

# Beleuchtungsmodelle

Proseminar-Ausarbeitung von

**Jonas Lehmann**

An der Fakultät für Informatik  
Institut für Visualisierung und Datenanalyse,  
Lehrstuhl für Computergrafik

18. November 2019

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
<b>2</b>	<b>Grundlagen</b>	<b>3</b>
2.1	Materialeigenschaften . . . . .	3
2.2	Die Rendergleichung . . . . .	4
2.3	Bidirektionale Verteilungsfunktionen . . . . .	6
<b>3</b>	<b>Lokale Beleuchtungsmodelle</b>	<b>8</b>
3.1	Übersicht . . . . .	8
3.2	Phong-Beleuchtungsmodell . . . . .	8
3.3	Blinn-Phong-Beleuchtungsmodell . . . . .	10
3.4	Cook-Torrance-Beleuchtungsmodell . . . . .	11
<b>4</b>	<b>Zusammenfassung</b>	<b>15</b>
4.1	Ergebnisse . . . . .	15
4.2	Ausblick . . . . .	15
	<b>Literaturverzeichnis</b>	<b>16</b>

# Abstract

Für die Darstellung von realitätsnahen Materialien nutzt man in der Computergrafik Verfahren, die die Beleuchtungsvorgänge in einer modellierten Szene nachahmen. Die Berechnung muss aufgrund der Komplexität in eine globale Strahlen- und Lichtwegverfolgung, sowie eine lokale Reflexionsmodellierung aufgeteilt werden.

Global kann Beleuchtung über die Rendergleichung beschrieben werden, an der Oberfläche dann über Bidirektionale Verteilungsfunktionen.

Für solche BRDF's gibt es viele verschiedenen Repräsentationsversuche. Einen davon stellt das Phong-Modell dar, das die Reflexion anschaulich über empirische Größen modelliert. In dieser Ausarbeitung wird zusätzlich aber auch auf das Blinn- und Cook-Torrance-Modell eingegangen, was im Falle des Blinn-Modells die Berechnung beschleunigt und im Falle des Cook-Torrance-Modells die Reflexion physikalisch korrekter darstellt.

# 1. Einleitung

Um die Funktionsweise moderner Bildsynthese zu verstehen, wie sie in Hollywood für Film-animation und Spezialeffekte genutzt wird, oder in der Videospielindustrie zu einem fast fotorealistischen Spielerlebnis führen kann, sollte man sich vorerst die zugrunde liegenden Eigenschaften der beteiligten Objekte ansehen.

Damit Materialien später in der Animation angezeigt werden können, muss man ihre Eigenschaften formell speichern können. Dafür gibt es viele Ansätze, die unterschiedliche Repräsentationen nutzen, um die Oberfläche eines Materials zu modellieren. Um die Effizienz der Bilderstellung zu steigern, macht es Sinn für unterschiedliche Materialien, jeweils eigene Modelle für die Berechnung heranzuziehen.

So eignen sich einfache Modelle wie das Phong- oder das Blinn-Phong-Modell für eine schnelle Bilderzeugung. Um jedoch physikalische Effekte wie Lichtbrechung umzusetzen, muss man auf Modelle zurückgreifen, die diese beim Berechnen mit einbinden. Hierzu wird später das Cook-Torrance-Modell beschrieben.

Um an einer Oberfläche die Eigenschaften der dortigen Reflexion allgemein zu beschreiben, eignet sich als mathematische Schnittstelle eine Bidirektionale Verteilungsfunktion. Diese beschreibt dann im Rendering-Prozess für Lichteinfallswinkel und den Ausfallwinkel zur virtuellen Kamera, wieviel des Lichtes in die gewünschte Richtung reflektiert wird.

Einen Ansatz hierfür bietet die Messung des reflektierten Lichtes unter realen Bedingungen mit einem Gonioreflektometer. Dadurch fallen aber große Datenmengen an und die Erfassungsdauer für einzelne Oberflächen liegt trotz moderner Messmethoden bei mehreren Stunden. Deshalb ist die Modellierung eines Materials, durch die Anpassung von Parametern in einer Reflektanzfunktion, sinnvoll.

Hierfür muss untersucht werden, welche Größen die Variationen zwischen Materialien ausmachen und wie sich diese dann in eine Funktion einbauen lassen.

## 2. Grundlagen

### 2.1 Materialeigenschaften

In der Natur sind Materialien sehr vielfältig. Ihre Oberfläche kann glatt sein und spiegeln, wie Metalle es tun, rau sein und Licht diffus zurückstrahlen, wie zum Beispiel Gesteine, oder das Licht transparent durchlassen, wie Wasser oder Glas. In der Computergrafik möchte man diese Eigenschaften möglichst realitätsnah nachbilden. Wie man in Abbildung 2.1 sehen kann, ist es möglich auf das immer selbe Primitiv der Kugel, durch verschiedene Oberflächenparameter, völlig unterschiedliche Materialien zu simulieren.

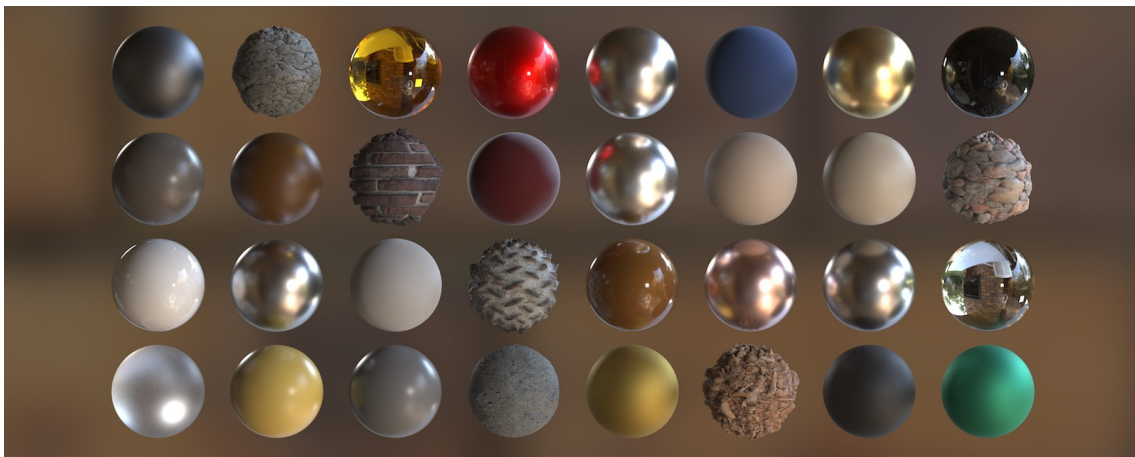


Abbildung 2.1: Materialeigenschaften auf Kugeln [Sid]

