

Ein Vergleich zwischen automatisiertem und manuellem Verfahren zur Licht- und Schattensetzung in 3D Szenen

Studiengang Medieninformatik

Bachelorarbeit

vorgelegt von

Joanna Skorupinska

geb. in Warschau

durchgeführt bei
Smeet GmbH

Referent der Arbeit: Prof. Dr. Cornelius Malerczyk
Korreferent der Arbeit: B.Sc. Natalie Göpfert
Betreuer bei Smeet: M.A. Ersin Soyer

Friedberg, 2015

Danksagung

Ich möchte mich an dieser Stelle bei denjenigen Menschen bedanken, ohne die diese Arbeit nicht möglich gewesen wäre. Allen voran bei meinem Referenten Prof. Dr. Cornelius Malerczyk, welcher mir bei der Ideenfindung und Umsetzung stets zur Seite stand und mir, mit meinen unzähligen Mails, viel Verständnis entgegenbrachte. Des Weiteren möchte ich mich auch herzlich bei meinem Betreuer bei Smeet M.A. Ersin Soyer, für seine Unterstützung bei allen Problemen und Fragen bedanken und selbstverständlich auch der Firma SMeet GmbH selbst, für das zur Verfügung gestellte Material.

Ein besonderer Dank gilt noch an meine Familie, ohne deren Unterstützung ich das Studium, wie auch diese Arbeit, wohl nie fertig bekommen hätte.

Vielen, vielen Dank.

Selbstständigkeitserklärung

Ich erkläre, dass ich die eingereichte Bachelorarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die von mir angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Friedberg, September 2015

Joanna Skorupinska

Inhaltsverzeichnis

Danksagung	i
Selbstständigkeitserklärung	iii
Inhaltsverzeichnis	v
Abbildungsverzeichnis	vii
Tabellenverzeichnis	viii
1 Einleitung	1
1.1 Motivation und Relevanz	1
1.2 Problemstellung	3
1.3 Zielsetzung	5
1.4 Aufbau der Thesis	6
2 Stand der Technik	7
2.1 Die Entwicklung der 3D Grafik in Spielen	7
2.1.1 Die heutige 3D Grafik in Spielen	9
2.1.2 Online Spiele	10
Browsergames	11
2.2 Arten der Schatten in Online Spielen	12
2.2.1 Dynamische Schatten	12
Verwandte Arbeiten	13
2.2.2 Statische Schatten	14
Baking	14
Bemalen der Texturen	15
Verwandte Arbeiten	16
2.3 Einsatzgebiete von Computergrafiken	18
2.3.1 Film	18
2.3.2 Medizin	20
2.3.3 Architektur	20
2.4 Zusammenfassung	21

3 Grundlagen	23
3.1 Das Licht in der Physik	23
3.1.1 Lichtstärke und Schatten	24
3.1.2 Reflexion, Brechung und Dispersion	25
3.2 Grundlagen der 3D Beleuchtung	27
3.2.1 Lichtquellentypen	27
3.2.2 Beleuchtungsmodelle	30
3.3 Zusammenfassung	31
4 Konzeptionierung	33
4.1 Auswahl der zu untersuchenden Methoden	33
4.1.1 Automatisiertes Verfahren	34
4.1.2 Manuelles Verfahren	37
4.2 Auswahl des Referenzitems	40
4.3 Anforderungen	42
4.4 Vergleichbarkeit	43
4.5 Zusammenfassung	46
5 Umsetzung	47
5.1 Anwendung am Referenzitem	47
5.1.1 Automatische Methode	48
5.1.2 Manuelle Methode	54
5.2 Zusammenfassung	55
6 Ergebnis	57
6.1 Vergleich	57
6.2 Evaluierung der Methoden	63
6.2.1 Zeit- und Arbeitsaufwand	63
6.2.2 Ästhetik	64
6.2.3 Möglichkeiten	67
6.3 Fazit	68
6.4 Zusammenfassung	69
7 Zusammenfassung und Ausblick	71
Glossar	75
Literaturverzeichnis	77

Abbildungsverzeichnis

1.1	Nacho Guzman zeigt in seinem Video die Beeinflussung durch Licht	2
1.2	Smeet Multiplayer	3
1.3	Formwahrnehmung durch Schatten	4
2.1	Battlezone: Das erste erfolgreiche 3D Spiel	8
2.2	Screenshot: Tom Clancy's The Division	10
2.3	Vergleich ohne und mit Vertexcolor	14
2.4	Textur Baking von High Poly auf Low Poly	15
2.5	Brad Pitts Charakter wurde komplett mittels CGI erstellt	18
2.6	Game of Thrones vor und nach Einsatz von Computergrafiken	19
3.1	Das Optische Spektrum des Lichts	24
3.2	Schlag-, Kern - und Halbschatten	24
3.3	Reflexion eines Lichtstrahls an einem ebenen Spiegel	25
3.4	Lichtbrechung eines Lichtstrahls	26
3.5	Lichtdispersion in einem Prisma	27
3.6	Die unterschiedlichen Schattenbildungen bei den jeweiligen Lichtquellen	29
4.1	Screenshot aus dem Viewport mit Default und Use all Lights	34
4.2	Screenshot zum Baking Verfahren	35
4.3	Die wichtigsten Werkzeuge der manuellen Methode	37
4.4	Pinsel Werkzeug: Unterschiedliche Einstellungen	37
4.5	Radiergummi Werkzeug: Unterschiedliche Einstellungen	38
4.6	Beispiel des Wischfinger Werkzeugs	38
4.7	UV Snapshot dient als Hilfestellung der manuellen Methode	39
4.8	Wireframe Rendering vom Mesh	40
4.9	Referenzitem: UV Layout und Textur	41
4.10	Skript zum Rendern der Ergebnisse beim Referenzitem	44
4.11	Screenshots des Studios zum Rendern der Ergebnisse	45
5.1	Erstes Licht Setup für Baking	48
5.2	Mental Ray Baking Options: Erste Einstellungen	49
5.3	Vergleich der Licht- und Schattenbildung mit Standard- und angepassten Einstellungen	49

5.4	Vorgenommene Änderungen in den Einstellungen der Spot Lights	50
5.5	Endgültiges Licht Setup für Baking	50
5.6	Schlechte Qualität der Textur	52
5.7	Verbesserung der Qualität	53
5.8	Vergleich der Texturen mit unterschiedlicher Deckkraft	54
5.9	Anwendung der manuellen Methode	55
6.1	Vergleich der Methoden anhand der Texturen	59
6.2	Vergleich der Methoden anhand der Renderings	60
6.3	Optische Unterschiede der Texturen	61
6.4	Schattenverlauf der manuellen Methode	64
6.5	Vergleich der Schattenbildung im Brustbereich	65
6.6	Ein Vergleich beider Methoden mit der Hauttextur	65
6.7	Hauttextur des Avatars und direkte Gegenüberstellung beider Methoden	66
6.8	Vergleich der Renderings mit zusätzlichen Details der manuellen Methode	67

Tabellenverzeichnis

6.1	Gegenüberstellung der Arbeitsschritte	58
6.2	Zeitaufwand der manuellen Methode	58
6.3	Zeitaufwand der automatischen Methode	59

Kapitel 1

Einleitung

1.1 Motivation und Relevanz

„The era of the post-photographic film has arrived, and it is clear that for the animator, the computer is essentially „another pencil““
- Paul Wells, 1996

Seien es Mobile, Social oder Browser Games: Online Spiele erobern immer mehr den Markt. Gab es früher Spiele exklusiv für Konsolen oder den heimischen Computer, so gibt es heute sogar mehr Abnehmer für Online, als für Offline Spiele. „Während [nämlich] seit 2004 die Nutzungszeit der klassischen audio-visuellen Medienangebote wie Fernsehen und Radio in Deutschland konstant bleibt oder gar sinkt, erhöht sich die Dauer der Internetnutzung. Auch die relative Verbreitung des Internet in Deutschland nimmt stetig zu, so dass durchaus von einem veränderten Nutzungsverhalten in Richtung neuer internetbasierter Medienangebote gesprochen werden kann. Ähnliches trifft auf die relativen Nutzerzahlen von Onlinespielen und Spielen im Internet zu.“[Sch10] Die NPD Group stellte dieses Jahr sogar eine Steigerung der Online Spieler von 57% seit 2012 fest.¹

Umso mehr an Bedeutung gewinnt die Visualisierung dieser Art von Spielen. Dabei spielt die passende Beleuchtung von Räumen, Personen und Objekten eine sehr große Rolle. „Denn Licht ist wichtig. ... Licht schafft Atmosphäre, lässt Räume groß, klein, freundlich, ungastlich erscheinen. Licht entscheidet mit, ob ... [die Räume, Personen oder Objekte] so ... [aussehen], wie gewünscht. Licht kann [die] Szene aufwerten oder zerstören.“[Lam04]. Es kann die Stimmung und das Wohlbefinden ändern oder auch die Wahrnehmung von Objekten und Menschen beeinflussen.

Wie in Abb. 1.1 zu sehen ist, lässt sich mit Licht sogar bis zu einem gewissen Punkt Modellieren. Dabei hat gute Beleuchtung nicht nur etwas mit Helligkeit zu tun, vielmehr geht es um ein Zusammenspiel von Licht und Schatten. „[Denn] Schatten [sind viel mehr als] einfach nur Stellen, an die kein Licht scheint. ... Durch gezielten Einsatz von Schatten

¹<https://www.npd.com/wps/portal/npd/us/news/press-releases/2015/average-time-spent-playing-games-on-mobile-devices-has-increased-57-percent-since-2012/> — 01.08.2015

werden Farbtöne satter, Schattierungen intensiver, und einzelne Bildelemente können miteinander verknüpft werden. Die Komposition insgesamt wird ... besser.“[Bir00]. Das Licht hat mit seinen Schatten demnach einen enormen Einfluss auf alle Arten der Visualisierung.



Abbildung 1.1: Nacho Guzman zeigt in einem Video am Beispiel einer Frau, wie sehr die Lichteinwirkung das Gesicht und die Stimmung verändern kann.²

Schon in den klassischen bildenden Künsten wie der Malerei verstand man, sich dieses Wissen zu Nutzen zu machen. Und auch die heutigen 3D Artists sollten von der Wichtigkeit des Zusammenspiels von Licht und Schatten wissen. Doch anders als in vielen anderen Produktionen, greifen die Spielentwickler bei Online Games oftmals auf Tricks zur Darstellung von diesem Zusammenspiel zurück. Es wird auf tatsächliche Lichtquellen und den damit entstehenden Schattenbildungen verzichtet und mit Hilfe von Texturen nur eine Illusion solcher erzeugt.

In dieser Arbeit werden zwei Methoden zur Erstellung solch einer Illusion einer künstlichen Lichtquelle untersucht und miteinander verglichen. Zum einen werden mittels „*Baking*“ direkt aus der 3D Szene eine Lichttextur gerendert, welche in Adobe Photoshop³ mit der Textur ohne die Schatteninformationen zusammengesetzt wird. Durch diese Zusammenwirkung beider Texturen entsteht dann der Eindruck von tatsächlicher Beleuchtung des Objektes in der Szene und später auch im Spiel.

Zum anderen werden nach dem eigenen Gefühl Schatten per Hand in Photoshop eingezeichnet. Der direkte Vergleich dieser beiden Methoden soll aufzeigen, ob und wie sehr sich der Arbeitsaufwand und die Ergebnisse voneinander unterscheiden.

²Bildquelle: <https://www.npd.com/wps/portal/npd/us/news/press-releases/2015/average-time-spent-playing-games-on-mobile-devices-has-increased-57-percent-since-2012/> - 01.08.2015

³<http://www.photoshop.com/>

1.2 Problemstellung

Während zum Beispiel in Filmen oder bei *AAA Spielen* eine äußerst komplexe und anspruchsvolle Visualisierung von großer Wichtigkeit ist und alle zur Verfügung stehenden Ressourcen ausgeschöpft werden können, so ist es bei Online Spielen oft genau das Gegenteil. Um so vielen Spielern wie möglich die Benutzung der Spiele zu ermöglichen, müssen die Entwickler auch Rücksicht auf die Leistungskapazität der Rechner ihrer Spieler nehmen. Denn um ein möglichst breites Publikum zu erreichen, sollte angenommen werden, dass nicht alle Nutzer Hardware auf höchstem Niveau besitzen. Werden diese Überlegungen ignoriert, kann dies den Entwicklern viele potentielle Spieler kosten. Insbesondere wenn es sich um Multiplayer Online Games handelt, in denen sich auch mal bis zu mehrere zehn Spieler in einem Raum aufhalten (siehe Abb. 1.2), kann es bei Realtime Renderings schnell zur Ausreizung der *Speicherbandbreite* kommen, wodurch die *Speicherlatenzzeiten* steigen und das Spiel nicht mehr flüssig läuft.



Abbildung 1.2: Ein Screenshot aus dem Online Multiplayer Game Smeet mit mehreren Spielern in einem Raum.⁴

Zwar werden die Prozessoren kontinuierlich besser und ihre Geschwindigkeit steigt auch stetig, allerdings kommen dabei die Speicherbandbreite und Speicherlatenzzeit nicht hinterher. Während die Leistungsfähigkeit der Prozessoren jedes Jahr bis zu 71% steigt, sind es bei der Speicherbandbreite nur 25% und bei der Speicherlatenzzeit gerade einmal 5%. [AMHH08]

Dabei ist es ein sehr verbreiteter Irrtum, dass die Polygonenanzahl die Bildfrequenz am meisten beeinflusst. In Wirklichkeit wird sie auch von der Anzahl der gesetzten Lichter, der Komplexität der Materialien, den Speicher, den sowohl Meshes als auch Texturen benötigen und vielem mehr stark gesteuert. [Bla11]

⁴Bildquelle: <http://virtualworldsforteens.com/games-like-smeet> - 31.07.2015

Aus diesem Grund wird gerade bei solchen Online Multiplayer Games oft auf die neuesten, technisch anspruchsvollsten Methoden der grafischen Darstellung mit mehreren hochqualitativen Lichtquellen verzichtet und lediglich eine Illusion von Beleuchtung und Schattierung erzeugt, um die Bildfrequenz möglichst hoch zu halten, während gleichzeitig versucht wird, das Spiel so hochwertig wie möglich wirken zu lassen. Auch wenn es sich dabei nur um eine Illusion von Beleuchtung handelt, wirkt die Umgebung durch diesen Trick direkt viel realistischer und auch angenehmer auf den Nutzer. „Denn erst durch die Beleuchtung eines Objektes und die damit verbundene Abschattung der lichtabgewandten Seiten wird die 3-dimensionale Form des Objektes im Gehirn des Beobachters rekonstruiert. Im Fachjargon nennt man diesen Vorgang „Formwahrnehmung aus Schattierung““ [NFHS12] (siehe Abb. 1.3)



Abbildung 1.3: Formwahrnehmung durch Schattierung aus [NFHS12]. Links: Ohne Schatten wirkt der Triceratops flach | Rechts: Durch Beleuchtung und Schattierung erscheint er 3-dimensional.

Es existieren mehrere Möglichkeiten mit Hilfe von Texturen einen Eindruck von Licht im Raum zu erwecken. Die Wahl der Methode hängt allerdings oftmals nur von den persönlichen Vorlieben und Vorkenntnissen eines Designers ab, denn direkte Gegenüberstellungen zwischen den Methoden mit ihren jeweiligen Vor- und Nachteilen gibt es kaum. Dabei können in Wahrheit mehrere Faktoren eine wichtige Rolle spielen, die sich direkt miteinander vergleichen lassen. Dies können Zeit- und Arbeitsaufwand, Genauigkeit oder die jeweiligen Grenzen der Darstellung sein. Aber auch die Realitätsnähe und physikalische Korrektheit können, abhängig vom gewünschten Ergebnis, zu einer Entscheidung verhelfen.

1.3 Zielsetzung

Erwähnenswert ist, dass der Schwerpunkt dieser Arbeit nicht beim Rendering liegt, sondern bei der Texturierung. Demnach werden keine Rendersysteme oder Einstellungen im Renderverfahren dokumentiert und verglichen, darüber existieren schon zahlreiche Dokumente. Dagegen konzentriert sich die folgende Bachelorarbeit auf den Vergleich zweier Verfahren, mittels Texturierung und ohne tatsächlichem Licht in der finalen Szene, einen Eindruck von eben diesem Licht zu erwecken. Mit Hilfe von Schatten können räumliche Strukturen und Details simuliert werden, wodurch sie bei der Darstellung von 3D Computergrafiken zu den wichtigsten Hilfsmitteln überhaupt gehören.

Im Rahmen dieser Bachelorarbeit werden zwei Methoden zur Darstellung von Schatten mittels Farb-Mapping behandelt. Dabei handelt es sich zum einen um das manuelle Handzeichnen von Schatten mit Pinseln in Adobe Photoshop, zum anderen werden die Schattierungen auf die Objekte „gebacken“. Während bei der ersten Technik alle Schatten nach subjektivem Gefühl des Designers eingezeichnet werden, verhält es sich beim Baking anders. Als Baking wird dabei das Speichern von vorberechneten Beleuchtungsdaten auf Texturen bezeichnet. Es wird grundsätzlich zwischen zwei Methoden unterschieden: Vertex Baking und Texture-based Baking. Näheres zum Baking Verfahren gibt es in Kapitel 2.2.2 Baking.

Ziel der Arbeit ist es, einen direkten Vergleich der Verfahren und Ergebnisse aufzustellen. In wie fern unterscheidet sich der Arbeitsaufwand und welche Methode eignet sich besser bei einem knappen Zeitplan. Des Weiteren wird untersucht, welche Methode die realistischeren Ergebnisse liefert, in wie fern sich diese voneinander unterscheiden und was für Vor- und Nachteile die jeweiligen Methoden besitzen. Beide Methoden werden anhand eines Referenzitems verglichen, welches von der Firma *Smeet GmbH* für diese Arbeit zur Verfügung gestellt wurde.

1.4 Aufbau der Thesis

Nach der Einführung in das Thema, dessen Problemstellung und die Zielsetzung dieser Arbeit, beschreibt das nächste Kapitel den heutigen Stand der Technik. Darin wird die Entwicklung der 3D Grafik in Spielen erläutert, sowie die unterschiedlichen Arten von Schatten in Online Spielen aufgezeigt. Zum Schluss werden noch kurz weitere Einsatzgebiete von Computergrafiken beschrieben, bevor das dritte Kapitel folgt.

Darin werden die Grundlagen, die für diese Arbeit wichtig sind, erläutert. Zunächst wird grob das Verhalten von Licht in der Physik erklärt, woraufhin die Grundlagen der 3D Beleuchtung vermittelt werden.

Im nächsten Kapitel beginnt die Methodik der Thesis, angefangen bei der Konzeptionierung. Darin werden die beiden Methoden beschrieben, die Auswahl des Referenzitems erklärt, sowie Anforderungen an die Umsetzung und Vergleichbarkeit aufgestellt.

In Kapitel 5 erfolgt die Umsetzung des Konzepts und beide Methoden werden am Referenzitem angewendet. Dabei wird die Verfahrensentwicklung dokumentiert und die entstandenen Texturen dargestellt.

Das darauffolgende Kapitel 6 behandelt die Ergebnisse der beiden Methoden und stellt die jeweiligen Arbeitsschritte, sowie den Zeitaufwand beider Verfahren gegenüber, woraufhin auch die Texturen und Renderings miteinander verglichen werden.

Es folgt eine Evaluierung der Methoden, ihren Unterschieden zwischen Zeit- und Arbeitsaufwand, der Ästhetik, sowie ihrer jeweiligen Möglichkeiten. Zum Schluss wird ein Fazit aus den gewonnenen Erkenntnissen gezogen.

Den Abschluss der Arbeit bildet eine Zusammenfassung der ganzen Thesis und ein kurzer Ausblick.

Kapitel 2

Stand der Technik

Im vorherigen Kapitel erfolgte eine Einführung in das Thema der Arbeit. Es wurde die Motivation für die Auswahl dieses Themas beschrieben und dessen Relevanz in der heutigen Zeit erläutert. Darauf folgte eine Beschreibung der Problemstellung und der Ziele, die im Laufe dieser Arbeit erreicht werden sollen und der Aufbau der Thesis. In diesem Kapitel wird die Geschichte der Grafik in Computerspielen beschrieben, um eine Vorstellung zu erhalten, wie sich diese im Laufe der Jahre entwickelt hat, welche Möglichkeiten zur Visualisierung bestehen und wie sich diese im Vergleich zu den Anfangszeiten geändert hat. Danach werden Arten von Schatten in Online Spielen erläutert, um einen Einblick in den möglichen Einsatz von Schattenalgorithmen zu geben. Damit wird aufgezeigt, welche anderen Möglichkeiten zur Implementierung von Schatten in Online Spielen - neben den in dieser Arbeit untersuchenden Methoden - existieren. Zuletzt wird noch auf die verschiedenen Einsatzgebiete von Computergrafiken eingegangen. Zwar werden die Methoden lediglich unter den Aspekt der Spielentwicklung untersucht, doch Computergrafiken sind heutzutage in viel mehr Gebieten zu finden. Somit wird in diesem Kapitel auch der unterschiedliche Einsatz dieser Grafiken erläutert.

2.1 Die Entwicklung der 3D Grafik in Spielen

Die Entwicklung der Videospiele ist streng genommen gleichzeitig eine Entwicklung der Technologie.¹ Beide Komponenten sind stark miteinander verknüpft und bauen gegenseitig aufeinander auf. Schon Anfang der 1950er Jahren wurde Computergrafik entwickelt, um das, was bis dato für das menschliche Auge unsichtbar war, sichtbar zu machen. Allerdings war sie zu ihren Anfangszeiten noch nicht für Designer und Spielentwickler ausgelegt, sondern diente hauptsächlich dem Militär, den Manufakturen oder der Datenverarbeitung. Erst während den 70er Jahren begannen sich zunehmend mehr Künstler für die Möglichkeiten der Computergrafik zu interessieren und so war dieses Zeitalter eine der bedeutendsten überhaupt für die dreidimensionale Animation. Viele der heute benutzten Techniken basieren auf den damals veröffentlichten Technologien.[Ker04] Schließlich fand 1973 Sigggraph, die heute

¹<https://www.jesperjuul.net/thesis/2-historyofthecomputergame.html> — 03.08.2015

2. STAND DER TECHNIK

weltweit berühmteste Tagung für Computergrafiken, das erste Mal statt.² Danach nahmen die Fortschritte und die damit verbundenen Möglichkeiten immer rasanter zu, bis sie endlich auch die Spielproduktion erreichten.

Anfangs waren Computerspiele nur mit zweidimensionalen Grafiken ausgestattet. Doch diese Art der Grafiken führten eine große Einschränkung mit: Der Spieler konnte sich nur nach rechts oder links fortbewegen, nicht frei, wie es heute in 3D Spielen üblich ist. Der Sprung von 2D zu 3D war demnach gewaltig und hatte den großen Vorteil der Bewegungsfreiheit.

Mit *Battlezone* brachte *Activision*³ 1980 das erste, erfolgreiche 3D Spiel auf den Markt (siehe Abb. 2.1). Zwar steht die damalige Grafik im Vergleich mit der heutigen in keiner Konkurrenz, dennoch war *Battlezone* für die damalige Zeit so realistisch, dass es sogar vom amerikanischen Militär zum Training für Panzereinsätze benutzt wurde.⁴

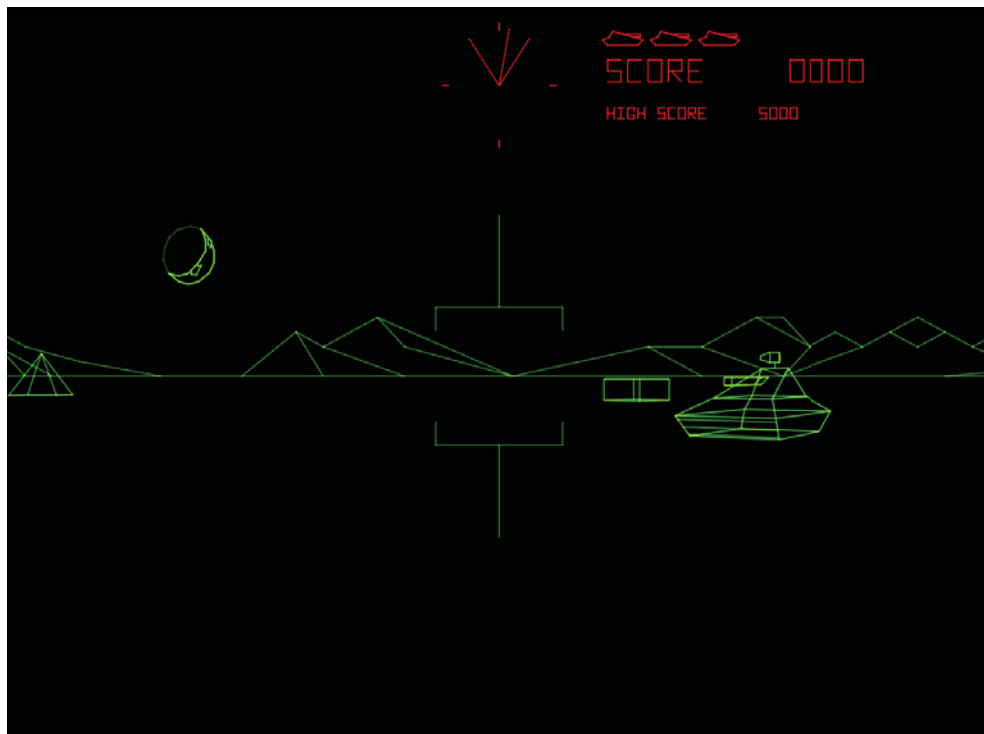


Abbildung 2.1: Battlezone: Das erste erfolgreiche 3D Spiel.⁵

Während es zur damaligen Zeit schon beeindruckend war, ein Spiel und deren Umgebung lediglich dreidimensional aussehen zu lassen, so sitzen die heutigen Ziele viel höher. Es reicht nicht mehr nur, eine 3D Umgebung zu erschaffen. Mittlerweile überschlagen sich Publisher mit Ideen und Mitteln, aus der Masse herauszustechen. Sei es eine unverwechselbare Grafik

²<http://www.siggraph.org/about/history> — 03.08.2015

³<https://www.activision.com/de/>

⁴www.techradar.com/news/gaming/the-evolution-of-3d-games-700995 - 03.08.2015.

