

Ontwerp en Implementatie van een Fysieke VR-game met Motion Capture Technologie

Portugaels Safwaan
Universiteit Hasselt
Hasselt, België
safwaan.portugaels@student.uhasselt.be

Abstract—Deze bachelorproef presenteert een systeem dat lichaamsbewegingen registreert en inzet als besturingsmethode voor een computerspel. In tegenstelling tot traditionele games wordt dit spel niet op een scherm weergegeven, maar geprojecteerd op de vloer. De speler bestuurt het spel met een fysieke tak in plaats van met een muis of toetsenbord. Het onderzoek richt zich op het ontwerp, de implementatie en de evaluatie van deze alternatieve spelvorm. Doelstelling is het ontwikkelen van een spel dat op een toegankelijke en aantrekkelijke manier fysieke activiteit stimuleert. Het systeem is geëvalueerd op zowel technische prestaties als gebruikservaring.

I. INLEIDING

Langdurig zitten heeft aantoonbaar negatieve effecten op de gezondheid, waaronder een verhoogd risico op cardiovasculaire aandoeningen en overgewicht [1]. Dit vormt een specifiek probleem bij gamers, die vaak gedurende lange tijd inactief blijven tijdens het spelen. Het ontwikkelen van methoden die fysieke activiteit tijdens het gamen stimuleren, is daarom van groot belang. Een veelbelovende technologie in dit verband is motion capture (MoCap), waarmee lichaamsbewegingen nauwkeurig kunnen worden gevolgd en als invoer voor games kunnen dienen.

MoCap-systemen zijn doorgaans duur en omvangrijk, wat de toegankelijkheid voor een breed publiek beperkt. Ze zijn echter bijzonder geschikt voor het ontwikkelen en testen van spelconcepten waarbij fysieke beweging centraal staat. Voor dit project werd gebruikgemaakt van een MoCap-systeem bestaande uit elf Qualisys-camera's in combinatie met de software Qualisys Track Manager (QTM). Deze software detecteert bewegingen van objecten met reflecterende markers en bepaalt hun positie en oriëntatie. Voor de besturing van het spel werd een houten tak voorzien van drie markers.

De spelsoftware werd ontwikkeld in C++ met behulp van de SDL-bibliotheek voor grafische weergave. De communicatie met het MoCap-systeem verliep via de `qualisys_cpp_SDK`. Het project onderzoekt de mogelijkheden van spelbesturing met een beperkt aantal markers en de impact van projectie op de vloer in plaats van een traditioneel scherm. Deze aanpak biedt de speler meer bewegingsvrijheid en stimuleert fysieke activiteit tijdens het gamen.

II. FUNGODS

De opdracht voor deze bachelorproef bestond uit het ontwikkelen van een tower defense-spel. In dit type spel



Fig. 1. De tak met drie tracking markers

verdedigt de speler een basis tegen vijanden door strategisch verdedigingsstructuren te plaatsen. Voor dit project werd gekozen voor een thema rond schimmels, wat zowel visueel als conceptueel een onderscheidend karakter toevoegt.

Het spel kreeg de naam *Fungods*, een samentrekking van *Fungus* (schimmel) en *Gods* (goden). De speler treedt op als beschermer en beheerder van de schimmels. Deze thematische keuze sluit aan bij de biologische inspiratie en draagt bij aan een samenhangende spelervaring.

A. Inputmethode

Voor het besturen van games bestaan diverse invoermethoden, elk met specifieke voor- en nadelen. In dit project is gekozen voor een motion capture-systeem waarbij een fysieke tak, voorzien van reflecterende markers, als invoermiddel wordt gebruikt. Elk object met ten minste drie reflecterende markers is geschikt voor tracking. De keuze voor een tak als inputmethode sluit aan bij het natuurlijke thema van *Fungods*, en de knopen van de tak bieden praktische bevestigingspunten

voor de tracking markers. Deze benadering resulteert in een unieke en thematisch passende interactiemethode.

Traditionele methoden zoals toetsenbord en muis zijn toegankelijk en nauwkeurig, maar beperken fysieke beweging. Gamecontrollers zijn ergonomisch, maar blijven beperkt tot handbewegingen. VR-controllers en handtracking bieden wel volledige bewegingsvrijheid, maar vereisen dure en complexe hardware.

