



De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren

Bachelor Scriptie

Auteur	A.B. Stokhof (12086339)
Opleiding	Mens en Techniek - Bewegingstechnologie, de Haagse Hogeschool
Datum	14 juni 2017
Begeleiders	dhr. H. Meulman & mevr. J. Koopman

Onderwijsinstelling

Onderwijsinstelling	: De Haagse Hogeschool : faculteit Gezondheid, Voeding & Sport : opleiding Bewegingstechnologie, Mens & Techniek
Adres	: Johanna Westerdijkplein 75 : 2521 EN : Den Haag
Telefoon	: 070 4458888
Afstudeercoördinator E-mailadres	: mevr. C.A.M. Doorenbosch : c.a.m.doorenbosch@hhs.nl
Afstudeercoördinator E-mailadres	: dhr. A. Lagerberg : a.lagerberg@hhs.nl
Afstudeerbegeleider E-mailadres	: dhr. H.N. Meulman : h.n.meulman@hhs.nl
Afstudeerbegeleider E-mailadres	: mevr. J. Koopman : j.koopman@hhs.nl

Onderwerp

Opdrachtgever Functie E-mailadres	: dhr. O. Hofmann : semiprofessioneel skiër : ophofmann@hotmail.com
Titel:	: De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren
Werkveld Beroepsrol Datum	: Sport : Praktijk gerichte onderzoeker & ontwikkelaar : 14 juni 2017

Student

Naam Studentnummer E-mail Telefoon	: Agneta Brigida Stokhof : 12086339 : 12086339@student.hhs.nl : 06 38327215
---	--

Voorwoord

Dit document is geschreven als afstudeeronderzoek voor de opleiding Bewegingstechnologie aan de Haagse Hogeschool te Den Haag. De focus van dit onderzoek ligt bij de ontwikkeling van een trainingshulpmiddel voor skiërs. Gedurende een periode van 14 weken is er aan dit onderzoek gewerkt. De twee hoofdonderwerpen hierin zijn de ontwikkeling van een meetsysteem met gebruik van druksensoren en de ontwikkeling van een user interface. Het document is bedoeld voor de afstudeercommissie, de docenten van de opleiding Bewegingstechnologie en voor allen die in dit onderwerp geïnteresseerd zijn.

Bij het opstellen van het projectplan werd snel duidelijk dat het doel wat ik voor ogen had, veel was voor de gegeven tijd. En van de twee bovengenoemde onderwerpen had al een periode van 14 weken kunnen invullen en had de kwaliteit van dit onderwerp vrijwel zeker verhoogd. Ik heb er echter bewust voor gekozen om met beide onderwerpen aan de gang te gaan, omdat een enkel onderwerp onvolledig zou zijn voor de gebruiker. Het meetsysteem op zich is voor de gebruiker nog niks tastbaars. Het analyseren van de data zou de gebruiker een eigen invulling moeten geven door bijvoorbeeld gebruik te maken van Excel. Tevens heeft de gebruiker niks aan een ontwikkelde interface wanneer er geen data is om te analyseren. Nu ik in dit onderzoek aan beide onderwerpen heb gewerkt, heeft de gebruiker direct iets tastbaars en kunnen video en data in een enkele interface geanalyseerd worden.

Graag spreek ik mijn dank uit naar mijn afstudeerbegeleiders dhr. H.N. Meulman en mevr. J. Koopman, vakdocenten van de opleiding Bewegingstechnologie. Daarnaast wil ik dhr. Lagerberg, dhr. Schrauwen en dhr. van de Kerkhof bedanken voor hun expertise in de ontwikkeling van het trainingshulpmiddel. Tevens wil ik Othmar Hofmann bedanken voor zijn input binnen dit onderzoek. Jouw kennis over skiën en enthousiasme voor dit onderzoek werken stimulerend.

Den Haag, Juni'17.

Agneta Stokhof

Samenvatting

De ontwikkeling van de trainingsfaciliteiten van sportbonden staat niet stil. Grote organisaties zoals sportbonden beschikken over geavanceerde hulpmiddelen ter bevordering van trainingen en prestaties (1). Binnen de wereld van het skiën zijn deze technologieën nog niet zo geavanceerd. Zodoende heeft de opdrachtgever gevraagd naar een hulpmiddel die in de skisport kan worden gebruikt. In dit hulpmiddel dienen de twee basiscomponenten van het skiën, de centrale balans en de stuurdruk van de alpine balans, te kunnen worden geanalyseerd. Eén van de zes aandachtspunten van de centrale balans is de belasting van de gehele voetzool. De druk die de skiër ervaart, wordt stuurdruk genoemd. Dit is een van de drie componenten van de alpine balans (2). Door middel van druksensoren kan inzicht worden verkregen van de grondreactiekracht en de gewichtsverdeling van de atleet. Voetzool druksensoren kwantificeren verticale krachten van de voet in de skischoen (3). Uit onderzoek (4) blijkt dat de bestaande voetzolen met druksensoren erg kostbaar zijn. Daarom dient er een meetsysteem te worden ontwikkeld dat minder kostbaar is.

Het doel van het onderzoek is de ontwikkeling van een trainingshulpmiddel voor coaches en atleten ter bevordering van de prestaties in de skisport. Met dit trainingshulpmiddel kan de centrale balans en de stuurdruk worden geanalyseerd. Het gehele trainingshulpmiddel, bestaande uit het meetsysteem en de interface, wordt samen met een handleiding geleverd. Algemene randvoorwaarden die vooraf opgesteld zijn: het gehele trainingshulpmiddel moet binnen het budget van € 800,00 moet vallen en in de interface dient de video van de skiër synchroon met de data te kunnen worden afgespeeld. De hoofdvraag die hieruit volgt: *Hoe ziet een trainingshulpmiddel voor skiërs eruit, waarin video en data vanuit druksensoren geanalyseerd kunnen worden?*

In de analysefase van het onderzoek is informatie vergaard betreffende de behoeften en eigenschappen van het trainingshulpmiddel. Aan de hand van een analyse zijn de eisen en wensen van het trainingshulpmiddel vastgesteld. Op basis daarvan is er naar een ontwerp gezocht die hieraan kan voldoen. Het ontwikkelde meetsysteem bestaat uit FlexiForce A201 druksensoren. Door een Arduino worden er twee sensoren per voet uitgelezen en wordt de gemeten data opgeslagen op een SD-kaart. Deze sensoren bevinden zich op de voorvoet en op de hiel van de gebruiker. De data die hieruit volgt dient te worden geanalyseerd in de daarvoor ontwikkelde User Interface. Hierin kan de gebruiker de video van de skiër en data vanuit de voetzolen inladen. Met de play-, pauze- en stopbutton bestuurt de gebruiker de video en de grafieken.

Het meetsysteem is getest op validiteit door middel van zeven oefeningen die zijn uitgevoerd. In de oefeningen komen de centrale balans en de stuurdruk in extreme vormen terug. Uit de resultaten van deze test bleek dat de sensoren onder de voorvoet niet de signalen doorgeven als verwacht. Enkel in een zeer extreme positie werd dit signaal bruikbaar en duidelijk. Dit probleem komt mogelijk doordat de schacht van de skischoen deze eerste krachten opvangt en er geen tot weinig overbrenging op de voet plaatsvindt. Het meetsysteem is in de huidige situatie maar door een enkele gebruiker getest. In de toekomst dienen meer proefpersonen te worden gebruikt om de betrouwbaarheid te verhogen. De interface is door acht potentiële gebruikers getest. Zij hebben de interface gebruikt en hier feedback op gegeven. Hieruit bleek dat de interface duidelijk was voor de gebruiker. Het uitlezen van de datagrafieken was echter niet voor alle proefpersonen even duidelijk. Het toevoegen van een versimpelde weergave zou dit gemakkelijker moeten maken. Een beperking van de huidige interface is dat het met Matlab ontwikkeld is. Dit programma heeft beperkte ontwerp mogelijkheden. Gebruik van andere software zou uitkomst kunnen bieden. Ook zou het de gebruiker gemakkelijker maken wanneer de interface kan draaien zonder afhankelijk te zijn van een ander besturingssysteem, zoals Matlab.

De hoofdvraag van het onderzoek is beantwoord. Het doel van het onderzoek is niet volledig bereikt. De centrale balans en de stuurdruk kunnen niet worden geanalyseerd, omdat de sensor onder de voorvoet geen bruikbaar signaal afgeeft. De interface functioneert wel naar behoren. De video van de skiër en de data vanuit de voetzolen kunnen hierin synchroon worden geanalyseerd.



Inhoudsopgave

Voorwoord	2
Samenvatting	3
Inhoudsopgave.....	4
Inleiding.....	6
Doelstelling.....	7
Randvoorwaarden	7
Onderzoeksvragen	8
Leeswijzer.....	8
1 Analyse	9
1.1 Meetsysteem.....	9
1.1.1 Sensoren	9
1.1.2 Meetbereik	11
1.1.3 Positie	12
1.1.4 Afmeting inlegzool.....	13
1.1.5 Arduino®	14
1.2 User interface	15
1.3 Eisen en Wensen	18
2 Ontwerp	19
2.1 Meetsysteem.....	19
2.1.1 Sensoren	19
2.1.2 Voetzolen.....	20
2.1.3 Arduino	20
2.1.4 Circuit	21
2.2 User Interface	23
2.2.1 Matlab script.....	23
2.2.2 Uiterlijk van de UI	24
2.3 Gebruikershandleiding	25
3 Evaluatie.....	26
3.1 Meetsysteem.....	26
3.1.1 Calibratie	26
3.1.2 Installatie meetsysteem.....	26
3.1.3 Uitvoering meting.....	27
3.1.4 Resultaten.....	28

3.1.5	Discussie	29
3.2	User Interface	30
3.2.1	Resultaten	30
3.2.2	Discussie	31
3.3	Evaluatie Eisen & Wensen	33
4	D-A-C	34
	Discussie	34
	Aanbevelingen	34
	Meetsysteem	34
	Interface	35
	Conclusie	35
5	Bibliografie	36
6	Bijlagen	38
	Bijlage I – Gebruikershandleiding	38
	Bijlage II – Meetprotocol	44
	Bijlage III – Resultaten Meetsysteem	45
	Bijlage IV – Arduino Scripts	81
	Bijlage V – Matlab script	85
	Bijlage VI – Evaluatie persoonlijke leerdoelen	108
	Bijlage VII – Projectplan	110

Inleiding

De ontwikkeling van de trainingsfaciliteiten van sportbonden staat niet stil. Veel sporters beschikken over faciliteiten en hulpmiddelen ter bevordering van trainingen en prestaties. Kleine elektronische apparaten zoals hartslagmeters, stappentellers en GPS-systemen zijn voor iedereen verkrijgbaar (1). Grote organisaties, zoals sportbonden, beschikken over meer geavanceerde soorten technieken. Zo ontvangen hockeycoaches tijdens wedstrijden live feedback van de assistent-trainer, die zich op de tribune bevindt met een tablet waarmee hij de wedstrijd gemakkelijk kan analyseren. Ook krijgt de hoofdcoach op zijn eigen tablet de hartslagen van zijn spelers te zien. Met deze informatie kan hij zijn team direct aansturen. Ook de Nederlandse turners beschikken over videosoftware waarin zij ,tijdens een training, hun oefening direct na uitvoering kunnen analyseren (5). Binnen de wereld van het skiën maakt men ook gebruik van technologieën ter bevordering van prestaties. Deze middelen zijn echter nog niet zo geavanceerd als de technologieën die hierboven genoemd zijn. Hier kwam dan ook de vraag van de opdrachtgever uit voort, om een hulpmiddel te ontwikkelen die in de skisport gebruikt kan worden. De opdrachtgever in dit onderzoek is Othmar Hofmann, semiprofessioneel skiër, werkzaam voor de Nederlandse Ski Vereniging en hoofdleraar van SkiDiscovery.

De twee basiscomponenten van het skiën die de opdrachtgever wil kunnen analyseren met behulp van het trainingshulpmiddel zijn de centrale balans (hierna CB) en de stuurdruk van de alpine balans (hierna AB). De CB zegt iets over de balans, de positie van het zwaartepunt, tijdens het skiën in het sagitale vlak. Een van de zes aandachtspunten van de CB is de belasting van de gehele voetzool. De CB wordt in voor- en achterwaartse richting verstoord, door onder andere oneffenheden in de ondergrond. Wanneer men op de hele voet staat is er minder kans op balansproblemen. Door de hele voetzool te belasten wordt de druk het beste over de ski verdeeld waardoor de ski optimaal glijdt en stuurt. In Figuur 1 wordt dit component van de CB visueel toegelicht.

De AB zegt iets over de balans in laterale richting in het frontale vlak, tijdens het maken van een bocht. De druk die de skiër ervaart wordt stuurdruk genoemd welke een van de drie componenten is van de AB. De term balans wordt gebruikt, omdat skiën dynamisch is (2).

In grote lijnen is het doel van een professionele skiër om zo snel mogelijk te zijn. Binnen de skiwereld heeft men de laatste jaren voornamelijk onderzoek gedaan naar de ontwikkeling en verbetering van materialen en technieken (6,7,3,8,9,10,11,12,13). Er is minder aandacht besteed aan de ontwikkeling van technologieën die de trainingfaciliteiten en middelen verbeteren.

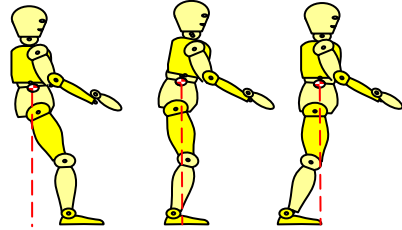
Skiërs moeten voortdurend trainen om de coördinatie van de bewegingen en het gevoel van de snelheid en krachten te kunnen blijven controleren. De trainer speelt hier een belangrijke rol in. Het perspectief tussen de trainer en de atleet verschilt altijd. Omdat atleten de technieken uitvoeren hebben zij een zeker gevoel over de beweging. Trainers observeren de atleten visueel en beoordelen de technieken en bewegingen op basis van hun ervaring en expertise. Bewegingen kunnen makkelijk worden geanalyseerd met standaard videoanalyse software, zoals Kinovea[®]. Krachten kunnen echter niet op deze manier worden geanalyseerd. Om deze krachten te kunnen meten dient er gebruik te worden gemaakt van sensoren. Tevens kunnen sensoren de relatie tussen de trainers visie en de atleet zijn gevoelens over de bewegingen en krachten versterken (1).

Door middel van kracht- en/of druksensoren kan inzicht worden verkregen in de grondreactiekracht en de gewichtsverdeling van de atleet. Twee verschillende meetsystemen zijn eerder gebruikt voor het bepalen van de krachten die op de ski werken: voetzool druksensoren en een 3D-krachtenplaat. Voetzool druksensoren kwantificeren verticale krachten van de voetzool in de skischoen (3). Met gebruik van voetzolen kan de skiër zijn eigen materiaal gebruiken. Tevens beïnvloeden de voetzolen de techniek en uitvoering niet, doordat de zooltjes dun en licht zijn.

Een krachtenplaat wordt tussen de binding in de ski geplaatst en is nadelig voor het gebruik van het eigen materiaal. Tevens beïnvloedt de krachtenplaat de techniek en uitvoering van de skiër.

In een onderzoek van Koop (4) is een overzicht opgesteld van de eigenschappen van bestaande voetzolen met druksensoren.

De voetzolen die in het onderzoek van Koop worden benoemd, zijn heel kostbaar. Daarom dient er dan ook een meetsysteem te worden ontwikkeld, die minder kostbaar is.



Figuur 1 Overzicht van CB posities.

A. Te veel achterovergebogen, ook wel 'achterin staan' genoemd

B. Correcte CB

C. Te veel voorovergebogen

Doelstelling

Het doel van het onderzoek is de ontwikkeling van een trainingshulpmiddel voor coaches en atleten ter bevordering van de prestaties in de skisport met gebruik van druksensoren onder de voet. Met dit trainingshulpmiddel kan de CB en de stuurdruk worden geanalyseerd. Wedstrijdskiërs en skileraren zijn de doelgroep van dit onderzoek.

Om dit hoofddoel te bereiken wordt dit onderzoek onderverdeeld in twee subdoelen.

- 1) De ontwikkeling van een meetsysteem waarmee de druk onder de voet gemeten kan worden;
- 2) De ontwikkeling van een interface waarmee videobeelden en data uit het meetsysteem geanalyseerd kunnen worden.

Het trainingshulpmiddel, bestaande uit het meetsysteem en de user interface, worden samen met een handleiding aan de opdrachtgever opgeleverd.

Randvoorwaarden

Voorafgaand aan het onderzoek zijn, door de opdrachtgever en met behulp van literatuurstudie randvoorwaarden opgesteld waar het onderzoek aan moet voldoen.

Meetsysteem

De CB en de stuurdruk dienen te worden gemeten. De gebruiker kan zelf uit de data interpreteren of er voldaan wordt aan de CB en stuurdruk.

Interface

In de interface dient de video van de skiër te kunnen worden afgespeeld en dient de data van vanuit de voetzolen zichtbaar te zijn. Deze twee componenten dienen synchroon te kunnen worden afgespeeld.

Algemene randvoorwaarde die voor het onderzoek geldt is dat het gehele trainingshulpmiddel binnen het budget van € 800,00 moet vallen. Tevens dient het hulpmiddel toepasbaar te zijn voor een leeftijd van 12 t/m 60 jaar.

Onderzoeksvragen

De hoofdvraag van dit onderzoek luidt als volgt: *Hoe ziet een trainingshulpmiddel voor skiërs eruit, waarin video en data vanuit druksensoren geanalyseerd kunnen worden?*

De volgende deelvragen dienen te worden beantwoord om de hoofdvraag te kunnen beantwoorden en te kunnen voldoen aan de vooraf opgestelde randvoorwaarden:

Meetsysteem

- 1) Welke sensoren dienen gebruikt te worden?
 - a. Hoe kan de data uit de sensoren uitgelezen worden?
 - b. Op welke positie dienen de sensoren te worden geplaatst?
- 2) Hoe valide is het ontwikkelde meetsysteem?
- 3) Hoe dient de data te worden doorgestuurd naar de interface?
- 4) Met welke factoren dient er rekening te worden gehouden tijdens het skiën op een rollerband en in de sneeuw?
 - a. Hoe kunnen deze factoren en hun gevolgen, zoals een sneeuwheuvel, onderscheiden worden?

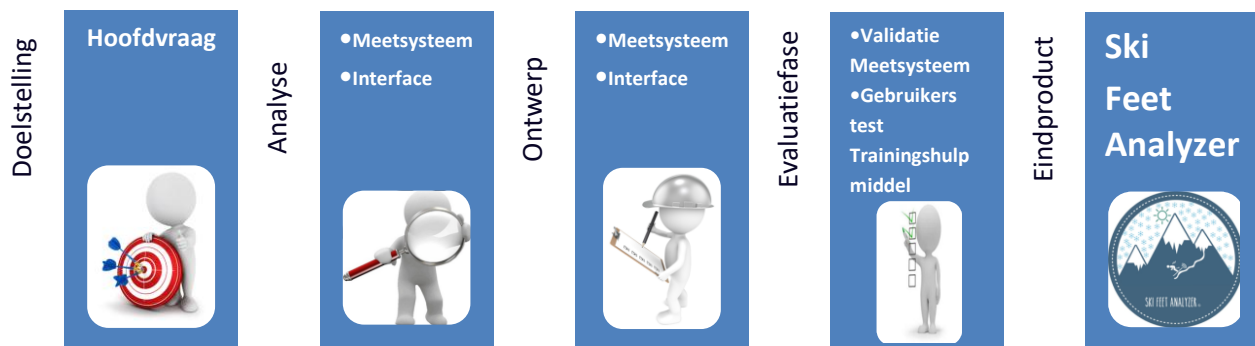
Ontwikkeling interface

- 1) Welke manier van databewerking resulteert in de gewenste waarden van de parameters.
- 2) Hoe dient de data geïnterpreteerd te worden?
- 3) Over welke lay-out en functies dient de interface te beschikken?

Leeswijzer

In Tabel 1 wordt de leeswijzer schematisch weergegeven. In de analysefase wordt onderzocht aan welke eigenschappen het meetsysteem en de User Interface moeten voldoen. Uit de analysefase volgen de eisen en wensen die in hoofdstuk 2 'Ontwerp', worden verwerkt. Het uiteindelijke ontwerp wordt in hoofdstuk 3 geëvalueerd. Het meetsysteem wordt hier gevalideerd en de interface wordt door de gebruikers getest. Het onderzoek wordt afgesloten met de discussie, aanbevelingen en conclusie.

Tabel 1 Leeswijzer





1 Analyse

Sensoren | Meetbereik | Positie | Afmetingen inlegzool | Arduino

Kleuren | Functies | Eisen & Wensen

1 Analyse

De analysefase van dit onderzoek wordt onderverdeeld in twee delen. In het eerste deel wordt onderzoek gedaan naar alle benodigde eigenschappen en bijbehorende onderdelen voor de ontwikkeling van het meetsysteem. In het tweede deel wordt onderzoek gedaan naar alle benodigde informatie voor de ontwikkeling van de interface. Uit deze analysefase volgen de eisen en wensen voor het ontwerp van het trainingshulpmiddel.

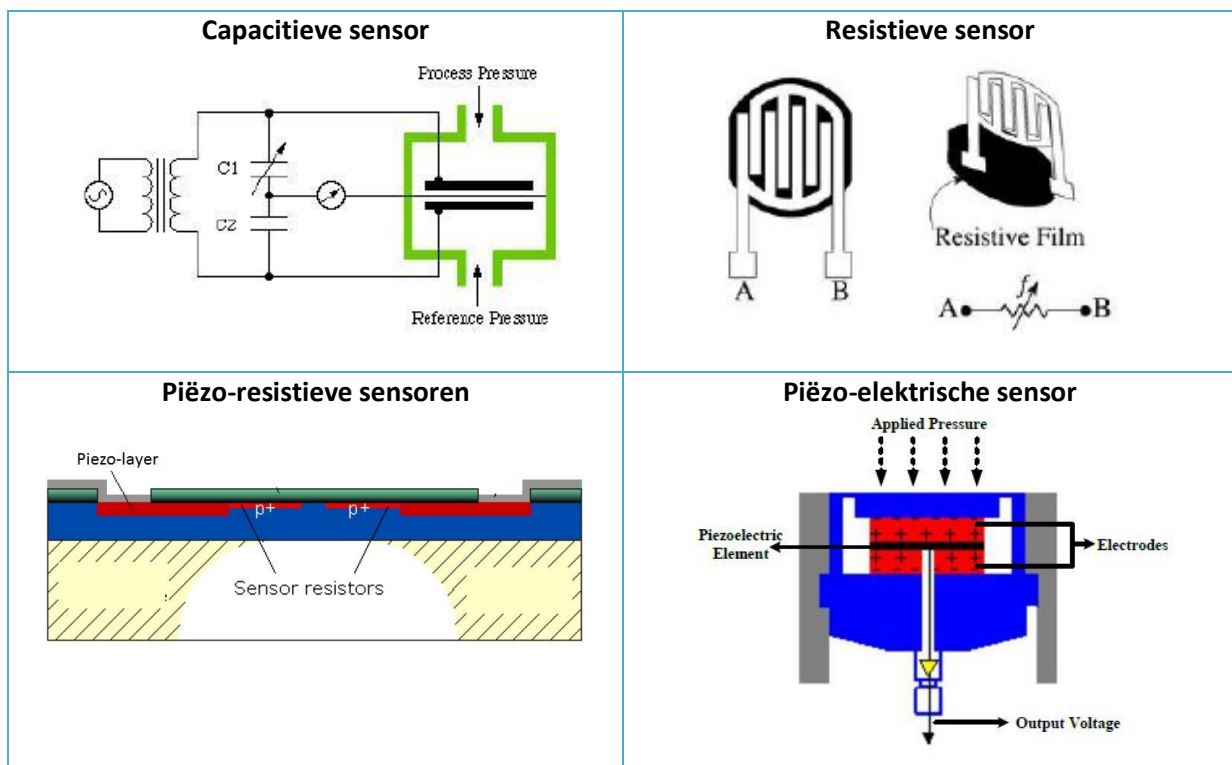
1.1 Meetsysteem

In de volgende paragrafen wordt er onderzoek gedaan naar alle benodigde eigenschappen en onderdelen van het meetsysteem.

1.1.1 Sensoren

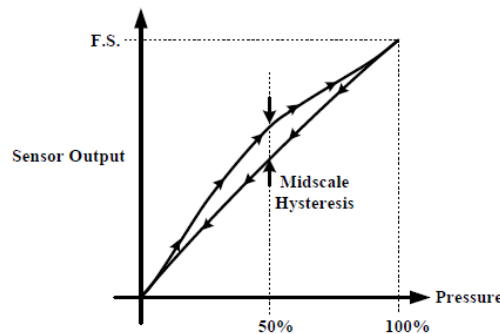
Het meten van plantaire voetdruk kan worden gedaan door middel van optische technieken en elektronische technieken (4,14). Een meting in een schoen wordt uitgevoerd met gebruik van elektronische technieken. Elektronische technieken kunnen worden onderverdeeld in het gebruik van verschillende soorten sensoren. Bij elk van deze technieken wordt fysieke druk omgezet in een elektrische stroom of spanning. Onderstaande beschrijving wordt ondersteund door Figuur 2.

- Capacitieve sensoren maken gebruik van verandering in elektrische capaciteit wanneer er twee platen, gescheiden door een diëlektrische laag, worden samen geperst aan elkaar.
- Resistieve sensoren bestaan uit twee elektroden die gescheiden zijn door een weerstand die verandert wanneer een kracht wordt uitgeoefend.
- Piëzo-resistieve sensoren werken op de verandering van een elektrische weerstand van een speciale inkt met geleidende eigenschappen.
- Piëzo-elektrische sensoren zijn gebaseerd op keramische, niet geleidende materialen. Wanneer er een kracht wordt uitgeoefend ontstaat er een elektrisch veld welke versterkt kan worden en kan worden omgezet in een spanning (15,16).



Figuur 2 Werking van de vier verschillende sensoren

De sleutelspecificaties voor sensorprestatie omvatten nauwkeurigheid, lineairiteit, hysteresis, temperatuur gevoeligheid en drukbereik (16). De nauwkeurigheid van de sensor is de mate waarin het resultaat de werkelijke waarde benadert. Echter is vaak de werkelijke waarde niet bekend en is er een calibratie nodig (17). Nauwkeurigheid is een belangrijke factor bij druksensoren. Voor de meeste sensoren bepaalt de nauwkeurigheid de prijs van de sensor. Hoe nauwkeuriger de sensor, des te kostbaarder deze is (4). De lineairiteit is de maximale afwijking van een rechtlijnige overdrachtsfunctie in het dynamische bereik. De lineairiteit geeft aan wat de maximale afwijking is ten opzichte van een lineair verband tussen de in- en uitgang. Als een sensor perfect lineair is, verandert de ingangsgrootte evenredig met de uitgangsgrootte (17). Wanneer er een toenemende en afnemende druk op de sensor wordt uitgevoerd, kan er een verschil ontstaan tussen de uitgangssignaal van beiden fases. Dit maximale verschil wordt uitgedrukt als het percentage van de maximale uitgangswaarde en wordt hysteresis genoemd. Dit wordt weergegeven in Figuur 3 (16).



Figuur 3 Hysteresis (16)

Ideaal gezien heeft een sensor een hoge nauwkeurigheid, zijn sensoren heel lineair, hebben ze een lage hysteresis, temperatuurgevoeligheid en een breed meetbereik.

Capacitieve sensoren hebben soms de neiging om na het ontladen van de sensor, traag terug te keren naar de nul stand (4). Dit maakt dat dit type sensoren niet geschikt is voor dit onderzoek. Een nadeel van piëzo-elektrische sensoren is hun overgevoeligheid voor temperatuur. Vanwege het werkveld waarin het te ontwikkelen meetsysteem in wordt gebruikt, is het gebruik van uitgesloten.

Een andere belangrijke eigenschap van de sensor is de dikte van de sensor. Omdat de sensor met gebruik van een zool onder de voet wordt geplaatst en deze de atleet niet mag beïnvloeden moet de sensor heel dun zijn.

Eis	De sensoren van het meetsysteem zijn piëzo-resistief of resistief.
Eis	De sensoren hebben zijn toepasbaar in een omgevingstemperatuur van -20°C t/m + 50°C
Wens	De sensoren hebben een zo laag mogelijke lineairiteit.
Wens	De sensoren hebben een zo laag mogelijke hysteresis.

1.1.2 Meetbereik

In 2011 heeft Nakazato (18) een onderzoek gedaan naar grondreactiekrachten tijdens het skiën. In dit onderzoek worden uitkomsten van een krachtenplaat vergeleken met de uitkomsten van een veel gebruikte drukzool van Novel (Pedar, Novel, München - Duitsland). In Tabel 2 worden de uitkomsten van de Novel drukzolen weergegeven. Twintig expert skiërs hebben deelgenomen aan dit onderzoek waarbij er geskied werd volgens twee verschillende technieken op twee verschillende ondergronden. De vlakke piste heeft een hellingsgraad van 15 graden en de steile piste een helling van 23 graden. De exacte snelheden van de skiërs die zijn gemeten, zijn niet bekend. Wel is bekend dat de snelheden op de steile pisten hoger zijn dan op de vlakke piste.

De eerste techniek betreft het skiën van korte bochten met een kleine radius met een actieve verticaal beweging¹. De tweede techniek is een schoolse techniek met een lange radius en een minder actieve verticaalbeweging.

¹Intermezo; de verticaalbeweging wordt gezien als het gaspedaal tijdens het skiën. Het uitsturen van de bocht begint met het buigen van de enkel en daarbij ook het buigen van de knie. Deze beweging zet druk op de skischoen welke de kracht overbrengt naar de ski. De enkels en knieën zijn dicht bij de ski en kunnen dus in een kort tijdsbestek, heel direct invloed uitoefenen op de ski (2).

Door het binnenbeen, ook wel berg ski genoemd, te buigen vanuit de enkel beweegt het lichaamsswaartepunt omlaag en neemt de kracht op het buitenbeen, ook wel dal ski genoemd, toe. Een actieve verticaalbeweging heeft een grote bewegingsuitslag waardoor de stuurdruk hierdoor groter is dan bij een schoolse verticaalbeweging.

Tabel 2 De grondreactiekracht [N/lichaamsgewicht] uitkomsten gemeten met de Novel drukzool tijdens drie verschillende fases in twee verschillende stijlen om twee verschillende pistes.

Techniek 1; korte radius actieve verticaalbeweging(dynamisch)

Techniek 2; grote radius en weinig verticaal beweging(schools) (18)

Expert groep					
		Techniek 1		Techniek 2	
		[Newton/Lichaamsgewicht]			
Fase	Variabele	Vlak	Steil	Vlak	Steil
Buitenbeen	Gem.	.87	.82	.83	.78
	Max	1.16	1.22	1.03	1.06
Binnenbeen	Gem.	.42	.42	.32	.32
	Min	.30	.26	.21	.21
Kantenwissel	Gem.	.60	.49	.58	.49
	Min	.36	.30	.31	.26

Het lichaamsgewicht van de doelgroep in dit onderzoek heeft een minimale waarde van 30 kg (P5 leeftijd van 12 jaar jongens). Het maximale lichaamsgewicht van de doelgroep bedraagt 105 kg (P95 20-60 jaar man). Niet alleen het lichaamsgewicht bepaald de totale massa van de skiër, maar ook het materiaal dat gebruikt wordt. Het materiaal dat de skiër nog met zich mee bedraagt, exclusief ski's en schoenen, +- 3kg.

Zoals te zien is in Tabel 2 is de maximaal gemeten waarde 1.22 N/lichaamsgewicht.

Dit betekend dat de maximale kracht, gemeten onder de voet van het buitenbeen op een steile piste van een skiër met een totaal gewicht van 110 kg, 1317 newton is.

$$\begin{aligned} \text{maximale grondreactiekracht}[N] &= \text{lichaamsgewicht} \cdot g \cdot 1.22 \\ \text{maximale grondreactiekracht}[N] &= 110 \cdot 9.81 \cdot 1.22 \\ \text{maximale grondreactiekracht}[N] &= 1317N \end{aligned}$$

Aan de hand van de oppervlakte van de voet, is het mogelijk om de kracht te berekenen die op de sensor werkt. In Tabel 3 wordt het oppervlakte van de voet berekend. De doelgroep van dit onderzoek wordt hierin gedefinieerd van p5 t/m p95 meet een leeftijd van 12 t/m 60 jaar.

$$\text{Voet oppervlakte} = 0.75 \cdot (\text{Voetlengte} * \text{Voetbreedte})$$

Tabel 3 Oppervlakte van de voet

Categorie	Geslacht en percentiel	Voetlengte	Voetbreedte	Oppervlakte
12 jaar	Meisje p5	210 mm	78 mm	122 cm ²
	Jongen p5	222 mm	83 mm	138 cm ²
	Meisje & Jongen p5	216 mm	79 mm	128 cm ²
20-60 jaar	Vrouw p95	260 mm	100 mm	195 cm ²
	Man p95	293 mm	111 mm	243 cm ²
	Vrouw & Man p95	286 mm	109 mm	234 cm ²

De grootste voet oppervlakte van de doelgroep bedraagt 234 cm² (P95, 20-60 jaar). Het kan zijn dat deze maximale kracht niet over de gehele voet wordt verdeeld, maar op een deel komt staan. Het oppervlakte van de voet is daarom verdeeld in derden. Wanneer alle kracht op alleen de hiel komt te staan, is de druk in de hiel 16.3 N/cm².

$$\begin{aligned} \text{max.kracht op hiel per cm}^2 &= \frac{1317N}{\frac{1}{3} 243 \text{ cm}^2} \\ \text{max.kracht op hiel per cm}^2 &= 16.3N/\text{cm}^2 \end{aligned}$$

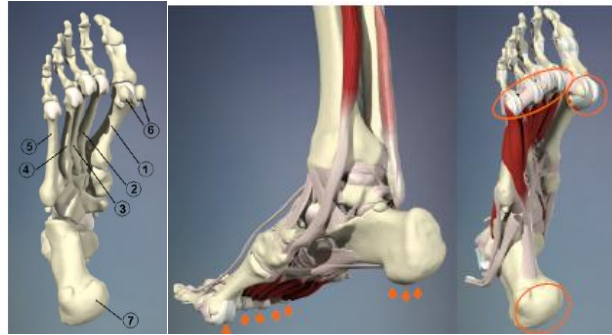
Eis Het meetsysteem moet een minimum meetbereik hebben van 16.3 N/cm²

1.1.3 Positie

Belangrijk voor het ontwerp van de voetzolen is de positie van de sensoren. De positie hiervan is bepaald vanuit de anatomie van de voet. De belangrijkste krachtoverdracht op de voet heeft zijn oorsprong in de onderbeen. Bij de enkel worden deze krachten verdeeld in drie kleinere krachten.

De eerste kracht loop recht door de calcaneus, de tweede door de eerste metatarsaal en de derde door de tweede t/m de vijfde metatarsaal (19). In Figuur 4a en 4b wordt dit weergegeven.

De voetzool kan worden ingedeeld in vier drukgebieden, zie Figuur 5: de hiel (1 t/m 3), de midden voet (4-5), de metatarsale (6 t/m 10) en de tenen (11 t/m 15). Deze gebieden ondersteunen het meeste lichaamsgewicht en verzorgen de lichaamstabiliteit (16,20).



Figuur 4a links, 4b rechts de anatomie van de voet

In Figuur 4a, de eerste t/m de vijfde teen (1 t/m 5) de mediale en laterale sesambotjes van de hallux(6) en de calcaneus(7).

In Figuur 4b voor de gebieden met de meeste druk in stand weergegeven in oranje (19).



Figuur 5 Anatomische voet gebieden (16,20)

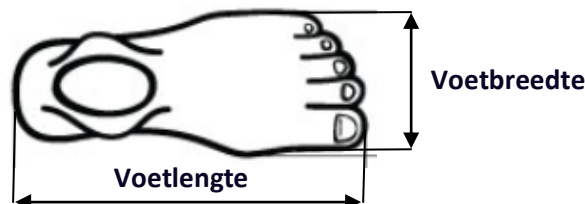
Eis De sensoren dienen te worden geplaatst op de hiel, middenvoet en/of metatarsale van de voet.

1.1.4 Afmeting inlegzool

De doelgroep van dit onderzoek is breed gedefinieerd. Door de verschillende voetlengtes en bijbehorende schoenmaten is het niet mogelijk om één inlegzool te gebruiken. Om voor dit probleem een ontwerp te kunnen maken moet bekend zijn welke voetlengtes en breedtes de doelgroep heeft. De afmetingen van de voet worden weergegeven in Tabel 4. In Figuur 6 worden de afmetingen gedefinieerd.

Tabel 4 Voetlengtes (21)

Categorie	Geslacht en percentiel	Voetlengte	Schoenmaat	Voetbreedte
12 jaar	Meisje p5	210 mm	33	78 mm
	Jongen p5	222 mm	36	83 mm
	Meisje & Jongen p5	216 mm	35	79 mm
20-60 jaar	Vrouw p95	260 mm	41	100 mm
	Man p95	293 mm	46	111 mm
	Vrouw & Man p95	286 mm	45	109 mm



Figuur 6 Definiëring van de afmetingen van de voet.

Eis De voetzolen moeten toepasbaar zijn voor een voetlengte van maat 35 t/m 46.

1.1.5 Arduino®

Een belangrijk onderdeel van het meetsysteem is de microcontroller. Een microcontroller is een geïntegreerde schakeling met een microprocessor die wordt gebruikt om elektronische apparatuur te besturen. De microcontroller is het brein van het meetsysteem en reguleert alle sensoren. Een populaire microcontroller, die veel gebruikt wordt en zeer toegankelijk is, is de Arduino. In dit onderzoek wordt gebruik gemaakt van de Arduino Uno, afgebeeld in Figuur 7. Een Arduino is open-source computer platform gebaseerd op een ATmega328p microcontroller. De Arduino Uno beschikt onder andere over een USB aansluiting, een DC-aansluiting, analoge inputs en digitale in- en outputs. De Arduino kan geprogrammeerd worden door gebruik te maken van de Arduino IDE software.

Het is van de Arduino niet bekend in welke omgevingstemperatuur deze kan werken zonder schade aan zichzelf toe te brengen. Wel is bekend dat een batterij sneller leeg gaat wanneer deze in zeer koude of warme temperaturen gebruikt wordt. Wanneer de Arduino wordt gebruikt wordt er warmte gegenereerd door het elektronische circuit. Verwacht wordt dat de omgevingstemperatuur, extremen daargelaten, geen invloed heeft op de werking van de Arduino.



Figuur 7 Arduino Uno

1.1.5.1 Data opslag SD kaart

De data die door de Arduino gemeten wordt, dient te worden uitgelezen op de computer, zodat de data te bekijken is in de hiervoor ontwikkelde interface. Een Arduino kan data op drie manieren verwerken;

1. Draadloos met gebruik van Xbee.
2. De Arduino direct aansluiten via USB met de computer.
3. Data opslaan.

Tijdens het skiën kan het meetsysteem niet direct aan de computer worden aangesloten. Ook geeft de omgeving van het skiën veel storing en zijn de afstanden te groot om draadloos te kunnen communiceren. Daarom is er voor gekozen om de data op te slaan op een SD kaart.

Met een uitbreiding van de Arduino Uno, een zogenoemde data-logging shield, kan de Arduino zo geprogrammeerd worden dat de data opgeslagen wordt op een SD kaart. De SD kaart interface werkt met FAT16 en FAT32 geformatteerde SD kaarten.

Om meerdere dataset op te kunnen slaan is het handig om de tijd en datum te weten waarop de dataset is opgeslagen. De Arduino heeft een ingebouwde klok genaamd millis. Echter werkt deze klok pas vanaf het moment dat de Arduino over stroom beschikt. Iedere keer wanneer de spanning van 5 Volt over de Arduino weg valt, start de klok opnieuw bij 0.

Tevens weet de Arduino niet welke dag het is. De SD kaart shield beschikt over een Real time Clock die wel de daadwerkelijke tijd en datum bijhoudt, ongeacht of deze is voorzien van stroom.

Eis Het meetsysteem dient te data op een SD kaart op te kunnen slaan.

De Arduino beschikt over zes analoge ingangen waarmee de sensoren uitgelezen kunnen worden. In de SD shield zijn twee van deze ingangen echter bezet met een andere functie. Hierdoor zijn er maar twee ingangen per voet beschikbaar

Eis Per voet dient er een sensor op de hiel en op metatarsale van de voet te worden geplaatst.

1.2 User interface

In dit onderzoek wordt een User Interface (hierna UI) ontwikkeld waarmee de gebruiker de data uit de voetzolen kan analyseren samen met de gemaakte video. De parameters waar de UI feedback over kan geven betreffen de centrale balans en de stuurdruk. Omdat het ontwerp van deze UI van belang is voor dit onderzoek dient de onderzoeker zich te verdiepen in de regels en methodes voor het ontwerpen van een UI.

De Van Dale geeft de volgende betekenis voor het woord 'Interface'; "Het geheel van middelen dat de gebruiker in staat stelt met een computersysteem te communiceren."

Het ontwerp van de UI richt zich op het anticiperen op wat de gebruikers zouden moeten doen om te zorgen dat de interface-elementen gemakkelijk te bereiken, te begrijpen en te gebruiken zijn (22). De kern van een UI-ontwerp ligt bij de logica en de methode (23).

"Alles moet zo simpel mogelijk gemaakt worden, maar niet eenvoudiger." ~Albert Einstein.

Functioneel minimalisme

Het bereik van mogelijk acties mag niet meer dan absoluut noodzakelijk zijn. Het verstrekken van te veel opties kan afbreuk doen aan de primaire functies en kan voor vermindering van de bruikbaarheid zorgen. Het is belangrijk dat de meest voorkomende of belangrijke functies het gemakkelijkst te vinden zijn. Deze functies worden dan ook als standaard ingesteld. De gebruiker heeft echter wel toegang tot geavanceerde functies (24).

Organisatie en structuur

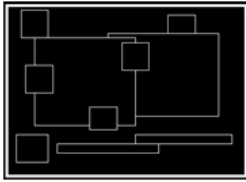
De lay-out dient een duidelijke en consistente structuur te hebben. Consistentie, schermindeling, relaties en bestuurbaarheid zijn belangrijke aspecten in de UI. In Figuur 8 wordt een voorbeeld weergegeven van een geordende lay-out (25).

In de lay-out moet ook de relatie van de onderlinge functies duidelijk zijn. De knop voor het inzoomen van de video moet zo worden geplaatst dat de gebruiker snapt dat deze functie bij de video hoort (24).

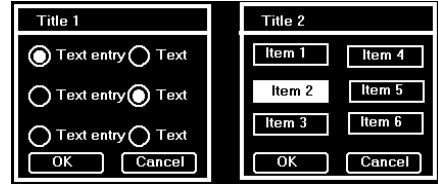
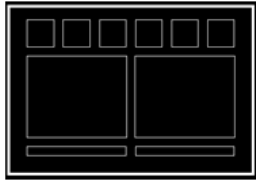
In de UI moet overal dezelfde zogenoemde interne consistentie worden toegepast. Interne consistentie zorgt voor herkenbaarheid en intuïtief gebruik. Hierdoor zal een gebruiker moeiteloos door de UI kunnen navigeren. In Figuur 9 is een voorbeeld te zien van de consistentie. Hetzelfde soort elementen worden getoond op dezelfde plaatsen. Degenen met een andere functie hebben een ander uiterlijk.

Betekenis

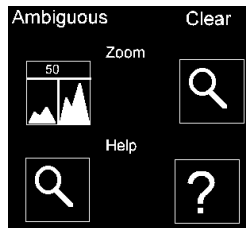
Bij software waarin video's kunnen worden afgespeeld wordt vaak gebruik gemaakt van iconen. Bij het gebruik van iconen moeten deze iconen, niet dubbelzinnig kunnen worden begrepen zoals te zien is in Figuur 10.



Figuur 8 Links: Chaotische lay-out. Rechts: geordende lay-out (25)



Figuur 9 Interne consistentie (25)



Figuur 10 Dubbelzinnige en duidelijke iconen (28)

Kleuren

Kleur is een van de meest complexere elementen van een succesvolle UI. Correct gebruik van kleur kan functies versterken en verduidelijken. De organisatie van de kleuren is belangrijk binnen het ontwerp. De consistentie moet duidelijk zijn. Dezelfde kleuren moeten worden gebruikt bij dezelfde functies. De betekenis van de kleuren moeten herkenbaar zijn voor de gebruiker. Het is dan ook gebruikelijk om twee tot maximaal vijf kleuren in de UI te gebruiken. Dit zorgt er onder andere voor dat de gebruiker de connectie tussen kleur en functie kan leggen (25).

De warmte van de kleuren, het licht, kan worden gebruikt om diepte te creëren. Lichtere tinten lijken dichterbij ons en donkere tinten verder weg. Dit is niet strikt hoe licht werkt, maar dit kan wel zo worden gesteld aangezien de gebruiker de UI gebruikt in een zeer eenvoudige omgeving van een computerscherm. Voorbeeld hiervan is te zien in Figuur 11. Wanneer een object wordt ingedrukt is het verder weg van ons en dus donkerder. Donkere kleuren worden ook gezien als warmer en helderder. Elementen die moeten opvallen hebben een warme kleur. De belangrijkste biedingselementen moeten in ieder geval opvallen (23).



Figuur 11 Links een niet ingedrukte knop. Rechts wanneer de knop is ingedrukt.

