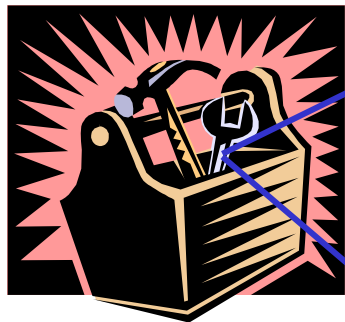
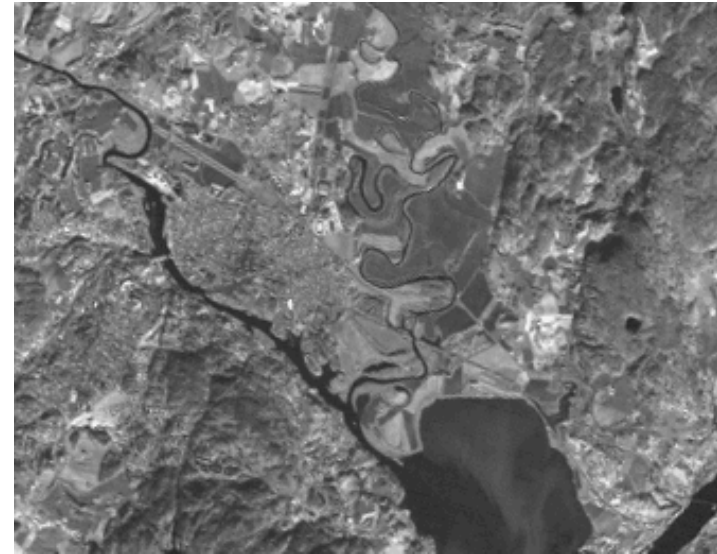
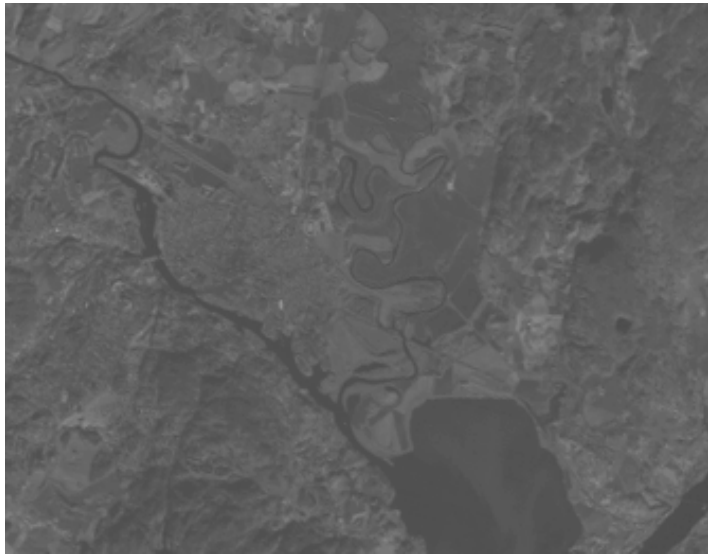

INF2310 – 7. februar 2018 – Ukens temaer

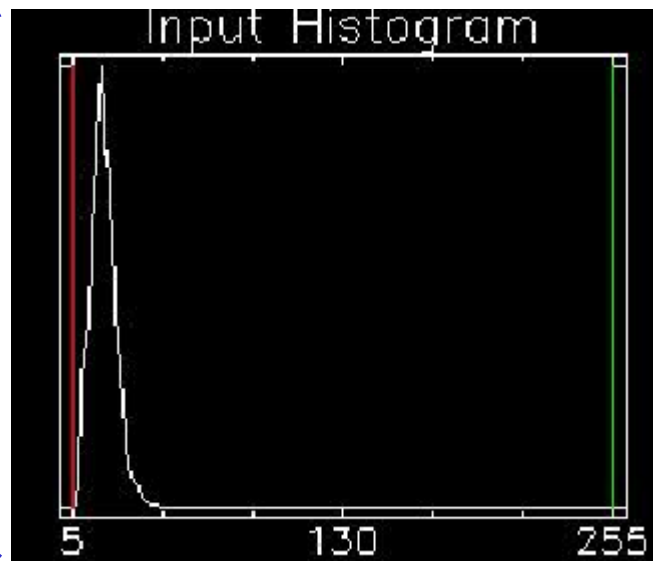
(Hovedsakelig fra kap. 3.1 og 3.2 i DIP)
(Histogrammer omtales i kap. 3.3)

- Histogrammer
- Lineære gråtonetransformer
- Standardisering av bilder med lineær transform
- Ikke-lineære, parametriske transformere
- **Neste uke:** Histogrambaserte transformere og lokal gråtonetransform

Hvordan endre kontrasten i et bilde?



Matematisk
verktøykasse



Men hva menes med kontrast?

- Kontrast sier noe om forskjellen i farge, lysstyrke, tekstur eller annen fysisk egenskap på et objekt og andre nærliggende objekter

- Vi holder oss til **lysstyrke** (luminositet), og **ikke** f.eks. →



- Noen vanlige metrikker:

- Weber-kontrast ("Weber fraction") $\frac{(I_f - I_b)}{I_b}$

- Michelson-kontrast ("Visibility") $\frac{(I_{max} - I_{min})}{(I_{max} + I_{min})}$

- RMS-kontrast (standardavvik) $\sqrt{\frac{1}{MN} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} (I(x, y) - \bar{I})^2}$

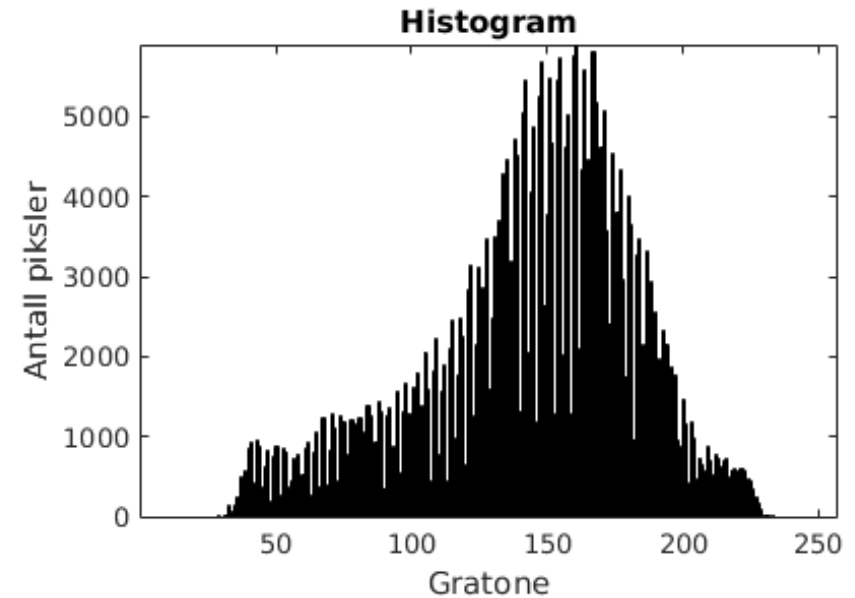
Nevneren knytter disse opp mot vårt perseptuelle system

Histogrammer

- Et histogram er en diskret funksjon som viser antall målinger innenfor (som oftest) uniforme intervaller i et datasett
- Vi jobber med bilder og får typisk
 - Et bilde som datasett
 - Pikkseleintensiteter som målinger
- Altså en oversikt over forekomsten til intensitetene i bildet
- Kan også ha histogrammer over avledede egenskaper i bildet

Gråtonehistogrammer

- Gitt et gråtonebilde med $n \times m$ piksler og G gråtoner
- Et histogram, $h(i)$, er slik at:
 $h(i)$ = antall piksler i bildet
med pikselverdi i
- Dannes ved å gå igjennom alle pikslene og telle gråtoner
- Vi har naturligvis at $\sum_{i=0}^{G-1} h(i) = n \times m$