Motion capture vormt een evenwichtige middenweg: het maakt gebruik van natuurlijke bewegingen en biedt meer vrijheid dan conventionele methoden. Voor ontwikkelings- en foutopsporingsdoeleinden is ook besturing via toetsenbord en muis beschikbaar gemaakt.

Aanvankelijk was het de bedoeling dat de speler de tak draaide voor besturing. Tests toonden echter twee nadelen: het object verdween geregeld uit het camerazicht en het bewegen boven het hoofd bleek onveilig in een ruimte met kwetsbare apparatuur. Daarom werd gekozen voor een aangepaste methode waarbij de tak dicht bij de grond wordt gebruikt. Deze aanpak bleek betrouwbaarder en veiliger. Ondanks beperkingen vormt de MoCap-interface een effectieve en innovatieve besturingsmethode die bijzonder geschikt is voor experimenteel onderzoek.

B. Gameplay

Het doel in *Fungods* is het tegenhouden van bacteriën. Dit gebeurt door schimmels te plaatsen die automatisch aanvallen via sporen of mycelium. Na plaatsing verschijnt het schimmellichaam en begint het mycelium zich uit te breiden, deze hebben de rol van sensor: bacteriën die zich binnen het bereik van het mycelium bevinden, worden aangevallen door de schimmel, waarbij de aard van de aanval afhangt van het specifieke schimmeltype.

Met de tak kan de speler verschillende acties uitvoeren, zoals:

- Tikken op het schimmelselectiepaneel linksboven om een andere schimmel te selecteren.
- Tikken op het middelste schimmelicoon om de volgende golf of het volgende level te starten.
- Tikken op het speelveld om een schimmel te plaatsen op de aangeraakte locatie.
- De tak boven de schimmels houden om spontane sporenvorming te activeren.
- De tak boven de sporen houden om deze aan te trekken, vergelijkbaar met een wervelstorm.

Levels zijn opgebouwd uit meerdere opeenvolgende golven van bacteriën (Figure 2). Een nieuwe golf wordt pas geactiveerd nadat alle vijanden van de voorgaande golf zijn geëlimineerd. De speler behaalt overwinning bij succesvolle voltooiing van alle golven binnen een level, terwijl het verlies wordt bepaald door de vernietiging van de centrale schimmel.

C. Aanpasbaarheid van Entiteiten

Het spel bevat diverse soorten schimmels, bacteriën en sporen. De eigenschappen van elk entiteitstype kunnen worden aangepast via een extern configuratiebestand

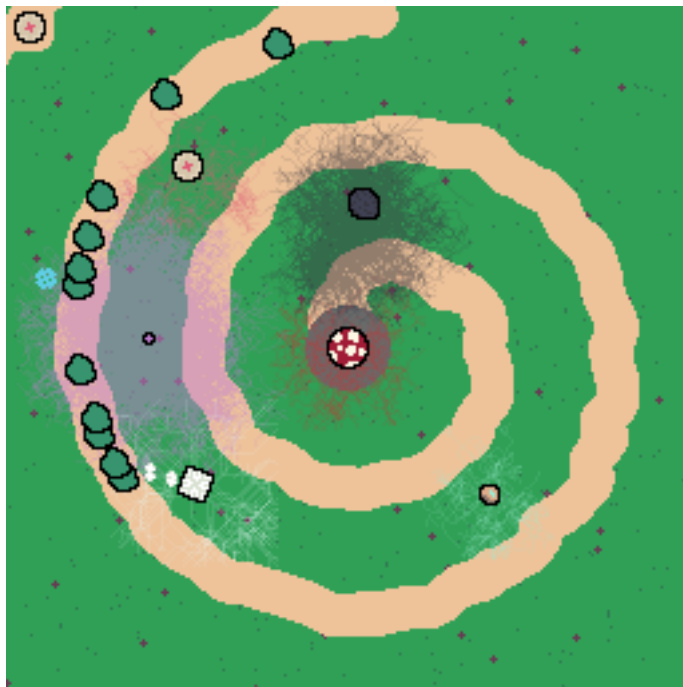


Fig. 2. Het spel in actie

genaamd `stats.fge` (Fungus Gods Entity). In dit bestand wordt elke eigenschap per regel vastgelegd in het formaat `[Prefix] [Waarde]`. Eigenschappen die niet expliciet worden opgegeven, krijgen een standaardwaarde toegewezen. In sommige gevallen wordt deze standaardwaarde automatisch bepaald aan de hand van de afbeelding die aan het object is gekoppeld.

