



وزارة البحث العلمي والتعليم العالي
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
جامعة عبد الحميد بن باديس مستغانم
Université Abdelhamid Ibn Badis Mostaganem
كلية العلوم والتكنولوجيا
Faculté des Sciences et de la Technologie
DEPARTEMENT DE GENIE ELECTRIQUE
N° d'ordre : M/GE/2020



MEMOIRE

Présenté pour obtenir le diplôme de

MASTER EN ELECTRONIQUE DES SYSTEME EMBARQUES

Par

Ould Mammar Mohammed Elamine

Midoun Mohamed Essedik

Réalisation d'un système embarqué à base d'un Raspberry
pour le contrôle d'accès à un parc automobile

Encadré par : **Mr Bentoumi Mohamed**

Soutenu le 02 juillet 2020 devant le jury composé de :

Président :	Mr Abed Mansour	Université de Mostaganem
Examineur :	Mr Benaouali Mohamed	Université de Mostaganem
Examineur :	Mr Rebhi Mustapha	Université de Mostaganem

Année Universitaire 2019/2020

Remerciements

Nous remercions tout d'abord, Allah de nous avoir donné le courage mais surtout le temps d'accomplir ce travail.

Nos remerciements vont directement dans un premier temps à notre encadreur Monsieur Bentoumi M. qui était à notre disposition et surtout de l'aide qu'il nous a apporté afin d'aboutir à ce travail et que dieu le bénisse davantage.

Ensuite nous tenons à remercier et à témoigner nos reconnaissances à tout le personnel de l'Université Abdelhamid Ibn Badis de Mostaganem en particulier celui du département de génie électrique de la faculté des sciences et technologies.

Sans oublier bien sûr nos camarades avec lesquels nous avons su mettre en valeur le travail d'équipe durant nos recherches et réalisation mais surtout de l'attention, l'hospitalité et aussi la considération qu'il ont démontré à notre égard durant ces cinq années.

Enfin, la liste ne pouvant pas être exhaustive, permettez-nous d'adresser nos sincères remerciements à nos parents, familles, amis, proches et toute personne qui nous ont encouragés et cru en nous dès le début.

Résumé :

Depuis l'avènement des systèmes de Reconnaissance Automatique des Plaques d'Immatriculations (RAPI) dès les années 80, ces derniers évoluent exponentiellement et deviennent de plus en plus nécessaires dans plusieurs domaines tels que le contrôle routier, le passage de frontières, l'accès aux parkings...

Les systèmes RAPI tournent autour de 2 grands axes : la détection de la plaque d'immatriculation et la reconnaissance de ses caractères alphanumériques. La détection de la plaque d'immatriculation et la reconnaissance des caractères s'effectuent par plusieurs méthodes. Chaque une a ses avantages et ses inconvénients.

L'objectif de ce projet de fin d'étude est de concevoir un système de reconnaissance automatique des plaques d'immatriculations algériennes pour contrôler l'accès aux parkings. Il est implémenté sur un Raspberry Pi sous Linux et programmé en langage python. Ce système est constitué de 2 grandes fonctions. La première s'occupe de la détection et la lecture de la plaque d'immatriculation et la deuxième sert à gérer les capteurs et les actionneurs pour assurer l'entrée et la sortie des véhicules du parking.

Abstract:

Since the advent of Automatic License Plate Recognition (ALPR) systems in the 1980s, they have grown exponentially and they becoming more and more necessary in several fields such as traffic control, border crossing and access to parking lots.

RAPI systems revolve around 2 main axes: License plate detection and recognition of its alpha-numeric characters. The detection of the license plate and the recognition of the characters are carried out by several methods. Each one has advantages and drawbacks.

The objective of this end-of-study project is to design an automatic recognition system for Algerian license plates to control access to car parking. It is implemented on a Raspberry Pi under Linux and programmed in python language. This system consists of 2 main functions. The first is responsible of detecting and reading the license plate and the second is used to manage the sensors and actuators to ensure the entry and exit of vehicles at the car parks.

ملخص:

منذ ظهور أنظمة التعرف التلقائي على لوحات ترقيم السيارات (ALPR) في بداية الثمانينيات، تطورت هاته الأخيرة وأصبحت ضرورية أكثر في العديد من المجالات مثل مراقبة حركة المرور، معابر الحدود والدخول إلى مواقف السيارات. تدور أنظمة (ALPR) حول محورين رئيسيين: الكشف عن لوحة الترخيم والتعرف على الرموز التي تحتويها. يتم الكشف عن لوحات الترخيم والتعرف على رموز التي تحتويها بعدة طرق، لكل إيجابياتها وسلبياتها. الهدف من مشروع نهاية الدراسة هذا هو تصميم نظام التعرف الآلي على لوحات الترخيم الجزائرية للتحكم في دخول إلى مواقف السيارات. هذا المشروع مطبق على لوحة Pi Raspberry التي تعمل على نظام التشغيل Linux و تم برمجته بلغة Python. هذا النظام مبني على وظيفتين رئيسيتين: الأولى مسؤولة عن كشف وقراءة لوحات الترخيم، والثانية مسؤولة عن إدارة أجهزة الاستشعار والمشغلات لضمان دخول وخروج المركبات في مواقف السيارات.

Sommaire

<u>Résumé</u>	<u>3</u>
<u>Introduction générale</u>	<u>8</u>
<u>I Chapitre 1 : Généralités sur l'image</u>	<u>11</u>
I.1 Introduction.....	12
I.2 L'image numérique.....	12
I.2.1 L'acquisition de l'image numérique.....	12
I.2.2 Caractéristiques de l'image numérique.....	13
I.3 Traitement d'image.....	13
I.3.1 Prétraitements.....	14
I.3.2 L'analyse (traitement).....	14
I.4 Conclusion.....	16
<u>II Chapitre 2 : Hardware / Software</u>	<u>17</u>
II.1 Introduction.....	18
II.2 Raspberry Pi.....	18
II.2.1 Historique.....	18
II.2.2 Evolution.....	19
II.2.3 Composants de base.....	19
II.3 Linux.....	20
II.3.1 Historique.....	20
II.4 Python.....	21
II.4.1 Historique.....	22
II.5 OpenCV.....	22
II.6 Conclusion.....	23
<u>III Chapitre 3 : Les Systèmes RAPI</u>	<u>24</u>
III.1 Introduction.....	25
III.2 Les systèmes RAPI.....	25
III.2.1 Historique.....	25
III.2.2 Composition des systèmes RAPI.....	26
III.2.3 Principe de fonctionnement.....	26
III.2.4 Méthodes de détection des plaques d'immatriculation.....	27
III.2.4.1. Technique basant sur les propriétés de la plaque.....	27

Table des matières

III.2.4.2. Technique basant sur les contours de la plaque.....	27
III.2.4.3. Technique basant sur l'intelligence artificiel.....	28
III.2.4.4. Technique basant sur la signature de la plaque	28
III.2.5 Caractéristiques des plaques d'immatriculation algérienne	28
III.2.6 Problèmes des systèmes RAPI.....	29
III.2.7 Domaines et exemples d'utilisation des systèmes RAPI	29
III.2.7.1 Le péage urbain	29
III.2.7.2 Société de parking	30
III.3 Conclusion	30
<u>IV Chapitre 4 : Implémentation du Système RAPI.....</u>	<u>31</u>
IV.1 Introduction.....	32
IV.2 L'emplacement physiques de notre système.....	32
IV.3 Le matériel : Raspberry et ses périphériques	33
IV.4 L'environnement logiciel (Software).....	34
IV.4.1 Présentation du fonctionnement du système.....	35
IV.4.2 Détection et lecture de la plaque d'immatriculation	41
IV.4.3 Base de données et historique	43
IV.5 Résultats obtenus du système RAPI.....	45
IV.5.1 Résultats corrects	45
IV.5.2 Résultats erronés	49
IV.5.3 Analyse des résultats.....	50
IV.5.3.1 La détection de la plaque	50
IV.5.3.2 La reconnaissance optique de caractère.....	50
IV.6 Les coûts	51
IV.7 Conclusion	51
<u>Conclusion et perspectives</u>	<u>52</u>
<u>Annexes.....</u>	<u>54</u>
<u>Bibliographie</u>	<u>59</u>

Liste des figures

Figure 0-1 : Les 4 fonctions d'un système RAPI	9
Figure I-1 : Exemple d'une image numérique.....	12
Figure I-2 : Schéma d'acquisition d'image	13
Figure I-3 : Matrice de pixel montrant les contours	15
Figure I-4 : Détection de contours par méthode de Canny.....	15
Figure II-1: Raspberry PI 4 model B.....	18
Figure II-2 : Evolution du Raspberry PI.....	19
Figure II-3 : Composants de base du Raspberry Pi 4	20
Figure II-4 : Interface système d'exploitation Linux-Debian.....	21
Figure II-5 : Logo Python.....	21
Figure II-6 : Programme factoriel avec Python	22
Figure II-7 : Localisation et suivi des véhicules avec OpenCV	22
Figure III-1 : Système RAPI.....	25
Figure III-2 : Composition des systèmes RAPI.....	26
Figure III-3 : Les principales fonctions d'un système RAPI	26
Figure III-4 : Détection de plaque d'immatriculation par couleur	27
Figure III-5 : Détection de plaque d'immatriculation par contour	27
Figure III-6 : Système de plaque d'immatriculation algérienne.....	28
Figure III-7 : Péage urbain	29
Figure III-8 : système d'accès automatique au parking.....	30
Figure IV-1 : Vue de dessus qui décrit l'emplacement du système RAPI	32
Figure IV-2 : Vue en plongée qui décrit l'emplacement du système RAPI	33
Figure IV-3 : Schéma électronique du système RAPI.....	34
Figure IV-4 : diagramme d'états de fonctionnement du système RAPI	37
Figure IV-5 : Organigramme de fonctionnement du système RAPI	38
Figure IV-6 : Suite de l'organigramme de fonctionnement du système RAPI	39
Figure IV-7 : Organigramme de la fonction RAPI	42
Figure IV-8 : Fichier <i>ma.csv</i> contenant les matricules des véhicules autorisés a entrées	43
Figure IV-9 : Fichier <i>mes.csv</i> l'historique de tous les véhicules.....	44
Figure IV-10 : Fichier <i>xx.csv</i> l'historique d'un des véhicules autorisés.....	44

Abréviation

RAPI : Reconnaissance Automatique des Plaques d'Immatriculations

OCR : Optical Character Recognition

ANPR : Automatic Number Plate Recognition

LAPI : Lecture Automatique des Plaques d'Immatriculations

AVI : Automatic Vehicule Identification

LPR : License Plate Recognition

CAN : Conversion Analogique Numérique

DPI (dpi): Dots Per Inch

PPM (ppm): Parts Per Million

FPGA: Field-Programmable Gate Array

RAM: Random-Access Memory

HDMI: High-Definition Multimedia Interface

ARM: Advanced RISC Machine

RISC: Reduced Instruction Set Computer

IHM: Interfaces Homme-Machine

CPU: Central Processing Unit

GPIO: General-Purpose Input/Output

4Kp60: Resolution (4000 Pixels 60 Hz)

LAN: Local Area Network

PoE HAT: Power Over Ethernet Hardware Attached on Top

Open CV: Open Source Computer Vision

PI : Plaque d'Immatriculation

B.e.m: Bouton entrée manuel

NI : Numéro d'Immatriculation

Introduction générale

Selon les statistiques, le nombre de voitures dans le monde avait dépassé le milliard de véhicules en 2010, ce chiffre s'approche à 2 milliards de véhicules en 2020. Les estimations disent que le nombre total de véhicules dans le monde pourrait doubler d'ici 2040.[1]

En Algérie, le parc automobile est apprécié à plus de 6.4 millions de véhicules en 2018.[2] En raison du grand nombre de voitures en circulation au niveau mondial, le numéro d'immatriculation est nécessaire. Il permet d'identifier tous les véhicules de manière unique.

Le numéro d'immatriculation est constitué d'une combinaison unique de caractères alphanumériques. Il est destiné à identifier facilement et de façon unique un véhicule.

Vu le grand flux de voitures en circulation, les systèmes de Reconnaissance Automatique de Plaques d'Immatriculation (RAPI) sont apparus. Les systèmes RAPI servent à automatiser la lecture et l'identification des plaques d'immatriculation.

Le système RAPI est constitué généralement d'une caméra, liée à un PC ou un serveur. La caméra envoie une image au PC pour la traiter et extraire le numéro d'immatriculation. Le numéro d'immatriculation permet d'avoir, à travers la base de données, des informations sur le véhicule et son propriétaire.

Les systèmes RAPI sont ainsi basés sur ces quatre fonctions :

- L'acquisition d'une image d'un véhicule.
- La détection de la plaque d'immatriculation du véhicule dans l'image (en utilisant les outils de traitement d'image).
- Segmentation des caractères de la plaque d'immatriculation.
- Reconnaissance optique de caractères (OCR).



Figure 0-1 : Les 4 fonctions d'un système RAPI

Les systèmes RAPI ont plusieurs applications telles que le contrôle, la surveillance, les passages de péages autoroutiers, les passages de frontières, le contrôle d'accès aux parkings...etc.

L'objectif de notre projet de fin d'étude est d'étudier et concevoir un système RAPI pour contrôler l'accès des véhicules au parking de notre université.

Après cette introduction, les problématiques suivantes peuvent se poser :

- Quels sont les systèmes RAPI existants, quels sont les avantages et les inconvénients de ces systèmes ? et quel est le système le plus approprié à notre cas d'étude ?
- Quelles sont les méthodes et les étapes à suivre pour qu'on puisse concevoir un système RAPI efficace et performant ?
- Quels sont les outils matériels et logiciels à utiliser pour réaliser notre système ?

Donc pour répondre à notre problématique nous avons organisé notre travail en quatre chapitres comme suit :

- Dans le premier chapitre, nous allons étudier quelques notions sur l'image et les outils de traitement d'image utilisés du fait notre système RAPI conçu repose sur la vision artificielle.
- Le deuxième chapitre contient des descriptions de Raspberry Pi et son environnement logiciel qui serviront à la réalisation de notre système.
- Le troisième chapitre présente les principes de fonctionnement d'un système RAPI, les méthodes de reconnaissances des plaques, ainsi que les caractéristiques des plaques algériennes.
- Dans le quatrième chapitre, on présente la mise en œuvre de notre système RAPI côté matériel et logiciel et enfin l'analyse des résultats obtenus.

Chapitre 1 : Généralités sur l'image

I.1 Introduction

Dans ce chapitre, nous allons présenter les notions nécessaires à la compréhension de l'image et de ses caractéristiques.

