UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i :	INF2310 — Digital bildebehandling
Eksamensdag :	Tirsdag 5. juni 2007
Tid for eksamen :	09:00 – 12:00
Oppgavesettet er på :	**5** sider
Vedlegg :	Ingen
Tillatte hjelpemidler :	Ingen, heller ikke kalkulator.

- Les gjennom hele oppgaven før du begynner å løse oppgaven. Kontroller at oppgavesettet er komplett før du begynner å besvare det. Dersom du savner opplysninger i oppgaven, kan du selv legge dine egne forutsetninger til grunn og gjøre rimelige antagelser, så lenge de ikke bryter med oppgavens "ånd". Gjør i såfall rede for forutsetningene og antagelsene du gjør.

- Det er 8 oppgaver.

- Alle svar skal begrunnes.
  Det er ikke tilstrekkelig å gi svar som ”Ja”, ”Nei”, eller bare en tallverdi.

- Merk at alle delspørsmål teller like mye.

- Det lønner seg å disponere tiden slik at man får besvart alle oppgavene. Hvis du står fast på enkeltoppgaver, gå videre slik at du i alle fall får gitt et kort svar på alle oppgaver.

1. **Sampling og kvantisering**

a. Hva sier Nyquist og Shannons samplingsteorem?

Svar: *Det kontinuerlige bildet kan rekonstrueres fra det digitale bildet dersom samplingsraten $f_s = 1/T_s$ er større enn $2f_{max}$, der f_{max} er den høyeste frekvensen i det kontinuerlige bildet. (Formulering fra forelesningsnotatet)*

b. Anta at vi har et avbildningssystem som gir en punktspredningsfunksjon med bredde 0.25mm. Altså vil det kunne skille punkter som har avstand 0.25 mm mellom seg. Hva er den minste samplingsraten (**frekvensen**) vi må benytte ifølge samplingsteoremet? Vær presis med benevningen.

Svar: *Samplingsteoremet krever $T_s <$ ½T, der T=0.25mm, følgelig må $f_s >$ 2*1/T = 2*1/0.25 = $8mm^{-1}$.*

c. Forklar prinsipielle problemer med rekkefølgen i dette avbildnings- og bildeanalysesystemet:

Avbildning => Sampling => Analyse av romlig oppløsning fra samplede bilde => Anti-aliasing => Videre strukturanalyse av bildet

Svar: Anti-aliasing kan ikke gjøres etter sampling, og deteksjon av romlig oppløsning med frekvens over Nyquist kan ikke gjøres i det samplede bildet (uten tilleggskunnskap, i hvert fall).

2. **Histogrambaserte operasjoner**

Gråtone-transformfunksjonen som gir et histogramutjevnet bilde er i det kontinuerlige tilfellet gitt generelt ved

$$T(i) = G\int_0^i p(x)\,dx$$

der *p* er bildets normaliserte histogram, og *G* er maksimal gråtoneverdi i bildet.

a. Anta at vi har et histogram som kan tilnærmes med den kontinuerlige funksjonen

$$f(i) = -\frac{i}{2} + 1, \qquad 0 \le i \le 2$$

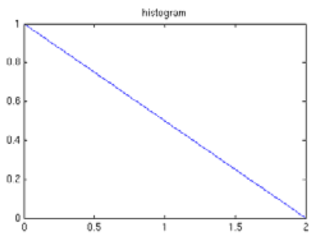
Hva er gråtone-transformfunksjonen, T(i), som vil histogramutjevne et bilde med *f* som histogram?

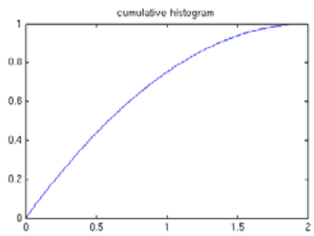
Svar: Histogrammet, *f*, har areal lik 1 og er allerede normalisert. Derfor kan vi plugge rett inn i T, med G=2 og få

$$T(i) = 2\int_0^i f(x)dx = 2(\frac{-i^2}{4} + i) = \frac{-i^2}{2} + 2i$$

b. Skissér funksjonene *f* og T, samt det resulterende histogrammet til bildet etter transformen T.

Svar:





f til venstre. T vil være figuren til høyre multiplisert med 2. Etter transformen får vi et flatt histogram. Dette siste viser om man har forstått hva målet med transformen er.

3. Gråtonetransformer

Anta gråtonetransformen T[i] = ai + b, der a og b er reelle tall.

a. Hvilke effekter har parametrene a og b på kontrasten og "lysheten" i det resulterende bildet?

Svar: a regulerer kontrasten, og b "lysheten". a>1 øker kontrasten, a<1 minker kontrasten. b>0 øker lysheten i bildet, b<0 minker lysheten (om antar at stor pikselverdi gir lys piksel). Pirk: Ved positive pikselintensiteter vil a>1 også øke middelverdien (, og omvendt ved a<1).

b. Man kan gi resultatbildet ønsket middelverdi og varians ved å benytte slike lineære transformer med bestemte a og b. Hvorfor vil man ofte standardisere bildeserier ved å gi de samme varians og middelverdi?

