Sudoku Solver

BREGNAC Pierre - CASSIER Logan - LAHBOUB Lina - LARDON Lelio

Vue d'ensemble

Le but de ce projet est de créer un programme capable de résoudre une grille de sudoku incomplète depuis une image stockée sur l'ordinateur. Ce projet a été codé en Matlab avec une partie en Python.

Objectifs

1. Lire et reconnaître les chiffres déjà présents dans la grille.
2. Résoudre le sudoku et remplir la grille.

Grandes étapes

I. Implémentation du deep learning pour la reconnaissance des chiffres

L'image mise en entrée de l'algorithme est d'abord pré-traitée. Elle est binarisée et les petits artéfacts sont supprimés en effectuant une ouverture (fonction bwareaopen de Matlab) afin de ne pas porter l'algorithme à confusion plus tard.

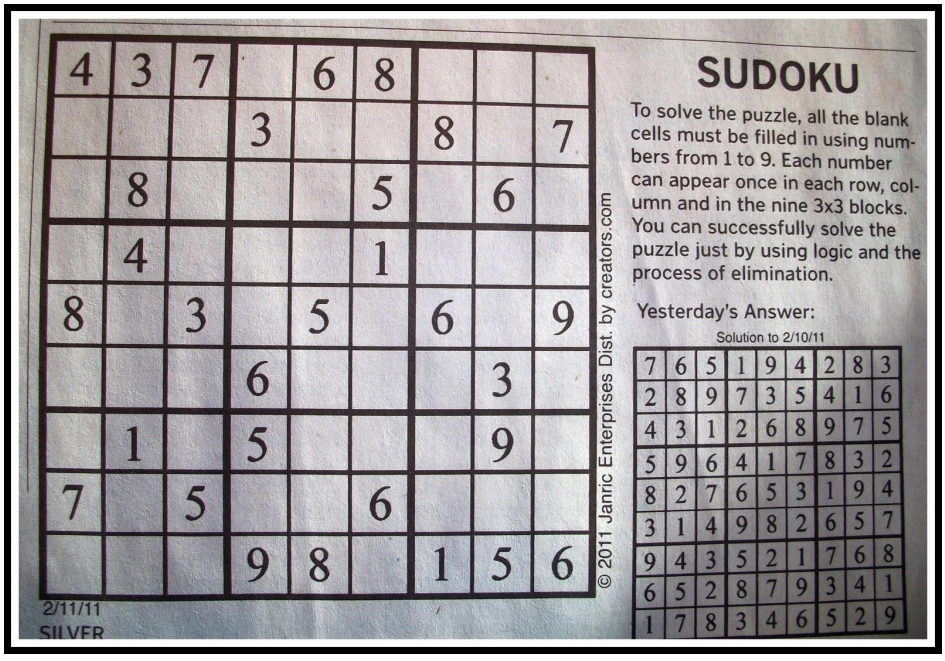

Fig 1 : Image en entrée

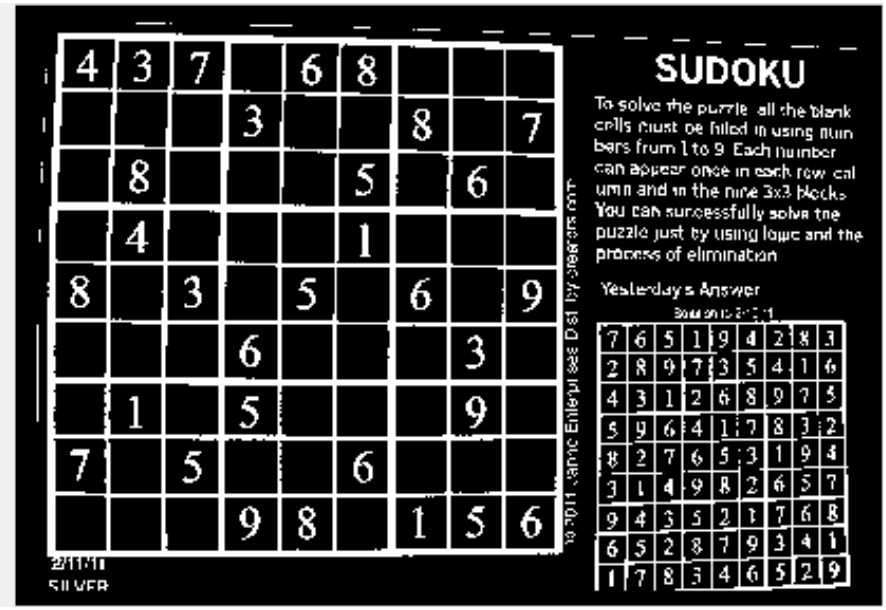

Fig 2 : Image pré-traitée

Pour reconnaître les chiffres présents sur l'image, il faut d'abord détecter les contours de la grille ainsi que ceux de chaque case contenant (ou non) un chiffre. On utilise alors la fonction regionprops de Matlab permettant de détecter une forme (ici une boîte rectangulaire).