Functies

Voor het bepalen van de functies zijn een aantal mediaplayers geanalyseerd. VLC media player, Windows media player, Kinovea, Potplayer en Quicktime zijn onder de loep genomen. Al deze mediaplayers beschikken over de buttons: play, pauze en stop. Ook beschikken al deze mediaplayers over een slider waarmee de gebruiker de video stukje voor stukje kan afspelen. Deze functies worden weergegeven in Figuur 12.



Figuur 12 Voorbeeld van de buttons waarmee de gebruiker de video kan besturen. De slider, play, pauze en stop button zijn hierin verwerkt. In dit voorbeeld verandert de play button in een pauze button zodra de video wordt afgespeeld.

Eisen

De UI beschikt over een play button

Wensen	De UI beschikt over een pauze button
	De UI beschikt over een stop button
	De UI beschikt over slider waarmee de gebruiker frame voor frame door de video kan bewegen.
	De organisatie van de UI is geordend.
	De UI heeft interne consistentie .
	De iconen en functies in de UI hebben een enkele en duidelijke betekenis.
	De UI beschikt over niet meer dan drie hoofdkleuren en twee secundaire kleuren.
	In de UI zijn de functie-kleur verhoudingen gelijk.

1.3 Eisen en Wensen

Hieronder volgt een overzicht van eisen en wensen na aanleiding van de analyse.

Eisen meetsysteem
De sensoren van het meetsysteem zijn piëzo-resistief of resistief.
De sensoren hebben zijn toepasbaar in een omgevingstemperatuur van -20°C t/m $+50^{\circ}\text{C}$
Het meetsysteem moet een minimum meetbereik hebben van 16.3 N/cm^2
De sensoren dienen te worden geplaatst op de hiel, middenvoet en/of metatarsale van de voet.
De voetzolen moeten toepasbaar zijn voor een voetlengte van maat 35 t/m 46.
Het meetsysteem dient te data op een SD kaart op te kunnen slaan.
Per voet dient er een sensor op de hiel en op metatarsale van de voet te worden geplaatst.

Wensen meetsysteem
De sensoren hebben een zo laag mogelijke lineairiteit.
De sensoren hebben een zo laag mogelijke hysteresis .

Eisen Interface
De UI kan feedback geven over de centrale balans en de stuurdruk.
De UI beschikt over een play button
De UI beschikt over een pauze button
De UI beschikt over een stop button
De UI beschikt over slider waarmee de gebruiker frame voor frame door de video kan bewegen.

Wensen Interface
De organisatie van de UI is geordend.
De UI heeft interne consistentie .
De iconen en functies in de UI hebben een enkele en duidelijke betekenis.
De UI beschikt over niet meer dan vijf hoofd kleuren.
In de UI zijn de functie-kleur verhoudingen gelijk.

Eisen Trainingshulpmiddel
In het trainingshulpmiddel kunnen video en data synchroon worden afgespeeld.
Het gehele trainingshulpmiddel valt binnen het budget van € 800,00.
De parameters centrale balans en stuurdruk kunnen in het trainingshulpmiddel worden geanalyseerd.



2 Ontwerp

Opbouw en werking meetsysteem & User Interface

2 Ontwerp

In dit hoofdstuk wordt er een ontwerp gemaakt waarmee aan de vooraf opgestelde eisen worden voldaan en wensen worden vervuld. In verband met de beperkte tijd van het project wordt er één methode worden opgesteld.

2.1 Meetsysteem

Het meetsysteem is opgebouwd uit diverse onderdelen die ieder apart toegelicht zullen worden.

2.1.1 Sensoren

Naar aanleiding van de analyse is de onderzoeker op zoek gegaan naar geschikte sensoren. Hieruit bleek dat deze moeilijk verkrijgbaar zijn in verband met vaak lange levertijden. Hieronder worden in Tabel 5 en 6 de eigenschappen van de twee sensoren beschreven die voor dit onderzoek beschikbaar zijn.

Tabel 5 Eigenschappen van de Interlink FSR 400 sensor.

Eigenschappen van de Interlink FSR 400 sensor	
Dikte	0.3 mm
Lengte	38.1 mm
Breedte	7.6 mm
Sensorisch gebied	Ø5.08 mm – 20.3mm ²
Aansluiting	2-pin man
Meetbereik	20N
Omgevingtemperatuur	-40°C +85°C
Lineairiteit	onbekend
Herhaaldelijkheid	+/- 2%
Hysteresis	+10 %
Drift	<5% bij constante belasting van 111N
Reactietijd	<3 micro seconden
Prijs	€ 10,50 per stuk

Tabel 6 Eigenschappen van de FlexiForce A201 sensor

Eigenschappen van de FlexiForce A201 sensor – Tekscan	
Dikte	0.208 mm
Lengte	203 mm
Breedte	14 mm
Sensorisch gebied	Ø9.53 mm – 73mm ²
Aansluiting	3-pin man (middelste in buiten gebruik)
Meetbereik	440N
Omgevingtemperatuur	-40°C + 60°C
Lineairiteit	+/- 3% 0 tot 50% van maximale belasting
Herhaaldelijkheid	+/- 2.5% op volledige schaal met 80% van maximale belasting
Hysteresis	<4.5% op volledige schaal met 80% van maximale belasting
Drift	<5% bij constante belasting van 111N
Reactietijd	<5 micro seconden
Prijs	€ 25,00 per stuk

Aan de hand van bovengenoemde eigenschappen is er voor gekozen om gebruik te maken van de FlexiForce A201 sensor. Deze sensor is weliswaar hoog in prijs, maar heeft een lage lineariteit en hysteresis. Tevens heeft de sensor een lage drift en is de sensor heel dun.

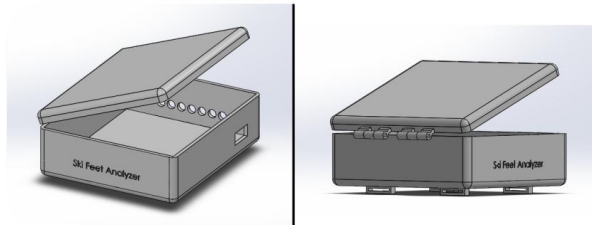
2.1.2 Voetzolen

Om de voetzolen toe te kunnen passen op de gehele doelgroep wordt er gebruik gemaakt van universele zooltjes die per gebruiker op maat geknipt kunnen worden. Deze zolen hebben een bereik van maat 35 t/m maat 46 en worden weergegeven in Figuur 13. De voetzolen hebben een aanschafwaarde van € 0,45 per paar.



Figuur 13 Voetzool die op maat kan worden geknipt.

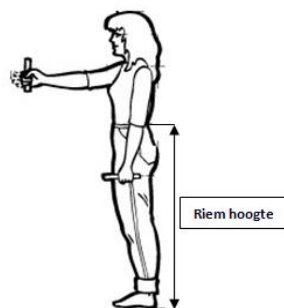
De voetzolen staan in verbinding met de Arduino. Omdat er vanaf beide voeten kabels naar de Arduino worden aangesloten, zullen alle kabels bij elkaar komen rond navel hoogte. Een speciaal ontwikkeld kastje met daarin alle benodigde elektronica kan hier door middel van een riem gedragen worden. Dit kastje is te zien in Figuur 14 en 15. Om de lengte van deze kabels te bepalen is de langste lengte van de doelgroep bepaald. De lengte zoals afgebeeld in Figuur 16 van p95 man 20-60 jaar bedraagt 1231 mm – 1.20m (21).



Figuur 14 Kastje waarmee de meetapparatuur door de gebruiker gedragen kan worden. Het kastje is voorzien van kabelingen en een ingang van de aan- en uitknop. Aan de onderzijde bevinden zich bevestigingspunten, voor bijvoorbeeld klittenband, waarmee het kastje daadwerkelijk aan de riem kan worden bevestigd.



Figuur 15 Het kastje die gedragen wordt



Figuur 16 Definiëring van de riemhoogte. (21)

2.1.3 Arduino

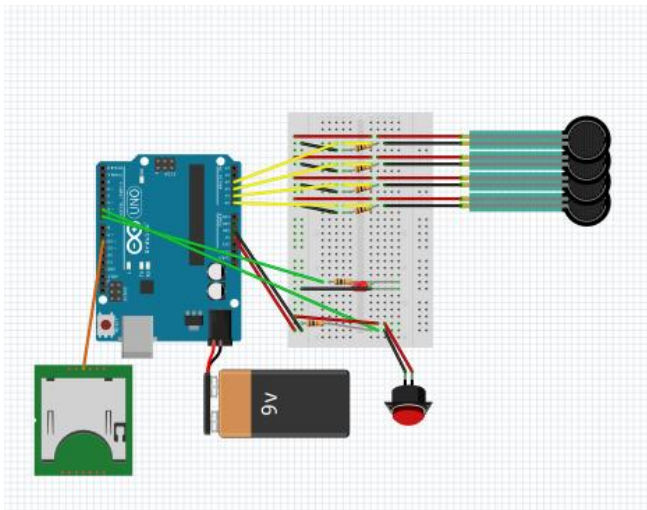
Het script, waarmee de Arduino verteld wordt wat hij moet uitvoeren, is geschreven in Arduino IDE. Dit is de software waarmee gecommuniceerd kan worden met de Arduino. Er zijn twee scripts geschreven die gebruikt dienen te worden bij het gebruik van het meetsysteem. De Arduino dient hierbij te zijn aangesloten op de pc waar de data weergave zichtbaar is. Wanneer het tweede script is ingeladen in de Arduino kan de gebruiker de aan- en uitknop gebruiken en wordt de data opgeslagen op de SD kaart. Beide scripts met uitleg zijn te vinden in Bijlage IV.

Het belangrijkste van dit laatste script is dat de meting pas mag worden uitgevoerd als de schakelaar is ingeschakeld. Hiervoor wordt al gecontroleerd of er een geldige SD kaart aanwezig is. Iedere keer dat de schakelaar wordt in- en uitgeschakeld, wordt er een nieuw bestand aangemaakt waar de data naartoe wordt geschreven. De sensoren worden met een frequentie van 120Hz uitgelezen.

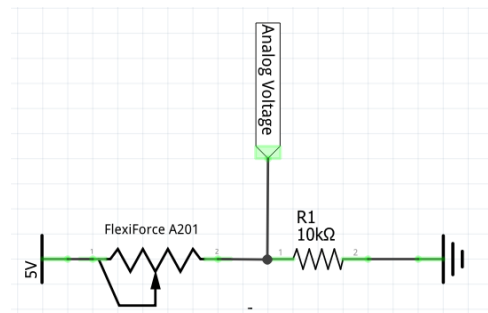
Direct na het eenmalig uitlezen van de sensoren wordt deze data direct overgeschreven naar het aangemaakte bestand op de SD kaart. De sensoren worden uitgelezen op vier analoge ingangen van de Arduino. Een analoge input van de Arduino ontvangt een input van 0 – 1023. Omdat de sensor een spanning van 0 tot 5 Volt heeft, wordt de analoge input van de Arduino omgezet naar voltage. 1023 komt hiermee overeen met 5000 mVolt. Een schematische weergave van de werking van de Arduino wordt weergegeven in Tabel 7.

2.1.4 Circuit

Het elektrische circuit waarmee de Arduino werkt, is te zien in Figuur 17. Alle sensoren zijn in parallelschakeling op een breadboard geplaatst. Iedere sensor is onderdeel van een eigen spanningsdeler. Deze opbouw van de spanningsdeler is te zien in Figuur 18. De volledige SD-shield is niet weergegeven in Figuur 17. Wel is zichtbaar dat de werking van de SD-shield in verbinding staat met digitale ingang 10 van de Arduino.

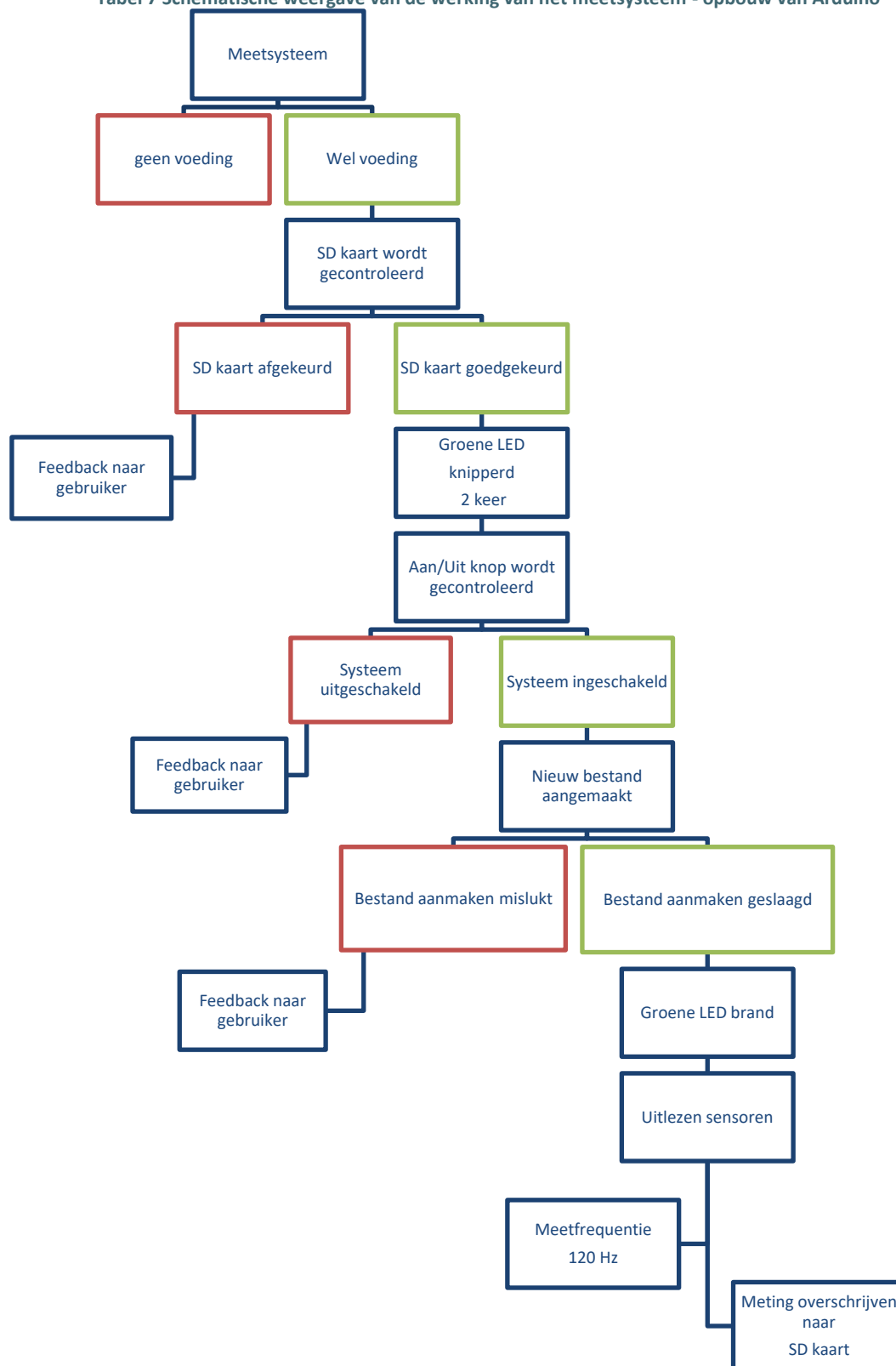


Figuur 17 Het circuit waarin het gehele meetsysteem werkt. De SD-shield is hier niet in verwerkt.



Figuur 18 Spanningsdeler van een sensor

Tabel 7 Schematische weergave van de werking van het meetsysteem - opbouw van Arduino



2.2 User Interface

In de UI wordt er in dit hoofdstuk onderscheid gemaakt tussen het script, waarmee de werking van de UI wordt geprogrammeerd en het uiterlijk van de UI. De UI is in Matlab ontwikkeld met gebruik van de GUIDE functie. GUIDE (GUI ontwikkelomgeving) biedt tools om gebruikersinterfaces te ontwerpen. GUIDE genereert dan automatisch een Matlab code voor het opbouwen van de gebruikersinterface. In deze code kan ontwerper het gedrag van de interface programmeren (26).

2.2.1 Matlab script

Het doel van het gehele matlab script is de gebruiker de mogelijkheid geven de video en de data uit het meetsysteem te analyseren. Hiermee kan de gebruiker uitspraken doen over de centrale balans en de stuurdruk van de skiër.

2.2.1.1 Werking van de UI

Samen met de opdrachtgever zijn de functies bepaald. De volgende functies zijn toegepast in de UI en zijn van belang bij het analyseren van de data en de video:

- Het apart inladen van de video en de data door middel van het klikken op een button.
- Het kunnen knippen van de video en data om het synchronisatie moment uit te kiezen. Dit kan zowel voor de video en voor de data.
- De video heeft de volgende buttons: play, pauze en stop. Zodra zowel de video als de data zijn geknipt, werken de callbacks van de buttons voor zowel de video als de data.
- Zowel de video als de data hebben een eigen slider. Voor het synchronisatie moment werken deze sliders apart van elkaar.
- In de grafieken die worden weergegeven is een gestippelde verticale lijn zichtbaar die overeenkomt met het beeldpunt dat wordt afgespeeld. Deze lijn kan worden aangestuurd door de slider van de data.
- Gebruiker kan aangeven welke gegevens zichtbaar dienen te zijn. Alle data is zowel gecalibreerd als genormaliseerd beschikbaar en kan worden weergegeven in verschillende grafieken.

Hieronder worden de belangrijkste punten benoemd over de opbouw van het Matlab script waarmee de UI is geprogrammeerd. De gehele uitleg van het script is terug te vinden in Bijlage V

2.2.1.2 Set- en getappdata

In de GUI functie in Matlab worden outputs niet bewaard. Dit in tegenstelling tot een standaard Matlab file, ook wel .m file genoemd. Om outputs te bewaren en bij meerdere callbacks te gebruiken, wordt er in het script gewerkt met set- en getappdata. Om een output te bewaren wordt deze aangemaakt met de setappdata. In een andere callback moet deze output worden aangevraagd met de getappdata, zodat de output ook gebruikt kan worden in de gekozen callback.

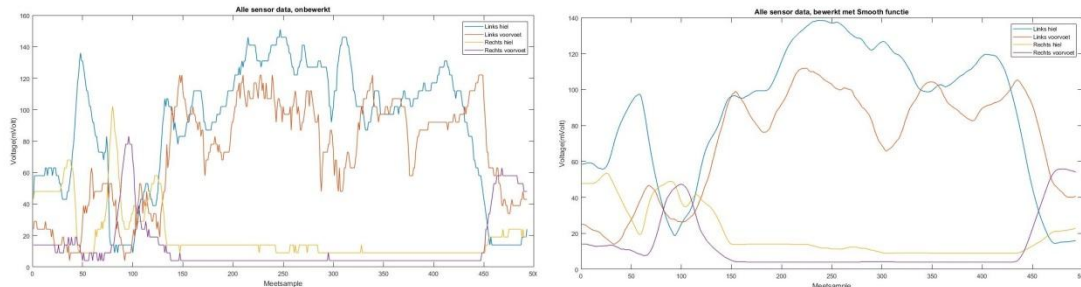
2.2.1.3 Status - weergave

De gebruiker heeft de mogelijkheid om de weergave van de datagrafiek te veranderen. In het script wordt dan ook gewerkt met zogenoemde statuswaardes.

Bijvoorbeeld: als de gebruiker de linkervoet aanvinkt krijgt linkervoet status 1. In de callback van de plot button wordt vervolgens deze waarde opgevraagd met de getappdata functie. In deze callback wordt vervolgens beschreven wat het programma moet doen als bijvoorbeeld de status van de linkervoet gelijk is aan 1 en de rechtervoet uitstaat en status 0 heeft.

2.2.1.4 Smooth functie

De data wordt enkel bewerkt met een zogenoemde 'smooth' functie. Deze functie berekend met een stapgrootte een gemiddelde over de elementen van de vector. De stapgrootte, ook wel vensterlengte genoemd, wordt heuristisch bepaald (27). Door gebruik te maken van de smooth functie, is het voor de gebruiker eenvoudiger om de data af te lezen. In Figuur 19 worden twee grafieken afgebeeld die de werking van de smooth functie weergeven.



Figuur 19 Weergaven van de werking van de smooth functie. In de linker grafiek de onbewerkte data. In de rechter grafiek wordt dezelfde data weergegeven, bewerkt met de smooth functie. Op de x-as de meetsamples en op de y-as voltage in microvolt.

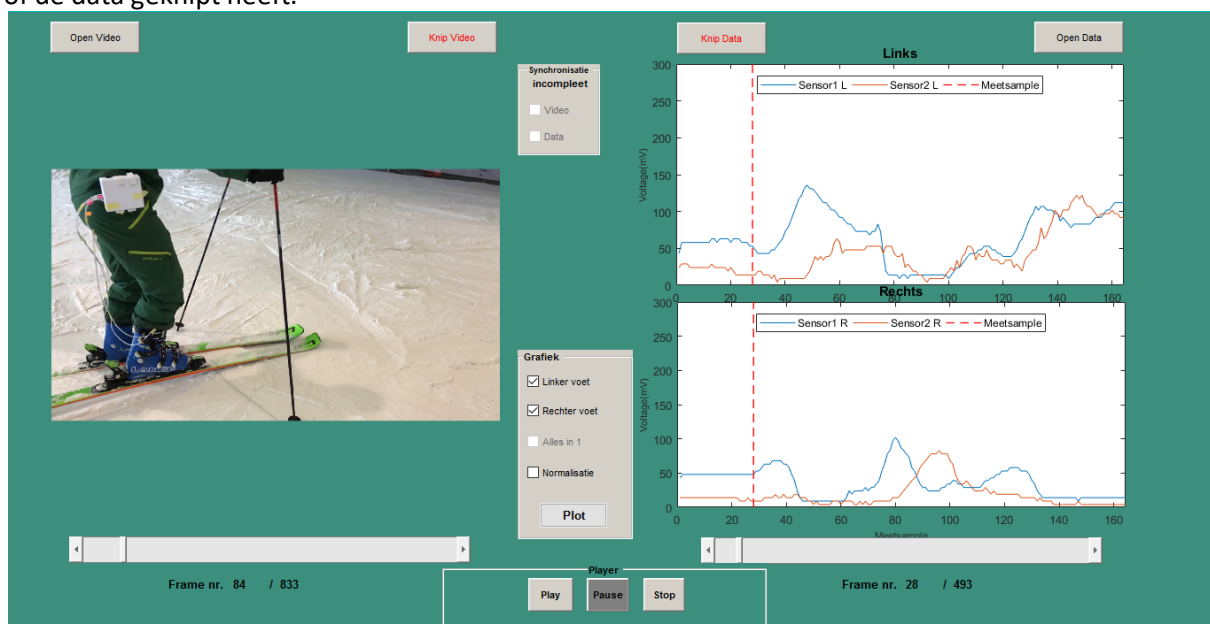
2.2.2 Uiterlijk van de UI

2.2.2.1 Lay-out

De lay-out van de UI is bepaald aan de hand van de gekozen functies. Alle functies worden toegelicht in paragraaf 2.2.2.1. In Figuur 20 wordt de ontworpen UI weergegeven.

Aan de linkerzijde van het scherm bevindt zich de video en de bijbehorende slider. Aan de rechterzijde worden de grafieken weergegeven en is de bijbehorende slider zichtbaar. Buttons voor het openen en knippen van de video en de data bevinden zich boven de video- en grafiekweergave. Vanwege de indeling van de UI en de beperkte ruimte die beschikbaar is, staat de button Group met daarin de checkboxes voor het plotten van de grafieken gecentreerd in de UI. Dat deze button Group bij de grafieken hoort, wordt duidelijk gemaakt met de titel van de button Group.

De knip buttons worden ondersteund door een button Group met checkboxes. De gebruiker kan deze checkboxes niet zelf in- of uitschakelen. Deze worden ingeschakeld zodra de gebruiker de video of de data geknipt heeft.



Figuur 20 De ontwikkelde UI.

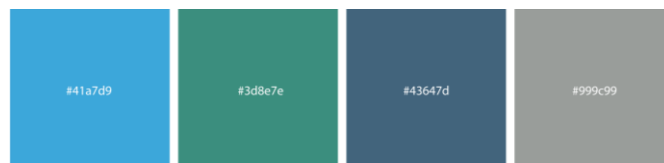
2.2.2.2 Kleuren

Zoals beschreven in hoofdstuk 1.2 zijn de kleuren die gebruikt worden in de UI, de meest complexe van een succesvolle interface. Via Pinterest (28) zijn veel gebruikte kleurschema's gezocht. In Figuur 21 zijn drie kleurschema's te zien. In Figuur 22 zijn de kleuren te zien die in de UI en tevens in dit document gebruikt worden. In de interface wordt ook gebruik gemaakt van het diepte effect na indrukken van de button. Zodra de play, pauze of stop button wordt ingedrukt wordt deze donker grijs en krijgen de andere twee buttons de beginwaarde kleur.

De tekst van beide knip buttons worden indien uitgeschakeld in het rood weergegeven. Zodra de gebruiker de button gebruikt verandert deze kleur naar het groen.



Figuur 21 Drie voorbeelden van kleurschema's (28)



Figuur 22 Definitieve kleuren voor de UI

2.3 Gebruikershandleiding

Voor het gebruik van het ontwikkelde trainingshulpmiddel is er een gebruikershandleiding opgesteld. In deze handleiding worden stapsgewijs alle acties toegelicht die de gebruiker dient uit te voeren bij het gebruik van Ski Feet Analyzer. Deze gebruikershandleiding wordt weergegeven in Bijlage I.



3 Evaluatie

Validatie | Calibratie
Gebruikerstest | Resultaten | Discussie

3 Evaluatie

In dit hoofdstuk worden het meetsysteem en de UI getest. Het meetsysteem zal worden getest door middel van het uitvoeren van een aantal ski oefeningen in de sneeuw. De UI wordt door de opdrachtgever, en tevens ook door andere gebruikers, getest. Zij zullen hun ervaring beschrijven en er zal onderling gediscussieerd worden.

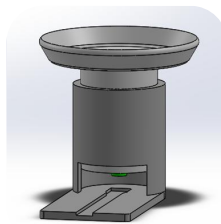
De belangrijkste punten van het meetprotocol voor de validatie van het meetsysteem, worden hieronder genoemd. Het gehele meetprotocol is te vinden in Bijlage II.

3.1 Meetsysteem

Het meetsysteem is met een enkele proefpersoon getoetst door middel van zeven verschillende oefeningen. Deze meting heeft plaats gevonden in SnowWorld te Zoetermeer en is uitgevoerd door de opdrachtgever. De omgevingstemperatuur bedraagt op deze locatie -6°C. De proefpersoon is een zeer hoog geschoolde skiër.

3.1.1 Calibratie

Voor het analyseren van de data dient het meetsysteem eerst gecalibreerd te worden. Calibratie is een methode om de relatie tussen de output van de sensor en de kracht die op de sensor werkt te bepalen. Voor dit proces worden er meerdere bekende krachten een voor een op de sensor geplaatst. Om te zorgen dat de gehele massa van deze bekende kracht gemeten wordt door de sensor dient er gebruik worden gemaakt van hiervoor een speciaal ontwikkeld instrument waarmee de sensor exact op het meetgebied wordt belast. Dit instrument wordt weergegeven in Figuur 23. Op puntje dat de sensor indrukt, is een stukje inlegzool geplakt. Dit materiaal kan voor verandering van data zorgen en dient te worden meegenomen in de calibratie. Door de outputs van de bekende krachten lineair te interpoleren, kan tijdens de uiteindelijke meting bepaald worden welke kracht er op de sensor wordt geleverd.



Figuur 23 Calibratie hulpmiddel waarmee de FlexiForce A201 precies kan worden belast op het meetgebied van de sensor.

3.1.2 Installatie meetsysteem

Het kastje met daarin de Arduino en het breadboard dient zorgvuldig op de gebruiker te worden geïnstalleerd. Op beide voetzolen worden twee sensoren geplaatst. In het midden van de hiel en ter hoogte van metacarpaal I worden de sensoren met tape bevestigd op de zool, zie Figuur 24. Om de data juist uit kunnen lezen is het van belang dat de sensoren op de juiste positie worden geplaatst. De positie van iedere sensor wordt weergegeven in Figuur 24 en Tabel 7.

Sensor	Positie
Sensor 1	Linker voet - hiel
Sensor 2	Linker voet – metacarpaal I
Sensor 3	Rechter voet - hiel
Sensor 4	Rechter voet – metacarpaal I

Tabel 7 Plaatsing van de sensoren



Figuur 24 Plaatsing van de sensoren op de voetzolen.

Zodra de sensoren op de voetzolen zijn geplaatst wordt het signaal getest met het daarvoor ontwikkelde script waarmee de Arduino de meetdata direct weergeeft op seriële monitor van Arduino IDE. De voetzolen zijn op dit moment nog niet in de skischoen geplaatst. Zodra deze meting goed wordt gevonden worden de voetzolen voorzichtig in de skischoen geplaatst. Afhankelijk van de schoen is mogelijk om eerst de binnenschoen uit de schacht te halen en de voetzolen eerst in de binnenschoen te plaatsen. Zodra ook dit signaal getest is kan de gebruiker de binnenschoen in de schacht van de skischoen plaatsen. Wederom wordt het signaal getest. Voordat de gebruiker het kastje met alle elektronische apparatuur bevestigd aan zijn of haar riem, dient eerst het Arduino script te worden geüpload waarmee de data van de sensoren op de SD kaart worden opgeslagen. In Figuur 25 wordt de gebruiker weergegeven met het dragen van het meetsysteem.



Figuur 25 Het meetsysteem waarmee de oefeningen worden uitgevoerd. Rechts op de heup is het kastje bevestigd. Aan de bovenzijde van het kastje bevindt zich de aan- en uitknop.

3.1.3 Uitvoering meting

De volgende zeven oefeningen zijn door de proefpersoon een voor een uitgevoerd:

1. Glijden in de vallijn* - centraal
2. Glijden in de vallijn - volledig achterin
3. Glijden in de vallijn - volledig voorin
4. Glijden in de vallijn - optillen linkerbeen
5. Glijden in de vallijn - optillen rechterbeen,
6. Schools skiën - optillen bergbeen
7. Schools skiën:
 - a. Drie bochten centraal
 - b. Drie bochten achterin
 - c. Drie bochten voorin
 - d. Drie bochten centraal.

De oefeningen worden afgebeeld in Tabel 8.

*glijden in de vallijn is in de skiwereld een veel gebruikte term. De vallijn is de rechte lijn naar beneden, berg afwaarts.

Tabel 8 Overzicht van de zeven oefeningen.

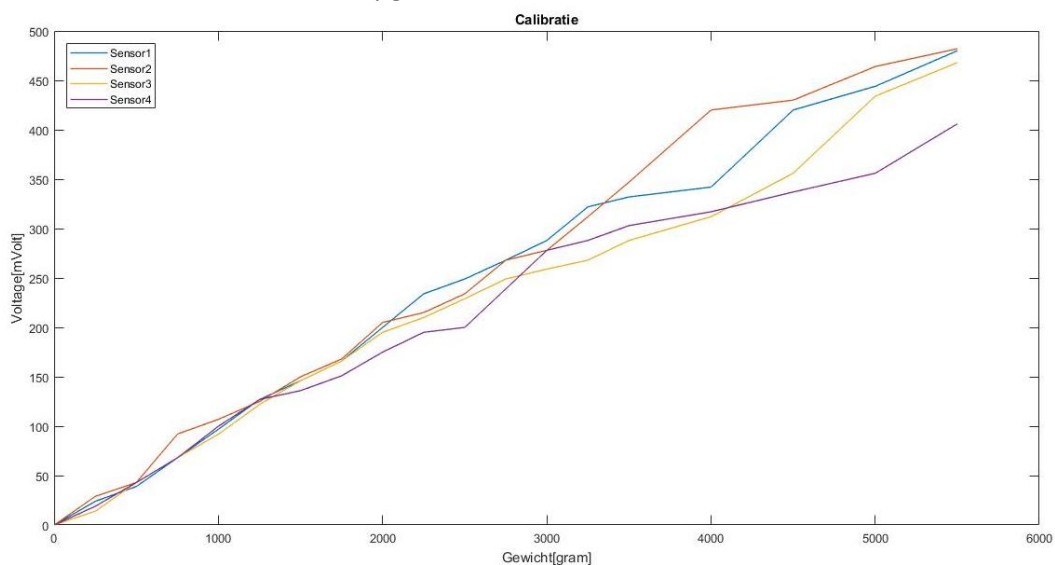
1.Glijden in de vallijn Centraal	2.Glijden in de vallijn Achterin	3.Glijden in de vallijn Voorin	4.Glijden in de vallijn, optillen linkerbeen
			
5.Glijden in de vallijn, optillen rechterbeen	6.Schools Optillen bergbeen	7.Schools CAVC	
			

Om het geheel zo gebruiksvriendelijk mogelijk te maken is er voor gekozen om de oefeningen op beeld vast te leggen met een smartphone welke kan filmen met een frequentie van 120Hz. Smartphones als de Iphone 7 en de Samsung Galaxy 7 hebben deze functies.

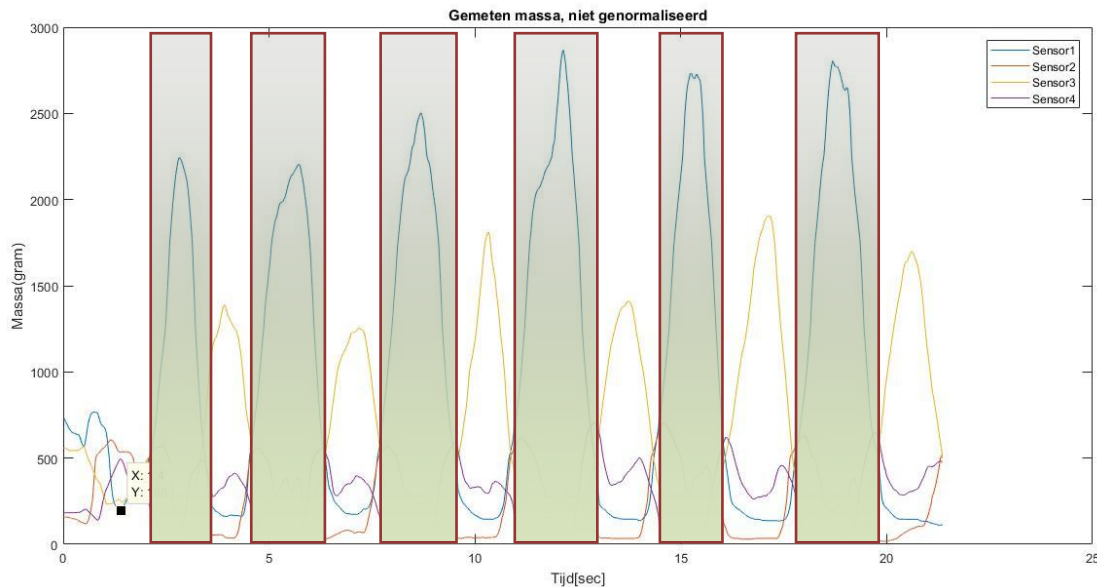
Voordat iedere oefening wordt gestart, wordt eerst een synchronisatie moment uitgevoerd. Het linker- of rechterbeen wordt kort opgetild en weer neergezet. Deze beweging is op de video duidelijk terug te zien.

3.1.4 Resultaten

In Bijlage II worden alle resultaten in vijf verschillende bewerkingen weergegeven. In Figuur 26 wordt de grafiek van de calibratie weergegeven. In Figuur 27 is een voorbeeld van deze resultaten te zien. In deze grafiek de gecalibreerde weergave van oefeningen 6; het schools skiën met optillen van het binnenbeen. De gemarkeerde vlakken weergeven de bochten naar rechts waarbij de skiër op zijn linkerbeen staat en het rechterbeen opgetild heeft.



Figuur 26 Calibratiegrafiek
X-as Gewicht[massa]. Y-as Voltage [mVolt].
Afstudeeronderzoek



Figuur 27 Schools skiën met optillen van binnenbeen.
X-as meetsample. Y-as massa [gram]. Synchronisatie op x-as 1.4sec

3.1.5 Discussie

Het doel van het onderzoek is de ontwikkeling van een trainingshulpmiddel voor coaches en atleten ter bevordering van de prestaties in de skisport met gebruik van druksensoren onder de voet. Met dit trainingshulpmiddel kan de CB en de stuurdruk worden geanalyseerd. Onderdeel van dit trainingshulpmiddel is het meetsysteem.

Uit de resultaten van validatietest van het meetsysteem blijkt dat de sensoren onder de voorvoet niet de signalen geven zoals verwacht. Dit is niet enkel omdat dat het signaal veel zwakker is dan de signalen van de sensoren onder de hiel. Waar er in de signalen van het hielen een duidelijk patroon zichtbaar is, is dit bij de voorvoet niet zo. Enkel in een zeer extreme positie is het signaal vanuit de voorvoet duidelijk. Wat wel zichtbaar is wanneer de skiër achterin komt te staan wanneer er rechtdoor wordt geleden en er geen stuurdruk wordt opgebouwd. Zodra er wordt geskied met bochten, en er sprake is van stuurdruk, kan er niet worden geconcludeerd of het signaal van de hiel nu toeneemt omdat de skiër achterin staat of omdat er stuurdruk wordt opgebouwd. Het signaal van de hiel wordt namelijk niet ondersteund door, indien de skiër achterin staat, verlaging van de meetwaarde onder de voorvoet.

Deze signaal zwakte levert niet enkel een probleem op voor het analyseren van de centrale balans. Ook voor de analyse van de stuurdruk geldt dit. Het kan zo zijn dat de gebruiker al eerder de bocht instuurt dan nu zichtbaar is met het signaal vanuit de hiel. Wanneer de skiër voorin zou staan bij het insturen van de bocht wordt dit momenteel niet geregistreerd.

Bovengenoemd probleem komt mogelijk doordat de schacht van de skischoen deze eerste krachten opvangt en daardoor weinig kracht in voet plaats vindt. De stijfheid van de schoen is hier een factor in, welke wordt bepaald door het ski niveau van de gebruiker. Om deze gedachten te versterken dient er een meting plaats te vinden in een normale schoen. Tevens kan dit worden onderbouwt door een meting uit te voeren met verschillende stijfheden van skischoenen per gebruiker.

Niet enkel het signaal van de voorvoet dient te worden benoemd in deze discussie. Ook blijkt uit de resultaten dat een meting met lange duur, in dit geval 40 seconden of meer, niet volledig wordt opgeslagen op de SD-kaart. Hierdoor worden delen van de oefening niet geregistreerd. De oorzaak is niet bekend.

Het meetsysteem is vooralsnog door een enkele gebruiker getest. Dit mede vanwege tijdsdruk en duidelijke problematiek van het meetsysteem. In de toekomst dienen er bij vervolgtesten meer proefpersonen te worden gebruikt om de betrouwbaarheid te verhogen.

Het huidige meetsysteem heeft met de Arduino maar vier kanalen die uitgelezen kunnen worden. Hierdoor kunnen slechts twee sensoren per voet gemeten worden. Bij het gebruik van meer sensoren kan er meer worden gezegd over de drukverdeling van de gehele voet. Om dit moment is dit dus beperkt tot twee sensoren, de voor- en achterzijde van de voet. Daarnaast hebben deze sensoren hebben ook een beperking. Het sensorisch gebied van deze sensor is klein. Wanneer dit kleine gebied niet volledig wordt belast geeft dit storing in het signaal. Wanneer er gekozen wordt voor grotere sensoren kan dit betekenen dat er een groter gebied wordt gemeten, dit is positief. Ook hier is het echter zo dat als niet het gehele sensorische gebied wordt belast, dit storing geeft van het signaal.

De sensoren worden onder een dunne inlegzool geplaatst. Het inbrengen van deze zool in de skischoen kan voor verplaatsing zorgen van de sensoren. Ondanks dat dit signaal getest wordt voor de daadwerkelijke meeting plaats vindt, is dit wel nadelig voor het gebruikersgemak van het systeem. Ook tijdens het skiën kan het gebeuren dat de voetzolen sensor iets verplaatst. Dit werkt voor de data analyse nadelig. Om dit te voorkomen kan er worden getest met een stijver materiaal. Dit stijvere materiaal zorgt echter ook voor een andere drukverdeling. De resultaten van deze testen kunnen vergeleken worden met de huidige voetzolen voor optimalisatie van het meetsysteem.

De deelvragen: 'Met welke factoren dient er rekening te worden gehouden tijdens het skiën op een rollerband en in de sneeuw?' en 'Hoe kunnen deze factoren en hun gevolgen, zoals een sneeuwheuvel, onderscheiden worden?' zijn in het onderzoek niet beantwoord. De evaluatie van het meetsysteem heeft plaats gevonden op een egale sneeuw ondergrond. Een meting op de rollerband en op een sneeuw ondergrond met veel oneffenheden zijn niet uitgevoerd en dienen in een vervolgonderzoek te worden toegepast.

3.2 User Interface

Om de UI te evalueren hebben er twee evaluatie processen plaatsgevonden. Als eerste is er voor de doelgroep een avond georganiseerd waar de toekomstige gebruikers met de UI kunnen werken. Deze avond heeft plaats gevonden op maandag 29 mei 2017 bij SkiDiscovery, te Den Haag. In de groep proefpersonen bestaat uit ervaren en hoog gekwalificeerde skileraren, trainers van wedstrijdskiteams en oud-wedstrijdskiërs. De gemiddelde leeftijd van de proefpersonen bedraagt 36 jaar (SD +- 8 jaar). Het minimale opleidingsniveau van de proefpersonen is HBO-bachelor. Het hoogste opleidingsniveau van de proefpersonen is WO-master.

De totaal acht gebruikers hebben zonder voorafgaand lezen van de gebruikershandleiding de UI gebruikt en hier hun feedback op gegeven. Op het moment van testen is de UI voor 95% volledig werkend. Het synchronon afspelen dan van de video en data was op dit moment nog niet mogelijk.

Als tweede heeft mevr. Jongmans de UI geëvalueerd en beoordeeld. Zij is User Experience Designer bij Mixit: Digital Workplace Design, te Rotterdam. Als deskundige op het gebied van het ontwikkelen van UI's zal zij de ontwikkelde UI beoordelen.

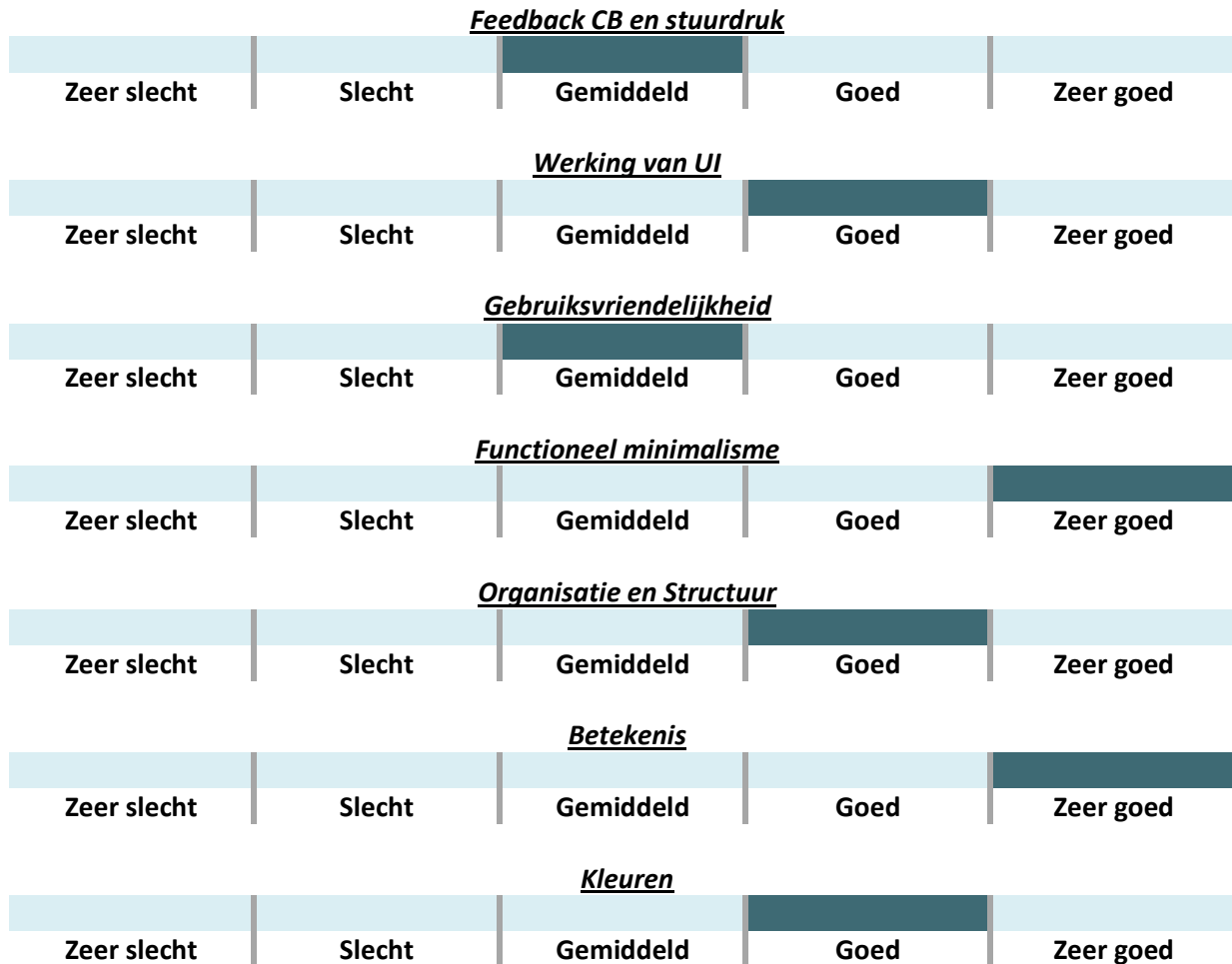
3.2.1 Resultaten

Hieronder volgen de resultaten van zowel de evaluatie van de doelgroep als van mevr. Jongmans.

