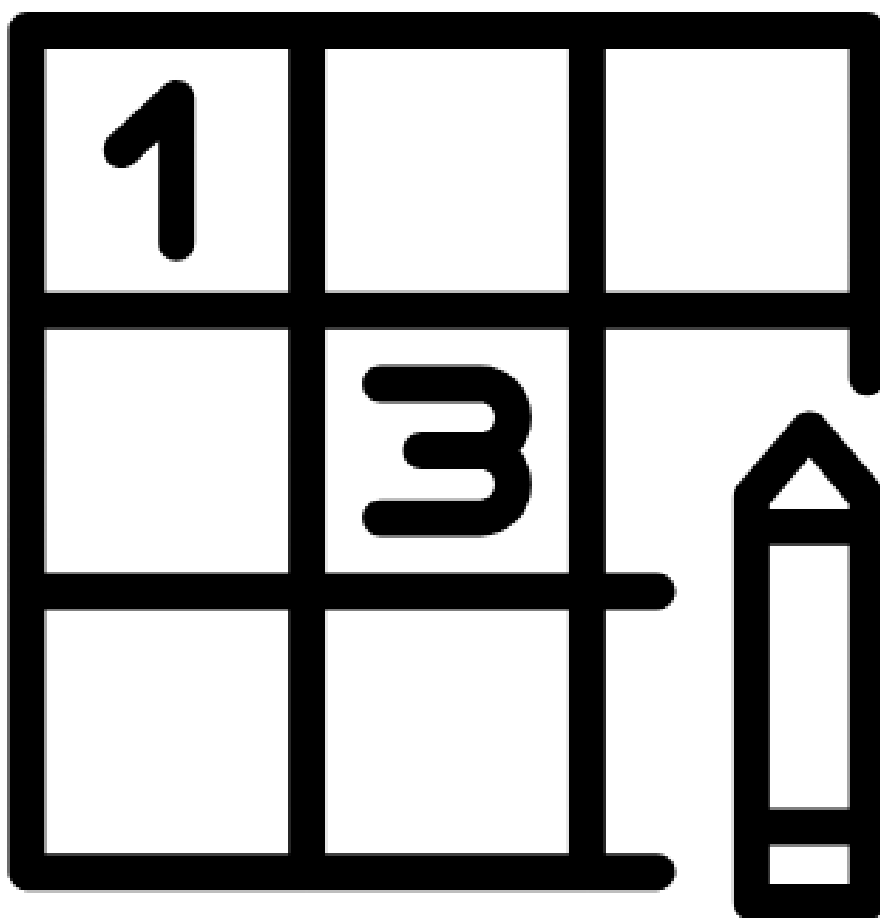


# Rapport de soutenance n°1

Les Hacheurs de C

Projet S3



## Contents

<b>1</b>	<b>Présentation OCR x Sudoku</b>	<b>3</b>
1.1	Introduction au projet . . . . .	3
1.2	Présentation des membres . . . . .	4
1.2.1	Valentin VEINANTE (C1) . . . . .	4
1.2.2	Alexandre BODIN (C1) . . . . .	4
1.2.3	Mark DELALOY (C1) . . . . .	5
1.2.4	Kevin PETIT (C1) . . . . .	5
<b>2</b>	<b>Etat du groupe (moral)</b>	<b>6</b>
2.1	Objectifs / Ambitions . . . . .	6
2.2	Gestion du travail d'équipe . . . . .	7
<b>3</b>	<b>Budget</b>	<b>8</b>
3.1	Coûts prévus . . . . .	8
3.2	Modèle économique (bénéfices attendus) . . . . .	8
3.2.1	Vente aux Sites de Génération de Sudoku : . . . . .	8
3.2.2	Intégration de la Publicité : . . . . .	8
3.3	Coûts finaux . . . . .	9
<b>4</b>	<b>Découpage du projet</b>	<b>10</b>
4.1	Les différentes parties du projet . . . . .	10
4.2	Répartition des tâches . . . . .	10
4.3	Planning visé au démarrage du projet . . . . .	11
<b>5</b>	<b>Développement et difficultés rencontrées</b>	<b>12</b>
5.1	Traitement d'image (70%) . . . . .	12
5.2	Grid Detection (90%) . . . . .	13
5.3	Sudoku Solver (100%) . . . . .	14
5.4	Réseau neuronal (35%) . . . . .	15
<b>6</b>	<b>Impressions du groupe</b>	<b>16</b>
6.1	Valentin . . . . .	16
6.2	Alexandre . . . . .	16
6.3	Mark . . . . .	17
6.4	Kevin . . . . .	18
<b>7</b>	<b>Conclusion</b>	<b>18</b>
7.1	Bilan . . . . .	18
7.2	Ce qu'il nous reste à faire... . . . . .	18



# 1 Présentation OCR x Sudoku

## 1.1 Introduction au projet

Le but fondamental de ce projet, baptisé "OCR Sudoku Solver", est de créer un logiciel sophistiqué capable de résoudre des grilles de sudoku à partir d'images.

