tour des scientifiques d'imaginer des machines pensantes. Le mathématicien britannique, Alan Turing, père de l'informatique moderne, publie un article intitulé « l'ordinateur et l'intelligence », dans lequel il décrit comment savoir si une machine s'approche d'une intelligence humaine. Il appelle cette méthode « le jeu de l'imitation », plus connu aujourd'hui sous le nom de test de Turing. Depuis, les recherches sur l'IA ont connu successivement des périodes d'essor et de gel. [CEA, 2018]

II.2.2.3. La méthode de résolution 1965

La Méthode de Résolution (Robinson), cette méthode a donné naissance au langage PROLOG, ainsi que l'apparition des premiers chatbots comme c'est le cas du programme ELIZA créé en 1966. Ces chatbots tenaient le coup pendant des conversations avec quelques échanges mais ne passaient pas le test de Turing. [Ezratty, 2018]

II.2.2.4. Les années 1970

La fin des années 60 marque une période de crise pour l'IA on parle d'hiver de l'IA, la recherche s'effondre suite à désillusion scientifique. Cette crise a duré une décennie, dès les années 1970 l'intelligence artificielle prend un nouveau tournant.

Avant : les méthodes sont des améliorations du combinatoire : on restreint l'énumération exhaustive à l'aide du bon sens, de fonctions d'évaluations et d'heuristiques.

Après : les chercheurs sont convaincus que les programmes doivent avoir une connaissance approfondie du domaine étudié.

Un programme d'IA doit avoir toute la connaissance nécessaire et ne l'utiliser qu'à bon escient. Ces idées ont conduit à la réalisation et à l'étude des systèmes experts ou systèmes à base de connaissances. [Pastre, 2000]

II.2.2.5. Les années 1980 et 1990

Le début des années 1980 vit la première renaissance avec les systèmes experts, mais cette vague n'a pas duré longtemps. La fin des années 80, l'IA subit une nouvelle crise avec l'émergence des nouveaux ordinateurs personnelles. [Ezratty, 2018]

Pendant les années 1990, l'IA réintégré des labos de recherche et le début du « big data » cela est dû grâce à l'augmentation importante des capacités de calcul des machines. D'autre part, les réseaux de neurones et le deep Learning ont connu un premier essor au milieu des années 1990. [Ezratty, 2018]

II.2.3. Domaines d'application de l'IA

Beaucoup de réalisations mettant à profit des techniques d'intelligence artificielle dépassent les facultés humaines ; L'intelligence artificielle est présente sous diverses formes et dans des domaines variés mais elle vise toujours à faciliter la vie des hommes, elle a un impact considérable sur de nombreux domaines, dont certains sont mentionnés ci-dessous :

[Zouinar, 2020]

- **Les jeux** : L'intelligence artificielle a très vite intégré les jeux vidéo. Évidemment, tout jeu est basé sur des programmes donnant l'illusion de réactions intelligentes par rapport aux actions qui s'y déroulent. La machine devrait alors être en mesure de penser à plusieurs étapes possibles basées sur la connaissance heuristique.
- **Les systèmes experts** : un logiciel capable de simuler le comportement d'un humain effectuant une tâche très précise. C'est un domaine où l'intelligence artificielle est incontestablement un succès, dû au caractère très précis de l'activité demandée à simuler.
- **Traitement du langage naturel** : Le langage naturel a d'abord été étudié dans le but de faire de la traduction automatique. Devant les difficultés, on s'est plutôt intéressé aux sous-problèmes de la compréhension et de la génération du langage naturel, domaines qui se rattachent aussi maintenant au dialogue et aux interfaces.
- **Les robots Intelligents** : La robotique n'est pas forcément de l'IA. Elle peut en être par l'aspect Reconnaissance des formes et Résolution de problèmes. Il peut être nécessaire de faire face à des situations imprévues. Quelques applications sont particulièrement utiles : travaux dangereux, réparations dans les centrales nucléaires ou dans l'espace

- **Autres :**
 - Interaction homme/machine
 - Médecine, Modélisation cognitive
 - Réseaux neuronaux et approche connexionnistes
 - Reconnaissance des formes et perception
 - Reconnaissance de la parole et des visages
 - Reconnaissance de l'écriture manuscrite... [Pastre, 2000]

II.3. Apprentissage profond

L'apprentissage profond « Deep Learning » est un ensemble de techniques d'apprentissage automatique qui a permis des avancées importantes en intelligence artificielle dans les dernières années. Dans l'apprentissage automatique, un programme analyse un ensemble de données afin de tirer des règles qui permettront de tirer des conclusions sur de nouvelles données. L'apprentissage profond est basé sur ce qui a été appelé, par analogie, des « réseaux de neurones artificiels », composés de milliers d'unités « neurones » qui effectuent chacune de petites opérations simples. Les résultats d'une première couche de « neurones » servent d'entrée aux calculs d'une deuxième couche et ainsi de suite. Les progrès de l'apprentissage profond ont été possibles notamment grâce à l'augmentation de la puissance des ordinateurs et au développement de grandes bases de données « big data ». [Boughaba et Boukhris, 2017]

II.3.1. Apprentissage automatique (Machine Learning)

L'apprentissage automatique (Machine Learning) est un domaine de recherche en informatique qui traite des méthodes d'identification et de mise en œuvre de systèmes et algorithmes par lesquels un ordinateur peut apprendre, ce domaine a souvent été associé à l'intelligence artificielle. On peut citer trois types d'algorithmes d'apprentissage : [Oulmi et Kaloune, 2018]

- Apprentissage supervisé.
- Apprentissage non supervisé.
- Apprentissage par renforcement.

II.3.1.1. Apprentissage supervisé

L'apprentissage supervisé consiste en la conception d'un modèle reliant des données d'apprentissage à un ensemble de valeurs de sortie. C'est-à-dire que les données d'entraînement qu'on fournit à l'algorithme comportent les solutions désirées, appelées étiquettes (en anglais, labels). Cette méthode permet donc à l'algorithme d'apprendre en comparant sa sortie réelle avec les sorties enseignées, afin de trouver les erreurs et modifier le modèle en conséquent. L'apprentissage supervisé confère au modèle la possibilité de prédire des valeurs d'étiquette sur des données non étiquetées supplémentaires. [Andriamiandrasoa, 2019]

Les algorithmes d'apprentissage supervisé sont les plus couramment utilisés, il y a deux types d'apprentissage supervisé :

- La classification : elle consiste à trouver le lien entre une variable d'entrée (X) et une variable de sortie discrète (Y), en suivant une loi multinomiale.
- La régression : elle consiste à prédire une valeur continue pour la variable de sortie.

Les algorithmes les plus célèbres utilisés dans cette approche sont les suivants :

- SVM (Machines à vecteurs de support) : est un apprentissage automatique très puissant et polyvalent modèle, capable d'effectuer la classification linéaire ou non linéaire, la régression, et même détection des valeurs aberrantes.
- Les arbres de décision : un arbre de décision sert à classer les futures observations, sachant qu'un corpus d'observations est déjà étiqueté.
- Les réseaux de neurones : c'est un modèle informatique qui partage certaines propriétés avec le cerveau humain, dans lequel de nombreuses unités simples travaillent en parallèle sans centralisation, ils permettent de trouver des patterns complexes dans les données, il se compose de valeurs d'entrées, poids, fonction de transfert et une valeur de sortie. [Bensiah, 2020]

II.3.1.2. Apprentissage non supervisé

Le ML non supervisé définit essentiellement toute méthode de ML qui tente d'apprendre la structure en l'absence d'une sortie identifiée [Çınar et al, 2020]. Les ordinateurs apprennent à identifier par eux-mêmes les similitudes et les distinctions entre les données, et à regrouper ceux qui partagent des caractéristiques communes sans aucune orientation humaine stricte et rigoureuse. [Mateyaunga, 2020]

L'apprentissage automatique non supervisé peut être comparé à un enfant qui apprend à identifier le type de fruit en observant le motif et la couleur, au lieu de mémoriser les noms avec l'aide d'une autre personne. [Fatoumata et Amor, 2021]

II.3.1.3. Apprentissage par renforcement

Il existe beaucoup de problèmes où les sorties désirées que l'apprentissage supervisé exige sont difficile à spécifier. On ne dispose souvent que d'une information qualitative permettant l'évaluation de la réponse calculée. [Dilmi, 2012]

L'apprentissage par renforcement utilise cette évaluation pour améliorer les performances du système. Cette forme d'apprentissage constitue une méthode d'apprentissage par essais et erreurs. [Mezaache, 2008]

II.3.2. Apprentissage automatique vs apprentissage profond

La majeure différence qu'on note entre ses 2 concepts provient de la manière dont les données sont présentées au système (modèle).

- Les algorithmes de machine learning (ML) nécessitent presque toujours des données structurées, alors que les réseaux d'apprentissage profond reposent sur des couches de réseaux de neurones artificiels (RNA).
- On voit aussi une différence au sein de l'architecture des modèles qui les composent, on note que les modèles type DL sont plus profond que les modèles type ML.
- Deep Learning n'utilise que les réseaux de neurones, alors que pour le ML les réseaux de neurones sont qu'une approche de conception des modèles parmi tant d'autres.

En considérant le fait que le DL est la prochaine étape de l'évolution du ML inculquant aux machines la manière de prendre leurs décisions de façon précise sans l'intervention de l'expert humain [Senani, 2019].

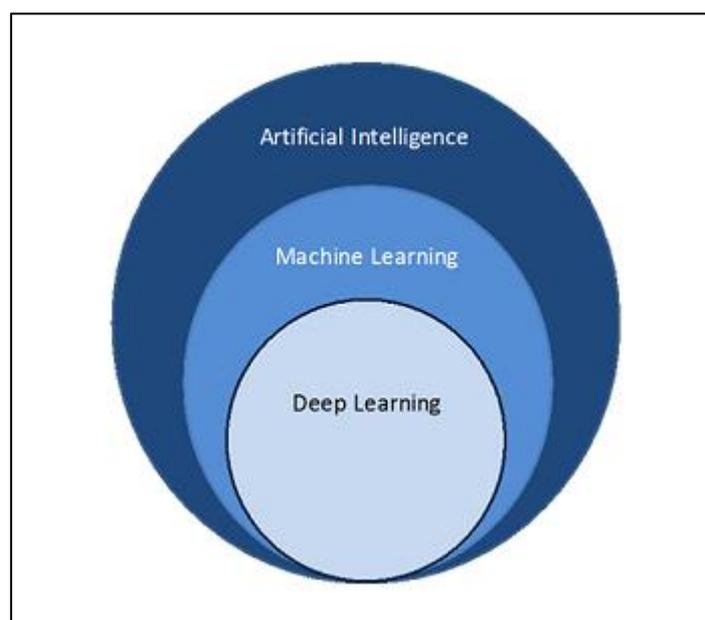


Figure 12: IA vs ML vs DP. [Medium, 2020]

II.4. Réseaux de neurones artificiels (RNA)

II.4.1. Qu'est-ce qu'un réseau de neurones artificiel

Un réseau de neurones artificiel est un modèle de calcul dont la conception est très schématiquement inspirée du fonctionnement de vrais neurones. Il est constitué d'un très grand nombre de petites unités de traitement identiques appelées neurones artificiels.

[KHODJA, 2011]

Il peut être caractérisé par :

- a. **La nature de ses entrées** ($x_1, x_2, \dots, x_i, \dots, x_n$), qui peuvent être les entrées du réseau ou les sorties d'autres neurones du même réseau.
- b. **La fonction d'entrée totale** définissant le prétraitement (combinaison linéaire) effectué sur les entrées comme $\sum W_i \cdot x_i + b$, dont W_i est le poids synaptique attaché à l'entrée i et le b désigne le seuil d'activation (biais).
- c. **Sa fonction d'activation**, ou d'état f , définissant l'état interne du neurone en fonction de son entrée totale. Cette fonction peut prendre plusieurs formes.
- d. **Sa fonction de sortie** calculant la sortie du neurone en fonction de son état d'activation.

[Bendaoud, 2014]

Le but est de concevoir des machines capables de non d'imiter, mais de se rapprocher au maximum du comportement de l'intelligence dont disposent les êtres humains, d'où la notion de "Intelligence Artificielle". [Gana, 2016]

II.4.2. Les couches des réseaux de neurones artificiels

Un RNA est généralement organisé en trois couches :

- **Une couche d'entrée** : Elle est constituée de l'ensemble des variables d'entrée.
- **Une ou plusieurs couches cachées** : Ce sont les couches qui se trouvent entre la couche d'entrée et la couche de sortie. Elles définissent l'activité interne du réseau. En général, les fonctions d'activations sont non linéaires au niveau de ces couches.
- **Une couche de sortie** : Elle est constituée de l'ensemble des neurones de sortie du réseau. C'est cette couche-là qui fournit les sorties principales. [Oukacine, 2012]

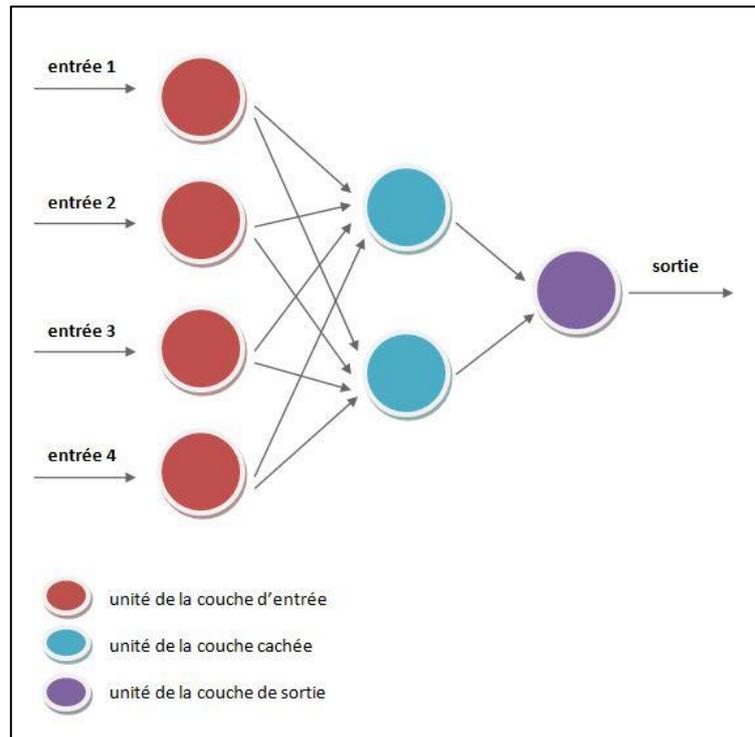


Figure 13: Organisation en couches d'un réseau de neurones artificiel. [Wikipédia]

Pour concevoir un réseau de neurones, quatre types de connexions entre les neurones doivent être établis : **directe**, **récurrente**, **latérale** et **à délais**.

