

# Bachelorarbeit

Malte Habighorst  
Matrikelnummer: 15413020

Welchen Einfluss haben Virtual Production Techniken auf den Workflow in  
Animationsfilmen jetzt und in der Zukunft?

Fachbereich 2: Medienproduktion  
Eingereicht: 09.02.2023  
Detmold Wintersemester 2022/2023

Erstprüfer: PROF. DIPL.-ING. Rico Dober  
Zweitprüferin: B. DES Jennifer Meier

Lizenz: CC-BY-SA 4.0

# Inhaltsverzeichnis

1	Einleitung .....	4
2	Aufbau des Workflows bei einem klassischen Animationsfilm .....	5
3	Motion Capture .....	7
3.1	Geschichte .....	7
3.2	Funktionsweise.....	9
3.2.1	Begriffsklärung.....	9
3.2.2	Motion Capture Methoden .....	9
3.2.3	Facial Capture .....	14
4	Virtuelle Kamera .....	16
4.1	Geschichte .....	16
4.2	Funktionsweise.....	17
5	Game Engine .....	19
5.1	Geschichte .....	19
5.2	Funktionsweise.....	20
5.2.1	Was bedeutet Echtzeit .....	20
5.2.2	Inhalte einer Game Engine.....	20
5.2.3	Wie funktioniert ein Raytracer .....	21
6	Stand der Technik.....	22
6.1	Realfilm .....	22
6.2	Animationsfilm.....	24
6.3	Pre-Visualisierung .....	26
6.3.1	Was ist Pre-Visualisierung.....	26
7	Unterschiede bei dem Workflow .....	28
7.1	Unterschiede zwischen Motion Capture und Keyframe Animation .....	28
7.2	Unterschiede zwischen virtueller Kamera und Keyframe Animierter Kamera.....	30
7.3	Unterschiede zwischen dem Arbeiten mit einer Game-Engine und einem klassischen 3D-Programmes .....	31
7.3.1	Rendering .....	32
7.3.2	Layout.....	36

7.4	Shading.....	38
7.4.1	Visuelle Effekte .....	39
8	Fazit.....	40
9	Literaturverzeichnis.....	42
10	Eidesstattliche Erklärung.....	47

# 1 Einleitung

Mit dem technischen Fortschritt in den letzten Jahren hat sich auch viel in der Film- und Spielebranche geändert. Viele Techniken haben sich immer weiter angeglichen. Mit neuen Rendertechnologien ist es mittlerweile einfach geworden, Bilder zu generieren, die von der Realität nicht mehr zu unterscheiden sind und das in Bruchteilen einer Sekunde. Seitdem ist das Problem nicht mehr technisch, sondern hauptsächlich kreativ. Hier kommt Virtual Production ins Spiel. Es kombiniert bekannte Techniken aus der Filmindustrie mit anderen aus der Spieleindustrie. Das Ergebnis in vielen Bereichen ist ein intuitiverer und schnellerer Workflow. Dennoch müssen einige Kompromisse eingegangen werden. Diese haben ebenfalls einen Effekt auf den Arbeitsablauf und das Endergebnis mit Vor- und Nachteilen. In dieser Arbeit soll es darum gehen, die Techniken zu erklären, wie sie entstanden sind und welche Unterschiede es zum klassischen Workflow gibt.

Um diese Werkzeuge den Artists zur Verfügung zu stellen, muss vorher entschieden werden, welche technischen Hilfsmittel in der Produktion genutzt werden sollen. Hierüber soll diese Arbeit einen Überblick geben. Es soll den Entscheidern aufzeigen, welche Möglichkeiten es gibt. Gleichzeitig gibt es den Artists die Möglichkeit nachzulesen, auf was sie in ihrem Bereich achten müssen.

Neben den technischen und kreativen Aspekten der verschiedenen Techniken soll es auch einen Überblick über die Geschichte der Arbeitsmethoden geben. Unter anderem auch, wie Virtual Production sich mit der Zeit entwickelt hat und wann und wo die verschiedenen Techniken kombiniert wurden.

## **2 Aufbau des Workflows bei einem klassischen Animationsfilm**

Nachdem das Drehbuch und das Storyboard feststehen, fängt das Layout-Department an ein erstes Blockout von der Szene zu erstellen. Währenddessen beginnt das Modelling-Department, die Modelle für die finale Szene zu modellieren. Wenn die Modelle finalisiert sind, werden sie an das Texture (oder auch Shading oder Surfacing Department genannt) übergeben. Die kümmern sich dann um die Texturen und Shader für die jeweiligen Objekte. Bei den Charakteren ist von Studio zu Studio unterschiedlich, ob diese Aufgaben vom Modelling Team übernommen oder in einem extra Department bearbeitet werden. Sobald die Charaktere grundlegend fertig sind, werden sie an das Rigging weitergegeben. Das Rigging Department bereitet die Charaktere und die Requisiten, die animiert werden sollen, dafür vor. Ein Rig versieht das 3D-Modell zum einen mit Deformern und oder mit einem Skeleton. Damit es animiert werden kann. Zum anderen wird ein „Control-Rig“ kreiert, damit kann der Animator das Modell animieren, ohne das 3D-Modell direkt zu beeinflussen. Dieser extra Schritt macht das Modell unabhängig von der Animation und gibt die Möglichkeit, das Modell auch während oder nach dem Animieren zu bearbeiten.

Sobald das Animation-Department seine Arbeit beendet hat, übernimmt das VFX-Department die Sequenzen. Dieses Department ist unterteilt in zwei voneinander unabhängige Departments. Das VFX-Department kümmert sich um die Effekte in der Umgebung und alles, was nicht an den Charakteren bearbeitet werden muss. Für die Charaktere ist das CFX-Department verantwortlich. Es kümmert sich um die Character Effects, es ist zuständig für Simulationen an den Charakteren, zum Beispiel die Haare, Kleidung oder, wenn notwendig, auch um die Muskelsimulationen.

Sobald diese Arbeiten alle final beendet sind, gehen sie zum zweiten Mal zum Layout-Department. Das finalisiert die Shots mit ein paar Details im Set-Dressing. Manchmal kümmert sich das Layout-Department auch noch um die Kameraeinstellungen und das Licht. Häufig wird das aber auch im Lightning Department erledigt. Dies schickt dann die finalen Shots zum Renderer. Nachdem der Renderer die Bilder berechnet hat, gehen die gerenderten Bilder zum Compositing. Dort bekommen die Bilder ihren finalen Schliff: Es werden kleinere Fehler aus dem Bild heraus retuschiert. Weitere Elemente eingefügt wie Matte-Paintings für den Hintergrund oder auch Regen, welcher häufig zu aufwendig ist zu rendern. Zum Schluss wird das finale Color Grading (The Digital Production Pipeline | SIGGRAPH Courses - YouTube 2023)

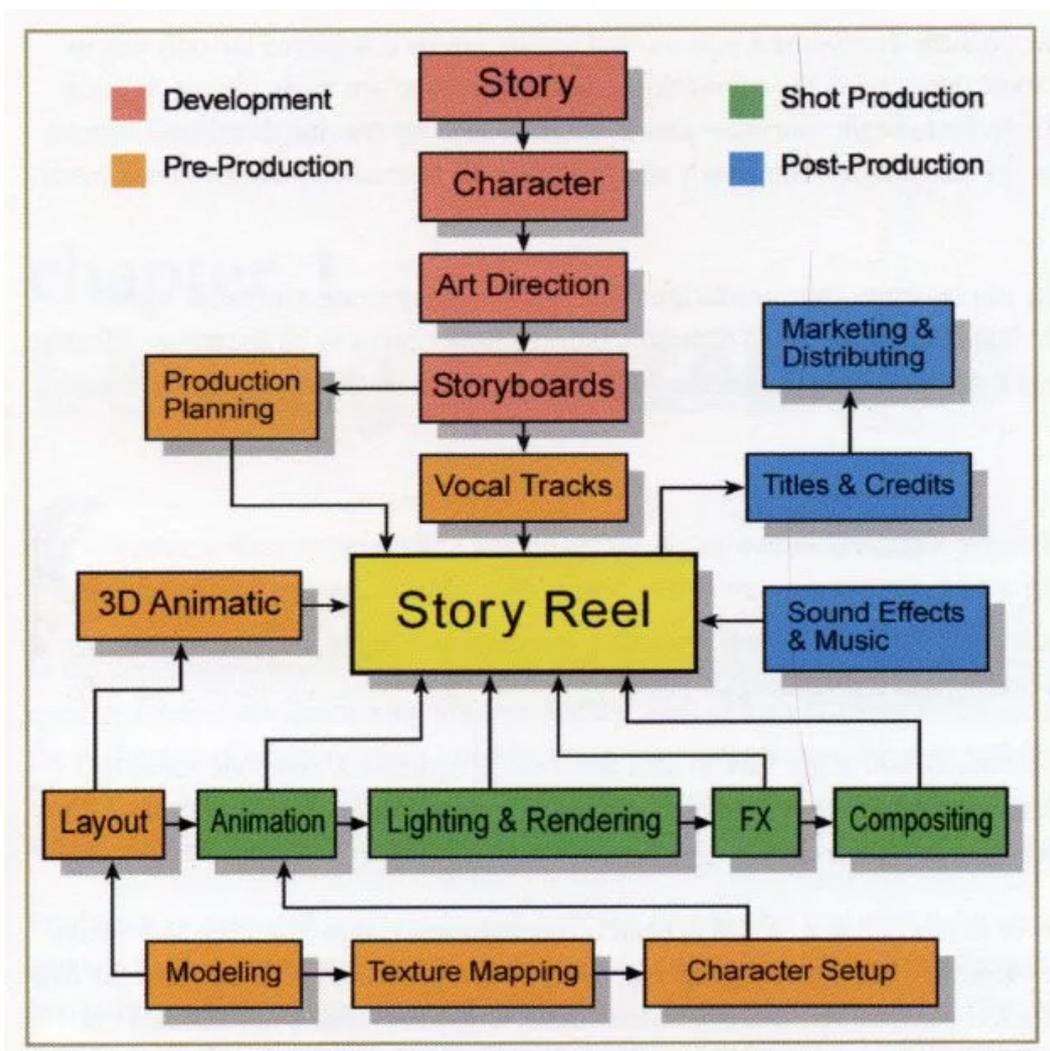


Abb. 1: Aufbau und Ablauf der Produktion eines klassischen Animationsfilms

## 3 Motion Capture

### 3.1 Geschichte

Motion Capture ist unter den in dieser Arbeit besprochenen Verfahren die Technik, die bereits am längsten in der Filmindustrie eingesetzt wird.

Als die ersten Versuche, Bewegungen einzufangen und durch eine andere Animation zu ersetzen, kann das schon früh entwickelte Rotoscoping Verfahren gesehen werden. Rotoscoping wurde 1914 von Max Fleischer entwickelt. Es beschreibt eine Apparatur, bei der ein Frame eines Films auf eine Mattscheibe projiziert wurde. Anschließend wurde davor eine Folie gespannt, auf die eine Maske gezeichnet werden konnte. Diese wurde dann über den Film gelegt und neu belichtet. Die grundlegende Technik wurde auch in die digitale Bearbeitung übertragen und wird bis heute verwendet.

