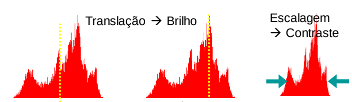
Transformações de Escalas de Intensidades em Imagens Monocromáticas

Transformações de intensidades

- Pode-se manipular as intensidades em uma imagem de modo a melhorar a qualidade perceptual da mesma
- Há duas formas de se fazer isso
  - Modificando-se a escala de intensidades de modo a alterar a média e o desvio-padrão
    - → **Transformação de escalas de cinza**
  - Modificando-se o histograma para uma forma especificada
    - → **Transformação de histograma**

Greyscale modification



- Podemos alterar as propriedades dos pixels através de transformações das intensidades
  - Transformações que mudam a escala de tons de cinza afetam os valores dos pixels
    - **Deslocam uma coluna inteira do histograma**
    - **Podem afetar significativamente os parâmetros do histograma → média e desvio-padrão**
  - Podem ser implementadas consistentemente de modo a alterar propriedades globais da imagem
    - **Como brilho, contraste**

Transformação de escalas de cinza

- Princípio geral
  - Consiste em re-mapear os valores das intensidades (tons de cinza) fazendo-os corresponder a novos valores, através de uma **função de transferência**
- Implementação
  - Pode ser realizada especificando-se a função de transferência analiticamente ou através de uma tabela

Transformação de Histograma

Introdução

As transformações de histograma são usadas para se conseguir um histograma com determinada forma que produza efeitos adequados

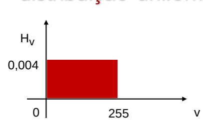
- Transformações de histograma
  - **Equalização → tornar o histograma uniforme**
  - **Normalização → para proporcionar comparação**
  - **Especificação (*histogram matching*) → especificar uma forma para o histograma**

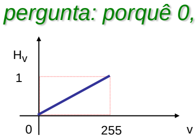
Equalização de Histograma

- A idéia aqui é encontrar uma forma de histograma que torne o contraste o maior possível sem haver perda de brilho
- Intui-se que a situação ideal seria alcançada se os diferentes valores de intensidades se distribuissem da maneira mais uniforme possível entre os pixels, de modo a produzir uma distribuição bem equilibrada
  - Isso é corroborado pela Teoria da Informação, que veremos mais adiante.

Equalização de histograma

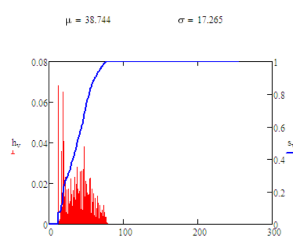
- Como resultado, pode-se concluir que seria ideal que o histograma se aproxime de uma distribuição uniforme.
- Exercício (para agora)
  - Esboce o histograma acumulado de uma distribuição uniforme (*pergunta: porquê 0,004 ?*)

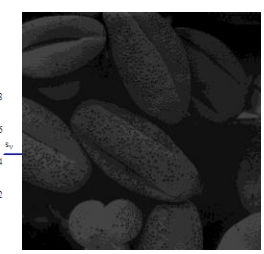




Exemplo

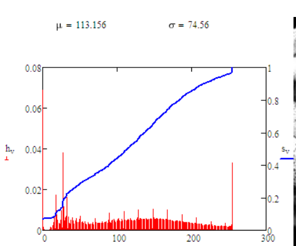
- Considere a imagem dos grãos de pólen e seu histograma:

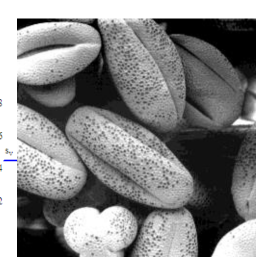




Exemplo

- Comparemos com esta versão:





Conclusões

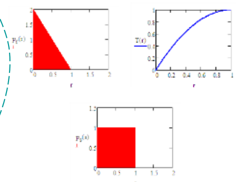
- A equalização do histograma é a transformação de uma imagem em outra tal que seu histograma se torne o mais uniforme possível (plano, *"flat"*)
- É feita através de uma modificação da escala de intensidades, aplicando-se uma lookup table (LUT) específica
- A LUT corresponde ao histograma acumulado (frequências absolutas) da imagem dada

Exercício : **(para agora)**

Dada a densidade de probabilidade $p_r(r)$

$$p_r(r) = -2r + 2 \qquad 0 \le r \le 1$$

obtenha $p_s(s)$ de modo que $s = T(r)$, sendo $T(r)$ igual à distribuição acumulada de r.



