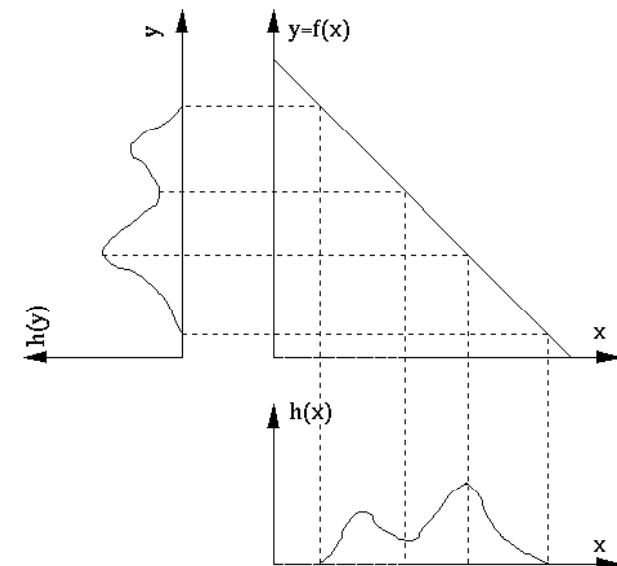

INF 2310 – Digital bildebehandling

FORELESNING 4 GRÅTONE-TRANSFORMASJONER

Fritz Albregtsen



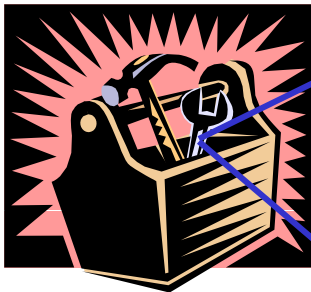
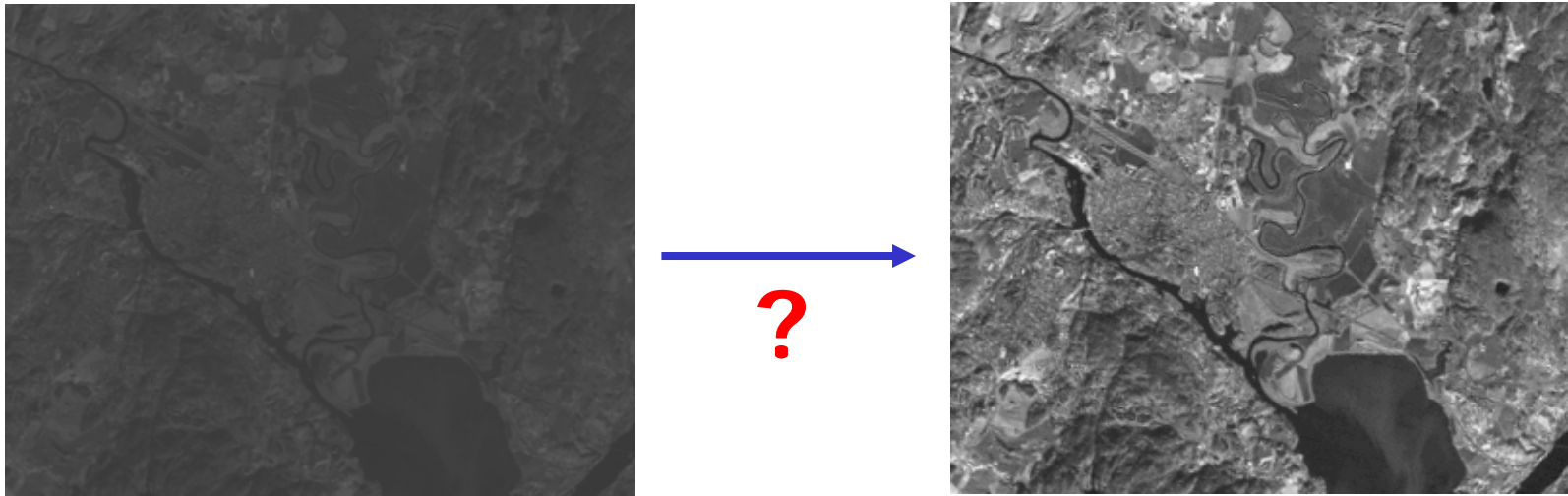
Temaer i dag

- Histogrammer
- Lineære gråtonetransformer
- Standardisering av bilder med lineær transform
- Ikke-lineære, parametriske transformeringer

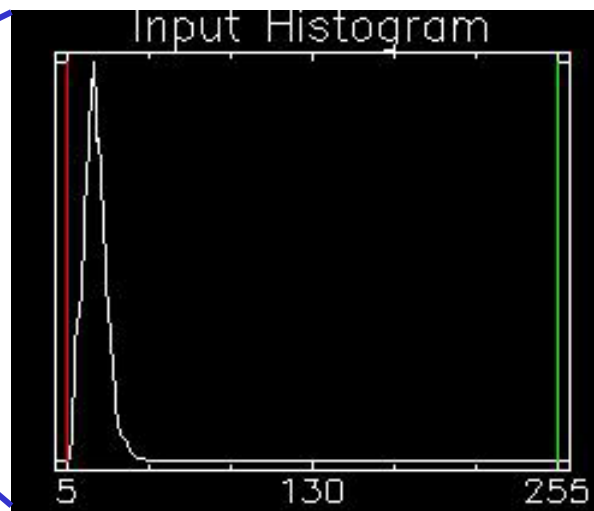
- Pensum: Kap. 3.1 - 3.2 i DIP

- Neste uke: Histogrambaserte operasjoner og lokale gråtonetransformer

Hvordan endre kontrasten i et bilde?



Matematisk
verktøykasse

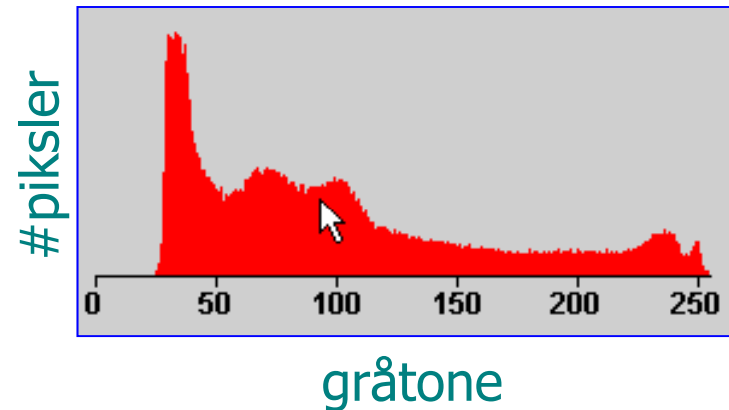


Histogrammer

- En diskret funksjon som viser antall målinger innenfor (som oftest) uniforme intervaller i et datasett
- Vi jobber med bilder og får
 - Et bilde som datasett
 - Pikkse-intensiteter som målinger
- En oversikt over hyppigheten til intensitetene i bildet
- Kan også ha histogrammer over andre parametre.

