

UNIVERSITE LIBRE DES PAYS DES GRANS LACS

ULPGL

FACULTE DES SCIENCES ET DES TECHNOLOGIES



BP 368 GOMA

Département de Génie Informatique

**CONCEPTION D'UN SYSTEME D'IDENTIFICATION
DES PLAQUES D'IMMATRICULATIONS BASÉ SUR
OPEN CV**

Présenté par **KAYIHEMBAKO MWENGE Edwige**

Mémoire présenté en vue de l'obtention du Diplôme de
bachelor en Sciences de l'ingénieur.

Option : **Génie Informatique**

Directeur : **Prof. DI. Dr. Tech. AKWIR NKIEDIEL Alain**

Encadreur : **MSc. Ing. NZANZU VINGI Patrick**

Mars 2024

EPIGRAPHE

Avec Dieu, nous ferons des exploits : c'est lui qui écrasera nos adversaires.

Psaumes 60 : 12

DEDICACES

A mes parents :

PALUKU MALIRO Edmond

KAHINDO KASONGYA Suzanne

KAYIHEMBAKO MWENGE Edwige

REMERCIEMENTS

Au terme de notre cycle de formation universitaire en département de Génie Informatique, il serait ingrat de notre part de ne pas exprimer notre profonde gratitude et d'adresser nos remerciements à tous ceux qui ont contribué de diverses manières à la réalisation de ce travail. En premier lieu, que la gloire soit infiniment rendue à notre Dieu Tout-Puissant qui, par son amour grand, nous a donné : vie, santé, courage pour la réalisation du présent travail.

Nos remerciements s'adressent aux autorités académiques ainsi qu'au corps professoral de l'Université Libre des Pays des Grands Lacs (ULPGL/GOMA), pour l'encadrement scientifique qu'ils nous ont assuré.

Au Professeur Prof. DI. Dr. Tech. AKWIR NKIEDIEL Alain et au MSc. Ing. NZANZU VINGI Patrick qui nous ont dirigé dans la réalisation de ce travail et qui ont mis tout leur savoir-faire à notre profit, nous disons merci.

Mes remerciements s'adressent à mon père PALUKU MALIRO Edmond et à ma mère KAHINDO KASONGYA Suzanne ainsi qu'à nos frères et sœurs pour tous les efforts consentis pour qu'il ne nous manque de rien pour la réalisation de ce travail.

A tous mes camarades avec qui, nous avons vécu et partagé les joies et les difficultés.

A tous ceux qui ont concouru de près ou de loin pour la réalisation de ce travail, que le Tout-Puissant vous comble de ses bénédictions.

KAYIHEMBAKO MWENGE Edwige

SIGLES ET ABREVIATIONS

ALPR : Automatic License Plate Recognition

CE : Clé Etrangère

CP : Clé Primaire

CV : Computer Vision

DCU : Diagramme des Cas d'Utilisation

FARDC : Forces Armées de la République Démocratique du Congo

IA : Intelligence Artificielle

IAPI : Identification Automatique des Plaques d'Immatriculation

MONUSCO : Mission de l'Organisation des Nations Unies Pour la Stabilité au Congo

MySQL : Système de gestion de bases de données relationnelles

OCR : Reconnaissance Optique des Caractères

RVB : rouge, vert et bleu.

Liste des Tableaux

Tableau1. 1: Avantages et inconvénients de quelques méthodes de détection de la plaque	13
Tableau1. 2: DCU-Acquisition des images	20
Tableau1. 3: DCU-Annotation des images.....	20
Tableau1. 4: DCU-Détection de la plaque	21
Tableau1. 5: DCU-Extraction des caractères de la plaque	21
Tableau1. 6 : DCU-Enregistrement des données de la plaque	21
Tableau1. 7 : DCU-Vérification de la plaque.....	21
Tableau1. 8: DCU-Identification de la plaque.....	22
Tableau3. 1: Exemple des plaques identifiées avec succès	34
Tableau3. 2 : exemple des plaques non identifiées car erronée.....	35

Liste des Figure

Figure 1. 1: Composition d'un système IAPI	5
Figure 1. 2: Voiture en control d'accès	6
Figure 1. 3: Voiture au poste frontière	6
Figure 1. 4: véhicules équipés de caméras IAPI	7
Figure 1. 5: Voiture dépassant la limite de vitesse	7
Figure 1. 6: plaque d'immatriculation type AAA 111	8
Figure 1. 7: plaque d'immatriculation type AA111AA	8
Figure 1. 8: plaque d'immatriculation type 11AA111	8
Figure 1. 9: Plaque d'immatriculation congolaise pour les particuliers	9
Figure 1. 10: Corps diplomatique.	9
Figure 1. 11: Forces armées.	10
Figure 1. 12: Nations unies.....	10
Figure 1. 13: Présidence de la République.	10
Figure 1. 14 : Agent de l'état.....	Erreur ! Signet non défini.
Figure2. 1 :Principales fonctions d'un système IAPI [3]	14
Figure2. 2: Diagramme des cas d'utilisation d'un système IAPI	20
Figure3. 1: image originale	30
Figure3. 2: image en niveau de gris	31
Figure3. 3: image binaire	31
Figure3. 4: exemple des caractères extraits et segmentés de la plaque	33

Sommaire

EPIGRAPHE	i
DEDICACES.....	ii
REMERCIEMENTS.....	iii
SIGLES ET ABREVIATIONS	iv
Liste des Tableaux.....	v
Liste des Figure	vi
Sommaire	vii
RESUME.....	viii
ABSTRACT.....	ix
INTRODUCTION GENERALE	1
CHAPITRE I. GENERALITES SUR L'IDENTIFICATION DES PLAQUES D'IMMATRICULATIONS	4
I.1. Définition	4
I.2. Domaine d'application d'identification des plaques d'immatriculations.....	5
I.3. Les différentes formes d'immatriculation dans le monde.....	8
I.4. Caractéristiques des plaques d'immatriculation Congolaise.....	9
I.5. Problèmes des systèmes IAPI	10
I.6. Conclusion	11
CHAPITRE II . METHODOLOGIE ET CONCEPTION DU SYSTEME D'IDENTIFICATION DES PLAQUES D'IMMATRICULATIONS	12
II .1. Méthodologie adoptée.....	12
II .1.1. Techniques de détection des plaques d'immatriculation.....	12
II .1.1.1. Techniques basées sur les contours de la plaque.....	12
II .1.1.2. Techniques basées sur la signature de la plaque	13
II .1.1.3. Techniques basées sur les caractères :	13
II .1.2. Avantages et inconvénients de quelques méthodes de détection de la plaque	13
II .1.3. Méthodologie utilisée.....	14
II .1.4. Principe de fonctionnement d'un système IAPI.....	14
II .1.4.1. L'acquisition de l'image.....	14
II .1.4.2. Prétraitement de l'image	14

II .1.4.3. Détection des contours	15
II .1.4.4. Détection de la plaque d'immatriculation	16
II .1.4.5. Segmentation des caractères	16
II .1.4.6. Reconnaissance de caractères.....	16
II .2. Conception du système	17
II .2.1. Architecture du système.....	17
II .2.2. Identification des spécifications fonctionnelles et non fonctionnelles du système.....	17
II .2.2.1. Spécifications fonctionnelles.....	17
II .2.2.2. Spécifications Non Fonctionnelles :	18
II .2.2.3. Spécifications techniques :	18
II .2.3. Diagramme des cas d'utilisations	18
II .2.3.1. Les acteurs du système	18
II .2.3.2. Les cas d'utilisation du système	19
II .2.4. Description des cas d'utilisation	20
II .2.5. Conception de la base des données	22
II .2.5.1. Les Entités.....	22
II .2.5.2 Les relations.....	23
II .2.5.3. Schéma de la base des données (modèle Entités-Association).....	23
II .2.6. Présentation des technologies.....	23
II .2.6.1. La détection de la plaque d'immatriculation	24
II .2.6.1.1. Python	24
II .2.6.1.2. Tesseract.....	25
II .2.6.2. Partie serveur (back-end).....	26
II .2.6.2.1. Flask.....	26
II .2.6.2.2. Base des données :.....	27
II .2.6.2.3. Autres outils.....	29
II .3. Conclusion partielle	29
CHAPITRE III . IMPLEMENTATION DU SYSTEME ET PRESENTATION DES RESULTATS.....	30
III .1. Reconnaissance et identification des plaques	30
III .1.1. Acquisition de l'image.....	30
III .1.2 Conversion en niveau de gris	30
III .1.4. Détection des contours de la plaqu	31
III .1.5. Extraction des caractères de la plaque	32

III.1.6. Reconnaissance de caractères	33
III.1.7. Vérification et identification de la plaque d'immatriculation	33
III.2. Constitution de la base de données	33
III.2. Résultats expérimentaux	34
III.2.1. Résultats corrects	34
III.2.2. Résultats incorrectes	35
III.3. Analyse des résultats et discussions	36
III.4. Présentation de l'interface utilisateur	37
III.5. Conclusion	39
CONCLUSION GENERALE	41
BIBLIOGRAPHIE	42
ANNEXES	44
Code de fonctionnement du système	44

RESUME

Le présent travail porte sur **la conception d'un système d'identification des plaques d'immatriculation basée sur l'Open CV.**

Le processus implique une série d'étapes clés, incluant la détection de la plaque d'immatriculation, la segmentation des caractères de la plaque, et l'extraction du texte grâce à l'utilisation de Tesseract OCR. Une base de données MySQL est employée pour stocker les informations des véhicules associées aux numéros de plaques identifiés et aux propriétaires. Le système parvient à répondre aux objectifs définis, offrant ainsi une solution pratique à une problématique tangible d'identification des véhicules à Goma.

La méthodologie de traitement d'image utilisée inclut des étapes telles que la conversion en niveaux de gris, le flou gaussien, le seuillage, l'identification des contours, l'extraction de régions d'intérêt, et la reconnaissance de caractères à l'aide de l'OCR. Enfin, une fonction est mise en place pour détecter et corriger les inversions potentielles dans les numéros de plaque extraits. L'ensemble du processus démontre une approche complète et efficace pour l'identification des plaques d'immatriculation, mettant en œuvre des techniques avancées de traitement d'image et de reconnaissance de texte pour des résultats précis et fiables.

Mots clés : Vision par ordinateur, IAPI, Plaque d'immatriculation.

ABSTRACT

The present work focuses on **designing a license plate identification system based on OpenCV.**

The process involves a series of key steps, including license plate detection, character segmentation of the plate, and text extraction using Tesseract OCR. A MySQL database is utilized to store vehicle information associated with identified license plate numbers and owners. The system successfully achieves the defined objectives, providing a practical solution to a concrete issue of vehicle identification in Goma.

The image processing methodology employed includes steps such as conversion to grayscale, Gaussian blur, thresholding, contour detection, region of interest extraction, and character recognition using OCR. Finally, a function is implemented to detect and correct potential inversions in the extracted plate numbers. The entire process illustrates a comprehensive and efficient approach to license plate identification, employing advanced image processing techniques and text recognition for accurate and reliable results.

Keywords: Computer Vision, IAPI, License Plate

INTRODUCTION GENERALE

L'identification automatique de véhicule est une technique de traitement d'image permettant d'identifier les véhicules à l'aide de leurs plaques d'immatriculation. La plaque d'immatriculation désigne une voiture d'une manière unique, et son image représente une information importante pour identifier son propriétaire. En effet, Les véhicules s'identifient manuellement ou automatiquement et, les plaques d'immatriculation sont utilisées pour l'identification des véhicules dans tous les pays [1].