Für das Phong-Beleuchtungsmodell werden grundlegend zwei Arten der Lichtinteraktion mit der Oberfläche modelliert. Zum Einen kann das Licht, durch Simulation einer diffusen Oberfläche, in alle Richtungen gleichmäßig wieder abgestrahlt werden. Zum Anderen ist es möglich spekulare Reflexionen zu erzeugen, die die Glanzlichter von Lichtquellen darstellen, die sich an der Oberfläche spiegeln.

Beispielhaft für eine diffuse Oberfläche ist die blaue Kugel in der ersten Zeile der Abbildung; sie wirkt matt und rau. Bei der ersten Kugel oben links erkennt man, dass der spekulare Anteil höher ist als bei der blauen Kugel, denn sie besitzt eine Lichtspiegelung.

Es gibt aber auch Effekte, für die das Phongmodell nicht mehr ausreicht. Dies ist der Grund sich näher mit physikalisch korrekteren Modellen zu beschäftigen. Zum Beispiel sollen Metalle das Glanzlicht in ihrer materialeigenen Farbe reflektieren und nicht wie beim Phongmodell in der Lichtquellenfarbe. Außerdem kann in Physikalischen Modellen der Fresnel-Effekt mit eingebunden werden, wodurch Lichtbrechung und Reflexion vor allem bei flachen Winkeln natürlich wirken.

Bei anderen Kugeln sieht man, dass sie die vor sich liegende Szene spiegeln, wie die rote Kugel der oberen Reihe. Oder dass sie die dahinter liegende Szene anzeigen, da sie eine transmittierende Oberfläche besitzen. Hierfür ist neben der Betrachtung am Oberflächenschnittpunkt wichtig, dass die Strahlen global weiterverfolgt werden, was lokal nicht über eine BRDF(2.3) möglich ist.

Zusätzlich erwähnt werden muss, dass Oberflächenreliefs(z.B. eine Kieselstruktur) nicht über das Beleuchtungsmodell sondern bestimmte Texturen erzeugt werden und somit nicht Teil dieser Arbeit sind.

## 2.2 Die Rendergleichung

Die Rendergleichung stellt die fundamentalen Beziehungen zwischen Objekten im Raum dar, die sich gegenseitig in ihrer Beleuchtung beeinflussen. Hierfür sind jedoch erstmal ein paar physikalische Größen wichtig, um das Reflexionsverhalten von Licht an einer Oberfläche zu beschreiben.

Entsprechend dem auf Ebenen definierten Bogenwinkel gibt es den **Raumwinkel** im Raum. Er gibt in der Einheit steradian [sr] an, welches Verhältnis eine Kugelteilfläche zum Quadrat des Kugelradius besitzt und wird mit  $\omega$  bezeichnet. **Radiance** (Strahldichte) ist die abstrahlende Energie eines Strahls pro effektiver Fläche und Raumwinkel (bezeichnet mit  $L$ , gemessen in  $\frac{W}{m^2 \cdot sr}$ ). Demgegenüber steht die **Irradiance** (Strahlungsflussdichte), die beschreibt welche Energie von einer Fläche empfangen wird (bezeichnet mit  $E$ , gemessen in  $\frac{W}{m^2}$ ). Während Radiance also nur die Energie in eine Richtung darstellt, beinhaltet die Irradiance die gesammelte Energie aus allen möglichen Richtungen. Diese wird durch die Integration der über die beleuchtete Hemisphäre einfallenden Leuchtdichte bestimmt.

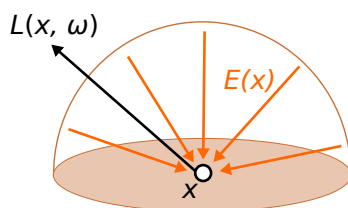


Abbildung 2.2: Radiance  $L$  und Irradiance  $E$

Möchte man nun die Radiance  $L$  von einem bestimmten Punkt  $x$  in die Richtung  $\omega_{out}$  herausfinden, beschreibt die Rendergleichung deren Zusammensetzung. Ihr Energiefluss ergibt sich zum einen aus der objekt-eigenen emittierten Lichtmenge  $L_e$  und aus der Reflektierten. Letztere erhält man jedoch erst, wenn man über alle möglichen Einfallswinkel  $\omega_{in}$  integriert, um herauszufinden von welchen anderen Objekten der Punkt indirekt beleuchtet wird. Die Einfallswinkel beschränken sich hierbei auf die positive Hemisphäre  $\Omega$ . Aus jeder Richtung wird der einfallende Energiefluss  $L_{in}$  mit dem Kosinus seines Einfallswinkels  $\theta_{in}$  gewichtet. Der Energiefluss der in den Punkt einfällt muss aber erst von seinem Herkunftspunkt aus rekursiv weiterberechnet werden. In der Praxis können Integralwerte über das Aussenden weiterer Strahlen im Raytracing berechnet werden und so kann das gesamte Integral später über Monte-Carlo-Algorithmen angenähert werden.

$$L(x, \omega_{out}) = L_e(x, \omega_{out}) + \int_{\Omega} f_r(\omega_{in}, x, \omega_{out}) L_{in}(x, \omega_{in}) \cos \theta_{in} d\omega_{in} \quad (2.1)$$

Der letzte übrige Term  $f_r$  ist eine bidirektionale Reflektionsverteilungsfunktion (BRDF 2.3), mit der berechnet wird, wie groß der reflektierte Anteil aus dieser Richtung in die gewünschte Ausfallrichtung ist. Die BRDF ist eine Proportionalitätskonstante und wird im nächsten Kapitel genauer beschrieben. Sie repräsentiert ein lokales Beleuchtungsmodell und setzt die Materialeigenschaften des beleuchteten Objektes um.

