



Uitdagingen en kansen van het nieuwe tijdperk: Handleiding voor oudere leerlingen

ARTIE



Co-funded by the
Erasmus+ Programme
of the European Union

ARTIE: Kunstmatige intelligentie in het onderwijs - Uitdagingen en kansen van een nieuw tijdperk: ontwikkeling van een nieuw leerplan, docentenhandleiding en online cursus voor studenten
Medegefinancierd door het Erasmus+-programma van de Europese Unie
2020-1-HR01-KA201-077800

ARTIE

Curriculum voor het onderwijzen van de elementen van kunstmatige intelligentie aan leerlingen in het basisonderwijs

Auteurs

Željko Krnjajić
Janko Radigović

Uitgever

Hrvatski robotički savez, Kroatië
"Kunstmatige intelligentie in het onderwijs - uitdagingen en kansen van het nieuwe tijdperk: ontwikkeling van leerplannen, handleidingen voor docenten en e-cursussen voor studenten", projectnummer 2020-1-HR01-KA201-077800 onder het Erasmus+-programma.

Adviseurs/recensenten

Katarzyna Garbacik
Andrzej Garbacik
Bogusław Klimczuk
Ivana Ružić
Ana Pina
Christina Eirini Karvouna

Grafisch Ontwerp & Illustraties

Christina Eirini Karvouna

Vertalers

Jura Cmrečak (Kroatisch)
Bogusława Denys (Pools)
Ana Pina (Portugees)
Christina Eirini Karvouna (Nederlands)





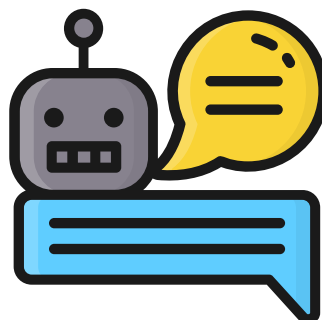
3	Kunstmatige intelligentie in robotica
11	Gezichtsdetectie en -herkenning in Scratch
19	Gezichtsherkenning programmeren in Kras
26	Gezichtsherkenning programmeren in Scratch
41	Objectdetectie en classificatie voor beginners
49	Objectdetectie programmeren in Scratch
57	Project with object classification
63	Spraakherkenning en generatie voor beginners in Kras
68	Erkenning programmeren van spraak in Scratch
73	Programmeren van spraakgeneratie in Scratch
77	Projectobject voor spraakbesturing
83	Inleiding tot hardware - microcontroller, camera en motorcontroller
93	Interactie met de microcontroller camera en computer, begin van de werkzaamheden
100	De robot in elkaar zetten
119	Programmeren van robotbewegingen
128	Robotprogrammering - object volgen
140	Een robot programmeren - regel volgen



AI in robotica

Kan AI in robotica de toekomst veranderen?
Wat is kunstmatige intelligentie in robotica?
Gebruiken we al robots met AI in ons dagelijks leven? Hoe?
Waar?

Doel: Leg uit wat kunstmatige intelligentie is en bespreek de huidige en toekomstige toepassingen ervan in de robotica





Onderwerpen voor discussie:

Wat is een robot?

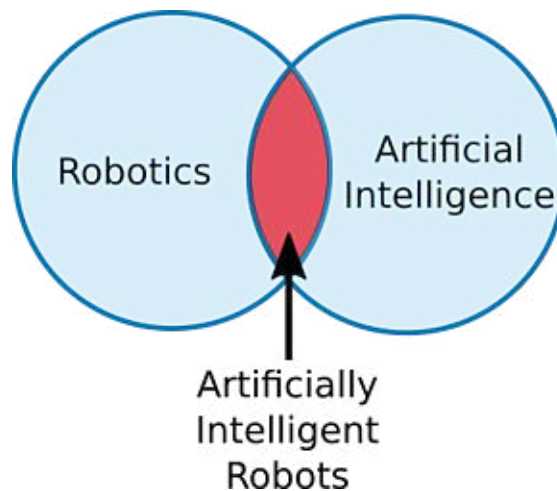
Werkt elke robot met AI?

Kent u enkele voorbeelden van AI in robotica?

Kunnen robots met AI mensen vervangen?

Het eerste dat moet worden verduidelijkt, is dat robotica en kunstmatige intelligentie helemaal niet hetzelfde zijn. In feite zijn de twee velden bijna volledig gescheiden.

Een Venn-diagram van de twee velden ziet er als volgt uit:



Er is een klein gebied waar de twee velden elkaar overlappen: Kunstmatig Intelligent & Robots. Het is binnen deze overlapping dat mensen de twee concepten soms verwarren.

Wat is kunstmatige intelligentie?

Kunstmatige intelligentie (AI) is een tak van de informatica. Het omvat het ontwikkelen van computerprogramma's om taken uit te voeren waarvoor anders menselijke intelligentie nodig zou zijn. AI-algoritmen kunnen leren, perceptie, probleemoplossing, taalbegrip en/of logisch redeneren aanpakken. Volgens de vader van kunstmatige intelligentie, John McCarthy, is het ook "De wetenschap en techniek van het maken van intelligente machines, vooral intelligente computerprogramma's".



Eenvoudig gezegd is kunstmatige intelligentie een manier om een computer, een computergestuurde robot of software intelligent te laten denken, op dezelfde manier als intelligente mensen denken.

Doelen van AI

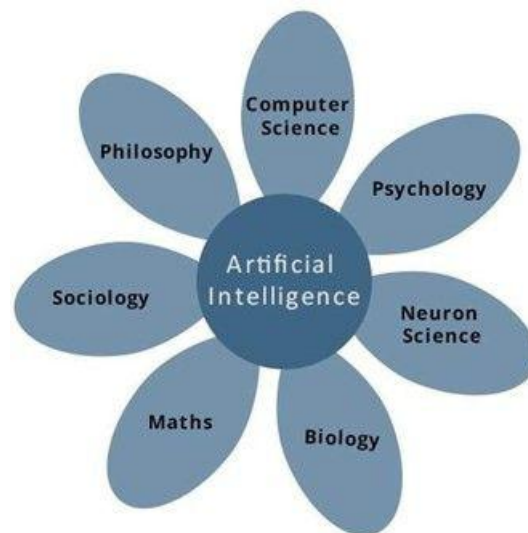
om expertsystemen te creëren – deze systemen vertonen intelligent gedrag, leren, demonstreren, leggen uit en adviseren haar gebruikers.

om menselijke intelligentie in machines te implementeren - dergelijke systemen zouden begrijpen, denken, leren en zich gedragen als mensen.

Wat draagt bij aan AI?

Kunstmatige intelligentie is een wetenschap en technologie gebaseerd op disciplines als informatica, biologie, psychologie, taalkunde, wiskunde en techniek. Een belangrijk onderdeel van AI is de ontwikkeling van computerfuncties die verband houden met menselijke intelligentie, zoals redeneren, leren en probleemoplossing.

Van de volgende gebieden kunnen een of meerdere gebieden bijdragen aan het bouwen van een intelligent systeem.



Wat is Robotica?

Robotica is een tak van AI, die is samengesteld uit elektrotechniek, werktuigbouwkunde en informatica voor het ontwerpen, bouwen en toepassen van robots.

Wat zijn robots?

Robots zijn kunstmatige agenten die in een echte wereld werken.

Het zijn programmeerbare machines die meestal autonoom of semi-autonoom een reeks acties kunnen uitvoeren.



Objectief

Robots zijn ontwikkeld om objecten te kunnen manipuleren door de fysieke eigenschappen van objecten waar te nemen, op te pakken, te verplaatsen, aan te passen, te vernietigen of om mankracht te bevrijden van repetitieve taken zonder zich te vervelen, afgeleid of uitgeput te raken.

Aspecten van Robotica

De robots hebben een **mechanische constructie**, vorm of vorm die is ontworpen om een bepaalde taak uit te voeren.

Ze hebben **elektrische componenten** die de machines aandrijven en besturen.

Ze bevatten een bepaald niveau van computerprogramma's dat bepaalt wat, wanneer en hoe een robot iets doet.

Kunstmatige intelligentie in robotica

AI in robotica helpt robots de cruciale taken uit te voeren met een mensachtige visie om verschillende objecten te detecteren of te herkennen. Robots worden ontwikkeld door machine learning en training en een groot aantal datasets wordt gebruikt om het computervisie-model te trainen, zodat robots verschillende objecten kunnen herkennen, de acties dienovereenkomstig kunnen uitvoeren en de gewenste resultaten kunnen bereiken. Computervisie is gewoon het proces van het waarnemen van de afbeeldingen en video's die beschikbaar zijn in digitale formaten. De AI in robotica helpt niet alleen om het model te leren bepaalde taken uit te voeren, maar het maakt machines ook intelligenter en daardoor in staat om in verschillende scenario's te handelen.

Hier zijn enkele voorbeelden van de meest geavanceerde humanoïde, industriële en servicerobots die de toekomst veranderen met behulp van kunstmatige intelligentie.

Sophia

De meest geavanceerde mensachtige robot van Hanson Robotics, Sophia, personificeert onze dromen voor de toekomst van AI. Als een unieke combinatie van wetenschap, techniek en kunstzinnigheid, Sophia is tegelijkertijd een door mensen vervaardigd sciencefictionpersonage dat de toekomst van AI en robotica uitbeeldt, en een platform voor geavanceerd onderzoek naar robotica en AI.

Het karakter van Sophia spreekt tot de verbeelding van een wereldwijd publiek. Ze is 's werelds eerste robotburger en de eerste robotinnovatie Ambassadeur voor het Ontwikkelingsprogramma van de Verenigde Naties. Sophia is nu een begrip, met optredens op de Tonight Show en Good Morning Britain, naast het spreken op honderden van conferenties over de hele wereld.

Maak kennis met Sophie: <https://www.youtube.com/watch?v=BhU9hOo5Cuc>





Digit

Digit is bedoeld om te helpen bij de zorg voor mensen thuis, bij rampenbestrijding en bij het afleveren van pakketten aan de voordeur. Met zijn behendige ledematen en een romp vol sensoren kan Digit door complexe omgevingen navigeren en taken uitvoeren zoals pakketbezorging. In mei 2019 kondigden Ford Motor Company en Agility een samenwerking aan om een last-mile-logistiekoplossing te ontwikkelen die de autonome voertuigtechnologie van Ford en Digit van Agility combineert.



Pepper

Pepper is 's werelds eerste sociale mensachtige robot die gezichten en fundamentele menselijke emoties kan herkennen. Pepper is overgenomen door meer dan 2000 bedrijven over de hele wereld. Perfect in de detailhandel en de financiële sector, Pepper heeft tal van functionaliteiten, waaronder het vergroten van het winkelverkeer door de aandacht van het winkelend publiek te trekken, gedenkwaardige ervaringen in de winkel te creëren, aankopen te stimuleren en klanten te behouden. Pepper kan ook uitgebreide gegevens verzamelen om het klantenbestand te verrijken en shopperinzichten te genereren.



<http://erasmus-artie.eu>

Atlas

Atlas is 's werelds meest dynamische mensachtige robot gebouwd door BostonDynamics, een bedrijf dat voorheen eigendom was van Google en nu van SoftBank. Atlas wordt jaar na jaar geavanceerder, dankzij de ultramoderne hardware en het algoritme waarmee het snel instructies kan begrijpen. Met zijn 28 hydraulische gewrichten, 1,5 meter hoog en 176 pond in gewicht, kan de robot zowel indrukwekkende als angstaanjagende handelingen uitvoeren, waaronder navigeren op oneffen terrein, springen over een parkourparcours en salto's maken. Al deze activiteiten demonstreren behendigheid op menselijk niveau, zodat de robot perfect kan zijn voor zoek- en reddingsoperaties en het uitvoeren van menselijke taken in omgevingen waar mensen niet kunnen overleven.





Spot

Spot is een robothond die is ontworpen voor industrieel gebruik, zoals het vervoeren van goederen door een magazijn en het inspecteren van een afgelegen locatie met een ongunstige omgeving voor menselijke operators. Het kan draaien op 5,2 voet per seconde, heeft 360-graden camera's en kan werken bij temperaturen van 4 tot 113 Fahrenheit. Met zijn API en flexibele payload-interface kan de robot eenvoudig worden aangepast voor de gewenste taken. Spot wordt ook geproduceerd door BostonDynamics en wordt nu verhuurd aan in aanmerking komende bedrijven.



HRP-5P

HRP-5P is een geavanceerde humanoïde robot die is ontworpen om autonoom te werken en zwaar werk te verrichten in gevaarlijke omgevingen. Het is uitgerust met omgevingsensoren en objectherkenning, bewegingsplanning en -besturing voor het hele lichaam, en taakbeschrijving en uitvoeringsbeheer. HRP-5P is gebaseerd op meer dan 20 jaar humanoïde onderzoek bij AIST. In die 20 jaar heeft het instituut 4 andere robots gemaakt die de voorloper zijn van HRP-5P.



<http://erasmus-artie.eu>

Surena IV

Surena IV is de vierde generatie van de Surena humanoïde robotserie, ontwikkeld door de Universiteit van Teheran in Iran. Met een hoogte van 5,6 voet en een gewicht van 154 pond kan deze robot lopen met een snelheid van 0,43 mijl per uur. De aangepaste krachtsensoren aan de onderkant van zijn voeten helpen de robot over oneffen oppervlakken te stappen door de hoek en positie van elke voet aan te passen.





Aquanaut

Aquanaut is an advanced unmanned underwater transformer that can transform itself from a nimble long-distance submarine into a half-humanoid robot capable of carrying out complex underwater manipulation tasks. Designed by Houston Mechatronics Inc, Aquanaut can inspect subsea oil and gas infrastructure, operate valves, and use subsea tools with just a few mouse clicks. Operating completely untethered and without support ships, Aquanaut can travel more than 124 miles in submarine mode, has a max speed of 7 knots and a maximum operational depth of 984 feet.



f

Stuntronic robot

A Stuntronic robot is an animatronic stunt double designed to entertain the crowds at Disney theme parks and resorts. With its onboard sophisticated sensors, it can make its own real-time decisions—all that while flying at 60 feet up in the air. It knows when to tuck its knees to perform a somersault, when to pull its arms to twist, and even when to slow down its spin to make sure it makes a perfect landing.



<http://erasmus-artie.eu>

Handle

Handle is another robot from Boston Dynamics. With its deep-learning vision software, this robot can identify and locate boxes, unload trucks, palletize, and depalletize at the push of a button. Its mobility enables it to operate in multiple work-cells, moving through facilities along with the flow of goods. It can pick up to 360 boxes/hr.



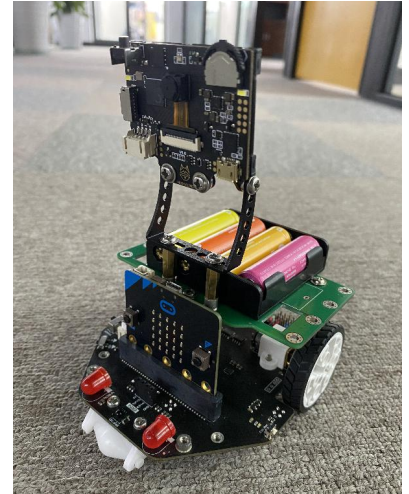


Tegenwoordig gebruiken we AI in robotica in de gezondheidszorg, de landbouw, de auto-industrie, in magazijnen, in de toeleveringsketen, enz. Later gaan we onze eigen robot bouwen met AI en hem trainen om gezichtsdetectie en -herkenning, objectdetectie en spraakherkenning te doen.

Hier is een mobiele robot met camera- en AI-mogelijkheden die exclusief voor dit project is ontwikkeld.

We laten je zien hoe je het maakt en gebruikt voor:

- Gezichtsherkenning
- Gezichtsherkenning
- Gezicht volgen
- Objectdetectie
- Object volgen
- Lijn volgen



Kunstmatige intelligentie is eindelijk hier en de meesten van ons maken er al actief gebruik van in ons dagelijks leven (zelfs zonder het te weten). Toekomstige generaties moeten allereerst begrijpen hoe ze AI moeten gebruiken! Alleen dan kunnen ze het gebruiken om leren te vergemakkelijken en problemen uit de echte wereld op te lossen.

Artificial Intelligence (AI) and Robotics are strongly connected today.

AI in robotica wordt steeds meer in het dagelijks leven gebruikt en heeft een belangrijke rol gespeeld in de verschillende domeinen zoals industrie, leger, geneeskunde, verkenning, entertainment.

Vergeet niet dat AI hoogstwaarschijnlijk de krachtigste technologie is die ooit door de mens is uitgevonden. Het kan zowel voor goede als slechte dingen worden gebruikt. Uiteindelijk is het aan ons hoe het te gebruiken.



<http://erasmus-artie.eu>

CONCLUSIE

Kunstmatige intelligentie is de manier waarop een computer, een computergestuurde robot of programma denkt intelligent na, op een vergelijkbare manier waarop intelligente mensen denken

Think!



What do I know now?



What did I learn?



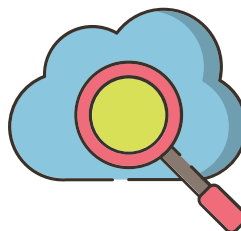
What do I want to know?



Gezichtsdetectie en -herkenning voor beginners in Scratch

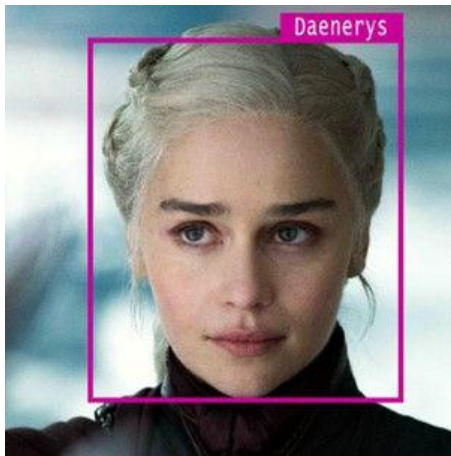
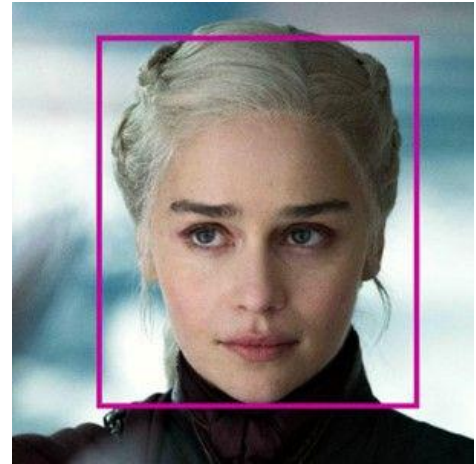
De termen gezichtsdetectie en gezichtsherkenning worden soms door elkaar gebruikt, maar er zijn enkele belangrijke verschillen. Laten we, om de zaken op te helderen, eens kijken naar de term gezichtsherkenning en hoe deze verschilt van de term gezichtsdetectie.

Doel: introductie tot gezichtsdetectie en -herkenning voor beginners door middel van voorbeelden van verschillende toepassingen.





Gezichtsdetectie is een proces dat de aanwezigheid van gezicht(en) in een stilstaand beeld of in een videoclip bepaalt. Een dergelijke functie is bijvoorbeeld beschikbaar in de meeste camera-software voor smartphones. Maar de gezichtsdetectiemodule bepaalt niet wiens gezicht zich in het frame bevindt.



Gezichtsdetectie onthoudt of slaat geen gezichtskenmerken op. Als de software een gezicht van een bepaalde persoon in het frame detecteert en later hetzelfde gezicht op een andere afbeelding aantreft, zal het niet bepalen dat het gezicht van dezelfde persoon is; het detecteert alleen de aanwezigheid van een gezicht in de frames. De software kan op elk frame gegevens verstrekken over leeftijd en geslacht van een persoon, maar niet meer dan dat.

Gezichtsdetectiesoftware kan bepaalde personen niet herkennen.

Gezichtsherkenning daarentegen heeft betrekking op het identificeren en herkennen van personen.

Het doel van op software gebaseerde gezichtsherkenning is het identificeren van personen die voorkomen in een stilstaand beeld of een videoclip door deze te vergelijken met een database. Voor een succesvolle identificatie moeten eerst de bijbehorende gezichten in de database worden ingevoerd. De software bepaalt de unieke kenmerken van een gezicht, slaat ze op en gebruikt ze voor latere identificatie. Later, tijdens het identificatieproces, zal de software de unieke kenmerken vergelijken en een gezicht van een bepaalde persoon identificeren als deze kenmerken overeenkomen.



Scratch/Scratch-gebaseerde en andere toepassingen om te gebruiken

Scratch (ML4KIDS)

<https://machinelearningforkids.co.uk/scratch3/>

Uitbreiding voor gezichtsdetectie met 3 blokken van het type reporter is beschikbaar. Als u een webcam als bron gebruikt, combineer deze dan met de extensie Videodetectie om cameravideo in of uit te schakelen en transparantie in te stellen.

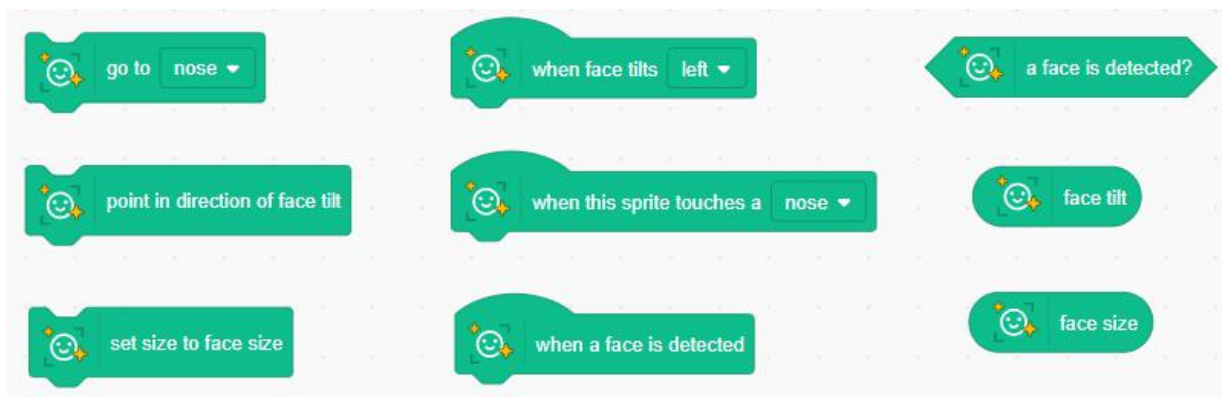


Scratch (MIT) - <https://lab.scratch.mit.edu/face/>

Klik op "Probeer het uit" en je hebt 9 blokken voor het afhandelen van gezichtsherkenning.

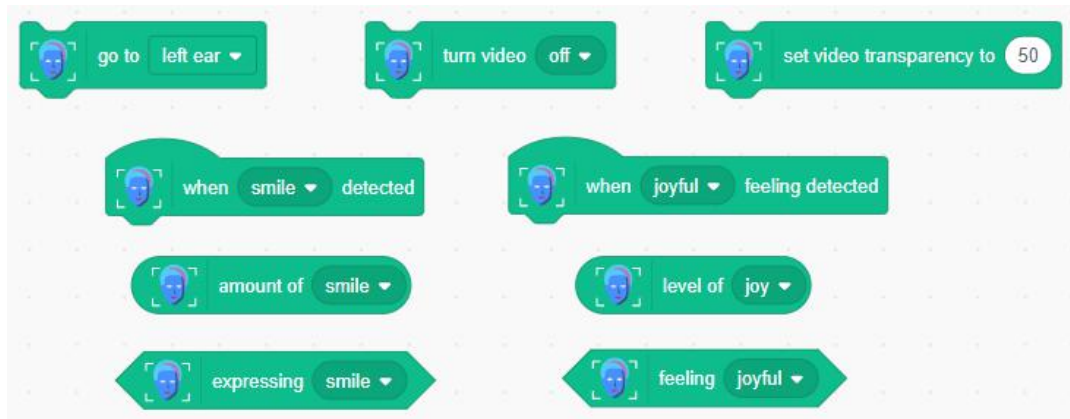


<http://erasmus-artie.eu>





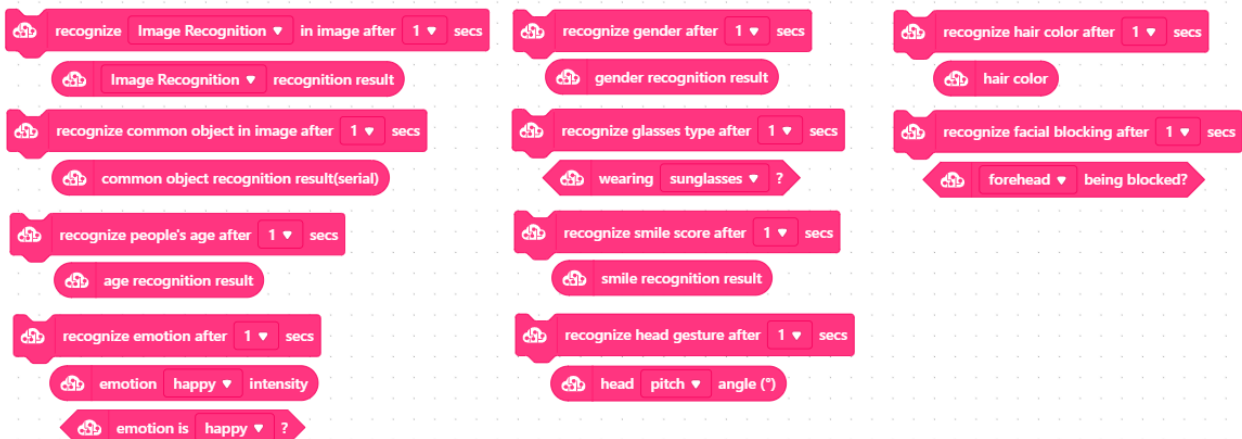
Scratch (MITMEDIALAB) - <https://mitmedialab.github.io/prg-extension-boilerplate/create/>
 Laad de extensie Gezichtsdetectie en u hebt 9 blokken voor het afhandelen van gezichtsdetectie. Sommige blokken worden gebruikt voor gezichtsuitdrukking en gevoelsherkenning. U kunt ook Teachable machine extension gebruiken in combinatie met Google Teachable machine.



Makeblock (mBlock) - <https://ide.mblock.cc/>

Laad Cognitive-services en Video Sensing-extensies en je hebt verschillende blokken. Er is geen specifiek blok voor gezichtsdetectie, maar u kunt een persoon binnen een "herken" -blok detecteren. Maar er zijn genoeg blokken om emoties, leeftijd, geslacht, glimlach, haarkleur, bril en zelfs een situatie waarin je een deel van je gezicht bedekt, aan te kunnen.

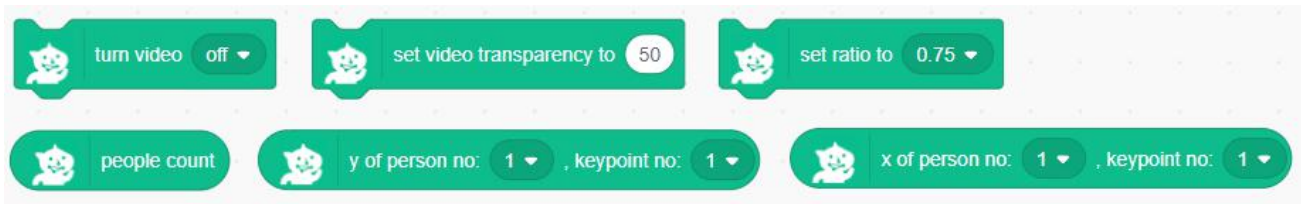
Makeblock biedt ook Teachable machine-extensie (niet te verwarren met die van Google) waar je tot 3 klassen kunt trainen en deze kunt gebruiken voor gezichtsherkenning of objectdetectie.





Stretch3 (github.io) - <https://stretch3.github.io/>

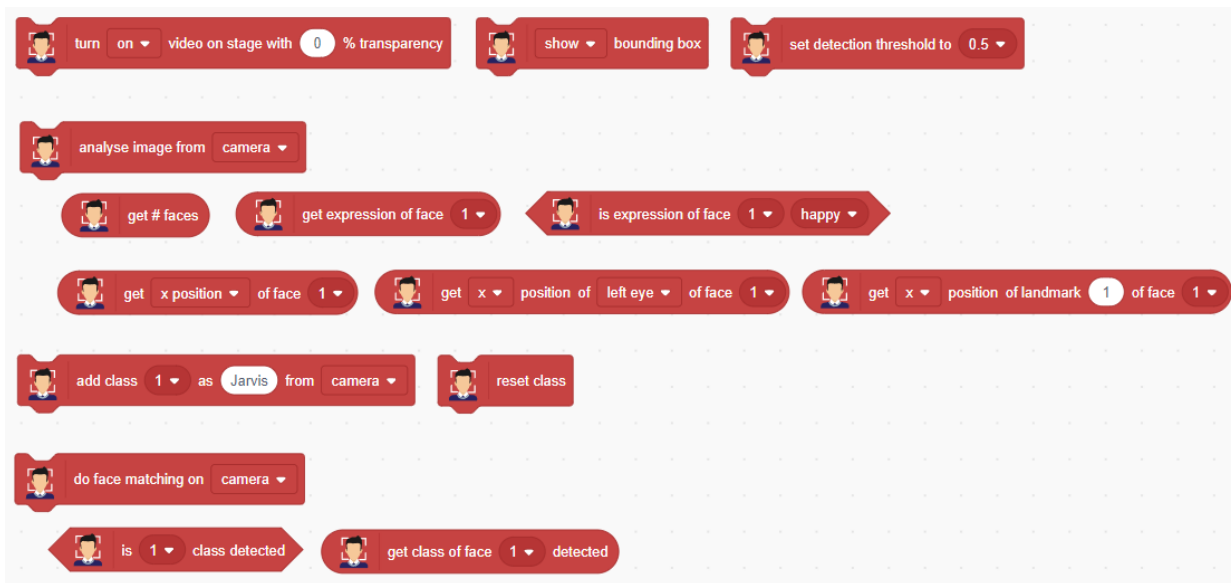
Laad de **Facemesh2Scratch-extensie** om 3 blokken te gebruiken voor gezichtsdetectie (er zijn nog 3 blokken voor het verwerken van video). Het belangrijkste kenmerk is de mogelijkheid om meerdere gezichten te detecteren, zodat u meer dan één persoon in uw camerastream kunt detecteren.



De laatste om te gebruiken is **PictoBlox**, een desktop-achtige applicatie die eerst moet worden geïnstalleerd vanaf <https://thetempedia.com/product/pictoblox/download-pictoblox/> (427 Mb) Gebruik na installatie de Face Detection-extensie en daar zie je een pure schat - je kunt meerdere gezichten en hun gezichtsuitdrukkingen detecteren. Er is ook een functie om klassen te trainen die ons rechtstreeks naar gezichtsherkenning leidt.



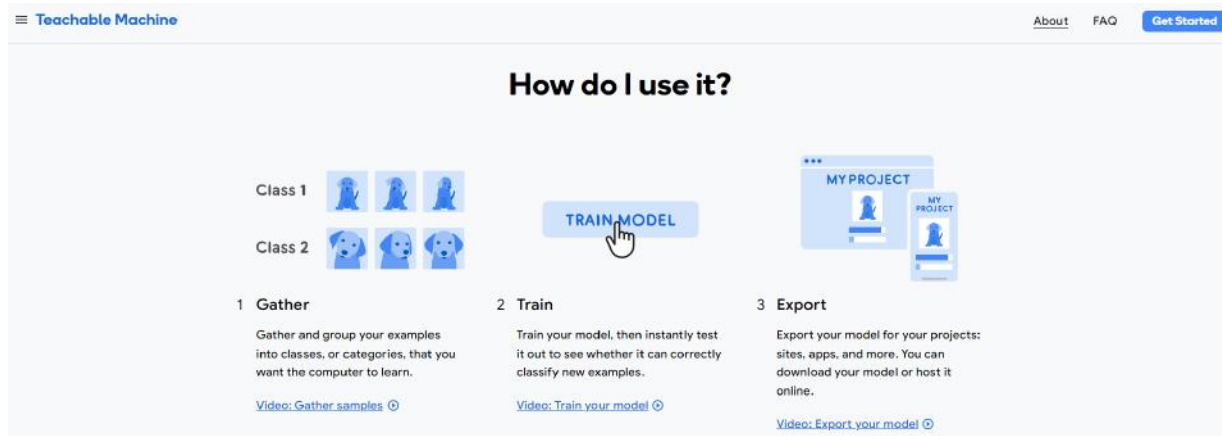
http://erasmus-artie.eu





Teachable machine (Google) - <https://teachablemachine.withgoogle.com/>

Deze applicatie wordt gebruikt om uw model te trainen en te gebruiken voor gezichtsherkenning in combinatie met Teachable machine extension beschikbaar in Scratch (MITMEDIALAB)



We hebben drie verschillende toepassingen gezien, maar ook een zeer vergelijkbare voor gezichtsdetectie en -herkenning.

Gezichtsdetectie verschilt van gezichtsherkenning (de termen mogen niet door elkaar worden gebruikt) doordat Gezichtsdetectie alleen betrekking heeft op de detectie van een gezicht in een digitale afbeelding of video. Het betekent simpelweg dat het gezichtsdetectiesysteem kan identificeren dat er een menselijk gezicht aanwezig is in een afbeelding of video - het kan de persoon niet identificeren. Gezichtsdetectie is een onderdeel van gezichtsherkenningssystemen – de eerste fase van gezichtsherkenning is in de eerste plaats het detecteren van de aanwezigheid van een menselijk gezicht. Gezichtsdetectie kan ook worden gebruikt in camera's om te helpen bij autofocus - je hebt waarschijnlijk gemerkt dat op sommige digitale camera's en telefoons een klein kader verschijnt rond de gezichten van mensen die in het beeld worden gedetecteerd, waardoor de camera prioriteit kan geven aan de focus op die mensen gezichten. Het identificeren van de aanwezigheid van een menselijk gezicht gebeurt met behulp van formules en algoritmen.

Meestal is het eerste waar een gezichtsdetectiesysteem naar zoekt, de ogen, aangezien dit een van de gemakkelijkste kenmerken is om te identificeren. Dan kan het ook zoeken naar de aanwezigheid van mond, wenkbrauwen, neus en neusgaten. Gezichtsdetectie is een belangrijk onderdeel van het gezichtsherkenningproces, maar vanuit veiligheidsoogpunt heeft een gezichtsdetectiesysteem geen onafhankelijk voordeel: het herkent gewoon dat er een gezicht aanwezig is, maar heeft geen idee van de identiteit van dat gezicht. Gezichtsherkenning speelt een cruciale rol in een groot aantal bedrijfstakken, met name bij grenscontrole en wetshandhaving. Het nauwkeurig identificeren van personen helpt de beveiliging en veiligheid op luchthavens en in dorpen en steden over de hele wereld te verbeteren, en dit kan alleen worden gedaan met behulp van toonaangevende gezichtsherkenningssystemen.



Zoals we kunnen zien, verandert gezichtsherkenningstechnologie de wereld waarin we leven, en het voelt alsof we nog maar aan het begin staan wat betreft mogelijke toepassingen van gezichtsherkenningsoftware. Hoewel toepassingen voor gezichtsherkenning eindeloos lijken, moeten we ervoor zorgen dat deze technologie op de juiste en verantwoorde manier wordt gebruikt.

CONCLUSIE

Gezichtsdetectie is een proces dat de aanwezigheid van een gezicht in een afbeelding of videoclip bepaalt terwijl gezichtsherkenning verwijst naar de identificatie en herkenning van personen.

Think!



What do I know now?



What did I learn?



What do I want to know?



<http://erasmus-artie.eu>



Gezichtsdetectie programmeren in Scratch

Hoe programmeer ik gezichtsherkenning in Scratch?

We zullen herhalen wat we in de vorige les hebben geleerd over gezichtsdetectie.

Laten we een paar eenvoudige voorbeelden bekijken van het programmeren van gezichtsherkenning in Scratch en op Scratch gebaseerde applicaties.

Je hebt waarschijnlijk gemerkt dat sommige applicaties op je smartphone een rechthoek rond het gezicht tekenen als gevolg van gezichtsdetectie. Dat kan ook in Scratch.

Doel: Inzicht in software voor gezichtsdetectie en het gebruik ervan door specifieke voorbeelden te bekijken.



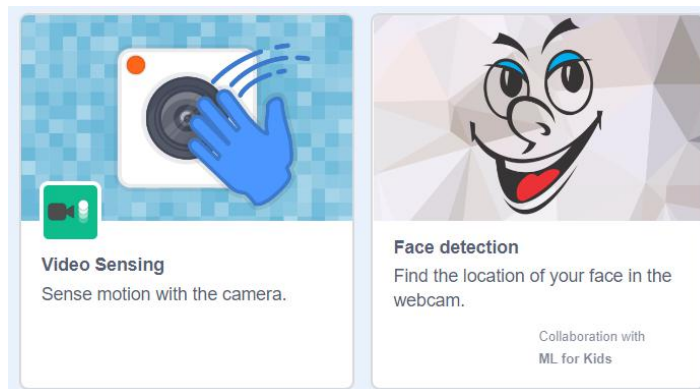
<http://erasmus-artie.eu>



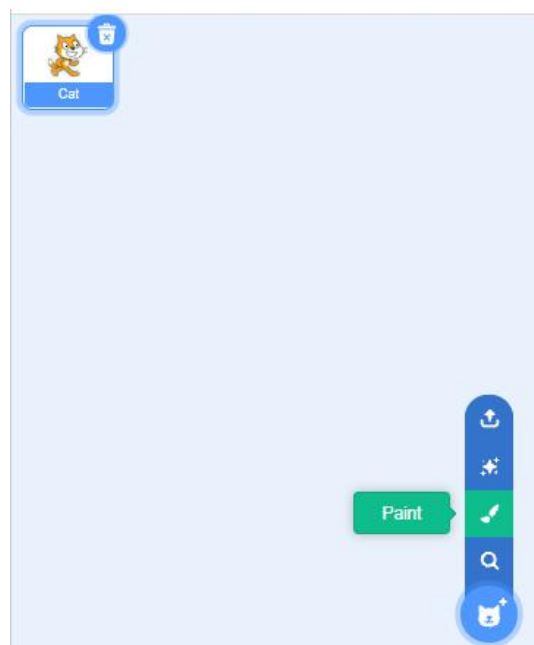
Eerste project- SCRATCH (ML4KIDS):

Stap 1: Open uw Chrome-webbrowser en ga naar:
<https://machinelearningforkids.co.uk/scratch3/>

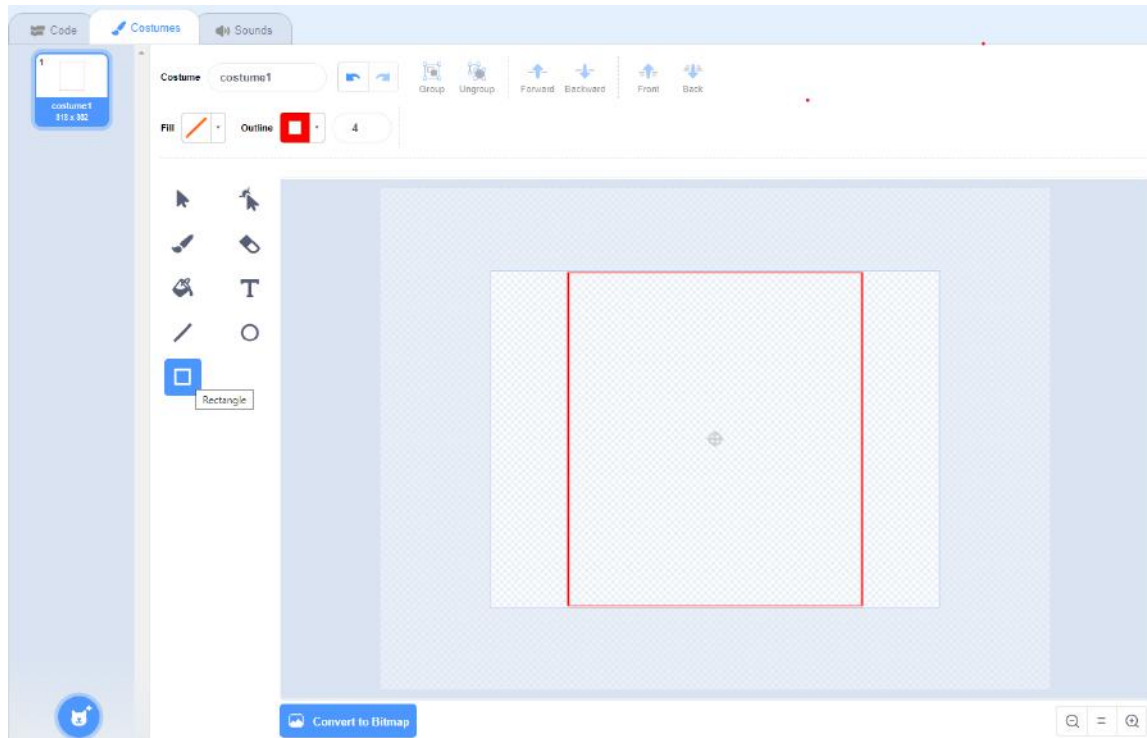
Stap 2: Laad extensies voor videodetectie en gezichtsdetectie en sluit uw webcamera aan
(als u geen ingebouwde camera heeft)



Stap 3: Verwijder Cat sprite door op te klikkenprullenbakpictogram (rechterbovenhoek) en kies de optie om een nieuwe sprite te schilderen

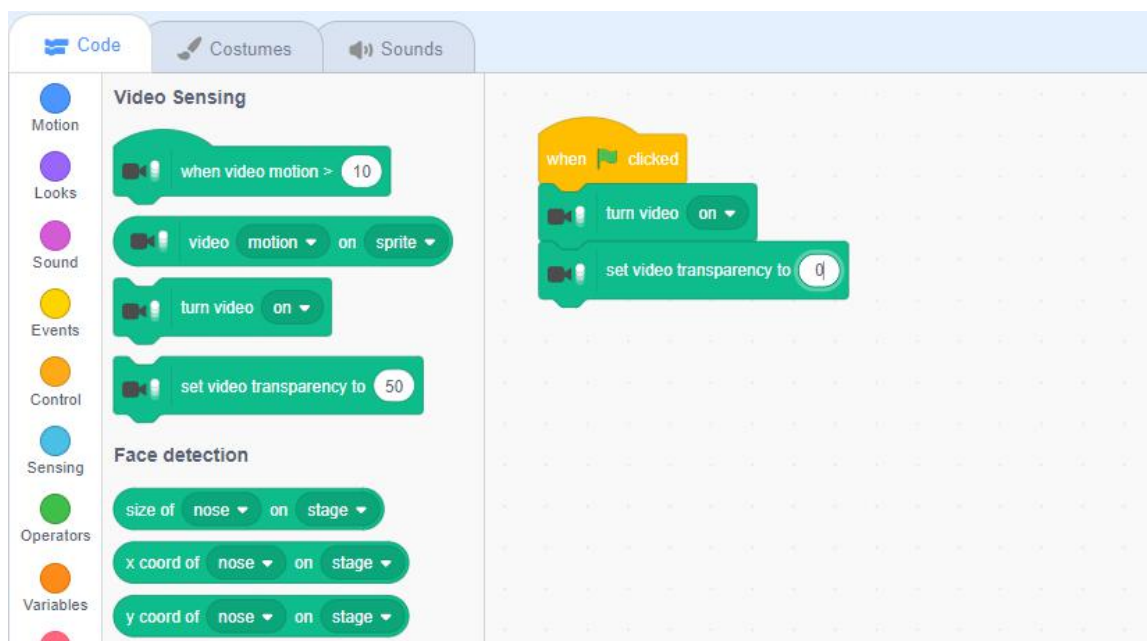


Step 4: Draw a rectangle (it will be used as a bounding box) with no fill and set the outline to red (4) as in picture below



f

Step 5: Switch to Code tab and start programming. First, we turn on the video and set the transparency to 0 (non-transparent).

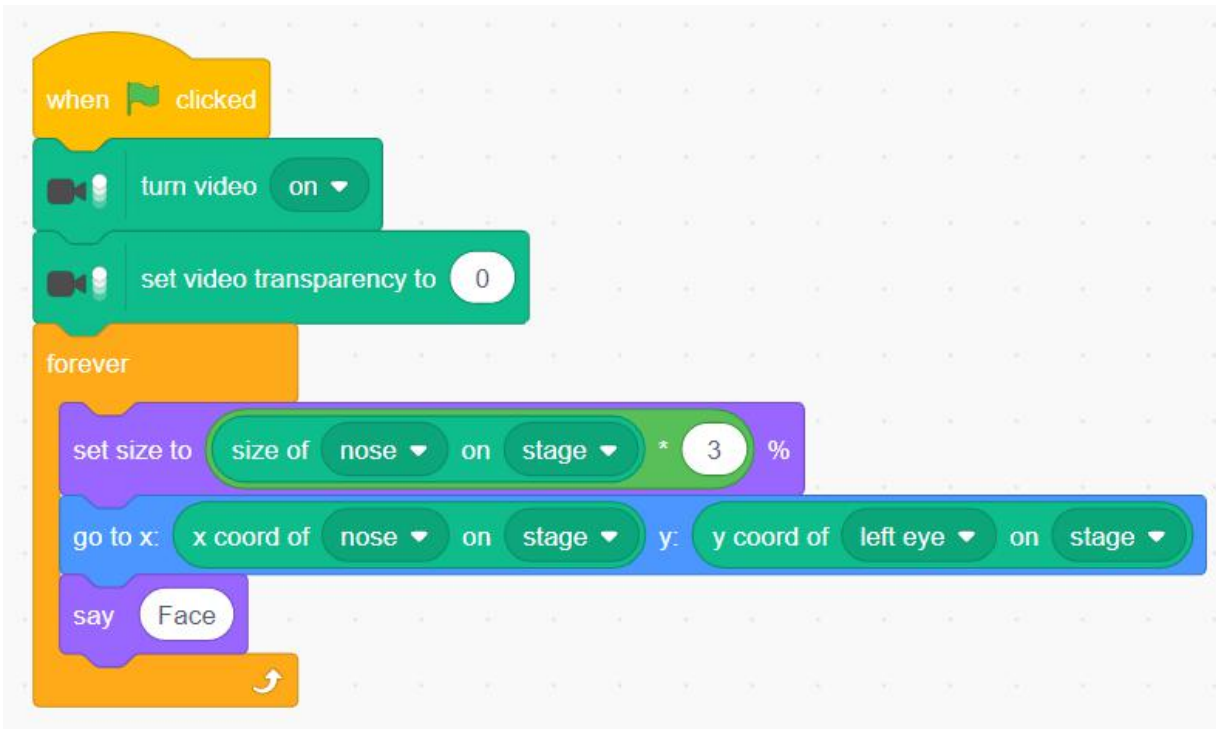


<http://erasmus-artie.eu>

21



Stap 6: Vervolgens is er een lus (voor altijd) met 3 blokken erin om de grootte van een rechthoek (neusmaat vermenigvuldigd met 3) en de x- en y-coördinaten in te stellen. Misschien wilt u de waarde van een vermenigvuldiger aanpassen van 4 naar een ander getal, vooral als u overschakelt naar een grotere trap (1,5 is beter).



