



Accueil > Blog > News > Reconnaissance de chiffres manuscrits utilisant une intelligence artificielle sur une carte FPGA à faible coût

# Reconnaissance de chiffres manuscrits utilisant une intelligence artificielle sur une carte FPGA à faible coût

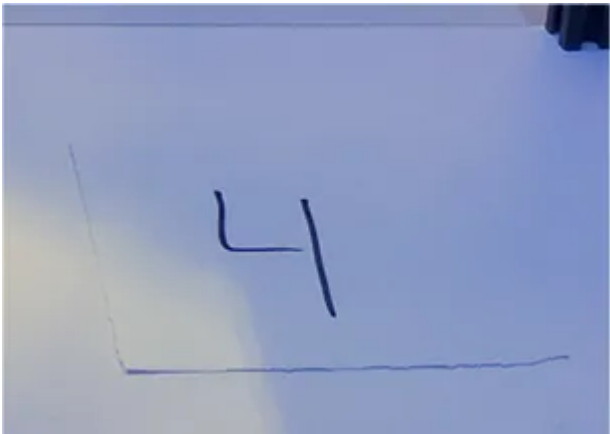
3240 views

Dans le monde de l'intelligence artificielle (IA), la reconnaissance des chiffres manuscrits prouve que vos neurones sont en état de fonctionner !

Cette application de l'intelligence artificielle est déjà assez ancienne - sa percée est intervenue en 1989 avec la réalisation d'une analyse syntaxique fiable et automatisée des codes postaux pour les services postaux.

Peu de temps après, il a été prouvé que les réseaux à feed-forward multicouches pouvaient implémenter toutes ces fonctions. Bientôt, les institutions financières ont adopté la technique d'analyse automatique des numéros de compte sur les bordereaux de versement pour les virements électroniques ou les chèques bancaires. Aujourd'hui, la reconnaissance des chiffres manuscrits à l'aide de différentes techniques d'IA est également réputée dans les milieux universitaires à des fins d'enseignement et d'apprentissage.

## La reconnaissance des chiffres manuscrits est complexe



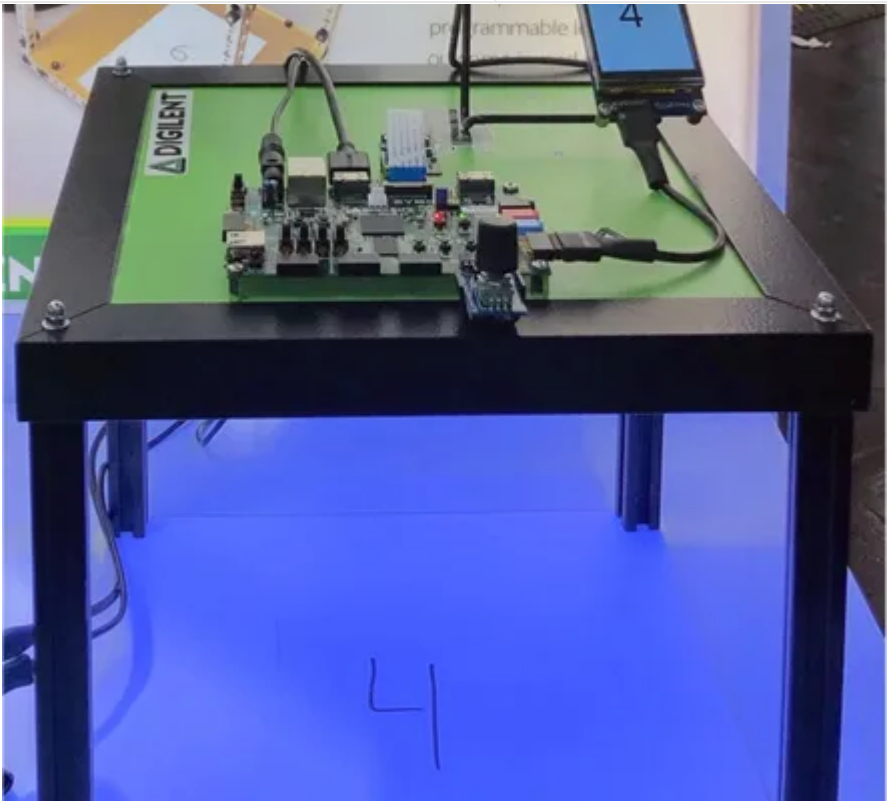
Bien que la présentation du problème lui-même n'ait pas changé au fil des ans, la puissance de calcul s'est considérablement accrue au cours des dernières décennies, ce qui permet désormais de transposer cette applications sur une puce FPGA (Field Programmable Gate Array) associée à une petite caméra numérique. Néanmoins, c'est toujours l'une des tâches les plus difficiles à entreprendre pour une intelligence artificielle. Les chiffres manuscrits diffèrent par leur taille, leur largeur et leur orientation, et l'écriture de chaque personne comporte des variations distinctes. Différentes cultures écrivent même les chiffres différemment. Par exemple, il existe des différences subtiles dans la façon dont les Américains et les Européens écrivent les chiffres 1, 4 ou 7.

