



UNIVERSITÄT  
LEIPZIG

Beleuchtung  
**COMPUTERGRAPHIK**

# Inhaltsverzeichnis

## 9 Beleuchtung

9.1 Farbe

9.2 Modelle

9.3 Lokale Beleuchtungsmodelle

9.4 Schattierung

9.5 Globale Beleuchtung

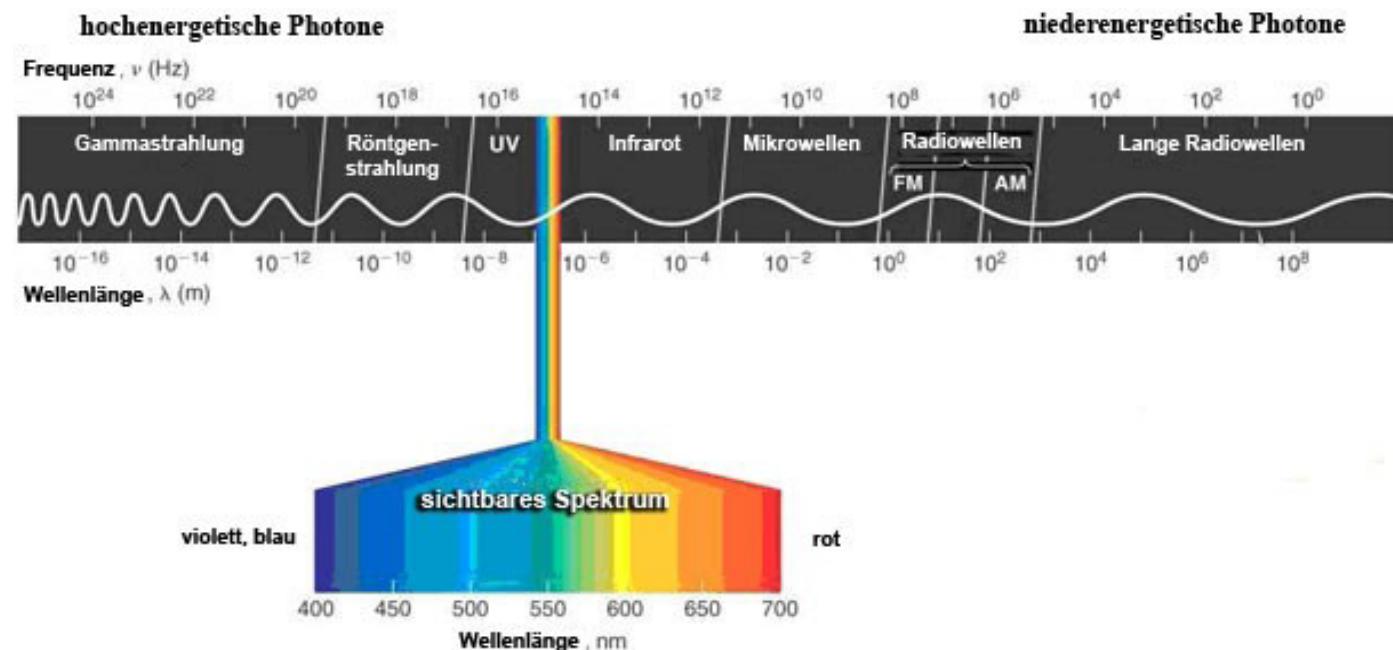
## 9.1 Farbe

- Farbe ist ein wesentlicher Bestandteil realistischer Computergraphik
  - Die Berechnung der Beleuchtung basiert darauf, für jeden darzustellenden Punkt einen Farbwert zu bestimmen
- 
- Fragestellungen:
    - Wie kann man eine bestimmte Farbe exakt spezifizieren?  
⇒ Farbräume, Farbmodelle
    - Wie exakt kann eine exakt spezifizierte Farbe auf einem Ausgabegerät angezeigt werden?  
⇒ geräteabhängige Farbräume
    - Wie viele verschiedene Farben können durch eine Graphikhardware bereitgestellt werden?  
⇒ technische Realisierbarkeit

## 9.1 Farbe

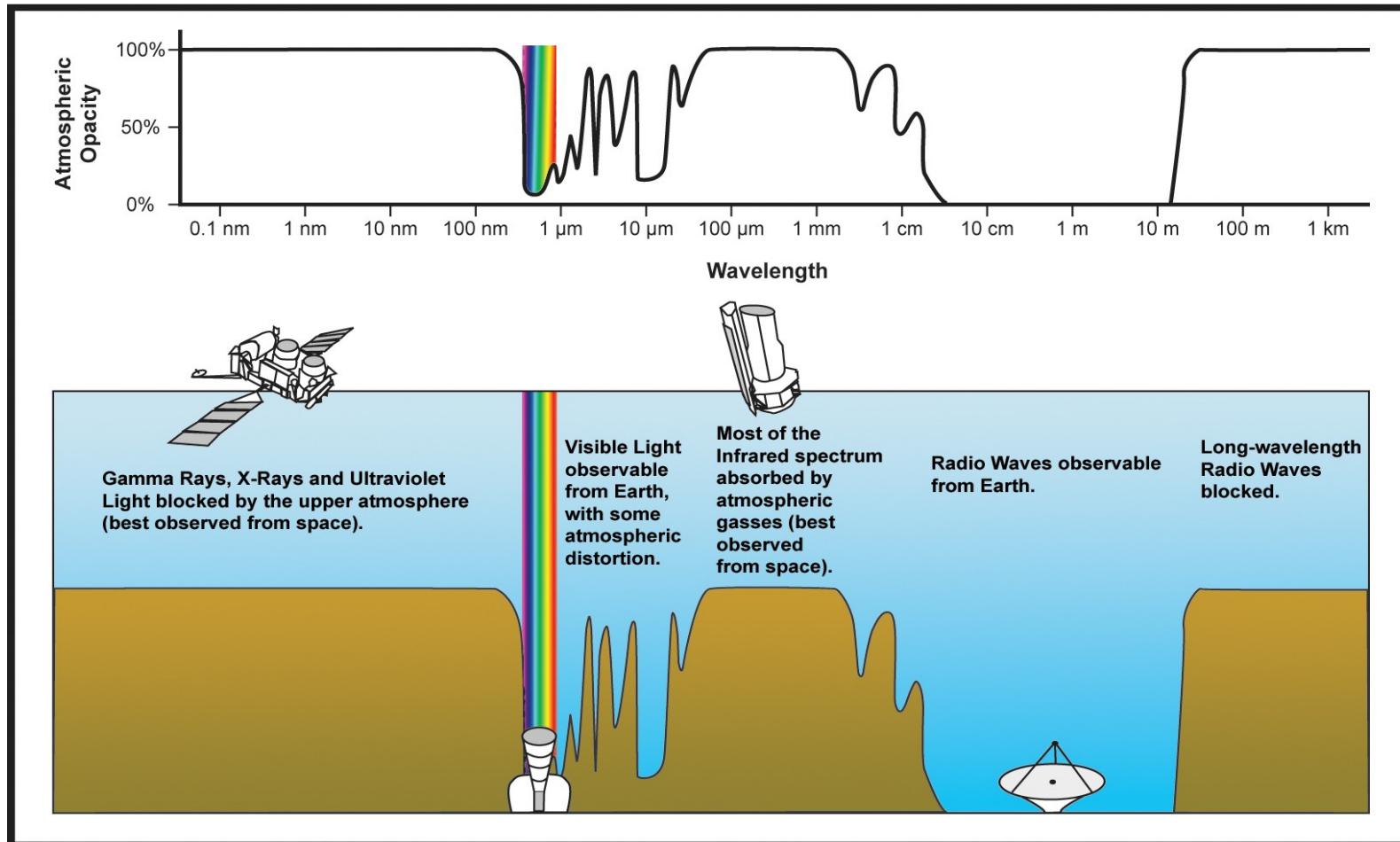
### Licht und das sichtbare Spektrum

- Das sichtbare Lichtspektrum (400 – 700 nm) ist ein kleiner Teil des elektromagnetischen Spektrums.
- Manche Tiere können
  - im Infrarotbereich sehen (Schlangen)
  - ultraviolettes Licht wahrnehmen (Insekten)



## 9.1 Farbe

### Elektromagnetische Wellen auf der Erde

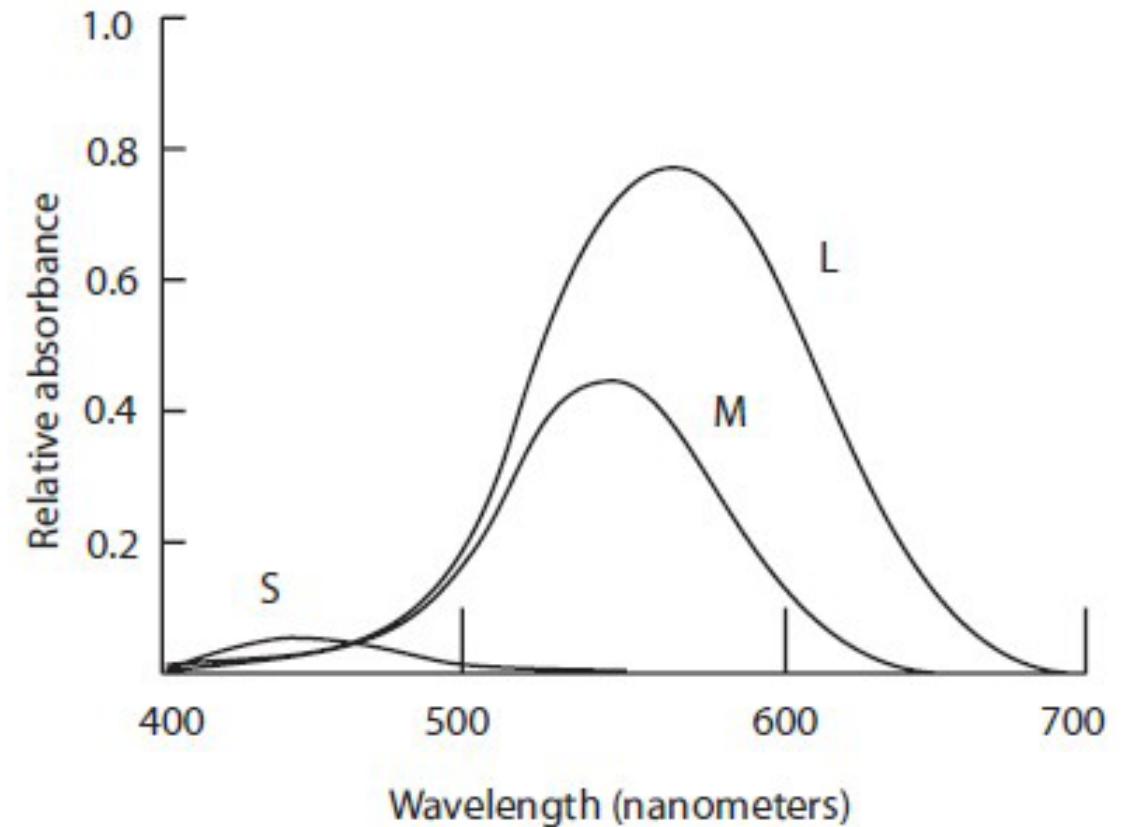


[[http://en.wikipedia.org/wiki/File:Atmospheric\\_electromagnetic\\_transmittance\\_or\\_opacity.jpg](http://en.wikipedia.org/wiki/File:Atmospheric_electromagnetic_transmittance_or_opacity.jpg)]

## 9.1 Farbe

### Dreifarbenlehre

- Der Mensch besitzt drei Typen von Zapfen
  - Jeder Zapfen hat seine maximale Sensitivität bei einer anderen Wellenlänge
  - S = **blau**, M = **grün**, L= **rot**
  - Ein Farbsystem bestehend aus drei Grundtönen reicht aus, um alle vom Menschen wahrnehmbaren Farben darzustellen
- Dreifarbenlehre



## 9.1 Farbe

### Dreidimensionaler Farbraum

- Je nach Anforderung können verschiedene standardisierte Farträume (Farbmodelle) verwendet werden
  - RGB
    - Additives Farbmodell
    - Monitore
    - Computergraphik
  - CMY
    - Subtraktives Farbmodell
    - Drucker
- HSV
  - zerlegt Farbe in Farbwert, Helligkeit und Sättigung
- CIE
  - internationaler Standard zur Farbspezifikation

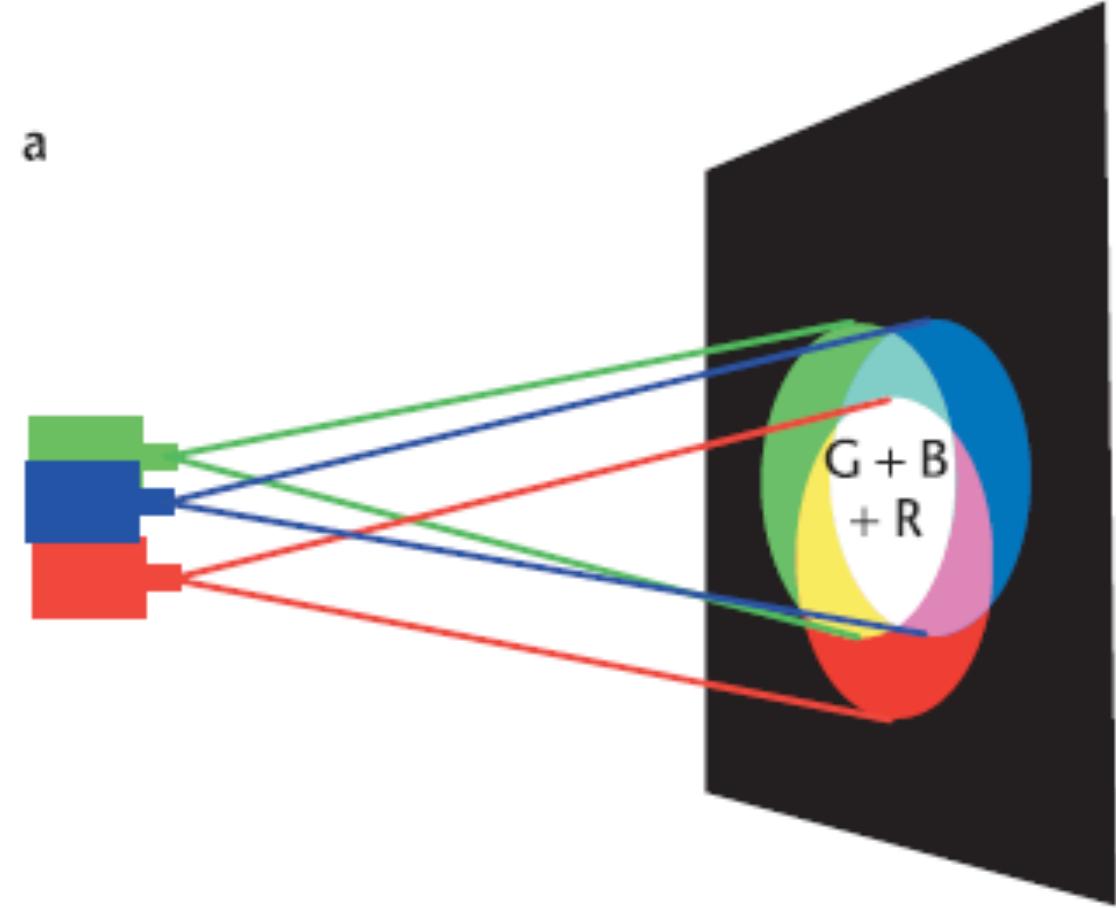
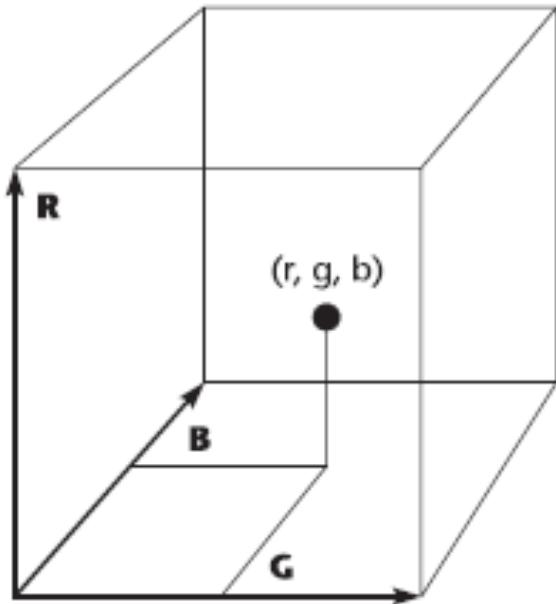
## 9.1 Farbe

### RGB

- Man kann eine beliebige (Licht-)Farbe C mittels der Grundfarben Rot (R), Grün (G) und Blau (B) als

$$C = rR + gG + bB$$

ausdrücken



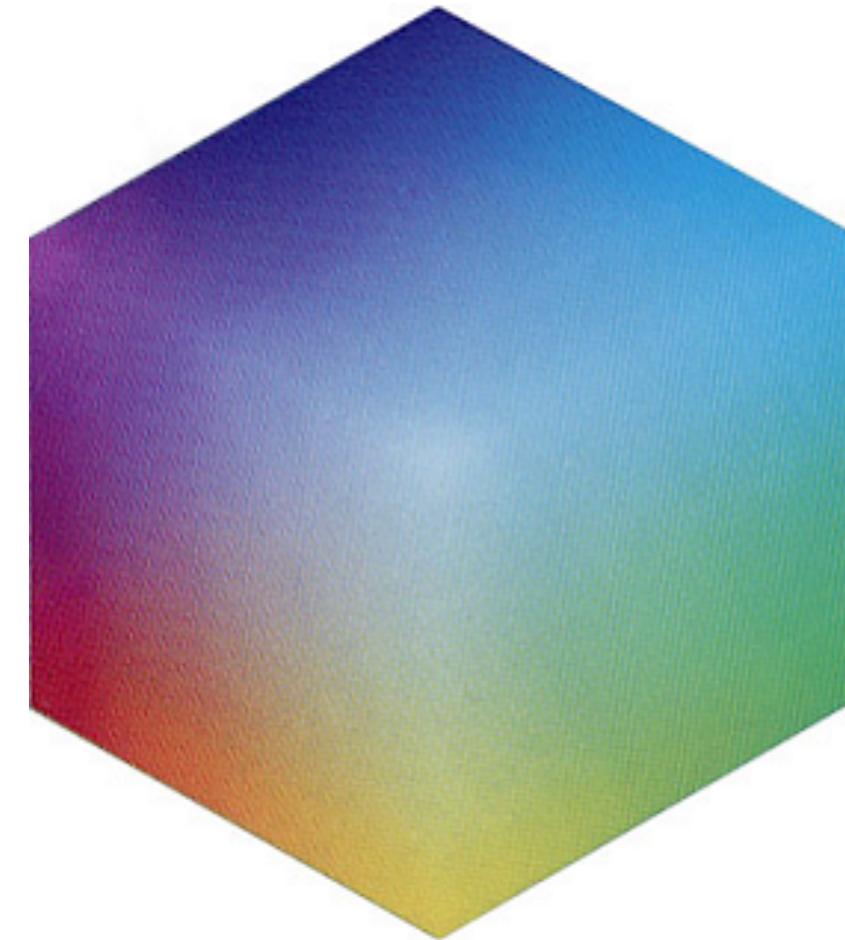
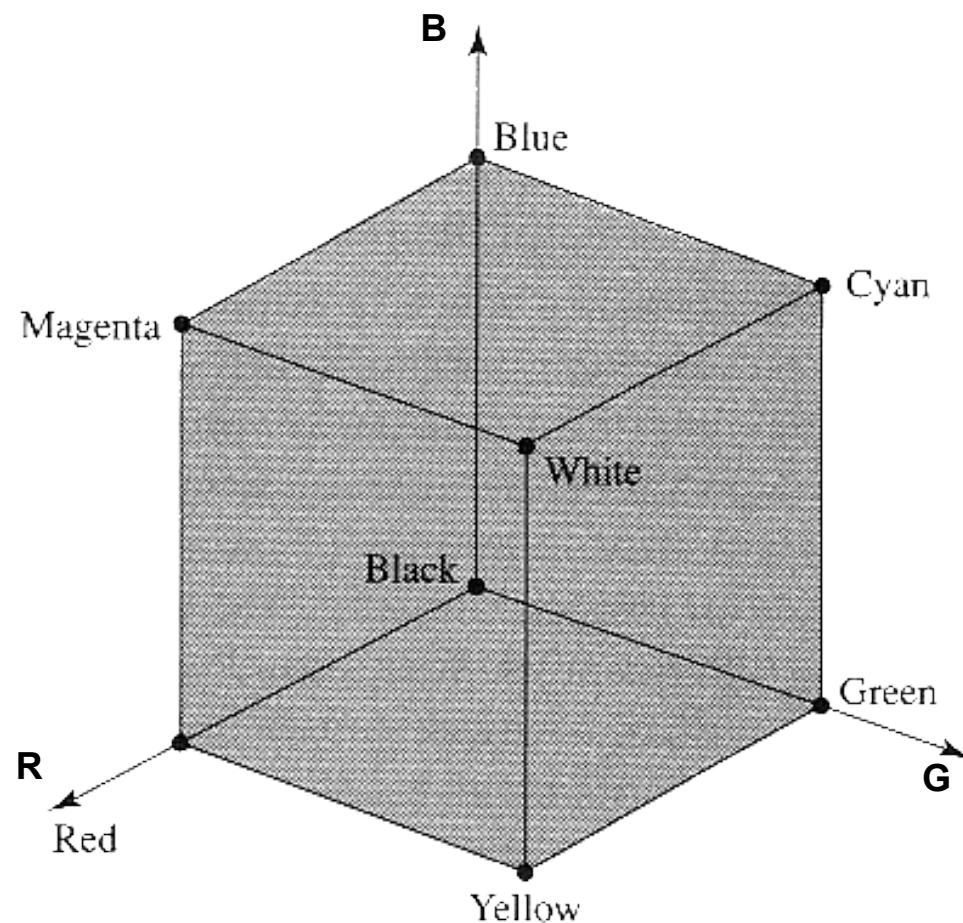
## 9.1 Farbe

### RGB

- Farben in den Eckpunkten des Würfels
  - $(0, 0, 0)$  = Schwarz
  - $(1, 0, 0)$  = Rot
  - $(0, 1, 0)$  = Grün
  - $(0, 0, 1)$  = Blau
  - $(1, 1, 0)$  = Gelb
  - $(1, 0, 1)$  = Magenta
  - $(0, 1, 1)$  = Cyan
  - $(1, 1, 1)$  = Weiß

## 9.1 Farbe

RGB



## 9.1 Farbe

### RGB

- Das Modell ist bezüglich der Farbwahrnehmung nicht linear
- Bei typischer Farbauflösung von 8 Bit pro Grundfarbe
  - Im Farbwürfel existieren Regionen
    - in denen benachbarte Punkte für das Auge denselben Farbeindruck hervorrufen
    - die Farben benachbarter Punkte für das Auge sehr unterschiedlich sind
  - Für den Anwender kann es schwierig sein
    - zu einer gewünschten Farbe ein entsprechendes (r, g, b)-Tupel zu ermitteln
    - Eine Farbe etwas abzuschwächen (erfordert ungleichmäßige Änderungen von r, g und b)

## 9.1 Farbe

### HSV

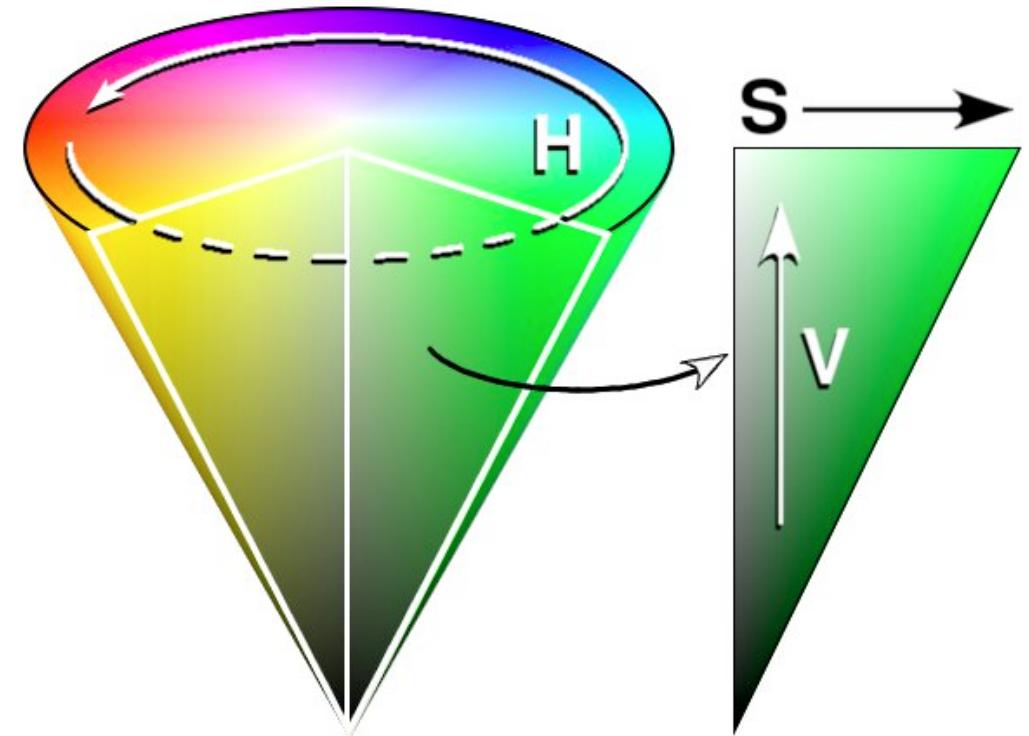
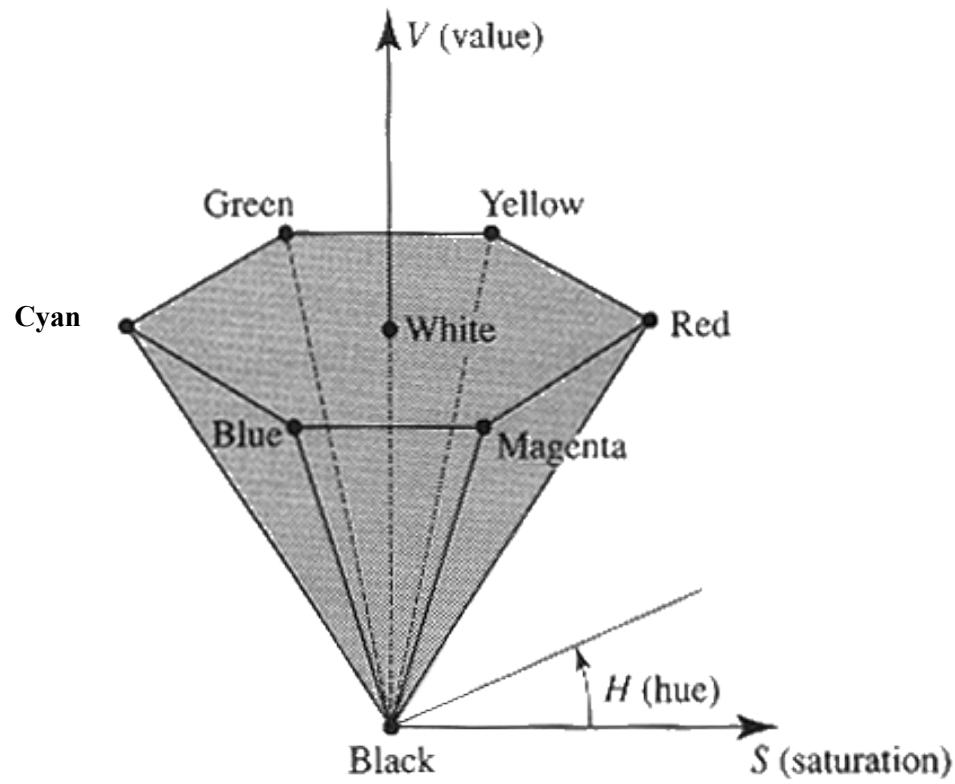
- Dieses Modell wurde zur Unterstützung einer intuitiven Farbauswahl entwickelt (wahrnehmungsorientiertes Farbmodell)
- Farbkörper im 3D-Farbraum ist Pyramide mit sechseckiger Grundfläche
- verwendet Polarkoordinaten:
  - Hue: Farbe („Farbfamilie“) als „Farbwinkel“ in Grad
$$0^\circ \leq H < 360^\circ$$
  - Saturation: Sättigung
$$0 \leq S \leq 1$$

(Verkleinerung addiert Weiß)
  - Value: Helligkeit
$$0 \leq V \leq 1$$

(Verkleinerung addiert Schwarz)

## 9.1 Farbe

HSV



## 9.1 Farbe

### HSV – RGB

- Die Grundfläche der HSV-Pyramide entsteht aus dem RGB-Würfel durch Projektion entlang der Raumdiagonalen von Weiß nach Schwarz auf eine dazu senkrecht stehende Ebene
- Im HSV-Modell besitzen Komplementärfarben eine Winkeldifferenz von  $180^\circ$  im  $H$ -Wert
- Korrespondierende Punkte

RGB	Farbe	HSV
(1, 0, 0)	Rot	(0, 1, 1)
(1, 1, 0)	Gelb	(60, 1, 1)
(0, 1, 0)	Grün	(120, 1, 1)
(0, 1, 1)	Cyan	(180, 1, 1)
(0, 0, 1)	Blau	(240, 1, 1)
(1, 0, 1)	Magenta	(300, 1, 1)

## 9.1 Farbe

### CMY

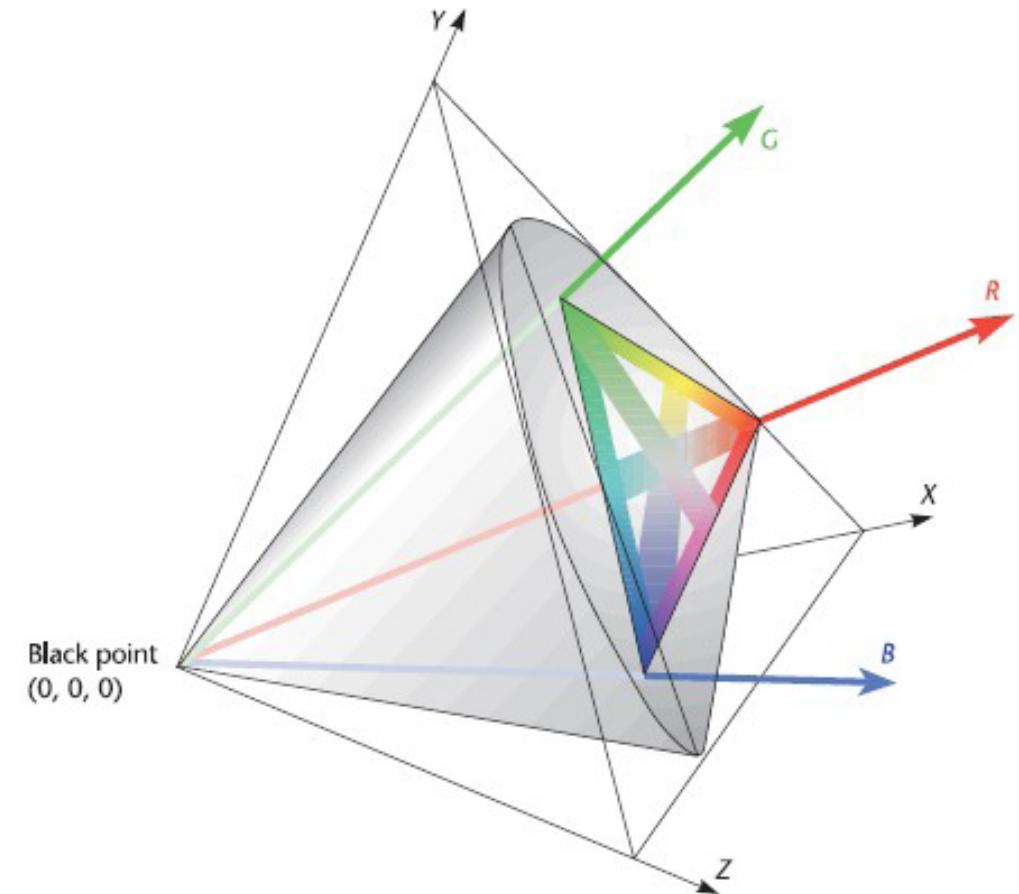
- Cyan-Magenta-Yellow-(CMY)-Farbraum
- Oft kommt es bei der Farbwiedergabe auf Papier zu kleineren Differenzen
- Drucker verwenden häufig noch
  - Schwarz
  - zusätzliche Farben
- Berechnung

$$\begin{pmatrix} C \\ M \\ Y \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