Condições de equalização

- A validade do método apresentado requer que se observe os seguintes requisitos:
  1. T(r) deve ser monotonicamente crescente (de modo a ser inversível, mantendo a ordem de transição preto → branco passando pelos cinzas com brilho progressivamente maior)
  2. $0 \le T(r) \le 1$ para $0 \le r \le 1$, devendo-se normalizar r e s pelos seus maiores valores

Algoritmo de equalização

- A abordagem mostrada baseou-se em r e s variando de modo contínuo.
- No caso das imagens, r e s representam tons de cinza, que variam de forma **discreta**.

$$s_k = T(r_k) = \frac{1}{K}\sum_{v=0}^{k} h_{r_v}$$

hr é o histograma de Frequências relativas dos pixels da imagem dada

- Como consequência, a equalização **não é 100% perfeita**, porém o histograma resultante $p_s(s)$ será mais uniforme que o original
- Questão → explique por que não é 100% perfeita

Caso discreto

- No caso discreto, a transformação geralmente não produz um histograma uniforme, apenas aproxima.
- Ela é dada por:

$$s_k = T(r_k) = \frac{1}{K}\sum_{v=0}^{k}\frac{n_{r_v}}{n} = \frac{1}{nK}\sum_{v=0}^{k} n_{r_v}$$

- Lembrando que $K = \frac{1}{s_{max}}$

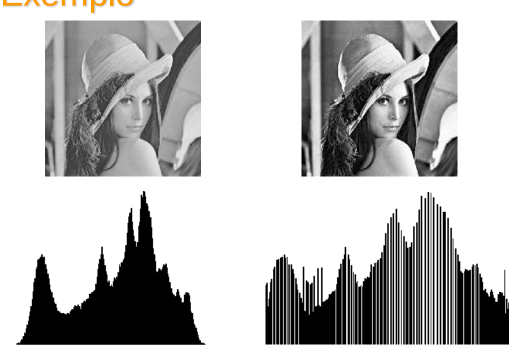
Caso discreto

- A transformação correta é dada por:

$$s_k = T(r_k) = \left\lfloor \frac{1}{nK}\sum_{v=0}^{k} n_{r_v} \right\rfloor$$

- onde $\lfloor \bullet \rfloor$ é o menor inteiro em $\bullet$

Exemplo



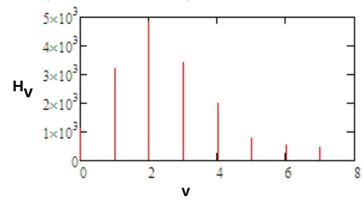
Algoritmo de equalização

1. Construir o histograma de frequências relativas dos pixels
2. A partir do mesmo, construir o histograma acumulado de frequências relativas
3. Utilizar o histograma acumulado como LUT para produzir a imagem equalizada
4. Arrendondar os resultados para o tom de cinza imediatamente inferior

- Exercício – Proponha um algoritmo para equalizar imagens monocromáticas.
- Exercício – implementar (em Python p.ex.) esse algoritmo de equalização de imagens monocromáticas

Exercício

- Considere o histograma dado pelo gráfico e a tabela como sendo de uma certa imagem monocromática. Calcular o histograma equalizado (manualmente)



v	H_V
0	1120
1	3214
2	4850
3	3425
4	1995
5	784
6	541
7	455

Roteiro de solução

1. Normalizar os valores dos tons de cinza, dividindo-os pelo maior (= 7 no caso)
2. Tabelar as frequências relativas, dividindo as absolutas pelo número total de pixels
3. Tabelar as frequências relativas acumuladas
4. Usar as frequências acumuladas como LUT para determinar o novo mapeamento dos tons
5. Construir o novo histograma

v	$r = v / v_{max}$	$H_V(r)$	$h_V(r)$	$\Sigma\, h_V$	LUT	$s = T^{-1}(r)$	$h_V(s)$