L'identification des véhicules par leurs plaques d'immatriculations est utilisée pour contrôler efficacement la circulation enfin de diminuer les cas d'accidents routiers en permettant d'identifier les véhicules et leurs conducteurs, et en facilitant l'application de la loi et les secours et est aussi utilisée comme application de sécurité telles que le contrôle d'accès aux zones restreintes et le suivi des véhicules recherchés [1].

Dès lors, la croissance d'automobile dans les villes rend difficile la surveillance de la circulation par les agents humains. En cas d'accident ou de recherche d'un véhicule, il est souvent difficile d'identifier le conducteur responsable, car celui-ci prend souvent la fuite. Par ailleurs, les infractions au code de la route sont de plus en plus nombreuses et ne sont pas toujours sanctionnées. Cela contribue à l'augmentation du nombre d'accidents.

C'est dans le cadre de chercher une solution à ces problèmes que nous nous sommes posés les questions suivantes :

- L'identification des plaques d'immatriculation peut-elle diminuer le taux d'accidents routiers et faciliter la recherche d'un véhicule ? si oui, comment ?
- Sur le plan sécuritaire, l'identification des plaques d'immatriculation peut-elle garantir la sécurité de la population ainsi que faciliter le contrôle d'accès aux zones restreintes ?

Pour répondre à ces questions, nous nous sommes fixés les hypothèses suivantes :

- Avec l'utilisation de l'identification des plaques d'immatriculation à partir des photos, il serait possible d'identifier les véhicules impliqués dans des infractions graves telles que

les excès de vitesse et les conduites sans permis, ce qui potentiellement pourrait contribuer à une diminution d'accidents routiers.

- La reconnaissance automatisée des plaques d'immatriculation à partir de photos pourrait être utilisée pour limiter l'accès aux zones restreintes, en identifiant rapidement les véhicules autorisés et en signalant toute présence non autorisée, ce qui renforcerait la sécurité de ces zones et réduirait les risques d'infractions.

L'objectif principale de ce travail est de concevoir un système qui pourra identifier les plaques d'immatriculations automatiquement et efficacement pour améliorer la sécurité routière, faciliter la recherche de véhicules, et renforcer la sécurité dans les zones restreintes.

Les objectifs spécifiques poursuivis sont de :

- Créer un algorithme qui détecte et délimite la région de la plaque d'immatriculation dans une image.
- Mettre en place un module d'extraction et de filtrage des caractères présents sur la plaque d'immatriculation déjà détectée puis les convertir en texte.
- Concevoir une base de données qui contient les numéros des plaques d'immatriculations qui seront associés aux informations de leurs propriétaires.
- Faire une correspondance entre les numéros des plaques extraits avec les informations des propriétaires déjà enregistrés dans la base des données enfin de lier chaque plaque à son propriétaire.
- Faire une évaluation de précision, de vitesse et d'efficacité du système par des tests pour améliorer la qualité des résultats.

Ce travail permet d'accélérer et simplifier l'identification des plaques d'immatriculations en offrant des avantages en matière de sécurité, mais également de recherche et suivi des véhicules. Il servira comme outil de recherche à la police en cas de fuite d'un véhicule lors d'un accident de circulation ou violation au code de la route en identifiant la voiture en délit pour après infliger une amende au propriétaire. Aussi, en cas de recherche d'un véhicule suspect, volé ou en surveillance le système pourra émettre une alerte lorsqu'il identifie la voiture. En fin pour la sécurité des entreprises, ce système servira à minimiser les risques d'intrusions en procédant

par un contrôle d'accès aux lieux sécurisés en vérifiant les voitures autorisées. Ce qui implique un plus sur la modernisation de la région.

Le présent travail se concentre sur la conception d'un système d'identification plaques d'immatriculation des véhicules. Notre étude se limite à la détection des véhicules, de leurs plaques d'immatriculation, ainsi qu'à la vérification des plaques dans une base de données.

Une fois qu'une correspondance est détectée, le système envoie instantanément un message à l'agent de contrôle de la circulation par le biais d'une application. Ce message inclurait l'image du véhicule en infraction, ainsi que des informations pertinentes concernant l'immatriculation du véhicule et sa situation (dans le cas où le véhicule est répertorié dans la base de données). Dans le cas contraire, si aucune information sur le véhicule n'est trouvée dans la base de données, cette situation serait également signalée dans le système comme véhicule non identifiée.

Outre l'introduction et la conclusion générales, ce travail comporte les chapitres suivants :

Chapitre 1. Généralités sur l'identification des plaques d'immatriculation : ce chapitre contient les idées émises par différents auteurs concernant notre sujet ;

Chapitre 2. Méthodologie et conception du système : dans ce chapitre nous présentons la méthodologie adoptée, le choix des technologies et outils utilisés, les étapes du développement qui nous ont permis de résoudre le problème, le schéma de la base des données et la description de l'architecture du système et algorithmes utilisés.

Chapitre 3. Implémentation du système et présentation des résultats : dans ce chapitre nous présentons les détails sur l'implémentation, les résultats obtenus lors de tests et l'interprétation des résultats.

CHAPITRE I. GENERALITES SUR L'IDENTIFICATION DES PLAQUES D'IMMATRICULATIONS

L'étude d'un système d'identification des plaques d'immatriculations qui est basée sur l'apprentissage automatique nécessite la connaissance de plusieurs notions qui décrivent les éléments qui composent ce système. Ces notions sont importantes car elles nous permettent d'acquérir certaines connaissances qui nous seront utiles lors de la conception de notre système d'identification des plaques d'immatriculations basée sur Open Cv. Dans ce chapitre, nous allons passer en revue toutes ces notions.

I.1. Définition

L'identification des plaques d'immatriculation, également appelée reconnaissance automatique des plaques d'immatriculation (ALPR), est un processus automatisé qui consiste à extraire des informations spécifiques des plaques d'immatriculation des véhicules à partir de données d'images ou de vidéos. Ces informations peuvent inclure le numéro d'immatriculation, le type de véhicule, le pays ou la région d'enregistrement, ainsi que d'autres caractéristiques pertinentes.

[2]

Les systèmes de reconnaissance des plaques d'immatriculation comprennent normalement les unités suivantes : [3]

- Caméra : Prend l'image d'un véhicule de l'avant ou de l'arrière.
- Éclairage : Une lumière contrôlée pouvant éclairer la plaque et permettre un fonctionnement de jour comme de nuit. Dans la plupart des cas, l'éclairage est infrarouge (IR), invisible pour le conducteur.
- Ordinateur : un PC sous Windows ou Linux. Il exécute le système d'application IAPI, lit les images, analyse et identifie la plaque et contrôle les interfaces avec d'autres applications et systèmes.
- Logiciel : L'application et le package d'identification.

- Base de données : Les événements sont enregistrés sur une base de données locale ou transmis sur le réseau. Les données incluent les résultats de la reconnaissance et (éventuellement) le fichier image du véhicule ou du conducteur.

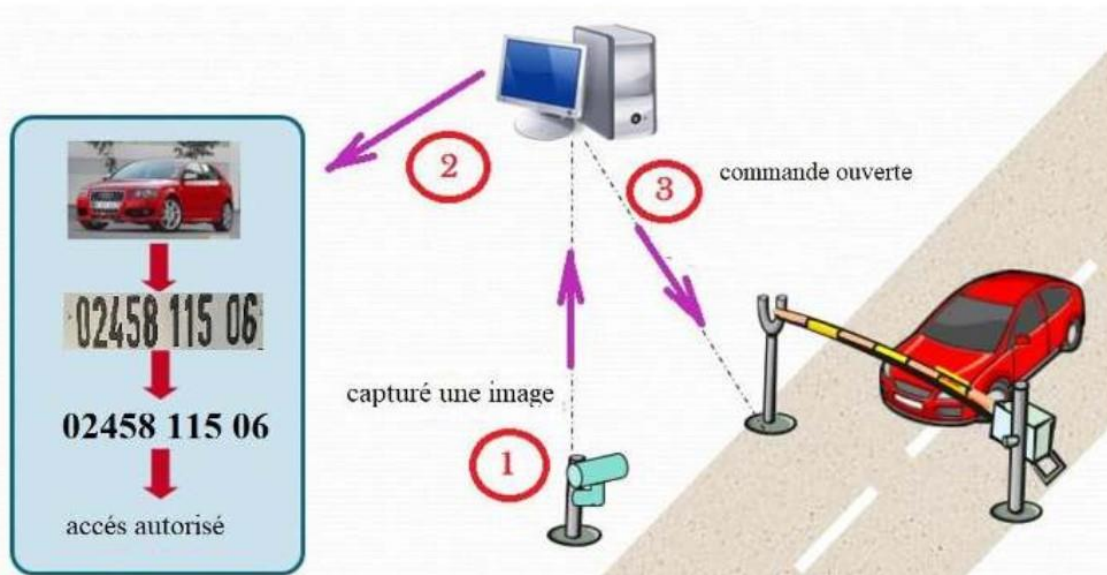


Figure 1. 1: Composition d'un système IAPI

I.2. Domaine d'application d'identification des plaques d'immatriculations

Industriellement, nous avons plusieurs applications pour de tels systèmes, citons quelques-unes d'entre elles : [3]

- Contrôle d'accès : l'ouverture automatique d'une porte pour les membres agrès dans une zone de sécurité. Ce genre de système est mis en place pour aider les agents de sécurité. Les événements sont enregistrés sur une base de données et peuvent être utilisés pour rechercher l'historique des événements en cas de besoin.



Figure 1. 2: Voiture en control d'accès

- Contrôle des frontières : le numéro de la voiture est enregistré à l'entrée ou à la sortie du pays, et utilisé pour surveiller les passages frontaliers. Chaque véhicule est enregistré dans une base de données centrale et lié à des informations supplémentaires telle que les données relatives aux passeports. Il est utilisé pour suivre tous les passages frontaliers.



Figure 1. 3:Voiture au poste frontière

- Recherche en cas de vol de véhicules : le système est déployé dans des voitures de police, et réalise une comparaison en temps réel entre les véhicules qui passent et la liste des véhicules volés. Lorsqu'un correspondant est trouvé, une alerte est déclenchée.



Figure 1. 4: véhicules équipés de caméras IAPI

- Accidents et Code pénal de la route : le numéro de plaque est utilisé pour produire une amende de violation de vitesse ou de feux rouges, ce qui peut être cause d'accidents routiers. Le processus manuel de préparation d'une amende de violation est remplacé par un processus automatisé qui réduit les surcharges et les délais.



Figure 1. 5: Voiture dépassant la limite de vitesse

I.3. Les différentes formes d'immatriculation dans le monde

Dans le monde existe de différentes plaques d'immatriculation, la plaque peut comporter des chiffres, des lettres ou des chiffres et des lettres dans un ordre spécifique ou dans un ordre quelconque, voici quelques modèles : [3]

- Suède, Belgique



Figure 1. 6: plaque d'immatriculation type AAA 111

- France, Italie



Figure 1. 7: plaque d'immatriculation type AA111AA

- Turquie



Figure 1. 8: plaque d'immatriculation type 11AA1111

I.4. Caractéristiques des plaques d'immatriculation

Congolaise

Vu qu'on s'intéresse aux plaques d'immatriculation congolaises, nous allons commencer par donner quelques caractéristiques et règles de base pour les plaques d'immatriculation de véhicules au Congo.

