

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
UNIVERSITÉ SAAD DAHLEB – BLIDA 1

Faculté de science  
Département d'informatique



Mémoire de Projet de Fin d'Études  
Pour l'obtention du diplôme de Master en Informatique

# Reconnaissance Automatique des Marques de Véhicules et leurs Plaques d'Immatriculation

**Présenté par**

Rania BOUTRIF

Safaa BENZEGHIOUA

**Proposé par**

Djamel BOUCHAFFRA, Directeur de Recherche, CDTA, Alger.

Fayçal YKHLEF, Maître de Recherche A, CDTA, Alger.

**Soutenu le :** 25 Juin 2023.

**Devant le jury composé de**

KAMECHE Abdallah Hicham, Université de Blida 1, Blida.

GUESSOUM Dalila, Université de Blida 1, Blida.

YKHLEF Fayçal, CDTA, Alger.

YKHLEF Hadjer, Université de Blida 1, Blida.

**Président**

**Examinatrice**

**Encadreur**

**Co-promoteur**

**Année Universitaire 2022-2023**

## Résumé

Notre projet vise à développer un système avancé de reconnaissance des plaques d'immatriculation des voitures ainsi que de leurs marques, dans le but d'automatiser l'accès aux parkings des institutions et d'augmenter le niveau de sécurité. Nous utiliserons des techniques avancées d'apprentissage profond pour atteindre cet objectif. Lorsqu'une voiture franchit la boucle de courant installée à l'entrée du parking, la caméra du poste de contrôle se déclenche automatiquement pour capturer une image du devant du véhicule. La reconnaissance des plaques s'effectue en deux phases : (i) la localisation de la plaque et (ii) la reconnaissance des chiffres. Pour la première phase, nous utilisons Yolo.v8. En ce qui concerne la deuxième phase, nous avons mis en œuvre trois modèles différents : (i) EasyOCR, (ii) Tesseract et (iii) un modèle de reconnaissance basé sur l'architecture AlexNet. La reconnaissance des logos est basée sur notre architecture CNN personnalisée. Nous nous sommes focalisés sur les matricules Algériens dans notre étude. L'évaluation de notre système sur une base de données de plaques d'immatriculation algériennes révèle que le modèle AlexNet a atteint un taux de reconnaissance correcte de 99,31%. En revanche, la reconnaissance des marques de véhicules a obtenu un taux de reconnaissance correcte de 97,05%. Les résultats obtenus démontrent que dans des conditions favorables telles qu'un éclairage adéquat et des angles appropriés, le système parvient à détecter et à identifier avec succès le numéro de la plaque d'immatriculation ainsi que la marque du véhicule. De plus, le système affiche des informations pertinentes sur le véhicule, y compris les informations personnelles du conducteur.

**Mot clé :** Vision par ordinateur, Localisation, Segmentation, Reconnaissance, Classification, CNN, Plaque d'immatriculation, OpenCV.

## **Abstract**

Our project aims to develop an advanced system for recognition of car license plates and their brands, in order to automate access to institutional parking lots and enhance security levels. We will utilize advanced deep learning techniques to achieve this objective. When a car passes through current loop installed at the entrance to the parking lot, the control post camera automatically triggers to capture an image of the front of the vehicle. The license Plate recognition is conducted in two phases: (i) plate localization and (ii) character recognition. For the first phase, we employ Yolo.v8. As the second phase, we have implemented three different models: (i) EasyOCR, (ii) Tesseract and (iii) a recognition model based on the AlexNet architecture. Logo recognition is based on our custom CNN architecture. We focused on Algerian license plates in our study. Evaluation of our system on a database of Algerian license plates reveals that AlexNet model achieved correct recognition rates of 99,31%. On the other hand, brand recognition attained a correct recognition rate of 97,05% The obtained results demonstrate that under favorable conditions such as proper lighting and suitable angles, the system successfully detects and identifies the license plate number as well as the vehicle brand. Moreover, the system displays relevant information about the vehicle, including the driver's personal information.

**Keywords:** Computer vision, Localization, Segmentation, Recognition, Classification, CNN, License plates, Open CV.

## ملخص

مشروعنا يهدف إلى تطوير نظام متقدم للتعرف على لوحات ترخيص السيارات وعلاماتها التجارية، من أجل الوصول إلى مواقف السيارات المؤسسية وتعزيز مستويات الأمان. سوف نستخدم تقنيات التعلم العميق المتقدمة لتحقيق هذا الهدف. عندما تمر سيارة عبر الحلقة الحالية المثبتة عند مدخل ساحة الانتظار، يتم تشغيل كاميرا مركز التحكم تلقائيًا لالتقاط صورة لمقدمة السيارة. يتم إجراء التعرف على لوحة الترخيص على مرحلتين: (1) توطين اللوحة و (2) التعرف على الأحرف. في المرحلة الأولى، نستخدم Yolo.v8 كمرحلة ثانية، قمنا بتنفيذ ثلاثة نماذج مختلفة، EasyOCR (1) : Tesseract (2) و (3) نموذج التعرف على أساس بنية AlexNet. يعتمد التعرف على الشعار على بنية CNN المخصصة لدينا. ركزنا في دراستنا على لوحات الترخيص الجزائرية. يكشف تقييم نظامنا على قاعدة بيانات اللوحات الترخيص الجزائرية أن نموذج AlexNet حقق معدلات التعرف الصحيحة بنسبة 99,31% من ناحية أخرى، حقق التعرف على العلامة التجارية معدل التعرف الصحيح بنسبة 97,05%. توضح النتائج التي تم الحصول عليها أنه في ظل ظروف مواتية مثل الإضاءة المناسبة والزوايا المناسبة، يكتشف النظام بنجاح ويحدد رقم لوحة الترخيص بالإضافة إلى العلامة التجارية للمركبة. علاوة على ذلك، يعرض النظام المعلومات ذات الصلة بالمركبة، بما في ذلك المعلومات الشخصية للسائق.

**الكلمات المفتاحية:** رؤية حاسوبية، توطين، تجزئة، تمييز، تصنيف، CNN ، لوحات ترخيص، سيرة ذاتية مفتوحة.

## REMERCIEMENTS

---

Nous exprimons notre gratitude à Allah le tout puissant pour nous avoir accordé la santé et la volonté de mener à bien ce projet, El Hamdoullillah !

Ce projet a pu être mené à terme grâce à l'assistance de nombreuses personnes à qui nous adressons nos sincères remerciements.

Tout d'abord, nous remercions chaleureusement notre promoteur, le professeur Djamel Bouchaffra, directeur de la division Architecture des Systèmes et Multimédia, pour son orientation, sa confiance en nous et son expertise dans le domaine. Grâce à ses encouragements et ses critiques constructives, nous avons pu repousser nos limites et trouver des solutions novatrices.

Nous tenons à exprimer notre gratitude envers notre promoteur, le Dr. Fayçal Ykhlef, maître de recherche au CDTA, pour sa disponibilité et ses précieux conseils qui ont grandement contribué à nos avancées. Nous le remercions également pour sa patience et son soutien constant.

Un grand merci également à notre Co-promoteur, le Dr. Hadjer Ykhlef, enseignante à l'université Saad Dahleb 1, pour sa disponibilité, sa précieuse contribution, Ses conseils et son expertise.

Nous souhaitons également remercier l'ensemble du personnel et des chercheurs du Centre de Développement des Technologies Avancées (CDTA), en particulier ceux de la division Architecture des Systèmes et Multimédia (ASM), pour leur accueil chaleureux et leur soutien, notamment Dr. Allam Abdelkarim.

Nos remerciements vont également aux membres du jury qui ont accepté d'évaluer notre travail et de l'enrichir de leurs connaissances et suggestions.

Nous tenons à exprimer notre reconnaissance envers le corps professoral et administratif de l'université Saad Dahleb 1, qui a contribué à notre réussite universitaire.

Nous remercions également tous les techniciens de Pravda pour leur précieuse aide.

Un grand merci aux clubs ITC et CSCC pour leur soutien.

Nous tenons à remercier tous ceux qui ont contribué, de près ou de loin, à la réalisation de ce projet.

Merci du fond du cœur.

## **DEDICACE**

---

*Je dédie ce modeste travail à ceux qui m'ont donné la vie et on fait de moi ce que je suis  
aujourd'hui mes chers parents*

*A ma lumière, mon idole celle qui a veillé à mes côtés depuis mon jeune âge jusqu'à  
aujourd'hui tu n'es pas seulement une mère pour moi mais une meilleure amie, une sœur tout  
simplement mon tout sache que tous les mots du monde ne suffisent pas pour décrire à que  
point je t'aime*

*Au roi de mon cœur, mon seul et unique amour, mon père l'épaule sur qui je peux compter tu  
as su me guider tout au long de mon parcours et a su me soutenir même quand on n'était pas  
du même avis, ta petite chouchoute qui t'aime plus que tout au monde*

*A la prune de mes yeux, ma confidente, ma moitié, ma vie n'aurait aucun sens sans toi tu  
as fait de ma vie un paradis et même si tu es la grande sœur pour moi tu resteras toujours ma  
protégée, à toutes nos nuits blanches passé ensemble, je t'aime au-delà des mots*

*A mon petit frère chéri dont je suis fière sache que tu es le frère que toute les filles rêve  
d'avoir, t'es la perfection masculine, sache que tout ce que je fais c'est pour ton bien, je  
t'aime énormément*

*Une dédicace spéciale s'adresse à mes grands-parents qui sont un exemple de sagesse pour  
moi, à mon cher grand père papa Tahar notre héros toi qui aimes tant partager ta culture et  
l'histoire avec moi merci pour ton savoir que tu me transmets de jour en jour, à ma maman  
Leila le pilier de la famille celle qui donne tout pour nous rendre heureux, merci pour tes  
petits plats avec lesquels on a grandi, à ma mamy chérie à la fois cool et sévère tes conseils  
avisés et tes remarques font de toi une grand-mère particulière, à mon défunt grand père, je  
vous aime infiniment*

*A ma chère tante Aicha que j'aime, à mon oncle Fahmy et sa femme, à mes oncles maternels  
et leurs femmes, à mes tontons et tatas.*

*A mes cousins Kamel, Islem, Fafido et moncef ; à mes cousines Sabrina, Ibtissem, Wissam,  
Lilia, Yasmine, Meriem et Lina.*

*A mes sœurs de cœurs Khadidja et Feriel et à mes chères copines Maroua Li, Imene Issolah,  
Khadidja, Wafa*

**Rania**

## DEDICACE

---

وَمَا تَوْفِيقِي إِلَّا بِاللَّهِ عَلَيْهِ تَوَكَّلْتُ وَإِلَيْهِ أُنِيبُ

*What an incredible journey it has been, spanning across all those years. There were nights devoid of sleep, days filled with stress, moments of laughter, instances of tears, and periods of pure joy. Throughout this rollercoaster ride, I have witnessed both highs and lows. However, what truly matters is the fact that I have arrived at this moment without a single ounce of regret.*

*my beloved father, I owe him a great deal of gratitude he stood by my side both emotionally and physically. I cannot quantify the number of days he accompanied me to the university, providing encouragement during exams and comforting me during stressful moments.*

*My lovely mother, dedicated countless days and nights to praying for my success. She tirelessly wiped away my tears and was always there for me whenever I needed her, just like a guardian angel she is the source of motivation.*

*I also want to express my heartfelt appreciation to my brothers, Imad, Rafiq, and Alaa. Undoubtedly, I caused them stress throughout this journey. Yet, despite everything, my brothers never grew weary of me. They managed to bring laughter into even the darkest of times.*

*Today, I know that my family takes immense pride in witnessing my growth, akin to a flower blossoming like my parents always say. Graduating after years of unwavering dedication and hard work is a testament to their unwavering support and my own perseverance. Lastly, I would like to extend my gratitude to myself. I acknowledge and appreciate my unwavering belief in my abilities and my refusal to take a single day off. I am immensely proud of the person I have become. It is important to understand that progress sometimes involves taking steps forward and backward, but never stopping. I will continue walking until I reach my ultimate goals.*

**Safaa.**

## **LISTE DES ACRONYMES ET ABREVIATIONS**

---

ACC : *Accuracy*

ALPR : *Automatic License Plate Recognition (Reconnaissance automatique de plaque d'immatriculation)*

BD : *Base de données*

CNN : *convolutional neural network (Réseau de neurones convolutifs)*

FN: *False Negative (Faux Négatif)*

FP: *False Positive (Faux positif)*

IA : *Intelligence Artificielle*

OCR : *(Optical Character Recognition) Reconnaissance Optique de Caractère*

PI : *Plaque d'immatriculation*

TN: *True Negative (Vrai Négatif)*

TP: *True Positive (Vrai Positif)*

YOLO: *You Only Look Once.*



# TABLE DES MATIERES

---

<b>1</b>	<b>CHAPITRE 1 : INTRODUCTION .....</b>	<b>1</b>
1.1	Motivations.....	1
1.2	Contributions.....	2
1.3	Impact du projet .....	2
1.4	Organisation du mémoire .....	3
<b>2</b>	<b>CHAPITRE 2 : SECURISATION DES ACCES AUX PARCS DE STATIONNEMENT .....</b>	<b>4</b>
2.1	Introduction .....	4
2.2	Système de contrôle d'accès aux parcs de stationnement.....	4
2.3	Méthodes de détection, segmentation et reconnaissance des caractéristiques des véhicules.....	6
2.4	Reconnaissance optique de caractères.....	13
2.5	Métriques d'évaluation.....	14
2.6	Revue scientifique de la reconnaissance des plaques d'immatriculation.....	15
2.7	Revue scientifique de la reconnaissance de logos.....	16
2.8	Conclusion.....	16
<b>3</b>	<b>CHAPITRE 03 : CONCEPTION D'UN SYSTEME DE RECONNAISSANCE DES PLAQUES D'IMMATRICULATION ET DES LOGOS</b>	<b>17</b>
3.1	Introduction .....	17
3.2	Conception d'un système de reconnaissance des plaques d'immatriculation.....	17
3.3	Conception d'un système de reconnaissance de logos.....	23
3.4	Analyse des besoins .....	25
3.5	Conclusion.....	27
<b>4</b>	<b>CHAPITRE 04: RESULTATS ET INTERPRETATION .....</b>	<b>28</b>
4.1	Introduction .....	28
4.2	Outils de développement.....	28
4.3	Environnement matériel .....	29
4.4	Environnement de conception.....	29
4.5	Bibliothèque .....	30
4.6	Implémentation du système de reconnaissance de plaques d'immatriculation.....	30
4.7	Implémentation de système de reconnaissance de logos .....	37
4.8	Présentation de l'application .....	44
4.9	Conclusion.....	49
<b>5</b>	<b>CHAPITRE 05: CONCLUSION .....</b>	<b>50</b>
5.1	Conclusion.....	50
5.2	Travaux futur.....	51

## LISTE DES FIGURES

---

<b>Figure 2.1:</b> Système de contrôle d'accès automatique. ....	5
<b>Figure 2.2:</b> Image en noir et blanc. ....	7
<b>Figure 2.3:</b> Image a niveau de gris. ....	8
<b>Figure 2.4:</b> Image en couleur. ....	8
<b>Figure 2.5:</b> Architecture AlexNet [18]. ....	13
<b>Figure 2.6:</b> Matrice de confusion [23]. ....	14
<b>Figure 3.1:</b> Schéma de conception ALPR. ....	17
<b>Figure 3.2:</b> Caractéristiques du matricule Algérien. ....	18
<b>Figure 3.3:</b> Echantillons de la BD Algerian Dataset. ....	19
<b>Figure 3.4:</b> Echantillons de la BD du CDTA. ....	19
<b>Figure 3.5:</b> Echantillons aléatoires de l'ensemble de données Chars74 dataset. ....	22
<b>Figure 3.6:</b> Conception de système de reconnaissance de logos. ....	24
<b>Figure 3.7:</b> Diagramme de cas d'utilisation de notre système globale. ....	26
<b>Figure 3.8:</b> Présentation du diagramme de séquence pour la vérification d'accès automatiques de véhicule. ....	27
<b>Figure 4.1:</b> Les deux Bases de données Fusionner et labéliser. ....	31
<b>Figure 4.2:</b> Résultats de détection de la PI dans une image. ....	32
<b>Figure 4.3:</b> Résultats de l'extraction des plaques. ....	32
<b>Figure 4.4 :</b> Graphes des métriques d'évaluation de la détection de PI par yolo8. ....	33
<b>Figure 4.5 :</b> Tache "blobs" des caractères. ....	34
<b>Figure 4.6 :</b> Segmentation des caractères avec des cadres délimitateurs (Bounding boxes) ..	35
<b>Figure 4.7 :</b> Evaluation d'Accuracy et de perte durant l'entraînement du modèle d'évaluation. ....	35
<b>Figure 4.8 :</b> Précision et rappel obtenue pour chaque chiffre. ....	36
<b>Figure 4.9 :</b> Matrice de confusion pour le modèle de reconnaissance de chiffre. ....	36
<b>Figure 4.10 :</b> Images défectueuse supprimer de la base de données de logos. ....	37
<b>Figure 4.11 :</b> Base de données de logos avant et après le rééquilibrage. ....	38
<b>Figure 4.12 :</b> Détection de logo dans une image. ....	38
<b>Figure 4.13 :</b> Extraction du logo à partir d'une image. ....	39
<b>Figure 4.14 :</b> graphes des métriques d'évaluation de detection de logos. ....	39
<b>Figure 4.15 :</b> Accuracy de l'entraînement et de la validation pour la classification de logos durant l'entraînement du CNN. ....	40
<b>Figure 4.16 :</b> Loss de l'ensemble d'entraînement et de la validation pour la reconnaissance de logos durant l'entraînement du CNN. ....	40
<b>Figure 4.17 :</b> Matrice de confusion de classification de logo partie 1. ....	43
<b>Figure 4.18 :</b> Page d'accueil. ....	45
<b>Figure 4.19 :</b> page de la base de données des employé. ....	45
<b>Figure 4.20 :</b> Page d'enregistrement d'une nouvelle personne. ....	46
<b>Figure 4.21 :</b> Page de la base de données des employés. ....	46
<b>Figure 4.22 :</b> Page d'enregistrement d'un nouvel véhicule. ....	47
<b>Figure 4.22 :</b> page de reconnaissance de PI et logo. ....	48

