

# **Realismus bei der 3D-Charaktermodellierung des Menschen**

Diplomarbeit

zur Erlangung des akademischen Grades  
Diplom-Informatiker

**HUMBOLDT-UNIVERSITÄT ZU BERLIN  
MATHEMATISCH-NATURWISSENSCHAFTLICHE FAKULTÄT II  
INSTITUT FÜR INFORMATIK**

eingereicht von: Schröder, Denis

Gutachter(innen): Herr Prof. Dr. Peter Eisert  
Frau Prof. Dr. Verena Hafner

eingereicht am:

## Inhaltsverzeichnis

<b>1</b>	<b><u>EINLEITUNG</u></b>	<b>1</b>
1.1	BEGRIFFSDEFINITION	1
1.2	MOTIVATION	2
1.3	AUFGABENSTELLUNGEN	3
1.4	GESCHICHTLICHER ÜBERBLICK	4
<b>2</b>	<b><u>KONZEPTION</u></b>	<b>7</b>
2.1	ANATOMIE DES MENSCHEN	7
2.2	FOTOS ALS VORLAGEN	11
2.3	NEUTRALE POSE	12
2.4	FAZIT	13
<b>3</b>	<b><u>MODELLIERUNG</u></b>	<b>14</b>
3.1	STANDARDTECHNIKEN	14
3.2	BOXMODELING	15
3.3	POLYGON-MODELING	15
3.4	POLYGONUNTEILUNGEN	19
3.5	3D-SCANNER	22
3.6	ANWENDUNGEN IN DER INDUSTRIE	23
3.7	FAZIT	24
<b>4</b>	<b><u>TEXTURIERUNG</u></b>	<b>26</b>
4.1	TEXTURMAPPING	26
4.2	TEXTURAUFLÖSUNG	28
4.3	TEXTURKANÄLE	29
4.4	BESONDERHEIT HAUT	34
4.5	ANWENDUNGEN IN DER INDUSTRIE	36
4.6	FAZIT	37
<b>5</b>	<b><u>HAARE</u></b>	<b>39</b>
5.1	FESTE HAARGEOMETRIE	39
5.2	HAARTEXTUREN	39
5.3	HAARSIMULATION	40

---

<b>5.4 ANWENDUNGEN IN DER INDUSTRIE</b>	<b>42</b>
<b>5.5 FAZIT</b>	<b>43</b>
<b>6 UMGEBUNG</b>	<b>45</b>
<hr/>	
<b>6.1 BILDBERECHNUNG</b>	<b>45</b>
<b>6.2 LICHT</b>	<b>48</b>
<b>6.3 SCHATTEN</b>	<b>54</b>
<b>6.4 ÄUßERE EINFLÜSSE</b>	<b>62</b>
<b>7 REALISMUS</b>	<b>66</b>
<hr/>	
<b>7.1 BEGRIFF REALISMUS</b>	<b>66</b>
<b>7.2 EXPERIMENTELLE UNTERSUCHUNG</b>	<b>67</b>
<b>7.3 ERFAHRUNG BEEINFLUSST WAHRNEHMUNG</b>	<b>69</b>
<b>7.4 EINFLUSS DER DETAILS AUF DEN REALISMUS</b>	<b>71</b>
<b>7.5 WIRKUNG DER UMGEBUNGSEINFLÜSSE</b>	<b>75</b>
<b>7.6 VERGLEICH</b>	<b>80</b>
<b>7.7 SCHLUSSWORT</b>	<b>82</b>
<b>8 AUSBLICK</b>	<b>84</b>
<hr/>	
<b>8.1 REALISMUSUNTERSUCHUNG</b>	<b>84</b>
<b>8.2 REALISMUS BEI ANIMATIONEN</b>	<b>84</b>
<b>A REFERENZEN</b>	<b>I</b>
<hr/>	
<b>B TABELLEN- UND ABBILDUNGSVERZEICHNIS</b>	<b>V</b>
<hr/>	
<b>C ABKÜRZUNGEN</b>	<b>VII</b>
<hr/>	
<b>D ANHANG</b>	<b>VIII</b>
<hr/>	

# 1 Einleitung

Virtuelle Charaktere sind in unserer heutigen Welt überall anzutreffen und an vielen Stellen gar nicht mehr wegzudenken. Im Kino oder im Fernsehen erobern virtuelle Charaktere unsere Herzen, während sie in Computerspielen mit uns agieren oder als Werbefigur unsere Aufmerksamkeit erringen wollen. Oft erkennt der Betrachter nicht mehr ob es sich um einen realen oder um einen virtuellen Charakter handelt. In Kinoproduktionen werden seit vielen Jahren die echten Schauspieler in gefährlichen oder unmöglichen Situationen durch ihre virtuellen Doubles ausgetauscht. Bei den Dreharbeiten zu dem Film *The Crow* (1994, Crowvision Inc. / Edward R. Pressman Film / Miramax Films) wurde der Hauptdarsteller Brandon Lee tödlich verletzt. Um die fehlenden Szenen des Filmes dennoch abdrehen zu können, wurde sein Gesicht digital über das eines Doubles gelegt.

Eine Forderung nach virtuellen Charakteren, die von ihren originalen menschlichen Vorbildern nicht mehr zu unterscheiden sind, ist die Folge. Gefordert sind virtuelle Charaktere, denen man ihre computergenerierte Herkunft nicht mehr auf den ersten Blick ansieht. Gefordert sind realistische virtuelle Charaktere.

## 1.1 Begriffsdefinition

Bevor wir dieser Forderung nach realistischen virtuellen Charakteren nachgehen können, sollte deren Begrifflichkeit zum besseren Verständnis und zu einer besseren Abgrenzung untersucht werden.

- Der Begriff „*virtuell*“ kommt ursprünglich aus dem Französischen und bezeichnet die Eigenschaft „nicht physikalisch vorhanden“ zu sein. Eine wirkliche Existenz in materiellem Sinn ist nicht vorhanden, diese Existenz ist nur scheinbar. Wenn von *virtuell* die Rede ist, sind meist CGI gemeint.
- Der englische Begriff „*CGI*“ steht für *Computer Generated Images* und bezeichnet die mit dem Computer erzeugten Bilder. Diese Definition findet hauptsächlich in der Filmtechnik Verwendung und wird unter den Spezialeffekten (Special Effects) zusammengefasst.
- Der Begriff „*Charakter*“ geht in seiner Herkunft in das Griechische zurück und bedeutete ursprünglich „das Geprägte“, später aber auch „Abdruck“ und „Eigenart“. Heute wird mit *Charakter* eine beliebige „Hülle“ gemeint, die eine Eigenart, eine Persönlichkeit aufweist. Ein „*Charakter*“ muss dabei keine humanoide Person sein, sondern kann genauso ein Tier, ein Fantasiewesen oder aber auch ein unbelebtes Objekt sein.
- Der Begriff „*Realismus*“ geht auf das lateinische Wort „*res*“ zurück, welches „das Ding“ oder „die „Sache“ bezeichnet und auf das Vorhandensein eines materiellen Bezugs hinweist. Als *realistisch* wird etwas bezeichnet, was der Wirklichkeit entspricht.<sup>1</sup>

---

<sup>1</sup> für eine weitergehende Begriffserklärung vgl. Kapitel 7.1 Begriff Realismus

Realistische virtuelle Charaktere bezeichnen zusammengefasst die vom Computer erzeugten und für den Betrachter zum Leben erweckten Persönlichkeiten, die von ihren Vorbildern nicht mehr zu unterscheiden sind.

## 1.2 Motivation

Die Forderung nach einer realistischen Darstellung einer menschlichen Person stellt einen 3D-Grafiker vor eine schwere Aufgabe. Mit Hilfe von Werkzeugen, die zur Modellierung und Animation zur Verfügung stehen, lassen sich ohne Probleme in Sekunden Objekte erzeugen, die aus einfachen geometrischen Formen bestehen. Innerhalb kürzester Zeit ist auch die Erzeugung komplexer Geometrien kein Problem. Die Modellierung eines realitätsgetreuen Gesichtes ist dagegen eine Aufgabe, die schwer zu beherrschen ist [Par]. Besonderes Augenmerk soll bei der Erstellung auf dem Realismus liegen. Warum gibt es scheinbar perfekte menschliche Abbilder und dennoch sieht man sofort die computergenerierte Komponente? Wo liegt das Hauptproblem, dass menschliche Ebenbilder nicht als solche wahrgenommen werden?

Die Schwierigkeit besteht darin, dass wir tagtäglich andere Menschen sehen. Wir sind sehr gut auf deren Aussehen und deren Bewegungen trainiert. Dadurch erkennt man sehr schnell, ob es sich um einen Menschen handelt oder doch nur um eine computergenerierte Nachbildung. Jeder kleinste Fehler wird erkannt, Kleinigkeiten reichen meist aus. Jedes einzelne Detail muss perfekt sein, um einen Betrachter zu überzeugen.

»Wenn es nach Fisch riecht, dann ist der Fisch nicht mehr frisch. Ähnliches gilt in der Computergrafik: Wenn es nach Computergrafik aussieht, ist es ein schlechtes Bild.« [Bir]

Darüber hinaus gibt es bei der Erstellung realistischer menschlicher Abbilder ein weiteres Problem mit dem sogenannten *Uncanny Valley*<sup>2</sup>. Hierunter versteht man den Umstand, dass der Betrachter künstlich erschaffene Figuren nur bis zu einem bestimmten Grad als realistisch empfindet und diese als solche akzeptiert. Ab einem bestimmten Grad an Realismus schwenkt diese Akzeptanz zu einer Ablehnung, bevor sie bei einer fast komplett realistisch wahrgenommenen Figur wieder als positiv wahrgenommen wird. Dieser Umstand könnte die Ursache sein, warum Filme wie „Final Fantasy: The Spirits Within“ (2001, Square Company) mit ihren fast perfekten menschlichen Akteuren weniger erfolgreich waren und warum Filmstudios lieber einen comicartigen und niedlichen Stil für ihre Figuren vorziehen, wie es zum Beispiel bei den Pixar-Filmen zu erkennen ist.

---

<sup>2</sup> Der Begriff *Uncanny Valley* wurde geprägt vom Robotikexperte Masahiro Mori [Mor].

### 1.3 Aufgabenstellungen

Der erste Teil dieser Diplomarbeit soll der Entwicklung eines virtuellen Charakters nachgehen. Die Modellierung eines menschlichen Charakters soll genauer beleuchtet werden. Dabei sollen besonders folgende Aufgabenstellungen untersucht und gelöst werden:

1. **Aktuelle Techniken:** Welche Techniken gibt es derzeit um einen 3D-Charakter erstellen zu können? Aktuelle Techniken sollen hierbei benannt und vorgestellt werden.
2. **Technikanalyse:** Diese benannten Techniken sollen hinsichtlich der Verwendung zur Herstellung virtueller Charaktere analysiert und bewertet werden. Besonderes Augenmerk soll dabei immer auf der Realismuswirkung der betrachteten Technik liegen. Welche Techniken liefern die realistischsten Ergebnisse? Ein weiterer wichtiger Aspekt bei der Analyse soll auf der zu erwarteten Renderzeit liegen. Sind die Renderzeiten einer Technik astronomisch hoch, liefert auch das realistischste Ergebnis keine effektive Technik mehr. Die Effektivität hinsichtlich des Kosten-/Nutzen-Faktors sollte immer im Vordergrund liegen.
3. **Techniken der Industrie:** Welche Techniken werden zum Beispiel in der Filmbranche bei der menschlichen Nachbildung angewendet? Einige Beispiele sollen hier aufgezeigt werden, sowohl geschichtlich als auch aktuell betrachtet.
4. **Erstellung eines Charakters:** Um eine Bewertung der Techniken durchführen zu können, soll im Rahmen dieser Diplomarbeit ein eigener virtueller Charakter mit den vorgestellten Techniken aufgebaut werden. Nur hierdurch kann eine Beurteilung hinsichtlich Anwendbarkeit und Effektivität erreicht werden.

Hierbei werden die gängigsten aktuellen Techniken zur Herstellung eines virtuellen Charakters betrachtet. Dazu gehören die verschiedenen Modellierungswerkzeuge um eine Grundgeometrie aufzubauen und anschließend weiter zu verfeinern. Desweiteren werden die Mappingverfahren, die verschiedenen Texturkanäle und die Möglichkeiten der Haarerzeugung erläutert. Abschließend wird sich dem Aufbau einer fertigen Szene mit Licht, Schatten und Umgebung gewidmet.

Die Technikauswahl für die Erstellung eines virtuellen Charakters wird heutzutage von den meisten Firmen subjektiv aus eigener Erfahrung getroffen. Der Herstellungsprozess wird meist von Character-Artists der Firmen übernommen, die anhand eines Zeitplans und eines Budgets die Herstellung planen und durchführen. Eine Produktion anhand einer berechneten Effektivität hinsichtlich eines Kosten-/Nutzen-Faktors der einzusetzenden Techniken findet nach meiner Erfahrung nicht statt<sup>3</sup>. Im zweiten Teil dieser Diplomarbeit soll auf diese Problematik eingegangen werden:

---

<sup>3</sup> Der Autor arbeitet seit mehreren Jahren als 3D-Grafiker für verschiedene Firmen.

5. **Realismusanalyse der Techniken:** Die Techniken sollen aufgrund ihres Realismusgewinns untersucht und kategorisiert werden. Zusätzlich soll eine Kosten-Nutzen-Analyse erfolgen, um die zu untersuchenden Techniken hinsichtlich ihrer Effektivität einordnen zu können. Unter Effektivität wird hierbei der Realismusgewinn hinsichtlich der Renderzeit verstanden.

Realismus ist bisher keine fassbare und keine messbare Größe. Durch diese Realismusanalyse soll sich dies ändern und die Modellierungstechniken sollen sich messbar einordnen lassen. Dadurch soll es für Firmen in Zukunft möglich sein, die Techniken anhand ihrer Effizienz einzusetzen und folglich den ganzen Erstellungsablauf effizienter und damit auch kostengünstiger zu gestalten. Zusätzlich sollen auch Hersteller von Renderprogrammen ihre Algorithmen anhand der Wirkung der einzelnen Funktionen optimieren.

Um den Rahmen einer Diplomarbeit nicht zu sprengen, werden in dieser Arbeit nur Stillbilder und keine Animationen betrachtet. Es wird an einigen Stellen an die Einsatzfähigkeit bei Animationen verwiesen und einige Hinweise dazu gegeben. Ferner konzentriert sich diese Arbeit mehr auf die Erstellung eines realistischen Kopfes, wenngleich auch ein kompletter Charakter aufgebaut wird und die nötigen Techniken dazu besprochen werden, die auch auf den ganzen Körper angewendet werden können.

Aussagen anderer Autoren wurden als solche kenntlich gemacht. Alle anderen Aussagen sind die Meinung des Autors und gehen auf dessen eigene Erfahrung zurück oder gelten als allgemein anerkannte Aussagen.

## 1.4 Geschichtlicher Überblick

Die Entwicklung realistischer virtueller Charaktere geht geschichtlich gesehen weit zurück. Schon vor 30.000 Jahren versuchte der Mensch sich bildlich als Höhlenmalerei nachzubilden und kann damit als erster Vorläufer gesehen werden. Generell kann die Malerei als Nachahmung realer Welten gesehen werden. Die erste geschichtliche Auseinandersetzung mit computergenerierten Bildern entstand in den 50er Jahren an den Bell Labs & Universities. In den folgenden Jahren wurden Computer zur Bildentwicklung allerdings hauptsächlich nur im universitären Umfeld eingesetzt.

In der Filmbranche wurden computergenerierte Bilder erst viel später verwendet. Da in dieser Branche für gewöhnlich ausreichend finanzielle Mittel zur Verfügung stehen bot sie das ideale Umfeld um solche Technologien zu entwickeln. Einige wichtige Entwicklungen sollen diesen geschichtlichen Fortschritt aufzeigen:

**1962** Ivan Sutherland entwickelten am Massachusetts Institute of Technology (MIT) mit *Sketchpad* das erste Computergrafikprogramm [Sut]. Mit Sketchpad war die direkte Eingabe von einfachen geometrischen Objekten in den Computer möglich.

**1969** entwickelte Ivan Sutherland und David Evans das erste Drahtgitter-Zeichnungs-System CAD<sup>4</sup>.

**1976** entstand die erste Computergrafik der Filmgeschichte: Edwin Catmull erstellte für den Film *Future World* (American International Pictures) ein Drahtgittermodell eines Kopfes und einer Hand, die in diesem Film auf einem Monitor gezeigt wurden. Es war keine direkte Integration von computergenerierter Grafik, es zeigte der Filmindustrie aber neue Möglichkeiten auf. [Riz]

**1981** wird das erste eigenständige Rendering System REYES<sup>5</sup> von Loren Carpenter und Robert L. Cook für LucasFilm entwickelt. Bemerkenswert ist, dass REYES später zu RenderMan weiterentwickelt wurde und sich ein Teil der Lucasfilm's Computer Graphics Research Group sich zu Pixar entwickelte [Wol].

**1982** erscheint in *Star Trek: The Wrath of Kahn* (Paramount Pictures) die erste computergenerierte Sequenz der Filmgeschichte. In einer 60-sekündigen Sequenz wurden zum ersten Mal Fraktale und Partikeleffekte gezeigt.

**1986** ist der erste computergenerierte 3D-Charakter in *Young Sherlock Holmes* (Pixar) zu betrachten. Zu sehen war ein Ritter, der aus einem Fensterglas zum Leben erweckt wurde. Im gleichen Jahr benutzte der Kurzfilm *Luxo Jr.* (ebenfalls Pixar) die ersten computergenerierten Schatten der Filmgeschichte.

**1988** war der Kurzfilm *Tin Toy* (Pixar) der erste digitale Film, in dem ein menschlicher Charakter dargestellt wurde [Par]. Zudem war die Komplexität der Szene ungewöhnlich hoch.

**1989** erschuf ILM<sup>6</sup> für *The Abyss* (Twentieth Century Fox / Lightstorm Entertainment) eine digitale organische Filmkreatur mit einem menschlichen Gesicht. Seit diesem Zeitpunkt wird die Computergrafik dazu verwendet, glaubhafte Spezialeffekte in Filme einzubauen.

**1993** wurden zum ersten Mal durch den Computer erschaffene realistische wirkende Tiere in *Jurassic Park* (Amblin / Universal Pictures) gezeigt.

---

<sup>4</sup> CAD steht für *Computer Aided Design*

<sup>5</sup> REYES steht für *Rendering Everything You Ever Saw*

<sup>6</sup> ILM (Industrial Light and Magic) wurde 1977 von George Lucas für die Effekte von *Star Wars* (1977, 20th Century Fox) gegründet.

**1995** *Toy Story* (Disney / Pixar) war der erste komplett computererschaffene abendfüllende Film in der Kinogeschichte. Mit *Judge Dredd* (Cinergi Pictures) kommt der erste Film in die Kinos in dem die Stuntleute durch 3D-Modelle ersetzt wurden.

**1996** erschuf ILM in *Jumanji* (Tri-Star) Tiere mit einer sehr realistischen Haarsimulation und erreichte damit eine neue Stufe an Realismus.

**1997** in dem Kurzfilm *Geris Game* (Pixar) werden zum ersten Mal Subdivision-Surfaces für einen virtuellen Charakter benutzt.

**2001** wurde mit *Final Fantasy: The Spirits Within* (2001, Square Pictures) der erste Versuch unternommen einen komplett mit dem Computer erzeugten Kinofilm mit fotorealistischen menschlichen Charakteren zu erschaffen. [Sto]

## 2 Konzeption

»Die Entwicklung eines virtuellen Charakters ist ein facettenreicher und äußerst komplexer Prozess.« [Olm]

Um einen realistischen Charakter zu erschaffen ist es notwendig, die gesamte Vorgehensweise vorher genau zu planen. Eine unkoordinierte Arbeitsweise führt in den seltensten Fällen zu den gewünschten Ergebnissen. Erst durch eine genaue Planung kann die Erzeugung eines komplexen, realistischen, virtuellen Charakters effektiv bezüglich der eingesetzten Ressourcen und hinsichtlich des Ergebnisses werden. Und letztendlich bedeutet auch die effektive Erschaffung eines virtuellen Charakters die optimale Ausnutzung des Budgets des Auftraggebers.

Einer der wichtigsten Punkte überhaupt ist hierbei eine gute Vorlage, nach der der virtuelle Charakter erstellt wird. Dies können Zeichnungen sein<sup>7</sup>, als idealere Vorlage gelten jedoch Fotos<sup>8</sup>. Sie bilden eine sehr gute Grundlage, um die reale Welt nachzubauen.

### 2.1 Anatomie des Menschen

#### 2.1.1 Proportionen des Menschen

Auf die richtigen Körperproportionen zu achten gehört bei der Charaktererstellung zu den wichtigsten Aspekten. Hier entscheidet sich schnell, ob eine Figur realistisch aussieht oder eben nicht. In unserer Umgebung sehen wir tagtäglich Menschen und wissen wie sie auszusehen haben. Um ein Fantasiewesen zu erschaffen, muss man nicht sehr auf die Körperproportionen achten – schließlich gibt es keine Referenzen, wie es auszusehen hat. Man kann aber sehr schnell erkennen, ob ein Charakter menschlich aussieht oder nicht.

Proportional heißt bestimmte Werte in Verhältnis zueinander setzen. Auf Verhältnisse zu achten gehört bei der Charaktererstellung zu den wichtigsten Themen. Allerdings muss an dieser Stelle auch erwähnt werden, dass nicht jede Regel zwingend genau angewendet werden soll. Sie dienen eher als Leitfaden und nicht als Vorschrift. Jeder Mensch hat sein eigenes Erscheinungsbild und sieht anders aus. Nur das macht ihn zum Individuum. Die folgenden Richtlinien erlauben mit einfachen Möglichkeiten einen menschlichen Charakter zu erstellen und mit leichten Abänderungen verschiedene Persönlichkeiten herauszuarbeiten.

##### 2.1.1.1 Körperproportionen

Schon im antiken Griechenland befasste man sich gründlich mit dem menschlichen Aufbau und mit

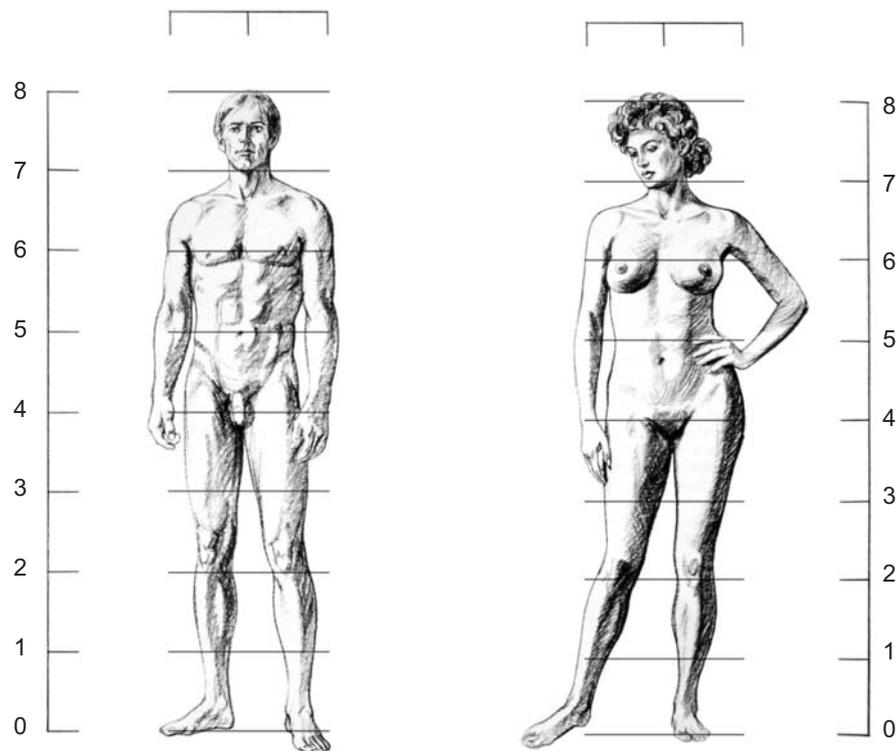
---

<sup>7</sup> vgl. Kapitel 2.1 Anatomie des Menschen

<sup>8</sup> vgl. Kapitel 2.2 Fotos als Vorlage

dessen Proportionen zueinander. Als einer der wichtigsten Vertreter hierfür wäre Polyklet zu nennen. Er war ein berühmter griechischer Bildhauer, der im 5. Jh. v. Chr. lebte und heute noch gültige Grundregeln im menschlichen Körperbau aufstellte.

Polyklet entwickelte ein Maßsystem, wonach jeder menschliche Körper in einem Verhältnis zum Kopf gesehen wird, den er als Kanon bezeichnete. Ein Kanon ist als eine Regel zu verstehen, um einen Körper mit Hilfe eines Maßes oder einer Proportion aufzuteilen. Ein Kanon wird also als eine Hilfseinheit zur Ermittlung anderer Maße benutzt. Polyklets Vorstellung nach war ein Mann  $7\frac{1}{2}$ -mal so groß wie dessen Kopf – der „ $7\frac{1}{2}$ fache Kanon“. Den Kopf als zentrales Element zu verwenden scheint auch logisch, da dieser immer zuerst modelliert wird und danach die anderen Körperteile.



**Abbildung 2-1: Proportionen des Menschen:** Unterschiede bei Mann und Frau. Gerastert nach der Kopfhöhe.

Im Laufe der Jahrhunderte änderte sich die ideale Vorstellung eines Menschen und die Menschen wurden größer. Für unsere heutige Zeit wirkt ein Mensch nach dem  $7\frac{1}{2}$ fachen Kanon eher als „etwas klein geraten, misst ca. 165 bis 170 cm und ist von eher unscheinbarer Gestalt. Sein Kopf ist relativ groß im Vergleich zum Rest des Körpers“ [Deb].

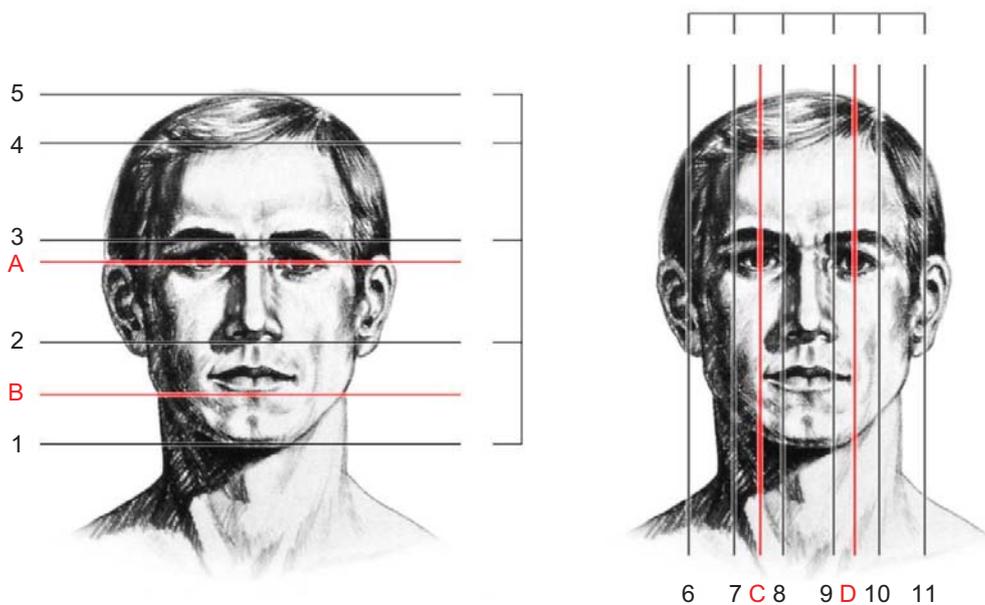
Der heutige Mann in Europa hat eine durchschnittliche Größe zwischen 169,8 cm in Portugal und 181,7 cm in den Niederlanden. In Deutschland misst der durchschnittliche Mann 177,0 cm [Ker]. Bezieht man dies mit ein, ist der 8fache Kanon der richtige Ausgangspunkt, um einen menschlichen

Charakter männlicher Gestalt unserer Zeit zu erschaffen.

Gemäß dem 8fachen Kanon verdeutlicht die Abbildung 2-1 diese Unterteilung. Weitere Merkmale können nun anhand dieser Rasterlinien leicht positioniert werden. Hiernach liegen die Kniescheiben knapp über der zweiten Rasterlinie. Die vierte Linie markiert bei dem 8fachen Kanon die exakte Mitte des Menschen. Hier sollten sich die Hüfte und der Schritt befinden. Außerdem enden hier ungefähr die Handgelenke. Eine Kopfhöhe weiter oben befindet sich die Taille (5) und noch einen Kopf weiter oben sollte sich die breiteste Stelle der Brust befinden (6). Zu guter Letzt folgt der Kopf (7).

### 2.1.1.2 Kopfproportionen

Auch bei den einzelnen Kopfproportionen kann man sich an die Vorarbeiten der antiken Gelehrten halten. So wie man den gesamten menschlichen Körper unterteilen kann, wird auch der Kopf in mehrere Abschnitte unterteilt, die in bestimmten Verhältnissen zueinander liegen. Eine grundlegende Basislänge beim Kopf bildet hier die Höhe der Stirn. Hiernach ist der gesamte Kopf  $3\frac{1}{2}$ -mal so hoch wie die Stirnhöhe. Diese Einteilung bestimmt wiederum mehrere Rasterlinien, welche die grobe Positionierung wiedergeben. Abbildung 2-2 verdeutlicht diese Einteilung:



**Abbildung 2-2: Kopfproportionen:** Horizontale und vertikale Rasterlinien zur Ermittlung der Proportionen.

Die erste Rasterlinie begrenzt den Kopf nach unten (1). Weiter nach oben folgen die untere Kante der Nase (2), die Augenbrauen (3) und der Haaransatz (4) bevor die letzte Rasterlinie schließlich die obere Grenze des Kopfes abschließt (5). Als erweiterte Grundregel gilt die Position der Augen: Sie liegen vertikal in der Mitte des Kopfes (A), während eine weitere Linie mittig gelegen zwischen (1) und (2) die Unterkante des Mundes bestimmt (B).

Auch in der Breite des Kopfes kann man Größen in Verhältnisse setzen. Als erste grobe Regel gilt, dass der Kopf  $2\frac{1}{2}$ -mal so breit ist wie die Stirnhöhe. Als zweite hilfreiche Größe gilt hier die Breite des

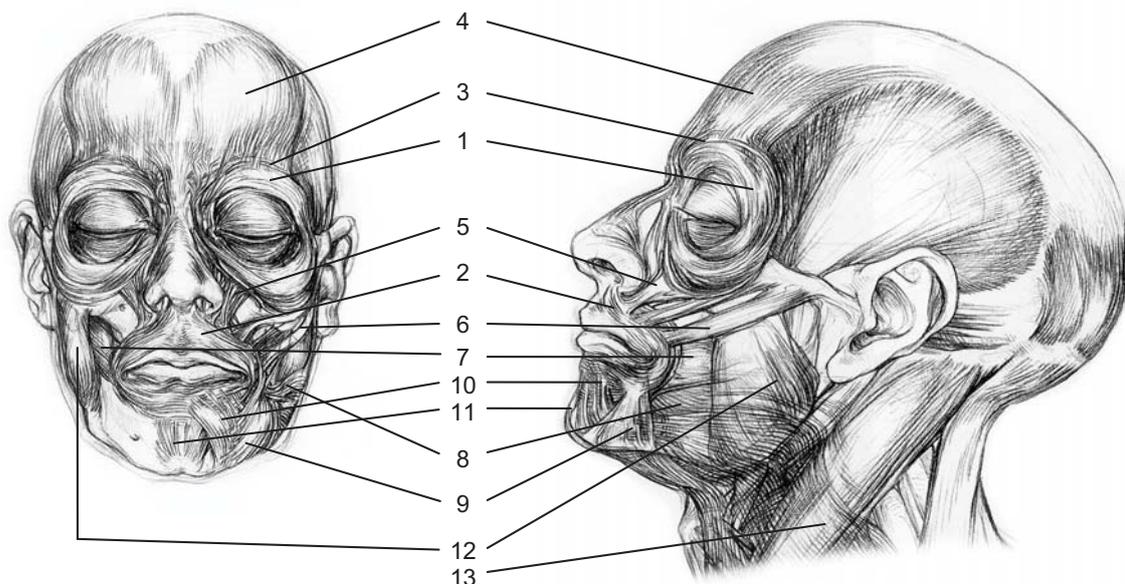
Auges. Hiernach wird der Kopf durch weitere Rasterlinien in fünf gleichgroße Teile geteilt. Die erste begrenzt den Kopf rechts außen (6), während die nächsten beiden Linien (7 und 8) das rechte Auge umschließen. Eine Augensbreite weiter folgt die Nase (zwischen 8 und 9), während die zwei nächsten Linien das linke Auge umschließen (9 und 10). Schließlich folgt der linke Abschluss des Kopfes (11). Zwei weitere hilfreiche Rasterlinien (C und D) gehen genau durch die Mitte beider Augen hindurch (mittig zwischen 7 und 8 bzw. 9 und 10). Diese Linien gehen nicht nur durch die Pupillen hindurch, sie geben auch die Breite des Mundes an.

Und auch im Profil gibt die Stirnhöhe weitere Hilfslinien für die Proportionen an. Der Kopf kann seitlich gesehen auch wieder in  $3\frac{1}{2}$  Teile geteilt werden, die so breit wie die Stirnhöhe sind, was zusammenaddiert wiederum exakt der Kopfhöhe entspricht. Der vordere Teil umfasst im Profil das gesamte Gesicht, während bei der zweiten Rasterlinie das Ohr positioniert wird. Mit all diesen Regeln zusammengefasst können nun alle Proportionen richtig zugeteilt werden.

### 2.1.2 Muskeln und Skelett

Um ein realistisches Ergebnis bei der Charaktererstellung zu erzielen, ist das Wissen von Muskelverläufen und vom Skelettaufbau vorteilhaft. Mit dessen Hilfe kann man die Positionierung der Polygone besser planen. Der Polygonverlauf sollte dem der Muskeln entsprechen, um die besten Ergebnisse zu erhalten. Auch für eine eventuelle Animation ist dies von Vorteil. Eine Veränderung der Geometrie auf Basis der Muskelverläufe wirkt realistisch und bildet keine Artefakte an unerwarteten Stellen. Falten sollten dort verlaufen, wo man sie erwartet.

Deshalb sollte man sich mit den Muskelverläufen genau befassen und die Polygonverläufe vor dem Modellieren weitestgehend planen.



**Abbildung 2-3: Gesichtsmuskeln:** Überblick über die wichtigsten Muskeln des menschlichen Gesichtes.

Schaut man sich die Gesichtsmuskeln an, so sind die ersten bedeutsamen Muskeln, deren Verläufe man kennen sollte, die der Augenringmuskeln *Orbicularis Oculi* (1) und die der Ringmuskeln des Mundes *Orbicularis Oris* (2). Die Polygone sollten in diesem Bereich ringförmig entsprechend den Muskelverläufen positioniert werden. Oberhalb der Augen liegt der *Corrugator Supercilii* (3) bei den Augenbrauen, der die Falten für einen zornigen Blick verursacht. Oberhalb folgt der große Stirnmuskel *Venter Frontalis* (4), der die großen Stirnfalten bilden kann. Weitere wichtige Muskeln, die für den Gesichtsausdruck verantwortlich sind wären die beiden Muskeln des Oberlippenhebers *Levator Labii Superioris* (5) und die beiden seitlichen Jochbeinmuskeln *Zygomaticus Major* und *Minor* (6), welche den Mund nach oben ziehen können. Seitlich folgen der Backenmuskel *Buccinator* (7) und der Lachmuskel *Risorius* (8), die den Mund zur Seite ziehen können. Der Mund wird weiterhin durch den *Depressor Anguli Oris* (9) und den *Depressor Labii Inferiores* (10) beeinflusst, die den Mund nach unten ziehen können. Weitere wissenswerte Muskeln sind der Kinnmuskel *Mentalis* (11), der Kaumuskel *Masseter* (12) und der Kopfnicker *Sternocleidomastoideus* (13).

Auch sollte man bei der Erstellung von virtuellen Charakteren den Skelettaufbau des Menschen kennen. An einigen wenigen Stellen ist es sogar vorteilhaft die Knochen mitzumodellieren. Spätestens aber bei der Animation muss man das menschliche Skelett nachbauen. An dieses geschaffene Skelett wird schließlich die Geometrie gebunden, um sie dann leicht und präzise zu verformen. Je genauer das Skelett der Wirklichkeit entspricht, desto besser sind schließlich die Ergebnisse.

## 2.2 Fotos als Vorlagen

Einen real existierenden Menschen als Vorlage nutzen zu können, bietet eine einfache Möglichkeit schnell Fehler zu erkennen und zu verbessern. Diese Diplomarbeit stellt sich zum Ziel einen möglichst realistischen 3-dimensionalen Charakter so detailliert und dennoch effizient wie möglich zu erstellen. Aus diesem Grund wird als Vorlage ein real existierender Mensch bevorzugt, um später die Unterschiede zwischen Vorlage und dem Ergebnis besser herausfinden zu können.



**Abbildung 2-4: Kopfvorlagen:** Fotos aus vier verschiedenen Blickwinkeln.

Realistisch bedeutet wirklichkeitsnah und glaubhaft zu sein. Um etwas Wirklichkeitsnahes zu erschaffen, bringt das direkte Nachbauen eines Originals die besten Resultate, da man das

erschaffene Ergebnis zu jeder Zeit mit dem Original vergleichen kann. So werden Fehler schnell erkannt und können korrigiert werden.

Als Besonderheit bei einer real existierenden Person ist anzumerken, dass es hier keine echte Symmetrien gibt. Kein Mensch hat exakt sich gleichende Gesichtshälften. Da man einen virtuellen Kopf aufgrund des sonst vorhandenen doppelten Aufwandes allerdings nur zur Hälfte aufbaut und später spiegelt, wird man nie sofort beim Modellieren das gewünschte Ergebnis erhalten. Man muss später noch etwas nachkorrigieren, was allerdings auch den letzten Schliff Realismus in einen virtuellen Charakter bringen kann.

Es empfiehlt sich mindestens eine Frontalansicht und eine Profilansicht als Vorlage zu nehmen. Eine weitere Vorlage, welche im 45°-Grad-Winkel zwischen Frontal- und Profilansicht aufgenommen wird, erweist sich als hilfreich, um Fehler zu finden und zu korrigieren. Diese Vorlagen müssen auf mögliche perspektivische Verzerrungen untersucht und mit einem einfachen 2D-Programm auf eine einheitliche Größe gebracht werden. Weitere Bearbeitungen wie Helligkeitsanpassungen, Kontrastregulierungen und sonstige Bereinigungen erweisen sich als vorteilhaft.

