

Concept and Development of a VR-Application

Konzept und Entwicklung einer VR-Anwendung

Miglena Dontschewa*, Denise Stamatov‡, Marin B. Marinov†

* University of Applied Sciences Vorarlberg, Department for Computer Sciences and Research Center for User-Centered Technologies, Hochschulstraße 1, AT-6850 Dornbirn, Austria, don@fhv.at

‡ University of Applied Sciences Vorarlberg, Department for Computer Sciences and Research Center for User-Centered Technologies, Hochschulstraße 1, AT-6850 Dornbirn, Austria, denise.daniele@web.de

† Technical University of Sofia, Faculty of Electronic Engineering and Technologies, Department of Electronics, 8, Kliment Ohridski Blvd., BG-1756 Sofia, Bulgaria, mbm@tu-sofia.bg

Abstract — The discussed work concept and develops a VR application under circumstances of a study led by the “Research Center User-Centered Technologies” at Vorarlberg University of Applied Sciences with aiming at inspecting rooms while making use of the HTC-VIVE in 360° mode. Therefore, a solution has to be worked out as well as an alternative will be analyzed. The goal is to present and evaluate techniques to capture real rooms. The captured data then shall be examined with given 360° observation tools by wearing a set of VR goggles.

Zusammenfassung — Die vorgestellte Arbeit konzipiert und entwickelt im Rahmen einer Studie im „Forschungszentrum Nutzerzentrierte Technologien“ an der Fachhochschule Vorarlberg eine VR-Anwendung, um einen Raum mit der HTC-VIVE in 360°-Modus inspizieren zu können. Es wird die Realisierung einer kompletten Lösung und eine Alternative berichtet. Die Arbeit beschäftigt sich mit Technologien zur Aufnahme von realen Räumen. Diese Aufnahmen sollen in 360°-Betrachtertools verwendet werden mit dem Ziel sie als VR-Sets mit VR-Brillen begutachten zu können.

I. EINFÜHRUNG

Anliegen dieser Arbeit ist gegenwärtige Entwicklungen im Bereich der „Virtuellen Realität“ gerecht zu werden. Diese Arbeit verlangt eine interdisziplinäre Herangehensweise, um technische Systeme zu erörtern, die zur Erfüllung der Aufgabenstellung beitragen könnten. Die Herausforderung besteht zum einen darin, dass man die Schnittstellen herausfinden soll und entsprechende Software-Pakete zu finden und/oder notwendige Software-Lösungen zu entwickeln, und zum anderen die Hardware- und Softwarekomponenten dementsprechend zu verbinden, um die Anforderungen der Aufgabenstellung zu erfüllen. Einige dieser Komponenten sind neue Entwicklungen sind und somit durchaus fehleranfällig, instabil und nicht direkt miteinander kombinierbar sind.

Im Rahmen eines geförderten Projekts im Forschungszentrum „Nutzer zentrierte Technologien“ (UCT) an der Fachhochschule Vorarlberg wurde eine Studie durchgeführt, um die Frage zu klären, wie es möglich ist, potentielle Nutzer eines bestimmten Präsentationsgerätes (in diesem Fall Cynap von WolfVision) auf diesem möglichst effektiv hinzuweisen. Die grundlegende Frage hiermit ist: Wie macht man es möglich einen beliebigen Raum dem Menschen in einer virtuellen Realität 360° zugänglich zu machen? Zu diesem Zweck wurden VR-Szenarien konzipiert und entwickelt, entsprechende VR-Geräte und Tools untersucht und Usertests durchgeführt. In diesem Schriftstück wird über die Arbeit berichtet, die eben oben genannte Frage bearbeitet. Es wird auf Related Works nicht eingegangen, denn im Rahmen der Recherche keine gefunden wurden und von partiellen Lösungen abgesehen wird. Die Arbeit ist komplex und behandelt neuartige Kombinationen.