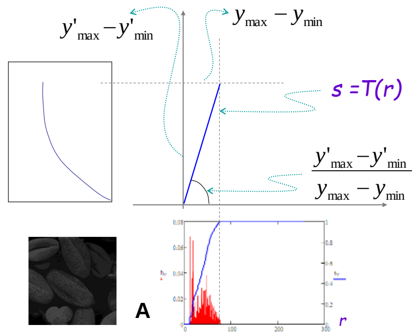
Normalização de Histograma

Constrast stretching

Normalização de histograma

- É uma outra transformação, no caso uma transformação linear, que expande o histograma com o objetivo de aumentar o contraste.
- Usa-se tal transformação quando a imagem foi adquirida de forma a não usar toda faixa dinâmica disponível

A transformação s = T(r) de escalas de cinza (LUT) fará um mapeamento linear baseado nas diferenças de intensidade.



Normalização de histograma

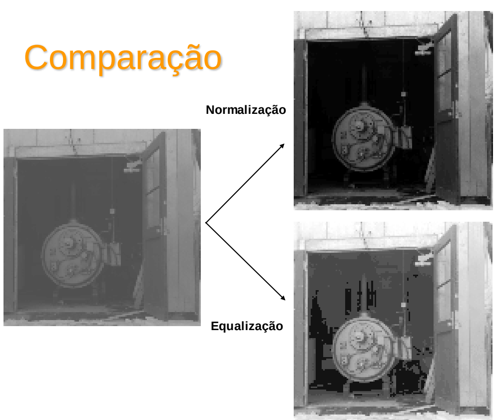
- A transformação é definida por:

$$y' = \frac{(y'_{max} - y'_{min})}{(y_{max} - y_{min})} \cdot (y - y_{min}) + y'_{min}$$

$y_{max} - y_{min}$ = diferença entre o maior e o menor valores da imagem dada

$y'_{max} - y'_{min}$ = diferença entre o maior e o menor valores da imagem resultante

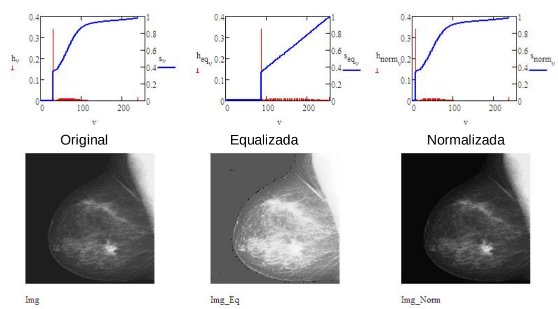
Comparação



Comparação

Equalizada – muito brilho, afeta a detecção das calcificações

Normalizada – melhora o contraste sem aumentar o brilho



Transformações pontuais em imagens binárias e em níveis de cinza

Transformações entre imagens

- Até agora vimos como fazer transformações entre imagens usando LUT e histogramas.
- Normalmente, essas transformações são feitas para melhorar a visualização das imagens e não para “facilitar” o trabalho de extração de informações.
- Tanto que muitas vezes ela é feita no dispositivo visualizador diretamente.
- A partir de agora vamos estudar transformações cujo domínio e o contra-domínio sejam o conjunto das imagens.

$$\phi : \text{imagens} \longrightarrow \text{imagens}$$

Operador Adição

- A adição é uma operação binária conhecida de nós desde bem pequenos.
- Quando tratamos de imagens, os operandos são imagens:

$$+(f, g)$$

- onde f e g são duas imagens.

Operador Adição

- Porém, também é usual definir a adição de uma imagem a uma constante. Neste caso, a constante é adicionada a todos os pixels da imagem. Isso é equivalente a somar uma imagens constante.

$$+(f, K)$$

- onde f é uma imagem e K uma constante.

Operador Adição

- Formalmente,

$$+(f, g)(x) = f(x) + g(x), x \in E$$

$$+(f, K)(x) = f(x) + K, x \in E$$

onde f, g são duas imagens, K é uma constante válida e E é o domínio da imagem.

Operador Adição - saturação

- O problema da definição anterior é que ela não respeita o tipo da imagem.
- O tipo da imagem é definido pelo contra-domínio da função que representa a imagem.
- Normalmente ele é um intervalo dos inteiros positivos, por exemplo, [0,255].
- Assim, a operação de adição precisa ser melhor definida.

