

# Rapport de Soutenance 1

Novembre 2023

## OCR Sudoku Solver

par Toujours à l'heure

Djallil Baizid	Dorian Wolff
Simon Arbiser	Rémi Brenaut

# Table des matières

<b>I</b>	<b>Introduction</b>	<b>4</b>
<b>II</b>	<b>Membres de l'équipe</b>	<b>5</b>
<b>1</b>	<b>Login</b>	<b>5</b>
<b>2</b>	<b>Présentation</b>	<b>5</b>
2.1	Djallil . . . . .	5
2.2	Dorian . . . . .	6
2.3	Simon . . . . .	6
2.4	Rémi . . . . .	6
<b>III</b>	<b>Récapitulatif du cahier des charges</b>	<b>7</b>
<b>3</b>	<b>Le Projet</b>	<b>7</b>
<b>4</b>	<b>Répartition des tâches</b>	<b>8</b>
<b>5</b>	<b>Calendrier et avancement des tâches</b>	<b>9</b>
<b>IV</b>	<b>Projet</b>	<b>10</b>
<b>6</b>	<b>Sudoku Solver</b>	<b>10</b>
6.1	Structure et composition . . . . .	10
6.2	Fonctionnement . . . . .	11
<b>7</b>	<b>Traitement d'image</b>	<b>13</b>

<b>8</b>	<b>Réseau de neurones</b>	<b>14</b>
<b>9</b>	<b>UI</b>	<b>15</b>
<b>V</b>	<b>Conclusion</b>	<b>16</b>

## Première partie

# Introduction

Ce projet a pour but de résoudre une grille de sudoku à partir d'une image. Pour le réaliser, nous devons faire un algorithme de résolution de sudoku, une reconnaissance d'image (binarisation , détection de grille et un découpage de l'image) et enfin un réseau de neurones (détection de chiffre dans une image).

Pour cela nous allons vous montrer le déroulement de ce projet à travers ces membres, son état actuel ainsi que le fonctionnement des différents aspects techniques de ce projet.

## Deuxième partie

# Membres de l'équipe

## 1 Login

Équipe	E-mails
Djallil Baizid	djallil.baizid@epita.fr
Dorian Wolff	dorian.wolff@epita.fr
Simon Arbiser	simon.arbiser@epita.fr
Rémi Brenaut	remi.brenaut@epita.fr

## 2 Présentation

### 2.1 Djallil

*Ladies and Gentlemen!* Mon nom est **Djallil BAIZID**, et vous vous apprêtez à assister à l'aube d'une nouvelle ère! Mon but pour ce super projet de la mort qui tue n'est ni la fortune, ni la gloire et encore moins le pouvoir, mais plutôt la satisfaction de mettre au monde ce qui me passe par la tête. Je vais faire de mon mieux pour vous proposer une expérience rocambolesque tout en acquérant des nouvelles compétences! El Psy Kongroo...

## 2.2 Dorian

Enchanté, je suis Dorian Wolff. Avec ce projet je souhaite approfondir mes connaissances en C en commençant par l'UI ainsi que la détection de grille. A nous quatre, je suis convaincu que nous pourrions apporter un projet qui s'apparente à un produit qualitatif et fonctionnel.

## 2.3 Simon

Je m'appelle Simon Arbiser, mon but dans ce projet est de faire un réseau de neurone qui fonctionne en C. Ayant déjà touché au réseau de neurones par le passé, je pense être à même de réussir à en coder un uniquement en C.

## 2.4 Rémi

Je trouve ce projet très intéressant, car il me permettra d'améliorer grandement mes connaissances en C. J'ai hâte d'en apprendre plus sur la création de réseaux de neurones et sur le traitement d'image. Ce projet est aussi l'occasion de rencontrer de nouvelles personnes.

## Troisième partie

# Récapitulatif du cahier des charges

### 3 Le Projet

L'objectif de ce projet est de réaliser un logiciel de type OCR (Optical Character Recognition) qui résout une grille de sudoku. L'application prendra donc en entrée une image représentant une grille de sudoku et affichera en sortie la grille résolue.

Dans sa version définitive, l'application devra proposer une interface graphique permettant de charger une image dans un format standard, de la visualiser, de corriger certains de ses défauts, et enfin d'afficher la grille complètement remplie et résolue. La grille résolue devra également pouvoir être sauvegardée.