II. METHODIK

Um die Aufgabenstellung zu erfüllen wurde eingehend eine Analyse der möglichen Lösungen durchgeführt und aus verschiedenen Gründen wurden eine VR-Anwendung favorisiert. In Abb. 1 sieht man die bearbeiteten Komponenten, Schnittstellen, Prozesse, Varianten und Workflow.

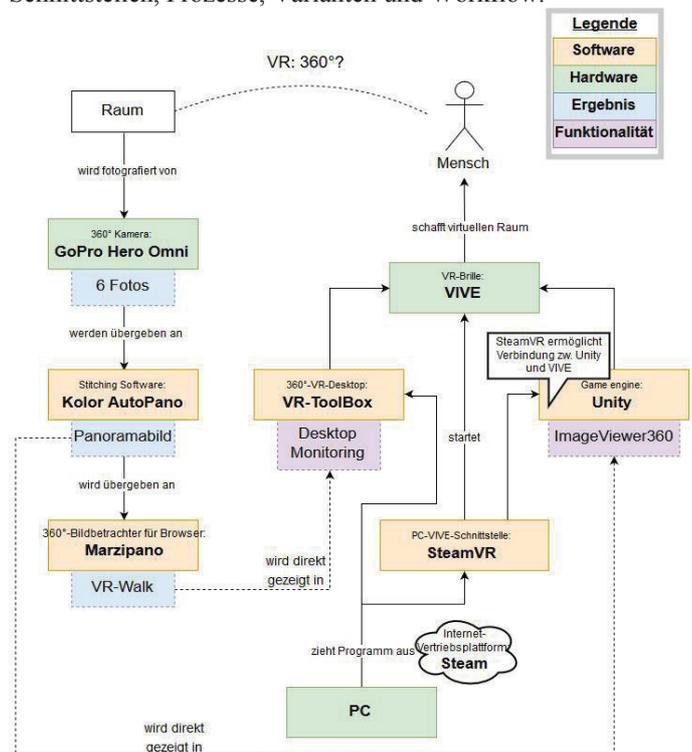


Abb. 1. Einzelne Komponente des Systems und dessen Verbindungen und Workflow

Die Grafik ist so gestaltet worden, dass je nachdem welche VR-Brille benutzt werden soll, sie so angepasst werden kann, dass sie den Ablauf auch für andere VR-Brillen darstellen kann. Dafür müssen die einzelnen Hardware-Pakete, Kamera, VR-Brille und Software-Pakete so verändert werden, dass sie zur bestimmten VR-Brille passen. Um VR-Raum zu generieren sind Panorama - Bilder des darzustellenden Raum notwendig, die Basis der VR-Walk sind. Es sind zwei Möglichkeiten (Virtueller Desktop und Imageviewer360°) in der Grafik zu sehen, die einen bestimmten Raum im virtuellen Raum zu betrachten und mit realen Bewegungen mit dem Raum zu „kommunizieren“ möglich machen.

III. VERSUCHSAUFBAU

A. Hardware

HTC-VIVE

Die HTC-VIVE ist ein HeadMountedDisplay mit einem Sichtfeld von 110° bestehend aus zwei OLED mit je einer Auflösung von 1080x1200 Pixel und eine Aktualisierungsrate von 90Hz. Die HTC-VIVE kann sitzend, stehend und raumfüllend bis zu ca. 25m² (ca. 5m x 5m) benutzt werden [1, 2, 3].



Abb. 2. Bestandteile der HTC-VIVE

Die einzelnen Bestandteile der HTC-VIVE sind in Abb. 2 dargestellt, die aus drei Komponenten besteht: einem VR-Headset bzw. HeadMountedDisplay, zwei Controllern und zwei sogenannten Basisstationen. Aktuell sind einige VR-Brillen auf dem Markt, wie zum Beispiel Meta2000, OculusRift, Hololens usw. Die ausgewählte VR-Brille entspricht den Zielen der vorgestellten Studie relativ gut.