I.2 L'image numérique

L'image numérique est une représentation visuelle d'un ou plusieurs objets qui peuvent être naturels ou artificiels. Cette dernière est traitée et stockée en binaire. Une image numérique est une grille composée d'un ensemble fini d'éléments, appelés « Picture Element », ou pixels (Voxels en 3D). Elle est représentée mathématiquement sous forme d'une fonction à deux dimensions de l'intensité lumineuse, $I(x, y)$, où x et y sont des coordonnées spatiales, et la valeur de I en (x, y) est proportionnelle à la luminosité de l'image en ce point (pixel).[5][6][13]

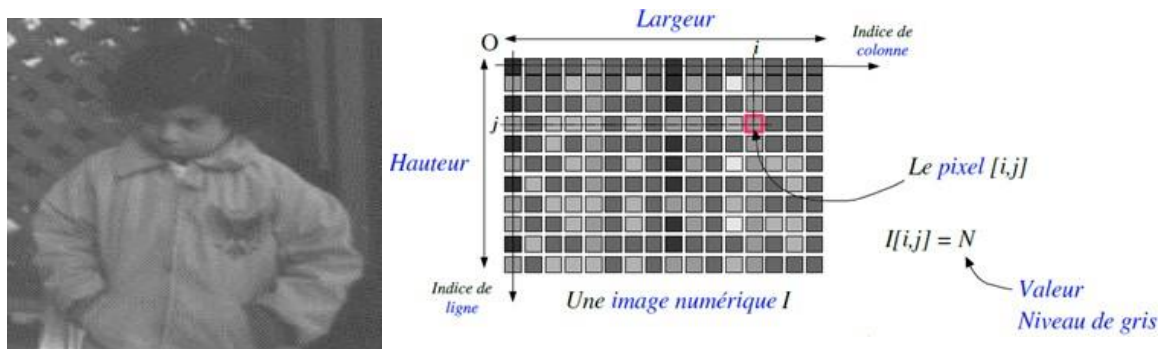


Figure I-1 : Exemple d'une image numérique

I.2.1 L'acquisition de l'image numérique :

L'acquisition d'image est un chaînon important dans la chaîne de production d'image numérique. L'acquisition consiste en un dispositif optique, généralement une caméra ou un scanner, qui capte une onde (lumière) d'une scène puis la numérise. L'objectif de la numérisation est de transformer le signal analogique qui contient une quantité infinie d'amplitudes en un signal numérique contenant lui une quantité finie de valeurs. Le passage de l'analogique au numérique consiste en 3 étapes successives : l'échantillonnage, la quantification et le codage. (conversion analogique-numérique (CAN)).[8]

- **L'échantillonnage** : L'échantillonnage consiste à prélever les valeurs d'un signal à intervalles réguliers définis par le théorème de Shannon. Il produit une suite de valeurs discrètes nommées échantillons.[10][25]
- **La quantification** : L'opération de quantification consiste à remplacer la valeur exacte analogique de l'échantillon par la plus proche valeur approximative extraite d'un ensemble fini. [11][25]
- **Le codage** : cette opération consiste à attribuer un nombre binaire à tout échantillon quantifié (niveaux de tensions issus de la phase de quantification).[25]

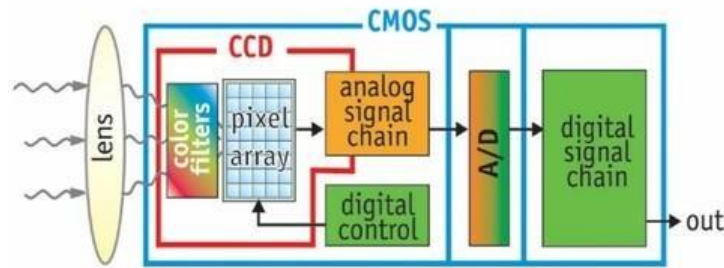


Figure I-2 : Schéma d'acquisition d'image

I.2.2 Caractéristiques de l'image numérique :

Une image numérique est caractérisée par :[9][12]

- **La Taille** : Appelée aussi la définition, c'est le nombre de pixels constituant l'image. Cette dernière est composée d'une grille de pixels. Ces pixels sont autant de petits carrés porteurs d'une information de couleur élémentaire.
- **La Résolution** : La résolution d'une image est le nombre de pixels par pouce (dpi) qu'elle contient (1 pouce = 2.54 centimètres). Elle peut être exprimée en « ppm » (point par millimètre). Une résolution plus élevée signifie plus de détails sur l'image.
- **Le Poids** : Le poids représente la taille de l'image en octets. Il dépend de la définition (le nombre de pixels) et le codage de la couleur de l'image.
- **Le codage de la couleur** : Le pixel est codé par un ou plusieurs bits. Ce nombre de bit détermine la couleur de pixel (1 bit pour l'image noir et blanc, 8 bits pour l'image niveau de gris, 24 bits pour l'image couleur). C'est ce qu'on appelle le codage de la couleur.

I.3 Traitement d'image :

Le traitement d'image numérique est l'ensemble d'outils et de techniques permettant de modifier une image numérique dans le but d'améliorer sa qualité d'en extraire des informations. Le traitement d'image est un traitement de signal 2D. L'étape de traitement d'image vient après celles d'acquisition dans la chaîne de production d'image. Parmi ses applications : la télédétection, l'imagerie médicale, les applications militaires, route intelligente, reconnaissance de caractères ... [7]

Le traitement d'image se divise en deux grandes parties :

- Le prétraitement
- L'analyse

I.3.1 Prétraitements :

Les prétraitements sont un ensemble de méthodes qui s'appliquent sur l'image acquise afin de faciliter l'extraction des informations. L'image est souvent distordue ou dégradée à cause d'un éclairage non uniforme ou une perturbation lors de l'acquisition.

Il existe 3 grandes familles de procédés pour corriger une image :

- **L'amélioration** : elle consiste à corriger l'aspect visuel de l'image c'est-à-dire une image plus contrastée. L'information contenue dans le pixel ne change pas. Parmi les méthodes de l'amélioration : L'extension dynamique, la correction d'exposition, l'égalisation d'histogramme, le renforcement de contraste...
- **Le dé-bruitage** : il consiste à restaurer l'information (la valeur de luminance originale) contenue dans le pixel. Cette information était infectée par des bruits (variation aléatoires) lors de l'acquisition ou la numérisation. Pour minimiser ce bruit, on applique les filtres cités ci-dessous : Le filtre moyenneur, le filtre gaussien, le filtre médian, le filtre de Nagao ...
- **Le flou** : C'est un phénomène qui masque les détails fins et diminuer les caractéristiques des contours. Ce phénomène est causé par un mouvement uniforme ou non uniforme. L'utilisation d'un filtre accentueur (*Sharpen filter*) ou un filtre passe-haut sur une image réduit l'effet de flou.

I.3.2 L'analyse (le traitement) :

L'analyse des images est un processus de décomposition de l'image prétraitée dans le but de trouver, détecter et reconnaître les objets contenus dans la scène observée, afin d'avoir une description du contenu sémantique de la scène.

L'analyse d'images est un domaine très vaste qui a connu, et qui connaît encore, un développement important depuis quelques dizaines d'années.

La segmentation est l'un des processus de base du traitement d'image numérique et de la vision par ordinateur. Elle consiste à définir des régions présentant des similitudes internes, telles que similitudes statistiques, géométriques, de forme... Une image est segmentée en régions pour permettre un traitement et une analyse indépendants de chaque région en fonction d'une politique ou d'un objectif de traitement. Les régions couvrent des zones plus petites que l'image globale mais généralement plus grandes que les entités ponctuelles locales, de sorte qu'une application peut utiliser des métriques régionales et locales de petits points d'intérêt (keypoints) en tant que signature d'objet.

La segmentation se base sur plusieurs méthodes. Parmi ces méthodes :

- Segmentation morphologique.
- Segmentation par couleur.
- Segmentation par seuillage.
- Segmentation par détection de contour.

La détection des contours est un outil courant dans le traitement d'images numériques. Il consiste à identifier les points d'une image numérique où la luminosité de l'image change fortement ou, plus formellement, présente des discontinuités. Les points auxquels la luminosité de l'image change brusquement sont généralement organisés en un ensemble de segments de ligne incurvés appelés contours. La détection de contour facilite la reconnaissance de formes et l'identification d'objets.

Il y a plusieurs méthodes pour détecter les contours dans l'image. Ces méthodes sont des filtres qu'on applique sur l'image. Nous citons quelques-uns : Filtre de *Prewitt*, filtre de *Sobel*, filtre de *Canny* ...

251	251	247	243	215	178	227	239	235	235	235	231	231	227	239
121	130	142	158	154	134	199	239	235	235	235	231	231	231	231
77	73	73	77	73	81	215	235	235	235	235	231	227	227	239
81	81	81	81	81	81	182	239	239	239	235	235	231	227	231
73	73	77	81	81	85	219	239	239	239	231	231	231	227	239
73	73	73	77	77	89	186	243	239	239	235	235	231	227	231
69	69	73	77	77	89	223	243	239	239	231	231	231	227	239
69	69	73	69	77	89	194	243	239	239	235	231	231	227	235
69	69	73	73	77	89	223	243	239	239	231	231	231	231	219
69	69	73	73	77	89	194	243	239	239	235	231	231	231	239
69	69	69	73	77	81	150	178	194	210	223	227	227	231	243
69	65	69	73	73	77	81	81	85	85	89	93	101	105	117
57	57	61	65	65	69	69	69	69	69	73	73	77	77	77
49	49	49	49	53	53	53	53	57	57	57	57	61	61	61
49	53	53	57	61	57	57	53	49	49	49	49	49	49	49

Figure I-3 : Matrice de pixel montrant les contours

Le filtre de *Canny* est l'un des algorithmes les plus connus pour la détection de contour. Il a été développé par *John Canny* en 1986. Il se décompose en étapes suivantes :

- Minimiser le bruit en appliquant un filtre gaussien.
- Calculer le gradient d'intensité de l'image.
- Suppression du non-maxima pour éliminer les parasites lors de la détection de contour.
- Appliquer un double seuillage.
- Différencier les contours par seuillage hystérésis.

L'algorithme de détection de contour de *Canny* est contrôlé par deux paramètres :

- La taille du filtre appliqué (ex : filtre gaussien de 3x3, 5x5 ou 7x7).
- Les deux seuils d'hystérésis.

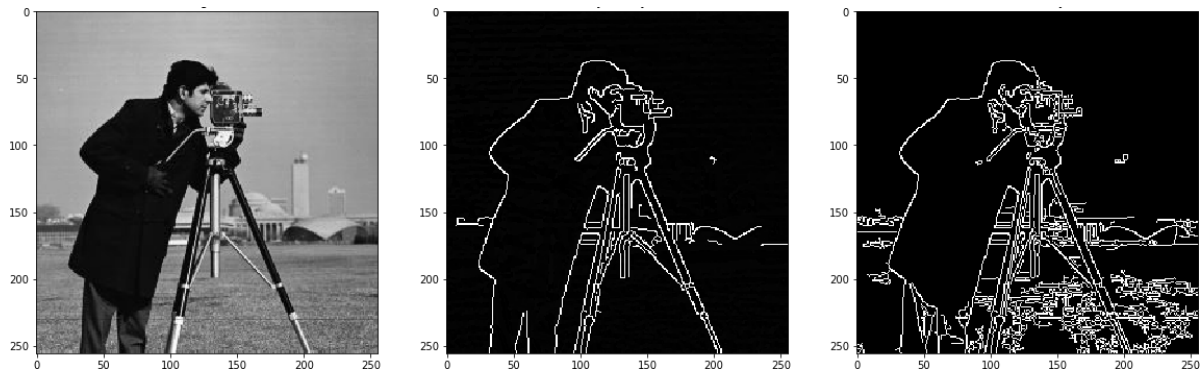


Figure I.4 : Détection de contour par méthode de Canny

I.4 Conclusion :

Dans ce chapitre nous avons présenté une vue générale sur l'image, plus précisément l'image numérique. Ou on a abordé les caractéristiques de cette dernière suivie par ces techniques de traitement.

Dans le chapitre suivant nous allons voir le hardware & le software utilisés dans la réalisation de notre projet.

Chapitre 2 : Descriptions matérielle et logicielle de la réalisation

II.1 Introduction

Le but de notre projet de fin d'étude est de concevoir un système RAPI comme un système embarqué. Et vu qu'on a besoin d'un système qui peut accepter des capteurs de présence et une caméra comme entrées, et des signaux de commande en sorties pour manipuler des LEDs, des moteurs, nous avons choisi le Raspberry Pi comme carte électronique pour notre système. Ainsi pour le langage de programmation, nous avons choisis Python car il est intégré dans linux, le système d'exploitation utilisé sur Raspberry Pi.

II.2 Raspberry Pi :

Le Raspberry Pi est un mini-ordinateur mono-carte, souvent utilisé avec le système d'exploitation linux mais il peut supporter d'autres systèmes d'exploitation comme Windows par exemple. Il est destiné à des applications d'informatique embarquée. Le cœur de l'ordinateur est un FPGA (Broadcom2836) intégrant un processeur quad-core ARM11 Cortex A7 cadencé à 700 MHz, de 256 Mo jusqu'à 4 Go de RAM selon la version et de nombreux périphériques.

Le Raspberry Pi peut être directement connecté à une IHM classique, souris/clavier/écran HDMI ou vidéo composite. Le connecteur d'extension supporte les entrées/sorties parallèles ainsi que la plupart des bus de communication comme le bus CAN, Ethernet, I2C... C'est un support particulièrement économique et puissant qui peut être facilement mis en œuvre dans des systèmes nécessitant un accès au monde physique par des capteurs/actionneurs disposant d'interfaces numériques.[15]



Figure II-1: Raspberry PI 4 model B

II.2.1 Historique :

Le Raspberry PI, ce petit ordinateur, était créé par des étudiants qui voulaient faire l'ordinateur le moins cher du monde afin d'encourager l'apprentissage de la programmation. En 2006, Eben Upton étudiant de la faculté des sciences informatiques de l'université de Cambridge en Angleterre et ses collègues ont réalisés le premier prototype sur l'Atmel AtMega 664. Ce premier prototype était de taille d'une clé USB, et contient qu'un port USB et HDMI comme périphériques d'entrées/sorties. Les premières cartes sont produites en Mars 2012 à Taiwan et en Chine. Ils étaient équipés d'une RAM de 256Mo et un port RJ45. Puis la production est passée au Royaume-Uni en 2013-2014. Aujourd'hui la fondation Raspberry PI annonce plus de 6 millions de Raspberry PI qui ont été vendus dans le monde entier.[15][16]

II.2.2 Evolution :

Le développement de l'ordinateur Raspberry Pi est un processus continu. Jusqu'à nos jours, la fondation Raspberry Pi a publié 12 modèles sur 4 générations d'ordinateur. Les modèles sont des versions mises à niveau de leurs versions précédentes pour rendre l'ordinateur plus efficace et pratique pour les utilisateurs, notamment en raison de sa faible consommation énergétique et de ses périphériques d'entrées/sorties. Les différences entre ces modèles sont le CPU utilisé, la taille de la RAM, la taille du GPIO, le nombre de ports USB, les périphériques réseau (Ethernet, WIFI, Bluetooth) et les périphériques de multimédia.[15]

La figure suivante présente les différentes générations du Raspberry Pi :