Doelgroep

Aan de hand van het gebruik van de UI, zijn er een aantal feedbackpunten meermaals genoemd. Tevens is er gezamenlijk met de proefpersonen een schaalverdeling gemaakt op vijf punten ter beoordeling van de UI.

De interface wordt duidelijk in gebruik gevonden. Het is voor de gebruiker duidelijk met welke buttons de actie kan worden uitgevoerd. De kleuren zijn neutraal en het is duidelijk wanneer een functie in gebruik is door onder andere gebruik van kleur wisselingen. Voor gebruikers met geen of weinig ervaring in het uitlezen van grafieken zijn de huidige grafiek moeilijk uit te lezen. Een versimpelde weergave van de data wordt gezien als een functie die nog mist.



Ervaringsdeskundige mevr. Jongmans

"Als User Experience Designer werk ik dagelijks aan de UI van onze klanten. Een consistente en gebruiksvriendelijke UI is essentieel wanneer gebruikers moeten interacteren met een systeem. De voorkant (UI) is daarbij het communicatiemiddel met het systeem en daarmee de enige mogelijkheid om er mee te interacteren. Om dit tot een consistent en gebruiksvriendelijke UI te vormen is kennis en ervaring nodig. Iets wat de onderzoeker wat betreft haar achtergrond niet direct heeft. Toch heeft zij door middel van een goed onderzoek en inleveringvermogen een UI weten te ontwerpen wat functioneel aansluit bij de context. Daarnaast pakt ze feedback die is gegeven goed op en is ze nieuwsgierig naar hoe de UI verbeterd en doorontwikkeld kan worden. Dit zorgt voor een UI die in de essentie de functionaliteiten heeft die, binnen de context van het systeem, doet wat het moet doen"

3.2.2 Discussie

De hoofdvraag van dit onderzoek luidt: *Hoe ziet een trainingshulpmiddel voor skiërs eruit, waarin video en data vanuit druksensoren geanalyseerd kunnen worden?*

Deze vraag is met de ontwikkeling van het trainingshulpmiddel beantwoord. Uit de evaluatie van UI volgt dat het gebruik als positief wordt ervaren. Wel zijn er een aantal punten die in deze discussie benoemd dienen te worden.



De huidige UI is met gebruik van Matlab ontwikkeld. Het gebruik van deze software heeft echter nadelen voor de gebruiksvriendelijkheid van de UI. De huidige interface werkt alleen wanneer de gebruiker Matlab tot zijn beschikking heeft. Zodra de UI kan worden gebruikt met een eigen besturingssysteem, wordt Matlab overbodig. Deze nieuwe software zou bijvoorbeeld als een applicatie kunnen worden gedownload via de Google Play Store van Android of de Appstore van Apple. Tevens heeft Matlab beperkte ontwerpmogelijkheden. Een ander besturingssysteem kan uitgebreidere ontwerpmogelijkheden bieden.

Ook is de werkruimte van Matlab, mede afhankelijk van de opslag van het apparaat van de gebruiker, heel klein. Dit zorgt er momenteel voor dat grote video's niet ingeladen kunnen worden. De gebruiker dient eerst met andere software de video te bewerken.

Het laatste discussiepunt betreft de werking van de sliders. Zodra de synchronisatie compleet is, kan de gebruiker de video en data synchroon afspelen. Beide sliders bewegen hierbij mee. Het is echter niet meer mogelijk om de sliders handmatig te gebruiken zodra deze synchronisatie compleet is. Het frame voor frame bekijken van de video en data is in deze UI nog niet mogelijk.

3.3 Evaluatie Eisen & Wensen

Eisen meetsysteem	
De sensoren van het meetsysteem zijn piëzo-resistief of resistief.	✓
De sensoren hebben zijn toepasbaar in een omgevingstemperatuur van -20°C t/m + 50°C.	✓
Het meetsysteem moet een minimum meetbereik hebben van 16.3 N/ cm ²	✓
De sensoren dienen te worden geplaatst op de hiel, middenvoet en/of metatarsale van de voet.	✓
De voetzolen moeten toepasbaar zijn voor een voetlengte van maat 35 t/m 46.	✓
Het meetsysteem dient te data op een SD kaart op te kunnen slaan.	✓
Per voet dient er een sensor op de hiel en op metatarsale van de voet te worden geplaatst.	✓

Wensen meetsysteem	
De sensoren hebben een zo laag mogelijke lineairiteit.	😊
De sensoren hebben een zo laag mogelijke hysteresis .	😊

Eisen Interface	
De UI kan feedback geven over de centrale balans en de stuurdruk.	✗
De UI beschikt over een play button	✓
De UI beschikt over een pauze button	✓
De UI beschikt over een stop button	✓
De UI beschikt over slider waarmee de gebruiker frame voor frame door de video kan bewegen.	✓

Wensen Interface	
De organisatie van de UI is geordend.	😊
De UI heeft interne consistentie .	😊
De iconen en functies in de UI hebben een enkele en duidelijke betekenis.	😊
De UI beschikt over niet meer dan vijf hoofd kleuren.	😊
In de UI zijn de functie-kleur verhoudingen gelijk.	😊

Eisen Trainingshulpmiddel	
In het trainingshulpmiddel kunnen video en data synchroon worden afgespeeld.	✓
Het gehele trainingshulpmiddel valt binnen het budget van € 800,00.	✓
De parameters centrale balans en stuurdruk kunnen in het trainingshulpmiddel worden geanalyseerd.	✗

4



Discussie

Conclusie

Aanbevelingen

4 D-A-C

Discussie

Gedurende het onderzoek zijn er een aantal discussiepunten aan het licht gekomen. In hoofdstuk 3 zijn de discussiepunten van het meetsysteem en de UI toegelicht. Er zijn echter ook een aantal algemene discussiepunten die dienen te worden benoemd.

De vooraf opgestelde planning was slechts de eerste paar weken een goede hou vast voor het onderzoek. Al snel werd duidelijk de ontwerpfase veel meer tijd innam dan vooraf gepland. Zoals eerder benoemd heeft dit er andere voor gezorgd dat er geen meting op zowel de rollerband als in de sneeuw heeft plaatsgevonden.

De gebruiker bepaalt in de UI het synchronisatiemoment in zowel de video als in de data. Het vaststellen van dit synchronisatiemoment kan, voornamelijk in de video, bijna niet 100% nauwkeurig worden gedaan. Dit omdat iedere sample, door de meetfrequentie van 120Hz, 0,0083 seconden lang is en een verschil slecht zichtbaar is in de video. De kans dat de gebruiker een fout maakt is zeer waarschijnlijk.

Aanbevelingen

In dit hoofdstuk worden de aanbevelingen naar aanleiding van dit onderzoek toegelicht die in hoofdstuk 3 nog niet zijn benoemd. Er wordt hierbij wordt onderscheid gemaakt tussen de twee hoofdonderwerpen; het meetsysteem en de User Interface.

Meetsysteem

Het aanpassen van het aantal uit te lezen kanalen van de Arduino is de eerste aanbeveling ter verbetering van het meetsysteem. De microcontroller van het huidige meetsysteem geeft de mogelijkheid om met vier sensoren te meten. Wanneer er meer sensoren gebruikt kunnen worden, kan er een beter beeld worden geschetst over de drukverdeling van de gehele voet.

In dit onderzoek is er voor gekozen om de data met de SD kaart op te slaan. Het kan in de toekomst van toegevoegde waarde zijn als het meetsysteem ook over een draadloze functie beschikt. De coach kan dan via communicatie, zoals oortjes, direct feedback geven op de skiër tijdens zijn of haar afdaling.

Het doel van dit onderzoek is om een trainingshulpmiddel van skiër te ontwikkelen waarin video en data vanuit voet druksensoren geanalyseerd kunnen worden. Het trainingshulpmiddel wordt in dit onderzoek als hulpmiddel voor bewegingsanalyse toegepast. In de toekomst kan deze ook voor andere doeleinden worden gebruikt. Skiërs op hoog niveau testen voorafgaand aan ieder seizoen, ontzettend veel paar schoenen en ski's. Alle combinaties van schoenen en ski's worden door de atleet getest en getoetst op gevoel. Hier worden geen vergelijkbare meetsystemen voor gebruikt. Zodra het meetsysteem uit dit onderzoek goed is doorontwikkeld, kan deze ook hier een bijdrage aan leveren. Het gevoel van de skiër kan hierbij worden ondersteund met getallen.

Ook op het gebied van materiaal kan het meetsysteem worden toegepast. Zo kunnen verenigingen al met jonge atleten metingen uitvoeren om het juiste materiaal te kiezen. Dit is momenteel bij skiërs van 12 t/m 18 jaar niet altijd even makkelijk. Verschillende stijfheden van schoenen en lengtes van ski's worden op dit moment op gevoel gebruikt, zo wel niet onnodig nadelig gebruikt. Zodra hier getallen aan worden gegeven kan dit de ontwikkeling van de jonge skiërs verbeteren.



Het meetsysteem is ook toepasbaar in wintersport speciaal zaken waar iedere jaar toeristen komen om nieuw materiaal te kopen. Voor de toerist kan het lastig zijn om het juiste materiaal te kiezen, omdat deze gebruiker het gevoel niet goed kan inbeelden en verwoorden. Het zou een toegevoegde waarde zijn als het meetsysteem gemakkelijk te verwisselen is in de schoenen, zodat een korte en snelle meting kan worden gedaan welke helpt de gebruiker een goede keuze te maken. Wanneer het meetsysteem gemakkelijk te verwisselen is in schoenen kan een snelle korte meting van toegevoegde waarde zijn voor de keuze van de koper.

Interface

De eerste aanbeveling voor de UI is de weergave van de grafieken. Momenteel beschikt de UI over een enkele soort grafiek die op diverse manieren kan worden toegepast. Deze grafieken blijken voor de gebruiker niet altijd gemakkelijk uit te lezen. Andere weergave opties zoals een afbeelding van een voet die met gebruik van kleuren de drukverdeling aangeeft. Een andere weergave mogelijkheid is het zien van een zogenoemd spoor. Dit spoor komt overeen met het spoor die achterblijft wanneer een skiër als eerst door een nieuw laagje sneeuw skiet. Parameters, zoals timing, kunnen hier gemakkelijk mee worden uitgelezen.

De tweede aanbeveling voor de UI is de locatie van de video en de datagrafieken. In de ontwikkelde UI worden de video en de datagrafieken horizontaal naast elkaar weergegeven. Met gebruik van een 'Setting' functie kan de gebruiker deze weergave ook boven elkaar instellen.

Als laatste is er nog een aanbeveling voor de gebruikershandleiding. Om de gebruiker te helpen bij het uitlezen van de grafieken, dienen er in de handleiding een aantal voorbeelden te worden weergegeven. Aan de hand van deze voorbeelden kan de gebruiker het uitlezen van de data wellicht gemakkelijker interpreteren.

Conclusie

De hoofdvraag van dit onderzoek luidt: *Hoe ziet een trainingshulpmiddel voor skiërs eruit, waarin video en data vanuit druksensoren geanalyseerd kunnen worden?*

Naar aanleiding van de evaluatie kan er worden geconcludeerd dat de hoofdvraag volledig is beantwoord. Met het ontwikkelde trainingshulpmiddel kan video en data vanuit druksensoren geanalyseerd worden.

Het doel van het onderzoek was de ontwikkeling van een trainingshulpmiddel voor coaches en atleten ter bevordering van de prestaties in de skisport met gebruik van druksensoren onder de voet. Met dit trainingshulpmiddel kan de CB en de stuurdruk worden geanalyseerd. Om dit hoofddoel te bereiken is dit doel onderverdeeld in twee subdoelen.

- 1) De ontwikkeling van een meetsysteem waarmee de druk onder de voet gemeten kan worden;
- 2) De ontwikkeling van een interface waarmee videobeelden en data uit het meetsysteem geanalyseerd kunnen worden.

Bovengenoemde doelstelling is niet volledig behaald. Zowel de centrale balans en de stuurdruk kunnen niet door de gebruiker worden geïnterpreteerd en geanalyseerd, omdat de sensor onder de voorvoet geen bruikbaar signaal afgeeft. Wel functioneert de User Interface zoals de opdrachtgever wenst. De video van de skiër en de data vanuit de voetzolen kunnen hierin synchroon worden geanalyseerd.





5

Bibliografie

5 Bibliografie

1. Michahelles F, Schiele B. Sensing and monitoring professional skiers. IEEE CS. 2005.
2. Nederlandse-Ski-Vereniging. Opleiding Skileraar A. In NskiV.. Den Haag; 2016.
3. Stricker G, Scheiber P, Lindenhofer E, Müller E. Determination of forces in alpine skiing and snowboarding: Validation of a mobile data acquisition system. 2010;(10): p. 31-41.
4. Koop J. Insole pressure measurement in speed skating. Bachelor Thesis. Den Haag: De Haagse Hogeschool, Mens en Techniek - Bewegingstechnologie; 2014.
5. Innosport-S'Hertogenbosch. innosportlabs. [Online].; 2013 [cited 2016 12 17. Available from: <http://www.innosportlabs-hertogenbosch.nl/projecten/turntrainerscockpit>.
6. Hirosea K, Dokia H, Kondoa A. Dynamic analysis and motion measurement of ski turns using inertial and force sensors. Procedia Engineering. 2013; 60: p. 355 – 360.
7. Kondo A, Doki H, Hirose K. An attempt of a new motion measurement method for alpine ski turns using inertial sensors. 2012;(34): p. 421-426.
8. Yoneyama T, Kagawa H, Unemoto M, Lizuka T, Scott NW. A Ski robot system for qualitative modeling of the carved turn. 2009;(11): p. 131-141.
9. Federolf P, Roos M, Luthi A, Dual J. Finite element simulation of the ski-snow interaction of an alpine ski in a carved turn. 2010;(12): p. 123-133.
10. Heinrich D, Mössner M, Kaps P, Nachbauer W. Calculation of the contact pressure between ski and snow during a carved turn in Alpine skiing. 2009 Maart 18; 2010(20): p. 485-492.
11. Yoneyama T, Kitabe M, Osada K. Investigation on the ski-snow interaction in a carved turn based on the actual measurement. 2010;(2): p. 2901-2906.
12. Yoneyama T, Kagawa H, Osada K. Measurement of ski snow-pressure profiles. 2007;(10): p. 145-156.
13. Brodie M,WA, Page W. Fusion motion capture: a prototype system using inertial measurement units and GPS for the biomechanical analysis of ski racing. 2008;(1-1): p. 17-28.
14. Ullly S. Plantar pressure-measurement sensors. Institute of Physics. 1991;(10): p. 16-32.
15. Rosenbaum D, Becker HP. Plantar pressure distribution measurements. Technical background and clinical applications. Foot and Ankle Surgery. 1997;(3): p. 1-14.
16. Razak AHA, Zayegh A, Begg RK, Wahab Y. Foot plantar pressure measurement System: a review. Sensors. 2012 Juli;(12).
17. Schrauwen M. Meten; sensoren en signaalconditionering. In. Den Haag; 2013.
18. Nakazato K, Scheiber P, Müller E. A comparison of ground reaction forces determined by portable force-plate and pressure-insole systems in alpine skiing. Journal of Sports Science and Medicine. 2011;(10): p. 754-762.
19. Holleczeck T, Rüegg A, Harms H, Tröster G. Textile pressure sensors for sport applications. IEEE Sensors. 2010 November;; p. 732-737.
20. Shu L, Hua T, Wang Y, Li Q, Feng DD. In-shoe plantar pressure measurement and analysis system based on fabric pressure sensing array. EEE Engineering in Medicine and Biology Society. 2010 Maart.



21. Delft T. Dined. [Online]. [cited 2017 Januari. Available from: <http://dined.io.tudelft.nl/en/database/tool>.
22. Usability. User Interface Design Basics. [Online]. Available from: <https://www.usability.gov/what-and-why/user-interface-design.html>.
23. Fadeyev D. Using Light, Color and Contrast Effectively in UI Design. [Online].; 2008 [cited 2017 Februari. Available from: <http://www.usabilitypost.com/2008/08/14/using-light-color-and-contrast-effectively-in-ui-design/>.
24. Trenchard P. 11 Principles of Interaction Design explained. [Online].; 2010 [cited 2017 februari. Available from: <https://shortboresurfer.com/2010/08/13/11-principles-of-interaction-design-explained/>.
25. Martin S. Effective Visual Communication for Graphical User Interfaces. [Online]. [cited 2017 februari. Available from: http://web.cs.wpi.edu/~matt/courses/cs563/talks/smartin/int_design.html.
26. Mathworks. Mathworks GUI. [Online].; 2017 [cited 2017 april. Available from: <https://nl.mathworks.com/discovery/matlab-gui.html>.
27. Mathworks. Mathworks - Smoothdata. [Online]. [cited 2017 mei. Available from: <https://nl.mathworks.com/help/matlab/ref/smoothdata.html>.
28. Pinterest. [Online]. [cited 2017 Januari. Available from: www.pinterest.com.
29. Schreiber P, Schwameder H, Müller E. Characteristics of the force application point - a method to identify learning processes in alpine skiing. 2006; 2: p. 564-567.
30. Müller E, Ebert C, Senner V. Approach for a realtime training feedback algorithm in skiing. 2007;; p. 47.
31. Nederlandse-Ski-Vereniging. Opleiding Skibegleiter. In NskiV.. Den Haag; 2016.
32. Othmar Hofmann. Ski specifiek bewegingsvaardigheden. [Online].; 2016 [cited 2016 11 21. Available from: http://www.skidiscovery.nl/index.php?id=13233&title=ski-specifieke_bewegingsvaardigheden.



6 Bijlagen

Gebruikershandleiding | Meetprotocol | Resultaten |
Arduino Script | Matlab script | Persoonlijke leerdoelen
Projectplan

6 Bijlagen

Bijlage I – Gebruikershandleiding

Welkom bij Ski Feet Analyzer!

Tabel 1 Overzicht van benodigdheden.

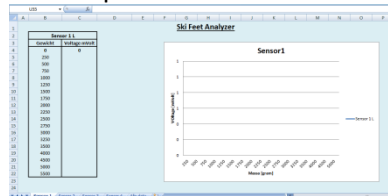
Benodigdheden	Opmerking
FlexiForce A201 sensoren	Aantal: 4x
Meetkastje	Inhoud: Arduino + SD Shield Breadboard met circuit 9Volt blok batterij + stekker 4x 2draads kabels SD kaart
Voetzolen	3mm dik en op maat knipbaar
Laptop	Benodigde software: Arduino IDE <ul style="list-style-type: none">- Script seriële monitor- Script SD kaart Matlab <ul style="list-style-type: none">- SkiFeetAnalyzer m.file- SkiFeetAnalyzer fig.file Excel Calibratie SkiFeetAnalyzer
Camera of Smartphone	Moet met frequentie van 120Hz kunnen filmen
Isolatietape	
Schilderstape	

Calibratie

Voor het analyseren van de data dient het meetsysteem eerst gecalibreerd te worden. Calibratie is een methode om de relatie tussen de output van de sensor en de kracht die op de sensor werkt te bepalen. Voor dit proces worden er meerdere bekende gewichten een voor een op de sensor geplaatst. Een voorbeeld van deze bekende gewichten is te zien in Figuur 1 Om te zorgen dat de gehele massa van deze bekende gewicht gemeten wordt door de sensor dient er gebruik worden gemaakt van hiervoor een speciaal ontwikkeld instrument waarmee de sensor exact op het meetgebied wordt belast. Dit instrument, ook wel de A201 toren genoemd, wordt weergegeven in Figuur 2.

Voor de calibratie dient er gebruik te worden gemaakt van het daarvoor opgestelde Excel-document 'Calibratie SkiFeetAnalyzer'. Iedere sensor heeft zijn eigen tabblad.

1. Plaats de sensor zorgvuldig in de A201 toren;
 - a. Wanneer de sensor goed is geplaatst rust het blokje precies op het sensorisch gebied van de sensor
2. Open het Calibratie Excel bestand en open de sheet van de sensor;



De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.

- Verbind de Arduino met de computer en laad via IDE het seriële monitor script in;

```
SkiFeetAnalyzer-SeriëleMonitor | Arduino 1.8.3 (Windows Store 1.8.6.0)
Bestand Bewerken Schets Hulpmiddelen Help

SkiFeetAnalyzer-SeriëleMonitor

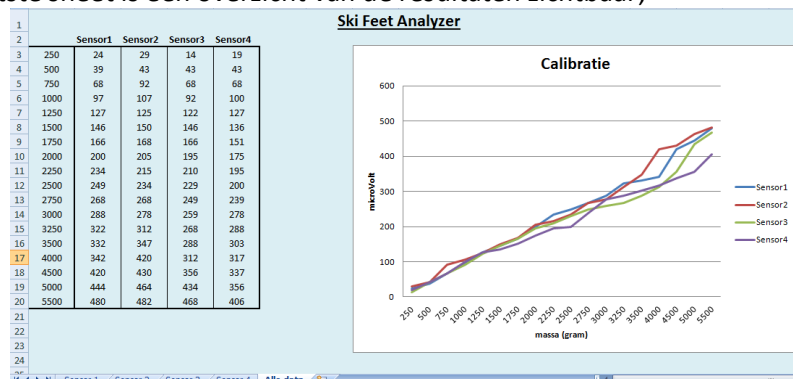
int FSR1 = A0;
int FSR2 = A1;
int FSR3 = A2;
int FSR4 = A3;

int voltage;
```

- Plaats het aangegeven gewicht op de sensor en lees in de seriële monitor het micro voltage af;
- Verwijder het gewicht, wacht 2 seconden en plaats opnieuw het gewicht op de A201 toren;
- Wanneer deze waarde overeen komt met de eerst keer dat de sensor de massa ontvangt, vul deze waarde in, in het Excel-bestand;

Sensor 1 L	
Gewicht	Voltage mVolt
0	0
250	
500	
750	
1000	
1250	
1500	
1750	
2000	
2250	
2500	
2750	
3000	
3250	
3500	
4000	
4500	
5000	
5500	

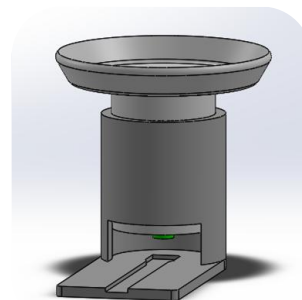
- Herhaal bovengenoemde stappen voor gewichten en voor alle sensoren;
- In de laatste sheet is een overzicht van de resultaten zichtbaar;



- Sla het document op;



Figuur 1 Voorbeeld van calibratie gewichten.



Figuur 2 Calibratie hulpmiddel waarmee de FlexiForce A201 precies kan worden belast op het meetgebied van de sensor.

Installeren meetsysteem

1. Knip de voetzolen in de geschikte schoenmaat;
2. Plaats de sensoren op de onderzijde van de voetzolen, zie Figuur 3 en Tabel 2;
 - a. Het is hierbij van belang dat de tape niet op het meetgebied van de sensor wordt geplakt;
 - b. De voetzolen zijn op dit moment nog niet in de skischoen geplaatst.
3. Verbind de Arduino met pc en laad het script 'SkiFeetAnalyzer - SerieleMonitor' in;
4. Open de seriële monitor;
5. Controleer de signalen van de sensoren;
 - a. Herpositioneer, indien nodig, de sensoren.
6. Plaats de voetzolen voorzichtig in de skischoen;
Let op: de sensoren zijn uiterst kwetsbaar. Doe dit heel voorzichtig!
7. De gebruiker kan voorzichtig de skischoen aantrekken;
Let op: de inlegzool kan verschuiven en/of kabels kunnen verstrikt raken.
Maak indien nodig gebruik van een plastic zakje om de wrijving te verminderen en het aantrekken te versimpelen.
8. Controleer wederom het signaal;
9. Laad het script 'SkiFeetAnalyzer - SD' in en ontkoppel de Arduino van pc;
10. Verbind de batterij met de Arduino;
11. Plaats een SD kaart in de Arduino;
12. Dicht het kastje met elektronica en bevestig het kastje aan de broekriem van de gebruiker, zie Figuur 4;
 - a. De aan- en uitknop dient zich aan de bovenzijde te bevinden.
13. Het meetsysteem is nu klaar voor gebruik!



Figuur 3 Plaatsing van de sensoren

Sensor	Positie
Sensor 1	Linker voet - hiel
Sensor 2	Linker voet – metacarpaal I
Sensor 3	Rechter voet - hiel
Sensor 4	Rechter voet – metacarpaal I

Tabel 2 Positie van de sensoren onder de voet.



Figuur 4 Het meetsysteem waarmee de oefeningen worden uitgevoerd. Rechts op de heup is het kastje bevestigd. Aan de bovenzijde van het kastje bevindt zich de aan- en uitknop

Uitvoering meting

1. Positioneer de camera op de gewenste positie;
 - a. De camera dient, net als het meetsysteem, te filmen met een frequentie van 120 Hz.
2. Start de meting met de aan- en uitknop en de filmopname;
3. Til twee seconden na het starten van de meting de linker of rechter voet op welke duidelijk zichtbaar dient te zijn op de camera. Dit is het synchronisatie moment en belangrijk voor het analyseren van de video met data;
4. Start het skiën!

Data analyse

Voor het analyseren van de data dient gebruik worden gemaakt van Matlab. Volg onderstaand stappen plan:

Volg onderstaand stappen plan:

1. Open Matlab;



2. Type in de command window 'GUIDE';

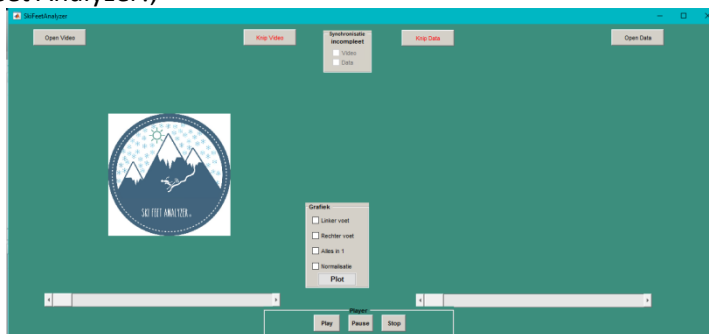
3. Open SkiFeetAnalyzer;

Let op: het wijzigen van instellingen kan zorgen dat het programma niet werkt! Wijzig hier dus niks aan!

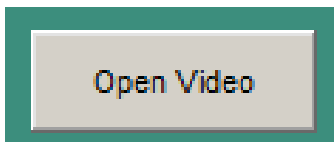
4. Klik bovenin op de groene pijl 'RUN';



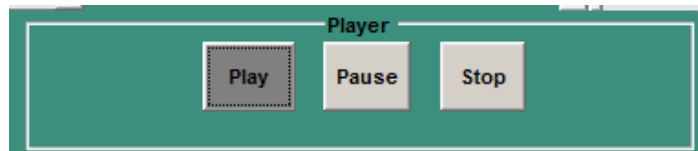
5. Welkom in Ski Feet Analyzer!;



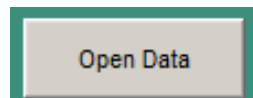
6. Upload de gewenste video;



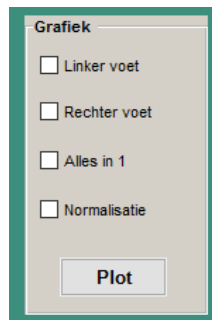
7. Speel de video af met gebruik van de play, pauze en stop button;
Indien de functie is ingeschakeld wijzigt de kleur in donker grijs.



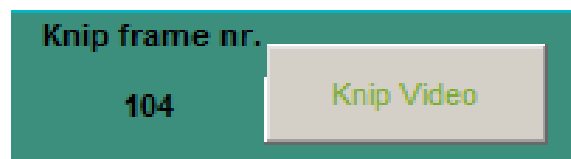
8. Upload de gewenste data;
Er worden om drie bestanden gevraagd; calibratie, normalisatie & de voetzooldata. Lees goed welk bestand er moet worden ingeladen.



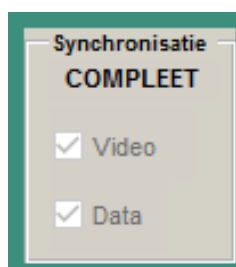
9. De weergave van de grafieken kan worden gewijzigd door de gewenste data aan te vinken en op 'PLOT' te klikken;



10. Bepaal het synchronisatie moment van de video en klik op 'Knip video';
De letters veranderen in groen. Het framenummer waarop u het knipmoment heeft bepaald verschijnt voor uw informatie nu in beeld. Tevens verschijnt er een vinkje in de synchronisatie checkbox.

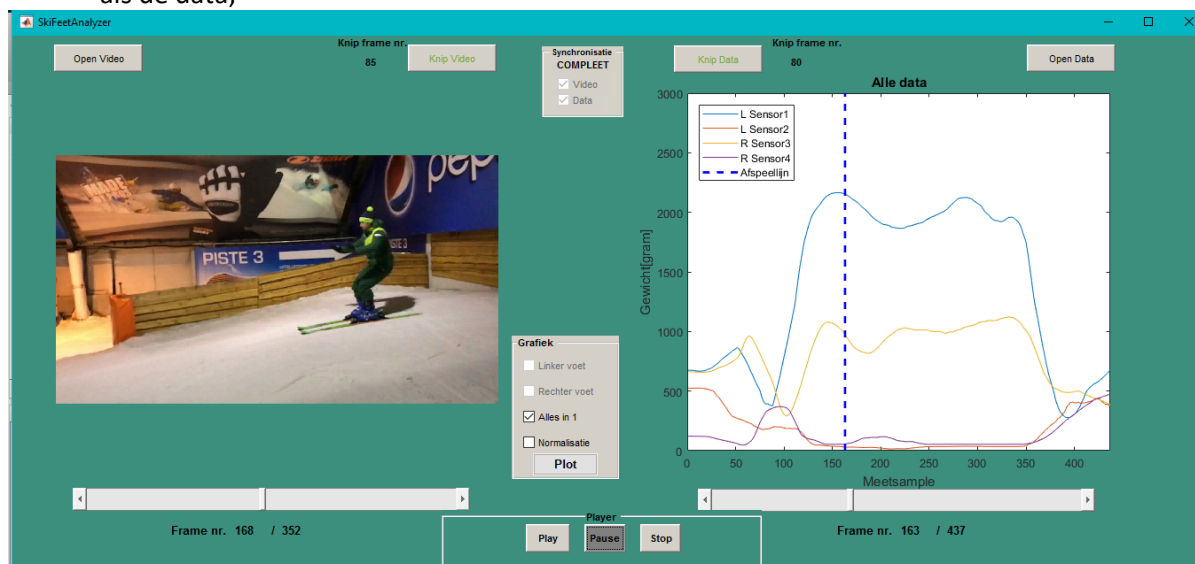


11. Bepaal het synchronisatie moment van de data en klik op 'Knip data';
De letters veranderen in groen. Het framenummer waarop u het knipmoment heeft bepaald verschijnt voor uw informatie nu in beeld. Tevens verschijnt er een vinkje in de synchronisatie checkbox. Zodra zowel de video als de data zijn geknipt komt er 'Synchronisatie COMPLEET' in uw beeld te staan.



De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.

12. Zodra de synchronisatie compleet is, bedienen de play- pauze- en stopbutton zowel de video als de data;



Bijlage II – Meetprotocol

De opdrachtgever voert vooraf bepaalde skioefeningen uit om de validiteit van het meetsysteem te kunnen bepalen. Dit vindt plaats in Snowworld Zoetermeer. De data wordt met behulp van Matlab geanalyseerd. Er is hiervoor een Matlab script opgesteld waarmee de data geanalyseerd kan worden. De parameters die met gebruik van dit meetprotocol getest worden zijn:

- Centrale balans: belasten van hele voetzool;
- Stuurdruk: de druk die de skiër ervaart.

Het stappenplan van dit meetprotocol is te vinden in Bijlage I – Gebruikershandleiding.

Voor deze validatietest is het gebruik van de UI 'SkiFeetAnalyzer' niet noodzakelijk.

Er gebruik worden gemaakt van de bestanden

'Uitlezendata.m' i.c.m. 'Normalisatie.m' en 'Calibratie.m'.

De volgende zeven oefeningen dienen door de proefpersoon een voor een te worden uitgevoerd:

- 1 Glijden in de vallijn* - centraal
- 2 Glijden in de vallijn - volledig achterin
- 3 Glijden in de vallijn - volledig voorin
- 4 Glijden in de vallijn - optillen linkerbeen
- 5 Glijden in de vallijn - optillen rechterbeen
- 6 Schools skiën - optillen bergbeen
- 7 Schools skiën
 - 7.1 drie bochten centraal
 - 7.2 drie bochten achterin
 - 7.3 drie bochten voorin
 - 7.4 drie bochten centraal

*glijden in de vallijn is in de skiwereld een veel gebruikte term. De vallijn is de rechte lijn, berg afwaarts.

In Tabel 1 worden de oefeningen met een afbeelding toegelicht.

Tabel 1 Weergave van de oefeningen die tijdens de meting worden uitgevoerd.

Glijden in de vallijn Centraal	Glijden in de vallijn Achterin	Glijden in de vallijn Voorin	Glijden in de vallijn Optillen linkerbeen
			
Glijden in de vallijn Optillen rechterbeen	Schools Optillen bergbeen	Schools a b c d	
			

Bijlage III – Resultaten Meetsysteem

De resultaten worden in volgorde van oefening weergegeven. Voor iedere oefeningen zijn er vijf grafieken weergegeven. In Tabel 3.1 wordt toegelicht wat er in de grafiek weergegeven wordt.

Tabel 3.1 Overzicht van weergave van de grafieken in de resultaten.

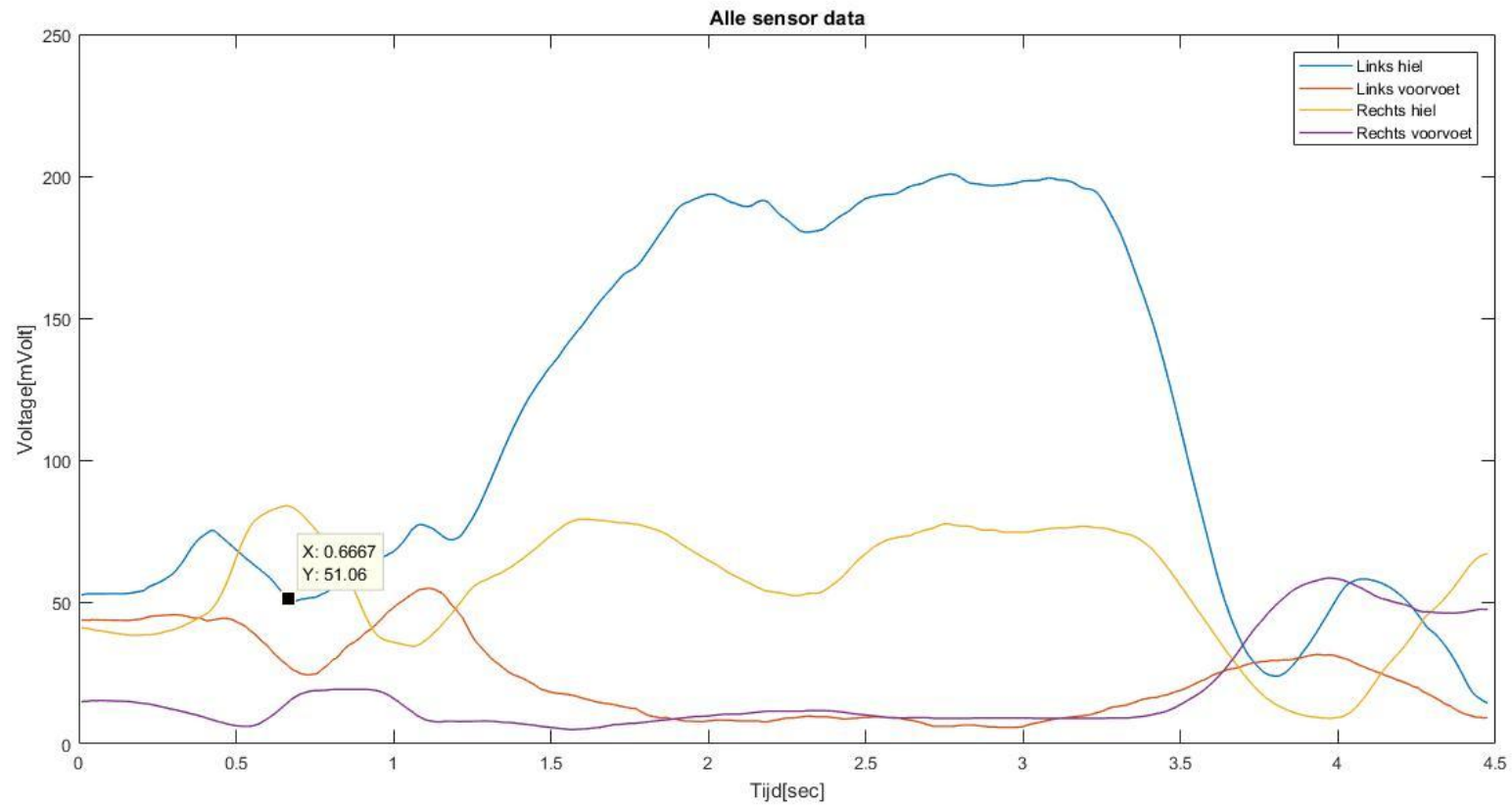
Grafiek	Wat wordt er weergegeven?	X-as	Y-as
1	Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.	Tijd[sec]	Voltage [mVolt]
2	Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.	Tijd[sec]	Voltage [mVolt]
3	Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd. Linker- en rechtersoet in een aparte plot.	Tijd[sec]	Voltage [mVolt]
4	Alle gecalibreerde data.	Tijd[sec]	Gewicht[gram]
5	Alle gecalibreerde data, genormaliseerd	Tijd[sec]	Gewicht[gram]

In Tabel 3.2 wordt het synchronisatie moment van iedere oefeningen weergegeven.

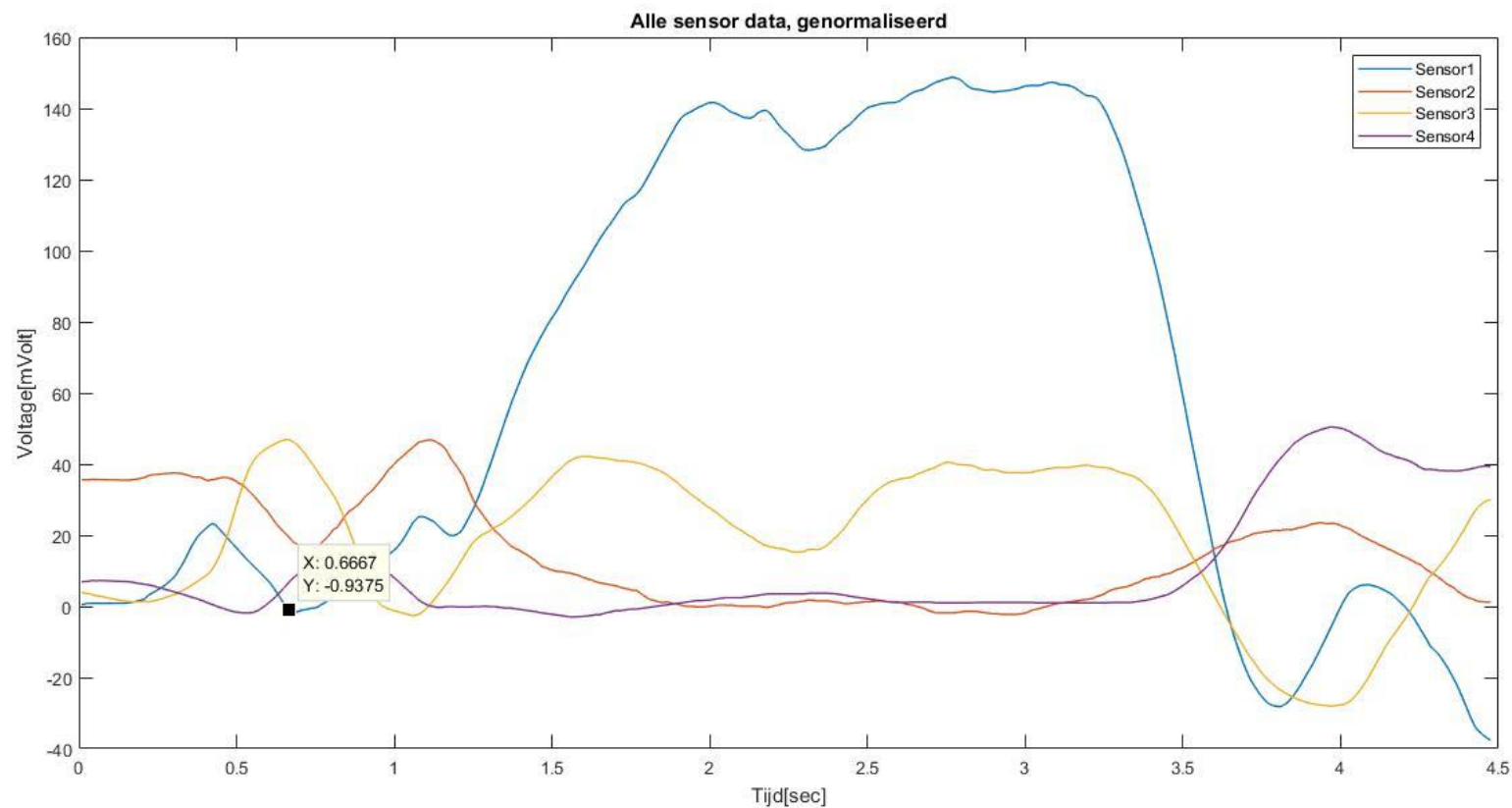
Tabel 3.2 Overzicht van synchronisatiepunten per oefening.

Oefening	Synchronisatiepunt x-as	Opmerking
1. Glijden in de vallijn, centraal	0.6667	Synchronisatie: optillen links
2. Glijden in de vallijn, achterin	0.65	Synchronisatie: optillen links
3. Glijden in de vallijn, voorin		Synchronisatie moment niet zichtbaar.
4. Glijden in de vallijn, optillen linkerbeen	0.775	Synchronisatie: optillen links
5. Glijden in de vallijn, optillen rechterbeen	0.825	Synchronisatie: optillen links
6. Schools skiën, optillen bergbeen	0.134	Synchronisatie: optillen rechts
7. Schools skiën CAVC	0.55	Synchronisatie: optillen rechts Overig: niet zichtbaar welke fase de skiër zich bevindt.