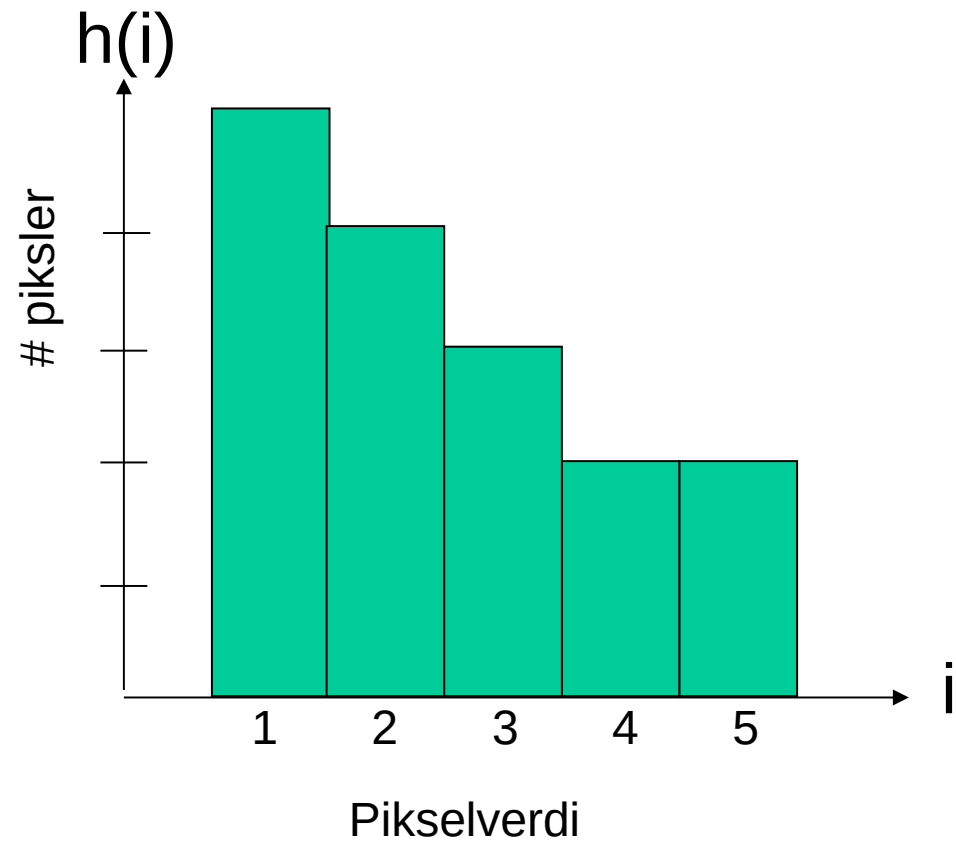


Eksempel - histogram

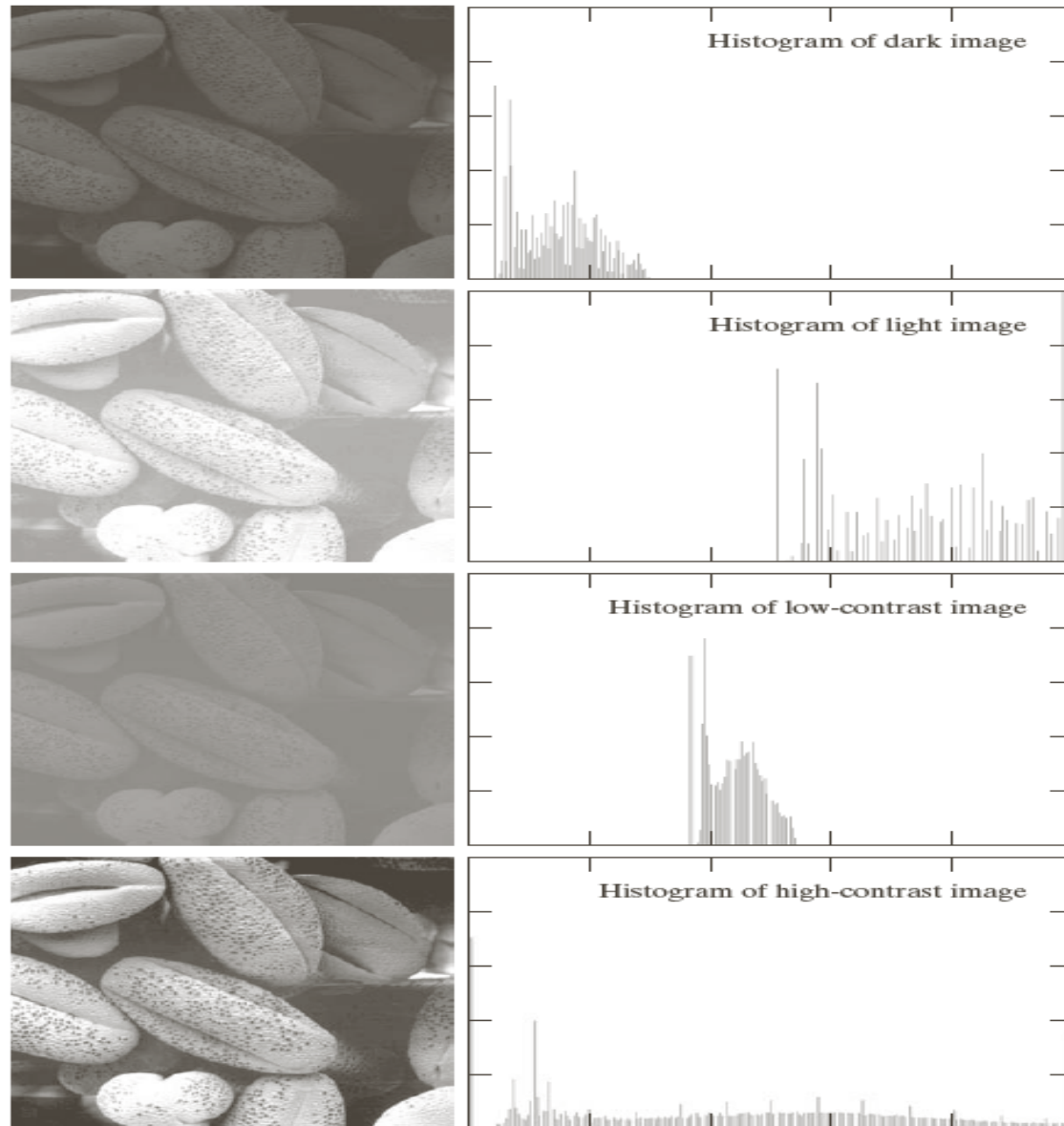
Bilde:

1	3	2	1
5	4	5	3
4	1	1	2
2	3	2	1

Histogram:



Eksempler



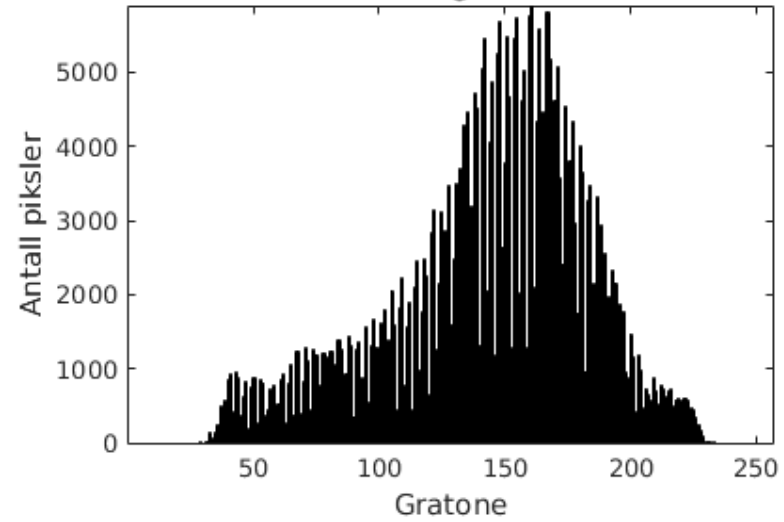
(Fig 3.16 s. 121)

Eksempler II

Image



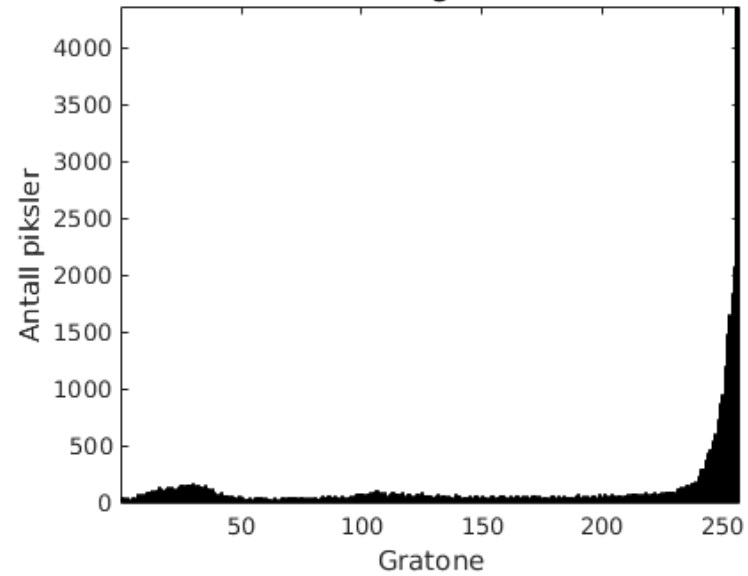
Histogram



Image

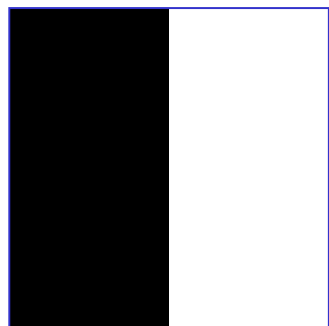


Histogram

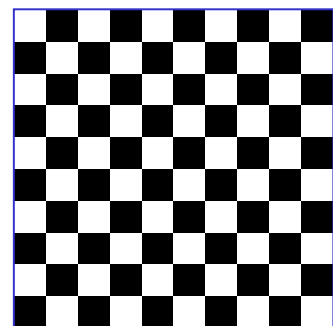


Oppgaver

Anta 8 bits uniform
intensititskvantisering



Hvordan ser histogrammet ut?



