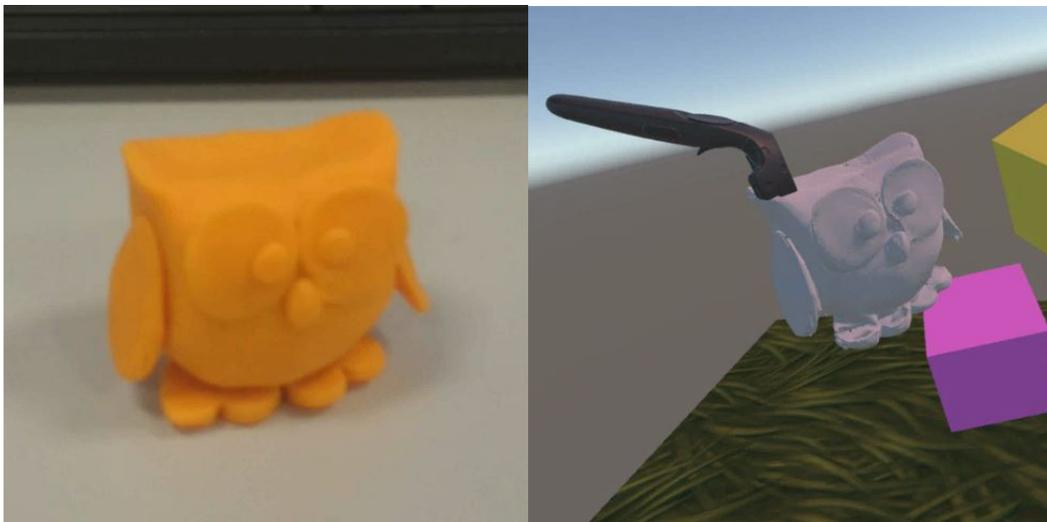


HAW Hamburg
Wintersemester 2017/2018

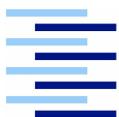
Einsteiger-Leitfaden zum Thema
Einbindung von 3D-Scans in
Virtual Reality



Tobias Latta

22. Januar 2018(v1)

Betreuer: Prof. Thomas Lehmann



**Hochschule für Angewandte
Wissenschaften Hamburg**
Hamburg University of Applied Sciences



**/* CREATIVE
SPACE FOR TECHNICAL
INNOVATIONS */**

Inhaltsverzeichnis

1 Motivation	2
1.1 Zielgruppe	2
1.2 Relevanz von VR	2
1.3 Gründe für Darstellung in VR	3
1.4 Inhalt des Leitfadens	3
1.5 Allgemeine Hinweise	6
2 Anforderungen an das reale Objekt	7
3 Reales Objekt digitalisieren	9
3.1 Benutzte Hardware - 3D Scanner "DAVID SLS-3"	9
3.2 Anleitung zur Digitalisierung mit DAVID SLS-3 Scanner [Video 1]	10
3.3 LEVEL-UNLOCK	14
4 Digitales Objekt anpassen für VR	14
4.1 Verwendete Programme	14
4.2 Anpassen der Objektgröße mit Programm "Instant Meshes" [Video 2]	16
4.3 Anpassen der Objektausrichtung mit Programm "Maya" [Video 3]	16
4.4 Abspeichern des Objektes [Video 4]	19
4.5 LEVEL-UNLOCK	19
5 Übertragung auf die VR-Brille	20
5.1 Programmierung mit Software "Unity"	20
5.2 Öffnen der Beispielszene in "Unity" [Video 5]	21
5.3 Erklärung der Benutzeroberfläche von "Unity" [Video 6]	23
5.4 Einfügen des eigenen Objekts in "Unity" [Video 7]	23
5.5 Anhängen der physikalischen Eigenschaften in "Unity" [Video 8]	24
5.6 Eintritt in den Virtuellen Raum mit Programm "Steam" [Video 9]	26

1 Motivation

1.1 Zielgruppe

Moin! Eigentlich hast du an dieser Stelle schon gewonnen. Du bist an diesen Leitfaden gekommen, da du dich anscheinend für Virtual Reality (kurz "VR") und 3D-Scans interessierst. Das ist ziemlich gut und besser zu lernen als du denkst. Ersteinmal klären wir aber nochmal ab, ob dieser Leitfaden wirklich das Richtige für dich ist:

- Du hast richtig Bock einen Einblick in die Virtual Reality zu bekommen
- Du weißt eigentlich noch gar nicht so viel über die Virtual Reality, möchtest das aber nachholen
- Du hast Zugang zu dem /* Creative Space of Technical Innovations */ am Steindamm 94 und kannst dort das Equipment benutzen
- Du hast keine Angst, etwas "kaputt zu machen" wenn du mit dem Equipment rumspielst. Rumspielen ist wichtig. Dabei lernst du viel und solltest dich nicht von der Furcht vor Technik zurückhalten lassen.
- Computer und Internet sind für dich nicht "Neuland" sondern Nachbarschaft
- Optional: Dieser Leitfaden wird dir noch leichter fallen, wenn du schonmal ein CAD-Tool verwendet hast (Beispielsweise Zeichenprogramme wie AutoCAD, 3D-Modellierungstools wie Blender oder aber Software aus der Creative Suite von Adobe)

1.2 Relevanz von VR

Warum wollen wir uns hier eigentlich mit VR auseinandersetzen? Nur weil ein paar Nerds immer häufiger mit diesen klobigen Gebilden auf dem Kopf rumlaufen und verzweifelt in der Luft herumfuchtelten sollten wir uns jetzt auch damit auseinandersetzen?

Ganz klare Antwort: **JA**. VR zieht durch Playstation VR, die Oculus Rift oder aber die HTC Vive in die privaten Haushalte ein und ermöglicht eine neue Gaming-Erfahrung. Aber ist Gaming tatsächlich der einzige Nutzen von VR?

Es ist wie mit dem Internet um die Jahrtausendwende. Man hat eine Technologie, weiß jedoch noch gar nicht so recht was man damit anfangen kann. Ersteinmal werden alte Probleme auf neue Weise versucht zu lösen. Man löste den analogen Briefverkehr durch schnelle e-Mail-Kommunikation ab oder digitalisierte Textverarbeitung durch Word und LaTeX. Interessant wird es dann, wenn ungestellte Fragen gelöst werden: Wer hat sich vor 2007 die Frage gestellt, wie er möglichst unkompliziert und auf einer einzigen Homepage mit allen seinen Freunde (international, national, ganz egal) in Kontakt stehen kann? Facebook hat auf diese ungestellte Frage ein Antwort geliefert, genauso wie es Google mit einer Online-Suchmaschine oder eBay mit einer Online-Auktionshalle getan hat.

Lange Rede, kurzer Sinn: Die Möglichkeiten von Virtual Reality sind noch lange nicht ausgeschöpft und es kommt jetzt auf die Menschen an, die aus anderen Bereichen diese Technik nutzen und Produkte kreieren, auf die ein eingefahrener VR-Entwickler nicht kommen würde. In welchem Bereich du also tätig bist, spielt keine Rolle. Du wirst eine Möglichkeit finden, diese Technik sinnvoll und erstaunlich einzusetzen.

1.3 Gründe für Darstellung in VR

Was habe ich davon, einen realen Gegenstand in die VR zu schicken?

Durch die Digitalisierung von Gegenständen kann man interessante Veränderungen an dem Gegenstand durchführen. Beispielsweise ist es durch das Verändern eines einzigen Wertes möglich, einen Alltagsgegenstand um Faktor 10 zu vergrößern. Auch können große Gegenstände verkleinert werden. Dazu kommt, dass jeder Gegenstand in VR für den User schwerelos ist. Ein Verdrehen oder Verschieben ist mit dem Controller möglich, was wiederum weitere Möglichkeiten bietet. Der Phantasie sind keine Grenzen gesetzt. Von Autoweitwurf, bis Schwimmen in der Kaffeetasse oder Laufen an der Decke ist in VR alles möglich. Im unternehmerischen Kontext wird die Technik genutzt wenn beispielsweise Designer ihre neuen Möbel in einem virtuellen Showroom stets verändern können und in unterschiedlichen Szenarien präsentieren. Schulungen in VR werden ebenfalls durchgeführt.

Digitale Objekte mit realen Vorbildern haben einen hohen Wiedererkennungswert. Der Einstieg in die VR wird erleichtert, wenn bekannte Objekte vorhanden sind, an welchen man sich orientieren kann. Der Nutzer ist dann nicht so schnell überfordert und kann Stück für Stück Veränderungen an dem Raum erfahren und vornehmen.