Oefening 1. Glijden in de vallijn, centraal

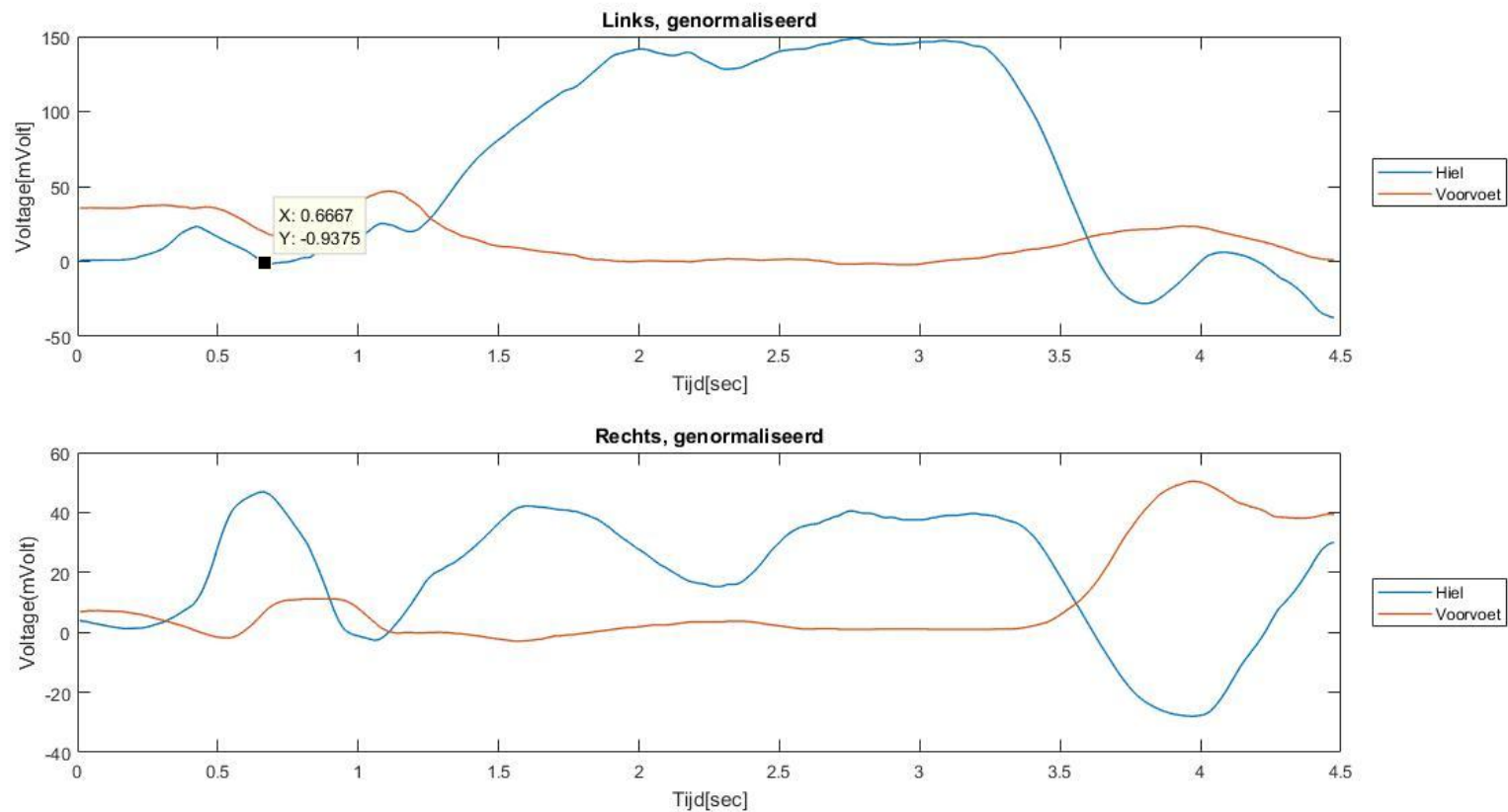


Figuur 1.1 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.
X-as Tijd[sec] Y-as Voltage [mVolt]

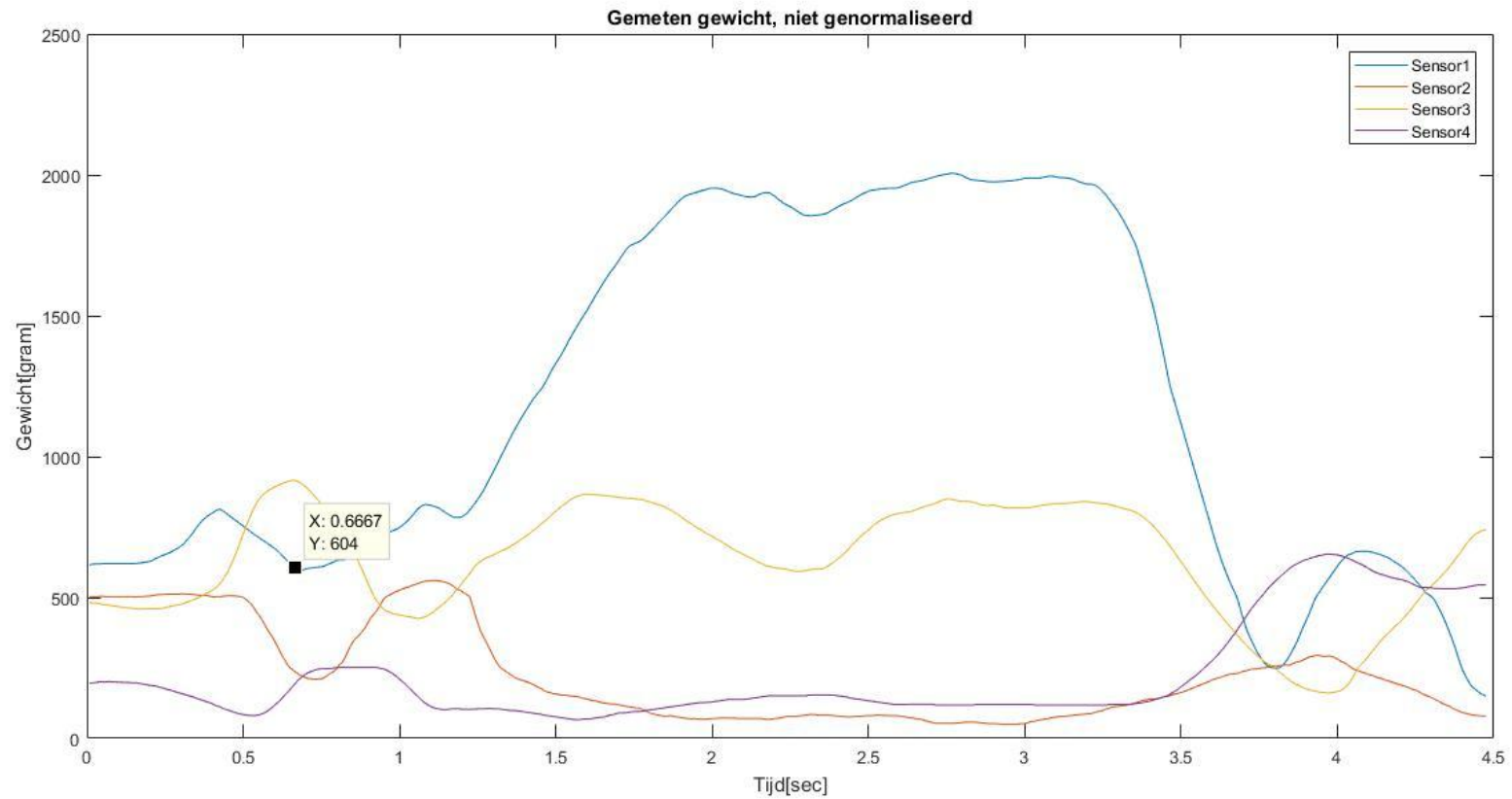


Figuur 1.2 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
X-as Tijd[sec] Y-as Voltage [mVolt]

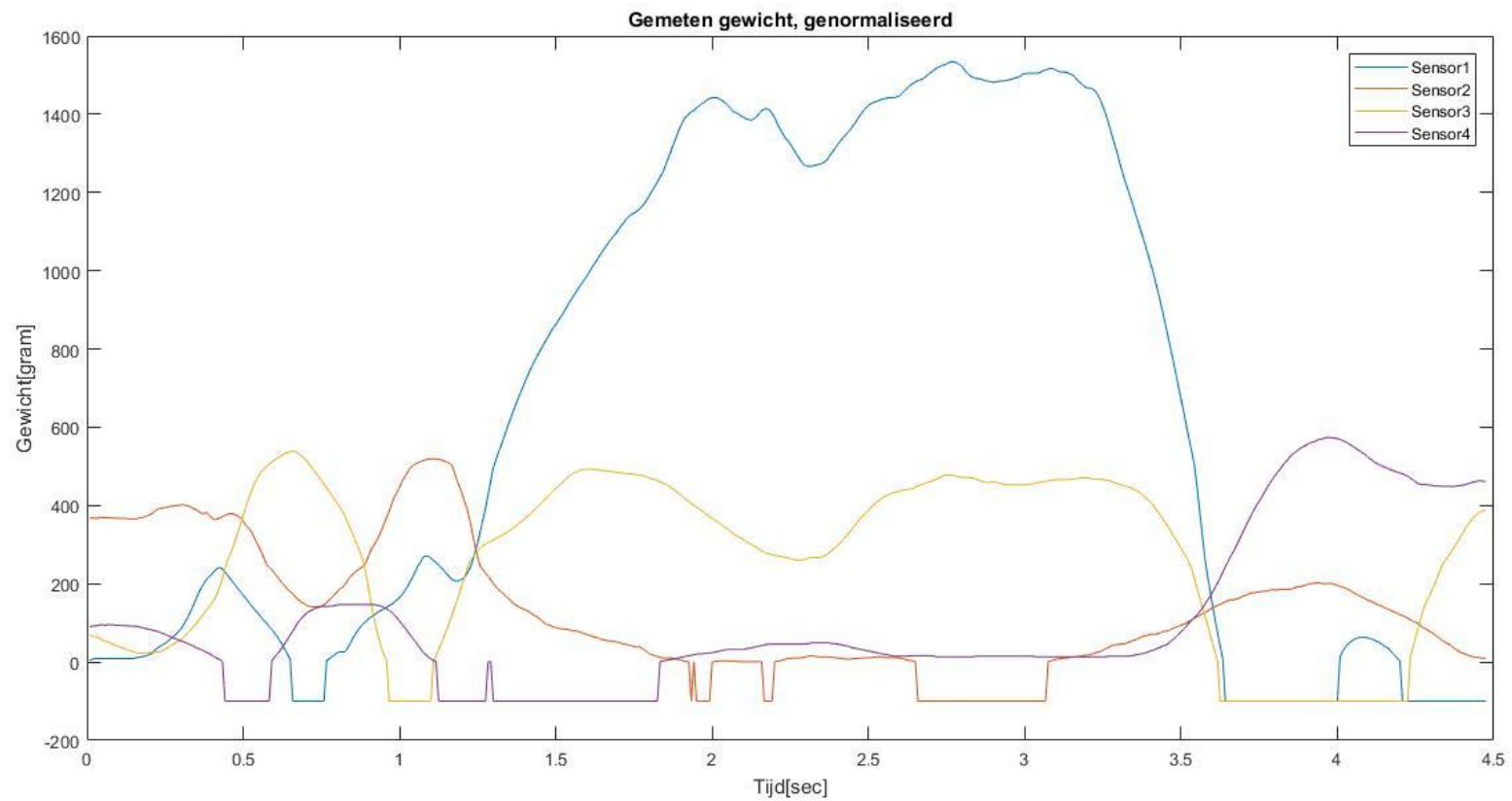
De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.



Figuur 1.3 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
Boven: Linkervoet. Onder: Rechtervoet X-as Tijd[sec] Y-as Voltage [mVolt]

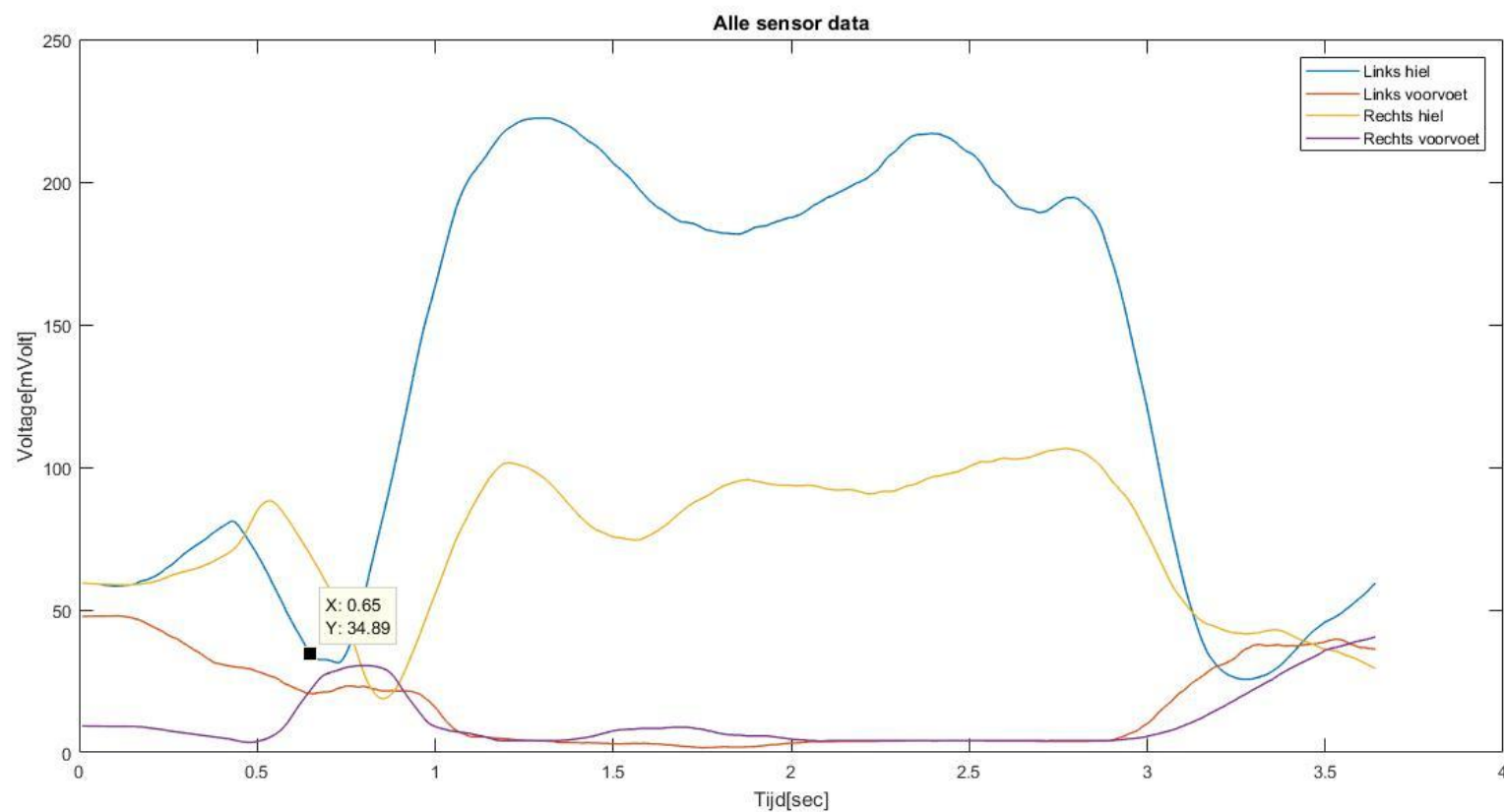


Figuur 1.4 Alle gecalibreerde data.
X-as Tijd[sec] Y-as Gewicht[gram]

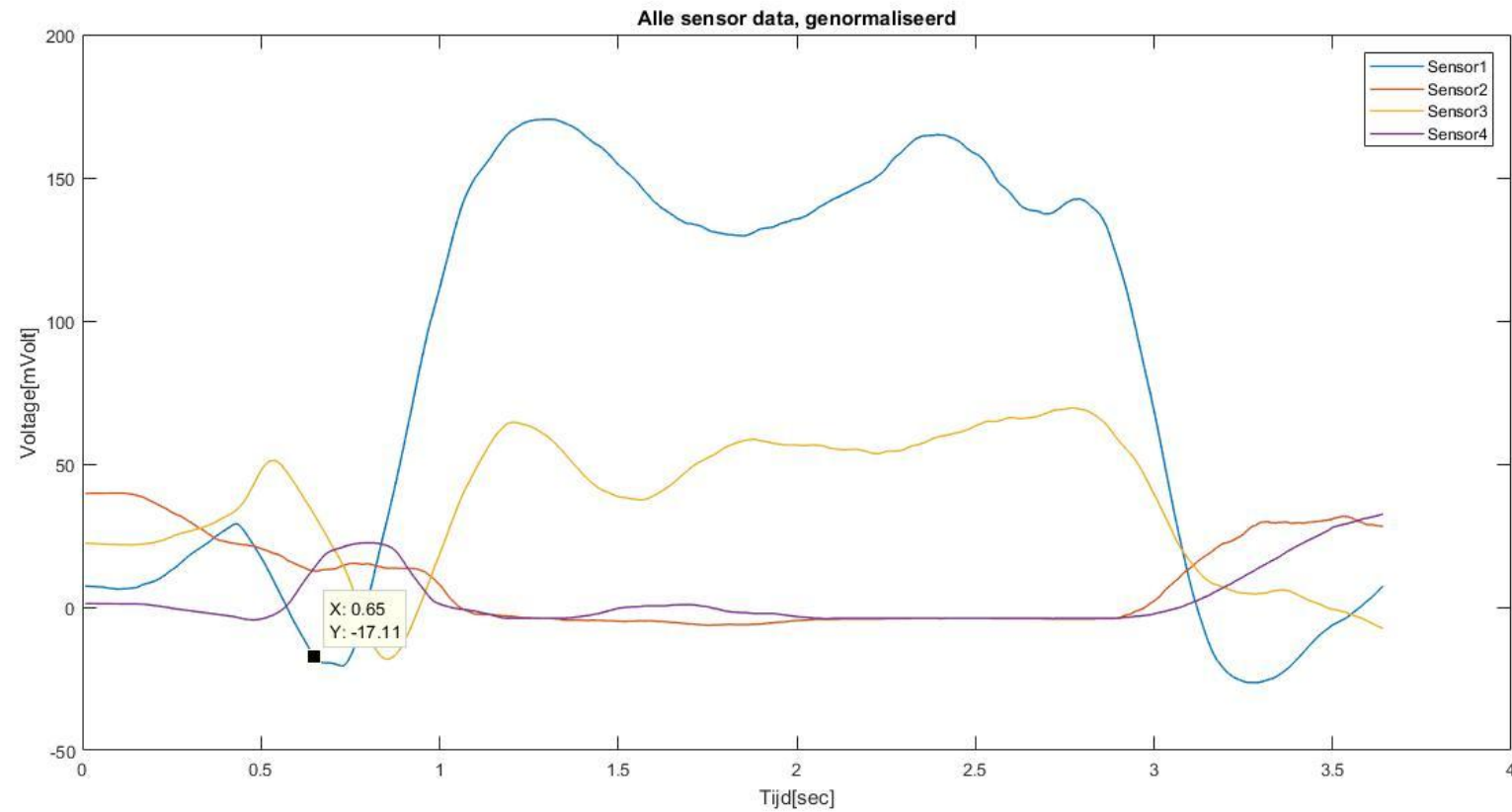


Figuur 1.5 Alle gecalibreerde data, genormaliseerd.
X-as Tijd[sec] Y-as Gewicht[gram]

Oefening 2. Glijden in de vallijn, achterin

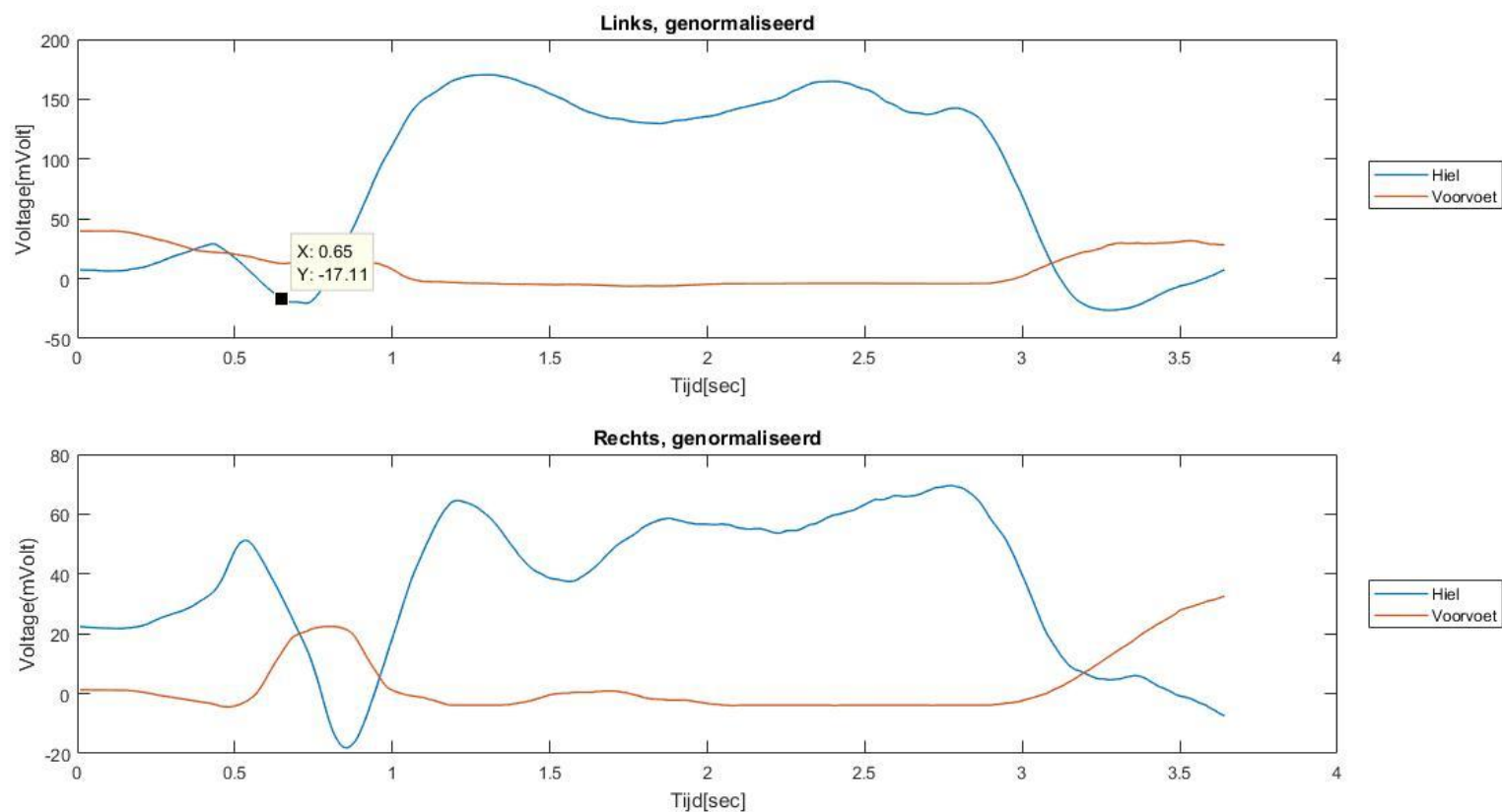


Figuur 2.1 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.
X-as Tijd[sec] Y-as Voltage [mVolt].

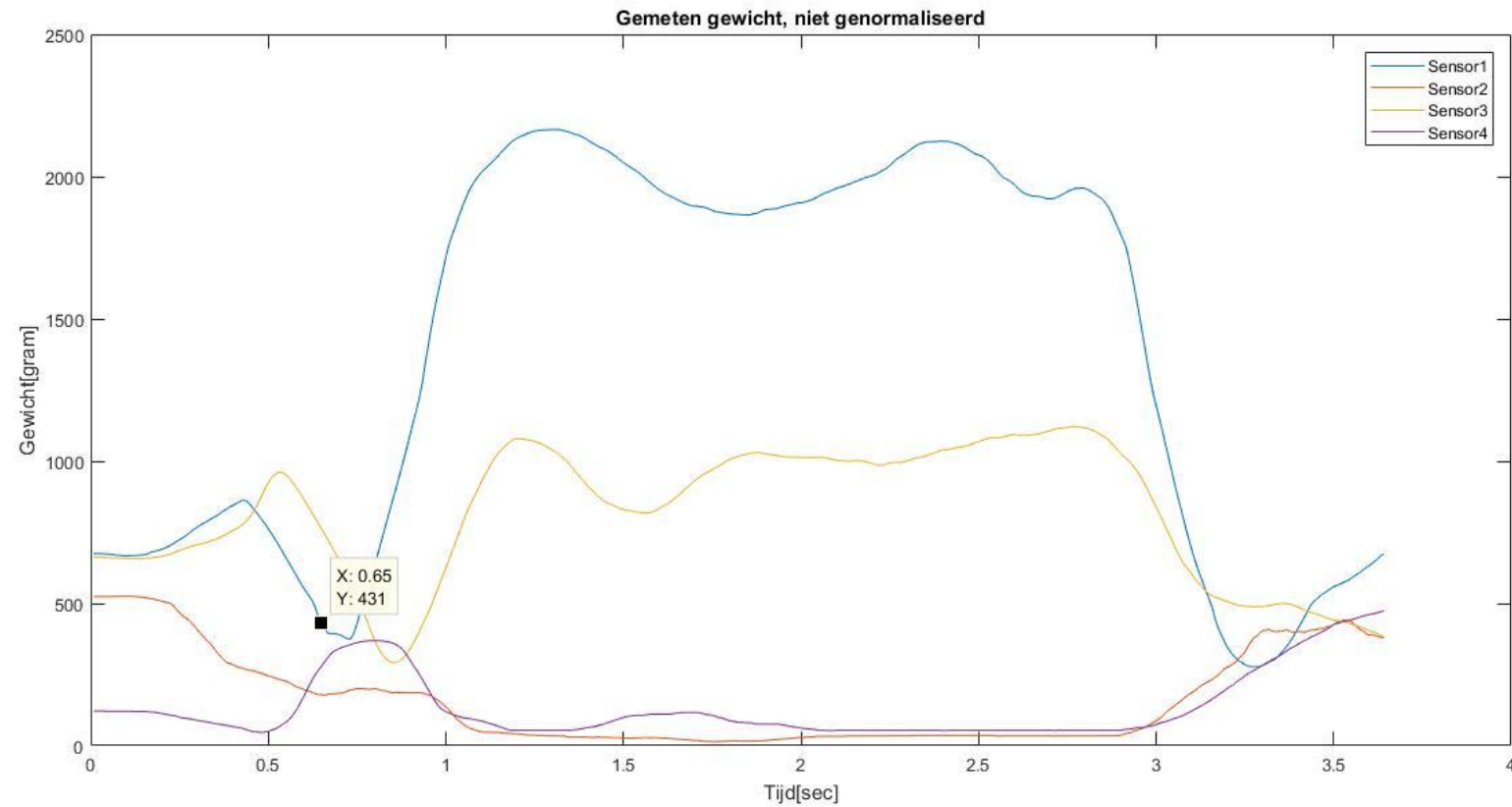


Figuur 2.2 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
X-as Tijd[sec] Y-as Voltage [mVolt]

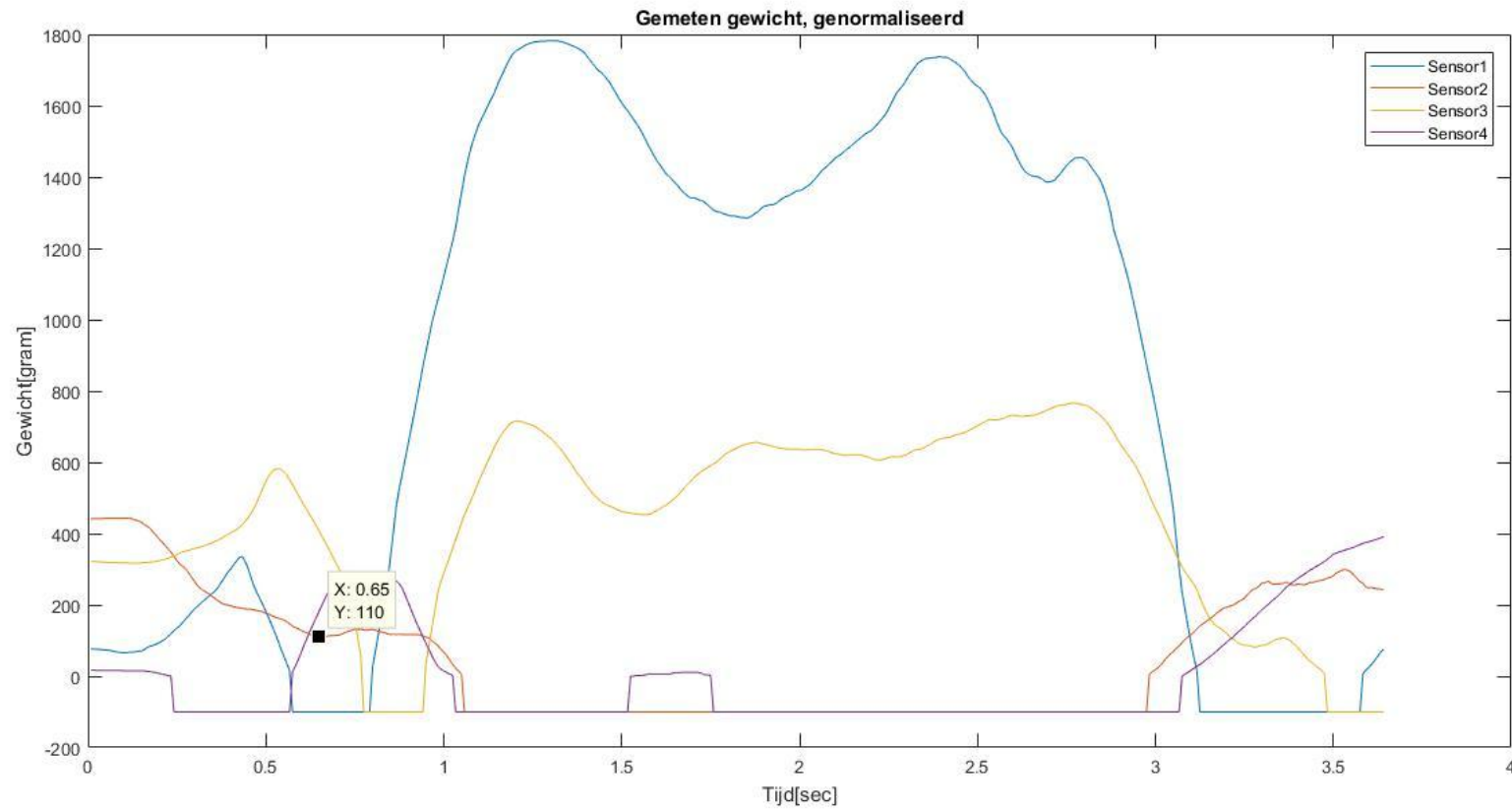
De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.



Figuur 2.3 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
Boven: Linkervoet. Onder: Rechtervoet X-as Tijd[sec] Y-as Voltage [mVolt]

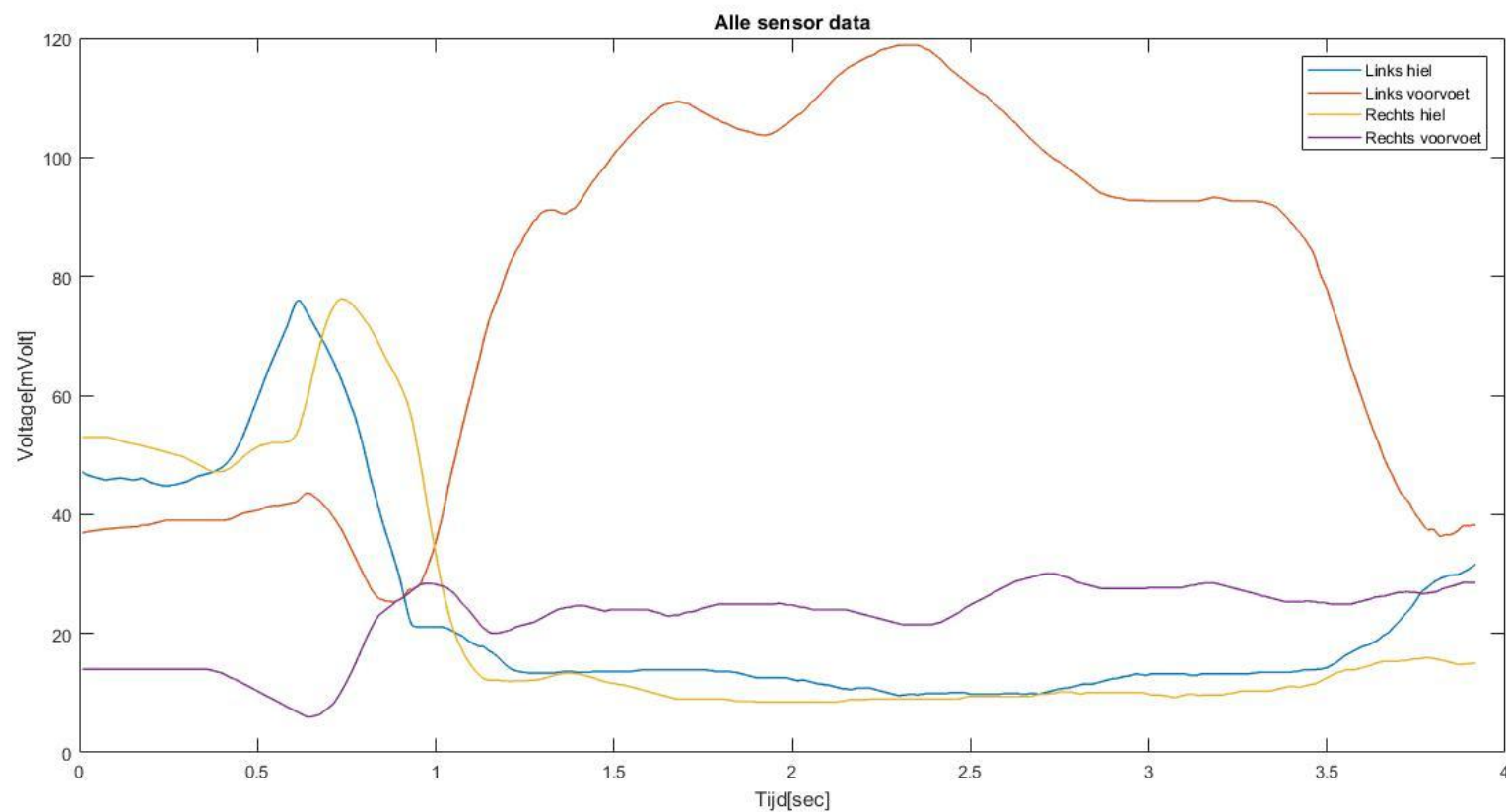


Figuur 2.4 Alle gecalibreerde data.
X-as Tijd[sec] Y-as Gewicht[gram]

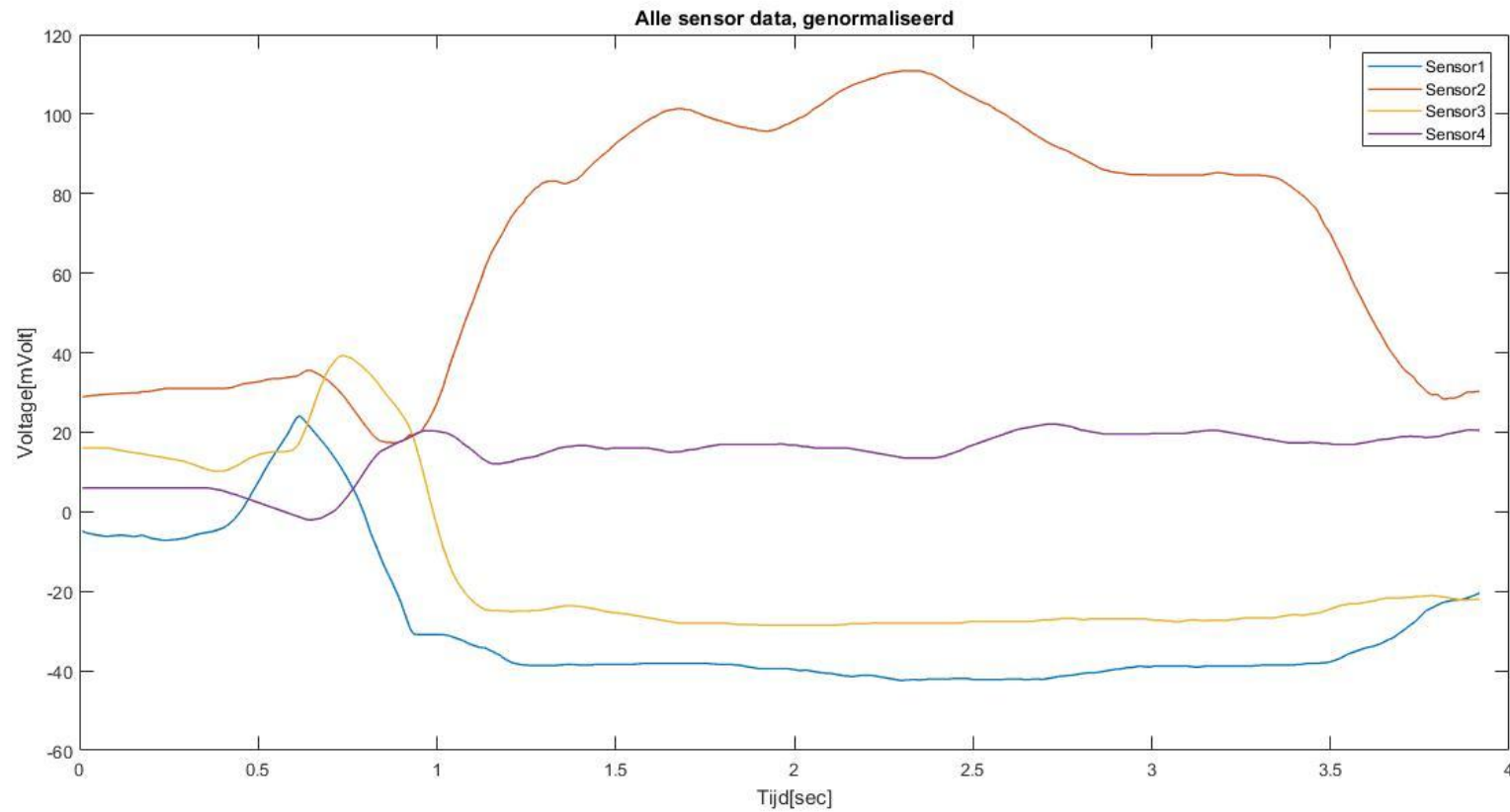


Figuur 2.5 Alle gecalibreerde data, genormaliseerd.
X-as Tijd[sec] Y-as Gewicht[gram]

Oefening 3. Glijden voorin

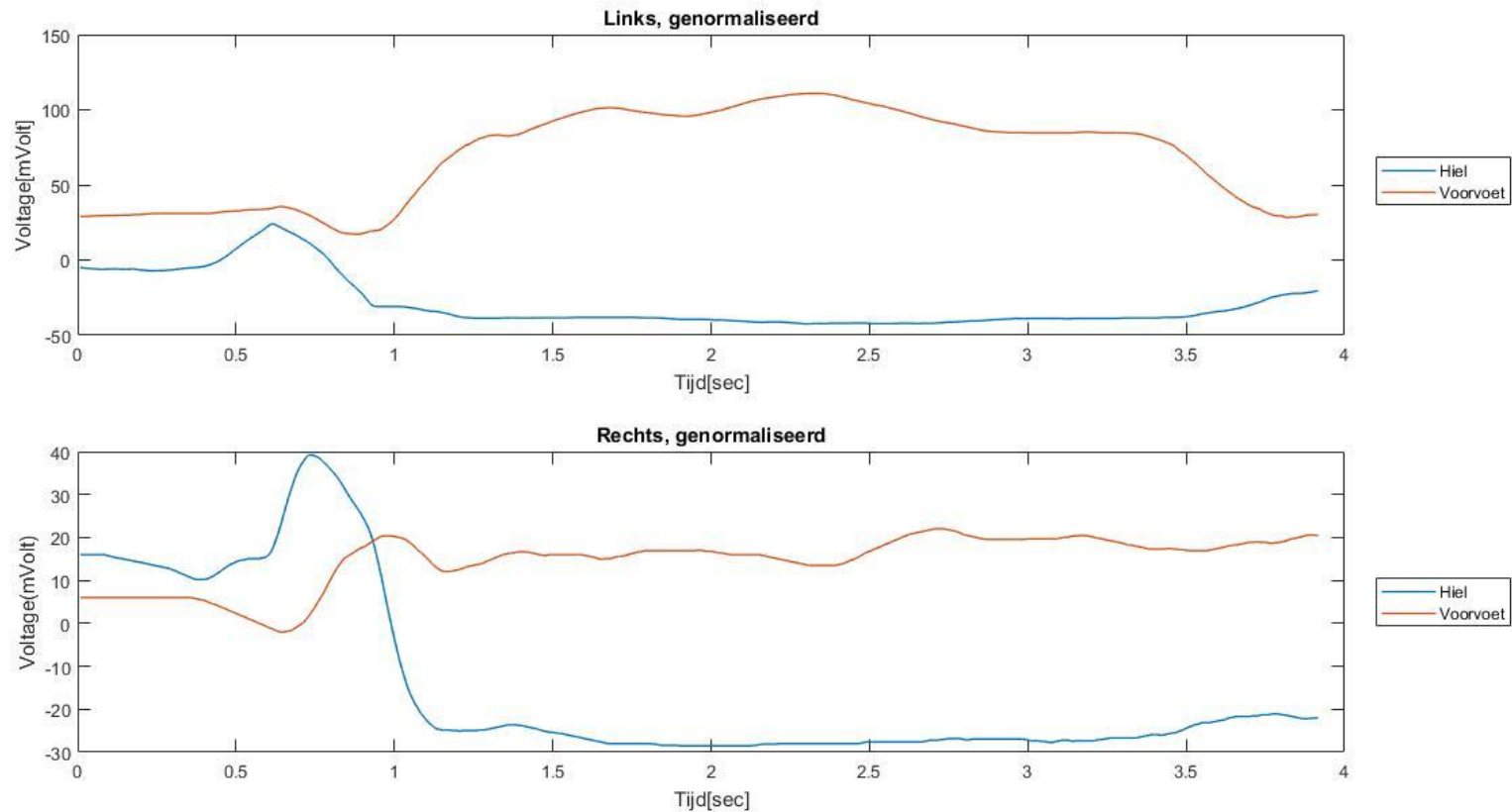


Figuur 3.1 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.
X-as Tijd[sec] Y-as Voltage [mVolt]

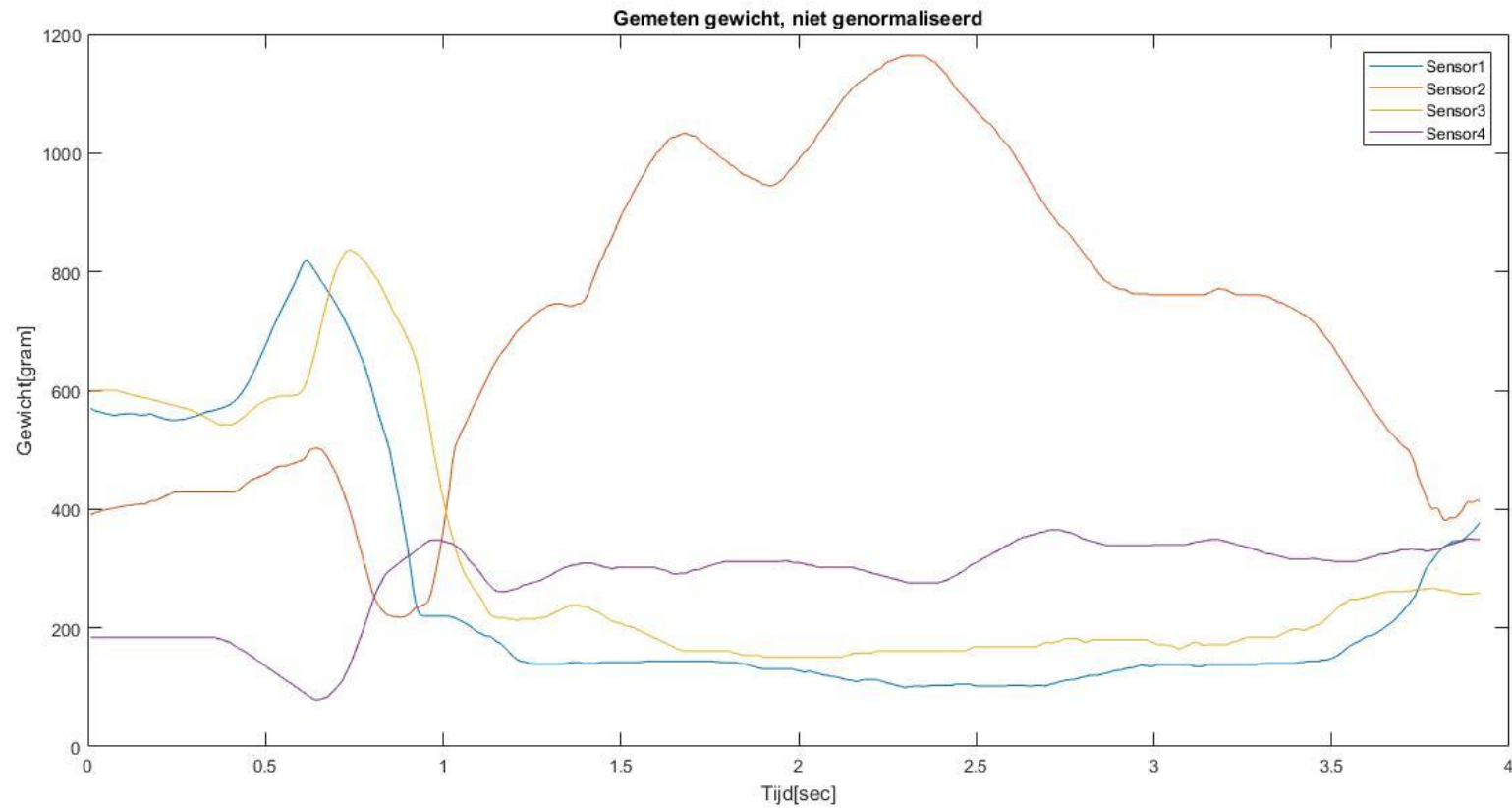


Figuur 3.2 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
X-as Tijd[sec] Y-as Voltage [mVolt]

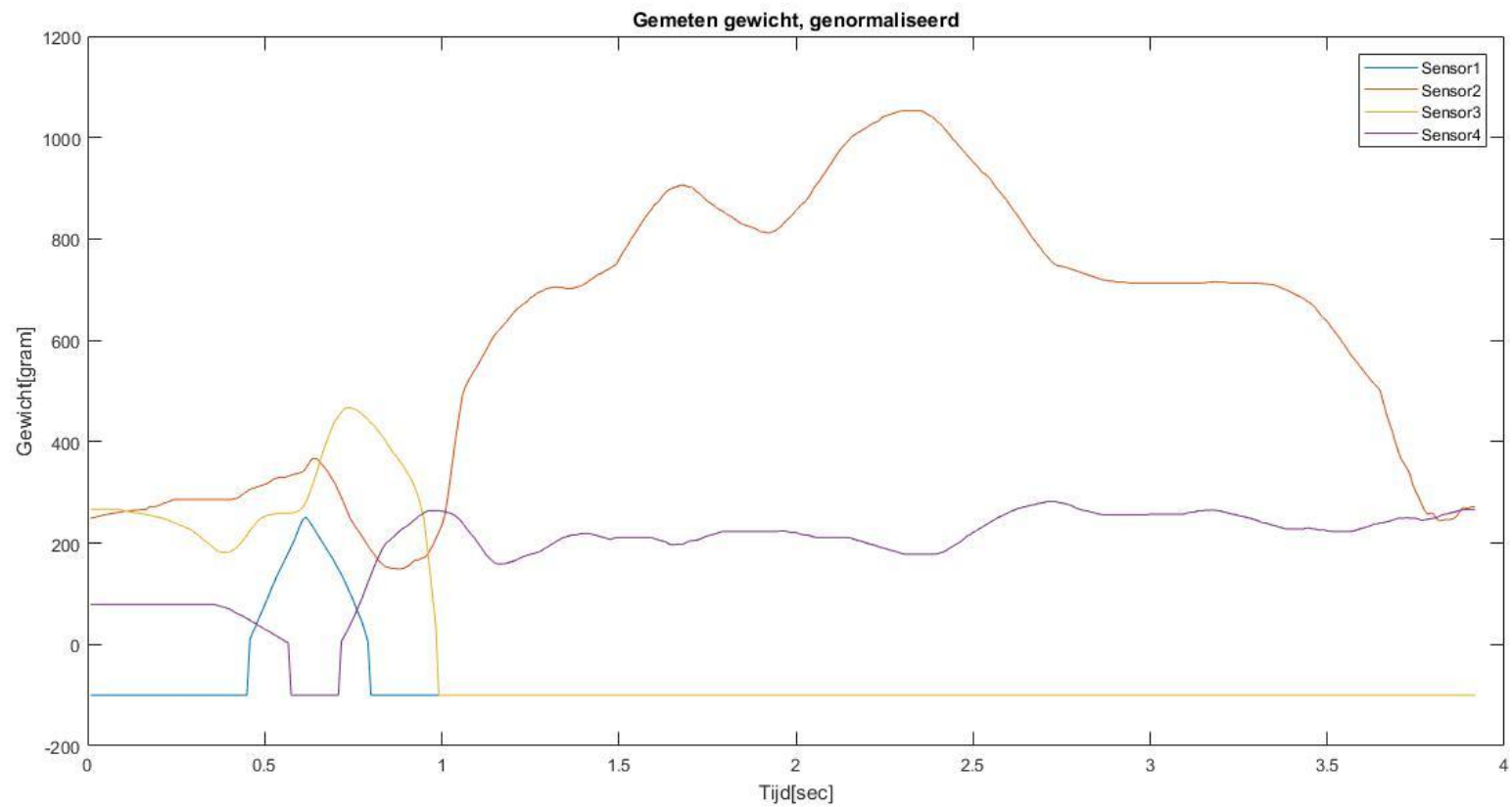
De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.



