REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE YAHIA FARES DE MEDEA



Faculté de **T**echnologie

Mémoire de fin d'études de Master

Filière : Télécommunication
Spécialité : Systèmes des télécommunications

Contrôle d'accès des véhicules par lecture automatique des plaques d'immatriculation

---

Présenté par :

Mlle : BOUZEMLAL Nawel

Mlle : SADOUKI Rima

Proposé et Dirigé par :

Mr. REBAI Mohamed.

Promotion 2019



Dédicace :

JE dédie ce modeste travail
à ma chère maman,
pour son soutient moral et financier
et d'être la lumière de ma vie,
qui m'a
toujours encouragé.

A mes tentes et mes oncles.
Mes cousines et mes cousins.
A tous nos chers amis(e) : ma binôme ***RIMA, FELLA, KAMI, NESRINE, les*** *trois* ***MANEL, SOUMEYA, NESRINE, NADIA, SIHAM, MOOH ROSSI*** *et mes chères frères* ***TAREK*** *et* ***ZINOU.***

Nawel bouzemlal

Dédicace :

Je dédie ce travail

A ma maman

Qui m'a soutenu et encouragé durant toutes ces années d'études.

Qu'elle trouve le témoignage de ma profonde reconnaissance

A toi mon père

Tu as toujours été a mes cotés pour me soutenir et m'encourager.

Que ce travail traduit ma gratitude et mon affection

A mes braves frères ***NAZIM*** *et* ***Anis*** *qui ont été toujours présent pour moi*

A mon grand père

A ma grand-mère

A mes oncles et mes tantes

A tous ceux qui m'ont aidé à faire ce travail

A tous mes chers amis : ma binôme ***NAWEL****, ma chérie* ***SOUMEYA****,* ***MANEL****,* ***AMEL****,*

NADIA*,* ***NESRINE****,* ***WIDED****, mes chers frères :* ***TAREK*** *et* ***ZINOU****.*

A tous ceux que j'aime.

Rima sadouki.

Remerciements :

Au tout début nous tenons à remercier le bon dieu pour le courage et la volonté et la force qui nous a donné,

L'espoir et la santé qu'il nous a octroyé afin que soit réalisée ce Modeste travail.

On remercie notre encadreur ***Mr .REBIAI***

Pour son soutien moral et pédagogique pour son Orientation bienveillante.

Nous tenons à remercier

tous les enseignants qui nous ont suivis durant notre formation et pour avoir accepté d'examiner notre travail

en particulier ***Mr.BAILLICH****.*

nos remerciement vont également à tous les Membres de jury, qui font l"honneur de juger notre travail.

Grand Merci À TOUS...

ملخص :

في الوقت الحاضر ، يوجد العديد من أنظمة التعرف على لوحات الترخيص ، فهذه الأنظمة لها غرض للتعرف على رقم السيارات. تعتمد هذه التقنية على معالجة الصور المسؤولة بشكل أساسي عن تحديد مكان اللوحة ثم تجزئتها للتعرف على الأرقام .هذا مجال مهم من مجالات البحث يساعد أيضًا في الحصول على النتيجة الصحيحة مقارنة بالنتيجة اليدوية.

الهدف الرئيسي من مذكرتنا هو تصميم تطبيق يسمح للنظام بالتعرف على لوحة ترخيص السيارة عند مدخل السيارات.

يعتمد النظام على كمبيوتر شخصي وبرنامج MATLAB وكاميرا رقمية تساعد على التقاط صور للمركبات. يتعرف البرنامج على رقم اللوحة ، ويقارن رقم اللوحة مع قاعدة بيانات مدمجة ، ويقرر ما إذا كان يسمح لسيارة بدخول المنطقة المحددة أم لا.

الكلمات المفتاحية: التعرف على لوحات الترخيص ,معالجة الصور,مواقف السيارات ,برنامج MATLAB ,قاعدة البيانات.

Résumé :

De nos jours il existe de nombreux systèmes de reconnaissance des plaques d'immatriculation, ces systèmes ont un but de reconnaitre le numéro de chaque véhicule. Cette technologie est basée sur le traitement d'image qui est responsable principalement à localiser la plaque par suite de faire la segmentation afin de reconnaitre les chiffres. C'est un domaine de recherche important qui aide également à obtenir le résultat correct par rapport à un résultat manuel.

L'objectif principal de notre mémoire est de réaliser une application permettant au système de reconnaître la plaque d'immatriculation à l'accès des voitures.

Le système sera basé sur un ordinateur personnel et le logiciel MATLAB et un appareil photo numérique qui aide à capturer des images de véhicules. Le logiciel reconnaît le numéro de plaque, compare le numéro de plaque à une base de données intégrée et décide si un véhicule est autorisé ou non à rentrer dans la zone désignée.

Les mots clés : reconnaissance des plaques d'immatriculation, le traitement d'image, parking, logiciel MATLAB, base de données.

Abstract:

Nowadays there are many license plate recognition systems; these systems have a purpose to recognize the number of each vehicle. This technology is based on image processing which is mainly responsible for locating the plate followed by a segmentation to recognize the numbers. This is an important area of research that also helps to get the correct result compared to a manual result.

The main objective of our memory is to make an application allowing the system to recognize the license plate of the vehicle at the entrance of the access of the car.

The system will be based on a personal computer and MATLAB software and a digital camera that helps capture images of vehicles. The software recognizes the plate number, compares the plate number with a built-in database, and decides whether or not a vehicle is allowed to enter the designated area.

Keywords: recognition of license plates, image processing, parking, MATLAB software, database.

Table des matières :

Liste des figures : I

Liste des tableaux : III

Liste des acrononymes : IV

Introduction générale : 1

Chapitre I : Généralités sur la reconnaissance de formes

I.1 Introduction : 2

I.2 La reconnaissance de formes : 2

I.3 Traitement d'images : 3

I.3.1 Définition : 3

I.3.2 Définition d'Image : 3

I.3.3 Image numérique : 3

I.3.3.1 Types d'images : 3

1. Image binaire : 3
2. Image en niveaux de gris : 3
3. Image couleur (ou RGB) : 4

I.3.3.2 Caractéristiques d'une image numérique : 4

1. Pixel : 4
2. Dimension : 5
3. Résolution : 5
4. Bruit : 5
5. Histogramme : 5
6. Luminance : 6
7. Contraste : 6

I.3.4 Filtrage : 6

I.3.4.1 Filtrage par la moyenne: 6

I.3.4.2 Filtre médian : 6

I.3.4.3 Le filtre de Canny : 7
I.3.5 Morphologies mathématique: 8
I.3.5.1 Erosion : 9
I.3.5.2 Dilatation : 9
I.3.5.3 L'ouverture : 10
I.3.5.4 La fermeture : 10
I.3.6 Segmentation d'images : 10
I.3.6.1 Différentes approches : 10
1. Approche contours : 11
2. Approche régions : 11
3. Approche par classification de pixels : 11
I.4 La reconnaissance automatique de plaque d'immatriculation : 11
I.4.1 Définition : 11
I.4.2 Caractéristiques des plaques algériennes : 12
I.4.3 Difficultés : 13
I.4.4 Etapes de reconnaissance de plaques d'immatriculations : 14
I.4.4.1 Image de véhicule : 14
I.4.4.2 Extraction de la plaque d'immatriculation (localisation de la plaque) : 14
I.4.4.3 La segmentation : 14
I.4.4.4 La reconnaissance des caractères : 14
I.4.5 Les différents systèmes existants : 15
I.4.6 Applications de système de reconnaissance automatique de formes : 15
I.5 Conclusion : 16
Chapitre II: Implémentation d'une application de reconnaissance de chiffres par MATLAB
II.1 Introduction : 17
II.2 Partie 1 : conception d'un système de reconnaissance de plaques d'immatriculations basée sur MATLAB : 17

II.2.1 Le fonctionnement du système de plaque d'immatriculation du véhicule : ... 17
II.2.2 Contraintes : ... 19
II.2.3 Outil de développement « MATLAB » : ... 20
II.2.3.1 Introduction sur matlab : ... 20
II.2.3.2 Les avantages de Matlab : ... 20
II.2.4 Schéma du système de reconnaissance de plaque d'immatriculation : ... 21
II.2.4.1 Image originale de véhicule : ... 21
II.2.4.2 Localisation de la plaque : ... 22
1. Convertir l'image en niveau de gris : ... 22
2. Modifier l'étalement de niveau de gris : ... 23
3. Filtre de « Canny » : ... 23
4. Dilatation morphologique : ... 24
5. Remplir les trous : ... 24
6. Suppression des petits objets : ... 25
7. Filtre médian : ... 25
8. Afficher la plaque seule : ... 26
II.2.4.3 Segmentation de la plaque : ... 26
II.2.4.4 La reconnaissance des chiffres et l'Affichage du numéro de la plaque : ... 26
II.3 Partie 2 : conception d'une application de reconnaissance de plaques d'immatriculations basé sur MATAB-GUI: ... 27
II.4 Conclusion : ... 31

Chapitre III: Réalisation d'un système d'accès des véhicules basé sur l'Arduino

III.1 Introduction : ... 32
III.2 Généralité sur l'Arduino : ... 32
III.2.1 Qu'est-ce que l'Arduino ? ... 32
III.2.2 But de la carte Arduino : ... 32
III.2.3 Les applications possibles : ... 32

III.2.4 Les outils Arduino : 33
III.2.4.1 Partie matérielle : 33
1. Le microcontrôleur 33
2. L'alimentation 33
3. La connectique 33
4. La platine d'expérimentation : 34
III.2.4.2 Partie logiciel : 35
1. L'environnement de la programmation : 35
2. Structure générale du programme (IDE Arduino) 36
III.3 La carte Arduino «UNO » : 36
III.3.1 Caractéristiques de la carte Arduino uno : 37
III.4 Les Accessoires de la carte Arduino : 38
III.5 Réalisation du projet "reconnaissance de plaque d'immatriculation" avec une carte Arduino UNO : 39
III.5.1 Présentation du projet : 39
III.5.2 Matériels utilisés 39
III.5.3 Algorithme du projet : 41
III.5.4 Les schémas électroniques : 42
III.5.4.1 Circuit intégré Arduino_ caméra : 43
III.5.4.2 Circuit intégré Arduino-infrarouge_ la LED : 44
III.5.4.3 Circuit intégré Arduino_moteur_bouton poussoir : 45
III.5.4.4 Réalisation du montage final: 46
III.6 Résultats et discussion : 47
III.7 Conclusion : 49
Conclusion générale : 50
Bibliographie 51

Liste des figures :

Chapitre1: Généralités sur la reconnaissance de formes

Figure I. 1 : *organigramme général de la reconnaissance de formes* 2

Figure I. 2:*image numérique* 3

Figure I. 3: *les types d'images* 4

Figure I. 4:*représentation des pixels et les dimensions d'une image* 5

Figure I. 5:*l'histogramme d'une image* 5

Figure I. 6:*Filtrage par la moyenne* 6

Figure I. 7:*Filtre médian* 7

Figure I. 8:*quelques types de filtre* 7

Figure I. *9:image filtrée par l'opérateur CANNY* 8

Figure I. 10:*exemple d'un élément structurant* 9

Figure I. 11:*érosion de X par B* 9

Figure I. 12:*dilatation de X par B* 10

Figure I. 13: *Ouverture par un carré 3×3* 10

Figure I. 14:*Fermeture par un carré 3×3* 11

Figure I. 15:*image d'un véhicule à reconnaitre à l'entré d'un parking* 13

Figure I. 16:*exemple d'une plaque d'immatriculation algérienne* 13

Figure I. 17:*les difficultés* 14

Chapitre2: Implémentation d'une application de reconnaissance de chiffres par MATLAB

Figure II. 1:*l'arrêt de véhicule* 17

Figure II. 2:*la détection de la présence de véhicule par infrarouge* 18

Figure II. 3:*l'ouverture de la barrière* 18

Figure II. 4:*l'organigramme de parking* 19

Figure II. 5:*Schéma général du système de reconnaissance de plaque d'immatriculation* 21

Figure II. 6:*image originale de véhicule sur MATLAB* 22

Figure II. 7:*image en niveau de gris sur MATLAB* 22

Figure II. 8:*modification de l'étalement de niveau de gris sur MATLAB* 23

Figure II. 9:*filtre de CANNY sur MATLAB* 23

Figure II. 10:*dilatation sur MATLAB* 24

Figure II. 11:*Remplir les trous sur MATLAB* 24

Figure II. 12:*enlevé les petits objets sur MATLAB* 25

Figure II. 13:*filtre médian sur MATLAB* 25
Figure II. 14 : *affichage de la plaque à reconnaitre sur MATLAB* 26
Figure II. 15:*segmentation de la plaque sur MATLAB* 26
Figure II. 16:*l'affichage de matricule sur MATLAB* 27
Figure II. 17:*la base de données* 27
Figure II. 18:*le bouton 1 qui affiche l'image sur l'application MATLAB* 28
Figure II. 19:*le bouton 2 qui convertit l'image en niveau de gris sur l'application MATLAB* 28
Figure II. 20:*le bouton 3 qui localise la plaque sur l'application MATLAB* 29
Figure II. 21:*le bouton 4 qui affiche la plaque à reconnaitre sur l'application MATLAB* 29
Figure II. 22:*le bouton 5 qui affiche les chiffres de la plaque sur l'application MATLAB* 30
Figure II. 23:*le bouton 6 qui compare les matricule sur l'application MATLAB* 30
Figure II. 24:*le bouton 7 qui permet de sortir de l'application MATLAB* 31

Chapitre3: Réalisation d'un système d'accès de véhicules basé sur l'Arduino

Figure III. 1: *plateforme Arduino* 33
Figure III. 2:*la connectique de la carte Arduino* 34
Figure III. 3: *des jumpers* 34
Figure III. 4:*la platine d'expérimentation* 35
Figure III. 5:*les connecteurs dans une platine* 35
Figure III. 6:*interface Arduino IDE* 36
Figure III. 7:*Arduino UNO* 37
Figure III. 8:*les modèles d'Arduino Uno* 37
Figure III. 9:*Exemple de Shield* 38
Figure III. 10: *le message pour ouvrir la barrière* 39
Figure III. 11:*le message pour garder la barrière fermée* 39
Figure III. 12: *organigramme de parking par Arduino* 42
Figure III. 13:*Circuit intégré Arduino_ caméra par fritzing* 43
Figure III. 14:*Circuit intégré Arduino-infrarouge_ la LED par fritzing* 44
Figure III. 15:*Circuit intégré Arduino_moteur_bouton poussoir par fritzing* 45
Figure III. 16:*le montage final par fritzing* 46
Figure III. 17:*le montage final de face* 47
Figure III. 18:*le montage final en haut* 47
Figure III. 19:*difficulté concernant la localisation de la plaque* 49

Liste des tableaux :

Chapitre 3 : Réalisation d'un système d'accès de véhicules basé sur l'Arduino:

Tableau III. 1: *caractéristique techniques de la carte Arduino UNO.* 37

Tableau III. 2: *matériels essentiel utilisés.* 39

Tableau III. 3:*la liste des matériels utilisés.* 40

Tableau III. 4: *le branchement du camera_Arduino* 43

Tableau III. 5: *le branchement du LED_PIR_Arduino*.....44

Tableau III. 6: *le branchement du servo_bouton_Arduino*.....45

LISTE D'ABRÉVIATIONS

AGL : Application de **G**estion **L**API.

ALPR : Automatic **L**icense **P**late **R**ecognition.

AVT: Asia **V**ision **T**echnology **L**imited.

C.I.E : Commission **I**nternationale de l'**É**clairage.

LAPI: Lecture **A**utomatique de **P**laques d'**I**mmatriculation.

LoG: Laplacian of **G**aussian.

RGB: Red **G**reen **B**lue.

RVB: **R**ouge **V**ert **B**leu.

Introduction générale :

Introduction générale :

L'écriture et la parole sont toujours au cœur des communications entre les êtres humains. Par Contre, la communication entre l'homme et la machine est réalisée à l'aide de boutons, de microphones ou de caméras.

Dans le monde d'aujourd'hui, avec le nombre croissant de véhicules jour par jour, il n'est plus possible de conserver un enregistrement de l'ensemble du véhicule. Il devient difficile de suivre et soumettre chaque voiture à la loi d'application et gestion du trafic. Pour cela il était nécessaire de développer des systèmes efficaces et fiables : la reconnaissance automatique des plaques d'immatriculations est une technologie qui trouve son essence dans ces 20 dernières années dans le développement des techniques de traitement d'image, elle apparait comme un sujet de recherche très vaste. Par conséquent, cette technologie est adoptée dans la gestion du système de stationnement de véhicule, l'accès des véhicules à une zone réglementée et le système de surveillance du trafic. L'objectif de La reconnaissance automatique de chiffres est de convertir des images qui sont compréhensibles par l'homme, en un code interprétable par un ordinateur. Ce mémoire possède 3 chapitres :

Dans le 1er chapitre, on va présenter des généralités sur la reconnaissance de formes, sa définition, ses caractéristiques ainsi que les difficultés auxquelles on peut faire face, ainsi une brève étude sur le traitement d'image.

Dans le 2éme chapitre, on va concentrer sur les étapes de reconnaissance de chiffres à l'aide du logiciel «Matlab » comme une première partie ensuite la deuxième partie, on va créer l'interface Matlab « GUI » pour l'implémenter.

Dans le 3eme chapitre, on va présenter une approche générale sur l'arduino dans la première partie, et dans la 2eme partie, on va réaliser un projet basé sur l'arduino qui permet de reconnaitre les matricules. Ces derniers vont être comparer avec une base de données des matricules afin d'ouvrir et de fermer la barrière.

Comme chaque véhicule porte une plaque d'immatriculation unique, le matricule représente un moyen efficace pour identifier les véhicules. Il s'agit d'une information unique pour chaque voiture. Le but principal de ce mémoire est d'étudier et de créer une application permettant au système de reconnaître la plaque d'immatriculation du véhicule à l'entrée du parking.

Le mémoire se termine par une conclusion générale.

Chapitre I:

Généralités sur la reconnaissance de formes

I.1 Introduction :

La reconnaissance des plaques d'immatriculation (Licence Plate Recognition) des véhicules est une partie critique du transport intelligent moderne, depuis que la plaque d'immatriculation est devenue un moyen simple et efficace pour identifier les véhicules. C'est un domaine très vaste qui est basé sur le traitement d'images, il a beaucoup évolué avec le temps, et devient de plus en plus une nécessité dans la gestion de plusieurs domaines comme la circulation et la sécurité routière, la gestion des parkings.etc.

Nous commençons avec ce premier chapitre qui présente des généralités sur la reconnaissance de formes, sa définition, ses caractéristiques et les difficultés auxquelles on peut faire face, ainsi qu'une brève étude sur le traitement d'image.

I.2 La reconnaissance de formes :

La reconnaissance de formes implique une phase d'apprentissage, mais Avant de pouvoir reconnaître une forme, il faut d'abord en connaître un modèle ou un ensemble de caractéristiques, afin de pouvoir faire une comparaison. L'ordinateur doit donc mémoriser ces données, qui devront être comparées avec celles des objets rencontrés, le schéma suivant donne explication générale : [1]

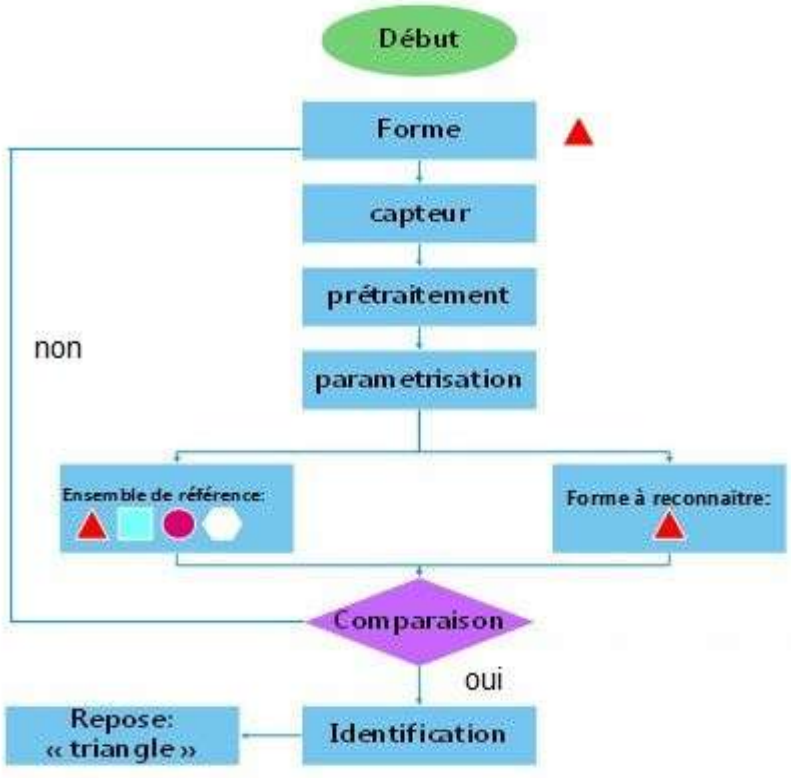


Figure I. 1 : *organigramme général de la reconnaissance de formes*

I.3 Traitement d'images :

I.3.1 Définition :

Le traitement d'images est une discipline de l'informatique et des mathématiques appliquées qui étudie les images numériques et leurs transformations, dans le but d'améliorer leur qualité ou d'en extraire de l'information. [2]

I.3.2 Définition d'Image :

Une image est avant tout un signal 2D (x, y), qui représente souvent une réalité 3D (x,y,z). D'un point de vue mathématique, une image est une matrice de nombres représentant un signal, plusieurs outils permettent de manipuler ce signal. [3]

I.3.3 Image numérique :

Une image numérique est constituée d'un ensemble de points appelés pixels (abréviation de Picture Élément). Les pixels sont approximativement rectangulaires, parfois carrés. Leur dimension peut être changée en réglant l'écran ou la carte graphique. [4]

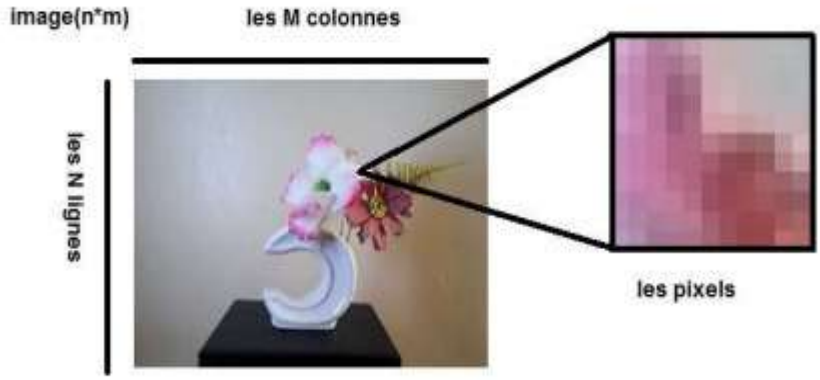


Figure I. 2:*image numérique*

I.3.3.1 Types d'images :

1. **Image binaire :**

Chaque pixel est soit blanc soit noir. Puisqu'il y a uniquement deux valeurs pour chaque pixel, un seul bit est utilisé pour le coder. [5]

2. **Image en niveaux de gris :**

Chaque pixel est un niveau de gris, allant de 0 (noir) à 255 (blanc). Cet intervalle de valeurs signifie que chaque pixel est codé sur huit bits (un octet). 256 niveaux de gris sont généralement suffisants pour la reconnaissance de la plus part des objets d'une scène. [1]

On peut convertir une image RVB en niveaux de gris selon plusieurs méthodes la plus simple est de faire $gris = \frac{bleu+vert+rouge}{3}$ c'est équivalent d'affecter la couleur grise à chacune des trois composantes RVB. [5]

La C.I.E (Commission Internationale de l'Éclairage) propose, de caractériser l'information de luminance (la valeur de gris) d'un pixel par deux formules :

Dans sa recommandation 709, qui concerne les couleurs « vraies » ou naturelles : Gris = 0.2125 Rouge + 0.7154 Vert + 0.0721 Bleu

Dans sa recommandation 601 pour les couleurs non-linéaires, c'est-à-dire avec correction du gamma (image vue à partir d'un écran vidéo) Gris = 0.299 Rouge + 0.587 Vert + 0.114 Bleu. [44]

3. Image couleur (ou RGB) :

Chaque pixel possède une couleur décrite par la quantité de rouge (R), vert (G) et bleu (B). Chacune de ces trois composantes est codée sur l'intervalle [0, 255], ce qui donne $255^3 = 16\ 777\ 216$ couleurs possibles. Il faut donc 24 bits pour coder un pixel. [6].le rouge de [0 à 255], le vert de [0 à 255], le Bleu de [0 à 255].