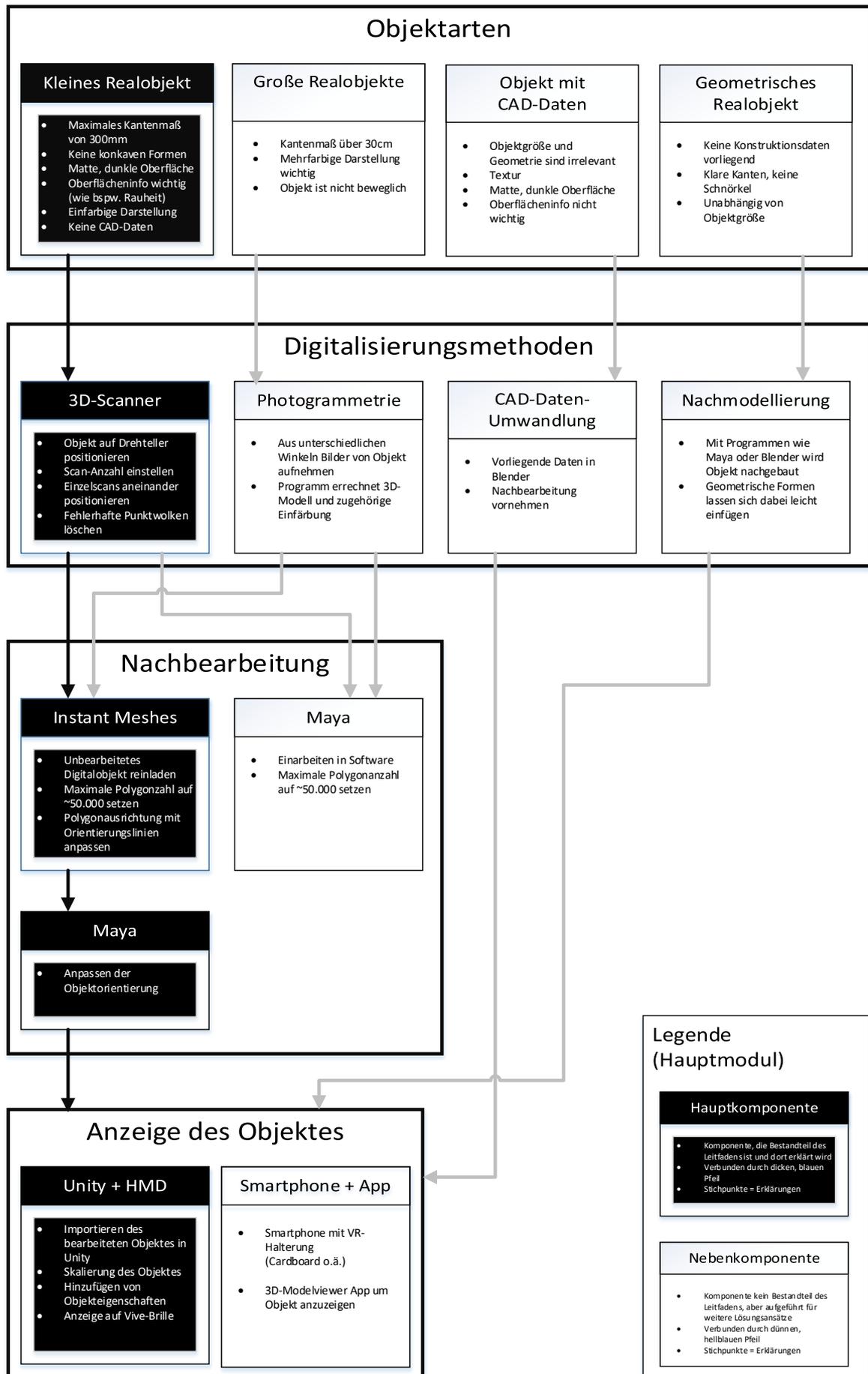
1.4 Inhalt des Leitfadens

Was ist alles Gegenstand dieses Leitfadens? Wie weit tauchen wir in die Materie ein?

"Soviel wie nötig, so wenig wie möglich" wissenschaftliche Korrektheit ist in diesem Leitfaden vertreten. Um nicht von den Fachbegriffen abgeschreckt zu werden und erstmal die technische Dokumentation raussuchen zu müssen, sind die wichtigsten Funktionen erklärt. So kann möglichst schnell das Ziel erreicht und doch der ein oder andere technische Begriff aufgeschnappt werden. Das macht sich dann bezahlt, wenn du Blut geleckt hast und dich weiter mit dem Thema auseinandersetzt.

Wir behandeln die wichtigsten Schritte in der Kette von realem zu digitalen Objekt. Auf der nächsten Seite ist ein Diagramm zu sehen, welches dir diesen Vorgang veranschaulicht. Die übergeordneten Kästen bilden dabei die Hauptschritte des Prozesses. Nachdem wir unser Objekt einer Kategorie zugeordnet haben (Objektarten) werden wir es 3D-scannen (Digitalisierungsmethoden) und anschließend mit Instant Meshes und Maya aufbereiten (Nachbearbeitung). Zum Schluss findet die Anzeige mit Unity + HTC Vive statt (Anzeige des Objektes). Manche Begriffe sind dir sicherlich unbekannt, eine erste Aufklärung zu benutzter Hard- und Software findet auf der übernächsten Seite statt.

1 Motivation



1 Motivation

Im nachfolgenden Teil ist die genutzte Hard- und Software (abgesehen von Computer + Windows etc.) kurz erklärt. Die ausführliche Erklärung und Anleitung findet in dem jeweiligen Kapitel statt:

- 3D-SLS-Scanner von Firma DAVID
 - Dabei handelt es sich um ein Projektor-Kamera-System mit Drehteller, welches Lichtmuster auf das Objekt legt und dann durch geschossene Bilder berechnet, welche Maße das Objekt an der Stelle hat. Um ein Objekt komplett zu digitalisieren, müssen Fotos (Scans) aus jeder Ansicht gemacht werden (dafür ist der Drehteller da). Der Scanvorgang findet, einmal eingestellt, automatisch statt. Die beigelegte DAVID-Software hilft dann ebenfalls bei der Nachbearbeitung von aufgenommenen 3D-Scans.
- Nachbearbeitungsprogramme "Instant Meshes" und "Maya"
 - Unsere Scans benötigen etwas Nachbearbeitung, da ein kompletter 3D-Scan erstmal zu hochauflösend ist für die gewünschte Größe in VR:
Das digitale Objekt sollte im Bereich von etwa 100.000 (100k) Bildpunkten bleiben, damit die Rechenleistung des Computers ausreicht und eine flüssige Bewegung in der Virtual Reality möglich ist. Wenn Bildpunkte verbunden werden entstehen Polygone, auch kurz "Polys". Mit Instant Meshes kann die Roh-Datei, welche in diesem Beispiel etwa 1 Million Polys hat, erst einmal grob reduziert werden. Dabei können Geometrien eingetragen werden, an denen sich die Polygon-Reduktion orientiert. Vorteile bringt das hinsichtlich einer symmetrischen Poly-Struktur. Die ist wiederum für spätere Texturierung (Einfärbung) notwendig.
Mit dem professionellen 3D-Modellierungsprogramm Maya kann im Nachhinein vieles nachkorrigiert werden. Unter anderem könnte aus unserem Scan ein ganz anderes Objekt "wie aus Knete" modelliert werden. Wir beschränken uns jedoch auf die Korrekturfunktionen im Sinne einer richtigen Skalierung, Drehung und Färbung unseres Objektes. Optional können Polygone weiter bearbeitet werden.
- Integration in Unity und Abspielen auf HTC Vive
 - Hier kommt der Showdown: Unser nachbearbeitetes Objekt wird in die Entwicklungsumgebung Unity für Virtual-Reality-Headsets (in unserem Fall HTC Vive) importiert. Diese Entwicklungsumgebung ist ein Programm, welches ermöglicht, bestimmte Inhalte und Programme auf der VR-Brille anzuzeigen. Eine Verbindung zwischen VR-Headset und Computer ist schon eingerichtet, genauso wie die Definition des begehbaren Bereichs im CSTI-lab. Wir können also direkt mit dem fun-part anfangen! hooray. Mit diesem Leitfaden kommt außerdem ein Beispielprogramm für Unity, in welches unser Objekt einfach eingefügt werden kann. Da dran wird anschließend weiter erklärt, wie diese Entwicklungsumgebung funktioniert.