GoProHero Omni

Die „GoPro Hero Omni“ ist eine Kamera, die die Möglichkeit gibt Panoramafotos zu erzeugen. Der Omni-Würfel, wie in Abb. 3 zu sehen, besteht aus einem leichten und mehrteiligen Aluminiumgestell, welches ein synchronisiertes Array aus sechs „GoPro Hero4“ Kameras schützt.



Abb. 3. GoProHero Omni

Diese Synchronisation erleichtert die Nachbearbeitung und das Zusammenfügen der einzelnen Bilder, auch Stitching genannt. Durch die GoPro Smart Remote-Fernbedienung, die dem Fotografen ermöglicht aus der Ferne Fotos zu erzeugen ohne sich im Bild zu verstecken, kann die Haupt- bzw. Masterkamera bis zu einer Entfernung von 180m bedient werden, die dann gleichzeitig den Auslöser für alle Kameras auslöst, da im eingebauten Schaltkreis alle sechs GroPro-Kameras automatisch synchronisiert sind.[4] Es können auch andere Kameras verwendet werden, die Panoramafotos erstellt werden. Alternativen dazu sind unter [5] aufgelistet.

B. Software

VR-Tool-System:Unity, ImageViewer360

Unter Unity ist eine Entwicklungsumgebung des Unternehmens Unity Technologies zu verstehen und hat als Zielplattformen PCs, Spielkonsolen, mobile Geräte und Webbrowser. Mithilfe dieser GameEngine ist es möglich interaktive 3D-Grafik-Anwendungen und Spiele auf unterschiedlichen Betriebssystem, also Windows, Linux und macOS, zu entwickeln.

Der ImageViewer360, der in dieser Arbeit verwendet wird, ist ein Unity-Projekt, welches ermöglicht Panoramabilder in 360° sehen zu können. Angelegt an diesem Blog [6] wird später dieses Unity-Projekt dafür verwendet, durch die HTC-VIVE ein Panoramabild im 360°-Modus zu sehen. Dieser ImageViewer360 funktioniert indem er im Inneren einer Sphäre ein Panoramabild hat und im Zentrum dieser Sphäre dann eine Kamera sitzt, die, wenn die VIVE durch Unity gestartet wird, das VIVE-Headset entspricht.

Zuerst wird eine Sphäre in die Szene gesetzt. Danach muss ein einfaches Gameobject, welches als Kamera-Container dienen wird, im Zentrum der Sphäre positioniert werden. Danach muss das Bild, wie bereits gesagt im Sphäreninneren gelegt werden. Um das zu machen, muss ein sogenannter Shader *Pano360Shader.shader* in der eigenen Programmiersprache von Unity geschrieben werden.[7,8, 9] Dieser Shader hat hier zum Zweck, dass das Panoramabild, welches sich in den Assets vom Projekt befinden sollte, auf die Innenseite der Sphäre gesetzt wird und es nicht zu sehr gestreckt wirkt. In Abb. 4 ist der Codeausschnitt zeigt, der das Panoramabild für die Innenseite der Sphäre spiegelt.

```
36 | | | IN.uv_MainTex.x = 1 - IN.uv_MainTex.x;
```

Abb. 4. Screenshot Codeausschnitt Shader: Panoramabild auf Innenseite der Sphäre spiegelt

Wenn das Projekt dann über die VIVE läuft und das Headset bewegt wird, muss die Kamera diesen Bewegungen folgen.

Dafür wird dem Kameracontainer ein C#-Script *CameraInputRotator.cs* gegeben, welches die Bewegungen in X- und Y-Richtung der Maus der Kamera im Sphäreninneren weitergibt und somit das Panoramabild angeschaut werden kann.

```
13 | | | float horizontal = Input.GetAxis("Mouse X");
14 | | | float vertical = Input.GetAxis("Mouse Y");
```

Abb. 5. Screenshot Codeausschnitt: Maus-Input speichern

Der Codeausschnitt in Abb. 5 zeigt im eben genannten Script, wie der Input der Mauskoordinaten in einer Variablen gespeichert wird.