Vorgestellt wurde die Gleichung 1986 von Jim Kajiya und stellt eine Basis dar, aus der die meisten Rendertechniken hergeleitet werden können [Kaj86]. Auf globaler Ebene sind hier besonders zwei Verfahren kurz zu erwähnen, um zu verstehen, woher das einfallende Licht für die lokalen Modelle kommt.

Die Globalen Beleuchtungsmodelle geben an, wie sich das Licht in der Szene ausbreitet, oder welchen Weg ein Strahl von einer Lichtquelle bis in die Kamera durchläuft. Beim Raytracing werden die Sichtstrahlen durch die Kamerapixel in die Szene geschossen. Sobald sie auf eine Oberfläche treffen werden Schattenstrahlen zu allen Lichtquellen gesendet um mögliche Verschattung zu überprüfen. Mit den Lichtquellen wird dann ein lokales Beleuchtungsmodell ausgewertet um den Wert des Oberflächenpunktes zu berechnen. Falls ein Objekt nur indirekt beleuchtet wird, können rekursiv weitere Strahlen ausgesendet werden, jedoch zu Lasten der Performanz.

Raytracing ist eine gute Wahl um spekulare Beleuchtungsstellen gut darzustellen, problematisch ist hingegen indirekte Beleuchtung. Wie man links in Abbildung 2.3 sieht, reflektiert die weiße Rückwand der Szene kein Licht auf die Kugeln.

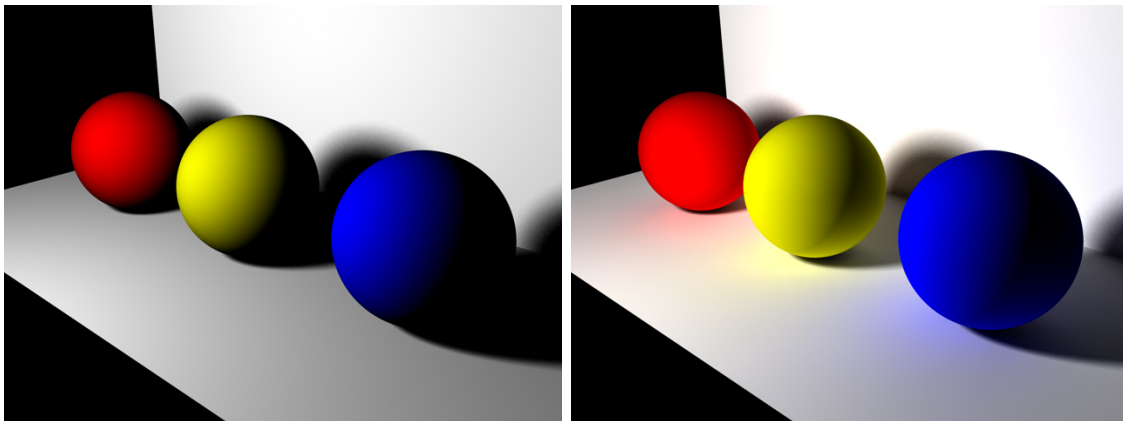


Abbildung 2.3: Rendering mit diffusem Raytracing bzw. Radiosity [Wik]

Mit Radiosity wird unter der Annahme, dass alle Oberflächen ideal diffus sind, der Lichtanteil der nicht absorbiert wird weiterreflektiert. Zwar treten hier keine Glanzlichter auf, aber indirekte Beleuchtung wird gut abgebildet. Rechts in Abbildung 2.3 erkennt man dies an den reflektierten Farben der Kugeln auf dem Boden und daran, dass die Kugeln auch von hinten beleuchtet werden. Vorteil an diesem Modell ist auch die Unabhängigkeit der Betrachtungsrichtung, denn für die diffuse Lichtausbreitung spielt es keine Rolle, von wo man schaut. Somit kann man eine Szene vorab rendern und spart beim Azeigen und Bewegen der Kamera die erforderliche Rechenzeit.

## 2.3 Bidirektionale Verteilungsfunktionen

Betrachten wir nun die lokale Reflexion an der Oberfläche. Zum Anzeigen eines Punktes der von einem ausgesendeten Sichtstrahl getroffen wird, benötigt man Information darüber, wieviel Licht von einer Quelle in die Ausgangsrichtung reflektiert wird. Hierfür können Bidirektionale Reflektions-(BRDF) und Bidirektionale Transmissionsverteilungsfunktionen(BTDF) genutzt werden, die das Reflexions- bzw. Transmissionsverhalten eines eintreffenden Lichtstrahls an der Materialoberfläche beschreiben.

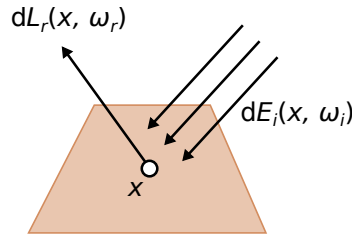


Abbildung 2.4: Radiance und Irradiance

Zur Herleitung betrachtet man das Verhältnis zwischen Radiance und Irradiance eines Punktes. Bei Steigerung der einfallenden Irradiance oder eines größeren Raumwinkels für die Bestrahlung wird auch das reflektierte Licht stärker. Da sich experimentell zeigen lässt, dass sich die Größen proportional zueinander ändern, kann man sie mit einem Faktor gleichsetzen.

$$dL_r(x, \omega_r) = f_r \cdot dE_i(x, \omega_i) \quad (2.2)$$

Dieser Faktor ist eine Proportionalitätskonstante: die BRDF. Umgeformt ergibt sie für zwei gegebene Richtungen das Verhältnis zwischen Radiance und Irradiance, wobei sich die Irradiance als Flussdichte in ihre Bestandteile zerlegen lässt.