Viele Techniken, die wir unter dem Begriff „Virtual Production“ zusammenfassen, wurde das heute übliche Motion Capture, zuerst in der Spieleindustrie eingesetzt. Als wichtiger Meilenstein gilt zum einen das 1993 herausgekommene „Mortal Kombat“. Dort wurden Schauspieler und Kampfkünstler vor einem Bluescreen gefilmt. Danach wurden die Personen freigestellt und in das Spiel eingefügt. Dieses Spiel gilt als das Erste, in dem reale Bewegungen in ein digitales Produkt übertragen wurden.

1994 wurde dann für das Spiel „Rise of the Robots“ zum ersten Mal ein Motion Capture System genutzt, das direkt die Bewegungen der Schauspieler in digitale Daten umwandelte. Seitdem wird Motion Capture in Spielen sehr aktiv genutzt. Das liegt unter anderem daran, dass Games, anders als Filme, nicht auf real gedrehtes Material zurückgreifen können. Gerade, wenn die Grafik einen fotorealistischen Anspruch hat, ist Motion Capture ein praktisches Werkzeug, um die Animationen so realistisch wie möglich zu gestalten. Damit konnten auch Schauspieler für die Animationen genutzt werden und die Grenze zwischen Filmen und Spielen wurden fließender. So wurden interaktive Filme möglich. Bestes Beispiel dafür ist „Beyond Two Souls“ von Quantic Dream aus dem Jahr 2013. In diesem Spiel wurden alle Zwischen- und Dialogszenen durch ein Performance Capture System aufgenommen und auf die digitalen Charaktere, die auf Scans der realen Schauspieler basieren, übertragen.

In der Filmindustrie wurde mit dem Aufkommen von CGI auch das Verlangen nach realistischen Animationen größer. Seit Anfang der 2000er gehört deshalb Motion Capture in den Werkzeugkoffer der Filmindustrie. Das erste Mal, dass Motion Capture in einem Animationsfilm genutzt wurde, um einen Großteil der Animationen zu erstellen, war „Sinbad Beyond the Veil of the Mists“ aus dem Jahr 2000.

2001 war „Der Herr der Ringe“ der erste Film in dem Motion Capture eingesetzt wurde und kommerziell erfolgreich war. Gollum wurde dort von Andy Serkis gespielt. Allerdings mussten die aufgenommenen Clips sehr stark nachbearbeitet oder auch komplett nachanimiert werden. Nach „Sinbad“ gab es immer wieder neue Versuche, Motion Capture auch im Animationsbereich zu nutzen. 2001 kam „Final Fantasy“ raus. Dieser Film wurde vor allem durch damals besonders fortschrittliche Renderqualität bekannt, wird aber fälschlicherweise immer als der erste Animationsfilm genannt, der Motion Capture genutzt hat. Der nächste große Schritt in der Motion Capture Technologie war 2009, als James Cameron „Avatar“ in die Kinos brachte. (Gray 2014)

Motion Capture hat seine Anwendung auch in medizinischen Bereichen gefunden. Es findet Verwendung in der Diagnose von Fehlstellungen oder in der Rehabilitationstherapie, im Sport zum Analysieren und wird zum Optimieren von Bewegungsabläufen genutzt. (Square-One-Case-Study-Vicon)

## **3.2 Funktionsweise**

Motion Capture beschreibt das digitale Aufzeichnen von Bewegungen. Dieses wird meist durch ein System, bestehend aus Sensoren und einer Software, gelöst. Auch wenn die Definition von Motion Capture nicht vorschreibt, dass es Lebewesen sein müssen, sind es doch hauptsächlich Menschen und manchmal Tiere, für die Motion Capture eingesetzt wird.

### **3.2.1 Begriffsklärung**

Performance Capture und Motion Capture werden häufig für dasselbe gehalten. Dies stimmt aber nur zum Teil. Während Motion Capture das Aufnehmen und Erfassen von Bewegungen des Körpers beschreibt, also nur die Arme, Beine sowie die des Torsos. Performance Capture beschreibt einen umfassenderen Ansatz, bei dem auch die kleineren Bewegungen eingefangen werden, zum Beispiel die der Hände oder des Gesichts. Während das Tracken der Hände in der Regel eine Erweiterung des Motion Capture Systems ist, wird für das Facial Tracking in der Regel ein externes System genutzt.

Für diesen Anwendungsbereich müssen die Systeme wesentlich feiner aufgelöst sein, wenn dies auch neben den großen Bewegungen von Armen und Beinen auch kleinere der Finger aufnehmen sollen. Ein Problem hierbei sind die Kosten für ein solches System.

### **3.2.2 Motion Capture Methoden**

In den letzten Jahren haben sich die verschiedenen Motion Capture Systeme sehr stark weiterentwickelt. Deren grundlegende Funktion lässt sich in zwei verschiedene Kategorien unterteilen: Optische Systeme und sensorbasierte Systeme. Beide haben verschiedene Vor- und Nachteile und sind für verschiedene Anwendungsbereiche entwickelt worden.

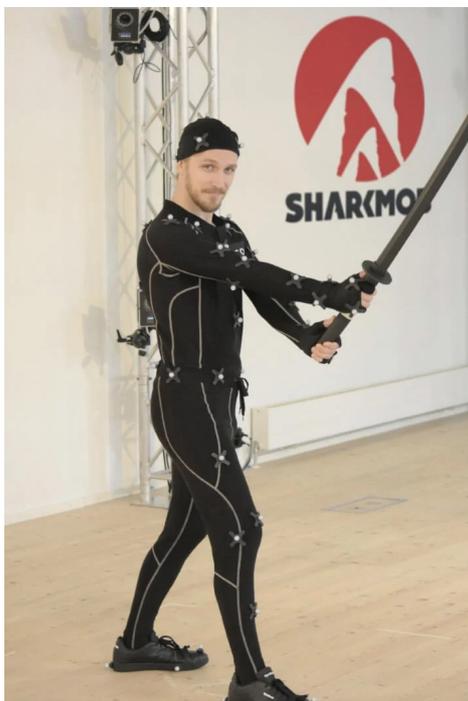
### 3.2.2.1 Optische Systeme

Optische Systeme erfassen die Bewegungsdaten mit der Hilfe von Kameras. Diese werden strategisch innerhalb des Studios aufgestellt. Dabei muss darauf geachtet werden, dass alle Winkel des Studios von mehreren Kameras gesehen werden. Für das Lokalisieren eines Punktes im Raum sind theoretisch drei Kameras notwendig. Jede weitere Kamera erhöht die Genauigkeit und damit auch die Qualität des Ergebnisses. Die Qualität ist dabei zusätzlich sehr stark von der Framezahl und der Auflösung der einzelnen Kameras abhängig. Damit die Bilder der verschiedenen Kameras erkannt und zusammengesetzt werden können, müssen die Position sowie die Eigenschaften der Kameras genau erfasst werden.

Optische Systeme werden in kontrollierten Umgebungen bevorzugt genutzt. Durch die Kameras hat die Produktion direkt Zugang zu Referenzvideos. Diese können von Animatoren zur Weiterverarbeitung oder zur Validierung von wissenschaftlichen Ergebnissen genutzt werden. Aufgrund der vielen Daten der Kameras müssen diese schnell vor Ort gespeichert und verarbeitet werden. Um dieses zu gewährleisten, ist ein Computer mit entsprechender Hardware notwendig.

### 3.2.2.1.1 Marker-basierte Systeme

Bei Marker-basierten Systemen werden von den Kameras die Marker erkannt und verfolgt. Diese Marker sind so an dem Objekt angebracht, dass sie gut für die Kameras zu sehen sind und auch die Bewegungen gut zu erkennen sind. Die Marker sind entweder stark reflektierend oder leuchten selbst meistens im infraroten Bereich. Dadurch kann die Software die Marker leichter im Raum erkennen und sie von der Umgebung unterscheiden. Für das Erkennen sind in der Regel spezielle Kameras notwendig. Außerdem müssen die Marker bei neuer Positionierung am Körper auch neu kalibriert werden. (Vicon 2022)



*Abb. 2: Motion Capture Suit der Firma Vicon*

### **3.2.2.1.2 Marker-lose Systeme**

Optische Systeme ohne Marker sind neu. Sie sind durch die Weiterentwicklung von künstlicher Intelligenz möglich geworden. Durch mehrere Kameras im Raum ist ein Programm in der Lage, Objekte im dreidimensionalen Raum zu lokalisieren. Diese Objekte können dann über die Zeit verfolgt werden. Die größte Herausforderung ist, anders als bei den auf Markern basierenden Systemen, keine eindeutig erkennbaren Objekte zu tracken. Deshalb ist bei dieser Arbeitsweise mehr Rechenkraft notwendig. Der Vorteil dieser Methode ist, dass keine extra Anzüge angezogen werden müssen, was diese Systeme besonders interessant für medizinische Anwendungen macht. (Vicon 2022)

### 3.2.2.2 Sensor-basierte Systeme

Bei Sensor-basierten Systemen kommt ein Anzug zum Einsatz, der Sensoren an den Gelenken hat, mit der die Rotation erkannt und gemessen werden kann. Die Sensoren werden als Inertial Measurement Units, kurz IMU bezeichnet. Sie erkennen die interne Beschleunigung durch einen Magneten. Durch mehrere strategisch am Körper platzierte Sensoren können die genauen Bewegungen der Gelenke erkannt und aufgezeichnet werden. Damit können die Bewegungsdaten für spätere Nachbearbeitungen aufgenommen werden.

Diese Anzüge werden gerne an Filmsets genutzt. Durch ihre Unabhängigkeit von externen Sensoren können sie am Set freier eingesetzt werden. Es besteht auch die Möglichkeit, die Anzüge unter die eigentlichen Kostüme zu ziehen. Die Bewegungsdaten bekommt man ohne zusätzliche Arbeit, durch das nachträgliche Animieren auf Basis des Filmfootage und ohne, dass sie im Bild stören.



*Abb. 3: Ein Sensor-basierte Motion Capture Suite der Firma Rokoko*

### 3.2.3 Facial Capture

Facial Capture beschreibt die Motion Capture Techniken, die eingesetzt werden, um die Bewegungen im Gesicht einzufangen. Die Schwierigkeit beim Aufnehmen von Bewegungsdaten des Gesichts ist, dass es sehr viele kleine und schnelle Expressionen gibt. Deshalb sind eine wesentlich höhere Abtastrate und Auflösung notwendig, als beim Aufnehmen des restlichen Körpers.

