

2017 • 2018
Faculteit Industriële ingenieurswetenschappen
master in de industriële wetenschappen: energie

Masterthesis

Spraakaansturing van een KUKA LBR iiwa R14 robot in combinatie met Google Home

PROMOTOR :

dr. ir. Johan BAETEN

PROMOTOR :

Dhr. David CLEEREN

Jondar Schumann

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie,
afstudeerrichting automatisering



Universiteit Hasselt | Campus Diepenbeek | Agoralaan Gebouw D | BE-3590 Diepenbeek
Universiteit Hasselt | Campus Hasselt | Martelarenlaan 42 | BE-3500 Hasselt



2017 • 2018

Faculteit Industriële ingenieurswetenschappen
master in de industriële wetenschappen: energie

Masterthesis

Spraakaansturing van een KUKA LBR iiwa R14 robot in combinatie met Google Home

PROMOTOR :

dr. ir. Johan BAETEN

PROMOTOR :

Dhr. David CLEEREN

Jondar Schumann

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: energie,
afstudeerrichting automatisering



KU LEUVEN

Woord vooraf

Het is een lange en intensieve periode geweest voor de totstandkoming van deze thesis. Dit is natuurlijk ook logisch omdat dit eindwerk de laatste stap is in het behalen van een diploma maar tegelijkertijd ook de eerste stap is in de wereld van een industrieel ingenieur. Het is een test om de kennis en vaardigheden die opgedaan zijn tijdens de academie jaren om te zetten in de realiteit. Uiteraard gaat deze test gepaard met goede en minder goede momenten. Gelukkig sta je er niet alleen voor en is er hulp van verschillende personen, die ik nu oprecht wil bedanken.

Als eerste wil ik mijn begeleiders, David Cleeren en Wim Persoons, hartelijk bedanken dat ze mij de kans hebben gegeven om de thesis bij KUKA te laten volbrengen. Ook bedank ik jullie voor de goede inzet, geduld en enthousiasme die jullie tijdens het onderzoek getoond hebben. Het is niet evident om naast jullie drukke agenda, ook nog de tijd en energie te vinden die jullie in mij gestoken hebben. Zonder jullie was het mij niet gelukt om op zo korte tijd zoveel bij te leren en mij klaar te stomen voor de wereld waar ik binnenkort in terecht kom. Nogmaals mijn oprechte dank hiervoor.

Op de tweede plaats wil ik mijn interne promotor, Johan Baeten, bedanken voor jouw goede begeleiding en constructieve en kritische mening doorheen het onderzoek. Dit zette er mij aan toe om ruimer te denken en te blijven doorzetten om dit eindwerk tot een goed einde te brengen.

Ten derde wil ik heel het KUKA-team bedanken voor hun hulp en steun die ze hebben geboden doorheen het onderzoek en testfasen. De suggesties en opmerkingen die ze gegeven hebben zijn van onschatbare waarde en hebben een grote invloed gehad op de uitwerking van het eindwerk.

Uiteindelijk bedank ik mijn familie en vrienden die mij gesteund hebben tijdens goede en minder goede perioden van het onderzoek. Onrechtstreeks waren ze voor mij een drijfveer om het eindwerk te voltooien.

Inhoudsopgave

Woord vooraf	3
Lijst van tabellen	7
Lijst van figuren	9
Verklarende woordenlijst	11
Abstract	13
Abstract in English	15
1 Introductie	17
1.1 Probleemstelling	18
1.2 Doelstellingen	19
1.3 Materiaal en methode	20
2 Alternatieve systemen van robot programmatie	21
2.1 Multimodale interactie	21
2.2 Spraak geactiveerde manipulator	21
2.3 Robuuste mens-robot collaboratie in industriële omgevingen	22
2.4 Mens-robot collaboratie via spraak en handbegeleiding	23
2.5 Spraak geactiveerde commando's en controle via spraakherkenning over een WiFi-kanaal	23
3 Procesverloop van spraakaansturing met Google Home	25
3.1 Overzicht proces	25
3.2 Google Home	25
3.3 Dialogflow	25
3.4 Heroku Websocket server	26
3.5 Service laptop	26
3.6 Controller Robot	26
4 Google Home en Dialogflow	27
4.1 Dialogflow	27
4.2 Verloop van een gesprek	27
4.2.1 Welkom	28
4.2.2 Vraagstelling	28
4.2.3 Verwerking	29
4.2.4 Afhandeling	29
4.2.5 Aanvullend gespreksonderwerp	30

4.3	Functies van dialogflow	30
4.3.1	Agents	30
4.3.2	Intents	31
4.3.3	Entities.....	34
4.4	Uitwerking programma	34
4.4.1	Default welcome intent	34
4.4.2	Default fallback intent.....	34
4.4.3	GetAxisPosition	34
4.4.4	GetWorldPosition	35
4.4.5	SetAxisPosAbs	35
4.4.6	SetAxisPosRel.....	35
4.4.7	SetWorldPosAbs.....	36
4.4.8	SetWorldPosRel	36
4.4.9	SavePosition	36
4.4.10	MoveToPosition	36
4.4.11	MoveToHome.....	37
4.4.12	StopRobot.....	37
4.4.13	GetActualTool	37
4.4.14	AttachTool.....	37
4.4.15	ToolsFunction	38
4.4.16	Search.....	38
4.4.17	Vibrate	38
4.4.18	RobotHelp	38
4.4.19	RobotJokes.....	38
4.4.20	ROBOTDIALOG	39
4.4.21	ActivateDialog	39
5	Achtergrond processen	41
5.1	Heroku Websocket Service.....	41
5.2	Service Laptop	41
6	Besluit.....	43
6.1	Toekomstig onderzoek	43
	Bibliografie	45

Lijst van tabellen

Tabel 1: Parameters.GetAxisPosition	34
Tabel 2: Parameters.GetWorldPosition	35
Tabel 3: Parameters.SetAxisPosAbs.....	35
Tabel 4: Parameters.SetWorldPosAbs	36
Tabel 5: Parameters.SavePosition	36
Tabel 6: Parameters.AttachTool	37
Tabel 7: Parameters.Toolsfunction	38
Tabel 8: Parameter.ROBOTDIALOG	39

Lijst van figuren

Figuur 1: Wereldwijde vestigingen van KUKA	17
Figuur 2: verschillende formulering van een instructie.....	22
Figuur 3: Onderverdeling in soorten objecten, overgenomen uit	22
Figuur 4: Assemblering van een auto met een cobot.....	23
Figuur 5: Schematisch overzicht van spraakherkenning.....	24
Figuur 6: Schematisch verloop van proces.....	25
Figuur 7: Welkomstboodschap.....	28
Figuur 8: Vraagstelling	28
Figuur 9: Verwerking boodschap.....	29
Figuur 10: Gesprek afhandeling	29
Figuur 11: Aanvullend gesprek	30
Figuur 12: Schematisch verloop van data verwerking.....	31
Figuur 13: Voorbeelden van intents.....	31
Figuur 14: Voorbeelden van training phrases	32
Figuur 15: Action parameters	32
Figuur 16: Afhandeling van gesprek	33
Figuur 17: WebSocketClient.....	41

Verklarende woordenlijst

LBR:	Leicht Bau Robotor (licht gewicht robot)
IIWA:	Intelligent Industrial Work Assistent
SAM:	Speech Activated Manipulator
SAPI:	Speech Application Program Interface
TCP:	Tool-Centre-Point

Abstract

Voor de integratie van industriële robots in nieuwe of bestaande processen is er speciaal opgeleid personeel nodig dat ervaring heeft met het gebruik van de fabrikantafhankelijke programmeeromgeving die op de Teachpad geïnstalleerd is. Dit is het bedieningspaneel voor de programmering van de robot. De integratie van robots is vooral voor kleine en middelgrote ondernemingen een grote kost en een tijdrovend proces. Daarom is er vraag naar andere, meer natuurlijke manieren van programmeren zodat minder ervaren werknemers een robot snel en intuïtief kunnen bedienen.

Deze masterthesis legt de focus op het uitwerken van een spraakherkenningsysteem voor het programmeren van een robot. De gebruikte collaboratieve robot is een KUKA LBR iiwa R14 820 met ingebouwde krachtsensoren.

Voor de realisatie van de spraakherkenning heeft KUKA Houthalen geopteerd voor een slimme luidspreker/microfoon van Google, de Google Home. Deze module is reeds enkele jaren op de markt beschikbaar en heeft een aanvaardbare prijs. Daarnaast is de software 'Dialogflow' geavanceerd en bevat het een ruime woordenschat.

Met de software zijn er een aantal basis functies uitgebouwd zoals het opvragen van de robotpositie, het toewijzen van een willekeurige positie, het opslaan en aanvaren van de toegewezen posities en het bedienen van het gebruikte gereedschap. De functionaliteit van het systeem wordt getest en geëvalueerd door verschillende testpersonen die via de robot een product opnemen en naar een andere positie brengen binnen een opgelegde tijdslimiet.

Abstract in English

For the integration of industrial robots in new or existing processes, specially trained employees are needed who know how to use the specific robot programming languages on the Teachpad. This is the control panel for programming the robot. Integrating these robot systems are a costly and time-consuming investment and not affordable for small and medium-sized enterprises. Therefore a more natural, human friendly way of programming these robots is demanded so less trained employees can operate them.