Pour ce faire, le logiciel devra être en mesure de prendre en entrée une image de la grille de sudoku, de reconnaître les chiffres qui y sont présents, de résoudre le puzzle en appliquant des algorithmes spécifiques, puis d'afficher la grille résolue en sortie.

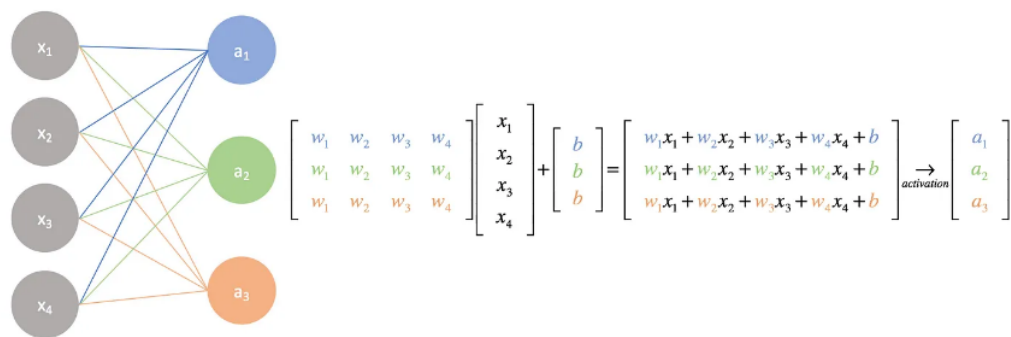
En somme, l'objectif est de développer un programme polyvalent qui peut effectuer toutes les étapes nécessaires pour passer d'une image de sudoku à une grille résolue, en utilisant des techniques d'OCR, de traitement d'image, d'apprentissage automatique (avec un réseau de neurones), et de résolution d'énigmes de sudoku.

Le projet inclut également la création d'une interface graphique conviviale pour faciliter l'interaction avec l'utilisateur. Ce projet nécessitera une collaboration efficace de quatre personnes pour atteindre ces objectifs ambitieux.

La difficulté est d'apprendre à l'ordinateur à identifier/reconnaitre les composants nous intéressant. Il devra les virtualiser pour pouvoir les manipuler afin de proposer une solution au problème posé, tout en gardant un aspect de simplicité pour l'utilisateur.

Input layer      Output layer

### A simple neural network



## 1.2 Présentation des membres

### 1.2.1 Valentin VEINANTE (C1)

Anciennement en Terminal Math/Physique-Chimie, je fais de la programmation depuis plusieurs années pour divers projets.

Ce projet S3 m'intéresse grandement car ce dernier nous permet d'avoir une introduction à l'intelligence artificielle ainsi qu'au Deep Learning.

Comme le projet S2, je pense qu'évoluer dans un environnement moins restreint que les TPs nous permettra de progresser sur divers aspects tel que l'autonomie, la technique et la cohésion de groupe.

### 1.2.2 Alexandre BODIN (C1)

Ayant effectué une Terminale Math-NSI, j'ai eu l'occasion de faire beaucoup de programmation depuis un moment. Cette 1ère année à EPITA a été source de nombreuses découvertes sur mes capacités en matière de réalisation de projet, de travail de groupe ou encore de connaissances acquises.

J'ai toujours été curieux de savoir comment une IA fonctionnait et la difficulté d'en réaliser une. J'ai donc beaucoup d'attentes envers ce projet.

Premièrement je m'attends à acquérir des compétences dans des domaines auxquels je ne me serais pas particulièrement intéressé de base.

Mais surtout j'espère qu'il nous permettra de nous dépasser et de finaliser un projet dont nous serons tous fiers.

J'aimerais également qu'il nous permette de nous améliorer non seulement techniquement mais aussi dans l'organisation de groupe.

Je suis assez curieux de voir combien de nos objectifs nous pourrions atteindre et je reste confiant dans l'issue de notre projet. Je nous pense capable de réussir ce défi comme nous avons su le faire jusque là !