Hvordan ser
histogrammet ut?

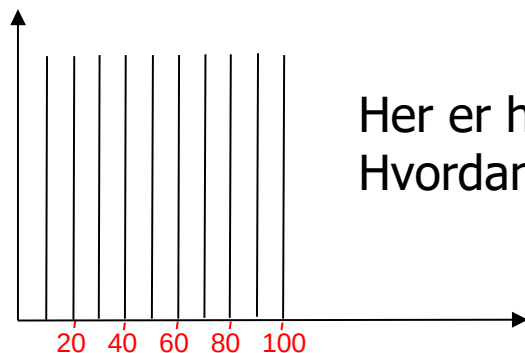
$\frac{1}{2}$

+

$\frac{1}{2}$

Hvordan ser
histogrammet ut?

Kan vi se dette ut i fra
de respektive histogrammer?



Normalisert histogram

- Vi har at $\sum_{i=0}^{G-1} h(i) = n \times m$
- Det normaliserte histogrammet:

$$p(i) = \frac{h(i)}{n \times m}, \quad \sum_{i=0}^{G-1} p(i) = 1$$

- $p(i)$ kan ses på som en **sannsynlighetsfordeling** for pikselintensitetene
 - "Uavhengig" av antall piksler / størrelsen på bildet

Kumulativt histogram

- Hvor mange piksler har gråtone mindre enn eller lik gråtone j ?

$$c(j) = \sum_{i=0}^j h(i)$$

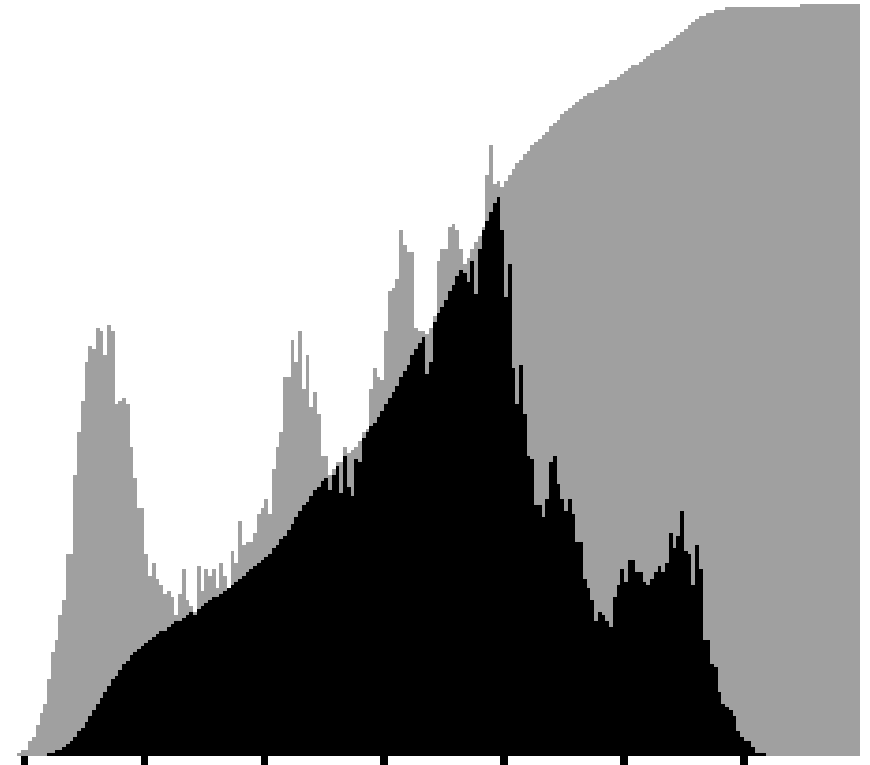
Hva er $c(G-1)$?

- Normalisert kumulativt histogram:

$$\frac{c(j)}{n \times m}$$

(Sannsynligheten for at en tilfeldig valgt piksel er mindre eller lik gråtone j)

Eksempel, kumulativt histogram



Histogram og (skalert) kumulativt histogram
i samme figur

Histogrammer av objekt-egenskaper

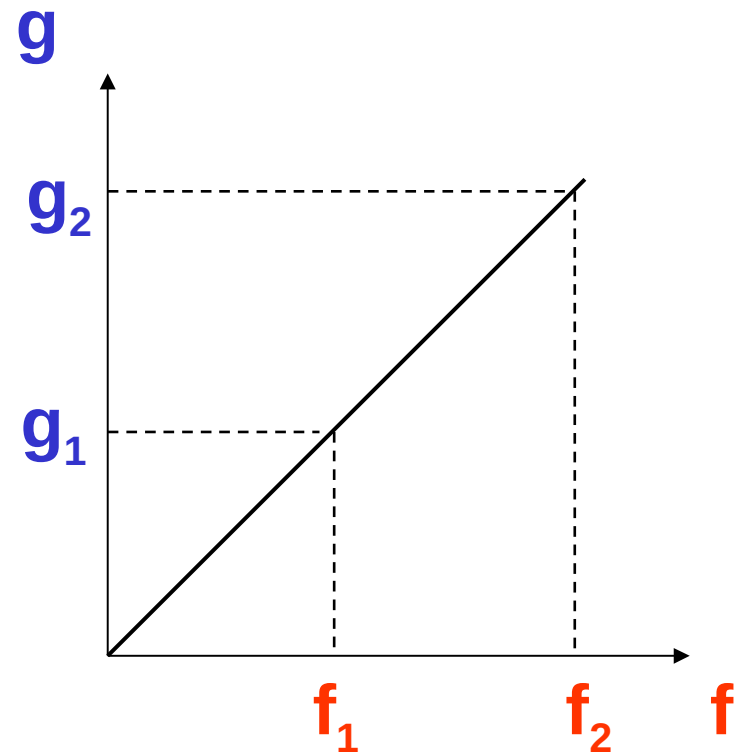
- Begrepsapparatet omkring histogrammer vil også komme til nytte i digital bildeanalyse
- Vi kan lage histogrammer over egenskaper, feks:
 - Objekt-størrelse:
 - Viser fordelingen av størrelsen på objektene, og danner grunnlag for å sette en terskel for å kunne fjerne små og uvesentlige objekter fra bildet (støy)
 - Objekt-momenter:
 - Viser fordelingen av beregnede momenter fra hvert objekt, og danner grunnlag for å samle grupper av objekter i klasser eller "clustre"

Gråtonetransformasjon

- Når vi viser en piksel på skjermen er intensiteten kontrollert av den tilhørende verdien i bildematriksen
- Vi kan opprette en *avbildnings-funksjon* mellom de tallene, f , som finnes i bildematriksen, og den intensiteten, g , vi ønsker på skjermen eller i vår nye bildematrise
 - For ett-båndsbilder: $g = T[f]$
 - T kan være en parametrisk funksjon, eller en tabell om antall mulige intensiteter er begrenset (f.eks. 8 bits bilder)
- Vi ser kun på ren gråtonetransformasjon, så ett og ett piksel transformeres uavhengig av nabopiksler, og uavhengig av posisjon i bildet → **global** transformasjon

Identitetsmapping

- Figuren viser sammenhengen mellom pikselverdien i inn-bildet (f) og pikselverdien til den samme pikselen i utbildet (g) etter en gråtonetransformasjon
- Hvis transformasjonen er en identitetsmapping, $g=f$, vil figuren vise en rett linje gjennom origo, med stigningstall 1
- $T[i] = i$