⁵Bildquelle: <http://top-img.com/b/battlezone-arcade-game> - 03.08.2015.

wie in *Borderlands*⁶, eine besonders große Spielwelt mit schier unzähligen Möglichkeiten wie bei *Grand Theft Auto*⁷, oder eine so fotorealistische Computergrafik, dass sie als solche kaum noch zu erkennen ist, wie bei *Halo*⁸. Mit den rasant entwickelten Technologien stehen Entwicklern und Designern mehr Mittel zur Verfügung, die gewünschten Ergebnisse zu erzielen, als jemals zuvor.

2.1.1 Die heutige 3D Grafik in Spielen

„Das heutige Computerspiel ist gekennzeichnet durch Nachbildung des dreidimensionalen Raumes, durch Bewegungs- und Interaktionsmöglichkeiten. ... Durch Kopf- und Körperdrehung verschaffen wir uns einen Überblick über den Raum. Dies wird mit der Steuerung des Computers beispielsweise durch Klicken und Ziehen mit der Maus und der so ermöglichten Rundumsicht von 360° innerhalb virtueller Welten erzielt.“[Kor04] Es existieren nahezu keine Einschränkungen in der Bewegung und Interaktion für die Spieler, es sei denn dies ist von den Entwicklern gewollt. Auch der Visualisierung sind, so lange Ansprüche an die Hardware keine große Rolle spielen, kaum Grenzen gesetzt. Mit der imposanten Entwicklung von 3D-Grafik Software wie Maya, 3ds Max, Blender oder Cinema 4D lassen sich alle möglichen bildlichen Vorstellungen umsetzen. Beispielsweise lassen sich mit dem Ray-Tracing Algorithmus tatsächliche Reflexionen und Transparenzen darstellen, was vor einigen Jahren noch unmöglich war.

Mit Hilfe von Programmierschnittstellen wie *DirectX* oder *OpenGL* können äußerst komplexe Spiele dargestellt werden, welche den Entwicklern einen enorm großen Freiraum in der Erstellung ihrer eigenen 3D-Welt bieten. So lassen sich mit DirectX, so lange die Hardware mitspielt, sehr anspruchsvolle Spiele mit schneller 3D-Grafik, glasklarem 3D-Sound, Netzwerkunterstützung und Ansteuerung aller Arten von Eingabegeräten (Tastatur, Maus, Joystick, Game-Pad, Lenkrad, Gaspedal...) entwerfen.[Sch04]

Game Engines wie Unity, Unreal Engine oder CryEngine sind ein komplettes Entwicklungsframework und beinhalten Render-, Script-, Animations- und Physiksysteme und noch mehr, womit sich ganze eigene 3D-Welten mit ihren eigenen Regeln entwerfen lassen. Durch die mittlerweile höchst akkuraten Physiksysteme sind Spiele in der Lage, die Welt in der wir leben nahezu identisch wiederzugeben, mit all ihren physikalischen Gesetzen und Begrenzungen. Dadurch können sich Spieler in einer realitätsgetreuen Umgebung bewegen, was hauptsächlich bei Simulationen, wie beispielsweise Flugsimulation, oder auch bei Shootern erwünscht und erstrebenswert ist, damit eine geworfene Handgranate einen Berg oder Hügel auch herunterkullert anstatt nach dem Aufkommen einfach darauf liegen zu bleiben.⁹

⁶<http://borderlandsthegame.com>

⁷<http://www.rockstargames.com/grandtheftauto>

⁸<https://www.halowaypoint.com>

⁹<http://www.uni-protokolle.de/Lexikon/Game-Engine.html> — 04.08.2015



Abbildung 2.2: Screenshot aus dem kommenden Spiel Tom Clancy's The Division zur Darstellung des möglichen Realismus der Snowdrop Engine.¹⁰

Als gutes Beispiel fungiert hier die neue *Snowdrop-Engine*, welche eigens für die sogenannten *Next-Gen-Spiele* von *Ubisoft Massive*¹¹ entwickelt und auch für das kommende Spiel *Tom Clancy's The Division*¹² eingesetzt wurde. „Snowdrop solle vor allem für mehr Realismus, bessere grafische Details und für ein neues Niveau an Spielphysik und Animation sorgen.“¹³ (siehe Abb. 2.2) Bei anderen Spielen kann sich der gewünschte Grad an Realismus wiederum genau im Gegenteil befinden, doch auch das stellt dank der heutigen Game-Engines keine Hürde dar. Die Entwickler können die Gesetze unserer Physik nach Belieben verzerren oder umdrehen, wodurch ganz eigene Spielerlebnisse entstehen.

2.1.2 Online Spiele

Es gibt eine Vielzahl von verschiedenen Arten der Online Spiele, welche allerdings allesamt so bezeichnet werden. So gibt es beispielsweise einen Unterschied ob ein Spiel *im* Internet gespielt wird, in diesem Fall befindet sich der Spielort tatsächlich im Internetbrowser, oder aber nur *über* das Internet, wodurch das Internet hierbei nur als Kanal dient um Botschaften, Handlungen oder Züge auszutauschen.[JDL08] Für diese Arbeit ist nur der erste Fall relevant, somit wird auch nur auf diesen eingegangen.

¹⁰Bildquelle: <http://www.pcgameshardware.de/The-Division-Spiel-37399/News/The-Division-Snowdrop-die-effizienteste-Engine-1124932/> — 04.08.2015

¹¹<http://www.massive.se/>

¹²<http://tomclancy-thedivision.ubi.com>

¹³www.pcgameshardware.de/The-Division-Spiel-37399/News/The-Division-Snowdrop-die-effizienteste-Engine-1124932/ — 04.08.2015

Browsergames

„Browsergames sind Onlinespiele, die ohne Download bzw. Installation eines Datenträgers auskommen, sondern den Web-Browser als Schnittstelle zwischen Spieler und Spielwelt nutzen.“[JDL08] „... [Sie] sind für viele Dinge bekannt, aber nur selten für ihre grafische Opulenz“¹⁴, was letztendlich daher kommt, dass sich bislang nur Indie-, nicht aber die großen Entwicklerstudios mit ihnen befassten. Dies könnte sich in der Zukunft allerdings ändern, denn mit der wachsenden Popularität der Spiele, beginnen sich immer mehr große Namen dafür zu interessieren, womit auch die Grafik-Engines immer mächtiger werden. Doch damit verlangen die grafisch hochwertigen Spiele den heimischen Rechnern einiges ab. Wer allerdings auf eine starke Belastung der Rechner verzichten möchte, der muss auch Abstriche bei der Grafik machen. Um aber auf eine schöne Optik nicht komplett verzichten zu müssen, gibt es eine Anzahl an Hilfsmittel, die den Spielentwicklern dabei helfen können, das Aussehen ihrer Spiele zu verbessern, ohne die Computer und Entwickler mit zu langen Render- und Produktionszeiten an ihr Limit zu treiben. Eine davon ist, Licht und Schatten nicht in Echtzeit während des Spiels zu berechnen, sondern beides einzig durch Einsatz von Texturen auf die Objekte und Umgebung zu bringen. Zwei Methoden zu dieser Lichtimitation werden in dieser Arbeit behandelt.

¹⁴<http://www.giga.de/spiele/drakensang-online/specials/browsergames-die-spiele-mit-der-besten-grafik/>
- 05.08.2015

2.2 Arten der Schatten in Online Spielen

Im vorherigen Punkt wurde die Geschichte und Entwicklung der Computergrafik beschrieben. Doch mit den steigenden Möglichkeiten in der Computergrafik, steigen auch die Erwartungen der Betrachter an den Grad von Realismus bei deren Einsatz. So ist ein gewisser Realismus Voraussetzung für die positive Aufnahme von Visualisierungen am Computer. Dabei sind Schatten der wohl wichtigste Faktor für solch eine realistische Umgebung. Sie sind demnach auch in Online Spielen unverzichtbar. Es wird grundsätzlich zwischen globalen und lokalen Schattenmodellen unterschieden. Da für diese Arbeit nur letztere relevant sind, wird auch nur auf diese eingegangen.

2.2.1 Dynamische Schatten

Bei der dynamischen Beleuchtung verwendet man richtige Lichter im endgültigen Spiel, so dass Schatten dynamisch während des Spielens in Echtzeit berechnet werden. Dies kann durch den Einsatz eines sogenannten Schattenalgorithmus erzielt werden.

Planar Shadows

Neben dem „*pre-rendered*“-Schattenalgorithmus ist dies der einfachste Algorithmus zur Berechnung von Schatten. Dabei werden alle schattenwerfenden Objekte mittels Schattenprojektionsmatrix als 2D-Projektion auf eine Ebene geworfen und in der Schattenfarbe gerendert. Der größte Vorteil dieser Methode ist ihre Schnelligkeit, dadurch ist sie gut für dynamische Szenen geeignet. Doch die, mit diesem Verfahren erzeugten Schatten, sind zwar dynamisch, werden jedoch immer auf eine Ebene projiziert und funktionieren somit nur mit planaren Ebenen. Des Weiteren entstehen bei dieser Methode Probleme, wenn die Schatten nur auf einen Teil der Ebene fallen sollen. Dies lässt sich mit dieser Methode nämlich nicht erreichen. Oder wenn sich halb transparente Schatten überlappen, was zu dunklen Flecken in den Schatten führt.

Shadow Mapping

1978 von Lance Williams entworfen [Wil78], basiert dieser Algorithmus auf der Idee, eine Szene aus der Perspektive der Lichtquelle zu rendern und den *Tiefenbuffer* als Shadow Map zu verwenden. Die Szene wird dabei zweimal gezeichnet. Zuerst befindet sich die Kamera bei der Lichtquelle und die Tiefenwerte der Pixel werden in der Shadow Map gespeichert. Beim zweiten Durchgang befindet sich die Kamera im Augpunkt und es werden die transformierten Tiefenwerte mit dem aktuellen Tiefenbuffer verglichen, um eine mögliche Verdeckung zu bestimmen. Dieser Algorithmus ist zwar sehr schnell und universell, daher selbst in komplexen Szenen gut anwendbar, allerdings kommt es oftmals zu starken *Aliasing*-Effekten.

Shadow Volume

Dieses Verfahren wurde erstmals 1977 von Frank Crow [Cro77] beschrieben, allerdings wurde es lange Zeit nicht eingesetzt. Das liegt unter anderem daran, dass der Algorithmus sehr rechenaufwändig ist und die Grafikkarten erst seit wenigen Jahren mit Beschleunigern von 3D-Schnittstellen ausgerüstet sind. Zunächst wird die Silhouette des schattenwerfenden

Objekts ermittelt. Dafür müssen die Normalen zu jedem Polygon des Objektes bestimmt werden und anhand dieser Normalen wird überprüft, ob die Polygone Richtung Licht zeigen. Daraufhin wird die Silhouette extrudiert, wodurch die Schattenpolygone erstellt werden, die die Shadow Volumes umschließen. Von der Kamera aus gesehen, wird alles hinter einem nach vorne zeigendem Schattenpolygon schattiert, während nach hinten zeigende Schattenpolygone dies wieder rückgängig machen. Im Grunde besteht der Algorithmus aus folgenden Schritten:

1. Bestimmung der Silhouetten-Kanten des schattenwerfenden Objekts
2. Die Silhouetten-Kanten werden extrudiert, wodurch Halbebenen erzeugt werden
3. In jedem Pixel werden die Schnittpunkte der Geraden vom Augpunkt zum Pixel mit dem Schattenvolumen bestimmt
4. Für jeden Schnittpunkt, bei dem die dazugehörige Halbebene des Schattenvolumens zum Augpunkt hin orientiert ist, wird ein Zähler erhöht, für jede andersherum orientierte Schnittfläche erniedrigt.
5. Wenn der endgültige Wert größer Null ist, liegt der Pixel im Schatten, andernfalls liegt dieser nicht im Schatten [Sch07]

Mit Hilfe einer Erweiterung, dem sogenannten Stencil Shadow Volume Algorithmus, wird der Stencil Buffer mit einbezogen. Dieser ist ein zusätzlich nutzbarer Speicherbereich, in welchem man Pixeln bestimmte Werte zuweisen kann, wodurch festgelegt werden kann, welche Pixel beleuchtet werden und welche nicht. Dieser Algorithmus hat den Vorteil, dass kein Aliasing entsteht. Der größte Nachteil dieser Methode ist allerdings, wie bereits erwähnt, der Rechenaufwand. Je komplexer das schattenwerfende Objekt ist, desto Aufwändiger wird der Algorithmus.

Verwandte Arbeiten

Es gibt viele Arbeiten darüber, wie dynamische Schatten in Echtzeit Umgebungen berechnet werden können: Die Diplomarbeit von Jens Freiling [Fre06] handelt von Echtzeitfähigen Schatten, allerdings in Mixed-Reality Umgebungen. Dennoch werden in dieser Arbeit die gängigen Schattenverfahren beschrieben und angewendet. Währenddessen beschäftigt sich Nico Hempe in seiner Studienarbeit [Hem04] nur mit Shadow Mapping und in [OSK⁺14] wird eine Möglichkeit zum Einsatz virtueller Shadow Maps für mehrere hundert Lichter in Echtzeit Systemen bei Spielen beschrieben. Auch unter Verwendung des JavaScript basierten X3DOM Framework lässt sich in [EJK15] nachlesen werden, wie dynamische Schatten in Echtzeit mittels WebGL auf dem Browser wiedergegeben werden können. Zur Verbesserung der Bildqualität und des Kohärenzverhaltens schrieb Thomas Annen in seiner Dissertation [Ann08] über ein neues Echtzeit-Schattenverfahren, welches das Filtern von Shadow Maps ermöglicht. Auch Stefan Brabec schreibt in seiner Dissertation [Bra04] über eine Schattentechnik für interaktive Echtzeit Anwendungen und in der Diplomarbeit von Timo Schairer [Sch07] werden die verschiedenen Schattenalgorithmen für Echtzeitanwendungen verglichen und bewertet.

2.2.2 Statische Schatten

„Der einfachste Schattenalgorithmus ist der „*pre-rendered Shadows*“-Algorithmus. ... [Er] wird mit einem externen Programm berechnet ... und im Spiel selbst als eine Textur verwendet. Der Schatten ist somit statisch - er verändert sich nicht.“[Sch03] Aber auch das triviale Einzeichnen von Schattierungen auf die Textur in einem Bildbearbeitungsprogramm kann statische Schatten für eine Szene erzeugen. Für die statische Beleuchtung bzw. Schattierung werden oftmals Lichtquellen und deren Schattenbildung in der Szene nur simuliert. Es befinden sich demnach keine tatsächlichen Lichtquellen in der endgültigen Szene, was eine einfache Implementierung, verkürzte Renderzeit und erhöhte Geschwindigkeit im Spiel zum Vorteil hat. Wie nämlich schon im Kapitel 1.2 Problemstellung beschrieben, kann bei komplexen Szenen das Rendern von Umgebungslicht, weichen Schatten, Reflektionen oder indirekter Beleuchtung in Echtzeit schnell sehr rechenintensiv werden. Die statischen, vorge-rendernten Schatten bieten sich hierbei als gute Lösung an.

Baking

Das Speichern von vorberechneten Beleuchtungsdaten wird allgemein als Baking bezeichnet. Es wird zwischen zwei gängigen Methoden unterschieden: Das Speichern der Daten in den Mesh Vertices (Vertex Baking), oder in Texturen (Texture-based Baking).

Vertex Baking wird üblicherweise bei Low-End Plattformen wie Smartphones, Tablets oder der Nintendo Wii verwendet, bei denen der Speicher- und Laufzeit Verbrauch dazu führt, auf Kompromisse eingehen zu müssen [BL08]. Bei diesem Verfahren werden die Informationen automatisch in den Vertex Farben des Meshs gespeichert. Jeder Vertex eines Meshs kann eine RGB und / oder einen Alpha-Wert speichern, dadurch wird ein Eindruck von Volumen erweckt, siehe dazu Abb. 2.3. Mit dem Backen dieser Farben kann im Endeffekt viel Rechenzeit eingespart werden.



Abbildung 2.3: Das gleiche Modell eines Baumes: Links: Ohne Vertex Color | Rechts: Mit Vertex Color ¹⁵

¹⁵Bildquelle: <http://wiki.polycount.com/wiki/VertexColor> - 11.08.2015

Beim Texture-based Baking wird eine sogenannte Light-Map erstellt, welche die Materialien, Texturen und Beleuchtung der gebackenen Objekte enthält. Für ein gutes Textur-Mapping Ergebnis, ist ein sehr ordentliches und fachgerechtes UV-Mapping erforderlich, andernfalls kann es zu Verzerrungen oder ungewollt sichtbaren Schnittkanten führen. Demnach ist ein einwandfreies UV-Mapping die Grundvoraussetzung für die erfolgreiche Implementierung dieser Methode. Zwar lässt sich eine UV-Map in den meisten 3D Grafik Programmen auch automatisch generieren, allerdings lassen die Ergebnisse dabei oftmals zu wünschen übrig, wodurch ein manuelles Verfahren empfehlenswerter ist.

Das Texture-based Baking wird oftmals dazu benutzt, Details von einem Modell auf ein anderes zu projizieren. Da Computerspiele mit der Menge an Informationen bei mehreren Modellen mit mehreren Millionen Polygonen nicht umgehen können und dadurch nicht flüssig laufen, kann mit diesem Verfahren „geschummelt“ werden. Dabei werden von einem Objekt gleich zwei Meshes erstellt: Ein *Low-Poly* und ein *High-Poly* Mesh. Danach werden alle relevanten Informationen des hochauflösenden Meshs mittels dessen UV-Map auf eine Textur gebacken und diese dann auf das niedrig auflösende Mesh transferiert. Somit entsteht der Eindruck, das Low-Poly Modell wäre hochauflösend und voller Details, ohne tatsächlichen Speicher oder Renderzeit zu beanspruchen. (siehe Abb. 2.4)



Abbildung 2.4: Mittels Baking werden auf ein Low-Poly Modell die Details des High-Poly Modells projiziert.¹⁶

Bemalen der Texturen

Neben dem Baking besteht noch eine weitere Möglichkeit zur Schattierung von Objekten unter Verwendung von Texturen. Licht und Schatten lassen sich nämlich auch einfach mittels eines Bildbearbeitungsprogrammes direkt auf die Texturen der jeweiligen Objekte malen. Auch hierfür ist natürlich ein ordentliches UV-Mapping die wohl wichtigste Voraussetzung

¹⁶Bildquelle: <https://forums.unrealtournament.com/showthread.php?14298-WORKFLOW-Create-a-Low-Poly-model-and-use-RoundedEdge-shader-to-create-High-Poly> - 11.08.2015

für ein optimales Ergebnis. Der große Vorteil bei diesem Verfahren ist, sich nicht auf berechnete Schatten oder einen visuellen Stil beschränken zu müssen. Hierbei sind der Kreativität keine Grenzen gesetzt und es können vielerlei Effekte und Details hinzu gemalt werden. Mit Hilfe von Programmen wie *Autodesk Mudbox*¹⁷ oder *Pixologic zBrush*¹⁸ kann direkt auf dem Objekt gezeichnet werden, während bei einem Bildbearbeitungsprogramm wie Adobe Photoshop das UV-Mapping zur Referenz als Hintergrundebene beim Zeichnen der Textur verwendet wird.

UV-Maps

Egal für welche Methode man sich entscheidet, eine ordentliche UV-Map ist die Grundvoraussetzung für den Transfer von 2D-Texturen auf 3D-Modelle. Anders als bei der Projektion, werden keine Maps durch den Raum projiziert, die Positionierung der UV-Map richtet sich hierbei nach den UV-Koordinaten der Oberfläche. Implizite UV-Koordinaten basieren dabei auf NURBS-Oberflächen, während explizite UV-Koordinaten einem Polygonnetz zugewiesen werden. Da das Referenzitem ausschließlich aus Polygonen besteht, wird hierbei nur auf explizite UV-Koordinaten eingegangen.

Polygone besitzen in ihrer Geometrie keine implizite Gitterstruktur und somit auch nicht die, für NURBS-Oberflächen typischen, impliziten UV-Koordinaten. Die UV-Koordinaten können ihnen lediglich zugewiesen werden, wobei ihre Verwaltung von der eingesetzten Software und der jeweiligen Version abhängig sind. So legen manche Programme bereits beim Erstellen eines primitiven Polygonnetzes UV-Koordinaten an. Andere Programme wiederum texturieren ein Polygonnetz mit Hilfe einer Projektion durch den Raum, woraufhin die Map in explizite UV-Koordinaten überführt wird. Dabei weist die Software jedem Scheitelpunkt, abhängig von seiner Position in der ursprünglichen Texturprojektion, die entsprechenden UV-Koordinaten zu. Diese expliziten UV-Koordinaten werden hauptsächlich für die Wiedergabe von Inhalten in Echtzeit, wie der Entwicklung von Computerspiel-Charakteren, eingesetzt [Bir00].

Verwandte Arbeiten

Es existieren Arbeiten und Artikel darüber, wie man der Szene und Objekten durch Baking Tiefe verleihen kann. [KBS11] beschäftigt sich mit der Optimierung von Vertex Baking. In [Sch12] wird derweilen das Vertex und das texture-based Baking miteinander fusioniert um somit eine effiziente Mischung der Vorteile beider Methoden zu erhalten, ohne die Renderzeit maßgeblich zu verlängern. Chris Green beschreibt währenddessen in seiner Arbeit [Gre07] für Siggraph eine interessante Möglichkeit, auch in Normal Maps Eigenschatten und Ambient Occlusion zu rendern. Allerdings weicht diese Ausarbeiten stark vom eigentlichen Thema ab, da es sich in dieser Arbeit einzig und allein um Farb-Mapping handelt, ohne den Einsatz von Bump Maps. Und auch Ralf Habel und Michael Wimmer beschreiben in ihrer Arbeit [HW10] welche Vorteile Irradiance Normal mapping, einer Verbindung aus Light- und Normal Mapping in Spielen bieten kann. Hierbei handelt es sich wieder um eine Technik unter Verwendung der Normal Maps. Auch eine Lösung für das erforderliche, einwandfreie

¹⁷<http://www.autodesk.com/products/mudbox/>

¹⁸<http://pixologic.com/>

UV-Mapping existiert und wurde in der Arbeit von [BL08] beschrieben, allerdings wurde eine erfolgreiche Implementation dieser Methode bislang lediglich auf High-End Grafik Hardware vorgestellt.

Es sind ebenfalls Arbeiten und Bücher zur Verwendung von Photoshop für Texturen vorhanden, wie beispielsweise [Ahe14]. Doch es gibt kaum direkte Vergleiche der Vor- und Nachteile vom Baking der Lichtdaten gegenüber der Handbemalung dieser Daten in Photoshop. Diese Arbeit wird sich mit so einem direkten Vergleich beider Methoden und dessen Evaluierung beschäftigen.