### 1.2.3 Mark DELALOY (C1)

Anciennement en Terminal Spécialités Mathématiques/Physique-Chimie, option Maths Expert, section anglophone, au lycée St Vincent de Senlis (60300), je suis un grand passionné de volley, de psychologie mais surtout d'informatique.

Ce monde m'est cher car il a bercé ma plus tendre enfance. Mes premiers souvenirs étant de regarder mon père monter/démonter des ordinateurs et d'essayer, tant bien que mal, de l'assister.

De plus, l'univers des jeux vidéos m'avait tellement plu que l'un de mes plus grand rêve était de travailler en tant que Game Designer ou dans le Game Art. Cependant, je me suis rendu compte que la plupart des éditeurs utilisaient notre passion pour nous exploiter et que ces métiers étaient trop spécialisés.

C'est donc après cette désillusion que je me suis tourné vers l'EPITA et le métier d'ingénieur informatique, bien plus polyvalent et libre.

Vous pouvez donc imaginer que c'est avec énormément de motivation que je vais m'impliquer dans ce projet, afin de développer mes connaissances dans un domaine que je ne connais pas encore et qui permettra de développer cette fameuse polyvalence dont il est question.

Je suis persuadé que ce travail en groupe nous permettra de sortir plus grands, nous aidera à développer notre capacité d'adaptation, à faire des compromis, et finalement notre qualité d'écoute.

### 1.2.4 Kevin PETIT (C1)

Ancien élève de terminale Maths/NSI (avec option Maths Expertes), je suis paradoxalement plus intéressé par ce projet de S3 que par le projet de S2. Malgré son aspect restreint pouvant être repoussant au premier abord, avec un cahier des charges déjà fixé, je trouve que ce cadre nous permet de nous concentrer intégralement sur la résolution du problème posé. L'an dernier se posait parfois la question du cadre (que nous nous imposions à nous même), selon nos objectifs de développement (pouvant grandement différer d'un groupe à l'autre en fonction des concepts de jeu), ici le cadre est rigoureusement identique pour tout le monde.

En cela, ce projet est peut-être aussi plus proche du contexte de travail auquel nous serons confrontés plus tard en entreprise : un cahier des charges a été établi et nous devons nous assurer d'y répondre le mieux possible avec le programme que nous développons.

## 2 Etat du groupe (moral)

### 2.1 Objectifs / Ambitions

Le premier objectif du projet pour tous les membres du groupe est d'apprendre plus sur la création d'un OCR. Etant des personnes de nature extrêmement curieuse, nous voyons ce projet de S3 comme étant une réelle opportunité de comprendre le fonctionnement de cet outil que l'on peut déjà avoir été amené à manipuler par le passé.

Le second objectif, plus scolaire, consiste à consolider nos bases en programmation. Avant de commencer notre projet, nous avons déjà la certitude que le temps passé sur ce dernier sera bénéfique à notre apprentissage et permettra de consolider considérablement nos bases.

De plus, pour la majorité d'entre nous, les connaissances/notions utiles pour la création de l'OCR, le Sudoku Solver et le Traitement d'Image étaient inexistantes, obligeant un réel travail de recherche en amont.

Pour la plupart d'entre nous, ce projet représente une seconde opportunité exigeant autant de travail sur une si courte période. Expérimenter de nouveau le travail en équipe, contrainte à laquelle nous serons exposés tout au long de notre future carrière d'ingénieur en informatique est nécessaire. Cela sera aussi une opportunité pour tous, d'apprendre à travailler la coordination d'un groupe tout en tenant les échéances imposées par un planning donné en amont.

Nous avons donc de grandes ambitions pour ce projet dans lequel nous nous investissons pleinement.



## 2.2 Gestion du travail d'équipe

Pour travailler sur chacun des points du projet nous avons formé 2 duos : Mark/Kevin et Alexandre/Valentin.

Chacun des duos a développé un point qu'ils avaient en commun puis à intervalle régulier, les deux équipes se sont retrouvées pour mettre à jour le travail effectué.

Ainsi chacun a pu se concentrer sur son travail personnel tout en faisant avancer l'ensemble du projet.

La méthode par duo a permis de se motiver mutuellement car la solitude pouvait facilement nous décourager. Cela permettait également d'alléger la charge de travail assez lourde sur les points principaux de développement et de la répartir équitablement en subdivisant alors un travail, énorme en apparence, en de plus petites tâches. Le duo permettait, en plus, un contrôle mutuel.