$$f_r(\omega_i, \omega_{out}) = \frac{dL_r(x, \omega_{out})}{dE_i(x, \omega_i)} = \frac{dL_r(x, \omega_{out})}{L_i(x, \omega_i) \cos(\theta_i) d\omega_i} \quad (2.3)$$

Um mit der BRDF in der Rendergleichung herauszufinden, wieviel Radiance von einem Punkt reflektiert wird, formt man nach  $L_r$  um und integriert über alle Einfallrichtungen.

$$L_r(x, \omega_{out}) = \int_{\Omega} f_r(\omega_i, x, \omega_{out}) L_i(x, \omega_i) \cos\theta_i d\omega_i \quad (2.4)$$

Lokale Beleuchtungsmodelle, dargestellt als BRDF, simulieren unterschiedlichste Oberflächen. Mit aufwendiger Messtechnik(Gonioreflektometer) kann man sie durch Wertegruppen darstellen, oder sie durch Funktionen approximieren. Außerdem gibt es Besonderheiten bei der Einordnung.

Wenn man bei einem Material die Oberfläche drehen kann, ohne dass sich dabei die Reflexion verändert, so heißt eine BRDF isotrop, wenn die Reflektion richtungsabhängig ist, anisotrop.

Aus der Vorlesung ist das Phong-Beleuchtungsmodell bekannt. Um zu zeigen, dass dieses empirische Modell Schwächen hat, muss definiert werden, was eine physikalisch plausible BRDF einhalten muss. Wenn eine Darstellung physikalisch korrekt sein soll, müssen dafür drei Eigenschaften gelten[Hei14].

- **Positiver Wertebereich:** Die BRDF darf keine negativen Werte zurückgeben.



- **Reziprozität:** Der Lichtweg muss umkehrbar sein, das heißt die BRDF muss beim Vertauschen von Ein- und Ausfallrichtung gleich bleiben.
- **Energieerhaltung:** Es darf maximal so viel Energie reflektiert werden, wie ankommt.

## 3. Lokale Beleuchtungsmodelle

### 3.1 Übersicht

Nur Anhand der Oberflächennormale und verschiedener Materialparameter, sowie dem Lichteinfallswinkel und dem gewünschten Ausfallswinkel in die Kamera, soll berechnet werden, wie hell der untersuchte Punkt dargestellt wird.

Hierfür gibt es empirische Modelle, wie das Phong-Modell, welches die einspielenden Größen über intuitive Beobachtung der Lichtzusammensetzung nutzt. Wenn man die einzelnen Teile betrachtet, aus der sich die abstrahlende Intensität bei Phong zusammensetzt, kann man das Modell aber auch in eine BRDF umformen. Hierüber lässt sich dann begründen, was an dem Modell physikalisch fehlerhaft ist. Außerdem kann es zum Blinn-Phong-Beleuchtungsmodell erweitert werden, was in der realen Berechnung deutlich schneller ist. Beide Modelle gewichten die ambienten, diffusen und spekularen Lichtanteile mit verschiedenen materialabhängigen Parametern. Dem gegenüber steht als physikalisch realitätsnäheres Modell das Cook-Torrance-Beleuchtungsmodell, welches über Mikrofacetten die Materialrauheit beschreibt[TS67].

### 3.2 Phong-Beleuchtungsmodell

Vorgestellt 1975 von Bùì T. Phong ist das Phong-Beleuchtungsmodell ein einfach zu implementierendes Modell, das ohne physikalische Grundlage die Beleuchtung durch verschiedene Summanden modelliert[Pho75]. Das in die Kamera reflektierte Licht  $I_{ges}$  setzt sich aus ambientem, diffusem und spekularem(gespiegelter) Anteil zusammen.

$$I_{ges} = I_a + I_d + I_s \quad (3.1)$$

Im Ambienten Term wird Umgebungslicht  $I_{Umgebung}$  unabhängig von realen Quellen oder indirekter Beleuchtung fest vorgeschrieben. Je nach Material wird die Helligkeit aber mit dem Parameter  $k_a$  verändert, um ein Objekt dunkler oder heller anzuzeigen und so mögliche Absorptionseigenschaften nachzubilden.

$$I_a = I_{Umgebung} \cdot k_a \quad (3.2)$$

Der diffuse Anteil wird über den Einfallswinkel abgeleitet. Nach dem Lambertschen Gesetz wird die Lichtmenge bei einem ideal diffusen Körper gleichmäßig in alle Richtungen und

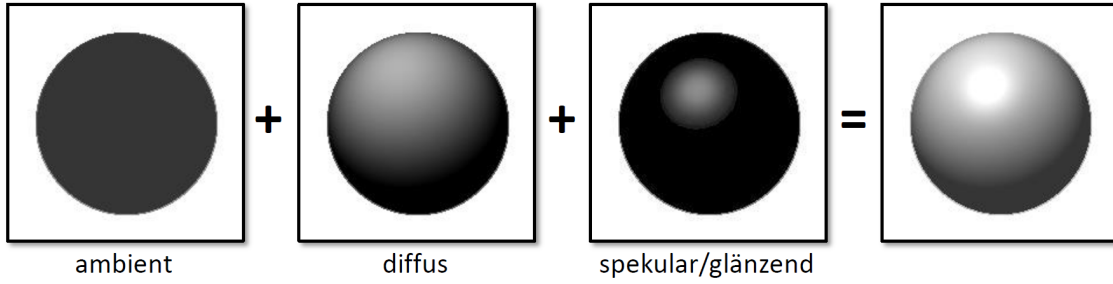


Abbildung 3.1: Die drei Anteile der Phong-Beleuchtung[Dac18]

unabhängig von der Betrachtungsrichtung reflektiert[SM09]. Der Kosinus des Einfallswinkels,  $\cos\theta$ , gewichtet die eingehende Lichtintensität  $I_{in}$  mit dem Verhältnis zwischen einer optimalen senkrechten Bestrahlung und der vorliegenden, eventuell schrägen Bestrahlung. Dieser Wert lässt sich mit dem Skalarprodukt des normierten Normalenvektor der Oberfläche und des normierten Lichteinfallsvektors berechnen. Auch in diesem Term spielt ein materialeigener Reflexionskoeffizient  $k_d$  eine Rolle, um den diffusen Anteil des Lichtstrahls zu regulieren.