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



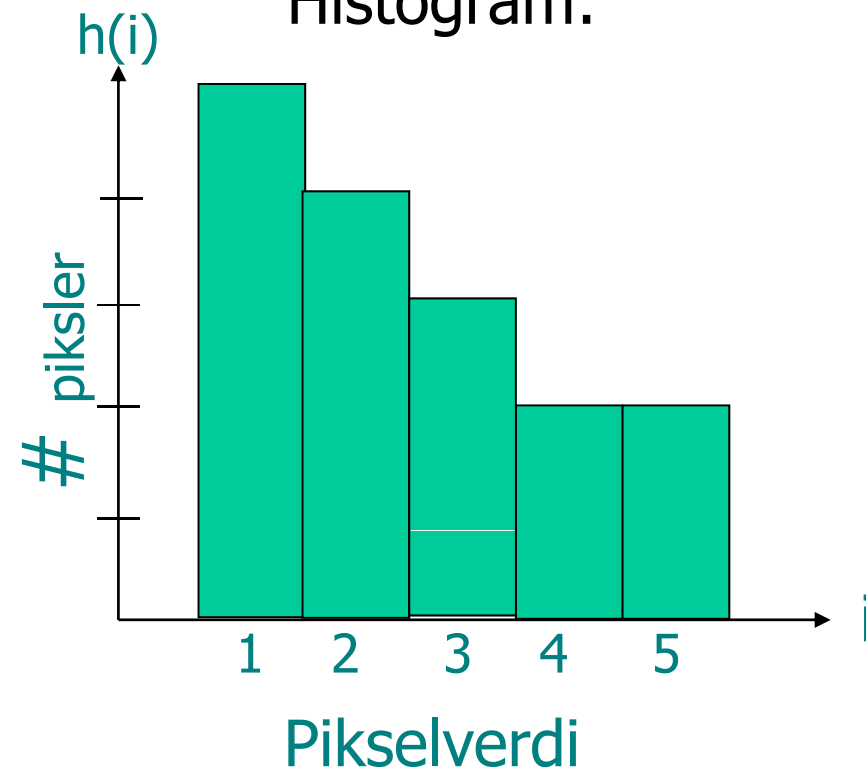
- Vi har naturligvis at
$$\sum_{i=0}^{G-1} h(i) = n \times m$$

Eksempel - histogram

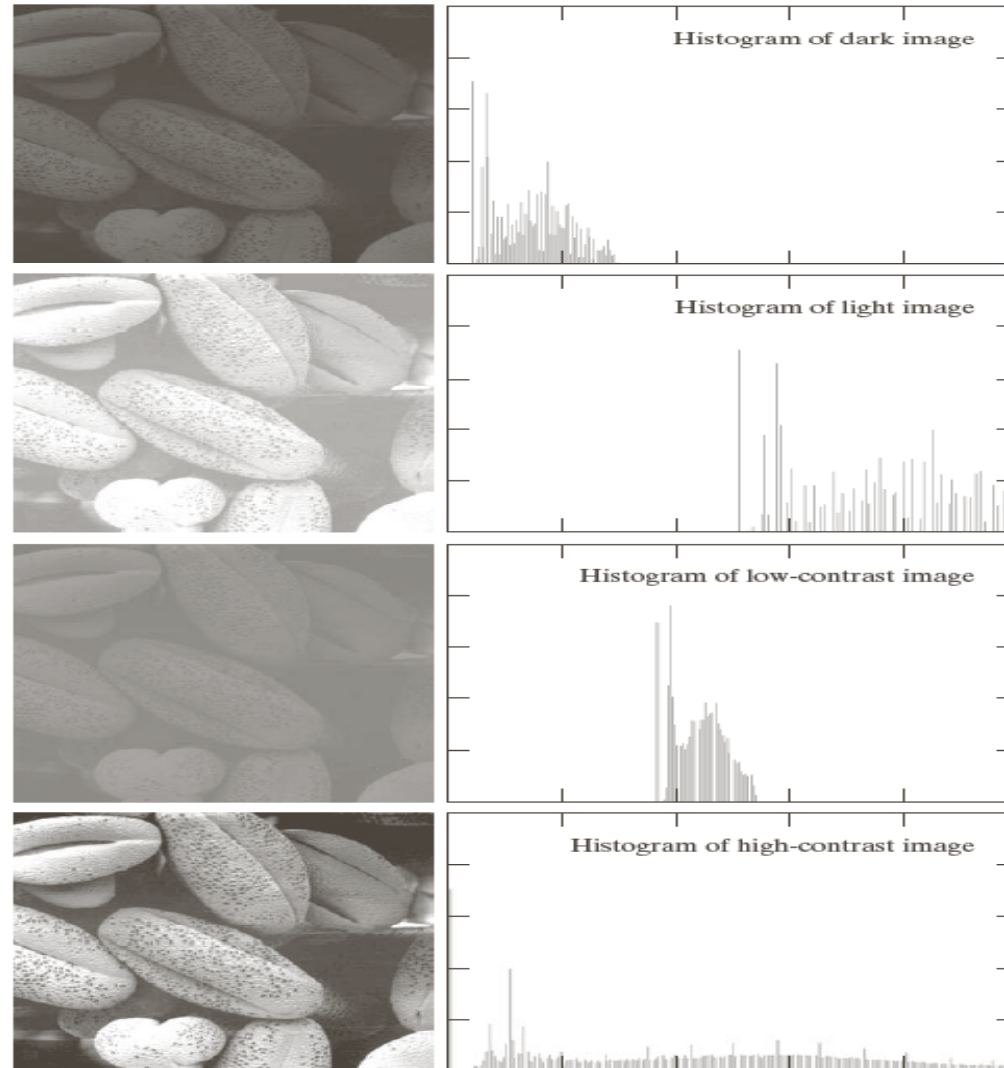
Bilde:

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

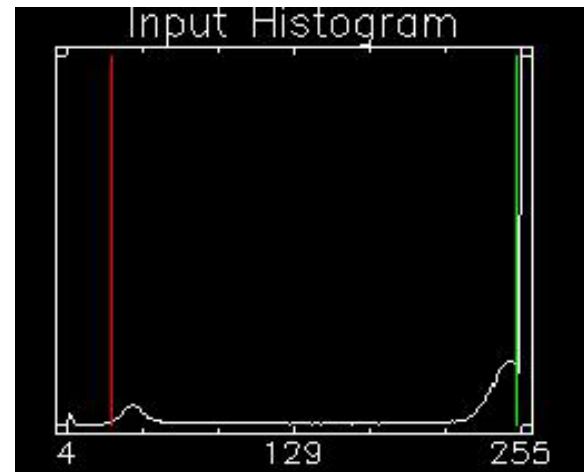
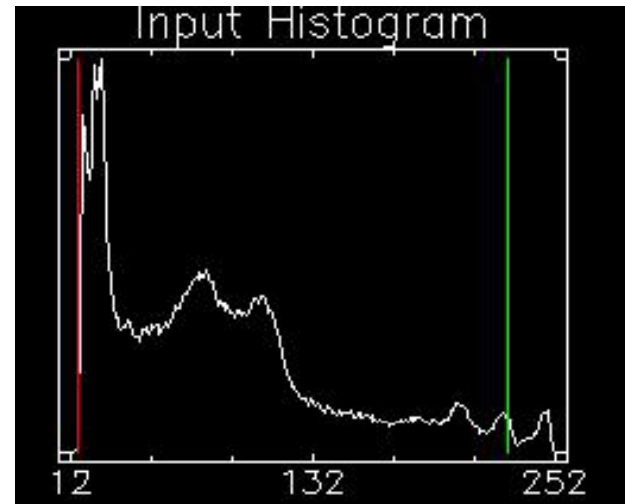
Histogram:



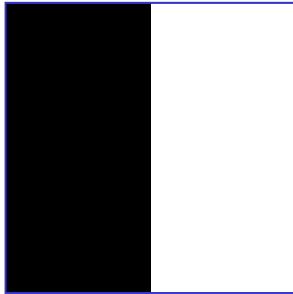
Eksempler



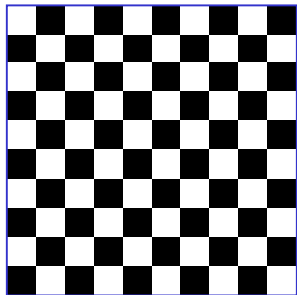
Eksempler II



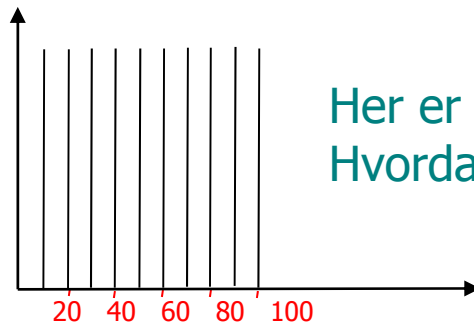
Oppgaver



Hvordan ser histogrammet ut?



Hvordan ser histogrammet ut?



Her er histogrammet.
Hvordan ser bildet ut?

Normalisert histogram

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

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

- $p(i)$ kan ses på som en **sannsynlighetsfordeling** for pikselverdiene i
 - "Uavhengig" av antall piksler i bildet
- Man kan si en del om bildet ut fra denne sannsynlighets-tetthetsfunksjonen

Kumulativt histogram

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

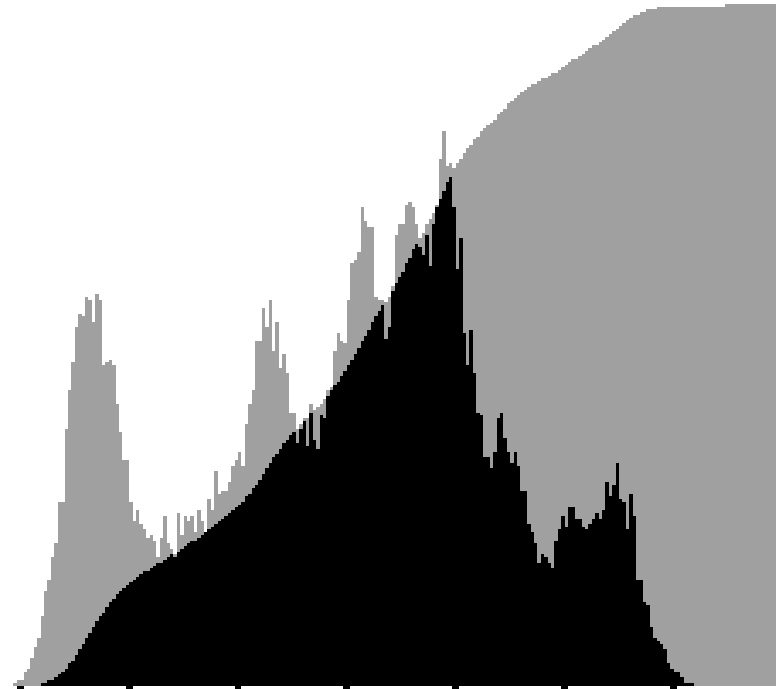
$$c(j) = \sum_{i=0}^j h(i)$$

- Normalisert kumulativt histogram:

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

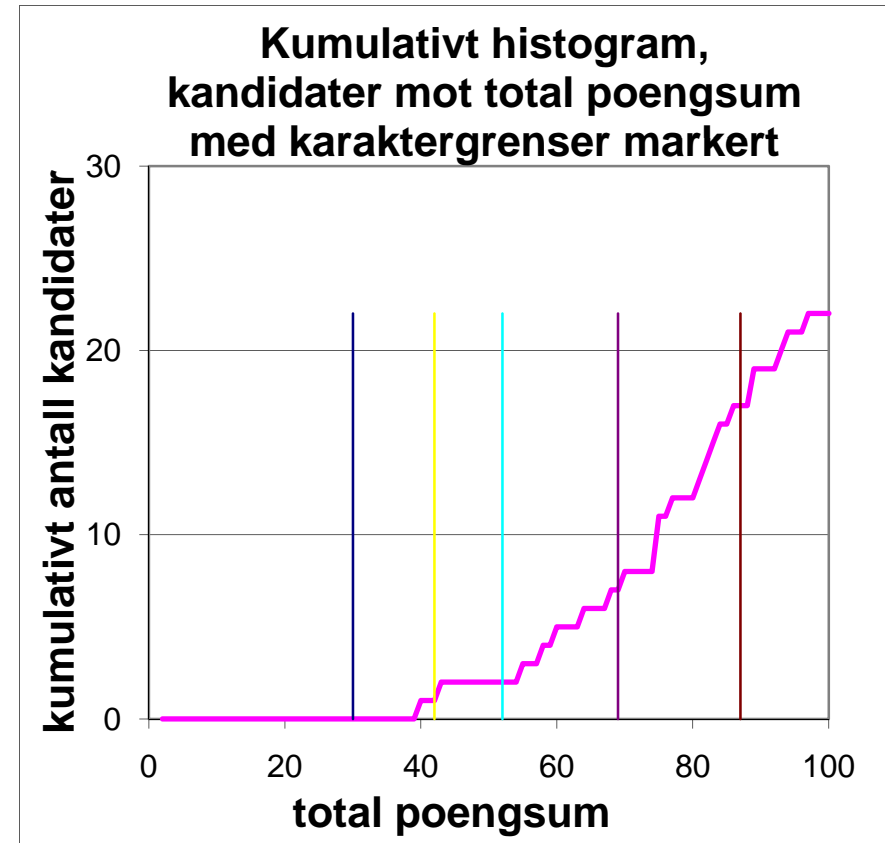
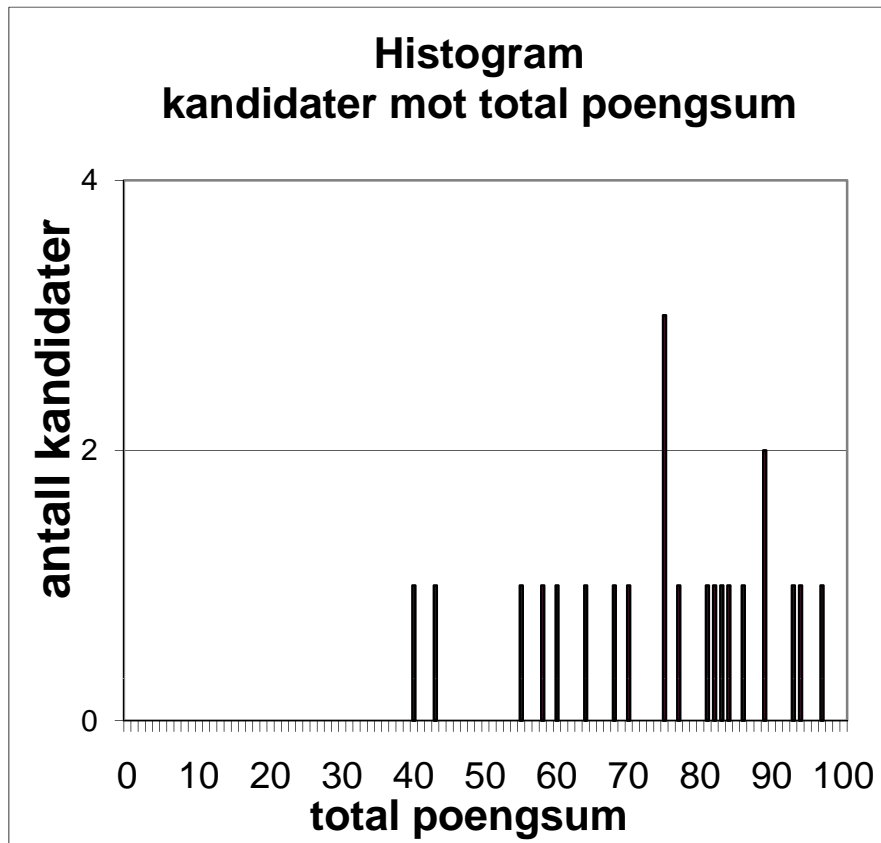
(Sannsynligheten for at en tilfeldig piksel er mindre eller lik gråtone j)