Tous les réseaux de neurones utilisent la *connexion directe* pour acheminer l'information de l'entrée vers la sortie. La *connexion récurrente* permet d'acheminer l'information de la sortie des neurones des couches supérieures vers les entrées des neurones précédents. Les réseaux de neurones qui doivent choisir un neurone gagnant utilisent la *connexion latérale* pour établir une relation entre les neurones de sortie et la maintenir. Finalement, les problèmes temporels sont résolus par les modèles de réseaux dynamiques avec des *connexions à délais*. [Chaoui, 2002]

II.4.3. Les types des réseaux de neurones artificiels

Les réseaux de neurones peuvent se diviser en réseaux sans couches cachées et réseaux avec une ou plusieurs couches cachées.

- **Le perceptron monocouche** : est la forme la plus simple d'un réseau de neurones, il modélise la perception visuelle. L'apprentissage du perceptron peut se faire avec plusieurs méthodes déjà utilisées, il n'y a qu'une seule couche de poids modifiables entre les cellules d'association et les cellules de décision. Le perceptron est limité dans ses applications. Premièrement, il ne peut être applicable que dans la classification dont les variables sont linéairement séparables et deuxièmement la sortie ne peut être que 0 ou 1. Il ne dispose que de deux couches : une couche d'entrée et une couche de sortie.

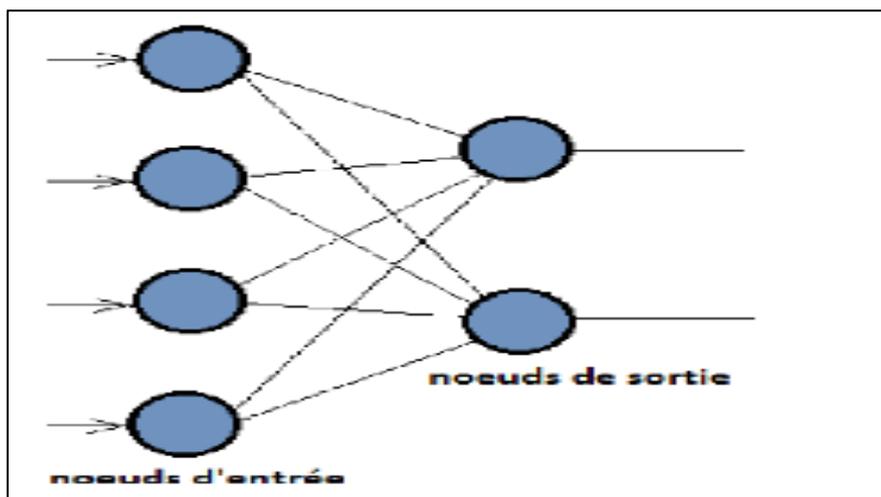


Figure 14: Perceptron monocouche *i*. [Bisimwa, 2011]

- **Le perceptron multicouche** : cette classe est la plus importante des réseaux de neurones car elle représente la généralisation du perceptron monocouche avec une fonction d'activation de type sigmoïde et une ou plusieurs couches cachées. Le vecteur d'entrée se propage dans le réseau de couche en couche jusqu'à la sortie, l'entraînement de celui-ci se fait avec l'algorithme par la rétropropagation de l'erreur. Ce réseau peut comporter une ou plusieurs couches cachées et un plus grand nombre de connexions permettant de résoudre la majorité des problèmes. [Chaoui, 2002]

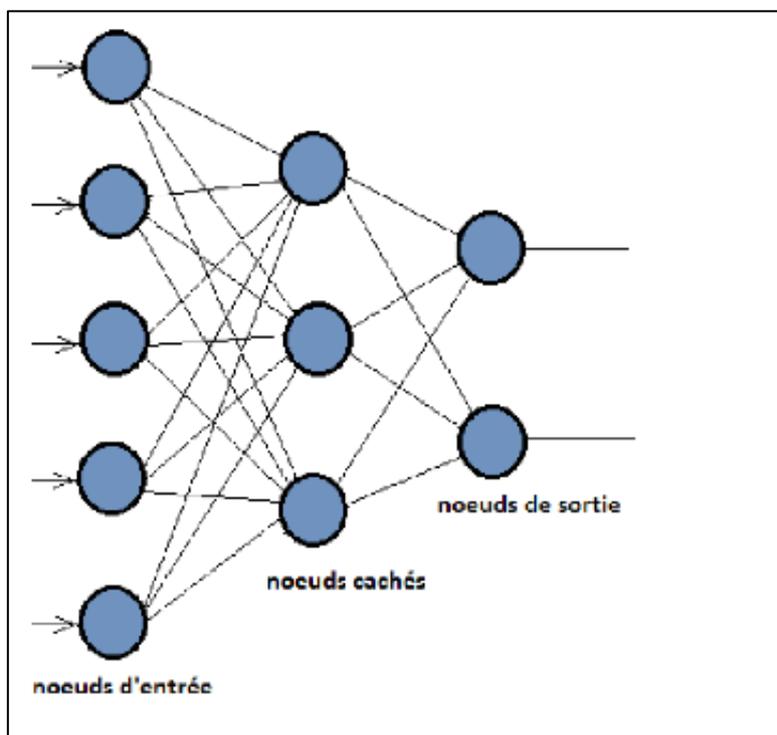


Figure 15: Perceptron multicouche. [Bisimwa, 2011]

II.5. Réseaux de neurones convolutifs (CNN)

II.5.1. Architecture des CNN

Comme tous les réseaux de neurones, les CNN sont composées de couches multiples, la première étant la couche d'entrée suivie d'un certain nombre de couches et puis par une couche de sortie qui donne le résultat de classification. [Bouhedadja, 2021]

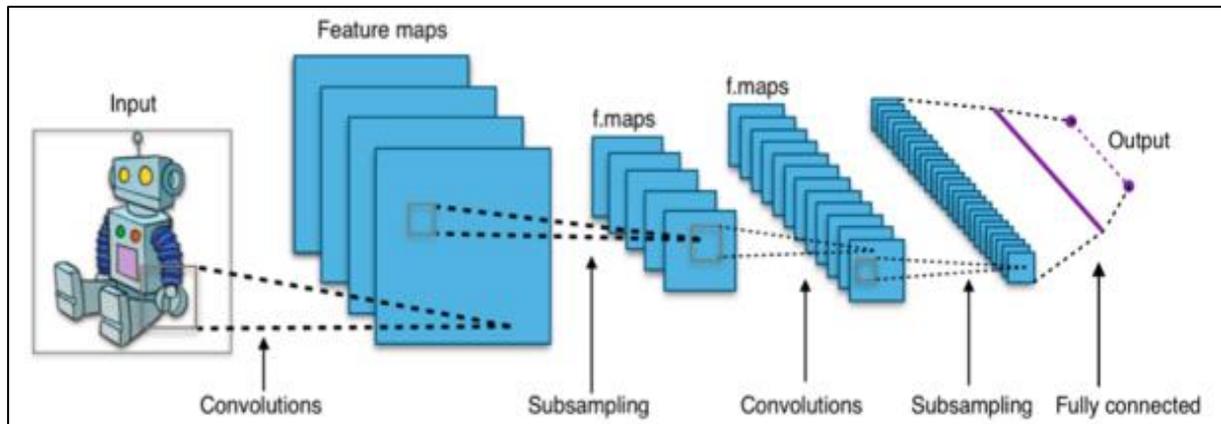


Figure 16: Architecture standard d'un réseau à convolutions. [Fabien, 2018]

II.5.1.1. Les couches de convolution (Convolution Layer CONV)

La couche de convolution a des noyaux (filtres) et chaque noyau a une largeur, une profondeur et une hauteur. Cette couche produit les cartes de caractéristiques à la suite du calcul du produit scalaire entre les noyaux et les régions locales de l'image.

Leur travail consiste à extraire les informations pertinentes de l'image (caractéristiques) grâce à une opération de convolution. Cette opération fait glisser un ensemble de filtres sur une image. Le poids de ces filtres est mis à jour lors de l'apprentissage et c'est grâce à eux que le réseau réussit à reconnaître les images. [Sekkil et Mebrouki, 2021]

II.5.1.2. Les couches de Pooling

Un autre outil très puissant utilisé par les CNNs s'appelle le pooling. Elle fait le sous-échantillonnage, ce qui permet de réduire la taille d'une image par extraction d'une valeur unique d'une région de valeurs, la valeur extraite dépend du type de regroupement utilisé, les types de regroupement les plus courants sont au maximum (max) pooling pour extraire la valeur la plus élevée du champ actuel, et pooling moyen (moy) qui extrait la valeur moyenne de la région actuelle.

Une couche de pooling a essentiellement des paramètres avec lesquels elle peut être configurée, d'abord il y a la stride qui est la distance sur laquelle le filtre est déplacé. Deuxièmement, il y a la taille du filtre f qui détermine la largeur et la hauteur du filtre utilisé pour extraire la valeur de l'entrée. [Bendjafer et Meddah, 2021]

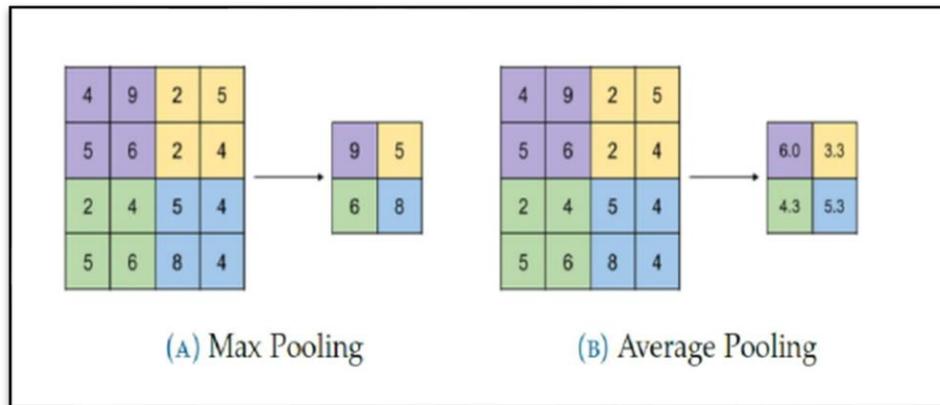


Figure 17: (A) Max pooling avec un filtre 2*2 et un pas de 2, (B) Moy pooling.

[Bendjafer et Meddah, 2021]

II.5.1.3. Les couches de connexion complète (Full Connected Layer)

La couche de connexion complète est une couche qui sert à effectuer la transformation sur la dimension de données afin que les données puissent être classées de manière linéaire. Chaque neurone de la couche de convolution doit d'abord être transformé pour que les informations contenues ne soient pas perdues. [Sellami, 2019]

II.5.1.4. La fonction d'activation

Permet d'améliorer l'efficacité du traitement en intercalant entre les couches de traitement une couche qui va opérer une fonction mathématique (fonction d'activation) sur les signaux de sortie. Elle remplace donc toutes les valeurs négatives reçues en entrées par des zéros. Elle joue le rôle de fonction d'activation.

La fonction ReLU (Rectified Linear Unit = Unités Rectifié linéaires) : $F(x) = \max(0, x)$. Cette fonction force les neurones à retourner des valeurs positives. La couche de correction ReLU remplace donc toutes les valeurs négatives reçues en entrées par des zéros, elle joue le rôle de fonction d'activation. [Santi, 2019]

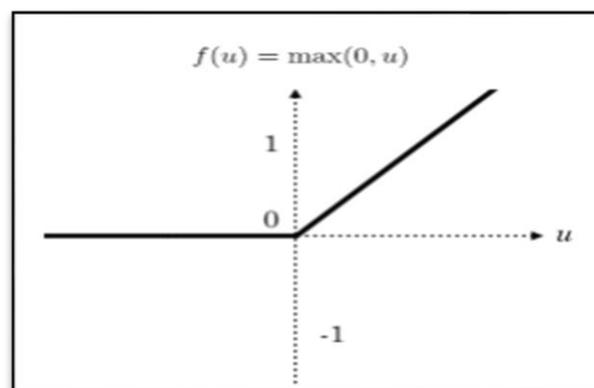


Figure 18: Allure de la fonction ReLU. [Santi, 2019]

II.5.2. Exemples d'architectures CNN

Il existe plusieurs architectures de réseaux de neurones convolutif, leur utilisation est relative au contexte. Ces architectures font souvent leurs preuves dans des défis d'apprentissage profond qui les rendent par la suite populaires et très utilisés. Voici notamment quelques-unes d'entre elles : [Bouhedadja, 2021]

II.5.2.1. LeNet-5 (1998)

Initié par Yann LeCun, première architecture CNN populaire, pour arriver à la version finale conçue et souvent utilisée pour la reconnaissance des caractères manuscrits (LeCun et al., 1998).