1.5 Allgemeine Hinweise

Hier sind noch ein paar Hinweise, die ihr beim Durch-Rocken des Leitfadens beachten solltet:

- Wenn bei Einstellungsmöglichkeiten nicht explizit ein ✓ angegeben ist, soll die Option un- ausgewählt sein
- Nur besprochene Einstellungsfenster müssen überprüft werden, weitere Einstellungsfenster sollten unverändert bleiben
- Sicherlich sind dir im Inhaltsverzeichnis schon die blauen [Video ...] Markierungen aufge- fallen. Du kannst sie anklicken um den entsprechenden YouTube-Clip aufzurufen, welcher den jeweiligen Abschnitt nochmal nachspielt.
Die HyperLinks sind auch in den Überschriften im laufenden Leitfaden zu finden, also brauchst du nicht immer zurück zum Inhaltsverzeichnis wechseln
- Wenn du den Leitfaden aus dem [SHARE-Ordner](#) des CSTI hast, kannst du auch den [LF-Videos-Ordner](#) in dem Oberverzeichnis [3D-Scanner-VR-Beginner-Guide](#) öffnen um dir dort die Videos lokal anzuschauen oder zu kopieren
- Abgesehen von den Überschriften, handelt es sich bei [blau markierten Texten](#) um Para- meter, welche in der entsprechenden Software so gefunden und eingestellt werden können. Anders gesagt, ist ein Stichpunkt in blau gefasst, sollte er in dem benutzten Programm auf den angegebenen Wert konfiguriert werden

Stichpunkte, die statt einem Spiegelstrich eine rote Box wie hier haben, können markiert werden. Dadurch kannst du stets erledigte Schritte abhaken und sogar deinen Fortschritt speichern, falls eine Kaffeepause ansteht oder etwas anderes unvorhergesehenes passiert (Stichwort :Zombie Apokalypse)

2 Anforderungen an das reale Objekt

Was kann denn eigentlich alles eingescannt werden? Ist mein Auto zu groß oder meine Katze zu lebendig um digitalisiert zu werden? Diese Kriterien sollte euer Objekt erfüllen um mit dem 3D-Scanner digitalisiert werden zu können.

- Das maximale Kantenmaß sollte 300 mm nicht überschreiten
- Wenig Hohlräume beim Objekt sind wichtig, da der Scanner diesen Bereich logischerweise "nicht einsehen" kann also somit nicht digitalisieren kann
- Die Oberfläche darf nicht spiegelnd oder glänzend sondern im Idealfall matt oder rau damit. Das liegt ebenfalls an dem Scanner, der bei spiegelnden Objekten die ausgesendeten Lichtmuster zur Objektabtastung nicht richtig zurückrechnen kann.
- Eine einfarbige Darstellung sollte für euch erstmal ok sein, da wir vorerst keine Textur übernehmen
- Es liegen keine CAD-Daten zu dem Objekt vor (CAD-Daten entstehen, wenn das Objekt am Computer mit AutoCAD oder Ähnlichem konstruiert worden wäre)

Und was kann unternommen werden, wenn das Objekt nicht den Kriterien entspricht?

Es gibt weitere Digitalisierungsmöglichkeiten, welche du dir jedoch selbst anschauen darfst. Setze dich bei Interesse selbstständig mit der 3D-Objekt Thematik auseinander und komme mit deinem ".obj-File" wieder hier an. Dann kann die Reise für dich in Kapitel 4 "Digitales Objekt anpassen für VR" weitergehen. Für weitere Recherchen seien dir hier aber noch ein paar andere Objektklassen mit an die Hand gegeben:

- Objekte, welche sehr viel größere($< 2x$) Abmaße haben und nicht per CAD konstruiert sind. Die Textur oder Oberfläche spielt eine Rolle.
 - Konkret ist das Kantenmaß > 300 mm
 - Es gibt eine mehrfarbige Textur, die essentiell für die digitale Repräsentation ist
 - Das Objekt ist eventuell nicht beweglich, muss also an Ort und Stelle digitalisiert werden
 - Wie kann hierbei ein 3D-Objekt erstellt werden?
 - * Über Photogrammetrie. Von dem Objekt werden mehrere Bilder aus verschiedenen Winkeln geschossen.
 - * Ein Programm errechnet das 3D-Modell mit Textur aus den Bilder-Daten
 - * Das Verfahren ist unabhängig von der Objektgröße
 - * Je mehr Bilder, desto detailgetreuer kann das 3D-Modell erstellt werden
 - * Programme sind häufig kostenpflichtig oder treten Rechte des 3D-Objekts beim Scanvorgang an den Programmhersteller ab. Beispielsweise "123D Catch" von Autodesk oder "PhotoScan" von AgiSoft.

2 Anforderungen an das reale Objekt

- Objekte, welche per CAD konstruiert wurden
 - Die Konstruktionsdaten liegen vor
 - Die Objektgröße und Geometrie sind dabei irrelevant
 - Textur ist für die digitale Repräsentation nicht wichtig
 - Wie kann hierbei ein 3D-Objekt erstellt werden?
 - * Es kann eine Umwandlung der digitalen CAD-Daten in ein digitales 3D-Objekt erfolgen
 - * Umwandlung erfolgt mit 3D-Modellierungsprogrammen wie beispielsweise Blender. Dort werden dann die CAD-Dateien importiert, können direkt angepasst werden (siehe Kapitel 4) und dann als .obj-File (das ist das benötigte Format) ausgegeben werden.
 - * Die Qualität des erzeugten 3D-Objekts ist jedoch nicht unbedingt gut. Es muss unter Umständen eine Nachbearbeitung wie in Kapitel 4 vorgenommen werden
- Objekte, welche geometrisch leicht zu bestimmen sind (also geometrischen Grundformen wie Zylinder, Kegel, Kugel ähnlich sind)
 - Keine Konstruktionsdaten vorliegend oder nötig
 - Klare Kanten und keine großen "Schnörkel"
 - Leicht nach zu zeichnen
 - Unabhängig von der Objektgröße
 - Wie kann hierbei ein 3D-Objekt erstellt werden?
 - * Das reale Objekt wird einfach mit einem 3D-Modellierungsprogramm am Computer nachgebaut
 - * Objektgeometrie sollte nicht zu ausgefallen sein um den Modellierungsaufwand im Rahmen zu halten und nicht zu viel Zeit zu verwenden
 - * Textuierung ist nicht direkt vorliegend, kann im Nachhinein aber sehr gut nachgetragen werden
 - * Auch kann eine Nachbearbeitung ausgelassen werden, da beim Nachmodellieren schon auf die wichtigsten Punkte Rücksicht genommen werden kann

Sofern du gerade ein Objekt bei dir hast, welches den Kriterien für einen Scan mit 3D-Scanner genügt, kannst du es gerne für den nachfolgenden 3D-Scan verwenden. Ansonsten frage einen Kollegen im CSTI, ob er dir etwas Knete geben kann um selbst ein Objekt zu formen. So oder so, solltest du mal wieder an einem Eimerchen Play-Doh-Knete riechen um ein Kindheits-Flashback zu bekommen.

3 Reales Objekt digitalisieren

Now it is getting serious. Nachdem jetzt ein geeignetes Objekt mit Knete geformt oder anderweitig erstellt/gefunden wurde, kann mit dem eigentlichen Teil angefangen werden:

Der erste Schritt in unserer Prozesskette ist das Kopieren des realen Objekts in die digitale Welt. Unser Objekt erfüllt die Anforderungen aus dem vorherigen Kapitel, also können wir den 3D-Scanner "DAVID 3D-SLS" verwenden. Es folgt eine kleine Einführung in den vorliegenden Aufbau mit den jeweiligen Funktionsweisen:

3.1 Benutzte Hardware - 3D Scanner "DAVID SLS-3"

- Projektor-Kamera-System mit Drehteller (DAVID SLS-3)
 - Ein bestimmtes Lichtmuster wird auf das zu scannende Objekt projiziert. Dabei erkennt die Kamera ein durch die Objektmaße verzogenes Lichtmuster und vergleicht es mit dem ursprünglichen Lichtmuster. Der entstandene Versatz wird umgerechnet in die dreidimensionale Anordnung von Bildpunkten aus Sicht der Kamera. Die Bildpunkte werden jeweils zusammengefasst zu Drei- und Vierecken. Bewegt sich der Drehteller, wird der Bereich verändert, welchen die Kamera einsehen kann. Dann findet ein neuer Einzel-Scan statt. Der Vorgang wird solange wiederholt, bis das Objekt sich einmal komplett um seine eigene Achse gedreht hat und von jeder Seite eingescannt wurde. Die dabei entstandenen Einzelscans werden anschließend zu einem Gesamtscan zusammengefügt.
 - Ein Versatz zwischen Projektor und Kamera sollte vor jeder Sitzung neu kalibriert werden um sicherzustellen, dass die Scan-Berechnung korrekt durchgeführt werden.
- Programm "DAVID 3D Scanner Pro v4.5.3"
 - Mit dieser Software wird der Scanner bedient. Es wird beispielsweise der Abstand zwischen Kamera und Projektor definiert. Da die Kamera nicht weiss, was zu dem Scan gehört und was Hintergrund ist, muss hier ebenfalls eine "Hintergrund-Entfernung" vorgenommen werden. Hier einmal die wichtigsten Parameter, die in diesem Programm festgelegt werden:
 - * Anzahl der Einzelscans
 - * Ergebnis-Filterung
 - * Automatisches Zusammenfügen mit vorherigen Scans
 - Des weiteren findet hier die zuvor angesprochene Kalibrierung statt. Die Einzelscans können manuell oder automatisch miteinander verknüpft werden und anschließend als 3D-Modell(!) exportiert werden.

3.2 Anleitung zur Digitalisierung mit DAVID SLS-3 Scanner[Video 1]

Jetzt darfst du endlich selbst Hand anlegen und dein Objekt digitalisieren. Hiernach ist die Verwandlung von Analog zu Digital schon vollbracht (Dr. Frankenstein wäre stolz). Die Anleitung ist hauptsächlich stichpunktartig um dir die Ausführung so kompakt wie möglich anzuzeigen. Du kannst erledigte Schritte direkt abhaken, im ausgedruckten Zustand sogar durchstreichen oder rot-okka-türkis markieren.

1. Überprüfen des Setups

Starten des Computers mit angeschlossenem 3D-Scanner

Öffnen der DAVID-Suite

Abnehmen der Objektivabdeckung an der Kamera

Starten des Scanners an -Button

Im oberem Reiter **Setup** auswählen und folgende Einstellungen vornehmen:

- **Setup Type = "DAVID SLS-3"**
- **Disable Texturing = ✓**
- **Screen ID = (stellt sich automatisch ein)**
- **Camera Setup = DAVID-CAM[...] auswählen**
- **Frame Rate (fps) = high**

Ein Vorschaubild des Scanners sollte in der Mitte des Programms ersichtlich werden.

Wenn kein Bild erscheint, einmal das USB-Kabel von der Kamera raus- und wieder reinstecken. Eventuell muss dann **Camera Setup = DAVID-CAM[...]** erneut ausgewählt werden. Es kann auch versucht werden, die **Screen ID** zu ändern. Diese ID gibt an, über welche Anzeige das SLS-Raster ausgegeben wird. Der Scanner hat in der Regel die **Screen ID = 2**, kann aber auch manchmal einen anderen Wert haben. Dieser muss dann durch Probieren eingestellt werden

2. Kalibrierung

Die Projektor-Scanner-Einheit am Griff der Halterung hoch kippen, bis die Scannervorschau im Programm, den Bereich des mittelgroßen Punktemusters zentriert abbildet

Punktemuster sollte "grob" innerhalb der vier großen, blauen Vierecken liegen

Im Programm auf **Calibration**-Reiter gehen

Folgende Werte unter **Calibration** einstellen:

- **Calibr. Scale [mm] = 60**
- **Orientation = horizontal**

Auf **Calibrate** klicken

Wenn nun ein Fehler erscheint, dass wichtige Marker nicht gefunden wurden, muss der Scanner etwas zentrierter auf das Muster zeigen. Dafür am besten aufstehen und hinter den Scanner stellen um das Lichtmuster möglichst mittig auf den gepunkteten Hintergrund zu richten, bevor erneut **Calibrate** geklickt wird

Projektor-Scanner-Einheit am Griff wieder auf den Drehteller richten

Drehteller sollte genauso "grob" in der Mitte liegen

3. Platzierung des Objekt und Start des Scan-Vorgangs

Das Objekt muss jetzt "grob" in die Mitte des Drehtellers gelegt werden, damit es bei der Drehteller-Rotation nicht "eiert". Der Scan wird dadurch unnötig ungenau.

Im Programm unter **Scanning** müssen jetzt folgende Einstellungen vorgenommen werden:

- **Profile = Default**
- **Enable Turntable = ✓**
- **Total Scan Angle = 360**
Hier wird der Gesamtwinkel angegeben, um welchen sich der Drehteller während des Scan-Vorgangs dreht. Da das einzuscannende Objekt aus jeder Richtung digitalisiert werden soll, muss dementsprechend auch eine komplette Drehung stattfinden
- **Number of Scans = 8**
Hier kann festgelegt werden, wieviele Scans während der 360°-Drehung gemacht werden sollen. Wählt man viele Scans (> 50), gestaltet sich das anschließende Zusammenfügen als recht kompliziert, da alle 50 Scans an dem jeweils vorherig aufgenommenen ausgerichtet werden müssen. Werden jedoch nur wenig Scans (< 6) aufgenommen, besteht das Risiko, dass Objektgeometrien nicht richtig digitalisiert werden. Für diesen Anwendungsfall ist der Wert auf 8 gesetzt. Ein gutes Mittelmaß
- **Auto Align To Previous = ✓**
- **Auto Add to List = ✓**
Damit werden die Einzelscans der Kompletliste hinzugefügt

Start drücken um den Scan-Vorgang zu starten. Es kann beobachtet werden, wie die Aufnahme der Einzelscans stattfindet und sich der Drehteller danach immer etwas weiter dreht. Die Scans werden automatisch aneinander ausgerichtet.

Treten Probleme bei der Software auf dabei oder sind Einzelscans falsch angeordnet worden, wird eine Korrektur nachfolgend vorgenommen

4. Neuordnen der falschen Einzelscans

Eine Änderung des Sichtfeldes wird durch Scrollen oder Mausbewegung bei gedrückter rechter Maustaste bewerkstelligt

Der Auswahlmodus wird jetzt durch Klick auf [Align Scans](#) ausgewählt. Daraufhin wird mit dem ersten Klick der falsch angeordnete Scan ([Scan A](#)) und mit dem zweiten Klick der korrekt angeordnete Basis-Scan ([Scan B](#)) ausgewählt. Nun wird A an B ausgerichtet

Bitte wiederhole den Vorgang, bis zweiter Scan optisch ausreichend gut mit gleichen Bereichen über erstem Scan liegt.

Sollte die Neuordnung auch nach dem vierten oder fünften Versuch nicht richtig funktionieren, können konkrete Kontaktpunkte ([Contact Pair Selection](#)) ausgewählt werden, die dann als Vorschlag für die Software dienen. Das Programm bekommt so eine erste Idee, an welchen Punkten Scan A und Scan B übereinander gelegt werden können

5. Löschen von falschen Punkten

Jetzt das Viereck unterhalb von [Cleaning](#) auswählen. Punkte, welche offensichtlich nicht zum realen Objekt passen sollten jetzt so hingedreht werden, dass sie frei vor dem Hintergrund liegen

Mit gedrückter Maustaste ein Viereck aufspannen und so die Punkte auswählen, die durch Klick auf das [rote X](#) oder [ENTF-Taste](#) gelöscht werden

6. Scans der Unterseite vom Objekt

Da durch den Drehteller nur eine Rotation um die y-Achse ermöglicht wird, muss nun noch eine Drehung um die x-Achse manuell durchgeführt werden, sprich die Unterseite des Objekts abgescannt werden.

Nachdem die Oberseiten-Scans ausgerichtet wurden, müssen sie jetzt mit [Shape Fusion](#) ausgeblendet werden.

Klick auf [Scans](#), was sich auf rechter Bildschirmhälfte befindet

Unterhalb von [List of Scans](#) auf das Auge klicken, sodass es abgewählt ist

Alle Scans sollten ausgeblendet sein und nicht mehr im Hauptfenster (zentrale Fenster, in welcher auch die Scans ausgerichtet wurden) angezeigt werden

Jetzt wieder auf [Setup](#) und einen neuen Scan ausführen. Dafür Schritte *Platzierung des Objekts und Start des Scan-Vorgangs, Neuordnen der falschen Scans* sowie *Löschen von falschen Punkten* wiederholen

7. Zusammenfügen von Unterseiten-Scans zu Oberseite

Die Oberseiten-Scans sollten ausgeblendet worden sein. Nun werden sie alle über [[Shape Fusion](#) -> [List of Scans](#) (rechte Bildschirmhälfte) -> Auge] wieder eingeführt

Da Oberseiten-Scans noch aneinander ausgerichtet sind, müssen neue Scans jetzt an Oberseite ausgerichtet werden.