Pour les particuliers, les plaques d'immatriculation de la République démocratique du Congo, noir sur blanc, portent le drapeau congolais, le sigle CGO, 4 chiffres, 2 lettres et 2 chiffres. Ce modèle est en application depuis avril 2009 et est construit exclusivement par Utsch Congo, filiale de la firme allemande Utsch AG (de) pour un contrat de 10 ans . [4]

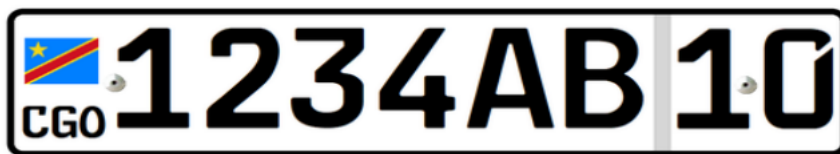


Figure 1. 9: Plaque d'immatriculation congolaise pour les particuliers

Il existe aussi des plaques spéciales en République Démocratique du Congo, entre autres : [4]

- Plaque de corps diplomatique : rouge sur jaune, 3 chiffres correspondant à un code de pays, le sigle CD, 2 ou 3 chiffres



Figure 1. 10: Corps diplomatique.

- Plaque de corps consulaire : rouge sur jaune, 3 chiffres correspondant à un code de pays, le sigle CC, 2 ou 3 chiffres
- Plaque des FARDC : blanc sur noir, FARDC suivi de 7 chiffres



Figure 1. 11: Forces armées.

- Plaque de l'ONU, de la MONUSCO : noir sur blanc, le sigle UN, 3 ou 4 chiffres



Figure 1. 12: Nations unies.

- Plaque de l'EUSEC : jaune sur bleu foncé, sigle EUSEC suivi de 3 chiffres
- Plaque de l'EUPOL : jaune sur bleu foncé, sigle EUPOL suivi de 3 chiffres
- Plaque d'immatriculation temporaire : blanc sur bleu foncé, 3 chiffres, le sigle IT, 3 chiffres
- Plaque d'autorités :
 - Présidentielle : noir sur blanc, le sigle PR suivi de 3 ou 4 chiffres



Figure 1. 13: Présidence de la République.

- Assemblée nationale : noir sur blanc, le sigle AA suivi de 3 chiffres
- Sénat : noir sur blanc, le sigle ASEN suivi de 3 chiffres
- Justice : noir sur blanc, le sigle AJus suivi de 3 chiffres
- Plaques des agents de l'état : Noir sur blanc, le sigle STA suivie de 4 chiffres et deux chiffres

I.5. Problèmes des systèmes IAPI

Il y a un certain nombre de difficultés possibles que les systèmes IAPI ne peuvent prendre en considération. Par exemple : [3]

- Pauvre résolution de l'image, généralement parce que la plaque est trop loin.

- Les images floues, en particulier le flou de mouvement.
- Mauvais éclairage et de faible contraste en raison de surexposition, réflexion ou ombres.
- Un obscurcissement d'objet (une partie de la plaque), bien souvent une barre de remorquage, ou de la saleté sur la plaque.
- Un type de police différente (certains pays n'autorisent pas de telles plaques, l'élimination du problème).

I.6. Conclusion

Dans ce chapitre nous avons présenté une étude sur le système IAPI, incluant le domaine d'application ainsi les différentes formes des plaques d'immatriculations dans le monde en précisant les caractéristiques des plaques d'immatriculations Congolaises ainsi que les problèmes que rencontre les systèmes IAPI existants.

CHAPITRE II. METHODOLOGIE ET CONCEPTION DU SYSTEME D'IDENTIFICATION DES PLAQUES D'IMMATRICULATIONS

Concevoir un système d'identification des plaques d'immatriculations basé sur l'apprentissage automatique nécessite la connaissance et la maîtrise des méthodes, des technologies, des algorithmes et outils permettant d'aboutir à une identification excellente des plaques d'immatriculation. Dans ce chapitre nous allons faire une brève présentation de la méthodologie adoptée, du choix des technologies et outils utilisés, des étapes du développement qui nous ont permis de résoudre le problème, du schéma de la base des données et de la description de l'architecture du système et algorithmes utilisés pour la conception de ce système.

II.1. Méthodologie adoptée

Ici, nous présentons un aperçu sur les techniques utilisées pour la détection des plaques d'immatriculation en évoquant les différentes méthodes utilisées ainsi que le principe de fonctionnement.

II.1.1. Techniques de détection des plaques d'immatriculation

Il existe plusieurs techniques et méthodes de détection, de localisation et d'extraction de la plaque d'immatriculation dans une image qu'utilisent les systèmes d'identification des plaques d'immatriculation(IAPI).

Parmi ces techniques, nous pouvons citer :

II.1.1.1. Techniques basées sur les contours de la plaque

Ces techniques utilisent les contours des caractères et des plaques comme des points de repère pour l'extraction. L'intensité des pixels dans les contours des caractères et des plaques est complètement différente de celle des voisins. Le point faible de cette technique est lorsque les bords de la plaque ne présentent pas une grande variation d'intensité par rapport au reste de l'image. [5] [6]

II.1.1.2. Techniques basées sur la signature de la plaque

Cette technique utilise un algorithme qui choisit une région du texte à partir d'un ensemble puis il exploite le fait qu'une plaque contient des caractères et des symboles qui sont clairement visibles même lorsque plusieurs propriétés de la plaque sont cachées. Cependant, le problème est que certaines autres régions, outre que la plaque, peuvent contenir du texte. [5] [6]

II.1.1.3. Techniques basées sur les caractères :

Les approches basées sur les caractères vont chercher toutes les régions de l'image possédant des caractères puis éliminent les faux résultats en prenant en considération des paramètres prédéfinies. Ces algorithmes ont l'avantage de détecter les plaques d'immatriculation même dans les cas les plus extrêmes mais furent délaissées à cause du nombre important de faux positifs lors de la détection. [5] [6]

II.1.2. Avantages et inconvénients de quelques méthodes de détection de la plaque

Voici quelques avantages et inconvénients de méthodes de détection de la plaque d'immatriculation :

Tableau 1. 1: Avantages et inconvénients de quelques méthodes de détection de la plaque

Méthodes	Raisonnement	Avantages	Inconvénients	Références
Utilisant les contours	Les contours de la plaque forment un rectangle	Simple, rapide et efficace	Trop de contours indésirables sur les images complexes	[7]
Utilisant les propriétés de la plaque	La plaque d'immatriculation possède une couleur spécifique	Utilisé lorsque la plaque d'immatriculation est inclinée sur l'image	RVB est limité à la condition d'illumination.	[7]
Utilisant les caractères	Il doit y avoir des caractères sur la plaque d'immatriculation	Détection même dans les cas extrêmes	Consomme beaucoup de temps et génère trop d'erreurs en présence d'autre texte dans l'image	[7]

II.1.3. Méthodologie utilisée

Nous avons utilisé deux méthodes combinées, celles utilisant les contours et celles utilisant les caractères, car le processus de détection et de segmentation des caractères d'une plaque d'immatriculation implique une séquence d'opérations allant de la détection des contours jusqu'à l'extraction des caractères à l'aide d'algorithmes de reconnaissance optique de caractères (OCR).

II.1.4. Principe de fonctionnement d'un système IAPI

Un système IAPI est généralement répartis en 6 grandes fonctions dont :

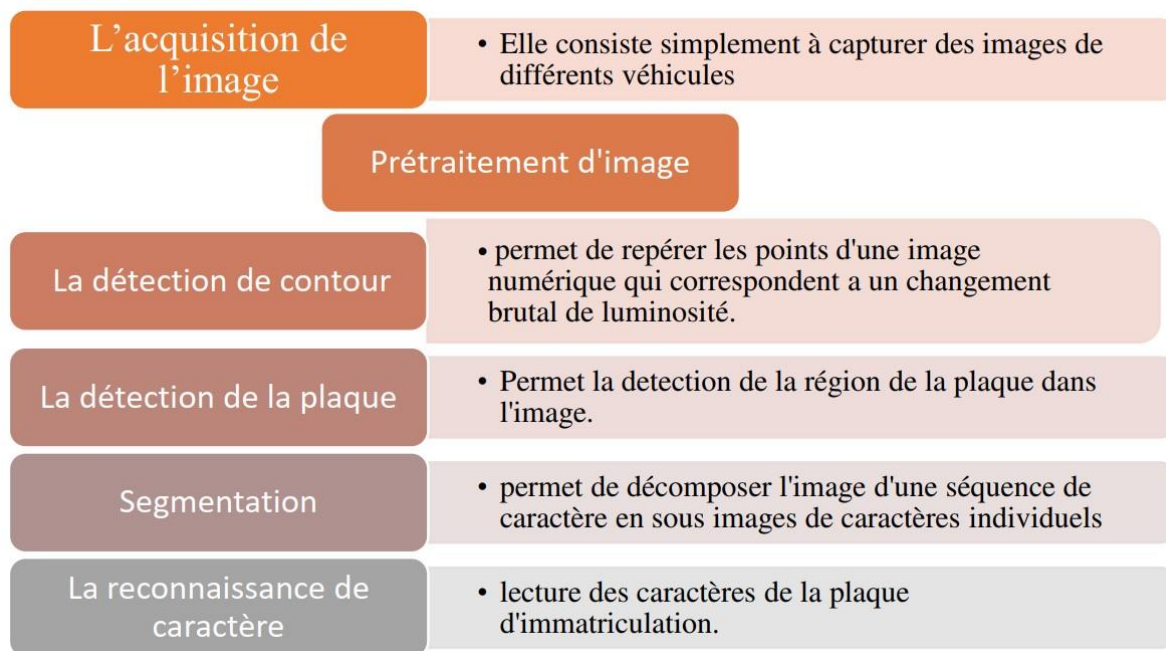


Figure2. 1 :Principales fonctions d'un système IAPI [3]

II.1.4.1. L'acquisition de l'image

L'acquisition d'images contenant la plaque d'immatriculation a été réalisée à Goma, en RDC, avec des smartphones. Les images ont été capturées dans toute la ville, mais surtout dans les deux campus de notre université ULPGL et sur la route les reliant. Les images seront utilisées pour l'entraînement du système ici proposé ainsi que pour l'évaluation de ses performances.

II.1.4.2. Prétraitement de l'image

Le prétraitement d'image a été utilisé pour améliorer la qualité des images et faciliter l'identification des caractères de la plaque d'immatriculation. Ce projet s'appuie sur la puissance d'Open CV, une bibliothèque open-source de traitement d'images et de vision par ordinateur,

pour accomplir les tâches de traitement et de reconnaissance d'images. Les opérations de prétraitement d'image que nous avons utilisées sont les suivantes : [8]

- ✓ **Conversion de l'image en niveau de gris** : La plupart des algorithmes de reconnaissance de caractères fonctionnent mieux avec des images en niveaux de gris. Cette étape permet de réduire la complexité du traitement, tout en préservant les informations nécessaires à la reconnaissance des caractères.
- ✓ **Redimensionnement et floutage de l'image** : L'image en niveaux de gris subit un redimensionnement, un flou médian et un flou gaussien à l'aide de `cv2.resize`, `cv2.medianBlur` et `cv2.GaussianBlur`. L'image est agrandie en lui appliquant un facteur d'échelle de 3 dans les deux directions (horizontale et verticale) à l'aide des paramètres `fx` et `fy` égaux à 3. Cela permet d'augmenter la taille de l'image pour potentiellement améliorer la détection des caractères lors des étapes ultérieures. Le flou gaussien et le flou médian ont été utilisés pour atténuer le bruit et les détails indésirables dans l'image. Ces techniques sont particulièrement utiles pour éliminer le bruit de fond, les imperfections et les détails inutiles qui pourraient interférer avec la détection des contours et la reconnaissance des caractères. Le flou gaussien est efficace pour réduire le bruit global tandis que le flou médian est plus adapté pour éliminer les pixels extrêmes qui pourraient perturber la détection des contours.
- ✓ **Seuillage (Otsu)** : On applique à l'image floutée un seuillage d'Otsu pour obtenir une image binaire à l'aide de `cv2.threshold` avec l'option `cv2.THRESH_OTSU`. Cela permet de convertir l'image en une forme binarisée, où les pixels sont soit noirs (0) soit blancs (255), facilitant ainsi la segmentation des caractères. Nous avons utilisé la binarisation inverse.
- ✓ **Dilatation** : Une opération de dilatation est effectuée à l'aide de `cv2.dilate` pour épaissir les caractères et les rendre plus distincts, ce qui peut faciliter la segmentation ultérieure.