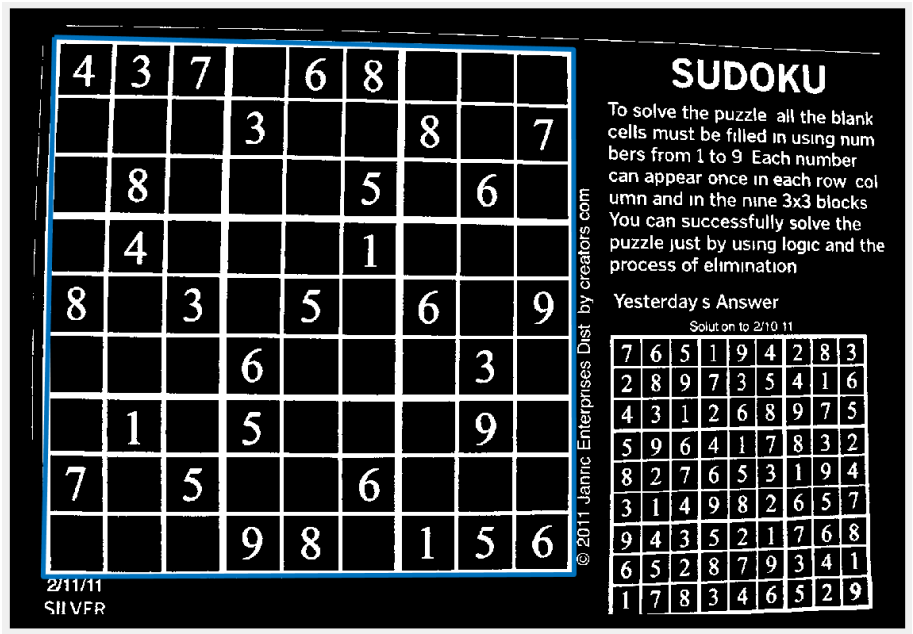

Fig. 3 : Cases et grille mises en évidence

Chaque petite case est enregistrée comme une image telle que celle ci-dessous et pré-traitée également afin de supprimer les bordures de la case et d'enlever le fond.

Fig. 4 : Imagette enregistrée puis pré-traitée

Puis, afin de reconnaître les chiffres de manière efficace, nous avons choisi d'utiliser un algorithme de deep learning implémenté en python. Le réseau de neurones de 6 couches est entraîné sur une base de données contenant des chiffres écrits en noir sur fond blanc. L'algorithme détecte le chiffre sur l'image et renvoie sa valeur.

Fig. 5 : Exemple d'image contenue dans la base de données

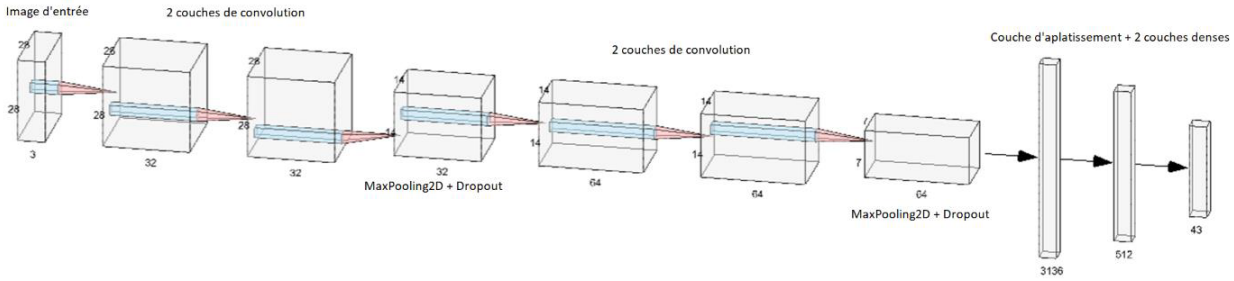

Fig. 6 : Architecture de notre réseau de neurone

Ce réseau est ainsi constitué de 4 couches de convolution et de 2 couches denses. Nous utilisons des fonctions d'activation RElu ainsi que des couches de MaxPooling et de DropOut.

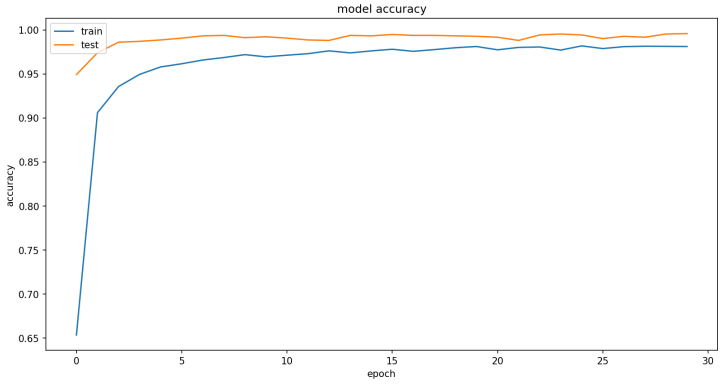

Fig. 7 : Justesse obtenue pour les bases d'entraînement et de test.

