

Classes Préparatoires Intégrées (CPI)
2^{ème} année

PROJET

Reconnaissance des chiffres imprimés par la méthode de corrélation

Sujet N° : **03**

Equipe N° : **21**

1. CHALAL Lyes
2. BENCHELLI Melissa
3. BRAHITI Sarah
4. GUERROUF Amar
5. NASRI Housseem
6. OUBAHI Abderrahmane

Encadrée par :

- Mme. AIT DAOUD Siham
- Mr. DELLYS Hachemi

ANNÉE : 2021 / 2022

Table des matières :

<i>I. Remerciements.....</i>	<i>3</i>
<i>II. Introduction.....</i>	<i>4</i>
<i>III. Analyse du cahier de charge :.....</i>	<i>5</i>
1. Présentation de la problématique et des objectifs :.....	5
2. Environnement matériel et logiciel :.....	5
3. L’interface utilisateur :.....	5
<i>IV. Organisation du projet</i>	<i>6</i>
1. Diagramme de Gantt :	6
2. Communication :	6
a. Entre les membres :	6
b. Avec les encadreurs :	7
3. Déroulement du projet :.....	8
a. Première phase : Recherche et documentation	8
b. Deuxième phase : Conception.....	8
c. Troisième phase : Réalisation.....	8
<i>V. Présentation du Sujet</i>	<i>9</i>
1. Introduction	9
2. Qu’est-ce qu’une image numérique ?	9
3. Qu’est-ce que la corrélation d’images numériques ?	9
4. Cas d’études :	9
<i>VI. Conception</i>	<i>10</i>
1. Introduction :	10
2. Développement :	10
3. Conclusion :	15
<i>VII. Réalisation</i>	<i>16</i>
1. Front-End :	16
2. Back-End :	17
3. Améliorations :	21
4. Présentation de l’application :	22
<i>VIII. Résumé :</i>	<i>24</i>
<i>IX. Conclusion Générale :.....</i>	<i>25</i>
<i>Perspectives :</i>	<i>25</i>
<i>Bibliographie :.....</i>	<i>26</i>

I. Remerciements

Tout d’abord, nous tenons à remercier nos encadrants, Mme Ait Daoud Siham et Mr Dellys Hachemi pour leurs patiences, leur disponibilité, les efforts qu’ils ont fournis pour nous guider à réaliser un travail affiné, et l’aide qu’ils nous ont apporté et qui nous a permis de toujours pousser plus loin notre réflexion.

On aimerait aussi remercier notre ami, Nazim Bendib, étudiant en 1CS qui a été d’un grand soutien intellectuel tout au long de notre projet.

Enfin, on aimerait se remercier les uns les autres, chacun pour son engagement et son sérieux. Merci à toute l’équipe pour les efforts fournis et merci à chacun d’avoir donné le meilleur de lui-même.

II. Introduction

Le monde d'aujourd'hui est spectateur de l'évolution informatique dans tous les domaines de la vie. L'informatisation ou plutôt l'automatisation des fonctionnalités utiles à la productivité et la rentabilité de la performance des entreprises de tout genre devient le quotidien de l'être humain ces dernières années et l'idée de ne pas connaître cette évolution devient strictement impossible à imaginer.

La reconnaissance de caractères est très utilisée dans la pratique notamment en OCR (Optical character recognition), correction automatique d'orthographe, l'archivage des documents etc. La reconnaissance des chiffres imprimés est nécessaire pour l'identification des plaques d'immatriculation des voitures et leur poursuite ou encore le code bar des produits dans les grandes surfaces afin de ne plus avoir recours à faire une saisie manuelle.

Pour ce projet, nous nous intéressons à la reconnaissance des chiffres imprimés par la méthode de corrélation.

Cette dernière est très utilisée en OCR qui désigne les procédés informatiques pour la traduction d'images de textes imprimés ou dactylographiés en fichiers de texte.

C'est donc selon cet objectif, et dans ce contexte que notre équipe a réalisé, dans le cadre de ce projet qui a lieu durant le second semestre de la deuxième année, un outil de reconnaissance de chiffres imprimés par la méthode de corrélation.

Dans ce rapport, nous allons donc commencer par décortiquer le cahier des charges qui nous a été transmis au début du projet, afin de déceler les objectifs de notre travail et ce qui est demandé.

Nous dresserons le déroulement du travail pour arriver aux objectifs fixés étape par étape. Nous commencerons par présenter le thème qui a fait objet de ce projet, à savoir, « la corrélation d'images numériques ». Nous enchaînerons ensuite avec une étude conceptuelle dans le chapitre dénommé « Conception », et pour mettre en pratique l'étude susmentionnée, nous aborderons dans le chapitre « Réalisation » les différents moyens techniques et atouts mis à notre disposition pour réaliser notre application.

Le rapport sera clôturé par une conclusion générale comportant les différents aspects que nous avons touchés tout au long de ce projet, les connaissances et les compétences acquises et également, des perspectives vis-à-vis le projet réalisé.

III. Analyse du cahier de charge :

1. Présentation de la problématique et des objectifs :

L'Objectif de notre projet est donc de développer un outil de reconnaissance des chiffres imprimés en implémentant une solution qui devait regrouper différentes fonctionnalités, notamment : la création et l'utilisation d'une base de données avec des images de référence et la comparaison de ces derniers avec des images d'entrée en utilisant la méthode de corrélation.

Notre projet consistera donc à créer un programme qui réalisera toutes les fonctionnalités principales suivantes :

- Téléchargement d'images de chiffres imprimés de différentes fontes et différents styles ;
- Traitement d'images : normalisation, binarisation, découpage ;
- Conception d'une base de données ;
- Implémentation de la méthode de corrélation (matching) ;
- Traitement de résultats et affichage du chiffre reconnu ;
- Calcul des erreurs de similarité ;
- Calculs de performances du système réalisé : taux positifs, taux négatifs.

2. Environnement matériel et logiciel :

Pour la réalisation de cet outil nous avons opté pour une application desktop.

Nous avons également eu la liberté de choisir l'environnement technologique qui nous convenait le mieux, entre Python, Java et Matlab, tout en discutant avec notre encadreur.

3. L'interface utilisateur :

L'interface utilisateur de cet outil devra être intuitive, les touches devront être claires, homogènes et dotées d'indications de leurs fonctionnalités.

IV. Organisation du projet

Étant une équipe de six, et n'ayant pas assez de temps pour nous permettre des erreurs, l'organisation était un trait nécessaire. On a alors consacré notre première réunion au choix des outils qui arrangent tout le monde et qui nous permettraient une meilleure collaboration et une gestion du temps stratégique.

