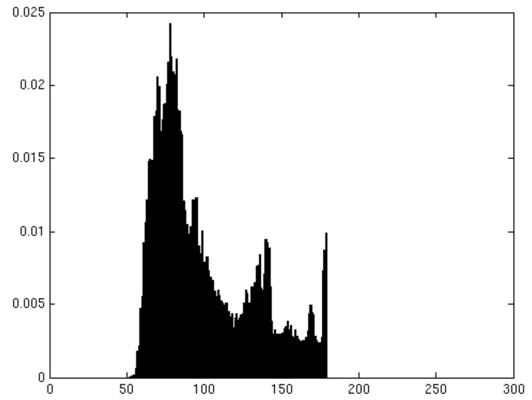
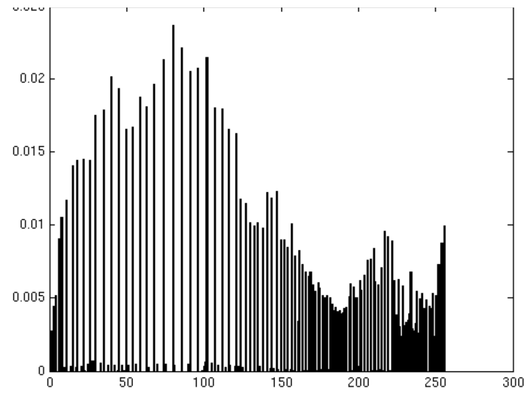
INF 2310 – Digital bildebehandling

FORELESNING 5

HISTOGRAM-TRANSFORMASJONER

Fritz Albregtsen





Temaer i dag

- Histogramtransformasjoner
  - Histogramutjevning
  - Histogramtilpasning
- Standardisering av histogram for billedserier
- Litt om histogramtransformasjoner i fargebilder
- Lokal gråtone-transformasjon
- Pensum: Hovedsakelig 3.3 i DIP
- Neste uke: Naboskapsoperasjoner, konvolusjon, filtrering.

Repetisjon av histogrammer I

- Gråtonehistogram:

  h(i) = antall piksler i bildet med pikselverdi i

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

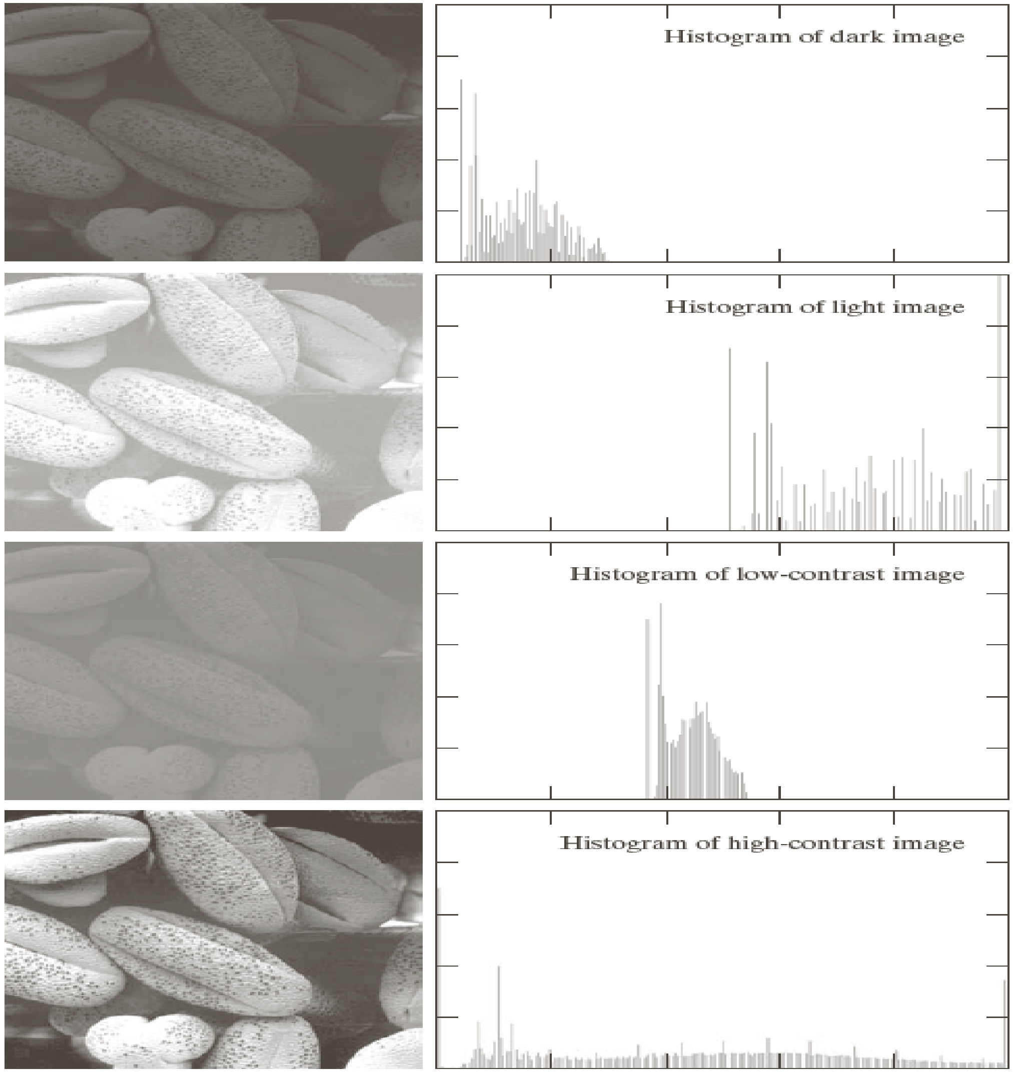
- Det normaliserte histogrammet

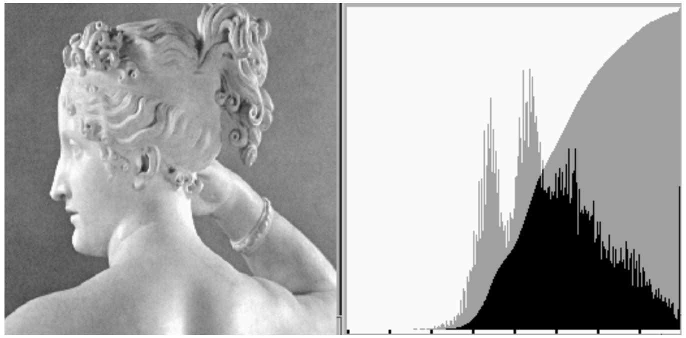
$$p(i) = \frac{h(i)}{n \times m}, \quad \sum_{i=0}^{G-1} p(i) = 1$$