Tweede project - gezichtsdetectie gecombineerd met augmented reality

Stap 1: Open Scratch op <https://machinelearningforkids.co.uk/scratch3/>

Stap 2: Voeg extensie "Gezichtsdetectie" toe

Stap 3: Voeg de extensie "Videodetectie" toe en sluit uw webcam aan (als u er geen hebt ingebouwd)

Stap 4: Cat-sprite verwijderen

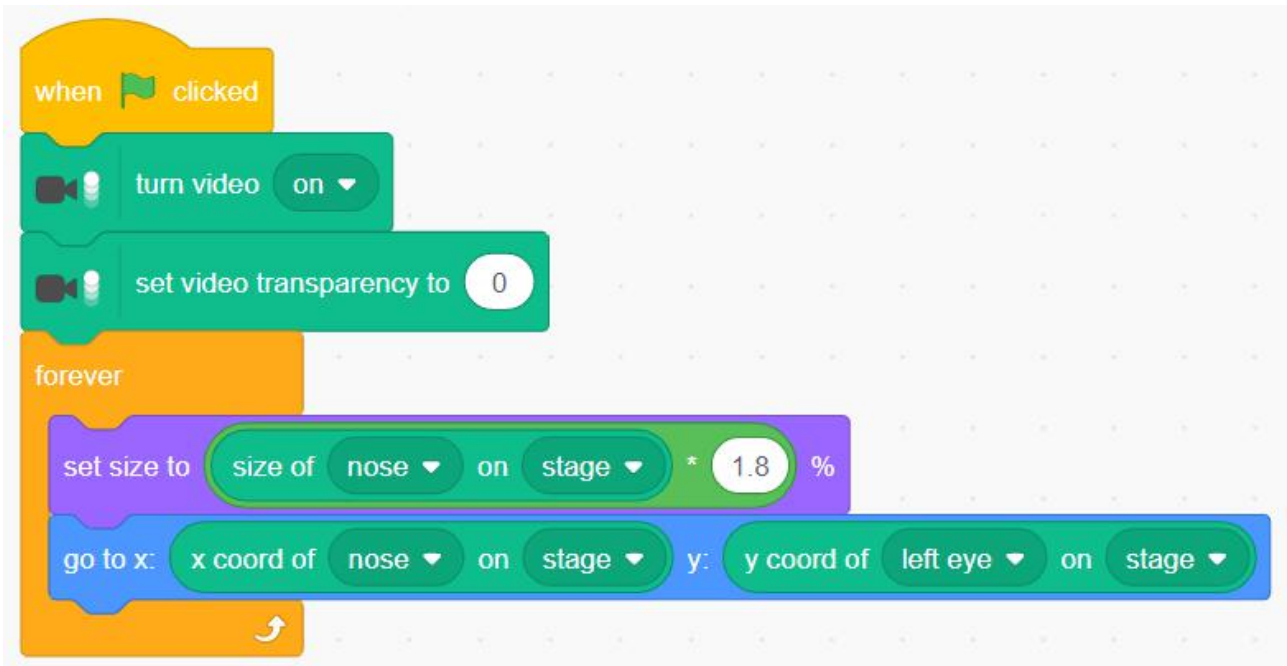
Stap 5: Download afbeelding van link https://toppng.com/transparent-glasses-PNG-free-PNG-Images_110945

Stap 6: Upload foto naar Scratch als sprite, hernoem het naar "brillen"

Stap 7: Sprite-code

Stap 8: Start het programma en beweeg je hoofd

Stap 9: Bespreek de nauwkeurigheid van het algoritme en hoe het kan worden verbeterd.



Derde project met PICTOBLOX (Desktop applicatie):

Stap 1a: Aangezien er geen online GUI beschikbaar is, moet u PictoBlox installeren vanaf:
<https://thestempedia.com/product/pictoblox/download-pictoblox/> (427 Mb)

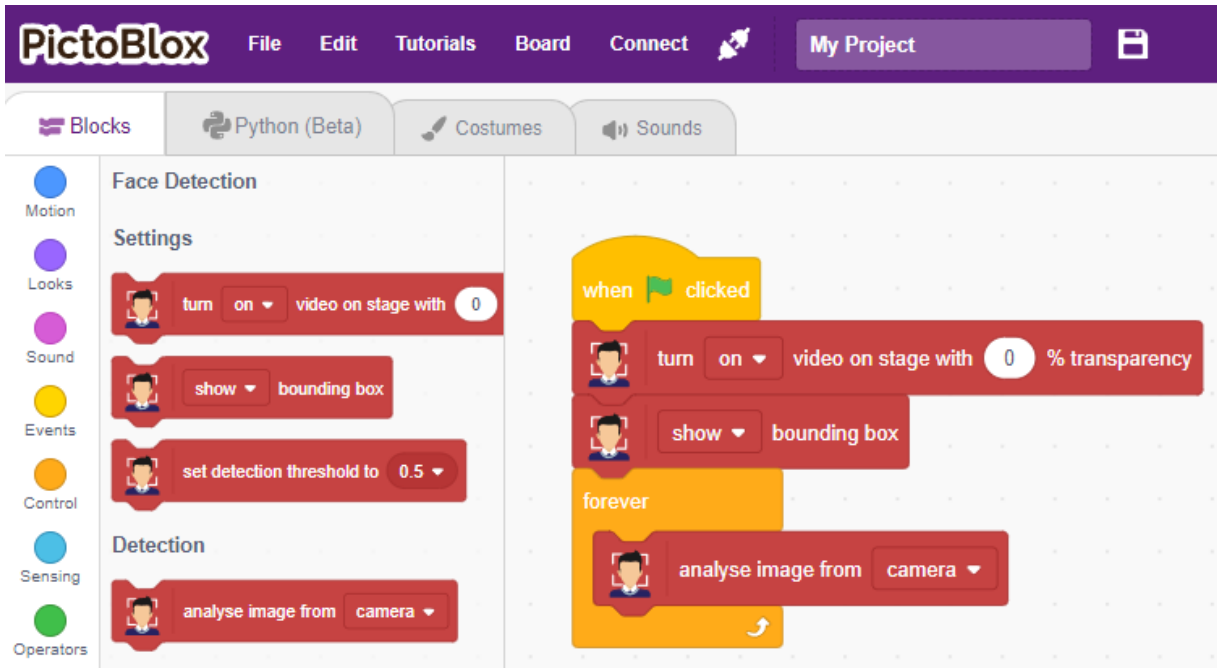
Stap 2a: Open PictoBlox en kies Gezichtsdetectie-uitbreiding



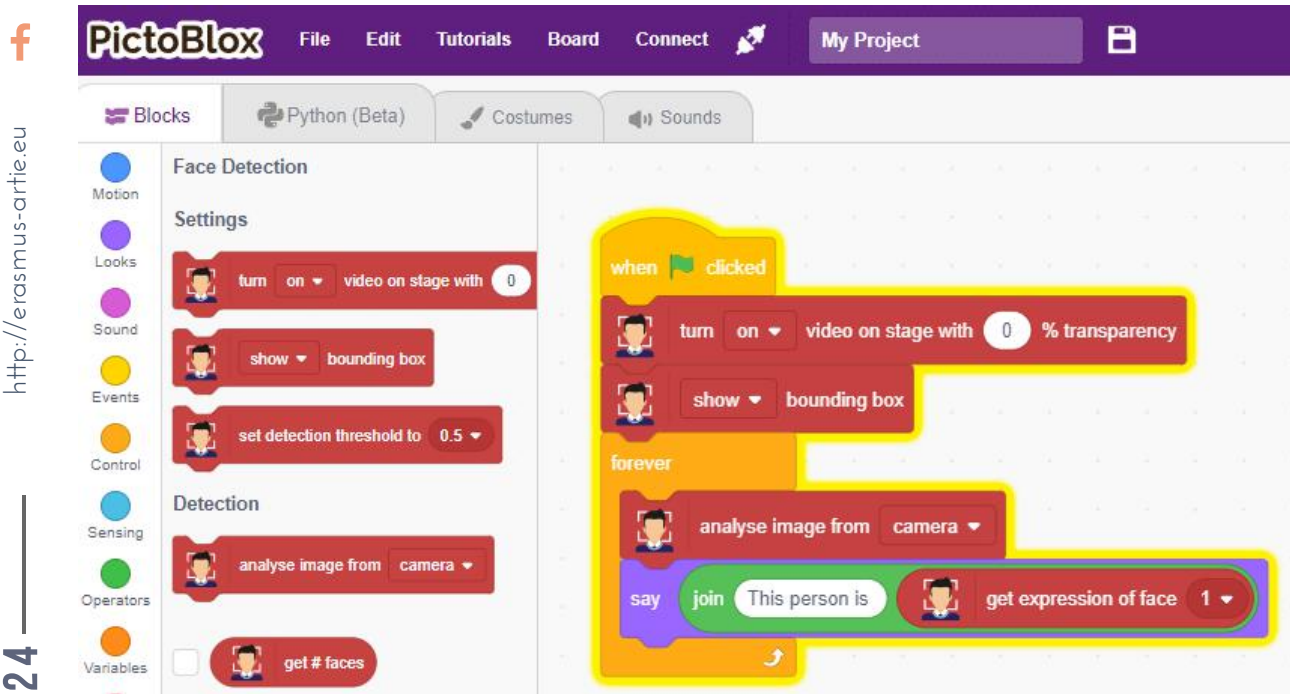
<http://erasmus-artie.eu>



Stap 3a: Gebruik blokken zoals in de afbeelding hieronder en het is heel eenvoudig om erachter te komen hoe het werkt. Nu hebben we een begrenskader als een blok en het is niet nodig om een rechthoek te tekenen. Maar het belangrijkste kenmerk is dat het meerdere gezichten kan detecteren. Bel iemand om mee te doen voor de camera om te zien hoe het werkt. Controleer het reportertype block get # faces om te zien hoeveel gezichten er zijn gedetecteerd.



Stap 4a: En laten we het wat spannender maken door de join-operator te gebruiken om de gezichtsuitdrukking van een persoon weer te geven. U kunt verder onderzoeken hoe het werkt met meerdere gezichten.





Tegenwoordig wordt software voor gezichtsdetectie op bijna elk gebied gebruikt, van mobiele apparaten tot gezichtsfilters voor snapchat tot verschillende beveiligingstoepassingen. Gezichtsdetectie helpt u gezichten, hun leeftijd, uitdrukkingen, geslacht, locatie en vele andere kenmerken te herkennen. Gezichtsdetectie is een bredere term voor elk systeem dat de aanwezigheid van een menselijk gezicht in een visueel beeld kan identificeren. Gezichtsdetectie heeft tal van toepassingen, onder meer bij het tellen van mensen, online marketing en zelfs de autofocus van een cameralens. Het belangrijkste doel is om de aanwezigheid van een gezicht te markeren. Gezichtsherkenning is de afgelopen jaren belangrijker en relevanter geworden vanwege de mogelijke toepassingen. Aangezien gezichten zeer dynamisch zijn en meer problemen en uitdagingen met zich meebrengen om op te lossen, hebben onderzoekers op het gebied van patroonherkenning, computervisie en kunstmatige intelligentie veel oplossingen voorgesteld om dergelijke problemen te verminderen en zo de robuustheid en herkenningsnauwkeurigheid te verbeteren.

CONCLUSIE

Gezichtsdetectie wordt gebruikt in verschillende geavanceerde systemen, en het is mogelijk om eenvoudige voorbeelden te maken in tools als Scratch en PictoBlox.



<http://erasmus-artie.eu>

Think!



What do I know now?



What did I learn?



What do I want to know?

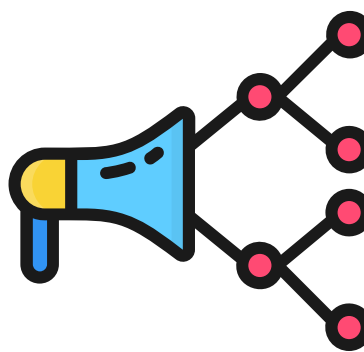


Gezichtsherkenning programmeren in Scratch

Het doel van deze les is om te leren hoe je gezichtsherkenning programmeert in Scratch?

Onderwerpen voor discussie:
Wat is gezichtsherkenning?
Hoe werkt gezichtsherkenning?
Welke praktische toepassingen kan het hebben?

Doel: Aan de hand van voorbeelden van één programma krijgt u een beter begrip van gezichtsherkenningprogramma's en het gebruik ervan.





Stap 1: Open uw webbrowser en download alle 20 afbeeldingen from: <https://bit.ly/daenerys-data>

Het zal worden gebruikt om klasse 1 te trainen

Stap 2: Open uw webbrowser en download alle 20 afbeeldingen van: <https://bit.ly/arya-data>

Het zal gebruikt worden om klasse 2 te trainen

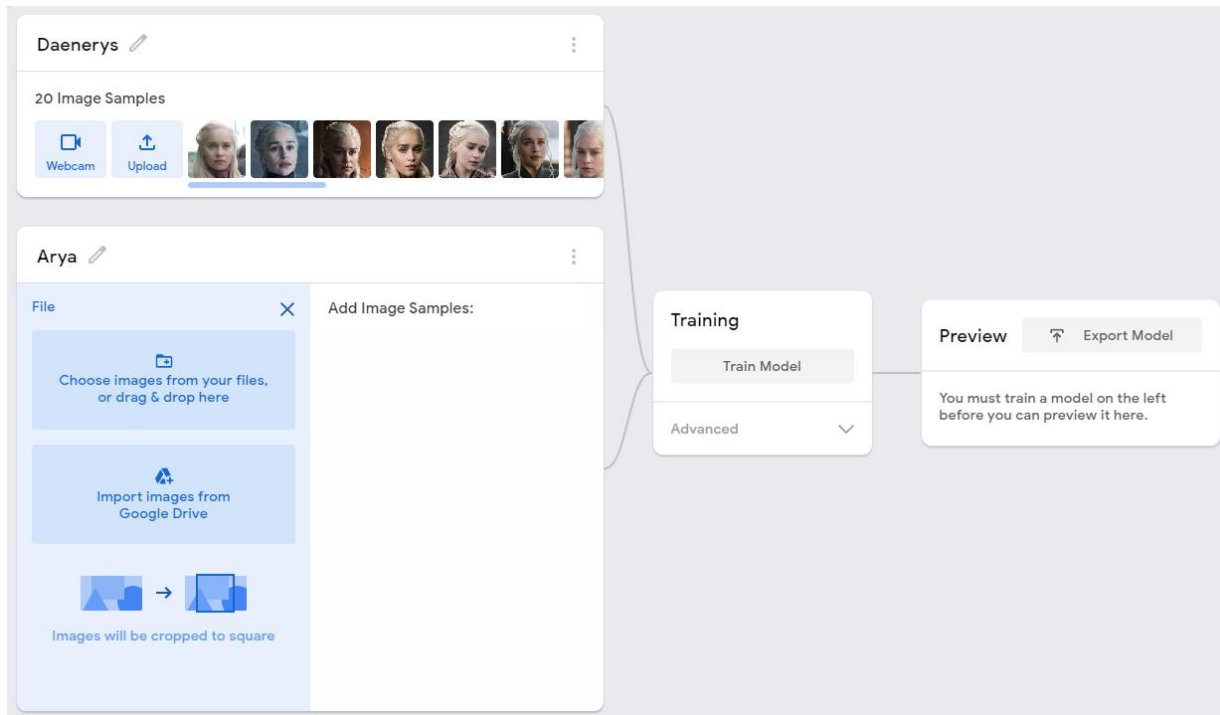
Stap 3: Open uw webbrowser en ga naar: <https://teachablemachine.withgoogle.com/>

Stap 4: Klik op Aan de slag.

Stap 5: Kies het afbeeldingsproject

Stap 6: Kies het standaard afbeeldingsmodel

Stap 7: Wijzig de naam van klasse 1 in Daenerys en klasse 2 in Arya. Upload Daenerys-afbeeldingen naar Daenerys-bestanden en Arya-afbeeldingen naar Arya-bestanden, zoals weergegeven in de onderstaande afbeelding





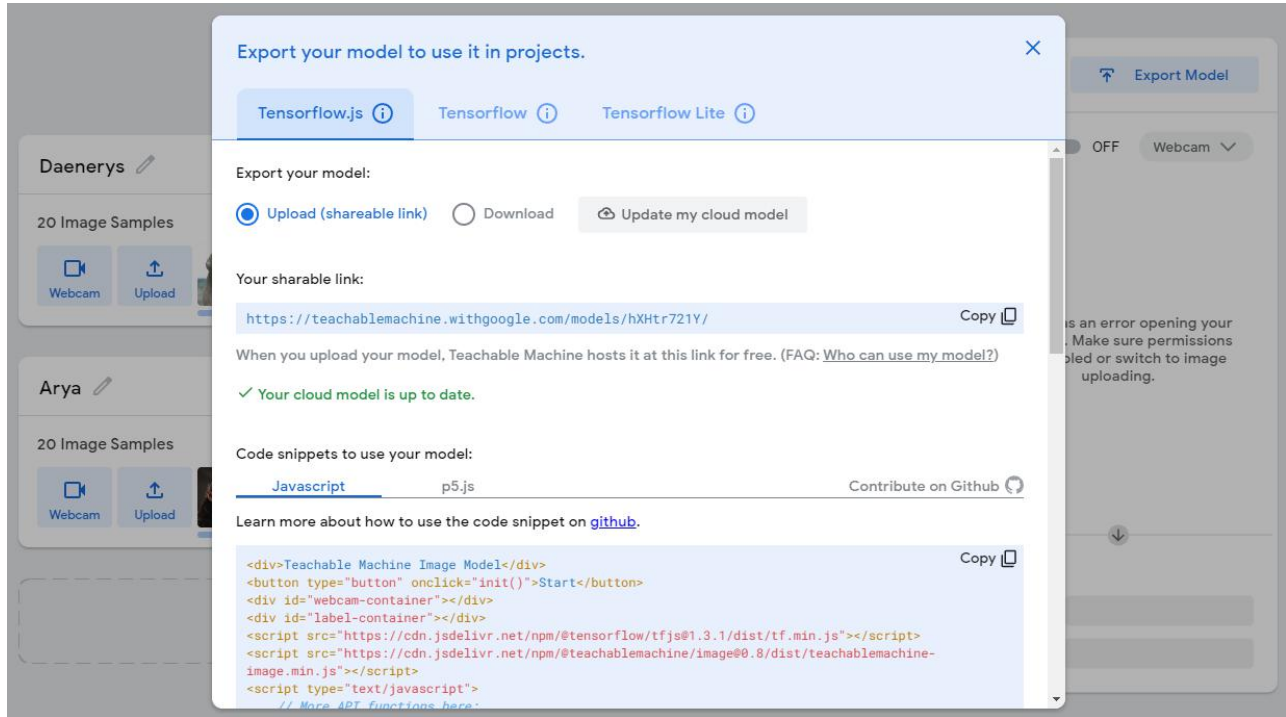
Stap 8: Train uw model. Wissel niet van browsertabblad tijdens het trainingsproces.



Stap 9: Exporteer uw model. Kies in het pop-upvenster om het naar de cloud te uploaden (derde optie) en Google zal uw gegevens gratis hosten.



Stap 10: Kopieer de link in het onderstaande tekstveld - dit is de URL van uw model.
In dit geval was dat zo <https://teachablemachine.withgoogle.com/models/hXHtr721Y/>



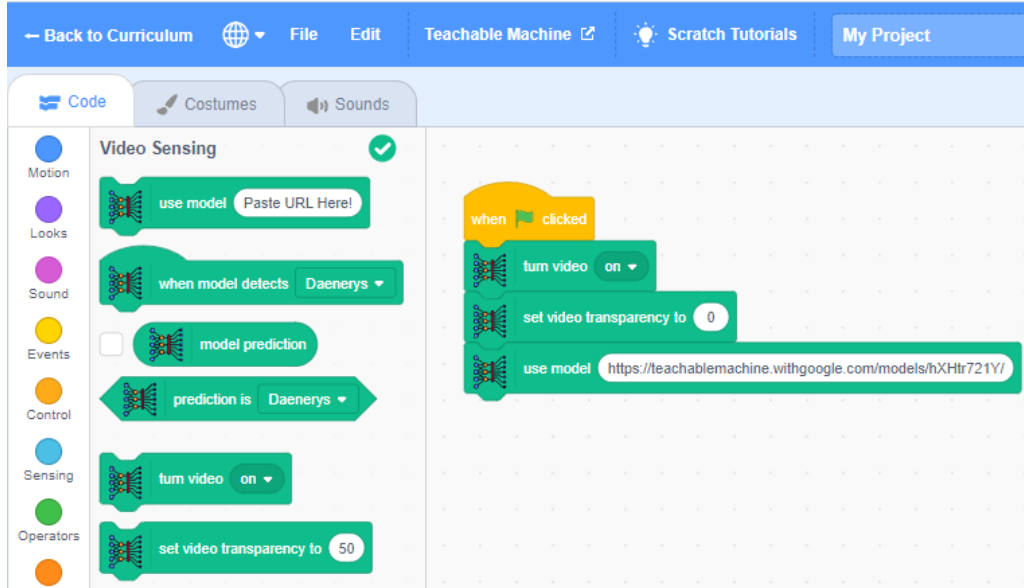
Stap 11: Uw model is klaar voor gebruik

Stap 12: Open Scratch GUI op: <https://mitmedialab.github.io/prg-extension-boilerplate/create/> en laad Teachable Machine-extensie.





Stap 13: Eerst moeten we video van de camera inschakelen en de transparantie instellen op 0 (niet-transparant). Vervolgens gebruiken we het model-URL-blok en plakken onze modellink daar.

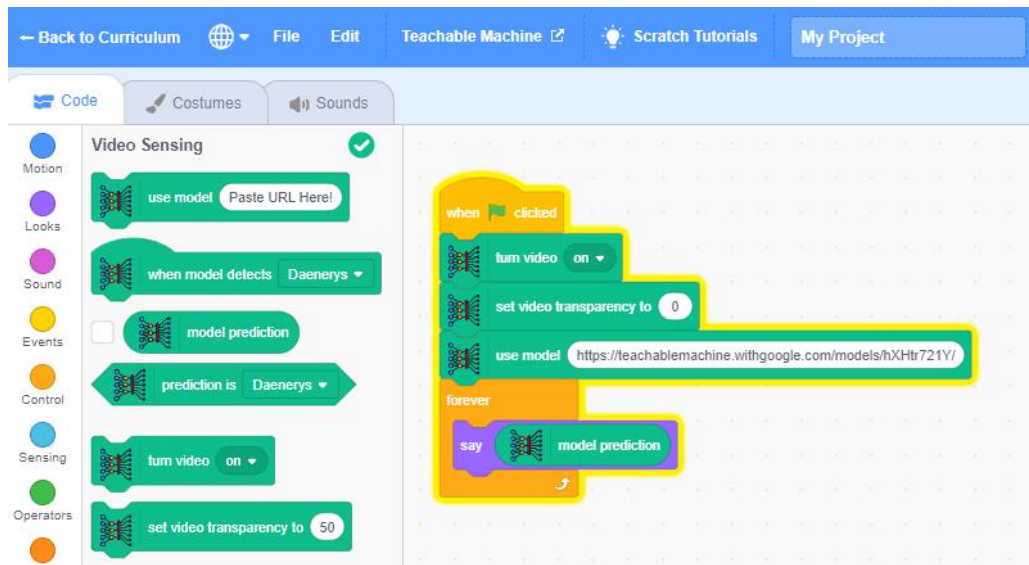


Stap 14: Laatste paar blokken zijn lus (voor altijd) en zeg blok dat een blok van het type reporter bevat met het voorspellingsresultaat. Ik ben er vrij zeker van dat je de echte Daenerys of Arya niet voor je camera kunt krijgen, dus gebruik je smartphone met hun afbeeldingen en richt deze op de camera om de resultaten te zien. Je kunt het model trainen met je eigen afbeeldingen of afbeeldingen van je vrienden.

*** Zorg ervoor dat u niemands foto maakt zonder hun toestemming.



http://erasmus-artie.eu





PICTOBLOX (Desktop applicatie):

Stap 1a: Aangezien er geen online GUI beschikbaar is, moet u PictoBlox installeren vanaf:<https://thestempedia.com/product/pictoblox/download-pictoblox/> (427 Mb)

Stap 2a: Open PictoBlox en kies Gezichtsdetectie-uitbreiding



Stap 3a: Open uw webbrowser en download alle 20 afbeeldingen van: <https://bit.ly/daenerys-data>

Het zal worden gebruikt om klasse 1 te trainen

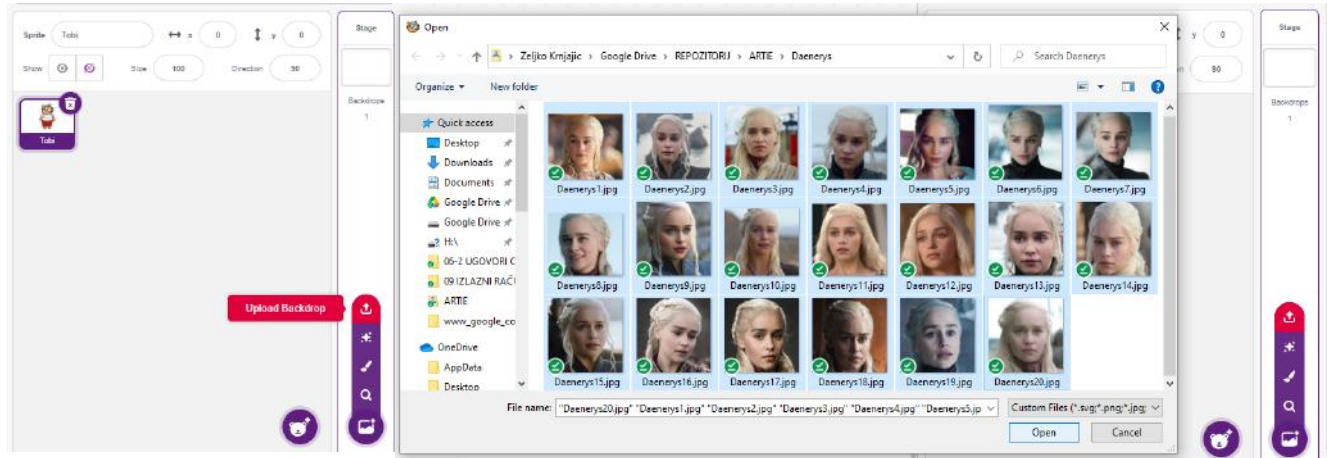
Stap 4a: Open uw webbrowser en download alle 20 afbeeldingen van:<https://bit.ly/arya-data>Het zal gebruikt worden om klasse 2 te trainen

Stap 5a: Verberg Tobi sprite van het podium door op het pictogram te klikken zoals op onderstaande afbeelding.

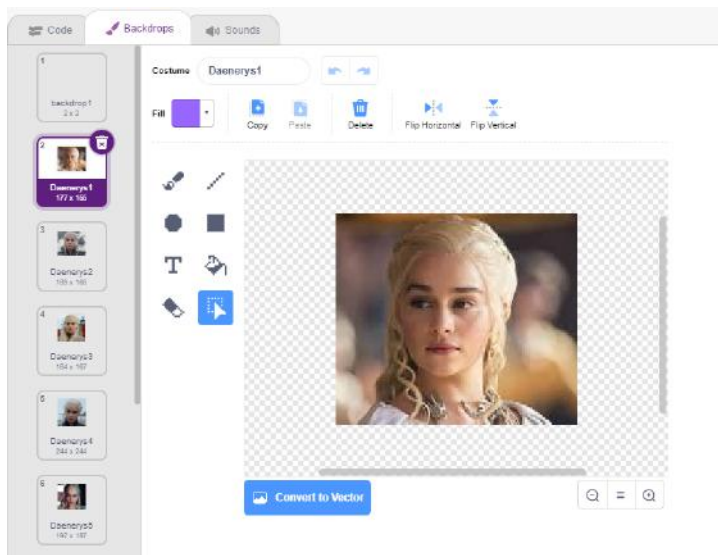




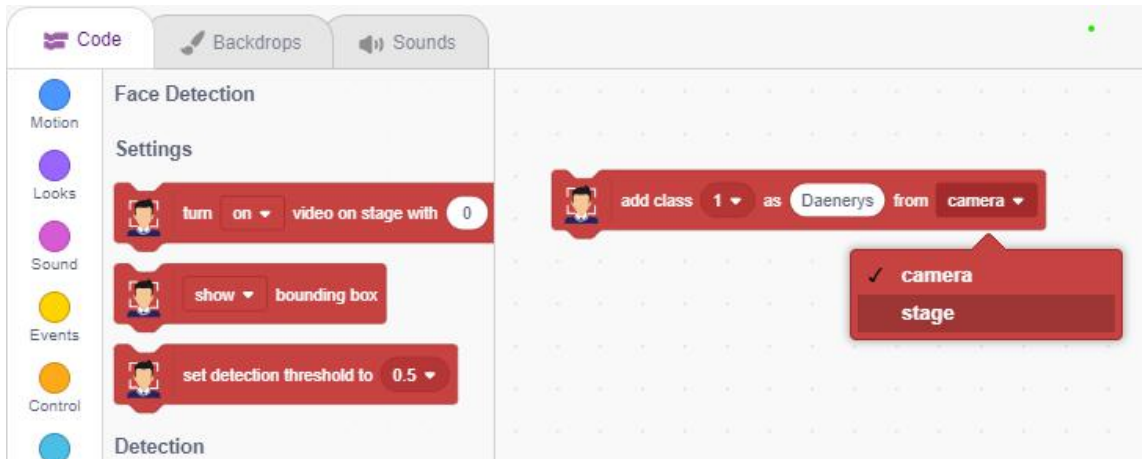
Stap 6a: Upload alle Daenerys-afbeeldingen naar de achtergrond (Upload achtergrond - Selecteer alle afbeeldingen - Openen)



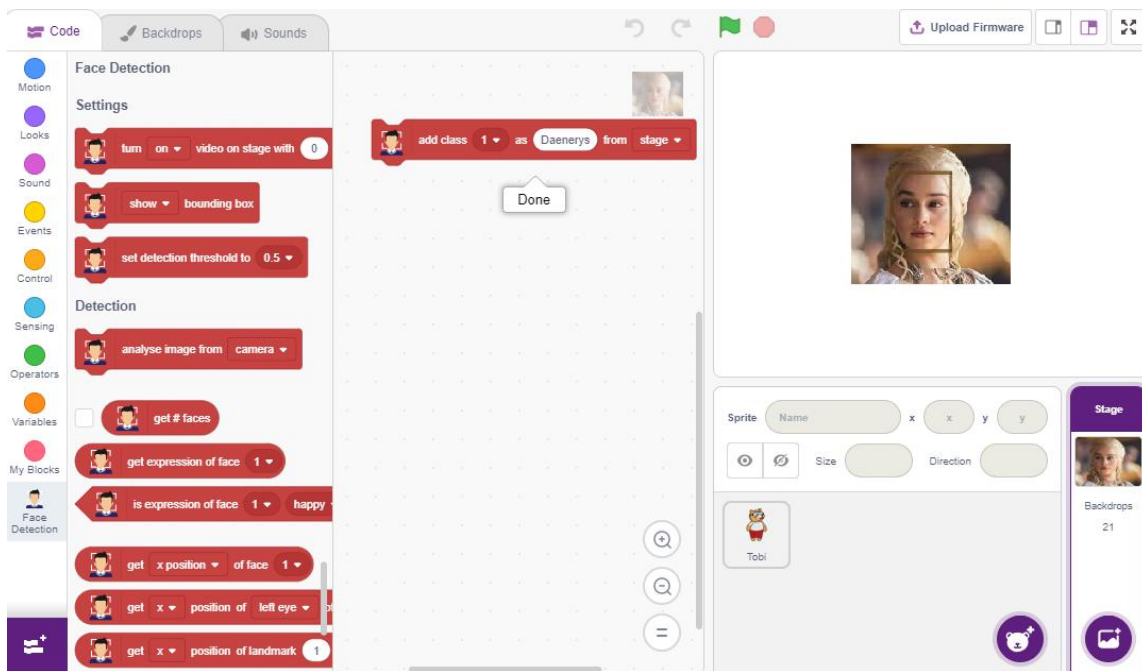
Stap 7a: Selecteer de Daenerys1-afbeelding en ga naar het tabblad Code



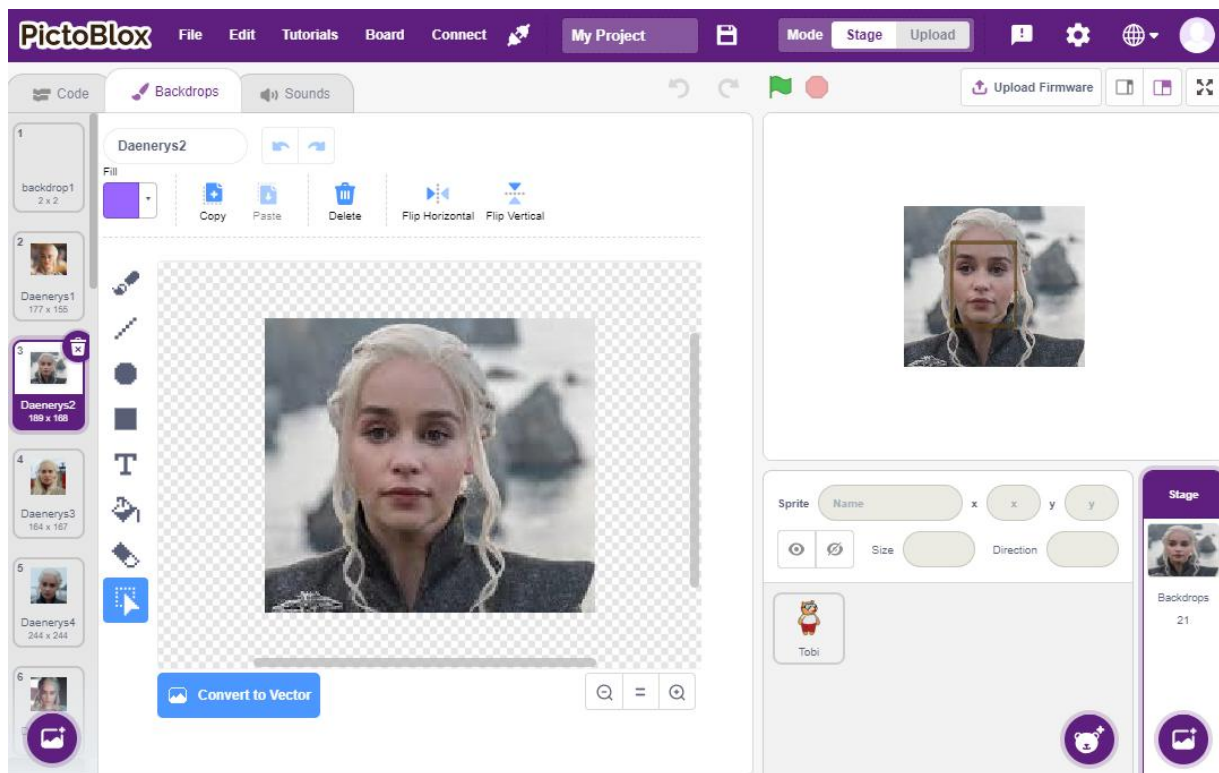
Stap 8a: Zoek in de groep Gezichtsdetectie naar klasseblok toevoegen, sleep het naar het codegebied en hernoem de naam van de klasse van Jarvis naar Daenerys en verander de bron van camera naar podium, zoals weergegeven in onderstaande afbeelding.



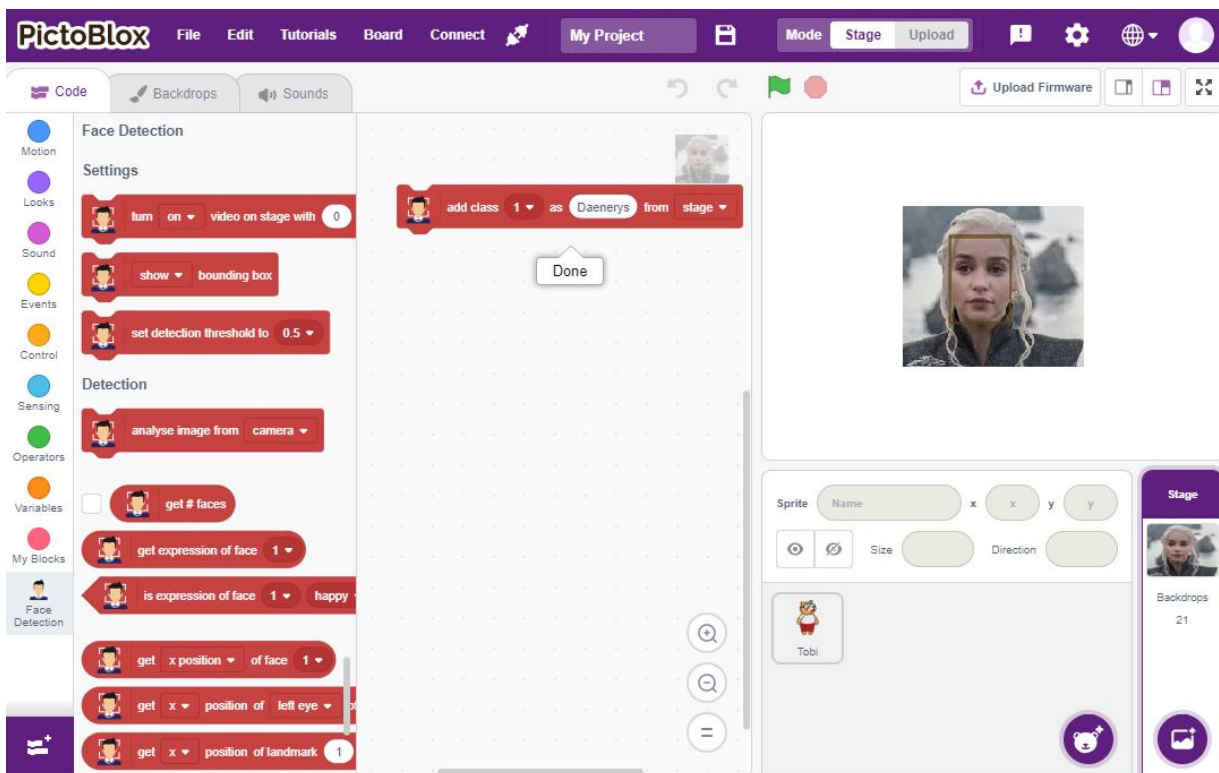
Stap 9a: Nadat de wijzigingen zijn aangebracht, klikt u gewoon op klasblok toevoegen om te beginnen met trainen. U ontvangt een bericht "Gereed" en ziet het omsluitende kader op het gezicht van Daenerys. De training met de 1e afbeelding is voltooid en u zult deze stappen voor elke andere afbeelding moeten herhalen.



Stap 10a: Schakel terug naar het tabblad Achtergronden en selecteer Daenerys 2-afbeelding.



Stap 11a: Schakel terug naar het tabblad Code en klik nogmaals op klasseblok toevoegen



f

<http://erasmus-artie.eu>

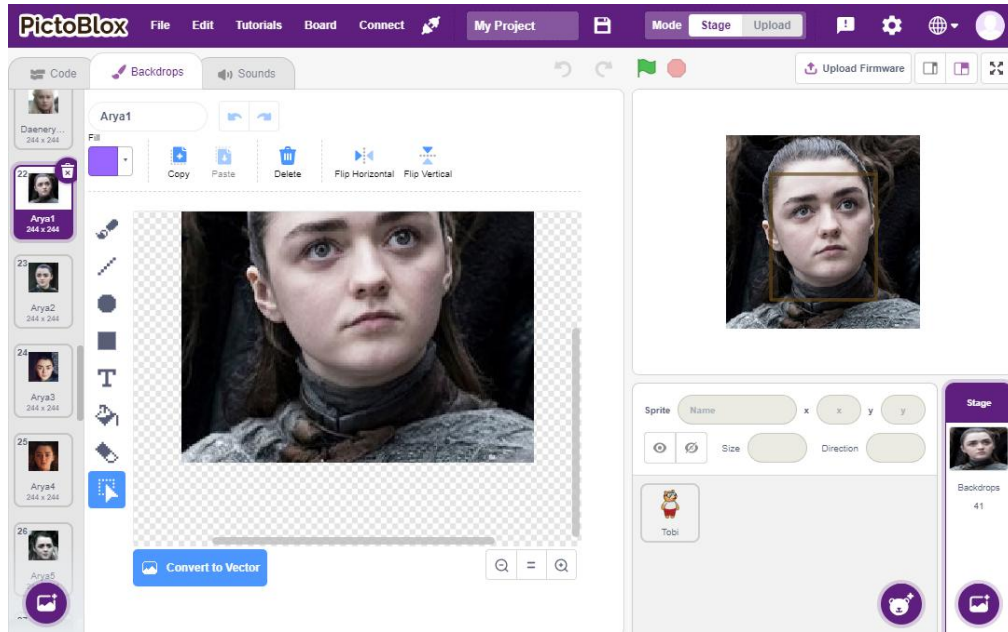
34



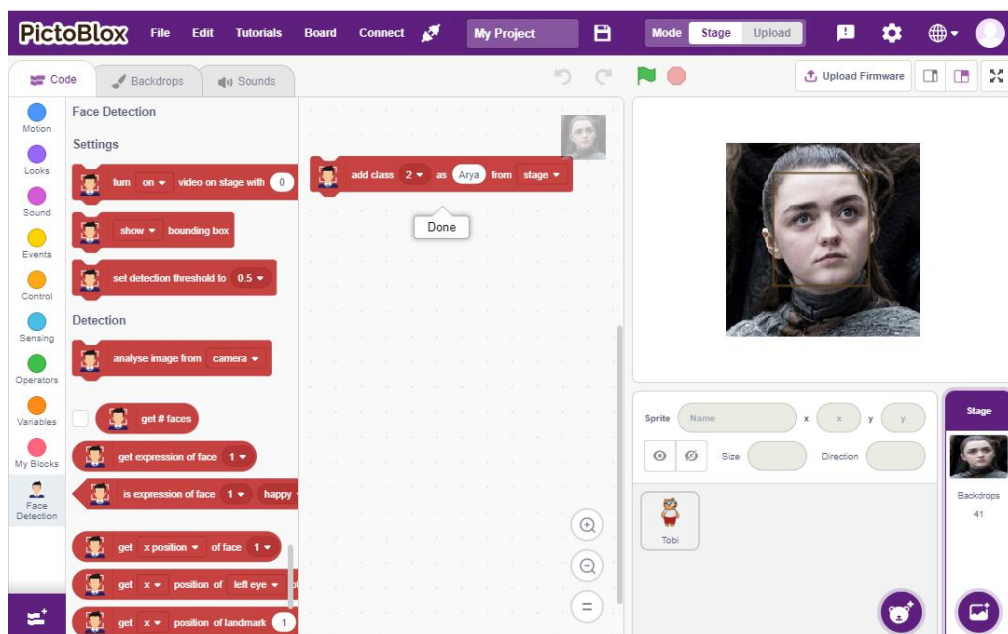
Stap 12a: Herhaal stap 10a en 11a voor elke afzonderlijke afbeelding (tot Daenerys20)

Stap 13a: Upload nu alle Arya-afbeeldingen naar de achtergrond op dezelfde manier als in stap 6a

Stap 14a: Selecteer Arya1 afbeelding



Stap 15a: Schakel terug naar het tabblad Code en verander (dit is belangrijk omdat we 2e klas trainen) de naam van de toegevoegde klasse van Daenerys naar Arya. Bron blijft hetzelfde - vanaf het podium. Klik daarna op dat blok om de 1e afbeelding van de Arya-klasse te trainen.





Stap 16a: Schakel over naar het tabblad Achtergronden en selecteer Arya2-afbeelding

Stap 17a: Ga naar het tabblad Code en klik op klasseblok toevoegen

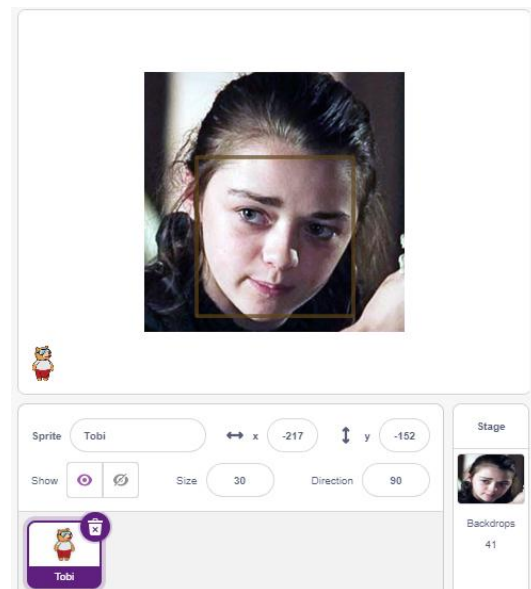
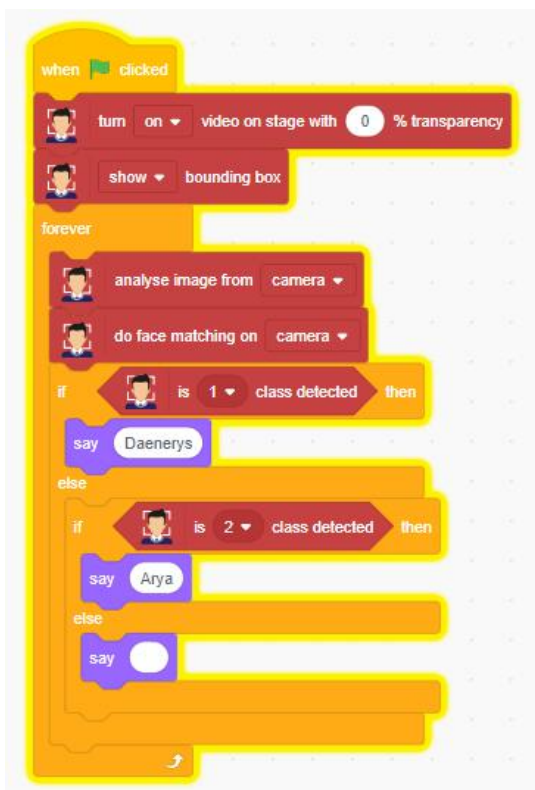
Stap 18a: Herhaal stap 16a en 17a voor elk afzonderlijk beeld (tot Arya20)

Stap 19a: Uw model is nu klaar om getest te worden. Selecteer Tobi sprite en maak het zichtbaar (show). Stel de Tobi-grootte in op 30% en verplaats deze van het midden naar de hoek.

