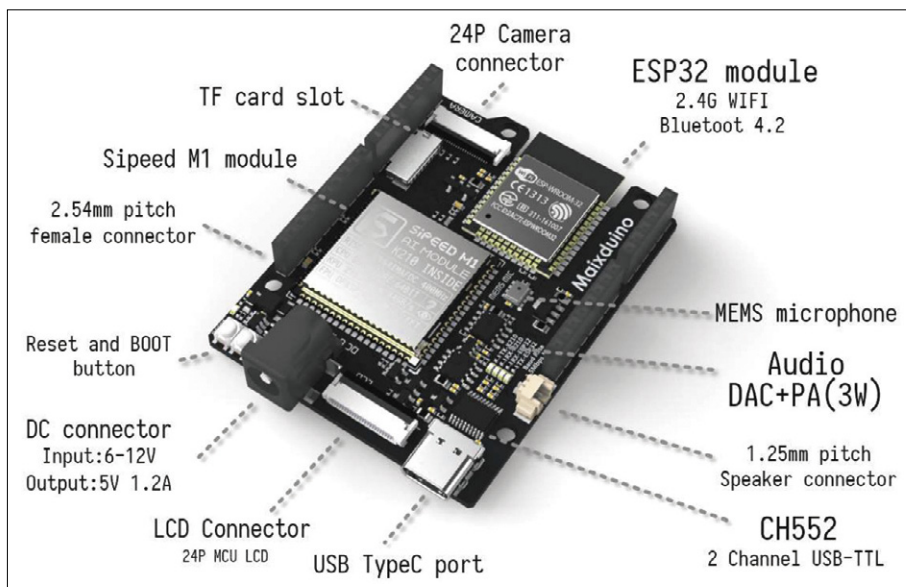


AI voor beginners (1)

objectherkenning met het Maixduino-board

Walter Trojan (Duitsland)

Geen enkel IT-onderwerp staat op dit moment zo in de belangstelling als kunstmatige intelligentie. Het Maixduino-board is een goede en goedkope kennismaking met de wereld van de spraak- en beeldherkenning en andere facetten van AI. Het is uitgerust met een camera en een klein LC-display en kost niet meer dan ongeveer € 30. Het is ook programmeerbaar via de bekende Arduino IDE. In het eerste deel van dit artikel introduceren we de rijke hardware van de Maixduino en enkele demoprogramma's, waaronder één dat duizend verschillende objecten kan herkennen.



Figuur 1. Uitrusting van de Maixduino.

Kunstmatige intelligentie is op veel gebieden in opmars en vindt zijn weg naar veel apparaten en toepassingen. Voor 2020 wordt voorspeld dat de wereldwijde verkoop van industriële applicaties op het gebied van AI \$ 4,8 miljard zal bedragen, en in 2025 zal dat naar verwachting meer dan \$ 30 miljard zijn. Wie al kennis heeft van dit vakgebied, heeft goede kansen op de arbeidsmarkt, omdat er dringend behoefte is aan AI-specialisten. Het kost wel wat moeite om vertrouwd te raken met dit onderwerp, maar het opent een heel nieuwe wereld van mogelijke toepassingen. En, last but not least, het

is ook heel leuk om er kennis mee te maken.

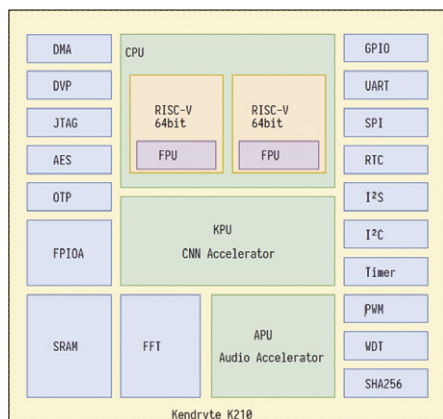
Onder kunstmatige intelligentie verstaan we onder andere het imiteren van intelligent gedrag. Een deelgebied daarvan is machinaal leren, waarbij een applicatie niet meer in alle details wordt uitgedacht door de ontwikkelaar, maar een universeel toepasbaar programma-raamwerk in de vorm van een neurale netwerk (NN) zichzelf de benodigde functies aanleert met behulp van een groot aantal voorbeelden. Een subcategorie hiervan is *Deep Learning*, waarbij complexe en geoptimaliseerde programmastructu-

ren worden gebruikt om de geleverde resultaten verder te verbeteren. Meer details over de hier genoemde technische termen vindt u in deze en de volgende delen van deze kleine serie artikelen.

