

# Stikkordliste over temaer gjennomgått før midtveiseksamen 2019

Kristine Baluka Hein - krisbhei@ifi.uio.no

---

*Denne stikkordlisten er ment til å gi et overordnet blikk over hva som har blitt gjennomgått i INF2310 2019 før midtveiseksamen. Den gir derimot ikke en utfyllende beskrivelse av pensum, men tenkt til å være et lite supplement til forelesnings-slidsene og læreboken.*

---

## 1 Avbildning

**Forhold mellom størrelse i bildeplan  $y'$  og "virkelighet"  $y$**

Funnet ved geometri:

$$\frac{y'}{y} = \frac{s'}{s} \text{ og } \frac{1}{s} + \frac{1}{s'} = \frac{1}{f}$$

**Rayleigh kriteriet**  $\sin \theta = 1.22 \frac{\lambda}{D} \Leftrightarrow \frac{y}{s} = 1.22 \frac{\lambda}{D}$

## 2 Sampling og kvantisering

**Romlig oppløsning** Avstand for å kunne skille 2 punktkilder, Rayleigh

**Samplingsteoremet**  $f_s > 2f_{max} \Leftrightarrow T_{max} > 2T_s$

**Detektorer** Avstand mellom detektorer kan anses som  $T_s$ . Må ta hensyn til samplingsteoremet.

**Aliasing** Fenomen som oppstår av å undersample. Får lavere frekvenser enn original.

**Anti-aliasing** Lavpassfiltrering, foretas før sampling.  
*Detektorer tar gjennomsnitt av målt lys på sitt område*

**Piksel** Resultat etter målt lys + gjennomsnitt (sempel) fra detektor i kamera.  
Representasjon av digitalt bilde.

**Kvantisering** Digitaliserer verdier i gitte nivåer

**Kvantiseringsfeil** Sum av hver piksel sin avrundingsfeil etter kvantisering.

### 3 Geometriske operasjoner

**Affin transformasjon**

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_0 & a_1 & a_2 \\ b_0 & b_1 & b_2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

**Forlengsmapping** unødvendige beregninger kan oppstå

Går gjennom alle  $(x, y)$  i  $f$ .

Finner  $(x', y')$  via  $(x, y)$ .

Setter  $g(x', y') = f(x, y)$  dersom  $(x', y') \in g$

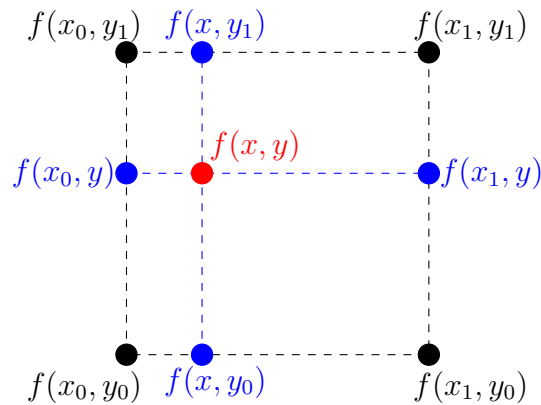
**Baklengsmapping** kun én verdi pr  $(x', y')$ , resample  $f$

Går gjennom alle  $(x', y')$  i  $g$

Finner  $(x, y)$  via  $(x', y')$ .

Setter  $g(x', y') = f(x, y)$  dersom  $(x, y) \in f$ , 0 ellers.

**Bilineær interpolasjon** Interpolér mellom fire kjente punkter; to ganger horisontal og én vertikal eller to ganger vertikal og én gang horisontal.



Figur 1: Illustrasjon av tanken bak bilineær interpolasjon. Her er det mulig å velge om en vil finne  $f(x_0, y)$  og  $f(x_1, y)$  eller  $f(x, y_1)$  og  $f(x, y_0)$  for å approksimere  $f(x, y)$ .

## 4 Gråtonemapping

### Histogram og kumulativt histogram

Histogram: hvor ofte en verdi repeteres.

Kumulativt: hvor mange forekomster av verdier før en verdi inklusiv verdien selv.

### Lineær transformasjon $T[i] = ai + b$

- b: endrer lyshet
- a: endrer kontrast

Endring av kontrast endrer også middelvei og standardavvik:

$$\mu_{ny} = a\mu + b \text{ og } \sigma_{ny}^2 = a^2\sigma^2$$

**Standardisering** Bildeserie skal bli statistisk like. Justering av middelvei og varians.

Fjerner døgnvariens i belysning, effekt fra støv på linse, aldringseffekter.

**Middelvei  $\mu$**  Gjennomsnitt av alle intensiteter i bilde

**Standardavvik  $\sigma$**  Differanse mellom alle verdier og middelvei i bildet. Mål på spredning av intensiteter.

### Endre til ønsket middelvei og standardavvik

$$a = \frac{\sigma_{ny}}{\sigma}$$

$$b = \mu_{ny} - a\mu$$

## 5 Histogrambaserte operasjoner

**Histogramutjevning** Ønsker: maksimal kontrast (*kun uniform histogram*)

Idéen:  $T(i) = G \int_0^i p(s) ds$

Ved diskret tilnærming:  $T[i] = \text{round}((G - 1)c(i))$

Ikke perfekt resultat pga tilnærming. Histogram søyler kan ikke 'splittes'

### **Histogramtilpasning**

Flytt på søyler i innbildets kumulative histogram  $c(i)$  slik at det ligner på det ønskede kumulative histogrammet  $c_q(i)$ :

For hver intensitet  $i$  finn intensitet  $j$  slik at  $c(i)$  er så lik som mulig  $c_q(j)$ .