C'est codage de la couleur qui est utilisé par la plupart des écrans d'ordinateurs actuellement. On constate qu'il est très gourmand en mémoire. Pour faciliter le stockage des images en mémoire. [1]



Figure I. 3: ***les types d'images***

I.3.3.2 Caractéristiques d'une image numérique :

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants : [7]

1. Pixel :

Le pixel est la contraction de l'expression anglaise " picture elements". Le pixel, étant le plus petit point de l'image, C'est une entité calculable qui peut recevoir une structure et une quantification. Dans une image couleur (R.V.B.), un pixel peut être représenté sur trois octets : un octet pour chacune des couleurs : rouge (R), vert (V) et bleu (B). [4]

2. Dimension :

C'est la taille de l'image.

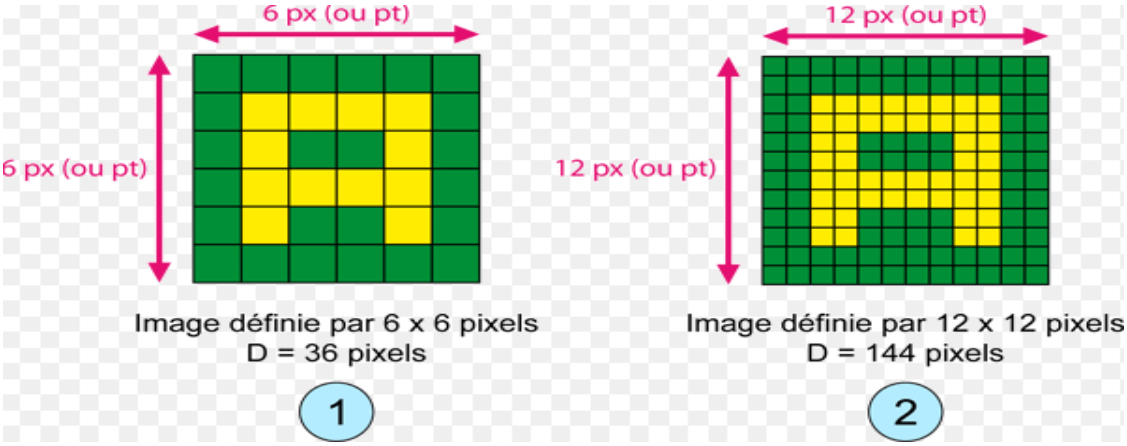


Figure I. 4:*représentation des pixels et les dimensions d'une image*

3. Résolution :

C'est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images.

4. Bruit :

Un bruit (parasite) dans une image est considéré comme un phénomène de brusque variation de l'intensité d'un pixel par rapport à ses voisins, il provient de l'éclairage des dispositifs optiques et électroniques du capteur. [7]

5. Histogramme :

Un histogramme est un graphique statistique permettant de représenter la distribution des intensités des pixels d'une image, c'est-à-dire le nombre de pixels pour chaque intensité lumineuse. Par convention un histogramme représente le niveau d'intensité en abscisse en allant du plus foncé (à gauche) au plus clair (à droite). [8]



Figure I. 5:*l'histogramme d'une image*

6. Luminance :

C'est le degré de luminosité des points de l'image.

7. Contraste :

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. [7]

I.3.4 Filtrage :

Les filtrages visent à modifier le contenu d'un pixel en prenant en compte une information locale, c'est-à-dire une information extraite du voisinage plus ou moins étendu du pixel. D'une façon générale, le filtrage est obtenu par convolution de l'image avec un noyau défini. Ce noyau peut-être interprété comme une petite image ou vignette contenant un gabarit de transformation (linéaire ou non linéaire) et que l'on applique sur chacun des pixels de l'image à filtrer pour créer une nouvelle image. [9]

I.3.4.1 Filtrage par la moyenne:

Le niveau de gris du pixel central est remplacé par la moyenne des niveaux de gris des pixels environnants. La taille de la fenêtre dépend de l'intensité du bruit et de la taille des détails significatifs de l'image traitée.

Le filtre par moyenne permet de :

- Permet de lisser l'image (smoothing).
- Remplace chaque pixel par la valeur moyenne de ses voisins.
- Réduit le bruit.
- Réduit les détails non-important.
- Brouille ou rend floue l'image (blur edges).

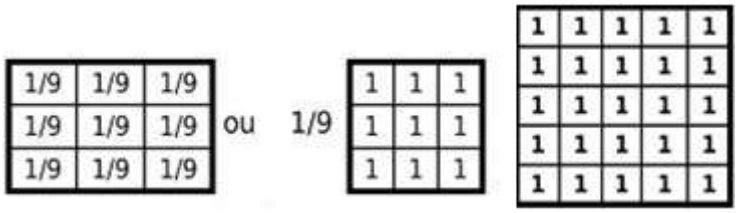


Figure I. 6:*Filtrage par la moyenne*

I.3.4.2 Filtre médian :

Le niveau de gris du pixel central est remplacé par la valeur médiane de tous les pixels de la fenêtre d'analyse centrée sur le pixel. La taille de la fenêtre dépend de la fréquence du bruit et de la taille des détails significatifs de l'image traitée. [10]

Etape 1 : Les niveaux de gris sont classés par Ordre croissant.

Etape 2 : La valeur médiane est choisie. [7]

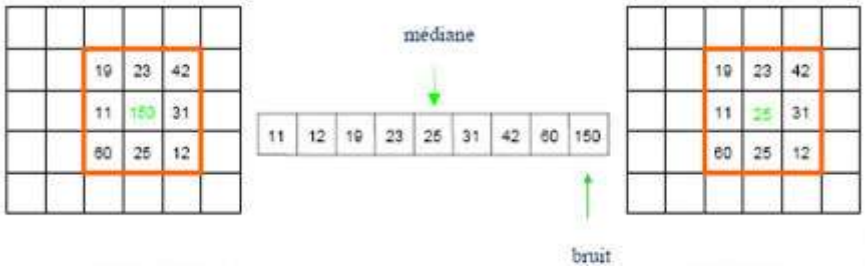


Figure I. 7:*Filtre médian*

Il existe d'autres filtres comme :

- ❖ Détection de Canny Edge.
- ❖ Détection de bord Sobel.
- ❖ Détection de bord de Prewitt.
- ❖ Détection de bord de journal. [11]



Figure I. 8:*quelques types de filtre*

Pour notre système on va utiliser le détecteur de bord CANNY.

I.3.4.3 Le filtre de Canny :

L'algorithme Canny est très efficace et offre une détection avancée des bords, détecte les bords importants.

Cette méthode repose sur deux critères:

- ❖ Le critère de localisation: il est dit que la distance entre la position réelle et le bord localisé devrait être minimisé.
- ❖ Le critère d'une réponse intégrant les différentes réponses à un bord unique.

La méthode peut être résumée comme suit:

- ❖ Un filtre gaussien, g (m, n), est lissé avec un écart-type spécifique, σ, afin de réduire le bruit.

$$g(m,n) = e^{\frac{m^2+n^2}{2\sigma^2}} \quad \text{(I.1)}$$

- Le dégradé local et la direction des bords sont calculés à chaque point de l'image.

$$g(m,n) = \sqrt{G_x^2(m,n) + G_y^2(m,n)} \quad \text{(I.2)}$$

$$\alpha(m,n) = \tan^{-1}\left(\frac{G_x(m,n)}{G_y(m,n)}\right) \quad \text{(I.3)}$$

- Un point de bord est défini comme un point dont la force est maximale locale dans la direction du dégradé.
- Les points de bord déterminés dans l'étape précédente donnent lieu à des crêtes dans l'image de la magnitude du gradient. L'algorithme suit ensuite le long des sommets de ces crêtes et met à zéro tous les pixels qui ne sont pas réellement sur le sommet de la crête afin de donner une ligne mince dans la sortie, un processus connu sous le nom non maximal suppression. Les pixels de crête sont alors seuil à l'aide de deux seuils, T1 et T2, avec T1 <T2. Les pixels de crête avec des valeurs supérieures à T2 sont dits «forts». Les pixels de crête avec des valeurs comprises entre T1 et T2 sont dits des pixels de bord «faibles».
- Enfin, l'algorithme effectue la liaison de bord en incorporant le facteur faible. 8 pixels connectés aux pixels forts. [12]



Figure I. 9*:image filtrée par l'opérateur CANNY*

I.3.5 Morphologies mathématique:

Brièvement, il s'agit de l'étude du contenu de l'image à partir d'un élément structurant, sachant qu'un élément structurant est un ensemble de points (de petite taille) muni d'une origine. Il permet de définir un voisinage pour chacun des pixels de l'image d'origine.

0	1	0
1	1	1
0	1	0

Figure I. 10:*exemple d'un élément structurant*

Les opérations simples de morphologie mathématique consistent à parcourir l'image et à considérer l'élément structurant B comme un masque binaire. [13]

Il existe 4 types de morphologie mathématique :

I.3.5.1 Erosion :

Cette opération est très importante avec la dilatation. Son principe est le suivant : soit un ensemble B, repéré par son centre p, B appartient à l'ensemble X et représente une forme quelconque. B est déplacé de telle sorte que son centre p occupe toutes les positions de l'espace. Est ce que l'ensemble B est entièrement inclus dans X pour chaque position de p ? Donc l'ensemble des p répondant à cette question forme une érosion de X par B notée : XΘB. [14]

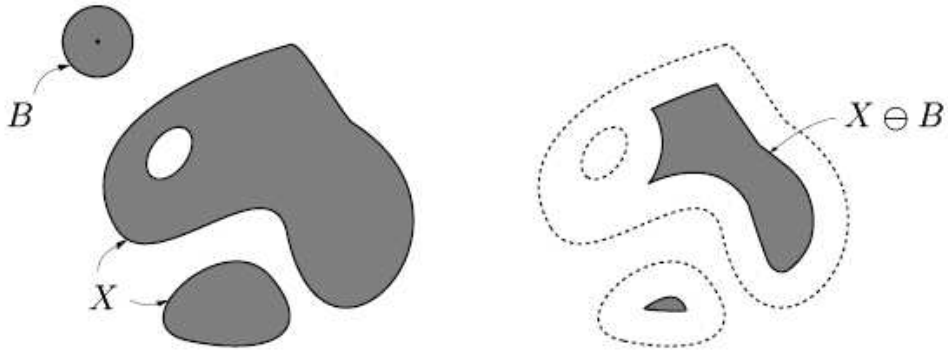


Figure I. 11:*érosion de X par B*

I.3.5.2 Dilatation :

L'opération de dilatation se définit d'une manière analogue, on obtient le dilaté de X par B en prenant la totalité de la surface de couverture lorsque l'origine de B occupe successivement toutes les positions dans X. Elle est notée par X⊕B. [15]

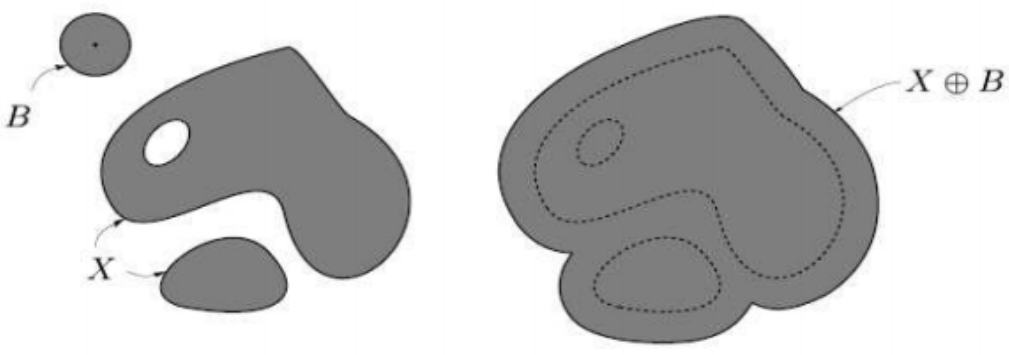


Figure I. 12:*dilatation de X par B*

I.3.5.3 L'ouverture :

L'ouverture a pour propriété d'éliminer toutes les parties des objets qui ne peuvent pas contenir l'élément structurant. [11]

Les Opérations "Erosion" et "Dilatations" peuvent être combinées et une opération d'érosion suivie d'une dilatation s'appelle une ouverture et notée par X ° B Où : X ° B = : (X Θ B) ⊕ B. [14]

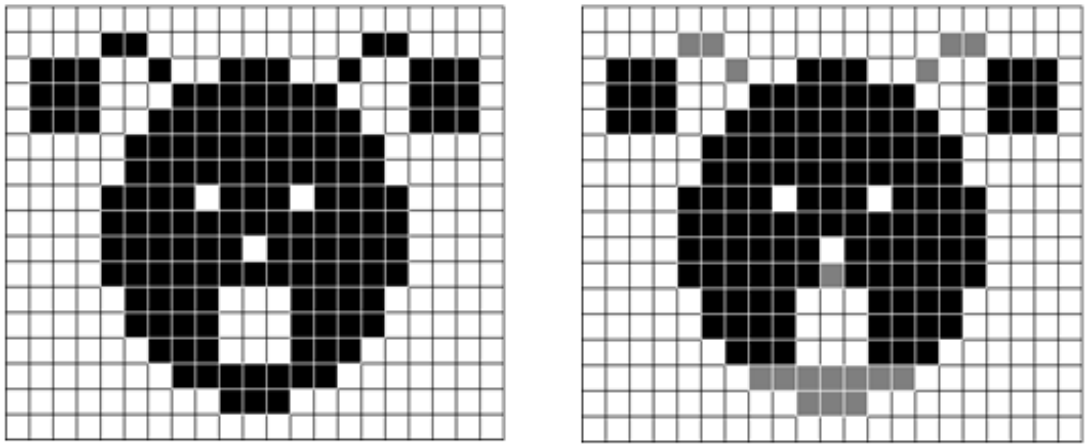


Figure I. 13: *Ouverture par un carré 3×3*

I.3.5.4 La fermeture :

L'opération de fermeture est obtenue en inversant l'ordre des opérations utilisées pour définir l'ouverture, elle notée par X ● B, Où : X ● B = : (X⊕ B) Θ B. [14] Il a pour propriété de combler tout ce qui est de taille inférieur à l'élément structurant. [11]

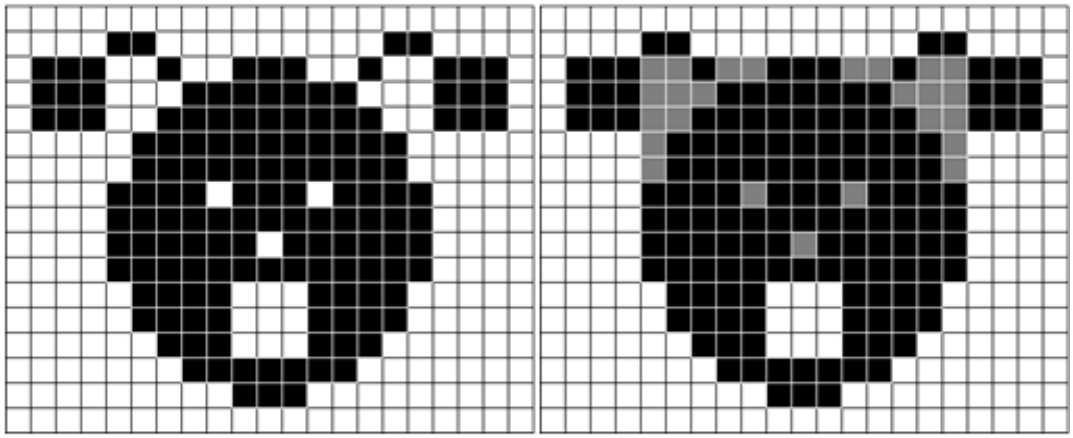


Figure I. 14:*Fermeture par un carré 3×3*

I.3.6 Segmentation d'images :

Les méthodes de segmentation s'appliquent sur des images matricielles, afin d'extraire une ou plusieurs zones d'intérêt connexes, qui ont un pouvoir discriminant spécifique vis-à-vis du domaine applicatif.

Les techniques de segmentation diffèrent selon le type d'application et selon l'information recherchée dans l'image. Les zones extraites peuvent être des points et leur voisinage, des lignes ou des régions. [16]

I.3.6.1 Différentes approches :

Le choix d'une méthode de segmentation est lié à plusieurs facteurs [17].Comme : la nature de l'image, les conditions d'acquisition (bruit), les primitives à extraire (contours, textures, …etc.) et bien évidemment les contraintes d'exploitation (fonctionnement en temps réel, type, mémoire vive et physique disponible). [18]

1. **Approche contours :**

De façon générale, un contour est défini comme étant la frontière entre deux régions. La détection de contours consiste à repérer les points d'une image numérique qui correspondent à un changement brutal de l'intensité lumineuse. [5]

Les méthodes de détection de contours donnent de bons résultats quand les contours de l'image sont bien définis. Cependant, dans le cas des images bruitées ou faiblement contrastées, les méthodes contours nécessitent une étape supplémentaire afin de fermer les bords des régions. [6]

2. Approche régions :

Les approches régions ont pour but de mettre en évidence les régions homogènes de l'image. Il s'agit de rechercher des ensembles de pixels partageant des propriétés communes. Les régions sont différenciées entre elles par des propriétés élémentaires basées sur des critères locaux tels que le niveau de gris de chaque pixels, ou bien sur un attribut estimé dans le voisinage du pixel tel que la valeur moyenne, la variance ou des paramètres de texture.

3. Approche par classification de pixels :

Ce type de méthode considère une région comme un ensemble de pixels connexes appartenant à une même classe. Elles supposent donc que les pixels qui appartiennent à une même région possèdent des caractéristiques similaires et forment un nuage de points dans l'espace des attributs. [5]

I.4 La reconnaissance automatique de plaque d'immatriculation :

I.4.1 Définition :

La reconnaissance de la plaque d'immatriculation d'un véhicule est un système de traitement d'image par lequel il est utilisé pour reconnaître les véhicules en identifiant la plaque d'immatriculation. Il peut être utilisé pour le contrôle de la circulation, la sécurité et le contrôle d'accès au stationnement (parking). [20] On distingue deux parties importantes du système :

- d'une part, le capteur vidéo permettant l'acquisition des images et d'autre part, le matériel informatique sur lequel fonctionnent les logiciels de traitement des images acquises.
- La partie logicielle du système s'exécute selon le cas sur un matériel PC standard ou sur un composant dédié, et peut parfois être liée à d'autres applications ou bases de données. [21]



Figure I. 15:*image d'un véhicule à reconnaitre à l'entré d'un parking*

I.4.2 Caractéristiques des plaques algériennes :

En Algérie depuis l'indépendance, on n'a eu que deux types de plaques. Les premières à caractères blancs sur fond noir se composaient de la première lettre du nom du département de l'époque (A pour Alger, B pour Batna, C pour Constantine, O pour Oran…), d'un chiffre et de deux lettres code. Avec le changement administratif intervenu durant les années 80, la plaque a été complètement modifiée. Sur un fond blanc et caractères noirs à l'avant et un fond jaune et caractères noirs à l'arrière, elle est constituée uniquement de chiffres, un numéro d'identification, suivi du millésime et du code de la wilaya. [22]



Figure I. 16:*exemple d'une plaque d'immatriculation algérienne*

Nous commençons par la droite :

- **26** correspond au code de la wilaya.
- **11** correspond à l'année de fabrication du véhicule (2011).
- **1** correspond au type de l'automobile. Ici c'est 1 car c'est une voiture touristique, 2 pour les camions, etc.
- **14050** signifie que ce véhicule est le 14050ème qui est entré dans la wilaya 26 en 2011.