Erst danach werden die Vorlagen in ein 3D-Programm eingeladen und in der entsprechenden Kameraausrichtung jeweils auf einer Fläche positioniert.

## 2.3 Neutrale Pose

Zuerst wird ein Objekt nach einer bestimmten Vorlage kreierte und anschließend für eine eventuelle spätere Animation vorbereitet. Bei dem sogenannten Rigging bekommt dieser Charakter dann ein Skelett, an dem die Geometrie ausgerichtet wird. So wird dann zum Beispiel die Handgeometrie an das Handskelett gebunden. Sind die Punkte/die Vertexe zu eng zueinander aufgebaut, so dass es zu Überschneidungen kommt, ist diese Zuordnungsarbeit äußerst kompliziert. Werden die Vertexe beim Riggen an ein falsches Bone<sup>9</sup> verwiesen, kommt es bei der Animation zu ungewollten Verformungen der Geometrie.

Um diese Überschneidungen möglichst auszuschließen, wird die neutrale Pose benutzt. Hierbei steht die Person der Vorlage in aufrechter Haltung mit beiden Armen zur Seite ausgestreckt und möglichst nicht mit den Beinen eng zusammen. Die Hände sollten keine Faust bilden. Diese Haltung sollte bei der Erstellung der Vorlagen immer mit eingeplant werden.

Hierbei wird schon das sogenannte „*Modeling for Animation*“ angewendet. Man baut ein Objekt im Hinblick auf dessen Verwendung auf. Die Geometrie eines Charakters wird optimalerweise so angelegt, dass bei einer späteren Animation möglichst wenig Verformungen entstehen. Weitere Vorteile bei dieser neutralen Pose zeigen sich bei der Erzeugung von Kleidung. Kleidung kann besser an Charaktere angepasst werden, die sich in der neutralen Pose befinden.

---

<sup>9</sup> Als Bone wird ein Skeletteil verstanden, durch das eine 3D-Geometrie verformt und animiert werden kann.



**Abbildung 2-5: Körpervorlagen:** Neutrale Pose der Vorlage (links) und Referenzbilder aus drei verschiedenen Blickwinkeln.

## 2.4 Fazit

Die Erstellung eines realistischen virtuellen Charakters sollte aufgrund seiner Komplexität vorher genau geplant sein. Zweck und Einsatzort sollten in den Erstellprozess mit eingerechnet werden. Desweiteren sollte der Detailgrad des Objektes vorher definiert werden. Eine genaue Vorplanung verringert die Erstellungszeit und reduziert die späteren anfallenden Probleme und die möglichen anfallenden Nacharbeiten.

Das Wissen über den anatomischen Aufbau des Menschen sollte Verwendung finden. Nur ein korrekt aufgebauter Charakter kann als realistisch wahrgenommen werden. Proportionsregeln helfen beim Aufbau, sollten allerdings auch nur als Anhaltspunkte zur Konstruktion genommen werden. Diese Proportionsregeln spiegeln ein Idealbild wieder. Manchmal wirken kleine Abweichungen von diesen Regeln natürlicher und somit realistischer.

Letztendlich ist die Erzeugung eines virtuellen Charakters ein künstlerischer Vorgang. Ein 3D-Artist bekommt eine Vorlage und weitere Vorgaben, nach denen er das gewünschte Ergebnis möglichst zielgenau umsetzt. Auf der anderen Seite kann der 3D-Artist nur einen so guten 3D-Charakter erschaffen, wie gut seine Vorlage ist. Als weiterer Qualitätsfaktor ist das eingesetzte Renderprogramm entscheidend. Dies wiederum ist abhängig von den verwendeten Techniken.

Zusammengefasst beeinflussen folgende Faktoren qualitativ das Ergebnis:

- Vorlage
- 3D-Artist
- Modellierungstechniken
- Renderalgorithmus

## 3 Modellierung

Das Wort Modell entstammt dem italienischen Wort *modulus* und bedeutet Form oder Muster. Modellieren bezeichnet hiermit das *in Form bringen*. Umgangssprachlich wird mit *Modellieren* die Erstellung eines meist vereinfachten Abbildes der Realität bezeichnet.

### 3.1 Standardtechniken

Es gibt eine Vielzahl von Techniken und Technologien, um ein 3-dimensionales Objekt zu erschaffen. Die meisten Charaktere bestehen aus Polygonen und können durch unterschiedlichste Art und Weise erstellt werden. Viele dieser Techniken besitzen unterschiedliche Bezeichnungen.

Die im Folgenden vorgestellten Techniken sind für die Realisierung virtueller Charaktere jedoch zu ineffizient.

- **Primitive-Modeling** bezeichnet die Technik, 3D-Modelle aus parametrischen Objekten zu erstellen. Parametrische Objekte sind nicht aus Punkten und Polygonen aufgebaut, sondern werden mit Hilfe von Formeln beschrieben, wie zum Beispiel Kugeln, Würfel und Zylinder. Durch den Einsatz Boolescher Operationen entstehen so aus den „Primitives“ unterschiedliche 3D-Objekte. Allerdings können durch Primitive-Modeling nur einfache Gebilde erstellt werden. Komplexe Objekte können nur durch einen erheblichen Mehraufwand realisiert werden.
- **Spline-Modeling** beschreibt die Möglichkeit der Objekterstellung allein durch Splines. Splines sind Pfade, die durch Punkte und Tangenten bestimmt werden. Durch die Aneinanderreihung von Splines baut sich das Modell langsam auf. Detaillierte Objekte sind durch diese Technik kaum realisierbar. Die Handhabbarkeit und Steuerung der Splines nimmt mit der Komplexität der Objekte dramatisch ab.
- **Metaballs** beschreibt eine andere Herangehensweise, um eine Geometrie zu entwickeln. Ausgangspunkt sind hier die Metaobjekte – meist einfache Kugeln. Sind die Entfernungen zwischen diesen Unterobjekten gering genug, stülpen sich die Oberflächen gegenseitig aus und beide Objekte verschmelzen regelrecht. Geometrien bauen sich also aus dem Zusammenspiel seiner Unterobjekte auf. Mit Metaballs ist es sehr schnell möglich, organische Formen zu erstellen, welche mit anderen Techniken langsamer und vor allem schwerer zu erstellen wären. Allerdings sind Metaballs weniger dazu gedacht, scharfe Kanten zu erzeugen.

## 3.2 Boxmodeling

Bei der Technik des *Boxmodeling* wird ein einfacher Grundkörper als Ausgangslage verwendet. Meist wird hierfür ein einfacher Würfel – also eine Box – genommen. Dieser Körper wird anschließend durch verschiedene Operationen unterteilt, verformt und erweitert, bis schließlich das gewünschte Endergebnis entsteht. Diese Technik kann man mit der Arbeit eines Bildhauers vergleichen, der am Anfang einen Steinblock hat und aus dem er Stück für Stück sein Ergebnis herausarbeitet.

Die Verwendung der Grundeigenschaften seiner Ausgangsobjekte erweist sich hier als Vorteil dieser Methode. Für das Herausmodellieren eines Kopfes kann man als Ausgangsobjekt eine Kugel verwenden, die den gewünschten Umrissen schon angepasst werden kann. Eine grobe Form eines gewünschten Objektes ist sehr schnell vorhanden.

Vorteil dieser Methodik ist, dass man offensichtlich keine genaue Planung benötigen muss. Man kann sofort losmodellieren und erarbeitet sich Schritt für Schritt sein Ergebnis. Allerdings kann man Probleme bekommen, je detaillierter ein Objekt ausgearbeitet sein soll, was für ein perfektes und detailreiches Menschenmodell sehr von Nachteil ist. Um neue Details einzubringen, ist ein ständiges Umorganisieren der Punkte und Kanten notwendig. Auch wenn die Ergebnisse zufrieden stellend ausgearbeitet werden können, ist eine exakte Positionierung aller einzelnen Punkte und Polygone langwierig.

Aus diesem nachteiligen Grund wird auf das Boxmodellierung bei der Erstellung eines virtuellen Menschen verzichtet. Weitere Techniken sind im Folgenden auf ihre Handhabbarkeit zu untersuchen.

## 3.3 Polygon-Modeling

Die Technik des *Polygon-Modeling* ist auf den ersten Blick ähnlich dem Boxmodeling, wobei jedoch keine Ausgangsobjekte verwendet werden. Anders als beim *Boxmodeling* wird das 3D-Objekt hierbei Punkt für Punkt und Polygon für Polygon nacheinander aufgebaut.

Hat man eine genaue Planung seines zukünftigen Objektes vor sich, ist dieses Modellierungsmethode zu bevorzugen. Die Möglichkeit der genauen Platzierung der einzelnen Punkte und der einzelnen Polygone macht die Entstehung eines detaillierten Objektes sehr einfach. Ein Charakter kann mit dieser Technik leichter z.B. entlang seiner Muskelverläufe aufgebaut werden, während beim Boxmodeling eine strukturelle Reorganisation der Punkte und Polygone notwendig wäre. Das Polygon-Modeling ist effektiver bei der Anwendung.

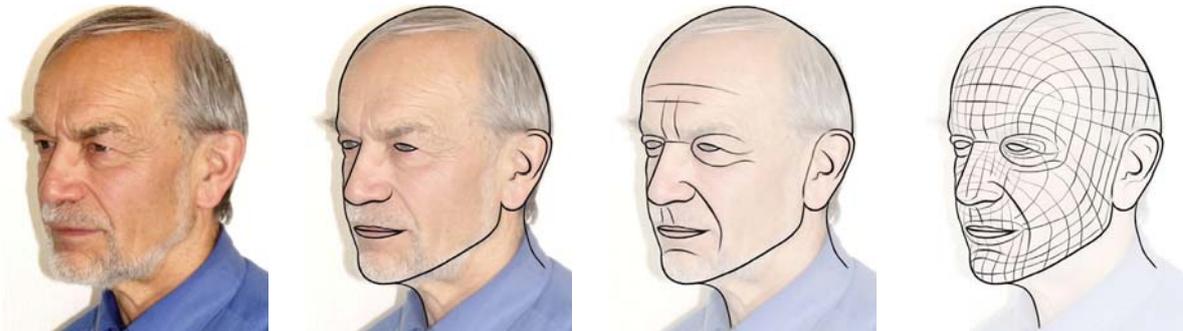
### 3.3.1 Modellierung eines virtuellen Charakters

Das Polygon-Modeling scheint für die Erstellung eines Charakters genau die richtige Technik zu sein. Ausgehend von den zuvor erstellten Vorlagen von Kopf, Körper und Händen wird das Mesh, also das Polygongerüst, Schritt für Schritt aufgebaut.

### 3.3.1.1 Topologie

Die *Topologie*, d.h. die Positionierung der Polygone, sollte auf keinen Fall wie schon in Kapitel 2 erwähnt planlos erfolgen. Je näher der Polygonverlauf an die tatsächlichen Muskelverläufe angelehnt wird, desto realistischer wird zum Schluss das Resultat sein [Olm]. Erstens bilden sie keine harten Kanten und verlaufen immer glatt, bilden also keine Falten an ungewöhnlichen Stellen, wo man sie nicht erwartet. Zum anderen werden solche harten Kanten vor allem nicht bei möglichen Animationen auftreten, da sie ja den Muskelverläufen nachempfunden werden und diese ansatzweise erscheinen lassen. Zu guter Letzt werden automatisch weniger Polygone verwendet um Details wie Falten zu kreieren, was wiederum der Rechenleistung und Rechenzeit zu Gute kommt.

Die Hauptkanten der Polygone sollten entsprechend den Falten der Vorlage positioniert werden. Diese Falten sind je nach Vorlage und Alter unterschiedlich stark ausgeprägt, aber bei jedem Menschen sehr ähnlich und an gleichen Positionen gelegen. Nachdem dieser erste Schritt das Grundgerüst liefert, werden in einem zweiten Schritt diese Hauptkanten mit Polygonen mit entsprechendem Verlauf aufgefüllt. Optimalerweise verlaufen die Polygonreihen kreisförmig und verbinden sich automatisch. Diese sogenannten Edge-Loops sollten sich besonders an Stellen wie um den Mund, die Augen, die Ohren etc. bilden. Edge-Loops haben den Vorteil, dass sie glatte Muskelverläufe am besten nachbilden.



**Abbildung 3-1: Topologieplanung:** Planung der Polygonverläufe auf Basis der Vorlage.

Solche Edge-Loops durch den ganzen Charakter durchzuführen ist allerdings äußerst schwierig. Kompromisse müssen an mehreren Stellen eingegangen werden, damit es nicht zu unnötigen Verdichtungen an einigen Stellen kommt. Somit wird es nicht ausbleiben, dass hier und da die weniger vorteilhaften Dreiecke oder gar Fünfecke auftauchen, welche zu unschönen Kanten führen können. Auch sind längliche Polygone zu vermeiden, solange sie nicht in Falten involviert sind.

### 3.3.1.2 Detailgrad des Modells

Zuerst sei zu beachten: Je mehr Punkte für ein Objekt gesetzt werden, desto detaillierter wird es und desto genauer gleicht es seiner Vorlage. Dies ist essentieller Schritt auf dem Weg, ein realistisches Abbild zu erschaffen. Allerdings bedeutet dieser hohe Detailgrad auch mehr Arbeit in der Erstellung und der Weiterverarbeitung. Bei der Texturierung oder auch bei der Animation werden all

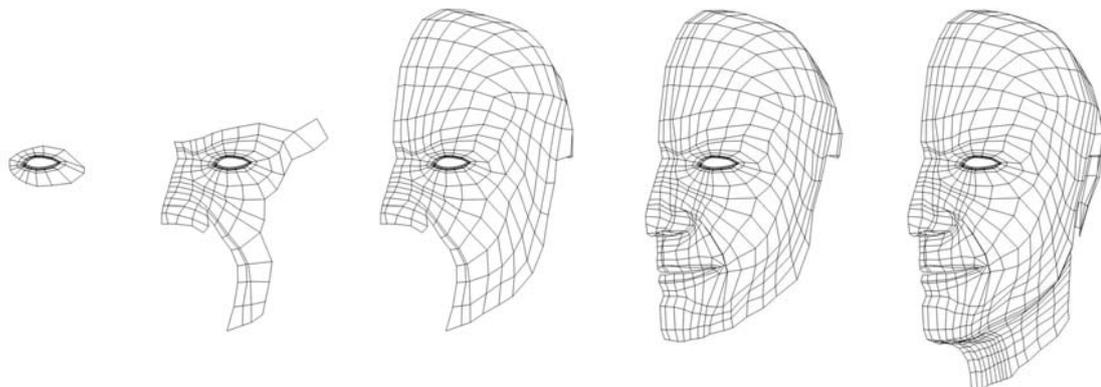
die erstellten Punkte noch einmal erfasst. Desweiteren bedeuten sehr hoch aufgelöste Modelle auch einen erhöhten Performanceverlust bei der Verarbeitung und schließlich beim Rendering.

Als Grundregel für den Detailgrad gilt: An Stellen mit wenig Details werden wenig Polygone verwendet und an detailreichen Stellen mehr. Zusätzlich sei schon hier angemerkt, dass mit dem Polygon-Modeling ein Grundgerüst erschaffen wird. Kleine Details, wie kleinere Falten auf der Stirn eines Menschen, werden nicht mit dem Polygonmodeling eingearbeitet. Nur die groben Details bis hin zu Faltenandeutungen werden ausmodelliert. Für kleinere Details sind andere Verfahren wirkungsvoller und effizienter.

### 3.3.1.3 Positionierung der Polygone

Die Außenlinien markanter Stellen bilden die erste Orientierung für das Setzen der ersten Punktelinien. Aus diesen Punktelinien werden dann die ersten Polygonreihen zusammengesetzt. So werden schnell die Punkte um die Augen erstellt und die ersten Polygonloops bilden das Auge nach. Durch mehrfaches Wechseln zwischen der Frontalansicht und der Seitenansicht werden die Punkte in die richtige Position gebracht. Die 45°-Ansicht dient immer wieder dazu, Fehler zu erkennen. Vor Allem sollte man immer wieder die 3D-Ansicht überprüfen, da man hier das fertige Resultat am besten erahnen kann.

Nach dem gleichen Schema werden weitere Polygonreihen ausgehend von dem Auge erstellt. Dabei wird versucht, eine gleiche Anzahl an Polygonen Reihe für Reihe aufzubauen, um die entstehenden Loops fortzuführen. Dies erstreckt sich von den Augen über die Wangen bis zur Nase. Bei der Modellierung des oberen Kopfes und des Hinterkopfes werden die Punkte weit von einander positioniert. Hier sind keine Details einzuarbeiten und eine hohe Anzahl von Polygonen wäre nicht sinnvoll.



**Abbildung 3-2: Polygonaufbau der Kopfhälfte.** Zu sehen sind Edge-Loops, z.B. um die Augen, und unvermeidbare Dreiecke, z.B. an der äußeren Seite der Stirn.

Weiter geht es mit dem Mund und dem Hals. Gerade an Stellen wie dem Mund oder der Nase kommt es öfters zu einer Aufspaltung der Polygone, da hier viele Polygone zusammentreffen. Die

Verwendung von Dreiecken ist die logische Konsequenz. Dies ist auch beim letzten Detail, dem Ohr, festzustellen. Aufgrund des Detailreichtums und der erhöhten Polygonansammlung werden an den Übergängen von vielen Polygonen zu wenigen Polygonen Dreiecke verwendet. Um keine Kanten zu erzeugen wird sehr sorgfältig darauf geachtet, dass sie möglichst planar zu den angrenzenden Polygonen platziert sind.

Auf ähnliche Weise wird der weitere Körper Schritt für Schritt aufgebaut. Die Kleidung wird mit modelliert und nicht durch komplexe Kleidungssimulationen berechnet. Da dieser Charakter nur für Stillbilder gedacht ist, ist eine einfache Polygonmodellierung der Kleidung die effektivere Variante. Die Augen sollten bei einem menschlichen Charakter immer viel Beachtung erhalten, denn sie tragen viel für den Gesamtausdruck bei.

»Sie [die Augen] sind extrem wichtig, um die jeweilige Stimmung des Charakters zu kommunizieren. Sie erwecken ihn erst richtig zum Leben.« [Andrew R. Jones<sup>10</sup>]

Das Augenobjekt an sich wurde aus zwei Polygonkugeln gebildet. Das Innere wurde an einer Stelle entsprechend der Pupille nachgebildet, während die zweite größere Polygonkugel den Glaskörper des Auges bildet. Weitere Bilder über Geometrie des gesamten Körpers sind im Anhang zu finden.

#### 3.3.1.4 Konturen

Meist wird bei organischen Objekten versucht, harte Kanten zu vermeiden. Braucht man allerdings einmal harte Kanten, so verwendet man einfach nebeneinander verlaufende Punktreihen. Durch Verschieben einer dieser Punktreihen entsteht sehr einfach eine harte Kontur.

Umgekehrt können durch Löschen mehrere Polygone, vor allem an Stellen mit erhöhter Polygonzahl, störende Schattierungen entfernt werden.

#### 3.3.1.5 Symmetrie

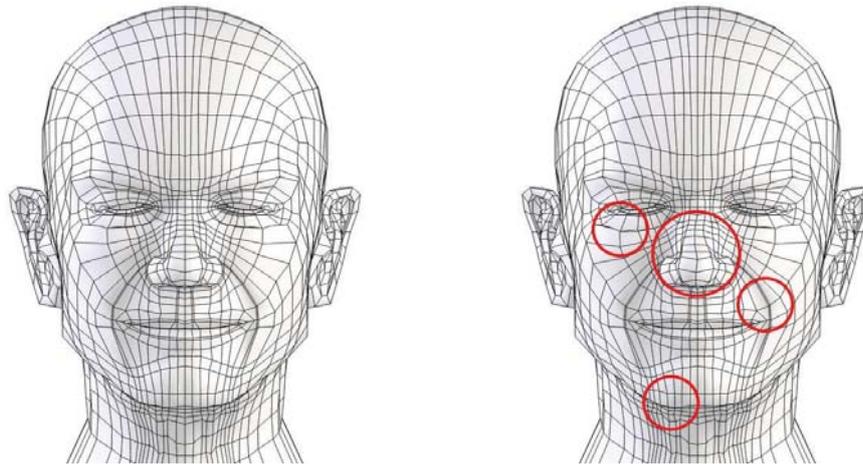
Der menschliche Körper besteht aus zwei nahezu identischen Körperhälften. Um nicht beide Körperhälften zu modellieren, wird ein Symmetrieobjekt verwendet, welches noch während des Modellierungsprozesses das Objekt spiegelt. Dadurch muss man nicht beide Gesichtshälften modellieren, sondern nur eine. Das Ergebnis kann aber auch sofort betrachtet werden, um eventuelle Fehler gleich zu erkennen.

Wie schon im Kapitel über die Planung angesprochen wurde, verfügt kein reales menschliches Gesicht über eine echte Symmetrie. Mehrere Studien zeigen zwar, dass je symmetrischer ein menschliches Gesicht ist, es umso attraktiver für uns erscheint [Tho]. Allerdings haben wir auch in unserem Leben gelernt, dass Menschen mehr oder minder asymmetrische Gesichter haben. Und je perfekter ein menschliches Gesicht ist, desto unglaubwürdiger und unrealistischer erscheint es uns.

---

<sup>10</sup> Andrew R. Jones war zuständiger Animationsregisseur beim Film "Final Fantasy: The Spirits Within" (2001, Square Films).

Nachdem das Gesicht fertig modelliert ist und unserer Erwartung genügt, wird die andere Kopfhälfte einfach gespiegelt und mit der anderen Hälfte zu einem Objekt verschmolzen. Anschließend wird an einigen Stellen eine Asymmetrie erzeugt, indem einige Punktwolken an einigen Stellen leicht und an anderen Stellen etwas mehr verschoben werden. Dies geschieht auch in Anlehnung an unsere Vorlage.



**Abbildung 3-3: Symmetrie bei Kopfdarstellungen.** Bei Nase, Kinn, Mundwinkel und im Bereich unter den Augen wurde eine gezielte Asymmetrie erzeugt.

### 3.4 Polygonunteilungen

Ein 3D-Objekt besteht immer aus flachen Polygonen, welche letztendlich beim Rendern in Dreiecke aufgespalten werden. Jedes dieser Polygone besitzt einen Normalenvektor, der sich aus der Positionierung der Eckpunkte des Polygons berechnet. Normalenvektoren stehen dabei immer senkrecht auf der Polygonenfläche. Aus den Winkeln, die sich zwischen Normalenvektor des Polygons, der Lichtquelle und der Kamera bilden, berechnet sich schließlich die Helligkeit des späteren Bildpunktes<sup>11</sup>.

Zwei benachbarte Polygone, die einen unterschiedlichen Winkel besitzen, haben letztendlich auch unterschiedliche Helligkeiten und diese Helligkeitsunterschiede fallen auf. Nachteilig ist, dass man unzählige Polygone erstellen müsste, damit diese Helligkeitsunterschiede so gering wie möglich ausfallen, um eine glatte Rundung nachzubilden. Dies dauert nicht nur sehr lange, dies ist zusätzlich kompliziert und weitere Veränderungen sind schwierig, da die gesamte Geometrie selbst bei kleinen Änderungen komplett angepasst werden muss. Letztendlich hat man auch eine geringere Rechenperformance.

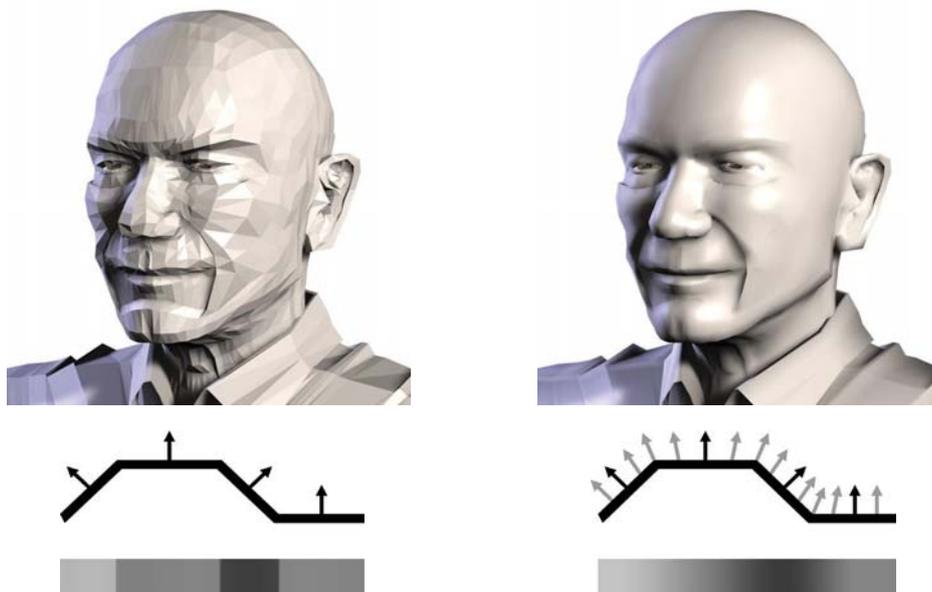
Eine erste Abhilfe bietet hier das *Phong-Shading*. Hierbei werden die Helligkeiten der Bildpunkte

---

<sup>11</sup> vgl. Kapitel 6.1.1 Raytracing

zwischen den Normalvektoren der beteiligten Polygone interpoliert. Die Helligkeit des einen Normalenvektors geht allmählich in die Helligkeit seines Nachbarn über. Dadurch sehen die Übergänge nun glatt aus, ohne dass man hunderte Polygone neu erzeugen müsste.

*Phong-Shading* ist eine leistungsstarke Eigenschaft, die man bei jeder Art von runden Übergängen aktiviert haben sollte, eine ideale Eigenschaft für organische Formen. Allerdings bleibt die Ausgangsgeometrie unangetastet. Dies hat den Vorteil, dass es schnell berechnet werden kann, allerdings sieht man an den Objektumrissen meist deutlich, dass die Geometrie unverändert bleibt. Nur die Helligkeiten, das Shading, werden hierbei angepasst.



**Abbildung 3-4: Phong-Shading:** Das Phong-Shading interpoliert Helligkeiten zwischen Polygonen, indem neue Helligkeitsabstufungen zwischen den Normalvektoren gebildet werden.

Eine weit mächtigere Möglichkeit der Rundungen bilden die Polygonunterteilungen, die die Geometrie nachträglich verändern. Meist wird hierbei ein 3D-Objekt erst einfach mittels Boxmodeling oder Polygon-Modeling aufgebaut und dann dieses einfache Polygonobjekt zum Beispiel mittels *NURBS (Non-Uniform Rational B-Splines)* unterteilt. Diese Unterteilung geschieht meist parametrisch und verfeinert das Modell, welches zum Schluss glatter und organischer wirkt. Diese Technik ist also ideal, um organische Charaktere einfach und schnell zu erstellen.

### 3.4.1 NURBS

Splines sind Pfade, die mittels Punkte und/oder Tangenten bestimmt werden. Unterschiedliche Spline-Arten werden durch unterschiedliche Arten der Interpolation zwischen den Punkten bestimmt. Während der Pfad bei den Beziersplines durch die Kontrollpunkte verläuft, wird der Pfad bei den B-Splines nicht nur durch die Stützpunkte beeinflusst und geht nicht durch diese hindurch. Weiter

auseinander liegende Punkte haben bei den B-Splines einen geringeren Einfluss auf den Kurvenverlauf als enger zusammenliegende Punkte.

Diese B-Splines bilden im 3-dimensionalen Raum die Grundlage der NURBS. Ein 3D-Geometrie steuert hierbei die NURBS-Fläche mit all seinen Punkten. Diese NURBS-Fläche wird mittels dieser Geometrie interpoliert und aufgebaut. In der Industrie sind NURBS-Flächen weit verbreitet. Vorteil der NURBS liegt Dank der Genauigkeit der Splines in der Skalierbarkeit. Dadurch ist eine hohe Genauigkeit bei der Modellierung möglich. Ein gutes Einsatzgebiet sind demzufolge Bereiche, wo ein hoher Detailgrad notwendig ist, wie bei technischen Geräten.

Beachtet werden sollte, dass NURBS letztendlich kleiner ausfallen, als die Polygonvorlage. Grund ist hierfür, dass die NURBS-Interpolation mittels Tangenten berechnet wird. Bei einer großen Anzahl an Polygonen wird dies nicht weiter ins Gewicht fallen, aber bei kleinen Objekten oder kleinen Details wird dies schon sichtbar.

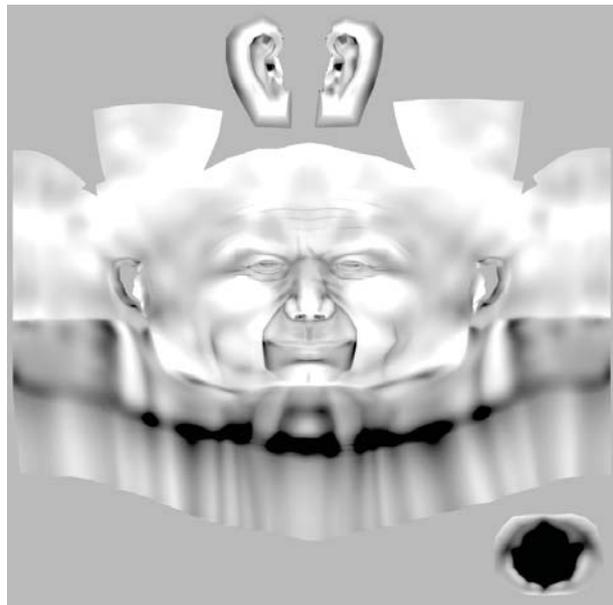
### 3.4.2 Subdivision Surfaces

Das Ausgangsgitter wird je nach angegebener Stufe mehrfach unterteilt. Die neuen Unterteilungspunkte werden je nach Art der Implementation unterschiedlich interpoliert und bilden letztendlich eine geglättete Version des Ausgangsnetzes. Die Subdivision Surfaces bringen ähnliche Ergebnisse wie NURBS-Unterteilungen.

Meist wird das *Subdivision Surfaces* zusammen mit dem *Boxmodeling* angewendet. Hierbei wird ein Grundkörper mittels Subdivision Surfaces unterteilt und anschließend weiter strukturiert. Werden an manchen Stellen weitere Details benötigt, so wird das Subdivision Surfaces nur an diesen Stellen angewendet, in ein Polygonobjekt umgewandelt und weiter modelliert. Diese teilweise Anwendung der Subdivisions hat gegenüber den NURBS den Vorteil, dass sie auch auf einzelne Teile der Geometrie angewendet werden können. Bei den NURBS wird hingegen immer die gesamte Geometrie mit einbezogen.

### 3.4.3 Displacement

Während die vorher beschriebenen Techniken der NURBS und des Subdivision Surfaces sehr gut dazu eingesetzt werden können, um Polygonobjekte zu unterteilen und gegebenenfalls die unterteilten Objekte weiter zu verfeinern, bieten *Displacement-Maps* eine weit mächtigere Möglichkeit.



**Abb. 3-5: Displacementmap:** Hiermit wird die Geometrie weiter gesteuert.

Hierbei kann man durch die Verwendung einer Textur<sup>12</sup> zusätzlich die Geometrie beeinflussen. Helligkeitsunterschiede in der Textur steuern die Verschiebung der Polygone entlang ihren Normalen. Dunkle Stellen in der Textur verschieben die Geometrie nach innen, während helle Stellen die Geometrie nach außen hin verschieben. Reines Schwarz und reines Weiß stellen dabei die beiden Extrema dar.

#### 3.4.3.1 Subpolygone-Displacement

Der Vorteil der Displacement-Maps liegt unweigerlich in der Beeinflussung der Geometrie. Allerdings kann nur die vorhandene Geometrie beeinflusst werden. Durch das Zusammenspiel aus Displacement-Maps und weiteren Unterteilungen ergibt dies das *Subpolygon-Displacement (SPD)*. Hierdurch ist es nun möglich, noch weitere Details einzuarbeiten, ohne dass die Hauptgeometrie beeinflusst wird. Die Geometrieunterteilung wird erst zum Ende beim Rendering berechnet, wodurch allerdings auch mit weitaus längeren Renderingzeiten zu rechnen ist.

### 3.5 3D-Scanner

Eine der einfachsten Varianten, ein 3D-Objekt von einer realen Vorlage zu digitalisieren, ist die Verwendung von 3D-Scannern. Dies ist allerdings auch eine kostenintensive Variante, da hier speziell dafür entwickelte Scanner in den verschiedensten Größen zum Einsatz kommen.

Bei den 3D-Scannern gibt es zwei verschiedene Ausführungen:

- Bei den **Digitizern** werden die zu digitalisierenden Objekte mit Markierungspunkten versehen und anschließend diese einzelnen Punkte in 3D-Daten umgewandelt.
- Bei den **Fullbody-Scannern** wird hingegen die komplette Oberfläche eines Objektes eingelesen. Hierbei tastet zumeist ein Laser die Oberfläche Stück für Stück ab. Dabei kommt es allerdings zu einer massiven Ansammlung von Daten und die so entstandenen 3D-Objekte müssen meist nachbearbeitet werden, d.h. die Polygonanzahl muss auf ein vernünftiges Maß reduziert werden, damit ein weiteres Arbeiten möglich ist.

Einige Scanner sind mittlerweile in der Lage, zusätzlich zu der eigentlichen Objektgeometrie die Oberfläche mit zu scannen. Das bedeutet, dass die Scanner auch die Texturen der zu scannenden Objekte mit erfassen.

3D-Scanner sind ein sehr gutes Hilfsmittel, um ein reales Objekt in ein digitales 3D-Objekt umzuwandeln. Der Vorteil ist offensichtlich die Einfachheit und die Schnelligkeit des Systems. Nach

---

<sup>12</sup> vgl. Kapitel 4 Texturen

kurzer Zeit hat man seine Daten vorliegen. Wie bereits angemerkt, weisen diese Rohdaten meist eine große unkontrollierbare Datenmenge auf, was das weitere Arbeiten unmöglich macht. Diese müssen gefiltert, bearbeitet und optimiert werden. Allerdings gibt es auch wiederum gute Algorithmen, die je nach angegebenen Parametern diese Punktwolken auf die gewünschte Anzahl an Punkten und Polygonen reduzieren. Eine intensive Nacharbeit ist meist unumgänglich, damit die Punkte die gewünschten Positionen haben.

### 3.6 Anwendungen in der Industrie

Für die Entwicklung der Filmtrilogie *Herr der Ringe* (2001-2003, New Line Cinema) war den Produzenten vornherein klar, dass die Umsetzung von J.R.R. Tolkiens Büchern nur mit den neuesten Techniken möglich ist, um dieser Fantasiewelt ein glaubwürdiges Leben einzuhauchen.

»Bei den digitalen Effekten ließ ich mich von der gleichen Philosophie leiten, wie bei allen anderen Aspekten des Films. Ich wollte, dass die Monster echt wirken - vom Dreck und den Fingernägeln eines Bergtrols zu den blutunterlaufenen, hervorquellenden Augen von Gollum« [Peter Jacksons<sup>13</sup>]

Für jede der Hauptfiguren wurde ein digitales Double angefertigt. Als Ausgangsbasis dienten dabei 3D-Scans der Schauspieler. Für alle virtuellen Figuren und Kreaturen wurden zunächst Tonfiguren angefertigt und anschließend mit einem Handscanner digitalisiert. Anschließend wurden NURBS-Oberflächen darüber modelliert.

Die Modellierung der wichtigsten Personen und Kreaturen erfolgte über NURBS-Patches (Patch-Modelling), was vor den Subdivision Surfaces die modernste Technik war. Da die Kreaturen auch realistisch bei der Bewegung aussehen sollten, wurden alle Knochen und Muskeln mit modelliert, was den Zeit- und Arbeitsaufwand enorm vergrößerte. Einerseits konnte man dadurch die Figuren besser über deren Knochen und Muskeln steuern, andererseits wollte man dadurch den typisch steifen Look bei digitalen Figuren vermeiden. Allerdings hatte die verwendete Software Probleme, die vielen NURBS-Flächen dynamisch zusammenzuhalten. Beim Rendering entstanden immer wieder Löcher in der Geometrie, die im Compositing / in der Nachbearbeitung mühselig entfernt werden mussten. Aus diesem Grund wurden die meisten Kreaturen aus Teil 2 und 3 der Trilogie auf Subdivision-Surfaces umgestellt.

»Ich glaube, dass der Gollum eines der fortgeschrittensten Digitalwesen ist, die man bisher gesehen hat.« [Richard Tayler<sup>14</sup>]

Die Hauptakteure in dem Film *Spider-Man* (2002, Marvel Enterprises / Columbia Pictures) wurden

---

<sup>13</sup> Peter Jacksons war Regisseur der *Herr der Ringe* - Trilogie (2001-2003, New Line Cinema).

<sup>14</sup> Richard Tayler<sup>14</sup> ist Chef bei Weta Digital, der Special Effects-Firma zuständig für *Herr der Ringe*.

komplett nachmodelliert. Die digitalen Versionen der Schauspieler wurden dann in extrem gefährlichen und in für den Menschen unmöglich zu bewältigenden Szenen eingesetzt. Man denke dabei z.B. an die eleganten Sprünge über die Dächer von New York, die mit realen Personen nicht machbar gewesen wären.

»Unser Ziel war ein einwandfreies Character Modeling. Spider-Man und der grüne Kobold sind die Stars der Show, also mussten die CG-Versionen unbedingt realistisch aussehen.« [Peter Nofz<sup>15</sup>]

Spider-Man selbst wurde mit NURBS modelliert. Allerdings gab es bei der Animation Probleme mit Beschädigungen der Oberfläche, da die NURBS später noch durch Subdivision Surfaces ausgetauscht wurden. Der grüne Kobold wurde mit Polygonen und NURBS erstellt [Tet].

Displacement-Maps werden in der Filmindustrie seit vielen Jahren zur Geometrierfeinerung eingesetzt. In Pixars Kurzfilm *For the Birds* von 1997 werden Displacement Maps dazu genutzt, um die Schnäbel der verschiedenen Vögel zu individualisieren.