**Figure 4.23** : Interface de reconnaissance avec l'accès autorisé..... 49

## LISTE DES TABLEAUX

---

<b>Tableau 2-1</b> : Avantages et inconvénient des parc conventionnels et intelligents.....	6
<b>Tableau 2-2</b> : Comparaison de la vitesse de détection et de la détection multiple entre SSD R-CNN et YOLO. ....	11
<b>Tableau 3-1</b> : Couches convolutionnelle de l'architecture AlexNet.....	23
<b>Tableau 3-2</b> : Couches entièrement connectés de l'architecture AlexNet. ....	23
<b>Tableau 3-3</b> : Couches convolutionnelle de l'architecture ce reconnaissance de logo.....	25
<b>Tableau 3-4</b> : Couches entièrement connecté de l'architecture de reconnaissance de logo. .	25
<b>Tableau 4-1</b> : Ressources de GoogleColab. ....	28
<b>Tableau 4-2</b> : Ressources locales des ordinateurs utilisé. ....	29
<b>Tableau 4-3</b> : Bibliothèques Utilisé et leur version. ....	30
<b>Tableau 4-4</b> : : Matrice de confusion pour la détection de plaque d'immatriculations. ....	33
<b>Tableau 4-5</b> : F1Score pour la reconnaissance de chiffre. ....	37
<b>Tableau 4-6</b> : : Matrice de confusion pour la détection de Logos .....	39
<b>Tableau 4-7</b> : Comparaison entre les résultats d'Accuracy de notre modèle et le modèle. ...	40
<b>Tableau 4-8</b> : Précision, rappel et F1Score des marques de véhicule.....	41

---

<b>Tableau 2-1</b> : Avantages et inconvénient des parc conventionnels et intelligents.....	6
<b>Tableau 2-2</b> : Comparaison de la vitesse de détection et de la détection multiple entre SSD R-CNN et YOLO. ....	11
<b>Tableau 3-1</b> : Couches convolutionnelle de l'architecture AlexNet.....	23
<b>Tableau 3-2</b> : Couches entièrement connectés de l'architecture AlexNet. ....	23
<b>Tableau 3-3</b> : Couches convolutionnelle de l'architecture ce reconnaissance de logo.....	25
<b>Tableau 3-4</b> : Couches entièrement connecté de l'architecture de reconnaissance de logo. .	25
<b>Tableau 4-1</b> : Ressources de GoogleColab. ....	28
<b>Tableau 4-2</b> : Ressources locales des ordinateurs utilisé. ....	29
<b>Tableau 4-3</b> : Bibliothèques Utilisé et leur version. ....	30
<b>Tableau 4-4</b> : : Matrice de confusion pour la détection de plaque d'immatriculations. ....	33
<b>Tableau 4-5</b> : F1Score pour la reconnaissance de chiffre. ....	37
<b>Tableau 4-6</b> : : Matrice de confusion pour la détection de Logos .....	39
<b>Tableau 4-7</b> : Comparaison entre les résultats d'Accuracy de notre modèle et le modèle. ...	40
<b>Tableau 4-8</b> : Précision, rappel et F1Score des marques de véhicule.....	41

# Chapitre 1 : Introduction

## 1.1 Motivations

Depuis la nette augmentation de vente automobile, l'identification des véhicules est devenue un besoin fondamental qui facilite la tâche de plusieurs applications telles que la surveillance du Traffic, le suivie de voitures volées, la gestion de péage de stationnement, l'application de violation de feu rouge, et l'entrée aux parkings des entreprises.

Le développement de l'intelligence artificielle (IA), particulièrement la vision par ordinateur, a permis de construire les systèmes de reconnaissance automatique des plaques d'immatriculation connus sous le nom de "Automatic License Plate Recognition" (ALPR). Ces derniers facilitent la tâche de l'identification des véhicules. Cependant, avec la notoriété croissante des systèmes ALPR, la falsification des plaques d'immatriculation (PI) est devenue une pratique des personnes malveillantes. Par conséquent, l'identification des véhicules uniquement par leurs PI s'est révélée insuffisante, rendant nécessaire l'utilisation d'autres caractéristiques moins connues, telles que la reconnaissance de logo, la couleur des véhicules, la reconnaissance du visage du conducteur, et l'identification des codes d'accès pour les véhicules autorisés. La combinaison de ces caractéristiques renforce la sécurité, lutte contre la fraude et permet d'identifier rapidement une plaque d'immatriculation falsifiée. Pour mettre en place un tel système robuste, la détection et la reconnaissance de ces caractéristiques sont des tâches cruciales.

Plusieurs travaux de recherche ont été proposés dans la littérature concernant la reconnaissance des plaques d'immatriculation . En revanche, la reconnaissance des caractéristiques mentionnées ci-dessus, telle que la reconnaissance de logos, est un domaine de recherche récent et peu d'études ont été publiées à ce sujet. La combinaison de ces technologies en vue de sécuriser l'accès aux parcs de stationnement n'a pas encore été largement explorée. A nos connaissances, aucun système combinant ces technologies n'a été déployé en Algérie jusqu'à présent.

En conséquence, la mise en place d'un logiciel de reconnaissance de plaque d'immatriculation et de logo pourrait donc répondre à un besoin urgent en matière de sécurité et de lutte contre la fraude. Un tel système permettrait de détecter rapidement les véhicules, de vérifier leurs légitimités et d'automatiser les processus d'accès et de contrôle des accès dans diverses installations, améliorant ainsi la sécurité et l'efficacité globale.

## 1.2 Contributions

Dans le cadre de ce projet, nous nous intéressons à la reconnaissance de deux caractéristiques essentielles pour l'identification de véhicules soit la plaque d'immatriculation et la marque (logo). Notre contribution consiste à concevoir un logiciel complet qui permet l'accès aux parkings sans l'intervention d'un agent de contrôle. Pour cela, ce travail est divisé en plusieurs tâches. La première concerne la détection du passage d'un véhicule sur la boucle de courant ; celle-ci actionne la caméra qui prend une image. Une fois l'image est acquise, la détection et la reconnaissance du matricule et du logo sont effectués. La troisième action consiste à comparer la paire de matricule et logo reconnus avec celles des employés enregistrés sur la base de données (BD). Enfin, si le véhicule est celui d'un employé autorisé à accéder au parc, notre logiciel ouvre la barrière et la referme dès que le véhicule passe.

La méthode de détection que nous avons implémentée est basée sur l'architecture de YOLO (You Only Look Once) et plus précisément de la version 8. Quant à la reconnaissance optique de caractères connue en anglais sous le nom *optical character recognition* (OCR), nous avons implémenté 3 modèles : (i) EasyOCR, (ii) Tesseract et (iii) un modèle basé sur les réseaux de neurones convolutifs connus sous le nom *convolutional neural network* (CNN). Ce dernier passe par deux principales étapes : (i) la segmentation par détection de contours et (ii) la reconnaissance de caractères à base d'AlexNet.

D'autres part pour la reconnaissance de logos, nous avons choisi de concevoir notre propre architecture CNN.

## 1.3 Impact du projet

Le travail réalisé dans le cadre de ce projet de fin d'études a un impact socio-économique significatif pour notre pays. En effet, la plupart des systèmes de contrôle d'accès aux parkings ne sont pas automatisés et nécessitent la présence d'un agent. Pour assurer une surveillance 24h/24, plusieurs agents doivent être embauchés, ce qui entraîne des coûts de gestion élevés pour les parkings. Notre système permet de réaliser des économies considérables sur ces dépenses.

La plupart des systèmes de contrôle d'accès automatiques de véhicules se concentrent sur la reconnaissance de la plaque d'immatriculation. Nous avons ajouté à cela la reconnaissance du logo pour rendre le système plus fiable et robuste. Celui-ci peut être utile pour plusieurs institutions tel que les entreprises, les banques, les ministères et les parkings d'immeubles privés.

## 1.4 Organisation du mémoire

La structure de notre mémoire se compose de cinq chapitres, qui sont définis de la manière suivante :

Le premier chapitre est une introduction qui établit le contexte de notre étude, expose la problématique à résoudre, définit les objectifs que nous visons à travers cette étude, ainsi que son impact.

Le deuxième chapitre, intitulé "Sécurisation des accès aux parcs de stationnement", explore les systèmes de contrôle d'accès et les caractéristiques d'identification automatique des véhicules. Ce chapitre met particulièrement l'accent sur le rôle des CNN dans l'identification et la reconnaissance des caractéristiques des véhicules. De plus, il présente une revue de la littérature pertinente concernant les systèmes ALPR et les systèmes de reconnaissance de logos.

Le troisième chapitre présente la conception de l'architecture du système ALPR et de classification des logos. Il décrit les méthodologies utilisées pour la mise en œuvre des deux systèmes, ainsi que les diagrammes UML utilisés.

Le quatrième chapitre illustre l'implémentation du système ALPR et de reconnaissance des logos. Il décrit les outils utilisés pour l'implémentation, les étapes suivies, ainsi que les résultats obtenus pour les deux systèmes.

Le cinquième chapitre comprend une conclusion générale et aborde les perspectives futures de cette étude.

## **Chapitre 2 : Sécurisation des accès aux parcs de stationnement**

### **2.1 Introduction**

Dans ce chapitre, nous présentons les divers systèmes de contrôle d'accès des véhicules. Nous exposons les avantages et les inconvénients de ces systèmes. Ensuite, nous abordons les différentes caractéristiques utilisées pour identifier un véhicule. Dans notre étude, nous nous concentrons sur la reconnaissance de la PI et des logos. Nous décrivons également les différentes techniques de détection et de reconnaissance, puis nous présentons quelques revues de la littérature sur l'ALPR et la reconnaissance de logos.

### **2.2 Système de contrôle d'accès aux parcs de stationnement**

Le contrôle d'accès est une mesure de sécurité visant à restreindre l'accès à un environnement donné, permettant uniquement aux personnes autorisées d'y entrer, de le consulter ou de l'utiliser. Les systèmes de contrôle d'accès sont présents dans divers éléments tels que les portes, les serrures à clé, les clôtures, les systèmes biométriques, les détecteurs de mouvement, les systèmes de badges, et d'autres dispositifs de sécurité [1]. Dans notre contexte, nous nous concentrons sur les systèmes de contrôle d'accès des véhicules aux parcs de stationnement. Ces systèmes assurent le contrôle de l'entrée des véhicules dans des espaces privés tels que les parkings, les garages et d'autres sites similaires. Actuellement, la plupart des systèmes installés reposent sur l'utilisation de barrières actionnées manuellement [2]. Cependant, de nos jours, un nouveau système de contrôle d'accès gagne en popularité : l'accès automatique sans intervention d'un agent.

#### **2.2.1 Parcs de stationnement conventionnels**

Dans cette catégorie, les entreprises choisissent souvent d'avoir un gardien ou un agent de surveillance pour assurer l'accueil et le contrôle des flux de véhicules entrants et sortant de leur parking. Leur rôle principal consiste à garantir la sécurité des lieux et à prévenir toute activité malveillante. De cette manière, ils comblent les éventuelles lacunes en matière de sécurité et renforcent la présence des caméras de surveillance [3].

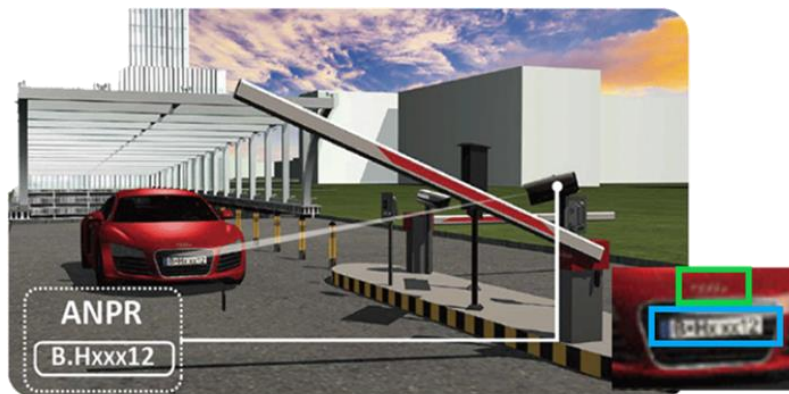


### 2.2.2 Parcs de stationnement intelligents

Le système de contrôle d'accès automatique des véhicules permet de gérer l'entrée des véhicules dans des espaces privés tels que les parkings, les garages et autres lieux, sans nécessiter l'intervention d'un agent. L'une des techniques les plus répandues pour ce contrôle d'accès des véhicules est l'ALPR [4]. Cette technologie permet de lire les PI à des distances de plus de 10 mètres, offrant ainsi une méthode pratique et fiable pour le contrôle d'accès automobile. Cependant, après une période de stagnation, de nouvelles avancées technologiques, telles que l'identification du conducteur ou la reconnaissance de certaines caractéristiques spécifiques du véhicule, émergent et nous poussent à réévaluer nos approches en matière d'identification des véhicules, à la fois pour leur praticité et leur sécurité [5].

Pour identifier un véhicule, on utilise différentes caractéristiques, la plus fiable et la plus couramment utilisée dans les systèmes d'accès est la PI. En effet, la PI est un numéro unique qui est associé à un seul véhicule dans le monde entier. Cependant, afin de confirmer que la PI n'est pas erronée, on peut également prendre en compte d'autres caractéristiques telles que le logo, la marque et la couleur du véhicule. En combinant ces informations, on renforce la précision de l'identification du véhicule [6].

La Figure 2.1 illustre un système de contrôle d'accès automatique basé sur l'ALPR et la reconnaissance de logo.



**Figure 2.1:** Système de contrôle d'accès automatique.

### 2.2.3 Comparaison des systèmes d'accès

Nous avons résumé les principaux avantages et inconvénients entre un système d'accès au parc de stationnement conventionnels et intelligent (voir Tableau 2-1).

**Tableau 2-1** : Avantages et inconvénient des parc conventionnels et intelligents.

Système	Avantages	Inconvénients
Contrôle d'accès conventionnelles	<ul style="list-style-type: none"> <li>- Ne commet pas d'erreur</li> <li>- Continue de fonctionner dans les cas de coupure de courant</li> </ul>	<ul style="list-style-type: none"> <li>- Nécessite plusieurs gardiennes pour une surveillance de 24h</li> <li>- Perte d'argent (embauche de plusieurs agents)</li> <li>- Un agent peut s'endormir sans se rendre compte</li> </ul>
Contrôle d'accès automatique	<ul style="list-style-type: none"> <li>- Economie d'argent</li> <li>- Processus de détection et reconnaissance rapide</li> <li>- Possibilité d'ajouter, supprimer et mettre à jour n'importe quel plaque</li> </ul>	<ul style="list-style-type: none"> <li>- Peut commettre des erreurs</li> <li>- Ne fonctionne pas en cas de panne de courant</li> </ul>

### 2.3 Méthodes de détection, segmentation et reconnaissance des caractéristiques des véhicules

Pour identifier des caractéristiques du véhicule telles que le logo et la PI, il est nécessaire d'utiliser des techniques de détection et de reconnaissance, qui sont des domaines clés de l'IA et de la vision par ordinateur. La détection d'objets consiste à localiser et à identifier la présence d'objets spécifiques dans une image ou une vidéo. Cela implique de trouver les emplacements des objets d'intérêt et de créer des cadres englobants autour d'eux. Cette étape permet de repérer les objets dans une scène et de fournir des informations sur leur position et leur taille. La reconnaissance d'objets va plus loin en identifiant précisément les objets détectés. Cela nécessite d'associer les objets à des catégories ou à des classes spécifiques, comme des chiffres, des marques, et des animaux. L'utilisation de techniques d'apprentissage automatique, en particulier de réseaux de neurones profonds tels que les CNN, a considérablement amélioré la capacité des systèmes de détection et de reconnaissance d'objets.

#### 2.3.1 Méthode de détection d'objet à base de la vision par ordinateur

La vision par ordinateur, également connue sous le nom de computer vision, est une technologie d'IA qui permet aux ordinateurs de "voir" de manière similaire aux êtres humains. Le concept de vision par ordinateur vise à enseigner aux ordinateurs à traiter les images au niveau des pixels et à les comprendre. Techniquement, les machines tentent d'extraire des

informations visuelles, de les traiter et d'interpréter les résultats à l'aide d'algorithmes logiciels spécifiques [7].

#### **a- Traitement d'image**

Le traitement d'images est une discipline de l'informatique et des mathématiques appliquées qui étudie les images numériques et leurs transformations, dans le but d'améliorer leur qualité ou d'en extraire de l'information [3]. Dans notre cas nous parlons de détection et segmentation d'objet dans une image.

#### **b- Codage de couleur**

Le codage des couleurs est une manière de représenter les couleurs numériquement, généralement en utilisant des nombres entiers ou des octets (valeurs de 0 à 255). Il existe plusieurs espaces de couleurs et systèmes de codage des couleurs, mais les plus utilisés sont [8]:

##### **- Image noir et blanc (binaire)**

Une image binaire est une image dans laquelle chaque pixel ne peut avoir que deux valeurs possibles : Noir (valeur 0) ou Blanc (valeur 1) (Voir Figure 2.2) [8].



**Figure 2.2:**Image en noir et blanc.

##### **- Image a niveaux de gris (grayscale image)**

La couleur du pixel d'image a niveau de gris peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires donc les valeurs comprises entre 0 et 255 [8] (voir Figure 2.3).



**Figure 2.3:**Image a niveau de gris.

#### - **Image en couleurs**

Une image en couleurs est une image numérique qui contient des informations de couleur pour chaque pixel. Les images en couleurs peuvent être créées en utilisant différents modèles de couleurs [8].

Les images en couleurs peuvent également être converties en images en niveaux de gris ou en images binaires pour des applications spécifiques telles que la reconnaissance de caractères, la détection de contours ou la segmentation d'objets (voir Figure 2.4).



**Figure 2.4:**Image en couleur.

### **2.3.2 Réseau de neurones Convolutifs**

Les CNN sont des outils largement utilisés pour l'apprentissage profond. Ils sont particulièrement adaptés à l'analyse d'images en tant qu'entrées, bien qu'ils puissent également être utilisés pour d'autres applications telles que le traitement du texte, des signaux et d'autres types de données continues [9]. Leur architecture se compose de deux blocs principaux. Le premier bloc est la caractéristique distinctive de ce type de réseau de neurones, car il agit comme un extracteur de caractéristiques. Le second bloc, bien qu'il ne soit pas spécifique aux CNN, se trouve généralement à la fin de tous les réseaux de neurones utilisés pour la classification [10].

**a- Architecture**

L'architecture d'un CNN est constituée d'une série de blocs de traitement visant à extraire les caractéristiques permettant de différencier les différentes classes d'appartenance d'une image. Un bloc de traitement peut être composé d'une ou plusieurs couches [10] :

- **Couche de convolutions (CONV)** : Une couche de convolution est une pile de convolutions. En effet, l'image est parcourue par plusieurs filtres de convolution, ce qui génère plusieurs cartes de caractéristiques en sortie. Chaque filtre de convolution possède des paramètres spécifiques pour extraire des informations pertinentes de l'image. Le choix des paramètres du filtre de convolution dépend de la tâche à résoudre [10].