AI is trouwens geen gloednieuw onderwerp, het bestaat al sinds de jaren vijftig van de vorige eeuw. De doorbraak kwam pas enkele jaren geleden, toen de rekenkracht van de processoren aanzienlijk werd vergroot. Grafische kaarten met honderden parallel werkende reken-eenheden en zeer gespecialiseerde AI-chips maken het nu mogelijk om AI-oplossingen te trainen en toe te passen. Dit maakte de weg vrij voor spraakherkenning, wat in veel persoonlijke assistenten wordt gebruikt, en voor beeldherkenning van allerlei soorten objecten. Van AI wordt ook verwacht dat het een belangrijke rol zal spelen bij het autonoom rijden. Op sommige gebieden is AI al superieur aan de mens, het verslaat de wereldkampioenen in schaken en go en kan in de gezondheidszorg tumoren betrouwbaarder detecteren dan ervaren medische specialisten. Maar er is geen reden voor ongerustheid: deze superioriteit geldt alleen bij getrainde speciale vaardigheden; ook in de toekomst zal het de mens zijn die de leiding heeft over het trainen van de computer.

Goedkope instap

Een goede en goedkope kennismaking met deze wereld is de Maixduino. Hij kost



Figuur 3. Blokschema van de Kendryte K210.

voorzien van geschikte pullup-weerstanden op de print.

Helaas geeft het datasheet van de K210 niet aan wat de maximale uitgangsstroom van de uitgangen is. Het zou voldoende moeten zijn om een LED aan te sturen met ongeveer 10 mA, maar

voor grotere stromen is een driver-IC aan te bevelen.

Het hart van de machine: de Kendryte K210

Het SoC (*System-on-a-Chip*) K210 van het Chinese bedrijf Kendryte is het hart van de Maixduino. Het wordt gefabriceerd in een energiebesparende technologie met een pitch van 28 nm en is sinds september 2018 op de markt. Voor het verwerken van de 'normale' werklast zijn twee 64bit-processoren beschikbaar, die worden geklokt op 400 MHz, waarbij overklokken tot 800 MHz mogelijk is. Hun binnenwerk is gebaseerd op de open RISC-V standaard, waardoor de fabrikant de licentiekosten voor het bedrijf ARM bespaart, wat iets bijdraagt aan de lage verkoopprijs. Beide processoren zijn uitgerust met een FPU (*Floating Point Unit*) ter ondersteuning van drijvende-komma-bewerkingen (enkele en dubbele precisie). **Figuur 3** toont de andere ingewanden van deze chip in detail.

Het bijzondere aan de K210 is de KPU (*Knowledge Processing Unit*) voor het afbeelden en berekenen van neurale netwerken. Deze heeft een voor dit prijssegment fenomenale rekenkracht van 0,46 TOPs, oftewel 460 miljard instructies per seconde. Dit zou kunnen worden verdubbeld door overklokken (waardoor bijvoorbeeld herkenning van 60 objecten per seconde mogelijk is). Deze hoge verwerkingssnelheid wordt bereikt door 64 rekenenheden die parallel werken aan een woordbreedte van 576 bits. Het energieverbruik is zeer laag in vergelijking met andere AI-systemen: 0,3 W. Voor de Jetson Nano adviseert Nvidia bijvoorbeeld een voeding van $5 \text{ V} / 4 \text{ A} = 20 \text{ W}$ voor een vergelijkbaar rekenvermogen van 0,4 TOPs.