- Det kumulative histogrammet

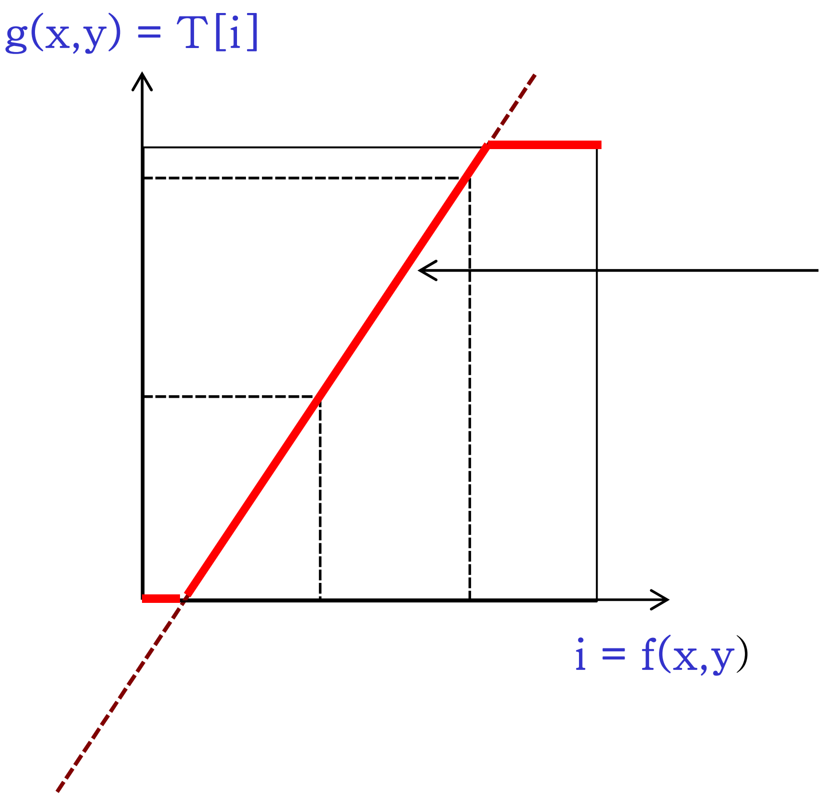
$$c(j) = \sum_{i=0}^{j} h(i)$$

Repetisjon av histogrammer II





Repetisjon av gråtonetransform



Forrige uke:

T[i] gitt som parametrisk funksjon.

Feks en linje i (f,g) planet:

T[i] = ai + b

NB: Klipping av verdier utenfor [0,G-1]

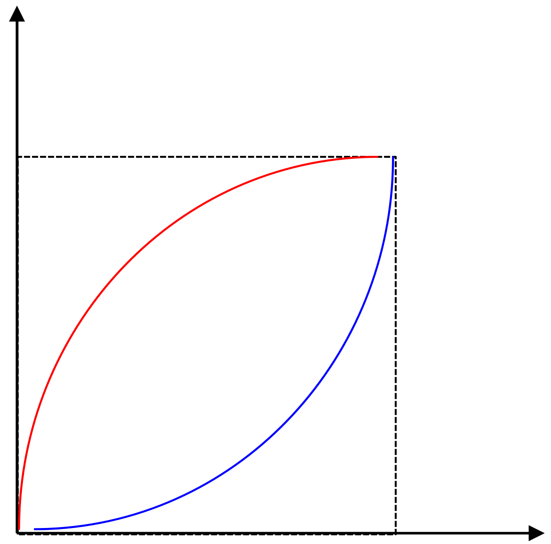
I dag:

Gitt et bilde, finn T[i] ved å spesifisere ønsket histogram.

Ikke-lineære transformer

- Vi har sett at logaritmiske og eksponensielle transformer endrer kontrasten i ulike deler av gråtoneskalaen.

- Kan vi oppnå noe av det samme med histogrammer?



Histogramutjevning (histogram equalization)

- Maksimal kontrast:
  Alle pikselverdier like sannsynlige
  - Histogrammet er uniformt (flatt)
- Ønsker en transformasjon av bildet slik at det transformerte bildet har uniformt histogram
  - Dvs. at bildet har like mange piksler for hver gråtone
- Tilnærmer ved å flytte på histogramsøyler
- Trenger en oversikt over
  hvor hver søyle skal flyttes: T[i]

Gråtonetransformasjon

- Trenger en transform
  s = T(i) som tilfredsstiller:

1) T(i) er monotont økende, dvs

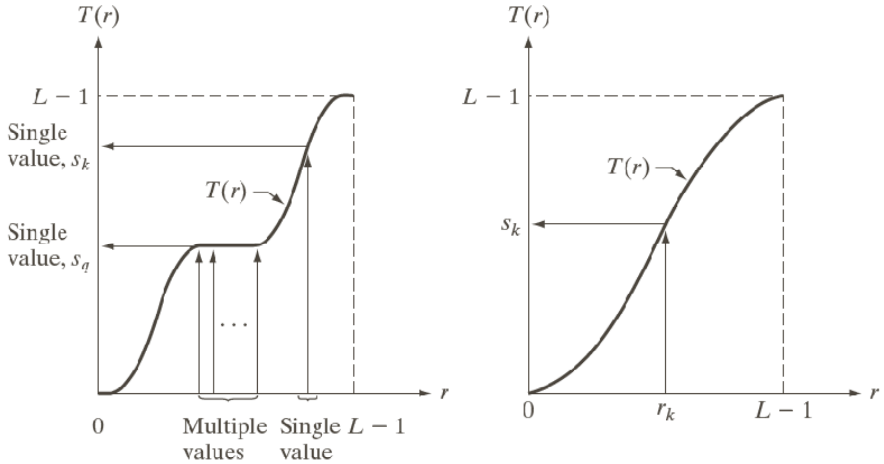
   $T(r_2) \geq T(r_1)$ hvis $r_2 > r_1$.

2) $0 \leq T(i) \leq G-1$

- Hvis den inverse transformen skal være veldefinert, må 1) endres til

1) $T(r_2) > T(r_1)$ hvis $r_2 > r_1$

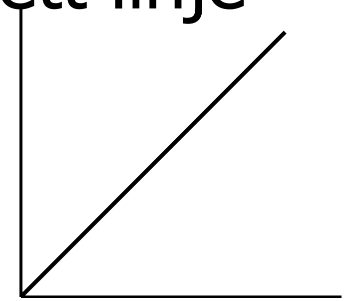
- Det siste trenger vi til histogramspesifikasjon.



Hvis transformen slår sammen to gråtoner, kan vi ikke finne tilbake til de originale gråtonene.

Et hint om en løsning:

- Hvis et bilde har uniformt histogram,
  så vil det kumulative histogrammet være
  tilnærmet en rett linje



=> Vi må finne en flytting av søylene som gir oss et kumulativt histogram som ligner mest mulig på en rett linje.

- Store mellomrom mellom høye søyler, og lite mellomrom der vi har lave søyler **=>** en transform med høyt stigningstall hvor det er mange piksler, og lavt stigningstall hvor det er få piksler
- Det **kumulative histogrammet** har akkurat disse egenskapene
- Histogramutjevnings-transformen, T[i], er gitt ved det skalerte kumulative histogrammet til innbildet.