Un autre domaine qui soulève des problèmes de reconnaissance des chiffres sont les similitudes entre certains chiffres: 3 et 8, 0 et 8, 5 et 6, 2 et 7, etc. Enfin, la qualité de l'image elle-même a une grande influence. S'il n'est pas correctement entraîné, l'IA classera incorrectement trop de chiffres pour que le produit final puisse être utilisé. Essayer de mettre en place un tel système de reconnaissance en utilisant l'approche de programmation classique va causer beaucoup de problèmes au programmeur avec beaucoup de détails et de cas particuliers.

## Implémentation de la reconnaissance

Mais comment fonctionne la reconnaissance ? C'est très simple de le regarder: une image du chiffre doit être capturée, traitée et analysée, et le résultat doit être présenté à l'utilisateur.

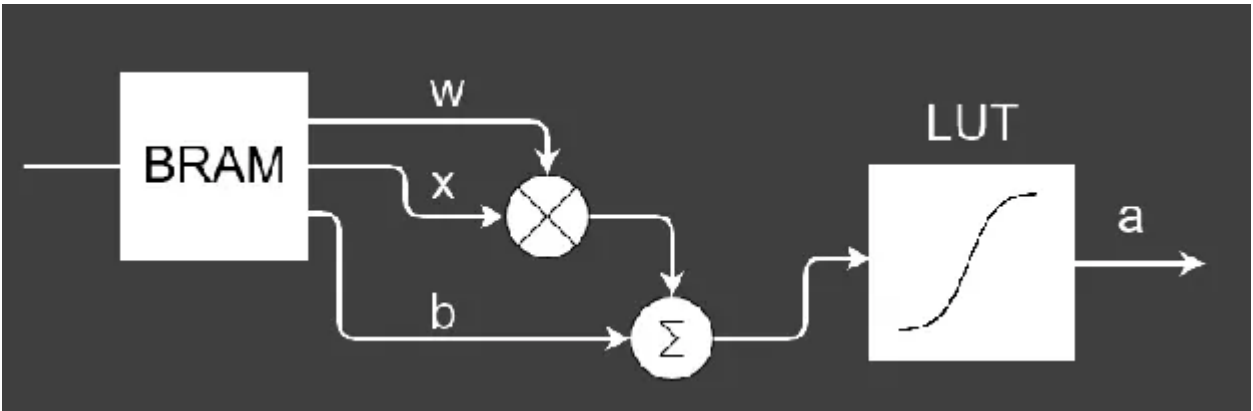
En effectuant une analyse plus approfondie, il devient vite évident que beaucoup plus de tâches doivent être exécutées et plus de problèmes devront être résolus. Le matériel pour un tel système n'est pas le véritable obstacle. La société Digilent, par exemple, a implémenté une reconnaissance de chiffres manuscrits faisant appel à l'intelligence artificielle comme preuve de concept en utilisant leur module de caméra couleur à focale fixe [Pcam 5C](#) de 5 mégapixels pour la capture d'images, ainsi que leur [carte de développement FPGA Zybo Z7 Xilinx Zynq-7000 ARM](#) et leur afficheur tactile multipoints [Pmod MTDS](#) (lequel sert d'interface utilisateur et de dispositif permettant d'afficher le résultat du processus de reconnaissance. A ce titre Digilent propose le pack [Embedded Vision](#) regroupant la carte FPGA et la caméra avec un prix très attractif).



Exemple de système de reconnaissance de chiffres manuscrits à faible coût utilisant différentes cartes de Digilent

Une fois que le module caméra Pcam 5C a capturé une image d'un chiffre avec un résolution de 1000 x 1000 pixels (ce qui est excessif pour cette application), il la transmettra à la platine Zybo Z7 via l'interface MIPI CSI-2 (interface série de la caméra MIPI). 2) afin d'y être pré-traité sur le FPGA. Ce dernier réduira l'image à 500 x 500 pixels, qu'il convertira en niveaux de gris et finalement numérisera de façon binaire en noir et blanc.