- **Couche de correction (fonction d'activation)** : L'application d'une fonction non linéaire aux cartes de caractéristiques en sortie de la couche de convolution permet de rendre les données non linéaires. Cela facilite l'extraction de caractéristiques complexes qui ne peuvent pas être modélisées par une simple combinaison linéaire d'un algorithme de régression [10].

- **Fonction d'activation ELU** : (Exponential Linear Unit) une fonction d'activation basée sur l'unité exponentielle, utilise les valeurs négatives grâce à la fonction exponentielle [11].

- **Fonction d'activation ReLu** : (Rectified Linear Unit) est appliquée à toutes les valeurs de la matrice de caractéristiques peu de temps après sa création. Cette fonction permet une formation rapide et précise des réseaux de neurones convolutifs. Le ReLU élimine toutes les valeurs négatives de la matrice, les remplaçant par des zéros, tandis que les valeurs positives restent inchangées [10].

- **Couche de pooling (POOL)** : L'étape de pooling est une technique de sous-échantillonnage couramment utilisée. Généralement, une couche de pooling est insérée de manière régulière entre les couches de convolution et de correction. En réduisant la taille des cartes de caractéristiques, cela permet de réduire le nombre de paramètres du réseau, accélérant ainsi le temps de calcul et réduisant le risque de surapprentissage (overfitting) [10].

- **Opération Flatten** : Le travail de cette opération consiste à convertir les matrices de caractéristiques en vecteurs qui peuvent être transmis à la couche entièrement connectée. Cette opération est souvent appelée aplatissement (flattening) car elle transforme les matrices de caractéristiques en une seule dimension [10].

- **Couche entièrement connecter** : Cette couche se situe à la fin du réseau et permet la classification de l'image en utilisant les caractéristiques extraites par les blocs de traitement

successifs. Elle est entièrement connectée, ce qui signifie que chaque neurone de cette couche est connecté à tous les neurones de la couche précédente. Ainsi, ils ont accès à l'ensemble des informations en entrée. Chaque neurone de cette couche attribue à l'image une valeur de probabilité correspondant à son appartenance à l'une des C classes possibles [12].

### **2.3.3 Modèles pour la détection d'objet**

Dans ce qui suit, nous présenterons les modèles les plus couramment utilisés et fiables pour l'identification, la détection et l'extraction automatique des PI et des logos. Enfin, nous procéderons à une comparaison de ces différents modèles afin de déterminer le meilleur parmi eux. Les modèles incluent :

#### **a- SSD (Single Shot MultiBox Detector)**

Le SSD, également connu sous le nom de "Décteur à boîtes multiples en un seul tir" en français, est une approche basée sur les réseaux neuronaux profonds pour la reconnaissance d'objets sur des images. Il a été proposé par Liu et al. en 2018 [13]. Le SSD divise l'espace de sortie des boîtes englobantes en un ensemble de boîtes par défaut avec différents ratios d'aspect et échelles pour chaque position sur la carte de caractéristiques. Lors de la prédiction, le réseau génère des scores pour la présence de chaque catégorie d'objet dans chaque boîte par défaut et effectue des ajustements pour mieux correspondre à la forme de l'objet. De plus, le réseau combine les prédictions provenant de plusieurs cartes de caractéristiques avec différentes résolutions pour gérer naturellement les objets de différentes tailles. Le SSD élimine complètement l'étape de génération de propositions ainsi que les étapes ultérieures de remise à l'échelle des pixels ou des caractéristiques, regroupant ainsi tous les calculs dans un seul réseau. Cela rend le SSD facile à entraîner et simple à intégrer dans des systèmes nécessitant une composante de détection [13].

#### **b- Faster R-CNN (Region based -CNN)**

En 2015, Girshick [14] a présenté l'architecture Faster R-CNN. Dans cette architecture, un réseau Faster R-CNN prend en entrée une image entière ainsi qu'un ensemble de propositions d'objets. Le réseau commence par traiter l'image entière à travers plusieurs couches de convolution (conv) et de max pooling, ce qui génère une carte de caractéristiques conv. Ensuite, pour chaque proposition d'objet, une couche de pooling de région d'intérêt (RoI) extrait un vecteur de caractéristiques de longueur fixe à partir de la carte de caractéristiques. Chaque vecteur de caractéristiques est ensuite propagé à travers une séquence de couches entièrement connectées qui se divisent finalement en deux couches de sortie similaires. La première couche

produit des estimations de probabilité softmax pour les K classes d'objets, incluant une classe "arrière-plan" générale. La deuxième couche produit quatre nombres réels pour chaque classe d'objet parmi les K. Chaque ensemble de quatre valeurs encode les positions affinées des boîtes englobantes pour l'une des K classes [14].

### c- YOLO (You Only Look Once)

En 2015, Redmon et al [15]. Ont présenté la première version de YOLO [16] est un réseau qui utilise des algorithmes d'apprentissage en profondeur (DL) pour la détection d'objets. Il est capable de classer certains objets dans une image et de déterminer leur emplacement précis. Par exemple, si une image d'un troupeau de moutons est introduite dans un réseau YOLO, celui-ci générera un vecteur de boîtes englobantes pour chaque mouton et les classera en conséquence. Le modèle YOLO est connu pour sa rapidité (45 images par seconde) et sa facilité d'optimisation, car il se base sur un seul réseau neuronal, à savoir le CNN, pour effectuer toutes les tâches nécessaires [15].

Au fil des années, plusieurs versions ultérieures de YOLO ont été publiées, telles que YOLO V2, V3, V4, V5, V6, V7 et V8. Il existe également quelques versions révisées plus limitées, comme YOLO-LITE.

### d- Comparaison entre SSD, R-CNN et YOLO

Dans le tableau suivant nous allons présenter les différences entre SSD, R-CNN et YOLO en termes de vitesse de détection, de précision et de détection de multiples objets. Nous avons attribué des classements numériques aux trois méthodes, où le chiffre 2 indique la performance optimale, le chiffre 1 indique une performance acceptable et le chiffre 0 représente une performance faible.

**Tableau 2-2 :** Comparaison de la vitesse de détection et de la détection multiple entre SSD R-CNN et YOLO.

Méthode	Vitesse de détection	Précision	Multi-objets (Détection multiple)
YOLO	2	1	2
SSD	1	0	1
Faster R-CNN	0	2	0

En conclusion, YOLO se distingue par sa vitesse de détection exceptionnelle et sa capacité à détecter plusieurs objets, ce qui en fait un choix privilégié pour les applications nécessitant

une détection en temps réel. Bien qu'il puisse avoir une précision légèrement inférieure à Faster R-CNN, il offre une gestion efficace des objets et des performances globales satisfaisantes.

### **2.3.4 Méthode de reconnaissance d'objet**

L'objectif de la reconnaissance d'objets est d'identifier et de classifier des objets spécifiques présents dans des images ou des vidéos. Cette tâche implique l'utilisation de techniques avancées de traitement d'images et d'apprentissage profond pour extraire des caractéristiques distinctives des objets et les associer à des catégories prédéfinies. Parmi les différentes approches utilisées, CNN joue un rôle essentiel. Il existe de nombreuses architectures de CNN dédiées à la reconnaissance d'objets, parmi lesquelles nous pouvons citer :

**VGGNet** : L'équipe VGG de l'Université d'Oxford a développé VGG (Visual Geometry Group), une architecture de réseau de neurones convolutifs. Ce qui distingue l'architecture VGG, c'est sa profondeur, avec des réseaux composés de 16 ou 19 couches. Elle est réputée pour sa simplicité et sa facilité d'interprétation [17].

**ResNet (Residual Network)**: Kaiming He et al [12]. Ont développé un réseau résiduel (ResNet). Cette approche utilise des connexions résiduelles pour résoudre le problème de la dégradation du gradient lors de l'entraînement de réseaux profonds.

**AlexNet** : L'architecture AlexNet a été l'un des premiers modèles de CNN profonds à connaître un immense succès dans le domaine de la vision par ordinateur. Elle a remporté avec succès le défi ImageNet Large Scale Visual Recognition Challenge en 2012. Conçue par une équipe dirigée par Geoffrey Hinton à l'Université de Toronto, comprenant notamment Alex Krizhevsky, l'architecte principal qui lui a donné son nom, ainsi que Ilya Sutskever, AlexNet a marqué un tournant majeur dans l'utilisation des réseaux neuronaux profonds [18].

AlexNet est constitué de plusieurs couches convolutives et de sous-échantillonnage, suivies de couches entièrement connectées (voir Figure 2.5). Il comprend au total huit couches, dont cinq couches convolutives et trois couches entièrement connectées. Cette architecture utilise des filtres de convolution de petite taille (3x3 et 5x5) avec des pas de convolution et du sous-échantillonnage pour extraire des caractéristiques à différentes échelles [18].



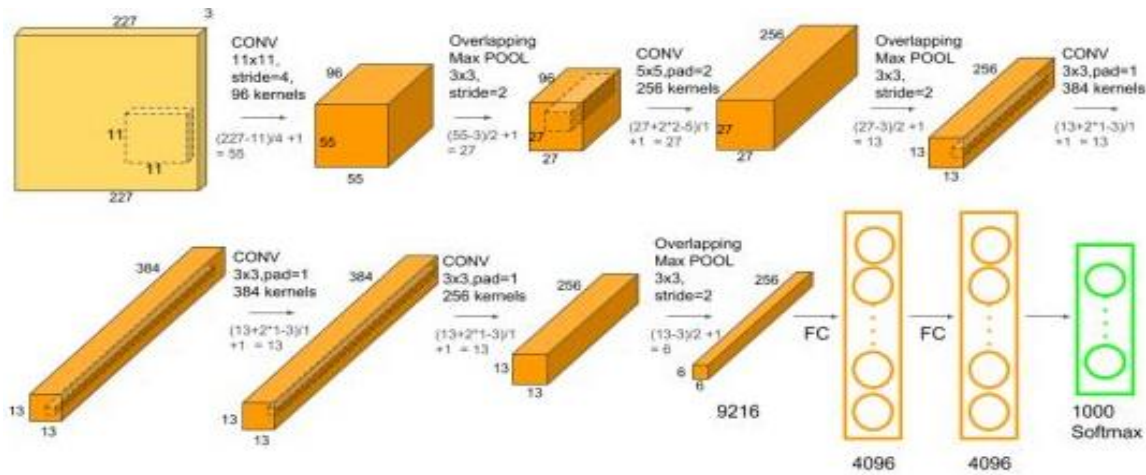


Figure 2.5: Architecture AlexNet [18].

## 2.4 Reconnaissance optique de caractères

La reconnaissance optique de caractères (Optical Character Recognition (OCR)) est une technologie permettant à un ordinateur de reconnaître des caractères spécifiques d'une langue en analysant une image. Le principe de cette technologie consiste à lire une image, détecter les formes présentes dans celle-ci, puis les comparer à une bibliothèque intégrée contenant des échantillons possibles pour chaque forme. Cette comparaison vise à faire correspondre chaque forme détectée à une forme de la bibliothèque afin de fournir le caractère alphanumérique correspondant. Les systèmes OCR avancés d'aujourd'hui sont devenus plus intelligents et sont capables de reconnaître pratiquement toutes les polices avec un haut niveau de précision, sans nécessiter d'apprentissage préalable [19] [20].

La reconnaissance optique des caractères implique généralement la capture d'une image du document, que ce soit par le biais d'un scan ou d'une photographie numérique. Cette image est ensuite convertie en une matrice de données que l'ordinateur peut interpréter. À l'aide d'algorithmes spécifiquement programmés, souvent utilisés dans le domaine de l'intelligence artificielle, l'ordinateur analyse les motifs présents dans l'image, qui correspondent aux caractères. Ensuite, le programme attribue des codes de caractères, généralement ASCII, qui correspondent aux caractères reconnus dans l'image d'origine. Dans la plupart des cas, les programmes d'OCR nécessitent un entraînement préalable pour améliorer leur capacité à reconnaître les caractères de manière plus précise [21].

Plusieurs logiciels OCR (Optical Character Recognition) sont bien connus et largement utilisés. Parmi ces logiciels, on peut citer Abbyy FineReader, Adobe Acrobat, Nuance OmniPage, Readiris, Amazon Textract, Tesseract et EasyOCR.

## 2.5 Métriques d'évaluation

Les métriques d'évaluation sont des mesures utilisées pour évaluer les performances et la qualité d'un modèle ou d'un système dans le domaine de l'apprentissage automatique et de l'intelligence artificielle. Elles permettent d'obtenir une évaluation objective de la précision, de la robustesse et de l'efficacité d'un modèle par rapport à un ensemble de données ou à un problème spécifique. Les métriques d'évaluation les plus couramment utilisées comprennent la précision, le rappel, la F-mesure et la matrice de confusion. Chaque métrique offre une perspective différente sur les performances du modèle. Dans cette partie, nous allons définir les métriques que nous avons utilisées pour évaluer notre système.

### 2.5.1 Matrice de confusion

La matrice de confusion, comme son nom l'indique, fournit une représentation sous forme de matrice qui décrit les performances globales d'un modèle [22].

		ACTUAL VALUES	
		POSITIVE	NEGATIVE
PREDICTED VALUES	POSITIVE	TP	FP
	NEGATIVE	FN	TN

Figure 2.6: Matrice de confusion [23].

Comme le montre la Figure 2.6 il y a quatre (4) termes importants :

- Vrais positifs (True Positive (TP)) : Les cas où nous avons prédit OUI et que la sortie réelle était également OUI [23].
- Vrais négatifs (True Negative (TN)) : Les cas où nous avons prédit NON et que la sortie réelle était également NON [23].
- Faux positifs (False Positive (FP)) : Les cas où nous avons prédit OUI alors que la sortie réelle était NON [23].
- Faux négatifs (False Negative (FN)) : Les cas où nous avons prédit NON alors que la sortie réelle était OUI [23].

### 2.5.2 Précision

C'est une méthode efficace pour évaluer si le modèle a un nombre élevé de faux positifs par rapport à sa précision en dehors des vrais positifs attendus (combien sont réellement positifs). Si la précision n'est pas suffisamment élevée, l'algorithme peut confondre les classes [24].

$$\text{Précision} = \frac{\text{vrai positif}}{\text{vrai positif} + \text{vrai négatif}} \quad (2.1)$$

### 2.5.3 Rappel

Le rappel est une mesure de la qualité d'une classe, calculé en divisant le nombre d'éléments correctement classés dans la classe (les vrais positifs) par le nombre total d'éléments dans la classe, c'est-à-dire les vrais positifs et les faux négatifs [24].

$$\text{Rappel} = \frac{\text{vrai positif}}{\text{vrai positif} + \text{faux négatif}} \quad (2.2)$$

### 2.5.4 Accuracy

C'est une métrique utilisée pour évaluer la fréquence à laquelle l'algorithme classe correctement un élément donné, c'est-à-dire les vrais positifs et les vrais négatifs [25]

$$\text{Accuracy} = \frac{\text{vrai positifs} + \text{vrai négatifs}}{\text{vrai positifs} + \text{vrai négatifs} + \text{faux positifs} + \text{faux négatifs}} \quad (2.3)$$

### 2.5.5 F1 Score

Le score F1 peut être considéré comme une mesure combinée de la précision et du rappel, avec une valeur optimale de 1 et une valeur minimale de 0. La contribution relative de la précision et du rappel au score F1 est égale :

$$\text{F1 Score} = \frac{2 \times \text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \quad (2.4)$$

## 2.6 Revue scientifique de la reconnaissance des plaques d'immatriculation

L'ALPR est une technique utilisée pour identifier les véhicules en se basant sur leurs PI. Elle joue un rôle essentiel dans de nombreuses applications liées aux transports [26] [27]. De plus, dans le domaine du traitement d'images, l'ALPR est un sujet d'étude extrêmement populaire et dynamique [28]

Dans une étude menée par Y. Jing et al [29]. Les chercheurs ont proposé un système performant basé sur une architecture de CNN pour la ALPR. Leur système de détection des contours a donné de bons résultats dans le prétraitement de l'image grâce à l'utilisation d'un

filtre médian, ainsi que l'application d'un masquage et d'une binarisation pour extraire la PI et les caractères associés. Leur modèle de reconnaissance des caractères a atteint une précision de 98,13 %.

C. Dias et al [30] ont développé un système ALPR spécifiquement conçu pour un campus universitaire en Inde. Leur approche comprenait trois modules clés. Tout d'abord, ils ont utilisé Faster R-CNN pour la détection des PI. Ensuite, ils ont utilisé des techniques de segmentation, telles que les transformations Top-hat et Black-hat, ainsi qu'un filtre de lissage gaussien pour améliorer la qualité des images. Enfin, pour la reconnaissance des caractères, ils ont utilisé l'OCR Tesseract.

P. Batra et al [31] ont développé un système de ALPR basé sur deux étapes principales : la détection de la PI et l'OCR. Pour la détection de la plaque, ils ont utilisé le modèle YOLOv5s, qui a obtenu une accuracy de 87,2%. En ce qui concerne la segmentation et la reconnaissance des caractères, ils ont choisi d'utiliser EasyOCR.

## **2.7 Revue scientifique de la reconnaissance de logos**

Dans leur étude, M.KUNDURACI et H.KAHRAMANLI ÖRNEK [32] proposent une méthode de reconnaissance automatique de marques de véhicules basée sur une seule étape, qui consiste à détecter chaque logo individuellement. Cette étude porte sur 20 classes de logos de véhicules, collectés à partir d'une base de données comprenant 1957 images étiquetées en XML. La base de données est divisée en deux ensembles : un ensemble d'entraînement et un ensemble de test. Ensuite, la méthode Faster RCNN est utilisée pour la détection de la marque. Différentes versions de cette méthode ont été développées. Dans cette étude, la version "ResNet 50" a été utilisée.

## **2.8 Conclusion**

Dans ce chapitre, nous avons présenté les systèmes de contrôle d'accès des véhicules, ainsi que les différentes méthodes de reconnaissance d'un véhicule unique. Nous nous sommes principalement concentrés sur deux caractéristiques clés qui permettent l'accès automatique aux parkings privés, à savoir la plaque d'immatriculation et le logo. Nous avons discuté du rôle de la vision par ordinateur et des réseaux neuronaux dans la reconnaissance de l'ALPR et des logos. Nous avons également abordé les principales architectures de réseaux neuronaux pour la détection et la reconnaissance, ainsi que les métriques d'évaluation que nous utiliserons pour évaluer nos modèles. Ensuite, nous avons présenté l'état de l'art de ces deux domaines. Dans le chapitre suivant, nous aborderons la conception de notre système.

