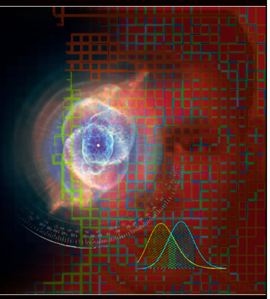


Introdução à

Visão Computacional

e ao

Processamento de Imagens

Roberto Hirata Jr. e João E. Kogler Jr.

3ª aula

Propriedades perceptuais e sua relação com o Histograma do mapa de intensidades

---

Mapa de intensidades

M =

	0	1	2	3	4
0	240	240	241	240	241
1	240	240	241	241	242
2	240	241	240	240	240
3	239	241	240	240	239
4	239	240	241	240	240
5	240	241	241	241	240
6	241	241	241	241	242
7	243	242	241	241	242
8	242	242	242	241	241
9	243	242	242	241	243

- As intensidades dos pixels traduzem:
  - As propriedades da iluminação da cena
    - **O sensor capta a quantidade de luz proveniente dos diversos pontos e direções**
  - As propriedades dos materiais da cena
    - **O sensor mede as radiâncias correspondentes aos diversos tipos de materiais → refletores, transmissores e emissores de radiação**

Mapas de intensidades

- Propriedades radiométricas dos mapas de intensidades (monocromáticas)
  - Brilho → qualidade subjetiva da imagem ser mais clara ou mais escura
  - Contraste → qualidade subjetiva das regiões da imagem apresentarem fronteiras de transição de luminosidade
    - **São qualidades GLOBAIS, da imagem**
    - **Apesar do brilho e o contraste serem subjetivos, podemos definir medidas objetivas dos mesmos**

Mapas de Intensidades - exemplos

Muito Brilho

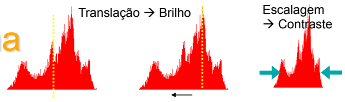
Pouco Brilho

Mapas de Intensidades - exemplos

Pouco Contraste

Muito Contraste

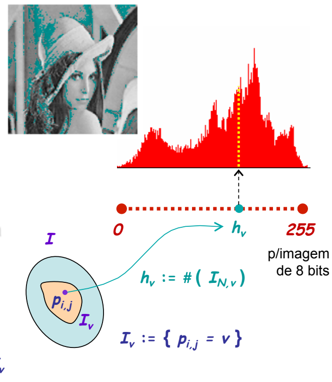
Histograma



- Balancear as qualidades visuais de uma imagem (monocromática) significa
  - modificar consistentemente propriedades globais da imagem que produzam melhor julgamento da qualidade (perceptual)
    - Como brilho, contraste
- A busca de um método objetivo para tanto começa com a análise de como se distribuem os valores dos pixels
  - Isto é, seus tons monocromáticos

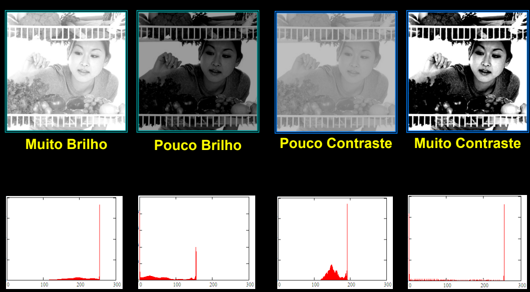
Histograma

- É uma relação que mapeia, para cada valor de intensidade que um pixel possivelmente possa ter, o número de vezes em que ela aparece na imagem.



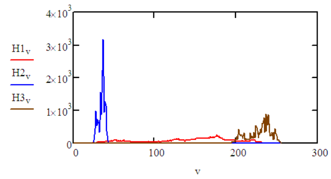
I → *imagem*
I_v → *subconjunto da imagem*
h_v → *cardinalidade de* I_v

Exemplos de Histogramas



Comparação dos Histogramas

- Para melhor comparar as imagens a partir de seus histogramas, introduziremos alguns parâmetros que caracterizam quantitativamente o histograma.