Zum Erfassen und Digitalisieren der Bewegungen wird eine Geometrie auf das Gesicht gelegt. Durch diese Geometrie werden die Verformungen erkannt und gespeichert. Die Bewegungen werden mit verschiedenen Versionen der Geometrie, sogenannten Blendshapes, abgeglichen. Auf der Basis der Blendshapes kann dann die Animation auf den neuen Charakter übertragen werden. Der Aufbau der Blendshapes basiert auf dem von Paul Eckman entwickelten FACS-System, das für die Kategorisierung von Gesichtsausdrücken entwickelt wurde und eigentlich in der Psychologie zum Einsatz kommt.



Abb. 4: Facial Capture System genutzt von ILM bei dem Film „Teenage Mutant Ninja Turtles“ (2014)



Abb.5: Bildschirmaufnahme des Facial Capture App der Firma Epic Games (Unreal Engine)

Beim Facial Capture kommen nur optische Systeme zum Einsatz, weil es nicht möglich ist, die vielen Muskeln im Gesicht mit entsprechenden Sensoren zu versehen. Auch hier wird sowohl auf Markern basierende als auch auf Marker-lose Systeme zurückgegriffen.

Marker-basierte Systeme nutzen Marker an bestimmten Punkten im Gesicht, die Rückschlüsse auf die Bewegung bestimmter Muskeln zulassen. Um die Bewegungen der Marker festzuhalten, werden eine oder zwei Kameras frontal vor dem Gesicht positioniert. Zwei Kameras haben den Vorteil, dass sie die Position des Markers im dreidimensionalen Raum erfassen. (Havaladar 2006)

Marker-lose Systeme benötigen in jedem Fall eine Methode, um eine dreidimensionale Repräsentation von dem Gesicht zu generieren, das heißt entweder mindestens zwei Kameras oder das Nutzen eines speziellen 3D-Scanners. Viele der Facial Capture Lösungen, die es für das iPhone gibt, basieren auf dem von Apple entwickelten „True Depth“, was im Grunde ein Structured-Light Sensor ist. Die Verformungen des durch den Scanprozess entstandenen Meshes, werden dann auf ein Basemesh übertragen, woraus die Blendshapes extrahiert werden. (Recording Facial Animation from an iOS Device 2023)

## 4 Virtuelle Kamera

### 4.1 Geschichte

Virtuelle Kameras wurden schon in vielen Filmen genutzt, um Computer-generierten Sequenzen eine realistische Kamerabewegung zu geben.

Die ersten Versuche, die Kamera im virtuellen Raum durch Tracken so bewegen zu lassen, wie sie es in der realen Welt tun würde, gab es mit dem Aufkommen des Motion Capture. Schon bei „Der Herr der Ringe“ wurden mithilfe des Motion Capture Systems Sensoren getrackt. Damit konnten die Bewegungen der Kamera am realen Set und die am Computer für die digitalen Charaktere und Environments optimal geplant werden. Bekannt wurde diese Arbeitsweise, als sie 2009 für den Film „Avatar“ von James Cameron genutzt wurde. Damit wurden die Aufnahmen in der außerirdischen Welt gefilmt. Es wurde auch im selben Jahr für die Zwischenszenen in dem Computerspiel „Resident Evil 5“ genutzt. Dieses System wurde von „Just Cause Production“ entwickelt und hieß „iKam“. Es war eines der ersten Systeme, die speziell für diesen Bereich entwickelt wurden. (Lowe 2009)

Seitdem wurde es bereits in verschiedenen Animationsfilmen genutzt wie zum Beispiel in „A Christmas Carol“. Ein weiterer Meilenstein war 2019 das Remake vom „Lion King“. Für die realistischen Kamerabewegungen wurden viele Techniken verwendet und zum Teil auch neu entwickelt. Dort wurden von Kamerakränen bis Dollys eine Vielzahl von Kamerahilfsmitteln genutzt, sei es durch Sensoren oder durch das Arbeiten mit digitalen Simulationen der Steuereinheiten. Sie haben für die Arbeit mit der Kamera die Animationen in eine Art Computerspiel integriert. Somit konnten die Verantwortlichen sich die Umgebung und die möglichen Kamerapositionen besser vorstellen. Dadurch wurde der Arbeitsablauf von einem Realfilm so nahe wie möglich simuliert. (Magnopus 2023)

## 4.2 Funktionsweise

Eine virtuelle Kamera beschreibt erst mal nur die Repräsentation einer realen Kamera im digitalen Raum. Im Bereich der Virtual Production beschreibt es das Aufnehmen der Bewegungen einer Kamera. Dafür können verschiedene Methoden zum Einsatz kommen, abhängig vom Endprodukt und den Voraussetzungen.

Die optische Methode ist die älteste und am weitesten verbreitete. In der Visual Effects Branche gehört es bis heute zu den Standardaufgaben, die Kamerabewegung auf Basis der Bilder zu tracken. Bei dieser Methode sind keine extra Apparaturen zum Aufnehmen der Kamerabewegungen notwendig. Allerdings ist dafür eine spezielle Software nötig. Durch diese Software lassen sich verschiedene Features im Footage über die Frames hinweg tracken. Durch die verschiedenen Positionen auf dem Bild kann sich die Software den Parallaxeeffekt zunutze machen. Dieser Effekt beschreibt die Eigenschaft, dass sich Objekte, die weiter weg sind relativ zu denen, die näher an der Kamera sind, langsamer bewegen. Damit lassen sich sowohl die Bewegung der Kamera im Nachhinein nachverfolgen, als auch die relative Position einzelner Elemente im Bild. Auch gibt es mittlerweile andere Systeme, die die Kamerabilder nutzen, um die Kamerabewegung in den digitalen Raum zu übertragen. Dies ist im Optimalfall eine exakte digitale Repräsentation der echten Kamera.

Die zweite Methode ist das direkte Tracken der Kamera durch einen Sensor oder eine Erweiterung des Motion Capture Systems. Dafür gibt es eine Reihe an Möglichkeiten diese Methode umzusetzen. Die wahrscheinlich kostengünstigste, ist die einen Sensor von einem handelsüblichen Virtual Reality Systems zu nutzen. Dies ist einfach einzustellen und sehr schnell arbeitsbereit. Weil diese Sensoren aber häufig nicht auf diese präzisen Bewegungen angepasst sind, braucht es jedoch etwas Nachbearbeitung. Die andere Methode ist es, Marker des Motion Capture System zu nutzen. Dies übernimmt dann häufig auch die Vor- und Nachteile des jeweiligen Systems, bringt aber noch den Vorteil mit sich, dass sich alle Marker im selben digitalen Raum befinden, mit demselben Scale und denselben Abmessungen. Was bedeutet, die Position der Kamera passt auch relativ zu den Charakteren im dreidimensionalen Raum. Diese Technik ist besonders gut für Einstellungen geeignet, die wie aus der Hand gefilmt wirken sollen.

Die letzte Methode ist die Steuereinheit verschiedener Kamerahalterungen zu imitieren. Dies kann von einfachen Stativköpfen bis hin zu komplexen Steuereinheiten von Dollys oder Kränen gehen. Dafür wird das Bedienungsgerät so angepasst, dass die Steuerbefehle erst in digitale Daten übersetzt und dann in eine Software gebracht werden, die in der Lage ist, eine virtuelle Kamera darzustellen. Damit die Immersion für den, der die Kamera bedient, möglichst gut ist, muss die Übertragung und die Darstellung des Kamerabildes möglichst schnell vonstattengehen. Dafür bietet sich in vielen Fällen eine Game Engine an. Alternativ gibt es auch spezielle Programme wie Autodesk Motionbuilder, die für diese Anwendungen genutzt werden.



*Abb.6:Marker basiertes Tracking System für eine Virtuelle Kamera*



*Abb.7:Sensor basiertes Tracking System für eine Virtuelle Kamera, basierend auf einem Sensor von einem VR-System*

## 5 Game Engine

### 5.1 Geschichte

Die Geschichte der Game Engines ist auch die Geschichte der Computerspiele. Das erste Computerspiel mit dreidimensionaler Grafik war 1987 das Spiel „Driller“. Es basierte auf der Freescape Engine von Incentive Engine. Fünf Jahre später kam dann mit der Doom Engine die erste Engine, die für die Darstellung der dreidimensionalen Geometrie Raycasting benutzte, eine Technik, die bis heute in einigen Computerspielen genutzt wird. Auch bei den meisten Viewports von Programmen zum Erstellen von 3D-Inhalten ist diese Technik Standard. Durch Raycasting, einer Vorstufe des Raytracings, war die Engine in der Lage, auch die Eigenschaften einer Kamera zu simulieren.

Für die Filmindustrie wurden die Game Engines 2018 interessant, als Nvidia Grafikkarten anfangen hardwareseitig Raytracing zu unterstützen, wodurch die Qualität der gerenderten Bilder noch einmal deutlich stieg. Kurz darauf kamen die ersten Spiele auf den Markt, die diese neue Technik unterstützen. Ab diesem Zeitpunkt konnten die Bilder, die in einer Game Engine gerendert wurden, mit denen aus einem Offline-Renderer mithalten.

Der gängigste Anwendungsbereich im Moment für Game Engines ist das Bespielen von LED-Walls für Film- und Fernsehproduktionen, wie sie bei der Serie „The Mandalorian“ eingesetzt wird. Dadurch konnten Computer-generierte realistische Hintergründe ohne Greenscreen hinter die Charaktere projiziert werden. Dies reduzierte den Aufwand in der Nachbearbeitung stark und es war möglich, stark reflektierende Requisiten zu nutzen, ohne die Probleme zu haben, dass sich das Grün aus der Umgebung in dem Objekt reflektiert. Was meistens dazu führte, dass sie entweder aus dem Skript gestrichen wurden oder nachträglich durch ein digitales Gegenstück ersetzt werden mussten.

## **5.2 Funktionsweise**

Eine Game Engine ist die Basis, auf der das Spiel aufbaut. Es bildet das Grundgerüst für die Programmierung, die Grafik und auch die Audioinhalte, die am Ende in dem Spiel zu erleben sind.

### **5.2.1 Was bedeutet Echtzeit**

Damit sich ein Spiel für den Spieler flüssig anfühlt, müssen die Bilder in Echtzeit berechnet werden. Das bedeutet, die vom Spieler eingegebenen Befehle müssen erkannt, in die Bewegungen im Spiel übersetzt und dargestellt werden. Das alles muss innerhalb von einem Sechzigstel einer Sekunde passieren. Ein Mensch ist ab ungefähr zwölf Bilder in der Sekunde nicht mehr in der Lage, einzelne Bilder zu erkennen. Allerdings wirken sie durch die fehlende Bewegungsunschärfe nicht flüssig. Erst ab ungefähr sechzig Bilder in der Sekunde wirken die Bilder ohne Bewegungsunschärfe auf den Menschen flüssig. Auch bei unserem Anwendungsbereich ist die schnelle Verarbeitung der Befehle notwendig, denn der Inputlag, so wird die Zeit genannt, die zwischen dem Eingeben des Befehls und dem Ergebnis liegt, kann die Arbeit zum Beispiel mit einer virtuellen Kamera unnötig erschweren.