## 9.1 Farbe

### CIE Standard für Farben

- Das Commission Internationale de l'Eclairage (CIE) System verwendet drei abstrakte Primärfarben X, Y, Z
- Wahrnehmbare Farben sind als graues Volumen dargestellt (die Messungen stammen von vor 1931)
- Die Farben, welche durch drei farbige Lichter in rot, grün und blau erzeugt werden können, sind durch die einbeschriebene Pyramide gekennzeichnet



## 9.1 Farbe

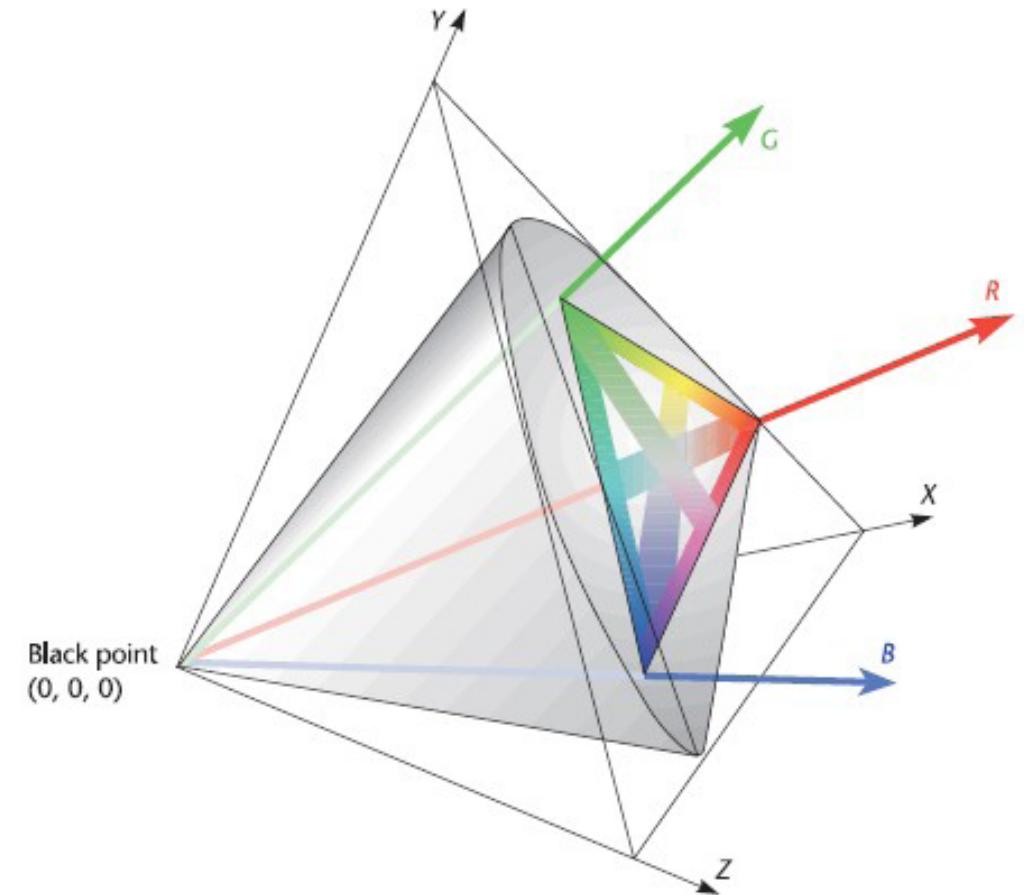
### CIE Standard für Farben

- Farben in XYZ Koordinaten sind nur schwer zu verstehen
- Leichter verständlich ist die Repräsentation in Farbwertanteilen (chromaticity coordinates)

$$x = \frac{X}{X + Y + Z}$$

$$y = \frac{Y}{X + Y + Z}$$

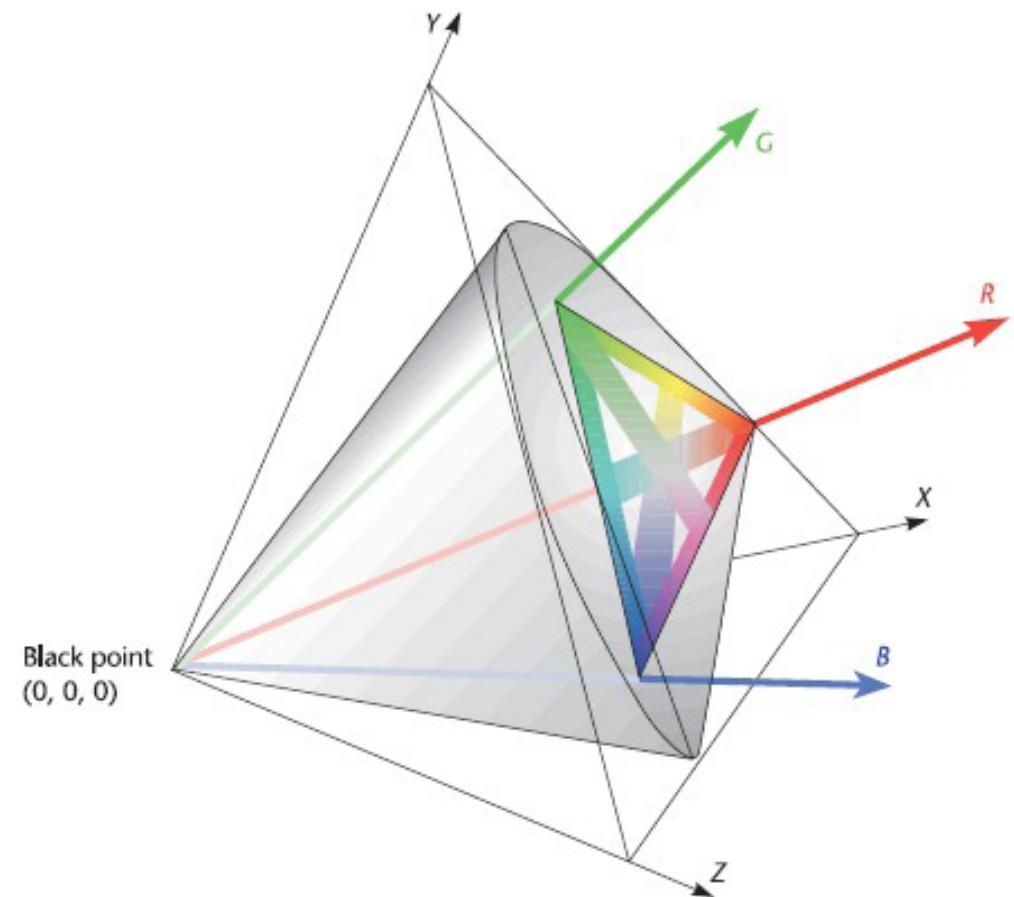
$$z = \frac{Z}{X + Y + Z}$$



## 9.1 Farbe

### CIE Standard für Farben

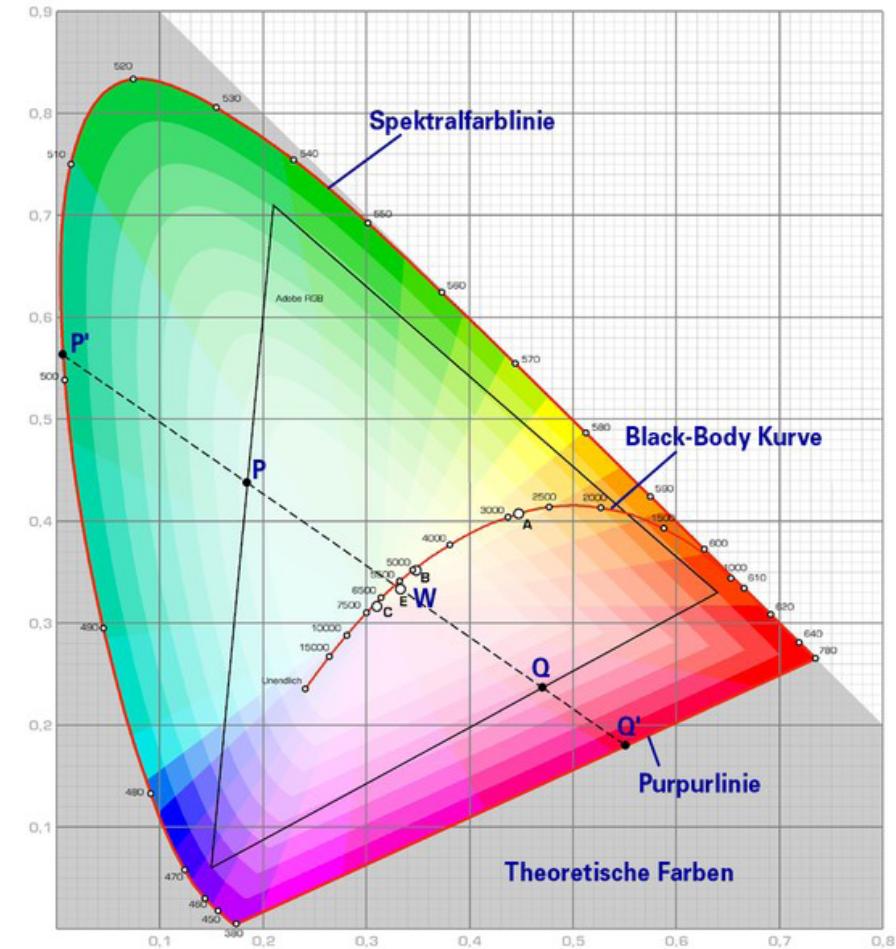
- Wegen  $x + y + z = 1$ , reicht es die x und y Werte anzugeben
- Üblicherweise werden Farben in Form von  $(x, y, Y)$  spezifiziert, wobei Y die Helligkeit angibt



## 9.1 Farbe

### CIE Normfarbtafel (Eigenschaften)

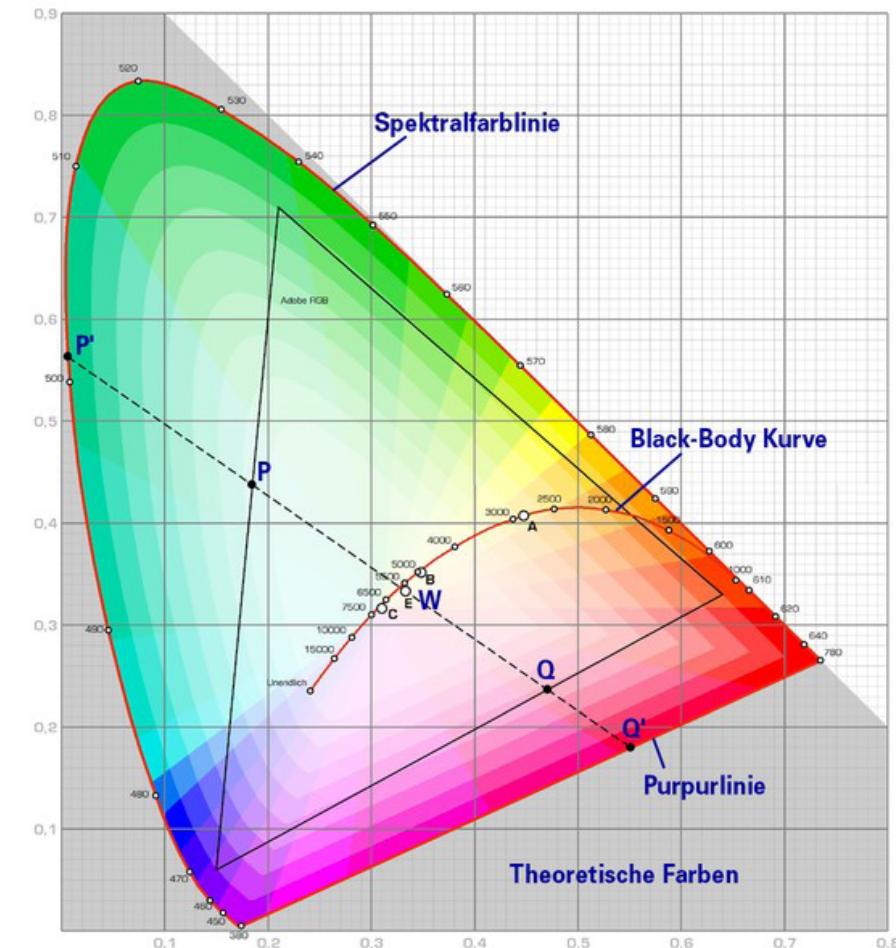
- Wenn zwei farbige Lichtquellen als Punkte im Diagramm dargestellt werden, liegen die Farben, welche als Mischung dieser beiden Lichtquellen erzeugt werden können, auf einer Linie
- Alle Farben innerhalb eines Dreiecks, dessen Eckpunkte drei farbige Lichtquellen repräsentieren, können mit Hilfe dieser Lichtquellen erzeugt werden



## 9.1 Farbe

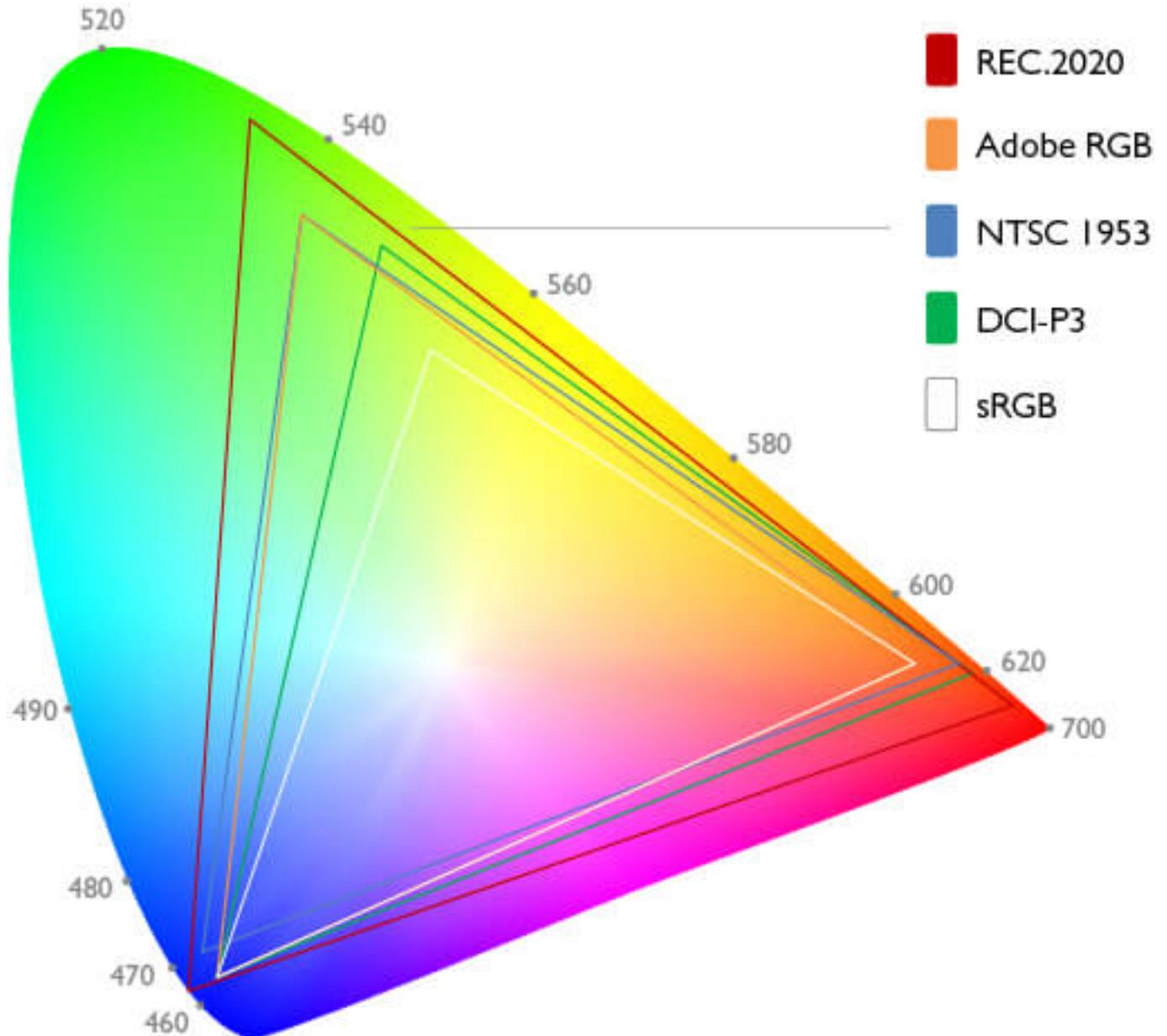
### CIE Normfarbtafel (Eigenschaften)

- Die hufeisenförmige Spektralfarblinie enthält alle Farben mit den höchsten Sättigungsgraden in den einzelnen Farbtönen. Man sieht sie, wenn man Licht betrachtet, welches nur aus einer Wellenlänge besteht
- Die Purpurlinie verbindet die beiden Enden der Spektralfarblinie



## 9.1 Farbe

### Gamut (Farbbereich)



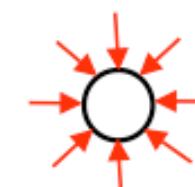
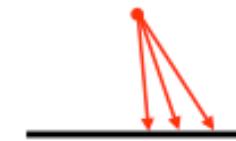
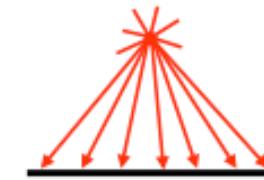
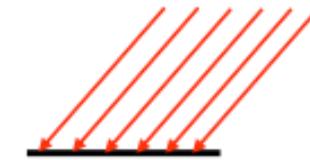
## 9.2 Modelle

- Beleuchtungsmodelle und Reflexionsmodelle
  - dienen zur Bestimmung der Farbe an einem Punkt eines Objektes
  - Inkonsistente Terminologie in der Literatur
    - Illumination-Model
    - Lighting-Model
    - Reflection-Model
    - NICHT: Shading-Model
- Schattierungsmodelle
  - dienen zur Bestimmung der Farbe eines Pixels auf dem Monitor

## 9.2 Modelle

### Lichtquellen

- Gerichtete Lichtquellen
  - Position im Unendlichen  $(x, y, z, 0)^T$
- Punktlichtquellen
  - Endliche Position  $(x, y, z, 1)^T$
- Spotlichtquellen
  - Cutoff-Winkel, Position, Richtung
- Ambientes Licht
  - Gleichförmig aus allen Richtungen



## 9.2 Modelle

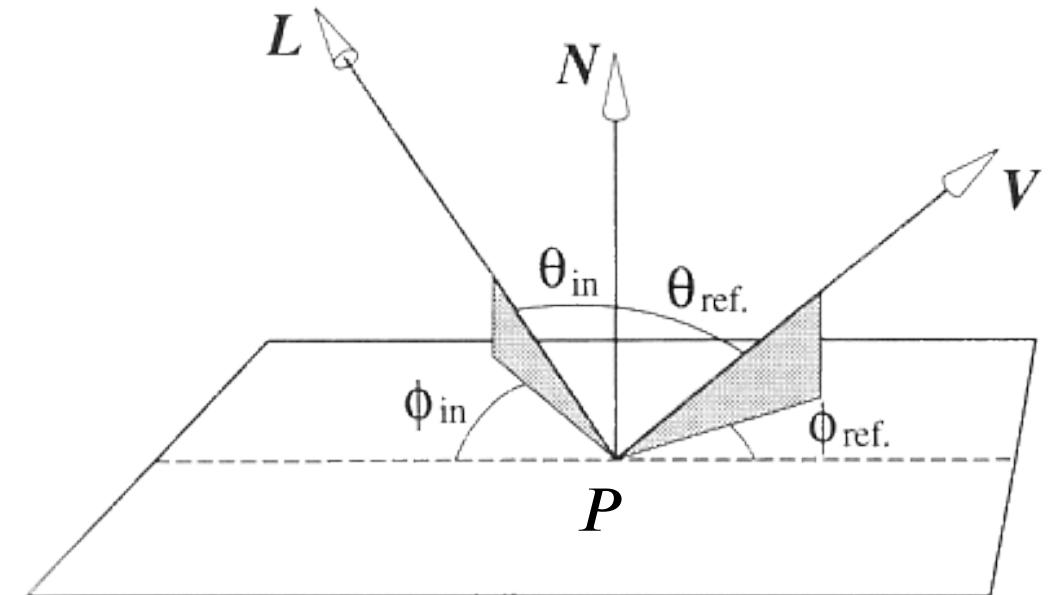
### Beleuchtungsmodelle

- Lokal
  - nur direkte Beleuchtung
  - Berechnung der Intensität (Farbe) eines Punktes in Abhängigkeit von
    - direktem Lichteinfall einer Lichtquelle
- Global
  - direkte und indirekte Beleuchtung
  - Berechnung der Intensität (Farbe) eines Punktes in Abhängigkeit von
    - direktem Lichteinfall einer Lichtquelle
    - indirekt einfallendem Licht
    - Reflexion
    - Transmission
  - benutzen oft erweiterte lokale Beleuchtungsmodelle

## 9.3 Lokale Beleuchtungsmodelle

### Geometrische Betrachtung

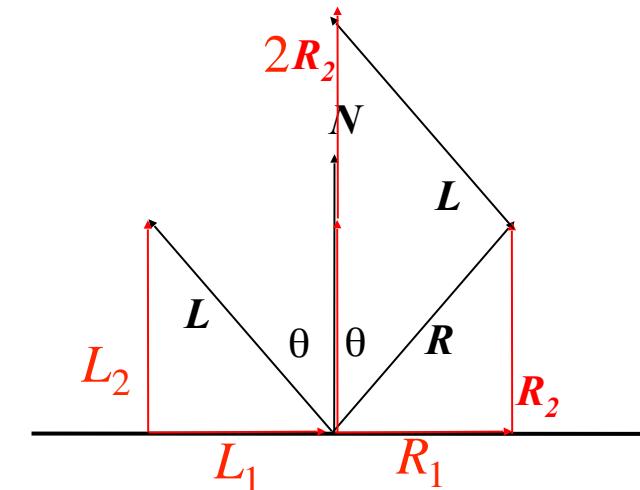
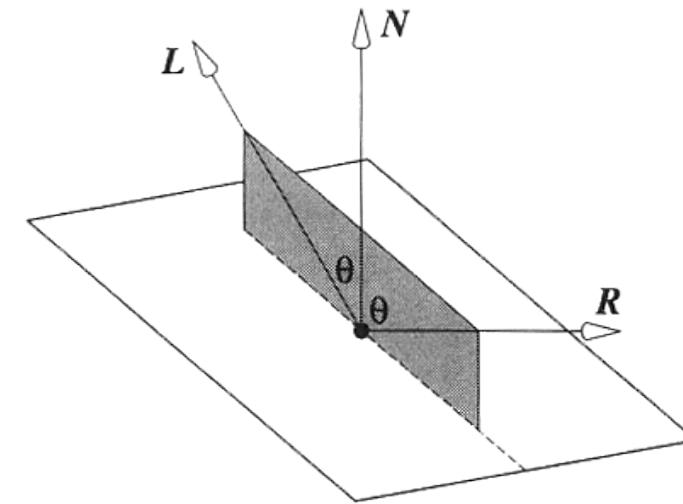
- $P$ : Punkt auf Objektoberfläche
- $N$ : Flächennormalenvektor in  $P$
- $L$ : Vektor von  $P$  zu einer Punktlichtquelle, normiert
- $V$ : Vektor von  $P$  zum Augpunkt (ViewingPoint), normiert
- $\phi_{in}, \Theta_{in}$ : Einfallswinkel
- $\phi_{ref}, \Theta_{ref}$ : Reflektionswinkel



## 9.3 Lokale Beleuchtungsmodelle

### Geometrische Betrachtung

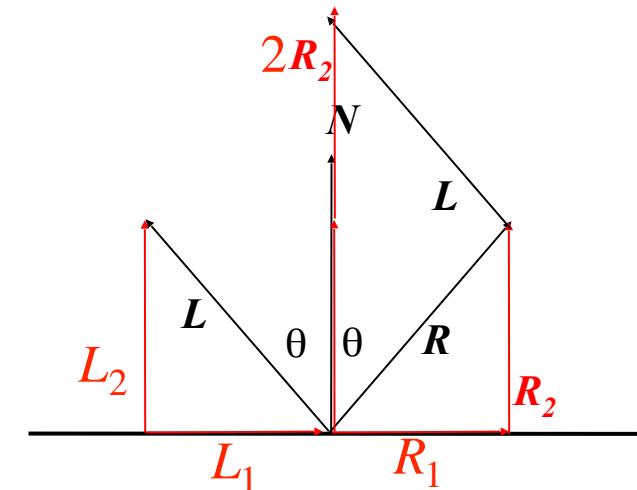
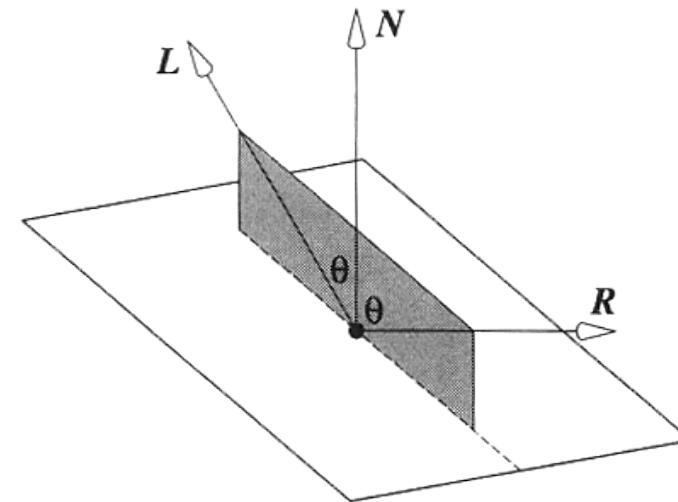
- $R$  : Vektor des reflektierten Lichts
- Es gilt
  - $L$  und  $R$  liegen in einer Ebene
  - $\Theta = \Theta_{in} = \Theta_{ref}$



## 9.3 Lokale Beleuchtungsmodelle

### Geometrische Betrachtung

- $R$  : Vektor des reflektierten Lichts
- Es gilt
  - $L$  und  $R$  liegen in einer Ebene
  - $\Theta = \Theta_{in} = \Theta_{ref}$
- $L = -L_1 + L_2$
- $R = R_1 + R_2$
- $R_2 = L_2 = (L \cdot N)N$
- $R_1 = L_1 = L_2 - L$
- $R = 2R_2 - L = 2(L \cdot N)N - L$



## 9.3 Lokale Beleuchtungsmodelle

### Einschub Fast Inverse Square Root

- Das Berechnen von Einheitsvektoren war lange Zeit sehr teuer
  - Benötigt Reziproke einer Gleitkommazahl
- FISR Methode konnte dies aber approximieren!