Dans ce chapitre nous allons décrire comment s'est faite cette organisation, allant du diagramme de Gantt, à la communication entre membres et avec l'encadreur.

1. Diagramme de Gantt :

Dans un travail comme le nôtre, il arrive souvent qu'une tâche dépende d'autres pour être complétée, il est aussi souvent nécessaire que deux tâches soient accomplies en parallèle.

Le diagramme de Gantt est un outil de gestion de projet. Il est justement conçu pour permettre au chef d'équipe de gérer et surveiller les tâches individuelles de chaque membre.

Étant donné qu'il est structuré de manière chronologique, Gantt offre aussi la possibilité d'avoir une vue d'ensemble sur le projet, et sa durée, en plus de pouvoir prévoir la durée de chaque tâche individuellement.

Notre chef d'équipe a donc utilisé le logiciel MS Project afin de rédiger et suivre le planning du groupe.

MS Project est un logiciel efficace qui permet la planification d'un projet à travers la réalisation d'un diagramme de Gantt. C'est une décomposition hiérarchique, axée sur les tâches, du travail que l'équipe de projet doit exécuter pour atteindre les objectifs du projet et produire les livrables voulus.



2. Communication :

a. Entre les membres :

Les étudiants de l'équipe ont tous des capacités et des connaissances différentes, et les forces de chaque membre constituent la force de l'équipe. Ainsi, chacun peut compléter les compétences de l'autre et partager les siennes avec lui.

Mais tout cela ne serait pas possible sans une bonne communication et on en avait parfaitement conscience au sein de notre équipe. On a donc choisi les outils les plus adéquats afin de créer une transparence totale et une collaboration effective entre les membres. Mais ce n'est pas tout, les outils qu'on allait choisir devaient aussi assurer un bon team-building.

Les membres de l'équipe effectuaient donc régulièrement des réunions afin de vérifier l'avancement de chacun, de se concerter entre les membres et de s'assurer que le projet avance de la manière la plus sereine possible. Pour le faire, on a décidé de travailler sur *Discord*.

Discord est un logiciel propriétaire gratuit de messagerie instantanée. Il fonctionne sur les systèmes d'exploitation Windows, MacOS, linux, Android, iOS ainsi que sur les navigateurs web. Ce logiciel offre un système de salons qui permet la catégorisation des partages selon leurs contenus.



Nous avons créé un serveur, dans lequel nous avons exploité cette organisation pour organiser nos réunions dans différents salons et les placer ainsi dans leur contexte. Lorsqu'un membre voulait se remémorer les détails d'une réunion écrite, ou voulait retrouver un lien partagé, la tâche devenait plus facile car chaque information était dans un salon distinct.

Pour garder une structure encore plus claire de notre travail, nous avons créé un *Google Drive* pour notre équipe et nous l'avons organisé de manière qu'il devienne notre espace de collaboration.

Google Drive est un service de stockage et de partage de fichiers dans le cloud lancé par la société Google. Ainsi on pouvait suivre de près chaque tâche finalisée car elle était déposée sur drive, et encore une fois, l'accès aux fichiers partagés se faisait beaucoup plus simplement.



Afin de suivre avec précision l'avancement du projet et des tâches de chacun, l'équipe a créé un workspace sur *Trello* et a attribué selon le diagramme de Gantt une to-do liste à chacun des membres.

Trello est un outil de gestion de projet en ligne gratuit qui repose sur une organisation des projets en planches listant des cartes, chacune représentant des tâches. Les cartes sont assignables à des utilisateurs qui sont mobiles d'une planche à l'autre, traduisant leur état d'avancement.



Mais la communication était tellement efficace à travers discord, que ce dernier s'est rapidement avéré inutile, on a donc arrêté de l'utiliser au bout de la cinquième semaine.

De plus, le travail se faisait énormément en équipe, dépendant chacun des connaissances des autres, et donc automatiquement, connaissait chacun l'état d'avancement des autres.

b. Avec les encadreurs :

Une réunion hebdomadaire est organisée avec l'encadreur durant laquelle on discute de toute éventuelle incompréhension ou problème rencontré.

Pendant la période en ligne, ces réunions se faisaient sur *Google Meet*.

Google Meet, est une plateforme de messagerie instantanée et de vidéoconférence développée par Google.



Le partage de fichiers et de documents avec le professeur encadrant se faisait à travers la plateforme *Classroom*.

Google Classroom est une plate-forme d'apprentissage dédiée aux écoles. Son but est de simplifier la création et la diffusion de cours et d'exercices de façon numérique, mais nous avons pu réadapter son utilisation selon nos besoins.

On pouvait ainsi poster sur notre classroom nos questions et nos PVs afin de recevoir les feedbacks du professeur.



3. Déroulement du projet :

Trois phases essentielles qui permettront de structurer la mise en œuvre de notre travail ont été établies :

a. Première phase : Recherche et documentation

Avant de se lancer dans la conception du projet, et pour une bonne prise de connaissance du sujet, des recherches approfondies sur notre thème, suivi par une analyse du cahier des charges transmis ont été effectuées. Pour la mise en œuvre du progiciel, nous nous sommes, également, documentés sur les outils techniques mis à notre disposition.

b. Deuxième phase : Conception

Sur la base de la recherche réalisée et la documentation collectée, la conception du projet a été entamée.

Durant cette phase on a décidé du paradigme de programmation le plus adéquat, on a donc opté pour la programmation orientée objet, car elle offre une meilleure structuration pour le code, ceci garantira la maintenance et l'évolutivité tels que grâce à l'indépendance des modules, la modification de l'un des constituants n'entraînera pas d'erreurs et n'affectera pas les autres constituants.

De plus, la structure de données a été définie, suivi par la réalisation d'un découpage modulaire permettant de créer un diagramme de classe qui détaillera la structure future de notre code.

c. Troisième phase : Réalisation

Cette phase a consisté à la concrétisation des éléments définis dans la phase de conception, en utilisant les outils techniques nécessaires et en commençant par la programmation, la création des fichiers de données, leur implémentations et enfin la mise en œuvre de l'interface graphique avec les changements adéquats, selon les cas.

V. Présentation du Sujet

1. Introduction

Avant de s'attaquer à la conception de notre programme, il est nécessaire d'effectuer des recherches concernant la méthode de corrélation, comment peut-on l'utiliser dans notre cas, quelle formule sera la plus appropriée et aussi comprendre exactement le concept de la reconnaissance des chiffres imprimés.

2. Qu'est-ce qu'une image numérique ?

Avant de commencer à parler sur l'image numérique, on parle d'abord de ce qui est numérisation, qu'est-ce que la numérisation ?