### **5.2.2 Inhalte einer Game Engine**

Die meisten aktuellen Game Engines stellen dem Entwickler des Spieles eine Grundlage für alle relevanten Bestandteile des Spieles. Dies beinhaltet neben der Renderengine, die für das Berechnen der Bilder zuständig ist auch eine Audioengine, die es ermöglicht, Audioquellen im dreidimensionalen Raum zu positionieren und so wieder auszugeben, dass sie möglichst genau vom Spieler lokalisiert werden können. Im Weiteren werden Hilfestellungen gegeben für das Programmieren der NPCs (Non-Playable Characters) und andere relevante Elemente eines Videospieles. Da Spiele mit verschiedenen Formen von Animationen arbeiten, haben viele bereits eine Möglichkeit integriert zum Bearbeiten und Erstellen von Animationen für Charaktere und Props.

### 5.2.3 Wie funktioniert ein Raytracer

Wenn ein Bild mit einer normalen Kamera aufgenommen wird, werden in der Realität Lichtstrahlen von einer Lichtquelle in die Welt ausgesendet. Diese Lichtstrahlen werden dann von den verschiedenen Objekten in der Welt reflektiert. Wenn wir dann ein Foto mit unserer Kamera aufnehmen, werden die Strahlen, die in diesem Moment auf unseren Sensor treffen, in elektrische Impulse und dann weiter in digitale Daten übertragen. Wird ein Bild mit einer virtuellen Kamera in einem digitalen Raum gemacht, wäre diese Vorgehensweise zu ineffizient, denn zu viele Strahlen würden sinnlos berechnet werden, weil sie nie auf den Sensor trafen. Stattdessen wird die Realität umgedreht. Die Strahlen werden von dem Sensor aus ausgesendet. Wenn sie auf ein Objekt treffen, werden sie abhängig von dem Winkel reflektiert. Wenn dann der Strahl eine Lichtquelle trifft, wird der Pixel entsprechend beleuchtet. Die Farbwerte des Pixels sind dabei abhängig von den Eigenschaften der Shader, die das Objekt hat, von dem er reflektiert und wie häufig er reflektiert wurde.

Für den Einsatz in einer Game Engine wurde diese Arbeitsweise noch mal etwas vereinfacht, damit die Menge an zu berechnenden Strahlen auf ein Minimum reduziert werden kann. In der Regel kann während des Erstellens der Szene eingestellt werden, ob und in welcher Form ein Objekt durch den Raytracer beeinflusst werden soll. Zuerst sendet der Raytracer Strahlen aus, um zu testen, ob auf diesem Pixel Objekte zu sehen sind, die vom Raytracing beeinflusst werden. Wenn das nicht der Fall ist, wird der Pixel durch die alte Methode Rasterizing berechnet. Nur wenn auf dem Pixel ein Objekt zu sehen ist, das durch Raytracing berechnet werden soll, wird die Art zur Berechnung genutzt. (McGuire et al. 2019)

## 6 Stand der Technik

Virtual Production ist zwar noch neu, dennoch wurde es bereits in verschiedenen Bereichen eingesetzt. Welche Techniken und in welcher Weise eingesetzt werden, variiert dabei von Projekt zu Projekt. Am stärksten hat der Bereich der Realfilme die Techniken adaptiert, aber auch die Animationsbranche hat sich bereits einige Techniken zunutze gemacht.

### 6.1 Realfilm

„Avatar“ 2009 von James Cameron war der Film ab dem Virtual Production bekannt und immer breiter genutzt wurde. Im Laufe der Dreharbeiten wurden ein Großteil der bereits genannten Techniken benutzt. Für die Bewegungen der im Film dargestellten Alien Rasse, wurde Performance Capture genutzt. Auch wurden für viele der Aufnahmen ein Trackingsystem für eine virtuelle Kamera verwendet. (Virtual Production | Visual Effects + | Wētā FX 2023)



Abb.8: Bild von dem Set von „Avatar Aufbruch nach Pandora“. Im Hintergrund sieht man den Schauspieler im Motion Capture Anzug. Im Vordergrund ist der Kameramann mit einer Virtuellen Kamera zu sehen

Ein weiteres Einsatzgebiet für die Techniken ist das Bespielen von LED-Volumes. Die angezeigten Inhalte müssen in Echtzeit und im dreidimensionalen Raum berechnet werden. Dafür sind Game-Engines optimal. Damit die Bewegungen der realen Kamera mit der im digitalen Raum zusammenpassen, muss die Kamera durch einen Sensor direkt getrackt werden. Diese Technik verbreitet sich immer weiter als teurere, aber hochqualitativere Alternative zum klassischen Greenscreen Shoot. Eine der ersten Produktionen die diese Methode genutzt haben war die Serie „The Mandalorian“. Die Firma Industrial Light&Magic war eine der ersten die, auf Basis dieser Technik, ein System entwickelt hat. Vor der Nutzung von Game Engines wurden häufiger große Leinwände oder LED-Wände genutzt, um Reflektionen von virtuellen Welten auf realen Objekten zu simulieren. (DNEG 2019) Erst die neuen Rendertechnologien ermöglichen es, die Hintergründe in Echtzeit und in der notwendigen Qualität darzustellen. (Forging new paths for filmmakers on The Mandalorian 2023)



*Abb.9: Abbildung aus dem LED-Volume der Firma Industrial Light & Magic, hier genutzt für die Serie „The Mandalorian“ 2021.*

## 6.2 Animationsfilm

„Der Polarexpress“ von Robert Lee Zemeckis adaptierte schon im Jahr 2004 einige der Techniken. Die Produktion des Films hat den Begriff „Performance Capture“ geprägt, durch die parallele Nutzung von Facial Capture und Motion Capture. Dies ermöglichte dem Schauspieler Tom Hanks auch mehrere Charaktere innerhalb des Films zu spielen. Für die Animation der Kamera kam eine Art der Virtuellen Kamera zum Einsatz. Hierfür wurde hauptsächlich ein Handkurbel-System genutzt, aber auch ein Sensor für Szenen mit einer Handkamera. Dies ermöglichte dem Kameramann ein natürlicheres Gefühl beim Bearbeiten der Kamerabewegungen. (Imageworks 2005)



Abb.10: Handkurbel System genutzt um die Virtuelle Kamera zu steuern.



Abb.11: Behind the Scenes vom den Motion Capture Aufnahmen

Im Remake vom „König der Löwen“ wurde für die Nutzung der virtuellen Kamera extra ein Programm entwickelt, das einem Computerspiel ähnelt und auf der heute in der Virtual Production viel verwendeten „Unity Engine“ basierte. Mit diesem Programm konnten verschiedene Formen von virtuellen Kameras gesteuert werden, während in Echtzeit die Animationen abgespielt werden konnten. In dieser Umgebung konnten dann auch durch die Zuhilfenahme von Virtual-Reality Headsets, virtuelle Set Besichtigungen mit der Filmcrew durchgeführt werden. Für die Kameras wurde eine Reihe an Kamerasystemen digital simuliert. (Magnopus 2023) Die final aufgenommenen Sequenzen konnten direkt wiedergegeben und an einen Cutter geschickt werden. Des Weiteren konnten durch das Nutzen einer Game-Engine, Animationen sowie das Licht in Echtzeit während des Drehs angepasst werden. (The Lion King 2023)



*Abb.12: Kameramann Caleb Deschanel beim Steuern der Virtuellen Kamera am Set vom Lion King Remake (2019)*

## 6.3 Pre-Visualisierung

### 6.3.1 Was ist Pre-Visualisierung

In der Pre-Visualisierung, kurz Previs, wird der Film oder einzelne Szenen vor dem eigentlichen Dreh animiert. Durch diesen Arbeitsschritt können Fragen beantwortet werden, welche die Planung der eigentlichen Produktion erleichtern. Ursprünglich im Realfilm genutzt, wird Previs heute auch im Bereich der Werbung und für Videospiele eingesetzt. Eine Pre-Visualisierung ist in der Regel eine reduzierte Animation in einer groben Repräsentation der finalen Umgebung.

Diese Art der Vorplanung kann viele Fragen klären. Wie viel vom Set ist im finalen Film zu sehen und wie viel muss am Ende gebaut werden. Auch Designentscheidungen können durch dieses Hilfsmittel geklärt werden. Die technischen Probleme sind die, die am Ende am teuersten werden und müssen daher rechtzeitig gelöst werden. Die finale Previs wird dann mit allen in der Produktion relevanten Abteilungen geteilt.



Abb.13: Bild aus der Previs (Links) und das finale Bild (Rechts). Beispiel aus dem Film „The Jungle Book“ (2016)

Viele der beschriebenen Techniken wurden früh in der Pre-Visualisierung eingesetzt. Um Zeit während der Animation zu sparen, werden häufig Motion Capture Systeme verwendet. Hier besteht die Möglichkeit vom Regisseur und anderen Verantwortlichen, die Bewegungen des Charakters direkt zu beeinflussen. Dies wird vor allem bei Bewegungen interessant, die eine besondere Relevanz innerhalb des Filmes haben. Virtuelle Kamera Systeme werden genutzt, um Kameraeinstellungen vor auszuplanen, Perspektiven zu testen und Wirkungen zu überprüfen. Häufig werden die Einstellungen und Bewegungen anschließend mit dem Visual Effects Team geteilt, um sie zum Beispiel bei Sequenzen, die vollständig digital sind, zu nutzen.

Auch Game Engines werden mittlerweile genutzt. Durch die hohe Qualität, die in Echtzeit erreicht werden kann, ist auch die Qualität der Previs gestiegen. Denn durch die Nutzung eines Raytracers kann nun auch die Nutzung und der Effekt von Licht stärker mit in die Planung integriert werden.

## 7 Unterschiede bei dem Workflow

### 7.1 Unterschiede zwischen Motion Capture und Keyframe Animation

Ein großer Vorteil vom Arbeiten mit Motion Capture ist die Möglichkeit der Kommunikation. Ein Regisseur oder Animation-Supervisor kann die Animation in Echtzeit beeinflussen. Wenn man mit Keyframe Animationen arbeitet, kann eine solche Iteration abhängig von der Komplexität mehrere Tage bis Wochen in Anspruch nehmen. So kann das Animationsstudio durch das Arbeiten mit Motion Capture Zeit und dadurch, dass die Iterationszyklen verkürzt werden, auch sehr viel Geld sparen. Was für die Verantwortlichen in dem Studio selbst gilt, gilt natürlich auch für den Kunden, der gegebenenfalls den Filmauftrag gegeben hat. Auch sie haben die Möglichkeit, direkt und schnell in Echtzeit, Einfluss auf die Animationen zu nehmen.

Aber der Workflow wird jedoch durch die erhöhte Nutzung von automatischen Lösungen wesentlich technischer, denn das Motion Capture System muss durch entsprechend ausgebildetes und geschultes Personal bedient werden. Dies kann zum Teil die kreative Arbeit einschränken.

