



Hochschule für Angewandte Wissenschaften Hamburg

*Hamburg University of Applied Sciences*

---

*Analyse und Vergleich von Softwaresystemen  
für interaktive Mehrfachprojektion*

---

Bachelor-Thesis  
zur Erlangung des akademischen Grades B.Sc.  
im Studiengang Medientechnik

*Jakob Dohrmann*  
*2079070*

1. Prüfer: Herr Prof. Dr. Ulrich Schmidt  
2. Prüferin: Frau Christina Becker

Hamburg, 21.01.2016

# Kurzfassung

---

Diese Arbeit beschäftigt sich mit der praktischen Realisierung einer interaktiven Mehrfachprojektion. Dazu werden zuerst die Eigenschaften einer solchen Projektion analysiert und dargestellt.

Anschließend werden die Softwaresysteme Ventuz und VVVV vorgestellt, die sich vor allem durch ihre grafische Programmieroberfläche auszeichnen. Anhand der im ersten Teil der Arbeit erarbeiteten Analysepunkte werden beide Systeme miteinander verglichen.

Den Abschluss dieser Arbeit bildet eine persönliche Einschätzung beider Systeme, aus den Erfahrungen aus über einem Jahr des intensiven Umgangs.

## Abstract

---

This thesis deals with the practical realization of an interactive multi-projector setup. The characteristics of it will be analysed in the first part.

Subsequently, the software systems Ventuz and VVVV will be introduced. The distinguishing feature of both systems is their graphical user-interface. In the next section, a test-setup is made on basis of the criteria of analysis developed in part one of this thesis. The realization of this setup has been done in Ventuz as well as in VVVV and the conclusion is presented.

Finally, a subjective evaluation of Ventuz and VVVV, based on personal experience after one year of intensive exposure to both systems, is presented to conclude this thesis.

# INHALTSVERZEICHNIS

---

<b>I. – Einführung</b> .....	<b>4</b>
<b>II. – Interaktive Mehrfachprojektion</b> .....	<b>6</b>
<b>II.I. – Interaktion im Zusammenspiel mit Projektion</b> .....	<b>6</b>
<b>II.II. – Mehrfachprojektion</b> .....	<b>7</b>
II.II.I. – Klassische Mehrfachprojektionen .....	8
II.II.II. – Projektions-Mapping.....	8
<b>II.III. – Hardwareanforderungen einer interaktiven Mehrfachprojektion</b> .....	<b>10</b>
II.III.I. – Ausspielsysteme und Medienserver .....	10
II.III.II. – Microsoft Kinect.....	11
II.III.III. – radarTOUCH Rotationslaser .....	14
II.III.IV. – MIDI Controller .....	15
<b>II.IV. – Softwareanforderungen einer interaktiven Mehrfachprojektion</b> .....	<b>16</b>
II.IV.I. – Erstellen und Abspielen von Content .....	17
II.IV.II. – Projektions-Mapping .....	17
II.IV.II.I. – Trapezkorrektur .....	17
II.IV.II.II. – Komplexes Projektions-Mapping .....	18
II.IV.II.III. – Anforderungen eines Projektions-Mappings .....	19
II.IV.III. – 3D-Objekte .....	19
II.IV.III.I. – Live-Animation von 3D-Objekten .....	20
II.IV.IV. – Audio-Reactive Content.....	20
II.IV.IV. – Netzwerkanbindungen .....	21
<b>III. – Einführung in Softwaresysteme</b> .....	<b>22</b>
<b>III.I. – Ausspielsysteme und Medienserver</b> .....	<b>22</b>
<b>III.II. – Game-Engines</b> .....	<b>26</b>
<b>III.II. – Hybride Systeme Ventuz und VVVV</b> .....	<b>27</b>
<b>IV. – Ventuz und VVVV</b> .....	<b>28</b>
<b>IV.I. – Erstellung von Anwendungen in Ventuz und VVVV</b> .....	<b>28</b>
<b>IV.II. – Die Installation</b> .....	<b>30</b>
IV.II.I. – Ventuz Installation.....	30
IV.II.I. – VVVV Installation.....	30
<b>IV.III. – Das User-Interface</b> .....	<b>31</b>
IV.III.I. – Ventuz User-Interface .....	31
IV.III.I. – VVVV User-Interface .....	32
<b>IV.III. – Performance und Stabilität</b> .....	<b>33</b>
IV.III.I. – Performance und Stabilität in Ventuz .....	33
IV.III.I. – Performance und Stabilität in VVVV .....	33
<b>IV.III. – Support</b> .....	<b>33</b>

<b>V. – Praktischer Versuchsaufbau .....</b>	<b>35</b>
V.I. – Die Projektionsfläche .....	35
V.II. – Abspielen von Audio und Video Dateien .....	36
V.III. – Content erstellen und bearbeiten .....	36
V.IV. – Interaktivität – Audio-Reactive Content .....	36
V.IV. – Interaktivität – Microsoft Kinect .....	36
V.IV. – Interaktivität – radarTOUCH.....	37
V.IV. – Steuerung – MIDI Controller .....	37
<b>VI. – Versuchsergebnisse.....</b>	<b>38</b>
V.I. – Die Projektionsfläche - Rendering .....	38
V.II. – Die Projektionsfläche - Mapping .....	38
V.III. – Abspielen von Audio und Video Dateien.....	40
V.IV. – Content erstellen und bearbeiten .....	40
V.V. – Interaktivität – Audio-Reactive Content.....	41
V.VI. – Interaktivität – Microsoft Kinect .....	41
V.VII. – Interaktivität – 3D-Objekte.....	42
V.VIII. – Interaktivität – TUIO Protokoll.....	43
V.IX. – Steuerung – MIDI Controller .....	43
<b>VII. – Persönliche Stellungnahme .....</b>	<b>45</b>
VII.I. – Der erste Eindruck .....	45
VII.II. – Systemoffenheit und ein unerklärlicher Fehler .....	46
VII.III. – Ventuz oder VVVV .....	47
VII.IV. – Schlussfolgerung .....	47
<b>X.I. – Abbildungsverzeichnis .....</b>	<b>49</b>
<b>X.II. – Quellenverzeichnis .....</b>	<b>51</b>
<b>X.II. – Eigenständigkeitserklärung .....</b>	<b>53</b>

# Kapitel I

## EINFÜHRUNG

In der folgenden Arbeit setze ich mich mit dem Thema der interaktiven Mehrfachprojektion auseinander. Dazu wird zunächst erläutert, welche Eigenschaften eine interaktive Mehrfachprojektion mit sich bringt. Anschließend werden die beiden Komponenten Interaktion und Mehrfachprojektion genauer analysiert.

Interaktive Mehrfachprojektionen sind komplexe, technisch sehr aufwändige Installationen, die bisher kaum verbreitet sind. Das liegt vermutlich auch daran, dass diese Installationen meist sehr kostenintensiv und nur mit einem guten Konzept sinnvoll einzusetzen sind. Bisher finden solche Installationen besonders in der Theater- und Kunstszene Beachtung und werden als Showelemente verwendet. Zunehmend finden sie auch in der Werbebranche immer mehr Anklang, wie beispielsweise Porsche im Rahmen der Vorstellung des neuen Porsche 911 im Jahr 2011 zeigt.

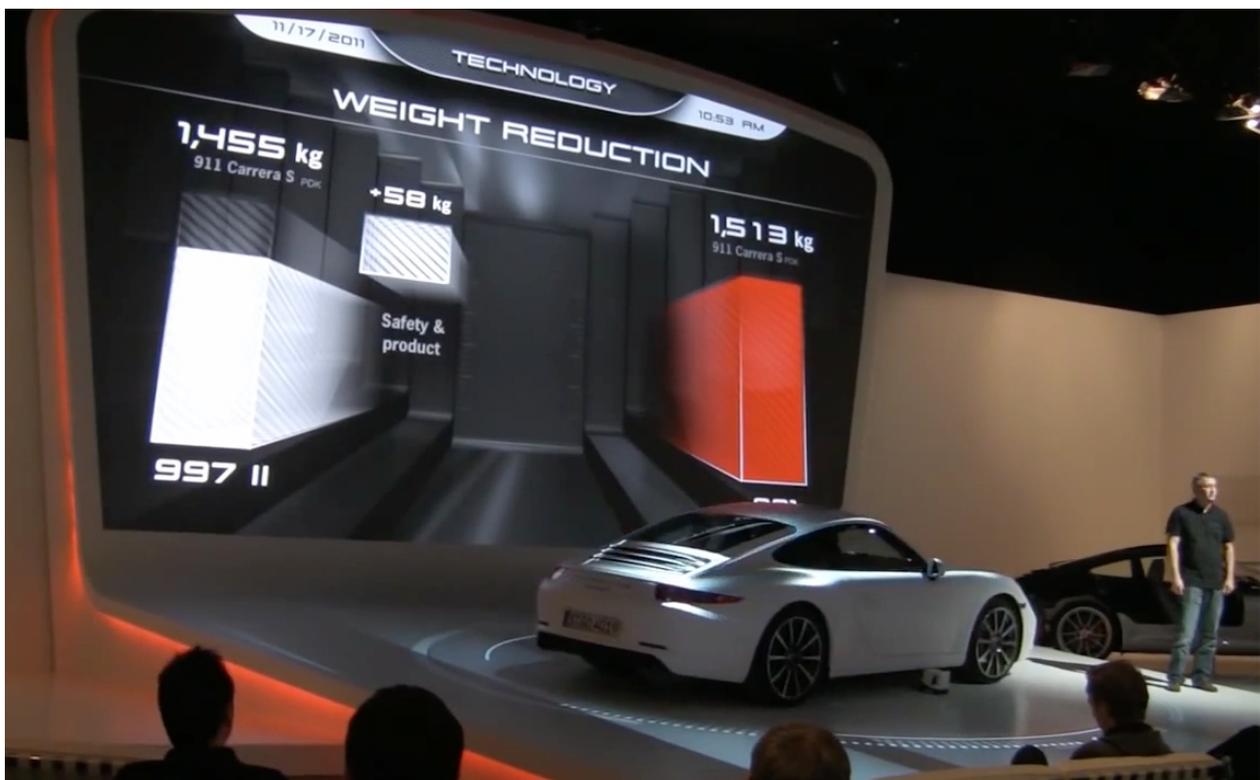


Abbildung 1.1. - Interaktive Präsentation des neuen Porsche 911

Ich gehe davon aus, dass in Zukunft interaktive Mehrfachprojektionen zunehmen und in immer neuen Anwendungsfeldern auftreten werden, da schon heute Interaktion wie z.B. bei einer Gestensteuerung immer mehr Technik-Entwickler beschäftigt. Im Zusammenhang mit anderen Trends, wie u.a. der Hologramm-Technologie, wird Interaktion ebenfalls eine große Rolle spielen, beispielsweise um Hologramme zu bewegen und diese damit so real wie möglich zu gestalten. Abbildung 2.2. (Seite 7) zeigt ebenfalls eine mögliche praktische Anwendung einer interaktiven Projektion. Dort ist eine projizierte Tastatur zu sehen, die mit einem Trackingsystem verknüpft ist und dadurch eine reale Tastatur ersetzen kann.

Im zweiten Teil der Arbeit wird anhand eines praktischen Versuchs untersucht wie geeignet die Softwarelösungen VVVV und Ventuz zur Realisierung einer interaktiven Mehrfachprojektion sind. Dazu habe ich eine eigene Installation konzipiert, die verschiedene Elemente einer interaktiven Mehrfachprojektion beinhaltet. Anhand der Elemente dieser Installation wird Ventuz mit VVVV verglichen. Dabei habe ich mich für diese Systeme entschieden, da sie beide durch objektbasiertes Programmieren dem Nutzer unzählige Anwendungen ermöglichen. Sie sind die einzigen mir bekannten Systeme, die in diesem Umfang objektbasiertes Programmieren unterstützen. In der Anwendung verhalten sie sich daher sehr ähnlich. Dennoch kostet eine Ventuz-Lizenz 6950,00€ wohingegen eine VVVV-Lizenz bei 500,00€ liegt. Bezogen auf interaktive Mehrfachprojektion erläutere ich in dieser Arbeit, welche Vorteile Ventuz bietet und ob VVVV vielleicht in einigen Bereichen die bessere Wahl ist.

Zum Abschluss beschreibe ich in einer persönlichen Stellungnahme meine Erfahrungen, die ich in über einem Jahr intensiven Umgangs mit Ventuz und VVVV gesammelt habe.

# Kapitel II

---

## INTERAKTIVE MEHRFACHPROJEKTION

Einführend werde ich in diesem Kapitel erläutern, was unter interaktiver Mehrfachprojektion zu verstehen ist. Dabei gehe ich sowohl darauf ein, wie Interaktion in Zusammenhang mit Projektion funktioniert, als auch auf die typischen Eigenschaften einer Mehrfachprojektion. Die daraus resultierenden Hardware- sowie Softwareanforderungen für eine interaktive Mehrfachprojektion werden im folgenden Schritt beschrieben und einige etablierte Systeme vorgestellt. Anschließend werde ich näher auf die Softwaresysteme Ventuz und VVVV eingehen.

### II.1. – Interaktion im Zusammenspiel mit Projektion

Interaktive Projektion ist ein stark anwachsender Trend, der heutzutage vor allem bei Installationen aber auch im Theater bei großen Bühnenshows<sup>1</sup> oder Präsentationen zum Einsatz kommt. Interaktive Projektionen zeichnen sich dadurch aus, dass die Projektion nicht wie ein Film linear abgespielt wird. Stattdessen nimmt sie Bezug auf eine oder mehrere Personen, wodurch sich die Projektion verändern bzw. steuern lässt. Dabei ist die wohl bekannteste Form der Interaktion das sogenannte Tracking. Beim Tracking werden je nach System eine oder mehrere Personen oder einzelne Körperteile wie z.B. Hände vom jeweiligen Trackingsystem erkannt und verfolgt. Je nach der Position verändert sich ein entsprechender Inhalt<sup>2</sup> der Projektion. Mit Hilfe des Trackings ist es möglich z.B. Personen von einem Hintergrund hervorzuheben oder wie auf Abbildung 2.1 die Hände einer Person zu verfolgen und diese mit Effekten zu versehen. In der Abbildung bewegt die Person ihre



Abbildung 2.1. - Tracking der Hände einer Person.

---

<sup>1</sup> Show: Als Show wird der (technische) Ablauf einer Veranstaltung bezeichnet.

<sup>2</sup> Inhalt: Im Videotechnischen Kontext wird der Inhalt häufig mit dem englischen Wort „Content“ bezeichnet.

Arme nach oben und an der Position der Hand entsteht ein Kreis, welcher sich mit der Zeit vergrößert.



Abbildung 2.2. - Ein Mini-Projektor und ein Trackingsystem ersetzen ein Touchdisplay.

Wie Abbildung 2.2 zeigt, ist es denkbar mit einem Projektor und einem Fingertracking ein Touchdisplay zu ersetzen.

Neben dem Tracking ist auch die Sprachsteuerung eine Methode zur Interaktion. Diese kommt jedoch deutlich seltener zum Einsatz, da sie visuell keinen Einfluss nehmen kann. Tatsächlich lässt sich dadurch die Projektion lediglich steuern. Andere Möglichkeiten, eine Projektion zu steuern, wären MIDI Controller oder andere haptische Geräte. Dabei wird die Projektion durch Drücken von Knöpfen oder Drehen von Reglern beeinflusst. Dies stellt zwar im weitesten Sinn eine Interaktion dar; im Gegensatz zum Tracking fühlt man sich bei einer solchen Interaktion jedoch nicht derartig in die Projektion integriert, wie es beim Tracking der Fall ist.

## II.II. – Mehrfachprojektion

Eine Mehrfachprojektion ist eine Projektion, die sich aus den Projektionen mehrerer einzelner Projektoren zusammensetzt. Dabei haben die Projektoren meist unterschiedliche Standorte. Bei Mehrfachprojektionen ist es daher wichtig, dass die einzelnen Projektoren aufeinander

abgestimmt sind; dies gilt insbesondere für die Ausrichtung der Projektoren, sowie für die Farbeinstellung oder Auflösung.

### II.II.I. - Klassische Mehrfachprojektionen

Simple Formen der Mehrfachprojektion sind Softedge- oder Tandemprojektionen. Eine Tandemprojektion ist eine Projektion mit zwei Projektoren die über- oder nebeneinander stehen und auf exakt dieselbe Fläche das gleiche Bild projizieren. Dies wird häufig eingesetzt, um die Helligkeit des projizierten Bildes zu erhöhen. Eine Softedgeprojektion ist eine Projektion mit mehreren Projektoren. Sie ermöglicht ein breiteres Bildformat mit einer hohen Auflösung und Helligkeit zu realisieren.