Cette méthode a permis ainsi de progresser à intervalles réguliers et d'avancer sur l'ensemble des points du programme simultanément et en cohésion avec l'équipe.



### 3 Budget

#### 3.1 Coûts prévus

Le tableau ci-dessous n'est pas exact, certes, mais permet d'avoir une approximation de la valeur de nos dépenses à prévoir pour le projet. Ce dernier sera complété au fur et à mesure de l'avancement du projet, si d'autres ressources sont jugées nécessaires.

Services	Budget
Transports	210 x 4 = 840
Internet	140 x 4 = 560
Stockage	1 To - SSD 500 Go SSD 512 Go
OS	Linux Windows 11 Windows 10

#### 3.2 Modèle économique (bénéfices attendus)

Ce plan d'affaires se concentre sur la monétisation de l'OCR x Sudoku Solver, en mettant en avant deux principaux axes :

##### 3.2.1 Vente aux Sites de Génération de Sudoku :

- L'OCR Sudoku Solver sera proposé sous forme d'offre premium aux sites web qui proposent des générateurs de sudoku.
- Tarifs compétitifs basés sur le nombre d'utilisateurs potentiels des sites partenaires.
- Personnalisation de l'application pour refléter la marque du site web partenaire.

##### 3.2.2 Intégration de la Publicité :

- Intégration de publicités non intrusives dans l'application pour la version gratuite.
- Sélection de publicités en adéquation avec les intérêts des utilisateurs.
- Optimisation constante de l'emplacement et du type de publicités pour maximiser les revenus.





### 3.3 Coûts finaux

Pour faire un bilan final de nos bénéfices il nous faut d'abord nous rendre compte de ce que notre projet nous aura coûté en tout.

Services	Budget
Caféine	$15\text{€} \times 4 = 60\text{€}$
Électricité + chauffage	$80\text{€} \times 4 = 320\text{€}$
Transports	$85\text{€} \times 2 = 170\text{€}$
Heures de sommeil	$40\text{h} \times 4 = 160\text{h}$

Si nous considérons qu'une heure de sommeil revient à  $14\text{€}$  le coût total de notre projet nous revient donc à  $14\text{€} \times 160 + 60\text{€} + 320\text{€} + 170\text{€} = 2\,790\text{€}$ .

En combinant la vente aux sites de génération de sudoku avec l'intégration de publicités, nous cherchons à maximiser les revenus tout en offrant une expérience de qualité aux utilisateurs. Cette approche offre une opportunité solide de monétisation pour l'OCR Sudoku Solver.

Cependant, l'aspect extrêmement spécifique et niche du programme peut entraîner un problème de demande. Il risque d'y avoir trop peu de demandes pour que ce système soit viable. Il faudra alors l'adapter aux vues de la situation.



## 4 Découpage du projet

### 4.1 Les différentes parties du projet

- Traitement d'image: appliquer différents filtres de convolution + identifier le sudoku dans l'image + découper le sudoku en 81 cases.
- Réseau de neurones : réseau neuronal permettant d'identifier les numéros dans les 81 cases découpées + recréer le sudoku de l'image dans un tableau de 81 cases.
- Sudoku Solver : algorithme permettant de résoudre le sudoku donné en paramètre et renvoyant le résultat sous la forme d'un tableau de 81 cases.
- Interface : esthétique du logiciel + permettre d'appliquer tous les filtres de façon simple et intuitive.
- Site Web : présentation du projet + infos sur ce dernier.

### 4.2 Répartition des tâches

Nous avons essayé de nous répartir au mieux les tâches citées précédemment en fonction de leur complexité et de la durée qu'elle prennent à être effectuées: des ajustements dans la répartition pourront également être faits au cas où certaines tâches s'avèrent plus difficiles ou chronophages que prévues.

Voici ci-dessous notre planning de répartition actuel.

	Valentin	Alexandre	Mark	Kevin
Site Web			X	♡
Traitement d'Image			X	♡
Réseau neuronal	♡	♡		
Sudoku Solver			♡	
Interface	X	♡	X	X

Légende : ♡ = Responsable de la tâche ; X = Travaille sur la tâche



### 4.3 Planning visé au démarrage du projet

Nous avons essayé de répartir les tâches de manière équitable, non pas en termes de complexité, mais en termes de temps de travail.