Un traitement ultérieur a lieu sur le processeur [ARM-Cortex-A9](#) double cœur 667 MHz, où une recherche dans un cadre de sélection est effectuée (la recherche de la plus petite image contenant le chiffre). Le chiffre est alors centré dans un cadre de sélection et le cadre de sélection est dimensionné à une taille d'image de 28 x 28 pixels (une autre possibilité aurait été de le centrer dans une fenêtre plus grande). Cette taille a été choisie, car l'ensemble d'entraînement utilisé pour le réseau de neurones utilise cette résolution.



Neurones dans le FPGA

À partir de cette image, chaque pixel est utilisé comme entrée dans l'un des 784 (28 x 28) neurones du réseau de neurones artificiels à alimentation directe (ANN) sur le FPGA. Chaque neurone utilise une représentation Q4.11 à virgule fixe et consiste en un multiplicateur multipliant par la largeur en pixels le poids dérivé de la formation et par un additionneur ajoutant le biais (également créé par les données d'entraînement) au résultat de la multiplication (voir l'image ci-dessus).

Enfin, une fonction sigmoïde est utilisée pour créer la sortie d'activation du neurone. Les équations 1 et 2 donnent les détails mathématiques:

$a = \sigma(w \cdot x + b)$  (équation 1)

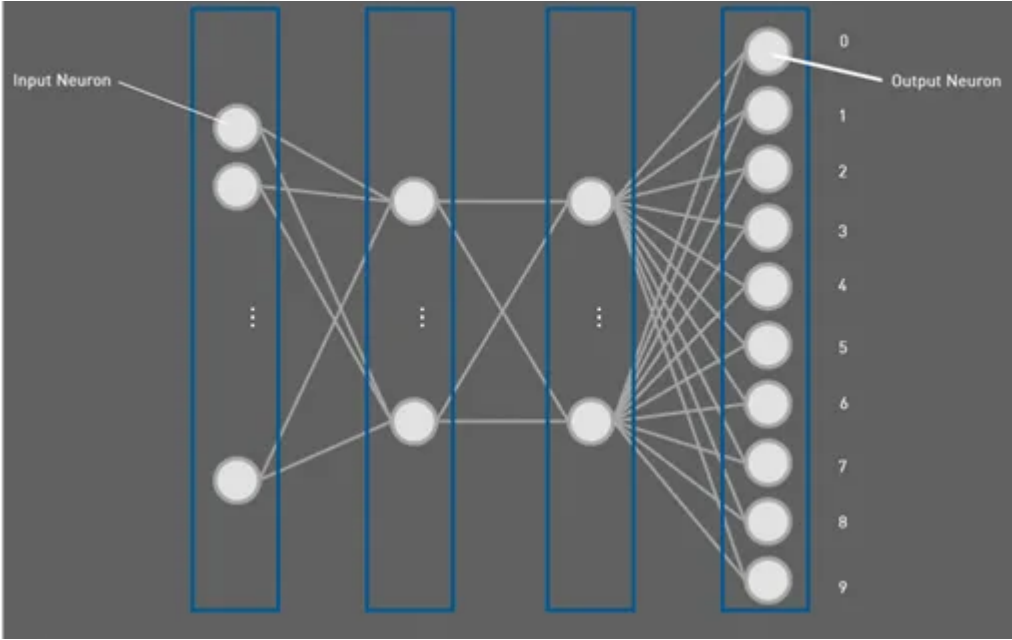
$\sigma(z) = 1 / (1 + e^{(-z)})$  (équation 2)

où w est le poids, b le biais et la fonction sigmoïde.

La sortie d'activation de la couche d'entrée est ensuite utilisée comme entrée pour les 16 neurones de la première couche cachée, le terme mystérieusement appelé «couche cachée» signifiant que les neurones appartenant à cette couche ne sont ni des neurones d'entrée ni de sortie. La sortie de la couche cachée 1 est ensuite utilisée comme entrée pour tous les neurones de la deuxième couche cachée, également composée de 16 neurones. Avec deux couches cachées, le système est techniquement qualifié de réseau neuronal profond (apprentissage approfondi), mais il ne peut être comparé à d'autres systèmes de plusieurs milliers de couches.