In Abbildung 2.3 bilden 2 Projektoren eine Softedge mit der Auflösung von 2880x1200 Pixeln, also einem

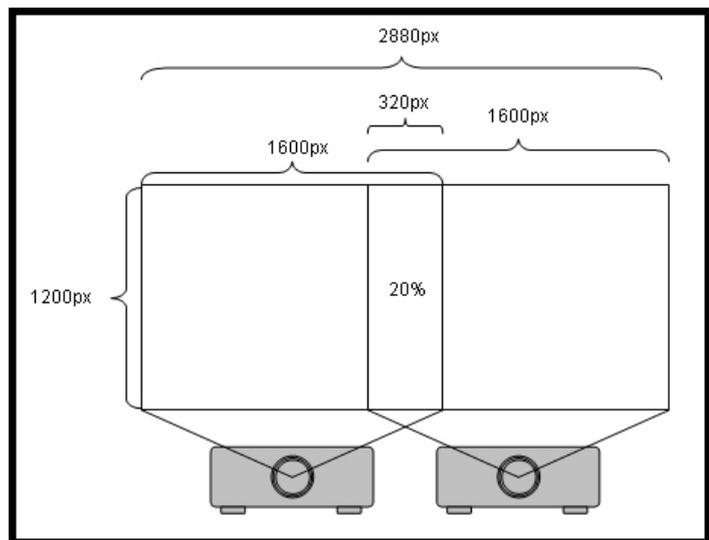


Abbildung 2.3. - Softedgeprojektion mit 2 Projektoren.

Bildformat von 12/5. Würde man dieses Bildformat mit nur einem dieser Projektoren abrufen, würde man eine Auflösung von 1600 x 667 erhalten. Zudem müsste der Projektor eine größere Fläche bespielen, wodurch das Bild dunkler wird. Wie auf der Abbildung zu sehen, überschneiden sich die Bilder der Projektoren bei der Softedge. Um einen nicht sichtbaren Übergang zwischen den Projektoren zu schaffen, werden dabei die Bilder an ihren Kanten überlagert. Dabei muss der überlappende Inhalt der Projektoren pixelgenau identisch sein, was ein hohes Maß an Präzision bei der Einrichtung erfordert.

Mehrfachprojektionen können jedoch auch andere, deutlich komplexere Formen annehmen. Es ist denkbar, dass eine Figur z.B. ein Würfel von allen Seiten bespielt wird, sodass Personen um die Projektion herum gehen können. Es ist ebenfalls möglich alle vier Seiten eines Raumes zu bespielen. Eine Person ist dann komplett von einer Projektion umgeben. Solche Projektionen werden als 360°-Projektion bezeichnet.

### II.II.II. – Projektions-Mapping

Eine besondere Form der Projektion ist das Projektions-Mapping. Mapping bedeutet in diesem Fall, dass der Content speziell auf die Fläche, auf die er projiziert werden soll,

abgestimmt wird. In der Regel wird dabei nicht mehr auf zweidimensionale Leinwände projiziert, sondern auf komplexere Flächen oder Formen, wie z.B. Häuserfassaden. Durch eine präzise Abstimmung von Projektionsfläche und Content vermischen sich die realen Strukturen mit den projizierten Inhalten und können in besonderen Fällen vom Zuschauer nicht weiter getrennt werden.



**Abbildung 2.4.**  
Projektions-Mapping einer real erscheinenden Fassade.



**Abbildung 2.5.**  
Projektions-Mapping einer einstürzenden Fassade.

Wie Abbildung 2.4 zeigt, legt sich das projizierte Bild eine Häuserfassade über die reale Fassade des Gebäudes. Auf den ersten Blick ist dabei nicht zu erkennen, dass die Fassade eine Projektion ist. Abbildung 2.5 zeigt die gleiche Projektion einen kurzen Moment später.

Wie dort zu erkennen ist, wirkt es so, als würde die Fassade einstürzen und dahinter einen neuen Raum freigeben. Anhand dieses Beispiels sieht man sehr gut, welche spannende Wirkung ein Projektions-Mapping erzeugen kann.

Projektions-Mapping muss dabei nicht immer eine Mehrfachprojektion sein, dieses Beispiel kann auch mit nur einem Projektor realisiert werden. Da aber bei einem Projektions-Mapping häufig sehr große Flächen bespielt werden, ist meist ein Einsatz von mehreren Projektoren sinnvoll. Bei komplexeren Projektionsflächen - wie dem Opernhaus in Sydney (Abbildung 2.6) - müssen zwangsläufig mehrere Projektoren eingesetzt werden um alle Bereiche der Projektionsfläche zu bespielen. In diesem extremen Beispiel wurden 17 Projektoren eingesetzt.



Abbildung 2.6. - „Lighting the Sails“ – Fassaden-Mapping auf dem Sydney Opera House.

### **II.III. – Hardwareanforderungen einer interaktiven Mehrfachprojektion**

Eine interaktive Mehrfachprojektion stellt also grundsätzlich gleich zwei wesentliche Anforderungen an die Hardware. Zum einen muss Hardware gegeben sein, um eine Mehrfachprojektion zu realisieren. Neben den Projektoren und der dazugehörigen Technik wie Objektive und Kabel muss ein entsprechendes Ausspielsystem vorhanden sein. Das Ausspielsystem generiert die entsprechenden Videosignale für die einzelnen Projektoren und sollte auf das Projekt abgestimmt sein. Zum anderen wird Hardware benötigt, welche eine Interaktion mit Personen realisieren kann. Dazu muss diese Hardware interaktive Daten erfassen und diese mittels Software dem Nutzer zur Verfügung stellen. In der Regel ist solche Hardware sehr spezifisch, da interaktive Projektionen bisher keinen wirtschaftlichen Markt darstellen, um Produkte extra für diesen Zweck zu entwickeln. Im Folgenden möchte ich einige Hardwarelösungen vorstellen und erläutern.

#### **II.III.I. – Ausspielsysteme und Medienserver**

Die meisten Ausspielsysteme werden als Medienserver bezeichnet und werden speziell für den Einsatz als Grafiksystem konzipiert. Im Prinzip kann aber auch jeder PC oder Laptop ein Ausspielsystem sein. Medienserver zeichnen sich durch die gezielte Konfiguration und eine sehr hohe Performance, besonders im Grafikbereich, aus. Besonders wichtig ist auch die Zuverlässigkeit dieser Systeme, da ein Absturz während einer Show in der Regel zum

Abbruch oder zumindest zu extremen Verzögerungen führt. Heutzutage sind fast alle gängigen Medienserver Software-gesteuert; dadurch lassen sich die Systeme einfach bedienen und auch komplexe Setups in kurzer Zeit konfigurieren. Außerdem lassen sich moderne Systeme miteinander synchronisieren, sodass eine Show durch mehrere Serversysteme realisiert werden kann. Dies kommt bei sehr großen Shows vor, die viele Outputs und eine hohe Renderleistung benötigen. Medienserver können von externen Systemen gesteuert werden oder die Software wird direkt auf einem System installiert.

Moderne Medienserver sollten mindestens zwei Ausspielwege anbieten. In Systemen, die nicht extern gesteuert werden, sollten also Grafikeinheiten mit mehr als zwei Ausgängen integriert sein, da zum Steuern der Anwendung ebenfalls Monitore benötigt werden. Wichtig ist dabei, dass die Rechenleistung im Verhältnis zu den Ausgängen steht und ausreicht, um alle Ausgänge mit einer angemessenen Auflösung zu bespielen.

### **II.III.II. - Microsoft Kinect**

Microsoft hat mit der Kinect eine Kamera auf den Markt gebracht, die speziell für Interaktion entwickelt wurde. Ausgelegt ist die Microsoft Kinect ursprünglich für die Spielkonsole XBOX. Da die Kinect für Interaktion konzipiert wurde, ist sie auch für andere Anwendungen nutzbar und es wurden verschiedene Treibersysteme entwickelt. Mit diesen Treibern ist es möglich die Kinect auch mit dem PC zu verbinden. Die Microsoft Kinect besteht aus verschiedenen Komponenten, neben einer RGB-Kamera, einem Mikrofon-Array, sowie einem Motor zur Neigung der Kamera ist auch ein System zur Messung eines Tiefenbildes verbaut. Die Tiefenmessung funktioniert mit Hilfe eines Infrarot-Projektors, welcher ein für Menschen nicht sichtbares Infrarot-Muster in den Raum projiziert. Dieses Muster wird von einer Infrarot-Kamera erkannt. Mittels Berechnungen wird ein Tiefenbild des Raumes erstellt.

Für die Verwendung einer Microsoft Kinect unter Windows stehen grundsätzlich zwei verschiedene Treibersysteme zur Verfügung. Einerseits gibt es das Microsoft Kinect Software Development Kit<sup>3</sup>. Dieses SDK wird von Microsoft zur Verfügung gestellt, um es Programmierern zu ermöglichen eigene Anwendungen für die Kinect zu entwickeln. Dabei stehen neben den Treibern auch verschiedene Code- oder Anwendungsbeispiele der Kinect zur Verfügung.

---

<sup>3</sup> Software Development Kit wird als SDK abgekürzt.

Andererseits gibt es Treiber von OpenNI, das steht für „Open Natural Interaction“. Dahinter verbirgt sich eine Organisation, welche Frameworks<sup>4</sup> und Programmierschnittstellen definiert. Nachdem PrimeSense, eines der wichtigsten Unternehmen der OpenNI-Gruppe, von Apple aufgekauft worden ist, ist auch die Website dieser Gruppe nicht mehr online verfügbar. Die Treiber von OpenNI werden aktuell nicht weiterentwickelt; für die neue Kinect2 von Microsoft bietet OpenNI keine Treiber an. Die OpenNI Treiber wurden vor dem SDK von Microsoft veröffentlicht, da nun aber Microsoft eigene Treiber bereitstellt, gibt es kaum noch Bedarf an den OpenNI Treibern, sodass davon auszugehen ist, dass diese in Zukunft keine Rolle spielen werden.

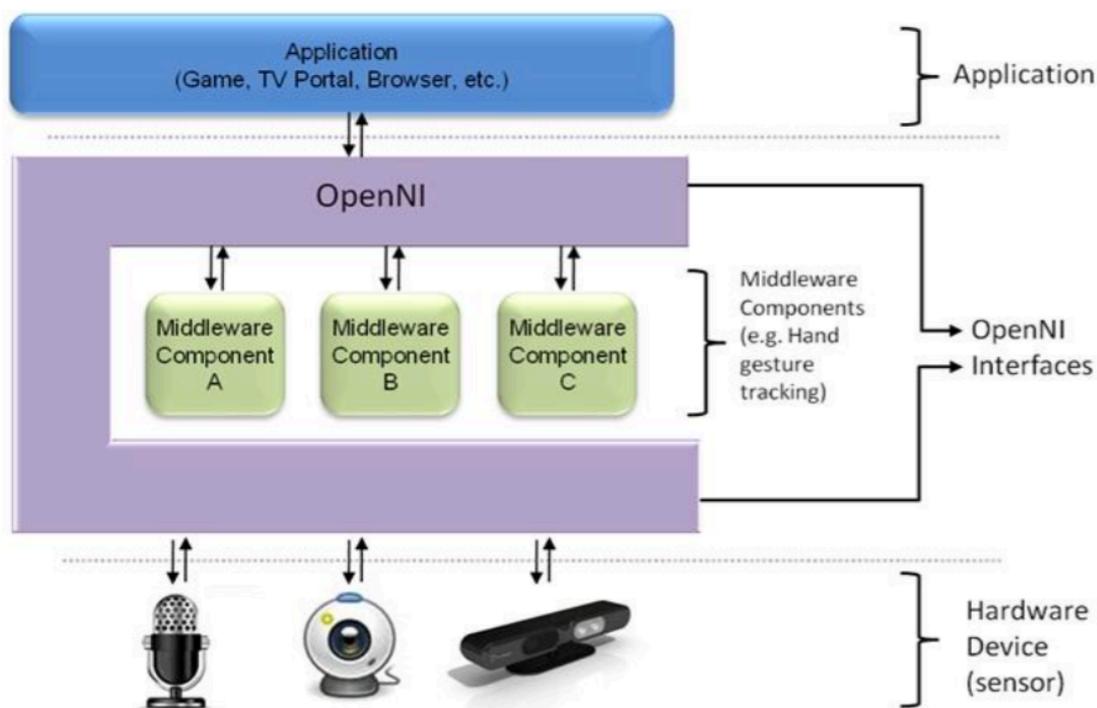


Abbildung 2.7. - Treiberkommunikation Microsoft Kinect

Wie in Abbildung 2.7 zu sehen ist, bilden die Treiber nicht nur die Verbindung zwischen der Hardware und der Anwendung, sie implementieren auch noch die sogenannte Middleware. Als Middleware werden hardwarespezifische Anwendungen bezeichnet, die vom Hersteller für die Hardware vorgesehen und programmiert wurden. Die Middleware der Microsoft Kinect stellt dem verschiedene Funktionen zur Verfügung. Neben dem Live-Bild, der RGB-Kamera und Mikrofonsignalen sind auch spezifische Anwendungen der Tiefenbildmessung möglich, z.B. ein Tracking einer Person.

<sup>4</sup> dt: Rahmenstrukturen – Frameworks dienen in der Programmierung zur Definition der Kommunikation zwischen Software und anderer Soft- oder Hardware.

Dabei ist die Funktionsweise des Trackings sowie die Anzahl und Positionen der Trackpunkte bereits durch die Middleware definiert. Der Nutzer bekommt vorgefertigte Programmiererelemente, welche er verwenden kann. Das erspart einen extremen Programmieraufwand, hindert im Gegensatz dazu aber auch, eigene Elemente hinzuzufügen oder vorhandene Elemente zu erweitern.

Im Fall der Microsoft Kinect werden dem Nutzer für ein Personentracking genau 20 Trackpunkte zur Verfügung gestellt. Diese arbeiten dank der Tiefenbildkamera der Kinect in einem 3-dimensionalen Raum. Abbildung 2.8 zeigt wie sich die Trackpunkte auf den Körper einer Person aufteilen.

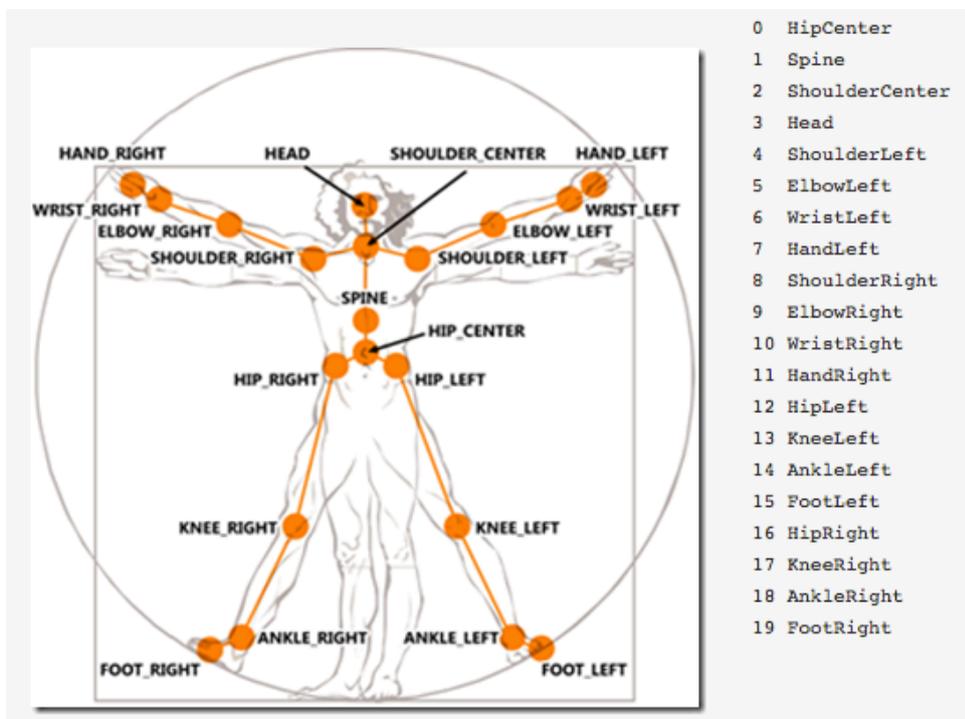


Abbildung 2.8: Trackpunkte der Microsoft Kinect

Diese Trackpunkte stehen dem Nutzer nun zur Verfügung, um eigene Anwendungen zu gestalten. Dabei müssen nicht alle Trackpunkte verwendet werden. Häufig beziehen sich interaktive Anwendungen auf den Kopf oder die Hände von Personen. Die Kinect bietet außer diesem sogenannten Skeleton-Tracking weitere Anwendungen, darunter eine Gestenerkennung und eine Sprachsteuerung.

Neben der Microsoft Kinect gibt es die Microsoft Kinect2 oder die Asus Xtion. Alle Kameras arbeiten nach einem ähnlichen Prinzip und liefern ähnliche Daten.