Stap 20a: sluit je camera aan (als je er geen hebt) en begin met coderen. Schakel video in met 0% transparantie en toon het selectiekader. Het volgende blok is een eeuwige lus en Tobi zal het gezichtsherkenningresultaat tonen op basis van het gezichtsherkenningresultaat. Het is een dubbel if-else-blok en het laatste else-geval retourneert een lege string als er geen gezicht wordt herkend.

Stap 21a: Start het programma. Gebruik je smartphone met afbeeldingen van Daenerys of Arya en richt hem op de camera om de resultaten te zien. Je kunt het model trainen met je eigen afbeeldingen of afbeeldingen van je vrienden.

*** Zorg ervoor dat u niemands foto maakt zonder hun toestemming.





PROJECT MET GEZICHTSERKENNING

Hier is een klein project dat in feite een use-case is van gezichtsherkenning. We hebben een getraind model van 8 vrouwelijke beroemdheden voor je gemaakt in Teachable Machine om het te vergelijken met je gezicht. We hebben Adriana Lima, Emilia Clarke, Gal Gadot, Natalie Portman, Selena Gomez, Emma Stone, Zoe Saldana, Maisie Williams. Je zult moeten uitzoeken welk gezicht uit de lijst de meeste overeenkomsten met jou heeft.

Stap 1b: Open Scratch GUI op:

<https://mitmedialab.github.io/prg-extension-boilerplate/create/> en laad Teachable Machine-extensie



Stap 2b: Eerst stellen we de grootte en positie van de sprite in en schakelen vervolgens video van de camera in en stellen de transparantie in op 0 (niet-transparant). Vervolgens gebruiken we het model-URL-blok en plakken we deze modellink:<https://teachablemachine.withgoogle.com/models/smuBDQTuY/>





Stap 3b: Vervolgens gebruiken we een gebeurtenistypeblok dat wordt geactiveerd door face-matching. En voeg gewoon een zegblok "Je ziet eruit als ..." eraan toe.

```
when clicked
  set size to 50 %
  go to x: -200 y: -120
  turn video on
  set video transparency to 0
  use model https://teachablemachine.withgoogle.com/models/smuBDQTuY/

when model detects Adriana
  say You look like Adriana Lima
```

Stap 4b: Voeg deze combinatie van blokken toe voor elke beroemdheid in de lijst.

```
when clicked
  set size to 50 %
  go to x: -200 y: -120
  turn video on
  set video transparency to 0
  use model https://teachablemachine.withgoogle.com/models/smuBDQTuY/

when model detects Adriana
  say You look like Adriana Lima

when model detects Emilia
  say You look like Emilia Clarke

when model detects Gal
  say You look like Gal Gadot

when model detects Natalie
  say You look like Natalie Portman

when model detects Selena
  say You look like Selena Gomez

when model detects Emma
  say You look like Emma Stone

when model detects Zoe
  say You look like Zoe Saldana

when model detects Maisie
  say You look like Maisie Williams
```





Stap 5b: Sluit uw webcamera aan of schakel deze in, start de code en kijk wie uw match is!

Stap 6b: Wilt u het huidige model wijzigen met uw eigen model? Er is een geweldige dataset met afbeeldingen van beroemdheden op: <https://www.kaggle.com/hereisburak/pins-face-recognition>

Mogelijk wordt u gevraagd om u bij Kaggle te registreren voordat u het kunt downloaden. Of u kunt afbeeldingen handmatig verzamelen via Google Zoeken. Nadat je alle gewenste afbeeldingen hebt verzameld, gebruik je de Teachable-machine op: <https://teachablemachine.withgoogle.com/> om uw model te trainen zoals u deed in Gezichtsherkenning programmeren in Scratch-scenario.

Gezichtsherkenning is een technologie waarmee een onderwerp kan worden geïdentificeerd of geverifieerd door middel van een afbeelding, video of een ander audiovisueel element van zijn gezicht. Over het algemeen wordt deze identificatie gebruikt om toegang te krijgen tot een applicatie, systeem of service. Voordat we beginnen met het programmeren van gezichtsherkenning, moeten we de foto's van een specifiek gezicht verzamelen en het model trainen. We hebben een applicatie gebruikt genaamd Teachable Machine. Teachable Machine is een webgebaseerde tool die het maken van machine learning-modellen snel, gemakkelijk en voor iedereen toegankelijk maakt.

Today, many consumers worldwide regularly interact with facial recognition technology.



Belangrijkste functionaliteiten van gezichtsherkenning:

Identiteit verificatie:

Identificeer personen en pas gespecificeerde regels toe op basis van de categorie waarin ze vallen, bijvoorbeeld VIP, geregistreerde bezoeker, geblokkeerde lijst, werknemer of student. Gebruik de informatie om processen te verbeteren en te automatiseren, zoals (1) toegangscontrole, (2) beveiligingsbescherming, (3) begroetingen van klanten of bezoekers en (4) tijdsklokken voor werknemers.

eKYC en preventie van spoofing:

Valideer de identiteit van een persoon met behulp van een live foto- of video-opname met een gescand (en geverifieerd) identiteitsbewijs. Dit wordt eKYC (electronic Know Your Customer) genoemd en wordt veel gebruikt in BFSI of soortgelijke gevallen.

Autorisatie:

Identificeer of een persoon zich in een vooraf geautoriseerde database bevindt om (1) geld op te nemen bij een geldautomaat, (2) toegang te krijgen tot een medische kast met beveiligde medicijnen, of (3) dure machines te ontgrendelen waarvoor goed opgeleide operators nodig zijn.

Klantsegmentatie en analyse:

Analyseer voor slimme advertenties de kenmerken van een persoon die voor een digitaal bord staat, zoals geslacht, leeftijd en emotie.



Gezondheidsmaatregelen:

Bevestig dat een persoon correct een masker draagt en verifieer of hij geen koorts heeft voordat hij toegang verleent tot een gebouw of restaurant.

Als we naar verticale markten kijken, springen 10 industrieën eruit als rijp voor de integratie van gezichtsherkenning en in veel gevallen omarmen ze het al:

1. Productie en opslag
2. Bankwezen, financiële diensten en verzekeringen (BFSI)
3. Slimme kantoren
4. Slimme huizen en wooncomplexen
5. Detailhandel
6. Openbaar vervoer en luchthavens
7. Zorginstellingen
8. Scholen en universiteiten
9. Gastvrijheid
10. Restaurants en bars

CONCLUSIE

Gezichtsherkenning wordt gebruikt in verschillende geavanceerde systemen, en het is mogelijk om eenvoudige voorbeelden te maken in tools als Scratch en PictoBlox.



Think!



What do I know now?



What did I learn?



What do I want to know?



Objectdetectie en classificatie voor beginners in Scratch

Objectdetectie en -classificatie hebben de afgelopen decennia veel aandacht gekregen. Op het gebied van computervisie heeft een van de meest voorkomende vragen te maken met het verschil tussen beeldclassificatie, objectdetectie en beeldsegmentatie.

Hoofddeee: een inleiding tot objectdetectie en -classificatie voor beginners door middel van praktische toepassingsvoorbeelden.

Laten we beginnen met te begrijpen wat beeldclassificatie is: Beschouw de afbeelding hieronder:





Je herkent het meteen. Het is een hond. Doe een stapje terug en analyseer hoe je tot deze conclusie bent gekomen. U kreeg een afbeelding te zien en u classificeerde de klasse waartoe deze behoorde (in dit geval een hond). En dat is in een notendop waar beeldclassificatie om draait. Zoals je het zag, is er hier maar één object: een hond. We kunnen eenvoudig een afbeeldingsclassificatiemodel gebruiken en voorspellen dat er een hond in de gegeven afbeelding staat. Maar wat als we zowel een kat als een hond in één afbeelding hebben



In dat geval kunnen we een multi-label classifier trainen. Maar we zullen de locatie van geen van beide dieren/objecten in de afbeelding weten. Dat is waar beeldlokalisatie in beeld komt. Het helpt ons om de locatie van een enkel object in de gegeven afbeelding te identificeren. In het geval dat we meerdere objecten aanwezig hebben, vertrouwen we op het concept van objectdetectie. We kunnen de locatie samen met de klasse voor elk object voorspellen met behulp van OD.



<http://erasmus-artie.eu>

Classification



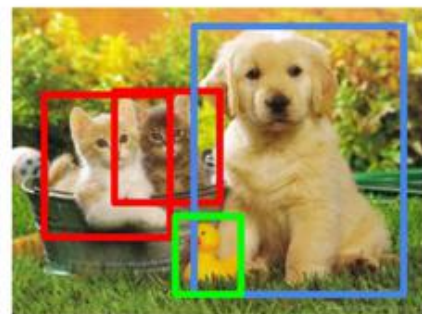
CAT

Classification + Localization



CAT

Object Detection



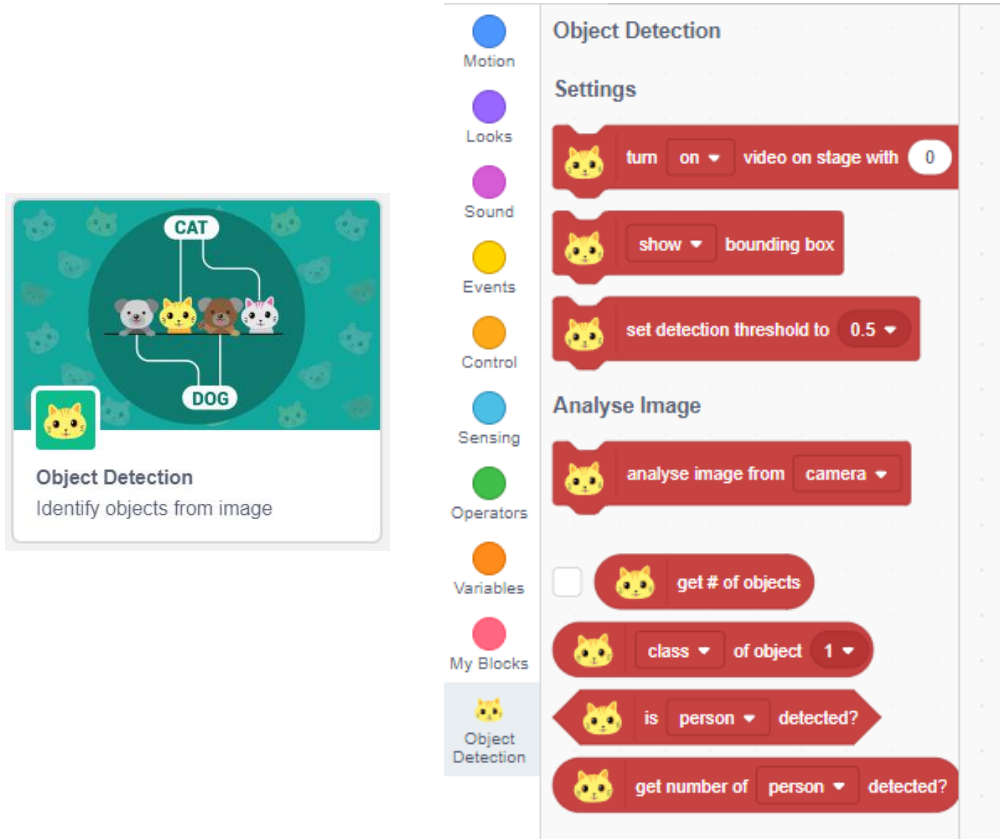
CAT, DOG, DUCK



Applicatie met objectdetectie

PictoBlox is momenteel de enige toepassing met objectdetectiemogelijkheden. Het is een applicatie van het desktoptype en u moet deze eerst installeren vanaf <https://thetempedia.com/product/pictoblox/download-pictoblox/> (427 Mb)

- Stap 1: Na installatie laadt u de Objectdetectie-extensie
- Stap 2: Dit zijn de blokken die beschikbaar zijn voor codering.

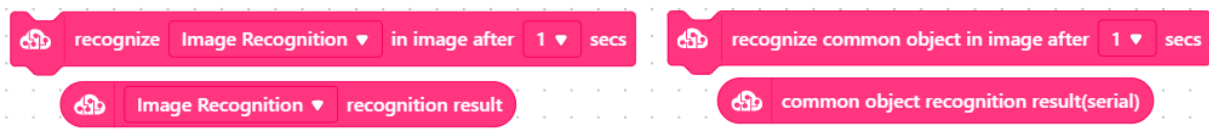


De eerste 3 blokken zijn instellingen voor camerastreams en de andere worden gebruikt voor analyse en rapportage.

Situatie met objectclassificatie is veel beter omdat we weinig applicaties meer hebben om te gebruiken (naast PictoBlox)

Makeblock (mBlock) - <https://ide.mblock.cc/>

Laad de Cognitive Services-extensie en u zult 4 blokken vinden die u kunt gebruiken voor objectherkenning (classificatie).





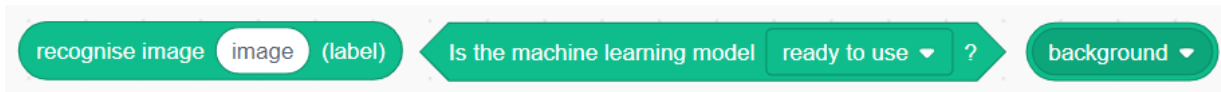
Makeblock biedt ook een leerbare machine-extensie (niet te verwarren met die van Google) waar je tot 3 klassen kunt trainen en deze kunt gebruiken voor objectclassificatie.

Scratch (ML4KIDS) - <https://machinelearningforkids.co.uk/scratch3/>

Laad de Imagenet-extensie.



Er zijn 3 blokken beschikbaar. Gebruik het in combinatie met de extensie Videodetectie om cameravideo aan/uit te zetten en transparantie in te stellen. Het is getraind om foto's van duizend gewone objecten te herkennen. Het machine learning-model is gebaseerd op MobileNet (een ML-model dat is ontworpen voor mobiele apparaten, dus er is niet veel rekenkracht voor nodig). Volledige lijst met objecten is hier beschikbaar: <https://storage.googleapis.com/download.tensorflow.org/data/ImageNetLabels.txt>



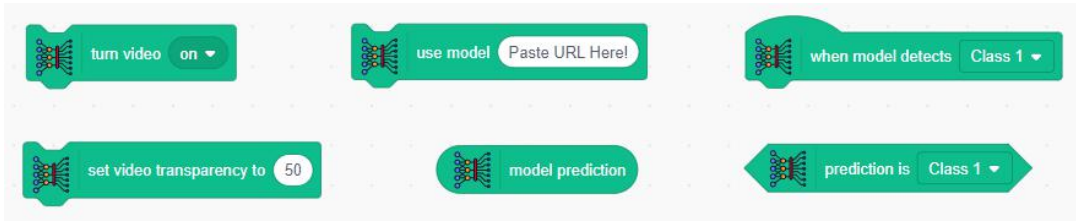
Scratch (MITMEDIALAB) - <https://mitmedialab.github.io/prg-extension-boilerplate/create/>

Laad leerbare machine-uitbreiding



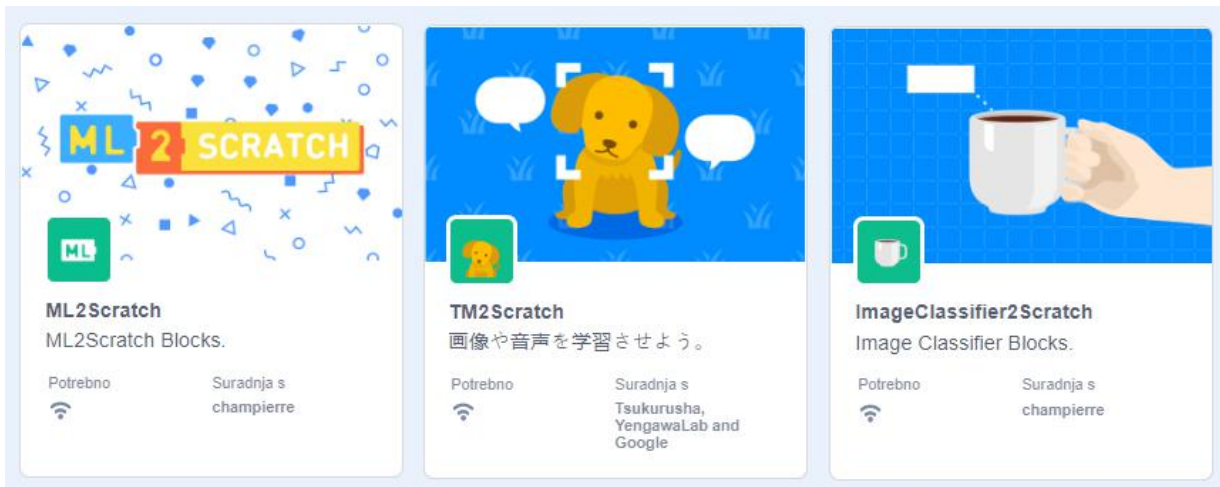


Gebruik deze blokken in combinatie met Google Teachable machine.



Stretch3 (github.io) - <https://stretch3.github.io/>

Laad ML2Scratch-, TM2Scratch- en ImageClassifier2Scratch-extensies om veel blokken te gebruiken voor objectclassificatie en training

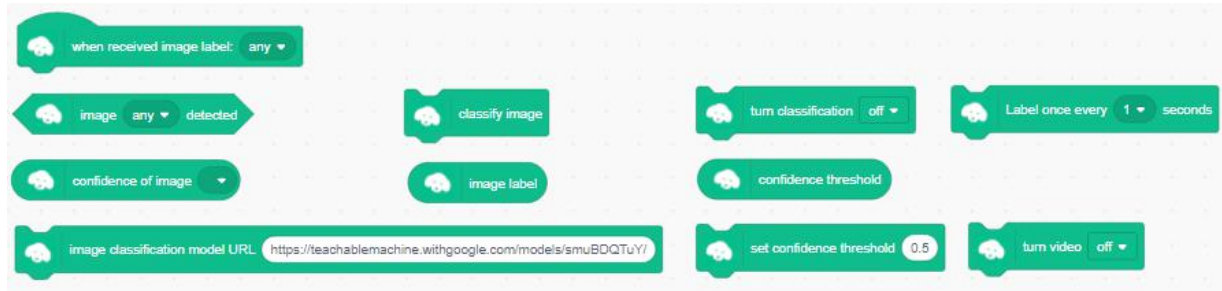


ML2Scratch-blokken

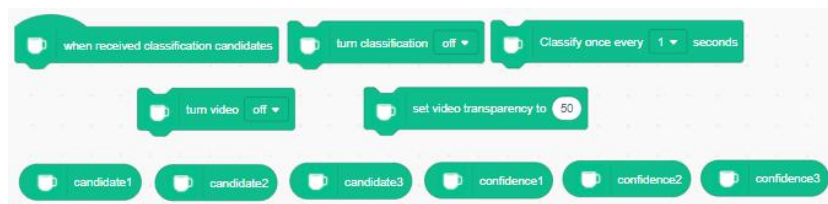




TM2Scratch-blokken



ImageClassifier2Kras blokken

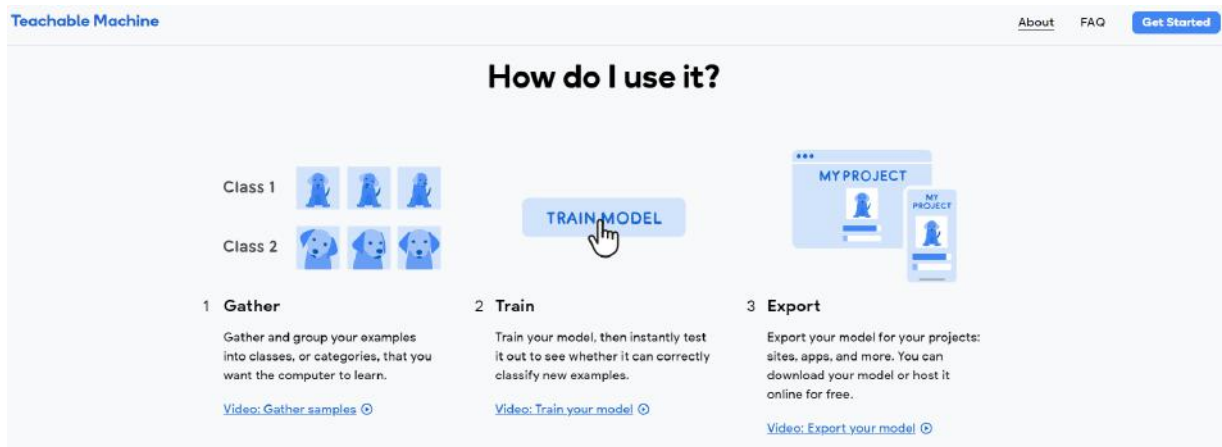


ImageClassifier2Kras blokken- <https://teachablemachine.withgoogle.com/>

Deze applicatie wordt gebruikt om uw model te trainen en te gebruiken voor objectherkenning in combinatie met Teachable machine-uitbreidingen beschikbaar in **PictoBlox**, **Scratch** (MITMEDIALAB) en **Stretch3**



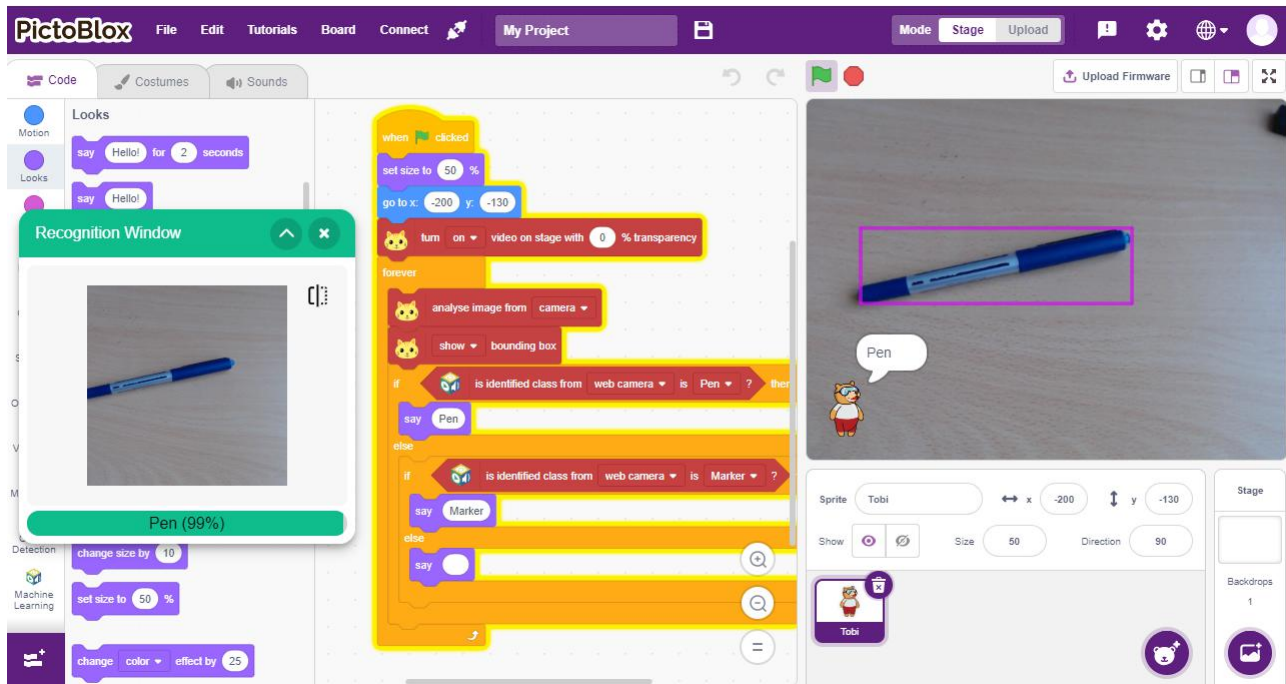
<http://erasmus-artie.eu>



Use case voor objectdetectie - pen of stift.

Leermachine wordt gebruikt voor het trainen van 2 lessen. Link van een getraind model is: <https://teachablemachine.withgoogle.com/models/FdWn0CA2a/>

Uitbreidingen die in PictoBlox worden gebruikt, zijn objectdetectie en machine learning.



Objectdetectie en -classificatie zijn twee belangrijke taken voor beeldbegrip. Het herkennen van objecten in een afbeelding vereist het combineren van veel verschillende signalen uit de onbewerkte afbeeldingsgegevens. Er worden vaak twee soorten informatie gebruikt: de lokale weergave die het object zelf beschrijft en de globale representatie die de beeldspecifieke informatie vastlegt. Deze twee soorten informatie worden vaak gebruikt in twee taken: objectdetectie en classificatie.

Classificatie

Classificatie is een machine learning-taak om te bepalen welke objecten in een afbeelding of video voorkomen. Het verwijst naar het trainen van machine learning-modellen om te herkennen welke klassen (objecten) aanwezig zijn. Classificatie is nuttig op het ja-nee-niveau om te bepalen of een afbeelding een object/afwijking bevat of niet.

Een andere taak dan classificatie is lokalisatie of het bepalen van de positie van de geclassificeerde objecten in de afbeelding of video.





Objectdetectie

Objectdetectie combineert classificatie en lokalisatie om te bepalen welke objecten zich in de afbeelding of video bevinden en om te specificeren waar ze zich in de afbeelding bevinden. Het past classificatie toe op afzonderlijke objecten en gebruikt begrenzingsvakken. Objectdetectie is handig bij het identificeren van objecten in een afbeelding of video. Use cases voor objectdetectie omvatten gezichtsdetectie met elke analyse na detectie, bijvoorbeeld expressiedetectie, leeftijdsschatting of slaperigheidsdetectie. Er zijn veel real-time objectdetectietoepassingen voor verkeersmanagement, zoals voertuigdetectiesystemen op basis van verkeersscènes. Zoals hierboven beschreven, zijn de meest populaire benaderingen van computervisie classificatie en objectdetectie om objecten in een afbeelding te identificeren en hun positie te specificeren.

CONCLUSIE

Objectdetectie en -classificatie zijn twee belangrijke taken voor beeldanalyse.

Think!



What do I know now?



What did I learn?



What do I want to know?



<http://erasmus-artie.eu>



Objectdetectie programmeren in Kras

Laten we, om objectdetectie te begrijpen, eens kijken wat we tot nu toe hebben geleerd.

Wat is objectdetectie?

Hoe werkt objectdetectie?

Objectdetectie is een computervisietechniek die tot doel heeft objecten in een afbeelding of video te identificeren en te lokaliseren. In het bijzonder tekent objectdetectie begrenzingskaders rond deze gedetecteerde objecten, waardoor we kunnen lokaliseren waar de objecten zich in een bepaalde scène bevinden (of hoe ze zich erdoorheen bewegen). Voordat we beginnen met het programmeren van objectdetectie, moeten we de foto's van specifieke gezichten verzamelen en het model trainen. We zullen een applicatie gebruiken genaamd Teachable Machine. Teachable Machine is een webgebaseerde tool die het maken van machine learning-modellen snel, gemakkelijk en voor iedereen toegankelijk maakt

Hoofdidee: objectdetectiesoftware en het gebruik ervan begrijpen aan de hand van voorbeelden.



TAAK: Is het een kat of een hond?

Maak een model en een programma dat detecteert of er een kat of een hond in de camerastream zit

WEB-GEBASEERDE APPLICATIE (geen software-installatie)

Stap 1: Open uw webbrowser, selecteer en download afbeeldingen van: <https://bit.ly/cats-image-dataset>

Het zal worden gebruikt om klasse 1 te trainen

Stap 2: Open uw webbrowser, selecteer en download afbeeldingen van: <https://bit.ly/dogs-image-dataset>

Het zal gebruikt worden om klasse 2 te trainen

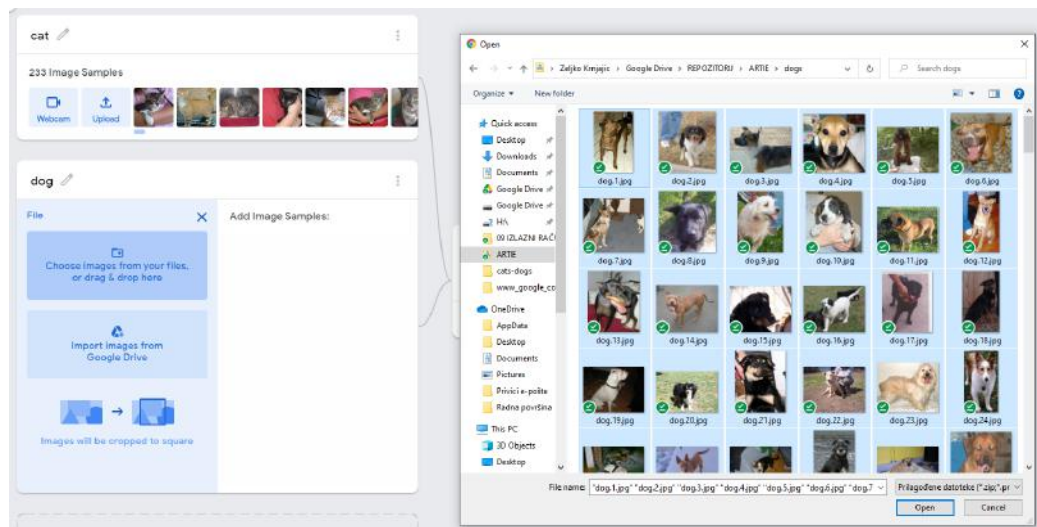
Stap 3: Open uw webbrowser en ga naar: <https://teachablemachine.withgoogle.com/>

Stap 4: Klik op Aan de slag.

Stap 5: Kies het afbeeldingsproject

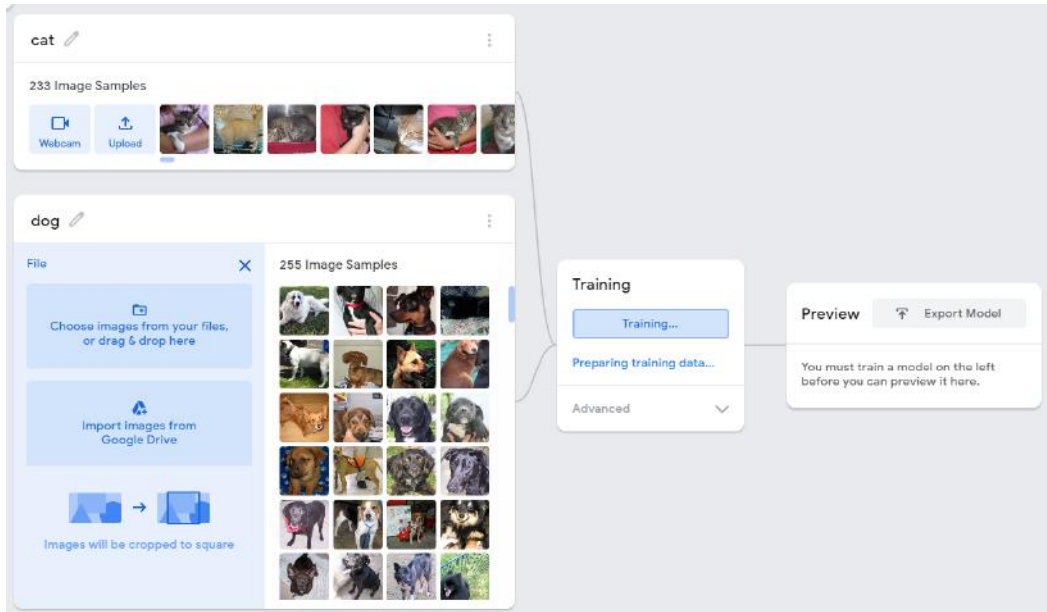
Stap 6: Kies het standaard afbeeldingsmodel

Stap 7: Wijzig de naam van klasse 1 in kat en klasse 2 in hond. Upload afbeeldingen van katten naar kattenbestanden en afbeeldingen van honden naar hondenbestanden, zoals weergegeven in de onderstaande afbeelding

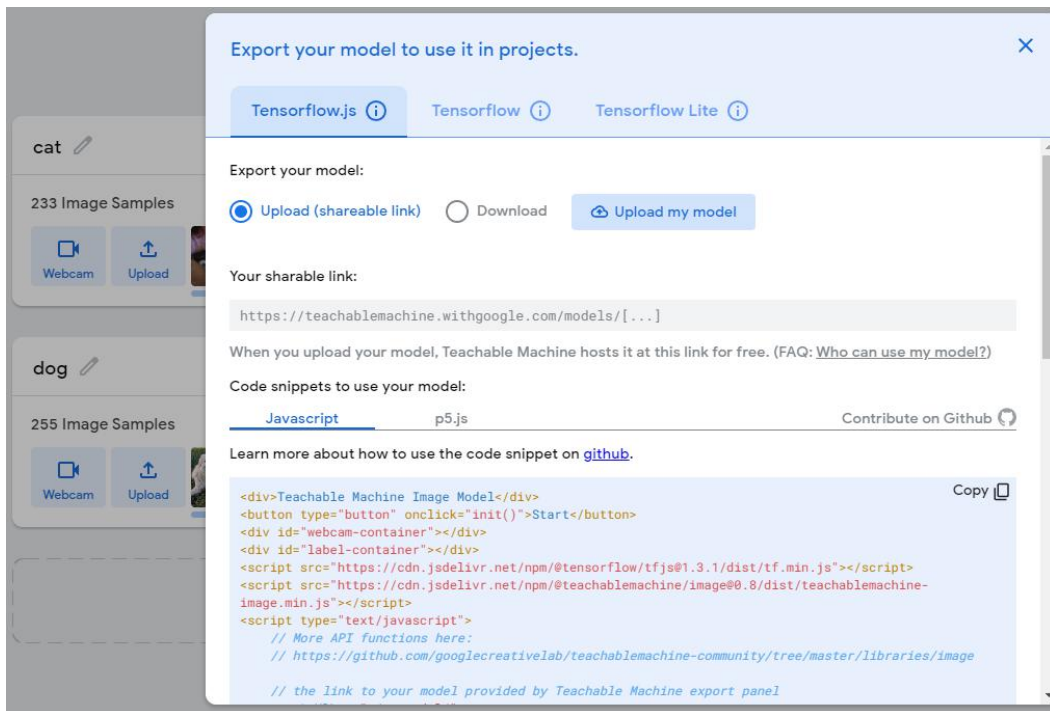




Stap 8: Train uw model. Wissel niet van browsertabblad tijdens het trainingsproces.



Stap 9: Exporteer uw model. Kies in het pop-upvenster om het naar de cloud te uploaden (derde optie - Upload mijn model) en Google zal uw gegevens gratis hosten.





Stap 10: Kopieer de link in het onderstaande tekstveld - dit is de URL van uw model.
In mijn geval was dat zo <https://teachablemachine.withgoogle.com/models/gS4NT1mgE/>

Upload (shareable link)
 Download

Your sharable link:

<https://teachablemachine.withgoogle.com/models/gS4NT1mgE/> Copy

When you upload your model, Teachable Machine hosts it at this link for free. (FAQ: [Who can use my model?](#))

✓ Your cloud model is up to date.

In mijn geval was dat zo

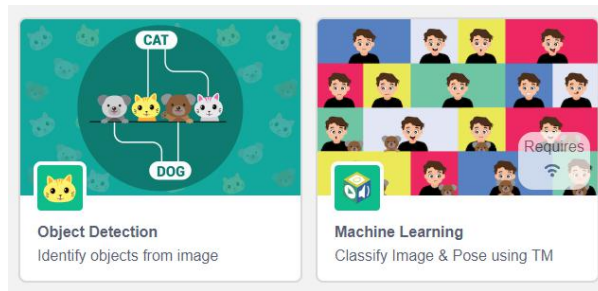
PICTOBLOX (Desktop applicatie):

Stap 1a: Download en installeer **PictoBlox**, aangezien dit momenteel de enige is met objectdetectiemogelijkheden. Het is een applicatie van het desктоptype en u moet deze eerst installeren vanaf <https://thetempedia.com/product/pictoblox/download-pictoblox/> (427 Mb)

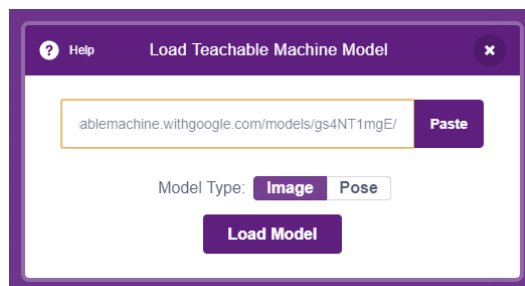
Stap 2a: Laad extensies Objectdetectie en Machine Learning.



http://erasmus-artie.eu

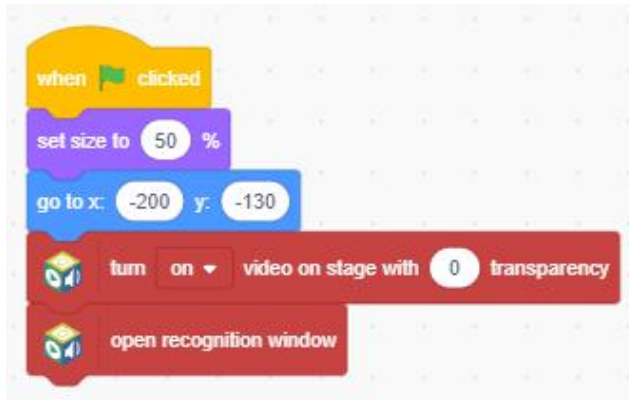


Stap 3a: Selecteer de groep Machine Learning en kies Load a Model. Plak de modellink van leerbare machine (hieronder): <https://teachablemachine.withgoogle.com/models/gS4NT1mgE/> en klik op Model laden





Stap 4a: We voeren tegelijkertijd een objectclassificatie en objectdetectie uit. Eerst moeten we het programma starten, het formaat wijzigen en de Tobi-sprite in de linker benedenhoek verplaatsen. De volgende paar blokken van de Machine Learning-extensie worden gebruikt om de video in te schakelen en het herkenningsvenster te openen. Gebruik gespiegelde video op het podium inschakelen met 0 transparantie als uw video is gespiegeld.

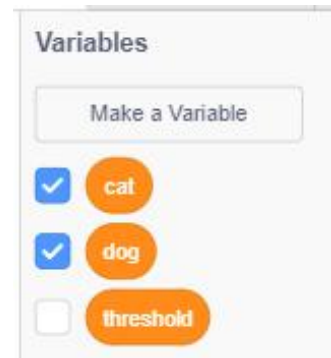
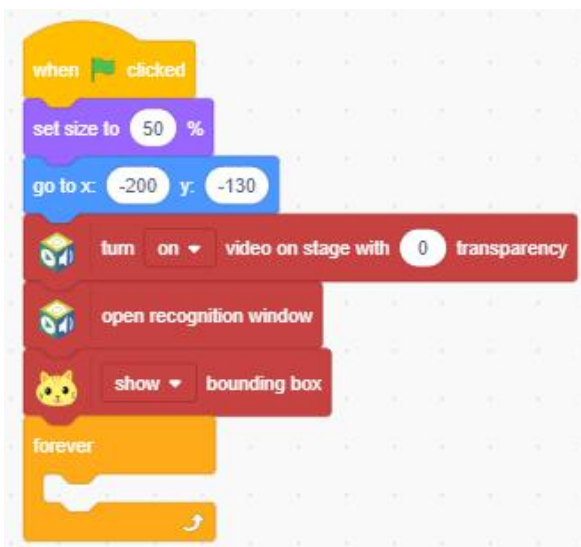


Stap 5a: Gebruik vanuit de extensie Object Detection show bounding box block om de positie van een object in de videostream weer te geven en voeg een forever loop-blok toe.

Stap 6a: Maak 3 variabelen **kat**, **hond** en **drempel**. Toon **kat** en **hond** op het podium door het aan te vinken. Laat de drempel ongecontroleerd.



<http://erasmus-artie.eu>



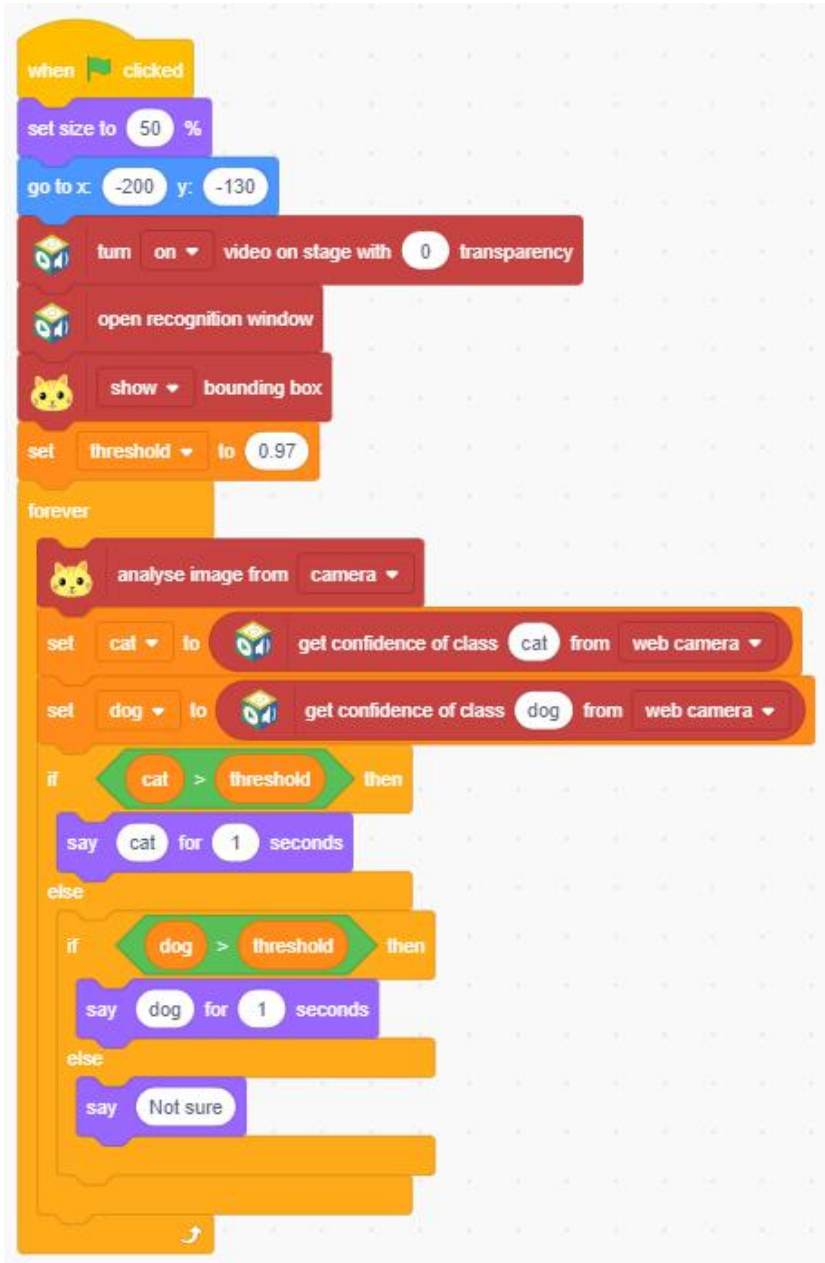


Stap 7a: Voeg eerst de ingestelde drempel toe aan het **blok 0,97** voor de forever-lus. En nu is hier het belangrijkste deel waarin we eerst het beeld **van de camera** analyseren en het vertrouwen van klassewaarden in variabelen opslaan



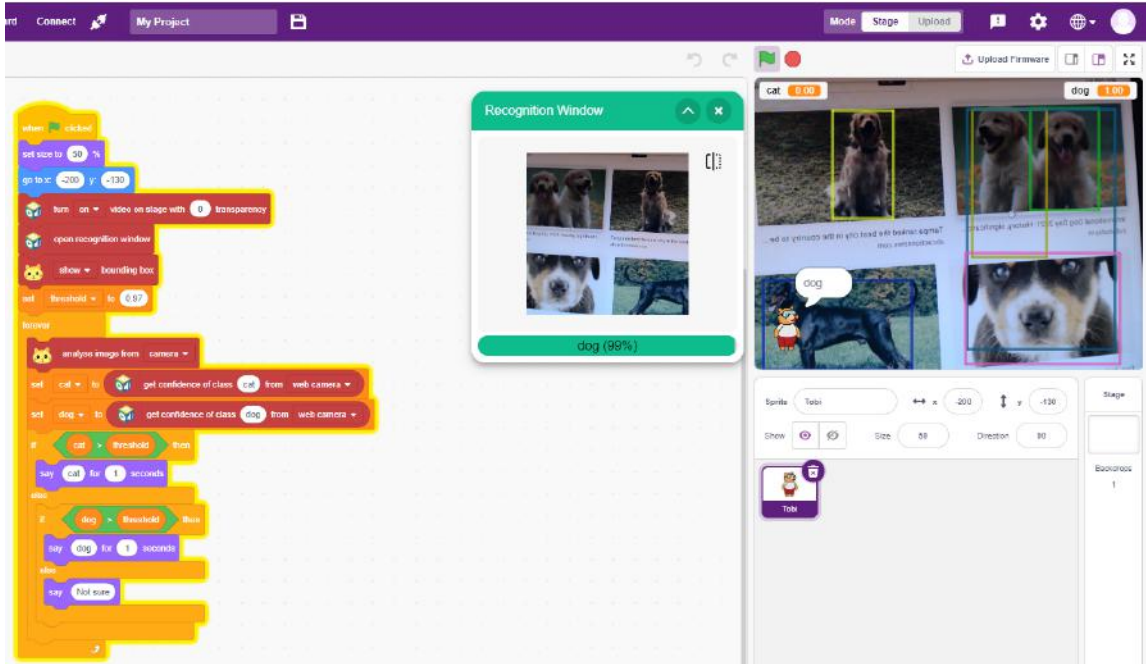
<http://erasmus-artie.eu>

Stap 8a: Het wordt gevolgd door een dubbel als-dan-anders-blok en 2 klassesdetectorblokken gevolgd door bijvoorbeeld "kat" of "hond". Als er niets wordt gedetecteerd, is de tekst in de tekstballon "Niet zeker".