Parâmetros do Histograma

- Os histogramas podem ser caracterizados por dois parâmetros (entre outros):
  - Parâmetro de posição:
    - **Média → indica a posição (centralização) do histograma**
  - Parâmetro de dispersão:
    - **Desvio-padrão → indica a largura (dispersão, espalhamento) do histograma**

Parâmetros do Histograma

Média

$$\mu = \frac{1}{N}\sum_{v=0}^{v_{max}} (vH_v)$$

$v_{max} = 2^b$ → b = nº de bits
Para 8 bits → vmax = 255

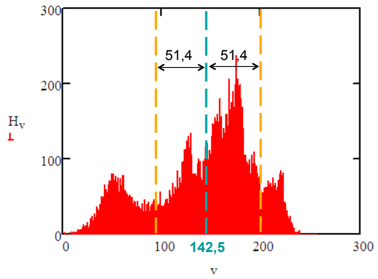
Nessas expressões estão indicados, de fato, os chamados **estimadores** (de máxima verossimilhança) da média e do desvio-padrão. Essas expressões são livres de vício de estimação.

Desvio-padrão

$$\sigma = \sqrt{\frac{\sum_{v=0}^{v_{max}} [(v-\mu)^2 H_v]}{N-1}}$$

Calculemos a média e o desvio-padrão da imagem Lena:

$H_v := n\_pixels(v)$ $\mu = 142.529$ $\sigma = 51.399$

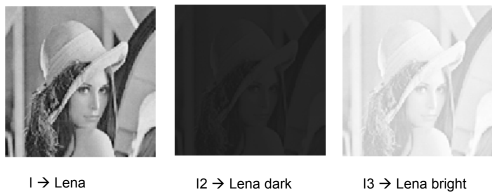


$m := rows(I)$ $m = 128$

$n := cols(I)$ $n = 128$

$N := m \cdot n$ $N = 16384$

$b := 8$ bits $2^b = 256$

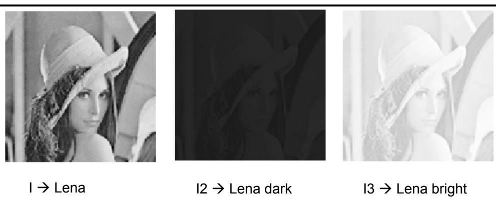


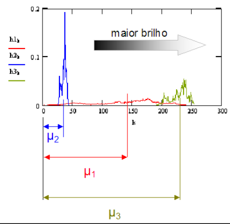
Calculemos, agora, as médias e os desvios-padrões dessas imagens:

$\mu 1 = 142.529$ $\sigma 1 = 51.399$

$\mu 2 = 34.979$ $\sigma 2 = 3.736$

$\mu 3 = 228.99$ $\sigma 3 = 13.869$





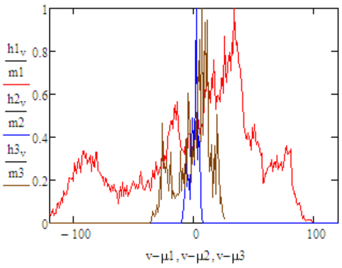
Nota-se aqui uma grande correlação entre o **valor médio** da intensidade nos pixels e o **brilho** da imagem

$\mu 1 = 142.529$

$\mu 2 = 34.979$

$\mu 3 = 228.99$

Para explorar agora a correlação entre o **desvio-padrão** e o **contraste** na imagem, empregaremos a técnica de centralizar os três histogramas na origem e escalar as amplitudes normalizando-as pelo valor máximo de cada histograma.



