

## Übungszettel 5 zur Vorlesung BV 1 im WS 2007/2008

Ausgabe am: 07.01.2008

Abgabe am: 21.01.2008, 15 Uhr s.t. (!)

### Konturrekonstruktion und Segmentierung

In der ersten Aufgabe geht es um eine Segmentierung mit Hilfe von Startpunkten in homogenen Umgebungen und einem Region-Growing Verfahren. Als Region-Growing-Verfahren soll ein adaptierter Grassfire-Algorithmus genutzt werden. Bisher haben wir den Algorithmus ja nur für Binärbilder kennen gelernt. Die zweite Aufgabe befasst sich mit der Hough-Transformation. Mit Hilfe dieser Funktion kann man unvollständige Konturen rekonstruieren.

#### Aufgabe 1: Segmentierung

10 Punkte

Ein Region-Growing-Verfahren fasst benachbarte Punkte von ähnlicher Intensität zu Regionen zusammen, ausgehend von einem Startpunkt. Da benachbarte Punkte oft nicht exakt denselben Grauwert haben, arbeitet das Verfahren mit einem Schwellwert  $t$ , den ihr als Parameter der Funktion realisieren sollt. Ein Punkt am Rande einer Region wird zu dieser Region hinzugefügt, wenn die Differenz seines Grauwertes zum mittleren Grauwert der Region innerhalb des Schwellwertes  $t$  liegt.

Um mit dem Region-Growing-Verfahren beginnen zu können, müssen zunächst geeignete Startpunkte bestimmt werden. Ausgehend von den Startpunkten soll mittels Grassfire die jeweilige Region Schritt für Schritt erweitert werden, bis das Abbruchkriterium erfüllt ist. Aktualisiert den mittleren Grauwert einer Region jeweils nach einem Iterationsschritt.

Die Startpunkte sollen dabei in homogenen Umgebungen liegen. Ein Punkt liegt in einer homogenen Umgebung, wenn sein Grauwert dem mittleren Grauwert seiner Umgebung ähnelt. Benutzt als Umgebung eine  $5 \times 5$  Nachbarschaft. Um die Ähnlichkeit zu bewerten, soll ebenfalls der Schwellwert  $t$  benutzt werden.

Eure Funktion soll zwei Bilder und Text ausgeben. Das erste Bild zeigt die segmentierten Regionen in unterschiedlichen Intensitätswerten kodiert. Benutzt dazu den mittleren Grauwert der segmentierten Region. Das zweite Bild ist ein Binärbild, bei dem alle segmentierten Punkte gesetzt sind. Gebt die Startpunkte des Region Growings als Text aus. Dazu lässt sich das `StringObject` als Ausgabeobjekt benutzen oder — falls ihr nicht Vista einsetzt — auch der `std::ostream log` des Operators für eine Ausgabe in der Konsole der GUI.

Überlegt euch, was für Auswirkungen eine Änderung des Parameters hat und probiert verschiedene Belegungen aus (z.B. 3, 5, 50, ...). Welche Auswirkungen

hat es, wenn er zu klein oder zu groß gewählt wird? Warum ist es sinnvoll, Startpunkte in homogenen Umgebungen zu wählen? Warum werden die Punkte mit dem Mittelwert der Region verglichen (und nicht z.B. mit ihren direkten Nachbarn)?

Anzuwenden ist die Funktion auf das Bild `Grauflächen.png`.

Abzugeben sind die Sourcen, DLL und schriftliche Beantwortung der Fragen.

### **Aufgabe 2:** Die Hough-Transformation

10 Punkte

Bei der Hough-Transformation werden zunächst die gesetzten Pixel im Eingabebild gesucht und die möglichen Geraden basierend auf der Hesseschen Normalform in einen „Akkumulator“ geschrieben. Anschließend wird im Akkumulator nach Einträgen größer einem Schwellwert  $t$  gesucht und die entsprechenden Geraden werden in das Ergebnisbild eingezeichnet. Der Schwellwert  $t$  soll als Parameter Eurer Funktion implementiert werden. Um den Aufwand in Grenzen zu halten, sollen nur die Richtungen  $0.0^\circ$ ,  $26.565^\circ$ ,  $45.0^\circ$ ,  $63.435^\circ$ ,  $90.0^\circ$ ,  $116.565^\circ$ ,  $135.0^\circ$  und  $153.435^\circ$  untersucht werden. Es sind also folgende Schritte auszuführen:

- Suche gesetzte Pixel im Eingabebild
- Berechne für diese Pixel die Abstände gemäß Hough-Transformation für die acht Geraden
- Schreibe die Ergebnisse in den Akkumulator
- Suche im Akkumulator nach Einträgen größer oder gleich  $t$
- Trage die entsprechenden Geraden in das Ergebnisbild ein

Die Geraden werden zunächst durch das ganze Bild gezogen. In einem letzten Schritt sollen sie so abgeschnitten werden, dass die mit der Hough-Transformation entsprechend vervollständigten Geradenstücke übrig bleiben. Bitte dokumentiert und begründet eure Vorgehensweise beim Abschneiden der Geraden im Sourcecode und in der schriftlichen Abgabe.

Als Testbild eignet sich das Bild `EinPixel.png`, bei dem im Ausgabebild ( $t = 1$ ) alle acht Geraden durch den Punkt  $(41, 41)$  laufen müssen. Anzuwenden ist die fertige Funktion auf das Bild `KonturSeg.png`.

Aus welchem Grund ist die Hough-Transformation wichtig? Denkt dabei daran, wie die Eingabebilder üblicherweise entstehen.

Abzugeben sind die Sourcen, DLL und schriftliche Beantwortung der Fragen.

Gesamtpunktezahl: 20