

TECHNISCHE EINORDNUNG

- **Technik:** Messerwerfen
- **Geräte:** Oculus Quest
- **Interaktionstechnik:** Zielen & Werfen
- **Herausforderung:** Präzision & Entfernung

AUFGABENSTELLUNG

Wie unterstützt man Spieler und Spielerinnen in VR beim Zielen, sei es nun mit Wurfobjekten oder mit Schusswaffen? In herkömmlichen Videospielen gibt es mittlerweile als Accessibility-Feature und auch für niedrige Schwierigkeitsstufen sogenanntes Aim Assist, also Hilfe beim Zielen. Hierbei wird das Fadenkreuz des Spielers automatisch auf das nächste oder das anvisierte Ziel gelenkt. In VR ist dies bisher so nicht umgesetzt, da meist nicht mit einem Joystick oder der Maus gezielt wird, sondern durch tatsächliche Bewegung der Arme. Es soll im Umfang dieser Arbeit ein Aim Assist System für Virtual Reality entwickelt werden, das das Zielen und Werfen in einer virtuellen Umgebung erleichtert.

TECHNISCHE HERAUSFORDERUNG

Aiming Assistance Funktionen in Videospielen zeichnen sich beispielsweise dadurch aus, dass sie das Fadenkreuz automatisch an das Ziel annähern, sogenannter "bullet magnetism" [2]. Es soll jedoch nicht gänzlich automatisch passieren, sondern lediglich assistieren.

Die Herausforderung besteht darin, das Zielen, das durch das Gesetz von Fitts immer schwieriger wird und längere Zeit in Anspruch nimmt je weiter weg und je kleiner das Ziel ist [1], zu vereinfachen, ohne dem Spielenden die gesamte Arbeit abzunehmen. In Virtual Reality ist das Zielen vor allem dadurch beeinträchtigt, dass das Gewicht, die Größe und die Handhabung des Controllers nicht mit denen des virtuellen Wurfobjekts übereinstimmt. Dies erschwert das Einschätzen der Wurfkraft und des Punktes, an dem losgelassen werden sollte. Auch fehlt ein Fadenkreuz, das in herkömmlichen Videospielen beim Zielen hilft.

Eine Aim Assist Funktionalität in VR muss diesen Probleme somit entgegenwirken und beispielsweise die Richtung und Wurfkraft anpassen, damit das Ziel getroffen wird. Vor allem bewegte Ziele sind nur schwer zu treffen, da die Wurfgeschwindigkeit nicht immer präzise vom Controller umgesetzt wird. Auch das Ausholen kann zu Problemen beim Tracking führen, wenn das VR System keine Tracking-Stations verwendet, sondern auf Kameras am HMD angewiesen ist.



Abbildung 1: Baseline Interaktion Messerwerfen

LÖSUNGSANSATZ

Wie auch beim Werfen in der Realität, soll der zweite Arm genutzt werden, um beim Zielen zu helfen. Dieser dient normalerweise dazu, das ungefähre Ziel des Projektils anzuvisieren, meist mit der Hand ausgestreckt in Richtung Ziel.

Dies soll ähnlich in VR umgesetzt werden. Hier soll die linke Hand durch einen Raycast dargestellt werden, mit dem bestimmte Ziele in der Szene anvisiert werden können. Wird ein valides Ziel anvisiert, kann es als gewünschtes Ziel ausgewählt werden und der unmittelbar darauffolgende Wurf wird in den Punkten Wurfkraft und Richtung angepasst. Diese Anpassung soll nicht perfekt sein, sondern lediglich den Wurf des Spielers beeinflussen.

Auch soll die Anvisierung nicht bestehen bleiben, wenn der Raycast vom Ziel entfernt wird. Stattdessen soll die linke Hand mit dem Raycast bei der gesamten Wurf-Interaktion möglichst genau auf das Ziel gerichtet sein.