« La numérisation de données, images ou sons, consiste à découper ces données sous forme d'informations élémentaires et à coder chaque élément obtenu sous forme de nombres entiers ». [Ameisen, D. (2013). Qu'est-ce qu'une image numérique.]

« Une image sera ainsi découpée en une mosaïque de pixels de façon à la convertir en mode points, en conséquent, l'image numérique est un code qui est acquis, stocké et interprété et qui peut être représenté par une matrice, un tableau de valeurs avec des coordonnées (x, y), stockée sous forme d'un fichier, une cartographie de valeurs des pixels exprimées en bits ». [GAUDIN, Jacques. Qu'est-ce qu'une image numérique?. 2002.]

3. Qu'est-ce que la corrélation d'images numériques ?

« La corrélation d'images est une **technique pour mesurer le champ de déplacement d'une surface d'une image déformée par rapport à une image de référence.** » [Wikipédia]

“... La fonction de corrélation croisée (aussi appelée fonction de corrélation par abus de langage) est un opérateur qui agit sur deux fonctions ($f(x, y)$, $g(x, y)$), correspondant chacune à une image.” [El Marhoune Jamal, Compte rendu de conférence, 2012]

4. Cas d'études :

Dans notre projet il s'agit de reconnaître des chiffres imprimés représentés sur des images numériques en utilisant une méthode de corrélation résumée en une formule que nous devons programmer. Celle-ci serait appliquée sur deux images numériques représentées en matrice de dimension $M \times N$, l'une de référence et l'autre entrée par un utilisateur. Ces deux images seront de taille et de couleurs identiques (images binaires à deux niveaux) ce qui facilitera la tâche. Afin d'assurer une reconnaissance minutieuse et précise, on optera pour des méthodes de traitements de résultats et de calculs d'erreurs pour optimiser les performances.

VI. Conception

1. Introduction :

Une conception bien réfléchie implique une réalisation dans les meilleures conditions, c'est pourquoi nous avons accordé toute l'attention nécessaire à cette phase, durant laquelle nous avons défini la structure du projet et identifié les fonctionnalités en partant des plus importantes aux rudimentaires.

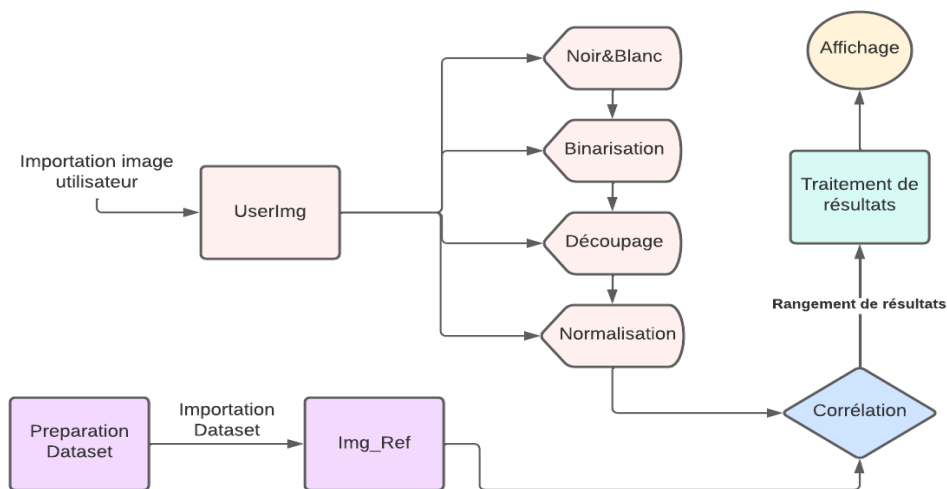
Les étapes de conception que nous allons aborder dans ce chapitre se résument alors ainsi :

- Présentation de la solution proposée ;
- Description de la structure de fichiers (base de données) conçue ;
- Description de la méthode de corrélation choisie ;
- Présentation du diagramme de classe ;
- Proposition de l'interface de notre application Desktop.

2. Développement :

- Solution proposée :

Elle se résume de manière globale, à travers le diagramme suivant :



- **Description de la structure de fichier (Base de données) conçue :**

Notre base de données n'est qu'un fichier nommé data-set et qui regroupe dix (10) fichiers en tout. Chacun de ces fichiers comportant des images d'un chiffre donné, et tous les fichiers contiennent le même nombre d'images avec des styles et des fontes différentes.

0	✓	4/4/2022 1:22 AM	File folder
1	✓	4/4/2022 12:24 AM	File folder
2	✓	4/4/2022 12:24 AM	File folder
3	✓	4/4/2022 12:24 AM	File folder
4	✓	4/4/2022 12:24 AM	File folder
5	✓	4/4/2022 1:31 AM	File folder
6	✓	4/4/2022 1:35 AM	File folder
7	✓	4/4/2022 12:24 AM	File folder
8	✓	4/4/2022 1:40 AM	File folder
9	✓	4/4/2022 1:20 AM	File folder

Evidemment, les images de notre data-set sont en noir et blanc, normalisées et découpées.



De la même manière, deux autres data-sets ont été conçus, l'un réservé aux lettres alphabétiques et l'autre aux caractères spéciaux. Ceci, afin de minimiser le taux d'erreurs et de faciliter le calcul des taux de faux positifs et de faux négatifs.

- **Description de la méthode utilisée :**

La méthode que nous avons adoptée, consiste à calculer le coefficient de corrélation qui n'est que le rapport entre la covariance et l'écart-type et ce, avec la formule suivante :

$$(1) \ r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}}$$

En appliquant cette formule entre deux images numériques, le résultat sera entre « -1 » et « 1 » et donc, pour un résultat égal à « 1 », les deux images entrées sont compatibles à 100%.

Dans le cas où on n'aurait pas d'images exactement compatibles, notre chiffre serait déterminé en fonction du coefficient qui est plus proche de « 1 », d'où la nécessité du rangement de résultats dans un tableau dynamique afin de définir le plus grand coefficient obtenu à la fin de l'opération.

