

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



Université Abderrahmane Mira Béjaïa
Faculté des Sciences Exactes
Département d'Informatique

Projet Fin de cycle Master

En vue de l'obtention du Diplôme de Master en Informatique

Option : Système d'information avancé

Présenté par :

SLIMANI Hamza

HAMANE Rafik

Thème

Conception et réalisation d'un système de détection et de reconnaissance des plaques d'immatriculation de camions.

Cas d'étude : Bejaia Méditerranéan Terminal (BMT)

Présidente : Mme. TAHAKOURTE Zineb

Encadrant : M. OUZEGGANE Redouane

Examineur : M. ATMANI Mouloud

Année universitaire : 2022-2023

REMERCIEMENTS

Avant tout propos, louange à Dieu tout puissant qui nous a donné la force, le courage, l'espoir nécessaire et la patience pour accomplir ce travail et surmonter l'ensemble des difficultés rencontrées.

En guise de reconnaissance, nous adressons nos sincères remerciements à notre encadreur Mr OUZEGGANE Redouane pour son soutien, sa disponibilité, ses précieux conseils et son aide tout au long de l'élaboration de ce travail.

Nous remercions les membres de jury de leur attention et intérêt portés

Enfin, nous nous acquittons, volontiers d'un devoir de gratitude et de remerciements à nos chères familles et à tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce projet.

DEDICACES

Je dédie cet événement marquant de ma vie :

À mon très cher père, mon support dans ma vie qui a toujours répondu présent dans les moments les plus difficiles, son soutien et son encouragement m'ont toujours donné la force. Autant de phrases et d'expressions aussi éloquents soit-elles ne sauraient exprimer ma reconnaissance. Que ce travail traduit ma gratitude et mon affection.

À ma très chère mère, la lumière de ma vie celle qui m'a arrosé de tendresse d'affection et d'espoirs tout au long de mon parcours, quoi que je fasse ou dise, je ne saurai point te remercier comme il se doit. Tu n'as cessé de me soutenir et de m'encourager durant toutes les années de mes études, tu as toujours été présente à mes côtés pour me consoler quand il fallait.

Je vous dois ce que je suis aujourd'hui et ce que je serai demain et je ferai toujours de mon mieux pour rester votre fierté. Que Dieu le tout puissant vous préserve, vous accorde santé, bonheur et vous protège.

À mon très chère frère, mon cher frère, les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je porte pour toi , tu as partagé avec moi les meilleurs moments de ma vie, aux moments les plus difficiles, tu as toujours été à mes côtés. Puisse dieu te donne santé, bonheur, courage et surtout réussite.

Rafik HAMANE

Je dédie cet événement marquant de ma vie :

À mon très cher père, mon support dans ma vie qui a toujours répondu présent dans les moments les plus difficiles, son soutien et son encouragement m'ont toujours donné la force. Autant de phrases et d'expressions aussi éloquentes soit-elles ne sauraient exprimer ma reconnaissance. Que ce travail traduit ma gratitude et mon affection.

À ma très chère mère, la lumière de ma vie celle qui m'a arrosé de tendresse d'affection et d'espoirs tout au long de mon parcours, quoi que je fasse ou dise, je ne saurai point te remercier comme il se doit. Tu n'as cessé de me soutenir et de m'encourager durant toutes les années de mes études, tu as toujours été présente à mes cotés pour me consoler quand il fallait.

Je vous dois ce que je suis aujourd'hui et ce que je serai demain et je ferai toujours de mon mieux pour rester votre fierté. Que Dieu le tout puissant vous préserve, vous accorde santé, bonheur et vous protège.

À mon très chère frère, mon cher frère, les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je porte pour toi , tu as partagé avec moi les meilleurs moments de ma vie, aux moments les plus difficiles, tu as toujours été à mes côtés. Puisse dieu te donne santé, bonheur, courage et surtout réussite.

À mes très chères Soeurs, mes chers frères, les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je porte pour vous, vous avez partagé avec moi les meilleurs moments de ma vie, aux moments les plus difficiles, vous avez toujours été à mes côtés. Puisse Dieu vous donne santé, bonheur, courage et surtout réussite.

SLIMANI Hamza

TABLES DES MATIERES

TABLES DES MATIERES	<i>i</i>
LISTE DES FIGURES	<i>iv</i>
LISTE DES TABLEAUX	<i>vi</i>
LISTE DES ABREVIATIONS	<i>vii</i>
Introduction Générale	1
Chapitre I : Organisme d'accueil et Recueil des Besoins	4
I.1- Introduction	4
I.2- Présentation d'Organisme d'Accueil	4
I.2.1- Présentation de BMT	4
I.2.2- Valeurs de l'Entreprise	4
I.2.3- Activités de l'Entreprise	4
I.2.4- L'organisation générale des composantes et les missions des directions	5
I.3- Problématique	9
I.4- Présentation de la solution	9
I.5- Conclusion	10
Chapitre II : Détection d'objets et reconnaissance de caractères	12
I.1- Introduction	12
II.2- Vision par ordinateur	12
II .2.1 Définition de la vision par ordinateur	12
II .2.2 Débuts de la vision par ordinateur	12
II.2. 3 Domaines d'application de la vision par ordinateur	13
II .2.4 Algorithmes de la vision par ordinateur	14
II.3Détection d'objet	14
II.3.1. Modes et types de la détection d'objet	15
II.3.2 Fonctionnement de la détection d'objet	16
II.3.3. Etapes de la réalisation d'une détection d'objet	17
II.4 Réseaux de neurones	17
II.4.1 Premiers neurones artificiels	18
II.4.2Perceptron	20
II.4.3Perceptron multicouches	21
II.4.4 Réseaux de neurones convolutifs (CNN)	22
II.5 Reconnaissance optique de caractères (OCR)	25
II.5.1 Fonctionnement de l'OCR	26
II.5.2 Types de l'OCR	28

II.6 Conclusion	29
Chapitre III : Techniques et modèle utilisés	31
III.1- Introduction	31
III.2 Description détaillée de la solution proposée	31
III.2.1 Détection de la zone d'intérêt (Plaque d'immatriculation)	31
III.3.2 Application de l'OCR sur la zone d'intérêt	39
III.4. Conclusion	40
Chapitre IV : Analyse & conception de l'application Web	42
IV.1- Introduction	42
IV.2 Présentation des acteurs du système	42
IV.3 Cas d'utilisation par acteur	43
IV. 3.1 Identification des cas d'utilisation	43
IV. 3.2 Description textuelle des cas d'utilisation	44
IV. 3.3 Diagramme de cas d'utilisation global	48
IV.4 Diagrammes de séquences	49
IV.4-1 Diagramme de séquence pour le cas d'utilisation « S'authentifier »	49
IV.4-2 Diagramme de séquence pour le cas d'utilisation « Consulter les entrées/Sorties »	50
IV.4-3 Diagramme de séquence pour le cas d'utilisation « Effectuer une recherche»	51
IV.5- Diagramme de classe du domaine	51
IV. 5.1 Diagramme de classe	51
IV. 5.3 Dictionnaire de données	52
IV. 5.4 Schéma Relationnel	53
IV. 6- Conclusion	53
Chapitre V : Phase de réalisation et tests	55
V.1- Introduction	55
V.2-Système de détection d'objet	55
V.2.1 Environnement de Programmation, Bibliothèques & Outils	55
V.2.2 Etapes d'entraînement de YOLO	56
V. 3- Le système de reconnaissance de caractères	59
V.3.1 Environnement de Programmation, Bibliothèques & Outils	59
V.3.1 Etapes de préparation	59
V.4- Application web	60
V.4.1-Environnement de Programmation, Bibliothèques & Outils	60
V.4.2 Interfaces d'application	61
V.5 - Conclusion	62
Conclusion Générale	64
Bibliographie	66

LISTE DES FIGURES

Figure I - 1 Organigramme de BMT	7
Figure I -2 Schéma de la solution proposée	10
Figure II – 1 Résultat d'une détection d'une voiture avec son étiquette [28].	15
Figure II – 2 Matching Learning vs Deep Learning [29].	16
Figure II – 3 Etapes de détection avec l'approche du Deep Learning	16
Figure II – 4 Neurone biologique	18
Figure II – 5 Neurone biologique avec modélisation.....	19
Figure II – 6 Etape d'agrégation[30].	19
Figure II – 7 Etapes de la modélisation du fonctionnement d'un neurone biologique.....	20
Figure II – 8 Formule de mise à jour des paramètres W	21
Figure II – 9 Représentation graphique des sorties finales en fonction des x_1, x_2	21
Figure II – 10 Deux couches de perceptrons	22
Figure II – 11 Représentation graphique des sorties finales en fonction des x_1, x_2	22
Figure II – 12 Ensemble de pixel d'une image	23
Figure II – 13 Architecture d'un réseau de neurone convolutif	25
Figure II – 14 Classement des zones par OCR.....	26
Figure II – 15 Le glyphe de la lettre 'a'.....	27
Figure II – 16 Le fonctionnement de l'OCR.....	28
Figure II – 17 La reconnaissance caractère par caractère	28
Figure II – 18 La reconnaissance mot par mot	29
Figure III – 1 Lieu des entrées sorties du port de Béjaia	32
Figure III – 2 Histogramme qui représente le nombre d'images traitées en une seconde par des algorithmes de détections.....	33
Figure III – 3 Architecture de YOLO	34
Figure III – 4 Résultat d'une détection avec YOLO	34
Figure III – 5 Image divisée en grille	35
Figure III – 6 Les cellules avec leurs valeurs de probabilités	36
Figure III – 7 Présentation des valeurs du vecteur Y	37
Figure III – 8 Etapes d'application de l'IOU	38
Figure III – 9 Application de l'IOU.....	38
Figure III – 10 Les différentes zones des caractères dans une plaque d'immatriculation	40
Figure IV – 1 Diagramme de cas d'utilisation	48
Figure IV – 2 Diagramme de séquence système pour le cas d'utilisation <s'authentifier>	49
Figure IV – 3 Diagramme de séquence système pour le cas d'utilisation <Consulter les entrées/sorties>	50
Figure IV – 4 Diagramme de séquence système pour le cas d'utilisation <Effectuer une recherche>	51

Figure IV – 5 Diagramme de classe.....	52
Figure V- 1 Ensemble d'images du dataset _____	56
Figure V- 2 Contenu d'un fichier d'annotation _____	57
Figure V- 3 Contenu du fichier dataset.yaml _____	57
Figure V- 4 Commande d'imporation du fichier best.onnx _____	58
Figure V- 5 Commande d'installation de pytesseract _____	59
Figure V- 6 Résultat d'une détection d'une plaque d'immatriculation pour une voiture _____	60
Figure V- 7 Résultat de la reconnaissance _____	60
Figure V- 8 Page d'authentification _____	61
Figure V- 9 Page d'affichage de matricules _____	62

LISTE DES TABLEAUX

Tableau IV – 1 Acteurs avec leurs rôles _____	43
Tableau IV – 2 Cas d'utilisation _____	44
Tableau IV – 3 Cas d'utilisation « S'authentifier » _____	45
Tableau IV – 4 Cas d'utilisation « Consulter les entrées /sorties» _____	46
Tableau IV – 5 Cas d'utilisation « Modifier les entrées /sorties» _____	47
Tableau IV – 6 Cas d'utilisation « Effectuer des recherches» _____	48
Tableau IV – 7 Dictionnaire de donnée _____	53

LISTE DES ABBREVIATIONS

BMT	Bejaia Mediterranean Terminal
CNN	Convolutional Neural Network
CPE	Conseil des Participations de l'Etat
DO	Direction des Opérations
EPB	Entreprise Portuaire de Bejaia
ICR	Intelligent Character Recognition
IOU	Intersection Over Union
NMS	Non Max Suppression
OCR	Optical Character Recognition
PSEPORTEK	Systems and Equipment
R-CNN	Region-Based Convolutional Neural Networ
UML	Unified Modeling Language
UP	Non Max Suppression
VOC	Visual Object Classes
XML	Extensible Markup Language
XP	eXtrem Programming
YOLO	You Only Look Once

INTRODUCTION

GÉNÉRALE

Introduction Générale

L'informatique, dans nos jours, est très importante dans la gestion et l'organisation de travail dans tous types d'organisme. Ça permet d'automatiser l'échange d'information, de capter les données importantes, de les stocker, transformer et les diffuser aux acteurs concernés. Par la suite, vient l'analyse de ces données à travers des outils d'analyse de données d'aide à la décision.

Dans le monde des entreprises, la sécurité et la traçabilité des opérations physiques sont très importantes, pour cela, les techniques de détection et de reconnaissance d'objet, et l'interprétation sémantique de ces objets est indispensable pour tout organisme.

Dans ce contexte, nous avons effectué un stage au niveau de Bejaia Medeteranean Terminal (BMT) afin de remédier une problématique importante, à savoir, la traçabilité des opérations d'entrée / sortie des camions de marchandise. L'efficacité de ces opérations portuaires sont freinées par la saisie et le suivi manuels entrées / sorties des camions.

Ainsi, l'automatisation des processus de suivi et de contrôle des camions aux entrées et sorties, revêt une signification essentielle en vue d'accélérer les flux de marchandises, d'améliorer la sécurité, et de garantir la conformité. Dans ce cadre, nous allons proposer une solution basée sur la détection et la reconnaissance des objets, ainsi que la lecture d'information numérique à travers la méthode OCR.

En plus de ça, nous allons concevoir et réaliser une application web qui permet à l'administrateur de consulter et de corriger les informations enregistrées par le Système de détection de reconnaissance des plaques d'immatriculation.

Pour bien présenter notre travail, ce mémoire est structuré en cinq à chapitres, à savoir :

- Le premier chapitre est consacré à la présentation de l'organisme d'accueil, dans lequel nous nous sommes focalisé sur la problématique du suivi manuel des entrées / sortie des camions. Par la suite, nous allons décrire notre solution et ses différentes composantes.

- En ce qui concerne le deuxième chapitre, nous allons passer en revue les notions et concepts liés à la détection d'objet ainsi la reconnaissance de caractères.

- En ce qui concerne les techniques et modèle de détection que nous choisirons pour réaliser notre solution, ceci fera partie du troisième chapitre.

- Le chapitre quatre est dédié à l'analyse et la conception de l'application Web, dans lequel nous avons établi différents diagrammes essentiels d'UML en suivant une démarche à mi-chemin entre XP et UP. Ce chapitre est conclu par un schéma relationnel.

- Le dernier chapitre sera consacré à la phase de réalisation de notre solution composée de deux parties :

 Système de détection d'objets et de reconnaissance de caractères, et une application Web qui permet d'afficher, via une interface Web, les différentes opérations d'entrées/sorties des camions détectés.

CHAPITRE I

ORGANISME D'ACCUEIL & RECUEIL DES BESOINS

CHAPITRE I

ORGANISME D'ACCUEIL ET RECUEIL DES BESOINS

I.1- Introduction

Dans ce premier chapitre nous allons présenter l'organisme d'accueil 'BMT', où nous avons effectué notre stage, ses différentes activités, son organigramme, ces missions et services. Sur cette base, nous allons établir la problématique à traiter ainsi qu'une description de la solution proposée.

I.2- Présentation d'Organisme d'Accueil

I.2.1- Présentation de BMT

BMT est une jointe venture entre l'Entreprise Portuaire de Bejaia (EPB) et PORTEK Système and Equipment (PSE), une société Singapourienne. BMT est spécialisée dans la gestion et l'exploitation du Terminal à conteneurs du port de Bejaia. Elle a été créée sur décision du Conseil des Participations de l'Etat (CPE) en Mai 2004 [1].

I.2.2- Valeurs de l'Entreprise

Les valeurs d'une entreprise sont les différents types de principes moraux et sociétaux sur lesquels elle se base pour aller de l'avant et prendre des décisions stratégiques qui vont avoir un impact sur le déroulement de l'entreprise [2].

- l'innovation
- la courtoisie
- l'intégrité
- productivité
- la sécurité

I.2.3- Activités de l'Entreprise

L'activité principale de BMT est la gestion et l'exploitation du Terminal à conteneurs. Sa mission principale est de traiter dans les meilleures conditions de

délais, de coûts et de sécurité, l'ensemble des opérations qui ont un rapport avec le conteneur. Pour ce faire, elle s'est dotée d'équipements performants et de systèmes informatiques pour le support de la logistique du conteneur afin d'offrir des services de qualité, efficaces et fiables pour assurer une satisfaction totale des clients.

I.2.4- L'organisation générale des composantes et les missions des directions

Dans cette section, nous allons voir l'organisation et la structure générales avec leurs directions et missions [3].

a. Organigramme

La figure I.1 représente le schéma organisationnel illustrant la structure de l'entreprise BMT.

b. Missions et services de différentes composantes

Les différentes missions des services et département de l'entreprise sont indiqués ci-dessous :

- Direction générale

Dirige l'entreprise, donne des directives pour les différentes structures et assure la coordination entre les directions de l'entreprise

- Département Audit interne

Assure l'audit des procédures et mesure leur efficacité.

