

# TRAITEMENT D'IMAGES

## avec le module Jetson Nano de Nvidia

### 1<sup>ère</sup> partie : matériel et logiciel

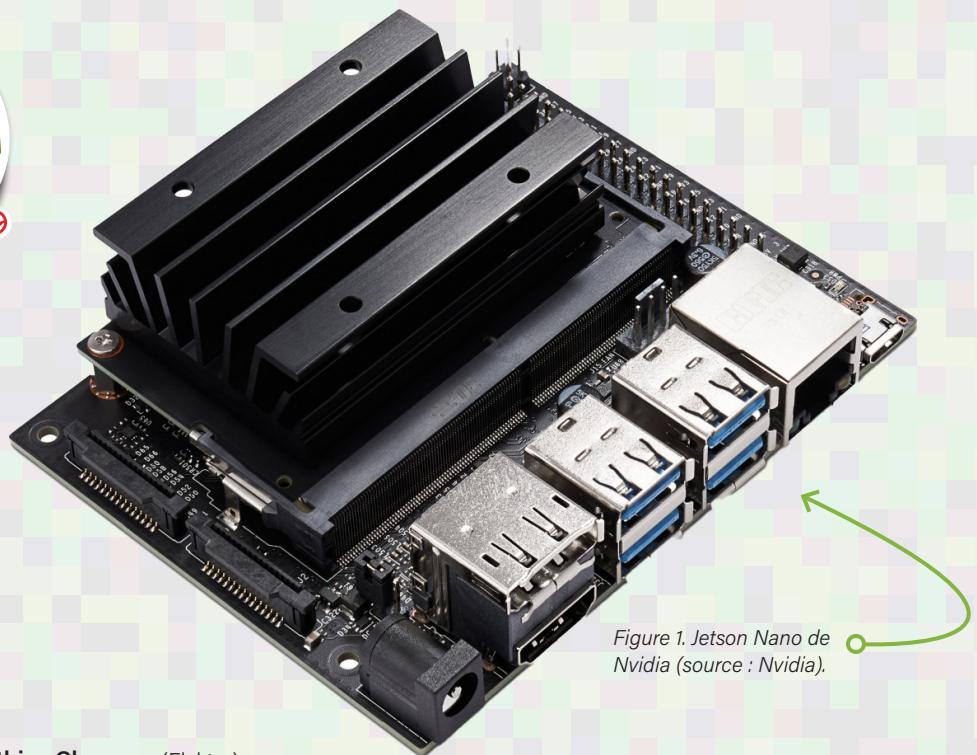


Figure 1. Jetson Nano de Nvidia (source : Nvidia).

Mathias Claussen (Elektor)

Des outils novateurs comme le module Jetson Nano de Nvidia permettent de se lancer facilement dans des projets d'intelligence artificielle (IA). Dans cette série d'articles, j'aborde en détail les applications possibles pour le traitement d'images ou la robotique. Commençons tout de suite par un aperçu du matériel et des logiciels...

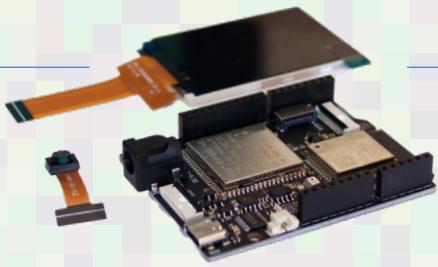


Figure 2. Carte Sipeed avec le microcontrôleur K210.



Figure 3. Carte FTHR MAX78000 avec accélérateur d'IA.



Figure 4. Carte Raspberry Pi 4.

Aucun développeur ne devrait ignorer le potentiel de l'intelligence artificielle (IA). Des outils comme le Maixduino de Sipeed et le MAX78000 de Maxim, déjà présentés dans le magazine Elektor, sont des options appropriées pour faire vos premiers pas dans l'univers de l'IA. Pour sa part, le module Jetson Nano de Nvidia (**fig. 1**) est un matériel beaucoup plus puissant de la classe des ordinateurs monocarte, doté d'un processeur ARM à quatre cœurs, de 4 Go de mémoire RAM et d'un processeur graphique assorti de 128 cœurs CUDA. Dans cette série d'articles, nous découvrons les applications possibles, notamment dans les domaines du traitement d'images ou de la robotique.

### L'IA au quotidien

L'IA s'invite de plus en plus dans notre vie quotidienne. Les réseaux neuronaux permettent par ex. de reconnaître des objets, d'améliorer des images et de convertir les paroles d'une conversation en texte. Et ce domaine passionnant va contribuer à façonner notre avenir. Pour vous aider à faire vos premiers pas dans l'univers enthousiasmant de l'IA, nous avons déjà présenté la carte Maixduino de Sipeed (**fig. 2**) [1] et la carte FTHR MAX78000 de Maxim (**fig. 3**) [2].

