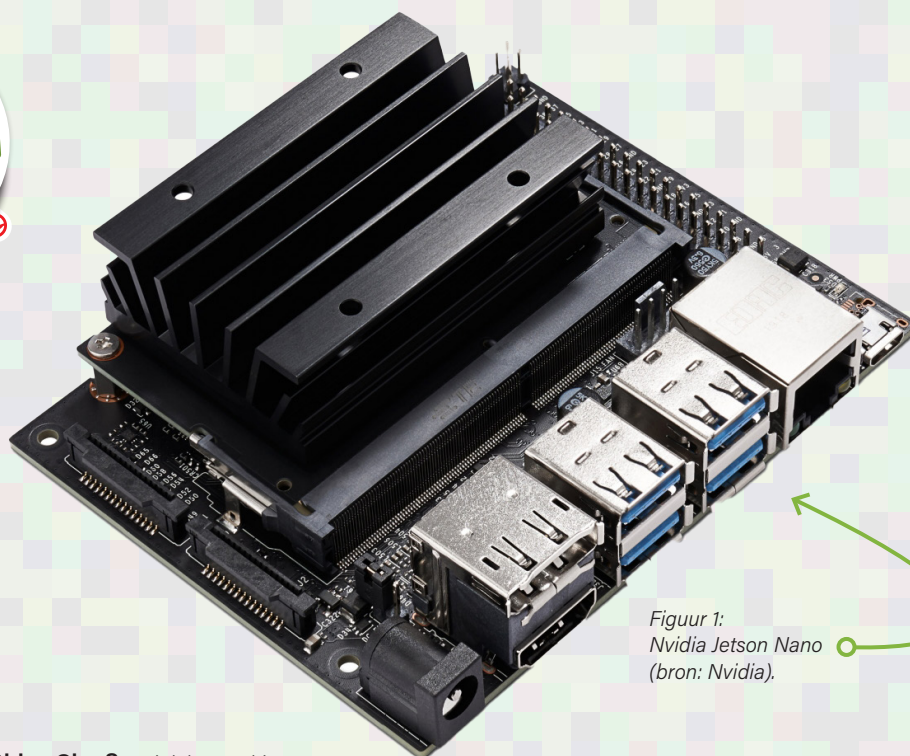


BEELD- VERWERKING

met de Nvidia Jetson Nano(deel 1)

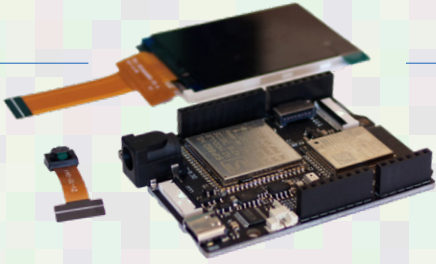
De hardware en software



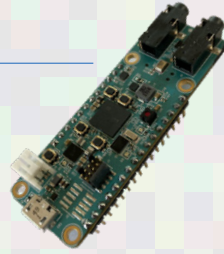
Figuur 1:
Nvidia Jetson Nano
(bron: Nvidia).

Mathias Claußen (Elektor Lab)

Innovatieve tools zoals de Nvidia Jetson Nano maken het gemakkelijk om aan de slag te gaan met kunstmatige intelligentie (AI) -projecten. In deze serie beschrijf ik mogelijke toepassingen voor beeldverwerking of robotica. Ik begin met een overzicht van hardware en software.



Figuur 2: Sipeed-bit met K210 MCU.



Figuur 3: MAX78000 FTBR-board met AI-versneller.



Figuur 4: Raspberry Pi 4-board.

Geen enkele ontwikkelaar mag het potentieel van kunstmatige intelligentie (AI) negeren. Tools zoals de Sipeed Maixduino en de Maxim MAX78000, die we al in Elektor hebben gepresenteerd, zijn geschikte opties om je eerste stappen in de wereld van AI te zetten. De Nvidia Jetson Nano (**Figuur 1**) biedt veel krachtige hardware op een single-board computer, het is uitgerust met een quad-core ARM-processor, 4 GB RAM en een GPU met 128 CUDA-kernen. In deze reeks artikelen laat ik je zien welke toepassingen mogelijk zijn op bijvoorbeeld het gebied van beeldverwerking of robotica.