Algoritme for histogramutjevning

- For et n×m bilde med G gråtoner:
  - Lag array p, c og T av lengde G med initialverdi 0
- Finn bildets normaliserte histogram
  - Gå igjennom bildet piksel for piksel.
  - Hvis piksel har intensitet i, la p[i]=p[i]+1
  - Deretter skalér, p[i] = p[i]/(n*m),   i=0,1,…,G-1
- Lag det kumulative histogrammet c
  - c[0] = p[0], c[i] = c[i-1]+p[i],   i=1,2,….,G-1
- Sett inn verdier i transform-array T

  T[i] = Round((G-1)*c[i]),   i=0,1,……,G-1

- **Gå igjennom bildet piksel for piksel,**

  Hvis inn-bildet har intensitet i,

  sett intensitet i ut-bildet til s=T[i]

Eksempel – histogramutjevning-1

- Tabell over pikselverdier, histogram, normalisert histogram, normalisert kumulativt histogram og histogram-transform T[i] for et 64x64 piksels 3-bits bilde, der transformen er gitt ved

$$T(i) = \text{Round}[(G-1)*c(i)]$$

i	h(i)	p(i)	C(i)	T[i]
0	790	0,19	0,19	1
1	1023	0,25	0,44	3
2	850	0,21	0,65	5
3	656	0,16	0,81	6
4	329	0,08	0,89	6
5	245	0,06	0,95	7
6	122	0,03	0,98	7
7	81	0,02	1,00	7

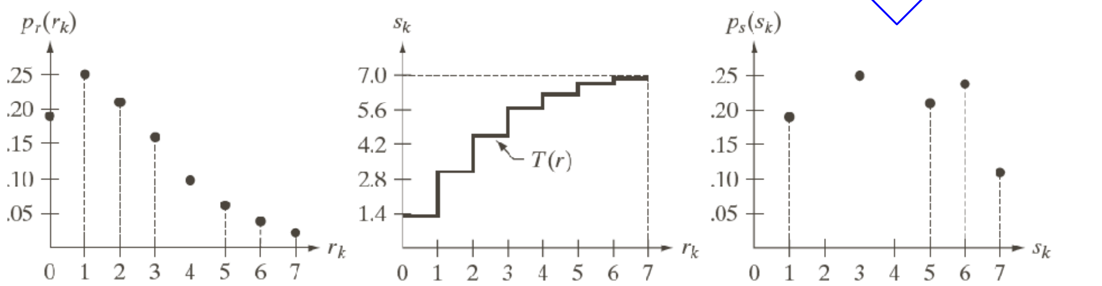
- Transformen T[i] er en “Look-Up-Table” (LUT).

Eksempel – histogramutjevning-2

- Hvordan blir ut-histogrammet $p_s(i)$, gitt inn-histogrammet $p_r(i)$ og LUT'en?

i	T[i]
0	1
1	3
2	5
3	6
4	6
5	7
6	7
7	7

$p_s(0) = 0$
$p_s(1) = p_r(0) \approx 0.19$
$p_s(2) = 0$
$p_s(3) = p_r(1) \approx 0.25$
$p_s(4) = 0$
$p_s(5) = p_r(2) \approx 0.21$
$p_s(6) = p_r(3) + p_r(4) \approx 0.24$
$p_s(7) = p_r(5) + p_r(6) + p_r(7) \approx 0.11$

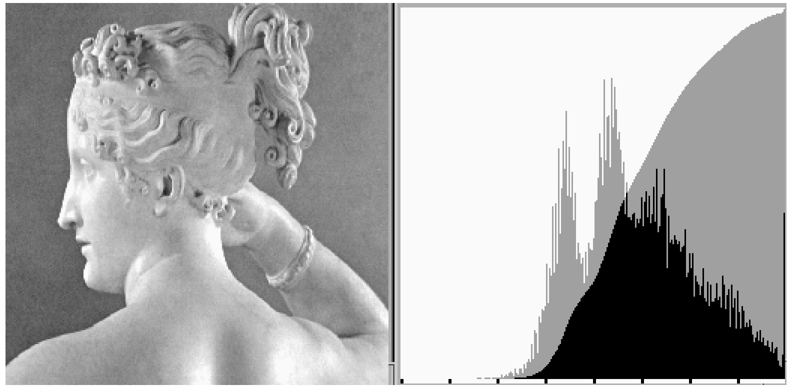


a b c

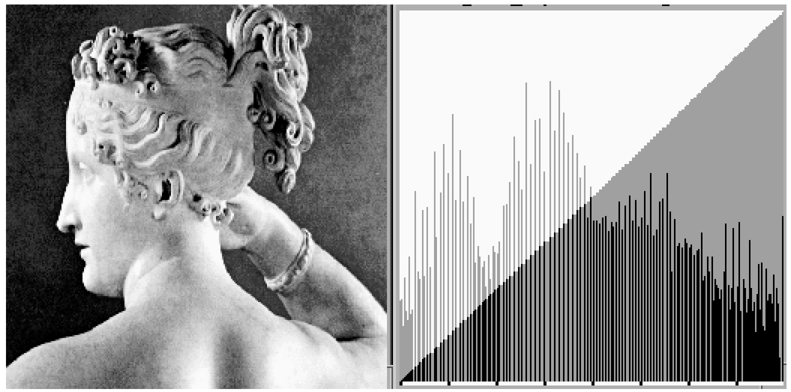
FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

Histogramutjevning, forts

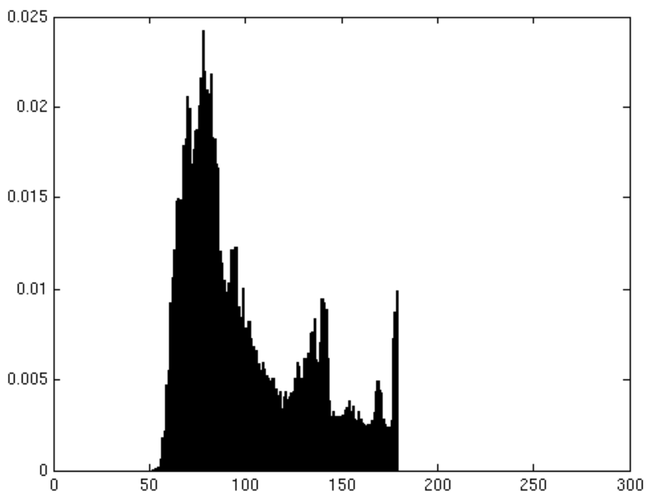
- Det resulterende histogrammet ser ikke flatt ut, men det kumulative histogrammet er en rett lineær rampe.

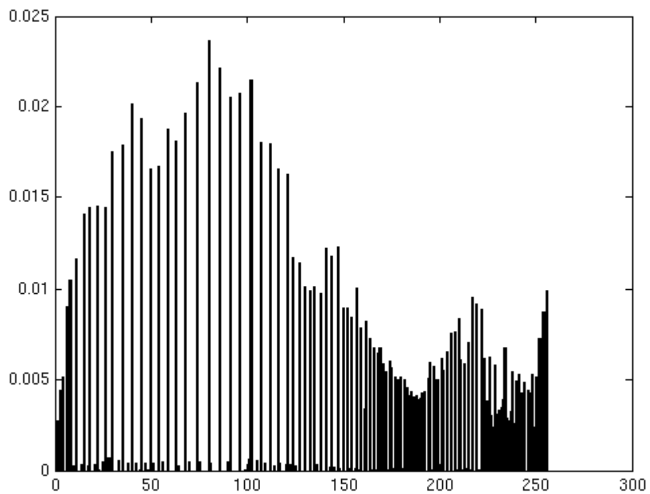


- Søylene kan ikke splittes for å tilfredstille et flatt histogram.