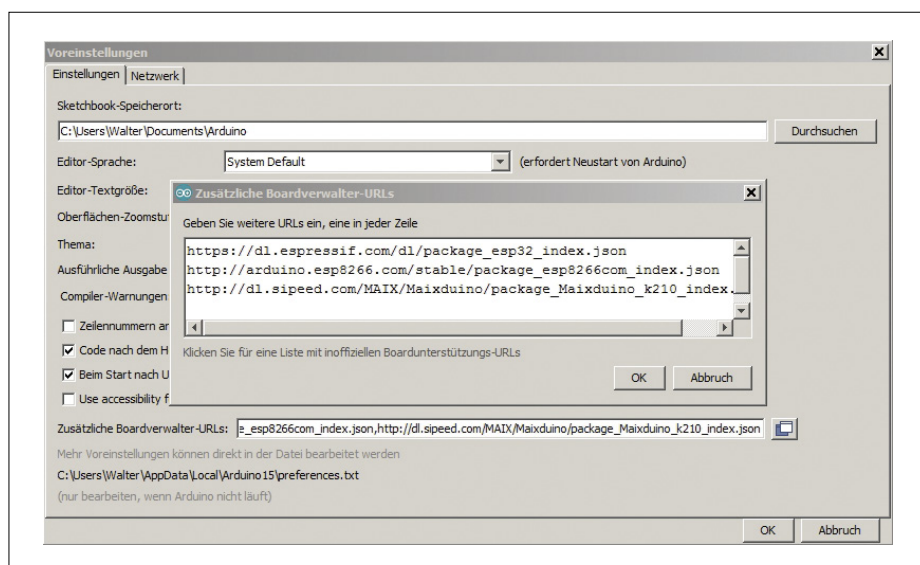
De KPU kan geavanceerde architecturen van neurale netwerken afbeelden, bijvoorbeeld convolutionele netwerken, die worden gekenmerkt door een bijzonder efficiënt oplossingsproces en door speciale filtermethoden bijzonder geschikt zijn voor beeldbewerking. Meer hierover in het tweede artikel. Het snelle hoofdgeheugen dat in het SoC is geïntegreerd, heeft een capaciteit van 8 MB, verdeeld over de processoren en de KPU in een verhouding van 6 op 2 MB en maakt het mogelijk om maximaal 5,9 MB te gebruiken voor neurale netwerken. Daarmee kunnen we middelgrote netwerken implementeren.

Maar de K210 heeft nog meer hardware-pareltjes te bieden. Ten eerste is er de audioprocessor (APU) voor het analyseren van audiodata, die met name nuttig is voor het voorbereiden van spraakanalyse. Deze heeft 8 kanalen (of 4 stereokanalen). De ingangssignalen kunnen worden opgenomen met een bemonsteringsfrequentie tot 192 kHz en het frequentiespectrum wordt ontleed door een FFT-unit (*Fast Fourier Transformation*).

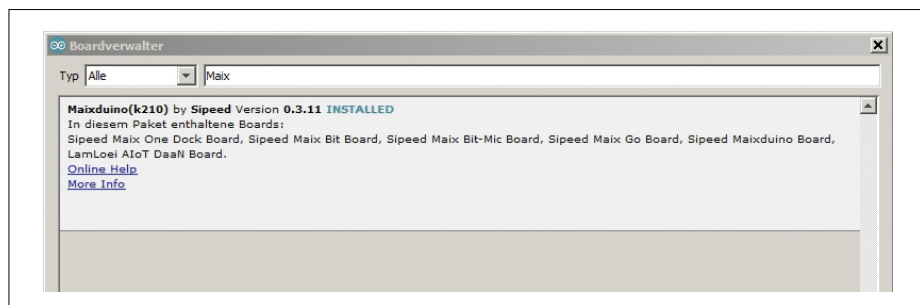
Ook vermeldenswaard zijn de versnellers voor encryptie, de AES Accelerator en de SHA256 Accelerator.

Natuurlijk is de gebruikelijke periferie zoals UART, I²C, SPI, I²S, timers, RTC en PWM ook aan boord.

Sipeed heeft het Maixduino-board ook uitgerust met 16 MB flashgeheugen, een geschikte voeding, microfoon en 3 W versterker en de grootse ESP32. Hardware-hart, wat wil je nog meer? Zie



Figuur 4. Invoer voor de Maixduino-core.



Figuur 5. Installatie van de Maixduino-core.

de weblinks [1][2][3] voor meer informatie over ingebruikname, schema's en andere details.

Software-ontwikkeling

Niet alleen het hardware-ontwerp maar ook de software-ontwikkeling is gebaseerd op de originele Arduino, daarbij wordt gebruik gemaakt van de Arduino-IDE. Om deze ontwikkelomgeving uit te breiden wordt de Maixduino-kern geïntegreerd in de IDE, net als bij de ESP8266 of de ESP32. Hiervoor wordt in het menu onder *File* → *Preferences* een aanvullende URL voor de boardmanager ingevoerd. Klik op de knop rechts van de invoerregel om een klein extra venster te openen dat de invoer gemakkelijk (figuur 4). Als u alleen de Maixduino wilt programmeren, dan is alleen de "sipeed"-regel nodig, omdat de ingebouwde ESP32 daarmee ook wordt geprogrammeerd.