II.1.4.3. Détection des contours

Après avoir effectué le prétraitement de l'image, le code `cv2.findContours` est utilisé pour détecter les contours des éléments binaires dans l'image seuillée. Cette étape est cruciale pour isoler les caractères individuels de la plaque d'immatriculation, ce qui aide à leur reconnaissance.

Cette technique est utilisée pour identifier la région de la plaque d'immatriculation à partir de l'image donnée. L'objectif principal de ce type de systèmes est de localiser la région de la plaque d'immatriculation à partir de l'image des véhicules capturés.

II.1.4.4. Détection de la plaque d'immatriculation

Il s'agit de chercher les coordonnées de la plaque dans l'image en utilisant de nombreuses méthodes. L'image doit être prétraitée pour avoir des bons résultats. Cette phase est complexe et importante. Elle détermine la rapidité et la robustesse du système.

II.1.4.5. Segmentation des caractères

La segmentation des caractères est une opération qui vise à décomposer une image en séquences des sous-images de symboles individuels. Cette méthode utilise la propriété que toutes les plaques d'immatriculation sont rectangulaires et que leurs caractères partagent le même arrière-plan. L'objectif de la segmentation des chaînes de caractères, c'est la séparation des caractères l'un de l'autre [8].

II.1.4.6. Reconnaissance de caractères

La dernière étape du système IAPI est la reconnaissance optique de caractères (OCR) qui est un procédé permettant de traduire une image de texte numérisée en un document texte. Cette technologie, l'OCR, permet de reconnaître automatiquement les caractères grâce à un mécanisme optique. Dans le cas des êtres humains, nos yeux sont un mécanisme optique. L'image vue par les yeux est une entrée pour le cerveau. La capacité à comprendre ces entrées varie chez chaque personne en fonction de plusieurs facteurs. L'OCR est une technologie qui fonctionne comme une capacité de lecture humaine, bien que l'OCR ne puisse pas rivaliser avec les capacités de lecture de l'être humain. Parmi les techniques existantes nous avons choisi d'utiliser Tesseract. [6]

Tesseract est un moteur de reconnaissance optique de caractères pour différents systèmes d'exploitation. C'est un logiciel libre, publié sous la licence Apache, version 2.0, et le développement est sponsorisé par Google depuis 2006. Tesseract est considéré comme l'un des moteurs de OCR open source les plus précis actuellement disponibles [9].

II.2. Conception du système

II.2.1. Architecture du système

Ce système est composé de quatre phases essentielles. Tout d'abord, une phase de localisation de la plaque est effectuée afin de détecter sa position précise sur l'image du véhicule. Ensuite, une phase de prétraitement est réalisée, comprenant l'extraction de la plaque d'immatriculation et l'ajustement de la géométrie de l'image pour l'observer sous un angle approprié. La troisième étape consiste à extraire les caractéristiques pertinentes de la plaque, qui seront utilisées dans la dernière étape d'identification.

II.2.2. Identification des spécifications fonctionnelles et non fonctionnelles du système

Dans cette section, nous détaillons les spécifications essentielles du système, à la fois du point de vue de ses fonctionnalités (spécifications fonctionnelles) et de ses performances, et d'autres aspects clés (spécifications non fonctionnelles). Ces spécifications servent de base pour la conception, le développement et l'évaluation du système.

II.2.2.1. Spécifications fonctionnelles

Le système doit être à mesure de :

- Annoter et acquérir des images de la plaque d'immatriculation du véhicule provenant de différentes sources (caméras de surveillance, smartphones, etc.) ;
- Localiser et extraire les zones potentielles des plaques d'immatriculation dans les images ;
- Analyser les caractères présents sur la plaque d'immatriculation pour réaliser une reconnaissance précise ;
- Enregistrer et afficher les résultats de reconnaissance (plaque d'immatriculation, date et heure, etc.).
- Consulter une base de données pour vérifier si la plaque du véhicule détecté correspond à une donnée de cette base ;
- Faire une identification de la plaques d'immatriculation détecté en temps réel ;

II.2.2.2. Spécifications Non Fonctionnelles :

- Performance : Le système doit traiter les images rapidement pour une identification quasi instantanée des plaques d'immatriculation.
- Fiabilité : Le système doit fonctionner de manière fiable sans erreurs ou pannes fréquentes.
- Évolutivité : Le système doit être conçu de manière à pouvoir être mis à jour et étendu pour prendre en charge de nouvelles fonctionnalités à l'avenir.

II.2.2.3. Spécifications techniques :

- Langages et technologies : utiliser des bibliothèques et des Framework d'apprentissage automatique, comme Open CV qui est une bibliothèque open source gratuite spécialisée dans le traitement d'images et la vision par ordinateur.
- Entraînement du modèle : collecter et annoter un ensemble de données d'images de plaques d'immatriculation pour entraîner le modèle d'apprentissage automatique.
- Prétraitement des images : appliquer des techniques d'amélioration d'image, de filtrage et de normalisation pour améliorer la qualité des images traitées.

II.2.3. Diagramme des cas d'utilisations

La conception du diagramme des cas d'utilisation du système d'identification des plaques d'immatriculation commence par l'identification des acteurs jouant un rôle dans le système. Par après, nous définissons les cas d'utilisation principaux et chaque cas d'utilisation est décrit avec ses actions, ses acteurs associés et ses interactions, fournissant une vue claire des fonctionnalités offertes par le système.

II.2.3.1. Les acteurs du système

Comme vu dans le cours de génie logiciel, un acteur est un utilisateur du système qui identifie un rôle, interagit directement avec le système et qui peut être impliqué dans plusieurs cas d'utilisations [10].

Ces acteurs sont :

- **L'opérateur de la caméra** : cet acteur capture les images des plaques d'immatriculation via un smartphone et les annoté.

- **Le système d'identification** : cet acteur localise et extrait des zones potentielles des plaques d'immatriculation, détecte et extrait la plaque d'immatriculation, enregistre les données de la plaque et fait la vérification des données de la plaque pour son identification.
- **L'administrateur du système** : cet acteur s'authentifie au système, envoie une photo de plaque d'immatriculation au système d'identification et reçoit un message de validation lorsque la plaque a été identifiée.

II.2.3.2. Les cas d'utilisation du système

Les cas d'utilisation sont une description "narrative" de comment est utilisé le système. Ils sont basés sur des scénarios qui sont un séquençement des événements se passant entre l'utilisateur et le système et sont toujours déclenchés par les acteurs. Ils représentent le système du point de vue de l'acteur [10].

Ces cas d'utilisations sont :

- **L'annotation des images** : l'opérateur acquiert les images des plaques d'immatriculations, les enregistre et fait une annotation.
- **Détection de la plaque** : le système localise et extrait des zones potentielles des plaques d'immatriculation
- **Extraction des caractères de la plaque** : le système détecte et extrait les caractères de la plaque d'immatriculation.
- **Enregistrement des données de la plaque** : le système enregistre les caractères extraits de la plaque dans la base des données.
- **Vérification et identification de la plaque** : le système et/ou l'administrateur fait une vérification des données extraites avec celles annotée pour l'identification.

Voici en bref le diagramme de cas d'utilisations :

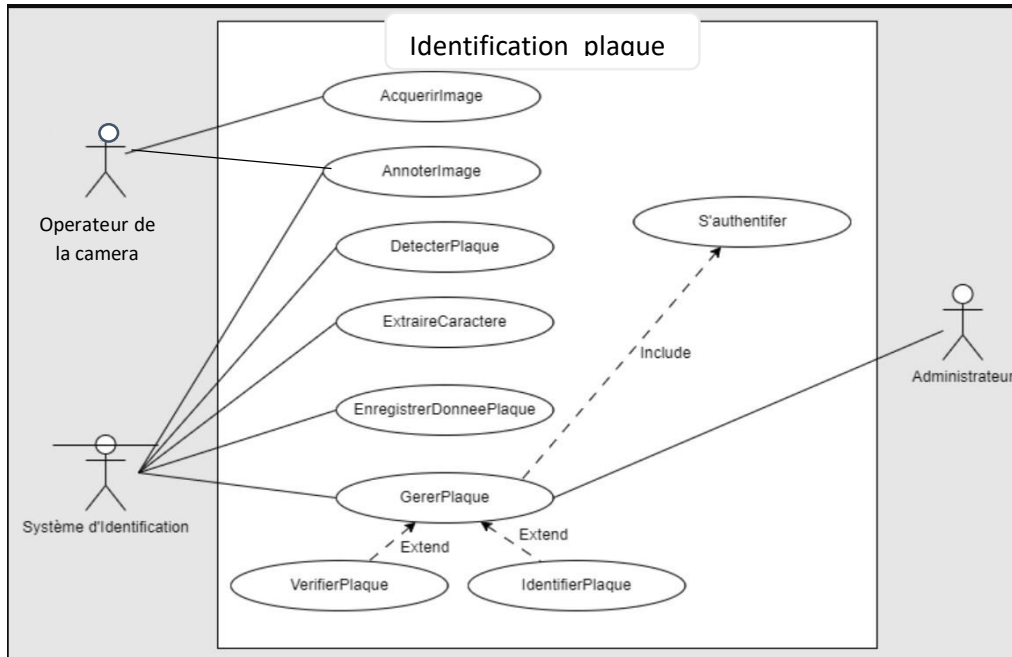


Figure2. 2: Diagramme des cas d'utilisation d'un système IAPI

II.2.4. Description des cas d'utilisation

La description des cas d'utilisation du système d'identification des plaques d'immatriculation détaille les étapes chronologiques des actions d'interaction entre les différents acteurs. À travers cette description, on visualise comment le système répond aux cas d'utilisation précités.

Tableau1. 2: DCU-Acquisition des images

Cas d'Utilisation	Acquisition des images
ID	1
Description Brève	Acquisition des photos des plaques d'immatriculation
Acteur	Operateur de la camera
Précondition	L'opérateur doit avoir un outil pour capturer les photos
Enchaînement principal	L'opérateur capture les photos des plaques et les envoie dans la base des données.

Tableau1. 3: DCU-Annotation des images

Cas d'Utilisation	Annotation des images
ID	2
Description Brève	Annotation des plaques d'immatriculation
Acteur	Operateur de la camera, Système d'identification
Précondition	L'opérateur doit avoir les plaques à annoter
Enchaînement principal	L'opérateur annote les plaques et les envoie dans la base des données.

Tableau 1. 4: DCU-Détection de la plaque

Cas d'Utilisation	Détection de la plaque
ID	3
Description Brève	le système localise et extrait des zones potentielles des plaques d'immatriculation
Acteur	Système d'identification
Précondition	L'opérateur doit fournir l'image de la plaque au système
Enchaînement principal	Le système prétraite l'image, fait la détection des contours puis celles de la plaque d'immatriculation et segmente les caractères.

Tableau 1. 5: DCU-Extraction des caractères de la plaque

Cas d'Utilisation	Extraction des caractères de la plaque
ID	4
Description Brève	Le système détecte et extrait les caractères de la plaque d'immatriculation.
Acteur	Système d'identification
Précondition	La plaque doit être détectée
Enchaînement principal	Le système doit analyser les caractères présents sur la plaque d'immatriculation pour réaliser une reconnaissance précise.