II.5.2.2. AlexNet (2012)

Le vainqueur du défi ImageNet ILSVRC2012 avec un résultat de 15.3% en top-5 d'erreurs, devançant le second au classement de 10%. Le réseau est développé par Alex Krizhevsky d'où le nom ainsi que les chercheurs Ilya Sutskever et Geoffrey Hinton (Krizhevsky, Sutskever, & Hinton, 2017). Assez similaire à LeNet-5 avec plus de profondeur et de largeur, cela est rendu possible notamment avec les avancées technologiques matérielles, en particulier, les processeurs graphiques et leur grand potentiel de calcul parallèle.

II.5.2.3. GoogleNet (2014)

Cette architecture a remporté la version 2014 du défi ImageNet ILSVRC, elle est développée par (Szegedy et al., 2015), du département de recherche chez Google. Le résultat obtenu diminue considérablement la marge d'erreur comparée aux résultats obtenus dans les précédentes versions du concours ainsi qu'avec son second concurrent. Ceci en utilisant des sous-réseaux nommés « module Inception » qui sous-entend le fait d'aller plus profondément dans le réseau.

II.5.2.4. VGGNet (2014)

Qui obtint la seconde place lors du ImageNet ILSVRC 2014, développé par (Simonyan & Zisserman, 2014), il a la particularité d'être simple avec deux ou trois couches de convolutions, puis une couche de pooling, s'ensuit une série similaire alternante de deux ou trois couches de convolutions et une de pooling jusqu'à avoir 16 couches de convolutions, et finir avec plusieurs couches complètement connectées successives.

II.5.2.5. ResNet (2015)

Rempporte le concours ILSVRC 2015. Optimisation du réseau par l'usage de Batch Normalization et l'apprentissage résiduel. Il est basé sur le saut des couches. Aussi il évite le problème du vanishing gradient et accélère la convergence du réseau. Couteux en mémoire et peu pratique sur les images de grandes dimensions. Utilise global AVG pooling au lieu du PMC à la fin.

II.6. Conclusion

Dans ce chapitre nous avons abordé certaines définitions concernant l'intelligence artificielle. Par la suite nous nous sommes concentrés sur les réseaux de neurones artificiels (RNA) et les réseaux de convolutions (CNN).

Aujourd'hui, l'apprentissage profond a prouvé sa grande efficacité dans plusieurs domaines, l'une des raisons de son succès est l'invention des CNNs.

Nous avons conclu de ce chapitre que le réseau de convolution est plus précis et plus rependu qu'un simple réseau de neurones pour la résolution des problèmes de classification car 'CNN' a une architecture complexe et de nombreuses couches multidimensionnelles qui font l'extraction de caractéristiques par contre 'Neural Network' fonctionne avec des vecteurs d'une dimension.

Dans le chapitre suivant, nous allons appliquer le deep learning pour reconnaître les véhicules et les détecter dans une séquence d'images en utilisant deux algorithmes différents (YOLOv5 et Faster R-CNN).

Chapitre III :
Conception et
implémentation

III.1. Introduction

Dans ce chapitre, nous allons présenter les différentes étapes de la conception et de l'implémentation pour détecter des différents véhicules (voiture, bus, motorcycle et camion) dans une séquence d'images avec un langage adéquat (dans notre cas Python), ainsi que les bibliothèques utilisées, les différents tests réalisés, les résultats obtenus, et une comparaison entre les deux algorithmes de l'apprentissage profond utilisés dans notre travail (YOLO V5 et Faster R-CNN).

III.2. Conception

III.2.1. *Choix des algorithmes*

III.2.1.1. *YOLOv5*

YOLO diminutif anglais de « *You Only Look Once* » est un algorithme de détection d'objets en temps réel dans des vidéos, des flux en direct ou des images, basé sur l'apprentissage en profondeur les plus populaires.

En tant que détecteur à un étage, YOLO utilise les caractéristiques de l'image entière pour prédire chaque boîte englobante. Il est très rapide au moment du test car il n'utilise qu'une seule architecture CNN pour prédire les résultats et la classe est définie de manière à traiter la classification comme un problème de régression, ce qui le rend beaucoup plus rapide que Faster R-CNN.

Le premier détecteur que nous avons testé pour la détection des véhicules est la version 5 de ce modèle. L'aspect limitant et désavantageux de l'algorithme YOLO est :

- Lutte pour détecter de petits objets ;
- Difficile de détecter les objets proches car chaque grille ne peut proposer que 2 boites englobante ;
- Plus d'erreurs de localisation par rapport à Faster R-CNN.

III.2.1.2. *Faster R-CNN*

Le second détecteur testé est l'algorithme Faster R-CNN diminutif de « *Région et CNN plus rapide* » il est parmi les meilleurs détecteurs à deux étages, il atteint la plus haute précision de détection d'objet mais avec une vitesse très lente par rapport à YOLO, cela signifie que le processus de génération des propositions de régions rend la vitesse bien inférieure aux exigences de la détection en temps réel.

- L'inconvénient majeur du Faster R-CNN était son coût de calcul élevé.

III.2.2. Architecture YOLO

L'architecture du YOLO est composée de 24 couches de convolution (*figure 19*) suivi de deux couches entièrement connectées (Fully-Connected Layers), la dimension des images d'entrée est de 448x448. [Khelalef, 2020]

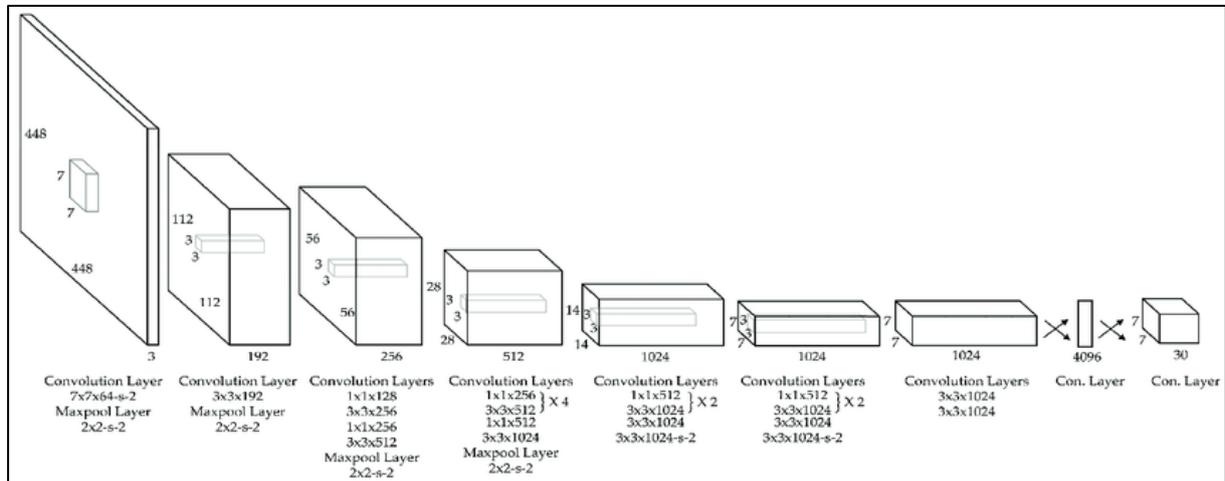


Figure 19: Architecture de YOLO. [Redmon et al, 2016]

Le principe du modèle est basé sur l'utilisation d'une succession de couches de convolution avec différentes tailles et nombre de filtres, cette architecture permet de diviser l'image en plusieurs zones qui englobent les différents objets ou portions d'objets dans l'image, l'application d'un seuillage des probabilités de chaque classe dans chaque zone permet de conserver uniquement les zones englobantes représentatives qui limitent les objets.

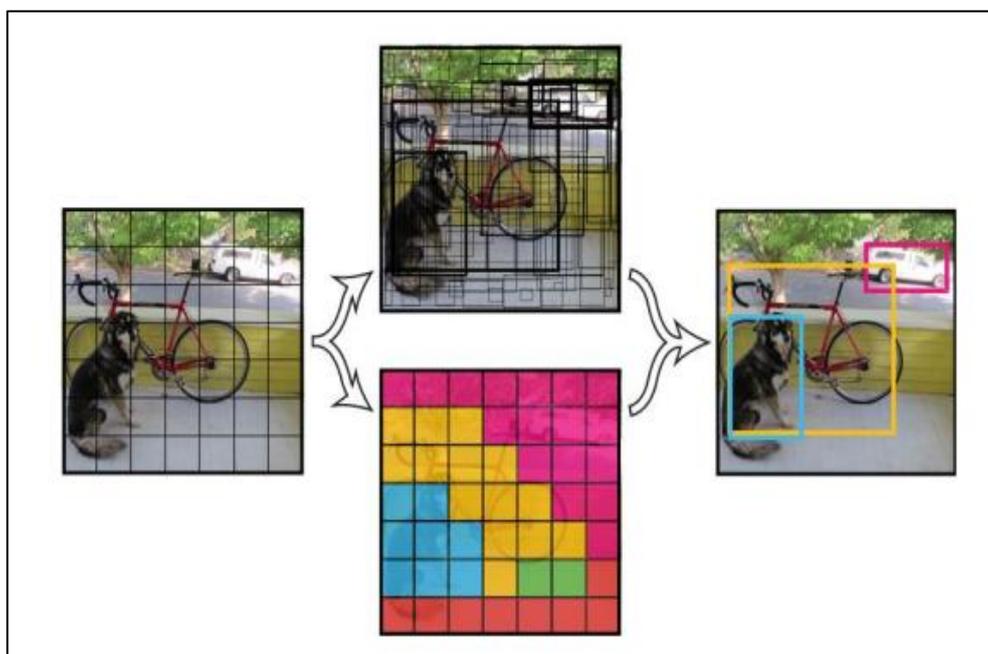


Figure 20: Principe de l'algorithme YOLO. [Redmon et al, 2016]

Plusieurs versions du YOLO ont été proposées à ce jour, chaque version a pour but d'améliorer les performances du modèle en matière de rapidité et taux de classification. Yolov5 ressemble presque à Yolov4 avec quelques différences suivantes : [Khelalef, 2020]

- Yolov4 est publié dans le framework Darknet, Yolov5 est basé sur le **framework PyTorch**. [Surya, 2022]
- Yolov4 utilise .cfg pour la configuration tandis que Yolov5 utilise le **fichier .yaml**.

Tous les modèles de base de YOLO ont été entraînés sur la base de données de COCO, mais il existe dans la littérature d'autres modèles entraînés sur d'autres bases de données telles que ImageNet de Google. [Khelalef, 2020]

III.2.3. Architecture R-CNN

Ce modèle combine des propositions de région avec des réseaux de neurones convolutifs pour détecter des objets dans une image via des bounding boxes (*figure 21*). La première étape de la détection est de générer des propositions de régions indépendantes des catégories contenant des objets probables. Au lieu de l'image entière, ces propositions sont transmises à un neurone convolutif profond du réseau qui renvoie un vecteur d'entité pour chaque proposition de région. La dernière étape est de passer ce vecteur à travers un ensemble de machines à vecteurs de support linéaire spécifiques à une classe (SVM). [KAZI TANI L., 2020]

Bien que R-CNN ait surpassé des architectures similaires, il y avait encore des lacunes. Le plus gros était la lenteur. Elle est causée principalement par trois éléments : la propagation directe à travers le CNN (chaque région de chaque image doit être passé séparément), par son triplet d'apprentissage (un réseau de génération de caractéristiques d'images, un réseau pour la décision de classe et le modèle de régression du Bounding Box), et par la génération de propositions de Bounding Boxes. [KAZI TANI L., 2020]

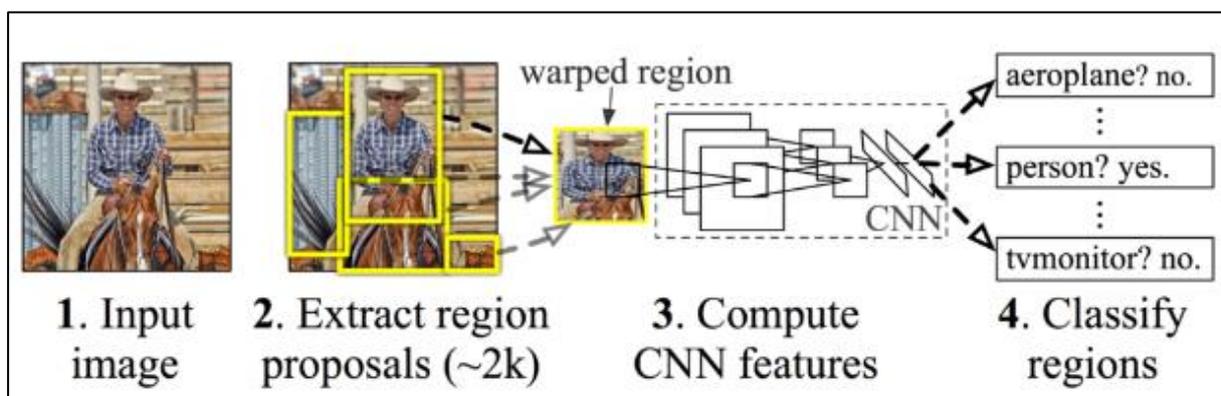


Figure 21: Architecture R-CNN. [Girshick et al., 2014]

III.2.4. Architecture Fast R-CNN

La raison principale d'une nouvelle architecture était d'accélérer le R-CNN, Ross Girshick a réussi à donner naissance à une nouvelle architecture nommée Fast R-CNN (*figure 22*). Outre l'objectif principal d'éviter les deux premiers problèmes, il améliore également sa précision.