2.3 Einsatzgebiete von Computergrafiken

Diese Arbeit untersucht hauptsächlich den Aspekt der Spielentwicklung, allerdings ist der Einsatz von Computergrafiken selbstverständlich nicht nur auf diesen Bereich beschränkt. Somit gibt es vielerlei verschiedene Einsatzgebiete von digitalen Bildern und Animation. In diesem Punkt werden die gängigsten Bereiche aufgezeigt.

2.3.1 Film

Neben der Spielentwicklung, hat sich die Filmbranche in den letzten Jahren wohl zu den Hauptnutzern von Computervisualisierung gesteigert. Mit den Jahren und den wachsenden Möglichkeiten im Bereich der Animation, stiegen die Anwendungen von Computern in der Filmentwicklung drastisch an. Kaum eine Produktion kommt heutzutage ohne den Einsatz von digitalisierten Bildern und Animationen aus. Und dabei ist nicht nur die Rede von Spezialeffekten, die jeden Betrachter zweifelsfrei erkennen lässt, dass diese am Computer generiert worden sind, wie die Affen aus *Planet der Affen: Revolution*, die Autobots aus Michael Bays *Transformers* Reihe oder nahezu jede Minute in *Pandora, der Welt aus James Camerons Avatar*.

Oftmals ist den Zuschauern überhaupt nicht bewusst, dass Teile der Landschaft so gar nicht existieren, sondern dass diese am Computer „verbessert“ bzw. angepasst wurden. Auch große Menschenmassen am Computer entstehen zu lassen, rentiert sich für die meisten Produktionen viel mehr, als mehrere hundert Statisten ans Set zu holen. So geschieht dies sehr häufig bei der HBO Serie *Game of Thrones*, wie auf Abb. 2.6 zu sehen ist.

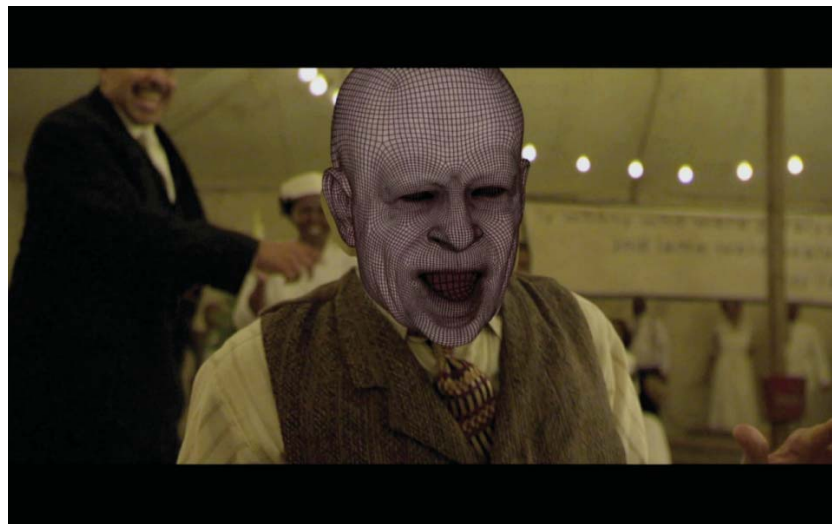


Abbildung 2.5: Brad Pitts Charakter im Film „Der seltsame Fall des Benjamin Button“ wurde die erste Hälfte des Films beinahe vollständig am Computer erstellt¹⁹

¹⁹Bildquelle: <http://america.aljazeera.com/watch/shows/techknow/blog/2014/3/21/digitizing-actorsforilmandtvafterdeathpresentsauniquechallenge.html> - 12.08.2015

Selbst Brad Pitt, welcher seinen rückwärts alternden Charakter Benjamin Button den ganzen Film über verkörperte, alterte mittels Computergrafik und nicht mit Make-Up und Maske. (siehe Abb. 2.5)

Wenn man bedenkt, wie oft Computervisualisierungen überhaupt nicht wahrgenommen werden und beinahe spurlos in ihre Umgebung einfließen, haben sie einen großen Weg hinter sich gebracht und sich somit einen festen Platz in der Filmproduktion erkämpft.

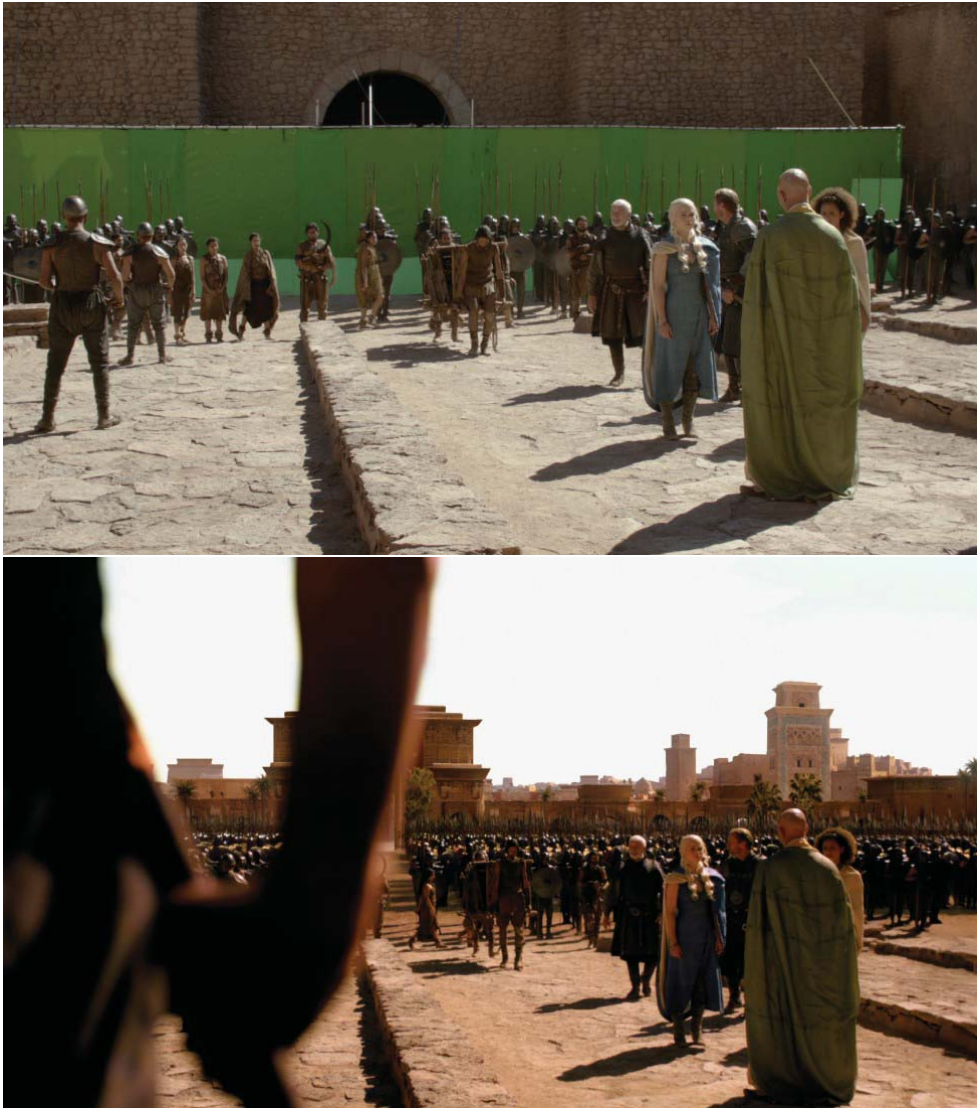


Abbildung 2.6: Oben: Eine Szene aus Game of Thrones ohne CGI | Unten: Die gleiche Szene mit CGI, wie sie die Zuschauer schlussendlich auch zu sehen bekamen²⁰

²⁰Bildquelle: <http://www.spinvm.com/work/game-of-thrones/> - 12.08.2015

1934 wurde schließlich der erste Oscar für „*visuelle Effekte*“ an *Cleopatra* verliehen. Zwar existierte so eine ähnliche Kategorie schon seit den 20er Jahren, allerdings hieß sie bis dahin nur „*Spezialeffekte*“ und war, anders als heute, ausschließlich an Spezialeffekte gerichtet, die am Set gedreht wurden.

Aber neben den fotorealistischen Einsatz von Spezialeffekten in Filmen, ist auch die Popularität der 3D Animationen erheblich gestiegen. Den Anfang machte Pixar 1995 mit *Toy Story*. Dem ersten, komplett am Computer animierten 3D Film, und setzte damit einen Meilenstein. Bis heute befindet sich Pixar, neben Walt Disney Animation Studios, DreamWorks Animation SKG oder Illumination Entertainment, an der Spitze der 3D Animationsstudios.

2.3.2 Medizin

Selbst in der Medizin wurden Computeranimation schon sehr früh verwendet. Sei es zur Darstellung von Wirkmechanismen oder der Funktionsweise medizinischer Geräte, aber auch zur Visualisierung von Operationen, wovon sowohl Ärzte, als auch Patienten profitieren. Erstere können dadurch alle möglichen Eventualitäten simulieren und sich somit besser auf ihre Operationen vorbereiten [KOJC12], während den Patienten somit die Handlungsschritte besser erklärt werden können, was die Beruhigung und das Vertrauen fördern kann. Solch medizinische Animationen sind auch bei der Aus- und Weiterbildung vom ärztlichen Personal sehr hilfreich. So können anatomisch korrekte und lebensechte Simulationen das Verhalten in Notsituationen oder ähnlichem bedeutend verbessern. Im Rahmen einer Studie, gelang es sogar den Teilnehmern, mit Hilfe einer Motion Capture Animation auf dem Handy, eine bessere CPR auszuführen, als mit einem Dispatcher [CPC⁺08]. Selbst in der Kriminaltechnik und der Forensik sind Computeranimation, beispielsweise bei der Rekonstruktion, eine große Hilfe. Mit dem Projekt *Virtopsy* wird die Anwendung moderner Techniken im Rahmen einer virtuellen Autopsie erforscht. [Buc10] Des Weiteren wird in [Wen15] sogar das weltweit erste realistische 3D-Simulationsmodell eines vollständigen menschlichen Herzens vorgestellt, was einen großen Fortschritt in Diagnose, Behandlung und Vorbeugung von Herzerkrankungen bedeutet. Mit solchen Beispielen lässt sich die Zukunft der Medizin sogar revolutionieren.

2.3.3 Architektur

„Architekturdarstellung ist die Kunst, das Mögliche sichtbar zu machen.“²¹ Durch die technische Entwicklung von Echtzeit-3D-Umgebungen ist es für Architekturbüros mittlerweile möglich, ihre Projekte als 3D Modelle oder mittels animierter Kurzfilme zu präsentieren. So können Modelle von Gebäuden oder anderen Bauvorhaben am Computer gestaltet werden, wodurch sie für den Betrachter visuell erfassbar sind. Dabei bietet die 3D Grafik gerade im perspektivischen Bereich eine Menge Möglichkeiten. So können die Licht- und Größenverhältnisse, Materialeigenschaften, Position und Ausrichtung aller Objekte, Blickrichtung oder auch die Position des Betrachters bzw. der Kamera im 3D Programm bestimmt und im Vergleich zu physischen Modellen schnell und einfach verändert werden.

²¹<http://www.architekturdarstellung.info/>

2.4 Zusammenfassung

In diesem Kapitel wurde zunächst die Entwicklung der 3D Grafik in Spielen beschrieben und in wie fern sich die Anfänge von der heutigen Zeit unterscheiden. So existierte bei den ersten Spielen noch keine 3D Umgebung, diese entwickelte sich erst mit der Zeit und trat das erste Mal erfolgreich 1980 im Spiel Battlevision auf. Des Weiteren wurde auf Computer- und speziell Onlinespiele eingegangen, welche keinen Download oder die Installation eines Datenträgers benötigen, so genannte Browsergames. Diese nutzen lediglich den Web-Browser als Schnittstelle. Als nächstes folgte ein Kapitel über die verschiedenen Arten von Schatten in Online Spielen, wobei zwischen dynamischen und statischen Schatten unterschieden wurde. Dabei wurde bei letzterem das Baking Verfahren und das Bemalen der Texturen erläutert. Während nämlich bei dynamischer Beleuchtung die Schattenbildung in Echtzeit berechnet wird, müssen die statischen Schatten auf die Texturen der Objekte gebracht werden, um so später den Eindruck von Licht im Spiel zu vermitteln. Zum Schluss folgte noch eine Aufzählung drei weiterer Einsatzgebiete von Computergrafiken. Im folgenden Kapitel werden die Grundlagen zum Verständnis dieser Arbeit erläutert.

Kapitel 3

Grundlagen

Im vorangehenden Kapitel erfolgte eine Beschreibung zum jetzigen Stand der Technik der 3D Grafik in Spielen im Vergleich zu ihren Anfängen. Sowie den Arten der Schatten in Online Spielen, um aufzuzeigen, welche weiteren Möglichkeiten, abseits der in dieser Arbeit verwendeten Methoden, zur Schattendarstellung in Online Spielen bestehen. Am Ende wurden weitere Einsatzgebiete von Computergrafiken aufgelistet und beschrieben.

Licht und Schatten sind unwiderruflich miteinander verbunden. Das Eine kann ohne das Andere nicht existieren und wo es Licht gibt, dort gibt es auch Schatten und umgekehrt. Deshalb beginnt das folgende Kapitel zuerst mit einer groben Erklärung von unserem Verständnis von Licht und welche wichtige Rolle es in der Visualisierung spielt. Danach werden die Prozesse der 3D Beleuchtung in 3D Programmen erklärt, um zu vermitteln, wie Licht und dessen unverzichtbare Schattenbildung in diesen Programmen dargestellt wird.

3.1 Das Licht in der Physik

Um nachvollziehen zu können, wie Licht in 3D Software dargestellt wird, ist es wichtig zu verstehen, wie es überhaupt funktioniert. Dieses Kapitel wird das Thema Licht in der Physik nur oberflächlich behandeln, um ein allgemeines Verständnis aufzubauen.

Abhängig von den jeweiligen Versuchen, besteht Licht sowohl aus Wellen, als auch aus Teilchen. Newton war bis zuletzt ein Befürworter der Teilchenhypothese, Huygens hingegen unterstützte die Wellentheorie. Doch wie durch die, aufgrund von Albert Einstein, anerkannte Quantentheorie, ist es bewiesen, dass Licht beide Eigenschaften besitzt: es verhält sich wie Wellen, als auch wie Teilchen. Ersteres bestätigten auch die Theorien des Physikers James Clerk Maxwell, die besagen, dass sich Licht als elektromagnetische Welle fortbewegt. Die meisten dieser elektromagnetischen Wellen, wie Radiowellen oder Röntgenstrahlung, sind für unser Auge nicht sichtbar. Die Wellenlänge des für uns sichtbaren Lichts liegt etwa zwischen $385nm$ und $765nm$. Dieser Bereich wird auch als *optisches Spektrum* bezeichnet.¹ (siehe Abb. 3.2)

¹<http://www.seilnacht.com/Lexikon/Licht.htm> - 14.08.2015.

²Bildquelle: <http://www.seilnacht.com/Lexikon/Licht.htm> - 14.08.2015.

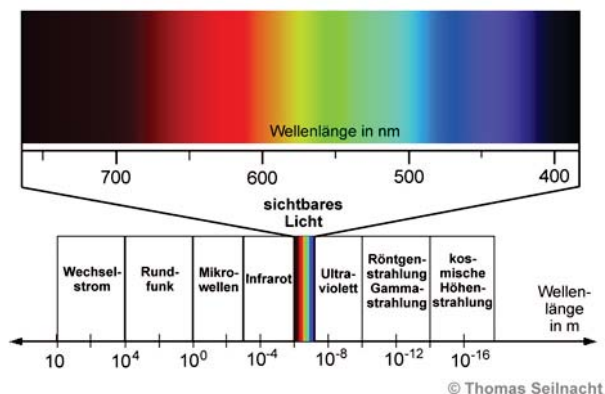


Abbildung 3.1: Optische Spektrum des Lichts.²

3.1.1 Lichtstärke und Schatten

Um die Helligkeit von verschiedenen Lichtquellen vergleichen und benennen zu können, wurde die Lichtstärke I_v mit ihrer Einheit Candela (cd) als physikalische Größe eingeführt. Sie ist eine Basisgröße des internationalen Einheitensystems. Während die Lichtstärke I_v eine Eigenschaft der Lichtquelle ist, gibt die Beleuchtungsstärke E_v an, wie groß der Anteil des Lichts ist, welches auf eine Fläche trifft.

Die Beleuchtungsstärke E_v , die in Lux (lx) angegeben wird, ist umso größer, je größer die Lichtstärke I_v der Lichtquelle ist. Insgesamt gilt:

$$E_v = \frac{I_v}{r^2}$$

Wird ein undurchsichtiger Körper von einer kleinen Lichtquelle beleuchtet, so entsteht auf der von der Lichtquelle abgewandten Seite ein Schlagschatten. Sobald dieser Körper aber von zwei Lichtquellen beleuchtet wird, bilden sich daraus zwei Schattenzonen: Ein Kern- und ein Halbschatten.

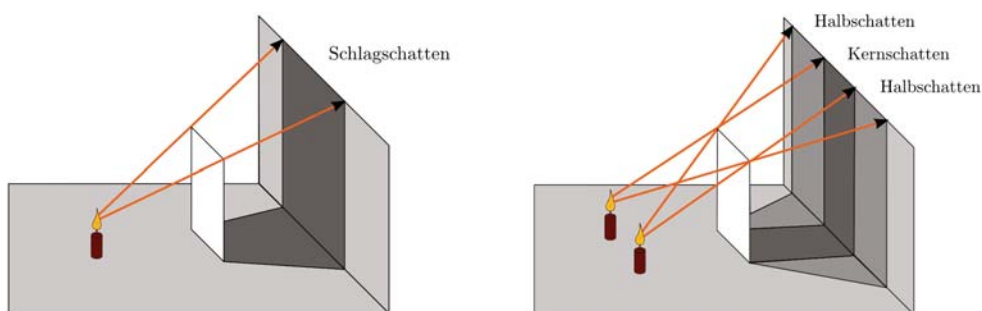


Abbildung 3.2: Links: Entstehung eines Schlagschattens durch eine Lichtquelle | Rechts: Entstehung von Kern- und Halbschatten durch zwei Lichtquellen³

3.1.2 Reflexion, Brechung und Dispersion

Die wohl wichtigsten Eigenschaften von Licht für 3D Software ist dessen Verhalten im sichtbaren Bereich. Die Reflexion, Brechung und Dispersion bilden unter anderem die Grundlage für Ray Tracing.

Das Reflexionsgesetz

Gegenstände, können, abhängig von Art und Material ihrer Oberfläche, einen Teil des auf sie fallenden Lichts zurückwerfen (siehe Abb. 3.3). Dieser Vorgang wird Reflexion genannt. Wird (fast) das gesamte Licht reflektiert, so wird von einer Spiegelung gesprochen.

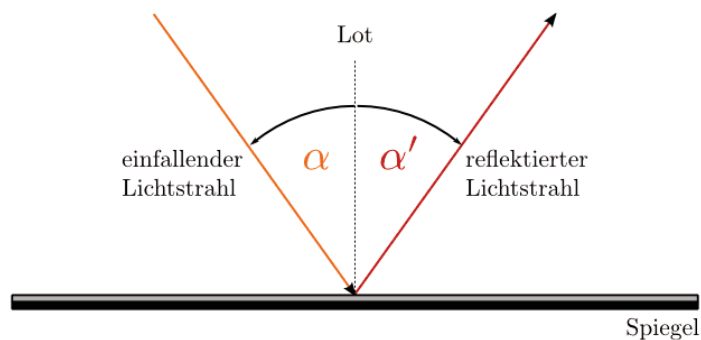


Abbildung 3.3: Reflexion eines Lichtstrahls an einem ebenen Spiegel.⁴

Der Einfallswinkel α und der Reflexionswinkel α' werden ausgehend von der Senkrechten zum Spiegel angegeben. Somit ergibt sich mathematisch $\alpha = \alpha'$.

Das Brechungsgesetz

Tritt Licht auf die Grenzfläche zweier verschiedener Medien, wird es zum Teil reflektiert und zum Teil verändert es seine Richtung (abgesehen vom senkrecht auftreffendem Licht), wie in Abb. 3.4 zu sehen ist. Die Sinus-Werte der Einfalls- und Brechungswinkel $\sin\alpha$ und $\sin\beta$ stehen im gleichen Verhältnis wie die Geschwindigkeiten c_1 und c_2 zueinander. Wenn das Licht von einem Vakuum auf ein transparentes Material trifft, ergibt das Verhältnis der Winkelgrößen einen konstanten, vom Material abhängigen Wert, die Brechzahl n .

$$n = \frac{\sin\alpha}{\sin\beta}$$

³Bildquelle: <http://grund-wissen.de/physik/optik/lichtausbreitung.html> - 14.08.2015.

⁴Bildquelle: <http://grund-wissen.de/physik/optik/lichtreflexion.html> - 14.08.2015.

⁵Bildquelle: <http://grund-wissen.de/physik/optik/lichtbrechung.html> - 14.08.2015.

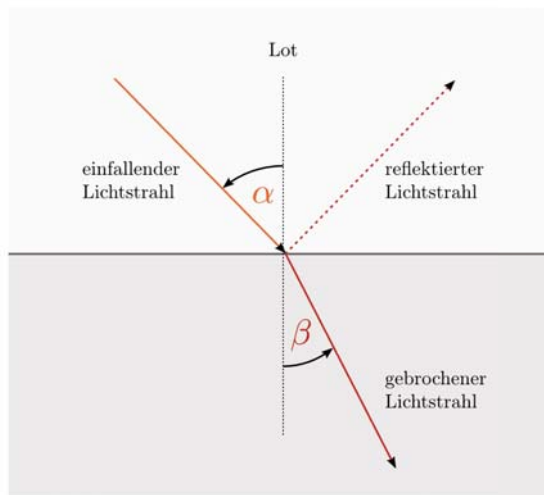


Abbildung 3.4: Lichtbrechung beim Übergang eines Lichtstrahls von einem optisch dünnen in ein optisch dickes Medium.⁵

Die Brechzahl n ist eine für jeden Stoff charakteristische Materialeigenschaft, gibt aber auch gleichzeitig das Verhältnis der Lichtgeschwindigkeit im Vakuum zur Lichtgeschwindigkeit im jeweiligen Stoff an.

Trifft ein Lichtstrahl von einem Stoff mit der Brechzahl n_1 in einen anderen, mit höherer Brechzahl n_2 über, so wird er zur Senkrechten hin gebrochen.

$$\frac{n_2}{n_1} = \frac{\sin\alpha}{\sin\beta}$$

Dispersion

Neben dem Brechungsindex, ist die Ausbreitungsgeschwindigkeit von Licht in einem optischen Medium auch noch von der Lichtfrequenz abhängig. Bei einem Durchgang durch ein Prisma werden die unterschiedlichen Spektralteile unterschiedlich stark gebrochen, so dass sich mittels eines Prismas weißes Licht in seine Farbanteile aufteilen lässt (siehe Abb. 3.5). Kurzwelliges Licht wird dabei generell stärker gebrochen als langwelliges. Dies lässt sich auch in der Natur in Form von Regenbögen beobachten.⁶

⁶<http://grund-wissen.de/physik/optik/index.html> - 14.08.2015.

⁷Bildquelle: <http://grund-wissen.de/physik/optik/wellenoptik.html> - 14.08.2015.

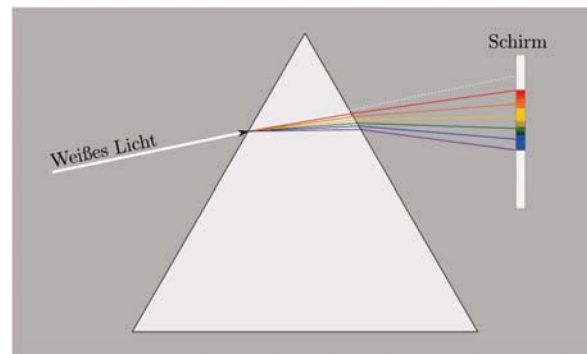


Abbildung 3.5: Lichtdispersion in einem Prisma.⁷

3.2 Grundlagen der 3D Beleuchtung

Nach der Einführung in das Thema Licht in der Physik, beschäftigt sich dieses Kapitel nun mit dem Licht in 3D Software Programmen. Dazu werden erst einmal die verschiedenen Arten von Lichtquellen, die in diesen Programmen anwendbar sind, erläutert, woraufhin die beiden unterschiedlichen Beleuchtungsmodelle beschrieben werden.