<http://erasmus-artie.eu>

Stap 9a: Start het programma en test het op uw kat of hond. Als u er geen bij u in de buurt heeft, zijn Google-afbeeldingen hier, zoek gewoon naar honden- / kattenfoto's en richt uw webcam op het scherm met resultaten. U ziet de begrenzingsvakken rond gedetecteerde objecten. Probeer de drempelwaarde te verlagen of te verhogen of probeer het detectie-algoritme te verwarren met afbeeldingen van specifieke katten- of hondenrassen.



Dus wat is objectdetectie precies?

Laten we om die vraag te beantwoorden beginnen met beeldclassificatie. In deze taak hebben we een afbeelding en we willen deze toewijzen aan een van de vele verschillende categorieën (bijv. auto, hond, kat, mens,...), dus eigenlijk willen we de vraag "Wat staat er op deze afbeelding?". Merk op dat aan één afbeelding slechts één categorie is toegewezen. Simpel gezegd, objectdetectie is een type beeldclassificatietechniek en naast classificatie identificeert deze techniek ook de locatie van de objectinstanties uit een groot aantal vooraf gedefinieerde categorieën in natuurlijke afbeeldingen.



http://erasmus-artie.eu

CONCLUSIE

Objectdetectie is een beeldclassificatietechniek en naast classificatie identificeert deze techniek ook de positie van een gevonden object uit een groot aantal vooraf gedefinieerde categorieën.

Think!



What do I know now?



What did I learn?



What do I want to know?



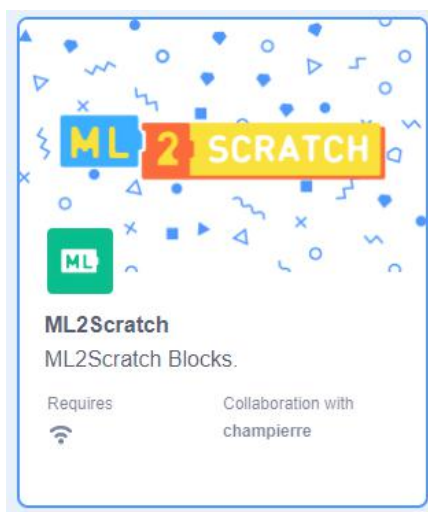
Objectclassificatieproject

*Wat is objectdetectie?
Hoe werkt objectdetectie?*

Hoofdidee: het proces van objectclassificatie en het gebruik ervan begrijpen door middel van een project met praktisch werk.

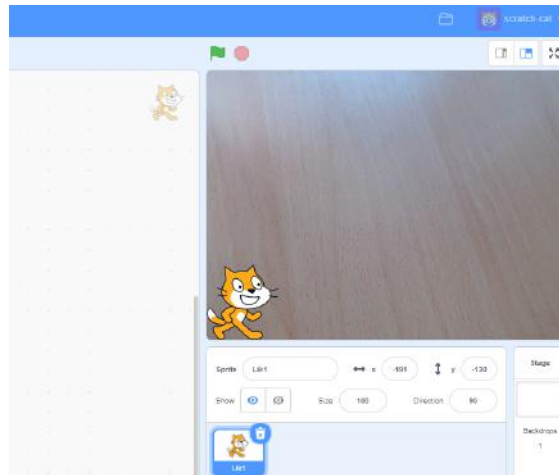
Stap 1: webcam aansluiten

Stap 2: Open Scratch op <https://stretch3.github.io/> en extensie toevoegen "ML2SCRATCH"

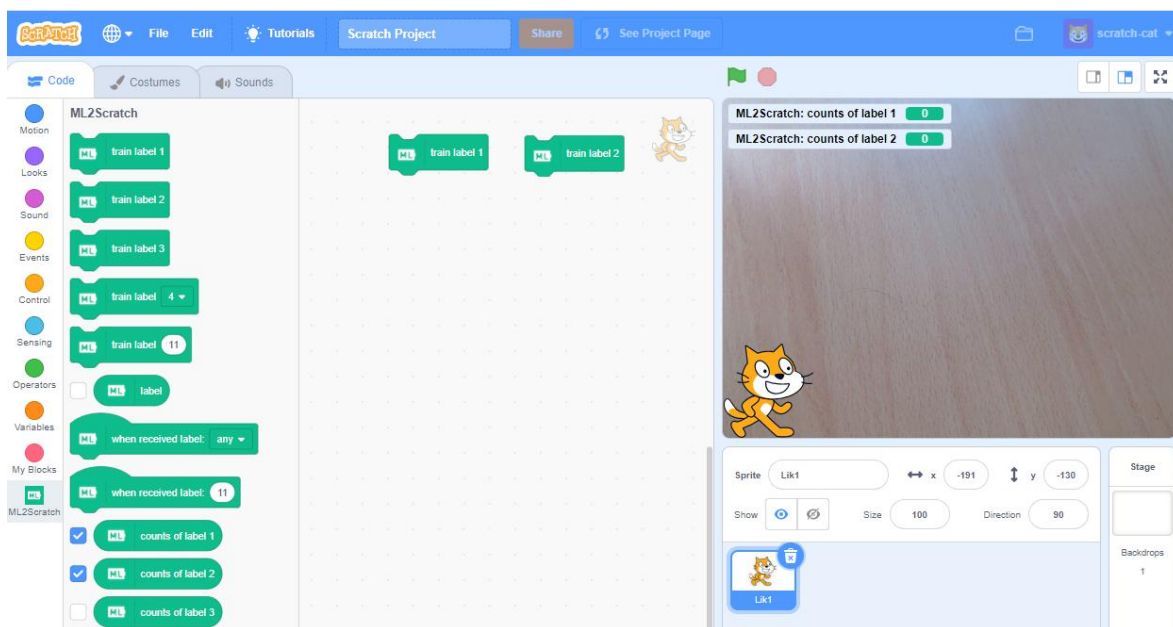




Stap 3: Ruim de bureauplug-in op en richt de webcam op een leeg gebied, verplaats een kattensprite naar de hoek zoals weergegeven



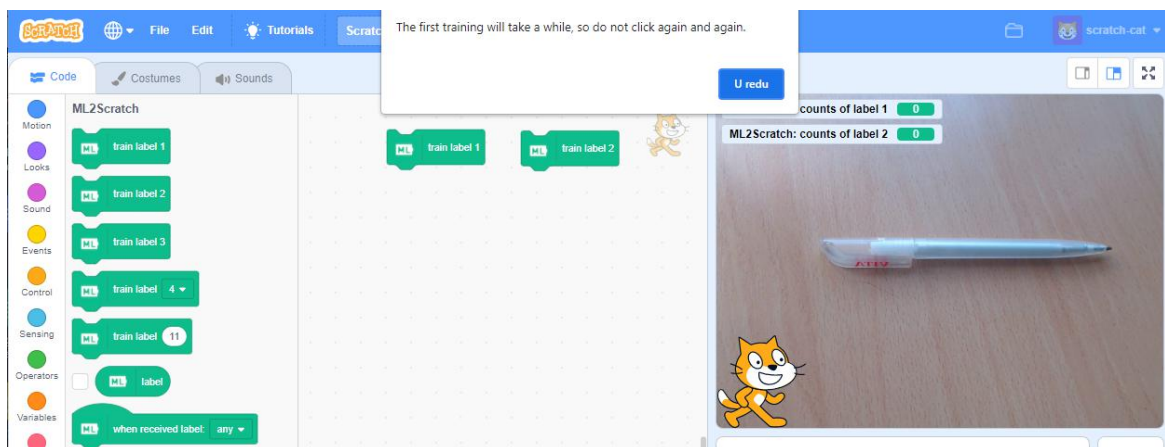
Stap 4: Kies uit de ML2SCRATCH-groep een **treinlabel 1** en **treinlabel 2**-blokken en plaats deze op een programmeergebied en controleer het **aantal labels 1** en het **aantal labels 2** zoals weergegeven



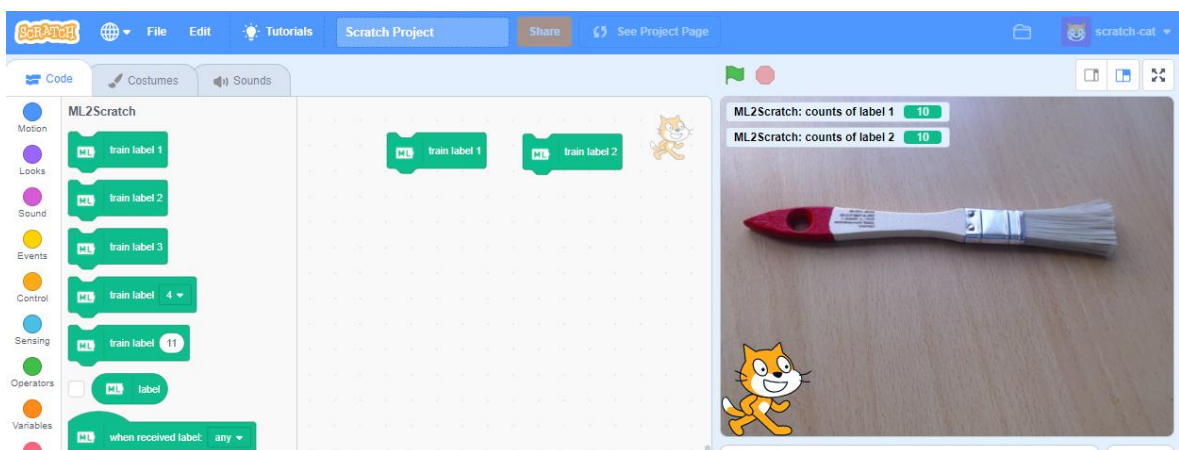


Stap 5: bereid twee groepen objecten voor op machine learning. In ons geval worden penselen en pennen gebruikt voor het trainen van de labels

Stap 6: Plaats een eerste object van de eerste groep in het gebied waarop de webcam is gericht en klik op treinlabel 1 blok - u krijgt een melding dat u even moet wachten, zoals weergegeven, en het aantal labels verandert in 1

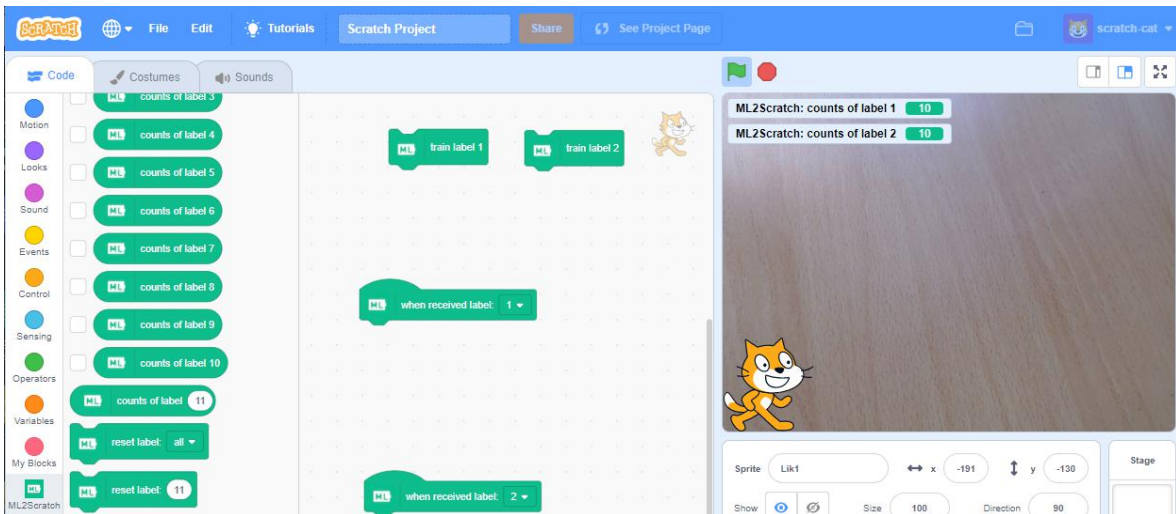


f Stap 7: Maak ongeveer 10 foto's van elk object en varieer een vorm en plaatsing. Vergeet niet het juiste label te trainen, meng objecten en labels niet door elkaar.

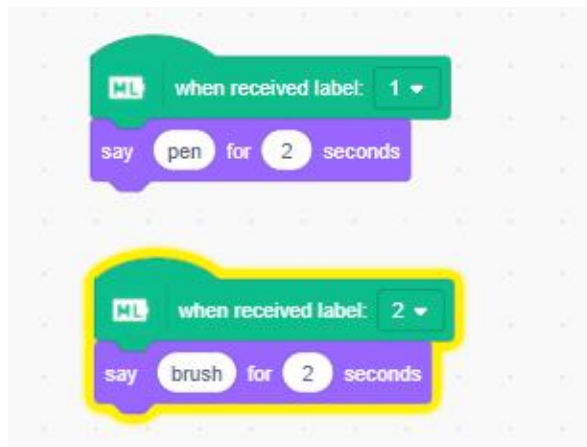




Stap 8: Kies nu twee blokken wanneer u een label ontvangt: elk en verander elk in 1 op het eerste blok en verander elk in 2 op het tweede blok, zoals weergegeven



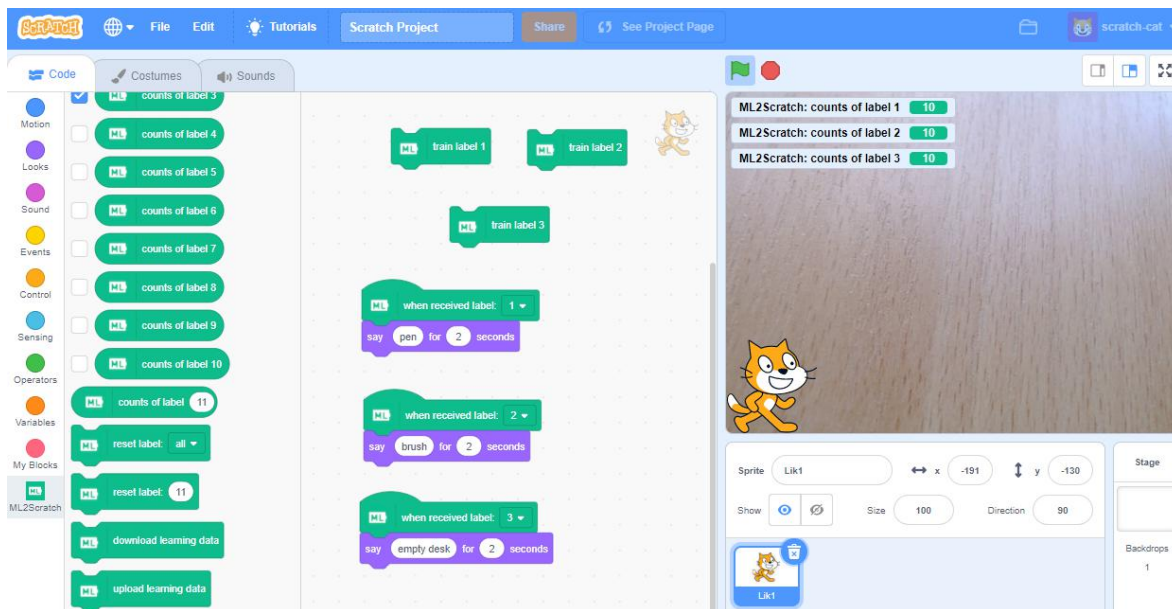
Stap 9: Kies er twee uit de Kijk-groep zeg hallo! blokken van 2 seconden en verander "Hallo!" om in het eerste blok te "pennen" en "Hallo!" om op het tweede blok te "poetsen" zoals afgebeeld





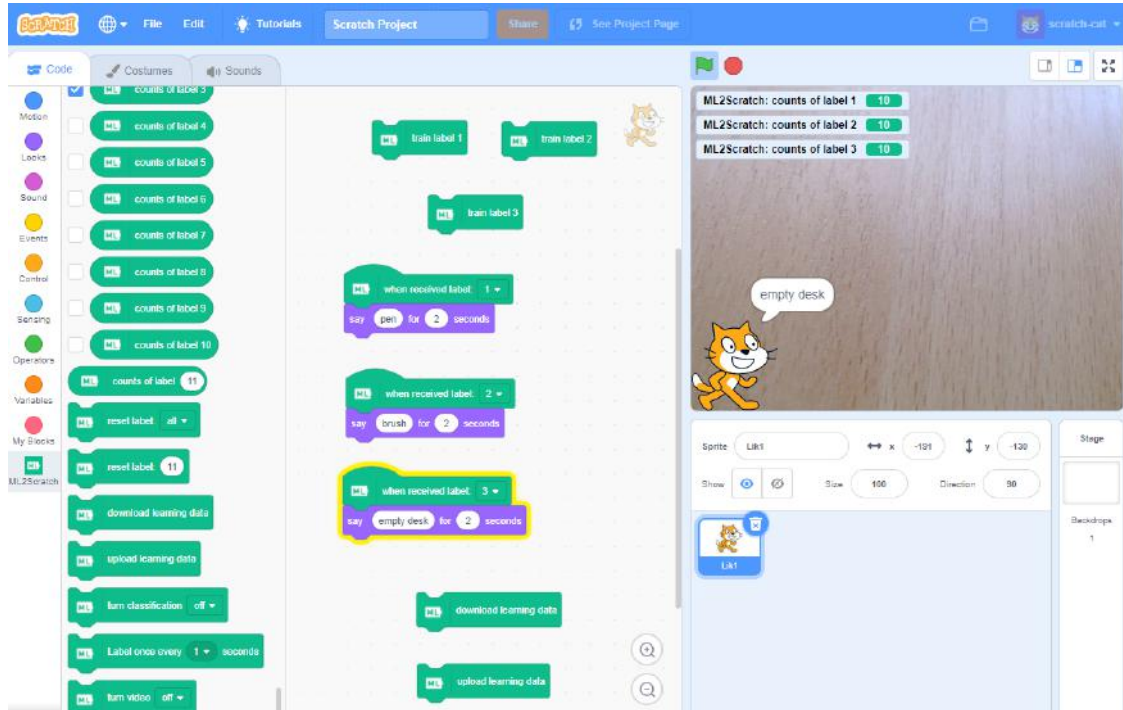
Stap 10: plaats willekeurig objecten uit twee groepen in het cameraweergegebied en noteer wat er gebeurt. Werkt het? Wat gebeurt er als er geen object wordt geplaatst? Hoe los je een leeg bureauprobleem op?

Stap 11: Oplossing: gebruik het derde label, train het op een leeg bureau en voeg het label 3-blok toe wanneer het is ontvangen met bijvoorbeeld een leeg bureau gedurende 2 seconden, zoals weergegeven



Stap 12: Bespreek de nauwkeurigheid van de voorspelling. Probeer elk label meer te trainen en vergelijk het met eerdere resultaten. Verbetert de nauwkeurigheid?

Stap 13: Gebruik meer groepen objecten om meer labels te trainen. Download en upload uw getrainde gegevens met downloadleergegevens en upload leergegevensblokken (klik gewoon op het blok om het .json-bestand op te slaan of te uploaden)



CONCLUSIE

Beeldclassificatie is een techniek die wordt gebruikt om de klasse van een bepaald object te classificeren of te voorspellen van het object op de foto. Het belangrijkste doel van deze techniek is om kenmerken in een afbeelding nauwkeurig te identificeren.



<http://erasmus-artie.eu>

Think!



What do I know now?



What did I learn?



What do I want to know?



Spraakherkenning en generatie voor beginners in Scratch

Inleiding tot spraakherkenning

Spraakherkenning is de mogelijkheid om een dictaat of gesproken woord te vertalen naar tekst. Het is ook bekend als spraak-naar-tekst (STT) en spraakherkenning.

Dit wordt bereikt door bepaalde stappen te volgen en de software die ervoor verantwoordelijk is, staat bekend als een "spraakherkenningssysteem". SR-systemen worden meestal geïmplementeerd in de vorm van dicteersoftware en intelligente assistenten in personal computers, smartphones, webbrowsers en vele andere apparaten.

Inleiding tot spraakgeneratie

Spraakgeneratie of -synthese (ook afgekort als TTS, Text-to-Speech) is, in tegenstelling tot spraakherkenning, geen technologie die gebruik maakt van de stem, maar deze produceert. Synthetische stemmen zijn over het algemeen de laatste fase van het proces en worden steeds democratischer omdat ze belangrijk zijn in de algehele ervaring van "stem".

Spraaksynthese (TTS) wordt gedefinieerd als de kunstmatige productie van menselijke stemmen. Het belangrijkste gebruik (en wat de creatie ervan veroorzaakte) is de mogelijkheid om een tekst automatisch in gesproken spraak te vertalen.

Hoofdidee:

Een inleiding tot spraakherkenning en -generatie voor beginners via een eenvoudig voorbeeldprogramma in de Makeblock-applicatie.



<http://erasmus-artie.eu>

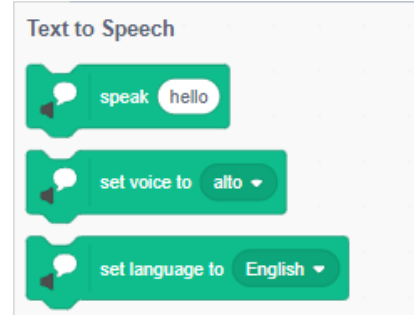


Applicaties

Scratch (MIT)

<https://scratch.mit.edu/projects/editor/>

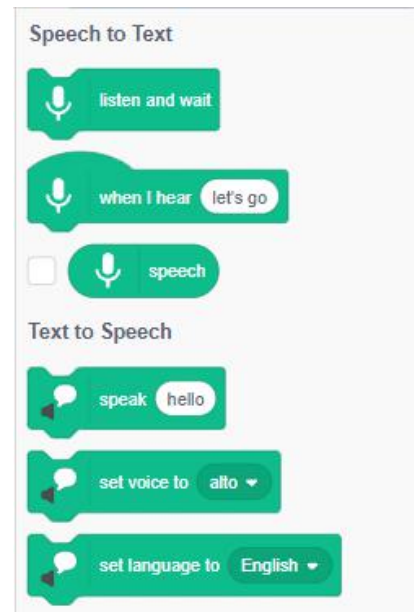
Alleen tekst-naar-spraak-extensie is beschikbaar (3 blokken)



Scratch (ML4KIDS)

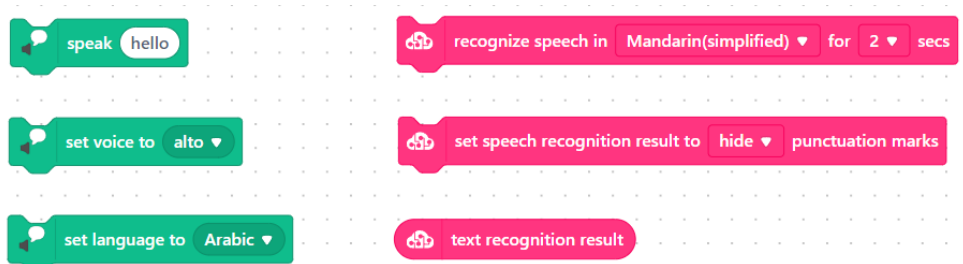
<https://machinelearningforkids.co.uk/scratch3/>

Tekst-naar-spraak (3 blokken) en spraak-naar-tekst (3 blokken) extensies zijn beschikbaar



Makeblock - <https://ide.mblock.cc/>

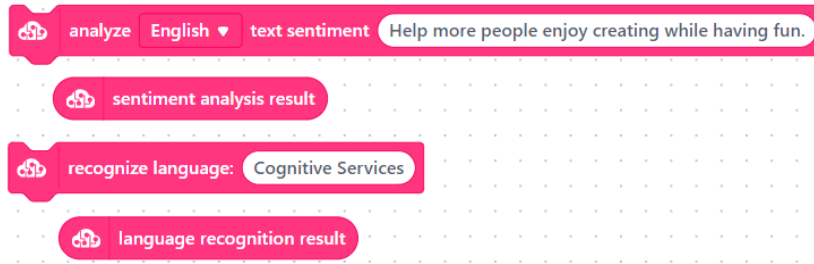
Tekst-naar-spraak (3 blokken), Cognitieve services (3 blokken gerelateerd aan spraak-naar-tekst)





**BONUS*

Makeblock bevat ook blokken voor taalherkenning en tekstsentimentanalyse



Sentimentanalyse is het proces van het detecteren van positief of negatief sentiment in tekst. Het wordt vaak door bedrijven gebruikt om sentiment in sociale gegevens te detecteren, de merkreputatie te peilen en klanten te begrijpen. Modellen voor sentimentanalyse richten zich op polariteit (positief, negatief, neutraal) maar ook op gevoelens en emoties (boos, blij, verdrietig, enz.), urgentie (urgent, niet urgent) en zelfs intenties (geïnteresseerd v. niet geïnteresseerd).

Meer informatie over sentimentanalyse: <https://monkeylearn.com/sentiment-analysis/>

Om dit te illustreren, nemen we een voorbeeld met Makeblock.



Stap 1: Makeblock-pagina openen: <https://ide.mblock.cc/>

Stap 2: Extensies toevoegen: Cognitieve services en Tekst naar spraak

Stap 3: Vink aan om de volgende blokken van het type reporter weer te geven:

- ✓ spraakherkenning resultaat
- ✓ resultaat taalherkenning
- ✓ resultaat van sentimentanalyse

Stap 4: gebruik het **resultaat van spraakherkenning** bij het **herkennen van taal** en het **analyseren** van tekstsentimentblokken

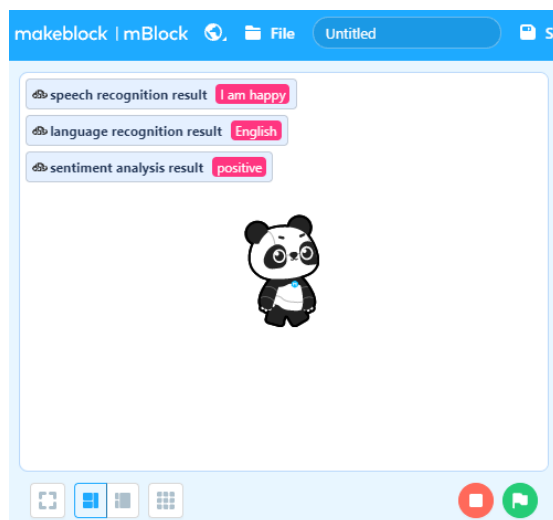
Stap 5: Gebruik deze reeks blokken



```
when clicked
  recognize speech in English for 2 secs
  recognize language: speech recognition result
  analyze English text sentiment speech recognition result
  wait 2 seconds
  set voice to tenor
  set language to English
  if sentiment analysis result = negative then
    speak This is not good
  if sentiment analysis result = positive then
    speak This is good
```

f

Stap 6: Zet je microfoon en speakers aan, start de code en zeg "I am happy" (u zult zien dat het pop-upvenster uw stem gedurende 2 seconden opneemt - als u meer nodig heeft, verhoogt u de waarde in spraakherkenning in het Engels gedurende 3 seconden (of langer). U hoort "Dit is goed" of "Dit is niet goed", afhankelijk van de resultaten van de sentimentanalyse





Stap 7: Start de code opnieuw, zeg iets anders en wacht op de resultaten van de sentimentanalyse

Spraakherkenning is de mogelijkheid om een dictaat of gesproken woord te vertalen naar tekst. Het is ook bekend als spraak-naar-tekst en spraakherkenning. Dit wordt bereikt door bepaalde stappen te volgen en de software die ervoor verantwoordelijk is, staat bekend als een "spraakherkenningssysteem". Spraakherkenningssystemen worden meestal geïmplementeerd in de vorm van dicteersoftware en intelligente assistenten in pc's, smartphones, webbrowsers en vele andere apparaten.



CONCLUSIE

Spraakherkenning is het vermogen om een dictaat of het gesproken woord om te zetten in tekst.
Spraakgeneratie is een technologie die een stem creëert.

Think!



What do I know now?



What did I learn?



What do I want to know?





Spraakherkenning programmeren in Kras

De laatste jaren is spraakherkenningstechnologie steeds wijdverspreider geworden. Deze technologie wordt vaak gebruikt door zowel bedrijven als particulieren vanwege de vele voordelen die het met zich meebrengt.

"Hé, Siri", "OK Google", enzovoort - Spraakherkenning, vaak bekend als spraakherkenningstechnologie, is geen nieuw concept (SRT). Het verwijst naar een type technologie dat gesproken woorden kan omzetten in machinaal leesbare vormen. U kunt nu communiceren met uw apparaten en ze laten handelen op uw commando's, net zoals in sciencefictionverhalen.

Omdat de meeste spraakherkenningstechnologieën een nauwkeurigheidsperscentage van meer dan 95% hebben, is het geen wonder dat de nieuwste statistieken voor gesproken zoekopdrachten onthullen dat bijna 50% van alle zoekopdrachten in 2022 wordt gedaan met spreken.

Hoofdidee:

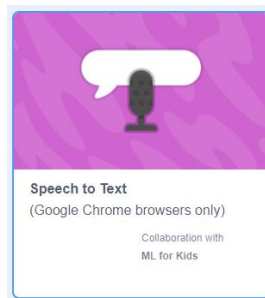
Inleiding tot het spraakherkenningsprogramma en het gebruik ervan op het voorbeeld van één programma.

Als we met spraakherkenning willen werken, zijn er extensies beschikbaar in Scratch- en Makeblock-applicaties.

SCRATCH (ML4KIDS):

Stap 1: Open uw Chrome-webbrowser en ga naar: <https://machinelearningforkids.co.uk/scratch3/>

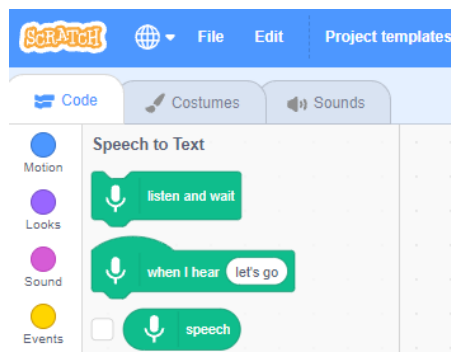
Stap 2: Laad extensie Spraak naar tekst



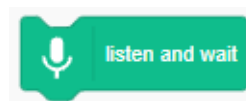
Stap 3: Je ziet de nieuwe groep in Block Palette genaamd „Spraak naar tekst“ en 3 nieuwe blokken daar



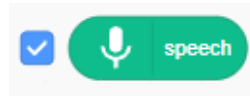
<http://erasmus-artie.eu>



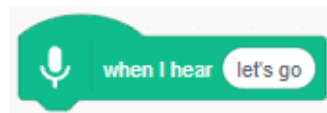
Stap 4: Luister- en wachtblok begint met luisteren en verwerken van gesproken woorden



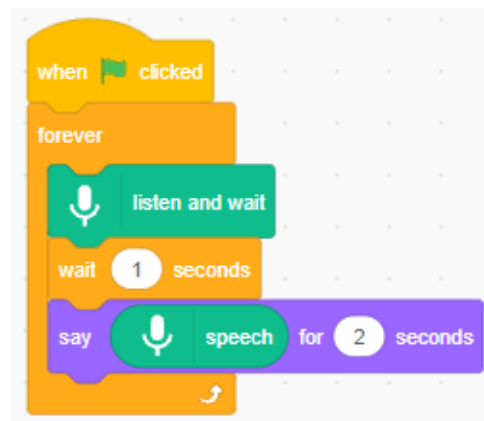
Stap 5: Resultaten van spraakherkenning worden weergegeven in het spraakreporterblok. Vink aan of u wilt dat het spraakherkenningsresultaat op het podium wordt weergegeven.



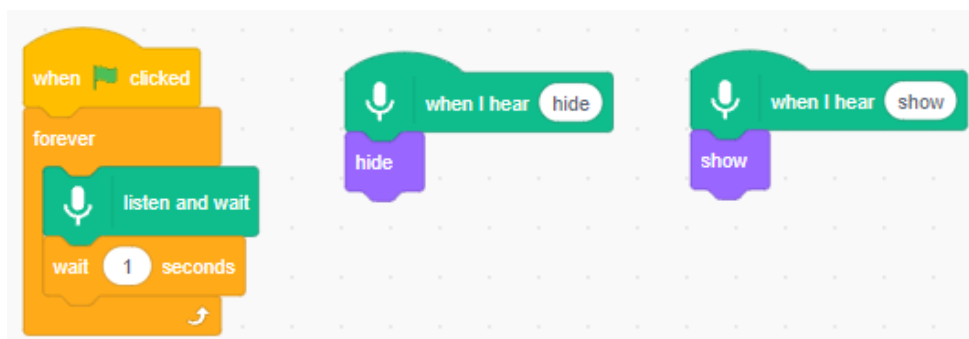
Stap 6: Het laatste blok is het gebeurtenis-triggerblok. Dit blok wacht op het woord in een witte ballon, zoals "iets" in dit voorbeeld en voert vervolgens de reeks blokken uit die eraan zijn gekoppeld.



Stap 7: Laten we dus een eenvoudig luister- en zegprogramma maken. Het enige wat u hoeft te doen is in een lus plaatsen, luisteren en wachten met bijvoorbeeld een blok om de spraakherkenning in actie te zien.



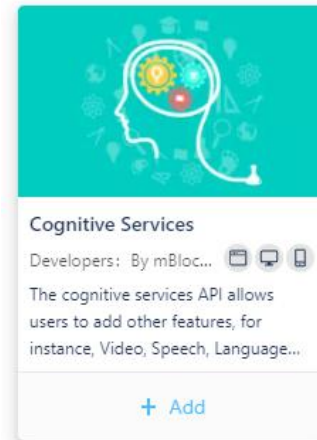
Stap 8: U kunt ook wachten tot een specifiek woord/woorden de gebeurtenis(sen) activeert. Zoals dit verstoppertje spel.



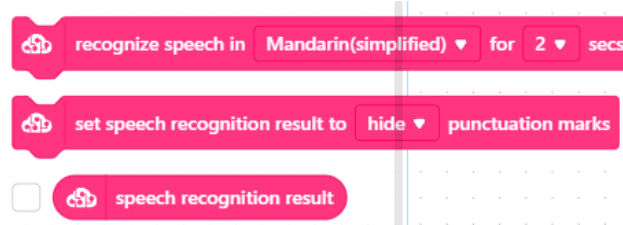
MAKEBLOCK

Stap 1a: Open uw webbrowser en ga naar:
<https://ide.mblock.cc/>

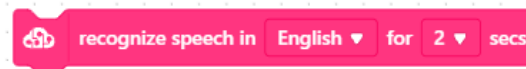
Stap 2a: Laad Sprite-extensie Cognitive Services



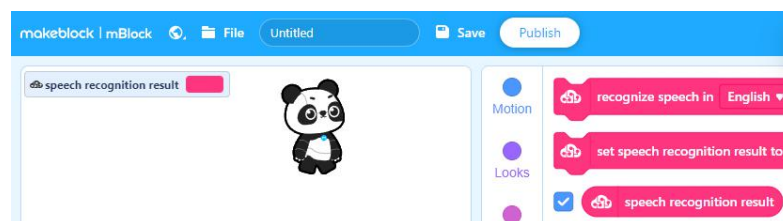
Stap 3a: Er zijn veel blokken in deze groep, maar er zijn er maar weinig die verband houden met spraakherkenning.



Stap 4a: **Herken spraak in <taal> voor <x> seconden** blok start spraakherkenning voor enkele seconden. Het pop-upvenster RECOGNITION verschijnt en u ziet een golfvorm wanneer u spreekt.



Stap 5a: Resultaten van spraakherkenning worden weergegeven in het resultatenblok spraakherkenning. Vink aan of u wilt dat het spraakherkenningsresultaat op het podium wordt weergegeven.



Stap 6a: het laatste blok toont of verbergt de leestekens van het spraakherkenningsresultaat



```

set speech recognition result to hide punctuation marks
  display
  ✓ hide

```

Stap 7a: Laten we dus een eenvoudig luister- en zegprogramma maken in Makeblock. Je zult zien dat het erg lijkt op de Scratch-versie.

```

when clicked
  forever
    recognize speech in English for 2 secs
    wait 1 seconds
    say speech recognition result for 2 seconds

```

CONCLUSIE

Spraakherkenningstechnologie maakt spraakcommunicatie tussen gebruikers en computers mogelijk.

Think!



What do I know now?



What did I learn?



What do I want to know?



Programmeren van spraakgeneratie in kras

Text-to-speech (TTS) is een soort ondersteunende technologie die digitale tekst hardop voorleest. Het wordt soms "voorleestechologie" genoemd. Met een klik op een knop of een vingeraanraking kan TTS woorden op een computer of ander digitaal apparaat omzetten in audio. TTS is erg handig voor kinderen en volwassenen die moeite hebben met lezen. Maar het kan ook helpen bij het schrijven en bewerken, en zelfs bij het scherpstellen.

Een Concept-to-Speech (CTS)-systeem zet de conceptuele representatie van een uit te spreken zin om in spraak. Hoewel sommige CTS-systemen bestaan uit onafhankelijk gebouwde tekstgeneratie- en Text-to-Speech (TTS)-modules, versterken de meeste bestaande CTS-systemen de verbinding tussen deze twee modules met een prosodische voorspellingsmodule die taalkundige kennis van de tekstgenerator gebruikt om prosodische tekst te voorspellen. functies voor het genereren van TTS.

Spraakgeneratie kan elke tekst in spraak omzetten. Spraakgeneratie is het produceren van gesproken berichten als reactie op signalen van een gegevensverwerkings- of controlesysteem. De selectie van berichten wordt geproduceerd door spraakklanken samen te stellen uit een reeks grondbeginselen die van kunstmatige oorsprong kunnen zijn of kunnen zijn geëxtraheerd door verwerking van die door mensen geproduceerde.

Hoofdidee:

Een inleiding tot de spraakgenerator en het gebruik ervan door middel van een voorbeeldprogramma

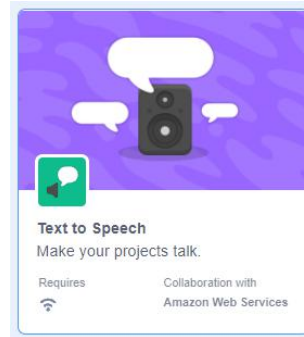


<http://erasmus-artie.eu>

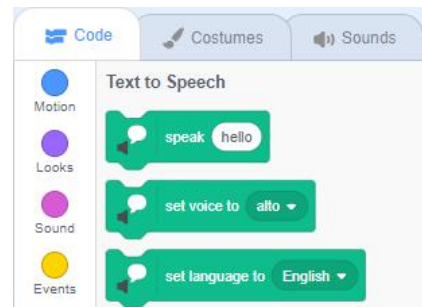


Stap 1: Open uw Chrome-webbrowser en ga naar: <https://machinelearningforkids.co.uk/scratch3/>

Stap 2: laad de extensie Tekst naar spraak

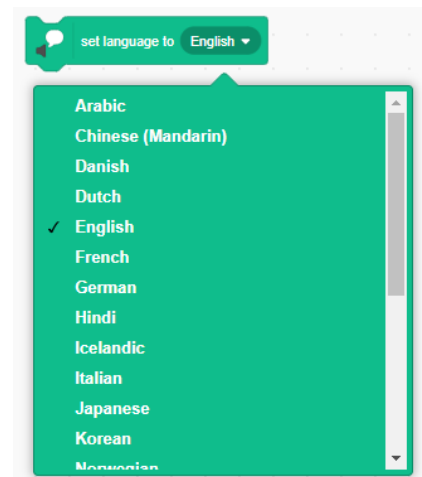


Stap 3: Je ziet de nieuwe groep in Block Palette genaamd "Tekst naar spraak" en 3 nieuwe blokken daar



Stap 4: Laten we van onderaf gaan - taal instellen blokkeren

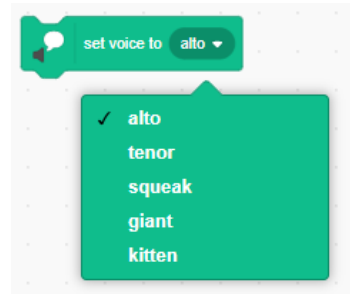
Dit blok stelt de uitvoertaal in - dat kan kies het uit de vervolgkeuzelijst





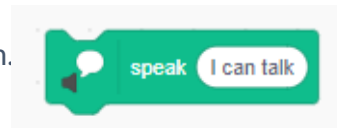
Stap 5: **Volgend blok** - stem instellen om het stemtype in te stellen.

U kunt kiezen: alt, tenor, piep, reus of kitten.

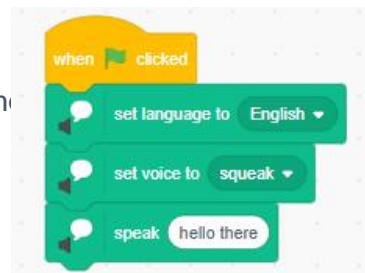


Stap 6: En het belangrijkste blok - **spreekblok**.

Dit blok "spreekt" de tekst in witte ballon, zoals "Ik kan praten" in het onderstaande voorbeeld. Wijzig dit in alles wat je wilt en klik op het blok om het te horen. Zorg ervoor dat uw luidsprekervolume is ingeschakeld voordat u gaat testen.

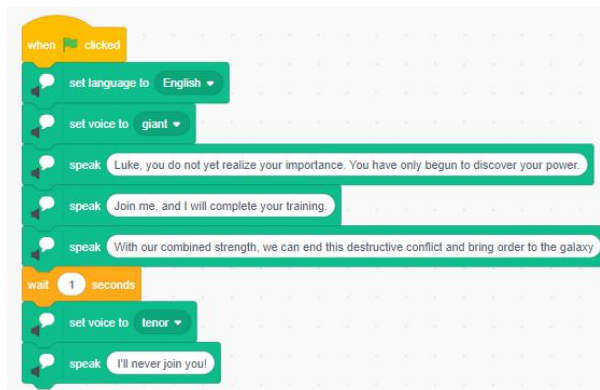


Stap 7: Kortom, niet het moeilijkste werk laat de Scratch-sprite praten. Alles wat je moet doen is om de taal en stem in te stellen en te beginnen met



Stap 8: Wat kun je ermee doen?

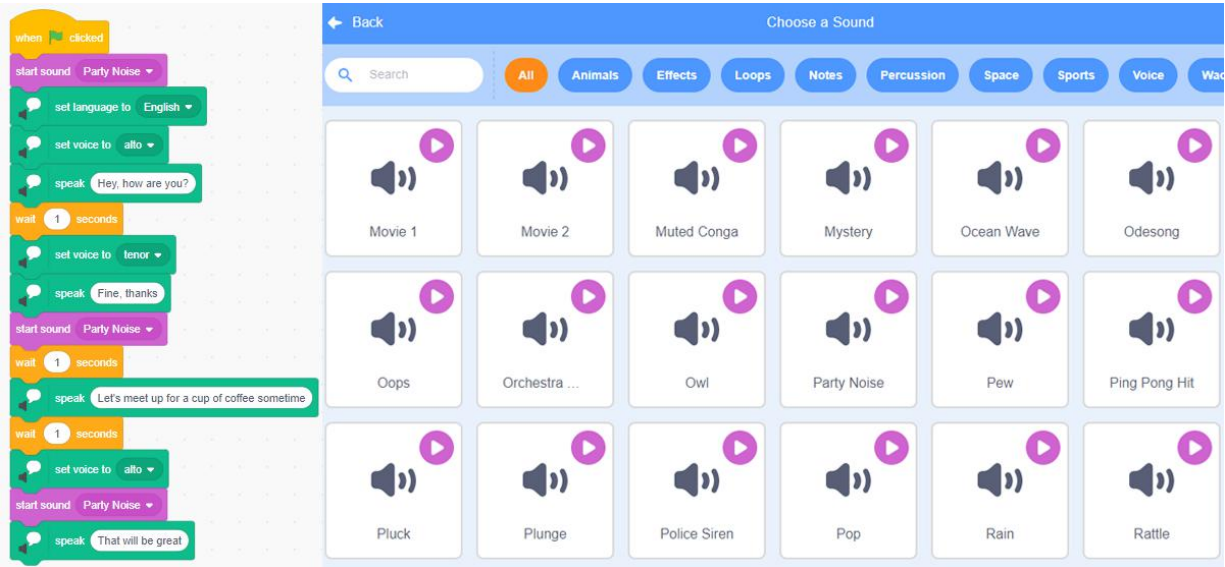
Je kunt de beroemde filmlijnen nabootsen, zoals deze uit Star Wars



Stap 9: Of maak je eigen film, vertel een verhaal...

Combineer het met andere clips uit de geluidsgalerij om de scène realistischer te maken.





Tegenwoordig hebben we veel apparaten voor het genereren van spraak. Spraakgenererende apparaten laten mensen woorden en zinnen elektronisch 'uitspreken'. Spraakgenererende apparaten zijn draagbare elektronische apparaten die woorden of zinnen spelen wanneer de gebruiker een schakelaar aanraakt of knoppen of toetsen indrukt. Sommige apparaten 'spreken' woorden terwijl de woorden op een toetsenbord worden getypt. Spraakgenererende apparaten stellen mensen die geen gesproken taal kunnen gebruiken in staat om elektronisch te 'spreken'. Spraakgenererende apparaten worden sinds de jaren negentig gebruikt om autistische kinderen te helpen communiceren.



<http://erasmus-artie.eu>

CONCLUSIE

Tekst naar spraak converteren (Tekst naar spraak - TTS) is een type technologie dat digitale tekst hardop voorleest.

Think!



What do I know now?



What did I learn?



What do I want to know?

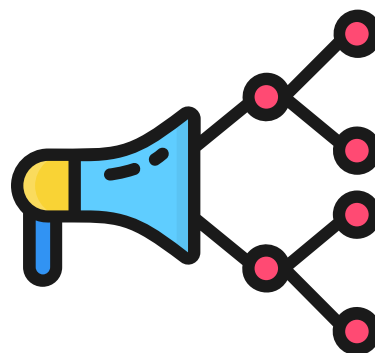


Project met spraakgestuurd object

Is het mogelijk om op de een of andere manier te profiteren van spraakgestuurde objecten?

Hoofdidée:

Inzicht in het stembesturingsalgoritme van de faciliteit en het gebruik ervan in praktisch werk.

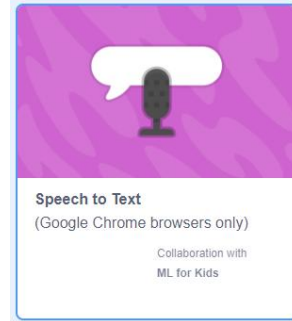




Door middel van een project instrueert de leraar de leerlingen over Scratch-commando's en -vaardigheden en hoe ze een model kunnen trainen om spraak in tekst om te zetten.

