



# Realistische Darstellung von Objekten

Proseminar Computergraphik

Sommersemester 2009

Dozent: Wilfried Mascolus  
Lehrstuhl für Computergraphik und Visualisierung  
Fakultät Informatik

Matthias Köngeter  
Matrikelnummer 3209261  
s9581863@mail.inf.tu-dresden.de

# Inhaltsverzeichnis

<b>1</b>	<b>Abstract</b>	<b>4</b>
<b>2</b>	<b>Darstellung von Kanten und Linien</b>	<b>5</b>
2.1	Der Aliasing-Effekt . . . . .	5
2.2	Antialiasing . . . . .	5
2.2.1	Unweighted Area Sampling . . . . .	5
2.2.2	Weighted Area Sampling . . . . .	5
<b>3</b>	<b>Lokale Beleuchtung, Reflexion, Schattierung und Transparenz</b>	<b>7</b>
3.1	Beleuchtung . . . . .	7
3.1.1	Ambientes Licht . . . . .	7
3.1.2	Punktförmige Lichtquellen . . . . .	7
3.2	Spiegelnde Reflexion . . . . .	8
3.2.1	Das Modell nach Torrance und Sparrow bzw. Blinn bzw. Cook und Torrance . . . . .	8
3.2.2	Das Modell von Phong Bui-Tong . . . . .	10
3.3	Schattierung . . . . .	10
3.3.1	Konstante Schattierung . . . . .	10
3.3.2	Gouraud-Schattierung . . . . .	11
3.3.3	Phong-Schattierung . . . . .	11
3.4	Transparente Objekte . . . . .	12
3.4.1	Interpolierte Transparenz . . . . .	12
3.4.2	Screen-Door-Transparenz . . . . .	13
3.4.3	Gefilterte Transparenz . . . . .	13
<b>4</b>	<b>Globale Beleuchtung</b>	<b>14</b>
4.1	Ray-Tracing . . . . .	14
4.2	Radiosity-Verfahren . . . . .	15
<b>5</b>	<b>Mapping-Techniken</b>	<b>16</b>
5.1	Texture-Mapping . . . . .	16
5.2	Bump-Mapping . . . . .	17
5.3	Environment-Mapping . . . . .	17
<b>6</b>	<b>Quellenangabe</b>	<b>18</b>

# Abbildungsverzeichnis

2.1	Ansätze zur Intensitätsberechnung . . . . .	6
3.1	Sichtbare Fläche . . . . .	9
3.2	Schattierungsverfahren am Beispiel einer Kugel . . . . .	12
3.3	Transparentes Polygon . . . . .	13
3.4	Screen-Door-Transparenz . . . . .	13
5.1	Texture-Mapping . . . . .	16
5.2	Anwendung des Bump-Mappings auf eine Kugel . . . . .	17

# 1 Abstract

Der Vortrag "Realistische Darstellung von Objekten" beschäftigt sich mit verschiedenen Aspekten, die bei der Modellierung wirklichkeitsgetreuer Szenen berücksichtigt werden müssen. Schwerpunkt dabei bilden sowohl die Berechnung der lokalen Beleuchtung, einschließlich Reflexionen, Schattierungen und Transparenz, als auch zwei wichtige Techniken zur Modellierung der globalen Beleuchtung, nämlich der RayTracing-Algorithmus und das Radiosity-Verfahren. Außerdem wird auf verschiedene Mapping-Techniken und Möglichkeiten zur Vermeidung des Aliasing-Effektes eingegangen.

Ziel des Vortrages ist es, den Hörern einen allgemeinen Überblick über oben genannte Verfahren zu verschaffen; auf mathematische Grundlagen und die Feinheiten der einzelnen Algorithmen wird zugunsten der Verständlichkeit und der erforderlichen Kürze weitestgehend verzichtet.

## 2 Darstellung von Kanten und Linien

### 2.1 Der Aliasing-Effekt

Rasterorientierte Ausgabegeräte, wie z.B. Monitore, verwenden zur Darstellung von Objekten diskrete Bildpunkte oder auch Pixel, die so angesteuert werden müssen, dass sie das darzustellende Objekt möglichst gut approximieren. Punkte werden also nicht an ihrem eigentlichen Ort, sondern an einem genäherten Rasterpunkt gezeichnet. Dadurch kommt es sehr häufig zu Verfremdungseffekten, beispielsweise können Linien treppenförmige Zick-Zack-Muster aufweisen oder nahe beieinander liegende Kurvenstücke können sich überlagern. Außerdem wirken diagonal verlaufende Linien aufgrund des größeren Pixelabstands oft blasser als horizontal oder vertikal verlaufende. Schwierigkeiten macht auch die Darstellung von kleinen Details, da ihre Abmessungen nicht ohne weiteres im Raster darstellbar sind und sie daher auf das passende Größenverhältnis konvertiert werden müssten.

Die intuitive Lösung all dieser Probleme wäre es, einfach die Auflösung zu erhöhen. Da das aber sehr teuer ist und das eigentliche Problem nicht beseitigt, versucht man, durch Graustufen weichere Übergänge zu erzielen und auf diese Weise die gewünschten Effekte zu vermeiden. Beide Möglichkeiten bezeichnet man als Antialiasing.

