

**Spécialité :** Licence Professionnelle en Infrastructure, analyse et traitement des données massives (BIG DATA)

Rapport de Stage de Fin d’Etude

Sous le thème

–

Effectué par : Oussama NAJEM Encadré par :

Période de stage : du 01/04/2024 au 01/06/2024

Présenté et soutenu le 09/06/2023, devant le Jury composé de :

Pr. Président

Pr. Examinateur

Lieu de stage : Ecole 1337 centre de formation Khouribga

Année universitaire : 2023-2024

*Remerciements*

*Je désire exprimer ma profonde gratitude à Dieu, le Tout-Puissant et le Miséricordieux, pour m'avoir accordé la force et la patience nécessaires pour accomplir ce modeste travail.*

*En second lieu, je tiens à exprimer ma sincère reconnaissance envers mon encadrant, Monsieur SABANI , pour ses précieux conseils et son soutien inestimable tout au long de cette période de travail.*

*Mes remerciements les plus vifs vont également aux membres du jury pour l'attention qu'ils ont portée à notre recherche en acceptant d'examiner notre travail et en le nourrissant de leurs suggestions constructives.*

*Je tiens également à exprimer ma profonde gratitude envers Monsieur le Directeur de l'EST, ainsi qu'à tous les professeurs et dirigeants de cette prestigieuse institution, pour leur dévouement continu à notre éducation et à notre développement.*

*Je souhaite également adresser mes sincères remerciements à toutes les personnes qui ont apporté leur aide et contribué à l'élaboration de ce mémoire, ainsi qu'à la réussite de cette année universitaire exceptionnelle.*

*À nos familles et à nos amis, je suis infiniment reconnaissant pour leurs prières et leurs encouragements constants, qui ont été une source de force et de motivation pour surmonter tous les défis.*

*Enfin, je voudrais exprimer ma reconnaissance à toutes les personnes, qu'elles aient été directement impliquées ou non, qui ont contribué, de près ou de loin, à la réalisation de ce travail.*

Table des matières

[*Remerciements* 2](#_Toc167545478)

[Content 3](#_Toc167545479)

[List des figures 4](#_Toc167545480)

[Résumé 5](#_Toc167545481)

[INTRODUCTION : 6](#_Toc167545482)

[I. CHAPITRE 1 : Context Générale 7](#_Toc167545483)

[1. Introduction : 7](#_Toc167545484)

[2. Présentation de l’organisation d’accueil: 7](#_Toc167545485)

[1) Présenta<tion école 1337 : 7](#_Toc167545486)

[2) Rôle et activités : 8](#_Toc167545487)

[3) Les espaces de l’école 1337 : 9](#_Toc167545488)

[4) Le système de formation de 1337 : 10](#_Toc167545489)

[5) Pour rejoindre l’école : 11](#_Toc167545490)

[6) Les partenaires de 1337 : 11](#_Toc167545491)

[7) Les débouchés professionnels : 11](#_Toc167545492)

[3. Présentation de projet 12](#_Toc167545493)

[1) Cadre de projet et problématique 12](#_Toc167545494)

[2) Objectifs de projet 13](#_Toc167545495)

[3) Démarche du projet 13](#_Toc167545496)

[4. Conclusion 13](#_Toc167545497)

[II. CHAPITRE 2 : Présentation des concepts 14](#_Toc167545498)

[1. Introduction 14](#_Toc167545499)

[2. Introduction au Machine Learning et Deep Learning 14](#_Toc167545500)

[3. YOLO (You only Look Once): 16](#_Toc167545501)

[4. SIFT (Scale-Invariant Feature Transform) 18](#_Toc167545502)

[5. EasyOCR 20](#_Toc167545503)

List des figures

[Figure 1: image extérieure de l337. 8](#_Toc167892281)

[Figure 2: salle de classe 9](#_Toc167892282)

[Figure 3: espace de travail collaboratif. 10](#_Toc167892283)

[Figure 4: espace détente. 10](#_Toc167892284)

[Figure 5: chaine de traitement 13](#_Toc167892285)

[Figure 6: l'architecture de Artificial Neural network 15](#_Toc167892286)

[Figure 7: l'architecture de Recurrent nueral network 15](#_Toc167892287)

[Figure 8 : CNN architecture 16](#_Toc167892288)

[Figure 9: Architecture du modèle Yolo, opérant un quadrillage à partir de convolutions successives. 18](#_Toc167892289)

[Figure 10: Localisation des objets 18](#_Toc167892290)

[Figure 11: Image quadrillée. 18](#_Toc167892291)

[Figure 12 : Carte de probabilité des classes 19](#_Toc167892292)

[Figure 13: Image de sortie post-NMS présentant les objets détectés par Yolo 19](#_Toc167892293)

[Figure 14: les differents echelles de fonction gaussienne 20](#_Toc167892294)

[Figure 15: maxima et minima points 21](#_Toc167892295)

[Figure 16: affichage des points détectes 21](#_Toc167892296)

[Figure 17 : Diagramme de cas d'utilisation 24](#_Toc167892297)

[Figure 18 : diagramme de classe 25](#_Toc167892298)

[Figure 19 : diagramme de séquence 26](#_Toc167892299)

[Figure 20 : react native logo 27](#_Toc167892300)

[Figure 21 : npm logo 28](#_Toc167892301)

[Figure 22: expo logo 28](#_Toc167892302)

[Figure 23 : vscode logo 28](#_Toc167892303)

[Figure 24 : python logo 29](#_Toc167892304)

[Figure 25 : firebase logo 29](#_Toc167892305)

[Figure 26 : flask logo 29](#_Toc167892306)

[Figure 27 : openCV logo 30](#_Toc167892307)

[Figure 28 : numPy logo 30](#_Toc167892308)

[Figure 29 : Splash Screen 31](#_Toc167892309)

[Figure 30 : Interface principale 32](#_Toc167892310)

Résumé

Cette étude offre une analyse exhaustive de la mise en œuvre d'un modèle de détection et de classification utilisant YOLO (You Only Look Once) et la reconnaissance optique de caractères (OCR). En s'appuyant sur des techniques avancées de vision par ordinateur et d'apprentissage profond, notre recherche explore la pertinence de diverses approches pour la détection d'objets et l'extraction automatique de texte, notamment l'utilisation de modèles de détection et de bibliothèques OCR et OpenCV.

Nous examinons également les défis spécifiques rencontrés dans la détection et la classification des objets, ainsi que dans l'extraction des informations textuelles sur des documents d'identité, tels que les différences de qualité d'image, la complexité des arrière-plans, et la variabilité des polices et des formats.

En utilisant un ensemble diversifié de données d'images comprenant des documents d'identité, nous évaluons les performances des modèles YOLO pour la détection d'objets et des algorithmes OCR pour la reconnaissance de texte. Nous discutons également des ajustements et des optimisations nécessaires pour améliorer la précision et la rapidité de ces modèles dans des applications réelles.

Les résultats de cette étude démontrent le potentiel significatif de la combinaison de YOLO et de l'OCR pour améliorer l'efficacité et la précision des tâches de détection et de classification dans divers contextes.

INTRODUCTION :

Mon stage à l'école 1337 à Khouribga a été une immersion enrichissante dans le monde fascinant de l'intelligence artificielle, avec un accent particulier sur le développement d'une application novatrice de détection et de classification utilisant YOLO et la reconnaissance optique de caractères (OCR). Pendant cette période, j'ai eu l'opportunité de mettre en pratique mes connaissances théoriques et d'explorer de nouvelles technologies telles que le Machine Learning (ML), le Deep Learning (DL) et la vision par ordinateur.

Ce rapport est dédié à mon expérience au sein de ce projet ambitieux, visant à fournir une solution efficace pour automatiser la détection d'objets et l'extraction d'informations textuelles à partir d'images. Je vais partager les détails de ma contribution à ce projet, ainsi que les enseignements tirés de cette expérience immersive.

Dans les prochains chapitres, je vais décrire l'école 1337 et son environnement propice à l'innovation et à l'apprentissage dans le domaine de l'intelligence artificielle. Ensuite, je vais me concentrer sur la description du projet, en mettant en lumière son objectif principal et les technologies utilisées pour le réaliser.

Le cœur de ce rapport sera consacré à la conception et à la réalisation de l'application de détection et de classification. Je vais détailler les différentes étapes du processus de développement, en mettant l'accent sur les techniques de vision par ordinateur (EasyOCR) et de Deep Learning utilisées pour entraîner les modèles YOLO .

Enfin, je conclurai en présentant l'environnement de travail dans lequel j'ai évolué, en détaillant les données utilisées, les méthodes et les outils qui ont facilité la mise en œuvre de notre solution d'intelligence artificielle. Ce stage m'a offert une expérience précieuse dans le domaine en pleine expansion de l'IA et a renforcé mon intérêt pour la recherche et le développement de technologies innovantes dans ce domaine.

# **CHAPITRE 1 :** Contexte Générale

## Introduction :

Ce chapitre offre une présentation générale de l'école 1337, en explorant son histoire, sa mission et sa vision. Nous détaillerons les principes pédagogiques qui sous-tendent son approche éducative, ainsi que les différentes infrastructures et ressources mises à disposition des étudiants pour favoriser un environnement d’apprentissage optimal.

Nous fournissons également une description complète du projet.

## Présentation de l’organisation d’accueil :

### Présentation école 1337 :

L'école 1337, également connue sous le nom de "école 42", est une école de programmation française basée sur le concept de l'apprentissage par les pairs. Fondée en 2013 par Xavier Niel, l'école 1337 n'a pas de professeurs, de programmes de cours ou de diplômes traditionnels. Au lieu de cela, les étudiants travaillent sur des projets pratiques en groupe, en utilisant une méthode de pédagogie inversée où ils doivent trouver des solutions à des problèmes par eux-mêmes et en s'entraidant.

L'école 1337 se concentre sur l'acquisition de compétences pratiques et de savoir-faire techniques dans le domaine de la programmation, en mettant l'accent sur la collaboration, l'apprentissage autonome et la résolution de problèmes. Les étudiants sont sélectionnés sur la base d'un processus de sélection rigoureux, qui se déroule en ligne et sur site, et qui évalue leur motivation, leur capacité à travailler en équipe et leur aptitude à la programmation.

****

Figure 1: image extérieure de l337.

### Rôle et activités :

Le rôle principal de l'école 1337 de Khouribga est de fournir une formation de qualité en programmation à des jeunes Marocains, en particulier ceux qui n'ont pas accès à une formation traditionnelle en informatique. Elle a pour objectif de développer les compétences techniques, la créativité, la collaboration et la résolution de problèmes chez les étudiants, en vue de les préparer à des carrières dans le domaine de la technologie.

Les activités de l'école 1337 de Khouribga incluent la formation en programmation, l'organisation de projets collaboratifs et la promotion de la communauté technologique locale. Les étudiants de l'école travaillent sur des projets pratiques en groupe, allant de la création de jeux vidéo à la conception de sites web en passant par la programmation de robots.