## 9.3 Lokale Beleuchtungsmodelle

### Einschub Fast Inverse Square Root

```
float Q_rsqrt( float number )
{
    long i;
    float x2, y;
    const float threehalves = 1.5F;

    x2 = number * 0.5F;
    y = number;
    i = *( long * ) &y;                                // evil floating point bit level hacking
    i = 0x5f3759df - ( i >> 1 );                      // what the fuck?
    y = *( float * ) &i;
    y = y * ( threehalves - ( x2 * y * y ) );        // 1st iteration
// y = y * ( threehalves - ( x2 * y * y ) );        // 2nd iteration, this can be removed

    return y;
}
```

- Die Methode wird John Carmack zu geschrieben, ist aber wohl älter
- Wurde zuerst in Quake 3 eingesetzt
- Heutzutage nur noch in FPGAs oder Hardware verwendet



## 9.3 Lokale Beleuchtungsmodelle

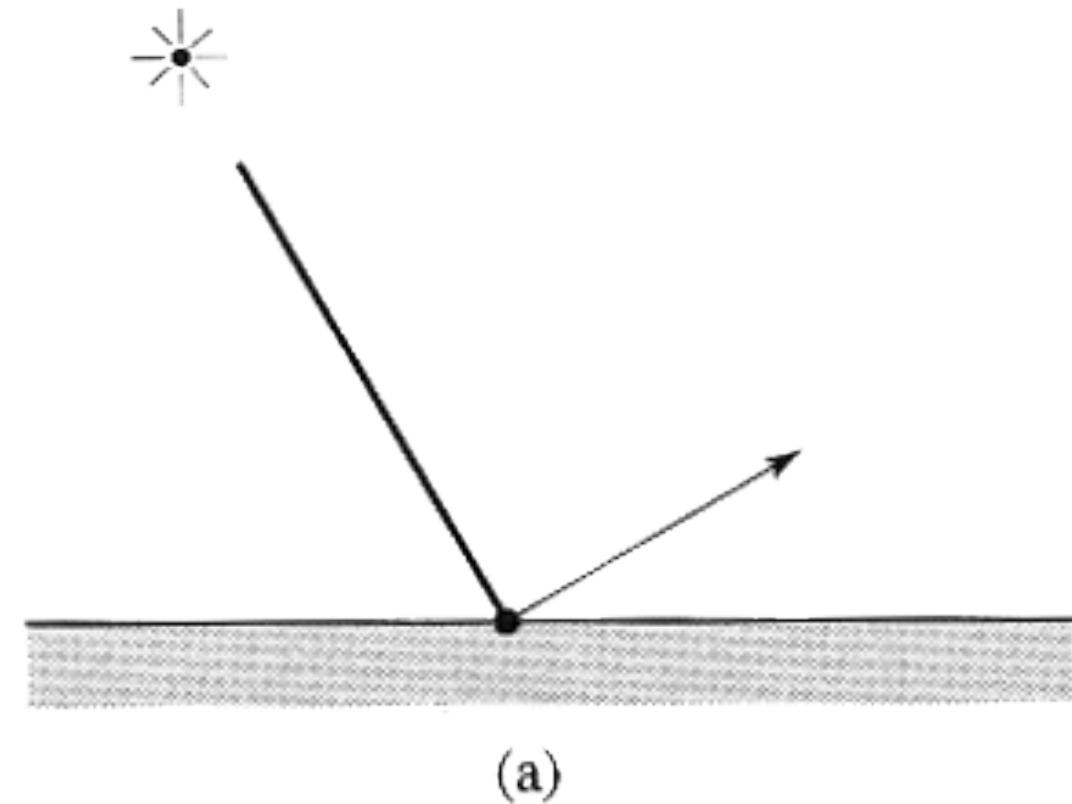
### Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- ACHTUNG
  - Nicht verwechseln mit Phong Shading
- Am häufigsten verwendetes lokales Beleuchtungsmodell
- Empirisches Modell
  - ohne physikalische Basis
  - mit guten praktischen Resultaten
- Das Modell simuliert folgende physikalischen Reflexionsphänomene:
  - Unvollkommen spiegelnde Reflexion
  - Perfekte/vollkommene diffuse Reflexion

## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

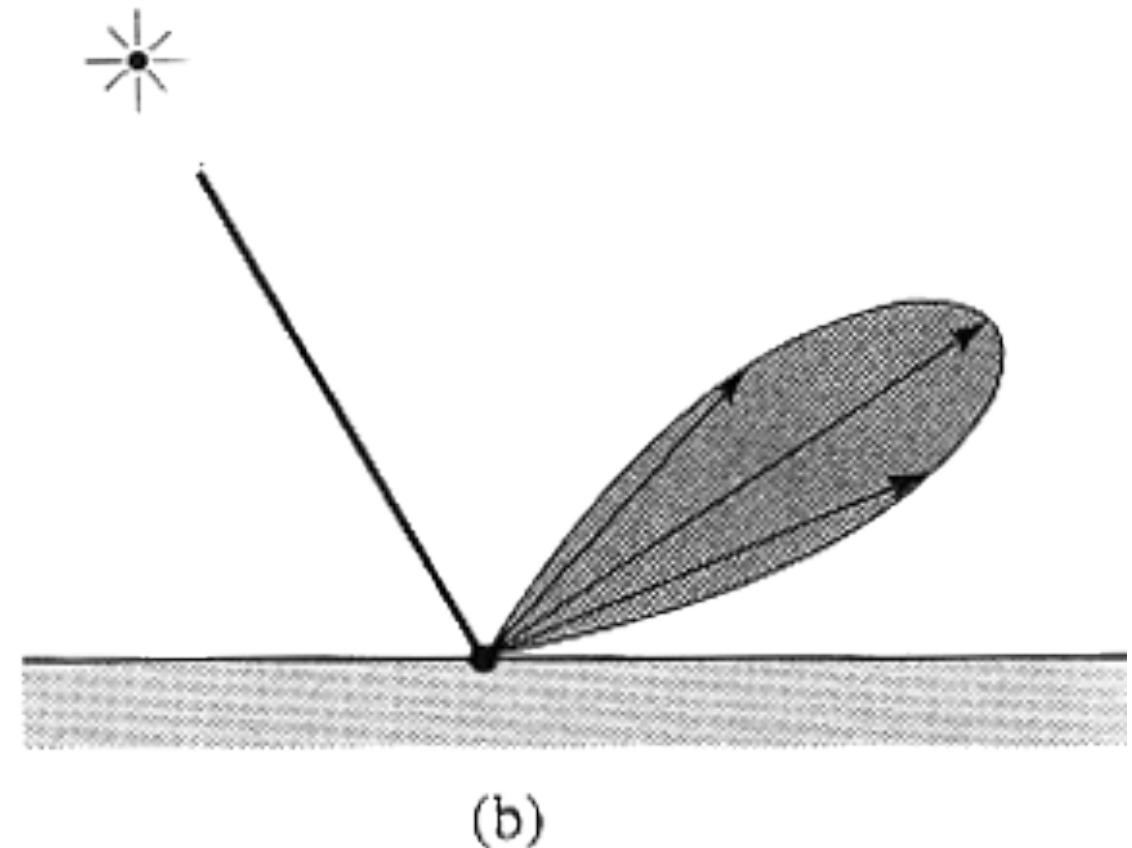
- Perfekte/vollkommene spiegelnde Reflexion
  - Ein Lichtstrahl wird, ohne sich aufzustreuen, perfekt nach dem Reflexionsgesetz reflektiert
  - Oberfläche: idealer Spiegel
    - existiert in der Realität nicht



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

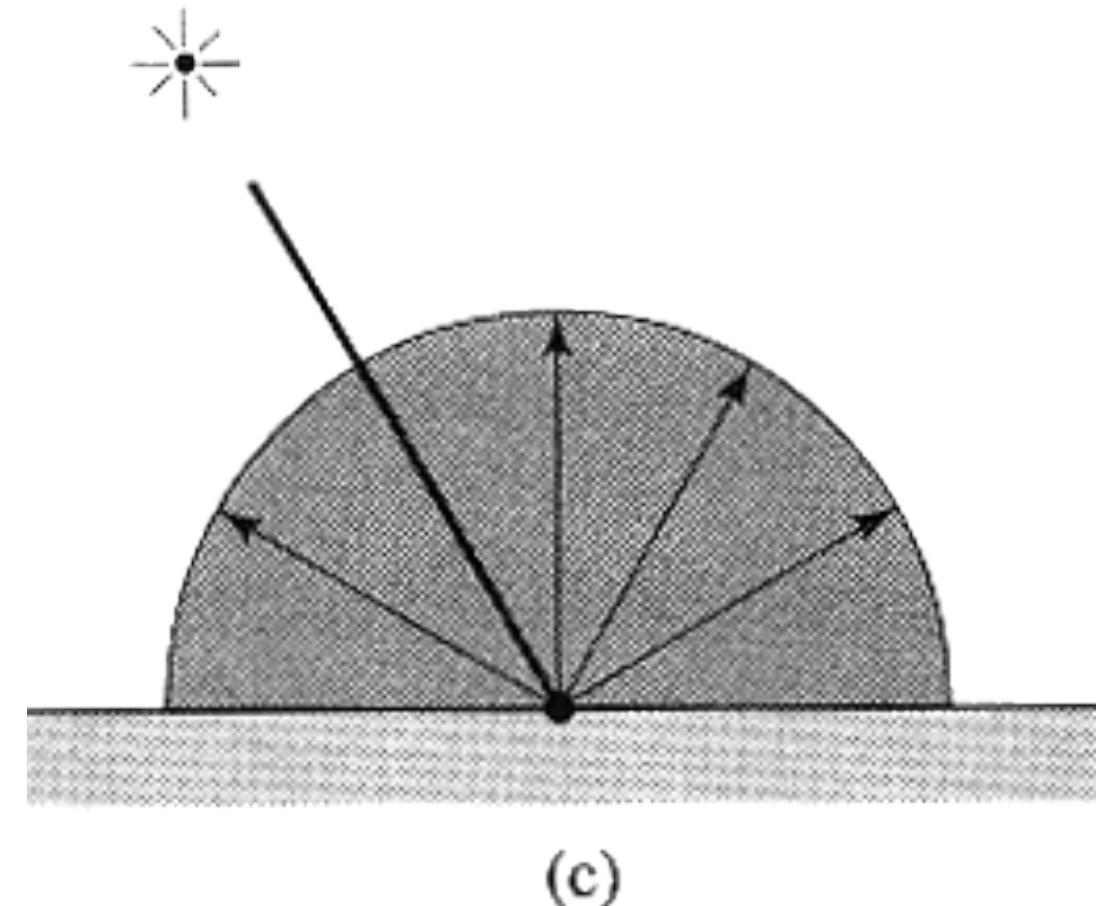
- Unvollkommene spiegelnde Reflexion
  - Der Lichtstrahl wird bei der Reflexion „aufgespalten“
  - es entsteht ein Reflexionskonus um die ausgezeichnete Reflexionsrichtung
- Oberfläche: unvollkommener Spiegel, raue Oberfläche
- Oberflächenelement ist mikroskopisch aus vielen kleinen perfekten Spiegeln mit leicht unterschiedlichen Ausrichtungen zusammengesetzt



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- Perfekte/vollkommene diffuse Reflexion
  - Der Lichtstrahl wird bei der Reflexion perfekt gestreut, d.h. mit gleichmäßiger Intensität in alle Richtungen
  - Oberfläche: ideale matte Oberfläche
    - existiert in der Realität nicht
    - annäherungsweise: feine Lage Puder



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- Ambientes Licht: Hilfskonstruktion
  - Konstant gewählt
  - Simuliert die globale bzw. die indirekte Beleuchtung
  - Grundbeleuchtung notwendig
    - Einige Objekte erhalten kein Licht von Lichtquelle(n): würden in dem Modell schwarz dargestellt
    - In der Realität werden solche Objekte indirekt beleuchtet
  - Ersatz für komplexere Berechnungen

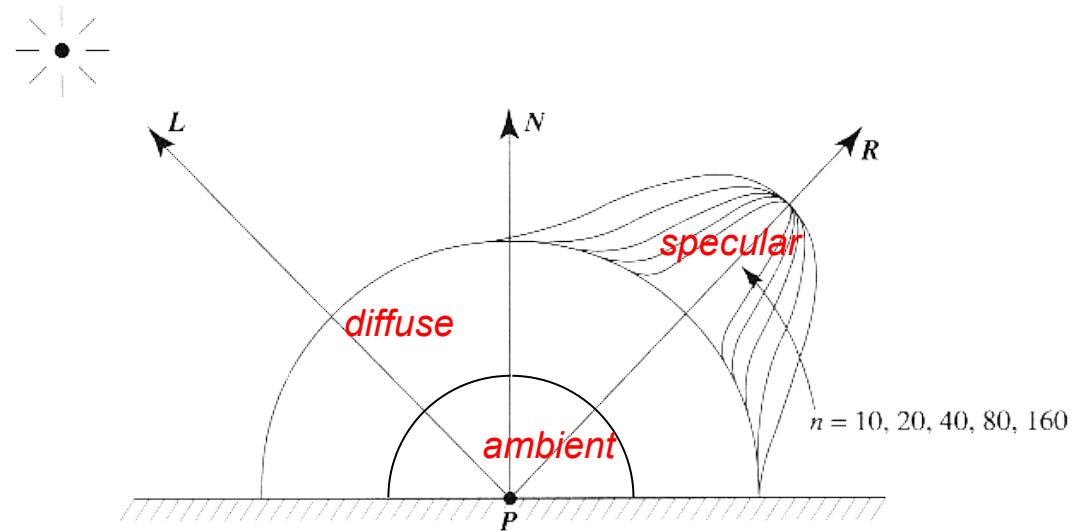
## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- Das reflektierte Licht (Intensität) eines Oberflächenpunktes wird dann aus den drei Reflexionskomponenten linear zusammengesetzt:

$$I = k_s \cdot I_s + k_d \cdot I_d + k_a \cdot I_a$$

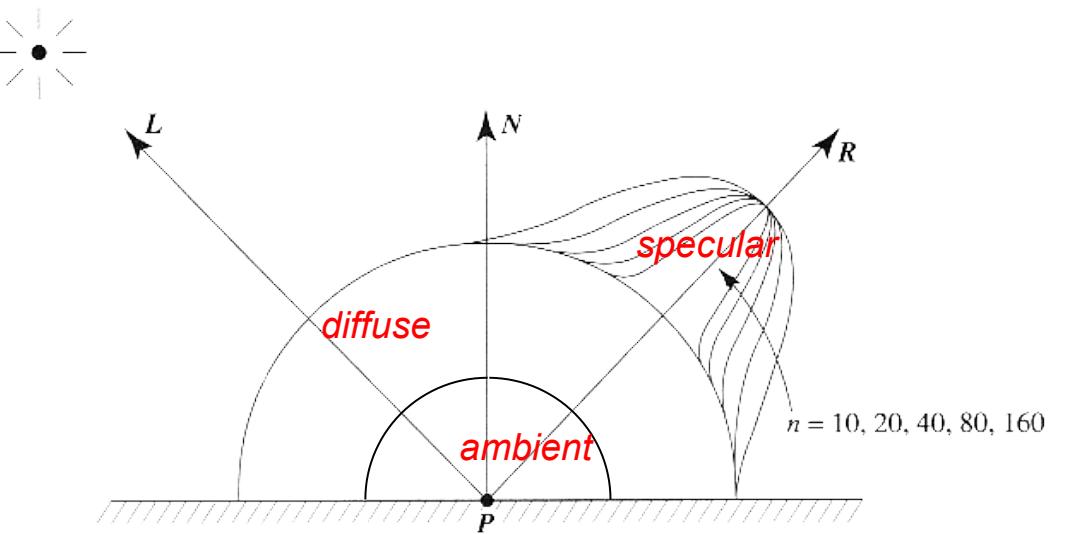
- $I_s$ : spiegelndes Licht (specular)
- $I_d$ : diffuses Licht
- $I_a$ : ambientes Licht (Umgebung)



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

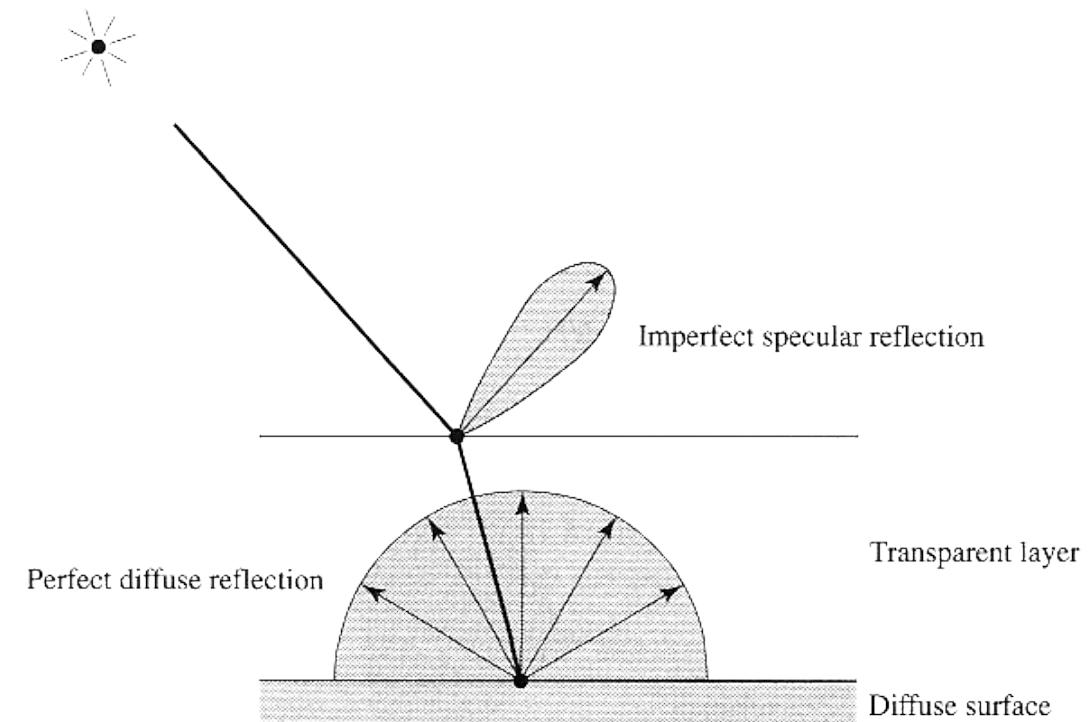
- $I = k_s \cdot I_s + k_d \cdot I_d + k_a \cdot I_a$
- Die Physik der Oberfläche wird über die Verhältnisse der einzelnen Komponenten modelliert
- Es gilt:  $k_s + k_d + k_a = 1$



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- Die lineare Kombination von spiegelnder und diffuser Reflexion entspricht zum Beispiel der Physik polierter Oberflächen
  - Poliertes Holz
    - Transparente Schicht: spiegelnd
    - Oberfläche: diffus



## 9.3 Lokale Beleuchtungsmodelle

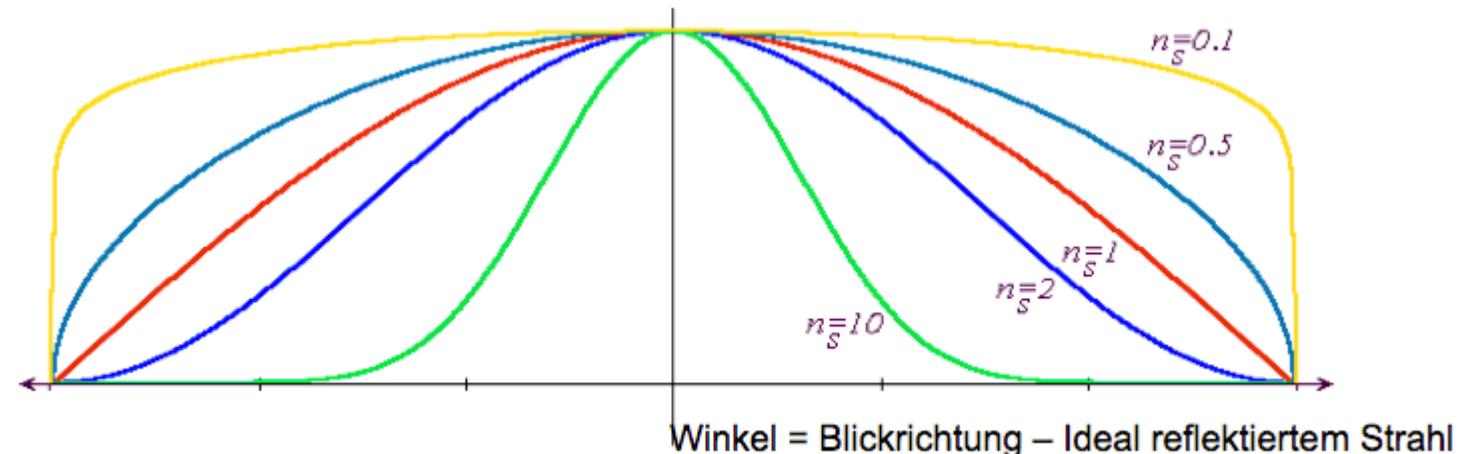
### Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- Spiegelnde Reflexion, Term  $k_s \cdot I_s$ :
  - Physikalisch gesehen besteht die spiegelnde Reflexion aus einem Abbild der Lichtquelle, welches über einem Teil der Oberfläche „verschmiert“ ist: Highlight
  - Highlight kann vom Betrachter nur gesehen werden, wenn seine Betrachtungsrichtung ( $V$ ) nahe der Reflexionsrichtung ( $R$ ) liegt.
- Simulation durch:
  - $I_s = I_i \cdot \cos^n(\Omega) = I_i \cdot (R \circ V)^n$
  - $I_i$  Intensität des einfallenden Lichts der Lichtquelle  $i$
  - $\Omega$ : Winkel zwischen  $V$  und  $R$
  - $n$ : simuliert Perfektionsgrad der Oberfläche
    - $n \rightarrow \infty$  heißt perfekter Spiegel
    - reflektiertes Licht nur in Richtung  $R$
    - keine Streuung

## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

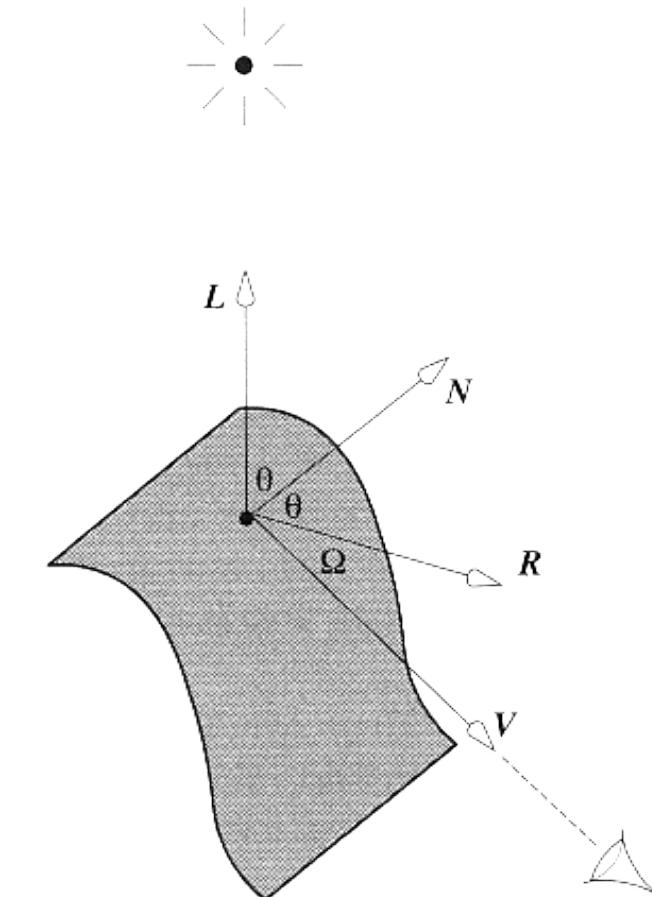
- Spiegelnde Reflexion:
  - Term wird kleiner mit zunehmenden Winkel zwischen Blickwinkel und (ideal) reflektiertem Strahl



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

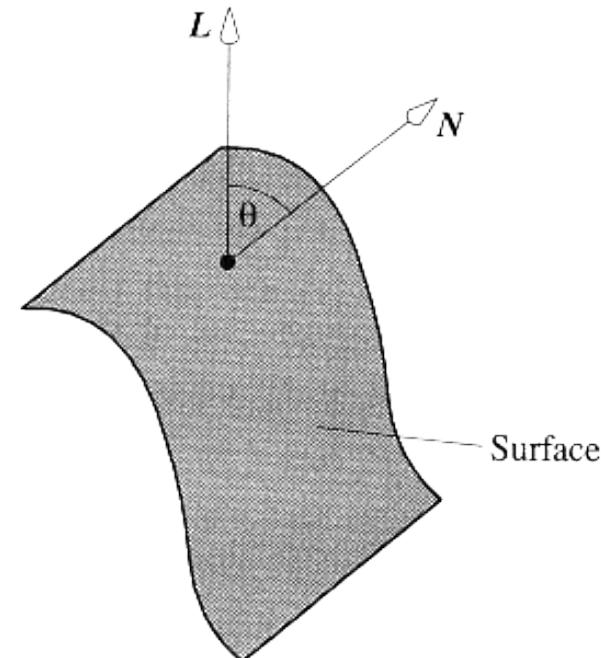
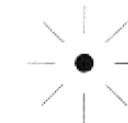
- Spiegelnde Reflexion Bemerkung
  - Für verschiedene  $L$  entsteht (bis auf seine Ausrichtung um  $R$ ) immer der gleiche Reflexions-Intensitätskonus
  - Dies entspricht nicht der realen Abhängigkeit von Spiegelungen von der Ausrichtung des Lichtvektors zum Normalenvektor
  - Gravierender Mangel des Modells



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

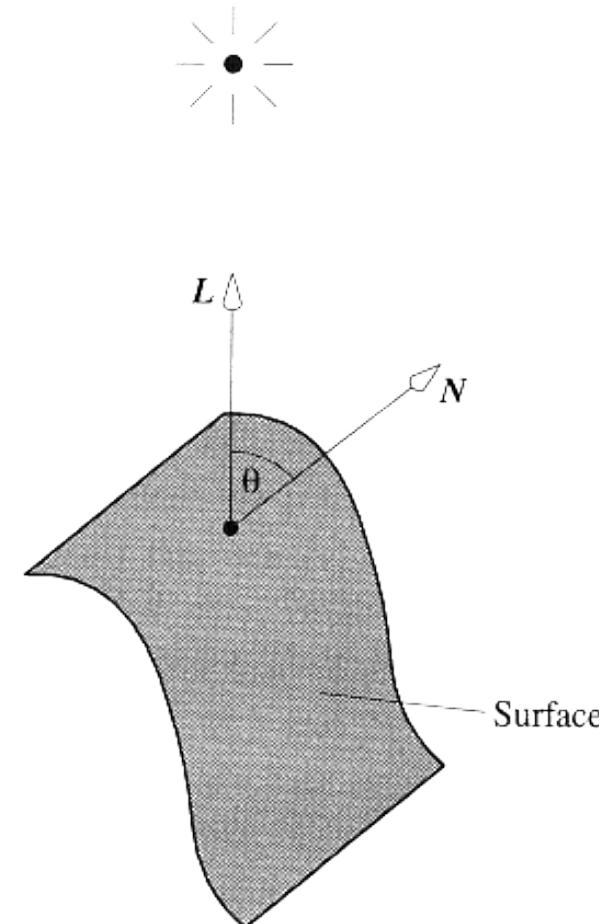
- Diffuse Reflexion, Term  $k_d \cdot I_d$ :
  - $I_d = I_i \cdot \cos\Theta = I_i \cdot (L \circ N)$
  - $I_i$ : Intensität des einfallenden Lichts der Lichtquelle  $i$
  - $\Theta$ : Winkel zwischen Punktnormale  $N$  und Lichtvektor  $L$



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

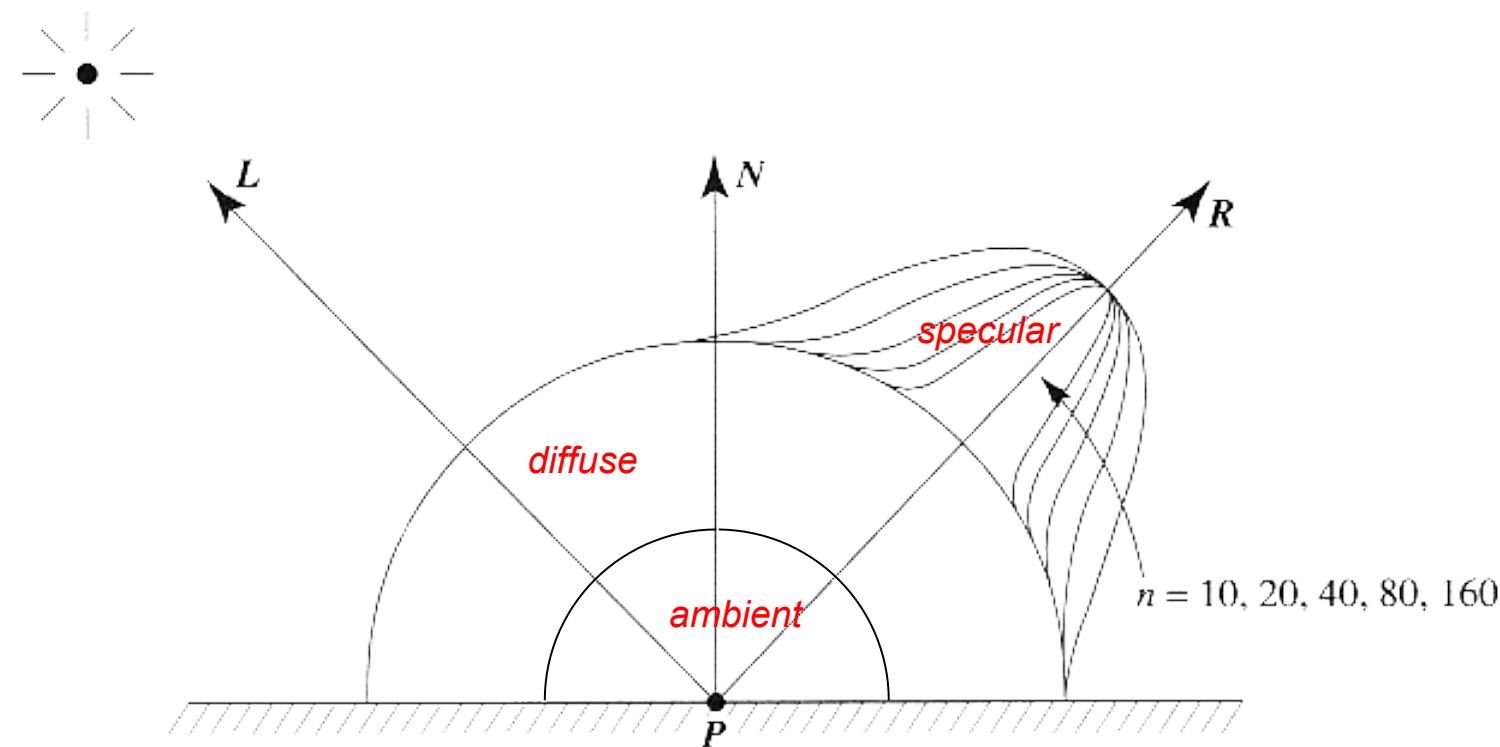
- Diffuse Reflexion
  - Die diffuse Komponente des Phong-Modells modelliert das Kosinusgesetz von Lambert
  - Bei ideal diffusen (matten) Oberflächen ist die Intensität des (in alle Richtungen gleich) reflektierten Lichtes eine Funktion des Kosinus zwischen Oberflächennormale und Lichtvektor



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

$$\begin{aligned} I &= k_s \cdot I_s + k_d \cdot I_d + k_a \cdot I_a \\ &= k_s \cdot I_i \cdot (R \circ V)^n + k_d \cdot I_i \cdot (L \circ N) + k_a \cdot I_a \end{aligned}$$



## 9.3 Lokale Beleuchtungsmodelle

Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- Beispiel

$k_a$  konstant

zunehmendes  $k_s$

zunehmendes  $n$



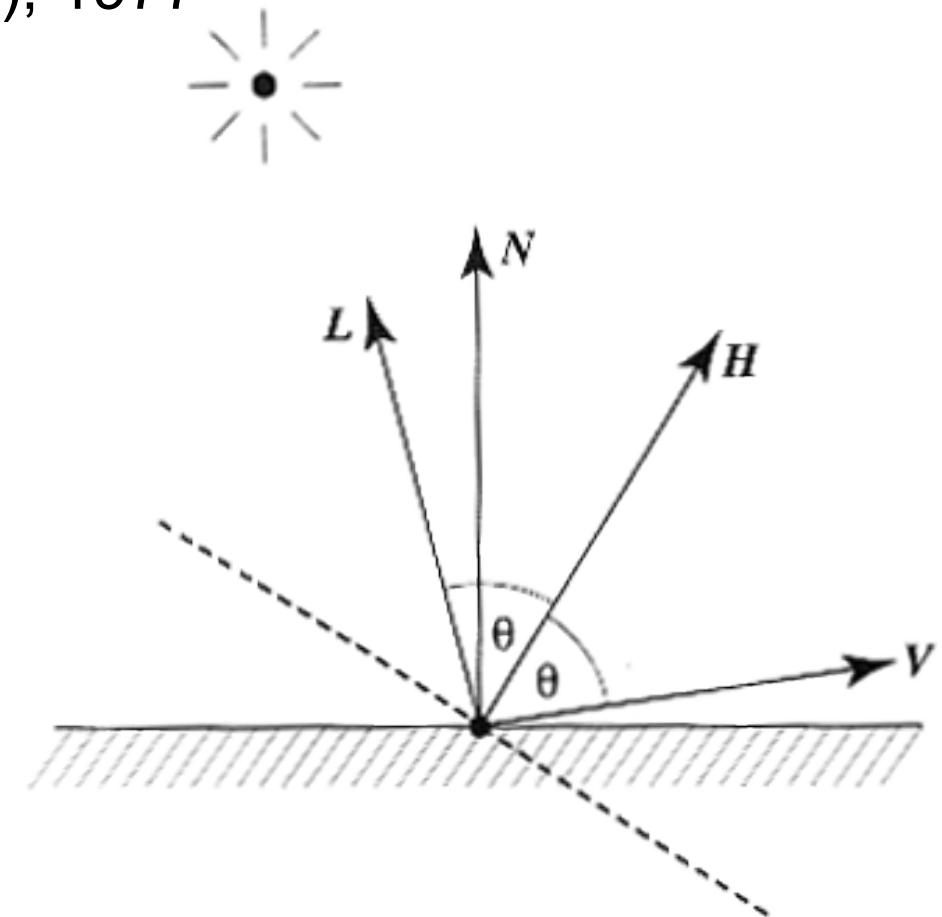
## 9.3 Lokale Beleuchtungsmodelle

Blinn-Phong Beleuchtungsmodell (James F. Blinn), 1977

- Es kann statt dem Reflexionsvektor ein konstanter Vektor eingesetzt werden

$$H = \frac{L + V}{\|L + V\|}$$

- Man betrachtet statt  $R \circ V$  jetzt  $H \circ V$ 
  - unterscheidet sich von  $R \circ V$
  - verhält sich auf die gleiche Weise
- Damit ergibt sich:  
$$I = k_s \cdot I_i \cdot (H \circ V)^n + k_d \cdot I_i \cdot (L \circ N) + k_a \cdot I_a$$
- Gute Annäherung, wenn Betrachter und Lichtquelle weit entfernt sind
- Schneller



## 9.3 Lokale Beleuchtungsmodelle

Blinn-Phong Beleuchtungsmodell (James F. Blinn), 1977

- Für farbige Objekte (Lichtquellen) wird das Modell getrennt auf die Farbkomponenten  $r, g, b$  angewendet

$$I_r = k_{s,r} \cdot I_i \cdot (N \circ H)^n + k_{d,r} \cdot I_i \cdot (L \circ N) + k_{a,r} \cdot I_a$$

$$I_g = k_{s,g} \cdot I_i \cdot (N \circ H)^n + k_{d,g} \cdot I_i \cdot (L \circ N) + k_{a,g} \cdot I_a$$

$$I_b = k_{s,b} \cdot I_i \cdot (N \circ H)^n + k_{d,b} \cdot I_i \cdot (L \circ N) + k_{a,b} \cdot I_a$$

- $k_{d,\{r,g,b\}}$ : Farbe des Objektes
- $k_{s,\{r,g,b\}}$ : Farbe der Lichtquelle (weiß:  $k_{s,r} = k_{s,g} = k_{s,b}$ )
- $k_{a,\{r,g,b\}}$ : Farbe des Umgebungslichtes

## 9.3 Lokale Beleuchtungsmodelle

### Phong-Beleuchtungsmodell (Bui-Tuong Phong), 1975

- Gravierende Mängel des Modells
  - Gegenseitige Reflektionen und Spiegelungen der Flächen werden durch den konstanten ambienten Term nur unzureichend beschrieben
  - Objektoberflächen wirken „plastikhaft“, z. B. lässt sich kein blankes Metall modellieren
- Alternative
  - Physikalisch basierte lokale Beleuchtungsmodelle
  - Simulation der BRDF (Reflexionsfunktion)
  - Mapping-Verfahren
  - ...