### II.III.III. – radarTOUCH Rotationslaser

Eine weitere interaktive Hardwarelösung ist das radarTOUCH System der Lang AG. Dahinter verbirgt sich eine Technologie, die ursprünglich aus der Industriesicherheit stammt und heute zur Umsetzung von großen Multitouchflächen verwendet wird. Mit Hilfe eines Rotationslasers wie z.B. dem ROD4plus der Firma Leuze electronic wird dabei eine 2-dimensionale Fläche gescannt und die Daten an eine eigene Software übertragen. Der Laser sendet Lichtwellenstrahlen in einer 2-dimensionalen Fläche aus, die in etwa eine Teilkreisfläche von  $190^\circ$  bildet. Treffen diese Strahlen auf eine feste Oberfläche, werden sie reflektiert. Mittels einer Laufzeitmessung bestimmt die Software die Entfernung vom Laser zum Objekt. Mit Hilfe der eigenen Software lässt sich eine Fläche definieren, in der die Touchpunkte erfasst werden. Relativ zu dieser Fläche werden entsprechende Koordinaten für die Touch-Eingaben vergeben. Die Software sendet die Daten als TUIO-Protokoll<sup>5</sup> in ein Netzwerk und kann von Systemen verwendet werden, die das TUIO-Protokoll unterstützen. Neben der Firma Lang AG vertreibt auch die Gruppe project:syntropy ein sehr ähnliches System als „synTouch radar“.

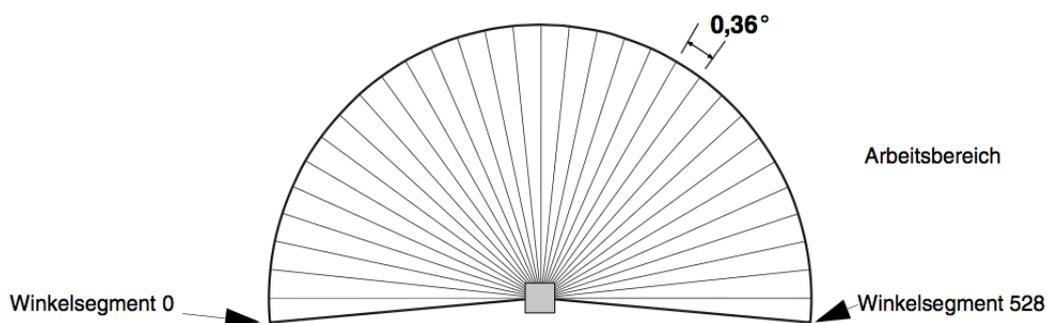


Abbildung 2.9. – grafische Darstellung des Arbeitsbereichs des Leuze Rod4 Plus Laser

Abbildung 2.9 zeigt wie sich die Lichtwellenstrahlen punktförmig vom Laser ausbreiten. Bedingt durch die Funktionsweise des Systems entstehen Schatten im Messbereich des Lasers, der laut Hersteller bis zu 200 Touchpunkte unterstützt. Objekte, die sich vom Laser aus gesehen hinter einem Hindernis befinden, können daher nicht erfasst werden. Dabei verhalten sich auch „Touch-Eingaben“ als Hindernisse.

<sup>5</sup> TUIO-Protokoll: Das TUIO-Protokoll, ist ein Protokoll zum Versenden von Daten in einem Computer-Netzwerk.

## II.III.IV. – MIDI Controller

Eine haptische Form der Interaktion mit Software bieten MIDI<sup>6</sup> Controller. MIDI ist ein etablierter Industriestandard zum Austausch von Steuersignalen. Ursprünglich kommt Midi aus der Musikbranche und wurde bereits 1982 eingeführt. MIDI Controller werden heutzutage häufig zur Steuerung von PC-basierter Software eingesetzt, vor allem in DJ- und Lichttechnikbereichen.

Im Vergleich zu der Kinect oder dem Rotationslaser, ist MIDI als Interaktion weniger beeindruckend, da der Nutzer beim Drücken einer Taste eine Reaktion erwartet. MIDI wird jedoch häufig benutzt, um Software nicht über Maus und Tastatur zu steuern, sondern wie ein Steuerpult aus der Veranstaltungstechnik.

MIDI Controller unterstützen verschiedene Steuersignale, die ich im Folgenden kurz vorstellen werde. Die Hardware des Controllers und das Kommunikationssignal werden beide über das MIDI Protokoll übermittelt, sodass MIDI Controller in der Regel Plug & Play-artig<sup>7</sup> funktionieren. Eine zusätzliche Software ist in der Regel nicht notwendig, kann aber hilfreich sein. Die Firma Novation bietet zu ihren Controllern die Software Automap an. Diese ermöglicht zusätzliche Features, wie das individuelle Anpassen von Kanälen und Funktionen des Controllers.



Wie auf dem Bild links zu sehen, hat der Novation Zero SL MkII, ein MIDI Controller der Firma Novation, verschiedene Steuerfunktionen. Im untersten Bereich des Bildes sehen wir die Buttons des Controllers. Diese können lediglich aktiviert oder deaktiviert sein. In Programmierumgebungen werden solche Buttons häufig als

Abbildung 2.10: Automap-Oberfläche des Novation Zero SL MkII

<sup>6</sup> MIDI: Musical Instrument Digital Interface

<sup>7</sup> Plug & Play: Ein System wird mit Plug & Play beschrieben, wenn es ohne Installation, sondern direkt beim Einstecken eines Steckers, funktioniert.

Toggles<sup>8</sup> bezeichnet. Wird ein Toggle einmal gedrückt, nimmt dieser den Wert eins an, solange bis er erneut gedrückt wird, dann springt er zurück auf null. Über die Automap Software ist es möglich diese Toggles auch als Button zu verwenden. Im Gegensatz zu einem Toggle behält ein Button seinen Wert nicht, sondern gibt nur während des Drückens eine Eins aus. Beim Loslassen eines Buttons springt dessen Wert zurück auf Null. Im linken Bereich liegen über den Toggles/Buttons acht berührungsempfindliche Buttons. Diese funktionieren ähnlich wie ein Button, da sie nur für den Moment, in dem sie gedrückt werden, den Wert ausgeben und anschließend wieder auf Null springen. Die Berührungsempfindlichkeit<sup>9</sup> ermöglicht eine Beurteilung, wie intensiv auf den Button gedrückt wird. In Relation zu einem definierbaren Maximum wird dann ein Wert zwischen Null und Eins ausgegeben, wobei Eins die maximale Anschlagkraft ist. Über diesen Buttons befinden sich zwei Reihen mit jeweils acht Potentiometern<sup>10</sup>. Dies sind Drehbare Knöpfe, die ihren Wert erhöhen wenn sie im Uhrzeigersinn gedreht werden und entsprechend den Wert verringern, wenn sie gegen den Uhrzeigersinn gedreht werden. Dabei haben die Potis ein fest definiertes Minimum und Maximum, da sie sich nicht endlos drehen lassen. Sie lassen sich in etwa um 240° drehen. Links angeschlagen ist ihr Wert null, rechts angeschlagen das definierte Maximum. Die Potis in der obersten Reihe lassen sich endlos drehen. Damit ist eine feinere Justierung der Werte möglich. Im oberen rechten Bereich des Controllers finden wir noch Schieberegler, sie werden auch als Fader bezeichnet. Diese Fader haben ihr Minimum null, wenn sie ganz nach unten gezogen sind, ihr Maximum ist ganz oben. Dank der vielseitigen Steuermöglichkeiten des Controllers ist dieser sehr gut geeignet, um eine Veranstaltung bzw. Show zu steuern.

#### **II.IV. – Softwareanforderungen einer interaktiven Mehrfachprojektion**

Ähnlich wie bei den Hardwareanforderungen einer interaktiven Mehrfachprojektion sind auch die Softwareanforderungen einer solchen Projektion nicht immer identisch. Auch bei der Software ist es wichtig zu berücksichtigen, welche Art von Interaktion geplant ist und wie komplex die Mehrfachprojektion tatsächlich aussieht. Dennoch möchte ich in diesem Abschnitt einige softwarespezifische Grundlagen einführen, die für die Entwicklung einer interaktiven und /oder Mehrfachprojektion relevant sein können.

---

<sup>8</sup> Toggle: aus dem Englischen für umschalten, schalten, hin und herspringen

<sup>9</sup> Berührungsempfindlichkeit: oder auch Anschlagdynamik ; engl: touch sensitivity

<sup>10</sup> Potentiometer: kurz: Poti; deutsch: Drehregler

### II.IV.I. – Erstellen und Abspielen von Content

Eine Software für eine interaktive Mehrfachprojektion muss in jedem Fall in der Lage sein den Projektionscontent wiederzugeben. In der Regel liegt Projektionscontent als Videodatei oder Bildfolgesequenz vor. Die Software zum Ausspielen sollte also die gängigen Dateiformate unterstützen. Dazu gehören u.a. .mp4, .mov, .avi und DPX-, PNG sowie JPEG-Bildsequenzen. Zusätzlich sollte die Software die gängigen Audioformate wie .wav oder .mp3 unterstützen, da zu einer guten Projektion auch ein entsprechendes Sounddesign gehört.

Neben dem Abspielen des Contents zeichnen sich eine gute Software durch vorgefertigte Filter und Effekte aus. Besonders wenn die Show interaktiv gestaltet werden soll, muss während der Projektion der Content veränderbar sein. Vorgefertigte Filter und Effekte können solche Veränderungen darstellen. Eine Software, die ein Video lediglich abspielen kann, ist ungeeignet für eine Interaktion.

### II.IV.II. – Projektions-Mapping

Mapping beschreibt im Projektionskontext eine Vorverzerrung des Projektionsinhalts, um diesen auf eine bestimmte Fläche passend zu projizieren. Da Projektoren grundsätzlich so gebaut sind, dass sie ein rechteckiges Bild auf eine senkrecht zum Projektor ausgerichtete Fläche projizieren, muss immer dann eine Vorverzerrung vornehmen, wenn die zu projizierende Fläche nicht mehr senkrecht zum Projektor ausgerichtet ist.

#### II.IV.II.I. – Trapezkorrektur

Die einfachste Form des Projektions-Mappings ist die Trapezkorrektur. Sie kommt dann zum Einsatz, wenn der Projektor nicht senkrecht zu einer ebenen Projektionsfläche positioniert ist. Nehmen wir an, der Projektor steht auf dem Boden und projiziert an eine Wand, dann entspricht das projizierte Bild einem Trapez. Grund ist, dass die Entfernung der oberen Bildkante zum Objektiv des

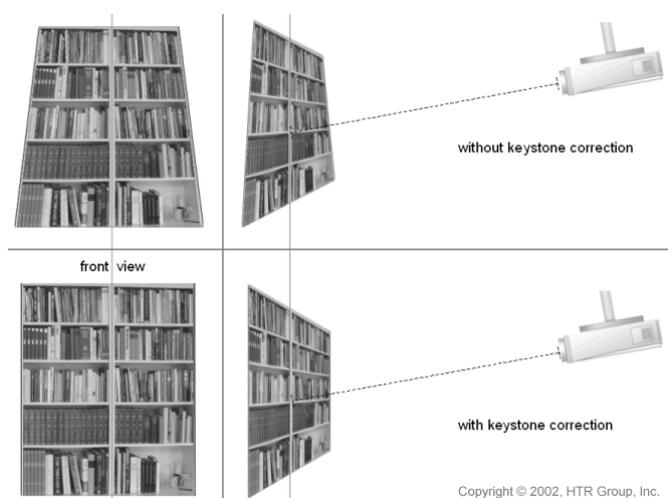


Abbildung 2.11. – Y-Achsen Trapezkorrektur

Projektors länger ist, als die Entfernung zwischen der unteren Bildkante und dem Objektiv. Um dennoch ein rechteckiges Bild projizieren zu können, muss der Projektor das Bild verzerren. Da der Projektor das Bild verzerrt, bevor es auf die Wand trifft, auf der das Bild ebenfalls verzerrt wird, wird hierbei von einer Vorverzerrung gesprochen. Die Vorverzerrung entspricht dabei genau der invertierten Verzerrung der Projektionsfläche. In diesem Fall wird von einer Y-Achsen Trapezkorrektur gesprochen, da die Entfernungsdifferenz auf der Y-Achse vorliegt. Entsprechend wird von einer X-Achsen Trapezkorrektur gesprochen, wenn die Entfernung der linken zur rechten Bildkante nicht übereinstimmen.

### II.IV.II.II. – Komplexes Projektions-Mapping

Das Projektions-Mapping wird immer dann deutlich aufwendiger, wenn die Fläche, welche bespielt werden soll, nicht mehr eben ist. In Abbildung 2.12. besteht die Projektionsfläche aus einer gleichseitigen Pyramide, bei der die Grundfläche viereckig ist und nach oben zeigt. Die Projektionsfläche teilt sich hierbei auf vier gleichseitige Dreiecke auf, wovon jeweils zwei Dreiecke von einem Projektor bespielt werden sollen.

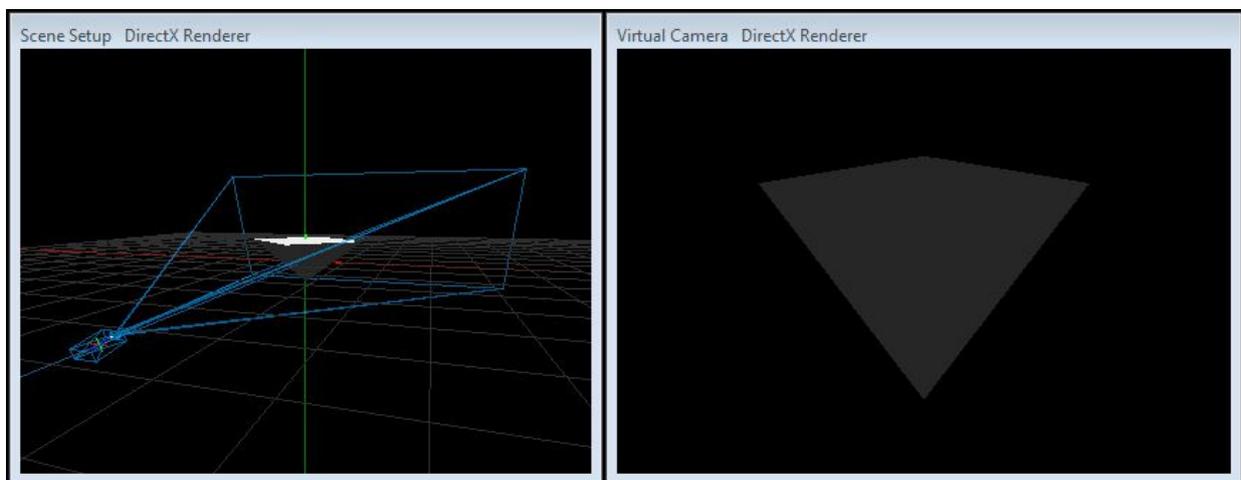


Abbildung 2.12. – Projektions-Mapping am Beispiel einer Pyramide, virtuell simuliert in VVVV.

Im linken Bild ist die Position des Projektors zu erkennen. Dabei ist zu sehen, dass die Pyramide mit einer Ecke zum Projektor zeigt. Im rechten Bild ist die Pyramide aus Sicht des Projektors zu sehen. Um am Ende also exakt auf die Pyramide zu projizieren, muss der Content des Projektors exakt auf den Bereich des grauen Vierecks im rechten Bild passen. Die Anforderung hierbei ist es, den Content so einzurichten, dass dieser auf die zwei Dreiecke der Pyramide passt. Man könnte den Content also bereits in der entsprechenden Dreiecksform anlegen, wie die Dreiecke oben dargestellt sind. Dies wird in der Regel jedoch nicht gemacht.

Tatsächlich wird der Content als gleichseitiges Dreieck angelegt, denn nur so hat man schon während der Vorbereitung eine konkrete Vorstellung, wie das Bild während der Projektion erscheint, denn spätestens dann wird der Content einem gleichseitigen Dreieck entsprechen.

### **II.IV.II.III. – Anforderungen eines Projektions-Mappings**

Während eine Trapezkorrektur häufig noch durch einen guten Projektor erfolgen kann, benötigt man für ein komplexeres Mapping eine entsprechende Software, um das Mapping durchzuführen. Die Projektion auf eine Pyramide mit dreieckigen Flächen ist dabei noch ein recht einfaches Projektions-Mapping. Extrem anspruchsvolles Projektions-Mapping finden wir z.B. in der Automobilindustrie. Dort werden zum Teil ganze Autos als Projektionsfläche genutzt.

Um letztendlich ein gutes Projektions-Mapping durchführen zu können, sollte die Software ein Mapping-Tool integriert haben. In der Regel sind solche Tools recht einfach zu bedienen und bieten einfache Möglichkeiten der Vorverzerrung via Drag & Drop<sup>11</sup>. Das bedeutet man könnte im Beispiel der Pyramide die einzelnen Ecken des Dreiecks per Mausklick so verschieben, dass sie exakt auf die Geometrie passen. Die Software errechnet anhand der verschobenen Eckpunkte die entsprechende Vorverzerrung.

### **II.IV.III. - 3D-Objekte**

Bei interaktiven Projektionen und Projektions-Mapping wird häufig in 3D-Räumen gearbeitet. 3-dimensionaler Content sieht häufig sehr beeindruckend aus und eignet sich gut für solche Projektionen. Außerdem bietet ein 3D-Raum die Möglichkeit (vgl. Kapitel II.III.) das Projektions-Mapping am PC zu visualisieren und somit z.B. Positionen von Projektoren schon im Vorfeld genau planen zu können.