Auch wenn das Motion Capture die Animationen direkt durch die Bewegungen der Schauspieler erstellt, heißt das nicht, dass es nicht nachträglich bearbeitet werden muss. Da das Motion Capture System auf jeden Frame einen Keyframe erstellt, ist die Menge an Keyframes sehr hoch und unübersichtlich. Als Beispielrechnung ein Schultergelenk, das mit 50 Frames pro Sekunde über einen Zeittraum von zehn Sekunden aufgenommen wurde:

$$50(\text{Fps}) \times 10(\text{Länge in Sekunden}) \times 3(\text{Anzahl der Achsen}) = 1500 \text{ Keyframes}$$

Diese 1500 Keyframes führen zu einigen Problemen. Zum einen kommt zu den Keyframes hinzu, dass das Motion Capture System im Berechnen der Positionen gewisse Toleranzen hat. Diese entstehen bei Sensor-basierten Systemen durch die Abtastrate der Sensoren, bei optischen Systemen durch die Auflösung oder die Menge der Kameras, die das Objekt sehen. Diese Toleranzen führen zu Unregelmäßigkeiten innerhalb der Animation. Unregelmäßigkeiten oder auch Rauschen genannt, werden meistens durch ein feines und hochfrequentes Zittern an dem getrackten und animierten Objekt sichtbar. Dies muss in der Nachbearbeitung herausgefiltert oder manuell entfernt werden.

Zum anderen müssen auch die Posen aus den Rohdaten angepasst werden. Entweder weil es von dem System fehlerhaft aufgenommen wurde oder weil die aufgenommenen Posen an den Stil des finalen Produktes angepasst werden müssen. Diese Arbeit muss immer noch von Animatoren erledigt werden. Zwar können Algorithmen und klassische Filter größtenteils und auch recht erfolgreich gegen das Rauschen eingesetzt werden, aber das nachträgliche Bearbeiten der eigentlichen Posen muss noch von Menschen erledigt werden. Auch hier gibt es mittlerweile Ansätze, diese Arbeit durch das Nutzen künstlicher Intelligenz zu reduzieren. Dies steckt bis heute noch in den Kinderschuhen. (Mo et al. 2021)

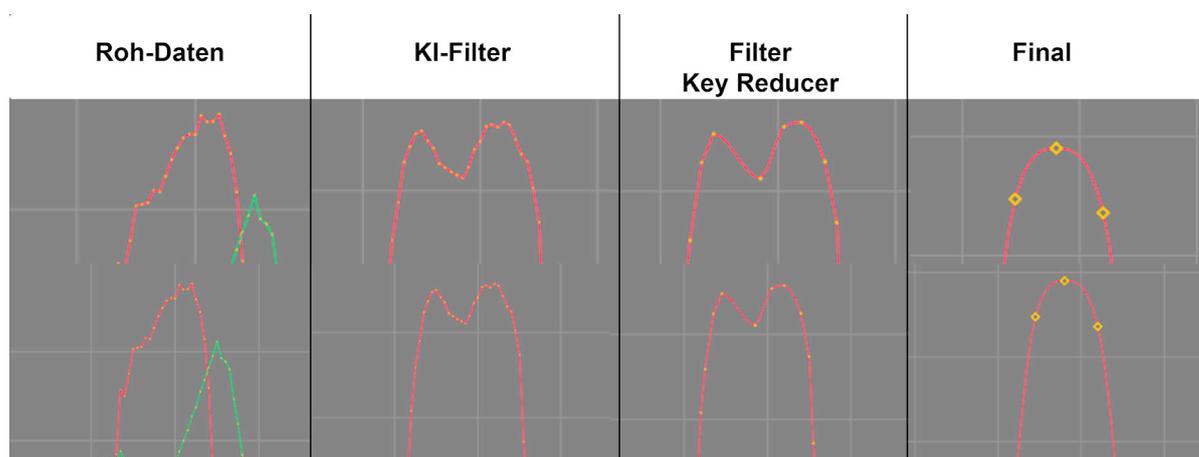


Abb.14: Animations-Kurven nach den verschiedenen Stadien der Nachbearbeitung

Ein großer Nachteil von Motion Capture ist, dass durch die Aufnahme von realen Schauspielern auch die Animationen sehr realistisch sind. Dies ist in vielen Animationsfilmen nicht gewollt, deshalb werden die MoCap-Daten in der Animationsbranche häufig nur als Referenz genutzt. Alternativ kann versucht werden, die Aufnahmen so zu bearbeiten, dass sie in den Stil des finalen Produktes passen. Beides kann den Workflow massiv verlangsamen und sollte vor der Produktion beachtet und abgewägt werden.

Facial Capture wird häufig für Zwischenszenen genutzt, in denen das Gesicht nicht das Hauptmotiv der Szene ist. Dort kann durch das Nutzen von Facial Capture sehr viel Zeit gespart werden. Das Animieren der Gesichter ist einer der zeitintensivsten Abschnitte der Charakter-Animation. Vor allem wenn das Gesicht nicht besonders relevant ist, kann durch das Facial Capture schnell das Gesicht lebendiger wirken, ohne viel Zeit in diese Szene zu investieren. (Sagar 2006)

## 7.2 Unterschiede zwischen virtueller Kamera und Keyframe Animierter Kamera

Genau wie bei der Nutzung von Motion Capture ist ein großer Vorteil die Kommunikation zwischen den Verantwortlichen. Die Animation kann auch von Kameras viel Zeit in Anspruch nehmen, vor allem wenn man von dem eigentlichen Plan aus dem Storyboard abweichen will. Wenn man sich für die Arbeit mit einer virtuellen Kamera entscheidet, ist in der Vorbereitung einiges zu beachten. Es muss schon ein Großteil der digitalen Umgebung bereits erstellt worden sein. Sonst kann es problematisch werden, die finalen Kameraeinstellungen zu finden. Dies stellt besondere Herausforderungen an die Verantwortlichen für die Environments.

Beim klassischen Workflow eines Animationsfilms wird deshalb nicht häufig vom Storyboard abgewichen, um diese Art von Mehraufwand zu reduzieren. Durch das Nutzen einer Virtuellen Kamera kann die Kontrolle der Kamera wieder an die verantwortlichen Kreativen zurückgegeben werden. Ansonsten waren sie dazu gezwungen, ihre Visionen an die entsprechenden Techniker und Kreativen zu kommunizieren, die dann ihre Interpretation umsetzen.

Für das Nutzen von einer Virtuellen Kamera muss vor der eigentlichen Produktion beachtet werden, ob das Budget es zulässt. Da eine virtuelle Kamera eine reale Kamera imitieren soll, müssen die entsprechenden Hilfsmittel vor dem Dreh benutzt werden. Das beinhaltet zum einen die Sensoren und gegebenenfalls auch Dollys, Kamerakräne und anderes Zubehör, dies bedeutet auch, dass während des Drehs für entsprechendes Personal gesorgt werden muss. Wenn das Personal nicht bereits mit einer virtuellen Kamera gearbeitet hat, muss auch noch Zeit investiert werden, das Team einzuarbeiten, zum Beispiel durch einen Workshop, um die Grundlagen zu vermitteln.

Ein Problem ist in vielen Fällen allerdings die Fokussierung auf das Replizieren von realen Kamerafunktionen. Dadurch wird es in vielen Bereichen unmöglich, nicht realistische Kameraeinstellungen zu drehen, weshalb dort klassische Keyframe-Animationen zum Einsatz kommen müssen. Eine mögliche Lösung ist der Weg mit einer klassischen Fernbedienung, die auch für Drohnen genutzt wird. Drohnen haben es in vielen Bereichen ermöglicht Einstellungen zu drehen, die sonst unmöglich wirkten. Dies kann dann als etwas intuitivere Alternative zur klassischen Animation in den Workflow integriert werden.

### **7.3 Unterschiede zwischen dem Arbeiten mit einer Game-Engine und einem klassischen 3D-Programmes**

Um die Vergleichbarkeit zu erhöhen, werden die verschiedenen Bereiche an konkreten Beispielen verglichen und die Unterschiede aufgezeigt.

„Unreal Engine 5“ steht als Beispiel für die Game Engine, weil es aktuell eine der modernsten Game Engines auf dem Markt ist. Gleichzeitig wird sie von dem Entwickler kostenlos zur Verfügung gestellt. Des Weiteren wurde die Unreal Engine bereits in der Branche für Experimente und Aufgaben im Bereich Virtual Production eingesetzt.

Als Beispiel für ein 3D Programm wird für diese Arbeit „Autodesk Maya“ genutzt. Maya ist seit vielen Jahren Industriestandard. Es wurde schon 2001 bei „Der Herr der Ringe“-Filmen genutzt, die einen wichtigen Meilenstein in der Virtual Production und dem Motion Capture darstellen. Einzige Ausnahme ist im Bereich visuelle Effekte. Dort werde ich mich primär auf das Programm „SideFX Houdini“ konzentrieren. Auch wenn ich mich in den folgenden Bereichen sehr stark auf diese Programme konzentriere, sind die meisten Punkte und beleuchteten Aspekte in dieser Arbeit auch auf andere 3D-Programme und Game Engines mit ähnlichem Funktionsumfang übertragbar.

## 7.3.1 Rendering

### 7.3.1.1 Wie funktioniert ein Pathtracer

Um die beiden Renderer miteinander vergleichen zu können, ist es notwendig zu verstehen, wie ein normaler Offline-Renderer arbeitet.

Ein typischer Offline-Renderer wird auch offiziell Pathtracer genannt. Der dahinter liegende Algorithmus ist in beiden Fällen sehr ähnlich. Allerdings ist der Algorithmus bei einem Pathtracer etwas komplexer gestaltet. Der offensichtlichste Unterschied ist die Menge an Strahlen, die pro Pixel berechnet werden. Die Anzahl von Strahlen werden als Samples bezeichnet. Während bei einem Raytracer meistens nur einer bis fünf Strahlen pro Pixel berechnet werden, sind es bei einem Pathtracer durchaus mehrere Hundert pro Frame. Auch kann das System besser an die jeweilige Szene angepasst werden, indem die Menge der verschiedenen Strahlen genau einstellt. Mehr Reflexionen oder das Berechnen von Subsurfaces brauchen mehr Samples, wodurch der Computer wieder mehr Zeit für die Berechnung braucht.