$$I_d = I_{in} \cdot k_d \cdot \cos\theta = I_{in} \cdot k_d \cdot (\mathbf{L} \cdot \mathbf{N}) \quad (3.3)$$

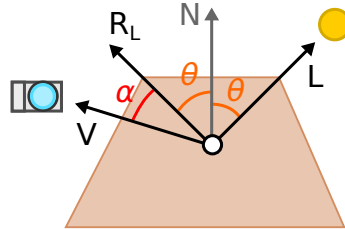


Abbildung 3.2: Alle notwendigen Vektoren bei der Berechnung

Der spekulare Anteil leitet sich darüber ab, wie dicht der Sichtstrahl V dem reflektierten Strahl ist. Hierfür muss zuerst der Reflektionsvektor  $\mathbf{R}_L$  des Lichteinfallstrahls erzeugt werden. Diesen berechnet man über die Formel:  $\mathbf{R}_L = 2(\mathbf{L} \cdot \mathbf{N})\mathbf{N} - \mathbf{L}$ . Die Glanzlichtstärke steuert man über den Phongexponenten  $n$ , der festlegt, wie weit Sichtstrahl und Reflexionsstrahl voneinander abweichen dürfen. Perfekte Spiegelung findet nur in Richtung  $\mathbf{R}_L$  statt, in andere Richtungen fällt die reflektierte Lichtmenge über den Kosinus von  $\alpha$  ab, der durch den Phongexponenten kontrolliert wird. Der Kosinus des Winkels lässt sich hierbei über das Skalarprodukt des Reflektierten Lichtvektors und des Sichtvektors berechnen, die beide normiert sein müssen. Der Parameter  $k_s$  ist dafür zuständig, wie stark das zugrunde liegende Material spiegeln soll.

$$I_s = I_{in} \cdot k_s \cdot \cos^n \alpha = I_{in} \cdot k_s \cdot (\mathbf{R}_L \cdot \mathbf{V})^n \quad (3.4)$$

Zusammengesetzt ergibt sich die Formel des Beleuchtungsmodells dann wie folgt:

$$I_{ges} = I_{Umgebung} \cdot k_a + I_{in} \cdot \left( k_d \cdot (\mathbf{L} \cdot \mathbf{N}) + k_s \cdot (\mathbf{R}_L \cdot \mathbf{V})^n \right) \quad (3.5)$$

Wenn man das Phongmodell nun als BRDF ( $f_{rPhong}$ ) darstellen möchte, vergleicht man, welche Größen einander entsprechen. Hierzu nochmal Gleichung (2.4):

$$L_r(x, \omega_{out}) = \int_{\Omega} f_r(\omega_i, x, \omega_{out}) L_i(x, \omega_i) \cos \theta_i d\omega_i$$

Da der ambiente Anteil nicht vom Lichteinfall abhängt ist er in der BRDF vernachlässigbar. Für die anderen beiden Summanden überträgt sich  $I_{in}$  auf  $L_i$ . Für den diffusen Anteil entspricht der Faktor  $(\mathbf{L} \cdot \mathbf{N})$  genau dem  $\cos \theta_i$ , wodurch einzig  $k_d$  in die BRDF  $f_r$  einfließt. Für den spekularen Anteil fließen aber sowohl  $k_s$  als auch  $\cos^n \alpha$  mit ein. Zusätzlich muss der spekulare Anteil innerhalb der BRDF noch durch  $\cos \theta_i$  geteilt werden, da dieser außerhalb nicht abgedeckt ist. Es ergibt sich also:

$$f_{rPhong}(\omega_i, x, \omega_{out}) = k_d + \frac{k_s \cdot \cos^n \alpha}{\cos \theta_i} \quad (3.6)$$

Physikalisch gesehen machen die zuvor genannten drei Eigenschaften(2.3) die Plausibilität aus. Betrachtet man nun die gewonnene BRDF beispielsweise unter der Bedingung der Reziprozität und tauscht Ein- und Ausfallsrichtung so fällt eine Veränderung auf. Zwar bleiben die beiden Koeffizienten und Winkel  $\alpha$  gleich, Winkel  $\theta_i$  liegt nun jedoch zwischen V und N statt zwischen L und N und kann deshalb ein anderer sein. Die Reziprozität ist somit verletzt[Dac19]. Auch die Energieerhaltung wird verletzt, da das Modell mehr Licht vom Umgebungslicht reflektieren kann als eigentlich vorhanden ist. Gegenüber Mikrofacettenmodellen lässt sich auch der Fresnel-Effekt bei flachem Lichteinfall nicht modellieren.

### 3.3 Blinn-Phong-Beleuchtungsmodell

Eine Erweiterung des Phong-Modells stellte James F. Blinn 1977 vor[Bli77]. Um die aufwendige Berechnung des Reflexionsstrahl zu umgehen, wie sie im Phong-Modell stattfindet, wird hier die Winkelhalbierende zwischen Licht und Sichtstrahl berechnet und mit der Oberflächennormalen verglichen. Darüber lässt sich ebenfalls aussagen welcher Lichtanteil in Blickrichtung gespiegelt wird, siehe Abbildung 3.3.