- La plaque a une forme de rectangulaire, Le style des caractères varie d'une plaque à l'autre et la distance entre les caractères n'est pas standardisé, Elle dépend du constructeur de la plaque.
- Les plaques sont blanches (à l'avant), jaunes (à l'arrière) et la couleur des caractères est noire.
- Récemment les gens aiment ajouter quelques illustrations sur la plaque d'immatriculation de leur voiture, comme par exemple le drapeau Algérien, un numéro de téléphone, le nom de la ville ou du pays, le nom d'une entreprise, etc. [3]

I.4.3 Difficultés :

Ces systèmes ont été développés il y'a de ça une vingtaine d'années et leurs performances dépend des conditions environnantes et doivent être capables de gérer un grand nombre de difficultés possibles. Parmi lesquelles :

- Une mauvaise résolution de l'image à cause d'une plaque trop éloignée ou d'une caméra de mauvaise qualité.
- Des images floues souvent à cause du mouvement.
- Un mauvais éclairage et un faible contraste, un reflet ou une ombre, un objet obscurcissant une partie de la plaque (barre de remorquage ou de la poussière).
- Une police de caractère trop originale.
- éclairage puissant. [12]



Figure I. 17:*les difficultés*

I.4.4 Etapes de reconnaissance de plaques d'immatriculations :

Le système commence par capturer l'image du véhicule dès que celui-ci atteint la zone de contrôle de sécurité. Les images capturées sont ensuite extraites à l'aide du processus de segmentation. La reconnaissance optique de caractères est utilisée pour identifier les caractères. Les données obtenues sont ensuite comparées aux données stockées dans leur base de données. Le système sera implémenté et simulé sur MATLAB et les performances sont testées sur des images réelles. [23]

I.4.4.1 Image de véhicule :

L'image du véhicule dont la plaque d'immatriculation doit être identifiée et capturée à l'aide d'un appareil photo numérique.

I.4.4.2 Extraction de la plaque d'immatriculation (localisation de la plaque) :

Dans cette étape, la plaque d'immatriculation est extraite en premier convertir une image RVB c'est-à-dire l'image capturée en image en niveaux de gris. Ici la morphologie mathématique est utilisée pour détecter la région et l'opérateur Canny sont utilisés pour calculer la valeur de seuil. Après cela, nous obtenons une image dilatée. Ensuite une fonction « imfill » est utilisé pour combler les trous afin que nous obtenions une image binaire claire. [24]

I.4.4.3 La segmentation :

L'un des sujets les plus populaires dans l'étude du traitement d'images est la segmentation d'images. Le processus de segmentation devient important pour le traitement de l'image afin de trouver les informations significatives à partir des régions significatives qui représentent un niveau de données supérieur. L'analyse de l'image nécessite une grande quantité de données de faible niveau, exprimées en pixels, pour être extraites en informations significatives.

Le but de la segmentation est de trouver des régions qui représentent des parties significatives des objets. Dans la segmentation, l'image sera divisée en régions en fonction de l'intérêt de l'étude.

I.4.4.4 La reconnaissance des caractères :

La reconnaissance des caractères est la tâche la plus importante dans la reconnaissance du numéro de plaque. La reconnaissance des caractères est un problème qui a beaucoup retenu l'attention dans les domaines du traitement d'images, de la reconnaissance des formes et de l'intelligence artificielle. C'est parce qu'il est fort possible que le caractère produit à partir de l'étape de normalisation diffère de celui de la base de données.

Les mêmes caractères peuvent différer par leur taille, leur forme et leur style, ce qui pourrait entraîner la reconnaissance d'un caractère faux, affecter l'efficacité et accroître la complexité de l'ensemble du système. Il est important que le système différencie correctement le caractère car il peut parfois être confus du fait des similitudes de forme. [25]

I.4.5 Les différents systèmes existants :

On trouve les différents systèmes existants pour la reconnaissance des plaques d'immatriculation comme suit :

- Autovu**.**
- LAPI ENGINE.
- SeeTec.
- Asia Vision Technology Limited (AVT).
- AGL (Application de Gestion LAPI).
- Système LAPI-Pryncar.

I.4.6 Applications de système de reconnaissance automatique de formes :

La lecture automatique de plaques minéralogiques peut également être utilisée pour:

- Les passages de frontière.
- Les stations-services (enregistrement quand un client part sans payer).
- Le contrôle d'accès des parkings ou routes privées: ouverture automatique, ou enregistrement de l'entrée.
- Un outil de marketing pour enregistrer les modes de consommation.
- Les systèmes de gestion de la circulation, qui calculent la vitesse de circulation en mesurant le temps entre les passages devant deux points de lecture.
- Comparer les plaques d'immatriculations au Fichier des véhicules volés.
- Pour certaines applications, le système peut être associé à d'autres algorithmes de reconnaissance de type facial ou de véhicule. [3]

I.5 Conclusion :

Après une présentation du système mise en œuvre dans ce projet qui a pour but la reconnaissance des caractères de la plaque, Une description des étapes a été expliquée en général afin de mieux comprendre l'objectif global de ce projet.

Ce chapitre était une entrée collective de ce mémoire ou on a présenté et décrit une approche générale sur la reconnaissance de formes ainsi qu'une étude courte et brève sur une base de traitement d'image.

Chapitre II:

Implémentation d'une application de reconnaissance de chiffres par MATLAB

II.1 Introduction :

La reconnaissance des plaques d'immatriculation comprend trois étapes essentielles qu'on a montrées dans chapitre précédant : la première étape est la détection de la région de la plaque d'immatriculation, la deuxième étape est une étape de segmentation pour isoler le caractère, et à la fin c'est la reconnaissance des chiffres.

Dans ce chapitre nous concentrons sur les étapes de reconnaissance de chiffres à l'aide du logiciel «Matlab » qui est le meilleur choix pour mettre en œuvre le système de détection et de reconnaissance de plaques d'immatriculation comme une première partie ensuite la deuxième partie ,on va créer l'interface Matlab « GUI » pour l'implémenter.

II.2 Partie 1 : conception d'un système de reconnaissance de plaques d'immatriculations basée sur MATLAB :

II.2.1 Le fonctionnement du système de plaque d'immatriculation du véhicule :

Le système de plaque d'immatriculation fonctionne comme suit:

Premièrement, le véhicule s'arrêtera au portique automobile. Le cycle commencera lorsque le véhicule franchira le détecteur. Il activera un signal au système de plaque d'immatriculation du véhicule pour signaler la présence du véhicule.(voir la figure II.1)

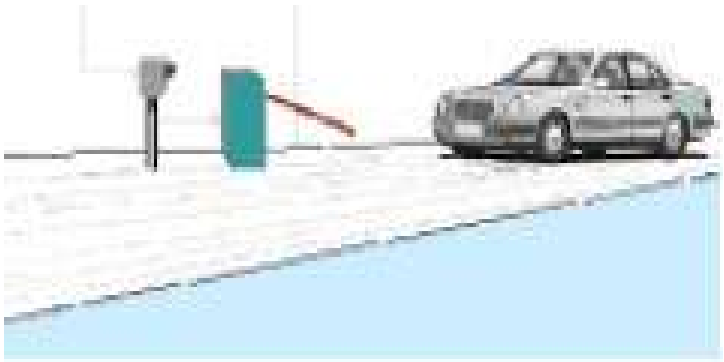


Figure II. 1:*l'arrêt de véhicule*

Deuxièmement, l'éclairage (infrarouge) sera activé et des images de la photo de face du véhicule seront prises. Le système lira les pixels d'information du véhicule et lancera le processus de reconnaissance.(voir la figure II.2)



Figure II. 2:*la détection de la présence de véhicule par infrarouge*

Troisièmement, le système appliquera certains algorithmes pour analyser l'image du véhicule. Si le numéro de la plaque d'immatriculation se trouve dans la liste prédéfinie, la barrière sera levée pour permettre au véhicule de traverser. Sinon, l'entrée du véhicule sera refusée. [20] (voir la figure II.3)



Figure II. 3:*l'ouverture de la barrière*

Les étapes de fonctionnement suivies par le logiciel sont montrées par l'algorithme ci-dessous:

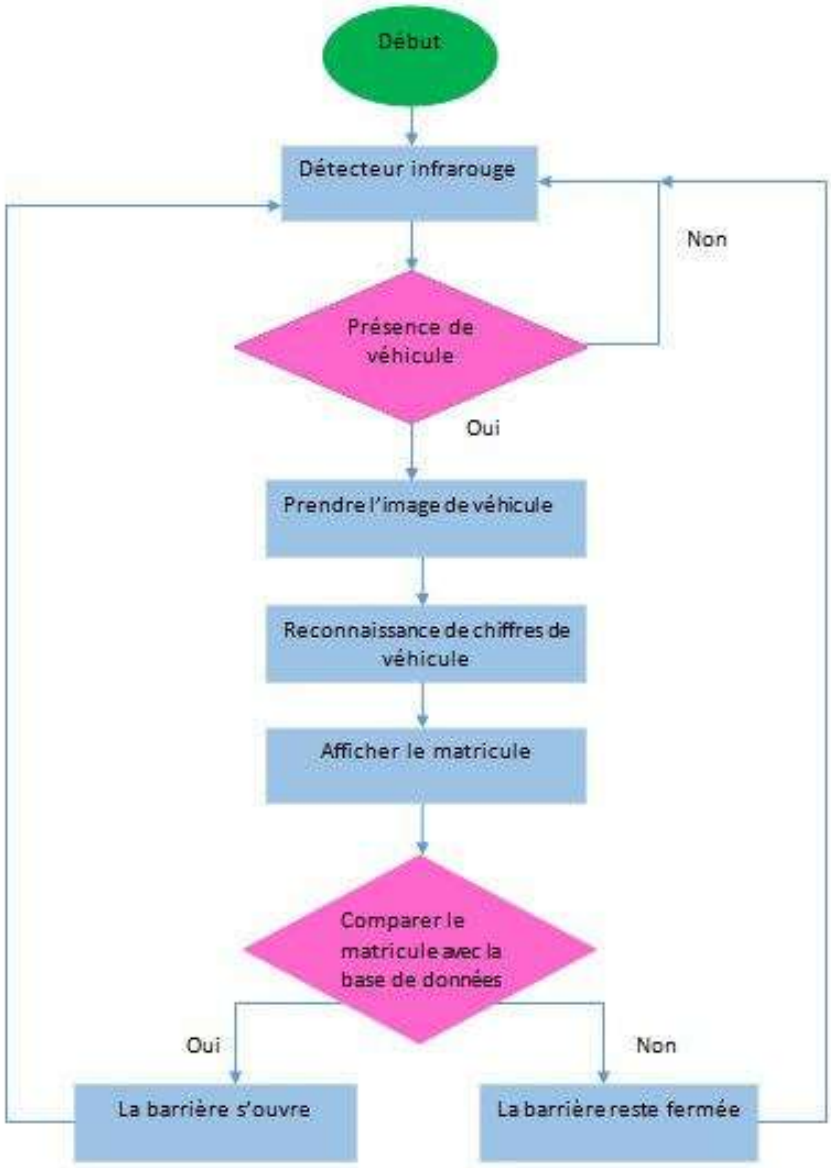


Figure II. 4:*l'organigramme de parking*

II.2.2 Contraintes :

En raison du temps limité dont nous disposons et du fait que nous traitons avec un logiciel de vision, il n'est pas recommandé d'inclure tous les cas possibles. Nous devons donc établir une liste de contraintes pour rendre le projet plus systématique et plus facile à gérer. La contrainte est listée ci-dessous:

- Image prise uniquement lorsque le véhicule est à l'arrêt.
- Image capturée du véhicule à distance fixe.
- Image capturée du véhicule à angle fixe.
- La position de la plaque d'immatriculation du véhicule doit être centrée.

- Prenez uniquement l'image de la voiture vue de face.
- Images capturées sur un lieu où la lumière est proportionnelle. [26]

II.2.3 Outil de développement « MATLAB » :

II.2.3.1 Introduction sur matlab :

MATLAB est choisi comme un développement principal de l'application de reconnaissance de plaque d'immatriculation.

Par ailleurs, MATLAB est également très efficace car il dispose d'outils fonctionnels intégrés pour le traitement des images.

II.2.3.2 Les avantages de Matlab :

- Les données peuvent être représentées sous forme de matrices ou de vecteurs.
- Le code MATLAB est plus court et plus simple.
- L'algorithme utilisé dans MATLAB peut être converti pour être utilisé dans le matériel.
- Le système fonctionnera plus rapidement avec l'application «approche matricielle».
- MATLAB possède des outils de visualisation graphique puissants. [27]

II.2.4 Schéma du système de reconnaissance de plaque d'immatriculation :

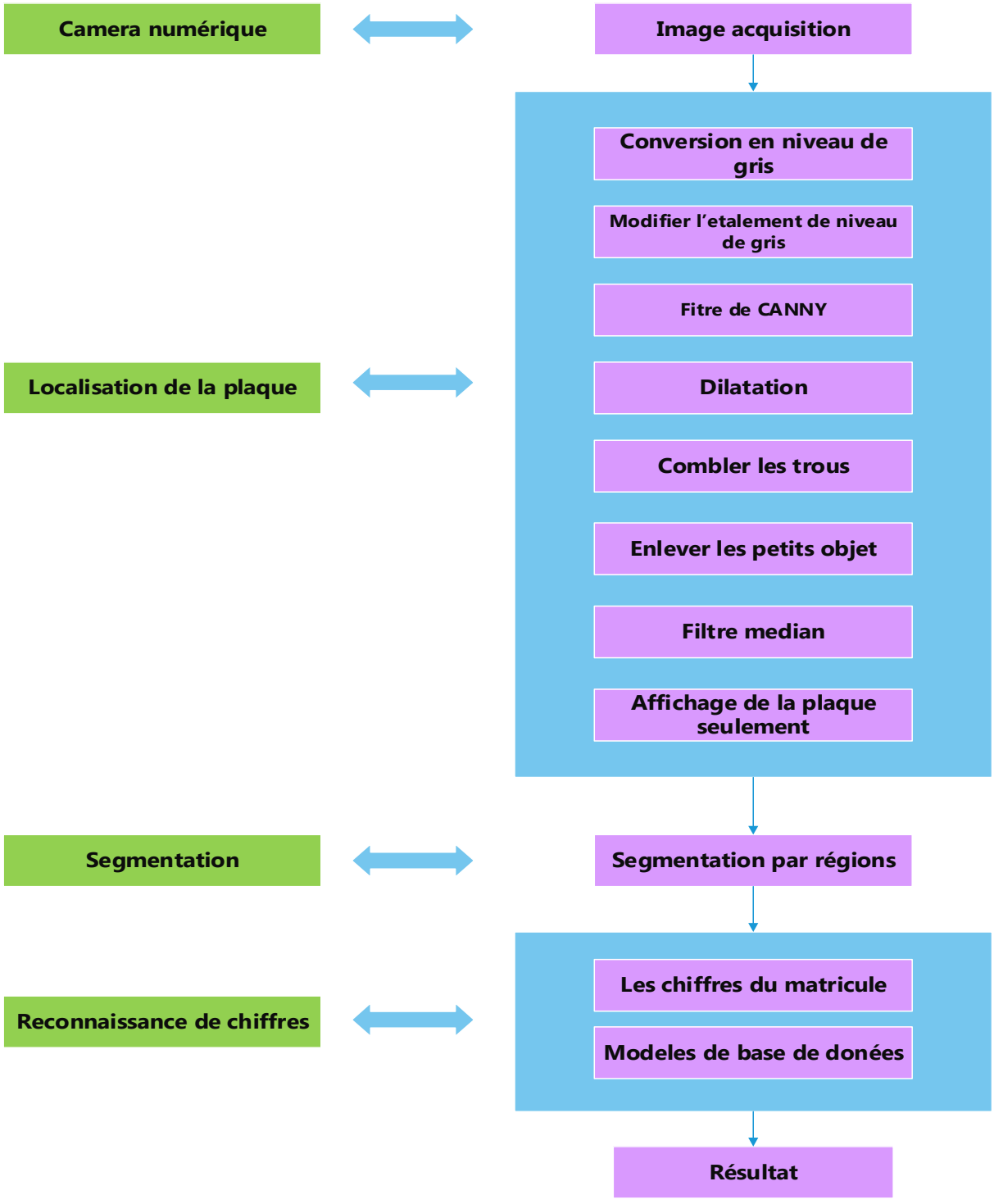


Figure II. 5:*Schéma général du système de reconnaissance de plaque d'immatriculation*

II.2.4.1 Image originale de véhicule :

La phase initiale du traitement des images pour la reconnaissance de la plaque d'immatriculation d'un véhicule consiste à obtenir des images de véhicules. Des dispositifs électroniques tels que des

caméras optiques (numériques / vidéo), des webcams, etc. peuvent être utilisés pour capturer les images acquises. [20] on a utilisé la commande « imread » qui permet la lecture de l'image.

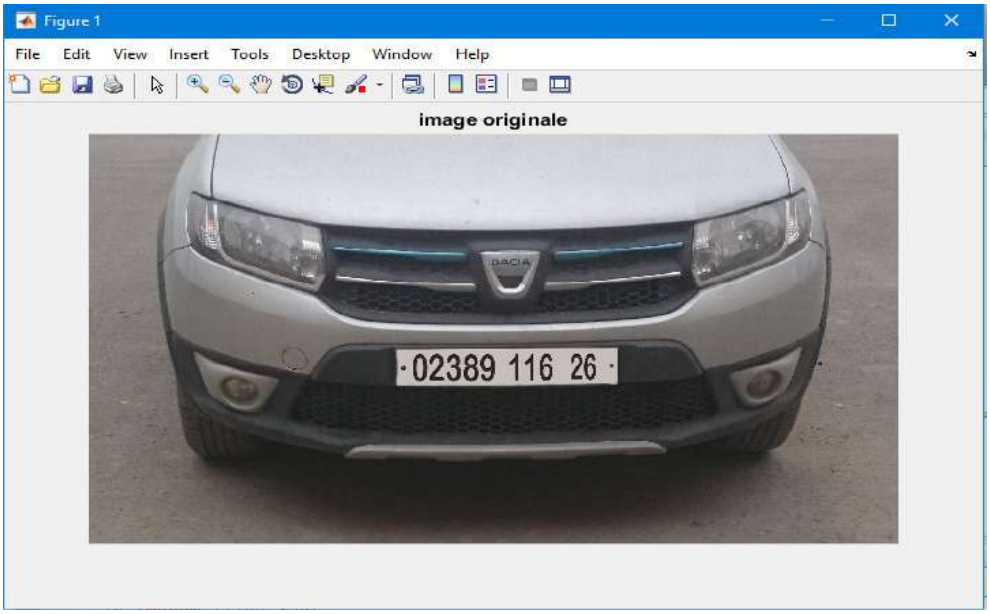


Figure II. 6:*image originale de véhicule sur MATLAB*

II.2.4.2 Localisation de la plaque :

1. Convertir l'image en niveau de gris :

Cette étape implique la conversion de l'image en niveaux de gris. Les images en couleur sont converties en image grise. Selon la valeur R, G, B de l'image, il calcule la valeur de gris et obtient l'image grise en même temps. [28] on a utilisé la commande « rgb2gray » pour convertir l'image originale en image en niveau de gris.

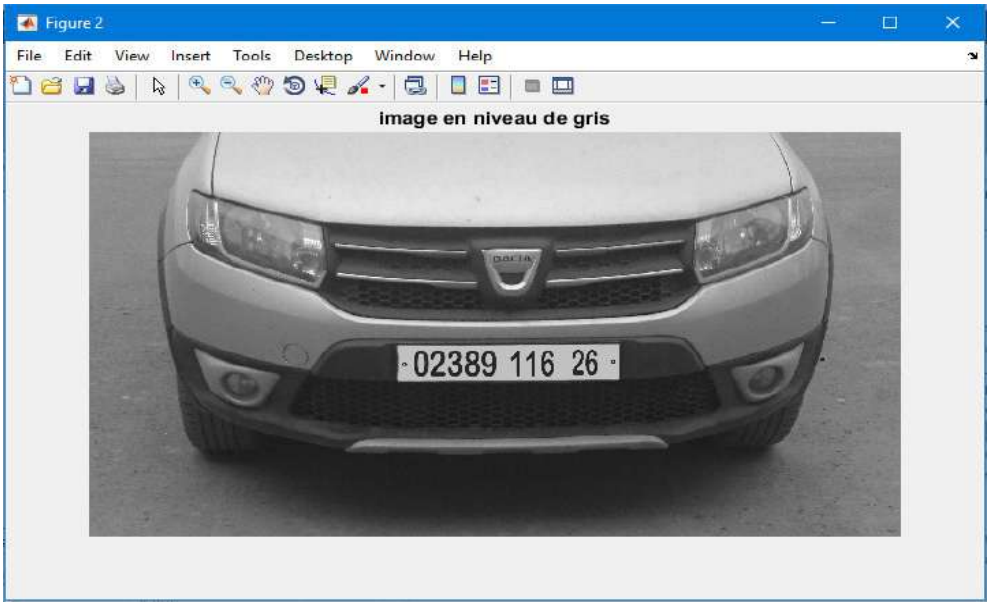


Figure II. 7:*image en niveau de gris sur MATLAB*

2. Modifier l'étalement de niveau de gris :

C'est-à-dire modifier le contraste de l'image en niveau de gris en faisant correspondre les valeurs d'intensité de l'image à des nouvelles valeurs tel que, par défaut, 1% des données est saturé à des intensités plus faibles et élevées des données d'entrée. [11] On a utilisé la commande « imadjust » pour cette conversion sur MATLAB.



Figure II. 8:*modification de l'étalement de niveau de gris sur MATLAB*

3. Filtre de « Canny » :

Pour notre système on va utiliser une approche traitée pour localiser la plaque, à cette fin, [28] le détecteur de contour Canny est un opérateur de détection de bord qui utilise un algorithme à plusieurs étapes pour détecter une large gamme de bords dans les images. [29] on a utilisé la commande « edge(i,'canny',0.4) » sur MATLAB.