De visuele weergave van de spelobjecten wordt geladen uit afbeeldingsbestanden zoals `S0.bmp`, uitgevoerd in een pixelart-stijl die goed aansluit bij de visuele identiteit en sfeer van het spel. Hoewel schimmels, bacteriën en sporen een gemeenschappelijke set eigenschappen delen, beschikt elk type over unieke kenmerken. Onderstaande tabel geeft een overzicht van zowel de algemene als type-specifieke eigenschappen.

```

stats.fgl
File Edit View
S 6
A 2
D 2000
SPORE basic
M_SIZE 28
M_BRANCH 12
M_TURN 2
M_COLOR 157:235:209

```

Fig. 3. Voorbeeld van een `stats.fgl` bestand voor een schimmel

Prefix	Type	Functie
Alles		
HP	int	maximale levens
DMG	int	schade
W	float	breedte (auto)
H	float	hoogte (auto)
R	float	straal (auto)
S	float	maximale snelheid
A	float	versnelling
DA	float	vertraging
D	int	herlaadticks
Schimmels		
SPORE	string	naam van sporen
M_SIZE	int	maximale myceliumgrootte
M_BRANCH	int	kans op myceliumvertakking
M_TURN	int	kans op myceliumdraaiing
M_COLOR	int[3]	kleur van mycelium
Sporen		
T	bool	automatisch volgen
L	int	maximale ticks voor sterven
Bacteriën		
P	int	score bij dood

D. Aanpasbaarheid van Levels

Net als de entiteiten worden levels gegenereerd op basis van externe configuratiebestanden. Het gaat hierbij om twee FGL-bestanden (*Fungus Gods Level*): `paths.fgl` en `waves.fgl`. Voor de visuele weergave van de levels wordt dezelfde sprite (`S0.bmp`) gebruikt, wat zorgt voor visuele consistentie binnen het spel.

Het bestand `paths.fgl` definieert het traject waarlangs bacteriën zich bewegen. Elke regel beschrijft een coördinaat in het formaat:

```
L [x]:[y]
```

waarbij `[x]` en `[y]` de positie aangeven van de punten die samen het pad vormen.

Het bestand `waves.fgl` bepaalt de opbouw en volgorde van vijandelijke golven. Elke regel heeft een specifieke functie:

- `S [naam]` – Voegt een bacterie van type `[naam]` toe aan de volgende golf.
- `D [ticks]` – Voegt een wachttijd van `[ticks]` iteraties toe vóór de plaatsing van de volgende bacterie. Als deze regel ontbreekt, volgt de plaatsing direct.
- `W _` – Geeft het begin van een nieuwe golf aan.

Door deze configuratiebestanden buiten de broncode te houden, kunnen levels eenvoudig worden aangepast of nieuwe levels worden toegevoegd, zonder wijzigingen aan de programmabestanden. Deze scheiding van data en logica vergroot de flexibiliteit en uitbreidbaarheid van het spel.

III. SOFTWARE

A. Gebruikte Technologie

Voor de ontwikkeling van dit project werd gekozen voor programmeren in C++. Deze programmeertaal is krachtig, objectgeoriënteerd en geschikt voor toepassingen waarbij prestaties en flexibiliteit belangrijk zijn.

Een belangrijk uitgangspunt was modulariteit: spelcomponenten moesten eenvoudig aangepast kunnen worden zonder de broncode te wijzigen. Dit werd gerealiseerd door configuraties onder te brengen in externe bestanden (zoals `stats.fgl`), die bij uitvoering worden ingelezen.

Voor de grafische weergave werd gebruikgemaakt van de Simple DirectMedia Layer (SDL). SDL biedt ondersteuning voor beeld, geluid en invoer. In de beginfase werd SDL ook gebruikt voor inputverwerking, maar in de definitieve versie diende het uitsluitend voor grafische projectie. SDL werd bewust gekozen boven alternatieven zoals Qt, omdat het meer controle geeft over low-level bewerkingen en beter aansluit bij de vereisten van dit project.

Voor de communicatie met het Qualisys Track Manager (QTM)-systeem werd de officiële Qualisys Software Development Kit (SDK) toegepast. Deze bibliotheek, voorzien van uitgebreide documentatie, maakte een robuuste koppeling met het MoCap-systeem mogelijk.