## Chapitre 03 : Conception d'un système de reconnaissance des plaques d'immatriculation et des logos

### 3.1 Introduction

Ce chapitre est consacré à la conception des systèmes d'ALPR et de reconnaissance de logos que nous souhaitons développer. Nous présentons d'abord la conception détaillée de chaque tâche, les mécanismes utilisés, ainsi que la méthodologie sur laquelle nous allons construire notre système. Nous décrivons les différentes approches conceptuelles adoptées pour chaque phase. Nous expliquerons également l'architecture globale de notre application à l'aide de diagrammes UML.

### 3.2 Conception d'un système de reconnaissance des plaques d'immatriculation

Notre système d'ALPR est basé sur quatre principales étapes : la détection de la plaque, l'extraction, la reconnaissance des caractères et la comparaison des résultats obtenus avec ceux de la BD. Afin d'obtenir un système fiable et performant qui donne des résultats satisfaisants, nous avons suivi différentes étapes, comme présenté dans la Figure (3.1).

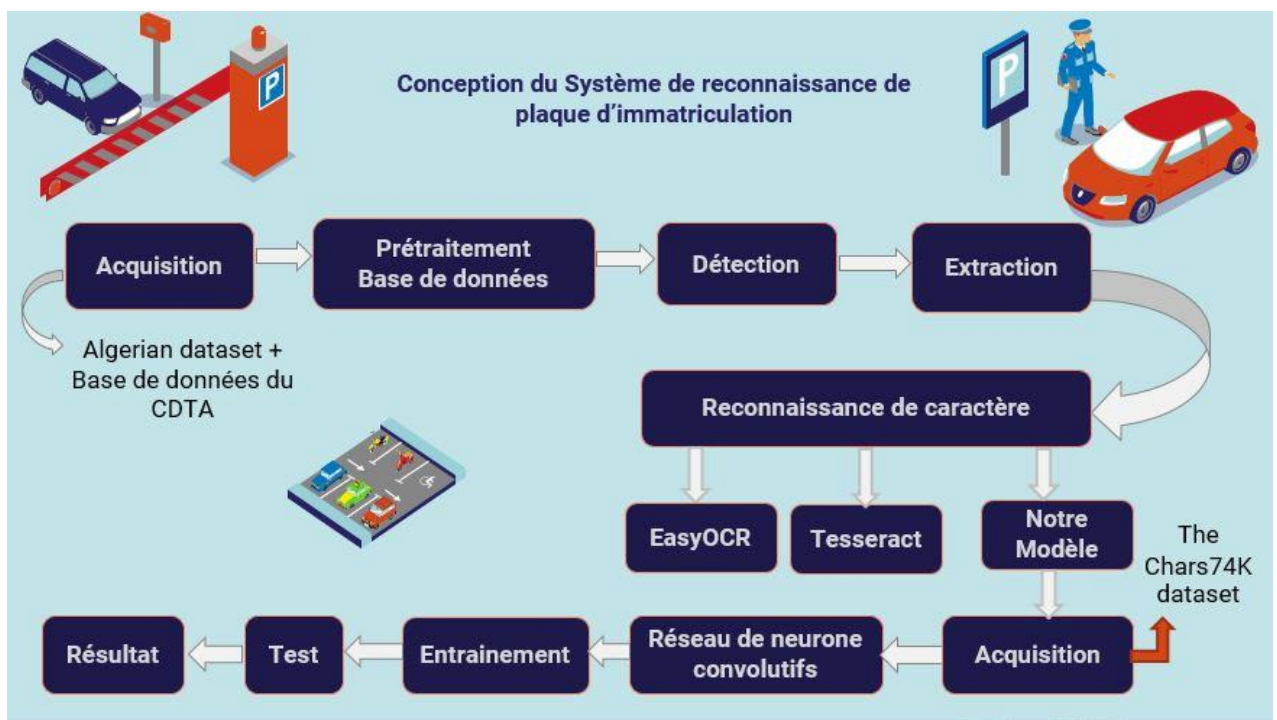


Figure 3.1: Schéma de conception ALPR.

## 3.2.1 Caractéristiques de la plaque d'immatriculation Algérienne

En Algérie, les PI suivent un format défini par le ministère des Transports et de la Mobilité Urbaine.

Le numéro d'immatriculation est composé de plusieurs éléments, lus de droite à gauche. Tout d'abord, un groupe de deux chiffres représente la wilaya (de 1 à 58). Ensuite, un groupe de trois chiffres comprend les informations suivantes : les deux premiers chiffres correspondent à l'année de mise en circulation (année de fabrication), et le troisième chiffre représente la catégorie du véhicule (par exemple, 1 pour les véhicules de tourisme et 2 pour les camions). Enfin, un groupe pouvant contenir jusqu'à six chiffres représente la position du classement d'enregistrement au niveau de la wilaya (Voir Figure 3.2) [33].

Certaines spécifications sont requises pour les plaques d'immatriculation en Algérie. Elles doivent avoir une longueur de 520 mm, une hauteur de 110 mm et une épaisseur de 1 mm. De plus, les plaques doivent être fabriquées en aluminium [33].

Ces informations sont importantes pour assurer la conformité et la cohérence des plaques d'immatriculation dans le pays [33].



**Figure 3.2:** Caractéristiques du matricule Algérien.

## 3.2.2 Acquisition de la base de données de véhicules

L'ensemble de base de données que nous avons utilisé a été collecté à partir de deux sources. La première base de données, appelée "Algerian Dataset"<sup>1</sup> (Figure 3.3), est disponible sur GitHub [34] Elle contient un ensemble d'images de véhicules avec des PI algériennes. L'ensemble de données est divisé en trois ensembles : (i) 2 289 images pour l'entraînement, (ii)436 pour les tests et (iii)118 pour la validation. Ces images ont été collectées à partir de

<sup>1</sup> [https://github.com/mouadb0101/License\\_Plates\\_of\\_Algeria\\_Dataset/tree/master/Detector](https://github.com/mouadb0101/License_Plates_of_Algeria_Dataset/tree/master/Detector)

## Conception d'un système de reconnaissance de plaques d'immatriculation et de logos

Facebook Marketplace et Ouedkniss. Les auteurs ont annoté ces images en utilisant l'outil "BBox-label-tool"<sup>2</sup>.



**Figure 3.3:** Echantillons de la BD Algerian Dataset.

La deuxième BD contient des images prises dans le parking du CDTA. Elle a été collectée par l'équipe du CDTA et est axée sur la problématique de l'impact de la luminosité sur la détection (de manière artificielle). Cette BD est constituée de 90 images. Nous les avons étiquetées manuellement à l'aide de la plateforme proposée par Roboflow<sup>3</sup>. L'échantillon de cette base de données est représenté dans la Figure 3.4.



**Figure 3.4:** Echantillons de la BD du CDTA.

### 3.2.3 Détection de la plaque d'immatriculation

Après avoir effectué une comparaison entre différents modèles de détection, nous avons choisi le modèle YOLOv8 qui donne de meilleurs résultats dans un laps de temps donné.

YOLOv8 effectue des augmentations d'images pendant la formation en ligne. À chaque époque, le modèle voit une variation légèrement différente des images qui lui sont fournies.

<sup>2</sup> <https://github.com/puzzledqs/BBox-Label-Tool>

<sup>3</sup> <https://roboflow.com/>

L'une de ces augmentations est appelée augmentation en mosaïque, où quatre images sont assemblées ensemble. Cela force le modèle à apprendre à détecter des objets dans de nouveaux emplacements, en tenant compte de l'occlusion partielle et des différents pixels environnants [35].

YOLOv8 utilise un réseau neuronal entièrement convolutif qui peut être divisé en deux parties principales : la colonne vertébrale et la tête. La colonne vertébrale de YOLOv8 est une version modifiée de l'architecture CSPDarknet53, qui comprend 53 couches convolutives. Cette architecture utilise des connexions partielles inter-étages pour améliorer le flux d'informations entre les différentes couches du réseau.

La tête de YOLOv8 est composée de plusieurs couches convolutionnelles suivies d'une série de couches entièrement connectées [36]. Ces couches sont responsables de prédire les boîtes englobantes, les scores d'objectivité et les probabilités de classe pour les objets détectés dans une image.

### **3.2.4 Extraction de la plaque d'immatriculation**

L'extraction de la PI est réalisée par YOLOv8 grâce à la fonction "crop". Cette fonction permet de découper et d'extraire des régions d'intérêt spécifiques, telles que la PI et le logo, à partir d'une image. Dans notre cas, elle est utilisée pour isoler les objets détectés par le modèle, en fournissant des sous-images contenant uniquement les objets d'intérêt. Cette approche facilite le traitement et l'analyse ultérieure des objets détectés, en se concentrant uniquement sur les parties pertinentes de l'image.

### **3.2.5 Reconnaissance de caractères à base de EasyOCR et Pytesseract**

Comme le montre la figure 3.1, pour la reconnaissance des caractères, nous avons choisi d'utiliser trois modèles différents, dont deux sont des OCR existants et un modèle que nous avons conçu. Dans cette partie, nous allons les modèles d'OCR.

#### **a- EasyOCR**

EasyOCR est une bibliothèque logicielle open-source utilisée pour la reconnaissance optique de caractères (OCR). Elle est conçue pour être facile à utiliser et permet de reconnaître et d'extraire du texte à partir d'images ou de documents scannés. EasyOCR prend en charge de nombreuses langues et offre une grande précision dans la reconnaissance du texte. Elle peut être utilisée pour diverses applications telles que l'automatisation de la saisie de données, la reconnaissance de documents, la traduction automatique, etc. EasyOCR utilise des modèles de



## *Conception d'un système de reconnaissance de plaques d'immatriculation et de logos*

réseaux neuronaux profonds pré-entraînés pour effectuer la reconnaissance de caractères et peut être intégrée dans des projets Python [37] .

### **b- Pytesseract**

Pytesseract est une enveloppe Python pour Tesseract-OCR. Cette enveloppe permet à Python d'utiliser ce dernier, de plus étant donné que Python peut lire un plus large éventail de formats d'images cela permet à Tesseract-OCR de lire les mêmes formats. Étant donné que le moteur est appelé depuis Python, il n'est pas limité à écrire ses hypothèses dans un fichier de sortie il peut aussi écrire le texte dans Python et l'utiliser dans un script [38].

Pour utiliser la bibliothèque Pytesseract il est nécessaire d'installer au paravent le logiciel Tesseract

#### o **Tesseract :**

Tesseract est un logiciel open source de reconnaissance optique de caractères (OCR) qui permet de convertir du texte imprimé au texte éditable [38]. Le moteur OCR de Tesseract permet à l'utilisateur de spécifier des paramètres, des valeurs et des configurations avec des modèles préformés en différentes langues, afin que le logiciel OCR puisse lire le texte de différentes manières.

Un programme OCR peut analyser et numériser une image avec un texte de plusieurs façons. Alors qu'une configuration peut être avantageuse pour une image avec un texte formant des mots et des phrases, une autre peut être préférable pour la lecture aléatoire de grands caractères. Ce logiciel OCR configurable est ce qui distingue Tesseract-OCR de l'OCR ordinaire [38].

### **3.2.6 Reconnaissance de caractères par CNN**

Dans le cadre de notre projet, nous avons développé un modèle spécifique pour la reconnaissance des PI. Notre modèle propose une approche basée sur deux étapes essentielles : la segmentation et la reconnaissance de caractères.

#### **a-Segmentation**

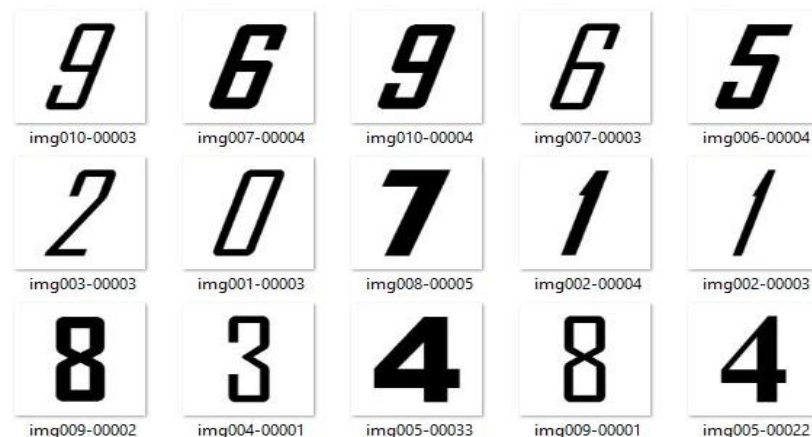
À la différence des modules EasyOCR et Pytesseract, qui contiennent leur propre segmentation, dans notre cas, la segmentation est une étape importante pour obtenir une bonne reconnaissance. Après avoir extrait la PI, nous avons procédé au traitement de l'image de celle-ci afin de détecter les contours des chiffres. Cette étape de détection des contours est essentielle pour isoler chaque caractère individuel et permettre une reconnaissance précise.

## b-Reconnaissance de chiffre

Pour atteindre notre objectif, nous avons procédé à la collecte de données de reconnaissance caractère de PI. Ensuite, nous avons conçu et développé une architecture CNN spécifiquement adaptée à la tâche de reconnaissance de chiffre.

### ○ Acquisition de la base de données de reconnaissance de caractère

"The Chars74K dataset"<sup>4</sup> propose un ensemble de BD contenant tous les caractères de 0 à 9 et de A à Z en majuscules et en minuscules. Il est divisé en deux BD : "EnglishHnd" pour les caractères écrits à la main et "EnglishFnt" pour les caractères en différentes polices informatiques avec 4 variantes. Dans notre cas, nous nous intéressons principalement à la BD "EnglishFnt", car les PI contiennent des chiffres dans un style de caractères informatiques. Après avoir téléchargé cette ensemble de données, nous avons effectué un prétraitement en supprimant toutes les images contenant des caractères de l'alphabet. Nous avons conservé uniquement les chiffres correspondant aux caractéristiques des PI algériennes, ce qui nous a donné une base de données contenant 10 160 images, soit 1 016 images pour chaque chiffre (Voir Figure 3.5).



**Figure 3.5:** Echantillons aléatoires de l'ensemble de données Chars74 dataset.

### ○ Architecture du réseau de neurone

Dans notre cas, nous avons opté pour une architecture similaire à celle d'AlexNet, celle-ci a fait ses preuves dans la reconnaissance de caractères et a résolu de nombreux problèmes de vision par ordinateur. La principale différence entre notre architecture et celle d'AlexNet réside dans la résolution d'image en entrée du réseau de neurones. Pour la première couche convolutive, nous avons choisi de conserver la résolution de nos images telle qu'elle est dans l'ensemble de données "The Chars74K dataset", c'est-à-dire  $(128 \times 128)$ , plutôt que de la changer à la résolution d'AlexNet qui est de  $(224 \times 224)$ . Les Tableaux 3-1 et 3-2 représentent

<sup>4</sup> <http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/>

## Conception d'un système de reconnaissance de plaques d'immatriculation et de logos

les différentes valeurs pour chaque paramètre dans chaque couche convolutive et entièrement connectée.

**Tableau 3-1 :** Couches convolutionnelle de l'architecture AlexNet.

Couches Conv	Filtre	Taille du filtre	padding	Fonction activation	Stride	Pooling		
						Taille	Stride	Padding
Couche 1	96	(11,11)	Valid	ReLU	(4,4)	(3,3)	(2,2)	Valid
Couche 2	256	(5,5)	Same	ReLU	(1,1)	(3,3)	(2,2)	Valid
Couche 3	384	(3,3)	Same	ReLU	(1,1)	/	/	/
Couche 4	384	(3,3)	Same	ReLU	/	/	/	/
Couche 5	256	(3,3)	Same	ReLU	(1,1)	(3,3)	(2,2)	Valid

**Tableau 3-2 :** Couches entièrement connectés de l'architecture AlexNet.

Couches Entièrement connecté	Dense	Fonction d'activation	Régularisation (Dropout)
Couche 1	4096	ReLU	0,5
Couche 2	4096	ReLU	0,8

### 3.3 Conception d'un système de reconnaissance de logos

La Figure 3.6 illustre la conception du système de reconnaissance de logos. Ce système passe par trois étapes clés : la détection, l'extraction du logo et la reconnaissance. Chacune de ces étapes est essentielle pour identifier et reconnaître les logos présents dans une image.

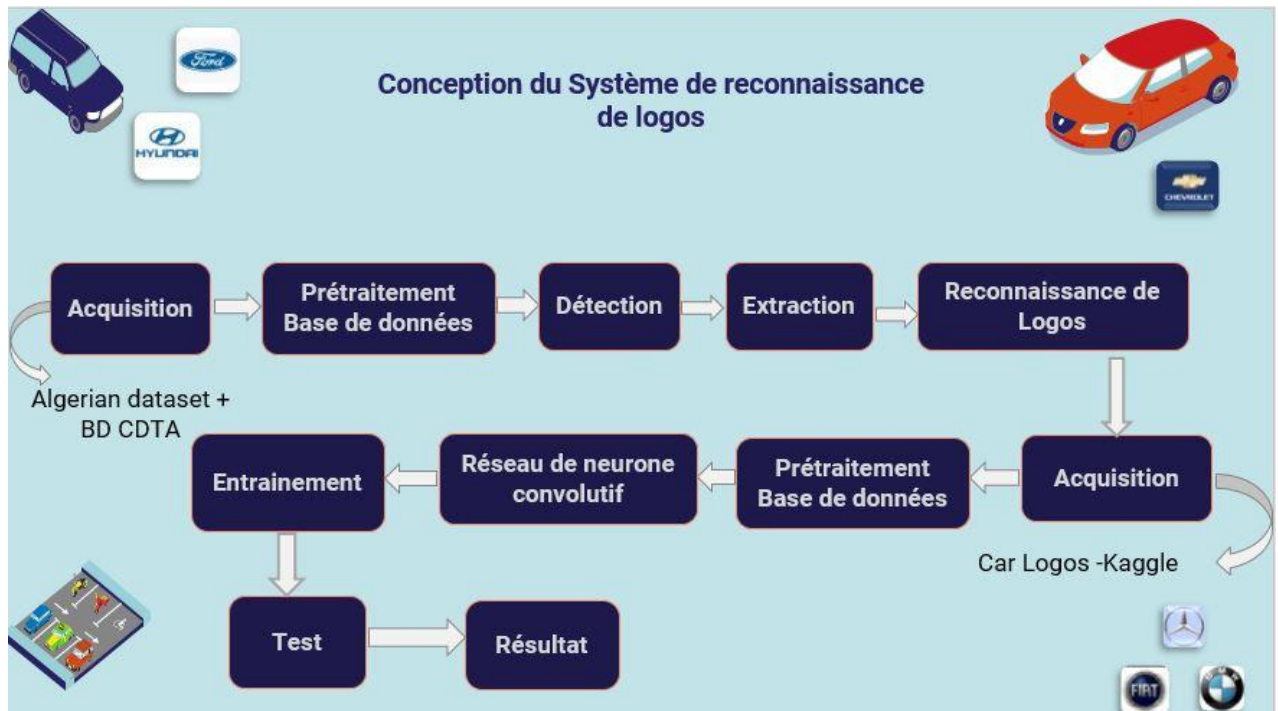


Figure 3.6: Conception de système de reconnaissance de logos.