### 3.7 Fazit

Während die einfachen Standardtechniken viel zu ineffizient sind um die komplexe Geometrie eines menschlichen Charakters aufzubauen bilden das Boxmodeling und das Polygonmodeling gute Techniken hierfür. Hat man eine Vorlage, gilt das Polygonmodeling als bevorzugte Modellierungstechnik, um eine Grundgeometrie zu erstellen, die noch gut handhabbar ist, um sie für die anschließende Texturierung und Animation weiter zu verarbeiten. Auch Edge-Loops sind beim Polygonmodeling am einfachsten einzuarbeiten. Auch wenn manche Character-Artist schneller mit dem Boxmodeling arbeiten können ist oft ein Umorganisieren der Geometrie notwendig um das gewünschte Endergebnis zu erreichen. Eine Verwendung des Phong-Shading ist bei organischem Modellieren immer anzuraten, da es eine optische Glättung bewirkt.

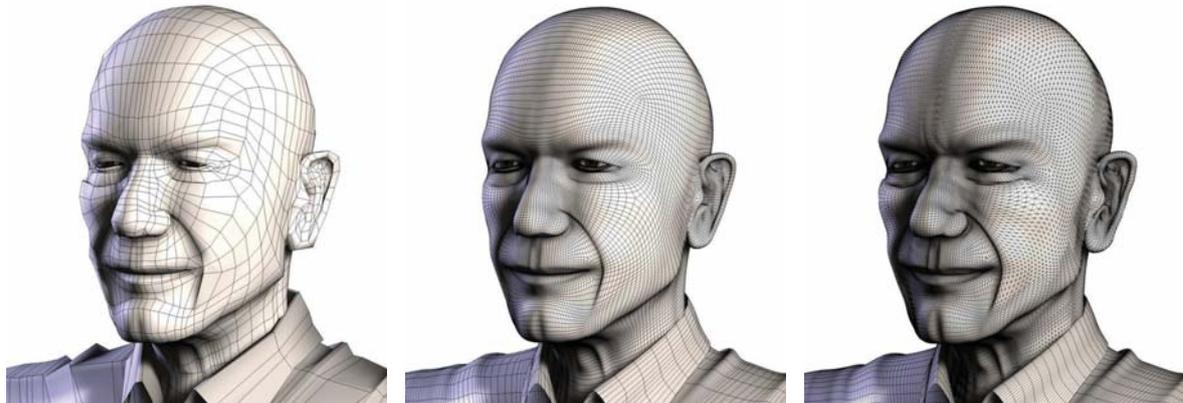
Die zusätzliche Benutzung einer Polygonunterteilung wie den NURBS oder Subdivision Surfaces rundet die erstellte Geometrie ab und lässt sie glatter und organisch erscheinen. Dies ist eine wichtige Eigenschaft bei der Erstellung von menschlichen Charakteren. Durch diese Polygonunterteilungen wird die Hauptgeometrie je nach Art kaum bis gar nicht verändert, wodurch sich das Objekt auch für die Texturierung und Animation einfach weiterverarbeiten lässt. Allerdings wird trotzdem die Polygonanzahl erhöht, was sich in einem Performanceverlust, spätestens beim Rendering, bemerkbar macht. Zusätzliche Details können mit Hilfe von Displacement-Maps eingearbeitet werden, allerdings muss hierfür die Geometrie entsprechend hoch sein, damit die vorhandenen Punkte beeinflusst werden können.

Am effektivsten gilt bei den Polygonunterteilungen das Subpolygon-Displacement, da sich hier am

---

<sup>15</sup> Peter Nofz war zuständiger CG Supervisor bei *Spider-Man* (2002, Marvel Enterprises / Columbia Pictures).

besten weitere Details einarbeiten und kontrollieren lassen. Die Geometrie bleibt noch handhabbar und die Details können dennoch zu einem sehr hohen Grad eingearbeitet werden – ein unübertrefflicher Vorteil bei der Charaktererstellung.



**Abbildung 3-6: Geometrie:** Ausgangsgeometrie, NURBS-Unterteilung und Subpolygon-Displacement unter Verwendung einer Textur zur Steuerung.

Abbildung 3-6 zeigt die unterschiedlichen Auswirkungen der Polygonunterteilungen bei dem Kopf des Charakters. Ganz links ist die Ausgangsgeometrie mit 2.308 Polygonen zu sehen. Offensichtlich sind dies noch zu wenige Polygone, um eine glatte Oberfläche zu imitieren. Ein hinzugefügtes Phong-Shading hilft an dieser Stelle nur unzureichend. In der Mitte weist eine dreifache NURBS-Unterteilung bereits 149.348 Polygone auf. Die Oberfläche ist weitaus glatter und organischer. Ganz rechts ist ein Subpolygon-Displacement mit vierfacher Unterteilung und 590.848 Polygonen zu sehen. Einzelne Details wie Falten oder Muskelverläufe treten nun gut in den Vordergrund.

Grundsätzlich ist beim Modeling immer zu beachten: Genug Polygone um alle Details wiederzugeben, aber so wenig Polygone wie möglich, um leichter damit arbeiten zu können. Je weniger Polygone ein Objekt besitzt, desto einfacher kann man die Texturen anpassen und desto weniger Polygone müssen bei einer eventuellen Animation bearbeitet werden.

## 4 Texturierung

Objekte sind in der Realität selten einfarbig und ohne Unebenheiten. Bei den Menschen ist dies genauso und Hautporen, Venen, Farbabstufungen und weitere Details gehören zum allgemeinen Erscheinungsbild dazu. Beim Texturieren bekommen die erstellten noch farblosen Objekte diese besondere Charakteristik. Die leeren und nackten Objekte bekommen eine Hülle, die Textur, übergestülpt. Diesen Vorgang kann man sich wie beim Tapezieren vorstellen. Dazu werden für jedes Objekt ein oder mehrere Materialien erstellt, in dessen Kanäle Bilder liegen und dessen spezielle Eigenschaft steuern.

Während sich in der Computergrafik Plastik und Metall sehr leicht darstellen lassen, handelt es sich bei der Umsetzung von Haut mit ihren verschiedenen Transparenzen und Unregelmäßigkeiten allerdings um eine ganz andere Herausforderung. [Hyp]

Eine Textur ist hierbei ein wichtiges Mittel, um 3D-Objekte realistisch wirken zu lassen. Erst hierdurch bekommt es seine eigene Oberfläche, die es unverwechselbar macht und zur Illusion des Realismus beiträgt.

### 4.1 Texturmapping

Es gibt verschieden Arten, eine Textur auf ein Objekt zu wickeln. Einfache Mappingmethoden orientieren sich an der Form von den Standardobjekten und stülpen eine Textur auf ähnliche Weise über. Das UV-Mapping hingegen bestimmt im Gegensatz dazu exakt polygongenau, wo die Textur aufgespannt werden soll.

#### 4.1.1 Standardmapping

Die einfachsten Mappingverfahren sind die *planaren Mappingverfahren* wie das Flächenmapping. Sie projizieren die Textur wie ein normaler Projektor einfach auf ein Objekt auf. Ein 3D-Objekt benötigt dadurch mehrere planare Projektionen, um überall eine Texturierung zu erhalten.

Andere Projektionsarten gehen weiter. Das *Quadermapping* definiert bestimmte Bereiche der Textur als oben, unten, links, rechts, vorn und hinten. Diese Bereiche werden dann von allen sechs Richtungen auf das Objekt projiziert, als wenn dieses ein Quader wäre. Beim *Zylindermapping* wird die Textur ausgehend von einem Zylinder, oder beim *Kugelmapping* ausgehend von einer Kugel auf das Objekt projiziert. Problematisch sind hier allerdings die jeweiligen Pole, an denen die Textur auf einen Punkt zusammenschrumpft. Solche Stellen müssen möglichst versteckt werden.

Für einfache Objekte reichen solche Projektionsarten aus, vor allem wenn nicht sehr viele Details auf der Oberfläche zu sehen sind. Kleinere Details kann man auch separat mit einzelnen kleinen Texturen, welche mit einem Alphakanal freigestellt sind, übereinander stapeln. Letztendlich kommen allerdings bei komplexen Objekten irgendwo immer Verzerrungen zum Vorschein, weshalb die

einfachen Projektionsarten nicht für ein realistisches Texturmapping eines Menschen in Frage kommen können. Problematisch sind vor allem Stellen, die in der Projektionslinie übereinander liegen, wie beispielsweise am Kopf beim Ohr. Hier kommen die einfachen Mappingverfahren schnell an ihre Grenzen.



**Abbildung 4-1: Standardprojektionsarten:** Angewendet auf die Kopfgeometrie: Flächen-, Quader-, Zylinder- und Kugelmapping bringen jeweils Verzerrungen.

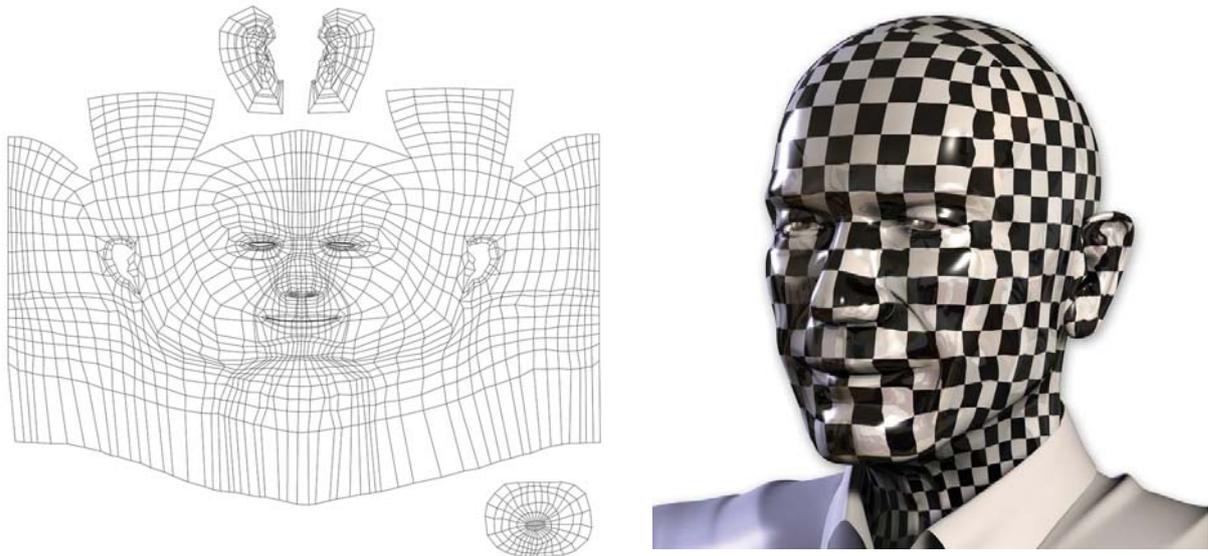
#### 4.1.2 UV-Mapping

Die einzige Möglichkeit, eine Textur perfekt auf ein komplexes Objekt aufzuwickeln, ist die Verwendung des *UV-Mappingverfahrens*. U und V entsprechen den Koordinaten einer Textur und werden den einzelnen Punkten des Objektes zugewiesen. So wie X, Y und Z den Punkt im Raum beschreiben, stellen U und V die Koordinaten der Textur da. Hiermit ist es nun möglich Texturen perfekt auf ein beliebiges Polygon im Raum zuzuweisen, ohne dass es zu Verzerrungen kommt. Ob ein UV-Mapping gut gewählt ist, lässt sich mit einer Karotextur überprüfen. Hierbei sieht man schnell, ob Verzerrungen vorhanden sind oder ob die Texturproportionen sich verändern.

Um UV-Koordinaten für den Kopf zu erstellen, der zuvor modelliert wurde, verwendet man für einzelne Polyongruppen mehrere planare Projektionen und führt die Polyongruppen schließlich wieder zusammen. Letztendlich ist es so, als wenn man die Haut abziehen würde und auf einer planaren Fläche ausbreiten würde. Für das Gesicht verwendet man demnach eine Projektion von vorn für die Front des Gesichtes, eine 45°-Projektion für leicht seitliche Gesichtspartien, eine weitere 90°-Projektion für die Seite und eine Projektion von hinten für den Hinterkopf.

Die Nähte der Polygone sollten so gewählt werden, dass sie möglichst an wenig sichtbaren Stellen

erfolgen, da die Übergänge sehr schwer weg zu retuschieren sind. Desweiteren sollten so wenige Nähte wie möglich verwendet werden, damit kaum Verzerrungen entstehen. Die Nähte werden dementsprechend so gewählt, dass sie letztendlich auf der Rückseite oder unter den Haaren liegen. Weiterhin ist darauf zu achten, dass keine Überlappungen entstehen und die UV-Koordinaten ähnliche Proportionen haben wie die Polygonproportionen.



**Abbildung 4-2: UV-Map und entsprechende UV-Projektion:** Die Karo-Textur ist nahezu perfekt auf die Geometrie gewickelt. Unsaubere Nahtstellen sind versteckt oder werden später von den Haaren bedeckt.

Sinnvoll ist es meist auch detaillierte Objekte aus seiner normalen Umgebung herauszunehmen und separat zu platzieren oder gleich eine eigene UV-Map zu verwenden. Bei dem Kopf wurden so zum Beispiel die Ohren aus seiner normalen Platzierung herausgelöst und weiter oben positioniert. Ansonsten wäre der Platz viel zu gering gewesen, um Details in die Textur zu integrieren und die Proportionen beizubehalten. Auf der Textur ist links unten der Mundinnenraum zu erkennen, bei dem entsprechend verfahren wurde.

Es ist wichtig, die UV-Koordinaten so sauber wie möglich zu setzen, denn nur dadurch lassen sich Verzerrungen und andere Artefakte vermeiden.

## 4.2 Texturauflösung

Texturen werden benutzt, um ein computergeneriertes Objekt realistisch aussehen zu lassen. Wenn man allerdings eine zu niedrig aufgelöste Textur verwendet und man später die Pixelkanten erkennt, bewirkt dies eher das Gegenteil.

Wie hoch muss also die Auflösung einer Textur sein? Generell kann man diese Frage nicht einfach beantworten. Ein Objekt, welches nur halb zu sehen ist, braucht eine kleinere Textur, als ein Objekt,

welches formatfüllend zu sehen ist. Weiterhin gibt es Objekte, bei denen nur ein kleiner Teil überhaupt zu sehen ist. Bei zum Beispiel einem Kopf sieht man immer nur eine Seite, während die Textur des Kopfes ringsherum geht. Hier sind höher auflösende Texturen notwendig.

Generell gilt: Je höher die Texturauflösung, desto mehr Details können aufgenommen werden. Allerdings gilt auch hier wiederum: Hohe Texturen werden in den Speicher eingeladen und verbrauchen Arbeitsspeicher. Details, die zum Schluss niemand sieht, verbrauchen nur unnötig Speicher und bremsen eventuell die gesamte Szene durch Einbußen an Arbeitsgeschwindigkeit aus.

Letztendlich sollte die Textur eines Objektes größer in der Auflösung sein, als sie zum Schluss beim Rendering gezeigt wird oder wenigstens im Verhältnis 1:1 stehen. Wird ein Film in Pal (720x576 Pixel) aufgenommen und ein Kopf eines Charakters wird formatfüllend gezeigt, sollte diese Textur an dieser Stelle mindestens 576 Pixel an Informationen in der Höhe aufweisen. Anhand der verwendeten Texturprojektion sollte man bei diesem Beispiel eine Textur von mindestens 1024x1024 Pixeln verwenden. [Byl]

Die meisten Computer und die meisten 3D-Programme arbeiten am effektivsten, wenn die Auflösung der Texturen Dimensionen von Zweierpotenzen hat. Gängige Dimensionen sind 512x512 Pixel, 1024x1024 Pixel oder 2048x2048 Pixel. Man beachte hierbei schon, dass eine einzelne Textur in der Größe von 4K schon 64 MByte an Arbeitsspeicher verbraucht. Bei mehreren Texturkanälen kann einem schon mit wenigen Objekten schnell der Arbeitsspeicher ausgehen. [Bir]

Zusammenfassend kann man sagen, dass Texturen an die Gegebenheiten angepasst werden müssen. Da hier ein realistischer Charakter entstehen soll, wird die Textur für das Gesicht meines Charakters zum Beispiel eine Auflösung von 4096x4096 Pixel haben. Nur so können auch alle Details ohne Probleme eingearbeitet werden, die notwendig sind und eine Nahaufnahme ist möglich.

### 4.3 Texturkanäle

Um die Oberfläche eines Objektes mittels einer Textur zu beeinflussen stehen mehrere Texturkanäle zur Verfügung. Ein jeder dieser Kanäle führt zu einem anderen Effekt beim letztendlichen Bild.

Um eine Hauttextur eines Menschen realistisch zu erstellen, sollte man folgende Texturkanäle benutzen:

- Farbe (Color)
- Diffusion (kann auch im Farbkanal integriert werden)
- Glanz (Specular)
- Spiegelung (Reflection)
- Relief (Bump)
- Displacement

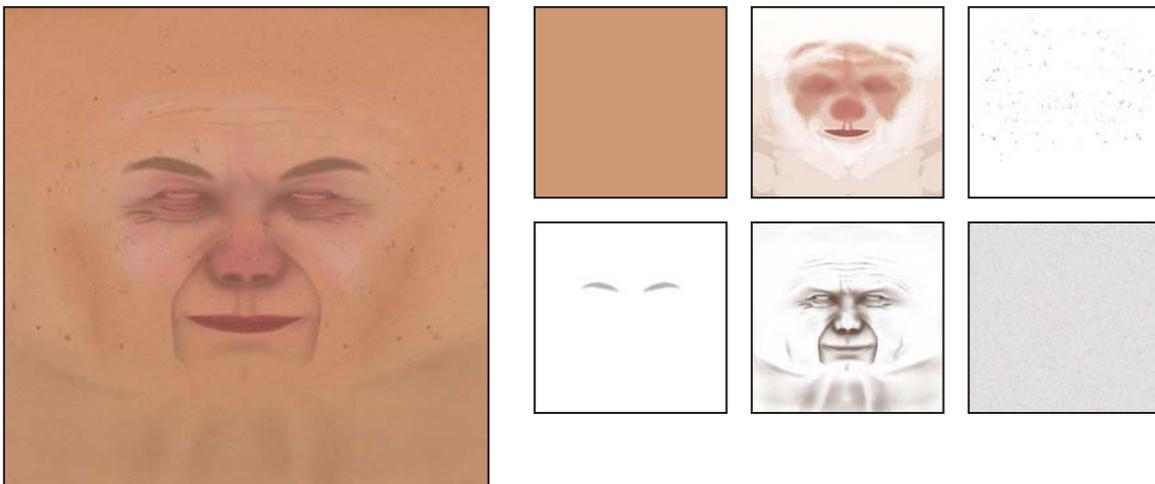
### 4.3.1 Farbkanal

Der Farbkanal (*Color*) ist der eigentliche Kanal einer Textur. Hier wird entschieden, welche Farbe oder welches Bild die spätere Oberfläche des Objektes bekommt.

Das Farb-Mapping ersetzt die Oberfläche innerhalb der diffusen Beleuchtung, aber hebt nicht die Beleuchtung und Schattierung des Renderers auf. Glanzpunkte, Schatten oder Beleuchtungsvariationen sollten vermieden werden, da der Renderer dies automatisch über die Lichtquelle erzeugt [Bir]. Hierzu gibt es allerdings wiederum Ausnahmen, wodurch man Renderzeit sparen kann. Ein integrierter Schatten im Farbkanal muss so später nicht erst berechnet werden.

Der Farblayer ist das wichtigste Stilmittel, um Haut nachzuahmen. Das was wir als Hautfarbe wahrnehmen ist eigentlich das Ergebnis aus dem Zusammenspiel mehrerer Schichten aus Blut, Poren, Harrfollikel, Schmutz, Venen, Blutgefäße und Geburtsmale. [Wad]

Die UV-Map und Referenzfotos geben wichtige Orientierungspunkte, um die Farbtexturen anzufertigen. Den ersten Layer bildet ein reiner Farblayer, der aus der Grundfarbe #D39975 besteht. Leichte Farbabstufungen an den Augen, Nase, Wangen und Mund werden hinzugefügt. Weitere Details, wie Leber- und Altersflecke und eine Grundierung der Augenbrauen werden hinzugefügt, um die Haut realistisch wirken zu lassen. Zuletzt wird ein einfaches Noise<sup>16</sup> mit geringer Intensität verwendet. Die Haut ist nie vollkommen glatt und perfekt. Durch diesen Noise-Layer wird der Haut ein unregelmäßiger Look gegeben.



**Abbildung 4-3: Farbtextur:** Die Colormap wird gebildet aus Grundfarbe, Farbabstufungen, Hautflecken, Haargrundierung, Ambient Occlusion und einem Noise.

Ein Verfahren, um das globale Licht zu simulieren, bietet Ambient Occlusion<sup>17</sup>. Dieser Effekt wird ausgegeben um mehr Kontrolle darüber zu haben und in die Farbtextur integriert. Zum Schluss werden all diese Layer übereinandergestapelt und ergeben die finale Farbtextur.

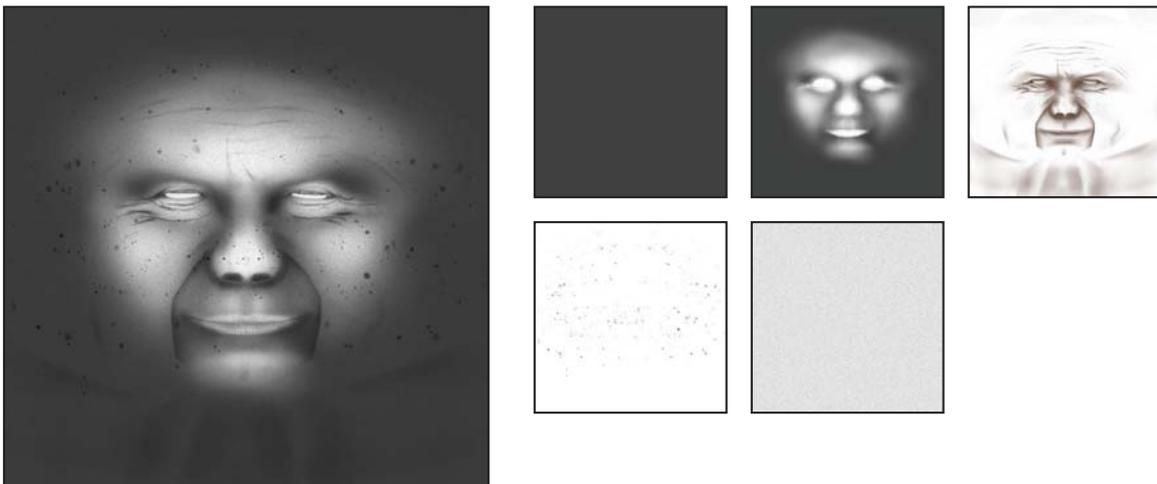
<sup>16</sup> Unter Noise wird ein Bild verstanden, welches nur aus Rauschen besteht.

<sup>17</sup> vgl. Kapitel 6.3.5 Ambient Occlusion

### 4.3.2 Glanz-Kanal

Der Glanz-Kanal (*Specular*) bestimmt, wie bestimmte Stellen eines Objektes die Lichtquellen reflektieren und so Glanzlichter auf der Oberfläche bilden. Bei Verwendung einer Textur bestimmen Grauwertangaben dann die Intensität. Reines Schwarz in der Textur reflektiert kein Licht, während ein reines Weiß eine Lichtquelle vollständig auf der Objektoberfläche spiegelt. Glanzlichter sagen viel über die Oberflächenbeschaffenheit und Härte eines Materials aus [Asa]. Möchte man also die Haut eines Menschen nachahmen, muss auch die Oberfläche eines virtuellen Charakters weich, dehnbar und dennoch glänzend wirken. Und genau diese Illusion kann man mit dem Setzen der Glanzpunkte perfekt erzeugen.

Die Stärke der Lichtglanzpunkte einer Haut variiert stark, abhängig davon, wo sie im Gesicht liegen. Die Menge an Öl und Feuchtigkeit einer Haut erhöhen die Intensität. Bereiche wie Nase, Stirn, Wangen, Kinn und Lippen gehören zu den Stellen, die sehr glänzend sind [Wad]. Als Orientierung für die Erstellung der Glanztextur sind auch hier wieder das Wireframe als UV-Map und die Fotovorlage sehr hilfreich um herauszufinden, welche Stellen wo und wie stark glänzen.



**Abbildung 4-4: Glanztextur:** Die Specularmap wird gebildet aus Grundfarbe, Glanzpunkte, Ambient Occlusion, Hautunreinheiten und einem Noise.

Grundlage für die Textur bildet hier ein dunkles Grau. Hierdurch wird das ganze Gesicht leicht glänzen. Als nächste Ebene werden die glänzenden Stellen hervorgehoben. Daraufhin findet wieder eine Ambient-Occlusion-Ebene Verwendung. Wo wenig Licht ist, kann auch wenig Licht gespiegelt werden. Nichtglänzende Details, wie Hautunreinheiten, werden integriert. Letztendlich wird auch hier wieder eine Noise-Ebene als Abschluss verwendet. Schließlich werden alle Ebenen zur finalen Glanztextur zusammengefasst.

### 4.3.3 Spiegel-Kanal

Der Spiegel-Kanal (*Reflection*) geht noch einen Schritt weiter als der Glanz-Kanal. Während beim Glanzkanal nur die Lichter gespiegelt werden, kann man im Spiegel-Kanal die komplette Spiegelung

eines Objektes beeinflussen. Da die gleichen Stellen, die Lichtquellen spiegeln, auch normal spiegeln sollen, wird auch die Glanztextur im Spiegel-Kanal verwendet. Einzig die Intensität der Spiegelung wird später noch etwas reduziert.

Generell sei gesagt: Jedes Objekt in unserer Welt spiegelt. Sei es direkt, was wir als spiegelnde Materialien wie Glas, Chrom etc. sehen oder all die anderen, die zumeist indirekt spiegeln<sup>18</sup>.

#### 4.3.4 Relief-Kanal

Mit Hilfe des Relief-Kanals (*Bump*) kann man eine Oberflächenstruktur simulieren, ohne, dass sich dabei die Geometrie des Objektes verändert. Hierbei wird lediglich die Schattierung der Glanzpunkte der Objekte verändert und so eine Geometrieänderung simuliert.

Aufgrund der Tatsache, dass das Bump-Mapping die Oberfläche nicht verändert, ergeben sich folgende Einschränkungen [Bir]:

- Die Außenlinien oder Silhouetten eines Objekts werden von dem Relief-Kanal nicht beeinflusst und bleiben glatt.
- Die Schatten des Objektes mit Bump-Mapping behalten ihre ursprüngliche Form, sind also unverändert.
- Schatten, die auf eine Relief-Oberfläche fallen, bleiben gerade und werden nicht wie auf einer wirklich rauen Oberfläche verzerrt.
- Im Gegensatz zu modellierten Strukturen oder Displacement-Details werfen Reliefs keine Schatten auf das Objekt an sich.
- Die Kontur der Überschneidungen zwischen einem Relief-Objekt und einem weiteren Modell wird nicht verändert, die reale Form bleibt also auch hier sichtbar.

Da allerdings die Geometrie nicht verändert wird und vor allem nicht komplexer wird, bleibt die Berechnungsgeschwindigkeit sehr gering und die Verwendung von Bump-Maps recht einfach. Bei kleineren Details oder bei Szenen, wo die Nachteile nicht sichtbar werden, sollte man auf alle Fälle Bump-Maps verwenden.

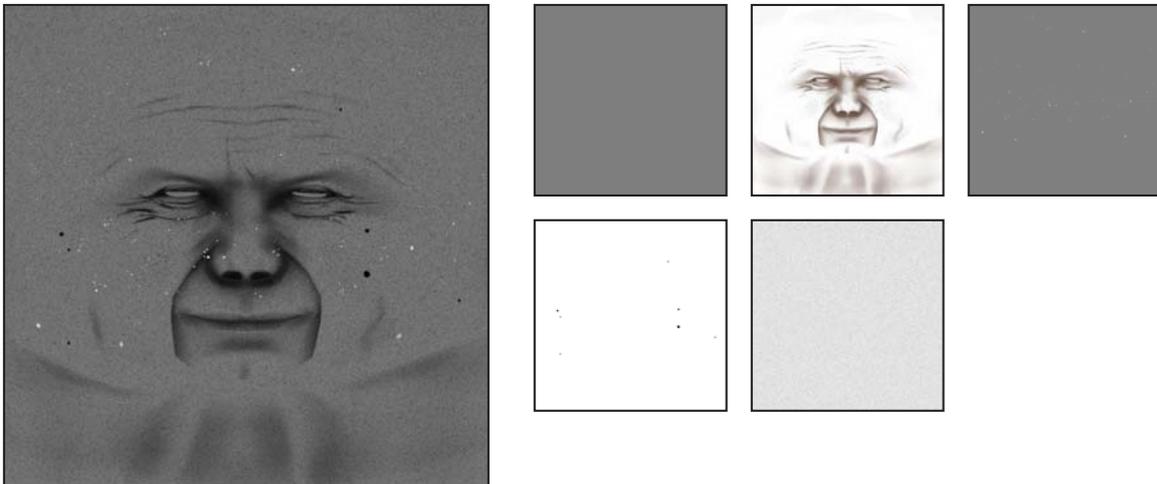
Zur Steuerung des Reliefs dient wieder eine Grauwert-Textur. Hierbei gilt reines Schwarz als Vertiefung und reines Weiß als Erhöhung. Meist wird dadurch ein mittleres Grau als Startwert genommen um Vertiefungen und Erhöhungen zu simulieren, je nach Anwendung, was mehr Verwendung finden soll.

Zur Erstellung einer Relief-Textur brauchen wir Vertiefungen und Erhöhungen. Also wird ein mittleres Grau #7F7F7F als Grundton genommen, der genau zwischen Weiß und Schwarz in der Mitte liegt. Die Ambient-Occlusion-Textur, welche grob genommen auch aus Höheninformationen besteht,

---

<sup>18</sup> vgl. Kapitel 6.2.3 Global Illumination

wird als nächstes eingearbeitet. Hierdurch kann man die Struktur noch mehr verdeutlichen. Danach werden weiße Punkte integriert für erhöhte Hautflecken und schwarze Punkte für tiefe Hautunreinheiten. Zusammen mit einer abschließenden Noise-Ebene bildet dies die Relief-Textur.



**Abbildung 4-5: Relieftextur:** Die Bumpmap wird gebildet aus einem mittleren Grau, Ambient Occlusion, hohe Flecken, tiefe Flecken und einem Noise.

#### 4.3.5 Displacement-Kanal

Im Vergleich zum Relief-Kanal bietet der Displacement-Kanal eine noch bessere Möglichkeit. Bei den Displacement-Maps<sup>19</sup> wird die Geometrie durch Verwendung einer Schwarz/Weiß-Textur unterschiedlich stark verändert. Hierbei werden die Polygone je nach Helligkeitswert der Textur entlang den Normalen verschoben. Reines Schwarz bedeutet maximal nach innen und reines Weiß bedeutet maximal nach außen verschieben, je nach eingestellter Stärke.

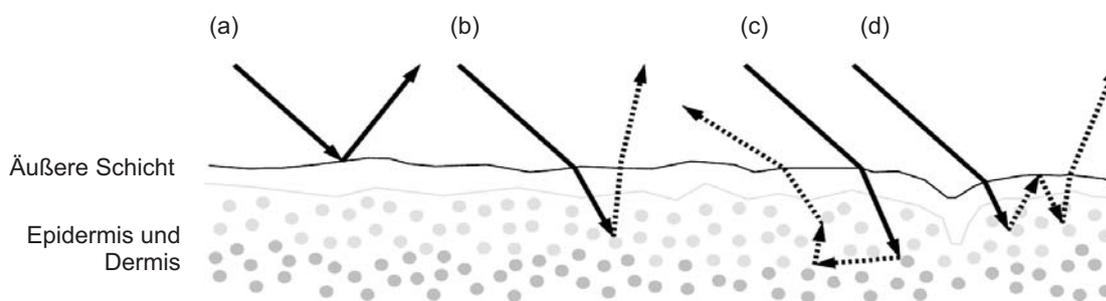
Um eine Displacementverformung zu erhalten muss die Geometrie allerdings auch genug unterteilt sein, da nur die vorhandenen Punkte berücksichtigt werden. Entweder die Geometrie wird vorher manuell unterteilt, was bei der weiteren Verarbeitung problematisch werden kann oder man benutzt das Subpolygon-Displacement, wodurch eine Unterteilung erst kurz vor der Bildberechnung erfolgt. Allerdings wirkt sich die Polygonunterteilung jeweils negativ auf die Rechenleistung und letztendlich auf die Renderzeiten aus.

<sup>19</sup> vgl. Kapitel 3.4.3 Displacement

## 4.4 Besonderheit Haut

Beim Versuch der fotorealistischen Darstellung eines menschlichen Charakters ist die Erscheinung der Haut ein wichtiges Merkmal für die Glaubwürdigkeit [Hel]. Einige der Grundmerkmale wurden bisher versucht mit dem Farbkanal, den Highlights und dem Spiegelkanal zu erschaffen. Allerdings haben Materialien wie Haut eine Besonderheit, die es schwer macht, sie realistisch nachzubilden. Haut ist leicht transparent und leitet, verteilt und absorbiert das eindringende Licht unter der Oberfläche weiter [Asa].

Schaut man sich den Aufbau genauer an, dann besteht die Haut des Menschen aus drei Schichten: Oberhaut (*Epidermis*), Lederhaut (*Dermis*) und schließlich die Unterhaut (*Subcutis*). Dieser prinzipiell an allen Hautabschnitten vorhandene dreischichtige Aufbau variiert jedoch hinsichtlich seiner Ausdehnung. Und auch diese drei Hautschichten bestehen wiederum aus weiteren verschiedenen Schichtungen, die sich wiederum in Form und Größe je nach Ort unterscheiden. [Kün]



**Abbildung 4-6: Lichtreflektionen der Haut.** Die unterschiedlichen Hautebenen reflektieren das Licht unterschiedlich. Einige Lichtstrahlen werden sofort reflektiert (a), während andere in die Haut eindringen, ihre Eigenschaft ändern und später wieder austreten (b-d).

Einige dieser Bestandteile sind nun lichtdurchlässig, während andere das Licht vollständig absorbieren, reflektieren und zusätzlich verfärben. Durch das Zusammenspiel dieser unterschiedlichen Eigenschaften wird das Licht scheinbar innerhalb der Haut zufällig gestreut und gibt der Haut ihre besondere transluzente Charakteristik (vgl. Abbildung 4-6).

### 4.4.1 Subsurface Scattering

Dieser Effekt der „Lichtstreuung unterhalb der Oberfläche“ wird *Subsurface Scattering* (SSS) genannt. Dieser wird an drei Stellen sichtbar [nach Bir]:

- Ein scheinbares Glühen des Objektes insbesondere an dünnen Stellen erfolgt bei einer Beleuchtung von hinten. Eine Lichtquelle direkt hinter einem Ohr oder der Nase positioniert bringt diese zum rötlichen leuchten. Dies wird durch das *Forward Scattering* bewirkt, wobei das Licht auf einer Seite eindringt, dort gestreut wird und auf der anderen Seite wieder

austritt.

- Der Terminator, also der Übergang zw. Beleuchteter und im Schatten liegenden Stellen, bekommt einen rötlichen Schimmer. Dieser Effekt ist auch auf das *Forward Scattering* zurückzuführen.
- Die Ränder der Schatten können rötlich gefärbt sein. Dieser Effekt kommt durch ein *Back Scattering* zustande, bei dem die eintretenen Lichtstrahlen gestreut werden und anschließend an fast identischer Stelle wieder austreten.

Um diesen Effekt des SSS bei einem virtuellen Charakter realistisch zu erzeugen gibt es mehrere Möglichkeiten. Am effektivsten scheinen ein Leuchtenkanal und eine SSS-Simulation dabei zu sein:

#### 4.4.1.1 Texturen

Bei nicht animierten Bildern kann der Effekt rechenarm und damit sehr effektiv mit Hilfe der Texturen nachgebildet werden. Die Farbtextur sollte eine etwas mehr gesättigte Farbgebung erhalten. Zusätzlich kommt der Leuchtenkanal zum tragen. Hierbei steuert eine Textur die Stellen, an denen die Haut mehr oder weniger leuchten soll. Ein geringes Grundleuchten sollte hierbei zusätzlich verwendet werden. Als Leuchtfarbe führt ein helles Rot zu den besten Ergebnissen.



**Abbildung 4-7: Subsurface Scattering.** Links ohne und rechts mit SSS. Der Effekt ist gut an Ohren oder Nase zu beobachten, wenn eine Lichtquelle direkt dahinter platziert wird.

Diese sehr einfache Nachbildung des SSS ist allerdings nur bei Stillbildern zu verwenden. Da sie abhängig von Kamerawinkel und Position der Lichtquelle ist, macht diese Technik bei Bewegtbildern wenig Sinn. Die Textur müsste für jedes weitere Bild neu erzeugt werden.

#### 4.4.1.2 SSS-Simulation

Mittlerweile gibt es allerdings auch viele verschiedene Algorithmen, die den Effekt des SSS relativ gut nachbilden können. Diese Funktionen sind heutzutage auch in den meisten 3D-Programmen integriert. Somit ist es nun möglich, den SSS-Effekt auch in Animationen problemlos einzusetzen.

Auch wenn man die Transluzenztiefe der Objekte bei dieser Funktion bestimmen kann, empfiehlt es sich für ein realistisches Ergebnis, die Knochen in groben Zügen mitzumodellieren. Beim Kopf sollte also eine Art Schädel in groben Zügen erstellt werden, welcher relativ schnell aus der vorhandenen Geometrie des Kopfes erzeugt werden sollte. Durch Entfernen von Ohren und Nase und eine weitere Glättung der Geometrie erhält man schnell einen Schädel, der hierfür ausreichen sollte und der im Inneren des Kopfes positioniert wird. Dem Kopf wird schließlich die SSS-Eigenschaft zugeordnet und dem inneren Schädel wird ein undurchdringliches Material zugewiesen.

Der Vorteil der SSS-Simulation ist der geringere Arbeitsaufwand und die schnellen und zufriedenstellenden Ergebnisse hinsichtlich des Realismuseffektes, Animationen inbegriffen. Dies wird allerdings erkauft mit einem erhöhten Verbrauch an Rechenleistung.

»Für eine realistische Wirkung kommt es nicht auf die extremen Darstellungen an, sondern auf die Erscheinung der alltäglichen Dinge - und zur Lichtdurchdringung der Haut kommt es jede Sekunde und überall.« [Dave Seager<sup>20</sup>]

### 4.5 Anwendungen in der Industrie

Um glaubwürdige und damit realistische Gesichter zu erschaffen, wurden für alle Figuren aus dem Film *Final Fantasy: The Spirits Within* (2001, Square Pictures) die Texturen mit möglichst vielen Details versehen. Alle Texturen erhielten Falten, Hautunreinheiten, Altersflecken und Sommersprossen. Einerseits wurden die Gesichter durch das Hinzufügen dieser Details glaubwürdiger [Ken], andererseits wurde durch die Integration von 3D-Strukturen wie den Falten in eine 2D-Textur die Szene im Hinblick auf Speicher und Renderzeit optimiert [Hyp].