- Direction des Opérations (DO)

Assure la planification des escales, la détermination des ressources ; équipes et équipements ; la gestion du terminal à conteneurs. Elle prend en charge les opérations de manutentions des navires porte-conteneurs le chargement et déchargement, effectue les opérations d'acconage telles que : le suivi des livraisons, le dépotage et l'emportage des conteneurs, la mise à disposition des conteneurs vides, le traitement des conteneurs frigorifiques, et assure la sécurité au sein du terminal.

- Service Acconage

Assure la gestion des opérations au niveau du terminal.

- Service Manutention

Assure la gestion des opérations aux navires.

- Service Ressources

Assure une meilleure affectation des ressources humaines et matérielles.

- Service Logistique

Assure le suivi des moyens logistiques ainsi que la prestation logistique globale

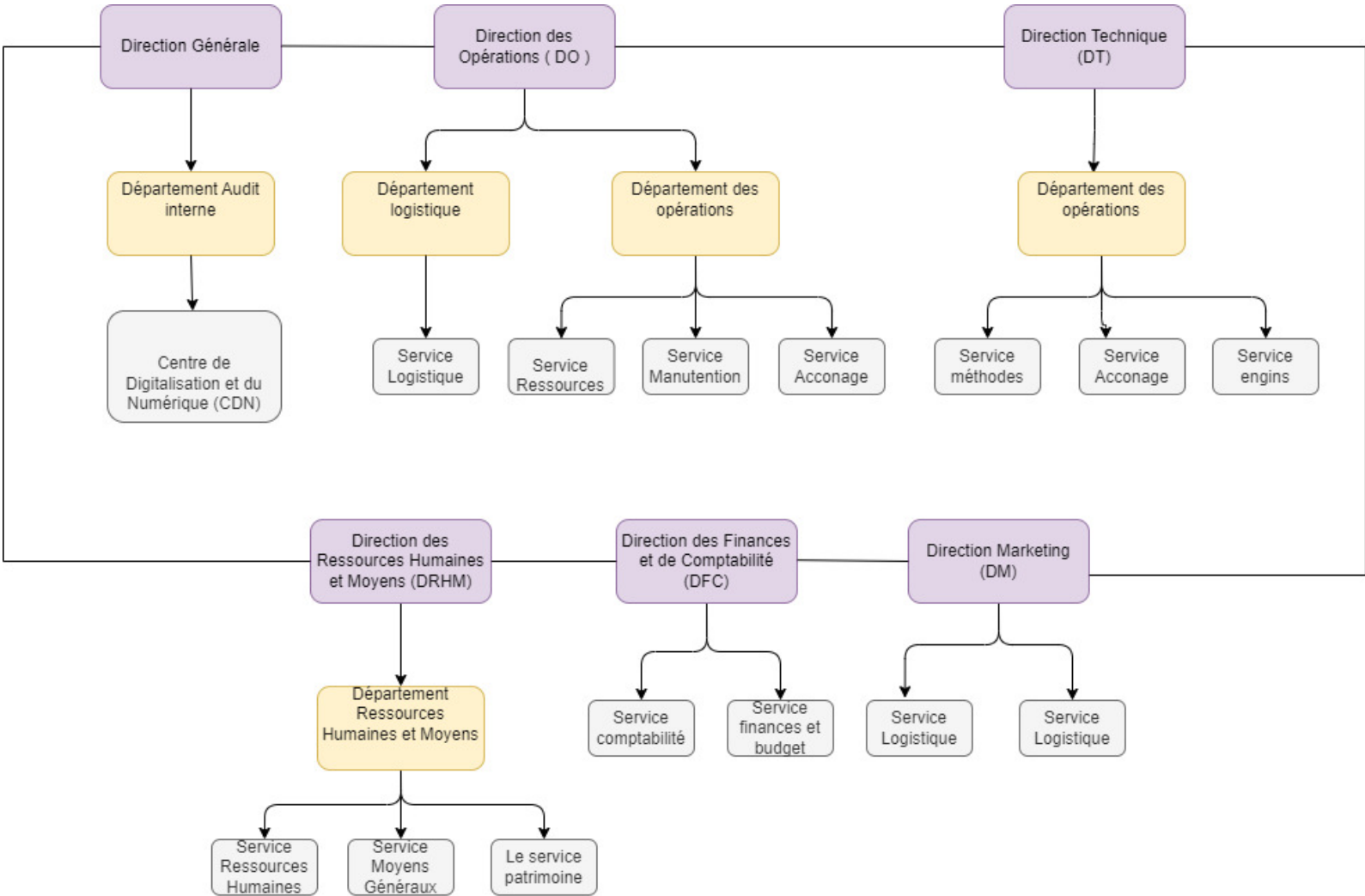


Figure I - 1 Organigramme de BMT

-Service Marketing

Assure la promotion de l'image de marque de l'entreprise et la mise en œuvre du plan marketing et commercial.

-Service Commercial

Procède à la facturation des prestations fournies et le recouvrement des créances.

-Service comptabilité

Procède au contrôle et à l'enregistrement de toutes les opérations de la société (achat, vente, investissement...).

-Service finances et budget

Assure le suivi de l'exécution du budget de la société et de la Comptabilité Analytique et la gestion de la trésorerie.

-Service engins

-Assure la maintenance des engins de manutention (Reach Stacker, MT Handler, chariots élévateurs et autres engins roulants ...)

-Service portiques

Assure la maintenance des portiques de quai et des grues mobiles.

-Service méthodes

Assure la mise en œuvre du plan de maintenance des équipements.

-Le service patrimoine

Assure la gestion des stocks et des immobilisations

-Service Moyens Généraux

Satisfaire les besoins des différentes structures en produits et prestations de services

-Service Ressources Humaines

Assure la gestion administrative du personnel et le développement des compétences

I.3- Problématique

Notre stage a été effectué au niveau de Direction Général, où nous avons identifié la problématique à traiter et qui s'agit de la gestion des entrées / sorties des camions au niveau du Port de Bejaia.

En effet, au niveau de l'entreprise BMT, la traçabilité des entrées / sorties des camions est effectuée manuelle, qui est une méthode moins efficace, surtout pour une entreprise qui reçoit quotidiennement des centaines de camions.

La procédure actuelle pour gérer les entrées / sorties des camions est comme suit : une entrée ou une sortie est notée par un agent sur papier avec les détails nécessaires comme le numéro d'immatriculation et de conteneurs. À la fin de la journée, l'agent transmet les papiers à un autre agent pour les saisir sur Excel.

Comme nous l'avons constaté, cette dernière activité nécessite beaucoup de temps et d'effort à cause de la rédaction manuelle, de plus, BMT sera exposé à un taux d'erreur de saisie et aussi un retard dans le traitement des entrées/sorties qui touchera l'entreprise et sa productivité négativement, surtout dans les périodes de fortes activités ou le nombre des camions et des conteneurs est élevés.

I.4- Présentation de la solution

Dans le cadre de notre mission, et pour remédier aux différents problèmes cités ci-dessus, nous avons proposé une solution qui consiste à automatiser la traçabilité des entrées / sorties des camions. Notre solution consiste à implémenter un système qui utilise des techniques de détection et de reconnaissance automatique des matricules qui les enregistre sur un fichier log (fichier journal), et tout ceci sera réalisé sans intervention humaine.

Notre solution sera constituée de ces parties suivantes voir (Figure I-2) :

- Un système de détection automatique des immatriculations des camions et deconteneur en temps réel avec une caméra de surveillance.
- Un système OCR(OpticalCaractère Recognition) appliqué sur les plaques d'immatriculations détectées, qui sert à reconnaître les numéros et les caractères sur la plaque et les transformer en format texte. Les différents matricules transformés en texte seront enregistrés sur un fichier journal.
- Un Dashboard (application web) qui nous permet de visualiser, et éventuellement corriger, les informations extraites à partir du fichier journal.

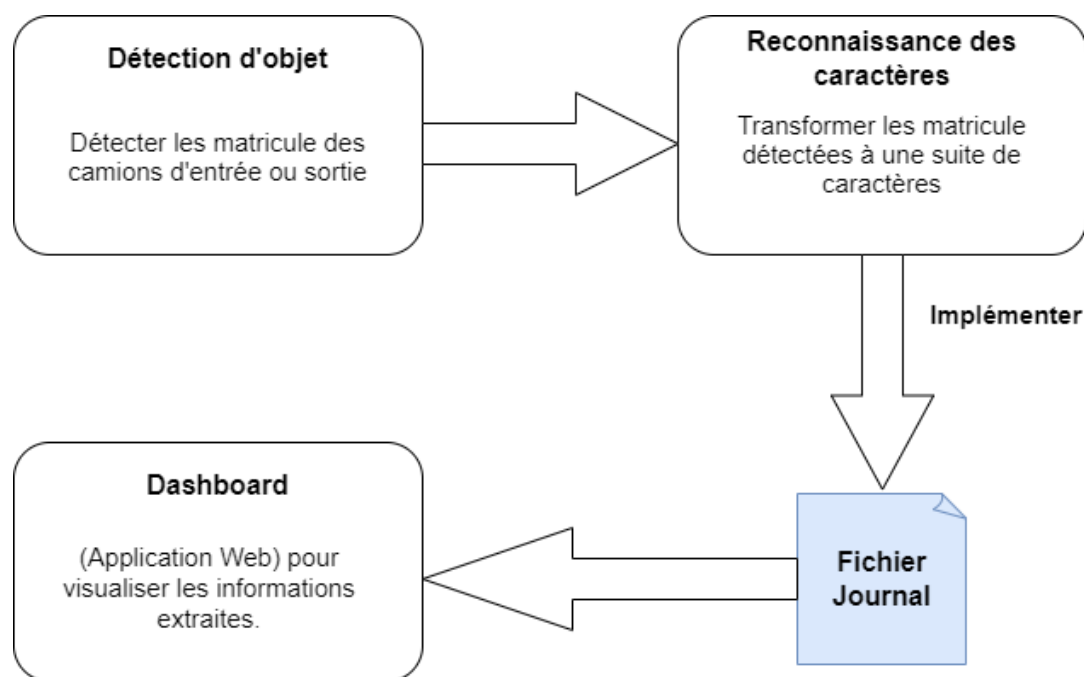


Figure I -2 Schéma de la solution proposée

I.5- Conclusion

Dans ce chapitre, nous avons présenté l'organisme d'accueil, à savoir BMT, son organisation et ses valeurs, et nous nous sommes focalisés sur la traçabilité des entrées / sorties des camions qui se fait manuellement au sein de l'entreprise. Ceci était la base de la problématique dégagée ainsi que l'idée première de notre solution qui se base sur la détection des objets et reconnaissance des caractères. Ainsi, et avant de passer à la conception et réalisation de notre solution, les concepts précédemment cités seront traités au niveau du chapitre suivant.

CHAPITRE II

DETECTION D'OBJETS ET
RECONNAISSANCE DE
CARACTERES

CHAPITRE II

DETECTION D'OBJETS ET RECONNAISSANCE DE CARACTERES

I.1- Introduction

Le nombre de données visuelles a augmenté rapidement arrivant à des chiffres très grands dans les années précédentes. Pour comprendre et utiliser ce type de données, il nous faudra des technologies spéciales qui feront cette tâche automatiquement et tout ça rentre dans le domaine de la vision par ordinateur (computer vision). Ainsi, dans ce chapitre, nous allons présenter brièvement ce domaine, et nous passerons par la suite au domaine de réseaux de neurones et leurs utilisations dans la manipulation des images (Réseaux de neurones convolutifs). Après ça, nous présenterons la détection des objets et reconnaissance du texte avec l'OCR.

II.2- Vision par ordinateur

Cette section est dédiée à la vision par ordinateur dans laquelle nous allons voir son historique, ses domaines d'application et les différents algorithmes et méthodes utilisés dans ce domaine.

II .2.1 Définition de la vision par ordinateur

La vision par ordinateur est une branche de l'intelligence artificielle permettant au système et aux ordinateurs d'acquérir les informations significatives et extraire les données à partir d'images numériques, de vidéos et d'autres entrées visuelles, et de prendre des mesures ou de formuler des recommandations sur la base de ces informations et ces données [4].

II .2.2 Débuts de la vision par ordinateur

La vision par ordinateur n'est pas une nouvelle technologie mais une ancienne qui est devenue plus développée grâce à des dizaines de recherches.

Elle commençait à l'Institut de technologie du Massachusetts dans un projet nommé 'Project on Mathematics and computation' en 1960. Ensuite, en 1966 le scientifique Marvin Minsky en collaboration avec Seymour Papert commençaient à

essayer d'appliquer l'idée de la connexion d'une caméra à un ordinateur en se basant sur les recherches précédentes ce qui a donné lieu au projet intitulé 'Summer Vision Project'.

En juillet 1966, Papert expliqua que son projet impliquait le développement d'un système qui sert à diviser, décrire et identifier les arrière-plans et les objets dans une image. Leurs tests furent sur des différentes images pleines de couleurs, de tailles de polices variantes et des différents types d'objet. Au début, le projet n'avait pas un grand succès, mais il servait au développement d'autres projets qui ont un grand effet sur le développement de la vision par ordinateur.

Arrivant au début des années 2000 où l'accès à l'internet et le pouvoir de partage des informations est devenu plus rapide, le computer vision faisait un très grand pas avec plusieurs découvertes. Le machine learning, les réseaux de neurones et le deep learning ont été ajoutés au monde de la vision par ordinateur pour s'en servir.

La première application de détection de visages en temps réel est développée en 2001. En 2005, un dataset qui est un ensemble de jeu de données est lancé sous le nom de PASCAL VOC qui pouvait être utilisé par les scientifiques dans la reconnaissance de classes d'objet. ImageNet est un dataset qui contient plus de données a été lancé juste après.

En 2012, une équipe de l'université de Toronto lancent un réseau de neurone nommé AlexNet qui réduit le taux d'erreur dans la reconnaissance des images. Depuis lors, la vision par ordinateur a connu ses plus grandes découvertes [5].

II.2. 3 Domaines d'application de la vision par ordinateur

Les domaines qui peuvent se servir de la vision par ordinateur sont nombreux, plusieurs tâches sont devenues plus faciles à être accomplies dans ces domaines.

Parmi les domaines nous trouvons [6]:

La robotique

- Détermination de la location du robot automatiquement
- Évitement d'obstacle
- Interaction humain-robot

La médecine

- Classification des cellules et la détection des tumeurs
- Reconstruction des organes humains
- Vision guidée par robot dans les opérations

Sécurité

- La biométrie (reconnaissance faciale, prise d'empreinte)
- Détection de certaines activités ou comportement lors de la surveillance

Automation industrielle

- L'assemblage
- Lecture des code-barres

II .2.4 Algorithmes de la vision par ordinateur

La vision par ordinateur est appliquée dans plusieurs domaines grâce à sa multitude d'algorithmes qui servent à réaliser plusieurs tâches. Ces algorithmes incluent des différentes méthodes qui sont utilisées pour comprendre les objets sur les images et extraire des informations de notre monde pour produire des informations numériques. Parmi les algorithmes nous trouvons [7]:

- Classification des objets.
- Identification d'objet.
- Détection d'objet.
- Segmentation d'objet.
- Vérification d'objet.
- Reconnaissance d'objets.
- Détection de points de repère d'objets

II.3Détection d'objet

Dans cette partie nous allons présenter un algorithme parmi ceux énumérés dessus, à savoir l'algorithme de détection d'objet.

La détection d'objet est une technique de vision par ordinateur qui est utilisée pour identifier et localiser des objets dans une image et vidéo. Spécifiquement, c'est le

fait de dessiner un label ou étiquette autour de cette détection ce qui nous permet de localiser cet objet en un moment précis et dans une scène donnée comme l'indique la figure suivante [8].

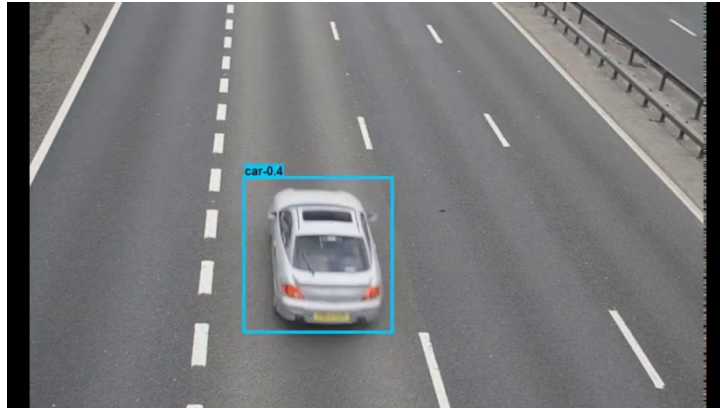


Figure II - 1 Résultat d'une détection d'une voiture avec son étiquette [28].

II.3.1. Modes et types de la détection d'objet

D'une manière générale, la détection d'objet peut être décomposée en deux approches, la première est celle de Matching Learning et l'autre de Deep Learning.

Dans les approches plus traditionnelles basées sur ML (Matching Learning), les techniques de vision par ordinateur sont utilisées pour examiner des diverses caractéristiques d'une image, telles que l'histogramme de couleur ou les bords, afin d'identifier des groupes de pixels pouvant appartenir à un objet. Ces caractéristiques sont ensuite introduites dans un modèle de régression qui prédit l'emplacement de l'objet ainsi que son étiquette.