Valores máximos de cada histograma

$m1 := max(h1)$ $m1 = 0.014$

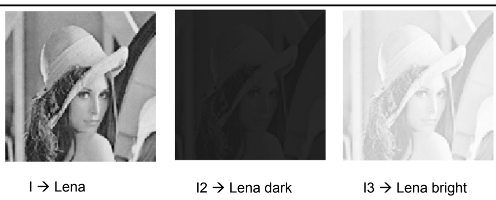
$m2 := max(h2)$ $m2 = 0.193$

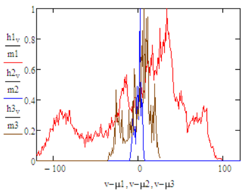
$m3 := max(h3)$ $m3 = 0.055$

$\sigma 1 = 51.399$

$\sigma 2 = 3.736$

$\sigma 3 = 13.869$





Nota-se aqui uma grande correlação entre o desvio-padrão da intensidade nos pixels e o contraste da imagem

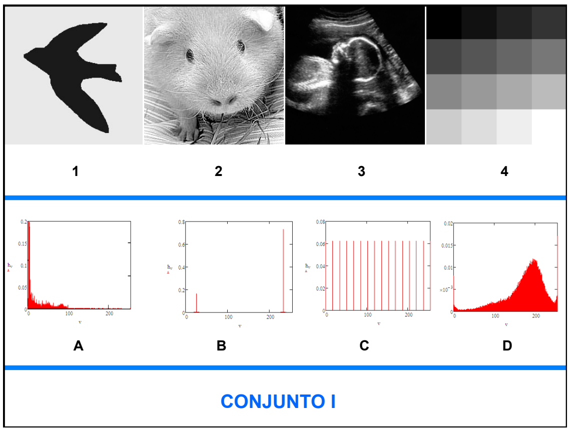
$\sigma 1 = 51.399$

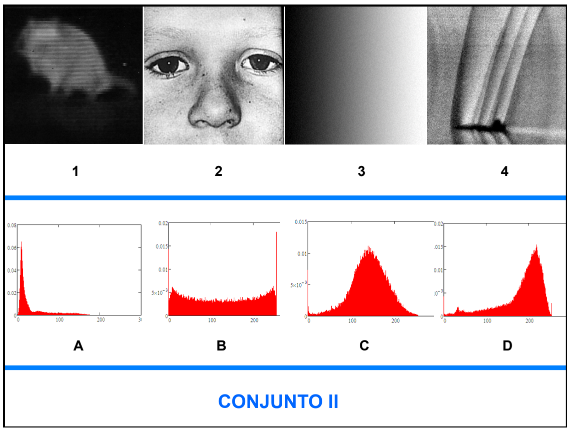
$\sigma 2 = 3.736$

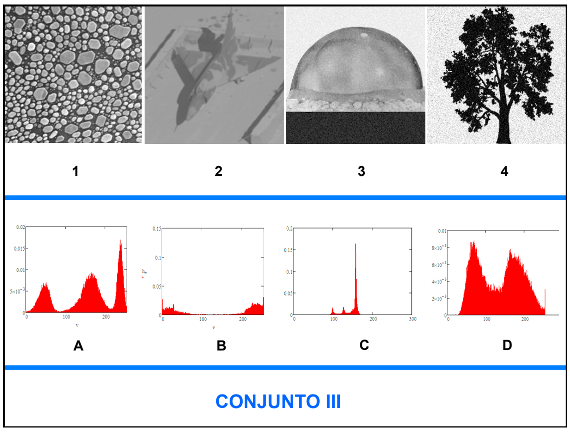
$\sigma 3 = 13.869$

Exercício sobre histogramas

- O próximos slides apresentarão conjuntos formados por 4 imagens e 4 histogramas cada um
- Associe a cada imagem de cada conjunto o seu respectivo histograma







Frequências relativas

Definição do histograma de frequências relativas

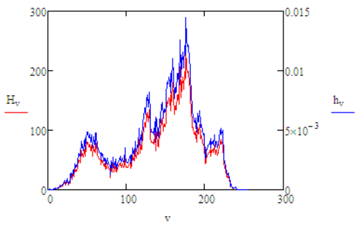
– Frequência relativa define-se por:

$$h_v = \frac{H_v}{N}$$

sendo N a quantidade de pixels na imagem, isto é

$$N = \sum_{v=0}^{v_{max}} H_v \Rightarrow \boxed{h_v = \frac{H_v}{\sum_{v=0}^{v_{max}} H_v}}$$

Frequências relativas - exemplo



Histograma Acumulado

- É definido como sendo a soma dos valores do histograma de 0 até um dado valor v.

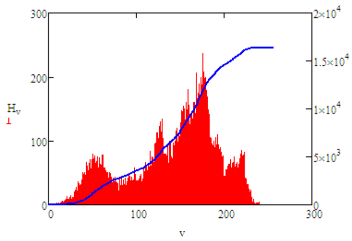
$$S_v = \sum_{k=0}^{v} H_k$$

– Podemos definir também o histograma acumulado de frequências relativas:

$$s_v = \frac{S_v}{N} \Rightarrow \boxed{s_v = \sum_{k=0}^{v} h_k}$$

Exemplo:

Frequências absolutas



$m := \mathrm{rows}(I)$ $m = 128$

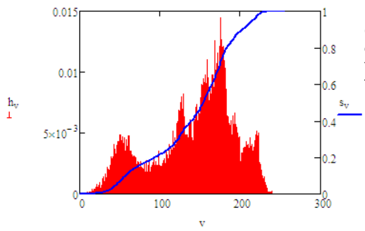
$n := \mathrm{cols}(I)$ $n = 128$

$N := m \cdot n$ $N = 16384$

$b := 8$ bits $2^b = 256$

Exemplo:

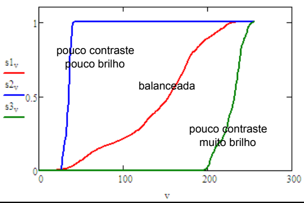
Frequências relativas



O histograma acumulado de frequências relativas termina sempre no valor final 1.

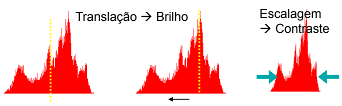


$s1_v := \sum_{k=0}^{v} h1_k$ $s2_v := \sum_{k=0}^{v} h2_k$ $s3_v := \sum_{k=0}^{v} h3_k$



Os histogramas acumulados relativos permitem comparar facilmente as qualidades de **brilho** (medida de posição) e **contraste** (medida de dispersão) das imagens.

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

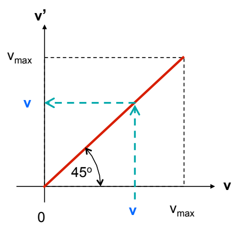
Transformação de escalas de cinza

- 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 escalas de cinza

- Função de transferência
  - Identidade
    - **Nada muda**
    - **Os valores são re-mapeados neles mesmos**

$$v' = v$$

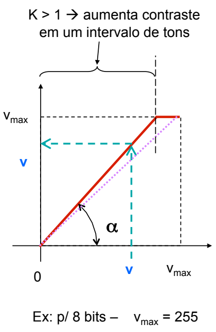


Ex: p/ 8 bits – $v_{max} = 255$

Transformação de escalas de cinza

- Função de transferência
  - Linear homogênea
    - **Altera o contraste**
      - Aumenta para k>1
    - **Preserva a origem**

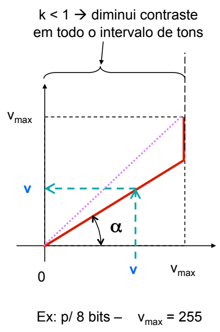
$$v' = kv$$



Transformação de escalas de cinza

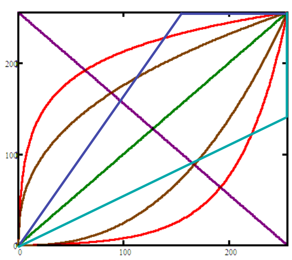
- Função de transferência
  - Linear homogênea
    - **Altera o contraste**
      - Aumenta para k>1
      - Diminui para k<1
    - **Preserva a origem**

$$v' = kv$$



Trasformações de escala de intensidades

- São aplicações de funções para re-mapear a escala de tons de cinza
  - São eficientemente implementadas via LUT
- Tipos:
  - Lineares
  - Polinomiais
  - Logaritmicas
  - Exponenciais
  - Inversoras



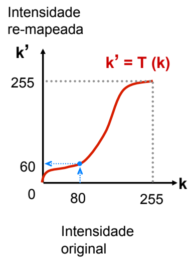
Demonstração - LUT

Transformações com LUT

- Construir um programa que leia os valores das intensidades nos pixels de uma imagem monocromática e os re-mapeie com diversas funções de transferência implementadas via Look-up table (LUT)

  a) Linear - com ganho β dado
  b) Logaritmica, base a (→ $\log_a k$)
  c) Exponencial, base a (→ a^k)
  d) Potência,expoente γ (◊ k^γ)
  e) Inversão da escala (negativo foto)

- Saturar os valores normalizando-os para o intervalo [0,255].