Figuur 3.3 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
Boven: Linkervoet. Onder: Rechtervoet X-as Tijd[sec] Y-as Voltage [mVolt]

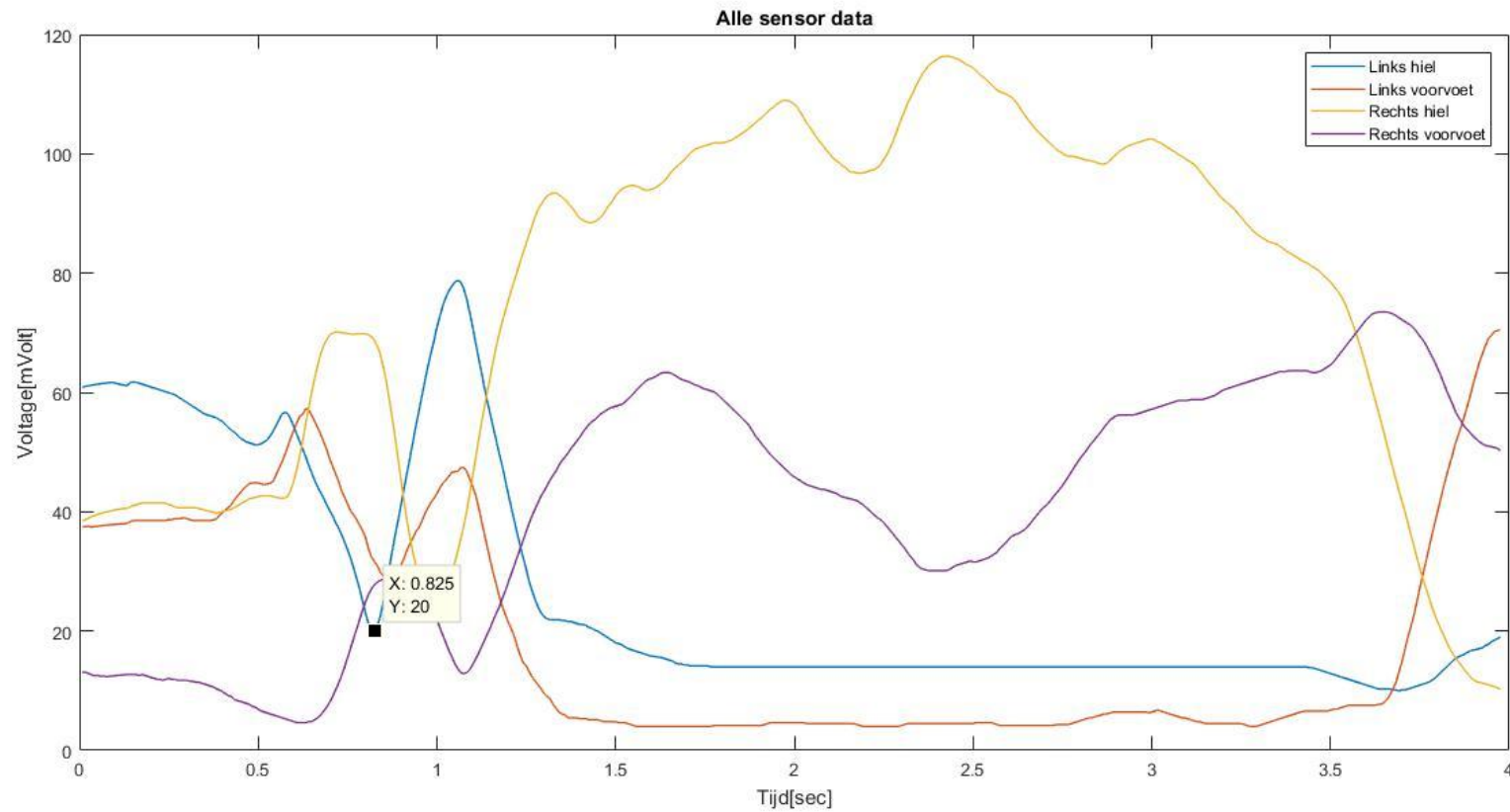


Figuur 3.4 Alle gecalibreerde data.
X-as Tijd[sec] Y-as Gewicht[gram]

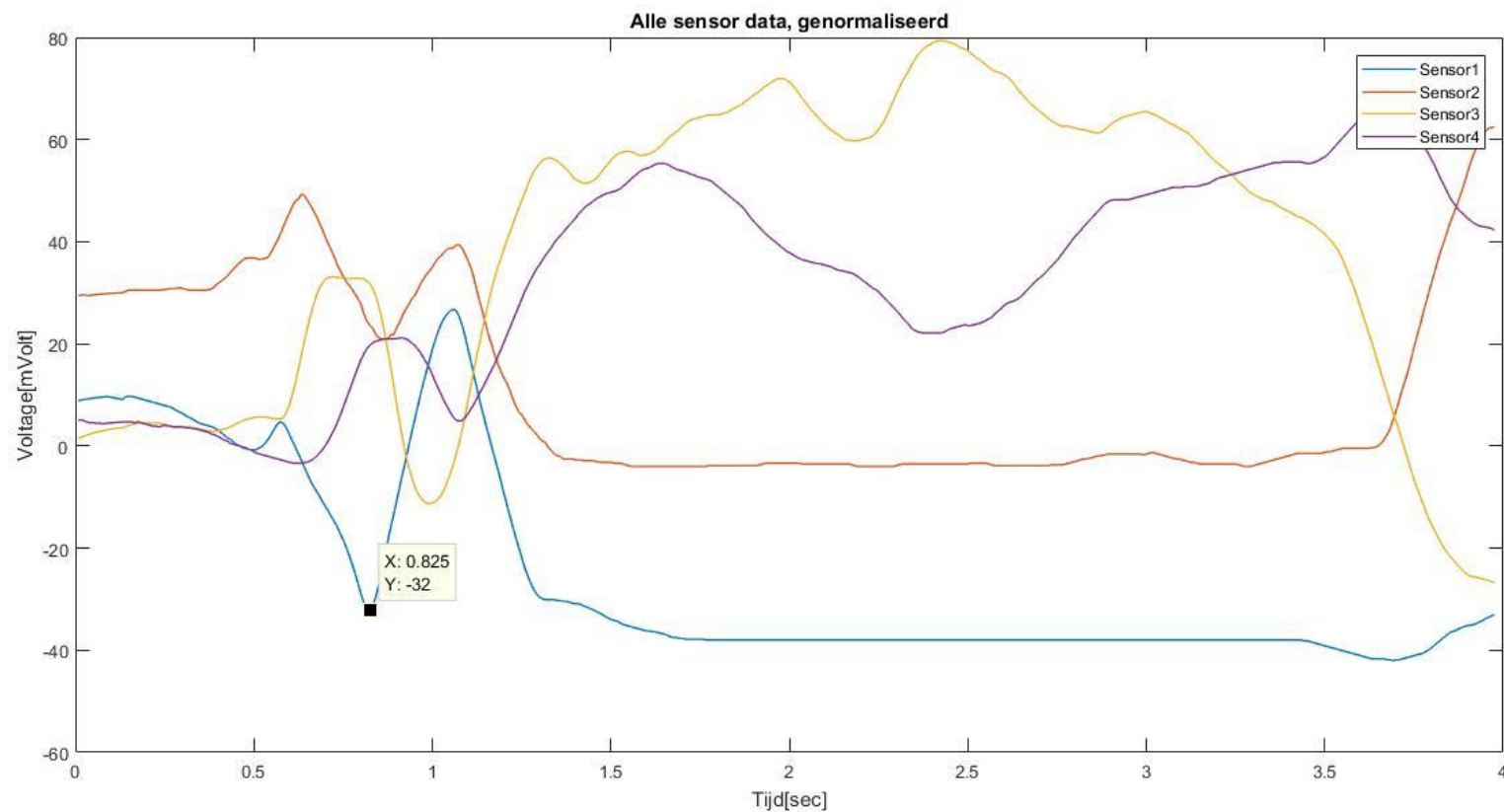


Figuur 3.5 Alle gecalibreerde data, genormaliseerd.
X-as Tijd[sec] Y-as Gewicht[gram]

Oefening 4. Glijden in de vallijn, optillen linker been

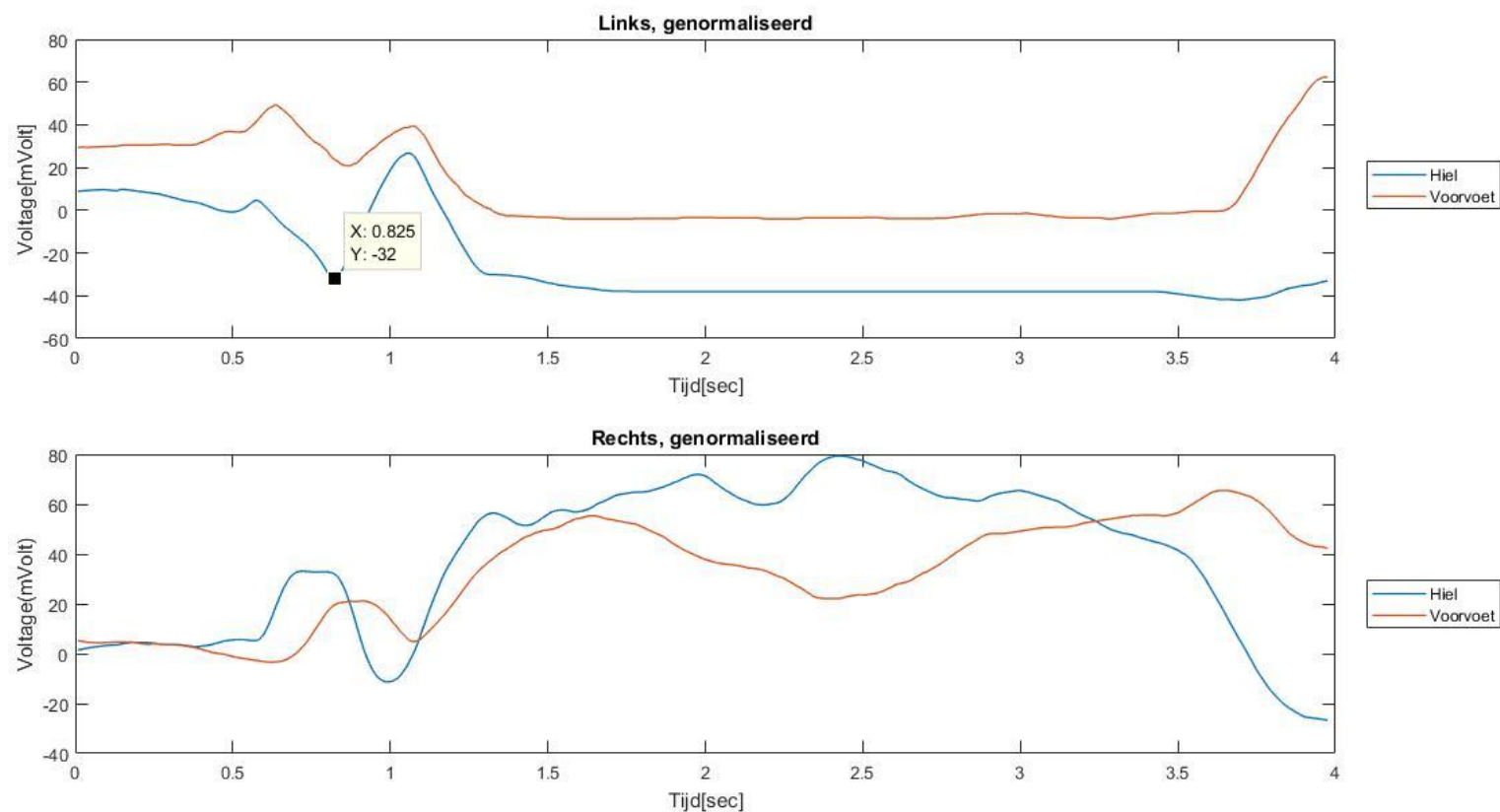


Figuur 4.1 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.
X-as Tijd[sec] Y-as Voltage [mVolt]

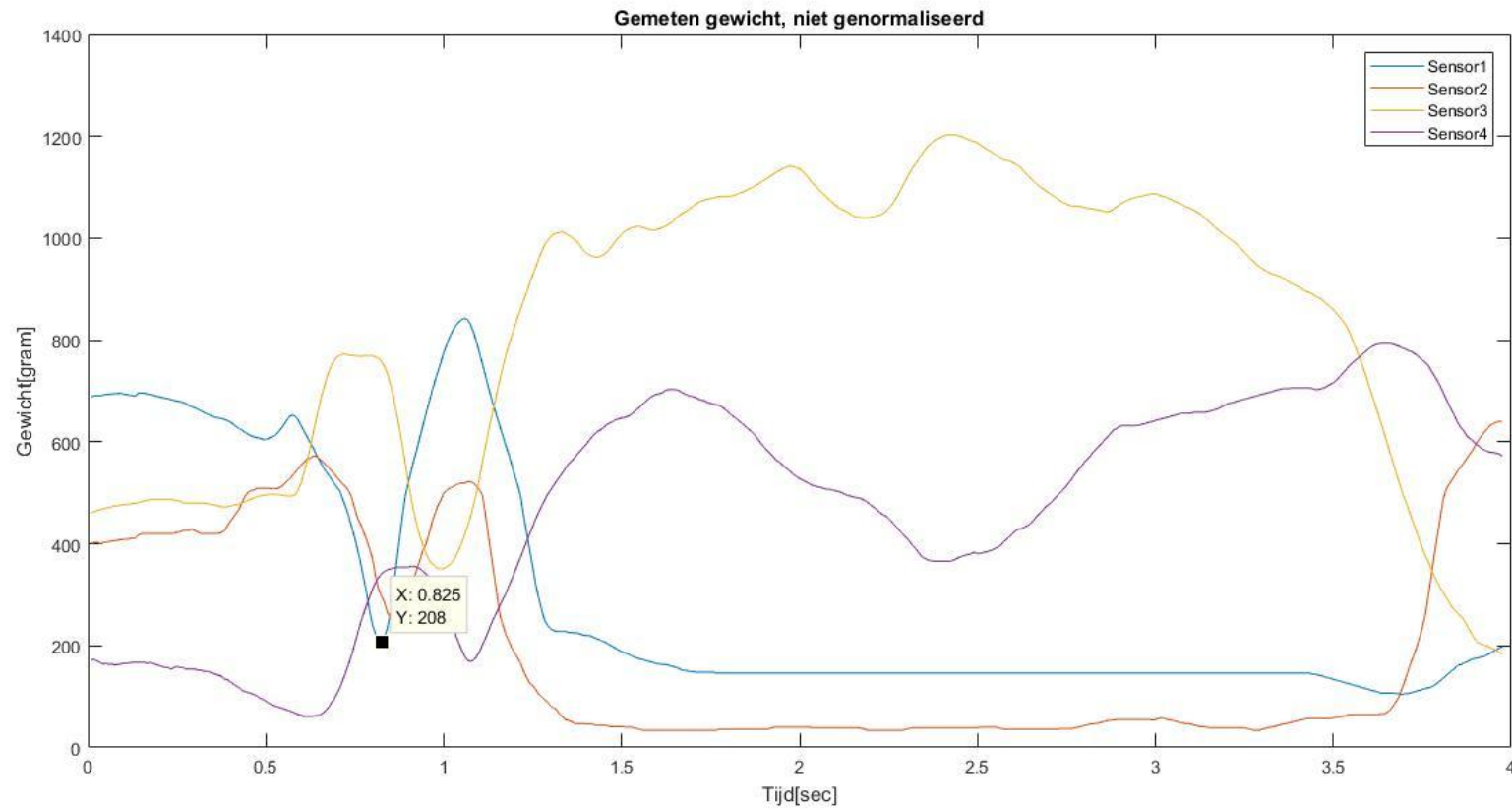


Figuur 4.2 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
X-as Tijd[sec] Y-as Voltage [mVolt]

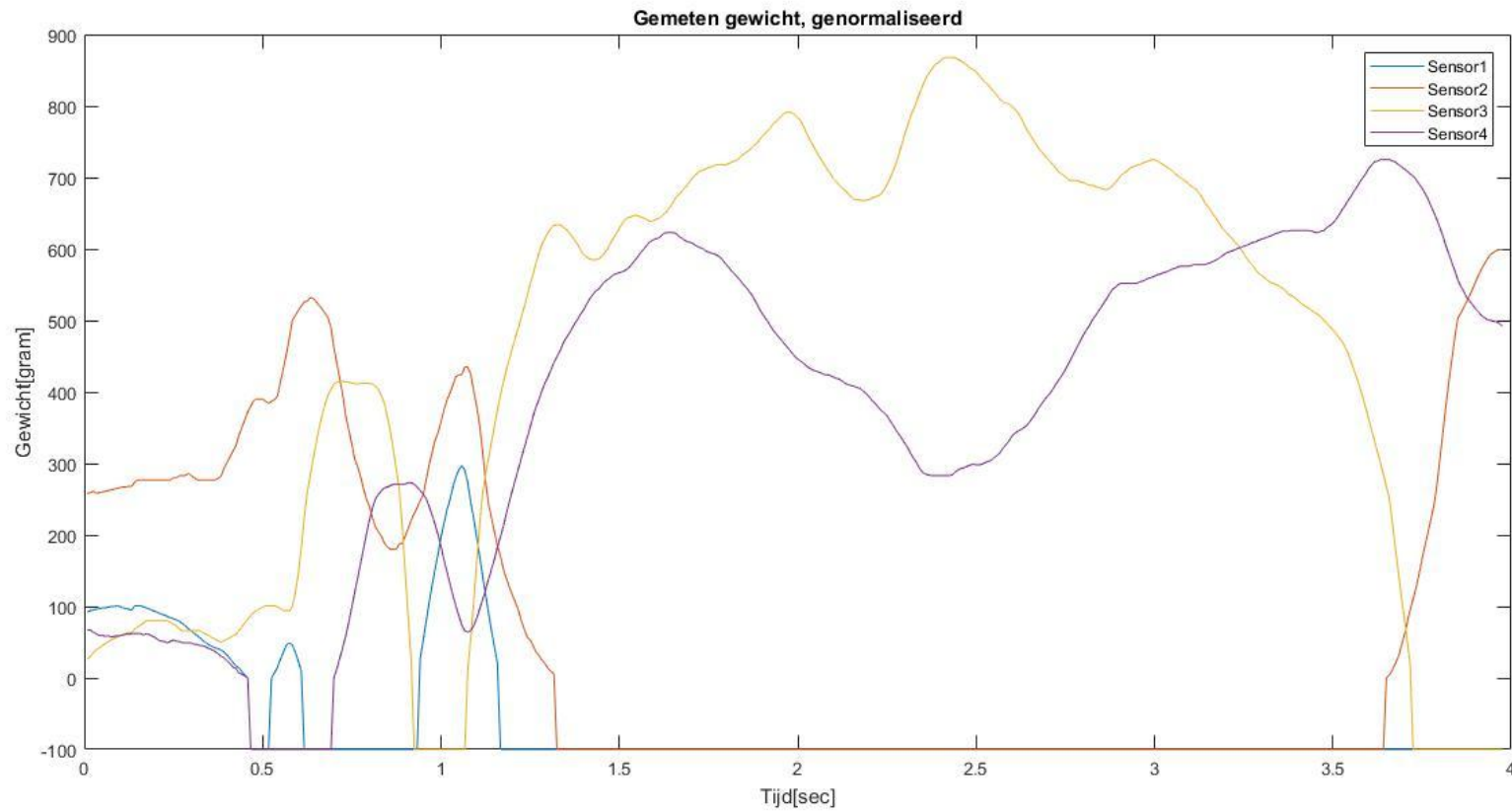
De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.



Figuur 4.3 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
Boven: Linkervoet. Onder: Rechtervoet X-as Tijd[sec] Y-as Voltage [mVolt]

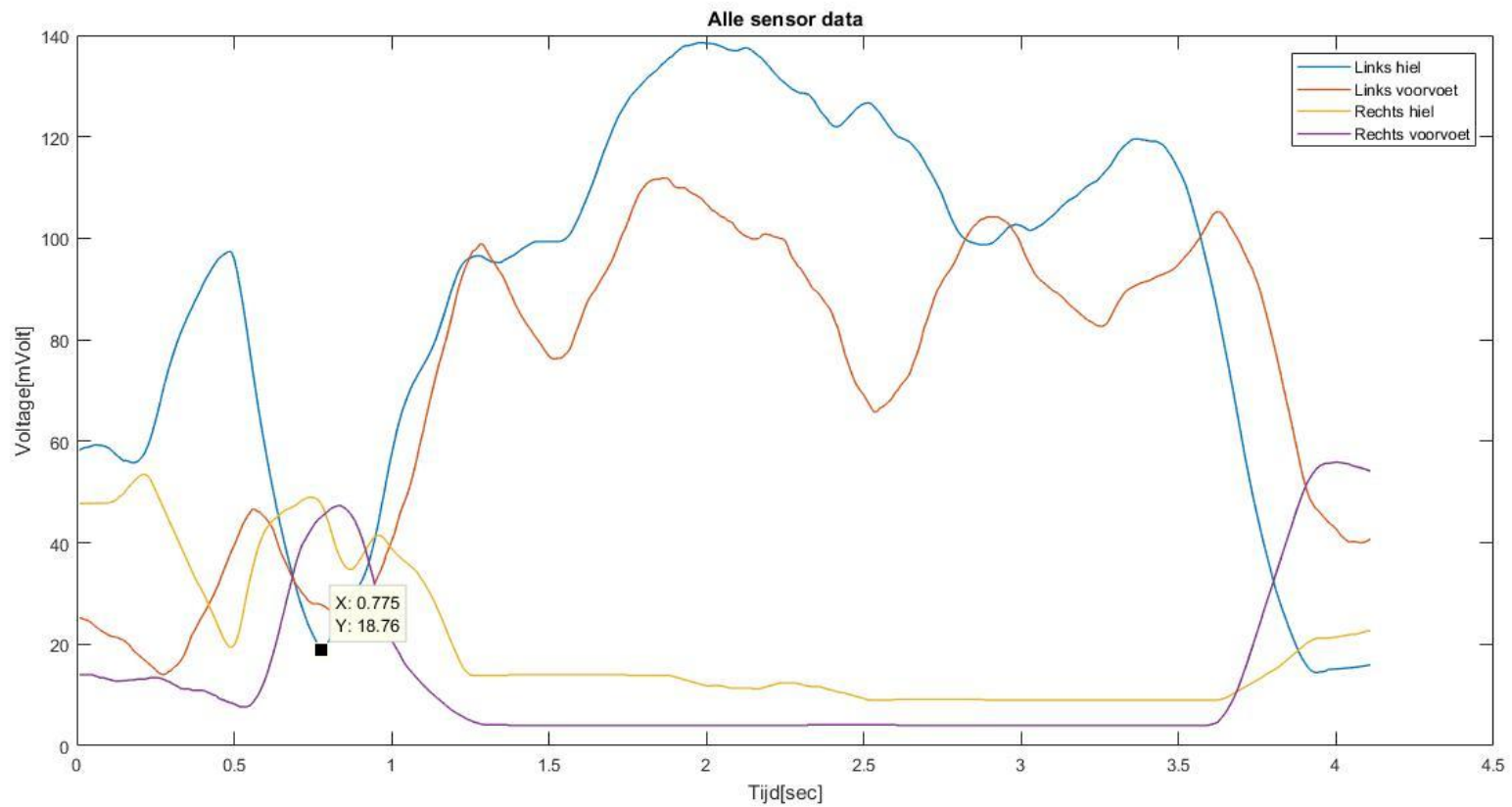


Figuur 4.4 Alle gecalibreerde data.
X-as Tijd[sec] Y-as Gewicht[gram]

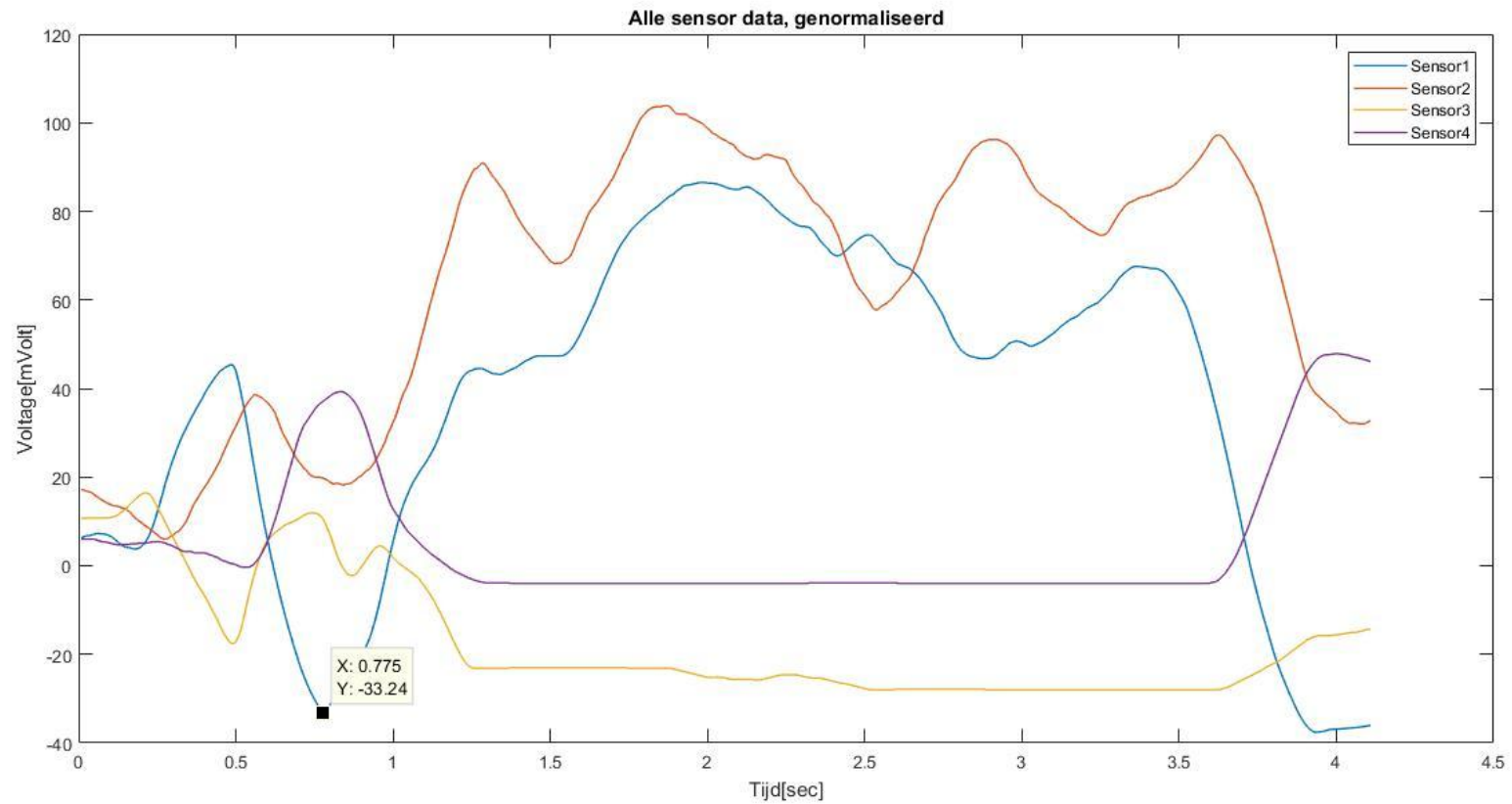


Figuur 4.5 Alle gecalibreerde data, genormaliseerd.
X-as Tijd[sec] Y-as Gewicht[gram]

Oefening 5. Glijden in de vallijn, optillen rechter been

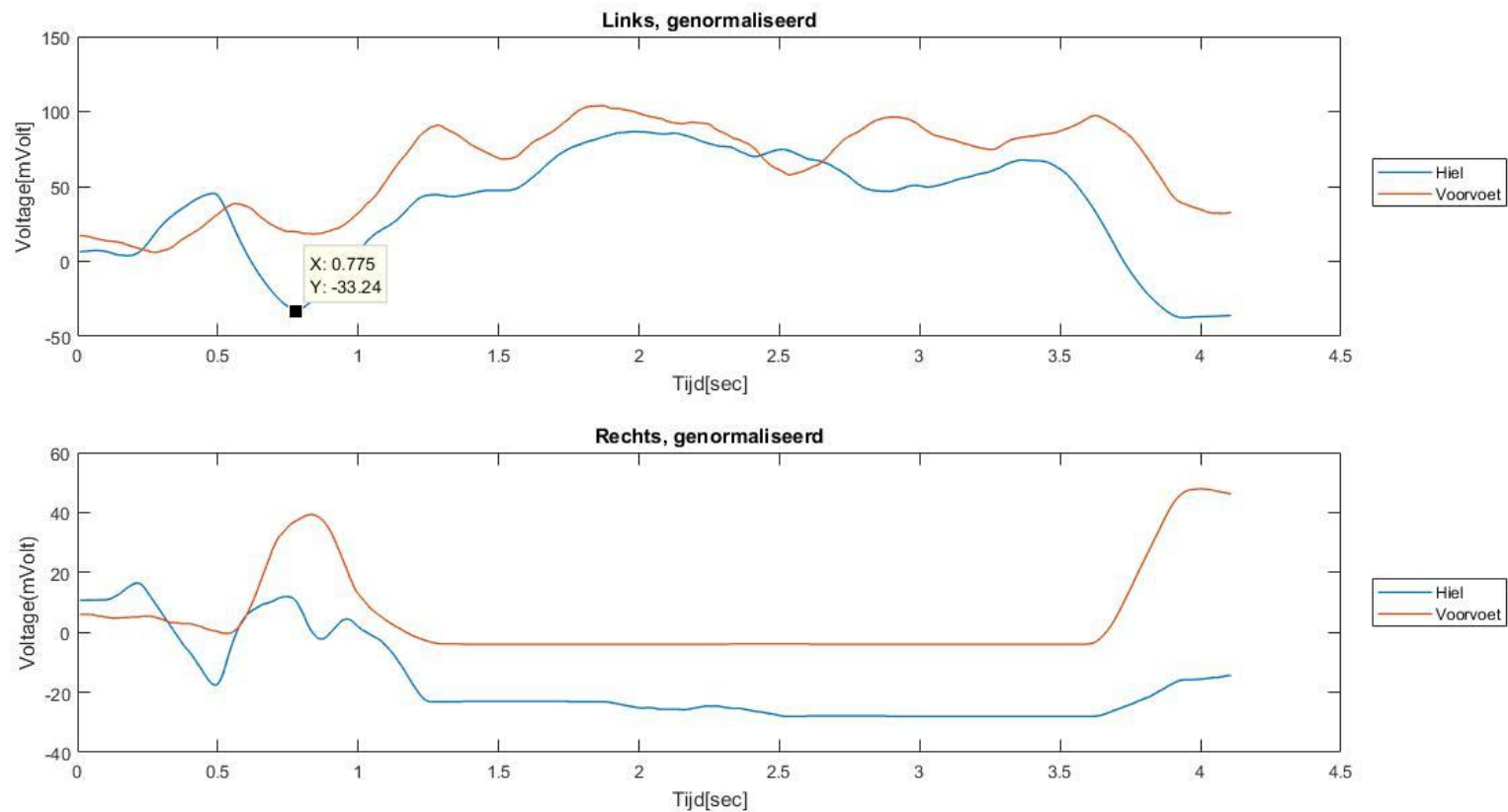


Figuur 5.1 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.
X-as Tijd[sec] Y-as Voltage [mVolt]

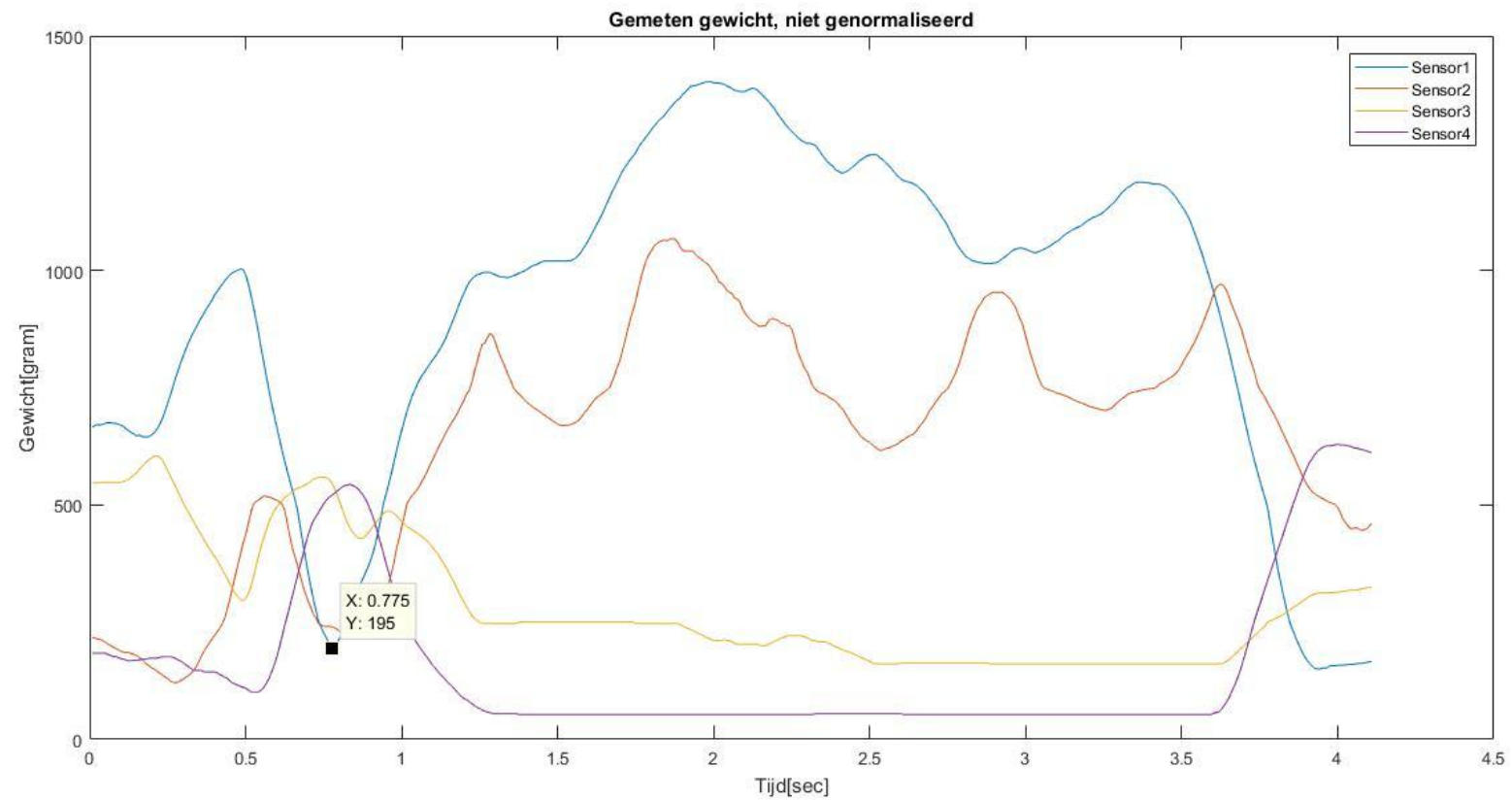


Figuur 5.2 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
X-as Tijd[sec] Y-as Voltage [mVolt]

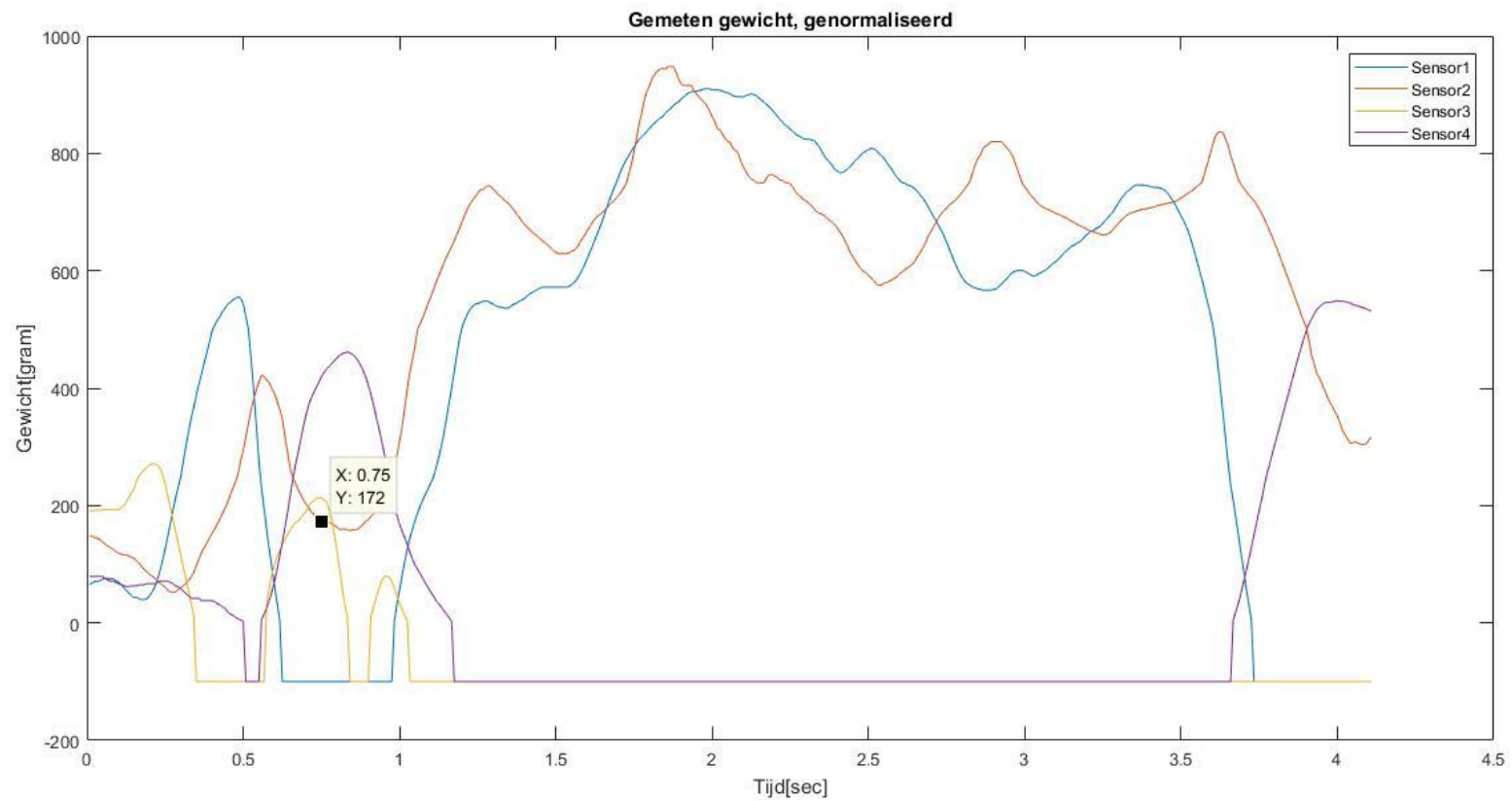
De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.