### 2.2 Antialiasing

Beim Antialiasing gibt es verschiedene Ansätze, zum einen die genannte Erhöhung der Auflösung, hauptsächlich aber die Anpassung der Pixelintensitäten, so dass weichere Übergänge entstehen. Das Grundprinzip dabei ist, dass Pixel, die nur anteilig überdeckt werden, auch nur anteilige Intensität erhalten. Zur Berechnung der Intensität gibt es zwei gängige Verfahren: Das *Unweighted Area Sampling* und das *Weighted Area Sampling*.

#### 2.2.1 Unweighted Area Sampling

Hierbei werden die einzelnen Pixel als Quadrate modelliert, deren Mittelpunkte den Gitterpunkten des Rasters entsprechen. Für jeden Pixel wird die zugehörige Graustufe durch den Flächenanteil bestimmt, den die Linie überstreicht. Dieses Verfahren verbessert die optische Wirkung bereits deutlich.

#### 2.2.2 Weighted Area Sampling

Beim Weighted Area Sampling wird zusätzlich zur Größe der überdeckten Fläche noch ihre Lage im Pixel mit einbezogen: Je näher sie der Pixelmitte ist, desto dunkler wird

sie eingefärbt. Die Pixel werden dabei nicht mehr als Quadrate modelliert, sondern als sich überlappende Kreise, deren Mittelpunkte wieder auf den Gitterpunkten des Rasters liegen. Der Radius entspricht dem Abstand zweier Gitterpunkte, der Kreis durchläuft also die Mittelpunkte aller seiner Nachbarkreise.

Die Pixelintensitäten berechnet man nun über verschiedene Integrale und Funktionen; es gibt aber auch eine einfachere und trotzdem effiziente Methode: Den sogenannten *Algorithmus von Gupta und Sproull*. Die Intensität eines Pixels wird hierbei als proportional zum Abstand  $d$  der Pixelmitte zur Mittelachse der zu zeichnenden Linie angenommen.  $d$  kann dabei 24 verschiedene Werte zwischen 0 und 1.5 annehmen, die 24 verschiedenen Intensitäten entsprechen. Zeichnet man nun eine Linie, muss anstelle von aufwändigen Integralen nur  $d$  berechnet und der zugehörige Intensitätswert aus einer Tabelle abgelesen werden.

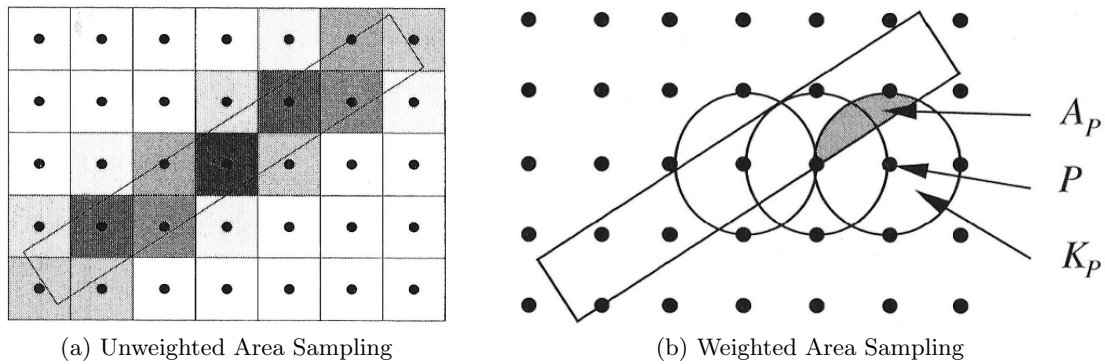


Abbildung 2.1: Ansätze zur Intensitätsberechnung

## 3 Lokale Beleuchtung, Reflexion, Schattierung und Transparenz

Die Realitätsnähe einer modellierten Szene steht und fällt mit ihrer Beleuchtung. Ohne Intensitäts- oder Helligkeitsunterschiede wirkt das Bild künstlich und synthetisch, und auch Glanz- und Spiegelungseffekte sind für die meisten Darstellungen unverzichtbar.

Die Beleuchtung allgemein unterteilt man üblicherweise in zwei Gebiete: Die *lokale Beleuchtung*, die sich mit den Lichtverhältnissen in einem bestimmten Punkt der Szene beschäftigt, und die *globale Beleuchtung*, die sich mit den Wechselwirkungen von Mehrfachreflexionen befasst.

### 3.1 Beleuchtung

#### 3.1.1 Ambientes Licht

Ambientes Licht, auch *Hintergrundbeleuchtung* oder *Indirekte Beleuchtung* genannt, ist das wohl einfachste Modell zur Beleuchtung einer Szene. Es simuliert Licht, das nicht direkt aus einer Lichtquelle kommt, sondern bereits vielfach reflektiert wurde und deshalb gleichmäßig und aus allen Richtungen leuchtet. In der Realität liegt der Anteil dieser Lichtart bei um die 30%, kann aber je nach Umgebung stark schwanken. Ambientes Licht kann - wie jedes andere auch - reflektiert, absorbiert oder, bei transparenten Objekten, durchgelassen werden.

Da eine alleinige Beleuchtung mit ambientem Licht dazu führt, dass jede Oberfläche unabhängig von ihrer Ausrichtung gleich hell beleuchtet wird, wirkt das Ergebnis ziemlich unecht – es fehlen einfach die Schatten. Daher kommt man um die Einführung zusätzlicher punktförmiger Lichtquellen nicht umhin.