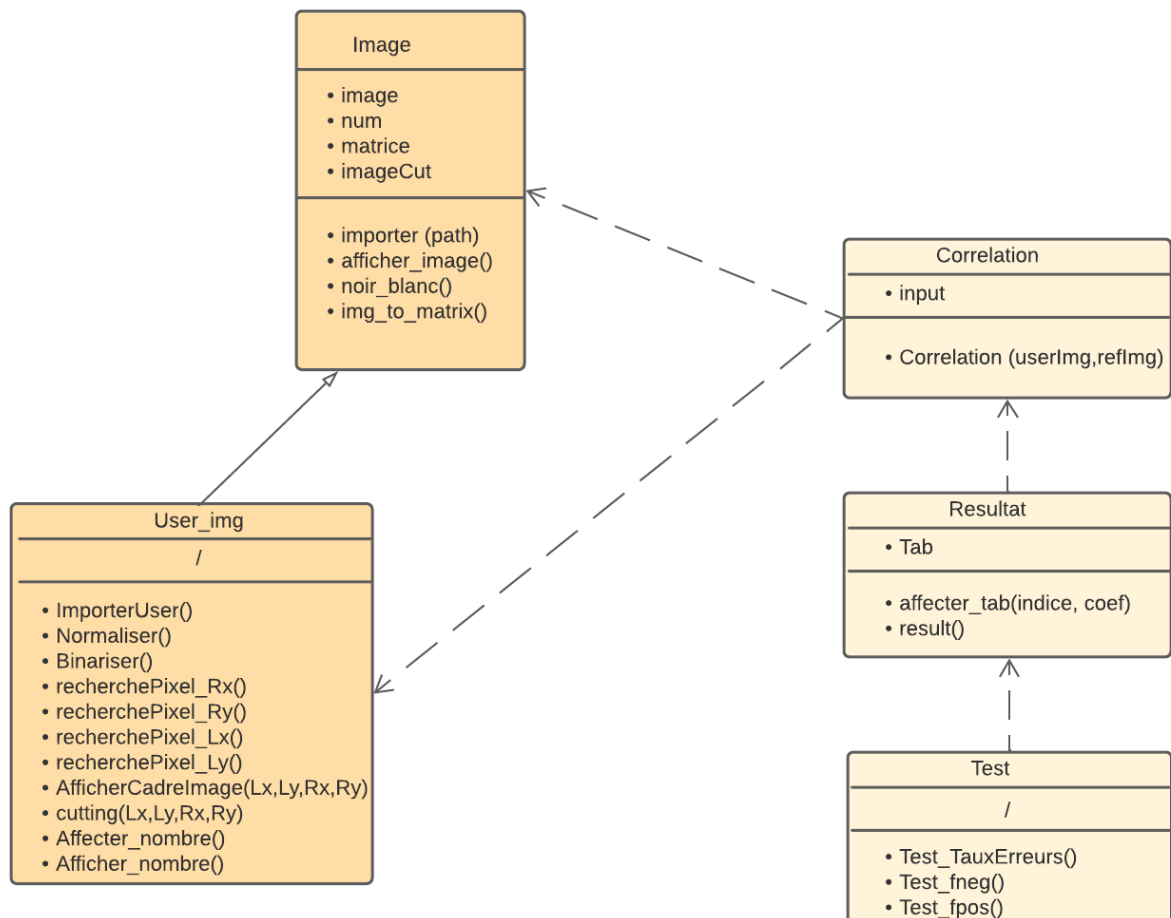
Pour éviter les erreurs et afin d'optimiser la solution proposée, on va déterminer un seuil (0.8 par exemple) de telle sorte qu'on ne considère pas les images générant un coefficient plus petit que celui-là.

Dans le cas où aucune image n'a un coefficient max supérieur au seuil déterminé, on se retrouve devant la possibilité que l'image entrée contienne un caractère qui n'est pas un chiffre. De ce fait, et afin de confirmer ce résultat, on applique la même formule (1) avec le data-set réservé aux lettres et aux caractères spéciaux. Si le coefficient obtenu est supérieur au premier coefficient max, on en déduit qu'effectivement, l'image entrée ne contient pas un chiffre.

En dernier lieu, et dans le but d'améliorer les performances du logiciel, on vérifie dans le tableau de résultats, après avoir obtenu le coefficient max, s'il y a un autre coefficient proche de ce dernier avec une différence de 0.02 au plus. Dans le cas affirmative, on calcule la moyenne des coefficients obtenus dans chaque dossier du data-set, et on choisit enfin, le chiffre correspondant au dossier dont la moyenne de ses coefficients est supérieure aux autres.

• **Présentation du diagramme de classes :**

➤ **Diagramme de classe :**



Ce diagramme ainsi que le diagramme de la solution proposée ont été réalisés à l'aide de *LucidChart*.

LucidChart est une plateforme de collaboration et de création de diagrammes basée sur le cloud pour créer et visualiser des données et d'autres diagrammes conceptuels.



➤ **Conception des classes et entités :**

Image :

Cette classe contiendra tous les attributs et éléments nécessaires pour le traitement d'images du Data-set

Class Image	
Liste des attributs	
Private Image	- L'image originale importée
Private Num	- Le chiffre représenté dans l'image
Private Matrice	- L'image binarisée
Private imageCut	- L'image découpée et dont on va travailler dessus
Liste des méthodes	
Public Void Importer ()	- Importer le Data-set sans la RAM
Public Void Afficher image ()	- Afficher l'image originale
Public Void Noir blanc ()	- Transformer l'image originale en noir et blanc
Public Void img to matrix ()	- Représenter l'image comme matrice

User_Img:

Cette classe est héritière de Image, elle est utilisée dans le contexte d'une image entrée par l'utilisateur seulement et donc elle contiendra les méthodes de traitement spécial de cette dernière.

Class User_Img Extends Image	
Liste des attributs	
/	
Liste des méthodes	
Public Void ImporterUser ()	- Importer une image du PC de l'utilisateur
Public Void Normaliser ()	- Normaliser l'image de l'utilisateur (64px)
Public Void Binariser ()	- Binariser l'image de l'utilisateur
Public Int RecherchePixels_Rx ()	- Rechercher et retourner le premier pixel valant 1 en haut droit de la matrice
Public Int RecherchePixels_Ry ()	- Rechercher et retourner le premier pixel valant 1 en bas droit de la matrice.
Public Int RecherchePixels_Lx ()	- Rechercher et retourner le premier pixel valant 1 en haut gauche de la matrice.
Public Int RecherchePixels_Ly ()	- Rechercher et retourner le premier pixel valant 1 en bas gauche de la matrice
Public Void AfficherCadreImg (Lx,Ly,Rx,Ry)	- Encadrer la partie essentielle de l'image
Public Void Cutting (Lx,Ly,Rx,Ry)	- Découper l'image
Public void Affecter_nombre ()	- Affecter le résultat à l'attribut Num de l'image
Public Void Afficher_nombre ()	- Afficher le résultat

Corrélation :

Cette classe est primordiale dans notre projet car la reconnaissance de chiffre dans notre cas se base sur le résultat retourné par la méthode de cette classe.