Um dann die Kamera VR und somit HTC-VIVE tauglich zu machen, muss für die eigentliche Kamera ein weiteres C#-Script *AugmentedRealityCamera.cs* erstellt werden, welches die Information des Gyroskops bzw. Bewegungssensors der VR-Brille so bearbeitet, dass die Kamera in der Sphäre sich bewegt und somit der VR-Brillenbenutzer das Panoramabild 360° im Raum betrachten kann. In Abbildung 9 ist ein Codeausschnitt zu sehen, wie der Input des Gyroskops erhalten und weiterverarbeitet wird.

```
50 transform.localRotation = Input.gyro.attitude * initialRot;
```

Abb. 6. Screenshot Codeausschnitt: Gyroscope-daten weiterverarbeiten

Zum Schluss muss der Sphäre das Material gegeben werden, welches das Panoramabild ist, indem das Bild aus den Assets, in Abb. 7 zu sehen, direkt auf die Sphäre in der Object-Hierarchy im Unity mit der Maus darauf gezogen wird. Da es drei unterschiedliche Sets gibt, werden drei Sphären erstellt, wie Abb. 8 zu sehen, die jeweils die drei unterschiedlichen Panoramabilder als Material haben.

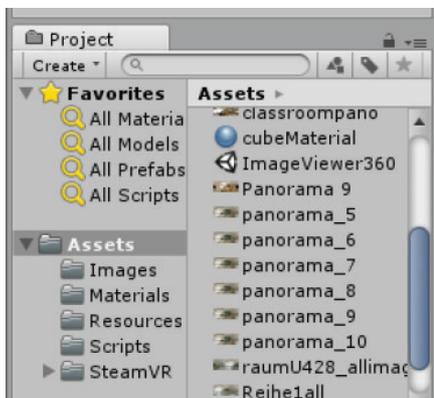


Abb. 7. Screenshot Project-Assets: Panoramabilder



Abb. 8. Screenshot Hierarchie: Unity-Projekt

Steam, SteamVR, VRToolBox

Das Unternehmen Valve entwickelte und betreibt die Internet-Vertriebsplattform Steam, die im Laufe der Jahre für die Betriebssysteme Windows, MacOSX, Android, iOS und Linux erhältlich wurde. Steam vertreibt unter anderem Computerspiele, Software, Filme und Serien. Um die HTC-VIVE überhaupt verwenden zu können, muss momentan ein Account bei Steam angelegt werden, auch wenn die HTC-VIVE die OpenVR API verwendet. Eine OpenSource Implementierung könnte eine Anmeldung umgehen und muss erst geschrieben werden [10].

Wie auch Steam ist auch SteamVR eine Valve-Entwicklung, die eine VirtualReality-Plattform-Erweiterung von Steam ist, d.h. diese Software kann über Steam heruntergeladen werden und ermöglicht eine 360° - VR-Erfahrung.[11] Sie unterstützt nicht nur die HTC-VIVE, sondern auch die Oculus Rift.

SteamVR ermöglicht dem Entwickler die Spielentwicklungsumgebung Unity mit der HTC-VIVE zu verbinden, indem ein Plugin SteamVR aus dem Unity-Assetstore, hier [12], heruntergeladen werden muss. Mithilfe dieses Plugins kann dann zum Beispiel in einem C#-Script die Information erhalten werden, welche dann an einem bestimmten Bestandteil der Unity-Szene gegeben wird, ob das Trackpad eines VIVE-Controllers gedrückt wird. Dieser Input kann dann so viele Würfel im VR-Raum erscheinen lassen, wie das Trackpad gedrückt wurde. Dieses Plugin erweitert somit die Funktionalität von Unity, da der Entwickler einen Zugriff auf die Sensoren, Knöpfen, etc. der HTC-VIVE erhält.