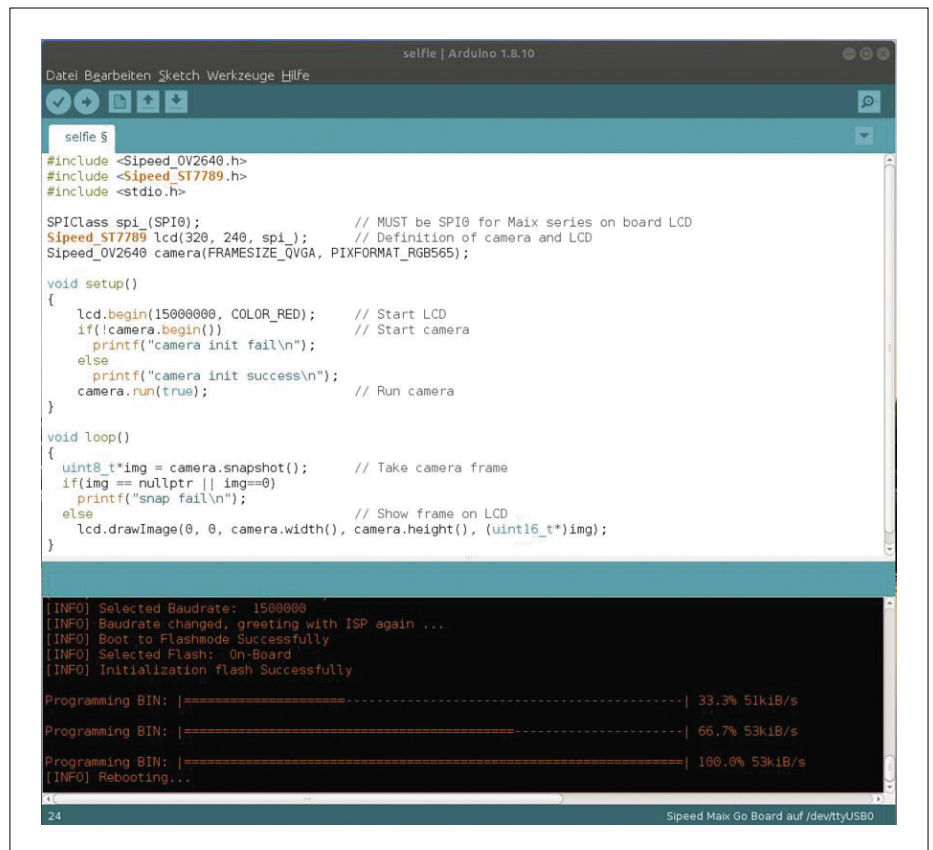
Om de Maixduino-core te installeren, wordt de boardmanager opgeroepen via de menukeuze *Tools* → *Board*. Na het invoeren van de zoekterm "Maix" (figuur 5) kan de installatie worden uitgevoerd.

En dan kunnen we meteen beginnen met programmeren. We slaan het gebruikelijke "hallo wereld" hier maar over en gaan meteen beginnen met het uitproberen van de camera en het display. Sluit de Maixduino aan via USB en stel de parameters voor het board in via het menu-item *Tools*:

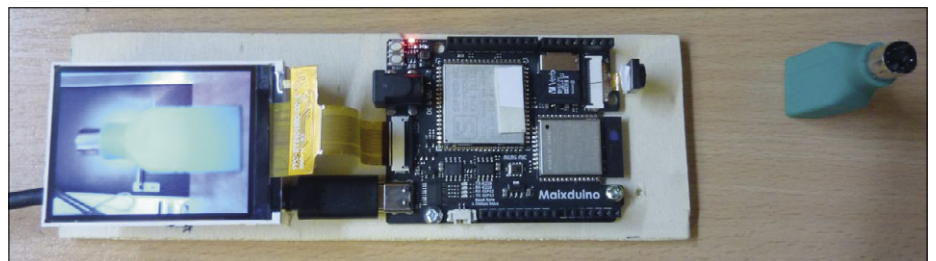
Board: [Sipeed Maixduino](#)
CPU Clock Frequency: [400 MHz](#)
Burn Tool Firmware: [open-ec](#)
Burn Baud Rate: [1.5 Mbps](#)
Tool Install Location: [Standard](#)
Port: [<Gebruikte COM-poort>](#)
Programmer: [k-flash](#)