#### 3.1.2 Punktförmige Lichtquellen

Von punktförmigen Lichtquellen, die mit Glühbirnen oder der Sonne vergleichbar sind, geht gerichtetes Licht aus. Daher ist die Helligkeit bzw. die Intensität eines Punktes auf einer Fläche abhängig von seiner Entfernung zur Lichtquelle und der Ausrichtung der Fläche ihr gegenüber. Reflektiert wird diffus, das heißt gleichmäßig und in alle Richtungen, weshalb nur der Einfallswinkel und nicht der Ausfallswinkel beachtet werden muss.

Durch die Einführung von punktförmigen Lichtquellen erscheinen die Bilder zwar immer noch relativ matt, Schattierungen sind aber immerhin schon gut erkennbar. Eine

weitere Verbesserung erreicht man durch die Einführung spiegelnder Reflexionen, bei denen das Licht im Gegensatz zur diffusen Reflexion in eine bestimmte Richtung reflektiert wird.

## 3.2 Spiegelnde Reflexion

Bei glatten oder glänzenden Oberflächen sollte das Licht nicht diffus in alle Richtungen geworfen, sondern vielmehr in eine bestimmte Richtung gespiegelt werden. Diese Richtung errechnet sich nach dem bekannten Gesetz „*Einfallswinkel = Ausfallswinkel*“. Aus diesem Gesetz folgt auch, dass nur ein kleiner Teil der Strahlen das Auge trifft, nämlich der, der die Oberfläche im passenden Winkel erreicht.

Allerdings sind in der Realität spiegelnde Oberflächen und sogar Spiegel an sich niemals vollkommen eben. Um das zu berücksichtigen, gibt es unterschiedliche Spiegelungsmodelle. Zwei davon sind das theoretisch fundierte *Modell nach Torrance und Sparrow* bzw. *Blinn* bzw. *Cook und Torrance* und das heuristische *Modell von Phong Bui-Tong*.

### 3.2.1 Das Modell nach Torrance und Sparrow bzw. Blinn bzw. Cook und Torrance

Dieses Modell geht von der Annahme aus, dass die Oberfläche aus sehr vielen kleinen, ebenen, unterschiedlich ausgerichteten Facetten besteht, die alle als perfekte Spiegel betrachtet werden. Für jeden Objektpunkt muss nun einiges berücksichtigt werden:

1. der Anteil der Facetten, die in die Richtung des Betrachters spiegeln und damit die Glanzeffekte verursachen;
2. die Anzahl der Facetten, die für Lichtquelle und Betrachter überhaupt sichtbar sind;
3. der Anteil des Lichts, der durch die Rauheit der Oberfläche verloren geht;
4. der Anteil des Lichts, der überhaupt reflektiert wird.

Zu 1) Der Anteil der in Richtung des Betrachters spiegelnden Facetten wird durch einen Parameter  $D$  beschrieben, der Werte zwischen 0 und 1 annehmen kann. Je exakter die Oberfläche ausgerichtet ist, desto größer ist der Wert von  $D$ . Für die genaue Berechnung gibt es zahlreiche Formeln, bei der hier gezeigten bezeichnet  $K$  einen frei definierbaren Glanzparameter, wobei ein kleines  $K$  eine sehr starke Winkelabhängigkeit und damit viel Glanz zur Folge hat.  $\langle n, h \rangle$  ist das Skalarprodukt aus Flächennormaler und Winkelhalbierender zwischen einfallendem Strahl und Strahl zum Auge.

$$D := \frac{K}{(K+1-\langle n, h \rangle)^2}$$



Zu 2) Der Anteil der Fläche, der für Lichtquelle und Betrachter sichtbar ist, verändert sich mit dem Ein- bzw. Ausfallswinkel des Lichts: Je flacher die Strahlen das Objekt überstreichen, desto größer ist die sichtbare Fläche. Berücksichtigt wird dies, indem die Skalarprodukte von der Flächennormalen mit dem einfallendem Strahl bzw. dem Strahl zum Auge in die Formel mit eingebaut werden.

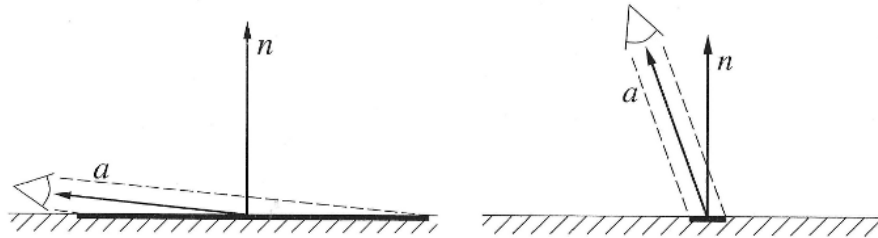


Abbildung 3.1: Sichtbare Fläche

Zu 3) Durch die Rauheit der aus Facetten aufgebauten Oberfläche kann es vorkommen, dass Facetten sich gegenseitig verschatten. Daher führt man einen Parameter  $G$  ein, der ebenfalls Werte zwischen 0 und 1 annehmen kann und der den Gesamtanteil des ungehindert verlaufenden Lichts angibt. Dieser entspricht dem Minimum des ungehindert ausfallenden bzw. ungehindert einfallenden Lichts und wird berechnet durch das Verhältnis zwischen Facettengröße und der Größe des blockierenden Facettenanteils:

$$G := \min\{1, G_l, G_a\} \quad G_l := 1 - \frac{l_b}{l_f} \quad \text{bzw.} \quad G_a := 1 - \frac{l_b}{l_f}$$