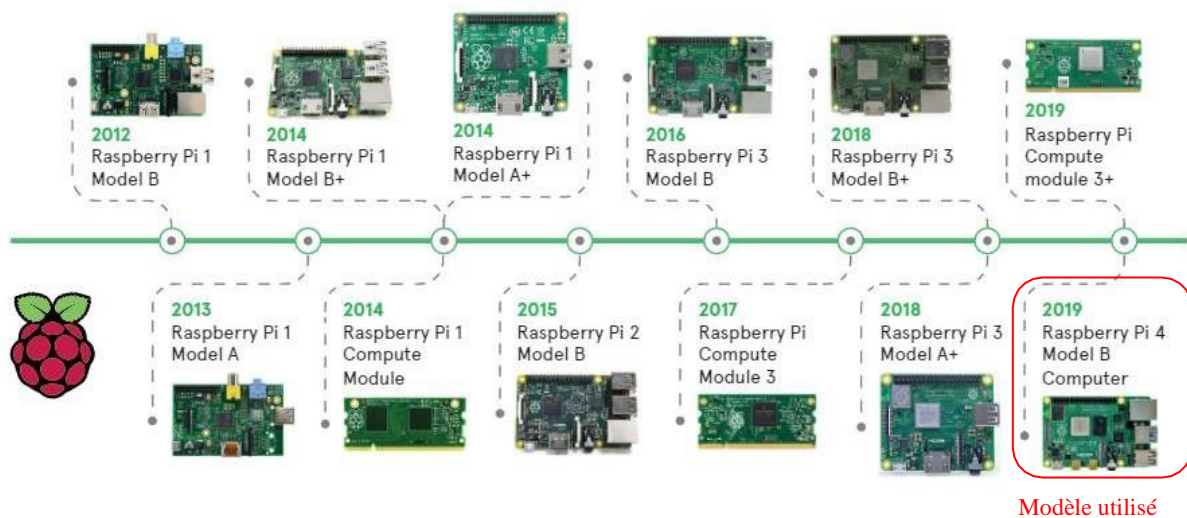


Figure II-2 : Evolution du Raspberry PI

II.2.3 Composants de base :

Le Raspberry Pi 4 model B est le dernier produit de la gamme d'ordinateurs Raspberry Pi. Il offre des augmentations révolutionnaires de la vitesse du processeur, des performances multimédias, de la mémoire et de la connectivité par rapport au Raspberry Pi 3 model B+ de la génération précédente, tout en conservant une compatibilité descendante et une consommation d'énergie similaire. Pour l'utilisateur final, le Raspberry Pi 4 modèle B offre des performances de bureau comparables aux systèmes PC x86 d'entrée de gamme. C'est pour cela que nous l'avion choisit.[15]

Les principales caractéristiques de ce produit sont :[15][17]

- Un processeur 64 bits quad-core ARM Cortex A72 cadencé à 1.5GHz intégré dans FPGA Broadcom2711.
- La prise en charge du double affichage à des résolutions allant jusqu'à 4K via une paire de ports micro-HDMI.
- Le décodage vidéo matériel jusqu'à 4Kp60
- Une Capacité de RAM jusqu'à 8 Go.
- Double bande LAN sans fil 2.4/5.0 GHz + Bluetooth 5.0

Chapitre 2

- Gigabit Ethernet.
- Capacité PoE (via un module complémentaire PoE HAT distinct).

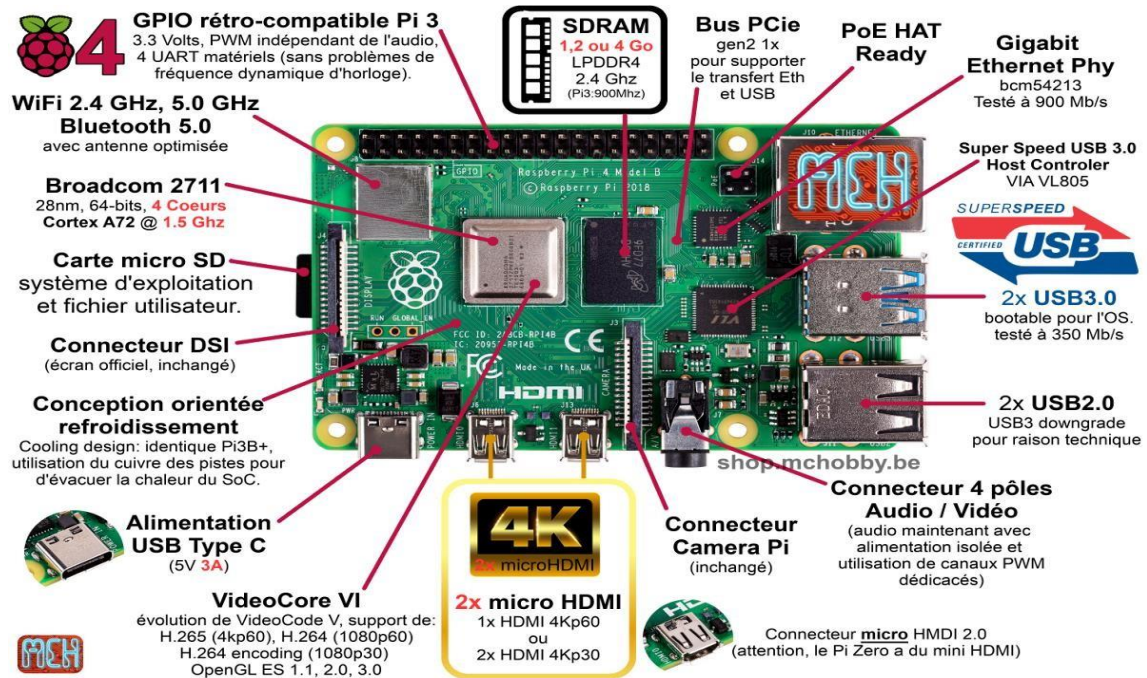


Figure II-3 : Composants de base du Raspberry Pi 4

II.3 Linux :

Avec la montée du concept 'open-source' et des logiciels libres, le monde du calcul informatique a grandement évolué en puissance et applications, et plus particulièrement dans le domaine des systèmes embarqués.

Linux est un système d'exploitation Open source basés sur le noyau Unix. C'est un système fiable, flexible et d'un niveau de sécurité très haut. Linux est intégré sous plusieurs distributions comme Ubuntu, Debian... Les distributions incluent le noyau Linux, les logiciels et les bibliothèques du système de support. Linux est le principal système d'exploitation sur les serveurs et les ordinateurs centraux, mais également sur les systèmes embarqués tel que les routeurs, les SmartTVs, les automobiles, les consoles de jeux et même les Smartphones qui utilisent Android, un dérivé de Linux.[19]

II.3.1 Historique :

Le noyau Linux a été réalisé en septembre 1991 par le finlandais Linus Torvalds, qui a décidé de le publier sous licence libre. Linux a connu une forte adoption au milieu des années 1990 par plusieurs organisations telles que la NASA, IBM, DELL. Aujourd'hui, Linux est utilisé dans la plupart des outils informatiques, depuis les systèmes embarqués jusqu'aux super ordinateurs. Le plus grand succès de Linux sur le marché des consommateurs est peut-être le marché des appareils mobiles, Android étant l'un des systèmes d'exploitation les plus dominants sur les Smartphones et très populaire sur les tablettes.[19]

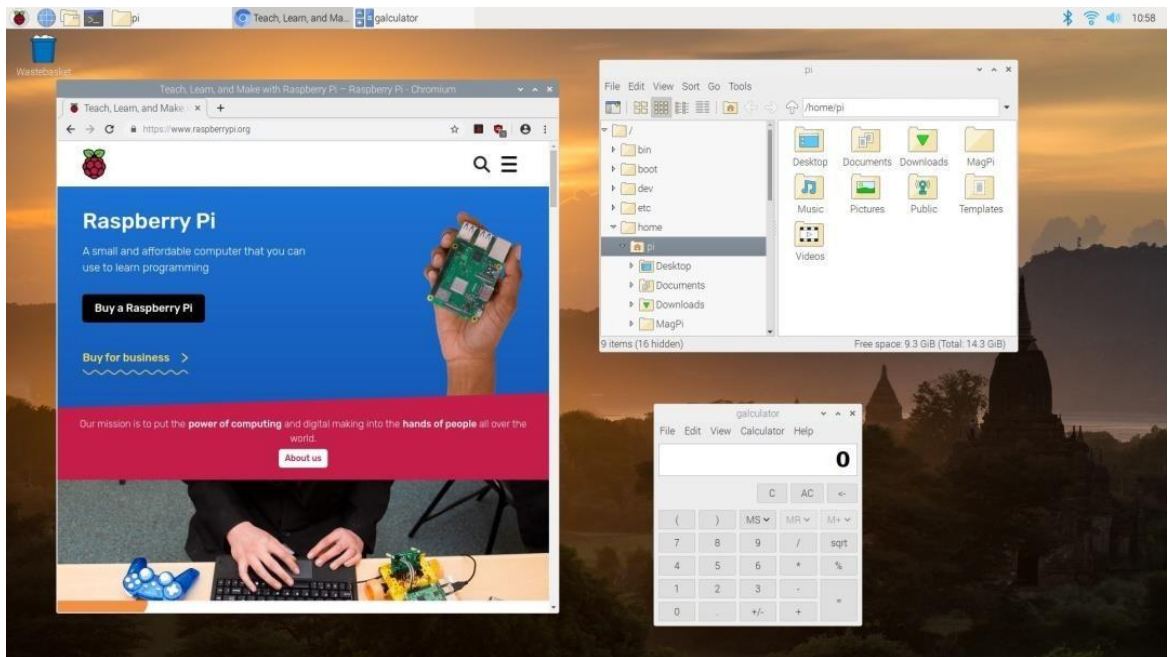


Figure II-4 : Interface système d'exploitation Linux-Debian

II.4 Python:

Python est un langage de programmation merveilleux, puissant et facile à utiliser (facile à lire est à écrire). Il permet l'interaction de Raspberry Pi avec le monde réel. La syntaxe Python est très propre, avec un accent sur la lisibilité. Il utilise des mots clés en anglais standard comme : print, input, if...

Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes (recyclage de mémoire) et d'un système de gestion d'exceptions.

Le langage python est placé sous une licence libre. Il fonctionne sur la plupart des plateformes informatiques, des Smartphones aux ordinateurs centraux, de Windows à Unix avec notamment GNU/Linux en passant par MacOS, ou encore Android, IOS. Il peut aussi être traduit en Java ou .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser.[18]



Figure II.5 : Logo Python

II.4.1 Historique :

En 1980, Guido van Rossum a conçu Python aux Pays-Bas afin de développer un langage de programmation interfaçable. En février 1991, la première version publique était sortie nommée Python 0.9.0. Puis Rossum a passé en Amérique pour développer son langage qui a connu plusieurs versions. En 2001, Python Software foundation a été créée pour que Python soit sous une licence libre. Aujourd'hui Python est arrivé à la version 3.8.[18]

```
1 def fact(n):
2     if n==1:
3         return 1
4     else:
5         return n*fact(n-1)
6 n=int(input())
7 print(fact(n))
8
```

Figure II.6: Programme factoriel avec Python

III OpenCV :

Open Source Computer Vision Library en anglais, est une librairie ou bibliothèque qui contient un ensemble de fonctions de programmation dédiées à la vision par ordinateur (traitement d'image ou vidéo) en temps réel et d'apprentissage automatique. Cette bibliothèque est développée par Intel en 1999 dans le but d'améliorer les applications intensives du CPU. OpenCV est constituée de plus de 2500 algorithmes. Ces algorithmes peuvent être utilisés pour détecter et reconnaître des visages, identifier des objets, classer les actions humaines dans des vidéos, suivre les mouvements de caméra, suivre des objets en mouvement, extraire des modèles 3D d'objets, produire des nuages de points 3D à partir de caméras stéréo, assembler des images pour produire une haute résolution l'image d'une scène entière, trouver des images similaires dans une base de données d'images, supprimer les yeux rouges des images prises au flash, suivre les mouvements des yeux, reconnaître les paysages et établir des marqueurs pour les recouvrir de réalité augmentée, etc.[20]

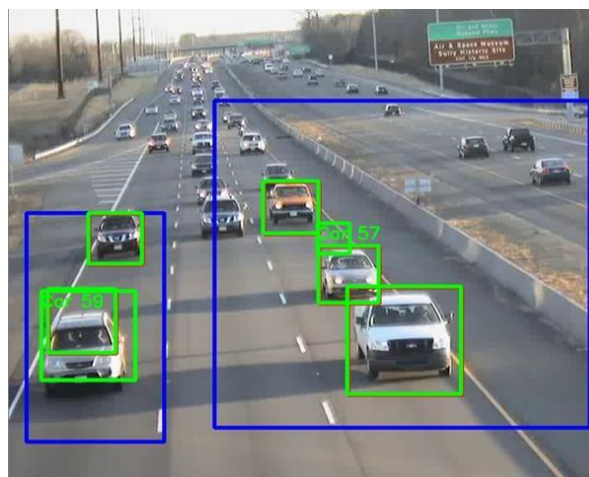


Figure II-7 : Localisation et suivi des véhicules avec OpenCV

II.6 Conclusion

Dans ce chapitre nous avons présenté le Raspberry PI, ainsi que le langage de programmation, en donnant quelques détails techniques, qui vont nous permettre de faire intégrer notre système.

Dans le chapitre suivant nous allons voir le systèmes RAPI, pour mieux comprendre et maîtriser le principe de fonctionnement. Pour que par la suite on sera on mesure de faire l'implémentation du système RAPI dans un système embarqué.

Chapitre 3 : Les Systèmes RAPI

III.1 Introduction :

Les systèmes RAPI (Reconnaissance Automatique des Plaques d'Immatriculation) est une technologie qui sert à automatiser la lecture et l'identification des plaques d'immatriculation. Cette technologie a trouvé sa naissance dans les années soixante-dix, son développement est lié aux techniques de traitement d'image ainsi que les techniques OCR (Optical Character Recognition, en anglais).

Nous étudions dans ce chapitre les systèmes RAPI : l'historique, le principe de fonctionnement, la composition, les méthodes de détection de la plaque d'immatriculation et les domaines d'applications.

Nous allons nous concentrer sur les systèmes RAPI pour l'accès au parking car c'est notre cas d'étude.