AI in het dagelijks leven

AI vindt steeds meer zijn weg in ons dagelijks leven. Neurale netwerken maken het mogelijk om objecten te herkennen, afbeeldingen te verbeteren en de woorden van een gesprek om te zetten in tekst. Dit spannende onderwerp zal helpen onze toekomst vorm te geven. Om je te helpen de eerste stappen te zetten in de wereld van AI, hebben we al gekeken naar de Sipeed Maixduino (**Figuur 1**) [1] en het Maxim MAX78000 FTBR-board (**Figuur 2**) [2]. Beide boards hebben iets gemeen: de microcontrollers en accelerators voor neurale netwerken zijn ingebouwd, maar met beperkte middelen. De Nvidia Jetson Nano (**Figuur 3**) [3] is een veel krachtiger stuk hardware op het gebied van single board computers (SBC).

Uitgerust met een quad-core ARM-processor, 4 GB RAM en een GPU met 128 CUDA-kernen, is de Jetson Nano een goed platform voor het onderzoeken van AI en neurale netwerken. Toepassingen variëren van spraakherkenning en het classificeren van objecten in videobeelden tot volledige robotbesturing. Er zijn tal van "accessoires" beschikbaar voor diverse gebieden - bijvoorbeeld de SparkFun JetBot AI Kit v2.1 Powered by Jetson Nano.

De GPU maakt het verschil

Bij de Nvidia Jetson Nano ligt een vergelijking met de populaire Raspberry Pi 4B (**Figuur 4**)

voor de hand. Beide zijn SBC's, beide hebben vier processorkernen en beide zijn verkrijgbaar met 4 GB RAM. Een Ubuntu 20.04 kan ook op beide kaarten worden geïnstalleerd, en beide hebben een pin-header voor uitbreidingen. De Raspberry Pi 4 heeft een Broadcom VideoCore6 grafische verwerkingseenheid (GPU). De Jetson Nano heeft een Maxwell-gebaseerde GPU met 128 CUDA-cores (GM20B). Deze kernen geven de Jetson Nano zijn prestatie voordelen. De CUDA-kernen zijn programmeerbare reken eenheden binnen de GPU die niet alleen bedoeld zijn voor het verwerken van grafische gegevens, maar die ook rekenkundige bewerkingen parallel kunnen uitvoeren (dwz ook bewerkingen welke nodig zijn voor het trainen en toepassen van een neuraal netwerk).

Trainen met de Jetson Nano

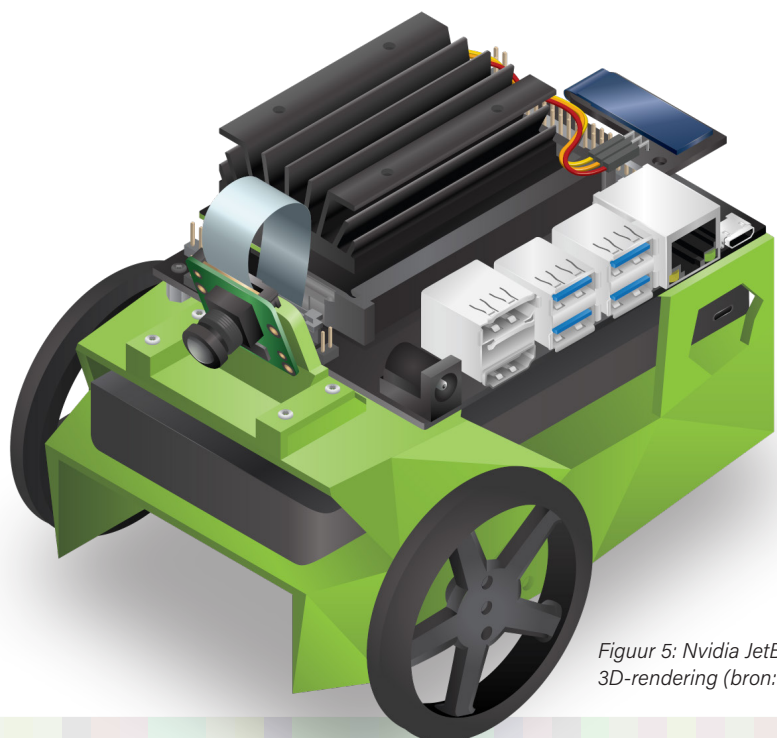
Hier zie je het verschil met een MAX78000. De training van het neurale netwerk moet worden gedaan op een Linux-pc, die