Construção da LUT

- Dada a função de transferência *f* e o valor máximo de intensidade maxI, a função table_map cria a LUT

Construção da LUT

- Dada a função de transferência *f* e o valor máximo de intensidade maxI, a função table_map cria a LUT

```
table_map(f,maxI) := | for k∈0..maxI
                     |   | T_k ← f(k)  if 0 ≤ f(k) ≤ 255
                     |   | T_k ← 0  if f(k) < 0
                     |   | T_k ← 255  if f(k) > 255
                     | T
```

Aplicação da LUT à imagem

- Dadas a LUT *T* e a matriz *M* contendo a imagem, a função **remap(M,T)** re-mapeia as intensidades da imagem seguindo a LUT dada T

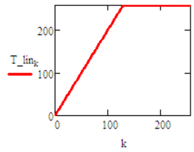
```
remap(M,T) := | for i ∈ 0..rows(M) − 1
              |   for j ∈ 0..cols(M) − 1
              |     | k ← M_{i,j}
              |     | MT_{i,j} ← T_k
              | MT
```

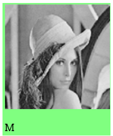
LUT Linear

seja : $\beta := 2$ por exemplo e $f(k) := \beta \cdot k$

T_lin := table_map(f,255)

MT := remap(M,T_lin)





seja : M := READBMP(".\img\Lena1.bmp")

LUT Logaritmica

$f(k) = \Lambda \cdot \log(k+1)$

A parcela +1 força a função f(k) passar pela origem.Devemos calcular Λ tal que f(255) = 255, isto é, a função passa pelo ponto diagonalmente oposto à origem. Temos:

$255 = \Lambda \cdot \log(255+1)$

Logo : $\Lambda := \frac{255}{\log(256)}$

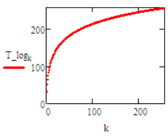
e portanto,

p.ex.- base 10

$f(k) := \Lambda \cdot \log(k+1)$ $\Lambda = 105.886$

MT := remap(M,T_log)

T_log := table_map(f,255)



LUT Expoencial

$f(k) = \left(e^{\frac{k}{\kappa}} - 1\right)$

A parcela -1 força a função passar pela origem. Devemos calcular o valor de κ tal que f(255) = 255. Temos:

$e^{\frac{255}{\kappa}} - 1 = 255$

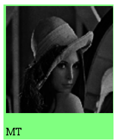
MT := remap(M,T_exp)

portanto :

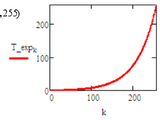
$\frac{k}{\kappa} = \ln(256)$ ou seja, $\kappa := \frac{255}{\ln(256)}$

portanto resulta :

$f(k) := \left(e^{\frac{k}{\kappa}} - 1\right)$ $\kappa = 45.986$



T_exp := table_map(f,255)



LUT Potência

Temos :

$f(k) = \rho \cdot k^{\gamma}$

Devemos calcular ρ de modo que f(255) = 255. Então:

$255 = \rho \cdot 255^{\gamma}$

MT := remap(M,T_pow)

Logo : $\rho = 255^{1-\gamma}$

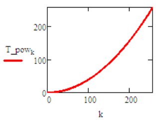
$f(k) = 255^{1-\gamma} \cdot k^{\gamma}$

exemplo : ***potência quadrada***

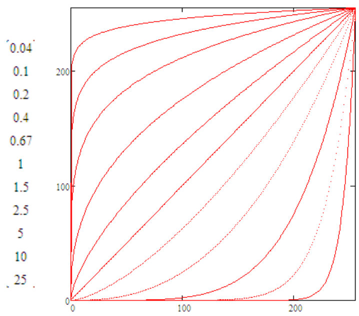
$\gamma := 2$

$f(k) := 255^{1-\gamma} k^{\gamma}$

T_pow := table_map(f,255)