****En résumé, l'école 1337 de Khouribga joue un rôle important dans la promotion des compétences techniques et la création d'opportunités d'emploi dans la région. Elle est un exemple de réussite en matière d'éducation technologique accessible et innovante au Maroc.

Figure 2: salle de classe

### Les espaces de l’école 1337 :

L'école 1337 de Khouribga dispose d'espaces dédiés pour l'apprentissage de laprogrammation et pour les projets collaboratifs des étudiants.

Les espaces comprennent :

**La salle de classe :** C'est l'espace principal où les étudiants assistent à des cours magistraux et des ateliers dirigés par les tuteurs. Elle est équipée de tables, dechaises et d'équipements audiovisuels pour les présentations.

**L'espace de travail collaboratif :** Cet espace est destiné aux projets collaboratifs des étudiants. Il est équipé de tables et de chaises pour les groupes de travail, ainsi que d'ordinateurs et de matériel de programmation pour les projets en cours.

Figure 3: espace de travail collaboratif.

**L'espace détente :** C'est un espace où les étudiants peuvent se détendre, discuter prendre une pause. Il est équipé de canapés, de tables et de chaises, ainsi que de jeux de société et d'autres activités pour se divertir.

Figure 4: espace détente.

En plus de ces espaces, l'école 1337 de Khouribga dispose également d'une cuisine pour les repas des étudiants, ainsi que de toilettes et de douches pour le confort des étudiants qui passent de longues heures à l'école.

### Le système de formation de 1337 :

Le système de formation de 1337 est basé sur l'apprentissage par projet et l'autonomie. Les étudiants sont encouragés à travailler en groupe sur des projets pratiques qui les obligent à apprendre par eux-mêmes. Chaque projet est conçu pour développer des compétences spécifiques, telles que la résolution de problèmes, la collaboration, la communication et la créativité. Les étudiants ne sont pas notés sur leurs projets, mais sur leur capacité à travailler en groupe, à respecter les délais et à atteindre les objectifs du projet.

### Pour rejoindre l’école :

Pour accéder à cette école, qui ne demande ni diplôme ni niveau scolaire spécifique et ni frais, les intéressés doivent passer par trois étapes : deux tests évaluant la mémoire et la logique, un déplacement au campus 1337 ou en ligne concerné pour une session de présentation de l’école et enfin le teste dit de «la piscine ».

Cette dernière étape est d’une durée d’un mois, au cours de laquelle les candidats baignent dans l’environnement et la philosophie de l’école, et où ils expérimentent la pédagogie Peer to Peer, c'est à dire qu'ils apprennent et enseignent les uns aux autres, tout en passant par des évaluations régulières.

### Les partenaires de 1337 :

1337 bénéficie de la collaboration de nombreux partenaires privés et publics qui contribuent au financement de ses activités et à l'amélioration de son offre de formation.

Parmi ces partenaires, on peut mentionner :

1- OCP Group.

2- Des géants de la technologie tels que Google, Microsoft et IBM.

3- Des organisations internationales de renom telles que la Banque mondiale et la Fondation Rockefeller.

### Les débouchés professionnels :

1337 est fière de ses résultats en matière d'employabilité de ses diplômés. Les étudiants de 1337 ont la possibilité de travailler pour des entreprises locales et internationales de renom, ou de créer leur propre startup. Les étudiants de 1337 ont également été recrutés par des entreprises technologiques telles que Google, Microsoft, IBM, Amazon, Oracle, et bien d'autres. Grâce à son approche innovante de l'éducation et à ses partenariats solides avec des entreprises technologiques, 1337 est en train de transformer le paysage de l'éducation en informatique au Maroc et à l'étranger.

Les campus de 1337 :

**Khouribga :** Située à environ 1,5 heure de route de Casablanca et à 3 heures de Marrakech. Khouribga est une jolie petite ville. Treize, Trente-Sept est situé au cœur de Mail Central, un endroit idéal pour étudier car il est entouré de parcs ainsi que de plusieurs infrastructures culturelles et sportives.

Les équipements de l'école incluent :

* 300 iMac.
* 2 aires de jeux.
* 8 salles de réunion.

**Ben Guérir :** Le campus de Ben Guérir offrira des opportunités supplémentaires aux jeunes marocains et africains subsahariens pour évoluer dans un environnement propice à l'innovation et au Peer Learning pour les professions du futur.

Les équipements de l'école incluent :

* 300 iMac.
* 2 terrains de jeu.
* 8 salles de réunion

**Med :** À une heure de Tanger et près de Cabo Negro, campus : 1337 MED, situé à Tetouan Shore, en plein centre d'un écosystème unique aux côtés d'entreprises d'offshoring et de centres éducatifs. 1337 Med offre le meilleur environnement pour l'apprentissage et l'épanouissement, avec de nombreux espaces verts et des infrastructures sportives étendues sur 12 000 m2.

* 210 iMac.
* 3 terrains de jeu.
* 4 salles de réunion.

## Présentation de projet

#### Cadre de projet et problématique

Dans le cadre de mon stage de fin d'études en Infrastructure, analyse et traitement des données massives (Big Data), j'ai entrepris un projet visant à améliorer le processus d'inscription des étudiants à l'école 1337 à Khouribga durant le début du période du piscine. L’inscription manuelle des informations à partir des cartes nationales des étudiants est sujette à des erreurs. Pour répondre à ce problème, le projet se concentre sur le développement d'un système intégrer dans une application mobile capable d'automatiser cette tâche en utilisant des techniques de machine learning et vision par ordinateur et de reconnaissance optique de caractères (OCR).

#### Objectifs de projet

* **Automatisation** : Réduire le temps nécessaire pour l'inscription des étudiants en automatisant l'extraction des informations des cartes nationales.
* **Précision** : Minimiser les erreurs humaines dans le processus de saisie des données.
* **Efficacité** : Améliorer l'efficacité globale du processus d'inscription.
* **Interactivité**: Créer une application mobile interactif avec l’utilisateur.

## Conclusion

Ce chapitre a fourni un aperçu de l’organisation d'accueil et de la description du projet de stage, établissant ainsi les bases nécessaires pour la suite de notre étude. Maintenant, nous allons maintenant passer à l'étape de présentation les concepts utilisés pour réaliser un environnement de développement approprié.

# CHAPITRE 2 : Etat de l’art

##### Introduction

Dans ce chapitre, nous explorerons des concepts et des algorithmes fondamentaux de la vision par ordinateur et de l'intelligence artificielle. Ces techniques jouent un rôle crucial dans de nombreuses applications modernes.

##### Introduction au Machine Learning et Deep Learning

1. Machine learning

**Le machine learning** (apprentissage automatique) est un domaine de l'intelligence artificielle qui permet aux systèmes informatiques d'apprendre à partir de données et de prendre des décisions ou des prédictions basées sur cette expérience. Plutôt que d'être explicitement programmés pour effectuer une tâche, les systèmes de machine learning sont entraînés à partir d'un ensemble de données d'entraînement, ce qui leur permet de s'améliorer au fil du temps. Les principaux types de machine learning incluent :

* Apprentissage Supervisé : L'algorithme apprend à partir de données d'entraînement étiquetées, où chaque exemple est associé à une réponse correcte.
* Apprentissage Non Supervisé : L'algorithme apprend à identifier des motifs dans des données sans réponses étiquetées.
* Apprentissage Semi-Supervisé : Combinaison des deux, où l'algorithme apprend à partir d'un petit ensemble de données étiquetées et d'un plus grand ensemble de données non étiquetées.

1. Deep learning

**Le Deep Learning (DL)** est une sous-catégorie du Machine Learning qui utilise des réseaux de neurones artificiels (Artificial Neural network) à plusieurs couches pour apprendre des représentations de données de plus en plus abstraites. Ces réseaux de neurones sont inspirés du fonctionnement du cerveau humain et sont capables d'apprendre des caractéristiques hiérarchiques à partir de données non structurées telles que des images, du texte ou du son. Le Deep Learning est particulièrement efficace pour des tâches telles que la reconnaissance d'images et vocale, la traduction automatique, la génération de texte, et bien d'autres.

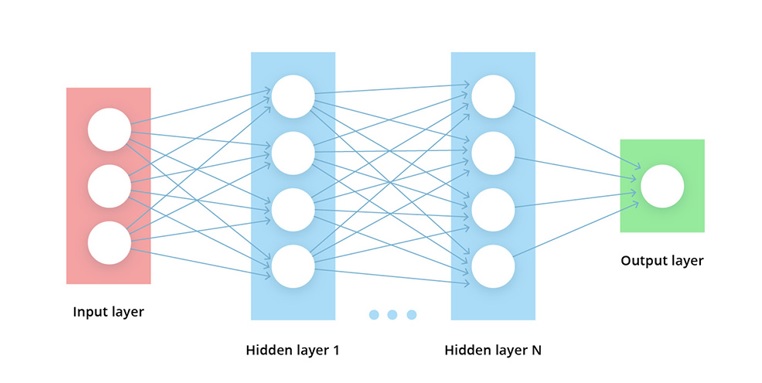


Figure 5: l'architecture de Artificial Neural network

1. Recurrent Neural Network (RNN)

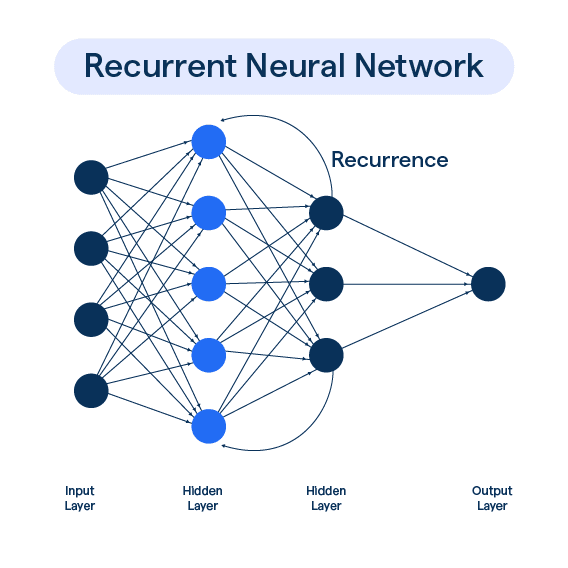
Les réseaux de neurones récurrents sont conçus pour comprendre les données temporelles ou séquentielles. Les RNN améliorent leurs prédictions en utilisant des points de données supplémentaires dans une séquence. Pour modifier la sortie, ils prennent en entrée et réutilisent les activations des nœuds antérieurs ou postérieurs dans la séquence.