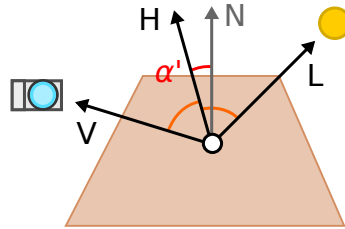


Abbildung 3.3: Der betrachtete Winkel ist etwa halb so groß wie bei Phong

Die Winkelhalbierende wird über die Normierung der Summe von Sichtvektor V und Lichteinfallvektor L berechnet:

$$\mathbf{H} = \frac{\mathbf{V} + \mathbf{L}}{\|\mathbf{V} + \mathbf{L}\|} \quad (3.7)$$

Außer der spekularen Komponente werden alle Anteile des Phong Modells übernommen. So lässt sich in der zugehörigen BRDF einfach der entsprechende Winkel anpassen. Aus  $\cos^n \alpha$  wird deshalb einfach  $\cos^{n'} \alpha'$  was  $(\mathbf{N} \cdot \mathbf{H})^{n'}$  entspricht. Die Glanzlichter sind hierbei aber schwächer. Um die gleichen Ergebnisse wie beim Phong-Modell zu erzielen, muss der entsprechende Exponent  $n'$  angepasst(erhöht) werden.

Blinn ist mathematisch nicht mehr korrekt, sobald Licht- und Sichtstrahl mit der Normalen nicht mehr in einer Ebene liegen. Jedoch ist die Abweichung meist nicht merkbar. Da es wegen der fehlenden Berechnung des Reflektionsvektors jedoch schneller als Phong ist, wird es beispielsweise in OpenGL verwendet.

### 3.4 Cook-Torrance-Beleuchtungsmodell

Wenn eine Oberfläche physikalisch realitätsnahe modelliert werden soll, ist dies über empirische Modelle, wie das Phong-Modell nicht möglich. Beispielsweise wirken Oberflächen im Phongmodell plastikhaft, da der spekulare Anteil die Glanzlichter weiß, beziehungsweise in der Farbe der Lichtquelle erscheinen lässt, aber der diffuse Teil die Objektfarbe besitzt. Metalle glänzen jedoch in der ihr eigenen Farbe, was somit schlecht mit dem Phongmodell generiert werden kann.

Das Cook-Torrance-Modell bietet hier Abhilfe, denn durch eine Oberflächenmodellierung mit Mikrofacetten lässt sich genauer festlegen, welche Lichtanteile reflektiert werden[TC81]. Um die zugehörige BRDF (Formel 3.8) zu verstehen, müssen die einfließenden Komponenten anhand der Vorgänge auf der Mikro-Oberfläche betrachtet werden.

$$\rho(\omega_{out}, \omega_{in}) = \frac{F(\omega_{out}, \omega_h) G(\omega_{out}, \omega_{in}, \omega_h) D(\omega_h)}{4 \cos \theta_{out} \cos \theta_{in}} \quad (3.8)$$

Hierfür muss man sich vorstellen, dass die Materialoberfläche aus vielen winzigen Flächen zusammengesetzt ist, die alle eben sind und spiegeln.

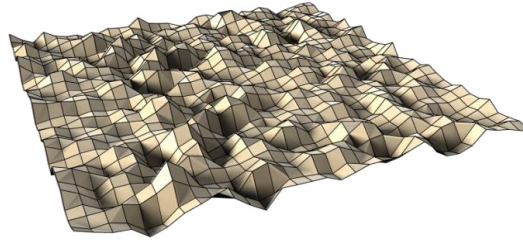


Abbildung 3.4: Facetten einer Mikro-Oberfläche[Dac18]

Die Dichtefunktion  $D$  ist hierbei für die Rauheit der Oberfläche zuständig. Für Oberflächennormalen der Mikro-Oberfläche gibt sie deren Verteilung an. Setzt man die Winkelhalbierende zwischen ein- und ausgehendem Strahl,  $\omega_h$ , ein, so erhält man den Anteil derjenigen Facetten, die so ausgerichtet sind, dass der Lichtstrahl gespiegelt wird.

Für die Funktion könnte man beispielsweise eine Gaußverteilung nutzen. Cook und Torrance empfehlen hierfür eine abgewandelte Version, die Beckmann-Verteilungsfunktion.

$$D = \frac{e^{-\left(\frac{\tan \theta}{m}\right)^2}}{m^2 \cos^4 \theta} \quad (3.9)$$

In diesem Fall ist  $\theta$  der Winkel zwischen der Winkelhalbierenden und der geometrischen Normalen,  $m$  bezeichnet die Wurzel, der gemittelten Quadrate der Neigung der Facetten.

Als nächstes kommt der Abschwächungsterm  $G$  hinzu. Er beschreibt die geometrische Sichtbarkeit einer Mikrofacette. Denn es kann vorkommen, dass einstrahlendes Licht schon von benachbarten Facetten abgefangen wird, wie es in Abbildung 3.5 zu sehen ist. Diesen Effekt nennt man Selbstverschattung. Zusätzlich kann das Licht auch nach der Spiegelung, während des Austritts aus der Oberfläche, durch Abblendung von anderen Facetten abgehalten werden. Dies ist der Maskierungseffekt. Beide Ereignisse führen dazu, dass das Licht nicht mehr in der gewünschten Ausfallrichtung beim Betrachter ankommt.

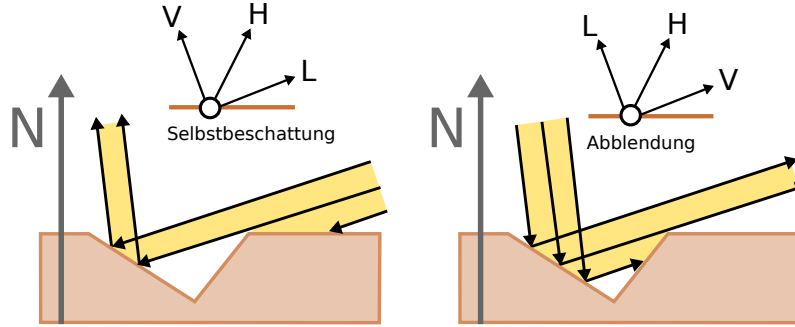


Abbildung 3.5: Geometrische Abschwächung des Lichts

Zusammengesetzt wird der Geometrieterm aus zwei Teilen: der Maskierung und der Verschattungsfunktion.

$$G = \min\left(1, \underbrace{\frac{2(N \cdot H)(N \cdot V)}{(V \cdot H)}}_{\text{Maskierung}}, \underbrace{\frac{2(N \cdot H)(N \cdot L)}{(V \cdot H)}}_{\text{Verschattung}}\right) \quad (3.10)$$

Beide Terme berechnen über die Mittlere Oberflächennormale  $N$ , wieviel Licht letztendlich wieder von der Oberfläche entkommen kann. Das Minimum beider Funktion ist dabei ausschlaggebend, welche der Funktionen abschwächender wirkt. Außer Acht gelassen ist hierbei jedoch die Interreflexion von denjenigen Strahlen, die nach mehrfacher Reflexion doch noch in Blickrichtung abgestrahlt werden.