Les deux cartes ont un point commun : elles contiennent des microcontrôleurs et des accélérateurs pour réseaux neuronaux, mais avec des ressources limitées. Le Jetson Nano de Nvidia (**fig. 1**) [3] est un matériel beaucoup plus puissant de la catégorie des ordinateurs monocarte (SBC).

Doté d'un processeur ARM quadricœur, de 4 Go de mémoire RAM et d'un processeur graphique avec 128 cœurs CUDA, le module Jetson Nano est une solution de choix pour étudier l'IA et les réseaux neuronaux. Les applications possibles vont de la reconnaissance vocale à la classification d'objets dans des images vidéo capturées, en passant par le contrôle intégral de robots. De nombreux « accessoires » sont disponibles

pour tous les domaines – par ex. le kit JetBot AI v2.1 de SparkFun, doté de la plateforme Jetson Nano.

### Le processeur graphique fait toute la différence

Concernant le Jetson Nano de Nvidia, la comparaison avec le populaire Raspberry Pi 4B (**fig. 4**) va de soi. L'un et l'autre sont des ordinateurs monocarte, possèdent quatre cœurs de processeur et sont proposés avec 4 Go de mémoire RAM. Ils ont également comme points communs : la possibilité d'installer le système Ubuntu 20.04, la présence d'un connecteur d'extension. Le Raspberry Pi 4 possède un processeur graphique (GPU) Broadcom VideoCore6. Le Jetson Nano, pour sa part, dispose d'un GPU à base Maxwell avec 128 cœurs CUDA (GM20B). Ces cœurs sont à l'origine des performances supérieures du Jetson Nano. Les cœurs CUDA sont constitués d'unités arithmétiques programmables incorporées dans le processeur graphique, mais sans exclusivité de traitement puisqu'ils peuvent aussi exécuter des opérations arithmétiques

génériques en parallèle (par ex. celles nécessaires à l'entraînement et à la mise en œuvre d'un réseau neuronal).

### Entraînement du réseau neuronal avec le Jetson Nano

C'est ici que vous pouvez voir la différence avec un MAX78000. L'entraînement du réseau neuronal doit être effectué sur un PC Linux, le mieux étant de disposer d'une carte Nvidia compatible CUDA. À cet effet, vous pouvez utiliser PyTorch et d'autres outils logiciels. Le Jetson Nano, quant à lui, possède déjà tous les ingrédients pour entraîner lui-même un réseau neuronal. Certes, ce n'est pas aussi rapide que sur un PC actuel équipé d'une carte Geforce 1660 ou 1050Ti. Mais le Jetson Nano est à vous pour à peine 130 €. Une carte graphique coûte deux fois plus cher, et encore faut-il qu'elle soit disponible.

Nvidia fournit un grand nombre de tutoriels [4, 5] et, évidemment, des logiciels [6]. Ces contenus montrent, entre autres, comment entraîner les réseaux neuronaux pour reconnaître/traiter des images à l'aide d'une webcam sur le Jetson Nano de Nvidia.

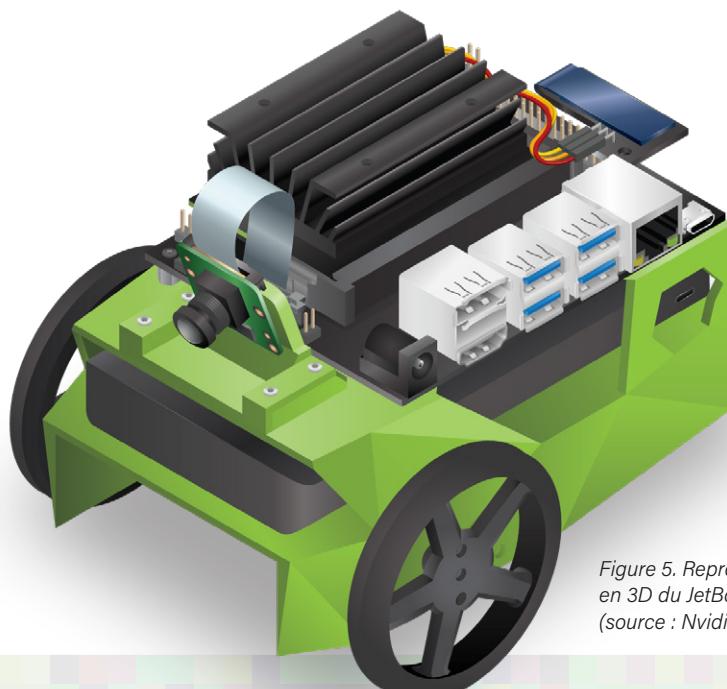


Figure 5. Représentation en 3D du JetBot de Nvidia (source : Nvidia).