Figure 6: l'architecture de Recurrent nueral network

1. Convolutional Neural Network (CNN)

Un réseau de neurones convolutifs est un modèle de machine learning inspiré par la structure et le fonctionnement du cortex visuel animal, conçu pour traiter des données structurées sous forme de grille, telles que les images. Les CNN sont composés de plusieurs couches, chacune ayant une fonction spécifique.

 **Couches Convolutives (Convolutional Layers)** :

* Ces couches appliquent des filtres (ou noyaux) sur les entrées pour produire des cartes de caractéristiques (feature maps). Chaque filtre détecte des caractéristiques spécifiques telles que les bords, les textures, etc.

 **Couches de Pooling (Pooling Layers)** :

* Ces couches réduisent la dimensionnalité des cartes de caractéristiques, généralement en utilisant des opérations comme le max-pooling ou l'average-pooling, tout en conservant les informations les plus importantes.
  + **Max Pooling**: Dans ce type de pooling, la carte de caractéristiques d'entrée est divisée en régions rectangulaires non chevauchantes, et la valeur maximale dans chaque région est sélectionnée comme valeur représentative pour cette région.
  + **Average Pooling**: Contrairement au max pooling, l'average pooling calcule la valeur moyenne dans chaque région de la carte de caractéristiques d'entrée. Il peut être utile lorsque vous souhaitez obtenir une représentation plus générale des caractéristiques sans vous concentrer uniquement sur les valeurs les plus dominantes

 **Couches Entièrement Connectées (Fully Connected Layers)** :

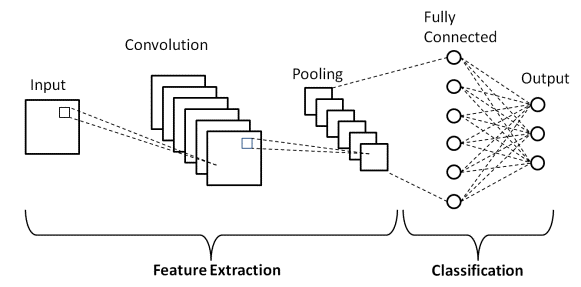
* Vers la fin du réseau, une ou plusieurs couches entièrement connectées (similaires aux réseaux de neurones classiques) sont utilisées pour effectuer la classification ou d'autres tâches de sortie.

Figure 7 : CNN architecture

1. Importance du ML et du DL

Le Machine Learning (ML) et le Deep Learning (DL) révolutionnent l'analyse des images et l'intelligence artificielle (IA) en offrant des capacités d'apprentissage et d'analyse sans précédent. Leur impact se manifeste dans divers aspects :

1. Extraction d'informations précieuses :

**Découverte de modèles cachés :** ML et DL permettent de détecter des patterns et des tendances subtiles dans les données, qui seraient invisibles à l'analyse humaine. Cela conduit à une meilleure compréhension des phénomènes et à des décisions plus éclairées.

**Analyse de données complexes :** Ces technologies s'attaquent aux données non structurées ou semi-structurées, comme le texte, les images et les vidéos, qui étaient auparavant difficiles à analyser.

1. Prédictions précises et automatisation :

**Prévisions futures :** ML et DL permettent de prédire des événements futurs, tels que la demande des consommateurs, les pannes d'équipement ou les tendances du marché, avec une précision accrue.

**Automatisation des tâches :** Les algorithmes peuvent automatiser des tâches répétitives et fastidieuses, comme dans notre cas la détection des cartes nationales et extraction du texte à partir d’une image, libérant du temps et des ressources pour des activités plus stratégiques.

##### Computer vision

1. YOLO (You only Look Once):

YOLO est un modèle de détection d'objets développé par Joseph Redmon. Il se distingue des autres méthodes de détection par son approche unique consistant à traiter l'image entière en une seule étape, plutôt que de diviser l'image en parties ou d'utiliser des fenêtres glissantes. L’objectif de la détection d’objet est de classifier de manière automatique.

1. Comment YOLO fonctionne :

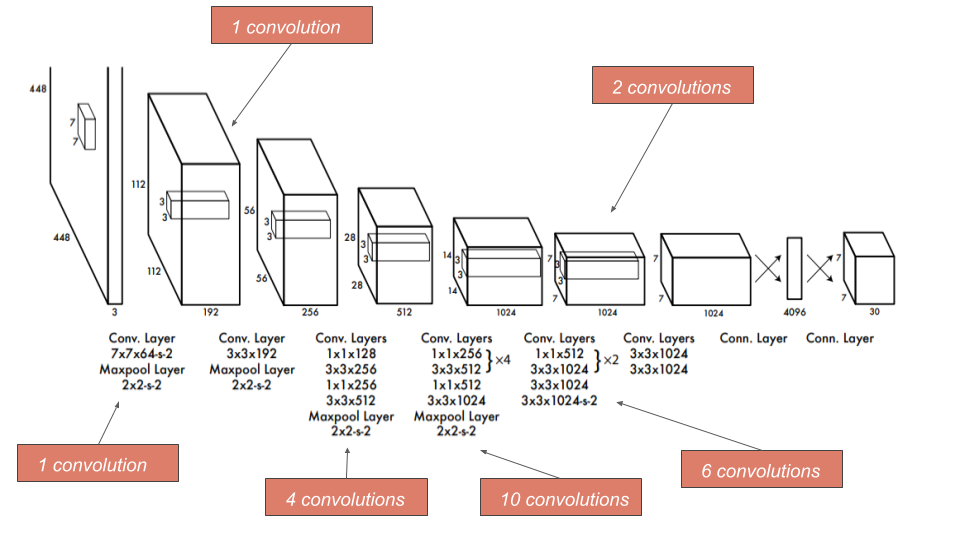
Les couches convolutives sont la pierre angulaire de YOLO. Elles permettent d'extraire des caractéristiques locales de l'image en appliquant des filtres convolutifs. Ces couches détectent des motifs comme les bords, les textures, et les formes simples, qui sont ensuite combinés pour

Figure 8: Architecture du modèle Yolo, opérant un quadrillage à partir de convolutions successives.

YOLO divise l'image d'entrée en une grille S×S comme illustrée dans la figure 10. Chaque cellule de la grille est responsable de prédire des boîtes englobantes (bounding boxes) et des probabilités de classes pour les objets dont le centre se trouve dans cette cellule.

Dans un premier temps, Yolo identifie tous les objets présents à l’aide de cadres en leur associant un degré de confiance (illustrée dans la figure 9).

Puis, l’algorithme attribue une classe à chaque boîte selon l’objet qu’il pense avoir détecté à partir de la carte de probabilité



Figure 9: Localisation des objets

Figure 10: Image quadrillée.

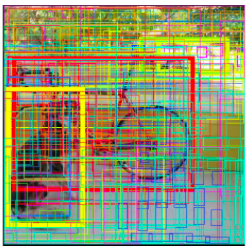
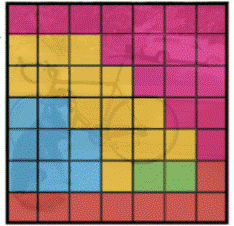


Figure 11: Détection des objets

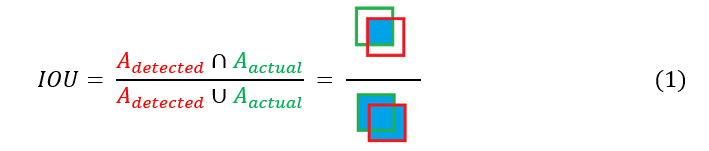
Figure 12 : Carte de probabilité des classes

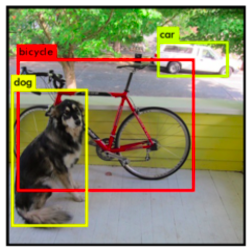
Enfin, Yolo supprime toutes les boîtes superflues à l’aide de la méthode NMS.

La méthode NMS(N**on-**M**axima**S**uppression**) se base sur un parcours des boîtes à haut indice de confiance, puis une suppression des boîtes superposées à celles-là en mesurant l’**IoU**. Pour cela, on suit 4 étapes. En partant de la liste complète des boîtes détectées :

1. Suppression de toutes les boîtes d’indice de confiance trop faible.
2. Identification de la boîte d’indice de confiance le plus grand.
3. Suppression de toutes les boîtes ayant un IoU trop grand (c’est-à-dire de toutes les boîtes trop similaires à notre boîte référence).
4. En ignorant la boîte de référence ainsi utilisée, répétition des étapes 2) et 3) jusqu’à avoir éliminé toutes les boîtes de notre liste originale (c’est-à-dire en prenant la 2nde boîte d’indice de confiance le plus grand, puis la 3ème, etc.)

**IoU** Intersection over Union est une métrique permettant de mesurer la précision de la localisation d’un objet. Comme son nom l’indique, elle est calculée à partir du ratio entre la zone d’intersection Objet détecté-Objet réel et de la zone d’union de ces mêmes objets (cf. quation 1). En notant Adétecté et Aréel les aires respectives de l’objet détecté par YOLO et de l’objet tel que réellement situé sur l’image,

****



On obtient alors le résultat suivant :

Figure 13: Image de sortie post-NMS présentant les objets détectés par Yolo

##### SIFT (Scale-Invariant Feature Transform)

Le SIFT (Scale-Invariant Feature Transform) est un algorithme développé par David Lowe en 1999 pour détecter et décrire les points d'intérêt dans une image. Il est largement utilisé en vision par ordinateur pour des tâches telles que la reconnaissance d'objets, la correspondance d'images et la reconstruction 3D. Le principal avantage de SIFT est sa robustesse face aux changements d'échelle, de rotation, et de luminosité.

1. **Comment SIFT fonctionne**
2. Espace d'échelle : L'image est convenue à différentes échelles en appliquant une fonction gaussienne qui rend l’image du couleur grise avec différents paramètres de sigma (Octave).

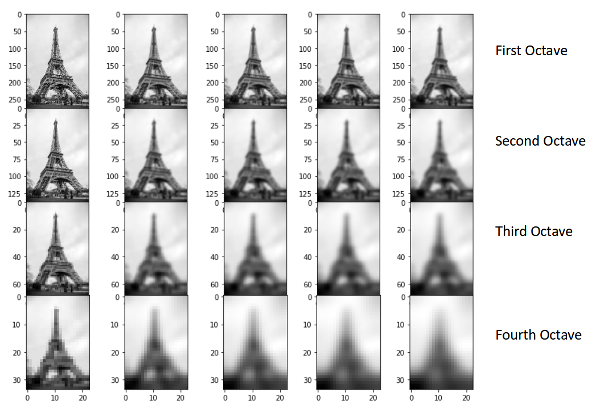


Figure 14: les differents echelles de fonction gaussienne

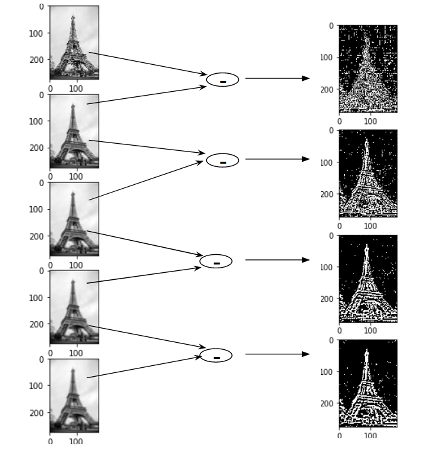
1. Extrema locaux : Pour chaque octave de l'espace d'échelle, des points extrêmes locaux (maxima et minima) sont détectés dans une différence de gaussiennes (DoG). Ces points sont potentiellement des points d'intérêt.