Een demoprogramma voor het vastleggen en weergeven van beelden op het LCD is al beschikbaar in de IDE; het wordt aangeroepen via *File* → *Examples* → *Sipeed_OV2640* → *selfie*. Na het uploaden wordt het programma gestart (druk, indien nodig, op de resetknop). De code is weergegeven in figuur 6.

Aan het begin van het programma zien we de definitie van vereiste Sipeed-bibliotheken voor camera en LCD, die via de SPI-bus op het board zijn aangesloten. De gekozen resolutie komt overeen met de QVGA-standaard en bedraagt 320x240 pixels (kleur, RGB565). De



Figuur 6. Arduino-IDE met selfie-programma.



Figuur 7. Opname van een adapter.

setup-routine initialiseert de beide apparaten. En in de eindeloze lus worden de opgenomen beelden direct op het LCD-scherm weergegeven. Een kind kan de was doen.

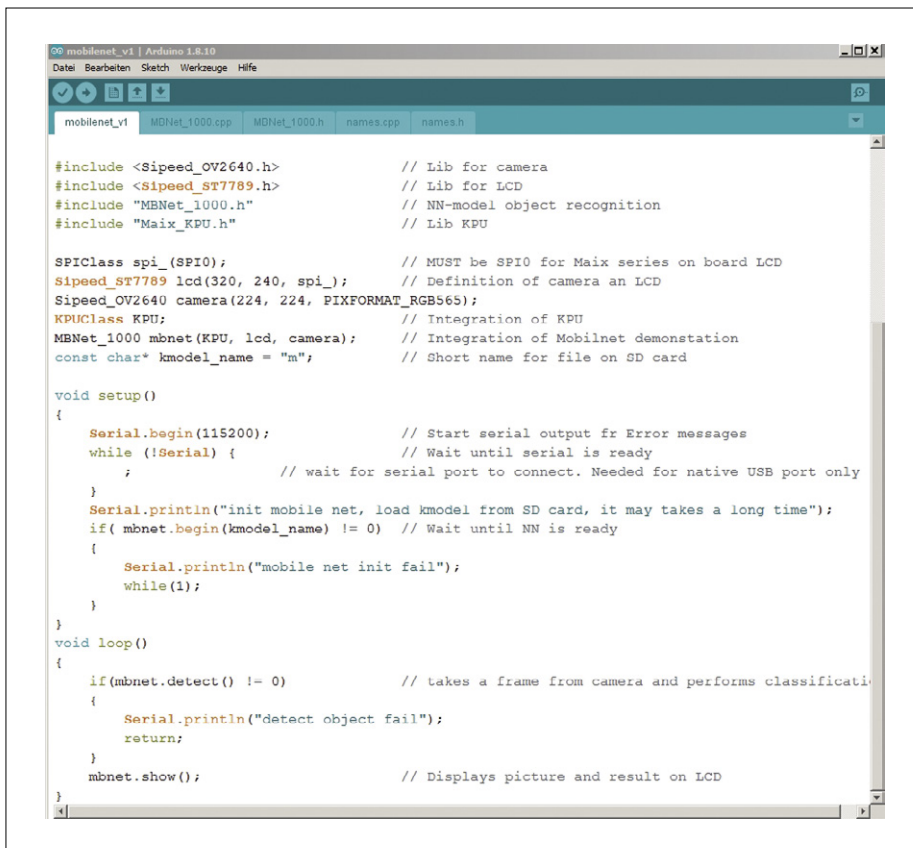
Het beeld is niet erg contrastrijk, maar het is wel scherp en de weergave is vloeiend. In figuur 7 zien we het resultaat. Het blijkt dat de Maixduino net zo toegankelijk is als een Uno, maar hij biedt veel meer mogelijkheden en we kunnen direct beginnen met meer veeleisende toepassingen.