Im Zusammenhang mit interaktiven Projektionen kommen häufig 3-dimensionale Elemente bzw. Effekte zum Einsatz. Zum Beispiel könnte in Abhängigkeit von der Position des Betrachters sich ein Objekt drehen oder skalieren. Für solche Anwendungen ist es sinnvoll, dass die Projektionssoftware 3D-Formate wie .obj, .3ds oder .dae unterstützen, damit uneingeschränkt mit den 3D-Objekten gearbeitet werden kann. 3D-fähige Software unterstützt meist auch virtuelle Projektoren und Kameras, wodurch sich viele kreative Anwendungsmöglichkeiten ergeben.

---

<sup>11</sup> Als Drag & Drop wird im PC-Bereich das Anklicken und Ziehen von Punkten oder Dingen in neue Bereiche bezeichnet.

### II.IV.III.I. – Live-Animation von 3D-Objekten

Für interaktive Anwendungen muss der Content live angepasst und verändert werden. Das ist verhältnismäßig simpel, wenn man davon ausgeht, dass Farben verändert oder Elemente bewegt werden. Deutlich schwieriger ist es 3D-Objekte live zu animieren.

Die Animation von 3D-Objekten findet im Regelfall in einer speziellen 3D-Software, wie Cinema 4D oder Blender statt. Soll die Animation jedoch interaktiv sein, so kann die Animation nicht im vorher produziert werden, sondern muss live realisiert werden. Angenommen das Ziel ist, die Bewegungen einer realen Person auf ein 3D-Objekt zu übertragen, sodass sich das 3D-Objekt genau so verhält, wie die Person in der Realität, dann muss sich das 3D-Objekt nicht nur bewegen, sondern auch verformen. Damit sich ein 3D-Objekt entsprechend verformen lässt, muss es sogenannte Skeletal Data<sup>12</sup> enthalten. Diese Daten kann man sich in etwa wie ein Skelett in dem 3D-Objekt vorstellen. Dabei lässt sich das Skelett individuell für eine 3D-Figur anpassen. Um das Skelett mit der 3D-Figur zu verknüpfen, muss definiert werden, welche Teile der 3D-Figur zu welchem Knochen<sup>13</sup> gehören. Das Verknüpfen zwischen dem Mesh<sup>14</sup> und den Bones nennt sich Skinning.

Nur wenn eine 3D-Figur ein Skinning und ein Skeleton implementiert hat, kann diese individuell animiert werden. Dazu muss natürlich auch die Software die entsprechenden Daten unterstützen. Dateitypen, die solche Daten unterstützen, sind u.a. .x, .fbx oder .dae. Dabei ist das .dae, welches als Collada bezeichnet wird ein sehr vielfältiges Format, da es eine Menge an Daten, wie Texturen, Mesh, Animation und Skinning-Daten unterstützt.

### II.IV.IV. - Audio-Reactive Content

Audio-reactive Content verändert sich in Abhängigkeit zum Sound. Es ist also keine Interaktion zwischen einer Person und der Software, sondern eine Interaktion zwischen dem Bild und dem Ton, wie z.B. die Visualisierung eines Equalizers.

Damit eine Software audio-reactive Content unterstützt, muss diese Software zum einen Zugriff auf die abgespielten Audiodateien haben und zum anderen eine Analyse der Audiodateien ermöglichen. Dabei kann die Audiodatei von der Software selbst abgespielt werden oder als Signal in die Software geroutet<sup>15</sup> werden. Um die Audiodateien z.B. mit einer Animation zu verknüpfen, müssen diese analysiert werden. In der Regel wird dies mit Hilfe

---

<sup>12</sup> Dt: Skelett-Informationen

<sup>13</sup> engl. Bezeichnung: Bone

<sup>14</sup> Als Mesh wird die Oberflächenstruktur einer 3D-Figur bezeichnet.

<sup>15</sup> Als Routing bezeichnet man in der Digitaltechnik das Leiten von Signalen.

einer FFT<sup>16</sup> gemacht. Die FFT filtert dabei die Audiodateien in verschiedenen Frequenzbereichen und gibt anschließend die mittlere Amplitude, also die Lautstärke, des entsprechenden Frequenzbereichs aus.

Mit diesen Werten lassen sich anschließend Animationsparameter oder Effekte bzw. Filter dynamisch zur Musik anpassen und regulieren.

### **II.IV.V. – Netzwerkanbindungen**

Die letzte Anforderung an eine Software zur Umsetzung von interaktiven Projektionen ist die Unterstützung von gängigen Protokollen zum Datenaustausch, wie dem MIDI Protokoll. Andere gängige Protokolle zum Datenaustausch in dieser Branche sind WindowsTouch, OSC und TUIO.

Externe Systeme von Drittanbietern, wie z.B. das radarTOUCH-System der Lang AG, analysieren Daten durch eigene Software, bereiten diese auf und senden sie dann über bestimmte Netzwerkwerkprotokolle zur nächsten Software. Die Unterstützung von solchen Netzwerkanbindungen kann dadurch zu einer Systemfrage werden, denn wenn ein bestimmtes System zum Einsatz kommen soll, so muss die Software auch die Daten von diesem System empfangen können.

---

<sup>16</sup> Fast Fourier Transformation

# Kapitel III

---

## EINFÜHRUNG IN SOFTWARESYSTEME

In diesem Kapitel werden verschiedene Softwareanwendungen vorgestellt. Wie in Kapitel II dargestellt, sind die Anforderungen an eine Software zur Realisierung einer interaktiven Mehrfachprojektion sehr vielseitig.

Im Folgenden werde ich erläutern, welche Softwaresysteme sich bereits am Markt etabliert haben und anschließend begründen, warum ich den Schwerpunkt dieser Arbeit auf den Vergleich zwischen Ventuz und VVVV gelegt habe. Die Vielseitigkeit der interaktiven Mehrfachprojektion beruht darin, dass hier zwei grundlegende Funktionen kombiniert werden sollen. Mehrfachprojektionen gibt es bereits seit vielen Jahren und mit der Zeit haben diese einen eigenen Markt erschaffen. Aufgrund dieses Marktes wurden Software- sowie Hardwarelösungen entwickelt, welche sich etablierten. Ebenso verhält es sich mit interaktiven Anwendungen, die ihren Ursprung in der Gamesbranche haben.

Welche Eigenschaften ein etabliertes Ausspielsystem in Bezug auf eine interaktive Mehrfachprojektion hat, möchte an dieser Stelle genauer darstellen. Außerdem gehe ich auf die Stärken und Schwächen einer Game-Engine ein.

### III.I. – Ausspielsysteme und Medienserver

Ausspielsysteme bzw. Medienserver sind softwarebasierte Systeme, die ihren Schwerpunkt im Abspielen von Content haben. Sie unterstützen dabei eine Vielzahl von Videoformaten bis hin zu 3D-Objekten. Die Besonderheit dieser Systeme liegt darin, dass sie entwickelt wurden, um komplexe Videosetups zu bespielen, z.B. eine Softedgeprojektion oder großformatige LED-Walls.

Über integrierte Tools kann der Nutzer ein individuelles Videosetup anlegen; er definiert in der Software u.a. wie viele Projektoren oder Videowände bespielt werden sollen, welche Auflösung das Setup hat. Diese Systeme kommen entsprechend immer dann zum Einsatz, wenn die Möglichkeiten von einem einfachen Ausspielsystem überschritten werden. Dabei muss der Medienserver präzise auf die Wiedergabe-Hardware angepasst werden, weshalb einige Hersteller die Software zusammen mit Ihrer Hardware vertreiben, z.B. der Pandoras Box Server von Coolux. Für extreme Videosetups gibt es die Möglichkeit Medienserver eines

Typs miteinander zu verknüpfen, sodass eine Show mit beispielsweise zwölf Projektoren von drei Medienservern parallel gerendert wird; dabei bespielt jeder Server vier Projektoren. Alle Medienserver sind dazu miteinander in einem Netzwerk verknüpft und bekommen über ein zentrales System die Anweisungen, welchen Content sie rendern.

Resolume Arena 5 ist ein solcher klassischer Medienserver. In Abbildung 3.1. wird das User Interface von Arena 5 gezeigt. Anhand des User Interfaces möchte ich die Arbeitsweise in solchen Medienservern kurz erläutern.



Abbildung 3.1. – User Interface von Resolume Arena 5

Die Abbildung 3.1. zeigt, dass Arena 5 ein layerbasiertes<sup>17</sup> Ausspielsystem ist. In der oberen Hälfte des User Interfaces sind drei Layer angelegt, welche übereinander gerendert werden. Dabei ist der Überblendungsmodus, also wie diese Ebenen sich aufeinander auswirken, frei wählbar. In diesem Fall werden sie auf einander addiert (Add). Man kann beliebig viele Ebenen erstellen und für jede Ebene lassen sich verschiedene Grundeinstellungen definieren, z.B. die Deckkraft, aber auch spezielle Effekte, die die ganze Ebene betreffen.

Für jede Ebene kann man eine eigene Playlist erstellen. Es ist außerdem möglich Live-Inputs hinzuzufügen, sofern die Hardware von Arena 5 unterstützt wird.

In der unteren Hälfte des User-Interfaces findet man die Preview- und Output Monitore. Rechts neben diesen ist das Einstellungsfenster. Hier lassen sich die grundlegenden

<sup>17</sup> Layer: dt. : Ebene

Kompositionseinstellungen konfigurieren. Als Komposition wird in diesem Fall die Summe aller Ebenen bezeichnet. Es ist aber auch möglich Einstellungen nur für eine Ebene vorzunehmen. Dies ermöglicht z.B. alle Clips einer Ebene in ihrer Position, Drehung oder Skalierung anzupassen, indem die Ebeneneinstellungen entsprechend verändert werden. Es stehen dazu eine Vielzahl an Effekten zur Verfügung, die alle entweder einem einzelnen Clip, einem Live-Input, einer Ebene oder der Komposition, also allen Clips und Inputs, zugeordnet werden können. Die Effekte betreffen u.a. die Farb- und Positionseinstellungen, Audioeffekte und Filter oder andere Effekte wie einem „Kaleidoscope“-Effekt.

Neben dem Einstellungsfenster für die Komposition und Layer ist es möglich Einstellungen für nur einen einzelnen Clip vorzunehmen. In der rechten unteren Ecke findet sich dann das Browser Fenster. Von hier können Video- und Audiodateien per Drag & Drop hinzugefügt werden.



**Abbildung 3.2. – Output Setup in Resolume Arena 5**  
Zu sehen ist eine Addition von verschiedenen Videoebenen und einem Live-Kamerabild.

Die Grundeinstellungen der Komposition lassen sich im Menü konfigurieren. Dabei ist es möglich die Kompositionsgröße pixelgenau zu definieren. Neben den einzelnen Effekten haben die Ebenen in Arena 5 auch noch die Funktion, dass ihnen beim Rendern eigene Positionen zugeordnet werden können. Wie die Abbildung 3.2 zeigt, können hier mehrere Screens hinzugefügt werden. Jeder Screen entspricht dabei einem Ausspielgerät, wie einem

Projektor, einem Monitor oder einer LED-Wall. Einem Screen wird von Arena 5 automatisch ein Slice zugeordnet, es ist aber auch möglich mehrere Slices für einen Screen einzurichten. Diesen Slices können nun Quellen zugeordnet werden. Das kann entweder die Komposition oder eine einzelne Ebene sein. Dadurch kann man verschiedene Ebenen auf verschiedenen Screens darstellen oder mehrere Ebenen auf einem Screen individuell anordnen.

Es besteht außerdem noch die Möglichkeit Masken auf die einzelnen Slices anzuwenden. Unter dem Reiter „Output-Transformation“ lässt sich via Drag & Drop für jeden einzelnen Slice eine eigene Verzerrung einrichten. In dem Beispiel von Abbildung 3.3 liegt ein rechteckiges Video zu Grunde. Mit Hilfe einer Maske wird aus dem Video nur ein dreieckiger Teil des Videos gezeigt. Dieses Dreieck wird anschließend verzerrt.



Abbildung 3.3. – Output Verzerrung in Resolume Arena 5

Medienserver wurden entwickelt, um auch komplexe Projektionen schnell und einfach realisieren zu können. Allerdings fehlen einem Medienserver interaktive Elemente. Zwar lassen sich die meisten Server via MIDI steuern und unterstützen ggf. eine Audioanalyse, jedoch sind die interaktiven Elemente nicht vergleichbar mit denen einer Game-Engine. In Arena 5 muss der Content vorproduziert werden und lässt sich anschließend nur durch Effekte anpassen. Zudem lässt sich kein eigener Content generieren, sodass man immer auf vorproduzierte Grafiken oder Videos angewiesen ist. Ein weiterer bekannter Medienserver ist der Pandoras Box Server von Coolux.

### III.II. – Game-Engines

Game-Engines kommen, wie der Name bereits sagt aus dem Bereich der Spielkonsolen und Computerspiele. Sie werden entwickelt um Spiele realisieren zu können. Einige Game-Engines bieten dem Nutzer zusätzlich auch die Möglichkeit selber Spiele zu entwickeln. Heutzutage arbeiten fast alle Spiele in einer 3D-Umgebung und laufen auf verschiedensten Systemen. Neben den PC-Betriebssystemen sind auch die Smartphone-Systeme Android und OSX, sowie Spielekonsolen wie die Playstation oder die XBOX wichtige Abspielsysteme von Spielen.

Spiele sind grundsätzlich immer interaktiv, da das System auf Eingaben des Nutzers reagiert. Daher müssen alle Spiele als Echtzeitgrafik produziert werden. Game-Engines sind dementsprechend sehr gut für interaktive Anwendungen geeignet. Moderne PC- oder Konsolenspiele bieten häufig sehr gute Grafik- und Renderleistungen, besonders in 3D-Umgebungen. Im Grunde sind Game-Engines dazu konzipiert, um interaktiven Content flüssig und optisch anspruchsvoll darzustellen. Zu den bekanntesten Game-Engines zählen Unity, Unreal und die CryENGINE.

Wie in Abbildung 3.4 zu sehen, ist die Unity-Engine einer 3D-Software wie Cinema4D oder Maya nachempfunden. In dieser Entwicklungsumgebung kann der Nutzer nun seine Spielwelt entwickeln. In der Hierarchy können alle Elemente als 3D-Objekte eingefügt werden, dabei hat Unity bereits eine umfangreiche Bibliothek an vorgefertigten 3D-Objekten. Mit Hilfe aller Objekte entwickelt der Nutzer eine individuelle 3D-Szene, in der das Spiel stattfindet. Natürlich können auch eigene Objekte importiert werden.

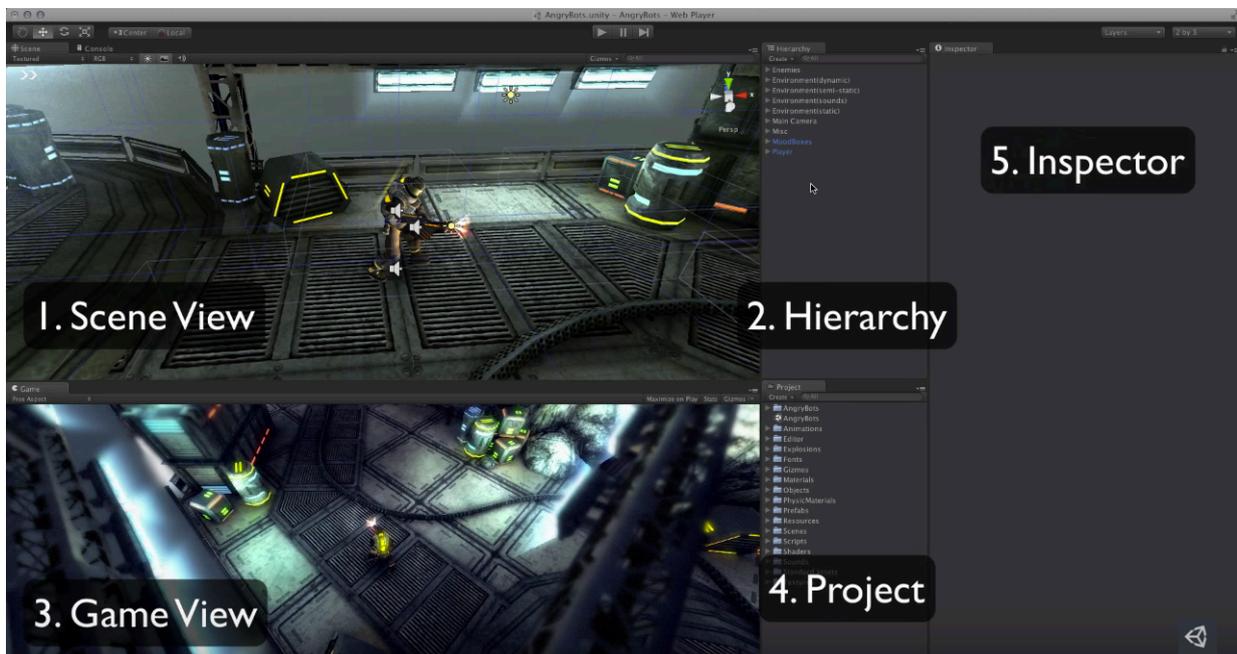


Abbildung 3.4. – Unity-Engine User Interface

Eine Stärke von Game-Engines ist der Umgang und die Animation von 3D-Inhalten. Dazu benutzen diese Programme sogenannte Physik-Engines, um Animationen realer darzustellen. Dazu werden physikalische Eigenschaften und Kräfte wie z.B. von Materialien in die Animation einberechnet. Alle gängigen 3D-Formate und Eigenschaften, wie z.B. Skeleton Animation, werden ebenfalls durch Game-Engines unterstützt.