III.2 Les systèmes RAPI :

Les systèmes RAPI ou aussi connue lecture automatique de plaques minéralogiques (ANPR ou sous divers autres termes comme LAPI, AVI, CPR, LPR... en anglais) est un système de surveillance de flux de véhicule, qui utilise le traitement d'image et la reconnaissance optique de caractères sur des images ou des vidéos pour détecter et lire les plaques d'immatriculation des véhicules. Ils peuvent aussi être utilisés pour stocker les images capturées par les caméras : l'image du véhicule ainsi qu'une photo du conducteur.[14]

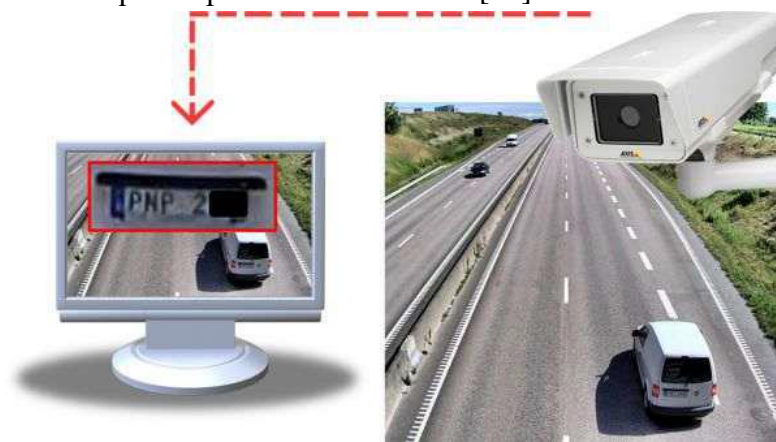


Figure III-1 : Système RAPI

III.2.1 Historique :

Les systèmes RAPI ont été inventés en 1976 par la *Police Scientific Development Branch* au Royaume-Uni. Les premiers prototypes ont été mis en œuvre en 1979, des contrats ont permis de produire des systèmes industriels, d'abord chez EMI Electronics, puis à *Recognition Systeme* dans *Wokingham*, Royaume-Uni. La première arrestation d'une voiture volée par un système RAPI a été réalisée en 1981.[14]

III.2.2 Composition des systèmes RAPI :

Les systèmes RAPI se composent de 3 blocs : Capteurs, Système d'analyse et de commande et Actionneurs.

Les systèmes RAPI se basent essentiellement sur une caméra comme un capteur. Cette caméra est liée au système d'analyse et de commande qui peut être un ordinateur ou un microcontrôleur. Les actionneurs peuvent être des barrières, des significations lumineuses.

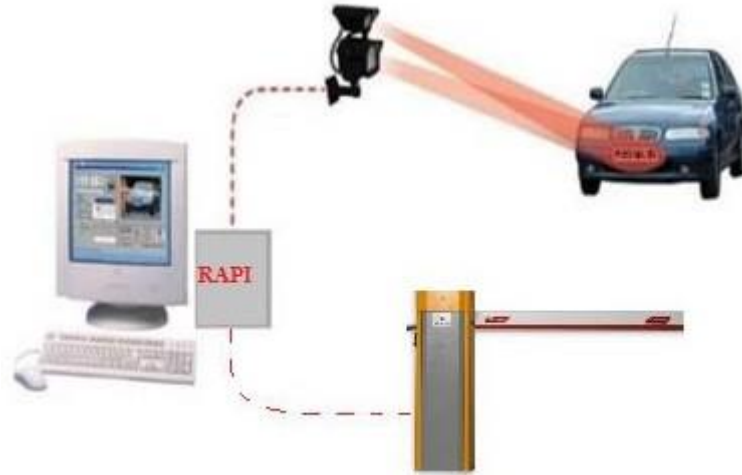


Figure III-2 : Composition des systèmes RAPI

III.2.3 Principe de fonctionnement :

Généralement un système RAPI est divisé en 4 grandes fonctions : [14]

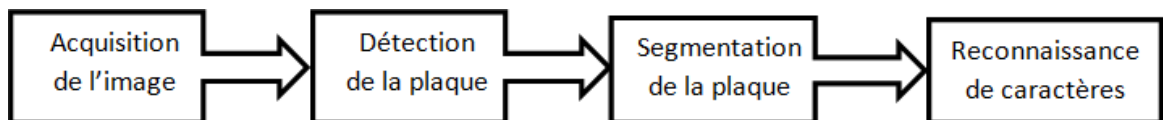


Figure III-3 : les principales fonctions d'un système RAPI

- L'acquisition de l'image : se fait à partir d'une séquence vidéo ou bien une image capturée avec un appareil photo, et son envoi vers le système.
- La détection de la plaque : il s'agit de chercher les coordonnées de la plaque dans l'image en utilisant de nombreuses méthodes. On va les décrire en détail par la suite. L'image doit être prétraitée pour avoir des bons résultats. Cette phase est complexe et importante. Elle détermine la rapidité et la robustesse du système.
- La segmentation : la plaque une fois détectée subira un ensemble de traitement pour être segmentée en séparant les caractères.
- La reconnaissance de caractère : se fait par les moteurs OCR (la reconnaissance optique de caractères). Ces systèmes doivent être capables de gérer un grand nombre de difficultés possibles. On va les citer par la suite dans les problèmes du système RAPI.

III.2.4 Méthodes de détection des plaques d'immatriculation :

Les systèmes de RAPI utilisent en général des séries de techniques et méthodes de traitement d'image afin de détecter, repérer et extraire les plaques d'immatriculation de l'image. Parmi ces techniques, nous citons :

III.2.4.1 Technique basée sur les propriétés de la plaque

Cette technique utilise les propriétés suivantes :

- La couleur de la plaque.
- Segmentation d'image noir et blanc après l'utilisation des seuils.
- La taille des caractères et la distance qui existe entre eux.

Les algorithmes basés sur les couleurs ne sont pas efficaces car les couleurs changent en fonction d'éclairage, donc la détection d'autres régions non correspondantes à la plaque.



Figure III-4 : Détection de plaque d'immatriculation par couleur

III.2.4.2 Technique basée sur les contours de la plaque :

Cette technique utilise les contours des caractères et des plaques comme des points de référence pour l'extraction. L'intensité des pixels dans les contours des caractères et des plaques est complètement différente que celle des voisins.

Le point faible de cette technique est lorsque les bords de la plaque ne présentent pas une grande variation d'intensité par rapport au reste de l'image.



Figure III-5 : Détection de plaque d'immatriculation par contour

III.2.4.3 Technique basée sur l'intelligence artificielle :

Cette technique utilise :

- Un réseau de neurones à deux entrées comme un filtre horizontal et vertical pour détecter les plaques, de telle façon l'intersection entre les deux filtres localise la région de la plaque.
- Un réseau de neurones pour déterminer les régions susceptibles qu'on va analyser par la suite par une transformé de fourrier pour détecter la bonne région, l'algorithme sera répété jusqu'à la détermination de la plaque. Mais la réussite de cet algorithme dépend de l'éclairage.

III.2.4.4 Technique basée sur la signature de la plaque :

Cette technique utilise un des algorithmes suivants :

- L'algorithme d'Adaboost.
- Un algorithme qui choisit une région du texte à partir d'un ensemble puis il exploite de telle façon une plaque contient des caractères et/ou des symboles qui sont clairement visibles. Le problème est que certaines autres régions peuvent contenir du texte.

III.2.6 Caractéristiques des plaques d'immatriculation algériennes :

La plaque d'immatriculation en Algérie permet, comme tous types de plaques minéralogiques, d'identifier les véhicules. Elle a une forme rectangulaire avec des taille standardisée, et de couleur blanche à l'avant et jaune à l'arrière. Le système d'immatriculation algérien est composé de trois groupes séparés de chiffres (NNNNNN TAA WW):[3]



Figure III-6 : Système de plaque d'immatriculation algérienne

- Le premier groupe (NNNNNN), contient jusqu'à six chiffres au maximum, ce qui correspond au numéro du véhicule.
- Le deuxième groupe (TAA), contient trois chiffres, le premier chiffre (T) permet d'identifier le type de véhicule, les deux suivants (AA) renvoient à l'année de mise en circulation du véhicule.
- Le troisième groupe (WW), contient deux chiffres qui identifier la wilaya d'immatriculation.

Certaines plaques d'immatriculations varient dans la taille des caractères et leur position. Les systèmes de lecture automatique des plaques minéralogiques doivent savoir traiter ces différences pour être vraiment efficaces. Les systèmes les plus évolués savent gérer les variations entre pays, bien que beaucoup de programme soient spécifiques à un pays.

III.2.7 Problèmes des systèmes RAPI

Il y'a un certain nombre de difficultés possibles que les systèmes RAPI ne peuvent prendre en considération. Par exemple :[4]

- Pauvre résolution de l'image, généralement parce que la plaque est trop loin mais parfois résultant de l'utilisation d'une caméra de faible qualité.
- Les images floues, en particulier le flou de mouvement.
- Mauvais éclairage et de faible contraste en raison de surexposition, réflexion ou ombres.
- Un obscurcissement (occultation) d'objet (une partie de la plaque), bien souvent une barre de remorquage, ou de la saleté sur la plaque.
- Un type de police différente (certains pays n'autorisent pas de telles plaques, l'élimination du problème).
- Technique de retournement.

III.2.8 Domaines et exemples d'utilisation des systèmes RAPI

III.2.7.1 Le péage urbain

Le système de péage urbain fonctionne grâce à l'analyse des plaques d'immatriculation enregistrées par un réseau de caméras de surveillances disposées en entrées et sorties des zones de péage des villes, autoroutes, ...etc. il compare ensuite les données recueillies avec la base de données centralisée pour vérifier dans les délais de paiement effectif. Ce type de système permet de réduire et faciliter la circulation.[4]



Péage urbain manuelle



Péage urbain automatique

Figure III-7 : Péage urbain

III.2.7.2 L'accès au parking

C'est presque le même principe du péage urbain, l'analyse des plaques d'immatriculations se fait avec une caméra, puis le système fait une comparaison avec la base de données pour donner l'accès au parking.[4]

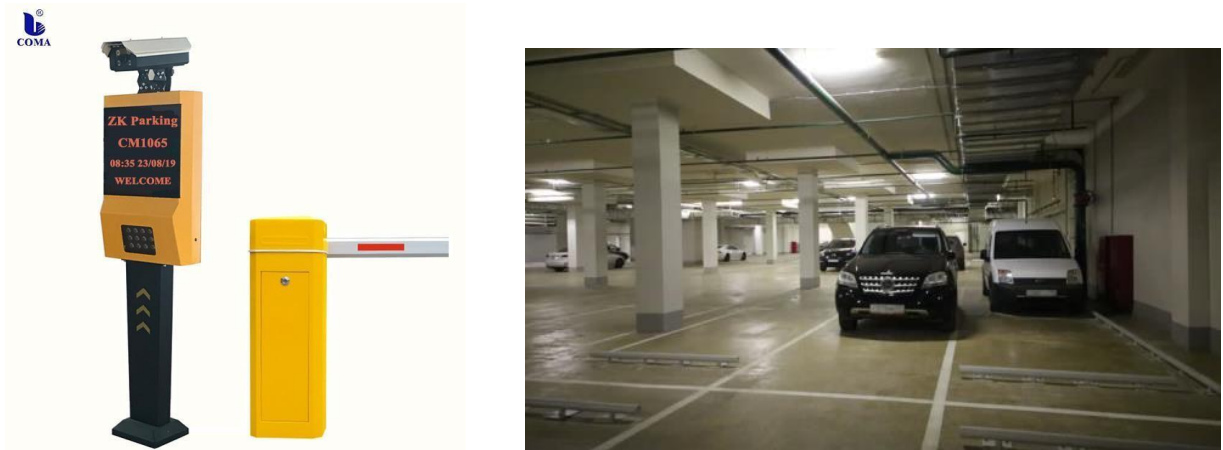


Figure III-8 : système d'accès automatique au parking

Comme il existe plusieurs autres domaines d'utilisation des systèmes de RAPI :[4][14]

- Les passages frontaliers.
- Système de gestion de trafic.
- Les forces de l'ordre (police, gendarmerie, ministère de la justice), pour trouver les voitures recherchées.
- Le contrôle de la section, pour mesurer la vitesse moyenne du véhicule sur des distances plus longues.

III.3 Conclusion

Dans ce chapitre nous avons présenté une vue générale sur le système RAPI, où nous avons pu comprendre le fonctionnement du système. Nous avons abordé aussi les problèmes à affronter pour avoir un système robuste. En dernier nous avons présentés les techniques de reconnaissances de plaques d'immatriculation suivi par quelques exemples et domaines d'utilisation.

Cela va nous permettre d'effectuer l'implémentation de notre système RAPI, qui sera le sujet du chapitre suivant.

Chapitre 4 : Implémentation du **Système RAPI**

IV.1 Introduction :

Dans ce chapitre, nous allons présenter la mise en œuvre du système développé RAPI : l'emplacement physique du système, l'environnement matériel (Hardware) et l'environnement logiciel (Software). Puis nous allons faire une analyse des résultats obtenus lors de sa mise en marche.

IV.2 L'emplacement physique de notre système :

Vu le nombre important de véhicules qui accèdent chaque jour au parking de notre université, nous avons décidé d'implanter notre système RAPI à l'entrée du parking de notre faculté. Voici des plans descriptifs qui identifient la position de chaque composant de notre système :

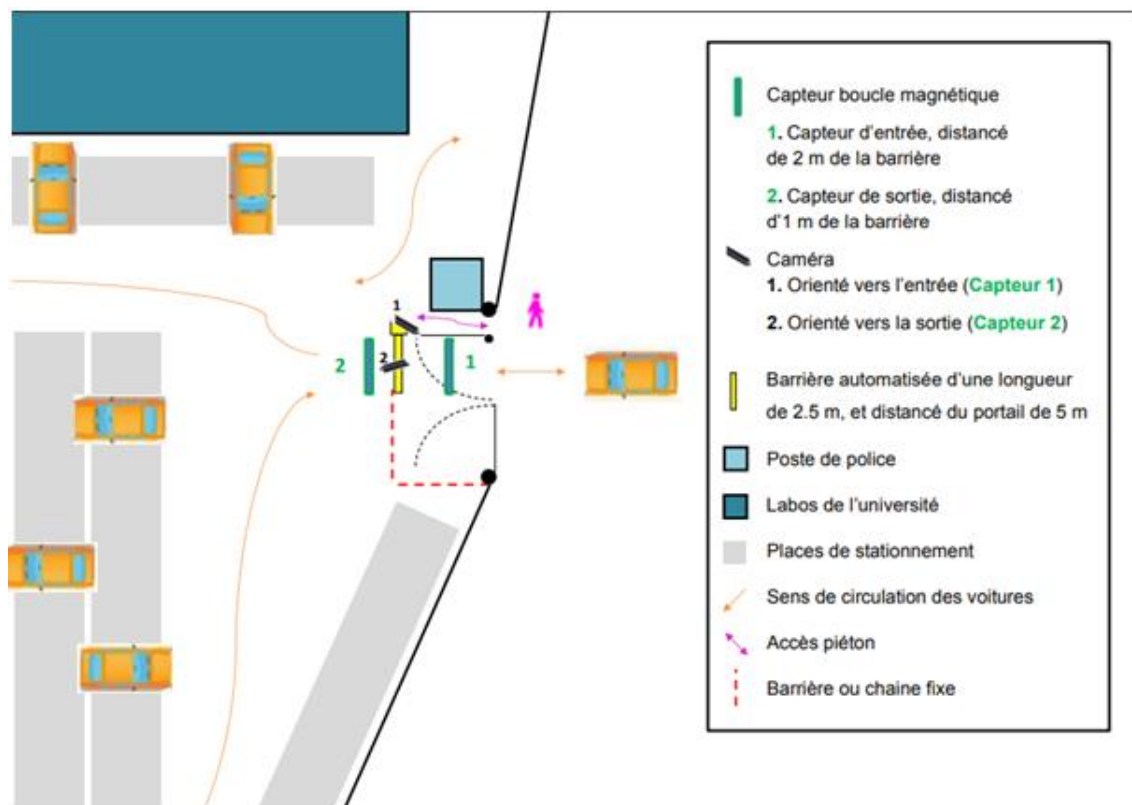


Figure IV-1 : Vue de dessus qui décrit l'emplacement de système RAPI dans le parking de notre faculté

