lectoraat opdrcaht 1

Automatische hand-oog kalibratie

Minoropdracht Fabriek van de Toekomst

**Versiebeheer**

|  |  |  |  |
| --- | --- | --- | --- |
| Versie | Datum | Omschrijving | Opmerkingen |
| 0.0 | 20-12-2024 | Uitleg hand-oog kalibratie | Document aangemaakt (Martijn Walter) |

Inhoud

[**H1 Inleiding** 4](#_Toc188622139)

[**H2 Theorie** 5](#_Toc188622140)

[2.1 onze situatie 5](#_Toc188622141)

[2.2 pingpongbal 6](#_Toc188622142)

[2.3 kalibratiemethode en uitkomst 6](#_Toc188622143)

[**H3 software applicaties** 7](#_Toc188622144)

[3.1 HalconCpp 7](#_Toc188622145)

[3.2 Boost-Beast 7](#_Toc188622146)

[3.3 K4A.sdk 7](#_Toc188622147)

[**H4 aanpassingen aan code** 8](#_Toc188622148)

[H4.1 kalibratie aanpassen 8](#_Toc188622149)

[H4.2 poses aanpassen 9](#_Toc188622150)

### H1 Inleiding

Het bedrijf Talen Robotics is een bedrijf dat robots ontwikkelt die flexibel inzetbaar zijn voor de houtbewerkings-industrie, waarmee klanten hun productieproces kunnen optimaliseren. Dit kan bijvoorbeeld gaan om de hoeveelheid hout die kan worden verwerkt of de snelheid waarmee opdrachten kunnen worden uitgevoerd.

Talen Robotics wil een systeem ontwikkelen dat houten planken met verschillende vormen (zoals rechthoeken, cirkels, ellipsen en planken met afgeronde hoeken) kan schuren. Het oppervlak en de zijkanten moeten worden gestuurd door een cobot. Om dit te doen is het belangrijk dat de machine de vorm van de plaat herkent en een schuur pad volgt dat is gecreëerd door een visionsysteem.

Een probleem van deze applicatie is dat de cobot moet kunnen bewegen op basis van het visionsysteem. Om dit professioneel te realiseren is een hand-oog kalibratie onmisbaar, in dit bestand wordt de automatische kalibratie toegelicht hoe deze werkt en hoe deze in toekomstige lectoraat projecten toegepast kan worden.

In hoofdstuk 2 wordt uitgelegd hoe de deze manier van kalibreren werkt met de theorie erachter.   
In hoofdstuk 3 wordt uitgelegd wat voor software applicaties er benodigd zijn om dit te laten werken.  
in hoofdstuk 4 wordt uitgelegd wat er gedaan moet worden met de cobot en welke data je de nodig hebt van de cobot.

### H2 Theorie

In dit hoofdstuk wordt de gebruikte manier van kalibreren toegelicht en hoe deze werkt. In dit document wordt geen vergelijking gedaan tussen verschillende kalibratiemethoden hiervoor bekijk de onderzoeksrapport van groepje Lectoraat 1 Talen Robotics.

### 2.1 onze situatie

Als eerste is het belangrijk om te weten waarom en wanneer je een kalibratie nodig hebten wat je uit de kalibratie haalt. Je hebt een kalibratie nodig op het moment als je wilt dat de cobot samen met een camera werkt voor bijvoorbeeld pick-and-place applicaties. Wat je uiteindelijk eruit krijgt is de positie van de camera ten opzichte van het werkvlak van de cobot. Dit houd in als je de cobot laat bewegen naar de coördinaten die uit de kalibratie zit de cobot exact 6

Je hebt om een Hand oog kalibratie te maken moet je een aantal dingen nodig. Als eerste moet er een vaste camera locatie hebben t.o.v. de cobot. In ons geval staat de camera met een 90 graden hoek op de lastafel op een hoogte van ongeveer 80 cm. Hieronder een schets van hoe de situatie eruit ziet

Afbeelding met schets, tekening, lijn, diagram

Automatisch gegenereerde beschrijving

Figuur schets TR demo

|  |  |
| --- | --- |
| 1 | Camera: azure kinect dk |
| 2 | Cobot: UR 10 cobot |
| 3 | Werkvlak |

De manier die gebruikt is en bescheren wordt in dit document werkt met deze opstelling, maar de manier werkt in theorie in alle situaties zolang de camera een duidelijk zicht heeft op de cobot flange en het werkvlak.

### 2.2 pingpongbal

Er zijn veel verschillende manieren om een hand-oog kalibratie uit te voeren. De manier die wordt gebruikt tijdens dit project en hier beschreven wordt maakt gebruik van surface-based-matching m.b.v. Mvtec Halcon. Deze methode werkt doordat je verschillende poses vergelijkt vanaf de camera en de cobot. Om dit te doen moet de cobot een pingpongbal vasthebben dit kan gedaan worden door bijvoorbeeld een vacuum gripper.