### 3.3.1 Acquisition de base de données de détection de logos

Pour la partie de détection et d'extraction, nous avons adopté une approche similaire à celle utilisée pour le système ALPR, en utilisant le même ensemble de bases de données. Cependant, une différence notable est que la labélisation des images de logos a été réalisée manuellement grâce à Roboflow, étant donné que notre ensemble de données initial ne contenait pas de labels pour les logos

### 3.3.2 Reconnaissance de logos

Pour atteindre notre objectif, nous avons procédé à la collecte de données de logos. Ensuite, nous avons conçu et développé une architecture CNN spécifiquement adaptée à la tâche de reconnaissance de logo.

#### a-Acquisition de base de données de logos

Pour la reconnaissance de logo, nous avons collecté notre ensemble de données à partir de la plateforme Kaggle. Nous avons trouvé une BD publier sous le nom de "Car logos"<sup>5</sup>. Celle-ci contient 20 778 images réparties en 39 marques de véhicules, notamment Alfa Romeo, Audi, BMW, Chevrolet, Citroën, Dacia, Daewoo, Dodge, Ferrari, Fiat, Ford, Honda, Hyundai, Jaguar,

<sup>5</sup> <https://www.kaggle.com/datasets/jp0909/car-logos/>

## Conception d'un système de reconnaissance de plaques d'immatriculation et de logos

Jeep, Kia, Lada, Lancia, Land Rover, Lexus, Maserati, Mazda, Mercedes, Mitsubishi, Nissan, Opel, Peugeot, Porsche, Renault, Saab, Seat, Subaru, Suzuki, Tata, Tesla, Toyota, Volkswagen et Volvo.

### 3.3.2.1 Architecture du réseau de neurone convolutif

L'architecture que nous avons utilisée comprend cinq couches, dont trois sont des couches convolutives et deux sont entièrement connectées. La première couche convolutive est composée de 64 noyaux (filtres) de taille (3×3), comme indiqué dans les Tableaux 3-3 et 3-4.

**Tableau 3-3 :** Couches convolutionnelle de l'architecture de reconnaissance de logo.

Couches convolutionnelles	Filtre	Taille du filter	padding	Fonction d'activation	Taille Pooling	Régularisation (Dropout)
Couche 1	32	(3,3)	Valid	ReLU	(2,2)	0,3
Couche 2	64	(3,3)	Valid	ReLU	(2,2)	0,25
Couche 3	128	(3,3)	Valid	ReLU	(2,2)	0,25

**Tableau 3-4 :** Couches entièrement connecté de l'architecture de reconnaissance de logo.

Couches Entièrement connecté	Dense	Fonction d'activation	Biais	Fonction de régularisation	Régularisation (Dropout)
Couche 1	4096	ReLU	Glorot-uniform	L2=0,01	0,8
Couche 2	4096	ReLU	Glorot-uniform	L2=0,01	0,8

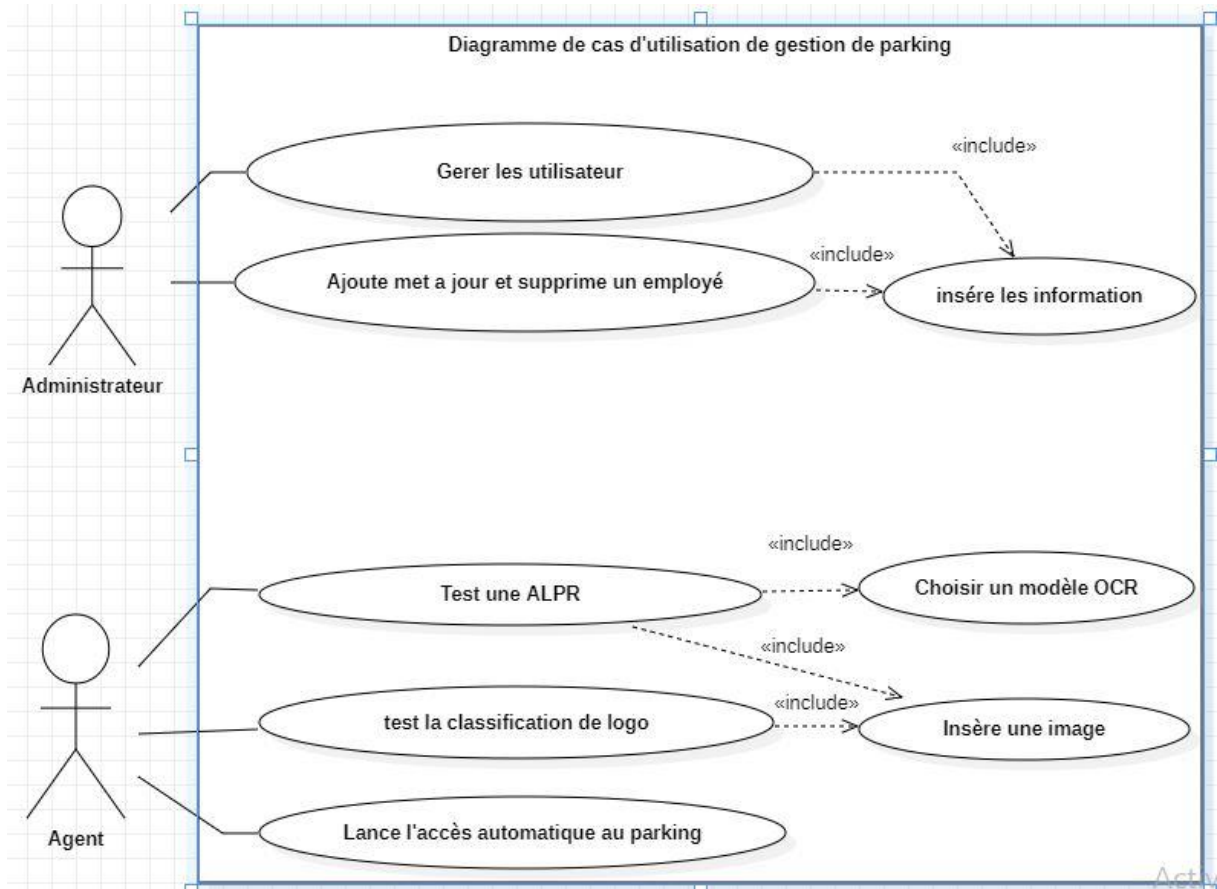
## 3.4 Analyse des besoins

Dans cette partie, nous aborderons l'analyse et la conception du système en utilisant deux types de diagrammes : le diagramme de cas d'utilisation, et le diagramme de séquence.

### 3.4.1 Diagramme de cas d'utilisation

La Figure 3.7 représente le diagramme de cas d'utilisation où nous identifions deux acteurs principaux. L'administrateur, chargé des tâches d'ajout, de suppression et de mise à jour des informations relatives aux visiteurs et aux employés. Il a également la possibilité d'ajouter ou de supprimer des informations sur les parkings. Le deuxième acteur représente l'agent, qui est

responsable de la réalisation des différents tests sur le système de ALPR et sur le système de reconnaissance de logos.

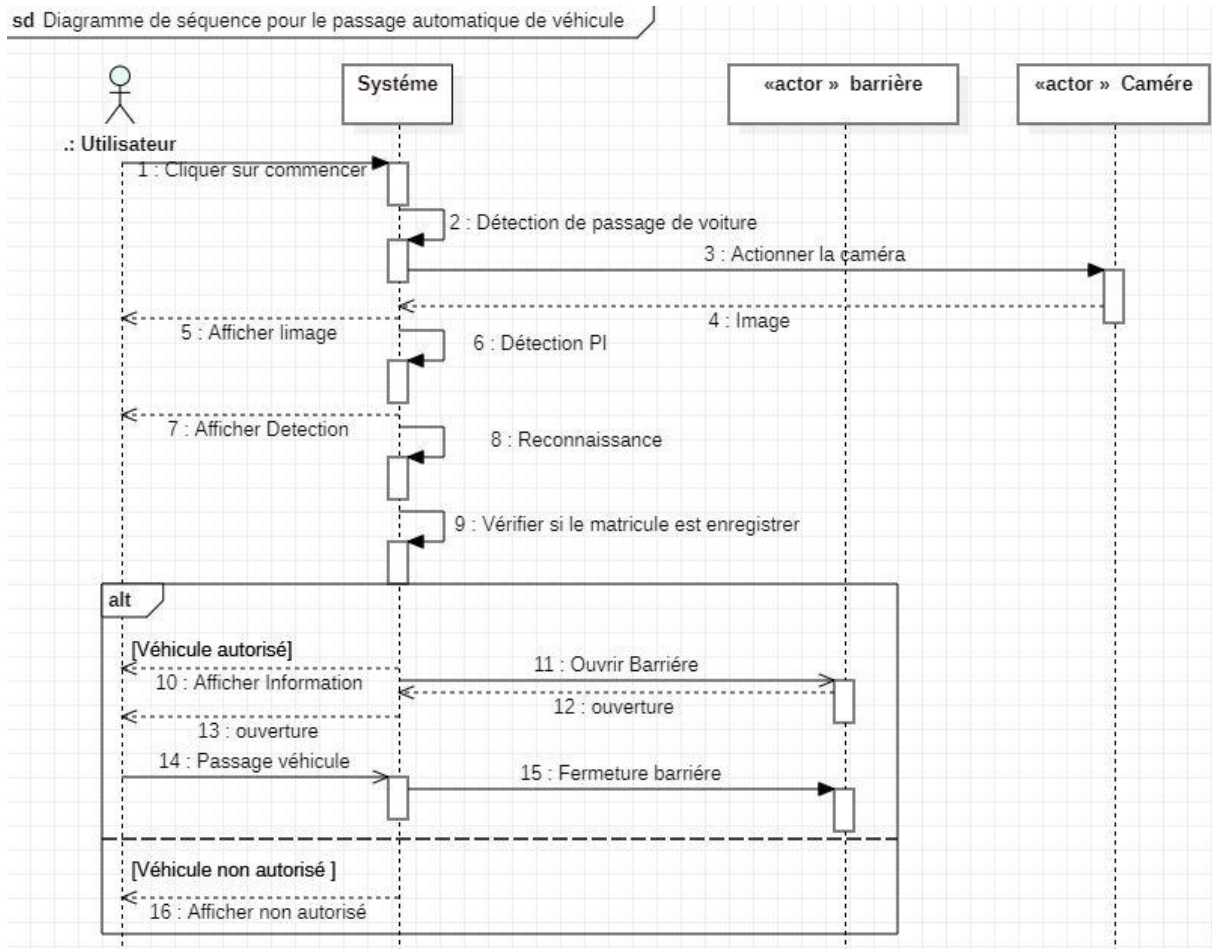


**Figure 3.7:**Diagramme de cas d'utilisation de notre système globale.

### 3.4.2 Diagramme de séquence

Dans cette partie, nous présentons le diagramme de séquence relatifs à l'accées automatiques de véhicules autorisé après la détection du passage de véhicule sur la boucle de courant (voir Figure 3.8).

## a-Diagramme de séquence pour la reconnaissance automatique de la plaque d'immatriculation



**Figure 3.8:** Présentation du diagramme de séquence pour la vérification d'accès automatiques de véhicule

### 3.5 Conclusion

Dans ce chapitre, nous avons présenté la conception de notre système, y compris l'utilisation de l'ensemble de base de données et les architectures choisies. De plus, nous avons décrit la modélisation de notre système.

Dans le chapitre suivant, nous allons aborder les techniques utilisées dans l'application, discuter de notre méthodologie d'implémentation et enfin présenter les résultats obtenus.

## Chapitre 04: Résultats et Interprétation

### 4.1 Introduction

Maintenant que nous avons compris l'ensemble de la théorie et que nous avons conçu nos systèmes, il est temps de passer à la pratique.

Dans ce chapitre, nous aborderons la réalisation du système ALPR et reconnaissance des logos. Nous commencerons par présenter les outils logiciels et matériels utilisés tout au long de ce travail. Ensuite, nous mettrons en avant les bibliothèques les plus importantes pour notre travail. Nous passerons ensuite à la partie du prétraitement effectué, suivie de la discussion des tests réalisés et des résultats obtenus. Enfin, nous conclurons cette section en présentant notre interface graphique.

### 4.2 Outils de développement

Pour le développement de notre système, nous avons utilisé Python [39] comme langage de programmation. Nous avons également utilisé SQLite comme moteur de base de données SQL transactionnelle. En ce qui concerne l'environnement d'exécution, nous avons opté pour VS Code [40] pour l'implémentation de notre interface. Pour l'entraînement des modules de détection et de reconnaissance de notre système, nous avons utilisé Google Colab<sup>6</sup> et Google Drive [41]. Google Colab nous a offert un accès gratuit à un environnement Jupyter Notebook hébergé sur les serveurs de Google. Cela nous a permis de collaborer avec d'autres utilisateurs et de profiter des puissantes ressources de Google, telles que les unités de traitement graphique (GPU) et les unités de traitement tensoriel (TPU) gratuites, afin d'accélérer l'exécution de code intensif en calcul. Dans notre cas, nous avons utilisé le GPU, comme indiqué dans le Tableau 4-1.

**Tableau 4-1 :** Ressources de GoogleColab.

Type	Nom	GPU	Mémoire	Temp gratuit
GPU	Tesla K80	12GB	13GB	6h/24h

Nous avons également utilisé QT Designer [42] pour concevoir et construire le design de notre interface. Pour la labélisation de nos images, nous avons fait appel à Roboflow. Quant à

<sup>6</sup> <https://research.google.com/colaboratory/faq.html>



la gestion de l'ouverture et de la fermeture automatique de la barrière, ainsi que pour la détection du passage du véhicule sur la boucle de courant, nous avons utilisé Arduino.

**Roboflow** :<sup>7</sup> C'est une plateforme de traitement d'images qui permet de simplifier le processus de création, d'annotation et de déploiement de modèles d'apprentissage automatique pour la vision par ordinateur. Elle permet d'importer votre propre ensemble de données (Dataset) et de le convertir dans le format souhaité, puis de l'utiliser pour entraîner un modèle YOLO.

**Arduino** : C'est une plateforme électronique open-source qui combine un matériel et un logiciel. Les cartes Arduino sont capables de lire des entrées telles que la lumière captée par un capteur, l'appui sur un bouton ou un message provenant de Twitter, puis de les convertir en sorties telles que l'activation d'un moteur, l'allumage d'une LED ou la publication en ligne. Vous pouvez donner des instructions à votre carte en lui envoyant un ensemble d'instructions au microcontrôleur intégré. Pour cela, vous utilisez le langage de programmation Arduino, basé sur Wiring, ainsi que le logiciel Arduino IDE, basé sur Processing [43].

### 4.3 Environnement matériel

**Ordinateur** : le Tableau suivant représente les caractéristiques des ordinateurs utilisés.

**Tableau 4-2** : Ressources locales des ordinateurs utilisés.

Windows	Processeur	Ram	Type de système
Windows 10	Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.71 GHz	8GO	Système d'exploitation 64bits
Windows 10	AMD Ryzen 5 5500U avec carte graphique Radeon 2.10GHz	8GO	Système d'exploitation 64 bits

### 4.4 Environnement de conception

Pour la conception de notre système, nous avons utilisé le logiciel StarUML, un outil populaire largement utilisé pour créer des diagrammes UML, notamment des diagrammes de classes, des diagrammes de cas d'utilisation et des diagrammes de séquence.

<sup>7</sup> <https://roboflow.com/>

## 4.5 Bibliothèque

Il existe plusieurs bibliothèques Python qui offrent des fonctionnalités dans le domaine du traitement d'image et de l'apprentissage profond. Ces bibliothèques fournissent un ensemble de fonctions et de classes permettant de résoudre de manière simple des problèmes complexes. Dans le Tableau 4-3 nous avons résumer les bibliothèques les plus importante pour la réalisation de ce travail ainsi que leur version.

**Tableau 4-3 :** Bibliothèques Utilisé et leur version.

Bibliothèque	Version
TensorFlow [44]	2.10.1
Keras [45]	2.10.0
Matplotlib	3.7.0
NumPy [46]	1.24.2
OpenCV [47]	4.7.0
Scikit-learn [48]	1.2.1
PyQt5 [49]	5.15.4
Ultralytics	8.0.87
EasyOCR [50]	1.7.0
Tesseract [38]	0.1.3
Pytesseract [38]	0.3.10
Torch [51]	1.13.1

## 4.6 Implémentation du système de reconnaissance de plaques d'immatriculation

Dans ce qui suit nous allons présenter les différentes étapes de l'implémentation du système ALPR

### 4.6.1 Prétraitement sur la base de données

Après avoir visualisé les deux ensembles de données, nous avons constaté que la résolution des images était différente et que certaines images ne montraient pas la PI complète. Cela nous a amenés à nettoyer la BD en supprimant ces images. Ensuite, nous avons fusionné nos deux bases de données et redimensionné l'ensemble à une résolution de (640 × 640). La base de données obtenue a été divisée en trois sous-ensembles : 60% pour l'entraînement, 20% pour les tests et 20% pour la validation. Enfin, nous avons transformé le type de libellé du texte en libellé

Yolov8 Label en utilisant la plateforme Roboflow. La Figure 4.1 illustre une partie de la base de données après le prétraitement.

