

Laboration 2

Grafisk teknik (TNM059)

Digital Rastring

S. Gooran (VT2007)

Introduktion

Denna laboration handlar om rastrening och är tänkt att fungera som komplement till rastrening-föreläsningar och lektioner. Efter denna laboration kommer ni att förstå principerna bakom olika rastreningstekniker och deras skillnader. Ni kommer att närmare bekanta er med tröskelrastrening, tabellrastrening, error diffusion (felspridning) samt en iterativ rastreningsteknik genom att använda dessa tekniker för att rastra minst en svart-vit bild. Ni kommer även att rastra en färgbild med minst två rastreningstekniker.

Den här laborationen måste förberedas innan laborationstillfället. Ett antal förberedelseuppgifter finns på kursens hemsida som måste lösas innan laborationstillfället. Laborationsassistenten går igenom era lösningar och de som inte har gjort uppgifterna får inte genomföra laborationen. Nödvändiga teorier för att kunna genomföra laborationen går igenom under kursens föreläsningar och lektioner som handlar om rastrening. Häftet "Digital Halftoning" och det kompletterande häftet "Rastrening, övningar" måste läsas noggrant innan labben.

Laborationsuppgifter

Uppgifter i denna laboration görs i programmet Matlab. De flesta Matlabfunktioner som används i denna laboration har redan gåtts igenom under en tidigare lektion och laboration.

Ni kan också få information om funktioner i Matlab genom att skriva *help* och sedan funktionens namn i matlabprompten.

Alla bilder och funktioner som ni kommer att behöva finns under S:/TN/M/TNM059/Lab2.

Börja med att läsa in bilden *kvarn.tif* i Matlab m. h. a. funktionen *imread*. Alla uppgifter, förutom sista uppgiften, görs på denna bild men ni kan gärna använda andra bilder som er testbild. I dessa uppgifter antar vi att bilden är normerad så att dess pixelvärden ligger mellan 0 och 1. Så, normera bilden efter att ni har läst in den. (Hur?)

1 Tröskelrastrening

1.1

Börja med att rastra bilden genom att tröskla den med en fix tröskel, t.ex. 0.5. Hur många grånivåer representerar den? Blir resultatet snyggt?

.....
.....

1.2

Ladda ner filen *troskelmatriser.mat*, se funktionen *load*. Nu har du fått tre tröskelmatriser, nämligen *tr1*, *tr2* och *tr3*. Ta först en titt på matriserna *tr1* och *tr2*!

Hur många grånivåer representerar vardera av dessa två tröskelmatriser?

.....

Rastrea bilden med de här två tröskelmatriserna genom att använda funktionen *troskel*. Glöm inte att matriserna skall normeras så att alla tröskelvärdena ligger mellan 0 och 1. (Hur?) Vad ser du för skillnad mellan resultatbilderna? Förklara!

.....

.....
.....
.....
.....
.....

1.3

Ta en titt på tröskelmatriken *tr3*! Hur många grånivåer representerar denna matris?

.....

Jämför denna matris med *tr2*. Ser du någon skillnad? Vad? Kan du förutse skillnaden mellan resultaten av att rastera bilden med dessa två matriser?

.....
.....
.....
.....

Rastera bilden med denna tröskelmatrix (Glöm inte att normera tröskelmatriken). Förklara skillnaden mellan den här rasterade bilden och den från uppgift 1.2. Ser resultaten ut som du hade förväntat dig?

.....
.....
.....
.....

1.4

Tröskla bilden med dina egna tröskelmatrixer. T. ex. linjeraster (både horisontell och vertikal), spiralraster, och även en tröskelmatrix av slumpal.

.....
.....
.....
.....

2 Tabellrastering

2.1

Använd funktionen *tabellrast* för att tabellrastera bilden. Det är en mycket enkel funktion som använder medelvärdet (summan) av varje 3 x 3 omgivning i inbilden som en index till ett rasteralfabet. Hur många grånivåer kan dessa representera? Jämför resultatet med resultat från föregående uppgifter.

.....
.....

.....
.....
.....
.....

2.2

Använd andra uppsättningar av punkter för varje 3 x 3 region i utbilden genom att ändra funktionen *tabellrast*.

Hur kan olika uppsättningar påverka resultatet?

.....
.....
.....
.....

Finns det en chans att kunna representera flera grånivåer med dessa 3 x 3 tabeller? Om nej, varför? Om ja, hur?

.....
.....
.....

3 Rastrering med felspridning (Error diffusion)

Rastrera bilden med hjälp av funktionen *errordif*.

3.1

Använd filtret som Floyd och Steinberg har föreslagit. Detta filter finns skrivet i *help* kommandot för *errordif*. Jämför nu resultatet med tidigare resultat.

.....
.....
.....
.....
.....

3.2

Använd följande filter som ditt felfilter och rastrera bilden. Detta filter presenterades av Jarvis, Judice och Ninke 1976.

$$\begin{bmatrix} & \bullet & 7 & 5 \\ 3 & 5 & 7 & 5 & 3 \\ 1 & 3 & 5 & 3 & 1 \end{bmatrix} (*1/48)$$

Begrunda skillnaderna mellan denna bild och bilden från tidigare uppgift.

.....

.....
.....
.....
.....

3.3

Använd följande filter:

$$\begin{bmatrix} & & \bullet & 0.841 & -0.171 \\ 0.247 & 0.459 & 0.344 & 0.071 & -0.164 \\ -0.415 & -0.218 & -0.172 & -0.145 & 0.001 \\ 0.137 & 0.101 & 0.008 & -0.028 & 0.104 \end{bmatrix}$$

.....
.....
.....

3.4

Rastrera bilden med minst två filter som du själv konstruerar. Använd filter med en vikt respektive två vikter.

.....
.....
.....

3.5

Lägg till lite brus på inbilden innan den rastreras med felspridningsmetoden. För att generera brus kan ni använda funktionen *rand*, men observera att **bruset måste ha medelvärdet noll**. Eftersom *rand* genererar slumpstal mellan noll och ett och använder likformigt spridning blir medelvärdet 0.5.

Använd första felfiltret, dvs. det presenterat av Floyd och Steinberg, uppgift 3.1, och jämför resultaten före och efter tillägningen av **lämplig mängd** av brus.

.....
.....
.....
.....
.....
.....

4 Iterativ rastrering

Använd funktionen *nearoptimal* för att rastrera bilden med near-optimal metoden. Near-optimal metoden går kortfattat igenom under en föreläsning. Observera att det kan ta ett par - tre minuter innan resultatet är klart! Jämför resultatet med de från föregående uppgifter.

.....
.....
.....
.....

5 Färgrastrering

En färgbild rasteras normalt genom att rastera alla dess färgkanaler var för sig. Skriv en Matlabfunktion som läser in en färgbild och rasterar bildens alla kanaler oberoende och med en och samma rasteringsmetod. Resultaten ska sedan läggas ihop för att få den slutgiltiga rasterade färgbilden.

Använd din funktion för att rastera bilden *musicians.tif* med felspridning samt near-optimal.

.....
.....
.....
.....