

---

<b>Team 12</b>	
Martijn Debeuf	h
Frédéric Blondeel	h
Toon Sauvillers	h
Dirk Vanbeveren	h
Bert Van den Bosch	h
Seppe Van Steenberghe	h

---

## Inhoudsopgave

<b>1</b>	<b>Inleiding</b>	<b>2</b>
<b>2</b>	<b>Sensoren</b>	<b>2</b>
2.1	Rotatie . . . . .	3
2.2	Translatie . . . . .	4
<b>3</b>	<b>Visuele aanwijzingen</b>	<b>4</b>
3.1	FAST . . . . .	5
3.2	BRIEF . . . . .	6
<b>4</b>	<b>Besluit</b>	<b>6</b>
	<b>Referenties</b>	<b>7</b>

# 1 Inleiding

## 2 Sensoren

Een andere manier om beweging van een apparaat te detecteren is met behulp van de sensoren. Vandaag de dag zitten onze smartphones vol met sensoren die allerlei verschillende soorten data verzamelen [1]. Diegene die nuttig kunnen zijn voor het bepalen van de beweging zijn de Accelerometer, Gyroscop en Magnetometer.

**De Accelerometer** meet de acceleratie die het apparaat ondervindt. Hiermee kan deze sensor de oriëntatie bepalen van het apparaat aan de hand van de zwaartekracht of de beweging in eender welke lineaire richting.

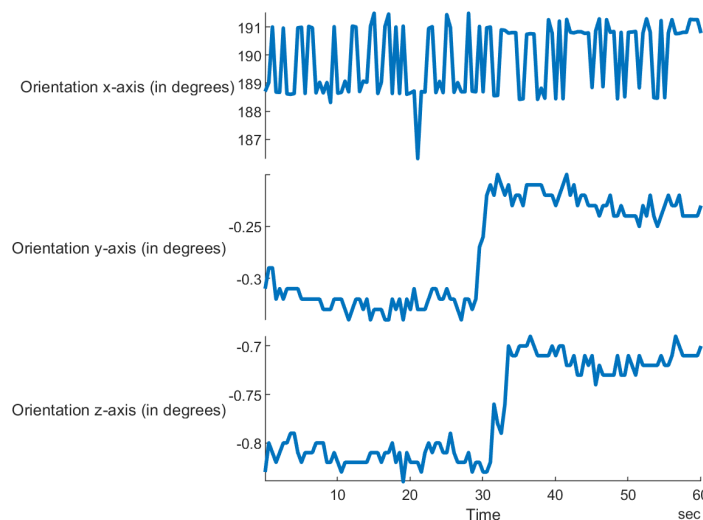
**De Gyroscop** geeft ook informatie over de oriëntatie van het apparaat maar is nauwkeuriger dan de accelerometer. Ook kan de gyroscop rotatie meten wat een accelerometer niet kan.

**De Magnetometer** kan magnetische velden detecteren. Wordt gebruikt om het noorden te vinden aan de hand van het aardmagnetisch veld.

De beweging van een apparaat kan opgesplitst worden in zijn rotatie en translatie. Voor beiden wordt gekeken naar de implementatie en hoe nauwkeurig de beweging bepaald kan worden.

## 2.1 Rotatie

Voor de rotatie wordt de **OrientationSensor** interface gebruikt. Hierbij is nog de keuze tussen de **AbsoluteOrientationSensor** interface en de **RelativeOrientationSensor** interface. Het verschil zit in welk assenstelsel gekozen wordt om de rotatie te definiëren. Bij de absolute is dit het referentie coördinatensysteem van de aarde. Bij de relatieve is dit een stationair coördinatensysteem. Hier wordt voor de relatieve gekozen omdat hierbij geen gebruik wordt gemaakt van de magnetometer voor het bepalen van het referentie coördinatensysteem van de aarde. Want deze kan verstoord worden door magnetische velden en dus de accuraatheid aantasten [2].

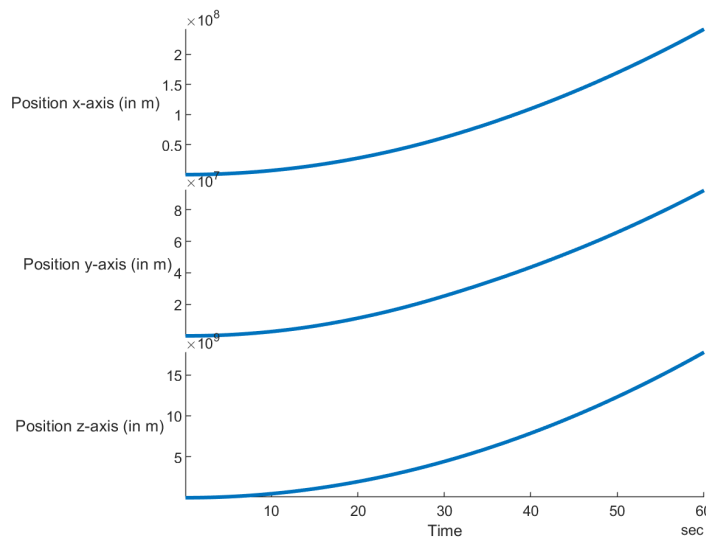


Figuur 1: Oriëntatie apparaat in rust, plat liggend op tafel gedurende 60 seconden volgens de x-as, y-as en z-as.