Cette architecture et cet entraînement nous permettent d'obtenir un score de 99,8% sur notre base de test. Cependant, les images utilisées par notre application peuvent être légèrement différentes. Les résultats en application réelle sont moins précis (environs 94%) mais nous avons tout de même une bonne reconnaissance.

La grille correspondant au sudoku est renvoyée sous forme de matrice de taille 9x9 contenant les chiffres détectés (0 en cas de case vide).

II. Résolution du sudoku et affichage du résultat

L'algorithme de résolution du sudoku est de type rétroactif avec un arbre de décision. On parcourt la grille ligne par ligne et on assigne un chiffre à chaque case vide. On continue comme ceci et dès qu'une erreur est trouvée (un chiffre qui apparaît deux fois), alors on remonte l'arbre jusqu'à ce que l'on n'aie plus d'erreur.

Finalement, une interface graphique a été créée, permettant à l'utilisateur de charger une image, de calculer les chiffres manquants et d'afficher le résultat.

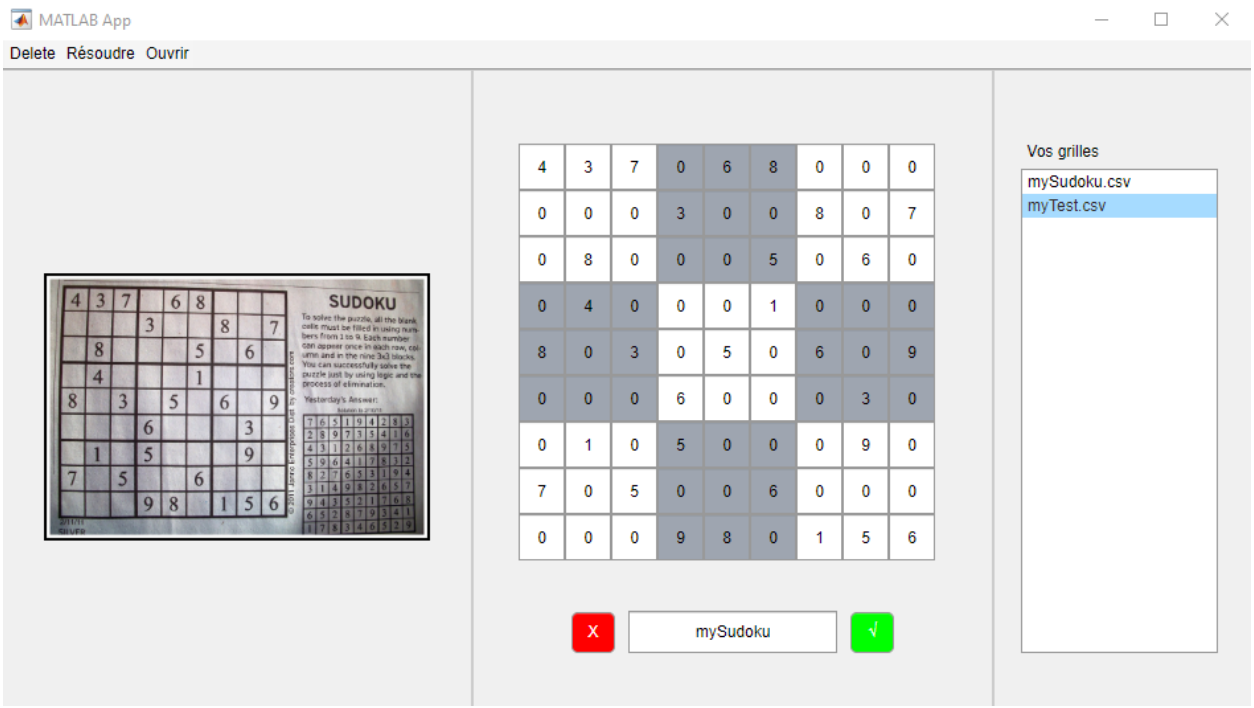

Fig 8 : Interface graphique

Conclusion et limites

Globalement, la reconnaissance de la grille fonctionne bien, même si celle-ci est un peu inclinée. L'algorithme de calcul des chiffres manquants ne fait pas d'erreurs non plus.

Cependant, plusieurs difficultés ont été rencontrées lors de ce projet. La plus grande concerne la reconnaissance des chiffres. L'algorithme de deep learning avait un très bon score lors de la phase d'apprentissage et de test, mais la prédiction sur les images que nous avons testées était souvent erronée. Nous avons donc dû tester plusieurs algorithmes entraînés sur des bases différentes et avons choisi de garder celui qui semblait renvoyer les meilleurs résultats.

Nous avions initialement prévu d'utiliser le programme avec une caméra avec laquelle l'utilisateur montrerait sa grille de sudoku à l'écran. La grille serait alors affichée en temps réel sur l'image. Nous n'avons malheureusement pas réussi à implémenter cette option.