3.2.1 Lichtquellentypen

Grundsätzlich können folgende Typen von Lichtquellen in 3D-Programmen eingesetzt werden:

Point Light (Punktlicht)

Point Light, auf Deutsch Punktlicht oder omnidirektionales Licht genannt, simuliert Lichtstrahlen, welche von einem infinitesimalen Punkt im Raum ausgehen. Es lässt das Licht gleichmäßig in alle Richtungen aussenden, genau wie beispielsweise bei einer Glühbirne. Solch eine Lichtquelle, die ihr Licht vollkommen gleichmäßig durch den Raum verteilt, ist kaum in der Natur zu finden. Wer von daher eine Glühbirne mit ungleichmäßigem Licht in einer Szene simulieren möchte, muss dementsprechend ungleichmäßige Strahlenschemas einsetzen. [Bir00]

Spot Light (Spotlicht)

Das Spotlicht ist wohl eines der beliebtesten Lichtquellen, was daraus resultiert, dass sich mit ihm Objekte gezielt beleuchten lassen. Es simuliert Licht, welches von einem bestimmten Punkt aus einen Lichtkegel ausstrahlt. Mit der Möglichkeit, das Spotlicht mit einem Zielobjekt zu verknüpfen, lassen sich damit beispielsweise auch Taschenlampen oder Autoscheinwerfer simulieren. Sie besitzen zudem auch so viele Optionen und Kontrollmöglichkeiten, wie man sie bei keinem anderen Lichttyp findet. So lassen sich die Breite des Lichtkegels

oder die Winkelöffnung beliebig einstellen. Auch die Lichtintensität zum Rand des Strahls hin lässt sich variieren. Wenn der Rand des Lichtkegels sehr weich eingestellt ist, kann die Quelle des Lichts nicht mehr eindeutig identifiziert werden. Dadurch kann ein ganzer Raum mit Hilfe dieser Spotlights erhellt werden. [Bir00]

Area Light (Bereichslicht)

Bereichslichter können, im Vergleich zu Punktlichtern, skaliert werden und somit ihre Intensität verändern. Für eine genauere Darstellung von Lichtverhältnissen können demnach Bereichslichter angewendet werden. Somit haben nicht alle Lichtstrahlen den gleichen Ursprung. Ist die Skalierung dieser Lichtquelle sehr klein, ähnelt deren Licht- und Schattensetzung der, einer Punktlichtquelle. Wird sie hingegen größer skaliert, erzeugt sie ein mattes Licht und sanfte Schatten. Mit dieser Lichtquelle kann eine sehr hohe Licht- und Schattenqualität erzielt werden, wodurch sie bei realistischen Renderings bevorzugt eingesetzt wird. Dabei wird zwischen sphärischem, planem und linearem Bereichslicht unterschieden. Bei einem sphärischen Bereichslicht geht das Licht aus einem kugelförmigen Bereich aus und strahlt gleichmäßig omnidirektional. Ein rechteckiges, also planes Bereichslicht, simuliert hingegen ein Lichtpaneel und ist die häufigste Form dieser Lichtquelle. Sie lässt sich beliebig in der Höhe und Länge skalieren. Auch die Drehung und Ausrichtung der Lichtquelle beeinflusst, im Gegensatz zum sphärischen Bereichslicht, den Lichtfall. Lineares Bereichslicht, die letzte Form dieser Lichtquelle, gleicht dabei einer Neonröhre und kann nur in eine Richtung skaliert werden. Sie sendet entlang ihrer Länge weiches Licht aus, als an den Enden und ist durch die wesentliche geringere Renderzeit unter anderem gut zur dezenten Ausleuchtung einer Szene geeignet. [Bir00]

Infinite Light (Gerichtetes Licht)

Bei dieser Lichtquelle spielt die Entfernung zu den beleuchteten Objekten keine Rolle. Allein die Positionierung und der Winkel ist entscheidend. Oftmals wird es auch Sonnenlicht genannt und auch hauptsächlich als dieses eingesetzt. Mit dem gerichtetem Licht können sogar nahezu parallele Schatten erzeugt werden, was nur bei unendlich weit entfernten Lichtquellen geschieht. Allerdings lässt sich diese Lichtquelle nicht so einfach auf eine bestimmte Stelle ausrichten, wie dies beim Punkt-, oder Spotlicht der Fall ist. Es sollte von daher eher dazu verwendet werden, sehr große Bereiche mit ambienten bzw. atmosphärischen Licht zu füllen. [Bir00]

⁸Bildquelle: <http://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2015/ENU/Maya/files/ENU-Maya-Direct-light-sources-htm.html> - 17.09.2015.



Abbildung 3.6: Die unterschiedlichen Schattenbildungen bei den jeweiligen Lichtquellen.⁸

3.2.2 Beleuchtungsmodelle

Beleuchtungsmodelle beschreiben die Reflektionseigenschaften von Materialien und ermitteln dadurch ihren Farbwert. Dabei können beim Auftreffen von Licht auf ein Objekt verschiedene Lichteinteile unterschiedlich reflektiert werden. Lokale Beleuchtungsmodelle berechnen Lichtreflektionen von idealisierten Oberflächen, wobei nur die direkte Licht-Objekt Interaktion berücksichtigt wird.⁹ Globale Beleuchtungsmodelle berechnen währenddessen sowohl das Licht ausgedehnter Lichtquellen, als auch das von anderen Oberflächen reflektierte oder transmittierte Licht, dabei beziehen sie das einfallende Licht aus allen Raumrichtungen mit ein. Dafür wird die Farbe an jedem Punkt einer Oberfläche mittels einer Integralformel berechnet, was zu einem höheren Rechenaufwand führt. Dies ist der Grund, dass in der interaktiven 3D-Computergrafik bisher ausschließlich ein lokales Beleuchtungsmodell verwendet wird. [NFHS12]

Lokale Beleuchtungsmodelle

Sie sind, aufgrund dessen dass sie nur die Lichtquellen und das eigene Material in die Berechnung der einzelnen Punkte einer Oberfläche mit einbeziehen, weitaus schneller zu berechnen als globale Beleuchtungsmodelle. Grundsätzlich wird zwischen dem Lambert- und Phong-Beleuchtungsmodell unterschieden. Das Lambert-Beleuchtungsmodell wird folgendermaßen berechnet:

$$I = I_{max} \cos(\theta)$$

Die maximale Lichtstärke wird dabei immer senkrecht zur Oberfläche reflektiert. Dieses Modell simuliert die Beleuchtung ideal diffus reflektierender Oberflächen. Im Phong-Modell wird zusätzlich zum Lambert-Modell noch eine Reflexion mit einbezogen und besteht aus drei verschiedenen Komponenten: Ambient (umgehend), Diffuse (ausbreitend) und Specular (spiegelnd). Es berechnet die farbliche Darstellung räumlicher Objekte und wurde nach seinem Entwickler Phong-Tuong benannt, welcher dieses Verfahren erstmals 1975 vorstellte.[BT75]

Globale Beleuchtungsmodelle

Das „*Ray Tracing Verfahren*“ beschränkt sich auf ideal glatte Oberflächen. Dabei wird der Lichtstrahl für jedes zu berechnende Pixel vom Augenpunkt rückwärts durch dieses Pixel in die Szene zurück verfolgt, bis es auf ein Objekt trifft. Dies wird Primärstrahl genannt. Dafür wird als erstes ein lokales Beleuchtungsmodell für den Punkt berechnet, in dem nur Punktlichtquellen und die spekularen und diffusen Anteile seiner Oberfläche berücksichtigt werden. Im Anschluss wird dann das globale Beleuchtungssystem für ideal glatte Oberflächen berechnet. Daraufhin werden die Sekundärstrahlen berechnet, in dem die Richtung eines reflektierten und eines gebrochenen Strahls berechnet wird um herauszufinden, ob die Lichtstrahlen anderer Oberflächen einen Beitrag zur Beleuchtung liefern. Dann wird das globale Beleuchtungssystem auf die sekundären Oberflächenpunkte angewendet, wodurch ein rekursives Strahlverfolgungsverfahren entsteht. Dadurch erhält man korrekte Objektspiegelungen und Schatten. Der Nachteil dabei ist, dass Beleuchtungsübergänge unnatürlich hart

⁹<http://computergrafik.informatiker-wissen.de/beleuchtungsrechnung.html>

erscheinen.

Bei der „*Radiosity Methode*“ werden zum einen die einzelnen Strahlen verfolgt, zum anderen wird auch der Strahlungsaustausch zwischen ideal diffusen Oberflächenstücken berechnet. Von jedem Flächenelement in einer Szene geht ein konstanter Lichtstrom aus. Dieser setzt sich aus einem emittierten und einem reflektierten Lichtstrom zusammen. Durch die Beschränkung auf ideal diffuse Reflexionen können mit dieser Methode keine spiegelnden Oberflächeneigenschaften dargestellt werden. [NFHS12]

3.3 Zusammenfassung

Dieses Kapitel begann mit dem Aufbau eines Grundverständnisses von Licht und wie es sich physikalisch verhält. Dabei wurden nur die, für die 3D Grafik wichtigen, Eigenschaften wie die Lichtstärke und Schatten, Reflexion, Brechung und Dispersion behandelt. Danach folgten die Grundlagen der 3D Beleuchtung, angefangen mit den grundsätzlichen Lichtquellentypen in 3D-Programmen: Point, Spot, Ambient und Infinite Light. Als nächstes wurden die Beleuchtungsmodelle beschrieben, wobei zwischen lokalen und globalen Beleuchtungsmodellen unterschieden wurde. Während bei lokalen Beleuchtungsmodellen lediglich direkte Licht-Objekt Interaktion berücksichtigt werden, berechnen globale Beleuchtungsmodelle auch das von anderen Oberflächen reflektierte oder transmittierte Licht. Ab dem nächsten Kapitel beginnt die Methodik dieser Arbeit, angefangen mit der Konzeptionierung.

Kapitel 4

Konzeptionierung

Im vorherigen Kapitel wurden die theoretischen Grundlagen von Licht behandelt, um ein grundlegendes Verständnis dafür aufzubauen, wie es sich optisch verhält und in wie fern dieses Wissen für die Erstellung von Licht in 3D Software genutzt wurde.

In diesem Kapitel wird ein Konzept erarbeitet, um einen optimalen Vergleich beider Methoden zur Licht- und Schattensetzung in 3D Umgebung erstellen zu können. Bei den Methoden handelt es sich zum einen um ein automatisches Verfahren, welches Licht und Schatten in Autodesk Maya berechnet und auf die Texturen backt, und zum anderen um ein manuelles Verfahren, in welchem Licht und Schatten nach eigenem Ermessen in Adobe Photoshop eingezeichnet werden. Diese Methoden werden anhand eines Referenzitems miteinander verglichen. Im darauffolgenden Kapitel wird dann die Umsetzung dieses Konzepts erfolgen und schriftlich dokumentiert.

4.1 Auswahl der zu untersuchenden Methoden

Wie bereits im Kapitel 1.2 Problemstellung beschrieben, kann das dynamische Berechnen von Licht und Schatten in Browsergames schnell zu Problemen bei der flüssigen Darstellung der Spiele kommen. Aufgrund dessen wird in dieser Arbeit die statische Darstellung von Schatten behandelt. Dies kann mit Hilfe von Texturen erzielt werden. Auf diese werden die Licht- und Schatteninformationen nicht während des Spielens, sondern schon vorher angewendet. Daraufhin werden die Objekte mit den Texturen versehen und die Nutzer erhalten den Eindruck, dass sich tatsächliches Licht im Spiel befindet.

Bei den untersuchten Methoden handelt es sich jeweils um ein automatisiertes und um ein manuelles Verfahren um Licht- und Schatteninformationen auf die Texturen zu bringen.

4.1.1 Automatisiertes Verfahren

Beim automatisiertem Verfahren werden mittels Baking die Licht- und Schatteninformationen auf eine Textur „gebacken“. Diese Informationen werden in einem 3D Programm automatisiert berechnet und rausgerendert. Hierfür wird vorab eine 2D Textur in Photoshop erstellt, allerdings ohne jegliche Schatten. Im 3D Programm wird dem dazugehörigem Objekt dann die fast fertige Textur zugeteilt. Daraufhin werden Lichter in die Szene gesetzt, so dass Schatten entstehen. Es können verschiedene Lichter in unterschiedlichen Positionen gesetzt werden, um das gewünschte Ergebnis der Schattierung zu erzielen. Auch die Einstellungen der jeweiligen Lichter können je nach Bedarf verändert werden. Sobald das Licht Setup fertig positioniert und eingestellt ist, kann in der Maya Szene im Viewport auch schon das spätere Ergebnis betrachtet werden. Denn so wie die Schatten in dieser Szene fallen, so werden sie auch auf die Textur gebacken. Um eine Vorschau der Lichter und deren Schattenwurf im Viewport zu erhalten (siehe Abb. 4.1), drückt man einfach im Viewport auf „Lights“ → „Use all Lights“, oder auch die Taste 7.

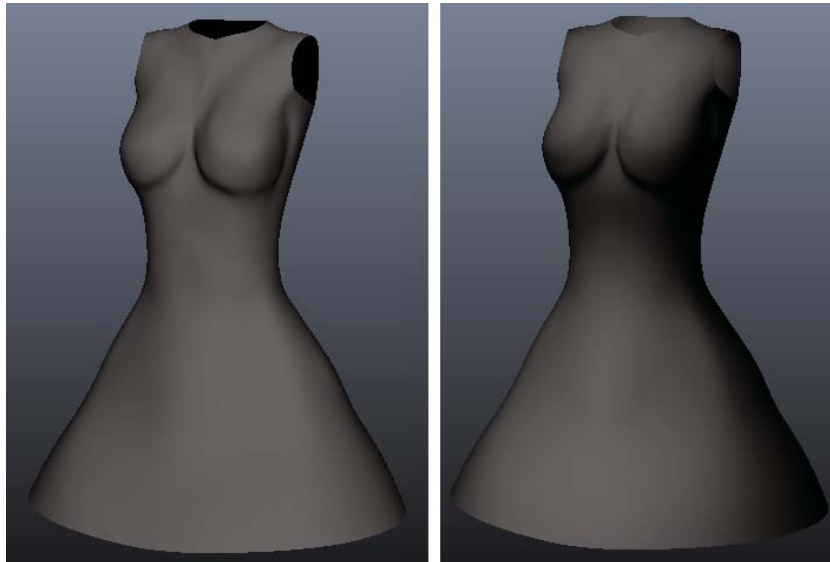


Abbildung 4.1: Screenshot aus dem Viewport. Links: Default Lights | Rechts: Use all Lights

Nachdem die Licht- und Schattenbildung die gewünschten Ergebnisse erzielen, lassen sich diese auch schon direkt auf die Textur backen. Für Mental Ray wird dafür unter „Rendering“ → „Lighting/Shading“ → „Batch Bake (mental ray)“ ausgewählt. Zuvor muss dem Mesh aber noch das gewünschte Bake Set zugewiesen werden, in diesem Fall Texture Bake Set. Der Button dazu befindet sich direkt oberhalb des Batch Bake Buttons. (siehe Abb. 4.2) An dieser Stelle werden die einzelnen Optionen der beiden Fenster erklärt und wieso genau diese für die Umsetzung ausgewählt wurden:

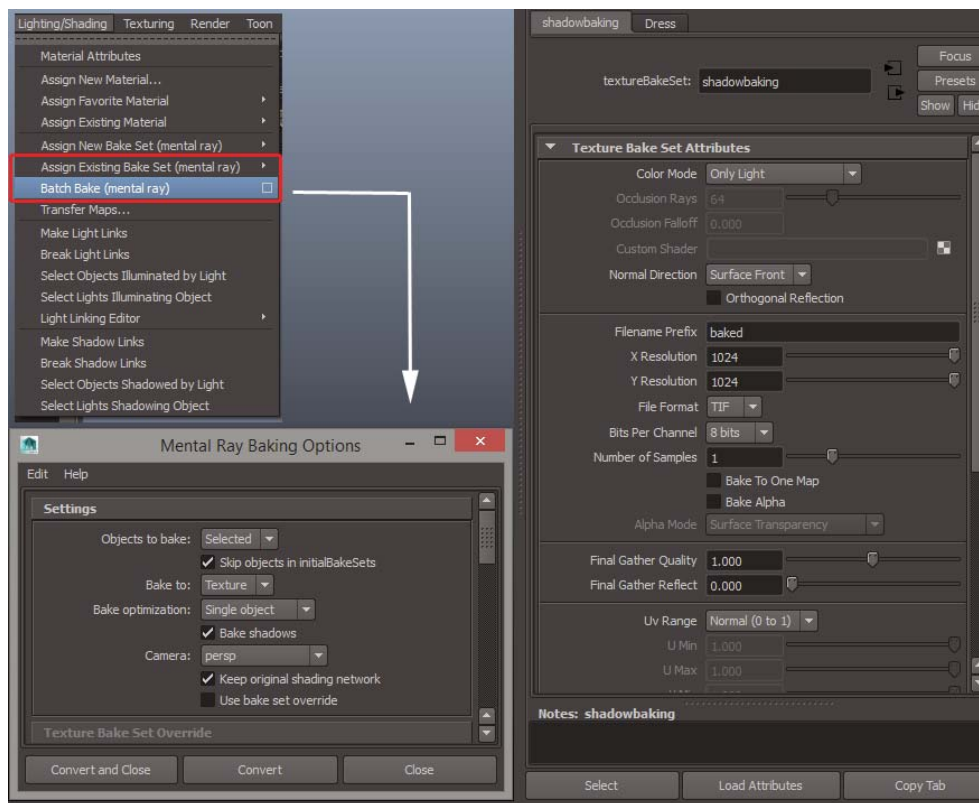


Abbildung 4.2: Screenshot zum Baking Verfahren

- Color Mode: Only Light**
 Zur Auswahl stehen noch Light and Color, Global Illumination und Occlusion. Da lediglich die Licht- und Schatteninformationen gebacken werden sollen, und eben nicht die Farbe des Shaders oder andere Informationen, wurde der Color Mode auf Only Light gestellt.
- Normal Direction: Surface front**
 Da die Normals des Items alle vom Item aus gerichtet sind, wird hier Surface front eingestellt. Wären die Normals nach innen gerichtet, müsste hier die Option Surface Back eingestellt werden. Eine weitere Möglichkeit ist noch Face Camera, in diesem Fall würden die Normals in Richtung der gerichteten Kamera zeigen.
- Orthogonal Reflection**
 Dies wurde nicht ausgewählt, da im Item selbst keine Reflektionen entstehen (sollen) bzw. zu sehen sind.
- Number of Samples: 1**
 Umso größer die Anzahl der Samples, umso besser die Qualität der gebackenen Schatten, wobei 4 der maximale Wert ist. Bei der Verfahrensentwicklung wurde jedoch

erkannt, dass die Qualität der Schatten schon bei dem voreingestellten Wert 1 vollkommen zufriedenstellend war und kein Aliasing-Effekt auftrat. Somit wurde diese Zahl nicht weiter erhöht.

- **Bake to one Map**
Wenn mehrere Objekte ausgewählt sind, jedoch nur eine gebackene Textur aus allen entstehen soll, kann diese Option ausgewählt werden. Allerdings ist es hierbei sehr wichtig, dass sich die UV Layouts der einzelnen Objekte nicht überlagern. Da in diesem Fall allerdings nur ein Objekt ausgewählt wurde, ist das Aktivieren dieser Option nicht notwendig.
- **Bake Alpha**
Falls Transparenz gebackenen werden soll, kann diese Option ausgewählt werden, woraufhin der Alpha Mode eingestellt werden kann.
- **Final Gather Quality: 1**
Ist der Wert höher als 0, wird zusätzlich noch eine Anzahl von Final Gather Points gebacken. Je höher der Wert, umso höher die Anzahl der gebackenen FG Points. Dennoch sollte diese Option nicht dafür benutzt werden, die Genauigkeit der gebackenen FG Points zu verbessern, dafür sollten die Rendereinstellungen genutzt werden. Bei der Verfahrensentwicklung stellte sich raus, dass ein höherer Wert bei sehr geringer Qualitätsverbesserung viel längere Renderzeiten brachte, wodurch der Wert 1 gelassen wurde.
- **Final Gather Reflect: 0**
Ein Wert höher als 0 ist nur nötig, wenn das Objekt Reflexionen beinhaltet, was beim Referenzitem nicht der Fall ist.
- **Fill Texture Seams: 3**
Dieser Wert brachte bei der Verfahrensentwicklung die besten Ergebnisse. Falls ein UV Layout Schnittkanten besitzt, können diese beim Backen als schwarze Streifen angezeigt werden, um dieses Problem zu vermeiden, kann der Wert erhöht werden.

4.1.2 Manuelles Verfahren

Beim manuellem Verfahren werden Licht und Schatten, anders als beim automatisiertem Verfahren, nicht berechnet, sondern frei per Hand in Photoshop eingezeichnet. Mit Hilfe von verschiedenen Pinseln, dessen Farbstärke und Dicke frei eingestellt werden können, lassen sich Schatten auf die Textur bringen. Durch mehrere Pinsel und Ebenen mit jeweils unterschiedlichen Einstellungen können so die Schatten beliebig auf die Textur eingezeichnet werden.

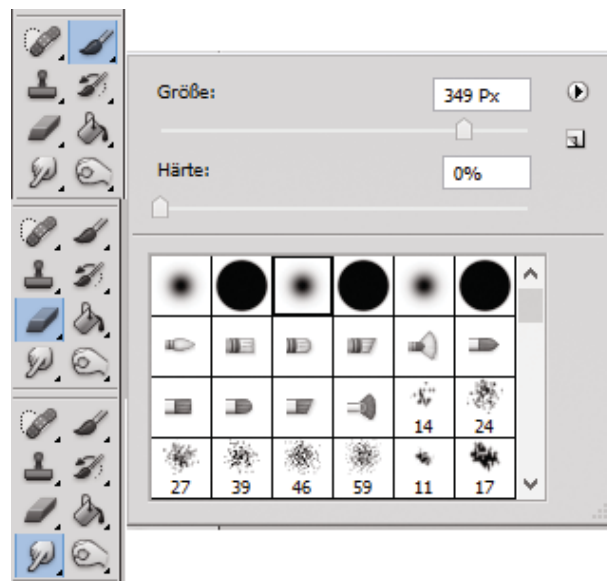


Abbildung 4.3: Die wichtigsten Werkzeuge der manuellen Methode

Wie in Abb. 4.3 zu sehen, sind die drei wichtigsten Werkzeuge bei der manuellen Methode das Pinsel Werkzeug, Radiergummi Werkzeug und Wischfinger Werkzeug. Mit ersterem lassen sich die Schatten einzeichnen. Dabei kann der Härtegrad, die Größe und auch die Art des Pinsels ausgewählt werden. Photoshop verfügt über eine überschaubare Anzahl von Pinseln, die sich aber beliebig erweitern lassen, in dem man Pinsel aus dem Internet runter lädt oder selbst welche erstellt und abspeichert. So führen die verschiedenen Einstellungen zu unterschiedlichen Ergebnissen, wie in der unteren Abb. 4.4 zu sehen.



Abbildung 4.4: Von Links nach Rechts: Härtegrad 100% | Härtegrad 0% | Jeweils unterschiedliche Pinsel

4. KONZEPTIONIERUNG

So wie sich beim Pinsel die Größe und Härte einstellen lässt, so ist dies auch beim Radiergummi Werkzeug möglich. In der Abb. 4.5 wurde mit Hilfe des Pinsel Werkzeugs ein schwarzer Punkt eingezeichnet. In der Mitte wurde an vier Seiten das Radiergummi Werkzeug mit einer Härte von 100% angewendet und im rechten Bild wurde die Härte des Radiergummis auf 0% gestellt. Wie zu erkennen ist, unterscheiden sich die Ergebnisse stark von einander. Für einen weichen Verlauf sollte demnach die Härte, sowohl beim Pinsel, als auch beim Radiergummi Werkzeug auf 0% eingestellt werden.