## 9.3 Lokale Beleuchtungsmodelle

### BRDF

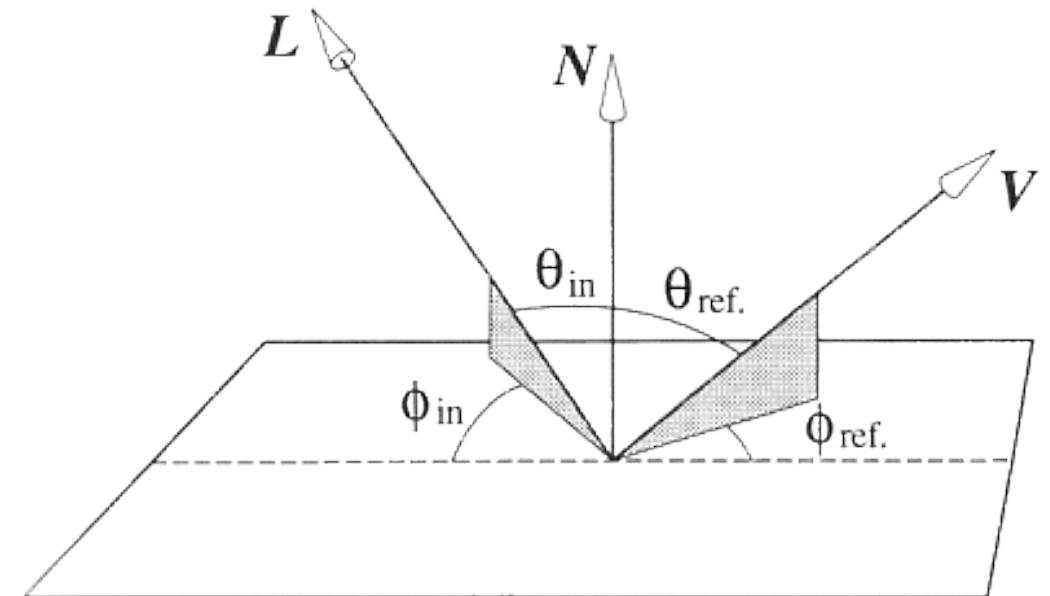
- Bi-directional Reflection Distribution Function: BRDF
  - Allgemein wird das von einem Punkt einer Oberfläche reflektierte Licht durch eine BRDF beschrieben
  - Dies betont insbesondere die Abhangigkeit des in einer beliebigen Richtung reflektierten Lichts von der Richtung des einfallenden Lichts

## 9.3 Lokale Beleuchtungsmodelle

### BRDF

- Sind die Richtungen von  $L$  und  $V$  gegeben, so wird der Zusammenhang zwischen den Intensitäten durch folgende Formel beschrieben

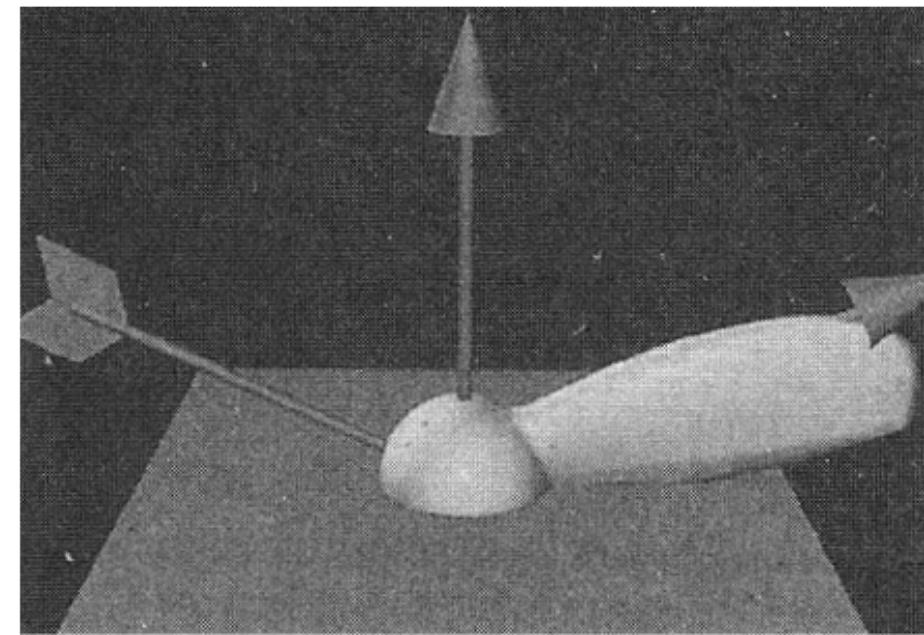
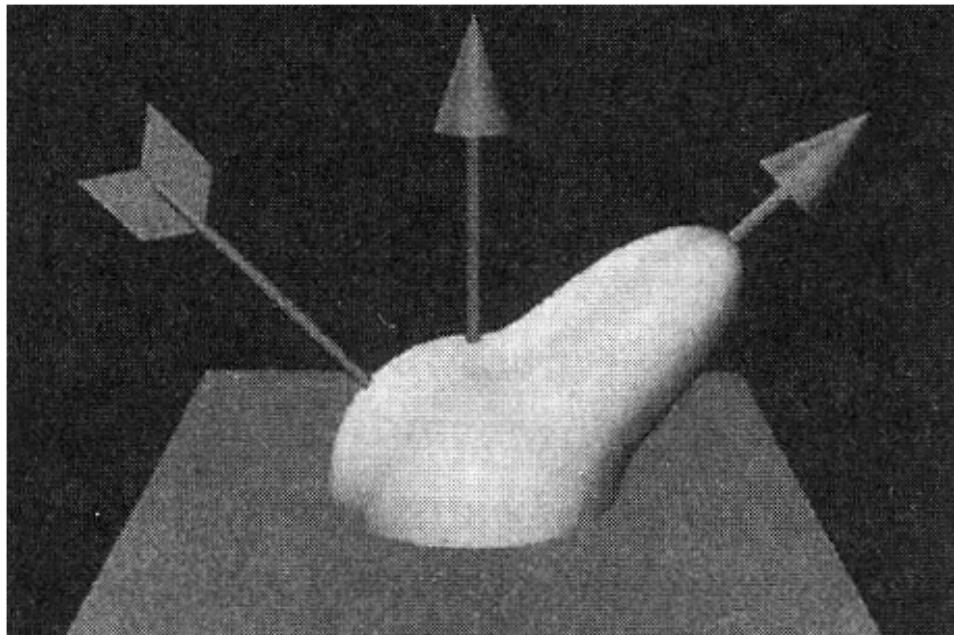
$$\begin{aligned} BRDF &= f(\Theta_{in}, \phi_{in}, \Theta_{ref}, \phi_{ref}) \\ &= f(L, V) \end{aligned}$$



## 9.3 Lokale Beleuchtungsmodelle

### BRDF

- Darstellung von (nach Blinn (1977)) erzeugten BRDFs für zwei verschiedene Richtungen des einfallenden Lichtes

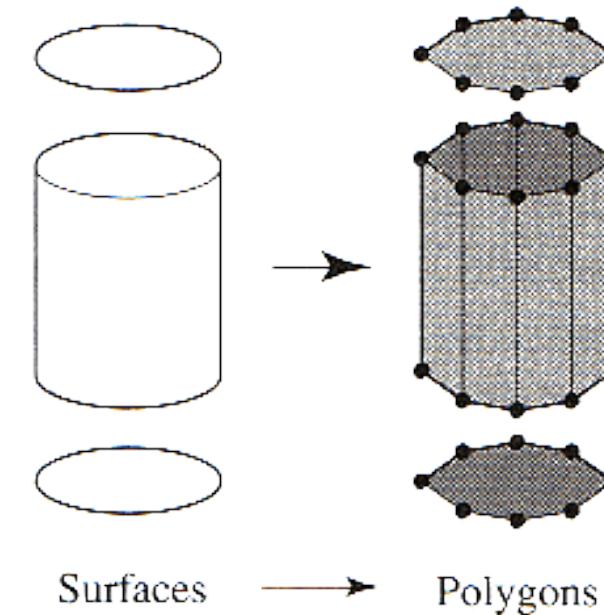


## 9.3 Lokale Beleuchtungsmodelle

- Nachteile „rein“ lokaler Beleuchtungsmodelle
  - Spiegeln Idealfall eines einzelnen von einer einzigen Punktlichtquelle beleuchteten Objektes in der Szene wider
  - Betrachten ausschließlich direkte Beleuchtung (bis auf Hilfskonstruktionen)
  - Interaktion mit anderen Objekten nicht modelliert (d. h. keine indirekte Beleuchtung, kein Schattenwurf!)
- Besser: globale Beleuchtungsverfahren

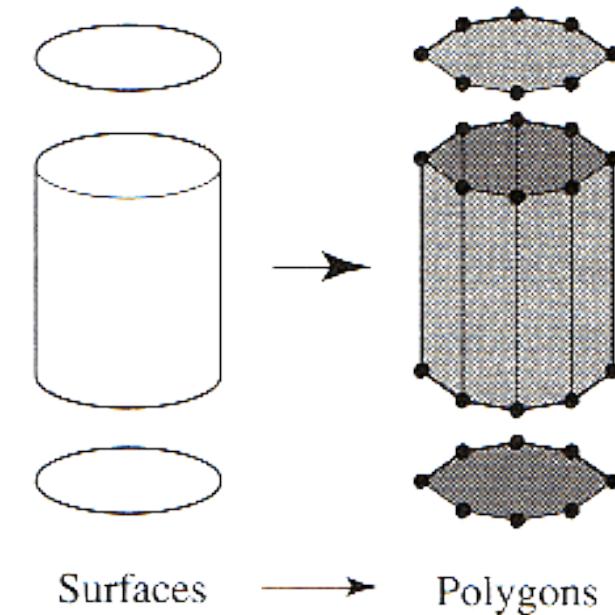
## 9.4 Schattierung

- Gegeben:
  - Polygonale Repräsentation eines Objekts
  - Punkt  $P$  auf der Oberfläche des Objekts
- Gesucht:
  - Farbe der Oberfläche am Punkt  $P$
- Probleme:
  - Glattheit an den Kanten zwischen zwei Polygone
  - Glattheit an den gemeinsamen Eckpunkten mehrerer Polygone



## 9.4 Schattierung

- Genauer: Bestimmung der Vektoren, welche für das Beleuchtungsmodell benötigt werden  
→ Schattierungsmodell (Shading Model)



## 9.4 Schattierung

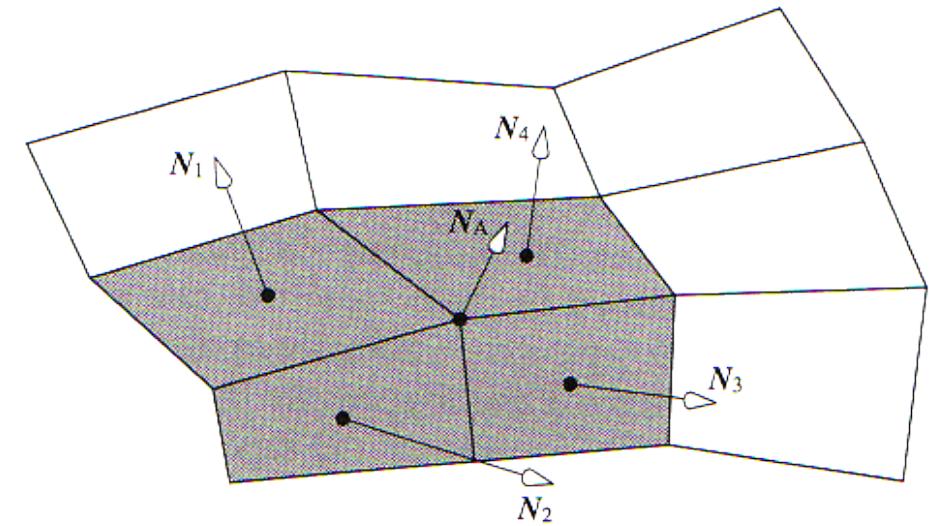
### Varianten

- Konstante Farbe pro Polygon
  - Flat/uniform Shading
- Interpolation
  - Der Farben: Gouraud-Shading
  - Der Normalen: Phong-Shading

## 9.4 Schattierung

### Flat/Uniform Shading

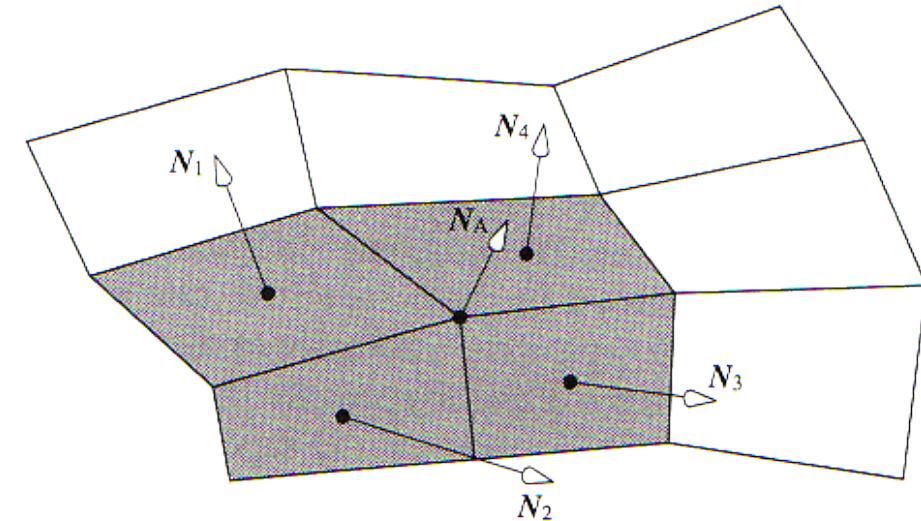
- Grundlage der Berechnung ist die Polygon- oder Oberflächennormale im Objektraum ( $N_1, N_2, N_3, N_4$ )
- Pro Polygon / Facette wird das verwendete Beleuchtungsmodell genau einmal in einem ausgewählten Oberflächenpunkt ausgewertet
- Die dort ermittelte Lichtintensität wird auch allen weiteren Punkten der Polygonoberfläche zugewiesen



## 9.4 Schattierung

### Flat/Uniform Shading

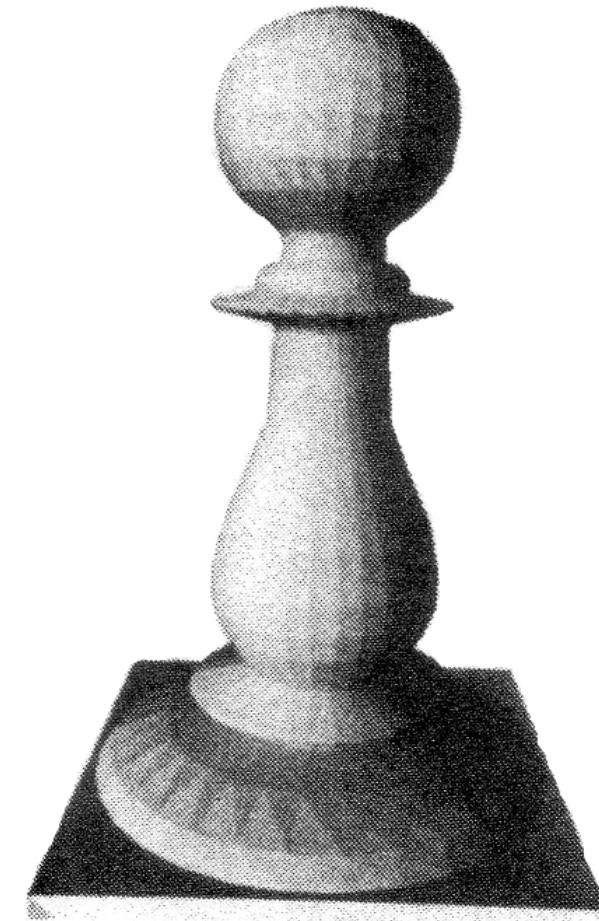
- Als ausgewählte Punkte werden die Polygonschwerpunkte oder Polygoneckpunkte gewählt
- Voraussetzung: Planarität des Polygons
  - Für Dreiecke immer erfüllt



## 9.4 Schattierung

### Flat/Uniform Shading

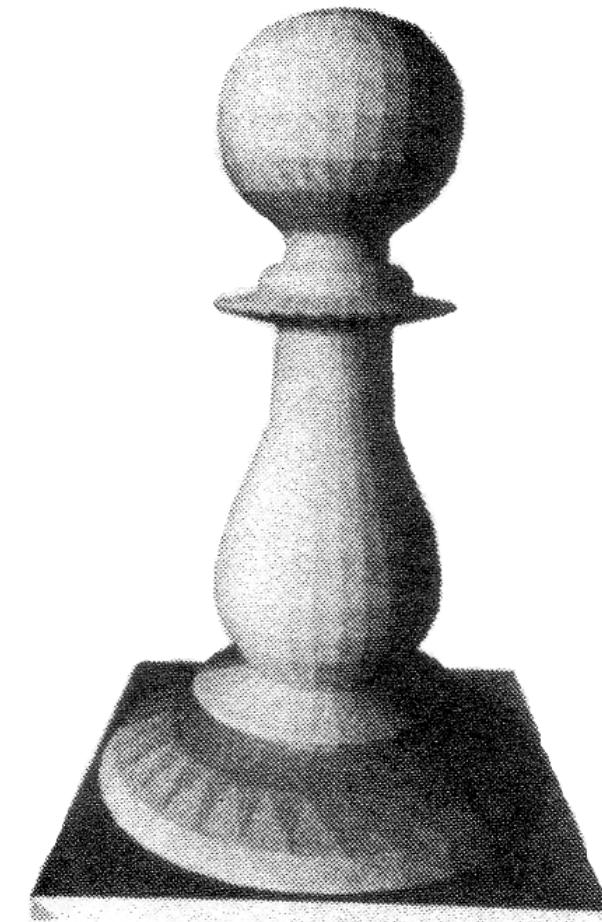
- Vorteile
  - Einfaches, kostengünstiges Verfahren
  - Keine Interpolation
- Nachteile
  - Unstetiger Intensitätsverlauf über die Polygonkanten
  - Kanten in Polygonnetzen bleiben bei der Darstellung sichtbar
  - Objekte werden facettiert dargestellt



## 9.4 Schattierung

### Flat/Uniform Shading

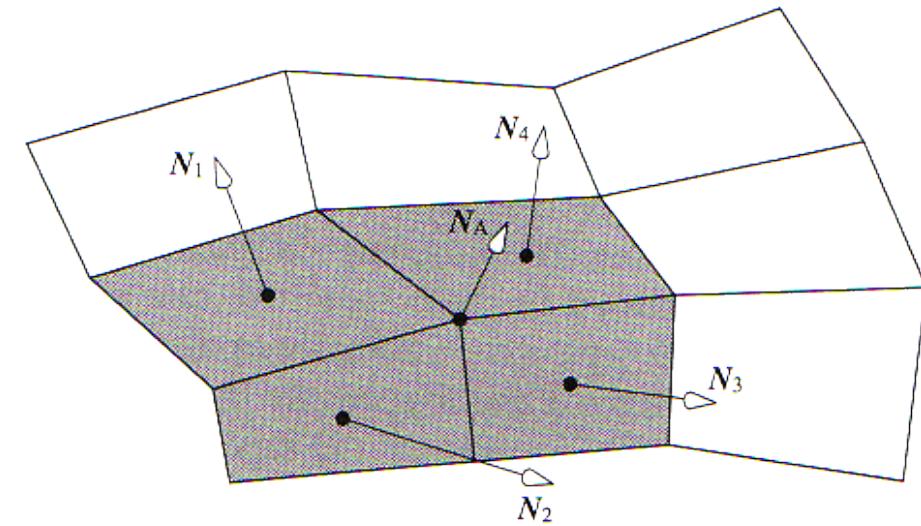
- Mögliche Anwendung
  - Visualisierung der polygonalen Auflösung für Qualitätsdarstellung



## 9.4 Schattierung

### Gouraud- und Phong-Shading

- Grundlage der Berechnung sind die Eckpunktnormalen in den gemeinsamen Polygoneckpunkten ( $N_A$ )
- Eine Eckpunktnormale wird als (gewichtetes) Mittel der Polygonnormalen aller angrenzenden Polygone mit dem entsprechenden gemeinsamen Eckpunkt berechnet
- Normalisierung notwendig



## 9.4 Schattierung

### Gouraud- und Phong-Shading

- Die Farbe an einem Punkt auf der Oberfläche ergibt sich dann durch Interpolation
  - Komplex für beliebige Polygone
  - Einfach für Dreiecke
- Ziel
  - Glätten (virtueller) Kanten zwischen einzelnen Polygonfacetten  
(Das Polygonnetz stellt die Approximation einer gekrümmten Oberfläche dar)

## 9.4 Schattierung

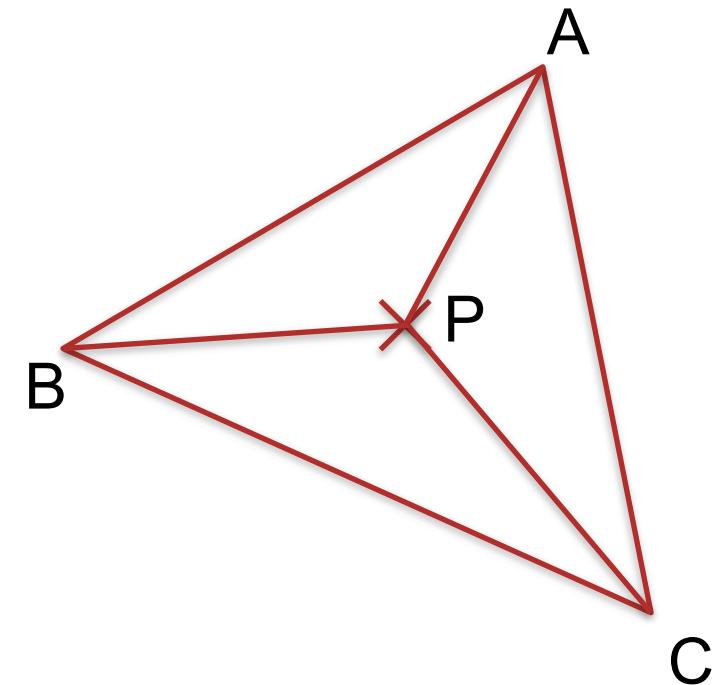
### Gouraud-Shading

- Die Auswertung des Beleuchtungsmodells erfolgt ausschließlich in den Polygoneckpunkten unter Ausnutzung der Eckpunktnormalen
- Intensitätswerte am Punkt  $P$  werden interpoliert
  - Verwendung von baryzentrischen Koordinaten
- Gegeben: Baryzentrische Koordinaten  $(\alpha, \beta, \gamma)$  eines Punktes  $P$  bezüglich eines Dreiecks  $(A, B, C)$
- Berechnung der Intensität
$$I_P = \alpha \cdot I_A + \beta \cdot I_B + \gamma \cdot I_C$$
- $I_P$ : Intensität Punkt  $P$
- $I_A$ : Intensität Punkt  $A$
- $I_B$ : Intensität Punkt  $B$
- $I_C$ : Intensität Punkt  $C$

## 9.4 Schattierung

### Baryzentrische Koordinaten

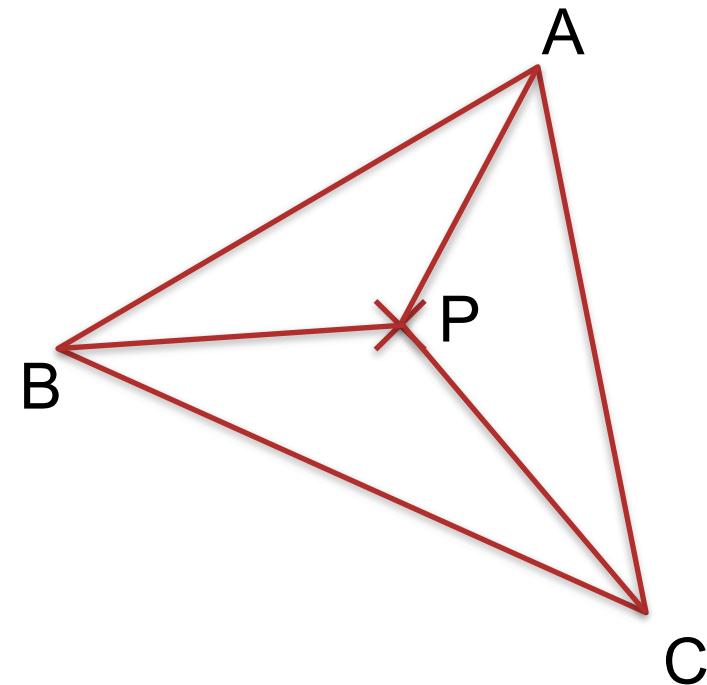
- Gegeben: Dreieck ( $A, B, C$ )
- Gesucht: Koordinaten von  $P$  bezüglich des Dreiecks ( $A, B, C$ )
- Ansatz:  
$$P = \alpha \cdot A + \beta \cdot B + \gamma \cdot C$$
- Nebenbedingung:  
$$\alpha + \beta + \gamma = 1$$
  
(Normalisierung)



## 9.4 Schattierung

### Baryzentrische Koordinaten

- $P = \alpha \cdot A + \beta \cdot B + \gamma \cdot C$
- Folgerungen:
  - $A = (1, 0, 0)$
  - $B = (0, 1, 0)$
  - $C = (0, 0, 1)$
- $\alpha = 0, \beta + \gamma = 1$ : Kante  $B - C$
- $\beta = 0, \alpha + \gamma = 1$ : Kante  $A - C$
- $\gamma = 0, \alpha + \beta = 1$ : Kante  $A - B$
- $0 \leq \alpha, \beta, \gamma \leq 1$ :  $P$  liegt innerhalb
- Sonst:  $P$  liegt außerhalb



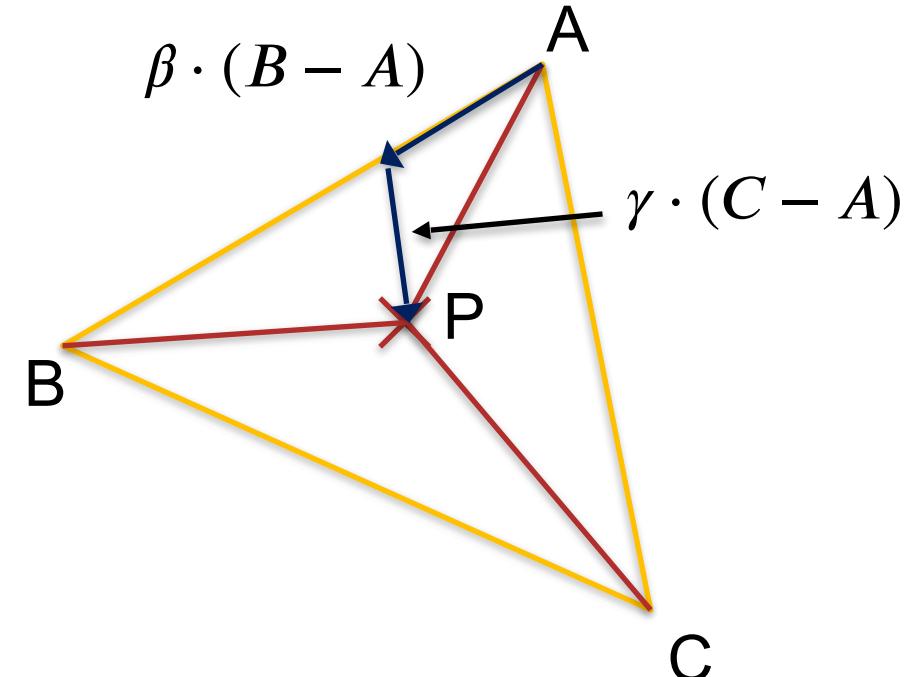
## 9.4 Schattierung

### Baryzentrische Koordinaten

- $P = \alpha \cdot A + \beta \cdot B + \gamma \cdot C$
- $$\begin{aligned} P &= \alpha \cdot A + \beta \cdot B + \gamma \cdot C \\ &= (1 - \beta - \gamma) \cdot A + \beta \cdot B + \gamma \cdot C \end{aligned}$$
- $= A + \beta \cdot (B - A) + \gamma \cdot (C - A)$