$l_b$ : Facettengröße

$l_f$ : Größe des blockierenden Facettenanteils

Zu 4) Da in der Realität nicht das ganze Licht reflektiert wird, sondern Teile davon je nach Betrachtungswinkel und Wellenlänge auch absorbiert oder durchgelassen werden können, benötigt man einen dritten Parameter  $F_\lambda$ , der sich wie  $D$  und  $G$  im Wertebereich zwischen 0 und 1 bewegt. Berechnet wird er nach der Gleichung von Fresnel, die unter anderem den Brechungsindex mit einbezieht. Dieser beschreibt das Verhältnis der Lichtgeschwindigkeit im Vakuum zur Lichtgeschwindigkeit im Medium.

$$F_\lambda := \frac{1}{2} \cdot \left(\frac{g-c}{g+c}\right)^2 \cdot \left(1 + \left(\frac{c(g+c)-1}{c(g-c)+1}\right)^2\right)$$

$g$ :  $\sqrt{\eta^2 + c^2 - 1}$

$c$ :  $\langle l, h \rangle$

$l$ : einfallender Lichtstrahl

$h$ : Winkelhalbierende zwischen einfallendem Lichtstrahl und Strahl zum Auge

$\eta$ : effektiver Brechungsindex (Quotient der Brechungsindizes beider Medien; abhängig von  $\lambda$ )

### 3.2.2 Das Modell von Phong Bui-Tong

Dieses Modell ist, verglichen mit dem vorherigen, stark vereinfacht, beruht allerdings auf rein heuristischen Annahmen. Es geht davon aus, dass die spiegelnde Reflexion am stärksten ist, wenn der reflektierte Strahl genau ins Auge trifft, und mit steigendem Winkel zwischen Ausfallsstrahl und Blicklinie schwächer wird.

Um die Stärke der Spiegelreflexion und ihre Winkelabhängigkeit zu beschreiben, verwendet man zwei farbunabhängige Materialparameter, nämlich den Spiegelreflexionskoeffizienten  $R_S^{(i)}$  und den Spiegelreflexionsexponenten  $k$ .  $R_S^{(i)}$  wird der Einfachheit halber nicht als winkelabhängig, sondern als konstant angenommen, wobei der Wert meist je nach Gefallen experimentell bestimmt wird.  $k$  kann Werte zwischen eins und einigen hundert annehmen, bei einem perfekten Spiegel wäre  $k$  unendlich. Als Näherung kann man auch die Formel  $\cos^n(\alpha)$  verwenden, wobei  $n$  vom Material abhängt und mindestens 1 sein muss. Es bestimmt die Rauheit der Oberfläche und damit die Begrenzung des Glanzlichts.  $\alpha$  gibt den Winkel zwischen ausfallendem Licht und der Blicklinie des Betrachters an.

Das Modell von Phong ist zwar durchaus brauchbar, kommt aber qualitativ nicht an ersteres heran. Es eignet sich für die Darstellung von glatten, plastikähnlichen Oberflächen; vor allem für Spiegelungen auf metallenen Oberflächen sollte man aber auf das Modell von Torrance und Sparrow bzw. Blinn bzw. Cook und Torrance zurückgreifen.

## 3.3 Schattierung

Als Schattierung, oder englisch *Shading*, bezeichnet man die Zuordnung von Helligkeits- bzw. Farbwerten zu einzelnen Bildpunkten. Etwas konkreter bedeutet das, dass nicht nur ein geeignetes lokales Beleuchtungsmodell gewählt werden muss, sondern auch entschieden werden muss, in welchen Bildpunkten dieses Modell wirklich angewandt werden soll bzw. in welchen Punkten weniger rechenintensive Interpolationsmethoden ausreichen. Letztere sind vor allem bei interaktiven Echtzeitanwendungen von Bedeutung, da die Intensitäts- und Farbwertberechnung auf Basis des Beleuchtungsmodells zu aufwändig ist, als dass man sie für jeden Pixel einzeln durchführen könnte. Daher sind die meisten Schattierungstechniken darauf ausgelegt, mit möglichst wenigen Intensitäts- und Farbwertberechnungen auszukommen oder aber ihre Berechnung zu vereinfachen.

### 3.3.1 Konstante Schattierung

Die konstante Schattierung, auch *Flat-Shading*, ist die einfachste Schattierungsmethode. Das lokale Beleuchtungsmodell wird hierbei nur in einem Punkt eines Polygons angewandt (beispielsweise im Schwerpunkt). Der ermittelte Wert, der sich aus Lichteinfallswinkel und Betrachtungswinkel ergibt, wird anschließend für alle Pixel des Polygons verwendet. Nachteil dieser sehr starken Vereinfachung ist, dass genannte Winkel auf einem Polygon nur dann konstant sind, wenn Lichtquelle und Betrachterstandpunkt unendlich weit entfernt sind. Außerdem kann es an den Kanten zu starken Intensitätssprüngen kommen, die insbesondere dann negativ auffallen, wenn das polygonale Netz eine gekrümmte

und glatte Oberfläche darstellen soll – selbst bei engen Polygonnetzen sind Flächen als Facetten sichtbar. Zusätzlich kann auch noch der Effekt der Machschen Streifen hinzukommen, der bewirkt, dass eine dunkle Fläche an der Grenze zu einer hellen Fläche noch dunkler wirkt und die helle Fläche an ebendieser Grenze noch heller. Dadurch wird die Kante, die eigentlich möglichst unauffällig sein sollte, noch zusätzlich betont.