Tableau 1. 6 : DCU-Enregistrement des données de la plaque

Cas d'Utilisation	Enregistrement des données de la plaque
ID	5
Description Brève	Enregistrement des caractères extraits
Acteur	Système d'identification
Précondition	Les caractères doivent être extraits
Enchaînement principal	Le système enregistre les caractères extraits de la plaque dans la base des données

Tableau 1. 7 : DCU-Vérification de la plaque

Cas d'Utilisation	Vérification des données
ID	6
Description Brève	Vérification de la plaque dans la base des données
Acteur	Système d'identification, administrateur
Enchaînement principal	On fait une vérification des données extraites avec celles annotées.

Tableau 1. 8: DCU-Identification de la plaque

Cas d'Utilisation	Identification de la plaque
ID	7
Description Brève	Identification de la plaque d'immatriculation
Acteur	Système d'identification, administration
Précondition	Les données doivent être enregistrer dans la base des données
Enchaînement principal	le système fait une identification puis affiche à l'utilisateur les résultats d'identification

II.2.5. Conception de la base des données

La conception d'une base de données pour un système d'identification des plaques d'immatriculation implique de définir les entités clés telles que les plaques annotées et les plaques extraits, d'établir des relations appropriées entre ces entités, d'indexer les colonnes importantes pour des performances optimales et de mettre en place un plan de sauvegarde régulier. Une conception réfléchie et bien structurée est essentielle pour assurer l'efficacité et la fiabilité du système dans son ensemble.

II.2.5.1. Les Entités

- **PlaqueAnnotée :**
 - id_plaque (clé primaire)
 - numero_plaque
 - photo_plaque
 - id_propriétaire (clé étrangère vers ProprietairePlaque)
 - date_enregistrement
- **PlaqueExtrait :**
 - id_plaque_extrait (clé primaire)
 - numero_plaque_extrait
 - date_enregistrement
- **ProprietairePlaque :**
 - id_propriétaire (clé primaire)
 - nom
 - post_nom

- numero_telephon
- mail
- adresse

II.2.5.2 Les relations

- Une plaque d'immatriculation est associée à un seul propriétaire (relation un-à-un entre PlaqueAnnotée et ProprietairePlaque).
- Un propriétaire peut avoir plusieurs plaques, mais chaque plaque est associée à un seul propriétaire (relation un-à-plusieurs entre ProprietairePlaque et plaque).

II.2.5.3. Schéma de la base des données (modèle Entités-Association)

- **PlaqueAnnotée** (id_plaque, numero_plaque, photo_plaque, id_proprietaire, date_enregistrement)
CP : id_plaque
CE : id_proprietaire (référence à **ProprietairePlaque**)
- **PlaqueExtrait** (id_plaque_extrait, numero_plaque_extrait, date_enregistrement)
CP : id_plaque_extrait
- **ProprietairePlaque** (id_proprietaire, nom, post_nom, numero_telephon, mail, adresse)
CP : id_proprietaire

II.2.6. Présentation des technologies

Dans le cadre du développement de notre système, nous avons utilisé un ensemble de technologies pour pouvoir répondre à nos objectifs, tel que les langages de programmation, les bibliothèques, les bases de données, les frameworks et autres outils essentiels à sa réalisation.

Notre système se compose de deux grandes parties interconnectées, chacune jouant un rôle essentiel dans le fonctionnement global de l'application.

- Nous commencerons par explorer la partie centrale de notre système, le traitement et l'identification des plaques d'immatriculation. Cette section présentera les techniques de traitement d'images, de vision par ordinateur et de reconnaissance de caractères que nous avons utilisées pour extraire efficacement les informations des plaques d'immatriculation pour leur identification.
- Enfin, nous examinerons le back-end du système proposé, qui assure la gestion des données, la logique métier et la communication avec d'autres composants.

II.2.6.1. La détection de la plaque d'immatriculation

II.2.6.1.1. Python

Python est un langage de programmation largement reconnu et utilisé dans le domaine de l'intelligence artificielle en raison de ses spécificités et de ses nombreux avantages. Voici quelques points clés concernant Python dans le contexte de l'Apprentissage automatique : [11]

- Réputation dans l'intelligence artificielle : [12]
 - ✓ **Popularité** : Python est l'un des langages de programmation les plus populaires pour le développement d'applications d'IA. Il est largement adopté dans la communauté de l'IA en raison de sa simplicité et de sa polyvalence.
 - ✓ **Richesse des bibliothèques** : Python dispose d'une vaste collection de bibliothèques dédiées à l'IA, notamment TensorFlow, Keras, PyTorch, scikit-learn, et bien d'autres. Ces bibliothèques fournissent des outils essentiels pour la création, la formation et le déploiement de modèles d'IA.
 - ✓ **Écosystème actif** : La communauté Python est active et collaborative, ce qui se traduit par des mises à jour fréquentes, des contributions ouvertes et des ressources d'apprentissage abondantes.
 - ✓ **Grande communauté** : Python bénéficie d'une communauté active de développeurs, ce qui signifie qu'il existe une abondance de tutoriels, de documentation et de forums de discussion pour obtenir de l'aide et des conseils.
 - ✓ **Bibliothèques puissantes** : Les bibliothèques Python, telles que TensorFlow et PyTorch, offrent des capacités avancées en matière d'apprentissage automatique et de deeplearning, simplifiant ainsi le développement de modèles complexes.

- ✓ **Support multiplateforme** : Python est compatible avec plusieurs systèmes d'exploitation, ce qui permet aux développeurs de créer des applications d'IA pour divers environnements.
- Avantages de Python : [12]
 - ✓ **Syntaxe lisible** : Python est réputé pour sa syntaxe propre et lisible, ce qui en fait un langage accessible aux débutants et facile à comprendre pour les développeurs expérimentés.
 - ✓ **Facilité d'apprentissage** : Python est considéré comme l'un des langages les plus simples à apprendre en raison de sa clarté et de sa simplicité. Cela en fait un choix idéal pour les personnes qui souhaitent se lancer dans l'IA sans avoir à surmonter une courbe d'apprentissage abrupte.
 - ✓ **Polyvalence** : Python est polyvalent et peut être utilisé pour diverses tâches en IA, de l'analyse de données à la création de modèles d'apprentissage automatique, en passant par le traitement du langage naturel et la vision par ordinateur.

II .2.6.1.2. Tesseract

Comme nous l'avons signalé précédemment, pour l'extraction des informations des plaques d'immatriculation, nous avons intégré Tesseract qui est une bibliothèque de reconnaissance optique de caractères (OCR) qui permet de convertir du texte à partir d'images. Elle se caractérise par sa simplicité d'utilisation et sa capacité à reconnaître du texte dans plusieurs langues, ce qui en fait un outil précieux pour diverses applications. [13]

- Avantages de Tesseract : [14]
 - ✓ **Précision de Reconnaissance** : Tesseract utilise des algorithmes avancés pour reconnaître avec précision les caractères, ce qui est crucial pour l'identification correcte des plaques d'immatriculation.
 - ✓ **Support Multilingue** : Tesseract prend en charge plusieurs langues, ce qui le rend polyvalent pour l'identification de plaques d'immatriculation dans des contextes internationaux.

- ✓ **Open Source** : En tant que logiciel open source, Tesseract est accessible et modifiable, ce qui permet aux développeurs d'adapter le système selon les besoins spécifiques du projet.
- ✓ **Entraînement Personnalisé** : Il offre la possibilité d'entraîner le modèle avec des données spécifiques, améliorant ainsi sa performance dans des scénarios particuliers, comme la reconnaissance de plaques d'immatriculation spécifiques à une région.
- ✓ **Évolutivité** : Tesseract peut être utilisé sur différentes plates-formes, ce qui le rend scalable et adaptable à divers environnements.
- ✓ **Communauté Active** : La communauté open source autour de Tesseract contribue constamment à son amélioration, assurant des mises à jour régulières et une résolution rapide des problèmes.

II.2.6.2. Partie serveur (back-end)

II.2.6.2.1. Flask

Flask est un framework web Python qui joue un rôle central dans la partie serveur (back-end) de l'application. Il est utilisé pour créer des services côté serveur, y compris des API, qui gèrent la logique métier, la persistance des données et la sécurité de l'application. Flask a été utilisé de manière spécifique pour gérer la partie serveur de l'application, en fournissant des routes pour les demandes HTTP, en définissant des modèles de données, et en permettant la communication avec une base de données. Par exemple, Flask peut être utilisé pour créer des points de terminaison API qui gèrent les demandes des clients et renvoient des réponses appropriées. [15]

- Avantages de Flask [15]
 - ✓ **Simplicité** : Flask est connu pour sa simplicité et sa facilité de prise en main. Son code est clair et facile à comprendre, ce qui permet aux développeurs de rapidement créer des applications back-end sans une courbe d'apprentissage abrupte.
 - ✓ **Flexibilité** : Vous pouvez choisir les bibliothèques, les outils et les modèles qui conviennent le mieux à votre projet.

- ✓ **Léger** : Flask est un framework léger avec un ensemble de fonctionnalités essentielles. Cela signifie que votre application n'inclura que ce dont elle a besoin, ce qui peut contribuer à des temps de chargement plus rapides. [16]
- ✓ **Soutien actif et communauté** : Flask bénéficie d'une communauté active de développeurs et de nombreuses ressources, ce qui signifie que vous pouvez trouver des solutions aux problèmes courants et obtenir de l'aide en cas de besoin.

II.2.6.2.2. Base des données :

MySQL est un système de gestion de base des données relationnelle open source largement utilisé. Il permet de stocker et d'organiser des données de manière structurée, ce qui est essentiel pour de nombreuses applications web et logicielles. [17]

- **Spécificités de MySQL pour un système d'identification des plaques d'immatriculation** [17]
 - ✓ **Open Source** : MySQL est distribué sous une licence open source, ce qui signifie que son code source est accessible et modifiable par la communauté.
 - ✓ **Stockage des Données** : MySQL peut stocker efficacement les données liées aux plaques d'immatriculation, y compris les numéros, les images associées, les informations du propriétaire, et les données de localisation.
 - ✓ **Prise en Charge des Données Image** : MySQL peut gérer des données d'image, ce qui est crucial pour stocker et traiter les images des plaques d'immatriculation capturées par des caméras.
 - ✓ **Gestion des Index** : L'utilisation d'index dans MySQL peut accélérer la recherche des plaques d'immatriculation, améliorant ainsi les performances du système.
 - ✓ **Langage SQL pour les Requêtes** : La puissance du langage SQL facilite la création de requêtes complexes pour filtrer et analyser les données des plaques d'immatriculation.

- ✓ **Sécurité des Données** : MySQL offre des fonctionnalités de sécurité telles que le chiffrement des données, ce qui est crucial pour protéger les informations sensibles liées aux plaques d'immatriculation et aux propriétaires.
- ✓ **Gestion des Utilisateurs** : Les fonctionnalités de gestion des utilisateurs de MySQL permettent de définir des privilèges d'accès appropriés, garantissant que seules les personnes autorisées peuvent accéder aux données du système.
- ✓ **Multiplateforme** : MySQL est compatible avec plusieurs systèmes d'exploitation, ce qui le rend polyvalent et adaptable à différentes configurations.
- ✓ **Communauté Active** : En tant que projet open source populaire, MySQL bénéficie d'une communauté active qui contribue au développement continu, fournit des conseils et propose des extensions.
- ✓ **Intégration** : MySQL est souvent intégré à de nombreuses applications web, serveurs d'applications et environnements de développement grâce à son API standard et à ses pilotes pour plusieurs langages de programmation.
- **Avantages de MySQL pour un système d'identification des plaques d'immatriculation** : [18]
 - ✓ **Évolutivité** : MySQL peut gérer une grande quantité de données, ce qui est essentiel pour un système d'identification des plaques d'immatriculation à grande échelle.
 - ✓ **Performance** : Avec ses performances élevées, MySQL peut traiter rapidement les requêtes liées à l'identification des plaques d'immatriculation, assurant une réponse rapide du système.
 - ✓ **Interopérabilité** : MySQL peut être facilement intégré avec d'autres technologies utilisées dans un système d'identification des plaques d'immatriculation, comme des logiciels de traitement d'image et des applications frontales.
 - ✓ **Redondance et Réplication** : La fonction de réplication de MySQL peut être utilisée pour créer des copies de sauvegarde des données, assurant ainsi une redondance et une disponibilité élevée du système.