Svar: Ved mål om å sammenligne struktur i bildene vil man følgelig begrense effektene av varierende avbildningsforhold. (Belysning, detektor-aldring, linse-støv, etc.)

c. Hvis den generelle "formen" på histogrammet ikke skal gå tapt, ville man da standardisert en billedserie ved bruk av histogramutjevning eller ved en lineær transform som beskrevet i innledningen til oppgaven?

Svar: Den lineære transformen endrer ikke på "formen" på histogrammet, den kun flytter og "strekker" histogrammet.

4. **Filtrering**

Gitt følgende 3 x 3 utsnitt av et 3 bits gråtonebilde med pikselverdier

3	1	7
2	5	7
1	7	3

Hvilken verdi vil senterpikslet her få hvis vi bruker:

a. Et 3x3 ideelt lavpassfilter?

Svar: (3+1+7+2+5+7+1+7+3)/9 = 36/9 = 4.

b. Et 3x3 kvadratisk medianfilter?

Svar: Sorterer : {1,1,2,3,3,5,7,7,7}. I midten finner vi medianen, M=3.

c. Et 3x3 pluss-formet medianfilter?

Svar: Sorterer og finner {1,2,5,7,7}; Midt i listen finner vi medianen, M=5.

For å få full uttelling kan du ikke bare gi et svar i form av en pikselverdi. Du må også gi en kort begrunnelse, eller en utregning.

5. **Median-filtrering av binære bilder**

Vi har tersklet et gråtonebilde, og vil bruke et kvadratisk medianfilter til å fjerne små objekter i det binære bildet, eventuelt ved å filtrere flere ganger. Vi antar at objektene ligger så langt fra hverandre at vi kan betrakte ett objekt av gangen.

a. Hvor mange ganger må vi filtrere med et kvadratisk 3x3 medianfilter for å fjerne objekter som er mindre eller lik 2x2? Gi en kort begrunnelse.

Svar: Alle rektangulære objekter vil miste alle sine fire hjørner ved første filtrering. Altså blir alle objekter som er 2x2 eller mindre borte ved en filtrering.

b. Hvor mange ganger må vi filtrere med et kvadratisk 3x3 medianfilter for å fjerne kvadratiske 3x3 objekteri det binære bildet? Gi en kort begrunnelse.

Svar: De fire hjørnene forsvinner ved første filtrering, pikslene midt på langsidene ved andre filtrering, og det siste pikslet ved tredje filtrering.

c. Hva skjer ved gjentatt filtrering av et binært 4x4 objekt med et 3x3 kvadratisk median-filter? Gi en kort beskrivelse.

Svar: De fire hjørnene forsvinner ved første filtrering. Ved neste filtrering vil det alltid være 5 eller 8 objekt-piksler innenfor filteret, slik at ingen pikselverdier endres.

6. Segmentering ved terskling

Anta at du har et digitalt gråtonebilde av størrelse 8 x 8 med G = 8 gråtoner.

- Det normaliserte kumulative histogrammet til bakgrunnspikslene stiger lineært fra 1/8 ved g = 0 til 1 ved g = 7.
- Det normaliserte kumulative histogrammet til forgrunnspikslene stiger lineært fra 0 ved g = 1 til 1 ved g = 3.
- Forgrunnspikslene har a priori sannsynlighet = 1/4.

a. Tegn en skisse av forgrunns- og bakgrunnshistogrammene, og gi en kort forklaring på hvordan du kommer fram til dem.

Svar: Siden det kumulative histogrammet til forgrunnspikslene stiger lineært må det være like mange piksler av hver gråtone. Det er 2 gråtoner i forgrunnen (3 - 1 = 2), og tilsammen 8x8/4 = 16 forgrunnspiksler, altså 8 piksler av hver av de to gråtonene.
Det kumulative histogrammet til bakgrunnspikslene stiger også lineært, så det må være like mange piksler av hver gråtone. Det er 8 gråtoner i forgrunnen, og tilsammen 3x8x8/4 bakgrunnspiksler, altså 6 piksler av hver gråtone.

b. Tegn en skisse av de normaliserte forgrunns- og bakgrunns-histogrammene skalert med a priori sannsynlighet, og gi en kort forklaring på hvordan du kommer fram til dem.

c.

Svar: Det er like mange av de 2 gråtonene i forgrunnen. I et normalisert histogram (sum = 1) får vi verdien 1/2 for hver gråtone.
Siden a priori sannsynligheten for forgrunn er ¼, får vi verdien 1/8 for hver av de 2 gråtoneverdiene etter skalering med a priori sannsynlighet.
Det er like mange piksler av hver av de 8 gråtonene i bakgrunnen. Det normaliserte bakgrunnshistogrammet blir da flatt, med verdi 1/8.
Skalert med (1-1/4) =3/4 får vi 3/32 for de 8 gråtonene fra 0 til 7.

d. Indiker hvor tersklen(e) må settes for å få minst mulig feilterskling, gi et uttrykk for hvordan du vil terskle (vær presis med terskelverdi og ulikhetstegn), og angi hvor stor del av for- og bakgrunnspikslene som nå blir feilsegmentert.

Svar: Vi får to skjæringer mellom de skalerte, normaliserte histogrammene, og må derfor ha to terskler, nemlig ved
T1 = 1 og T2 = 3

Uttrykket for tersklingen blir da

$$g(x,y) = \begin{cases} 0 \text{ hvis } & f(x,y) \le T_1 \\ 1 \text{ hvis } & T_1 < f(x,y) \le T_2 \\ 0 \text{ hvis } & f(x,y) > T_2 \end{cases}$$

Bakgrunnspikslene fra og med g = 2 til og med g = 3 blir feilsegmentert. Det er 1/4 av bakgrunnspikslene.

Ingen forgrunnspiksler blir feilsegmentert.

Siden bakgrunnspikslene utgjør ¾ av bildet, blir den totale feilsegmenteringen 3/16, eller 18.75 %, men det har vi ikke spurt om.

7. Run-length transform og Huffman-koding uten kalkulator

Vi har følgende 2 bits gråtonebilde med 8 x 8 piksler:

2	**3**	**3**	**3**	**3**	**3**	**3**	**3**
2	**2**	**3**	**3**	**3**	**3**	**3**	**1**
2	**2**	**2**	**3**	**3**	**3**	**1**	**1**
2	**2**	**2**	**2**	**3**	**1**	**1**	**1**
2	**2**	**2**	**0**	**1**	**1**	**1**	**1**
2	**2**	**0**	**0**	**0**	**1**	**1**	**1**
2	**0**	**0**	**0**	**0**	**0**	**1**	**1**
0	**0**	**0**	**0**	**0**	**0**	**0**	**1**

a. Forklar hvorfor en naturlig binærkoding med like lange kodeord for hver pikselverdi er optimal for dette bildet.

Svar: Sannsynlighetene er like for alle de fire gråtonene, p=1/4.
Dermed er entropien H= 4 *(1/4)*log2(4) = 4*(1/4)*2 = 2.
En naturlig binærkode med 2 bit per piksel gir altså det samme antall bit per piksel som entropien til bildet, som angir det teoretisk laveste gjennomsnittlige antall bit per piksel ved koding av bildet.

b. Gjør en løpelengdetransform linje for linje i bildet, slik at vi får en kontinuerlig sekvens av enkelt-siffer, vis det samlede histogrammet for løpelengder og gråtoner.

Svar: Det er 22 løpelengder i bildet. Siden gråtonene ligger mellom 0 og 3 og løpelengdene ikke kan bli lengere enn 8 piksler, er både gråtoner og løpelengder en-sifret. Løpelengdetransformen gir derfor en kontinuerlig sekvens på 44 enkelt-siffer (blanke er satt inn her for å skille tallparene):

21 37 22 35 11 23 33 12 24 31 13 23 01 14 22 03 13 21 05 12 07 11

Dette gir følgende histogram:

0	4
1	13
2	11
3	10
4	2
5	2
7	2

c. Finn kodeboken for en Huffman-koding av resultatet av løpelengdetransformen, og beregn det totale antall bit som trengs for å kode bildet med denne teknikken.

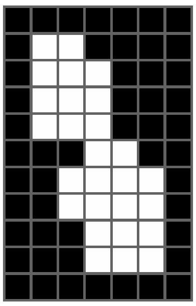
Svar: Kodeboken kan bli som vist nedenfor:

1	13	00
2	11	01
3	10	10
0	4	111
7	2	1101
5	2	11000
4	2	11001

Og dette gir 34*2 biter + 4*3 biter + 2*4 biter +4*5 biter = 108 biter.

8. Binær morfologi

La hvitt være 1 og svart være 0 i bildet under. Anta at vi benytter et 3x3 kvadratisk strukturelement med origo i midten.

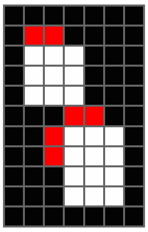


a. Hva vil slutt-resultatet være ved gjentatte morfologiske erosjoner?

Svar: All struktur i bildet forsvinner. Kun bestående av 0-ere.

b. Utfør og vis resultatet av en morfologisk åpning. (Morfologisk åpning er morfologisk erosjon etterfulgt av morfologisk dilasjon.)

Svar: Piksler merket med rødt vil forsvinne.



(Figuren over er en del av løsningsforslaget)

c. Hvis man etter en segmentering satt igjen med et binært bilde hvor objektene hadde litt rufsete kanter og små, uønskede hull spredt rundt omkring, hvilke morfologiske operasjoner ville du da benyttet for å ”rengjøre” bildet?

Svar: Morfologisk lukking, siden det vil fylle hullene.

Takk for oppmerksomheten !