Die virtuelle Figur des Dr. Sid aus *Final Fantasy* wurde allgemein als die Figur mit dem realistischsten Gesicht aufgenommen [Ken]. Dem Animation Director Andy Jones zufolge sei dies darauf zurückzuführen, dass man bei diesem älteren Herren mehr solcher Details hinzufügen konnte, als bei den anderen. Weiterhin meint er: »Das Hinzufügen von Altersflecken und ähnlichen Details führt dazu, dass ein Charakter realistischer aussieht.« Bei anderen Figuren brachte dieser einfache Ansatz allerdings Probleme. Die Person der Aki sollte einerseits hübsch und perfekt wirken, aber für eine realistische Wirkung mussten Details wie Hautunreinheiten integriert werden. Die Balance zu finden ist unter solchen Voraussetzungen schwerer.

Der Effekt des SSS, der dafür sorgt, dass Haut leicht lichtdurchlässig ist und nicht wie eine Plastik wirkt, wird zur realistischen Simulation der Haut seit einigen Jahren in der Filmindustrie

---

<sup>20</sup> Dave Seager war Lightning Supervisor bei *Final Fantasy: The Spirits Within* (2001, Square Pictures).

standardmäßig angewendet. Schon im ersten Shrek-Film (2001 / Universal) wurde versucht den Effekt des SSS durch verschiedene Texturlayer zu simulieren. Auch bei Yoda aus *Star Wars: Episode III - Revenge of the Sith* (2005, Lucasfilm / Twentieth Century Fox) [Rob5] oder Gollum aus der *The Lord of the Rings: The Two Towers* (2002, New Line Cinema) [Ros] wurde dieser Effekt beispielsweise verwendet.

*Spider-Man 2* (2004, Marvel Enterprises / Columbia Pictures) glänzt mit vielen atemberaubenden Effekten wie digitalem Wasser, Feuer und Explosionen. Doch die Effekte, die am wenigsten zu bemerken waren, gehörten zu den technisch innovativsten Techniken: Die Rede ist von den digital erschaffenen Spider-Man und Doc Ock. [Rob4]

»Mir war von Anfang an klar, dass unser Doc Ock die größte Herausforderung sein würde, da er in vielen Nahaufnahmen gezeigt wird.« [Scott Stokdyk<sup>21</sup>]

Bei der Herstellung der digitalen Charaktere entschied sich Stokdyk gegen Texturen mit Subsurface Scattering, weil ihm damit zu viele künstlerische und technische Entscheidungen getroffen werden mussten. Im Gegensatz dazu wurde die *Light-Stage-Technologie* verwendet, die von Paul Debevec entwickelt und am Institut for Creative Technology der University of Southern California verwirklicht worden war [Debe].

Dabei wurden die Gesichter beider Schauspieler am Institute for Creative Technology mit verschiedenen Lichtverhältnissen von allen Seiten mit Kameras aufgenommen. Dadurch erhält man Informationen, wie das Licht die Personen ausleuchtet. Im nächsten Schritt werden HDRI-Umgebungsbilder verwendet, welche als eine Kugel über den digitalen Charakter diesen dann ausleuchten, wobei die gesammelten Informationen verwendet werden.

»Diese Technik ist teuer im Aufbau, verlangt viel Arbeit und jede Menge Speicherplatz. Und man benötigt die Schauspieler. Daher hoffe ich, dass das Subsurface Scattering im Laufe der Zeit verbessert und vereinfacht wird.« [Peter Nofz<sup>22</sup>]

## 4.6 Fazit

Mit Texturen können Farben und weitere Details ressourcenschonend in einen virtuellen Charakter eingearbeitet werden, die den Realismusgehalt deutlich steigern können. Ein 3D-Objekt wie ein virtueller Charakter kann durch Texturen optisch mehr Falten und weitere Hautunebenheiten bekommen, ohne dass die Geometrie tatsächlich verändert wird. Texturen können dadurch auf sehr einfache Weise das Ergebnis deutlich verändern und glaubwürdiger gestalten. Aus diesem Grund sollte den Texturen bei dessen Erstellung sehr viel Aufmerksamkeit beachtet werden.

---

<sup>21</sup> Scott Stokdyk war Visual Effects Supervisor bei *Spider-Man 2* (2004, Marvel Enterprises / Columbia Pictures).

<sup>22</sup> Peter Nofz war CG-Supervisor bei *Spider-Man 2* (2004, Marvel Enterprises / Columbia Pictures).

Als Regel sollte man bei virtuellen Charakteren gelten: Den Farb- und Diffuskanal wird benutzt, um kleine Details wie Hautunreinheiten und kleine Flecken einzuarbeiten. Mit Bump-Maps können weitere Details wie Falten vorgetäuscht werden und Displacement Maps sollten für die größeren Details benutzt werden. Glanz und Spiegelkanal simulieren die Hautreflektionen der Hautoberfläche.

Eine realistische Hautoberfläche eines Menschen zu erreichen ist eine schwierige Aufgabe und ist nur unter Ausnutzung mehrere Texturkanäle möglich. Eine optimale Haut ist insbesondere bei Animationen effektiv nur durch Sub Surfaces Scattering zu ermöglichen. Ohne ein richtiges SSS sieht die Haut bei virtuellen Charakteren meist wie Plastik und damit sehr künstliche aus.

Die Verwendung einer Light-Stage kann man Perfektionierung des SSS ansehen, ist aber aufgrund der nötigen Hardware nicht für jedermann alltagstauglich und für kleinere Firmen und Privatpersonen nicht kosteneffizient. Kleine Dinge können auch hier zu großen Problemen führen: Die Schauspieler Alfred Molina, der den Doc Ock spielte trug am Tag der Light-Stage-Aufnahmen Monate vor dem eigentlichen Dreht noch Kotletten, die er im Film nicht trägt. Und auch der Schauspieler Tobi Maguire trug während der Light-Stage-Aufnahmen eine andere Haarfarbe. Solche Fehler mussten korrigiert werden und dabei noch auf jedem Frame identisch aussehen. Eine Heidenarbeit für die Nachbearbeiter. [Rob4]

In Abbildung 4-7 sind die angewendeten Texturen zusammenfassend dargestellt.



**Abbildung 4-8: Angewendete Texturen:** Ausgangsgeometrie, Displacementmap, Bumpmap, Subsurface Scattering, Specular, Colormap und finales Rendering mit allen Texturen.

## 5 Haare

Der Mensch besitzt insgesamt 300.000 bis 500.000 Haare, wovon ca. 25 Prozent auf die Kopfbehaarung entfallen [Meye]. Ein blonder Mensch hat üblicherweise zwischen 120.000 bis 150.000 Haare auf dem Kopf, ein dunkelhaariger Mensch etwa 90.000 bis 110.000 Haare [Asa].

Durch diese hohe Anzahl an Haaren ist es schon ersichtlich, dass die Erzeugung realistischer Haare bei einem virtuellen Charakter Schwierigkeiten aufweist. Es ist fast ausgeschlossen diese Anzahl an Haaren auf geometrischen Weg zu erzeugen und damit noch weiter zu arbeiten. Beispielsweise eine Bewegung der Haare realistisch zu animieren, ist wegen der hohen Anzahl so gut wie unmöglich. Allerdings gibt es mehrere Möglichkeiten, Haare dennoch realistisch zu erzeugen.

### 5.1 Feste Haargeometrie

Die einfachste Weise Haare nachzubilden, ist eine geometrische Nachbildung der Haare als Ganzes. Es wird nicht jedes einzelne Haar betrachtet, sondern die Form aller Haare. Anschließend wird diese Geometrie mit einer Haartextur überzogen. Durch die Verwendung einer einfachen Geometrie als Haarnachbildung liegt der Vorteil in der Einsparung an Rechenleistung. Nicht unzählige Haare müssen verarbeitet und gerendert werden, sondern nur eine einfache Geometrie.

Allerdings weist diese Art der Haarerzeugung auch mehrere Nachteile auf:

- Was aus der Ferne gut aussieht, scheint bei näherer Betrachtung wenig mit einem realistischen Haarverlauf zu tun zu haben.
- Je länger die Frisur ist, desto unrealistischer sieht sie aus. Nur Kurzhaarfrisuren machen bei dieser Technik Sinn.
- Eine Animation ist ausgeschlossen. Dadurch wirken diese Haare bei animierten Szenen schnell wie ein Fremdkörper, wie Comic-Haare.

### 5.2 Haartexturen

Eine weitere einfache Möglichkeit, Haare zu erzeugen, ist das Ausnutzen des Alphakanals<sup>23</sup> der Texturen. Anstelle tausend kleiner Haare zu erzeugen, nimmt man eine flache Geometrie (meist eine einfache Fläche) und projiziert darauf eine Haartextur. Unter Benutzung des Alphakanals erhält man mit einer einfachen Geometrie den Eindruck, viele Haare erzeugt zu haben. Meist reichen hierzu schon wenige solcher Flächen aus, damit das Ergebnis im Gegensatz zu der festen Haargeometrie realistisch aussieht. Auch ist nun eine Animation realistischer zu erzeugen, da man nun mehrere „Haarsträhnen“ zur Verfügung hat, um sie zu bewegen.

---

<sup>23</sup> Ein Alphakanal steuert die Sichtbarkeit eines Elementes. Texturen können zu dieser Steuerung benutzt werden.

Diese Technik ist noch recht schonend im Umgang mit der Rechenperformance im Gegensatz zu tausend erstellten Geometriehaaren. Allerdings wird mehr Speicher verbraucht, als bei der ersten Variante, da hier erst alle Alphakanäle berechnet werden müssen.

Auch wenn diese Technik schnell und einfach zu realisieren ist, weist sie auch einige weitere Nachteile auf:

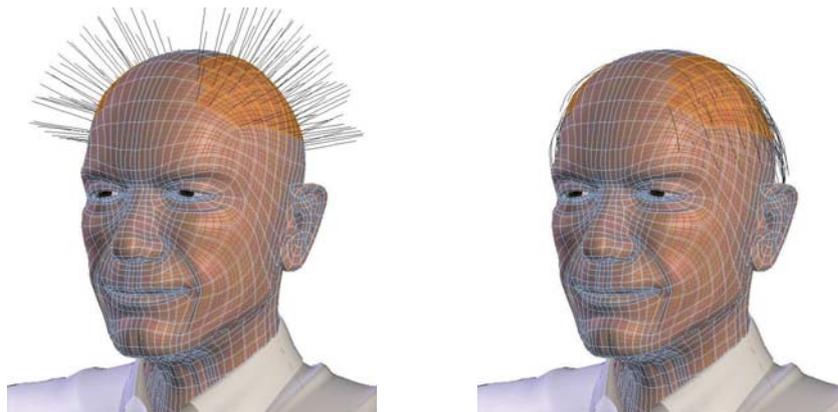
- Auch diese Technik sieht von weiten gut aus. Aber aus unmittelbarer Nähe betrachtet und besonders aus unterschiedlichen Blickwinkeln betrachtet, fallen die Haarflächen auf.
- Der Eindruck von Volumen ist schwer zu realisieren.

Diese Art der Haarsimulation wird sehr oft an Stellen genutzt, bei der die Haare sehr klein und unscheinbar sind. So werden zum Beispiel oft die Augenlider bei virtuellen Charakteren mit dieser Technik nachempfunden.

### 5.3 Haarsimulation

Die meisten 3D-Modellierungsprogramme bieten heutzutage eigene Funktionen an, um Haare mit einer hohen Anzahl äußerst realistisch, handhabbar und mit einem akzeptablen Verbrauch an Rechenleistung dazustellen.

Um eine Haarpracht zu simulieren sollte man mit ca. 10.000 bis 20.000 Haaren auskommen [Asa]. Mit einfachen kleinen Tricks kann man dem auch etwas nachhelfen. So kann man bei den Texturen<sup>24</sup> an manchen Stellen eine Haargrundierung in der Farbe der Haare einarbeiten und somit mit etwas weniger Haaren auskommen. Diese Farbgrundierung entspricht den Füllhaaren, die meist dazu genutzt werden um die Kopfhaut zu verdecken.



**Abbildung 5-1: Haarguides:** Angewendet auf markierten Polygone und endgültige Form und Länge der Guides auf den entsprechenden Polygonen.

<sup>24</sup> vgl. Kapitel 4.3.1 Farbkanal

### 5.3.1 Guides anlegen

*Guides* sind die wichtigsten Hilfsmittel, um die Haare zu steuern. Dabei repräsentiert ein Guide meist mehrere Hunderte von Haaren, indem die Haare zwischen den Guides einfach interpoliert werden. Dies hat den Vorteil, dass man die Haare nur mit einer überschaubaren Anzahl von Guides steuert, was die Handhabung sehr erleichtert. Zusätzlich wird die Rechenperformance weniger beansprucht.

Um die Guides zu positionieren, wählt man in den meisten 3D-Programmen die gewünschten Polygone des Kopfes aus und weist sie anschließend einem Haarobjekt zu. Anschließend werden die Guides entsprechend der gewünschten Lage verformt und positioniert und schließlich die verschiedenen Parameter eingestellt, wie Anzahl, Länge, Unterteilung, Gravitation, Kollisionsobjekte usw.

### 5.3.2 Texturen zum Steuern der Haare

Um die Haare zu positionieren, kann man wie eben beschrieben, mehrere Polygone des Kopfes auswählen. Eine weiterführende Möglichkeit, die Haare zu positionieren, ist die Verwendung von Grauwert-Texturen. Der große Vorteil an den Texturen ist, dass man nicht an Positionen der Polygone beschränkt ist, wenn man zum Beispiel kleine Flächen wie die Augenbrauen positionieren möchte. Ein weiterer Vorteil der Texturen ist, dass man mit den Farb- bzw. Grauwertverläufen die Haare besser und genauer steuern kann. Reines Weiß in der Textur heißt volles Haar bzw. 100 Prozent des zu generierenden Haares, während reines Schwarz keine Haare bedeutet. Mit den Grauwerten dazwischen kann man leicht Übergänge schaffen, damit das Haar nicht mit einmal aufhört und sich eher natürlich verhält.



**Abbildung 5-2: Texturen zum Steuern der Haarverteilung:** Haupthaar, Haare oben, zusätzliche Haare oben, Barthaare, Bartstoppel und Augenbrauen.

Mit diesen Texture-Maps kann man im Grunde mehrere Haareigenschaften steuern. Folgende sind aber vollkommen ausreichend:

- **Länge:** Einerseits wird die Länge der Haare lokal durch die Guides bestimmt. Durch die Textur können diese Werte verfeinert werden, wobei reines Weiß hier 100 Prozent der Länge der Guides entspricht und reines Schwarz 0 Prozent. Dadurch steuert man indirekt die Verteilung der Haare.
- **Dicke:** An Stellen, wo wenige Haare sein sollen, wollen wir keine kurzen Stoppeln haben, also wird die Dicke der Haare ebenso über die gleiche Textur gesteuert.

### 5.3.3 Haarmaterial

Viele essentielle Eigenschaften der Haare werden über die Definition des Haarmaterials gesteuert. Dies kann ähnlich den Textur-Kanälen verstanden werden<sup>25</sup>. So wird die vorher beschriebene Länge und Dicke, die über die Textur feinjustiert wird, hier eingestellt. Weitere wichtige Eigenschaften sind die Folgenden:

- **Farbe:** Die Farbe sollte mehrere Unterteilungen bekommen, da Haare selten die gleiche Farbe aufweisen. Bei diesem Charakter wird ein helles Grau an der Wurzel, über ein dunkles Grau in der Mitte, hin zu einem hellen Grau an der Spitze verwendet. Die Barthaare bekommen zwei Abstufungen eines sehr hellen Graus, die Augenbrauen ein mittleres Grau und die Bartstoppeln ein dunkleres Grau.
- **Glanzlicht:** Haare spiegeln das umgebene Licht. Dadurch werden auch Highlights in den Haaren, je nach Position der Lichter, zu sehen sein. Dem wird hier mit mehreren flachen Glanzlichtern Rechnung getragen.
- **Verformungen:** Haare verhalten sich unterschiedlich. Unterschiedlichste Verformungen können eingestellt werden, wie zum Beispiel Kräuseln, Büschelbildung, Locken, Wellen etc. Bei unserem Charakter reicht etwas Büschelbildung und etwas verbiegen bei dem Haupthaar aus. Bei dem Bart und den Augenbrauen wird noch etwas Kräuseln hinzugefügt.



**Abbildung 5-3: Haarsimulation:** Dargestellte Guidelines; interpolierte Haare; Haare verrechnet mit Haartexturen und Farbe; finales Rendering.

## 5.4 Anwendungen in der Industrie

Aufgrund der beschränkten Rechenmittel von 1995 machte Pixar aus der Not eine Tugend und erstellte sämtliche Haare in dem Animationsfilm *Toy Story* (1995 USA / Disney) durch eine feste

<sup>25</sup> vgl. Textur-Kanäle in Kapitel 4 Texturen

Geometrie.

2001 standen schon weit mehr Rechenkapazitäten zur Verfügung. Die Figur Aki in dem Animationsfilm *Final Fantasy: The Spirits Within* (2001, Square Pictures) wurde mit 60.000 Haaren ausgestattet. Die Haare waren der zeitraubendste und rendering-intensivste Teil. Um die Rechenzeit nicht weiter in die Höhe zu treiben, erhielten alle anderen Figuren extrem kurze Frisuren [Hyp]. Kontrolliert wurden all die Haare über Guides, die die Länge, Dichte und weitere Parameter bestimmten [Ken]. Auch bei den Produktionen von Pixar wurde inzwischen eine eigene Haarsimulation entwickelt, die die Bewegung der Haare physikalisch korrekt steuern ließ. Zum Einsatz kam diese zum Beispiel im Film *Die Monster AG* von 2001.

Die Entwicklung von Hui Buhs Haaren aus dem Film *Hui Buh - Das Schlossgespenst* (2006, Rat Pack / Constantin Film) beanspruchte fast genauso viel Zeit, wie die eigentliche Charakter-Entwicklung. Auch hier wurde eine Haarsimulation benutzt. Da allerdings am Anfang der Produktion die Kopfhaut immer wieder durchschimmerte, musste ein System mit drei Kopfhaarschichten entwickelt werden: Ein dunkles Haarsystem als „Futter“, ein zweites als „Füller“ und ein drittes, um den strähnigen Look zu erreichen [Sch].

## 5.5 Fazit

Die Verwendung von einer einzigen Haargeometrie um tausende Haare zu simulieren ist heutzutage nicht mehr gebräuchlich. Diese Verwendung stammt aus einer Zeit, wo noch weniger Rechenleistung vorhanden war. Selbst bei aktuellen Echtzeitrenderern werden Haare derzeit mit Polygonflächen und Benutzung der Alphakänale erzeugt, um möglichst realistische Ergebnisse zu erhalten. Als Beispiel kann man hier das Spiel *Resistance 3* (2011, *Insomniac Games*, *Playstation 3*) nennen (vgl. Abbildung 5-4).



**Abbildung 5-4: Haare in Resistance 3:** Die Haare wurden mit Polygonflächen und Alphakanal erzeugt. Erkennbar sind die Polygonkanten der einzelnen Flächen.

Aufgrund ihrer guten Handhabung und der aus heutiger Sicht recht annehmbaren Rechenperformance sind Haarsimulationen bei den prerendered Bildern die perfekte Wahl, um Haare zu kreieren. Es ist schnell möglich, mehrere Tausende Haare zu simulieren. Die angewendeten Haare bei dem hier erstellten Charakter wiesen zum Schluss eine Anzahl von ca. 120.000 auf dem Kopf auf. Durch das Alter der Referenzperson entfielen die meisten Haare allerdings auf den Bart:

- 29.000 Haare für das Haupthaar
- 86.500 Haar für den Bart (davon 45.000 Haare als kurze Bartstoppeln)
- 5.000 Haare für die Augenbrauen

Allerdings sollte auch hier beachtet werden, dass nur so viel Haare simuliert werden sollten wie unbedingt notwendig. Wenn Haare kaum sichtbar oder nur an kleinen Stellen sichtbar sind, reichen weniger Haare aus, die etwas dicker sind. An manchen Stellen, wie den Augenlidern, können auch Haartexturen mit angewandtem Alphakanal Verwendung finden.

## 6 Umgebung

In den vorangegangenen Kapiteln wurde ein Mensch als 3D-Geometrie aufgebaut. Dieser muss aber erst in „Szene“ gesetzt werden. Hierzu gehört die Positionierung der Kamera, das Hinzufügen von Lichtern, die Erzeugung von Schatten und das Einfügen von Bildstörungen zur Erhöhung der realistischen Wirkung, bevor das finale Bild erstellt werden kann.

### 6.1 Bildberechnung

Um die Arbeitsweise der Licht- und Schattenberechnung besser beleuchten zu können, soll an dieser Stelle eine kurze Einführung in die Bildberechnung gegeben werden.

#### 6.1.1 Raytracing

Das Rendern, also die Bildberechnung aus den vorhandenen 3D-Objekten, erfolgt bei vielen gängigen Renderern mittels *Raytracingverfahren*. Dieser Algorithmus arbeitet dabei nach den physikalischen Grundsätzen der realen Welt. In der Natur senden Lichtquellen Strahlen aus, die von Objekten absorbiert, aufgespalten oder reflektiert werden. Das menschliche Auge interpretiert das aufgefangene Licht entsprechend seiner Intensität und dem Lichtspektrum nach als Helligkeit und Farbe.

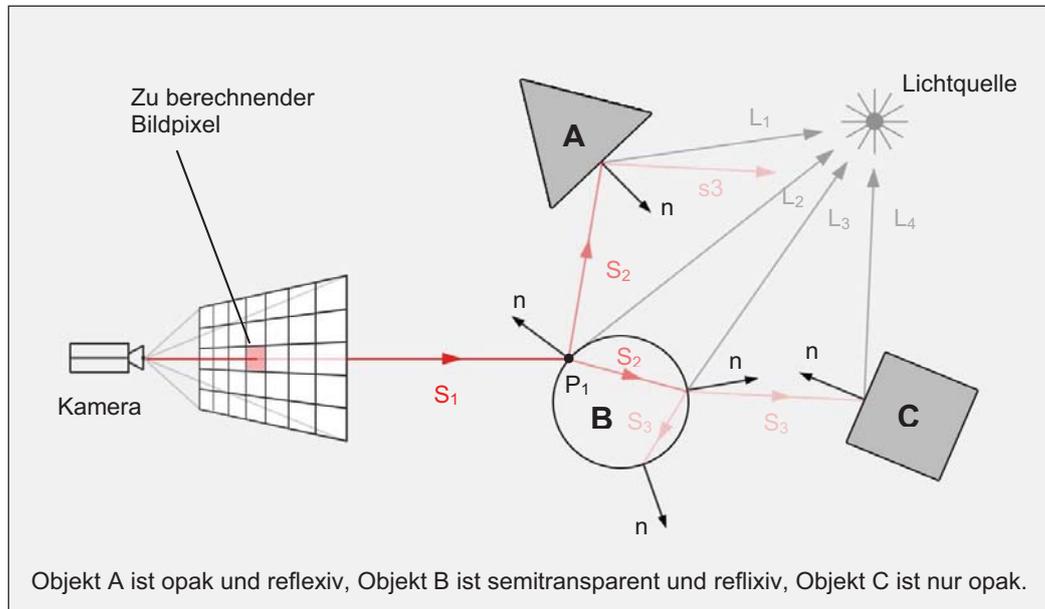
Beim Raytracing<sup>26</sup> wird dieser natürliche Prozess in entgegengesetzter Richtung durchgeführt. Von einer gesetzten virtuellen Kamera wird für jeden Bildpunkt ein Strahl ausgehend vom Ursprung der Kamera in die Szene geschickt [Cha]. Entscheidend dafür sind die Position, der Blickwinkel und die eingestellten Auflösungen der Kamera. Trifft ein solcher Sichtstrahl auf ein Objekt, existiert ein Schnittpunkt. Dieser Schnittpunkt und dessen Entfernung zur Kamera werden abgespeichert und weiter interpretiert. Befinden sich mehrere Schnittpunkte auf einem Strahl, so wird nur der Punkt mit der geringsten Entfernung zur Kamera weiter verarbeitet. Entferntere Schnittpunkte entsprechen hierbei verdeckten Objekten. Um die Geschwindigkeit zu optimieren, werden vor dem Raytracing intern alle parametrischen und sonstigen Objekte in Polygonflächen aufgeteilt, da diese schneller berechnet werden können.

Wurde ein Schnittpunkt für den zu interpretierenden Bildpixel gefunden, wird im nächsten Schritt der Schnittwinkel des Strahls zum betreffenden Objekt berechnet. Hierfür wird der Normalenvektor des geschnittenen Polygons verwendet, mit dessen Hilfe der Schnittwinkel einfach berechnet werden kann. In einem weiteren Schritt wird für den gefundenen Schnittpunkt in ähnlicher Weise der Winkel zu den in der Szene vorkommenden Lichtquellen berechnet. Aus den Winkeln berechnet sich dadurch die Helligkeit des Bildpixels. Materialeigenschaften steuern die Farbe bei und können nun den Bildpixel definieren.

---

<sup>26</sup> Raytracing, engl., für Strahlenverfolgung

Es können weitere Strahlenberechnungen notwendig werden. Haben Objekte spiegelnde Oberflächen, dann müssen die Strahlen entsprechend den Normalenvektoren nach reflektiert werden und der Vorgang wird entsprechend von vorn wiederholt. Auch bei Transparenzen können mehrere Strahlen entstehen, die berechnet werden müssen. Aufgrund der möglichen rekursiven Strahlberechnungen gibt es meist eine maximale Strahlentiefe als Abbruchkriterium der Berechnung.



**Abbildung 6-1: Raytracingverfahren:** Der Sichtstrahl  $S_1$  spaltet sich am Schnittpunkt  $P_1$  durch Reflektionen und Transparenz in zwei Strahlen  $S_2$  und in weitere Strahlen  $S_3$  auf. An den Objektschnittpunkten werden entsprechend den Winkeln zwischen Normalen  $n$  und den Lichtstrahlen  $L$  die Helligkeiten zugewiesen.

### 6.1.2 Perspektive

Die virtuelle Kamera, von der aus die Bildberechnung erfolgt, sollte sich ungefähr auf Augenhöhe des virtuellen Charakters befinden. Dadurch entstehen die natürlichsten Aufnahmen [Bir]. Der Mensch ist es gewöhnt, andere Menschen auf ungefähr gleicher Höhe in die Augen blicken zu können. Dadurch wirkt ein Bild von einem Menschen auch aus dieser Perspektive am vertrautesten.

Intern arbeiten diese virtuellen Kameras, wie im vorhergehenden Kapitel beschrieben, mit einzelnen Sehstrahlen, die die Szene abtasten, um einen Bildpunkt zu berechnen. Letztendlich arbeitet solch ein Renderprogramm also wie eine einfache Lochkamera. Dennoch versuchen die Renderalgorithmen echte Kameras nachzuahmen und verhalten sich wie abbildende Linsensysteme, um die Bilder zu erzeugen [Max]. Dementsprechend gibt es im Sinne der realen Kameras bei den virtuellen Kameras Einstellmöglichkeiten für Brennweite, Bildgröße und Gesichtsfeld.

Bei der Positionierung der Kameras gelten schließlich die gleichen Gesetzmäßigkeiten, wie bei der realen Fotografie: Die Kamera sollte nicht zu nah an eine Person herangerückt werden, da sich hier bedingt durch die niedrige Brennweite Verzerrungen bilden können. Im Extremfall kann ein Kopf eine

zu große Nase bekommen, während die Kopfform ungewöhnlich klein für sein Gesicht wirkt. Im Gegensatz dazu führt eine zu große Entfernung und dabei verwendete hohen Brennweiten zu einem sehr platten Kopf, bei dem das Gesicht im Verhältnis zum gesamten Kopf ungewöhnlich klein wirkt und die Ohren seitlich abzustehen scheinen (vgl. Abb. 6-2). Bei der Erzeugung der Bilder des 3D-Charakters wird die Kamera mit einer Entfernung von ca. 2m und der Benutzung einer 72 mm Brennweite gerendert und als am natürlichsten eingestuft.



**Abbildung 6-2: Unterschiedliche Brennweiten:** 24 mm (links), 72 mm (Mitte), 400 mm Brennweite (rechts).

### 6.1.3 Antialiasing

Standardmäßig wird ein Bild beim Raytracing pixelweise aufgebaut, wobei jeder Pixel für sich per Strahlenverfolgung berechnet wird und unabhängig zu seinen Nachbarn interpretiert wird. Dabei entstehen auflösungsbedingt an Kantenübergängen kontrastreiche Farbabstufungen bei den Pixeln [Cha]. Ein Pixel charakterisiert hierbei eine Fläche, während sein Nachbarpixel schon den Hintergrund anzeigt. Bei diagonalen Kanten kann dieser Effekt deutlich durch Bildung einer unerwünschten Pixelstufe zu sehen sein. Aliase bezeichnen hierbei diese Pixelstufen oder sonstige pixelige Artefakte.

Eine gängige Variante, diese Aliase zu minimieren ist das Oversampling. Hierbei wird der Raytracing-Algorithmus dahingehend abgeändert, dass für einen Bildpixel mehr als nur ein Sehstrahl in die Szene geschickt wird. Bei einem 2x2-Oversampling wird zum Beispiel ein Pixel aus den Werten von vier Sehstrahlen zusammengesetzt und bekommt dessen durchschnittlichen Wert zugewiesen. Streng genommen kann man ein 2x2-Oversampling mit der Verdopplung der Render-Auflösung und dem anschließenden Verrechnen auf die Ursprungsgröße ansehen. Ein Anstieg der Renderzeit ist die logische Folge, die sich bei noch größeren Sample-Größen weiter erhöht. Das Adaptive Over-Sampling ist eine verbesserte Variante, bei der an Flächenkanten, d.h. an Bereichen mit erhöhtem Kontrast, ein Over-Sampling mit größeren Samples eingesetzt wird. An Stellen ohne Kanten wird entsprechend ein minimales bis kein Over-Sampling eingesetzt. Schnellere und genauere Ergebnisse sind die Folge.

Eine Verwendung von Antialiasing ist bei Pre-rendered Bildern trotz deutlicher Erhöhung der Renderzeiten Standard, bei Echtzeit-Berechnungen anzutreffen, aber nicht Standard.

## 6.2 Licht

Licht ist nicht nur ein Helligkeitsfaktor, der zum Erhellen einer Szene beiträgt. Mit dem richtigen Licht kann man entscheidend dazu beitragen, ein erstelltes Bild für den Betrachter realistisch wirken zu lassen oder nicht:

»Die stimmige Ausleuchtung einer Szene entscheidet maßgeblich darüber, ob eine Szene realistisch wirkt oder in den gefürchteten 3D-Look verfällt.« [Asa]

Objekte haben in der realen Welt je nach Positionierung zur Lichtquelle eine helle und eine dunkle Seite. Je stärker dieser Effekt vorhanden ist, desto einfacher kann der Betrachter es als plastisches Objekt wahrnehmen. Sind die Kontraste dagegen weniger stark ausgeprägt, dann wirkt das Objekt flach und wird schwerer als dreidimensionales Objekt angenommen. Mit einer guten Beleuchtung kann man die Qualität eines Bildes maßgeblich verbessern, indem der Betrachter es einfacher hat das Bild zu lesen und es zu verstehen. Dadurch wird die Glaubwürdigkeit des Bildes erhöht und es wirkt somit realistischer. Die Art der Lichtquellen, deren Anzahl und deren Positionierung sollten dementsprechend mit Bedacht gewählt werden.

### 6.2.1 Standard-Lichttypen

Bei den meisten 3D-Programmen gibt es mehrere Möglichkeiten, Lichter zu generieren. Die unterschiedlichen Lichttypen, die einem zur Verwendung stehen, bilden die unterschiedlichen Lichtquellen nach, wie sie in der realen Welt vorkommen können. Hierbei hat jeder Lichttyp seine eigenen Vor- und Nachteile. Die sechs gebräuchlichsten Lichttypen sind folgende:

- **Punktlichter** oder auch Omnidirektionale Lichter gehören zu den primitivsten Lichtern. Sie entsprechen der Funktionsweise einer kleinen Sonne. Sie geben in alle Richtungen gleichmäßig ihr Licht ab und sind einfach zu handhaben. Punktlichter besitzen keine Größe, sie sind unendlich klein.
- **Spotlichter** strahlen das Licht nur in eine bestimmte Richtung ab. Der sich dabei bildende Lichtkegel kann kreisförmig aber auch viereckig sein. Der Winkel der Lichtöffnung ist dabei meist einstellbar, genau so ob der Lichtkegel hart oder weich begrenzt ist. Der Vorteil der Spotlichter ist, dass man mit ihnen gezielt arbeiten kann. Mit ihnen ist es möglich, nur bestimmte Stellen zu beleuchten und damit hervorzuheben. Ein weiterer Vorteil ist dessen Speicherverbrauch. Da sie nur in eine Richtung Licht werfen, verbrauchen sie nur ein Sechstel an Speicher, welches Punktlichter benötigen würden. Wie Punktlichter besitzen sie auch keine Ausdehnung.
- **Distanzlichter** simulieren direktionales Licht, welches aus einer unendlichen Distanz sein Licht abgibt. Es ist nicht entscheidend, aus welcher Entfernung das Licht kommt, sondern

aus welcher Richtung. Alle Objekte werden aus dem gleichen Winkel beleuchtet. Es eignet sich sehr gut, um Sonnenlicht zu simulieren.

- **Parallellichter** sind ähnlich den Distanzlichtern. Sie haben im Gegensatz zu den Distanzlichtern allerdings einen Ursprung.
- **Röhrenlichter** sind ähnlich den Punktlichtern mit der zusätzlichen Eigenschaft ausgestattet, dass sie nicht auf einen Punkt beschränkt sind, sondern eine Linie bilden können. Sie geben ähnlich wie Neonröhren das Licht ab. Da sie technisch aus mehreren Punktlichtern gebildet werden, ist der Speicherverbrauch dementsprechend höher und nur in wenigen Fällen empfehlenswert.
- **Flächenlichter** sind die Erweiterung der Röhrenlichter um eine weitere Dimension. Während Röhrenlichter eine Lichtlinie nachbilden, bilden die Flächenlichter eine Fläche oder weitere geometrische Formen nach. Vorteil ist, dass dieses Licht im Gegensatz zu den Punktlichtern nun eine definierte Ausdehnung hat und der Realität am nächsten kommt. „Die Qualität der mit einem Flächenlicht generierten Beleuchtung und der Schatten macht es zu einer guten Wahl, wenn es um realistische Bildberechnung geht.“ [BIR] Nachteil ist der Speicherverbrauch und dementsprechend die erhöhte Renderzeit, da Flächenlichter wie Röhrenlichter intern durch mehrere Punktlichter repräsentiert werden.

Aufgrund der Einfachheit finden die omnidirektionalen Punktlichter bei der Ausleuchtung von virtuellen Charakteren oftmals Verwendung, wie meine eigene Erfahrung bei mehreren Produktionen zeigt. Durch den niedrigen Speicherverbrauch ist die Verwendung von Spotlichtern nach Möglichkeit allerdings vorzuziehen. Auch im Hinblick auf die Verwendung von Shadow-Maps arbeiten die Spotlichter effizienter.

### 6.2.2 Lichtpositionierung

Die Positionierung der Lichtquellen ist entscheidend, um einen virtuellen Charakter „in das rechte Licht zu rücken“. Eine einfache und gängige Variante, einen Charakter optimal zu beleuchten, ist die sogenannte *Drei-Punkt-Beleuchtung*. Bei 3D-Produktionen kommt sie jeweils auf die entsprechende Situation abgestimmt oft zum Einsatz. Sie ist den klassischen Hollywood-Produktionen entlehnt [Bir] und kann als grundsätzliche Richtschnur Verwendung finden [Asa].

Hierbei werden die Szenen hauptsächlich durch drei verschiedene Lichtquellen beleuchtet (vgl. Abbildungen 6-3 und 6-4).

- **Hauptlicht** oder auch Key-Light ist, wie der Name schon sagt, das wichtigste Licht. Mit diesem Licht wird der Hauptteil der Szene ausgeleuchtet. Dieses Licht befindet sich traditionell leicht seitlich erhöht vor der Szene in einem Bereich von 30 bis 60 Grad. Hierdurch wird das Zielobjekt nicht zu frontal beleuchtet und wirkt nicht zu platt. Andererseits bekommt es noch genug Licht und verschwindet nicht in seinem eigenen

Schattenbereich. Der Charakter wirkt bei dieser Lichtpositionierung sehr plastisch und wird als 3-dimensionales Objekt wahrgenommen. Das Key-Licht sollte weißes bis helles gelbes Licht werfen, um sonnenartiges Licht zu simulieren. Bedingt durch die höchste Helligkeit, sollte dieses Licht auch den dunkelsten Schatten hervorrufen.