- Berechnung von  $(\alpha, \beta, \gamma)$

$$\alpha = \frac{\Delta(P, B, C)}{\Delta(A, B, C)}$$

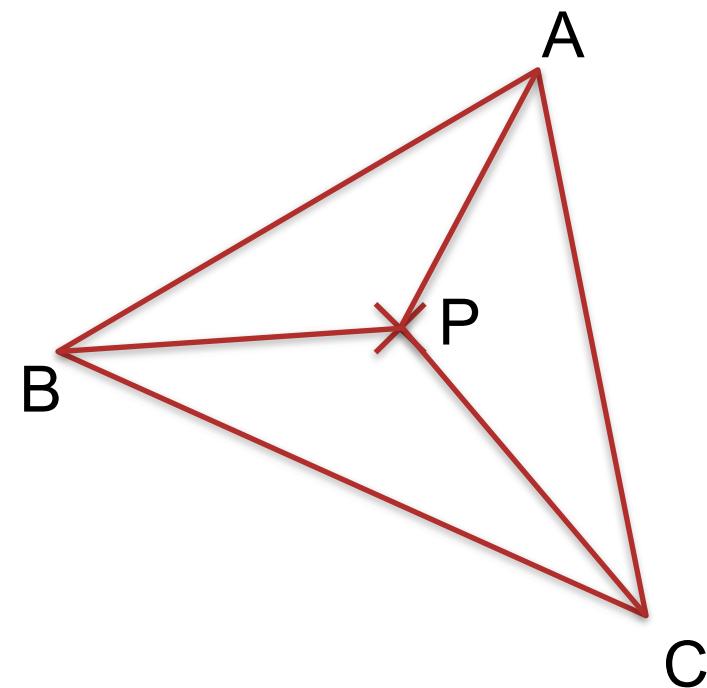


$$\Delta(P, B, C) = ((B - P) \times (C - P)) \circ \left( \frac{(B - A) \times (C - A)}{\|(B - A) \times (C - A)\|} \right)$$

## 9.4 Schattierung

### Baryzentrische Koordinaten

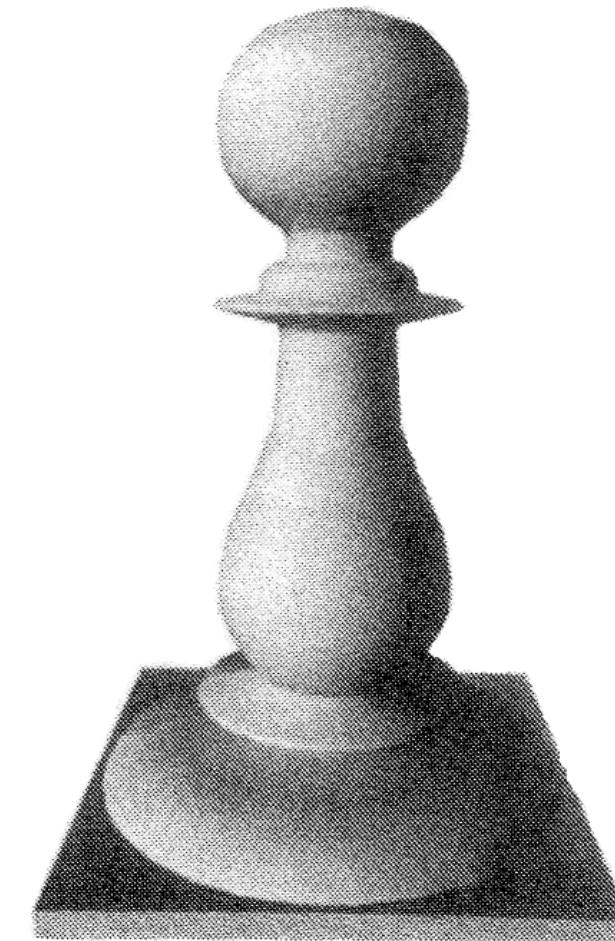
- Effizientere Berechnung und Algorithmen:  
[Christer Ericson, Real Time Collision Detection, Morgan Kaufman – Elsevier, 2005]



## 9.4 Schattierung

### Gouraud-Shading

- Nachteil
  - Highlights werden durch Abtastfehler „verschluckt“ oder erscheinen polygonförmig (statt rund)
- Gängig: Kombination von Gouraud-Shading und Beleuchtungsmodell mit ausschließlich diffuser Reflexionskomponente
- War lange Zeit Standard in der Graphikhardware



## 9.4 Schattierung

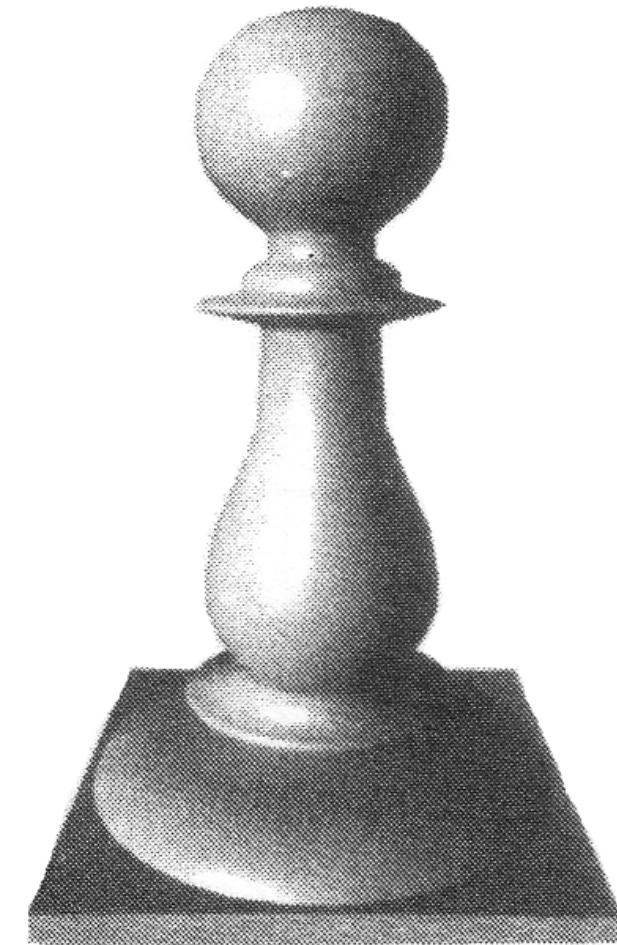
### Phong-Shading

- Die Auswertung des Beleuchtungsmodells erfolgt direkt am Punkt  $P$
- Die Oberflächennormale am Punkt  $P$  wird durch Interpolation der Stützpunktnormalen ermittelt
- Normalisierung nötig
- Auswertung des Beleuchtungsmodells für jedes Pixel nötig

## 9.4 Schattierung

### Phong-Shading

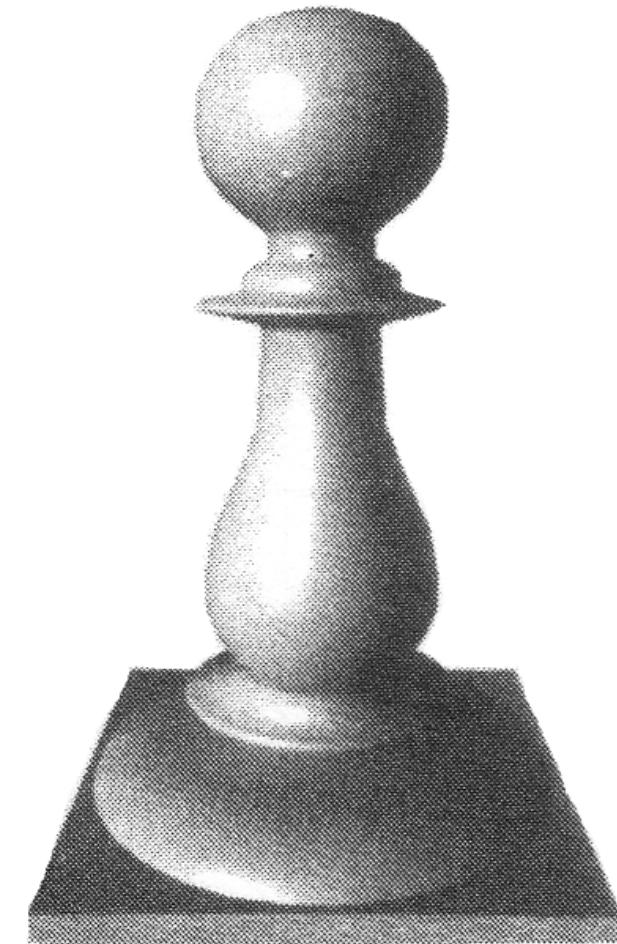
- Vorteile:
  - Intensitätsverlauf über die Polygonkanten ist stetig und glatt
  - Gute Annäherung des Erscheinungsbilds der realen gekrümmten Oberfläche durch interpolierte Normalen
  - Highlights werden adäquat dargestellt



## 9.4 Schattierung

### Phong-Shading

- Nachteile
  - Rechenaufwändiges Verfahren
- Wird (mittlerweile) von der Graphikhardware unterstützt



## 9.4 Schattierung

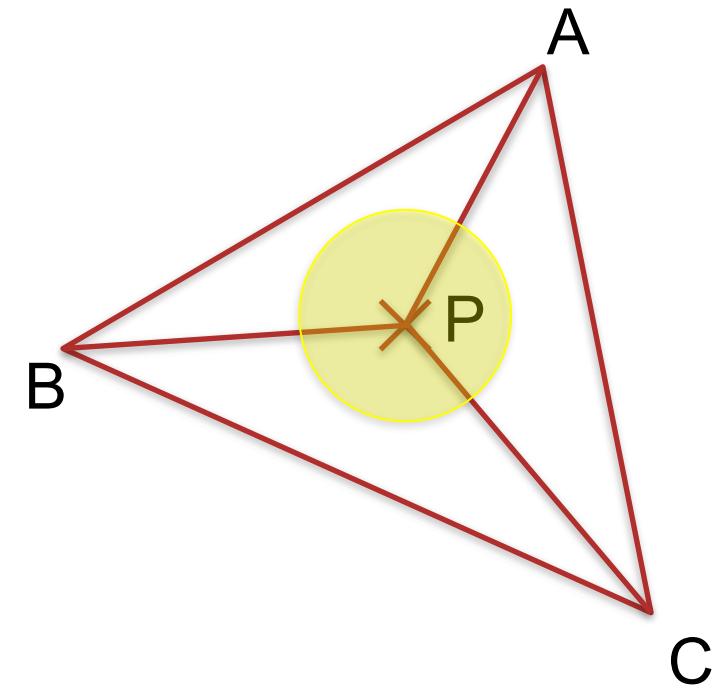
### Vergleich

- Gouraud-Shading
  - Ermittle Intensitäten an den Eckpunkten des Dreiecks  
→ Intensitäten in  $A, B, C$
  - Interpoliere Intensitäten  
→ Intensität in  $P$
- Phong-Shading
  - Interpoliere Normalen der Eckpunkte des Dreiecks  
→ Normale in  $P$
  - Ermittle Intensität in  $P$

## 9.4 Schattierung

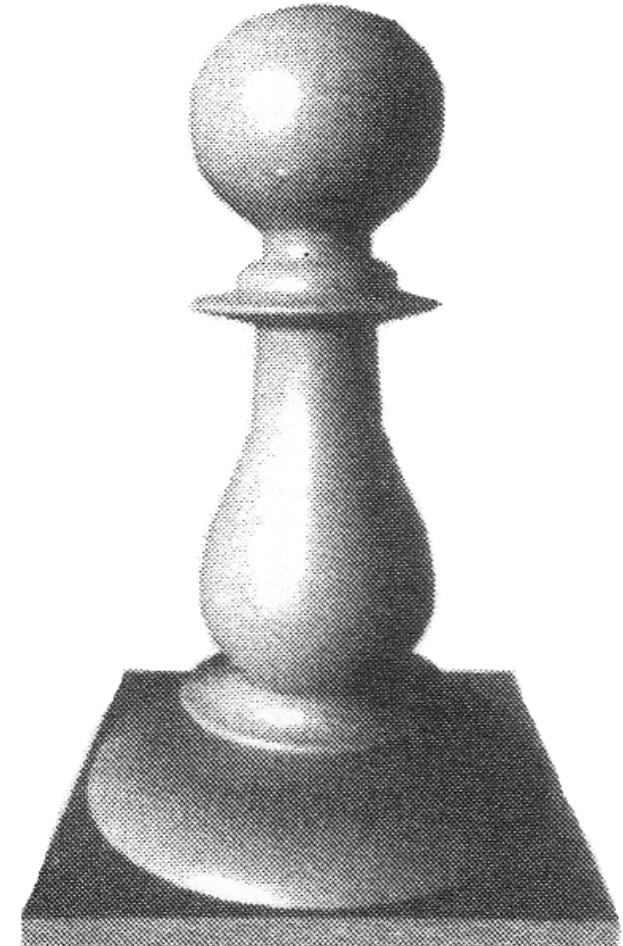
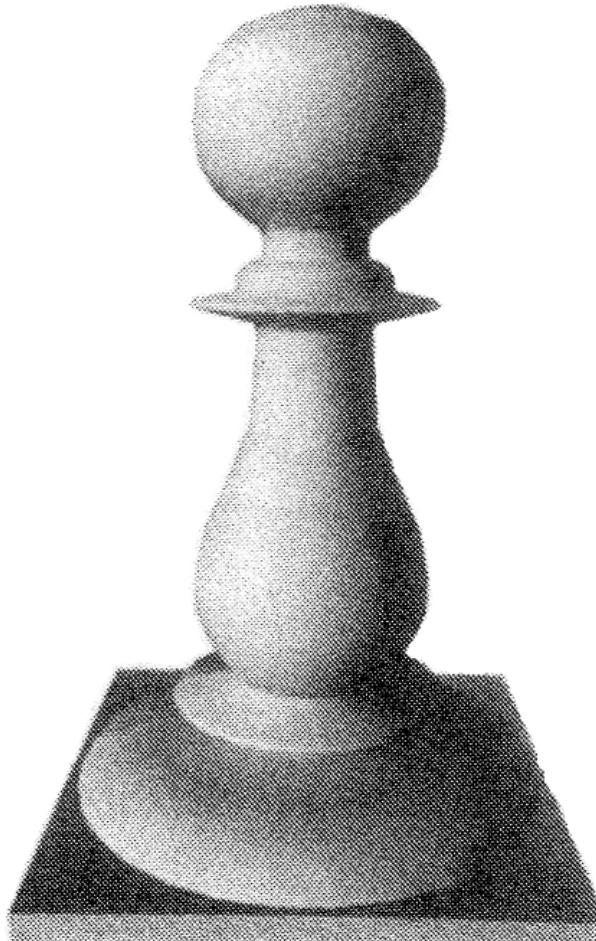
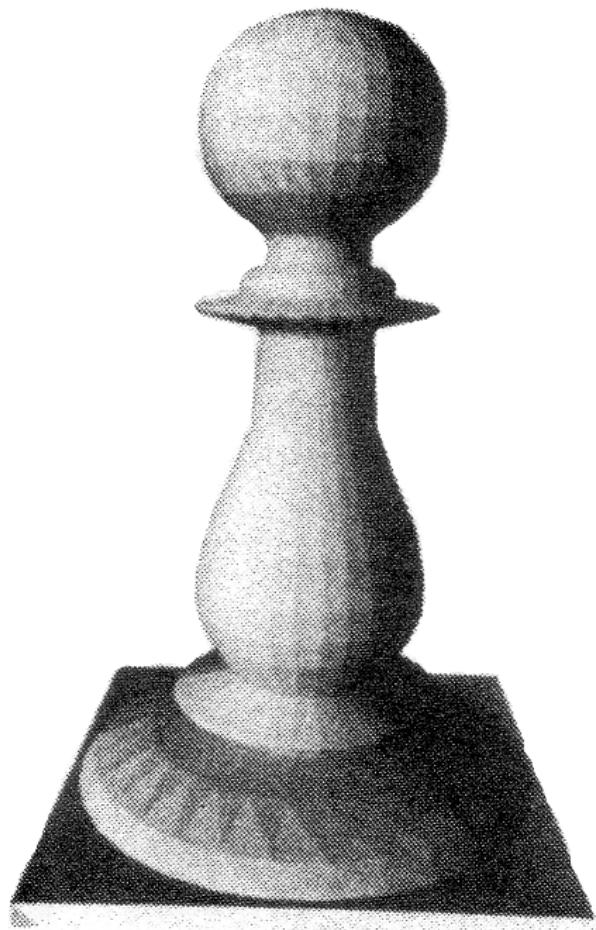
### Vergleich

- Gouraud-Shading
  - Punkt  $P$  ist nur dann beleuchtet, wenn mindestens einer der Eckpunkte beleuchtet ist  
→ Spotlights funktionieren nur über den Eckpunkten, aber nicht im Innern des Dreiecks
- Phong-Shading
  - hat dieses Problem nicht, da die Intensität direkt in  $P$  berechnet wird



## 9.4 Schattierung

Vergleich aller drei Verfahren



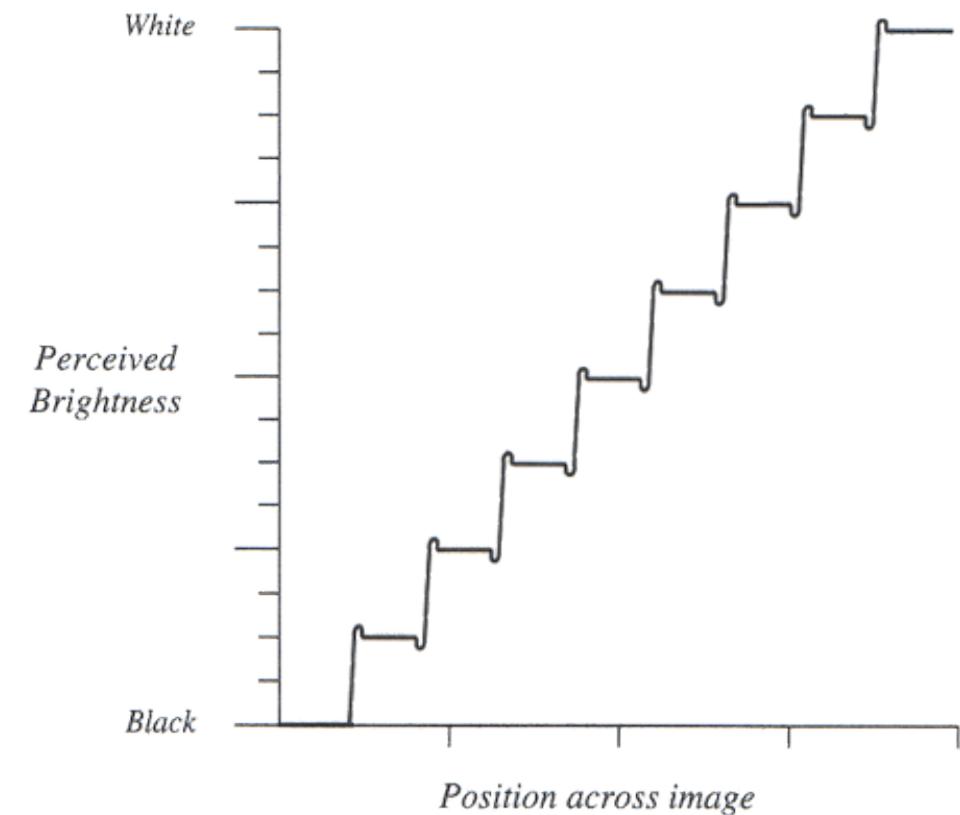
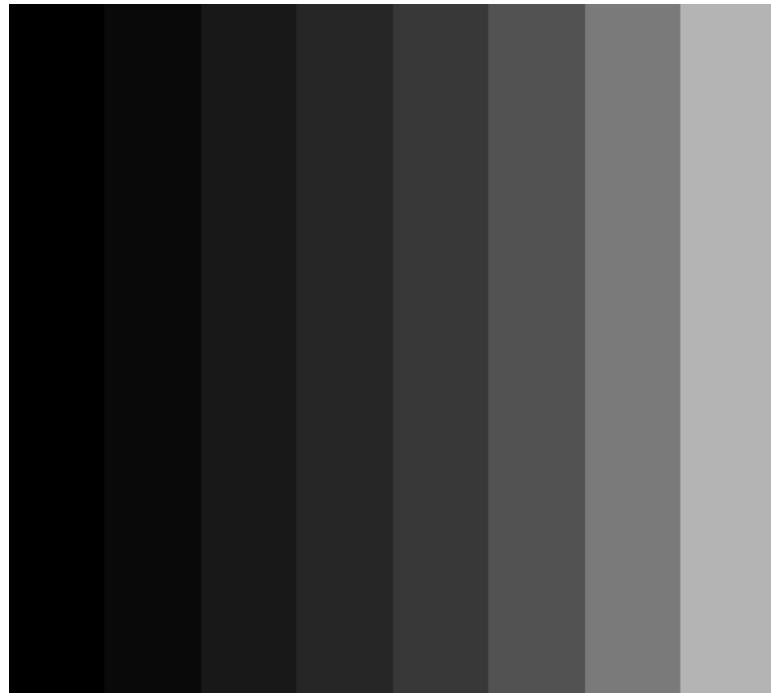
## 9.4 Schattierung

### Mach Band-Effekt

- Interaktion der Lichtrezeptoren im Auge betont Intensitätsänderungen
- Dieser low-level Mechanismus der Kantenbetonung bei Intensitätsübergängen verhilft unserer visuellen Wahrnehmung zu einer automatischen Konturenschärfe

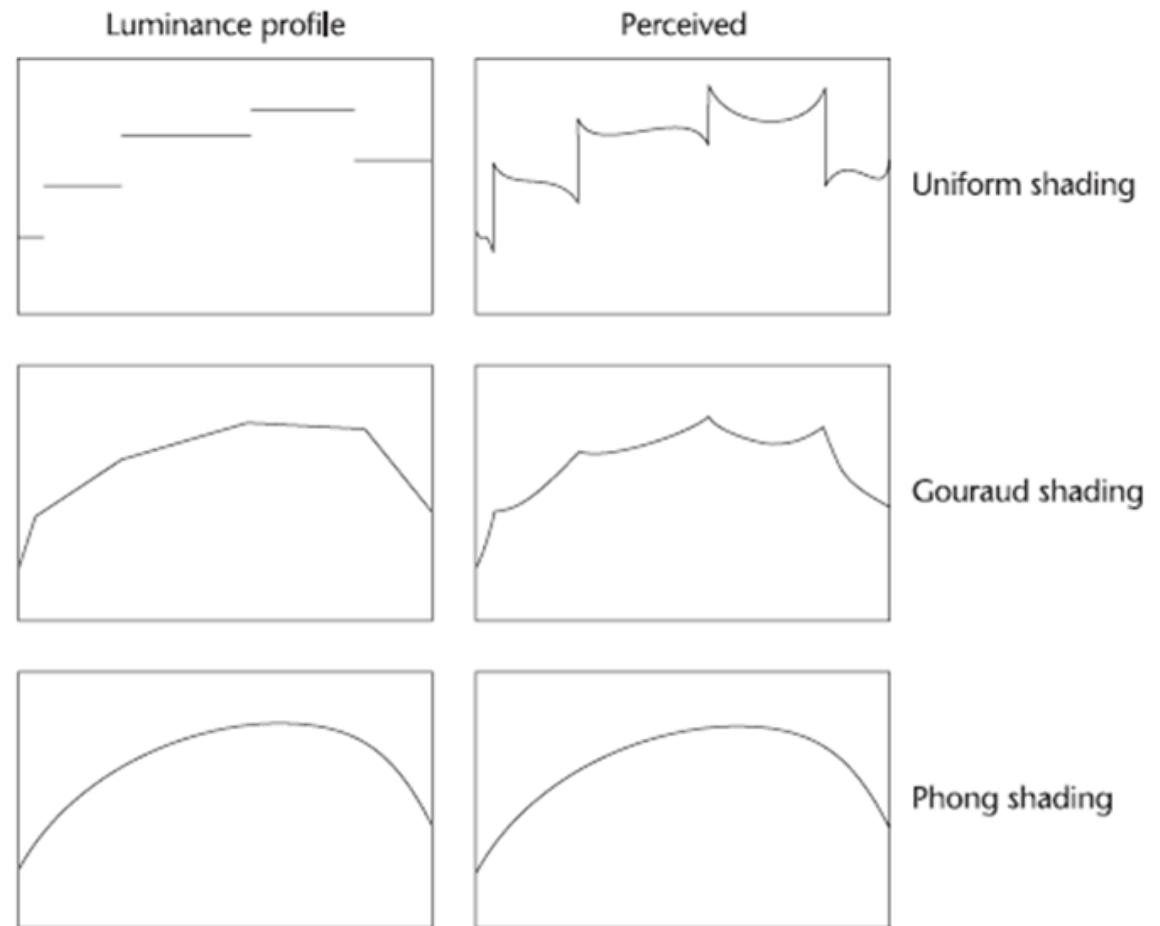
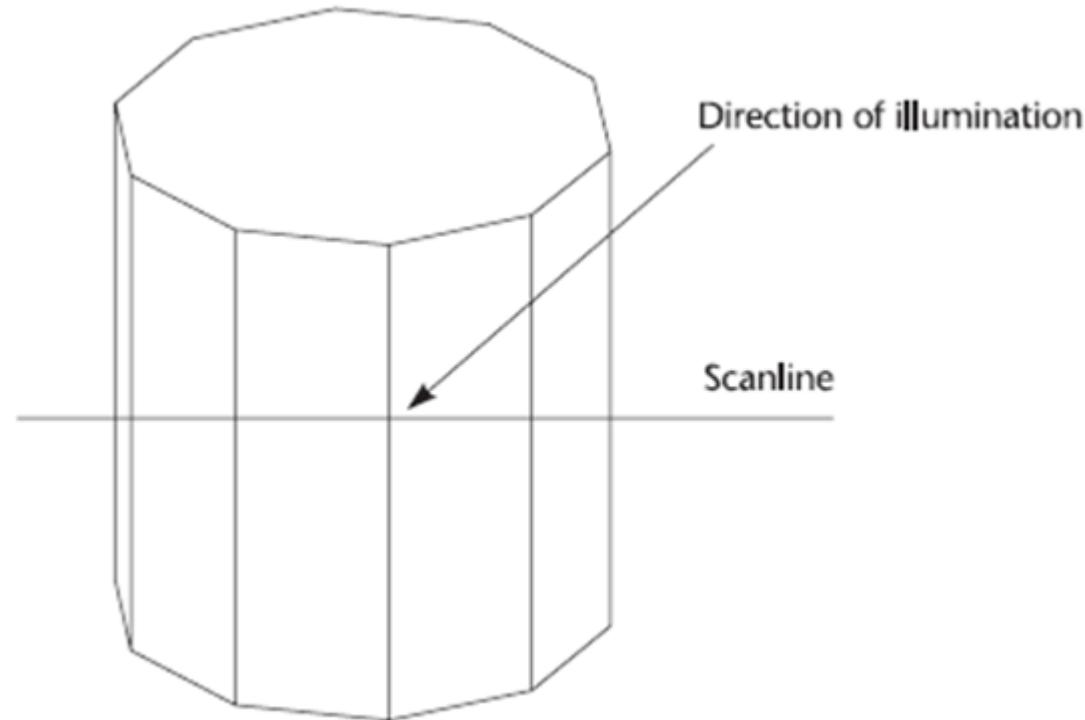
## 9.4 Schattierung

### Mach Band-Effekt



## 9.4 Schattierung

### Mach Band-Effekt & Shading



## 9.4 Schattierung

### Mach Band-Effekt & Shading

- Flat/Uniform Shading
  - unstetige Intensitätswechsel
  - *sehr starke* Mach Band-Effekte
- Phong-Shading
  - glatte Intensitätswechsel
  - *reduzierte* Mach Band-Effekte
- Gouraud-Shading
  - stetige Intensitätswechsel
  - abhängig von Polygonalisierung
    - starke* Mach Band-Effekte

## 9.5 Globale Beleuchtung

- Für photorealistische(re) Darstellung ist globale Beleuchtung notwendig
  - Physikalisch basierte Modellierung der Lichtausbreitung
  - Beachtet auch Wechselwirkung zwischen Objekten

## 9.5 Globale Beleuchtung

- Lokal, nur diffus
- Lokal, diffus und spiegelnd



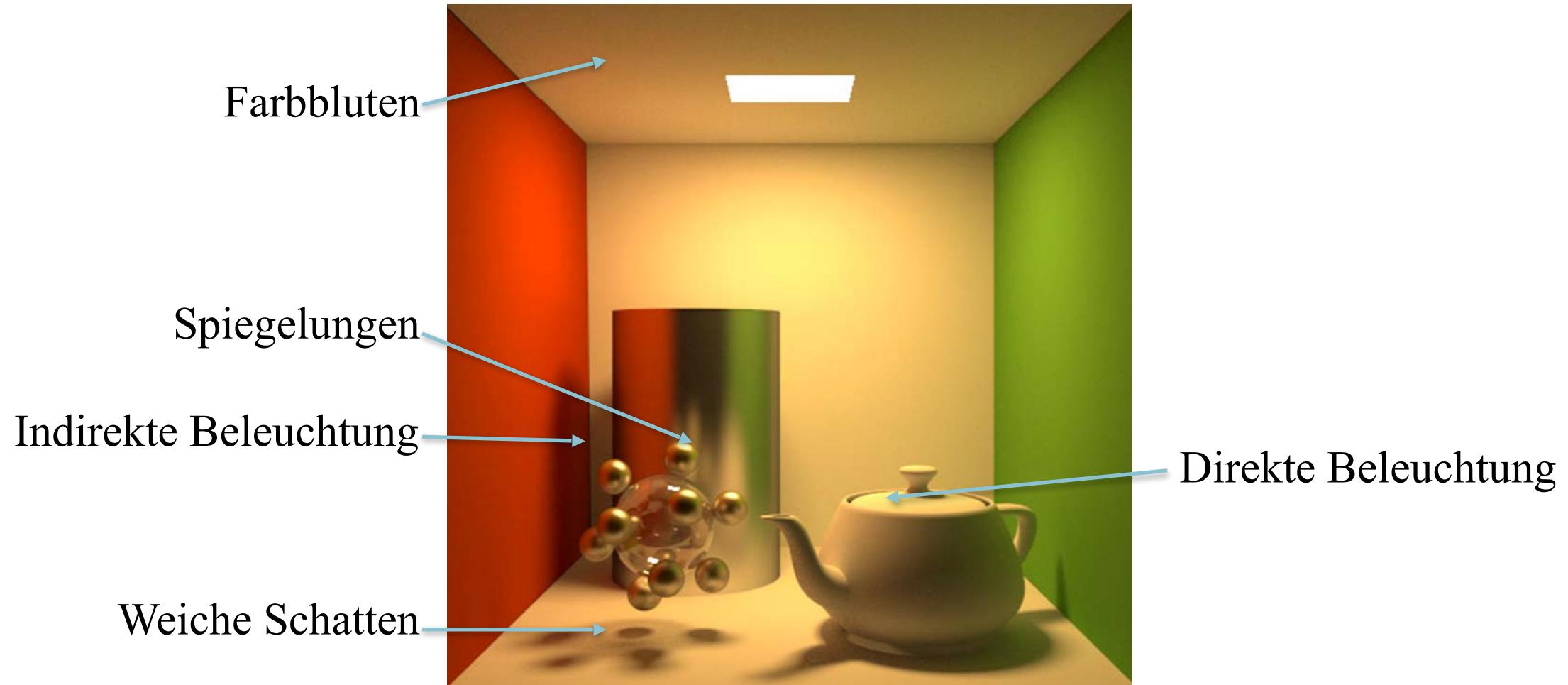
## 9.5 Globale Beleuchtung

- Global
  - Mehrfachreflexion
  - Weiche Schatten
- Lokal, diffus und spiegelnd



## 9.5 Globale Beleuchtung

### Beleuchtungsphänomene



## 9.5 Globale Beleuchtung

Beleuchtungsphänomene

- Kaustiken



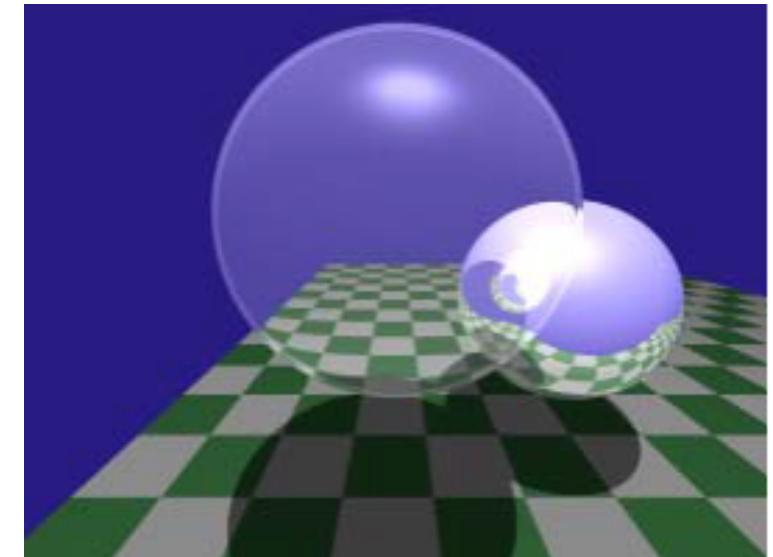
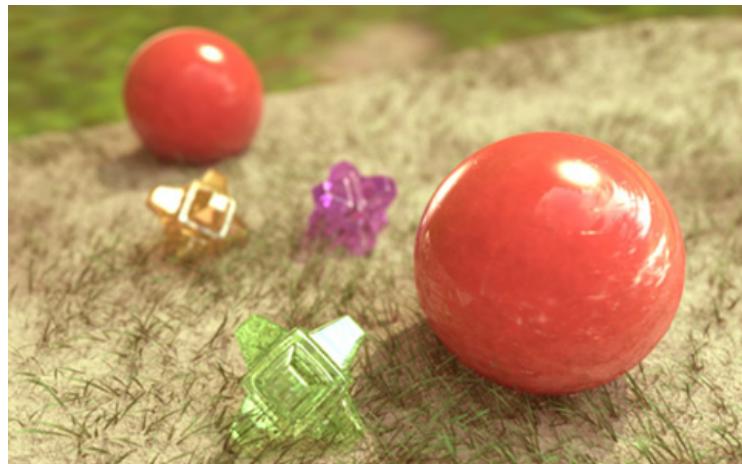
## 9.5 Globale Beleuchtung