Stap 1: Open uw Chrome-webbrowser en ga naar:
<https://machinelearningforkids.co.uk/scratch3/>

Stap 2: Laad extensie Spraak naar tekst
(STT - alleen Google Chrome-browsers)

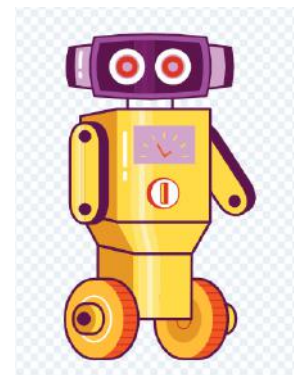


Stap 3: Verwijder Cat sprite door op het sprite-prullenbakpictogram te klikken

Stap 4: Download maze.png van:
<https://drive.google.com/file/d/11YBBhQcIhVfHYMWeLkhgAYKSswfv33pT5/view?usp=sharing> en upload het naar Scratch als aangepaste sprite



Stap 5: Kies in de sprite-galerij Retro Robot en gebruik het tweede kostuum (Retro Robot b)



Stap 6: Kies in de sprite-galerij de *Home-button*





Stap 7: Maak 3 variabelen (voor alle sprites):

- gameover (laat zien hoe het spel eindigde)
- xm (x positie van robot)
- ym (y-positie van de robot)

Stap 8: Maak een lijst en wijzig de naam in: endtalk (geeft het game over-bericht weer, afhankelijk van hoe de game is geëindigd)

Stap 9: Schakel over naar doolhof sprite en begin met coderen. De eerste paar blokken bepalen de positie, grootte en zichtbaarheid van het doolhof. Forever lus die volgt is een collision checker. In het geval van een botsing wordt de gameover-variabele op 1 gezet.

Stap 10: Schakel over naar Home Button sprite. De code is bijna hetzelfde, behalve dat bij een botsing een plopgeluid wordt afgespeeld en de game over-variabele op 2 wordt gezet.

```
when green flag clicked
  go to x: 0 y: 0
  set size to 120 %
  show
  forever
    if touching Retro Robot ? then
      set gameover to 1
```

```
when green flag clicked
  set size to 100 %
  go to x: 195 y: -120
  show
  forever
    if touching Retro Robot ? then
      play sound pop until done
      set gameover to 2
```





Stap 11: De hoofdcode wordt toegewezen aan de Retro-robotcode en er zijn 5 threads die gelijktijdig starten.

11.1 Vlag (start) code stelt de variabelen, lijstvermeldingen (en het verwijderen van alle voorgaande vermeldingen), positie en zichtbaarheid in. Aan het einde is er een luisterblok (van STT-extensie) in de lus.

11.2 De volgende 4 threads worden geactiveerd door spraakherkenning en elke herkende spraakopdracht wordt afgehandeld door een reeks blokken.

Als er geen botsing wordt gedetecteerd (gameover is 0), beweegt de robot in een specifieke richting (door de xm- of ym-waarde te wijzigen en de robot naar de berekende positie te verplaatsen).

Als een botsing wordt gedetecteerd (gameover is 1 of 2), wordt het game over-bericht (uit de lijst) weergegeven en stopt het programma.

```
when clicked
delete all of endtalk
insert Game over! Robot crashed! at 1 of endtalk
insert You won! Robot returned to base! at 2 of endtalk
set gameover to 0
set xm to -200
set ym to 0
go to x: xm y: ym
go to front layer
set size to 30 %
show
forever
listen and wait
```





```
when I hear up
if (gameover = 0) then
  change ym by 20
  glide 0.25 secs to x: xm y: ym
else
  start sound computer beeps1
  say item gameover of endtalk for 3 seconds
  stop all

when I hear left
if (gameover = 0) then
  change xm by -20
  glide 0.25 secs to x: xm y: ym
else
  start sound computer beeps1
  say item gameover of endtalk for 3 seconds
  stop all

when I hear right
if (gameover = 0) then
  change xm by 20
  glide 0.25 secs to x: xm y: ym
else
  start sound computer beeps1
  say item gameover of endtalk for 3 seconds
  stop all

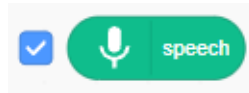
when I hear down
if (gameover = 0) then
  change ym by -20
  glide 0.25 secs to x: xm y: ym
else
  start sound computer beeps1
  say item gameover of endtalk for 3 seconds
  stop all
```





Stap 12: Probeer spraakopdrachten in uw moedertaal te gebruiken en kijk hoe het werkt.

Besprek hoe (on)juiste spelling de spraakherkenning beïnvloedt.
Schakel spraakherkenningsresultaten in (controleer).



Objectvolrobots kunnen, als ze slim met spraak kunnen worden bestuurd, een enorme hulp zijn voor lichamelijk gehandicapten. Een spraakherkenningssysteem wordt gebruikt om een reeks vooraf gedefinieerde commando's te herkennen, zoals vooruit, achteruit, links, rechts en draaien bij een bepaalde draaihoek. De robot navigeert zijn weg volgens het spraakcommandosignaal, terwijl hij ook het gewenste object volgt. De verwerking van het spraakcommandosignaal wordt in realtime uitgevoerd met behulp van een on-time cloudserver die het omzet in tekstvorm. De opdracht signaaltekst wordt vervolgens via het Bluetooth-netwerk naar de robot verzonden om de differentiële aandrijving te regelen. Het prototype van een slimme robot bestaat uit drie subsystemen: een spraakherkenningssysteem, een objectvolgsysteem en het op differentiële aandrijving gebaseerde bewegingscontrolesysteem. De nauwkeurigheid en efficiëntie van het spraakherkenningssysteem wordt onderzocht door middel van een reeks experimenten. Het effect van factoren als geluid en afstand etc. wordt onderzocht, met bemoedigende resultaten. Het prototype van de robot kan de spraakopdrachten binnen een Bluetooth-bereik, d.w.z. 10 m, herkennen. Ook worden mogelijke uitbreidingen besproken die kunnen leiden tot een breed scala aan verdere toepassingen.

CONCLUSIE

Met onze stem kunnen we objecten in een computerprogramma aansturen.

Think!



What do I know now?



What did I learn?



What do I want to know?



Inleiding tot hardware - microcontroller, camera en Motor bestuurder

Een deel van dit curriculum is gerelateerd aan echte, fysieke apparaten met AI-mogelijkheden. Daartoe heeft de Croatian Robotic Association een kleine mobiele robot ontwikkeld om programmeerkennis uit eerdere lessen te implementeren. Laten we eerst eens kijken waaruit onze robot zou moeten bestaan. Het is duidelijk dat mechanische onderdelen - behuizing, wielen en gelijkstroommotoren - als aandrijving zullen worden gebruikt. Hoe zit het met elektronische onderdelen? Net als de meeste levende wezens, heeft onze robot de hersenen (microcontroller), de ogen (AI-camera) en de zenuwen/impulsen (gelijkstroommotoraandrijving) nodig. Laten we die onderdelen eens bekijken.

Er zijn 2 opties om een robot te bouwen:

Maqueen Plus robotkit + HuskyLens AI-camera (eenvoudig - geschikt voor beginners)

micro: Maqueen Plus is een geavanceerde STEM educatieve robot voor Micro:bit. Krachtig en slim, deze micro:bit-robot heeft geoptimaliseerd energiebeheer en een krachtige voeding: hij kan volledig compatibel zijn met de AI HuskyLens vision-sensor, waardoor hij een toegankelijk leermiddel is.

Micro:Maqueen Plus heeft een groot en stabiel chassis, ingebouwde functies en meerdere uitbreidingspoorten. Het is niet alleen geschikt voor klassikaal onderwijs, maar kan ook worden gebruikt voor uitgebreide naschoolse trainingen en robotwedstrijden.

Arduino UNO + Motor driver + HuskyLens AI camera (complexer - alleen voor ervaren gebruikers)



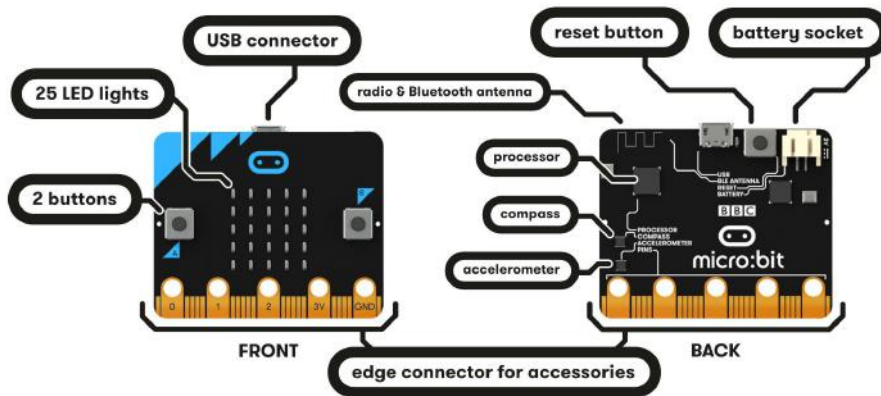
<http://erasmus-artie.eu>



Hersenen 1 - micro:bit

Een "brein" in de elektronische wereld wordt een processor of, in dit specifieke geval, een microcontroller genoemd.

Micro:bit is een gebruiksvriendelijke, krachtige en kosteneffectieve microcontroller in zakformaat die is ontworpen om kinderen en beginners te leren programmeren, zodat ze hun ideeën eenvoudig kunnen omzetten in digitale doe-het-zelfgames, interactieve projecten en robotica.

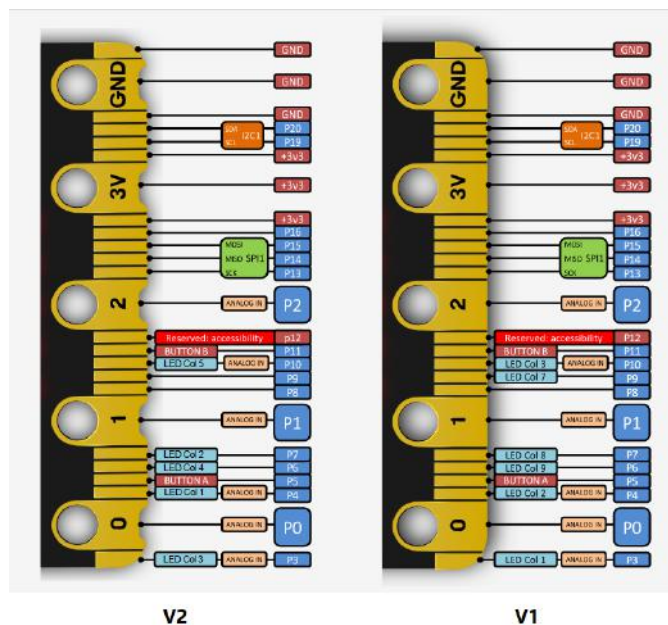


Dankzij de externe I/O-poorten en hardware-ondersteuning is Micro:bit zeer geschikt voor diverse robotgerelateerde leer- en ontwikkelingsactiviteiten.

Micro:bit layout en technische specificaties

Er zijn twee versies van micro:bit op de markt - lees dit artikel om erachter te komen welke je hebt:

<https://kitronik.co.uk/blogs/resources/explore-micro-bit-v1-microbit-v2-differences>





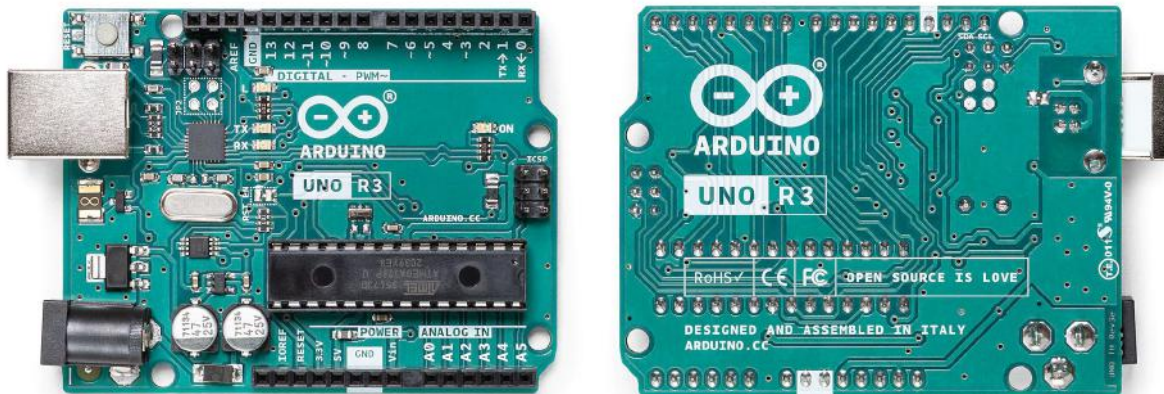
- Een klein bord dat qua grootte vergelijkbaar is met een creditcard (4 cm x 5 cm)
- Ingebouwde modules, zoals versnellingsmeter, kompas en Bluetooth® Smart-module
- Een microcontroller in zakformaat
- Een 5x5 LED-matrix (ondersteunt ook lichtdetectie)
- Licht- en temperatuursensoren en andere veelgebruikte detectieapparaten

Uitgerust met de M0-processor van ARM, kan micro:bit de meeste fundamentele functies van een robot uitvoeren.

Introduction to micro:bit https://www.youtube.com/watch?v=POkeI_2NXMo

Hersenen 2 - Arduino UNO

Arduino UNO is de meest populaire micro-controller ter wereld met een enorme online community en tal van projecten die beschikbaar zijn op internet. Er zijn veel UNO-klonen op de markt. Een kloon betekent dat de kernarchitectuur, in termen van elektronica, vergelijkbaar is met Arduino UNO, maar er zijn enkele wijzigingen aangebracht om de extra functies aan het bord te bieden. Deze aanpassingen zijn speciaal ontworpen en ontwikkeld voor studenten om codering en microcontroller-architectuur te leren. Laten we Arduino UNO eens bekijken.



Elk van de 14 digitale pinnen op de Uno kan worden gebruikt als invoer of uitvoer en ze werken op 5 volt. Elke pin kan 20 mA leveren of ontvangen als aanbevolen bedrijfsconditie en heeft een interne pull-up weerstand (standaard losgekoppeld) van 20-50k ohm. Een maximum van 40mA is de waarde die op geen enkele I/O-pin mag worden overschreden om permanente schade aan de microcontroller te voorkomen.

- Bovendien hebben sommige pinnen gespecialiseerde functies:
- Serieel: 0 (RX) en 1 (TX). Gebruikt om TTL-seriële gegevens te ontvangen (RX) en te verzenden (TX). Deze pinnen zijn verbonden met de overeenkomstige pinnen van de ATmega8U2 USB-naar-TTL seriële chip.
- Externe onderbrekingen: 2 en 3. Deze pinnen kunnen worden geconfigureerd om een onderbreking te activeren bij een lage waarde, een stijgende of dalende flank of een verandering in waarde.
- PWM: 3, 5, 6, 9, 10 en 11. Zorg voor een 8-bits PWM-uitvoerfunctie.



SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Deze pinnen ondersteunen SPI-communicatie met behulp van de SPI-bibliotheek.

LED: 13. Er is een ingebouwde LED die wordt aangestuurd door digitale pin 13. Als de pin een HOGE waarde heeft, is de LED aan, als de pin LAAG is, is hij uit.

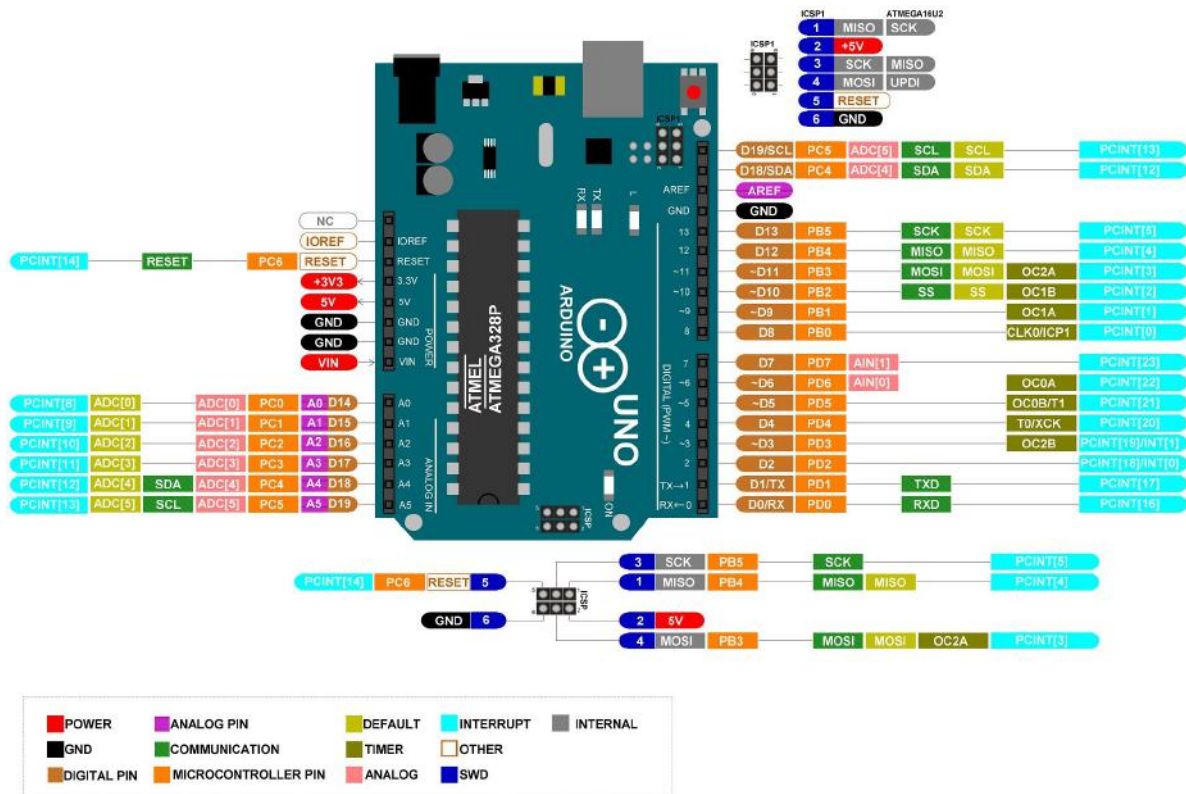
TWI: A4- of SDA-pin en A5- of SCL-pin. Ondersteuning van TWI-communicatie met behulp van de Wire-bibliotheek.

De Uno heeft 6 analoge ingangen, gelabeld A0 tot en met A5, die elk 10 bits resolutie bieden (d.w.z. 1024 verschillende waarden). Standaard meten ze van aarde tot 5 volt, maar het is mogelijk om de bovenkant van hun bereik te wijzigen met behulp van de AREF-pin. Er zijn een paar andere pinnen op het bord:

- AREF. Referentiespanning voor de analoge ingangen.
- Resetten. Breng deze lijn LAAG om de microcontroller te resetten. Meestal gebruikt om een resetknop toe te voegen aan schilden die degene op het bord blokkeren

Arduino UNO-indeling en technische specificaties

Arduino UNO-bord met pinnen uitgelegd:





MICROCONTROLLER	ATmega328P
OPERATING VOLTAGE	5V
INPUT VOLTAGE (RECOMMENDED)	7-12V
INPUT VOLTAGE (LIMIT)	6-20V
DIGITAL I/O PINS	14 (of which 6 provide PWM output)
PWM DIGITAL I/O PINS	6
ANALOG INPUT PINS	6
DC CURRENT PER I/O PIN	20 mA
DC CURRENT FOR 3.3V PIN	50 mA
FLASH MEMORY	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
CLOCK SPEED	16 MHz
LED_BUILTIN	13
LENGTH	68.6 mm
WIDTH	53.4 mm

De hoofdvoeding voor Arduino UNO is via een USB-verbinding, maar hij kan worden gevoed door batterijen die zijn aangesloten via een pin-header (VIN- en GND-pinnen).

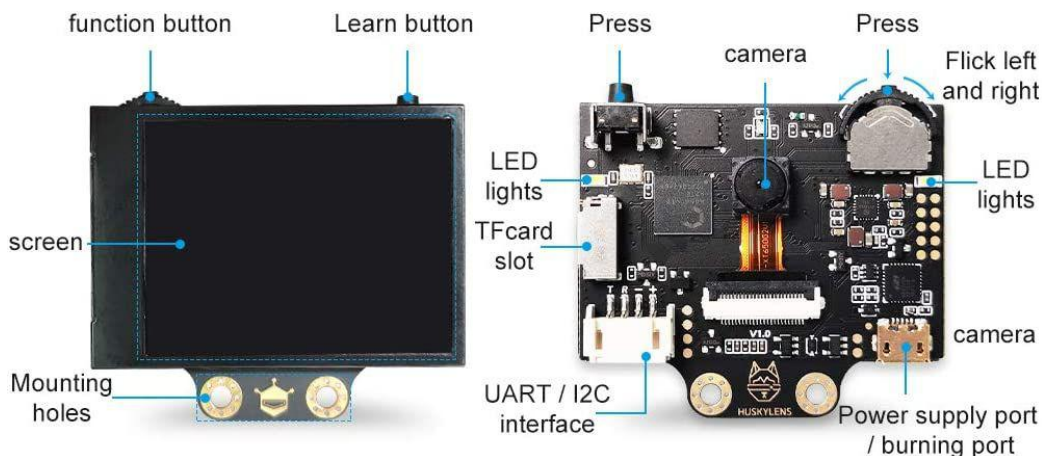
Arduino UNO introduction: <https://www.youtube.com/watch?v=bniUECtJkeU>

Ogen - HuskyLens Kendryte K210 (alle scenario's)

Met deze camera kan onze mobiele robot de omgeving waarnemen. HuskyLens is een gebruiksvriendelijke AI-computervisiesensor met 7 ingebouwde functies: gezichtsherkenning, objecttracking, objectherkenning, lijntracking, kleurherkenning, tagherkenning en objectclassificatie.

Het kan eenvoudig worden aangesloten op elk Arduino/Arduino-compatibel apparaat en micro:bit. Nu kunt u zeer creatieve projecten uitvoeren, zelfs zonder kennis van complexe machine learning-algoritmen.

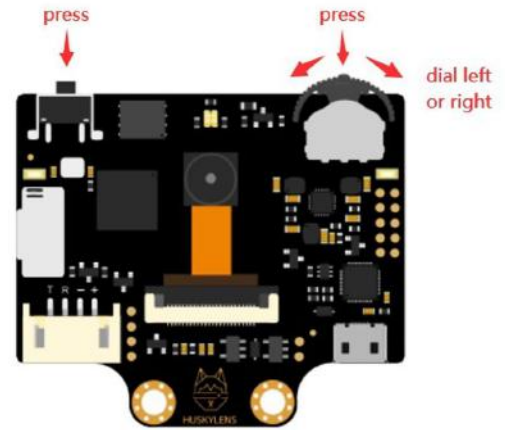
Laten we eens naar de camera kijken.





Buttons

- Op de HuskyLens zitten twee knoppen, de functiekноп en de leerknop. De basisbediening van deze twee knoppen wordt als volgt weergegeven:
- Draai de "functiekноп" naar links of rechts om tussen verschillende functies te schakelen.
- Druk kort op de "leerknop" om het opgegeven object te leren; druk lang op de "leerknop" om het opgegeven object continu vanuit verschillende hoeken en afstanden te leren; als HuskyLens het object eerder heeft geleerd, drukt u kort op de "Leerknop" om het te laten vergeten.
- Druk lang op de "functiekноп" om het menu op het tweede niveau (parameterinstelling) in de huidige functie te openen. Kies links, rechts of druk kort op de "functiekноп" om gerelateerde parameters in te stellen.

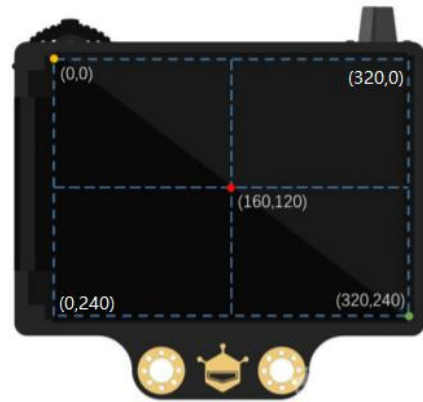


Coördinatie systeem

Wanneer HuskyLens een object detecteert, wordt het doel automatisch geselecteerd door een kleurenkader op het scherm. De coördinaten van de kleurkaderpositie x en y worden toegewezen volgens het volgende coördinatensysteem. Nadat u de coördinaten van de UART / I2C-poort hebt verkregen, kunt u de positie van het object kennen. Formaat:x,y)

Functies:

- 1) Gezichtsherkenning
- 2) Object volgen
- 3) Objectherkenning
- 4) Lijn volgen
- 5) Kleurherkenning
- 6) Tagherkenning
- 7) Objectclassificatie



Kleur instructies

In elke functie zijn de kleurdefinities van het frame en het symbool "+" in het midden van het scherm zijn allemaal hetzelfde, wat helpt u kent de huidige status van HuskyLens.

Colour	Status
From orange to yellow, then from yellow to orange	Have not learned the object yet but ready to learn
Yellow	Learning the new object
Blue	Have learned the object and recognized it



De RGB LED-indicator wordt gebruikt om de status van de gezichtsherkenningsfunctie aan te geven. De kleuren zijn als volgt gedefinieerd.

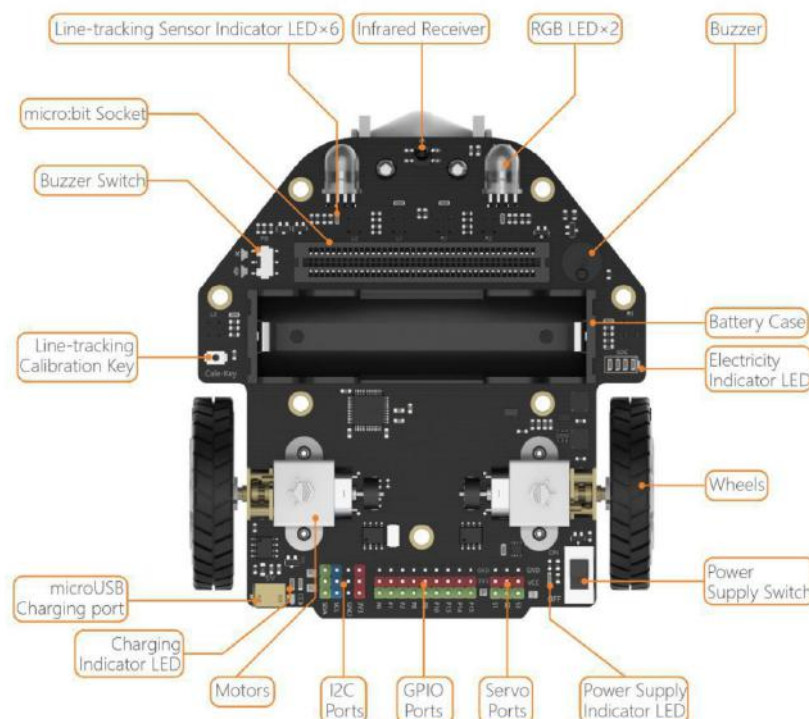
Colour	Status
Blue	Have not learned the face yet, but detected the face
Yellow	Learning the new face
Green	Have learned the face and recognized it

Volledige handleiding: https://wiki.dfrobot.com/HUSKYLENS_V1.0_SKU_SEN0305_SEN0336

Zenuwen/impulsen/spieren 1 - Maqueen Plus robotbord

Maqueen Plus is een slimme programmeerbare educatieve robot die is ontworpen voor beginners. Het kan worden geprogrammeerd met de programmeerplatforms Mind+ en MakeCode. Het is geoptimaliseerd met beter energiebeheer en een voeding met een grotere capaciteit. Het is ideaal voor gebruik met de HuskyLens AI Vision Sensor en wordt geleverd met een groter en stabielere chassis. Het beschikt ook over meer ingebouwde functies en meer uitbreidingspoorten. Het is geschikt voor klassikaal onderwijs en voor uitgebreide naschoolse oefeningen en robotwedstrijden.

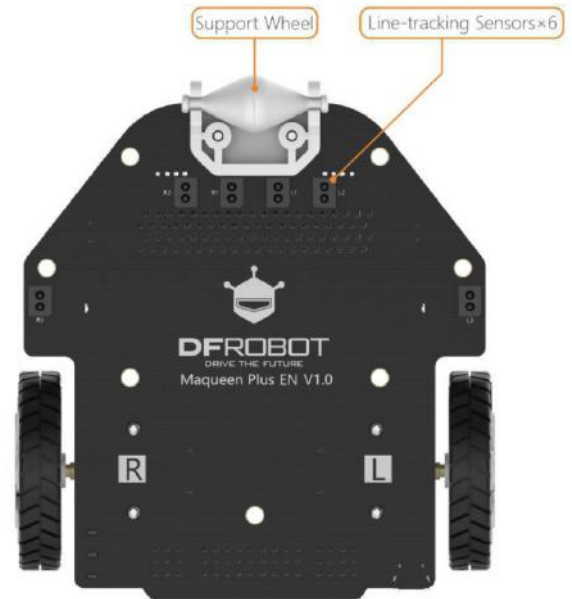
Dit zijn de belangrijkste onderdelen van het Maqueen Plus-bord:





Maqueen plus technische specificaties

- Voeding: 3.7V-18650 lithiumbatterij
- Laadspanning: 5V
- Laadstroom: 900mA, oplaadtijd: 4 uur
- Batterij-indicator: 4 LED's
- Aandrijfmotor: N20-motor 260 tpm
- Zoemer * 1
- RGB-LED * 2
- GPIO-uitbreidingspoorten: P0 P1 P2 P8 P12 P13 P14 P15 P16
- I2C-uitbreidingspoorten * 3
- Servo-uitbreidingspoorten *3
- Lijnvolgsensoren * 6
- Uitgangsgegevens lijnvolgsensor: analoog + digitaal
- Lijnsensorkalibratie: ondersteuning
- Infrarood-ontvangstsensor *1
- Ultrasonische sensor: URM10
- Bovenste metalen plaat * 1
- M3 schroefdraadverbindingen *12
- Kaartformaat: 50cm*50cm
- Afmeting: 107 x 100 mm (4,21 x 3,94 inch)



Maqueen Plus is nu toegankelijk via het Mind+ programmeerplatform. Mind+ is een op Scratch3.0 gebaseerd grafisch programmeerplatform van DFRobot, dat python, Arduino en andere programmeerplatforms ondersteunt. Op dit moment wordt Mind+ toegepast op allerlei soorten sensoren, modules en aanverwante educatieve producten.

Lees meer over Maqueen Plus:

https://github.com/DFRobot/Maqueen_Plus_Basic_Tutorial/blob/master/MBT0021-EN-Maqueen%20Plus%20Basic%20Tutorial.pdf

Terwijl dit scenario wordt voltooid, is de V2-versie van deze robot op de markt verkrijgbaar.

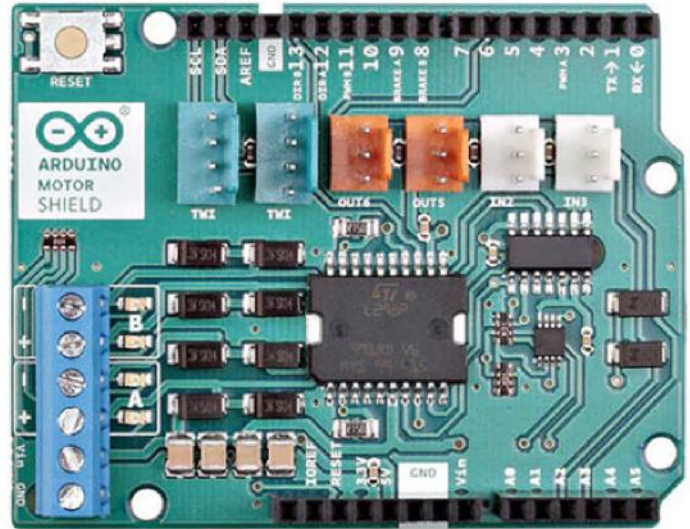




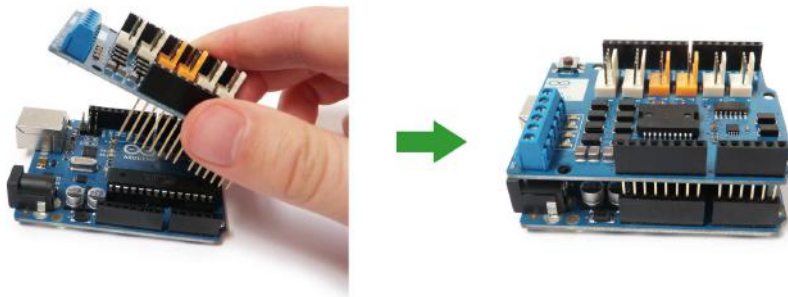
Zenuwen/impulsen/spieren 2 - Arduino Motor Shield Rev 3

Er zijn verschillende manieren waarop we een DC-motor kunnen aansturen; misschien is de gemakkelijkste gewoon door er kracht op uit te oefenen. Zeer vroege uitvindingen die de gelijkstroommotor gebruikten, werkten zo: voeg een stroombron toe en de motor begint te draaien, verander de polariteit en je verandert de richting.

Maar als we iets meer willen doen dan alleen een motor op volle snelheid in twee richtingen laten draaien, hebben we een motorbesturingscircuit nodig. Meer specifiek de dubbele full-bridge driver L298P (chip), die we kunnen vinden op de Motor Shield Rev3.



Motorschild is stapelbaar met Arduino UNO en dat betekent dat u de verbinding met de microcontroller niet hoeft te bedraden. Sluit gewoon de afschermingspennen (mannelijk) aan op de microcontrollerpennen (vrouwelijk).



Het motorschild heeft 2 kanalen, waarmee twee gelijkstroommotoren of 1 stappenmotor kunnen worden aangestuurd.

Het heeft ook 6 headers voor de aansluiting van Tinkerkit-ingangen, -uitgangen en communicatielijnen. Het gebruik van deze pinnen is enigszins beperkt en wordt daarom niet behandeld in deze tutorial.

Met een externe voeding kan het motorschild veilig tot 12V en 2A per motorkanaal leveren (of 4A naar een enkel kanaal).

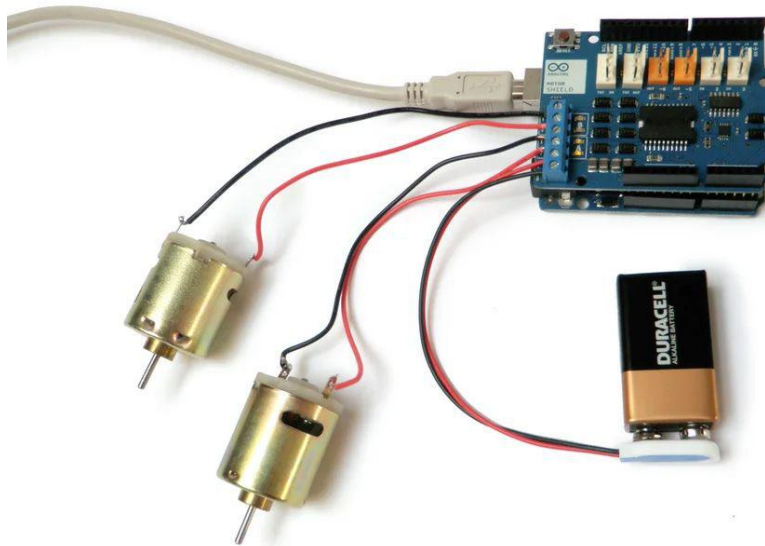
Er zijn pinnen op de Arduino die altijd in gebruik zijn door het schild. Door deze pinnen te adresseren, kunt u een motorkanaal selecteren om te initiëren, de motorrichting (polariteit) specificeren, de motorsnelheid (PWM) instellen, de motor stoppen en starten en de stroomopname van elk kanaal bewaken.

De pinverdeling is als volgt:



Function	Channel A	Channel B
Direction	Digital 12	Digital 13
Speed (PWM)	Digital 3	Digital 11
Brake	Digital 9	Digital 8
Current Sensing	Analog 0	Analog 1

En tot slot, zo zou het eruit moeten zien als het is aangesloten:



http://erasmus-artie.eu

Lees meer op deze Instructables-pagina:

<https://www.instructables.com/Arduino-Motor-Shield-Tutorial/>

CONCLUSIE

Net als de meeste levende wezens heeft onze ARTIEBot-robot hersenen (microcontroller), ogen (AI-camera) en zenuwen/impulsen (motorcontroller).

Think!



What do I know now?



What did I learn?

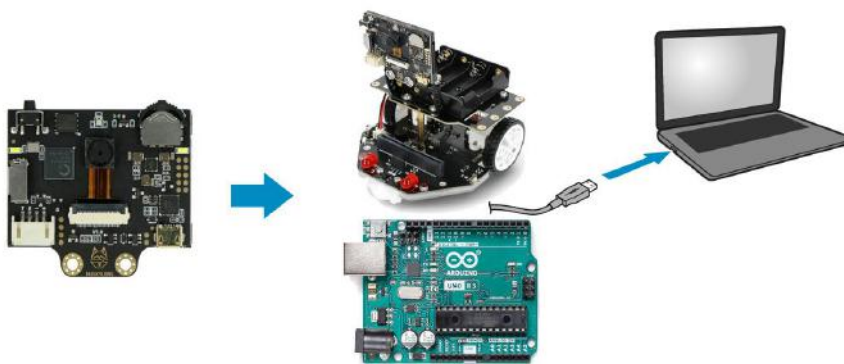


What do I want to know?



Microcontroller aansluiten met camera en pc en aan de slag

Voordat we beginnen met programmeren, moeten we de camera aansluiten op de microcontroller (micro:bit of Arduino UNO) en de microcontroller op de computer.



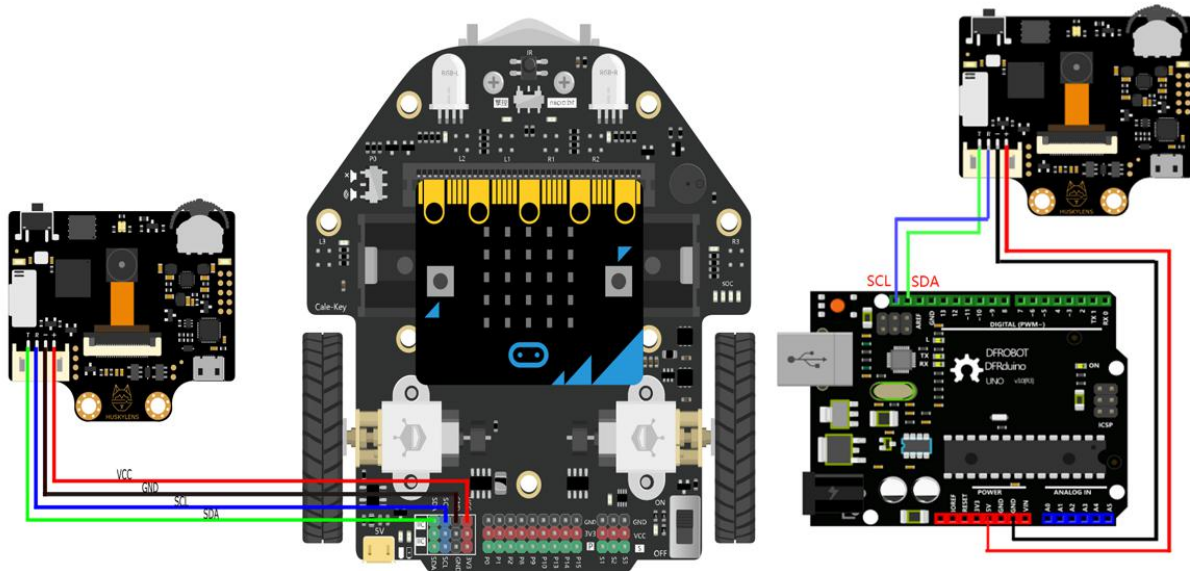
Optie 1: Camera > I2Connection > micro:bit/Maqueen plus
> USB cable > Laptop or PC

Optie 2: Camera > I2Connection > Arduino UNO > USB
cable > Laptop or PC

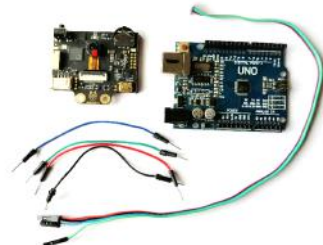
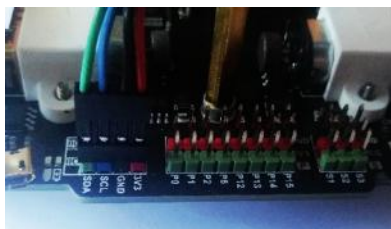


Er is niet veel nieuws te zeggen over USB-verbinding, maar je hebt waarschijnlijk nog nooit van I2C gehoord.

De Inter-Integrated Circuit (I2C)-bus is een tweedraads seriële interface die oorspronkelijk is ontwikkeld door de Phillips Corporation voor gebruik in consumentenproducten. Het volgt een master/slave-hiërarchie, waarbij de master wordt gedefinieerd als het apparaat dat de bus klokt, de slaves adresseert en gegevens schrijft of leest naar en van registers in de slaves. De slaves zijn apparaten die alleen reageren wanneer ze worden ondervraagd door de master, via hun unieke adres. De I2C-bus gebruikt slechts twee bidirectionele lijnen, een seriële datalijn (SDA) en een seriële kloklijn (SCL).

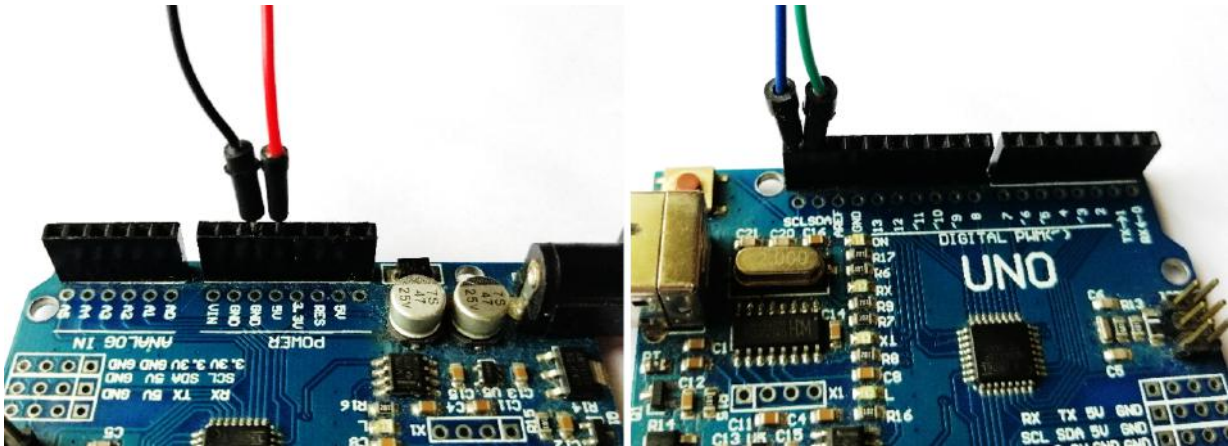


In Optie 1 - de oplossing is heel eenvoudig en het enige wat u hoeft te doen is de kleuren van de draden en de connector op elkaar af te stemmen.



Het zal een beetje lastig zijn in Optie 2 om de camerakabel en de Arduino UNO aan te sluiten, omdat beide connectoren van het vrouwelijke type zijn, dus we hebben 4 jumperdraden van het mannelijke naar het mannelijke type nodig, die te vinden zijn in bijna elke elektronikawinkel die doe-het-zelfonderdelen verkoopt.

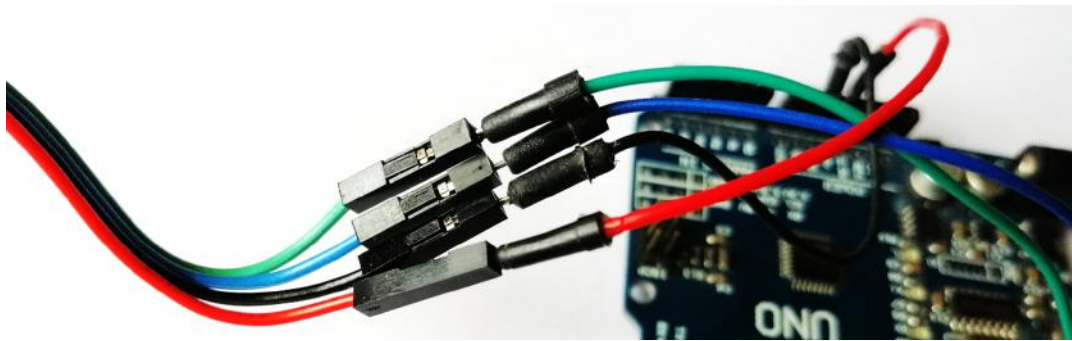




Kies 4 jumpers (rood, zwart, groen en blauw) die overeenkomen met de corresponderende camerapinnen.

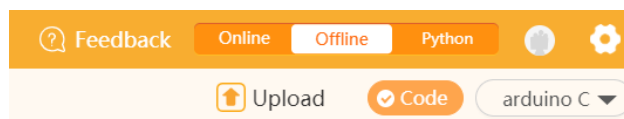
Sluit eerst de rode jumper aan op de 5V-pin op de Arduino UNO, de zwarte jumper op GND, de blauwe jumper op SCL en de groene jumper op de SDA-pin zoals weergegeven in de onderstaande afbeeldingen.

Het enige dat u nog hoeft te doen, is de andere kant van deze jumpers op de cameraconnector te pluggen. Stem de kleuren van de jumpers af op de cameraconnectordraden (zwart op zwart, rood op rood, enz...).



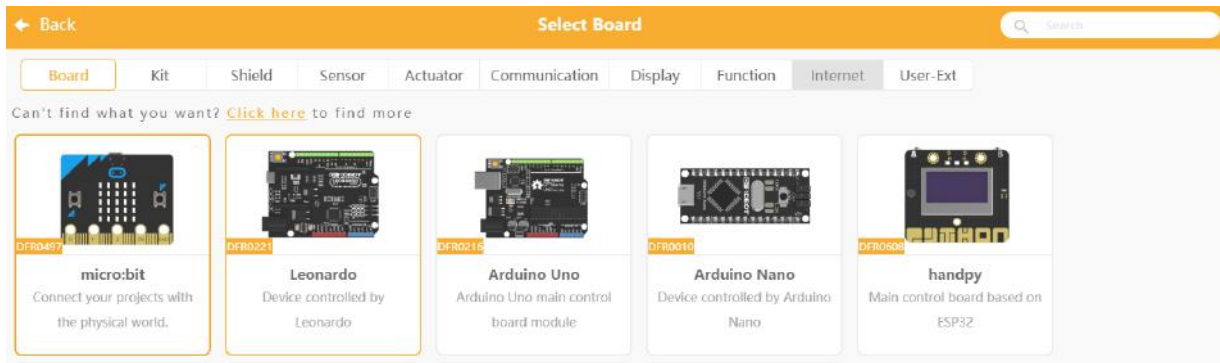
Je hebt nu je camera aangesloten op de microcontroller. Sluit de USB-kabel aan op uw Maqueen plus-robot of Arduino UNO-bord en verbind deze met uw laptop of pc.