Le premier problème, la propagation vers l'avant distincte pour chaque proposition de région, était résolu en propageant l'image entière pour obtenir une feature map avant la proposition de région. Pour chaque proposition d'objet, la feature map extrait un vecteur de caractéristiques de longueur fixe par la couche pooling de regroupement de régions d'intérêt (RoI) (a Region of Interest pooling layer). [KAZI TANI L., 2020]

Pour le deuxième problème, au lieu de trois modèles distincts, toutes les étapes sont jointes en un seul modèle en ajoutant la classification (en utilisant une couche softmax au lieu d'un SVM séparé) et une régression de bounding box en couches parallèles jusqu'à la fin du modèle. [KAZI TANI L., 2020]

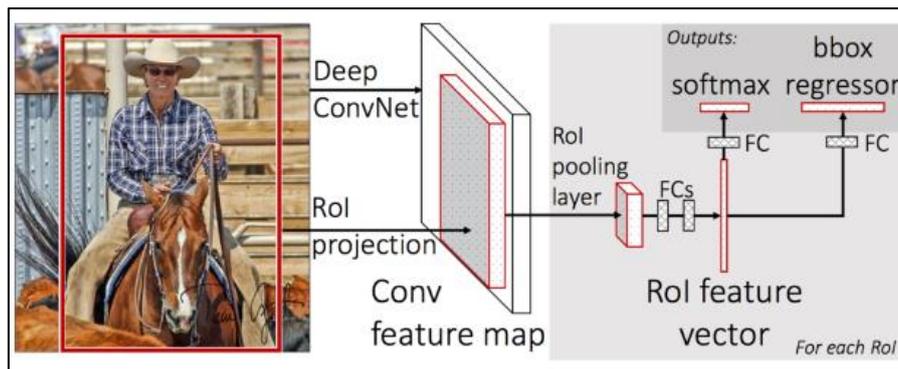


Figure 22: L'architecture Fast R-CNN. [GIRSHICK, 2015]

III.2.5. Architecture Faster R-CNN

Le troisième problème de vitesse, qui est la proposition de région basé sur la recherche sélective, a aussi été résolu en 2015 dans une architecture nommée Faster R-CNN.

L'architecture Faster R-CNN illustrée dans la (*figure 23*) est composée de plusieurs parties importantes. La première est un réseau CNN partagé chargé d'extraire les caractéristiques de l'image traitée. Ensuite, intervient le Region Proposal Network (RPN ou réseau de propositions de régions) dont le rôle est de trouver des propositions de boîtes rectangulaires pouvant contenir l'objet recherché. Enfin, après un regroupement des régions, chaque proposition de boîte retenue fait l'objet d'une classification afin de déterminer de quel chiffre il s'agit. [KAZI TANI L., 2020]

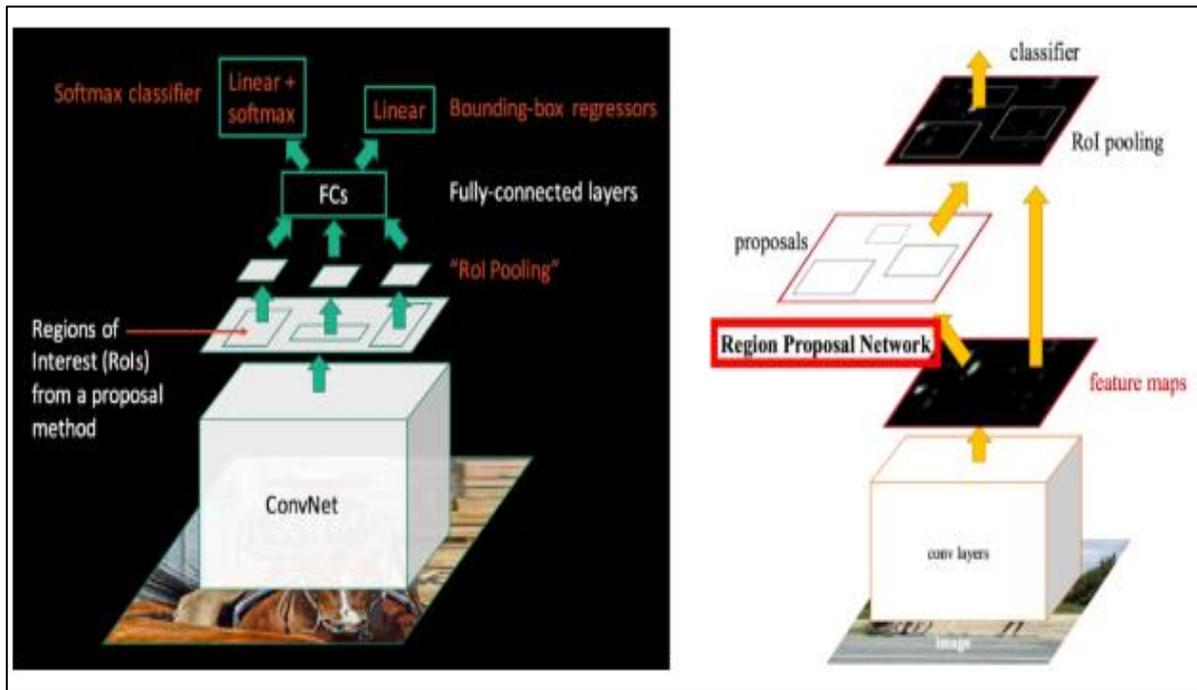


Figure 23: Architecture de FasterR-CNN. [Redmon et al, 2016]

Faster R-CNN a une architecture de deux étages comme VGG16, AlexNet, ResNet ou d'autres pour l'entraînement. Dans notre cas nous avons choisi de travailler avec ResNet-50.

Le tableau 3 donne une comparaison des caractéristiques importantes des modèles que nous avons parcourus.

	R-CNN	R-CNN rapide	R-CNN plus rapide
Méthode des propositions de régions	Recherche sélective	Recherche sélective	Réseau de proposition de région
Calendrier de prédiction	40-50 s	2 secondes	0,2 seconde
Calcul	Temps de calcul élevé	Temps de calcul élevé	Temps de calcul réduit
Le mAP sur l'ensemble de données de test Pascal VOC 2012 (%)	53.3	68,4 (lors de la formation avec VOC 2007 et 2012 les deux)	75,9 (lors de la formation avec VOC 2007 et 2012 et COCO)

Tableau 3: Comparaison de R-CNN, Fast R-CNN et Faster R-CNN.

III.2.5.1. ResNet-50

Les réseaux résiduels profonds comme le modèle populaire ResNet-50 sont des réseaux neuronaux convolutifs (CNN) de 50 couches de profondeur. Un réseau neuronal résiduel (ResNet) est un réseau neuronal artificiel (ANN) d'un type qui empile des blocs résiduels les uns sur les autres pour former un réseau. [Gaudenz Boesch, 2022]

ResNet, abréviation de Residual Networks, est un réseau de neurones classique utilisé comme colonne vertébrale pour de nombreuses tâches de vision par ordinateur. La percée fondamentale avec ResNet a été qu'il nous a permis de former des réseaux de neurones extrêmement profonds avec plus de 150 couches. [Aditi Rastogui, 2022]

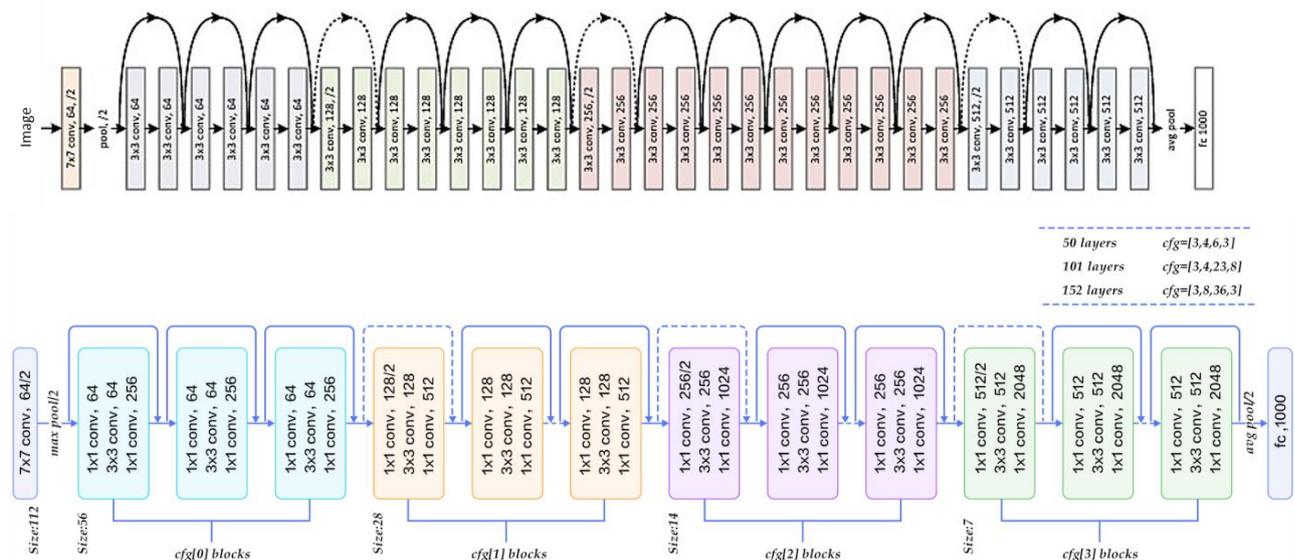


Figure 24: L'architecture du ResNet-50. [Aditi Rastogui, 2022]

Les réseaux de neurones convolutifs présentent un inconvénient majeur « **Vanishing Gradient Problem** » le problème de gradient de disparition. Pendant la rétropropagation, la valeur du gradient diminue de manière significative, donc pratiquement aucun changement ne vient aux poids. Pour surmonter cela, ResNet est utilisé. Il utilise "SKIP CONNECTION".

SKIP CONNECTION : est une connexion directe qui saute certaines couches du modèle. La sortie n'est pas la même en raison de ce saut de connexion. [Aditi Rastogui, 2022]

- Le problème de gradient de disparition

Dans Machine Learning, le problème de gradient de disparition est rencontré lors de la formation de réseaux de neurones avec des méthodes basées sur le gradient (exemple, rétropropagation). Ce problème rend difficile l'apprentissage et le réglage des paramètres des couches antérieures du réseau.

Le problème des gradients de fuite est un exemple de comportement instable qu'on peut rencontrer lors de la formation d'un réseau neuronal profond. [Dinesh Kumar, 2020]

Ce problème est causé par la dérivée de la fonction d'activation utilisée pour créer le réseau de neurones.

La solution la plus simple au problème consiste à remplacer la fonction d'activation du réseau. Car le problème peut être pire dans les réseaux de neurones profonds utilisant la fonction d'activation sigmoïde, cependant au lieu de sigmoïde, utilisez une fonction d'activation telle que ReLU. [Tina Jacob, 2022]

III.2.6. Mask R-CNN

Le Mask R-CNN a été proposé en 2017 par Facebook AI Research (**Figure 25**). Faster R-CNN a surpassé la plupart de ses concurrents dans le domaine de la détection d'objets. Mais c'était encore inutilisable pour certains utilisateurs à la recherche d'un réseau neuronal pour la segmentation. [KAZI TANI L., 2020]

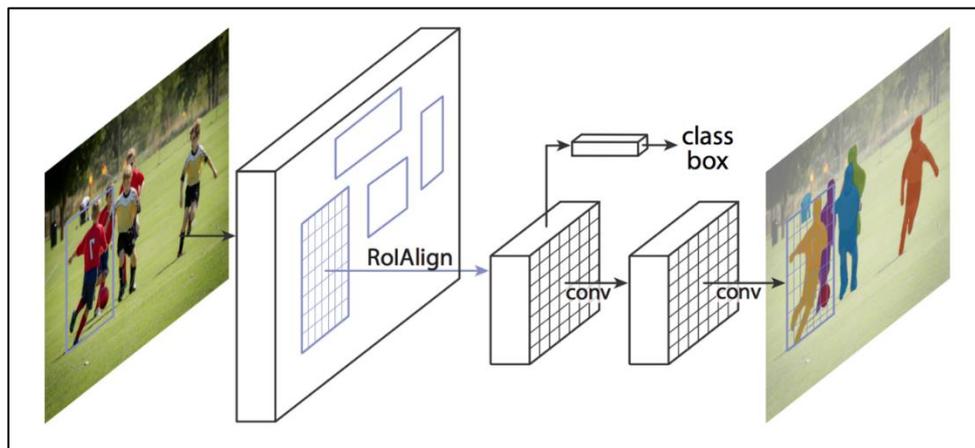


Figure 25: L'architecture du Mask RCNN. [KAZI TANI L., 2020]

Ils avaient un réseau puissant pour la détection d'objets, ils ont proposé une approche plus avancée, l'instance segmentation. Pour se faire avec une précision suffisante, il y avait un besoin de quelques changements dans l'architecture de Faster R-CNN. L'instance segmentation a deux objectifs clés à atteindre : effectuer une version de détection d'objets pour dessiner les bounding boxes autour de chaque instance d'une classe ; et effectuer une segmentation sémantique sur chacun des bounding boxes. [KAZI TANI L., 2020]

Un des avantages intéressants de ce modèle, est que si nous sommes capables de résoudre indépendamment le problème des différents bounding boxes et le problème de segmentation sémantique, nous avons également essentiellement résolu la tâche de l'instance segmentation.

