Utkast med løsningshint inkludert

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i : INF2310 — Digital bildebehandling

Eksamensdag : Onsdag 2. juni 2010

Tid for eksamen : 09:00 – 12:00

Oppgavesettet er på : XXX sider

Vedlegg : Ingen

Tillatte hjelpemidler : Ingen

- 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.

- Merk at alle delspørsmål teller like mye. Det er tilsammen 20 delspørsmål. Det lønner seg derfor å disponere tiden slik at man får besvart alle oppgavene. Hvis dere står fast på enkeltoppgaver, gå videre slik at dere får gitt et kort svar på alle oppgaver.

- **Alle svar skal begrunnes.** Gjør rede for bruken av eventuelle teoremer, prinsipper eller forutsetninger slik at en tredjeperson kan følge dine resonnementer.

Sampling og geometriske transformer

Oppgave 1

Anta at vi har et (kontinuerlig) båndbegrenset bilde med en høyeste frekvens $f_{max}=5mm^{-1}$.

a) Det som er avbildet er bl.a. noen punktkilder som står så nær hverandre at de så vidt kan skilles fra hverandre i bildet. Gi en nedre grense på hvor tett disse punktkildene står.

Plasser en mengde tenkte punktkilder med avstand T i mellom seg på en rekke. Siden punktene kan skilles får vi da et signal med periode T i bildet. T kan ikke være mindre enn minste periode i bildet: $T >= T_{min} = 1/f_{max}=1/5mm=0.2mm$.

b) Hvor tett må vi sample dette bildet for å unngå aliasing? Gi en nedre grense for *samplingsfrekvensen*, f_s. La oss videre anta at vi har samplet bildet med en rate så vidt over denne grensen.

Samplingsteoremet krever $f_s>2f_{max}$, altså må $f_s>2*5mm^{-1}=10mm^{-1}$.

c) Anta den geometriske transformen

$$x' = 0.5x + 10$$
$$y' = 0.5y + 20$$

der x og y er koordinatene i ”innbildet”, x' og y' er de transformerte koordinatene, og det vil benyttes en ”vanlig” resampling ved baklengstransformasjon.

Hva vil den effektive samplingsraten være etter en slik transform, og hvilke (uønskede) effekter vil dette kunne gi opphav til?

Den effektive, nye samplingsraten blir halvert: $f_{sNy}=0.5f_s$. Altså vil vi i vårt bilde, hvor vi har samplet med så lav rate vi kan, ende opp med aliasingproblemer.

d) Anta den samme geometriske transformen som i oppgave c). Et alternativ til vanlig resampling vil være å la hvert utpiksel være gjennomsnittet av et 2x2 eller 3x3 pikselnabolag. En slik løsning vil her trolig gi bedre resultat. Forklar hvorfor.

Dette vil være det samme som å først gjøre en 2x2 eller 3x3 lavpassfiltrering, så velge ut piksler (resample). Altså utfører vi en **antialiasing** hvor vi demper frekvenser den nye raten ikke takler før vi (re)sampler.

Konvolusjon

Oppgave 2

Du har gitt konvolusjonskjernene A=[1 1] og B=[1 -1]. La C=(A*A^T*A^T*B), der "*" uttrykker konvolusjon og T er den transponerte.

a) Utfør konvolusjonene og vis hvilken konvolusjonskjerne C tilsvarer.

<LØSNING>

$$C = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

</LØSNING>

b)

Hva slags filter er dette, og hva brukes det til?

Det er "Hx" som brukes som konvolusjonskjerne i Sobel-operatoren for å

finne vertikale kanter.

Oppgave 3

Laplace-operatoren, $\nabla^2 f(x)$, er i 1D kun den andrederiverte av funksjonen f(x). Anta at vi benytter en enkel pikseldifferanse til å estimere den deriverte: f'(i) = f(i)-f(i-1). Utled filtermasken for 1D Laplace-operatoren basert på dette.

Siden f'(i) = f(i)-f(i-1), blir f''(i) = f'(i)-f'(i-1) = [f(i)-f(i-1)]-[f(i-1)-f(i-2)] = f(i)-2f(i-2)+f(i-2). Så sentrerer vi om piksel i og bytter fortegn og får ∇^2=[-1 2 -1]

Oppgave 4

Hvordan bruker vi Laplace-operatoren ved kantdeteksjon?

Vi får to ekstremverdier på $|\nabla^2|$ når vi passerer en kant. Kantens eksakte posisjon finnes fra nullgjennomgangen til ∇^2.

Fourier-transform

Oppgave 5

Anta at vi har et NxN bilde, *f*, og at vi har gjort en diskret Fourier-transform (i praksis en FFT) av dette bildet og lagret resultatet i NxN-matrisen *F*, der *F*(0,0) gir "DC-komponenten"/nullfrekvensen. Hvis nå alle koeffisientene i *F* er null bortsett fra *F*(5,3) og *F*(N-5,N-3), beskriv hvordan bildet ser ut.

Bildet må være en lineærkombinasjon av to basisbilder: et cosinus- og et sinusbilde med hele antall sinus/cosinus-repetisjoner lik 5 og 3 i de to dimensjonene. Vi har altså et bilde av en "bølge" som har 5 repetisjoner langs ene aksen og 3 langs den andre. Alle faseforskyvninger av denne bølgen gir også det beskrevne resultat.

Oppgave 6

Én måte å implementere et lavpassfilter på er å gjøre en Fourier-transform, sette alle koeffisientene som tilsvarer frekvenser over en viss terskel til null, for så å transformere tilbake. Hvilke typiske uønskede effekter i bildet vil dette kunne resultere i?

"Brå overgang" -> **ringing-effekter**. Fremgangsmåten tilsvarer en filtrering i bildedomenet med en filterkjerne som ligner en trunkert sinc-funksjon.

Oppgave 7

Hva sier konvolusjonsteoremet?

Konvolusjon i bildedomenet tilsvarer en multiplikasjon i frekvensdomenet. Og omvendt.

Oppgave 8

En venn gir deg en tallmatrise, og hun forklarer: "Hvis jeg filtrerer bildene mine med et konvolusjonsfilter med en kjerne lik denne matrisen, får jeg sykt kult resultat. Jeg lurer veldig på om filteret er et lavpass, høypass eller båndpass, og hvordan dette eventuelt endrer seg med retningene på strukturene i bildet. Du kan jo en del om data og digitale bilder og sånn, kunne du hjulpet meg?".

Du starter opp Matlab/Octave mens du ytrer: "Selvsagt, ikke noe problem, vi skal nok snart ha analysert din filterkjerne og få svar på dine spørsmål."

Så, forklar hvordan du ville gått frem for å besvare spørsmålene hennes.

Nøkkelord: Nullutvidelse. Spekter. Konvolusjonsteoremet.

Kompresjon

Oppgave 9

Gitt en symboltabell med symboler og deres hyppighet, hva vil entropien til denne fortelle deg?

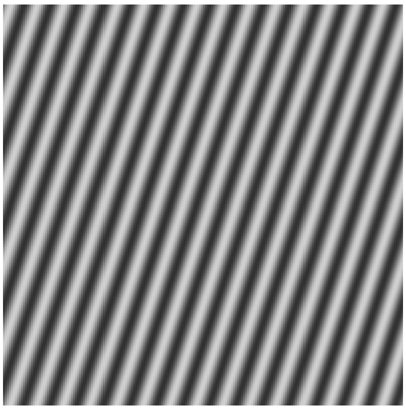
Entropien gir en nedre grense for antall bit vi trenger pr. symbol.

Oppgave 10

Hvordan brukes histogrammet til en sekvens av symboler i Huffman-koding, og hvordan bør histogrammet se ut dersom Huffman-koding skal være en godt egnet komprimeringsmetode?

Huffman-koding bruker histogramme til å tilordne kodelengder avhengig av antall forekomster av et symbol. Dette egner seg mest dersom noen symboler forekommer mye oftere enn andre, for eksempel etter differansetransform, eller på sekvenser som i utgangspunktet har symboler med ulik hyppighet.

<u>Oppgave 11</u>



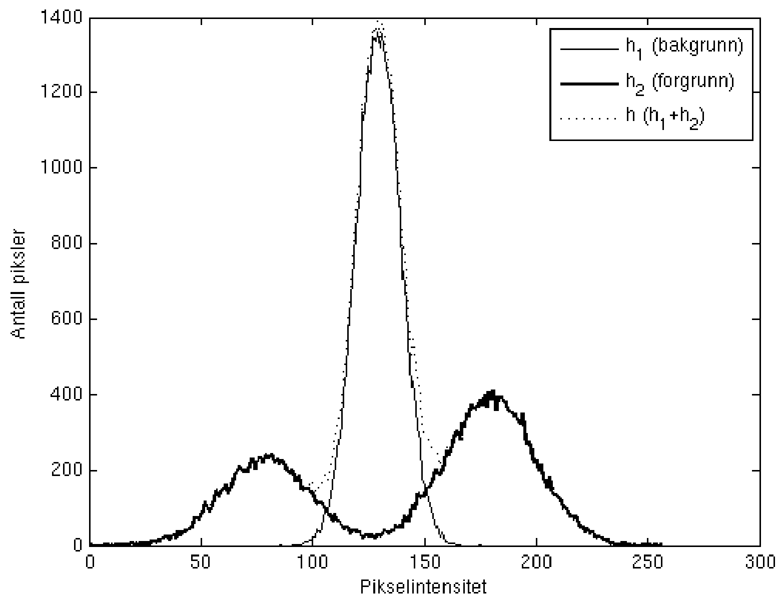
Bildet over viser en enkel todimensjonal "sinusbølge" med en viss frekvens og retning. Diskuter (kort) hvilken av følgende kompresjonsmetoder som er mest egnet for kompresjon av dette bildet:

- Huffman-koding av det originale histogrammet.
- Løpelengdetransform diagonalt i bildet.
- Differensetransform diagonalt i bildet.
- JPEG-koding.

Stikkord: vi har kun en sinus og bør da velge JPEG som bruker frekvensbasert koding ved DCT.

Segmentering ved terskling / gråtoneklassifikasjon

<u>Oppgave 12</u>



I figuren over vises histogrammet, *h*, til et bilde som består av to klasser: forgrunn og bakgrunn. Videre har vi tegnet inn histogrammene til disse to klassene, h_1 og h_2. Vi har altså $h=h_1+h_2$.

Vårt mål her er å benytte pikslenes gråtoneverdier til å bestemme om en piksel skal settes til forgrunn eller bakgrunn (klassifisere).

a) Basert på *h*, h_1 og h_2 i figuren, for hvilke pikselintensitetsintervaller ville du klassifisert til bakgrunn og for hvilke intervaller ville du klassifisert til forgrunn? (Forklar hvorfor.)

Minimerer antall feilklassifiserte piksler ved å velge forgrunn der $h_2(i)>h_1(i)$..

b) Ved en slik intervallinndeling som du kom frem til i oppgave a), benytt h_1 og h_2 til å gi et uttrykk for antall feilklassifiserte piksler.

Ved ”fasit”-regelen over; antall feilklassifiserte piksler ved hver intensitet summert, altså \sum_{i=0}^{G-1} min(h_1(i),h_2(i)).

c) La p_1 og p_2 være normaliserte bakgrunns- og forgrunnshistogrammer, og la *B* og *F* være a-priori sannsynlighet for henholdsvis bakgrunn og forgrunn.

Uttrykk *h* ved hjelp av p_1, p_2, *B*, *F* og *N*, der *N* er antall piksler totalt i bildet.

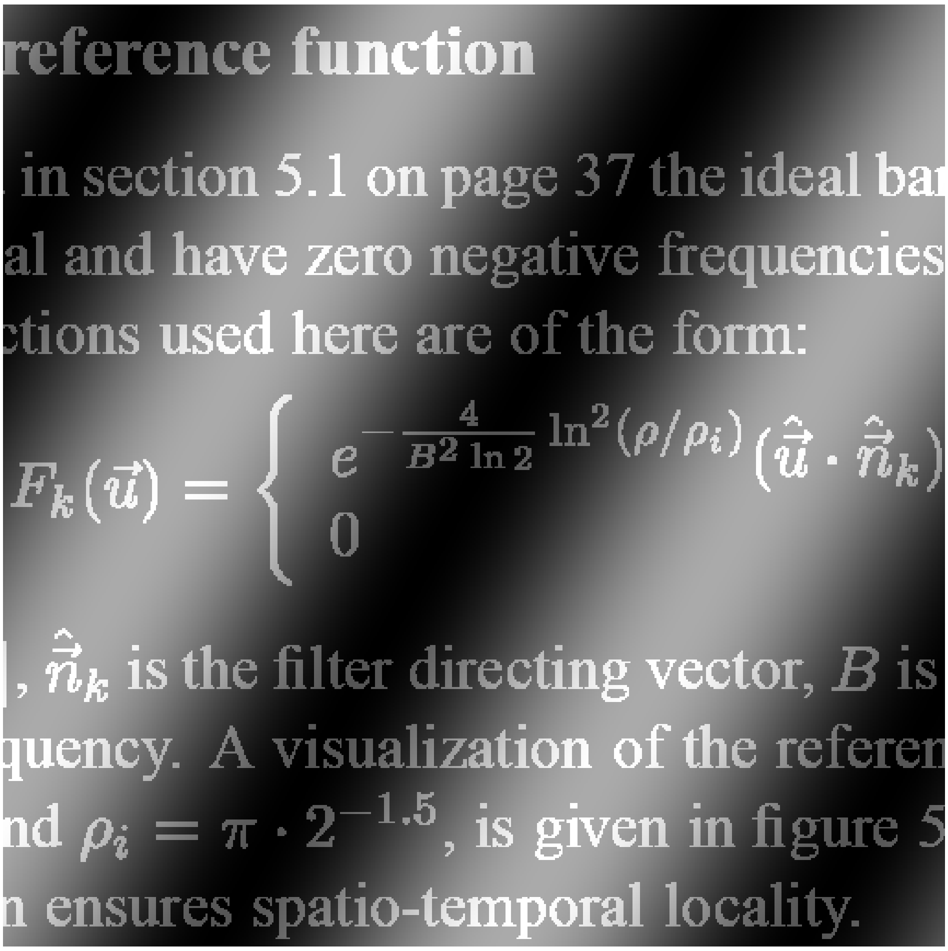
N*B er antall bakgrunnspiksler, N*F antall forgrunnspiksler. Vi har dermed h1=N*B*p1, h2=N*F*p1, altså er h = N*B*p1+N*F*p2 = N(B*p1+F*p2).

d) For et gitt, vilkårlig bilde har vi ikke h_1 og h_2 tilgjenglig, vi har kun det totale histogrammet *h*. Hvilken tilleggsinformasjon til bildet må vi ha for å finne de *eksakte* histogrammene h_1 og h_2 (ikke estimater)?

Vi trenger "fasiten", hva vi søker, altså hvilke piksler som er forgrunn og bakgrunn, for å finne h1 og h2.

<u>Oppgave 13</u>

Vi har gitt bildet under. Bildet inneholder tekst med varierende belysning (der bakgrunnen har store lokale variasjoner i gråtoner).



La oss si at vi er interessert i å klassifisere pikslene i bildet over til enten å være en del av forgrunnen (teksten) eller bakgrunnen, altså segmentere bildet. Gi et forslag til hvordan du ville gått frem i dette tilfellet.

Med tankegangen/notasjonen fra forrige oppgave: h1 og h2 overlapper (nærmest fullstendig), så selv med "riktig" h1 og h2 kan vi ikke finne en enkel global terskel. Studentene bør komme opp med et begrunnet svar hvor lokal/adaptiv terskling og/eller preprosessering er med.