Løsningsforslag

UNIVERSITETET I OSLO

Det matematisk-naturvitenskapelige fakultet

Eksamen i:	INF2310 — Digital bildebehandling
Eksamensdag:	Mandag 6. juni 2016
Tid for eksamen:	14:30 – 18:30
Løsningsforslaget er på:	**9 sider**
Vedlegg:	**Ingen**
Tillatte hjelpemidler:	**Ingen**

- Det er **5** oppgaver i dette oppgavesettet.

- Les gjennom hele oppgavesettet før du begynner å løse oppgavene! Kontroller at oppgavesettet er komplett før du begynner å besvare det. Dersom du savner opplysninger i en oppgave, 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 tilsammen **20** deloppgaver. **Hver deloppgave teller like mye.** Det lønner seg derfor å disponere tiden slik at man får besvart alle deloppgavene. Hvis du står fast på en deloppgave, gå videre slik at du får gitt et kort svar på alle deloppgavene.

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

1. Tema: Konvolusjon

La intensiteten omkring pikselposisjonen (x,y) være modellert ved polynomet

$$f(x,y) = k_1+k_2x +k_3y+k_4x^2+k_5xy+k_6y^2$$

I et lokalt 3x3 område rundt posisjonen (x,y) vil da intensitetene være

$k_1-k_2-k_3+k_4+k_5+k_6$	$k_1-k_3+k_6$	$k_1+k_2-k_3+k_4-k_5+k_6$
$k_1-k_2+k_4$	k_1	$k_1+k_2+k_4$
$k_1-k_2+k_3+k_4-k_5+k_6$	$k_1+k_3+k_6$	$k_1+k_2+k_3+k_4+k_5+k_6$

a) Vis at filtermasken

$$\begin{bmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{bmatrix}$$

gir et skalert estimat av korrekt lokal Laplace-verdi for denne modellen.

Svar: Den korrekte lokale Laplace-verdien av denne modellen er gitt ved

$$\nabla^2(f(x,y)) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} = 2(k_4+k_6)$$

Den gitte filtermasken gir $-6(k_4+k_6)$, som altså er et skalert estimat.

b) Forklar hvordan vi kan gjøre Laplace-filteret robust for støy, og vis resultatet.

Svar:Vi kunne gjort en lavpassfiltrering av bildet, og så en Laplace-filtrering. I stedet kan vi utnytte kommutativitets-egenskapen til konvolusjonsoperasjonen, og konvolvere Laplaceoperatoren med et lavpassfilter, og får 5x5 filteret:

$$\begin{bmatrix} -2 & 1 & -2 \\ 1 & 4 & 1 \\ -2 & 1 & -2 \end{bmatrix} * \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} -2 & -3 & -2 & -3 & -2 \\ -3 & 0 & 6 & 0 & -3 \\ -2 & 6 & 16 & 6 & -2 \\ -3 & 0 & 6 & 0 & -3 \\ -2 & -3 & -2 & -3 & -2 \end{bmatrix}$$

c) Forklar hva vi mener med at et 2D filter er "separabelt". Er det støy-robuste Laplace-filteret du laget separabelt? Forklar..

Svar: Et filter er separabelt hvis filtreringen kan utføres som to sekvensielle filtreringer, dvs hvis vi kan bruke assosiativitets-egenskapen ved konvolusjon ($f*h=(f*h_1)*h_2$, der $h=(h_1*h_2)$). Selve LoG-filteret er ikke separabelt, men vi har sett at hvis vi lager Laplace-filteret som en sum av to andre-deriverte konvolvert med en Gauss, så kan hver av komponentene som ble brukt til å produsere det, separeres. Filteret ovenfor er ikke separabelt i annen forstand enn at lavpassfilteret kan separeres.

2. Tema : Kompresjon og koding

Gitt følgende data, fra en komprimert melding: 00100010110111011100, samt Huffman-kodeboken:

Symbol	**Kode**
L	00
K	01
Y	100
E	101
T	110
I	111

a) Gjør dekomprimeringen og skriv den ukomprimerte meldingen.

Svar: Den ukomprimerte meldingen: LYKKETIL

b) Hvor mange biter hadde vi brukt per symbol hvis vi skulle kodet meldingen i forrige oppgave med vanlig binær koding?