Een eerste AI-model

Eigenlijk willen we pas in het tweede deel meer in detail ingaan op kunstmatige

intelligentie en Deep Learning, maar we demonstreren hier toch alvast een eerste toepassing. Het is de demo *Mobilenet*, die niets te maken heeft met mobiele telefonie, maar wel 1000 objecten uit het dagelijks leven kan herkennen en classificeren. Hiervoor worden aan een neurale netwerk, een softwarestructuur bestaande uit in lagen gerangschikte knooppunten, in een complexe trainingsprocedure duizenden beelden van de objecten gepresenteerd.

Aangezien het bestand met de foto's voor de training een omvang heeft van ongeveer 200 GB en het leerproces zeer complex is, zou het nauwelijks haalbaar zijn op de Maixduino. Maar er is



Figuur 8. Demoprogramma Mobilenet.

de mogelijkheid om een reeds getraind model te gebruiken en direct te beginnen met het herkennen van objecten. En dat is precies wat we hier willen laten zien. De bijbehorende software is beschikbaar onder link [4] en bevat de volgende componenten:

- **mobilenet_v1.ino**: C++ hoofdprogramma voor het uitvoeren van de demo
- **MBNet_1000.h**: header-bestand voor demoroutines
- **MBNet_1000.cpp**: C++ routines voor het verwerven, herkennen en

weergeven van objecten

- **names.h**: header-bestand voor de objectnamen
- **names.cpp**: C++ routine met de namen van de herkenbare objecten

Download deze bestanden en plaats ze allemaal in één map. Het vooraf getrainde AI-model *mobilenet_0x300000.kfpkg* is te vinden onder [5], gecomprimeerd is het maar een paar MB groot. Na het uitpakken met 7zip vindt u een map *mobilenet_0x300000* die twee bestanden bevat, één daarvan is het bestand "m" met het model. Kopieer dit bestand



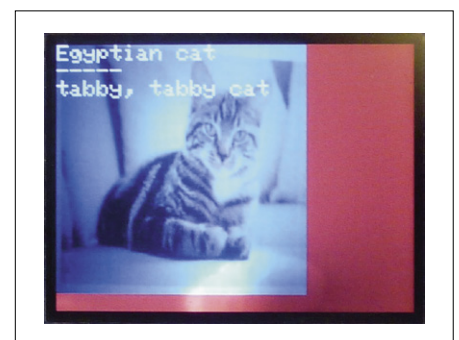
Figuur 9. Fotograferen van een object.

naar de hoofdmap (de bovenste map) van een microSD-kaart en plaats de kaart in de sleuf van de Maixduino.

Een blik op de programmacode in **figuur 8** leert ons dat zelfs zeer complexe toepassingen met weinig moeite kunnen worden geprogrammeerd. Dat is natuurlijk te danken aan de grote prestaties van de bibliotheken. Op het gebied van AI zijn veel verschillende, zeer efficiënte libraries beschikbaar. Het eerste deel van het programma wordt gebruikt om de camera, het LC-display, de KPU en de demoapplicatie te koppelen en de bijbehorende parameters in te stellen. Daarbij wordt de camera ingesteld op het formaat van de trainingsfoto's, namelijk 224x224 pixels. Tenslotte krijgt het object **mbnet** de resources KPU, camera en LCD toegewezen.

Dan wordt in de setup-routine de demonstratie gestart en in de lus worden de objecten die voor de camera verschijnen geïdentificeerd en weergegeven op het LCD. Dit is een heel gemakkelijkere manier om een objectdetectie te programmeren.

Als testobject toonden we de Maixduino een foto van een kat (**figuur 9**), die hij onmiddellijk identificeerde als een cyprese kat (**figuur 10**). Het programma classificeert ca. 5 frames per seconde en zou dus zelfs bij bewegende beelden bruikbare resultaten kunnen produceren. Maar de belichting moet goed zijn en de achtergrond mag niet te verwarrend zijn. **Figuur 10** laat zien dat het beeld nu een blauw-witte kleur heeft om een hogere detectiesnelheid te bereiken. Dit is een vaak toegepaste methode: in plaats van te werken met een beeld in drie kleuren, wordt er met één kleur gewerkt om het aantal pixels en dus de nodige rekeninspanning te verminderen. Het gebruik van een vooraf getraind NN, d.w.z.