Operador Adição - saturação

- A adição respeita os limites da imagem

$$\dot{+}(a, b) = \begin{cases} a + b & \text{if } 0 \leq a + b \leq K \\ K & \text{if } a + b > K \end{cases}$$

- onde a e b são números naturais e K é o maior valor do intervalo

Operador Adição 2 - saturação

- Neste caso, há limites diferentes para entrada e saída.

$$\dot{+}(a, b) = \begin{cases} a + b & \text{if } 0 \leq a + b \leq K_1 \\ K_2 & \text{if } a + b > K_1 \end{cases}$$

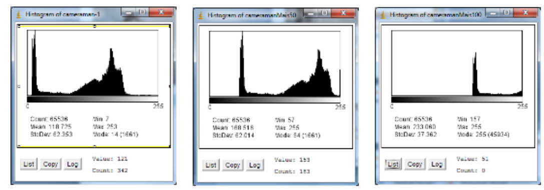
- onde $K_1(K_2)$ é o máximo valor possível da imagem de entrada (saída)

Operador Adição

- Vamos começar exemplificando o caso da adição de uma constante a uma imagem.

Original | Original + 50 | Original + 150

Original | Original + 50 | Original + 150



Operador Adição

- Vamos somar uma imagem em outra, mantendo o limite da entrada e da saída.

frame 0 | frame 1 | frame 0 + frame 1

Operador Adição

- Vamos somar uma imagem em outra, agora com o limite da entrada e da saída -16bits

frame 0 | frame 1 | frame 0 + frame 1

Operador Adição

- Vamos somar uma imagem em outra, mudando o limite da saída.

frame 0 | frame 1 | frame 0 + frame 1

Operador Subtração

- O operador subtração é equivalente ao operador adição.

$$-(f, g)$$

$$-(f, K)$$

Operador Subtração

- Formalmente,

$$-(f, g)(x) = f(x) - g(x), x \in E$$

$$-(f, K)(x) = f(x) - K, x \in E$$

onde f, g são duas imagens, K é uma constante válida e E é o domínio da imagem.

Operador Subtração - saturação

- A adição respeita os limites da imagem

$$\dot{-}(a, b) = \begin{cases} a - b & \text{if } 0 \leq a - b \leq K \\ 0 & \text{if } 0 < a - b \end{cases}$$

- onde a e b são números naturais e K é o maior valor do intervalo

Operador Subtração 2

- Neste caso, há limites diferentes para entrada e saída.

$$\dot{-}(a, b) = \begin{cases} a - b & \text{if } -K_1 - 1 \leq a - b \leq K_1 \\ -K_2 - 1 & \text{if } a - b < -K_1 - 1 \end{cases}$$

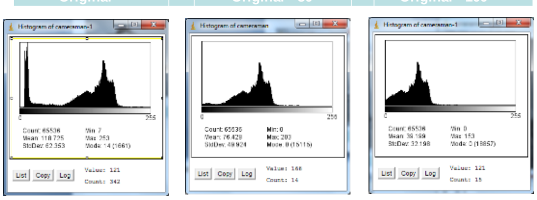
- onde $K_1(K_2)$ é o máximo valor possível da imagem de entrada (saída)

Operador Subtração

- Vamos começar exemplificando o caso da adição de uma constante a uma imagem.

Original | Original - 50 | Original - 100

Original | Original - 50 | Original - 100



Operador Subtração

- Vamos subtrair uma imagem da outra, mantendo o limite da entrada e da saída.

frame 0 | frame 1 | frame 1 - frame 0

Operador Subtração

- Vamos subtrair uma imagem da outra, agora com o limite da entrada e da saída -32bits

frame 0 | frame 1 | frame 1 - frame 0

Operador Subtração

- Agora vamos subtrair uma imagem da outra, mudando o limite da saída.

frame 0 | frame 1 | frame 1 - frame 0

Operador Multiplicação

- A multiplicação pode ser feita usando-se um número inteiro, que equivale a somar a imagem com ela mesma várias vezes.
- Ou multiplicar uma imagem por outra, isto é, ponto a ponto, multiplicam-se os valores.