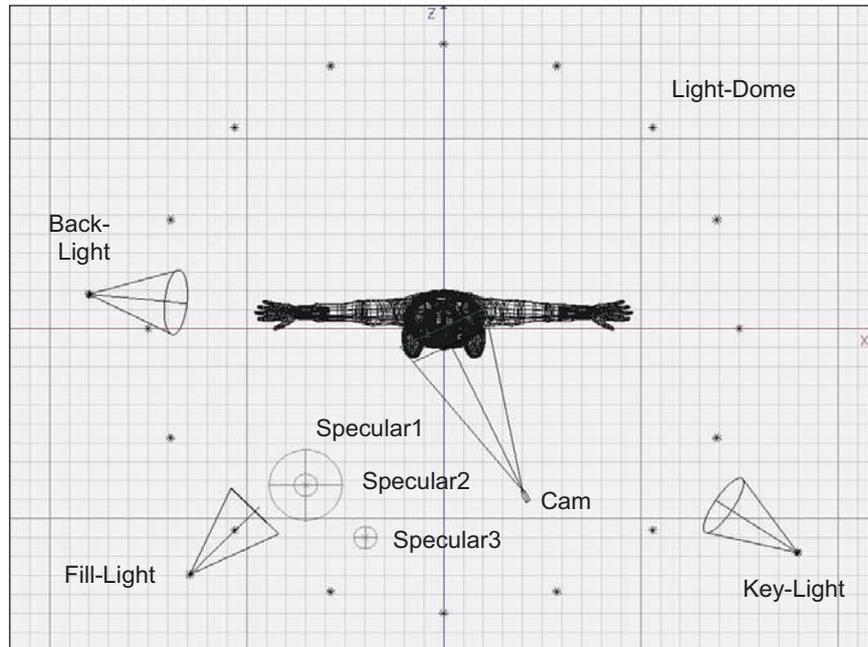
- Das **Fülllicht** oder auch Fill-Light unterstützt das Hauptlicht. Es beleuchtet die Stellen, die vom Hauptlicht nicht genug ausgeleuchtet werden. Dabei besitzt dieses Licht nur eine sehr geringe Leuchtkraft, damit das Zielobjekt seine Tiefe nicht verliert. Fülllichter werden dazu benutzt, um indirektes Licht zu imitieren, welches durch diffuse Reflexionen das gewünschte Objekt weiter aufhellen würde. Dementsprechend sollten Fülllichter auch tiefer als Hauptlichter positioniert werden, da indirektes Licht mehr von unten und seitlich abgestrahlt wird. Zusätzlich nehmen Fülllichter bis zu einem bestimmten Maß die Farbe der reflektierenden Objekte an. Bei dem erstellten Charakter wird ein warmer Rot- bis Orangeton für das Fülllicht verwendet.
- Das **Streiflicht**, Back-Light oder auch Rim-Light gibt dem Objekt sein Profil. Es wird leicht seitlich hinter dem Zielobjekt positioniert und leuchtet mit voller Stärke. Dadurch kann es das Zielobjekt gut vom Hintergrund abheben, indem es seine Kanten verstärkt und dessen Silhouette besser zum Vorschein bringt. Farblich sollte sich das Licht komplementär zum Fülllicht verhalten. Bei einem Fülllicht mit warmen Orangeton sollte das Back-Light ein kaltes blaues Licht aussenden.



**Abbildung 6-3: Angewendete Lichter:** Back-Light, Fill-Light, Key-Light und alle Lichter zusammen

Je nach Szene können weitere Modifikationen angewendet werden. Für den erstellten Charakter wurde nicht nur ein Fill-Light verwendet, sondern zusätzlich ein *Light-Dome*. Ein Light-Dome ist eine Anordnung von mehreren Fill-Lights kreisförmig um das Zielobjekt herum. 16 auf das Zielobjekt ausgerichtete Lichtquellen mit sehr geringer Intensität reichen oft aus, um ein indirektes Licht gut zu simulieren. Mehr Lichter bringen meist kaum erkennbare Verbesserung bei erhöhtem Speicherbedarf. Wichtig bei einem Light-Dome ist die unbedingte Deaktivierung der Schattenberechnung dieser Lichtquellen, da sonst die Renderzeit in astronomische Höhen schnellen könnte.

Glanzlichter oder *Specular-Lights* bilden eine weitere Art der Lichter. Sie besitzen keine eigene Beleuchtung und erhellen somit nicht die Szene. Sie dienen ausschließlich dem Setzen zusätzlicher Highlights. Bei nassen oder spiegelnden Materialien sind Highlights, also Spiegelungen der Lichter zu erkennen. Da Lichter in 3D-Programmen aber meist unendlich klein sind, werden sie unabhängig von der Lichtquelle geregelt. Highlights werden zum Beispiel zusätzlich für die Spiegelungen in den Augen gesetzt, da die hauptsächlichen Lichter sich dort meist nicht spiegeln. Mit dem Setzen von zusätzlichen Highlights kann man die Materialeigenschaften und Form von bestimmten Objekten besser verdeutlichen und damit realistischer gestalten.



**Abbildung 6-4: Lichtpositionierungen:** Positionen des Hauptlichtes (Key-Light), des Fülllichts (Fill-Light) und des Back-Lights, sowie der 16 Lichter des Light-Domes und der drei Glanzlichter (Specular).

Um die Liste der Lichter zu vervollständigen sollten noch die *Bounce-Lights* erwähnt werden. Sie sind den Fill-Lights sehr ähnlich, simulieren diffuses Licht und können auch als indirektes Licht Verwendung finden. Entscheidender Unterschied ist deren Lichtintensität, die dazu ausreicht, dass Bounce-Lights Szenen komplett erhellen können.

### 6.2.3 Globale Illumination

Standardmäßig benutzen die meisten aktuellen Renderprogramme ein lokales Beleuchtungsmodell um die Lichtquellen zu berechnen. Hierbei werden nur die Flächen beleuchtet dargestellt, die auch direkt von einer gesetzten Lichtquelle angestrahlt werden. Flächen, die nicht direkt von Lichtquellen angestrahlt werden, bleiben hierbei tiefschwarz. Dies entspricht allerdings nicht dem Lichtverhalten in der realen Welt.

In der Realität werden Lichtstrahlen von jedem Objekt teilweise absorbiert. Der Teil, der nicht

absorbiert wird, wird reflektiert und je nach Spektrum von unseren Augen als Farbe interpretiert. Alle Objekte, die sichtbar sind, reflektieren also auch einen Teil des Lichtes, welches wiederum andere Objekte in der Szene beleuchten kann. Die Simulation dieses indirekten Lichtes wird als *globale Illumination* (kurz GI) bezeichnet und entspricht der natürlichen Beleuchtung mit seinen diffusen Lichtreflexionen sehr viel eher als standardmäßige Raytracing-Beleuchtungsmodelle. Kennzeichnend für Bilder, die mit globaler Illumination berechnet wurden, ist der geringere Schwarzanteil, da nun die meisten Flächen sich gegenseitig beleuchten. Hierdurch werden auch keine Fülllichter mehr gebraucht, die genau das indirekte Licht simulieren sollen. Anzumerken sei, dass indirektes Licht immer die Farbe weiterschickt, die das reflektierende Objekt besitzt.

Technisch gesehen ähnelt die Funktionsweise von GI der Berechnung eines Raytracers. Hierfür werden ausgehend von der gesetzten virtuellen Kamera Sichtstrahlen in die Szene geschickt. Schneidet ein solcher Sehstrahl ein Objekt, so werden ausgehend von diesem Schnittpunkt (*Shading-Punkt*) weitere Sehstrahlen generiert, die nun kugelförmig in alle Richtungen weitergeschickt werden. Die Helligkeit und die Farbe dieses Punktes werden aus den Informationen gebildet, die die weitergesendeten Sehstrahlen, die sogenannten *Stochastischen Strahlen*, generieren. Mit jeder weiteren Reflektionstiefe nimmt die Auswirkung des entsprechenden Strahls ab und wird weniger in der Berechnung des entsprechenden Bildpunktes mit einbezogen. Wiederum ist eine Abruchtiefe zu definieren, da dieser Vorgang sich immer wieder rekursiv wiederholt.

Aufgrund der vielen Strahlberechnungen von GI ist diese Berechnungsart extrem rechenintensiv und langsam. Bei sehr komplexen Szenen mit vielen detaillierten Objekten liefert GI sehr realistische Ergebnisse, die nur schwer mit anderen Mitteln zu realisieren wären. Bei Architekturvisualisierungen ist der Einsatz von GI zum Beispiel vorteilhaft. Bei der Beleuchtung von virtuellen Charakteren, die nicht mit einer komplexen Szene interagieren, gibt es rechenärmere Methoden, ein realistisches Beleuchtungsmodell zu erstellen. Der Einsatz von Fülllichtern bringt in Hinsicht Renderzeit und Ergebnis effizientere Ergebnisse. Auch ein Einsatz von Ambient Occlusion<sup>27</sup> kann als Immitation von GI in Betracht gezogen werden.

#### 6.2.4 HDRI

Die Steigerung zur GI für das Erzeugen absolut realistischer Bilder ist das HDRI-Rendering [Eck]. HDRI steht hierbei für *High Dynamic Range Images* und bezeichnet die Verwendung dieser HDR-Bilder zur Ausleuchtung der Szene. Diese Bilder werden dabei als Textur einer der Szene umgebenen Sphäre zugewiesen und das GI aktiviert. Aufgrund eines höheren Helligkeitsumfanges der HDRI als normaler Bilder wird die Szene allein davon optimal ausgeleuchtet. Einfache Lichter werden nicht benötigt, können aber dennoch unterstützend integriert werden.

#### 6.2.5 Anwendungen in der Industrie

Viele Animationsfilme benutzten in der Vergangenheit ein einfaches Beleuchtungsmodell mit der Positionierung einfacher Lichter ganz im Stil der alten Hollywoodfilme. Als gutes Beispiel kann hier

---

<sup>27</sup> vgl. Kapitel 6.3.5 Ambient Occlusion

Shrek (2001, PDI / Dreamworks) genannt werden. Drei Jahre nach dem ersten Teil dieser Filmreihe kam die Fortsetzung in die Kinos. Mit steigender Rechenleistung, die für den zweiten Teil zur Verfügung stand, kam neben SSS ein neues Beleuchtungsmodell zum Einsatz. Neben dem Einsatz von nur einem Bounce-Light um Renderzeit zu sparen, wurde wie in vielen späteren Filmen der Einsatz von GI zum Standard. Als Beispiel hierfür kann der zweite Teil der Fluch der Karibik-Reihe *Pirates of the Caribbean: Dead Man's Chest* (2006, Walt Disney Pictures / Jerry Bruckheimer Films) benannt werden, bei dem GI bei allen computergenerierten Charakteren zum Einsatz kam [Rob6]. Einer der ersten CG-Filmen, die GI benutzten war bereits *Final Fantasy: The Spirits Within* (2001, Square Pictures) in einer Zeit, wo kaum andere Filme diesen Effekt verwendeten [Sto].

Auch die Verwendung von HDRI gehört neben GI in der Filmbranche trotz dessen benötigter hoher Rechenleistung zum Standard. Die HDRI-Beleuchtung kam zum Beispiel bei dem Film *Hui Buh - Das Schlossgespenst* (2006, Rat Pack / Constantin Film) zum Einsatz. Nach dem Dreh standen für über 200 Lichtsituationen Fotos zur Verfügung, die zu einem HDR-Environment zusammengefügt wurden und wodurch die computergenerierten Charaktere beleuchtet und berechnet werden konnten [Sch].

### 6.2.6 Fazit

Bei der Beleuchtung einer menschlichen Figur ist die Verwendung von Omnidirektionalen Lichtern und Spotlichtern die beste Wahl, wobei Spotlichter nach Möglichkeit zu bevorzugen sind. Als beste Positionierung der Lichter gilt die klassische Drei-Punkt-Beleuchtung, die heutzutage immer noch aktuell ist und aufgrund des Kosten-/Nutzenfaktors am effektivsten arbeitet.

Die GI führt in umfangreichen Szenen zu einer besseren Gesamtausleuchtung, verlangt jedoch auch deutlich mehr Rechenleistung. Die Erweiterung von GI bildet die Verwendung von HDR-Bildern. Diese liefert ohne großen Aufwand nicht nur die besten und realistischste Beleuchtungen, sondern auch die größten Renderzeiten mit hinzu. Man braucht hierfür also viel Rechenpower. Mit etwas längerer Arbeit können aber meist Szenen mit Standardlichtern und dadurch schnelleren Renderzeiten annähernd gute Ergebnisse liefern.

## 6.3 Schatten

### 6.3.1 Vorbetrachtungen

Überall wo Licht ist kommen in der Realität auch Schatten vor. Der Mensch sieht tagtäglich Schatten und ist daran gewöhnt. Eine Szene ohne Schatten sollte für den Menschen also unrealistischer erscheinen als eine mit Schatten.

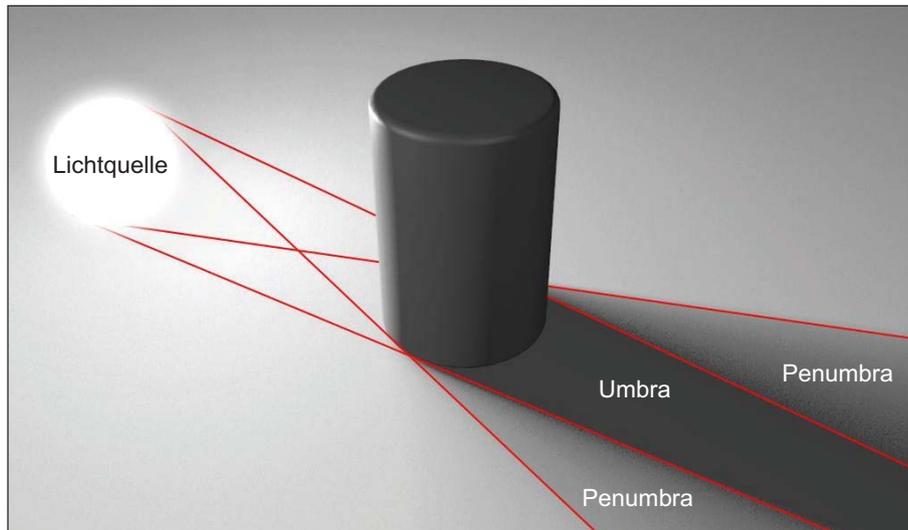
Der Schatten eines Objektes bezeichnet laut Definition den „von einer Lichtquelle nicht oder teilweise bestrahlten Raum hinter einem beleuchteten lichtundurchlässigen Körper.“ [BI] Dieser unbeleuchtete Raum lässt sich unterteilen in

- *Unbeleuchteter Raum* zwischen beleuchtetem Objekt und dahinter liegendem Objekt oder Fläche. Dieser Schattenbereich ist meist nur bei schlechten Witterungsverhältnissen, wie zum Beispiel Nebel, zu erkennen.
- *Projiziertes Umrissbild* des beleuchteten Objektes auf das dahinter liegende Objekt oder Fläche. Diese Fläche wird umgangssprachlich als Schatten bezeichnet.

Schattenbereiche, in dessen Raum kein Lichtstrahl eindringt, bezeichnet man als *Umbra* oder *Kernschatten*. Aufgrund der Eigenschaft des Lichtes, dass es sich nicht parallel, sondern in alle Richtungen gleichförmig ausbreitet und der Tatsache, dass Lichtquellen eine gewisse Ausdehnung haben und nicht unendlich klein sind, gibt es einen Übergang zwischen Licht und Schatten. Zusätzlich können die Lichtstrahlen an den Objektgrenzen gebeugt werden, wodurch es auch bei idealen punktförmigen Lichtquellen zu Schattenübergängen kommt [Mey]. Diese Bereiche, die gleichzeitig im Schatten liegen und von der Lichtquelle beleuchtet werden, nennt man *Penumbra* oder *Halbschatten* (vgl. Abbildung 6-5). Die Sonne kann aufgrund ihrer großen Entfernung als Flächenlicht interpretiert werden. Dennoch ist auch ihr Schatten nicht vollständig scharf [Bir]. Ein Schattenübergangsbereich der Penumbra ist in der Realität mehr oder minder überall anzutreffen. Scharfe Schattenkanten hingegen sind folglich als unnatürlich und unrealistisch einzustufen.

Schatten übernehmen eine weitere wichtige Aufgabe. Sie helfen dabei, die räumliche Zuordnung von Objekten zu erkennen. Ein Objekt, das zum Beispiel auf dem Boden liegt, wirft einen anderen Schatten als ein Objekt, das einige Zentimeter über dem Boden schwebt. Meist hilft erst ein Schatten dabei, diese räumliche Positionierung zu erkennen.

Ein weiterer Aspekt ist, dass Schatten Zusatzinformationen über die zu betrachteten Gegenstände liefern können. Ein Objekt, welches zum Beispiel aus der Sicht der Kamera verborgen ist, kann durch eine seitliche Lichteinwirkung einen sichtbaren Schatten werfen und damit dessen Vorhandensein anzeigen. Das Objekt braucht nicht einmal vor der Kamera zu liegen, ein Schattenwurf kann aber im sichtbaren Bereich liegen und Information über Objekte außerhalb des sichtbaren Bereichs geben.



**Abbildung 6-5: Schattenübergang:** Durch die Ausdehnung einer Lichtquelle entsteht neben dem Kernschatten (Umbra) ein Halbschatten (Penumbra) im Schattenübergangsbereich.

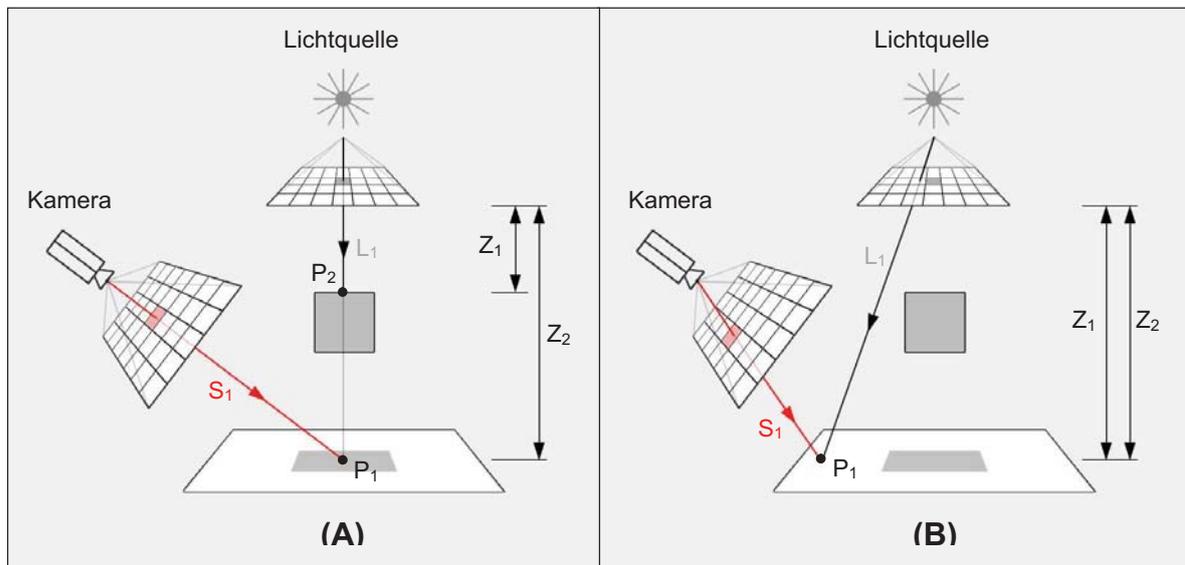
### 6.3.2 Shadow-Maps

Die Idee der *Shadow-Maps* oder Schattenmaps liegt darin, von der Lichtquelle aus zu schauen, ob die zu rendernden Pixel zu sehen sind oder nicht. Sind sie von der Lichtquelle aus zu sehen, werden sie beim finalen Rendering als beleuchtet gerendert. Andernfalls müssen sie im Schatten liegen und werden dementsprechend schattiert gerendert. Für die Umsetzung der Shadow-Maps erfolgt das Rendern der gesamten Szene dementsprechend in mehreren Schritten.

1. **Erstellung der Shadow-Maps:** Vor dem eigentlichen Rendern wird für jede Lichtquelle aus die Szene gerendert. Dabei wird nicht das eigentliche Bild erstellt, sondern es wird eine Tiefenmap (Depth Map) erstellt. Hierfür wird für jeden ausgesendeten Schattenstrahl  $L$  die Entfernung  $Z_1$  von Lichtquelle zum nächsten Objekt gemessen und als Wert in der Schattenmap abgespeichert (vgl. Abbildung 6-6).
2. **Abgleich beim Rendern:** Für jeden zu berechnenden Bildpixel wird nun die Entfernung  $Z_2$  zwischen Bildpunkt  $P_1$  zur Lichtquelle berechnet und mit der möglichen Tiefe der Schattenmap  $Z_1$  verglichen. Ist die Entfernung gleich der Entfernung in der Schattenmap ( $Z_1=Z_2$ ) wird dieser Bildpunkt von der Lichtquelle beleuchtet. Ist der Abstand zwischen Lichtquelle und Bildpunkt größer als der entsprechende Wert der Tiefenmap ( $Z_1 < Z_2$ ), liegt ein Objekt zwischen Lichtquelle und Bildpunkt und demzufolge muss er im Schatten liegen.
3. **Farbzuweisung:** Die Pixel werden entsprechend der Position im Licht oder im Schatten hell oder dunkel gerendert. Der Renderer muss nicht mehr alle Objekte überprüfen, ob sie von den Lichtstrahlen eines Lichts getroffen werden oder von einem Objekt verdeckt

werden. Anhand der Tiefenmap kann der Renderer diesen Vorgang schon vorher stoppen und arbeitet dadurch schneller.

Vorteilhaft bei den Schattenmaps sind deren recht einfacher Algorithmus und die daraus resultierende Geschwindigkeit.



**Abbildung 6-6: Shadow-Maps:**

(A) Die Entfernung  $Z_2$  zwischen Lichtquelle und Bildpunkt  $P_1$  ist größer als in der Schattenmap ( $Z_1$ ) abgespeichert.  $P_1$  liegt im Schatten.

(B) Die Entfernung  $Z_2$  ist gleich groß wie  $Z_1$ .  $P_1$  wird von der Lichtquelle angeleuchtet.

Maßgeblich für die Qualität der Schatten, aber auch für den benötigten Speicher und der daraus resultierenden Berechnungsdauer, ist hierbei wiederum die Auflösung der Schattenmaps. Kleinere Schattenmaps können schneller berechnet werden, können aber auch ungenau sein. Höhere Schattenmaps führen zu genaueren Schatten, können aber auch zu drastischen Speicher- und damit Renderzeiterhöhungen führen. Für die Tiefeninformationen verwenden viele Renderprogramme Fließkommazahlen, die 4 Bytes beanspruchen. Bei einer kleinen Schattenmap mit einer 256er Auflösung ergibt dies ein Speicherverbrauch von 262 KByte<sup>28</sup>. Bei größeren Maps mit zum Beispiel 4096er Auflösung werden schon mehr als 67 MByte an Speicher verbraucht.

Damit man die zu geringe Auflösung der Schattenmaps beim fertigen Bild nicht als pixelige Treppen wahrnimmt, werden sie vom Renderer oft direkt weichgezeichnet, wodurch Schattenmaps auch als weiche Schatten bezeichnet werden. Mit der Auflösung der Schattenmap steuert man dann letztendlich die Weichheit der Schatten. Geringere Auflösungen ergeben sehr weiche Schatten,

<sup>28</sup>  $256^2 * 4 \text{ Byte} = 262 \text{ KByte}$

während höhere Auflösungen weniger weiche Kanten abbilden. Prinzipbedingt kommt hier der erste große Nachteil zum tragen: Steht ein Objekt auf einer Fläche, auf dessen sich ein Schatten mittels Schattenmap abbilden soll, so kann es passieren, dass bei zu weichen Schatten sich Abstufungen bilden, an denen sich eigentlich kein Schatten bilden sollte. Oft kann man diesen Fehler mit einer einfachen Verschiebung des Schattens berichtigen.

Um die Berechnung mit Schattenmaps weiter zu optimieren, sind Lichtspots die beste Wahl. Zusätzlich sollte der Lichtkegel auf ein Minimum reduziert werden, damit die Schattenmaps die höchste Auflösung erhalten. Im Gegensatz zu Lichtspots müssen bei Punktlichtern mehrere Schattenmaps für alle Richtungen berechnet werden.

### 6.3.3 Raytraced Shadows

Die *Raytraced Shadows* oder Raytracing-Schatten stellen eine Erweiterung des Raytracing-Algorithmus<sup>29</sup> dar. Für jeden zu berechnenden Bildpunkt wird bei dieser Schattenberechnung überprüft, ob sich ein Objekt zwischen Bildpunkt und Lichtquelle befindet. Hierfür wird ein Schattenstrahl von dem zu berechnenden Bildpunkt P in Richtung verfügbare Lichtquellen geschickt (vgl. Abb. 6-7). Wird auf dem Weg ein Schnittpunkt gefunden, so existiert ein Objekt. In diesem Fall liegt der Punkt im Schatten und muss mit der Schattenfarbe der getesteten Lichtquelle eingefärbt werden. Ist dies nicht der Fall, erhält der Punkt seine vorberechnete Helligkeit der Lichtquelle zugewiesen. Ist ein darüber hinaus ein verdeckendes Objekt transparent, so wird die Helligkeit der entsprechenden Lichtquelle nicht auf null reduziert, sondern die Schattenfarbe entsprechend der Transparenz prozentuell angepasst. Im Gegensatz zu Schattenmaps können Raytracing-Schatten transparenter Objekte sehr leicht berechnen.

Prinzipbedingt sind bei Raytraced Shadows keine Vorberechnungen mehr nötig. Allerdings müssen sehr viele Strahlen berechnet werden und die gesamte Szene wird in den Speicher geladen, da der Raytracingalgorithmus zu jeder Zeit auf alle Daten Zugriff haben muss. Dadurch ist diese Schattenart meist speicherlastiger und letztendlich langsamer als bei Shadow Maps. Ein weiterer Nachteil ist die zuvor angesprochene Berechnungstiefe<sup>30</sup> bei den Raytracing-Algorithmen. Ist dieser Wert zu gering eingestellt, kann es vorkommen, dass ein Objekt gar kein Schatten wirft.

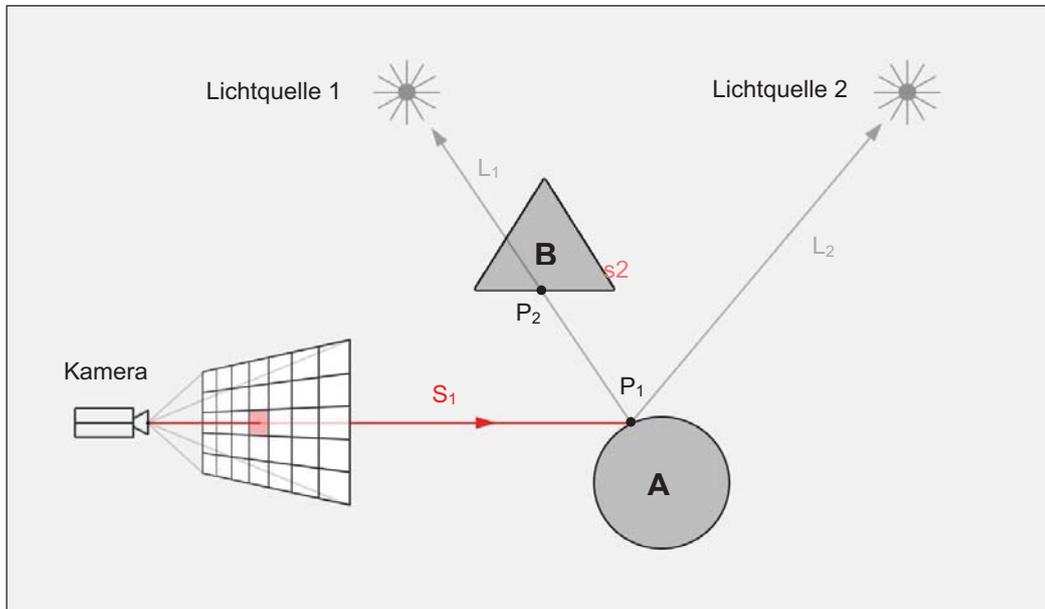
Standardmäßig besitzen Raytracing-Schatten aufgrund ihrer einzelnen Strahlenberechnung immer harte Schattenkanten, wodurch sie oft auch als harte Schatten bezeichnet werden. Diese harten Schatten sehen aber künstlich aus und sind bei der Erstellung von realistischen 3D-Bildern nur selten zu benutzen.

»Raytracing-Schatten [...] sehen unnatürlich scharf aus, ganz anders als jeder natürlich vorkommende Schatten.« [Bir]

---

<sup>29</sup> vgl. Kapitel 6.1.1 Raytracing

<sup>30</sup> vgl. maximale Strahlentiefe in Kapitel 6.1.1 Raytracing



**Abbildung 6-7: Raytraced Shadows:** Der zu berechnende Schnittpunkt  $P_1$  wird auf zwei Lichtquellen hin untersucht. Lichtquelle 1 wirft einen Schatten auf  $P_1$ , da ein Schnittpunkt  $P_2$  existiert und auf das Objekt B hinweist. Lichtquelle 2 wirft keinen Schatten.

### 6.3.4 Area Shadows

Bei den Raytraced Shadows werden standardmäßig Lichtquellen mit unendlich kleiner Ausdehnung zur Berechnung herangezogen, wodurch die Schatten prinzipbedingt harte Kanten werfen. Bei den weichen Schatten der Shadow-Maps hingegen sind die Kanten immer gleich weich. Beide Varianten entsprechen nicht den in der Realität vorkommenden Schattenverläufen. Echte Schatten werden schärfer, je näher sich ein Objekt an einem anderen befindet. Je weiter die Objekte voneinander entfernt sind, desto weicher werden auch deren Schattenübergänge. Die *Area-Shadows* oder auch Flächen-Schatten ahmen diesen Effekt am realistischsten nach. Dies wird erreicht, indem Raytraced Shadows mit einem virtuellen Flächenlicht berechnet werden, egal welche Lichtart eingestellt wurde. Das Flächenlicht wird wiederum durch mehrere Lichtquellen zusammengesetzt, wodurch letztendlich die Lichtquelle eine reale Ausdehnung erhält und es somit zu Licht-Schatten-Übergängen kommen kann.

Diese Schattenart ist sehr rechenintensiv und die daraus folgenden Renderzeiten sind meist zu hoch, um diese Schattenart standardmäßig zu verwenden.

### 6.3.5 Ambient Occlusion

Globale Illumination<sup>31</sup> ist ein Verfahren, das das indirekte Licht simuliert und damit das Beleuchtungsmodell realistischer macht. Allerdings ist GI eine äußerst rechenintensive Methode, um

<sup>31</sup> vgl. Kapitel 6.2.3 Globale Illumination

realistisches Licht nachzuahmen. *Ambient Occlusion (AO)* kann als Umgebungsverdeckung übersetzt werden, dem die gleiche Idee zu Grunde liegt. Während bei der GI Licht reflektiert wird und dabei andere Flächen der Szene aufhellt geht AO den entgegengesetzten Weg und dunkelt Bereiche ab, die durch indirektes Licht auch nicht aufgehellt worden wären. Dies sind Bereiche, die durch andere Objekte oder durch die eigene Geometrie verdeckt werden und vom indirekten Licht zu gering bis gar nicht erreicht werden. Das Ergebnis der Beleuchtungsverteilung beider Verfahren ist folglich ähnlich und entspricht den natürlichen diffusen Lichtreflexionen in der Umwelt.



**Abbildung 6-8:** **Ambient Occlusion:** AO-Pass (links), Szenen ohne aktivierte Schatten (Mitte) und Charakter mit aktiviertem AO (rechts).

Technisch gesehen wird dies erreicht, indem der gerade zu rendernde Bildpunkt die unmittelbare Umgebung in alle Richtungen abtastet. Dabei werden ausgehend von diesem Punkt AO-Strahlen in alle Richtungen losgeschickt und die Anzahl und Entfernung aller möglichen Schnittpunkt ausgewertet. Je mehr Strahlen auf andere Flächen treffen und je kürzer die ausgewerteten Entfernungen sind, desto dunkler ist die endgültige AO-Helligkeit. Um die Berechnungsgeschwindigkeit einzugrenzen, wird die maximale Strahltiefe und die Menge an Strahlen begrenzt.

Durch den Einsatz von AO wirken Objekte weniger flach, da die Struktur durch die Helligkeitsunterschiede kontrastreicher hervorgehoben wird. Dadurch können menschliche Charaktere plastischer und dadurch realistischer wirken (vgl. Abbildung 6-8). Allerdings sollte AO in Maßen eingesetzt werden, sonst sieht es schnell verdreckt und wiederum weniger realistisch aus. Generell kann AO als Ergänzung oder aufgrund der viel schnelleren Berechnung als Ersatz zu GI betrachtet werden. Verglichen mit sehr weichen Schatten sieht AO sehr ähnlich aus und kann entsprechend auch hier als Ersatz oder als Ergänzung angesehen werden. In der Regel benötigt AO auch hier weniger Renderzeit [Bir].

### 6.3.6 Prerendered Shadows

Eine einfache und sehr effektive Art einen Schatten zu generieren, ist die Verwendung von *Prerendered Shadows*. Dabei werden die Schatten unabhängig von ihrer Art vor dem eigentlichen

Rendern berechnet und in eine Textur „gebacken“. Hierbei entsteht ein Bild der Schatten, welches schon fertig in der Textur vorliegt und beim endgültigen Rendern nicht mehr berechnet werden muss. Man spart die Renderzeit und den Speicherbedarf der Schattenberechnung ein.

Das Ambient Occlusion des erstellten Charakters wurde so zum Beispiel gebacken und in die Texturen des Farb-Kanals, des Spiegel-Kanals und des Relief-Kanals integriert, wodurch die Wirkung nicht nur schneller und arbeitsspeicherschonender berechnet werden konnte, sondern auch zur besseren Steuerung der Spiegelungen und des Reliefs benutzt wurde<sup>32</sup>.

Bei Animationen kommen bei dieser Art der Schattenberechnung allerdings Probleme auf. Wenn sich eine Figur bewegt, dann ändern sich auch die Schattenverläufe. Die gebackenen Schatten würden nicht mehr passen. Und auch bei dynamischen Lichtern kann diese Technik nicht immer angewendet werden.

### 6.3.7 Anwendungen in der Industrie

In der Filmbranche sind die einfachen Shadow Maps die gebräuchlichsten Schatten, was aufgrund der guten Ergebnisse bei den geringsten Renderzeiten keine Verwunderung ist. Und auch die bekannte Filmfabrik Pixar gebraucht in seinen Filmen die Shadow-Maps. Dessen hauseigene sehr populäre Rendersoftware Renderman arbeitete anfänglich nicht als Raytracer, wurde aber inzwischen um dessen Eigenschaften erweitert [Cri].

»Schatten-Maps sind derzeit in der professionellen Beleuchtung für Filmproduktionen die populärsten Schattenart.« [Bir]

Und auch die Verwendung von Ambient Occlusion hat in den Hollywood-Produktionen Einzug erhalten. Trotz der Verwendung umfangreicher Beleuchtungsmodelle wird AO unterstützend eingesetzt. Als Beispiel hierfür wäre wieder *Pirates of the Caribbean: Dead Man's Chest* (2006, Walt Disney Pictures / Jerry Bruckheimer Films) zu nennen [Rob6].

Dass die Schattenberechnung ein komplexes Thema ist, kann man am besten an der geschichtlichen Verwendung in Echtzeitrenderern - den Computerspielen - sehen: Das erste Spiel, welches 3D-Grafik verwendete und Schatten zeigte war *Alone in the Dark* (1992, Infogrames). Aufgrund der damals nicht vorhandenen Rechenleistung waren nur wenige Objekte als 3D dargestellt und Schatten wurden nur bei unbeweglichen Objekten in den Hintergrund gemalt. Den ersten Schatten, den ein 3D-Charakter erhielt, war die Spielfigur *Lara Croft* in dem Spiel *Tomb Raider* (1996, Eidos Interactive) in Form eines Blob Shadow. Als Blob Shadow wird ein anfänglich kreisrunder Schatten genannt, der einfach als statischer Schatten per Textur unter eine Figur gelegt wird. Und auch die Umgebung in *Lara Croft* erhielt Schatten, allerdings nur als Bestandteil von Texturen. Bei den ersten echten 3D-Engines wie der Quake Engine (ab 1996, id Software) wurden anfänglich auch nur Blob Shadows für die Charaktere verwendet. Mittlerweile werden bei den meisten

---

<sup>32</sup> vgl. Kapitel 4 Texturen

Computerspielen, wie in der Filmindustrie, vorwiegend Shadow Maps verwendet. Bei komplexen Szenen wird aber oft immer noch ganz auf Schatten verzichtet.

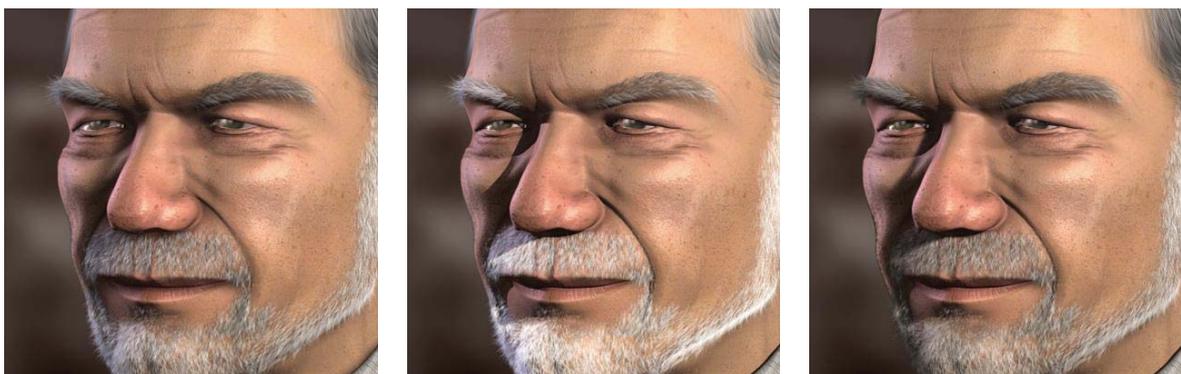
### 6.3.8 Fazit

In der Realität haben Lichtquellen eine definierte Ausdehnung, wodurch ein Übergang zwischen Licht und Schatten entsteht. Auch die Sonne, welche aufgrund ihrer großen Entfernung auf der Erde praktisch ein Parallellicht aussendet, verfügt über einen kleinen Schattenübergang. Grund sind hierfür diffuse Lichteigenschaften der Erdatmosphäre und Lichtkrümmungen an Objektkanten.

»Harte Schatten [der Raytraced Shadows] sehen eher unrealistisch aus, weil sich in der Natur nur selten Schlag-Schatten mit exakter und scharfer Schattengrenze finden lassen. « [Max]

Sehr komplexe Objekte wie virtuelle Charaktere benötigen zudem sehr lange bei der Berechnung von Raytracing-Shadows. Objekte wie Haare sorgen dafür, dass die Renderzeiten bei der Erstellung von virtuellen Charakteren verglichen mit Shadow-Maps ineffektiv hoch sind. Die weichen Schatten der Shadow Maps eignen sich folglich besser. Sie sehen natürlicher aus und sind schneller zu berechnen, solange die Shadows-Maps angemessen gewählt werden. Flächenschatten gelten als die realistischste Variante, um Schatten nachzuahmen, da sie die Schatten physikalisch am korrektesten berechnen. Allerdings sind die Renderzeiten bei detaillierten Objekten wie einem menschlichen Charakter mit Haaren ineffizient hoch.

Für detaillierte Renderings von virtuellen Charakteren gelten folglich die weichen Schatten der Shadow-Maps als am Bestens geeignet. Ambient Occlusion auf einem sehr subtilen Level ergänzt die realistische Wirkung.



**Abbildung 6-9: Schattenarten im Vergleich:** Shadow-Maps mit 2.000er Auflösung (links), Raytraced Shadows (Mitte) und Area Shadows (rechts).