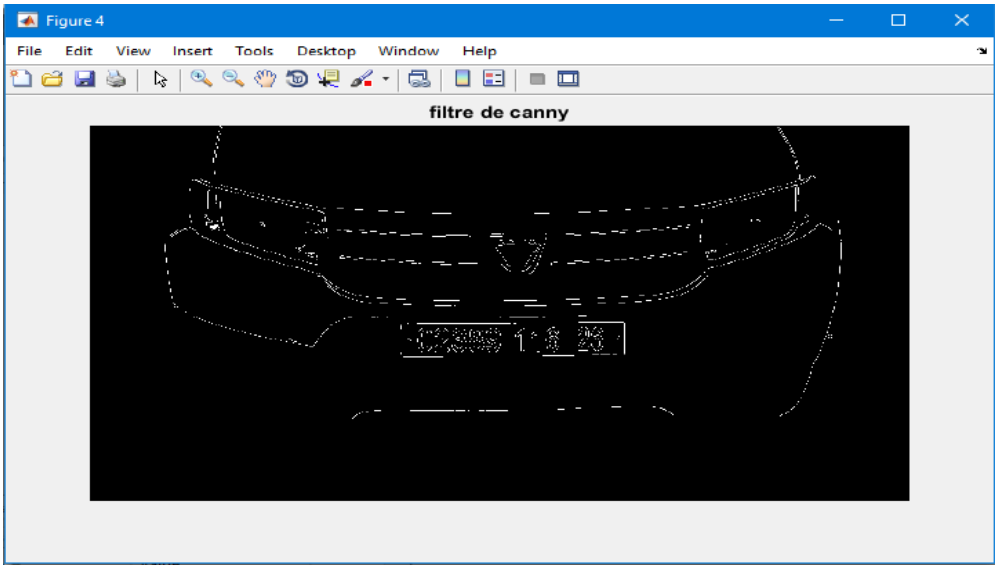


Figure II. 9:*filtre de CANNY sur MATLAB*

4. Dilatation morphologique :

Lors d'une dilatation tous les objets vont "grossir" d'une partie correspondant à la taille de l'élément structurant [12]. Elle consiste à improviser une image donnée en comblant les trous dans une image. En utilisant la dilatation, le bruit dans une image peut également être supprimé. [30] On a utilisé la commande « imdilate » pour cette opération sur MATLAB.

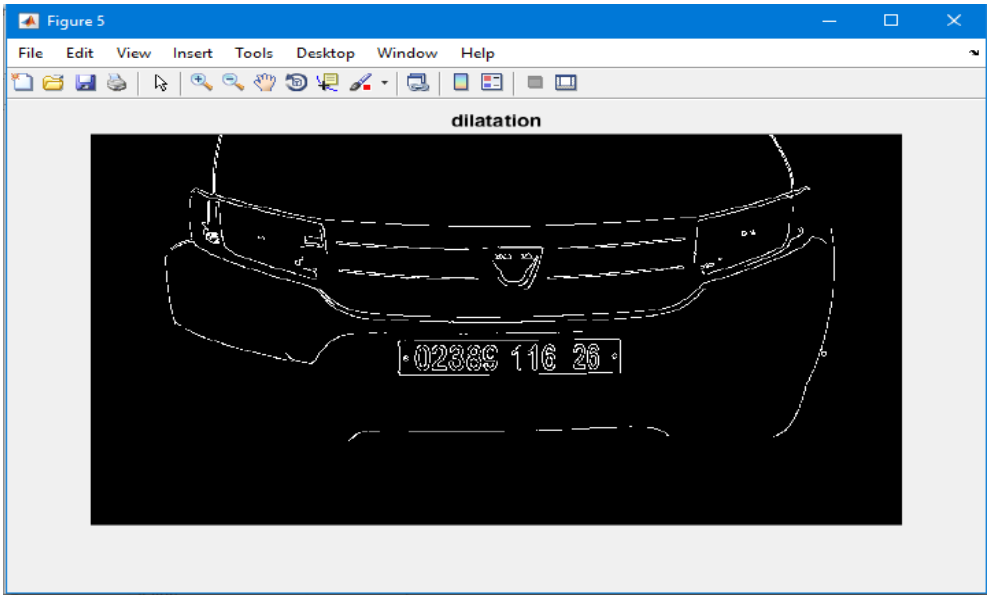


Figure II. 10:*dilatation sur MATLAB*

5. Remplir les trous :

S'il existe des trous dans les objets, c'est à dire des "morceaux" de fond à l'intérieur des objets, ils seront comblés [12]. On a utilisé la commande « imfill » sur MATLAB pour remplir les trous.

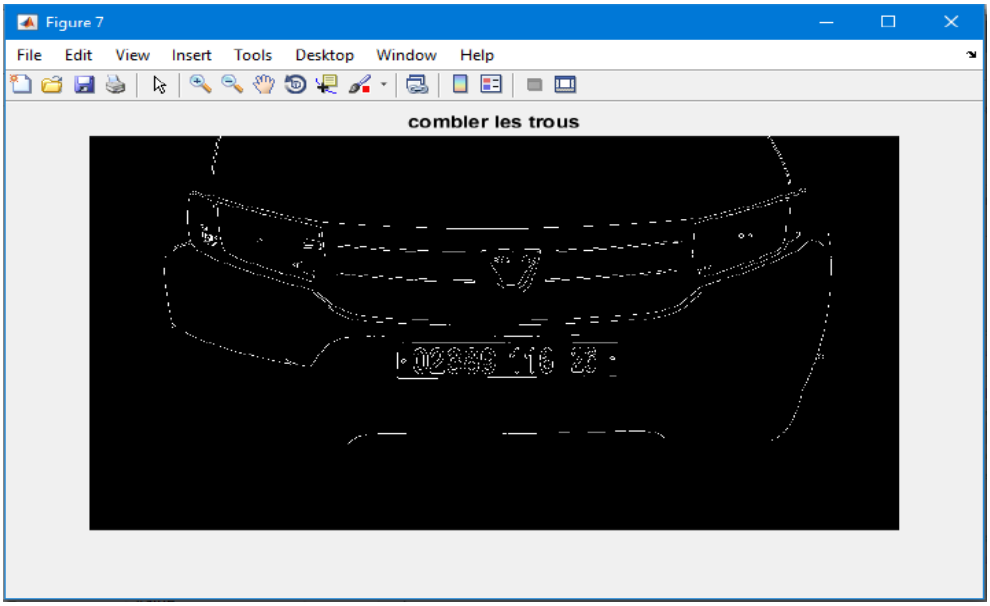


Figure II. 11:*Remplir les trous sur MATLAB*

6. Suppression des petits objets :

Si des objets sont situés à une distance moins grande que la taille de l'élément structurant, ils vont fusionner. [12] La commande « bwareaopen », basée sur un analyse en composantes connexes, permet de supprimer des régions de trop petites tailles dans une image binaire. [31]

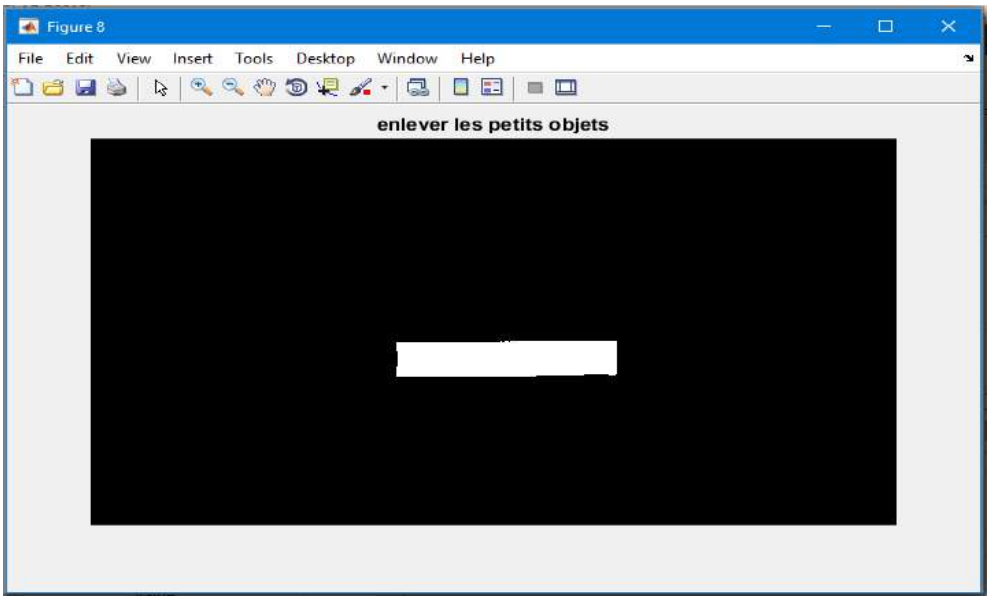


Figure II. 12:*enlevé les petits objets sur MATLAB*

7. Filtre médian :

L'objectif avoué du filtrage est de réduire les variations d'intensité au sein de chaque région de l'image tout en respectant l'intégrité des scènes, L'idée principale du filtre médian est de remplacer chaque entrée par la valeur médiane de son voisinage. [32] la commande « medfilt2 » est utilisé pour cette opération.

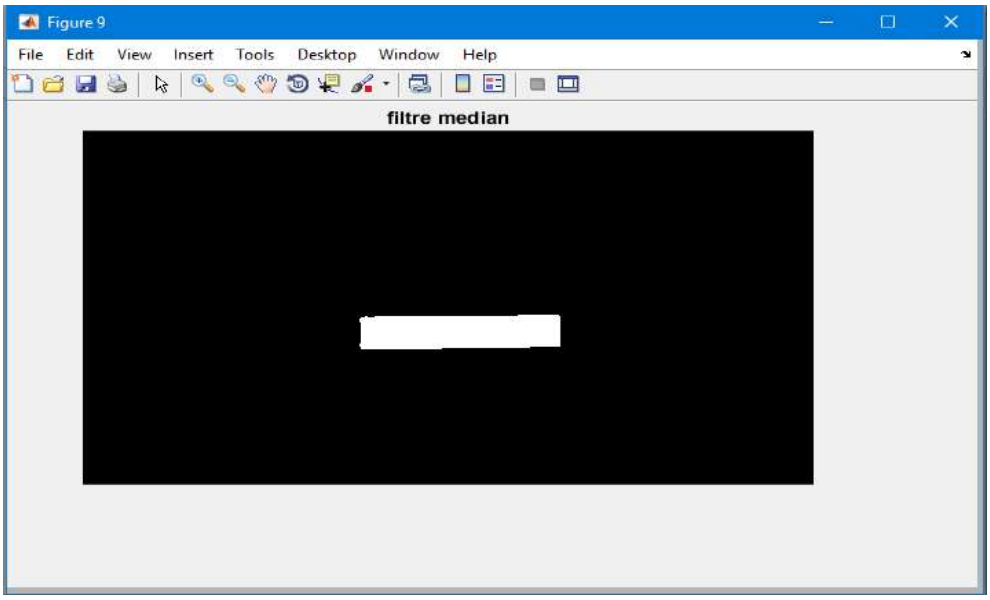


Figure II. 13:*filtre médian sur MATLAB*

8. Afficher la plaque seule :

C'est-à-dire on a supprimé tout le noir autour la plaque et on l'a affiché toute seule sur MATLAB.



Figure II. 14 : *affichage de la plaque à reconnaitre sur MATLAB*

II.2.4.3 Segmentation de la plaque :

La segmentation d'une image implique la division de l'image en régions d'attribut similaire. [33] Dans cette méthode, nous souhaitons regrouper les régions afin que chacun des pixels de la région ait une valeur similaire pour les propriétés. [25]

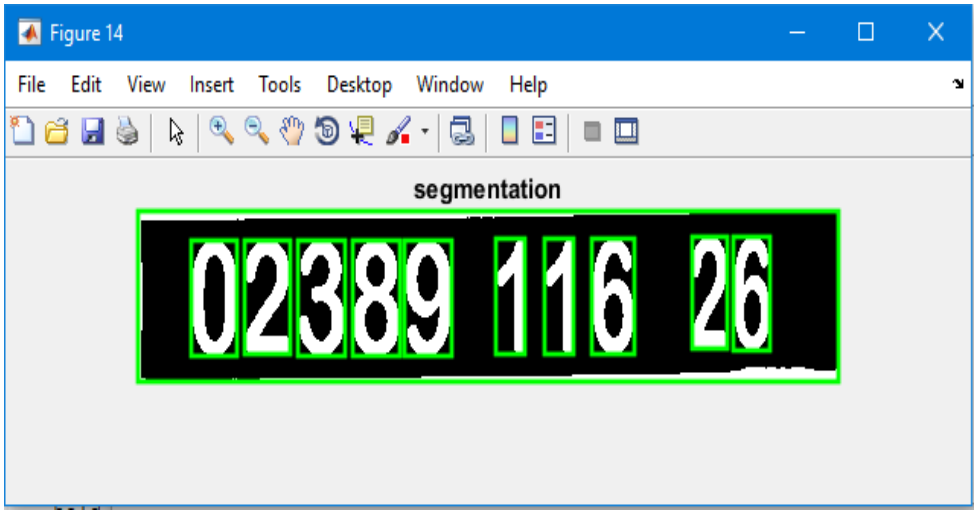


Figure II. 15:*segmentation de la plaque sur MATLAB*

II.2.4.4 La reconnaissance des chiffres et l'Affichage du numéro de la plaque :

C'est un ensemble de techniques informatiques qui réalisent une reconnaissance optique de la forme des caractères, après captation d'image par procédé optique. Le résultat est transmis sous forme d'un texte, la décision qui est basée sur un modèle, l'image segmentée est comparée à une image qui est stocké dans la base de données appelée image modèle. [31]

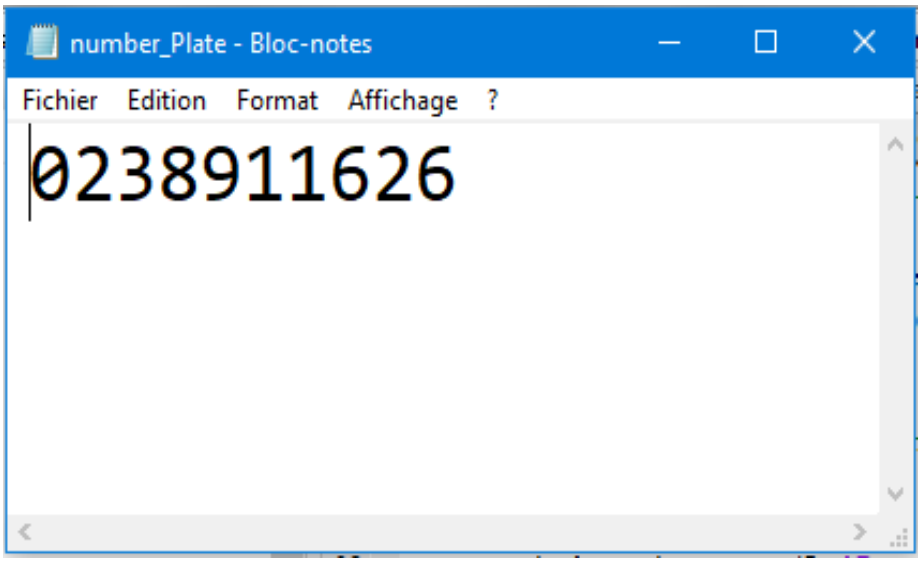


Figure II. 16:*l'affichage de matricule sur MATLAB*

- **La base de données :**

La base de données de correspondance de modèle et le caractère extraits de l'image de la voiture ont été corrélés pour reconnaître les lettres et les chiffres. Dans ce cas, les caractères extraits ont été mis en corrélation avec la base de données de modèles, composée de 0 à 9 chiffres, La taille de chaque image de caractère a été normalisée en fonction de la taille des caractères stockés dans la base de données, chaque caractère ajusté à 42 x 24 était corrélé pour une correspondance maximale. [32]

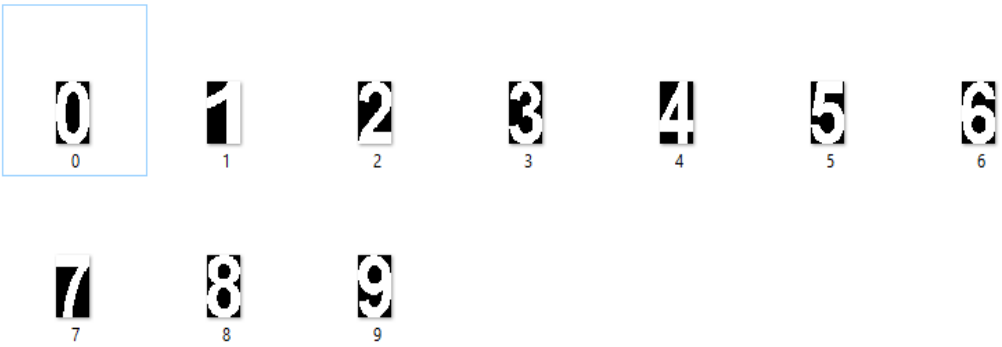


Figure II. 17:*la base de données*

II.3 Partie 2 : conception d'une application de reconnaissance de plaques d'immatriculations basé sur MATAB-GUI:

Toutes les opérations de traitement et de contrôle de la reconnaissance de plaque d'immatriculation et de l'acquisition d'images avec affichage des détails du numéro

sont contrôlées à l'aide de l'interface graphique MATLAB. Le modèle créé comporte principalement 7 boutons pour contrôler les opérations du système :

1. **Le bouton 1 :** sert à afficher l'image originale.



Figure II. 18:*le bouton 1 qui affiche l'image sur l'application MATLAB*

2. **Le bouton 2 :** permet de modifier l'image en image en niveau de gris.

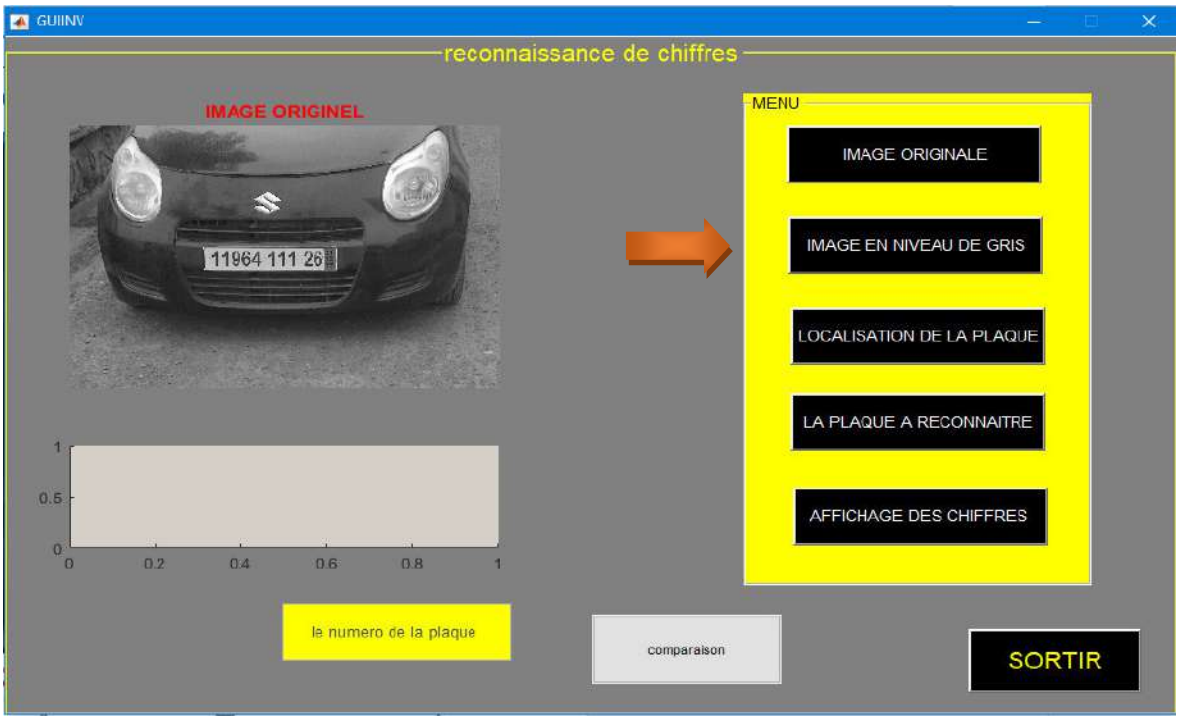


Figure II. 19:*le bouton 2 qui convertit l'image en niveau de gris sur l'application MATLAB*

3. **Le bouton 3** : utilisé pour effectuer la localisation de la plaque.