Operador Multiplicação

Imagem Original | Imagem gradiente | Imagem resultante

Operador Multiplicação

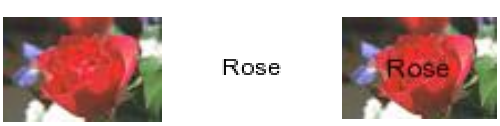
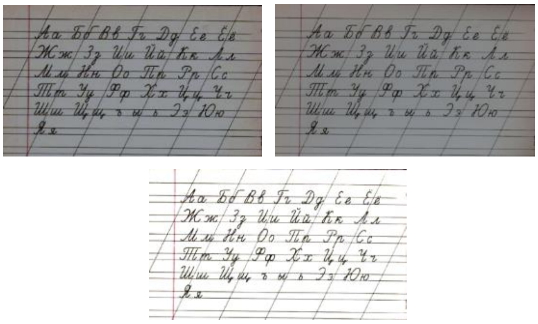


Imagem Original | Imagem rótulo | Imagem resultante

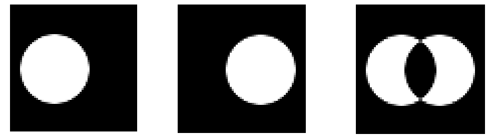
Operador Divisão

- A divisão pode ser feita usando-se um número inteiro diferente de 0.
- Ou dividir uma imagem por outra, isto é, ponto a ponto, dividem-se os valores e acerta-se o intervalo.

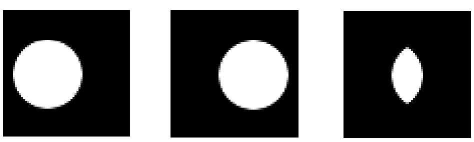
Operador Divisão



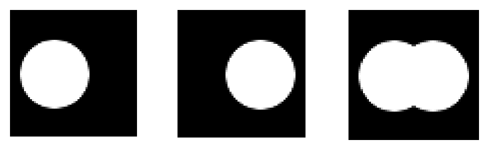
Operador Diferença Absoluta



Operador E

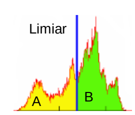


Operador Ou



Limiarização (Thresholding)

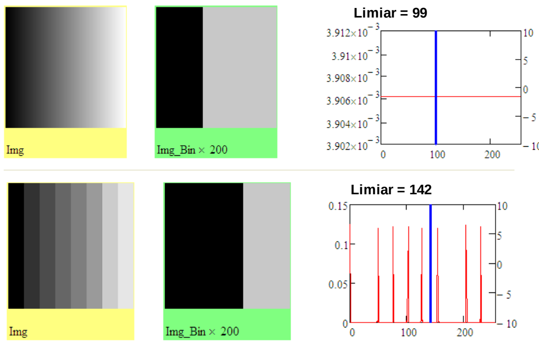
Limiarização



- Limiar (threshold) → valor limite L para tomar uma <u>decisão</u>
  - X ≤ L → decisão A
  - X > L → decisão B
- No caso de limiarização da intensidade, ou limiarização de níveis de cinza (*grey-level thresholding*):
  - X = intensidade ou nível de cinza
  - Decisão → classificar o nível X como pertencente à classe A ou B
  - A limiarização particiona o histograma em 2 distribuições que correspondem a duas *classes*
- A limiarização é um caso particular de classificação de regiões
  - Classificação binária com critério global baseado no histograma

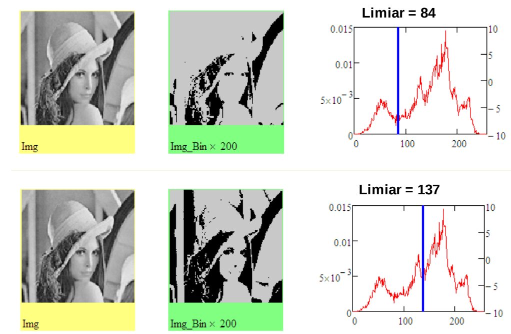
$Img\_Bin_{i,j} := if(Img_{i,j} > Limiar, 1, 0)$