Es bietet sich an, jeweils einen Unterseiten-Scan wieder mit [[Align Scans](#) und Linksklick] auszuwählen und ihn mit dem Oberseiten-Scan zu verbinden

Wenn die beiden Scans jetzt noch nicht gut übereinander liegen, bitte markante Punkte des angenäherten Unterseiten-Scans mit markanten Punkten des Oberseiten-Scans verknüpfen, um dem Programm erneut das Übereinanderlegen vorzuschlagen.

Ist der erste Unterseiten-Scan nun befriedigend an dem Oberseiten-Gebilde angeschlossen, soll der Vorgang für die nachfolgenden Scans wiederholt werden bis alle Scans zusammen das realen Objekt gut abbilden

Gegebenenfalls kann jetzt nochmal *Löschen von falschen Punkten* durchgeführt werden

8. Fertigstellen des Roh-Modells

Nun wird in **Shape Fusion** die **Fusion-Option Close Holes = ✓** gesetzt. Hiermit werden Bereiche, welche nicht abgescannt wurden, abgeschätzt und aus den detailreichen Einzel-Scans ein Kompletmodell zusammengefügt, welches "wasserfest" ist, also keine Löcher hat

Klick auf **Fuse** um die Verbindung der Scans zu einem Gesamtkörper auszulösen. Das Modell kann nach Berechnung (rote Statusleiste am unteren Bildschirmrand) betrachtet werden

Fallen Fehler auf, können in rechter Bildschirmhälfte unter [**Project -> Scans**] die Einzelscans wieder angezeigt und der "Fuse" rückgängig gemacht werden

Gefällt das Modell, soll es jetzt mit [**Fusions->Export**] als .obj-Datei in **3D2VR -> 3D-Objekt** abgespeichert werden

Jetzt kann die DAVID-Suite geschlossen werden

3.3 LEVEL-UNLOCK

Du hast Erfahrungspunkte gesammelt und eine neue Fähigkeit erlernt. Sie heisst *Real 2 Digital (w) DAVID* - kurz $R2D^2$ und befähigt dich kleine Realobjekte in die digitale Welt umzuwandeln.

4 Digitales Objekt anpassen für VR

Mit unserem Roh-Objekt können wir in VR nur bedingt etwas anfangen. Die Auflösung ist wie bei einem hochauflösenden Digitalfoto zu groß und muss runtergedreht werden, damit der Computer eine Darstellung in VR flüssig berechnen kann. Außerdem hat der 3D-Scanner noch keine Definition vorgenommen, wo Ober- und Unterseite unseres Objektes sind. Einfach gesprochen, weiss die VR somit noch nicht, wie unser Objekt richtig zu stehen hat.

Dies und weitere Optimierungen werden im folgenden Kapitel erklärt.

4.1 Verwendete Programme

- Instant Meshes
 - Hierbei handelt es sich um ein Open Source Programm zum Aufbereiten des 3D-Modells. Es findet eine Aufbereitung der Bildpunkte statt, welche zu Anfang zahlreich und eher "wirr" angeordnet sind. Die Anordnung dieser Bildpunkte nennt sich "Mesh" (engl. für Netz). Drei Punkte bilden dabei das absolut kleinste Netz, jeweils ein planares Dreieck. Wiederum bilden viele Zusammensetzung der Dreiecken("tris") dann das Objekt ab.
 - Es gibt auch Anordnung von vier Punkten. Dann ist das Objekt aus Vierecken zusammengesetzt, sogenannten "quads".

Es sollte prinzipiell immer eine Anordnung der tris oder quads in Bezug auf die Geometrie stattfinden. Beispiel: Ein Würfel sollte aus einem Mesh bestehen, welches sich an den Kanten orientiert. Das macht den Unterschied zwischen einem guten digitalen Objekt und einem schlechten digitalen Objekt.

4 Digitales Objekt anpassen für VR

Warum spielt das überhaupt eine Rolle? Wenn in Zukunft das Objekt eingefärbt werden soll, also eine "Textur" dem Objekt seine Farbe gibt, dann ist diese Textur eine zweidimensionale "Haut". Sie muss also um das Objekt gelegt werden. Bei einem Würfel hätte man beispielsweise sechs Seiten, welche richtig zusammengeklappt um den Würfel herum gelegt werden könnten. Ist der Würfel jedoch nun nicht in sechs Quadrate ("quads") aufgeteilt, fällt es schwer, die Seiten richtig zuzuordnen. Ein Punkt in der Textur wird einem Punkt auf dem Objekt zugeteilt. Und dieser Prozess lässt sich strukturierter erledigen, wenn die einzelnen Punkte ähnlich der Objektgeometrie verlaufen.

- * Durch Instant Meshes können "gedachte Linien" eingetragen werden um die Bildpunkte an der Objektgeometrie zu orientieren. Dieser Vorgang nennt sich in der Fachsprache: Retopologie des Meshes.

Die Auflösung der Bildpunkte kann reduziert werden, da die 3D-Modelle in Relation zur darzustellenden Größe viel zu hochauflösend sind. (Erinnerung: Größe real = Größe virtuell 100000 Bildpunkte)

- Maya

- Dies ist eine Software zur 3D-Visualisierung, -Modellierung und Animation aus dem Hause Autodesk. Eingesetzt wird sie in professionellem Kontext wie Film- und Fernsehproduktion. Es handelt sich um ein sehr großes und mächtiges Programm. In diesem Projekt wird das Programm verwendet um die Drehung des digitalen Objektes anzupassen (also zu definieren wo quasi vorne, hinten etc. ist). Optional kann weitere Polygon-Reduktion mit diesem Programm vorgenommen werden, was jedoch kein Hauptbestandteil dieses Leitfadens ist.

4.2 Anpassen der Objektgröße mit Programm "Instant Meshes" [Video 2]

1. Starten und Reinladen des Objektes in Instant Meshes

Instant Meshes starten

Über [Open Mesh](#) das obj.-File aus [3D2VR](#) -> [3D-Objekt](#) laden

Mit Scrollen heran- und rauszommen sowie mit gedrückter rechter Maustaste die Ansicht verändern und das Objekt betrachten

2. Reduzieren des Meshes und Neuordnen der Quads/Tris

Die Zahl etwas rechts unten von [Target Vertex Count](#) gibt an, wieviel tausend(k) Dreiecke in dem Mesh enthalten sind. Mit dem Schieberegler links von der Zahl wird jetzt eine Anzahl von 50k eingestellt. Später kann hier noch optimiert werden

Nächster Schritt ist Klicken des Buttons [Solve](#), der innerhalb der Kategorie [Orientation field](#) steht

Nachdem [Solve](#) geklickt wurde, findet die Berechnung der häufigsten Dreieck-Orientierungen statt. Sie werden anschließend durch unterschiedlich-farbige Linien repräsentiert

Anschließend wird der [Solve](#) Button unter [Position field](#) gedrückt. Nun werden die Quads ersteinmal berechnet

Jetzt wird das Lineal-Icon unterhalb von [Orientation field](#) angeklickt um Orientierungslinien zu ziehen. Mit gedrückter linken Maustaste werden jetzt Linien gezogen, die sich an prägnanten Geometrien orientieren. Bei einem eingescannten Würfel wären es beispielsweise die Kanten

Nachdem Orientierungs-Linien gezogen sind, werden durch erneutes Klicken des unteren [Solve](#) Buttons die Neu-Orientierungen übernommen

Jetzt wird die endgültige Mesh-Struktur durch das Gitternetz angedeutet

Fallen einem noch unbefriedigende Strukturen auf, kann man erneut mit dem Lineal Orientierungslinien ziehen und neu auf [Solve](#) klicken

3. Exportieren des optimierten Objektes

Jetzt in dem linken Auswahlmnü den Punkt [Export](#) auswählen

Es öffnet sich eine neue Auswahlebene. Nun [Extract mesh](#) und anschließend [Save](#) klicken