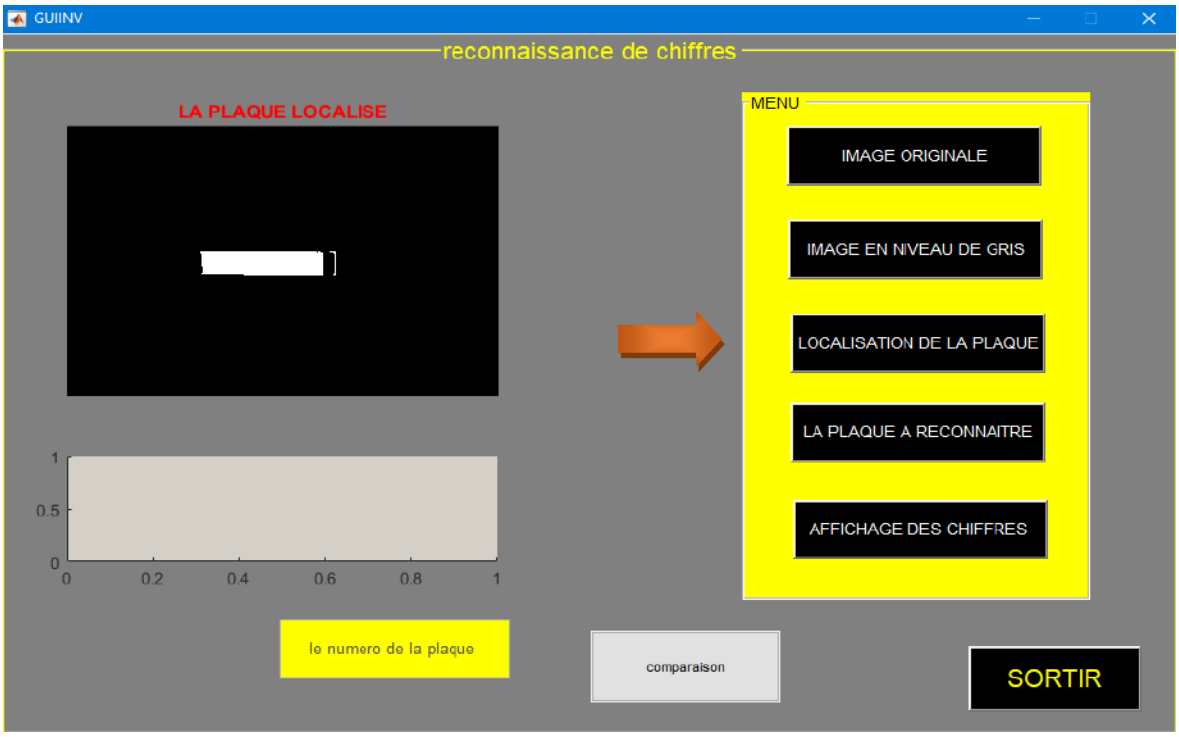


Figure II. 20:*le bouton 3 qui localise la plaque sur l'application MATLAB*

4. **Le bouton 4 :** afficher la plaque avec les chiffres.

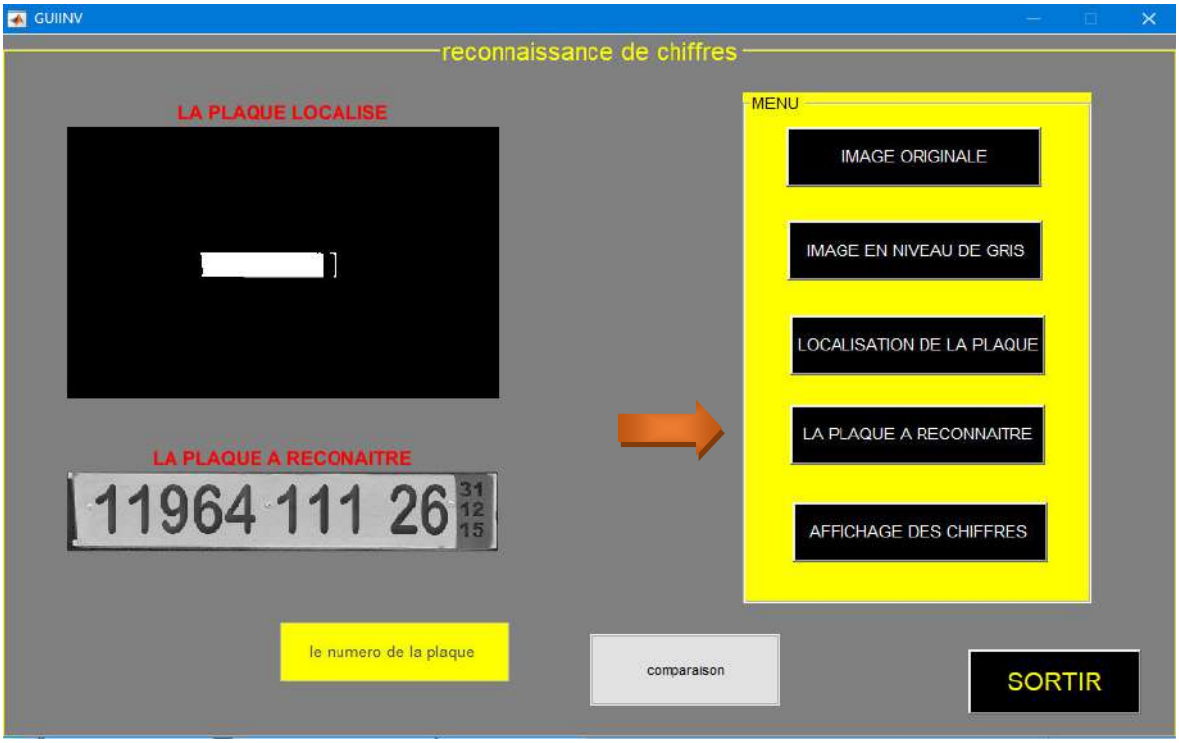


Figure II. 21:*le bouton 4 qui affiche la plaque à reconnaitre sur l'application MATLAB*

5. **Le bouton 5 :** permet de reconnaitre et afficher les numéros de la plaque.

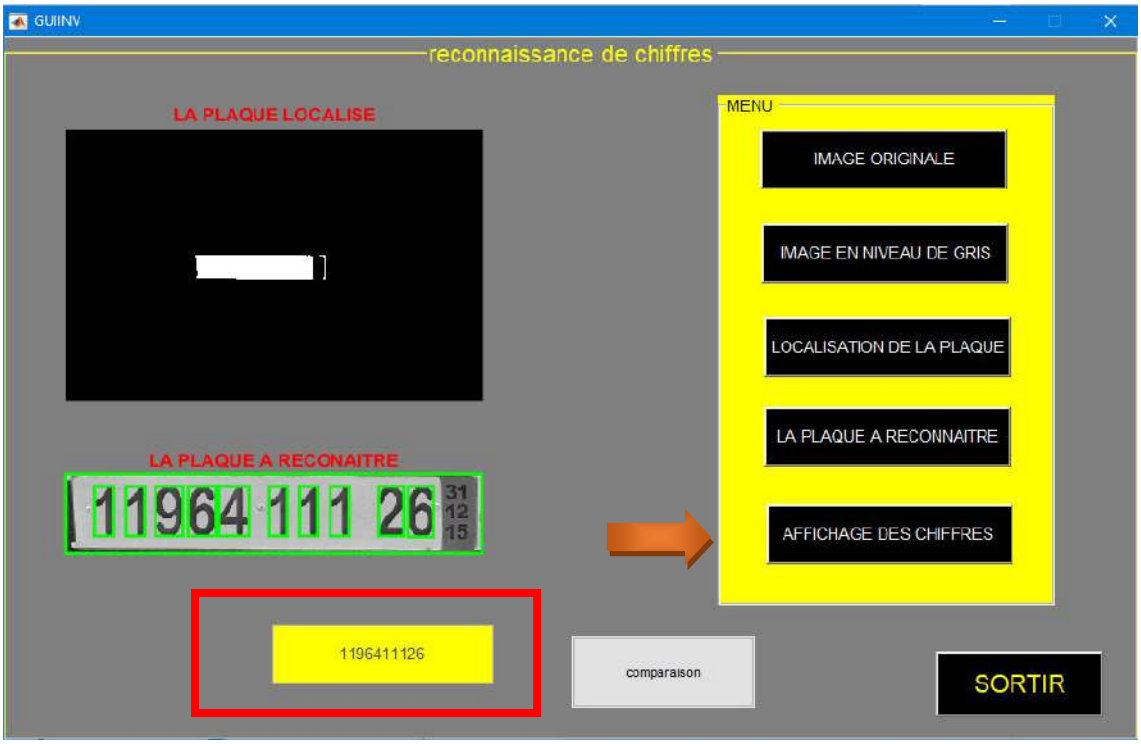


Figure II. 22:*le bouton 5 qui affiche les chiffres de la plaque sur l'application MATLAB*

6. **Le bouton 6 :** sert à comparer le matricule qu'on obtient avec la base de données qui contient les autres matricules qu'on va voir dans le prochain chapitre (chapitre III).

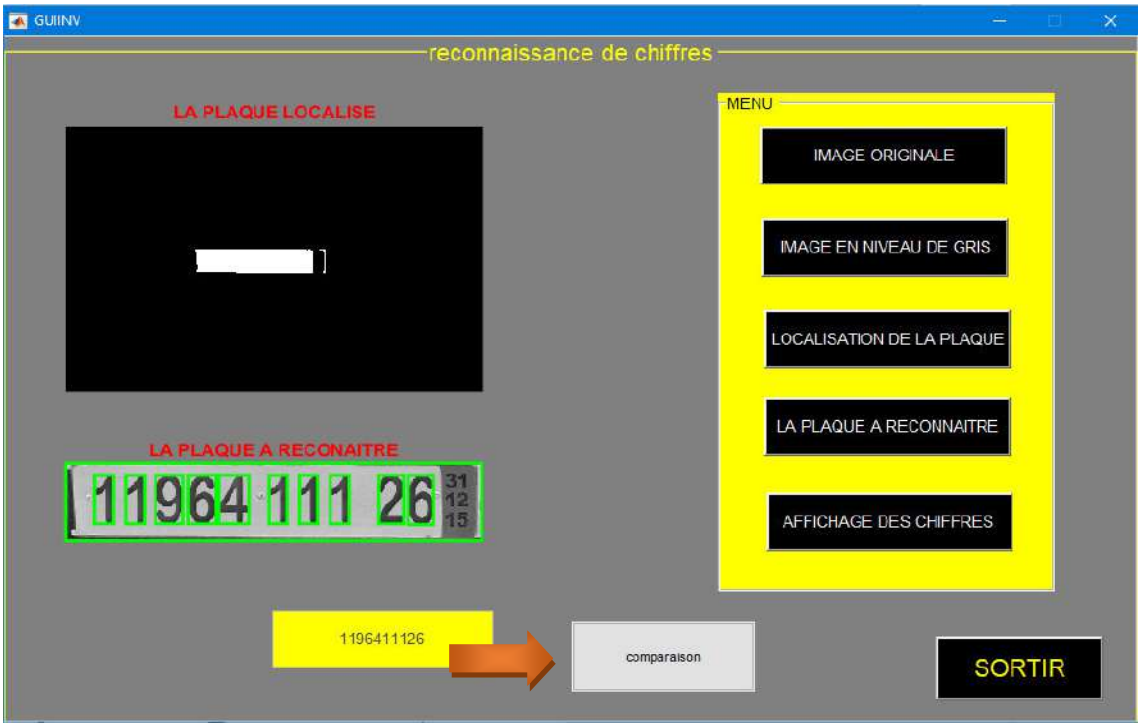


Figure II. 23:*le bouton 6 qui compare les matricule sur l'application MATLAB*

7. **Le bouton 7**: le dernier bouton, il permet de sortir de l'application.

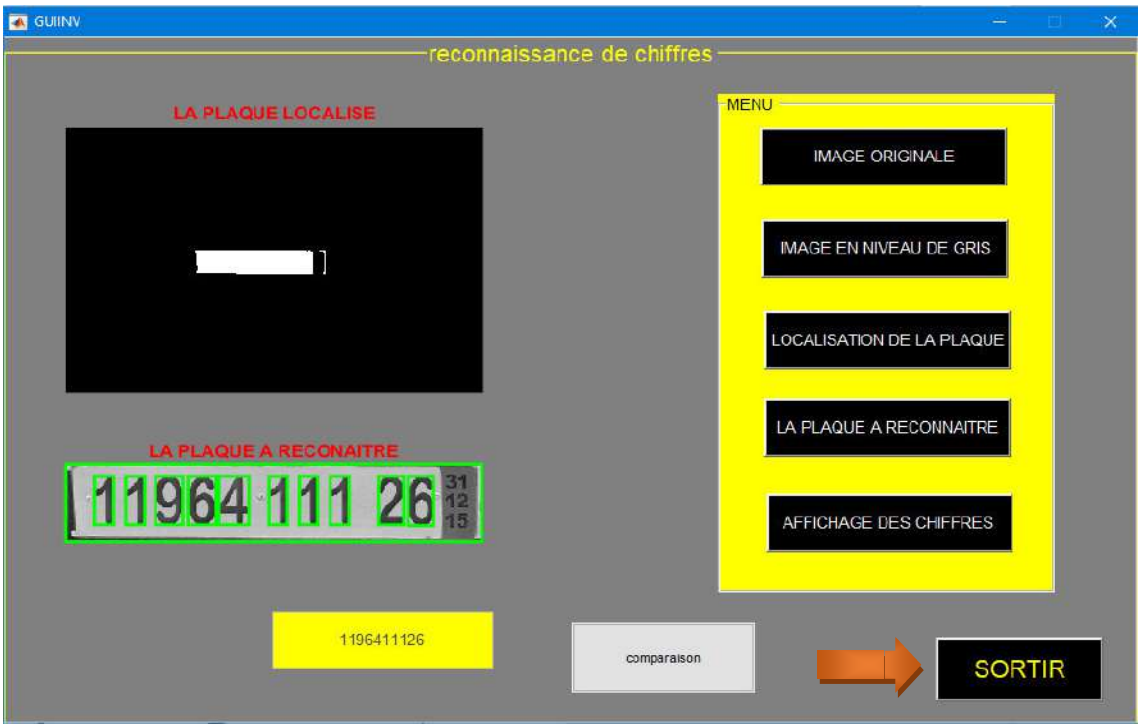


Figure II. 24:*le bouton 7 qui permet de sortir de l'application MATLAB*

II.4 Conclusion :

Dans ce 2ème chapitre nous avons crée une application qui détecte le numéro de plaque d'immatriculation du véhicule à l'aide de MATLAB qui est basé sur le traitement d'image.

Notre projet a réussi à créer un système de reconnaissance de chiffres à l'aide de MATLAB, Tout d'abord, nous avons extrait l'emplacement de la plaque. Nous avons ensuite séparé les caractères de la plaque individuellement par la segmentation. Enfin, nous avons appliqué un modèle correspondant par l'utilisation de la corrélation pour reconnaître la plaque.

On a également résumé notre travail sous forme d'une interface graphique simple et conviviale pour afficher les résultats du traitement de l'image et de l'identification du propriétaire du véhicule.

Chapitre III:

Réalisation d'un système d'accès de véhicule basé sur l'Arduino

III.1 Introduction :

La technologie domine notre vie quotidienne par l'utilisation des équipements électroniques à partir des projets qui rend notre vie plus simple et facile. L'arduino est l'un des moyens qui nous permet de faire des projets et qui donne la possibilité de regrouper l'électronique et la programmation afin de réaliser un grand nombre de plans dans tous les domaines.

Dans le 3éme chapitre nous allons présenter une approche générale sur l'arduino, ce qu'il y a réellement sur une carte Arduino dans la 1 ère partie. La 2éme partie sera consacré par faire un projet sous le nom « reconnaissance des plaques d'immatriculation »à l'aide de l'Arduino et par le logiciel Matlab qu'on a déjà crée et illustré dans le chapitre précédant.

III.2 Généralité sur l'Arduino :

III.2.1 Qu'est-ce que l'Arduino ?

Arduino est un projet créé par une équipe de développeurs, composée de six individus : Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, David Mellis et Nicholas Zambetti. Cette équipe a créé le "système Arduino". C'est un outil qui va permettre aux débutants, amateurs ou professionnels de créer des systèmes électroniques plus ou moins complexes.

III.2.2 But de la carte Arduino :

Le système Arduino, nous donne la possibilité d'allier les performances de la programmation à celles de l'électronique. Plus précisément, nous allons programmer des systèmes électroniques. Le gros avantage de l'électronique programmée c'est qu'elle simplifie grandement les schémas électroniques et par conséquent, le coût de la réalisation, mais aussi la charge de travail à la conception d'une carte électronique.

III.2.3 Les applications possibles :

Le système Arduino nous permet de réaliser un grand nombre de choses, qui ont une application dans tous les domaines, l'étendue de l'utilisation de l'Arduino est gigantesque. Voici quelque exemple :

- contrôler les appareils domestiques.
- donner une intelligence à un robot.
- réaliser des jeux de lumières.
- permettre à un ordinateur de communiquer avec une carte électronique et différents capteurs.

- télécommander un appareil mobile (modélisme). . . . etc. [35]

III.2.4 Les outils Arduino :

III.2.4.1 Partie matérielle :

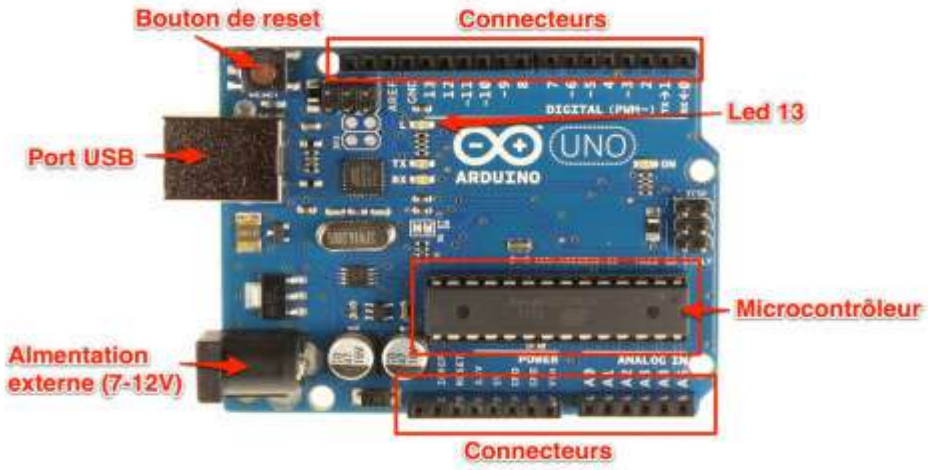


Figure III. 1: *plateforme Arduino*

1. **Le microcontrôleur**

C'est le cerveau de notre carte. Il va recevoir le programme que nous allons créer et va le stocker dans sa mémoire avant de l'exécuter. Grâce à ce programme, il va savoir faire des choses, qui peuvent être : faire clignoter une LED, afficher des caractères sur un écran, envoyer des données à un ordinateur, mettre en route ou arrêter un moteur…etc.

2. **L'alimentation**

Pour fonctionner, une carte Arduino a besoin d'une alimentation. Le microcontrôleur fonctionnant sous 5V, la carte peut être alimentée en 5V par le port USB ou bien par une alimentation externe qui est comprise entre 7V et 12V. Un régulateur se charge ensuite de réduire la tension à 5V pour le bon fonctionnement de la carte.

3. **La connectique :**

A part une LED sur la broche 13, la carte Arduino ne possède pas de composants (résistances, diodes, moteurs...) qui peuvent être utilisés pour un programme. Il est nécessaire de les rajouter. Mais pour cela, il faut les connecter à la carte. C'est là qu'interviennent les connecteurs, aussi appelés broches. Sur les Arduino et sur beaucoup de cartes compatibles Arduino, les connecteurs se trouvent au même endroit. Cela permet de fixer des cartes d'extension, appelée shields en les empilant.

Exploration des connecteurs Arduino :

- 0 à 13 Entrées/sorties numériques.
- A0 à A5 Entrées/sorties analogiques.
- GND Terre ou masse (0V).
- 5V Alimentation +5V.
- 3.3V Alimentation +3.3V.
- Vin Alimentation non stabilisée (= le même voltage que celui à l'entrée de la carte).

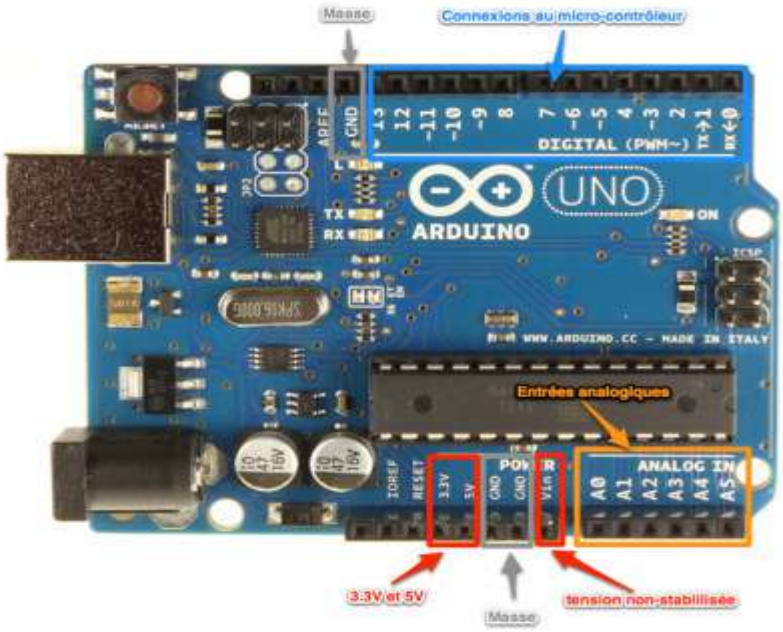


Figure III. 2:*la connectique de la carte Arduino*

Les connexions entre les composants sont réalisées par des jumpers, sortes de petits câbles.



Figure III. 3: *des jumpers*

4. La platine d'expérimentation :

Une platine d'expérimentation (appelée breadboard) permet de réaliser des prototypes de montages électroniques sans soudure et donc de pouvoir réutiliser les composants.