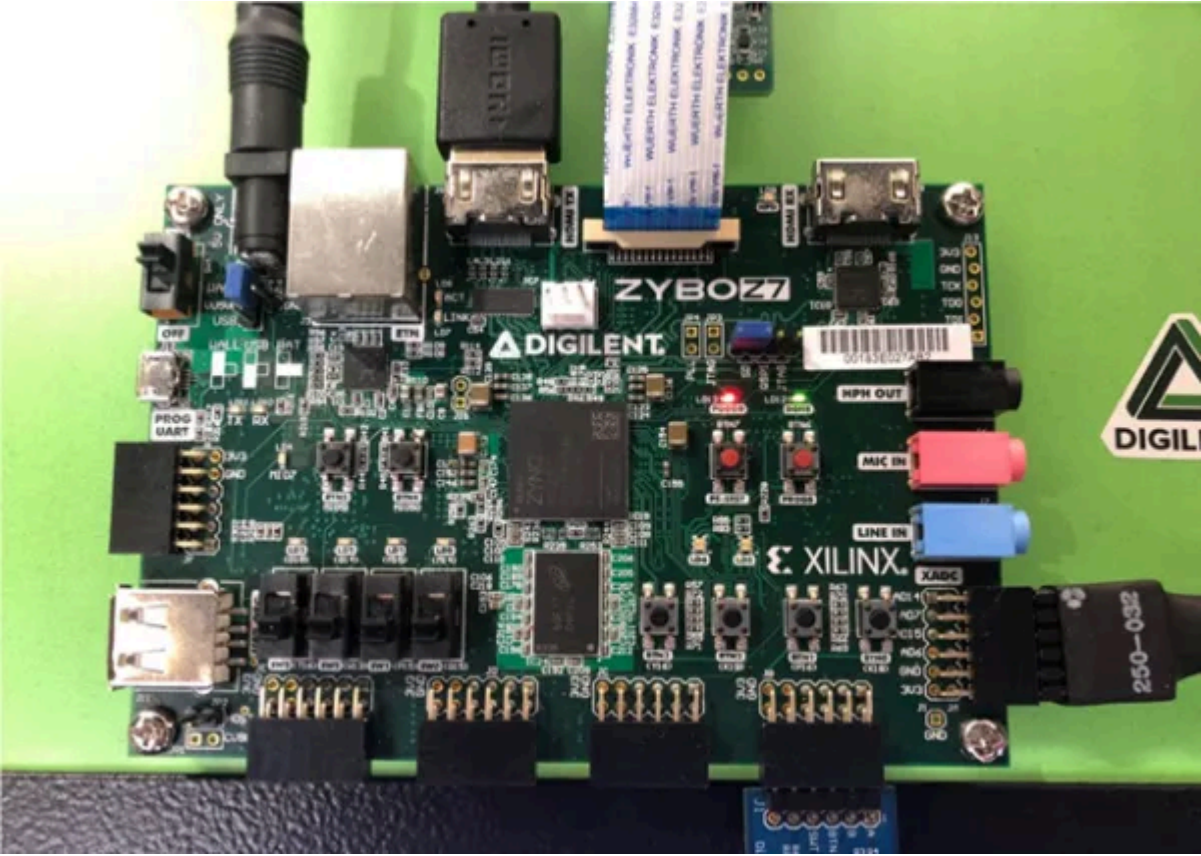
Chaque neurone de sortie correspond à un chiffre

La couche de sortie comprend finalement 10 neurones, un pour chaque chiffre compris dans la plage [0... 9]. Chaque neurone de sortie a une valeur d'activation et le neurone ayant la valeur d'activation la plus élevée est considéré comme la supposition correcte. Si, par exemple, le neurone de sortie représentant le chiffre 4 a la valeur la plus élevée, nous considérons le chiffre 4 comme la meilleure estimation de la reconnaissance.



Structure du réseau neuronal artificiel

Comme les sorties du réseau de neurones ne sont pas des probabilités en soi, le degré de certitude du résultat ne peut pas être précisé. Mais cela pourrait être résolu en ajoutant une couche Softmax au réseau, ce qui transforme les nombres en probabilités. Le résultat final de la reconnaissance est ensuite affiché sur le système d'affichage Pmod Multi-Touch de Digilent.



C'est tout ce qu'il faut pour exécuter une reconnaissance manuscrite des chiffres optimisée par l'intelligence artificielle