Das bearbeitete .obj-File ebenfalls in [3D-Objekt](#) speichern. Bitte darauf achten, dass der Dateiname in folgendem Format ist, damit nachfolgend keine Verwechslung auftritt: *DATEINAME*_reduc.obj

Es ist wichtig, die Endung ".obj" hinter den Dateinamen zu schreiben

Jetzt kann Instant Meshes geschlossen werden

4.3 Anpassen der Objektausrichtung mit Programm "Maya" [Video 3]

Im Gegensatz zu Instant Meshes verfügt das 3D-Modellierungs- und Animationsprogramm "Maya" über viele weitere Funktionen als nur Mesh-Reduktion. Wir bearbeiten mit Maya jedoch nur das Koordinatensystem unseres 3D-Objektes. Anders gesagt soll festgelegt werden, wo in dem Objekt die Seiten oben, unten, vorne und hinten sind. Zwar wurden mit dem Scanner Ober- und Unterseiten-Scans aufgenommen, jedoch ist damit noch nicht sicher gestellt, dass die korrekte Standardausrichtung besteht. Um einen kleinen Einblick in die Software zu bekommen und erste Erfahrungen zu sammeln, wird die sogenannte "Koordinatentransformation" durchgeführt:

1. Öffnen des Roh-Objektes

Maya 2018 starten

Über **File** -> **Import** den Auswahldialog aufrufen, und das zuvor erstellte .obj-File auswählen. Das Objekt wird nun, ähnlich zu Instant Meshes, im zentralen Bildschirm dargestellt

Falls eine andere Ansicht eingestellt ist, wird an der oberen Ecke des zentralen Bildschirms **Panel**s -> **Saved Layouts** -> **Single Perspective View** ausgewählt um die Standardansicht zu aktivieren

Irgendwo in der Objektanzeige ist ein viereckiges Gitter, welches den Referenzboden für Maya abbildet. Die Ansicht kann gedreht werden über **ALT-Taste + Linksklick + Mausbewegung**. Sofern unser Objekt nicht genau auf diesem Gitter steht, müssen noch Anpassungen bei dem Koordinatensystem vorgenommen werden

2. Anzeige und Zentrierung des Koordinatensystems

Jetzt das Objekt anklicken um es auszuwählen und das Koordinatensystem anzuzeigen. Nun sehen wir drei unterschiedlich farbige Balken, welche für die drei räumlichen Achsen unseres Objektes stehen

(rot = x-Achse, grün = y-Achse, blau = z-Achse).

Diese Achsen sind insofern wichtig, da sie in direkter Verbindung mit dem Objekt stehen und wir sie quasi als "Griffe" für die Verschiebung nutzen können.

Nun gibt es drei unterschiedliche Modi, in welchen jeweils eine Veränderung in der Skalierung, der Rotation (Verdrehung) oder der Translation (Verschiebung) bewirkt werden kann.

Der ausgewählte Modus lässt sich an der Form ablesen, welche jeweils am Ende der drei Achsen hängt:

- Kegel = Verschiebung (W-Taste)
- Kugel um Objekt = Verdrehung (E-Taste)
- Würfel = Skalierung (R-Taste)

Es fällt sicherlich auf, dass das Koordinatensystem nicht wirklich zentral liegt. Spätestens bei der Rotation könnte das zu Problemen führen, wenn nicht um den Objektmittelpunkt gedreht wird

Zentriere das Koordinaten-system/-kreuz über die oberste Auswahlleiste: **Modify** -> **Center Pivot**. Pivot steht in diesem Fall für unser Koordinatensystem

3. Verdrehung und Position des Koordinatensystems optimieren

Wenn der Pivot zentral liegt, die E-Taste drücken um die Drehung zu verändern

Nun verhält es sich wie mit einem Lenkrad, um jede Achse ist ein Kreis gezogen, welcher wenn gedrückt gehalten, eine Verdrehung ermöglicht. Jetzt sollte überlegt werden wo der Boden des Objektes ist. Das Objekt bitte so verdrehen, dass es mit seinem Boden auf dem Gitter stehen könnte, wenn es dort hin verschoben wäre.

Jetzt ist die richtige Position an der Reihe. Drücke die W-Taste um den Modus zu aktivieren, bei dem die Achsen mit Kegeln abschließen und so Pfeile darstellen. Ziel ist es jetzt, das Objekt möglichst zentral auf die Gitterebene zu ziehen

Die benötigten Achsen werden dann mit gedrückter linker Maustaste aktiviert und mit der Mausbewegung in gewünschte Richtung gezogen

Ist das Objekt noch viel größer als die Gitterebene, kann in dem [Outliner](#) in der linken Bildschirmhälfte das `_*OBJEKTNAME*_reduc:Mesh` angeklickt werden. In der rechten Bildschirmhälfte können jetzt unter dem Reiter [Transform Attributes](#) die Skalierungswerte in allen drei Dimensionen mit [Scale](#) verkleinert werden.

Bitte den Scale von 1 in allen drei Feldern auf 0.5 setzen

Abgeschlossen wird der Vorgang mit dem Neusetzen des aktualisierten Koordinatensystems. Klicke dafür [Modify -> Freeze Transformation](#) um das verbesserte Koordinatensystem zu übernehmen und die Verdrehungs-/Verschiebungswerte auf 0 zu setzen. Wenn jetzt beispielsweise die Werte auf 0 geresetzt werden, landet das Objekt immer wieder in dieser Position

4. Ändern der Ansicht um Optimierung zu überprüfen

Eine Überprüfung wird jetzt über die unterste Auswahlleiste in der oberen Bildschirmhälfte vorgenommen.

Dafür wird die Ansicht verändert. Über [Panels -> Saved Layouts -> Four View](#) eine vierfache Anzeige aus unterschiedlichen Perspektiven aktivieren.

Falls in einigen Anzeigen die Außengeometrie des Objektes nicht ersichtlich ist, kann durch Anklicken der Ansicht und scrollen einfach rein- und rausgezoomt werden

Das Objekt sollte nun in den vier Ansichten gut zu betrachten sein. Dabei ist es wichtig, dass die Ansichtsnamen mit den tatsächlich zu betrachtenden Ansichten übereinstimmen.

Beispiel: [top - Y](#) sollte auch tatsächlich die Ansicht von oben sein. Die Ansichtstitel sind in dem jeweiligen Fenster unten-mittig positioniert.

5. Koordinatensystem aus dem Mittelpunkt verschieben

Abschließend muss das Koordinatensystem auf den Objektboden gezogen werden, damit es später in der Entwicklungsumgebung für die VR-Brille auch tatsächlich auf dem Boden steht und von dort aus positioniert werden kann:

Dafür jetzt die Einfügen-Taste drücken. Zu dem Koordinatenkreuz wird jetzt wieder eine Kugel zum Verdrehen angezeigt. Außerdem steht über dem Mittelpunkt **align**, was für "Ausrichten" (des Koordinatenkreuzes) steht

Jetzt bleibt das Objekt stehen und das Koordinatenkreuz kann verschoben werden. Mit den drei Achspfeilen bitte nun das Koordinatensystem in der **front-Z**-Ansicht an den Boden des Objekts ziehen.

In der **top-Y**-Ansicht bitte ebenfalls das Koordinatensystem unter dem Objekt zentrieren

4.4 Abspeichern des Objektes [Video 4]

- Unser Objekt ist nun fertig und sollte auch richtig abgespeichert werden:

Um die ganze Bearbeitungsvergangenheit zu löschen, die unnötig Speicherplatz aufbraucht, wird **Edit -> Delete All By Type -> History** ausgewählt

Falls das Objekt im Nachhinein doch nochmal bearbeitet werden soll, kann die Maya-Projektdatei abgespeichert werden. Dafür **File -> Save Scene As** anklicken und den 3D-Objekt-Ordner auswählen.

Anschließend den Dateinamen = *DATEINAME*_reduc_coord.obj bei **File name:** eintragen und auf **Save As** klicken

Zum Schluss muss die .obj-Datei exportiert werden, welche im nächsten Schritt in Unity eingebunden wird. **File -> Export All** auswählen und den Projektordner von eben raussuchen

Den gleichen Namen wie für das Projekt eingeben und bei **Files of type:** die letzte Option **OBJexport** auswählen. Klick auf **Export All** um den Vorgang abzuschließen

Sofern nicht noch **3D-Objekt** als letzter Ort ausgewählt ist, bitte dort hin wechseln und abspeichern

Jetzt kann Maya geschlossen werden

4.5 LEVEL-UNLOCK

Du hast Erfahrungspunkte gesammelt und etwas neues erlernt. Es ist die Fähigkeit (zur) Reduktion (und) Orientierung (des) digitalen Objekts - kurz FRODO. Sie befähigt dich komplexe Modifikationen an einem 3D-Objekt vorzunehmen. Ziemlich mächtig!

