
Human-Computer Interaction

Bearbeitung zu *Interaktive Computergrafik*, WiSe 2015/16

Betreuer: Eike Langbehn, Susanne Schmidt, Dennis Krupke

Übung 3

Hinweis: Das Übungsblatt ist bis Dienstag, den 17.11.15, um 12 Uhr zu bearbeiten.

Aufgabe 1 (Gruppenaufgabe, 4 Punkte)

Ziel dieser Übung ist die Entwicklung eines eigenen Fragment-Shaders. Eine entsprechende Einführung in die Shadersprache GLSL ES wird in der Übung am 12.11. gegeben.

Der Fragment-Shader soll die Helligkeit aller Pixel ändern, die innerhalb eines Rings mit einem bestimmten Radius und einer festgelegten Breite liegen. Hierzu sollen die R-, G- und B-Komponenten aller betroffenen Pixel mit einem Faktor $k > 1$ multipliziert werden werden.

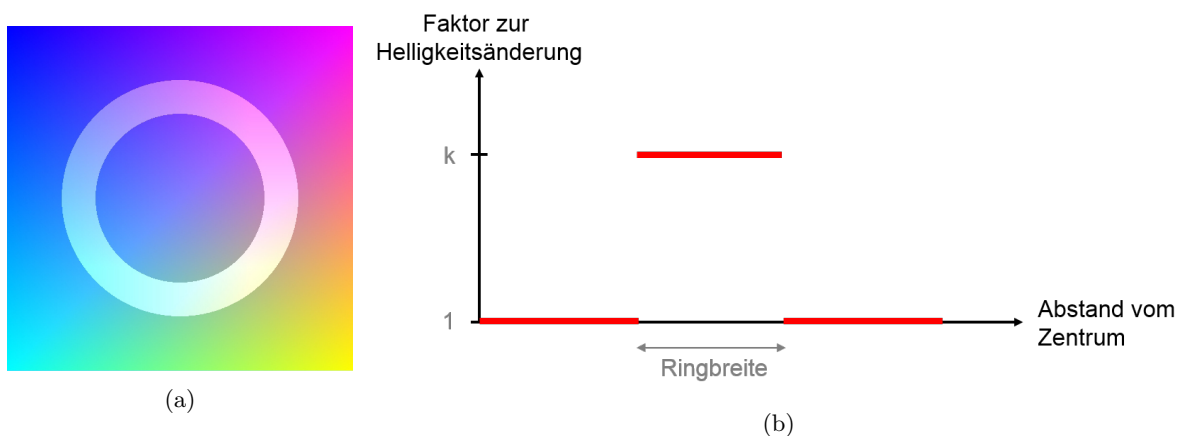


Abbildung 1: Illustration des Fragment-Shaders: (a) Ergebnis in WebGL mit konstanter Helligkeitsänderung wie in (b) dargestellt

Aufgabe 2 (Gruppenaufgabe, 6 Punkte)

Die zweite Aufgabe besteht darin, den soeben programmierten Shader interaktiv zu gestalten. Erweitern Sie hierzu zunächst Ihre JavaScript-Datei um eine Funktion, die Nutzereingaben entgegennimmt und verarbeitet. Übergeben Sie anschließend Ringhelligkeit und -durchmesser an den Fragment-Shader, der die Werte in die entsprechenden Fragmentberechnungen einbeziehen muss.

Folgendes Verhalten soll erreicht werden:

- ↑: Helligkeit des Rings stufenweise erhöhen bis ein Maximalwert erreicht ist.
- ↓: Helligkeit des Rings stufenweise verringern bis der Ring nicht mehr sichtbar ist.
- : Durchmesser des Rings vergrößern. Dabei soll der Ring nie komplett den Canvas verlassen.
- ←: Durchmesser des Rings verringern. Der Ring sollte auch beim Minimalwert nicht verschwinden.

Zusatzaufgabe

Passen Sie den Shader aus Aufgabe 1 so an, dass die Stärke der Helligkeitsänderung dem positiven Teil einer Sinuswelle entspricht. Der Verlauf der Helligkeitsänderung und das gewünschte Ergebnis sind in Abbildung 2 dargestellt.

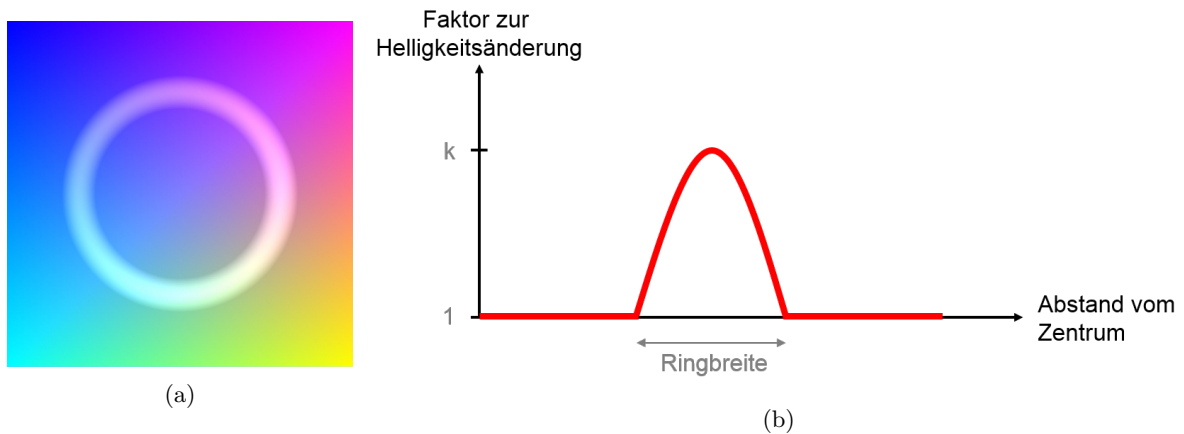


Abbildung 2: Illustration des Fragment-Shaders: (a) Ergebnis in WebGL mit sinusförmiger Helligkeitsänderung wie in (b) dargestellt