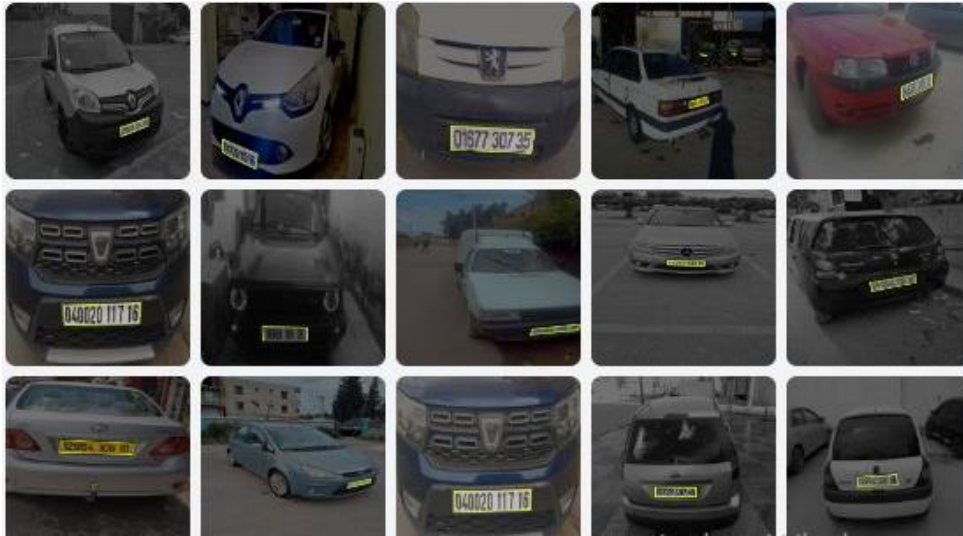


Figure 4.1: Les deux Bases de données Fusionner et labéliser.

#### 4.6.2 Détection et extraction de la plaque d'immatriculation

Ce modèle pré-entraîné prend en charge les tâches de détection, de segmentation et de classification. Pour ré-entraîner ce modèle, nous avons choisi la tâche de détection et le mode "train" pour indiquer que nous sommes en mode d'entraînement. Nous avons spécifié 25 époques, c'est-à-dire le nombre de fois où l'ensemble de données d'entraînement sera parcouru. Chaque époque correspond à un passage complet à travers cet ensemble de données. En tant que paramètre optionnel, nous avons ajouté "plots" avec la valeur true pour demander au modèle de tracer la courbe de perte (Loss) pendant l'entraînement. De plus, nous avons défini le paramètre "crop" sur true, ce qui nous a permis d'extraire automatiquement l'objet détecté de l'image et de l'enregistrer dans un dossier appelé "crops". Cette étape d'extraction de la PI sera utile pour la segmentation ultérieure des caractères.

Nous avons effectué la détection et l'extraction de la PI à l'aide du module de détection proposé par yolov8, soit le modèle yolov8s.pt. Il s'agit d'un modèle pré-entraîné qui prend en charge les tâches de détection, segmentation et classification.

Ce modèle pré-entraîné prend en charge les tâches de détection, de segmentation et de classification. Pour ré-entraîner ce modèle, nous avons choisi la tâche de détection et le mode "train" pour indiquer que nous sommes en mode d'entraînement. Nous avons spécifié 25 époques, c'est-à-dire le nombre de fois où l'ensemble de données d'entraînement sera parcouru. Chaque époque correspond à un passage complet à travers cet ensemble de données. En tant que paramètre optionnel, nous avons ajouté "plots" avec la valeur true pour demander au modèle

de tracer la courbe de perte (Loss) pendant l'entraînement. De plus, nous avons défini le paramètre "crop" sur true, ce qui nous a permis d'extraire automatiquement l'objet détecté de l'image et de l'enregistrer dans un dossier appelé "crops". Cette étape d'extraction de la PI sera utile pour la segmentation ultérieure des caractères.

#### 4.6.3 Test de détection de la plaque d'immatriculation

La Figure 4.2 représente la détection des plaques d'immatriculation d'une image prise d'une perspective éloignée à l'aide d'une caméra de téléphone. Nous remarquons que la détection a réussi, et quatre plaques d'immatriculation ont été détectées.



Figure 4.2: Résultats de détection de la PI dans une image.

Après la détection des différentes plaques d'immatriculation, des sous-images contenant les plaques détectées ont été générées (Voir Figure 4.3).



Figure 4.3: Résultats de l'extraction des plaques

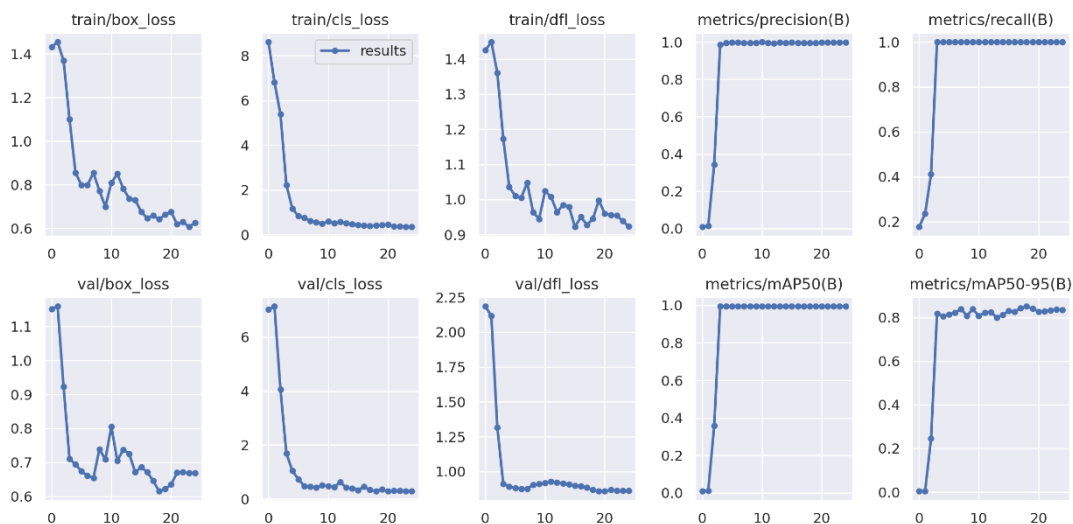
#### 4.6.4 Résultats de détection

Le Tableau 4.4 représente les résultats de la matrice de confusion de la détection de plaques d'immatriculation.

**Tableau 4-4 :** Matrice de confusion pour la détection de plaque d'immatriculations.

	PI	Absence de PI
PI	1.00	0
Absence de PI	0	0

La Figure 4.6 présente les graphiques obtenus lors de l'entraînement du modèle de détection. Sur l'axe des x, nous représentons les époques. Il est intéressant de noter que les graphiques de précision, de rappel et de mAP atteignent une valeur maximale de 1, démontrant ainsi une performance optimale du modèle. En ce qui concerne le cls-loss, il diminue progressivement pour l'ensemble d'entraînement et l'ensemble de validation, atteignant une valeur de 0,1. Cela indique que le modèle parvient à bien classer les objets détectés. Quant au box-loss, il présente également une diminution constante, atteignant une valeur de 0,6 pour l'ensemble d'entraînement et 0,68 pour l'ensemble de validation. Cette tendance démontre que le modèle parvient à ajuster les boîtes de détection de manière précise et cohérente. Ces résultats mettent en évidence les performances solides du modèle de détection, avec des valeurs élevées de précision et de rappel, ainsi qu'une diminution régulière des pertes cls-loss et box-loss. Cela confirme l'efficacité de notre approche d'entraînement et renforce notre confiance dans la qualité du modèle obtenu.



**Figure 4.4 :** Graphes des métriques d'évaluation de la détection de PI par yolo8

#### 4.6.5 Segmentation

Dans l'image extraite, il y a plusieurs taches ou régions appelées "blobs". Notre objectif est de déterminer quels "blobs" correspondent aux caractères de la plaque d'immatriculation. Pour ce faire, nous utilisons un algorithme appelé analyse des composantes connectées (connectedComponents). Cet algorithme nous permet d'identifier et de regrouper les régions connectées entre elles, afin de distinguer les caractères de la plaque d'immatriculation des autres éléments présents dans l'image. Ainsi, grâce à l'analyse des composantes connectées, nous sommes en mesure de sélectionner précisément les zones pertinentes pour la reconnaissance automatique des plaques d'immatriculation. (Voir Figure 4.5)

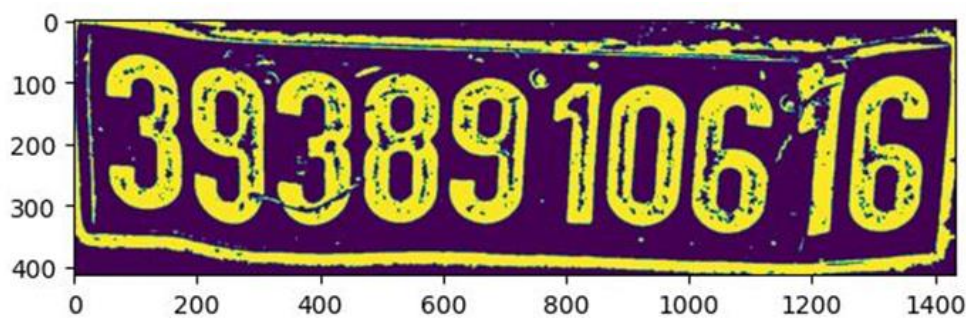


Figure 4.5 : Tache "blobs" des caractères

Pour déterminer si un "blob" correspond à un caractère, une approche heuristique est utilisée. À partir de l'image binaire, nous observons que le nombre de pixels pour chaque caractère se situe dans une plage spécifique. Ainsi, nous établissons une limite inférieure (100) et une limite supérieure (20) qui définissent le nombre de pixels attendu pour chaque composante connectée. Après la sélection des composantes nous utilisons la fonction findContours qui est une fonction d'OpenCV utilisée pour détecter et extraire les contours d'une image. Elle prend en entrée une l'image binaire ainsi que la plage de pixels pour les composants et identifie les contours qui représentent les limites des objets présents dans l'image.

Enfin Les contours obtenus ne sont pas ordonnés. Avant de poursuivre, il est nécessaire de trier les boîtes englobantes de gauche à droite et de haut en bas afin de lire les caractères dans le bon ordre. Pour cela, nous utilisons la fonction de tri Sorted qui permet de trier les boites englobantes dans un certain ordre. Elle prend en entrée la liste de caractère désordonner, et retourne une nouvelle séquence contenant les éléments triés (voir Figure 4.6).



**Figure 4.6 :** Segmentation des caractères avec des cadres délimitateurs (Bounding boxes)

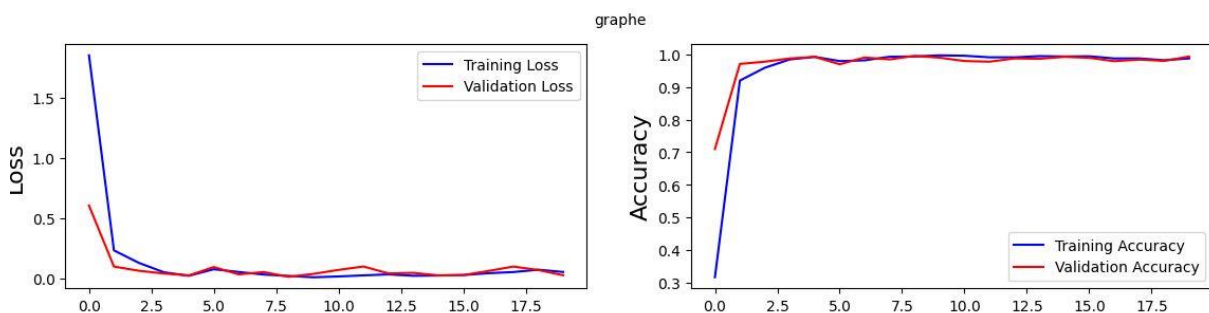
#### 4.6.6 Entraînement du modèle de reconnaissance de chiffre

Pour l'entraînement, nous avons choisi un batch-size de 128, fixé le nombre d'époques à 20, et spécifié la largeur et la hauteur cible des images à 128 afin de conserver la même résolution que les images d'origine. Afin de préserver les couleurs originales des images, nous avons fixé le paramètre TARGET\_DEPTH à 3, ce qui signifie que nous voulons obtenir des images avec une profondeur de 3 canaux de couleur (RVB). Cela nous permet de maintenir les images en couleur et de préserver les canaux RVB. Enfin, nous avons utilisé l'accuracy comme métrique d'évaluation pendant l'entraînement.

Le modèle obtenu sera pas la suite utiliser après la segmentation pour prédire les chiffres du matricule.

#### 4.6.7 Teste et résultats du modèle de reconnaissance de chiffre

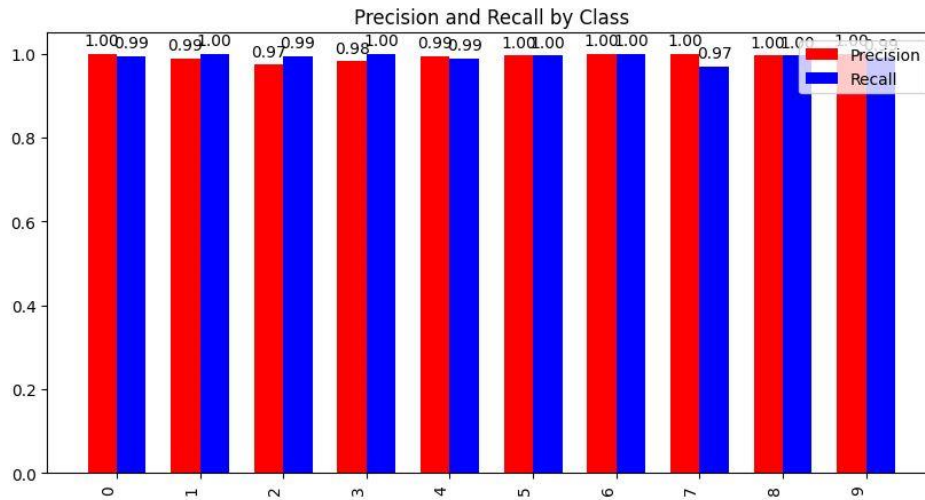
La Figure 4.7 représente l'évaluation de l'exactitude (accuracy) et de la fonction de perte (loss) sur les ensembles d'entraînement et de validation pendant l'entraînement du modèle. Les résultats sont présentés sous forme de graphique, où la courbe bleue représente l'ensemble d'entraînement et la courbe rouge représente l'ensemble de validation.



**Figure 4.7 :** Evaluation d'Accuracy et de perte durant l'entraînement du modèle d'évaluation

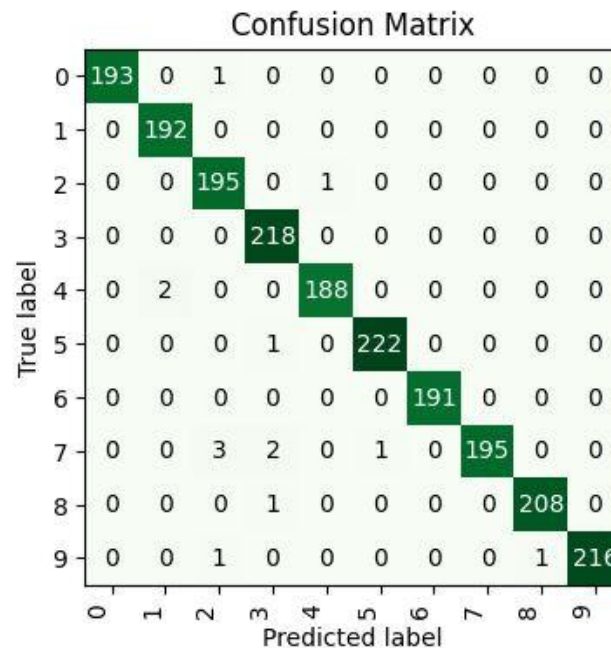
Enfin, après avoir testé le modèle sur l'ensemble de test, nous avons obtenu une Accuracy de 99,31%.et un Loss de 0,023

La Figure 4.8 présente la précision et le rappel obtenus pour chaque chiffre, où la couleur rouge représente la précision et la couleur bleue représente le rappel.



**Figure 4.8 :** Précision et rappel obtenue pour chaque chiffre.

La Figure 4.9 présente la matrice de confusion obtenue à partir d'un ensemble de test. Nous remarquons que la diagonale, qui représente les vrais positifs, présente des chiffres élevés. En revanche, le reste de la matrice, comprenant les faux positifs, les faux négatifs et les vrais négatifs, contient généralement des zéros.



**Figure 4.9 :** Matrice de confusion pour le modèle de reconnaissance de chiffre.

Le Tableau 4-5 représente les résultats du F1 Score pour la reconnaissance des chiffres. Ces résultats démontrent la haute précision et la performance globale exceptionnelle de la reconnaissance.



**Tableau 4-5 :** F1Score pour la reconnaissance de chiffre.

Classe	F1- score
0	1.00
1	0.99
2	0.98
3	0.99
4	0.99
5	1.00
6	1.00
7	0.98
8	1.00
9	1.00

## 4.7 Implémentation de système de reconnaissance de logos

Dans cette partie nous allons présenter les différents techniques du processus d'implémentation du système de reconnaissance de logos.

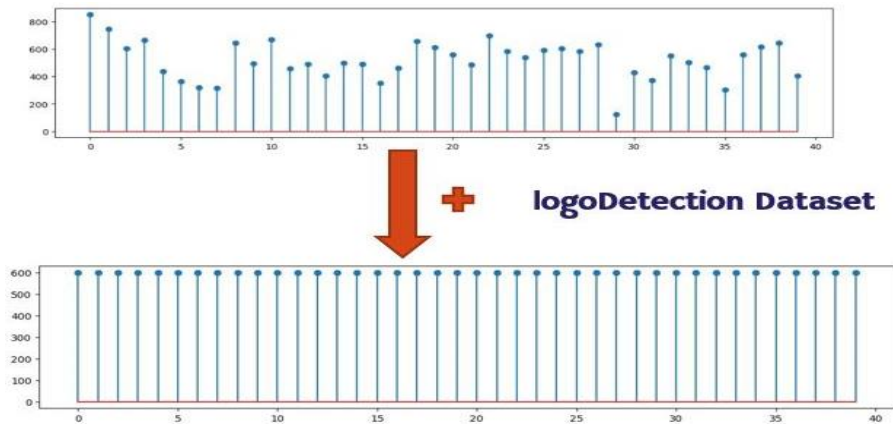
### 4.7.1 Prétraitement sur la base de données

Après avoir téléchargé la base de données, nous avons remarqué la présence de plusieurs images défectueuses (voir Figure 4.10). Certaines images étaient mal classées, par exemple une Skoda était classée comme une Opel. Nous avons également constaté la présence d'images contenant quatre logos, qui se répétaient dans chaque classe, ainsi que des images de tasses de café avec plusieurs motifs ou des t-shirts avec un logo placé à gauche. Nous avons donc commencé par supprimer ces images afin d'obtenir de meilleurs résultats.