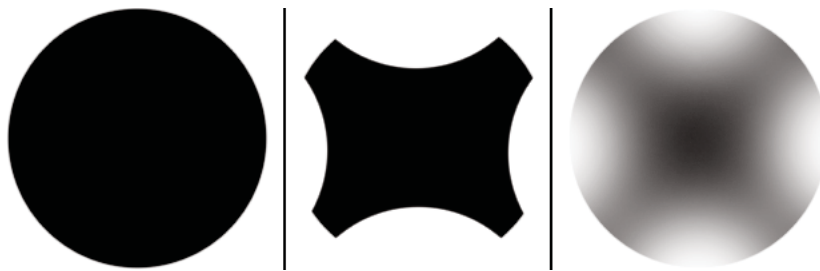


Abbildung 4.5: Links: Ohne Radiergummi | Mitte: Härtegrad 100% | Rechts: Härtegrad 0%

Zuletzt ist noch das Wischfinger Werkzeug eine große Hilfe beim Formen der Schatten. So lassen sich mit dem Pinsel Werkzeug die Schatten grob an die gewünschten Stellen einzeichnen, doch mit Hilfe des Wischfinger Werkzeugs lassen sich diese auch formen. Auch hier kann die Größe und Härte des Werkzeugs eingestellt werden, wodurch sanfte Übergänge entstehen. (siehe Abb. 4.6)

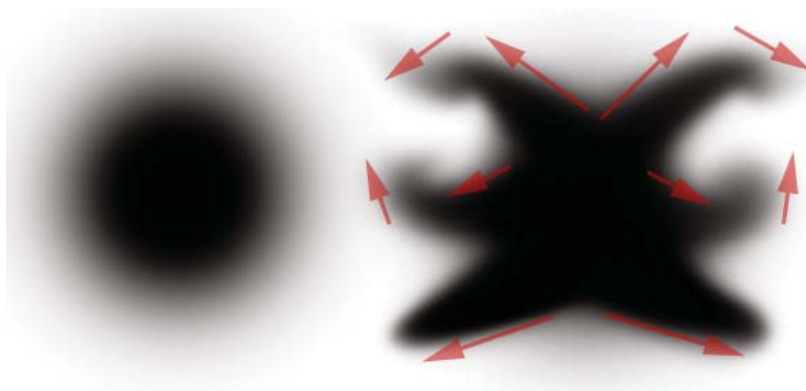


Abbildung 4.6: Links: Ohne Wischfinger | Rechts: Pfeile zeigen die Richtung, in welche das Werkzeug eingesetzt wurde.

4.1. Auswahl der zu untersuchenden Methoden

Als Hilfestellung dient dabei zusätzlich noch ein Snapshot des UV Layouts, welcher schnell im 3D Programm erstellt und abgespeichert werden kann, wodurch die Formen und Schnittkanten der Objekte kenntlich gemacht werden. (siehe Abb. 4.7)

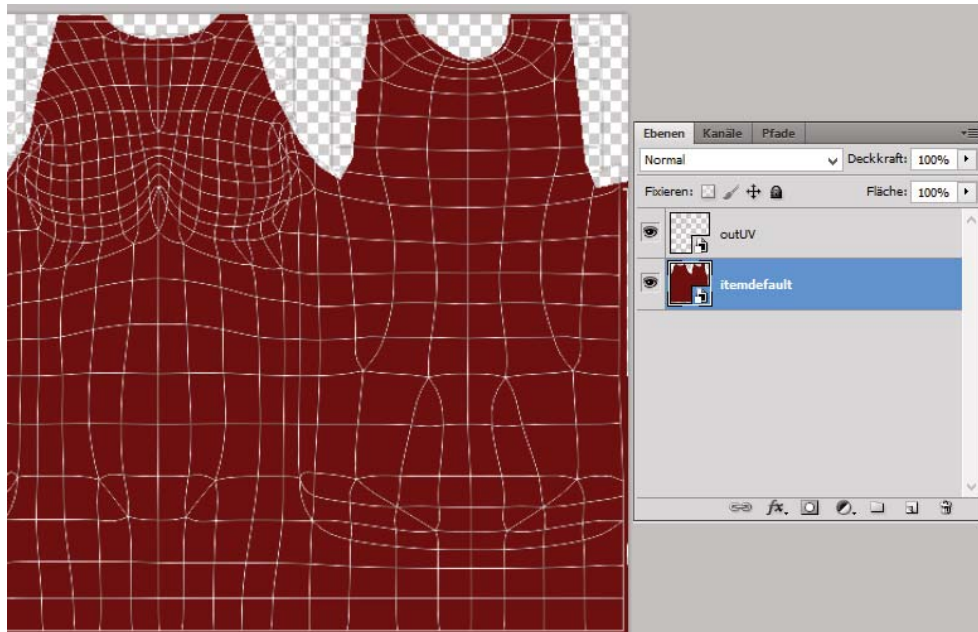


Abbildung 4.7: Mit Hilfe des UV Layouts lassen sich die Formen des Meshs in Photoshop erkennen und die Schatten somit besser positionieren

4.2 Auswahl des Referenzitems

Für den Vergleich beider Methoden wurde ein Kleidungsstück als Referenz ausgewählt. Es handelt sich dabei um ein Kleid am Oberkörper eines weiblichen Avatars. Aufgrund der Brust und des daraus resultierenden Schattenwurfes, eignet sich ein Item für einen weiblichen Avatar besser zum Vergleich der Schattenbildungen, als ein Item, welches für einen männlichen Avatar gebaut wurde. Und da die beiden Methoden auch nur an einem Item durchgeführt werden, wurde ein Kleid ausgewählt, da es im Vergleich zu einem Oberteil oder Hose ein größeres Mesh mit größerer Textur ist und somit auch eine größere Fläche zur Licht- und Schattenbildung besitzt. Auch der ausfallende Rock passt gut, um die Möglichkeiten zur Erweiterung der manuellen Methode aufzuzeigen, wie später im Kapitel 6. Ergebnis unter Möglichkeiten aufgezeigt. Das fertige Mesh wurde von der Firma SMeet GmbH für diese Bachelorthesis zur Verfügung gestellt. In Abb. 4.8 ist zum Einen das Wireframe Rendering des Items mit der Anzahl der Polys (502) und Triangles (950) zu sehen, zum Anderen zeigt die Abb. 4.9 noch das dazugehörige UV Layout und die Textur "itemdefault.png".

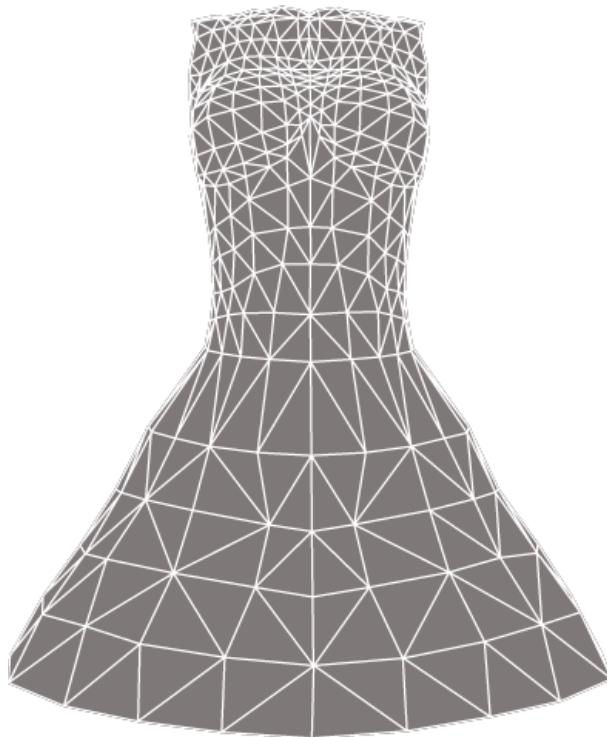


Abbildung 4.8: Wireframe Rendering vom Mesh des Items
Polys: 502
Triangles: 950

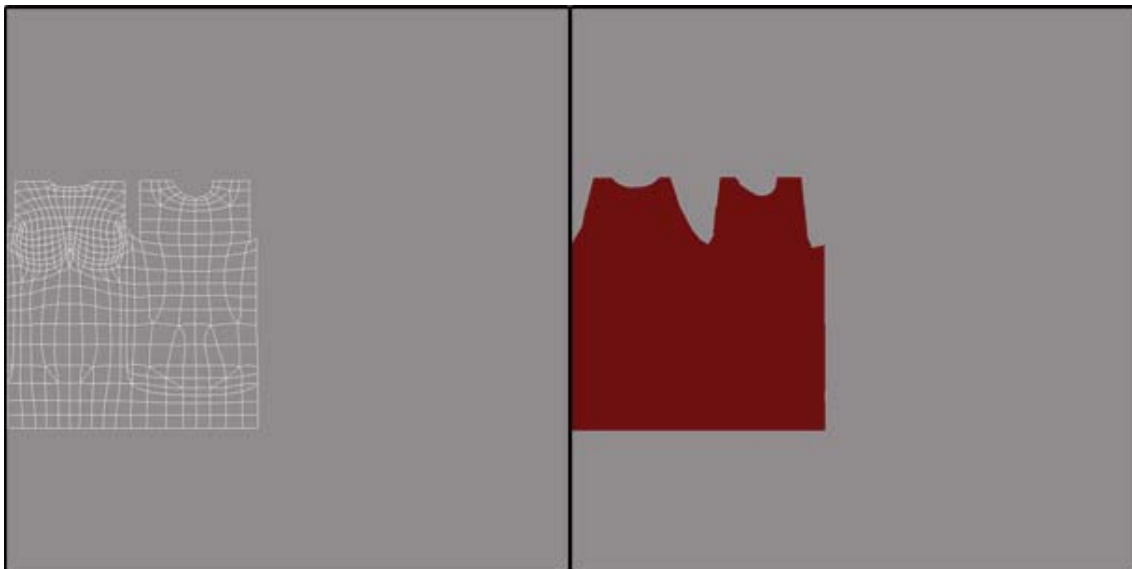


Abbildung 4.9: Links: UV Layout vom Item | Rechts: Textur vom Item

4.3 Anforderungen

In diesem Punkt werden die Anforderungen an die verwendete Szene und Methoden definiert. An diese Anforderungen wird sich während der Verfahrensentwicklung und Umsetzung gehalten.

- **Licht Setup**

Für das automatisierte Verfahren braucht es, im Gegensatz zum manuellen Verfahren, schon in der 3D Szene Lichtquellen. Die Lichtquellen müssen und ihre jeweiligen Einstellungen müssen schriftlich dokumentiert werden. Außerdem ist die Lichtsetzung anhand einer Grafik festzuhalten und die benutzten Lichter kenntlich gemacht werden. Des Weiteren wird ein Licht Setup für den optischen Vergleich der Renderings beider Methoden benötigt. Dieses wird ebenfalls dokumentiert und muss bei beiden Verfahren identisch und möglichst einfach sein, da es sich hierbei nur um die Ausleuchtung der Szenen mit den unterschiedlichen Texturen handeln soll. Dadurch können beide Methoden nicht nur anhand der Texturen, sondern auch anhand von gerenderten Bildern optisch verglichen werden.

- **Rendereinstellungen**

Die Rendereinstellungen, welche beim automatisiertem Verfahren vorgenommen werden, sind zu dokumentieren. Ziel ist es, ein zufriedenstellendes Gleichgewicht zwischen Qualität der gebackenen Textur und der Renderzeit zu finden. Ebenso sind die Einstellungen, wie sie zum Vergleich der gerenderten Bilder bei beiden Methoden angewendet werden, schriftlich festzuhalten und dürfen sich nicht voneinander unterscheiden. Nur mit den gleichen Einstellungen zum Rendern der Szenen und Items bei beiden Verfahren lassen sich optische Vergleiche ziehen.

- **Vergleich der beiden Methoden**

Die Umsetzung der beiden Methoden ist schriftlich zu dokumentieren. Der Vergleich beider Methoden erfolgt jeweils im Zeitaufwand und der Ästhetik. Somit ist der Zeitaufwand bei jeder Methode schriftlich festzuhalten. Gemessen wird die Zeit in Minuten. Auch die Anzahl der Arbeitsschritte ist festzuhalten und wird gegenübergestellt.

Der visuelle Vergleich erfolgt anhand einer Gegenüberstellung der Texturen und den gerenderten Bildern. Die optischen Unterschiede sollen dabei in den Texturen kenntlich gemacht werden. Hierbei ist zu dokumentieren, wo Schatten entstehen, ob es ein weicher oder harter Schattenwurf ist und wie stark bzw. dunkel die Schatten sind. Gerade im Brustbereich sollen die Schatten dokumentiert und deren Unterschiede kenntlich gemacht werden, da besonders an dieser Stelle ein Schattenwurf zu erwarten ist. Als Referenz dient noch eine weitere Textur, welche erst nach der Umsetzung beider Methoden eingesetzt wird: die Hauttextur des Avatars. Mit ihr können die Ergebnisse beider Methoden gegenübergestellt werden um zu erkennen, welche der Methoden einen natürlicheren Schattenwurf im Brustbereich aufzeigt.

- **Gewünschte Licht- und Schattenbildung**

Ziel bei beiden Methoden sind möglichst weiche Lichtpunkte und Schatten, welche die Grundrisse des Modells unterstreichen sollen. Dafür soll beim Referenzitem der Eindruck von oben befindlichen Lichtquellen erweckt werden, welche sowohl die Vorder- als auch die Rückseite des Items erhellen. Somit sollen an den Seiten des Items weiche Schatten geworfen und der Brustbereich durch darunter befindliche Schattenbildung betont werden.

4.4 Vergleichbarkeit

Um einen unverfälschten Vergleich aufstellen zu können, müssen die Rahmenbedingungen für beide Methoden festgelegt und dürfen während der Umsetzung nicht verändert werden.

- **Vergleichbarkeit der Texturen**

Um die Endergebnisse der Texturen und deren optische Unterschiede vergleichen zu können, müssen diese bei beiden Methoden die gleiche Auflösung haben. Alle Texturen für das Item besitzen demnach eine Auflösung von 1024x1024px und 72dpi. Das Item wird mit einem Lambert Shader versehen, während als Farbe die Datei der Textur fungiert. Der Shader selbst besitzt somit keinerlei eigene Reflexionen der Lichtquellen und die Licht- und Schattenbildung lässt sich besser vergleichen.

- **Vergleichbarkeit der Renderings**

Neben den Vergleich der Texturen, wird das Objekt unter identischen Rahmenbedingungen gerendert, um auch noch einen Vergleich der Renderings aufstellen zu können. Dafür wird die Umgebung eines Studios nachgestellt. Das Studio wird mit Hilfe eines MEL Skripts von Fxfox.net¹ erstellt, welches durch Betätigen der Buttons die nötigen Objekte, Lichter und Rendereinstellungen setzt. (siehe Abb. 4.10) Das Studio besteht dabei aus einer Sphere (Kugel), einem Boden und Wand und zwei Area Lights, wie in Abb. 4.11 zu sehen.

Das Objekt wird jeweils mit den fertigen Texturen beider Methoden versehen und daraufhin aus drei Kameraperspektiven gerendert. Die Licht- und Rendereinstellungen, sowie die Größe und Position des Objekts, als auch die Kameraeinstellungen bleiben bei jedem Rendering identisch. Lediglich die Texturen werden verändert. Somit lassen sich auch die optischen Unterschiede anhand von jeweils drei Renderings dokumentieren.

¹<http://www.fxfox.net/fxfox-maya-real-studio-2-is-available-now/>

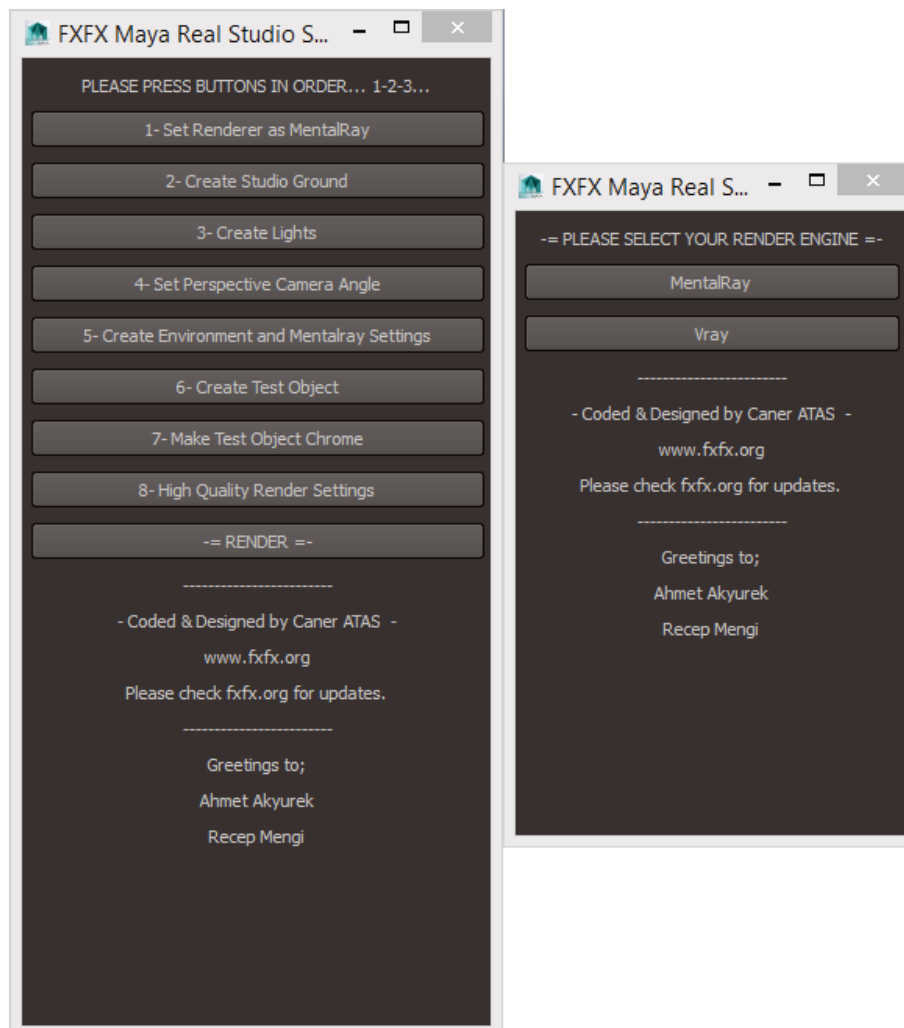


Abbildung 4.10: Skript zum Rendern der Ergebnisse beim Referenzitem

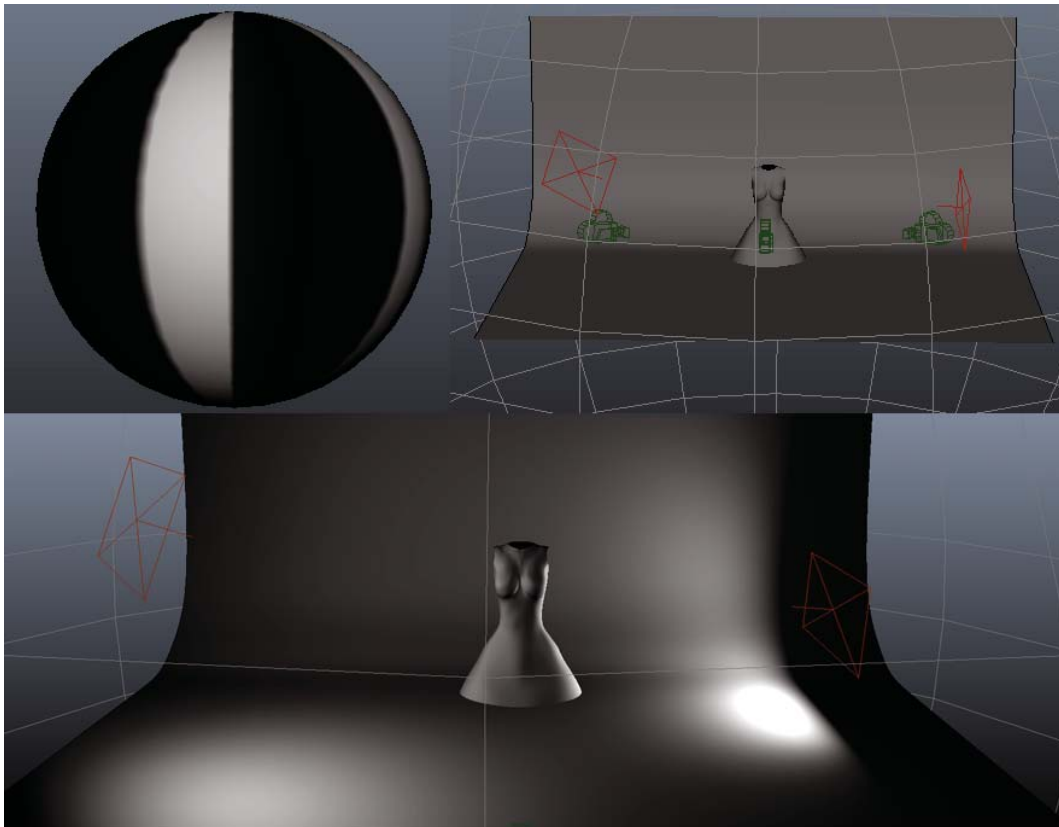


Abbildung 4.11: Screenshots des Studios zum Rendern der Ergebnisse. Oben Links: Von außen betrachtet | Oben Rechts: Von innen betrachtet | Unten: Use all Lights eingeschaltet, so dass die Lichter zu sehen sind.

4.5 Zusammenfassung

Zu Beginn des Kapitels wurde die Auswahl der Methoden beschrieben, woraufhin die Methoden selbst erläutert wurden. Dabei handelt es sich zum einen um eine automatisierte, zum anderen um eine manuelle Methode Licht und Schatten auf eine Textur zu projizieren um diese daraufhin einem Objekt zuzuweisen und somit den Eindruck von Beleuchtung dieses Objektes zu erwecken. Das automatisierte Verfahren beinhaltet das Backen der Lichter und deren Schattenbildungen aus der Maya Szene heraus, während beim manuellen Verfahren diese Informationen per Hand in Photoshop eingezeichnet werden. Danach wurde die Auswahl des Referenzitems, an dem beide Methoden angewendet werden sollen, beschrieben. Bei dem Item handelt es sich um ein Kleid für einen weiblichen Avatar, welches von der Firma SMeet GmbH für diese Arbeit zur Verfügung gestellt wurde und aus 502 Polygonen besteht. Um einen Vergleich zwischen beiden Methoden aufstellen zu können, wurden noch zusätzlich die Anforderungen an die Verfahrensentwicklung, Umsetzung und deren Dokumentation beschrieben. Den Abschluss des Kapitels bildet der Punkt Vergleichbarkeit, in dem die Auflösung der verwendeten Texturen und die Rahmenbedingungen zum Rendern der Vergleichsbilder festgelegt wurden um ein weiteres Mal einen unverfälschten Vergleich beider Methoden gewährleisten zu können. Im nächsten Kapitel folgt die Umsetzung der beiden Methoden am Referenzitem.

Kapitel 5

Umsetzung

Das vorherige Kapitel behandelte die, für die Umsetzung nötige, Konzipierung. Angefangen mit der Auswahl der zu untersuchenden Methoden. Dabei handelt es sich um zwei Verfahren, Licht und Schattenbildungen mittels Textur auf ein Objekt zu bekommen. Daraufhin folgte die Auswahl des Items, an welchem die beiden Methoden angewendet werden sollen. Danach wurden die Anforderungen an die Szene in Maya und den verwendeten Methoden definiert, an welche sich während der Umsetzung zu halten ist. Den Abschluss bildete das Festlegen der Rahmenbedingung für den Einsatz beider Methoden. Somit kann die Aufstellung eines unverfälschten Vergleiches sichergestellt werden. In diesem Kapitel folgt nun die Verfahrensentwicklung und schlussendlich die Umsetzung beider Methoden, während im darauffolgenden Kapitel der Vergleich aufgestellt und die Ergebnisse dokumentiert werden.

5.1 Anwendung am Referenzitem

Für das automatische Verfahren müssen sowohl Lichter in der Szene ausgewählt, als auch die Rendereinstellungen angepasst werden, bis ein zufriedenstellendes Ergebnis entsteht. Im folgenden Punkt wird die Verfahrensentwicklung der automatischen Methode für das Item dokumentiert, bevor sie daraufhin angewendet wird. Dies umfasst zum einen die Auswahl, Positionierung und Einstellung der Lichter in der Szene, zum anderen das Anpassen der Rendereinstellungen, bis ein zufriedenstellendes Ergebnis der gebackenen Textur erreicht wird. Daraufhin folgt die Umsetzung der manuellen Methode.

5.1.1 Automatische Methode

Für das automatische Verfahren müssen zunächst Lichter in der Szene ausgewählt werden, bis ein zufriedenstellendes Ergebnis entsteht. Im folgenden Punkt wird die Verfahrensentwicklung der automatischen Methode für das Item dokumentiert, bevor sie daraufhin angewendet wird.

Auswahl der Lichter

Wie in der Konzeptionierung beschrieben, soll der Eindruck eines von oben kommendem Licht erweckt werden, welches sehr weiche Schatten wirft. Im ersten Durchgang müssen also zunächst einmal die richtigen Lichter ausgesucht werden, bis die gewünschte Licht- und Schattenbildungen auf dem Item erscheinen.

Als erstes wird dem Mesh in der Szene die dazugehörige Textur ohne Schatten zugeteilt. Es soll sowohl vorne als auch hinten jeweils ein Lichtpunkt an der Brust und ein breiter Lichtpunkt am Rock erscheinen. Alle vier Lichter sollen weiche Schatten werfen. Zu diesem Zweck werden vier Spot Lights in die Szene gesetzt, wie in Abb. 5.1 zu sehen ist.