Figure 15: maxima et minima points

1. Les points clés détectés sont affinés en supprimant les points instables et les bords. Cette étape améliore la robustesse et réduit le nombre de faux positifs.

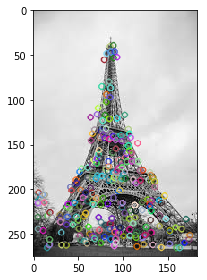


Figure 16: affichage des points détectes

##### EasyOCR

EasyOCR est une bibliothèque de reconnaissance optique de caractères (OCR) qui permet d'extraire du texte à partir d'images. Développée par Jaided AI, elle est conçue pour être facile à utiliser et prend en charge plus de 80 langues.

1. **Fonctionnement de EasyOCR**

Le processus de fonctionnement de EasyOCR peut être simplifié en trois étapes principales : détection de texte, reconnaissance de texte et post-traitement.

1. **Détection de Texte**

La première étape du processus consiste à détecter les zones de texte dans une image. EasyOCR utilise un modèle basé sur des réseaux de neurones convolutifs (CNN) pour localiser les régions de l'image qui contiennent du texte. Le modèle de détection de texte est entraîné pour identifier des boîtes englobantes autour des zones de texte.

1. **Reconnaissance de Texte**

Une fois les zones de texte détectées, EasyOCR utilise un modèle de reconnaissance pour lire et interpréter le texte dans chaque boîte englobante. Cette étape implique de convertir les pixels de la région de texte en caractères lisibles.

* Transformation d'Image : Les images des boîtes englobantes sont transformées pour être adaptées à la taille d'entrée du modèle de reconnaissance.
* Interprétation des Caractères : Utilisation de réseaux neuronaux récurrents (RNN) pour interpréter et convertir les images de texte en séquences de caractères.

1. **Post-Traitement**

Après la reconnaissance initiale du texte, EasyOCR applique plusieurs techniques de post-traitement pour améliorer la précision des résultats et les formater de manière appropriée.

* Correction d'Erreurs : Application de règles de correction d'erreurs basées sur le contexte pour corriger les erreurs courantes de reconnaissance.
* Formatage du Texte : Organisation du texte reconnu en fonction de la disposition initiale dans l'image, y compris le respect des lignes, paragraphes et colonnes.

##### K-Nearest Neighbors (KNN)

L'algorithme K-Nearest Neighbors (KNN) est une méthode de classification non paramétrique couramment utilisée dans le domaine de l'apprentissage automatique. Son principe repose sur la similarité entre les données pour effectuer des prédictions.il est utilisée pour résoudre les problèmes de classification et de régression.

L'algorithme K-NN fonctionne en trouvant les K voisins les plus proches d'un point de données donné sur la base d'une métrique de distance, telle que la distance euclidienne. La classe ou la valeur du point de données est ensuite déterminée par le vote majoritaire ou la moyenne des K voisins. Cette approche permet à l'algorithme de s'adapter à différents modèles et de faire des prédictions basées sur la structure locale des données.

1. **Distance euclidienne**

Ce n'est rien d'autre que la distance cartésienne entre les deux points. La distance euclidienne peut également être visualisée comme la longueur de la ligne droite qui relie les deux points considérés. Cette métrique nous aide à calculer le déplacement net effectué entre les deux états d'un objet.

1. **Distance de Manhattan**

La métrique Manhattan Distance est généralement utilisée lorsque nous nous intéressons à la distance totale parcourue par l'objet au lieu du déplacement. Cette métrique est calculée en additionnant la différence absolue entre les coordonnées des points en n dimensions.

1. **Distance de Minkowski**

On peut dire que la distance euclidienne, ainsi que la distance de Manhattan, sont des cas particuliers de la distance de Minkowski.

À partir de la formule sous-dessus, nous pouvons dire que lorsque p = 2 alors c'est la même chose que la formule de la distance euclidienne et lorsque p = 1 alors nous obtenons la formule de la distance de Manhattan.

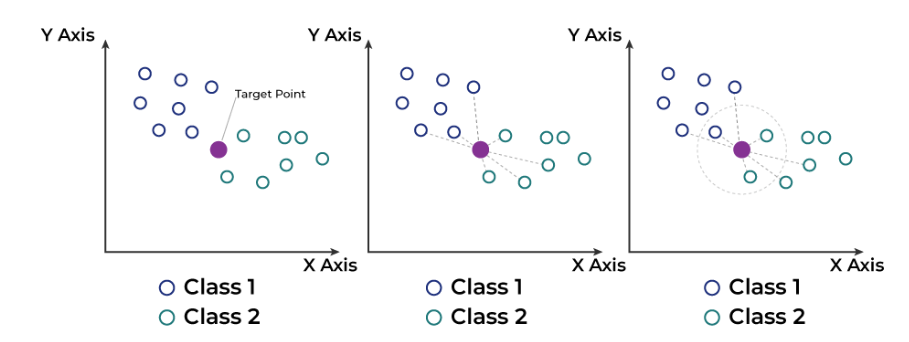
L'algorithme K-Nearest Neighbours (KNN) fonctionne sur le principe de similarité, dans lequel il prédit l'étiquette ou la valeur d'un nouveau point de données en considérant les étiquettes ou les valeurs de ses K voisins les plus proches dans l'ensemble de données d'entraînement.

Figure 17 : fonctionnement de KNN

Le fonctionnement de l'algorithme KNN peut être expliqué en plusieurs étapes :

1. Sélection de la Valeur de K : K représente le nombre de voisins les plus proches à considérer pour la prédiction.
2. Calcul de la Distance : La distance entre chaque point de données de l'ensemble d'entraînement et le point cible est calculée, généralement en utilisant la distance euclidienne.
3. Identification des Voisins les Plus Proches : Les k points de données ayant les plus petites distances au point cible sont identifiés comme les voisins les plus proches.
4. Classification ou Régression :

* Classification : Un vote majoritaire parmi les étiquettes de classe des k voisins les plus proches est effectué. La classe la plus fréquente est attribuée au point cible.
* Régression : La moyenne des valeurs cibles des k voisins les plus proches est calculée et attribuée comme prédiction pour le point cible.

En résumé, étant donné un nouvel échantillon x, KNN calcule la distance entre x et chaque point de l'ensemble d'entraînement X. Les K points ayant les distances les plus courtes sont sélectionnés. Pour la classification, l'étiquette la plus fréquente parmi ces K voisins est choisie comme prédiction. Pour la régression, la moyenne des valeurs des K voisins est utilisée comme prédiction.

##### Conclusion

Ce chapitre a fourni une vue d'ensemble des outils et des concepts clés qui sous-tendent notre application, offrant ainsi un cadre théorique solide pour les développements et analyses ultérieurs.

# Chapitre 3 : Etude analytique et conceptuelle

La conception est une étape très importante dans le développement d'un logiciel. Elle consiste principalement à comprendre les besoins et à analyser le problème pour trouver une solution adaptée. Nous sommes actuellement en train de concevoir la solution qui doit être conforme à l'architecture choisie pour chaque partie du système. Cette étape nous permettra de définir les différentes parties du système de manière globale.

1. Diagramme de cas d’utilisation

Le diagramme de cas d'utilisation est le premier diagramme du modèle UML. II permet de représenter les interactions fonctionnelles entre les acteurs et le système étudié. Il est particulièrement important pour l'organisation et l'identification des grandes fonctionnalités du système.

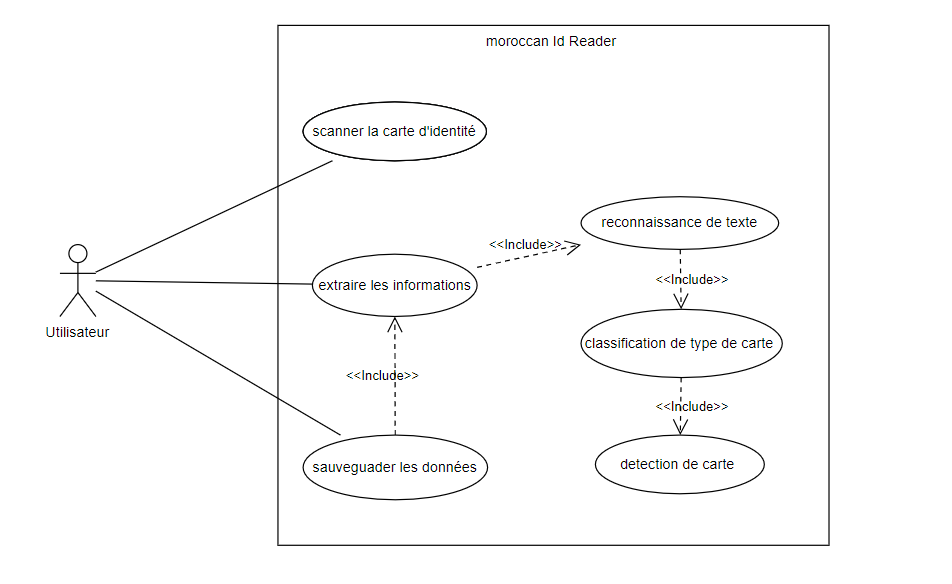
 Ce diagramme de cas d'utilisation (figure 16) décrit les différentes fonctionnalités disponibles pour l'utilisateur du lecteur de la carte nationale.

Figure 18 : Diagramme de cas d'utilisation

* Scanner la carte nationale : cette tâche permet à l’utilisateur de prendre des photos de la carte (recto et verso).
* Extraire les informations : cette tâche permet à l’utilisateur de lancer le traitement pour la lecture des informations à partir de la carte photographiée.
* Détection de carte : cette tâche lance le processus pour détecter la carte dans l’image.
* Classification de type de carte : cette tâche sert à classifier le type de carte entre {nouveau / ancienne ; recto / verso}.
* Reconnaissance de texte : cette tâche permet de lire des informations à partir d’une image.
* Sauvegarder les données : cette tâche permet l’utilisateur de sauvegarder les données après leur extraction.