LUT Potência



Exemplos de funções de transferência tipo potência de diversos valores de expoentes

Transformação de Histograma

Introdução

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

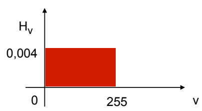
- Transformações de histograma
  - **Equalização → tornar o histograma uniforme**
  - **Normalização → para proporcionar comparação**
  - **Especificação (*histogram matching*) → especificar uma forma para o histograma**

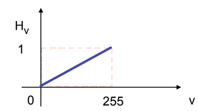
Equalização de Histograma

- 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

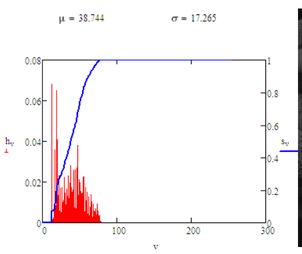
- Como resultado, pode-se concluir que seria ideal que o histograma se aproxime de uma distribuição uniforme.
- Exercício (para agora)
  - Esboce o histograma acumulado de uma distribuição uniforme (*pergunta: porquê 0,004 ?*)

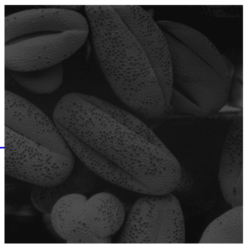




Exemplo

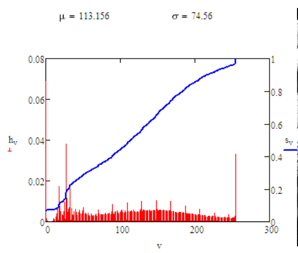
- Considere a imagem dos grãos de pólen e seu histograma:

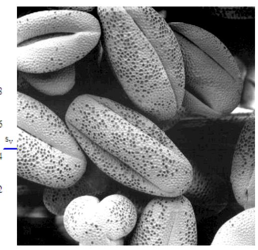




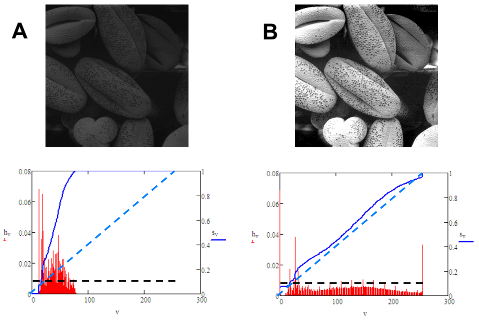
Exemplo

- Comparemos com esta versão:

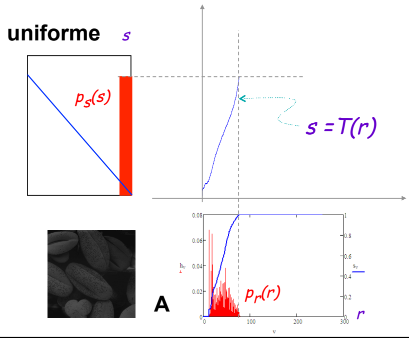




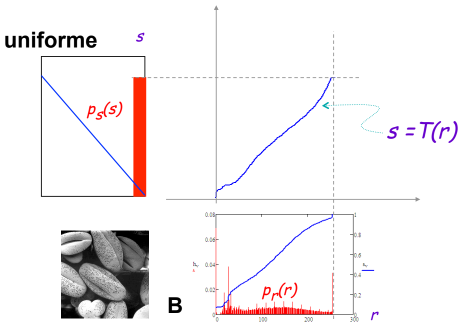
A imagem B apresenta melhor definição, com mais contraste e brilho adequado. Seus histogramas aproximam-se mais da distribuição uniforme.



Existe uma transformação s = T(r) de escalas de cinza (LUT) que torne a distribuição da imagem A mais próxima de uma uniforme ?