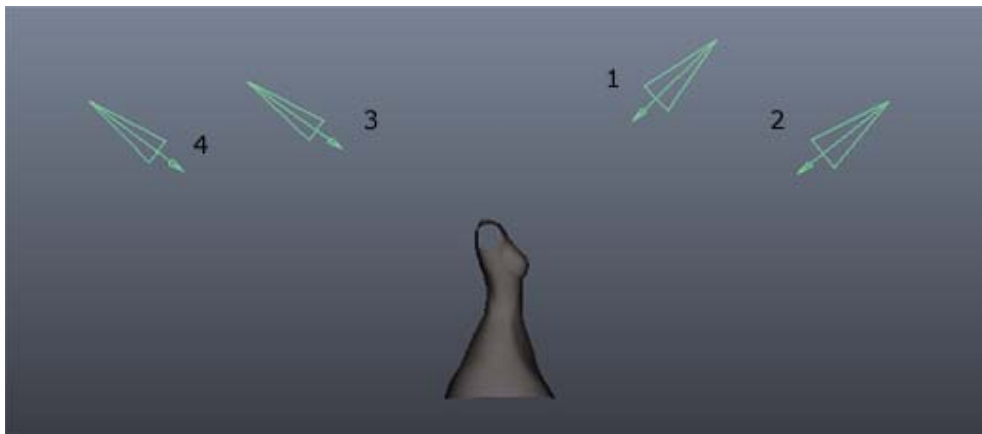


Abbildung 5.1: Referenzitem: Erstes Licht Setup für Baking

Um erste Ergebnisse zu sehen, werden werden die Lichter zunächst mit den Standardeinstellungen gebacken. Für physikalisch möglichst realistische Ergebnisse des Schattenabfalls werden allerdings die „Decay Rate“ auf „Quadratic“ umgestellt und die „Intensity“ dementsprechend auf „10000.00“ erhöht.

Zum Backen der Lichttextur wird dem Mesh als erstes ein neues Bake Set zugeteilt. Dafür wird das Mesh ausgewählt und unter „Rendering“ → „Lighting/Shading“ → „Assign new Bake Set (mental ray)“ → „Texture Bake Set“ ein Bake Set zugewiesen. Dieses wird in „shadowbaking“ umbenannt. Der „Color Mode“ wird auf „Only Lights“ und „Resolution“ auf 1024x1024px umgestellt. Abgesehen davon werden sonst keine Veränderungen in den Einstellungen vorgenommen.

Als nächstes wird, während das Mesh immer noch ausgewählt ist, unter „*Rendering*“ → „*Lighting/Shading*“ → „*Batch Bake (mental ray)*“ das Kontrollkästchen geklickt, woraufhin das Fenster „*Mental Ray Baking Options*“ erscheint. Die Attribute werden wie in Abb. 5.2 eingestellt.

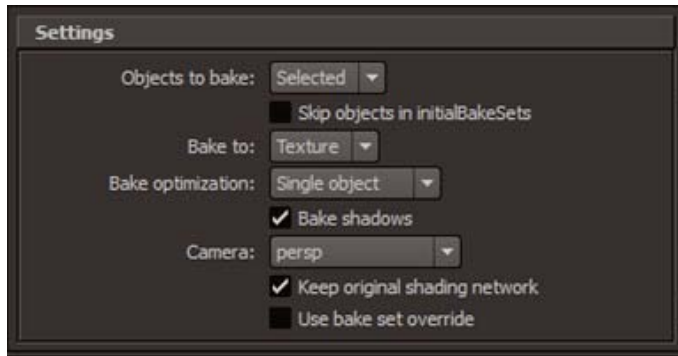


Abbildung 5.2: Referenzitem: Einstellungen für Mental Ray Baking Options

Doch die Standardeinstellungen der Lichtquellen liefern dabei keine schönen Ergebnisse. Die Lichter überlagern sich und es entstehen auch nicht die gewünschten Schatten. Abgesehen davon sind die Schatten noch viel zu hart. Also müssen die Einstellungen der Lichter so verändert werden, bis das Ergebnis von Licht- und Schattenbildung zufriedenstellend ist. (siehe Abb. 5.4)

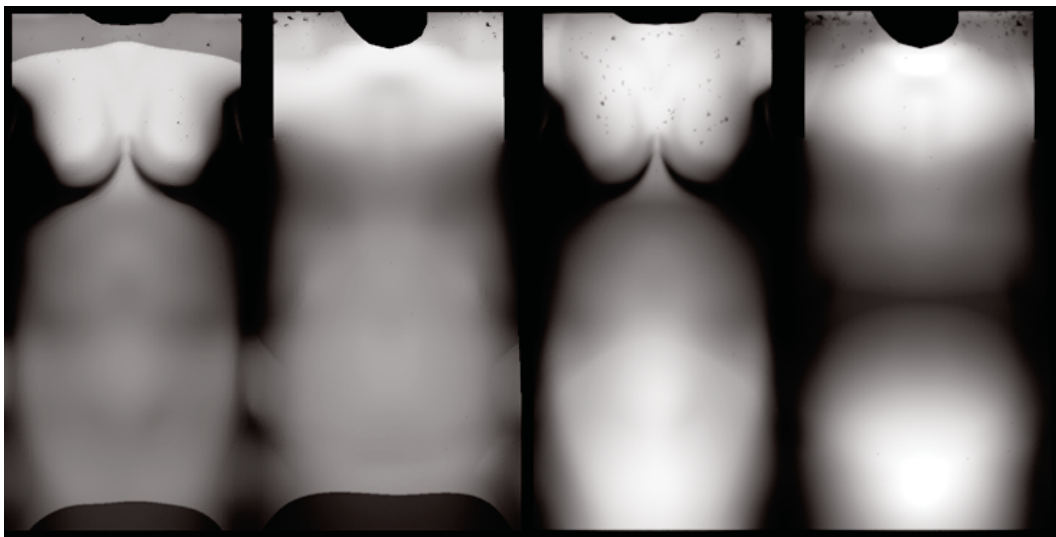


Abbildung 5.3: Links: Standardeinstellungen | Rechts: Angepasste Einstellungen

5. UMSETZUNG

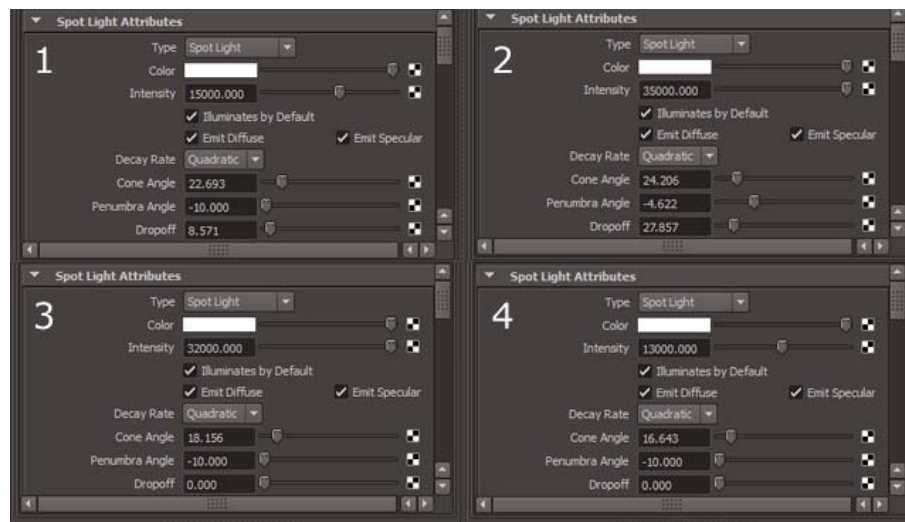


Abbildung 5.4: Vorgenommene Änderungen in den Einstellungen der Spot Lights

Wie in Abb. 5.3 zu sehen ist, entstehen mit den angepassten Einstellungen der Spot Lights wie gewünscht sowohl vorne, als auch hinten jeweils zwei Lichtpunkte, welche weiche Schatten werfen. Um die Texturen noch etwas zu erhellen, werden noch zwei Area Lights gesetzt. Somit befinden sich insgesamt sechs Lichtquellen in der Szene (siehe Abb. 5.5).

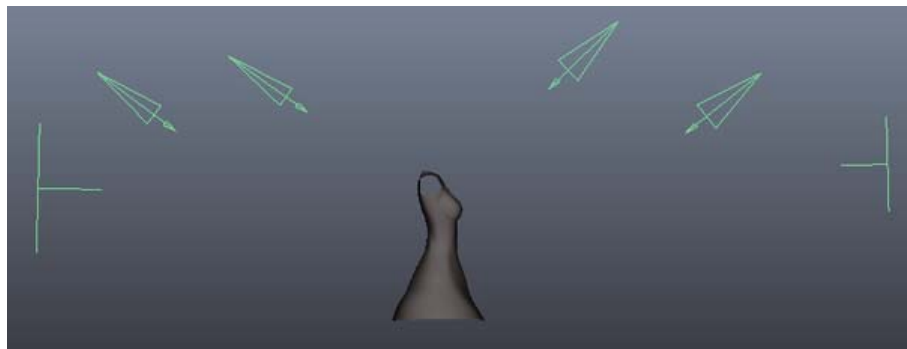


Abbildung 5.5: Endgültiges Licht Setup für Baking

Anpassung der Qualität

Nachdem das gewünschte Ergebnis der Licht- und Schattenbildung, wie in Abb. 5.3 zu sehen ist, erreicht wurde, wird nun die Qualität der gebackenen Textur verbessert. Dafür müssen sowohl die Render- als auch die Baking Einstellungen angepasst werden. Um das zu erreichen, wird „*Sampling Mode*“ → „*Legacy Sampling Mode*“ und „*Custom Sampling*“ eingestellt und die weiteren Einstellungen so lange verändert, bis sich die Qualität und die Renderzeit in einem guten Rahmen befinden.

Um die Punkte, die in Abb. 5.6 zu sehen sind, zu entfernen und die Qualität der Schatten weiter zu erhöhen, wird in den Rendersettings unter „*Indirect Lighting*“ → „*Final Gathering*“ aktiviert und die „*Accuracy*“ auf 256, sowie die „*Point Density*“ auf 0.1 und „*Point Interpolation*“ auf 10 gesetzt.

Da dies die Punkte allerdings noch nicht vollständig verschwinden lässt, wird bei den „*Texture Bake Set Attributes*“ → „*Fill Texture Seams*“ auf 3.0000 eingestellt. Somit entsteht die abschließende Textur mit dem Namen „**bakingfinal.tif**“ (siehe Abb. 5.7).

Diese kann nun mit der Textur „**itemdefault.png**“, welche bis jetzt noch keine Schatten besitzt, in Photoshop zusammengeführt werden.

Bevor dieser Vorgang im nächsten Absatz erläutert wird, folgt an dieser Stelle noch eine kurze Aufzählung, was bisher für die automatische Methode entwickelt wurde:

- Jeweils vorne und hinten an der Oberseite des Items ein konzentrierter Lichtpunkt
- Jeweils vorne und hinten an der Unterseite des Items ein weiter Lichtpunkt
- Weiche Schattenbildung an den Seiten des Items
- Schattenbildung unter der Brust
- Gute Qualität der gebackenen Lichter und Schatten bei kurzer Renderzeit

Auf den Zeitaufwand für die einzelnen Schritte und in der Summe wird in Kapitel 6. näher eingegangen.

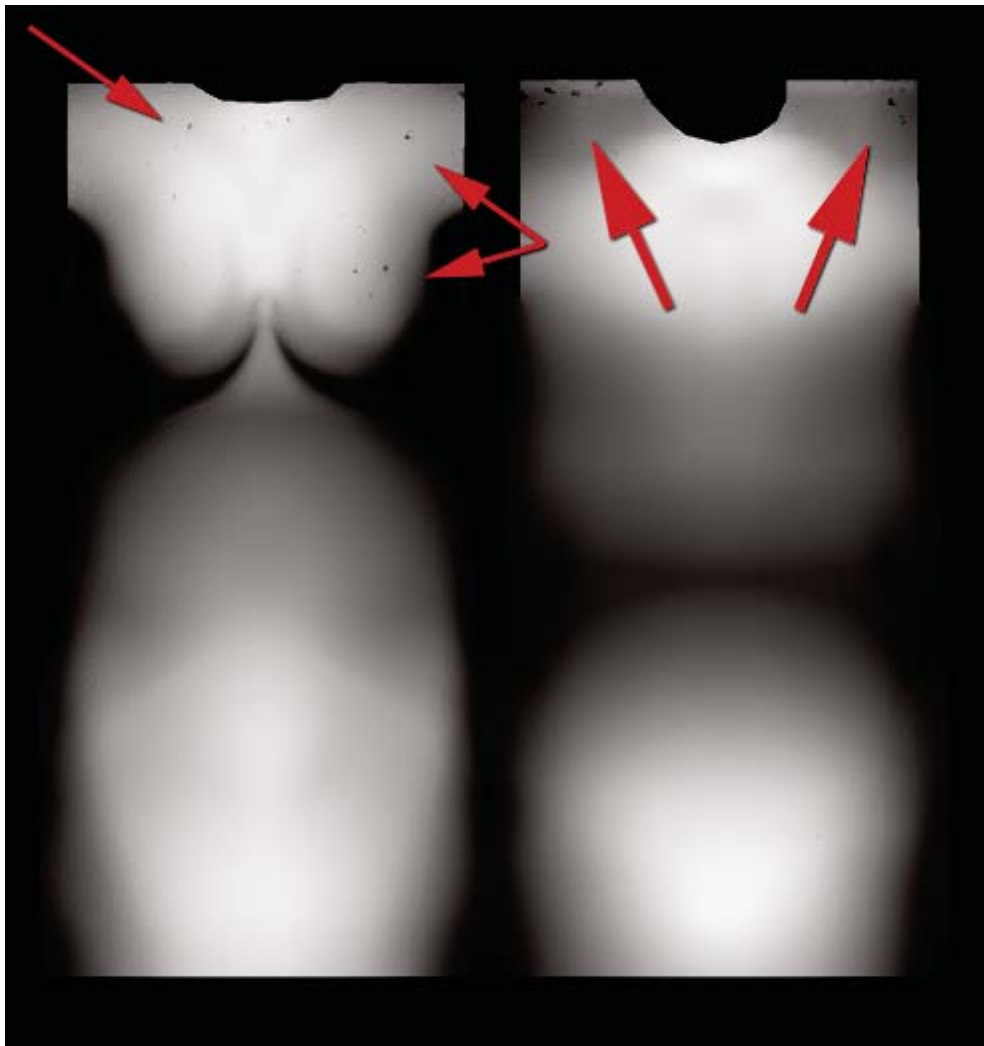
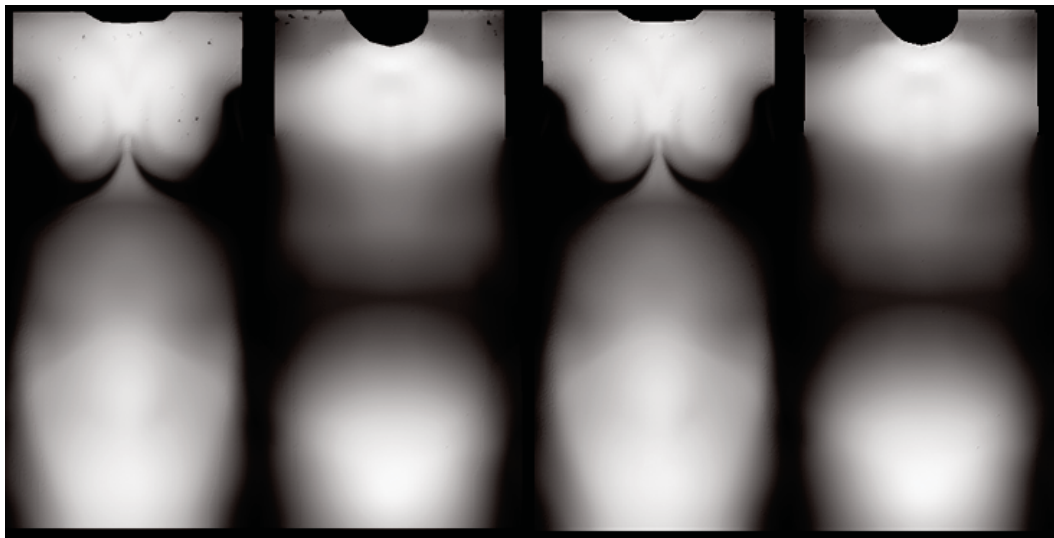


Abbildung 5.6: Die Qualität der gebackenen Licht- und Schatteninformationen ist noch nicht ausreichend



Min Sample Mode	-2
Max Sample Mode	1
Anti Aliasing	0.050
Filter	Gauss
Shadows	Simple
Final Gathering	aus
Renderzeit	0:08 min

Min Sample Mode	-1
Max Sample Mode	4
Anti Aliasing	0.10
Filter	Mitchell
Shadows	Segments
Final Gathering	an
Accuracy	256
Point Density	0.100
Point Interpolation	10

Renderzeit	0:25 min
------------	----------

Abbildung 5.7: Vergleich der Renderings mit unterschiedlichen Rendereinstellungen

Anwendung

Nachdem die Licht- und Rendereinstellungen so angepasst wurden, dass die Textur „**baking-final.tif**“ sowohl die gewünschten Licht- und Schattenbildungen, als auch eine gute Qualität vorweist, kann sie nun an der Textur „**itemdefault.png**“ angewendet werden.

Dafür wird die Textur „**itemdefault.png**“ in Photoshop geöffnet und die Textur „**baking-final.tif**“ in einer neuen Ebene auf die Erste gelegt.

Damit die Schatten am besten zur Geltung kommen, wird die Füllmethode der „*gebackenen*“ Textur auf „*Multiplizieren*“ geändert. Allerdings sind die Schatten der Textur noch etwas zu dunkel, also wird die Deckkraft der Ebene auf 60% verringert.

Der Vergleich ist in der Abb. 5.8 zu sehen. Damit ist die Textur mit Licht- und Schatteninformation der automatischen Methode mittels „*baking*“ fertig. Im nächsten Punkt wird die manuelle Methode angewendet.



Abbildung 5.8: Links: Deckkraft der bakingfinal Ebene bei 100% | Rechts: Deckkraft der bakingfinal Ebene bei 60%

5.1.2 Manuelle Methode

Wie in der Konzeptionierung beschrieben, handelt es sich bei der manuellen Methode um das Einzeichnen der Licht- und Schatteninformationen per Hand in Photoshop. Mit Hilfe des Pinsel Werkzeugs, dessen Größe, Deckkraft und Härte beliebig verändert werden können, lassen sich die gewünschten Schatten auf die Textur zeichnen. Mit dem UV Layout als Ebene, kann die Aufteilung der Textur nachvollzogen werden. Für jeden neue Schattengruppe wird eine neue Ebene erstellt, sodass diese jeweils unabhängig voneinander manipulieren werden können. So entstehen, wie in Abb. 5.9 zu sehen ist, Schritt für Schritt die gewünschten Schatten. Es folgt eine kurze Auflistung, was in den einzelnen Schritten unternommen wurde:

1. Die ersten Schatten an den Seiten des Items wurden mit dem Pinsel eingezeichnet
2. Die Schatten aus dem ersten Schritt wurden mit Hilfe des Wischwerkzeugs und Radiergummis weicher gezeichnet
3. Schatten unter der Brust wurden eingezeichnet
4. Schatten am Brustbereich wurden wieder mit Hilfe der beiden Werkzeuge aus dem zweiten Schritt angepasst
5. Schatten am Rücken und an den Schultern wurden eingezeichnet

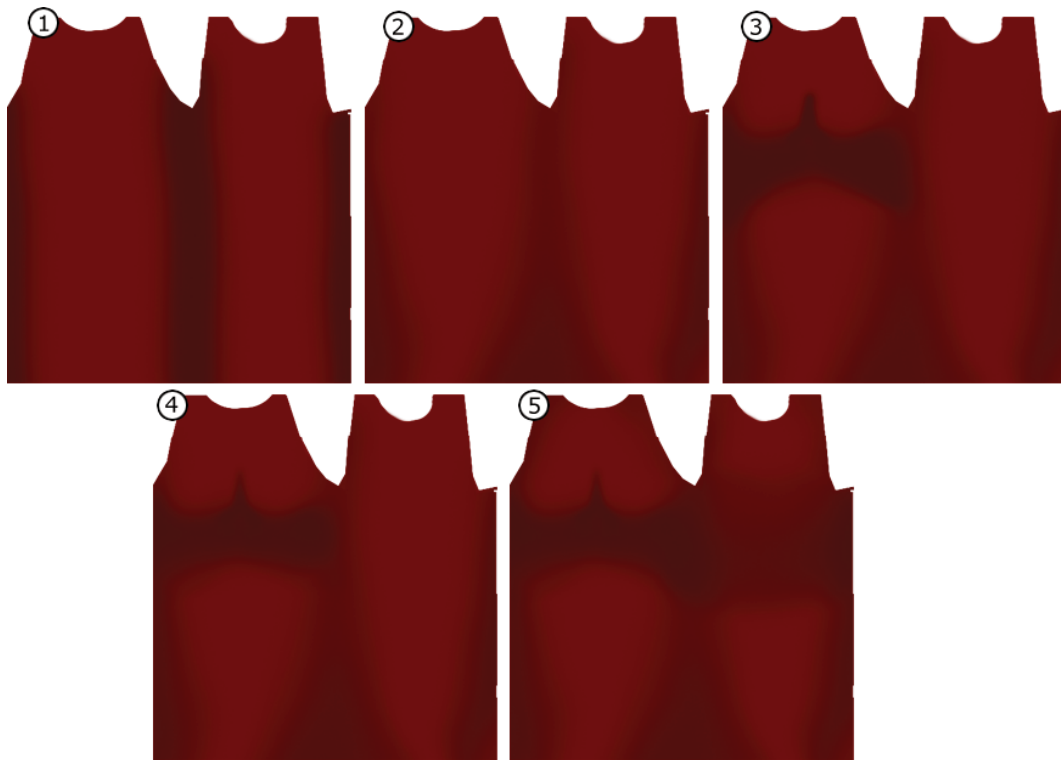


Abbildung 5.9: Die einzelnen Schritte der Anwendung

5.2 Zusammenfassung

In diesem Kapitel erfolgte die Umsetzung der zuvor beschriebenen Methoden für die Simulation von Licht am Objekt. Als erstes wurde die automatische Methode umgesetzt, wofür zunächst die richtige Auswahl, Positionierung und Einstellung der, in der Szene befindlichen, Lichter entwickelt wurde. Angefangen mit vier Spot Lights und deren Standardeinstellungen, erwiesen sich dessen Ergebnisse als nicht zufriedenstellend. Daraufhin wurden die Einstellungen der Lichter so lange angepasst, bis die gewünschte Licht- und Schattensetzung am Item erreicht wurde. Als nächstes wurden die Rendereinstellungen so weit verändert, bis die Qualität der gebackenen Lichttextur gut genug war. Zum Schluss wurde diese Textur mit der ursprünglichen Textur des Items in Photoshop zusammengebracht und die Deckkraft der Ebenen angepasst, womit die automatische Methode vollständig war. Als nächstes wurde die manuelle Methode umgesetzt und jeder neu gesetzte Schatten und dessen Anpassung dokumentiert. Der abschließende Vergleich des Zeit- und Arbeitsaufwandes, sowie den optischen Unterschieden folgt im nächsten Kapitel.

Kapitel 6

Ergebnis

Im vorangegangenen Kapitel wurden die einzelnen Schritte der Verfahrensentwicklung und Umsetzung der manuellen und automatischen Methode anhand einer Referenzszene und einem Referenzitem beschrieben. Im folgenden Kapitel werden nun die Ergebnisse dieser Umsetzungen aufgezeigt. Dabei wird jeder Durchgang einzeln behandelt und sowohl die Anzahl der Arbeitsschritte, die Dauer der Umsetzung und die optischen Unterschiede der Texturen und Renderings gegenübergestellt und miteinander verglichen. Nach dem Vergleich dieser Merkmale folgt im nächsten Unterpunkt noch die Evaluierung dieser Methoden anhand der hier vorgestellten Ergebnisse und ein Fazit.