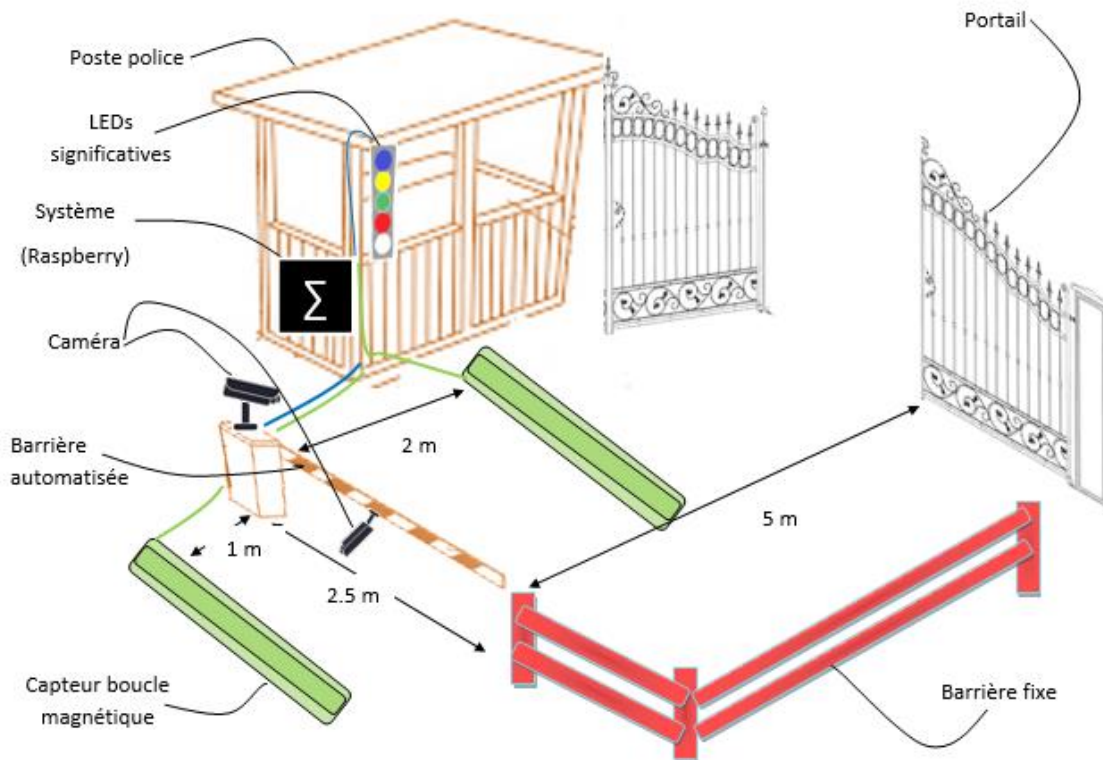


Figure IV-2 : Vue en plongée qui décrit l'emplacement de système RAPI dans le parking de notre faculté

IV.3 Le matériel : Raspberry et ses périphériques

Le système RAPI développé est constitué de :

- Un micro-ordinateur Raspberry Pi 4 Model B (voir chapitre2).
- 2 Webcam USB d'une résolution de (640x480) avec un capteur de taille (4.86x3.64mm²), sont utilisées pour acquérir des images des véhicules, l'une pour l'entrée et l'autre pour la sortie.
- 2 Boucles d'induction magnétique utilisées comme des capteurs de présence (voir annexe1).
- 1 servomoteur avec une vitesse de rotation de 60 degrés/0.1sec et de plage de tension de fonctionnement de 4.8V-6V. Il joue le rôle d'une barrière.

Chapitre 4

- Un système de feux colorés :

Signification des feux colorés :	
Bleu permanent	Système prêt à fonctionner
Bleu clignotant	Système occupé après passage
Blanc	Acquisition d'image par caméra(s)
Vert	Accès autorisé
Rouge	Accès non autorisé
Orange	Ouverture/Fermeture de la barrière

- La carte électronique qui interface le Raspberry PI aux périphériques :

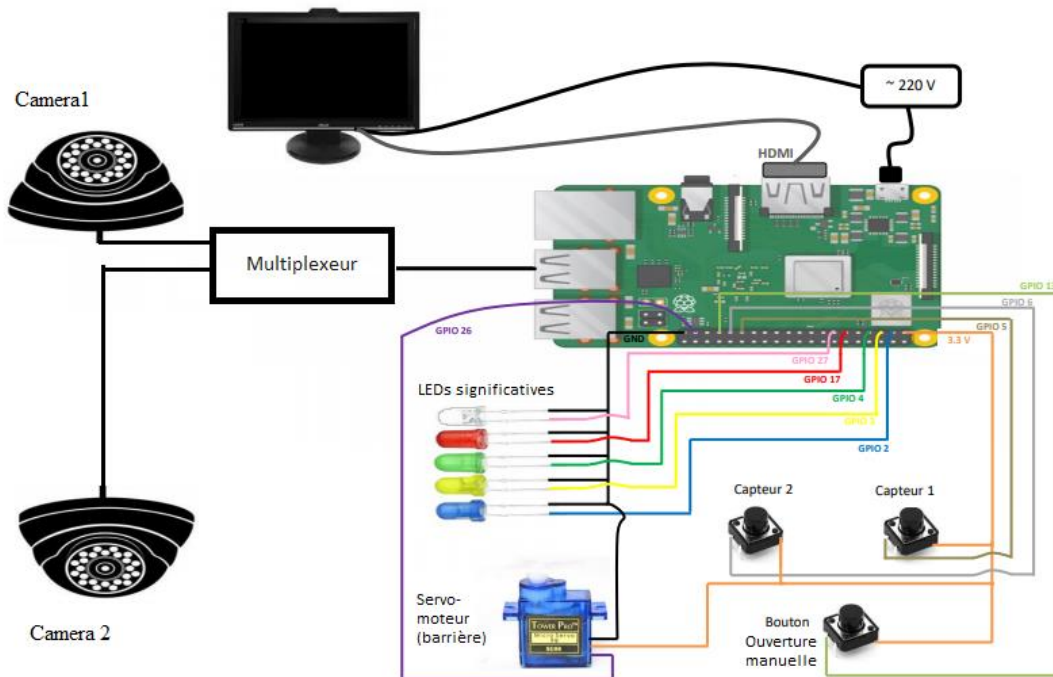


Figure IV-3 : Schéma électronique du système RAPI

IV.4 L'environnement logiciel (Software) :

La gestion des périphériques, leurs états et les actions sur ces derniers, et des différentes situations (entrées/sorties de véhicule) est réalisée sous forme de programmes implémentés sur Raspberry, écrits en langage Python sous Linux et font appel à la bibliothèque OpenCV (voir annexe 3).

IV.4.1. Présentation du fonctionnement du système :

Lorsque le système est en marche une LED bleue est allumée en permanence. Trois cas d'évènements sont considérés (Voir l'organigramme) :

- A. Présence d'un véhicule à l'entrée (procédure d'entrée) ;
- B. Présence d'un véhicule à la sortie (procédure de sortie) ;
- C. Fonctionnement manuel.

A. Procédure d'entrée :

Le système entame cette procédure si détection d'un véhicule à l'entrées (Capteurs : $C1=1$, $C2=0$) ; alors → La LED bleue s'éteint, la LED blanche s'allumera indiquant que la caméra prend une image du véhicule. L'image est envoyée à l'unité de traitement (Raspberry). Ensuite l'image acquise est traitée par la fonction RAPI pour extraire le numéro d'immatriculation. Deux situations peuvent se présentées :

- Numéro d'immatriculation reconnu → Le véhicule est autorisé à rentrer en indiquant au conducteur par clignotement de la LED orange pendant l'ouverture de la barrière, puis la LED verte s'allume pour signifier l'accès autorisé. Après que le capteur de présence C2 soit franchi (séquence d'activation et désactivation) par le véhicule entrant, la barrière se referme puis la LED bleue se rallume. Le NI est enregistré dans un historique d'entrées /Sortie.
- Numéro d'immatriculation non reconnu → Le véhicule n'est pas autorisé à rentrer. La LED rouge s'allume pour indiquer l'accès non autorisé. Le NI est enregistré dans un historique d'entrées /Sortie.

B. Procédure de sortie :

Dans le cas d'une voiture sortante (Capteurs : $C1=0$, $C2=1$) → La LED bleue s'éteint, la LED blanche s'allumera pendant que la caméra prend une image du véhicule et elle est envoyée à l'unité de traitement (Raspberry). Ensuite l'image acquise est traitée par la fonction RAPI pour extraire le NI et l'enregistrer dans l'historique d'entrées /Sorties. Ensuite la barrière s'ouvre et en même temps la LED orange clignote.

Après que le capteur C1 soit franchi (séquence d'activation et désactivation) par le véhicule sortant la barrière se referme puis la LED bleue se rallume.

C. Fonctionnement manuel :

Cette procédure est abordée lorsqu'un véhicule d'urgence ou prioritaire veut rentrer ; entrée manuelle (B.e.m=1) → La LED bleue s'éteint, la LED blanche s'allumera indiquant que la caméra prend une image du véhicule. L'image est envoyée à l'unité de traitement (Raspberry). Ensuite l'image acquise est traitée par la fonction RAPI pour extraire le numéro d'immatriculation et l'enregistrer dans un historique d'entrées/sorties. Ensuite la barrière s'ouvre et en même temps la LED orange clignote. Après que le capteur C2 soit franchi (séquence d'activation et désactivation) par le véhicule sortant la barrière se referme puis la LED bleue se rallume.

Voici ci-dessous un diagramme d'états et un organigramme expliquant la logique de fonctionnement du système :

