Onderzoek Document



|  |  |
| --- | --- |
| Studentnaam | Tim Spieringhs |
| Coach | Docent naam |
| Versie | 3 |
| Datum laatste verandering | 28-4-2021 |

|  |  |
| --- | --- |
| **Versiebeheer** | |
| **Versie** | **Veranderingen** |
| 1 | Onderzoek gemaakt |
| 2 | Feedback verwerkt van rop heb inhoud toegevoegd versiebeheer en titelpagina en een voorwoord en heb deelvragen verbeterd. |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Datum** | **Versie** | **Belangrijke wijziging** |
| 19-5 | 0.2 | Rationale toegevoegd over competenties die voldoende aanwezig zijn. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Voorwoord

In dit onderzoek ga ik voor de proftaak onderzoeken hoe we mensen gaan tellen die de trein in en uit gaan. Uiteraard kan dit maar op een paar manieren en de manier die mogelijk is voor mij zijn sensoren en hier gaat het onderzoek dan ook over. Ik ga namelijk onderzoeken welke sensor het best werkt voor deze situatie en waarom dat zo is.

Inhoud

[Voorwoord 2](#_Toc70683312)

[Probleemstelling 2](#_Toc70683313)

[Aanleiding 2](#_Toc70683314)

[Hoofdvraag en deelvragen 3](#_Toc70683315)

[Oplossing en bestaande oplossingen 4](#_Toc70683316)

[Onderzoek Deelvraag 1 5](#_Toc70683317)

[Library 5](#_Toc70683318)

[Lab 5](#_Toc70683319)

[Workshop 8](#_Toc70683320)

[Conclusie 8](#_Toc70683321)

[Onderzoek Deelvraag 2 9](#_Toc70683322)

[Field 9](#_Toc70683323)

[Library 9](#_Toc70683324)

[Lab 9](#_Toc70683325)

[Conclusie 10](#_Toc70683326)

[Onderzoek Deelvraag 3 11](#_Toc70683327)

[Eind Conclusie 11](#_Toc70683328)

[Bronnenlijst 11](#_Toc70683329)

# Probleemstelling

Om te meten hoeveel mensen in de trein zitten moeten we kunnen meten hoeveel mensen de trap op en af gaan via een sensor. Maar omdat het soms heel druk is op de trappen kunnen sommige sensoren dit niet accuraat meten. Hiervoor moeten we dus een sensor vinden die kan weten of iemand de trap op of afgaat en de sensor geen probleem heeft met drukte.

# Aanleiding

Omdat er grote groeperingen voorkomen op het perron word er een app gecreëerd waar je kan zien waar in de trein zitplekken vrij zijn. Hierbij kan je ook zien waar op het perron je moet staan voor de vrije zitplek. Om te weten of er zitplekken zijn moet erbij de trappen gemeten worden hoeveel mensen naar binnen en buiten gaan via een sensor. Maar omdat in de trein het soms heel druk kan zijn moeten we een sensor vinden die deze drukte aankan, zodat tijdens drukke tijden de sensor ook accuraat is.  
  
Doelstelling

Waarom kan een sensor moeite hebben met drukte.

* Als mensen dicht langs elkaar staan de sensor ze moeilijk uit elkaar kan houden

Waarom is het moeilijk voor een sensor om mensen uit elkaar te houden als te dicht op elkaar staan

* Als mensen dicht op elkaar staan kunnen sensoren soms niet zien dat er 2 of meer tegelijk naar binnen of naar buiten gaan

Waarom kunnen sensoren niet zien dat er 2 of meer tegelijk naar binnen of naar buiten gaan wanneer ze dicht op elkaar staan

* Omdat de data niet grote verschillen heeft tussen metingen

Waarom levert geen grote meting verschillen problemen op

* Omdat sommige sensoren niet accuraat genoeg kunnen meten en dus meerdere mensen als 1 meten

Waarom zijn sommige sensoren niet accuraat genoeg

* Omdat ze soms niet snel genoeg achter elkaar kunnen meten of de data niet 100% accuraat is

# Hoofdvraag en deelvragen

**Onderzoek/hoofdvraag:** Hoe kun je met een sensor tijdens drukte inkomende en uitgaande mensen meten?  
**Hypothese:** Via een combinatie van time of flight sensoren kan je bepaalde data onthouden waardoor je mensen uit elkaar kan houden.

**Deelvraag 1:** Welke sensor kan het snelst accuraat achter elkaar metingen doen?  
**Hypothese:** Time of flight sensoren kunnen heel snel en accuraat metingen doen.

**Deelvraag 2:** Hoe kan je het best mensen uit elkaar houden met de sensor?  
**Hypothese:** Via lengte en snelheid.

**Deelvraag 3:** Op welke richting moet de sensor wijzen voor de meest foutloze metingen?   
**Hypothese:** Een klein beetje schuin van elkaar af zou de meest foutloze metingen geven.