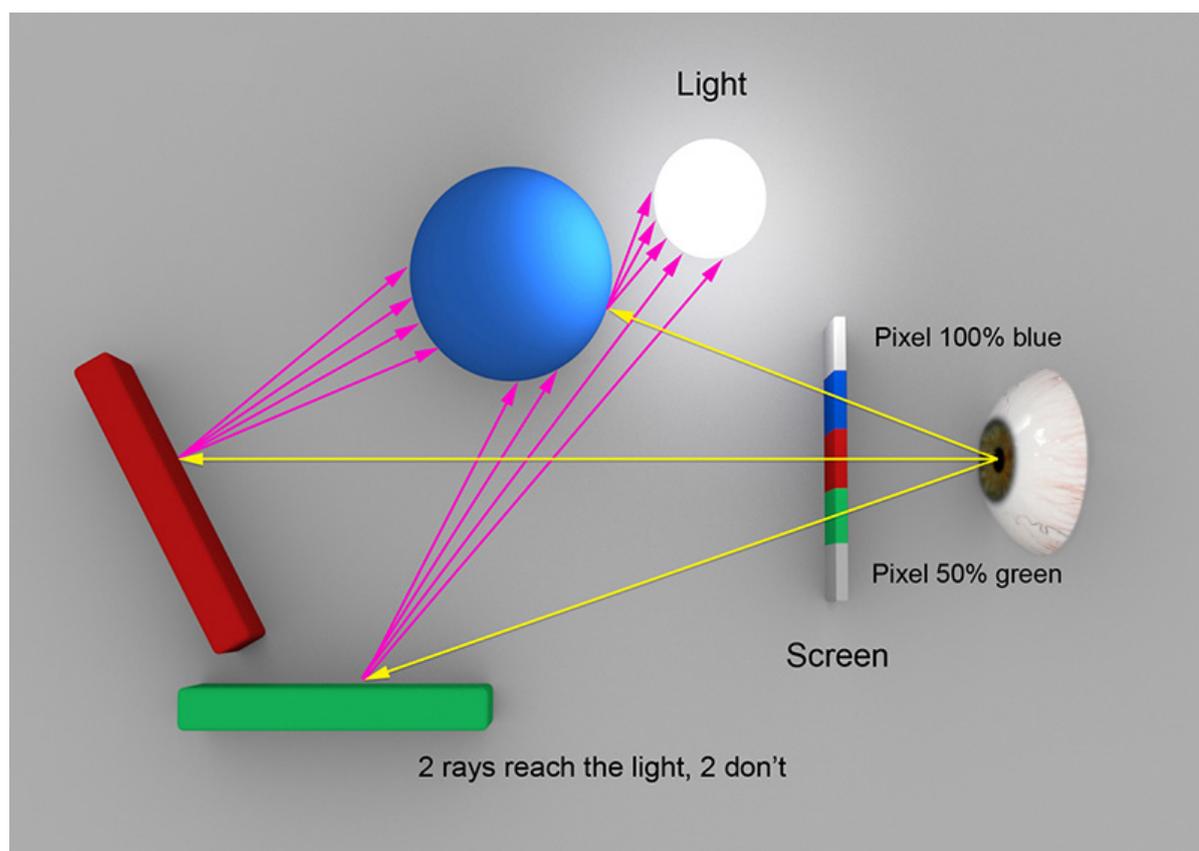


Abb.15: Skizze Funktionsweise eines Pathtracers

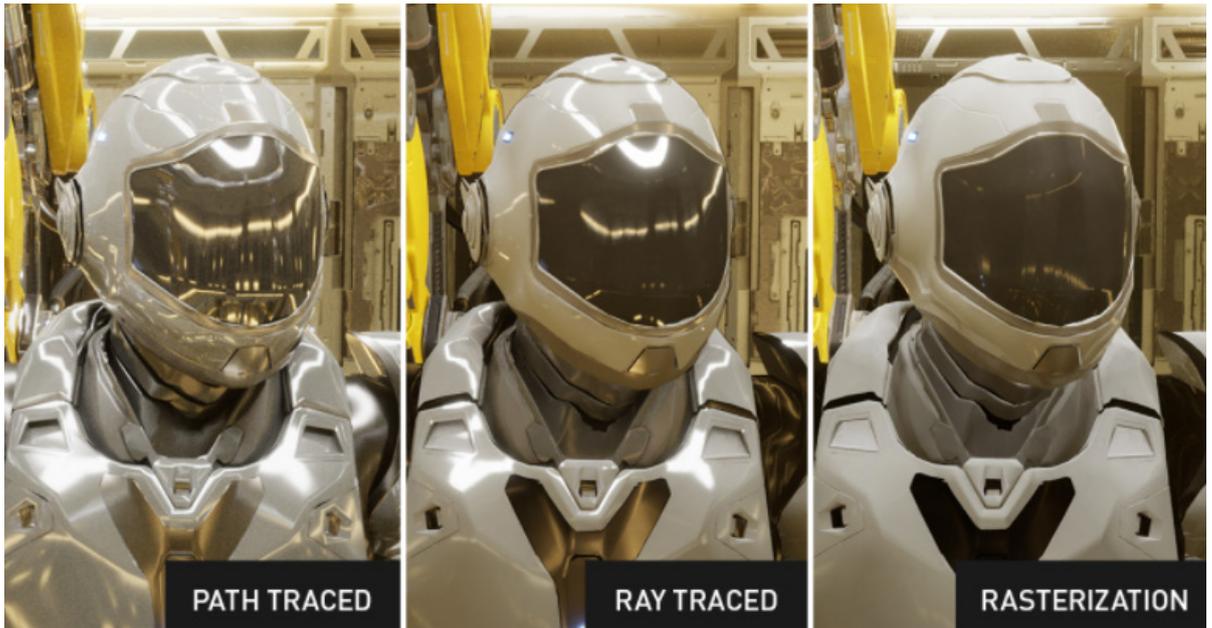


Abb.16: Qualitätsunterschiede zwischen einem Pathtracer, einem Raytracer und einem durch Rasterizing gerenderten Bildes

Dies ermöglicht auch komplexere physikalische Phänomene zu berechnen wie zum Beispiel Subsurface Scattering. Subsurface Scattering beschreibt die Eigenschaft bestimmter Materialien, in denen Teile der Strahlen in die Oberfläche eindringen und dort hin und her reflektiert werden. Ein bekanntes Material mit dieser Eigenschaft ist die menschliche Haut. Auch komplexere Szenen mit viel indirektem Licht können durch die erhöhte Zahl an Samples profitieren.

Auch das Darstellen von Kaustiken ist mit einem Pathtracer möglich. Entweder durch das Simulieren durch das sogenannte Photon-Mapping oder durch das Nutzen eines „Bidirektional Pathtracers“. Bei dem Photon-Mapping werden vor dem eigentlichen Renderprozess eine bestimmte Menge an Partikeln von der Lichtquelle aus in die Szene geschossen. Auf Basis dieser Karte beginnt der Pathtracer seine Arbeit. Beim bidirektionalen Pathtracer werden die Strahlen von beiden Seiten losgeschickt. Bei Annäherung werden dann die beiden Strahlen zu einem verbunden. Deshalb nennt man bidirektionale Pathtracer auch VCM (Vertex Connecting and Merging). Beide Herangehensweisen führen zu einem sehr ähnlichen Ergebnis, Photon-Mapping ist jedoch in der Regel schneller. Bei komplexeren Szenen kann aber ein bidirektionaler Pathtracer bessere und hochqualitativere Ergebnisse erzielen.

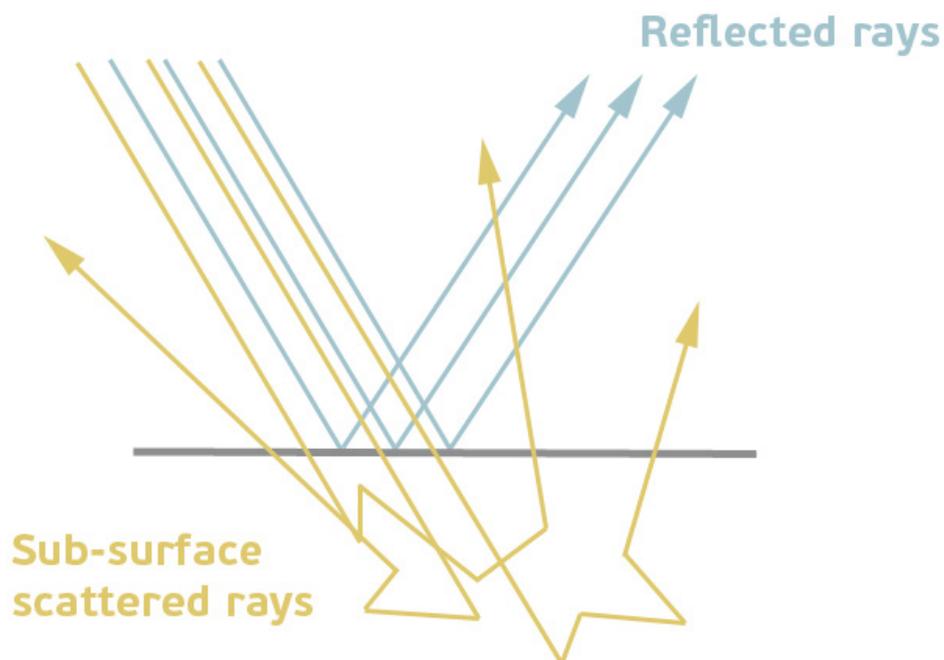


Abb.17: Skizze zur Darstellung der Berechnung von Subsurface Scattering

### 7.3.1.2 Denoiser

Um die Bilder mit dem Raytracer in Echtzeit anzeigen zu können, müssen die Bilder innerhalb von 16 Millisekunden bei Spielen und bis zu 40 Millisekunden bei Filmen berechnet werden. Dies ist allerdings mit der aktuellen Hardware kaum möglich, beziehungsweise sehr ineffizient. Deshalb kommt, nachdem die Strahlen den Großteil der Pixelwerte berechnet haben, ein Denoiser zum Einsatz. Dieses Programm versucht, mithilfe verschiedener Variablen, das Rauschen zu reduzieren, das durch die geringe Anzahl an Samples hervorgerufen wird.

Während viele der Denoiser auf mathematischen Gleichungen und klassischen Bildverarbeitungsalgorithmen basieren, wie zum Beispiel dem Medianfilter oder einem Weichzeichnerfilter wie der Gaußscher Weichzeichner. Diese folgen auf der einen oder anderen Weise derselben Herangehensweise. Die Filter verrechnen den Pixelwert des bestehenden Pixels mit denen der nebenanliegenden Pixeln. Dadurch werden die nebeneinanderliegenden Werte der Pixel einander angeglichen, was zu einem weicheren und gleichmäßigeren Bild führt.