**Figure 4.10 :** Images défectueuse supprimer de la base de données de logos.

Une fois les images contenant de fausses informations supprimées, nous avons tracé un graphique des sommets pour chaque classe afin de visualiser le nombre d'images disponibles. Nous avons constaté un déséquilibre dans notre base de données, ce qui nous a conduit à équilibrer celle-ci en ajoutant des images provenant d'une autre base de données trouvée sur

GitHub, appelée "logoDetection"<sup>8</sup>. La Figure 4.11 représente notre base de données avant et après l'équilibrage :



**Figure 4.11 :** Base de données de logos avant et après le rééquilibrage.

Après l'équilibrage de notre ensemble de données nous avons obtenu un ensemble de 23400 images. Nous avons divisé cet ensemble en trois sous ensemble soit 60% pour l'entraînement 20% pour le test et 20% pour la validation.

#### 4.7.2 Détection et Extraction de logos

Pour la détection et l'extraction des logos, nous avons suivi la même approche que pour la détection et l'extraction de la PI, en utilisant la même base de données.

#### 4.7.3 Test de détection de logo

La Figure 4.12 représente la détection du logo dans une image.



**Figure 4.12 :** Détection de logo dans une image.

<sup>8</sup> <https://github.com/SelcukYolcu/logoDetection/tree/master/>

La Figure 4.13 montre l'image extraite de la figure 4.12, cet objet par la suite vas nous permettre d'effectuer la reconnaissance du logos à l'aide de notre modèle.



Figure 4.13 : Extraction du logo à partir d'une image.

#### 4.7.4 Résultats de détection de logos

Le Tableau 4-6 représente les résultats de la matrice de confusion de la détection de logos.

Tableau 4-6 : : Matrice de confusion pour la détection de Logos

	Logos	Absence de Logos
Logos	0.90	0.00
Absence de Logos	0.10	1.00

La Figure 4.14 présente les graphiques obtenus lors de l'entraînement du modèle de détection de logos. Sur l'axe des x, nous représentons les époques. Il est intéressant de noter que les graphiques de précision, de rappel et de mAP atteignent 0,9. En ce qui concerne le cls-loss, il diminue progressivement pour l'ensemble d'entraînement et l'ensemble de validation, atteignant une valeur de 0,1. Cela indique que le modèle parvient à bien classer les objets détectés. Quant au box-loss, il présente également une diminution constante, atteignant une valeur de 1,1 pour l'ensemble d'entraînement et 1,96 pour l'ensemble de validation.

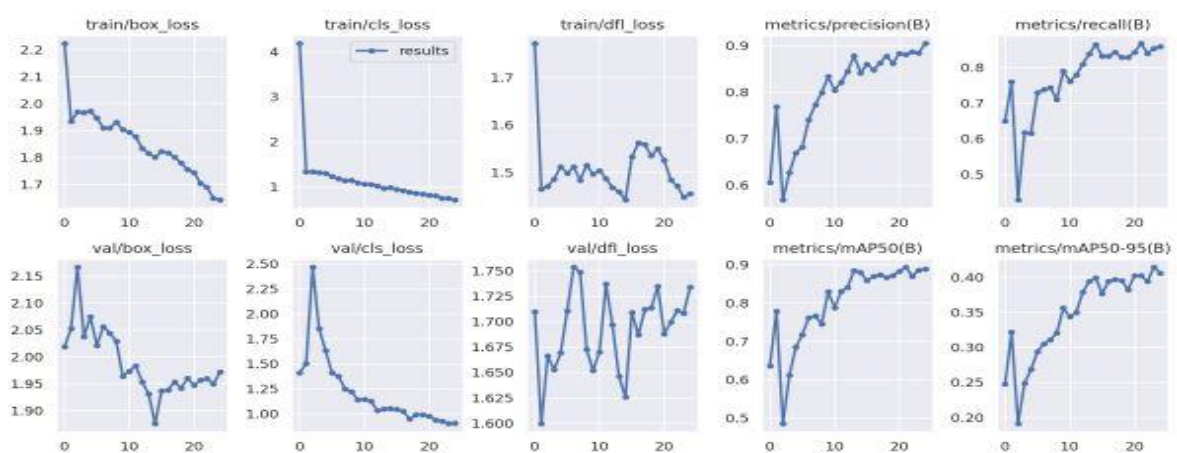
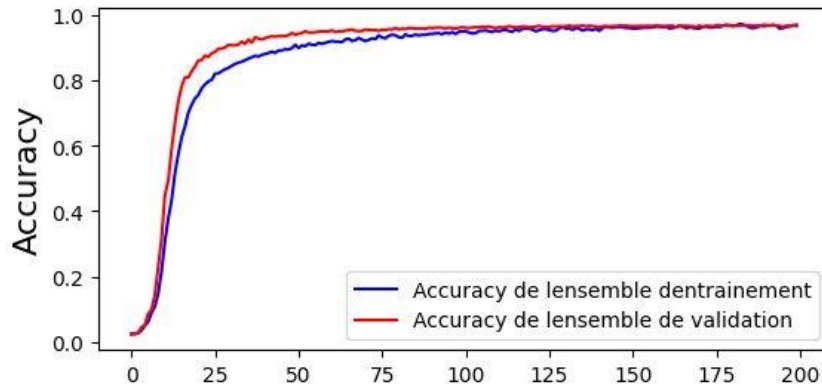


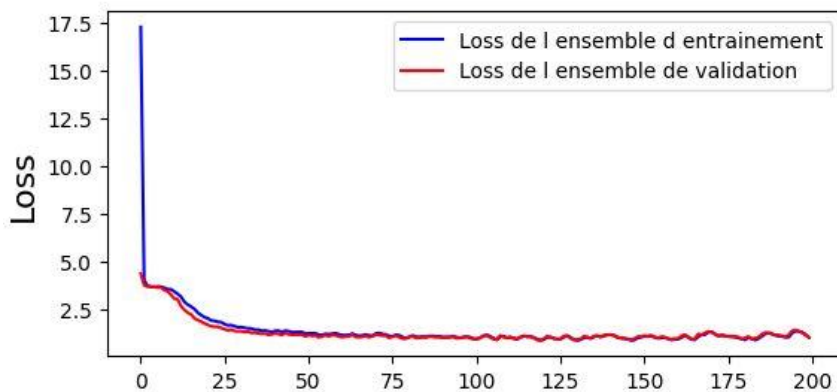
Figure 4.14 : graphes des métriques d'évaluation de détection de logos.

#### 4.7.5 Test et Résultats de classification de logo

Les Figures 4.15 et 4-16 présente respectivement l'évolution de l'accuracy et de la fonction de perte d'entraînement et de validation du modèle de reconnaissance de logos au cours du processus d'entraînement, sous forme de graphiques.



**Figure 4.15 :** Accuracy de l'entraînement et de la validation pour la classification de logos durant l'entraînement du CNN.



**Figure 4.16 :** Loss de l'ensemble d'entraînement et de la validation pour la reconnaissance de logos durant l'entraînement du CNN.

Après avoir évalué l'ACC de notre modèle sur l'ensemble de test, nous avons obtenu un résultat exceptionnel de 97.04%. Comme le montre le Tableau 4-7 cette performance remarquable dépasse de loin celle du modèle de [32] .

**Tableau 4-7 :** Comparaison entre les résultats d'Accuracy de notre modèle et le modèle.

Modèle	Accuracy
Notre modèle	97.04%
Modèle [32]	67.66%

Dans le tableau 4-8 nous pouvons observer les mesures de précision et de rappel obtenues pour chaque marque, ainsi que le F1Score obtenue, nous remarquons que les valeurs les plus

faibles sont ceux de la marque Alfa Romeo qui donne 0.81 pour la précision, 0.90 pour le rappel, et 0.85 pour le F1Score

**Tableau 4-8 :** Précision, rappel et F1Score des marques de véhicule.

Marque	Précision	Rappel	F1Score
Alfa Romeo	0.81	0.90	0.85
Audi	0.82	0.92	0.87
BMW	0.99	0.95	0.97
Chevrolet	0.99	0.95	0.97
Citroën	1.00	0.99	1.00
Dacia	0.98	0.98	0.98
Daewoo	0.98	0.96	0.97
Dodge	0.99	0.96	0.99
Ferrari	0.99	0.98	0.98
Fiat	0.97	0.99	0.98
Ford	0.98	0.98	0.98
Honda	1.00	0.99	1.00
Hyundai	0.98	0.98	0.98
Jaguar	0.95	0.98	0.96
Jeep	0.97	0.90	0.94
Kia	0.98	0.98	0.98
Lada	1.00	0.98	0.99
Lancia	0.94	0.98	0.96
Land Rover	0.97	0.97	0.97
Lexus	0.97	0.98	0.97
Maserati	0.98	0.97	0.98
Mazda	0.98	0.98	0.98
Mercedes	0.99	0.95	0.97
Mitsubishi	0.97	0.99	0.98
Nissan	0.98	0.99	0.99
Opel	1.00	0.99	1.00
Peugeot	1.00	0.97	0.99
Porsche	0.99	0.96	0.98
Renault	0.99	0.96	0.98
Saab	0.99	0.96	0.98
Seat	0.98	0.99	0.99
Skoda	0.95	0.97	0.96
Subaru	0.99	0.98	0.99
Suzuki	0.96	0.98	0.97
Tata	1.00	0.98	0.99
Tesla	0.98	0.98	0.98
Toyota	1.00	0.95	0.98
Volkswagen	0.98	0.95	0.96
Volvo	0.98	0.98	0.98

<b>Moyenne</b>	<b>0.97</b>	<b>0.97</b>	<b>0.97</b>
----------------	-------------	-------------	-------------

La Figure 4.17 représente la matrice de confusion obtenue à partir de notre modèle de reconnaissance de logos. Cette matrice met en évidence les performances de notre modèle en termes de classification correcte et incorrecte des logos des différentes marques automobiles. Nous pouvons constater que la diagonale de la matrice, présente des valeurs élevées pour la plupart des marques. Cela témoigne de la capacité de notre modèle à reconnaître et à classifier avec précision les logos des marques automobiles. Les valeurs en dehors de la diagonale, représentant les erreurs de classification (faux positifs et faux négatifs), sont généralement faibles, ce qui confirme la fiabilité de notre modèle dans la reconnaissance des logos. Ces résultats encourageants démontrent l'efficacité de notre approche de reconnaissance de logos et soulignent le potentiel de notre modèle dans des applications pratiques.



## 4.8 Présentation de l'application

Dans cette section, nous allons présenter notre interface et expliquer son fonctionnement. La Figure 4.18 illustre la page d'accueil, celle-ci constitue le point de départ pour accéder aux différentes fonctionnalités. Cette page comporte 5 boutons, chacun ayant un rôle spécifique :

1. **Employé** : En cliquant sur ce bouton, la liste complète des employés s'affiche, offrant ainsi une vue sur l'ensemble des membres de l'entreprise et facilite leur gestion.
2. **Véhicule** : Ce bouton permet d'afficher l'ensemble des informations sur les véhicules et d'effectuer des actions de gestion telles que l'ajout, et la suppression des informations d'un véhicule.
3. **Notre modèle** : En cliquant sur ce bouton, vous accédez à l'interface de reconnaissance qui intègre notre propre modèle. Cela vous permet de bénéficier de notre algorithme de reconnaissance avancé.
4. **Pytesseract** : Ce bouton vous redirige vers l'interface de reconnaissance utilisant la bibliothèque de Tesseract. Cette option peut être utile si vous souhaitez comparer les performances de notre modèle avec celles de Tesseract.
5. **EasyOCR** : En cliquant sur ce bouton, vous pouvez accéder à l'interface de reconnaissance basée sur la bibliothèque EasyOCR. Cette alternative offre une autre approche de la reconnaissance de texte.
6. **Exit** : Permet de fermer l'application.

Ces différents boutons fournissent des fonctionnalités variées pour faciliter la gestion des utilisateurs, la visualisation des employés, l'ajout de nouvelles compagnies et l'utilisation de différents modèles de reconnaissance de texte.



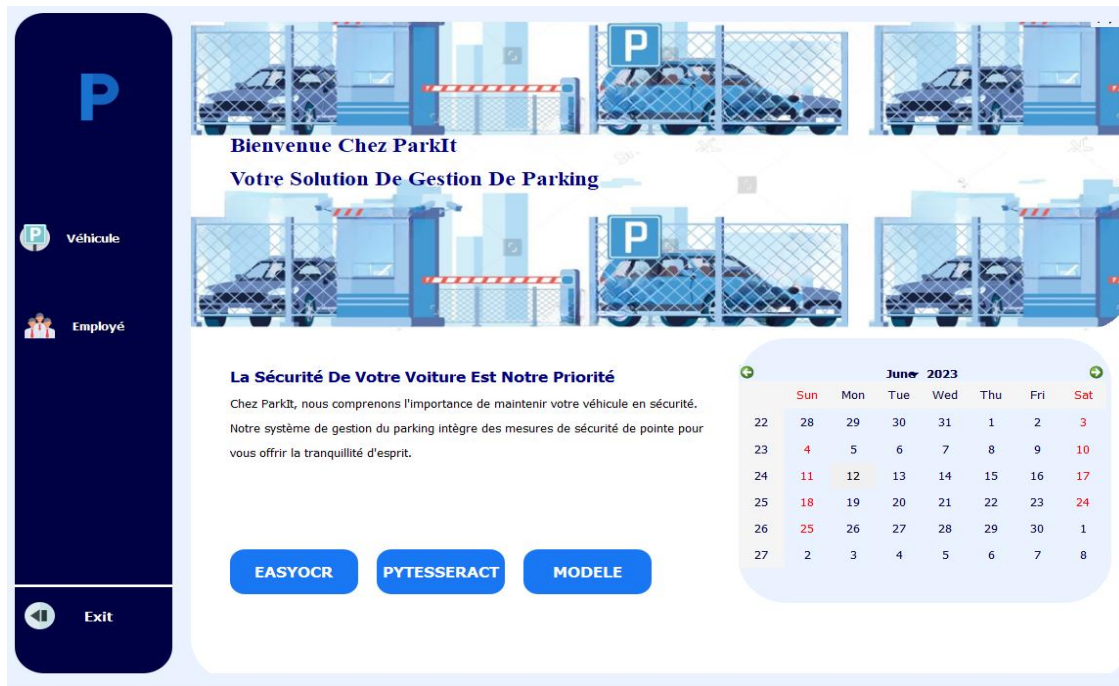


Figure 4.18 : Page d'accueil.

La Figure 4.19 présente l'interface permettant d'afficher les informations de tous les employés enregistrés dans la base de données. Cette interface offre également la possibilité d'ajouter un nouvel employé ou de supprimer un employé existant. Pour accéder à cette interface, il suffit de cliquer sur le bouton "Information".

Cette interface offre une vue globale et organisée des informations des employés, permettant une consultation facile et rapide.

Base de Donnees								
P								
Afficher Supprimer								
	Prenom	Poste	Grade	Telephone	N.Bureau	Email	Matricule	Marque
1	Bouchaffra	DR ASM	DR	336	102	dbouchaffra@c...	5636511616	Dacia
2	Benmehidi	Soutien	Ingénieur	387	147	nbenmehidi@c...	3865410816	Hyundai
3	Bouldjennet	Chef Section ...	Ingénieur	269	15	hbouldjennet@...	5408511316	Ford
4	Bouderbal	Administrateur ...	Non Disponible	404	38	bouderbal@cdt...	1058111616	Renault
5	Bouzektit	Technicien	Non Disponible	269	Non_Disponible	Non_Disponible	622310016	Daewoo
6	Achoui	Technicien	Non Disponible	269	Non_Disponible	Non_Disponible	05203210516	Fiat
7	Benbelkacem	Maître de ...	DR	382	172	sbenbelkacem...	0923710016	Non_Disponible
8	Kechad	Ingénieur	Ingénieur	421	158	rkechad@cdda.dz	0247011642	Hyundai
9	Gheriani	Non Disponible	Ingénieur	418	Open_Space	agheriani@cdda...	03728411516	seat
10	Bellabas	Soutien	Technicien ...	400	48	zbellabas@cdda...	01070411616	KIA
11	Laddada	Attaché de ...	Magister	485	209!	sladdada@cdda...	0914511335	Renault
12	Saidoun	Administrateur ...	Juriste	141	263!	zsaidoun@Cdda...	05640911316	Chevrolet
13	Kaloune	Résponsable de...	Ingénieur de ...	418!	Open_Space!	Skaloune@cdda...	9370111216	chevrolet
14	Oulefki	Chercheur	Maitre de ...	433!	70!	aoulefki@cdda.dz	0276711134	Renault

Figure 4.19 : page de la base de données des employé .

La Figure 4.20 présente l'interface permettant d'ajouter un nouvel employé à notre base de données SQL. Les champs vides sont destinés à être remplis avec les informations de la personne concernée. Une fois cela fait, l'utilisateur a la possibilité d'annuler l'opération en cliquant sur le bouton "Annuler", ou bien d'enregistrer les informations en cliquant sur le bouton "Enregistrer" si les informations sont correctement saisies.

**Figure 4.20 :** Page d'enregistrement d'une nouvelle personne.

La Figure 4.21 présente l'interface dédiée à l'affichage des informations relatives aux véhicules enregistrés dans la BD. Cette interface propose également des fonctionnalités permettant d'ajouter de nouveaux véhicules ou de supprimer des véhicules existants. Il est important de noter que chaque véhicule est associé à un numéro d'employé unique, ce qui facilite l'accès aux informations du conducteur correspondant dans la base de données des employés.

	NEmploye	Matricule	Marque
1	1	5636511616	Dacia
2	2	3865410816	Hyundai
3	3	5408511316	Ford
4	4	1058111616	Renault
5	5	622310016	Daewoo
6	6	05203210516	Fiat
7	7	0923710016	Non_Disponible
8	8	0247011642	Hyundai
9	9	03728411516	Seat
10	10	01070411616	KIA

**Figure 4.21 :** Page de la base de données des employés.

La Figure 4.22 présente l'interface permettant d'ajouter un nouveau véhicule à notre base de données SQL. Les champs vides doivent être remplis avec les informations du véhicule concerné. Cette opération nécessite également d'insérer le numéro de l'employé afin de relier le véhicule enregistré à celui-ci. Une fois cela fait, l'utilisateur a la possibilité d'annuler l'opération en cliquant sur le bouton "Fermer" ou d'enregistrer les informations en cliquant sur le bouton "Enregistrer" si les informations sont correctement saisies.