- Ray-Tracing
  - Whitted, 1979
  - Modelliert Lichtausbreitung nach Gesetzen der idealen Spiegelung und Brechung
  - Für spiegelnde und transparente Oberflächen geeignet
  - Blickrichtungsabhängig
- Radiosity-Verfahren
  - Goral & Greenberg, 1984
  - Lichtausbreitung basierend auf Energieerhaltung
  - Vorberechnung aller Interaktionen des Lichts mit Objekten der Szene
  - Diffuse Reflexion
  - Unabhängig von der Blickrichtung

## 9.5 Globale Beleuchtung

### Ray-Tracing

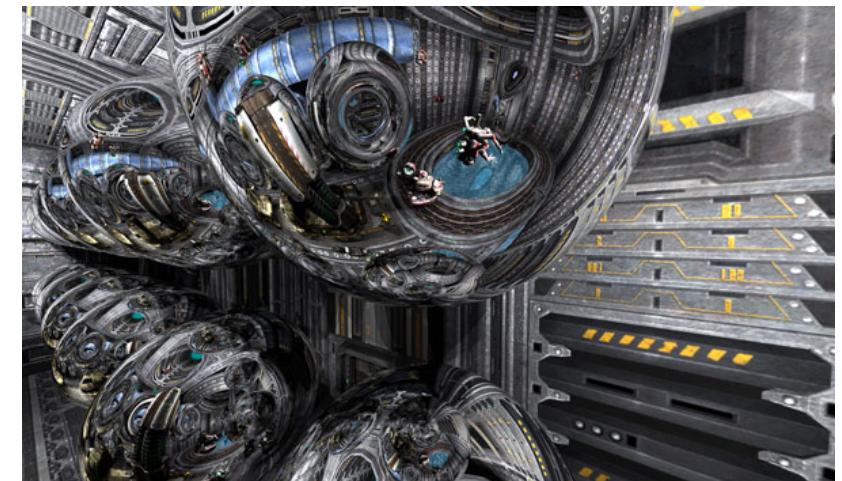
- Besonders geeignet für Szenen mit
  - hohem spiegelnden
  - hohem transparenten Flächenanteil



# 9.5 Globale Beleuchtung

## Ray-Tracing

- Weitere Beispiele



## 9.5 Globale Beleuchtung

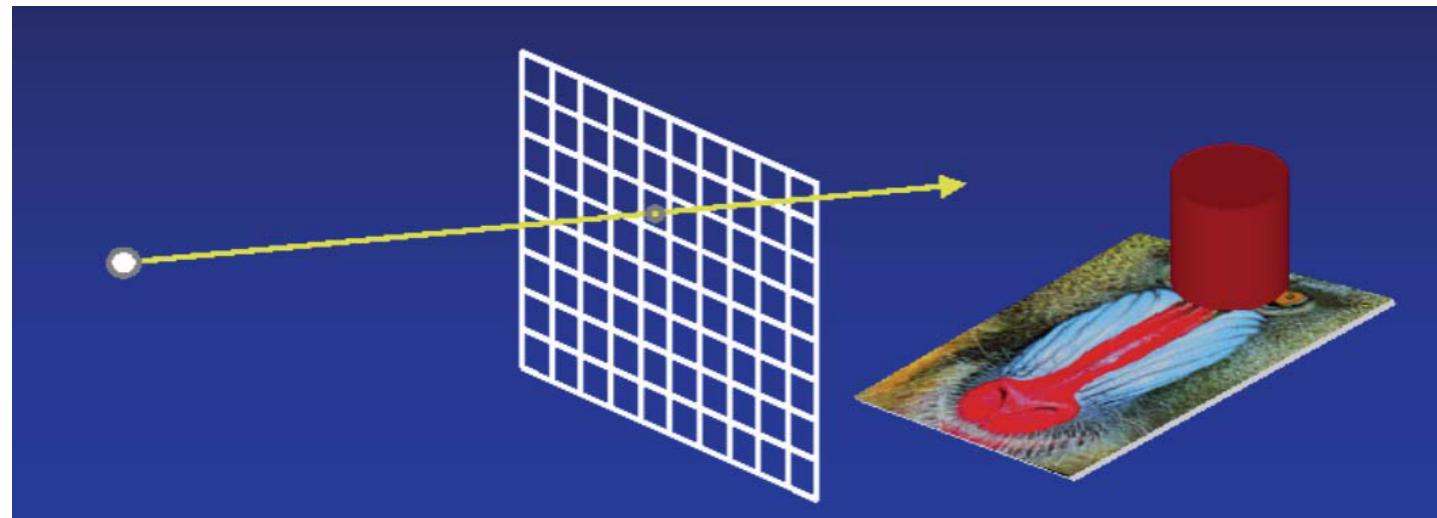
### Ray-Tracing

- Grundprinzip des rekursiven Raytracing
  - Verfolgung der Lichtstrahlen von der Quelle bis zum Auge
  - Aber: Nur wenige Strahlen erreichen das Auge
- Umkehrung des Verfahrens (Reziprozität der Reflexion)
  - Vom Augpunkt wird durch jedes Pixel des Bildschirms ein Strahl in die Szene gesendet
  - Bei Schnittpunkten mit Objekten werden
    - ideal reflektierte
    - ideal gebrocheneStrahlen weiterverfolgt
  - Rekursive Auswertung des lokalen Beleuchtungsmodells (Schattierung)

## 9.5 Globale Beleuchtung

### Ray-Tracing

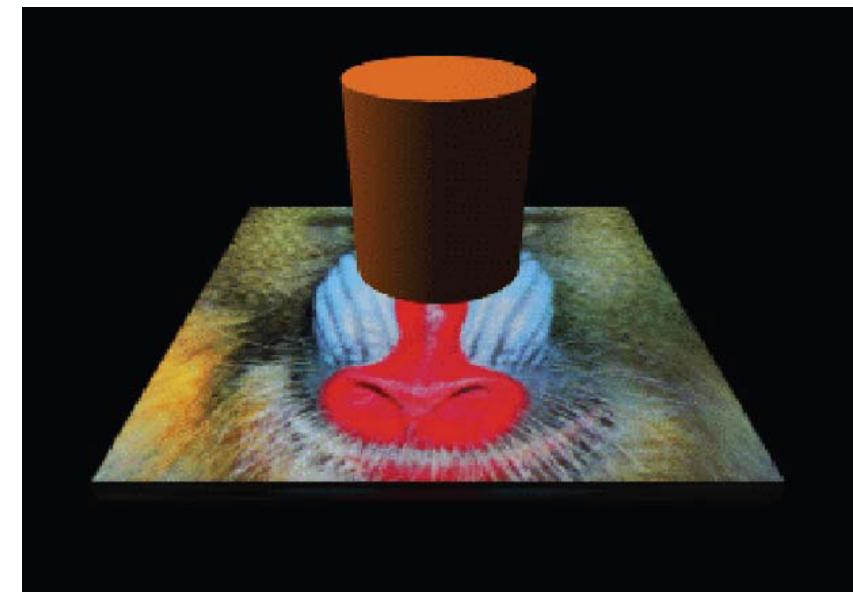
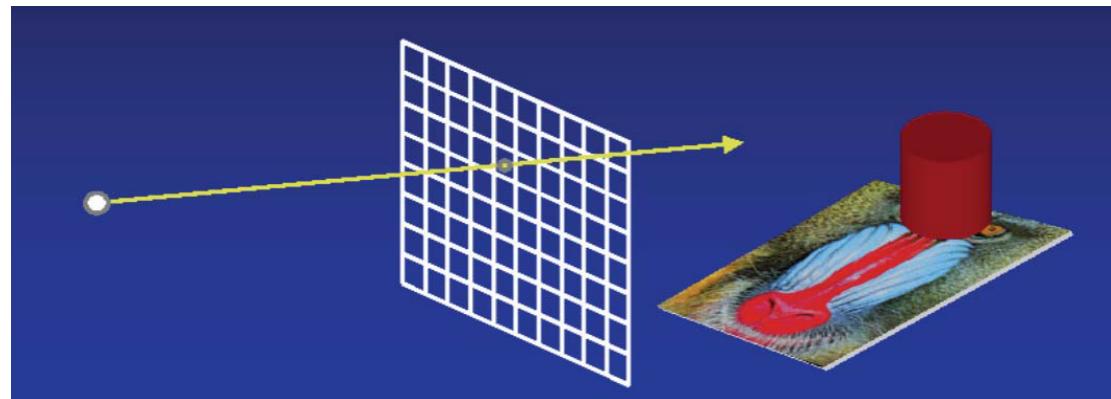
- Vom Augpunkt werden Strahlen durch die Pixel in die Szene geschickt



## 9.5 Globale Beleuchtung

### Ray-Tracing

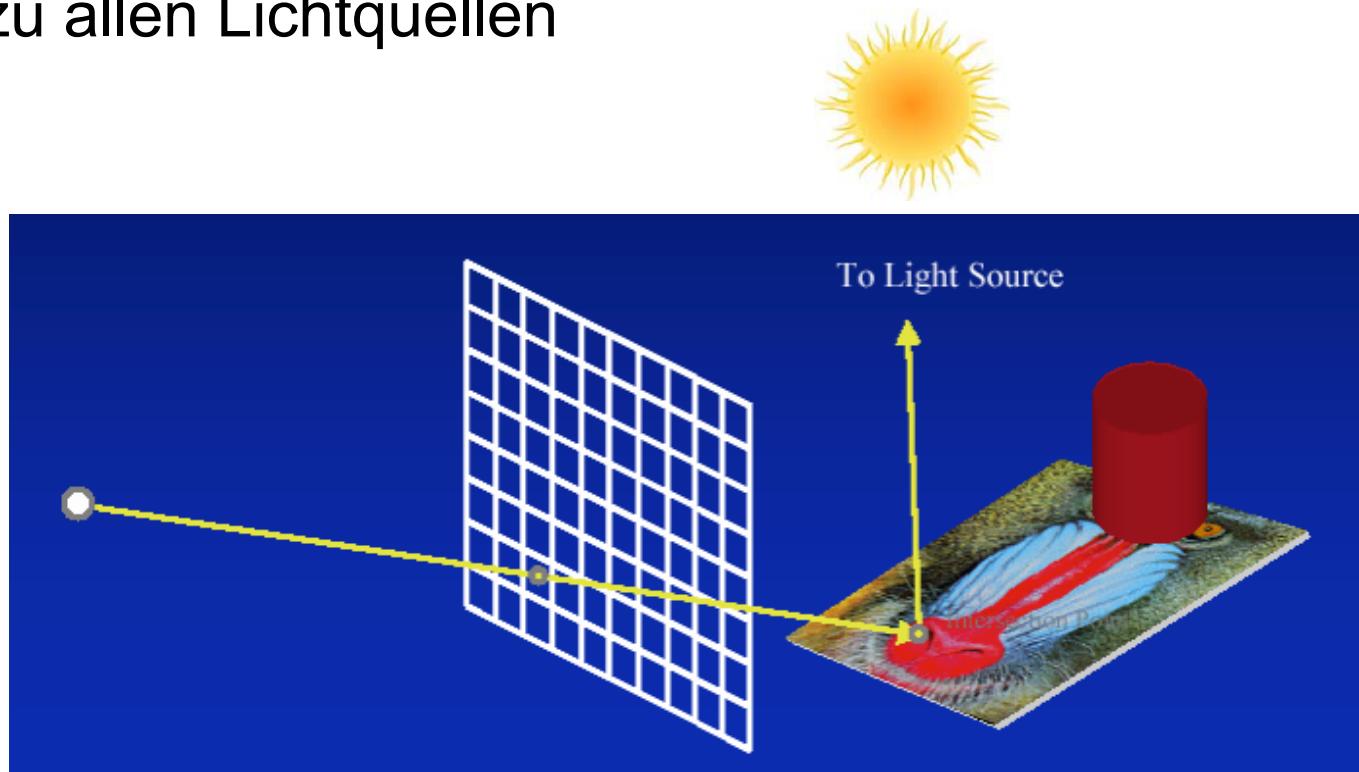
- Schneidet ein Strahl mehr als ein Objekt, wird das nächstliegende gezeichnet
- Sonst erscheint die Hintergrundfarbe



## 9.5 Globale Beleuchtung

### Ray-Tracing

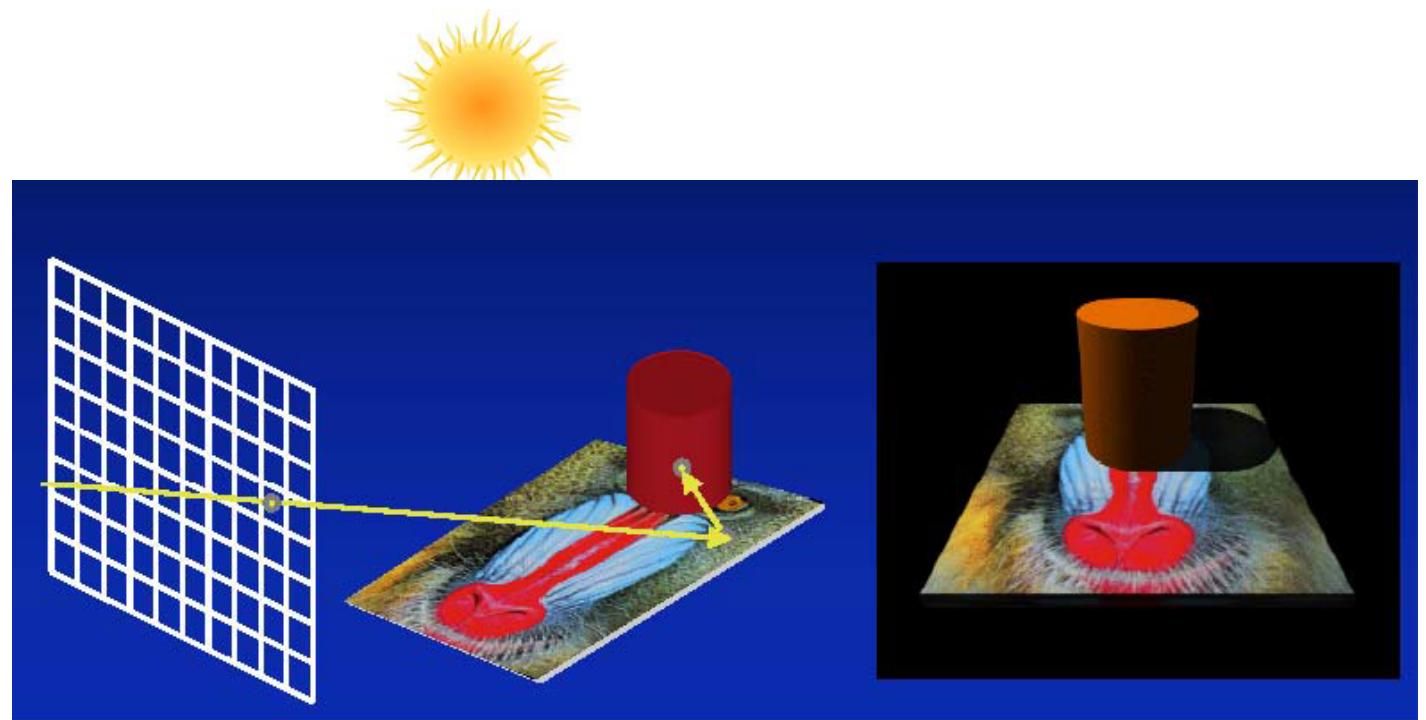
- Schneidet ein Strahl ein Objekt, so werden Schattenstrahlen vom Schnittpunkt zu allen Lichtquellen gesendet



## 9.5 Globale Beleuchtung

### Ray-Tracing

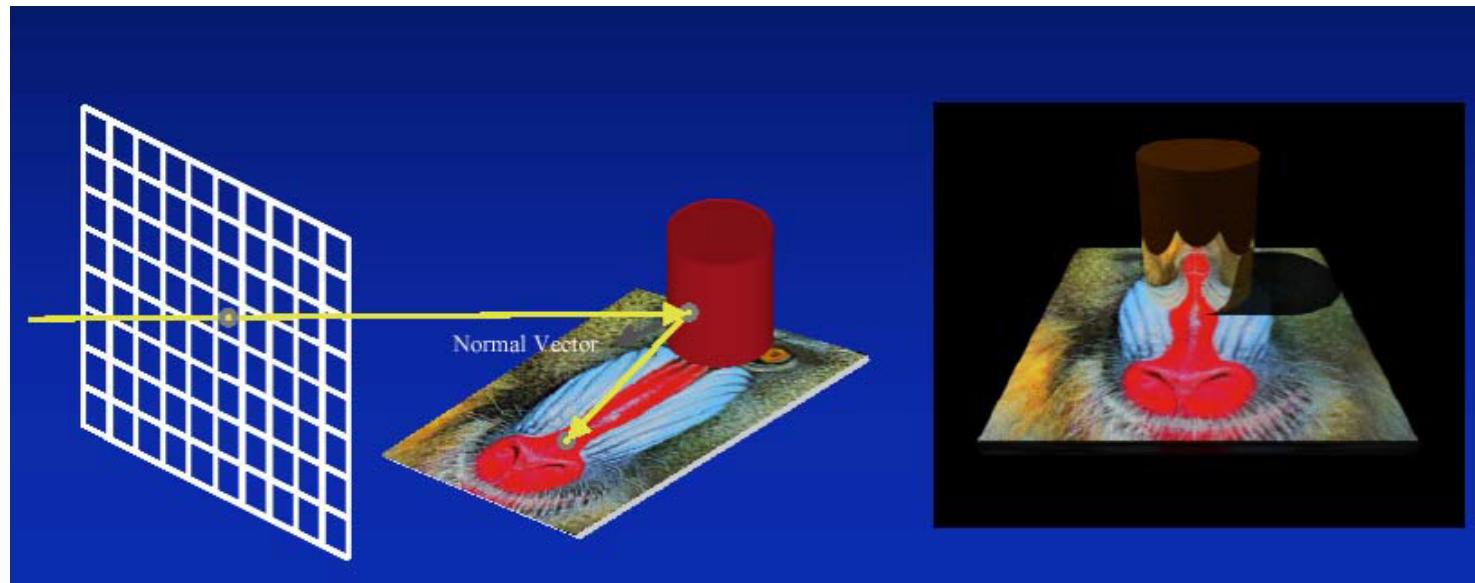
- Treffen die Schattenstrahlen auf ein Objekt, so liegt der Punkt im Schatten



## 9.5 Globale Beleuchtung

### Ray-Tracing

- Ist die Oberfläche spiegelnd, wird ein reflektierter Strahl verfolgt



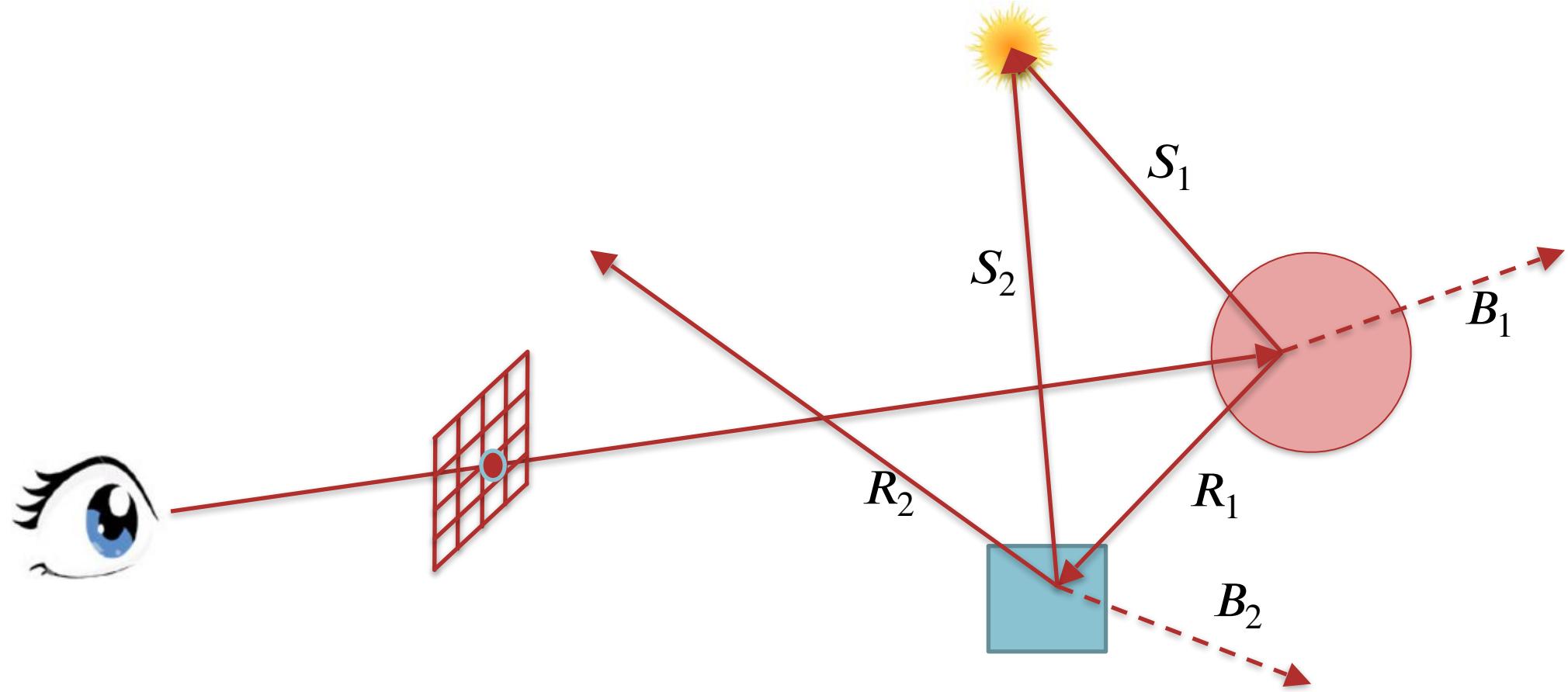
## 9.5 Globale Beleuchtung

### Ray-Tracing: Algorithmus

- Für jeden Strahl
  - Bestimme Schnittpunkt mit nächstliegendem Objekt
  - Für jede Lichtquelle
    - Sende Schattenstrahl zur Lichtquelle
    - Falls Schattenstrahl kein Objekt schneidet, werte Phong-Beleuchtungsmodell im Schnittpunkt aus
  - Verfolge ideal reflektierten Lichtstrahl und addiere Leuchtdichte aus dieser Richtung (Rekursion)
  - Gib summierte Leuchtdichte zurück

## 9.5 Globale Beleuchtung

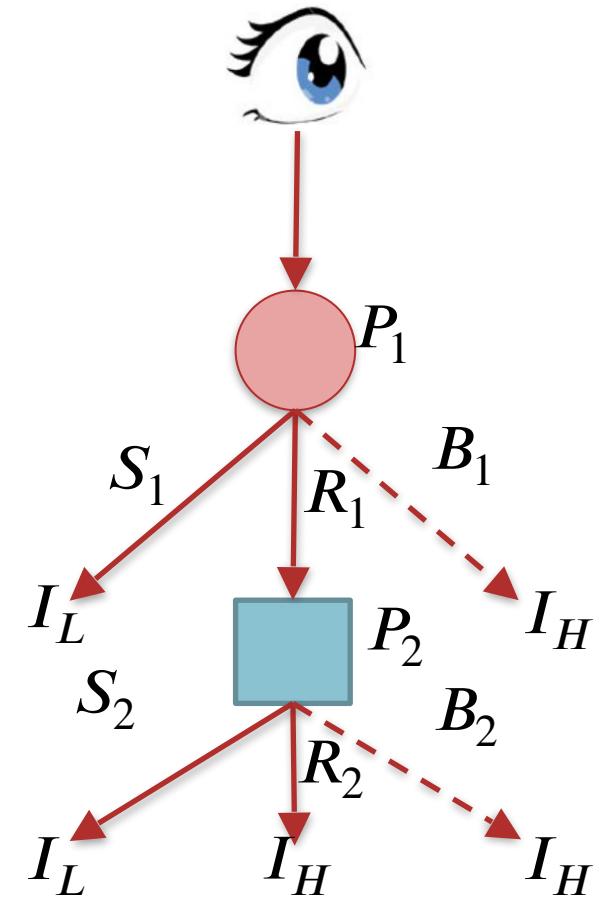
### Ray-Tracing



## 9.5 Globale Beleuchtung

### Ray-Tracing

- Strahlenbaum
  - Beschreibung der rekursiven Strahlverfolgung
  - Traversierung zur Bestimmung der Pixel-Intensität, berücksichtigt:
    - Lichtquellenintensität  $I_L$
    - Hintergrundintensität  $I_H$
    - Materialkoeffizienten  $r$
    - Verdeckung von Lichtquellen  $V$

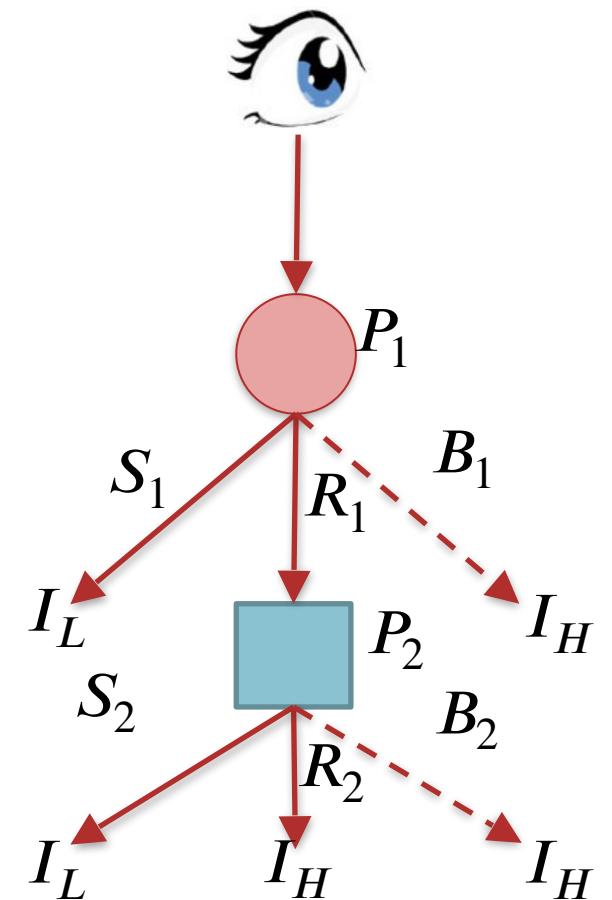


## 9.5 Globale Beleuchtung

### Ray-Tracing

$$I_{ges} = V(P_1, L) \cdot I_{phong, P_1} + r_{r, P_1} \cdot I_{R_1} + r_{b, P_1} \cdot I_{B_1}$$

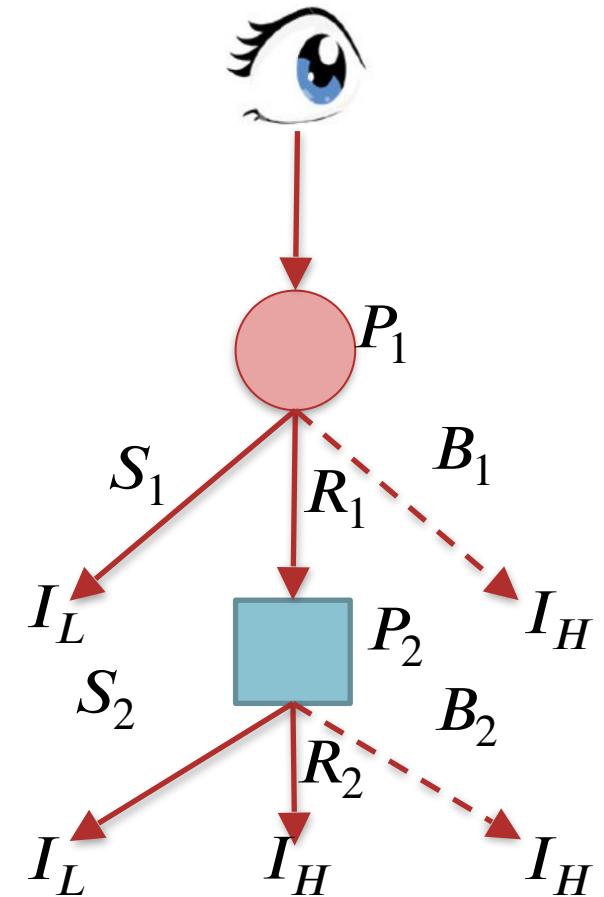
$$I_{R_1} = V(P_2, L) \cdot I_{phong, P_2} + r_{r, P_2} \cdot I_{R_2} + r_{b, P_2} \cdot I_{B_2}$$