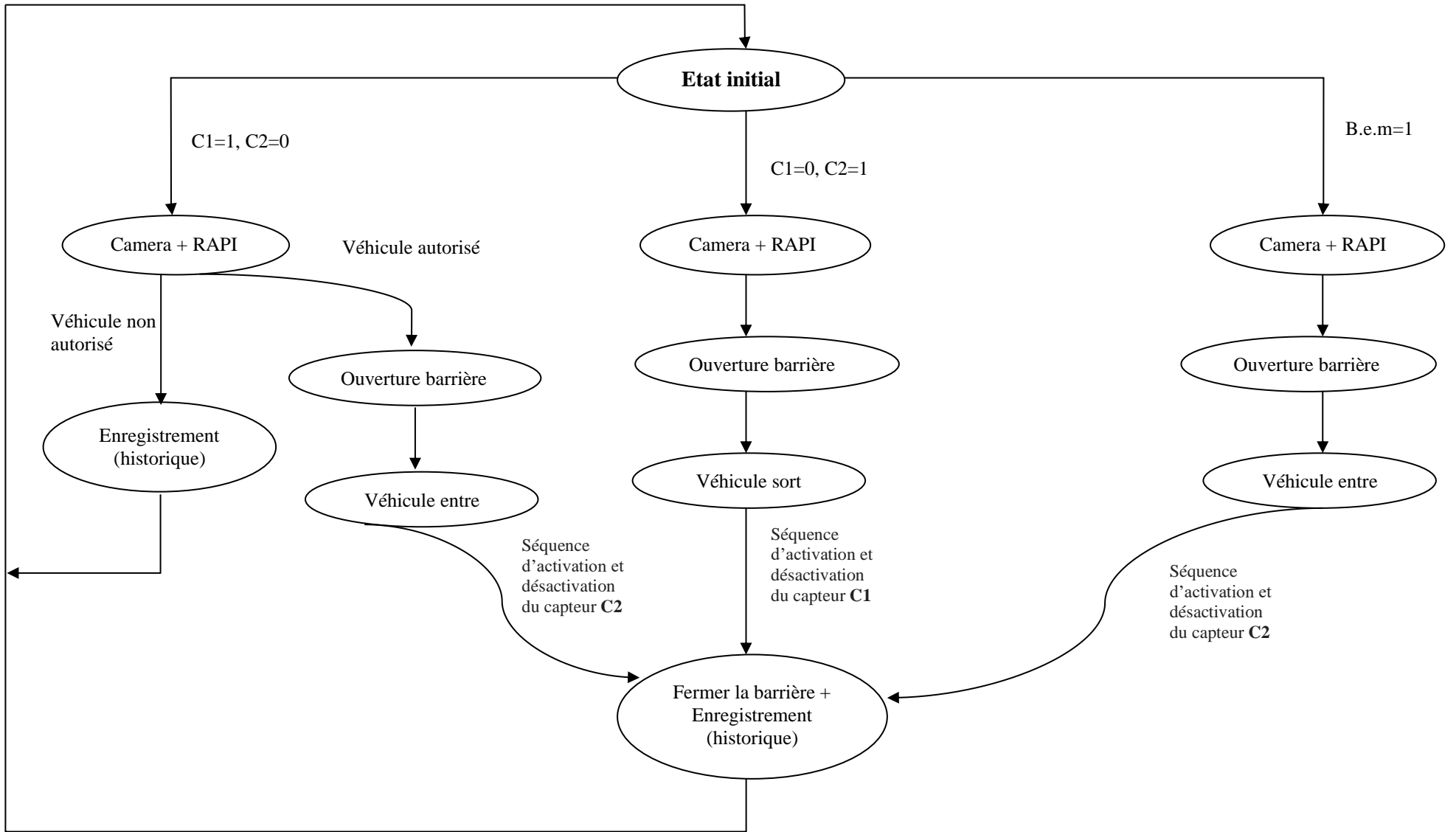


Figure IV-4 : Diagramme d'états de fonctionnement du système RAPI

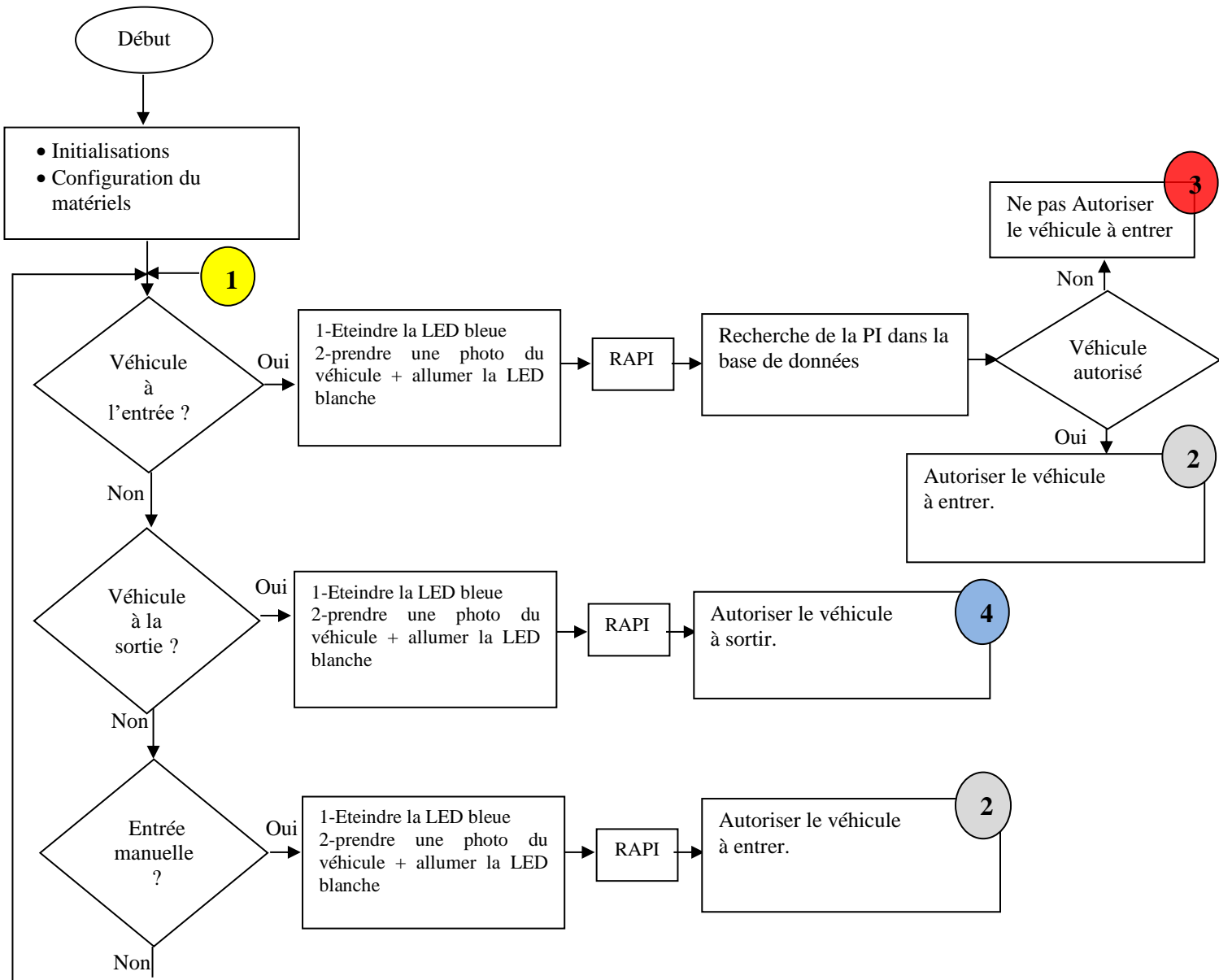
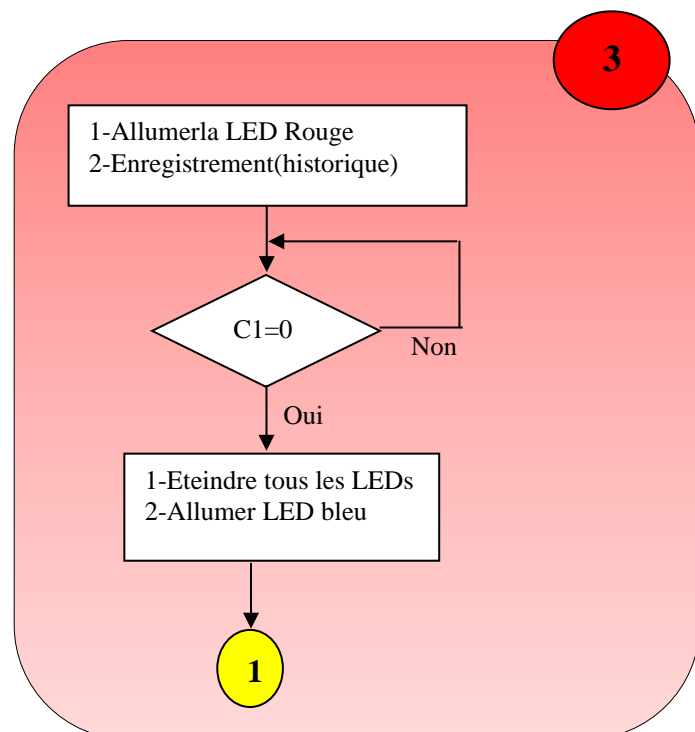
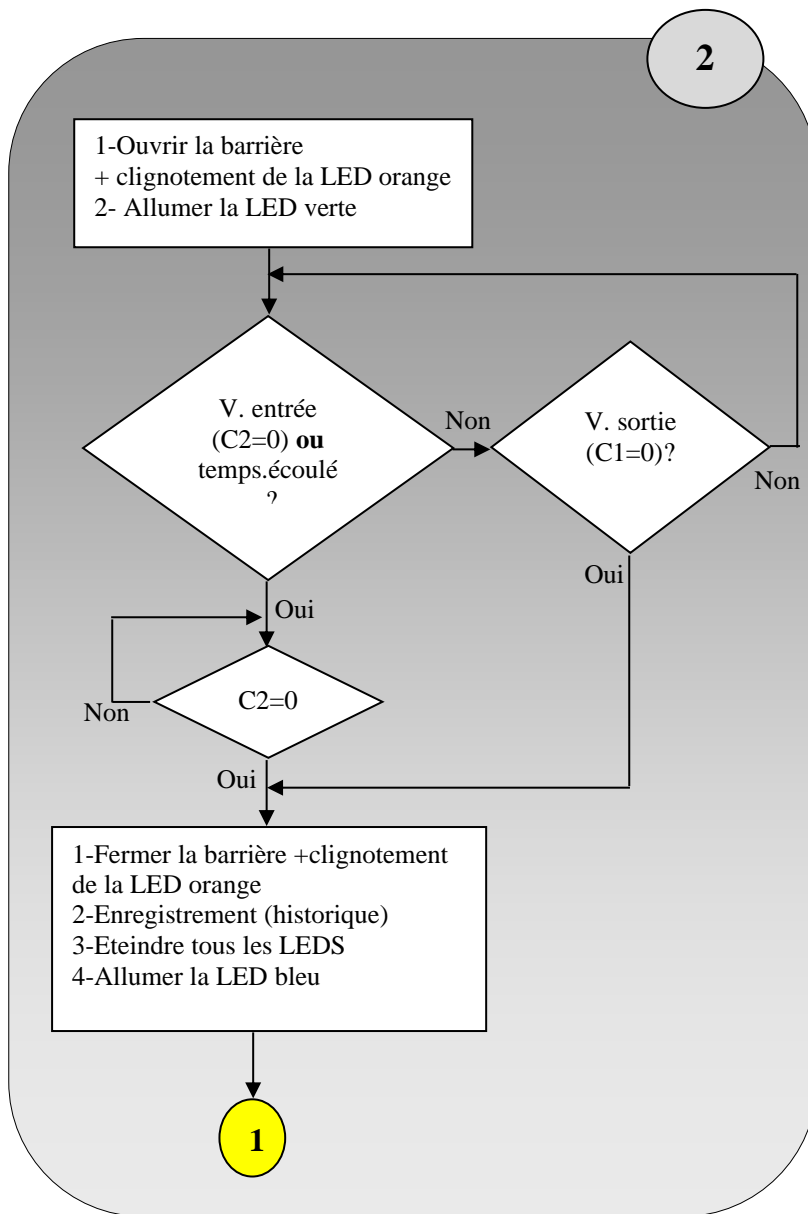


Figure IV-5 : Organigramme de fonctionnement du système RAPI



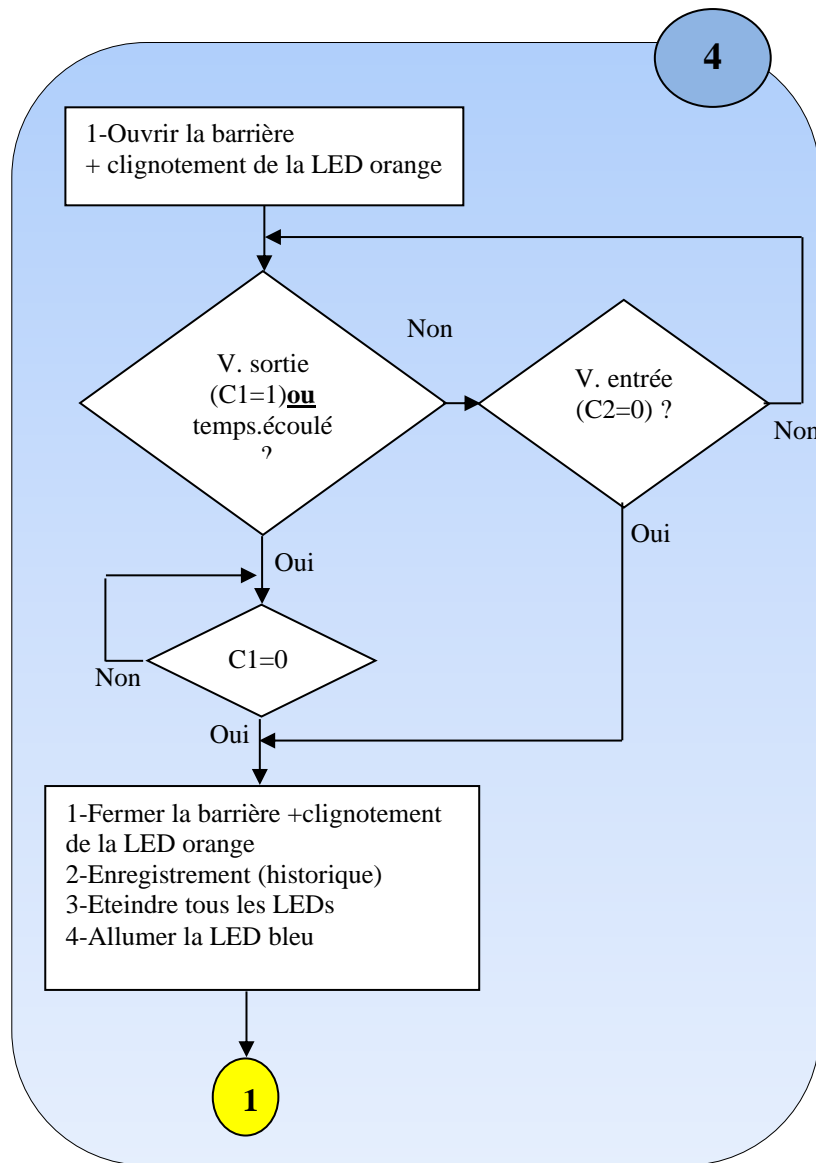


Figure IV-6 : Suite de l'organigramme de fonctionnement du système RAPI

IV.4.2. Détection et lecture de la plaque d'immatriculation (la fonction RAPI) :

La détection des plaques d'immatriculations avec les systèmes RAPI se basent sur plusieurs méthodes et techniques, certaines sont simples et d'autres sont plus complexes (voir chapitre3). Pour notre système RAPI, nous avons utilisé la méthode de détection par contours car elle donne de bons résultats et elle utilise des méthodes de traitement d'image simples, mais elle a aussi ses défauts. L'image traitée doit être bien claire et contrastée (prétraité), les contours doivent être francs et les seuils doivent être bien choisis.

Nous avons développé en parallèle la méthode de détection par couleur. Cette méthode nous a donné de bons résultats sur les plaques d'immatriculations jaunes (figure III.3). Pour les plaques d'immatriculations blanches, nous avons rencontré un problème des reflets de lumière sur les véhicules qui apparaissent en blanc.

A propos de la reconnaissance optique de caractères, il existe plusieurs méthodes pour la réaliser. Nous avons opté pour l'utilisation d'une fonction OCR appelé Tesseract [24] qui permet d'extraire du texte à partir une image photographiée ou scannée (voir annexe 2).

Voici un organigramme qui présente la structure de la fonction RAPI (détection et lecture de la PI) :

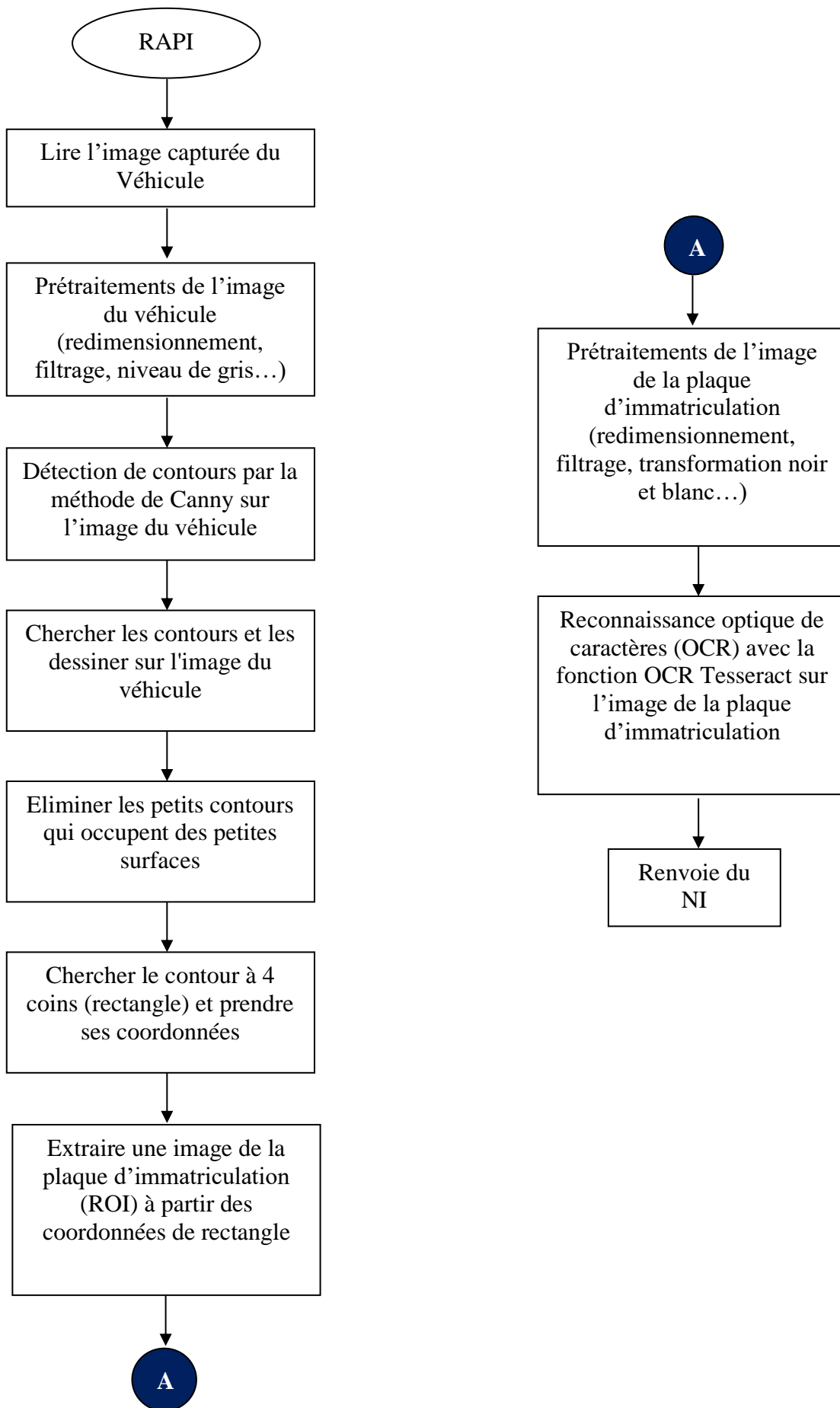


Figure IV-7 : Organigramme de la fonction RAPI

IV.4.3. Base de données et historique :

Dans notre projet on a utilisé les fichier CSV (Excel) pour faire la base des données et l'historique.