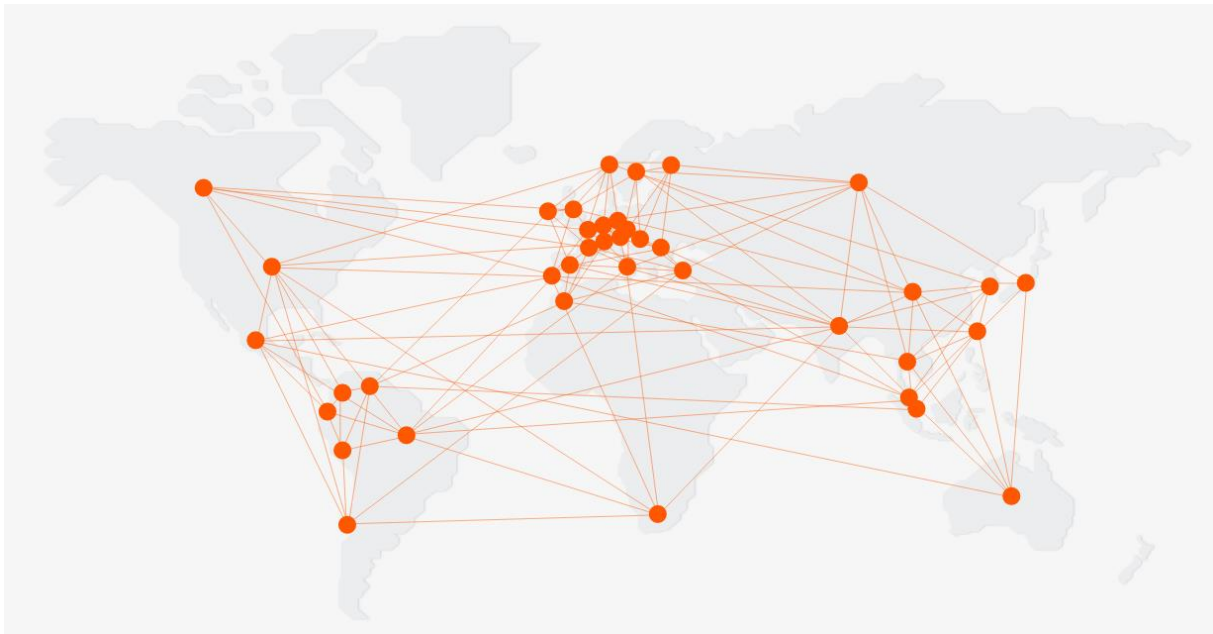
In this master's thesis a speech-recognition system is proposed to program a collaborative robot. A KUKA LBR iiwa R14 820 with intern force sensors is used in combination with Google's smart speaker, the Google Home. KUKA Houthalen made this choice because the Google Home is on the market for quite some years and the cost is acceptable. The software 'Dialogflow' is very advanced and has a huge grammar on board.

With the software, several basic functions have been built like requesting the position of the robot in joint- or Cartesian coordinates, give the robot a new position in joint- or Cartesian coordinates, save and go to a teached position and activate the robot's tool functions. The systems functionality is tested and evaluated by several test persons through a demonstration. Hereby, a product is picked up and placed in an another position in a specific time limit.

1 Introductie

KUKA Automatisering + Robots N.V. is een wereldwijd gevestigde onderneming met hoofdvestiging in Augsburg, Duitsland. KUKA telt meer dan 12.300 medewerkers. Ze bieden intelligente state-of-the-art automatiseringsoplossingen aan voor diverse sectoren zoals o.a. de automobielandustrie of de voedingsindustrie. Hierbij speelt de integratie van industriële robots vaak een grote rol en dit is waar KUKA al meer dan 30 jaar in gespecialiseerd is. Om te kunnen blijven concurreren met andere robotfabrikanten, bv. FANUC en ABB, worden bij KUKA nieuwe innovatieve technieken ontwikkeld die de inzetbaarheid van hun robots naar een hoger niveau tillen.

Keller und Knappich GmbH ontstond in 1898 in Augsburg door Johann Keller en Jacob Knappich. Aanvankelijk verzorgde het bedrijf de installatie verlichting in huizen en straten waarna het zich ook toespitste tot andere openbare producten en werd zo marktleider op vlak van openbaar vervoer in Europa.



Figuur 1: Wereldwijde vestigingen van KUKA, overgenomen van <https://www.kuka.com/nl-be>

De vestiging in Houthalen voorziet deze automatiseringsoplossingen voor de Benelux waarbij de projecten bij de klant of in het bedrijf zelf worden opgebouwd en getest [1]. Daarnaast worden hier ook innovatieve ideeën ontwikkeld zoals bv. de aansturing van een robot via handbegeleiding of via spraakbegeleiding. De verwezenlijking van dit laatste punt is de opdracht van mijn masterproef

1.1 Probleemstelling

De klassieke programmering van een industriële robot gebeurt via een bedieningspaneel waarmee het pad en de taken aan de robot worden aangeleerd. Afhankelijk van de complexiteit van de taak en de omgeving van de robot kan dit een tijdrovend en kostelijk proces zijn. Voor kleine en middelgrote bedrijven is het daarom minder aantrekkelijk om industriële robots in hun bedrijf te integreren. Daarnaast is het ook van cruciaal belang dat de veiligheid van de programmeur en anderen steeds gewaarborgd blijft. Dit wordt verwezenlijkt door een hekwerk met beveiligingen rond de robot te plaatsen waardoor een werknemer niet in de buurt kan komen van een bewegende robot. Een nadeel hierbij is dat er kostbare ruimte verloren gaat. Daarnaast bestaat de kans dat de beveiliging wordt overbrugd en de veiligheidsfunctie wegvalt.

Vanaf de jaren 2000, met de opkomst van collaboratieve robots, oftewel cobots, wordt er steeds meer onderzoek verricht naar andere natuurlijke manieren van programmeren. Dit kan gerealiseerd worden door bijvoorbeeld handbegeleiding of spraakherkenning om de aansturing vlotter te laten verlopen [2]. Hierdoor is het mogelijk dat er minder getraind personeel nodig is om de robot te programmeren en de kleine en middelgrote bedrijven eerder geneigd zijn om over te stappen naar de integratie van industriële robots.

De aansturing via spraak, ook wel 'Teach-by-Speech' genoemd, kan hierbij een mogelijke oplossing zijn. Een eerste voordeel is dat de aansturing vereenvoudigt t.o.v. de klassieke manier. Voorheen had je een basiscursus nodig om de programmeertaal te begrijpen. Dit wordt nu vervangen door simpele instructies die men aan de robot vertelt. Een tweede voordeel is dat de programmeur zijn handen vrij heeft tijdens het programmeren, wat voorheen niet mogelijk was omdat tijdens de programmatie een (zogenoemde) dodemans knop moest bediend worden.

1.2 Doelstellingen

De kerndoelstelling van het onderzoek houdt in om een applicatie te verwezenlijken waarbij een industriële collaboratieve robot aangestuurd wordt via spraak. Hierbij kan de dialoogvoering enerzijds door de mens gestart worden of anderzijds door de robot. In het eerste geval heeft de programmeur kennis van deze aansturing. In het tweede geval wordt er een dialoog gevoerd waarbij de opties die op het bedieningspaneel staan geactiveerd kunnen worden.

In de tijdspanne van de thesis is het niet mogelijk om alle functies van de robot via spraak te kunnen bedienen. Daarom worden er een aantal basisfuncties uitgewerkt die hieronder opgesomd worden:

- Positie opvragen van de robot,
- Positie toewijzen aan de robot,
- Positie opslaan en terug aanvaren,
- Functies van het aanwezige gereedschap,
- Een aantal extra functies,
- Spraak ondersteunde dialoog.

De programmatie van de spraakapplicatie gebeurt via het softwarepakket 'Dialogflow', een spraakherkenningssoftware van Google. Hierbij wordt er een 'agent' ontwikkeld die instaat voor de communicatie tussen de mens en de robot. Het komt erop neer om de bevelen van de programmeur op te nemen, uit deze data de benodigde parameters te selecteren en door te sturen naar de controller van de robot. Daarnaast moet het ook mogelijk zijn dat de robot een dialoog start met de programmeur.

De mogelijkheden van de spraakbesturing wordt op het einde van de thesis via een demonstratie voorgesteld en getest aan verschillende testpersonen. Naar gelang van de bevindingen van deze personen worden er eventuele verbeteringen aangebracht. Daarnaast wordt er ook achterhaald als ze de applicatie als een meerwaarde zouden zien in de industrie.

1.3 Materiaal en methode

Voor de realisatie van deze applicatie wordt er gebruik gemaakt van een KUKA LBR iiwa 14 R820 robot gecombineerd met een slimme luidspreker/microfoon van Google, de Google Home. De programmering van de Google home wordt gedaan via het programma 'DialogFlow'. Deze software is vrij geavanceerd en gratis te gebruiken. Ook maken diverse applicaties reeds gebruik van deze software wat er op duidt dat deze niet meer in haar kinderschoenen staat. Een nadeel aan dit programma is dat er een constante internetverbinding vereist is tijdens het uitvoeren van de applicatie. In het geval dat de verbinding niet aanwezig is kan de slimme luidspreker dus ook geen communicatie maken met de robotcontroller.

De dialoogvoering van de robot naar de mens toe maakt gebruik van cURL-instructies. Deze worden geprogrammeerd in de software van de robot. De dialoogvoering vanuit de robot heeft als functie dat wanneer er een dialoog actief is op het Teachpad, de mogelijke opties geactiveerd kunnen worden via spraak.