Svar: Meldingen har 6 symboler, altså måtte vi brukt 3 biter per symbol.

c) Finn det gjennomsnittlige antall biter per symbol når vi brukte Huffman-koding. Hva blir kompresjonsraten sammenlignet med å bruke naturlig binærkoding?

Svar: Kompresjonsraten er gitt av $CR = \frac{\mathrm{b}}{\mathrm{c}}$, hvor b er antall biter per symbol med binærkoding, og c er gjennomsnittlig biter per symbol i den komprimerte datamengden. $CR = \frac{3}{\left(\frac{20}{8}\right)} = \frac{24}{20} = 1.2$.

d) Entropien er gitt ved $H = -\sum_{i=0}^{G-1} p(i)\,\log2(\mathrm{p(i)})$, hvor $p(i)$ er sannsynligheten for symbol i.

Hva blir entropien for meldingen i oppgave 1a? *Husk at* $log2\left(\frac{1}{2^n}\right) = -n$.

Svar:

$$H = -\sum_{i=0}^{G-1} p(i)\,\log2\big(\mathrm{p(i)}\big) = -\left[\frac{2}{8}log2\left(\frac{2}{8}\right) + \frac{2}{8}log2\left(\frac{2}{8}\right) + \frac{1}{8}log2\left(\frac{1}{8}\right) + \frac{1}{8}log2\left(\frac{1}{8}\right) + \frac{1}{8}log2\left(\frac{1}{8}\right) + \frac{1}{8}log2\left(\frac{1}{8}\right)\right] = -[2 * \frac{2}{8}log2\left(\frac{2}{8}\right) + 4 * \frac{1}{8}log2\left(\frac{1}{8}\right) = -\left[\frac{4}{8} * (-2) + \frac{4}{8} * (-3)\right] = 20/8 = 2.5.$$

e) Hva sier dette om hvor effektiv Huffman-kompresjonen av meldingen er?

Svar: Siden gjennomsnittlig antall biter for den komprimerte meldingen er lik entropien har vi fått en optimal komprimering gitt at vi bruker en teknikk som gjør komprimeringen symbol for symbol.

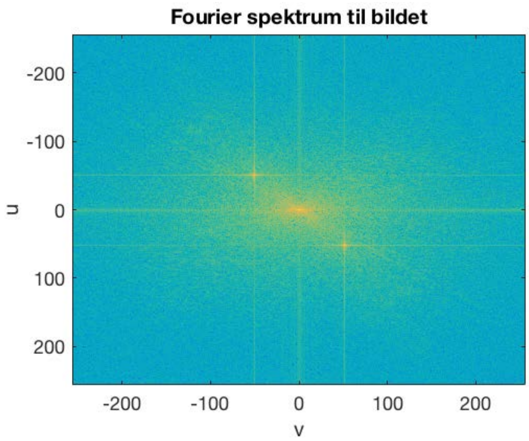
3. Tema : Fourier-filtrering

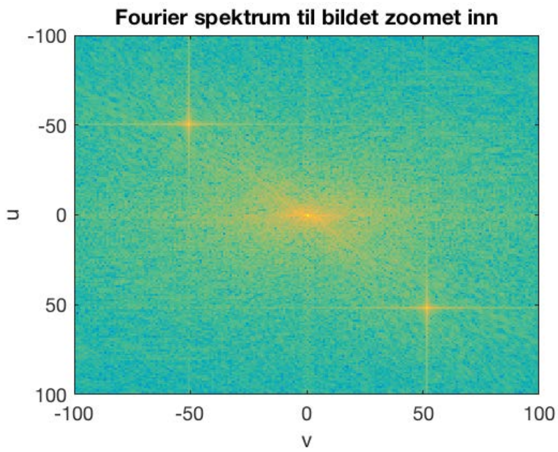
a) Hva innebærer konvolusjonsteoremet?

Svar: Konvolusjonsteoremet sier at sirkel konvolusjon i bildedomenet er det samme som multiplikasjon i Fourier-domenet og omvendt.

b) Gitt bildet nedenfor og bildets Fourier-spektrum; beskriv fremgangsmåten for å lage et konvolusjonsfilter som fjerner den periodiske støyen i bildet.