5 Übertragung auf die VR-Brille

5.1 Programmierung mit Software "Unity"

Unity ist eine "Spiel-Engine", also Entwicklungsumgebung für Spiele. Diese Umgebung(Plattform) liefert wichtige Grundsätze für die Spiele-Entwicklung wie: grafische Darstellung; Programmierer; Physik-Eigenschaften

Eine passende Metapher ist dabei der Tuschkasten und das zu malende Bild. Mit dem Tuschkasten (Engine) werden wichtige Werkzeuge geliefert, jedoch gibt es auf lange Sicht auch Restriktionen. Es ist nur eine bestimmte Auswahl an Farben (Grafiken, physikalischen Genauigkeiten) vorhanden und so werden die Tuschkästen erweitert und optimiert.

Für die meisten Maler (Programmierer/Spiele-Entwickler) sind die vorhandenen Farben jedoch völlig ausreichend und es kann ein Bild (Computerspiel) nach dem anderen gemalt (entwickelt) werden.

Ein "Computerspiel" ist in diesem Fall alles, was gestartet werden kann und eine bestimmte Funktionalität bietet. In diesem Fall handelt es sich um das Anzeigen eines Objektes und die Möglichkeit es zu greifen sowie sich innerhalb des Aktionsradius frei bewegen zu können.

"Malen" bedeutet in diesem Fall das Erstellen von digitalen Räumen sowie die Programmierung von bestimmten Abläufen und Eigenschaften.

Programmiert wird in C-sharp, eine für Microsoft entwickelte Sprache.

"Sprache" als Analogie bietet sich sehr an, da es beim Programmieren prinzipiell darum geht, die Vorstellung welche im eigenen Kopf ist, für den Computer zu "übersetzen" und verständlich zu machen.

5.2 Öffnen der Beispielszene in "Unity" [Video 5]

Da wir uns jetzt in großen Schritten Richtung VR-Brille bewegen, sollte dieser Teil des Leitfadens an einem Computer ausgeführt werden, der über ein angeschlossenes VR-Set verfügt. Im CSTI sind das die Rechner CSTI-PCGH08 und CSTI-PCGH05, die direkt vor der Traverse stehen. Sie verfügen über genügend Rechenleistung um die Darstellung gut umzusetzen. Außerdem ist das Setup der Hardware, beispielsweise die Definition des begehbaren Bereiches in VR, schon durchgeführt und wir können direkt mit dem spaßigen Teil fortfahren.

Falls du es noch nicht getan hast, kopiere deine bearbeiteten Projektdateien auf einen USB-Stick, logge dich an einem der oben genannten Rechner ein und kopiere den Projektordner auf den Desktop. Ich verweise an dieser Stelle auf die [Feisty Crab Studios](#), von denen ich Programmcode aus dem Tutorial "Throwing Objects" kopiert habe, um dieses Beispielprojekt zu erstellen.

1. Unity starten und Profil erstellen

Auf dem Desktop machst du außerdem einen Doppelklick auf das [Unity 2017.2.0f3](#)-Logo um Unity zu starten

Der Startbildschirm erwartet nun eine Anmeldung in das eigene Entwicklerkonto. Wähle eine der drei Anmeldemöglichkeiten ([Sign in with google](#), [Sign in with facebook](#), If you don't have a Unity ID, please [create one](#).) um fortzufahren

Die Anmeldung bei Unity hat unter anderem den Vorteil, bezahlte Programmerweiterungen (wird später drauf eingegangen) auf unterschiedlichen Geräten verfügbar zu machen.

2. Projekt öffnen und mit Fehlermeldungen umgehen

Hat die Anmeldung funktioniert, jetzt in der oberen rechten Bildschirmhälfte über **Open** das Projekt-Auswahlfenster öffnen

In dem Auswahlfenster bitte den Ordner **3D2VR** suchen

Bitte ein Backup des Unity-Projektordners erstellen, indem der Unity-Projektordner **3D2VR -> Objektumgebung_v1** in **3D2VR** kopiert wird.

Jetzt den Originalordner auswählen und mit **Open folder** öffnen

Es erfolgt als erstes ein Versionsvergleich der Entwicklungsumgebungen. Ist das Projekt mit einer älteren Version erstellt, als aktuell benutzt, kann es zu Inkompatibilitäten kommen.

Aus diesem Grund sind häufig mehrere Unity-Versionen auf den Computern zu finden. Dieses Beispiel ist jedoch recht grundsätzlich gehalten und dadurch auch mit der neuen Version problemlos zu nutzen

Einfach mit **Continue** fortfahren

Anschließend erfolgt eine Überprüfung benutzter "APIs". Bei einem **Application Programming Interface** handelt es sich um die Schnittstelle zwischen Programmiersprache und programmierter Hardware.

Die Hardware spricht noch keine Hochsprache (C-sharp) sondern muss von dem Computer mit einfacheren Befehlen angesprochen werden. Diese Verbindung stellt die API dar

Es kann vorkommen, dass die APIs nicht mehr aktuell sind und deshalb Unity einen Fehler anzeigt. Es wird daher vorgeschlagen ein Backup zu machen.

Bestätigt wird diese Fehlermeldung durch **I Made a Backup, Go ahead!**.

Nun öffnet sich das Hauptfenster von Unity und eventuell werden weitere Fehlermeldungen eingeblendet.

"Failed to write license file" beispielsweise einfach durch **OK** bestätigen. In diesem Beispiel irrelevant.

Auch kann "SteamVR" um eine Optimierung des Projektes bitten. Einfach mit **Ignore All** ignorieren und andere Aktualisierungsaufforderungen mit ähnlichen Buttons wegklicken. Unsere Version funktioniert auch mit älteren Versionen

5.3 Erklärung der Benutzeroberfläche von "Unity" [Video 6]

Unity überwältigt erst einmal mit mehreren Unterfenstern. Abgesehen von bekannten Funktionen sind jedoch gar nicht so viele neue Fenster zu erkennen.

Der obere Bereich wird durch eine Auswahlleiste gefüllt, welche die hauptsächlichen Einstellungen wählbar macht. Findet man sich nicht zurecht oder sucht einen sicheren Weg zur benötigten Funktion, sollte man in dieser Auswahlleiste suchen.

Im linken Bereich des übergeordneten Fensters befindet sich die Projektstruktur ([Hierarchy](#)). Dort ist das erste Level (Szene) geöffnet und alle eingebundenen Objekte sind in einer Baumstruktur untergeordnet.

Wird beispielsweise ein Spiel mit mehreren Leveln programmiert, kann man die Level auch mit Szenen gleich setzen. Es besteht also eine Film-Analogie und man sieht, dass Unity auch angelegt ist um größere Projekte zu realisieren. Dieses Beispiel kommt jedoch erst einmal mit einer Szene aus.

Das prominente, mittlere Fenster beinhaltet die Übersicht über die komplette Szene ([Scene](#)).

Den Mauscursor über diesem Fenster halten und die rechte Maustaste halten, um den Blickwinkel zu ändern. Die Bewegung nach vorne und hinten durch Mausrad-Scrollen ebenfalls testen.

Die Sicht des Spielers, ist über das rechte zentrale Hauptfenster ersichtlich ([Game](#))

Am rechten Rand können Eigenschaften der Szenen-Objekte bearbeitet werden ([Inspector](#)). Nun beispielsweise [[Camera Rig](#)] anklicken, um dort Parameter wie Drehung oder Position der Spieler-Ansicht zu verändern.

Im unteren Bereich des Bildschirms ist ebenfalls eine Baumstruktur zu sehen, über welche alle möglichen Arten von Objekten und Programmen (hier als Skripte benannt, die prinzipiell mit jedem Objekt arbeiten sollen) auswählbar sind ([Project](#)).

Hier einen Ordner auswählen, um den Inhalt im zentralen, unteren Fenster anzuzeigen

5.4 Einfügen des eigenen Objekts in "Unity" [Video 7]

1. Um nun das eigene Objekt in die Szene einzufügen muss es zuerst dem Projekt beigefügt werden. Das Hinzufügen findet wie folgt statt:

Aus dem Windows-Datei-Explorer nun die .obj-Datei per Drag-n-Drop in den [Objects](#)-Ordner in der Baumstruktur von [Project](#) ziehen

Nachdem der Ordner ausgewählt ist, wird der Inhalt im unteren zentralen Bereich angezeigt. Das Objekt ist jetzt dem Projekt hinzugefügt. Jedoch wurde es noch keiner Szene zugeordnet.