De eerste fase van de masterproef voorziet het leren werken met het programma 'Dialogflow' en het uittesten van de mogelijkheden. De aansturing van een robot bevat veel en deze kunnen niet allemaal uitgewerkt worden in de tijdspannen van de thesis. Daarom worden er een aantal basis functies geprogrammeerd en daarna getest in combinatie met de robot en de Google Home. Tijdens de testen zal de bedrijfspromotor aanwezig zijn om het resultaat mee te valideren en bij te sturen.

In de tweede fase wordt er een demonstratie uitgewerkt waarbij de mogelijkheden van de interactieve applicatie getest worden door verschillende interne of externe testpersonen. Op basis van de bevindingen van deze personen worden er eventuele verbeteringen aangebracht.

2 Alternatieve systemen van robot programmatie

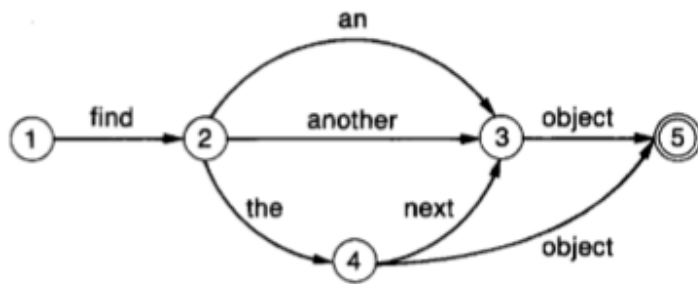
Vooraleer de masteropdracht te bespreken wordt in deze sectie besproken welke verschillende manieren van programmering in aanmerking komen om een industriële robot aan te sturen. Bij de klassieke programmering wordt er gebruik gemaakt van een bedieningspaneel en de wijze van programmering kan zeer complex zijn en dit vertaalt zich in een tijdrovend en duur proces. Voor grote bedrijven vormt dit niet meteen een probleem maar voor kleine en middelgrote bedrijven is het niet evident om een robot in hun proces te integreren. Daarom wordt er reeds decennia lang onderzoek verricht naar andere, meer natuurlijke manieren van programmering van een industriële robot.

2.1 Multimodale interactie

Een uitgebreide studie rond een multimodaal interface wordt gegeven in [2]. Hierbij wordt er eerst een analyse gegeven van welke menselijke output kanalen zoals handbegeleiding, oogcontact of spraak kunnen gecombineerd worden om tot een robuuste programmeringseenheid te verkrijgen. Daarnaast worden diverse taken van een robot opgesomd en welke aandachtspunten hieraan verbonden zijn. In hun werk wordt een grafische user interface ontworpen die verschillende input kanalen gebruikt om een industriële robot te programmeren. Uiteindelijk wordt er een evaluatie gegeven van het voorgesteld systeem over de ervaring die de gebruikers hebben opgedaan.

2.2 Spraak geactiveerde manipulator

Een specifiek ontwikkeld robot spraakprogrammeringstaal, Speech activated manipulator (SAM), wordt voorgesteld in [3]. De gebruikte robot is een zes-assige cartesische robot met een werktafel voor zich. De robot wordt bediend via een telefoon die de spraakinstructies opneemt. Daarnaast is er aan het gereedschap van de robot een camera, een afstandssensor en een krachtsensor bevestigd. Voor de spraakherkenning hebben de onderzoekers een eigen systeem ontwikkeld die de spraakopdrachten analyseert en vergelijkt met een database. Ook is het mogelijk om nieuwe woorden aan te leren via het systeem. In het artikel wordt ook beschreven hoe voor een bepaalde instructie de zinnen kunnen variëren, een voorbeeld hiervan wordt gegeven in figuur twee.

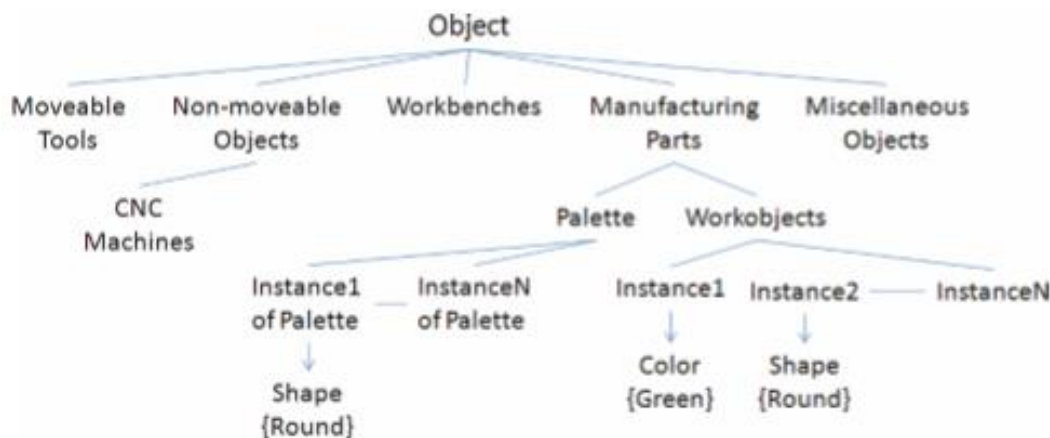


Figuur 2: verschillende formulering van een instructie, overgenomen uit "SAM: A Persepective Spoken Language Understanding Robot" [3]

Om het systeem te valideren zijn er een aantal objecten met variërende vormen en kleuren voor de robot geplaatst. Met specifieke instructies is het dan mogelijk om een bepaald opbject op te pakken en naar andere plaatsen te brengen.

2.3 Robuuste mens-robot collaboratie in industriële omgevingen

In het artikel van [4] wordt een industriële robot bediend via spraak instructies van de mens. Via een robot simulatie pakket worden de acties van de robot visueel weergegeven. De spraakherkenning software van Microsoft, Microsoft Speech API 5.1, wordt gebruikt om de spraak instructies van de mens op te nemen en om te zetten naar tekst. Deze tekst wordt dan doorgestuurd naar de simulatie software en voert de robot de specifieke actie uit. In het artikel wordt er een onderscheid gemaakt tussen diverse objecten die aanwezig zijn bij de robot opstelling, zie figuur drie.

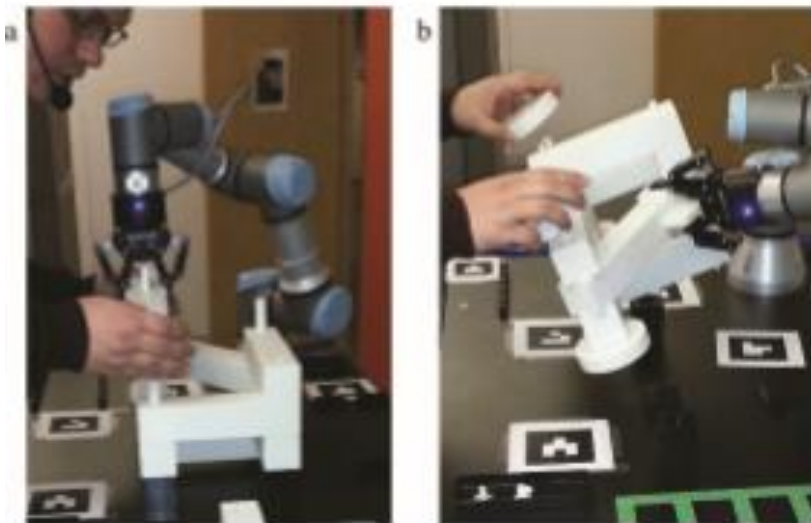


Figuur 3: Onderverdeling in soorten objecten, overgenomen uit "Towards Robust Human Robot Collaboration in Industrial Environments" [4]

Via deze verdeling is het mogelijk om bepaalde acties te weigeren die niet bedoeld zijn voor een object. Hierdoor worden fouten geminimaliseerd en wordt het systeem robuuster gemaakt.

2.4 Mens-robot collaboratie via spraak en handbegeleiding

In het artikel van [5] wordt een systeem voorgesteld waarbij handbegeleiding en spraak gecombineerd worden om een industriële robot te bedienen. De gebruikte robot is een UR3 van Universal Robots vanwege zijn collaboratieve eigenschappen en heeft een twee-vingerig gereedschap aan de flens verbonden. De spraakherkenning software is van Microsoft, Microsoft Speech API 11 (SAPI 11) en de instructies worden opgenomen via een microfoon van Sennheiser, de ME 3 EW microfoon. Het voordeel van SAPI 11 is dat het niet cloud gebaseerd is en dus geen permanente internet verbinding vereist. In de industrie komt dit eerder sporadisch voor maar het biedt toch een voordeel. De functionaliteit van het systeem werd voorgesteld door een demonstratie waarbij een houten auto geassembleerd wordt, zie figuur vier. Het systeem wordt door verschillende groepen getest en geëvalueerd om eventuele verbeteringen aan te brengen.