Lineær kontrastendring

- Lineær/affin strekking

$$T[i] = ai + b$$

$$g(x, y) = af(x, y) + b$$

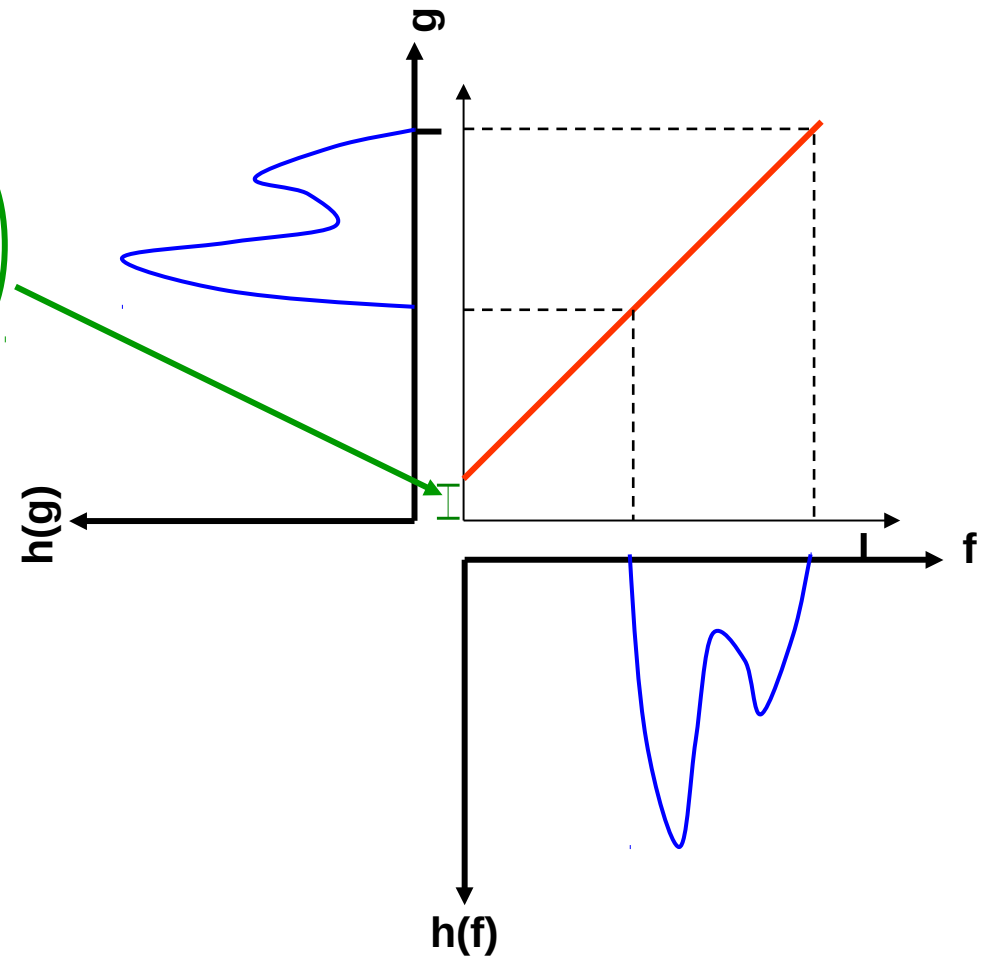
- **a** regulerer kontrasten, og **b** "lysheten"
- **a**>1: mer kontrast
- **a**<1: mindre kontrast
- **b**: flytter alle gråtoner **b** nivåer
- Negativer: **a**=-1 , **b**=maxverdi for bildetype

Endre "lysheten" (brightness)

- Legge til en konstant b til alle pikselverdiene ($a=1$)

$$g(x, y) = f(x, y) + b$$

- Hvis $b > 0$, alle pikselverdiene øker, og bildet blir lysere
- Hvis $b < 0$, bildet blir mørkere
- Histogrammet flyttes opp eller ned med b



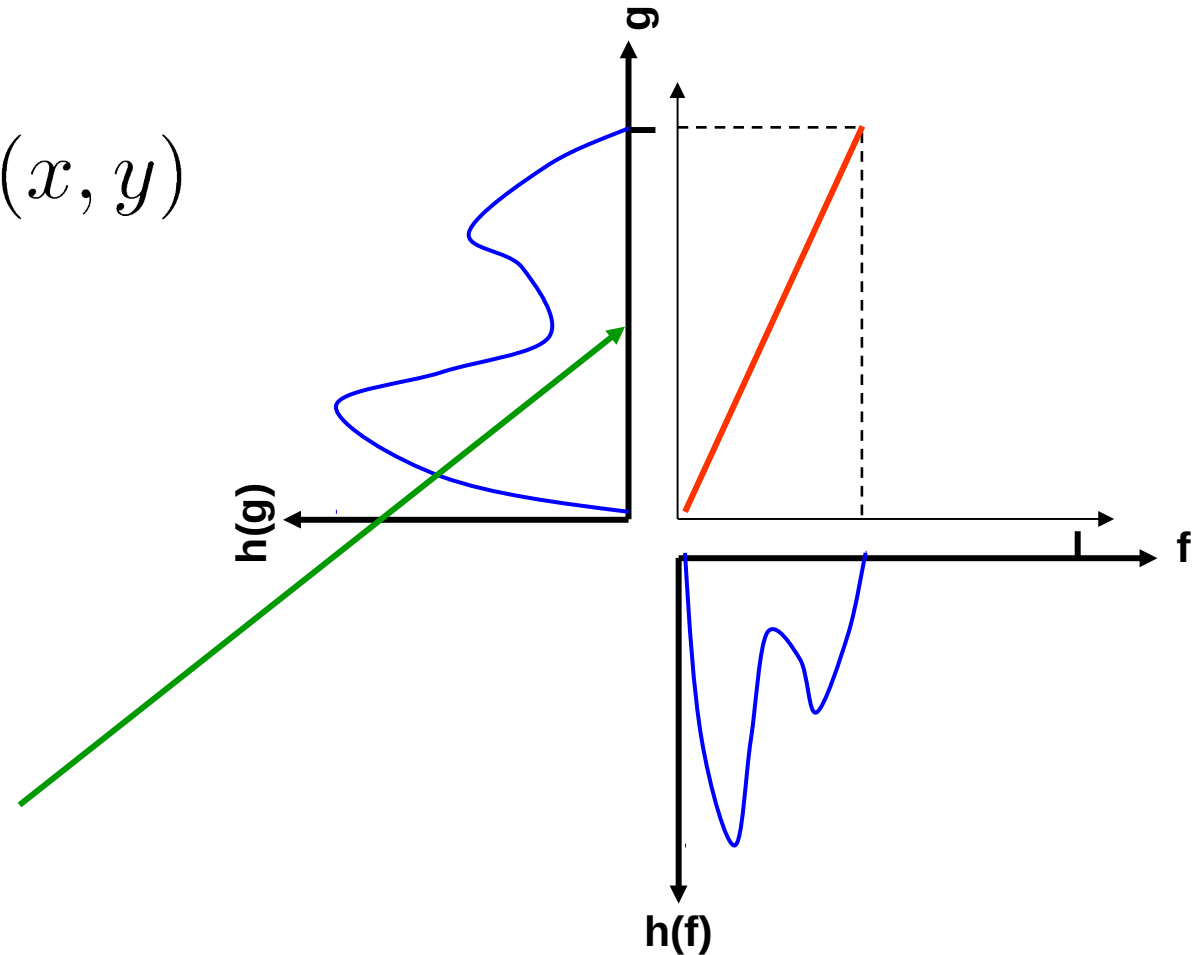
Endre kontrasten

- Multiplisere hver pikselverdi med en faktor a :

$$g(x, y) = a f(x, y)$$

- Hvis $a > 1$,
kontrasten øker
- Hvis $a < 1$,
kontrasten minker

- Eks: Bruke hele intensitetsskalaen
 - **Q:** Hva skjer med middelveiden?