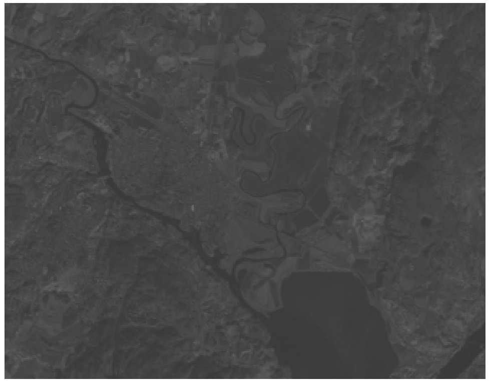


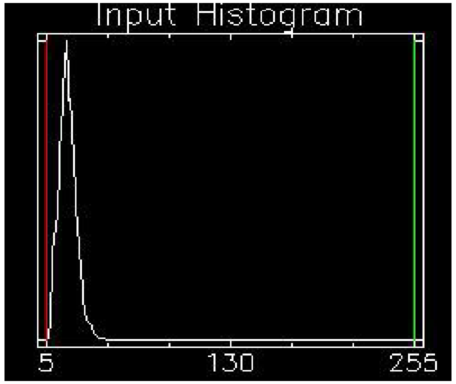
Eksempel 1 - histogramutjevning

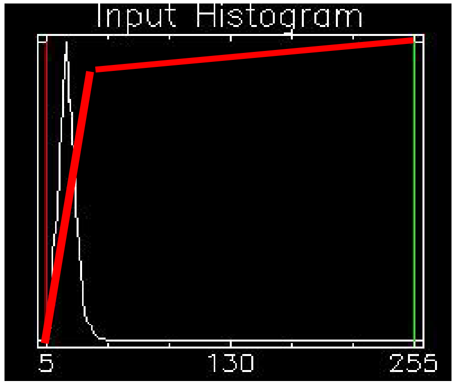


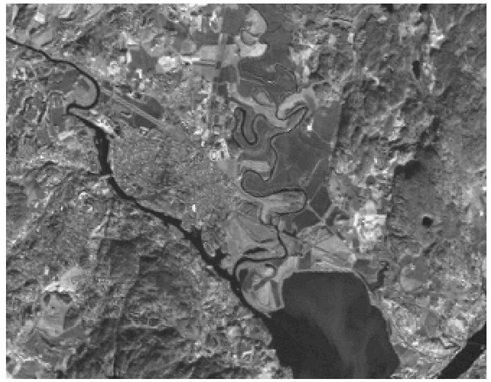


Eksempel 2 - histogramutjevning









Histogramutjevning gir ikke alltid det beste resultatet!

Histogramutjevning

- Histogramutjevning kan flytte histogramsøyler
- Kan også slå sammen søyler
- Men kan ikke splitte søyler

- Gir bare tilnærmet flatt histogram i utbildet
- Utbildets kumulative histogram stiger lineært

Q: Hva er resultatet av en histogramutjevning av et allerede histogramutjevnet bilde?

Histogramtilpasning

- Histogramutjevning gir tilnærmet flatt histogram
- Kan hende at vi ønsker å spesifisere annen form på resultathistogrammet:

1. Gjør histogramutjevning på innbildet, finn $s=T(i)$
2. Spesifiser ønsket nytt histogram $g(z)$
3. Finn den transformen T_g som histogramutjevner $g(z)$ og inverstransformen T_g^{-1}
4. Inverstransformer det histogramutjevnede bildet fra punkt 1 ved $z=T_g^{-1}(s)$

Algoritme - histogramspesifikasjon

- Finn normalisert histogram, $p_r(i)$, for inputbildet, f(r).
- Lag det kumulative histogrammet c(i).
- Sett s(i) = Round((G-1)*c[i]),   i=0,1,......,G-1

- Gitt ønsket histogram, $p_z(i)$, for bildet g(z).
- Beregn kumulativt spesifisert histogram, skalér, avrund til nærmeste heltall i [0,G-1], og lagre $G_z(q)$.

- For i=0,1,…,G-1, finn q slik at $G_z(q)$ er nærmest mulig s(i), og lagre alle disse matchene i en array $T_{ny}(i)$.
  - Hvis flere q gir samme match, velg den minste.
- Kombiner så de de to transformene til en ny mapping.

Eksempel-histogramspesifikasjon

- Gitt:

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

og spesifisert histogram p_z :

z_q	Specified $p_z(z_q)$	Actual $p_z(z_k)$
$z_0 = 0$	0.00	0.00
$z_1 = 1$	0.00	0.00
$z_2 = 2$	0.00	0.00
$z_3 = 3$	0.15	0.19
$z_4 = 4$	0.20	0.25
$z_5 = 5$	0.30	0.21
$z_6 = 6$	0.20	0.24
$z_7 = 7$	0.15	0.11

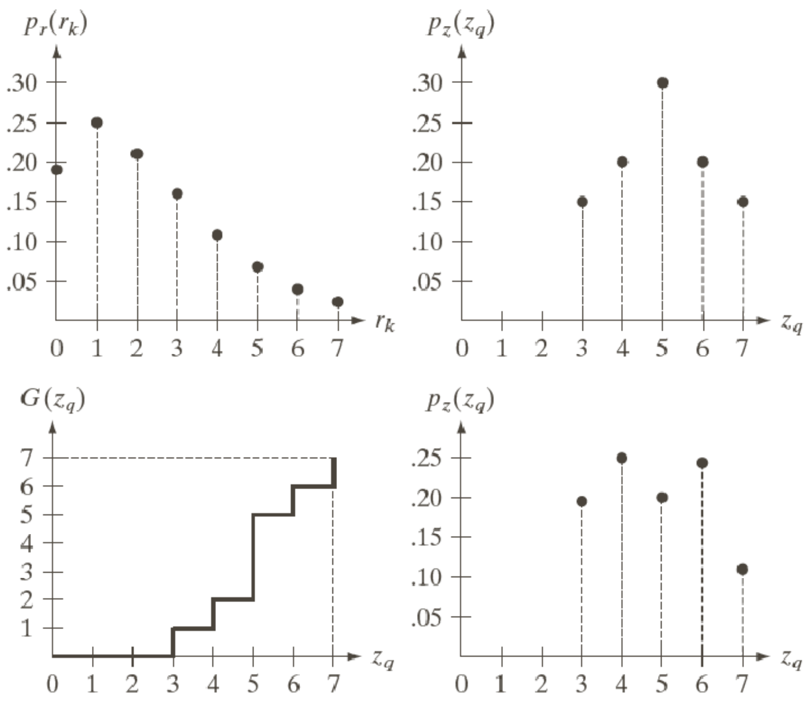
- Vi fant at T(0)=1, T(1)=3, T(2)=5, T(3)=T(4) = 6, T(5)=T(6)=T(7) =7.
- Regn ut: G(0)=G(1)=G(2)=0.0, G(3)=1.05, G(4) = 2.45, G(5)=4.55, G(6)=5.95, G(7)= 7.00;
- Avrundet til: 0,0,0,1,2,5,6,7.

z_q	$G(z_q)$
$z_0 = 0$	0
$z_1 = 1$	0
$z_2 = 2$	0
$z_3 = 3$	1
$z_4 = 4$	2
$z_5 = 5$	5
$z_6 = 6$	6
$z_7 = 7$	7

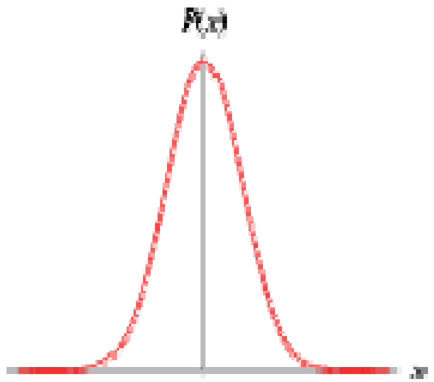
Finner mappingen mellom histogram-utjevnet og histogram-spesifisert bilde:

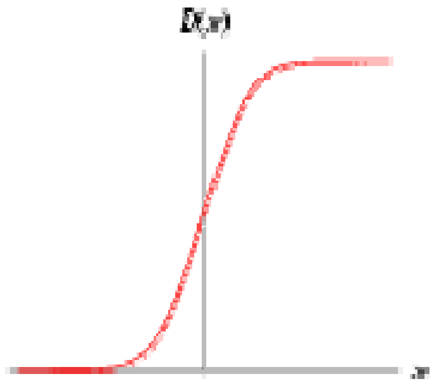
s_k	$\rightarrow$	z_q
1	$\rightarrow$	3
3	$\rightarrow$	4
5	$\rightarrow$	5
6	$\rightarrow$	6
7	$\rightarrow$	7

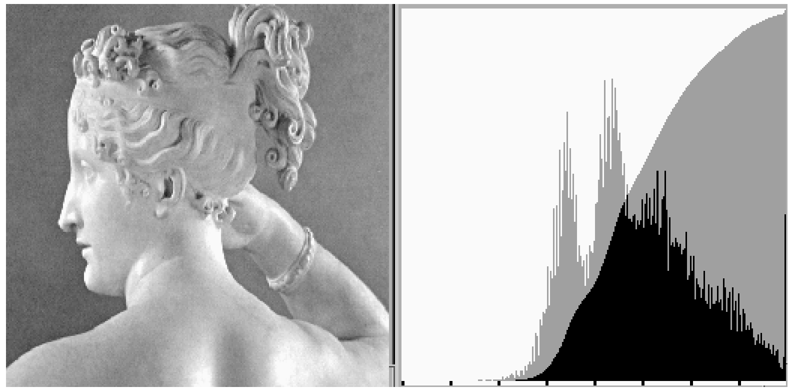
Eksempel, forts.

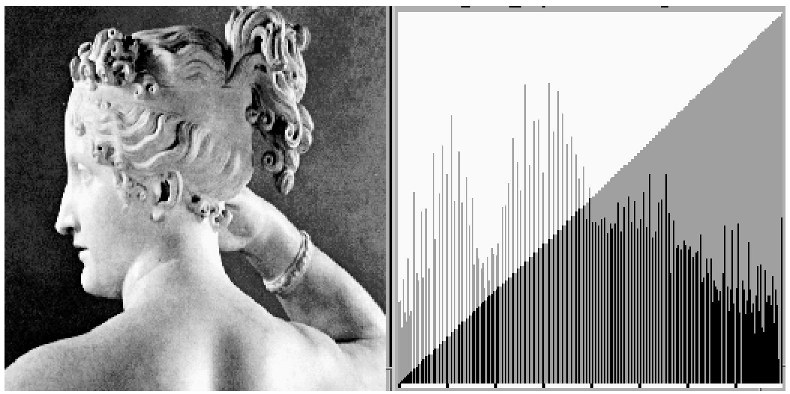


Tilpasning til Gauss-profil

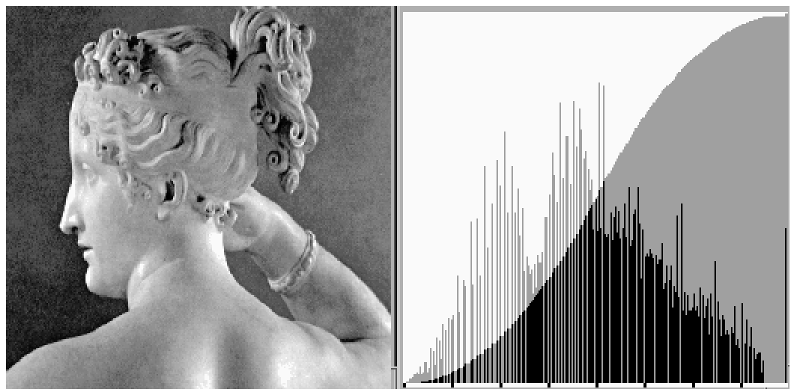








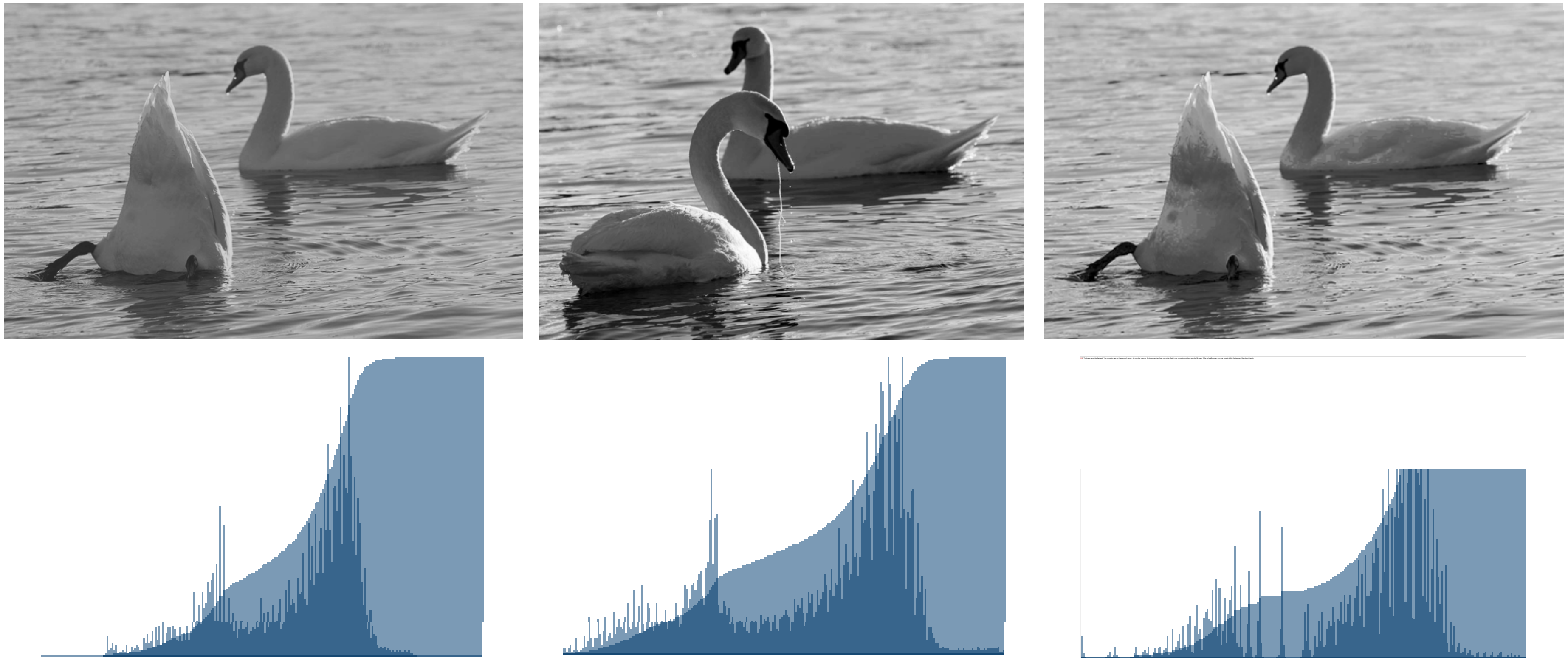
Histogram-utjevnet



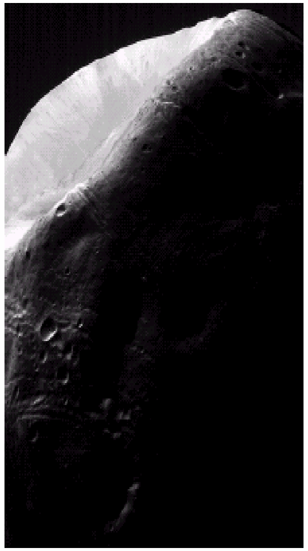
Tilpasset Gauss-form

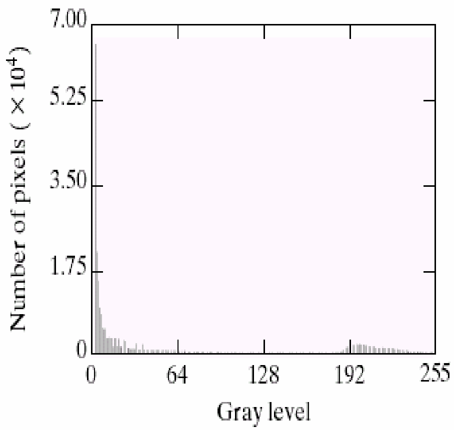
”Histogram matching”