Figuur 5.3 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
Boven: Linkervoet. Onder: Rechervoet. X-as Tijd[sec] Y-as Voltage [mVolt]

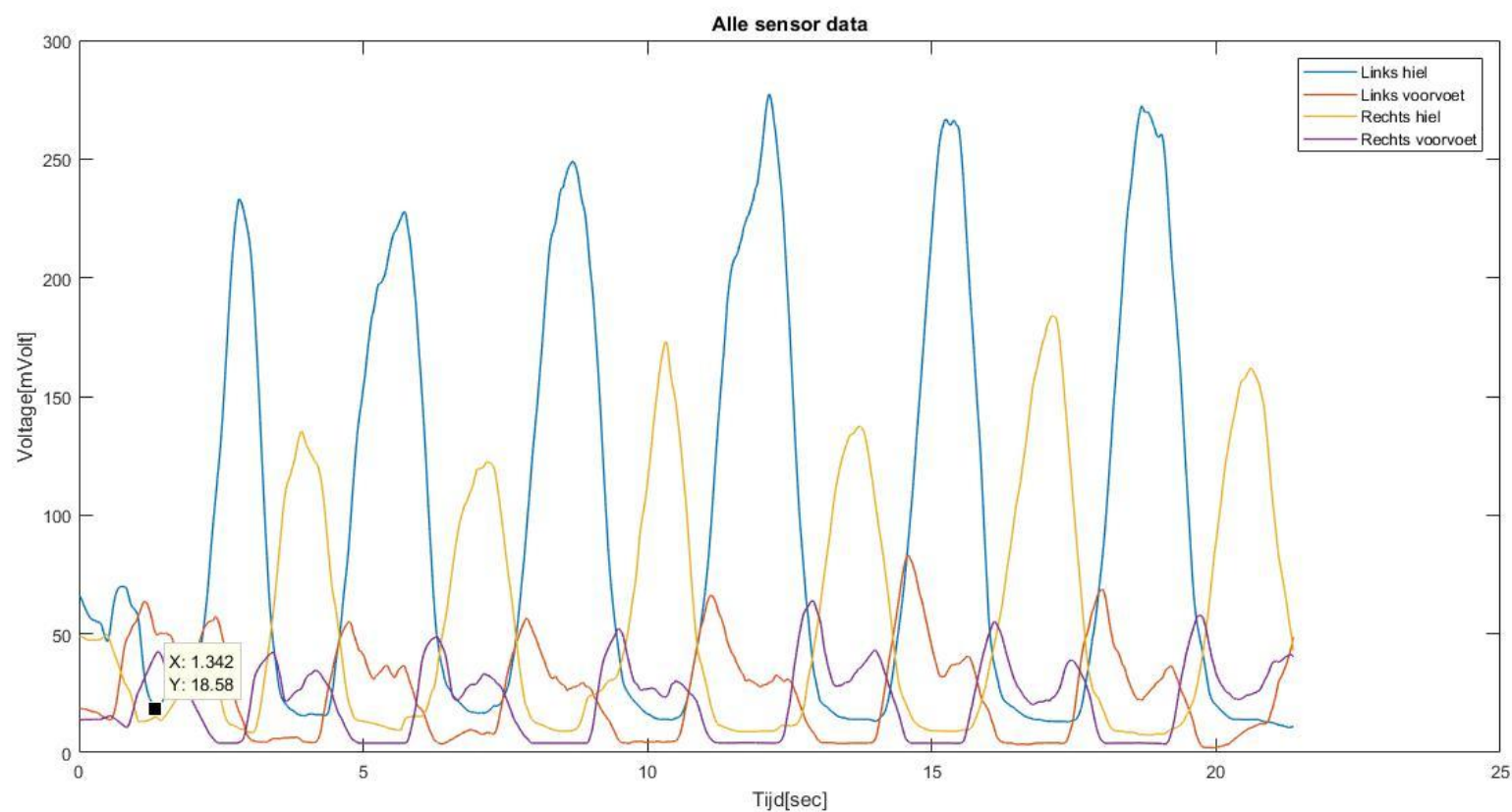


Figuur 5.4 Alle gecalibreerde data.
X-as Tijd[sec] Y-as Gewicht[gram]

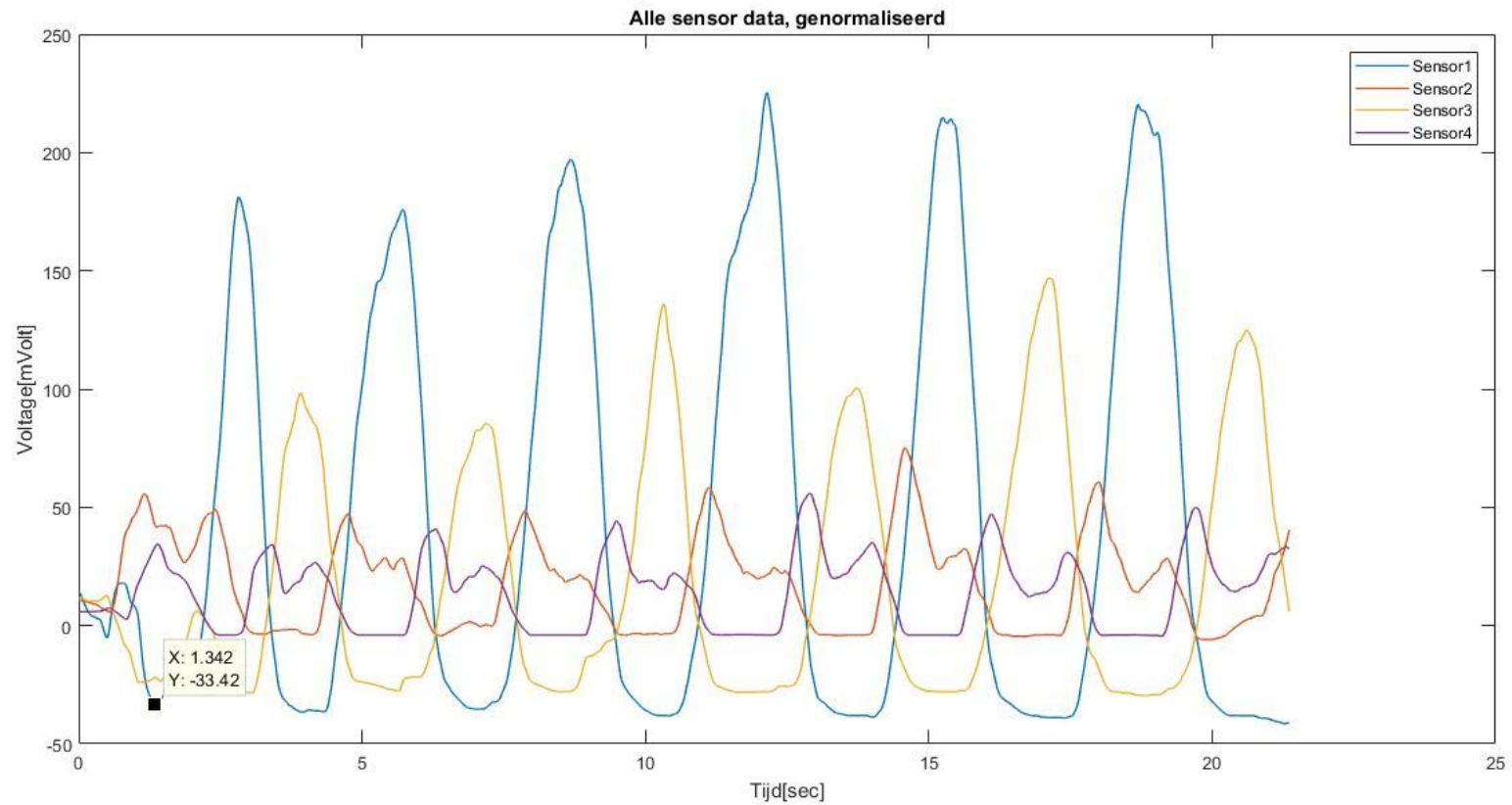


Figuur 5.5 Alle gecalibreerde data, genormaliseerd.
X-as Tijd[sec] Y-as Gewicht[gram]

Oefening 6. Schools skiën met optillen bergbeen

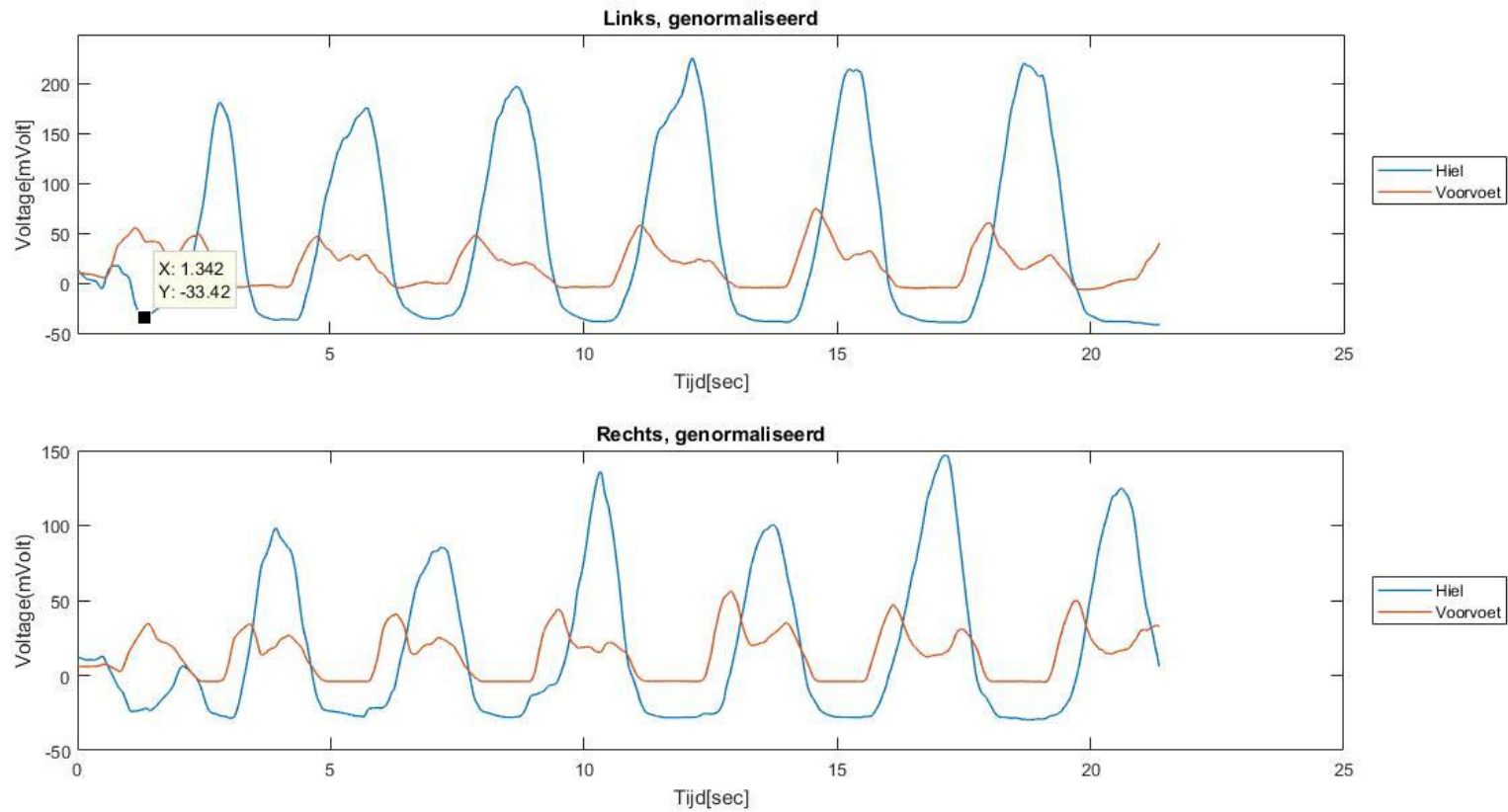


Figuur 6.1 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.
X-as Tijd[sec] Y-as Voltage [mVolt]

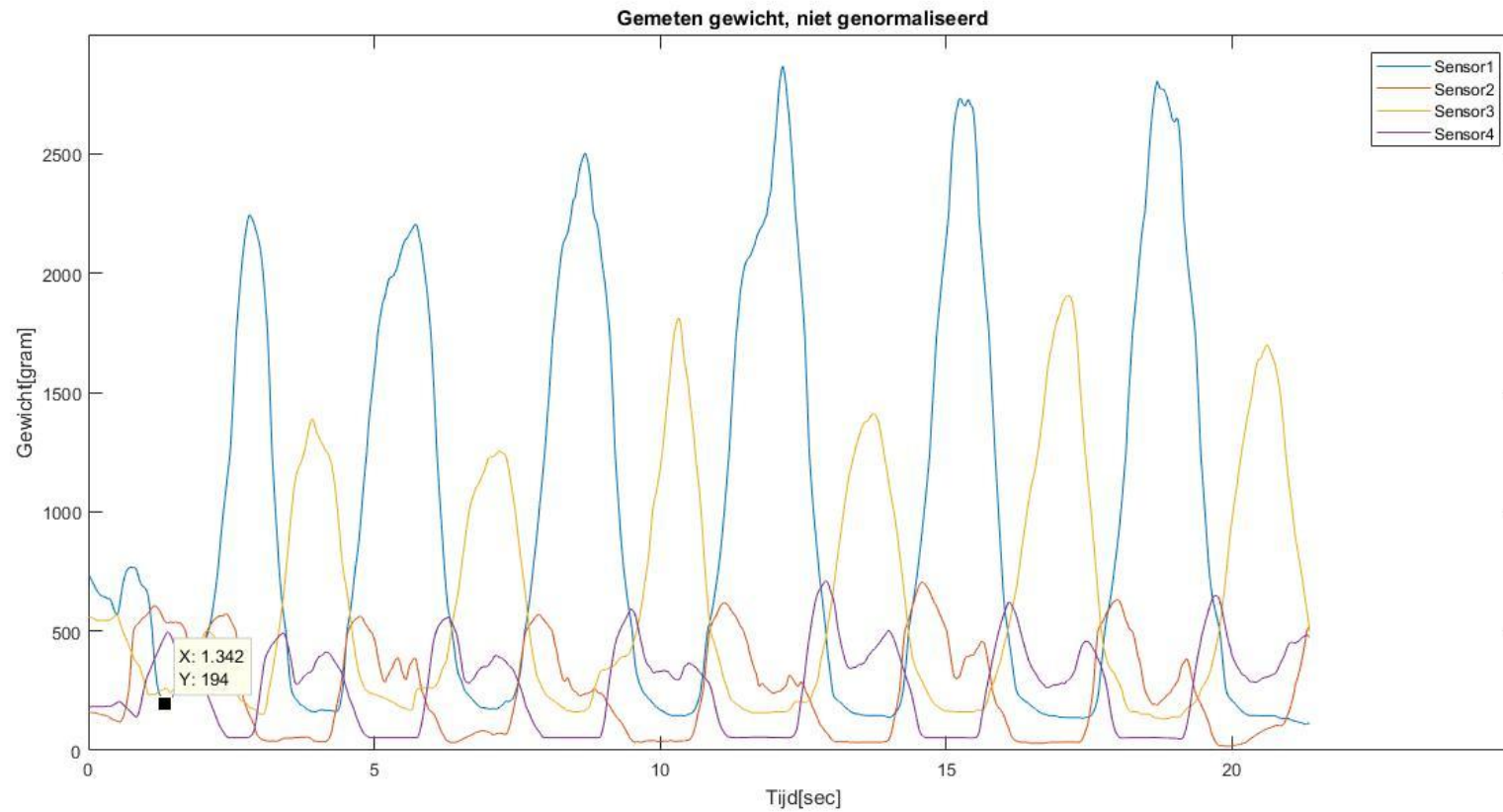


Figuur 6.2 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
X-as Tijd[sec] Y-as Voltage [mVolt]

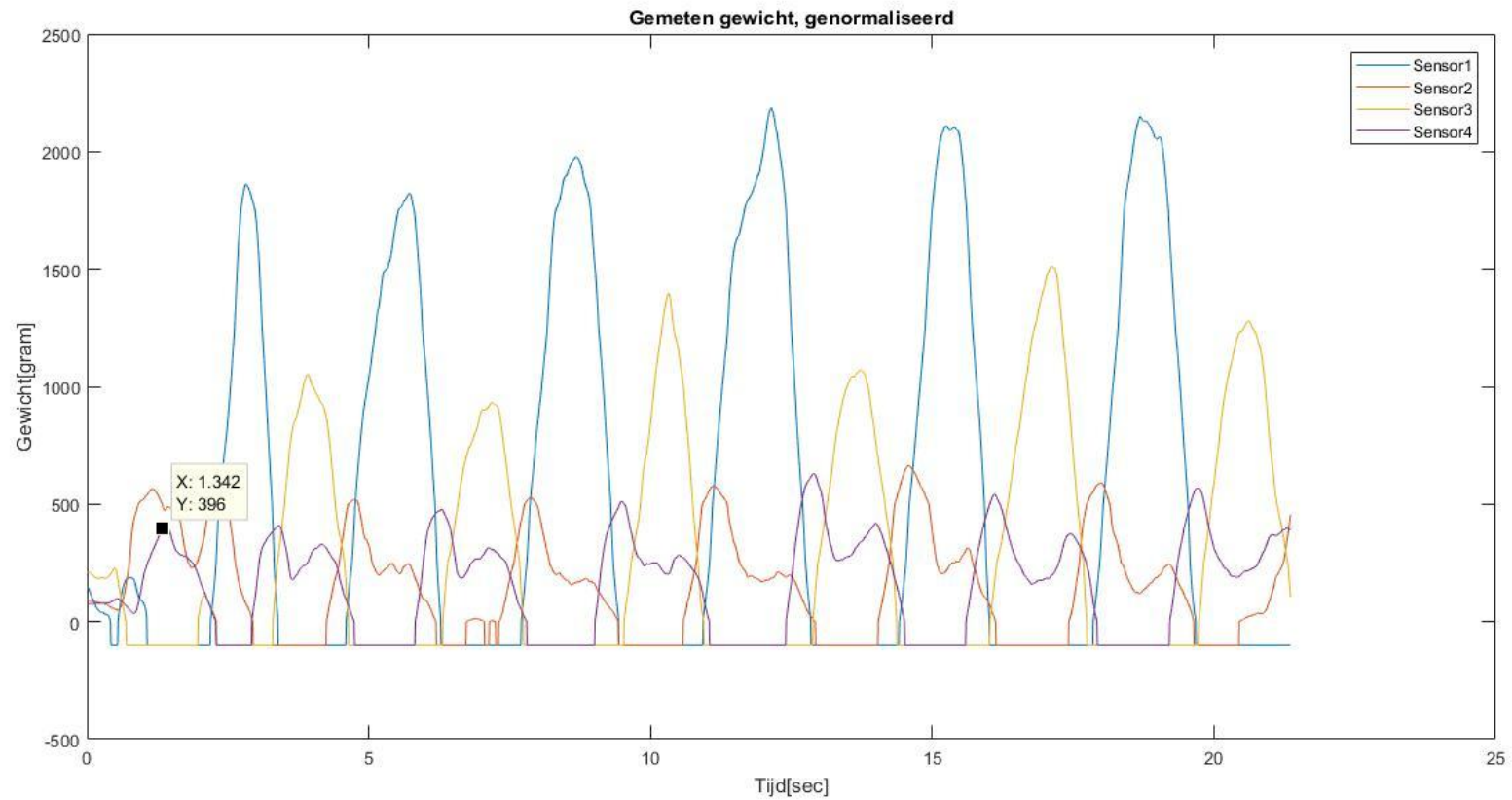
De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.



Figuur 6.3 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
Boven: Linkervoet. Onder: Rechervoet X-as Tijd[sec] Y-as Voltage [mVolt]

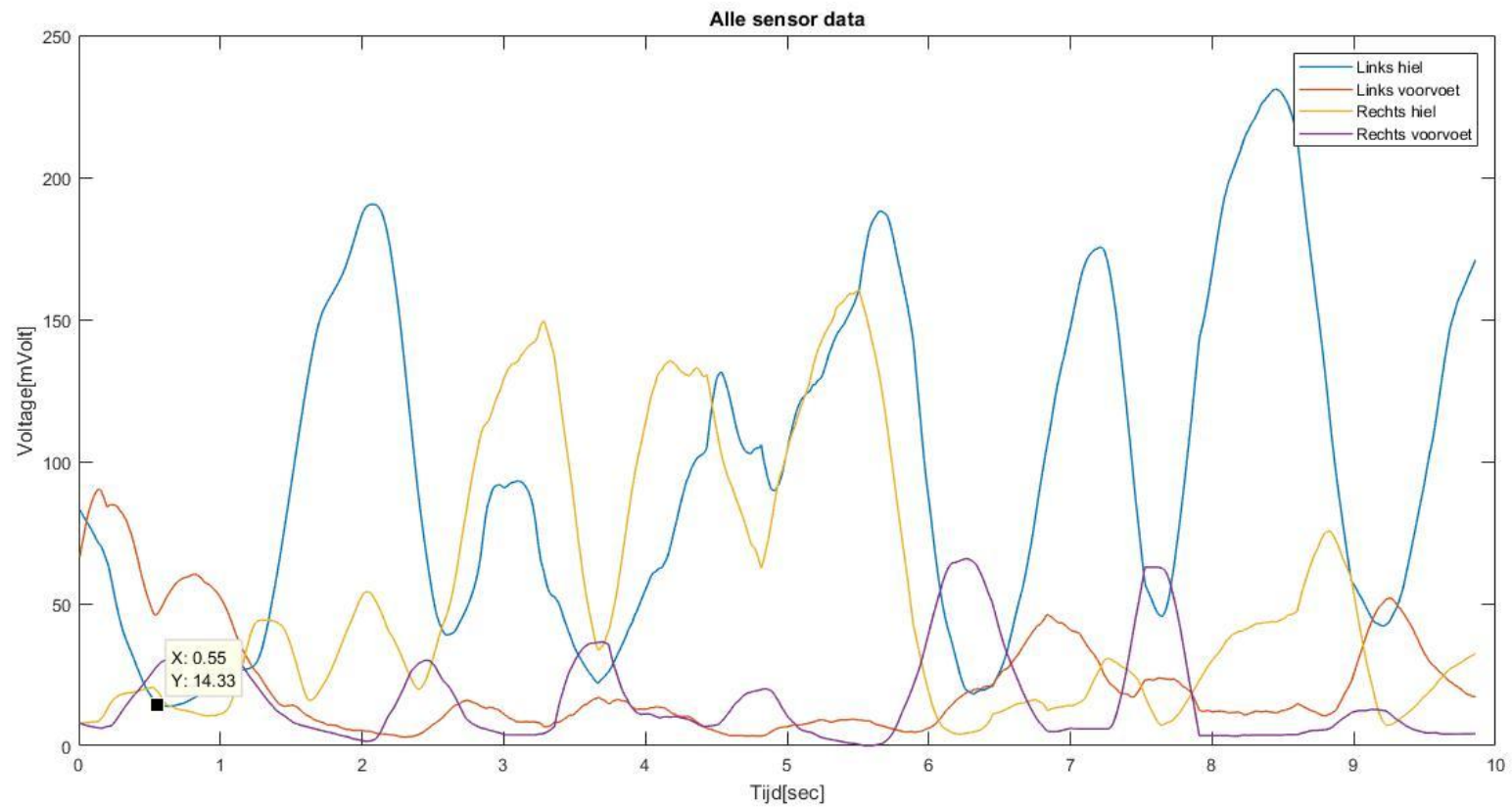


Figuur 6.4 Alle gecalibreerde data.
X-as Tijd[sec] Y-as Gewicht[gram]

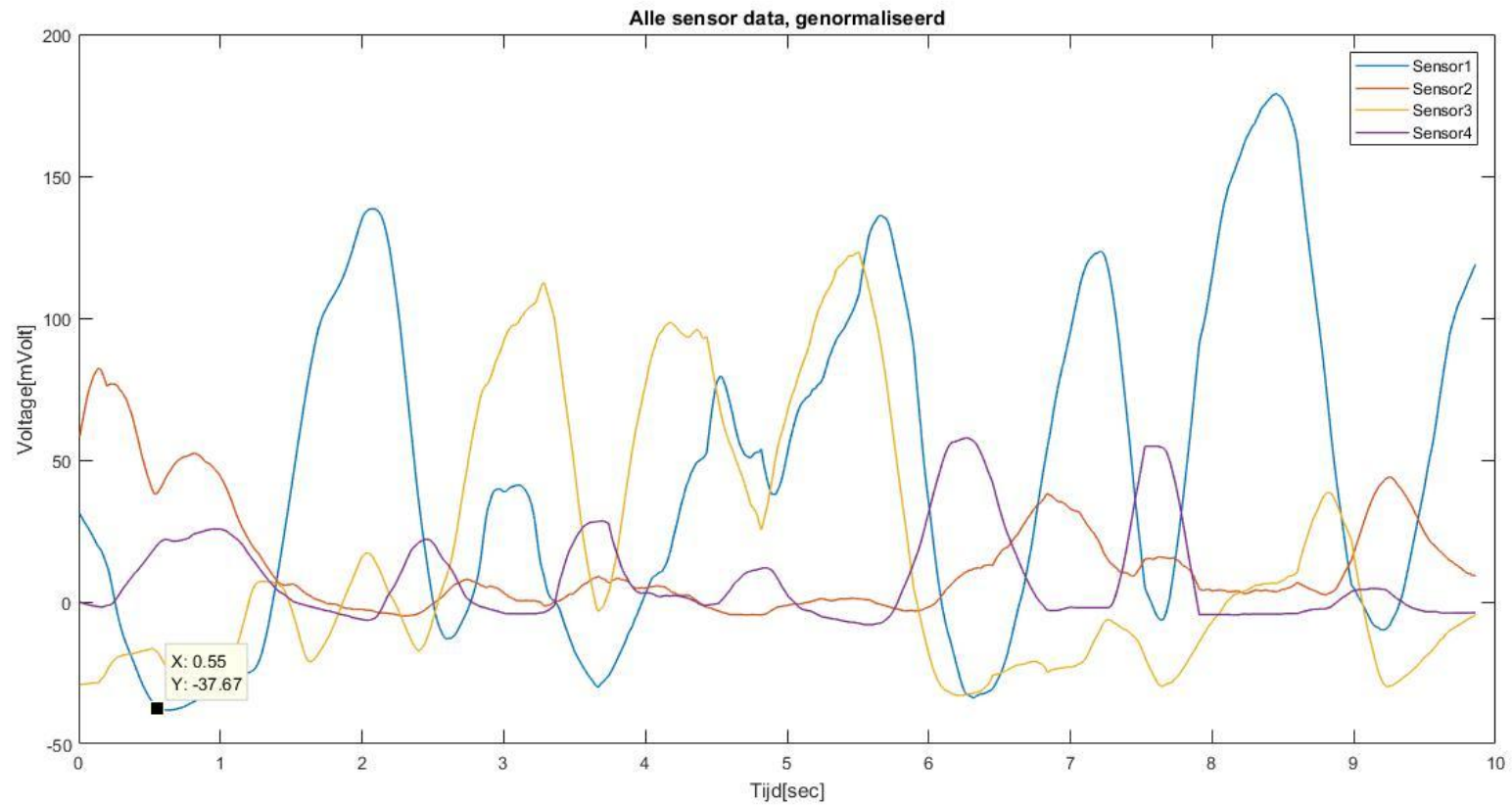


Figuur 6.5 Alle gecalibreerde data, genormaliseerd.
X-as Tijd[sec] Y-as Gewicht[gram]

Oefening 7: Schools skiën Centraal-Achterin-Voorin-Centraal

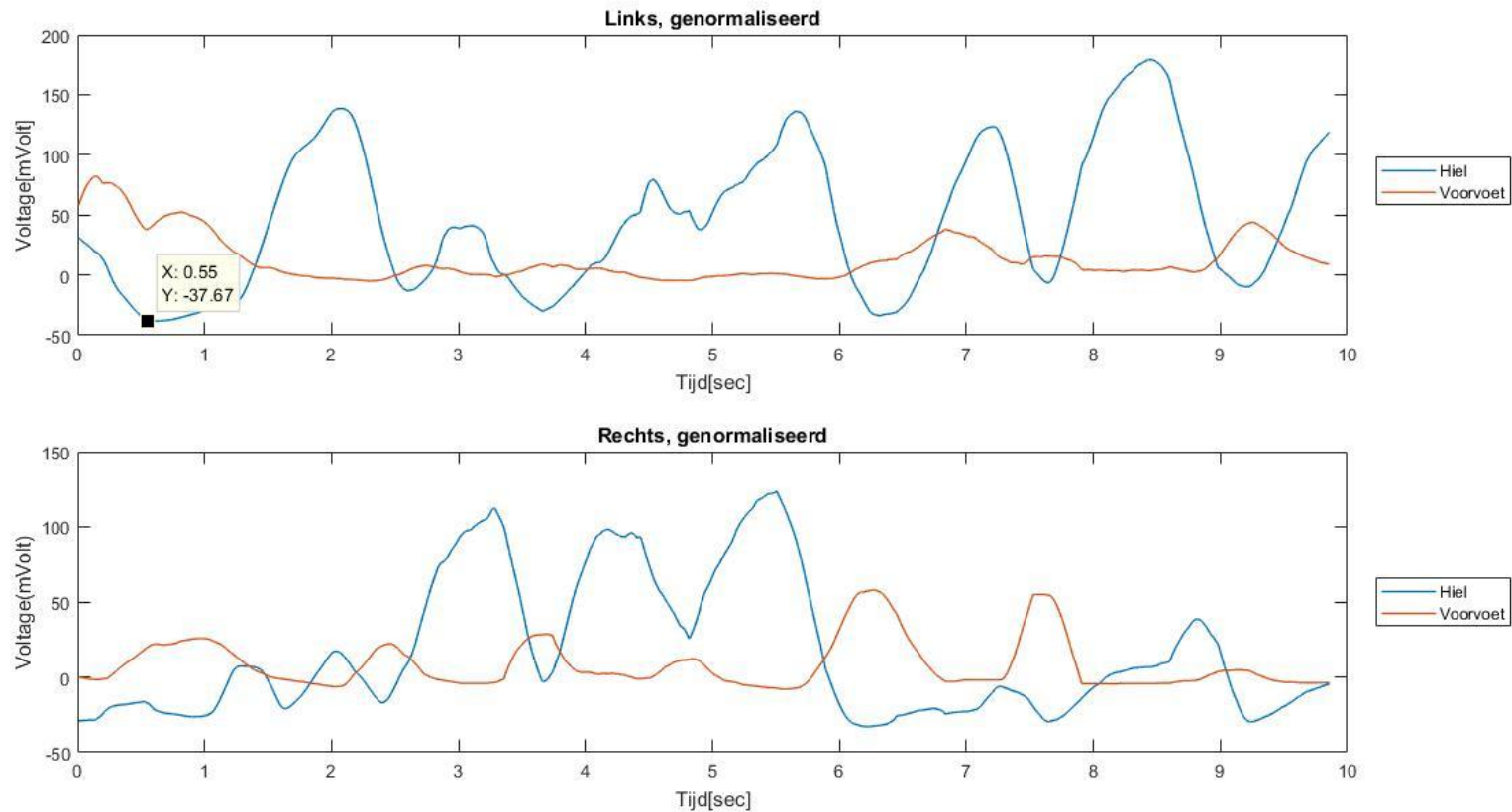


Figuur 7.1 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting.
X-as Tijd[sec] Y-as Voltage [mVolt]

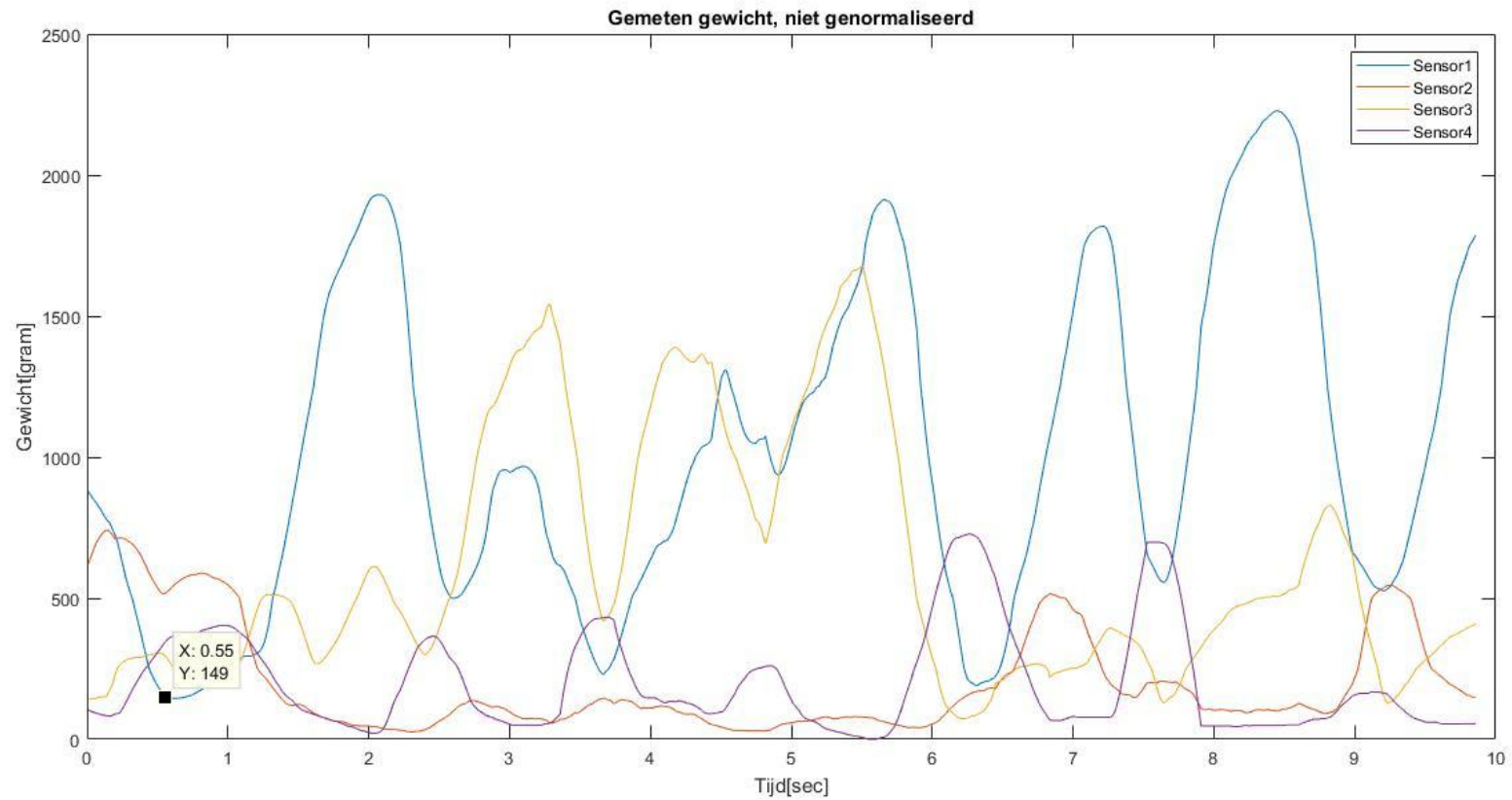


Figuur 7.2 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
X-as Tijd[sec] Y-as Voltage [mV]

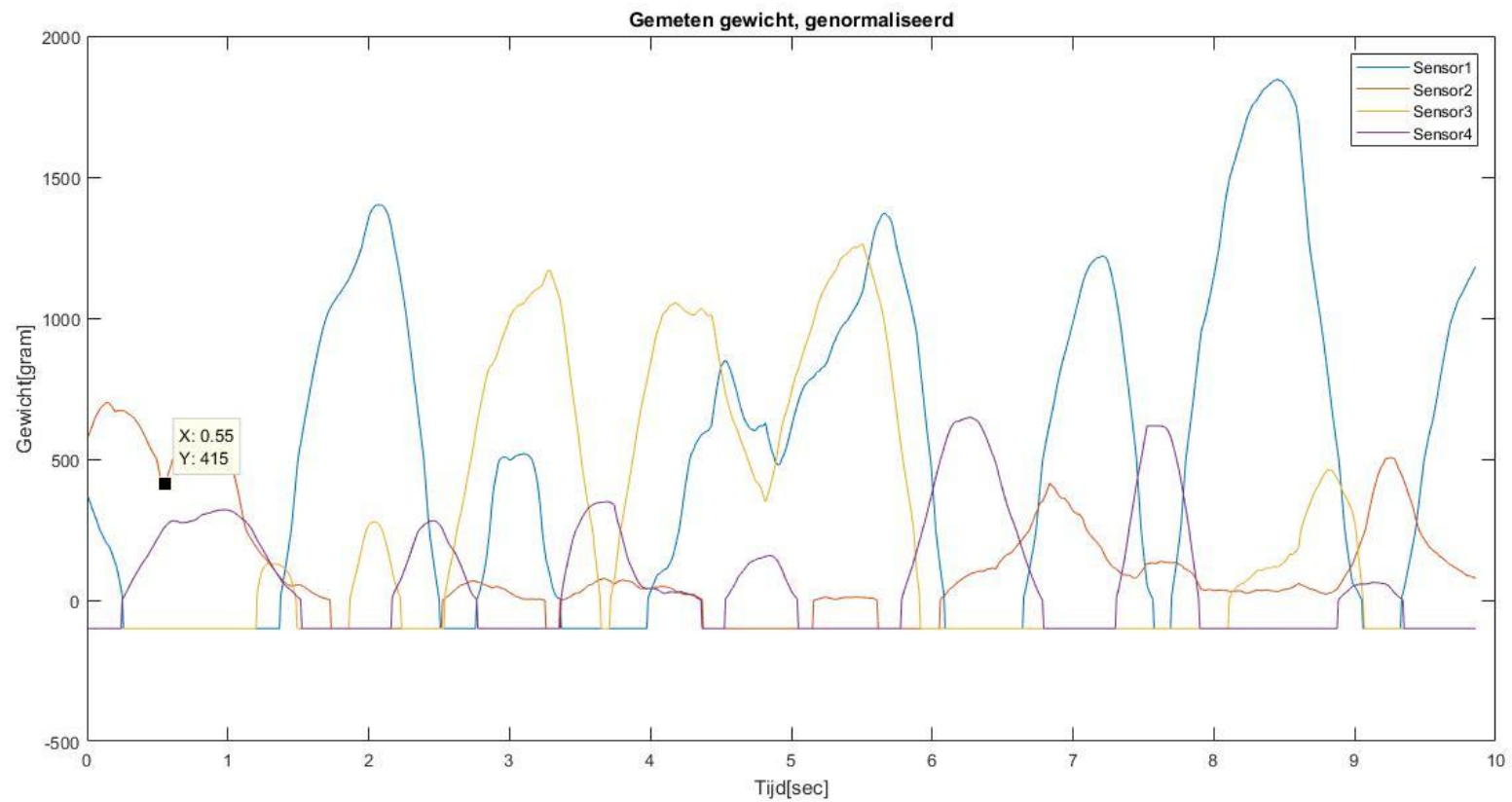
De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.



Figuur 7.3 Alle data zoals deze op de SD-kaart worden opgeslagen tijdens de meting, genormaliseerd.
Boven: Linkervoet. Onder: Rechervoet X-as Tijd[sec] Y-as Voltage [mVolt]



Figuur 7.4 Alle gecalibreerde data.
X-as Tijd[sec] Y-as Gewicht[gram]



Figuur 7.5 Alle gecalibreerde data, genormaliseerd.
X-as Tijd[sec] Y-as Gewicht[gram]

Bijlage IV – Arduino Scripts

Weergave Seriële monitor

In de eerste regels van het script worden de sensoren en de parameters geïdentificeerd.

```
int FSR1 = A0;  
int FSR2 = A1;  
int FSR3 = A2;  
int FSR4 = A3;  
int voltage;
```

Hierna wordt de void setup eenmalig toegepast. Hierin wordt contact gemaakt met de seriële monitor en worden hier de getallen 1 t/m 4 weergegeven. Dit refereert met de vier sensoren. Vervolgens worden FSR1 t/m FSR4 als OUTPUT gezien.

```
void setup()  
{  
  Serial.begin(9600);  
  Serial.print(" ");  
  Serial.print("1");  
  Serial.print(" ");  
  Serial.print("2");  
  Serial.print(" ");  
  Serial.print("3");  
  Serial.print(" ");  
  Serial.print("4");  
  Serial.print(" ");  
  Serial.println();
```

```
  for (int FSR = A0; FSR<A4; FSR++){  
    pinMode(FSR, INPUT);  
  }  
}
```

De void loop wordt oneindig herhaalt. Hierin worden de vier sensoren uitgelezen. Omdat deze door de Arduino worden ontvangen op een analoge poort krijgen deze de waarde 0 t/m 1023. Dit wordt in verhouding omgezet naar de 0 t/m 5000m Volt die de Arduino door de sensoren heen stuurt. De meetwaardes worden iedere 0,10sec gemeten en naast elkaar weergegeven in de seriële monitor.

```
void loop() {  
  for (int FSR = A0; FSR<A4; FSR++){  
    int sensorVal = analogRead(FSR);  
  
    voltage = map(sensorVal, 0, 1023, 0, 5000);  
  
    Serial.print(voltage);  
    Serial.print(" ");  
  }  
  Serial.println();  
  delay(100); }
```

Opslaan op SD kaart

In de eerste regels van het script worden de sensoren en de parameters geïdentificeerd. En wordt er contact gemaakt met de realtimeclock die zich in de SD-shield bevindt. De #include --.h bestanden maken het mogelijk dat de Arduino met een andere shield kan communiceren.

```
#include <SPI.h>
#include <SD.h>
#include "RTCLib.h"
```

```
int FSR1 = A0;
int FSR2 = A1;
int FSR3 = A2;
int FSR4 = A3;
```

```
const int pushbutton = 6;
const int ledgreen = 7;
const int chipSelect = 10;
```

```
int voltage;
File myfile;
```

```
RTC_PCF8523 rtc;
```

Hierna wordt de void setup eenmalig toegepast. Hierin worden de vier sensoren uit OUTPUT gezien. Vervolgens wordt er contact gemaakt met de SD-shield en gecontroleerd of de SD kaart aanwezig is. Tijdens het checken van de SD kaart wordt dit op de seriële monitor, indien aangesloten, weergegeven. Indien aanwezig knippert het groene LED kort en ook dit wordt weergegeven in de seriële monitor.

```
void setup() {

    // PINMODE
    pinMode(pushbutton, INPUT);
    pinMode(ledgreen, OUTPUT);
    pinMode(chipSelect, OUTPUT);

    for (int FSR = A0 ; FSR < A4; FSR++) {
        pinMode(FSR, INPUT);
    }

    //start monitor
    Serial.begin(9600);

    //Controleren of SD aanwezig is
    Serial.print("Initializing SD card.....");

    if (!SD.begin(chipSelect)) {
        Serial.println("Card failed, or not present");
        return;
    }
    else {
        Serial.println("Card initalized succesful");
        digitalWrite(ledgreen, HIGH);
        delay(1000);
        digitalWrite(ledgreen, LOW);
        delay(1000);
    }
}
```



```
digitalWrite(ledgreen, HIGH);}}
```

De void loop bestaat uit twee delen; Aan/uit knop staat uit en de aan/uit knop staat aan.

Als deze is uitgeschakeld wordt op de seriële monitor weergegeven dat de gebruiker deze knop aan moet zetten. Deze melding wordt iedere 3 seconden herhaalt.

```
void loop() {
```

```
while (!digitalRead(pushbutton)) {  
Serial.println("Turn pushbutton ON to start measurement");  
digitalWrite(ledgreen, LOW);  
myfile.close();  
delay(3000);  
}
```

Als de aan/uit knop is ingeschakeld wordt de datum en tijdstip weergegeven in de seriële monitor. Vervolgens wordt er een bestand aangemaakt waarin de data opgeslagen kan worden. De eerste regel in dit bestand weergeeft de datum en tijdstip waarop de meting is gestart.

```
if (digitalRead(pushbutton)){  
// Datum weergegeven in Serial  
rtc.begin();  
  
DateTime now = rtc.now();  
Serial.println();  
Serial.print(now.day(), DEC);  
Serial.print('/');  
Serial.print(now.month(), DEC);  
Serial.print('/');  
Serial.print(now.year(), DEC);  
Serial.print(" ");  
Serial.print(now.hour(), DEC);  
Serial.print(':');  
Serial.print(now.minute(), DEC);  
Serial.print(':');  
Serial.print(now.second(), DEC);  
Serial.print("");  
Serial.println();  
  
// bestand aanmaken  
char filename[] = "LOGGER00.txt";  
for (uint8_t i = 0; i < 100; i++) {  
filename[6] = i / 10 + '0';  
filename[7] = i % 10 + '0';  
if (!SD.exists(filename)) {  
myfile = SD.open(filename, FILE_WRITE);  
break;  
}  
}  
if (!myfile) {  
Serial.println("could not create a new file");  
}  
else {  
Serial.print("Logging to: ");  
Serial.print(filename);  
Serial.println();
```

```
myfile.println();  
myfile.print(now.day(), DEC);  
myfile.print('/');  
myfile.print(now.month(), DEC);  
myfile.print('/');  
myfile.print(now.year(), DEC);  
myfile.print(" ");  
myfile.print(now.hour(), DEC);  
myfile.print(':');  
myfile.print(now.minute(), DEC);  
myfile.print(':');  
myfile.print(now.second(), DEC);  
myfile.print(" ");  
myfile.println();  
}  
}
```

Vervolgens worden de vier sensoren met een frequentie van 120Hz uitgelezen. Direct nadat er vier sensoren een keer zijn gemeten worden deze vier waardes naar de SD kaart geschreven.

```
while (digitalRead(pushbutton)) {  
  
    digitalWrite(ledgreen, HIGH);  
    //uitlezen fsr sensoren en data weergegeven in monitor en opslaan op SD  
    for (int FSR = A0 ; FSR < A4; FSR++) {  
        int sensorVal = analogRead(FSR);  
        voltage = map(sensorVal, 0, 1023, 0, 5000);  
  
        myfile.print(voltage);  
        Serial.print(voltage);  
        myfile.print(" ");  
        Serial.print(" ");  
    }  
    myfile.println();  
    Serial.println();  
    delay(8.33);  
    myfile.flush();  
}  
myfile.close();  
return;  
}
```

Bijlage V – Matlab script

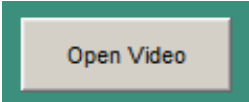
Hieronder volgt een uitleg van het Matlab Script. Dit script maakt de werking van de ontwikkelde UI mogelijk.

In de eerste 74 regels wordt de ontworpen UI opgestart. In de 2^e regel is te lezen welk fig.file bij het script hoort.

```
1. function varargout = SkiFeetAnalyzer(varargin)
2. % SKIFEETANALYZER MATLAB code for SkiFeetAnalyzer.fig
3. %     SKIFEETANALYZER, by itself, creates a new SKIFEETANALYZER or raises the exist
   ing
4. %     singleton*.
5. %
6. %     H = SKIFEETANALYZER returns the handle to a new SKIFEETANALYZER or the handle
   to
7. %     the existing singleton*.
8. %
9. %     SKIFEETANALYZER('CALLBACK',hObject,eventData,handles,...) calls the local
10.%     function named CALLBACK in SKIFEETANALYZER.M with the given input arguments.
11.%
12.%     SKIFEETANALYZER('Property','Value',...) creates a new SKIFEETANALYZER or rais
   es the
13.%     existing singleton*. Starting from the left, property value pairs are
14.%     applied to the GUI before SkiFeetAnalyzer_OpeningFcn gets called. An
15.%     unrecognized property name or invalid value makes property application
16.%     stop. All inputs are passed to SkiFeetAnalyzer_OpeningFcn via varargin.
17.%
18.%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
19.%     instance to run (singleton)".
20.%
21.% See also: GUIDE, GUIDATA, GUIHANDLES
22.
23.% Edit the above text to modify the response to help SkiFeetAnalyzer
24.
25.% Last Modified by GUIDE v2.5 13-Jun-2017 12:18:53
26.
27.% Begin initialization code - DO NOT EDIT
28.gui_Singleton = 1;
29.gui_State = struct('gui_Name',      mfilename, ...
30.                  'gui_Singleton',  gui_Singleton, ...
31.                  'gui_OpeningFcn', @SkiFeetAnalyzer_OpeningFcn, ...
32.                  'gui_OutputFcn',  @SkiFeetAnalyzer_OutputFcn, ...
33.                  'gui_LayoutFcn',  [] , ...
34.                  'gui_Callback',    []);
35.if nargin && ischar(varargin{1})
36.    gui_State.gui_Callback = str2func(varargin{1});
37.end
38.
39.if nargin
40.    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
41.else
42.    gui_mainfcn(gui_State, varargin{:});
43.end
44.% End initialization code - DO NOT EDIT
45.
46.
47.% --- Executes just before SkiFeetAnalyzer is made visible.
```

```
48.function SkiFeetAnalyzer_OpeningFcn(hObject, eventdata, handles, varargin)
49.% This function has no output args, see OutputFcn.
50.% hObject    handle to figure
51.% eventdata  reserved - to be defined in a future version of MATLAB
52.% handles    structure with handles and user data (see GUIDATA)
53.% varargin   command line arguments to SkiFeetAnalyzer (see VARARGIN)
54.
55.% Choose default command line output for SkiFeetAnalyzer
56.handles.output = hObject;
57.
58.% Update handles structure
59.guidata(hObject, handles);
60.
61.% UIWAIT makes SkiFeetAnalyzer wait for user response (see UIRESUME)
62.% uiwait(handles.figure1);
63.
64.% --- Outputs from this function are returned to the command line.
65.function varargout = SkiFeetAnalyzer_OutputFcn(hObject, eventdata, handles)
66.% varargout   cell array for returning output args (see VARARGOUT);
67.% hObject    handle to figure
68.% eventdata  reserved - to be defined in a future version of MATLAB
69.% handles    structure with handles and user data (see GUIDATA)
70.
71.% Get default command line output from handles structure
72.varargout{1} = handles.output;
73.
```

In regel 74 t/m 114 is de eerste *Callback* van het script, geprogrammeerd. Zodra de gebruiker op **pushbutton1** klikt wordt deze functie uitgevoerd.