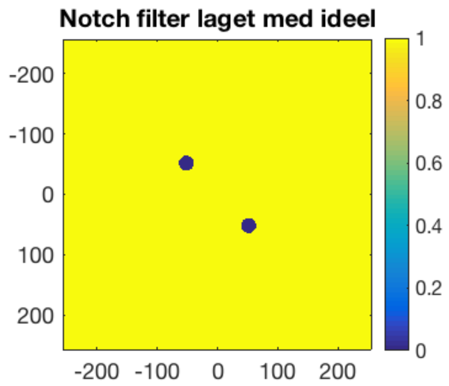
Bilde med støy

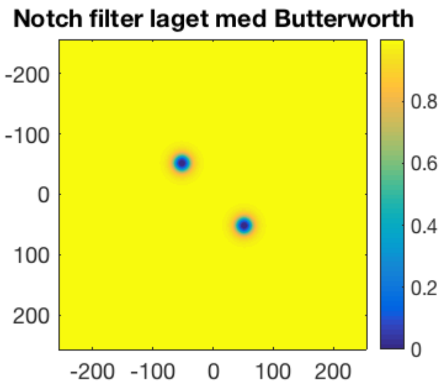




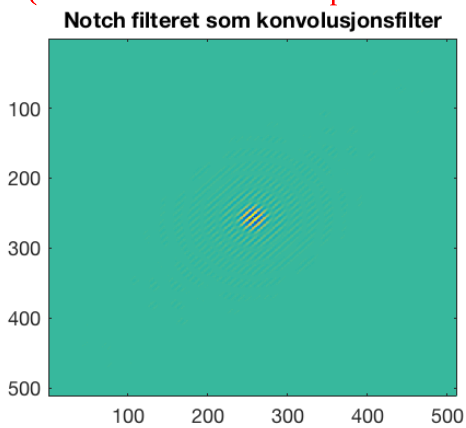
Svar: Her må vi for det første designe filteret i Fourierdomenet.

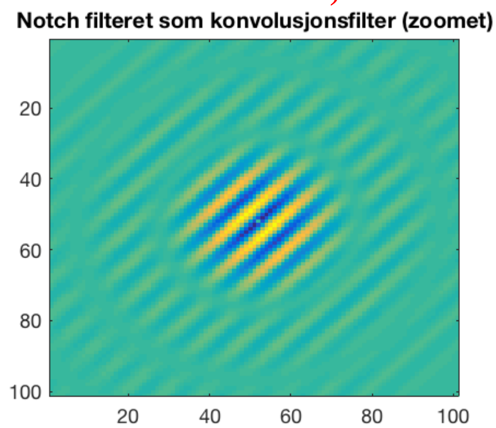
1. Vi må manuelt lokalisere frekvensen til støyen, i dette tilfellet ser frekvensen ut til å være omtrent u=-50,v=-50 og u=50,v=50.
2. Videre må vi legge inn notch-filtere rundt de uønskede frekvensene. Notch-filter lages ved å sentrere et høypassfilter rundt frekvensområdet vi ønsker å fjerne. Vi har lært at vi kan ha tre typer ”overganger” mellom passbåndet og stoppbånd: ideel, Buttwerworth og Gaussisk.





3. Oppgaven spesifiserer at vi vil ha et konvolusjonsfilter, vi må derfor ta invers 2D Fouriertransformen til filteret for å få et konvolusjonsfilter i bildedomenet. Dette vil bli slik (her har vi satt DC komponenten til 0 for å bedre illustrere filteret):





4. Vi kan så filtrere bildet med dette filteret, legg merke til av vi kun trenger den sentrale delen i bildedomenet, og kan begrense oss til denne uten særlig tap.

c) Nedenfor er det to bilder av "Lena" hvor bildet har blitt lavpassfiltrert med to forskjellige lavpassfiltre med cutoff-frekvens $D_0 = 0.2$. Vi har lært om tre forskjellige typer lavpassfiltre i kurset, hvilke filter tror du har blitt brukt på hver av bildene? Hva heter effekten som har oppstått i det første bildet, og hvorfor oppstår den?

Lena filtrert med filter 1

Lena filtrert med filter 2

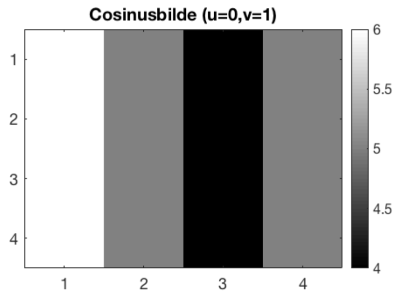
Svar: Det første bildet har blitt filtrert med et ideelt-lavpassfilter, mens det andre har blitt filtrert med et Butterworth med n = 2. Det andre bildet kunne også vært filtrert med et Gaussisk-lavpassfilter. Effekten som har oppstått i det første bildet kalles ”ringing” og oppstår fordi overgangen i Fourier-domenet mellom passbåndet og stoppbåndet er for bratt. Dette fører til at filteret i bildedomenet får ”ringing”. Jo, brattere overgangen mellom passbåndet og stoppbåndet er jo mer ringing får vi i bildet. Det idelle filteret har maksimalt bratt overgang og får derfor mest ringing. Butterworth har en faktor n som styrer hvor bratt overgangen er, jo høyere n jo brattere overgang desto mer ringing. Et Gaussisk filter har ingen ringing siden en Gauss i frekvensdomenet er en Gauss i bildedomenet. Kandidaten bør være innom flere av disse elementene.

4. Tema : Fourier-Transformasjonen

Den diskrete 2-dimensjonale Fouriertransformasjonen (2D DFT) er gitt som

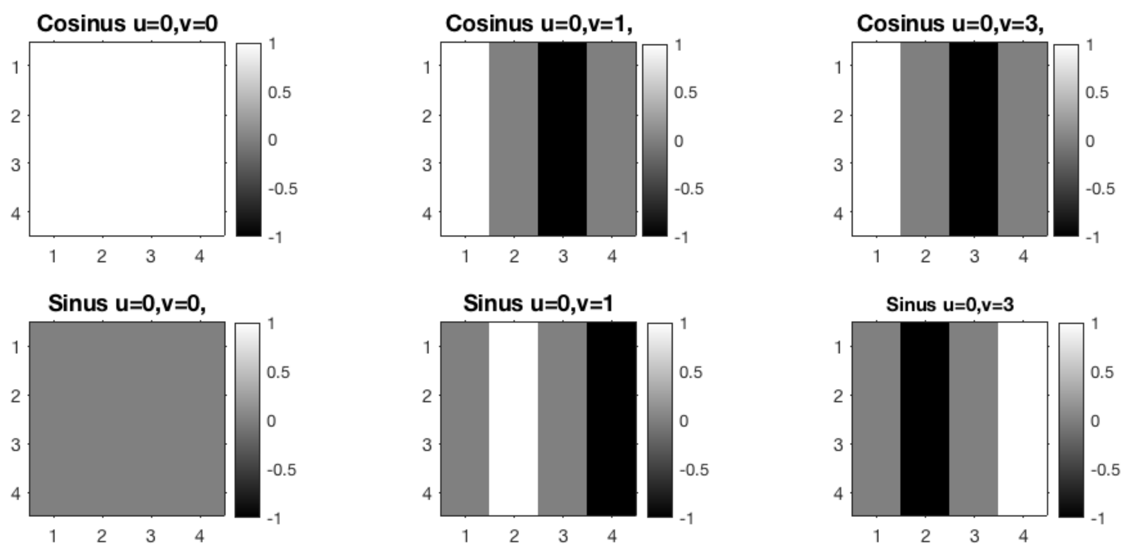
$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}.$$

Vi har et 4x4 cosinusbilde med vertikal frekvens u = 0 og horisontal frekvens v = 1. Middelverdien i bildet er 5, og bildet viser akkurat en hel periode.



6	5	4	5
6	5	4	5
6	5	4	5
6	5	4	5

a) Finn Fourier-transformasjonen til bildet og tegn en skisse av resultatet. Vær nøye med å indikere vertikal og horisontal frekvens på aksene, samt verdien for Fourier-koeffisientene. Du kan trenge basisbildene gitt nedenfor.



Svar:

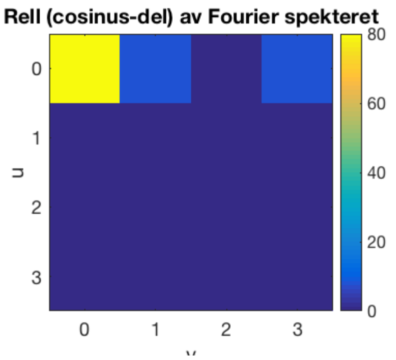
Cosinus (reell) del:

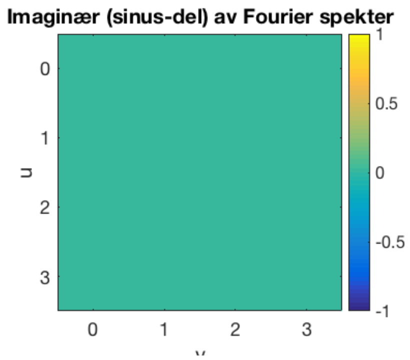
80	8	0	8
0	0	0	0
0	0	0	0
0	0	0	0

Sinus (imaginær) del:

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Svar: Dette kan man komme frem til ved å enten gjøre utregningen med formelen, som vi har lært er det samme som å ta summen av resultatet av å punktmultiplisere bildet med basisbildene oppgitt. Kandidaten bør også vite at Fourier-transformasjonen av et cosinusbilde med frekvens u=0, v=1, kun vil ha verdier for u=0,v=1, samt u=0,v=3 grunnet symmetrien i Fourierdomenet. Vi vet også at DC komponenten, frekvens (0,0) er proposjonal med gjennomsnittet N*M*mean(f(:)), altså 4*4*5=80. Alle koeffisientene vil kun ha relle verdier siden det opprinnelige bildet var et kosinusbilde.





b) Om vi ønsker en kompakt måte å representere bildet fra oppgave 4a på kan vi gjøre en kompresjon av Fourier-koeffisientene. Vi må da komprimere den reelle, cosinus, delen og den imaginære, sinus, delen hver for seg. Første steg kan feks være å gjøre en løpelengdetransform av verdiene ved å sette de etter hverandre rad for rad. Gjør dette.

Om du ikke løste forrige oppgave gjør nødvendige antagelser angående Fourier-koeffisientene.

Svar: Løpelengetransformen for cosinus delen, her er løpelendeparene (tallet,antall): (80,1) (8,1) (0,1) (8.1) (0.12).

Løpelendetransformen for sinus delen: (0,16).

c) Videre kan løpelengdeparene fra cosinus-delen og sinus-delen settes etter hverandre før vi gjør en Huffman-koding av løpelengdeparene. Lag Huffman-kodetabellen og lag den endelige Huffman-koden av Fourier-koeffisientene.

Hvor mange biter trenger vi totalt for å representere bildet om vi ser bort fra kodeboken?

Om du ikke løste forrige oppgave gjør nødvendige antagelser angående løpelengdeparene.

Svar: Alle løpelendeparene etter hverandre (80,1) (8,1) (0,1) (8.1) (0.12) (0,16).

Symbol	Kode	Sannsynlighet	Antall
80	111	1/12	1
1	00	4/12	4
8	100	2/12	2
0	01	3/12	3
12	101	1/12	1
16	110	1/12	1

Svar: Koden blir: 11100100000100100000110101110 Altså trenger vi kun 29 biter for å representere et 4x4 bilde om vi ser bort i fra kodeboken.

d) Hva må mottakersiden vite for å kunne dekomprimere koden tilbake til det originale bildet?