## 6.4 Äußere Einflüsse

### 6.4.1 Umgebung

Logisch erscheint, dass es effizient ist, wenn nur die Objekte erstellt werden, die auch von der Kamera erfasst werden. Allerdings können auch Elemente außerhalb des sichtbaren Bereiches Einfluss auf den sichtbaren Bereich haben. Die Beleuchtung befindet sich meist außerhalb des Bildes. Ebenso können Schatten von nicht sichtbaren Objekten im sichtbaren Bereich zu sehen sein und Rückschlüsse auf dessen Existenz bilden. Glanzpunkte und vor allem Reflektionen können durch Lichter und andere Objekte außerhalb des Kamerawinkels beeinflusst werden. Die Umgebung um das gewünschte Zielobjekt herum kann großen Einfluss auf die Bildaussage haben und sollte als genau so wichtig genommen werden, wie der Bereich vor der Kamera.

Der erstellte Charakter besitzt reflektierende Materialien. Sowohl die Augen als auch die Haut haben eine reflektierende Eigenschaft zugewiesen bekommen. Folglich muss eine Umgebung geschaffen werden, um zu verhindern, dass eine schwarze Umgebung wiedergespiegelt wird und den Charakter abdunkeln würde. Sehr ineffizient wäre die Konstruktion einer kompletten Umgebung. Als effiziente Alternative erweist sich ein 360°-Panorama-Bild, welches auf ein kugelförmiges Objekt projiziert wird und um das Zielobjekt, also den Charakter positioniert wird. In dieser Szene sind schließlich alle relevanten Objekte vorhanden und die Szene ist vollständig.



**Abbildung 6-10: Umgebungstextur:** Eine eigene Textur der Umgebung wird auf ein kugelförmiges Objekt projiziert und als Umgebung interpretiert.

### 6.4.2 Tiefenunschärfe

Das in der Computergrafik von allen Programmen verwendete Abbildungsmodell entspricht der einer Lochkamera mit unendlicher Schärfe [Max]. Die Renderalgorithmen berechnen die Objekte der

Szene prinzipbedingt immer gleich scharf. Dies entspricht allerdings in keiner Weise der natürlich Welt, in der bei einem Linsensystem der Fokus immer nur an einer Stelle scharf ist. Je weiter ein Objekt von diesem Fokus entfernt ist und je höher die verwendete Brennweite ist, desto unschärfer erscheint dieses Objekt für den Betrachter. Auch wenn man dies als Abbildungsfehler interpretieren kann, sind Menschen daran gewöhnt und es trägt zu einem realistischen Bild bei.

»Der Effekt der Tiefenunschärfe trägt sehr zum realistischen Eindruck der Szene bei und ist ein probates Mittel, um die Aufmerksamkeit der Betrachter gezielt zu lenken.« [Asa]



**Abbildung 6-11: Tiefenunschärfe:** Tiefenmap, gerendertes Ausgangsbild, Bild mit gaußscher Unschärfe und angewendetes Bild mit Tiefenunschärfe.

Diese Tiefenunschärfe wird generell in der Postproduktion erstellt. Hierbei muss beim Rendern eine *Tiefenmap* (*Depth-Map* oder auch *Z-Depth*) mit erzeugt werden. Dadurch, dass beim standardmäßigen Rendern die Tiefeninformationen immer berechnet werden müssen, ergibt die Erzeugung einer Tiefenmap keine weiteren Renderzeiterhöhungen. Anschließend wird diese Tiefenmap mit einem weichgezeichneten Renderbild und dem originalen Renderbild verrechnet. Das weichgezeichnete Bild wurde hierbei mit einer gaußschen Normalverteilung und einem Radius von zwei Pixeln weichgezeichnet. Das Hintergrundbild wurde separat mit einem Radius von 13 Pixeln weichgezeichnet, um den Effekt noch weiter zu erhöhen.

Bei Filmproduktionen wird die Tiefenunschärfe standardmäßig eingesetzt. Bei *Final Fantasy* wurden zum Beispiel die Bilder mit mehreren Pässen ausgerendert. Zu den Pässen gehörte das eigentliche Bild als RGB-Pass, einem Alpha-Pass und einem Tiefen-Pass. Letztere diente der Berechnung der Tiefenschärfe und dem Blur [Hyp].

Aufgrund der positiven Wirkung und der geringen zusätzlichen Berechnungszeit sollte der Tiefeneffekt möglichst immer angewendet werden.

### 6.4.3 Bildstörungen

Wie im vorangegangenen Kapitel bereits angesprochen sind die Linsensysteme, die wir in unserer Welt kennen nie so perfekt, wie ein Computer ein Bild standardmäßig berechnen würde. Neben der Tiefenunschärfe sind manche Fotografien auch leicht unscharf. Oft ist auch eine gewisse Körnung zu betrachten. Durch Linsenverzerrungen können auch Bildteile verzerrt werden und durch die Tatsache, dass es keine perfekten Linsen gibt, kommen weitere physikalische Effekte vor, an die wir uns bei Kameras gewöhnt haben und als selbstverständlich hinnehmen. Chromatische Abberationen, dunklere Bildecken, Strahlenkränze und Halos bei direkten Lichtern oder auch bei Glanzlichtern sind nur einige davon. Solche Störungen wirken natürlich und ein Computerbild ohne gleichwertige Störungen kann schlimmstenfalls steril und unnatürlich wirken.



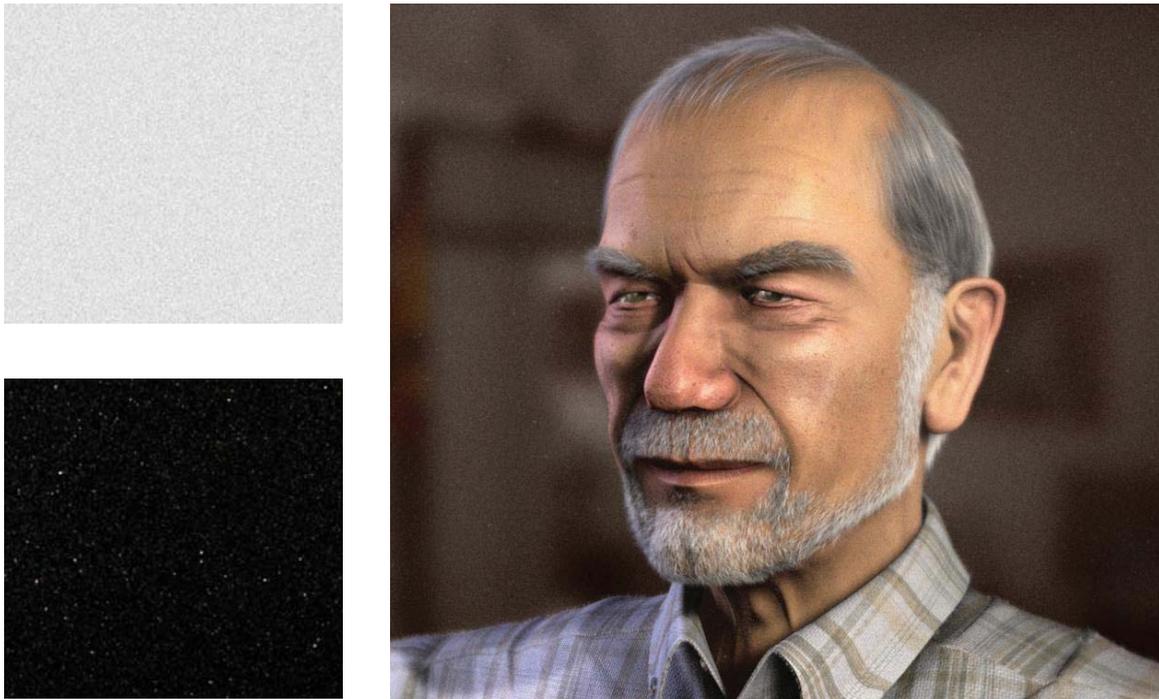
**Abbildung 6-12: Partikeleffekt Resistance 3:** Partikel im Playstation-3-Spiel Resistance 3 von Insomniac Games.

Staubpartikel, Blütenpollen und ähnliche Partikel fliegen durch die Luft und sollten in den Bildern ebenso integriert werden. Bei Hollywoodproduktionen werden solche Partikel schon länger eingesetzt und auch bei aktuellen Computerspielen ist die Verwendung von Staub- und sonstigen Partikeln vermehrt anzutreffen. Als Beispiel kann man auch hier *Resistance 3* nennen, bei der an fast allen Stellen solche Partikel Verwendung finden (vgl. Abbildung 6-12). Bei solchen Produktionen werden bedingt durch die Animationen umfangreiche Partikelsimulationen benötigt. Bei Einzelbildern können viele dieser Bildstörungen sehr schnell nachgeahmt werden und als Posteffekt in ein Bild integriert werden. Wichtig ist, dass solche Effekte nicht im Vordergrund stehen und nur auf subtiler Ebene wahrgenommen werden.

Für die Integration solcher Bildstörungen wurden für den erstellten Charakter folgende Anpassungen vorgenommen:

- Ein dupliziertes Bild mit 0,5 Pixel-Radius Gauß'scher-Weichzeichner wurde mit 25-prozentiger Sichtbarkeit über das Bild gelegt, um die Schärfe zu reduzieren.

- Der Kontrast wurde verringert. Zusätzlich wurden Helligkeitsanpassungen und Farbanpassungen durchgeführt.
- Eine Noise-Ebene wurde eingefügt um ein körniges Bild nachzuahmen.
- Ein Staubpartikel-Bild wurde integriert. Als Staubpartikel wurde ein einfaches frei verfügbares Sternbild der Nasa [Nas] verwendet.



**Abbildung 6-13: Bildstörungen:** Ein einfaches Noise und ein Partikelbild bilden hinzugefügt das finale Bild.

Die Erstellung des virtuellen Charakters ist damit fertig und das Bild vollendet.

## 7 Realismus

In den vorangegangenen Kapiteln wird beschrieben, mit welchen Techniken ein virtueller Charakter realistisch und dennoch einfach und effektiv aufgebaut werden kann. Dabei wurde eine Vorlage gewählt, anhand derer der Charakter erstellt wurde. Die Art und Weise, wie diese Vorlage umgesetzt wird, liegt dabei, wie auch in allen Filmstudios, bei Computerspielerherstellern oder sonstigen Firmen, allein bei dem Charakterersteller. Dieser erstellt einen virtuellen Charakter und versucht diesen Schritt für Schritt der Vorlage immer ähnlicher zu machen und seiner eigenen Vorstellung von Realismus immer näher zu kommen [Rad].

Die Charaktererstellung ist derzeit mehr ein subjektiver Prozess, der von den Renderzeiten begrenzt wird. Es bleibt aber eher ein künstlerischer Ablauf – nicht umsonst werden die Personen, die virtuelle Charaktere erschaffen, als Character-Artists bezeichnet.

Ziel sollte es sein, diesen reinen subjektiven Vorgang zu beenden und näher zu betrachten, von welchen Faktoren der Realismus-Faktor beeinflusst wird. Auf diesen Faktoren aufbauend sollten neue Renderalgorithmen erstellt werden, die dies mehr berücksichtigen. Aber stattdessen wird versucht diese Renderprogramme mehr und mehr physikalisch korrekt arbeiten zu lassen. Desweiteren sollten die Techniken zur Herstellung von virtuellen Charakteren mehr auf Basis des Kosten-Nutzen-Faktors hinsichtlich Realismusgewinns bei Beachtung der Renderzeit vorgenommen werden.

Die Fragestellung lautet hier folglich: Welche Techniken sind wichtig, effizient und damit kostengünstig für den Realismusfaktor?

### 7.1 Begriff Realismus

Was aber ist Realismus überhaupt? Erst wenn diese Frage beantwortet ist, können weitere Schlüsse gezogen werden. Als realistisch bezeichnet man etwas, das realitätsnah ist.

»Realität ist all das, was unabhängig von unserer subjektiven Einschätzung gegeben, also nachweisbar und überprüfbar ist.« [Hen]

Die Realität beruht also auf Fakten und bezeichnet alles, was nachweisbar existiert. Zusammengefasst definiere ich Folgendes:

**Definition 7-1:** Als realistisch sehen wir Alles an, was nachweisbar existiert.

**Definition 7-2:** Der Realitätsfaktor gibt den Grad an Realismus an, von realistisch bis nicht realistisch/virtuell.

## 7.2 Experimentelle Untersuchung

### 7.2.1 Realismusfaktor

Wie kann man den Realismusfaktor verschiedener Bilder messen? Hierzu gibt es verschiedene Arbeiten die sich mit diesem Thema beschäftigen. Einige Arbeiten zu diesem Thema beschäftigen sich mit dem menschlich Aufbau und dessen Wahrnehmung [Bru]. Andere Arbeiten gehen mehr in die vergleichende Analyse von einem realen und einem computergenerierten Bild [Rus], [Mcn]. Wie kann man aber den Realitätsfaktor eines Bildes bestimmen, von dem es kein vergleichendes Bild gibt? Hierzu existieren leider kaum Arbeiten. [Cha] und [Red] versuchen die Frage mit einer Befragung von Testpersonen zu lösen, wobei Letztere gute Ansätze für die vorliegende Arbeit liefert. Bei [Red] werden darüber hinaus der Realismus anhand von realen Fotos untersucht.

Für den Menschen ist die Welt, in der er lebt, die Realität. Diese Welt kennt er von Geburt an und er speichert seine Erfahrungen als Wissen ab. Dieses gespeicherte Wissen nutzt er nun, um zwischen der Realität und einer virtuellen Realität zu unterscheiden. Für einen Menschen entscheiden demnach seine Erfahrungen, ob ein Bild realistisch erscheint oder nicht. Der Realitätsfaktor sollte demzufolge ein subjektiver Faktor sein, der nur durch eine experimentelle Umfrage erfasst werden kann.

### 7.2.2 Experimentalaufbau

Versuchspersonen wurden verschiedene Bilder gezeigt, bei denen sie angeben mussten, wie realistisch sie die Bilder empfinden. Die Antwortmöglichkeiten reichen hierbei auf einer Skala von 1 (nicht realistisch) bis 5 (realistisch).

Da man den Realismusfaktor nicht als Ganzes erfassen kann, wird dieser auf die einzelnen Teilbereiche heruntergebrochen. Es werden Referenzbilder genommen und von diesen ausgehend weitere Bilder mit jeweils einem Faktor als Unterscheidungsmerkmal verändert. Alle anderen Parameter müssen unbedingt so konstant wie möglich bleiben, um eine Aussagekraft bezüglich des einen geänderten Faktors zu erhalten. Kameraperspektive und Lichtmenge zum Beispiel sollten gleich bleiben, um die Ergebnisse vergleichen zu können. Zusätzlich werden einige Kontrollbilder angezeigt, auf denen mehr als ein Faktor geändert werden. Um den Umfang für die Testpersonen in Grenzen zu halten, wurden nur einige ausgewählte Funktionen untersucht.

Gezeigt wurden den Testpersonen mehrere gerenderte Bilder des zuvor beschriebenen in dieser Diplomarbeit erstellten 3D-Charakters. Dabei wird das Augenmerk wieder auf den Kopf gelegt. Zu den zu untersuchenden Faktoren gehörten auch folgende Fragestellungen:

- **Schattenart:** Einfache Schatten mit geringer Komplexität werden mit den rechenintensiven Area Shadows verglichen. Rechtfertigen die langen Renderzeiten die komplexen Schatten oder erzielen die einfachen Schatten schon realistische Ergebnisse?

- **Anzahl der Lichter:** Klassische Beleuchtungsmodelle bevorzugen mindestens drei Lichter. Zahlt sich der Mehraufwand auch im Realitätsfaktor aus?
- **Details:** Je mehr Details eingebaut werden, desto mehr Realismus sollte erreicht werden. Wie gewichtig ist diese Aussage? Und welche Technik bietet hier den besten Kosten-Nutzen-Faktor? Zu den untersuchten Detailmöglichkeiten gehören: Modellunterteilung (Displacement- und Bumpmapping), Texturen (Farbkanal, Glanz, Spiegelung) und Haare.
- **Antialiasing:** Wie sind dessen Auswirkungen auf den Realismusfaktor?
- **Umgebung:** Einfluss der Umgebung durch Tiefenunschärfe und Bildstörungen.

Funktion	Renderzeit (in Sekunden)	Veränderung* (in Sekunden)	Veränderung* (in Prozent)
Farbtextur: aus	280	-	-
Farbtextur: an	299	19	7
Glanz & Spiegelung: aus	144	-	-
Glanz & Spiegelung: an	299	155	107,6
Displacement: aus, Bump: aus	243	-	-
Displacement: an, Bump: an	299	56	23,0
Haare: aus	186	-	-
Haare: an	299	113	60,8
Lichteranzahl: eins	158	-	-
Lichteranzahl: drei	299	141	89,2
Schatten: aus	229	-	-
Schatten: Shadow Maps 250 Px	289	60	26,2
Schatten: Shadow Maps 2.000 Px	299	70	30,6
Schatten: Raytraced	475	246	107,4
Schatten: Area Shadows	3.539	3.310	1.445,4
Tiefenschärfe & Bildstörungen: aus	299	-	-
Tiefenschärfe & Bildstörungen: an	302	3	1,0
Antialiasing: aus	128	-	-
Antialiasing: an	299	171	133,6

\* Veränderung zum relevanten Faktor (Referenzbild)

**Tabelle 7-1: Kostenvergleich:** Renderzeiten der einzelnen untersuchten Faktoren und ihre Veränderungen in Sekunden und Prozent.

Die Kosten eines computergenerierten Bildes bilden sich immer aus der Erstellungszeit, also der Renderzeit, die benötigt wird, um das Bild zu erstellen. Diese Zeit wird bei allen Bildern gemessen. Einerseits bei einem Referenzbild, andererseits wird der gewünschte Faktor ein- bzw. ausgeschaltet. Die Differenz dieser gemessenen Renderzeiten bildet die Kosten für einen bestimmten Faktor.

Insgesamt wurden 13 unterschiedliche Bilder<sup>33</sup> mit den zuvor genannten Faktoren in der Größe 554 x 500 Pixel angefertigt und den Testpersonen über das Internet zugänglich gemacht. Hierzu wurde eine Website erstellt, auf der die Bilder nacheinander gezeigt wurden und jeweils bewertet werden konnten. Jedes Bild konnte beliebig oft angeschaut und mit allen anderen Bildern verglichen werden, bevor die Umfragepersonen ihre Meinung darüber abgaben.

### 7.2.3 Fehlerbetrachtung

Es gilt mehrere Fehlerquellen bei dieser Umfrage zu eliminieren, wenngleich einige ungewollte Einflüsse nicht vermieden werden konnten.

So wurde unter anderem die Anzahl der Lichter verändert, was aber eine Änderung der Lichtstärke insgesamt zur Folge hatte. Diese Szene musste also wieder aufgehellt werden und konnte so ungewollt Einfluss auf den Realismusfaktor haben. Auch bei der Beeinflussung der Schatten kam es zu Veränderungen. Die Schärfe der Schatten beeinflusst auch die Intensität der Schatten, wodurch manche Stellen dunkler und andere heller waren. Dieser Einfluss zeigte sich sehr deutlich bei den Haaren, was den Realitätsfaktor beeinflussen konnte.

### 7.2.4 Durchführung und Analyse

Insgesamt nahmen 366 Testpersonen an der Umfrage<sup>34</sup> teil und lieferten valide Testergebnisse. Invalide Datensätze wurden entfernt und nicht in das Umfrageergebnis eingerechnet.

Die Antwortmöglichkeiten für den Realismusfaktor (im Folgenden als R bezeichnet) werden für die Analyse normiert. Die Antwort 1 (nicht realistisch) wird mit 0 gleich gesetzt. Die Antwort 5 (realistisch) wird mit 1 gleichgesetzt. Hierdurch ergeben sich für die Antwortmöglichkeiten 1 bis 5 die Werte  $R=0,0$ ;  $R=0,25$ ;  $R=0,5$ ;  $R=0,75$  und  $R=1,0$  die mit den Testpersonen verrechnet werden. Ein Wert von 1,0 entspricht also einem Foto und der Wert 0,0 ganz klar einem computergeneriertem Bild.

## 7.3 Erfahrung beeinflusst Wahrnehmung

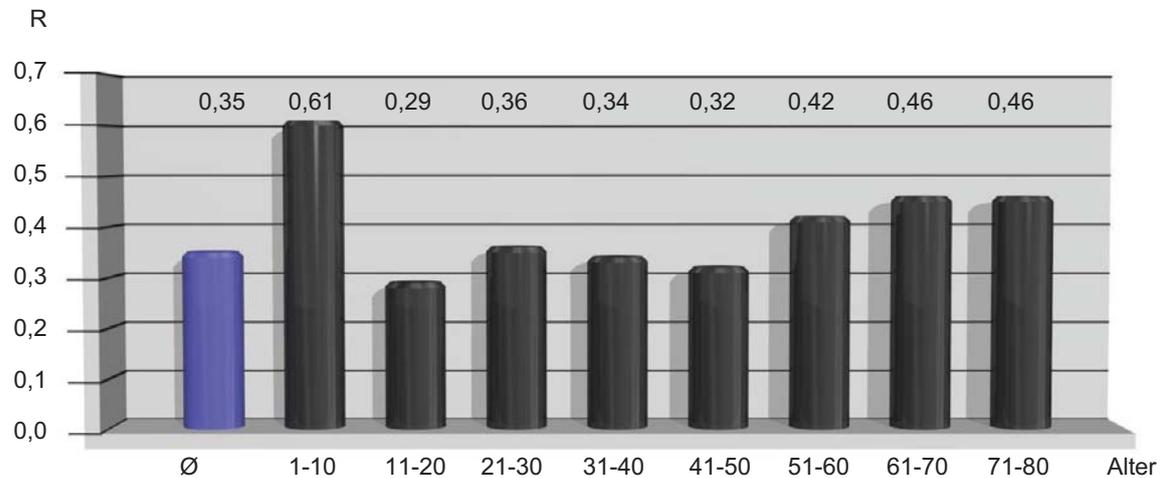
Der Mensch lernt, dass die Welt um ihn herum die Realität ist. Dabei lernt er auch, dass die computergenerierte Welt nicht der Realität entspricht und wie in der Malerei nur der realen Welt nachempfunden ist. Hiermit mache ich die Annahme und stelle die Hypothese auf, dass man – je besser man die Realität kennt und je besser man sich mit der computergenerierten Welt auskennt – den Unterschied erkennt.

---

<sup>33</sup> Alle in der Umfrage gezeigten Bilder sind in Kapitel D Anhang ersichtlich.

<sup>34</sup> Die Umfrageergebnisse setzen sich aus einer Umfrage Ende 2010 und einer Umfrage Anfang 2012 zusammen.

Subjektive Erfahrungen der Testpersonen sollten demnach nachweislich einen Einfluss auf den Realitätsfaktor der Umfrage haben. Um diese Annahmen zu bestätigen, wurden die 366 Testpersonen bei der Umfrage zusätzlich nach ihrem Alter gefragt. Die Ergebnisse, welche in Abbildung 7-1 dargestellt sind, scheinen die Annahmen zu bestätigen.



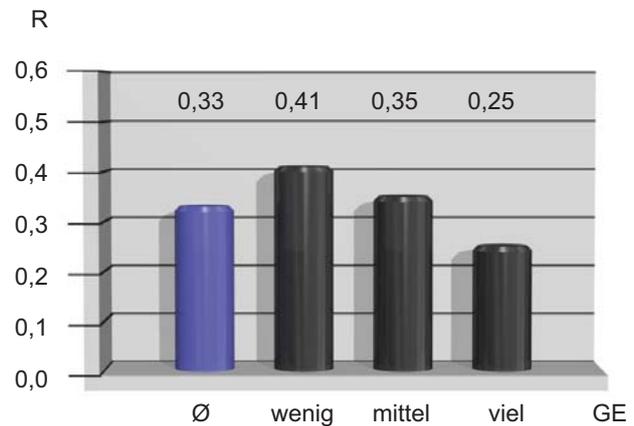
**Abbildung 7-1: Abhängigkeit des Realismusfaktors R im Bezug auf das Alter.**

Die Gruppe der jüngsten Teilnehmer (bis 10 Jahre) mit der anzunehmenden geringsten Lebenserfahrung wertete die Bilder realistischer, als alle anderen älteren Gruppen. Ausgehend von den Testergebnissen kann man die Gruppe bis 12 Jahre erweitern, wobei der durchschnittliche Realismusfaktor bei  $R = 0,54$  (+31% über dem Durchschnitt) liegt. Die nächste erkennbare Gruppe scheint bei 13 bis 50 Jahren zu liegen. Diese Gruppe kommt auf einen durchschnittlichen Realismusfaktor  $R = 0,34$  (-2%) und scheint bei der Beurteilung der Bilder am kritischsten zu sein. Eine mögliche Erklärung wäre, dass diese Gruppe wohl am meisten Kontakt mit computergenerierten Bildern hat. Die letzte Gruppe liegt bei 51 Jahren und älter mit einem Realismusfaktor  $R = 0,43$  (+13%). Diese Testpersonen weisen wieder weniger Kontakt zu computergenerierten Bildern auf.

Aufgrund der sehr unausgewogenen Verteilung der Testteilnehmer im Hinblick auf ihr Alter sind diese Ergebnisse allerdings mit Vorsicht zu genießen. Bei insgesamt 366 Testteilnehmern gehören 8 Personen (ca. 2% aller Teilnehmer) zur Gruppe bis 12 Jahre, 316 (ca. 86%) zur Gruppe 13 bis 50 Jahre und 39 (ca. 11%) zur Gruppe 51 Jahre und älter. 3 Teilnehmer gaben keine Altersangaben an.

Um sicherere Aussagen treffen zu können wurde bei 219 Testteilnehmern zusätzlich gefragt, wie viel Erfahrung sie mit Computerspielen haben. Damit sollte gewährleistet werden, dass die Gruppen gleich ihrem Kontakt zu computergenerierten Bildern entsprechen zugeordnet werden konnten. Hierbei ordneten sich 57 Personen (ca. 26% aller Teilnehmer) zu der Gruppe mit wenig Erfahrung bei Computerspielen, 87 (ca. 40%) zu der mit durchschnittlich viel Erfahrung und 75 Personen (ca. 34%) zu der Gruppe mit viel Computerspielerfahrung. Eine bessere Aufteilung der Gruppen ist nun ersichtlich.

Wie erwartet zeigte sich die Gruppe mit wenig Erfahrung bei Computerspielen am wenigsten kritisch. Sie wertete die Bilder im Durchschnitt mit einem Realismusfaktor von  $R = 0,41$ . Dies waren 22% mehr als der Durchschnitt aller 219 Testpersonen, die eine Wertung von  $R = 0,33$  abgaben. Die Annahme des Einflusses der geringeren Erfahrung mit computergenerierten Grafiken scheint sich zu bestätigen. Die mittlere Gruppe wertete mit einem durchschnittlichen Realismusfaktor  $R = 0,35$  (+6% über dem Durchschnitt) und die Gruppe mit der meisten Computerspielerfahrung wertete die Bilder mit  $R = 0,25$  (-24%). Dementsprechend definiere ich Folgendes:



**Abbildung 7-2: Abhängigkeit des Realismusfaktors R und der Computerspielerfahrung GE.**

**Definition 7-3:** Der Realismusfaktor ist ein subjektiv wahrgenommener Faktor des Menschen und beruht auf dessen Interpretation der Umwelt.

Folgende Auswertung der Befragung soll helfen, bei der Erstellung realistischer Charaktere effektiver zu arbeiten. Wenn man weiß, welche Funktionen mehr Realismus bringen, kann man als Hersteller von 3D-Charakteren sein Hauptaugenmerk mehr auf diese lenken. Als zweiter Schritt wird zusätzlich bei diesem Experiment auf die Kosten geachtet. Als Kosten werden hierbei die Renderzeiten angesehen. Schnellere Renderzeiten bedeutet weniger Computer, die rendern müssen und geringere Produktionszeiten, wodurch die Produktionskosten gesenkt werden können. Wenn man die Wirkung der Funktionen in Bezug auf Realismus und Kosten kennt, kann man zielgerichteter und effektiver realistische 3D-Charaktere erstellen. Desweiteren können Hersteller von Renderalgorithmen ihre Programme zielgerichtet auf die Funktionen hin optimieren, welche den größten Nutzen im Hinblick auf die Kosten haben. Dies wäre vielleicht effektiver, als der bisherige Versuch, die Renderalgorithmen immer physikalisch korrekter arbeiten zu lassen.

## 7.4 Einfluss der Details auf den Realismus

In vielen Bereichen des heutigen Lebens müssen die Dinge perfekt sein, dürfen keine Makel haben. Models zum Beispiel haben scheinbar reine Haut ohne eine Unebenheit. Auch bei Film und Fernsehen werden die Personen vor der Kamera geschminkt, um möglichst perfekt auszusehen. Allerdings wissen wir auch, dass die wenigsten Menschen perfekt sind und Falten, Hautrötungen, Leberflecke etc. zu jedem Menschen dazu gehören. Ein zu perfekter Mensch wirkt künstlich und unrealistisch. Und genau das ist auch bei computergenerierten Charakteren zu sehen – so jedenfalls

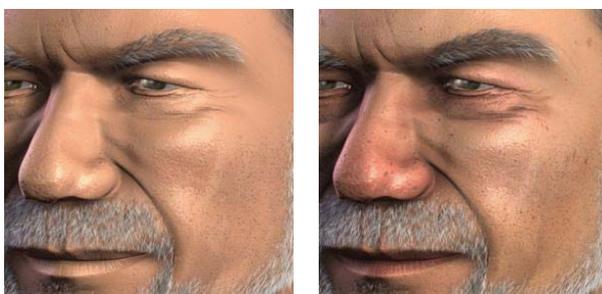
die weit verbreitete Annahme. Das Experiment sollte diese Annahme überprüfen und aufzeigen, welche Faktoren am realistischsten sind und welche Faktoren den Realismusgrad am effizientesten bewirken.

Für das Experiment wurden mehrere Bilder gerendert, bei denen einige Faktoren an- bzw. ausgeschaltet waren. Zu den Faktoren, welche kleinere und größere Details bilden können, wurde die Wirkung von Farbtexturen (Colormap), Glanz & Spiegelung (Specular & Reflection), Haaren, Displacement- und Bumpmapping untersucht.

### 7.4.1 Farbtextur

Die Farbtextur beinhaltet alle oberflächlichen Strukturen der Haut, wie Farbunterschiede an Augen, Nase, Wangen und Mund. Vor Allem die Einfärbung der Lippen ist als markanter Unterschied sofort sichtbar. Hinzu kommen weitere Details wie Leber- und Altersflecke. Bei dem Experiment wurden zwei Bilder gezeigt, einmal mit und einmal ohne aktivierte Farbtextur. Der Charakter im gezeigten Referenzbild ohne aktivierte Farbtextur bekam eine Durchschnittsfarbe der Farbtextur, um eine Farbabweichung insgesamt so gering wie möglich zu halten. Das Interesse liegt ja nur auf den Details der Textur und nicht auf der Farbgebung.

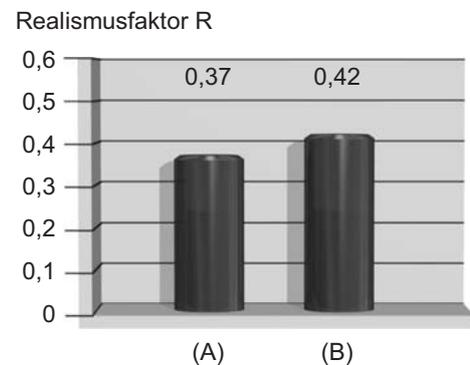
Die Ergebnisse zeigen, dass das Referenzbild ohne Farbtextur von den Testpersonen mit einem Realismusfaktor  $R = 0,37$  bewertet wurde. Das Bild mit aktivierter Farbtextur wurde mit  $R = 0,42$  um  $0,05$  realistischer betrachtet. Dies ist eine positive Veränderung um knapp 15 Prozent.



Ohne Farbtextur (A)

Mit Farbtextur (B)

**Abb. 7-3: Unterschied Farbtextur:** Ausschnitte aus den gezeigten Bildern ohne und mit aktivierter Farbtextur.



**Abb. 7-4: Der Realismusfaktor in Abhängigkeit von einer Farbtextur.**

Es kann also davon ausgegangen werden, dass eine Farbtextur das Realismusempfinden erhöht, wenn auch nur in einem geringen Maße. Die Annahme, dass Details das Realismusempfinden erhöhen, kann hiermit bestätigt werden.

### 7.4.2 Glanz- und Spiegeltextur

Die menschliche Haut hat keine matte Oberfläche. Durch die Eigenschaft der Haut, aber vor allem gefördert durch die Produktion von Schweiß durch die Hautporen glänzt die Haut. Zur Nachahmung

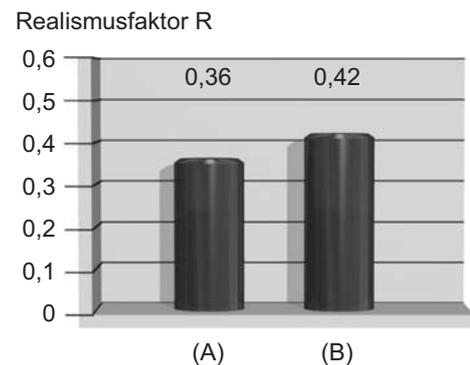
dieses Effekts gibt es bei den Texturen die Spiegelungsfunktion, die die Umgebung spiegeln kann. Ergänzt wird dies durch Glanztexturen, wodurch die Spiegelungen der Lichter nachgeahmt werden, welche naturgemäß stärker als Spiegelung wahrgenommen werden.

Für das Experiment wurden zwei Bilder gerendert. Bei dem Referenzbild wurden diese Funktionen deaktiviert und bei dem Vergleichbild wurden Spiegelfunktion und Glanzfunktion aktiviert. Die Testpersonen werteten das Referenzbild ohne Spiegel- und Glanzfunktion mit einem Realismusfaktor  $R = 0,36$ . Das Bild mit aktivierter Spiegelungs-/Glanzfunktion wurde mit  $0,42$  um  $0,07^{35}$  realistischer bewertet. Dies ist eine Zunahme von 18 Prozent. Glanz und Spiegelung wird von den Testpersonen als Eigenschaft angesehen, die einen virtuellen Charakter realistischer erscheinen lässt.



Ohne Glanz+Spiegelung (A) Mit Glanz+Spiegelung (B)

**Abb. 7-5: Unterschiede Glanz+Spiegelung:**  
Ausschnitt einmal ohne und einmal mit  
Glanz und Spiegelung.



**Abb. 7-6: Der Realismusfaktor in  
Abhängigkeit von  
Glanz+Spiegelung.**

### 7.4.3 Displacement- und Bumpmapping

Eine der effektivsten Varianten um kleine Details in eine 3D-Geometrie einzuarbeiten, ohne die Hauptgeometrie zu verändern, ist die Verwendung von Displacementmaps. Hierbei wird je nach eingestellter Stärke die Oberfläche erst unterteilt und dann anhand einer Textur weiter verformt. Dies eignet sich ideal, um kleinere Strukturen wie Falten einzuarbeiten. Ergänzt wird dies durch die Bumpmaps, welche die Geometrie nicht verformen, aber eine solche optisch mit einigen Einschränkungen erscheinen lässt. Mit Bumpmaps werden kleinere Falten und Hautporen kreiert.

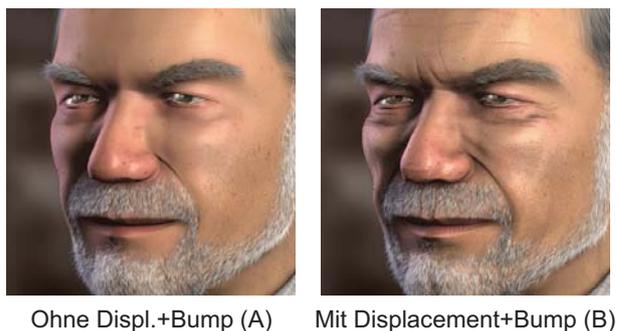
Diese Vorgehensweise – 3D-Geometrie erstellen, mit Displacementmaps Details einarbeiten und letztendlich mit Bumpmaps noch weitere kleinere Details einarbeiten – stellt die derzeitige weit verbreitete Arbeitsweise dar und gilt als Standard. Ist die Verwendung von Displacement- und Bumpmaps aber auch effektiv? Dies sollte auch dieses Experiment zeigen, bei dem beide Verfahren zusammengefasst wurden, um die Bilderanzahl insgesamt für die Testpersonen im Rahmen zu halten.

Für das Experiment wurde ein Bild nur mit der Hauptgeometrie gerendert, wobei das Phong-Shading aktiv war. Als Vergleichbild wurde ein Bild genommen, wo zusätzlich weitere Details wie die

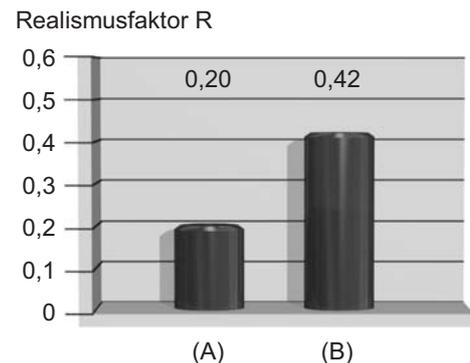
<sup>35</sup> Wegen Rundungsdifferenzen kann es zu Abweichungen kommen. Weitere berechnete Werte siehe Anhang.

angesprochenen Falten und Poren durch ein aktiviertes Displacement- und Bumpmapping erzeugt wurden. Annahme war, dass sich der Realismusfaktor durch die hinzugefügten Falten und Poren erhöhen sollte.

Und tatsächlich wurde das Bild ohne Displacement- und Bumpmapping mit  $R = 0,20$  als das unrealistischste Bild von allen gezeigten Bildern im ganzen Test<sup>36</sup> bewertet. Das Referenzbild mit Aktivierung der Funktionen wurde mit einem Realismusfaktor  $R = 0,42$  um  $0,22$  realistischer bewertet. Dies entspricht einer Realismuszunahme von 109 Prozent und ist, um dies schon vorwegzunehmen, die Funktion mit dem größten prozentualen Realismuszuwachs.



**Abb. 7-7: Unterschiede Displacement+Bump:**  
Ausschnitt einmal ohne und einmal mit Displacement und Bump.



**Abb. 7-8: Der Realismusfaktor in Abhängigkeit von Displacement+Bump.**

Eine Verwendung von Displacement- und Bumpmaps, um kleinere Details bei einem virtuellen Charakter hinzuzufügen, führt zu einer Erhöhung des wahrgenommenen Realismusempfindens. Diese Erhöhung ist nicht nur nachweislich vorhanden, sondern auch bemerkenswert hoch.