ANWENDUNGSBEISPIEL

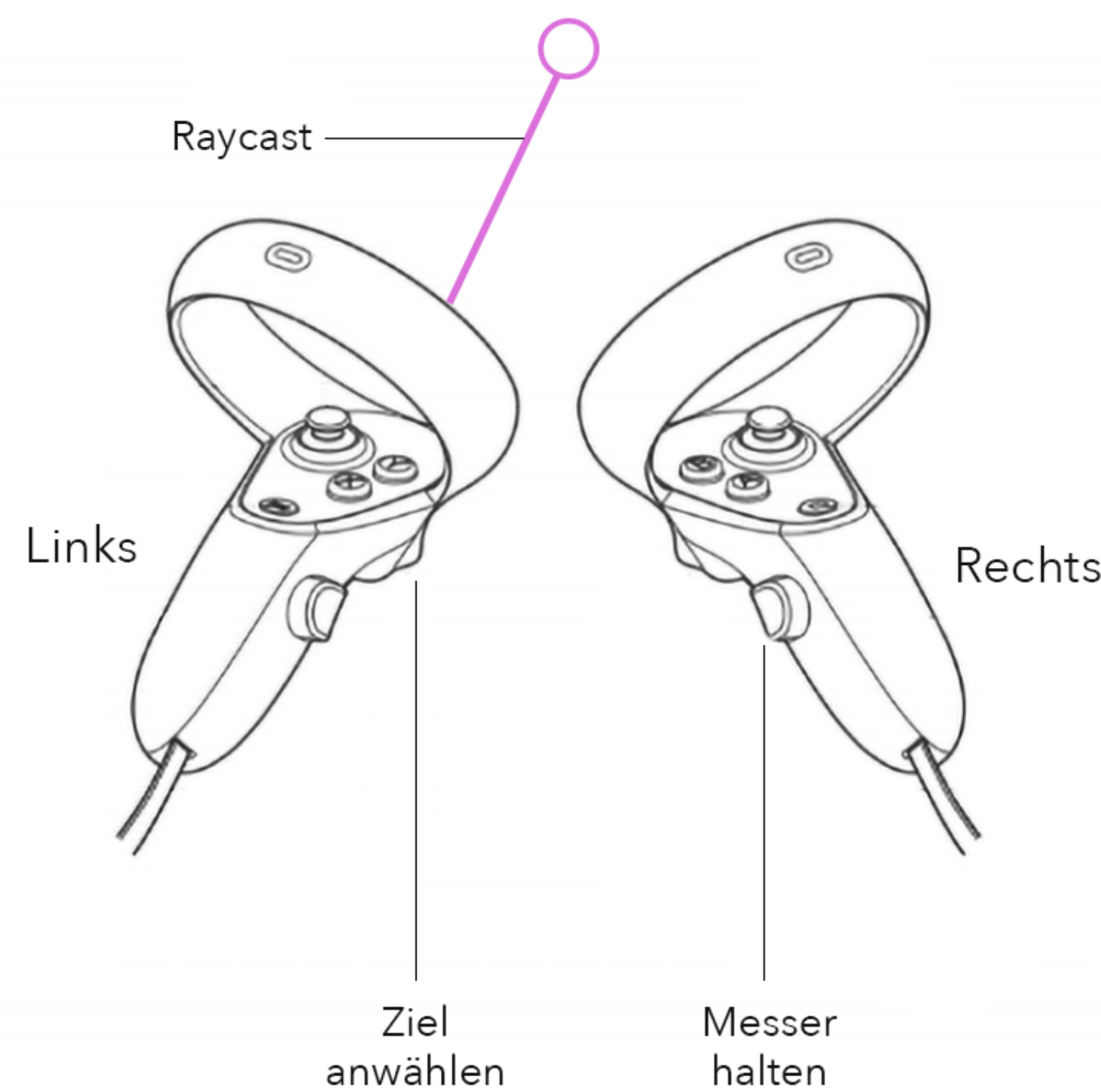


Abbildung 2: Steuerung der Anwendung

Der Spieler befindet sich in einer Szene in der sich sowohl statische Ziele (Dummies) als auch bewegte Ziele (Luftballons) befinden. Ausschließlich diese Objekte können mittels Aim Assist anvisiert werden. Die linke Hand steuert den Raycast, der im Normalzustand grün ist, während die rechte Hand frei ist, um Messer zu halten und zu werfen. Der Raycast kann auf valide Ziele gerichtet und mittels des "Trigger-Buttons" aktiviert werden. Wurde das Ziel ausgewählt, so ändert sich die Farbe des Rays zu blau.

Der Wurfarm kann sich mithilfe des "Grip-Buttons" immer wieder neue Messer aus seinem "Rucksack" holen und sie durch das Loslassen des Buttons inklusive Armbewegung werfen. Wird währenddessen mittels des Raycasts ein Ziel anvisiert und wird das Messer in einen bestimmten Umkreis dieses anvisierten Punktes geworfen, passt sich dessen Flugbahn und Geschwindigkeit automatisch an, um das Ziel möglichst genau zu erreichen.