B. Codearchitectuur en Stijl

De code is opgebouwd met het oog op herbruikbaarheid en onderhoudbaarheid. Dit werd gerealiseerd door toepassing van objectgeoriënteerd programmeren: elke klasse heeft een afgebakende verantwoordelijkheid.

Belangrijke richtlijnen waren:

- Klassen met duidelijke namen en enkelvoudige verantwoordelijkheden.
- Gebruik van overerving om hergebruik en uitbreidbaarheid te bevorderen.
- Abstracte klassen voor gemeenschappelijke eigenschappen en consistentie.
- Vermijden van cirkelvormige afhankelijkheden bij het compileren.
- Systematische indeling van bestanden in mappen naar functie.

- Korte functies die elk één specifieke taak uitvoeren.

Hoewel verdere optimalisaties mogelijk zijn (bijvoorbeeld door meer gebruik van contracts of verdere opsplitsing van functies), is de huidige structuur overzichtelijk en logisch.

C. Implementatie

De kern van de uitvoering bevindt zich in de SDL-loop, die via `SDL_AppIterate()` wordt aangestuurd. In elke iteratie voert de engine de volgende stappen uit:

- Updaten van het level.
- Verwerken van alle actieve entiteiten (schimmels, bacteriën en sporen).
- Simuleren van wind voor sporen.
- Renderen van de spelwereld.
- Wachten tot de volgende iteratie.

Coördinaten worden beheerd door een `Location`-object, dat ook hulpfuncties bevat zoals afstandsrekening. Voor efficiëntie is er een methode die de kwadraatafstand retourneert zonder vierkantswortel, wat rekenlast beperkt.

Het mycelium start als een enkele pixel en groeit volgens probabilistische regels, zoals *splitkans* en *buigkans*. Hiermee kunnen verschillende groeipatronen worden gesimuleerd (Figure 4).

Bacteriën volgen een vooraf bepaald traject bestaande uit een reeks opeenvolgende punten en bewegen van punt naar punt totdat het eindpunt is bereikt, waarna zij zich naar de centrale schimmel verplaatsen.

De positie van de reflecterende markers op de tak wordt geregistreerd door het Qualisys-systeem. Deze ruwe meetgegevens worden via de Qualisys C++ SDK doorgegeven aan het MoCap-object, waar de coördinaten worden getransformeerd naar het coördinatensysteem van het spel. Op basis van de hoogte van de tak wordt vervolgens een event gegenereerd dat aan de game-engine wordt doorgegeven, welke de invoer interpreteert en de spelbesturing aanpast conform de positie van de tak.

Sprites worden beheerd door het `SpriteMaster`-object, dat duplicatie voorkomt door afbeeldingen slechts één keer in het geheugen te laden. Entiteiten verwijzen naar sprites via een index, wat efficiënt renderen mogelijk maakt.

D. Optimalisaties

C++ en SDL leveren standaard al goede prestaties, maar extra optimalisaties waren noodzakelijk voor schaalbaarheid.

Een belangrijke ingreep was het gebruik van multithreading: het spel en het MoCap-systeem draaien in afzonderlijke threads en communiceren via events. Hierdoor hoeft het spel niet te wachten op trackingdata en wordt optimaal gebruik gemaakt van meerdere processorkernen.

Ook binnen het mycelium werd efficiëntie bereikt. Elke pixel van het mycelium kan fungeren als sensor, maar het voortdurend controleren van alle pixels zou te belastend zijn. Daarom werd slechts een deel geactiveerd als sensor, wat het aantal berekeningen sterk verminderde zonder verlies aan functionaliteit.

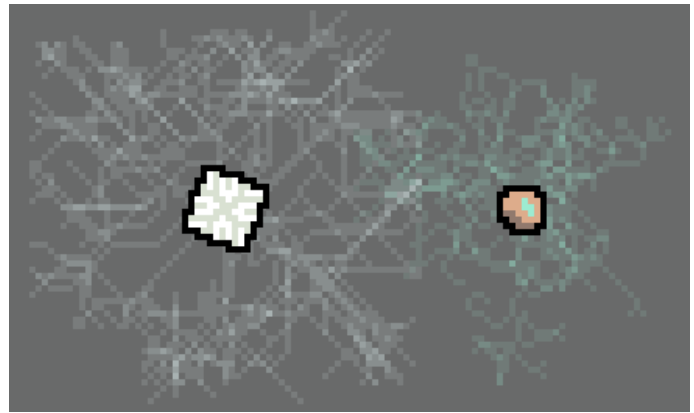


Fig. 4. Voorbeelden van groeiende myceliumstructuren

Dankzij dergelijke optimalisaties behaalt het spel framerate's van 333 tot meer dan 1000 FPS, wat ruim boven de vereiste 24 FPS ligt. Dit waarborgt een vloeiende spelervaring, zelfs op minder krachtige systemen.