#### 7.4.4 Haare

Haare sind ein wesentlicher Bestandteil eines jeden Menschen. Demzufolge sollten Haare auch ein wesentliches Detail sein, welches bei virtuellen Charakteren den Realismusfaktor erhöhen sollte. Hierbei sei nicht nur das Haupthaar auf dem Kopf gemeint, sondern auch die Augenbrauen, die Gesichtshaare wie der Bart, aber auch kleinere Haare im Gesicht. Zusätzlich ging es auch um Fasern auf der Kleidung, die mittels Haarsimulationen generiert wurden.

Um die Wirkung der Haare zu beurteilen wurden für das Experiment zwei Bilder gerendert. Bei dem ersten Bild wurde ganz auf Haare verzichtet. Bei dem Vergleichsbild wurden wiederum alle Haare aktiviert, die mittels Haarsimulation bei unserem virtuellen Charakter erstellt wurden.

Die Testpersonen werteten den Charakter ohne Haare auf einen Realismusfaktor  $R = 0,21$ . Das Vergleichsbild mit Haaren wurde hingegen mit  $R = 0,42$  um  $0,21$  weit aus höher bewertet. Der genaue Realismuszugewinn beläuft sich auf ca. 107 Prozent. Auch bei diesem Detail gilt folglich: Bei der

<sup>36</sup> Eine Übersicht über die Testergebnisse ist in Kapitel D Anhang zu sehen.

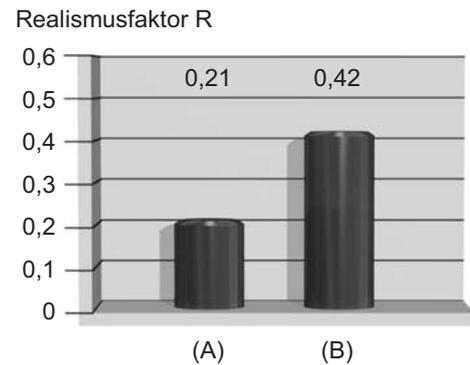
Verwendung von Haarsimulationen wird der Realismusfaktor nachweislich stark erhöht.



Ohne Haare (A)

Mit Haaren (B)

**Abb. 7-9: Unterschied Haare:** Ausschnitte aus den gezeigten Bildern ohne und mit aktivierten Haaren.



**Abb. 7-10: Der Realismusfaktor in Abhängigkeit von Haaren.**

## 7.5 Wirkung der Umgebungseinflüsse

Bei den bisherigen Tests wurde untersucht, welche Wirkung sich einstellt, wenn man das Objekt, also den Charakter selber, manipuliert. Es gibt aber auch viele Eigenschaften, die die Umgebung manipulieren. Nun soll untersucht werden, welche Wirkung diese Eigenschaften erzielen können. Hierunter gehören zum Beispiel die Lichtquellen. Hat die Anzahl an Lichtquellen einen Einfluss auf das Realismusempfinden? Wirkt ein Bild mit Schatten realistischer als ohne? Wenn ja, welche Schattenart wirkt am realistischsten? Welche Wirkung hat Tiefenunschärfe und weitere Störungen? Welchen Vorteil bringt ein Antialiasing bei einem virtuellen Charakter? Diese Fragen sollen im Folgenden durch die Testergebnisse beantwortet werden.

### 7.5.1 Anzahl Lichtquellen

Das Standardlichtsetup, das bei den meisten 3D-Szenen zum Einsatz kommt, ist die klassische 3-Punkt-Beleuchtung, wie Sie auch bei Fotos und Filmen am meisten eingesetzt wird. Etwas seltener, aber immer noch oft zu sehen, ist eine 2-Punkt-Beleuchtung. Ein Beleuchtungsmodell mit nur einem Licht ist selten anzutreffen und wirkt schaurig und beklemmend. Nicht umsonst findet man diese Beleuchtung in beängstigenden Filmszenen oder Horrorfilmen.

Für die Realismusuntersuchung wurden zwei Bilder erstellt. Das erste Bild wurde mit nur einem Punktlicht ausgeleuchtet, während das zweite Bild mit insgesamt drei Punktlichtern ausgeleuchtet wurde. Die Lichtintensitäten wurden leicht variiert, so dass die beiden Bilder dennoch ein ähnliches Helligkeitsempfinden aufweisen. Angenommen wird eine Erhöhung des Realismusfaktors bei mehr Lichtquellen, da dieses Lichtsetting öfter wahrgenommen werden sollte.

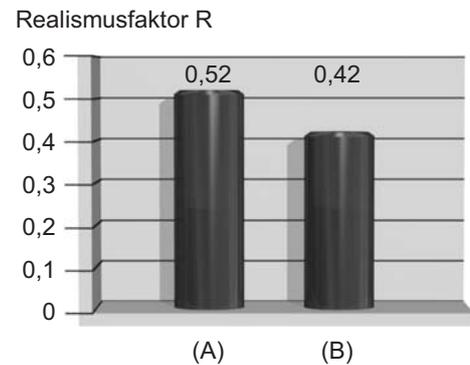
Die Testpersonen werteten das Bild mit einer Lichtquelle mit einem Realismusfaktor  $R = 0,52$ . Das Bild mit den drei Lichtquellen wurde hingegen mit  $R = 0,42$  um 0,10 weniger realistisch angegeben. Mehr Lichtquellen reduzieren in diesem Fall den Realismusfaktor um 18 Prozent. Dies zeigt, dass im

Gegensatz zu der vorher getroffenen Annahme mehr Lichtquellen nicht den empfundenen Realismusgrad erhöhen. Ein stärkerer Schatten, der bei mehreren Lichtern reduziert wird, könnte der Grund hierfür sein.

Ohne, dass man sich jetzt schon die Kosten der gerenderten Bilder anschaut, kann man abschließend zu den Lichtquellen sagen, dass so wenig Lichter wie möglich den größten Nutzen hinsichtlich des Realismusgrades bringen.



**Abb. 7-11: Unterschied Licht:** Ausschnitte aus den gezeigten Bildern mit einem und drei aktivierten Lichtern.



**Abb. 7-12: Der Realismusfaktor in Abhängigkeit von der Anzahl an Lichtern.**

### 7.5.2 Schatten

Es gibt verschiedene Arten, um Schatten zu erzeugen. Je nach Art der Schatten dauert die Berechnung zur Erzeugung des Schattens unterschiedlich lang.

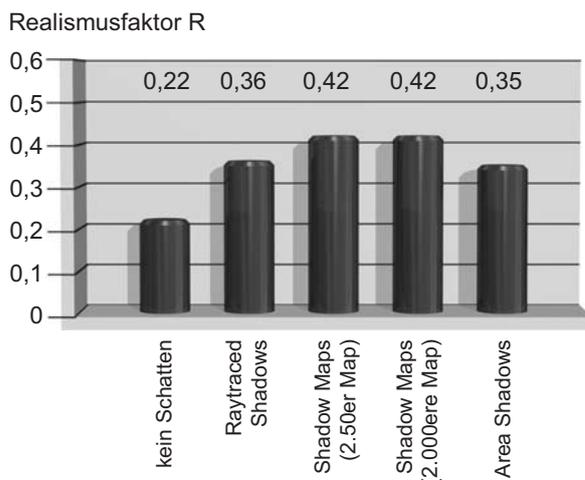
Für das Experiment wurden fünf verschiedene Bilder gerendert, wobei die gängigsten Schattenarten angewandt wurden: Die einfachen harten Schatten der Raytraced Shadows, weiche Schatten der Shadow Maps mit einer Schattenmap von 250 Pixel Größe und Schattenmap mit einer Größe von 2.000 Pixel, welche genauer, aber auch etwas zeitintensiver sind. Schließlich noch die Area Shadows, die die Welt physikalisch am korrektesten wiedergeben, aber auch deutlich am zeitintensivsten zu berechnen sind (vgl. Tab. 7-1). Als Referenz wurde ein Bild ohne Schatten erstellt.



**Abbildung 7-13: Ausschnitte der Kategorie Schatten:** Bild ohne Schatten, mit Raytraced Shadows, Shadow Maps mit 250er Map, Shadow Maps mit 2.000er Map und Bild mit Area Shadows.

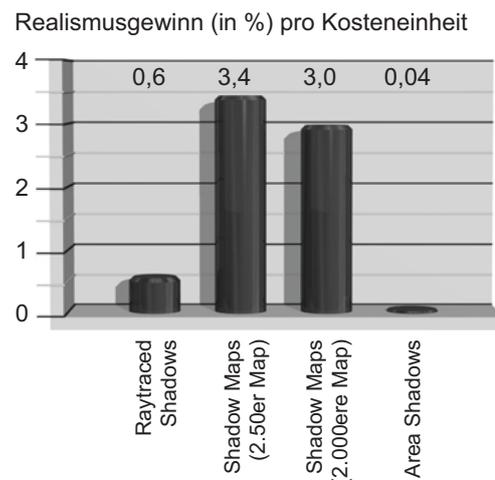
Unterschiede, die keinen sichtbaren Zusammenhang mit den Schatten hatten und dennoch durch die unterschiedlichen Schattenarten auftraten, wurden versucht so minimal wie möglich zu halten. Beispielsweise haben die Schatten einen starken Einfluss auf die Wirkung der Haare. Unterschiedlicher Schatten, der unter den Haaren liegt, lässt diese dunkler oder heller erscheinen.

Die Ergebnisse des Experimentes zeigten deutliche Unterschiede in der Wahrnehmung der gezeigten Bilder auf das Realismusempfinden der Testpersonen. Wie erwartet empfanden die Probanden das Bild ohne Schatten am unrealistischsten ( $R = 0,22$ ). Das Bild mit aktivierten harten Schatten der Raytraced Shadows wurde mit  $R = 0,36$  ( $\Delta = 0,14 / +60\%$ ) etwas realistischer empfunden und gehört zu den am wenigsten realistischsten Schatten. Dies deckt sich mit der weit verbreiteten Meinung, dass harte Schatten unrealistisch wirken. Dies liegt wohl auch daran, dass in der Wirklichkeit solche harten Schatten sehr selten zu sehen sind, da die Lichtquelle extrem möglichst weit weg, aber stark genug sein muss. Erstaunlicherweise wurden die rechenintensiven Area Shadows ähnlich schlecht beurteilt mit  $R = 0,35$  ( $\Delta = 0,13 / +56\%$ ). Am besten wurden die weichen Schatten der Shadow Maps beurteilt, wobei die weichen Schatten der 250 Pixel großen Schattenmap mit  $R = 0,42$  ( $\Delta = 0,20 / +90\%$ ) fast identisch bewertet wurde wie der 2.000 Pixel großen Schattenmap mit  $R = 0,42$  ( $\Delta = 0,20 / +90\%$ ). Die weichen Schatten werden hier folglich am Realistischsten wahrgenommen.



**Abb. 7-14: Realismusfaktoren der Schatten:**

Ein Aktivieren von Schatten ergibt einen deutlichen Realismuszugewinn.



**Abb. 7-15: Realismuszugewinn mit Kosteneinbeziehung aller Schatten.**

Die Ergebnisse zeigen, dass ein Aktivieren der Schatten unabhängig von der Art des Schattens eine Zunahme an Realismus bei den Testpersonen bewirkt. Hervorzuheben sei, dass die physikalisch korrekt arbeitenden komplexeren Area Shadows nicht wie erwartet die realistischsten Resultate brachten. Im Gegenteil, die weichen Schatten wurden als weit realistischer wahrgenommen. Zukünftige Renderalgorithmen müssen also nicht immer physikalisch korrekt arbeiten, um realistische Charaktere zu erstellen. Eine Optimierung der Renderalgorithmen in anderen Bereichen könnte im Hinblick auf virtuelle Charaktere mehr Gewinn erzielen.

Um die Schattenarten noch besser zu vergleichen, sollte man sich nicht nur die Wirkung (Realismusfaktor) sondern auch die Kosten (Renderzeit) anschauen. Um einen Vergleichswert zu erhalten, wird hierfür der Realismusgewinn  $\Delta R$  durch die Anzahl Kosteneinheiten geteilt. Eine Kosteneinheit wird hierbei als ein Prozentpunkt Renderzeitveränderung definiert (siehe Tabelle 7-1). Somit bekommt man einen Wert, der aussagt, wie viel Realismusgewinn man pro einprozentiger Renderzeiterhöhung bekommt. Je größer der Wert ist, umso besser. Hiermit gilt meine folgende Definition:

**Definition 7-4:**

$$\text{Realismusgewinn pro Kosteneinheit (RK)} = \frac{\text{Realismusfaktor } R}{\Delta \text{ Renderzeit}}$$

Schaut man sich nun diese Werte für die einzelnen Schattenfunktionen an (vgl. Abb. 7-15), so haben die weiter oben getroffenen Aussagen über die Schattenarten weiter Bestand. Sie werden sogar noch ein wenig extremer. Harte Schatten der Raytraced Shadows haben einen Realismusgewinn pro Kosteneinheit (RK) von 0,6. Dies ist recht wenig, wenn man sich anschaut, dass die weichen Schatten der Shadow Maps mit Werten von  $RK = 3,4$  (250er Schattenmap) und  $RK = 3,0$  (2.000er Schattenmap) mehr als fünfmal so effektiv sind. Einsames Schlusslicht in der Kategorie Schatten sind die Area Shadows. Die hohen Renderzeiten der Area Shadows drücken die Effektivität auf nahezu Null herunter. Mit einem Wert  $RK = 0,04$  ist ihr Realismusgewinn von 56% deutlich vorhanden, aber in Anbetracht der Kosten nahezu ineffektiv.

Zusammengefasst kann man sagen, dass die Shadow Maps den meisten Realismusgewinn vorweisen können und in Anbetracht der Renderzeiten auch am effektivsten sind. Raytraced Shadows bringen einen Realismusgewinn, welcher im Vergleich zu den Shadow Maps aber zu gering ist. Sie brauchen einfach länger in der Berechnung. Dieser Effekt stellte sich auch bei diesem 3D-Charakter ein, wodurch sich die Raytraced Shadows auf einem vorletzten Platz einfinden. Der letzte Platz geht an die physikalisch korrekt berechneten Area Shadows.

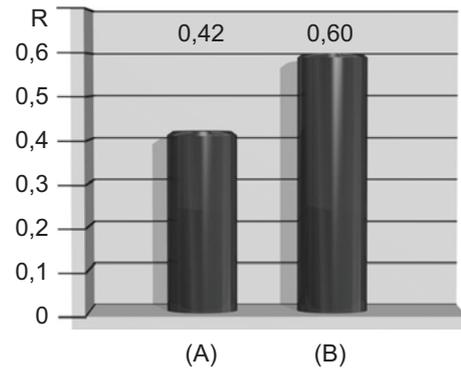
### 7.5.3 Unschärfe und Bildstörungen

Damit Bilder nicht zu klar wirken, können Unschärfen und Bildstörungen integriert werden, wie sie auf alten Fotos zu sehen sind. Dadurch sollte der typisch scharfe und klare Look von computergenerierten Bildern verschwinden und das Bild realistischer wirken. Um diese Annahme zu überprüfen, wurde ein Bild ohne Effekte gerendert und bei einem zweiten Bild wurden Tiefenunschärfe und zusätzlich Störungen integriert.

Die Testpersonen werteten das Bild ohne Effekte mit einem Realismusfaktor  $R = 0,42$ . Das Bild mit Tiefenunschärfe und Bildstörungen wurde hingegen mit  $R = 0,60$  um 0,18 realistischer bewertet. Dies entspricht einer Erhöhung von 42 Prozent. Die o.g. Annahme wurde folglich bestätigt. Das Hinzufügen von Tiefenunschärfe und Bildstörungen erhöht nachweislich den Realismusfaktor.



**Abb. 7-16: Unterschied Unschärfe+Störungen:** Ausschnitte ohne und mit Tiefenunschärfe und simplen Bildstörungen.



**Abb. 7-17: Der Realismusfaktor in Abhängigkeit von Unschärfe+Bildstörungen.**

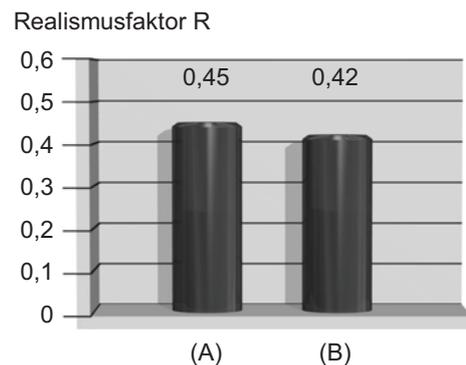
### 7.5.4 Antialiasing

Antialiasing ist mittlerweile Standard bei Computergrafiken. Gerade Linien sollen als gerade Linien dargestellt werden und nicht die Pixel in Form von Treppen sichtbar werden. Das Experiment soll zeigen, welche Wirkung ein Antialiasing bei Grafiken mit virtuellen Charakteren hat. Hierfür wurden den Testpersonen zwei Bilder gezeigt. Einmal mit Aliase, also mit Pixeltreppen und ein zweites Bild mit Antialiasing, also ohne erkennbare Treppen. Die Bilder hatten die Größe von 554 x 500 Pixel und damit eine Auflösung bei der die Pixel bei einem deaktivierten Antialiasing erkennbar sein sollten.

Das Experiment ergab für das Bild ohne Antialiasing einen Realismusfaktor von  $R = 0,45$ . Das Bild mit Antialiasing wurde mit  $R = 0,42$  um 0,03 unrealistischer bewertet. Dies entspricht einer geringfügigen Änderung von knapp -7 Prozent.



**Abb. 7-18: Unterschied Antialiasing:** Ausschnitte aus den gezeigten Bildern ohne und mit aktiviertem Antialiasing.



**Abb. 7-19: Der Realismusfaktor in Abhängigkeit vom Antialiasing.**

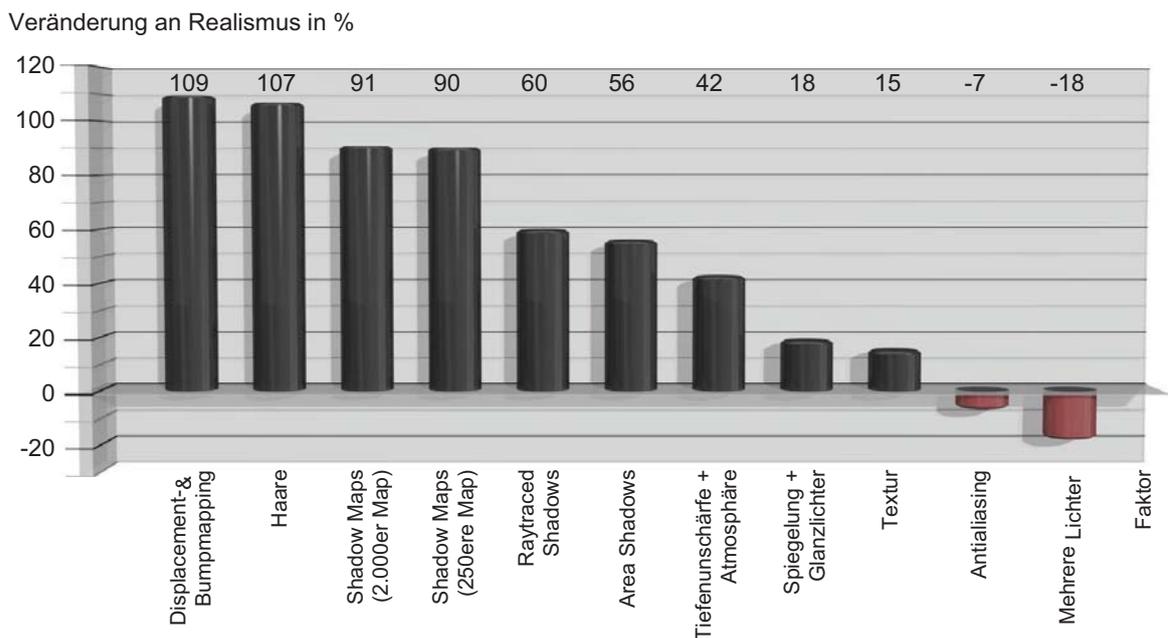
Es scheint, dass es bei virtuellen Charakteren keine Vorteile eines Antialiasing gibt. Dies scheint in unserer heutigen Welt etwas verwirrend zu erscheinen. Fast überall wird versucht, ein Antialiasing zu benutzen, wird versucht die Bildauflösung in unvorstellbare Höhen zu schrauben, um die Grafiken und

Texte auf dem Computer, Handy oder Tablet noch klarer und schärfer darzustellen. Warum erscheint dann ein Bild mit Antialiasing nicht besser als ohne? Unregelmäßigkeiten erscheinen je nach Stärke realistischer als klare Linien in einem Gesicht von virtuellen Charakteren. Diese Erkenntnis zeigte schon das Hinzufügen von Unschärfe und Störungen.

## 7.6 Vergleich

Abschließend soll geklärt werden, welche der Funktionen unbedingt bei der Erstellung realistischer Charaktere verwendet werden sollten und welche vernachlässigt werden könnten, welche der Funktionen am effektivsten sind und welche nicht. Um die einzelnen Funktionen miteinander zu vergleichen, schaut man sich zunächst den prozentuellen Realismuszugewinn an.

Dabei ist zu erkennen, dass man mit Details den größten Realismuszugewinn erzielt (vgl. Abb. 7-20). Das Hinzufügen von kleineren Strukturen wie Falten und Hautporen (durch ein Displacement- und Bumpmapping), aber auch von Haaren, bringt eine Erhöhung des Realismuszugewinns um jeweils mehr als 100 Prozent. Zu glatte und perfekte Oberflächen wirken unrealistisch und computergeneriert, was auch auf virtuelle Charaktere zutrifft. Je mehr Unregelmäßigkeiten, desto realistischer.



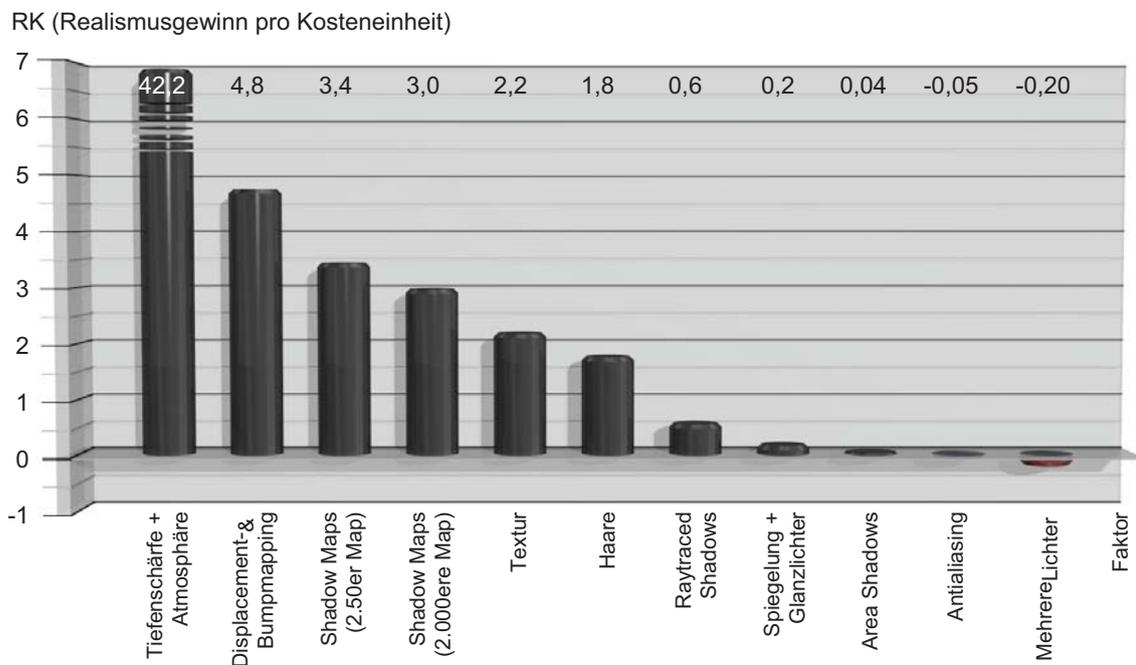
**Abbildung 7-20: Realismuszugewinn im Vergleich:** Kleine Details durch Displacement- und Bumpmaps bringen den größten Realismuszugewinn.

Shadow Maps bringen ebenfalls einen hohen Realismuszugewinn von knapp über 90 Prozent. Raytraced Shadows und Area Shadows schneiden wie bereits erwähnt schlechter ab. Das Hinzufügen von Tiefenunschärfe und Bildstörungen schneidet mit einem Gewinn von 42 Prozent zwar noch gut ab, landet aber im hinteren Mittelfeld. Das Hinzufügen von Spiegelung und Glanz sowie von Details durch Texturen bringt zwar einen erkennbaren, aber dennoch geringen Realismuszugewinn. Ein Antialiasing

und das Hinzufügen von vielen Lichtern bringen keine Erhöhung des Realismusfaktors. Demzufolge können diese beiden Varianten bei der Erstellung von virtuellen Charakteren vernachlässigt werden.

Die Reihenfolge in Abbildung 7-20 soll bei der Planung und dem Aufbau einer realistischen 3D-Figur helfen die Prioritäten richtig zu setzen, um das Optimum zu erzielen. Dabei sind aber noch keine Kosten (Renderzeiten) berücksichtigt. Es geht bisher ausschließlich um den Wirkungsgrad der ausgewählten Funktionen.

Um die Funktionen inklusive Kosten miteinander zu vergleichen, muss man, wie es schon im Kapitel 7.6.2 (Schatten) beschrieben wurde, den Realismusgewinn  $\Delta R$  durch die Anzahl Kosteneinheiten teilen. Als Ergebnis bekommt man den Wert RK, der aussagt, wie viel Realismusgewinn man pro einprozentiger Renderzeiterhöhung bekommt. Dieser Wert gibt also an, wie effizient eine Funktion ist. Schaut man sich nun diese Werte der einzelnen Funktionen an (vgl. Abb. 7-21), so fallen die Ergebnisse drastischer aus, als die Werte der Realismusveränderung separat.



**Abbildung 7-21: Realismusgewinn der einzelnen Faktoren in Bezug auf ihre Kosten:** Das Hinzufügen von Tiefenunschärfe+Athmosphäre ist mit Abstand am kosteneffizientesten.

Das Hinzufügen von Tiefenunschärfe & Bildstörungen bringt einen Realismusgewinn pro Kosteneinheit (RK) von 42,2. Dieser Wert ist im Vergleich zu allen anderen Funktionen um ein Vielfaches höher. Für die Tiefenunschärfe wird nach dem eigentlichen Rendern eine Tiefenmap erstellt, was nur wenige Sekunden dauert. Anschließend wird ausgehend von dieser Tiefenmap die Unschärfe über das Bild gelegt, gefolgt von einigen Bildstörungen, die auch einfach über das Bild gelegt werden. Dieser Effekt ist also nichts weiter als ein einfacher Posteffekt, der nicht lange dauert. Durch seine Schnelligkeit und seine vernünftige Wirkung ergibt sich ein sehr effektiver Effekt, der bei

jeder Charaktererstellung integriert werden sollte, um den Realismus zu erhöhen.

Die Verwendung von Displacement- & Bumpmaps ist mit einem RK = 4,8 nicht nur sehr wirkungsvoll in Bezug auf einen Realismusergebnis, sondern auch in Anbetracht der Renderzeiten sehr effektiv. Auch die weichen Schatten der Shadow Maps sind mit einem RK = 3,4 (250er Map) und einem RK = 3,4 (2.000er Map) wieder im vorderen Feld zu finden. Die Verwendung von Texturen ist aufgrund seiner Schnelligkeit beim Rendern mit einem RK = 2,2 weiter vorne zu finden, während die Benutzung von Haaren durch dessen relativ lange Renderzeiten einen Wert von 1,8 erreichen und damit in das Mittelfeld der Effektivität abrutscht. Harte Schatten der Raytraced Shadows haben mit RK = 0,6 schon einen sehr geringen Wert. Spiegelung & Glanz sollte mit seinem sehr geringen Wert von RK = 0,2 nur in individueller Abwägung der vorhandenen Rechenpower Verwendung finden. Die verbliebenen drei Funktionen Area Shadows, Antialiasing und die Verwendung von mehreren Lichtern können mit Werten RK unter 0,1 als nicht mehr effektiv benannt werden. Während die Area Shadows zu lang zum Rendern brauchen um effektiv zu sein, waren das Antialiasing und die Verwendung von mehreren Lichtern vorher schon nicht wirkungsvoll genug, um effektiv zu sein.

## 7.7 Schlusswort

Diese Untersuchung sollte aufzeigen, dass das Realismusergebnis individuell unterschiedlich und von der Erfahrung einer jeden Person abhängig ist. Daraus soll ersichtlich werden, dass ein Realismusergebnis nicht global durch einfache Bildanalyse berechnet werden kann. Obwohl es in unserer Welt glatte, perfekte Objekte gibt lehrt uns unsere Erfahrung, dass strukturierte und weniger perfekte Objekte alltäglicher sind, wodurch sie auch als realistischer wahrgenommen werden.

Weiterhin sollte aufgezeigt werden, dass der empfundene Realismusergebniswert, der Realismusergebnisfaktor, dennoch erfassbar ist. Die unterschiedlichen Funktionen die es gibt, um einen menschlichen 3D-Charakter aufzubauen, haben unterschiedliche Wirkungen auf den Realismusergebnisfaktor des entstandenen Bildes. Es nutzt wenig, wenn ein Charakter-Designer sehr viel Zeit einsetzt, um eine bestimmte Stelle herauszuarbeiten, die zum Schluss kaum einen Realismusergebnisgewinn bringt. Hier sollten Prioritäten gesetzt werden. Wenn Firmen dies bei ihrer Erstellung virtueller Charaktere verstehen und beginnen diese Erstellung hinsichtlich der Wirkung und der Kosteneffizienz der Funktionen zu planen, lassen sich Zeit und Geld sparen.

Zusätzlich sollten die Algorithmen der Renderprogramme hinsichtlich der Wirkung der einzelnen Funktionen optimiert werden. Eine immer weitergehende Nachahmung der physikalischen Welt scheint nachvollziehbar zu sein. Es ist aber nicht immer wirkungsvoll, wie diese Untersuchung zum Beispiel bei den Schatten gezeigt hat. Physikalisch korrekt arbeitende Schatten sind nicht realistischer als die schneller berechenbaren weichen Schatten der Shadow Maps. Dieser Umstand sollte bedacht werden.

Im Detail zeigte die Untersuchung vor allem eines: Je mehr Details bei einem virtuellen Charakter zu sehen sind, desto realistischer wirkt er. Je mehr kleine Falten, Hautporen, Leberflecke, Haare er hat, desto realistischer wird er wahrgenommen. Details, die zeigen, dass die Haut nicht glatt und nicht

perfekt ist, sind vorteilhaft. Was für den Charakter selber gilt, lässt sich auch auf dessen Umgebung anwenden: Störungen wie Unschärfe haben mehr Wirkungskraft als die Veränderung bei der Anzahl der Lichter, die keine Wirkung zeigt. Bei der Nachahmung realistischer menschlicher Charaktere sollte man das bedenken, was uns unsere Erfahrung schon längst gelehrt hat:

*Die Welt ist nicht perfekt!*

## 8 Ausblick

Realismus ist, wie in dieser Arbeit ersichtlich wurde, ein sehr breit gefächertes Thema. Es gibt sehr viele verschiedene Techniken, die angewendet zu einem realistischen menschlichen Charakter führen können. Aufgrund der hohen Anzahl an möglichen Modellierungstechniken wurde nur ein kurzer Einblick in die jeweilige Techniken gegeben und hinsichtlich der Realismuswirkung bei einem virtuellen Charakter untersucht. Für tiefergehende Informationen wird hier auf weiterführende Literatur verweisen.

### 8.1 Realismusuntersuchung

Wie die vorliegende Arbeit zudem gezeigt hat, führen manche Techniken zu einer größeren realistischen Wirkung als andere. Zusätzlich sollte immer auf den Speicherverbrauch und letztendlich auf die Renderzeit geachtet werden. Nur Techniken, die hinsichtlich dieser Ausnutzung gut arbeiten, können auch als effektiv bezeichnet werden und sollten bei einer menschlichen Charaktererstellung nicht fehlen.

Die Untersuchung<sup>37</sup> hinsichtlich Realismuswirkung und der Effektivität sollte aufweisen, dass dem Realismusfaktor eine messbare und damit greifbare Größe zugeordnet werden kann, um die einzelnen Techniken besser vergleichen zu können. Ziel ist die bessere Planbarkeit bei der Erstellung realistischer, virtueller Charaktere. Aufgrund der enormen Anzahl an einzusetzenden Techniken wurden nur einige hiervon genauer untersucht und bewertet. Diese Arbeit erhebt dadurch nicht den Anspruch auf Vollständigkeit, sondern soll nur als Anregung verstanden werden weitere Modellierungstechniken auf gleiche Weise zu untersuchen und auszuwerten.

### 8.2 Realismus bei Animationen

Untersucht wurden in dieser Arbeit nur virtuelle Charaktere in festen Posen. Menschen realistisch zu animieren stellt ein umfangreiches weiteres Themengebiet dar. Menschen glaubwürdig in Bewegung zu zeigen ist eines der größten Schwierigkeiten bei CG-Filmen. Leichter zu animieren sind dagegen Figuren wie die Dinosaurier aus *Jurassic Parc* (1993, Universal Pictures) oder die großen Monster aus der *Herr der Ringe*-Trilogie (2001-2003, New Line Cinema). [Tet]

»Wer weiß schon, wie Dinos wirklich gelaufen sind? Solange die Dinger sich flüssig bewegen und sie ein gewisses Körpergewicht simulieren, wirkt die Animation glaubwürdig.« [Bill Diaz<sup>38</sup>]

Wir Menschen sehen hingegen tagtäglich andere Menschen. Wir sind dadurch sehr gut auf deren Bewegungsmuster trainiert. Wir merken sehr schnell, ob sich ein computergenerierter Mensch auch

---

<sup>37</sup> vgl. Kapitel 7 Realismus

<sup>38</sup> Bill Diaz war zuständiger *CG Character Animation Lead* bei *Spider-Man*.

wirklich wie ein Mensch bewegt. Welche Techniken hierfür die realistischsten Ergebnisse liefern, könnte in einer weiterführenden Untersuchung analog zu der hier eingeführten Variante durchgeführt werden. Ähnlich könnte die Wirkung erfasst und anschließend die Effektivität berechnet werden.

Die Bewegungen von *Spider-Man* (2002, Marvel / Columbia Pictures) waren eine Herausforderung, da sich Spider-Man wie ein Superheld bewegen sollte, aber dennoch die Bewegungen menschlich wirken sollten. Eine Hohe Bedeutung kommt der Gewichtung der einzelnen Knochen der animierten Figur zu. Mit wenig Gewichtung wirkt eine Figur schnell und wendig, während eine hohe Gewichtung eher Größe und Behäbigkeit vermittelt. Im Film wurde Spider-Man in den meisten Szenen in diesem Hinblick gut animiert. Allerdings gibt es auch Szenen, zum Beispiel die Kampfszenen mit den Verbrechern, in denen Spider-Man eher wie ein Springball umherfliegt. Etwas mehr Beweglichkeit und etwas langsamere Bewegungen hätten hier geholfen. [Tet]

In der Vorproduktionsphase wurden bei Spider-Man verschiedene Techniken durchgetestet, um eine glaubwürdige Animation zu erhalten [Tet]. Welche Techniken bei Animationen die besten Ergebnisse liefern ist allerdings situationsabhängig. Bei Spider-Man entschied man sich für eine einfache Key-Frame-Animation und verzichtete auf eine Verwendung von Motion Capturing. Normalerweise wird bei Filmproduktionen lieber auf Motion Capturing gesetzt, da diese auf aufgenommenen menschliche Bewegungen basieren und meist weniger künstlich erscheinen, als das von Hand animieren bei der Key-Frame-Animation. Die Key-Frame-Animationen lassen allerdings mehr Freiheiten zu und wurden auch dadurch den Motion Capturing vorgezogen [Tet].

Welche Techniken bei Animationen Verwendung finden ist situationsabhängig.  
Diese Aussage sollte bei Animationen wie auch bei Stillbildern trotz aller  
Erkenntnisse Beachtung finden.

## A Referenzen

Literaturverweise bestehen aus einem Kürzel, welche in eckigen Klammern aus den ersten Buchstaben des Autors bestehen. Wird auf eine bestimmte Stelle eines Buches verwiesen, wird die Seitenzahl durch ein Komma nach dem Autorenkürzel dargestellt.