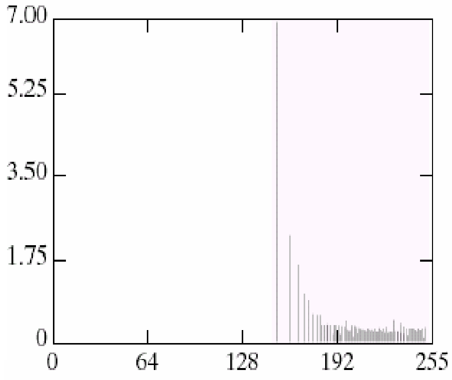
- Histogramtilpasning hvor det ene bildets histogram benyttes som ønsket form

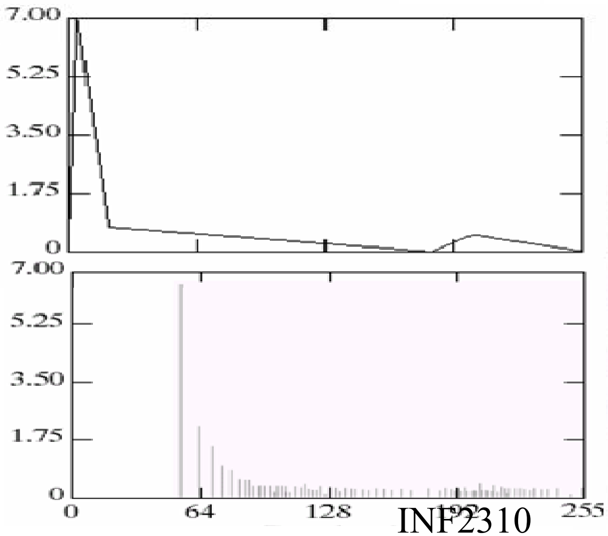


Tilpasning til annen kurve









(Bilder hentet
fra DIP/NASA)

Standardisering av histogram

- Hensikt:
  - Sørge for at alle bildene i en serie har like histogrammer
- Metoder:
  - Histogramutjevning
  - Histogramspesifikasjon (f.eks. til oppgitt Gauss-profil)
- Hvorfor? Fjerne effekten av
  - Døgnvariasjon i belysning
  - Aldringseffekter i lamper og detektorer
  - Akkumulering av støv på linser etc.
- Hvor:
  - Produkt-inspeksjon i industri
  - Ansiktsgjenkjenning
  - Mikroskopering av celler
  - ...

Når bør du IKKE gjøre dette?

- Du mener at:
  - Det kan være ”reelle” variasjoner i middelverdi og varians til bildene i en bildeserie
- Du vet ikke:
  - Om noen senere vil bruke (1. ordens) histogram-parametre til klassifikasjon av bildene
- Hva gjør du?
  - Behold originalene, og jobb på kopier
  - Gjør lineære gråtonetransformasjoner på bildene
    - Dette vil bevare strukturene i histogrammet, selv om (μ,σ) endres
- Eksempel:
  - Mikroskopering av kreft-celler.

(Fra B. Nielsen et.al)

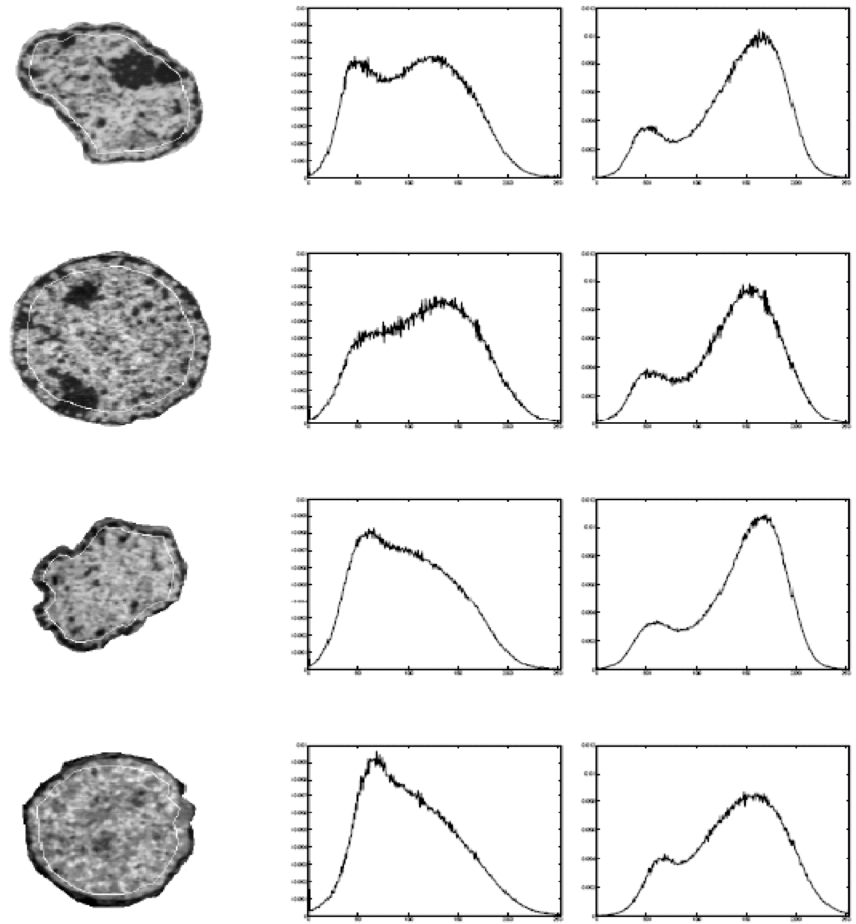


Figure 1: **First column:** *Examples of liver cell nuclei from normal, regenerating, noduli and tumor samples. The borders between the 30% peripheral and 70% central part are outlined as a thin white line.* **Second column:** *The mean gray level histograms from all cell nuclei within each of the four classes, based on the 30% peripheral part of nuclei.* **Third column:** *The mean gray level histograms from all cell nuclei within each of the four classes, based on the central 70% of the nuclei.*