# Oplossing en bestaande oplossingen

Zoals in de hypotheses gemeld is wil ik het gaan oplossen met meerdere time of flight sensors die hoogte en breedte van mensen meten en via deze data ze uit elkaar te houden. Maar ook via 2 meetpunten zodat je weet welke kant ze opgaan.

Deze oplossing is met inspiratie bedacht van verschillende bestaande apparaten (bijvoorbeeld: https://www.youtube.com/watch?v=c91Ve-g0J2U).

# Onderzoek Deelvraag 1

**Deelvraag 1:** Welke sensor kan het snelst accuraat achter elkaar metingen doen?

## Library

**Infrarood sensor**

Infrarood sensor is heel simpel en kan goed kleine stromen van mensen tellen. Echter krijgt het problemen als er meerdere mensen tegelijk binnen komen daardoor is dit een slechte optie. Dit is omdat er in de trein grote groepen tegelijk naar binnen of naar buiten gaan.

**Ultrasonische sensor**

Ultrasonische sensor zijn net zoals de infrarood sensor heel simpel en kunnen op korte afstanden heel accuraat meten. Echter hebben ze de zelfde problemen als de infrarood sensor, ze kunnen namelijk heel lastig grote groepen uit elkaar houden. Maar de sensor kan ook heel erg gestoord worden door het geluid van de omgeving.

**Time of flight sensor**

Een time of flight sensor is volgens People Counting Solutions een van de beste opties om mensen te tellen. Dit is omdat de sensor niet gevoelig is voor omgevingslicht en het betrouwbaar en nauwkeurig is voor mensen stromen te meten.

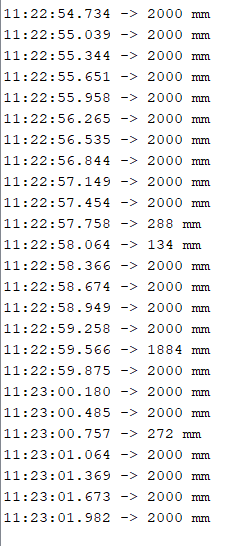
**Design**

Om te zorgen dat de time of flight sensor goed werkt en meerdere mensen kan meten hebben we meerdere er van nodig. 2 op het platfond en 1 of 2 op de muur hierdoor weet je welke richting ze op gaan en hoeveel er zijn. Echter zijn de 1 of 2 extra op de muur niet perse nodig maar zorgen alleen als een security op de 2 sensoren op het platfond.



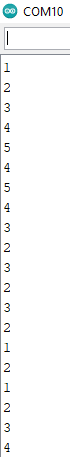
## Lab

**Component testing met TOF10120 sensor (time of flight sensor)**



Figuur 1 De meetwaarden van de sensor via de arduino

In figuur 1 kan je goed zien dat ik mijn hand 2x over de sensor heb gehouden. Echter zie je ook een minimaal probleem dat de sensor soms metingen geeft rond de 1900. Dit is een minimaal probleem omdat dit altijd rond de 1900 is en dit te verwerken is in code. Ook heb ik de sensor tegen het platfond aangehouden en gekeken of dit goed werkte en tegen de muur en vloer en alle 3 werkte hij goed (de meetwaarde waren van het platfond). Ook heb ik getest als ik heel snel achter elkaar me hand erover doet of hij dan ook 2000 mm ertussen meet. Dit deed hij ook en hierdoor blijkt dat de sensor makkelijk meerdere mensen kan meten vanwege de snelle metingen.



Figuur 2 Meetwaarden van 2 sensoren via de arduino

In figuur 2 zie de resultaten van een meer uitgewerkte code. In de code heb ik rekening met de rare spikes van ongeveer 1900 en heb ik via state gedrag gezorgd dat hij nauwkeurig bij houdt waar in de proces de arduino zit (zie proces flowchat in figuur 4). Hieruit blijkt ook dat de Time Of Flight sensor (TOF) goed werkt voor de situatie.



Figuur 3 De TOF10120 sensor

## Workshop

**Prototyping**

**Code review**

Mijn eerste code was heel rommelig en was heel onoverzichtelijk en had meerdere dubbele codes. Om de code op te schonen ga ik gebruik maken van state gedrag en switch cases. Hierdoor kan je makkelijk zien wat er gebeurd en wanneer. Om dubbele code te vermijden gebruik een bool statement in de SensorRead functie. Als de bool op true staat leest hij het adres van sensor 1 en als hij op false staat dan leest hij die van sensor 2. De bool word getoggled op het begin van de loop dus hierdoor gaat hij omste beurt de sensoren langs.

## Conclusie

De conclusie voor de deelvraag: welke sensor kan het snelst accuraat achter elkaar metingen doen?.   
Uit mijn onderzoek blijkt dat er de time of flight sensor word aangeraden vanwege zijn snelheid en hoe accuraat hij is. Dit heb ik ook getest met simpele component tests en een prototype hier uit bleek ook dat het waar is dat de time of flight heel goed werkt om mensen snel en accuraat te tellen.