Seitdem die Hardware schnell genug geworden ist und die Forschung im Bereich der künstlichen Intelligenz sich in den letzten Jahren weiterentwickelt hat, wird jetzt auch künstliche Intelligenz genutzt, um das Rauschen aus den Bildern zu entfernen. Diese wurden meistens durch eine Reihe an verschiedenen Bildern trainiert. Die Bilder wurden in verschiedenen Qualitätsstufen mit dem dementsprechenden Renderer gerendert. Aus diesen Beispielen kann die künstliche Intelligenz lernen, wie bei dem Render-Algorithmus das Rauschen entsteht. Dies kann dann bei Bildern eingesetzt werden, die nicht so viele Samples eingestellt haben. So kann die Renderzeit reduziert werden. (Martin-Karl Lefrançois 2017)

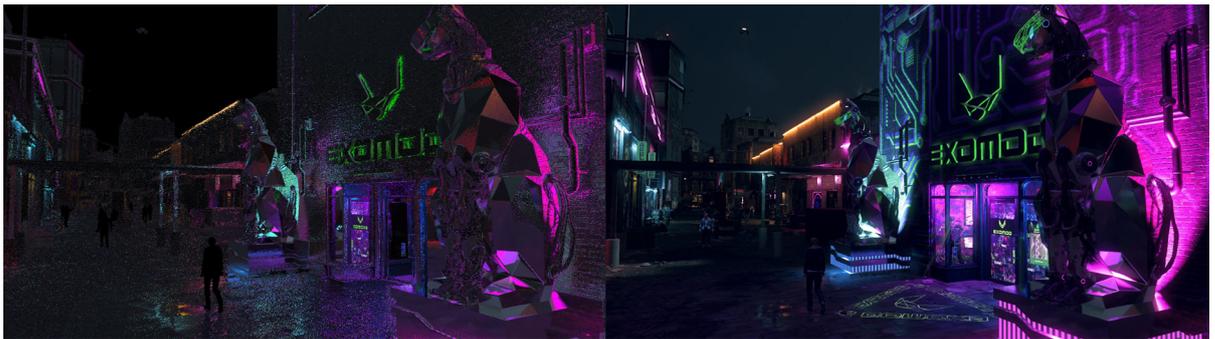


Abb. 18: Vorher/Nachher eines mit einem Raytracer gerenderten Bildes, anschließend mit einem Echtzeit-Denoiser bearbeitet. Szene aus dem Spiel Cyberpunk 2077 (CD Projekt Red).

### 7.3.2 Layout

Der Layoutprozess profitiert am meisten durch die Arbeit mit einer Game-Engine. Der Vorteil ist, dass man während des Arbeitens immer das finale Bild vor Augen hat. Was den Arbeitsprozess sehr erleichtert, besonders wenn viele kleine Details bearbeitet werden müssen.

Normalerweise muss beim Bearbeiten auf den „In Production Renderer“ gesetzt werden. Das ist eine reduzierte Variante des normalen Renderers, der für das finale Bild verwendet wird. Meistens werden rechenaufwendige Elemente deaktiviert sowie die Auflösung reduziert. Des Weiteren wird die Menge an Strahlen auf das notwendigste Minimum reduziert. Auch wenn der schneller ist, als das finale Bild, kann das Berechnen des Bildes mehrere Minuten in Spruch nehmen, was die Arbeitsgeschwindigkeit reduziert und die gesamte Arbeit weniger intuitiv macht.

Der Echtzeitanpruch kann auch hier schnell zu einem Problem werden. Die Komplexität der Szene ist stark abhängig von der Grafikkarte. Genauer die Menge an VRAM oder Videospeicher genannt, ist der limitierende Faktor. Dieser beeinflusst sowohl die Menge an Polygonen, die gleichzeitig in dem Bild angezeigt werden als auch die Menge und Auflösung der Texturen. Es gibt aber einige Methoden diese Limitationen zu umgehen.

Um das Problem für das Polygonlimit anzugehen, gibt es eine Reihe an Lösungen. Die am weitesten verbreitete sind LODs (Level of Detail). Dabei wird das Modell in verschiedene Versionen aufgeteilt. Diese Versionen haben unterschiedlich viele Polygone. Die verlorenen Details werden in der Regel in die Displacement und Normal Map übertragen. Die verschiedenen Versionen werden dann abhängig von verschiedenen Variablen und Parameter abgerufen, um die Performance zu verbessern. Die optischen Parameter sind zum einen die Entfernung zur Kamera und die Menge an Pixeln, die das Objekt auf dem Bild einnimmt. Zum anderen, wie schon erwähnt, limitieren die technischen Eigenschaften der Grafikkarte, deren Videospeicher in der Regel kleiner ist als der normale Arbeitsspeicher des Computers.

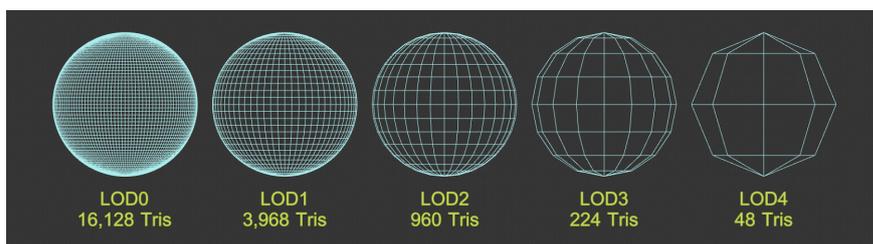


Abb. 19: LODs sichtbar an einer Kugel

Epic Games hat für die neueste Version ihrer Engine, der Unreal Engine einen neuen Ansatz entwickelt. Sie nennen das System Nanite. Es ist ein System zur Virtualisierung von polygonalen Meshes. Dadurch kann die Menge der Polygone dynamisch angepasst werden, ohne dass die visuelle Qualität eingeschränkt wird.

Ein ähnliches System existiert bereits für die Texturen, damit nicht immer die Bilddateien in der vollen Auflösung in den Videospeicher geladen werden müssen. Die notwendige Auflösung wird dabei abhängig von der Auflösung auf den finalen Bildern berechnet.

## 7.4 Shading

Gerade beim Shading sind die Einschränkungen von der Render Engine besonders stark zu beachten. Deshalb sind gerade hier die kreativen Unterschiede zum Arbeiten mit einem Offline-Renderer besonders sichtbar. Die Menge an einstellbaren Parametern sind aufgrund des vereinfachten Renderalgorithmus reduziert.

Diese Einschränkungen sind vor allem bei Materialien zu erkennen mit komplexen physikalischen Eigenschaften. Dort müssen dann von dem verantwortlichen Artist entsprechende Lösungen gefunden werden. Die Lösungen sind dabei sehr stark von der Renderengine und dem Problem abhängig. Sie erfordern in der Regel ein hohes Maß an kreativem und technischem Verständnis. Es gibt viele Materialien und physikalische Eigenschaften, die häufig zu Problemen führen. Dies sind beispielsweise alle Materialien, die Transparenzen aufweisen, also Glas oder auch Haut, weil dort Licht durch das Objekt durchscheint.

Weil die Qualität nach Möglichkeit nicht unter der Entscheidung leiden soll, muss dies bei den vorausgehenden Planungen beachtet werden. In vielen Bereichen können mit Hilfe des Shaders alternative Lösungen gefunden werden. Dadurch kann der Effekt simuliert werden, ohne ihn aufwendig zu berechnen. Gute Beispiele dafür sind Glas oder auch Haut.

## 7.5 Visuelle Effekte

Der Workflow beim Arbeiten mit visuellen Effekten in einer Game-Engine, auch für Filme gleicht eher dem für Videospiele als dem für Filme. Vor allem ist hier auch wieder der Renderer der limitierende Faktor. Visuelle Effekte benötigen verschiedene Varianten von Simulationen. Diese haben alle verschiedene Äquivalente in der für die Echtzeit Anwendungen.

Der Bereich, der am schwierigsten in einer Game-Engine umzusetzen ist, sind Volume Simulationen. Game-Engines sind in der Regel nicht in der Lage, Volumes zu rendern. Aufgrund der komplexen zu Grunde liegenden Berechnungen ist es schwierig diese in Echtzeit durchzuführen. Deshalb werden dort, so weit wie möglich, Sprites eingesetzt. Dies sind kleine, gering aufgelöste Bildelemente, die auf Partikel kopiert werden. Diese sollen die Dichte und Streuung von Rauch in der Umgebung simulieren. Sprites sind allerdings schwer zu kontrollieren und führen schnell zu einem unruhigen Bild und Gesamteindruck.

In vielen Bereichen wird hier auf Simulation direkt in der Game-Engine verzichtet. Stattdessen werden die Simulationen werden im Vorhinein in einem spezialisierten Programm bearbeitet. Industriestandard, sowohl für die Games als auch für die Filmindustrie, ist SideFX Houdini. Die finalen Simulationen werden als Geometrycache exportiert und können von der Game Engine gelesen und wiedergegeben werden.

## 8 Fazit

Virtual Production hat in vielen Bereichen die Animations- und Visual Effects Pipeline bereits revolutioniert. Auch wenn sie nur selten vollständig und ohne Ausnahmen genutzt werden, profitieren schon heute sehr viele Produktionen von den neuen Möglichkeiten.

Am meisten nutzt die Visual Effects Branche von den neuen Techniken. Hauptsächlich liegt es daran, dass sie primär mit realen Filmaufnahmen arbeiten. Diese Bildsprache muss auch auf die vollständig am Computer generierten Bilder übertragen werden. Dort werden sie schon jetzt häufig eingesetzt. Durch die immer günstiger werdenden Lösungen werden sie auch immer mehr zum Standard in der Industrie.

In der Animation hat die Virtual Production einen etwas schwierigeren Stand. Animationsfilme leben sehr stark von den stilisierten Bildern und Bewegungen. Diese sind nicht immer mit den realistischen Ergebnissen von Motion Capture und Virtuellen Kameras vereinbar. Dennoch kann die Branche sehr stark durch die vereinfachte Kommunikation profitieren. Das Nutzen einer realen Kamera in einem Studio, auch wenn die Welten und Charaktere digital sind, hilft besonders Unerfahrenen in diesen Bereich einzusteigen.

Der klassische Animationsfilm wird sich durch die Virtual Production in der Zukunft auf jeden Fall weiterentwickeln. Es werden vermutlich wesentlich mehr Produktionen auf die schnelleren Rendertechnologien zurückgreifen. Die klassische Keyframe-Animation, sowohl von Charakteren als auch von Kameras, wird durch die neuen Möglichkeiten nicht vollständig ersetzt werden können. Diese Befürchtung gab es bereits Anfang der 2000er, als mit „Der Herr der Ringe“ und dem Film „Der Polarexpress“ Motion Capture aufkam. Diese Techniken werden im Animationsbereich hauptsächlich unterstützend eingesetzt und ersetzen nur wenige Arbeitsschritte vollständig. Vor allem ist zu beobachten, dass die Echtzeit-Render Lösungen immer breitere Anwendung in der Branche finden.

Es ist allerdings vorstellbar, dass in Zukunft mehr Menschen, die heute im Realfilm arbeiten, stärker in die Produktion von Animationsfilmen integriert werden. Neben den neuen Arbeitsbereichen die sich entwickeln und positive Aussichten auf die Arbeitswelt geben, können auch viele Menschen, die jetzt noch in anderen Bereichen arbeiten, von dieser Entwicklung profitieren. Berufssparten, die davon betroffen sein werden, sind unter anderem Kameraleute und Schauspieler.

Diese Arbeit konnte nur einen Überblick über die Techniken und ihre Möglichkeiten beziehungsweise deren Grenzen geben. Deshalb gibt es immer noch viele kreative und technische Fragen, die es in der Zukunft wert sind, genauer betrachtet zu werden. Besonders die verschiedenen Techniken im Detail. Auch die Frage, wie mit den Limitationen der neuen Methoden umgegangen oder sie reduziert werden können, ist in der Zukunft sehr interessant zu verfolgen. Wie können die Aufnahmen zum Beispiel, von einem Motion Capture System am besten stilisiert werden und gegebenenfalls auch automatisiert und für Artists kontrollierbar gemacht werden?