Über Steam im Browser über den Link [13] oder über die Desktopvariante kann das Programm SteamVRToolBox heruntergeladen werden. Dieses Programm schafft für den Benutzer der HTC-VIVE einen virtuellen Desktop. Mit den Controllern kann der Benutzer im VR-Desktop interagieren, das heißt er kann mit den Controllern den Desktop personalisieren und seinen Wünschen nach einrichten.

Bildverarbeitung: Kolor Autopano, Marzipano

Generell können auch andere Stitching-Software verwendet werden, die Alternativen zu den hier vorgestellten Programmen können unter [14] entnommen werden.

Kolor Autopano

Programm Kolor Autopano, das über [15] heruntergeladen werden kann, ist eine Stitching-Software für Fotos und Videos. Stitching bzw. Zusammenfügen bedeutet, wenn aus mehreren einzelnen Fotos eine ganzes Großes erstellt wird.[16] Sie wird zum Erstellen von Panoramafotos (Abb. 9) oder Gesamtansichten von Objekten verwendet. Die Fotos, die zum Beispiel nach dem Stitching ein Panoramafoto darstellen sollen, können direkt per drag&drop ins Programm gezogen werden. Danach wird überprüft, ob die Fotos ein Panorama erzeugen können, indem auf dem Play-Knopf geklickt wird. Das mögliche Resultat wird dann rechts von den einzelnen Fotos dargestellt. Durch Klicken auf das Rädchen öffnet sich ein Fenster, was dazu dient, Einstellungen vorzunehmen und Informationen zu geben, wie Dateiname und -format, Speicherort etc., damit das Rendern gestartet werden kann [17].



Abb. 9. Mögliches Panoramabild

Marzipano

Marzipano ist ein 360°-Bildbetrachter, der im Browser unter [18], verwendet werden kann. Indem das Tool [19] auf dieser Seite verwendet wird, können VR-Walks (Abb.10) aus Panoramafotos erstellt werden. Hierbei wird das Panoramabild per drag&drop ins Browserfenster gezogen und nach einer Lade- und Aufbereitungsphase, können Info- und Link-Hotspots, Startblickpunkt, View-Kontroll-Knöpfe, Fullscreen-Knopf und anderes eingestellt werden.

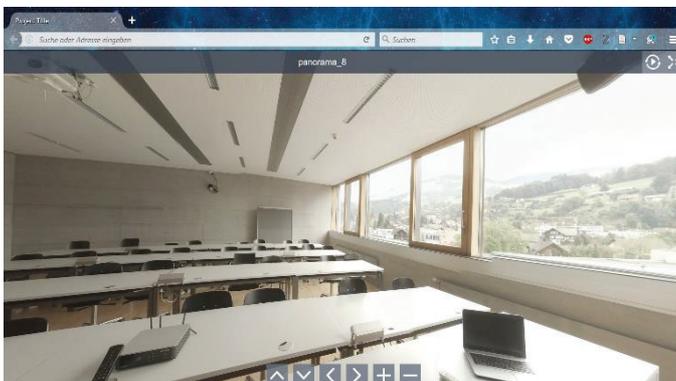


Abb. 10. Screenshot: VR-Walk

Danach kann dieses 360°-Panoramabild als Paket, unter anderem mit einer .html-Datei, exportiert und gespeichert werden.

IV. EXPERIMENTELLE ERGEBNISSE

Die Handhabung der 360°-Kamera birgt eine besondere Spezifik, die verschiedene Einstellungen (Abb. 11) und Tests für die Positionierung notwendig macht. Die Position der Kamera im Prozess der Erstellung der Bilder bestimmt die nachfolgende Perspektive.



