



Computational Geometry and Virtual Reality - RunAR

Christian Schaf, Jannis Lindenberg, Vural Yilmaz

01. Januar 2019

Inhaltsverzeichnis

1	Einleitung	3
2	Grundlagen	4
2.1	Abgrenzung zu VR im Rahmen des Moduls	4
2.2	A* Algorithmus	4
2.2.1	Funktionsweise	4
2.3	Unity	5
3	Recherche	6
3.1	A* Algorithmus	6
3.2	Imagetargets	6
3.3	Objekterkennung	6
3.3.1	Tensorflow	6
3.3.2	OpenCv	6
3.3.3	RayCast	6
3.4	Spielkonzept	6
4	Konzept	7
4.1	Definition von Funktionsanforderungen	7
4.2	Spielkonzept	7
5	Umsetzung	8
5.1	Implementierung des A* Algorithmus	8
5.2	ImageTargets	8
5.3	Erkennung von Objekten	8
6	Zusammenfassung und Ausblick	9
6.1	Zusammenfassung	9
6.2	Ausblick	9
7	Anhang	10
7.1	A* Beispiel	10

Abbildungsverzeichnis

Kapitel 1

Einleitung

Projektidee, Motivation, Projektplan Einleitung übernehmen

Kapitel 2

Grundlagen

2.1 Abgrenzung zu VR im Rahmen des Moduls

2.2 A* Algorithmus

Der A*-Algorithmus ist ein Wegfindungs-Algorithmus. Sein Ziel ist es den schnellsten Weg vom Start zum Ziel zu finden. Der Ausgangspunkt bildet ein zwei-dimensionales Array in dem es Felder gibt, die entweder Pfad und Hindernisse darstellen. Es wird ein Start- sowie Zielfeld festgelegt. Beide sind dem Algorithmus während der Ausführung bekannt.

2.2.1 Funktionsweise

Sobald das Spielfeld erstellt sowie ein Start- und Zielfeld gewählt wurde, sind alle Voraussetzungen erfüllt.

2.3 Unity

Unity ist eine GameEngine die in diesem Projekt genutzt wird, um das Spiel zu entwickeln. Des Weiteren besteht Unity aus einem sehr umfangreichen Editor, mit dem ein Spieleentwickler eine große Auswahl an Tools zur Entwicklung erhält. Unity bietet sowohl die Möglichkeit in 2D, als auch in 3D zu entwickeln.

Kapitel 3

Recherche

3.1 A* Algorithmus

Das Thema Wegfindung spielt in diesem Projekt eine wichtige Rolle. Der Computergegner soll einen Wegfindungs-Algorithmus benutzen, um den kürzesten Weg zum Ziel finden. Dabei soll die Wegfindung schnell und zuverlässig sein, auch wenn dynamisch ein oder mehrere Hindernisse auftauchen.

Zu den bekanntesten Algorithmen der Wegfindung gehören Dijkstra und A*. -hier warum A*-

3.2 Imagetargets

3.3 Objekterkennung

3.3.1 Tensorflow

3.3.2 OpenCv

3.3.3 RayCast

Donnerstag zusammenmachen

3.4 Spielkonzept

Kapitel 4

Konzept

Nachdem sich die Idee des Projektes verfestigt hat, muss genau ausdefiniert werden, welche Funktionen die Anwendung am Tag der Abgabe erfuellen soll. Nicht alle Visionen lassen sich in dem vorgegebenen Zeitraum umsetzen, sodass eine Liste aus Funktionsanforderungen gefertigt wird. Unumgaenglich war jedoch die Wahl fuer einen Algorithmus zur Merkmalserkennung, der das Herz der Anwendung darstellt. Somit ist bei dessen Wahl besondere Vorsicht geboten, weswegen hierfuer eine Gegenueberstellung von moeglichen Algorithmen ausgearbeitet wurde. Als Bibliothek wurde sich fuer OpenCV entschieden, welche mit der Sprache C++ genutzt wird. Der Grund fuer diese Wahl sind die deutlich weitreichenderen Funde bei Recherchen, sie uebersteigen die von Java oder Python deutlich.

4.1 Definition von Funktionsanforderungen

Bei der Abgabe des Projektes sollen folgende Anforderungen Teil des Funktionsumfang sein.

Donnerstag zusammenmachen

4.2 Spielkonzept

Auf der Skizze aus Abbildung 1 ist ein Musterspielfeld abgebildet. Die Spielfigur (gruener Pull-over), welche sich zu Beginn des Spiels am unteren Rand des Spielfelds befindet, muss den Hindernissen ausweichen. Hindernisse koennen zum Einen aus Image Targets bestehen, auf die in der Spielumgebung 3D Objekte gemappt werden, oder zum Anderen aus realen Objekten, die Im Spielfluss erkannt werden und so ebenfalls ein 3-dimensionales Hindernis fuer den Spieler darstellen. Der Start und das Ziel sind ebenfalls durch Muster gekennzeichnet.

Kapitel 5

Umsetzung

5.1 Implementierung des A* Algorithmus

5.2 ImageTargets

Design der Images und Erkennung der Images

5.3 Erkennung von Objekten

Kapitel 6

Zusammenfassung und Ausblick

6.1 Zusammenfassung

6.2 Ausblick

Kapitel 7

Anhang

7.1 A* Beispiel