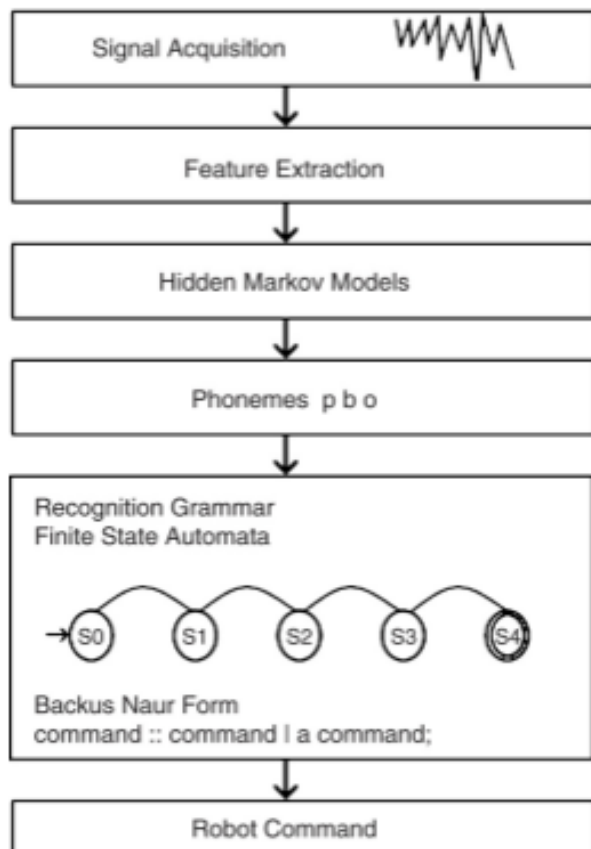


Figuur 4: Assemblage van een auto met een cobot, overgenomen uit "Human-Robot collaboration Demonstrator Combining Speech Recognition and Haptic Control" [5]

2.5 Spraak geactiveerde commando's en controle via spraakherkenning over een WiFi-kanaal

Het artikel van [6] behandelt een systeem waarbij een framework wordt uitgewerkt om een Lego Mindstorm robot aan te sturen via spraakherkenning. De onderzoekers gebruiken drie verschillende spraakherkenning software, Sphinx4, Microsoft SAPI en Java speech API om deze met elkaar te vergelijken op betrouwbaarheid en robuustheid.

Het artikel geeft ook meer uitleg over het proces achter spraakherkenning. Een schema hiervan wordt afgebeeld in figuur vijf.



Figuur 5: Schematisch overzicht van spraakherkenning, overgenomen uit "Voice activated command and control with speech recognition over WiFi" [6]

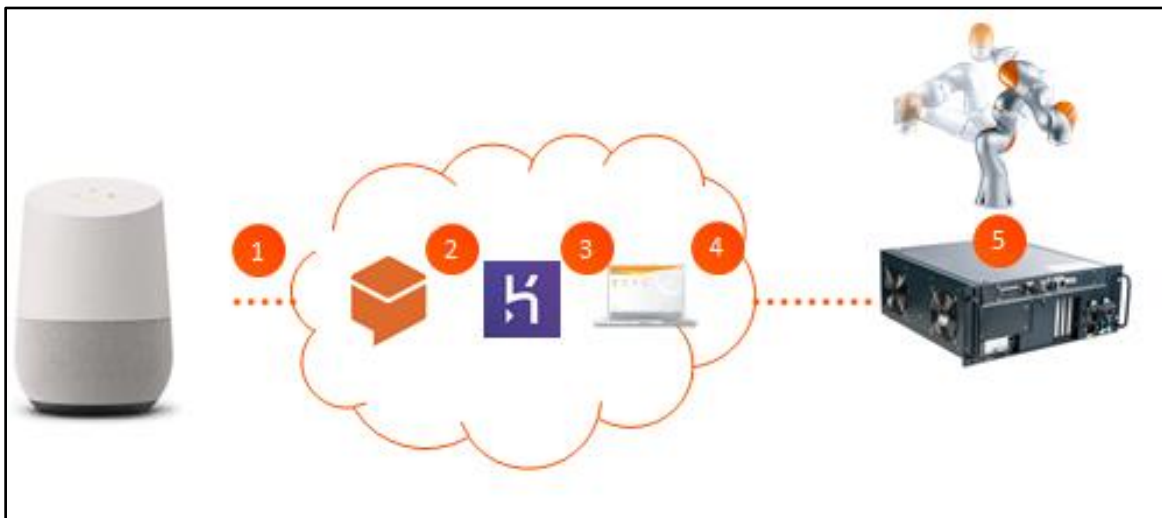
De eerste stap in spraakherkenning is om de spraak op te nemen via een microfoon. Op deze ruwe data wordt dan een feature extraction uitgevoerd en wordt dit aangeboden aan een Hidden Markov Model. Dit model zet de data om naar letters. Via een database worden deze letters omgevormd naar woorden en kunnen ze uiteindelijk aan de robot geboden worden om een bepaalde actie te ondernemen. Uit de testen is gebleken dat de spraakherkenningsoftware van Microsoft, SAPI 5, de beste resultaten.

3 Procesverloop van spraakaansturing met Google Home

In deze sectie komt er eerst een beschrijving van het systeem en de componenten die tijdens de masterproef zijn gebruikt. Daarop volgt een uitleg over de software om de spraakherkenning te programmeren.

3.1 Overzicht proces

Afbeelding 6 geeft een schematisch verloop van de communicatie tussen mens en robot.



Figuur 6: Schematisch verloop van proces, overgenomen van KUKA Teach-by-Speech presentatie

1. Google Home with Google action activated to API.AI Intent
2. Dialogflow Intent roept een Webhook server op via de HEROKU website
3. Heroku Websocket server communiceert met en service laptop via .NET Websocket client
4. Service laptop met .NET UDP Socket communiceert met de 'robot background task'
5. 'Robot background task' met 'ProcessData interface' om de robot taak uit te voeren op sunrise java application server

3.2 Google Home

De Google Home is een draadloze, slimme luidspreker die via spraakherkenningssoftware opdrachten kan opnemen en verwerken. Hierbij wordt gebruik gemaakt van Google Assistant als intelligent personal assistant (IPA).

3.3 Dialogflow

Dit is de applicatie van Google om een agent te ontwikkelen die instaat voor de verwerking van de tekst. Deze software wordt gebruikt om de opdrachten te definiëren. Dit onderdeel wordt verder behandeld in hoofdstuk 4.

3.4 Heroku Websocket server

De Heroku Websocket server is een web applicatie die het mogelijk maakt om een bidirectioneel kanaal op te zetten, om data uit te wisselen tussen de cliënt en de server, in dit geval de computer en de robot. Meer informatie omtrent de websocket server staat onder rubriek

3.5 Service laptop

Op de service laptop is er een programma geïnstalleerd (WebsocketClient) en dient ervoor om de informatie die verkregen wordt via de websocket, door te sturen naar de controller van de robot.

3.6 Controller Robot

De controller van de robot ontvangt de instructies en voert deze uit. Daarnaast stuurt het op zijn beurt informatie terug naar de google home box dat een antwoord terug geeft naar de programmeur.

4 Google Home en Dialogflow

Deze sectie geeft meer informatie omtrent Google Home, de slimme luidspreker van Google en de daaraan gekoppelde software 'Dialogflow' . Het toestel vormt de interface om de spraakopdrachten van de mens op te nemen en zet deze om naar tekst die dan gebruikt wordt voor bepaalde acties [7]. Voor dit onderzoek is er geopteerd om de Google Home te gebruiken omdat de spraakherkenningssoftware van Google geavanceerd is en deze een ruime woordenschat bevat. De prijs voor dit toestel bedraagt ongeveer €150 en dit maakt het een geschikte keuze om te gebruiken voor het onderzoek.

Het toestel heeft zelf al een aantal functies ingebouwd, zoals de tijd of de weersvoorspellingen op te vragen. Deze functies zijn niet belangrijk om de robot te programmeren maar ze geven een goed voorbeeld om met de software te leren werken.

4.1 Dialogflow

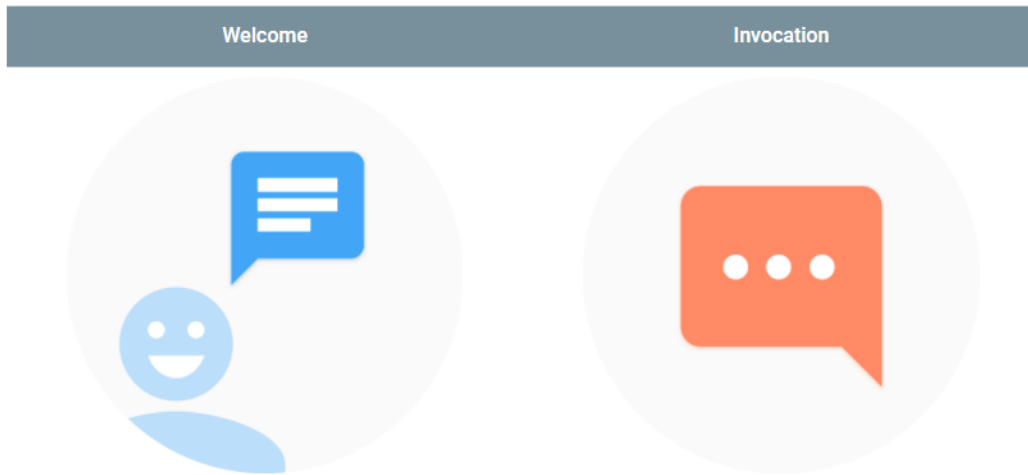
Deze sectie behandelt het programma Dialogflow van Google. Eerst wordt een omschrijving gegeven van hoe een normaal gesprek verloopt. Daarna volgt de uitleg over de programmering met Dialogflow. [8]

4.2 Verloop van een gesprek

Een eenvoudig gesprek met een vraag en antwoord tussen mens en machine heeft een gelijkaardig verloop als bij een normaal gesprek tussen twee personen (persoon A en persoon B). De digitale partner wordt echter een agent genoemd die de rol overneemt van persoon B. Het verloop van een gesprek wordt verder verduidelijkt met enkele figuren.