1. Diagramme de class

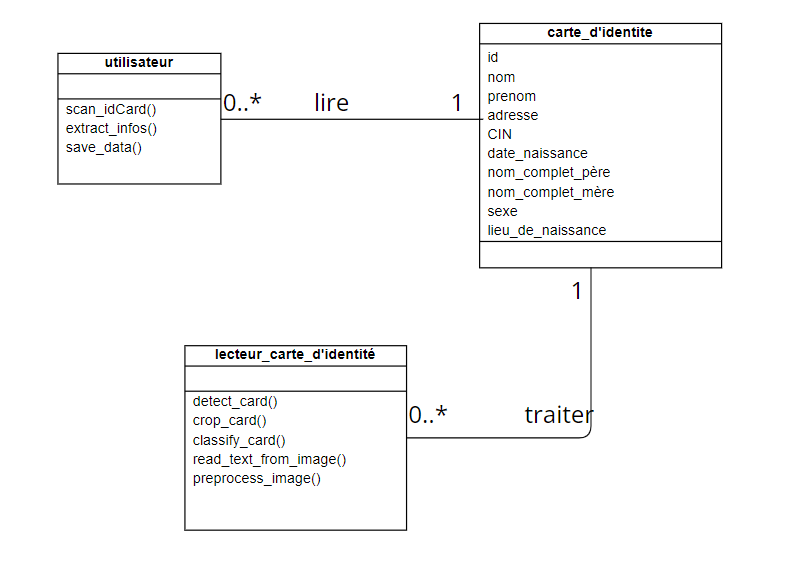
Un diagramme de classe a pour but de présenter les classes et les interfaces des systèmes ainsi que leurs relations. Il représente la vue statique du système en modélisant les concepts du domaine, les entités abstraites nécessaires à l'implémentation de l'application et les relations que ces concepts ou entités entretiennent entre eux.

Figure 19 : diagramme de classe

* La classe utilisateur représente l’utilisateur de l’application elle possède les méthodes pour scanner la carte d’identité, extraction ses informations et les sauvegardes.
* La classe carte\_d’identité représente les informations que la carte nationale contient
* La classe lecteur\_carte\_d’identité représente le system qui automatise l’extraction les informations à partir des images de carte d’identité elle possède les méthodes qui permet la detection de la carte dans l’image, coupage de sa position, sa classification, lecture du texte dans l’image et le prétraitement de l’image.

1. Diagramme de séquence

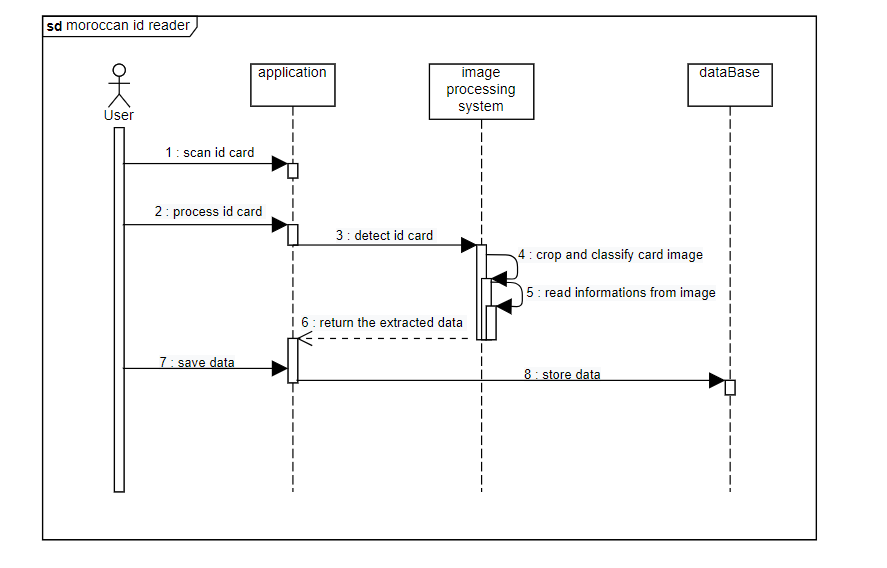
Les diagrammes de séquence sont des outils puissants utilisés dans la modélisation de systèmes pour représenter les interactions entre différents objets ou composants au sein d'un système. Ils permettent de visualiser la chronologie des messages échangés dans un scénario spécifique, facilitant ainsi la compréhension et la conception des fonctionnalités du système.

Figure 20 : diagramme de séquence

Ce diagramme de séquence décrit le processus de l’extraction des données à partir d’image de la carte d’identité, l’utilisateur commence par prendre les photos de la carte puis il lance le processus de l’extraction de données.

Après le système reçoit l’image premièrement il détecte l’emplacement de la carte dans l’image et la couper, après le coupage il lire le texte dans l’image et ensuite il le renvoie à l’application pour l’affichage.

Donc après l’extraction des données l’utilisateur pourrai sauvegarder ses données à la base de données.

# Chapitre 4 : Méthodologie

1. Introduction

Dans ce chapitre, nous allons détailler la mise en œuvre pratique de notre projet ainsi que l'environnement de travail dans lequel il a été développé. Après avoir abordé les concepts théoriques et les bases technologiques dans les chapitres précédents, nous nous concentrerons ici sur l'application concrète de ces connaissances.

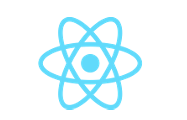
1. L’environnement de développement

Figure 21 : react native logo

**React Native** est un framework open-source développé par Facebook qui permet de créer des applications mobiles pour les plateformes iOS et Android en utilisant le langage de programmation JavaScript. Il a été créé pour permettre aux développeurs de créer des applications mobiles performantes et réactives avec une expérience utilisateur de qualité tout en utilisant un code JavaScript partagé entre les deux plateformes. Ce langage offre plusieurs avantages :

* Développement rapide et efficace : React Native permet de développer des applications mobiles pour les deux principales plateformes (iOS et Android) en utilisant un seul code base, ce qui permet de réduire considérablement le temps et les coûts de développement.
* Performances élevées : Les applications mobiles créées avec React Native ont des performances élevées et une grande réactivité grâce à l'utilisation de la technologie de rendu en temps réel.
* Expérience utilisateur fluide : React Native permet de créer des interfaces utilisateur fluides et réactives qui ressemblent à des applications natives, ce qui améliore l'expérience utilisateur.



Figure 22 : npm logo

**NPM** (Node Package Manager) est un gestionnaire de packages pour JavaScript. Il est utilisé principalement avec Node.js pour installer, gérer et partager des bibliothèques et des modules JavaScript. NPM facilite la gestion des dépendances de projet, permettant aux développeurs d'ajouter et de mettre à jour des packages externes nécessaires à leurs applications.

Figure 23: expo logo

**Expo** est une plateforme de développement d'applications mobiles qui facilite la création d'applications natives pour iOS, Android et le web en utilisant JavaScript et React Native. Il fournit un ensemble d'outils, de bibliothèques et de services qui permettent aux développeurs de créer, tester et déployer des applications rapidement et efficacement. Expo simplifie le processus de développement en gérant de nombreuses tâches techniques, telles que la configuration de l'environnement de développement, la compilation de l'application et le déploiement sur les app stores. Il offre également un large éventail de fonctionnalités intégrées, comme l'accès aux capteurs du téléphone (appareil photo, gyroscope, etc).



Figure 24 : vscode logo

**Visual Studio Code** est un éditeur de code source développé par Microsoft. Il est disponible sur plusieurs plateformes, notamment Windows, macOS et Linux. Il offre une gamme de fonctionnalités pour faciliter le développement de logiciels, telles que la coloration syntaxique, l'autocomplétions, la détection d'erreurs, le débogage, l'intégration avec des systèmes de contrôle de version tels que Git, et bien plus encore.

**Firebase** est une plateforme de développement d'applications mobiles et web développée par Google. Elle offre un large éventail de services et d'outils pour aider les développeurs à créer des applications rapidement et facilement.

Figure 25 : python logo

Figure 26 : firebase logo

**Python** est le langage de programmation open source le plus employé par les informaticiens. Ce langage s’est propulsé en tête de la gestion d’infrastructure, d’analyse de données ou dans le domaine du développement de logiciels.

Figure 27 : flask logo

Flask est un micro-framework web écrit en Python qui permet de développer des applications web rapidement et facilement en utilisant des outils simples et concis. Il est connu pour sa simplicité et sa flexibilité, ce qui en fait un choix populaire pour les petites applications web, les prototypes et les projets expérimentaux. Flask est également extensible, ce qui signifie que vous pouvez ajouter des fonctionnalités supplémentaires grâce à des extensions tierces pour répondre à vos besoins spécifiques.

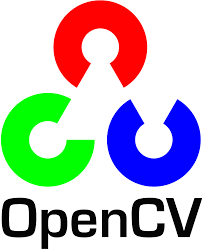


Figure 28 : openCV logo

OpenCV (Open Source Computer Vision Library) est une bibliothèque open-source spécialisée dans le traitement et l'analyse d'images et de vidéos. Elle fournit une large gamme d'algorithmes et de fonctions pour les tâches courantes en vision par ordinateur, telles que la reconnaissance d'objets, la détection de visages, le suivi de mouvements, et la reconstruction 3D. Créée par Intel et maintenant maintenue par une communauté active de développeurs, OpenCV est largement utilisée dans les applications de vision artificielle,



Figure 29 : numPy logo

NumPy est une bibliothèque Python qui est principalement utilisée pour le calcul scientifique et le traitement de tableaux multidimensionnels. NumPy fournit des fonctions pour effectuer des opérations mathématiques telles que l'addition, la soustraction, la multiplication, la division, etc. sur des tableaux. Il est largement utilisé pour la manipulation des données dans les domaines tels que la science des données, l'apprentissage automatique, l'analyse numérique, etc.

1. Processus de system

Au manque des données des cartes nationales marocaines pour réaliser un modèle de ML qui automatise l’extraction les informations j’ai travaillé sur une chaîne de traitement illustrée dans la figure 5. La chaîne a pour entrer deux images et comme sortie un document JSON contenant les informations extraites à partir de ces deux photos de la carte nationale :

* Détection en utilisant le model YOLO et Coupage la position la carte dans l’image.
* Classifier les types des cartes nationales (nouveau-recto / nouveau-verso ; ancienne-recto / ancienne-verso) avec l’algorithm SIFT et KNN.
* L’extraction des informations à partir des images avec EasyOCR.