Class Correlation	
Liste des attributs	
Private Input	- L'image importée et traitée de l'utilisateur
Liste des méthodes	
Public Double Correlation (userImg, refImg)	- Calculer et retourner le coefficient de corrélation entre les deux arguments

Résultat :

Cette classe contient la structure de rangement des résultats de traitements de l'image avec chaque dossier du Data-set

Class Resultat	
Liste des attributs	
Private Tab	- Un tableau de dix (10) cases contenant les plus grands coefficients obtenus dans chaque dossier du Data-set
Liste des méthodes	
Public Void Affecter_Tab (indice, coef)	- Ranger les maximums des coefficients dans les cases correspondantes du tableau
Public Int Result ()	- Retourner le résultat final, le chiffre reconnu

Test :

Cette classe contient les méthodes de calculs de performances du logiciel

Class TestImg	
Liste des attributs	
/	
Liste des méthodes	
Public Double Test_fneg()	- Calculer le taux des faux négatifs
Public Double Test_tauxErreurs()	- Calculer le taux d'erreurs
Public Double Test_fpos()	- Calculer le taux des faux positifs

- **Charte Graphique et ergonomie de l'application :**
Pour l'identité visuelle, nous avons opté pour des couleurs simples et un minimal sans-serif font family.



3. Conclusion :

Après avoir défini la composition et les fonctionnalités de l'application à construire, et prévu une charte graphique couvrant l'aspect visuel de l'application, la deuxième phase de conception touche à sa fin laissant place à l'étape suivante, celle de la réalisation pour enfin concrétiser notre projet.

VII. Réalisation

Le chapitre suivant portera donc sur la phase de réalisation du logiciel. Elle sera divisée en deux grandes parties : le front-end et le back-end. Dans ce chapitre nous allons décrire le processus de travail des deux parties.

1. Front-End :

Après avoir finalisé la partie conception de notre outil, le processus de la réalisation des interfaces graphiques a été entamé, en commençant par un prototype qui a été réalisé à l'aide du logiciel *Figma*, en utilisant la charte graphique établie lors de la conception.

Figma est une application d'édition de prototypes et de graphiques vectoriels pour les équipes. Elle permet d'établir des systèmes de conception qui aident au développement pour les concepteurs et développeurs d'interface utilisateur et d'UX.



Pour le logo nous avons opté pour un logo vectoriel simple, qui rappelle immédiatement l'utilité de notre programme et qui utilise aussi les couleurs de la charte graphique.



Une fois ceux-ci finalisé, l'équipe se partagea les interfaces et commença à les réaliser avec *HTML5*, *CSS*, et *JavaScript* :

HTML5 est l'abréviation de " HyperText Markup Language “, qui se traduit par "langage de balises hypertexte". Il est utilisé pour générer et représenter les informations et la structure d'une page web.

CSS signifie "Cascading Style Sheets" (feuilles de style en cascade), ce langage informatique est utilisé pour la mise en forme des pages web (HTML ou XML).

JavaScript est un langage de programmation qui permet de construire des pages Web interactives dynamiques. Il est conçu pour gérer les événements qui se produisent localement.



2. Back-End :

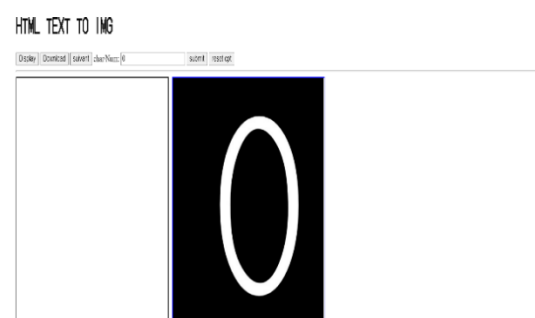
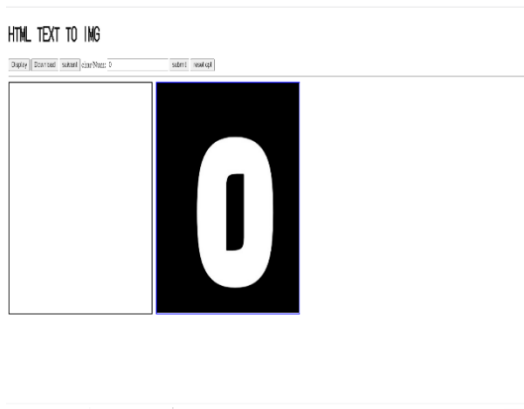
Lors de la réalisation de l'application Desktop, le Back-end est passée par les étapes qui suivent :

- 1) Implémentation de la Base de données.
- 2) Implémentation des différentes classes et méthodes sous Python.
- 3) Tests, calculs d'erreurs, et optimisation.
- 4) Intégration des interfaces avec le Back-end.

- **Implémentation de la Base de données :**

Pour réaliser le Data-set présenté au-dessus, on a procédé comme suit :

- La création d'une page html minimaliste liée à un code JavaScript qui consiste à générer à chaque clique un chiffre avec une fonte et un style différent.



- L'automatisation du processus de génération d'images avec le code suivant :

```
1 from tkinter.tix import Tree
2 from selenium import webdriver
3 from selenium.webdriver.chrome.options import Options
4 import time
5 option = Options()
6 option.add_experimental_option("debuggerAddress", "localhost:9222")
7 # driver = webdriver.Chrome(options=option)
8
9 driver = webdriver.Chrome(executable_path="chromedriver.exe")
10 driver.get("file:///C:/Users/celin/OneDrive/Bureau/index/index.html")
11 char="0123456789"
12 inputchar = driver.find_element_by_id("sampleText")
13 inputchar.clear()
14 inputchar.send_keys('5')
15 submit = driver.find_element_by_id("submit")
16 submit.click()
17 search_bar = driver.find_element_by_id("btnSuivant")
18 search_bar.click()
19 for i in range(120):
20     search_bar = driver.find_element_by_id("btnDisplay")
21     search_bar.click()
22
23     # search_bar = driver.find_element_by_id("btnDownload")
24     # search_bar.click()
25
26     search_bar = driver.find_element_by_id("btnSuivant")
27     search_bar.click()
28
29
30 input('Press ENTER to exit')
31
```

- La réalisation d'un autre script en python pour télécharger ces images générées, les traiter, et enfin les stocker dans des dossiers spécifiés.

```
def load(self):
    directory="dataset/"
    tab=[]
    print("loading data")
    for j in range(10):
        directory=directory+str(j)
        for filename in os.listdir(directory):
            f = os.path.join(directory, filename)
            self.importer(f)
            self.noir_blanc()
            # self.binariser()
            tab.append(self.get_imgcut())
        i=0
        directory="dataset/"
        print(str(j*10)+"%")
    print("100%")
    return tab
def load_tabchar(self):
    tabchar=[]
    directory="datasetchar"
    for filename in os.listdir(directory):
        f = os.path.join(directory, filename)
        self.importer(f)
        self.noir_blanc()
        # self.binariser()
        tabchar.append(self.get_imgcut())
    return tabchar
```

- **Implémentation des différentes classes et méthodes :**

L'équipe ayant choisi le paradigme orienté objet pour la réalisation de ce projet, nous avons opté pour l'utilisation du langage de programmation *Python*, que nous avons jugé idéal à nos besoins.

