

Übungen zu Computergrafik

Sommersemester 2016

Blatt 4

Aufgabe 4.1: Farb-Konvertierung (30 Punkte)

In dieser Aufgabe sollen Sie eine Konvertierung zwischen RGB und CMY sowie RGB und HSV implementieren. In der gegebenen Anwendung sind die zu bearbeitenden Stellen mit Kommentaren markiert. Sie müssen also nur `CmyColor.java` und `HsvColor.java` anpassen.

Die gegebene Applikation besitzt eine Oberfläche zur Auswahl von Farben in unterschiedlichen Kodierungen. Sobald Sie alle vier Methoden implementiert haben, verändern sich alle Slider entsprechend, wenn Sie einen bestimmten Slider verändern. Außerdem wird die Farbe mit Hilfe von drei Kreisen angezeigt. Die eingestellte Farbe ist dort zu sehen, wo sich alle drei Kreise überschneiden.

Aufgabe 4.2: Median Cut (45 Punkte)

In dieser Aufgabe sollen Sie den MedianCut-Algorithmus implementieren. Ihre Implementation wird genutzt, um ein Bild mit den so reduzierten Farben anzuzeigen.

Die bereitgestellte Applikation kann Bilder verschiedener Formate laden und darstellen. Durch Klick auf den Button *Reduzierte Version anzeigen* wird der (noch nicht implementierte) MedianCut-Algorithmus ausgeführt und das Bild mit den Ersatzfarben angezeigt. Neben dem Laden von Bildern über den *Datei auswählen*-Button wird beim Starten der Applikation versucht, die Datei `totoro.png` aus dem Arbeitsverzeichnis zu laden. Dies ist z. B. das Eclipse-Projektverzeichnis bzw. das Verzeichnis, in dem `java` ausgeführt wird. Diese Funktionalität und die ebenfalls bereitgestellte `totoro.png` können zum schnellen Testen während der Entwicklung genutzt werden.

Der MedianCut-Algorithmus soll hauptsächlich in `ColorTable.java` implementiert werden. Hier sind zwei wichtige Methoden markiert: Im Konstruktor soll das gegebene Bild analysiert, also der eigentliche Algorithmus ausgeführt, werden. Die Methode `getReducedColor` wird für jeden Pixel im Ursprungsbild einmal aufgerufen, um die reduzierte Farbe aus der Farbtabelle zu erhalten. Dort müssen sie dann lediglich auf die Daten, die im Konstruktor erzeugt wurden, zugreifen.

Sie können natürlich gerne weitere Klassen hinzufügen, um eventuell Code auszulagern. Eine Klasse zur Darstellung einer Box im RGB-Raum könnte beispielsweise nützlich sein. Sie müssen bei der Implementation nicht besonders auf Performance achten – selbst eine recht naive Implementierung sollte bei 16 Farben und dem gegebenen Testbild in wenigen Sekunden terminieren.

Diese Aufgabe ist bewusst etwas komplexer als andere Aufgaben dieser Veranstaltung und sollte daher inkrementell, also Schritt für Schritt, angegangen werden.

Aufgabe 4.3: Fragen (25 Punkte)

Beantworten Sie Ihrem Tutor Fragen zu den Inhalten der Vorlesung. Machen Sie sich hierzu insbesondere mit den Themen *Farbmodelle*, *Pixeldateien* und *Grafikkomppressionsmethoden* vertraut.