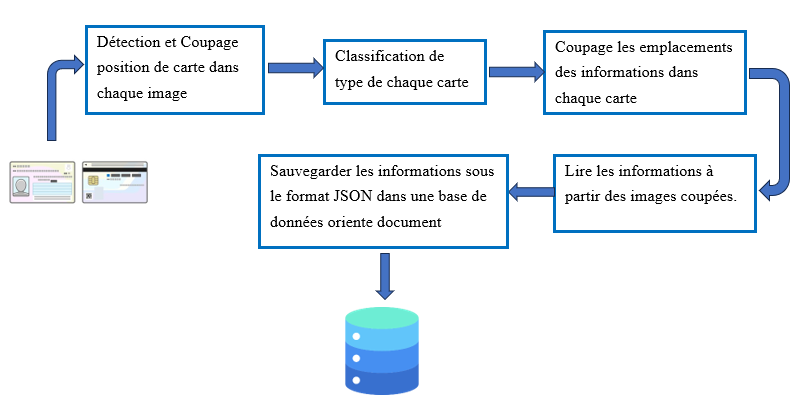
1. Le fine tuning du modèle YOLO

Figure 30: chaine de traitement

Pour le fine-tuning du modèle YOLO destiné à la détection des bords de carte et En raison de l'absence de cartes marocaines annotées, j'ai utilisé un jeu de données de cartes d’identité disponibles sur le site Roboflow, les cartes internationales ont été choisies en raison de leurs caractéristiques visuelles similaires, notamment les bords et les contours, qui sont pertinents pour la tâche de détection. Cette approche pragmatique permet de tirer parti de la disponibilité immédiate des données annotées, tout en offrant la possibilité de former un modèle efficace pour la tâche ciblée.

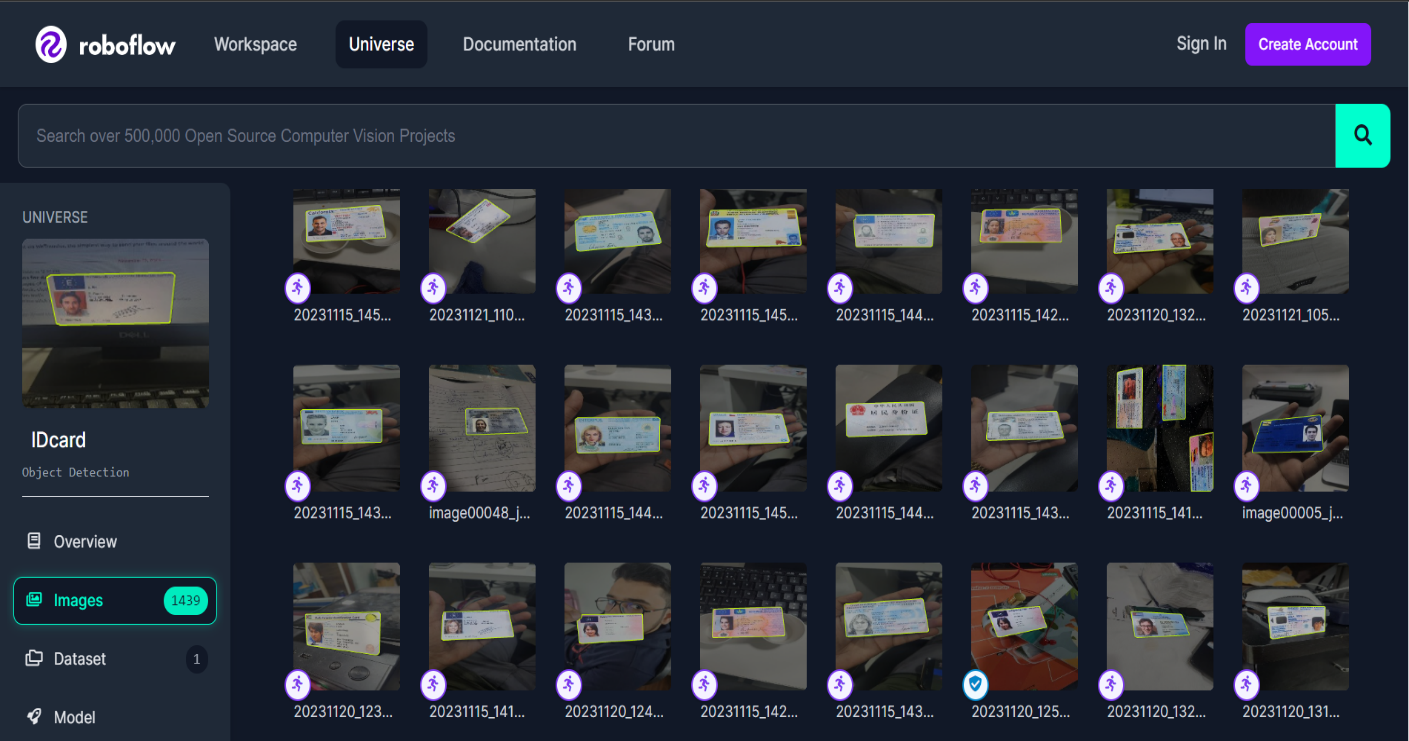


Figure 31 : exemples des images du dataset

1. **dataSet**

Le jeu de données comprend :

**Images** : Des images de cartes, téléchargées depuis le site Roboflow.

**Annotations** : Chaque image est accompagnée d'un fichier d'annotation au format YOLO.

Les annotations YOLO sont stockées dans des fichiers texte, où chaque ligne correspond à un objet détecté dans l'image. Voici comment ces annotations sont structurées :

* **Classe** : Un entier représentant la classe de l'objet. Dans mon cas il y’a deux classes {ID-Card ; background(pas de id-card)}.
* **Coordonnées Normalisées** :
* **Centre de la boîte englobante (bounding box)** : Les coordonnées x et y du centre de la boîte, normalisées par la largeur et la hauteur de l'image (valeurs comprises entre 0 et 1).
* **Dimensions de la boîte englobante** : La largeur et la hauteur de la boîte, également normalisées par les dimensions de l'image.

1. **Prétraitement des données**

Ce prétraitement a été réalisé par l'auteur afin d'assurer la qualité et la cohérence des données utilisées pour le fine-tuning du modèle YOLO. Ces étapes de prétraitement et d'augmentation des données sont essentielles pour améliorer les performances du modèle et garantir sa capacité à généraliser sur des données non vues.

**Auto-Orientation** : Les images sont automatiquement orientées pour s'assurer qu'elles sont dans la position correcte, indépendamment de la manière dont elles ont été capturées.

**Redimensionnement** : Les images sont redimensionnées pour s'adapter à une résolution standard de 640x640 pixels.

**Saturation** : La saturation des images est ajustée aléatoirement entre -65% et +65%.

**Bruit** : Ajout de bruit aléatoire jusqu'à 1.09% des pixels.

1. **L’entrainement de modèle YOLO**

Le processus de fine-tuning permet d'adapter le modèle YOLO pré-entraîné pour répondre spécifiquement à la tâche de détection des bords de cartes. En ajustant les paramètres du modèle à partir des données annotées, il devient capable de repérer les bords et les absences de cartes avec une grande précision. Le modèle est entraîné pendant 60 époques avec des images redimensionnées à 640x640 pixels et une taille de lot de 64, en utilisant la mise en cache pour accélérer le processus. Une graine fixe (123) assure la reproductibilité des résultats, et une validation continue est effectuée pour évaluer les performances du modèle et éviter le surapprentissage. Ces hyperparamètres permettent d'optimiser le modèle YOLO pour la tâche spécifique de détection des bords de cartes, assurant ainsi une performance robuste et fiable.

1. **Validation du modèle**

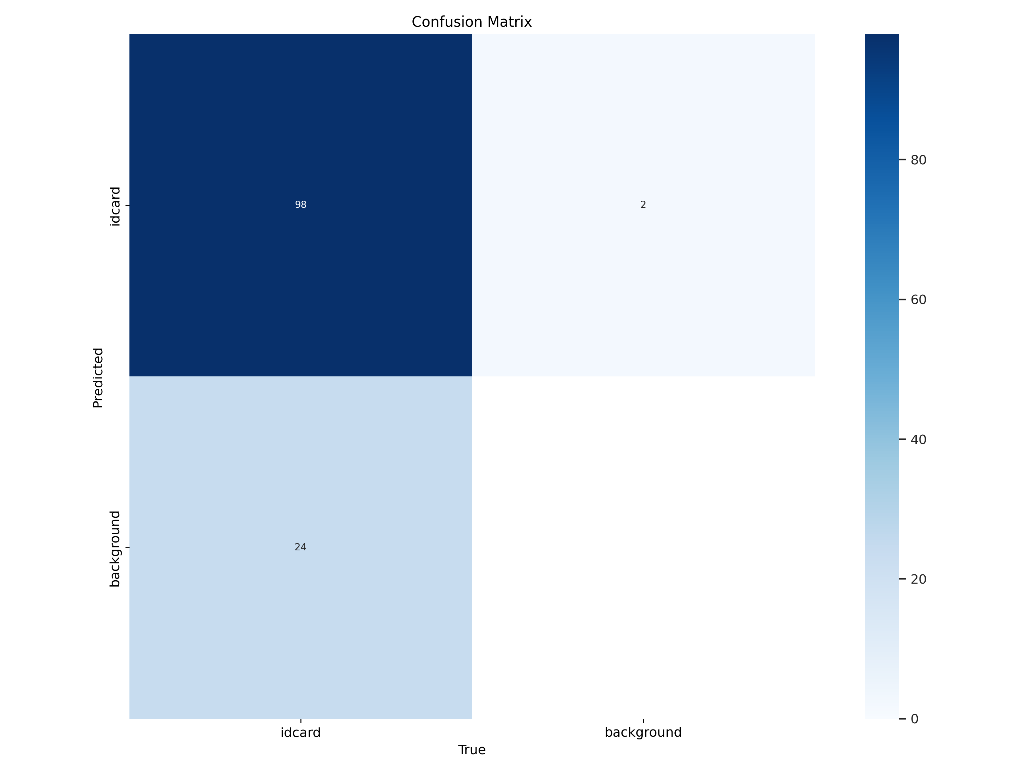
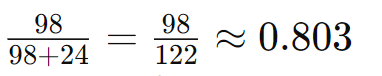
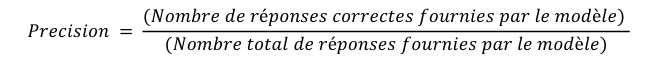
* **Matrice de confusion :**

Figure 32: matrice de confusion

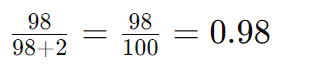
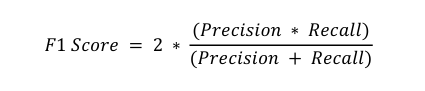
Une matrice de confusion est un outil utilisé pour évaluer les performances d'un modèle de classification. Elle permet de visualiser les prédictions correctes et incorrectes du modèle et de calculer diverses métriques de performance.

**N.B : l’évaluation est effectuée sur des images de validation (124 images)**

**La précision** mesure la proportion de réponses correctes fournies par le modèle parmi toutes les réponses qu'il a générées.

Et à partir de la matrice de confusion on peut calculer la précision :

**Rappel** (Recall) : Le rappel mesure la proportion de réponses correctes fournies par le modèle parmi toutes les réponses correctes réelles dans les données de référence. Il se calcule de la manière suivante :

**F1 Score** est une mesure de performance qui évalue à la fois la précision et le rappel d'un modèle de question-réponse. Il est utilisé pour obtenir une mesure plus globale de la qualité des réponses fournies par le modèle.

