

# Bildsynthese - Erklärung zu Aufgabenblatt 2

Moritz Hamann - 2508568  
Boitumelo Ruf - 2835202

---

## Aufgabe 2.1

Siehe Quelltext.

## Aufgabe 2.2

Den Aliasingeffekten kann man durch sogenanntes Supersampling entgegengewirkt werden. Hierbei werden mehrere Strahlen durch ein Pixel geschickt und diese anschließend gewichtend aufaddiert. Diese Gewichte können unterschiedlich gesetzt werden. Wir haben hierzu einen Kauskern verwendet. Die Variable “stepsize” gibt an mit welcher Schrittweite bzw. mit welchem Abstand der einzelne Pixel abgetastet wird. Diese ist dabei so gewählt, dass es genauso viele Samples pro Pixel, wie es Werte Gaußkern gibt.

Die Qualität des Anti-Aliasing kann durch die Menge der Abtastwerte verwendet werden. Je mehr Werte desto besser werden die Artefakte aus dem Bild entfernt. Wir haben drei Voreinstellungen für die Größe des Gaußkerns implementiert, die einzeln ein-, bzw. auskommentiert werden können.

## Aufgabe 2.3

Um eine Blende (und die daraus entstehende Schärfentiefe) zu simulieren, muss der Strahlengang durch eine fiktive dünne Linse berechnet werden. Dazu müssen als erstes ein paar Konstanten definiert werden. Die Brennweite der Linse *focal\_length*, sowie der Abstand der Linse zum virtuellen Sensor der Kamera *f*. Dadurch kann mit Hilfe der Formel für dünne Linsen:

$$\frac{1}{f} + \frac{1}{S_2} = \frac{1}{focal\_length}$$

der Abstand  $S_2$  der Ebene  $E_2$  berechnet werden, in der sich alle Strahlen - ausgehend vom gleichen Punkt auf dem Sensor - durch die dünne Linse wieder schneiden. Anschliessend kann man mit Hilfe der Blendenzahl *dof*, sowie dem Abstand *f* von Sensor und Blende, die Öffnung *D* der Blende berechnet werden:

$$D = \frac{f}{dof}$$

Weiterhin benötigt man den Schnittpunkt *R* auf der Ebene  $E_2$  auf der sich alle Strahlen schneiden. Diesen berechnet man indem man den Strahl durch

den Mittelpunkt der Linse (über *getRayPerspective* im Code) mit  $E_2$  schneiden lässt. Nun wählt man kreisförmig Punkte auf Linse auf (deren Abstand zu  $g\_eye < D$  ist) und addiert die Radiance der Strahlen von diesen Punkten durch  $R$  auf. Nach abschließender Normierung durch die Anzahl der Strahlen erhält man den Farbwert für diesen Pixel.