Zie figuur 1 voor de oriëntatie van een apparaat dat plat ligt op een tafel en niet beweegt. Gedurende deze 60 seconden aan data is het duidelijk dat er altijd een fout zit op de gemeten waarde. Deze is wel beperkt tot maximaal enkele graden wat accuraat genoeg is voor onze applicatie. Voor de rotatie te berekenen moet dan enkel gekeken worden naar het relatief verschil tussen twee opeenvolgende oriëntaties.

## 2.2 Translatie

Voor de translatie moet gekeken worden naar de acceleratie van het apparaat. Hiervoor wordt de **LinearAccelerationSensor** interface gebruikt. Hierbij is de zwaartekracht al van de gemeten waarde afgetrokken. De positie wordt dan berekend aan de hand van een dubbele integraal over de acceleratie omdat de acceleratie de tweede afgeleide is van de positie.



Figuur 2: Positie apparaat in rust, plat liggend op tafel gedurende 60 seconden volgens de x-as, y-as en z-as.

Zie figuur 2 voor de positie van een apparaat dat plat ligt op een tafel en niet beweegt gedurende 60 seconden. Hierop is te zien dat de positie een kwadratische drift heeft. Dit komt doordat ook bij de accelerometereer een fout zit op de gemeten waarde. Deze kwadratische drift zorgt ervoor dat de translatie van een apparaat berekenen aan de hand van sensoren niet mogelijk is voor onze applicatie.

## 3 Visuele aanwijzingen

Met behulp van de camera, kan een apparaat een beweging detecteren. Deze beweging vertaalt zich vervolgens in een transformatie. Om beweging te detecteren zijn er punten nodig waarop het algoritme zich kan focussen, dit zijn de **keypoints**. Tussen verschillende frames zullen deze keypoints zich verplaatsen. Het linken van de keypoints tussen deze twee frames zal resulteren in een beweging.

Een algoritme dat in de *Technical Committees* naar voren kwam is het **ORB**

algoritme [3]. ORB staat voor **O**riented **F**AST and **R**otated **B**RIEF, in *ORB: an efficient alternative to SIFT or SURF (2011)* bespreken E. Rublee, V. Rabaud, K. Konolige en G. Bradski waarom ORB te verkiezen is boven SIFT of SURF, twee andere keypoint detectie algoritmes.

Dit verslag bespreekt eerst FAST, een algoritme om keypoints te detecteren. Vervolgens gaat het over naar BRIEF, dit algoritme houdt zich bezig met de gevonden keypoints te koppelen aan elkaar.

### 3.1 FAST

**FAST** staat voor *Features from Accelerated Segment Test* en werd geïntroduceerd door E. Rosten en T. Drummond in 2005 [4]. FAST zoekt naar hoeken in de afbeelding waar de intensiteit van de omgeving hard verschilt. Een pixel  $p$  wordt als keypoint beschouwd wanneer in een *Bresenham cirkel* met straal drie twaalf opeenvolgende pixels de intensiteit verschilt met minimaal  $t$ , een voorafbepaalde drempelwaarde. Om het algoritme snel genoeg te laten verlopen maar niet teveel keypoints te vinden, wordt een drempelwaarde tussen de 20 en 70 gebruikt. Afhankelijk van de omgeving kan deze manueel worden aangepast in de applicatie.

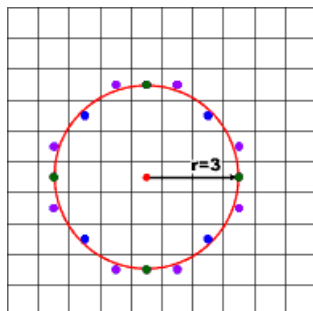
**Intensiteit** De intensiteit van een pixel is gelijk aan zijn grijswaarde. Bij de implementatie wordt de foto eerst omgezet naar een grijswaarde foto. De intensiteit zal zo een gewogen waarde worden van de RGB-waarden van de pixel. De coëfficiënten zijn de waarden gebruikt en vastgelegd door de *NTSC*<sup>1</sup>.

$$Intensiteit = 0.299 * R + 0.587 * G + 0.114 * B$$

**Bresenham cirkel** Het Bresenham algoritme is geïntroduceerd in de *Computer Graphics* om lijnen en andere vormen als pixels af te beelden. Door dit goed vast te leggen zal elke cirkel die genomen wordt, éézelfde vorm aannemen.[5] Zie afbeelding 3 voor een visualisatie van een Bresenham cirkel met straal drie.

---

<sup>1</sup>NTSC is *The National Television System Committee*, een analoge norm voor kleuren-tv geïntroduceerd in 1954.



Figuur 3: Bresenham cirkel met straal drie. De aangeduide pixels worden gebruikt om de cirkel te definiëren.

### 3.2 BRIEF

**Brief** staat voor *Binary Robust Independent Elementary Features*. Dit algoritme gebruikt de gevonden keypoints volgens 3.1

## 4 Besluit

## Referenties

- [1] Manisha Priyadarshini. Which Sensors Do I Have In My Smartphone? How Do They Work?, 2018.
- [2] Kenneth Rohde Christiansen and Annsi Kostiainen. Orientation Sensor, 2019.
- [3] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. *2011 International Conference on Computer Vision*, 2011.
- [4] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. *Tenth IEEE International Conference on Computer Vision (ICCV05) Volume 1*, 2005.
- [5] Jurg Nievergelt and Claus Hinrichs. *Algorithms and data structures - applications to graphics and geometry*. Textbook Equity Editions, 2014.