6.1 Vergleich

Gegenüberstellung des Arbeitsaufwandes

Der Arbeitsaufwand erwies sich aufgrund der Lichtsetzung bei der automatischen Methode größer als bei der manuellen. Durch die unzufriedenen Ergebnisse der Licht- und Schattenformung musste erst das richtige Licht Setup gefunden werden, bevor dieses auf die Textur gebacken werden konnte. So mussten die Lichter nach jeder neuen Positionierung, Ausrichtung und Einstellungen auf ein Neues gerendert werden um herausfinden zu können, ob die gebackene Textur den Vorstellungen entspricht. Im Vergleich dazu konnten bei der manuellen Methode die Schatten direkt an den gewünschten Position eingezeichnet werden, jedoch musste jeder neue gesetzte Schatten mit dem Wischfinger und Radiergummi Werkzeug angepasst werden. In der Übersicht der Arbeitsschritte 6.1 wird sofort deutlich, dass in diesem Fall die automatische Methode einen größeren Aufwand benötigt. Hinzu kommt, dass die Schritte 1-7 mehrere Male wiederholt werden mussten, bis die Lichtsetzung und Qualität der gebackenen Textur ein zufriedenstellendes Ergebnis lieferte. Währenddessen musste bei der manuellen Methode kein einziger Arbeitsschritt wiederholt werden.

Manuelle Methode	Automatische Methode
1. Erste Schatteneinzeichnung an den Seiten	1. Passende Lichtquellen finden
2. Erste Schatten formen	2. Lichter positionieren
3. Schatten am Brustbereich setzen	3. Lichter einstellen
4. Schatten am Brustbereich formen	4. Lichter positionieren
5. Schatten am Rücken und an den Armen setzen	5. Testrendering der Lichter
6. Schatten am Rücken und Armen formen	6. Rendereinstellungen zum Backen setzen
	7. Testen der Rendereinstellungen
	8. Gebackene Textur rendern
	9. Texturen in Photoshop zusammenfügen

Tabelle 6.1: Gegenüberstellung der Arbeitsschritte

Gegenüberstellung des Zeitaufwandes

In den Tabellen 6.2 und 6.3 wird der Zeitaufwand von beiden Methoden aufgelistet:

Arbeitsschritt	Zeitaufwand (Minuten)
1. Erste Schatteneinzeichnung an den Seiten	3:15
2. Erste Schatten formen	4:10
3. Schatten am Brustbereich setzen	2:02
4. Schatten am Brustbereich formen	1:38
5. Schatten am Rücken & den Armen setzen	00:33
6. Schatten am Rücken & Armen formen	1:32
7. Ebenen umstellen	00:28

Insgesamt: ~ 13 Minuten

Tabelle 6.2: Zeitaufwand der manuellen Methode

Wie den drei Tabellen zu entnehmen ist, ist genau wie beim Arbeits- auch der Zeitaufwand bei der automatischen Methode größer, was allerdings hauptsächlich an der Suche nach den passenden Lichtern und richtigen Rendereinstellungen zum Backen liegt. Wären diese Faktoren von vornherein klar gewesen, lässt sich das eigentliche „*baking*“ Verfahren und Zusammenführung der Texturen schneller anwenden, als die manuelle Methode.

Arbeitsschritt	Zeitaufwand (Minuten)
1. Auswahl der passenden Lichter	12:45
2. Lichter positionieren	10:11
3. Lichter richtig einstellen	26:33
Licht Setup einstellen: ~ 49 Minuten	
4. Rendereinstellungen setzen	18:45
4. Baking Verfahren	00:20
6. Endgültige Renderzeit der gebackenen Textur	00:25
7. Textur in Photoshop zusammenfügen & Ebene umstellen	4:02

Insgesamt: 72:45 Minuten bzw. ~ 1 Stunde 12 Minuten

Tabelle 6.3: Zeitaufwand der automatischen Methode

Gegenüberstellung der Texturen und Renderings

Für eine optische Gegenüberstellung dienen, wie schon in der Konzeptionierung definiert, sowohl die fertigen Texturen beider Methoden, als auch jeweils drei gerenderte Bilder. Eine kurze Wiederholung der Renderumgebung: Das Item wurde hierbei unter identischen Umständen in genau der gleichen Umgebung und Position gerendert. Der einzige Unterschied bei den gerenderten Bildern besteht lediglich in der Textur.



Abbildung 6.1: Links: Manuelle Methode | Rechts: Automatische Methode

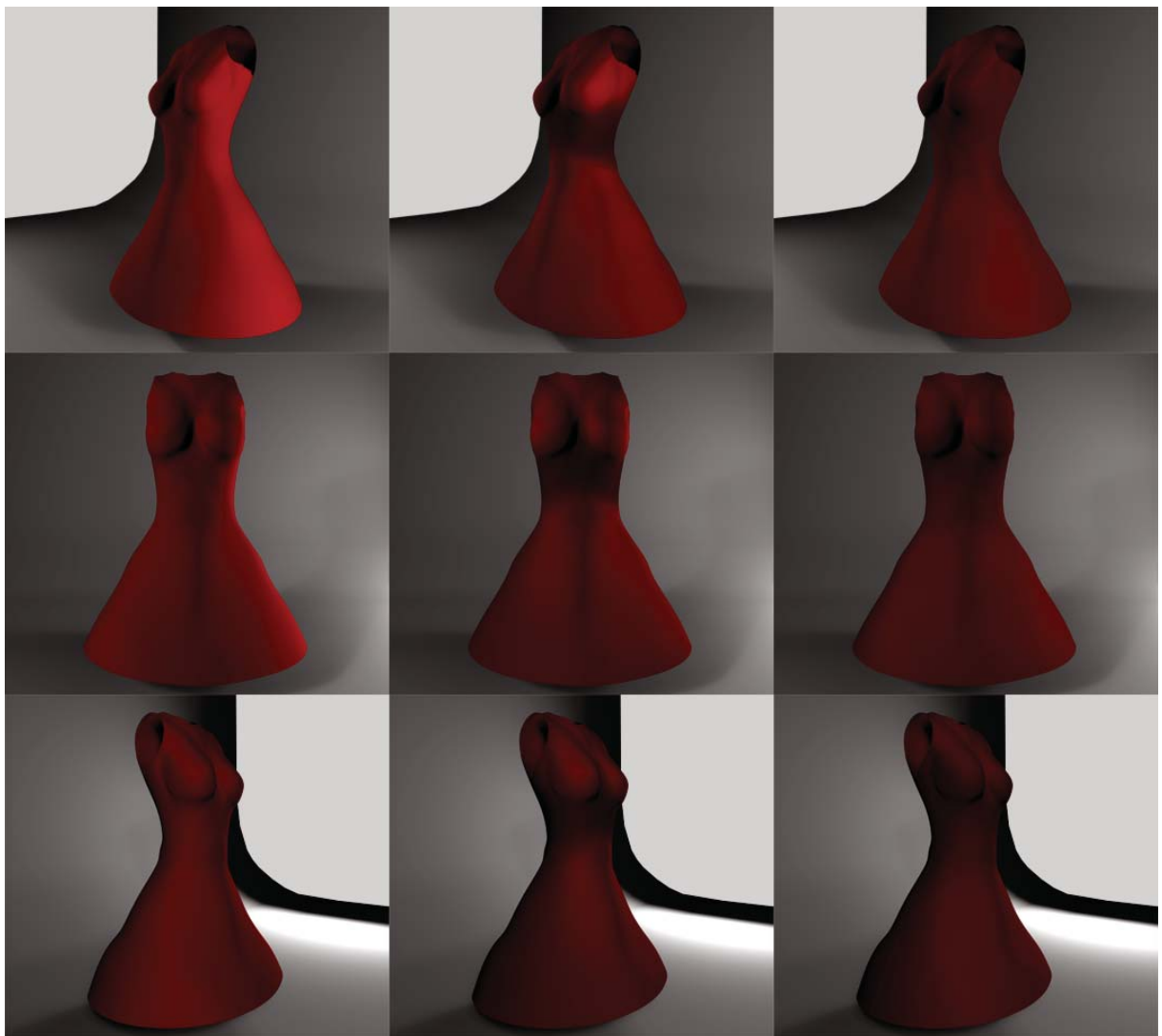
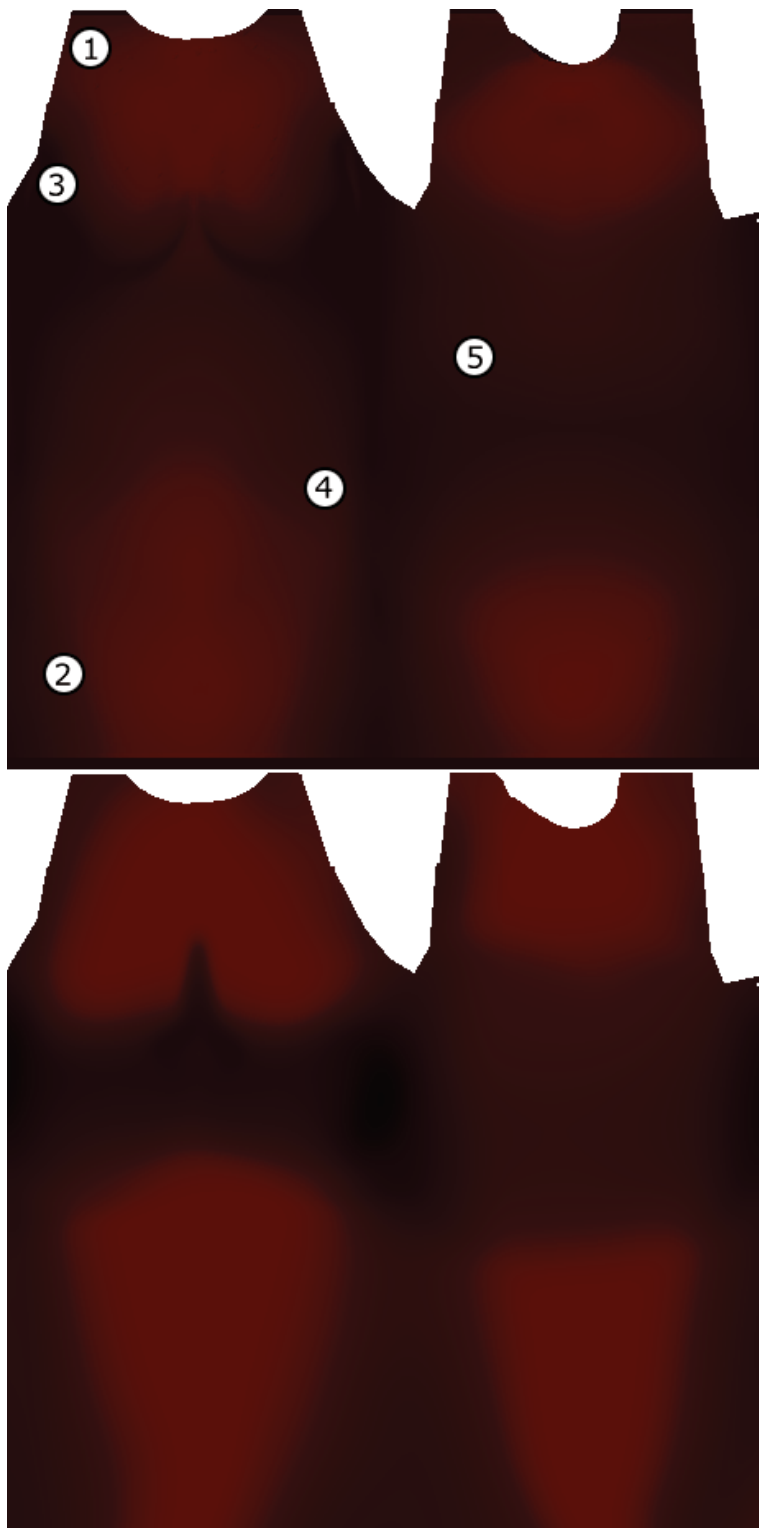


Abbildung 6.2: Links: Textur ohne Schatten | Mitte: Manuellen Methode | Rechts: Automatische Methode

Auf den ersten Blick ist sofort zu erkennen, dass sich die Schattenbildungen am Brustbereich deutlich voneinander unterscheiden. Die gebildeten Schatten der automatischen Methode sind im Vergleich zur manuellen viel präziser und wirken viel realistischer. Trotz der Hilfestellung des UV Layouts ist es nicht gelungen, die Schattenbildungen am Brustbereich präzise darzustellen. Auch sind sie bei der automatischen Methode, neben dem Brustbereich, auch am Rücken und an den Armen viel weicher, während sie an den Schnittkanten der Textur im Vergleich doch konzentrierter und dunkler sind.



1. **Schultern**
2. **Seitenschatten**
3. **Brustbereich**
4. **Überlappung**
5. **Rücken**

Abbildung 6.3: Optische Unterschiede der Texturen
Oben: Automatische Methode | Unten: Manuelle Methode

1. **Schultern**

Die Schatten an den Schultern sind bei der automatischen Methode stärker und dunkler als bei der manuellen. Sie scheinen auch konzentrierter zu sein und nicht so weich „*auszulaufen*“.

2. **Seitenschatten**

Die Schatten, die sich an den Seiten des Items bilden, sind wiederum bei der automatischen Methode konzentrierter und auch kleiner. Gerade im unteren Drittel der Textur verlaufen die Schatten bei der manuellen Methode viel stärker in die Mitte als bei der automatischen. Des Weiteren ist der Schattenabfall bei der automatischen Methode viel besser zu erkennen, während er bei der manuellen Methode, trotz des Einsatzes des Wischfinger und Radiergummi Werkzeugs, kaum existiert. Dadurch wirkt der Schattenwurf bei der manuellen Methode viel flacher und unrealistischer als bei der automatischen.

3. **Brustbereich**

Der wohl größte Unterschied der Schatten ist am Brustbereich zu erkennen. Im Vergleich zu der manuellen Methode, „*ummanteln*“ die Schatten bei der oberen Textur die Brust und es ist ein deutlicher Unterschied zwischen den Schatten um die Brustform herum und unterhalb des Brustbereiches zu erkennen.

4. **Überlappung der Schatten**

An beiden Texturen sind die Überlappungen der Schatten an den Seiten zusammen mit denen am Brustbereich und Rücken zu erkennen. Dabei sind sie bei der manuellen Methode stärker ausgeprägt als bei der automatischen.

5. **Rücken**

Auch hier läuft die Schattenbildung bei der automatischen Methode viel weicher aus, als bei der manuellen. Es existiert ein Verlauf vom dunkelsten Punkt (direkt unterhalb der Zahl 5) bis hin zu den Lichtpunkten nach oben und untenhin. Währenddessen scheint bei der manuellen Methode der untere Bereich der Textur viel heller ausgeleuchtet zu sein.

6.2 Evaluierung der Methoden

Nachdem im vorherigen Punkt die Ergebnisse beider Methoden vorgestellt wurden, folgt in diesem nun ihre Evaluierung. Dabei erfolgt die Bewertung anhand des Zeit- und Arbeitsaufwandes, der subjektiven Einschätzung der Ästhetik und der jeweiligen Möglichkeiten jeder Methode.

6.2.1 Zeit- und Arbeitsaufwand

Der größte Unterschied im Zeitaufwand der beiden Methoden besteht in der optimalen Lichtsetzung beim automatischen Verfahren. Die meiste Zeit wurde für die richtige Auswahl, Positionierung und Einstellung der Lichter in der Szene gebraucht. Zwar konnte, dank der Konzeptionierung, die Auswahl der passenden Lichtquellen schnell getroffen werden, doch wie im Kapitel Umsetzung zu sehen ist, lieferten die Standardeinstellungen der Lichter keine zufriedenstellenden Ergebnisse.

Gerade beim Referenzitem waren die Lichter viel zu konzentriert und auch die Schattenbildungen waren nicht wie gewünscht und überlagerten sich noch zusätzlich. Im Vergleich zu der manuellen Methode war der Prozess zur Findung der richtigen Lichteinstellungen also sehr langwierig. Dieses Problem bestand bei der manuellen Methode durch den Einsatz des UV Layouts nicht. Indem in Maya ein Snapshot des UV Layouts gemacht und abgespeichert werden konnte, wurde dieses als eigene Ebene in Photoshop auf die Textur gelegt. Damit waren die Schnittkanten der Textur und Form des Meshs auf einem Blick zu erkennen und die Schatten konnten direkt nach Wunsch eingezeichnet werden. Somit konnte die Position der Schatten schnell bestimmt werden. Ihre endgültige Form zu erreichen, war bei der manuellen Methode allerdings am zeitaufwändigsten. Bis das letztendliche Ergebnis erreicht wurde, mussten die Schatten mehrere Male eingezeichnet, verwischt, ausradiert und wieder eingezeichnet werden.

Die richtigen Rendereinstellungen für eine gute Qualität der gebackenen Lichter und Schatten zu finden, erwies sich als vergleichsweise sehr einfach und ging schneller als zuvor angenommen. Somit dauerte es nicht lange, bis die passenden Einstellungen zum Rendern gefunden worden sind, ebenso das eigentliche Rendern der gebackenen Textur. Dabei ist der alleinige Prozess des „bakings“ mit nur wenigen Klicks erledigt. Sobald die richtige Auswahl der Lichtquellen und deren Einstellungen getroffen worden sind, lässt sich die Methode demnach sehr einfach und schnell anwenden.

6.2.2 Ästhetik

Wie in Abb. 6.1 zu sehen ist, unterscheiden sich beide Methoden optisch stark voneinander. Die optischen Unterschiede sind dabei sogar größer als anfangs vermutet. Auffällig ist, dass die Schatten der automatischen Methode viel weicher auslaufen als bei der manuellen, auch wenn bei letzterer wiederholt das Radiergummi Werkzeug eingesetzt wurde, um genau solch ein Auslaufen zu erstellen. (siehe Abb. 6.4) Es wäre natürlich möglich, die Textur der manuellen Methode erneut in Photoshop zu öffnen und weiterhin anzupassen, so dass der Verlauf der Schatten stärker zu erkennen ist, doch um einen ordentlichen Vergleich beider Methoden aufzustellen, wird keine der beiden nachträglich noch angepasst.

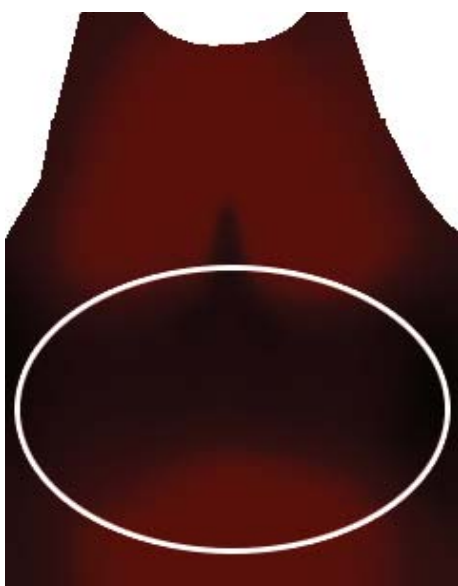


Abbildung 6.4: Schattenverlauf der manuellen Methode

Während der Umsetzung schienen die Schattenbildungen bei der manuellen Methode ebenfalls einen schönen und realistischen Verlauf zu besitzen. Erst im direkten Vergleich der Texturen und Renderings sind die deutlichen Unterschiede zu erkennen. So ist die entstandene Licht- und Schattensetzung bei der automatischen Methode subtiler und optisch auch ansprechender, während sie im Vergleich dazu bei der manuellen an einigen Stellen viel zu stark ist. Genau unterhalb der Brust ist der geworfene Schatten der manuellen Methode zu stark und ein richtiges Auslaufen ist nicht zu erkennen. Im direkten Vergleich in Abb.6.5 fällt sofort auf, dass das Auslaufen des Schattens bei der automatischen Methode besser zu erkennen ist und auch die Schatten um die Brustform herum unterstreichen die Form des Items besser als die eingezeichneten.

Der geworfene Schatten unterhalb des Brustbereiches wirkt bei der manuellen Methode besonders unrealistisch, da er im Vergleich viel zu konzentriert ist. Er ist zu dunkel und verläuft zu schnell ins helle, wodurch der Eindruck von sehr hartem Licht entsteht, was ja nicht beabsichtigt war. Wenn man beide Methoden im direkten Vergleich zu einander sieht, fällt demnach auf, dass bei der manuellen Methode eine bessere Bearbeitung der Schatten nötig gewesen wäre um einen weichen Verlauf zu schaffen.



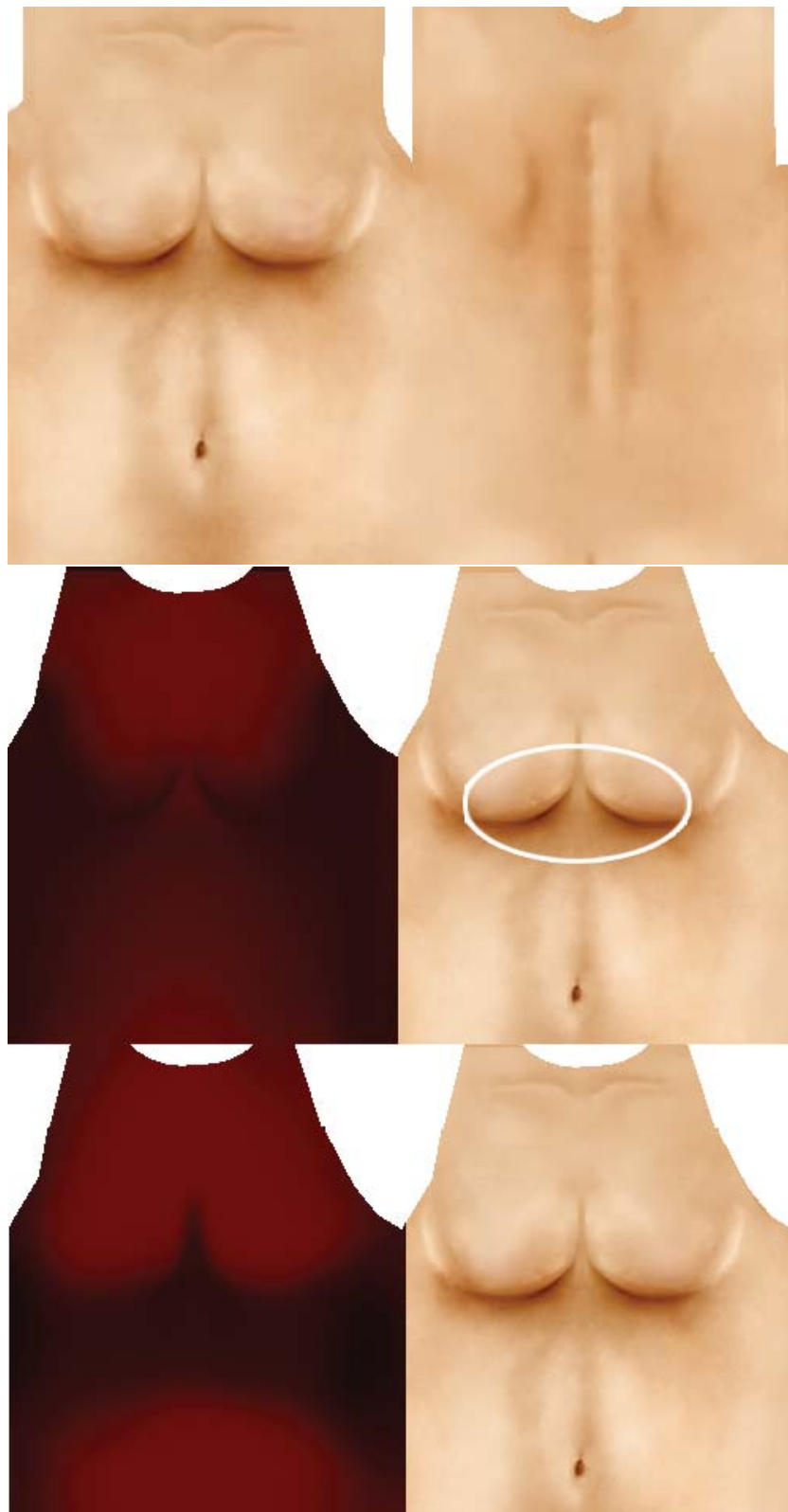
Abbildung 6.5: Links: Manuelle Methode | Rechts: Automatische Methode

Des Weiteren ist deutlich zu sehen, dass bei der automatischen Methode mehrere Schatten unterhalb der Brust entstanden sind, wobei durch deren Überdeckung und Auslaufen ein Zusammenspiel entsteht, welches die Brustform umschmeichelt. Zur besseren Einschätzung, wurde eine Hauttextur, welche bei der Firma SMeet GmbH für die Avatare genutzt wird, als Referenz hinzugezogen. Für einen besseren optischen Vergleich wurden die Texturen beider Methoden jeweils über die Hauttextur gelegt und deren Ebene auf „*Hartes Licht*“ eingestellt, so ist in Abb.6.6 sowohl die Brustform der Hauttextur, als auch die Schattenbildungen der jeweiligen Methoden zu sehen. Dadurch sind nun ein weiteres Mal die Unterschiede der Schatten im Brustbereich beider Methoden sichtbar. Während sich die Schatten auf der linken Seite der Abbildung stark voneinander unterscheiden, wirken diese auf der rechten Seite beinahe identisch.