Limiarização



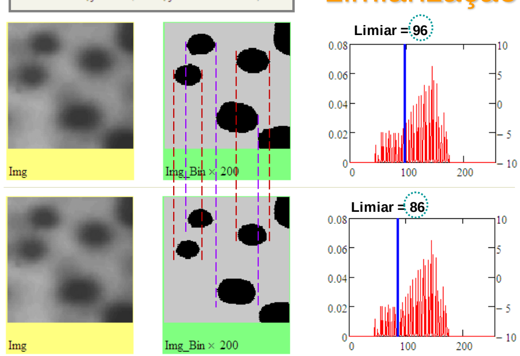
$Img\_Bin_{i,j} := if(Img_{i,j} > Limiar, 1, 0)$

Limiarização



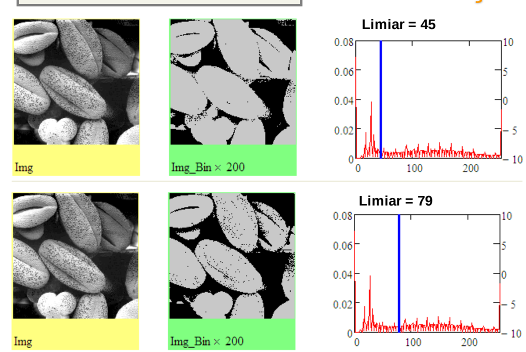
$Img\_Bin_{i,j} := if(Img_{i,j} > Limiar, 1, 0)$

Limiarização



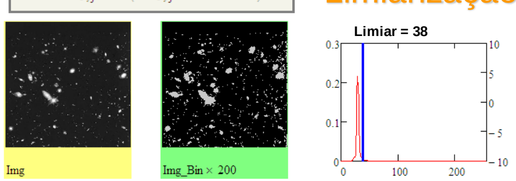
$Img\_Bin_{i,j} := if(Img_{i,j} > Limiar, 1, 0)$

Limiarização



$Img\_Bin_{i,j} := if(Img_{i,j} > Limiar, 1, 0)$

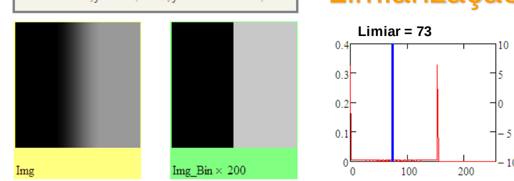
Limiarização



Empregando-se escala logarítmica para o histograma →

$Img\_Bin_{i,j} := if(Img_{i,j} > Limiar, 1, 0)$

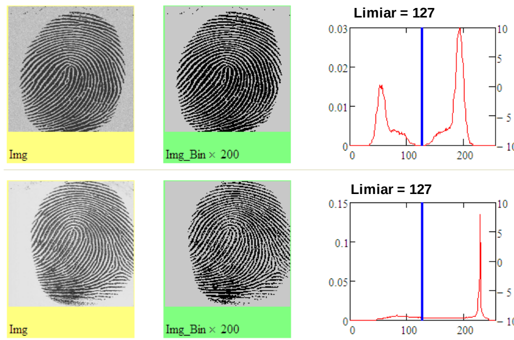
Limiarização



Empregando-se escala logarítmica para o histograma →

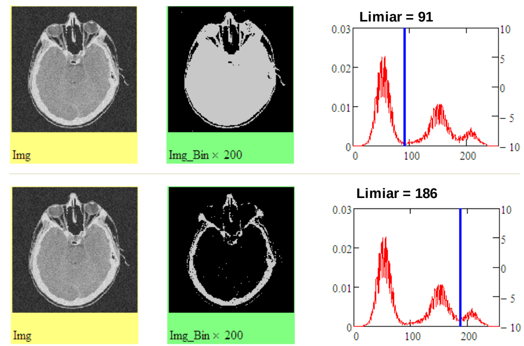
$$Img\_Bin_{i,j} := \text{if}\left(Img_{i,j} > Limiar, 1, 0\right)$$

Limiarização



$$Img\_Bin_{i,j} := \text{if}\left(Img_{i,j} > Limiar, 1, 0\right)$$

Limiarização



$$Img\_Bin_{i,j} := \text{if}\left(Img_{i,j} > Limiar, 1, 0\right)$$

Limiarização