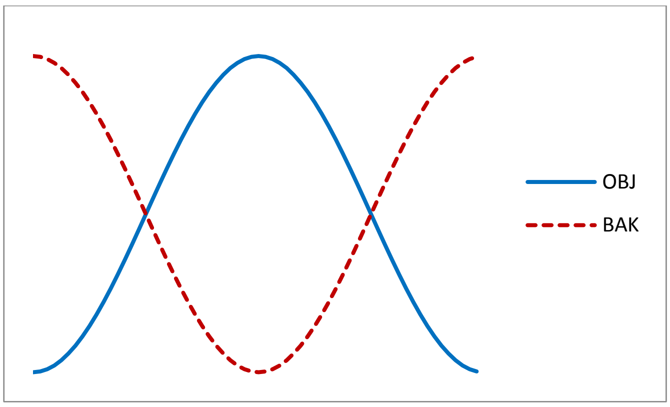
Svar: Mottakersiden må vite at vår komprimeringsalgoritme består av stegene: 1. Fourier-Transformasjon, 2. Løpelendetransform av cosinus-delen etterfulgt av sinus-delen. 3. Huffman-koding av løpelengdeparene. Mottakersiden må ha kodeboken, og videre vite at løpelendeparene skrives på formen (tallet,antall), samt at det aktuelle bildet er et 4x4 bilde.

5. Tema : Terskling og morfologi

I et analogt bilde med gråtoner $0 \le z \le \pi$ er histogrammene til bakgrunn og forgrunn gitt som kvadrerte sinusoider :

$$p(z) = \begin{cases} \cos^2(z) & \textit{bakgrunn} \\ \sin^2(z) & \textit{forgrunn} \end{cases}$$

Histogrammene ser altså slik ut:



a) Med *a priori* sannsynlighet F for forgrunn, finn et enkelt uttrykk for hvor to terskler må ligge for at vi skal få minst mulig feilterskling av pikslene.

Svar: Det er skjæringspunktene mellom de to fordelingene veiet med de respektive *a priori* sannsynlighetene som gir oss terskelverdiene som gir minst mulig total feil, altså er den implisitte løsningen (ligningene er en del av løsningsforslaget):

$$F \cdot p_f(T) = (1-F)\cdot p_b(T)$$

$$\Rightarrow F \cdot \sin^2(T) = (1-F)\cdot \cos^2(T)$$

$$\Rightarrow F \sin^2(T) - (1-F)\left(1-\sin^2(T)\right) = 0$$

$$\Rightarrow \underline{\underline{\sin^2(T) = 1-F \Leftrightarrow \cos^2(T) = F}}$$

Eller eksplisitt $T = \sin^{-1}\sqrt{1-F} \Leftrightarrow T = \cos^{-1}\sqrt{F} \Leftrightarrow T = \tan^{-1}\sqrt{\frac{1-F}{F}}$

b) Hvis vi antar at forgrunnsfordelingen har a priori sannsynlighet F = 0.5, og så bruker Ridler og Calvard's iterative tersklingsalgoritme på det totale histogrammet, hvilken terskelverdi i området [0;π] vil vi da ende på? Forklar.

Svar: Med F=0.5 er det totale histogrammet er helt flatt, slik at vi vil ende med T=π/2.

c) Beskriv en adaptiv tersklingsmetode som bruker enkel statistikk fra et glidende vindu i bildet.

Svar:

- Beregn middelverdi og standardavvik innenfor et glidende (w x w) vindu over hele bildet.
- Niblack's metode: Sett den lokale terskelverdien til

$$T(i,j) = \mu_w(i,j) + k\,\sigma_w(i,j)$$

- La ut-bildet være gitt ved

$$g(i,j) = \begin{cases} 0 \text{ hvis } & f(i,j) \le T(i,j) \\ 1 \text{ hvis } & f(i,j) > T(i,j) \end{cases}$$

d) Hvordan kan vi finne alle pikslene i et objekt i et binært bilde ved hjelp av morfologiske operasjoner?

Svar: Erodering fjerner piksler langs omrisset av et objekt. Vi kan altså finne kantene av objektene i bildet ved å subtrahere et erodert bilde fra originalbildet:
g = f - (f ⊖ S), der S er strukturelementet.

e) Hvordan kan vi styre tilkoblingstypen for pikslene langs kanten av resultat-objektet?

Svar: Det benyttede strukturelementet avgjør kantens tilkoblingstype:
Hvis alle 8-naboene til senterpiklset i et 3x3 strukturelement er 1,
får vi Sammenhengende kanter ved bruk av 4-tilkobling.
Hvis 4-naboene til senterpikslet i et 3x3 strukturelement er 1,
får vi sammenhengende kanter hvis (og bare hvis) man bruker 8-tilkobling.

TAKK FOR OPPMERKSOMHETEN!