## Jetson Nano et le robot mobile JetBot

Une fois familiarisé avec l'environnement, il y a de fortes chances que l'envie d'une application concrète vous saisisse. En raison de l'accent mis sur le traitement des images, un robot autonome s'impose. Le JetBot (**fig. 5**) [7] appartient à cette catégorie. Adapté aux makers, il comporte une liste de pièces et de modèles pour l'impression 3D. Un robot mobile offrant la reconnaissance d'objets et les interactions permet de voir directement en action l'intelligence des réseaux neuronaux automatiquement entraînés. Le webinaire « AI for Makers: Learn with JetBot » montre comment entraîner un robot de ce type pour lui permettre d'interagir avec son environnement [8].

## Développement dans un navigateur web

Pour développer le logiciel du JetBot, un simple navigateur web suffit. L'environnement de développement exécuté dans le navigateur est le JupyterLab (**fig. 6**) [9]. Il permet de créer, modifier et exécuter des scripts Python, mais aussi d'accéder à un grand nombre de bibliothèques tierces. Les pilotes de moteur ou les capteurs (fournis par ex. par Sparkfun ou Adafruit) peuvent être gérés directement en Python.

JupyterLab peut également lancer l'entraînement des réseaux neuronaux. Les nouvelles données d'entraînement produites pour les réseaux neuronaux servent à améliorer le comportement d'un robot grâce aux données issues du fonctionnement, mais aussi à ajouter de nouvelles fonctions. L'IDE est toujours accessible pour vous dans un navigateur.

Cependant, développer à partir de zéro la commande du robot et d'autres services est fastidieux et inutile. Vous pouvez faire appel pour cela au système d'exploitation ROS (*Robot Operation System*) [10] qui sert de base flexible pour tous les types de robots. Vous avez ainsi à votre disposition un noyau établi et une base logicielle bien documentée. Le système ROS, basé sur Linux, offre également des fonctions comme la création de cartes d'environnement. Une image appropriée du système d'exploitation avec ROS préinstallé est disponible pour le Jetson Nano de Nvidia et le robot JetBot.

## Kit tout-en-un de SparkFun

Vous pouvez télécharger des images, préparer le Jetson Nano, vous procurer des composants pour le JetBot auprès de différentes sources, et éventuellement les imprimer en 3D. Il est également possible de configurer les différentes parties du logiciel et de n'aborder l'IA qu'ensuite. Il est en effet plus facile d'utiliser un kit prêt à l'emploi pour disposer immédiatement d'un matériel défini et fonctionnel et pouvoir ensuite se plonger pleinement dans l'IA. Le kit JetBot AI v2.1 de SparkFun (**fig. 7**) comprend tout le matériel dont vous avez besoin pour expérimenter l'IA et la robotique. Basé sur le robot JetBot de Nvidia, il donne accès à un vaste écosystème de tutoriels et de logiciels, renforcé par la documentation pratique de SparkFun à propos du matériel [11].

Il n'est pas non plus nécessaire d'assembler les composants du logiciel : le kit contient une carte SD prête à l'emploi où figure déjà tout le nécessaire. L'achat d'un tel kit peut sembler coûteux, puisqu'il faut débourser environ 270 €. Cependant, pour ce prix, vous avez le Jetson Nano, les pièces mécaniques du robot, les moteurs, les pilotes de moteur, un afficheur OLED, un adaptateur Wi-Fi, une caméra, une carte SD et toutes les pièces nécessaires pour vous lancer (**fig. 8**).

## Et maintenant ?

Il y a beaucoup à apprendre et autant à découvrir. L'IA met en évidence des approches inédites et passionnantes pour doter un véhicule d'une commande autonome. Pour

la plupart des lecteurs, cependant, les notions de bibliothèque PyTorch et de système ROS doivent encore être replacées dans le bon contexte. PyTorch se charge de modifier et d'entraîner les processus d'IA. Comme tout logiciel, PyTorch doit être utilisé de manière appropriée et, de préférence, avec des fichiers de configuration, comme à l'accoutumée pour nombre de programmes de l'univers Linux. De même, le contrôle véhicule/robot est géré par le système ROS.

Le maniement de ces deux outils représente à lui seul un domaine suffisamment vaste pour écrire des livres entiers. Nous aborderons donc progressivement les différents composants et nous plongerons bientôt dans les détails !

(210318-04)

## Contributeurs

Conception et texte : **Mathias Claußen**

Rédaction : **Jens Nickel**

Mise en page : **Harmen Heida**

Traduction : **Pascal Godart**

## Des questions, des commentaires ?

Envoyez un courriel à l'auteur ([mathias.claussen@elektor.com](mailto:mathias.claussen@elektor.com)) ou contactez Elektor ([redaction@elektor.fr](mailto:redaction@elektor.fr)).