	Soutenance 1	Soutenance 2
Site Web	0%	100%
Traitement d'Image	90%	100%
Réseau neuronal	30%	100%
Sudoku Solver	100%	100%
Interface	0%	100%

Légende : *Espérance d'avancement sur chaque tâche avant la première soutenance*



## 5 Développement et difficultés rencontrées

### 5.1 Traitement d'image (70%)

Pour implémenter les traitements nécessaires à notre projet, nous avons retenu la bibliothèque SDL Image. Il a donc tout d'abord fallu mettre en place des fonctions de chargement d'images, permettant de stocker celles-ci dans des surfaces SDL préalablement à l'application des traitements.

Une fois le terrain correctement préparé, la conversion en niveaux de gris a été la première fonction à coder. Son implémentation a été rapide, ne nécessitant que l'application d'une formule pondérant l'intensité des composantes rouge, verte, et bleue de chaque pixel.

En revanche, la conversion en noir et blanc, pouvant pourtant paraître simple au premier abord, a nécessité bien plus de réflexion afin de faire face aux problématiques rencontrées.

Une première version naïve avec un threshold fixe a été implémentée : tout pixel inférieur au threshold devenait noir, sinon il devenait blanc. Cette version (avant tout réalisée à des fins de test) a montré ses limites évidentes. Seules les premières images bénéficiaient d'une conversion correcte, le threshold étant alors tantôt trop bas, tantôt trop haut pour les images plus complexes.

La méthode d'Otsu a alors été choisie pour améliorer la conversion en noir et blanc. Cette méthode procède à une analyse de l'image et détermine, grâce à un histogramme des niveaux de gris, quel est le threshold optimal à appliquer sur l'image. Cette méthode a alors abouti sur des résultats corrects sur toutes les images citées en exemple, sauf la 4e. En effet, la méthode d'Otsu ne fait que déterminer un threshold global à appliquer à toute l'image. Or, l'image 4 présente une grande disparité de contraste et de luminosité, ce qui produit alors du noir indésirable au niveau des zones d'ombre après application d'Otsu.

Une fonction de normalisation a donc été implémentée, dans l'espoir de mieux répartir les niveaux de gris sur l'ensemble du spectre pour améliorer le traitement. Cette fonction analyse l'histogramme des niveaux de gris de l'image et "normalise" ces niveaux de gris en les étalonnant de manière plus équilibrée entre 0 et 255, améliorant ainsi le contraste de l'image. Cependant, cela n'a pas permis d'améliorer la conversion en noir et blanc. (Cette fonction a tout de même été conservée dans le code source pour sa possible utilité dans de futurs traitements/filtres).

La fonction désormais retenue pour le noir et blanc est la méthode de Sauvola. Contrairement à Otsu qui ne définit qu'un threshold global, Sauvola détermine un threshold par bloc de pixels analysées. Cela permet ainsi de séparer le traitement de zones aux luminosités différentes et de rendre un meilleur résultat malgré les "défauts" de l'image originale.

Pour améliorer la conversion en noir et blanc et également pour se préparer à la détection des bordures, deux fonctions de flou gaussien ont été réalisées. Ces deux méthodes de flou gaussien se basent sur le principe des filtres de convolution en opérant sur des matrices : l'une en 3x3, pour un léger flou, l'autre en 5x5, pour un flou un peu plus intense. Le flou gaussien nous permet de réduire le bruit de l'image préalablement à l'application des traitements nommés précédemment. Le résultat de la chaîne de traitement est ainsi amélioré.

Une fonction de rotation et une fonction de redimensionnement ont été implémentées. La fonction de rotation permet notamment le traitement de la grille d'exemple n°5. La rotation crée une nouvelle surface de la même dimension que l'originale, puis applique des règles de trigonométrie afin de localiser l'emplacement du pixel source à copier dans le pixel de l'image destination. Ainsi l'on est sûr que, malgré les arrondis des fonctions trigonométriques, tous les éléments de la destination auront forcément une source.

La rotation faisant disparaître une partie des bords de l'image, le redimensionnement permet à l'utilisateur d'élargir les bordures si nécessaire avant la rotation, afin de conserver la totalité de la grille de Sudoku dans le cadre de l'image.

Enfin, pour effectuer la détection de la grille (et par la suite le découpage de celle-ci en images individuelles), la méthode de la transformation de Hough a été retenue.