# Onderzoek Deelvraag 2

**Deelvraag 2:** Hoe kan je het best mensen uit elkaar houden met sensoren?

## Field

**Probleem analyse**

Er zijn 2 grote problemen met mensen uit elkaar te houden met sensoren. De eerste is dat je moeilijk kan zien welke richting de persoon op gaat. De tweede is dat de sensor de afstand meet en niet personen zelf, er moet dus met de afstand bepaald worden of iemand er onderloopt of staat.

**Probleem 1**

Het probleem kan heel makkelijk ontmanteld worden om twee sensoren te pakken en hiermee te bepalen welke richting de persoon heen loopt. Dit doe je door de sensoren dicht bij elkaar te zetten en via welke als eerst geactiveerd word te meten welke richting de persoon heen loopt. Je kunt het goed zien bij figuur 4 waar de twee rode cirkels de sensoren voor stellen en met de pijlen kan je goed zien dat je kan weten welke richting de persoon oploopt via welke sensor als eerst activeert.



Figuur Visueel voorbeel van de oplossing

**Probleem 2**

Het probleem is wat moeilijker om optelossen echter kan je met twee sensoren en state gedrag goed bijhouden of iemand stil staat of doorloopt. Ik ga de state gedrag verder uitwerken in de lab.

## Library

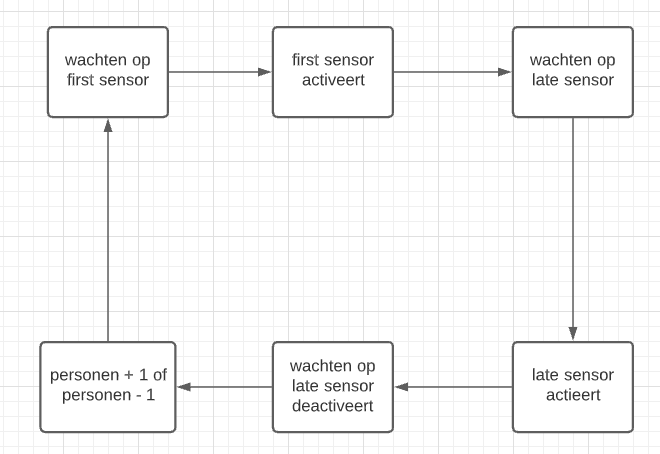
**Onderzoek via embedded systems**

Ik heb via het vak embedded systems onderzoek gedaan naar state gedrag. Hier heb ik gekeken waar je state gedrag precies voor kan gebruiken en wanneer het niet nodig is. Hierin vond ik dat het goed te gebruiken is als het proces maar in een state mag zijn en niet als het programma in meerdere states kan zijn. Omdat in mijn situatie het programma maar in een state kan zijn werkt state gedrag heel goed om te zorgen dat je mensen uit elkaar kan houden.

## Lab

**State gedrag uitleg**

Eerst wacht de arduino of er een sensor een waarde onder de 1500mm meet als dit zo is gaat hij naar de state: first sensor activated. Hier wacht op de sensor die nog niet af gegaan is (deze sensor noem ik late sensor). Als de late sensor afgaat dan wacht hij tot deze niet meer meting onder de 1500 mm meet en dan word er pas een persoon bij geteld. Er is 1 probleem hiermee als iemand bijvoorbeeld sensor 1 activeert en wegloopt zonder sensor 2 te activeren werkt het systeem omgedraaid vanaf dan (als iemand dan binnen loopt activeert hij eerst de late sensor en dan de first sensor dus telt het systeem hierdoor omgedraaid).



Figuur State Diagram

**Stage gedrag testing**

Ik heb verschillende tests gedaan met en zonder states. Het bleek zonder states dat er soms meer bij word geteld dan er werkelijk bij geteld moet worden, met states gebeurd dit bijna nooit. Je ziet bij figuur 2 dat het ook goed werkt en er steeds maar 1 bij of afgaat.

## Conclusie

Om mensen goed uit elkaar te houden moet je twee sensoren langs elkaar zetten die met state gedrag werken. En omdat je twee sensoren gebruikt kan je weten welke richting de persoon heen gaat.

# Onderzoek Deelvraag 3

**Deelvraag 3:** Op welke richting moet de sensor wijzen voor de meest foutloze metingen?

## Field

**Probleem analyse voor gekanteld richting 90°**Als de 2 sensoren langs elkaar 90 graden van de muur (dus recht naar beneden) staan dan komen er problemen dat ze heel makkelijk beide aan kunnen staan.

**Probleem analyse voor gekanteld richting 220°**

**Probleem analyse voor gekanteld richting 250°**

# Eind Conclusie

Bronnenlijst

|  |  |  |
| --- | --- | --- |
| **Auteurs** | **Laatst gewijzigd** | **Geraadpleegd van** |
| People Counting Solutions | Niet gegeven | https://eu.beasensors.com/nl/segment/people-counting-solutions/personentelling/ |
| Redactie KVK | 11 aug 2020 | https://www.kvk.nl/advies-en-informatie/veiligzakendoen/klantentellers-efficientie-versus-privacy/ |
| Canvas | N.V.T. | https://fhict.instructure.com/ |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |