|  |  |
| --- | --- |
| Computergrafik | |
| **RayTracing-Algorithmus** | |
| Benotete Abgabe | |
| von | |
| MN1 | |
| MN2 | |
| und 9482203 | |
|  | |
| Abgabedatum:10.01.2021 | |
|  |  |
| Bearbeitungszeitraum | 6 Wochen |

Inhaltsverzeichnis

[Abbildungsverzeichnis 3](#_Toc59907515)

[Quellenverzeichnis 7](#_Toc59907516)

Abbildungsverzeichnis

* **Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

# Theoretische Prinzipien

## Grundlagen Raytracing

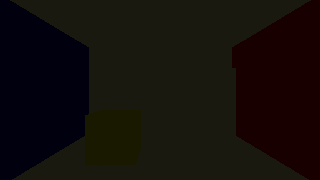
## Phong

Das Phong-Beleuchtungsmodell ist Beleuchtungsmodell, bei dem die Beleuchtung eines Punktes im Raum aus drei Werten, “Ambient”, “Diffuse” und “Specular, zusammengesetzt wird. Dabei wird nicht das Ziel verfolgt, die Realität möglichst realistisch abzubilden, sondern ein einfaches und schnell zu berechnendes Verfahren zu haben.

### Ambient Beleuchtungswert

Der “Umgebungswert” eines Punkts ist nur abhängig von dem Ambient Wert der Oberfläche und dem Ambient Wert der Lichtquelle, bzw. Der Lichtquellen. Er stellt somit eine konstante Beleuchtung des gesamten Raums dar, unabhängig von der Position von Lichtquelle oder Betrachter.

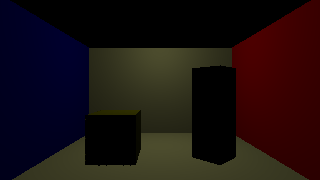
In Figure 1 sieht man ein Bild was ausschließlich mit ambienter Beleuchtung erzeugt wurde. Alle Wände sind gleichmäßig beleuchtet.

Figure 1: Bild nur mit Umgebungsbeleuchtung

### Diffuse Beleuchtungswert

Der diffuse Beleuchtungswert ist abhängig von den „Diffuse“-Werten der Lichtquelle und dem Objekt sowie dem Winkel der Oberfläche zur Lichtquelle. Steht der Normalenvektor der Oberfläche des Objekts auf dem Vektor zur Lichtquelle ist der diffuse Wert maximal.

In Figure 2 sieht man das Bild nur mit diffuser Beleuchtung. Die Flächen sind jetzt unterschiedlich beleuchtet und man sieht auf der dem Licht abgewandten Seite der Quader kein Licht, da das Skalarprodukt aus der Richtung des Lichts und des Normalenvektors der Oberfläche <= 0 ist.

Figure 2: Bild nur mit diffuser Beleuchtung

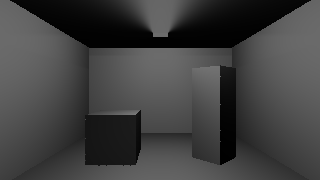
Da der „Ambient“- und der „Diffuse“-Wert nicht von der Position des Betrachters abhängen, kann man diese Werte bei einer Echtzeitanwendung im Voraus berechnen.

### Specular Beleuchtungswert

Der „Specular“-Beleuchtungswert soll spiegelnde, besonders helle Punkte einer Oberfläche betonen, in denen der Winkel zum Licht und zum Betrachter von der Oberflächennormale relativ ähnlich sind. Zur Berechnung wird die optimale Reflektionsachse bestimmt und mit der realen Reflektionsachse (der Oberflächennormale) verglichen.

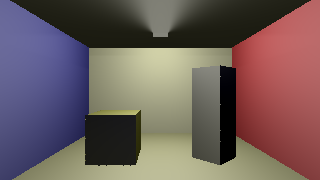
Je näher sich diese Werte sind (also je größer das Skalarprodukt zwischen den beiden Werten), desto höher ist der Specular-Beleuchtungswert für diesen Punkt. Dieses Skalarprodukt wird dann noch mit einem Wert „shinyness“ potenziert. Durch einen höheren „shinyness“-Wert erhält man eine Fläche die glänzender, bzw. Spiegelnder erscheint, während ein niedriger Wert zu einer eher matten Oberfläche führt.

Figure 3 zeigt das Bild nur mit „Specular“-Beleuchtung. Besonders an der Lichtquelle kann man die „harten“ Reflektionen gut sehen.

Figure 3: Bild nur mit "Specular" Beleuchtung

### Zusammensetzung der Beleuchtungswerte

Für jeden Punkt werden die drei Werte addiert. Dadurch ergibt sich das Bild in Figure 4. Durch das Anpassen von den Parametern kann das Bild entsprechend verändert werden.

Figure 4: Zusammengesetztes Bild

# Algorithmische Umsetzung

## **Kollisionerkennung von Rays mit Objekten**

## Quadratische Lichtquelle

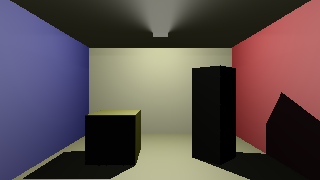
Da eine quadratische Lichtquelle gefordert ist, aber die in Kapitel 1 beschriebene Phong-Methode nur mit eindimensionalen Lichtquellen arbeitet, stellt das Berechnen der Beleuchtung eines Punktes ein besonderes Problem dar. Zusätzlich ist der einfache Ansatz zur Schattenbestimmung eines Punktes durch Werfen eines Strahls zur (punktförmigen) Lichtquelle, bedingt durch die zweidimensionale Ausdehnung der Lichtquelle, nicht möglich. Um weiche Schatten zu erhalten muss also ein anderer Ansatz gewählt werden.

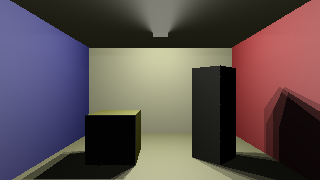
### Berechnung der Schatten

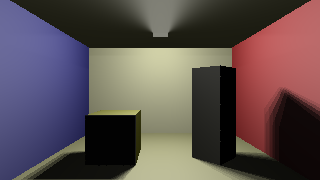
Zur Berechnung der Schatten wird ein primitiver Algorithmus verwendet, der die Schattenbestimmung von einer eindimensionalen Lichtquelle auf eine mehrdimensionale Lichtquelle portiert. Statt zu überprüfen, ob zwischen zwei Punkten ein Objekt liegt, wird überprüft wie viele Strahlen ungestört von einem Punkt auf dem Objekt zu mehreren anderen Punkten auf dem Licht reichen. Es werden also von dem zu prüfenden Punkt mehrere „Schattenfühler“ (Shadow Rays) zur Lichtquelle geschickt und aus dem Anteil der angekommen Fühler ein Schattenwert für diesen Punkt berechnet.

TODO: Skizze mit Zeichentablet einfügen

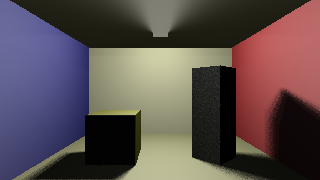
Diese Methode approximiert die Lichtfläche nur durch mehrere Punkte und stellt daher kein perfektes Ergebnis dar und nur stufenförmige Schatten(siehe ). Die Punkte, durch die die Lichtquelle repräsentiert wird, können dabei systematisch(siehe Figure 5, Figure 6 und Figure 7), zufällig(siehe Figure 9) oder gemischt(siehe) gewählt werden. Für den finalen Render wurden jedoch nur systematische Shadow Rays benutzt, da diese für ein schöneres Bild sorgen und ab einem gewissen Grad der Weiterentwicklung Random Shadow Rays verbuggt waren (siehe Figure 9).

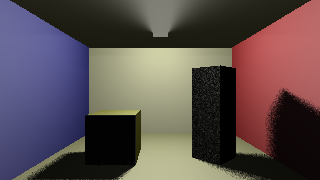
Figure 5: Bild mit 1 systematischen Shadow Ray

Figure 6: Bild mit 4 systematischen Shadow Rays

Figure 7: Bild mit 9 systematischen Shadow Rays

### kjkhj

Figure 8: Bild mit 9 systematischen und 8 zufälligen Shadow Rays

Figure 9: Bild mit 8 zufälligen Shadow Rays

### Berechnung der Beleuchtung

Bei der Berechnung der Beleuchtung für eine quadratische Lichtquelle nach dem Phong-Modell ist nur der „Diffuse“- und „Specular“-Wert wichtig, da nur diese von der Position des Lichts abhängen. Für die Berechnung dieser Werte wird daher der für den Punkt auf dem Objekt der optimale Punkt auf der Lichtfläche zur Berechnung verwendet.

Bei der Berechnung des Specular-Wertes wird anhand der realen Reflektionsachse (dem Normalenvektor der Oberfläche) ein reflektierter Ray von der Kamera aus gebildet. Wie weit man diesem Ray folgen muss um auf der Lichtebene zu landen wird mit folgender Formel berechnet:

Wenn der Ray parallel zur der Lichtebene ist () wird zu der Richtung des Rays ein kleiner Offset hinzugefügt um den Schnittpunkt nahe Unendlich zu legen. Der herausgefundene Schnittpunkt wird anschließend durch eine Basisänderung so vom dreidimensionalen in den zweidimensionalen projiziert, dass ein Wert im Bereich 0<=x, y<=1 einem Punkt auf der Lichtebene innerhalb des Lichts entspricht (Für die Transformation wird die Pseudoinverse einer Matrix, die aus den aufspannenden Vektoren des Lichts besteht, benutzt). Ist n positiv wird der zu dem Schnittpunkte nächste Wert im Bereich 0<=x, y<=1 rücktransformiert, wenn n negativ ist, der der am weitesten entfernteste. Dieser rücktransformierte, nun wieder im dreidimensionalen Raum liegende Punkt wird dann für die Berechnung der optimalen Lichtachse verwendet.

Die Berechnung für die optimale Position des Lichts für den diffusen Wert verläuft ähnlich, nur wird hier als initialer Ray nicht der reflektierte Vektor von der Kamera verwendet, sondern der Normalenvektor der Oberfläche.

//TODO: ganz viele skizzen einfügen

# Implementierung

## Funktionen

## Klassenmodell

Quellenverzeichnis

**Im aktuellen Dokument sind keine Quellen vorhanden.**