Alternativ illustrasjon

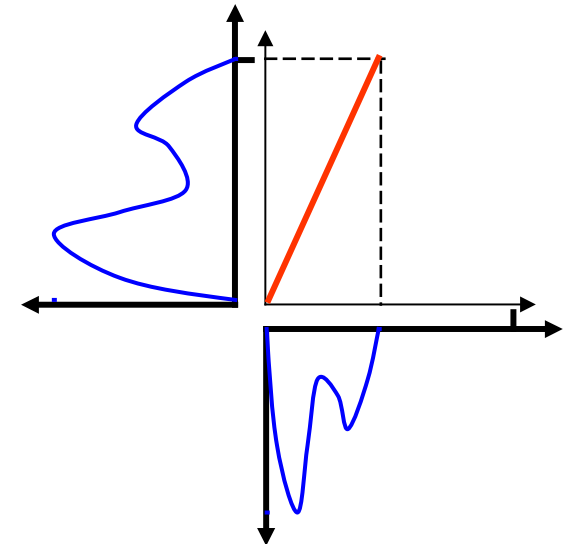
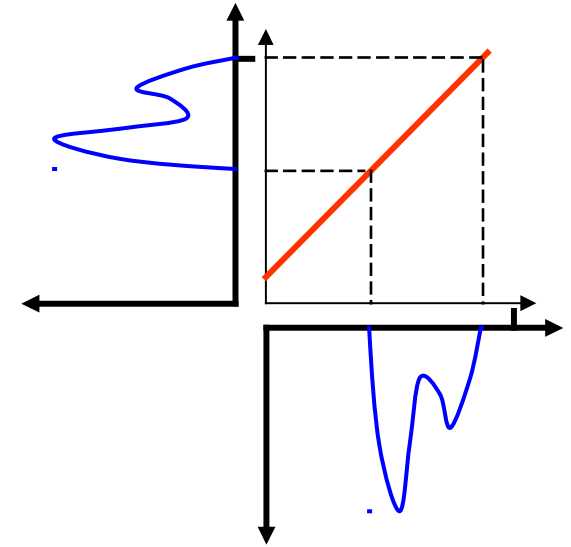
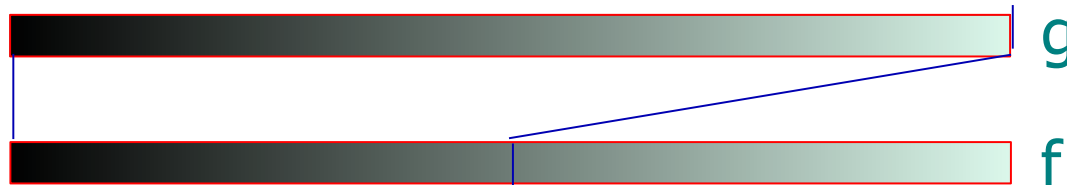
- Endre "brightness":

$$g(x, y) = f(x, y) + b$$



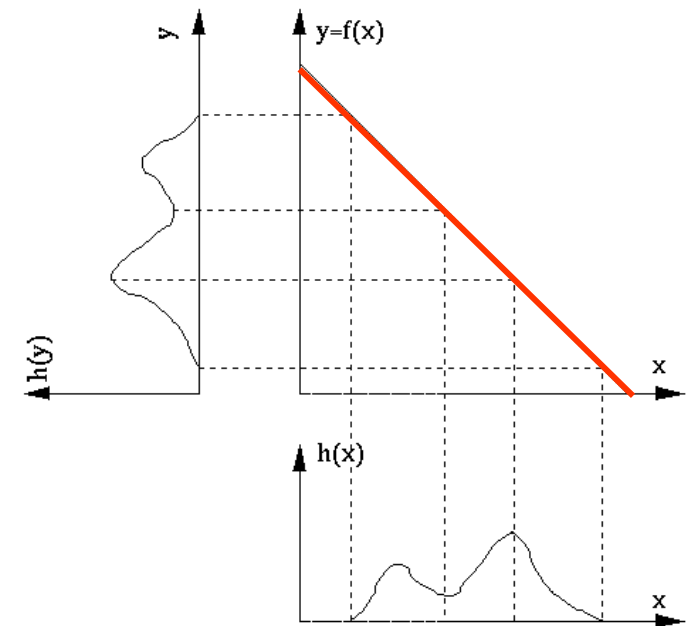
- Endre kontrast:

$$g(x, y) = a f(x, y)$$



Invertert gråtonebilde

- Danner bildets “negativ” ved å sett $a=-1$ og $b=\text{maksverdien}$
- Bildet får ikke negative verdier, men avbildningsfunksjonen har negativt stigningstall



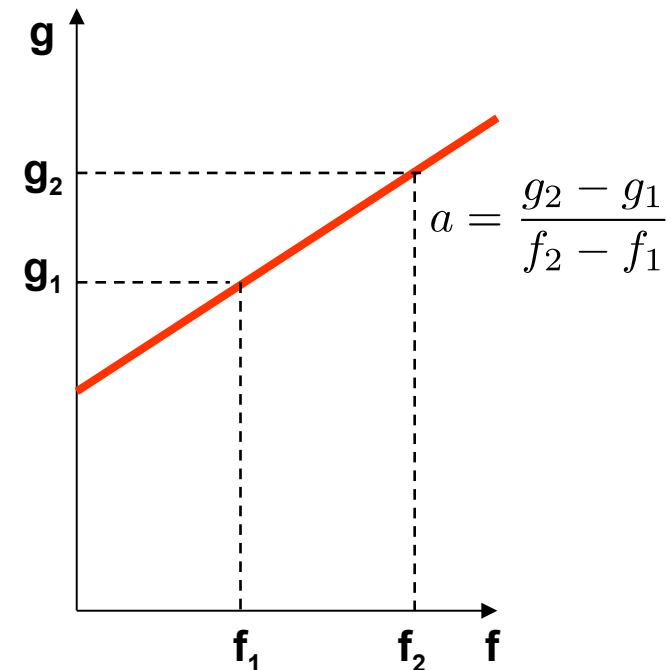
Fra gråtonenivå $[f_1, f_2]$ til $[g_1, g_2]$

- Endre intensiteter i intervallet $[f_1, f_2]$ til å ligge i $[g_1, g_2]$
- En lineær (affin) mapping fra f til g :

$$g(x, y) = g_1 + \frac{g_2 - g_1}{f_2 - f_1} [f(x, y) - f_1]$$

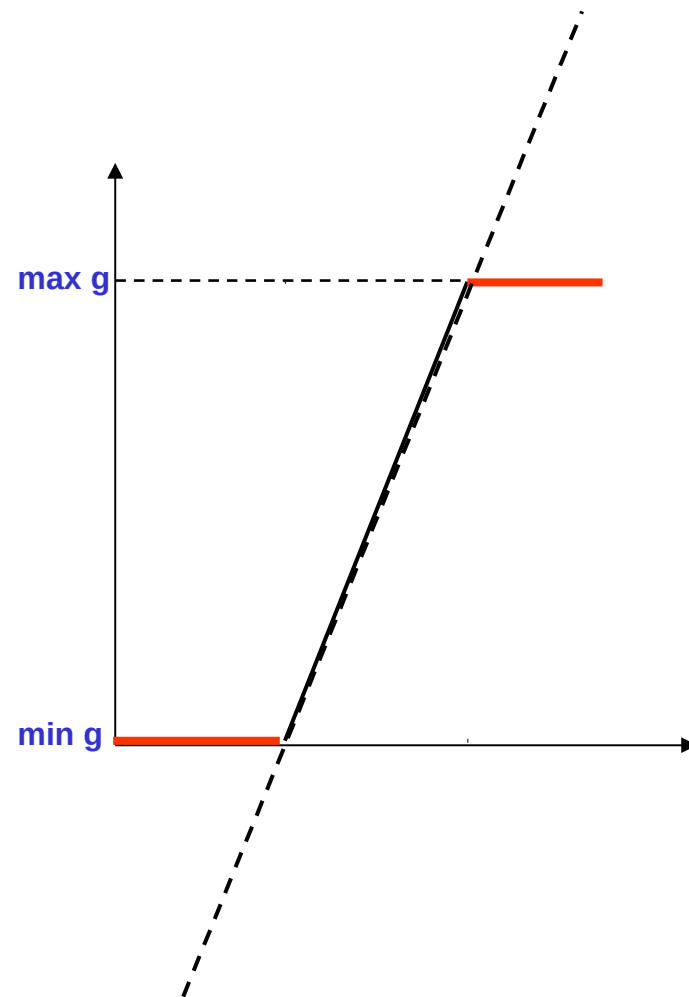
- Rett linje med stigningstall

$$a = (g_2 - g_1) / (f_2 - f_1)$$



Klipping etter transform

- Om $g(x,y)$ får verdier utenfor det støttede intervallet, foretas som oftest klipping av verdiene
- F.eks for et 8 bit «unsigned» bilde vil g bli tvunget innenfor intervallet $[0, 255]$



Standardisering av bilder

- Hensikt:
 - Fjerne variasjoner i «lyshet» og kontrast i en serie bilder
- Hvorfor? Fjerne effekten av
 - Døgnvariasjon i belysning
 - Aldringseffekter i lamper og detektorer
 - Akkumulering av støv på linser etc.
- Metode:
 - Justere middelveien og variansen til gråtoneverdiene i bildet ved hjelp av en lineær gråtonetransform
- Hvor:
 - Produkt-inspeksjon i industri
 - Medisinsk avbildning
 - Mikroskopering av celler
 - ...