Das Zustandekommen der beiden Terme für die Maskierung und Verschattung, lässt sich durch geometrische Projektionen veranschaulichen; im Folgenden am Beispiel der Maskierung beschrieben (Verschattung analog).

In Cook und Torrance Theorie zu V-Cavities wird vereinfacht angenommen, dass ein Faceteneinschnitt ein symmetrisches Tal erzeugt. Bei der Maskierung muss nun berechnet werden, welcher Anteil der Facettenfläche, der insgesamt in Blickrichtung projiziert wird ( $p$ ), nicht von der gegenüberliegenden Facette abgeschildert wird ( $p'$ ), also:  $\frac{p'}{p}$ .

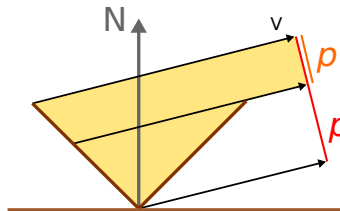


Abbildung 3.6: Der Anteil  $p'$  wird nicht abgeschildert.

Hierfür ist wichtig, dass die Größe einer projizierten Fläche  $p$ , im Vergleich zur Referenzfläche  $r$  um den Kosinus des Abstrahlwinkels  $\cos\theta$  skaliert wird, was dem Skalarprodukt zwischen Referenzflächennormalen und der Projektionsrichtung entspricht (Abbildung 3.7).

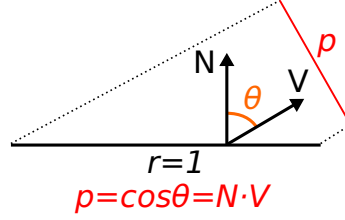


Abbildung 3.7: Projektionsfläche  $p = (N \cdot V) \cdot$  Referenzfläche  $r$

Hierrüber lassen sich jetzt durch mehrfache Projektion die Größen  $p$  und  $p'$  aus Abbildung 3.6 wieder ableiten.

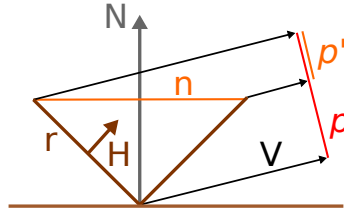


Abbildung 3.8: Projektion von  $r$  auf  $p$  und  $p'$ .

Die ursprüngliche Facettenfläche  $r$  wird durch  $V \cdot H$  auf die Fläche  $p$  projiziert. Dies ist der Anteil der Facette deren Sicht ohne Maskierung möglich wäre. Um  $p'$  zu berechnen muss zunächst die relative Größe von  $n$  berechnet werden, die sich durch die Projektion in Richtung  $N$  aus  $r$  ergibt. Aufgrund der Symmetrie des Einschnittes ist  $n$  doppelt so lang wie die einzelne Projektion von  $r$ :  $n = r \cdot 2(N \cdot H)$ . Diese Fläche kann nun Richtung  $V$  projiziert werden, sodass für  $p'$  gilt:  $p' = n \cdot (N \cdot V) = r \cdot 2(N \cdot H)(N \cdot V)$ . Nimmt man nun an, dass beide Größen als prozentualer Anteil der Facettengröße angegeben werden, so gilt für den Anteil der sichtbaren Fläche, der nicht maskiert wird:

$$G_{\text{Maskierung}} = \frac{p'}{p} = \frac{2(N \cdot H)(N \cdot V)}{(V \cdot H)} \quad (3.11)$$

Da der Anteil maximal 100% sein darf, wird er durch die Minimumsfunktion in Gleichung 3.10 auf 1 begrenzt. Dies ist wichtig, da  $p'$  beispielsweise bei senkrechtem Einfall doppelt so groß wie  $p$  würde.

Den dritten wichtigen Teil der Reflektionsverteilungsfunktion  $\rho$  macht der Fresnel-Term  $F$  aus. Er gibt an, wieviel Energie überhaupt reflektiert wird. Denn reale Oberflächen können auch einen Teil der ankommenden Energie absorbieren oder transmittieren, falls sie durchsichtig sind. Außerdem lässt sich hierbei auch verschieden polarisiertes Licht verarbeiten[Geb], womit physikalische Phänome wie die Interferenz modellierbar werden.

Abhängig von der Polarisierung und Wellenlänge des eintreffenden Lichts, wird der Strahl reflektiert und transmittiert. Jedes Material hat hierfür eine komplexe Brechzahl  $N$ , die aus der relativen Lichtgeschwindigkeit im Medium und einem Absorptionskoeffizienten besteht. Die relative Brechzahl ergibt sich dann als Quotient  $N = \frac{N_2}{N_1}$ . Beim Übergang ins dichtere Medium wird nach dem Snellschen Gesetz der Strahl zum Lot hin, ansonsten vom Lot weg gebrochen.