L'application devra aussi posséder un aspect apprentissage, qui pourra être séparé de la partie principale, et qui permettra d'entraîner le réseau de neurones, puis de sauvegarder et de recharger le résultat de cet apprentissage.

Le traitement effectué par l'application sera approximativement le suivant :

- Chargement d'une image ;
- Suppression des couleurs (niveau de gris, puis noir et blanc) ;
- Prétraitement ;
- Détection de la grille ;
- Détection des cases de la grille ;
- Récupération des chiffres présents dans les cases ;
- Reconnaissance de caractères (ici les chiffres) ;
- Reconstruction de la grille ;
- Résolution de la grille ;
- Affichage de la grille résolue ;
- Sauvegarde de la grille résolue.

## 4 Répartition des tâches

	Solver	Réseau de Neurones	Détection d'Images
Djallil			1
Dorian			1
Simon		1	
Rémi	1		

*Tableau de répartition des tâches*



## 5 Calendrier et avancement des tâches

Pour illustrer l'accomplissement de nos objectifs pour la première soutenance, voici un comparatif d'avancement des tâches entre les estimations de notre cahier des charges et ce que nous avons réalisé.

Comparatif d'avancement par tâches :

Tâches\Deadlines	Objectifs	Avancée
Sudoku solver	100%	100%
Traitement de l'image	80%	100%
Réseau de neurones	50%	95%
Interface utilisateur	50%	80%

## Quatrième partie

# Projet

## 6 Sudoku Solver

La première étape pour créer une IA qui résout des sudokus est le solveur de sudoku. L'objectif est de concevoir un programme efficace, capable de résoudre des grilles de Sudoku à partir d'un fichier d'entrée spécifique, écrit en langage C, qui prend en compte les contraintes du format de fichier imposé.

### 6.1 Structure et composition

Ce programme en C, permet de résoudre un sudoku de façon récursive. Il est composé de deux fonctions : *isValid* qui vérifie si la position du chiffre dans la grille est valide, et *solve* qui remplit la grille. Pour faciliter le parcours de la grille de sudoku, on crée deux structures.

La structure *Cell* représente une case de la grille de sudoku et possède deux attributs : *value* qui renvoie la valeur du chiffre présent dans la case, et *fixed* qui marque les chiffres déjà présents dans la grille d'origine. La structure *SudokuGrid* est un double tableau de *Cell*.

Ainsi, on peut facilement accéder à n'importe quelle case de la grille de sudoku, savoir quel chiffre y est présent et si ce chiffre était déjà présent dans la grille d'origine.

## 6.2 Fonctionnement

Le programme prend en entrée le nom d'un fichier représentant une grille de Sudoku au format spécifié. Il lit ce fichier et génère une structure de données en mémoire pour représenter la grille.

Avant de commencer la résolution, le programme vérifie la validité de la grille selon les règles du Sudoku. Il s'assure qu'aucun chiffre n'est répété dans les lignes, colonnes ou blocs.

L'algorithme de résolution repose sur une approche de backtracking. Le solver tente différentes combinaisons de chiffres pour chaque case vide, vérifiant la validité à chaque étape, jusqu'à trouver une solution ou déterminer qu'aucune solution n'est possible.

Une fois la grille remplie, le solver génère un fichier de sortie portant le même nom que le fichier d'entrée, mais avec l'ajout de l'extension ".result". Ce fichier contient la grille résolue au même format que le fichier d'origine.

Le solver constitue une composante essentielle du projet Sudoku Solver, offrant une solution robuste et efficace pour la résolution de grilles de Sudoku conformes au format spécifié.

```

Grille initiale :
. . . | . . 4 | 5 8 .
. . . | 7 2 1 | . . 3
4 . 3 | . . . | . . .
-----+-----+-----
2 1 . | . 6 7 | . . 4
. 7 . | . . . | 2 . .
6 3 . | . 4 9 | . . 1
-----+-----+-----
3 . 6 | . . . | . . .
. . . | 1 5 8 | . . 6
. . . | . . 6 | 9 5 .

Grille résolue :
1 2 7 | 6 3 4 | 5 8 9
5 8 9 | 7 2 1 | 6 4 3
4 6 3 | 9 8 5 | 1 2 7
-----+-----+-----
2 1 8 | 5 6 7 | 3 9 4
9 7 4 | 8 1 3 | 2 6 5
6 3 5 | 2 4 9 | 8 7 1
-----+-----+-----
3 5 6 | 4 9 2 | 7 1 8
7 9 2 | 1 5 8 | 4 3 6
8 4 1 | 3 7 6 | 9 5 2

Résultat écrit dans grid_00.result

```

FIGURE 1 – Fonctionnement pas à pas du solver

## 7 Traitement d'image

Le traitement d'image consiste en la modification et l'analyse d'images pour en extraire des informations significatives ou les améliorer. La conversion en niveaux de gris (grayscale) est une technique qui transforme une image couleur en une image en nuances de gris, où chaque pixel a une valeur représentant l'intensité lumineuse. Cela simplifie l'image tout en conservant les informations essentielles.