La formation est la plus importante

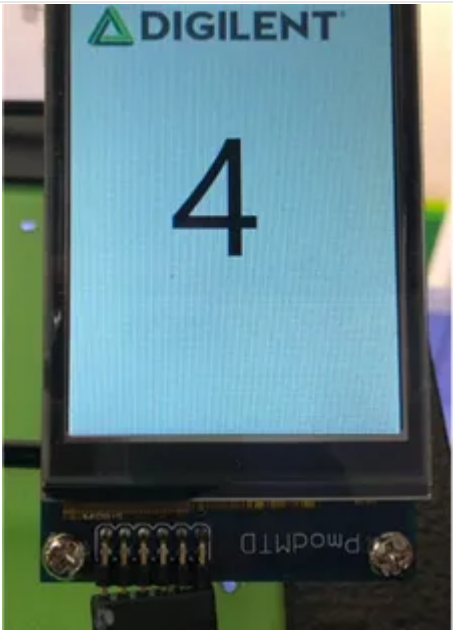
Comme avec tous les réseaux de neurones artificiels, le système doit être formé avant de pouvoir être utilisé. Dans le cas de notre exemple, le théorème de l'universalité a été appliqué pour calculer les poids et les biais d'images dans la base de données largement utilisée MNIST (NIST modifiée), qui comprend 60 000 images de formation et 10 000 images de test, les deux ensembles comportant des images étiquetées en écriture manuscrite (chiffres compris entre zéro et dix). La base de données s'appelle MNIST, car il s'agit d'un sous-ensemble des ensembles de données collectés par le NIST, l'Institut national des normes et de la technologie des États-Unis. La formation a été supervisée et effectuée sur un PC en utilisant Python. Chaque ensemble d'apprentissage est une paire d'entrées et le résultat souhaité et l'algorithme d'apprentissage ont été extraits du manuel à code source ouvert de Michael A. Nielsen [1].

Qu'est-ce qui a été réalisé ?

Une fois formé, le taux de reconnaissance du réseau de neurones artificiels formés sur les images de test MNIST a atteint 95,2%. Des taux plus élevés peuvent être atteints si un réseau de neurones à convolution (CNN) est utilisé. Les meilleurs résultats académiques avec un taux d'erreur de 0,23% ont été obtenus avec une hiérarchie de 35 réseaux de neurones de convolution, ce qui aurait été au-delà de la portée de cette preuve de concept. Il est également apparu que le taux de reconnaissance en direct dépendait beaucoup de l'asymétrie numérique, étant donné que l'atténuation automatique n'était pas mise en œuvre, mais exécutée manuellement. Le chiffre le plus critique s'est avéré être le chiffre «9», rarement reconnu par le démonstrateur. Une fois activé, il était généralement reconnu comme «6», ce qui montre bien que le recyclage du système devrait aider.

Le pré-traitement a pris 1664 ms, s'il était effectué uniquement sur le processeur double cœur ARM-Cortex-A9 de la carte de développement Zybo Z7 de Digilent et à 999 ms s'il était exécuté sur le cœur ARM et le FPGA. Le temps d'exécution du réseau de neurones artificiels était de 73 ms sur l'ARM et de 57 ms si l'opération était effectuée sur l'ARM plus le FPGA.





Le résultat de la reconnaissance est affichée sur un module Pmod MTDS

Cette mise en œuvre manuelle d'une reconnaissance de chiffres manuscrits à l'aide d'un tableau FPGA à faible coût a démontré qu'il était possible de mettre en œuvre un tel réseau de neurones artificiels avec un apprentissage en profondeur sur un tel système. La carte de développement ARM / FPGA Xilinx Zynq-7000, ainsi que la caméra Pcam 5 et le module d'affichage Pmod MTDS sont proposés par Digilent et disponibles en parti à prix réduit dans le pack [Embedded Vision Bundle](#).

[1] [Michael A. Nielsen, «Réseaux de neurones et apprentissage en profondeur», Determination Press. 2015 \(http://neuralnetworksanddeeplearning.com/chap1.html\)](#)

Traduction réalisée à partir du [Blog de Digilent](#)

# Construisez une gamme de systèmes embarqués et d'applications

## ECONOMISEZ 20% SUR LES PLATEFORMES SOC ZYNQ 7020 DIGILENT'



VISION EMBARQUÉE





LINUX EMBARQUÉ



ZYNQ POUR PYTHON

SUPER OFFRE



**DIGILENT**  
A National Instruments Company

Accueil	
▶ Capteurs et Cartes programmables	▼
▶ Robotique Imprimantes 3D	▼
▶ Mesure et interfaces	▼
▶ Composants electroniques	▼
▶ Câbles Boitiers Connectique	▼
▶ alimentations piles et accus	▼
▶ Fer a souder et outillage pour électronique	▼
▶ Son loisirs kits livres	▼
▶ Alarmes, vidéo, contrôle d'accès et domotique	▼
▶ Nouveautes	
▶ Produits obsolètes	
▶ Offres spéciales	▼

https://www.lextronic.fr/actualites/news/reconnaissance-de-chiffres-manuscrits-utilisant-une-intelligence-artificielle-sur-une-carte-fpga-a-faible-cout

4/5