Eksempel RGB-bilde

Bånd 1: R

Bånd 2: G

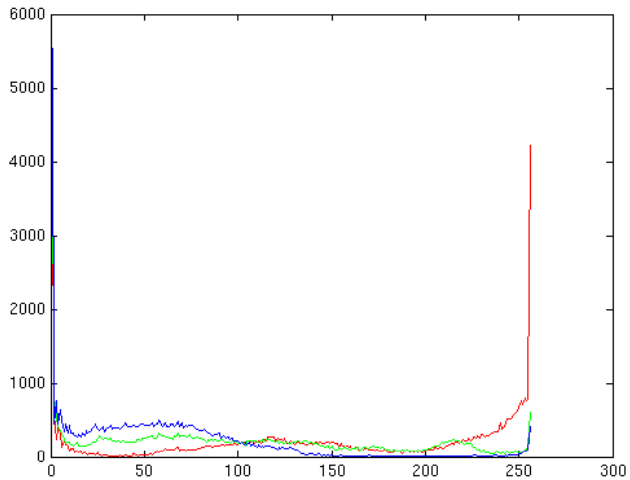
Bånd 3: B

Alle båndene projisert samtidig med forskjellig bølgelengde

1D histogram fra fargebilder

- Vi kan lage et histogram for hver kanal i et RGB-bilde

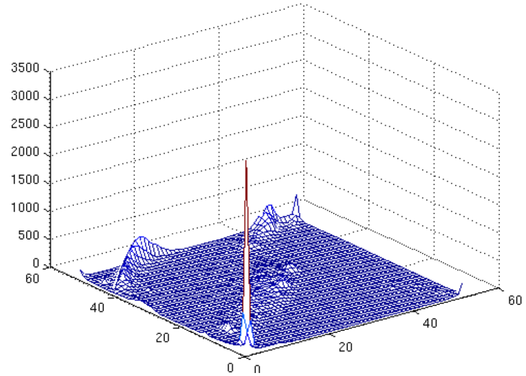
- Vi får 3 grafer



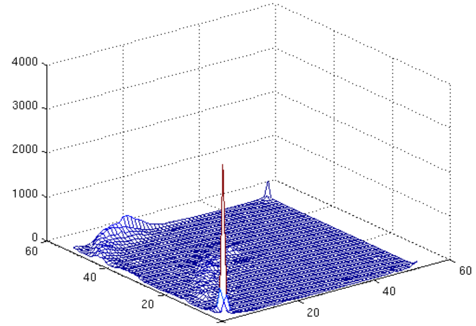
- Dette sier ikke noe om mengden av piksler som har verdien (r_1,g_1,b_1) i forhold til (r_2,g_2,b_2)

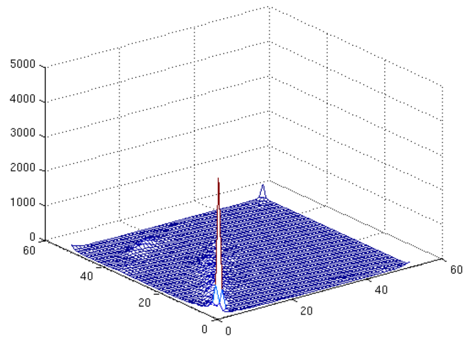
2D histogrammer fra fargebilder

- Vi kan lage 2D histogrammer for de tre kombinasjonene av 2 og 2 kanaler.



- Dette gir informasjon om forekomsten av piksler med gitte verdier av (r,g), (r,b) og (b,g).

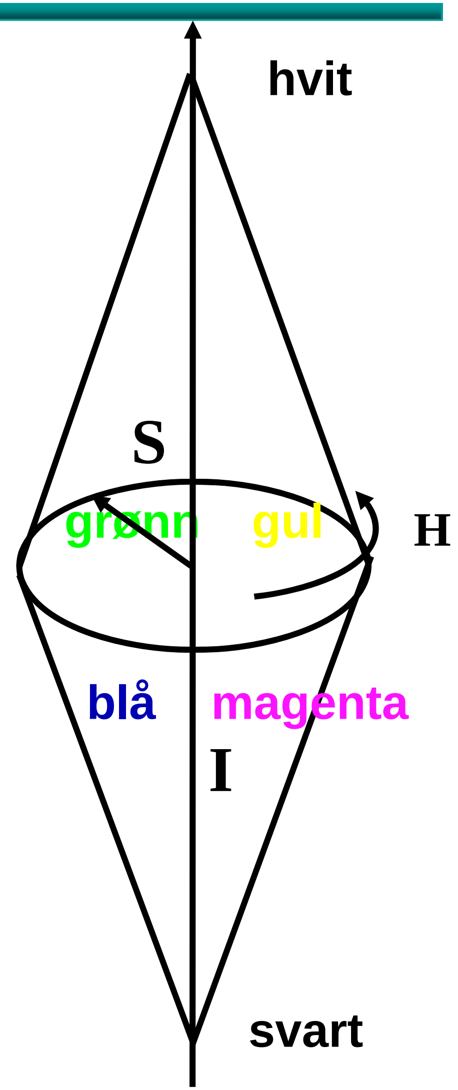




Histogramutjevning av RGB-bilder

- Histogramutjevning på hver komponent (r,g,b) uavhengig av hverandre
  - Kan føre til endring i fargetonene i bildet

- Alternativt benytte HSI:
  - Transformér bildet fra RGB til HSI
  - Gjør histogramutjevning på I-komponenten
  - Transformer HSI_{ny} tilbake til RGB



Eks: Histogramutjevning RGB vs HSI

Originalbilde

Histogramutjevning på RGB

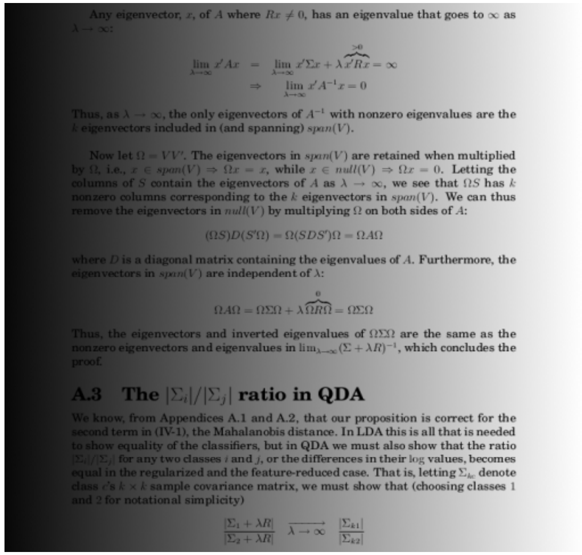
Histogramutjevning i intensitet i HSI

Lokal gråtonetransform (GTT)

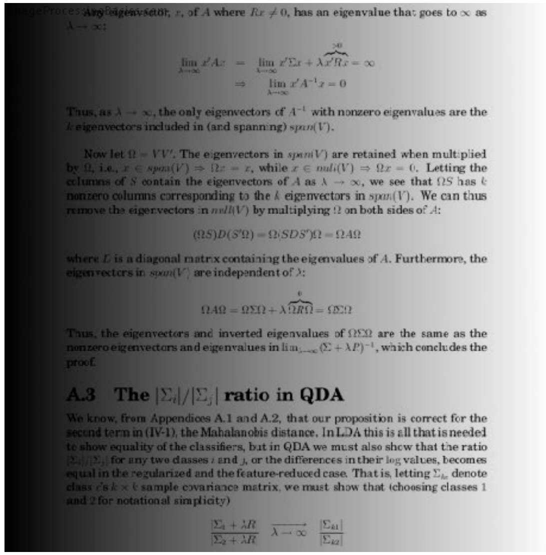
- Vil standardisere den **lokale** kontrasten
  - Samme kontrast over hele bildet

- Transformasjonene vi har sett på kan beregnes ut fra piksel-verdiene i en **lokal omegn** (kvadratisk vindu) omkring punktet (x,y)
  - Kun pikselverdien g(x,y) bestemmes av transformen basert på dette vinduets piksler
  - Altså egen transform for hvert piksel i bildet (lokal adaptivitet).