A rectangular button with a green border and a light gray background, containing the text "Open Video".

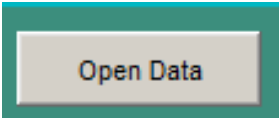
Als eerst worden **axes1**, **text2** en **text3** zichtbaar gemaakt. Bij het opstarten van de UI zijn deze namelijk niet direct zichtbaar. Indien de gebruiker voor de tweede keer een nieuwe video inlaad worden **checkboxvideo** en **pushbuttonknipvideo**, terug gezet naar hun begin waarde. Hier wordt de gebruiker gevraagd om een .avi film bestand te openen. Het script leest het bestand uit en het eerste frame van de video wordt getoond in **axes1**. Tevens wordt **slider1** ingesteld en zijn **text1**, **2** en **3**, zichtbaar.

A green rectangular box with white text "Frame nr. 1 / 352".

```
74.%-----OPEN VIDEO-----
75.% --- Executes on button press in pushbutton1.
76.function pushbutton1_Callback(hObject, eventdata, handles)
77.% hObject    handle to pushbutton1 (see GCBO)
78.% eventdata  reserved - to be defined in a future version of MATLAB
79.% handles    structure with handles and user data (see GUIDATA)
80.set(handles.axes1,'Visible','on');
81.statusplay = 1;
82.
83.set(handles.text1,'Visible','on');
84.set(handles.text2,'Visible','on');
85.set(handles.text3,'Visible','on');
86.set(handles.pushbuttonknipvideo,'ForegroundColor',[1.0 .0 .0]);
87.set(handles.checkboxvideo,'Value',0);
88.set(handles.text19,'Visible','off');
89.set(handles.textknipfilm,'String','1');
```

```
90. set(handles.textknipfilm, 'Visible', 'off');
91.
92. [FileNaam MapNaam] = uigetfile('*.avi', 'Selecteer een AVI file');
93. Naam = fullfile(MapNaam, FileNaam);
94. setappdata(gcf, 'FilmNaam', Naam);
95. VideoObject = VideoReader(Naam);
96. VideoFrames = read(VideoObject);
97. numFrames = VideoObject.NumberOfFrames;
98.
99. set(handles.text3, 'String', [' / ', num2str(numFrames)]);
100.     set(handles.text2, 'String', '1');
101.     set(handles.slider1, 'Value', (1));
102.     axes(handles.axes1);
103.     axis(handles.axes1, 'off');
104.     frame_1 = read(VideoObject, 1);
105.     imshow(frame_1)
106.     drawnow;
107.
108.     %% Slider 1 instellen
109.     set(handles.slider1, 'Min', 1)
110.     set(handles.slider1, 'Max', numFrames)
111.     set(handles.slider1, 'Value', 1)
112.     KleineStap = 1/(numFrames-1);
113.     GroteStap = 5/(numFrames-1);
114.     set(handles.slider1, 'SliderStep', [KleineStap GroteStap]);
```

De volgende *Callback* is die van *pushbutton2*: het openen van de data. Mocht de gebruiker voor de twee keer gebruik maken van deze *Callback*, worden *slider1*, *text5*, *text20* en *textknipdata* terug gezet naar de beginwaarden.



Open Data

Hierna wordt de gebruiker gevraagd om drie bestanden te openen: de data die geanalyseerd gewenst te worden, het Normalisatie bestand en het Calibratie-bestand. Met gebruik van de *xlsread* functie worden de Excel-bestanden uitgelezen. De totale lengte van het te analyseren bestand wordt zichtbaar in *text6*. De ruwe data van het te analyseren bestand wordt eerst gefilterd met de zogenoemde 'smoothfunctie'. Deze functie berekend met een stapgrootte een gemiddelde over de elementen van de vector. Dit zorgt er voor dat de data vloeiend wordt, zonder ruis.

Om de data te normaliseren wordt het Normalisatie-bestand uitgelezen en wordt de normalisatie waarde bepaald. Van iedere sensor wordt de gemiddelde waarde genomen en hier wordt een marge van 15% op ingebouwd. Dit omdat de waardes in het normalisatie bestand een spreiding hebben. Deze waarde wordt namelijk van de te analyseren data afgetrokken. Hiermee kan worden bepaald wanneer de gebruiker meer kracht levert dan zodra hij stil en centraal zou staan. De waarde 0 is dan gelijk aan 'normaal'. Zodra dit een min-waarde wordt, wordt er minder kracht dan 'normaal' gemeten.

De data wordt ook gecalibreerd. Dit wordt gedaan met behulp van Calibratie-bestand dat wordt uitgelezen. Door deze data te interpoleren wordt er per 100gram duidelijk welk Voltage er gemeten wordt. Uit het te analyseren bestanden volgen namelijk voltages, welke via de calibratie worden omgezet naar gewicht in gram. Zodra dit alles is berekend worden alle signalen in *axes2* zichtbaar gemaakt.

```
116.     %%-----OPEN DATA-----
117.     %% % --- Executes on button press in pushbutton8.
118.     function pushbutton2_Callback(hObject, eventdata, handles)
```

```

119.      % hObject      handle to pushbutton8 (see GCBO)
120.      % eventdata    reserved - to be defined in a future version of MATLAB
121.      % handles       structure with handles and user data (see GUIDATA)
122.      set(handles.pushbuttonknipdata,'ForegroundColor',[1.0 .0 .0]);
123.      set(handles.checkboxdata,'Value',0);
124.      set(handles.slider2,'Value',1);
125.      set(handles.text5,'String','1');
126.      set(handles.text20,'Visible','off');
127.      set(handles.textknipdata,'String','1');
128.      set(handles.textknipdata,'Visible','off');
129.      setappdata(gcf,'plotstatus1',0);
130.
131.      %uitlezen data en plotten
132.      [fileName pathName] = uigetfile('.xlsx','Selecteer de voetzool data');
133.      excelfile = fullfile(pathName, fileName);
134.      B = xlsread(excelfile);
135.      Voetzooldata = smoothdata(B); %smoothdata
136.
137.      Size = length(Voetzooldata);
138.      samples = 1:1:Size;
139.      tijd = samples/120;
140.
141.      setappdata(gcf,'Size',Size);
142.      setappdata(gcf,'tijd',tijd);
143.      set(handles.text4,'Visible','on');
144.      set(handles.text5,'Visible','on');
145.      set(handles.text6,'Visible','on');
146.      set(handles.text6,'String',[' / ',num2str(Size)]);
147.
148.      FSR1a = Voetzooldata(:,1);
149.      FSR2a = Voetzooldata(:,2);
150.      FSR3a = Voetzooldata(:,3);
151.      FSR4a = Voetzooldata(:,4);
152.
153.      %% normalisatie
154.      [fileName pathName] = uigetfile('.xlsx','Selecteer Normalisatie bestand');
155.      normalisatiefile = fullfile(pathName, fileName);
156.      VoetzooldataN = xlsread(normalisatiefile);
157.
158.      %Gemiddelde met marge van 15%
159.      normSensor1 = round(mean(VoetzooldataN(:,1)*(0.85)));
160.      normSensor2 = round(mean(VoetzooldataN(:,2)*(0.85)));
161.      normSensor3 = round(mean(VoetzooldataN(:,3)*(0.85)));
162.      normSensor4 = round(mean(VoetzooldataN(:,4)*(0.85)));
163.
164.      for i = 1:length(Voetzooldata)
165.          FSR1na(i) = FSR1a(i)-normSensor1;
166.          FSR2na(i) = FSR2a(i)-normSensor2;
167.          FSR3na(i) = FSR3a(i)-normSensor3;
168.          FSR4na(i) = FSR4a(i)-normSensor4;
169.      end
170.
171.      FSR1na = FSR1na(:);
172.      FSR2na = FSR2na(:);
173.      FSR3na = FSR3na(:);
174.      FSR4na = FSR4na(:);
175.
176.      %% Calibratie
177.      [fileName pathName] = uigetfile('.xlsx','Selecteer Calibratie bestand');

```



```

178.     calibratiefile = fullfile(pathName, fileName);
179.     Gewicht = xlsread(calibratiefile,1,'B4:B22');
180.
181.     VoltageData(:,1) = xlsread(calibratiefile,1,'C4:C22');
182.     VoltageData(:,2) = xlsread(calibratiefile,2,'C4:C22');
183.     VoltageData(:,3) = xlsread(calibratiefile,3,'C4:C22');
184.     VoltageData(:,4) = xlsread(calibratiefile,4,'C4:C22');
185.
186.     %interpoleren van data. Voltage van 0 naar 5,5kg met stappen van 100g
187.     xr = (0:1:5500) ;
188.     CaliVoltage(:, :) = interp1(Gewicht(:, :), VoltageData(:, :), xr);
189.     Gewichtinter = 0:1:5500;
190.
191.     %calibreren niet genormaliseerde data
192.     for i = 1:Size;
193.         waarde1 = FSR1a(i,1);
194.         difference1 = abs(CaliVoltage(:,1)-waarde1);
195.         [x(i) y(i)] = min(difference1);
196.         gevonden1(i) = CaliVoltage(y(i));
197.         FSR1b(i) = Gewichtinter(y(i));
198.     end
199.
200.     for i = 1:Size;
201.         waarde2 = FSR2a(i,1);
202.         difference2 = abs(CaliVoltage(:,2)-waarde2);
203.         [x(i) y(i)] = min(difference2);
204.         gevonden2(i) = CaliVoltage(y(i));
205.         FSR2b(i) = Gewichtinter(y(i));
206.     end
207.
208.     for i = 1:Size;
209.         waarde3 = FSR3a(i,1);
210.         difference3 = abs(CaliVoltage(:,3)-waarde3);
211.         [x(i) y(i)] = min(difference3);
212.         gevonden3(i) = CaliVoltage(y(i));
213.         FSR3b(i) = Gewichtinter(y(i));
214.     end
215.
216.     for i = 1:Size;
217.         waarde4 = FSR4a(i,1);
218.         difference4 = abs(CaliVoltage(:,4)-waarde4);
219.         [x(i) y(i)] = min(difference4);
220.         gevonden4(i) = CaliVoltage(y(i));
221.         FSR4b(i) = Gewichtinter(y(i));
222.     end
223.
224.     VoetzooldataB(:,1) = FSR1b(:);
225.     VoetzooldataB(:,2) = FSR2b(:);
226.     VoetzooldataB(:,3) = FSR3b(:);
227.     VoetzooldataB(:,4) = FSR4b(:);
228.
229.     LinksB(:,1) = FSR1b(:);
230.     LinksB(:,2) = FSR2b(:);
231.     RechtsB(:,1) = FSR3b(:);
232.     RechtsB(:,2) = FSR4b(:);
233.
234.     setappdata(gcf, 'Voetzooldata', VoetzooldataB);
235.     setappdata(gcf, 'Links', LinksB);
236.     setappdata(gcf, 'Rechts', RechtsB);

```

```
237.     setappdata(gcf,'Size',Size);
238.
239.     %Calibreren genormaliseerde data
240.     for i = 1:Size
241.         waarde1n = FSR1na(i,1);
242.         if waarde1n < 0
243.             FSR1nb(i) = (-100);
244.         else
245.             difference1 = abs(CaliVoltage(:,1)-waarde1n);
246.             [x(i) y(i)] = min(difference1);
247.             gevonden1n(i) = CaliVoltage(y(i));
248.             FSR1nb(i) = Gewichtinter(y(i));
249.         end
250.     end
251.
252.     for i = 1:Size
253.         waarde2n = FSR2na(i,1);
254.         if waarde2n < 0
255.             FSR2nb(i) = (-100);
256.         else
257.             difference2 = abs(CaliVoltage(:,2)-waarde2n);
258.             [x(i) y(i)] = min(difference2);
259.             gevonden2n(i) = CaliVoltage(y(i));
260.             FSR2nb(i) = Gewichtinter(y(i));
261.         end
262.     end
263.
264.     for i = 1:Size
265.         waarde3n = FSR3na(i,1);
266.         if waarde3n < 0
267.             FSR3nb(i) = (-100);
268.         else
269.             difference3 = abs(CaliVoltage(:,3)-waarde3n);
270.             [x(i) y(i)] = min(difference3);
271.             gevonden3n(i) = CaliVoltage(y(i));
272.             FSR3nb(i) = Gewichtinter(y(i));
273.         end
274.     end
275.
276.     for i = 1:Size
277.         waarde4n = FSR4na(i,1);
278.         if waarde4n < 0
279.             FSR4nb(i) = (-100);
280.         else
281.             difference4 = abs(CaliVoltage(:,4)-waarde4n);
282.             [x(i) y(i)] = min(difference4);
283.             gevonden4n(i) = CaliVoltage(y(i));
284.             FSR4nb(i) = Gewichtinter(y(i));
285.         end
286.     end
287.
288.     VoetzooldatanB(:,1) = FSR1nb(:);
289.     VoetzooldatanB(:,2) = FSR2nb(:);
290.     VoetzooldatanB(:,3) = FSR3nb(:);
291.     VoetzooldatanB(:,4) = FSR4nb(:);
292.
293.     LinksnB(:,1) = FSR1nb(:);
294.     LinksnB(:,2) = FSR2nb(:);
295.     RechtsnB(:,1) = FSR3nb(:);
```

```

296.     RechtsnB(:,2) = FSR4nb(:);
297.
298.     setappdata(gcf,'VoetzooldataN',VoetzooldatanB);
299.     setappdata(gcf,'LinksN',LinksnB);
300.     setappdata(gcf,'RechtsN',RechtsnB);
301.
302.     %% plot
303.     cla(handles.axes2);
304.     cla(handles.axes3);
305.     set(handles.axes2,'Visible','on');
306.     set(handles.axes3,'Visible','off');
307.     setappdata(gcf,'plotstatus1',Voetzooldata);
308.     set(handles.axes2,'Position',[0.564, 0.227, 0.353, 0.666]);
309.     plot(handles.axes2,VoetzooldataB(:,:));
310.     setappdata(gcf,'plot2visible',1);
311.
312.     title(handles.axes2,'Alle data');
313.     legend(handles.axes2,'L Sensor1','L Sensor2','R Sensor3','R Sensor4');
314.     legend(handles.axes2,'Orientation','vertical','Location','northwest');
315.     xlabel(handles.axes2,'Meetsample');
316.     ylabel(handles.axes2,'Gewicht[gram]');
317.     xlim(handles.axes2, [0 Size]);
318.     ylim(handles.axes2, 'auto');
319.     pan(handles.axes2, 'on');
320.
321.     %instellen data slider
322.     set(handles.slider2,'Min',1)
323.     set(handles.slider2,'Max',Size)
324.     set(handles.slider2,'Value',1)
325.     KleineStap = 1/(Size-1);
326.     GroteStap = 5/(Size-1);
327.     set(handles.slider2,'SliderStep',[KleineStap GroteStap]);

```

In de *Callback* van de videoslider wordt de code zo geprogrammeerd dat het framenummer, welke overeenkomt met de 'value' van de *slider1*, wordt getoond in *axes1*.

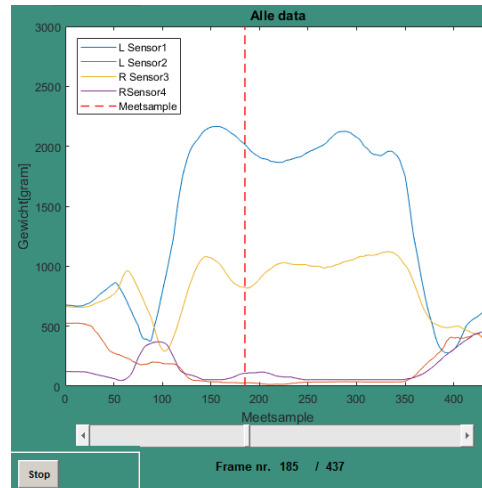
```

329.     %%-----Slider video-----
330.     % --- Executes on slider movement.
331.     function slider1_Callback(hObject, eventdata, handles)
332.     % hObject      handle to slider1 (see GCBO)
333.     % eventdata    reserved - to be defined in a future version of MATLAB
334.     % handles      structure with handles and user data (see GUIDATA)
335.
336.     % Hints: get(hObject,'Value') returns position of slider
337.     %          get(hObject,'Min') and get(hObject,'Max') to determine range of slid
er
338.     bnr = get(hObject,'value'); %%bnr=beeldnummer
339.     Naam = getappdata(gcf,'FilmNaam');
340.     VideoObject = VideoReader(Naam);
341.     VideoFrames = read(VideoObject);
342.     for k = bnr:bnr
343.         set(handles.text2,'String',num2str(k));
344.         set(handles.slider1,'Value',(k));
345.         axes(handles.axes1);
346.         frame = read(VideoObject,k);
347.         imshow(frame);
348.         drawnow;
349.         axis(handles.axes1,'off');
350.     end

```



Dit zelfde gebeurd ook in de *Callback* van de data slider, *slider2*. In deze *Callback* wordt er gecheckt welke data de gebruiker wenst te zien welke wordt opgeroepen via *plotstatus1* en *plotstatus2*. *plotstatus2* en *axes3* worden alleen opgeroepen als de gebruiker zowel *checkboxboxlinks* & *checkboxboxrechts* zijn aangevinkt. Bij het gebruiken van *slider2* wordt de gewenste data geplot en is er in de grafiek een lijn zichtbaar welke correspondeert met het framenummer/waarde van de slider.



```

352. % --- -----Slider DATA-----
353. function slider2_Callback(hObject, eventdata, handles)
354. % hObject    handle to slider2 (see GCBO)
355. % eventdata  reserved - to be defined in a future version of MATLAB
356. % handles    structure with handles and user data (see GUIDATA)
357.
358. % Hints: get(hObject, 'Value') returns position of slider
359. %         get(hObject, 'Min') and get(hObject, 'Max') to determine range
        of slider
360. plotslider = get(hObject, 'value');
361. set(handles.text5, 'String', num2str(plotslider));
362. plot2visible = getappdata(gcf, 'plot2visible');
363. plot3visible = getappdata(gcf, 'plot3visible');
364. plotstatus1 = getappdata(gcf, 'plotstatus1');
365. plotstatus2 = getappdata(gcf, 'plotstatus2');
366.
367. Voetzooldata = getappdata(gcf, 'Voetzooldata');
368. Links = getappdata(gcf, 'Links');
369. Rechts = getappdata(gcf, 'Rechts');
370. Size = getappdata(gcf, 'Size');
371. tijd = getappdata(gcf, 'tijd');
372. Knipdatastatus = getappdata(gcf, 'Knipdatastatus');
373. Knipdatastart = getappdata(gcf, 'Knipdatastart');
374.
375. checkboxlinks = get(handles.checkboxlinks, 'Value');
376. checkboxrechts = get(handles.checkboxrechts, 'Value');
377. checkboxnorm = get(handles.checkboxnorm, 'Value');
378.
379. bnr = 1;
380. k = bnr:1:Size;
381. b = get(handles.slider2, 'Value');
382.

```

```

383.     if plot2visible == 1
384.         set(handles.axes3,'Visible','off');
385.         cla(handles.axes2);
386.         b = plotslider;
387.         set(handles.text5,'String',num2str(b));
388.         plot(handles.axes2,plotstatus1(k,:));
389.         hold(handles.axes2,'on');
390.         plot(handles.axes2,[b b],[0 3000], 'LineWidth',1,'Color','r','Line
Style','--');
391.         xlabel(handles.axes2,'Meetsample');
392.         ylabel(handles.axes2,'Gewicht[gram]');
393.         xlim(handles.axes2, [0 Size]);
394.         ylim(handles.axes2, 'auto');
395.         pan(handles.axes2, 'on');
396.
397.         if checkboxlinks == 1
398.             elseif checkboxrechts == 1
399.                 legend(handles.axes2,'Sensor1','Sensor2','Meetsample');
400.                 legend(handles.axes2,'Orientation','vertical','Location','northwes
t');
401.             else
402.                 legend(handles.axes2,'L Sensor1','L Sensor2','R Sensor3','RSensor4
','Meetsample');
403.                 legend(handles.axes2,'Orientation','vertical','Location','northwes
t');
404.             end
405.         end
406.
407.     if plot3visible == 1
408.         set(handles.axes2,'Visible','on');
409.         set(handles.axes3,'Visible','on');
410.         cla(handles.axes2);
411.         cla(handles.axes3);
412.         plot(handles.axes2, plotstatus1(k,:));
413.         plot(handles.axes3, plotstatus2(k,:));
414.         hold(handles.axes2,'on');
415.         hold(handles.axes3,'on');
416.         b = plotslider;
417.         set(handles.text5,'String',num2str(b));
418.         plot(handles.axes2, [b b], [0 3000],'LineWidth',1,'Color','r','Lin
eStyle','--');
419.         plot(handles.axes3, [b b], [0 3000],'LineWidth',1,'Color','r','Lin
eStyle','--');
420.         hold(handles.axes2,'on');
421.         hold(handles.axes3,'on');
422.         legend(handles.axes2,'Sensor1 L','Sensor2 L','Meetsample');
423.         legend(handles.axes2,'Orientation','horizontal','Location','north'
);
424.         legend(handles.axes3,'Sensor1 R','Sensor2 R','Meetsample');
425.         legend(handles.axes3,'Orientation','horizontal','Location','north'
);
426.         xlabel(handles.axes2,'Meetsample');
427.         ylabel(handles.axes2,'Gewicht[gram]');
428.         xlim(handles.axes2, [0 Size]);
429.         ylim(handles.axes2, 'auto');
430.         pan(handles.axes2, 'on');
431.         xlabel(handles.axes3,'Meetsample');
432.         ylabel(handles.axes3,'Gewicht[gram]');
433.         xlim(handles.axes3, [0 Size]);

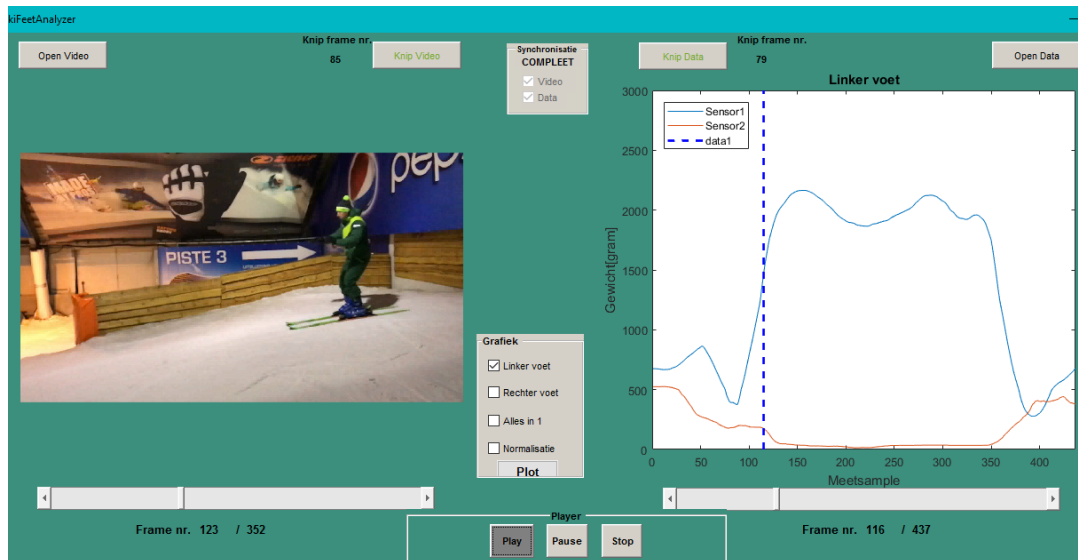
```

```
434.         ylim(handles.axes3, 'auto');
435.         pan(handles.axes3, 'on');
436.     end
```

In regel 438 t/m 458 zijn de *CreateFcn* van *slider1* en *slider2* weergegeven. Deze dienen in het script aanwezig te zijn maar hier wordt niks mee gedaan. Zij hebben geen functie in het script.

```
438.     % --- Executes during object creation, after setting all properties.
439.     function slider2_CreateFcn(hObject, eventdata, handles)
440.     % hObject    handle to slider2 (see GCBO)
441.     % eventdata  reserved - to be defined in a future version of MATLAB
442.     % handles    empty - handles not created until after all CreateFcns called
443.
444.     % Hint: slider controls usually have a light gray background.
445.     if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundC
olor'))
446.         set(hObject,'BackgroundColor',[.9 .9 .9]);
447.     end
448.
449.     % --- Executes during object creation, after setting all properties.
450.     function slider1_CreateFcn(hObject, eventdata, handles)
451.     % hObject    handle to slider1 (see GCBO)
452.     % eventdata  reserved - to be defined in a future version of MATLAB
453.     % handles    empty - handles not created until after all CreateFcns called
454.
455.     % Hint: slider controls usually have a light gray background.
456.     if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundC
olor'))
457.         set(hObject,'BackgroundColor',[.9 .9 .9]);
458.     end
```

In de *Callback* van de *pushbuttonplay* wordt het mogelijk gemaakt om de video en de data frame voor frame af te spelen. Wanneer de gebruiker nog niet heeft gesynchroniseerd werkt de *pushbuttonplay* enkel met de video. Zodra de synchronisatie compleet is, wordt ook de datagrafiëk hiermee bediend. Wat belangrijk is in dit script is dat er gecontroleerd wordt of de pauze- of stop button worden ingedrukt. Dit zorgt ervoor dat de *Callback* van de *pushbuttonplay* niet meer worden uitgevoerd. Wederom wordt gecheckt welke data de gebruiker wilt zien. Tevens wordt gecheckt of de gebruiker al een synchronisatie moment heeft bepaald voor de video en/of data. Zodra de gebruiker dit heeft gedaan, is het eerste framenummer het framenummer van het synchronisatiemoment. Als de synchronisatie compleet is en de gebruiker de *pushbuttonplay* indrukt, is er in de datagrafiëk een blauwe stippellijn zicht, dit is de afspeellijn.



Wat er ook veranderd zodra de *pushbuttonplay* wordt ingedrukt, is de kleur van de button. Zodra de play-pauze- of stopbutton wordt ingedrukt, wordt deze donkergrijs. De andere twee buttons worden lichtgrijs, de gebruiker ziet nu duidelijk welke button actief is.

```

460.      %%-----PLAY-----
461.      % --- Executes on button press in pushbuttonplay.
462.      function pushbuttonplay_Callback(hObject, eventdata, handles)
463.      % hObject      handle to pushbuttonplay (see GCBO)
464.      % eventdata    reserved - to be defined in a future version of MATLAB
465.      % handles      structure with handles and user data (see GUIDATA)
466.      set(hObject,'BackgroundColor',[.502 .502 .502]);
467.      set(handles.pushbuttonstop,'BackgroundColor',[.831 .816 .784]);
468.      set(handles.pushbuttonpauze,'BackgroundColor',[.831 .816 .784]);
469.
470.      setappdata(gcf,'pauze',0);
471.      pauzeVisited = getappdata(gcf,'pauze');
472.      setappdata(gcf,'stop',0);
473.      stopVisited = getappdata(gcf,'stop');
474.
475.      plot2visible = getappdata(gcf,'plot2visible');
476.      plot3visible = getappdata(gcf,'plot3visible');
477.      plotstatus1 = getappdata(gcf,'plotstatus1');
478.      plotstatus2 = getappdata(gcf,'plotstatus2');
479.      checkboxlinks = get(handles.checkboxlinks,'Value');
480.      checkboxrechts = get(handles.checkboxrechts,'Value');
481.
482.      Synchroonstatus = getappdata(gcf,'Synchroonstatus');
483.      Knipfilmstatus = getappdata(gcf,'Knipfilmstatus');
484.      Knipfilmstart = getappdata(gcf,'Knipfilmstart');
485.      Knipdatastatus = getappdata(gcf,'Knipdatastatus');
486.      Knipdatastart = getappdata(gcf,'Knipdatastart');
487.
488.      if Synchroonstatus == 1
489.          Synchroonstatus = 1;
490.      else
491.          Synchroonstatus = 0;
492.      end
493.
494.      bnrfilm = get(handles.text2,'String');
495.      bnrfilm = str2num(bnrfilm);
496.

```

```
497.     Naam = getappdata(gcf, 'FilmNaam');
498.     VideoObject = VideoReader(Naam);
499.     numFrames = (VideoObject.NumberOfFrames);
500.
501.     if Synchroonstatus == 0
502.         for frameCount = bnrfilm:numFrames
503.             setappdata(gcf, 'frameCount', frameCount)
504.
505.             if pauzeVisited == 0 && stopVisited == 0
506.                 set(handles.text2, 'String', num2str(frameCount));
507.                 set(handles.slider1, 'Value', (frameCount));
508.                 axes(handles.axes1);
509.                 frame = read(VideoObject, frameCount);
510.                 imshow(frame);
511.                 drawnow;
512.                 axis(handles.axes1, 'off');
513.                 pauzeVisited = getappdata(gcf, 'pauze');
514.                 stopVisited = getappdata(gcf, 'stop');
515.             end
516.         end
517.
518.         if bnrfilm == numFrames
519.             set(handles.text2, 'String', num2str(1));
520.             set(handles.slider1, 'Value', (1));
521.         end
522.     end
523.
524.     if Synchroonstatus == 1
525.         for frameCount = bnrfilm:numFrames
526.             setappdata(gcf, 'frameCount', frameCount)
527.
528.             difference = (Knipfilmstart - Knipdatastart);
529.             setappdata(gcf, 'difference', difference);
530.
531.             if pauzeVisited == 0 && stopVisited == 0
532.                 set(handles.text2, 'String', num2str(frameCount));
533.                 set(handles.slider1, 'Value', (frameCount));
534.                 axes(handles.axes1);
535.                 frame = read(VideoObject, frameCount);
536.                 imshow(frame);
537.                 drawnow;
538.                 axis(handles.axes1, 'off');
539.                 axes(handles.axes2);
540.                 plotcursor
541.                 lijndata = getappdata(gcf, 'lijndata');
542.                 set(handles.text5, 'String', num2str(lijndata));
543.                 set(handles.slider2, 'Value', (lijndata));
544.             end
545.
546.             if plot3visible == 1;
547.                 axes(handles.axes3);
548.                 plotcursor
549.             end
550.
551.             if bnrfilm == numFrames
552.                 set(handles.text2, 'String', num2str(1));
553.                 set(handles.slider1, 'Value', (1));
554.             end
555.
```



```

556.         if checkboxlinks == 1
557.             elseif checkboxrechts == 1
558.                 legend(handles.axes2, 'Sensor1', 'Sensor2', 'Afspeellijn');
559.                 legend(handles.axes2, 'Orientation', 'vertical', 'Location', 'nort
hwest');
560.             else
561.                 legend(handles.axes2, 'L Sensor1', 'L Sensor2', 'R Sensor3', 'R Se
nsor4', 'Afspeellijn');
562.                 legend(handles.axes2, 'Orientation', 'vertical', 'Location', 'nort
hwest');
563.             end
564.             pauzeVisited = getappdata(gcf, 'pauze');
565.             stopVisited = getappdata(gcf, 'stop');
566.         end
567.     end
568.
569.     function plotcursor()
570.         oldcursor = getappdata(gcf, 'oldcursor');
571.         frameCount = getappdata(gcf, 'frameCount');
572.         difference = getappdata(gcf, 'difference');
573.         lijndata = abs(frameCount-difference);
574.         cursor = line([lijndata lijndata], [0 3000], 'LineWidth', 2, 'Color', 'b', 'Lin
eStyle', '--');
575.         nocursor = isempty(oldcursor);
576.         if nocursor == 0
577.             delete(oldcursor)
578.         end
579.         setappdata(gcf, 'oldcursor', cursor)
580.         setappdata(gcf, 'lijndata', lijndata);

```

In regel 582 t/m 615 wordt de *Callback* van de pauzeknop uitgevoerd. *PauzeVisited* krijgt bij het indrukken waarde 1. Dit wordt door de *pushbuttonplay* gecontroleerd welke ervoor zorgt dat de *Callback* van de *pushbuttonplay* wordt gestaakt. Zodra de gebruiker op de *pushbuttonpauze* drukt, wordt de waarde van *text2* opgevraagd. Deze waarde correspondeert met het frame welke op dat moment wordt afgespeeld. Het afspelen wordt gestaakt en enkel framenummer van *text2* wordt getoond.

```

582.%-----Pauze-----
583.%--- Executes on button press in pushbuttonpauze.
584.function pushbuttonpauze_Callback(hObject, eventdata, handles)
585.% hObject    handle to pushbuttonpauze (see GCBO)
586.% eventdata reserved - to be defined in a future version of MATLAB
587.% handles    structure with handles and user data (see GUIDATA)
588.set(hObject, 'BackgroundColor', [.502 .502 .502]);
589.set(handles.pushbuttonplay, 'BackgroundColor', [.831 .816 .784]);
590.set(handles.pushbuttonstop, 'BackgroundColor', [.831 .816 .784]);
591.
592.
593.pauzeVisited = getappdata(gcf, 'pauze');
594.if pauzeVisited ==1
595.setappdata(gcf, 'pauze', 0)
596.
597.else
598.setappdata(gcf, 'pauze', 1)
599.end
600.
601.stopVisited = getappdata(gcf, 'stop');
602.setappdata(gcf, 'stop', 0);
603.
604.Naam = getappdata(gcf, 'FilmNaam');
605.VideoObject = VideoReader(Naam);

```



```
606.numFrames = (VideoObject.NumberOfFrames);
607.bnr = get(handles.text2, 'String');
608.bnr = str2num(bnr);
609.
610.if pauzeVisited == 1 && stopVisited == 0;
611.axes(handles.axes1);
612.image(bnr);
613.drawnow;
614.axis(handles.axes1, 'off');
615.end
```

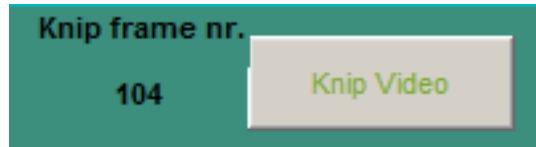
De *Callback* van de *pushbuttonplay* zorgt ervoor dat het afspelen wordt gestaakt en de video en data worden teruggezet naar de beginpositie. Dit is voor het synchroniseren frame 1 van de video. Na het synchroniseren zijn dit de synchronisatiemomenten *Knipfilmstart* en *Knipdatastart*. Niet enkel de frame weergave maar ook *text2* en *text5*, en beide sliders worden teruggezet naar deze beginwaarde.

```
617.      %%-----STOP-----
618.      % --- Executes on button press in pushbuttonstop.
619.      function pushbuttonstop_Callback(hObject, eventdata, handles)
620.      % hObject      handle to pushbuttonstop (see GCBO)
621.      % eventdata    reserved - to be defined in a future version of MATLAB
622.      % handles      structure with handles and user data (see GUIDATA)
623.      set(hObject, 'BackgroundColor', [.502 .502 .502]);
624.      set(handles.pushbuttonplay, 'BackgroundColor', [.831 .816 .784]);
625.      set(handles.pushbuttonpauze, 'BackgroundColor', [.831 .816 .784]);
626.
627.      stopVisited = getappdata(gcf, 'stop');
628.      if stopVisited == 1
629.          setappdata(gcf, 'stop', 0)
630.
631.      else
632.          setappdata(gcf, 'stop', 1)
633.      end
634.
635.      pauzeVisited = getappdata(gcf, 'pauze');
636.      if pauzeVisited == 1
637.          setappdata(gcf, 'pauze', 0)
638.      else
639.          setappdata(gcf, 'pauze', 1)
640.      end
641.
642.      synchroonstatus = getappdata(gcf, 'synchroonstatus');
643.      Knipfilmstatus = getappdata(gcf, 'Knipfilmstatus');
644.      Knipfilmstart = getappdata(gcf, 'Knipfilmstart');
645.
646.      if Knipfilmstatus == 1
647.          bnr = Knipfilmstart;
648.      else
649.          bnr = 1;
650.      end
651.
652.      set(handles.text2, 'String', (num2str(bnr)));
653.      set(handles.slider1, 'Value', bnr);
654.      Naam = getappdata(gcf, 'FilmNaam');
655.      VideoObject = VideoReader(Naam);
656.      axes(handles.axes1);
657.      axis(handles.axes1, 'off');
```

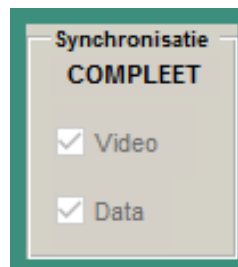


```
658.     frame1 = read(VideoObject, bnr);
659.     imshow(frame1);
660.     drawnow;
```

In de *Callback* van de *pushbuttonknip* video en *pushbuttonknipdata* worden het framenummers van de synchronisatie momenten bepaald en worden *checkboxboxvideo* en *checkboxboxdata* aangevinkt. Het is voor de gebruiker niet mogelijk om deze checkboxes zelf aan te klikken. Zodra een van de twee wordt geactiveerd, wordt het framenummer zichtbaar en veranderen de letters van de button naar het groen.



Zodra beide synchronisatiemomenten zijn bepaald wordt zichtbaar voor de gebruiker dat de synchronisatie compleet is.



```
662.     %%-----KNIP Video-----
663.     % --- Executes on button press in pushbuttonknipvideo.
664.     function pushbuttonknipvideo_Callback(hObject, eventdata, handles)
665.     % hObject      handle to pushbuttonknipvideo (see GCBO)
666.     % eventdata    reserved - to be defined in a future version of MATLAB
667.     % handles      structure with handles and user data (see GUIDATA)
668.     set(hObject, 'ForegroundColor', [.47 .67 .19]);
669.     set(handles.checkboxvideo, 'Value', 1);
670.     synchroonstatus = 0;
671.
672.     Knipfilmstatus = 1;
673.     Knipfilm = get(handles.text2, 'String');
674.     Knipfilm = str2num(Knipfilm);
675.     set(handles.text19, 'Visible', 'on');
676.     set(handles.textknipfilm, 'Visible', 'on');
677.     set(handles.textknipfilm, 'String', num2str(Knipfilm));
678.
679.     checkboxstatus1 = get(handles.checkboxvideo, 'Value');
680.     checkboxstatus2 = get(handles.checkboxdata, 'Value');
681.     if (checkboxstatus1 == 1) && (checkboxstatus2 == 1)
682.         set(handles.text9, 'String', 'COMPLEET');
683.         synchroonstatus = 1;
684.     else
685.         set(handles.text9, 'String', 'incompleet');
686.     end
687.
688.     setappdata(gcf, 'Synchroonstatus', synchroonstatus);
689.     setappdata(gcf, 'Knipfilmstart', Knipfilm);
690.     setappdata(gcf, 'Knipfilmstatus', Knipfilmstatus);
691.
692.     %%-----KNIP Data-----
693.     % --- Executes on button press in pushbuttonknipdata.
```



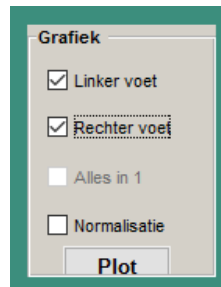
```
694.     function pushbuttonknipdata_Callback(hObject, eventdata, handles)
695.     % hObject      handle to pushbuttonknipdata (see GCBO)
696.     % eventdata    reserved - to be defined in a future version of MATLAB
697.     % handles      structure with handles and user data (see GUIDATA)
698.     set(hObject,'ForegroundColor',[.47 .67 .19]);
699.     set(handles.checkboxdata,'Value',1);
700.     Synchronoonstatus = 0;
701.
702.     Knipdatastatus = 1;
703.     setappdata(gcf,'Knipdatastatus',Knipdatastatus);
704.     Knipdatastart = get(handles.text5,'String');
705.     Knipdatastart = str2num(Knipdatastart);
706.     set(handles.text20,'Visible','on');
707.     set(handles.textknipdata,'Visible','on');
708.     set(handles.textknipdata,'String',num2str(Knipdatastart));
709.
710.     setappdata(gcf,'Knipdatastart',Knipdatastart);
711.
712.     checkboxstatus1 = get(handles.checkboxvideo,'Value');
713.     checkboxstatus2 = get(handles.checkboxdata,'Value');
714.     if (checkboxstatus1 == 1) && (checkboxstatus2 == 1)
715.         set(handles.text9,'String','COMPLEET');
716.         Synchronoonstatus = 1;
717.     else
718.         set(handles.text9,'String','incompleet');
719.     end
720.
721.     setappdata(gcf,'Synchronoonstatus',Synchronoonstatus);
```

Zoals gemeld kan de gebruiker checkboxvideo en checkboxdata niet zelfstandig aanklikken. Deze Callbacks zijn dan ook leeg.

```
723.     % --- Executes on button press in checkbox3.
724.     function checkboxvideo_Callback(hObject, eventdata, handles)
725.     % hObject      handle to checkbox3 (see GCBO)
726.     % eventdata    reserved - to be defined in a future version of MATLAB
727.     % handles      structure with handles and user data (see GUIDATA)
728.
729.     % Hint: get(hObject,'Value') returns toggle state of checkbox3
730.
731.     % --- Executes on button press in checkbox3.
732.     function checkboxdata_Callback(hObject, eventdata, handles)
733.     % hObject      handle to checkbox3 (see GCBO)
734.     % eventdata    reserved - to be defined in a future version of MATLAB
735.     % handles      structure with handles and user data (see GUIDATA)
736.
737.     % Hint: get(hObject,'Value') returns toggle state of checkbox3
```

In regel 739 t/m 808 worden de *checkboxlinks*, *checkboxsrechts*, *checkboxbeide* en *checkboxnorm* ingesteld. Met deze functies kan de gebruiker aanklikken welke grafieken hij wilt zien. De *Callbacks* zijn zo geprogrammeerd zodat de gebruiker *checkboxbeide* niet kan aanvinken zodat *checkboxlinks* en *checkboxrechts* zijn aangevinkt. Andersom geldt dit ook.





```
739. % --- Executes on button press in checkboxlinks.
740. function checkboxlinks_Callback(hObject, eventdata, handles)
741. % hObject    handle to checkboxlinks (see GCBO)
742. % eventdata  reserved - to be defined in a future version of MATLAB
743. % handles    structure with handles and user data (see GUIDATA)
744.
745. % Hint: get(hObject,'Value') returns toggle state of checkboxlinks
746. checkboxlinks = get(handles.checkboxlinks,'Value');
747. checkboxrechts = get(handles.checkboxrechts,'Value');
748.
749. if (checkboxlinks == 1) && (checkboxrechts == 1)
750.     set(handles.checkboxbeide,'Enable','off');
751.     set(handles.checkboxbeide,'Value',0);
752. end
753.
754. if (checkboxlinks == 0) && (checkboxrechts == 0)
755.     set(handles.checkboxbeide,'Enable','on');
756.     set(handles.checkboxbeide,'Value',0);
757. end
758.
759. % --- Executes on button press in checkboxrechts.
760. function checkboxrechts_Callback(hObject, eventdata, handles)
761. % hObject    handle to checkboxrechts (see GCBO)
762. % eventdata  reserved - to be defined in a future version of MATLAB
763. % handles    structure with handles and user data (see GUIDATA)
764.
765. % Hint: get(hObject,'Value') returns toggle state of checkboxrechts
766. checkboxlinks = get(handles.checkboxlinks,'Value');
767. checkboxrechts = get(handles.checkboxrechts,'Value');
768.
769. if (checkboxlinks == 1) && (checkboxrechts == 1)
770.     set(handles.checkboxbeide,'Enable','off');
771.     set(handles.checkboxbeide,'Value',0);
772. end
773.
774. if (checkboxlinks == 0) && (checkboxrechts == 0)
775.     set(handles.checkboxbeide,'Enable','on');
776.     set(handles.checkboxbeide,'Value',0);
777. end
778.
779. % --- Executes on button press in checkboxbeide.
780. function checkboxbeide_Callback(hObject, eventdata, handles)
781. % hObject    handle to checkboxbeide (see GCBO)
782. % eventdata  reserved - to be defined in a future version of MATLAB
783. % handles    structure with handles and user data (see GUIDATA)
784.
785. % Hint: get(hObject,'Value') returns toggle state of checkboxbeide
786. checkboxbeide = get(handles.checkboxbeide,'Value');
787.
```



```

788.     if checkboxbeide == 1
789.         set(handles.checkboxlinks,'Enable','off');
790.         set(handles.checkboxlinks,'Value',0);
791.         set(handles.checkboxrechts,'Enable','off');
792.         set(handles.checkboxrechts,'Value',0);
793.     end
794.
795.     if checkboxbeide == 0
796.         set(handles.checkboxlinks,'Enable','on');
797.         set(handles.checkboxlinks,'Value',0);
798.         set(handles.checkboxrechts,'Enable','on');
799.         set(handles.checkboxrechts,'Value',0);
800.     end
801.
802.     % --- Executes on button press in checkboxnorm.
803.     function checkboxnorm_Callback(hObject, eventdata, handles)
804.     % hObject      handle to checkboxnorm (see GCBO)
805.     % eventdata    reserved - to be defined in a future version of MATLAB
806.     % handles      structure with handles and user data (see GUIDATA)
807.
808.     % Hint: get(hObject,'Value') returns toggle state of checkboxnorm