het beste uitgerust kan worden met een CUDA-compatibele Nvidia-kaart. Hiervoor worden PyTorch en andere softwaretools gebruikt. De Jetson Nano heeft daarentegen al alle ingrediënten om zelf een neuraal netwerk te trainen. Dit is zeker niet zo snel als op een alledaagse pc met een Geforce 1660 of Geforce 1050Ti, maar de Jetson Nano is te koop voor ongeveer € 130. Een grafische kaart, als deze al voorhanden is, kost twee keer zoveel.

Nvidia biedt een groot aantal tutorials [4], [5] en natuurlijk ook software [6]. Ze laten onder andere zien hoe neurale netwerken beeldherkenning en -verwerking kunnen aanleren op de Nvidia Jetson Nano met behulp van een webcam.

Jetson Nano en de bewegende JetBot

Als je eenmaal bekend bent met de Jetson-omgeving, kan de wens voor een praktische toepassing ontstaan. Door de focus op



Figuur 5: Nvidia JetBot als 3D-rendering (bron: Nvidia).

beeldverwerking helpt een zelfrijdende robot zichzelf. De JetBot (**Figuur 5**) [7] is zo'n robot. Het is hobbyist-vriendelijk, een onderdelenlijst en 3D-modellen zijn beschikbaar voor het zelf printen van onderdelen. Een bewegende robot met objectherkenning en interactie laat de intelligentie van zelfgetrainde neurale netwerken direct in de praktijk zien. Het webinar "AI voor makers: leer met JetBot" laat zien hoe je met zo'n robot leert omgaan met zijn omgeving [8].

Ontwikkeling via webbrowser

Een webbrowser is voldoende om de software van de JetBot te ontwikkelen. De ontwikkel-omgeving die in de browser wordt uitgevoerd, is JupyterLab (**Figuur 6**) [9]. Het maakt het schrijven, bewerken en uitvoeren van Python-scripts mogelijk. Dit geeft ook toegang tot een flink aantal bibliotheken van andere partijen. Motor drivers of sensoren (bijvoorbeeld van Sparkfun of Adafruit) kunnen rechtstreeks vanuit Python worden aangesproken.

JupyterLab kan ook worden gebruikt om neurale netwerktraining te starten. Het is mogelijk om nieuwe trainingsdata voor de neurale netwerken te genereren om het gedrag van een robot te verbeteren met data van de lopende applicatie en om nieuwe functies toe te voegen. Je hebt de IDE altijd bij je in een browser.

Het is echter tijdrovend om de besturing van de robot en andere services vanaf nul te ontwikkelen en ook onnodig. Het Robot Operation System (ROS) [10] kan worden gebruikt als flexibele basis voor alle soorten robots. Hiermee kunt je voortbouwen op een specifieke kern en een goed gedocumenteerde software-basis. ROS als besturings-systeem op basis van Linux maakt echter ook functies mogelijk zoals het maken van kaarten van de omgeving en andere acties. Een geschikte OS-image met vooraf geïnstalleerde ROS is beschikbaar voor de Nvidia Jetson en JetBot.

De alles-in-één kit van SparkFun

Afbeeldingen downloaden, de Jetson Nano voorbereiden, componenten voor de JetBot uit verschillende bronnen aanschaffen en

eventueel printen in 3D? Eerst de afzonderlijke software onderdelen configureren op de juiste manier en dan pas met AI beginnen? Het is gemakkelijker om een kant-en-klare kit te gebruiken om meteen te beschikken over gedefinieerde en werkende hardware, zodat je je volledig kunt verdiepen in AI. De SparkFun JetBot AI Kit v2.1 (**Figuur 7**) bevat alle hardware die je nodig hebt om aan de slag te gaan met AI en robotica. Het is gebaseerd op Nvidia's JetBot en geeft je toegang tot een groot ecosysteem van tutorials en software, met SparkFun's duidelijke hardware-documentatie [11].