D'autre part, les approches basées sur l'apprentissage en profondeur (Deep Learning) utilisent des réseaux de neurones convolutifs (CNN) pour performer une détection d'objets de début jusqu'à la fin, dans laquelle les caractéristiques n'ont pas besoin d'être définies et extraites séparément.

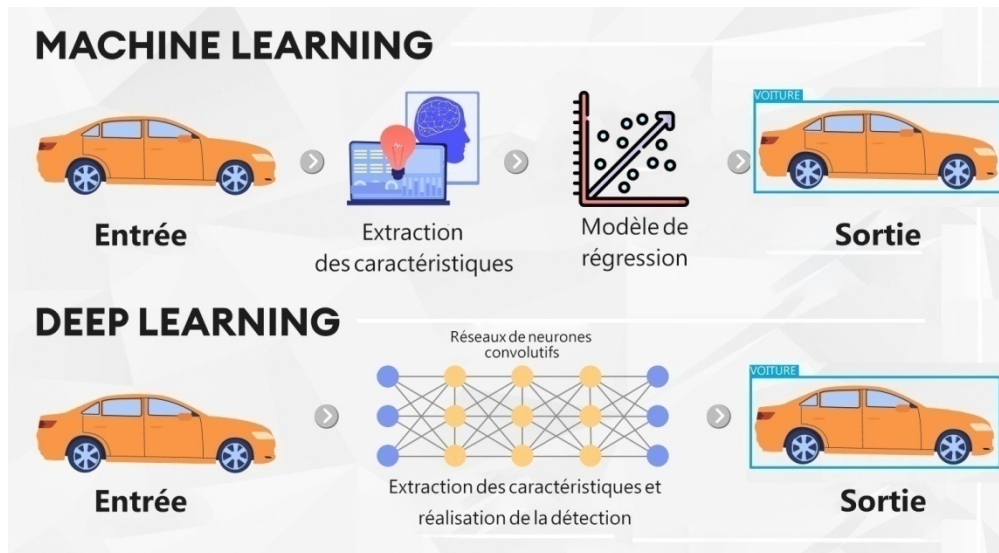


Figure II - 2Maching Learning vs Deep Learning [29].

II.3.2 Fonctionnement de la détection d'objet

Comme nous l'avons déjà indiqué, il existe deux approches pour réaliser une détection d'objet, et nous avons choisi l'approche de Deep Learning qui se base sur les réseaux de neurones convolutifs.

L'opération de la détection d'objet, par la méthode Deep Learning (voir la figure II.3), est divisée en deux parties. La première partie d'encodage, prend une image comme entrée, et à travers une série de couches et blocs, permet d'extraire des informations qui vont être utilisées dans la localisation des objets présents dans l'image. La deuxième partie permet de réaliser ledécodage des informations extraites précédemment afin de prédire l'emplacement exact et le type des objets avec leurs étiquettes.

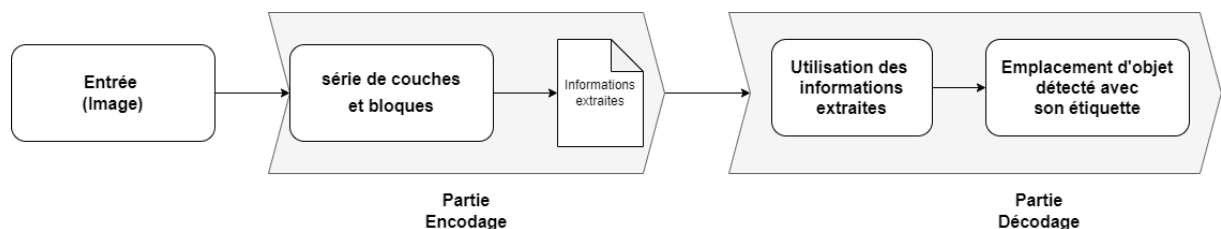


Figure II - 3Etapes de détection avec l'approche du Deep Learning

Ce mécanisme sera expliqué en détail dans les prochaines sections.

II.3.3. Etapes de la réalisation d'une détection d'objet

Pour réaliser la détection d'objets, des étapes doivent être appliquées, à savoir :

- La collecte des données : C'est de rassembler un ensemble des images ou vidéos qui contiennent les objets qu'on veut détecter.
- Annotations des données : on délimite les objets d'intérêt à l'aide des bounding boxes (Boîtes de délimitation) pour chaque image ou vidéo collectée.
- Prétraitement des données : Cela peut inclure le redimensionnement, la normalisation des couleurs...etc.
- Modèle de détection : En utilisant les différents modèles du DEEP LEARNING tels que :
 - La famille du modèle R-CNN :
 - R-CNN.
 - Fast R-CNN.
 - Faster R-CNN.
 - La famille du modèle YOLO : YOLOv1, YOLOv2, YOLOv3, ... YOLOv8.
 - Entraînement du modèle : Entraîner notre modèle en utilisant le data set, tout ça pour la précision de notre objet.
 - Validation et ajustement : Il s'agit d'analyser les résultats et apporter les ajustements.
 - Évaluation du modèle : Il s'agit d'évaluer la performance du modèle sur un Ensemble de testes.
 - Intégration du modèle dans l'application : c'est la détection.

Dans ce qui suit, nous allons présenter les réseaux de neurones, vue leur utilisation dans les modèles d'apprentissage DEEP LEARNING (famille R-CNN et famille YOLO)

II.4 Réseaux de neurones

Les réseaux de neurones sont une réflexion du fonctionnement de notre cerveau. Il nous permet de résoudre des problèmes dans le cadre de l'intelligence artificielle, Machine Learning et Deep Learning. Leur structure est basée sur une succession de

plusieurs nœuds de couches qui s'exécutent l'un après l'autre. Chaque couche exécute un certain calcul pour arriver à la résolution d'un problème.

Pour comprendre les réseaux de neurones, nous allons découvrir en premier leur histoire, et comment leur premier modèle a été construit.

II.4.1 Premiers neurones artificiels

Les premiers réseaux de neurones ont été inventés en 1943 par deux mathématiciens et neuroscientifiques Warren McCulloch et Walter Pitts inspirant de fonctionnement de neurones biologiques. En biologie, les neurones sont des cellules excitables connectées les unes aux autres et ayant pour rôle de transmettre des informations dans notre système nerveux. La figure suivante montre les compositions d'un neurone biologique.

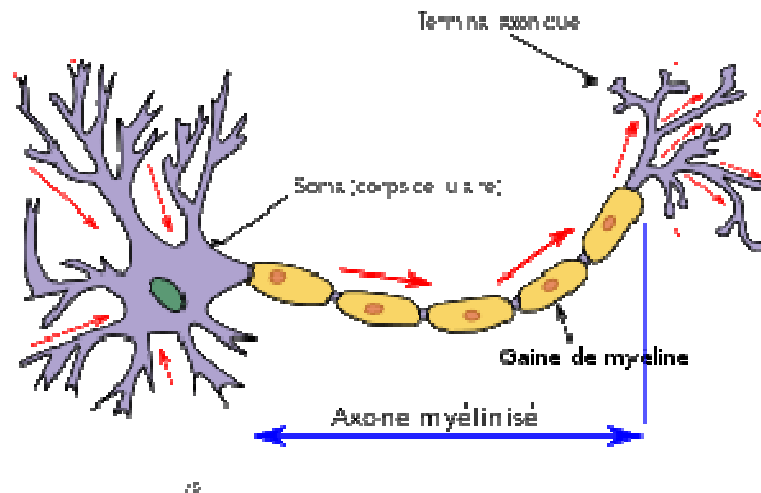


Figure II - 4 Neurone biologique

Comme nous voyons dans la figure précédente, chaque neurone se compose de plusieurs dendrites, corps cellulaire et d'un axone. Les dendrites sont en quelque sorte les portes d'entrées d'un neurone, c'est à cet endroit que le neurone reçoit des signaux lui provenant des neurones qui le précèdent. Lorsque la somme de ces signaux dépasse un certain seuil, le neurone s'active et produit alors un signal électrique, ce dernier circule le long de l'axone en direction des terminaisons axonales pour être envoyé à son tour vers d'autres neurones de notre système nerveux.

Ce que les deux scientifiques ont essayé de faire est de modéliser le fonctionnement précédent en considérant qu'un neurone pouvait être représenté

par une fonction de transfert qui prend des signaux X et qui retourne une sortie Y comme la montre la figure suivante :

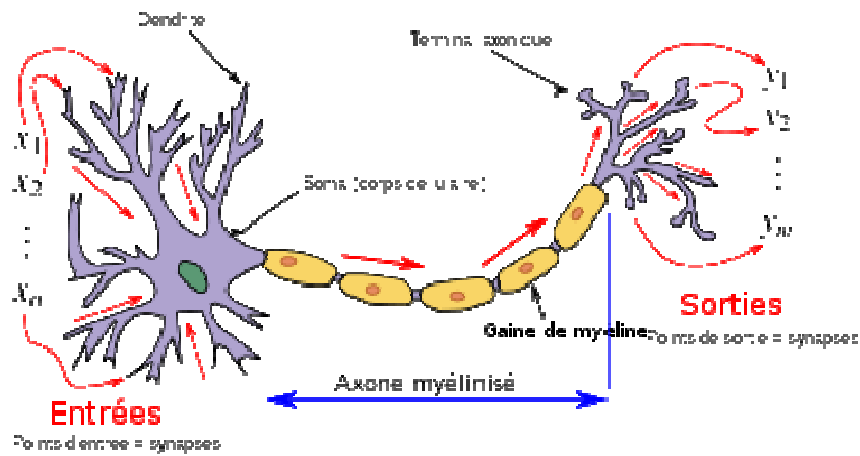


Figure II - 5 Neurone biologique avec modélisation

A l'intérieur de la fonction on trouve deux grandes étapes. La première étape est une étape d'agrégation, nous faisons la somme de toutes les entrées du neurone en multipliant chaque entrée par un coefficient w qui représente l'activité synaptique, c'est-à-dire le fait que le signal soit excitateur auquel w vaut $+1$ ou bien inhibiteur auquel il vaut -1 . Nous obtiendrons donc l'expression d'agrégation finale qui résume la première étape.

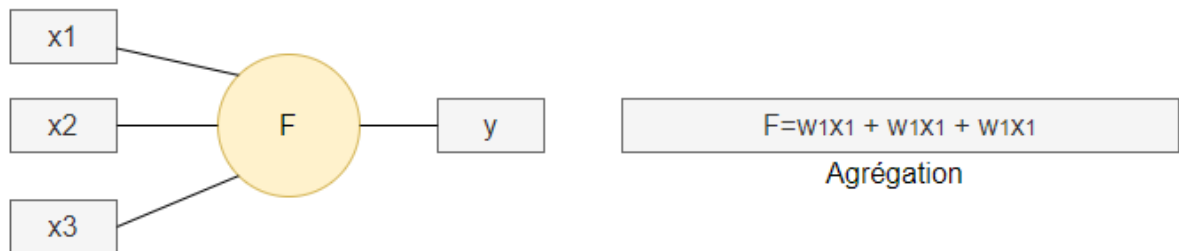


Figure II - 6 Etape d'agrégation[30].

Une fois la première étape est réalisée, nous passons à la phase d'activation. Nous regardons le résultat du calcul effectué dans la première étape et si celui-ci dépasse un certain seuil, en général 0, alors le neurone s'active et retourne une sortie $y=1$, sinon il reste à 0. Donc pour résumer ce que nous avons dit, nous appuyons sur l'illustration suivante.

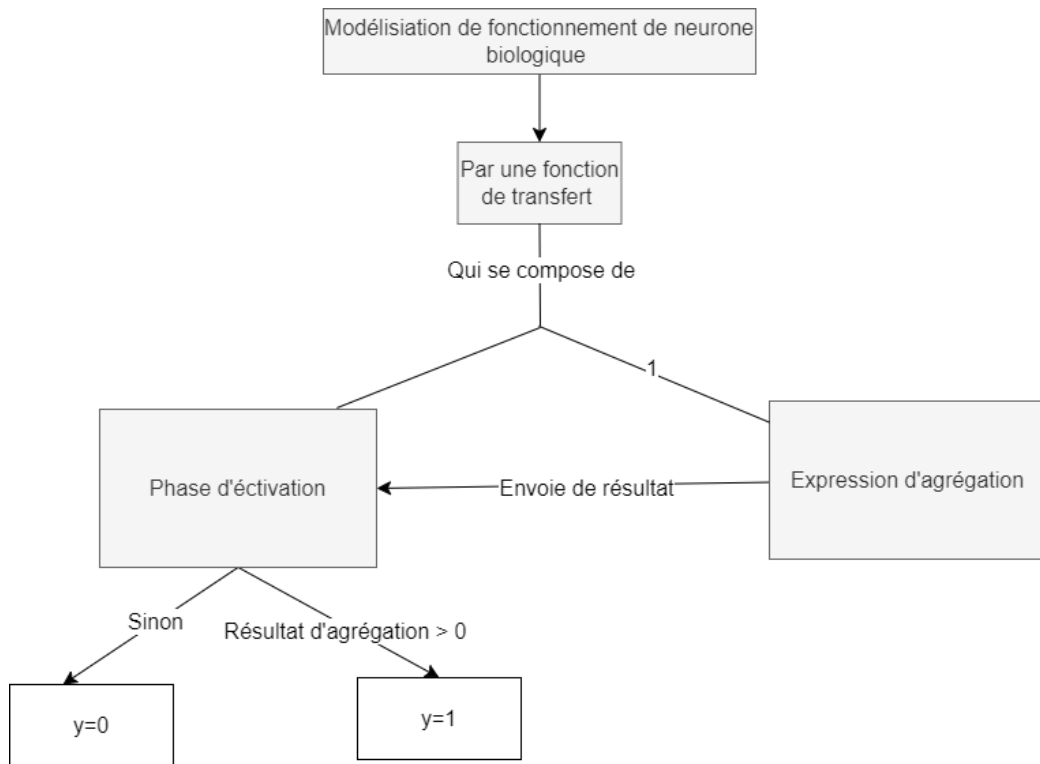


Figure II - 7 Etapes de la modélisation du fonctionnement d'un neurone biologique

Ce modèle premier des réseaux de neurones n'était conçu que pour traiter des entrées logiques qui valent soit 0 ou 1, en connectant plusieurs de ces fonctions l'une aux autres il sera possible de résoudre n'importe quel problème de logique booléenne.

II.4.2 Perceptron

Le modèle de perceptron appuie sur le modèle précédent avec une disposition d'un algorithme d'apprentissage qui lui permet de trouver les valeurs de ses paramètres W à fin d'obtenir les sorties Y qui nous conviennent. Frank Rosenblatt qui a développé cet algorithme entraînant un neurone artificiel sur des données (X,y) pour que celui-ci renforce ses paramètres W à chaque fois qu'une entrée X est activée en même temps que la sortie y présente dans ces données. La formule présente dans la figure suivante nous montre comment les paramètres W sont calculés et mis à jours.

$$W = W + \alpha (y_r - y) X$$

y_r : sortie de référence

y : sortie produite par le neurone

X : entrée de neurone

α : pas d'apprentissage

Figure II - 8 Formule de mise à jour des paramètres W

Le souci avec ce modèle est qu'il peut résoudre que des problèmes linéaires (séparer deux classes de points comme il est montré dans la figure ci dessous), alors que la majorité des phénomènes dans notre monde ne sont pas de phénomènes linéaires. Et dans ces conditions, le perceptron lui seule n'est pas très utile.

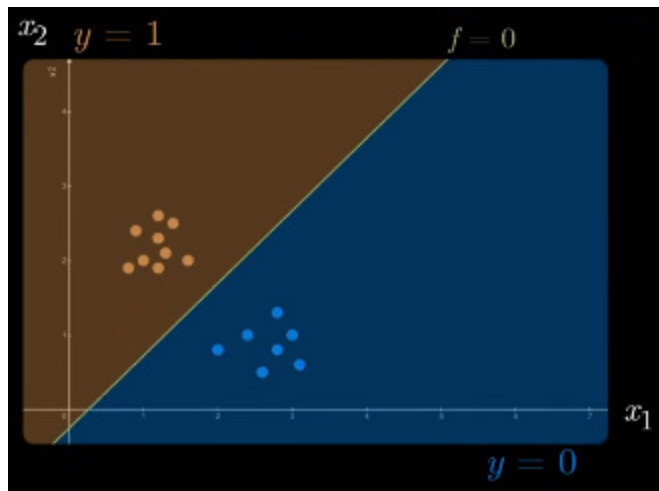


Figure II - 9 Représentation graphique des sorties finales en fonction des x_1 , x_2

II.4.3 Perceptron multicouches

Si nous connectons Plus de 2 perceptrons ensemble, nous serons capables de résoudre des phénomènes non linéaires donc plus complexes. C'est l'idée qui est derrière la multitude des couches de perceptrons. La figure suivante montre 2 couches, la première couche contient deux perceptrons et la deuxième couche contient un perceptron, ce qui réalise la multitude des couches.