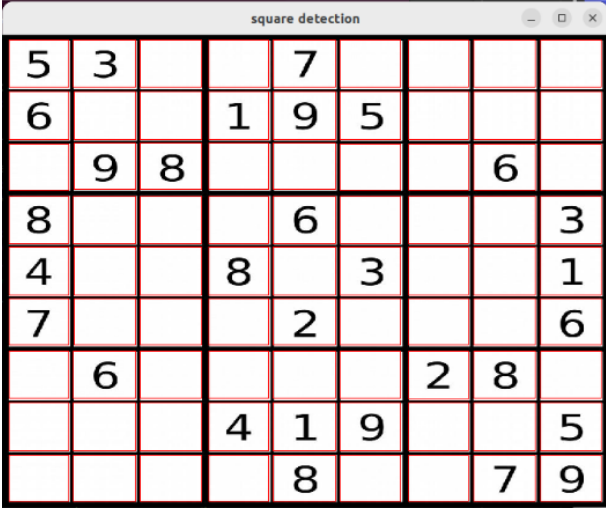
## 5.2 Grid Detection (90%)

Nous avons réalisé un programme permettant de faire le lien entre l'ensemble du prétraitement, qui nous permet d'obtenir une image exploitable, et l'OCR qui reconnaît le chiffre d'une image.

Pour ce faire nous avons opté pour une simple détection de carrés.

A l'aide de 2 fonctions, nous pouvons détecter les contours d'un carré dans l'image et lister tous les carrés que nous avons trouvés. Il nous suffit ensuite de parcourir cette liste et de garder les carrés faisant une certaine taille (pour qu'ils correspondent à des cases du sudoku).





5	3			7			
6			1	9	5		
	9	8					6
8				6			3
4			8		3		1
7				2			6
	6					2	8
			4	1	9		5
				8			7
						7	9

Une fois ces carrés détectés, nous prenons les tailles du plus petit (pour simplifier les images) et nous sauvegardons toutes les sous-images contenues dans les carrés dans un dossier d'extraction. Ces images pourront donc être exploitées par l'OCR par la suite.

Si jamais il y'en a plus ou moins que 81, nous pourrions ajuster les détéctions ou les filtres mais pour le moment les paramètres sont identiques pour toutes les cases.

### 5.3 Sudoku Solver (100%)

Afin de réaliser le Sudoku Solver, nous nous sommes basés sur le principe de "backtracking" traduisible par "retour en arrière".

Le principe de l'algorithme est plutôt simple à comprendre.

Le backtracking consiste dans la résolution par approche récursive afin de remplir les cases vides de la grille. Pour cela, L'algorithme commence par choisir un chiffre possible pour une case vide, en veillant à ce qu'il n'entre pas en conflit avec les chiffres déjà présents dans la même ligne, colonne et carré tous associés. Ensuite, il explore de manière récursive les cases suivantes, répétant ce processus.

Lorsque l'algorithme détecte une erreur, par exemple, un chiffre en conflit avec un autre dans la même unité (ligne, colonne ou carré), ou lorsqu'il atteint une impasse, il revient en arrière (backtrack) à la dernière décision prise, c'est-à-dire la case précédemment remplie.

À ce stade, il modifie le chiffre choisi et reprend l'exploration récursive à partir de ce point. Cette séquence se répète jusqu'à ce que la grille soit complètement remplie sans erreur ni impasse, assurant ainsi une solution valide respectant les règles du sudoku soit retenue.

En créant un algorithme basé sur cette méthode, nous nous sommes assurés d'obtenir des résultats cohérents et rapides.

Bien qu'il existe d'autres algorithmes un peu plus optimisés que celui-ci, le gain



de temps d'exécution a été jugé négligeable pour une implémentation qui se serait avérée bien plus complexe.

#### 5.4 Réseau neuronal (35%)

Pour le réaliser, le travail a été décomposé en plusieurs étapes dont la première était bien entendu la compréhension du fonctionnement d'un réseau neuronal (Racourci NN). Cela étant passé par la lecture de multiple articles et documentations afin de pouvoir appréhender ce sujet vaste et complexe.

Après avoir compris les grandes lignes nous nous sommes donc lancés dans la conception puis l'implémentation d'un NN permettant de réaliser une fonction Xor. Ce réseau est composé de trois couches distinctes, l'Input, une couche cachée et l'Output. De par sa simplicité, l'implémentation de ce réseau se fit rapidement et sans grande difficulté.

Nous nous sommes donc lancés avec confiance dans l'implémentation du NN servant d'OCR et ce en partant de la base solide que représentait le Xor. Nous l'avons donc adapté afin que ce réseau puisse fonctionner avec des arrays, nous permettant de gérer un grand nombre de poids et de neurone. Ce dernier est, lui aussi, composé de trois couches distinctes dont une couche cachée et ce dernier est réalisé sur une base de fully-connected-network.