Het is ook niet nodig om software-componenten samen te stellen. De kit bevat een kant-en-klare SD-kaart die alles bevat wat nodig is. De aanschaf van zo'n kit van rond de € 270 lijkt misschien duur, maar het bevat de Jetson Nano, de mechanische onderdelen van de robot, motoren, motorstuurprogramma's, een OLED, WiFi-adapter, camera, SD-kaart en alle andere onderdelen die nodig zijn om aan de slag te gaan (**Figuur 8**).

Wat is de volgende stap met de Jetson Nano?

Er is veel te leren en veel te ontdekken. AI toont veel nieuwe, spannende manieren om een voertuig uit te rusten met autonome besturing. Voor de meeste lezers moeten termen als PyTorch en ROS echter zeker nog in de juiste context worden geplaatst. PyTorch zorgt voor het aanpassen en aanleren van de

AI. Zoals elke software wil PyTorch op de juiste manier worden bediend en, zoals gewoonlijk met veel programma's uit de Linux-wereld, bij voorkeur met configuratiebestanden. De besturing van het voertuig of de robot wordt dan beheerd door de ROS.

Alleen al het hanteren van deze twee tools biedt voldoende materiaal om hele boeken te vullen. Daarom gaan we langzaam aan de afzonderlijke componenten bestuderen en dieper in de details duiken! ◀

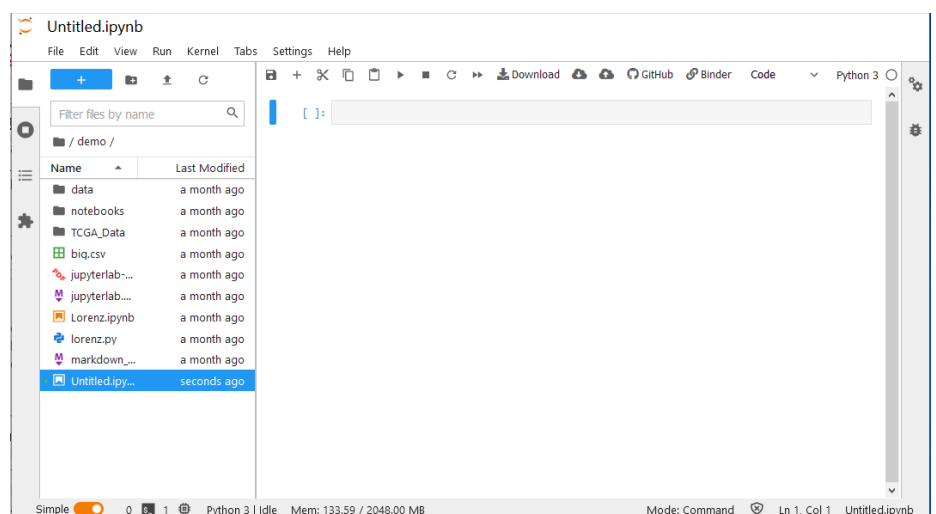
210318-03

Een bijdrage van:

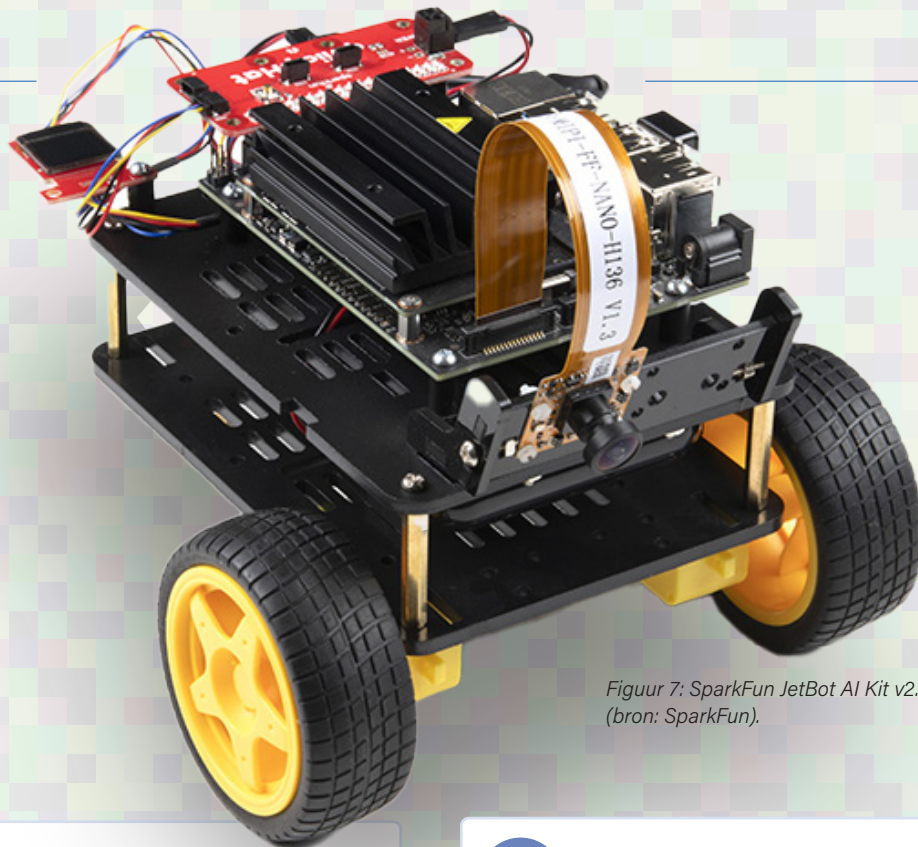
Tekst en foto's: Mathias Claußen
Redactie: Jens Nickel
Layout: Harmen Heida
Vertaling: Hans Adams

Vragen of opmerkingen?

Heb je technische vragen of opmerkingen over dit artikel? E-mail de auteur via mathias.claussen@elektor.com of neem contact op met Elektor via editor@elektor.com.



Figuur 6: Jupyter Lab in een browser.



Figuur 7: SparkFun JetBot AI Kit v2.1
(bron: SparkFun).



Figuur 8: Onderdelen in de SparkFun JetBot AI Kit v2.1 (bron: SparkFun).



GERELATEERDE PRODUCTEN

- > **NVIDIA Jetson Nano Developer Kit (SKU 19001)**
www.elektor.nl/nvidia-jetson-nano-developer-kit
- > **SparkFun JetBot AI Kit v2.1 (incl. NVIDIA Jetson Nano Developer Kit) (SKU 19576)**
www.elektor.nl/sparkfun-jetbot-ai-kit-v2-1-incl-nvidia-jetson-nano-developer-kit
- > **SparkFun JetBot AI Kit v2.1 (without NVIDIA Jetson Nano Developer Kit) (SKU 19685)**
www.elektor.nl/sparkfun-jetbot-ai-kit-v2-1-without-nvidia-jetson-nano-developer-kit

WEBLINKS

- [1] **Walter Trojan, Artificial Intelligence voor beginners, ElektorMagazine.nl:**
www.elektormagazine.nl/news/gratis-nieuw-artikel-ai-voor-beginners-1
- [2] **Luc Lemmens, AI met het MAX78000 Feather Board: Hardware Essentials, ElektorMagazine.nl:**
www.elektormagazine.nl/news/ai-met-het-max78000-feather-board-de-hardware
- [3] **Nvidia Jetson Nano:** <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>
- [4] **Jetson AI Fundamentals Course Outline:**
https://developer.nvidia.com/embedded/learn/jetson-ai-certification-programs#course_outline
- [5] **Jetson AI Fundamentals op Youtube:** www.youtube.com/watch?v=uvU8AXY1170&t=322s
- [6] **Jetson Download Center:** <https://developer.nvidia.com/embedded/downloads#?search=Jetson%20Nano>
- [7] **Nvidia JetBot op GitHub:** <https://github.com/NVIDIA-AI-IOT/jetbot>
- [8] **AI for Makers - Learn with JetBot:** www.youtube.com/watch?v=zOCSRzDUI-Y&t=2248s
- [9] **JupyterLab:** <https://jupyter.org/>
- [10] **ROS.org:** www.ros.org/
- [11] **SparkFun JetBot AI Kit v2.1 Powered by Jetson Nano:** www.sparkfun.com/products/16417