III.2.7. Architecture de détection de véhicule avec YOLOv5

III.2.7.1. Base d'entraînement :

Vehicles-OpenImages Dataset est la base que nous avons choisie pour entraîner notre modèle à différencier une voiture, un bus, une moto, une ambulance et un camion.

Cet ensemble de données contient 627 images de différentes classes de véhicules pour la détection d'objets. Ces images sont dérivées des ensembles de données open source de vision par ordinateur Open Images.

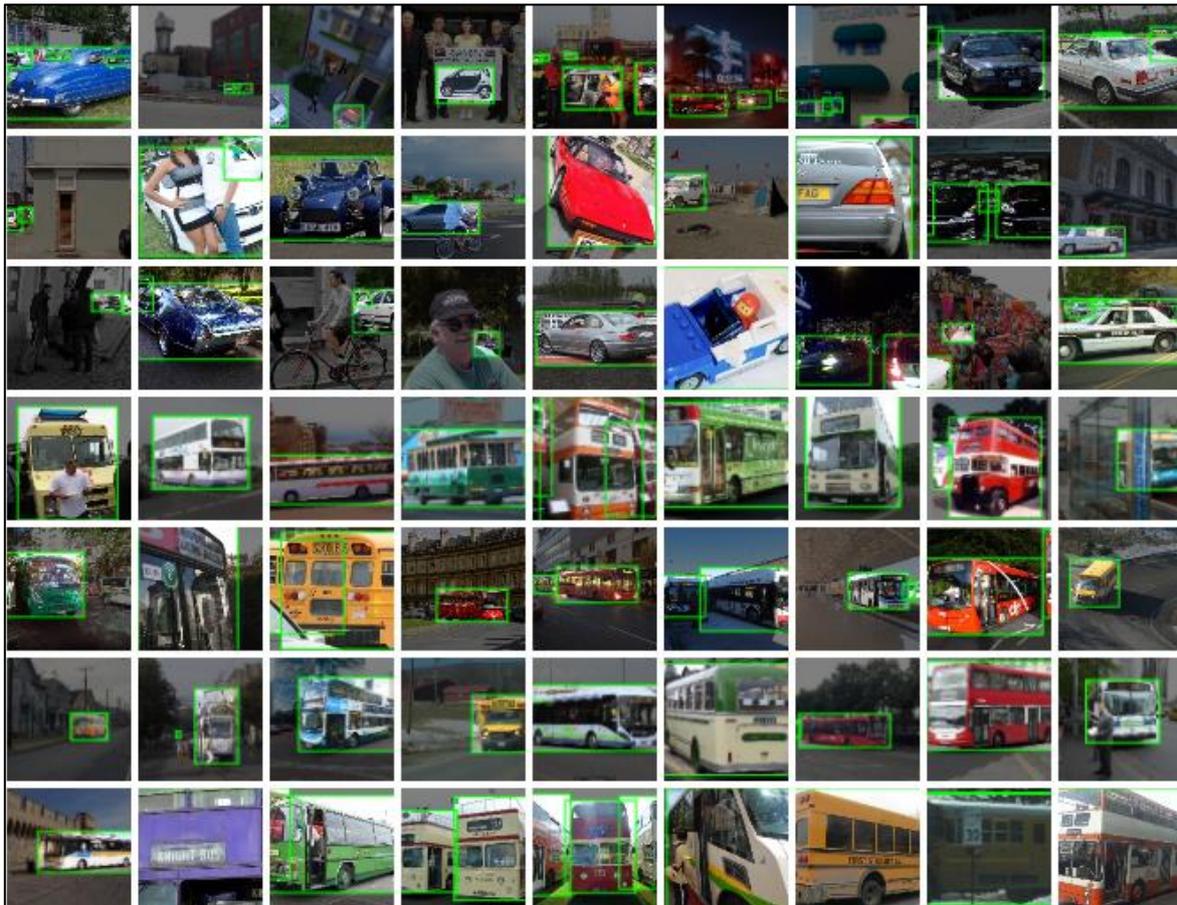


Figure 26: Quelques images de jeu de données vehicles-OpenImages.

III.2.7.2. Paramètres utilisés

- **Seuil de confiance** : confiance minimale pour qu'une boîte soit détectée.

```
--conf 0.7
```

- **La taille d'image** : 416*416

```
--img 416
```

- **Taille du lot d'entraînement** : contrôle le nombre d'échantillons d'apprentissage. On a choisi 32, chaque lot prend en charge 32 images à la fois.

- **Nombre d'itération(epochs)** : le nombre de fois ou l'algorithme d'apprentissage est reproduit sur l'ensemble des données. Dans notre cas on a pris une constante de 300.

```
--batch 32 --epochs 300
```

III.2.7.3. Résultats sur des images fixes

Les images dans (*figure 27*) nous montrent les différents résultats sur des images fixes obtenus avec YOLOv5.

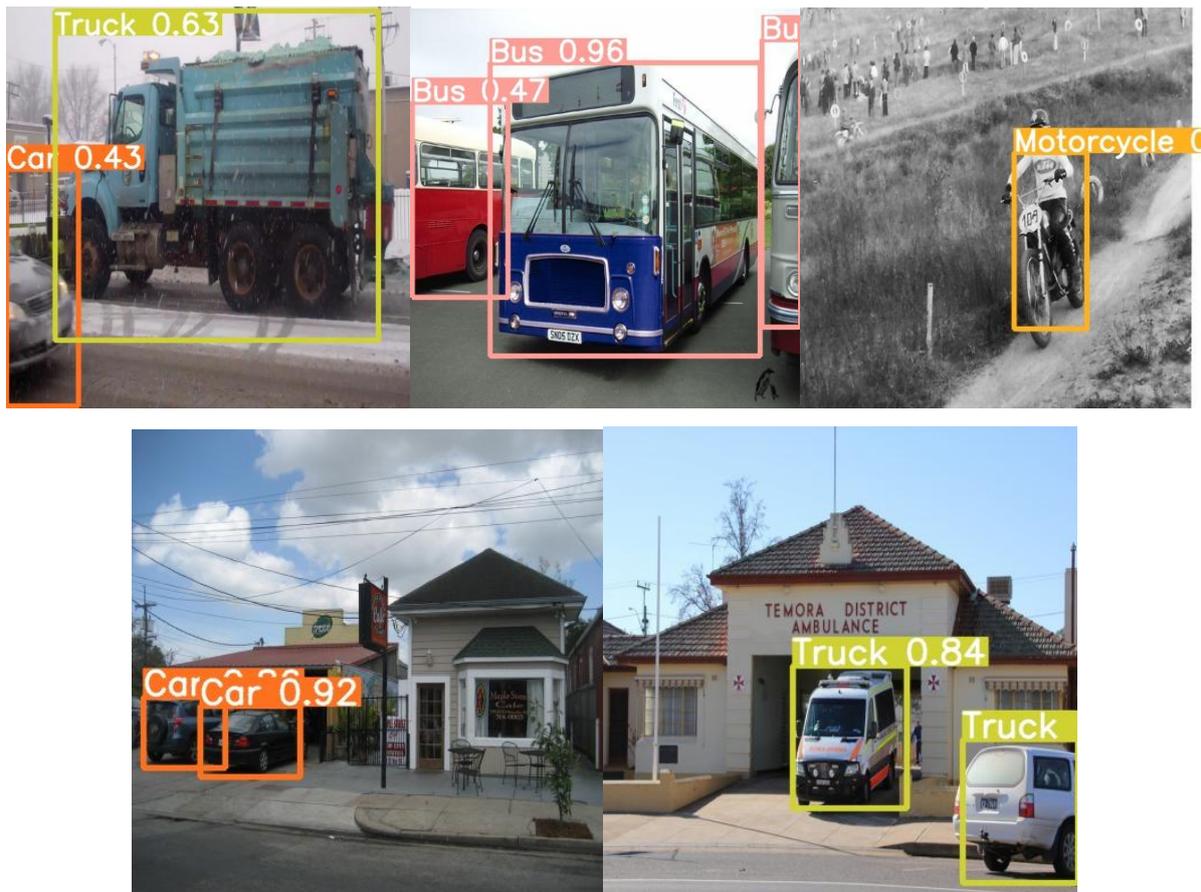


Figure 27: Résultats du modèle YOLOv5.

III.2.8. Détection de véhicule avec Faster R-CNN

L'architecture de notre système de détection de véhicules dans une vidéo, est résumée dans la (*figure 25*). Pour cet algorithme nous n'avons pas fait d'entraînement sur une base mais juste un teste du pré-traitement de Faster R-CNN sur la détection de 5 véhicules (car, bus, truck, bicycle, motorcycle). Chaque classe de la base d'apprentissage a un identifiant.

```
{2: {'id': 2, 'name': 'bicycle'},
 3: {'id': 3, 'name': 'car'},
 4: {'id': 4, 'name': 'motorcycle'},
 6: {'id': 6, 'name': 'bus'},
 8: {'id': 8, 'name': 'truck'},}
```

Tout d'abord, il identifie les régions d'intérêt, puis transmet ces régions à un réseau neuronal convolutif. Les cartes de caractéristiques générées sont transmises à une machine à vecteurs de support (SVM) pour classification. La régression entre les boîtes englobantes prédites et les boîtes englobantes de vérité terrain est calculée.

III.2.8.1. Base d'apprentissage

MsCOCO (Microsoft Common Objects in Context) est un ensemble de données d'images à grande échelle contenant 328 000 images d'objets du quotidien et d'humains. L'ensemble de données contient des annotations qu'on peut utiliser pour entraîner des modèles d'apprentissage automatique à reconnaître, étiqueter et décrire des objets.

III.2.9. Evaluation des performances

III.2.9.1. Matrice de confusion

Il s'agit d'un tableau de taille $n \times n$ pour visualiser les résultats des modèles prédictifs dans les problèmes de classification, où n est le nombre de classes dans l'ensemble de données (voir le tableau ci-dessous). Dans cette matrice on croise les classes cibles réelles avec les classes prédites obtenues. Ceci nous donne le nombre d'instances correctement classées et mal classées. [Fatoumata et Amor, 2021]

		Classes actuelles	
		Positive	Négative
Classes prédites	Positive	VP	FP
	Négative	FN	VN

Tableau 4: Matrice de confusion pour une classification binaire.

- **VP** : vrais positifs est le nombre d'instances positives correctement classifiées.
- **FP** : faux positifs est le nombre d'instances négatives et qui sont prédites comme positives.
- **FN** : faux négatifs est le nombre d'instances positives classifiées comme négatives.
- **VN** : vrais négatifs est le nombre d'instances négatives correctement classifiées.

À partir de la matrice de confusion on peut calculer plusieurs métriques. [Fatoumata et Amor, 2021]

- Sensitivité

La sensibilité ou le rappel (recall) est le pourcentage des instances positives correctement identifiées. [Mifdal, 2019]

$$\text{Recall} = \frac{VP}{VP + FN}$$

- Précision

La précision (precision) est le pourcentage de prédictions positives qui sont correctes. [Fatoumata et Amor, 2021]

$$\text{Precision} = \frac{VP}{VP + FP}$$

- Spécificité

La spécificité (specificity) est le pourcentage de cas négatifs qui ont été prédits comme négatifs. [Fatoumata et Amor, 2021]

$$\text{Specificity} = \frac{VN}{VN + FP}$$

III.3. Implémentation

III.3.1. Environnement et outils de développement

III.3.1.1. Google Colab

Colaboratory ou ‘Colab’, permet d’écrire et d’exécuter le code Python de son choix par le biais du navigateur. Offert par Google (gratuit), basé sur Jupyter Notebook et destiné à la formation et à la recherche dans l’apprentissage automatique. Cette plateforme permet d’entraîner des modèles de Machine Learning directement dans le Cloud. Colab permet :

- D’améliorer les compétences de codage en langage de programmation Python.
- De développer des applications en Deep Learning en utilisant des bibliothèques Python populaires telles que Keras, TensorFlow, PyTorch et OpenCV.
- D’utiliser un environnement de développement (Jupyter Notebook) qui ne nécessite aucune configuration, Mais la fonctionnalité qui distingue Colab des autres services est l’accès à un processeur graphique GPU, totalement gratuit. [Djaballah, 2021]



Figure 28: Interface de Google Colab.

III.3.1.2. Mode d'exécution GPU

Pour passer en mode GPU, dans la barre des options choisir "exécution" puis "modifier le type d'exécution" et mettre l'option "accélérateur matériel" en mode GPU.

Nous serons obligés de ré-exécuter tout le notebook. Cette fois-ci l'exécution du précédent prendra environ 4 minutes. Pour cet exemple l'entraînement sur un GPU est 60 fois plus rapide qu'un entraînement classique sur un CPU. [Touahri, 2020]

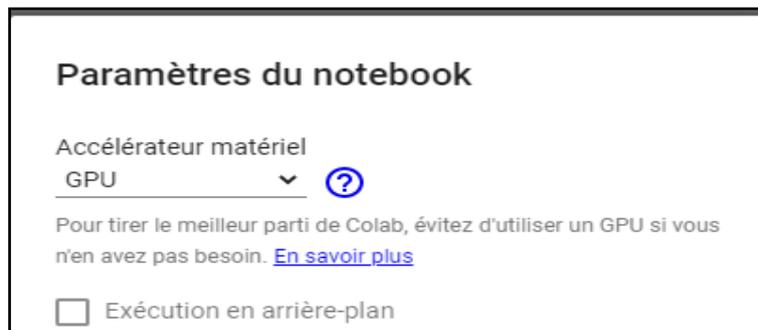


Figure 29: Paramètres du notebook sur Google Colab.