Ga naar: <http://mindplus.cc/download-en.html> en downloadversie voor uw computerbesturingssysteem. Installeer en start de Mind+. Schakel eerst naar de offlinemodus.





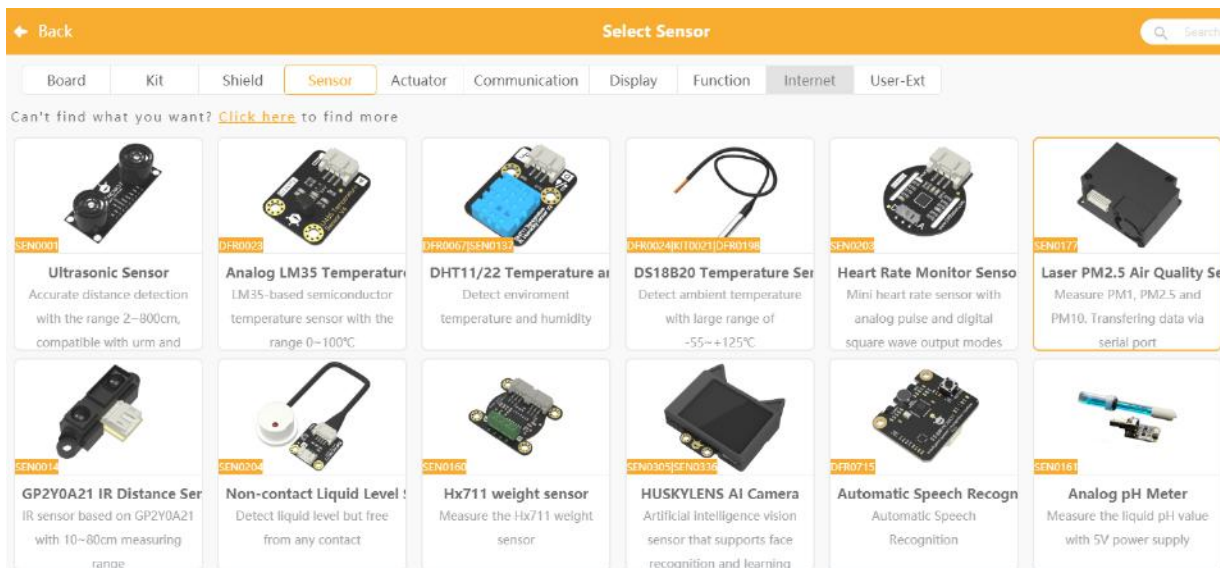
Open Extensies en selecteer Board: Optie 1) micro:bit (als je werkt met) Maqueen plus
Optie 2) Arduino UNO



Alleen voor Optie 1) Schakel over naar het tabblad Shield en selecteer Maqueen Plus of Maqueen Plus V2 (afhankelijk van uw versie)



Zowel Optie 1) als Optie 2) Schakel over naar het tabblad Sensor en selecteer sensor - HUSKYLENS AI-camera

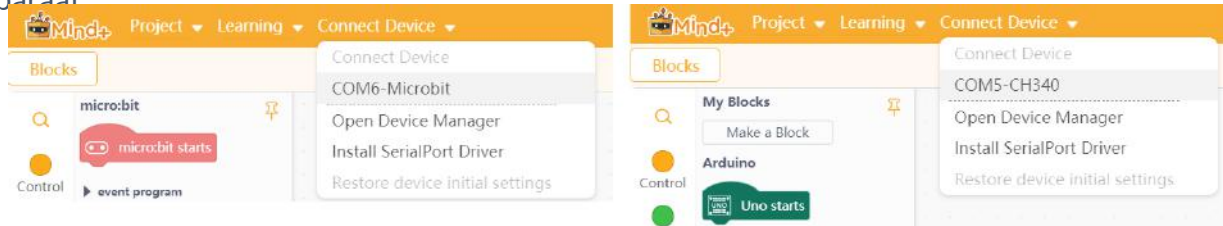


http://erasmus-artie.eu



Klik na het selecteren op <- **Back** en u bent klaar om het geselecteerde bord/robot en de camera te gebruiken. Laten we een test doen om te zien of dit werkt.

Daarvoor moet u het apparaat aansluiten. Klik op Connect Device en selecteer uw poort en apparaat

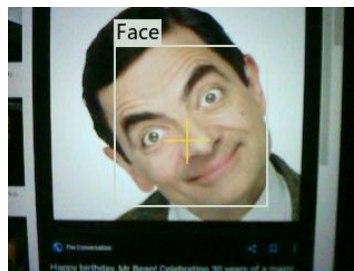


Pak de camera en draai de functieknop naar links tot het woord **"Face recognition"** wordt bovenaan het scherm weergegeven.



Leren en detecteren

1. Face Detection: Richt de HuskyLens op alle gezichten. Wanneer een gezicht wordt gedetecteerd, wordt dit automatisch geselecteerd door een wit kader met woorden "Face" op het scherm.



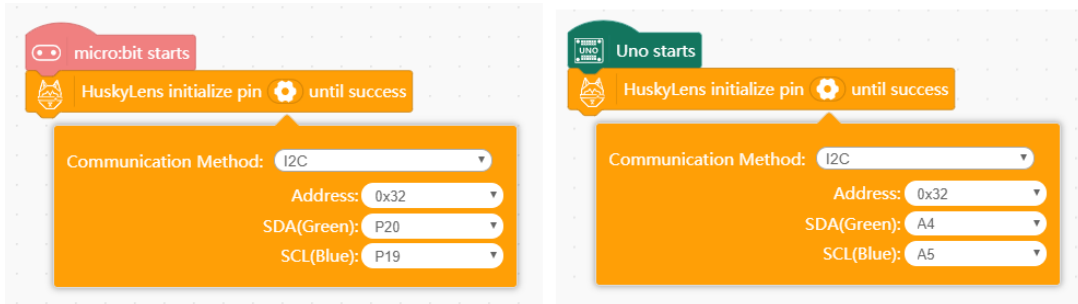
Tips: als u wilt dat HuskyLens uw gezicht leert of herkent, dat wil zeggen een selfie maakt, u het scherm op dit moment niet kunt zien, kunt u de status bepalen aan de hand van de verschillende kleuren van de RGB-indicator.

2. Face Learning: Richt het "+" symbool op een gezicht, druk kort op de "learning button" om het gezicht te leren. Als hetzelfde gezicht wordt gedetecteerd door HuskyLens, verschijnt er een blauw kader met woorden **"Face: ID0"** wordt weergegeven op het scherm, wat aangeeft dat HuskyLens het gezicht heeft geleerd en het nu kan herkennen.



Bezoek: https://wiki.dfrobot.com/HUSKYLENS_V1.0_SKU_SEN0305_SEN0336#target_15 voor volledige gezichtsherkeningsreferentie.

Ga naar Mind+ en begin met programmeren. Nadat micro:bit/Uno is gestart, gebruikt u HuskyLens om de pin te initialiseren tot het succesblok en configureert u deze voor I2C zoals weergegeven in de onderstaande afbeelding voor elke optie.



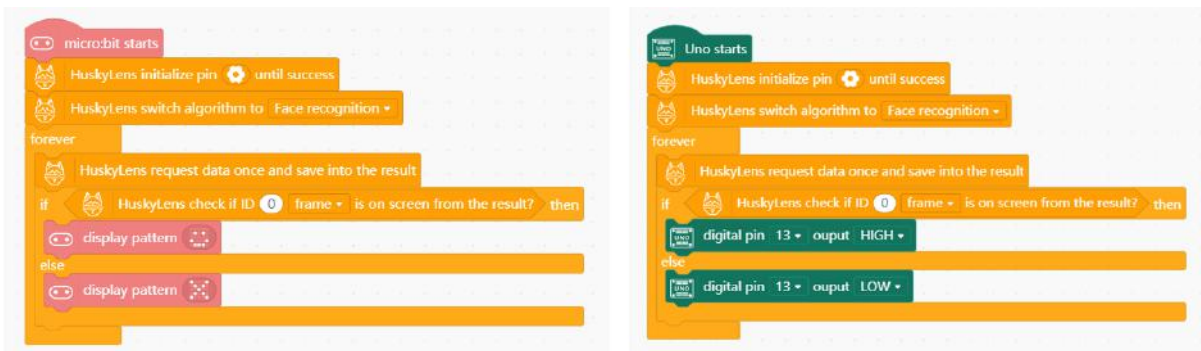
Gebruik vervolgens het blok HuskyLens switch algorithm to Face recognition voor beide opties.



Het wordt gevolgd door de eeuwige lus die detecteert of het gezicht op de camera wordt herkend als Face ID0.

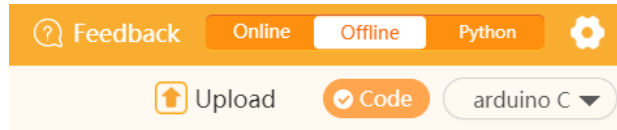
Als er een gezicht wordt gedetecteerd in optie 1), ziet u een gezicht op het micro:bit-scherm. Zo niet, dan ziet u het X-teken.

Als er een gezicht wordt gedetecteerd in optie 2), wordt de ingebouwde LED (D13) ingeschakeld. Zo niet, dan is de LED uitgeschakeld.





Druk de **Upload** om deze code over te zetten naar de micro:bit of Arduino UNO.



Nadat de code is overgedragen, richt u de camera op het “learned” gezicht en LED op D13 moeten worden ingeschakeld. Als u het van het gezicht af beweegt, wordt D13 uitgeschakeld.

Als het werkt zoals beschreven, is alles in orde en klaar voor gebruik in onze mobiele AI-robot.

CONCLUSIE

HuskyLens is een gebruiksvriendelijke AI-computervisiesensor met 7 ingebouwde functies: gezichtsherkenning, objecttracking, objectherkenning, lijntracking, kleurherkenning, tagherkenning en objectclassificatie.

Via de UART/I2C-poort kan HuskyLens worden aangesloten op Arduino en micro:bit om u te helpen bij het maken van zeer creatieve projecten zonder ingewikkelde algoritmen.



Think!



What do I know now?



What did I learn?



What do I want to know?

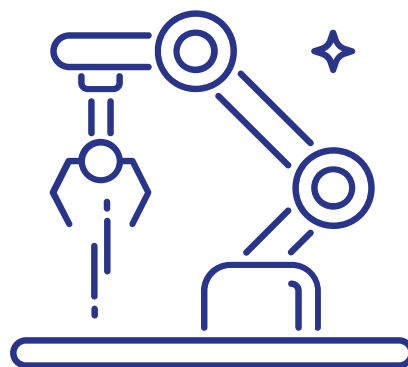


Robots programmeren - bewegen

Laten we eerlijk zijn, robots zijn cool. In deze les bieden we een stapsgewijze, gemakkelijk te volgen zelfstudie (met codevoorbeelden) die u door het proces leidt van het programmeren van een eenvoudige autonome mobiele robot om te bewegen.

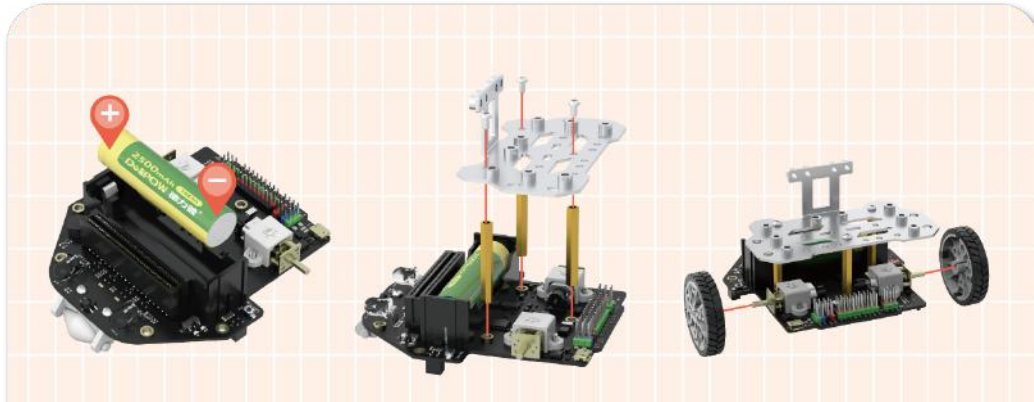
Doel:

Om onze ARTIEbot voor het eerst te programmeren en te zien hoe je hem kunt verplaatsen.





Optie 1a - Montage van de V1 Maqueen Plus Robot (aanbevolen voor beginners)

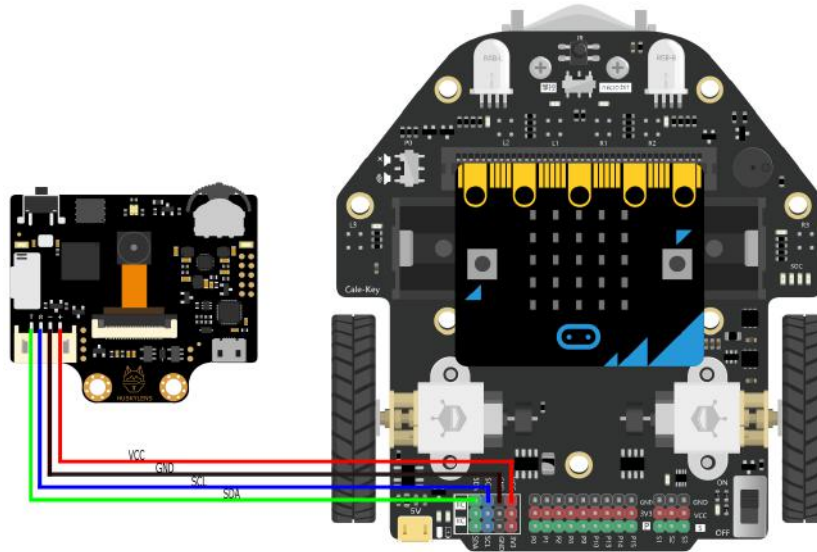


1) plaats de 18650-batterij 2) installeer de beugel 3) bevestig de wielen



4) Monteer de HuskyLens-camera 5) Plaats het micro:bit-bord in de sleuf





6) Sluit de Maqueen Plus-interface aan op de HuskyLens-camera

Gebruik de USB-micropoort (oplaadpoort) aan de achterkant van de robot om de 18650-batterij op te laden:



Battery indicator

Charging port

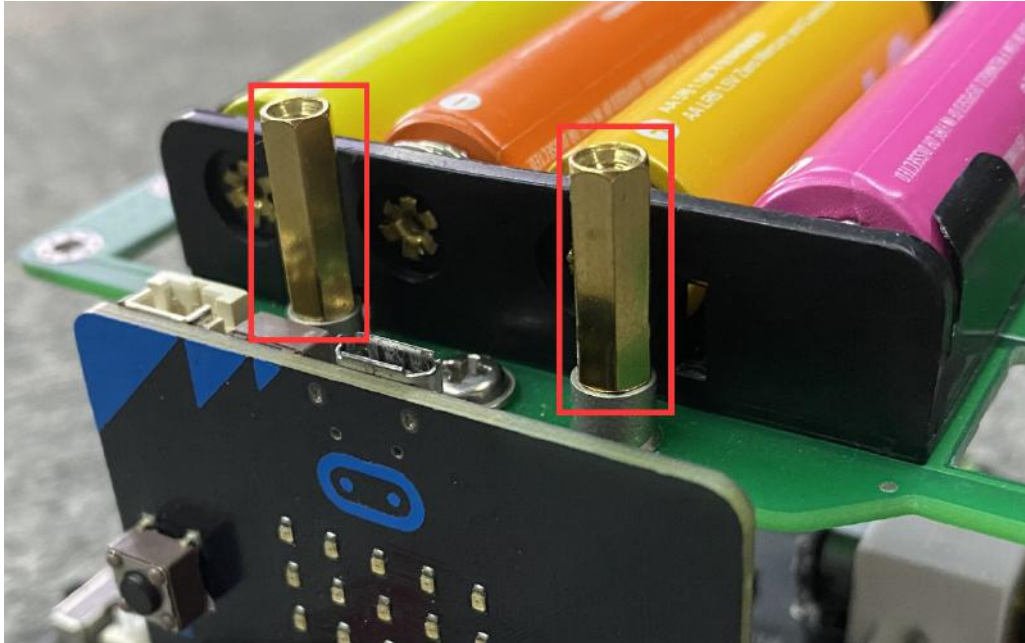
Als de batterij volledig is opgeladen, lichten alle LED's op. De LED's gaan één voor één uit naarmate de batterijcapaciteit geleidelijk afneemt. Als alle lampjes uitgaan, moet de batterij worden opgeladen. Gebruik een smartphone-oplader en sluit deze aan op de oplaadpoort.

Houd er rekening mee dat de oplaadpoort alleen bedoeld is om op te laden. Om de Maqueen Plus-robot te programmeren, gebruik je de USB-microconnector op de micro:bit.

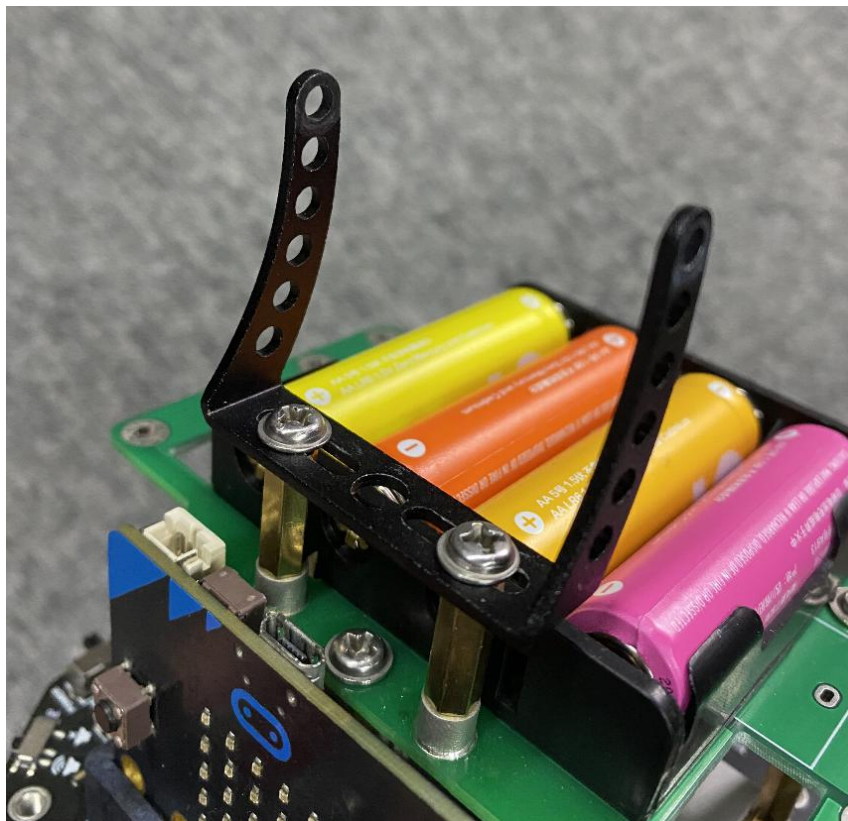
Optie 1b - Maqueen Plus Robot V2 monteren (ook aanbevolen voor beginners)

De Maqueen plus V2-robot wordt voorgeassembleerd geleverd en u hoeft er alleen maar de HuskyLens-camera op te plaatsen

1. Installeer twee koperen afstandhouders (afstanden) die in de kit met materialen zitten op de plaatsen zoals aangegeven in de afbeelding.

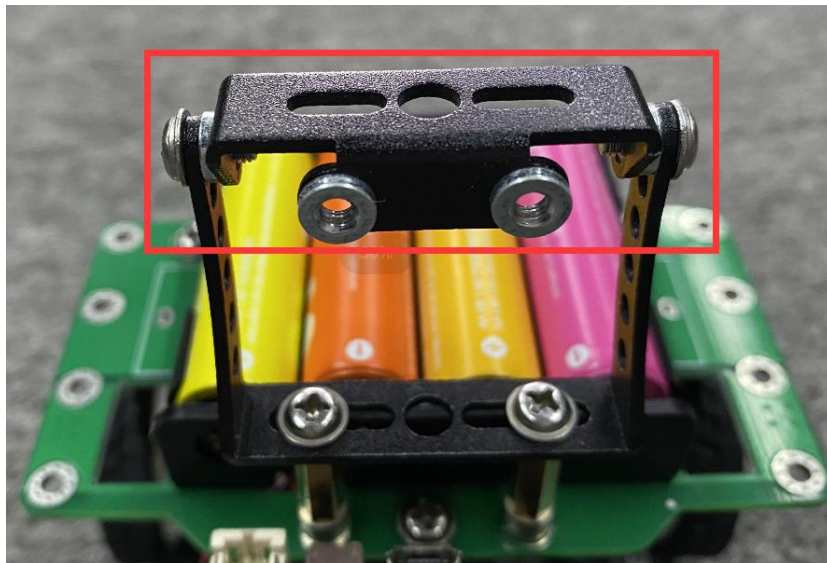


2. Bevestig de boogbeugel (beugel en montageschroeven geleverd bij HuskyLens) aan de koperen paal met schroeven.

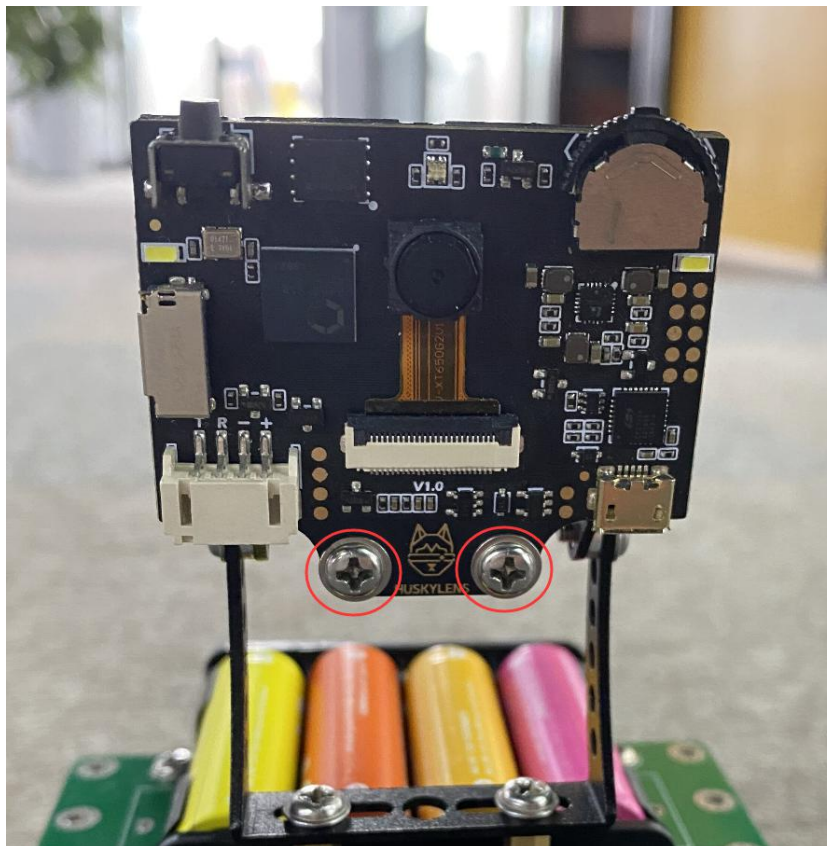




3. Installeer de tweede beugel (de beugel en montageschroeven zijn inbegrepen in de set die bij de HuskyLens wordt geleverd)

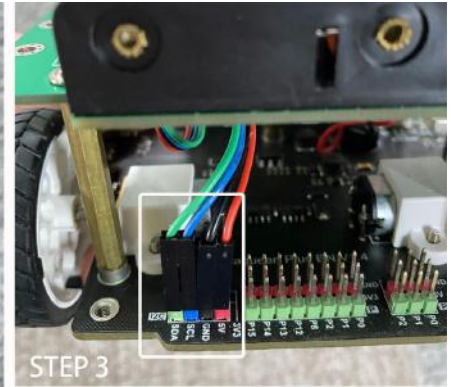
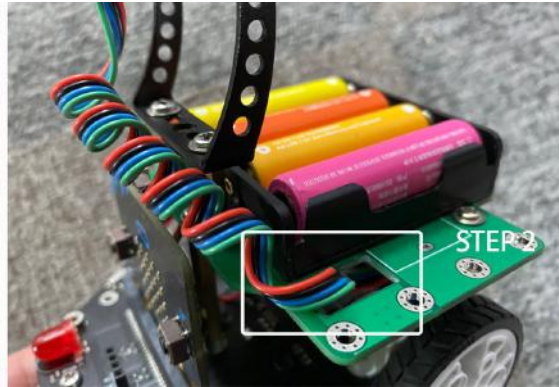
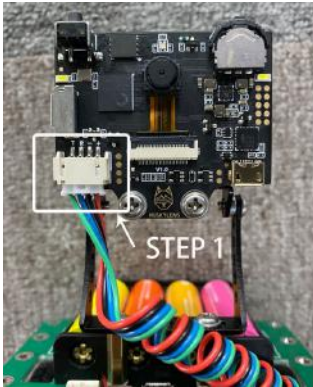


4. Bevestig de HuskyLens AI-camera

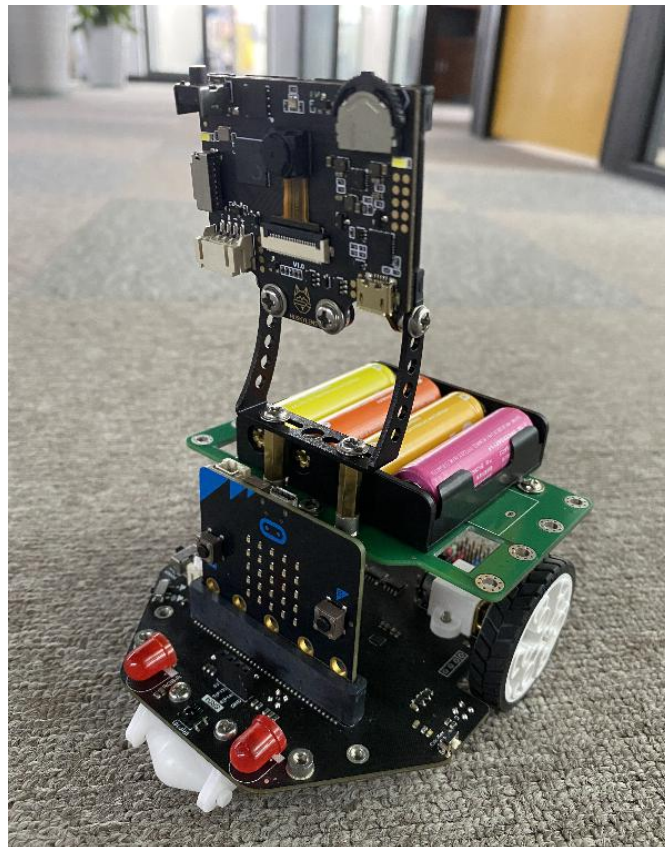




5. De AI-camera aansluiten op de robotinterface



6. De installatie van de camera is voltooid.





Optie 2 - Een Arduino Robot in elkaar zetten (alleen voor ervaren doe-het-zelvers)

Materiaallijst (componentenlijst)

- 1 x Set materialen voor een slimme auto

<https://botland.store/chassis-for-robots/7283-chassis-rectangle-2wd-2-wheel-robot-chassis-with-dc-motor-drive-5904422335649.html>

- 1 x Set truien

<https://botland.store/various-wires/1022-connecting-cables-male-male-65pcs.html>

- 1 x Batterijhouder (4 x AA)

<https://botland.store/battery-holders/173-battery-holder-4-x-aa-r6-5904422329389.html>

- Diverse schroeven en moeren

<https://botland.store/screws-and-nuts/637-screws-nuts-and-washers-set-330pcs-5410329304478.html>

- 1 met Arduino UNO (origineel of kloon)

<https://store.arduino.cc/collections/boards/products/arduino-uno-rev3>

<https://botland.store/arduino-compatible-boards-dfrobot/2683-dfduino-uno-v3-compatible-with-arduino.html>

- 1 x Arduino motorschild R3 (Arduino motorschild R3)

<https://store.arduino.cc/collections/shields/products/arduino-motor-shield-rev3>

- 1 x HuskyLens AI-camera

<https://store.arduino.cc/collections/dfrobot/products/gravity-huskylens-ai-machine-vision-sensor>

- 8 x AA-batterijen (probeer oplaadbare batterijen te gebruiken)

- Kabelbinders (als u draden en batterijhouders wilt bevestigen)

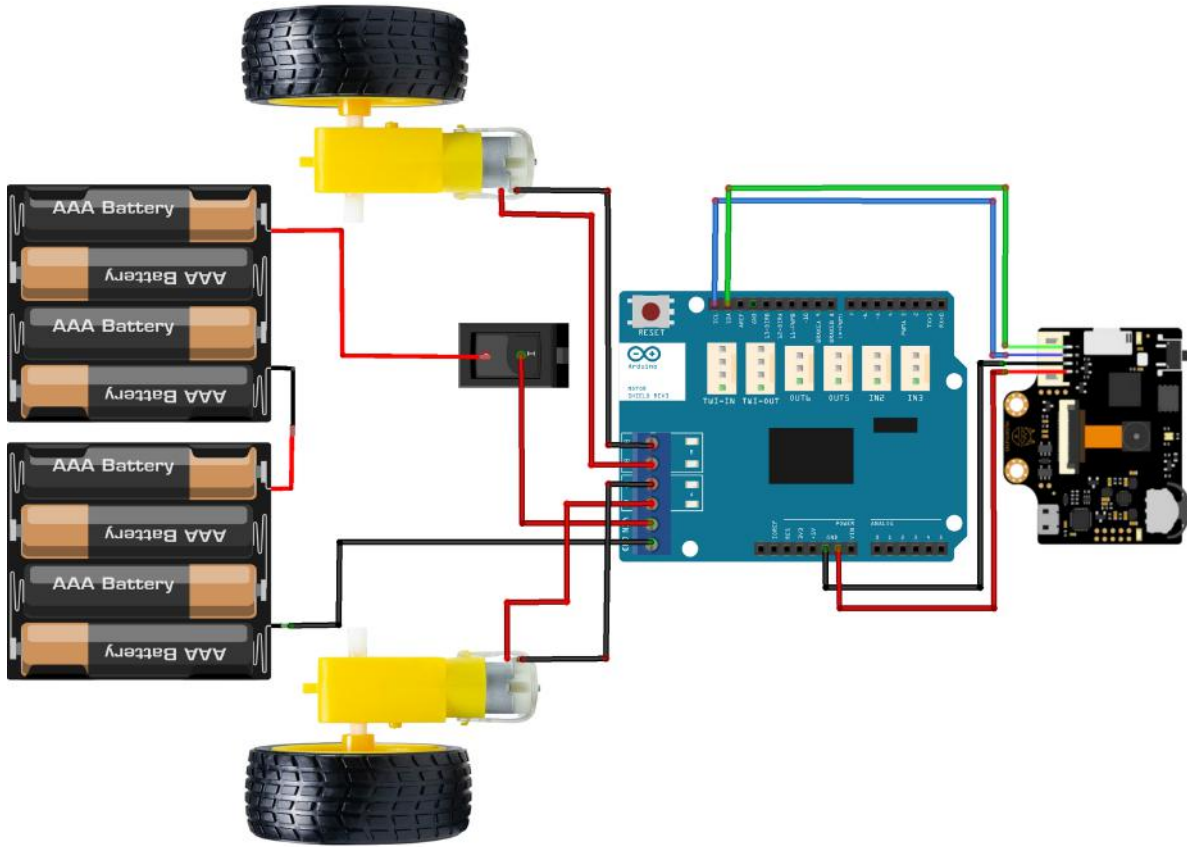
Noodzakelijk gereedschap

- Schroevendraaiers (u heeft een platte en een kruiskop nodig)
- Lemilica
- Isolatie tape of krimpkousen
- Snij tang

Schema

Hier is het bedradingsschema. Merk op dat we een dubbele batterijhouder in een serieschakeling zullen gebruiken, zodat elk van de batterijen in de houder is aangesloten (+ op - en - op +). 8 AA-batterijen zouden meer spanning moeten leveren om de consument van stroom te voorzien en te voorkomen dat de Arduino of HuskyLens reset wanneer de motoren beginnen te draaien. Als alternatief kunt u 3 18650-batterijen in serieschakeling gebruiken.



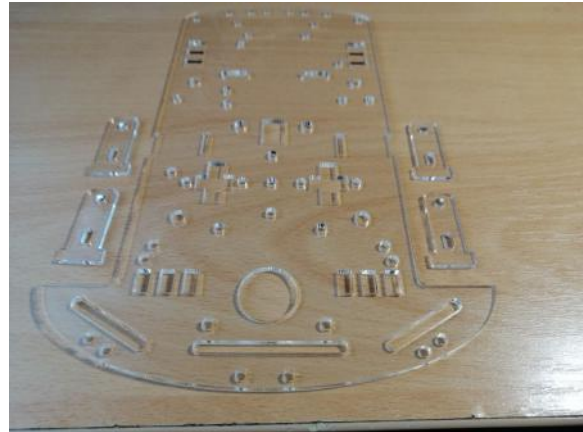


Smart Car Material Kit - Open de plastic zak met onderdelen

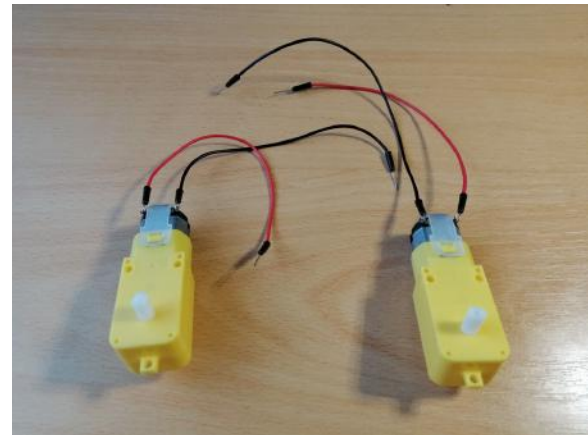




Verwijder de beschermfolie van de plaat en de motorsteun



Soldeer de draden (of jumpers) aan de motoren



Bekijk deze korte video om te zien hoe het werkt:

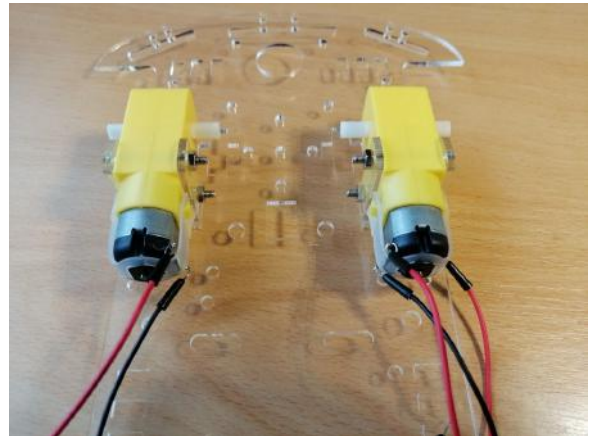
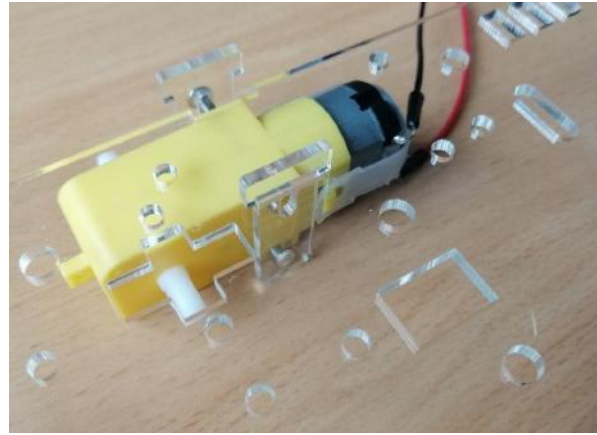
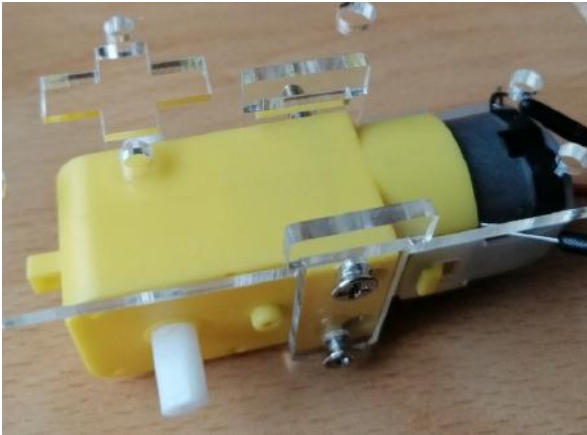
<https://www.youtube.com/watch?v=xSWKnnvGWBs>

Vraag je leraar of ouder om je hierbij te helpen als je geen eerdere ervaring of hulpmiddelen hebt.

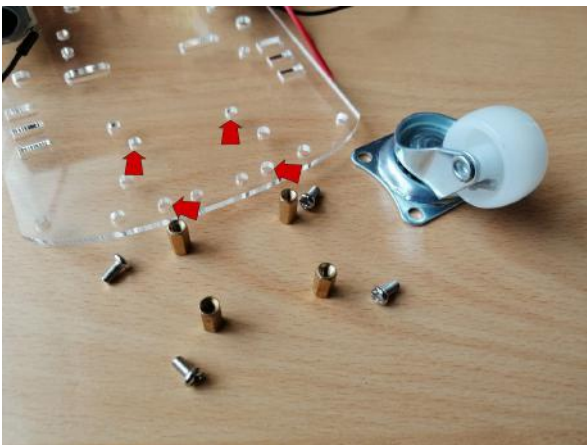


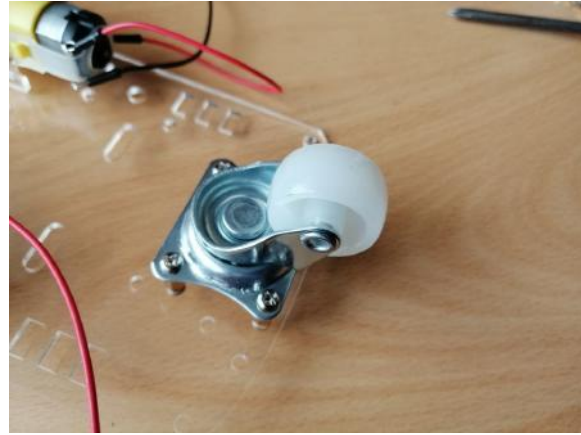
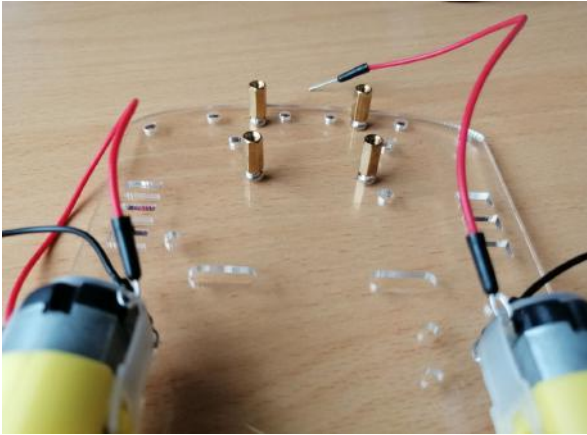


Bevestig de motoren aan de "T-beugels" met behulp van bouten en moeren

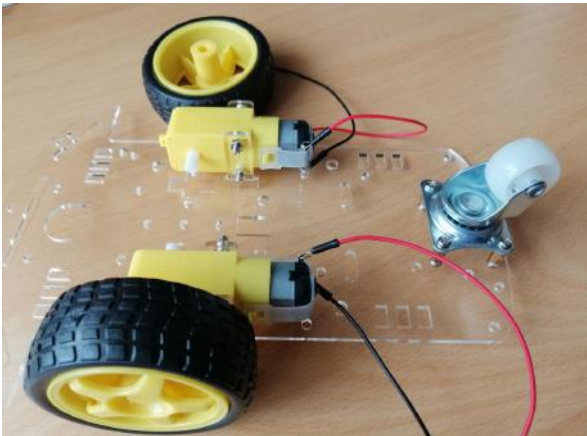


Zet het kleine (achter)wiel vast met behulp van schroeven en afstandsrings.