- [Asa] Andreas Asanger: *“Cinema 4D 10 – Das Praxisbuch zum Lernen und Nachschlagen”*, Galileo Design, Bonn, 2007
- [Bl] *„BI Handlexikon in Zwei Bänden“*, Herausgegeben von der Lexikonredaktion des VEB Bibliographisches Institut, Leipzig, 1983
- [Bla] Wendon Blake: *„Aktzeichen mit Bleistift, Kreide und Kohle”*, Urania-Ravensburger, Berlin, 1998
- [Bir] Jeremy Birn: *„Lightning & Rendering“*, Addison-Wesley, München, 2007
- [Bru] Vicki Bruce, Patrick Green, Mark Georgeson: *„Visual Perception: Physiology, Psychology, and Ecology”*, Psychology Press, 1996
- [Byl] Leigh van der Byl: *„Size Does Count. Answering that eternal question of ‘What size must my textures be?’”*, 2005
- [Cha] Alen Chalmers, Scott Daly, Ann McNamara, Karol Myszkowski, Tom Troscianko: *„Image Quality Metrics”*, SIGGRAPH, 2000
- [Cri] Per Christensen, Pixar Animation Studios: *„Ray tracing for the movie ‘Cars’”*, Ray Tracing Symposium 2006
- [Deb] Chris Debski: *“Cinema 4D 8 Charakteranimation – Prinzipien und ihre Anwendung”*, Galileo Design, Bonn, 2003
- [Debe] Paul Debevec, Tim Hawkins, Chris Tchou, H.P. Duiker, Westley Sarokin and Mark Sagar: *„Acquiring the Reflectance Field of a Human Face”*, SIGGRAPH, 2000
- [Der] Tony DeRose, Michael Kass, Tien Truong: *„Subdivision Surfaces in Character Animation“*, Pixar Animation Studios, 1998

- [Eck] Maik Eckardt: „*Cinema 4D R12 Prime und Studio*“, mitp-Verlag, 2010
- [Hel] „*Helden aus dem Computer – Shrek*“, kein Autor, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“; Ausgabe 02/01, ACT GmbH, Pullach, 2001
- [Hen] Alexander Henning: „*Die andere Wirklichkeit – Virtuals Reality – Konzepte, Standards, Lösungen*“, Addison-Wesley Longman Verlag GmbH, Bonn, 1997
- [Ken] Steven L. Kent, David Waybright [HRSG]: „*The Making of Final Fantasy: The Spirits Within*“, Brady Games, Indianapolis, USA 2001
- [Hyp] „*Hyperrealistische Fantasie*“, kein Autor, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“; Ausgabe 03/01, ACT GmbH, Pullach, 2001
- [Ker] Marion Kerstholt: „*Andere Länder - Andere Größen*“, [http://www.wdr.de/tv/quarks/sendungsbeitraege/2006/0124/006\\_din.jsp](http://www.wdr.de/tv/quarks/sendungsbeitraege/2006/0124/006_din.jsp) (vom 24.01.2008)
- [Kul] „*Das Raytracing Prinzip*“, Kunst Universität Linz, <http://www.dma.ufg.ac.at/app/link/Grundlagen%3A3D-Grafik/module/8653> (vom 30.10.12)
- [Kün] Dr. Med. Dolf Künzel: „*Der menschliche Organismus*“, VEB Verlag Volk und Gesundheit, Berlin, 1983
- [Mcn] Ann McNamara, Alan Chalmers, Tom Troscianko, Iain Gilchrist: „*Comparing Real & Synthetic Scenes using Human Judgement of Lightness*“, University of Bristol, 2000
- [Max] „*Cinema 4D R9 Referenz-Handbuch*“, Maxon Computer GmbH, Friedrichsdorf, 2004
- [Mey] „*Myers Universal Lexikon*“ in vier Bänden, Herausgegeben von der Lexikonredaktion des VEB Bibliographisches Institut, Leipzig, 1979
- [Meye] Meyers Lexikon Online, <http://lexikon.meyers.de/meyers/> (vom 24.01.2008)
- [Mor] Masahiro Mori: „*The Uncanny Valley*“ Translated by Karl F. MacDorman and Takashi Minato, Energy, 1970
- [Nas] Nasa (National Aeronautics and Space Administration) Image Gallery: „*Tycho Star Map*“, [http://www.nasa.gov/multimedia/3d\\_resources/assets/tycho8.html](http://www.nasa.gov/multimedia/3d_resources/assets/tycho8.html) (vom 09.08.2010)

- [Olm] Pablo Olmos: „*Virtuelle Charaktere mit 3ds max*“, Galileo Press, Bonn, 2004
- [Par] Frederic I. Parke, Keith Waters: „*Computer Facial Animation*“, A K Peters, Wellesley, Massachusetts, 1996
- [Rad] Paul Rademacher, Jed Lengyel, Edward Cutrell, Turner Whitted: „*Measuring the Perception of Visual Realism in Images*“, University of North Carolina at Chapel Hill, Microsoft Research, 2001
- [Riz] Samad Rizvi: „*Ed Catmull's 'Computer Animated Hand' Added To National Film Registry*“, <http://www.pixartimes.com/2011/12/28/ed-catmulls-computer-animated-hand-added-to-national-film-registry>, 2011
- [Rob4] Barbara Robertson: „*Spinne im freien Fall*“, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“, Ausgabe 05/04, ACT GmbH, Pullach, 2004
- [Rob5] Barbara Robertson: „*Star Wars im Superlativ*“, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“, Ausgabe 04/05, ACT GmbH, Pullach, 2005
- [Rob6] Barbara Robertson: „*Mocap-Piraten und Tentakel*“, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“, Ausgabe 05/06, ACT GmbH, Pullach, 2006
- [Ros] Veit Rosenberger: „*Der Herr der Ringe – wie Gollum entstand*“, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“, Ausgabe 06/03, ACT GmbH, Pullach, 2003
- [Rus] Holly Rushmeier, G. Larson, C. Piatko, P. Sanders, B. Rust: „*Comparing Real and Synthetic Images: Some Ideas About Metrics*“, Proc of Eurographics Rendering Workshop 1995, SpringerWien, New York, 1995
- [Sch] Daniel Schilling: „*Ein Geist aus Shadern und Blendshapes*“, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“, Ausgabe 05/06, ACT GmbH, Pullach, 2006
- [Sim] Sarah Simblet: „*Anatomy for the Artist*“, Dorling Dindersley Limited, London, 2001
- [Sut] Ivan Edward Sutherland: „*Sketchpad: A man-machine graphical communications system.*“, *Technical Report 296, MIT Lincoln Laboratories*. 1963, 2003

- [Sto] Jon Stokes, Jonathan Ragan-Kelley: „*Final Fantasy: The Technology Within*“, Ars Technical, 2001
- [Tet] Tülay Tetiker: „*Spider-Man*“, in „*Digital Production – Das Fachmagazin für Computergrafik, Video und Film*“; Ausgabe 03/02, ACT GmbH, Pullach, 2002
- [Tho] Randy Thornhill, Steven W. Gangestad: „*Facial attractiveness*“ in „*Trends in Cognitive Sciences, Vol. 3*“, 1999
- [Wad] Daniel Wade [HRSG], Paul Hellard [HRSG]: „*D’Artiste: Character Modeling*“, Ballistic Publishing, Australia, 2005
- [Wol] Ellen Wolff: „*RenderMan@20: Ed Catmull and Dana Batali Reflect On Pixar’s Killer App*“, VFX World Magazine, August 2008

## B Tabellen- und Abbildungsverzeichnis

Abbildungen von anderen Autoren wurden als solche kenntlich gemacht. Ist keine Quellenangabe angegeben, so wurden die die Abbildungen vom Autor dieser Arbeit erstellt.

Tabelle 7-1: „Kostenvergleich“

---

Abbildung 2-1: „Proportionen des Menschen“  
aus [Bla]

Abbildung 2-2: „Kopfproportionen“  
aus [Bla]

Abbildung 2-3: „Gesichtsmuskeln“  
aus [Sim]

Abbildung 2-4: „Kopfvorlagen“

Abbildung 2-5: „Körpervorlagen“

Abbildung 3-1: „Topologieplanung“

Abbildung 3-2: „Polygonaufbau der Kopfhälfte“

Abbildung 3-3: „Symmetrie bei Kopfdarstellungen“

Abbildung 3-4: „Phong-Shading“

Abbildung 3-5: „Displacementmap“

Abbildung 3-6: „Geometrie“

Abbildung 4-1: „Standardprojektionsarten“

Abbildung 4-2: „UV-Map und entsprechende UV-Projektion“

Abbildung 4-3: „Farbtextur“

Abbildung 4-4: „Glanztextur“

Abbildung 4-5: „Relieftextur“

Abbildung 4-6: „Lichtreflektionen der Haut“,  
aus [Debe]

Abbildung 4-7: „Subsurface Scattering“

Abbildung 4-7: „Angewendete Texturen“

Abbildung 5-1: „Haarguides“

Abbildung 5-2: „Texturen zum Steuern der Haarverteilung“

Abbildung 5-3: „Haarsimulation“

Abbildung 5-4: „Haare in Resistance 3“,  
Screenshot aus Resistance 3 (2011, *Insomniac Games, Playstation 3*)

Abbildung 6-1: „Raytracingverfahren“,  
Aufbau nach [Kul]

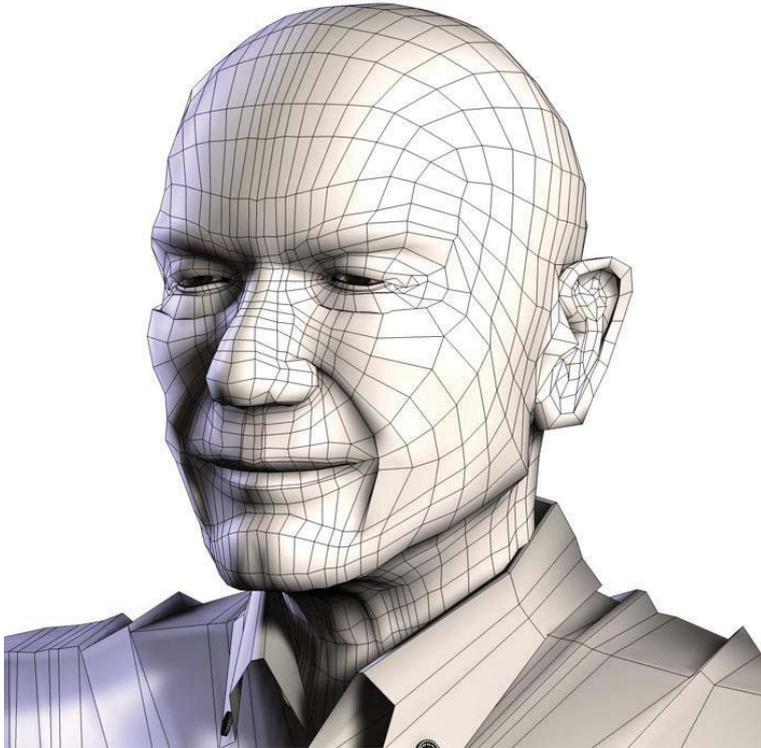
---

Abbildung 6-2:	„Unterschiedliche Brennweiten“
Abbildung 6-3:	„Angewendete Lichter“
Abbildung 6-4:	„Lichtpositionierungen“
Abbildung 6-5:	„Schattenübergang“
Abbildung 6-6:	„Shadow-Maps“
Abbildung 6-7:	„Raytraced Shadows“
Abbildung 6-8:	„Ambient Occlusion“
Abbildung 6-9:	„Schattenarten im Vergleich“
Abbildung 6-10:	„Umgebungstextur“
Abbildung 6-11:	„Tiefenunschärfe“
Abbildung 6-12:	„Partikeleffekt Resistance 3“, Screenshot aus Resistance 3 (2011, <i>Insomniac Games, Playstation 3</i> )
Abbildung 6-13:	„Bildstörungen“
Abbildung 7-1:	„Abhängigkeit des Realismusfaktors R im Bezug auf das Alter“
Abbildung 7-2:	„Abhängigkeit des Realismus-faktors R und der Computer-spielerfahrung GE“
Abbildung 7-3:	„Unterschied Farbtextur“
Abbildung 7-4:	„Der Realismusfaktor in Abhängigkeit von einer Farbtextur“
Abbildung 7-5:	„Unterschiede Glanz+Spiegelung“
Abbildung 7-6:	„Der Realismusfaktor in Abhängigkeit von Glanz+Spiegelung“
Abbildung 7-7:	„Unterschiede Displacement+Bump“
Abbildung 7-8:	„Der Realismusfaktor in Abhängigkeit von Displacement+Bump“
Abbildung 7-9:	„Unterschied Haare“
Abbildung 7-10:	„Der Realismusfaktor in Abhängigkeit von Haaren“
Abbildung 7-11:	„Unterschied Licht“
Abbildung 7-12:	„Der Realismusfaktor in Abhängigkeit von der Anzahl an Lichtern“
Abbildung 7-13:	„Ausschnitte der Kategorie Schatten“
Abbildung 7-14:	„Realismusfaktoren der Schatten“
Abbildung 7-15:	„Realismusgewinn mit Kosteneinbeziehung aller Schatten“
Abbildung 7-16:	„Unterschied Unschärfe+Störungen“
Abbildung 7-17:	„Der Realismusfaktor in Abhängigkeit von Un-schärfe+Bildstörung“
Abbildung 7-18:	„Unterschied Antialiasing“
Abbildung 7-19:	„Der Realismusfaktor in Abhängigkeit vom Antialiasing“
Abbildung 7-20:	„Realismusgewinn im Vergleich“
Abbildung 7-21:	„Realismusgewinn der einzelnen Faktoren in Bezug auf ihre Kosten“

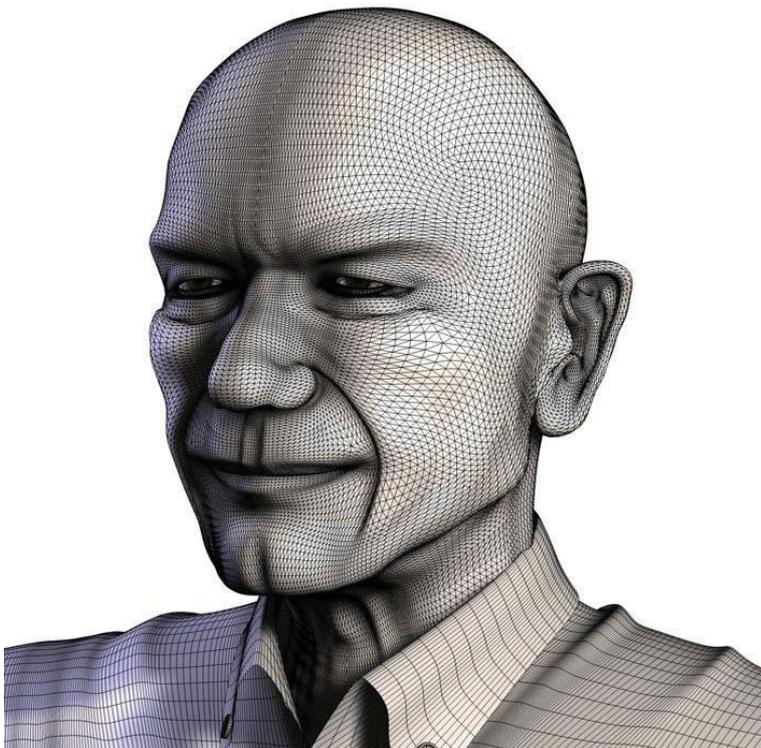
## C Abkürzungen

In dieser Diplomarbeit werden verschiedene Abkürzungen verwendet. Die gebräuchlichsten Abkürzungen sind hier aufgelistet:

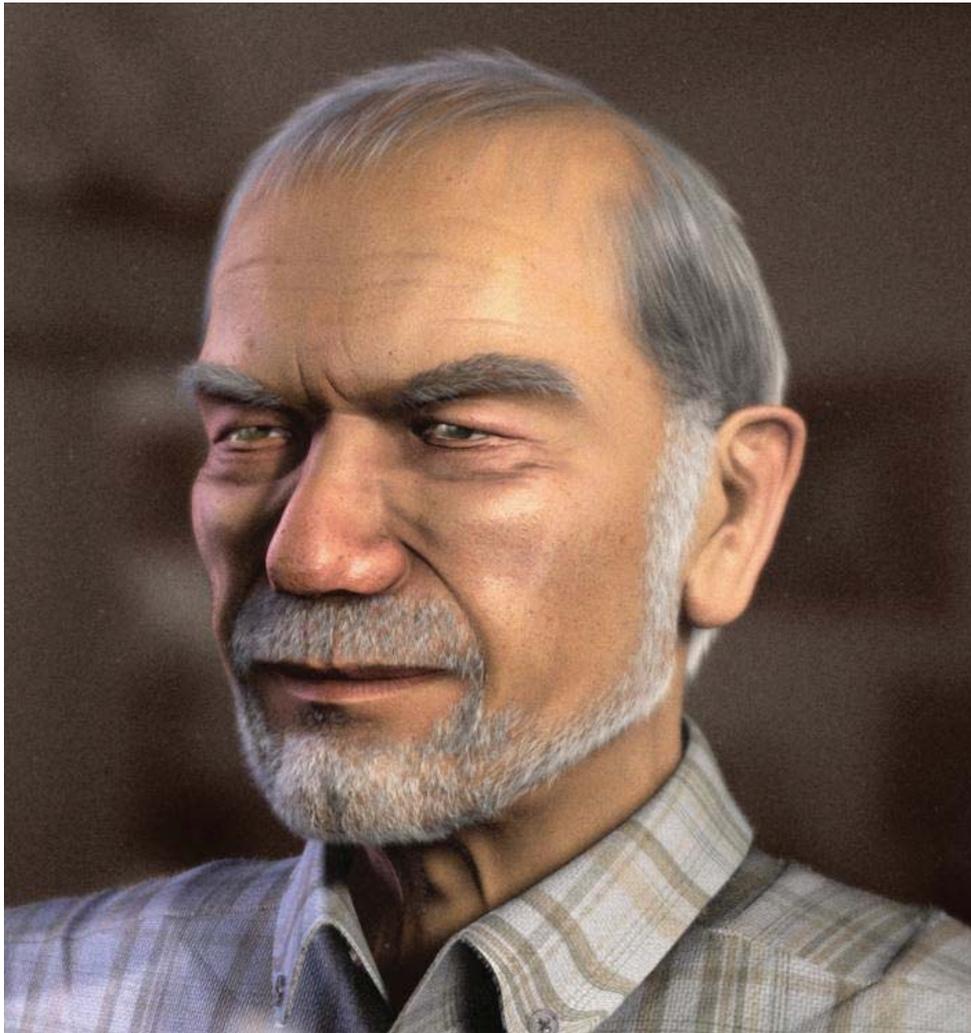
AO	Ambient Occlusion
CAD	Computer Aided Design
CG	Computer Generated
CGI	Computer Generated Images
GE	GameExperience / Computerspielerfahrung
GI	Globale Illumination
HDRI	High Dynamic Rang Images
Px	Pixel
R	Realismusfaktor/-wirkung
REYES	Rendering Everything You Ever Saw
RK	Realismugewinn pro Kosteneinheit
SPD	Subpolygon Displacement
SSS	Subsurface-Scattering

**D Anhang**

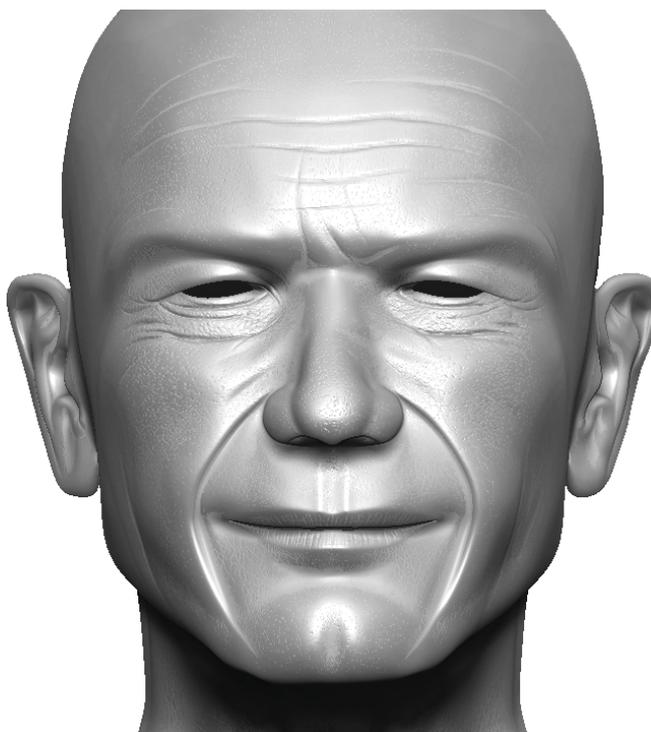
**Abbildung D-1:**  
Grundgeometrie des  
Kopfes



**Abbildung D-2:**  
Geometrie mit 2fach-  
unterteiltem Displacement



**Abbildung D-3:**  
Finales Rendering



**Abbildung D-4:**  
Finale Geometrie des  
Kopfes (4fach unter-  
teiltes Displacement)



**Abbildung D-5:**  
Finale Geometrie des  
Hemdes



**Abbildung D-6:**  
Finale Geometrie der  
Hand

Statistische Auswertung der Umfrage von: [http://www.meltus.de/char]. Sortiert nach Realismusfaktor.  
Stand: 09.07.2012, 16:07:35

Umfragedaten zw. 2010-09-01 00:00:00 und 2012-06-30 00:00:00.

Umfrageteilnehmer:	Gruppe	Anzahl	Anteil	Ø Real.faktor	Veränd.
	0-100 Jahre	366	100.0 %	0.35	0%
	1-12 Jahre	8	2.2 %	0.54	+31%
	13-50 Jahre	316	86.3 %	0.34	-2%
	51-100 Jahre	39	10.7 %	0.43	+13%
	männl.	168	45.9 %	0.36	
	weibl.	192	52.5 %	0.35	
	CGameExperience	219	100.0 %	0.33	0%
	wenig	57	26.0 %	0.41	+22%
	mittel	87	39.7 %	0.35	+6%
	viel	75	34.2 %	0.25	-24%

Auf 100 Prozent fehlende Anteile: Keine Angabe.

Pos (Nr)	Bild (+Link)	Bildname	Renderzeit (Referenzvergleich)	Nr. # Anzahl
		Alter	# Realismusfaktor (normiert)	[1 = 0.0 (nicht realistisch), 5 = 1.0 (realistisch)]

1. (7)		<b>char_h119_ps.jpg</b> Renderzeit: 302sek (ΔRef: 3sek / 1.0%)		Verteilung: 1: 19 2: 47 3: 114 4: 131 5: 53
		0-100 Jahre	364 <b>0.60</b>	
		1-12 Jahre	8 0.69	
		13-50 Jahre	314 0.60	
		51-100 Jahre	39 0.67	
		R=0.60 (ΔR=0.18/42.4%) RTP=42.23 RefNr: 3		

2. (12)		<b>char_h120_light-one2.jpg</b> Renderzeit: 158sek (ΔRef: 141sek / 89.2%)		Verteilung: 1: 44 2: 71 3: 104 4: 101 5: 42
		0-100 Jahre	362 <b>0.52</b>	
		1-12 Jahre	8 0.59	
		13-50 Jahre	312 0.51	
		51-100 Jahre	39 0.56	
		R=0.52 (ΔR=-0.09/-18.0%) RTP=-0.20 RefNr: 3		

3. (4)		<b>char_h120_alias.jpg</b> Renderzeit: 128sek (ΔRef: 171sek / 133.6%)		Verteilung: 1: 52 2: 94 3: 111 4: 75 5: 29
		0-100 Jahre	361 <b>0.45</b>	
		1-12 Jahre	8 0.53	
		13-50 Jahre	311 0.44	
		51-100 Jahre	39 0.58	
		R=0.45 (ΔR=-0.03/-6.7%) RTP=-0.05 RefNr: 3		

4. (3)		<b>char_h119.jpg</b> Renderzeit: 299sek (ΔRef: 70sek / 30.6%)		Verteilung: 1: 62 2: 88 3: 122 4: 75 5: 14
		0-100 Jahre	361 <b>0.42</b>	
		1-12 Jahre	8 0.50	
		13-50 Jahre	311 0.41	
		51-100 Jahre	39 0.55	
		R=0.42 (ΔR=0.20/90.8%) RTP=2.97 RefNr: 8		

5. (10)		<b>char_h120_shadow-smoother.jpg</b> Renderzeit: 289sek (ΔRef: 60sek / 26.2%)		Verteilung: 1: 56 2: 104 3: 114 4: 71 5: 17
		0-100 Jahre	362 <b>0.42</b>	
		1-12 Jahre	8 0.63	
		13-50 Jahre	312 0.41	
		51-100 Jahre	39 0.52	
		R=0.42 (ΔR=0.20/90.2%) RTP=3.44 RefNr: 8		

6. (9)		<b>char_h120_te.jpg</b> Renderzeit: 280sek (ΔRef: 19sek / 6.8%)		Verteilung: 1: 59 2: 128 3: 128 4: 44 5: 6
		0-100 Jahre	365 <b>0.37</b>	
		1-12 Jahre	8 0.59	
		13-50 Jahre	315 0.35	
		51-100 Jahre	39 0.49	
		R=0.37 (ΔR=0.05/14.8%) RTP=2.18 RefNr: 3		

7. (1)		<b>char_h120_re+sp.jpg</b> Renderzeit: 144sek (ΔRef: 155sek / 107.6%)		Verteilung: 1: 91 2: 111 3: 93 4: 56 5: 15
		0-100 Jahre	366 <b>0.36</b>	
		1-12 Jahre	8 0.56	
		13-50 Jahre	316 0.34	
		51-100 Jahre	39 0.49	
		R=0.36 (ΔR=0.07/18.4%) RTP=0.17 RefNr: 3		

8. (11)		<b>char_h120_shadow-hard.jpg</b> Renderzeit: 475sek (ΔRef: 246sek / 107.4%)		Verteilung: 1: 70 2: 141 3: 88 4: 52 5: 10
		0-100 Jahre	361 <b>0.36</b>	
		1-12 Jahre	8 0.53	
		13-50 Jahre	311 0.33	
		51-100 Jahre	39 0.51	
		R=0.36 (ΔR=0.13/59.7%) RTP=0.56 RefNr: 8		

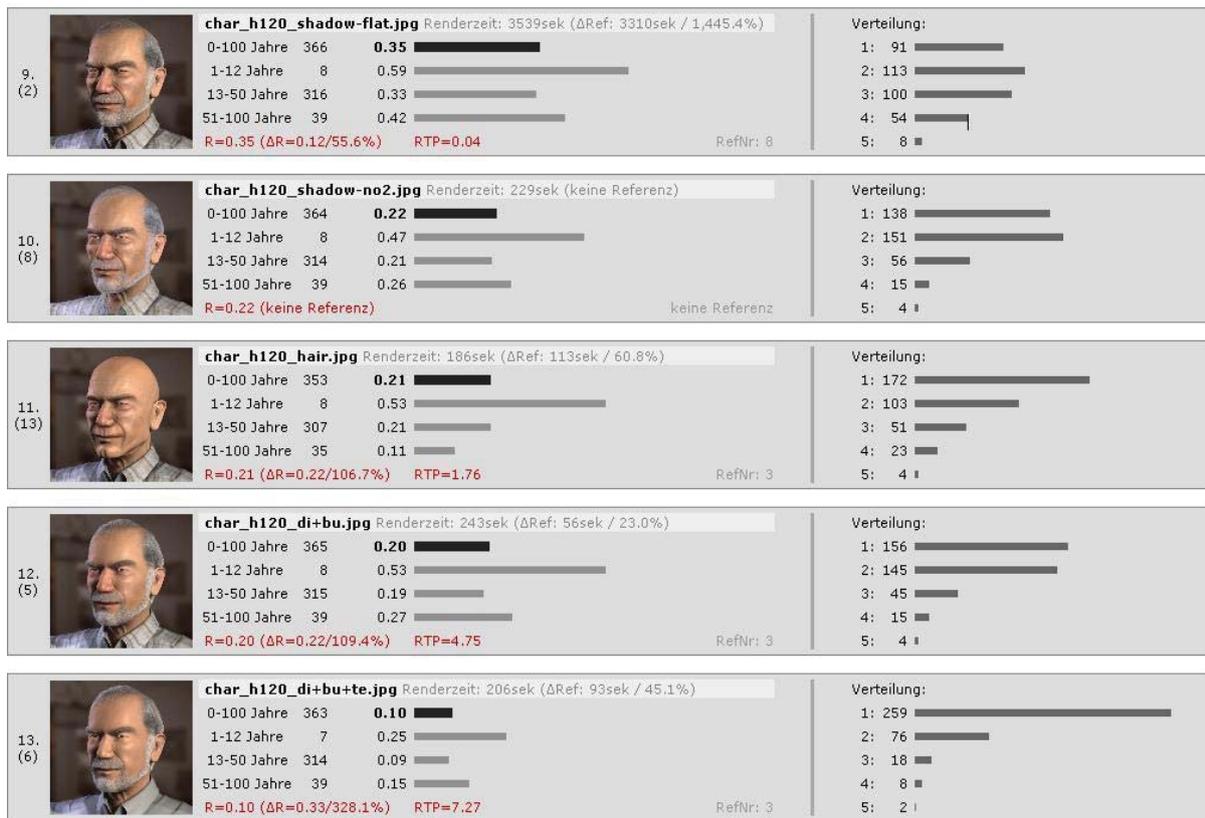
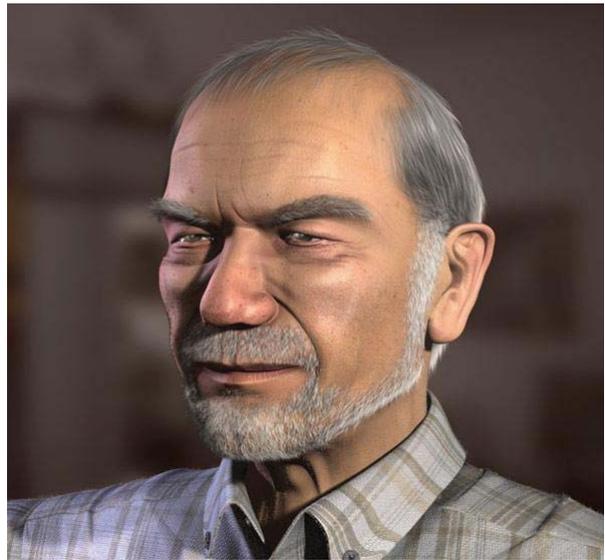


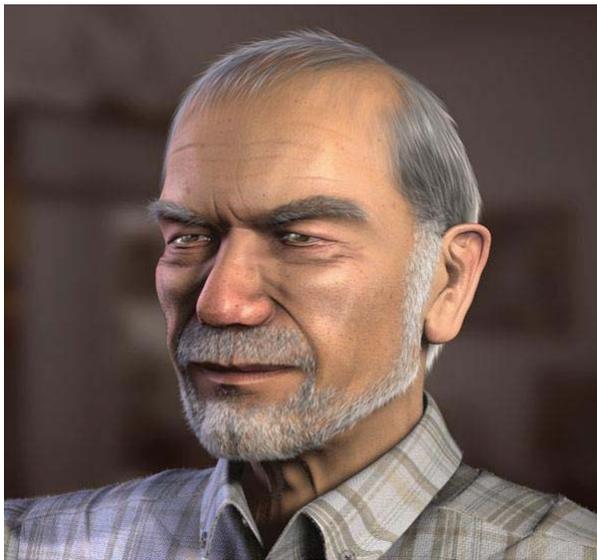
Abbildung D-7: Umfrageauswertung



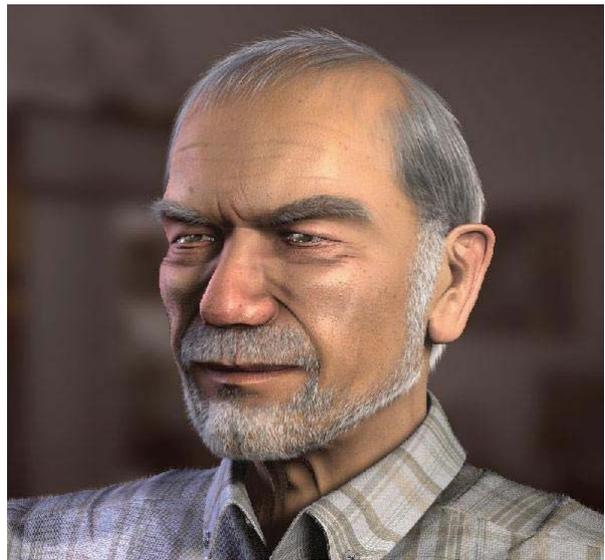
**Abb. D-8:** Umfragebild 1:  
Ohne Specular und Reflection



**Abb. D-9:** Umfragebild 2:  
Schatten: Area Shadows



**Abb. D-10:** Umfragebild 3:  
Schatten: Shadow Maps mit  
2.000er Map



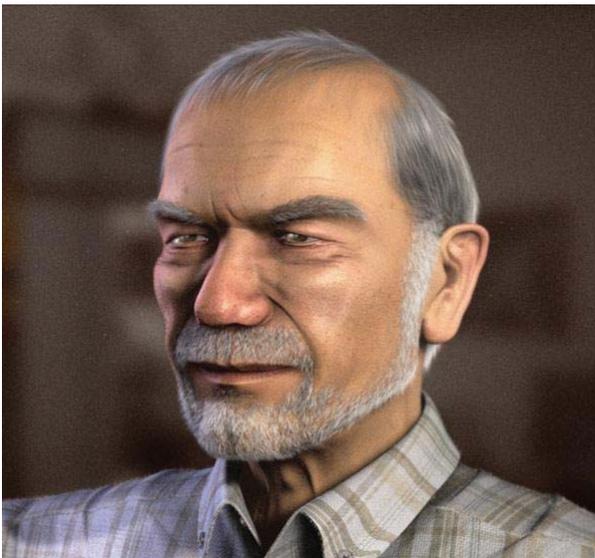
**Abb. D-11:** Umfragebild 4:  
Ohne Antialiasing



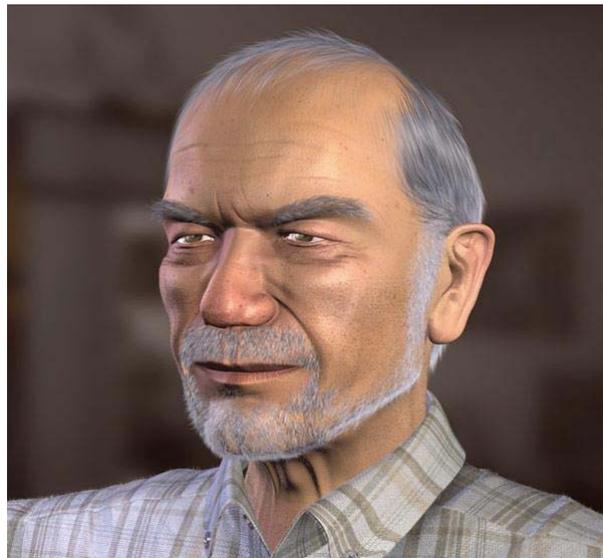
**Abb. D-12:** Umfragebild 5:  
Ohne Relief und Displacement



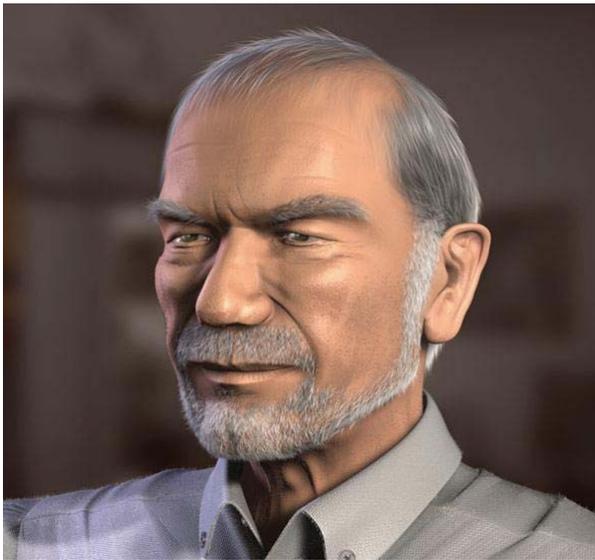
**Abb. D-13:** Umfragebild 6:  
Ohne Relief, Displacement und  
ohne Farbtextur



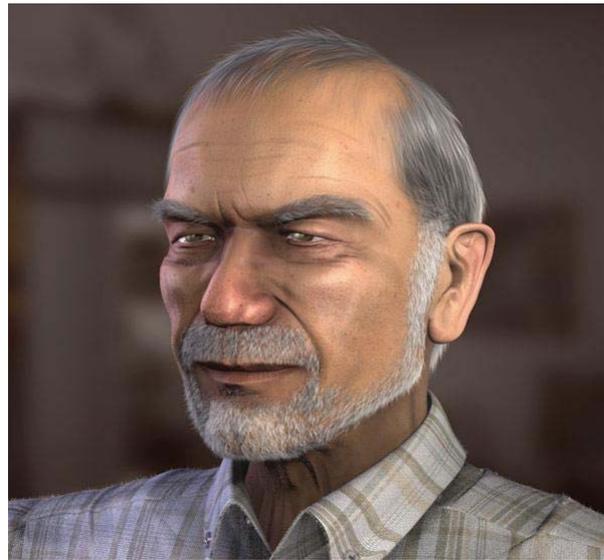
**Abb. D-14:** Umfragebild 7:  
Mit Tiefenunschärfe und Bild-  
störungen



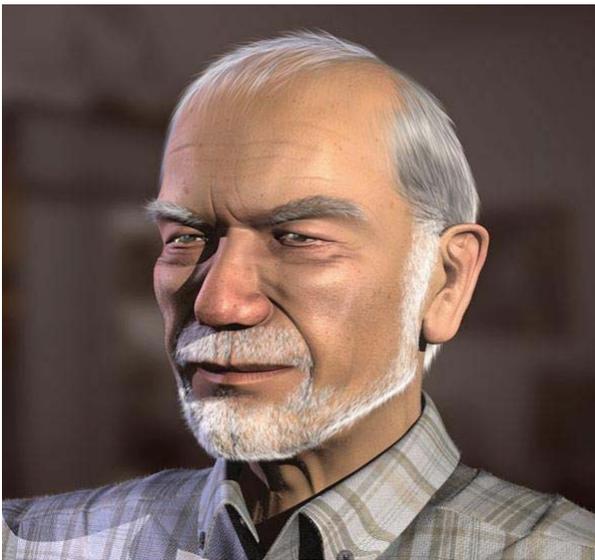
**Abb. D-15:** Umfragebild 8:  
Kein Schatten



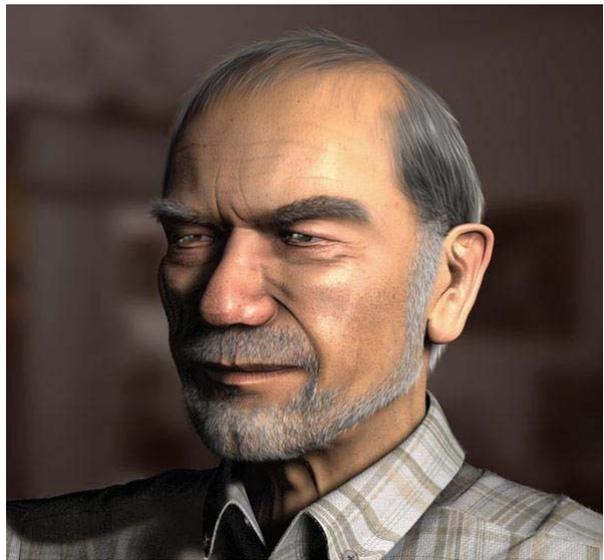
**Abb. D-16:** Umfragebild 9:  
Ohne Textur



**Abb. D-17:** Umfragebild 10:  
Schatten: Shadow-Maps mit  
250er Map



**Abb. D-18:** Umfragebild 11:  
Schatten: Raytraced Shadows



**Abb. D-19:** Umfragebild 12:  
Mit einer Lichtquelle



**Abb. D-20:** Umfragebild 13:  
Ohne Haare

**Selbständigkeitserklärung**

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe. Weiterhin erkläre ich, eine Diplomarbeit in diesem Studiengang erstmalig einzureichen.

Berlin, den 03.12.2012

.....