III.3.1.3. Langage Python

C'est un langage de programmation de haut niveau interprété (il n'y a pas d'étape de compilation), interactif et orienté objet avec une sémantique dynamique.

Python est conçu pour être hautement lisible et utilise fréquemment des mots clés en anglais, alors que d'autres langages utilisent la ponctuation, il a moins de constructions syntaxiques. Il est très sollicité par une large communauté de développeurs et de programmeurs. [Ounissi et Harnane, 2020]

III.3.1.4. Bibliothèques utilisées

- Torch

Torch est une bibliothèque d'apprentissage automatique open source, un cadre de calcul scientifique et un langage de script basé sur le langage de programmation Lua. Il fournit une large gamme d'algorithmes pour l'apprentissage en profondeur et utilise le langage de script LuaJIT et une implémentation en C sous-jacente. [Ghediri et Sakri, 2021]

- Matplotlib

Est une bibliothèque complète pour créer des visualisations statiques, animées et interactives en Python, il offre une alternative open source viable à MATLAB. [Zerargui et Benzaoui, 2021]

- TensorFlow

TensorFlow est la meilleure bibliothèque de toutes, Car elle est conçue pour être accessible à tous. La bibliothèque TensorFlow intègre différentes API pour construire à l'échelle des architectures d'apprentissage profond telles que CNN ou RNN.

TensorFlow est basé sur le calcul de graphe, il permet au développeur de visualiser la construction du réseau neural avec Tensorboard. Cet outil est utile pour déboguer le programme. Enfin, TensorFlow est conçu pour être déployé à grande échelle. Il fonctionne sur le CPU et le GPU. [Atif et Misseraoui, 2021]

- NumPy

NumPy est l'abréviation de « Numerical Python » et c'est un ensemble fondamental pour le calcul scientifique en Python. NumPy fournit à Python une vaste bibliothèque mathématique capable d'effectuer des calculs numériques de manière efficace et efficiente afin de pouvoir travailler avec des tableaux multidimensionnels et des structures de données matricielles, très courante dans les domaines de la science des données et de l'apprentissage automatique.

Pourquoi utiliser NumPy ?

- Les tableaux Numpy utilisent moins de mémoire que les listes python normales.
- Une liste python normale est un groupe de pointeurs pour séparer les objets Python (par exemple les nombres à l'intérieur de la liste).
- Un tableau Numpy est conçu pour être un tableau de valeurs uniformes, sans utiliser d'espace pour les pointeurs de type.
- Numpy peut également lire les informations plus rapidement et dispose de nombreuses opérations de diffusion pratiques qui peuvent être effectuées à travers les dimensions du tableau. [Bellahmer, 2020]

III.3.2. Résultats d'entraînement

	Précision	Rappel	Moyenne de précision (mAP)	Itérations
Car	0.397	0.311	0.305	300
Bus	0.61	0.478	0.506	300
Truck	0.393	0.173	0.252	300
Motorcycle	0.494	0.348	0.359	300
Ambulance	0.667	0.781	0.747	300

Tableau 5: Rapport de classification.

Les résultats que nous avons obtenus (*tableau 5*) ne sont pas bon, ils sont tous inférieur à 50%, il faut entrainer beaucoup plus notre modèle. Les images des ambulances sont bien classifiées, car parmi les cinq classes, l'ambulance est celle qui présente les plus grands nombres de résultat.

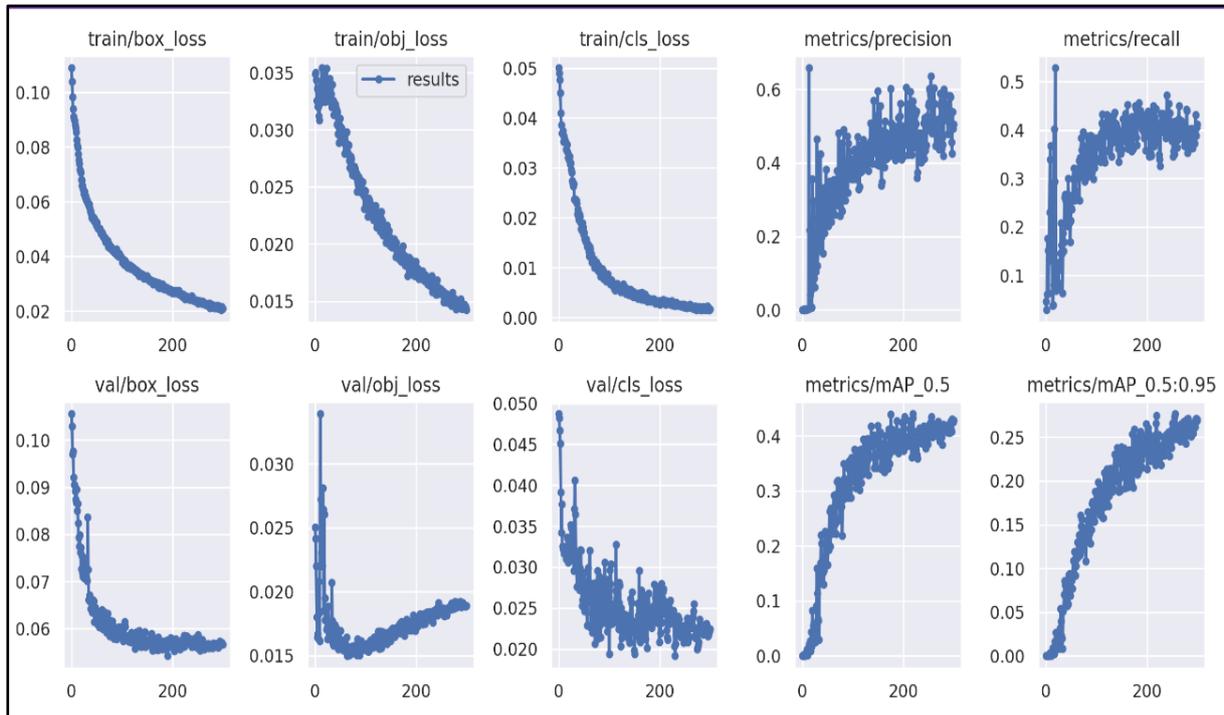


Figure 30: Graphes de métriques (YOLOv5).

- La fonction de perte

Dans le domaine de l'apprentissage profond et de l'apprentissage automatique, la « Perte » est la perte d'une mauvaise prédiction. D'une autre façon, la perte représente la pénalité générée lorsque l'estimation de la cible fournie par le modèle d'apprentissage-machine n'est pas parfaitement égale à la cible. Une fonction de perte quantifie cette pénalité sous la forme d'une valeur individuelle. [HOURLANE Oumaima, 2018]

Une tâche de prédiction peut être considérée comme une simple tâche d'optimisation. Le modèle tente d'optimiser ses performances en prédisant la valeur correcte. La valeur réelle que nous optimisons, s'appelle la « perte » (que nous essayons de minimiser, bien sûr). Pour ce faire, le modèle doit pouvoir mesurer ce que l'on appelle la « perte », qui dépend du problème.

$$L(W) = \underbrace{\frac{1}{N} \sum_{i=1}^N L_i(f(x_i, W), y_i)}_{\text{Data loss: Model predictions should match training data}} + \underbrace{\lambda R(W)}_{\text{Regularization: Model should be "simple", so it works on test data}}$$

Généralement, la fonction de perte, L , est une fonction sur l'entrée, un ensemble de paramètres (appelés "poids") et le vrai label (Dans la famille multi-classe, L_i est défini comme la perte sur la classe i).

R est une fonction de régularisation. Cette fonction est utilisée pour pénaliser W "complexe" (par exemple, elle "préférer" W s plus petit). Cela force le modèle à préférer les modèles plus simples aux plus complexes. [HOURLANE Oumaima, 2018]

Lambda, le coefficient de R , est un autre paramètre que ce processus optimisé.

- Matrice de confusion

La matrice de confusion (*figure 31*) nous permet de voir exactement ce que notre modèle donne comme résultats de classification. La première classe a donné le meilleur pourcentage de classification (72%) comme nous l'avons décrit avec le rapport de classification présenté au Tableau 4 donnant la précision.

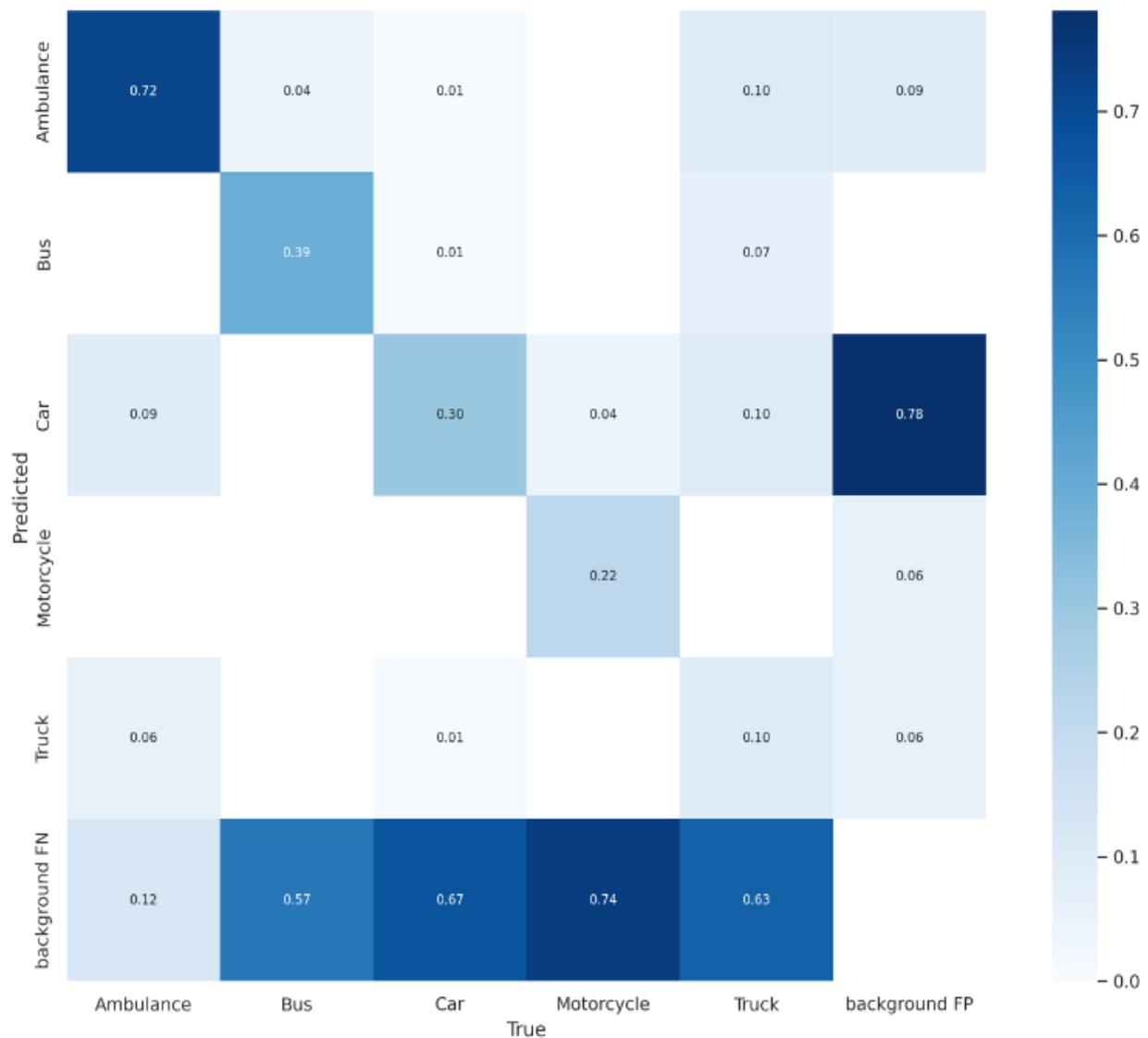


Figure 31: Matrice de confusion (YOLOv5).

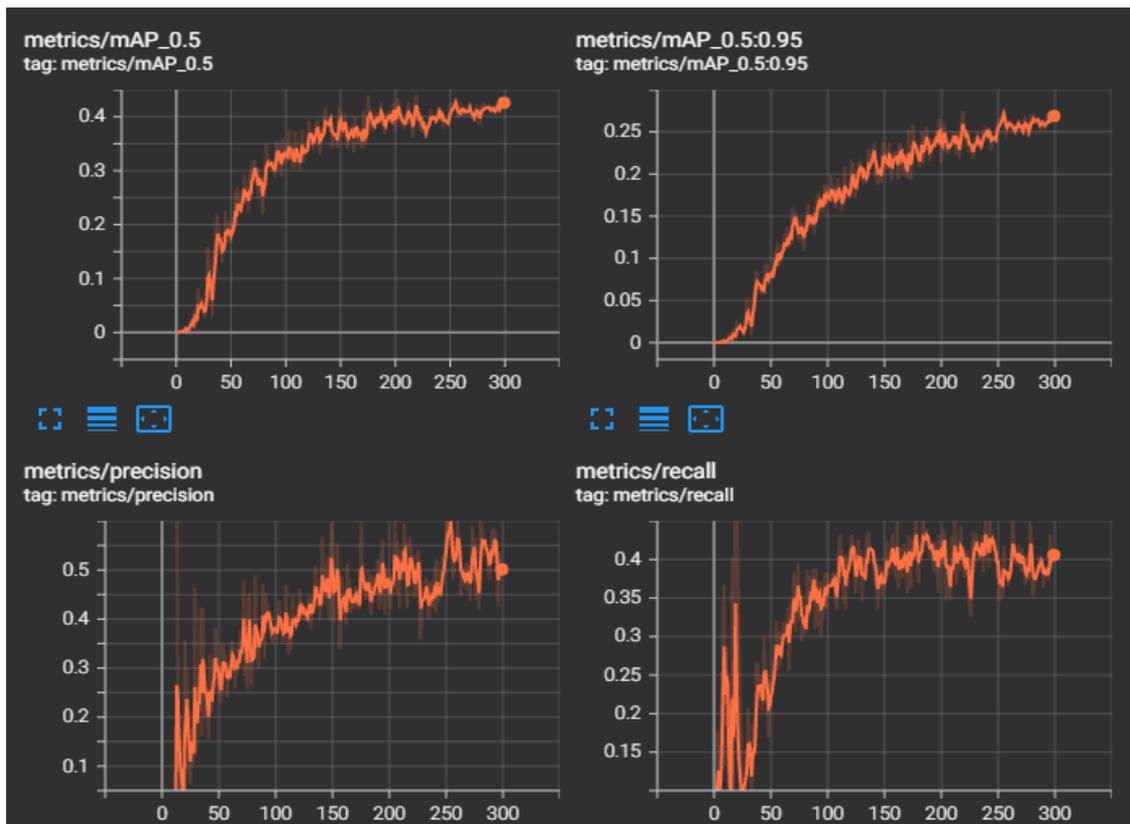


Figure 32: Estimateurs (metrics) (YOLOv5).