Eksempel, kumulativt histogram



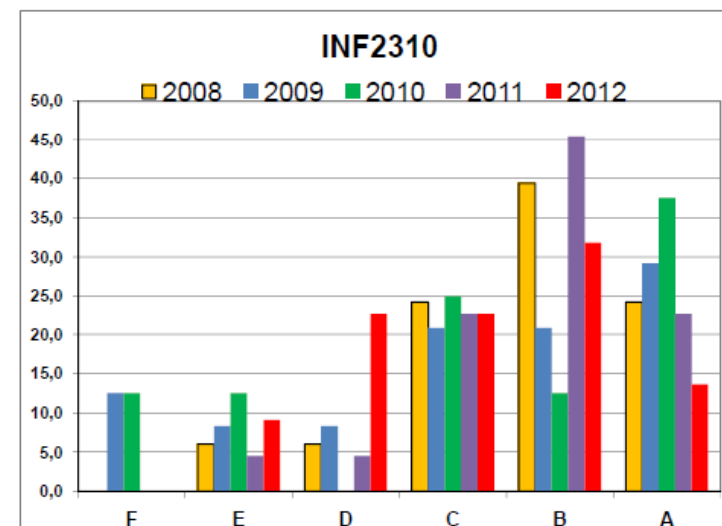
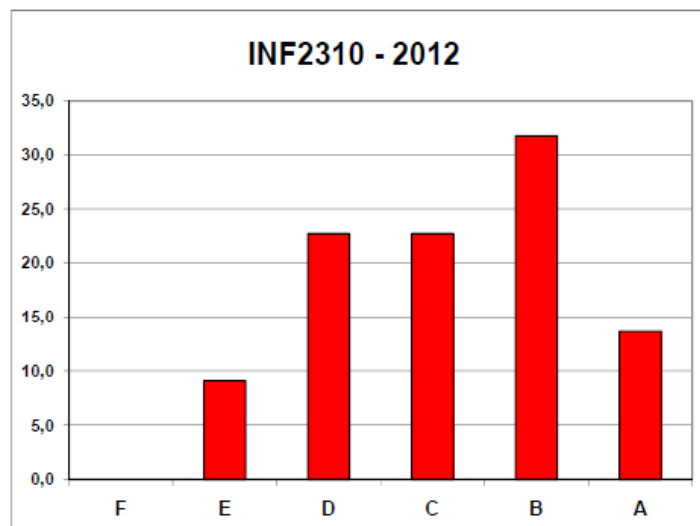
Histogram og kumulativt histogram i samme figur

Histogrammer – full oppløsning



Skalerte histogrammer – redusert oppløsning

- * Oppløsningen i histogrammer kan reduseres
 - for eksempel ved overgang fra poengsum til karakter.
- * Histogrammet kan skaleres til sum = 1 eller sum = 100%.



"Normen" er 10%, 25%, 30% 25%, 10%.

Er dere bedre enn "normen", så får dere gode karakterer.

Histogrammer av objekt-egenskaper

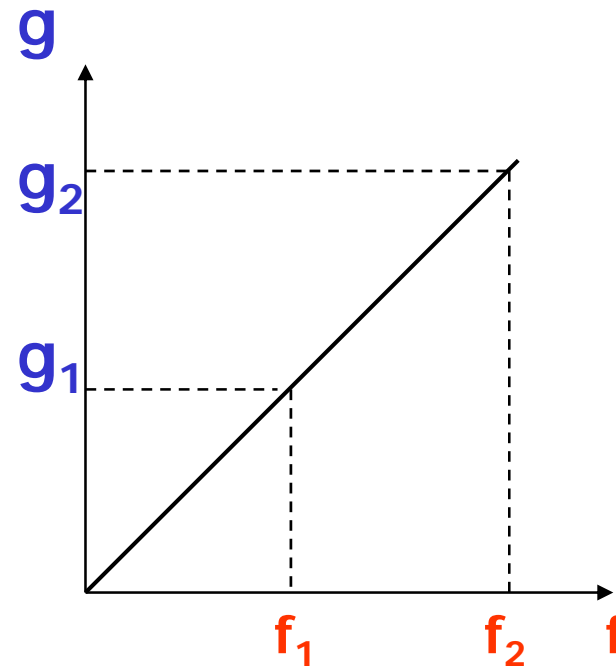
- Begrepsapparatet omkring histogrammer vil også komme til nytte i digital bildeanalyse
- Vi kan lage histogrammer over egenskaper, feks:
 - Objekt-størrelse:
 - Viser fordelingen av størrelsen på objektene, og danner grunnlag for å sette en terskel for å kunne fjerne små og uvesentlige objekter fra 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 et bilde på skjermen er intensiteten kontrollert av den tilsvarende verdien i bildematriksen
- Vi kan opprette en avbildnings-funksjon mellom de tallene som finnes i bildematriksen, v_{in} , og den intensiteten vi ønsker på skjermen, v_{out}
- For ett-båndsbilder er $v_{out} = T[v_{in}]$
- T kan være en parametrisk funksjon eller en tabell
- Ren gråtonetransformasjon, så ett og ett piksel transformeres uavhengig av nabopiksler
- **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 transformasjon er en identitetsmapping, $g=f$, vil figuren vise en rett linje gjennom origo, med stigningstall 1.
- $T[i] = i$