Im Gegensatz zu den Medienservern bietet eine Game-Engine kaum Möglichkeiten seine Output-Settings anzupassen; es werden lediglich die Standard-Monitorauflösungen unterstützt.

### **III.III. – Hybride Systeme Ventuz und VVVV**

Für eine interaktive Mehrfachprojektion müssen also die Funktionen einer Game-Engine mit denen eines Medienservers kombiniert werden. Die Programme Ventuz und VVVV versprechen genau diese Kombination. Auf der einen Seite implementieren sie 3D-Funktionen und anspruchsvolle Grafiksysteme, auf der anderen Seite aber auch uneingeschränkte Möglichkeiten in der Anpassung der Outputs.

In den folgenden Kapiteln möchte ich also genau diese beiden Programme hinsichtlich interaktiver Mehrfachprojektion vergleichen. Dazu werden zunächst beide Programme vorgestellt und anschließend dargestellt, wie man mit ihnen arbeitet. Da sich beide Programme besonders in der grundsätzlichen Arbeitsweise sehr ähneln, halte ich diesen Vergleich für angemessen.

# Kapitel IV

---

## VENTUZ UND VVVV

In diesem Kapitel wird zunächst beschrieben, wie die Arbeitsweise von Ventuz und VVVV funktioniert. Beide Programme setzen auf eine hybride Arbeitsweise mit dem Schwerpunkt einer grafischen Programmierung. Da diese recht speziell ist, erläutere ich zunächst, wie sie funktioniert und stelle anschließend die Vor- und Nachteile dar. Dabei werde ich auch berücksichtigen, dass Ventuz aus verschiedenen Komponenten besteht, während in VVVV nur eine einzelne Anwendung verwendet wird.

Nach dieser Einführung werden in vier Abschnitten die äußeren Gegebenheiten beider Systeme dargestellt. Dabei wird die Installation, das User Interface, die Performance und der Support betrachtet. Obwohl hier nicht direkt das Arbeiten mit den Programmen betrachtet wird, spielen diese Punkte dennoch eine wichtige Rolle für das Arbeiten mit einem dieser Systeme.

### Kapitel IV.I. – Erstellung von Anwendungen in Ventuz und VVVV

Im Gegensatz zu herkömmlichen Medienservern oder Game-Engines funktioniert die Erstellung einer Anwendung in Ventuz oder VVVV etwas anders, denn beide Programme arbeiten mit einer sogenannten grafischen Programmieroberfläche. Durch diese Programmieroberfläche wird es dem Anwender ermöglicht, eigene Programmstrukturen zu entwickeln. Dazu stehen eine Vielzahl an Bausteinen, sogenannte Nodes, zur Verfügung, die miteinander verknüpft werden können.

Jede Node besitzt In- und Outputs. Über die Inputs werden der Node Daten bereitgestellt, die diese je nach ihrer Funktionalität verarbeitet. Anschließend wird das Ergebnis an den Outputs ausgegeben. Abbildung 4.1. zeigt einen VVVV-Patch<sup>18</sup>, welcher als Beispiel für diese Programmiermethode erläutert werden soll.

---

<sup>18</sup> Als Patch wird eine in VVVV erstellte Programmstruktur bezeichnet.

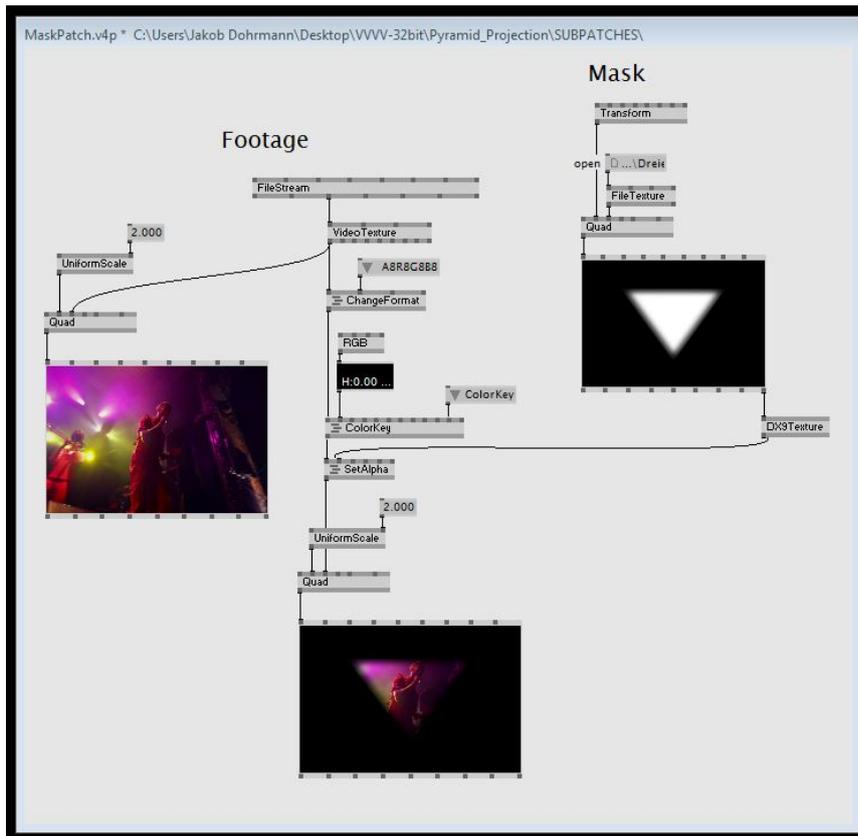


Abbildung 4.1. – Mask-Patch in VVVV

Die kleinen grauen Vierecke sind die Nodes, ihre Inputs sind die Punkte oben, die Punkte unten sind die Outputs. Die „VideoTexture“-Node hat eine Verknüpfung von einem Input zu der „FileStream“-Node, welche eine Mediendatei von dem lokalen PC lädt. Die „VideoTexture“-Node wandelt nun diese Videodatei in eine Textur um und gibt sie an zwei weitere Nodes weiter. Die „Quad“-Node ist im Prinzip ein Viereck auf das nun die Textur dargestellt wird. Die „Quad“-Nodes geben einen Layer aus, dieser wird von einer „Renderer“-Node letztendlich gerendert, also visualisiert.

Die „VideoTexture“-Node gibt diese Textur ebenfalls an eine zweite Node, die „ChangeFormat“-Node, weiter. Das Format der Textur wird durch diese Node geändert und anschließend wird ein Alphawert<sup>19</sup> hinzugefügt. Der Alphawert der Textur wird nun durch den Alphawert des Dreiecks ersetzt. Das Ergebnis wird erneut über eine „Quad“-Node an einen Renderer übergeben und visualisiert.

<sup>19</sup> Alphawerte sind Transparenzwerte von Bilddateien/Texturen, dabei haben schwarze Pixel einen Alphawert von null. Durch den Alphawert werden diese Pixel nicht als schwarz dargestellt, sondern die Pixel der Ebene darunter werden sichtbar.

Der in Abbildung 4.1. gezeigte Patch hat die Funktion, ein Video zu laden und dieses Video mit einer Alpha-Maske zu versehen.

Das Videobild ist im linken Renderer zu sehen; die Maske, welche verrechnet werden soll, ist in einem Renderer rechts zu sehen. Das endgültige Bild ist im Renderer unten mittig dargestellt.

Durch diese Methode der Programmierung können auch unerfahrene Programmierer eigene Programme entwerfen, ohne eine Programmiersprache zu beherrschen. Die Node-Bibliothek der Programme kann in beiden Fällen durch Script-Nodes erweitert werden, sodass erfahrene Programmierer ihre eigenen Nodes mit C# programmieren und dem Programm hinzufügen können.

Die grafische Programmierung bietet den enormen Vorteil, dass Nutzer fast alle beliebigen Teile seines Programms miteinander verknüpfen kann. Alle Inhalte können sich gegenseitig beeinflussen, wodurch sich interaktive Anwendungen sehr gut realisieren lassen. Auch Effekte lassen sich so eigens erstellen. Zudem arbeiten Ventuz und VVVV im 3D-Raum und unterstützen grundsätzlich 3D-Modelle. Beide bieten außerdem die Möglichkeit individuelle Rendereinstellungen vorzunehmen, sodass auch komplexe Ausspielsetups realisierbar sind.

In meinem Testsetup, welches in den Kapiteln fünf und sechs beschrieben wird, gehe ich noch weiter auf die Arbeit mit beiden Programmen ein und vergleiche die 3D-Fähigkeiten, die Rendermöglichkeiten, sowie weitere Vergleichspunkte anhand eines konkreten Beispiels.

## **Kapitel IV.II. - Die Installation**

Vorweg sei gesagt, dass sich beide Programme nur unter aktuellen Windows Versionen installieren lassen. Dies ist vermutlich darauf zurückzuführen, dass beide Programme auf das umfangreiche NET Framework von Microsoft zurückgreifen, in dem zahlreiche Programmiersprachen implementiert sind. Außerdem basieren beide Programme auf DirectX, welches ebenfalls nur für Windows entwickelt wurde.

### **IV.II.I. – Ventuz Installation**

Die Installation von Ventuz funktioniert unkompliziert über eine .exe Datei, welche auf der Ventuz-Website frei zum Download bereit steht. Mit dieser Datei werden alle Komponenten, also der Ventuz Designer, der Ventuz Configuration Editor und der Ventuz Director, installiert.

Auf die einzelnen Komponenten wird im Abschnitt „User Interface“ weiter eingegangen.

### **IV.II.II. – VVVV Installation**

Die Installation von VVVV ist an dieser Stelle schon etwas umständlicher. Über die Website kann man sich eine .zip Datei herunterladen und auf dem PC entpacken. In diesem Ordner

findet sich dann eine Setup.exe Datei. Mit Hilfe dieser Setup.exe können alle Framework-Bibliotheken, welche von VVVV benötigt werden, heruntergeladen werden. Zusätzlich sollte man sich zur Verwendung von VVVV noch das entsprechende Add-On Pack herunterladen, welches das Programm um eine Vielzahl an Nodes und Funktionen erweitert. Dieses muss in die Ordnerstruktur von VVVV am korrekten Platz eingefügt werden.

## Kapitel IV.III - Das User Interface

Das User Interface ist die Benutzeroberfläche eines Programms und somit der Zugang des Benutzers zum Programm. Vor allem neue Anwender brauchen das User Interface, um sich in einem Programm zurechtzufinden. Dabei sollte es möglichst selbsterklärend und intuitiv gestaltet sein.

### IV.III.I. - Ventuz Userinterface

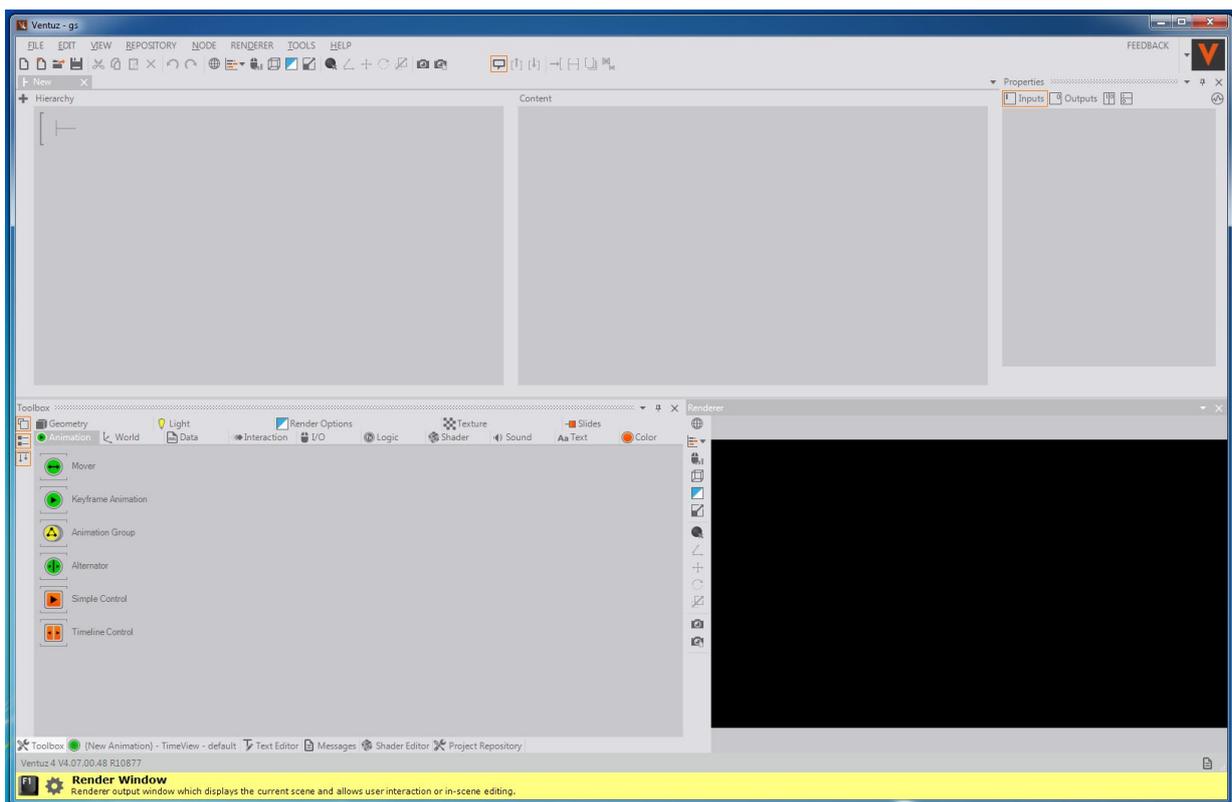


Abbildung 4.2. - User Interface Ventuz Designer

Das User Interface in Ventuz entspricht dem, was ich von einer entsprechend teuren Software erwarte. Beim Öffnen des Ventuz Designers erwarten einen fünf Fenster mit entsprechenden Beschriftungen: „Hierarchy“, „Content“, „Toolbox“, „Properties“ und „Renderer“. Dazu gibt es eine Leiste, in der die wichtigsten Funktionen anwählbar sind, z.B. „New Scene“,

„Statistics“ oder „Fullscreen“ sind hier per Mausklick wählbar. Alle weiteren Funktionen finden sich in der Kopfzeile unter den Reitern „File“, „Edit“, „View“, „Repository“, „Node“, „Renderer“, „Tools“ und „Help“ wieder.

Ventuz bietet neben dem Designer, in dem sämtliche Programmierarbeit geleistet wird und somit die eigentliche Anwendung entsteht, noch zwei weitere Programmkomponenten. Zum einen gibt es den Ventuz Configuration Editor. In diesem werden alle Einstellungen zur Hardware angepasst, z.B. werden hier Inputkarten hinzugefügt und Input-Formate definiert, oder die Rendereinstellungen und das Mapping angepasst. Auch Clusterfunktionen, also die Verknüpfung von mehreren Ventuz-Maschinen, wird über den Configuration Editor vorgenommen.

Zum anderen gibt es noch den Ventuz Director. Dieser eignet sich um große Shows mit wiederkehrenden Elementen zu steuern. Für diesen wird jedoch eine zusätzliche Lizenz benötigt. Für meinen Vergleich zwischen Ventuz und VVVV soll der Ventuz Director keine Beachtung finden, da er keine neuen Funktionalitäten bietet. Lediglich das Handling und die Steuerung werden durch den Ventuz Director vereinfacht.

#### IV.III.II. – VVVV User-Interface

Ein wirkliches User-Interface (vgl. Abbildung 4.3.) bietet VVVV eigentlich nicht, denn beim Öffnen von VVVV sieht man nur einen grauen Kasten. Ebenfalls etwas unüblich ist das Hauptmenü, welches man bei VVVV durch die mittlere Maustaste erreicht. Mit doppeltem Linksklick kommt man in eine Übersicht der Nodes, oder man sucht einfach den Namen mit Hilfe der Tastatur. Insgesamt dauert die Einfeldung in VVVV, aufgrund der ungewöhnlichen Bedienoberfläche, aber recht lang und ist ohne Hilfe durch Tutorials sehr aufwendig.