In Abb.6.7 ist der Vergleich dieser Textur mit beiden Methoden zu sehen. Wie in der Abbildung markiert, ähnelt die Schattenbildung mittels automatischer Methode der Hauttextur viel eher als die der manuellen Methode.



Abbildung 6.6: Links: Manuelle Methode | Rechts: Automatische Methode



6.2.3 Möglichkeiten

Während die Schatten beim automatischen Verfahren sicherlich präziser und mit Hilfe der Berechnungen von Maya physikalisch korrekter ausfallen, müssen alle Details und Formen, welche im Ergebnis zu sehen sein sollen, tatsächlich modelliert werden. Dies kann sich sowohl sehr negativ auf den Zeit- und Arbeitsaufwand auswirken, wie auch die Anzahl der Polygone stark erhöhen. Die Möglichkeit, den Objekten Form, Tiefe und Details zu verschaffen, einzig durch die Freiheit, Schatten beliebig auf die Textur einzuzeichnen, ist somit wohl der größte Vorteil der manuellen Methode gegenüber der automatischen. In Abb. 6.8 ist der direkte Vergleich der Endergebnisse aus der automatischen und der manuellen Methode mit eingezeichneten Details zu sehen. Obwohl der Rock vom Mesh eben ist, wird durch eingezeichnete Schatten der Eindruck eines Faltenrocks erweckt.



Abbildung 6.8: Links: Manuelle Methode mit zusätzlich eingezeichneten Schatten am Rock | Rechts: Automatische Methode

Bei der automatischen Methode hingegen, lässt sich tatsächliche physikalische Korrektheit des Licht- und Schatteneinfalls erreichen. Damit lassen sich Szenen und Objekte realistisch ausleuchten, während dies bei der manuellen Methode so gut wie unmöglich ist. Sicherlich ist die künstlerische Fähigkeit des Designers dabei entscheidend, dennoch dürfte das physikalisch korrekte Einzeichnen der Licht- und Schattenbildungen sehr schwer zu erreichen sein.

6.3 Fazit

Wie bereits in den vorherigen Punkten gezeigt wurde, unterscheiden sich beide Methoden wesentlich voneinander. So ist der Zeit- und Arbeitsaufwand bei der automatischen Methode größer, als bei der manuellen. Dies liegt allerdings hauptsächlich an den eingesetzten Lichtern in der Szene. Bis dessen Positionierung und endgültigen Einstellungen gefunden wurden, verstrich mehr Zeit als ursprünglich erwartet. Wie in der Tabelle 6.3. Zeitaufwand der automatischen Methode zu sehen, wurde mehr als die Hälfte der benötigten Zeit dazu verwendet, das richtige Licht Setup zu finden. Danach dauerte es weitere 18 Minuten die passenden Rendereinstellungen zu setzen, damit die Lichter und Schatten auch in angemessener Qualität gebacken werden. Wären diese Umstände von vornherein bekannt gewesen, hätte die automatische Methode viel weniger Arbeitsschritte und einen kürzeren Zeitaufwand benötigt, als hier vorgestellt.

Dies ist eine der wichtigsten Erkenntnisse, welche durch die Umsetzung gewonnen werden konnte. Wenn man bedenkt, dass es möglich ist, ein Licht Setup und die Rendereinstellungen abzuspeichern, um es für jedes weitere Item benutzen zu können, kann dadurch eine Menge Zeit gewonnen werden.

Unter Berücksichtigung des Zeit- und Arbeitsaufwandes ist unter erstmaligen Anwendung mit Sicherheit die manuelle Methode zu empfehlen. Wenn die eingesetzten Lichter und Rendereinstellungen jedoch schon von Anfang an bekannt sind, wodurch sich die Anzahl der Arbeitsschritte und dessen Zeitaufwand für diese Schritte erübrigt, lässt sich, aufgrund der besseren optischen Ergebnisse, eine deutliche Empfehlung für die automatische Methode aussprechen.

Doch die manuelle Methode bietet im Vergleich zu der automatischen einen anderen großen Vorteil: Mit ihr lassen sich Details einzeichnen, die im Mesh so überhaupt nicht vorhanden sind. Durch das Einzeichnen von Schatten, kann mit Hilfe der optischen Täuschung aus einem geraden Rock, ein Faltenrock entstehen. Dies macht die manuelle Methode zu einem stärkeren Werkzeug, als die automatische. Bei letzteren bilden sich nämlich lediglich nur die Schatten, welche durch die tatsächliche Form des Meshs entstehen. Wäre hierbei ein Faltenrock erwünscht, so müsste das Mesh des Rocks auch dementsprechend verändert werden. Der Faltenrock dient hierbei nur als Beispiel, die Möglichkeit zur optischen Täuschung lässt sich bei der manuellen Methode auf vielerlei Arten einsetzen. Wäre der Eindruck von abstehenden Knöpfen erwünscht, ohne weitere Meshs erstellen zu wollen, so könnten die Knöpfe und dessen Schattenwurf in der manuellen Methode eingezeichnet werden. Während ersteres auch vor der Anwendung der automatischen Methode möglich wäre, würden diese allerdings keinen Schatten werfen, da sie ja nur auf der Textur vorhanden wären.

Eine klare Empfehlung für eine der beiden Methoden, unter Berücksichtigung aller Punkte, lässt sich also nicht aussprechen. Die Auswahl des Verfahrens ist abhängig von der verfügbaren Zeit und den gewünschten Ergebnissen. Wer wenig Zeit zur Verfügung hat und wem die Auswahl und Positionierung der Lichter in 3D Szenen schwer fällt, sollte auf die manuelle Methode zurückgreifen. Hinzu kommt, wie bereits erwähnt, dass diese Methode noch weitere Manipulation von Licht- und Schattenbildungen ermöglicht. Wer allerdings etwas mehr Zeit zur Verfügung hat und auf physikalische Korrektheit oder die Qualität der Ästhetik mehr Wert legt, dem ist die automatische Methode zu empfehlen.

6.4 Zusammenfassung

In diesem Kapitel wurden als Ergebnis der Methodik Vergleiche zwischen beiden Methoden aufgestellt. Angefangen mit der Gegenüberstellung des Arbeitsaufwandes, wobei unter diesem Punkt die einzelnen Arbeitsschritte dokumentiert und deren Anzahl miteinander verglichen wurde. Danach folgte die Gegenüberstellung des Zeitaufwandes beider Methoden und schlussendlich wurden die Texturen und Renderings miteinander verglichen. Zusammenfassend war sowohl der Arbeits- als auch der Zeitaufwand bei der automatischen Methode größer und die optischen Unterschiede zwischen den beiden Methoden auffälliger als erwartet. Im nächsten Punkt folgte die Evaluierung der Ergebnisse, wobei zwischen Zeit- und Arbeitsaufwand, Ästhetik und der jeweiligen Möglichkeiten beider Methoden aufgeteilt wurde. Zum Schluss wurde ein Fazit gezogen, welches auf den Erkenntnissen beruht, die aus den Ergebnissen gewonnen werden konnten. Das nächste Kapitel ist gleichzeitig das letzte dieser Arbeit und beinhaltet die Zusammenfassung der Ergebnisse und einen Ausblick.

Kapitel 7

Zusammenfassung und Ausblick

Diese Arbeit befasst sich mit der Möglichkeit, Licht und Schatten auf die Textur eines Items zu projizieren, um eine Beleuchtung im späteren Spiel zu simulieren. Ziel war es dabei, zwei Methoden für solch eine Licht- und Schattensetzung anzuwenden und miteinander zu vergleichen. Im ersten Kapitel erfolgt eine Einführung in das Thema. Angefangen mit der Motivation, genau über dieses Thema eine Arbeit zu verfassen, gefolgt von der Beschreibung der Problemstellung, bis hin zur Formulierung der Zielsetzung. Das Ende des ersten Kapitels bildet eine Zusammenfassung dieser Arbeit, um einen Überblick zu verschaffen, bevor es im nächsten Kapitel um den Stand der Technik geht. In diesem wurde die Entwicklung der 3D Grafik in Spielen beschrieben, um aufzuzeigen, wie sehr sich die heutigen Darstellungsmöglichkeiten in der Computergrafik, im Vergleich zu seinen Anfangszeiten, verändert hat. Daraufhin folgte eine Aufzählung der Schatten in Online Spielen, wobei zwischen dynamischen und statischen Schatten unterschieden wird.

Bei den dynamischen Schatten handelt es sich um Schatten, die während des Spielens in Echtzeit berechnet werden. Es befindet sich demnach tatsächliches Licht im endgültigen Spiel und die Schatten ändern sich je nach Position und Bewegung der Kamera oder des Charakters. Dazu wird ein Schattenalgorithmus benötigt, welcher sich auf verschiedene Arten berechnen lässt: Planar Shadows, Shadow Mapping oder Shadow Volume. Das Gegenstück dazu bieten die statischen Schatten, die sich während des Spielens nicht verändern und mittels Textur auf ein Objekt gebracht werden. Zwei Verfahren dieser statischen Schatten wurden daraufhin erklärt: Das Baking und das Bemalen der Texturen in Photoshop. Genau diese zwei Methoden werden auch in dieser Arbeit angewendet und miteinander verglichen. Da beim Baking Verfahren die Licht- und Schattenbildungen durch tatsächliche Lichtquellen in der 3D Szene entstehen, welche das Objekt, abhängig von der Auswahl, Positionierung und Einstellungen dieser Lichtquellen, beleuchten und diese Informationen auf eine Textur gespeichert werden, gilt das Baking Verfahren als eine automatisierte Methode zur Licht- und Schattensetzung. Im Gegensatz dazu werden die Lichter und Schatten bei der manuellen Methode per Hand in Photoshop eingezeichnet. Nachdem beide Methoden erläutert wurden, folgten noch drei weitere Einsatzgebiete von Computergrafiken: Film, Medizin und Architektur.

Das nächste Kapitel behandelte die Grundlagen von Lichtverhalten in der Physik, sowie die

verschiedenen Lichtquellentypen, wie sie in 3D Programmen typischerweise zu finden sind. So wird grundsätzlich zwischen Point, Spot, Area und Infinite Light unterschieden. Für welche Lichtquelle man sich entscheidet, hängt von den gewünschten Ergebnissen ab. Die jeweilige Schattenbildungen der einzelnen Lichtquellen wurden am Ende in einer Grafik aufgezeigt. Zum Schluss folgte noch eine kurze Beschreibung der zwei Beleuchtungsmodelle: die lokalen Beleuchtungsmodelle, also Lambert und Phong, und die globalen Beleuchtungsmodelle, also das Ray Tracing Verfahren und die Radiosity Methode.

Im darauffolgenden Kapitel beginnt auch schon die Methodik dieser Arbeit, angefangen bei der Konzipierung. Es wurden die beiden Methoden erläutert, die angewendet und im Nachhinein miteinander verglichen werden sollen. Daraufhin folgte eine Erläuterung zur Auswahl des Referenzitems mit dem passenden Wireframe Rendering. Dabei handelt es sich um ein Kleid, welches aus 502 Polygonen bzw. 950 Triangles besteht und welches von der Firma SMeet GmbH für diese Arbeit zur Verfügung gestellt wurde. Als nächsten kommen die Anforderungen an die verwendete Szene und angewandte Methoden, an die sich während der Umsetzung zu halten war. Daraufhin folgte noch eine Definition für die Vergleichbarkeit der Methoden. Um einen späteren Vergleich aufstellen zu können, müssen die Rahmenbedingungen vorher bestimmt werden. So wurde in diesem Punkt festgelegt, dass alle verwendeten Texturen eine Auflösung von 1024x1024px haben müssen und dem Item ein Lambert Shader zugewiesen wird. Für einen Vergleich der Renderings wurden auch die Rahmenbedingungen für das Rendern der Ergebnisse festgelegt. Dafür wurde ein Skript zur Erstellung einer Studio Beleuchtung benutzt.

Die zuvor ausgearbeitete Konzeptionierung wurde im nächsten Kapitel der Umsetzung eingesetzt. Zunächst einmal wurde die automatische Methode umgesetzt, wofür als erstes eine Auswahl der Lichter in der Szene getroffen werden musste. Nach einigen Fehlversuchen der gebackenen Ergebnisse, wurde das endgültige Licht Setup für die automatische Methode gefunden und bestand aus vier Spot Lights mit unterschiedlichen Einstellungen und zwei Area Lights, zum allgemeinen Aufhellen des Items. Insgesamt befanden sich also sechs Lichtquellen in der Szene. Nachdem die Auswahl der Lichter abgeschlossen war, wurde noch die Qualität der gerenderten Licht- und Schattensetzung verbessert. Zum Schluss wurde die gebackene Textur mit der ursprünglichen Textur in Photoshop zusammengebracht und diese dann abgespeichert. Somit war die automatische Methode abgeschlossen. Als nächstes wurde die manuelle Methode umgesetzt. Hierfür wurde die Textur ohne Schatten in Photoshop geöffnet und mit Hilfe des UV Layouts die ersten groben Schatten eingezeichnet. Danach wurden diese mit dem Wischfinger und Radiergummi Werkzeug so lange angepasst, bis ein zufriedenstellendes Ergebnis erreicht worden ist. Im nächsten Schritt wurden weitere Schatten gesetzt, welche wieder geformt wurden. So wurde nach und nach in insgesamt sechs Schritten die manuelle Methode angewandt und abgeschlossen.

Im letzten Kapitel der Arbeit, abgesehen von diesem, wurden die Ergebnisse vorgestellt. Dabei handelt es sich nicht um die gewonnenen Texturen der jeweiligen Methoden, sondern um einen Vergleich der beiden. Somit wurde der Zeit- und Arbeitsaufwand dargestellt und jeweils miteinander verglichen. Auf diese Weise wurden auch die optischen Unterschiede der jeweiligen Texturen und der Renderings aufgezeigt. Am Ende dieses Kapitels folgte die Evaluierung dieser Methoden, in denen die gewonnenen Erkenntnisse aus den Ergebnissen beschrieben wurden. So besitzt jede Methode ihre eigenen Vor- und Nachteile. Eine physikalische Kor-

rektheit ist zum Beispiel so gut wie nur mit der automatischen Methode zu vergleichen, die auch, subjektiv betrachtet, schönere Ergebnisse liefert, dafür aber einen längeren Zeit- und Arbeitsaufwand benötigt. Dafür ist die manuelle Methode schneller umzusetzen und bietet auch mehr Möglichkeiten zur Darstellung und Manipulation der Schatten, doch die Licht- und Schattenbildungen sind dabei optisch nicht so hochwertig.

Im Rahmen dieser Arbeit wurden beide Methoden lediglich anhand eines Referenzitems verglichen. Um einen besseren Vergleich aufstellen zu können, wäre es noch möglich, beide Methoden an einem Shader Template anzuwenden, sowie an einer Realszene aus einem Spiel, mit mehreren Meshes und komplexeren Texturen.

Glossar

AAA Spiele	Kostenaufwändige, produktiv hochwertige Spiele, welche ein sehr hohes Budget bei der Entwicklung zur Verfügung haben.
Speicherbandbreite	Ist die Datentransferrate von Speicherbausteinen und beschreibt die Anzahl der Bits und Bytes, die in einer Zeiteinheit transferiert werden können.
Speicherlatenzzeiten	Bezeichnet die Zeit, die zwischen einer Anfrage an den Speicher und der Antwort vom Speicher verstreicht.
Next-Gen-Spiele	Bezeichnet im allgemeinen Spiele mit starkem Leistungsanspruch und besserer Grafik für die neuesten Konsolen.
Tiefenpuffer	Eine Tabelle oder Matrix, welches für jedes Fragment im Framebuffer einen Tiefenwert speichern kann. Mit dessen Hilfe kann bestimmt werden, ob ein Fragment eines zu zeichnenden Objektes sichtbar ist oder nicht
Aliasing	Wenn Kanten nicht Glatt sind, sondern einen „Treppenstufen-Effekt“ aufweisen, spricht man vom Aliasing-Effekt. Dieser tritt auf, wenn eine Reduzierung der Informationen in der Computergrafik durchgeführt wird.
Low-Poly	Bezeichnet ein Polygon Mesh in einem 3D Programm, welches aus relativ wenigen Polygonen besteht.
High-Poly	Im Gegensatz zu Low-Poly bezeichnet der Begriff High-Poly ein Polygon Mesh mit einer großen Anzahl an Polygonen.

Literaturverzeichnis

- [Ahe14] AHEARN, L.: *3D Game Textures: Create Professional Game Art Using Photoshop*. Taylor & Francis, 2014 <https://books.google.de/books?id=b1YtBAAAQBAJ>. – ISBN 9781317557982
- [AMHH08] AKENINE-MÖLLER, Tomas ; HAINES, Eric ; HOFFMAN, Naty: *Real-Time Rendering 3rd Edition*. Natick, MA, USA : A. K. Peters, Ltd., 2008. – 1045 S. – ISBN 987-1-56881-424-7
- [Ann08] ANNEN, Thomas: *Efficient Shadow Map Filtering*, Universität des Saarlandes, Doctoral dissertation, 2008
- [Bir00] BIRN, J.: *Digital Lighting and Rendering*. New Riders, 2000 (New Riders Professional Library). <https://books.google.de/books?id=1jX-1PbDH2sC>. – ISBN 9781562059545
- [BL08] BURLEY, Brent ; LACEWELL, Dylan: Ptex: Per-Face Texture Mapping for Production Rendering. In: *Eurographics Symposium on Rendering 2008*, 2008, S. 1155–1164
- [Bla11] BLACKMAN, S.: *Beginning 3D Game Development with Unity: All-in-one, multi-platform game development*. Apress, 2011 <https://books.google.de/books?id=yngwIejtvBEC>. – ISBN 9781430234234
- [Bra04] BRABEC, Stefan: *Shadow Techniques for Interactive and Real-Time Applications*, Universität des Saarlandes, Doctoral dissertation, 2004
- [BT75] BUI-TUONG, Phong: Illumination for Computer Generated Pictures. In: *Graphics and Image Processing* (1975)
- [Buc10] BUCK, Ursula: Laserscanning in der Kriminalistik. In: *zfv* (2010), Nr. 3, 190 - 198. <http://dx.doi.org/http://dx.doi.org/10.1016/j.resuscitation.2007.10.023>. – DOI <http://dx.doi.org/10.1016/j.resuscitation.2007.10.023>
- [CPC⁺08] CHOA, Minhong ; PARK, Incheol ; CHUNG, Hyun S. ; YOO, Sun K. ; SHIM, Hoshik ; KIM, Seungho: The effectiveness of cardiopulmonary resuscitation instruction: Animation versus dispatcher through a cellular phone. In: *Resuscitation* 77 (2008), Nr. 1, 87 - 94. <http://dx.doi.org/>

- org/http://dx.doi.org/10.1016/j.resuscitation.2007.10.023. – DOI
http://dx.doi.org/10.1016/j.resuscitation.2007.10.023. – ISSN 0300–9572
- [Cro77] CROW, Franklin C.: Shadow Algorithms for Computer Graphics. In: *SIGGRAPH Comput. Graph.* 11 (1977), Juli, Nr. 2, 242–248. <http://dx.doi.org/10.1145/965141.563901>. – DOI 10.1145/965141.563901. – ISSN 0097–8930
- [EJK15] EICKE, Tim N. ; JUNG, Yvonne ; KUIJPER, Arjan: Stable dynamic webshadows in the {X3DOM} framework. In: *Expert Systems with Applications* 42 (2015), Nr. 7, 3585 - 3609. <http://dx.doi.org/http://dx.doi.org/10.1016/j.eswa.2014.11.059>. – DOI <http://dx.doi.org/10.1016/j.eswa.2014.11.059>. – ISSN 0957–4174
- [Fre06] FREILING, Jens: *Echtzeitfähige Schatten in Mixed Reality-Umgebungen*, Universität Koblenz-Landau, Campus Koblenz, Diplomarbeit, 2006
- [Gre07] GREEN, Chris: Efficient Self-shadowed Radiosity Normal Mapping. In: *ACM SIGGRAPH 2007 Courses*. New York, NY, USA : ACM, 2007 (SIGGRAPH '07). – ISBN 978–1–4503–1823–5, 1–8
- [Hem04] HEMPE, Nico: Shadow Mapping. In: *Universität Koblenz Landau* (2004)
- [HW10] HABEL, Ralf ; WIMMER, Michael: Efficient Irradiance Normal Mapping. In: *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games 2010*, ACM, Februar 2010. – ISBN 978–1–60558–939–8, 189–195
- [JDL08] J.SCHMIDT ; DREYER, S. ; LAMPERT, C.: *Spielen im Netz: Zur Systematisierung des Phänomens Online-Games*. Arbeitspapiere des Hans-Bredow-Instituts Nr. 19, 2008 <https://www.hans-bredow-institut.de/de/forschung/spielen-im-netz-zur-systematisierung-des-phaenomens-online-games>
- [KBS11] KAVAN, Ladislav ; BARGTEIL, Adam W. ; SLOAN, Peter-Pike: Least Squares Vertex Baking. In: *Computer Graphics Forum* (2011). <http://dx.doi.org/10.1111/j.1467-8659.2011.01991.x>. – DOI 10.1111/j.1467–8659.2011.01991.x. – ISSN 1467–8659
- [Ker04] KERLOW, I.V.: *The Art of 3D: Computer Animation and Effects*. Wiley, 2004 <https://books.google.de/books?id=5G-5iyasSk8C>. – ISBN 9780471430360
- [KOJC12] KERSTEN-OERTEL, Marta ; JANNIN, Pierre ; COLLINS, D. L.: DVV: A Taxonomy for Mixed Reality Visualization in Image Guided Surgery. In: *IEEE Trans. Vis. Comput. Graph.* 18 (2012), Nr. 2, 332–352. <http://dblp.uni-trier.de/db/journals/tvcg/tvcg18.html#Kersten-OertelJC12>
- [Kor04] KORN, Andreas: *Zur Entwicklungsgeschichte und Ästhetik des digitalen Bildes*, Universität Duisburg-Essen, Diss., 2004

- [Lam04] LAMA, T.: *3D-Welten: professionelle Animationen und fotorealistische Grafiken mit Raytracing*. Hanser, 2004 <https://books.google.de/books?id=6VnkiKCrW1gC>. – ISBN 9783446228719
- [NFHS12] NISCHWITZ, A. ; FISCHER, M. ; HABERÄCKER, P. ; SOCHER, G.: *Computergrafik und Bildverarbeitung*. Vieweg+Teubner Verlag, 2012 <https://books.google.de/books?id=LNCjfPfx7QkC>. – ISBN 9783834883001
- [OSK⁺14] OLSSON, Ola ; SINTORN, Erik ; KÄMPE, Viktor ; BILLETTER, Markus ; ASSARSSON, Ulf: Efficient Virtual Shadow Maps for Many Lights. In: *Proceedings of the 18th Meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*. New York, NY, USA : ACM, 2014 (I3D '14). – ISBN 978-1-4503-2717-6, 87–96
- [Sch03] SCHÄFER, Sebastian: Ausarbeitung zum Thema Schatten. In: *Johann Wolfgang Goethe Universität Frankfurt am Main* (2003)
- [Sch04] SCHERFGEN, D.: *3D-Spieleprogrammierung: mit DirectX 9 und C+*. Hanser, 2004 <https://books.google.de/books?id=JTBWAAAACAAJ>. – ISBN 9783446228696
- [Sch07] SCHAIRER, Timo: *Echtzeitschatten und Radiosity*. 2007
- [Sch10] SCHULTHEISS, Daniel: *Spielen im Internet*. Universitätsverlag Ilmenau, 2010. – ISBN 978-3-939473-83-1
- [Sch12] SCH, Henry: Memory efficient light baking. In: *Computers and Graphics* 36 (2012), Nr. 3, 193 - 200. <http://dx.doi.org/http://dx.doi.org/10.1016/j.cag.2011.12.001>. – DOI <http://dx.doi.org/10.1016/j.cag.2011.12.001>. – ISSN 0097-8493. – Novel Applications of VR
- [Wen15] WENDLAND, Carola von: Die Zukunft der Medizin gestalten. 3D-Simulation. In: *MedPLAST* 66 (2015), S. 44–45. – ISSN 0032-1338
- [Wil78] WILLIAMS, Lance: Casting Curved Shadows on Curved Surfaces. In: *SIGGRAPH Comput. Graph.* 12 (1978), August, Nr. 3, 270–274. <http://dx.doi.org/10.1145/965139.807402>. – DOI 10.1145/965139.807402. – ISSN 0097-8930