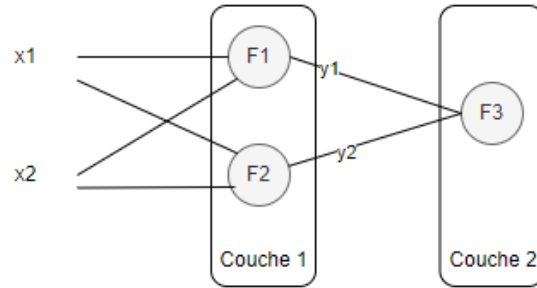


Figure II - 10 Deux couches de perceptrons

En exploitant les deux couches précédentes pour résoudre un problème de séparation de deux classes de points, nous obtiendrons la représentation graphique suivante pour les sorties finales en fonction des x_1 , x_2 qui n'est pas linéaire et bien plus intéressante.

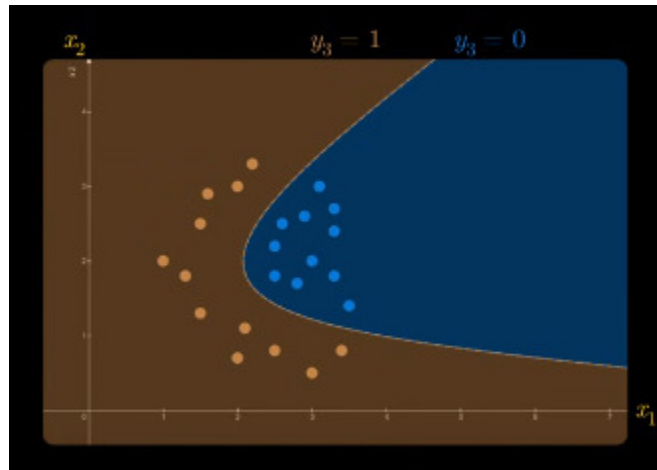


Figure II - 11 Représentation graphique des sorties finales en fonction des x_1 , x_2

II.4.4 Réseaux de neurones convolutifs (CNN)

Les réseaux de neurones convolutifs (CNN : Convolutional Neuron Network) sont une architecture particulière qui permet de manipuler des images avec un réseau de neurones. Ils sont pertinents et leur utilisation est très fréquente en Deep Learning.

Les CNN forment une architecture de réseau de neurones qui utilisent des couches de convolution et de pooling pour pouvoir construire des modèles invariants par translation et diminuer le nombre de variables.

Puisque ce type de réseaux de neurones se comporte avec des images, nous allons découvrir quelques bases sur les images numériques avant de passer aux composantes d'un réseau de neurones convolutifs.

II.4.4.1 Encodage numérique d'une image

II.4.4.1.1 Les pixels

Une image numérique est représentée par des blocs de couleurs ordonnés sous forme de tableau, où chaque bloc est un pixel. Un appareil photo, lui, dispose aussi de photorécepteurs, semblables aux cônes présents dans notre œil. La particularité est que pour chaque cellule photovoltaïque (récepteur), il y aura 3 cônes : un pour le rouge, un pour le vert et un pour le bleu. On retrouve les composantes RGB de la lumière.

Pour chaque cellule, on stocke trois valeurs, une pour chaque composante. Chaque valeur est située entre 0 et 255, car elles sont codées sur un octet. Chaque pixel est donc une liste de trois valeurs, ordonnées sous forme (RGB).

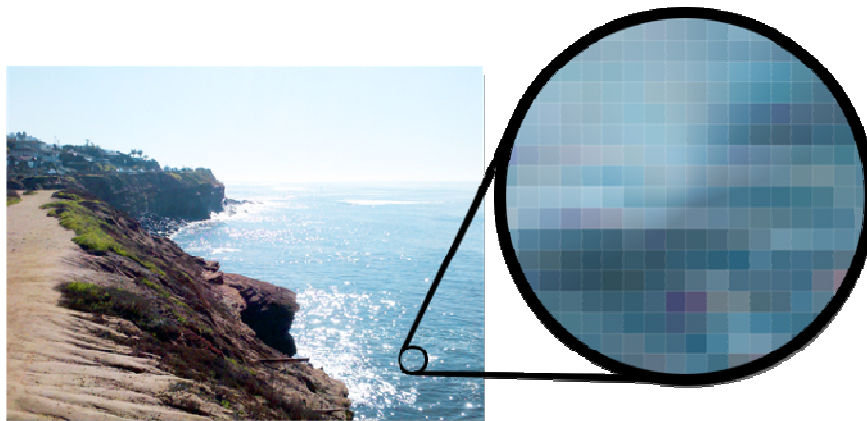


Figure II - 12Ensemble de pixel d'une image

II.4.4.1.2 Les filtres

Outre la manipulation de couleurs, on a très souvent recourt à des techniques de traitement d'images où l'on s'intéresse aux voisins des pixels. Cela est très utile

pour plein de situations : détecter les contours, appliquer un flou, etc. La notion de filtre est très utilisée dans les réseaux de neurones convolutifs.

Un filtre est une matrice de taille $w \times h$ qui est appliquée pas-à-pas sur l'image d'origine. À chaque pas, on multiplie chaque valeur de la portion de l'image d'origine recouverte par les valeurs du filtre, dont on somme le tout. L'image obtenue est appelée la convoluée.

Maintenant que nous avons découvert quelques bases sur les images numériques, nous pouvons détailler les différentes couches qui composent un réseau de neurones convolutif.

II.4.4.2 Convolution

Le premier type des couches que nous avons dans un réseau de neurone est celui de la convolution. Cette dernière applique une opération de convolution sur l'image d'entrée pour passer les résultats à la prochaine couche. L'opération n'est pas appliquée sur l'intégralité de l'image, mais avec un filtre avec une taille fixée à l'avance.

Les étapes d'application d'une convolution sur une image sont :

- Choix des dimensions du filtre F avec ses valeurs, exemple : $F = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$
- Effectuer des calculs, si l'on recalcule spécifiquement la première valeur de la convoluée, nous avons :

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} = \underbrace{1+0+1}_{1\text{ere ligne}} + \underbrace{0+1+0}_{2\text{eme ligne}} + \underbrace{0+0+1}_{3\text{eme ligne}} = 4$$

Cela revient à faire la multiplication terme-à-terme et à sommer le résultat.

II.4.4.3 Pooling

Une fois la convolution appliquée, nous souhaitons réduire la résolution de l'image. Cela ne veut pas dire que l'on va réduire sa taille, car l'objectif de la réduction de la résolution est plutôt de conserver l'information lors d'une réduction de la taille : c'est l'objectif du pooling.

Le pooling va permettre d'appliquer un filtre tout comme la convolution, mais contrairement à cette dernière, le pooling va agréger l'information en une seule valeur. Cette valeur peut être différentes formes, comme la moyenne ou le maximum, bien qu'en pratique ce soit le maximum qui soit le plus utilisée.

Le pooling a les mêmes paramètres que la convolution, la seule différence étant l'agrégation à effectuer. Avec le max pooling, chaque portion de l'image sélectionnée par le filtre sera résumée en une seule valeur, celle la plus élevée.

En pratique, les CNN cherchent donc à faire des successions de quelques convolutions (1 à 5) puis à appliquer un pooling. Ce schéma est ensuite répété plusieurs fois. Les architectures classiques de CNN vont donc être des paquets de plusieurs de couches de convolution / couches de pooling empilées successivement.

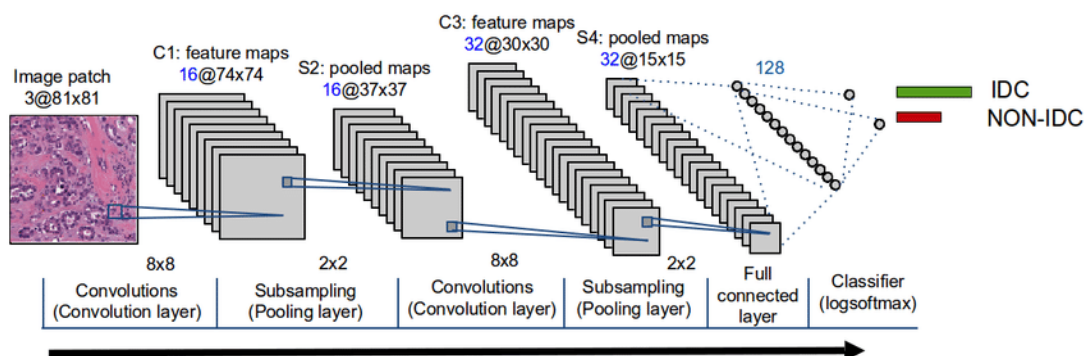


Figure II - 13 Architecture d'un réseau de neurone convolutif

II.5 Reconnaissance optique de caractères (OCR)

L'OCR (Optical Character Recognition) est un domaine de l'intelligence artificielle dont le rôle est de convertir une image bidimensionnelle contenant du texte (texte imprimé ou manuscrit) de sa représentation d'image en texte lisible et utilisable par un ordinateur.

L'OCR est particulièrement utile pour dématérialiser des documents papiers imprimés et les remplacer par des bases de données exploitables par ordinateur. De

nombreuses entreprises utilisent l'OCR pour lire les informations qui se trouvent sur une pièce d'identité ou bien les images de facture.

II.5.1 Fonctionnement de l'OCR

L'OCR fonctionne en effectuant les étapes suivantes :

Acquisition d'image

L'OCR analyse l'image numérisée et classe les zones claires comme fond et les zones sombres comme texte.

Dans la figure ci-dessous, on représente l'acquisition d'image, mais le texte n'était pas acquis correctement :

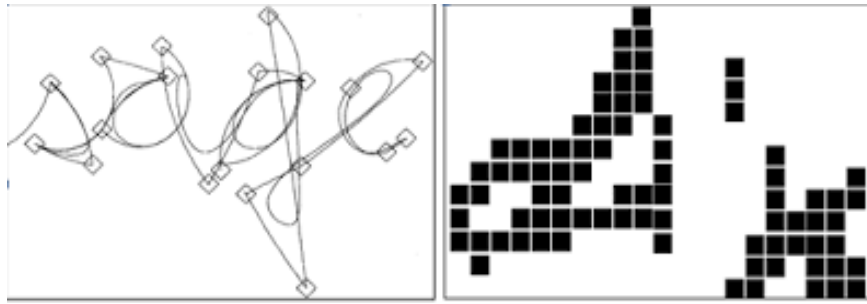


Figure II - 14 Classement des zones par OCR

Prétraitement :

L'OCR nettoie d'abord l'image et prélève les erreurs pour la préparer à la lecture. Voici quelques une des étapes de nettoyage :

- Désalignement ou inclinaison légère du document numérisé pour résoudre les problèmes d'alignement.
- Lissage des bords des images de texte ou prélèvement des tâches.
- Nettoyage des cases et des lignes dans l'image.
- Reconnaissance d'écriture pour la technologie OCR multilingue.

Reconnaissance de texte

L'OCR utilise deux principaux algorithmes pour la reconnaissance du texte, ils sont appelées la correspondance de motifs et l'extraction de caractéristiques.

Correspondance de motif

La correspondance de motif fonctionne en isolant l'image du caractère, appelée glyphe, et en la comparant à un glyphe stocké de manière similaire. La reconnaissance des formes ne fonctionne que si le glyphe stocké à une police et une échelle similaire au glyphe d'entrée.

Extraction de caractéristiques

L'extraction de caractéristique décompose les glyphes en caractéristiques tels que les lignes, les boucles fermées, la direction des lignes et les intersections de lignes. Il utilise ensuite ces caractéristiques pour trouver la meilleure correspondance ou le plus proche voisin parmi ses différents glyphes stockés.

Dans la figure ci-dessous représente les différents glyphes pour la lettre a :



Figure II - 15 Le glyphe de la lettre 'a'

Post traitement :

Après l'analyse, le système convertit les données textuelles extraites en un fichier informatisé comme le représente la figure ci-dessous :



Figure II - 16 Le fonctionnement de l'OCR

II.5.2 Types de l'OCR

L'OCR dispose de quatre types, à savoir :

Reconnaissance optique simple de caractères :

Un système OCR simple fonctionne en stockant de nombreux modèles différents de polices et d'images de texte comme modèles. L'OCR utilise des algorithmes de reconnaissance de motifs pour comparer les images de texte, caractère par caractère, à sa base de données interne. Voici un exemple ci-dessous :



Figure II - 17 La reconnaissance caractère par caractère

Reconnaissance intelligente de caractères :

Les systèmes OCR modernes utilisent la technologie de reconnaissance intelligente des caractères (ICR) pour lire le texte de la même manière que les

humains. Ils utilisent des méthodes avancées qui entraînent les machines à se comporter comme des humains en utilisant le Machine Learning. Un système de Machine Learning, appelé réseau neuronal, analyse le texte sur plusieurs niveaux en traitant l'image de manière répétée.

Reconnaissance intelligente des mots :

Fonctionnent sur les mêmes principes que l'OCR, la différence est seulement dans le traitement. Ce dernier traite les images mot par mot, mais l'OCR prétraite l'image caractère par caractère.

Dans la figure ci-dessous on représente la reconnaissance mot par mot :



Figure II - 18La reconnaissance mot par mot

Reconnaissance optique des marques :

La reconnaissance optique des marques identifie les logos dans un document par exemple.

II.6 Conclusion

Dans ce chapitre nous avons parlé du computer vision qui est une discipline informatique qui fait appel à l'intelligence artificiel pour comprendre des données visuelles. Et nous avons parlé de ses algorithmes tels que la détection d'objet et la reconnaissance des caractères (l'OCR). Enfin, nous avons fini par une bonne explication pour les réseaux de neurones qui est à la base de toute cette discipline.

CHAPITRE III

TECHNIQUES ET MODELES
UTILISES

CHAPITRE III

TECHNIQUES ET MODELE UTILISES

III.1- Introduction

Il existe des différentes techniques pour moderniser et informatiser les systèmes anciens utilisés dans les entreprises, afin de rendre les tâches de plus en plus faciles. Dans ce chapitre nous allons décrire en détail notre solution proposée avec ses différentes parties et expliquer les techniques de la vision par ordinateur choisies qui sont la détection d'objet et le système OCR.

III.2 Description détaillée de la solution proposée

On peut appliquer plusieurs solutions pour résoudre la problématique de la gestion manuelle des entrées/sorties au niveau de l'entreprise BMT, l'idéale est l'utilisation de Machine learning et ses outils : les réseaux de neurones convolutifs et la reconnaissance optique des caractères. Notre solution est divisée en deux étapes à savoir :

- La détection des plaques d'immatriculation.
- La reconnaissance du texte sur les plaques d'immatriculation.

III.2.1 Détection de la zone d'intérêt (Plaque d'immatriculation)

La zone d'intérêt ou l'objet sur lequel on s'intéresse dans notre cas est la plaque d'immatriculation du camion et l'immatriculation de contenir. La figure suivante illustre le lieu des entrées sorties du port de Béjaia montrant l'emplacement des caméras et leurs positionnement.

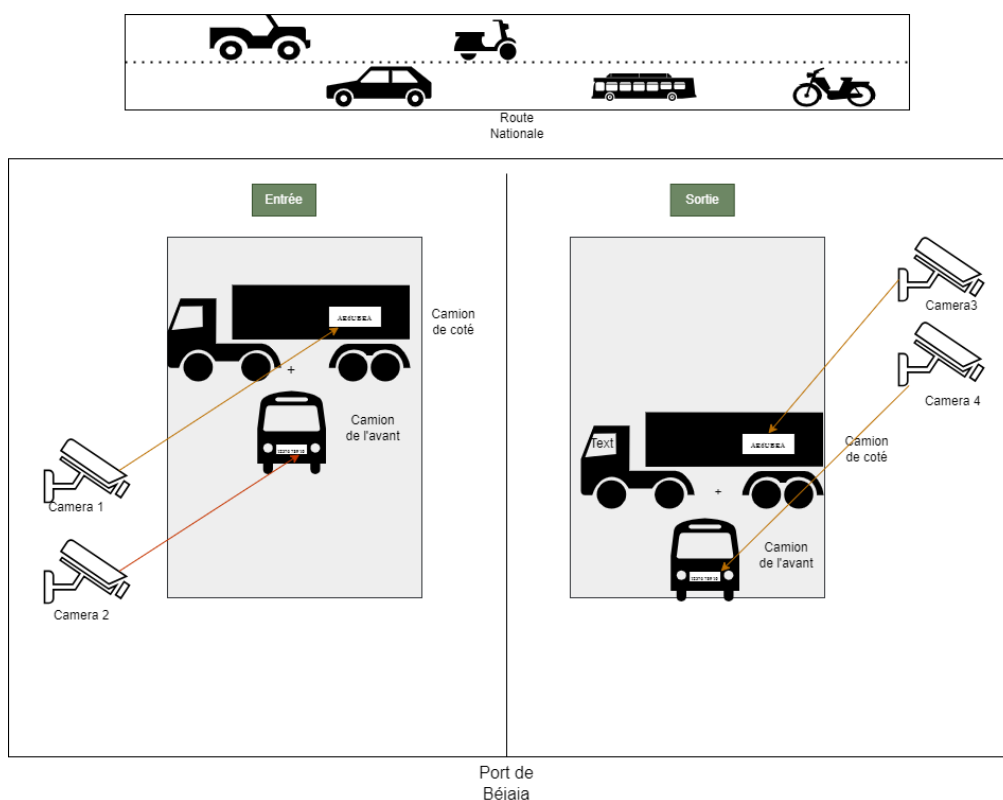


Figure III - 1 Lieu des entrées sorties du port de Béjaia

Nous avons 4 caméras placées, deux au niveau de l'entrée et deux autres à la sortie. La caméra 1 est fixée en haut pour détecter la matricule des camions qui vont entrer, la Caméra 2 pour la détection de la matricule de contenir attaché avec le camion d'entrée. Pour le coté de la sortie, la même chose avec la caméra 3 qui a la tache de détecter la matricule de chaque camion qui sors avec la caméra4 qui est prête à détecter son contenir attaché.