Lineær avbilding

- Lineær strekking

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

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

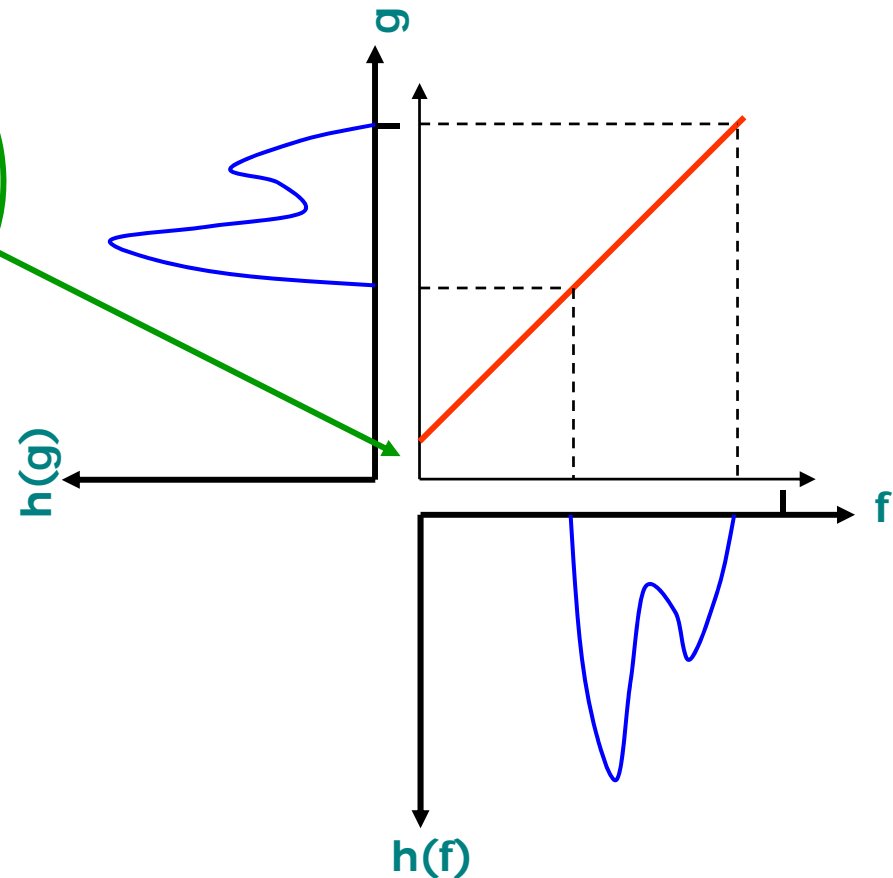
- a regulerer kontrasten, og b "lysheten"
- $a > 1$: mer kontrast
- $a < 1$: mindre kontrast
- b : flytter alle gråtoner b nivåer
- Negativer: $a = -1$, $b = \text{maxverdi for bildetype}$

Endre "lysheten" (brightness)

- Legge til en konstant b til alle pikselverdiene

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

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



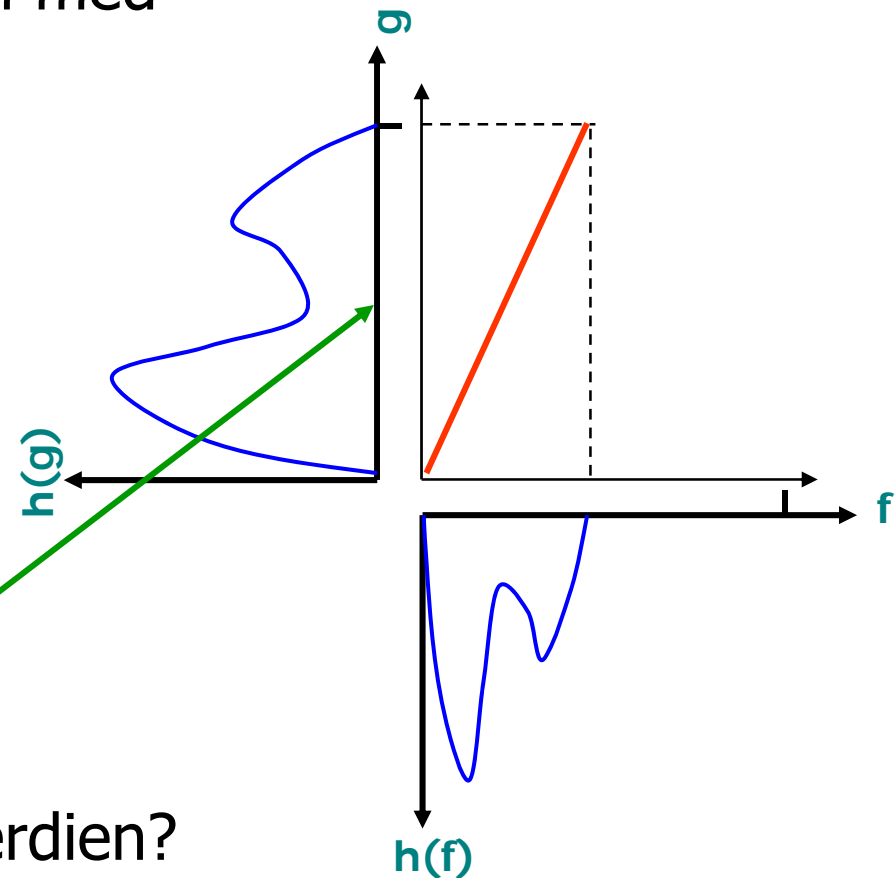
Endre kontrasten

- 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?