Um die Probleme der konstanten Schattierung zu vermeiden, geht man über zur interpolierten Schattierung. Hierbei gibt es zwei Ansätze: Zum einen die *Gouraud-Schattierung*, bei der das lokale Beleuchtungsmodell in den Eckpunkten des Polygons angewendet und über den Rest interpoliert wird, und zum anderen die *Phong-Schattierung*, bei der nicht Intensitäten, sondern Normalenvektoren interpoliert werden.

### 3.3.2 Gouraud-Schattierung

Bei dieser Methode wird über die Intensitäten bzw. Farben in den Polygonecken interpoliert. Dadurch werden Unregelmäßigkeiten entlang der Kanten vermieden. Allerdings muss man zunächst einmal die Ecknormalen berechnen, um sie mit den Vektoren des Lichteinfalls und der Blicklinie des Betrachters in Beziehung zu setzen. Dazu kann man eine Mittelung der Flächennormalen anliegender Polygone durchführen, oder aber man verwendet, falls vorhanden, eine analytische Beschreibung der Oberflächen. Hat man nun die Ecknormalen, werden die jeweiligen Eckintensitäten berechnet, danach werden die Kantenintensitäten durch lineare Interpolation gewonnen und diese wiederum werden zeilenweise linear interpoliert, um die Intensitäten im Inneren der Polygone zu bestimmen.

Auf diese Weise werden glatte Übergänge geschaffen; allerdings sind auch Nahtstellen realisierbar: Dazu muss man nur für einen Eckpunkt zwei Ecknormale berechnen, und zwar aus den Flächennormalen von jeweils einer Seite der gewünschten Kante. Dadurch ergeben sich zwei unterschiedliche Intensitäten, die zur Interpolation auf den jeweiligen Seiten der Kante herangezogen werden, und als Folge entsteht ein deutlich sichtbarer Intensitätssprung.

Die Gouraud-Schattierung liefert bereits deutlich bessere Ergebnisse als die konstante Schattierung, allerdings können, da nur die Eckpunkte betrachtet werden, Effekte in der Flächenmitte nicht erkannt werden.

### 3.3.3 Phong-Schattierung

Bei der Phong-Schattierung werden nicht Farbintensitäten, sondern Normalenvektoren interpoliert. Dazu benötigt man wieder die Ecknormalen, die man wie bei der Gouraud-Schattierung bestimmt. Aus ihnen bildet man durch lineare Interpolation und anschließender Normierung die Normalen längs der Kanten. Nach dem lokalen Beleuchtungsmodell werden aus den Normalen die Intensitäten auf der Kante errechnet, und im letzten Schritt werden durch lineare Interpolation der Kantennormalen die Normalen im Inneren des Polygons bestimmt und aus diesen letztendlich die Intensitäten im Polygoninneren ermittelt.

Die Phong-Schattierung ist wesentlich aufwändiger als die Gouraud-Schattierung, so muss zum Beispiel jeder Normalenvektor zusätzlich normiert und das lokale Beleuchtungsmodell in jedem Punkt angewandt werden. Trotzdem lässt sie sich softwaretechnisch effizienter implementieren, während die Gouraud-Schattierung hardwaretechnisch große Vorteile hat. Deshalb kommt bei Echtzeitanwendungen meistens die Gouraud-Schattierung zum Einsatz.

Von der Qualität her liefert die Phong-Schattierung deutlich bessere Ergebnisse, vor allem weil durch die bessere Approximation der Normalen gekrümmte Flächen mitsamt ihren Licht-, Glanz- und Spiegelungseffekten wirklichkeitsgetreuer dargestellt werden können. Die Gouraud-Schattierung dagegen vergrößert durch die lineare Interpolation der Intensitätswerte den Einflussbereich von eigentlich kleinflächigen Lichtspiegelungen, wodurch diese abgeschwächt und verwässert werden.

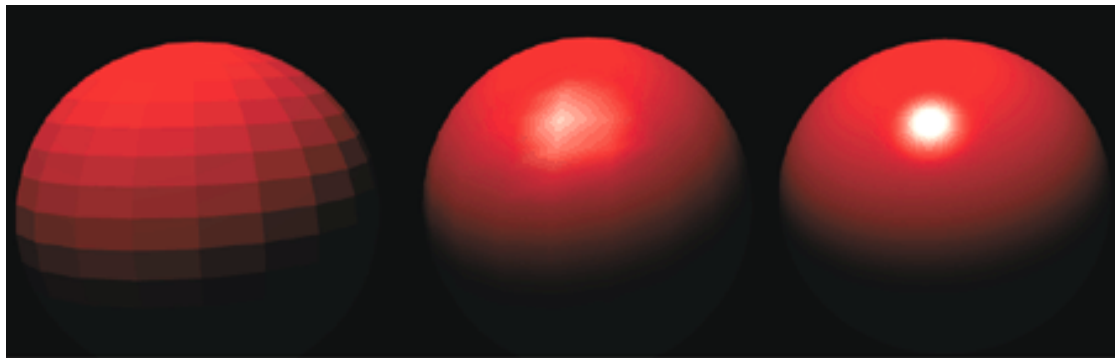


Abbildung 3.2: Schattierungsverfahren am Beispiel einer Kugel: Konstante Schattierung, Gouraud, Phong (v.l.n.r.)