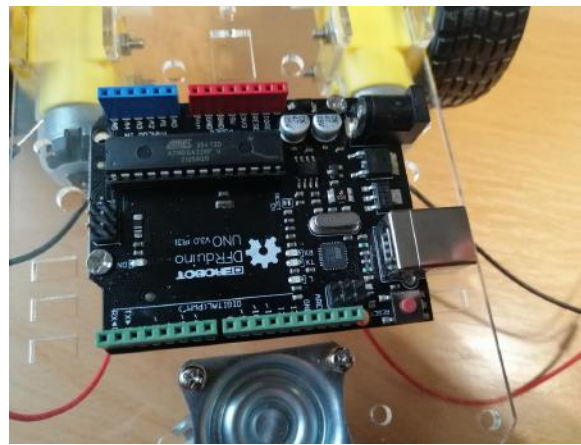
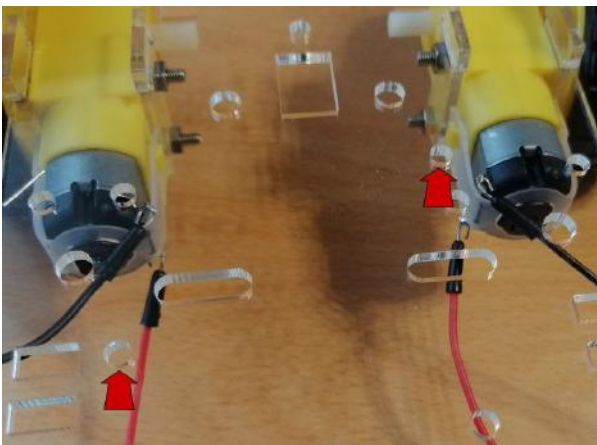


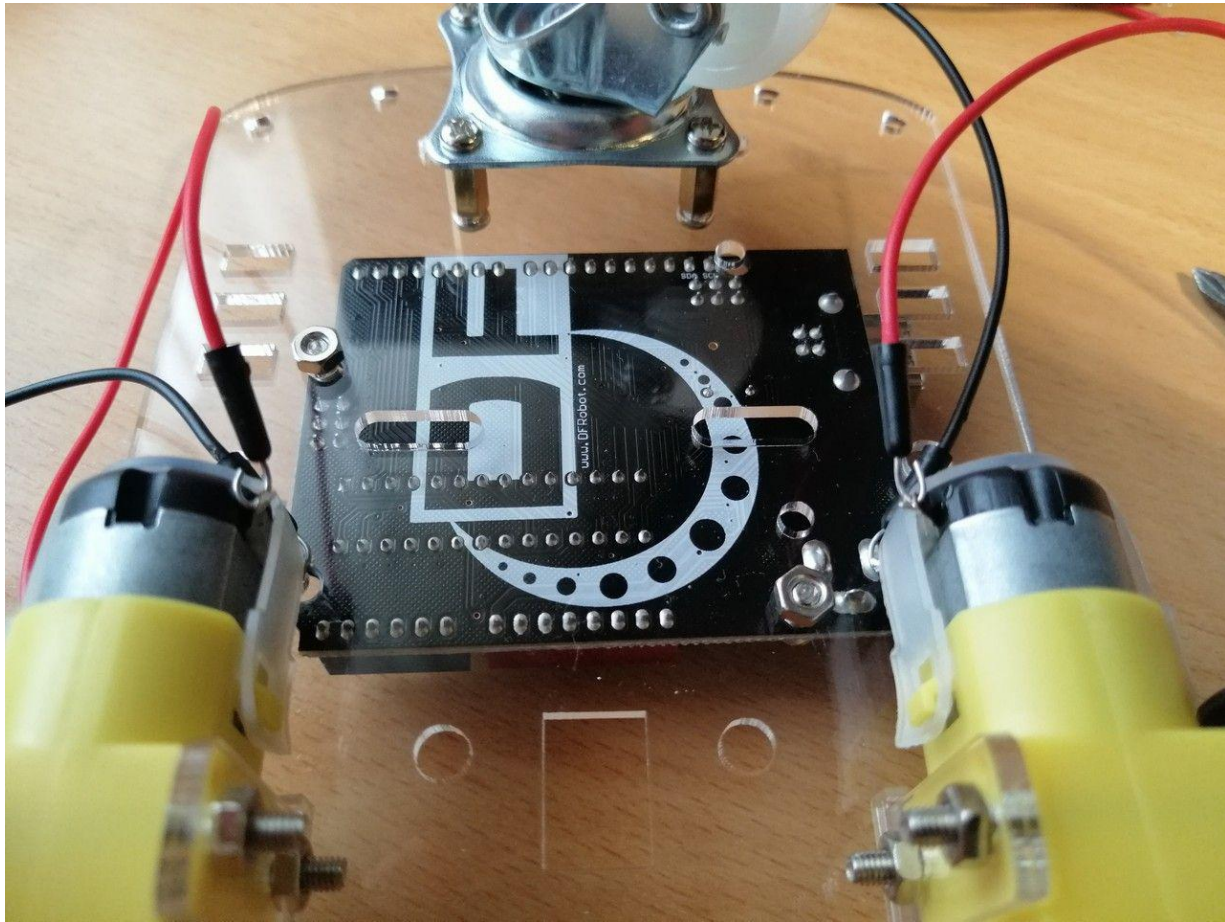


Bevestig de wielen aan de motorassen - kijk naar de vorm van de as en de vorm van de wielopening



Bevestig het Arduino UNO-bord aan de gemarkeerde gaten



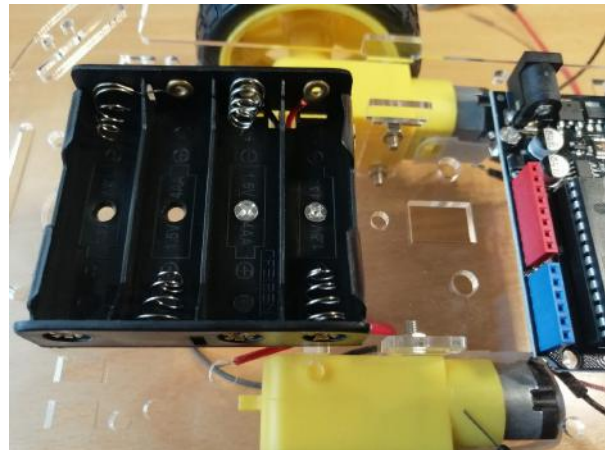
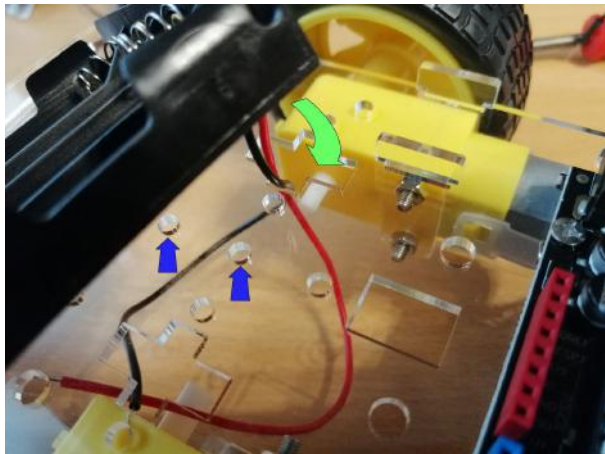


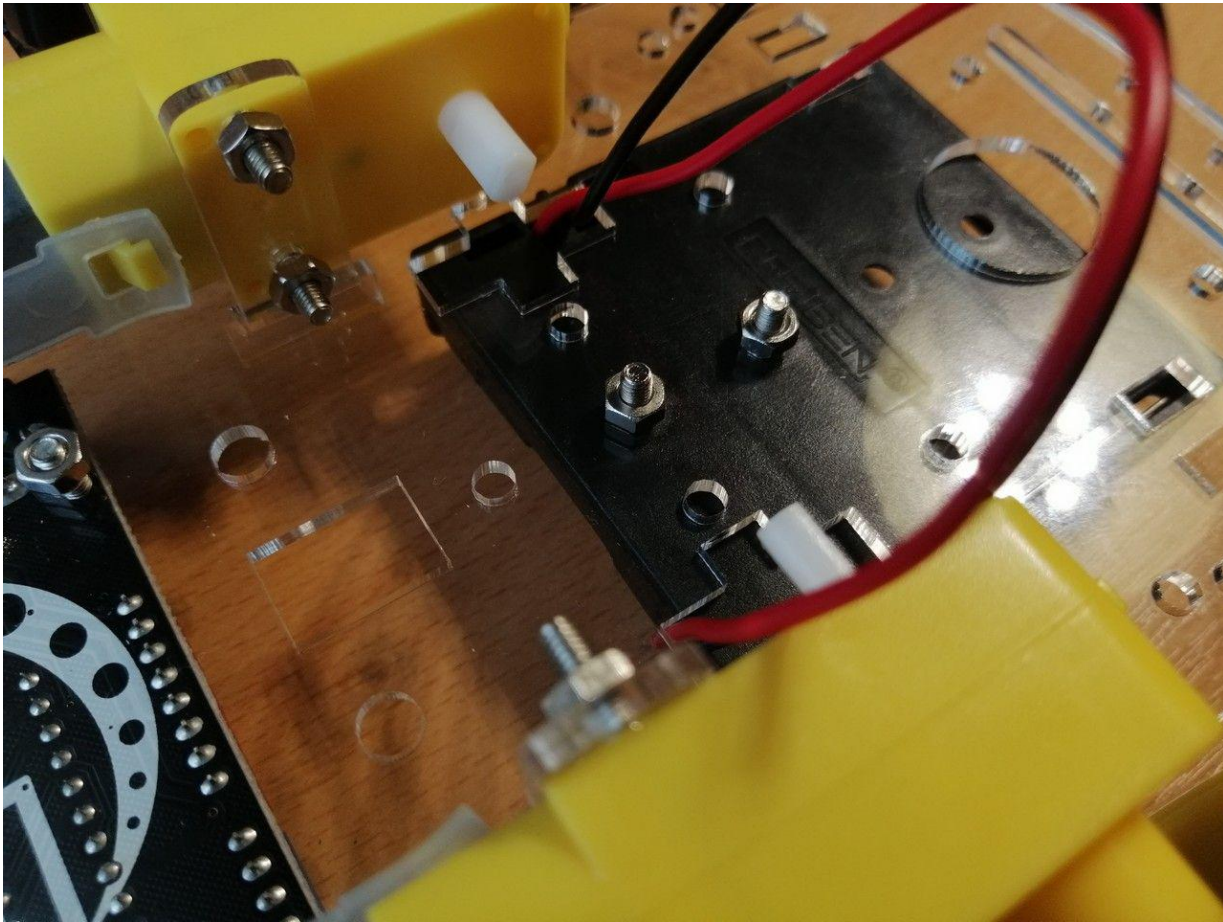
Je kunt enkele details in deze video zien, hoewel we een iets andere aanpak hanteren:

<https://www.youtube.com/watch?v=3a-bE1VlaU8>

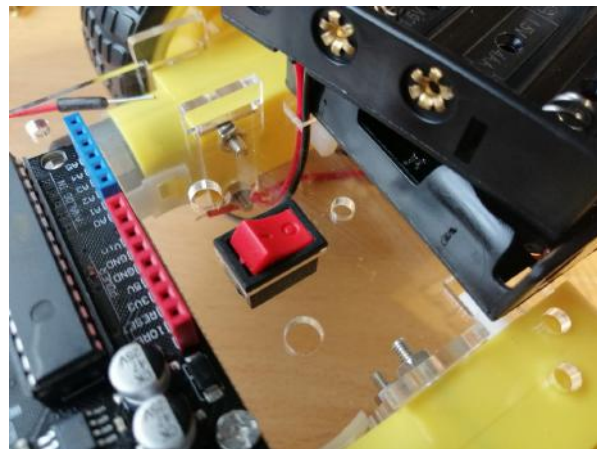
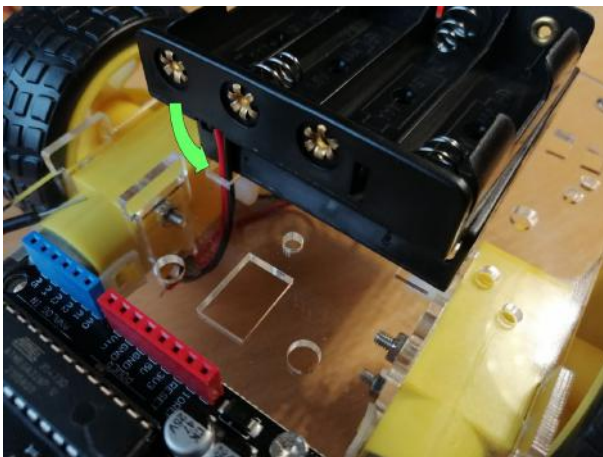
Stap 10) Bevestig de eerste (onderste) batterijhouder met behulp van de schroeven in de gemarkeerde gaten (blauwe pijlen)

Leid eerst de beugelkabels door het gat (gemarkeerd met een groene pijl)





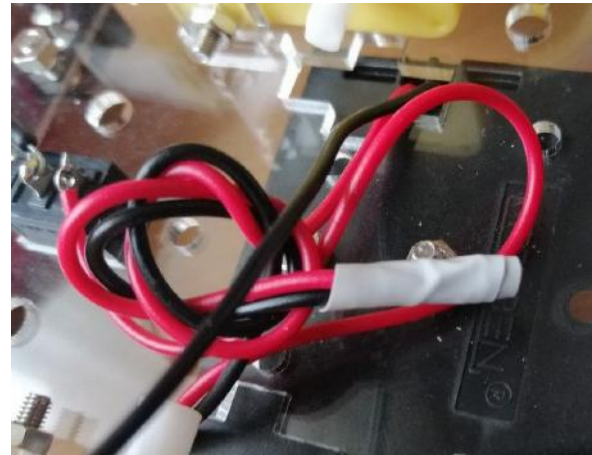
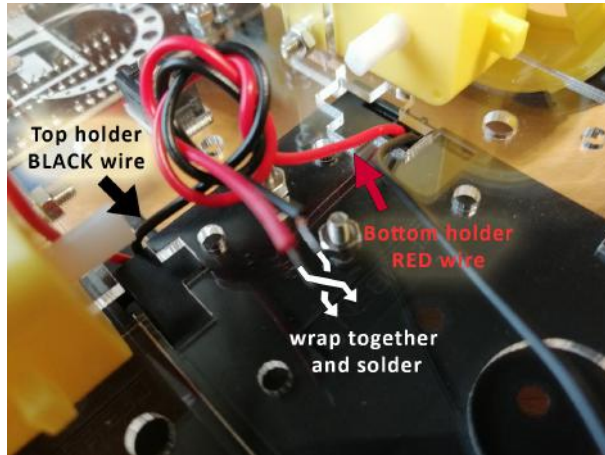
Trek de kabel van de tweede (bovenste) batterijhouder door het gemarkeerde gat (groene pijl)
Plaats de schakelaar in het rechthoekige gat tussen de Arduino UNO en de batterijhouder





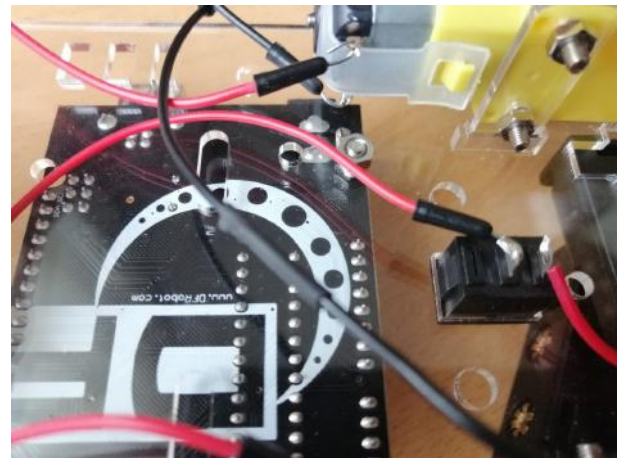
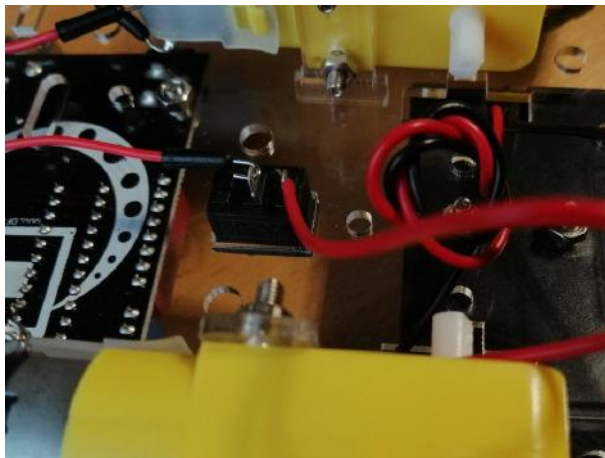
Sluit de rode draad van de onderste batterijhouder aan op de zwarte draad van de bovenste batterijhouder

Isoleer de verbinding met krimkous of isolatietape.



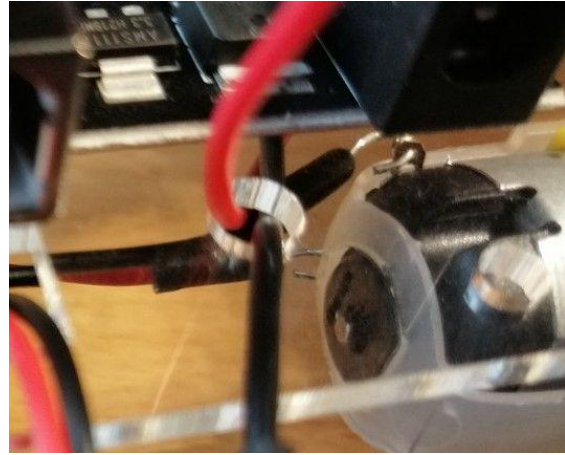
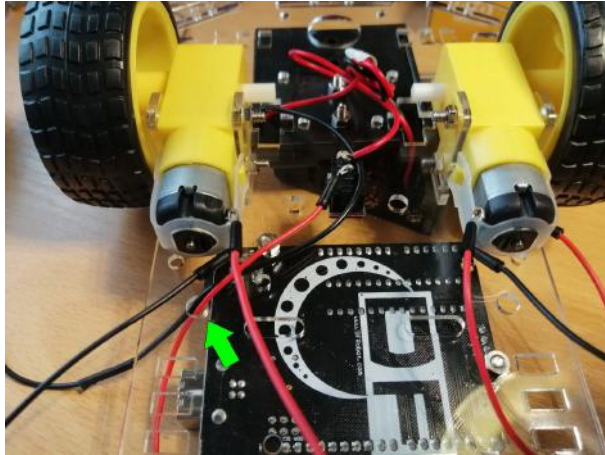
Soldeer de rode draad van de trapas naar de schakelaar

Soldeer daarna de rode jumper (of draad) aan het tweede contact van de schakelaar. Soldeer de zwarte jumper aan de trapas van de zwarte draad. Isoleer de verbinding met krimkous of isolatietape.



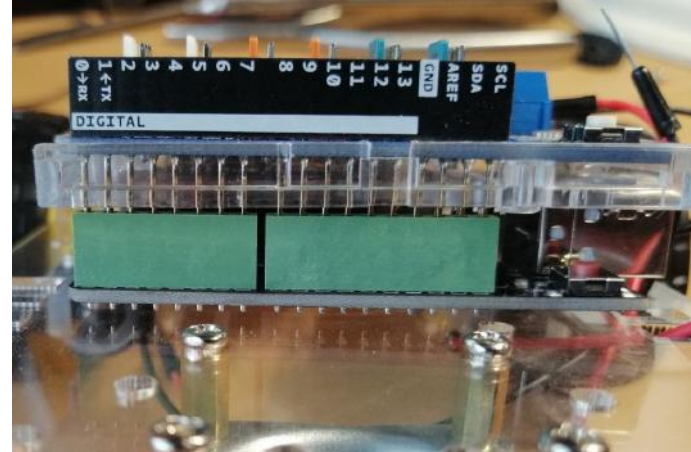
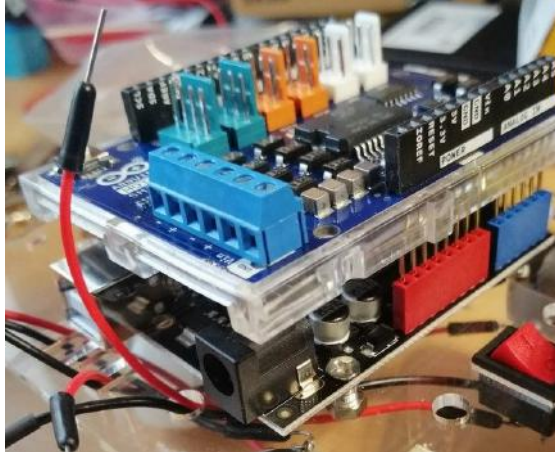


Trek de zwarte en rode voedingsdraden door het gemarkeerde gat (groene pijl)



Plaats het Arduino-motorschild in het Arduino UNO-bord

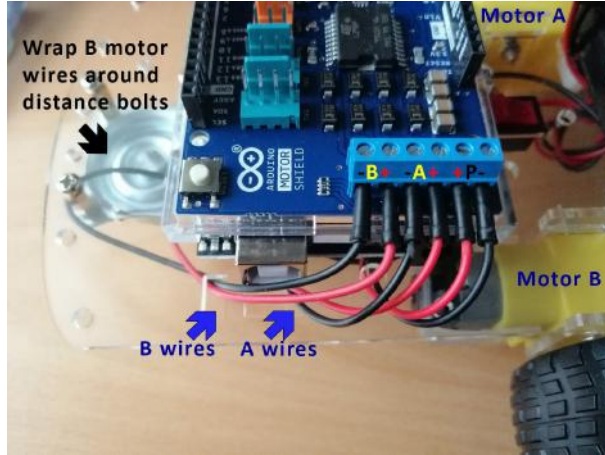
Lijn de poten (pinnetjes) van het schild en de Arduino UNO uit en duw het schild naar beneden.





Leid de motordraden door de gemarkeerde gaten.

Kort de uiteinden van de jumpers in met een kniptang en verbind ze met de afschermingsklemmen voor de motoren door de schroeven vast te draaien zoals aangegeven (- Motor B + - Motor A + + Power P -).



Open de HuskyLens AI-cameradoos en haal de materialenset eruit

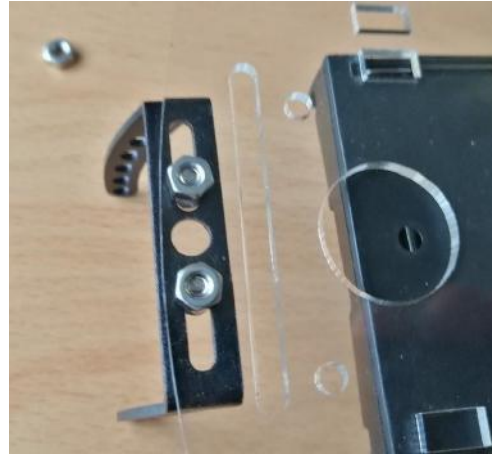


Bevestig de kleinere beugel aan de camera - gebruik de schroeven uit de set

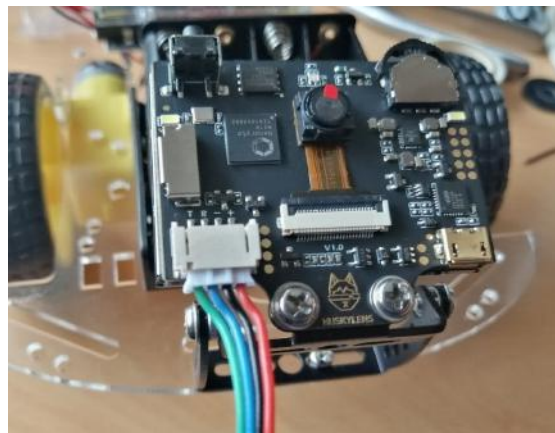




Bevestig de grotere camerabevestiging aan de robotbehuizing



Verbind de twee beugels met schroeven aan elkaar en sluit de camerakabel aan

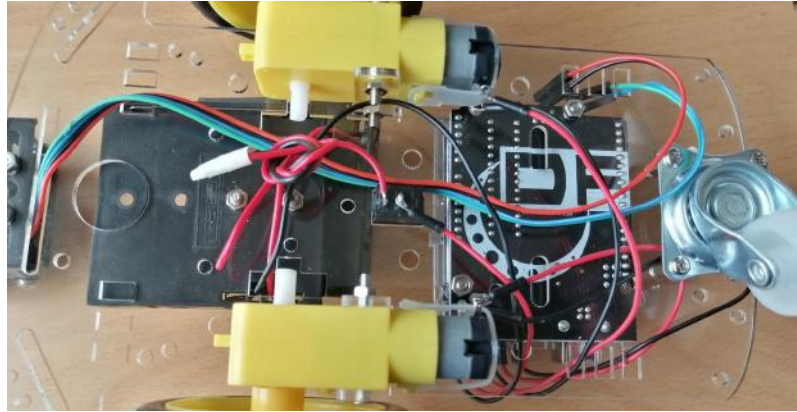


Trek de camerakabel door het gat

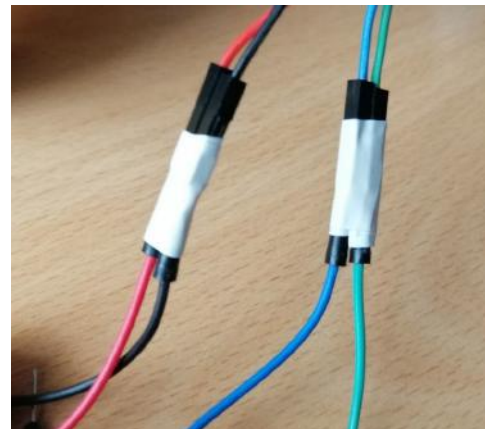
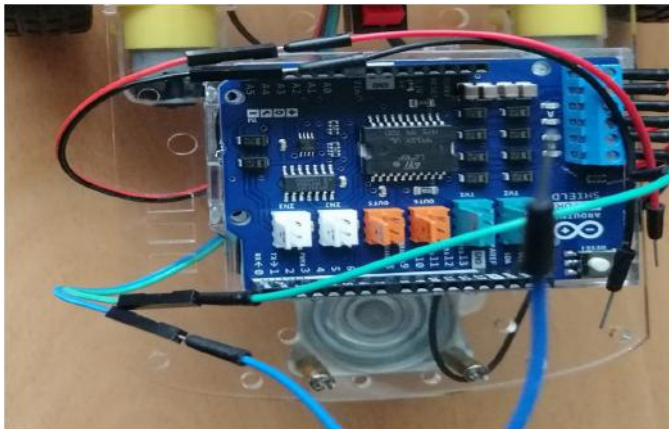




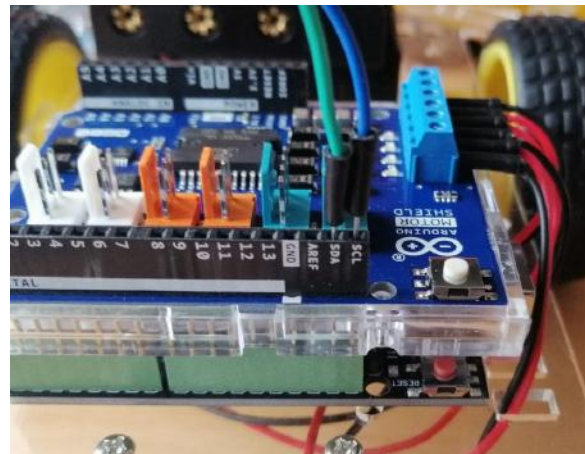
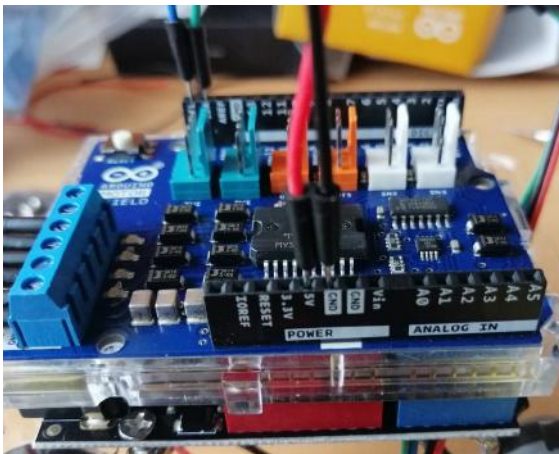
Draai de robot om en steek de camerakabel door de gaten zoals weergegeven in de onderstaande afbeelding



Sluit de jumpers aan op de camerakabel - gebruik jumpers met bijpassende kleuren
Zet de verbinding vast met isolatietape



Bevestig de voedingsjumpers en I2C-aansluitingen aan het motorschild
Rode jumper gaat naar 5V, zwart naar GND, blauw naar SCL en groen naar SDA-poort





Plaats de batterijen in beide houders

Gebruik 2 kabelbinders om de bovenste en onderste batterijhouders vast te zetten (niet te vast aandraaien - u zou de bovenste houder heen en weer moeten kunnen bewegen)



Zet de schakelaar aan - als je alles correct hebt aangesloten - gaan de Arduino UNO en de HuskyLens-camera aan.



<http://erasmus-artie.eu>

Think!



What do I know now?



What did I learn?



What do I want to know?

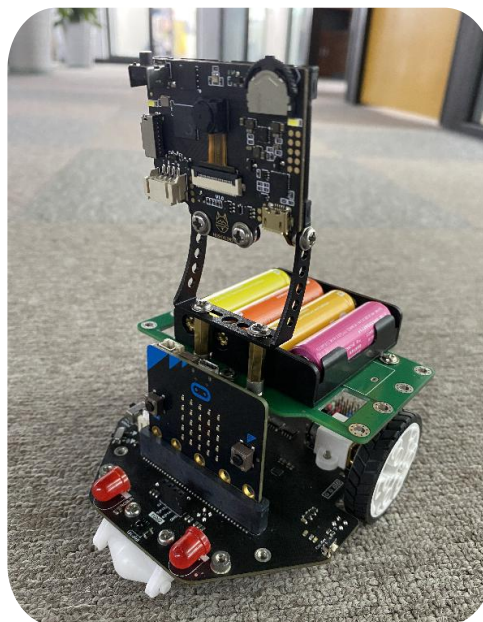


Een robot programmeren - object volgen

Robots zijn cool. In deze les laten we stapsgewijze, gemakkelijk te begrijpen voorbeelden zien van het programmeren van de bewegingen van een autonome mobiele robot.

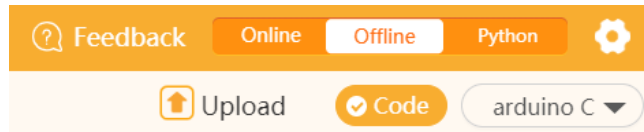
Doel:

Schrijf ons eerste programma waarmee we de robot gaan starten.

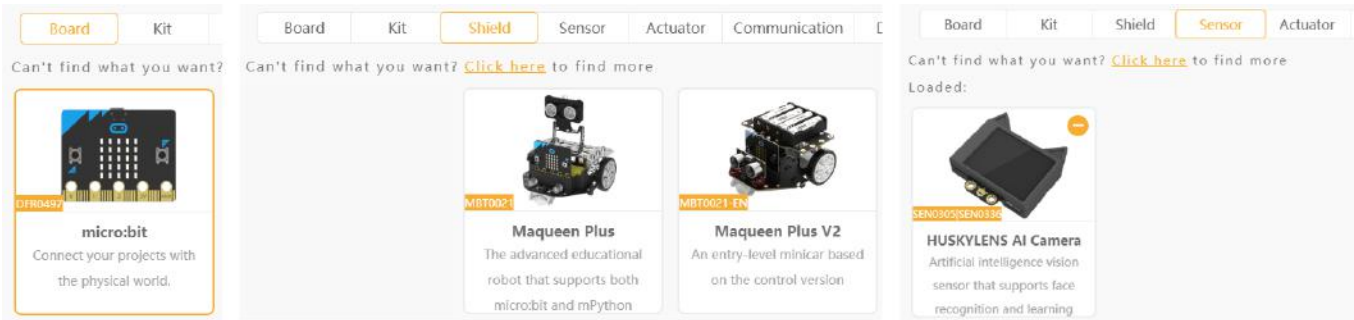




Stap 1 (beide opties): Ga naar: <http://mindplus.cc/download-en.html> en download de versie voor het besturingssysteem van uw computer. Installeer en voer Mind+ uit. Schakel na het starten over naar de offline-modus (offline).

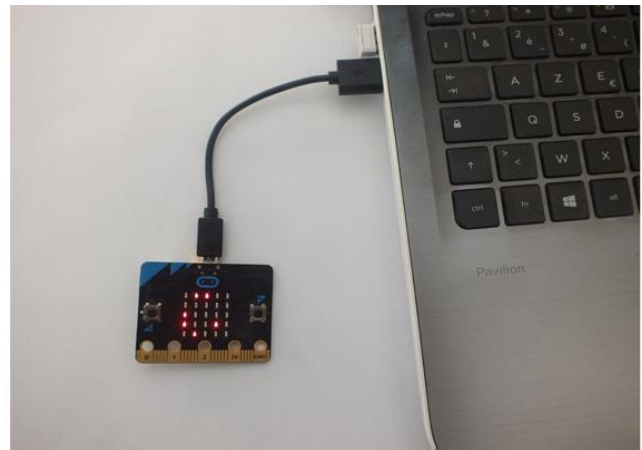


Als u met een Arduino werkt, gaat u direct naar stap 2.
Maqueen Plus



<http://erasmus-artie.eu>

Klik op **Extensions** en op de **Board** tab selecteren **micro:bit**, selecteer op het tabblad Schild Maqueen Plus of Maqueen Plus V2 en selecteer op het tabblad Sensor HUSKYLENS AI-camera. Klik op Terug en je programma is klaar om de geselecteerde modules te gebruiken.
Sluit de micro:bit aan op je computer via een micro-USB-kabel en de voedingsindicator op het bord gaat branden.





Klik op **Connect Device** en selecteer **micro:bit**. Installeer indien nodig apparaatstuurprogramma's.

Test de overdracht naar de micro:bit met deze blokken:

```

micro:bit starts
  set all motor direction rotate forward speed 200
  wait 1 seconds

```

Klik op **Upload**. De robot moet een seconde naar voren bewegen en dan stoppen. Probeer deze volgorde - de robot zou moeten bewegen zoals beschreven in de opmerkingen. (2 seconden vooruit, 2 seconden achteruit, 2 seconden linksom draaien en 2 seconden rechtsom draaien waarna de robot stopt)

```

micro:bit starts
  set all motor direction rotate forward speed 200
  wait 2 seconds
  set all motor direction rotate backward speed 200
  wait 2 seconds
  set left motor direction rotate backward speed 200
  set right motor direction rotate forward speed 200
  wait 2 seconds
  set left motor direction rotate forward speed 200
  set right motor direction rotate backward speed 200
  wait 2 seconds
  set all motor stop

```

Drive forward for 2 seconds

Drive backward for 2 seconds

Turn counter-clockwise for 2 seconds

Turn clockwise for 2 seconds

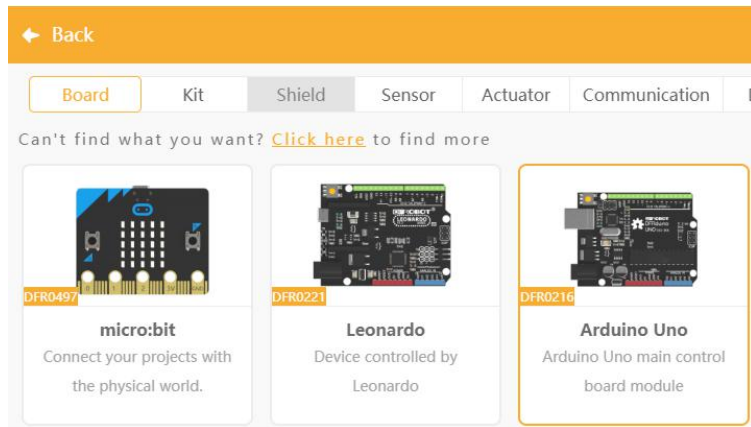


<http://erasmus-artie.eu>

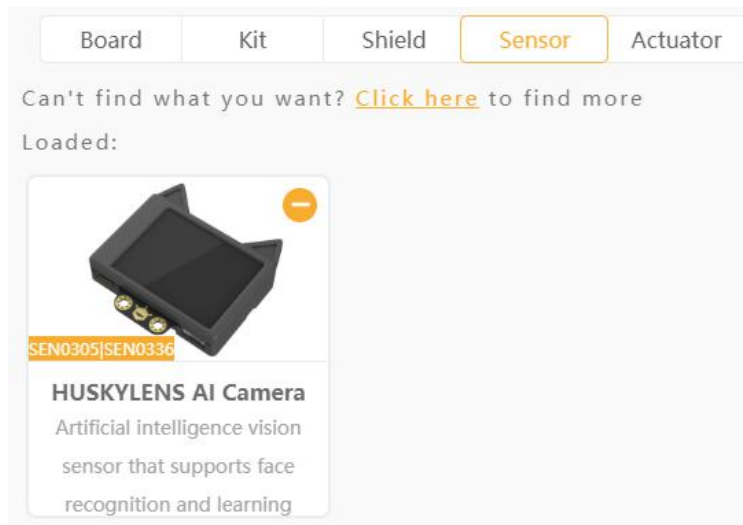
Klik op **Upload**. De robot moet bewegen zoals beschreven in de opmerkingen (gele blokjes). Probeer verschillende snelheden om je robot sneller of langzamer te laten gaan.



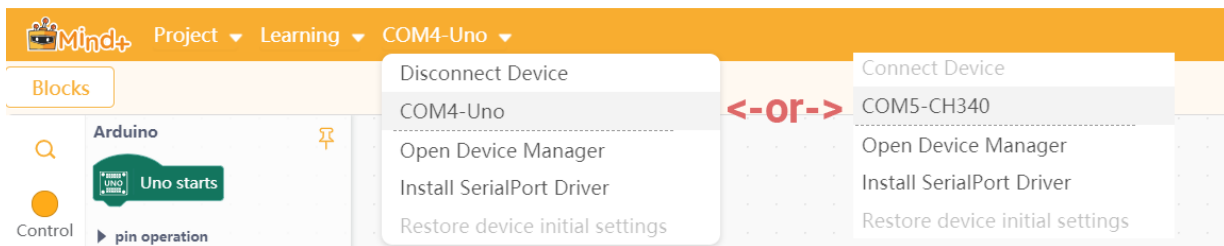
Stap 2: Open Extensies en selecteer **Board** - Arduino UNO



Stap 3: Ga naar het tabblad Sensor en selecteer de sensor - HUSKYLENS AI-camera



Stap 4: Eenmaal geselecteerd klikt u op <- **Back** en je bent klaar om Arduino- en Sensor-blokken te gebruiken. Laten we het testen om te zien of het werkt. Daarvoor moet u het apparaat aansluiten. Sluit uw Arduino UNO aan via een USB-kabel en selecteer COM X-Uno (of CH340), afhankelijk van de Arduino-fabrikant.





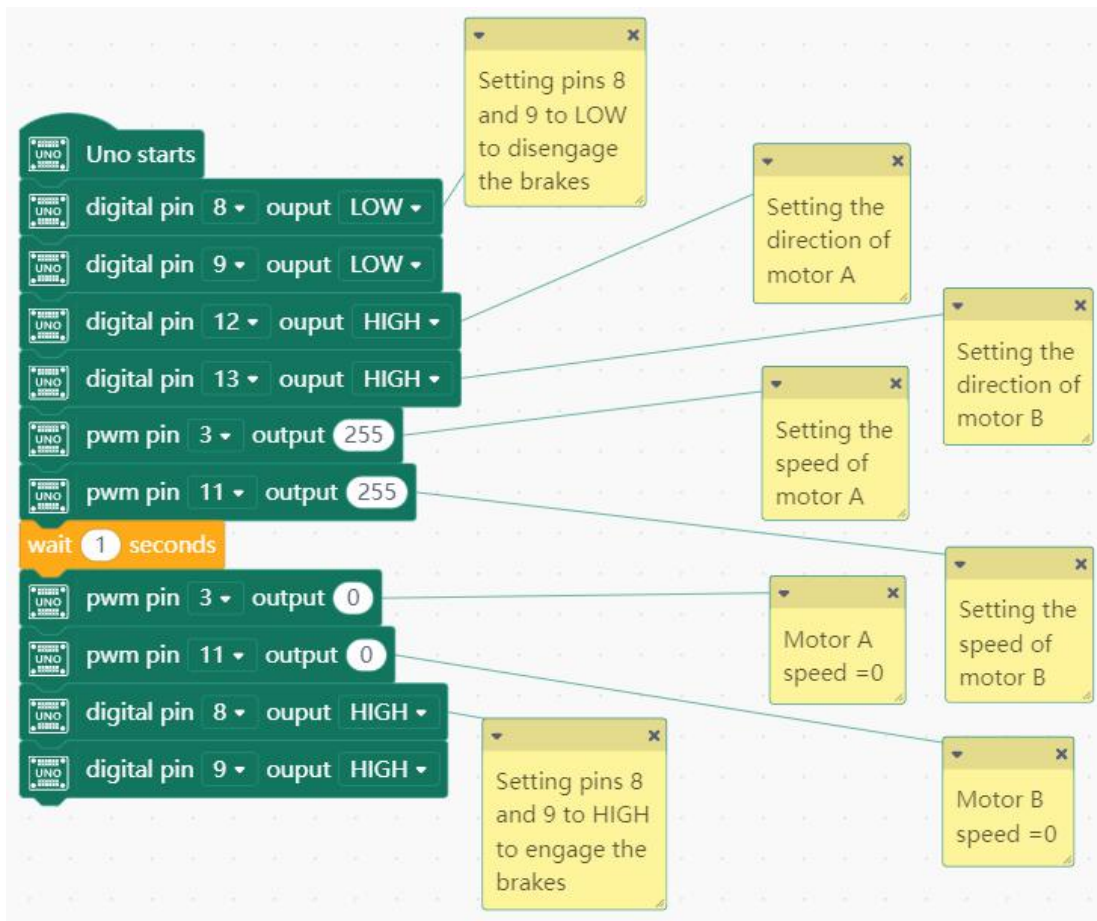
Basisbewegingen

Herinnert u zich deze tabel nog uit het hardware-implementatiescenario?

Function	Channel A	Channel B
Direction	Digital 12	Digital 13
Speed (PWM)	Digital 3	Digital 11
Brake	Digital 9	Digital 8
Current Sensing	Analog 0	Analog 1

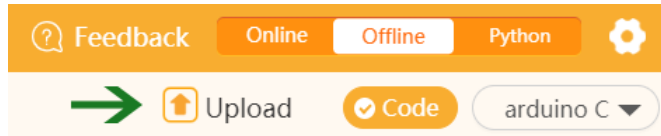
De twee motoren (A en B) zijn linkse en rechtse motoren. Digitale pinnen 12 en 13 worden gebruikt om de richting te veranderen (waarden HIGH - één richting en LOW - de tegenovergestelde richting), en PWM-pinnen 3 en 11 worden gebruikt om de snelheid aan te passen (waarden 0-255). Pinnen 9 en 8 schakelen remmen in/uit (HIGH - remmen aan, LOW - remmen uit).

Hieronder ziet u een voorbeeldcode met opmerkingen aan de rechterkant om u te helpen begrijpen hoe het werkt.





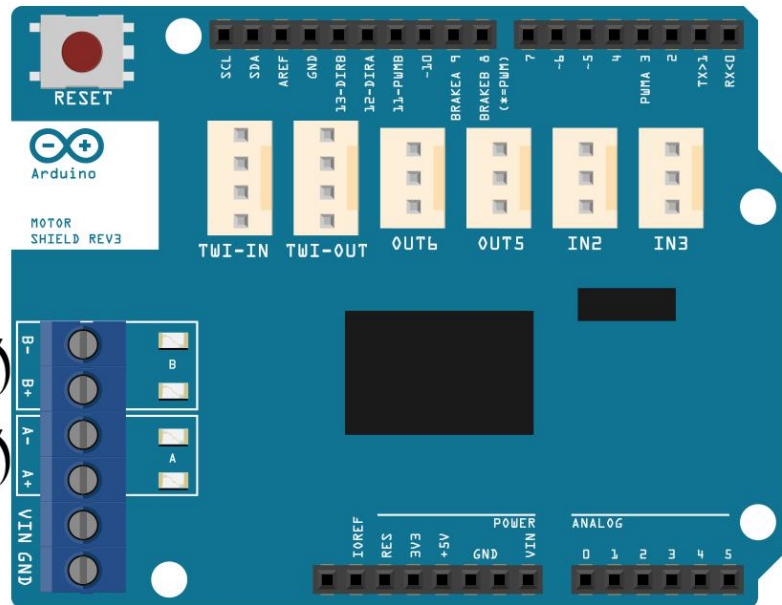
Stap 1a: Druk op Upload om deze code naar de Arduino UNO te uploaden.



Stap 2a: Wees voorzichtig, de robot begint direct te bewegen nadat de overdracht is voltooid! De robot moet een seconde naar voren bewegen en dan stoppen.

Problemen oplossen - als de robot:

- achteruit rijden - vervang de rode en zwarte draden in de connectoren van beide motoren.
- draait met de klok mee - verwissel de rode en zwarte draden in de B-motorconnectoren.
- draait tegen de klok in - verwissel de rode en zwarte draden in de A-motorconnectoren.



switch wires for motor B ↻

switch wires for motor A ↻



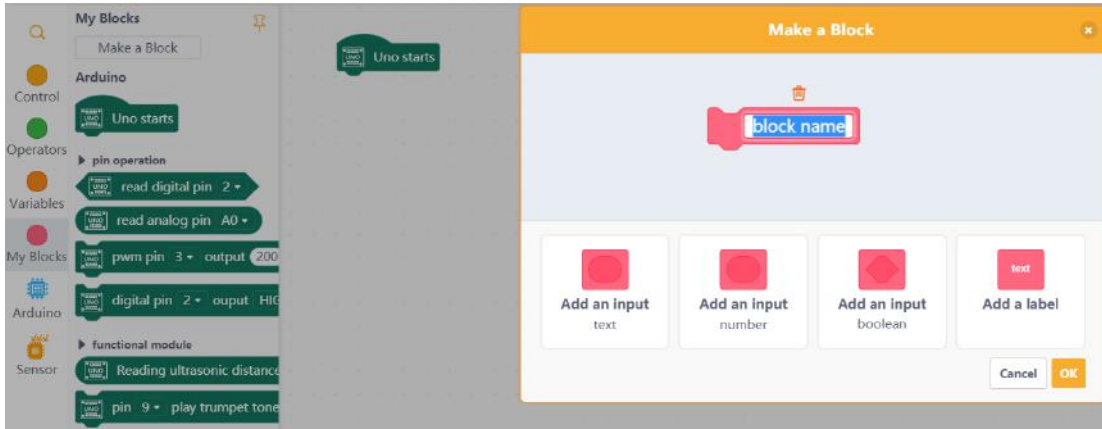
<http://erasmus-artie.eu>

Je zou nu je robot vooruit moeten laten rijden met een HOGE status op pinnen 12 en 13. Er is ook een eenvoudigere manier om de beweging te programmeren - we gebruiken onze aangepaste blokken in plaats van een hele reeks blokken te herhalen om de pinnen te besturen.

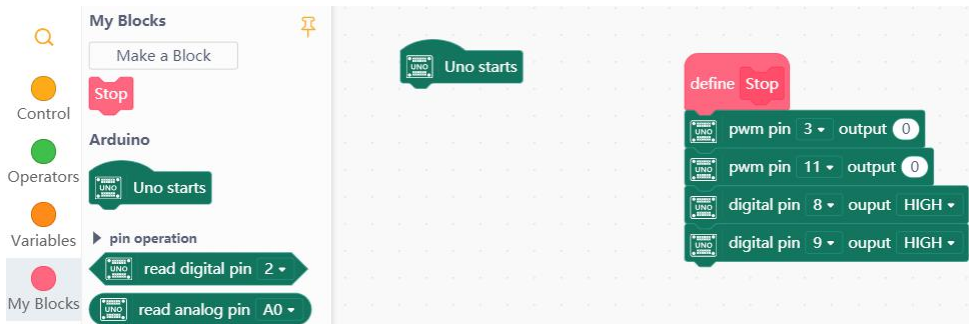
Stap 1b: Klik op de **My Blocks** groep (rood).

Stap 2b: Klik op de **Make a Block** knop en wijzigen "block name" naar "Stop".

Stap 3b: Klik op OK.



Je hebt je eerste blok gemaakt, maar het is op dit moment nergens mee verbonden, dus we moeten het eerst definiëren.



<http://erasmus-artie.eu>

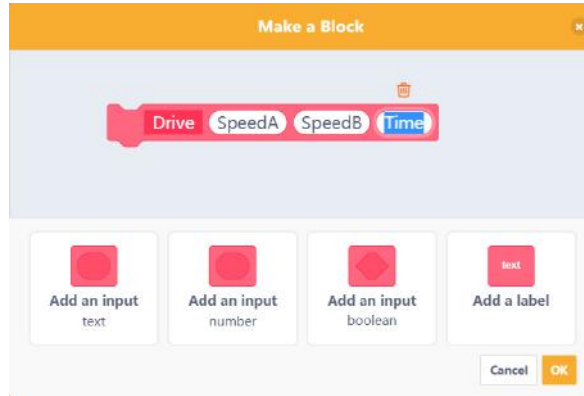
Het stopblok bevat 4 blokken. De eerste twee blokken zetten de snelheid van beide motoren op 0, en de laatste twee zijn blokken die, met een HOGE toestand, de motoren afremmen. Als je het wilt gebruiken om de motoren te laten uitlopen zonder uit te lopen, stel dan gewoon de status van digitale pinnen 8 en 9 in op laag (LOW).

Nu is het tijd om een invoerblok te maken met 3 numerieke parameters die zowel het motortoerental als de duur bevatten. Klik nogmaals op de knop Create Block, klik vervolgens op Add an input - number - 3 keer en je zou dit moeten krijgen:





Wijzig de naam van het blok in **Drive**, verander de eerste vlotter in **SpeedA**, de tweede vlotter naar **SpeedB**, en de derde vlotter naar **Time**, klik dan op de **OK** button.



Het basisidee is om de SpeedA- en SpeedB-waarden (aanvaardbaar bereik: -255 tot 255) te krijgen, vervolgens te controleren of een van beide getallen negatief is (of beide) en zo ja - draai de richting om door de waarde voor de overeenkomstige pin in te stellen. U moet de absolute waarde op de PWM-pin gebruiken om de motor te starten.