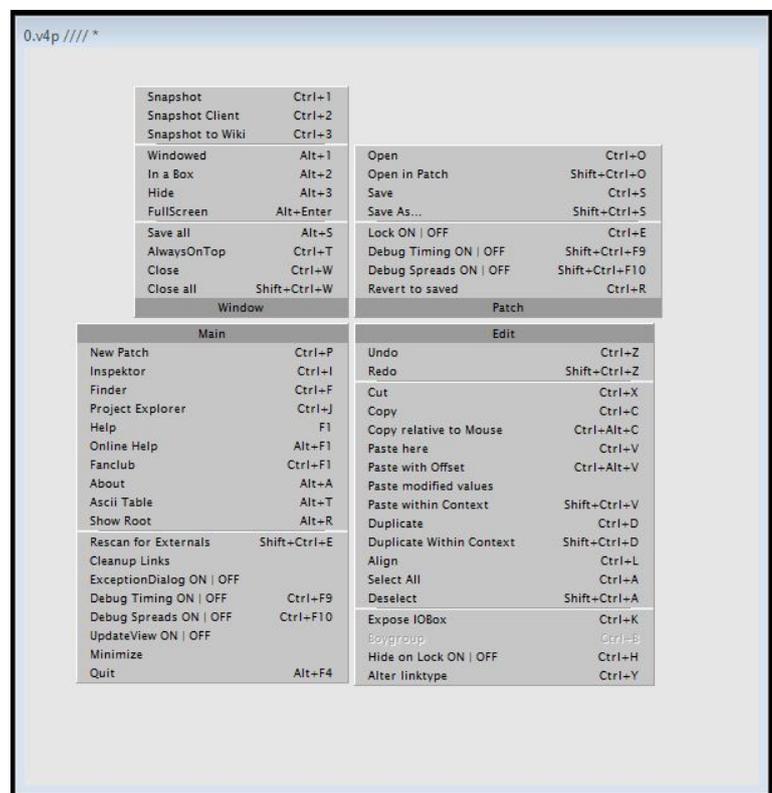


Abbildung 4.3. – VVVV User-Interface und geöffnetes Hauptmenü.

## **Kapitel IV.IV. – Performance und Stabilität**

Für Programme, die Echtzeitgrafik bieten, ist die Performance ein extrem wichtiger Punkt. Dabei setzen beide Programme auf DirectX, eine Schnittstelle zwischen Software und Hardware. DirectX ist eine sogenannte API<sup>20</sup>, die einem Programm ermöglicht die Hardware des PCs zu analysieren und entsprechende Rendervorgänge auf der Grafikkarte umzusetzen.

### **IV.IV.I. – Performance und Stabilität in Ventuz**

Ventuz arbeitet aktuell mit DirectX 9. Da bei Ventuz ein Entwicklerteam die gesamte Software gemeinsam entwickelt, sind die Abläufe und Softwarekomponenten sehr gut aufeinander abgestimmt. Alle Nodes arbeiten problemlos mit DirectX 9 zusammen.

### **IV.IV.II. – Performance und Stabilität in VVVV**

In VVVV kann man selbst entscheiden, ob man mit DirectX 9 oder DirectX 11 arbeiten möchte. DirectX wird mit jeder neuen Version leistungsfähiger, sodass DirectX 11 komplexere Berechnung ermöglicht als DirectX 9.

VVVV bietet also in diesem Fall grundsätzlich eine höhere Leistungsfähigkeit. Allerdings gibt es bei VVVV neben DirectX 11 auch noch den DirectX 9 und den GDI Renderer. Die verschiedenen Renderer setzen also auch entsprechend verschiedene Datentypen an ihren Inputs voraus, was bedeutet, dass nicht alle Nodes ohne Umstände miteinander kompatibel sind.

## **Kapitel IV.V. – Support**

Beide Programme bieten grundsätzlich ein umfangreiches Support-Angebot. Die beste Hilfe ist für mich meist die Hilfe-Szene bzw. der Help-Patch. Darin wird eine Node genauer und beispielhaft dargestellt und mit Kommentaren erklärt. Dieses Feature gibt es sowohl bei Ventuz als auch bei VVVV. Beide Programme bieten ebenfalls eine lange Liste an Video-Tutorials auf YouTube an. Alle grundlegenden Funktionen und Nodes werden hier sehr gut und ausführlich erklärt. Dadurch kann man die Tutorials nachprogrammieren und während des Programmierens die Software erlernen.

Für Ventuz gibt es ein komplettes User Manual im Internet auch hier wird noch einmal das wichtigste erläutert. Ebenso gibt es auch für VVVV ausführliche Dokumentation auf ihrer Website z.B. über Datentypen und Nodes.

---

<sup>20</sup> Application Programming Interface

Beide Programme bieten also eine sehr gute Selbsthilfe an, sodass es im Prinzip selten zu direkten Fragen kommen sollte. Dazu bieten sowohl Ventuz als auch VVVV ein Forum, in dem man Fragen posten kann. Sollte dies nicht zu einer hilfreichen Antwort führen, kann man versuchen die Entwickler persönlich anzusprechen. Ich habe von beiden Parteien zügig Antwort auf meine E-Mails bekommen.

Mit dem Support bin ich insgesamt sehr zufrieden und konnte zwischen den Programmen keinen Unterschied ausmachen.

# Kapitel V

---

## PRAKTISCHER VERSUCHSAUFBAU

Wie in Kapitel IV beschrieben, sind Ventuz und VVVV vielseitige Systeme. Trotz der vielen Gemeinsamkeiten ist es daher nicht möglich ein System als grundsätzlich besser zu deklarieren. Beide Systeme haben auch in Bezug auf eine interaktive Mehrfachprojektion Vor- und Nachteile. Daher habe ich mich entschieden den Vergleich dieser Systeme anhand eines konkreten Testsetups durchzuführen. Dieses stelle ich in diesem Kapitel dar und erkläre es.

### Kapitel V.I. – Die Projektionsfläche

Als Projektionsfläche (Abbildung 5.1.) habe ich eine gleichseitige Pyramide gewählt, die mit der Spitze nach unten zeigend aufgehängt wurde. Es sollten alle vier Seitenflächen der Pyramide bespielt werden.

Diese Art der Projektionsfläche erfordert, dass mindestens mit zwei Projektoren gearbeitet wird, da andernfalls die Pyramide nicht von allen Seiten bespielt werden kann. Es wäre auch denkbar mit vier Projektoren zu arbeiten.

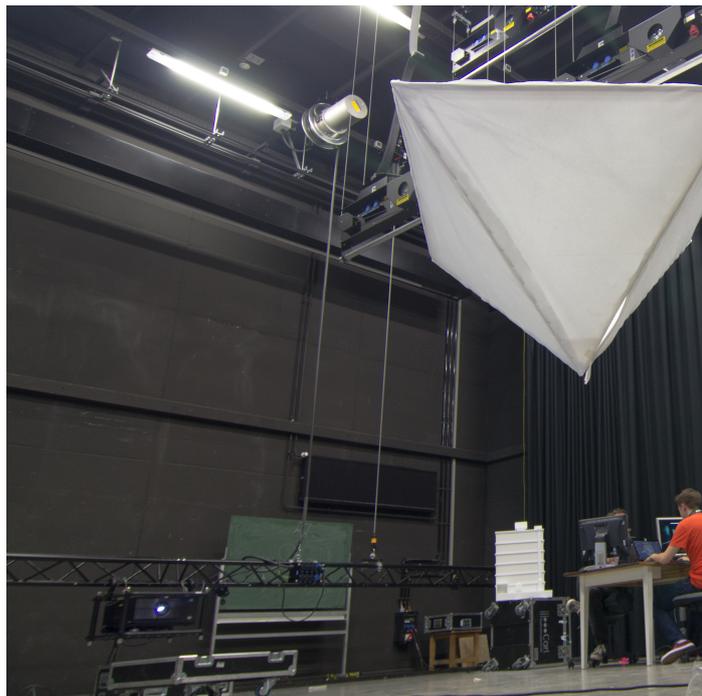


Abbildung 5.1. – Aufbau Pyramide + Barco RLM W8 Projektor

Allerdings ist es meiner Meinung

nach anspruchsvoller, mit nur zwei Projektoren zu arbeiten, da jeder Projektor zwei Seiten bespielen muss. Dadurch wird das Zusammenführen des Contents und das Mapping etwas komplexer wird, was in meinem Test einen intensiveren Umgang bedeutet. In der Praxis würde die Reduzierung der Projektoren zu geringeren Kosten führen.

Ein noch schwierigeres Mapping würde sich ergeben, wenn die Projektionsflächen gewölbt wären. Für eine Aussage über die Funktionalitäten der Mappingtools ist dies aber nicht unbedingt notwendig, sodass ich es bei ebenen Projektionsflächen belassen habe.

### **Kapitel V.II. – Abspielen von Audio- und Videodateien**

Das Abspielen von Audio- und Videodateien ist eine der wichtigsten Eigenschaften für eine Software zur Umsetzung einer interaktiven Mehrfachprojektion. Aber auch hier können kleine Unterschiede entscheidend werden.

Für meine Show dienten drei Musikvideos als Leitfaden durch die Show; sie sollten nacheinander abgespielt werden.

### **Kapitel V.III. – Content erstellen und bearbeiten**

Keine Projektion funktioniert ohne Content. Eine gute Software für Projektionen sollte also die Möglichkeit bieten, Content selbst zu erstellen und vorhandenen Content mit Effekten zu verändern. Gerade wenn die Projektion interaktiv sein soll, muss der Content innerhalb der Software veränderbar sein. Je mehr Möglichkeiten bestehen, desto besser.

Neben dem Abspielen der Musikvideos wollte ich eigenen Content gestalten, meist nur einfache Formen wie Dreiecke, die sich an die Pyramide anpassen.

### **Kapitel V.IV. – Interaktivität – Audio-Reactive Content**

Um Content abhängig vom Sound zu verändern, muss die Software die Audiospur analysieren. Dabei gilt, je mehr Daten zur Verfügung stehen, desto besser. Um meinen eigenen Content etwas spannender zu gestalten, sollten die Dreiecke durch die Musik beeinflusst werden, z.B. sollten sie sich abhängig von der Lautstärke der Musik verändern.

### **Kapitel V.V. – Interaktivität – Microsoft Kinect**

Um meine Projektion interaktiv zu gestalten, habe ich eine Microsoft Kinect eingesetzt. Sie ist die vermutlich populärste Tiefenbildkamera und somit schnell verfügbar.



Abbildung 5.2. – Microsoft Kinect

Das Ziel war es, die Bewegungen einer Person auf eine 3D-Figur live zu übertragen. Die 3D-Figur wurde auf der Pyramide visualisiert. Da die Microsoft Kinect nur eine Person zuverlässig tracken kann, wurde in dem Raum der Projektion ein Bereich gekennzeichnet, in dem sich nur eine Person aufhalten durfte. Dieser Bereich wurde von der Kinect erfasst.

### **Kapitel V.VI. – Interaktivität – radarTOUCH**

Eine weitere Form der Interaktivität sollte durch das synTouch-System realisiert werden. Da dieses System grundsätzlich in einer 2D-Ebene arbeitet, habe ich mich entschieden die Personen im Raum zu tracken und ihre Position im Raum auf die Pyramide zu übertragen. Da das Laser-System jedoch aus einem Punkt heraus funktioniert und Personen „Schatten“ werfen<sup>21</sup>, wurde die Ebene in einer Höhe von ca. zwei Metern installiert. Dadurch wurden nicht alle Personen getrackt, sondern nur in dem Fall, wenn die Personen einen Arm hob. Dadurch wurde die Zahl der potentiellen Fehler durch Schatten deutlich reduziert und der Besucher konnte aktiver auf die Projektion einwirken.

Um die Positionen der Nutzer auf die Pyramide zu übertragen, musste die Fläche des Laser-Systems auf die vier Seiten der Pyramide umgerechnet werden, wodurch eine komplexe logische Verknüpfung entstand.

Da dieses System über eine eigene Software ausgewertet wird und die Daten anschließend als TUIO-Protokoll weitersendet werden, kann hier sehr gut betrachtet werden, ob und wie die Software dieses Protokoll unterstützen.

### **Kapitel V.VII. – Steuerung – MIDI Controller**

Um die verschiedenen Effekte und Visualisierungen ein- und auszublenden und die Effekte anzupassen, habe ich die Show über einen MIDI Controller gesteuert. Das ist intuitiver und übersichtlicher als die Steuerung an einer PC-Oberfläche. Die Steuerung über MIDI sollte in jedem guten Programm für interaktive Shows implementiert sein und sich leicht realisieren lassen.

---

<sup>21</sup> Vgl. Kapitel II.III.III.

# Kapitel VI

---

## VERSUCHSERGEBNISSE

In diesem Kapitel werden die in Kapitel V vorgestellten Versuchskriterien direkt auf Ventuz und VVVV angewendet und kurz beurteilt. Für eine direkte Gegenüberstellung wird jedes Kriterium zuerst auf VVVV und direkt im Anschluss auf Ventuz bezogen; anschließend wird das nächste Kriterium betrachtet.

### Kapitel V.I. – Die Projektionsfläche - Rendering

#### VVVV:

Mit mehreren Projektoren zu arbeiten, ist in VVVV kein Problem. Für jeden Projektor eignet sich eine eigene „Renderer“-Node. In VVVV lassen sich beliebig viele Renderer erstellen. Durch die Node-basierte Programmierung ist es möglich einzelnen Renderern verschiedene Ebenen oder Effekte zuzuweisen. Die Möglichkeit mit mehreren Renderern zu arbeiten, ist sehr praktisch, da sich der Patch beliebig erweitern oder verkleinern lässt.

#### VENTUZ:

In Ventuz gibt es grundsätzlich nur einen Renderer, welcher die gesamte Szene visualisiert. Um mehrere Projektoren zu bespielen, müssen diese als Displaygruppe<sup>22</sup> angeordnet werden, sodass der Renderer über zwei Monitore bzw. Projektoren im Full-Screen angezeigt wird, z.B. im Bildformat 32:9. Dies wird jedoch nur von leistungsstarken Grafikkarten unterstützt. Um Ventuz entsprechend zu verwenden, muss also entsprechende Hardware bereitstellen.

### Kapitel V.II. – Die Projektionsfläche - Mapping

#### VVVV:

Grundsätzlich funktioniert ein Mapping in VVVV. Das Tool „BadMapper“ (Abbildung 6.1), welches im VVVV AddonPack enthalten ist, bietet die Möglichkeit Eckpunkte via Drag & Drop zu verschieben, sodass Flächen präzise angepasst werden können.

---

<sup>22</sup> Diese Funktion nennt sich bei AMD Grafikkarten Eyefinity, bei NVIDIA Mosaic oder nView.

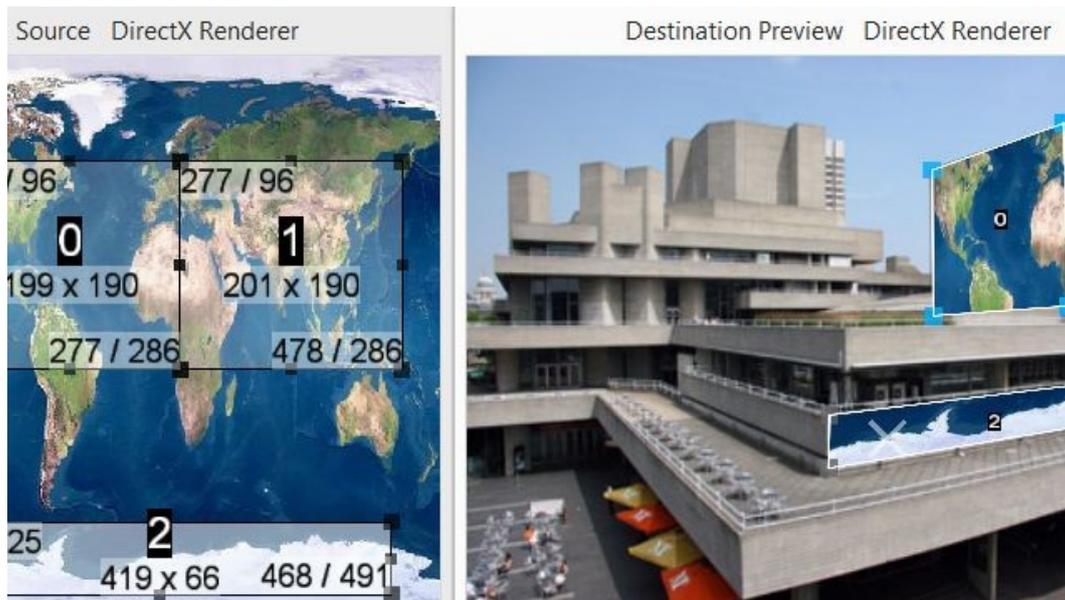


Abbildung 6.1. – BadMapper in VVVV

### VENTUZ:

Im Gegensatz zu VVVV wird ein Mapping in Ventuz über den Configuration Editor (vgl. Abbildung 6.2.) realisiert. Hier lassen sich komplexe Output Settings erstellen und in Presets<sup>23</sup> abspeichern. Die Einstellungsmöglichkeiten sind in Ventuz dabei umfangreicher und intuitiver. Die Auslagerung in den Configuration Editor schafft zusätzlich Übersichtlichkeit in der eigentlichen Szene.