- ✓ **Communauté Active** : En bénéficiant d'une communauté active, MySQL peut profiter des contributions et des améliorations continues, assurant un support à long terme pour le système d'identification des plaques d'immatriculation.

II.2.6.2.3. Autres outils

- ✓ **L'éditeur VS code** : Visual Studio Code est un éditeur de code gratuit et multiplateforme, développé par Microsoft. Il permet de créer et de modifier du code source dans différents langages de programmation, comme Python, Java, C#, HTML, CSS, etc. Il offre des fonctionnalités comme la coloration syntaxique, l'auto complétion, le débogage, le contrôle de version, les extensions et bien plus encore. VS Code est un outil puissant et personnalisable pour les développeurs web et les programmeurs. Nous l'avons donc utilisé pour taper les codes sources. [19]

II.3. Conclusion partielle

Dans ce chapitre, nous avons présenté la méthodologie, la conception et les technologies utilisées pour le développement d'un système d'identification des plaques d'immatriculation. La méthodologie utilisée est basée sur une approche agile, qui permet de développer le système de manière itérative et incrémentale. Cela permet de garantir que le système répond aux besoins des utilisateurs tout au long du processus de développement. La conception du système est basée sur une architecture modulaire, qui permet de faciliter le développement, la maintenance et l'évolution du système. Le système est divisé en deux grandes parties : le back-end et le traitement et l'identification des plaques d'immatriculation. Les choix technologiques ont été faits en fonction des besoins spécifiques du système et des contraintes techniques.

CHAPITRE III. IMPLEMENTATION DU SYSTEME ET PRESENTATION DES RESULTATS

Dans ce chapitre, nous plongeons dans les détails de la mise en œuvre du système ici proposé ainsi que dans la présentation des résultats obtenus tout au long du parcours de son développement. Ce chapitre marque le point de convergence entre la conception conceptuelle et la réalisation concrète de la solution innovante que nous proposons.

III.1. Reconnaissance et identification des plaques

III.1.1. Acquisition de l'image

En premier lieu, nous capturons une image qui est ensuite stockée au format JPG ou PNG et envoyée au système pour son traitement.



Figure3. 1: image originale

III.1.2 Conversion en niveau de gris

Ici, on transforme l'image originale en couleur en une image en niveaux de gris. Le niveau de gris réduit le bruit de l'image dans une certaine mesure tout en facilitant le traitement ultérieur de l'image.

L'instruction ci-dessous permet la conversion en niveau de gris.

```
gray = cv2.resize( gray, None, fx =3, fy =3, interpolation = cv2.INTER_CUBIC)
```



Figure3. 2: image en niveau de gris

III.1.3. Conversion en niveau binaire

Ici on réduit les informations contenues dans l'image de 256 nuances de gris à 2 : noir et blanc ; cela facilite la reconnaissance des caractères.

L'instruction ci-dessous permet la conversion d'une image en niveau de gris en une image binariser :

```
ret, thresh = cv2.threshold(blur, 0, 255, cv2.THRESH_OTSU |  
cv2.THRESH_BINARY_INV)
```



Figure3. 3: image binaire

III.1.4. Détection des contours de la plaque

Ici le système localise et extrait des zones potentielles des plaques d'immatriculation en utilisant un algorithme de détection des contours suivante :

```
try:
```

```

    contours, hierarchy = cv2.findContours(dilation, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
except:
    ret_img, contours, hierarchy = cv2.findContours(dilation, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
sorted_contours = sorted(contours, key=lambda ctr: cv2.boundingRect(ctr)[0])

```

III.1.5. Extraction des caractères de la plaque

Après la détection de la plaque on extrait et segmente les caractères de la plaque pour leur reconnaissance via cet algorithme :

```

for cnt in sorted_contours:
    x,y,w,h = cv2.boundingRect(cnt)
    height, width = im2.shape

    # if height of box is not a quarter of total height then skip
    if height / float(h) > 4: continue
    ratio = h / float(w)
    # if height to width ratio is less than 1.5 skip
    if ratio < 1.5: continue
    area = h * w
    # if width is not more than 45 pixels skip
    if width / float(w) > 45: continue
    # if area is less than 100 pixels skip
    if area < 100: continue
    # draw the rectangle
    rect = cv2.rectangle(im2,(x,y), (x+w, y+h), (0,255,0),2)
    roi = thresh[y-5:y+h+5, x-5:x+w+5]
    roi = cv2.bitwise_not(roi)
    roi = cv2.medianBlur(roi, 5)
    roi = cv2.bitwise_not(roi)
    cv2.imwrite(f"character{character_count}.jpg", roi)
    character_count += 1
    print("Nombre total de caractères filtrés et sauvegardés :",
character_count)
    cv2.imshow("ROI", roi)
    cv2.waitKey(0)

```

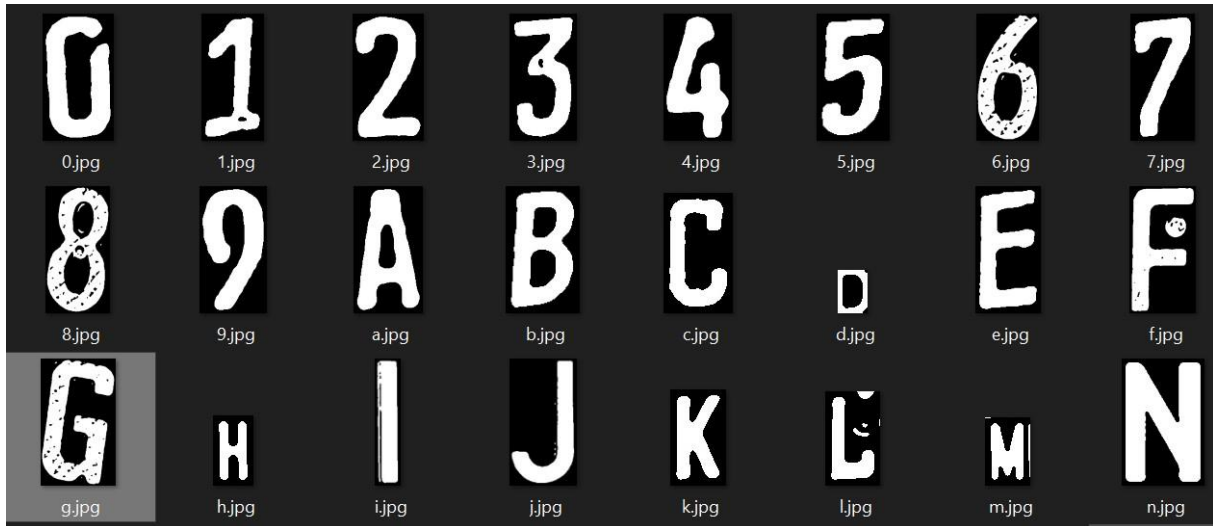


Figure3. 4: exemple des caractères extraits et segmentés de la plaque

III.1.6. Reconnaissance de caractères

A cette étape, on utilise pytesseract pour reconnaître les caractères extraits de la plaque et les stocker dans une variable via cet algorithme :

```
text = pytesseract.image_to_string(roi, config='-c
tessedit_char_whitelist=0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ --psm 8 --oem 3')
plate_num += text.strip() # on Utilise strip() pour enlever les espaces
indésirables
```

III.1.7. Vérification et identification de la plaque d'immatriculation

Ici le système fait une vérification des données extraites avec celles annotée pour l'identification puis affiche à l'utilisateur les résultats d'identification.

III.2. Constitution de la base de données

Nous avons créé une base de donnée qui contient trois tables, l'une contenant les photos des plaques capturées (240) ainsi que leur annotation, l'autre contenant les informations sur les propriétaires des véhicules et la troisième contenant les plaques extraites et les caractères reconnus et identifiés.

III.2. Résultats expérimentaux

III.2.1. Résultats corrects

Voici un extrait des résultats correctes d'identification des plaques d'immatriculation :

Tableau3. 1: Exemple des plaques identifiées avec succès




Image	Plaque réelle	Plaque extraite	Caractère non reconnu
		8912AA07	0
		8134AF19	0
		2755AF19	0
		9921AC19	0
		3177AB19	0







		7326AF19	0
		2214AG19	0
		9635AB19	0
		3323AG19	0

III.2.2. Résultats incorrectes

Voici un extrait des résultats incorrectes d'identification des plaques d'immatriculation :

Tableau3. 2 : exemple des plaques non identifiées car erronée

Image	Plaque réelle	Plaque reconnue	Caractère non reconnu
 Ombre		2488AG49	2 3 à 4 , 1 à 4
 Mauvaise éclairage	Plaque non extraite	/	/

 <p>Reffet solaire</p>		498GAG4	4 5 à 9, 0 à G, 1 à 4, le chiffre 9 non reconnu
 <p>Fantaisie sur la plaque</p>		1506ANI	2 9 à 0, le chiffre 0 non lue
 <p>mauvaise prise</p>		/	/

III.3. Analyse des résultats et discussions

Notre système IAPI a été testé en temps différé sur quelques images. Ici, nous allons discuter les résultats de localisation et de reconnaissance.

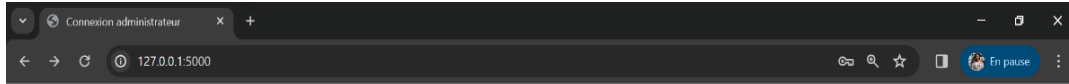
Pour les cas de mauvaise reconnaissance des chiffres : Cela est due à plusieurs raisons tel que le reflet de lumière qui peut conduire à une fausse reconnaissance du chiffre comme par exemple un 0 reconnu comme étant un 9 ou un E reconnu comme étant un F en raison d'une mauvaise écriture, les motifs ajoutés (plaques fantaisistes, non conformes) sur la plaque affectant la bonne détection, l'ombre conduisant à une fausse reconnaissance des chiffres.

Pour les cas des plaques non détectées Cela est dû à l'éclairage faible de la scène où la voiture est capturée en entraînant des fausses détections de contours. C'est aussi dû à un mauvais emplacement de la camera qui capture la plaque de façon inclinée.

En conclusion, il est important de tenir compte de ces éléments afin d'obtenir des résultats corrects et satisfaisants.