La détection de grille dans un projet de résolution de Sudoku à partir d'une image implique la capacité à identifier et isoler la grille du Sudoku dans l'image initiale. Cela peut nécessiter des techniques de traitement d'image pour détecter les lignes et les contours, et diviser l'image en cases de grille distinctes. Cette étape est cruciale pour extraire chaque cellule du Sudoku individuellement afin de reconnaître les chiffres qui y sont écrits et résoudre le puzzle.

Cette partie a donc pour but d'isoler et de simplifier de nouvelles images des cases présentes sur l'image du sudoku et de les envoyer au réseau de neurones qui s'occupera de les reconnaître.

## 8 Réseau de neurones

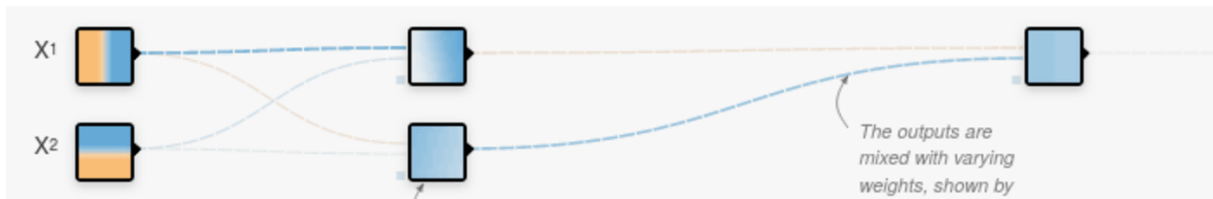


FIGURE 2 – Représentation du réseau de neurones pour la fonction XOR a l'aide de du site Tensorflow

Afin de créer un réseau de neurones en C, nous avons fait le choix de recoder une implémentation de matrice.

Le réseau pour la fonction xor est composé de 3 neurones avec comme fonction d'activation sigmoïde, l'entropie croisé binaire en fonction de perte (Binary Cross Entropy).

Il n'est pas nécessaire d'avoir un biais. Bien que dans la fatalité le fait d'avoir reprogrammé une implémentation matrice et d'une vraie descente de gradient pour xor, cela sert déjà a l'implémentassions du réseau pour la soutenance final. À la différence qu'il y aura  $28 \times 28 \times 10$  neurones, un Re LU comme fonction d'activation pour la première couche, un softmax pour la deuxième couche, une descente de gradient stochastique en optimiser et une entropie croisée comme fonction de perte.

## 9 UI

L'interface utilisateur (UI) développée en C avec GTK et SDL2 offre une expérience interactive aux utilisateurs. Cette interface comporte plusieurs boutons, chacun attribué à une fonctionnalité spécifique. L'un des boutons permet d'ouvrir un explorateur de fichiers, permettant à l'utilisateur de sélectionner un fichier à traiter. Un autre bouton est dédié au prétraitement de l'image, offrant des fonctionnalités pour manipuler et préparer l'image sélectionnée. Les autres boutons sont pour l'instant configurés comme des remplisseurs, prêts à être associés à des fonctionnalités futures, et ils servent actuellement de réservation pour des améliorations ou des fonctionnalités à venir.

## Cinquième partie

# Conclusion

Ce projet d'application de résolution d'images de Sudoku représente pour nous une avancée en tant que projet de groupe, en explorant des domaines qui nous étaient jusqu'à présent assez flous. En combinant des techniques de détection de grille, de conversion en niveaux de gris et d'analyse de motifs, cette application offre une solution pour résoudre des grilles de Sudoku à partir d'images. En permettant la capture et la transformation d'une image contenant un Sudoku, l'application démontre la capacité à extraire des informations visuelles complexes pour résoudre des casse-tête. Cette solution apporte une nouvelle perspective quant à l'automatisation de la résolution de Sudoku et nous ouvre la voie pour de futur projets dans le domaine du traitement d'image appliqué aux jeux et aux énigmes.