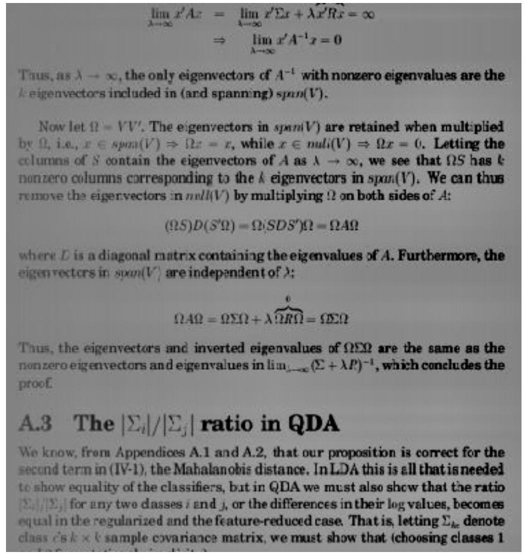
Lokal GTT – Eksempel I



Original



Global histogram-utjevning



Lokal endring av middelverdi og kontrast

Lokal GTT – Eksempel II

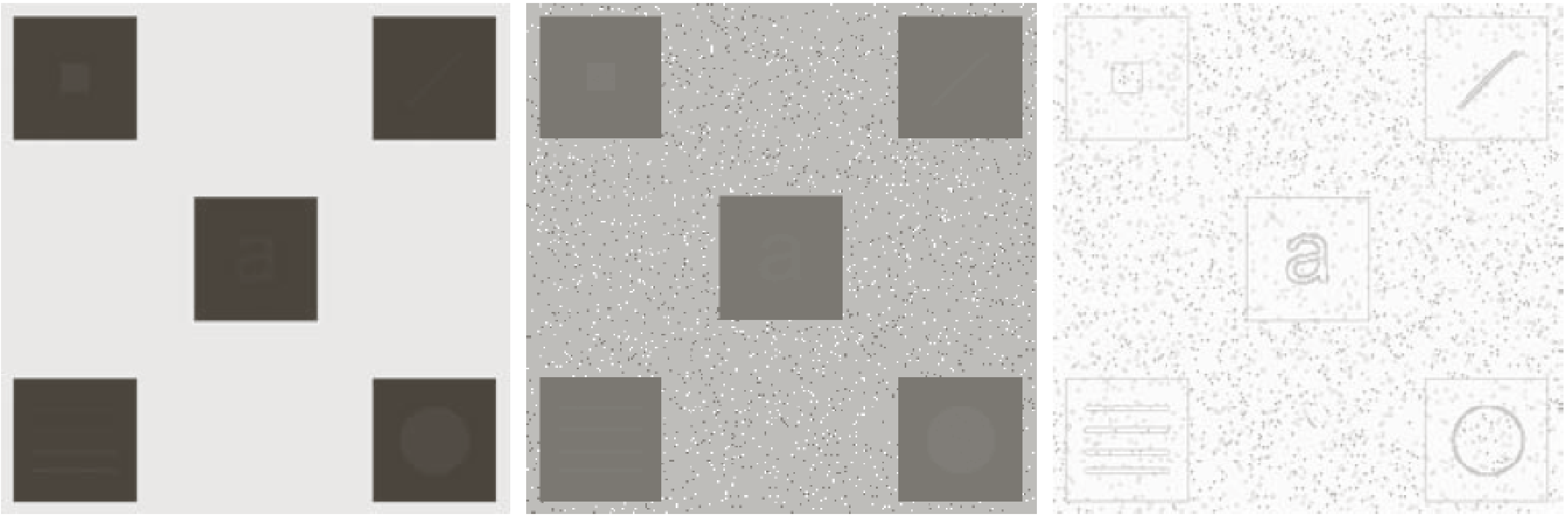
"Original"

Global histogram-utgjevning

Lokal endring av middelverdi og kontrast

Endret bildeutsnitt.
Hvorfor ?

Global HistEq vs Lokal HistEq



a b c

FIGURE 3.26 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization applied to (a), using a neighborhood of size 3 × 3.

(Fra DIP, Gonzales & Woods)

Lokal GTT - 2

- Utfør lokal GTT som gir samme kontrast over hele bildet
  - Histogramspesifikasjon
    - Beregn kumulativt histogram i et vindu sentrert om (x,y)
    - Endre senterpikselen ved den resulterende transformen

  - Lineær standardisering av σ til σ_0
    - Beregn μ(x,y) og σ(x,y) i et vindu sentrert om (x,y)
    - Transformer f(x,y) til g(x,y) med en lineær transform som gir nytt standardavvik σ_0 innenfor vinduet

$$g_1(x,y) = \mu(x,y) + \left(f(x,y) - \mu(x,y)\right)\frac{\sigma_0}{\sigma(x,y)}$$

Vi kom fram til disse uttrykkene forrige uke!

Lokal GTT - 3

- Ønsker vi lokal GTT som også gir en ny middelverdi μ_0 , så bruker vi transformen

$$g_2(x,y) = \mu_0 + \left(f(x,y) - \mu(x,y)\right)\left(\frac{\sigma_0}{\sigma(x,y)}\right)$$

- Men dette vil gi et "flatt" bilde
- Parameteren β kan styre hvor kraftig vi endrer μ:

  β = 0 => uforandret middelverdi over hele bildet

  β = 1 => lik middelverdi over hele bildet

$$g_3(x,y) = \beta\mu_0 + (1-\beta)\mu(x,y) + \left(f(x,y) - \mu(x,y)\right)\left(\frac{\sigma_0}{\sigma(x,y)}\right)$$

Lokal GTT - 4

- Hva er karakteristisk for homogene områder i et bilde?

$$\sigma(x,y)=0$$

- Her får vi problemer, fordi vi har σ i nevneren:

$$g_3(x,y)= \ldots + \left(f(x,y)-\mu(x,y)\right)\frac{\sigma_0}{\sigma(x,y)}$$

- Innfører parameteren δ:

$$g_4(x,y)= \beta\,\mu_0 + (1-\beta)\mu(x,y)$$
$$+ \left(f(x,y)-\mu(x,y)\right)\left(\frac{\sigma_0}{\sigma(x,y)+\delta\,\sigma_0}\right)$$

- Lokal pikselverdi-mapping gir økt regnearbeid.

Lokal GTT - Implementasjon

- Lokal konstrast-endring er regnekrevende
  - Histogramspesifikasjon: Beregne nytt lokalt kumulativt histogram for hver piksel
  - Lineær transform:

    Beregne ny μ og σ for hver piksel

- Benytt overlappet mellom vinduene i det man flytter til neste piksel
  - Løpende oppdatere både histogrammet, μ og σ
  - Alternativ: Bruk median og percentil.

Sentrale temaer i dag

- Histogramtransformasjoner
  - Histogramutjevning
  - Histogramtilpasning
- Standardisering av histogram for billedserier
  - Fjerne effekten av variasjoner i avbildningsforhold (døgnvariasjon, lampe, støv etc)
  - Ikke lurt med histogramtilpasning hvis histogram-formen inneholder informasjon som senere skal benyttes
  - Alternativ til standardisering av bilder med lineær transform
- Litt om histogramtransformasjoner i fargebilder
- Lokal gråtone-transformasjon
  - Samme kontrast (og middelverdi) over hele bildet
  - Beregn og benytt transformene på lokalt vindu rundt hver piksel
  - Dette er regnekrevende.