III.4. Présentation de l'interface utilisateur

Page d'authentification

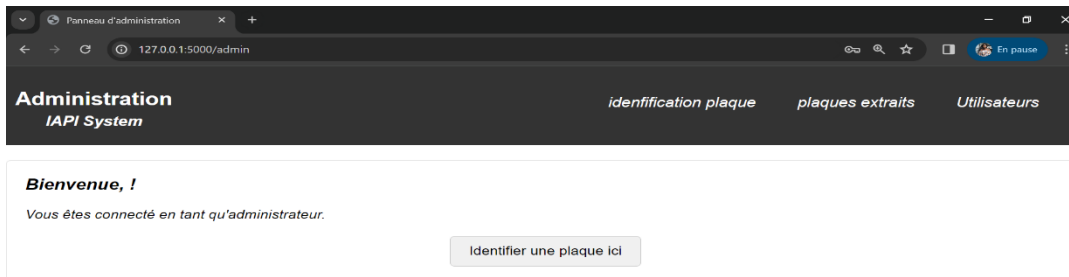


Connexion

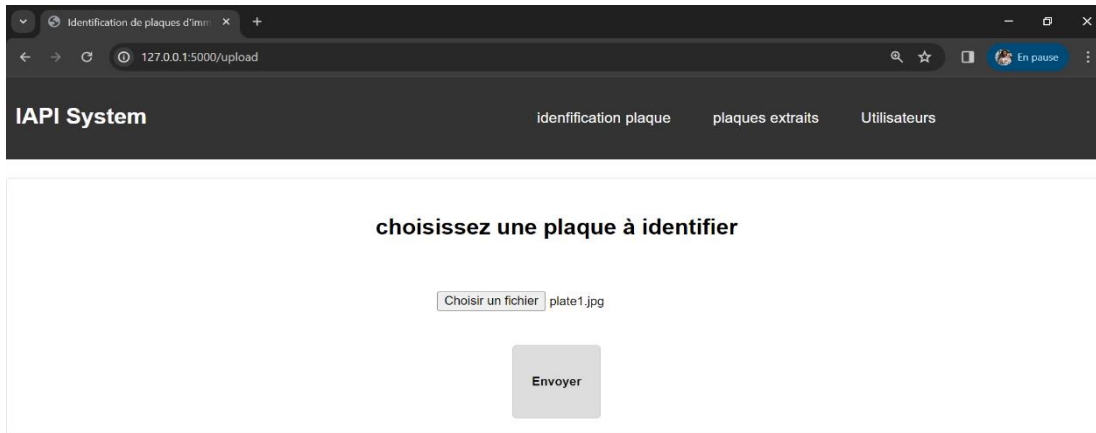
Nom d'utilisateur

Mot de passe

Se connecter



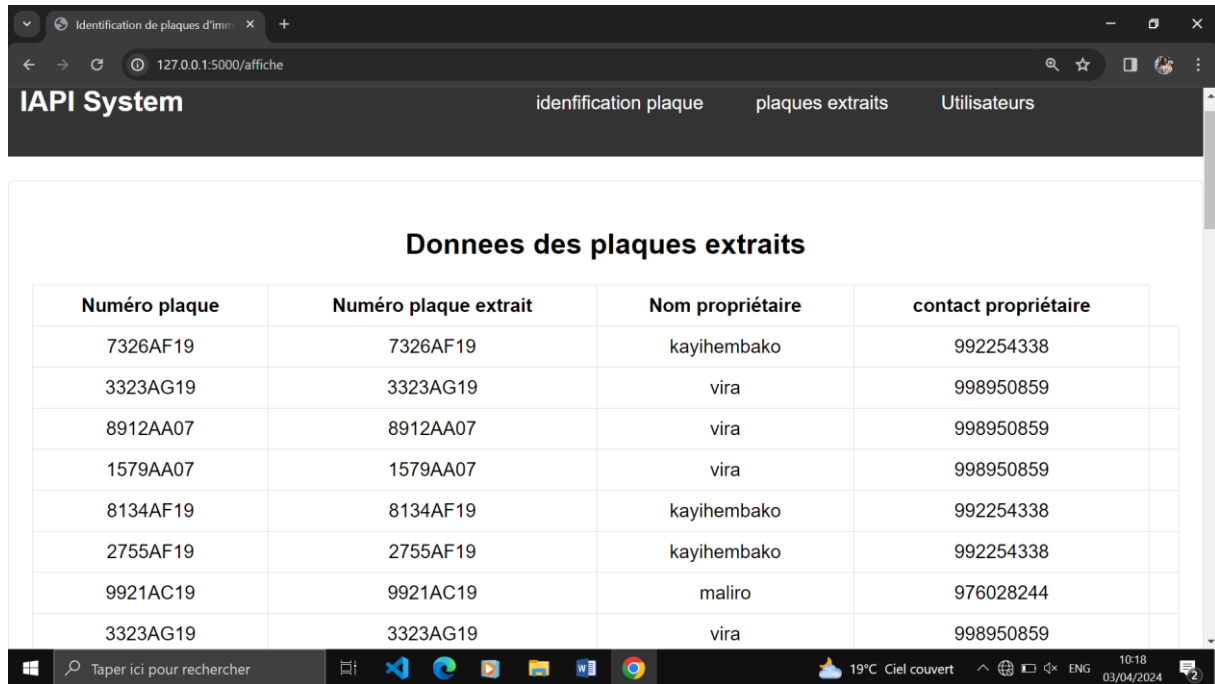
Ici on choisit une plaque à identifier puis on la passe au système d'identification pour son traitement



Puis après traitement, le système renvoie un message d'une identification réussi :

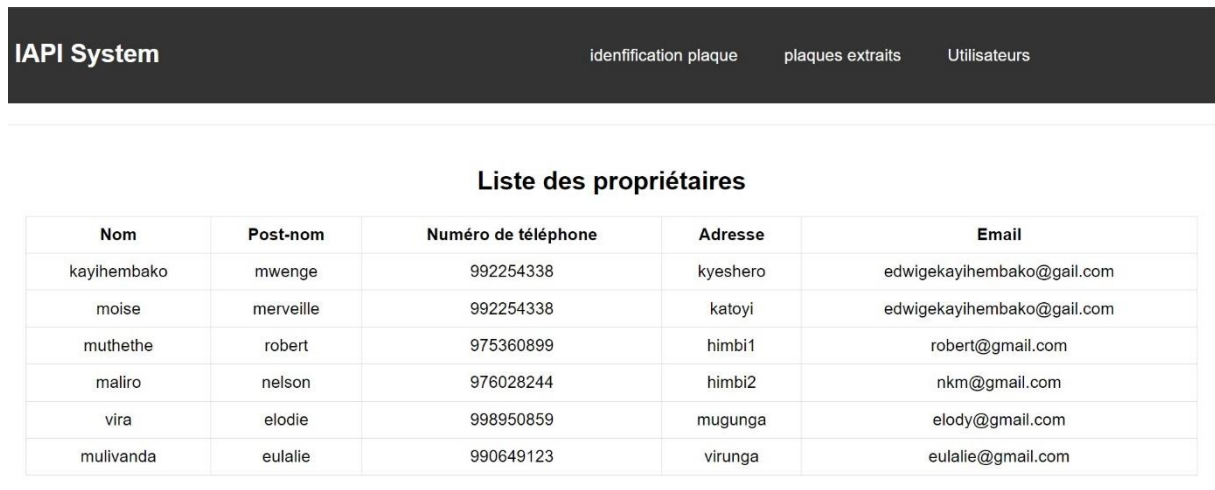


Visualisation des dernières plaques extraites



Numéro plaque	Numéro plaque extrait	Nom propriétaire	contact propriétaire
7326AF19	7326AF19	kayihembako	992254338
3323AG19	3323AG19	vira	998950859
8912AA07	8912AA07	vira	998950859
1579AA07	1579AA07	vira	998950859
8134AF19	8134AF19	kayihembako	992254338
2755AF19	2755AF19	kayihembako	992254338
9921AC19	9921AC19	maliro	976028244
3323AG19	3323AG19	vira	998950859

Visualisation des propriétaires des voitures



Nom	Post-nom	Numéro de téléphone	Adresse	Email
kayihembako	mwenge	992254338	kyeshero	edwigekayihembako@gail.com
moise	merveille	992254338	katoyi	edwigekayihembako@gail.com
muthethe	robert	975360899	himbi1	robert@gmail.com
maliro	nelson	976028244	himbi2	nkm@gmail.com
vira	elodie	998950859	mugunga	elody@gmail.com
mulivanda	eulalie	990649123	virunga	eulalie@gmail.com

III.5. Conclusion

Dans ce chapitre, on a détaillé le développement et les résultats d'un système de reconnaissance et d'identification des plaques d'immatriculation (I-API). Le système I-API est composé de plusieurs étapes, de l'acquisition de l'image à la vérification et l'identification de la plaque. Une base de données a été créée pour stocker les informations relatives aux plaques et aux

propriétaires. Le système a été testé sur un ensemble d'images et a montré des résultats prometteurs, avec une identification correcte dans la plupart des cas. Cependant, quelques erreurs ont été observées dans des conditions d'éclairage et de prise de vue non optimales. Des améliorations futures peuvent être apportées pour améliorer la performance et la fiabilité du système, notamment en utilisant des algorithmes plus robustes et en enrichissant la base de données.

CONCLUSION GENERALE

Dans ce mémoire, nous avons présenté un système d'identification des plaques d'immatriculation couvrant différentes étapes clés du processus. Nous avons débuté par une exploration des méthodes et des outils de traitement d'image essentiels pour nos programmes, incluant des techniques telles que la binarisation d'image, indispensables à la manipulation des données visuelles. Par la suite, l'accent a été mis sur l'extraction de la plaque d'immatriculation en utilisant une méthode de détection de contour, suivie de la segmentation des caractères de la plaque. Nous avons ensuite mis en œuvre une bibliothèque de reconnaissance optique de caractères (OCR), en l'occurrence pytesseract, pour la reconnaissance et l'extraction des caractères de la plaque, facilitant ainsi l'identification de cette dernière en comparaison avec les données de la base créée spécifiquement à cet effet. La présentation du système a été accompagnée d'une mise en œuvre détaillée et des résultats concluants des tests effectués, démontrant l'efficacité et la viabilité de l'approche adoptée. Néanmoins, des opportunités d'amélioration ont été identifiées et seront abordées dans les perspectives futures.

Perspectives :

- L'utilisation des techniques de prétraitements plus robustes pour avoir des résultats de traitement plus claires.
- Améliorer les résultats obtenus en analysant les cas de défaillances.
- Faire face au problème de luminosité et éclairage, notamment l'ombre et la réflexion de la lumière sur les matricules.
- Utilisation des méthodes plus performantes pour la localisation des plaques d'immatriculation.
- Il est intéressant d'intégrer le détecteur de plaque d'immatriculation dans un système embarqué à temps réel.
- Tester d'autres techniques de reconnaissance des chiffres tel que les réseaux de neurones.

BIBLIOGRAPHIE

- [1] A. BOUMIMEZ et M. DJIDJELI, SYSTEME DE RECONNAISSANCE AUTOMATIQUE DES PLAQUES D'IMMATRICULATION EN UTILISANT L'APPRENTISSAGE ARTIFICIEL, Algerie, 2020-2021.
- [2] <https://www.axis.com/fr-fr/solutions/license-plate-recognition#top>, "Reconnaissance automatisée des plaques d'immatriculation," 2023.
- [3] Y. BABA AHMED and A. BELAS, Reconnaissance automatique de plaque d'immatriculation de véhicules, 2021/2022.
- [4] "https://fr.wikipedia.org/wiki/Plaque_d%27immatriculation_en_R%C3%A9publique_d%C3%A9mocratique_du_Congo".
- [5] H. Abdelalim, Système de Détection et Reconnaissance Automatique des Plaques d'Immatriculation, maroc, 21/06/23.
- [6] K. Smail, Reconnaissance Automatique des plaques d'immatriculation (Implémentation sur Raspberry Pi), algerie, 2018-2019.
- [7] "Présentation PowerPoint (cdnsw.com)," [Online]. Available: Présentation PowerPoint (cdnsw.com).
- [8] G. Gabriel and J. Prateek, OpenCV 3.x with Python By Example Second Edition, Birmingham, 2018.
- [9] R. Smith, An Overview of the Tesseract OCR engine. In Document Analysis and, 2020.
- [1] M. N. V. Patrick, support du cours GENIE LOGICIEL, goma, 2023-2024.
0]
- [1] G. Swinnen, Apprendre à programmer avec Python, Paris: O'Reilly (Paris), 2000-2005 .
1]
- [1] K. M. Joel, Conception et mise en place d'un système de contrôle de la solvabilité des véhicules
2] basé sur la détection des plaques d'immatriculation, goma, 2022-2023.
- [1] C. @. 2. P. T. Inc., "Un guide complet de l'OCR avec Tesseract, OpenCV et Python," 5 decembre
3] 2019. [Online]. Available: <https://www.zephyrnet.com/fr/un-guide-complet-d%27ocr-avec-tesseract-opencv-et-python/>. [Accessed 19 janvier 2024].
- [1] K. A. B.V., "Tesseract OCR: Qu'est-ce que c'est, et pourquoi le choisir en 2024?," 9 janvier 2024.
4] [Online]. Available: <https://www.klippa.com/fr/blog/informations/quest-ce-que-tesseract-ocr/>. [Accessed 19 janvier 2024].