### 3.4 Transparente Objekte

In der Realität werden Lichtstrahlen nicht nur diffus/spiegelnd reflektiert oder absorbiert, sondern können, je nach Material, auch durchgelassen werden. Da die dabei auftretende Brechung des Lichts nicht ganz einfach zu modellieren ist, wird sie bei den einfacheren Modellen zur Realisierung von Transparenz ignoriert.

Bei den Verfahren ohne Lichtbrechung unterscheidet man zwischen drei Ansätzen: der *interpolierten Transparenz*, der *Screen-Door-Transparenz* und der *gefilterten Transparenz*.

#### 3.4.1 Interpolierte Transparenz

Die resultierende interpolierte Intensität  $I_p$  in einem Pixel wird durch folgende Formel berechnet:  $I_p := (1 - k_t) \cdot I_1 + k_t \cdot I_2$

$I_1$  und  $I_2$  sind die Intensitäts- bzw. Farbwerte für Polygon 1 und 2 in dem betrachteten Pixel,  $k_t$  ist der Transmissionskoeffizient und gibt an, wie durchsichtig das Material von

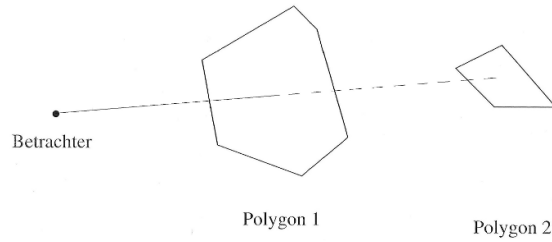


Abbildung 3.3: Transparentes Polygon

Polygon 1 ist.  $k_t = 1$  bedeutet, dass Polygon 1 völlig transparent, fast schon unsichtbar ist, während bei  $k_t = 0$  überhaupt kein Licht durchgelassen wird.

### 3.4.2 Screen-Door-Transparenz

Bei diesem Verfahren werden einige Pixel mit dem Intensitäts- bzw. Farbwert des durchsichtigen Objekts eingefärbt, die anderen Pixel mit dem Wert des dahinterliegenden Objekts – es gilt also  $I_p \in \{I_1, I_2\}$ . Die Verteilung der Pixel folgt einem regelmäßigen Muster, das vom Grad der Transparenz abhängig ist. Bei mehreren transparenten Objekten hintereinander kann es bei diesem Verfahren vorkommen, dass sich die Muster unglücklich überlagern.

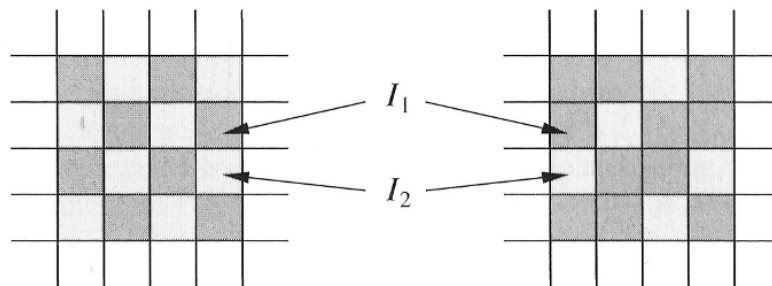


Abbildung 3.4: Screen-Door-Transparenz

### 3.4.3 Gefilterte Transparenz

Hierbei kann das transparente Polygon als Filter wirken und Licht bestimmter Wellenlängen bevorzugt durchlassen:  $I_{P,\lambda} := (1 - k_t(\lambda)) \cdot I_1 + k_t(\lambda) \cdot I_2$

Es wird dieselbe Formel verwendet wie bei der Interpolierten Transparenz, mit dem einzigen Unterschied, dass der Transmissionskoeffizient  $k_t(\lambda)$  hier von der Wellenlänge abhängt. Auf diese Weise lässt sich beispielsweise rotes Glas simulieren.

## 4 Globale Beleuchtung

Die globale Beleuchtung beschäftigt sich, wie bereits erwähnt, nicht mit den Lichtverhältnissen an einem bestimmten Punkt der Szene, sondern den Licht-Wechselwirkungen aller Objekte der Szene miteinander.

Zur Modellierung dieser Wechselwirkungen gibt es verschiedene Ansätze. Die zwei grundlegendsten sind zum einen das sogenannte *Ray-Tracing*, das auf spiegelnde Reflexion ausgelegt ist, aber diffuse Beleuchtung nicht wiedergeben kann, und zum anderen das *Radiosity-Verfahren*, das diffuse Beleuchtung perfekt wiedergibt, aber keine spiegelnden Reflexionen beherrscht. Es gibt zahlreiche Ansätze, die beide Verfahren mit mehr oder minder großem Erfolg kombinieren.

### 4.1 Ray-Tracing

Ray-Tracing ist nicht nur das älteste, sondern auch das einfachste Verfahren zur Modellierung der globalen Beleuchtung. Es simuliert mühelos Effekte wie Schattenwurf, ideale spiegelnde Reflexion und Lichtbrechung, vernachlässigt aber die diffuse Beleuchtung, was zum Teil sehr synthetische Bilder mit zu scharfen Schattenwürfen und harten Übergängen zur Folge hat.