The code block structure is as follows:

- define** Drive SpeedA SpeedB Time
- digital pin** 8 **output** LOW
- digital pin** 9 **output** LOW
- if** SpeedA > 0 **then**
 - digital pin** 12 **output** HIGH
- else**
 - digital pin** 12 **output** LOW
- if** SpeedB > 0 **then**
 - digital pin** 13 **output** HIGH
- else**
 - digital pin** 13 **output** LOW
- pwm pin** 3 **output** abs of SpeedA
- pwm pin** 11 **output** abs of SpeedB
- wait** Time seconds

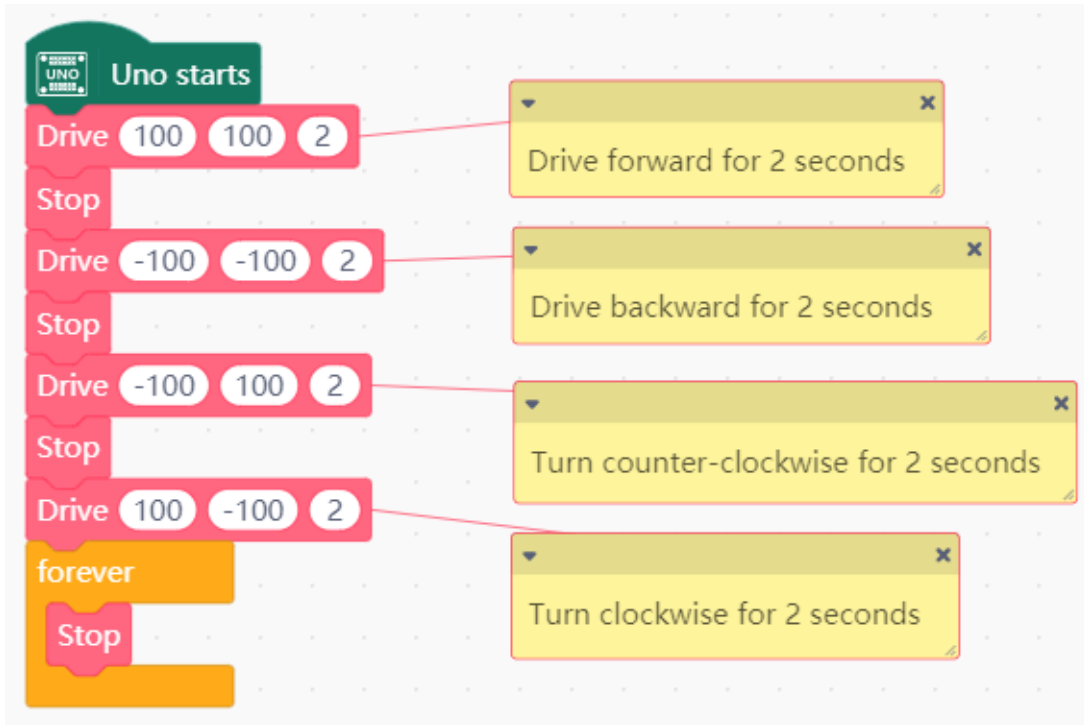
Callouts explain the logic:

- Release the brakes
- SpeedA is positive - drive forward
- SpeedA is negative - drive backward
- SpeedB is positive - drive forward
- SpeedB is negative - drive backward
- PWM value can't be negative, so we take an absolute value
- Time duration of driving sequence





Dus laten we proberen onze robot te besturen met de volgende blokken.



<http://erasmus-artie.eu>

Dus laten we proberen onze robot te besturen met de volgende blokken. Wijzig de waarden in de **Drive** block, upload dan je programma naar de Arduino UNO en kijk hoe snel je robot beweegt.

U bent nu klaar om uw ARTIEbot te gebruiken in complexere voorbeelden, waaronder die met de HuskyLens-camera.

CONCLUSIE

Laten we eerlijk zijn, robots zijn cool. Ze kunnen op een dag de wereld regeren, en hopelijk zullen ze op dat moment genade hebben met hun nietige scheppers (de robotingenieurs) en ons helpen een hemel in de ruimte te bouwen. Het was natuurlijk een grap, maar hij weet het.

Think!



What do I know now?



What did I learn?



What do I want to know?



Een robot programmeren - object volgen

In onze vorige les hebben we geleerd hoe je een robot kunt laten bewegen.

Check bij je leerlingen of ze alles begrepen hebben van de vorige les en of ze klaar zijn voor de volgende stap.

Laten we onze robots nu naar een specifiek object verplaatsen. Eerst moeten we dat object detecteren en volgen. Het volgen van een bewegend object vereist zowel visuele objectvolgtechnologie als handmatige bediening.

Hoofdidée:

Hoe bereid je een robot voor op het volgen van objecten?





Object volgen is een belangrijke opdracht in computervisie. Het verwijst naar het proces van het continu afleiden van de staat van objecten in videosequenties. Het beeld wordt verzameld door een enkele camera en de beeldinformatie wordt verzonden naar een microcontroller. Na analyse en verwerking wordt de relatieve positie van het bewegende object berekend. Tegelijkertijd wordt de robotdragende camera bestuurd om het object in realtime te roteren en te volgen.

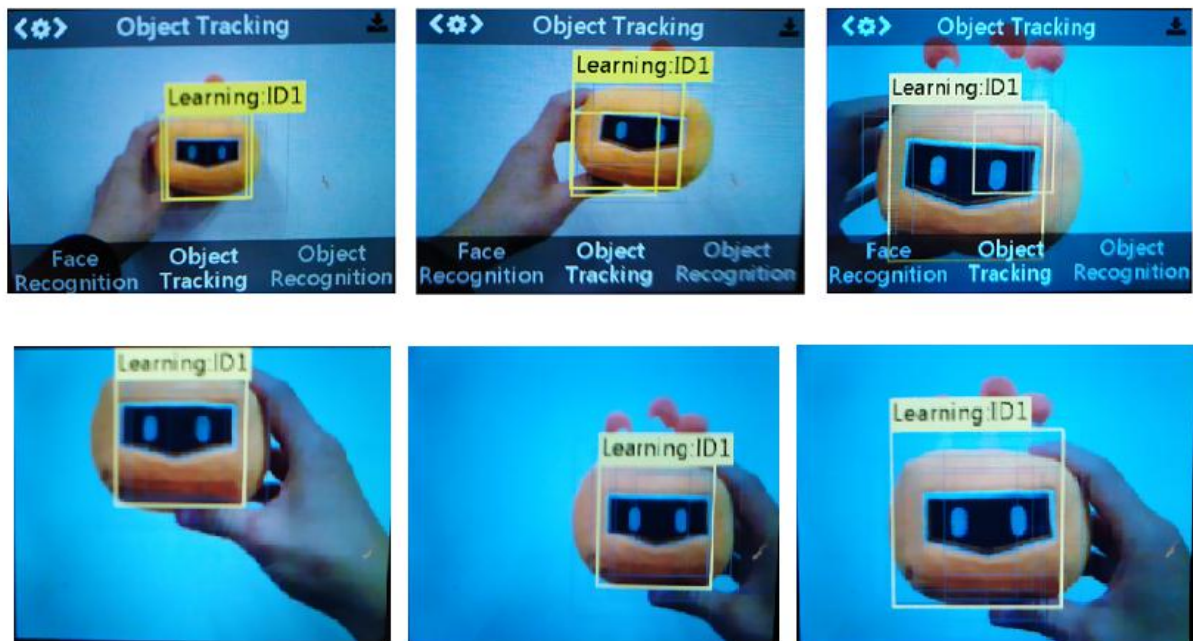
Wanneer het objectvolgsysteem de volgfunctie uitvoert, is het hoofdzakelijk verdeeld in 4 stappen:

- Object herkenning
- Object volgen
- Analyse van objectbewegingen
- Het besturen van de robot (of een ander systeem) met camera

Objectherkenning - Leren

Sluit de micro:bit of Arduino UNO met HuskyLens-camera aan op je laptop of desktopcomputer. Richt de HuskyLens op het doelobject, pas de afstand aan tot het object zich in het oranje kader in het midden van het scherm bevindt. Het is ook acceptabel dat slechts een deel van het object in de doos zit, maar binnen verschillende kenmerken.

Druk vervolgens lang op de "learning button" om het object vanuit verschillende hoeken en afstanden te leren. Tijdens het leerproces wordt het oranje vak met de woorden "Learning: ID1" op het scherm weergegeven.





Wanneer HuskyLens het object onder verschillende hoeken en afstanden kan volgen, laat dan de "learning button" los om het leren te voltooien. Als er geen oranje vakje in het midden van het scherm staat, betekent dit dat de HuskyLens al een object heeft geleerd. Als u een ander object wilt volgen, selecteert u "Forget Learned Object" en leert u opnieuw.

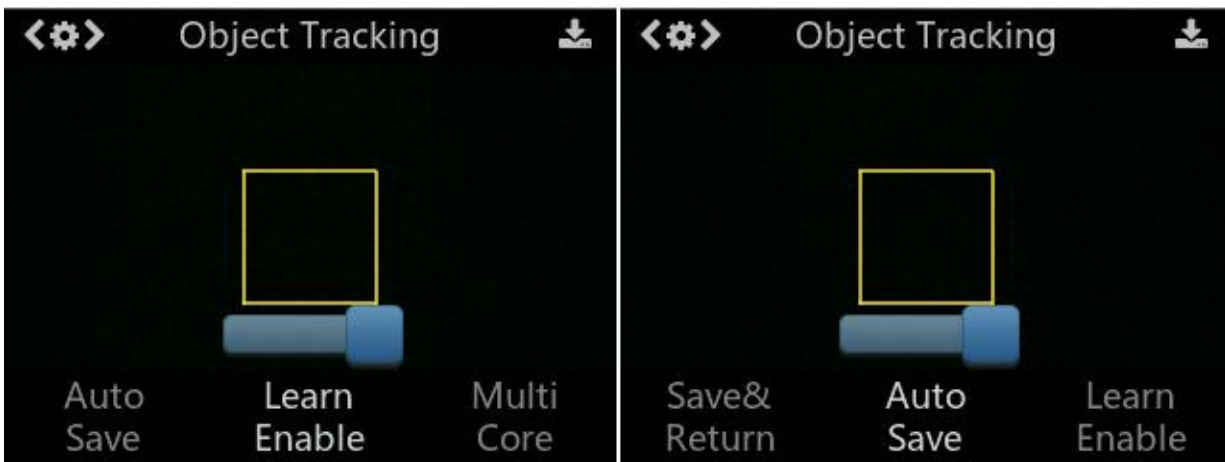
Onder de functie voor het volgen van objecten kan HuskyLens blijven leren, dat wil zeggen, zolang de camera het geleerde object ziet, zal hij de huidige status van het object blijven leren, wat bevorderlijk is voor het vastleggen van dynamische objecten. Bedieningsmethode: druk lang op de functieknop om de parameterinstelling van de objectvolgfunctie in te voeren.



f

Duw de functietoets naar rechts om naar de parameter Learn Enable te gaan, druk vervolgens kort op de functietoets om deze te selecteren en duw hem vervolgens naar rechts om in te schakelen **Learn Enable**, dat wil zeggen, de schuifregelaar op de balk staat in de juiste positie. Druk vervolgens nogmaals kort op de functietoets om deze parameter te bevestigen. Verlaat het submenu door kort op de te drukken **Save&Return** item om de wijzigingen op te slaan.

<http://erasmus-artie.eu>



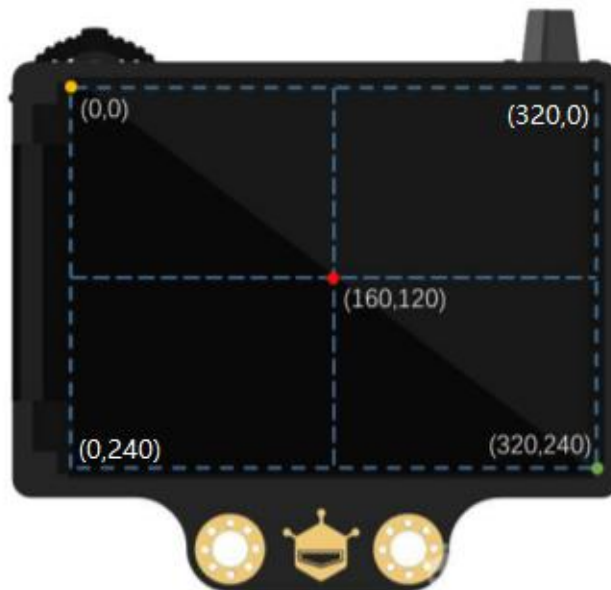


Bij het herstarten van HuskyLens wordt het laatst geleerde object niet standaard opgeslagen en kunt u de **Auto Save** schakelen zoals eerder uitgelegd.

Hoe te werken: Hetzelfde als hierboven, verplaats na het invoeren van de parameterinstelling de schuifregelaar in het item Automatisch opslaan naar de juiste positie. Op die manier hoeft je het object maar één keer te leren. Als u de camera opnieuw start, wordt het object dat u het laatst hebt geleerd, opgeslagen.

Object volgen

De schermresolutie van de HuskyLens-sensor is 320*240, zoals weergegeven in de volgende afbeelding.



De coördinaten van het objectmiddenpunt die via het programma zijn verkregen, liggen ook binnen dit bereik. Als de verkregen coördinaatwaarden bijvoorbeeld (160, 120) zijn, bevindt het object dat wordt gevolgd zich in het midden van het scherm.

"X-coördinaten" en "Y-coördinaten" verwijzen naar de positie van het middenpunt van de doos in de schermcoördinaat. "Objectbreedte" en "Objecthoogte" verwijzen naar de grootte van het frame.

Onder de objectvolgfunctie is het frame vierkant, dus de breedte en hoogte zijn gelijk.

Test het volgen van objecten - Optie 1 (Maqueen Plus/HuskyLens)

Open je Mind+ en laadextensies voor werk met Maqueen Plus en HuskyLens-camera's.

Gebruik deze code:



```

micro:bit starts
HuskyLens initialize pin until success
HuskyLens switch algorithm to Object tracking
set motor All stop

forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result? then
    if HuskyLens check if ID 1 frame is on screen from the result? then
      serial output join "X: " HuskyLens get X center of ID 1 frame from the result in string, No-Wrap
      serial output join ", Y: " HuskyLens get Y center of ID 1 frame from the result in string, No-Wrap
      serial output join ", W: " HuskyLens get width of ID 1 frame from the result in string, No-Wrap
      serial output join ", H: " HuskyLens get height of ID 1 frame from the result in string, Wrap
    wait 1 seconds
  
```

Spring naar **Checking results**

Test het volgen van objecten - Optie 2 (Arduino UNO/HuskyLens)

Open je Mind+ en laadextensies voor werk met Arduino UNO en HuskyLens-camera. Gebruik deze code met Arduino/HuskyLens:

```

Uno starts
HuskyLens initialize pin until success
HuskyLens switch algorithm to Object tracking

forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result? then
    if HuskyLens check if ID 1 frame is on screen from the result? then
      serial output join "X: " HuskyLens get X center of the ID 1 No. 1 frame from the result in string, No-Wrap
      serial output join ", Y: " HuskyLens get Y center of the ID 1 No. 1 frame from the result in string, No-Wrap
      serial output join ", W: " HuskyLens get width of the ID 1 No. 1 frame from the result in string, No-Wrap
      serial output join ", H: " HuskyLens get height of the ID 1 No. 1 frame from the result in string, Wrap
    wait 0.5 seconds
  
```



<http://erasmus-artie.eu>

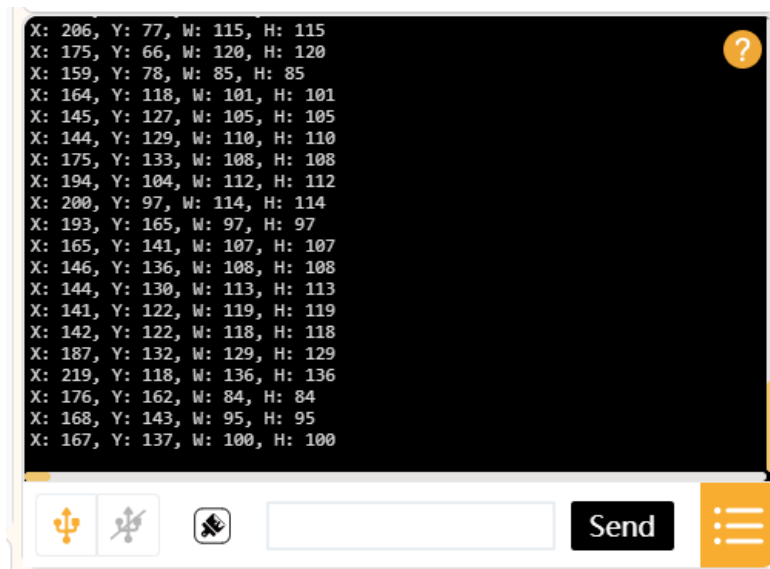


Resultaten controleren op seriële monitor (beide opties)

Open de seriële monitor door op het USB-pictogram rechtsonder in het Mind+-scherm te klikken.



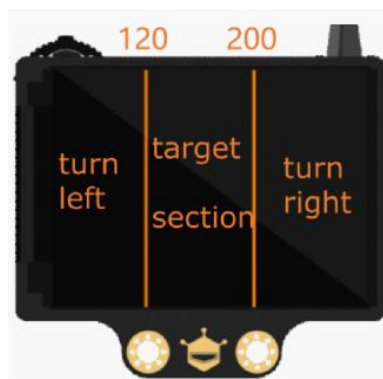
Probeer het object naar links en rechts te bewegen om de numerieke verandering van het X-centrum waar te nemen. Beweeg het object op en neer om de numerieke verandering van het Y-centrum te observeren. Beweeg het object heen en weer om de numerieke verandering van breedte en hoogte te observeren.



<http://erasmus-artie.eu>

Bewegingsanalyse van objecten

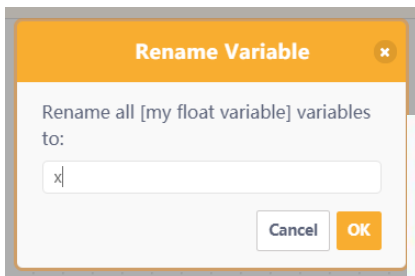
Zoals te zien is in de volgende afbeelding, is het scherm verdeeld in 3 secties volgens de X-as van het coördinatensysteem van het camerascherm, en het middelste gedeelte is ons doelgedeelte.





Wanneer de camera continu de toestand van het doelobject op de foto detecteert, is het X-centrum 120-200, wat betekent dat het doel zich in het midden van het gezichtsveld bevindt en de robot zijn positie niet hoeft aan te passen; het X-centrum is 0-120, onze robot moet zich aanpassen door naar rechts te draaien; het X-centrum is 200-320, ARTIEbot moet naar links draaien om aan te passen.

Nu is het tijd om het grootste deel van de code te schrijven om de robot naar het object te draaien. **Beide opties-** Hernoemen **my float variable** naar **x**. Klik met de rechtermuisknop op variabele -> Hernoem numerieke variabele.



Optie 1 - volg het object met Maqueen Plus

Gebruik en configureer blokken zoals op de foto:

```

micro:bit starts
  HuskyLens initialize pin until success
  HuskyLens switch algorithm to Object tracking
  set motor All stop
  forever
    HuskyLens request data once and save into the result
    if HuskyLens check if ID 1 is learned from the result? then
      if HuskyLens check if ID 1 frame is on screen from the result? then
        set x to HuskyLens get X center of ID 1 frame from the result
        if x >= 120 and x <= 200 then
          set motor All stop
        else if x > 0 and x < 120 then
          set motor Left move by 20 speed Forward
          set motor Right move by 80 speed Forward
        else if x > 200 and x < 320 then
          set motor Left move by 80 speed Forward
          set motor Right move by 20 speed Forward
        else
          set motor All stop
      else
        set motor All stop
    else
      set motor All stop
  
```



<http://erasmus-artie.eu>



Optie 2 - volg het object met ArtieBot

Definieer eerst blokken **Drive** en **Stop** zoals beschreven in de vorige les (Programmeren van de robot)

```
define Drive SpeedA SpeedB Time
  digital pin 8 output LOW
  digital pin 9 output LOW
  if SpeedA > 0 then
    digital pin 12 output HIGH
  else
    digital pin 12 output LOW
  if SpeedB > 0 then
    digital pin 13 output HIGH
  else
    digital pin 13 output LOW
  pwm pin 3 output abs of SpeedA
  pwm pin 11 output abs of SpeedB
  wait Time seconds

define Stop
  pwm pin 3 output 0
  pwm pin 11 output 0
```





zie en configureer blokken zoals op onderstaande afbeelding:

```

Uno starts
HuskyLens initialize pin until success
HuskyLens switch algorithm to Object tracking
forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result? then
    if HuskyLens check if ID 1 frame is on screen from the result? then
      set x to HuskyLens get X center of the ID 1 No. 1 frame from the result
      if x >= 120 and x <= 200 then
        Stop
      else if x > 0 and x < 120 then
        Drive 20 80 0.25
      else if x > 200 and x < 320 then
        Drive 80 20 0.25
      else
        Stop
    else
      Stop
  else
    Stop

define Stop
  pwm pin 3 output 0
  pwm pin 11 output 0

define Drive SpeedA SpeedB Time
  digital pin 8 output LOW
  digital pin 9 output LOW
  if SpeedA > 0 then
    digital pin 12 output HIGH
  else
    digital pin 12 output LOW
  if SpeedB > 0 then
    digital pin 13 output HIGH
  else
    digital pin 13 output LOW
  pwm pin 3 output abs of SpeedA
  pwm pin 11 output abs of SpeedB
  wait Time seconds
  
```



<http://erasmus-artie.eu>

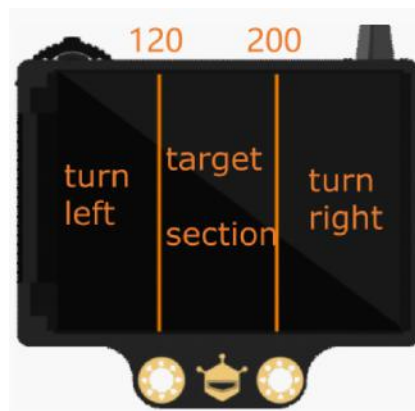
BEIDE opties - Controleer hoe het werkt

Upload het programma naar uw robot.

Pas indien nodig de snelheid van MotorA of MotorB aan.

Wanneer de doos met het geïdentificeerde object in het midden van het scherm staat, stopt de robot.

Wanneer de doos zich aan de linker- of rechterkant van het scherm bevindt, past de robot automatisch de positie naar links of rechts aan totdat de doos zich in het doelgedeelte van het scherm bevindt.





Volg het voorwerp

We hebben onze robots laten draaien om bezwaar te maken, maar ze volgen nog steeds niet. Om dit te bereiken, moeten we de grootte van het object detecteren om erachter te komen of het groot (dicht bij de camera) of klein (ver van de camera) is.

Maak een nieuwe variabele (numeriek) en noem deze h. Het behoudt de hoogte van het object dat we volgen. Als de objecthoogte tussen 60 en 100 ligt, zal de robot de huidige positie behouden. Als het kleiner is dan 60, is het ver en moeten we de robot vooruit laten rijden. Als het hoger is dan 100, zou de robot achteruit moeten gaan.

Optie 1 - Code voor Maqueen Plus

```
micro:bit starts
HuskyLens initialize pin until success
HuskyLens switch algorithm to Object tracking
set motor All stop

forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result? then
    if HuskyLens check if ID 1 frame is on screen from the result? then
      set x to HuskyLens get X center of the ID 1 No. 1 frame from the result
      set h to HuskyLens get height of the ID 1 No. 1 frame from the result
      if x >= 120 and x <= 200 and h >= 60 and h <= 100 then
        set motor All stop
      else if x > 0 and x < 120 then
        set motor Left move by 20 speed Forward
        set motor Right move by 80 speed Forward
      else if x > 200 and x < 320 then
        set motor Right move by 80 speed Forward
        set motor Left move by 20 speed Forward
      else if h > 0 and h < 60 then
        set motor All move by 80 speed Forward
      else if h > 100 and h < 240 then
        set motor All move by 80 speed Backward
    else
      set motor All stop
```



<http://erasmus-artie.eu>



Optie 2 - Code voor Arduino (ArtieBot):

```

Uno starts
HuskyLens initialize pin until success
HuskyLens switch algorithm to Object tracking
forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result? then
    if HuskyLens check if ID 1 frame is on screen from the result? then
      set x to HuskyLens get X center of the ID 1 No. 1 frame from the result
      set h to HuskyLens get height of the ID 1 No. 1 frame from the result
      if x >= 120 and x <= 200 and h >= 60 and h <= 100 then
        Stop
      else if x > 0 and x < 120 then
        Drive 20 80 0.25
      else if x > 200 and x < 320 then
        Drive 80 20 0.25
      else if h > 0 and h < 60 then
        Drive 80 80 0.25
      else if h > 100 and h < 240 then
        Drive -80 -80 0.25
      else
        Stop
    else
      Stop
  else
    Stop

define Drive SpeedA SpeedB Time
  digital pin 8 output LOW
  digital pin 9 output LOW
  if SpeedA > 0 then
    digital pin 12 output HIGH
  else
    digital pin 12 output LOW
  if SpeedB > 0 then
    digital pin 13 output HIGH
  else
    digital pin 13 output LOW
  pwm pin 3 output abs of SpeedA
  pwm pin 11 output abs of SpeedB
  wait Time seconds

define Stop
  pwm pin 3 output 0
  pwm pin 11 output 0

```

f <http://erasmus-artie.eu>

Beide opties - Controleer hoe het werkt

Upload het programma naar micro:bit/Arduino UNO om te controleren hoe het werkt. Voer de correcties uit om de beweging soepel te laten verlopen door de snelheid en de rijtijd van de motor aan te passen.

Nadat HuskyLens klaar is met het leren van een object, zullen robots het object automatisch volgen en vooruit, achteruit, links en rechts bewegen, waarbij de objectbox in het midden van het scherm en op een geschikte afstand wordt gehouden.

Wanneer de robot wordt gebruikt als volgrobot, kan hij worden geprogrammeerd om elk doel te lokaliseren met de HuskyLens-camera. Het betekent dat je van dit project een persoonsvolger kunt maken en de robot mensen kunt laten volgen.



Het volgen van objecten is de taak van het verkrijgen van een reeks waarden bij objectdetectie, het creëren van een unieke identificatie voor elk van de gedetecteerde objecten en het vervolgens volgen van elk van de objecten terwijl ze bewegen en ze markeren met een vak in de video, met inachtneming van de toegekende identifiers. State-of-the-art methoden omvatten het samenvoegen van gegevens van RGB- en andere camera's om een betrouwbaardere objecttracering te bieden. Nu begrijpen we de basisprincipes van het volgen van objecten en hebben we de functie voor het volgen van objecten van HuskyLens geleerd. We weten ook hoe we HuskyLens moeten gebruiken zodat onze robot een object kan volgen.



CONCLUSIE

Om een object met succes te volgen, moeten we dat object eerst detecteren en vervolgens volgen.
 Het volgen van een bewegend object vereist zowel visuele objectvolgtechnologie als handmatige bediening.



<http://erasmus-artie.eu>

Think!



What do I know now?



What did I learn?



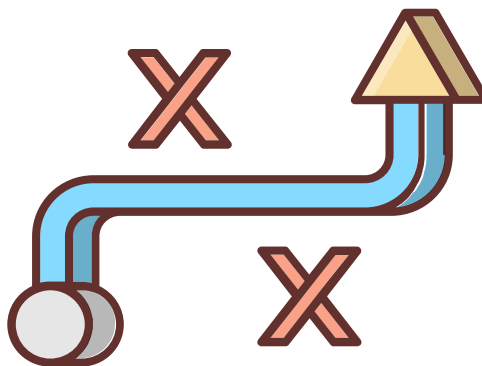
What do I want to know?



Een robot programmeren - regel volgen

In de vorige les hebben we geleerd hoe we robots kunnen programmeren om objecten te volgen.
Zorg ervoor dat je leerlingen alles begrijpen van de vorige les en klaar zijn voor de volgende stap.
Laten we nu leren hoe we een lijn kunnen vinden en volgen.

Hoofdidée:
Hoe bereid je de robot voor om de lijn te volgen?



<http://erasmus-artie.eu>



Lijntracering verwijst naar het proces waarbij objecten langs een aangewezen route bewegen. Een volledig functionele robot voor het volgen van lijnen gebruikt een mobiele robot als drager, een camera voor zichtbaar licht, een infrarood warmtebeeldcamera en andere detectie-instrumenten als laadsysteem, een multi-veldinformatiefusie van machinevisie, elektromagnetisch veld, GPS en GIS als een navigatiesysteem voor autonome beweging en tracking van de robot, en een embedded computer als software- en hardware-ontwikkelingsplatform voor het besturingssysteem.

Lijnvolgsensoren

Type	Infrared Line Tracking Sensor	Visual Sensor
Comparison Items		
Cost	Low	High
Range of Vision	The sensor has a small range of view; it needs to be close to the ground .	The range of vision is wide, and the movement state can be adjusted in advance according to the line changes.
Environment Adaptation	When the usage environment is changed, the sensitivity of the sensor needs to be adjusted, and the adjustment process is complicated.	When the use environment is changed, only the lines need to be relearned, and the operation is simple.
Map Adaptation	Generally, only suitable for simple maps with clear background lines, black and white lines or solid lines	Suitable for maps with clear background lines, multi-colour lines, solid lines, dotted lines and other complex conditions.



<http://erasmus-artie.eu>

HuskyLens algoritme voor het volgen van lijnen

HuskyLens lijnvolgfunctie is gebaseerd op Pixy, een open-sourceproject van Carnegie Mellon University. Het algoritme van Pixy kan de kleur van afbeeldingen herkennen. Het basisidee is om de kleurruimte te gebruiken om de achtergrond te verwijderen waarin gebruikers niet geïnteresseerd zijn en om de voorgrond (zoals lijnen) te extraheren.

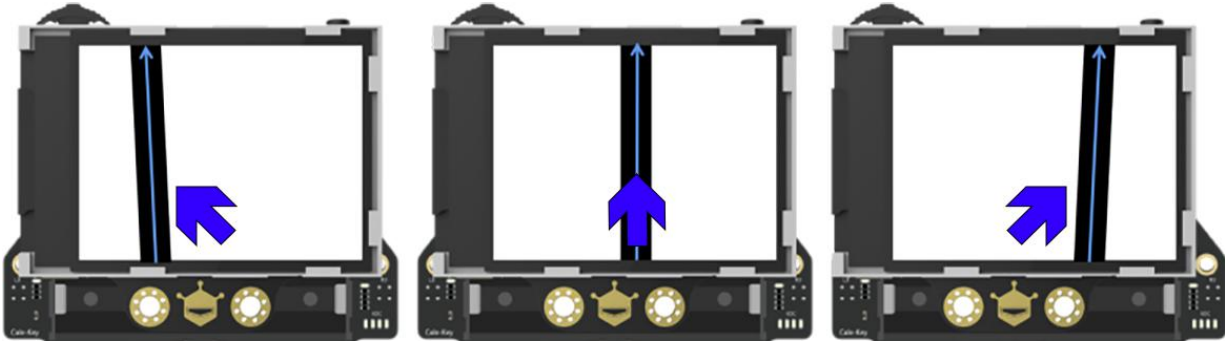


Hoe kan de robot de zwarte lijn volgen op de volgkaart (die zwarte lijnen heeft op een witte ondergrond)? In feite hoeven we alleen de relatieve positie van ARTIEbot ten opzichte van de zwarte lijn te weten.



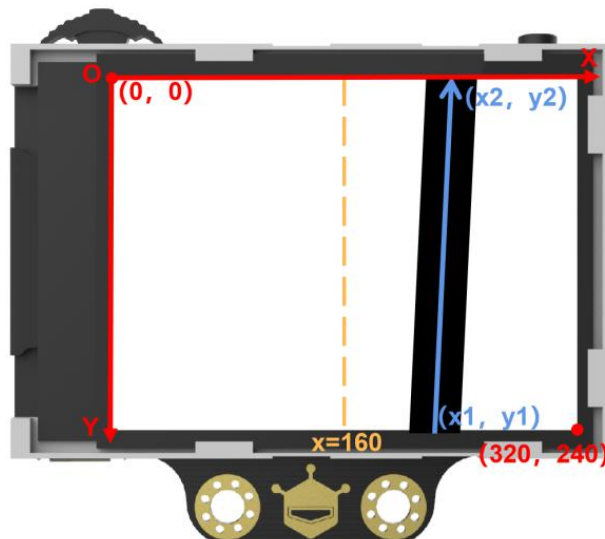
In principe hebben we de volgende drie situaties:

1. Als de robot zich aan de rechterkant van de zwarte lijn bevindt, moet hij naar links draaien
2. Als de robot in het midden staat, uitgelijnd met de zwarte lijn, moet hij rechtdoor gaan
3. Als de robot zich aan de linkerkant van de zwarte lijn bevindt, moet hij naar rechts draaien



Implementatie

De resolutie van het HuskyLens-scherm is 320×240 . Het O-punt in de linkerbovenhoek van het scherm is de oorsprong van de coördinaten van het scherm $(0, 0)$, de horizontale richting naar rechts is de positieve richting van de X-as en de verticale neerwaartse richting is de positieve richting van de Y-as, dus de coördinaten in de rechter benedenhoek van het scherm zijn $(320, 240)$. De oranje stippellijn in de afbeelding is de centrale as van het scherm en de x-waarde van deze lijn is 160. De zwarte lijn in het scherm hieronder is de kaartlijn "gezien" door de HuskyLens-camera. De blauwe pijl is de lijnrichting berekend door HuskyLens. De startpuntcoördinaten van de blauwe pijl zijn (x_1, y_1) en de eindpuntcoördinaten zijn (x_2, y_2) .





Om het simpel te houden: we hoeven alleen het startpunt (x1) van de blauwe pijl ten opzichte van de centrale as ($x=160$) te weten om lijntracering te implementeren.

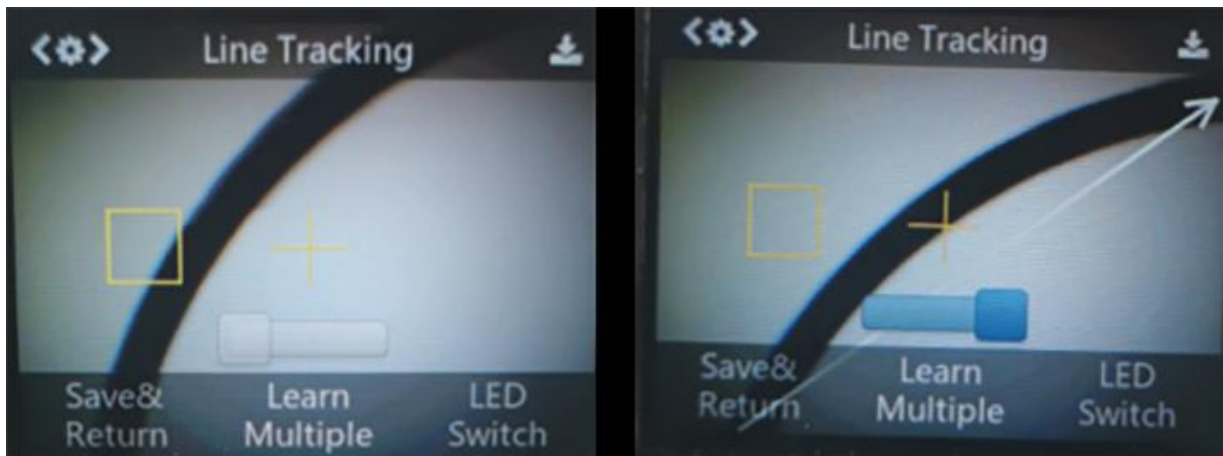
Deze functie kan lijnen met gespecificeerde kleuren volgen en padvoorspellingen doen. De standaardinstelling is om lijnen van slechts één kleur te volgen en dit project gebruikt lijntracering in één kleur.

Camera instellingen

Stap 1: Draai de functieknop naar links of rechts totdat het woord "Line Tracking" bovenaan het scherm wordt weergegeven.

Stap 2: Druk lang op de functieknop om de parameterinstelling van de lijnvolgfunctie in te voeren.

Stap 3: Draai de functieknop naar rechts of links totdat "Learn Multiplie" is geselecteerd, druk vervolgens kort op de functieknop en draai deze naar links om de schakelaar "Learn Multiplie" uit te schakelen, dat wil zeggen het vierkante pictogram op de voortgang balk is naar links gedraaid. Druk vervolgens kort op de functieknop om deze parameter te voltooien.

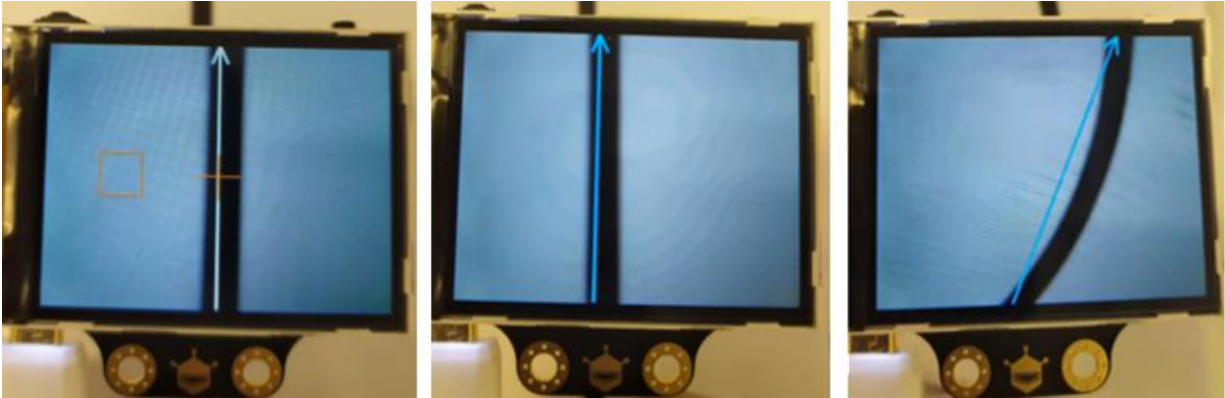


Stap 4: U kunt de LED ook inschakelen door "LED Switch". Dit is erg handig in de donkere omgeving. Verwijzend naar de bovenstaande methode, schakelt u de "LED Switch".

Stap 5: Draai de functieknop naar links tot "Save & Return" is geselecteerd en druk kort op de functieknop om de parameters op te slaan en deze zal automatisch terugkeren.

Leren en volgen

Lijn leren: richt het "+" symbool op de lijn en wijs vervolgens het oranje vak op het achtergrondgebied. Het wordt aanbevolen dat er geen andere lijnen op het scherm staan. Probeer HuskyLens evenwijdig aan de doellijn te houden; HuskyLens zal de lijn automatisch detecteren en er verschijnt een witte pijl. Druk vervolgens kort op de "leerknop" en de witte pijl verandert in een blauwe pijl.



- Bij het leren van de lijn moeten we de positie van HuskyLens aanpassen om evenwijdig aan de lijn te zijn.
- HuskyLens kan lijnen leren in elke kleur die een duidelijk kleurcontrast heeft met de achtergrond, maar deze lijn moet monochroom zijn om het lijnvolgproces stabiel te houden.
- HuskyLens kan meerdere lijnen met verschillende lijnkleuren leren en volgen, maar al deze lijnen moeten monochroom zijn en zichtbaar contrasteren met de achtergrond. In dit voorbeeld gebruiken we een zwarte lijn (zwarte isolatietape op een witte achtergrond zoals papier of wit MDF).
- De zichtbaarheid van de lijn hangt sterk af van het omgevingslicht. Probeer bij het volgen van de lijn het omgevingslicht zo stabiel mogelijk te houden en gebruik indien nodig HuskyLens LED.



Open je Mind+ en laad extensies.

Zorg er bij het gebruik van de micro:Maqueen plus-robot voor dat u de juiste versie selecteert (V1 of V2).

Hernoemen `my float variable` naar `x`. Right click on variable -> Rename numeric variable.

Algoritme

- Lees `x1` (of `x` begin) van de HuskyLens lijnvolgfunctie - dit is het beginpunt van de blauwe pijl
- Als de zwarte lijn zich aan de linkerkant van het scherm bevindt ($x1 < 150$), moet de robot naar links draaien.
- Als de zwarte lijn zich aan de rechterkant van het scherm bevindt ($x1 > 170$), moet de robot naar rechts draaien.
- Als de zwarte lijn in het midden van het scherm staat ($150 \leq x1 \leq 170$), moet de robot rechtdoor gaan.



Optie 1 - Lijn volgen met Maqueen Plus

Gebruik deze code:

```
micro:bit starts
HuskyLens initialize pin until success
HuskyLens switch algorithm to Line tracking
forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result? then
    if HuskyLens check if ID 1 arrow is on screen from the result? then
      set my float variable to HuskyLens get X beginning of ID 1 arrow from the result
      if my float variable >= 150 and my float variable <= 170 then
        set motor All move by 60 speed Forward
      else if my float variable < 150 then
        set motor Right move by 90 speed Forward
        set motor Left move by 30 speed Forward
      else if my float variable > 170 then
        set motor Left move by 90 speed Forward
        set motor Right move by 30 speed Forward
      else
        set motor All stop
```





Optie 2 - Lijn volgen met Arduino (ArtieBot)

Definieer eerst de blokken Rijden en Stoppen zoals beschreven in de les Programmeren van de robot. Het is een goed idee om de veelgebruikte scripts in een rugzak te stoppen en ze gemakkelijk tussen projecten te verplaatsen.

```
define Drive SpeedA SpeedB Time
  digital pin 8 output LOW
  digital pin 9 output LOW
  if SpeedA > 0 then
    digital pin 12 output HIGH
  else
    digital pin 12 output LOW
  if SpeedB > 0 then
    digital pin 13 output HIGH
  else
    digital pin 13 output LOW
  pwm pin 3 output abs of SpeedA
  pwm pin 11 output abs of SpeedB
  wait Time seconds

define Stop
  pwm pin 3 output 0
  pwm pin 11 output 0
  digital pin 8 output HIGH
  digital pin 9 output HIGH
```





```

Uno starts
HuskyLens initialize pin until success
HuskyLens switch algorithm to Line tracking
define Stop
  pwm pin 3 output 0
  pwm pin 11 output 0
forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result? then
    if HuskyLens check if ID 1 arrow is on screen from the result? then
      set x to HuskyLens get X beginning of ID 1 arrow from the result
      if x >= 150 and x <= 170 then
        Drive 90 90 0.25
      else if x < 150 then -
        Drive 40 90 0.25
      else if x > 170 then -
        Drive 90 40 0.25
      else
        Stop
  
```

f

```

define Drive SpeedA SpeedB Time
  digital pin 8 output LOW
  digital pin 9 output LOW
  if SpeedA > 0 then
    digital pin 12 output HIGH
  else
    digital pin 12 output LOW
  if SpeedB > 0 then
    digital pin 13 output HIGH
  else
    digital pin 13 output LOW
  pwm pin 3 output abs of SpeedA
  pwm pin 11 output abs of SpeedB
  wait Time seconds
  
```

<http://erasmus-artie.eu>



Beide opties - Test uw algoritme

Bereid het witte oppervlak (papier of MDF) voor en gebruik de isolatietape om een lijn te trekken (zie voorbeeld hieronder of maak iets soortgelijks).



Upload het programma naar je robot, plaats de robot ergens op de lijn en kijk hoe hij beweegt. Volgt het de lijn?

Valt de lijn weg? Is er een manier om ermee om te gaan? De volgende tips helpen je:

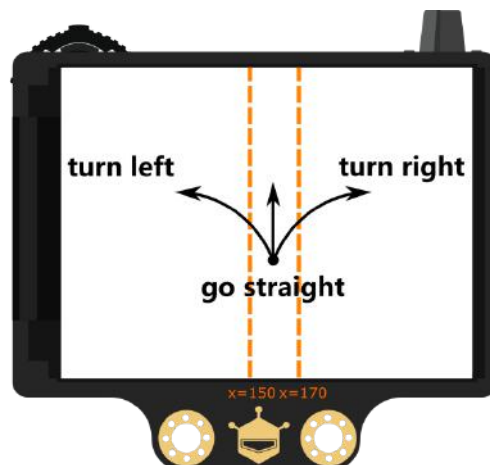
- Probeer de hoek van de camera te veranderen
- Proberen te krijgen **X beginning** van ID 1 pijl in plaats van X eindpunt

Voeg faalveilige code toe met behulp van de volgende hint:

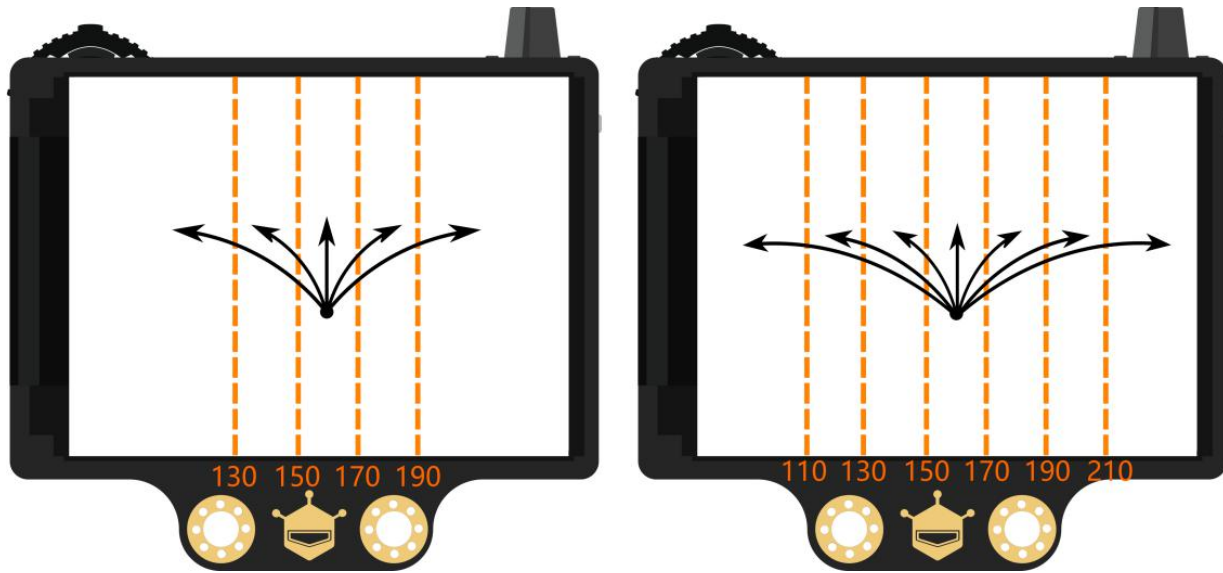
- Als er een lijn aanwezig is en een pijl wordt gedetecteerd, gaat u verder met de pijlverwerkingscode en slaat u de inhoud van x-variabele op in een nieuwe variabele genaamd lastx

- Als de lijn verloren is en de pijl niet wordt gedetecteerd, gebruikt u lastx in plaats van x om de lijn opnieuw te bereiken.

En tot slot het belangrijkste - pas de correcties toe om de beweging soepel te laten verlopen door de motorsnelheid en rijtijd aan te passen. Dit algoritme analyseert de positie van x; het heeft 3 gevallen zoals in de afbeelding hieronder.



Can you optimise this algorithm for 5 cases or even 7 to make the movement smoother?
See pictures below to get the idea how to do this.



Samenvatting kennis

1. Begrijp de belangrijkste principes van lijntracering
2. Leer de HuskyLens lijnvolgfunctie toe te passen
3. Pas het regelvolgende algoritme toe en optimaliseer het

CONCLUSIE

We leerden de belangrijkste principes van lijntracering met behulp van de HuskyLens-functie en optimaliseerden het algoritme voor lijntracering.

Think!



What do I know now?



What did I learn?



What do I want to know?