### 2.3 kalibratiemethode en uitkomst

De kalibratie wordt uitgevoerd door het te vergelijken van de verschillende cobot poses met de camera poses door een functie in Halcon. Hieruit komt een matrix waar de kalibratie instaat dit ziet er als volgt uit

De 4de in de array is de x waarde, de 8ste is de y waarde en de 12de is de z waarde.

Deze z, y en z coordinaten is de positie van de camera in het cobot coördinatenstelsel.

### H3 software applicaties

Om de code te gebruiken moet je een paar verschillende library’s downloaden en installeren in dit hoofdstuk staatwelke je nodig hebt en waarom.

### 3.1 HalconCpp

Halcon is een belangrijk deel van de code deze library zorgt ervoor dat de gemaakt 3d pointcloud gefilterd wordt en wordt in de pointcloud de bal gezocht. Halcon doet ook de daadwerkelijke kalibratie berekening met alle waargenomen informatie.

### 3.2 Boost-Beast

Boost-Beast is nodig om een socket connectie te maken met de cobot om hem naar de voorgeprogrammeerde punten te sturen.

### 3.3 K4A.sdk

Deze library is nodig om de microsoft azure kinect 3d camera aan te sturen. met deze library kan je een pointcloud maken die gemaakt moet worden bij elke pose. Op het moment dat je een andere camera gebruikt moet je de sdk van die camera toepassen en de camera code aanpassen in de code.

### H4 aanpassingen aan code

In dit hoofdstuk wordt uitgelegd welke aanpassingen er gedaan moeten worden aan de code afhankelijk van jou situatie

### H4.1 kalibratie aanpassen

Het enige dat er aangepast moet worden aan de code is de filtering dit is namelijk afhankelijk van de situatie die jij nodig hebt hieronder leg ik uit hoe je deze het beste kan aanpassen en hoe je aan de juiste waardes komt.

Afbeelding met tekst, schermopname, Lettertype, software

Automatisch gegenereerde beschrijving

In dit stukje code wordt de pointcloud gefilterd naar een bruikbare scene. Regel 378 tot 380 filtert de scene op x, y en z waarden ik adviseer om een pointcloud te maken van de opstelling en in CloudCompare (software) de punten aan te klikken die van belang zijn en deze hier invullen.

Regel 382 tot 387 filtert de scene nog verder. De eerste functie kijkt naar de afstand tussen de punten en verbind in dit geval de punten die 0.004m van elkaar af zitten. Dit wordt gebruikt omdat de pingpongbal een kleine afstand had van de cobot flange dus kan hij de pointcloud van alleen de pingpongbal eruit halen

Afbeelding met plastic, overdekt, gereedschap, fles

Automatisch gegenereerde beschrijving

Deze waarde moet je misschien aanpassen afhankelijk van hoever de pingpongbal is van de arm zie bovenstaand figuur voor onze situatie.

Vervolgens regel 384 filtert deze afgebroken stukken pointcoud op de hoeveelheid punten. Afhankelijk van de resolutie en afstand van de bal heeft deze altijd een pointcloud dikte van ongeveer 165 en 430 punten. Dit kan variëren dus moet je een beetje “rondspelen” met deze 2 waardes. Daarna is er een if functie om de foutieve poinclouds weg te halen. Dit kan met een if omdat de pointcloud van de pingpongbal altijd als eerste in de array staat omdat deze het dichstbij de camera is en dus wordt als eerste opgeslagen.

Voor deze filtering waarden is het soms best lastig om deze te verkrijgen hiervoor raad ik halcon editor aan. Aan het einde van de filtering moet het er zo uitzien waardoor ie dan de bal kan zien en correct in de scene te plaatsen.

Afbeelding met ruimte, cirkel, Hemellichaam, Universum

Automatisch gegenereerde beschrijvingAfbeelding met tekenfilm, schermopname, kunst

Automatisch gegenereerde beschrijving

Ik heb de testcode toegevoegd van mijn Halcon project

### H4.2 poses aanpassen

Voor de kalibratie is het belangrijk dat de pingpongbal recht naar de camera wijst dit is bij onze situatie zo voor deze posen maar misschien heb jij wel een andere camera opstelling. Afbeelding met tekst, schermopname, Lettertype, software

Automatisch gegenereerde beschrijving

Dit is het stukje code dat je moet aanpassen met de poses. Ik raad aan om deze poses aan te maken om de cobot te bewegin binnen het zichtveld van de camera waarbij de pingpongbal wijst naar de camera. Als te tevreden bent met 1 pose ga je naar de MOVE tap en selecteer je linksmidden “base” daarna kan jee de waarde invullen in de array op de volgende manier

[X, Y, Z, RX, RY, RZ]

Hierbij zijn RX, RY en RZ in radialen.