4.2.1 Welkom

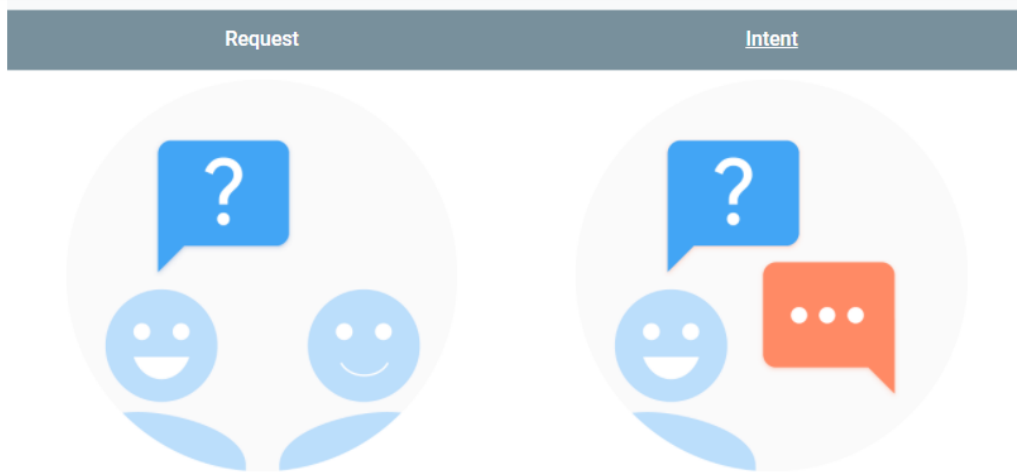
Om een gesprek te starten tussen twee personen wordt een persoon aangesproken met 'hallo' of 'goedendag', eventueel gevolgd door de voornaam. Hierop geeft de aangesproken persoon een welkomstbericht terug. Om een gesprek te voeren met de robot wordt eerst de slimme luidspreker geactiveerd door het commando 'OK Google' of door een gespecificeerd commando. Dit wordt schematisch voorgesteld in figuur 7.



Figuur 7: Welkomstboodschap, overgenomen van
<https://dialogflow.com/docs/getting-started/basics>

4.2.2 Vraagstelling

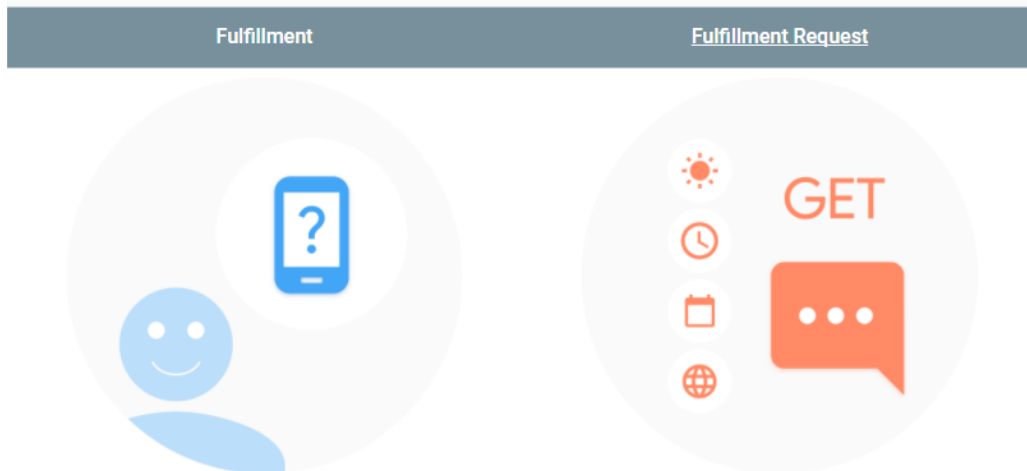
Dan volgt de vraagstelling door persoon A, zoals bv. 'Wat voor weer wordt het vandaag?' Met deze informatie weet persoon B over welk onderwerp de vraag handelt en kan hij de nodige data opvragen. In Dialogflow wordt een intent aangesproken die dit specifieke gedeelte behandelt. Een intent is een pakket van verschillende commando's die verband hebben met hetzelfde onderwerp, bv. 'Wat voor weer wordt het vandaag?'. Een schematisch voorbeeld wordt gegeven in figuur 8.



Figuur 8: Vraagstelling, overgenomen van
<https://dialogflow.com/docs/getting-started/basics>

4.2.3 Verwerking

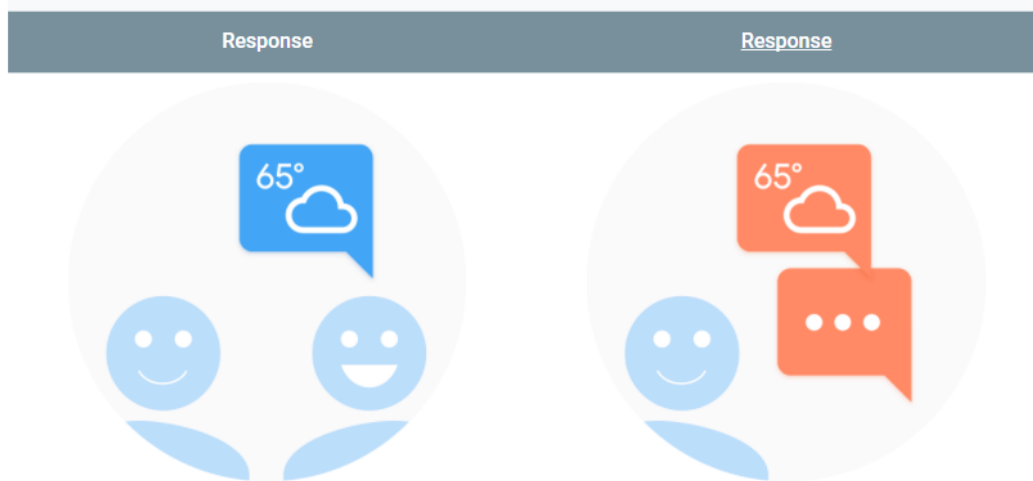
Met de informatie uit de vraag kan persoon B verschillende bronnen raadplegen om een gepast antwoord te formuleren. Via huidige technologie, bv. een smartphone, zoekt persoon B naar een applicatie die informatie biedt over het onderwerp. Dialogflow heeft toegang tot diverse website's om een gepast antwoord te vinden op de vraag. Zie figuur 9 voor een schematisch voorbeeld.



Figuur 9: Verwerking boodschap, overgenomen van <https://dialogflow.com/docs/getting-started/basics>

4.2.4 Afhandeling

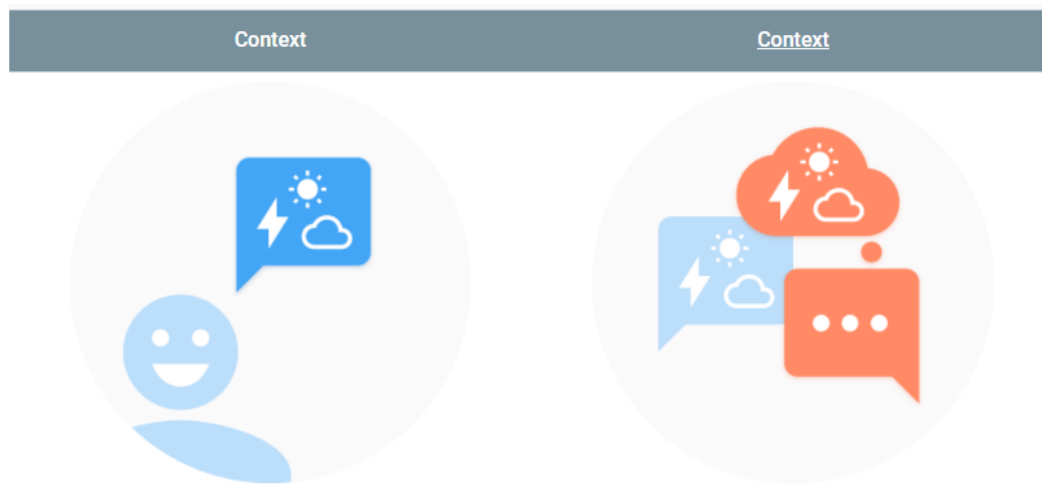
Wanneer persoon B de nodige data heeft bemachtigd, deelt hij deze mee met persoon A die nu een antwoord heeft op zijn vraag. In Dialogflow gebeurt dit evenzo. Als de nodige data gevonden is, wordt deze aangeboden aan de gebruiker. Dit wordt weergegeven in figuur 10.



Figuur 10: Gesprek afhandeling, overgenomen van <https://dialogflow.com/docs/getting-started/basics>

4.2.5 Aanvullend gespreksonderwerp

Wanneer een bepaald onderwerp aangehaald is, kan hier nog extra informatie over gevraagd worden zoals bv. 'En wat voor weer wordt het morgen?' Persoon B weet over welk onderwerp het handelt en gebruikt dezelfde applicatie om de vraag te beantwoorden. Dit is eveneens mogelijk in Dialogflow. De data wordt als het ware een tijd vastgehouden. Zodra deze tijd verstreken is moet de gebruiker de intent opnieuw oproepen. Zie figuur 11 voor een schematisch voorbeeld.