Maintenant la question qui se pose, comment procéder à cette détection et la mettre en œuvre. Nous avons bien défini la détection dans le chapitre précédent en mentionnant les modèles qui sont très connus et efficaces. Nous avons choisi le modèle YOLO lequel nous allons détailler et expliquer prochainement et comment l'exploiter pour accomplir la première partie de notre solution.

III.2.1 .1 Le modèle YOLO (You Only Look Once)

YOLO (You only look once) est un algorithme de détection d'objet en temps réel proposé par Redmon et al. Il est nommée 'YOLO' due à sa méthode spéciale de détection dans laquelle il résume les étapes standards de détection en une seule, l'étape d'identification de la zone d'intérêt, la classification et la localisation. Grace à sa méthode de fonctionnement et son architecture que nous allons aborder prochainement, cet algorithme nous permet d'atteindre un très bon niveau de précision avec une vitesse de détection élevée, ci-dessous nous avons une comparaison de nombre d'images par seconde traitées en entrée par plusieurs algorithmes de détections.

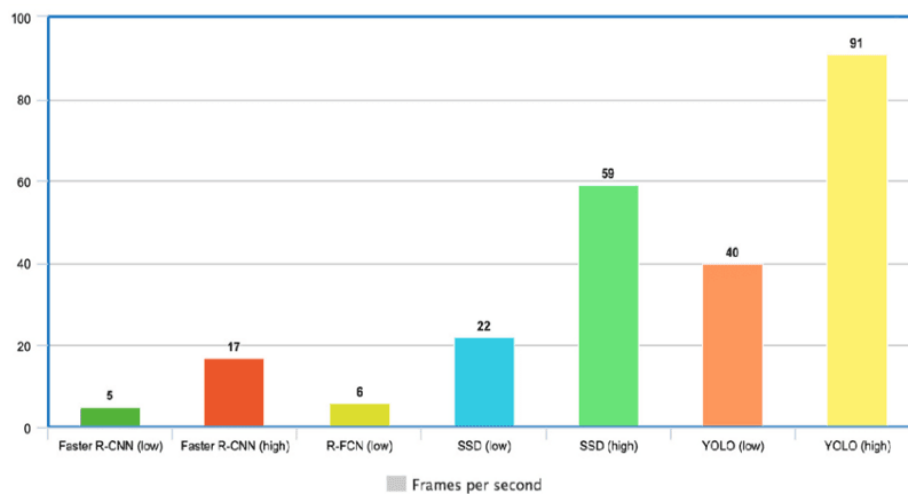


Figure III - 2 Histogramme qui représente le nombre d'images traitées en une seconde par des algorithmes de détections

En analysant la graphique à barre, nous remarquons que YOLO traite 91 images par seconde ce qui lui met en avance par rapport aux autres.

III.2.1 .2 Architecture de YOLO

YOLO utilise dans sa démarche les réseaux de neurones convolutifs. Il un réseaux spéciale avec ses différentes couches que nous voyons da la figure suivante.

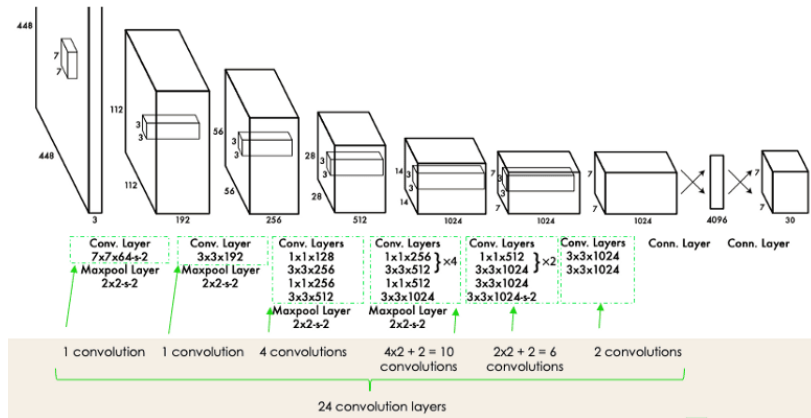


Figure III - 3 Architecture de YOLO

III.2.1 .3 Principe de fonctionnement de YOLO

La question qui se pose maintenant est comment arriver à exploiter le réseau de neurones convolutifs expliqué précédemment pour performer ses étapes de détection sachant que dans notre problématique l'objet voulu à être détecté est la plaque d'immatriculation.

Ci-dessous un exemple d'une image d'entrée d'un camion, de son côté l'image elle même après la détection de la plaque d'immatriculation effectuée par YOLO.

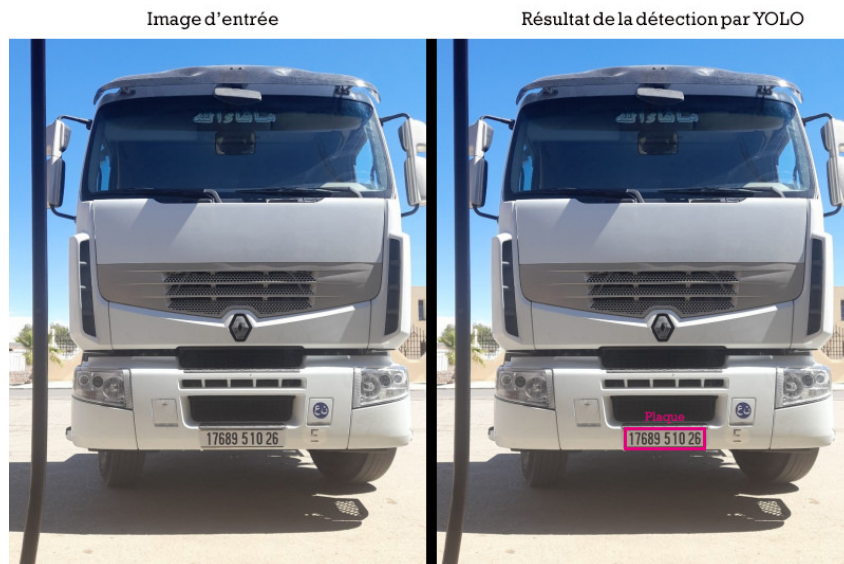


Figure III - 4 Résultat d'une détection avec YOLO

Pour arriver au résultat montré ci-dessus et plus exactement le rectangle qui encadre la matricule détecté, YOLO déclenches plusieurs opérations que nous ne voyons pas, mais elles se déroulent en arrière plan :

A- Division de l'entrée en grille :

La première étape débute avec la division de l'image d'entrée en grille $S \times S$, comme il est apparu dans l'image ci-dessus. Chaque Cellule de la grille est responsable sur la localisation et la prédiction de la classe de l'objet qui la couvre en définissant les valeurs de probabilités et la confiance, plusieurs objets dans une cellule peuvent être détectés et attribués à leurs classes appropriées.



Figure III - 5 Image divisée en grille

B- Définition des labels des objets détectés :

La prochaine étape est la détermination des labels ou des rectangles qui encadre tous les objets détectés sur l'image avec leurs informations nécessaires, ils sont aussi appelés 'bounding box'. Nous pouvons avoir plusieurs labels (boundig box) détectés par rapport aux nombre d'objets existants sur l'image. YOLO détermine des attributs pour ces labels dans un vecteur $Y=[pc,bx,by,bh,bw,c1]$.

Le vecteur Y contient les attributs suivant :

Pc : cette valeur correspond à la probabilité qu'une cellule de la grille qui divise l'image contient un objet. En utilisant l'image ci-dessus, nous remarquons les grilles en rouge ont un score de probabilité supérieur à zéro, donc il est possible qu'un objet

est détecté à l'intérieur de cette cellule, contrairement aux cellules jaunes qui ont probabilité qui est égale à zéro ce qui signifie l'absence de l'objet voulu qu'il soit détecté dedans.

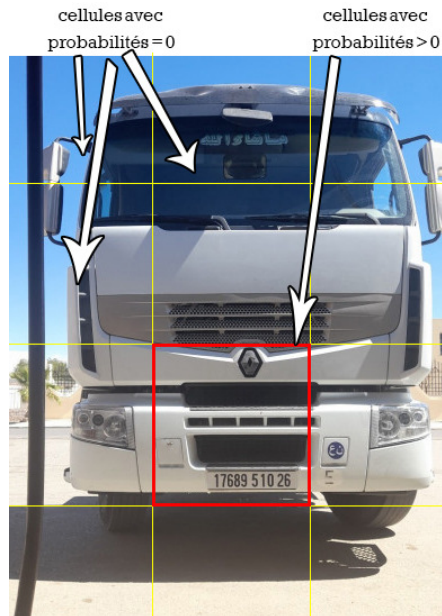


Figure III - 6 Les cellules avec leurs valeurs de probabilités

bx, by : ce sont les deux coordonnées de centre de label par rapport la cellule.

bh, bw : correspond à la hauteur et la largeur de rectangle de label.

c1 : correspond à la classe Plaque. Nous pouvons avoir plusieurs classes à savoir nos besoins.

Nous découpons la cellule en rouge qui contient l'objet détecté et nous essayons de représenter les informations précédentes extraites présentées dans le vecteur **Y** nous aurons :

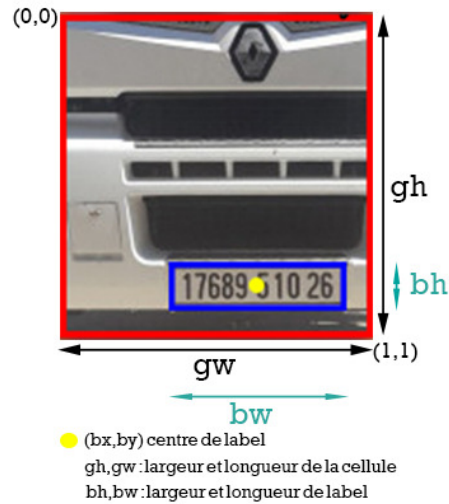


Figure III - 7Présentation des valeurs du vecteur Y

C- Application de l'IOU (intersection over union)

La plupart des temps, un seul objet dans une image peut avoir plusieurs cellules de détections qui concurrencent entre eux pour sa prédiction, malgré qu'elles ne sont pas toutes pertinentes. Le but de l'IOU est d'éliminer les cellules indésirables et de garder les plus pertinentes. Nous appliquons les étapes suivantes pour arriver au résultat final voulu qui est l'élimination des cellules non désirables :

- Définition de seuil de l'IOU par l'utilisateur, qui est en générale mis à 0.5.
- YOLO calcule la valeur de l'IOU de chaque cellule qui partage un même objet détecté.
- Finalement, il ignore les cellules qui ont une valeur de l'IOU < valeur de seuil définie et préserver celles qui ont une valeur d'IOU > valeur de seuil définie.

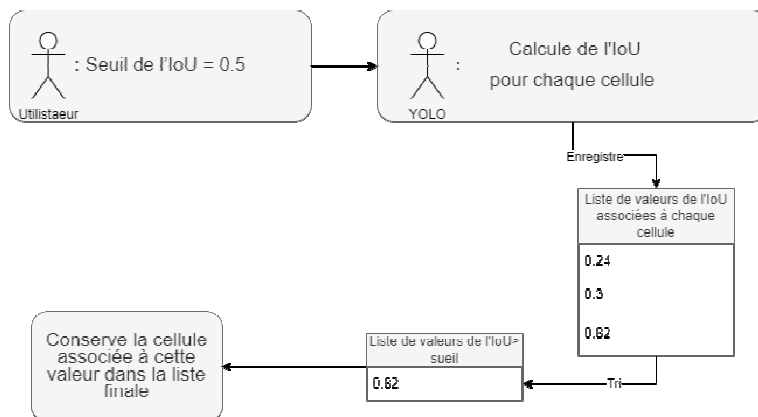


Figure III - 8 Etapes d'application de l'IOU

Maintenant la question qui se pose est comment calculer la valeur de l'IOU. Pour répondre à cette dernière, nous appuyons sur l'image suivante qui montre deux cellules qui partagent un même objet détecté, nous voulons choisir qu'une seule en appliquant l'IOU :

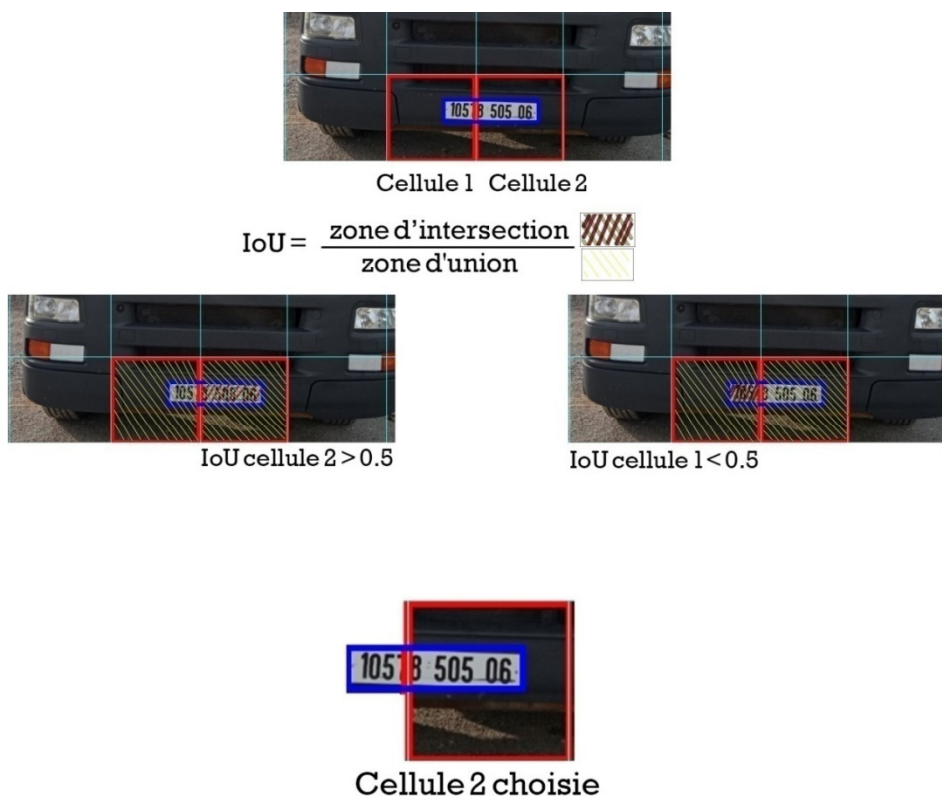


Figure III - 9 Application de l'IOU

Le calcul de l'IOU qui est mentionnée dans l'image précédente se fait par la division de la zone d'intersection sur la zone d'union. Au final des toutes les étapes

mentionnées, nous aurons la cellule avec la valeur supérieure à la valeur définie par l'utilisateur, cette cellule va être prise pour la suite de la détection.

D- Application de NMS (Non Max Suppression):

Une dernière méthode sera appliquée qui est la 'non max suppression'. Elle sert à éliminer les labels dupliqués détectés pour un seul objet en évitant le bruit. Au final de cette dernière étape, le label avec la plus grande valeur de probabilité sera maintenu.

III.3.2 Application de l'OCR sur la zone d'intérêt

Lorsqu'une entrée ou une sortie est détectée, notre système est prêt à appliquer la reconnaissance optique des caractères immédiatement sur les deux zones détectées (la matricule de camion et celle de contenir). On transformera les deux matricules en texte manipulable qui va être envoyé à notre base de donnée avec tous les détails nécessaires comme : le matricule en texte, l'heure et la date de la détection de la matricule, son type

Comme nous avons indiqué dans le chapitre précédent, il existe plusieurs types d'OCR qui diffèrent selon leurs fonctionnements. Dans notre cas, nous avons travaillé avec le deuxième type du système OCR qui est la reconnaissance optique simple de caractères.

- Système de reconnaissance optique simple de caractères

Nous avons choisi ce dernier car il est le plus simple pour extraire le numéro d'immatriculation, avec une rapidité dans l'opération grâce à ses algorithmes appliqués sur la plaque.

Dans la **figure III-10**, nous représentons la division des zones pour des caractères dans la plaque d'immatriculation :



Figure III - 10 Les différentes zones des caractères dans une plaque d'immatriculation

III.4. Conclusion

Dans ce chapitre nous avons parlé et bien détaillé les deux techniques de la vision par ordinateur choisies pour résoudre notre problématique. La première est la détection d'objet avec le modèle YOLO, qui sert à identifier la zone où se trouvent les plaques d'immatriculation sur les vidéos des caméras de surveillance. La deuxième est l'OCR qui est dans le but d'extraire les numéros des plaques sous forme d'un texte lisible.