Der Algorithmus funktioniert folgendermaßen: Von einem angenommenen Betrachterstandpunkt aus wird durch jedes einzelne Bildpixel ein Strahl in die Szene geschossen. Trifft dieser Primärstrahl kein Objekt, erhält das Pixel die Hintergrundfarbe. Verläuft er aber durch ein oder mehrere Objekte, wird der vorderste Schnittpunkt mit einem Objekt errechnet und von diesem drei Arten von Sekundärstrahlen ausgesandt: ein perfekt reflektierter Strahl (vorausgesetzt, das Material ist nicht völlig matt und reflektiert), ein perfekt gebrochener Strahl (sofern das Objekt lichtdurchlässig ist) und die sogenannten Schattenstrahlen, die in Richtung aller Lichtquellen verlaufen. Kommt ein Schattenstrahl bei einer Lichtquelle an, ohne vorher von einem anderen Objekt blockiert worden zu sein, dann wird das betrachtete Objekt von dieser Lichtquelle direkt beleuchtet und sie muss bei der Berechnung der Helligkeit des Punktes mit einbezogen werden. Da der Spiegelungs- und der Brechungsstrahl ebenso wie zuvor der Primärstrahl behandelt werden müssen, sprich ihre Schnittpunkte mit anderen Objekten errechnet und von diesen wieder neue Sekundärstrahlen ausgesandt werden müssen, ist der Raytracing-Algorithmus rekursiv und könnte ohne weiteres bis ins unendliche laufen. Deshalb sollte man nicht vergessen, entweder eine maximale Rekursionstiefe oder aber eine minimale Lichtstärke festzulegen. Je tiefer aber die Rekursion, desto besser das Ergebnis.

Hat man nun also alle Sekundärstrahlen hinreichend verfolgt, kann man aus ihrer Art und Richtung und unter Berücksichtigung der Reflexionseigenschaften des Materials rekursiv und auf Basis des lokalen Beleuchtungsmodells die Lichtstärke im Schnittpunkt

des Primärstrahls mit dem Objekt berechnen. Die mit dem Ray-Tracing nicht modellierbare diffuse Reflexion wird üblicherweise durch einen konstanten Term für ambientes Licht berücksichtigt.

Da in Innenräumen der Anteil des ambienten Lichts zwischen 30-80% beträgt, ist der Raytracing-Algorithmus für ihre Berechnung nur bedingt geeignet.

## 4.2 Radiosity-Verfahren

Als Radiosity bezeichnet man den ausgehenden Energiefluss pro Zeit und Flächeneinheit, in unserem Fall natürlich den Energiefluss des Lichts. Das nach dieser Größe benannte Verfahren basiert auf einem ganz anderen Ansatz als das strahlenorientierte Ray-Tracing, so werden zum Beispiel alle Flächen als diffus angenommen und Spiegelungen werden ignoriert. Die Lichtquellen selbst werden ebenfalls als Objekte aufgefasst. Die diffusen Flächen haben zur Folge, dass die austretenden Strahlen nicht mehr von der Einfallsrichtung abhängig sind, wodurch die Berechnung deutlich vereinfacht wird. Ein weiterer angenehmer Nebeneffekt ist, dass das Radiosity-Verfahren auch nicht auf einen festgelegten Betrachterstandpunkt angewiesen ist, was es einem ermöglicht, die globale Beleuchtung einmalig für die gesamte Szene zu berechnen und nicht nur für eine bestimmte Position.

Zur Berechnung werden alle Oberflächen mit einem Netz von ebenen Oberflächenstücken überzogen, wobei die Lichtquellen, die ja als Objekte aufgefasst werden, mit einbezogen werden. Um den Lichttransport zwischen diesen Oberflächenstücken zu beschreiben, werden sogenannte Formfaktoren eingeführt, die den Anteil des Energieflusses angeben, der ausgehend von einem Stück bei einem anderen Stück ankommt. Sind die Formfaktoren berechnet, erhält man ein System linearer Gleichungen, das numerisch gelöst werden kann. Das ist zwar ziemlich zeitaufwändig, dafür hat man aber hinterher die Radiosity-Werte aller Oberflächenstücke und kann ohne größeren Aufwand bestimmte Ansichten generieren, was z.B. bei Echtzeit-Flügen durch eine Szene sehr hilfreich ist. Die Bilder selbst wirken natürlicher als beim Ray-Tracing, aber aufgrund der fehlenden Spiegelungs- und Glanzeffekte nicht immer realistisch.

# 5 Mapping-Techniken

Für die photorealistische Darstellung einer Szene reichen ausgeklügelte Farb- und Intensitätsberechnungen wie die bereits vorgestellten nicht immer aus. Oft möchte man Oberflächen auch eine bestimmte Struktur verleihen, wie zum Beispiel Holzmaserung, Verputzung, Rost oder Ähnliches. Aber auch für kleine plastische Details, die sich nur unter großem Aufwand als 3D-Objekte realisieren ließen, benötigt man effizientere Methoden. Deshalb greift man zu sogenannten Mapping-Techniken, von denen die drei wichtigsten im Folgenden kurz erläutert werden.