IV. MOTION CAPTURE

A. Synchronisatie en Kalibratie

Bij het opzetten van een motion capture (MoCap) systeem is het cruciaal dat de Qualisys Miquis-camera's nauwkeurig worden gesynchroniseerd met elkaar en met externe systemen. De camera's ondersteunen diverse synchronisatieopties, zoals SMPTE, GENLOCK, IRIG en TTL, die via een sync-unit kunnen worden ontvangen. Voor een stabiele synchronisatie moeten de interne klokken van de camera's worden gekalibreerd en vergrendeld op het inkomende signaal voordat de opname start. Bovendien kunnen meerdere Miquis-camera's via een daisy-chain configuratie worden verbonden, waardoor zowel datatransmissie als stroomvoorziening over één kabel mogelijk is [3].

Om 3D-coördinaten te verkrijgen binnen Qualisys Track Manager (QTM) is een kalibratiestap vereist. Onder normale omstandigheden wordt dit uitgevoerd met behulp van een kalibratieset; door het ontbreken van een L-frame werd de kalibratie handmatig uitgevoerd door markers nauwkeurig op de vloer te positioneren. Na meerdere pogingen kon een correcte kalibratie worden gerealiseerd.

Omdat het systeem een projectie combineert met het MoCap-systeem, is het essentieel om de positie van het geprojecteerde beeld in de fysieke ruimte nauwkeurig te bepalen. Hiervoor is, naast de QTM-kalibratie, een tweede kalibratiestap vereist die momenteel hardgecodeerd is.

B. Verwerking van MoCap-invoer

Het MoCap-systeem vormde het hart van dit project. Hiervoor werd gebruikgemaakt van Qualisys Miquis-camera's en de Qualisys Track Manager (QTM)-software. Een fysieke tak met drie reflecterende markers fungeerde als invoer.

Om de Motion Capture (MoCap)-gegevens als invoer in het systeem te gebruiken, moet het Qualisys Track Manager (QTM) actief zijn en een opnameproces uitvoeren. Alleen

wanneer QTM in deze staat verkeert, kunnen realtime trackingdata worden verwerkt door het spel.

De integratie met de Qualisys C++ SDK verliep aanvankelijk moeizaam, maar werd later succesvol gerealiseerd. De software maakt gebruik van 6 Degrees of Freedom (6DOF)-objecten [2], wat voordelen biedt:

- Losse markers die niet tot een object behoren, worden genegeerd.
- Objecten kunnen handmatig gedefinieerd en als invoermiddel gebruikt worden.
- 6DOF-objecten zijn eenvoudig aan te maken en beheren in QTM.
- Elk object levert zowel positie in x, y, z als een 3×3 rotatiematrix, voldoende voor nauwkeurige besturing.

V. OBJECTIEVE VALIDATIE

De validatie bestond uit technische metingen en functionele tests.

1) Technische metingen:

- **Framerate:** het spel draaide stabiel op minimaal 24 FPS, zonder haperingen, zelfs tijdens intensieve situaties.
- **Trackingnauwkeurigheid:** bij interactie met de tak bedroeg de maximale afwijking van de beoogde projectiepositie ongeveer 5 cm. Gezien de schaal van het speelveld werd dit verschil als acceptabel beschouwd.

2) *Functionele controle:* Alle kernfunctionaliteiten (selecteren en plaatsen van schimmels, starten van golven, aantrekken van sporen en interactie met bacteriën) werkten conform het ontwerp. De motion capture-input functioneerde stabiel en zonder onderbrekingen.

Hieruit kan worden geconcludeerd dat het systeem zowel technisch betrouwbaar als functioneel bruikbaar is en de doelstelling – het stimuleren van fysieke activiteit via gaming – effectief ondersteunt.