Figuur 10. Gedetecteerd en geïdentificeerd object.

training en classificatie op verschillende platforms, is in deze sector vrij gebruikelijk. Veel gebruikers gebruiken de grote rekenkracht van AWS, Microsoft Azure of Google Cloud voor het trainen en draaien het NN op een platform met een kleinere reken capaciteit. De halfgeleiderindustrie speelt hier al op in. Zo heeft Intel onlangs de Neural Network Processors Nervana™ NNP-T en NNP-I aangekondigd. De T-versie heeft een groot rekenvermogen voor het trainen en de I-versie is alleen bedoel om het getrainde netwerk op te gebruiken.

Wordt vervolgd

Met deze demonstratie van een AI-applicatie konden we het onderwerp alleen maar introduceren, maar hopelijk



IN DE STORE

→ Sipeed MAix BiT Kit voor RISC-V AI+IoT

www.elektor.nl/sipeed-maix-bit-kit-for-risc-v-ai-iot

is uw interesse in lerende machines hiermee gewekt.

In het volgende artikel wordt dieper ingegaan op de structuur en werking van een neuraal netwerk. Dan komt er een nieuwe ontwikkelomgeving, want we willen u vertrouwd maken met Linux en de programmeertaal Python. Daar zijn de krachtigste bibliotheken en frameworks voor beschikbaar. Die maken bijvoorbeeld het ontwerp van een neuraal

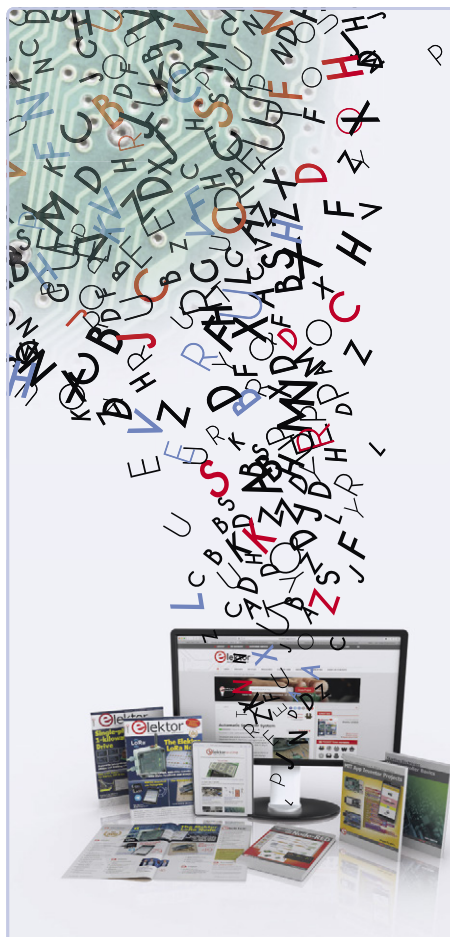
netwerk bijna net zo eenvoudig als het in elkaar passen van Legostenen. Als extra AI-toepassing gaan we dan kijken naar gezichtsherkenning en het ontwikkelen van eigen NN's. Wie niet kan wachten, zou het boek "Neuronale Netze selbst programmieren" van Tariq Rashid moeten lezen. ◀

200023-04

Weblinks

- [1] Beschrijving van de Maixduino: <https://wiki.sipeed.com/en/maix/board/maixduino.html>
- [2] Sipeed-forum: <https://bbs.sipeed.com/>
- [3] Maixduino-schema: <http://dl.sipeed.com/MAIX/HDK/Maixduino/Maixduino-4.30/>
- [4] Mobilenet-demo: https://github.com/sipeed/Maixduino/tree/master/libraries/Maix_KPU/examples/mobilenet_v1
- [5] Mobilenet-model: http://dl.sipeed.com/MAIX/MaixPy/model/mobilenet_0x300000.kfpgk

Advertentie



Ga schrijven... ...en verzet je zinnen!

De Corona-maatregelen grijpen diep in onze manier van leven in, maar kunnen ook mogelijkheden bieden.

Als je een teveel aan vrije tijd nuttig wilt besteden, kun je je **elektronicakennis** met anderen delen – door een **videocursus** te maken of een **artikel** of een **boek** te schrijven. Heb je een goed idee? Ga er dan nu mee aan de slag. Elektor **helpt** je met het **schrijven**, de **redactie** en de **vormgeving**. En het geeft niet alleen veel voldoening, maar je verdient er ook nog geld mee.

Stuur ons je idee, en wij nemen contact met je op.

elektor.com/ga-schrijven

elektor
design > share > sell