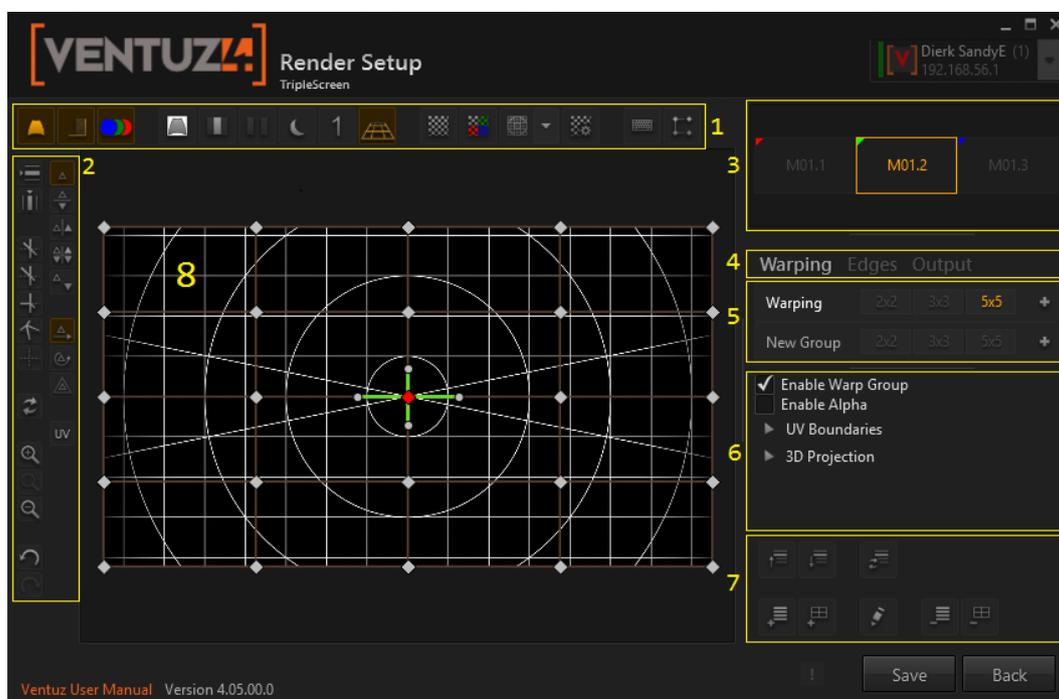


Abbildung 6.2. – Ventuz Render Setup im Configuration Editor

<sup>23</sup> Preset: Bezeichnet abgespeicherte Einstellungen für eine bestimmte Anwendung.

Im Vergleich zu VVVV lassen sich im Ventuz Configuration Editor mehr Punkte zu einer Fläche hinzufügen, sodass nicht nur die Eckpunkte, sondern auch Punkte dazwischen verzogen werden können. Dadurch lassen sich auch komplexe Mappings auf gewölbte Oberfläche oder gecurvten Leinwänden realisieren, während VVVV nur eine geradlinige Verzerrung ermöglicht.

Zudem bieten beide Programme die Möglichkeit ein Projektionssetup durch virtuelle Projektoren zu simulieren. Dadurch lässt sich ebenfalls ein Projektions-Mapping realisieren, ist in der Praxis aber recht umständlich, da sich der virtuelle Aufbau in der Realität meist nicht exakt nachbilden lässt, sodass kleine Ungenauigkeiten entstehen.

### **Kapitel V.III. – Abspielen von Audio- und Videodateien**

#### **VVVV:**

Das Abspielen von Audio- und Videodateien ist in VVVV über eine „FileStream“-Node möglich. Diese bietet u.a. eine Restart- und eine Search-Funktion, sodass die wichtigsten Features vorhanden sind. Einschränkungen gibt es in den Dateiformaten, da nicht alle gängigen Formate ohne Umstände unterstützt werden.

#### **VENTUZ:**

Ventuz unterstützt im Gegensatz zu VVVV mehr Dateiformate und macht es dem Nutzer dadurch einfacher Dateien abzuspielen. Darüber hinaus bietet Ventuz weitere Einstellungsmöglichkeiten, z.B. wie viele Frames gebuffert<sup>24</sup> werden, was zu einer höheren Zuverlässigkeit führt.

### **Kapitel V.IV. – Content erstellen und bearbeiten**

#### **VVVV:**

In VVVV stehen nur einfachste Formen, wie Linien, Würfel oder Kugeln, zur Verfügung. Wirklich anspruchsvolle 3D-Objekte lassen sich hier also nicht realisieren. Zwar lässt sich auch durch einfachste Formen eindrucksvoller Content entwickeln, dennoch würden mehr Möglichkeiten im Bereich 3D-Objekte die Programmfunktion sinnvoll erweitern.

Texturen sind sehr leicht und in fast allen Formaten integrierbar, sodass dadurch extrem viele Möglichkeiten entstehen, eine Grafiksoftware, wie z.B. Photoshop, sollte allerdings vorhanden sein.

---

<sup>24</sup> Videobuffer: Im Videobuffer werden Frames gerendert und zwischengespeichert.

**VENTUZ:**

Im Bereich der Content-Erstellung verhalten sich Ventuz und VVVV sehr ähnlich, sodass hier auf den ersten Blick keine gravierenden Unterschiede festzustellen sind. Hinsichtlich meines Testaufbaus war jedoch die Implementierung des Collada-3D-Formats in Ventuz nicht so vielseitig wie in VVVV, da in Ventuz die für die 3D-Animation benötigten Skeleton-Daten nicht unterstützt werden<sup>25</sup>.

**Kapitel V.V. – Interaktivität – Audio-Reactive Content****VVVV:**

In VVVV lassen sich abgespielte Audiodateien mittels einer „FFT“-Node analysieren. Die FFT bietet dabei die wichtigsten Einstellungsmöglichkeiten, wie z.B. die FFT-Size, dadurch ergibt sich eine Vielzahl von Anwendungsmöglichkeiten.

**VENTUZ:**

Auch in Ventuz steht eine FFT-Funktion zur Verfügung, die im Grunde die gleichen Möglichkeiten wie VVVV bietet. Allerdings lässt sich diese Funktion in Ventuz nur mit Audiodateien verknüpfen, die in Windows als Input vorliegen. Ein simples Abspielen und Analysieren einer Datei ist somit nicht ohne weiteres möglich.

**Kapitel V.VI. – Interaktivität – Microsoft Kinect****VVVV:**

Die Implementierung der Microsoft Kinect in VVVV ist recht simpel. Zuerst müssen die für die Treiber, welche für die Kinect benötigt werden, installiert werden. Anschließend lässt sich die Kinect mit Hilfe einer „Kinect“-Node einbinden. Die entsprechenden Nodes stellen dem Nutzer die Daten der Kinect frei zur Verfügung. Die „Skeleton(Kinect)“-Node stellt die Daten des Skeleton-Trackings der Kinect<sup>26</sup> für jeden einzelnen Trackpunkt bereit, der Nutzer muss diese Daten nun entsprechend weiter verarbeiten. Dadurch lässt sich z.B. eine Gestensteuerung oder die von mir beschriebene 3D-Objekt-Animation realisieren.

**VENTUZ:**

Ventuz unterstützt die Microsoft Kinect nicht. Über die Script-Node ist es jedoch denkbar, dass erfahrene C#-Programmierer die Kinect mit Hilfe der Kinect-SDK implementieren könnten. Es werden auch keine ähnlichen Systeme, wie z.B. die Asus Xtion unterstützt, sodass bei einer Anwendung dieser Art eine Drittsoftware zum Einsatz kommen müsste.

---

<sup>25</sup> Vgl: Kapitel V.VI. und Kapitel II.III.II.

<sup>26</sup> Vgl: Abbildung 2.8.

Es gibt für die Kinect z.B. das Tool OSCeolon. Dieses Tool wandelt die Skeleton-Daten in das OSC Protokoll um. OSC ist ein weit verbreitetes Protokoll, welches von vielen Systemen, darunter auch Ventuz und VVVV, unterstützt wird. Allerdings ist auch das OSCeolon bereits etwas veraltet und arbeitet noch mit dem alten OpenNI Treiber.

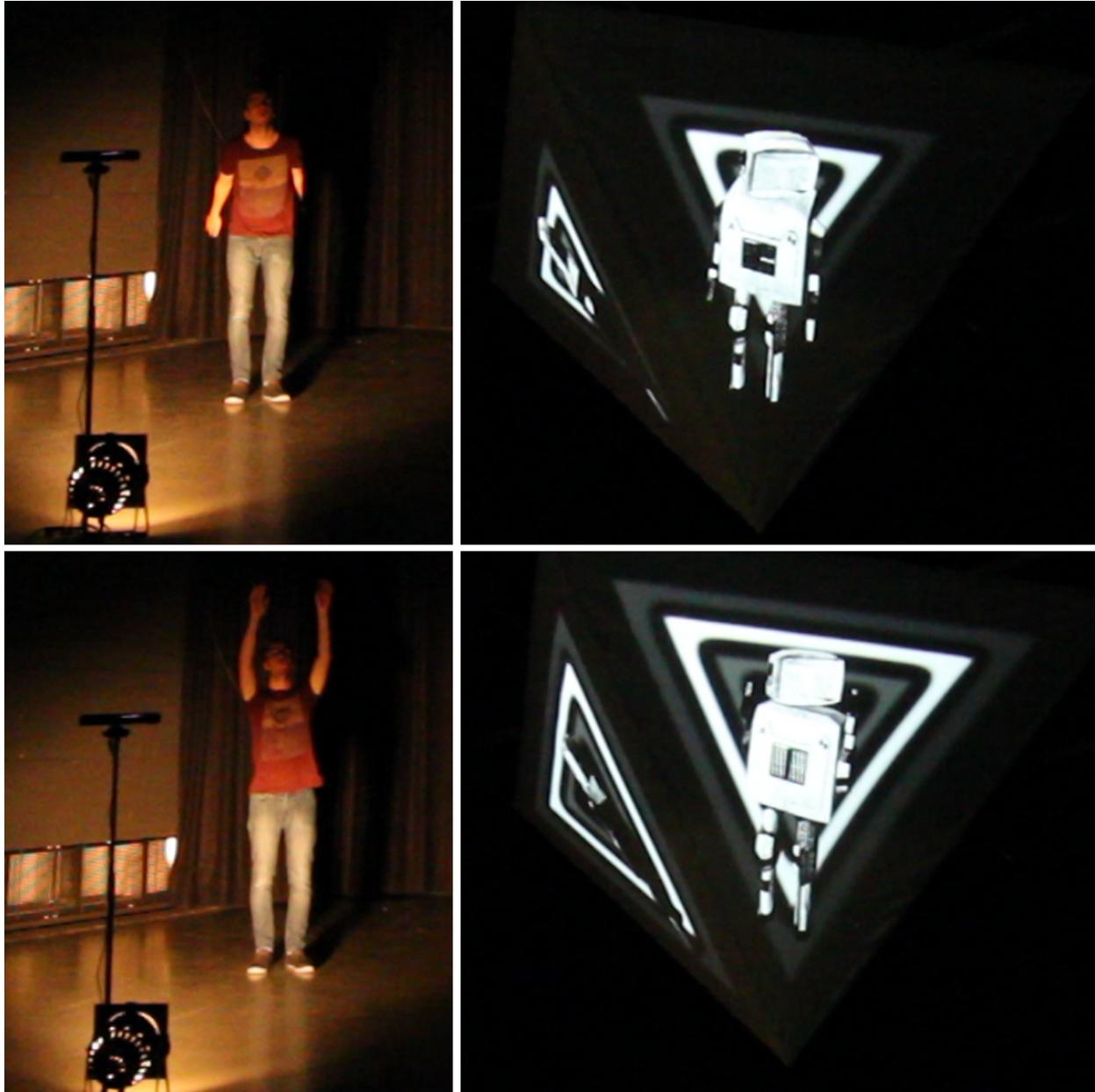


Abbildung 6.3. – 3D-Objekt Animation mit Hilfe der Microsoft Kinect in VVVV.

## Kapitel V.VII. – Interaktivität – 3D-Objekte

### VVVV:

VVVV unterstützt mehrere 3D-Formate, wie das X-Format oder das DAE-Collada Format. Dabei werden alle Daten des DAE-Formats unterstützt und in VVVV importiert, wozu u.a. Geometrie, Skinning-Daten, Animations-, Material- und Texturdaten gehören. Für eine

Character-Animation, wie in meinem Beispiel, sind somit alle Bedingungen erfüllt um eine solche Animation in VVVV zu realisieren, sofern eine entsprechende Collada-Datei vorliegt.

#### **VENTUZ:**

Ventuz unterstützt aktuell fünf 3D-Formate: DAE, 3DS, OBJ, RTG und X. Jedoch gibt es bei dem Import von Collada-Dateien Einschränkungen, sodass Geometrie, Material, Textur, Kamera und Animationsdaten in Ventuz verfügbar sind, jedoch keine Skeleton-Daten. Eine Animation kann in Ventuz also nicht wirklich selbstständig entwickelt werden. Über die Animationsdaten kann allerdings eine vorgefertigte Animation abgespielt werden. Es wäre also denkbar in einer 3D-Software eine Reihe an Animationen zu erstellen und diese dann in Ventuz je nach Eingabe abzuspielen.

### **Kapitel V.VIII. – Interaktivität – TUIO-Protokoll**

#### **VVVV:**

In VVVV kann das TUIO-Protokoll über eine dafür vorgesehene Node aus dem Netzwerk eingelesen und weitergegeben werden. Dabei stehen die Koordination der Touch-Punkte und weitere Daten frei zur Verfügung. In meinem Fall musste ich diese Koordination umrechnen, um sie auf die Pyramide zu übertragen, da die aufgefaltete Pyramide keine rechteckige Form ergibt. Die Umrechnung ist in VVVV kein Problem.

#### **VENTUZ:**

In Ventuz wird das TUIO-Protokoll nicht über eine Node, sondern über den Configuration-Editor implementiert. Dadurch stehen die Koordination nicht direkt im Programm zur Verfügung und können auch nicht umgerechnet werden. Die Touch-Koordinaten des TUIO-Protokolls beziehen sich in Ventuz also immer zwangsläufig auf das gesamte Renderfenster. Um die Daten in Ventuz anzupassen, könnte man eine transparente Ebene mit Touch-Elementen integrieren und diese Daten der Touch-Elemente dann entsprechend weiterverarbeiten. Dies wäre zwar etwas umständlich, führt aber auch zu den erwünschten Effekten. Beide Programme unterstützen neben TUIO auch weitere Protokolle, wie OSC und WindowsTouch.

### **Kapitel V.IX. – Steuerung – MIDI Controller**

#### **VVVV:**

Die Integration eines MIDI-Controllers in VVVV ist sehr simpel und funktioniert auch hier über eine eigene MIDI-Node. An dieser muss der entsprechende Controller und Kanal

definiert werden, dann gibt diese Node die Daten des Controllers am Output aus und die Daten können im Programm z.B. an einen Alpha-Wert geknüpft werden.

### VENTUZ:

In Ventuz muss zur Verwendung eines MIDI-Controllers dieser im Configuration-Editor angelegt werden. Im Ventuz Designer stehen dann eine Reihe von MIDI-Nodes zur Verfügung mit deren Hilfe, genau wie in VVVV, die MIDI-Daten an alle denkbaren Elemente der Scene geknüpft werden können.



Abbildung 6.4. – MIDI Controller: Novation ZeRO SL MkII

In Bezug auf MIDI Controller bieten beide Programme alle wünschenswerten Funktionen an, zudem lassen sich die Controller sehr schnell und intuitiv integrieren. Inwieweit beide Programme alle Controller unterstützen, kann ich jedoch nicht beurteilen, das sollte für jeden Controller einzeln getestet werden.

# Kapitel VII

---

## PERSÖNLICHE STELLUNGNAHME

Ich möchte in diesem Kapitel die Ergebnisse aus den vorhergehenden Kapiteln, sowie meine persönlichen Erfahrungen im Umgang mit Ventuz und VVVV darstellen und erläutern. Aufgrund der Vielfältigkeit und Komplexität der beiden Programme ist es meiner Meinung nach nicht möglich eine allgemein gültige Aussage zu treffen, welche Software die bessere ist. Da sich beide Programme über „Script“-Nodes erweitern lassen, würde ein C#-Programmierer vermutlich zu anderen Ergebnissen kommen als ich. Deshalb werde ich in diesem Kapitel die Vor- und Nachteile, die ich subjektiv empfunden habe, darstellen.

### Kapitel VII.I. – Der erste Eindruck

Den besseren ersten Eindruck macht Ventuz. Die Installation ist gewohnt einfach, wohingegen die Installation von VVVV etwas umständlicher ist. Im Regelfall sollte die Installation von VVVV für jemanden, der sich mit dieser Software auseinandersetzen möchte, kein Problem sein. In Stresssituationen unmittelbar vor einer Show oder Veranstaltung ist die einfache Installation von Ventuz aber ganz klar im Vorteil, da sich hier keine Fehler einschleichen können, besonders wenn für VVVV noch zusätzliche Nodes aus dem Internet heruntergeladen wurden und diese für die Show benötigt werden. Dann kosten das Zusammensuchen und die Einordnung der Ordner wertvolle Zeit, während Ventuz in der Regel keine zusätzlichen Komponenten erfordert.

Auch beim User-Interface macht Ventuz den deutlich besseren Eindruck. Der Nutzer findet sich sofort zurecht und kann mit dem Entwickeln einer Szene beginnen. Der graue Kasten, den VVVV bietet, macht dagegen keinen professionellen Eindruck. Dass das Hauptmenü in VVVV über die mittlere Maustaste erreicht wird, ist ebenfalls etwas ungewohnt.

Nach einer gewissen Einarbeitungszeit, die man am besten mit Hilfe der umfangreichen Tutorials verbringt, findet man sich in beiden Programmen gut zurecht. Das User-Interface spielt in der reinen Programmentwicklung nur eine untergeordnete Rolle, da die Nodes in der Regel über einen Tastatur-Shortcut gesucht und eingefügt werden.