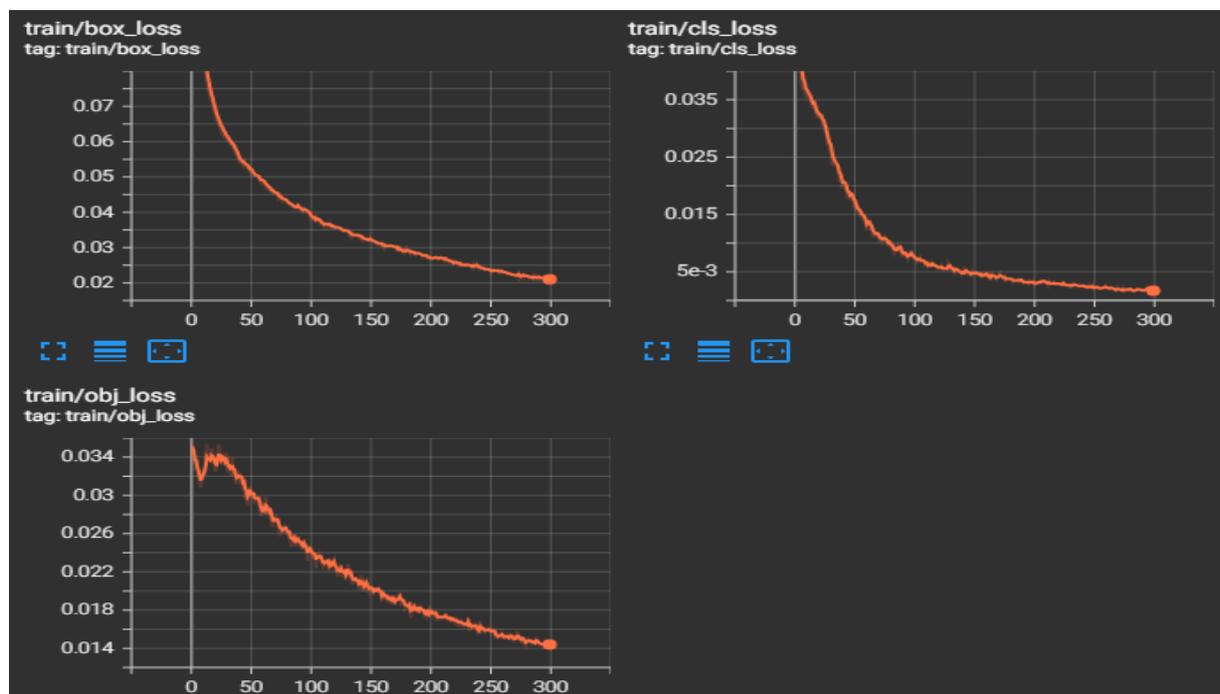


Figure 33: La fonction loss (YOLO V5).

Le tableau de bord Scalars (*figure 32 et 33*) montre comment la perte et les métriques changent à chaque époque nous pouvons également voir la vitesse d'apprentissage, le taux d'apprentissage et d'autres valeurs scalaires.

Le but de l'algorithme d'apprentissage profond est de minimiser la fonction de perte qui reflète l'erreur entre la prédiction et la vérité terrain que nous avons fourni.

Pour un bon modèle d'entraînement la tendance à la baisse des courbes perdues pour chaque époque et éventuellement aplanir. Dans nos graphes nous voyons que notre modèle a besoin de beaucoup d'entraînement pour avoir de meilleurs résultats.

La précision est une métrique que notre algorithme d'apprentissage profond utilise pour évaluer la qualité du modèle pendant la formation cela devrait avoir une tendance à hausse indiquant que la qualité du modèle s'améliore effectivement et que la précision ne fournit pas d'information sur la qualité de la segmentation pour chacune des classes et de l'arrière-plan de nos images.

- Comparaison

Cette partie représente une comparaison entre le YOLOv5 et le Faster R-CNN :

- YOLOv5 est une version récente de la famille de modèles YOLO, il fonctionne plus rapidement que le Faster R-CNN en raison de son architecture plus simple. Contrairement au Faster R-CNN, il est formé pour effectuer la classification et la régression de la boîte englobante en même temps.
- Faster R-CNN est plus lent, donc il ne représente pas le bon choix pour la prédiction de la détection en temps réel.
- Le taux de précision est plus élevé avec l'architecture Faster R-CNN.
- YOLOv5 a un avantage en termes de vitesse d'exécution, il fonctionne environ 2,5 fois plus vite.
- Faster R-CNN est bien meilleur pour la détection des objets les plus petits contrairement au Modèle YOLO ;
- Les résultats du modèle YOLOv5 sont avec peu ou pas de cases qui se chevauchent.

Les figures suivantes montrent une démonstration de la détection de véhicules par les deux algorithmes en utilisant les mêmes séquences d'images :



Figure 34: Détection de véhicules par YOLOv5 (1).



Figure 35: Détection véhicules par Faster R-CNN (1).

Séquence 1	Méthode	Précision
Car	YOLO	73%
	Faster R-CNN	94%

Tableau 6: Comparaison entre les deux méthodes (séquence 1).



Figure 36: Détection de véhicules par YOLO (2).



Figure 37: Détection de véhicules par Faster R-CNN (2).

Séquence 2	Méthode	Précision
Motorcycle	YOLO	32%
	Faster R-CNN	85%

Tableau 7: Comparaison entre les deux méthodes (séquence 2).



Figure 38: Détection de véhicules par YOLO (3).



Figure 39: Détection de véhicules par Faster R-CNN (3).

Séquence 3	Méthode	Précision
Car	YOLO	72%
	Faster R-CNN	100%

Tableau 8: Comparaison entre les deux méthodes (séquence 3).



Figure 40: Détection de véhicules par YOLO (4).



Figure 41: Détection de véhicules par Faster R-CNN (4).

Séquence 4	Méthode	Précision
Car	YOLO	90%
	Faster R-CNN	98%

Tableau 9: Comparaison entre les deux méthodes (séquence 4).

III.4. Conclusion

Ce chapitre a été divisé en deux parties : la 1^{ère} partie est la conception dans laquelle nous avons vu les architectures des deux modèles choisis YOLOv5 et Faster R-CNN ainsi que ResNet-50. La 2^{ème} partie concerne l'implémentation, où nous avons vu notre environnement de travail, les bibliothèques utilisées avec l'utilité de chacune d'elles et les résultats de la détection de véhicules dans une séquence d'images obtenus.

Nous avons clôturé ce chapitre par une comparaison entre les deux algorithmes et d'après l'étude comparative on peut déduire que YOLOv5 a l'intérêt d'être robuste permettant la détection de véhicules dans une vidéo avec une vitesse plus rapide que RCNN.

Conclusion générale

Au-delà d'être un problème général de la vision par ordinateur, la reconnaissance d'objets dans des catégories très variables est une étape indispensable pour de nombreuses applications spécifiques.

Notre projet se place dans le cadre du contrôle routier notamment d'une étude de différents algorithmes de détection de véhicules. Nous avons pu ici passer en revue la théorie et les résultats de deux algorithmes largement utilisés dans la détection de ces primitives.

D'autre part l'étude approfondie de la littérature sur la détection de véhicules nous a permis de définir une taxinomie des méthodes proposées, et de remarquer que la plupart des solutions consistent à considérer les séquences vidéo comme des successions d'images.

Ce projet nous a aussi permis de découvrir ce qu'est le travail de recherche, plus précisément de développer cet esprit de recherche sur un sujet qui nous était totalement inconnu et d'appliquer les diverses connaissances acquises au cours de notre parcours universitaire, que ce soit les notions mathématiques mais aussi l'intelligence artificielle et les algorithmes de classification que nous avons beaucoup étendu durant nos cours.

De plus, ce travail nous a permis de découvrir un domaine particulier qu'est la détection et la reconnaissance de véhicules dans un contexte très concret qu'est celui de la sécurité dans les transports.

Pour la réalisation de notre mémoire, nous avons utilisé les réseaux de neurones convolutifs et nous avons fait nos tests sur 4 classes de véhicules (voiture, camion, bus, bicyclette) qui décidera par la suite de la classe de chaque véhicule détecté.

L'objectif final est la détection ainsi que la classification des véhicules dans une séquence d'images.

Nous avons envisagé quelques perspectives qui sont les suivantes :

- Tester les algorithmes sur des images satellitaires ;
- Comparer les résultats de méthodes entre plus que deux architectures de CNN ;
- Entraîner les modèles plus longtemps sur d'autres base de données ;
- Utilisation de vidéo haute résolution (HD).
- Evaluer une application pour l'utiliser comme un radar qui permettra de détecter la vitesse des véhicules et leurs matricules.

Bibliographie

ABIDI Amina, SLIMANI Mabrouka (2016), « *Suivi d'objet en mouvement dans une séquence d'images* », Mémoire de fin d'étude, Université Ahmed Draia – Adrar, p19.

Aditi Rastogui (2022), « *ResNet50* », Publié dans Génie du développeur le 14 Mars, [consulter le 28 juin 2022]. Disponible sur: <https://blog.devgenius.io/resnet50-6b42934db431>

Ali Tourani et al (2019), « *A RobustVehicleDetectionApproachbased on Faster R-CNN Algorithm* », 978-1-7281-1621-1.

AMMAR M, Ana G, BENNAMOUN M, et Senjian A. (2018), « *Automatic Hierarchical Classification of Kelps Using Deep Residual Features?* ».

ANDRIAMIANDRASOA I. (2019), « *SYSTEME DE DETECTION D'OBJET BASE SUR LE DEEP LEARNING DANS UN MAGASIN INTELLIGENT* », Mémoire de fin d'études, Université D'ANTANANARIVO, Août, 101 p.

ATIF Rahil et MISSERAOUI Nacera (2020), « *Evaluation d'un algorithme de Deep Learning pour l'amélioration du taux de détection d'objets multiples dans des scènes denses* », Mémoire de Master, Université de Blida, p32.

ATIK.M, CEPNIS et DURAN.Z (2020), « *Vehicle Detection Using Different Deep Learning Algorithms from Image Sequence* », vol 8, N0.2,347-358, p 357.

Aurélié Bugeau (2011), « *Détection et suivi d'objets en mouvement dans des scènes complexes, application à la surveillance des conducteurs.* », Thèse, Université de RENNES I, janvier, p17.

BELLAHMER H. (2020), « *Implémentation et évaluation d'un modèle d'apprentissage automatique pour l'estimation de la valeur marchande de propriétés immobilières* », Mémoire de fin d'études, UNIVERSITÉ MOULOUD MAMMERI DE TIZI-OUZOU.

Benahmed Khelifa, Benyamina Zakaria, Bounaama Fateh (2017), « *les systèmes de transport intelligente (STI)* », Journées portes ouvertes sur la Faculté des Sciences Exactes JFSE 2017, Juin, 15p.

BENDAOU Y. (2014), « *Prédiction Des Résistances Mécaniques Des Bétons à Base Des Ciments Composés En Utilisant Les Réseaux Neurones Artificiels* », Mémoire, Université CONSTANTINE I, juillet, 107 p.

BENSIAH O. (2020), « *La proposition d'une nouvelle approche basée Deep Learning pour la prédiction du cancer du sein* », Mémoire de fin d'études, Université L'arbi Ben M'hidi Oum El Bouaghi, 86 p.

BISIMWA M. (2011), « *Caractérisation et extraction informatique de la structure d'un tableau par une méthode implémentant un réseau de neurones* », Mémoire, Institut Supérieur Pédagogique.

Boughaba M & Boukhris B. (2017), « *L'apprentissage profond (Deep Learning) pour la classification et la recherche d'images par le contenu* », Mémoire de fin d'études, Université KASDI MERBAH OUARGLA, 43 p.

DILMI A. (2012), « *APPRENTISSAGE PAR RENFORCEMENT EN UTILISANT LES RESEAUX DE NEURONES ARTIFICIELS : SIMULATION DE LA NAVIGATION D'UN ROBOT MOBILE* », Mémoire de fin d'études, Université de MSILA, juin, 74 p.