## 9.5 Globale Beleuchtung

### Ray-Tracing

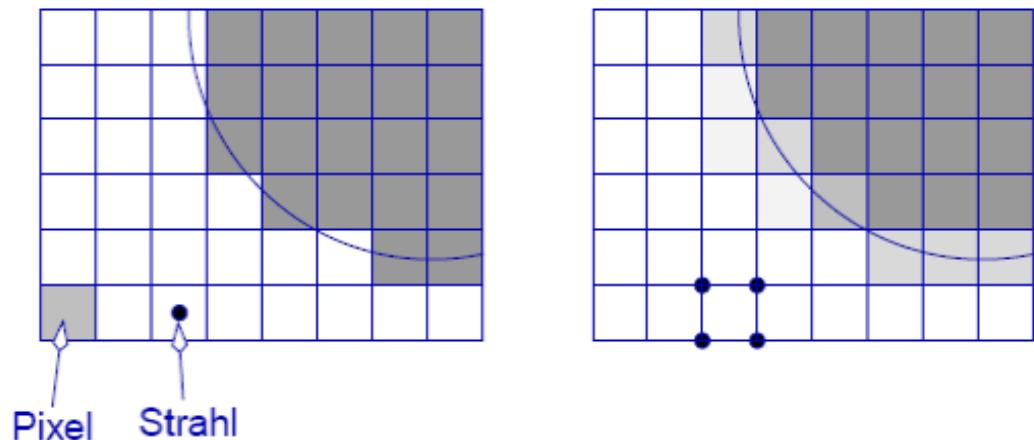
- Abbruchbedingung der Rekursionen
  - Verfolgte Strahlen schneiden keine Objekte mehr (verlassen die Szene, das Frustrum)
  - Vorgegebene maximale Baumtiefe ist erreicht
  - Farbanteil an der Pixelfarbe eines zu verfolgenden Teilstrahls ist zu gering (*adaptive Rekursionstiefenkontrolle*)



## 9.5 Globale Beleuchtung

### Ray-Tracing

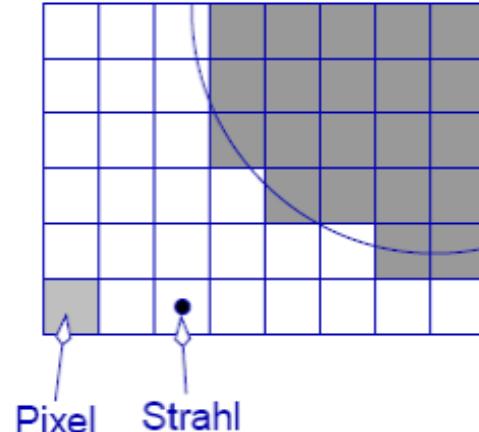
- Problem Aliasierung
  - Strahlen ohne Ausdehnung verursachen Abtastfehler



## 9.5 Globale Beleuchtung

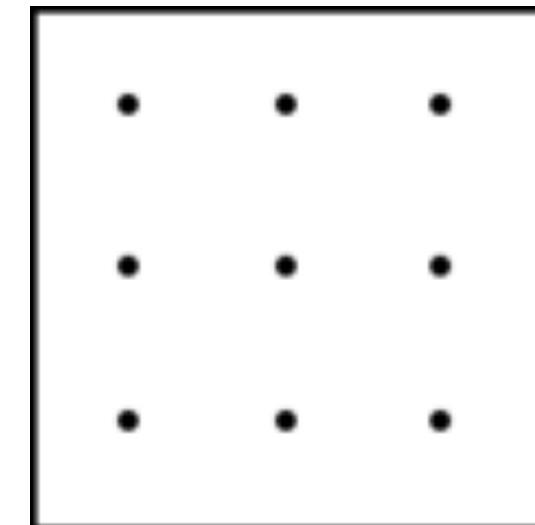
### Ray-Tracing

- Problem Aliasierung
  - Strahlen ohne Ausdehnung verursachen Abtastfehler

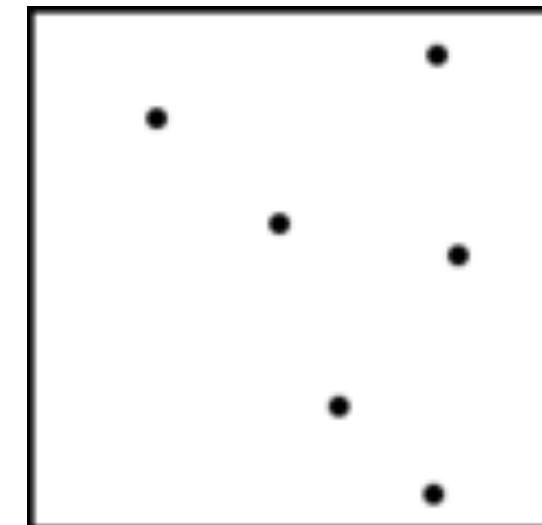


- Lösung
  - Supersampling:
    - Mehrere Strahlen pro Pixel
    - Mittelung der Farbwerte

Gitter



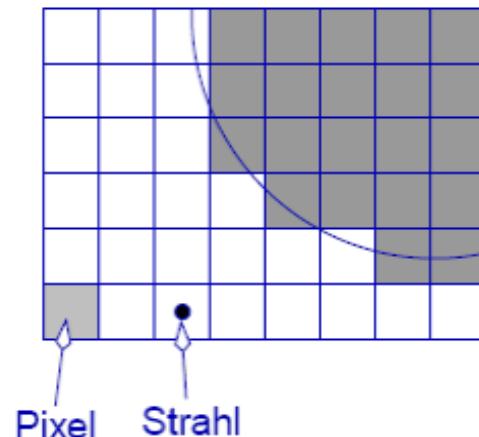
Zufällig



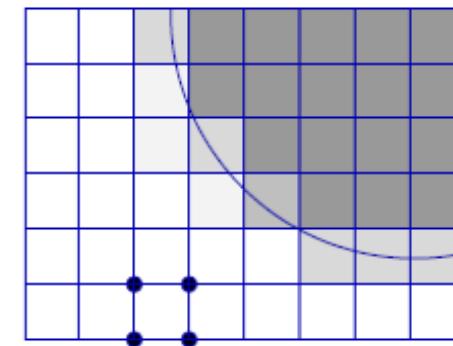
## 9.5 Globale Beleuchtung

### Ray-Tracing

- Problem Aliasierung
  - Strahlen ohne Ausdehnung verursachen Abtastfehler



- Lösung
  - Adaptives Supersampling
    - An Grenzen mehr Strahlen als auf homogenen Flächen
    - Vorgehen:
      - Senden einiger Strahlen pro Pixel
      - Unterscheiden sich die Werte, werden weitere gesendet
    - Stochastische Verteilung möglich



## 9.5 Globale Beleuchtung

### Ray-Tracing

- Zeitaufwand
  - Schnittpunktberechnungen  
Strahl-Objekt aufwändig
  - Viele Berechnungen  $O(n \cdot m)$ 
    - $n$ : Anzahl der Objekte
    - $m$ : Anzahl der Strahlen
  - Rekursion in der Strahlenverfolgung
  - Anzahl der Lichtquellen
  - Flächige Lichtquellen
- Beschleunigung
  - Weniger Schnittpunktberechnungen
    - Hüllkörperhierarchien (z.B. Octrees)
  - Light Buffer
  - Bounding Boxes
    - Schneller Ausschluss von Schnitten
  - Weniger Strahlen
    - Adaptive Rekursionstiefenkontrolle
    - Adaptives Anti-Aliasing

## 9.5 Globale Beleuchtung

### Ray-Tracing

- Vorteile
  - Hervorragend geeignet für
    - Spiegelungen
    - Punktlichtquellen
  - Gute Simulation des physikalischen Beleuchtungsvorgangs (Strahlengang)
  - Sichtbarkeitsproblem automatisch gelöst
  - Implizite Schattenberechnung
  - Große Wirklichkeitsnähe
- Nachteile
  - Weniger geeignet für
    - diffuse Reflektion
    - Flächige Lichtquellen
  - Großer bis sehr großer Rechenaufwand
  - Ideale Reflexion und Brechung entspricht nicht der Realität und lässt gerenderte Szenen unecht erscheinen
  - Keine Unterscheidung nach Farbe ( $\lambda$ ) bei Reflexion und Brechung

## 9.5 Globale Beleuchtung

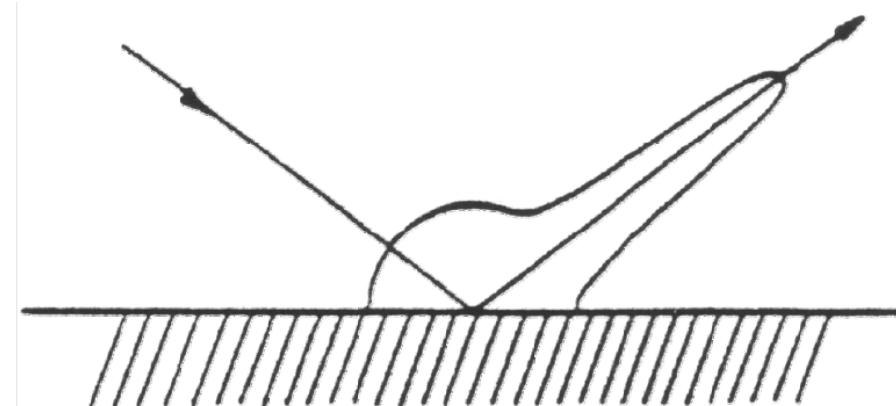
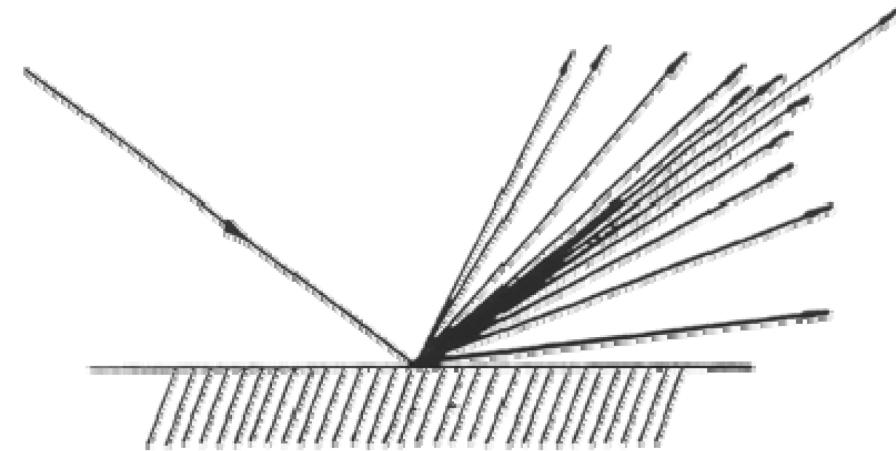
### Distributed Ray-Tracing

- Verbesserung:
  - Distributed ray tracing
    - Auch: distribution ray tracing
    - stochastic ray tracing
- Ziel: Realistisch unscharfe Effekte
  - Verschwommene Reflexionen und Brechungen
  - Weiche Schatten
  - Tiefunschärfe
  - Bewegungsunschärfe

## 9.5 Globale Beleuchtung

### Distributed Ray-Tracing

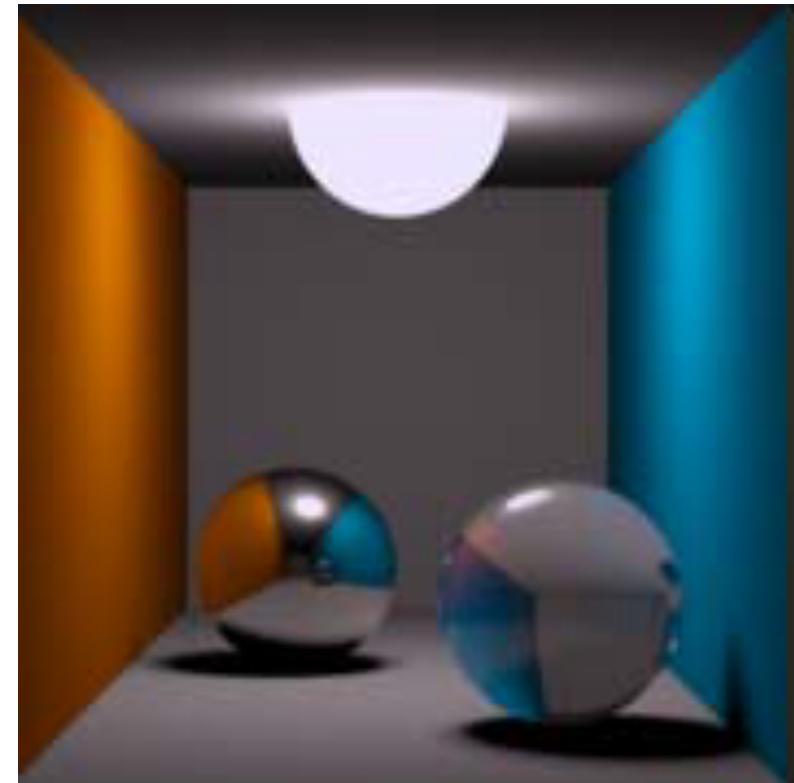
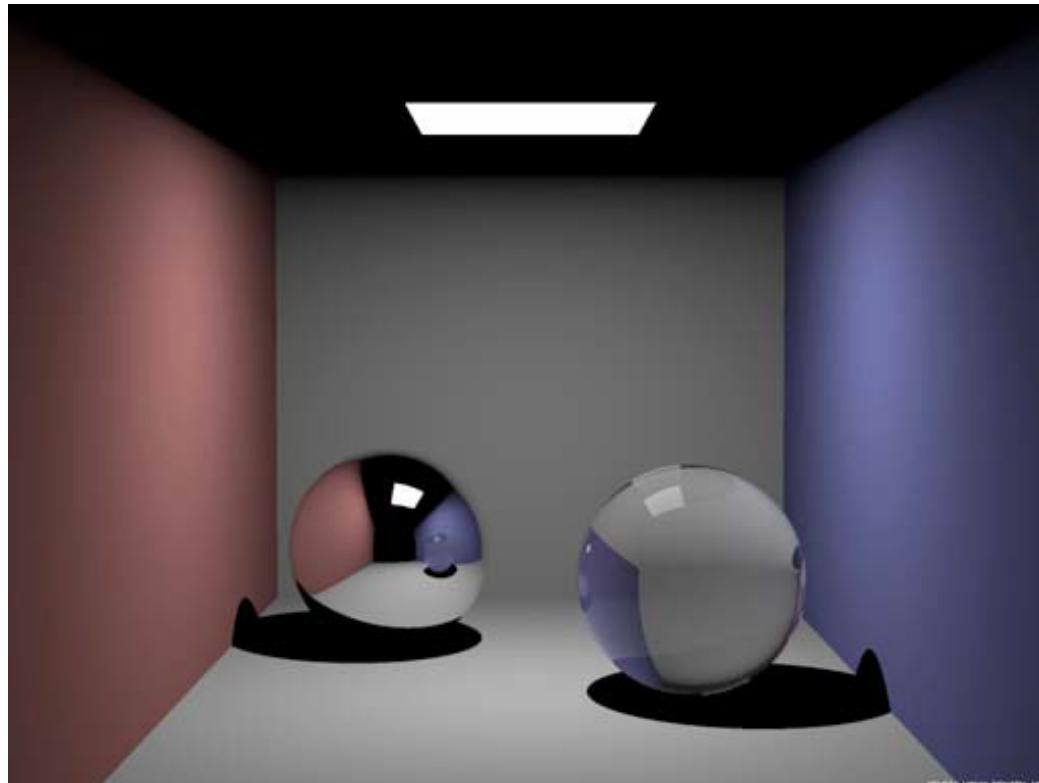
- Verschwommene Reflexionen
  - Vervielfachung der reflektierten Strahlen
  - Stochastische Verteilung der Strahlrichtungen
  - Anschließende Mittelung der Farbwerte
  - Auch für Brechung anzuwenden



## 9.5 Globale Beleuchtung

### Distributed Ray-Tracing

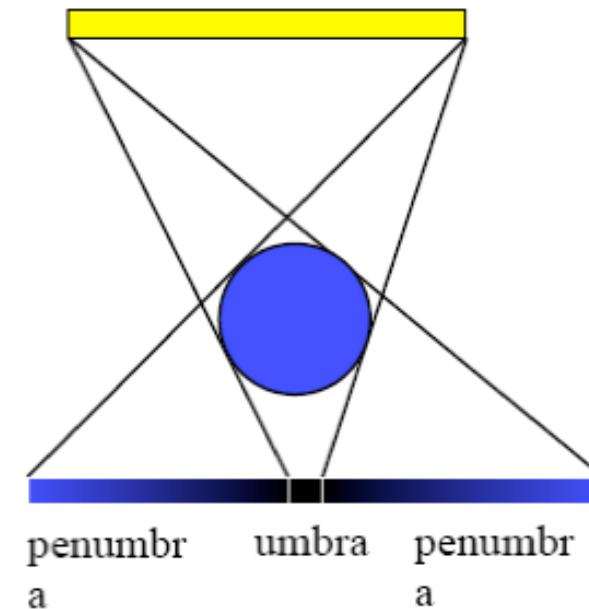
- Verschwommene Reflexionen



## 9.5 Globale Beleuchtung

### Distributed Ray-Tracing

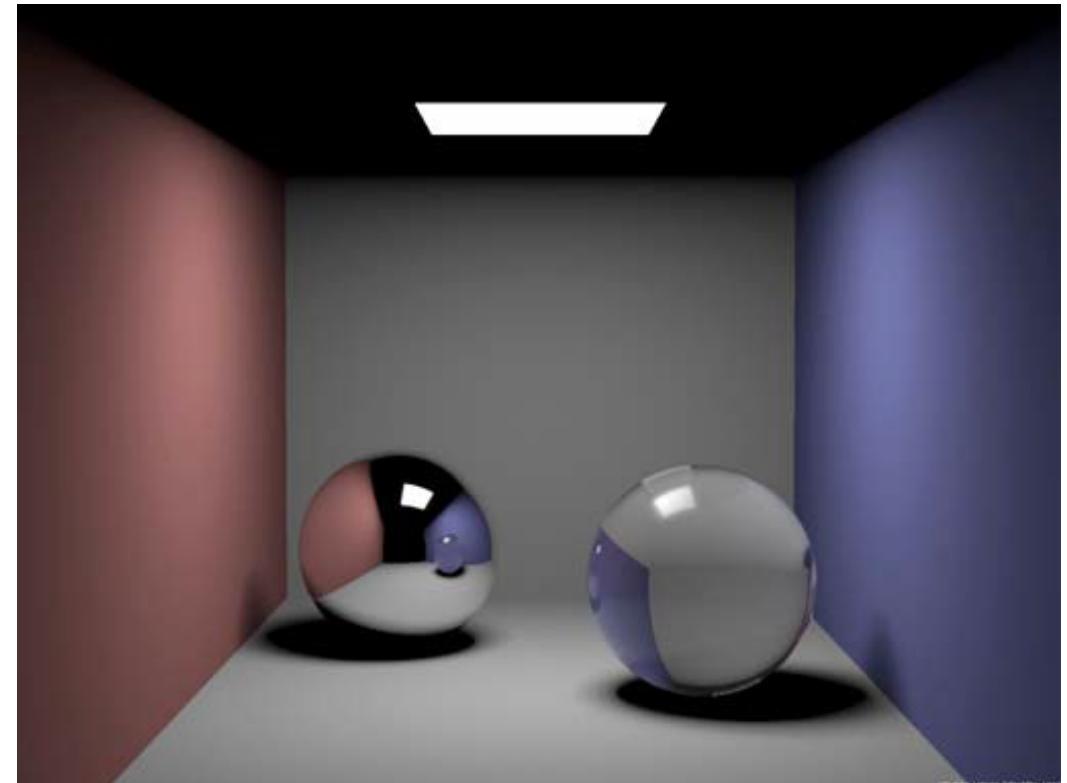
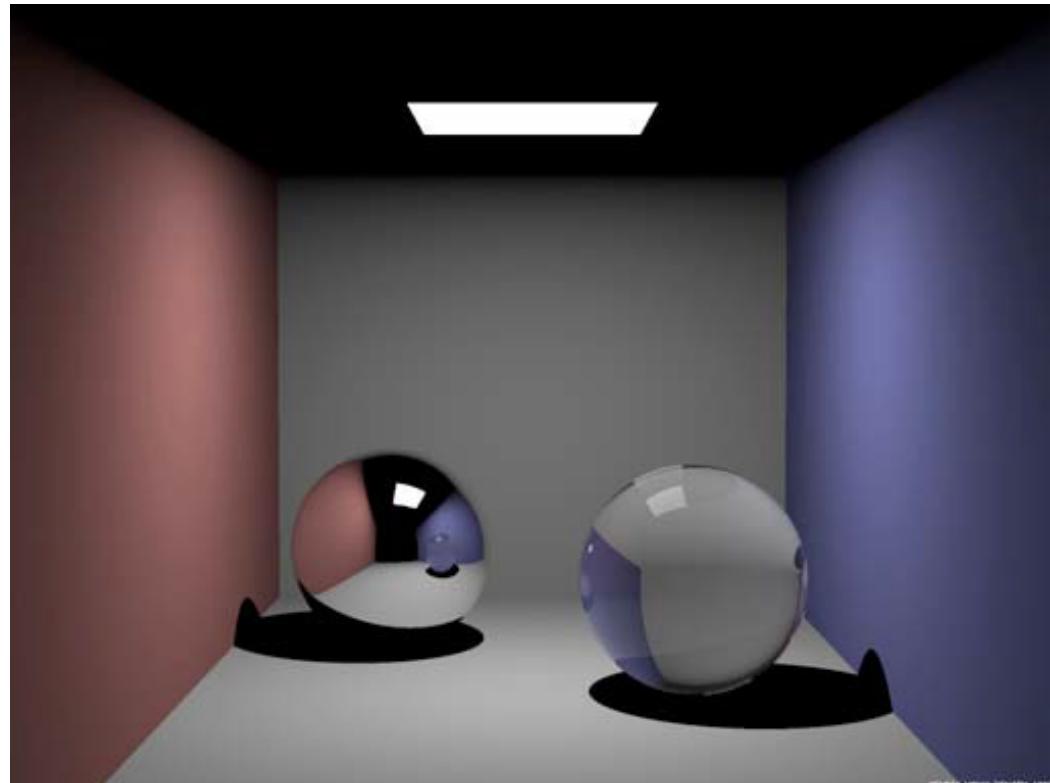
- Weiche Schatten
  - Verwendung von Flächenlichtquellen
  - Simuliert durch Vielzahl punktförmiger Lichtquellen
  - Stochastische Verteilung der Schattenstrahlen
  - Stärke des Schattens entspricht Verhältnis von verdeckten zur Gesamtzahl an Schattenstrahlen



## 9.5 Globale Beleuchtung

Distributed Ray-Tracing

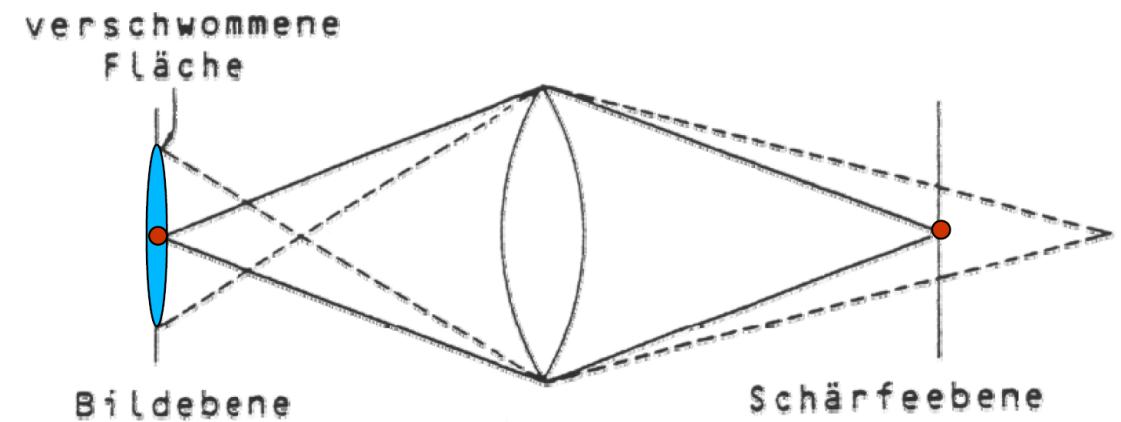
- Weiche Schatten



## 9.5 Globale Beleuchtung

### Distributed Ray-Tracing

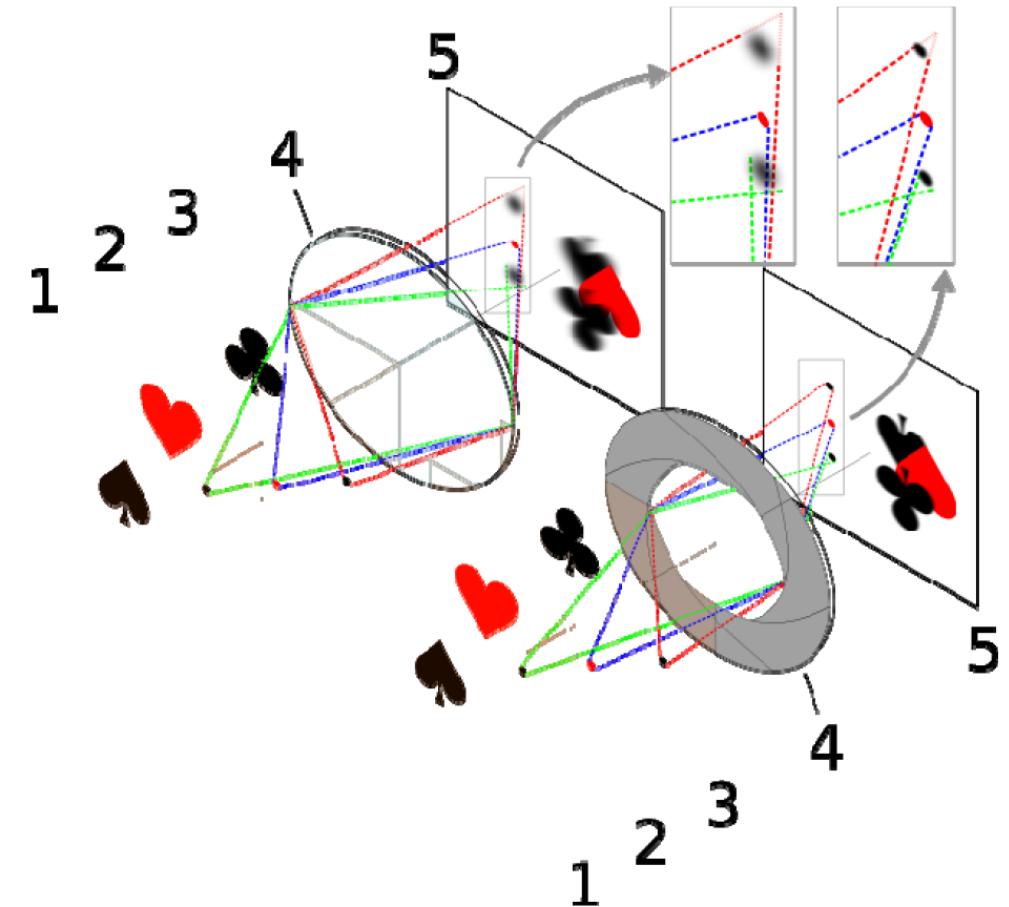
- Tiefunschärfe
  - Simulation der Blendenöffnung einer Kamera
  - Punkte außerhalb der Schärfeebeine erscheinen verschwommen (wenn größer als circle of confusion)
  - Korrekte Berechnung der Linsenbrechung mit stochastischer Strahlverteilung über Linsenoberfläche



## 9.5 Globale Beleuchtung

### Distributed Ray-Tracing

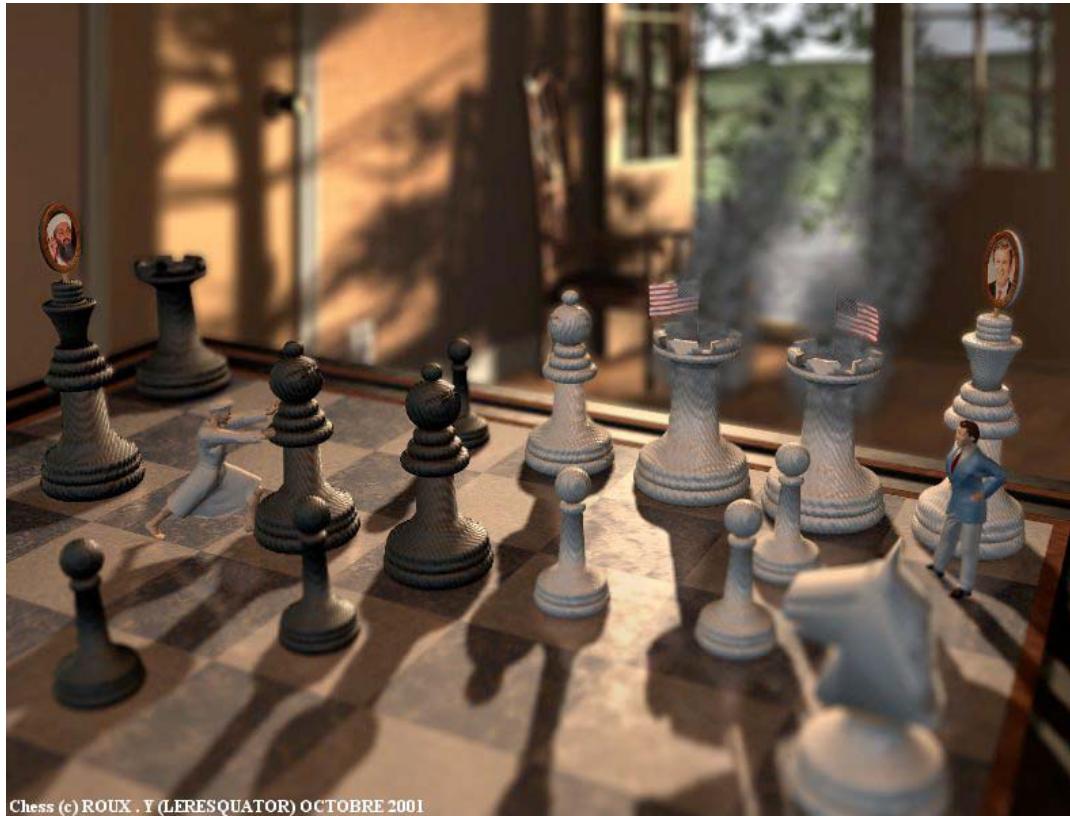
- Tiefunschärfe
  - Blendenöffnung bestimmt Bereich der Szene, der scharf abgebildet wird (depth of field)



## 9.5 Globale Beleuchtung

### Distributed Ray-Tracing

- Tiefenunschärfe



## 9.5 Globale Beleuchtung

### Distributed Ray-Tracing

- Bewegungsunschärfe
  - Zeitliche Verteilung der Augstrahlen
  - Senden von Strahlen durch gleichen Punkt mit zeitlicher Verzögerung
  - Filter bestimmt Verschlusszeit und damit Stärke des Motion Blur