C'est à dire que l'entièreté des noeuds d'une couche sont connectés avec l'entièreté des noeuds de la couche suivante.

Quant à la fonction d'activation, celle actuellement utilisée est une simple sigmoïde (ainsi que sa dérivée pour la correction des poids lors de la phase de Back propagation).

Malheureusement ce réseau est beaucoup plus complexe à implémenter et son développement c'est avéré extrêmement chronophage à cause du fait que le débogage d'un réseau neuronal est très fastidieux et ce malgré sa proximité de conception avec le réseau neuronal Xor vu plus haut.

Le débogage du NN de l'OCR est d'abord passé par l'ajout de biais, qui n'étaient pas utilisés auparavant. Ces derniers ont permis à l'OCR de reconnaître un chiffre unique lorsque l'IA a été entraîné dessus mais cela ne lui a malheureusement pas octroyé la capacité de faire une prédiction viable lorsqu'elle a été entraînée sur un ensemble de chiffres.

A ce jour, nous n'avons malheureusement pas encore réussi à réaliser une prédiction viable avec cette IA mais nous avons la conviction d'y parvenir prochainement.

## 6 Impressions du groupe

### 6.1 Valentin

Sur la période précédant la première soutenance j'ai été chargé de l'élaboration et de l'implémentation des réseaux neuronaux, durant m'a tâche j'ai eu la chance d'être assisté par Alexandre qui c'est avéré être d'une grande aide.

Comme dit précédemment dans la partie sur le réseau neuronal, l'implémentation de celui ci réalisant une fonction Xor a été relativement simple.

La véritable difficulté résidant dans le coeur de ce projet : L'OCR.

Le réseau neuronal dédié à l'OCR a pour but de reconnaître un chiffre sur une image afin de le traduire en entier pour l'ordinateur.

Ce dernier a requis la majeure partie de notre temps à Alexandre et à moi de par les nombreux problèmes rencontrés et la difficulté de débogage.

Malgré toutes ces difficultés, je suis très heureux d'avoir été affecté à l'élaboration des réseaux neuroneaux car cela m'a permis d'explorer un nouveau pan de l'informatique que je n'aurais sûrement pas exploré autrement.

Grâce à cela j'ai pu découvrir et apprendre énormément sur le fonctionnement des réseaux neuroneaux, du deep learning et plus généralement des IAs.

Ce projet m'aura permis d'élargir mon champ de compétence et de compréhension de l'Informatique. surtout avec les possibilités infinies que représente l'IA et ce en peu de temps.

Le réseau neuronal mis à part, j'ai aussi participé, avec Mark et Alexandre. Son élaboration était elle aussi assez compliquée et chronophage mais nous avons réussi (en grande partie grâce à la mise en commun des idées de chacun) à résoudre ce problème. Cela est très motivant pour la suite et j'espère que l'on réussira, pour la deuxième soutenance, à faire fonctionner l'OCR à l'instar de la détection de grille. Pour conclure, je peux dire sans crainte que, malgré les difficultés rencontrées, notre groupe arrivera à surmonter ces problèmes grâce à sa grande cohésion afin de fournir un travail pour la deuxième soutenance dont nous pourrions également être fier.

### 6.2 Alexandre

Mon rôle était le développement avec l'aide de Valentin d'un réseau neuronal capable de reconnaître une porte XOR, puis d'améliorer ce réseau afin de créer le centre de ce projet : l'OCR (Optical character recognition) soit un algorithme apprenant par lui-même à comprendre quel chiffre est représenté dans une image.

Avec Valentin nous avons été sujets à de nombreux problèmes, les réseaux neuronaux étant extrêmement complexes à corriger, réajuster et debugger s'avère être une tâche plus que difficile surtout pour nous qui n'avons jamais eu l'occasion de développer ce genre de logiciel.



Cela dit, cela nous a permis d'acquérir des compétences uniques et une compréhension des IAs étonnantes. Je fus ravi d'avoir pu aller jusque là pour cette 1ere soutenance même si les difficultés rencontrées m'ont un peu démoralisé.

Les réseaux de neurones restent passionnants et impressionnants et je suis ravi d'avoir pu les découvrir au travers de notre travail.

J'espère que nous allons aller de l'avant et utiliser ces erreurs comme une force pour avancer de plus belle et réaliser un projet que nous n'avions pas soupçonné être capable de faire il y a quelques mois.