CHAPITRE IV

ANALYSE & CONCEPTION DE L'APPLICATION WEB

CHAPITRE IV

ANALYSE & CONCEPTION DE L'APPLICATION WEB

IV.1- Introduction

Dans ce chapitre, nous allons nous concentrer sur l'analyse et la conception de notre application web. Nous allons tout d'abord présenter les acteurs de notre système. Ensuite, nous allons élaborer notre diagramme de cas d'utilisation après une extraction de tous les cas avec des descriptions textuelles pour chacun. Pour définir le comportement des acteurs avec le système, nous allons utiliser les diagrammes de séquence. Finalisant les diagrammes avec le diagramme de classe qui représente les différentes entités, leurs attributs et leurs relations. Ce diagramme nous permettra de comprendre la structure du système et d'identifier les différentes classes nécessaires pour l'implémentation. Enfin, nous aborderons la transformation du modèle de classes vers le modèle relationnel, qui permettra de définir la structure. Ceci est réalisé en utilisant le langage de modélisation UML (Unified Modeling Language), et en suivant une démarche, située à mi-chemin entre le processus UP (Unified Process) et XP (eXtrem Programming) et qui possède les caractéristiques suivantes :

- conduite par les cas d'utilisation, comme UP, mais bien plus simple.
- relativement légère et restreinte, comme XP, mais sans négliger les activités de modélisation en analyse et conception.
- fondée sur l'utilisation d'un sous-ensemble nécessaire et suffisant du langage UML (modéliser 80 % des problèmes en utilisant 20 % d'UML) [12].

IV.2 Présentation des acteurs du système

Un acteur d'un système est une entité de monde réel qui interagit avec lui. Les acteurs peuvent être des êtres humains ou autres systèmes externes. Avant qu'un système fonctionne très bien, il est précédé par une bonne identification des acteurs avec leurs rôles et c'est ce que nous allons aborder [6].

Acteur	Rôle
Utilisateur	Permet de consulter les entrées sorties, modifier les entrées sorties, effectuer des recherches.

Tableau IV - 1 Acteurs avec leurs rôles

IV.3 Cas d'utilisation par acteur

Plusieurs définitions des cas d'utilisation existent dans la littérature. De [13], les cas d'utilisation modélisent le comportement de système et aide à trouver ses exigences. Un acteur représente un rôle d'un utilisateur qui interagit avec le système. Un cas d'utilisation décrit une fonction que le système effectue pour atteindre le but de l'acteur. En résumant, nous aurons une description de ce que le système est censé de faire et comment l'acteur utilise ce système.

IV. 3.1 Identification des cas d'utilisation

Nous allons maintenant énumérer les différents cas d'utilisation pour chaque acteur du système. Afin de mieux présenter ces cas d'utilisation, le tableau ci-dessous présente les différents cas identifiés pour notre système :

Cas d'utilisation	Acteur	Messages Emis /Reçus par les acteurs
S'authentifier	Utilisateur	Emet : Informations d'email et mot de passe. Reçoit : L'interface appropriée à l'utilisateur.
Consulter les entrées /sorties	Utilisateur	Emet : Consultation des entrées/sorties. Reçoit : L'interface appropriée à l'utilisateur.
Modifier les entrées/sorties	Utilisateur	Emet : Modification, suppression d'une entrée ou sortie.

		Reçoit : Confirmation de modification et de suppression.
Effectuer des recherches	Utilisateur	Emet : Recherche par date, type et numéro d'immatriculation. Reçoit : Exécution de la recherche.

Tableau IV - 2Cas d'utilisation

IV. 3.2 Description textuelle des cas d'utilisation

La section suivante présente les différents cas d'utilisation pour chaque acteur du système. La description détaillée de chaque cas d'utilisation aide à mieux les mieux comprendre.

Cas d'utilisation « S'authentifier »

Le tableau ci-dessous permet de décrire le cas d'utilisation « S'authentifier » :

Sommaire d'identification	
Titre	S'authentifier
But	Cette fonctionnalité permet d'identifier l'utilisateur afin d'accéder aux fonctionnalités de l'application.
Résumé	Pour accéder à l'application, l'utilisateur doit renseigner son mail et son mot de passe puis valider le formulaire d'authentification. Le système vérifie ensuite la validité de ces informations .Si une erreur est détectée, un message d'erreur s'affiche et l'utilisateur est invité à recommencer l'authentification. Dans le cas contraire, l'utilisateur est redirigé vers l'interface principale de l'application.
Acteur	L'utilisateur
Description des enchainements	
Pré-conditions	/

Post-conditions	/
Scénario nominal	<p>Ce cas d'utilisation est déclenché lorsque l'utilisateur veut accéder au système.</p> <p>Enchainement (a) – S'identifier par une adresse email et mot de passe</p> <p>L'utilisateur fournit une adresse email et un mot de passe, et envoie les informations au système afin d'accéder son accueil. Le système effectue des vérifications.</p> <p>Si l'utilisateur a omis l'identifiant ou mot de passe alors il faut exécuter [Exception 01 : champ(s) vide(s)].</p> <p>Si l'identifiant et/ou mot de passe ne sont pas correcte alors il faut exécuter [Exception 02 : champ(s) incorrect(s)].</p>
Exceptions	
<p>[Exception 01 : Champ(s) vide(s)] : Le système notifie une erreur à l'utilisateur lui indiquant qu'il a oublié un ou plusieurs champs à saisir (email ou mot de passe), et l'invite à compléter les champs manquants.</p>	
<p>[Exception 02 : Champ(s) incorrect(s)]: Le système indique à l'utilisateur qu'une erreur est détectée liée à son email et/ou à son mot de passe, il l'invite à re-saisir son mail et/ou son mot de passe.</p>	

Tableau IV - 3 Cas d'utilisation « S'authentifier »

Cas d'utilisation « Consulter les entrées /sorties»

Le tableau ci-dessous permet de décrire le cas d'utilisation « Consulter les entrées/sorties » :

Sommaire d'identification	
Titre	Consulter les entrées/sorties
But	Cette fonctionnalité permet consulter toutes les entrées et sorties.

Résumé	Pour accéder à toutes les entrées et sorties qui entrent au fur et à mesure et les consulter, l'utilisateur doit accéder à la section de consultation des entrées sorties.
Acteur	L'utilisateur
Description des enchainements	
Pré-conditions	L'utilisateur doit s'authentifier pour accéder à l'interface consulter les entrées sorties
Post-conditions	/
Scénario nominal	Ce cas d'utilisation est déclenché lorsque l'administrateur veut consulter les entrées sorties

Tableau IV - 4 Cas d'utilisation « Consulter les entrées /sorties»

Cas d'utilisation « Modifier les entrées /sorties»

Le tableau ci-dessous permet de décrire le cas d'utilisation « Modifier les entrées/sorties » :

Sommaire d'identification	
Titre	Modifier les entrées sorties
But	Cette fonctionnalité permet faire une modification sur une entrée ou une sortie
Résumé	Pour modifier une entrée ou sortie, l'utilisateur accède à la page de modification ou il peut supprimer, modifier une valeur d'une entrée ou sortie et sauvegarder les changements au final.
Acteur	L'utilisateur
Description des enchainements	

Pré-conditions	L'utilisateur doit s'authentifier pour accéder à l'interface consulter les entrées sorties
Post-conditions	L'utilisateur peut consulter les entrées sorties après les changements effectués
Scénario nominal	/

Tableau IV - 5 Cas d'utilisation « Modifier les entrées /sorties»

Cas d'utilisation « Effectuer des recherches»

Le tableau ci-dessous permet de décrire le cas d'utilisation « Effectuer des recherches » :

Sommaire d'identification	
Titre	Effectuer des recherches.
But	Cette fonctionnalité permet de faire une recherche sur les entrées ou sorties.
Résumé	Pour faire une recherche sur une entrée ou une sortie, l'utilisateur doit la faire par rapport à la date, le type ou le numéro d'immatriculation.
Acteur	L'utilisateur
Description des enchainements	
Pré-conditions	/
Post-conditions	/
Scénario nominal	<p>Ce cas d'utilisation est déclenché lorsque L'utilisateur veut chercher sur quelque chose.</p> <p>Enchainement (a) – Ecrire les informations (le type, numéro d'immatriculation, ou la date) pour exécuter la recherche</p> <p>L'utilisateur fournit une des informations précédentes, et</p>

	<p>envoie les informations au système afin d'accéder aux résultats. Le système effectue des vérifications :</p> <p>Si l'utilisateur a mis des informations incorrectes alors il faut exécuter [Exception 01 : les informations n'existent pas].</p>
--	---

Tableau IV - 6 Cas d'utilisation « Effectuer des recherches »

IV. 3.3 Diagramme de cas d'utilisation global

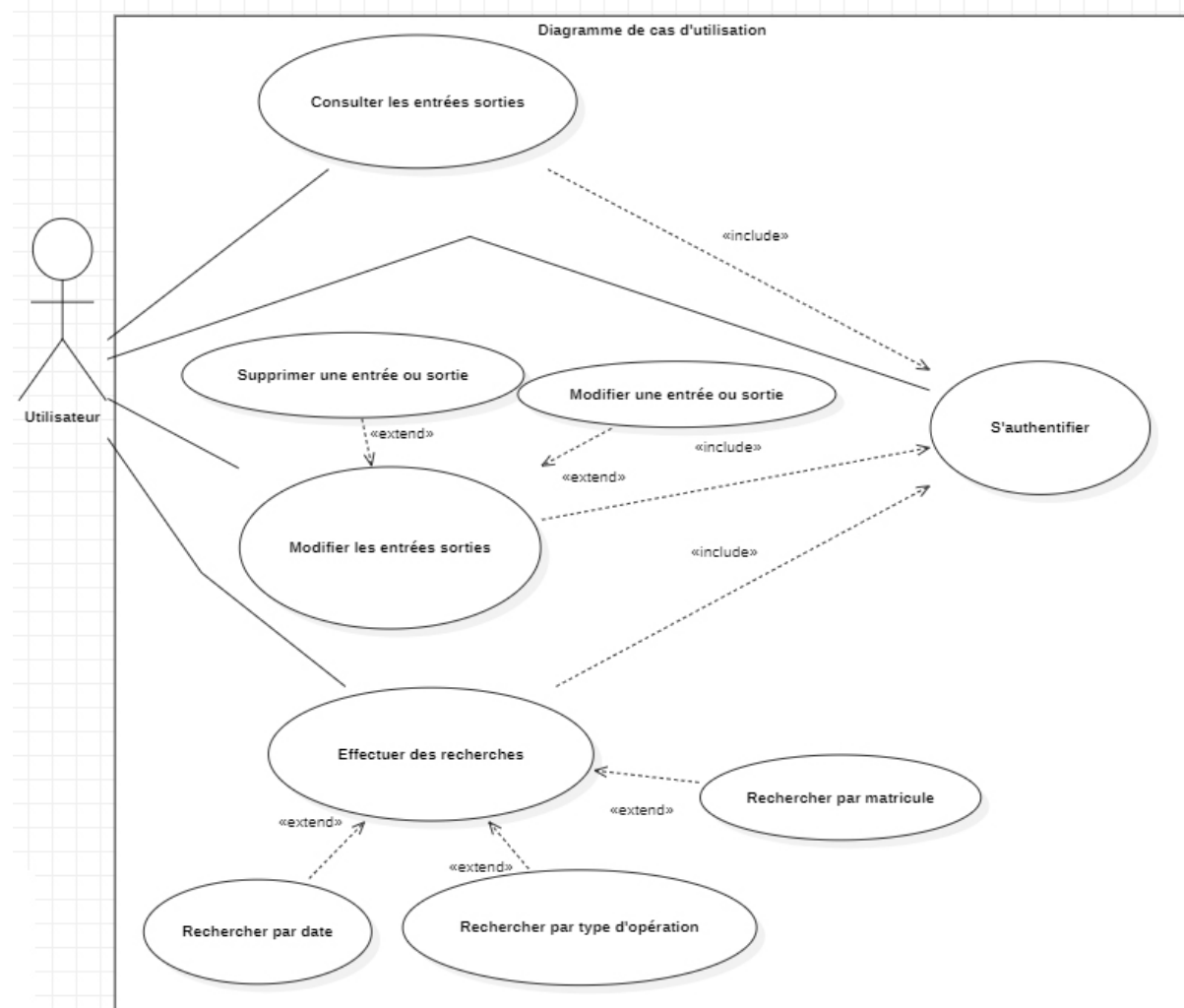


Figure IV - 1 Diagramme de cas d'utilisation

IV.4 Diagrammes de séquences

De nous pouvons dire que lors d'une analyse des besoins, un diagramme de séquence système est un outil important pour fournir des informations supplémentaires sur le cas d'utilisation en découvrant les interactions entre les acteurs et le système. Ce diagramme permet de représenter chaque acteur et le système à l'aide d'une ligne de vie, et les échanges de messages entre les acteurs et le système sont représentés par des flèches de communication. Pour chaque cas d'utilisation, un diagramme de séquence système est établi[14].

IV.4-1 Diagramme de séquence pour le cas d'utilisation « S'authentifier »

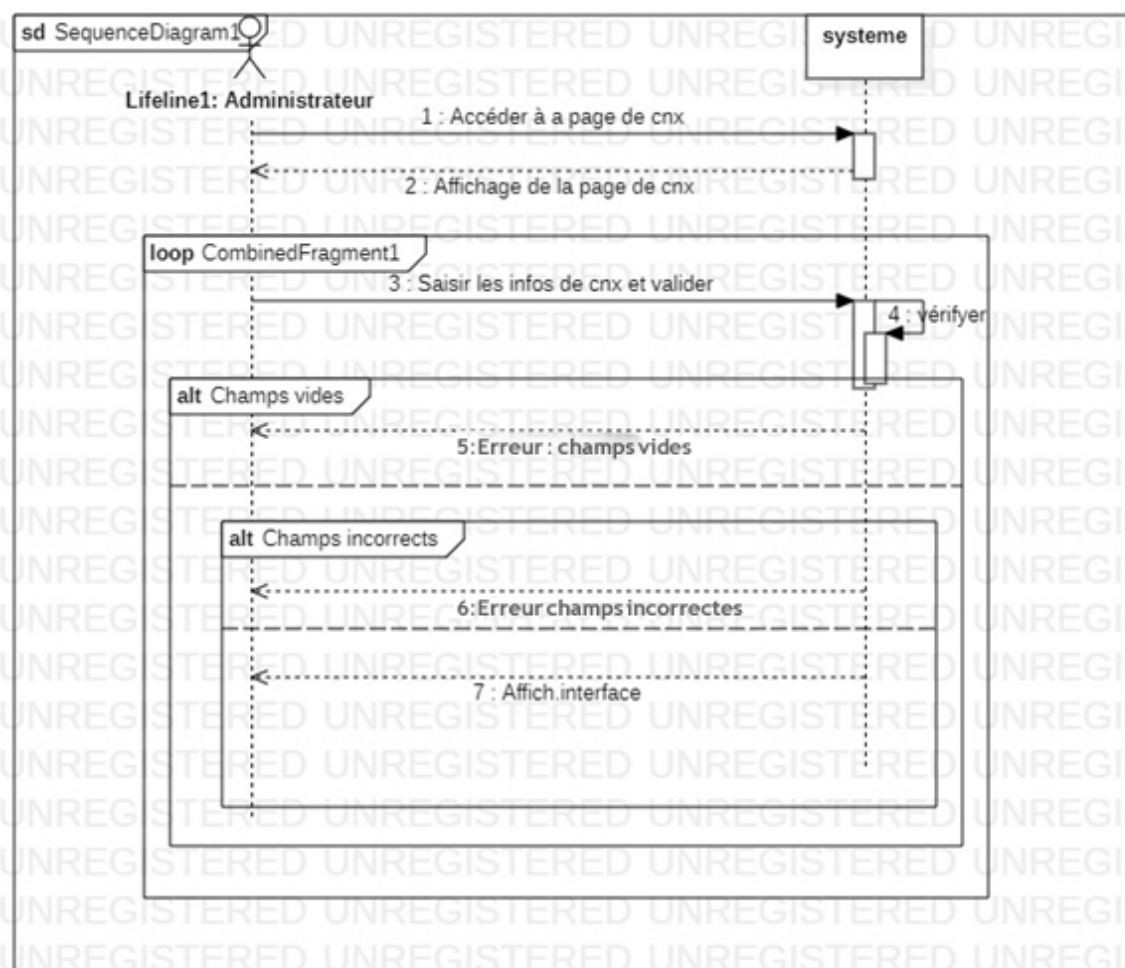


Figure IV - 2 Diagramme de séquence système pour le cas d'utilisation <s'authentifier>

IV.4-2 Diagramme de séquence pour le cas d'utilisation « Consulter les entrées/Sorties »