## 5.1 Texture-Mapping

Das Texture-Mapping ist eine der grundlegenden Mapping-Techniken. Dabei wird eine sogenannte Texture-Map verwendet, eine Matrix aus einzelnen Bildelementen („Texel“, von Textur und Pixel), die die Intensitäts- bzw. Farbinformationen enthält. In ihr wird die aufzutragende Textur in rechteckiger Form gespeichert und anschließend mittels einer Abbildung auf die entsprechende Oberfläche abgebildet. Um den Intensitäts- bzw. Farbwert für einen Pixel zu berechnen, müssen dessen vier Eckpunkte vom Bildschirmausschnitt über die darzustellende Oberfläche des Objekts zurück auf die Texture-Map abgebildet werden. Die vier Eckpunkte werden durch gerade Linien verbunden (nicht-ebene Objektoberfläche!), wodurch ein Viereck entsteht, das das Bild des ursprünglichen Pixels approximiert und andere Texel partiell überdecken kann. Zur Bestimmung des gewünschten Werts für den Pixel müssen nun die Intensitäts- bzw. Farbwerte dieser Texel dem Anteil ihrer Überdeckung entsprechend gewichtet und aufsummiert werden.

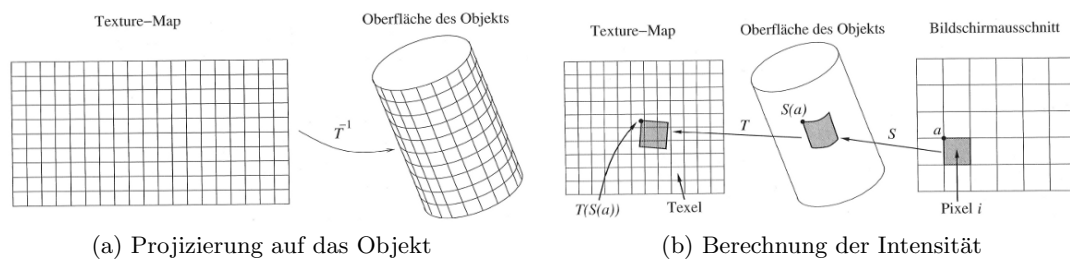


Abbildung 5.1: Texture-Mapping



## 5.2 Bump-Mapping

Das Bump-Mapping ist eine Variante des Texture-Mappings und ermöglicht die Darstellung reliefartiger Texturen. Dazu wird eine Bump-Map verwendet, die der Texture-Map ähnelt, aber Einträge der Form  $B(i,j)$  enthält, die angeben, um wieviel die Oberfläche im Gitterpunkt  $(i,j)$  in Normalenrichtung deformiert werden soll.  $B(u,v)$  kann auf der gesamten Textur als Funktion angewendet werden, indem einfach zwischen den Gitterpunkten  $(i,j)$  bilinear interpoliert wird.

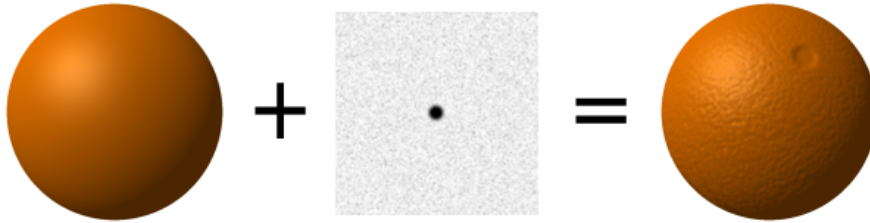


Abbildung 5.2: Anwendung des Bump-Mappings auf eine Kugel

## 5.3 Environment-Mapping

Das Environment-Mapping oder auch Reflection-Mapping ist eine weitere verwandte Technik, mit Hilfe derer sich Spiegelungen leicht realisieren lassen. Dazu wird die Umgebung des Objekts, in dem sie sich spiegeln soll, auf eine das Objekt umgebende Sphäre projiziert, die dann als zweidimensionale Texture-Map betrachtet wird. Auf diese Weise lassen sich Spiegelungen zwischen Objekten auf einfachere Art als mittels Ray-Tracing realisieren.

# 6 Quellenangabe

## Literatur

Bungartz, Griebel, Zenger: Einführung in die Computergraphik Vieweg (2002)

Schneider, Werner: Taschenbuch der Informatik Fachbuchverlag Leipzig (2001)

Zeppenfeld: Lehrbuch der Grafikprogrammierung Spektrum (2004)

## Abbildungen

- 2.1 Ansätze zur Intensitätsberechnung:  
Einführung in die Computergraphik (s. Literatur)
  
- 3.1 Sichtbare Fläche:  
Einführung in die Computergraphik (s. Literatur)
  
- 3.2 Shading-Techniken am Beispiel einer Kugel:  
[http://img.zdnet.com/techDirectory/\\_SHADING.GIF](http://img.zdnet.com/techDirectory/_SHADING.GIF)
  
- 3.3 Transparentes Polygon:  
Einführung in die Computergraphik (s. Literatur)
  
- 3.4 Screen-Door-Transparenz:  
Einführung in die Computergraphik (s. Literatur)
  
- 5.1 Texture-Mapping:  
Einführung in die Computergraphik (s. Literatur)
  
- 5.2 Anwendung des Bump-Mappings auf eine Kugel:  
[http://upload.wikimedia.org/wikipedia/commons/e/ef/Bump\\_mapping.png](http://upload.wikimedia.org/wikipedia/commons/e/ef/Bump_mapping.png)