J'ai confiance en notre groupe et en notre persévérance pour la prochaine soutenance.

### 6.3 Mark

Pour cette première soutenance j'ai entièrement géré la partie Sudoku Solver.

Cela n'était pas d'une difficulté extrême mais il faut admettre que c'était une tâche extrêmement chronophage. En effet, j'ai du coder entièrement le SudokuSolver 2 fois car j'étais parti d'un mauvais axiome.

En effet, je pensais qu'il n'existait qu'une seule et unique façon de résoudre un sudoku.

Quelle ne fut pas ma surprise, une fois que je pensais en avoir terminé, que de me rendre compte que mon programme ne fonctionnait que dans très peu de cas.

Il était alors plus simple de tout recommencer plutôt que de réadapter entièrement ce dernier.

De plus, m'étant mieux renseigné et ayant gagné de l'expérience en C au cours de l'année, j'ai décidé d'utiliser la méthode de backtracking (expliquée précédemment) afin de refaire un code, cette fois-ci, entièrement fonctionnel.

Pour autant, ce premier échec n'a pas été perçu comme tel: me tromper m'a permis de mieux comprendre le fonctionnement de l'algorithme et m'a obligé à acquérir de nombreuses autres connaissances.

En plus de cela, j'ai dû m'intéresser au traitement d'image afin d'essayer de comprendre comment on pouvait reconnaître la grille du Sudoku, ce qui se révélera être une tâche beaucoup plus ardue. Nous nous sommes tous retrouvés à passer plusieurs nuits à réfléchir, et à essayer de trouver des solutions à ce problème.

Après être passé par de nombreux états d'âmes (désespoir, fatigue mentale, colère...) c'est la joie et la fierté de voir enfin notre travail fructifier qui restera la conclusion de cette première soutenance.

Voir le fruit de nos travaux mis bout-à-bout et se compléter les uns les autres a été une source de joie et d'enthousiasme telle, que la motivation n'est que plus grande. Faire notre propre OCR est un rêve à porté de main et je suis fier d'être arrivé jusque là.



## 6.4 Kevin

J'ai été pour cette première période de développement principalement en charge du traitement de l'image.

Ce travail a nécessité beaucoup de documentation en ligne sur les nombreux filtres et autres transformations graphiques existant déjà.

Il était inutile d'essayer de réinventer la roue quand tant de développeurs et de mathématiciens talentueux avaient déjà conçu et théorisé les algorithmes nécessaires pour résoudre notre tâche.

Une grande partie du travail a alors nécessité la compréhension de pseudo-code et de formules mathématiques afin de réaliser leur implémentation avec la bibliothèque SDL Image. Là où d'autres bibliothèques (non accessibles pour ce projet) telles qu'OpenCV offrent à l'utilisateur beaucoup de traitements pré-programmés, SDL ne donne que peu de solutions clés en main au programmeur. Il a donc fallu se charger d'implémenter les différentes méthodes de traitement d'images utiles à notre projet en manipulant les images et leurs composantes pixel par pixel.

La conversion en niveaux de gris a été très simple à implémenter, tout comme le flou de Gauss, qui reste assez trivial à comprendre. En revanche l'un des éléments donnant le plus de fil à retordre a été la conversion en noir et blanc, pour laquelle différents essais avec plusieurs algorithmes de traitements ont été nécessaires avant d'arriver à des résultats corrects sur la majorité des images.

L'autre élément complexe à gérer a été la reconnaissance de la grille et son découpage. Pour réaliser cette tâche difficile, j'ai donc été en partie assisté par mes coéquipiers.

## 7 Conclusion

### 7.1 Bilan

### 7.2 Ce qu'il nous reste à faire...

En conclusion, nous pouvons dire qu'au travers de tous les points abordés, nous allons réussir à faire notre OCR, nous rendant fiers et étant suffisamment stable pour être considéré comme durable.

Chaque membre du groupe a décidé de se spécialiser en acquérant un maximum de connaissances pour une bonne réalisation et surtout un bon rendu du projet afin d'optimiser un maximum le temps qui nous était imparti.

Le projet est pour nous le moyen d'enrichir notre connaissance de la programmation tout en développant un concept. C'est avec une volonté de redoubler d'efforts que nous sommes certains de réaliser un OCR qui nous rendra fiers.

Nous prévoyons de développer pour la prochaine soutenance: Toute l'interface graphique du logiciel, améliorer nos filtres et, s'il nous reste du temps, créer le site internet présentant le projet.