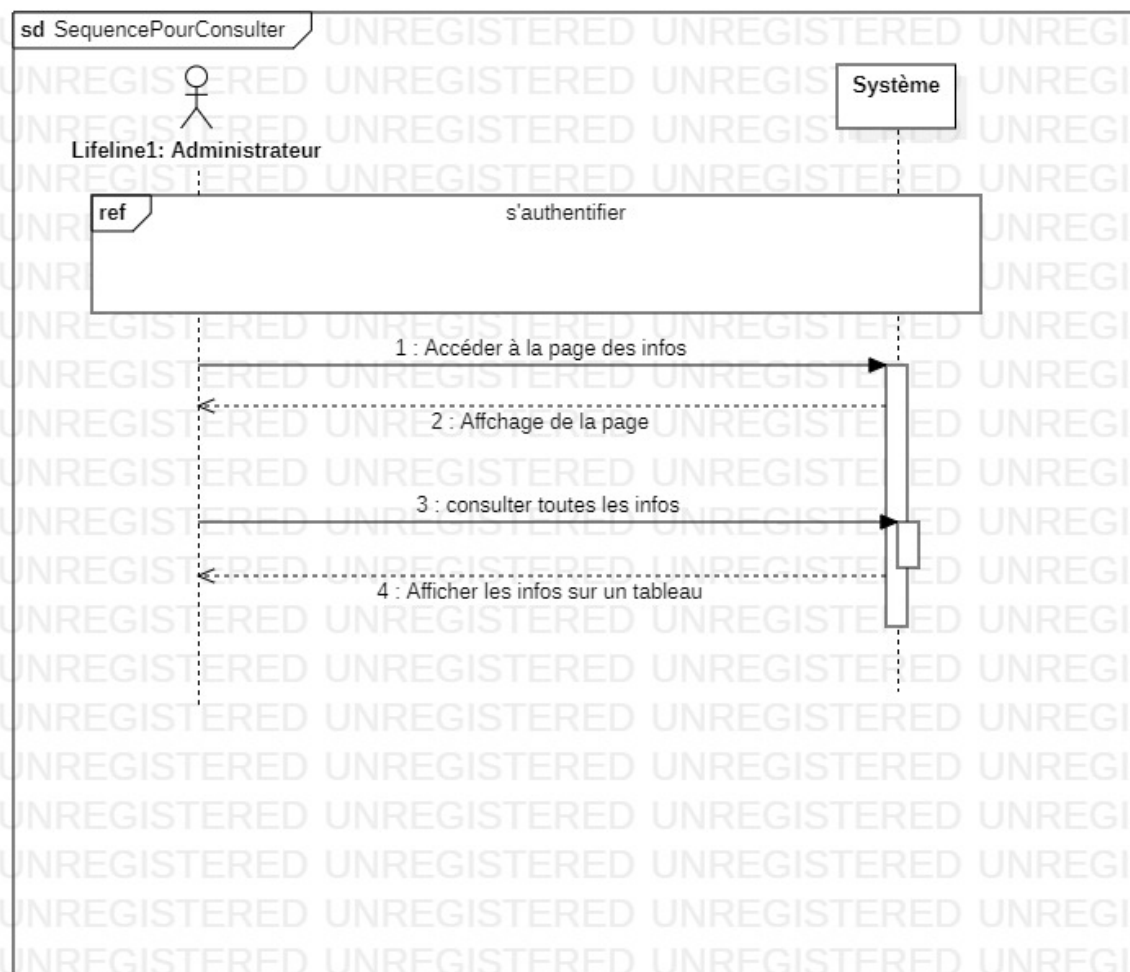


Figure IV - 3 Diagramme de séquence système pour le cas d'utilisation <Consulter les entrées/sorties>

IV.4-3 Diagramme de séquence pour le cas d'utilisation « Effectuer une recherche »

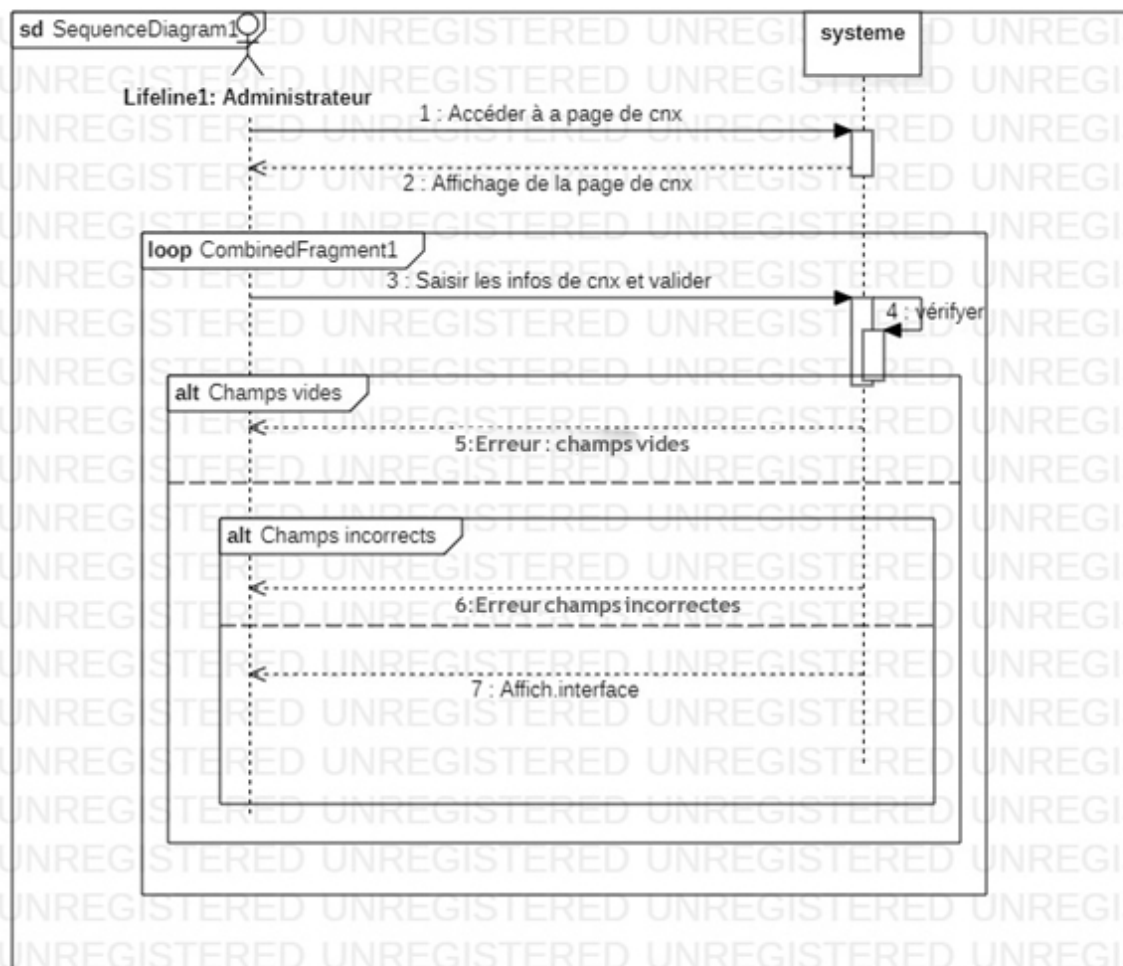


Figure IV - 4 Diagramme de séquence système pour le cas d'utilisation <Effectuer une recherche>

IV.5- Diagramme de classe du domaine

IV. 5.1 Diagramme de classe

Le diagramme de Classe est un diagramme statique qui représente la vue de notre application. Il est utilisé pour la visualisation, description de tous les aspects de notre système. Le diagramme de classe regroupe une collection de classes, interfaces, associations, collaborations et contraintes. Il est connu aussi comme un diagramme de structure. Une classe est une description d'un objet avec un rôle similaire dans notre

système qui consiste des attributs et des opérations. Une classe peut être impliquée dans une relation ou plusieurs avec d'autres classes[13] .

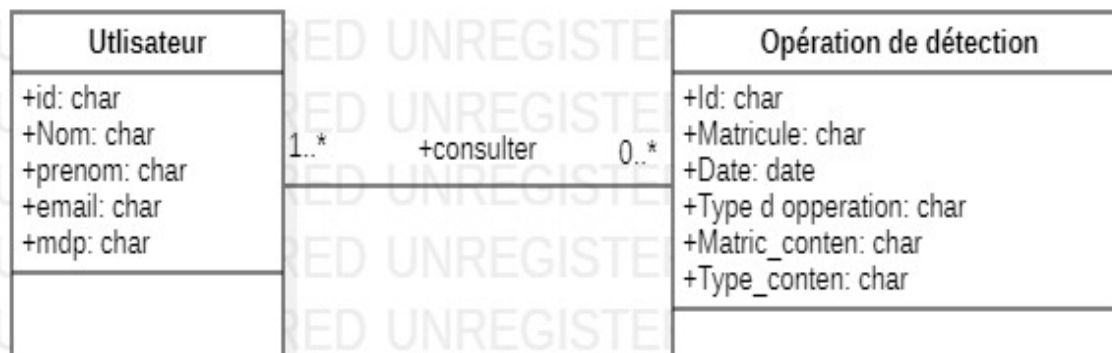


Figure IV - 5Diagramme de classe

IV. 5.3 Dictionnaire de données

Le dictionnaire de données présente la définition, le type (texte, numérique...) et le format (nombre de caractères, de décimales...) de l'ensemble des données gérées dans la base de données [15].

Pour faciliter la conception ultérieure des bases de données, il est recommandé de définir un dictionnaire de données qui regroupe les propriétés atomiques et les informations élémentaires qui ne peuvent pas être décomposable. Le dictionnaire des données du notre cas est représenté dans le tableau suivant :

Code	Désignation	Format	Table
id	identifiant	Alphanumérique	Utilisateur
Nom	Nom	Alphabétique	
prenom	Prénom	Alphabétique	
email	E-mail	Alphanumérique	
mdp	Mot de passe	Alphanumérique	
id	Identifiant	Alphanumérique	Opération_camion

Matricule	Matricule	Numérique	
Date	Date	Alphanumérique	
Type_d_operation	Type d'opération	Alphabétique	
Matric_conten	Matricule du conteneur	Alphabétique	
Type_conten	Type du conteneur	Alphabétique	

Tableau IV - 7 Dictionnaire de donnée

IV. 5.4 Schéma Relationnel

En utilisant notre diagramme de classe comme point de départ, nous pouvons dériver le schéma relationnel, aussi appelé le modèle logique de données.

-Utilisateur (id utilisateur, nom, prénom, email, mot_de_passe)

-Operations_camion (id operation, matricule, Date, type_operat, matric_conten, type_conteneur)

-Consulter (#id utilisateur,#id op camion)

IV. 6- Conclusion

Dans ce chapitre nous avons répondu à la question suivante : comment le système va implémenter les différents cas d'utilisation ? En utilisant des diagrammes de séquences afin d'expliquer les différents mécanismes pour chaque cas d'utilisation. Nous avons aussi utilisé un diagramme de classe pour visualiser les relations entre ces deux classes. En fin nous avons créé un dictionnaire de donnée et schéma relationnel pour notre diagramme de classe.

CHAPITRE V

PHASES DE REALISATION ET TESTS

CHAPITRE V

PHASE DE REALISATION ET TESTS

V.1- Introduction

Ce chapitre sera destiné à la mise en place de notre solution avec ses deux parties, en mettant tous les outils et techniques nécessaires pour atteindre l'objectif. Ensuite, nous allons présenter quelques captures de système lors de son fonctionnement pour visualiser sa performance et ses résultats.

V.2-Système de détection d'objet

V.2.1 Environnement de Programmation, Bibliothèques & Outils

-Python : est un langage de programmation de haut niveau interprété, orienté objet et doté d'une sémantique dynamique. Il est très utilisé dans l'intelligence artificielle grâce à ses bibliothèques qui facilite plusieurs taches dans ce domaine. La version que nous avons utilisé est 3.11. (Site officiel va être mentionnée si nous avons pris qu'une phrase[16] ?

-Jupyter Notebook : est une application Web open source qui permet de créer et de partager des documents contenant du code en direct, des équations, des visualisations et du texte narratif. Les utilisations incluent le nettoyage et la transformation des données, la simulation numérique, la modélisation statistique, la visualisation des données, l'apprentissage automatique et bien plus encore [17].

- OS : Le module os en Python est une bibliothèque standard qui permet de travailler avec le système d'exploitation. Il fournit un ensemble de fonctions qui peuvent être utilisées pour effectuer des opérations de bas niveau telles que la création de processus, l'allocation de mémoire et la gestion des signaux [18].

-LabelImg : est un outil d'annotation d'images graphiques. Il est écrit en Python et utilise Qt pour son interface graphique. Les annotations sont enregistrées sous forme de fichiers XML au format PASCAL VOC, le format utilisé par ImageNet. En outre, il prend également en charge les formats YOLO et CreateML [19].

-**YOLO** : (You only look Once)est un modèle de détection d'objet avec sa version V5 qui nous permet de détecter en temps réel des objets en utilisant son architecture.

-**Google Colab** : est un service cloud très puissant basé sur **Jupyter Notebook** , offert par GOOGLE et qui permet d'entraîner des modèles de Machine Learning et deep learning dans le cloud [20].

- **Google Drive** : Google Drive est un service de stockage et de partage de fichiers dans le cloud lancé par la société Google [21].

V.2.2 Etapes d'entraînement de YOLO

-**Préparation de dataset (jeux de donnée)**: Nous allons collecter un ensemble d'images contenant des camions avec leurs matricules, ce qui nous permet de personnaliser le modèle YOLO à détecter cet objet à travers les étapes suivantes.

Nous avons collecté 500 images qui représentent des camions avec une matricule bien claire.



Figure V- 1 Ensemble d'images du dataset

- **Annotation des images du dataset** : utilisant **LabelImg** pour labelliser chaque image en indiquant la zone de la matricule à fin de s'en sortir avec ses coordonnées dans un fichier texte. Chaque image a son fichier texte correspondant et les deux prennent le même nom.

Chaque ligne du fichier texte présente un objet présent dans l'image, et elle est sous forme de 5 variables (class x_center y_center width height) comme il est présenté dans la figure suivante :

0	0.573698	0.785648	0.061979	0.032407
class	x_center	y_center	width	height

Figure V- 2 Contenu d'un fichier d'annotation

- **Sélection des ensembles d'entraînement et validation** : mettre entre 70% des images de notre dataset comme ensemble d'entraînement, 30% pour la validation dans deux dossiers séparés puis nous les regroupons dans un seul dossier.

- **Création du fichier dataset.yaml** : il contient le chemin de nos deux ensembles, d'entraînement et de validation avec une spécification de classe d'objet qu'on veut détecter. Nous avons une seule classe que nous allons nommer plaque_immatriculation.

```
train: data_images/train
val: data_images/test
nc: 1
names: [
  'plaque_immatriculation'
]
```

Figure V- 3 Contenu du fichier dataset.yaml

- **Préparation des conditions d'entraînement** : Pour entraîner notre modèle YOLOv5 sur notre dataset préparé dans un temps pas si long, il nous faudra un GPU très rapide et puissant, malheureusement nous n'en acquérons pas et c'est pour cela que nous avons utilisé **Google Colab** qui offre.

Nous lançons les commandes présentées ci-dessous pour importer la bibliothèque **OS** qui nous permet de fixer notre chemin de travail qu'on nous allons mettre dans Google Drive.

```
import os
os.chdir('/content/drive/MyDrive/Licence_plate_yolo')
```

- **Importation du modèle YOLO sur Google Colab** : Nous exécutons la commande présente ci-dessous pour installer sur notre chemin défini précédemment tous les fichiers nécessaires pour le fonctionnement de YOLOv5 lors de l'entraînement.

```
!git clone https://github.com/ultralytics/yolov5
```

- **Importation du notre Dataset et fichier Yaml** : Nous allons importer le dossier de dataset et le fichier dataset.yaml sur le chemin crée déjà sur Google Drive .

- **Lancement d'entraînement** : sur Google Colab, nous exécutons la commandeci-dessous où nous trouvons notre fichier yaml, taille des images d'entrée, le batch-size qui définit le nombre d'images qui vont être propagés à travers le réseaudans une itération :

```
!python train.py --data data.yaml --weights yolov5s.pt --batch-size 8 --name Model --img 640 --epochs 150
```

- **Exportation du fichier des poids** : à la fin de l'opération d'entraînement, nous exportons le fichier des poids généré par YOLO en format .ONNX avec la commande suivante :

```
!python export.py --weights runs/train/Model/weights/best.pt --include onnx --simplify --opset 12
```

- **Exploitation du fichier .ONNX** : après l'exportation du fichier sous le nom best.ONNX, nous allons utiliser la bibliothèque OpenCVpour l'importation du fichier et l'inclure dans notre système de détection (figureV- 4)

```
#Loading ONNX File (wight file)  
net = cv2.dnn.readNetFromONNX('./Model/weights/best.onnx')  
#setting the backend  
net.setPreferableBackend(cv2.dnn.DNN_BACKEND_OPENCV)  
#setting the target  
net.setPreferableTarget(cv2.dnn.DNN_TARGET_CPU)
```

Figure V- 4Commande d'imporation du fichier best.onnx

Mais avant d'appliquer cette dernière étape, nous devons découvrir les étapes de préparation du système de reconnaissance de caractères pour fusionner les deux systèmes à fin de s'en sortir avec : une détection de plaque d'immatriculation + reconnaissance des chiffres de la plaque en même temps.

V. 3- Le système de reconnaissance de caractères

V.3.1 Environnement de Programmation, Bibliothèques & Outils

Cette partie est réservée à la liste de différentes technologies mises en œuvre dans l'implémentation de notre système de reconnaissance de caractères.