- [1] DataScientest, "Flask – Un des frameworks les plus populaires de Python," 22 novembre 2023.
5] [Online]. Available: <https://datascientest.com/avantages-et-fonctionnement-de-flask>. [Accessed 19 janvier 2024].
- [1] h. L, "Guide détaillé sur le framework Flask," 21 Aout 2023. [Online]. Available:
6] <https://blog.bbnd.eu/guide-detaillé-framework-flask-simplifiez-projets-web-python/>. [Accessed 19 janvier 2024].
- [1] B. L, "Qu'est-ce que SQL ? Tout savoir sur le langage des bases de données," 6 novembre 2023.
7] [Online]. Available: <https://www.lebigdata.fr/sql-tout-savoir-guide>. [Accessed 7 janvier 2024].
- [1] R. Peterson, "Qu'est-ce que SQL ? Apprenez les bases de SQL, le formulaire complet SQL et
8] comment l'utiliser," 6 decembre 2023. [Online]. Available: <https://www.guru99.com/fr/what-is-sql.html>. [Accessed 4 janvier 2024].
- [1] BDM, flavien chantrel, "Visual Studio Code," 2007. [Online]. Available:
9] <https://blogdumoderateur.com/>. [Accessed 18 decembre 2023].
- [2] [Online].
0]

ANNEXES

Code de fonctionnement du système

```

from flask import Flask, redirect, render_template, request, url_for
import cv2
import pytesseract
import numpy as np
import pymysql

app = Flask("Identification plaque")
pytesseract.pytesseract.tesseract_cmd = r'C://Program Files//Tesseract-OCR//tesseract.exe'
def allowed_file(filename):
    allowed_extensions = {'png', 'jpg', 'jpeg', 'gif'}
    return '.' in filename and filename.rsplit('.', 1)[1].lower() in
allowed_extensions
# Base de données d'utilisateurs factice
users = {"admin": {"password": "admin123"}}
# Fonction pour traiter l'image
def process_image(img):
    # code de traitement d'image ici
    plate_num = ""
    hierarchy=None

    character_count = 0
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    gray = cv2.resize(gray, None, fx=3, fy=3, interpolation=cv2.INTER_CUBIC)
    blur = cv2.GaussianBlur(gray, (5, 5), 0)
    gray = cv2.medianBlur(gray, 3)
    ret, thresh = cv2.threshold(blur, 0, 255, cv2.THRESH_OTSU |
cv2.THRESH_BINARY_INV)
    rect_kern = cv2.getStructuringElement(cv2.MORPH_RECT, (3,3))

    # apply dilation
    dilation = cv2.dilate(thresh, rect_kern, iterations = 1)
    # create copy of image
    im2 = dilation.copy()

    # find contours
    try:
        contours, hierarchy = cv2.findContours(im2, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
    except:

```

```

    ret_img, contours, hierarchy = cv2.findContours(im2, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)
    sorted_contours = sorted(contours, key=lambda ctr:
cv2.boundingRect(ctr)[0])

    photo_extrait=""
    # Boucle principale pour traiter les contours
    # cpt = 0
    for cnt in sorted_contours:

        # if cpt > 9:
        #     break
        x, y, w, h = cv2.boundingRect(cnt) # Récupération des coordonnées du
rectangle englobant
        height, width = im2.shape
        if height / float(h) > 4: continue
        ratio = h / float(w)
        # if height to width ratio is less than 1.5 skip
        if ratio < 1.5: continue
        area = h * w
        # if width is not more than 45 pixels skip
        if width / float(w) > 45: continue
        # if area is less than 100 pixels skip
        if area < 100: continue
        # draw the rectangle
        rect = cv2.rectangle(im2,(x,y), (x+w, y+h), (0,255,0),2)
        roi = thresh[y-5:y+h+5, x-5:x+w+5]
        roi = cv2.bitwise_not(roi)
        roi = cv2.medianBlur(roi, 5)
        roi = cv2.bitwise_not(roi)

        # Sauvegarde de La ROI
        cv2.imwrite(f"character{character_count}.jpg", roi)
        character_count += 1
        # Utilisation de Tesseract pour L'OCR
        text = pytesseract.image_to_string(roi, config='-c
tessedit_char_whitelist=0123456789ABCDEFGHIJKLMN0PQRSTUVWXYZ --psm 8 --oem 3')

        print(f" Contour : {len(contours)}")
        print(f" character_count : {character_count}")

        # Mise à jour du numéro de plaque
        plate_num += text.strip()

        print(f"Numéro de plaque : {plate_num}")
        # cpt += 1
    try:
        # connection à la base de données MySQL

```

```

mydb = pymysql.connect(
    host="localhost",
    user="root",
    password="",
    database="identification_plaque"
)

mycursor = mydb.cursor()
# Insérer L'enregistrement dans La base de données
mycursor.execute("INSERT INTO plaqueextraite (numero_plaque_extraite,
date_enregistrement) VALUES (%s, NOW())", (plate_num,))

mydb.commit()
mydb.close()

except Exception as e: print(f"Une erreur est survenue : {e}")
print(f"Numéro de plaque : {plate_num}")
# print(f" hierarchy {hierarchy}")
return plate_num
pass

def detect_and_correct_inversions(plate_num):
    window_size = 2
    for i in range(len(plate_num) - window_size + 1):
        window = plate_num[i:i+window_size]
        if window[0] > window[1]:
            plate_num = plate_num[:i] + window[::-1] +
plate_num[i+window_size:]
    return plate_num
@app.route('/',methods=['GET', 'POST'])
def ajout():
    return render_template('index.html')

@app.route('/insert',methods=['GET', 'POST'])
def insert():
    return render_template('insert.html')
@app.route('/upload', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        # Traitement de L'image après L'avoir reçue dans La requête POST
        # try:
        if 'file' not in request.files:
            return "Aucun fichier trouvé."

        file = request.files['file']

        if file.filename == '':
            return "Aucun fichier sélectionné."

```

```

    img = cv2.imdecode(np.frombuffer(file.read(), np.uint8),
cv2.IMREAD_COLOR)
    # print("+++++++ image :",img.shape)
    # Appel de la fonction process_image pour traiter l'image
    processed_result = process_image(img)

    plate_existance = check_plate(processed_result)

    print("+++++++ resultat ",plate_existance)
    try:
        mydb = pymysql.connect(
            host="localhost",
            user="root",
            password="",
            database="identification_plaque"
        )

        mycursor = mydb.cursor()

        # Exécutez la requête pour récupérer les informations du
propriétaire
        query = """
            SELECT p.nom, p.post_nom,mail, p.numero_telephon,
p.adresse,q.numero_plaque
            FROM proprietaireplaque p join plaqueannote q ON
p.id_proprietaire = q.id_proprietaire
            JOIN plaqueextrait pe ON q.numero_plaque =
pe.numero_plaque_extrait
            WHERE q.numero_plaque = %s
            limit 1;
        """
        mycursor.execute(query, (processed_result,))
        result = mycursor.fetchone()

        mycursor.close()
        mydb.close()

    except Exception as e:
        print(f"Erreur lors de la récupération des données du propriétaire
: {str(e)}")
        result = None

    # Affichage des résultats dans le template resultat.html
    if plate_existance == False :

```

```

        return render_template('resultat.html',
processed_result=plate_existance,error="Erreur lors de la récupération des
informations du propriétaire")
    else:
        return render_template('resultat.html',
processed_result=processed_result,result=result)
    # except Exception as e:
    #     return f"Erreur lors du traitement des données : {str(e)}"

    return render_template('insert.html')

@app.route('/affiche')
def affiche():
    try:
        mydb = pymysql.connect(
            host="localhost",
            user="root",
            password="",
            database="identification_plaque"
        )

        mycursor = mydb.cursor()

        # Exécutez la requête pour récupérer les informations nécessaires
        query = """
        SELECT pa.numero_plaque,
pe.numero_plaque_extrait,pa.id_proprietaire,pp.nom,pp.numero_telephon
FROM plaqueannote pa
JOIN proprietaireplaque pp ON pa.id_proprietaire = pp.id_proprietaire
JOIN plaqueextrait pe ON pa.numero_plaque = pe.numero_plaque_extrait

        """
        mycursor.execute(query)
        resultats = mycursor.fetchall()

        mycursor.close()
        mydb.close()

        return render_template('affiche.html', resultats=resultats)
    except Exception as e:
        print(f"Erreur lors de la récupération des données : {str(e)}")
        return render_template('affiche.html', error="Erreur de base de
données")

@app.route("/admin/index", methods=["GET", "POST"])
def index():
    if request.method == "POST":

```

```
username = request.form["username"]
password = request.form["password"]

if username in users and users[username]["password"] == password:
    # Authentification réussie
    return redirect(url_for("admin"))
else:
    # Erreur d'authentification
    return render_template("index.html", error="Nom d'utilisateur ou mot de
passe incorrect")

return render_template("insert.html")

@app.route("/admin")
def admin():
    # Vérifier si l'utilisateur est authentifié (session ou cookie)
    # Si non authentifié, rediriger vers la page de connexion
    return render_template("admin.html")
@app.route("/user")
def user():
    try:
        mydb = pymysql.connect(
            host="localhost",
            user="root",
            password="",
            database="identification_plaque"
        )

        mycursor = mydb.cursor()

        # Exécution de la requête pour récupérer Les données
        query = "SELECT nom, post_nom, numero_telephon, adresse, mail FROM
proprietaireplaque"
        mycursor.execute(query)
        resultats = mycursor.fetchall()

        mycursor.close()
        mydb.close()

        return render_template('user.html', resultats=resultats)
    except Exception as e:
        print(f"Erreur lors de la récupération des données : {str(e)}")
        return render_template('user.html', error="Erreur de base de données")

def check_plate(plate_text):
    try:
        mydb = pymysql.connect(
            host="localhost",
```

```
        user="root",
        password="",
        database="identification_plaque"
    )

    mycursor = mydb.cursor()

    # Exécution de la requête pour récupérer Les données
    # query = "SELECT COUNT(*) FROM plaqueannotate WHERE numero_plaque =
plate_text "
    # print(f"+++++++= query {query}")
    # mycursor.execute(query)
    # result = mycursor.fetchall()

    plate = plate_text
    query = "SELECT count(*) FROM plaqueannotate WHERE numero_plaque =
%s" # Remplacer colonne_contenant_valeurs par Le nom de la colonne dans la
table plaqueannotate
    mycursor.execute(query, plate) # row[0] correspond à la valeur de la
colonne identification_plaque de la table plaqueextract
    result = mycursor.fetchall()
    result = result[0][0]

    mycursor.close()
    mydb.close()

    if result >=1:
        return True
    else:
        return False

    except Exception as e:
        print(f"Erreur lors de la récupération des données : {str(e)}")
        return render_template('user.html', error="Erreur de base de données")

if __name__ == '__main__':
    app.run(debug=True)
```