Bei bewegten Zielen ist es wichtig, schnell zu werfen, damit sich der Ballon nicht zu weit vom anvisierten Punkt entfernt, bevor das Messer ihn erreicht.

ERGEBNISSE

Die Umsetzung mittels des Ray Interactors von UnityEngine.XR erwies sich als durchaus möglich, brachte jedoch auch Probleme mit sich. Physics Raycasts verhalten sich in einem VR Android Build anders als Unity Editor und sollten, wenn nicht mittels des XR Ray Interactor umgesetzt, möglichst gemieden werden.

Das Zielen ist zunächst nicht intuitiv, kann aber schnell gelernt werden und funktioniert gut, wenn richtig angewandt. Zu bemängeln ist dabei die für eine Wurfbewegung unnatürliche stille Haltung der linken Hand. Hier muss starkes Verbesserungspotenzial angemerkt werden, da oft die kleinste Bewegung dazu führt, dass die Aim Assist Funktion das Ziel verliert.

DISKUSSION

Der Lösungsansatz kann durchaus als erfolgreich gewertet werden, birgt jedoch in mehreren Punkten Raum zur Verbesserung.

Zunächst ist anzumerken, dass das Stillhalten der linken Hand zum Zielen der realen Bewegung des Körpers beim Werfen widerspricht. Während die Wurfhand nach vorne bewegt wird, sollte die andere Körperhälfte nach hinten bewegt werden. Dies würde in der VR Umsetzung dazu führen, dass das Ziel des Aim Assists verloren geht. Hier ist es möglicherweise sinnvoller, den ausgewählten Punkt für kurze Zeit abzuspeichern, um dem Spielenden die Möglichkeit zu geben, eine vollständige Wurfbewegung durchzuführen.

Zusätzlich ist das Stillhalten der linken Hand zum Zielen besonders bei Zielen in größerer Entfernung zunehmend schwierig. Kleinste Bewegungen des Controllers führen dazu, dass die Aim Assist Funktion das Ziel verliert. Da dies häufig bei der Wurfbewegung passiert, ist die Funktion besonders für weit entfernte Ziele, bei denen sie eigentlich am meisten benötigt wird, nicht immer hilfreich. Weniger ausschlaggebend, jedoch trotzdem ablenkend, ist die inkorrekte Drehung des Messers in der Hand. Es führt dazu, dass der Spielende möglicherweise am falschen Zeitpunkt der Wurfbewegung loslässt.

Jedoch ist das Zeigen und Selektieren in VR bereits eine weit verbreitete Interaktionsmöglichkeit, die intuitiv genutzt wird und auch auf das Konzept des Anvisierens gut angewandt werden kann [1]. Das Werfen fühlt sich zwar durch die Nutzung des Aim Assists deutlich unrealistischer an, da um einiges weniger Geschwindigkeit in der Wurfbewegung benötigt wird, um das Messer unnatürlich weit zu werfen. Allerdings ist dies vor allem für die bewegten Ziele durchaus wünschenswert, da sich das Treffen der Ballons ohne Aim Assist als sehr schwer entpuppte. Für den Spielspaß eines VR Spiels kann es also somit von Bedeutung sein, dass das Werfen und Treffen nicht unmöglich erscheint, sondern gerade für weniger erfahrene Spieler oder auch Spieler mit motorischen Einschränkungen machbar ist. In diesem Sinne ist die Funktionalität einer Aim Assist Funktion in VR als Accessibility Option umsetzbar, für viele Spieler und Spielerinnen von großem Nutzen und kann in Zukunft mit den angeführten Verbesserungsvorschlägen optimiert werden.

AR: Augmented Reality • **HCI:** Human Computer Interaction • **Mixality:** Mixed Reality Laboratory • **MR:** Mixed Reality • **VR:** Virtual Reality

References

- [1] Ralf Dörner et al. "Grundlagen und Methoden der Virtuellen und Augmentierten Realität". In: *Virtual and Augmented Reality (VR/AR)*. Springer, 2022, pp. 201–244. DOI: <https://doi.org/10.1007/978-3-030-79062-2>. URL: <https://link.springer.com/book/10.1007/978-3-030-79062-2>.
- [2] Adrian Schneider et al. "How Game Balancing Affects Play". In: ResearchGate, 2017, p. 2. DOI: <https://doi.org/10.1145/3064663.3064664>. URL: https://www.researchgate.net/publication/317422610_How_Game_Balancing_Affects_Play_Player_Adaptation_in_an_Exergame_for_Children_with_Cerebral_Palsy.