- **Pytesseract:**Pytesseract ou Python-tesseract est un outil de reconnaissance optique de caractères (OCR) pour python. Autrement dit, il reconnaîtra et lira le texte intégré dans les images.Python-tesseract est un wrapper pour le moteur Tesseract-OCR de Google. Il est également utile en tant que script d'invocation autonome pour tesseract, car il peut lire tous les types d'imagerie Pillow et Leptenica notamment jpeg, png, gif et autres. De plus s'il est comme script, il imprimera le texte reconnu au lieu de l'écrire dans un fichier [22].

V.3.1 Etapes de préparation

- En accédant au site officiel, nous allons télécharger et installer tesseract.
- Après, Nous allons installer pytesseract en écrivant la commande de l'installation

`pip install --upgrade pytesseract` sur l'invite de commande, comme le montre la figure ci-dessous :

```
(base) C:\WINDOWS\system32>pip install --upgrade pytesseract
Collecting pytesseract
  Downloading pytesseract-0.3.7.tar.gz (13 kB)
Requirement already satisfied, skipping upgrade: Pillow in c:\users\srikanth\anaconda3\lib\site-packages (from pytesseract) (8.0.1)
Building wheels for collected packages: pytesseract
  Building wheel for pytesseract (setup.py) ... done
  Created wheel for pytesseract: filename=pytesseract-0.3.7-py2.py3-none-any.whl size=13948 sha256=00c6ed5106a54b76a92951fbae0357707611314876e17edc08c0642903521fb4
  Stored in directory: c:\users\srikanth\appdata\local\pip\cache\wheels\ac\5b\xf4\d5bcc930771126a32285e058c576eda84e43691453a9f7ad71
Successfully built pytesseract
Installing collected packages: pytesseract
Successfully installed pytesseract-0.3.7

(base) C:\WINDOWS\system32>
```

Figure V- 5Commande d'installation de pytesseract

- **Importation de la bibliothèque :** Pour importer la bibliothèque `pytesseract` nous utilisons la commande suivante : `import pytesseract as pt`

- **Exécution du programme de détection et de reconnaissance**

Après l'importation des bibliothèques nécessaires, nous programmons les différentes fonctions qui nous permettent de manipuler les deux systèmes. Dans les figures qui suivent, nous montrons le résultat d'une détection d'une plaque

d'immatriculation d'une voiture avec sa reconnaissance de caractères que nous trouvons dans le fichier journal sous format texte:



Figure V- 6 Résultat d'une détection d'une plaque d'immatriculation pour une voiture

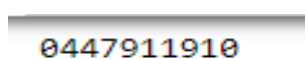


Figure V- 7 Résultat de la reconnaissance

V.4- Application web

Dans cette partie nous présentons l'application web qui permet de visualiser notre système de détection et de reconnaissance des plaques d'immatriculation.

V.4.1-Environnement de Programmation, Bibliothèques & Outils

- **Visual studio code** : Visual Studio Code est un environnement de développement intégré (IDE) gratuit et open-source créé par Microsoft. Il offre une gamme complète de fonctionnalités pour la programmation, y compris la coloration syntaxique, l'auto complétion, le débogage et la prise en charge de multiples langages de programmation. Disponible sur plusieurs plateformes, Visual Studio Code est devenu extrêmement populaire auprès des développeurs grâce à sa flexibilité, sa performance et son écosystème d'extensions riche[23].

- **Wampserver** :Wampserver est un environnement de développement web gratuit et open-source qui regroupeles logiciels Apache, MySQL et PHP. Il permet de

créer et de gérer des applications web sur un serveur local. Wampserver offre une interface conviviale pour configurer et administrer ces composants essentiels du développement web.[24]

-**HTML** : HTML est un langage de balisage utilisé pour organiser et présenter le contenu des pages web. Il se base sur une syntaxe composée de balises et d'éléments afin de définir la structure, le style et le contenu d'une page web[25].

-**CSS** : CSS est un langage de feuille de style qui permet de décrire la présentation et le style des documents HTML. Il offre la possibilité de contrôler l'apparence des éléments d'une page web, tels que la disposition, les couleurs, les polices, et bien d'autres aspects visuels[26].

- **NodeJs** : Node.js est une plateforme côté serveur basée sur le moteur JavaScript de Google Chrome. Il a été créé par Ryan Dahl en 2009 et la version la plus récente est la v0.10.36. Il s'agit d'un environnement d'exécution multiplateforme permettant de développer des applications côté serveur et réseau open source. Les programmes Node.js sont écrits en JavaScript et exécutés sur le runtime Node.js sous OS X, Microsoft Windows et Linux [27].

V.4.2 Interfaces d'application

Nous allons présenter ci-dessous quelques captures de notre application :



Figure V- 8 Page d'authentification

La *figure V-9* présente un exemple de reconnaissance d'une plaque d'immatriculation.

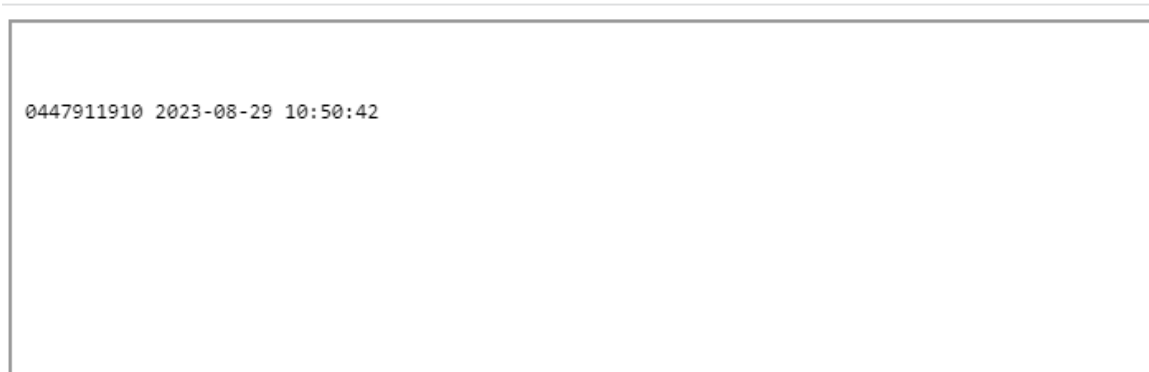


Figure V- 9 Page d'affichage de matricules

V.5 - Conclusion

Dans ce dernier chapitre, qui représente la phase finale du processus de développement de notre système et de l'application web, nous avons présenté différents outils, packages et langages informatiques utilisés pour développer notre système et notre application.

CONCLUSION

GENERALE

Conclusion Générale

Dans le contexte des entreprises, la sécurité et la traçabilité des opérations physiques est un projet d'une grande envergure. L'objectif majeur de notre projet était de répondre aux besoins actuels de notre organisme d'accueil en matière de mise en place d'un système de détection d'objet et de reconnaissance des caractères.

Ce travail a permis de concevoir une solution automatisée et conviviale, capable de détecter les plaques d'immatriculation et de reconnaître leurs contenues, garantissant ainsi une traçabilité précise et en temps réel. En développant cette solution, nous avons pu répondre aux défis rencontrés par l'entreprises en matière de la saisi manuel des plaques d'immatriculations.

Tout au long de ce rapport, nous avons présenté les différentes étapes de développement de ce projet. Notre choix, concernant la démarche à suivre, s'est porté sur une méthode simple et générique qui se situe à mi-chemin entre UP (Unified Process) et XP (eXtreme Programming).

Par ailleurs, ce projet nous a également permis d'approfondir nos connaissances et compétences dans les différents domaines de l'intelligence artificielle, tels que le computer vision et le Deep-Learning.

En fin, ce travail peut offrir une base solide pour l'amélioration continue de la gestion portuaire, avec un potentiel considérable pour élargir ses applications à d'autres domaines de la sécurité et de la traçabilité.

BIBLIOGRAPHIE

Bibliographie

[1] : BMT - Bejaia Mediterranean Terminal. 'Présentation du partenariat' . Site web, URL : <https://bejaiamed.com/presentation-du-partenariat/> Date de consultation : Mai 2023.

[2] : Maxime SERRA . 'Les valeurs au sein d'une entreprise, un levier de performances' . Site web, URL : <https://www.l-expert-comptable.com/a/532220-les-valeurs-au-sein-d-une-entreprise-un-levier-de-performances.html> . Mis à jour le 12 janvier 2009 17/11/2022. Date de consultation : Mai 2023.

[3] : BMT - Bejaia Mediterranean Terminal . 'Organisation' . Site web, URL : <https://bejaiamed.com/organisation/> . Date de consultation : Mai 2023.

[4] : IBM. 'Computer-vision'. Site web, URL : <https://www.ibm.com/fr-fr/topics/computer-vision#:~:text=La%20vision%20par%20ordinateur%20est,sur%20la%20base%20de%20ces>. Date de consultation : Juin 2023.

[5] : AI DATA. 'Computer vision through the ages'. Site web, URL : <https://www.telusinternational.com/insights/ai-data/article/computer-vision-through-the-ages>. Publie le 15 juillet 2021. Date de consultation Juin 2023.

[6] : Rensselaer Polytechnic Institute (RPI) . '3dvision_intro'. Site web, URL : https://sites.ecse.rpi.edu/~qji/CV/3dvision_intro.pdf. Date de consultation Juin 2023 .

[7] : Simplilearn. 'What Is Computer Vision: Applications, Benefits and How to Learn It'. <https://www.simplilearn.com/computer-vision-article#:~:text=Computer%20Vision%20Benefits&text=Faster%20and%20simpler%20process%20%2D%20Computer,well%20will%20commit%20zero%20mistakes>. Publié le 10 Aout 2023. Date de consultation 25 Aout 2023.

[8] : Fritz ai. 'Object Détection Guide'. Site web, URL : <https://fritz.ai/object-detection/>. Date de consultation : Aout 2023.

[9] : TURING. 'How Neural Network Models In Maching Learning'. Site web, URL : <https://aws.amazon.com/fr/what-is/neural-network/#:~:text=A%20neural%20network%20is%20a,that%20resembles%20the%20human%20brain>. Date de consultation Juin 2023 .

[10] : Amazon Web Services, Inc. ou ses sociétés apparentées. 'Qu' est ce que la reconnaissance optique de caractere '.Site web, URL : <https://aws.amazon.com/fr/what-is/ocr/>. Date de consultation Juin 2023 .

[11] : Zoumana Keita. 'YOLO Object Detection Explained'. Site web. URL : <https://www.datacamp.com/blog/yolo-object-detection-explained> . Publié en septembre 2022. Date de consultation : Juin 2023

[12] : Laurant AUDIBERT. 'UML 2 - De l'apprentissage à la pratique'. Site web, URL : <https://laurent-audibert.developpez.com/Cours-UML/?page=mise-en-oeuvre-uml>. Publié le 31 octobre 2006 - Mis à jour le 12 janvier 2009 . Date de consultation : Août 2023.

[13] : Fatima-Zohra KETTAF . 'module « UML »'. Site web, URL : <https://prive.iutenligne.net/iuRxM0CThIXDIjzG/informatique/langages/kettaf/UML/general/index.html>. Date de consultation : Août 2023.

[14] : Lucid Software Inc . 'diagramme-de-sequence-uml'. Site web, URL : <https://www.lucidchart.com/pages/fr/diagramme-de-sequence-uml> . Date de consultation : Août 2023.

[15] : Emilie Gauthier .dictionnaire de donnée. 2006. marben 2.1 – guide utilisateur. lemar (iuem/ubo), brest. p1 . Date de consultation : Août 2023.

[16] : Wikipedia . 'Python (langage)' . Site web, URL : [https://fr.wikipedia.org/wiki/Discussion:Python_\(langage\)](https://fr.wikipedia.org/wiki/Discussion:Python_(langage)) . Date de consultation : Août 2023 .

[17] : Dst Team. 'Jupyter Notebook : Un outil indispensable en partage de code'. Site web. URL : <https://datascientest.com/jupyter-notebook-tout-savoir>. Publié le 30 mars 2023 . Date de consultation : Août 2023.

[18] : PYTHONTECH . 'Le module OS Python'. Site web. URL : <https://www.tresfacile.net/le-module-os-python/#:~:text=Le%20module%20os%20en%20Python,et%20la%20gestion%20des%20signaux>. Publié le 30 avril 2023. Date de consultation : Août 2023.

[19] : OnWorks. 'Télécharger Labellmg pour Windows' . Site web. URL : <https://www.onworks.net/fr/software/windows/app-labelimg>. Date de consultation : Août 2023.

-
- [20] : Henri Michel. ' Google Colab : Le guide Ultime'. Site web. URL : <https://ledatascientist.com/google-colab-le-guide-ultime/#:~:text=Google%20Colab%20ou%20Colaboratory%20est,Learning%20directement%20dans%20le%20cloud>. Date de consultation : Août 2023.
- [21] : Wikipédia. 'Google Drive'. Site web . URL : https://fr.wikipedia.org/wiki/Google_Drive. Date de consultation : Août 2023.
- [22] : Python Software Foundation.' Python-tesseract is a python wrapper for Google's Tesseract-OCR'. Site web . URL : <https://pypi.org/project/pytesseract/> . Date de consultation : Août 2023.
- [23] : Microsoft Corporation,' Visual Studio Code',Site web. URL : <https://code.visualstudio.com/> . Date de consultation : Août 2023.
- [24] : Wampserver,Site web. URL : <https://www.wampserver.com/> . Date de consultation : Août 2023 .
- [25] : Mozilla Corporation's , ' Notions de base en HTML',Site web. URL : https://developer.mozilla.org/fr/docs/Learn/Getting_started_with_the_web/HTML_basics . Date de consultation : Août 2023 .
- [26] : World Wide Web Consortium (W3C),' CSS Current Work' . Site web. URL : <https://www.w3.org/>.Date de consultation : Août 2023 .
- [27] : Ravikiran A S . 'How to use Node Js for Backend Web Development in 2023'. Site web. URL : <https://www.simplilearn.com/tutorials/nodejs-tutorial/nodejs-backend> . Publié le 4 Mai 2023 .Date de consultation : Août 2023 .
- [28] : Ngene . 'Car Detection in Traffic with Deep Learning Toolkit for LabVIEW' . Site web. URL : <https://www.youtube.com/watch?app=desktop&v=QTaK4JPGDww> . Publié le 13 février 2017. Date de consultation : Août 2023 .
- [29] : Arne Wolfewicz, 'Deep Learning vs. Machine Learning – What's The Difference ?'. Site web . URL : <https://levity.ai/blog/difference-machine-learning-deep-learning> . Publié le 15 février 2023. Date de consultation : Août 2023 .
- [30] : Machine Learnia , ' FORMATION DEEP LEARNING COMPLETE (2021)'. Site web. URL : https://www.youtube.com/watch?v=XUFLg6dKQok&list=PLO_fdPEVlfKoanjvTJb1bd9V5d9Pzp8Rw . . Publié le 25 avril 2021. Date de consultation : Août 2023 .
-

Résumé

Ce mémoire a été rédigé en vue de l'obtention du diplôme de fin de cycle en Informatique parcours système d'information avancé. Il traite des problèmes posés par le processus de traçabilité des entrées sorties au sein de l'entreprise BMT. La solution proposée consiste à implémenter deux systèmes qui servent à détecter automatiquement les plaques des immatriculations des camions entrants ou sortants avec une reconnaissance des chiffres en temps réel. Une application web est réalisée pour compléter la solution avec laquelle les informations collectées par les deux systèmes précédents vont être visualisées et contrôlées. Pour mettre en œuvre notre solution, nous avons utilisé YOLO qui est un modèle de détection d'objet en temps réel basé sur le DEEP LEARNING, avec la bibliothèque PYTESSERACT pour reconnaissance de caractères. Nous avons utilisé une démarche à mi-chemin entre Up et Xp pour l'analyse et la conception de l'application web, et UML comme langage de modélisation. Les outils de réalisation sont : Visual studio code pour éditer notre code, HTML et CSS pour la structure des pages web, wampserver pour implémenter notre base de donnée et finalement NodeJS pour gérer le coté back end.

Mots clés : Application web, YOLO, DEEP LEARNING, PTESSERACT, UP, XP, UML, Visual studio code, HTML, CSS, wampserver, NodeJS.

Abstract

This dissertation was written with a view to obtaining the end-of-cycle diploma in Computer Science, information system course. It deals with the problems posed by the input-output traceability process within the BMT company. The proposed solution consists of implementing two systems which are used to automatically detect the registration plates of incoming or outgoing trucks with real-time digit recognition. A web application is created to complete the solution with which the information collected by the two previous systems will be visualized and controlled. To implement our solution, we used YOLO which is a real-time object detection model based on DEEP LEARNING, with the PYTESSERACT libraries for character recognition. We used an approach halfway between Up and XP for the analysis and design of the web application, and UML as a modeling language. The production tools are: visual studio code to edit our code, HTML and CSS for the structure of the web pages, wampserver to implement our database and finally NodeJS to manage the back end side.

Keywords: Webapplication, YOLO, DEEP LEARNING, PTESSERACT, UP, XP, UML, visual studio code, HTML, CSS, wampserver, NodeJS.