## 9.5 Globale Beleuchtung

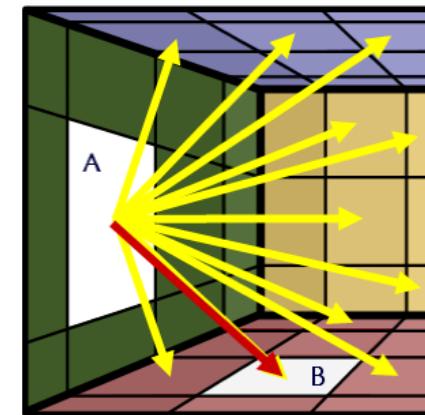
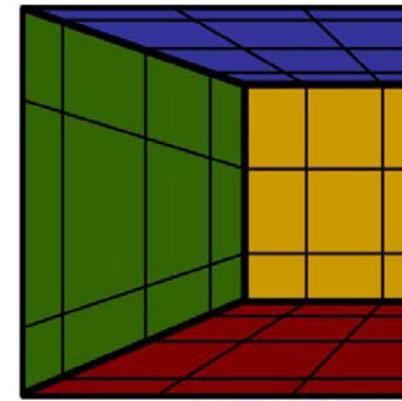
### Radiosity-Verfahren

- Modelliert rein diffuse Reflexion
- Ausbreitung des Lichts unter Beachtung des Energiegleichgewichts in einem geschlossenem System
- Gleichgewicht des
  - durch Lichtquellen zugeführten
  - durch Oberflächen absorbierten Lichtstroms
- Modellierung von unterschiedlichem Einfall des ambienten (indirekten) Lichts besser als beim klassischen Raytracing
  - Modellierung von Flächen-Lichtquellen
  - Modellierung von flächigen Schatten
- Gut geeignet für Lichtverteilung in Gebäuden

## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

- Gegeben:
  - Szene in Flächensegmente unterteilt
  - Geometrische Informationen über die gegenseitige Lage aller Objekte
  - Lichttechnische Kenngrößen der Objekte
- Gesucht:
  - Radiosity für jede Fläche
  - Dient dann zur Farbgebung beim Rendern der Szene



## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

$$B(x) = E(x) + \rho(x) \int_{x' \in S} B(x') \frac{1}{\pi r^2} \cos \phi_x \cos \phi_{x'} \cdot V(x, x') dx'$$

- $B(x)$  gesamte vom Punkt  $x$  abgestrahlte Energie als Leistung pro Flächeneinheit, genannt Bestrahlungsstärke (Radiosity)
- $E(x)$  im Punkt  $x$  abgegebene Eigenstrahlung
- $\rho(x)$  Reflexionsfaktor im Punkt  $x$
- $S$ : alle Oberflächen der Szene
- $r$  = Abstand der Punkte  $x$  und  $x'$
- $\phi(x)$  Winkel zwischen der Normalen im Punkt  $x$  und der Verbindungsline zwischen den Punkten  $x$  und  $x'$

$$V(x, x') = \begin{cases} 1, & \text{wenn } x \text{ von } x' \text{ aus sichtbar} \\ 0, & \text{sonst} \end{cases}$$

## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

$$B(x) = E(x) + \rho(x) \int_{x' \in S} B(x') F(x, x') dx'$$

- $B(x)$  gesamte vom Punkt  $x$  abgestrahlte Energie als Leistung pro Flächeneinheit, genannt Bestrahlungsstärke (Radiosity)
- $E(x)$  im Punkt  $x$  abgegebene Eigenstrahlung
- $\rho(x)$  Reflexionsfaktor im Punkt  $x$
- $S$ : alle Oberflächen der Szene
- $F(x, x')$  Formfaktor, Anteil an der von  $x'$  abgegebenen Energie, die  $x$  erreicht

## 9.5 Globale Beleuchtung

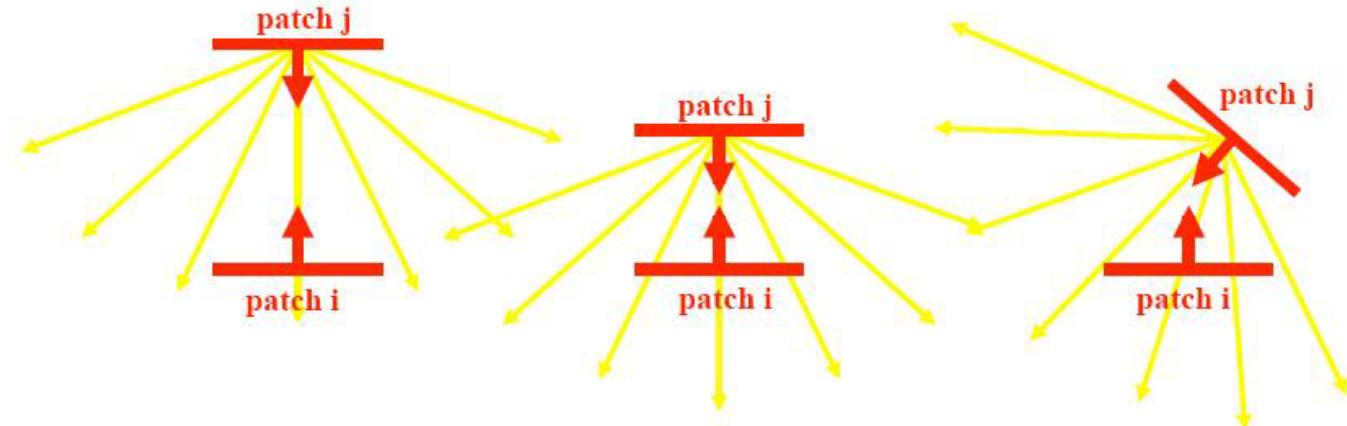
### Radiosity-Verfahren

- Diskretisierung
- Statt infinitesimaler Flächenelemente  $dx$ :
  - Finite-Elemente-Methode
  - Kleine, zusammenhängende Teilflächen (Facetten oder Patches)  $A_i$
- Aus Integralgleichung wird lineares Gleichungssystem
  - $$B_i = E_i + \rho_i \sum_{j=1}^n B_j F_{ij}$$
  - $B_i$  gesamte von Fläche  $A_i$  abgestrahlte Energie
  - $E_i$  von der Teilfläche i abgegebene Eigenstrahlung
  - $F_{ij}$  Anteil an der von Fläche j abgegebenen Energie, die auf Fläche i auftrifft
  - $n$  Anzahl der Teilflächen

## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

- Formfaktoren  $F_{ij}$ 
  - Anteil des Lichtstroms von  $A_i$ , der  $A_j$  erreicht
  - Nur abhängig von
    - Geometrie
    - Abstand
    - Orientierung
    - Größe
    - Sichtbarkeit



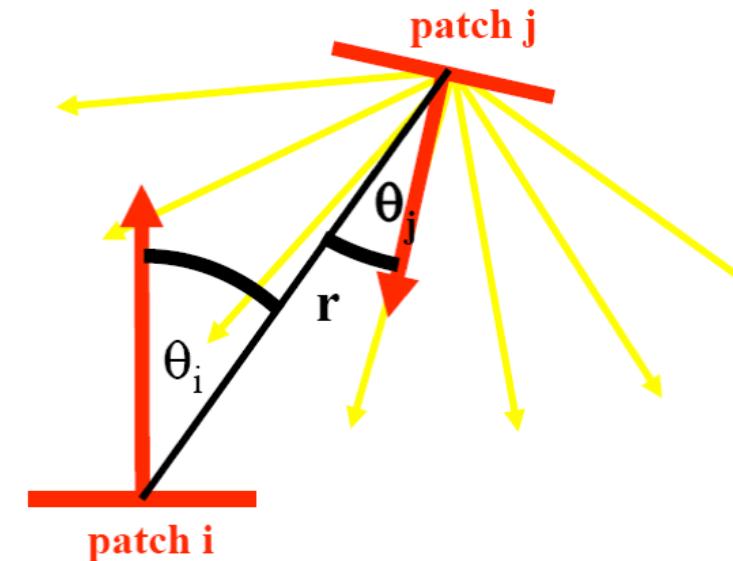
## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

- Berechnung der Formfaktoren  $F_{ij}$

$$F_{i,j} = \frac{1}{A_i} \int_{v \in A_i} \int_{u \in A_j} \frac{1}{\pi r^2} \cos \phi_u \cos \phi_v V(u, v) dA_j dA_i$$

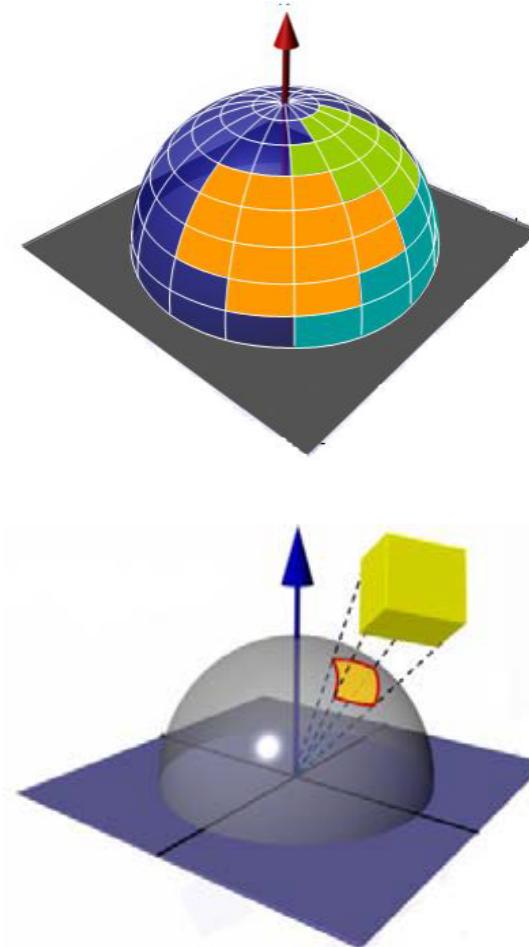
- $r$  Abstand
- $\phi$  Winkel zwischen der Normalen und der Verbindungsline
- $V(u, v) = \begin{cases} 1, & \text{wenn } i \text{ von } j \text{ aus sichtbar} \\ 0, & \text{sonst} \end{cases}$
- Einfachste Approximation
- $F_{i,j} \approx \frac{\cos \phi_u \cos \phi_v}{\pi r^2} A_i V_{ij}$



## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

- Approximation der Formfaktoren  $F_{SE_i}$ 
  1. Senderfläche  $S$  in Patches  $S_i$  unterteilen
  2. Hemisphäre für jedes Patch in Normalenrichtung erstellen
  3. Empfängerflächen  $E_i$  auf Hemisphäre projizieren (Verdeckung beachten)
  4. Für jedes  $E_i$  Anteil an Hemisphäre bestimmen, der von ihr bedeckt wird
  5. Beiträge von  $E_i$  über alle Patches  $S_i$  mitteln



## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

- Lösung des Radiosity-Gleichungssystems

$$B_i = E_i + \rho_i \sum_{j=1}^n B_j F_{ij}$$

- Matrixschreibweise

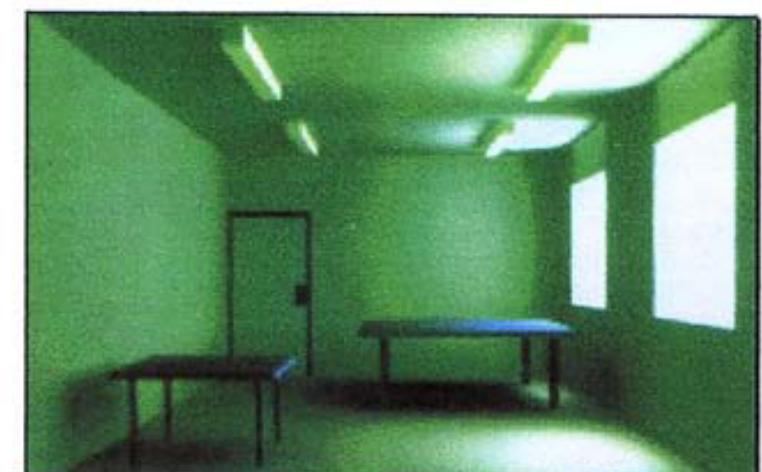
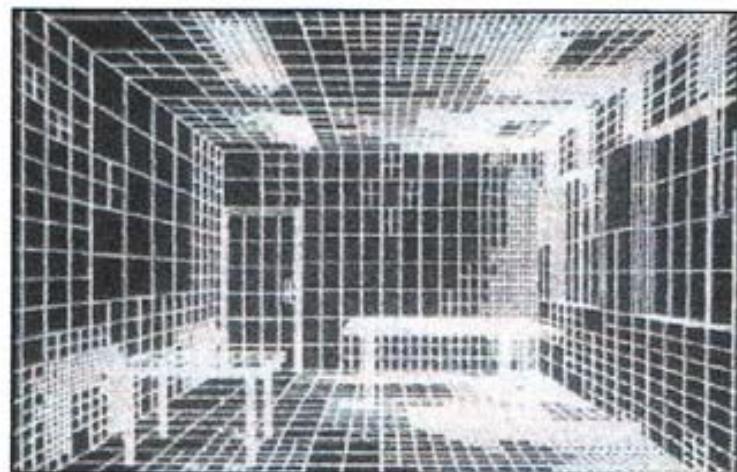
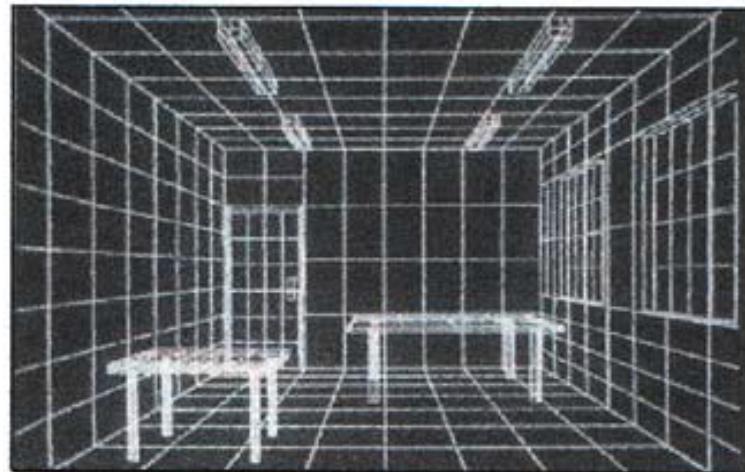
$$B = E + \varrho F B \Rightarrow (I - \varrho F) B = E$$

- Dieses System wird getrennt für die verschiedenen Frequenzen des Lichts (z.B. RGB) gelöst
- Die Koeffizientenmatrix ist diagonaldominant, lösbar mit
  - Iterativem Verfahren, wie Gauss-Seidel
  - Progressive refinement
  - „Gathering“ bzw. „Sammeln“

## 9.5 Globale Beleuchtung

### Radiosity-Verfahren

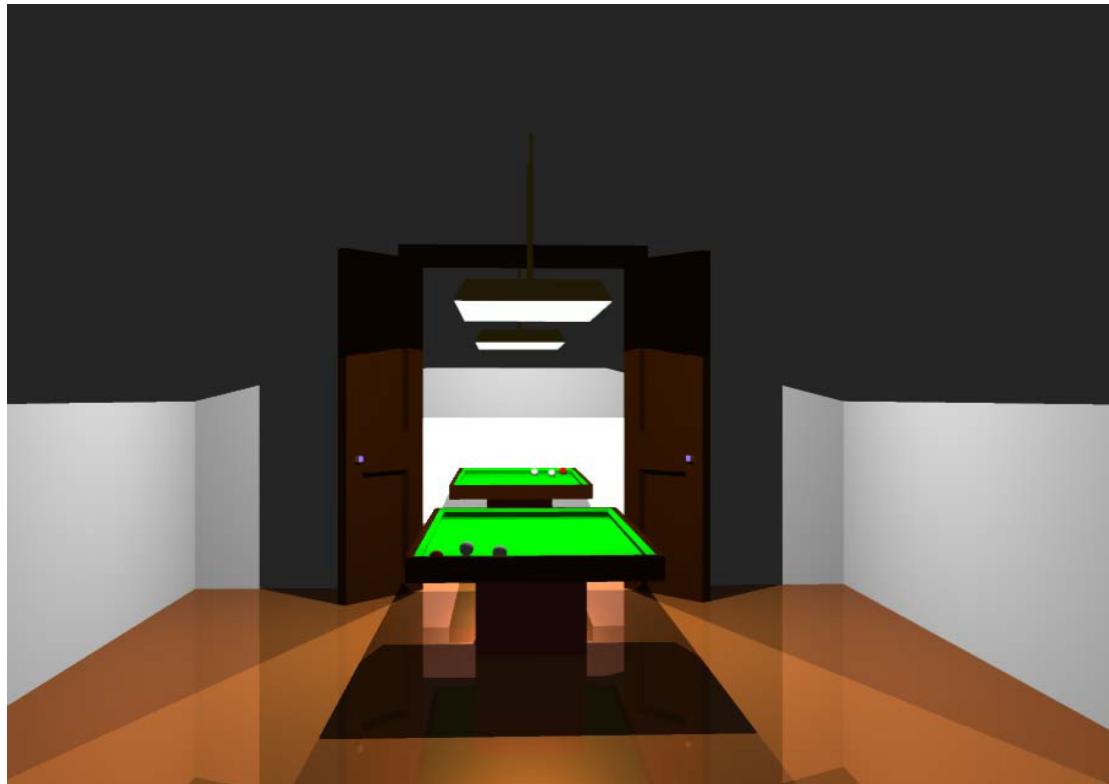
- Je feiner die Aufteilung der Szene in Patches, desto besser das Ergebnis
- Aufwand steigt mit Anzahl  $n$  der Patches, z.B. Anzahl der Formfaktoren  $O(n^2)$



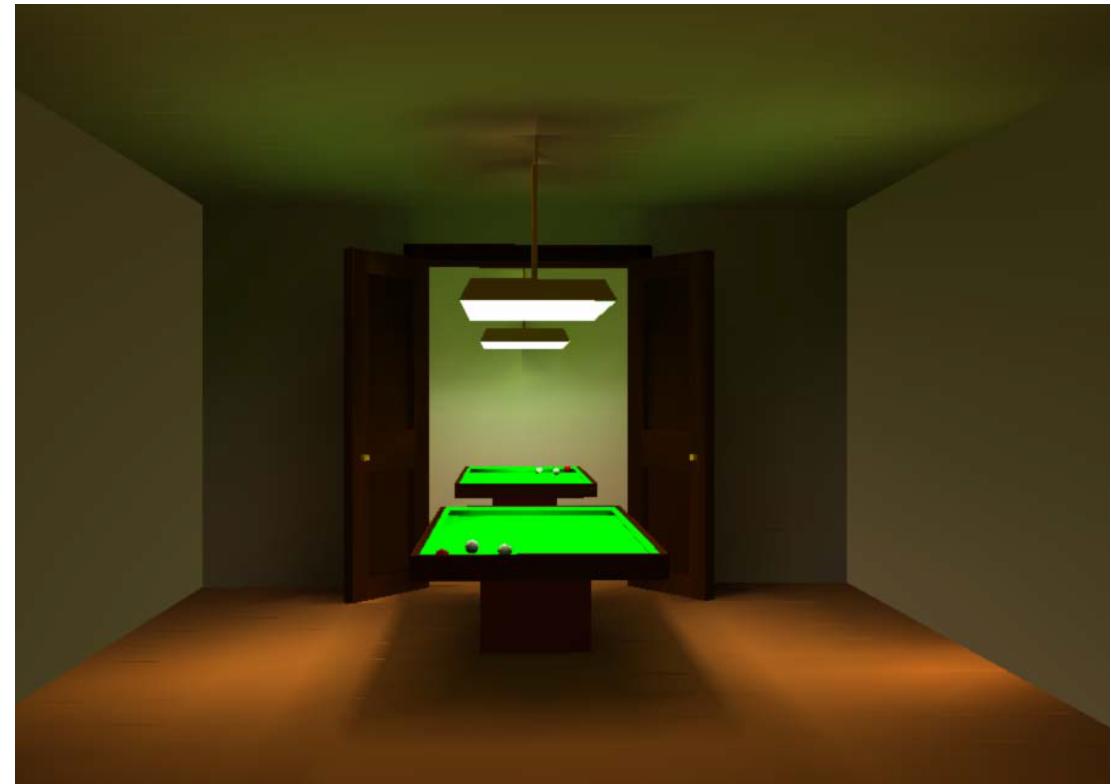
## 9.5 Globale Beleuchtung

### Ergebnisse

– Raytracing



– Radiosity



## 9.5 Globale Beleuchtung

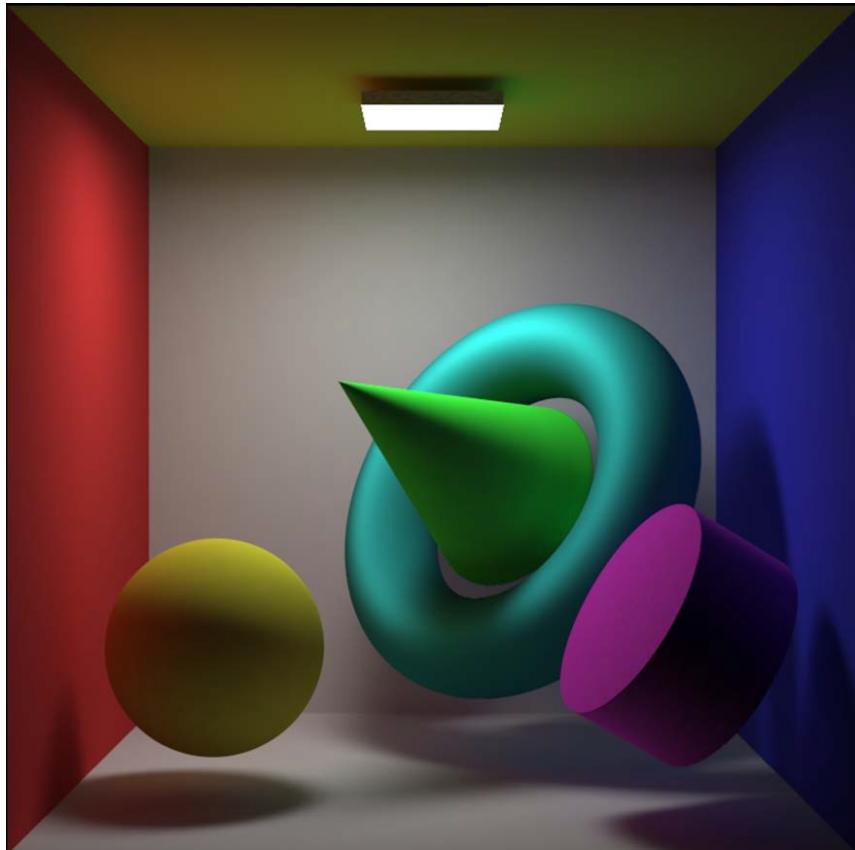
### Radiosity-Verfahren

- Vorteile
- Blickpunktunabhängigkeit
- Einfache indirekte, ideal diffuse Lichtreflexion
- Nachteile
- nur diffuse Reflexion
- keine analytischen Primitive
- Hoher Speicher- und Zeitbedarf

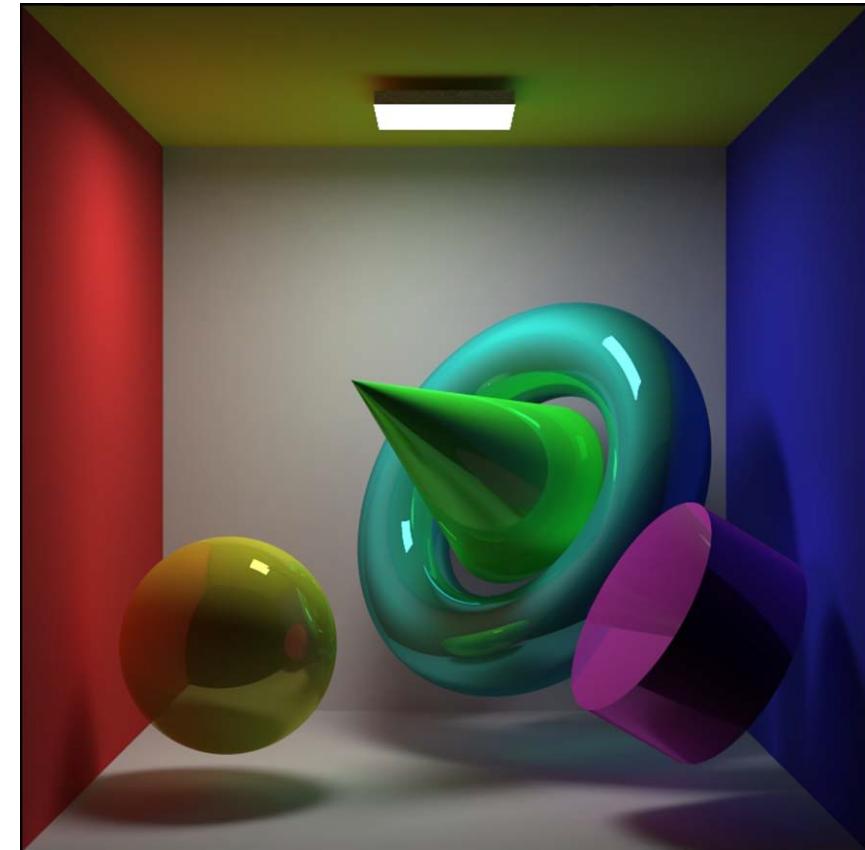
## 9.5 Globale Beleuchtung

### Ergebnisse

- Radiosity



- Raytracing & Radiosity



## 9.5 Globale Beleuchtung

### Path Tracing

- Kajiya's Rendergleichung (James Kajiya 1986)

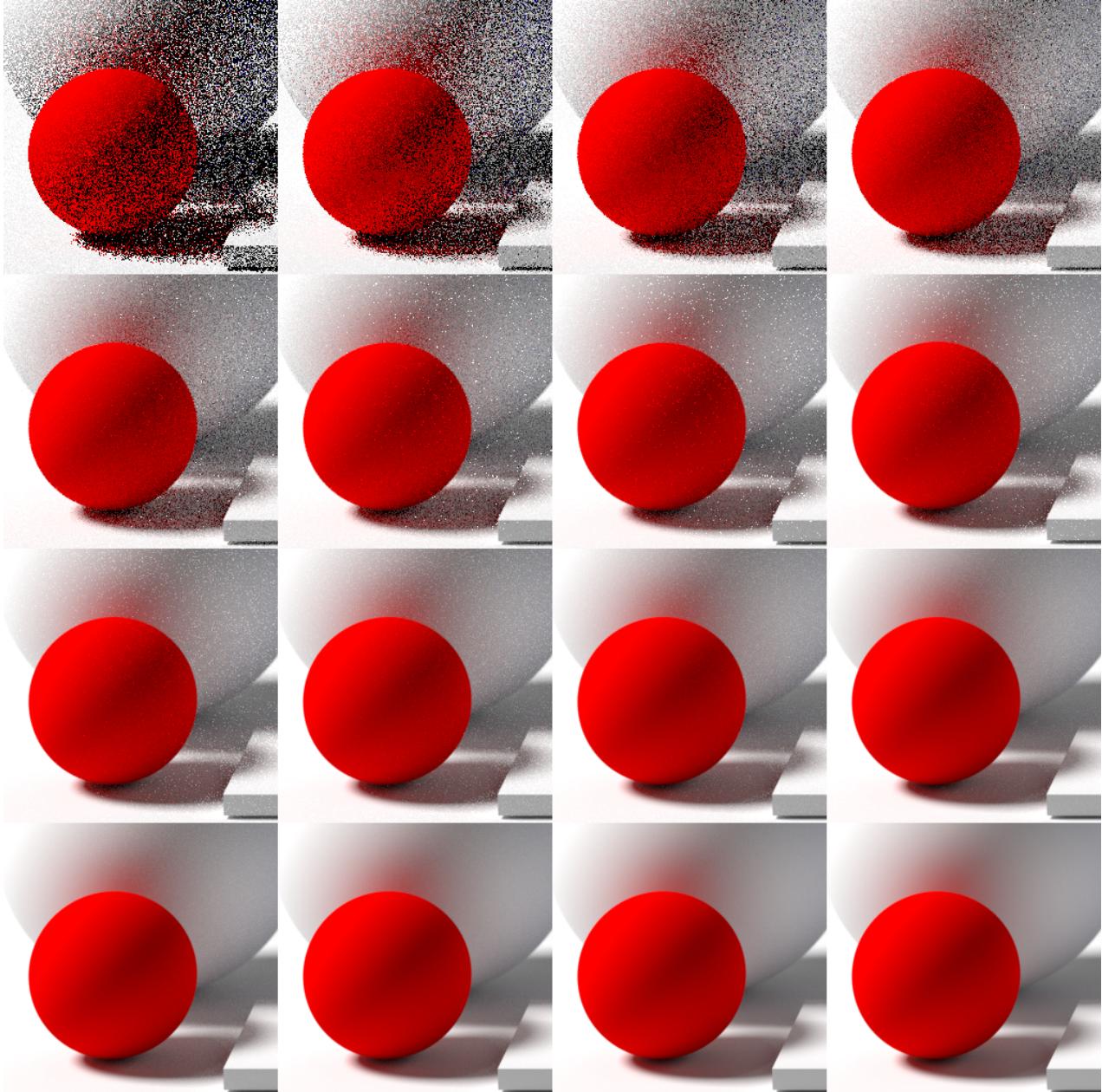
$$L(x, \vec{\omega}) = L_e(x, \vec{\omega}) + \int_{\Omega} f_r(x, \vec{\omega}', \vec{\omega}) L(x, \vec{\omega}') (\vec{\omega}' \cdot \vec{n}) d\vec{\omega}'$$

- $L(x, \vec{\omega})$  Energiefluss: wie viel Licht wird von  $x$  aus in Richtung  $\omega$  abgestrahlt
- $L_e(x, \vec{\omega})$  Emissionsterm: wie viel Licht wird von  $x$  aus in Richtung  $\omega$  ausgestrahlt
- $f_r(x, \vec{\omega}', \vec{\omega})$  Streuungsterm (BRDF) mit Einfallswinkel  $\omega'$  und Reflexionswinkel  $\omega$
- $n$  Normalenvektor der Oberfläche im Punkt  $x$
- $\Omega$  Gesamtheit aller Winkel der Kugel über der Oberfläche

## 9.5 Globale Beleuchtung

### Path Tracing

- Kajiya's Rendergleichung (James Kajiya 1986)
- Wird approximativ durch Monte Carlo Integration gelöst
- zufällige Richtung bei der Aussendung und jeder Reflexion oder Brechung
- anschließende Mittelung
- sehr aufwendig (ca. 5000 Strahlen pro Pixel nötig)
- physikalisch



## Quellen

- Computergraphik, Universität Leipzig  
(Prof. D. Bartz)
- Graphische Datenverarbeitung I,  
Universität Tübingen  
(Prof. W. Straßer)
- Graphische Datenverarbeitung I, TU  
Darmstadt  
(Prof. M. Alexa)
- Wikipedia: <http://en.wikipedia.org>