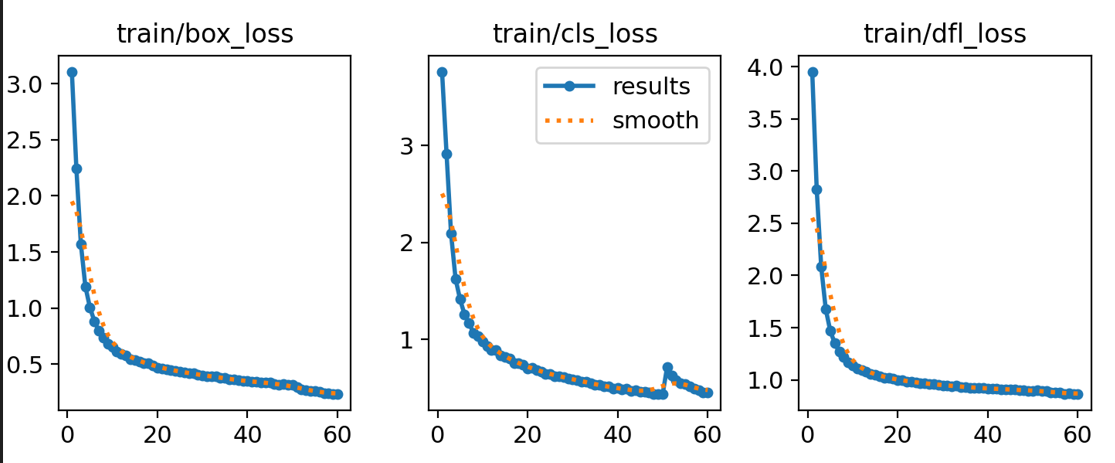


Figure 33 : l'évaluation avec les graphes de pertes

 **train/box\_loss** : Cette courbe montre une tendance nettement à la baisse au fil des époques, indiquant que le modèle apprend efficacement à prédire des boîtes englobantes précises pour les objets.

 **train/cls\_loss** : la courbe de perte de classification présente également une diminution significative au fil des époques. Cela suggère que le modèle améliore sa capacité à classer correctement les objets détectés.

 **train/dfl\_loss** : La courbe de perte focale semble fluctuer autour d'une valeur constante après une diminution initiale. Cela implique que le modèle pourrait avoir atteint un plateau en termes d'apprentissage de l'estimation de la longueur focale.

 Exemple de prédiction avec les images de validation :

Figure 34 : exemples de prédiction

1. Classification de type de ID-carte avec SIFT et KNN
2. Détection et Extraction des Caractéristiques avec SIFT

Le premier algorithme, SIFT, est utilisé pour analyser et extraire des caractéristiques uniques des images. Voici comment cela fonctionne :

* **Détection des Points Clés** : SIFT identifie des points d'intérêt dans une image, appelés points clés, qui sont distincts et invariants par rapport aux changements d'échelle et de rotation.
* **Extraction des Descripteurs** : Pour chaque point clé, SIFT calcule un descripteur, qui est un vecteur de caractéristiques représentant les informations locales autour du point clé. Ces descripteurs capturent l'essence visuelle de l'image de manière robuste.

En appliquant SIFT à la fois sur l'image d'entrée et sur les images de référence (anciennes et nouvelles cartes d'identité), le code obtient des ensembles de descripteurs pour chacune d'elles.

1. Correspondance des Caractéristiques avec KNN

Le second algorithme, KNN, est utilisé pour comparer les descripteurs extraits et trouver les correspondances les plus proches entre eux. Voici les étapes impliquées :

* **Correspondance des Descripteurs** (features matching): le matcher KNN trouve les k (ici k=2 signifie que pour chaque descripteur de l'image d'entrée, l'algorithme KNN cherche les deux descripteurs les plus similaires dans les images de référence) descripteurs les plus proches dans les images de référence pour chaque descripteur de l'image d'entrée.
* **Filtrage des Correspondances** : Les correspondances sont filtrées pour ne garder que les "bonnes" correspondances, en vérifiant que la distance de la meilleure correspondance est significativement plus petite que celle de la seconde meilleure correspondance.

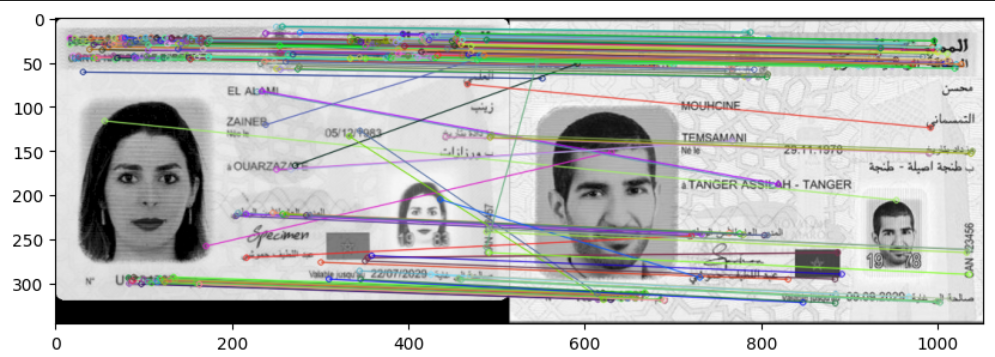


Figure 35 : exemple de Featers matching avec SIFT

1. Identification de la Carte la Plus Similaire

En combinant les résultats de SIFT et KNN :

* **Extraction des Caractéristiques** : SIFT extrait des descripteurs des images d'entrée et de référence.
* **Correspondance des Caractéristiques** : KNN trouve et filtre les meilleures correspondances de descripteurs entre l'image d'entrée et chaque image de référence.
* **Calcul des Scores de Similarité** : Le nombre de bonnes correspondances est utilisé pour évaluer la similarité entre l'image d'entrée et chaque image de référence.
* **Détermination du Type de Carte** : Le type de carte d'identité le plus similaire est déterminé par le score de correspondance le plus élevé.

1. Extraction du texte à partir d’image avec EasyOCR

Le prétraitement

Avant d'utiliser EasyOCR pour la reconnaissance de texte sur les cartes d'identité, un prétraitement de l'image est effectué pour améliorer la qualité et la lisibilité des informations textuelles. Voici une explication détaillée de ce processus de prétraitement :

 **Amélioration du Contraste** : L'image est passée par un rehausseur de contraste pour accentuer les différences entre les régions claires et sombres, ce qui rend le texte plus distinct.

**Augmentation de la Luminosité** : La luminosité de l'image est augmentée pour rendre le texte plus visible, en particulier dans les zones sombres.

** Affinement de la Netteté** : La netteté de l'image est améliorée pour rendre les bords du texte plus clairs et définis.

 **Conversion en Niveaux de Gris** : L'image améliorée est convertie en niveaux de gris, c’est une étape courante dans le prétraitement des images pour la reconnaissance de texte.

 **Binarisation Adaptative** : Une binarisation adaptative est appliquée pour convertir l'image en une image binaire (noir et blanc). Cela facilite la distinction du texte du fond.

**Flou Gaussien** : Un flou gaussien est appliqué à l'image binaire pour réduire le bruit et les petites imperfections, améliorant ainsi la qualité globale de l'image pour la reconnaissance de texte.

**Masque de Netteté Non Accentué** : Enfin, un masque de netteté non accentué est appliqué pour encore améliorer la clarté du texte en combinant les versions floue et nette de l'image.

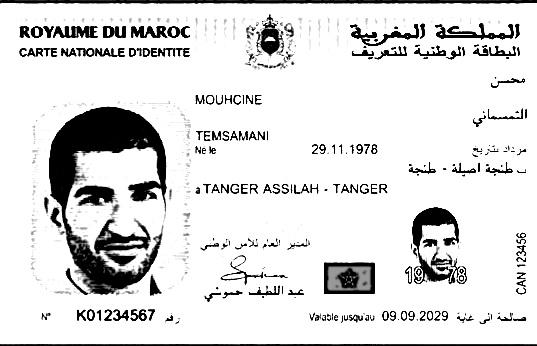
Exemple de prétraitement d’image :

Figure 36 : image après le traitement

Figure 37 : image avant le traitement

L’extraction du texte

Après la classification du type de carte d'identité, le processus de traitement se poursuit par l'extraction et la lecture des informations spécifiques sur la carte à l'aide de EasyOCR. Voici les étapes détaillées de ce processus :

**Conversion des Coordonnées YOLO en Pixels :** Les coordonnées des boîtes englobantes prédéfinies au format YOLO sont converties en coordonnées de pixels. Cette conversion permet de localiser précisément les zones d'intérêt qui contient les informations sur la carte.

**Découpage des Zones Spécifiques :** À l'aide des coordonnées converties, les zones spécifiques de l'image de la carte d'identité sont découpées. Chaque zone découpée correspond à une région prédéfinie où des informations importantes sont situées, comme le numéro de la carte, le nom, la date de naissance, etc.

**Lecture du Texte avec EasyOCR :** EasyOCR est utilisé pour lire et extraire le texte des images découpées. EasyOCR est une bibliothèque de reconnaissance de texte qui utilise des modèles d'apprentissage profond pour lire le texte dans plusieurs langues.

**Extraction et Retour des Résultats :** Les résultats de la reconnaissance de texte sont collectés et retournés. Ces résultats incluent le texte lu dans chaque zone spécifique de la carte d'identité, et puis ils sont stockés dans un dictionnaire de données pour un prochain traitement.

1. Mise en œuvre de l’application mobile

Après avoir conçu et analysé le projet, et présenté les outils de travail nécessaires, nous arrivons maintenant à la phase de concrétisation et d'implémentation de tout ce qui a été précédemment élaboré. Dans ce qui suit, je vais présenter les interfaces du projet.

1. splash screen

La première interface qui s’affiche après l'ouverture de l’application est le splash screen. Cette page d'accueil, visible pendant 5 secondes, sert plusieurs objectifs importants. Elle améliore l'expérience utilisateur en offrant une introduction visuellement agréable à l'application, crée une première impression positive, et donne un temps suffisant pour charger les ressources nécessaires en arrière-plan. De plus, elle renforce l'identité de marque en affichant le logo et le nom de l'application, tout en signalant à l'utilisateur que le lancement est en cours. Le splash screen agit donc comme une transition fluide entre le lancement de l'application et l'accès aux fonctionnalités principales.