Python est un langage de programmation structuré et orienté objet. Il est doté d'un typage dynamique fort, et d'un système de gestion d'exceptions, ainsi il offre des bibliothèques riches en fonctions. Il est destiné à augmenter la productivité des programmeurs en fournissant des outils de haut niveau et une syntaxe simple.



Pour l'environnement, on a opté pour *VS Studio Code*, sachant que nous avons tous une petite expérience avec ce dernier

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et MacOS. Parmi ses fonctionnalités les plus intéressantes, la prise en charge du débogage, la mise en évidence de la syntaxe et la complétion intelligente du code.



- **Calcul de performances :**

- **Faux positifs :**

Un faux positif est lorsqu'on donne au logiciel une image quelconque sauf un chiffre et qu'il la reconnaît comme étant un chiffre.

Et c'est là justement où on perçoit l'importance du seuil déterminé, et par lequel on décide si la valeur de l'image sera acceptée ou rejetée.

Pour calculer le taux des faux positifs de notre logiciel, on a conçu un autre Data-set contenant que des images de lettres et de caractères spéciaux. Nous avons ensuite fait entrer ces images l'une après l'autre avec un algorithme permettant de parcourir ce Data-set et d'incrémenter un compteur toutes les fois qu'on a obtenu un coefficient supérieur au seuil déjà déterminé, par la suite, on calcule le taux des faux positifs par la formule suivante :

$$(2) \text{ Taux de faux positifs} = \frac{\text{compteur} \times 100}{\text{nombre total d'images du Dataset}}$$

- **Faux négatifs :**

Un faux négatif est lorsqu'on donne au logiciel une image contenant un chiffre et qu'il la rejette, justement parce que le coefficient obtenu a été inférieur au seuil déterminé.

Pour calculer le taux des faux négatifs de notre logiciel, on a conçu un autre Data-set de mille deux cents (1200) images différentes de ceux déjà existantes dans le premier Data-set et, suivant la même logique, un algorithme parcourant ce Data-set et calculant le nombre de coefficients inférieur au seuil déterminé est établi, pour que, à la fin on calcule le pourcentage qui n'est que le taux des faux négatifs par la formule qui suit :

$$(3) \text{ Taux de faux négatifs} = \frac{\text{compteur} \times 100}{1200}$$

➤ **Taux d'erreurs :**

Pour calculer le taux d'erreurs, on a utilisé le même Data-set utilisé dans le calcul de faux négatifs, sauf que dans ce cas, on incrémente le compteur lorsque le résultat obtenu est différent du résultat voulu. Donc le taux d'erreurs est calculé par la formule suivante :

$$(4) \text{ Taux d'erreurs} = \frac{\text{compteur de resultats erronés} \times 100}{1200}$$

• **Intégration des interfaces avec le Back-end :**

En utilisant la bibliothèque EEL de Python, on a défini de nouvelles fonctions en utilisant les méthodes déjà implémentées, et qui, ne seront que des valeurs de retours de ces fonctions.

```
eel.init(f'{os.path.dirname(os.path.realpath(__file__))}/web')

@eel.expose
def get_img_user():
    global imgtest1
    imgtest1 = User_img()
    return imgtest1.loaduser()

@eel.expose
def get_num():
    test = correlation()
    result=resultat()
    return result.result(tab,test,imgtest1,tabchar)
eel.start('/html/main.html')
```

Ensuite, on appelle ces mêmes fonctions avec un code JavaScript pour lier leurs valeurs de retour avec notre interface graphique comme le montre l'exemple suivant :

```
const affiche_num=document.getElementById('affiche_num')
const path_img=document.getElementById('img_user')

async function get_user_img_python(){
    if (await eel.get_img_user()==1){
        path_img.src= "../img/img_user.png" ;
        path_img.height="60px";
        path_img.style.visibility="visible";
        document.getElementById('img_user_div').style.visibility="hidden"
        tab = await eel.get_num();
        affiche_num.innerHTML= tab[0];
        affiche_num.style.color="black";
        affiche_num.style.fontSize="30px";
        fiabilitefunction(100*Number.parseFloat(tab[1]).toFixed(2));
        for (var i = 0; i < 20; i++) { tableau[i]=tab[i+2] }
        import_tf=true;
        generate_table();
    }
}

document.getElementById('import_bt').addEventListener('click',()=>{
    get_user_img_python();
})
```

Pour organiser le travail Back-end, l'équipe a choisi de mettre en place un espace de travail sur *Github* auquel l'ensemble des codes ont été ajoutés au fur et à mesure, toujours dans l'esprit de collaborer et afin de maintenir le maximum de transparence au sein de l'équipe.

GitHub est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git.



3. Améliorations :

Notre équipe a pensé à rajouter des fonctionnalités en plus à l'application tout en restant fidèle à l'approche utilisée précédemment :

- **Reconnaissance de lettres alphabétiques latines :**

Par l'établissement d'un autre data-set réservé aux lettres latines, et en suivant la même organisation du dernier data-set (c'est-à-dire 26 dossiers chacun contenant des images d'une lettre latine en styles et fontes différentes), on a pu appliquer la même méthode de corrélation et reconnaître, désormais, les lettres imprimés.

- **Reconnaissance de nombres :**

Puisque un nombre est composé de plusieurs chiffres, on a pensé à implémenter la reconnaissance de nombres aussi, et ce, en les décomposant en chiffres.

Pour ceci, on a importé de la bibliothèque cv2 de python la fonction de segmentation qui permet de donner les coordonnées de chaque chiffre dans une image pixélisée.

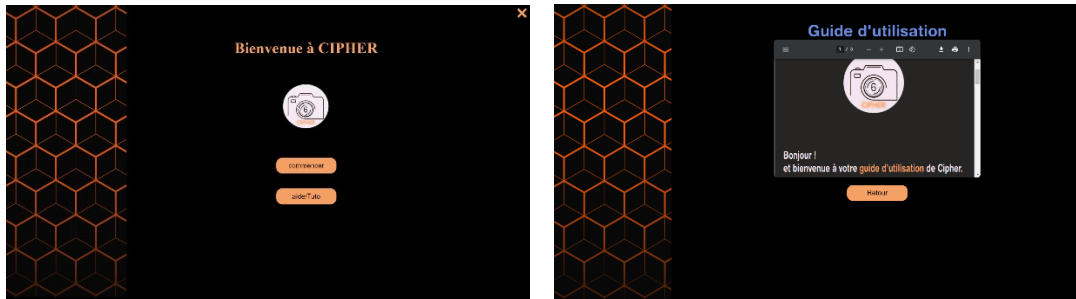
Ainsi, on procède par ranger ces coordonnées dans un tableau, de découper l'image pour obtenir chaque chiffre séparément, ensuite, de reconnaître ces chiffres par la méthode de corrélation utilisée précédemment et enfin, rassembler les chiffres reconnus pour obtenir le nombre correspondant à l'image entrée par l'utilisateur.