Neste uke: Kan også standardisere bildene med **histogramspesifikasjon**, men vil da ikke beholde "formen" på histogrammet

Middelverdien av gråtonene

- Middelverdien av pikselverdiene i et bilde med $n \times m$ piksler og G gråtoner kan finnes
 - enten fra pikselverdiene direkte
 - eller indirekte fra bildets histogram, evt normalisert histogram

$$\mu = \frac{1}{n \times m} \sum_{x=0}^{n-1} \sum_{y=0}^{m-1} f(x, y)$$

$$= \frac{1}{n \times m} [0 \times h(0) + 1 \times h(1) + \dots + (G - 1) \times h(G - 1)]$$

$$= \frac{1}{n \times m} \sum_{i=0}^{G-1} ih(i) = \sum_{i=0}^{G-1} ip(i)$$

Hvorfor en fordel med det siste alternativet?

$$p(i) = \frac{h(i)}{nm}$$

(Normalisert histogram)

Varians av gråtonene

- Variansen av pikselverdiene i et bilde med $n \times m$ piksler og G gråtoner kan også finnes fra bildets histogram

$$\sigma^2 = \frac{1}{n \times m} \sum_{x=0}^{n-1} \sum_{y=0}^{m-1} [f(x, y) - \mu]^2$$

$$= \frac{1}{n \times m} \sum_{i=0}^{G-1} h(i) [i - \mu]^2$$

$$= \sum_{i=0}^{G-1} p(i) [i - \mu]^2$$

$$= \sum_{i=0}^{G-1} i^2 p(i) - \left[\sum_{i=0}^{G-1} i p(i) \right]^2$$

Kvadratroten av
variansen kalles
standardavvik:

σ^2 : varians
 σ : standardavvik

Variansen/
standardavviket sier
noe om kontrasten
i bildet

Justering av μ og σ^2

- Gitt inn-bilde med middelerverdi μ og varians σ^2
- Anta en lineær gråtone-transform $T[i]=ai+b$
- Ny middelerverdi μ_T og varians σ_T^2 er da gitt ved:

$$\mu_T = \sum_{i=0}^{G-1} T[i]p(i) = a\mu + b$$

- Dvs.

$$a = \sigma_T / \sigma, \quad b = \mu_T - a\mu$$

- Vi kan altså

- velge nye μ_T og σ_T^2 ,
- beregne a og b ,
- anvende $T[i]=ai + b$ på inn-bildet
- og få et ut-bilde med ønsket μ_T og σ_T^2

$$\begin{aligned}\sigma_T^2 &= \sum_{i=0}^{G-1} T[i]^2 p(i) - \left[\sum_{i=0}^{G-1} T[i] p(i) \right]^2 \\ &= \sum_{i=0}^{G-1} (a^2 i^2 + 2aib + b^2) p(i) - \left[\sum_{i=0}^{G-1} (ai + b) p(i) \right]^2 \\ &= a^2 \left[\sum_{i=0}^{G-1} i^2 p(i) - \left[\sum_{i=0}^{G-1} ip(i) \right]^2 \right] = a^2 \sigma^2\end{aligned}$$

Eksempel 1: Justering av σ

- Vil beholde middelveiden, slik at

$$\mu_T = \mu,$$

men ønsker ny σ_T .

- Bestem a og b i ligningen $T[i] = ai + b$:

$$a = \frac{\sigma_T}{\sigma}, \quad b = \mu_T - a\mu = \mu \left[1 - \frac{\sigma_T}{\sigma} \right]$$

$$\Rightarrow T[i] = \frac{\sigma_T}{\sigma} i + \mu \left[1 - \frac{\sigma_T}{\sigma} \right] = \boxed{\mu + (i - \mu) \frac{\sigma_T}{\sigma}}$$

Eksempel 2: Justering av μ og σ

- Ønsker at alle bildene i en serie skal ha samme (μ_T, σ_T) .
- Bestem a og b i ligningen $T[i]=ai+b$:

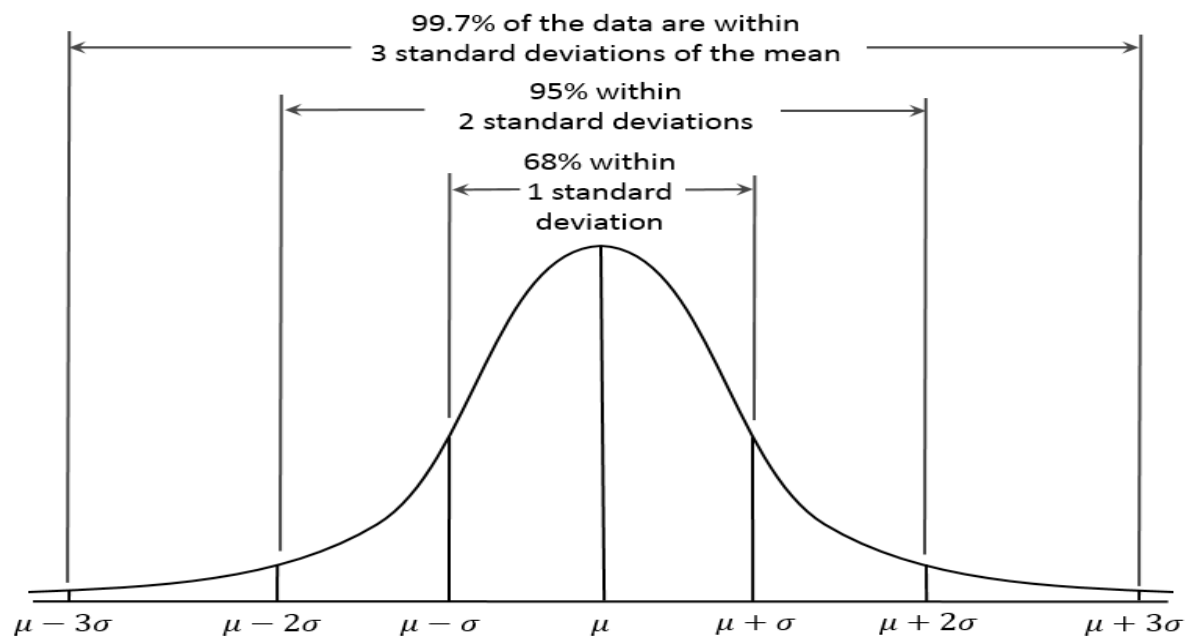
$$a = \frac{\sigma_T}{\sigma}, \quad b = \mu_T - a\mu = \mu_T - \mu \frac{\sigma_T}{\sigma}$$

$$\Rightarrow T[i] = \frac{\sigma_T}{\sigma}i + \mu_T - \mu \frac{\sigma_T}{\sigma} = \boxed{\mu_T + (i - \mu) \frac{\sigma_T}{\sigma}}$$

- For hvert bilde må vi finne bildets (μ, σ)

Valg av standardavvik

- Anta at histogrammet til innbildet er normalfordelt $N(\mu, \sigma)$, og at vi velger $\mu_T \approx G/2$.
- Hva er da optimalt valg av σ_T ?
- Hvor stor percentil blir klipt?

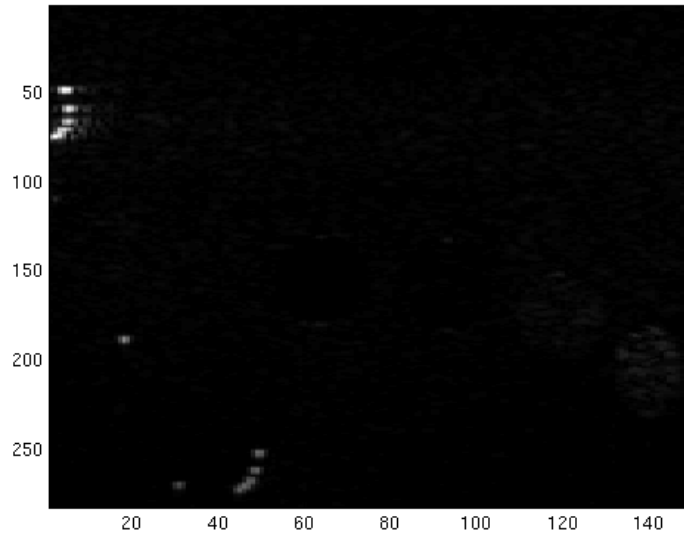


Ikke-lineær transform

- Logaritmisk skalering
 - Eks: Desibel og radarbilder, Fourier-transform
- Eksponentiell skalering
- Gamma-skalering
- Stykkevis-lineær skalering