Figuur 11: Aanvullend gesprek, overgenomen van
<https://dialogflow.com/docs/getting-started/basics>

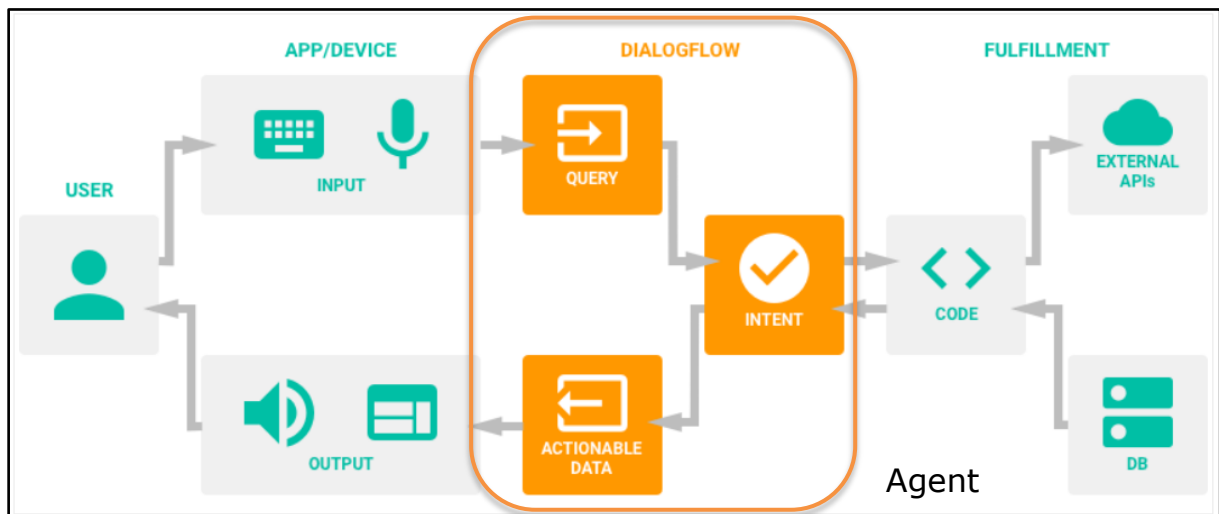
4.3 Functies van dialogflow

Dit onderdeel behandelt de diverse functies van Dialogflow. Deze beschrijving kan ook gevonden worden op de site van Dialogflow waar ook enkele voorbeelden worden gegeven bij de verschillende functies.

4.3.1 Agents

De agent is een module die de instructies van de gebruiker verstaat en hieruit de nodige parameters bepaalt of de juiste acties onderneemt. De agent staat dus tussen de mens en de robot in en kan gezien worden als een digitale tolk. Een schematisch verloop van de dataflow is te zien in figuur 12.

Bij de opstart van de agent wordt deze eerst op de juiste wijze ingesteld. Dit houdt in dat de juiste taal en tijdszone worden ingesteld en wordt er een korte beschrijving gegeven van de functies van de agent. Een agent kan in meerdere talen opereren. De Nederlandse versie wordt nog niet ondersteund maar is nog in ontwikkelingsfase.

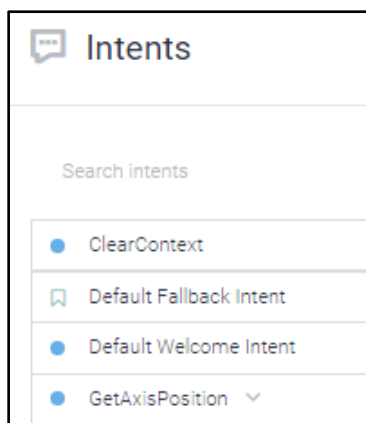


Figuur 12: Schematisch verloop van data verwerking, overgenomen van <https://dialogflow.com/docs/agents>

Google heeft zelf al diverse voorgeprogrammeerde agents ontwikkeld zoals bv. een weer agent. Deze zijn gratis ter beschikking en kunnen ook dienen om de werking van een agent beter te begrijpen.

4.3.2 Intents

Een intent is een link tussen wat de gebruiker zegt en welke actie er moet ondernomen worden. Het is belangrijk om op voorhand de structuur van de intents vast te leggen om overlappingsen te vermijden. Dit vraagt enige ervaring van de ontwikkelaar van de agent. Zie figuur 13 voor een voorbeeld van enkele intents.



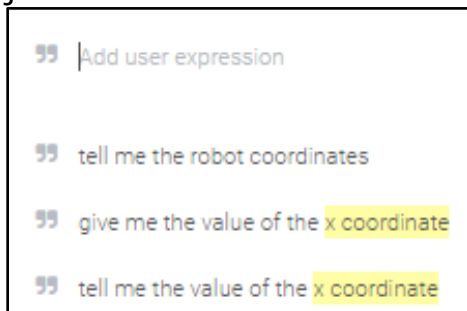
Figuur 13: Voorbeelden van intents, overgenomen van <https://dialogflow.com/docs/intents>

Bij de programmering van de intents zijn er vier belangrijke onderdelen:

1) Training phrases

In dit onderdeel worden voorbeeldzinnen opgegeven van hoe de gebruiker een bepaalde intent kan oproepen.

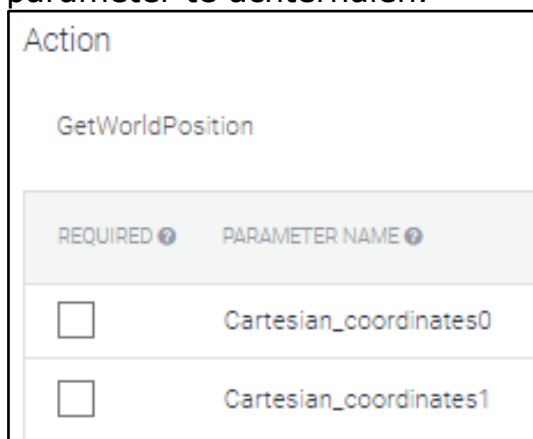
In deze voorbeeldzinnen worden parameters aangeduid die verwacht worden van de gebruiker, zie figuur 14. Het is belangrijk om de voorbeeldzinnen bij de verschillende intents goed van elkaar te onderscheiden anders heeft de software moeilijkheden om de juiste intent te selecteren.



Figuur 14: Voorbeelden van training phrases, overgenomen van <https://dialogflow.com/docs/intents>

2) Action and parameters

Dit onderdeel dient voor de definiëring van een actie naam die voor de applicatie gebruikt wordt om een specifieke actie te ondernemen. Daarnaast worden hier ook de parameters gedefinieerd die kunnen voorkomen bij de voorbeeldzinnen in de vorige sectie. Voor de parameters wordt er ingesteld wat voor soort type dit is, zoals bv. een getal of een datum. Daarnaast kan er ook ingesteld worden als deze parameter een vereiste is. In het geval dat de gebruiker niet genoeg informatie meegeeft is het mogelijk de agent specifieke vragen te laten stellen om de waarde van de parameter te achterhalen.



Figuur 15: Action parameters, overgenomen van <https://dialogflow.com/docs/intents>

3) Events

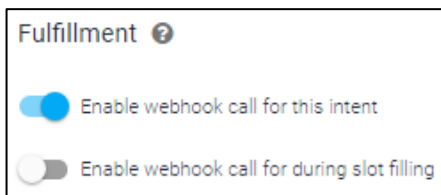
Events worden gebruikt om een intent te triggeren zonder daarbij de voorbeeld zinnen te gebruiken. Zo wordt de standaard welkom intent op deze manier geactiveerd.

4) Response

Dit onderdeel bevat de mogelijke antwoorden die de agent terug geeft aan de gebruiker. Bij de robot-toepassing bijvoorbeeld wordt na elke bewegingscommando een herhaling gegeven van de bewegingsparameters als controle voor de gebruiker.

5) Fullfillment

Hierbij is er de keuze of er een 'webhook call' uitgevoerd moet worden na het doorlopen van de intent. Als deze optie geactiveerd is, zal na het doorlopen van de intent een oproep gedaan worden naar een webhook. De webhook staat tussen Dialogflow en de eindgebruiker en dient om bepaalde gegevens door te sturen. Extra informatie over de webhook vindt u onder rubriek 5.1. Figuur 16 laat zien hoe de webhook call wordt geactiveerd.



Figuur 16: Afhandeling van gesprek, overgenomen van <https://dialogflow.com/docs/intents>

4.3.3 Entities

Entities zijn parameters die de gebruiker zelf kan aanmaken. Dit zijn parameters die specifiek zijn voor de toepassing. Voor de robot toepassing is er een entity aangemaakt met de naam "Direction". In deze entity zijn verschillende parameters opgenomen, bijvoorbeeld x+ of forward, y of left, z+ of up ...

Daarnaast zijn er ook systeem entities die standaard aanwezig zijn in Dialogflow zoals bv. "sys.number" om aan te geven dat de parameter een getal is in een integer formaat.

4.4 Uitwerking programma

Deze sectie geeft een uitgebreid overzicht van de uitwerking van het dialogflow programma. Er wordt telkens een bondige beschrijving gegeven van de diverse intents en waar nodig wordt er een tabel gegeven met een opsomming van de gebruikte parameters.

4.4.1 Default welcome intent

De default welcome intent dient als opstart intent en wordt automatisch aangemaakt bij het creëren van een agent. De intent wordt gestart via een 'WELCOME' event die verzonden wordt als de applicatie wordt opgeroepen. Daarbuiten heeft deze intent geen functie.

4.4.2 Default fallback intent

De default fallback intent wordt ook automatisch aangemaakt en heeft als functie om terug te vallen als de zin niet goed werd begrepen door de software. De intent vraagt dan om de zin te herhalen.

4.4.3.GetAxisPosition

De functie 'GetAxisPosition' dient om een as-positie op te vragen van de robot. De KUKA LBR iiwa robot heeft zeven assen en het is mogelijk om elke stand op te vragen. Een voorbeeld zin hierbij kan zijn: 'Wat is de waarde van as 1?' De parameters die in de zinnen voorkomen worden in tabel 1 opgesomd.

Parameter	Type	Waarde
AxisNumber0	Integer	[1...7]
AxisNumber1	Integer	[1...7]
AxisNumber2	Integer	[1...7]

Tabel 1: Parameters GetAxisPosition

4.4.4 GetWorldPosition

De intent 'GetWorldPosition' lijkt sterk op de vorige intent met het verschil dat nu een cartesische waarde opgevraagd kan worden. Deze waarde heeft betrekking met het center punt van het gebruikte gereedschap, of ook tool centre-point (TCP). Een voorbeeld zin hierbij is bijvoorbeeld: 'Geef me de positie van het X- en Y coördinaat. De nodige parameters bij deze functie worden in tabel twee weergegeven. Voor deze parameters werd er een eigen type gecreëerd om aan te geven dat het om een cartesisch element betreft.

Parameter	Type	Waarde
Direction0	Cartesian_coordinate	[X,Y,Z,A,B,C]
Direction1	Cartesian_coordinate	[X,Y,Z,A,B,C]
Direction2	Cartesian_coordinate	[X,Y,Z,A,B,C]

Tabel 2: Parameters GetWorldPosition

4.4.5 SetAxisPosAbs

De functie 'SetAxisPosAbs' dient om een as van de robot naar een gewenste positie te brengen. Het gaat hierbij om een absolute waarde. Voordat de beweging wordt uitgevoerd zal er eerst om bevestiging gevraagd worden vanuit de robot. Dit ter controle of de gewenste waarde correct is doorgegeven. Daarnaast is het ook mogelijk een beweging te annuleren. Een voorbeeldzin kan als volgt luiden: 'Zet as 1 naar 90 graden.' Hierop krijgt de gebruiker een bevestiging: 'As 1 naar 90 graden?' De beweging wordt dan uitgevoerd als de gebruiker 'Ja' zegt of wordt afgebroken door 'Nee'. De parameters die in de zinnen voorkomen worden in tabel 3 opgesomd.

Parameter	Type	Waarde
AxisNumber0	Axis	[1...7]
AxisValue0	Integer	[0...±180°]
AxisNumber1	Axis	[1...7]
AxisValue1	Integer	[0...±180°]

Tabel 3: Parameters SetAxisPosAbs

4.4.6 SetAxisPosRel

De functie 'SetAxisPosRel' komt overeen met de vorige intent met het verschil dat het hier een relatieve beweging betreft. Dit betekent dat een as van zijn huidige positie met een gewenste waarde wordt verdraaid. De moeilijkheid bij deze functie is om ervoor te zorgen dat de voorbeeldzinnen niet lijken op de vorige intent. Dit probleem is opgelost door specifieke werkwoorden te gebruiken in de voorbeeldzinnen, bijvoorbeeld: 'Draai as 1 met 10 graden.' Ook hier wordt eerst weer om een bevestiging gevraagd om te controleren of de juiste as en de juiste waarde is doorgekomen.

Het is ook mogelijk om deze beweging herhaaldelijk uit te voeren, zonder dat hierbij de robot om bevestiging vraagt. Dit maakt de aansturing vlotter en spaart tijd. De nodige parameters bij deze intent zijn dezelfde als bij de vorige intent.

4.4.7 SetWorldPosAbs

De intent 'SetWorldPosAbs' dient om de robot-TCP naar een absolute cartesische waarde te brengen. Ook hier vraagt de robot eerst om een bevestiging. Een voorbeeld zin kan als volgt gezegd worden: '*Positioneer* naar 100 mm in de X-richting.' De nodige parameters komen aan bod in tabel vier.

Parameter	Type	Waarde
Direction0	Direction	[X, Y, Z, A, B, C]
CartValue0	Integer	[0...±XXX]
Unit0	Unit	[mm, cm, dm, m, inch]
Direction1	Direction	[X, Y, Z, A, B, C]
CartValue1	Integer	[0...±XXX]
Unit1	Unit	[mm, cm, dm, m, inch]

Tabel 4: Parameters SetWorldPosAbs

4.4.8 SetWorldPosRel

De intent 'SetWorldPosRel' heeft ook de functie om de TCP van de robot naar een cartesische positie te brengen maar ditmaal relatief. Dit betekent dat het verschoven wordt vanuit zijn huidige positie. Evenals bij de as-specifieke beweging moet ook hier opgelet worden dat de twee intents van zinsopbouw verschillen. Een voorbeeld zin: '*Beweeg* 10 cm naar links.' De parameters hierbij zijn dezelfde als bij de vorige intent.

4.4.9 SavePosition

De intent 'SavePosition' heeft als functie om de huidige positie van het TCP op te slaan in de robotcontroller. Een voorbeeldzin: '*Sla* deze positie op als 2.' Eveneens wordt hier ook om een bevestiging gevraagd zodat een vorige opgeslagen positie niet zomaar overschreden wordt. De nodige parameter staat vermeld in tabel 5.

Parameter	Type	Waarde
PointNumber	Integer	[1...100]

Tabel 5: Parameters SavePosition

4.4.10 MoveToPosition

Hierbij is het mogelijk om naar een opgeslagen positie te bewegen. Als deze positie niet bestaat is het ook niet mogelijk om naar het punt te bewegen. De beweging wordt als een 'joint' uitgevoerd wat betekent dat de robot het gemakkelijkste pad bepaalt. Ook hierbij wordt om een bevestiging gevraagd. Een voorbeeld zin luidt als volgt: '*Ga* naar positie 1.' De parameter bij deze intent is dezelfde als bij 'SavePosition'.

4.4.11 MoveToHome

De functie 'MoveToHome' dient om de robot naar de 'HOME'-positie te brengen. Dit is een vaste positie voor elke robot. Eveneens wordt hier om een bevestiging gevraagd. Een voorbeeldzin: 'Ga naar de HOME positie.' Bij deze intent zijn er geen parameters vereist.

4.4.12 StopRobot

De functie 'StopRobot' dient om de robot te stoppen als deze in beweging is. Het kan gebeuren dat de robot in zijn pad tegen een object botst en kan dit verhinderd worden door volgende zin: 'Stop de robot.' Het is niet mogelijk om alleen 'Stop' te zeggen omdat hierdoor de Google Home de robot applicatie sluit. Bij deze intent zijn er ook geen parameters nodig.

4.4.13 GetActualTool

De intent 'GetActualTool' vraagt het huidige gereedschap op van de robot die actueel ingesteld is. Het is van groot belang om het juiste werktuig te kiezen omdat de afmetingen afhankelijk zijn van het gebruikte gereedschap en het massatraagheidsmoment hier ook vanaf hangt. Bij deze intent wordt geen parameter mee gegeven.

4.4.14 AttachTool

De functie 'AttachTool' selecteert een bepaald werktuig die in de robotcontroller is opgeslagen. Dit werktuig is dus afhankelijk van de taak en de omgeving van de robot. Een voorbeeldzin om een werktuig te selecteren kan als volgt geformuleerd worden: 'Selecteer de vacuüm grijper.' De robot vraagt dan eerst om een bevestiging zodat het juiste gereedschap gekozen wordt. De parameter die naar de robot wordt doorgestuurd staat in tabel 6.

Parameter	Type	Waarde
GripperType	GripperType	[clamp, vacuum, balloon]

Tabel 6: Parameters AttachTool

4.4.15 ToolsFunction

De functie 'ToolsFunction' maakt het mogelijk om de specifieke acties van het gebruikte gereedschap te activeren/deactiveren. Zo kan een grijper geopend/gesloten worden of kan bij een zuignap de vacuüm aan/uit gezet worden. Het is daarom van groot belang om op voorhand het werktuig te kennen om zo de juiste acties te gebruiken. Een voorbeeld zin om het gereedschap te bedienen kan zijn: 'Sluit de grijper.' Voor deze intent zijn er twee acties geformuleerd: 'OPEN' en 'SLUIT' en afhankelijk van de ingestelde werktuig wordt een andere uitgang aangestuurd. De parameters voor deze intent worden opgesomd in tabel zeven.

Parameter	Type	Waarde
GripperState	GripperState	[open, close]

Tabel 7: Parameters Toolsfunction

4.4.16 Search

Via de intent 'Search' is het mogelijk om de robot 100 mm in de richting van het gereedschap te bewegen tot deze een tegenwerkende kracht ondervindt. Op deze manier is het mogelijk om de robot boven een object te bewegen en dan naar het product te zoeken. Ook is het mogelijk om deze functie te herhalen tot de robot ergens tegen een object aankomt. Een voorbeeld zin voor deze functie te activeren is als volgt: 'Zoek naar object.' De robot vraagt hierbij ook weer om een bevestiging.

4.4.17 Vibrate

De 'Vibrate' functie dient ervoor om de robot te laten trillen rond zijn TCP voor drie seconden. Deze functie is standaard ingebouwd in de robotcontroller en kan voor specifieke doeleinden gebruikt worden. Een voorbeeld zin om deze functie op te roepen kan als volgt: 'Start vibratie.' Eveneens vraagt de robot hierbij om een bevestiging.

4.4.18 RobotHelp

De 'RobotHelp' functie dient ervoor om een onervaren programmeur gedeeltelijk te begeleiden tijdens de spraakprogrammering. Zo vertelt de robot meer over hoe een positie opgevraagd kan worden, naar een gewenste positie kan bewogen worden of hoe de functies van het gereedschap geactiveerd worden. Deze intent kan gestart worden door volgend voorbeeld zin: 'Robot, help mij.' Hierop geeft de robot dan een aantal keuzes die de programmeur kan selecteren om meer informatie te vergaren.

4.4.19 RobotJokes

Als kleine extra is er een intent aangemaakt waar een aantal robot grappen zijn in geformuleerd. Deze functie is aangemaakt om tijdens een lange beweging de gebruiker te entertainen.

4.4.20 ROBOTDIALOG

De 'ROBOTDIALOG' functie dient om een dialoog, welke actief is op de Teachpad van de robot, via spraak te activeren. (Figuur toevoegen van speech dialog). De robot somt de verschillende mogelijkheden op die afhankelijk zijn van het actieve dialoog venster. Naderhand kan de gebruiker een van deze opties selecteren en wordt deze optie geactiveerd nadat een bevestiging wordt gegeven. De parameter voor deze intent is weergegeven in tabel 8.

Parameter	Type	Waarde
Answer	DialogAnswer	[Dialoog afhankelijk]

Tabel 8: Parameter ROBOTDIALOG

4.4.21 ActivateDialog

Deze functie dient om de 'ROBOTDIALOG' intent te activeren. Via de robotcontroller is het (nog) niet mogelijk om de 'ROBOTDIALOG' intent automatisch te laten starten als er een dialoogvenster actief is. Daarom is deze intent aangemaakt zodat de gebruiker de dialoogvoering kan starten. Dit kan uitgevoerd worden door volgend voorbeeld zin: 'Activeer dialoog.'

5 Achtergrond processen

5.1 Heroku Websocket Service

Dit onderdeel geeft een korte beschrijving van de Heroku Websocket Service. De Websocket-protocol wordt gebruikt bij moderne real-time web applicaties [9]. Een bidirectioneel kanaal wordt tot stand gebracht om data uit te wisselen tussen cliënten en de servers. Daarnaast biedt het de flexibiliteit van een TCP connectie en de beveiliging.

Het WebSocket-protocol introduceert een status in een over het algemeen stateloze toepassingsarchitectuur. Het biedt een systeem voor het maken van continue verbindingen met een knooppunt in een stateloos systeem, bijvoorbeeld een webbrowser die een verbinding maakt met een enkel webproces. Daarom is er van elk webproces vereist om de status van zijn eigen WebSocket-verbindingen te behouden. Als applicatiegegevens over de processen worden gedeeld, moet ook de globale status worden gehandhaafd.

Stel je een chattoepassing voor die berichten van een Redis Pub / Sub-kanaal naar alle aangesloten gebruikers pusht. Elk webproces zou een verzameling WebSocket-verbindingen open hebben staan voor actieve gebruikers. Elke gebruiker zou echter niet zijn eigen abonnement op het Redis-kanaal hebben. Het webproces zal een enkele verbinding met Redis onderhouden en de status van elke verbonden gebruiker zal dan worden bijgewerkt zodra binnenkomende berichten aankomen.

Voor de robot applicatie is er een URL aangemaakt met volgend adres: <https://kukabe-google-home-speech.herokuapp.com/Rest>.

De parameters van de verschillende intents worden vanuit Dialogflow naar deze URL gestuurd.

5.2 Service Laptop

Dit onderdeel vertelt kort over het achtergrond programma Websocket Client die de gegevens van de Heroku Websocket Service opvraagt en op zijn buurt naar de controller van de robot doorstuurt. Figuur 17 geeft een voorbeeld van het WebsocketClient programma.

```
28/05/2018 11:51:25.479: DialogFlow --> Robot: >> { "originalRequest": { "source": "google", "version": "2", "data": { "isInSand
box": true, "surface": { "capabilities": { { "name": "actions.capability.AUDIO_OUTPUT" }, { "name": "actions.capability.MEDIA_RESPON
SE_AUDIO" } } }, "inputs": { { "rawInputs": { { "query": "talk with kuka robot", "inputType": "VOICE" } }, "intent": "actions.intent.MAIN
" }, "user": { "lastSeen": "2018-05-28T09:34:57Z", "locale": "en-US", "userId": "ABwppHEiHfahyEhPNazLmqKL3zpbR-3zGLjxJq6Pwwb_oFF
exJH8x8245Ep3JNmlvBewfr2Uawtj6JYENG" }, "conversation": { "conversationId": "1527501084912", "type": "NEW", "availableSurfaces"
: { { "capabilities": { { "name": "actions.capability.SCREEN_OUTPUT" }, { "name": "actions.capability.AUDIO_OUTPUT" }, { "name": "actio
ns.capability.WEB_BROWSER" } } } } }, "id": "fc2c5a30-883b-46b7-8079-4088b7eaf9ac", "timestamp": "2018-05-28T09:51:25.019Z", "lan
g": "en-us", "result": { "source": "agent", "resolvedQuery": "GOOGLE ASSISTANT WELCOME", "speech": "", "action": "input.welcome", "a
ctionIncomplete": false, "parameters": { }, "contexts": { { "name": "google.assistant.welcome", "parameters": { }, "lifespan": 0 }, { "na
me": "actions.capability.audio.output", "parameters": { }, "lifespan": 0 }, { "name": "google.assistant.input.type.voice", "paramet
ers": { }, "lifespan": 0 }, { "name": "actions.capability.media.response.audio", "parameters": { }, "lifespan": 0 } }, "metadata": { "inte
ntId": "031f1226-88ea-4c55-8209-57dac258f67a", "webhookUsed": "true", "webhookForSlotFillingUsed": "false", "nluResponseTime":
1, "intentName": "Default Welcome Intent" }, "fulfillment": { "speech": "this response should not be heard, the robot should se
lect another intent. This could be due to connection problems.", "messages": { { "type": 0, "speech": "this response should not
be heard, the robot should select another intent. This could be due to connection problems." } }, "score": 1, "status": { "c
ode": 200, "errorType": "success", "sessionId": "1527501084912" } } } }
28/05/2018 11:51:25.604: Robot --> DialogFlow: >> { "Action": "SendToWebSocket", "Response": { "followupEvent": { "name
": "ROBOTMAIN" } } }
```

Figuur 17: WebsocketClient

6 Besluit

Dit onderzoek heeft laten blijken dat de spraakherkenning van Google in een gevorderde staat is en dat hiermee een robot kan bestuurd worden. De uitgewerkte functies maken het mogelijk om een demonstratie uit te werken waarbij de robot een product moet oppakken en naar een andere locatie moet brengen. Bij sommige functies, en vooral bij de beweging instructies, is het belangrijk om de instructies duidelijk te definiëren zodat mismatching van de functies vermeden wordt.

Via de demonstratie is ook gebleken dat de aansturing van de grijper een handige functie kan zijn om in de industrie te gebruiken, zeker in het geval de robot en de mens nauw samenwerken. Daarnaast is ook gebleken dat wanneer de spraakbesturing zou gekoppeld worden met handbegeleiding, deze twee aansturingen elkaar goed aanvullen.

6.1 Toekomstig onderzoek

Bij een eventueel toekomstig onderzoek zijn er nog een aantal verbeteringen mogelijk. Zo heeft Dialogflow tijdens het onderzoek een update gekregen naar versie 2 en is het noodzakelijk om enkele veranderingen aan de applicatie te brengen om deze ook naar versie 2 te updaten. Deze veranderingen staan uitgebreid uitgelegd in de handleiding van Dialogflow.

Een ander punt is de verbetering van de bewegingsinstructies. In enkele gevallen kan het voorkomen dat de verkeerde bewegingsfunctie wordt opgeroepen door een foutief gebruik van de gedefinieerde zinnen. Een betere aanpak van deze zinnen kan de mismatching voorkomen en wordt de applicatie robuuster gemaakt ten opzichte van deze fouten.

De uitgewerkte functies zijn maar een klein deel van de functies die een robot kan uitvoeren. Bij een volgend onderzoek kunnen er nog extra functies toegevoegd worden als bijvoorbeeld de aanmaak van een programma waarbij de instructies opgeslagen worden en naderhand dit programma afgespeeld kan worden.

Bibliografie

- [1] „KUKA,” [Online]. Available: <https://www.kuka.com/nl-be>. [Geopend 27 Maart 2017].
- [2] S. Profanter, „Implementation and Evaluation of multimodal input/output channels for task-based industrial robot programming,” arXiv, München, 2015.
- [3] M. K. Brown, B. M. Buntschuh en J. G. Wilpon, „SAM: A Perseperspective Spoken Language Understanding Robot,” *IEEE Transactions on Systems, Man and Cybernetics*, p. 13, 1992.
- [4] B. Akan, B. Çürüklü, G. Spampinato en L. Asplund, „Towards Robust Human Robot Collaboration in Industrial Environments,” IEEE, Sweden, 2010.
- [5] P. Gustavsson, A. Syberfeldt, R. Brewster en L. Wang, „Human-Robot collaboration Demonstrator Combining Speech Recognition and Haptic Control,” in *The 50th CIRP Conference on Manufacturing Systems*, Sweden, 20147.
- [6] T. Ayres en B. Nolan, „Voice activated command and control with speech recognition over WiFi,” Elsevier, Ireland, 2005.
- [7] „Google Home Help,” Google, [Online]. Available: <https://support.google.com/googlehome>. [Geopend 27 September 2017].
- [8] „Dialogflow,” Google, [Online]. Available: <https://dialogflow.com>. [Geopend 27 September 2017].
- [9] H. Team, „Heroku Dev Center,” Salesforce, 12 September 2017. [Online]. Available: <https://devcenter.heroku.com/articles/websockets>. [Geopend 2017 Oktober 2017].

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
Spraakaansturing van een KUKA LBR iiwa R14 robot in combinatie met Google Home

Richting: **master in de industriële wetenschappen: energie-automatisering**
Jaar: **2018**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Schumann, Jondar

Datum: **5/06/2018**