Enfin, avant de passer à la présentation de l'application, et après les calculs et les optimisations effectuées par l'équipe, les performances de notre logiciel se présentent comme suit :

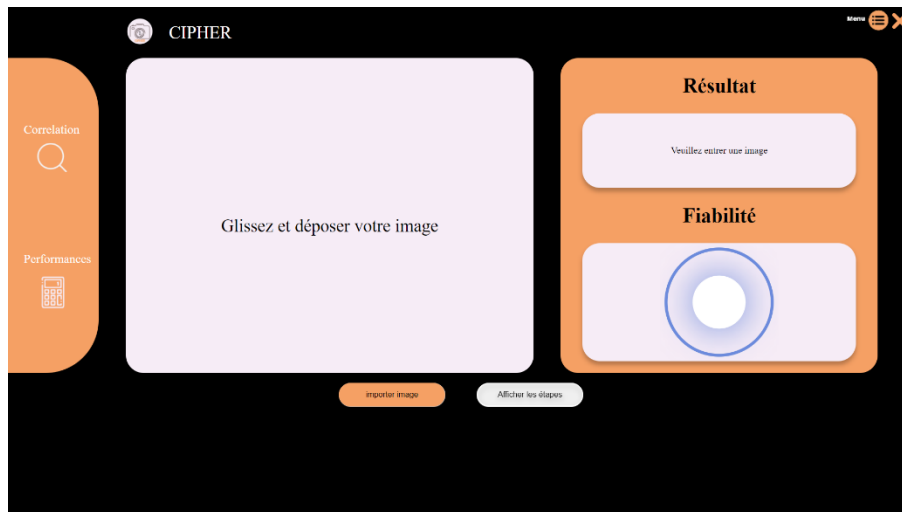
- Taux d'erreurs : 3.8% ;
- Taux de faux positifs : 6.57% ;
- Taux de faux négatifs : 0.75% ;
- Seuil déterminé : 0.7.

4. Présentation de l'application :

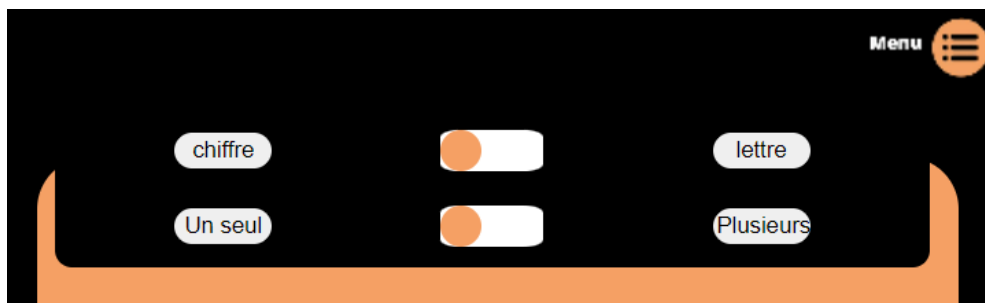
A l'ouverture de l'application, l'utilisateur aura le choix de visualiser un manuel d'utilisation :



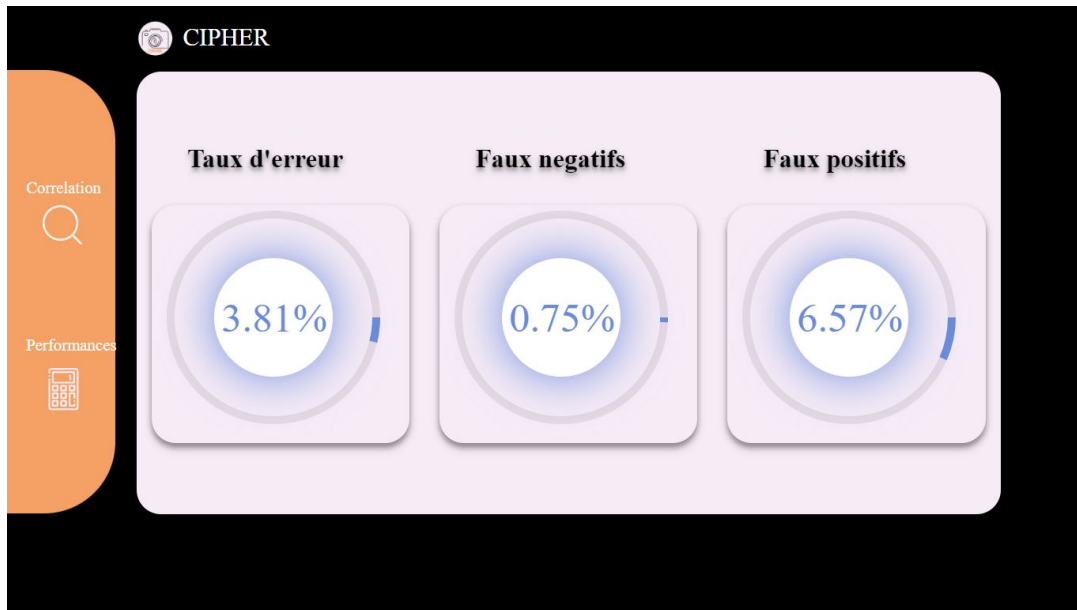
Ensuite, l'application s'ouvre sur la page principale comme présenté ci-dessous :