Dinesh Kumar (2020), « *The Vanishing Gradient Problem* », le 19 septembre, Great Learning, [consulter le 28 juin 2022]. Disponible sur : <https://www.mygreatlearning.com/blog/the-vanishing-gradient-problem>

DJABALLAH M. (2021), « *Système de prédiction de la consommation d'énergie basé Deep Learning* », Mémoire de fin d'études, Université de 8 Mai 1945 – Guelma, septembre, 35p.

Dominique PASTRE (2000), « *L'INTELLIGENCE ARTIFICIELLE DEFINITION - GENERALITES - HISTORIQUE – DOMAINES* », cours, Module Intelligence Artificielle, Université Paris 5, Février, 3-8p.

FABIEN M (2018), « *IA : vers des robots intelligents ?* », Revue de l'Electricité et de l'Electronique.

FARDJALLAH Rachid, BOUGHAZI Wafa (2014), « *Analyse d'un tronçon routier et suivi des véhicules pour le monitoring du trafic urbain* », Mémoire de Master, Université de GUELMA, juin, p37.

FATOUMATA Y & AMOR A. (2021), « *Machine learning pour la maintenance prédictive* », Mémoire de fin d'études, Université Larbi Ben Mhidi d'Oum El-Bouaghi, juillet, 42p.

GANNA M. (2016), « *Implémentation d'un réseau de neurones dans un microcontrôleur* », Mémoire de fin d'études, Université MOULOUD MAMMERI TIZI-OUZOU, juin, 76 p.

Gaudenz Boesch (2022), « *Deep Residual Networks (ResNet, ResNet50) – Guide in 2022* », Viso.ai, [consulter le 28 juin 2022]. Disponible sur : <https://viso.ai/deep-learning/resnet-residual-neural-network/>

GHEDIRI N & SAKRI Z. (2021), « *ETUDE ET EXPERIMENTATION DES RESEAUX RESNET-50 ET INCEPTION-V3 DANS LA CLASSIFICATION DE CANCER DE LA PEAU* », Mémoire de fin d'études, Université Larbi Ben M'Hidi d'Oum El Bouaghi.

HARIANTO A et al. (2021), « *Data Augmentation and Faster RCNN Improve Vehicle Detection and Recognition* », 978-1-6654-0514-0, Avril, 128p.

HICHAM CHAOUI (2002), « *CONCEPTION ET COMPARAISON DE LOIS DE COMMANDE ADAPTATIVE À BASE DE RÉSEAUX DE NEURONES POUR UNE ARTICULATION FLEXIBLE AVEC NON-LINÉARITÉ DURE* », Mémoire, Université du QUÉBEC À TROIS-RIVIÈRES, Décembre, p28.

HOURRANE Oumaima (2018), « *Brève introduction aux fonctions de perte (Loss Function)* », DataScienceToday le 26 juin, [consulter le 30 juin 2022]. Disponible sur : <https://www.datasciencetoday.net/index.php/fr/machine-learning/149-breve-introduction-aux-fonctions-de-perte-loss-function>

Innocent Mateyaunga, « *Prédictive Maintenance Using Machine Learning* », la thèse de master, sous la direction de Hadj Abdelkader, Faculté de technologie de l'université de Tlemcen, 2020, 61p.

Intelligence artificielle Vs Machine learning Vs Deep learning Vs Data Science, août 2020, [consulté le 15 juin 2022]. Disponible sur : <https://medium.com/@designer.web174/artificial-intelligence-vs-machine-learning-vs-deep-learning-vs-data-science-10e22cbc6ebe>

Jean-Paul HATON, Marie-Christine HATON (2013), *Résolution de problèmes et modèles de raisonnement*, 4 décembre 2013 [consulté le 13 mars 2022]. Disponible sur : <https://lacademie.tv/conferences/resolution-de-problemes-et-modeles-de-raisonnement>

John W.Branch, , Jorge E.Espinosa, Sergio et A.Velastin (2017), «*Vehicle Detection Using Alex Net and Faster R-CNN Deep Learning Models: A Comparative Study*», 10645, p. 3-15.

José M. Costa (2006), *Systèmes de transport intelligents, manuel sur les communications mobiles terrestres*, Union internationale des télécommunications, 3p, ISBN : 92-61-11872-0.

KAZI TANI L. (2020), « *Conception et Réalisation d'un Outil Semi-Automatique Dédié à l'Analyse et la Recherche du Contenu Vidéo du Web* », Thèse, Université d'Oran, juin, 59p.

KHELAL A. (2020), « *Reconnaissance d'activités humaines en utilisant les descripteurs spatio-temporels 2D/3D* », Thèse, Université Batna 2 Mostefa Ben Boulaïd, juin, 114p.

KHODJA F. (2011), « *Conception d'un système intelligent à base de réseaux de neurones artificiels pour l'étude de la dynamique des streamers à la surface des polymères* », Mémoire, Université des Sciences et de la Technologie d'Oran, octobre, 101 p.

L'histoire de l'intelligence artificielle, CEA, 11 avril 2018 [consulté le 8 février 2022]. Disponible sur : <https://www.cea.fr/multimedia/Pages/videos/culture-scientifique/technologies/histoire-intelligence-artificielle.aspx#:~:text=%E2%80%8BLes%20premi%C3%A8res%20machines%20pensantes,d'imaginer%20des%20machines%20pensantes.>

MARGOT G. (2021), « *Détection et reconnaissance de chiffres manuscrits à partir de plans cadastraux anciens par apprentissage profond* », Mémoire de fin d'études, le cnam GeF, septembre.

M. BENBRAHIM (2020), « *Chapitre I : Introduction à l'intelligence artificielle* », Cours pour master, Université Batna 2, 2p.

Mezaache H. (2008), « *Les réseaux de Neurones formels Et Les systèmes Neuro-Flous Pour l'apprentissage par renforcement* », Mémoire de fin d'études, Université El Hadj Lakhdar Batna, juillet, 76 p.

M. Gérard VOISIN (2009), « *Le déploiement de systèmes de transport intelligents dans le domaine du transport routier et d'interfaces avec d'autres modes de transport* », rapport d'information, LA COMMISSION DES AFFAIRES EUROPÉENNES, n°2134, Décembre, 14p.

MIFDAL R, « *Application des techniques d'apprentissage automatique pour la prédiction de la tendance des titres financiers* », L'obtention De La Maîtrise, Sous la direction de M. Edmond Miresco, École De Technologie Supérieure Université Du Québec, 2019, p 176.

MOHAMED.A. BOUHEDADJA (2021), « *UTILISATION DE L'APPRENTISSAGE PROFOND POUR LA CLASSIFICATION DES MAUVAISES HERBES DANS LE BLEUET NAIN* », Mémoire, UNIVERSITÉ DU QUÉBEC, Avril, p38.

- Moustafa ZOUINAR (2020), « *Évolutions de l'Intelligence Artificielle : quels enjeux pour l'activité humaine et la relation Humain-Machine au travail ?* », activités n°4941, Janvier, 3p.
- OUKACINE N. (2012), « *Utilisation des réseaux de neurones pour la reconstitution de défauts en évaluation non destructive* », Mémoire de Magister, Université Mouloud Mammeri Tizi-ouzou, juillet, 111p.
- Olivier Boisard (2020), « *Brève histoire de l'Intelligence Artificielle* », cours, Ecole Centrale de Lille, Mars, 1p.
- Olivier EZRATY (2018), Les usages de l'intelligence artificielle, Paternité-Pas d'Utilisation Commerciale-Pas de Modification 2.0 France, 21p,23p, ISBN.
- OULMI M & KALOUNE S. (2018), « *Classification d'objets avec le Deep Learning* », Mémoire de fin d'études, Université Akli Mohand Oulhadj de Bouira, 56 p.
- OUNISSI M & HARNANE Z. (2020), « *Modélisation et classification avec Deep Learning Application à la détection du Coronavirus Covid-19* », Mémoire de fin d'études, Université Mohamed Larbi Ben M'hidi - Oum El Bouaghi, 30 p.
- RERBAL Ghouti, FOUKIA Salah Eddine (2017), « *Détection et comptage des véhicules dans des séquences vidéo* », Mémoire de Master, Université de Tlemcen, juin, p35.
- S.Karungaru, L.Dongyang, et K.Terada (2021), « *Stephen Karungaru, LyuDongyang, and Kenji Terada* », Vol. 11, No. 4, Juillet, 310p.
- Senani S. (2019), « *Réseaux de neurones convolutionnels pour la détection précoce de la rétinopathie diabétique* », Mémoire de fin d'études, Université Mouloud Mammeri de Tizi-Ouzou, juillet, 103 p.
- SURYA G. *Algorithme de détection d'objets - Architecture YOLO v5*, août 2021, [consulté le 15 juin 2022]. Disponible sur : <https://medium.com/analytics-vidhya/object-detection-algorithm-yolo-v5-architecture-89e0a35472ef>
- SYSTEMES DE TRANSPORT INTELLIGENTS (STI), SINGLE WINDOW FOR LOGISTICS, 16 novembre 2018 [consulter le 7 mars 2022]. Disponible sur : <https://logistics.public.lu/fr/why-luxembourg/logistics-infrastructure/intelligent-transportation-system.html>
- Tina Jacob (2022), « *Problème de gradient de fuite, expliqué* », le 25 février 2022, [consulter le 29 juin 2022]. Disponible sur : <https://www.kdnuggets.com/2022/02/vanishing-gradient-problem.html#>
- TOUAHRI (2019), « *Détection et Classification des Véhicules par Réseaux de Neurones à Convolution CNN* », Mémoire de Master, Université de Guelma.
- Trafic routier Alger : 25 carrefours « *intelligents* » Opérationnels à la fin février. Radio Algérienne, 4 février 2019 [consulté le 12 avril 2022]. Disponible sur : <https://radioalgerie.dz/news/fr/article/20190204/161607.html>
- Virginie MATHIVET (2014), L'Intelligence Artificielle pour les développeurs. Concepts et implémentations en C#, Editions ENI, 21,23p, ISBN: 978-2-7460-9215-0.

Zeki Murat Çınar, Abubakar Abdussalam Nuhu, Qasim Zeeshan, Orhan Korhan, Mohammed Asmael and Babak Safaei, « *Machine Learning in Predictive Maintenance towards Sustainable Smart Manufacturing in Industry 4.0* » Sustainability, Octobre 2020.

ZERARGUI F & BENZAOUI O. (2021), « *Le discours de haine sur le web et les médias sociaux* », Mémoire de fin d'études, Université de Bordj Bou Arreridj Mohammed El Bachir El Ibrahimi, 28p.

Résumé

La sécurité routière et autoroutière est devenue un enjeu prioritaire pour les autorités publiques en raison de l'augmentation massive des accidents de la route chaque année. A cet effet, la détection de véhicules perçus dans une vidéo a simplifié la question du trafic et l'identification des automobiles ou des véhicules suspects. Les techniques d'apprentissage profond sont apparues comme une stratégie puissante offrant une solution alternative à l'imitation de la vision humaine. Ces techniques sont basées sur l'apprentissage profond.

Ce mémoire entre dans le cadre du domaine de la vision par ordinateur. Notre objectif est de détecter des véhicules dans un flux d'images et les classer selon leur type 'bus, voiture, motorcycle, camion, bicyclette'. Pour ce faire, nous avons adopté les deux modèles YOLOv5 et Faster RCNN pour la détection de véhicules. A l'issue d'une étude comparative entre ces modèles, les meilleurs résultats ont été obtenus par la méthode Faster RCNN avec une valeur de précision de 89%.

Mots-Clés : La détection de véhicules, réseau de neurones, apprentissage profond, CNN, 'ResNet-50', 'YOLO', 'Faster R-CNN'.

ملخص

أصبحت سلامة الطرق والطرق السريعة قضية ذات أولوية للسلطات العامة بسبب الزيادة الهائلة في حوادث المرور كل عام. وتحقيقاً لهذه الغاية، فإن اكتشاف المركبات في مقطع فيديو أدى إلى تبسيط مسألة المرور وتحديد السيارات أو المركبات المشبوهة. ظهرت تقنيات التعلم العميق كاستراتيجية قوية تقدم حلاً بديلاً لتقليد الرؤية البشرية. تعتمد هذه التقنيات على التعلم العميق. تقع هذه الأطروحة في نطاق رؤية الكمبيوتر. هدفنا هو اكتشاف المركبات في مجموعة من الصور وتصنيفها وفقاً لنوعها "حافلة، سيارة، دراجة نارية، شاحنة، دراجة". للقيام بذلك، اعتمدنا النموذجين YOLOv5 و Faster RCNN للكشف عن المركبات. في نهاية هذه الدراسة، تم الحصول على أفضل النتائج باستخدام طريقة Faster RCNN بقيمة الدقة تبلغ 89%.

كلمات البحث: اكتشاف المركبات، الشبكة العصبية، التعلم العميق، CNN، "ResNet-50"، "YOLO"، "Faster R-CNN".

Abstract

Road and highway safety has become a priority issue for public authorities due to the massive increase in road accidents every year. To this effect, the detection of vehicles perceived in a video has simplified the issue of traffic and the identification of suspect cars or vehicles. Deep learning techniques have emerged as a powerful strategy offering an alternative solution to human vision imitation. These techniques are based on deep learning.

This thesis falls within the domain of computer vision. Our goal is to detect vehicles in a stream of images and classify them according to their type 'bus, car, motorcycle, truck, bicycle'. For this purpose, we adopted the two models YOLOv5 and Faster RCNN for vehicle detection. After a comparative study between these models, the best results were obtained by the Faster RCNN method with an accuracy value of 89%.

Keywords: Vehicle detection, neural network, deep learning, CNN, 'ResNet-50', 'YOLO', 'Faster R-CNN'.