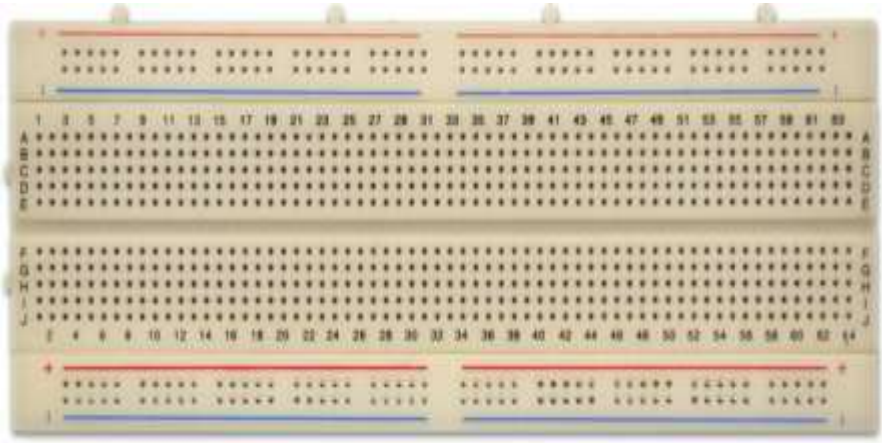


Figure III. 4:*la platine d'expérimentation*

Tous les connecteurs dans une rangée de 5 sont reliés entre eux. Donc si on branche deux éléments dans un groupe de cinq connecteurs, ils seront reliés entre eux. Il en est de même des alignements de connecteurs rouges (pour l'alimentation) et bleus (pour la terre). Ainsi, les liens peuvent être schématisés ainsi: [36]

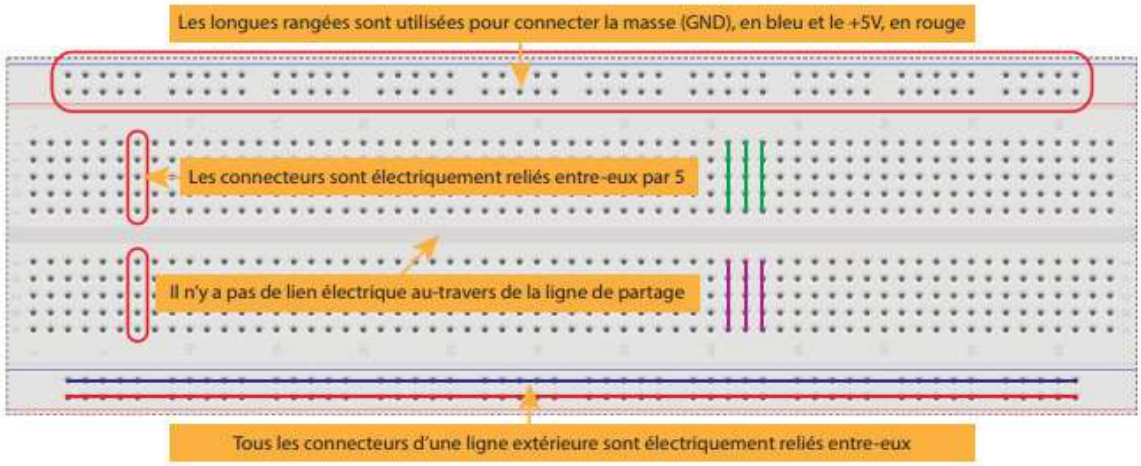


Figure III. 5:*les connecteurs dans une platine*

III.2.4.2 Partie logiciel :

1. L'environnement de la programmation :

Le logiciel de programmation de la carte Arduino sert d'éditeur de code (langage proche du C). Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information ce programme appelé IDE Arduino. [37]

2. Structure générale du programme (IDE Arduino)

Comme n'importe quel langage de programmation, une interface souple et simple est exécutable sur n'importe quel système d'exploitation Arduino basé sur la programmation en C. [35]

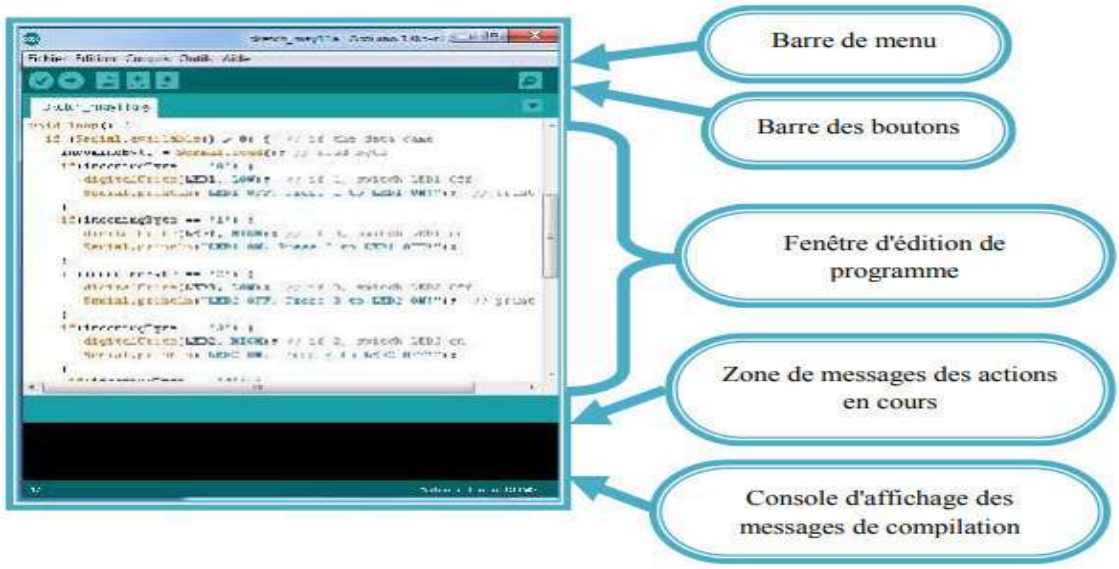


Figure III. 6:*interface Arduino IDE*

III.3 La carte Arduino «UNO » :

- Le modèle UNO de la société ARDUINO est une carte électronique dont le cœur est un microcontrôleur ATMEL de référence ATMega328. L'ATMega328 est un microcontrôleur 8bits de la famille AVR dont la programmation peut être réalisée en langage C/C++.
- La carte Arduino uno dotée de 14 entrées/sorties digitales (dont 6 peuvent être utilisée en tant que sortie PWM) ,6entrées analogiques, une horloge a 16 MHZ, une connexion USB, une prise jack d'alimentation, un en-tête et un bouton reset.
- La carte Arduino Uno diffère de toutes les cartes précédentes car elle n'utilise pas le circuit intégré FTDI USB-vers-série. A la place, elle utilise un Atmega8U2 programmé en convertisseur USB-vers-série.
- "Uno" signifie un en Italien et ce nom marque la venue prochaine de la version 1.0 du logiciel Arduino. La carte UNO et la version 1.0 du logiciel seront la référence des versions Arduino à venir. La carte Uno est la dernière d'une série de carte USB Arduino, et le modèle de référence des plateformes Arduino [38]



Figure III. 7:*Arduino UNO*

Il existe deux modèles d'Arduino Uno: l'un avec un microcontrôleur de grande taille, et un autre avec un microcontrôleur dit SMD (SMD: Surface Mounted Device, soit composants montés en surface, en opposition aux composants qui traversent la carte électronique et qui sont soudés du côté opposé). D'un point de vue utilisation, il n'y a pas de différence entre les deux types de microcontrôleurs. [36]

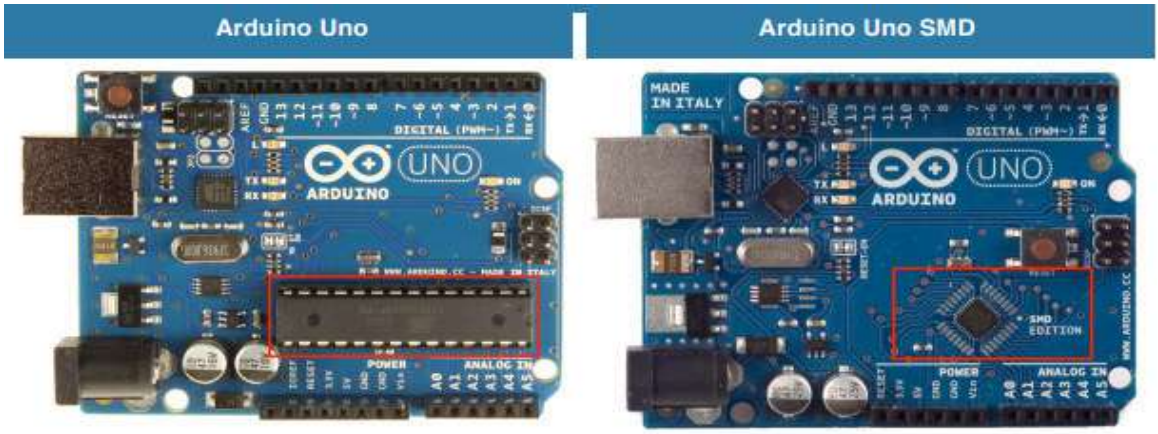


Figure III. 8:*les modèles d'Arduino Uno*

III.3.1 Caractéristiques de la carte Arduino uno :

Tableau III. 1: *caractéristique techniques de la carte Arduino UNO. [38]*

Microcontrôleur	ATmega328P
Tension de fonctionnement	5V
Tension d'entrée (recommandé)	7-12V
Tension d'entrée (limite)	6-20V
E / S numériques Pins	14 (dont 6 fournissent la sortie PWM)

PWM numérique E / S Pins	6
Pins d'entrée analogique	6
DC Courant par I O Pin /	20 mA
Courant DC pour 3.3V Pin	50 mA
Mémoire flash	32 KB (ATmega328P) dont 0,5 KB utilisé par boot loader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Vitesse de l'horloge	16 MHz
Longueur	68,6 mm
Largeur	53,4 mm
Poids	25 g

III.4 Les Accessoires de la carte Arduino :

Il existe de nombreux shields que l'on traduit parfois dans les documentations par «boucliers ». Un «Shield» Arduino est une petite carte qui se connecte sur une carte Arduino pour augmenter ses fonctionnalités. Quelques exemples de « shields » :

- afficheur graphique.
- Ethernet et carte SD.
- GPS.
- carte de prototypage (type labdec).
- . . . etc. [39]



Figure III. 9:*Exemple de Shield*

III.5 Partie expérimentale : Réalisation du projet avec une carte Arduino UNO :

III.5.1 Présentation du projet :

Ce projet a pour but de la conception et la réalisation d'un système de reconnaissance automatique des plaques d'immatriculation à l'aide d'un ensemble des outils matériels et logiciels.

Nous utilisons donc dans ce projet des outils matériels tels que le microcontrôleur Arduino et des autres composants, en plus de ça le logiciel Matlab qui nous permet de faire le traitement d'image de la photo prise par la caméra, ces outils matériels seront ensuite présentés dans ce chapitre.

Lorsque la voiture s'arrête à l'entrée de la barrière, l'infrarouge détecte sa présence donc la LED s'allume, ensuite une image est prise par la camera manuellement par une application java, cette image est traité par suite par l'application Matlab-GUI afin d'extraire le numéro de la plaque. Après avoir le matricule, ce dernier est comparé avec une base de données qui possède plusieurs matricules déjà enregistrés, s'il en existe, on reçoit un message pour ouvrir la barrière, on donne une commande à l'Arduino pour l'ouvrir en appuyant sur le bouton poussoir.

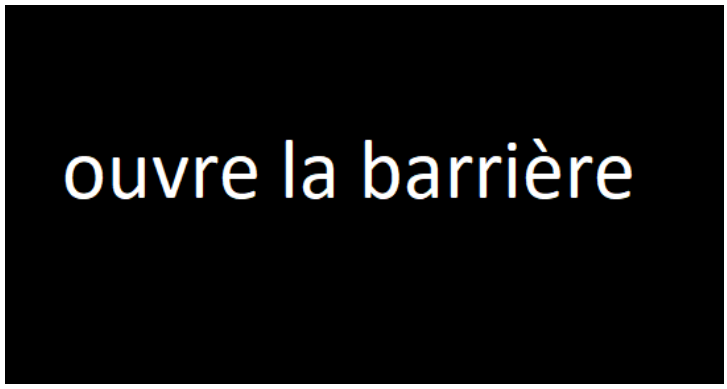


Figure III. 10: *le message pour ouvrir la barrière*

Sinon, on reçoit un message de garder la barrière fermée.

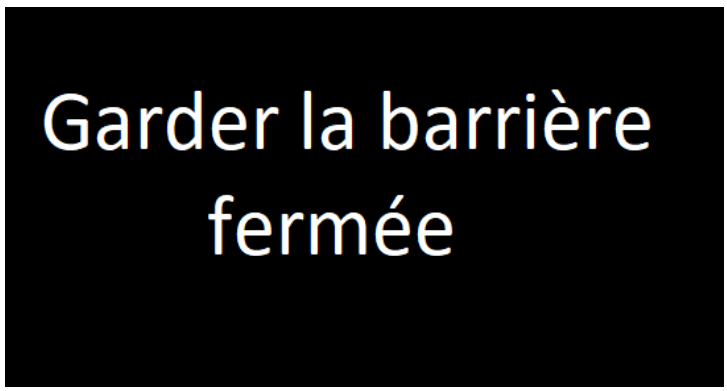


Figure III. 11:*le message pour garder la barrière fermée*

Les approches de contrôle retenues seront analysées et leurs conceptions détaillées, la partie expérimentale sera effectuée et évaluée.

III.5.2 Matériels utilisés

3 éléments essentiel n'apparaitront pas dans les listes et sont indispensables :

Tableau III. 2: *matériels essentiel utilisés.*

Une Arduino UNO	Une Breadboard (plaque d'essai)	Un lot de fils pour brancher le tout
		

Tableau III. 3:*la liste des matériels utilisés.*

Désignation	Quantité	Photo	Description
LED	1		Ce composant est une sorte de lampe un peu spéciale. Nous nous en servirons principalement pour faire de la signalisation. [35]
Résistance (2kOhm)	4		La résistance est un composant de base qui s'oppose au passage du courant. On s'en sert pour limiter des courants maximums mais aussi pour d'autres choses. [35]
Résistance (10kOhm)	2		

Bouton Poussoir	1		Un bouton poussoir sert à faire passer le courant lorsqu'on appuie dessus ou au contraire garder le circuit "éteint" lorsqu'il est relâché. [35]
Caméra (OV7670)	1		Le module caméra OV7670 est un petit capteur d'images, à fonctionnement basse tension, qui offre toutes les fonctions d'une puce Unique caméra VGA. [40]
Capteur infrarouge (C7288)	1		Permettent de détecter la présence d'humains mobiles dans le champ du capteur. ils sont utilisés dans divers systèmes de sécurité et détecteurs de mouvements. [41]
Un servomoteur (SG90)	1		Minuscule et léger avec une puissance de sortie élevée. Servo peut tourner d'environ 180 degrés (90 dans chaque sens), et fonctionne exactement comme les types standard, mais plus petites. [42]

III.5.3 Présentation de l'organigramme du projet :

Avant de passer à l'exécution de notre projet, nous devons réaliser un organigramme qui explique le développement des différentes séquences : le premier organigramme comporte une condition de l'ouverture et la fermeture du servo (barrière), après la détection de présence du véhicule. Et le deuxième représente le traitement des images qui se fait par le PC à l'aide d'une caméra Arduino.

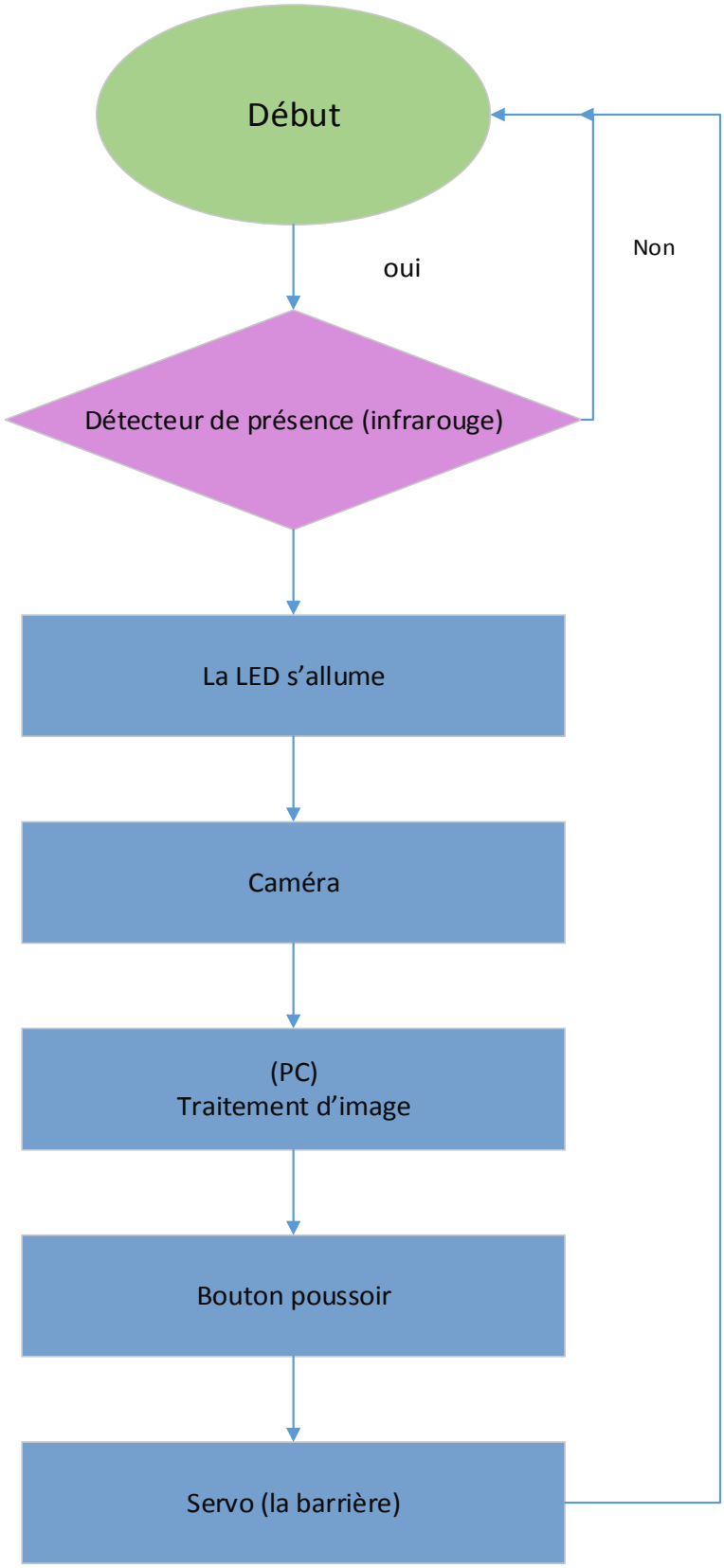


Figure III. 12: *organigramme de parking par Arduino*

III.5.4 Les schémas électroniques :

On a tracé les schémas électriques du système sur le logiciel Fritzing.

Le logiciel Fritzing est un outil de création des projets et des circuits électroniques, il permet aussi l'édition de circuit imprimé, il est disponible gratuitement sur internet. Il a notamment pour vocation de favoriser l'échange de circuits électroniques libres et d'accompagner l'apprentissage de la conception de circuit. [43]

III.5.4.1 Circuit liaison Arduino_ caméra :

La connexion entre l'Arduino et la caméra OV7670 se fait comme suit :

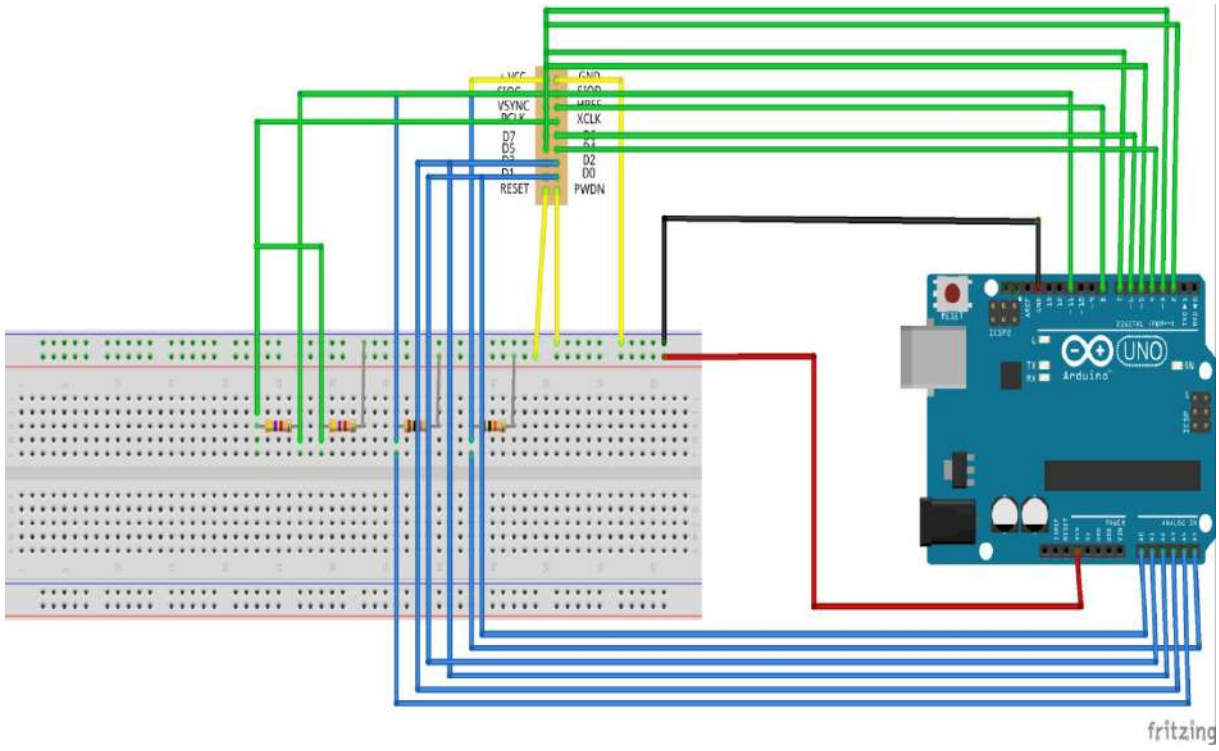


Figure III.13 :*Circuit intégré Arduino_ caméra par fritzing*

On a relié les pins entre l'Arduino et la camera OV7670 comme suit :

Tableau III. 4: *le branchement du camera_Arduino*

Broche OV7670	PINs Arduino
3V3	3V3
GND	GND
SIOC	A5
SIOD	A4
VSYNC	D3
HREF	D8
PCLK	D2
XCLK	D11
D7	D7
D6	D6
D5	D5
D4	D4
D3	A3
D2	A2
D1	A1
D0	A0
RESET	3V3
PWDN	GND

III.5.4.2 Circuit liaison Arduino-infrarouge_ la LED :

La connexion entre l'Arduino et infrarouge et la LED se fait comme suit :