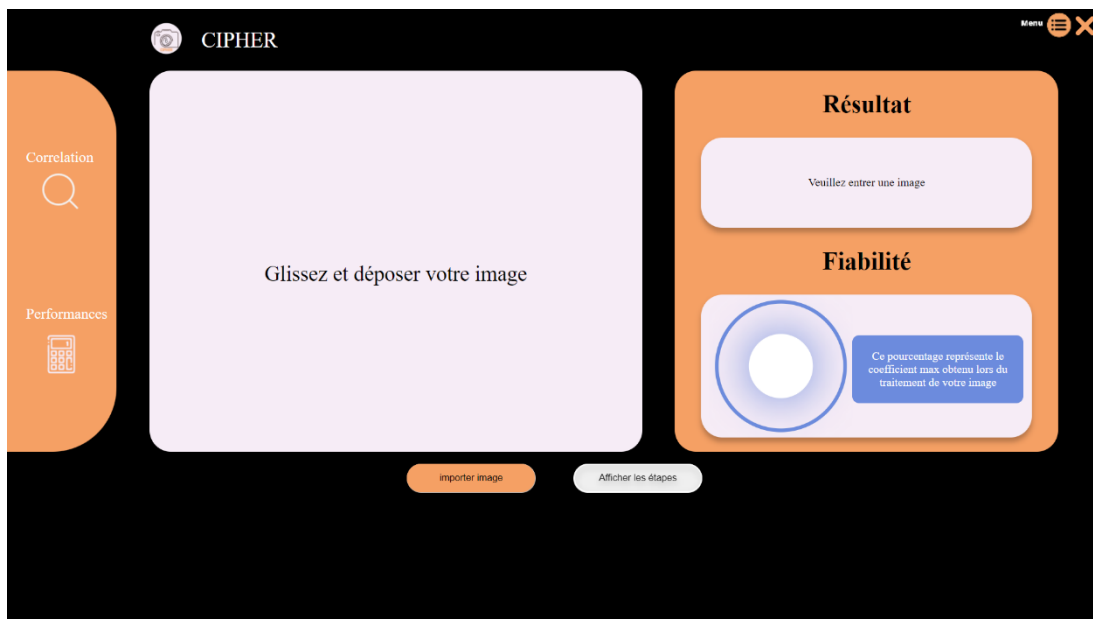
En cliquant sur le menu, l'utilisateur pourra choisir entre introduire des images contenant des chiffres ou bien des lettres, ainsi qu'activer ou désactiver la fonction de segmentation en cas de reconnaissance de nombres :



En cliquant sur « Performances » l'utilisateur pourra visualiser l'interface des performances de l'application :



Au cas où l'utilisateur a besoin d'aide, une aide sous chaque fonctionnalité sera affichée avec un simple clic droit de sa part expliquant le rôle de l'objet sous la souris.



VIII. Résumé :

Notre projet est axé sur la reconnaissance des caractères qui est pratiquée notamment en OCR – Optical Character Recognition -.

L'analyse du cahier de charges nous a menés vers la problématique suivante : concevoir un outil de reconnaissance de chiffres imprimés par la méthode de corrélation.

Après plusieurs recherches et concertations entre les membres de l'équipe, nous sommes arrivés à la méthode du coefficient de corrélation, c'est-à-dire calculer ce dernier avec une formule qui regroupe deux matrices de pixels (celle de l'image entrée avec chaque images de référence du dataset) pour qu'à la fin le résultat soit traité avec des algorithmes de traitement de résultats.

Nous avons pu atteindre les objectifs et les performances voulus notamment 3.8% de taux d'erreurs, un taux de faux positifs de 6.57% et un taux de faux négatifs de 0.75%.

IX. Conclusion Générale :

Durant les semaines passées à travailler sur ce projet, nous avons su user de nos compétences et les joindre afin d'aboutir à un produit final qui satisfait les objectifs fixés dans notre cahier de charge et ce grâce à une motivation et une détermination sans pareille ainsi qu'une excellente communication qui a fait que nous avons pu atteindre la plupart de nos objectifs en temps et en heure.

A l'issue de ce projet nous avons réalisé une application desktop « Cipher » qui répond à la problématique posée, en d'autres termes une application qui permet de reconnaître les chiffres imprimés par la méthode de corrélation

Dans ce rapport, nous avons détaillé les différentes étapes par lesquelles est passée la réalisation de ce projet : **une étude du cahier de charge** pour éliminer toute ambiguïté et bien définir les objectifs de l'outil et les attentes du client, l'organisation du travail où nous avons établi un planning à suivre pour les semaines de travaux qui étaient à venir. **Une phase conception** où nous avons posé les bases de notre solution, puis **une phase réalisation** où nous sommes préoccupés de la construction de l'application. Enfin nous avons effectué quelques tests et améliorations sur notre application puis apporté quelques modifications sur l'aspect visuel de cette dernière.

Le chemin ne fut pas des plus simples, nous avons fait face à la contrainte du temps ainsi qu'à des contraintes techniques mais nous avons très rapidement pu s'en défaire grâce à la collaboration entre nos membres et à l'esprit de partage qui s'est instauré au sein de l'équipe.

Pour conclure, ce projet nous a permis de vivre une immersion modeste dans la vie d'un ingénieur logiciel ce qui nous a permis d'étendre nos perspectives et connaissances ainsi que de nous trouver diverses nouvelles passions concernant le développement de solutions informatiques, découvrant où est-ce qu'on est doué et où est-ce qu'on l'est un peu moins.

Perspectives :

A travers cette expérience marquante, nous avons pu tracer des ambitions et des objectifs professionnels futurs à commencer par l'amélioration continue de ce que nous venons d'apprendre.

Nous songeons à améliorer notre logiciel, à travers la reconnaissance de types et de domaines d'utilisation des numéros entrés en s'aventurant dans le monde du Deep Learning, ainsi que la reconnaissance de texte contenant des lettres et des chiffres au même temps comme les codes hexadécimaux par exemple, et enfin, pouvoir tracer l'historique des utilisateurs pour un calcul continu de faux négatifs et positifs, et ce, afin de prouver, en quelque sorte, les performances de notre logiciel et essayer d'autres méthodes de corrélation pour des cas plus spécifiques permettant de minimiser ces taux d'erreurs au maximums.

Durant ce projet, nous avons réalisé une version desktop du logiciel, nous espérons pouvoir réaliser une version mobile pour permettre aux utilisateurs de l'application d'importer ou bien photographier les images avec leurs mobiles. Ainsi, de transporter cet outil partout et ne pas se limiter aux ordinateurs de bureaux.

Webographie et Références

Webographie :

OpenCV Documentation: [Geometric Image Transformations — OpenCV 3.0.0-dev documentation](#) .

Stack Overflow: <https://stackoverflow.com/>

MatLab: https://www.mathworks.com/help/images/ref/corr2.html?s_tid=srchtitle_corr2_1

Wikipédia : [Corrélation d'images — Wikipédia \(wikipedia.org\)](#)

Bibliographie :

Ameisen, D. (2013). Qu'est-ce qu'une image numérique. In *Conference Paper* (Vol. 57, pp. 169-172).

El Marhoune Jamal, S. N. (s.d.). Mesure des champs de déformation par la technique de corrélation d'images numériques. Ecole des mines Douai, Compte rendu de conférence, 2012

Gaudin, J. (2002). Article : médiamorphoses, Qu'est-ce qu'une image numérique?