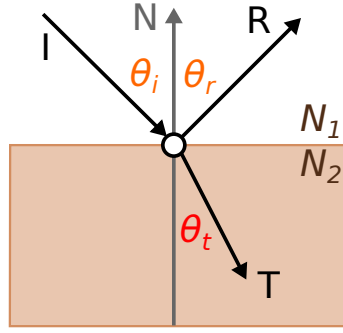


Abbildung 3.9: Aufspaltung eines Lichtstrahls

Um nun den Anteil des reflektierten Lichtes zu berechnen, wird zuerst der senkrecht- und parallel-polarisierte reflektierte Anteil des Strahls bestimmt (Fresnelsche Formeln). Hierbei wird  $\cos_r$  zwischen gewünschtem Ausfallstrahl  $\omega_{out}$  und der Winkelhalbierenden  $\omega_h$  als Lot gemessen. Hieraus lässt sich dann auch  $\cos_t$  ableiten.

$$R_s = \left| \frac{N_1 \cos_r - N_2 \cos_t}{N_1 \cos_r + N_2 \cos_t} \right|^2 \quad \text{und} \quad R_p = \left| \frac{N_1 \cos_t - N_2 \cos_r}{N_1 \cos_t + N_2 \cos_r} \right|^2 \quad (3.12)$$

Der Fresnell-Term  $F$  für die BRDF des Modells ergibt sich jetzt als die unpolarisierte Reflexion. Diese ist der Durchschnitt der zuvor berechneten polarisierten Anteile.

$$F(\omega_{out}, \omega_h) = R = \frac{(R_s + R_p)}{2} \quad (3.13)$$



## 4. Zusammenfassung

### 4.1 Ergebnisse

Da die korrekte Darstellung von Materialeigenschaften sehr komplex ist, wurden über die Zeit von unterschiedlichen Pionieren der Computergrafik Beleuchtungsmodelle entwickelt, um Objekte detailgetreu nachzuahmen. Hierbei genügen empirische Modelle, um eine Szene schnell und einfach zu rendern. Diese wirken auch schon sehr realistisch. Um die Renderzeit zu reduzieren kann außerdem über alternative Berechnungen Zeit gespart werden, beispielhaft an der Berechnung der Winkelhalbierenden im Blinn-Modell gezeigt. Um wirklichen Fotorealismus zu erreichen kann aber ein physikalisches Modell herangezogen werden, um auch spezielle Oberflächenereignisse zu simulieren.

Im Vergleich der lokalen Beleuchtungsmodelle lässt sich außerdem sagen, dass das Phong- und Blinn-Phong-Modell dazu geeignet sind, eine Oberfläche mit Glanzlicht einfach zu modellieren. Der Rechenaufwand ist gegenüber dem Cook-Torrance-Verfahren deutlich niedriger und macht eine schnellere Bildsynthese möglich. Um aber beispielsweise Metalle darzustellen und die spekularen Reflexionen realistischer wirken zu lassen, ist die Nutzung eines physikalisch basierten Systems besser.

### 4.2 Ausblick

Mit zukünftig immer stärkeren Grafikkarten lassen sich beim Raytracing parallel viel mehr Strahlen gleichzeitig verfolgen, was es ermöglicht, auch in Echtzeit diese Verfahren für Computerspiele und Simulationen zu nutzen. Animationsstudios erzeugen mit Software wie beispielsweise Pixar's RenderMan Bilder, die kaum noch auf die virtuelle Synthese rückschließen lassen. Auch die physikalischen Beleuchtungsmodellen bieten neben dem Cook-Torrance-Modell noch eine große Bandbreite, die auch schwierig zu simulierende Effekte abbilden.

Beispielsweise Stoffe, Flüssigkeiten, oder andere durchsichtige Materialien werden dabei simuliert, oder Haare und Fell dargestellt, bei denen es lange schwer war, unnatürliches Aussehen zu vermeiden.

# Literaturverzeichnis

- [Bli77] James F. Blinn: *Models of light reflection for computer synthesized pictures*. SIGGRAPH 1977 Proceedings, ACM Press, 3:192–198, 1977.
- [Dac18] Bildquelle: Carsten Dachsbacher: *Computergrafik Vorlesung, Kapitel 2: Raytracing, Folie 84*. 2018.
- [Dac19] Carsten Dachsbacher: *Vorlesung Fotorealistische Bildsynthese, Kapitel 3: BRDF*. 2019.
- [Geb] Nikolaus Gebhardt: *Einige BRDF Modelle*. <http://www.irrlicht3d.org/papers/BrdfModelle.pdf>.
- [Hei14] Eric Heitz: *Understanding the Masking-Shadowing Function in Microfacet-Based BRDFs*. Journal of Computer Graphics Techniques (JCGT), 3, 2014. <http://jcgt.org/published/0003/02/03/>.
- [Kaj86] James T. Kajiya: *The rendering equation*. ACM Press, 20:143–150, 1986. <http://www.cs.princeton.edu/courses/archive/fall02/cs526/papers/kajiya86.pdf>.
- [Pho75] Bui Tuong Phong: *Illumination for Computer Generated Pictures*. Comm. of the ACM, 1975. <https://doi.org/10.1145/360825.360839>.
- [Sid] Bildquelle: SideFx.com: *Houdini 17.5, shading and rendering*. <https://www.sidefx.com/docs/houdini/news/16/rendering.html>.
- [SM09] Peter Shirley und Steve Marschner: *Fundamentals of Computer Graphics*. CRC Press, third edition, 2009.
- [TC81] K. E. Torrance und R. L. Cook: *A Reflectance Model for Computer Graphics*. SIGGRAPH '81 Proceedings), 15, 1981. <http://inst.eecs.berkeley.edu/~cs283/sp13/lectures/cookpaper.pdf>.
- [TS67] K. E. Torrance und E. M. Sparrow: *Theory of off-specular reflection from roughened surfaces*. Journal of the Optical Society of America A 57), 15:1105–1114, 1967.
- [Wik] Bildquelle: Wikipedia: *Radiosity(Computergrafik): Radiosity-no.jpg, Radiosity-yes.jpg*. [https://de.wikipedia.org/wiki/Radiosity\\_\(Computergrafik\)](https://de.wikipedia.org/wiki/Radiosity_(Computergrafik)).

# Erklärung

Ich versichere, dass ich die Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet habe. Die Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt und von dieser als Teil einer Prüfungsleistung angenommen.

Karlsruhe, den 18. November 2019

(Jonas Lehmann)