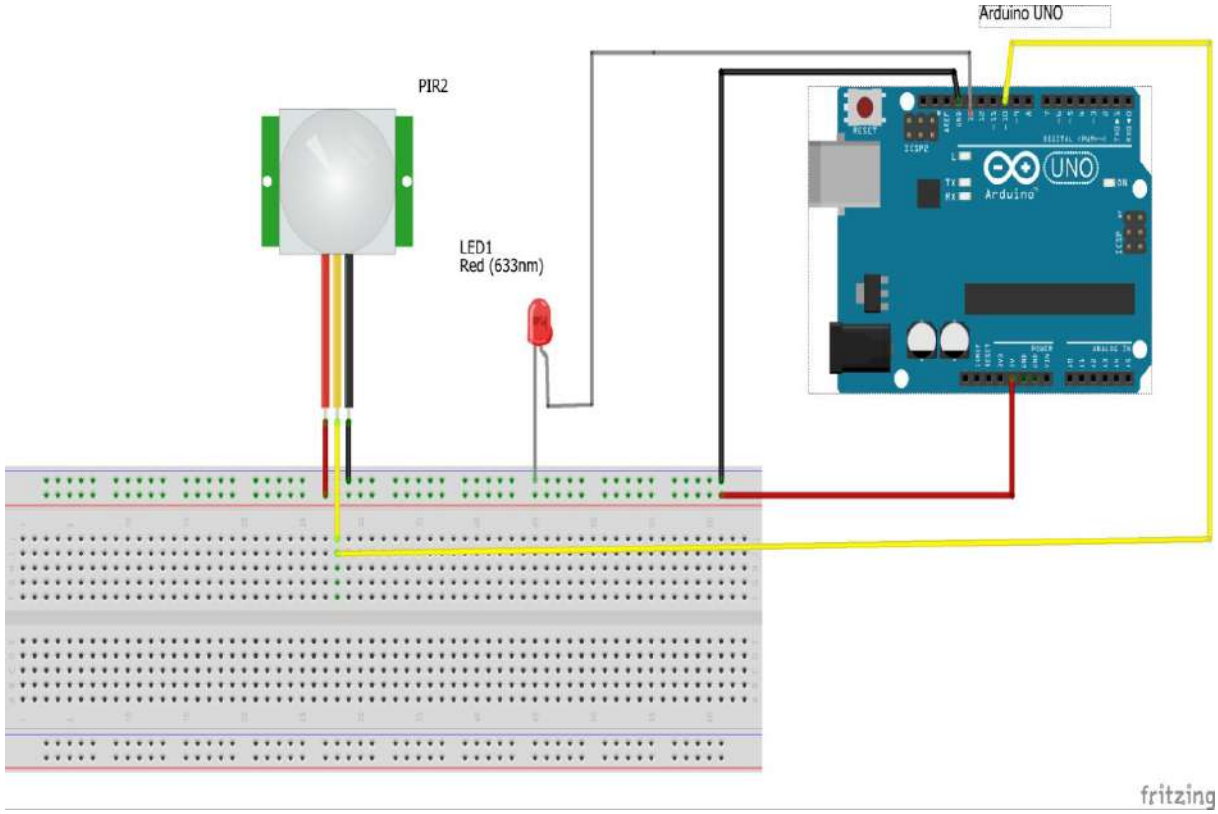


Figure III.14:*Circuit intégré Arduino-infrarouge_ la LED par fritzing*

On a relié les pins entre l'Arduino et la LED et l'infrarouge comme suit :

Tableau III. 5: *le branchement du LED_PIR_Arduino*

Broches LED	**PINs Arduino**
CATHODE	GND
ANODE	D13
Broches PIR	**Arduino**
GND	GND
VCC	5V
DATA	D10

III.5.4.3 Circuit liaison Arduino_moteur_bouton poussoir :

La connexion entre l'Arduino et le moteur et le bouton poussoir se fait comme suit :

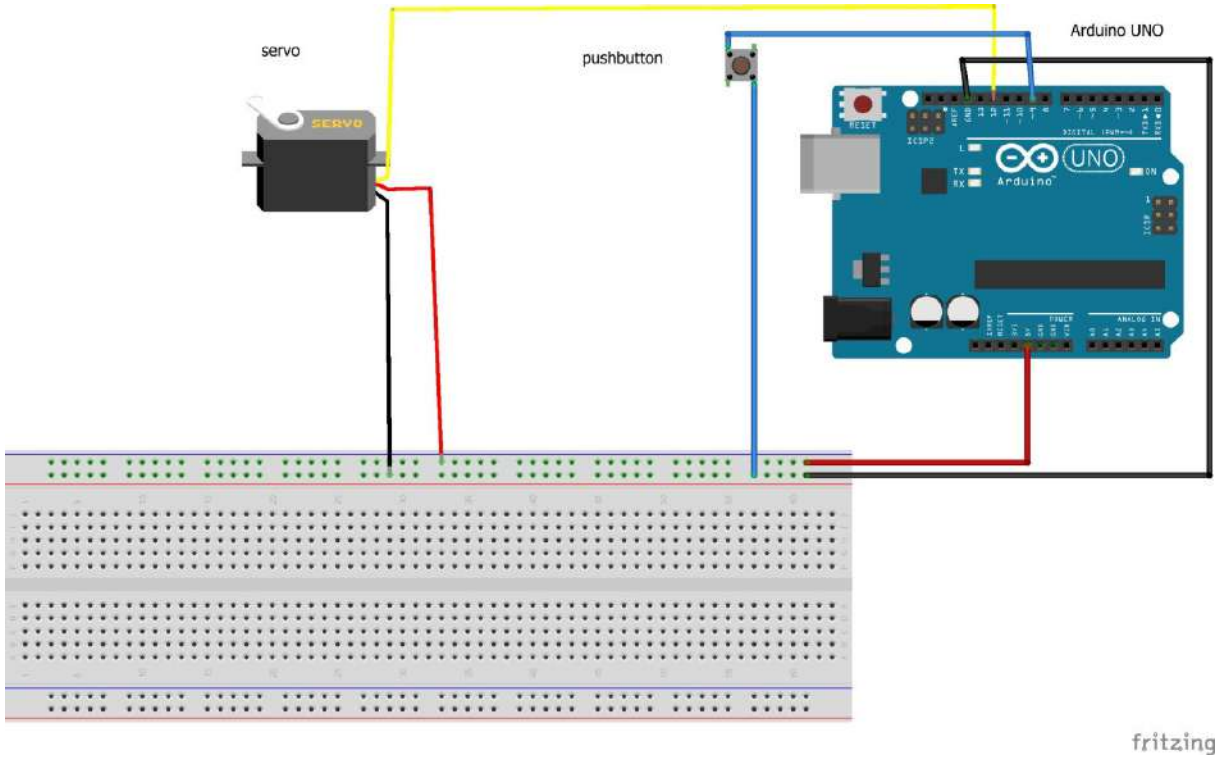


Figure III.15:*Circuit intégré Arduino_moteur_bouton poussoir par fritzing*

On a relié les pins entre l'Arduino et le servo et le bouton poussoir comme suit :

Tableau III. 6: *le branchement du servo_bouton_Arduino*

Broches servo	**PINs Arduino**
GND	GND
VCC	5V
DATA	D12
Broches bouton	**PINs Arduino**
GND	GND
DATA	D9

III.5.4.4 Réalisation du montage final:

La connexion entre l'Arduino et tous les composants pour réaliser notre projet se fait comme suit :

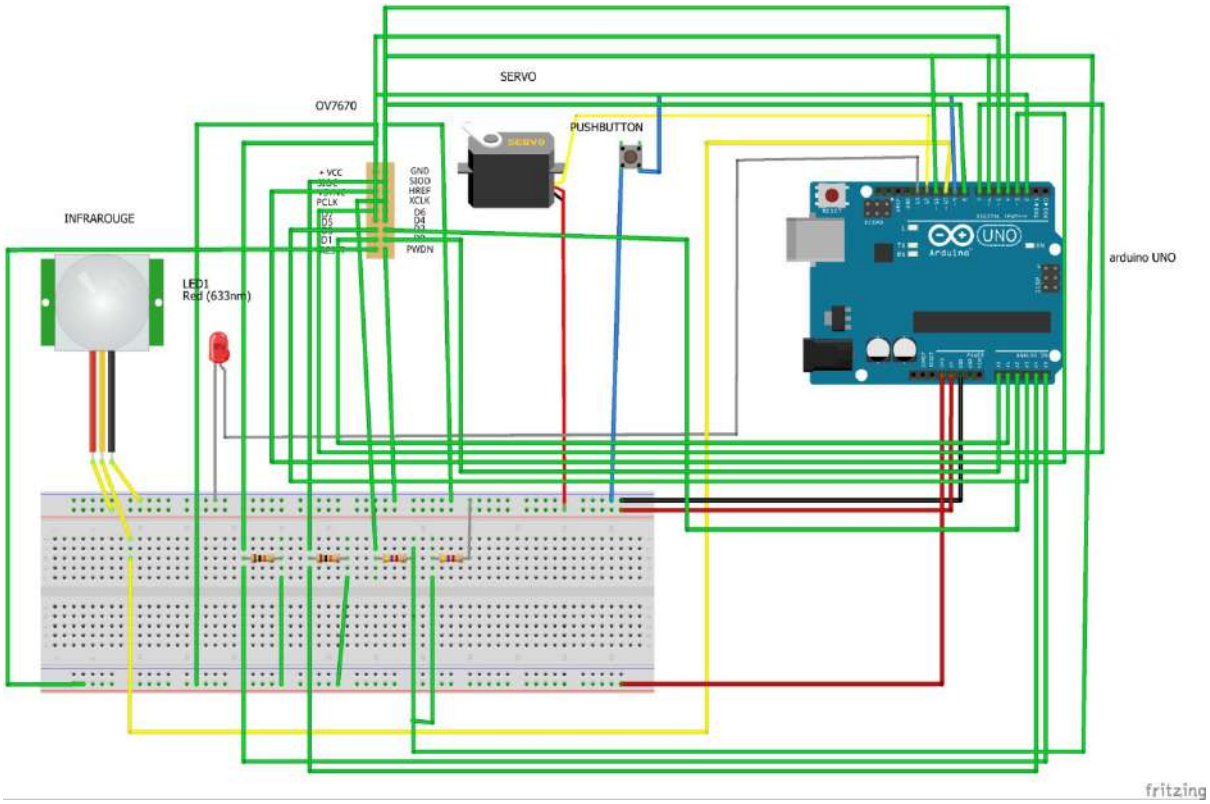


Figure III.16:*le montage final par fritzing*

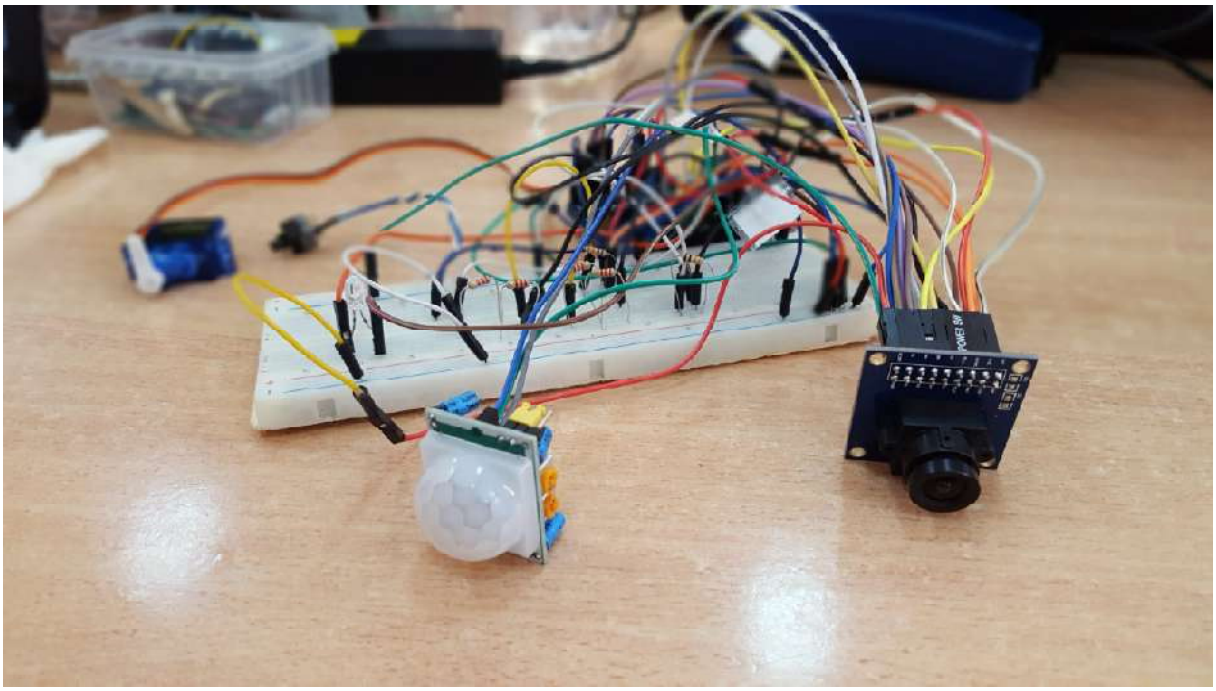


Figure III.17:*le montage final de face*

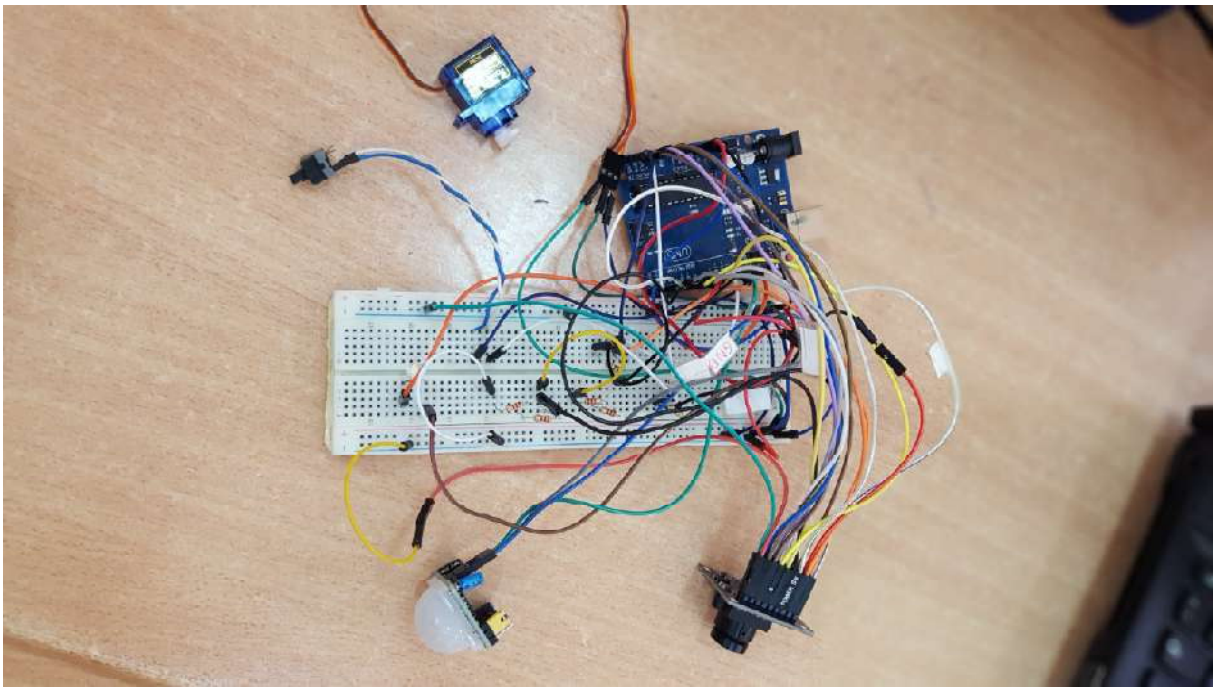


Figure III.18:*le montage final en dessus*

III.6 Résultats et discussion :

Dans ce projet nous créons un système contrôle d'accès qui permet la reconnaissance des plaques d'immatriculation, donc on a fait une combinaison des connaissances entre le 1er et 2ème chapitre. On peut les citer ci-dessous :

- La réalisation du projet à l'aide d'un ensemble d'outils électronique et informatique comme : le PC, Arduino, la caméra…etc. et à l'aide de l'interface Arduino IDE et le logiciel Matlab.
- La création d'une application par Matlab GUI qui nous permet de réaliser notre projet.
- Matlab est également très efficace car il nous a aidé de développer notre projet en faisant le traitement d'image. Et aussi il nous a permis de faire des différentes tâches pour localiser la plaque d'immatriculation et reconnaitre son numéro comme l'extraction, la segmentation et la reconnaissance des chiffres en final.
- La carte électronique Arduino a permis de faire la communication entre les différents outils électronique.
- Dans l'ensemble, notre projet a été réalisé avec succès en faisant les étapes de localisation de plaque et extraire le numéro le numéro, ensuite l'application Matlab_GUI pour compiler nos programmes.
- Un capteur infrarouge a été ajouté afin de recevoir des signaux pour vérifier si la voiture est arrivée à l'entré.
- La caméra OV7670 c'est pour prendre une photo de la voiture et faire le traitement ensuite par le PC pour extraire le numéro.
- Le moteur SG90 c'est pour tourner la barrière après avoir appuyer le bouton poussoir.

On a trouvé quelques difficultés qui peuvent être résolu :

- En raison des variations des formes de polices des chiffres, notre système ne reconnait pas tous les matricules ; on a utilisé une forme de police standard pour les reconnaitre. pour régler ce problème il suffit juste d'ajouter plusieurs bases de données de différents police, le système va comparer le chiffre à reconnaitre avec celui de toutes les bases de donnée jusqu'à ce qu'il trouve une ressemblance avec le chiffre correspondant.
- On a utilisé une caméra de très faible qualité et qui prend beaucoup de temps pour que l'image soit un peu visible, vu qu'on n'a pas les moyens pour ramener une caméra plus professionnelle.
- L'Arduino devient un peu saturé à cause de nombre de PIN.
- La mémoire de l'Arduino UNO est très petite ; on ne peut pas exécuter les deux programmes (caméra et de servo) en même temps ; il faut donc remplacer l'aduino UNO avec un Arduino MEGA.
- La localisation de la plaque de quelques voitures n'était pas complètement réussite, malgré ça, le matricule était correctement extrait.

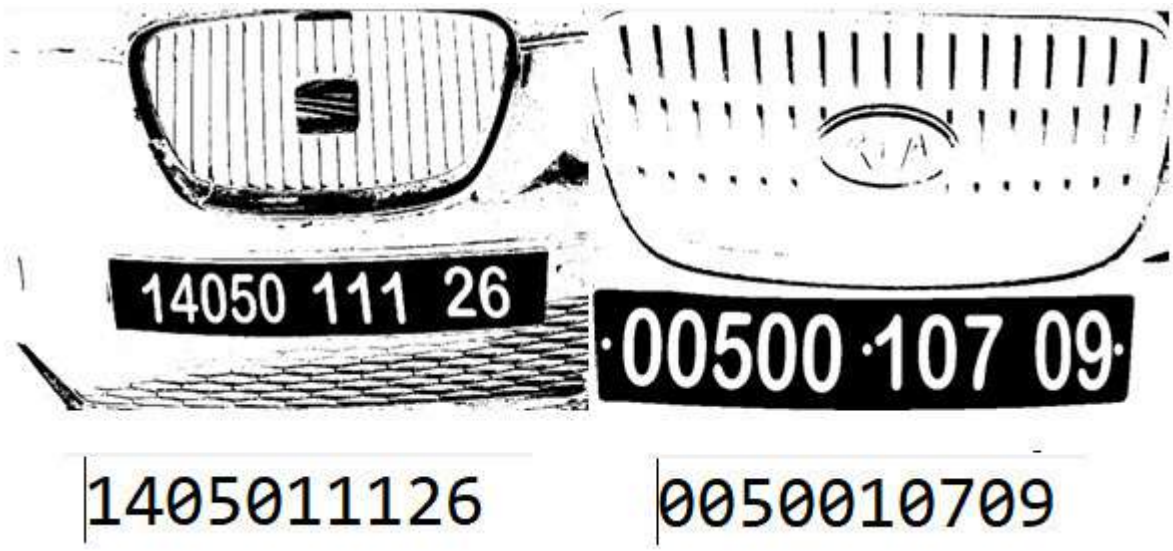


Figure III.19:*difficulté concernant la localisation de la plaque*

A la fin, on peut dire que notre travail est réussi malgré toutes les difficultés et le manque de moyens et de matériels, on a trouvé des solutions qui nous ont aidé à surmonter ces difficultés.

III.7 Conclusion :

Dans ce chapitre nous avons brièvement détaillé et réalisé notre projet concernant la reconnaissance de la plaque d'immatriculation par l'outil de commande « Arduino UNO » qui nous a donné la possibilité d'allier les performances de la programmation à celles de l'électronique. Ainsi que le langage de programmation « Matlab » qui est utilisé dans tous les domaines et qui nous a facilité aussi le traitement de l'image.

Plus précisément, le gros avantage que nous l'avons atteint c'est de programmer les systèmes électroniques et réaliser un projet simplifié de sorte qu'il facilite et effectué des taches dans des environnements de travail qui nécessite un projet pareil.

Conclusion générale :

Conclusion générale :

Le travail que nous avons élaboré s'inscrit dans le cadre de projet de fin d'études universitaire, nous avons essayé de réaliser un moyen robuste permettant la reconnaissance automatique des plaques d'immatriculation en pratiquant les connaissances acquises durant nos études universitaire.

On a pris le plaisir de travailler sur ce thème, qui peut aider et améliorer le domaine des automobiles en facilitant l'accès des voitures dans les différents endroits en toute sécurité.

Notre projet était en relation avec plusieurs domaines comme l'automatique, l'électronique, l'informatique qui nous a permis de réaliser ce mémoire.

Dans le 1er chapitre, on a parlé sur la reconnaissance de formes d'une façon générale, ensuite la reconnaissance des plaques d'immatriculation en particulier comme le champ de vision de la caméra ,les difficultés et les étapes de la reconnaissance ainsi les applications de ce système en indiquant les différents endroits à utilisé et qui est basé sur un grand point qui est le traitement d'image.