Allerdings bietet meiner Meinung nach das User-Interface von Ventuz einen beträchtlichen Vorteil, denn durch die Aufteilung von Content und Hierarchy-Fenster wird das entwickelte Programm deutlich übersichtlicher. In VVVV sind zu jeder Zeit alle Elemente sichtbar und das Programm erscheint immer größer, außerdem verlaufen auch die Verknüpfungen der Nodes quer durch den Bildschirm. Im Gegensatz dazu ist Ventuz durch die Aufteilung in zwei Fenster deutlich übersichtlicher. Im Content-Fenster werden nur die im Hierarchy-Fenster angeklickten Elemente dargestellt und die verknüpften Nodes im Content-Fenster können via Rechtsklick Ein- bzw. Ausgeblendet werden.

Ventuz überzeugt zudem auch in der Programmstruktur, denn auch durch die Erweiterung des Configuration Editors bleibt die Entwicklungsumgebung der Szene übersichtlich, da die Einstellungen die hier vorgenommen werden in VVVV in den Patch mit eingearbeitet werden müssen. Durch die Erstellung von Sub-Patches lässt sich zwar auch VVVV übersichtlich gestalten, das ist allerdings aufwendiger und kostet Zeit.

Hier sehe ich ein klares Plus bei Ventuz. Gerade wenn mehrere Personen an einer Szene/Patch arbeiten, findet man sich in Ventuz schneller in der Arbeit einer anderen Person zurecht. Auch in der Fehlersuche hilft diese Funktion, da man in Ventuz schneller Zugriff auf Programmteile hat und diese einfach aktivieren, bzw. deaktivieren kann.

### **Kapitel VII.II. – Systemoffenheit und ein unerklärlicher Fehler**

VVVV ist im Gegensatz zu Ventuz offener gestaltet. Über die Website der Software können sich Entwickler die VVVV-SDK herunterladen und eigene Nodes dem Addon-Pack von VVVV hinzufügen. Dies bietet sowohl Vor- als auch Nachteile, denn natürlich werden durch die Bereitstellung der SDK spezielle Funktionen zu der Software hinzugefügt, welche es sonst nicht geben würde. Im Gegenzug verliert die Software dadurch vermutlich an Stabilität. Während bei Ventuz alle Programmelemente gemeinsam entwickelt und vor der Veröffentlichung ausgiebig getestet werden, ist dies bei Ventuz nicht der Fall.

In meinem Testsetup hat mir nur VVVV die Möglichkeit geboten eine Skeleton-Animation einer 3D-Figur mit Hilfe der Kinect durchzuführen. Ventuz hat weder die Skinningdaten der Collada-Datei, noch die Implementierung der Kinect ermöglicht, sodass hier VVVV ganz klar im Vorteil ist. Allerdings hatte ich dabei auch Programm-Fehler, welche ich mir nicht erklären kann:

Beim Starten meines Patches wurden mir zwei Ebenen teilweise nicht angezeigt, obwohl diese mit einer Gruppe verbunden waren, die vom Renderer dargestellt wurden. Logisch betrachtet müssten die Layer dargestellt werden. Durch Löschen und Erneuern dieser Verbindung wurde dieser Fehler behoben, obwohl an der Logik nichts verändert wurde. Da die Verbindung entweder besteht oder eben nicht, die Reihenfolge, in der Verknüpfungen erstellt werden, aber keine Rolle spielen sollten, kann ich diesen Fehler bisher nicht nachvollziehen.

Einen derartigen Fehler habe ich in über einem Jahr des intensiven Umgangs mit Ventuz bisher nicht erlebt, sodass meiner Einschätzung nach, Ventuz das zuverlässigere System ist.

### **Kapitel VII.III. – Ventuz oder VVVV**

Für mein persönliches Testsetup habe ich VVVV benutzt, nur so konnte ich die Kinect-Animation durchführen. Die optimale Lösung, wenn man die Kosten nicht mit einbezieht, wäre aber eine Kombination aus Ventuz und VVVV. Dazu würde ich die Kinect-Animation mit Hilfe von VVVV realisieren und diese Animation über eine Input-Karte in Ventuz integrieren und über Ventuz die Projektoren bespielen. Ventuz bietet dazu einfach die höhere Zuverlässigkeit und bessere Möglichkeiten in der Anpassung des Mappings. Alle Funktionen die dabei in Ventuz möglich sind, würde ich auch mit Ventuz durchführen, zum einen aufgrund der Zuverlässigkeit, zum anderen wegen der besseren Übersicht.

Da Ventuz aber nicht alle Funktionen bietet, die VVVV bietet, bleibt immer abzuwägen, welche Software zum Einsatz kommen soll.

Generell sind jedoch beide Systeme sehr empfehlenswert und bieten aufgrund der node-basierten Programmierung sehr vielseitige Einsatzmöglichkeiten. Dabei unterscheiden sie sich grundlegend von herkömmlichen Medienservern, wie Resolume Arena 5, oder Game-Engines, da der Nutzer jede Funktion selber logisch entwickeln muss. Die Programmierung einer Szene bzw. eines Patches ist aber entsprechend komplex, sodass der Nutzer sich intensiver mit der Software auseinandersetzen muss, um sie zu beherrschen.

### **Kapitel VII.IV. – Schlussfolgerung**

Wie eingangs erwähnt, kann ich nicht sagen, dass eine Software besser ist als die andere. Beide Systeme eignen sich in jedem Fall sehr gut, um eine interaktive Mehrfachprojektion zu realisieren, obwohl sie auch entsprechende Unterschiede aufweisen.

VVVV bietet durch die offene Systemstruktur fast alle denkbaren Funktionen zu einem extrem günstigen Preis von nur 500,00€. Im Gegensatz dazu bietet Ventuz einen geringeren

Funktionsumfang für einen höheren Preis von 6950,00€. Die Frage, welche Software zum Einsatz kommen soll, hängt meiner Meinung nach davon ab welches, Vorhaben konkret besteht.

VVVV bietet für kleinere Projekte mit einer einfachen Programmierung sicherlich eine solide Grundlage zu einem guten Preis. Je komplexer und aufwändiger die Programmierung wird, desto mehr Vorteile bietet Ventuz; Es bleibt übersichtlich, bietet mit dem Configuration Editor viele Optionen das Mapping oder die Hardwarekonfigurationen anzupassen und ist vor allem absolut zuverlässig. Bei speziellen Anforderungen, wie z.B. der Integration der Kinect, bietet Ventuz jedoch nicht immer eine passende Lösung an, sodass dabei ggf. wieder VVVV im Vorteil ist.

In einem persönlichen Gespräch mit Ralf Stanke<sup>28</sup> habe ich erfahren, dass Ventuz bewusst nicht jede Funktion integriert. Wie bereits an dem Preis zu erkennen ist, spricht Ventuz nur professionelle Anwender an. Diese Anwender erwarten stets solide und zuverlässige Software, eine Integration von einer Kinect könnte dabei als unprofessionell empfunden werden, da die Kinect im Grunde für eine Spielkonsole entwickelt wurde.

---

<sup>28</sup> Vorstand und Entwickler bei Ventuz

# Anhang I

---

## Abbildungsverzeichnis

1.1. - Interaktive Mehrfachprojektion bei der Vorstellung des Porsche 911 .....	S. 4
<a href="https://www.youtube.com/watch?v=s39Q9Icbk3s">https://www.youtube.com/watch?v=s39Q9Icbk3s</a> (Screenshot) (04.01.2016 14:23 Uhr)	
2.1. - Tracking der Hände einer Person .....	S.
<a href="https://youtu.be/7gwSlNitdz0">https://youtu.be/7gwSlNitdz0</a> (Screenshot) (23.10.2015 11:01 Uhr)	
2.2. - Tastatur-Projektion .....	S.
<a href="http://thecoolgadgets.com/ufo-concept-unidentified-flinching-object-display-manufacturers-be-warned/">http://thecoolgadgets.com/ufo-concept-unidentified-flinching-object-display-manufacturers-be-warned/</a> (23.10.2015 11:41 Uhr)	
2.3. - Skizze einer Softedgeprojektion .....	S.
<a href="http://www.coolux.de/root/images/tutorials/softedge/overlap.gif">http://www.coolux.de/root/images/tutorials/softedge/overlap.gif</a> (23.10.2015 12:08 Uhr)	
2.4./2.5. - Fassadenprojektion .....	S.
<a href="https://www.youtube.com/watch?v=XSR0Xady02o">https://www.youtube.com/watch?v=XSR0Xady02o</a> (Screenshot) (23.10.2015 12:43 Uhr)	
2.6. - „Lightning the Sails“, Projection Mapping .....	S.
<a href="http://www.urbanscreen.com/lightning-the-sails/">http://www.urbanscreen.com/lightning-the-sails/</a> (23.10.2015 - 13:14 Uhr)	
2.7. - Treiberkommunikation der Microsoft Kinect.....	S.
UserGuide Open NI	
2.8. - Trackpunkte der Microsoft Kinect .....	S.
<a href="http://vrvv.org/documentation/kinect">http://vrvv.org/documentation/kinect</a> (23.10.2015 - 13:25 Uhr)	
2.9. - Leuze Rod4 Laser .....	S.
ROD4plus User Manual	
2.10. - Automap User Interface.....	S.
Screenshot, Jakob Dohrmann	
2.11. - Skizze einer Trapezkorrektur.....	S.
<a href="http://diy-community.de/attachment.php?attachmentid=98108">http://diy-community.de/attachment.php?attachmentid=98108</a> (07.11.2015 – 18:46 Uhr)	
2.12. - Projektionsmapping in VVVV .....	S.
Screenshot, Jakob Dohrmann	
3.1. - Resolume Arena User-Interface .....	S. 23
Screenshot, Jakob Dohrmann	
3.2. - Resolume Arena Output Settings .....	S. 24
Screenshot, Jakob Dohrmann	
3.3. - Resolume Arena Output Vorverzerrung.....	S. 25
Screenshot, Jakob Dohrmann	

3.4. - Unity3D User-Interface .....	S. 26
<a href="http://unity3d.com/learn/tutorials/modules/beginner/editor/interface-overview?playlist=17090">http://unity3d.com/learn/tutorials/modules/beginner/editor/interface-overview?playlist=17090</a> (Screenshot) (31.11.2015 – 19:23 Uhr)	
4.1. - Beispiel VVVV-Programmierung, Mask-Patch .....	S. 29
Screenshot, Jakob Dohrmann	
4.2. - Ventuz User-Interface.....	S. 31
Screenshot, Jakob Dohrmann	
4.3. - VVVV User-Interface.....	S. 32
Screenshot, Jakob Dohrmann	
5.1. - Aufbau Pyramide und Projektor .....	S. 35
Foto: Jakob Dohrmann, Selbstausröser	
5.2. - Microsoft Kinect.....	S. 36
<a href="https://msdn.microsoft.com/en-us/library/hh438998.aspx">https://msdn.microsoft.com/en-us/library/hh438998.aspx</a> (11.01.2016 - 18:15 Uhr)	
6.1. - VVVV, BadMapper – Projektions-Mapping.....	S. 39
<a href="https://vovv.org/contribution/badmapper">https://vovv.org/contribution/badmapper</a> (11.01.2016 - 17:24 Uhr)	
6.2. - Ventuz Render Setup .....	S. 39
<a href="http://www.ventuz.com/support/help/V4_06/WarpingSoftEdgingMasking.html#TheUserInterface">http://www.ventuz.com/support/help/V4_06/WarpingSoftEdgingMasking.html#TheUserInterface</a> (11.01.2016 - 17:25 Uhr)	
6.3. - 3D-Animation, Microsoft Kinect .....	S. 42
Screenshot Videoaufnahme, Jakob Dohrmann	
6.4. - MIDI Controller - Novation ZeRO SL MkII.....	S. 44
<a href="http://www.upbeat.ch/shop/de/controller/novation-remote-zero-sl-mkii.html">http://www.upbeat.ch/shop/de/controller/novation-remote-zero-sl-mkii.html</a> (11.01.2016 - 17:48 Uhr)	

# Anhang II

---

## Quellenverzeichnis

<http://www.i-programmer.info/news/194-kinect/7004-openni-to-close-.html>  
(21.10.2015 - 14:29 Uhr)

[https://github.com/OpenNI/OpenNI/blob/master/Documentation/OpenNI\\_UserGuide.pdf](https://github.com/OpenNI/OpenNI/blob/master/Documentation/OpenNI_UserGuide.pdf)  
(21.10.2015 - 14:34 Uhr)

<https://www.microsoft.com/en-us/kinectforwindows/meetkinect/features.aspx>  
(21.10.2015 - 14:55 Uhr)

<https://msdn.microsoft.com/en-us/library/hh855348.aspx>  
(21.10.2015 - 14:57 Uhr)

<http://support.xbox.com/en-US/xbox-on-other-devices/kinect-for-windows/kinect-for-windows-v2-setup>  
(21.10.2015 - 17:00 Uhr)

<https://support.xbox.com/de-DE/xbox-360/kinect/kinect-sensor-components>  
(21.10.2015 - 17:03 Uhr)

<http://www.xida.de/2012/01/31/wie-funktioniert-kinect/>  
(21.10.2015 - 18:13 Uhr)

[http://www.ventuz.com/support/help/v3\\_01/GeometryImport.html](http://www.ventuz.com/support/help/v3_01/GeometryImport.html)  
(21.10.2015 - 18:33 Uhr)

<http://www.midi.org/aboutmidi/index.php>  
(22.10.2015 - 14:28 Uhr)

<http://novationmusic.de/software/automap#>  
(22.10.2015 - 14:33 Uhr)

<http://www.gizmodo.com.au/2013/05/lighting-the-sails-behind-the-scenes-on-vivid-sydneys-most-ambitious-project/>  
(23.10.2015 - 16:01 Uhr)

<http://www.lang-ag.com/de/produkte/touch-solutions/radartouch/products/Product/show/rtouch-500me-5.html>  
(04.11.2015 - 17:30 Uhr)

<http://www.project-syntropy.de/syntouch-radar/>  
(04.11.2015 - 17:33 Uhr)

[http://www.gamedev.net/page/resources/\\_/technical/game-programming/working-with-the-directx-x-file-format-and-animation-in-directx-90-r2079](http://www.gamedev.net/page/resources/_/technical/game-programming/working-with-the-directx-x-file-format-and-animation-in-directx-90-r2079)  
(10.11.2015 - 15:08 Uhr)

<https://docs.unrealengine.com/latest/INT/Engine/Content/FBX/SkeletalMeshes/index.html>  
(10.11.2015 - 15:15 Uhr)

<http://resolume.com/software>  
(12.12.2015 - 13:08 Uhr)

<http://resolume.com/download/>  
(10.12.2015 - 15:08 Uhr)

<https://vvvv.org>  
(10.12.2015 - 15:44 Uhr)

[www.ventuz.com](http://www.ventuz.com)  
(10.12.2015 - 15:48 Uhr)

[https://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/what\\_is\\_directx.mspx?mfr=true](https://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/what_is_directx.mspx?mfr=true)  
(20.10.2015 – 14:23 Uhr)

<http://www.nvidia.de/object/nvidia-mosaic-technology-de.html>  
(27.12.2015 - 12:01)

<http://www.nvidia.de/object/nvidia-nview-desktop-management-software-de.html>  
(27.12.2015 - 12:02)

<https://vvvv.org/documentation/collada>  
(04.01.2016 - 16:56 Uhr)

<https://vvvv.org/documentation/vvvv-sdk>  
(09.01.2016 - 13:01 Uhr)

<http://www.ventuz.com/impressum>  
(09.01.2016 - 13:24 Uhr)

<https://vvvv.org/blog/vvvv-in-numbers-2014>  
(09.01.2016 - 13:26 Uhr)

<https://vvvv.org/documentation/dx11-overview>  
(10.01.2016 - 10:11 Uhr)

<https://vvvv.org/forum/3d-object-import>  
(10.01.2016 - 10:11 Uhr)

<http://www.incas-training.de/projekte/software-kaufen/ventuz>  
(20.01.2016 – 09:07 Uhr)

[http://www.leuze-electronic.de/selector/ci\\_img/Heiler/Volumes/Volume0/opasdata/d100001/medias/docus/15/\\$v4/TD\\_ROD4plus\\_de\\_50106879.pdf](http://www.leuze-electronic.de/selector/ci_img/Heiler/Volumes/Volume0/opasdata/d100001/medias/docus/15/$v4/TD_ROD4plus_de_50106879.pdf)  
(20.01.2016 – 09:12 Uhr)

# Anhang III

---

## **Eigenständigkeitserklärung**

Hiermit versichere ich, dass ich die vorliegende Bachelor-Thesis mit dem Titel:

### ***Analyse und Vergleich von Softwaresystemen für interaktive Mehrfachprojektion***

selbständig und nur mit den angegebenen Hilfsmitteln verfasst habe. Alle Passagen, die ich wörtlich aus der Literatur oder aus anderen Quellen wie z. B. Internetseiten übernommen habe, habe ich deutlich als Zitat mit Angabe der Quelle kenntlich gemacht.

(Unterschrift)