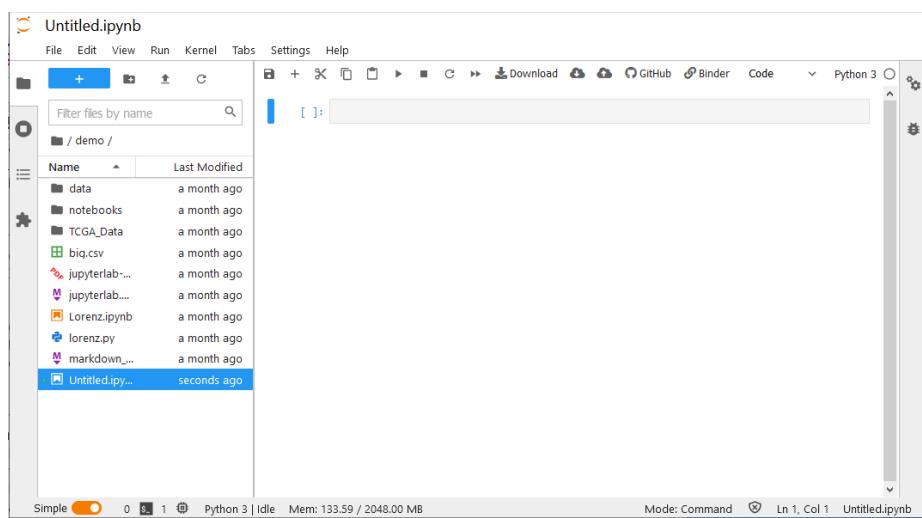


Figure 6. Jupyter Lab dans un navigateur.



Figure 7. Kit JetBot AI v2.1 de SparkFun  
(source : SparkFun).



Figure 8. Composants du kit JetBot AI v2.1 de SparkFun  
(source : SparkFun).



## PRODUITS

- **Module Jetson Nano de Nvidia**  
[www.elektor.fr/nvidia-jetson-nano-developer-kit](http://www.elektor.fr/nvidia-jetson-nano-developer-kit)
- **Kit AI JetBot v2.1 de SparkFun (avec kit de développement NVIDIA Jetson Nano inclus)**  
[www.elektor.fr/sparkfun-jetbot-ai-kit-v2-1-incl-nvidia-jetson-nano-developer-kit](http://www.elektor.fr/sparkfun-jetbot-ai-kit-v2-1-incl-nvidia-jetson-nano-developer-kit)
- **Kit AI JetBot v2.1 de SparkFun (sans kit de développement NVIDIA Jetson Nano inclus)**  
[www.elektor.fr/sparkfun-jetbot-ai-kit-v2-1-without-nvidia-jetson-nano-developer-kit](http://www.elektor.fr/sparkfun-jetbot-ai-kit-v2-1-without-nvidia-jetson-nano-developer-kit)

## LIENS

- [1] Walter Trojan, « Intelligence artificielle pour les ±nuls (1) », ElektorMagazine.fr : [www.elektormagazine.fr/articles/intelligence-artificielle-pour-debutants-1](http://www.elektormagazine.fr/articles/intelligence-artificielle-pour-debutants-1)
- [2] Luc Lemmens, « Applis d'IA avec la carte MAX78000 au format Feather : les fondamentaux du matériel », ElektorMagazine.fr : [www.elektormagazine.fr/news/applis-ia-avec-la-carte-max78000-au-format-feather](http://www.elektormagazine.fr/news/applis-ia-avec-la-carte-max78000-au-format-feather)
- [3] Nvidia Jetson Nano : <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>
- [4] Cours en ligne sur Jetson AI : [https://developer.nvidia.com/embedded/learn/jetson-ai-certification-programs#course\\_outline](https://developer.nvidia.com/embedded/learn/jetson-ai-certification-programs#course_outline)
- [5] Fondamentaux sur Jetson AI, Youtube : [www.youtube.com/watch?v=uvU8AXY1170&t=322s](http://www.youtube.com/watch?v=uvU8AXY1170&t=322s)
- [6] Centre de téléchargement Jetson : <https://developer.nvidia.com/embedded/downloads#?search=Jetson%20Nano>
- [7] Nvidia JetBot sur GitHub : <https://github.com/NVIDIA-AI-IOT/jetbot>
- [8] « AI for Makers - Learn with JetBot » : [www.youtube.com/watch?v=zOCSRzDUI-Y&t=2248s](http://www.youtube.com/watch?v=zOCSRzDUI-Y&t=2248s)
- [9] JupyterLab : <https://jupyter.org/>
- [10] ROS.org : [www.ros.org/](http://www.ros.org/)
- [11] Kit AI JetBot v2.1 de SparkFun, animé par Jetson Nano : [www.sparkfun.com/products/16417](http://www.sparkfun.com/products/16417)