**Lokal Gråtone Transformasjon (GTT)** Lokal transformasjon i et vindu om  $(x,y)$

To metoder:

- Lokal histogramutjevning / tilpasning
- Lineær standardisering

## **6 Naboskapsoperasjoner I**

**Konvolusjon** Romlig filtrering

Kan gjøres på to måter ettersom hvordan en ønsker å behandle bilderandproblemet:

- 1: Respons kun ved fullstendig overlapp mellom bilde og filter
- 2: Randutvid bildet og utfør filtreringen på randutvidet bilde

**Randutvidelse** Forskjellige metoder gir forskjellig resultat avhengig av hva slags filtrering

- Utvid m/ konstant verdi
- Nærmeste pikselverdi
- Symmetrisk utvidelse
- Periodisk utvidelse

**Korrelasjon** Brukes til mønstergjenkjenning

**Filter** naboskap + transformasjon

**Lavpassfilter** Utsmører/blurrer bildet

**Ulike lavpassfiltre**

Middelverdi, Gauss, Rang, Median, Alfa-trimmet, K Nearest Neighbour, K Connected Neighbour, Minimal Mean Square Error, Max Homogenitet

**Separabelt filter** Kan uttrykke et filte som ytre produkt av to vektorer. Raske utregning av filtrering numerisk.

## 7 Naboskapsoperasjoner II

**Høypassfilter** Kan uttrykkes som Høypass = original - lavpass. Høy respons ved kanter/skarpe overganger

### Estimere gradient, mer støyrobuste versjoner enn 1D

x horisontal retning og y vertikal retning

- Prewitt:  $h_x = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}, h_y = h_x^T$
- Sobel:  $h_x = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}, h_y = h_x^T$
- Frei-Chen:  $h_x = \begin{pmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -1 & -\sqrt{2} & -1 \end{pmatrix}, h_y = h_x^T$

Alle kan separeres

**Laplace** Bedre til å detektere kanter (nullgjennomganger)

Nullgjennomgang er der kanter befinner seg.

**LoG** Laplace og Gauss filter

Gauss bidrar med å dempe støy, Laplace finner kanter. LoG er derfor mer støyrobust.

Må velge størrelse på Gauss med omhu; må ikke være større enn bildestruktur, men større enn ramper.

**Canny** Idé: Finn så nøyaktig som mulig hvor i bildet det er kant.

Filtrér med Gauss gitt standardavvik, estimer retning og styrke (magnitude) til gradient, tynn gradient-magnitude og utfør tilslutt hystereseterskling.

## 8 Farger

**Kromasitet** Dominerende bølgelengde og metning (mengde) av farge

**Farge** Kromasitet + 'lyshet' i bilde

**RGB** Bildet er delt i tre lag: Red Green Blue.

Representasjon kan variere fra hvilket utstyr en bruker for å hente og vise farge.

**HSI** Bildet deles i Hue Saturation Intensity.

**Histogramutjevning av RGB** Bytt til HSI, utjevn I

**CMYK** Representerer bildet ved Cyan Magenta Yellow Key (sort).

Brukes til trykk/printing.

**Pseudofarger** Mappe gråtoneintensiteter til RGB gjennom LUT. Visualisering og presentering av data.

## 9 Morforlogi

**Dilasjon** Utvider forgrunns-elementer. Fyller opp hull og legger til forgrunns-piks-ler (legger til:  $\oplus$ ).

$$(f \oplus s)(x, y) = \begin{cases} 1, & \text{dersom } s \text{ har noe overlap med } f \text{ om punktet } (x, y) \\ 0, & \text{ellers} \end{cases}$$

**Erosjon** Forminsker forgrunns-elementer og forstørker hull. Fjerner støy som er mindre enn strukturelementet  $s$  (trekker fra:  $\ominus$ ).

$$(f \ominus s)(x, y) = \begin{cases} 1, & \text{dersom } s \text{ overlapper fullstendig } f \text{ om punktet } (x, y) \\ 0, & \text{ellers} \end{cases}$$

### Finne kanter til forgrunnsobjekt

To måter:

- Dilatere bildet med et strukturelement, så trekke fra originalbildet
- Trekke fra originalbildet med erodert bilde med et strukturelement.

Naboskap til strukturelement vil påvirke naboskapet til kantene.

**Region-fylling**  $c_k = (c_{k-1} \oplus f) \cap f^c, k = 1, 2, \dots$

Henter ut region som tilsvarer hull i en forgrunns-komponent.

$c_0$  settes lik 1 i regionen en ønsker å fylle.

### Dualitet

- Dilasjon:  $f \oplus s = (f^c \ominus \hat{s})^c$
- Erosjon:  $f \ominus s = (f^c \oplus \hat{s})^c$

**Morfologisk Åpning**  $f \circ s = (f \ominus s) \oplus s$

Erosjon, så dilasjon. Erosjon for å fjerne støy, dilasjon for å få tilbake omtrentlige originale størrelser.

**Morforlogisk lukking**  $f \bullet s = (f \oplus s) \ominus s$

Dilasjon, så erosjon. Dilasjon for å tette hull, erosjon for å få tilbake omtrentlige originale størrelser.

**Hit-or-miss**  $f \circledast s = (f \ominus s_1) \cap (f^c \ominus s_2)$

Løser for hver av parentesene, så finner de pikslene som er felles. Brukes til å finne spesifikke mønstre, fjerne enkeltpiksler og tynne og utvide områder i forgrunnen

**Morfologisk tynning**  $f \otimes S = f \cap (f \circledast S)^c$

der  $S$  er mengde av ulike strukturelementer. For hver iterasjon, utfør hit-or-miss for hvert strukturelement i  $S$  og oppdater bildet som jobbes på med resultatet. Fortsett iterasjonene til ingen av strukturelementene gir noen forandring.