Figure 38 : Splash Screen

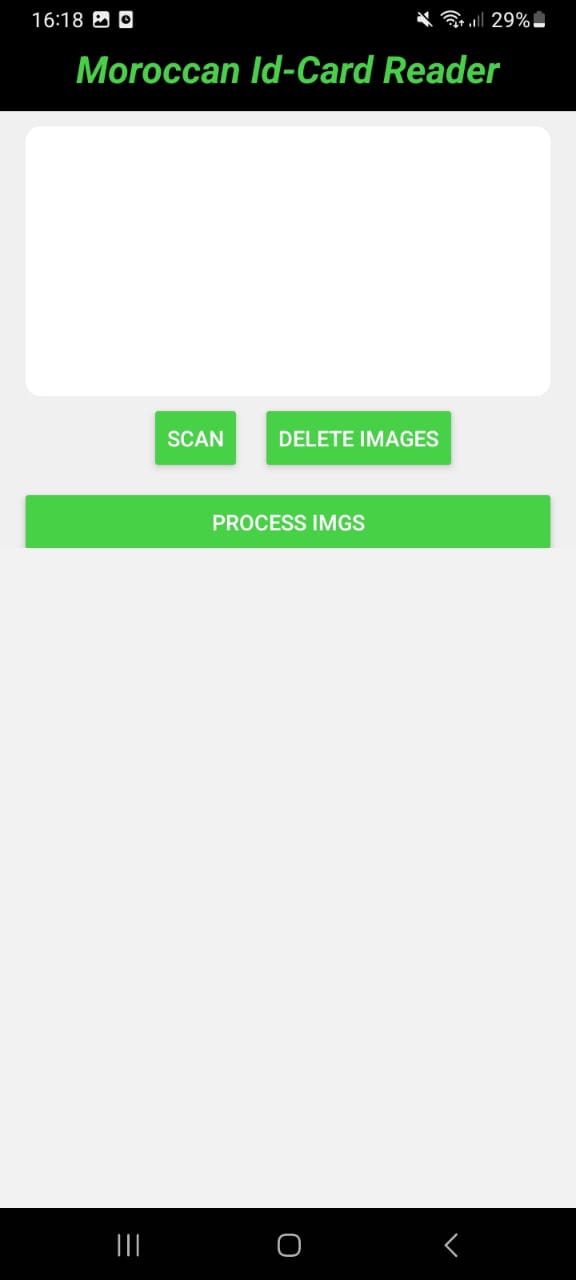
1. Page principale

Figure 39 : Interface principale

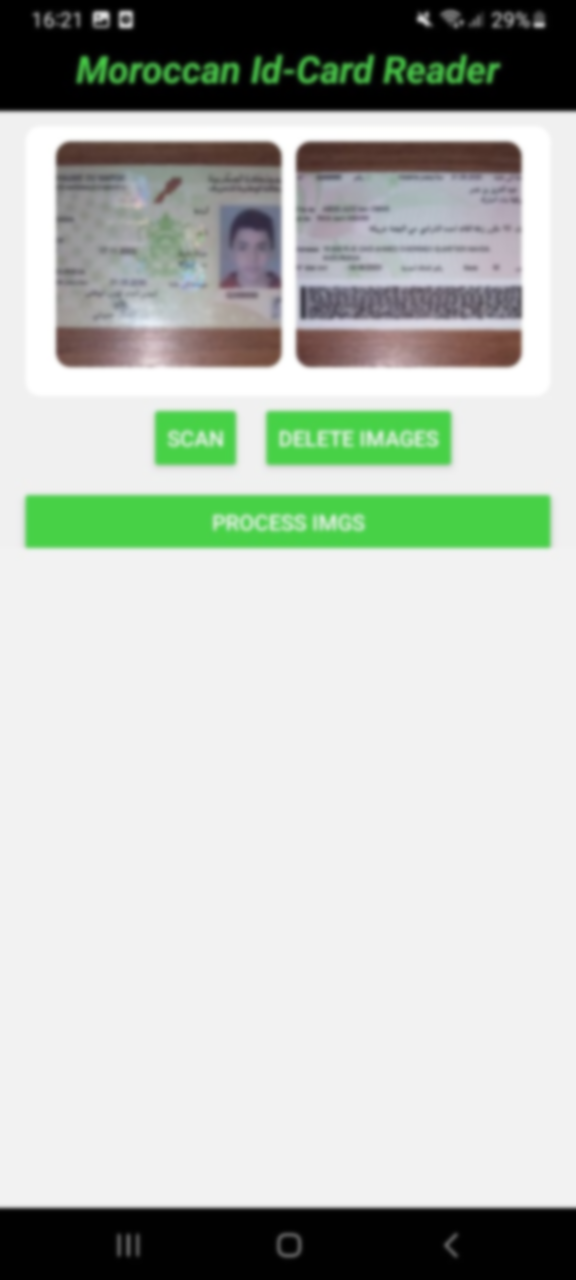
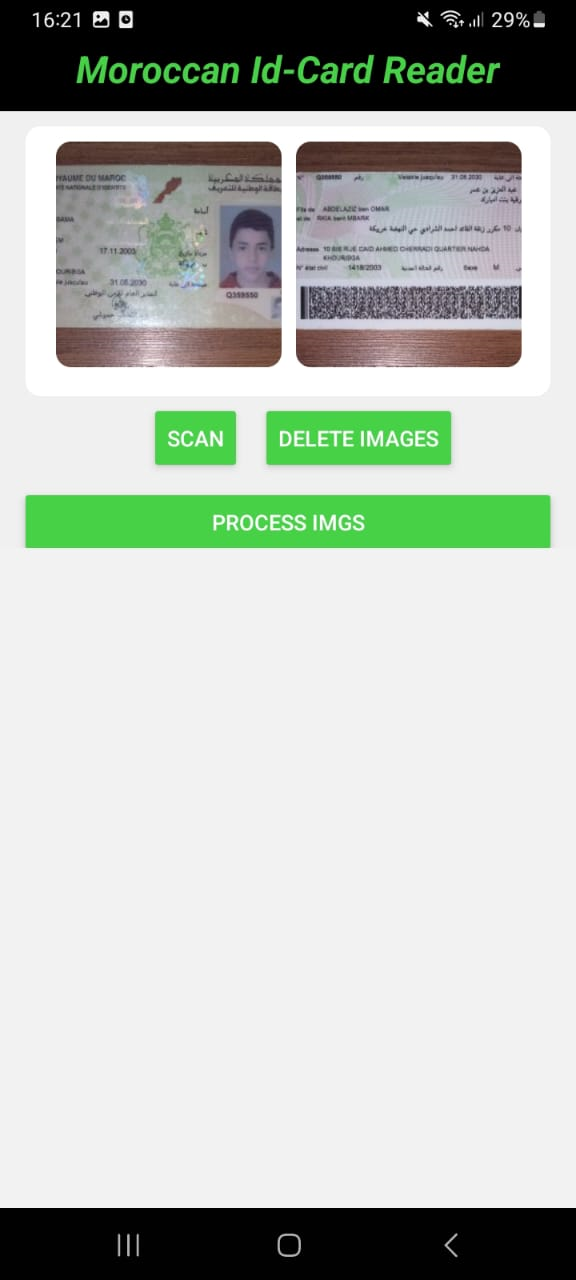
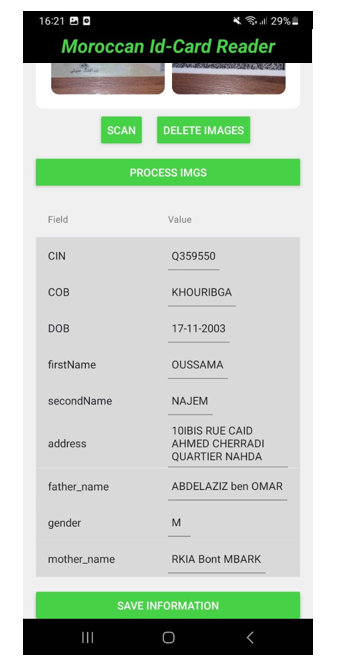
Après l’expiration de Splash Screen cette interface s’affiche elle joue un rôle crucial dans le processus de capture et de traitement des images de cartes d'identité. Elle permet aux utilisateurs de capturer jusqu'à deux photos de la carte d'identité (recto et vesro) à l'aide de la caméra de leur appareil, via le bouton "Scan". Les images capturées sont affichées sur l'interface utilisateur dans des cadres dédiés (figure 30), et en l'absence d'images, un cadre de couleur blanc est affiché en guise d’espace réservé. Les utilisateurs peuvent supprimer les images capturées en appuyant sur le bouton "Delete Images", ce qui réinitialise les cadres des images pour une nouvelle capture. Une fois les images capturées, les utilisateurs peuvent les envoyer au serveur pour traitement en appuyant sur le bouton "Process Images". Les images sont alors converties en base64 et envoyées via une requête HTTP POST au serveur spécifié. Après le traitement des images par le serveur et l’extractions des informations, les résultats sont affichés dans un tableau (figure 31). Chaque ligne du tableau représente un champ de données extrait de la carte d'identité, avec la possibilité d'éditer les valeurs extraites directement dans l'application. Et finalement l’utilisateurs peut sauvegarder les informations extraites dans une base de données Firestore en appuyant sur le bouton "Save Information".

Figure 31 : l’affichages des résultats extraits

Figure 30: l’affichages des images

# Conclusion

Dans ce projet, nous avons réussi à atteindre pleinement nos objectifs initiaux, consistant à développer une application mobile automatisant le processus d'inscription des étudiants à la piscine de l'école 1337. Notre application offre des fonctionnalités avancées, notamment la capture et la reconnaissance automatique des informations clés sur les cartes d'identité, simplifiant ainsi considérablement le processus d'inscription et améliorant l'efficacité administrative de l'établissement.

L'impact potentiel de notre solution dans le domaine de l'éducation est significatif. En automatisant les tâches administratives, notre application libère du temps pour les équipes administratives et garantit une gestion plus précise des données des étudiants, tout en réduisant les erreurs humaines.

Malgré le succès de notre solution, nous reconnaissons également ses limitations. Par exemple, l'application peut rencontrer des difficultés avec les cartes d'identité mal imprimées ou endommagées, ce qui peut affecter la précision de la reconnaissance automatique. De plus, l'application dépend de la qualité des images capturées par les utilisateurs, ce qui peut varier en fonction de divers facteurs tels que l'éclairage et la résolution de la caméra.

Pour améliorer notre solution, nous envisageons de collecter un ensemble de cartes nationales marocaines pour entraîner notre modèle de reconnaissance automatique. Cette approche pourrait améliorer la précision et la fiabilité de notre application, en réduisant sa sensibilité aux variations dans les images des cartes d'identité.

Bibiliography

https://www.coursera.org/articles/ai-vs-deep-learning-vs-machine-learning-beginners-guide

<https://blent.ai/blog/a/detection-images-yolo-tensorflow>

<https://www.analyticsvidhya.com/blog/2019/10/detailed-guide-powerful-sift-technique-image-matching-python/>

<https://www.jaided.ai/easyocr/documentation/>

https://www.geeksforgeeks.org/k-nearest-neighbours/