**Figure 4.22 :** Page d'enregistrement d'un nouvel véhicule.

Les Figures 4.23 et 4.24 présente l'interface principale de notre système. L'objectif de cette page est de vérifier si un véhicule est enregistré au nom d'un employé grâce à l'ALPR et la reconnaissance de logo. L'agent a deux options : soit cliquer sur "Capturer une image" pour ouvrir la caméra IP et prendre une photo en appuyant sur la touche d'espace, soit cliquer sur "Parcourir" pour accéder aux fichiers de l'ordinateur et choisir une image existante. Dans les deux cas, l'image sera affichée dans l'espace prévu.

Ensuite, l'agent doit cliquer sur le bouton « Détecter » pour détecter à la fois la PI et le logo. Ensuite, nous avons six autres boutons avec les fonctions suivantes :

- Ouvrir/Fermer : permet d'ouvrir ou de fermer manuellement la barrière. Ce bouton est utile lorsque l'employé arrive avec un véhicule différent sans avoir préalablement averti, ou lorsque la plaque est très sale et illisible par le système.

- Reconnaissance : reconnaît la PI et le logo et affiche le texte dans le champ de reconnaissance. Il vérifie s'ils sont présents dans la base de données. Si c'est le cas, il affiche "Accès autorisé" et les informations de l'employé (nom, prénom, poste, grade, téléphone,

numéro de bureau, e-mail) récupérées de la base de données, comme le montre la figure 4.26. Ensuite, la barrière s'ouvre pour laisser passer le véhicule.

- Reconnaître le logo : utilise le modèle de classification de logo et affiche la classe prédite parmi les 40 classes dans le champ "Marque".
- Retour : permet de revenir à la page d'accueil
- Effacer : permet d'effacer les résultats affichés.
- Fermer : permet de fermer le programme.

The screenshot shows the 'Parking Management' application interface. At the top, there is a dark blue header with a large white 'P' logo on the left and the text 'Parking Management' in white on the right. Below the header, the interface is divided into several sections. On the left, there are three buttons: 'Capturer une image' (light blue), 'Parcourir' (dark blue), and 'Ouvrir/Fermer' (light blue). Below these buttons, there is a 'Detecter' button (dark blue) and a 'Reconnaissance' button (dark blue). Underneath, there are two input fields: 'Reconnaissanc' and 'Verification d'accès', both with light blue backgrounds. At the bottom left, there is an 'Effacer' button (light blue). On the right side, there is a section titled 'Information du Conducteur' with ten input fields for: Nom, Prenom, Poste, Grade, Telephone, N.bureau, E-mail, Matricule, and Modele/ Marque. At the bottom right, there is a 'Fermer' button (light blue).

**Figure 4.23 :** page de reconnaissance de PI et logo.

The screenshot shows a web application window titled "Management" with a "Parking Management" header. The interface includes a central image of a silver car's rear with a license plate "02773 117 31" and a logo "0272". Below the image are buttons for "Capturer une image", "Parcourir", "Ouvrir/Fermer", "Detecter", and "Reconnaissance". To the right, there is a form for "Information du Conducteur" with fields for Nom, Prenom, Poste, Grade, Telephone, N.bureau, E-mail, Matricule, and Modele/ Marque. Below the image, there are input fields for "Reconnaissance" (0277311731) and "Verification d'accès" (AUTHORISED), along with an "Effacer" button. A "Fermer" button is located in the bottom right corner.

Figure 4.24 : Interface de reconnaissance avec l'accès autorisé.

## 4.9 Conclusion

Dans ce chapitre, nous avons présenté une méthode complète d'ALPR et de classification de logos. Nous avons exposé les différentes étapes nécessaires et partagé les résultats obtenus.

Les modules de détection ont affiché un taux de réussite de 99%. Pour le modèle de reconnaissance de caractères, le taux de réussite est de 99,03%. Quant à notre module de segmentation, s'est avéré efficace sur la majorité des images testées, et le modèle de classification de logos a obtenu une précision de plus de 97%.

En termes de temps d'exécution, les deux systèmes se sont révélés relativement rapides, nécessitant entre 2 et 7 secondes pour accomplir leurs tâches respectives.

Ces performances illustrent l'efficacité et la fiabilité des algorithmes proposés, renforçant ainsi la robustesse de notre système.

## Chapitre 05: Conclusion

### 5.1 Conclusion

Ce travail consiste en la conception et l'implémentation d'un système de contrôle d'accès automatique basé sur la détection et la reconnaissance en temps réel des plaques d'immatriculation (PI) et des logos. Nous avons réalisé la détection de ces objets en utilisant l'architecture YOLO.v8. En ce qui concerne la reconnaissance des plaques d'immatriculation, nous avons utilisé trois approches différentes : (i) EasyOCR, (ii) Tesseract et (iii) la segmentation des caractères, suivie par la reconnaissance à l'aide d'un modèle basé sur l'architecture ALEXNet et reposant sur un CNN. Pour la reconnaissance des logos, nous avons développé une architecture CNN spécifiquement conçue pour cette tâche. Le système que nous proposons vise à fournir une solution efficace et fonctionnelle en temps réel.

Dans le deuxième chapitre de notre mémoire, nous avons abordé les notions générales des systèmes de contrôle d'accès. Notre attention s'est particulièrement portée sur les diverses méthodes de détection et de reconnaissance des caractéristiques des véhicules, dans le but de faciliter l'automatisation des accès.

Dans le troisième chapitre, nous avons présenté l'architecture globale du système proposé, ainsi que les différentes méthodes utilisées pour sa conception et son évaluation.

Dans le quatrième chapitre, nous avons abordé les divers outils nécessaires à la réalisation d'un tel système. De plus, nous avons présenté et discuté les résultats obtenus, accompagnés de graphiques pertinents. Pour évaluer les performances de détection des plaques d'immatriculation (PI) et des logos, nous avons utilisé des métriques telles que la perte (loss), la précision, le rappel et le Mean Average Precision (mAP). Pour les modèles de reconnaissance de caractères et de logos, nous avons utilisé des métriques d'évaluation telles que l'exactitude (ACC), la perte, la précision, le rappel et le score F1.

Les résultats obtenus pour le modèle de reconnaissance des chiffres sont remarquables, avec un taux d'exactitude de 99,31%, une perte de 0,02, une moyenne de F1Score de 0,99, une précision de 0,993 et un rappel de 0,993. Pour ce qui est de la reconnaissance des logos, les résultats sont les suivants : (i) un taux d'exactitude de 97,05%, (ii) un F1Score, une précision et un rappel de 0,97, et (iii) une perte de 1,0104. Les performances obtenues pour les deux modèles

de reconnaissance, soit 99,3% et 97,05%, sont très satisfaisantes et confirment la fiabilité de notre système.

Afin de vérifier la segmentation, nous avons soumis notre méthode à un large échantillon de plus de 90 images, et les résultats obtenus sont extrêmement prometteurs. La segmentation s'est avérée précise et cohérente, offrant des contours nets et une séparation précise des caractères de la PI.

Notre méthode possède l'avantage d'être :

- (i) Rapide et efficace
- (ii) Implémentable en temps réel.

Tout au long de ce travail, nous avons été confrontés à diverses difficultés et obstacles qui ont enrichi notre expérience. Parmi ceux-ci, nous avons rencontré des défis liés à la collecte de données pour notre base de données, ainsi que des contraintes liées à l'utilisation de nos ordinateurs, qui sont plus anciens par rapport aux dernières technologies. De plus, la limitation des ressources GPU dans Google Colab a également été un défi à surmonter.

Malgré ces difficultés, ce projet s'est avéré extrêmement bénéfique pour nous. Il nous a permis d'acquérir de nouvelles connaissances et d'améliorer celles que nous avons déjà acquises. En particulier, nous avons pu approfondir nos compétences dans le domaine du traitement d'images, en explorant des techniques avancées de segmentation, de détection et de reconnaissance.

## **5.2 Travaux futur**

A l'issue des travaux menés dans le cadre de ce mémoire, nous citons quelques perspectives qui peuvent être réalisées :

- Ajouter à notre système de classification de logos une classe de rejet car notre modèle ne traite que 40 logos et si nous testons un logo qui ne fait pas partie de notre base de données notre classifieur prédit une classe parmi les 40 et ne précise pas que le logo ne fait pas partie de notre BD

- Ajouter une reconnaissance faciale des employées, cela est utile quand une employée change de véhicule et n'avertit pas l'entreprise ou quand il pleut des cordes et la reconnaissance de PI échoue.

- Améliorer le système de ALPR pour des images prises à partir de caméras de surveillance ou des images aéroporté.

**Bibliographie**

- [1] Physical security access control systems | greetly. (s. d.). Consulté 12 Décembre 2022, à l'adresse <https://www.greetly.com/blog/physical-security-access-control-systems>.
- [2] ZERKOUK, M. (2015). Modèles de contrôle d'accès dynamiques (Thèse de Doctorat, Université des sciences et technologies, Oran)..
- [3] Razier, M. (2022, décembre 20). Contrôle d'accès au parking d'entreprise : Lequel privilégier? Sharvy. Consulté 23 Décembre 2022 à l'adresse <https://www.sharvy.com/blog/controle-acces-au-parking-dentreprise-comment-choisir/>.
- [4] Li, S., & Chen, Y. (2011, June). License plate recognition. In 2011 International Conference on Computer Applications and Industrial Electronics (ICCAIE 2011).
- [5] Trends and developments in vehicle access control. (s. d.). Nedap Identification Systems. Consulté 23 Décembre 2022, à l'adresse <https://www.nedapidentification.com/insights/trends-and-developments-in-vehicle-access-control/>.
- [6] Jin, B., Prasad, R. R., Angelini, B., & Rabinowitz, J. Vehicle License Plate and Color Recognition using Computer Vision.
- [7] Wu, C., Xu, S., Song, G., & Zhang, S. (2018). How many labeled license plates are needed?. In Pattern Recognition and Computer Vision: First Chinese Conference, PRCV 2018, Guangzhou, Chine, 23-26 Novembre, 2018, Acte , Partie IV 1 (pp. 334-346), Springer International Publishing.
- [8] Zéboudj, R. (1988). Filtrage, seuillage automatique, contraste et contours: du pré-traitement à l'analyse d'image (Doctoral dissertation, Saint-Etienne).
- [9] Ramdane, M. & Djarer, M. 2019. " Contrôle intelligent par réseaux de neurone pour la détection d'anomalie en radiothérapie".thèse de Master. Université Saad Dahleb Blida 1,Blida.
- [10] Nguyen, K. L., Almhdie-Imjabbar, A., Toumi, H., Jennane, R., & Lespessailles, E. (2020). Combinaison de la texture trabéculaire osseuse et des réseaux de neurones convolutifs pour la prédiction de la progression de la gonarthrose:, données des cohortes de l'OsteoArthritis Initiative et de la Multicenter Osteoarthritis Study. *Revue du Rhumatisme* , 87, A90.
- [11] Dubey, S. R., Singh, S. K., & Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*.
- [12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778)..
- [13] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In Computer Vision–ECCV 2016: 14th European



- Conference, Amsterdam, The Netherlands, 11-14 Octobre, 2016, Acte, Part I 14 (pp. 21-37). Springer International Publishing.
- [14] Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).
- [15] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.
- [16] Zhiqiang, W., & Jun, L. (Juillet 2017). A review of object detection based on convolutional neural network. In 2017 36th Chinese control conference (CCC) (pp. 11104-11109). IEEE.
- [17] rédac, T. (2021, avril 27). VGG : En quoi consiste ce modèle ? Daniel vous dit tout ! Formation Data Science | DataScientest.com. Consulté 10 janvier 2023 à l'adresse <https://datascientest.com/quest-ce-que-le-modele-vgg>.
- [18] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25.
- [19] Karandish, F. (2019, septembre 10). Comprendre la reconnaissance optique de caractères : Le guide. Moov AI. Consulté 20 janvier 2023, à l'adresse <https://moov.ai/fr/blog/reconnaissance-optique-de-caracteres-ocr>.
- [20] Qu'est-ce que l'OCR ? – La reconnaissance optique de caractères expliquée – AWS. (s. d.). Amazon Web Services, Inc. Consulté 20 janvier 2023, à l'adresse <https://aws.amazon.com/fr/what-is/ocr/>.
- [21] Reconnaissance optique de caractères (Ocr) – Définir Tech. (2021, mars 29). Consulté 20 janvier 2023, à l'adresse <https://definir-tech.com/reconnaissance-optique-de-caracteres-ocr/>.
- [22] Provost, F., & Kohavi, R. (1998). On applied research in machine learning. MACHINE LEARNING-BOSTON-, 30, 127-132.
- [23] Gaur, L., Bhandari, M., Razdan, T., Mallik, S., & Zhao, Z. (2022). Explanation-driven deep learning model for prediction of brain tumour status using MRI image data. Frontiers in genetics, 13, 448.
- [24] Nachef, A. & Mekid, H. (2022). "Design and Implementation of an Intelligent & Real-Time Vehicle Access Control System". (Thèse de Master, Université de Saad Dahlab-Blida 1)..
- [25] Massiot, A. (2020, mai 5). NLP : Une classification multilabels simple, efficace et interprétable. OCTO Talks ! Consulté 02 Février 2023, à l'adresse <https://blog.octo.com/nlp-une-classification-multilabels-simple-efficace-et-interpretable/>..

- [26] Abtahi, F., Zhu, Z., & Burry, A. M. (2015, May). A deep reinforcement learning approach to character segmentation of license plate images. En 2015 dans 14th IAPR international conference on machine vision applications (MVA) (pp. 539-542). IEEE..
- [27] Selmi, Z., Halima, M. B., & Alimi, A. M. (2017, Novembre). Deep learning system for automatic license plate detection and recognition. En 2017 dans 14th IAPR international conference on document analysis and recognition (ICDAR) (Vol. 1, pp. 1132-1138)..
- [28] Selmi, Z., Halima, M. B., & Alimi, A. M. (2017, November). Deep learning system for automatic license plate detection and recognition. En 2017 dans 14th IAPR international conference on document analysis and recognition (ICDAR) (Vol. 1, pp. 1132-1138).. IEEE.
- [29] Jing, Y., Youssefi, B., Mirhassani, M., & Muscedere, R. (2017, Avril). An efficient FPGA implementation of optical character recognition for license plate recognition. En 2017 30th Canadian Conference on Electrical and Computer Engineering (pp. 1-4).
- [30] Dias, C., Jagetiya, A., & Chaurasia, S. (2019, Septembre). Anonymous vehicle detection for secure campuses: A framework for license plate recognition using deep learning., En 2019 2nd international conference on intelligent communication and computational techniques (pp. 79-82).IEEE.
- [31] Batra, P., Hussain, I., Ahad, M. A., Casalino, G., Alam, M. A., Khalique, A., & Hassan, S. I. (2022). A novel memory and time-efficient ALPR system based on YOLOv5. *Sensors*, 22(14), 5283..
- [32] KUNDURACI, M. F., & ÖRNEK, H. K. (2019). Vehicle Brand Detection Using Deep Learning Algorithms. Dans *International Journal of Applied Mathematics Electronics and Computers*, 7(3), 70-74..
- [33] COCKPIT. (2021, juillet 30). Normes des plaques d'immatriculation algériennes (MÀJ janvier 2023). COCKPIT. Consulté 10 Mars 2023, à l'adresse <https://www.cockpitz.com/post/normes-des-plaques-d-immatriculation-algeriennes>.
- [34] B.Mourad, "Elaboration d'un système de détection et de reconnaissance des plaques minéralogiques algériennes", Thèse de Master, Université Mohamed Sadik Benyahia de Jijel , Jijel,2019..
- [35] Bensouilah, M. 2019. Elaboration d'un système de détection et de reconnaissance des plaques minéralogiques algériennes", Thèse de Master, Université Mohamed Sadik Benyahia de Jijel , Jijel.
- [36] Solawetz, J. (2023). What is YOLOv8? The Ultimate Guide..
- [37] Salinas Herrera, D. F., & Jácome Galarza, L. R. T. (2022). Implementación de un sistema de visión artificial para el reconocimiento de vehículos y automatización del libro diario de registro para la empresa lavadora LISTOCAR (Thèse de doctorat, LOJA/UIDE).
- [38] Larsson, S., & Mellqvist, F. (2019). Automatic Number Plate Recognition for Android.

- [39] Welcome to Python.org. (2023, juillet 1). Python.Org. Consulté 02 juin 2023, à l'adresse <https://www.python.org/>.
- [40] Visual studio code. (2023). In Wikipedia. Consulté 1 juin 2023, à l'adresse [https://en.wikipedia.org/w/index.php?title=Visual\\_Studio\\_Code&oldid=1153753514](https://en.wikipedia.org/w/index.php?title=Visual_Studio_Code&oldid=1153753514).
- [41] What is google drive? | definition from techtarget. (s. d.). Mobile Computing. Consulté 01 juin 2023, à l'adresse <https://www.techtarget.com/searchmobilecomputing/definition/Google-Drive>.
- [42] Qt designer manual. (s. d.). Consulté 22 Mai 2023, à l'adresse <https://doc.qt.io/qt-6/qt designer-manual.html>.
- [43] What is arduino? | arduino documentation. (s. d.). Consulté 06 juin 2023, à l'adresse <https://docs.arduino.cc/learn/starting-guide/whats-arduino>.
- [44] Tensorflow. (s. d.). TensorFlow. Consulté 06 juin 2023, à l'adresse <https://www.tensorflow.org/?hl=fr>.
- [45] Keras. (2023). In Wikipedia. Consulté 06 juin 2023, à l'adresse <https://en.wikipedia.org/w/index.php?title=Keras&oldid=1146575076>.
- [46] What is NumPy? —NumPy v1.25 Manual. (s. d.). Consulté 06 juin 2023, à l'adresse <https://numpy.org/doc/stable/user/whatisnumpy.html>.
- [47] Home. (s. d.). OpenCV. Consulté 06 juin 2023, à l'adresse <https://opencv.org/>.
- [48] Scikit-learn : Machine learning in python—Scikit-learn 1. 3. 0 documentation. (s. d.). Consulté 06 juin, à l'adresse <https://scikit-learn.org/stable/>.
- [49] Pyqt—Quick guide. (s. d.). Consulté 06 juin 2023, à l'adresse [https://www.tutorialspoint.com/pyqt/pyqt\\_quick\\_guide.htm](https://www.tutorialspoint.com/pyqt/pyqt_quick_guide.htm).
- [50] Rosebrock, A. (2020, septembre 14). Getting started with easyocr for optical character recognition. PyImageSearch. Consulté 04 juin 2023, à l'adresse <https://pyimagesearch.com/2020/09/14/getting-started-with-easyocr-for-optical-character-recognition/>.
- [51] Pytorch. (s. d.). Consulté 06 juin 2023, à l'adresse <https://www.pytorch.org>.