Alles in allem wird Virtual Production einen immer wichtigeren Teil in der Film- und TV-Branche einnehmen. Die Entwicklung wird in der Zukunft einen immer größeren Anteil bei Produktionen haben. Nicht zuletzt durch den immer größer werdenden Serienmarkt muss in der Zukunft stärker auf effizientere Methoden zugegriffen werden.

## Literaturverzeichnis

- DNEG (2019): First Man - DNEG. Online verfügbar unter <https://www.dneg.com/show/first-man/>, zuletzt aktualisiert am 28.06.2019, zuletzt geprüft am 28.01.2023.
- Forging new paths for filmmakers on The Mandalorian (2023). Online verfügbar unter <https://www.unrealengine.com/en-US/blog/forging-new-paths-for-filmmakers-on-the-mandalorian>, zuletzt aktualisiert am 29.01.2023, zuletzt geprüft am 29.01.2023.
- Gray, Ali (2014): A Brief History of Motion-Capture in the Movies - IGN. In: IGN, 11.07.2014. Online verfügbar unter <https://www.ign.com/articles/2014/07/11/a-brief-history-of-motion-capture-in-the-movies>, zuletzt geprüft am 31.12.2022.
- Havaladar, Parag (2006): Sony Pictures Imageworks. In: John Finnegan (Hg.): ACM SIGGRAPH 2006 Research posters. ACM SIGGRAPH 2006 Courses. Boston, Massachusetts, 7/30/2006 - 8/3/2006. Association for Computing Machinery. New York, NY: ACM (ACM Conferences), S. 5.
- Imageworks (2005): from Mocap to Movie The Polar Express.
- Lowe, Scott (2009): The Tech Behind RE5 - IGN. In: IGN, 28.02.2009. Online verfügbar unter <https://www.ign.com/articles/2009/02/28/the-tech-behind-re5>, zuletzt geprüft am 26.01.2023.
- Magnopus (2023): Lion King | Magnopus. Online verfügbar unter <https://www.magnopus.com/lion-king>, zuletzt aktualisiert am 27.01.2023, zuletzt geprüft am 27.01.2023.
- Martin-Karl Lefrançois (2017): AI Denoiser: Enhance Rendering Quality. Online verfügbar unter <https://on-demand.gputechconf.com/siggraph/2017/video/sig1754-martin-karl-lefrancois-train-your-own-denoiser.html>, zuletzt aktualisiert am 04.08.2017, zuletzt geprüft am 01.01.2023.
- McGuire, Morgan; Shirley, Peter; Wyman, Chris (2019): Introduction to real-time ray tracing. In: ACM SIGGRAPH 2019 Courses. SIGGRAPH ,19: Special Interest Group on Computer Graphics and Interactive Techniques Conference. Los Angeles California, 28 07 2019 28 07 2019. ACM Special Interest Group on Computer Graphics. New York,NY,United States: Association for Computing Machinery (ACM Digital Library), S. 1–155.
- Mo, Clinton; Hu, Kun; Mei, Shaohui; Chen, Zebin; Wang, Zhiyong (2021): Keyframe Extraction from Motion Capture Sequences with Graph based Deep Reinforcement Learning. In: Heng Tao Shen, Yueting Zhuang, John R. Smith, Yang Yang, Pablo Cesar, Florian Metze und Balakrishnan Prabhakaran (Hg.): Proceedings of the 29th ACM International Conference on Multimedia. MM ,21: ACM Multimedia Conference. Virtual Event China, 20 10 2021 24 10 2021. New York, NY, USA: ACM, S. 5194–5202.
- Recording Facial Animation from an iOS Device (2023). Online verfügbar unter <https://docs.unrealengine.com/4.27/en-US/AnimatingObjects/SkeletalMeshAnimation/FacialRecordingiPhone/>, zuletzt aktualisiert am 18.01.2023, zuletzt geprüft am 25.01.2023.

Sagar, Mark (2006): Facial performance capture and expressive translation for King Kong. In: John Finnegan (Hg.): ACM SIGGRAPH 2006 Research posters. ACM SIGGRAPH 2006 Courses. Boston, Massachusetts, 7/30/2006 - 8/3/2006. Association for Computing Machinery. New York, NY: ACM (ACM Conferences), S. 26. Square-One-Case-Study-Vicon.

The Digital Production Pipeline | SIGGRAPH Courses - YouTube (2023). Online verfügbar unter <https://www.youtube.com/watch?v=I9ql13bEzmc>, zuletzt aktualisiert am 01.01.2023, zuletzt geprüft am 01.01.2023.

The Lion King (2023). Online verfügbar unter <https://www.technicolor.com/thelionking#virtual-production>, zuletzt aktualisiert am 27.01.2023, zuletzt geprüft am 27.01.2023.

Vicon (2022): What is Motion Capture? | What Can I Use Motion Capture For? Online verfügbar unter <https://www.vicon.com/about-us/what-is-motion-capture/>, zuletzt aktualisiert am 20.10.2022, zuletzt geprüft am 28.01.2023.

Virtual Production | Visual Effects + | Wētā FX (2023). Online verfügbar unter <https://www.wetafx.co.nz/research-and-tech/technology/virtual-production/>, zuletzt aktualisiert am 26.01.2023, zuletzt geprüft am 26.01.2023.

## 10 Abbildungsverzeichnis

Abbildung 1:

Aufbau klassischer Animationspipeline

<https://warriorsfate.files.wordpress.com/2016/05/9bb9fcec70d04f41db731338f61872d0.jpg?w=2000&h=>

Abgerufen: 03.01.2023

Abbildung 2:

Markerbasierter Motion Capture Suit

[https://www.vicon.com/wp-content/uploads/2021/05/sharkmob\\_vicon\\_1.jpg](https://www.vicon.com/wp-content/uploads/2021/05/sharkmob_vicon_1.jpg)

Abgerufen: 30.12.2022

Abbildung 3:

Sensorbasierter Motion Capture Suit

[https://assets-global.website-files.com/5e6b63ac3b6e253f11889f39/6351c7f83ed49b7c9340f092\\_ss3-min.webp](https://assets-global.website-files.com/5e6b63ac3b6e253f11889f39/6351c7f83ed49b7c9340f092_ss3-min.webp)

Abgerufen: 04.01.2023

Abbildung 4:

Facial Capture System genutzt bei dem Film „Teenage Mutant Ninja Turtles“ (2014)

[https://www.ilm.com/wp-content/uploads/2021/03/TMNT-vt0100.bg2\\_.SRC\\_.1095.jpg](https://www.ilm.com/wp-content/uploads/2021/03/TMNT-vt0100.bg2_.SRC_.1095.jpg)

Abgerufen: 04.01.2023

Abbildung 5:

Facial Capture System der Unreal Engine

Eigene Darstellung

Abbildung 6:

Markerbasiertes System für eine virtuelle Kamera

<https://deva90sapmc8w.cloudfront.net/insightVCSPProLarge.png>

Abgerufen: 30.12.2022

Abbildung 7:

Sensorbasiertes System für eine virtuelle Kamera

Eigene Darstellung

Abbildung 8:

Behind the Scenes von Avatar 2009

<https://www.youtube.com/watch?v=HmPitTWKudI> (Screenshot)

Abgerufen: 25.01.2023

Abbildung 9:

Behind the Scenes von „The Mandalorian“

<https://www.youtube.com/watch?v=-gX4N5rDYeQ&t=160s> (Screenshot)

Abgerufen: 25.01.2023

Abbildung 10:

System für die Virtuelle Kamera vom Film „Der Polarexpress“

<https://library.imageworks.com/pdfs/imageworks-library-from-Mocap-to-Movie-The-Polar-Express.pdf>

Abgerufen: 04.02.2023

Abbildung 11:

Motion Capture System vom Film „Der Polarexpress“

<https://library.imageworks.com/pdfs/imageworks-library-from-Mocap-to-Movie-The-Polar-Express.pdf>

Abgerufen: 04.02.2023

Abbildung 12:

Kameramann bei der Arbeit am „Lion King“ Remake

<https://www.technicolor.com/sites/default/files/2019-07/the-lion-king-virtual-production-stills-2.jpg>

Abgerufen: 28.01.2023

Abbildung 13:

Vergleich Previs und Finales Bild vom Film „The Jungle Book“

[https://thethirdfloorinc.com/wp-content/uploads/2016/08/TTF\\_VAD\\_ColdLairs\\_C-1280x759.jpg](https://thethirdfloorinc.com/wp-content/uploads/2016/08/TTF_VAD_ColdLairs_C-1280x759.jpg)

[https://thethirdfloorinc.com/wp-content/uploads/2016/08/Disney\\_Jungle\\_Book\\_Final4-1280x693.jpg](https://thethirdfloorinc.com/wp-content/uploads/2016/08/Disney_Jungle_Book_Final4-1280x693.jpg)

Abgerufen: 25.01.2023

Abbildung 14:

Vergleich verschiedene Arbeitsschritte beim CleanUp von Motion Capture Daten

Eigene Darstellung

Abbildung 15:

Skizze Funktionsweise eines Pathtracers

<https://help.autodesk.com/cloudhelp/ENU/AR-Core/images/core-light-sampling.png>

Abbildung 16:

Vergleich verschiedene Rendertechniken

<https://blogs.nvidia.com/wp-content/uploads/2022/03/path-tracing-ray-tracing-rasterization.png>

Abgerufen: 25.01.2023

Abbildung 17:

Skizze Berechnung Subsurface Scattering

<https://help.autodesk.com/cloudhelp/ENU/AR-Core/images/ac-subsurface-scattering-image2014-3-7-8-37-44-7.png>

Abgerufen: 25.01.2023

Abbildung 18:

Einfluss eines Echtzeit-Denoisers auf das Bild

[https://developer.nvidia.com/sites/default/files/akamai/gamedev/camden2.output\\_1080.jpg](https://developer.nvidia.com/sites/default/files/akamai/gamedev/camden2.output_1080.jpg)

[https://developer.nvidia.com/sites/default/files/akamai/gamedev/camden2.input\\_1080.png](https://developer.nvidia.com/sites/default/files/akamai/gamedev/camden2.input_1080.png)

Abgerufen: 25.01.2023

Abbildung 18:

Aufbau Unterschiede zwischen verschiedenen LODs

[https://cdnb.artstation.com/p/media\\_assets/images/images/000/750/789/large/sphereLODs.jpg?1614305873](https://cdnb.artstation.com/p/media_assets/images/images/000/750/789/large/sphereLODs.jpg?1614305873)

Abgerufen: 09.02.2023



## Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Bachelorarbeit selbstständig und ohne unerlaubte fremde Hilfe angefertigt, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Bielefeld, 09.02.2023



Malte Habighorst