- Comment ça marche ?

Comme en vient de préciser précédemment, la fonction RAPI qui est charger de la détection et lecture de la plaque d'immatriculation va retourner un texte, ce texte représente le numéro d'immatriculation.

Ce numéro d'immatriculation (NI) retourner par la fonction RAPI, il va être comparer avec un ensemble de NI enregistré dans le fichier **ma.csv** (**m**atricules **a**utorisé).

	A	B
1	245011831 a1	
2	210011627 a2	
3	37411039 a3	
4	83264711525 a4	
5		
6		
7		

Figure IV-8 : Fichier *ma.csv* contenant les matricules des véhicules autorisés a entrées

Après que la vérification et la comparaison est faite, on peut tomber sur deux cas :

- Soit le NI existe dans la base de données, donc le véhicule est autorisé à entrer, dans ce cas l'enregistrement se fait dans le fichier **mes.csv** (**m**atricules **e**ntrées/**s**orties) et dans le fichier **xx.csv** (xx correspond au nom du fichier de chaque véhicule).
- Soit le NI n'existe pas dans la base de données, donc le véhicule n'est pas autorisé à entrer, dans ce cas l'enregistrement se fait dans le fichier **mes.csv**.

Les fichiers **mes.csv** et **xx.csv** représentes l'historique des entrées, des sorties et des tentatives d'entrées (véhicule non autorisés). Ces fichiers vont contenir le NI, la date, l'heure, le types d'événement (entrée ou sortie ou entrée manuel) et le type d'autorisation.

Chapitre 4

	A	B	C	D	E	F	G
208	entree manuel	2020:05:23	10:51:31				
209	entree manuel	2020:05:23	10:52:39				
210	entree	2020:05:23	11:32:46		non autorise		
211	sortie	2020:05:23	11:33:02		non autorise		
212	entree manuel	2020:05:23	11:33:20				
213	sortie	2020:05:23	11:40:58	245011831	autorise		
214	sortie	2020:05:23	11:41:08	245011831	autorise		
215	entree	2020:06:03	10:33:20	83264711525	autorise		
216	entree	2020:06:03	10:34:31	3264711525	non autorise		
217	entree	2020:06:03	10:35:58	83264711525	autorise		
218	sortie	2020:06:03	10:38:35	83264711525	autorise		
219	entree	2020:06:03	10:43:05	726501153	non autorise		
220	sortie	2020:06:03	10:45:07	72450115	non autorise		
221	entree manuel	2020:06:03	10:46:59	72450115			
222	entree	2020:06:03	10:47:38	450115	non autorise		
223	entree	2020:06:03	10:50:29	83264711525	autorise		
224	entree	2020:06:03	10:51:54	3264711525	non autorise		
225	entree	2020:06:03	10:52:58	83264711525	autorise		










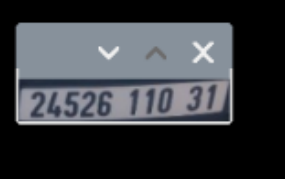
Figure IV-9 : Fichier mes.csv l'historique de tous les véhicules


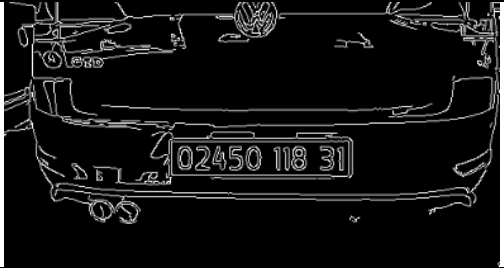
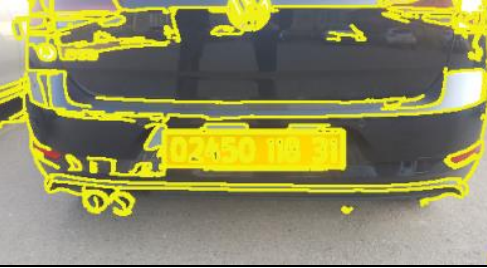

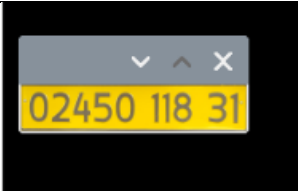
	A	B	C	D	E	F	G	H
1								
2	entree	2020:05:15	12:10:24					
3	sortie	2020:05:15	12:10:44					
4	entree	2020:05:15	12:16:36					
5	sortie	2020:05:15	12:16:56					
6	entree	2020:05:17	14:33:15					
7	sortie	2020:05:17	14:33:31					
8	entree	2020:05:17	14:35:44					
9	entree	2020:05:17	14:36:14					
10	entree	2020:05:17	14:36:53					
11	entree	2020:05:17	14:40:23					
12	sortie	2020:05:17	14:41:34					





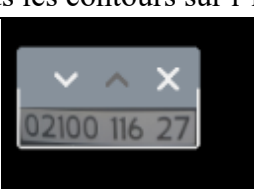
Figure IV-10 : Fichier xx.csv l'historique d'un des véhicules autorisés





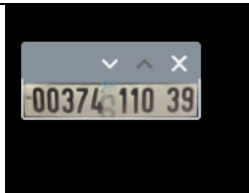
IV.5 Résultats obtenus du système RAPI :





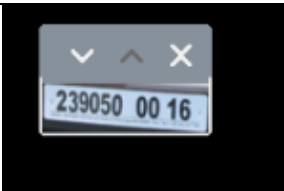
IV.5.1. Résultats corrects :

F i g u r e A		
	1-Image originale	2-Détection de contours-Canny
		
	3-Tous les contours sur l'image	4-Détection de la plaque
		<pre>Le numero d'immatriculation 13413 109 27] 13413 109 27 1341310927 <input type="checkbox"/></pre>
	5-Plaque extraite	6-Résultats
F i g u r e B		
	1-Image originale	2-Détection de contours-Canny
		
	3-Tous les contours sur l'image	4-Détection de la plaque
		<pre>Le numero d'immatriculation. 24526 110.31 24526 110 31 2452611031 <input type="checkbox"/></pre>
	5-Plaque extraite	6-Résultats







F i g u r e C		
	1-Image originale	2-Détection de contours-Canny
		
	3-Tous les contours sur l'image	4-Détection de la plaque
		<pre>Le numero d'immatriculation. 02450 118 31 02450 118 31 0245011831 □</pre>
	5-Plaque extraite	6-Résultats

F i g u r e D		
	1-Image originale	2-Détection de contours-Canny
		
	3-Tous les contours sur l'image	4-Détection de la plaque
		<pre>Le numero d'immatriculation. 02100 116 27 02100 116 27 0210011627 □</pre>
	5-Plaque extraite	6-Résultats

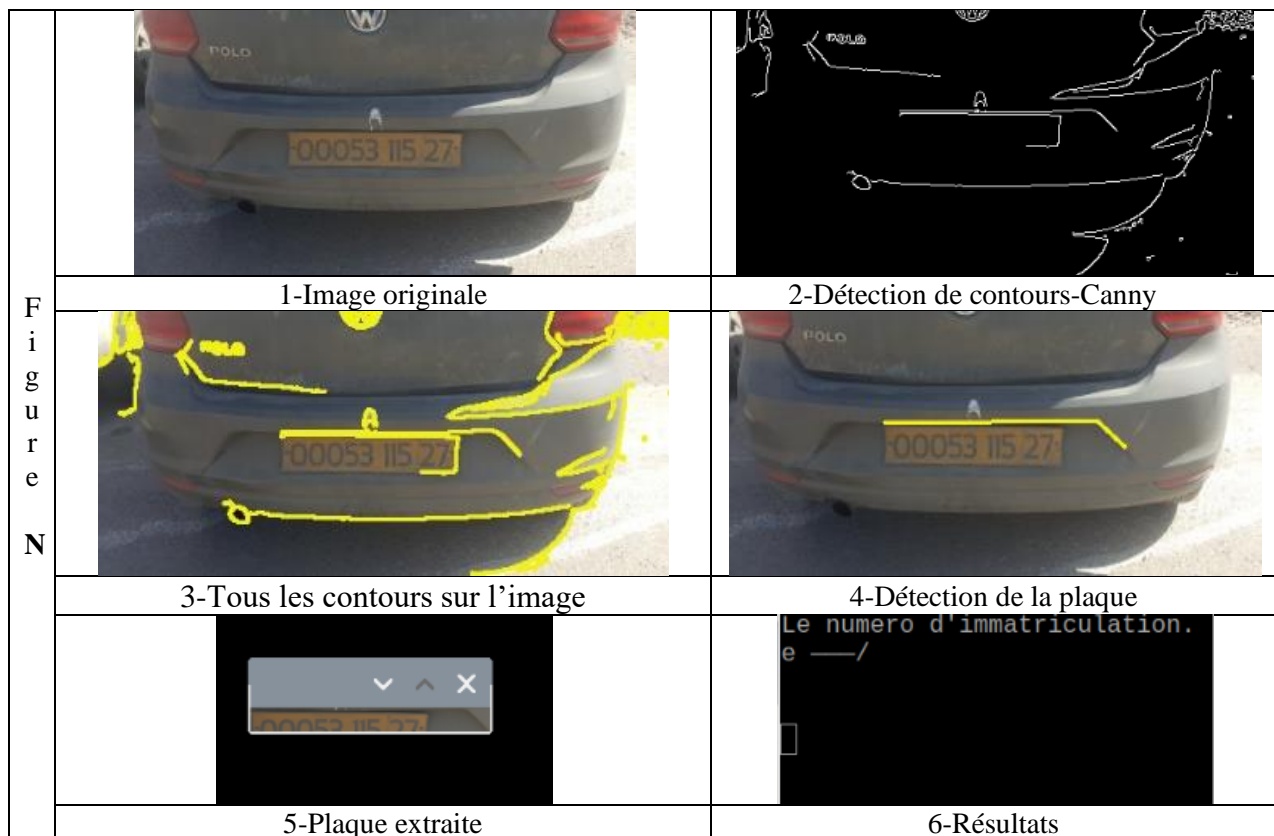
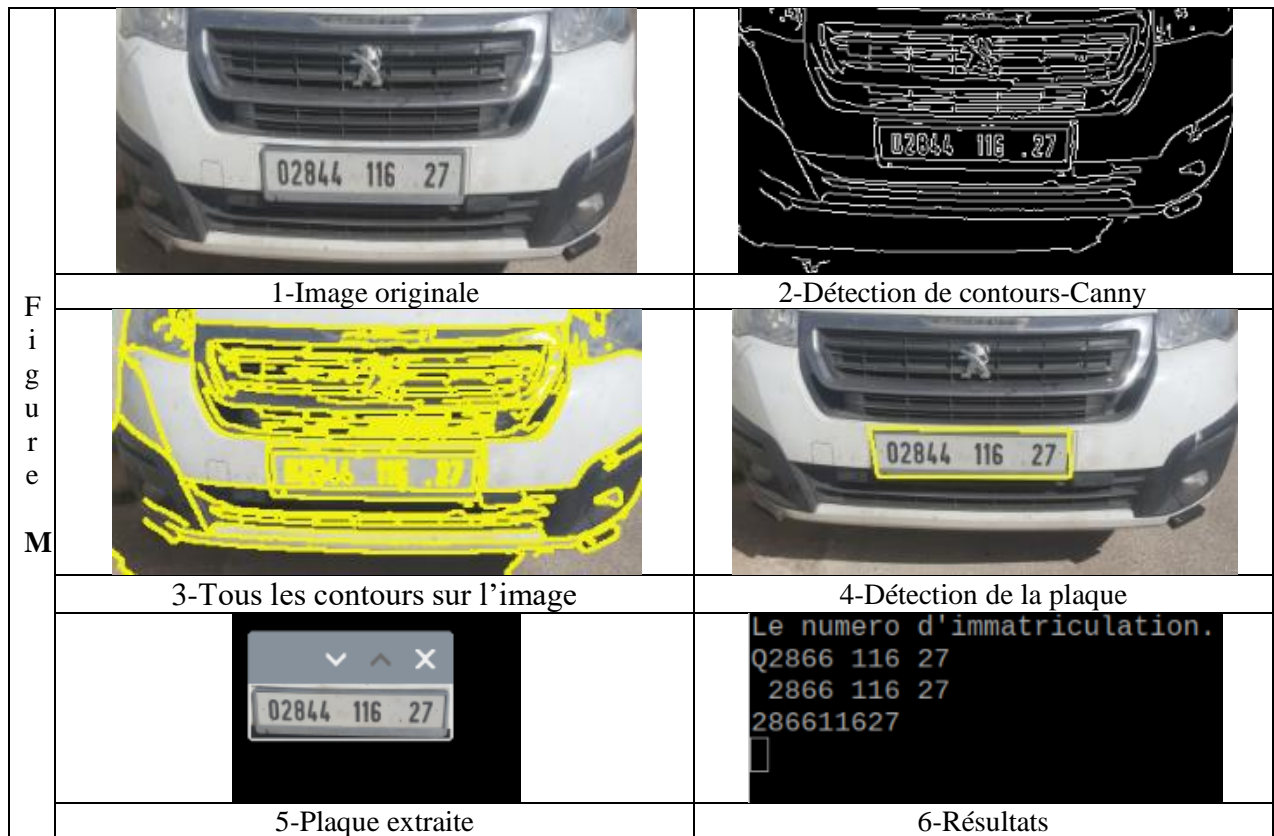
F i g u r e E		
	1-Image originale	2-Détection de contours-Canny
		
	3-Tous les contours sur l'image	4-Détection de la plaque
		<pre>Le numero d'immatriculation. 00374 110 39 00374 110 39 0037411039 □</pre>
	5-Plaque extraite	6-Résultats

F i g u r e F		
	1-Image originale	2-Détection de contours-Canny
		
	3-Tous les contours sur l'image	4-Détection de la plaque
		<pre>Le numero d'immatriculation. 239050 00 16 239050 00 16 2390500016 □</pre>
	5-Plaque extraite	6-Résultats

Chapitre 4

Figure	Image de plaque extrait (ROI)	Résultats
Figure G		Le numero d'immatriculation. 695195 00 16 695195 00 16 6951950016 □
Figure H		Le numero d'immatriculation. (01222 112 36 01222 112 36 0122211236 □
Figure I		Le numero d'immatriculation. 0582711331: 0582711331 0582711331 □
Figure J		Le numero d'immatriculation. 00513 111 39 00513 111 39 0051311139 □
Figure K		Le numero d'immatriculation. 832647 115.25 832647 115 25 83264711525 □
Figure L		Le numero d'immatriculation. 7245011531 7245011531 7245011531 □

IV.5.2. Résultats erronés :



IV.5.3. Analyse des résultats :

L'analyse des résultats obtenus est scindée en deux parties à savoir la détection de la plaque d'immatriculation et la reconnaissance des caractères de la plaque :

IV.5.3.1. La détection de la plaque :

La détection de la plaque dans l'image capturée a souvent été bien réalisée. Ceci peut être observé à travers les figures A à F et M.