Abb. 11. Kamerapositionierung

V. ZUSAMMENFASSUNG

Die VR-Anwendung in der jetzigen Form ist funktionsfähig. Eine Erhöhung der Bildqualität in der VR-Brille, sowie eine performantere Hardwareunterstützung, ist wünschenswert. Es sollten alternative Lösungen bearbeitet

werden, um die Wartbarkeit zu gewährleisten, denn viele Komponenten der derzeitigen Lösung sind von Drittanbietern und zum Teil als Blackbox zu betrachten. Weitere Anwendungsgebiete mit der aufgestellten Lösung sollten recherchiert werden und entsprechend untersucht werden. Die Unity-Lösung kann und soll weiterentwickelt werden, um die Qualität zu erhöhen und weitere Funktionalitäten einzubauen. Eine mögliche Funktionalität könnte die Begehrbarkeit des Raumes sein, um die gezeigten Raumbestandteile von der Nähe betrachten zu können. Auch Interaktivität könnte ausgebaut werden. Die Einsatzfähigkeit der VRToolBox-Lösung soll in ähnlichen Test untersucht werden. Controllerfreie Entwicklung in diesem Zusammenhang sollen stattfinden, zum Beispiel mit LeapMotion oder Kinect.

LITERATURVERZEICHNIS

- [1] Patrick Schulze, "HTC Vive: So kannst Du in die virtuelle Realität eintauchen", Online, 05.02.2016. Available at <https://goo.gl/Gvvrk7>
- [2] Darius Schuiszill, "HTC Vive im Test: Abtauchen und Berge erklimmen", Online, 14.12.2016. Available at <https://goo.gl/EcDdg7>
- [3] HTC, "HTC Vive", Online, 21.10.2017. Available at <https://goo.gl/W9VMgs>
- [4] "Omni", Online, 31.10.2017. Available at <https://goo.gl/LxSmuo>
- [5] "20 best alternatives to GoPro Omni as of 2017", Online, 31.10.2017. Available at <https://goo.gl/AR8RTF>
- [6] Verónica Valls, "How to make a 360°image viewer with Unity3D", Online, 14.03.2016. Available at <https://goo.gl/rqcfMz>
- [7] Unity Technologies, "ShaderLab Syntax", Online, 06.10.2017. Available at <https://goo.gl/WNPFRc>
- [8] "Getting Started with Shaders", Online, 04.11.2013. Available at <https://goo.gl/sPTgBM>
- [9] "How to write unlit surface shader?", Online, 23.06.2012. Available at <https://goo.gl/9h3VzV>
- [10] "Vive Port: HTC bringt eigene App... Steam-Account Pflicht?", Online, 05.08.2016. Available at <https://goo.gl/62EWro>
- [11] "SteamVR", Online, 17.06.2016. Available at <https://goo.gl/fz3TtA>
- [12] "SteamVR Plugin", Online, 29.06.2017. Available at <https://goo.gl/9kFBTK>
- [13] Steam, "VRToolbox: 360 Desktop", Online, 26.12.2016. Available at <https://goo.gl/epsGUH>
- [14] Abhijith N Arjunan, "8 Best Photo Stitching Software", Online, 24.03.2017. Available at <https://goo.gl/tEft7y>
- [15] Kolor, "Autopano", Online, 31.10.2017. Available at <https://goo.gl/M36GaF>
- [16] "Stitching", Online, 26.07.2017. Available at <https://goo.gl/XXooAZ>
- [17] Kolor, "Autopano Documentation", Online, 31.10.2017. Available at <https://goo.gl/n7D3tS>
- [18] Marzipano, "Marzipano", Online, 31.10.2017. Available at <http://www.marzipano.net/>
- [19] Marzipano "Marzipano Tool", Online, 31.10.2017. Available at <http://www.marzipano.net/tool/index.html>
- [20] D. Stamatov, Concept and Development of a VR-Application, Dornbirn, Austria, 2017.