Dans le deuxième chapitre et comme une première partie on a parlé sur le fonctionnement, les contraintes de ce système ensuite sur le logiciel MATLAB son but et ses avantages, et pour terminer cette partie on a fait la conception détaillé sur le système de reconnaissance de plaque d'immatriculation. Comme une deuxième partie on a réalisé l'application « Matlab-GUI » afin que toutes les opérations de traitement et de contrôle de la reconnaissance soient effectuées avec une simple clique.

Dans le dernier chapitre on a fait la réalisation pratique du projet , en parlant sur la carte électronique « Arduino » qu'on a utilisé dans la pratique ,ensuite la réalisation du montage contenant la carte Arduino UNO, un moteur pour l'ouverture et la fermeture du barrière, l'infrarouge et une caméra pour capturer la plaque d'immatriculation et un PC qui nous a aidé de faire le traitement d'image afin d'extraire le numéro de la plaque et vérifier les matricules qui existe dans notre base de donné afin d'ouvrir le barrière .

Ce mémoire nous a permis la combinaison entre les différents domaines qui facilite la mise en œuvre, la création et la réalisation d'un système destiné pour faciliter la vie.

Bibliographie :

[1] : Benali Moustafa : Reconnaissance Automatique des Chiffres Manuscrits : Mémoire de fin d'études Pour l'obtention du diplôme de Master en Informatique. Option : Modèle Intelligence et Décision (M.I. Université Abou Bakr Belkaid– Tlemcen .Faculté des Sciences : Département d'Informatique.2016 – 2017) :53p.

[2]: Henry. Maître. " Le traitement des images ": Hermes Lavoisier IC2. 2003 : 364p. France.

[3] : NOR Imane. SIDHOUM Souad : Développement d'une application de détection et de reconnaissance de plaques d'immatriculation(LAPIA) : Mémoire de fin d'études Pour l'obtention du diplôme de master en Informatique : Université Abou Bakr Belkaid– Tlemcen Faculté des Sciences : Présenté le 02 Juillet 2017 : 63p.

[4] : Amouche Yacine : développement d'une interface pour la segmentation d'image basée sur les coupes normalisées : mémoire de fin de cycle en vu de l'obtention du diplôme de master professionnel en informatique : option Administration et sécurité des réseaux : 2013/2014 : université Abderrahmane Mira-Bejaia : p21

[5] : Melle ACHAT Thinhinane : Melle BENHAMAOUCHE Fatima : Segmentation d'image par classifieurs flous: Application à l'Imagerie par Résonance Magnétique (IRM) : Projet de fin d'études En vue de l'obtention du diplôme Master Filière : Electronique Option : Télécommunication: Université Abderrahmane MIRA – Bejaia Faculté de Technologie Département de Génie électrique : Promotion 2015 :39p.

[6] : Melle Aissou Nabila Mr. Melle Benkaci Amira : Méthode coopérative pour la segmentation d'image : Application a l'imagerie médicale : en vue de l'obtention du diplôme Master : filière électronique option télécommunication : promotion n 2016 :p48.

[7]: MEDJAOUI Amina et FARES Fadia : Segmentation des Images par Contours Actifs : Application sur les Images Satellitaires à Haute Résolutions : Mémoire de fin d'études Pour l'obtention du diplôme de Master en Informatique : Université Abou Bakr Belkaid– Tlemcen : Soutenu le 02 Juillet 2012: p59.

[8] : J.P.Gastellu-Etchegorry, "acquisition et traitement d'image numérique", IUTD département de mesures physique : Université Paul Sabatier : Avril 2008 : France (Toulouse) :150p.

[9] : Girard Michel Claude, Girard Colette, Traitement de données de télédétection, 1999

[10]: Marechal Menrt : Phoenix Publication un sujet qui Mérite Débat la ville Américaine : 2002 :53p.

[11] : ELOUARIACHI Ilham : extraction des descripteurs en utilisant la théorie des moments : application a la reconnaissance des plaques d'immatriculations des véhicules : Projet de fin d'étude : master science et technologies : Systèmes intelligents et réseaux : 13 Juin 2016 : Maroc: 79p.

[12] : Ameur CHHAYDER et Imene BELHADJ MOHAMED : Système de Reconnaissance Automatique des Plaques Minéralogiques. 5th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications. March 22-26, 2009 – TUNISIA.

[13] : Auteur Jiménez Zozaya, Lourdes : Conseiller Malanda Trigueros, Armando Lazarova, Milena : Automatique number plate recognition : Université publique de Navarre : Département de génie électrique et électronique : Date 2013.

[14] : Zerougui Aissam et Sari Nabil : traitement d'image monochromes détection de contour, filtrage (spatial et fréquentiel) et segmentation par réseaux de neurones: mémoire de fin d'étude en vue de l'obtention du diplôme de master informatique industrielle: université Larbi Ben M'hidi, Oum El Bouaghi: juin 2017:p72

[15]: V. DROOGENBROECK : « Traitement d'image numériques au moyen d'algorithmes en utilisant la morphologie mathématique et la notion d'objet : application au codage » : thèse de docteur de l'école nationale supérieure des mines de Paris : centre de morphologie mathématique : 1994.

[16] : BENMOHAMED Abderrahim : Une Approche semi-automatique pour l'indexation de documents anciens : MEMOIRE Présenté en vue de l'obtention du diplôme de MAGISTER : UNIVERSITE BADJI MOKHTAR - ANNABA : Année 2009 : p102.

[17] : Wassima Ait Fares : Détection et suivi d'objets par vision fondés sur segmentation par contour actif basé région : THÈSE En vue de l'obtention du DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE : Délivré par l'Université Toulouse III - Paul Sabatier Discipline ou spécialité : Systèmes Embarqués. Automatique / Robotique. : présenté le 27/09/2013 : France.

[18] : B. Mohammed : "Construction d'une technique d'aide au diagnostic en imagerie médicale application à la compression d'images : " Thèse de Doctorat : Université de Tlemcen : 2012.

[19]: B. U. Shanker: "Novel Classification and segmentation techniques with application to remotely sensed images: " Transactions on Rough Sets 7, Springer Belin, Heidelberg Edition: 2007: pp. 295-380.

[20]: Mohammad Zakariya Siam: International Journal of Engineering and Innovative Technology (IJEIT) Volume 4: Issue 4.Practical Design of an Automatic License Plate

Recognition Using Image Processing Technique: Electrical Engineering Department. ISRA : Université. Amman-Jordan. October 2014 :179p.

[21]: Rédacteur : Ludovic Simon. CETE Ile-de-France : Rapport d'études Systèmes et équipements de lecture automatique de plaques d'immatriculation des véhicules Principes et aperçu des applications : juin 2013.

[22] : le soir d'Algérie : Disponible sur : www.lesoirdalgerie.com publié par LSA le 15.08.2018.

[23]: Narendra Singh Tomar1, Prakhar Sachan2, Pranav Mittal3, And Shivani Agarwal4: VEHICLE NUMBER PLATE DETECTION USING MATLAB: International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 05 Issues: 05 May-2018: 2839p.

[24]: Ragini Bhat1, Bijender Mehandia: RECOGNITION OF VEHICLE NUMBER PLATE USING MATLAB: INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING Vol. 2: Issue: August 2014: India: 1903p.

[25]: Nuzulha Khilwani Ibrahim, Emaliana Kasmuri, Norazira A Jalil, Mohd Adili Norasikin, Sazilah Salam, Mohamad Riduwan Md Nawawi: License Plate Recognition (LPR): A Review with Experiments for Malaysia Case Study: The International Journal of Soft Computing and Software Engineering [JSCSE]: Vol. 3, No. 3: Special Issue: The Proceeding of International Conference on Soft Computing and Software Engineering 2013 [SCSE'13]: San Francisco State University: CA, U.S.A: March 2013: 93p.

[26]: Wong Eng Yong: Supervisor Name Mr. Daniel Chua Peng Huat: Vehicle License Plate Registration Recognition System: Sim University: la chine: Jul07: 110p.

[27]: Bhawna Tiwari1, Archana Sharma2, Malti Gautam Singh3, Bhawana Rathi:

Automatic Vehicle Number Plate Recognition System using Matlab: *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 11, Issue 4, Ver. II (Jul.-Aug .2016).*

[28]: Ms. Shilpi Chauhan and Vishal Srivastava: Matlab Based Vehicle Number Plate Recognition: Research Scholar, Dept. of CSE, Arya Insitute of Engineering and IT:Associate Professor, Dept. of CSE, Arya College of Engg. And IT: Jaipur, India: Number 9 (2017): 2288p.

[29]: Samarth Borker: Implementation of Vehicle License Plate Recognition Using Canny Edge Detection: Dept of Electronics & Telecommunication: Goa College of Engineering, Ponda-Goa: India: 06 Jun 2015: 379p.

[30]: Naikur Bharatkumar Gohil: CAR LICENSE PLATE DETECTION: B.E. Dharmsinh Desai University: PROJECT Submitted in partial satisfaction of the requirements for the degree of MASTER OF SCIENCE: India: 2006: 60p.

[31]: EL OUAARABI Mohamed DAHI Zouhir : SYSTEME DE RECONNAISSANCE AUTOMATIQUE DES PLAQUES D'IMMATRICULATION MAROCAINE : Université Mohammed V Ecole Normale Supérieure de l'Enseignement Technique Rabat : Master GE : Maroc : 27p.

[32]: B. Hongliang ET L. Changping: « A hybrid license plate extraction method based on edge statistics and morphology »: in Proceedings of the 17th International Conference on Pattern Recognition: ICPR 2004, vol. 2: p 831-834.

[33]: Rana Gill, Navneet Kaur: Indian Vehicle Number Plate Detection by Image Processing Technique in MATLAB: International Journal of Computer Science And Technology IJCST Vol. 3: Issue 2: April - June 2012: India: 16p.

[34]: Munuo, Cosmo: Vehicle Plate Number Detection and Recognition Using Improved Algorithm: 2014.

[35] Astalaseven, Eskimon et olyte : Arduino pour bien commencer en électronique et en programmation : Licence Creative Commons BY-NC-SA 2.0 : Dernière mise à jour le 4/08/2012 : p 294.

[36] Frédéric Genevey : Arduino à l'école : Edition 2016 :p44.

[37] Jean- Noël Montagné : « livret Arduino en français »: centre de ressources art sensitif: novembre 2006 :p35.

[38] Troudi Khaled: Conception et réalisation d'un système Smart House: MEMOIRE DE MASTER: Sciences et Technologies Automatique et informatique industrielle: Université Mohamed Khider de Biskra Faculté des Sciences et de la Technologie : Département de génie électrique: p65

[39] présenté par Abdelaziz DJERIOUAT et Leila Hanane KLOUCHE DJEDID : Intitulé du mémoire Mise en œuvre de cartes de commande basées sur le module de prototypage Arduino pour moteurs moteurs pas à pas : MEMOIRE DE FIN D'ETUDE POUR L'OBTENTION DU DIPLOME DE MASTER ACADEMIQUE : le 17 Juin 2015 : p61.

[40]:http://www.clubic.com/telecharger-fiche378430-matlab.html (consulté le 12/09/2017).

[41]: http://nagashur.com/blog/2015/08/14/capteur-pir-et-arduino-tutoriel/

[42] : Nezzar Ikram : Conception et réalisation d'un robot quadrupède : MEMOIRE DE FIN D'ETUDES en vue de l'obtention du DIPLOME DE MASTER : Mémoire de fin d'étude soutenu publiquement à OUM EL BOUAGHI : Le 05/06/2016 : p42.

[43]: Mekhalfia Toufik et Ghadbane Toufik : étude et réalisation d'un système de commande à distance des installations électriques pour la domotique : mémoire présenté pour l'obtention du diplôme de Master : université Mohamed Boudiaf M'sila : p63.
[44]: https://www.developpez.net: 23/04/2008:08h28.

Annex :

- **Le programme général de la reconnaissance de formes :**

```
clc;
clear all;
f=imread('003.jpg'); %  lecture de l'image
figure(1),imshow(f), title('image originale');
g= rgb2gray(f);  % convertir l'image en niveau de gris
figure(2) , imshow(g), title('image en niveau de gris');
i=imadjust(g);   % permet de modifier l'étalement des niveaux de gris
figure(3) , imshow(i), title('modification de l etalement de nv de gris');
i2 = edge(i,'canny',0.4);  % filtre de canny
figure(4),imshow(i2);title('filtre de canny');
se = strel('square',2); % Un objet strel représente un élément structurant
%  morphologique plat, élément essentiel des opérations de dilatation et
d'érosion morphologiques.
i3 = imdilate(i2,se);  % dilatation
figure(5),imshow(i3),title('dilatation');
i4 = imfill(i3,'holes'); % remplit les trous dans l'image binaire d'entrée
figure(7),imshow(i2),title('combler les trous');
[Ilabel num] = bwlabel(i4);
% la focntion bwlabel te permet de passer d'une image binaire
% avec des regions differentes (genre apres une segmentation)
% vers une image ou chaque label (ou region) de ton image aura un label(un
numero) different.
BW2 = bwareaopen(i4,9000); % Supprimer les petits objets de l'image binaire
figure(8),imshow(BW2),title('enlever les petits objets');
d = bwmorph(BW2,'remove'); % applique une opération morphologique
spécifique à l'image binaire
d =medfilt2(BW2,[3 3]);  %filtre median
figure(9),imshow(d),title('filtre median');

[m,n]=size(d);
iv=1;jv=1;
for i=1:m
    for j=1:n
        if d(i,j)==1
            vi(iv)=i;
            vj(jv)=j;
            iv=iv+1;
            jv=jv+1;
        end
    end
end
dpi=min(vi);
fpi=max(vi);
dpj=min(vj);
fpj=max(vj);
fp=f(dpi:fpi,dpj:fpj);
figure(10),imshow(fp);
B=~im2bw(fp);
figure(11),imshow(B);title('la plaque a reconnetre');
% segmenttion----------------------------------

picture2=bwareaopen(B,200); % i reste que le texte c-a-d enlever les petits
objets
figure(14),imshow(picture2);title('segmentation');

[L,Ne]=bwlabel(picture2);
% Sélectionnez les composants connectés dans une image binaire 2D cad
connecte les composants un par un
```

```
% l donne la matrice qui a l'information sur la plaque d'immatriculation
% et Ne donne le nombre de chiffres ou de caractères
propied=regionprops(L,'BoundingBox'); % Mesurer les propriétés des régions
d'image
hold on
pause(1)
for n=1:size(propied,1)

rectangle('Position',propied(n).BoundingBox,'EdgeColor','g','LineWidth',2)
%   dessiner un rectangle
end
hold off
% comparaison---------------------------------
figure
final_output=[];
t=[];
for n=1:Ne
  [r,c] = find(L==n);
  n1=B(min(r):max(r),min(c):max(c));
  n1=imresize(n1,[42,24]);
%    la taille de la base de données est 42,24 donc il est redimensionné
%    afin que nous puissions le faire correspondre à la base de données
  figure(15), imshow(n1);
  pause(0.2)
  x=[ ];
load('imgfildata.mat')
totalLetters=size(imgfile,2);

 for k=1:totalLetters

    y=corr2(imgfile{1,k},n1);
    x=[x y];

 end
 t=[t max(x)];
 if max(x)>.45
 z=find(x==max(x));
 out=cell2mat(imgfile(2,z));

final_output=[final_output out];
end
end

file = fopen('number_Plate.txt', 'wt');
    fprintf(file,'%s\n',final_output);
    fclose(file);
    winopen('number_Plate.txt')
```

- **Le programme de l'application MATLAB-GUI:**

```
function varargout = GUIINV(varargin)
% GUIINV MATLAB code for GUIINV.fig
%      GUIINV, by itself, creates a new GUIINV or raises the existing
%      singleton*.
%
%      H = GUIINV returns the handle to a new GUIINV or the handle to
%      the existing singleton*.
%
%      GUIINV('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in GUIINV.M with the given input arguments.
%
%      GUIINV('Property','Value',...) creates a new GUIINV or raises the
```

```
%      existing singleton*.  Starting from the left, property value pairs
are
%      applied to the GUI before GUIINV_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property
application
%      stop.  All inputs are passed to GUIINV_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help GUIINV

% Last Modified by GUIDE v2.5 12-Jun-2019 10:36:04

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @GUIINV_OpeningFcn, ...
                   'gui_OutputFcn',  @GUIINV_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before GUIINV is made visible.
function GUIINV_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to GUIINV (see VARARGIN)

% Choose default command line output for GUIINV
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes GUIINV wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = GUIINV_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```
% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
im= imgetfile
im=imread(im);
axes(handles.axes1);
imshow(im);
title('IMAGE ORIGINEL','Color','R')
save w1 im


% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
load w1;
g= rgb2gray(im);
axes(handles.axes1);
imshow(g);
title('IMAGE ORIGINEL','Color','R')
save w2 g


% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
load w2;
i=imadjust(g);   % permet de modifier l'étalement des niveaux de gris
i2 = edge(i,'canny',0.4);  % filtre de canny
se = strel('square',2); % Un objet strel représente un élément structurant
%  morphologique plat, élément essentiel des opérations de dilatation et
d'érosion morphologiques.
i3 = imdilate(i2,se);  % dilatation
i4 = imfill(i3,'holes'); % remplit les trous dans l'image binaire d'entrée
[Ilabel num] = bwlabel(i4);
% la focntion bwlabel te permet de passer d'une image binaire
% avec des regions differentes (genre apres une segmentation)
% vers une image ou chaque label (ou region) de ton image aura un label(un
numero) different.
BW2 = bwareaopen(i4,9000); % Supprimer les petits objets de l'image binaire
d = bwmorph(BW2,'remove'); % applique une opération morphologique
spécifique à l'image binaire
d =medfilt2(BW2,[3 3]);  %filtre median
axes(handles.axes1);
imshow(d)
title('LA PLAQUE LOCALISE','Color','R')
save w3 d

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```
load w3
load w2
[m,n]=size(d);
iv=1;jv=1;
for i=1:m
    for j=1:n
        if d(i,j)==1
            vi(iv)=i;
            vj(jv)=j;
            iv=iv+1;
            jv=jv+1;
        end
    end
end
dpi=min(vi);
fpi=max(vi);
dpj=min(vj);
fpj=max(vj);
B=g(dpi:fpi,dpj:fpj);
% B= ~im2bw(B);
axes(handles.axes2);
imshow(B),
title('LA PLAQUE A RECONAITRE','Color','R')
save w4 B

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
load w4
B= ~im2bw(B);
picture2=bwareaopen(B,200); % il reste que le texte c-a-d enlever les
petits objets


[L,Ne]=bwlabel(picture2);
% Sélectionnez les composants connectés dans une image binaire 2D cad
connecte les composants un par un
% l donne la matrice qui a l'information sur la plaque d'immatriculation
% et Ne donne le nombre de chiffres ou de caractères
propied=regionprops(L,'BoundingBox'); % Mesurer les propriétés des régions
d'image
hold on
pause(1)
for n=1:size(propied,1)

rectangle('Position',propied(n).BoundingBox,'EdgeColor','g','LineWidth',2)
%   dessiner un rectangle
end
hold off
% comparaison-----------------------------------

final_output=[];
t=[];
for n=1:Ne
  [r,c] = find(L==n);
  n1=B(min(r):max(r),min(c):max(c));
%   r et c sont les dimensions du rectangle
  n1=imresize(n1,[42,24]);
%    la taille de la base de données est 42,24 donc il est redimensionné
%    afin que nous puissions le faire correspondre à la base de données
```

```
  pause(0.2)
  x=[ ];
load('imgfildata.mat')
totalLetters=size(imgfile,2);

 for k=1:totalLetters

    y=corr2(imgfile{1,k},n1); % on compare la base de données avec le
rectange
    x=[x y];

 end
 t=[t max(x)];
 if max(x)>.45
 z=find(x==max(x));
 out=cell2mat(imgfile(2,z));
%  convertit un tableau de cellules en un tableau ordinaire. Les éléments
du tableau
% de cellules doivent tous contenir le même type de données, et le tableau
résultant
% est de ce type de données.

final_output=[final_output out]
end
end
set (handles.resultat, 'string',final_output)
save w5 final_output


function resultat_Callback(hObject, eventdata, handles)
% hObject    handle to resultat (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of resultat as text
%        str2double(get(hObject,'String')) returns contents of resultat as
a double


% --- Executes during object creation, after setting all properties.
function resultat_CreateFcn(hObject, eventdata, handles)
% hObject    handle to resultat (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close()
close()
```

```
% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton7 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
load w5
mat=final_output
mat1= '0238911626';
mat2= '1196411126';
mat3= '0303911635';
if (mat==mat1)
    k = imread('ouvre.PNG');
    figure(17) , imshow(k);
elseif (mat==mat2)
      k=imread('ouvre.PNG');
    figure(17) , imshow(k);
elseif (mat== mat3)
       k =imread('ouvre.PNG');
    figure(17) , imshow(k);
else
    k=imread('ferme.PNG');
    figure(17) , imshow(k);
 end
```

- **Programme de l'ouverture de la fermeture de barrière (servo) :**

```
#include<Servo.h>

 int bouton = 9;
 int alert_led = 13;
 int mouvement_cap = 10;
 int val_mouvement = 0;
 int val_bouton;
 Servo barriere;
void setup() {
  // put your setup code here, to run once:
pinMode(bouton,INPUT_PULLUP);
pinMode(mouvement_cap,INPUT);
pinMode(alert_led,OUTPUT);
barriere.attach(12); //servo pin
barriere.write(0); //servo start position
Serial.begin(9600);
}
void loop() {
  // put your main code here, to run repeatedly:
  val_mouvement=digitalRead(mouvement_cap);
  if (val_mouvement==HIGH)
  {
    digitalWrite(alert_led,HIGH);
  }
```

```
 else {
   digitalWrite(alert_led,LOW);
 }
val_bouton=digitalRead(bouton);
 if (val_bouton == 0)
 {
     barriere.write(100);
     delay(4000);
     barriere.write(0);
 }
}
```

- **L’application du camera :**