A. Fysieke Stimulatie

Door de omvang van het speelveld wordt de speler actief gestimuleerd tot fysieke verplaatsing, bijvoorbeeld om sporen tijdig naar de bacteriën te werpen of om schimmels efficiënt te selecteren en te plaatsen. De snelheid van de bacteriën is zorgvuldig afgestemd: deze is hoog genoeg om snelle reacties van de speler te vereisen, maar niet zo hoog dat het verdedigen onmogelijk wordt. Dergelijke ontwerpkeuzes vergroten de betrokkenheid van de speler bij het spel en bevorderen continue beweging, wat bijdraagt aan een fysiek stimulerende spelervaring. Hiermee vormt het spel een doelbewust ontworpen, gezonder alternatief voor traditionele, voornamelijk zittende gameplay, waarbij cognitieve uitdagingen worden gecombineerd met fysieke activiteit.

B. Robuustheidstesten

Om de stabiliteit van de software te beoordelen, werden diverse edge cases getest zoals:

- Het ontbreken of loskoppelen van het Qualisys-systeem.
- Plaatsing van entiteiten buiten het speelveld.

- Onrealistische waarden in configuratiebestanden (zeer groot, nul of negatief).
- Laden van een uitzonderlijk groot aantal objecten.
- Onverwachte acties via toetsenbord of muis.

VI. TOEKOMSTIGE UITBREIDINGEN

Het project biedt diverse mogelijkheden voor verdere ontwikkeling.

Een belangrijke uitbreiding is een lokale multiplayermodus waarin twee spelers elkaar strategisch kunnen uitdagen. Hiervoor is ondersteuning voor meerdere invoertakken vereist.

Het toevoegen van een startscherm zou de gebruikservaring verbeteren, doordat spelers vanaf één centrale interface een level kunnen selecteren of instellingen kunnen aanpassen.

Op gameplay-niveau kunnen tevens oneindige levels worden geïmplementeerd, waarin de moeilijkheidsgraad progressief toeneemt; deze levels zouden dan ook selecteerbaar zijn via het startscherm.

De handmatige kalibratie van de MoCap-input ten opzichte van het geprojecteerde scherm kan worden vereenvoudigd door een speloptie te implementeren waarbij de speler op bepaalde punten op de projectie moet tikken met de tak, waardoor het systeem automatisch de kalibratie kan uitvoeren.

Daarnaast kan muziek geïntegreerd worden die gebaseerd is op elektrische signalen van echte schimmels. Dit zou het biologische thema versterken en een unieke sfeer toevoegen. Geluidseffecten voor acties zoals het planten van schimmels of het verslaan van bacteriën kunnen de beleving verder verdiepen.

Ook animaties zijn een waardevolle toevoeging: bijvoorbeeld het realistisch groeien van mycelium of het visueel weergeven van aanvallen. Dit zou bijdragen aan een dynamischer en aantrekkelijker geheel.

CONCLUSIES

De ontwikkeling van het spel en de bijbehorende software is in ruime mate succesvol verlopen. Hoewel er nog ruimte is voor verdere optimalisatie, functioneren de kerncomponenten van het MoCap-systeem betrouwbaar en voldoen ze aan de gestelde functionele eisen.

De actieve aard van het spel bevordert fysieke betrokkenheid, waardoor het concept een overtuigend alternatief biedt voor conventionele, sedentair gerichte gameplay. De praktische ervaring met Qualisys-hardware en -software leverde waardevolle inzichten op, waarbij het gebruik van één 6DOF-object voldeed voor een stabiele en nauwkeurige besturing.

Gezien het aantoonbare potentieel van het concept vormt dit project een solide basis voor toekomstige uitbreidingen en verfijningen, zowel op technisch vlak als in termen van spelervaring, waarmee de relevantie en toepasbaarheid verder kan worden vergroot.

DANKWOORD

De auteur wenst de volgende personen te bedanken:

- Prof. dr. Kris Luyten voor de begeleiding tijdens dit project.

- Prof. dr. Gustavo Roveló Ruiz voor het verstrekken van de Qualisys-licentie.
- Mevr. Kristel Renaers en dr. Danny Leen voor hun hulp bij het vinden en installeren van de projector.
- ChatGPT voor ondersteuning bij taalkundige correcties.

REFERENCES

- [1] A. Choobineh, *Adverse Effects of Prolonged Sitting Behavior on the General Health of Office Workers*, Shiraz University of Medical Sciences, juni 2017.
- [2] E. Schoonderwaldt en D. Thompson, Learn about 6DOF, Qualisys, mei 2016.
- [3] Qualisys, Miquis Cameras, Qualisys, 2025.