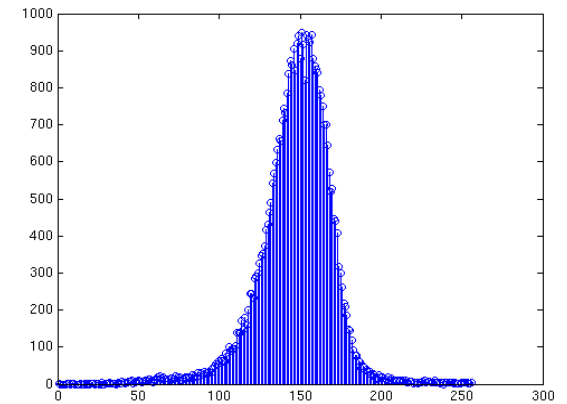
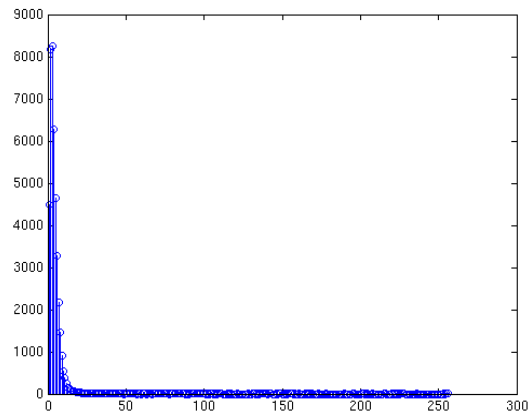
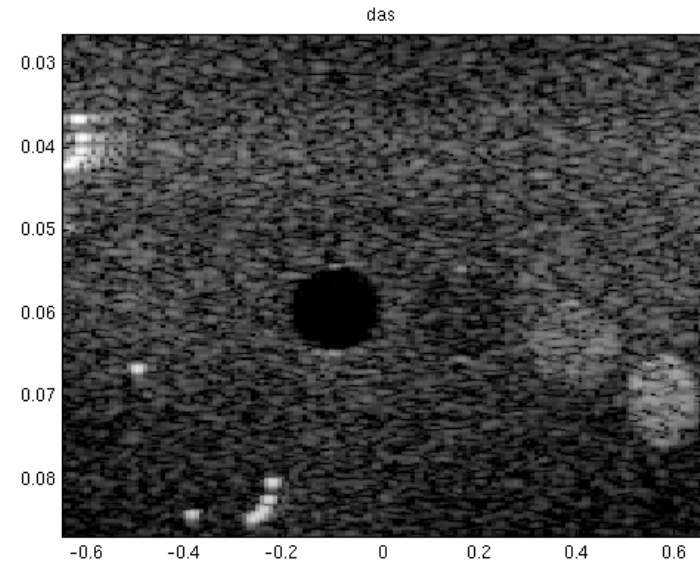
- Hva gjøres med kontrasten i de mørke og lyse delene av bildet etter slike skaleringer
 - Tegn skisse av funksjonene og se Δf mot Δg (lokalt stigningstall)

Logaritmisk mapping



$$T[i] = \log(i)$$

→

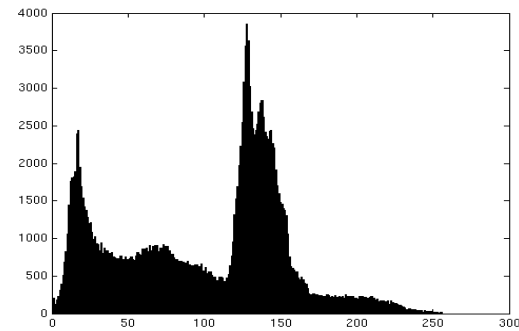
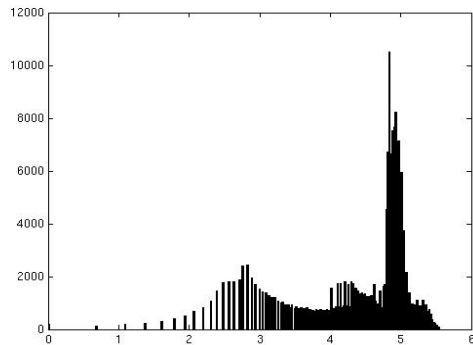


EkspONENTIELL mapping



$$T[i] \sim e^i$$

→

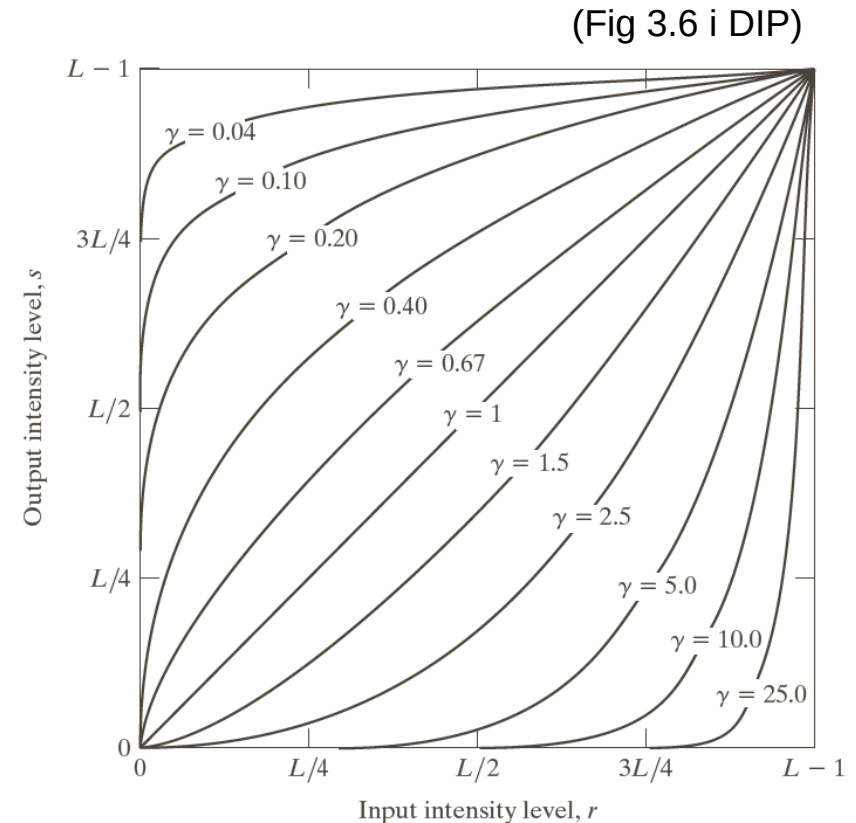


Power-law (gamma)-transformasjoner

$$T[i] = i^\gamma$$

der $T[i]$ er ut-intensiteten ved en input i .

- $\gamma < 1$: den mørke delen av skalaen strekkes ut
 - $\gamma = 1$: identitets-transform
 - $\gamma > 1$: den lyse delen av skalaen strekkes ut
- Mange bildeproduserende apparater har et slikt input/output-forhold
 - Generell kontrast-manipulasjon
 - Brukervennlig med kun én variabel
 - For mer optimal bruk av kvantiseringsnivåer
 - mer perseptuelt uniform nivåinndeling



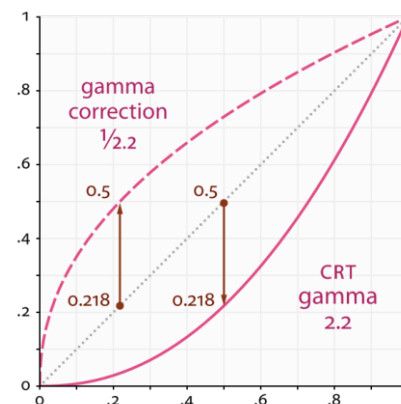
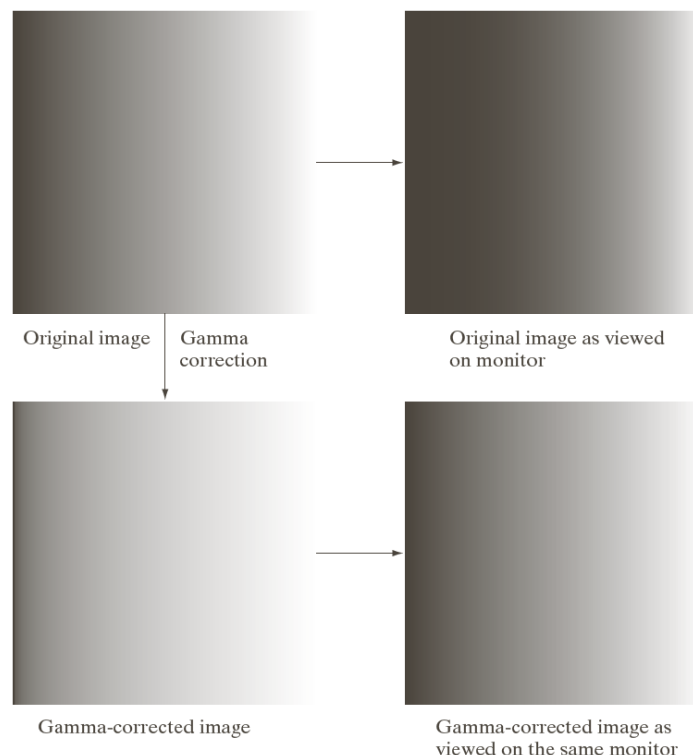
Gamma-korreksjon før fremvisning