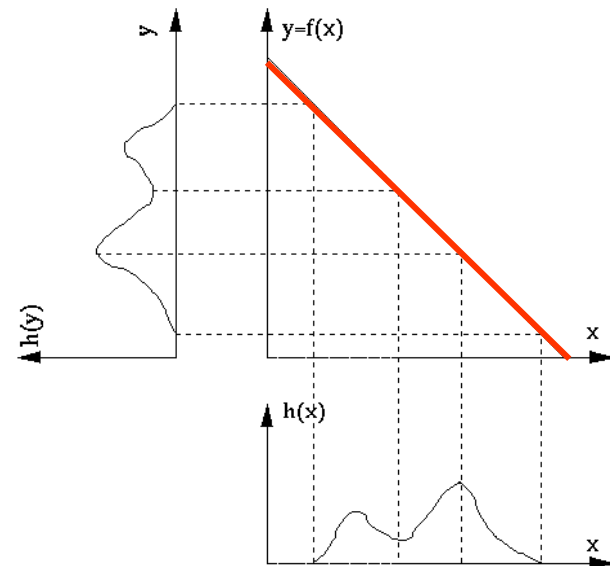


Invertert gråtonebilde

- Danner bildets "negativ" ved å sett $a=-1$ og $b=\text{maksverdien}$ (antall gråtoner = G)

$$g(x, y) = (G - 1) - f(x, y)$$

- Bildet får ikke negative verdier, men avbildningsfunksjonen har negativt stigningstall



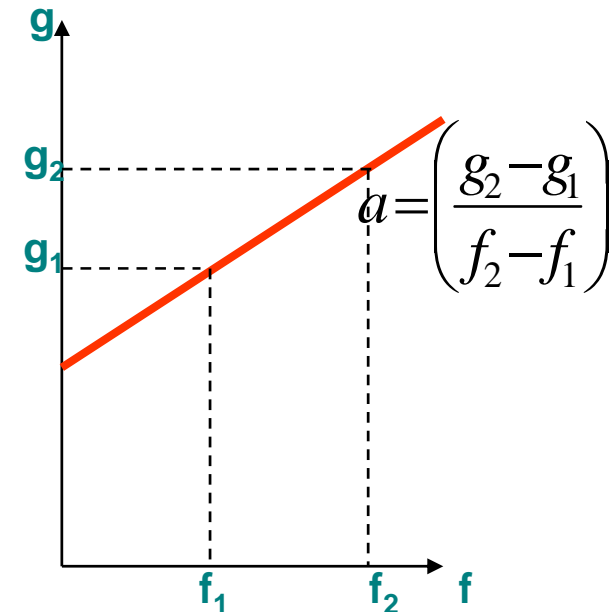
Fra gråtonenivå $[f_1, f_2]$ til $[g_1, g_2]$

- Endre intervallet $[f_1, f_2]$ til å bli $[g_1, g_2]$
- En lineær mapping fra f til g :

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

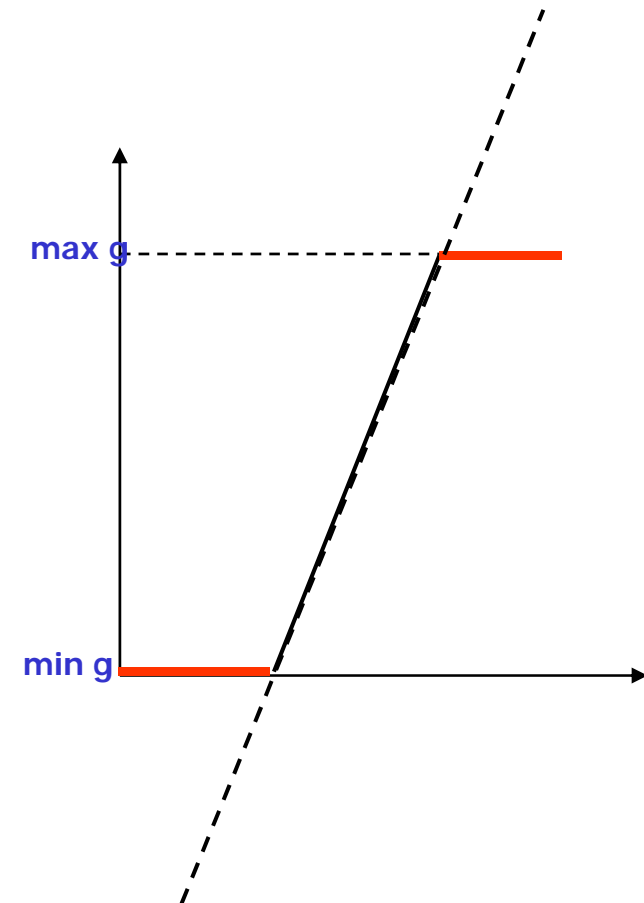
– Rett linje med stigningstall

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



Klipping etter transform

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



Standardisering av bilder

- Hensikt:
 - Sørge for at alle bildene i en serie er statistisk like (1. orden)
- Metode:
 - Justere middelveien og variansen til gråtoneverdiene i bildet ved hjelp av en lineær gråtonetransform
- Hvorfor? Fjerne effekten av
 - Døgnvariasjon i belysning
 - Aldringseffekter i lamper og detektorer
 - Akkumulering av støv på linser etc.
- Hvor:
 - Produkt-inspeksjon i industri
 - 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 bildet
 - eller fra bildets histogram, evt fra normalisert histogram

$$\begin{aligned}\mu &= \frac{1}{n \times m} \sum_{x=0}^{n-1} \sum_{y=0}^{m-1} f(x, y) \\ &= \frac{1}{n \times m} [0 \times h(0) + 1 \times h(1) + \dots + (G-1) \times h(G-1)] \\ &= \frac{1}{n \times m} \sum_{i=0}^{G-1} i h(i) = \sum_{i=0}^{G-1} i p(i)\end{aligned}$$

Hvorfor en fordel med det siste alternativet?

$$\text{der: } p(i) = \frac{h(i)}{nm}, \quad \sum_{i=0}^{G-1} p(i) = 1$$

(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

$$\begin{aligned}\sigma^2 &= \frac{1}{n \times m} \sum_{x=0}^{n-1} \sum_{y=0}^{m-1} [f(x, y) - \mu]^2 \\ &= \frac{1}{n \times m} \sum_{i=0}^{G-1} h(i) [i - \mu]^2 \\ &= \sum_{i=0}^{G-1} p(i) [i - \mu]^2 \\ &= \sum_{i=0}^{G-1} i^2 p(i) - \left(\sum_{i=0}^{G-1} i p(i) \right)^2\end{aligned}$$

Justering av μ og σ^2

- Gitt inn-bilde med middelerdi μ og varians σ^2
- Anta en lineær gråtone-transform $T[i]=ai+b$
- Ny middelerdi 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 riktig μ_T og σ_T^2

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

Eksempel 1: Justering av σ

- Vil beholde middelveiden, slik at

$$\mu_T = \mu,$$

men ønsker ny σ_T .

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

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

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

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 $T[i]=ai+b$:

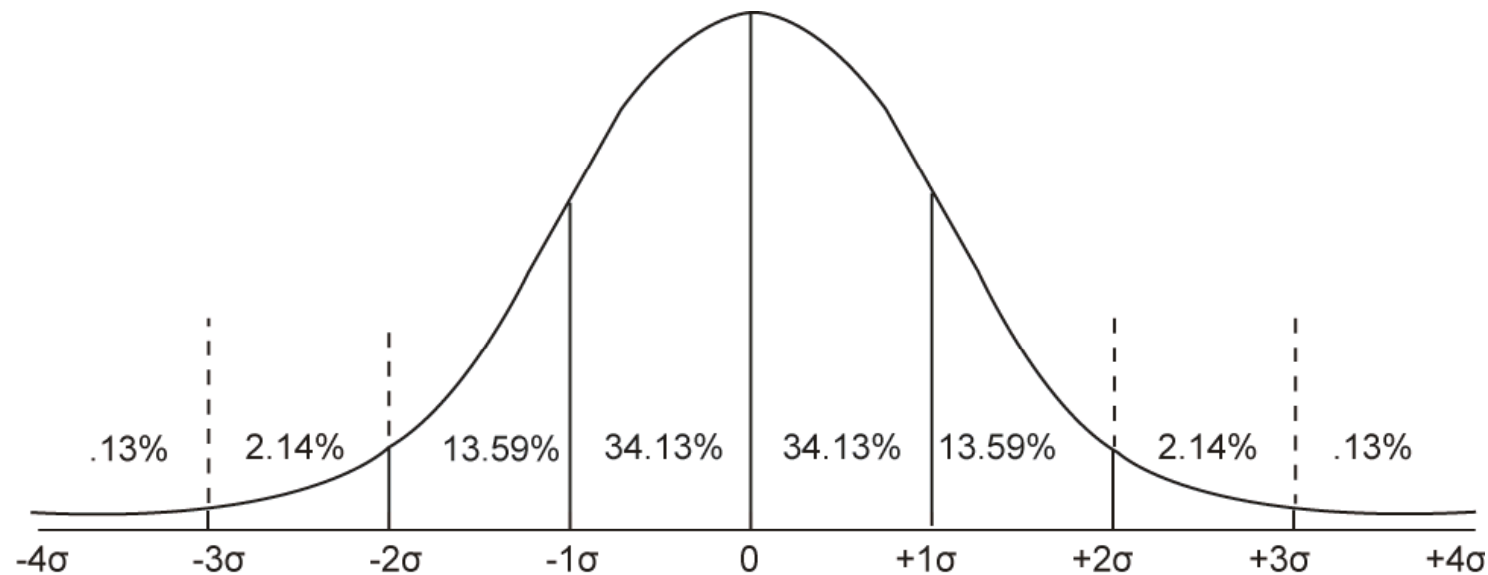
$$a = \frac{\sigma_T}{\sigma}, \quad b = \mu_T - a\mu = \mu_T - \mu \frac{\sigma_T}{\sigma}$$

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

- 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?



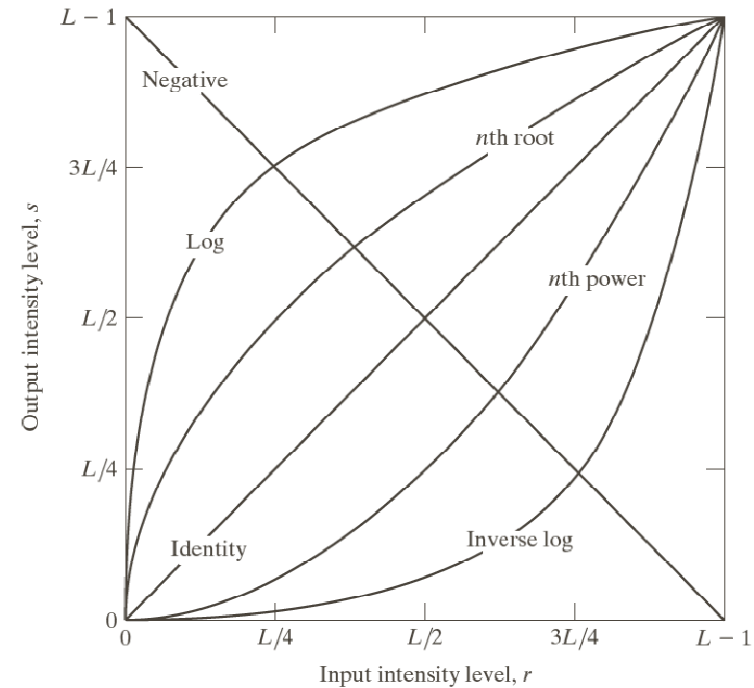
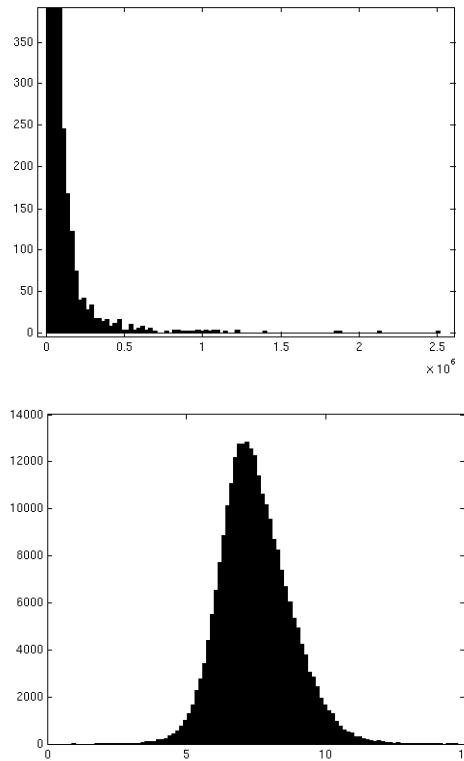
Ikke-lineær transform

- 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

Logaritmiske transformasjoner

- Hvilken av transformasjonene til høyre er brukt her?



(Fig 3.3 i DIP)

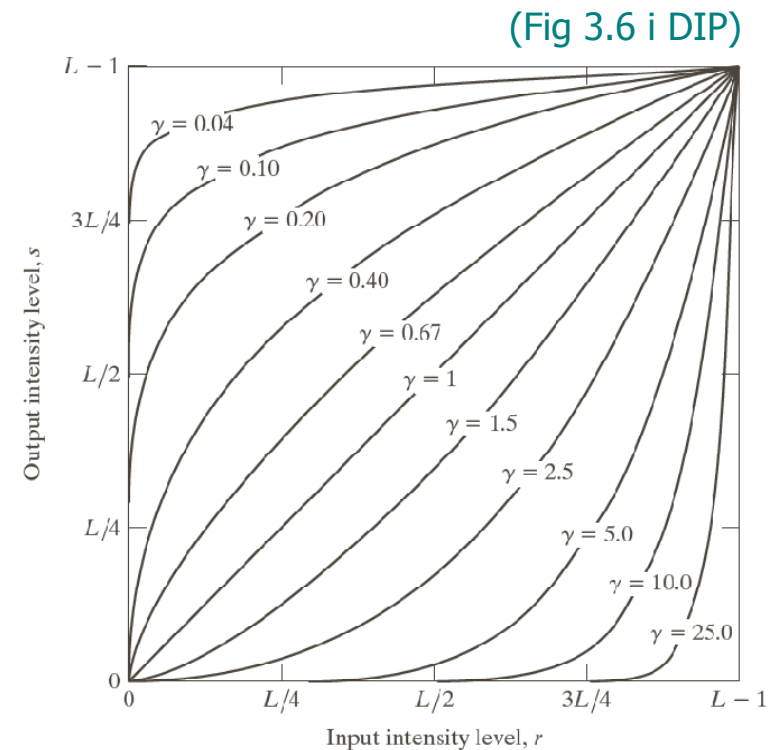
Power-law (gamma)-transformasjoner

- Mange bildeproduserende apparater har et input/output-forhold som kan beskrives som:

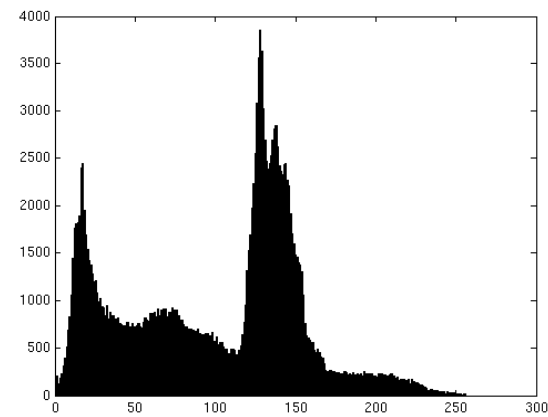
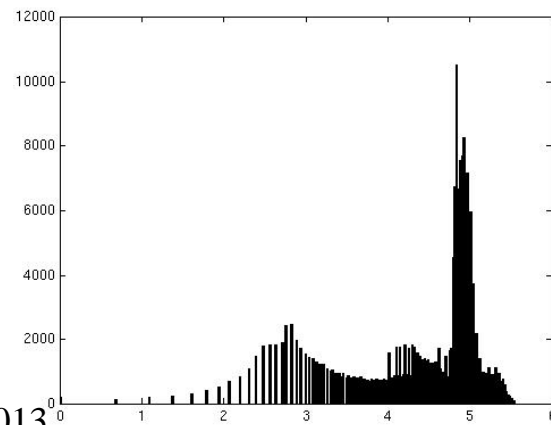
$$s = ci^\gamma$$

der s er ut-intensiteten ved en input i

- Kan korrigeres ved gråtonetransformen $T[i] = i^{1/\gamma}$
- Generell kontrast-manipulasjon
 - Brukervennlig med kun én variabel



EkspONENTIELL mapping



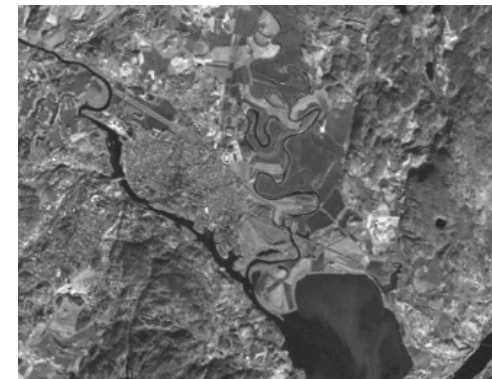
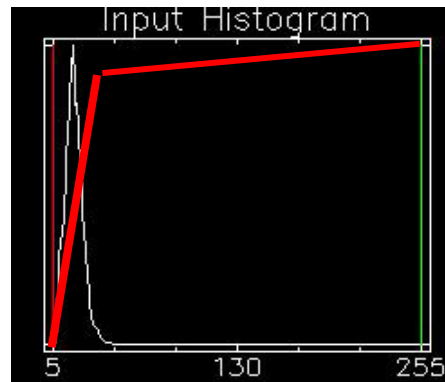
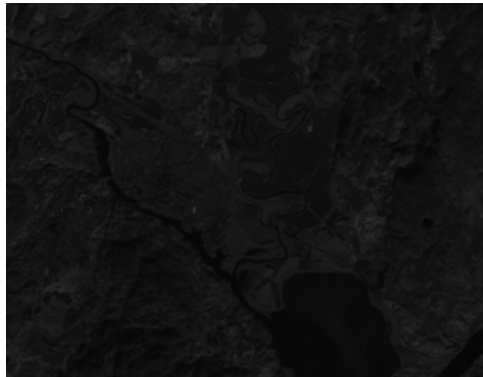
12.02.2013

INF2310

34

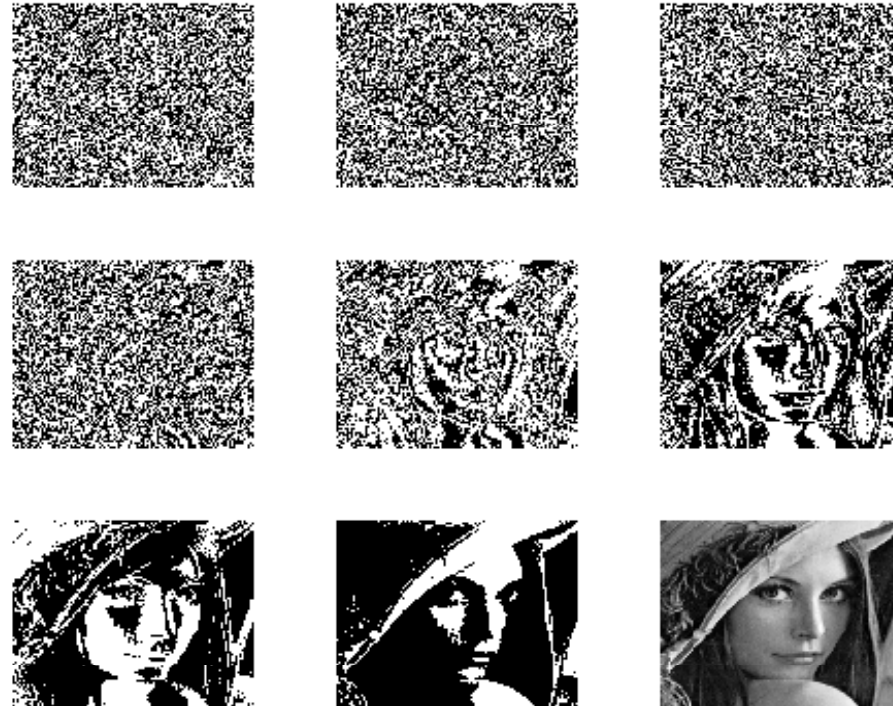
Stykkevis lineær mapping

- Brukerspesifisert stykkevis lineær mapping for å fremheve visse intervaller



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 bit-plan
 - Effektivt å kode binære bilder (f.eks "runlength")



Terskling

- Dette er et grense-tilfelle av lineær transformasjon, der alle ut-verdiene g settes lik 0 for inn-verdier f i et intervall $0-T$, mens alle andre ut-verdier 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=0:width-1  
  for y=0:height-1  
    g(x,y)=a*f(x,y) + b
```

} direkte implementasjon

```
for g=0:nGreyLevels-1  
  T[g]=a*g+b
```

} ved bruk av LUT

```
for x=0:width-1  
  for y=0:height-1  
    g(x,y)=T[f(x,y)]
```

} endring av pikselverdiene

Oppsummering

- Gråtonehistogrammer
- Lineær transform
 - Forstå effekten av parametrene **a** og **b**
- Standardisering av bilder med lineær transform
 - Fjerner 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 skaleringer
 - Tegn skisse av funksjonene og se på Δf mot Δg