Sofern das Vorschaubild, in etwa dem eigenen Objekt entspricht, bitte mit der linken Maustaste gedrückt, irgendwo in das [Hierarchy](#)-Fenster ziehen

Jetzt befindet sich das Objekt in der Szene. Bitte das Objekt in dem [Scene](#)-Fenster finden und anklicken. Die Navigation innerhalb der Szene findet, wie zuvor genannt, über die gedrückte rechte Maustaste und das Scrollrad statt

2. Nun ist die Wahrscheinlichkeit recht hoch, dass die Größe viel zu hoch ist. Das liegt daran, dass Unity eine gewisse Menge an Bildpunkten auf eine gewisse Größe skaliert. Das Objekt muss an dieser Stelle verkleinert werden.

Linksklick auf das Object in der Szenen-Hierarchie um die Objektparameter in dem **Inspector** anzuzeigen

Bei **Transform** sind **Position**, **Rotation** und **Scale** angezeigt.

Nun bitte alle drei **Scale**-Werte (x-,y-,z-Komponenten) verkleinern, bis die Größe in etwa der realen Größe entspricht. Verändere die Werte von **1** zu **0.1**

Das Objekt wurde jetzt um Faktor 10 verkleinert. Wenn es immer noch zu groß ist, nimm statt **0.1** jetzt **0.01**.

Bitte die Skalierung verändern, bis die Größe in Ordnung ist. Wichtig ist dabei jedoch, dass alle drei Komponenten jeweils den gleichen Wert haben. Das Objekt würde sonst verzogen werden, da die Proportionen nicht mehr stimmen

3. So weit, so gut! Dein kreierte Objekt ist eingefügt und eigentlich müsste jetzt doch schon die Brille aufgesetzt werden können um das Objekt in der VR zu betrachten...

5.5 Anhängen der physikalischen Eigenschaften in "Unity" [Video 8]

Wie bei so mancher Versetzung in der Schule macht die Physik nun Stress. Nicht die reale Physik, sondern die virtuelle Physik. Dem Objekt muss erst noch ein Gewicht-Parameter hinzugefügt werden. Es könnte ja schließlich aus Eisen oder leichtem Holz sein. Sogar die Schwerelosigkeit ist in der VR durch ein paar Klicks zu bewerkstelligen.

Wie wird dem Objekt diese Eigenschaften angehängt? Das Stichwort lautet **Rigid Body**.

Die wörtliche Übersetzung lautet "starrer Körper" und definiert in der technischen Mechanik ein Modell mit bestimmter Masse. Er ist nicht verformbar. Sehr gut, das Objekt kann in der VR also nicht zerstört werden.

1. Rigid-Body Eigenschaft dem eingefügten Objekt anhängen:

Beim **Inspector** unterhalb von **Scale** auf **Add Component** klicken

Die verschiedenen Kategorien von möglichen Objektparametern werden angezeigt. Wir interessieren uns für **Physics** und klicken drauf

Nun sind die zugehörigen Eigenschaften eingeblendet. Wähle **Rigidbody** aus

Unterhalb von **Transform** steht nun **Rigidbody** mit seinen Parametern. Unter anderem kann hier die Masse (**Mass**) oder der Luftwiderstand (**Drag**) festgelegt werden. Von Interesse ist für uns erst einmal die Masse, welche auf 1 eingestellt wird. Die entsprechende Einheit ist Kilogramm.

Nun muss außerdem **Is Kinematic** = gesetzt sein und der **Tag** rechts oben unter **Inspector** und Objektname auf **Pickupable** gesetzt werden.

Um das Objekt anheben zu können, läuft im Hintergrund ein C-Sharp-Skript, welches jedoch nur Objekte verschiebbar macht, die dementsprechend getaggt sind

2. Nun sind wir nur noch wenige Schritte von dem Virtuellen Raum entfernt.

Unser Objekt wurde mit dem 3D-Scanner zwar digitalisiert, jedoch fehlt Unity noch die Information über die Außenmaße, an welchen das Objekt gehalten werden kann oder auch abgestoßen wird.

Collider hinzufügen:

Beim **Inspector** unterhalb von **Rigidbody** wieder auf **Add Component** klicken

Wieder **Physics** auswählen

Nun sind unter **Rigidbody** mehrere Collider eingeblendet. Zur Auswahl stehen für uns einer der folgenden drei: **Box Collider**, **Sphere Collider**, **Capsule Collider**. Wähle den Collider, von dem du sagen würdest, dass er am ehesten wie das Objekt aussieht

Zu sehen ist der Collider nach dem Einfügen wahrscheinlich noch nicht. Die Größe und Position muss angepasst werden. Für die unterschiedlichen Parameter der Collider-Typen findet es wie folgt statt:

Cursor als Erstes auf den Parameternamen setzen und mit gedrückter linker Maustaste den Wert nach rechts (größer) oder links (kleiner) ziehen. Da die Collider-Formen unterschiedlich definiert sind, unterscheiden sich auch die Parameter. Vorzunehmende Änderungen sind hier erklärt:

- Capsule: **Height** anpassen, bis Höhe in etwa mit Objekt übereinstimmt. Vorgang wiederholen für **Radius**. Wahrscheinlich umschließt Capsule das Objekt nicht komplett. Die **Center**-Komponenten also anpassen, bis Capsule das Objekt komplett einhüllt
- Box: **Size**-Parameter so groß wählen, bis die Größen in etwa übereinstimmen. **Center**-Wert so verändern, bis Collider und Objekt in etwa deckungsgleich sind
- Sphere: **Radius**-Wert vergrößern, bis Objekt in die Kugel passen würde. **Center**-Werte so verändern, bis die Kugel das Objekt komplett einschließt

Grundsätzlich müssen Collider und Objekt nicht perfekt übereinanderliegen. Eventuell lässt sich das Objekt bei schlechter Positionierung des Colliders nicht ganz so gut greifen, aber ein Problem beim Ausführen wird keinesfalls entstehen.

5.6 Eintritt in den Virtuellen Raum mit Programm "Steam" [Video 9]

Es handelt sich hierbei um eine Plattform, welche hauptsächlich als Marktplatz für PC-Videospiele bekannt geworden ist. Unter anderem liefert sie aber auch das Programm "SteamVR", welches als Treiber-Software für unterschiedliche VR-Brillen funktioniert. Unter anderem wird unsere HTC Vive unterstützt.

Drücke die Windows-Taste und suche nach dem Programm [Steam](#)

Starte es und falls es sich nicht automatisch an einem Profil anmeldet oder du noch keinen Account besitzt, erstelle dir einen neuen über [CREATE NEW ACCOUNT](#)

Gebe die benötigten Informationen ein und klicke dich durch

Nachdem der Account erstellt wurde, logge dich mit deinen neuen Benutzerdaten ein und wechsel zurück zu Unity

Du hast es geschafft. Du bist wirklich nur noch einen Klick vom Ziel entfernt. Hoffentlich kannst du etwas aus dieser Erfahrung mitnehmen. Im allerbesten Fall bist du jetzt richtig interessiert und arbeitest dich weiter in die Thematik ein. Kopiere dir doch die Projektdateien um in Zukunft eine Grundlage für neue Szenen zu haben. Ich freue mich sehr, dass du mit diesem Leitfaden gearbeitet hast und bedanke mich nochmal ausdrücklich beim kompletten CSTI-Team, welches mir erst ermöglicht hat, diesen Leitfaden aufzubauen.

Sicherlich bist du an der einen oder anderen Stelle mal hängen geblieben und bist dir nicht sicher ob du das alleine gebacken bekommst. Im Internet wirst du auf die merkwürdigsten Fragen eine Antwort finden, scheu dich nicht zu googlen! Selbst bezahlte Entwickler formulieren ihre Fragen im Netz oder müssen Sachen in der offiziellen Dokumentation nachschlagen. Profitiere davon indem du dich schlau machst!

Suche dir einen VIVE-Controller und drücke den Knopf unterhalb der großen, runden Touchfläche. Neben dem Button ist eine LED, welche jetzt anfangen sollte zu leuchten

Drücke auf den Play-Button in der obere Bildschirmmitte

Setze die HTC Vive auf, welche mit dem Computer verbunden ist

Geniesse