- Anta at intensiteten i et bilde som vises på et display er gitt ved:

$$s = i^{2.5}$$

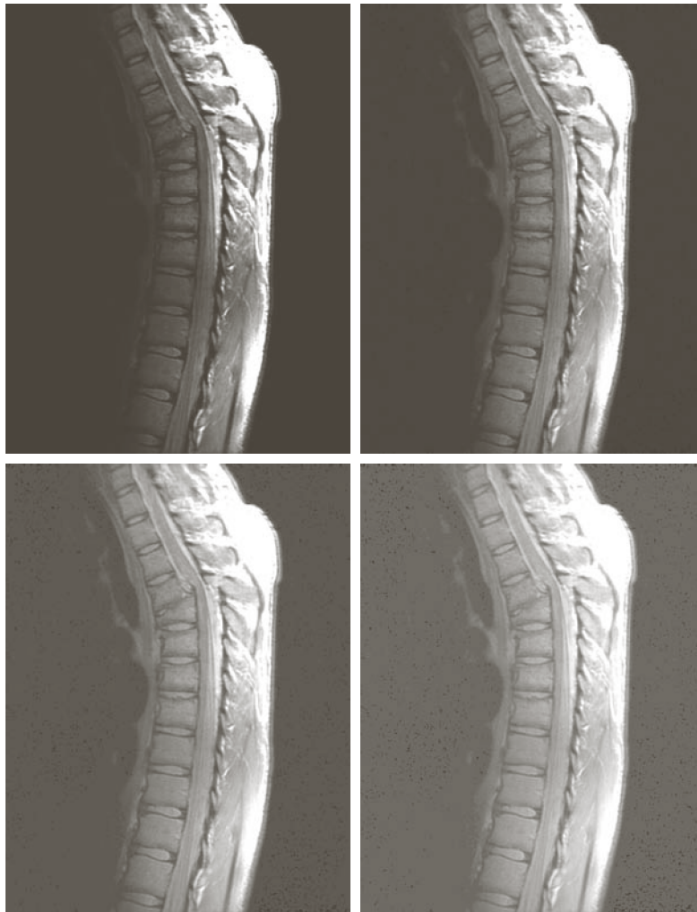
der s er ut-intensiteten ved en input i

- Vi har sett at for $\gamma > 1$ vil bildet bli mørkere enn det skal være
- Vi kan korrigere dette ved gråtonetransformen $T[i] = i^{0.4}$ før vi sender bildet til fremvisning
- Samme gjelder for scannere og printere
 - Man må kjenne eller finne parametrene til $s = (i+\epsilon)^\gamma$



Gamma-styrt bildeforbedring

(Fig 3.8 i DIP)



$\gamma = 1$ $\gamma = 0.6$
 $\gamma = 0.4$ $\gamma = 0.3$

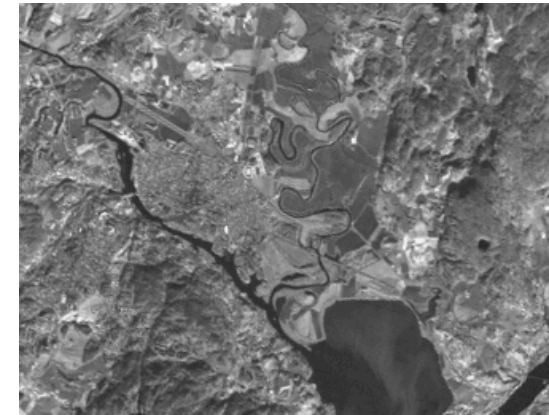
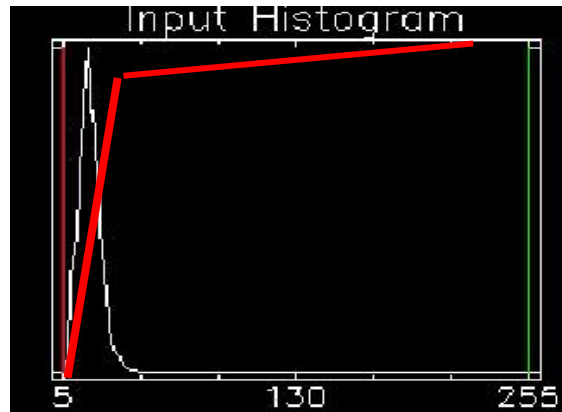
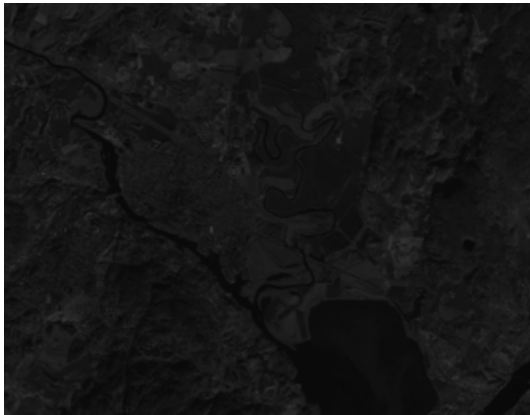
(Fig 3.9 i DIP)



$\gamma = 1$ $\gamma = 3$
 $\gamma = 4$ $\gamma = 5$

Stykkevis lineær mapping

- Brukerspesifisert stykkevis lineær mapping for å fremheve visse intervaller av gråtoneskalaen



Bit-plan-oppdeling

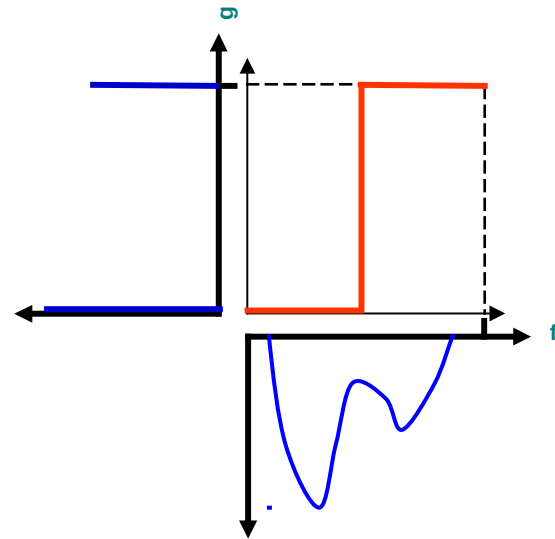
- Gir binært bilde basert på om pikslenes n -te bit er satt
- I eksemplet; kun de siste 4 bit inneholder visuell signifikans
- Kan benyttes i kompresjon
 - Kun beholde visse plan
 - Effektivt å kode binære bilder (f.eks "runlength")



8 bit inputbilde

Terskling

- Dette er et grense-tilfelle av lineær transformasjon, der alle ut-verdiene settes lik 0 for «inn-verdier» i et intervall 0-T, mens alle andre settes lik 1
- Dette gir et to-nivå (binært) ut-bilde



Implementasjon: Oppslagstabeller (LUT)

- Mål: Effektivisere implementasjonen
- Avbildningsfunksjonen utføres på alle mulige intensiteter og resultatene lagres i en tabell (LUT=look up table)
- Gråtone-avbildningen utføres så som oppslag i en tabell
- Hardware
 - LUT-operasjonen utføres på data-strømmen mellom hukommelse og display "on the fly" (på grafikkortet)
 - Innholdet i bilde-matrisen endres ikke
 - Kontrastendring ved kun å endre tabellverdiene
- Software
 - Utregning av avbildningsfunksjonen for hvert piksel blir byttet ut med enkelt tabelloppslag

Implementasjon av gråtoneoperasjoner

for x=1:N

for y=1:M

$$g(x,y)=a*\log(f(x,y))+b$$

} Direkte implementasjon

for i=0:nGreyLevels-1

$$T[i]=a*\log(f(x,y))+b$$

} Fyll inn en LUT..

for x=1:N

for y=1:M

$$g(x,y)=T[f(x,y)]$$

} .. så endring av pikselverdiene

Oppsummering

- Gråtonehistogrammer
- Lineær transform
 - Forstå effekten av parametrene **a** og **b**
- Standardisering av bilder med lineær transform
 - Fjerne effekten av variasjoner i avbildningsforhold (døgnvariasjon, lampe, støv etc)
 - Hvordan bestemme **a** og **b** for å få ønsket μ_T og σ_T
- Ikke-lineære, parametriske transformer
 - Logaritmisk, eksponentiell, "gamma", stykkevis lineær
 - Hva gjøres med kontrasten i de mørke og lyse delene av bildet etter slike transformer
 - Tegn skisse av funksjonene og se Δf mot Δg