Cependant il y a des cas où la détection ne se fait pas correctement (exemple le cas de la figure N).

On peut citer quelques causes qui font que la détection peut faillir :

- Les contours de la plaque ne sont pas suffisamment francs (l'intensité lumineuse des contours de plaques ne change pas fortement par rapport à celles des voisins).
- La surexposition ou la sous-exposition de lumière sur scène, c'est à dire un contraste insuffisant, ce qui produit des contours non fermés ou même absence de contours.

Par ailleurs :

- Les PI non rectangulaires ne sont pas détectées.
- Les PI occultées partiellement sont difficilement détectables.

IV.5.3.2. La reconnaissance optique de caractères :

La fonction OCR Tesseract nous a donné de bons résultats de reconnaissance de caractères (ici des chiffres) contenus dans l'image ROI c'est-à-dire l'image de la plaque.

Les figures A à L montrent cette bonne reconnaissance et ce, sous des conditions diverses :

- Couleurs : jaune, gris, blanche.
- Ligne de base légèrement inclinée, figures F et L.
- Fontes : diverses et même l'italique (figure G).
- Tache lumineuse (reflet), figure H.

Cependant il y a des cas où la reconnaissance ne se fait pas correctement. Ceci se voit dans les figures M et N. On peut citer quelques-unes des causes probables qui conduisent aux résultats erronés :

Chapitre 4

- La plaque d'immatriculation est inclinée.
- L'espace entre les caractères est très réduit.
- Le type de police non retrouvée dans la base de données de Tesseract.

IV.6 Les coûts :

Pour évaluer le travail sur ce projet, nous avons réalisé ce tableau qui contient les différents coûts :

Volume horaire de travail /Durée / Etude		Environ 600 heures/4 mois	
Compétences exploitées		Electronique, le système Raspberry, OS.Linux, Traitement d'Image, langage Python, OpenCV library.	
Programmation		Environ 500 lignes	
Coût financier	Prototype (sur maquette)	Raspberry Pi + câble d'alimentation + SD CARD + boîtier + ventilateur	21 000 DZD
		Cable HDMI	1000 DZD
		Clavier + souris	1300 DZD
		Caméra	1000 DZD
		Servo moteur	600 DZD
		Boutons poussoir + LEDs	100 DZD
		Totale	25 000 DZD
	Système réel (estimation)	Raspberry Pi + câble d'alimentation + SD CARD + boîtier	~ 21 000 DZD
		Caméras	~ 14000 DZD
		Capteurs boucle magnétique	~ 7000 DZD
		Cable HDMI	~ 4000 DZD
		Ecran	~ 18 000 DZD
		Totale	~ 64 000 DZD
		Barriere automatique (on option)	~ 50 000 DZD
Totale	~ 114 000 DZD		

IV.7 Conclusion :

Dans ce chapitre nous avons présenté la conception et l'implémentation de notre système RAPI, les résultats obtenus par le système, ainsi des futures améliorations sur le système.

Conclusion et perspectives

Conclusion :

Ce projet de fin d'étude avait pour ambition de réaliser un Système Embarqué de Reconnaissance Automatique de Plaque d'Immatriculation (RAPI) pour le contrôle d'accès aux parkings.

Il a fallu dans un premier temps se renseigner sur les méthodes et les techniques de traitement d'image pour la détection des plaques d'immatriculation, ainsi que la reconnaissance optique des caractères. Nous avons choisi la méthode de détection de contour de Canny car elle est très utilisée et donne de bons résultats.

Ensuite nous avons choisi un dispositif sur lequel le système doit être implémenté. Ce dispositif doit répondre à certains critères comme : la robustesse, la connectivité avec le monde réel (capteurs, caméras, actionneurs...). Donc nous avons utilisé le Raspberry PI 4 model B.

Puis nous avons étudié la conception et l'implémentation du système, ainsi l'emplacement de ce dernier dans le parking de notre faculté.

Le système réalisé dans ce PFE a montré de bonnes performances. Cependant il y a quelques points qui nécessitent des améliorations que nous allons présenter dans les perspectives.

Perspectives :

Dans le but d'améliorer notre système RAPI et le rendre plus fiable, nous pensons à :

- L'utilisation des techniques de prétraitements plus robustes pour avoir des résultats de traitement (détection de contours) plus propres.
- L'amélioration de la méthode de détection de la plaque d'immatriculation ainsi que la reconnaissance optique des caractères (l'intégration de la méthode de détection par couleurs, les réseaux neurones...).
- L'amélioration de la partie gestion des capteurs/Actionneurs.
- La détection de présence des véhicules par caméra (traitement vidéo).
- L'adaptation du système pour une utilisation durant la nuit (éclairage automatique et ajustable).

Annexe 1 : Boucle d'induction magnétique

Les boucles d'induction magnétiques sont utilisées pour détecter les véhicules arrivants ou passant par un certain point, par exemple un feu de circulation, et sur les autoroutes, à gérer le trafic. Une boucle conductrice isolée est installée dans la chaussée. Les composants électroniques transmettent l'information dans les câbles à des fréquences comprises entre 10 kHz et 200 kHz, selon les installations.[21]

Ce système se comporte comme un circuit oscillant dans lequel la boucle et le câble d'arrivée font office d'éléments inductifs. Lorsqu'un véhicule passe ou est à l'arrêt sur la boucle, le véhicule induit des courants de Foucault dans la boucle, ce qui fait baisser leur inductance. Cette diminution actionne le relais de sortie, ce qui envoie un signal au système de régulation du trafic, signifiant le passage ou la présence d'un véhicule.[21]



Boucle d'induction magnétique installée sur une route pour détecter les voitures.

Annexe 2 : Reconnaissance optique de caractères

La **reconnaissance optique de caractères** (en anglais *Optical Character Recognition* **OCR**), désigne les procédés informatiques pour la traduction d'images de textes imprimés ou dactylographiés en fichiers de texte. Un ordinateur réclame pour l'exécution de cette tâche un logiciel d'OCR. Celui-ci permet de récupérer le texte dans l'image d'un texte imprimé et de le sauvegarder dans un fichier pouvant être exploité dans un traitement de texte pour enrichissement, et stocké dans une base de données ou sur un autre support exploitable par un système informatique.[22]

Les étapes de traitement peuvent être schématisées ainsi :[22]

1. **Préanalyse** de l'image : le but est d'améliorer éventuellement la qualité de l'image. Ceci peut inclure le redressement d'images inclinées ou déformées, des corrections de contraste, le passage en mode bicolore (noir et blanc, ou plutôt papier et encre), la détection de contours.
2. **Segmentation** en lignes et en caractères (ou Analyse de page) : vise à isoler dans l'image les lignes de texte et les caractères à l'intérieur des lignes. Cette phase peut aussi détecter le texte souligné, les cadres, les images.
3. **Reconnaissance** proprement dite des caractères : après normalisation (échelle, inclinaison), une instance à reconnaître est comparée à une bibliothèque de formes connues, et on retient pour l'étape suivante la forme la plus « proche » (ou les N formes les plus proches), selon une distance ou une vraisemblance (*likelihood*). Les techniques de reconnaissance se classent en quelques grands types¹:
 - a. Classification par Caractéristiques (*Features*) : une forme à reconnaître est représentée par un vecteur de valeurs numériques - appelées *features* en anglais - calculées à partir de cette forme. Le nombre de *features* est de l'ordre de 100 à 300. Si les *features* sont bien choisies, une classe de caractères (par exemple l'ensemble des A majuscules) sera représentée par un « nuage » contigu de points dans l'espace vectoriel des *features*. Le rôle du classificateur est de déterminer à quel nuage (donc à quelle classe de caractères) la forme à reconnaître appartient le plus vraisemblablement. La classification fait généralement appel à divers types de réseaux de neurones artificiels entraînés sur de vastes bases de formes possibles.
 - b. Méthodes métriques : consistent à comparer directement la forme à reconnaître, au moyen d'algorithmes de distance, avec un ensemble de modèles appris. Ce type de méthode est peu utilisé et peu valorisé par les chercheurs, car souvent plus naïf et vraisemblablement moins efficace que les méthodes à base de *features*.
 - c. Méthodes statistiques : dans le domaine de la reconnaissance d'écriture manuscrite, il est fréquemment fait appel aux méthodes probabilistes/statistiques.
 - d. Génération du format de sortie, avec la mise en page pour les meilleurs systèmes.



Segmentation des caractères d'une plaque
d'immatriculation

Annexe 3 : Instructions du programme de la fonction RAPI

La fonction de détection et lecture de la plaque d'immatriculation (la fonction RAPI) est composée comme suit :

- **Etape de prétraitement :**

Redimensionnement :

```
cv2.resize(img, (nbr de lignes, nbr de
colones), interpolation =cv2.INTER AREA)
```

Conversion niveau de gris:

```
cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

Filtrage de bruit :

```
cv2.bilateralFilter(gray, Diametre, Sigma,
```

- **Etape de détection de contour**

Détection de contour par Canny :

```
cv2.Canny(filtred, Seuil 1, Seuil 2)
```

Recherche des contours :

```
cnts, new = cv2.findContours(edged.copy(),
cv2.RETR_LIST, cv2.CHAIN_APPROX_SIMPLE)
```

Eliminer les petits contours qui occupent des petites surfaces :

```
cnts = sorted(cnts, key = cv2.contourArea,
reverse = True)[:30]
```

Dessiner les contours sur l'image :

```
cv2.drawContours(img1, cnts, -
```

Chercher le contour à 4 coins et segmentation de la plaque d'immatriculation :

```
        if len(approx) == 4:
screenCnt = approx
x, y, w, h = cv2.boundingRect(c)
new_img=img[y:y+h, x:x+w]
        cv2.imwrite('./'+str(idx)+'.png', new_im
g)
break
```

- **Étape de prétraitements de l'image de la plaque d'immatriculation :**

Redimensionnement :

```
cv2.resize(ROI, (nbr de lignes, nbr de  
colonnes), interpolation=cv2.INTER_AREA)
```

Conversion niveau de gris :

```
cv2.cvtColor(ROI, cv2.COLOR_BGR2GRAY)
```

Filtrage de bruit :

```
cv2.bilateralFilter(ROI, Diamètre, Sigma, Sigma)
```

Seuillage par Otsu :

```
cv2.threshold(ROI_filtred, 0, 255,  
cv2.THRESH_BINARY+cv2.THRESH_OTSU)
```

- **Étape de reconnaissance optique de caractères**

```
text=pytesseract.image_to_string(ROI  
, Configuration du moteur OCR pour reconnaitre des  
chiffres)
```

Bibliographie

- [1]. « Il y a environ 1,2 milliard de voiture dans le monde ! », LESAVIEZVOUS.net, janvier 2019.
URL : <https://www.lesaviezvous.net/societe/il-y-a-environ-12-milliard-de-voitures-dans-le-monde.html>
- [2]. B.Nassim, « Algérie : Voici le nombre de véhicules en circulation (parc automobile) », Dzair Daily, janvier 2020. URL : <https://www.dzairdaily.com/algerie-nombre-voitures-circulation-parc-automobile-algerien-2020/>
- [3]. https://fr.wikipedia.org/wiki/Plaque_d%27immatriculation_alg%C3%A9rienne. Dernière modification le 09 Mars 2020.
- [4]. https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_reconnaissance_automatique_des_plaques_min%C3%A9ralogiques. Dernière modification le 11 Décembre 2019.
- [5]. A.Tauvy-N.Carayon-S.Soissons, « traitement des images numériques », école nationale supérieure des télécommunications, Paris.
URL : http://www.tsi.enst.fr/pages/enseignement/ressources/mti/egalhisto/rapport.htm#_Toc36969443
- [6]. <https://fr.wikipedia.org/wiki/Image>. Dernière modification le 27 Mai 2020.
- [7]. https://fr.wikipedia.org/wiki/Traitement_d%27images. Dernière modification le 23 février 2020.
- [8]. <http://www.map.toulouse.archi.fr/works/panoformation/imagenum/imagenum.htm>
- [9]. M.Lossendière, « caractéristiques d'une image numérique », 31 Aout 2016.
URL : <http://www.lossendiere.com/2016/08/31/caracteristiques-dune-image-numerique/>
- [10]. [https://fr.wikipedia.org/wiki/%C3%89chantillonnage_\(signal\)](https://fr.wikipedia.org/wiki/%C3%89chantillonnage_(signal)). Dernière modification le 04 février 2019.
- [11]. [https://fr.wikipedia.org/wiki/Quantification_\(signal\)](https://fr.wikipedia.org/wiki/Quantification_(signal)). Dernière modification le 1 Avril 2020.
- [12]. <http://fpt113-vg.espaceweb.usherbrooke.ca/dohtml/caracteristique-image.htm>
- [13]. « Image numérique », Cour en ligne, Université rennes2, France. URL : https://www.sites.univ-rennes2.fr/arts-spectacle/cian/image_numFlash/pdf/chap3_tout.pdf
- [14]. https://fr.qwe.wiki/wiki/Automatic_number-plate_recognition#Usage
- [15]. https://fr.wikipedia.org/wiki/Raspberry_Pi. Dernière modification le 20 Juin 2020.

Bibliographie

- [16]. BINAIRE, « Raspberry pi :la petite histoire d'une grande idée », Le monde.fr, 28 Décembre 2015.
URL : <https://www.lemonde.fr/blog/binaire/2015/12/28/raspberry-pi-la-petite-histoire-dune-grande-idee/>
- [17]. <https://www.dzduino.com/uk-raspberry-pi-4-modele-b-soc-bcm2711-ram-2go-ddr4-usb-3-0-poe?search=Raspberry%20pi&description=true>
- [18]. [https://fr.wikipedia.org/wiki/Python_\(langage\)](https://fr.wikipedia.org/wiki/Python_(langage)). Dernière modification le 08 Juin 2020.
- [19]. <https://fr.wikipedia.org/wiki/Linux>. Dernière modification le 21 Juin 2020.
- [20]. <https://en.wikipedia.org/wiki/OpenCV>. Dernière modification le 10 Juin 2020.
- [21]. https://fr.wikipedia.org/wiki/Boucle_d'induction_magn%C3%A9tique. Dernière modification le 10 novembre 2019.
- [22]. https://fr.wikipedia.org/wiki/Reconnaissance_optique_de_caract%C3%A8res. Dernière modification le 06 avril 2020.
- [23]. Ameer Chhayder et Imene Belhadj Mohamed, « Système de reconnaissance automatique des plaques minéralogiques », IPEIS, Tunisie, Mars 2009.
URL : http://www.setit.rnu.tn/last_edition/setit2009/Image%20and%20Video/18.pdf
- [24]. [https://en.wikipedia.org/wiki/Tesseract_\(software\)](https://en.wikipedia.org/wiki/Tesseract_(software)). Dernière modification le 23 juin 2020.
- [25]. « Conversion d'un signal analogique en un signal numérique », MAXICOURS.com. URL : <https://www.maxicours.com/se/cours/conversion-d-un-signal-analogique-en-signal-numerique/>