```

Zodra de gebruiker heeft aangevinkt welke data hij wilt zien, wordt deze weergegeven zodra de gebruiker op *pushbuttonplot* drukt. Hierin wordt gecheckt welke checkboxes zijn aangevinkt en welke data hierbij hoort. Met deze informatie kan wordt bepaald wat de inhoud is van *plotstatus1* en/of *plotstatus2* en welke grafiek zichtbaar moet zijn; *plot2visible* en/of *plot3visible*. *Axes3* wordt enkel gebruikt wanneer de gebruiker *checkboxlinks* en *checkboxrechts* heeft aangeklikt. Er worden dan twee subplots getoond.

```

810.     %-----PLOT-----
811.     % --- Executes on button press in pushbuttonplot.
812.     function pushbuttonplot_Callback(hObject, eventdata, handles)
813.     % hObject      handle to pushbuttonplot (see GCBO)
814.     % eventdata    reserved - to be defined in a future version of MATLAB
815.     % handles      structure with handles and user data (see GUIDATA)
816.     checkboxlinks = get(handles.checkboxlinks,'Value');
817.     checkboxrechts = get(handles.checkboxrechts,'Value');
818.     checkboxbeide = get(handles.checkboxbeide,'Value');
819.     checkboxnorm = get(handles.checkboxnorm,'Value');
820.
821.
822.     Size = getappdata(gcf,'Size');
823.     Duur = (Size/120);
824.     tijd = getappdata(gcf,'tijd');
825.     xas = Size/3;
826.
827.     Voetzooldata = getappdata(gcf,'Voetzooldata');
828.     Links = getappdata(gcf,'Links');
829.     Rechts = getappdata(gcf,'Rechts');
830.     VoetzooldataN = getappdata(gcf,'VoetzooldataN');
831.     LinksN = getappdata(gcf,'LinksN');
832.     RechtsN = getappdata(gcf,'RechtsN');
833.
834.     Synchroonstatus = getappdata(gcf,'Synchroonstatus');
835.     Knipdatastatus = getappdata(gcf,'Knipdatastatus');
836.     Knipdatastart = getappdata(gcf,'Knipdatastart');
837.
838.     plot2visible = 0;

```



```

839.     plot3visible = 0;
840.     setappdata(gcf,'plot2visible',plot2visible);
841.     setappdata(gcf,'plot3visible',plot2visible);
842.     setappdata(gcf,'plotstatus1',0);
843.     setappdata(gcf,'plotstatus2',0);
844.
845.     bnr = 1;
846.     k = bnr:1:Size;
847.
848.     %%alleen links
849.     if checkboxlinks == 1 && checkboxrechts == 0 && checkboxbeide == 0 && checkbo
        xnorm == 0
850.         cla(handles.axes2);
851.         cla(handles.axes3);
852.         set(handles.slider2,'Value',bnr);
853.         set(handles.text5,'String',num2str(bnr));
854.         setappdata(gcf,'plotstatus1',Links);
855.         set(handles.axes2,'Visible','on');
856.         setappdata(gcf,'plot2visible',1);
857.         set(handles.axes3,'Visible','off');
858.         plot(handles.axes2,Links(k,:));
859.         hold(handles.axes2,'on');
860.         set(handles.axes2,'Position',[0.564, 0.227, 0.353, 0.666]);
861.         title(handles.axes2,'Linker voet');
862.         legend(handles.axes2,'Sensor1','Sensor2','Meetsample');
863.         legend(handles.axes2,'Orientation','vertical','Location','northwest');
864.         xlabel(handles.axes2,'Meetsample');
865.         ylabel(handles.axes2,'Gewicht[gram]');
866.         xlim(handles.axes2, [0 Size]);
867.         ylim(handles.axes2, 'auto');
868.         pan(handles.axes2, 'on');
869.     end
870.
871.     %%alleen rechts
872.     if checkboxlinks == 0 && checkboxrechts == 1 && checkboxbeide == 0 && checkbo
        xnorm == 0
873.         cla(handles.axes2);
874.         cla(handles.axes3);
875.         set(handles.slider2,'Value',bnr);
876.         set(handles.text5,'String',num2str(bnr));
877.         setappdata(gcf,'plotstatus1',Rechts);
878.         set(handles.axes2,'Visible','on');
879.         setappdata(gcf,'plot2visible',1);
880.         set(handles.axes3,'Visible','off');
881.         plot(handles.axes2,Rechts(k,:));
882.         hold(handles.axes2,'on');
883.         set(handles.axes2,'Position',[0.564, 0.227, 0.353, 0.666]);
884.         title(handles.axes2,'Rechter voet');
885.         legend(handles.axes2,'Sensor1','Sensor2');
886.         legend(handles.axes2,'Orientation','vertical','Location','northwest');
887.         xlabel(handles.axes2,'Meetsample');
888.         ylabel(handles.axes2,'Gewicht[gram]');
889.         xlim(handles.axes2, [0 Size]);
890.         ylim(handles.axes2, 'auto');
891.         pan(handles.axes2, 'on');
892.     end
893.
894.     %%beide groot
    
```



```

895.     if checkboxlinks == 0 && checkboxrechts == 0 && checkboxbeide == 1 && checkbo
        xnorm == 0
896.         cla(handles.axes2);
897.         cla(handles.axes3);
898.         set(handles.slider2,'Value',bnr);
899.         set(handles.text5,'String',num2str(bnr));
900.         set(handles.axes2,'Visible','on');
901.         setappdata(gcf,'plot2visible',1);
902.         setappdata(gcf,'plotstatus1',Voetzooldata);
903.         set(handles.axes3,'Visible','off');
904.         plot(handles.axes2,Voetzooldata(k,:));
905.         hold(handles.axes2,'on');
906.         set(handles.axes2,'Position',[0.564, 0.227, 0.353, 0.666]);
907.         title(handles.axes2,'Alle data');
908.         legend(handles.axes2,'L Sensor1','L Sensor2','R Sensor3','R Sensor4');
909.         legend(handles.axes2,'Orientation','vertical','Location','northwest');
910.         xlabel(handles.axes2,'Meetsample');
911.         ylabel(handles.axes2,'Gewicht[gram]');
912.         xlim(handles.axes2, [0 Size]);
913.         ylim(handles.axes2, 'auto');
914.         pan(handles.axes2, 'on');
915.     end
916.
917.     %%beide klein
918.     if checkboxlinks == 1 && checkboxrechts == 1 && checkboxbeide == 0 && checkbo
        xnorm == 0
919.         cla(handles.axes2);
920.         cla(handles.axes3);
921.         set(handles.slider2,'Value',bnr);
922.         set(handles.text5,'String',num2str(bnr));
923.         set(handles.axes2,'Visible','on');
924.         setappdata(gcf,'plot2visible',1);
925.         setappdata(gcf,'plotstatus1',Links);
926.         plot(handles.axes2,Links(k,:));
927.         hold(handles.axes2,'on');
928.         set(handles.axes2,'Position',[.552, .562, .37, .354]);
929.         title(handles.axes2,'Alle data');
930.         legend(handles.axes2,'Sensor1 L','Sensor2 L');
931.         legend(handles.axes2,'Orientation','horizontal','Location','north');
932.         xlabel(handles.axes2,'Meetsample','FontSize',8);
933.         ylabel(handles.axes2,'Gewicht[gram]','FontSize',8);
934.         xlim(handles.axes2, [0 Size]);
935.         ylim(handles.axes2, 'auto');
936.         pan(handles.axes2, 'on');
937.
938.         set(handles.axes3,'Visible','on');
939.         setappdata(gcf,'plot3visible',1);
940.         setappdata(gcf,'plotstatus2',Rechts);
941.         plot(handles.axes3,Rechts(k,:));
942.         hold(handles.axes3,'on');
943.         set(handles.axes3,'Position',[.553, .206, .37, .329]);
944.         legend(handles.axes3,'Sensor1 R','Sensor2 R');
945.         legend(handles.axes3,'Orientation','horizontal','Location','north');
946.         title(handles.axes3,' ');
947.         xlabel(handles.axes3,'Meetsample','FontSize',8);
948.         ylabel(handles.axes3,'Gewicht[gram]','FontSize',8);
949.         xlim(handles.axes3, [0 Size]);
950.         ylim(handles.axes3, 'auto');
951.         pan(handles.axes3, 'on');

```




```

952.         end
953.
954.         %%linksnorm
955.         if checkboxlinks == 1 && checkboxrechts == 0 && checkboxbeide == 0 && checkbo
           xnorm == 1
956.             cla(handles.axes2);
957.             cla(handles.axes3);
958.             set(handles.slider2,'Value',bnr);
959.             set(handles.text5,'String',num2str(bnr));
960.             set(handles.axes2,'Visible','on');
961.             setappdata(gcf,'plot2visible',1);
962.             setappdata(gcf,'plotstatus1',LinksN);
963.             set(handles.axes3,'Visible','off');
964.             plot(handles.axes2,LinksN(k,:));
965.             hold(handles.axes2,'on');
966.             set(handles.axes2,'Position',[0.564, 0.227, 0.353, 0.666]);
967.             title(handles.axes2,'Links, genormaliseerd');
968.             legend(handles.axes2,'Sensor1','Sensor2');
969.             legend(handles.axes2,'Orientation','vertical','Location','northwest');
970.             xlabel(handles.axes2,'Meetsample');
971.             ylabel(handles.axes2,'Gewicht[gram]');
972.             xlim(handles.axes2, [0 Size]);
973.             ylim(handles.axes2, 'auto');
974.             pan(handles.axes2, 'on');
975.         end
976.
977.         %%rechtsnorm
978.         if checkboxlinks == 0 && checkboxrechts == 1 && checkboxbeide == 0 && checkbo
           xnorm == 1
979.             cla(handles.axes2);
980.             cla(handles.axes3);
981.             set(handles.slider2,'Value',bnr);
982.             set(handles.text5,'String',num2str(bnr));
983.             set(handles.axes2,'Visible','on');
984.             setappdata(gcf,'plot2visible',1);
985.             setappdata(gcf,'plotstatus1',RechtsN);
986.             set(handles.axes3,'Visible','off');
987.             plot(handles.axes2,RechtsN(k,:));
988.             hold(handles.axes2,'on');
989.             set(handles.axes2,'Position',[0.564, 0.227, 0.353, 0.666]);
990.             title(handles.axes2,'Rechts, genormaliseerd');
991.             legend(handles.axes2,'Sensor1','Sensor2');
992.             legend(handles.axes2,'Orientation','vertical','Location','northwest');
993.             xlabel(handles.axes2,'Meetsample');
994.             ylabel(handles.axes2,'Gewicht[gram]');
995.             xlim(handles.axes2, [0 Size]);
996.             ylim(handles.axes2, 'auto');
997.             pan(handles.axes2, 'on');
998.         end
999.
1000.        %%beidenorm groot
1001.        if checkboxlinks == 0 && checkboxrechts == 0 && checkboxbeide == 1 && checkbo
           xnorm == 1;
1002.            cla(handles.axes2);
1003.            cla(handles.axes3);
1004.            set(handles.slider2,'Value',bnr);
1005.            set(handles.text5,'String',num2str(bnr));
1006.            set(handles.axes2,'Visible','on');
1007.            setappdata(gcf,'plot2visible',1);

```



```

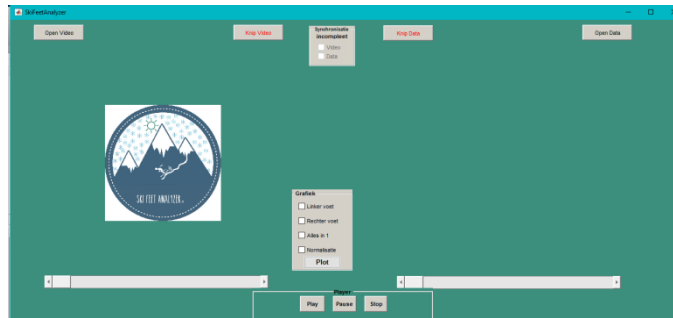
1008.     setappdata(gcf,'plotstatus1',VoetzooldataN);
1009.     set(handles.axes3,'Visible','off');
1010.     plot(handles.axes2,VoetzooldataN(k,:));
1011.     hold(handles.axes2,'on');
1012.     set(handles.axes2,'Position',[0.564, 0.227, 0.353, 0.666]);
1013.     title(handles.axes2,'Alle data, genormaliseerd');
1014.     legend(handles.axes2,'L Sensor1','L Sensor2','R Sensor3','R Sensor4');
1015.     legend(handles.axes2,'Orientation','vertical','Location','northwest');
1016.     xlabel(handles.axes2,'Meetsample');
1017.     ylabel(handles.axes2,'Gewicht[gram]');
1018.     xlim(handles.axes2, [0 Size]);
1019.     ylim(handles.axes2, 'auto');
1020.     pan(handles.axes2, 'on');
1021.     end
1022.
1023.     %%beidenorm klein
1024.     if checkboxlinks == 1 && checkboxrechts == 1 && checkboxbeide == 0 && checkbo
        xnorm == 1
1025.         cla(handles.axes2);
1026.         cla(handles.axes3);
1027.         set(handles.slider2,'Value',bnr);
1028.         set(handles.text5,'String',num2str(bnr));
1029.         set(handles.axes2,'Visible','on');
1030.         setappdata(gcf,'plot2visible',1);
1031.         setappdata(gcf,'plotstatus1',LinksN);
1032.         plot(handles.axes2,LinksN(k,:));
1033.         hold(handles.axes2,'on');
1034.         set(handles.axes2,'Position',[.552, .562, .37, .354]);
1035.         title(handles.axes2,'Genormaliseerd');
1036.         legend(handles.axes2,'L Sensor1','L Sensor2');
1037.         legend(handles.axes2,'Orientation','horizontal','Location','north');
1038.         xlabel(handles.axes2,'Meetsample','FontSize',8);
1039.         ylabel(handles.axes2,'Gewicht[gram]','FontSize',8);
1040.         xlim(handles.axes2, [0 Size]);
1041.         ylim(handles.axes2, 'auto');
1042.         pan(handles.axes2, 'on');
1043.
1044.         set(handles.axes3,'Visible','on');
1045.         setappdata(gcf,'plot3visible',1);
1046.         setappdata(gcf,'plotstatus2',RechtsN);
1047.         plot(handles.axes3,RechtsN(k,:));
1048.         set(handles.axes3,'Position',[.553,.206,.37,.329]);
1049.         legend(handles.axes3,'R Sensor1','R Sensor2');
1050.         legend(handles.axes3,'Orientation','horizontal','Location','north');
1051.         xlabel(handles.axes3,'Meetsample','FontSize',8);
1052.         ylabel(handles.axes3,'Gewicht[gram]','FontSize',8);
1053.         xlim(handles.axes3, [0 Size]);
1054.         ylim(handles.axes3, 'auto');
1055.         pan(handles.axes3, 'on');
1056.     end

```



De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren.

De laatste regels van het script zorgen ervoor het logo van Ski Feet Analyzer zichtbaar is zodra de gebruiker de UI opstart.



```
1058. % --- Executes during object creation, after setting all properties.
1059. function axes7_CreateFcn(hObject, eventdata, handles)
1060. % hObject    handle to axes7 (see GCBO)
1061. % eventdata  reserved - to be defined in a future version of MATLAB
1062. % handles    empty - handles not created until after all CreateFcns called
1063.
1064. % Hint: place code in OpeningFcn to populate axes7
1065. axes(hObject)
1066. imshow('Logo.jpg')
```

Bijlage VI – Evaluatie persoonlijke leerdoelen

Leerdoel 1. Programmeren; de bewegingstechnoloog kan zelfstandig softwareprogramma's ontwikkelen en toepassen voor randapparatuur en bewegingstechnologische toepassingen.

In dit onderzoek wordt er in zowel in Matlab[®] als in Arduino IDE[®] geprogrammeerd. Gedurende de opleiding is er veel in Matlab[®] gewerkt en de kennis hiervan is uitgebreid met het volgen van de minor Sporttechnologie. Een groot deel van deze opdrachten gedurende de opleiding was in groepsverband of de opdracht was aan meerdere projectgroepen gesteld. De onderzoeker is in dit onderzoek echter niet de enige persoon die aan het onderzoek werkt. Natuurlijk kan er hulp gevraagd worden aan experts vanuit De Haagse Hogeschool, maar de onderzoeker dient alles voornamelijk zelf uit te werken. Gedurende opleiding is er geringe aandacht geweest voor het programma en werken met de Arduino[®]. De onderzoeker dient zich dan ook uitgebreid in te lezen in het werken met deze hard- en software.

Evaluatie leerdoel 1.

Op het gebied van programmeren heeft de onderzoeker veel geleerd. Zowel het werk in Matlab als in Arduin IDE. Zoals hierboven benoemd is dit de eerste opdracht die zelfstandig is uitgevoerd. Wat duidelijk is geworden gedurende het project is het dat meer tijd kost dan vooraf ingepland. Voornamelijk het uitzoeken van programmeer methodes was tijdrovend. De onderzoeker heeft ook gebruik gemaakt van de experts vanuit De Haagse Hogeschool en is hun zeer dankbaar voor hun bijdrage. Verwacht wordt dat de onderzoeker, zonder externe expertise, in tijdnood geweest gekomen om de deadline van dit onderzoek te behalen.

De onderzoeker is zeer tevreden met het uiteindelijke resultaat van de geprogrammeerde delen.

Leerdoel 2. Zelfstandig werken; de bewegingstechnoloog kan op een weloverwogen wijze zelfstandig oplossingsmethoden kiezen, toepassen en controleren.

Evaluatie leerdoel 2.

Het zelfstandig werken is de onderzoeker zeer goed bevallen. Gedurende de opleiding zijn er maar weinig projecten volledig zelfstandig uitgevoerd. De overgang naar zelfstandig werken is echter zonder problemen gegaan, mede door Dalton achtergrond. De onderzoeker kan zich goed concentreren en uren werken zonder het door te hebben.

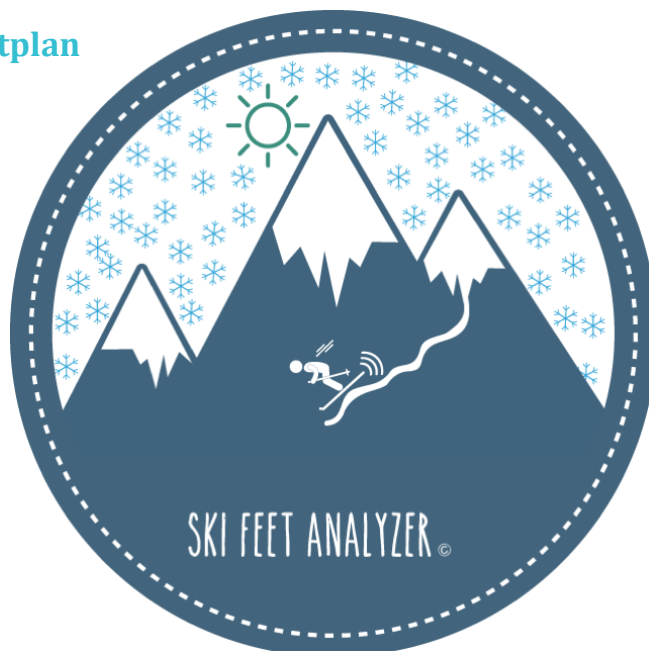
Leerdoel 3. Management; de bewegingstechnoloog kan zijn eigen werk, projecten zelfstandig plannen, organiseren, coördineren en evalueren. Hij is flexibel en klantgericht in zijn gedrag en combineert doorzettingsvermogen en ondernemerschap om de vooraf bepaalde doelstelling(en) te bereiken.

Zoals al vermeld, is dit een volledig, zelfstandig onderzoek. De werkwijze en planning worden door de onderzoeker zelf opgesteld en dienen alleen door de onderzoeker te worden nagestreefd. Natuurlijk worden er ook afspraken met begeleiders gemaakt waaraan voldaan dient te worden, de onderzoeker dient echter zichzelf te controleren of alles naar planning verloopt. Tevens heeft de onderzoeker maar één denkwijze en geen invloeden van andere projectleden. Om meerdere denkwijzen te ontvangen en deze feedback te kunnen verwerken in de studie zal de onderzoeker zelf naar deze invloeden moeten vragen bij derden.

Evaluatie leerdoel 3.

Het volledig zelfstandig managen van het onderzoek is in grote lijnen met tevredenheid verlopen. De vooraf opgesteld planning heeft minder rugsteun gegeven als vooraf verwacht. Vanaf +- onderzoekswEEK 6 kon deze planning niet worden behaald. Mede omdat het ontwikkelen van het meetsysteem vertraging opliep hebben de metingen veel later als gepland plaatsgevonden. Tevens is er geen meting op de rollerband uitgevoerd. De onderzoeker heeft ook moeite gehad met het benaderen van de experts vanuit de opleiding. 'Eerst veel zelf proberen en dan pas hulp vragen.' Deze werkwijze heeft invloed gehad op de uiteindelijke vertraging van de geplande taken. Dit had voorkomen kunnen worden door eerder deze hulp in te schakelen.

Bijlage VII – Projectplan



De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren

| Projectplan |

Auteur	Agneta Stokhof (12086339)
Opleiding	Mens en Techniek - Bewegingstechnologie, de Haagse Hogeschool
Datum	Woensdag 18 januari 2017
Begeleider	mevr. J. Koopman



Algemene gegevens

Naam: Agneta Brigida Stokhof
Studentnummer: 12086339
E-mail: agnetastokhof@hotmail.com
Datum: 18 januari 2017
Begeleider: mevr. J. Koopman jkoopman@hhs.nl

Studievoortgang

Behaalde studiepunten in module 9 t/m 11:	36
Minor Sporttechnologie	afgerond
Stage 2 afgerond:	nee
Totaal aantal behaalde vrije studiepunten:	20
Openstaande toetsen:	geen

Onderwerp

Titel:	De ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van voetzool druksensoren
Werkveld:	Sport
Beroepsrol:	Praktijk gerichte onderzoeker & ontwikkelaar
Extern project:	Nee



Inhoudsopgave

Inhoudsopgave.....	113-113
1 Inleiding.....	114-114
1.1 Aanleiding & probleemstelling	114-114
1.1.1 Centrale balans & Alpine balans	115-115
2 Methode.....	116-116
3 Planning.....	118-118
4 Persoonlijke leerdoelen	118-118
5 Bibliografie	119-119
6 Bijlagen.....	120-120
6.1 Bijlage I	120-120
6.2 Bijlage II	Fout! Bladwijzer niet gedefinieerd. -122

1 Inleiding

1.1 Aanleiding & probleemstelling

De ontwikkeling van de trainingsfaciliteiten van sportbonden staat niet stil. Veel sportbonden beschikken over faciliteiten en hulpmiddelen ter bevordering van trainingen en prestaties. Zo ontvangen hockeycoaches tijdens wedstrijden live feedback van de assistent-trainer, die zich op de tribune bevindt met een tablet waarmee hij de wedstrijd gemakkelijk kan analyseren. Ook krijgt de hoofdcoach op zijn eigen tablet de hartslagen van zijn spelers te zien. Met deze informatie kan hij zijn team direct aansturen. Ook de Nederlandse turners beschikken over videosoftware waarin zij tijdens een training hun oefening direct na uitvoering kunnen analyseren (1).

Binnen de wereld van het skiën maakt men ook gebruik van videoanalyse. Dit hulpmiddel is echter niet specifiek op de skisport gericht. Dit onderzoek focust zich op het verbeteren van de middelen voor coaches en atleten ter bevordering van de prestaties in de skisport. Ook zal het hulpmiddel toegankelijk zijn voor skileraren. Zij kunnen met de output van het ontwikkelde hulpmiddelen de eigenvaardigheid verbeteren en op deze manier de informatie doorspelen naar de cursisten.

De skiwereld heeft zich de laatste jaren voornamelijk ontwikkeld met verbetering van materiaal en technieken (2). Het bepalen van de krachten die meespelen tijdens het skiën zijn belangrijk voor het ontwikkelen van ski's, bindingen, skischoenen en voor coaches en atleten voor het verbeteren van de prestaties (3). Er zijn diverse studies die zich hier op richten. Zo is er een skirobot ontwikkeld om te onderzoeken welke gewrichtshoeken het beste zijn tijdens gesneden bochten (4). Ook is er een wiskundige simulatie ontwikkeld waarin de ski-sneeuw interactie wordt gesimuleerd om zo informatie te verzamelen voor de eigenschappen van een ski (5,6). Een ander onderzoek (7,8) heeft de relatie tussen de skibuiging, druk tussen ski en ondergrond en de druk tussen de skischoen en ski onderzocht met gebruik van twee ski's met verschillende stijfheden. In een onderzoek uit 2008 (9) is er een 3D-model ontwikkeld waarin de kinematica van het skiën kan worden geanalyseerd. Hier werd onder andere gebruik gemaakt van een Inertial Motion Unit (IMU), GPS en RS-scan voetzolen.

Tijdens het skiën bepalen de krachten die de skiër uitoefent op de ski's, het gedrag van de ski's. Deze krachten worden geleverd door de gewrichtshoeken in de centrale balans (hierna CB) en alpine balans (hierna AB). Deze twee punten worden toegelicht in hoofdstuk 1.1.1.

De overdracht van al deze krachten tussen de skiër en het materiaal vindt plaats in de voeten. Deze hebben namelijk direct contact met het materiaal. Het is dus belangrijk om dit te kunnen analyseren. In dit onderzoek wordt er een interface ontwikkeld waarin skiërs en coaches video in combinatie met informatie uit druksensoren in de voetzool van de skischoen kunnen analyseren.

Bewegingen kunnen makkelijk worden geanalyseerd met standaard videoanalyse software, zoals Kinovea[®]. Krachten kunnen echter niet op deze manier worden geanalyseerd. Om deze krachten te kunnen analyseren dient er gebruik te worden gemaakt van sensoren.

Twee verschillende meetsystemen zijn eerder gebruikt voor het bepalen van de krachten die op de ski werken: voetzool druksensoren en een 3D-krachtenplaat dynamometers. Voetzool druksensoren (zoals Pedar(Novel), Parotec(Paromed) & Tekscan(F-scan)) kwantificeren verticale krachten en het centrum van druk van de voetzool in de skischoen (3). Scheiber (10) analyseerde skiërs en maakte gebruik van Novel Insoles. Müller (11) ontwikkelde een real-time training feedback apparaat voor het verstrekken van informatie over de actuele krachten van de alpine skiër door gebruik van een eigen ontwikkelde voetzool met druksensoren.

Twee voordelen van het gebruik van druk voetzolen zijn te benoemen. Ten eerste kan de skiër, in vergelijking met dynamometers, zijn eigen materiaal gebruiken. Ten tweede beïnvloedt het meetsysteem met voetzolen de techniek en uitvoering niet, omdat de zooltjes dun en licht zijn. De voetzolen hebben echter ook nadelen ten opzichten van de 3D-dynamometers. De krachten die gemeten worden door de voetzooltjes komen niet overeen met de daadwerkelijke krachten die worden gemeten op de bindingen en ski's. De overdracht van de skischoen op de binding en ski wordt met de voetzooltjes niet gemeten. Een dynamometer wordt tussen de ski en de binding geplaatst en meet dus wel de krachten die daadwerkelijk in de ski optreden (3). De onderzoeker is echter geïnteresseerd in de krachten die door de skiër gegenereerd worden. Er zal in dit onderzoek daarom gebruik worden gemaakt van voetzolen. In een onderzoek van Koop (12) wordt een overzicht opgesteld van de eigenschappen van bestaande voetzolen met druksensoren. De voetzolen die in het onderzoek van Koop worden benoemd zijn helaas niet beschikbaar voor dit onderzoek. Voor het onderzoek wordt dan ook door de onderzoeker voetzolen ontwikkeld. De parameters die volgen uit deze voetzolen zijn de CB en de stuurdruk van de AB. Deze twee parameters worden hieronder toegelicht.

1.1.1 Centrale balans & Alpine balans

Zoals eerder vermeld bepalen de CB en AB de krachten op de ski's. Dit zijn dan ook de twee basiscomponenten van het skiën. Voor dit onderzoek dient de definitie van de CB en de AB uitgelicht te worden.

De CB zegt iets over de balans, de positie van het zwaartepunt, tijdens het skiën in het sagitale vlak. De AB zegt iets over de balans in laterale richting, in het frontale vlak, tijdens het maken van een bocht (figuur 1). Er wordt gesproken over balans, omdat skiën dynamisch is (31).

De CB wordt in zes verschillende aandachtspunten verdeeld waarvan de volgende drie punten van belang zijn voor dit onderzoek:

1. Hele voetzool is belast; het lichaamsgewicht is over de gehele voet verdeeld.
Wanneer men op de hele voet staat is er minder kans op balansproblemen. Door de hele voetzool te belasten wordt de druk het beste over de ski verdeeld waardoor de ski optimaal glijdt en stuurt. Tevens maakt het belasten van de gehele voetzool de bewegingsbereidheid optimaal. Op de tenen of hiel balanceren is lastiger dan als men op de gehele voet staat. Dit geldt ook voor tijdens het skiën.
2. Enkel en knieën licht gebogen, heup is licht gevouwen.
Tijdens het glijden dient men bewegingsbereid te zijn om oneffenheden en de massa traagheid op te kunnen vangen. Door de enkels, knieën en heupen licht te buigen kan men hier mee omgaan zonder de balans te verstoren. Het is van belang dat alle drie de gewrichten hierin meebewegen. Wanneer niet alle drie de gewrichten functioneren raakt men ook uit balans.
3. Lichte scheendruk.
Door lichte scheendruk maakt men contact met de ski. Deze kan af- en toenemen om de balans te bewaren. Door lichte scheendruk is men bereid om de druk te vergroten of te verkleinen (31).

De CB wordt in voor- en achterwaartse richting verstoord door onder andere oneffenheden in de ondergrond die zorgen voor balansverstoringen. Wanneer de skiër te weinig of geen scheendruk heeft en de voetzool dus niet geheel belast is, staat de skiër 'achterin', ook wel Rücklage (DE) genoemd (figuur 2A.). De druk op de achterkant van de ski's is dan groter dan over de voorkant en dit heeft gevolgen op het sturen van de ski's. Wanneer de skiër te veel scheendruk heeft en voornamelijk op de tenen staat, heeft dit ook gevolgen op het sturen, de achterkanten van de ski's zullen uitbreken (figuur 2C.) De centrale positie is de basis van de bewegingspiramide van het skiën en dus van essentieel belang. Deze bewegingspiramide wordt weergegeven in figuur 3 in bijlage I (14).

Uit de combinatie van de CB en zijwaarts bewegen volgt de AB. De AB heeft drie kenmerken: kanthoek, aaneengesloten bewegen en stuurdruk. In dit onderzoek is de stuurdruk van belang. De druk die de skiër ervaart wordt stuurdruk genoemd. De stuurdruk onder de ski wordt bepaald door de mate van het in de richting draaien van de ski's, de mate van kantelen van de ski's en het fixeren van de ski's op de helling. Hierdoor buigt de ski in de lengterichting waarmee de sturing tot stand komt. De grote van de krachten, die hierbij door de skiër ontvangen worden, zijn afhankelijk van het moment waar de skiër zich in de bocht bevindt. Door de centrale balans en de alpine balans wordt deze druk gereguleerd.

Het belasten van de gehele voetzool en de stuurdruk die wordt uitgeoefend op de skiër is te meten met behulp van druksensoren. Door meerdere sensoren onder iedere voet te plaatsen is het mogelijk om te meten of de skiër de gehele voetzool belast. Omdat de sensoren in beide voeten gelijktijdig meten is de onderlinge verhouding in stuurdruk te bepalen.

Het doel van het onderzoek is de ontwikkeling van een trainingshulpmiddel voor skiërs met gebruik van druksensoren onder de voet. Met dit trainingshulpmiddel wordt de CB en de stuurdruk van de AB geanalyseerd.

2 Methode

Om de doelstelling van dit onderzoek te behalen dient er een methode opgesteld te worden. Deze methode wordt uitgevoerd over een periode van 14 weken. De ontwikkeling van het meetsysteem zal eerder van start gaan.

Voor dit onderzoek wordt er een meetsysteem ontwikkeld door de onderzoeker in samenwerking met externe personen die beschikken over de expertise om dit systeem, de voetzolen met druksensoren, te kunnen ontwikkelen. Uit reeds vergaarde informatie vanuit deze experts zal het meetsysteem beschikken over driesensoren per voet en zal gebruik worden gemaakt van een Arduino[®]. De Arduino[®] dient zo geprogrammeerd te worden dat de data opslagen wordt op een Secure Digital kaart, zodat de data achteraf kan worden geanalyseerd.

Om de interface te kunnen ontwerpen dient duidelijk te zijn hoe de data uit de voetzolen eruit ziet. Het uitwerken van het script om de gewenste parameters uit de data te krijgen, kan gestart worden wanneer er data uit de voetzolen is vergaard en de wensen van de doelgroep bekend zijn. Dit script zal met gebruik van Matlab[®] (Mathworks 2015b) worden opgesteld. De uiteindelijke interface zal worden ontworpen in de GUIDE functie van Matlab[®].

Het meetprotocol waarmee data zal worden verzameld om de interface te ontwikkelen kan op dit moment grof worden opgesteld. Specificeren en uitbreiden van dit protocol zal volgen uit de analysefase van de studie. Zo is nu al gesteld dat er minimaal met een high speed camera gefilmd dient te worden en er synchronisatie moment moet worden toegepast, zodat de data uit de voetzolen te vergelijken is met de videobeelden. Ook is het vrijwel zeker dat de data uit de voetzolen geresampled dient te worden omdat verwacht wordt dat de video en voetzolen niet met dezelfde frequentie meten.

Het trainingshulpmiddel dient na afronding van het onderzoek toegepast te kunnen worden. Omdat het skiën niet enkel in de sneeuw plaats vindt maar ook op indoor rollerbanden, dient het trainingshulpmiddel op beide ondergronden toepasbaar te zijn. Bekend is dat deze twee verschillende ondergronden om een andere data bewerking vergen. Een sneeuw ondergrond is in tegenstelling tot een indoor rollerband, niet egaal. Wanneer de skiër over een bult of kuil skiet, geeft dit een verandering/storing van de stuurdruk. Om dit probleem te kunnen verhelpen dient er op beide ondergronden data te worden gewonnen.

Om het einddoel te bepalen en de methode invulling te geven zijn er deelvragen opgesteld. Al deze vragen dienen gedurende de tijdsperiode van het onderzoek beantwoord te worden.

Ontwikkeling meetsysteem

- 4 Welke sensoren dienen gebruikt te worden?
- 5 Hoe dient de Arduino[®] geprogrammeerd en opgebouwd te worden?
- 6 Hoe dient het ontwikkelde meetsysteem gevalideerd te worden?
- 7 Met welke frequentie dient er gemeten te worden?
 - 7.1 Wat is de maximale frequentie waarmee de Arduino[®] data kan opslaan op de Secure Digital kaart?
- 8 Waar moet het meetsysteem aan voldoen zodat de doelgroep bereikt wordt?
 - 8.1 Dienen de sensoren bevestigd te worden op een voetzool?
 - 8.1.1 Welke schoenmaat is veel voorkomend in de doelgroep?
 - 8.1.2 Op welke manier is de voetzool aan te passen aan de gebruiker?

Ontwikkeling interface

- 4) Door middel van welke databewerking volgen de waardes van de gewenste parameters. Hoe dient de data geïnterpreteerd te worden?
 - a. De keuze op de ondergrond invloed op deze databewerking:
 - i. Met welke factoren dient rekening te worden gehouden bij het meten op de rollerband en de sneeuw?
 1. Kracht vanuit de ondergrond door bijv. sneeuw heuvel moet goed worden onderscheiden in data. Hoe kunnen deze krachten van elkaar onderscheiden worden?
- 5) Over welke lay-out en functies dient te interface te beschikken?
 - a. Wat zijn veel gebruikte functies in (video)analyse software?

Doelgroep

- 1) In welke vorm wil de doelgroep de informatie/uitkomsten ontvangen?
 - a. Moet de werkelijkheid gemeten worden met specifieke waardes?
 - b. In welke vorm dient de data zichtbaar te zijn? Grafiek, afbeeldingen etc.?
 - c. Zijn de wensen van de doelgroep haalbaar binnen de onderzoeksperiode van 14 weken?
- 2) Hoe toegankelijk is de gebruikte software voor het ontwikkelen van de interface, voor de doelgroep?
 - a. Kan de interface (GUI bestand) gebruikt worden door de doelgroep zonder de volledige Matlab software geïnstalleerd te hebben?

3 Planning

De planning van deze studie wordt weergegeven in Bijlage II. Dit is een grove en globale planning. Informatie die gewonnen zal worden gedurende deze studie zal de planning verduidelijken en uitbreiden.

4 Persoonlijke leerdoelen

- 1) Programmeren; de bewegingstechnoloog kan zelfstandig softwareprogramma's ontwikkelen en toepassen voor randapparatuur en bewegingstechnologische toepassingen.

In dit onderzoek wordt er in zowel in Matlab[®] als in Arduino IDE[®] geprogrammeerd. Gedurende de opleiding is er veel in Matlab[®] gewerkt en de kennis hiervan is uitgebreid met het volgen van de minor Sporttechnologie. Een groot deel van deze opdrachten gedurende de opleiding was in groepsverband of de opdracht was aan meerdere projectgroepen gesteld. De onderzoeker is in dit onderzoek echter niet de enige persoon die aan het onderzoek werkt. Natuurlijk kan er hulp gevraagd worden aan experts vanuit De Haagse Hogeschool, maar de onderzoeker dient alles voornamelijk zelf uit te werken. Gedurende opleiding is er geringe aandacht geweest voor het programma en werken met de Arduino[®]. De onderzoeker dient zich dan ook uitgebreid in te lezen in het werken met deze hard- en software.

- 2) Zelfstandig werken; de bewegingstechnoloog kan op een weloverwogen wijze zelfstandig oplossingsmethoden kiezen, toepassen en controleren.
- 3) Management; de bewegingstechnoloog kan zijn eigen werk, projecten zelfstandig plannen, organiseren, coördineren en evalueren. Hij is flexibel en klantgericht in zijn gedrag en combineert doorzettingsvermogen en ondernemerschap om de vooraf bepaalde doelstelling(en) te bereiken.

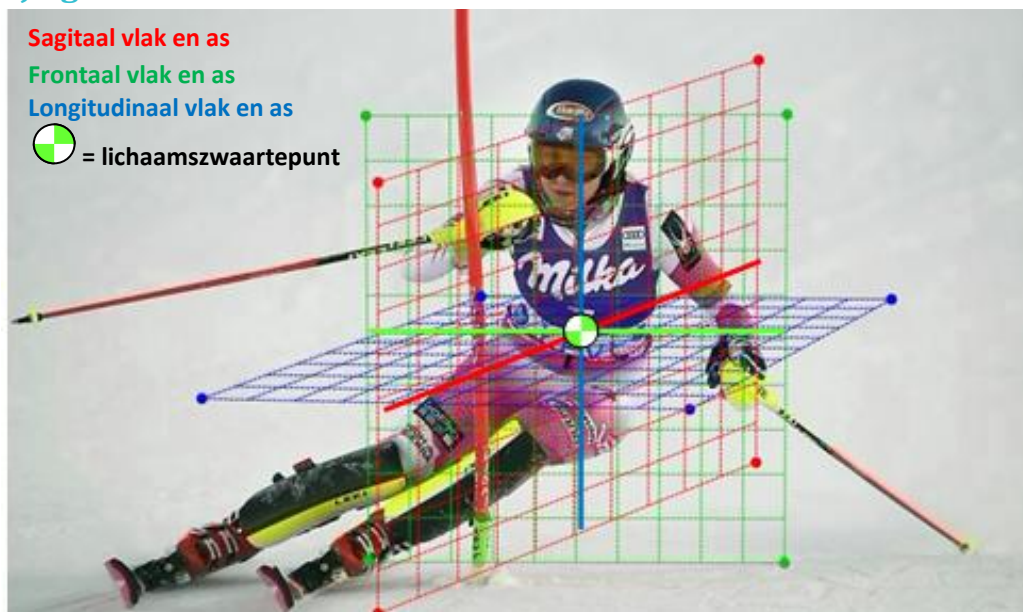
Zoals al vermeld, is dit een volledig, zelfstandig onderzoek. De werkwijze en planning worden door de onderzoeker zelf opgesteld en dienen alleen door de onderzoeker te worden nagestreefd. Natuurlijk worden er ook afspraken met begeleiders gemaakt waaraan voldaan dient te worden, de onderzoeker dient echter zichzelf te controleren of alles naar planning verloopt. Tevens heeft de onderzoeker maar één denkwijze en geen invloeden van andere projectleden. Om meerdere denkwijzen te ontvangen en deze feedback te kunnen verwerken in de studie zal de onderzoeker zelf naar deze invloeden moeten vragen bij derden.

5 Bibliografie

1. Innosport-S'Hertogenbosch. innosportlabs. [Online].; 2013 [cited 2016 12 17. Available from: <http://www.innosportlabs-hertogenbosch.nl/projecten/turntrainerscockpit>.
2. Kondo A, Doki H, Hirose K. An attempt of a new motion measurement method for alpine ski turns using inertial sensors. 2012;(34): p. 421-426.
3. Stricker G, Scheiber P, Lindenhof E, Müller E. Determination of forces in alpine skiing and snowboarding: Validation of a mobile data acquisition system. 2010;(10): p. 31-41.
4. Yoneyama T, Kagawa H, Unemoto M, Lizuka T, Scott NW. A Ski robot system for qualitative modeling of the carved turn. 2009;(11): p. 131-141.
5. Federolf P, Roos M, Luthi A, Dual J. Finite element simulation of the ski-snow interaction of an alpine ski in a carved turn. 2010;(12): p. 123-133.
6. Heinrich D, Mössner M, Kaps P, Nachbauer W. Calculation of the contact pressure between ski and snow during a carved turn in Alpine skiing. 2009 Maart 18; 2010(20): p. 485-492.
7. Yoneyama T, Kitabe M, Osada K. Investigation on the ski-snow interaction in a carved turn based on the actual measurement. 2010;(2): p. 2901-2906.
8. Yoneyama T, Kagawa H, Osada K. Measurement of ski snow-pressure profiles. 2007;(10): p. 145-156.
9. Brodie M,WA, Page W. Fusion motion capture: a prototype system using inertial measurement units and GPS for the biomechanical analysis of ski racing. 2008;(1-1): p. 17-28.
10. Schreiber P, Schwameder H, Müller E. Characteristics of the force application point - a method to identify learning processes in alpine skiing. 2006; 2: p. 564-567.
11. Müller E, Ebert C, Senner V. Approach for a realtime training feedback algorithm in skiing. 2007;; p. 47.
12. Koop J. Insole pressure measurement in speed skating. Bachelor Thesis. Den Haag: De Haagse Hogeschool, Mens en Techniek - Bewegingstechnologie; 2014.
13. Nederlandse-Ski-Vereniging. Opleiding Skibegeleider. In NskiV.. Den Haag; 2016.
14. Othmar Hofmann. Ski specifiek bewegingsvaardigheden. [Online].; 2016 [cited 2016 11 21. Available from: http://www.skidiscovery.nl/index.php?id=13233&title=ski-specifieke_bewegingsvaardigheden.

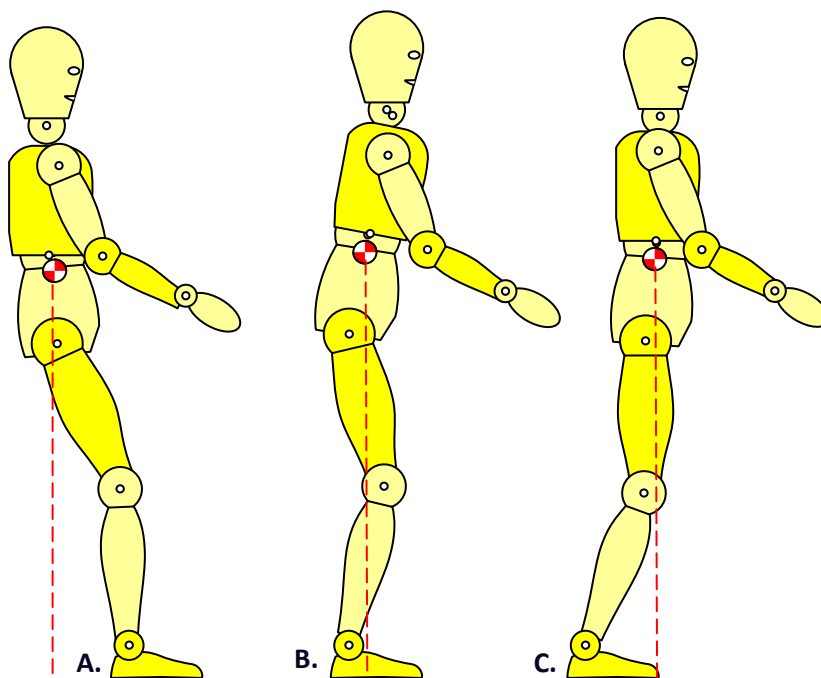
6 Bijlagen

6.1 Bijlage I



Figuur 16, Gedefinieerd assenstelsel van een skiër. Mikaela Shiffrin (USA) 26-11-2016 Levi (FIN)

Rood = sagitaal vlak en as. Groen = frontaal vlak en as. Blauw = longitudinaal vlak en as. ● = lichaamszwaartepunt



Figuur 2. Overzicht van CB posities.

A. Teveel achterovergebogen, de zogenoemde Rücklage

B. Correcte CB

C. Te veel voorovergebogen



Figuur 3. Ski specifieke bewegingsvaardigheden in de bewegingspiramide (14)

6.2 Bijlage II

		Afstudeer Planning																
Afstudeer Week		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Kalender Week			10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Taken																		
Concept Projectplan	15 dec																	
Definitief Projectplan	18 jan																	
Ontwikkeling meetsysteem	< 6 mrt																	
Validatie voetzolen																		
Analysefase																		
Literatuurstudie																		
Doelgroep onderzoek																		
Kennis vergaren ontwerp interface																		
Opstellen Meetprotocol																		
Grof opstellen interface																		
Uitvoerfase																		
Proefmeting																		
Ontwerpen Interface																		
Meting rollerband																		
Meting sneeuw																		
Verwerken data																		
Tussentijdse feedback doelgroep																		
Verwerken feedback doelgroep																		
Algemeen																		
Contact met begeleider																		
Documentatie Scriptie																		
Inleveren Scriptie																		
Voorbereiden verdediging																		
Verdediging en Eindbeoordeling																		