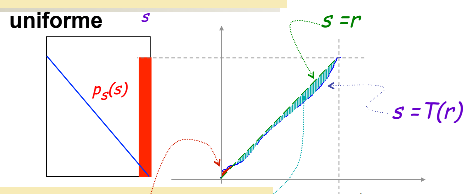


~~Mas a questão de qual deveria ser T(r)~~ ... B, ... estão de comodidade, embora B seja uma imagem relativamente boa.



- Usando-se a função identidade T(r)=r como LUT, os tons seriam remapeados nos mesmos valores originais

...rmação de escala ...udar a quantidade ...ejar valores de

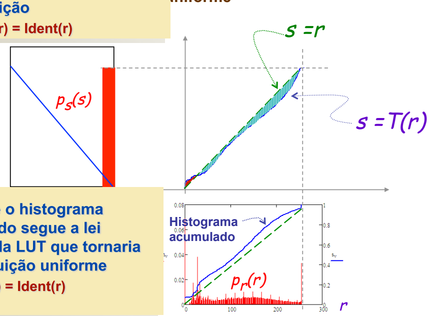


- Na distribuição un... intensidades estão... quantidades iguais...

- LUTs abaixo da identidade provocam o escurecimento da imagem
- LUTs acima da identidade clareiam

- Ou seja, se usarmos como T(r) o histograma $h_v(r)$, produziremos uma LUT capaz de equalizar a distribuição
  $h(r).T^{-1}(r) = Ident(r)$

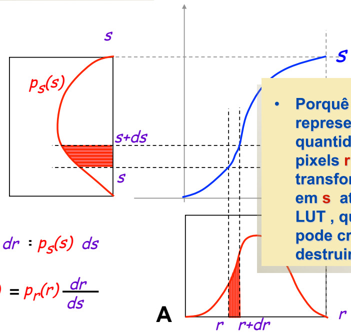
...a figura transfere valores ...mais escuros e valores ...n mais claros, em proporções ...uniforme



- Note que o histograma acumulado segue a lei inversa da LUT que tornaria a distribuição uniforme
  $T(r).T^{-1}(r) = Ident(r)$

- Porquê essas áreas devem ser iguais ?

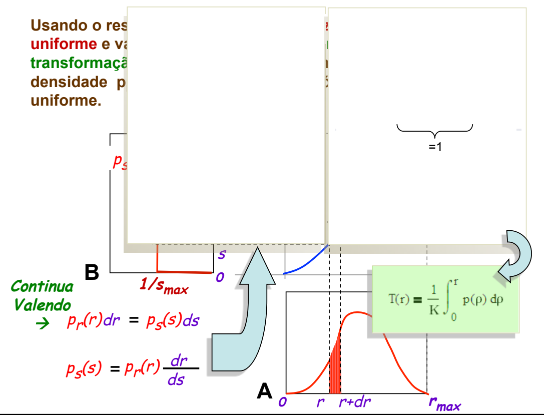
...r a forma de uma transformação ...T) que torne a densidade $p_r(r)$ ...densidade $p_s(s)$ da imagem B genérica.



$p_r(r)\ dr : p_s(s)\ ds$

$p_s(s) = p_r(r)\dfrac{dr}{ds}$

- Porquê elas representam as quantidades de pixels r que foram transformados em s através da LUT , que não pode criar nem destruir pixels



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

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. $0 \leq T(r) \leq 1$ para $0 \leq r \leq 1$, 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 $K = \frac{1}{s_{max}}$

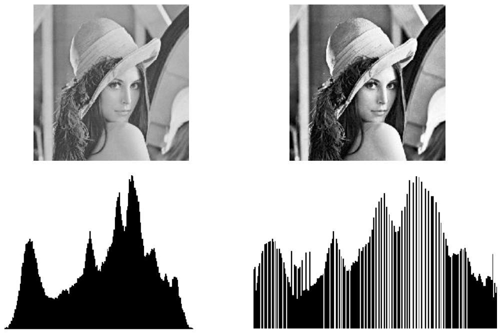
Caso discreto

- 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

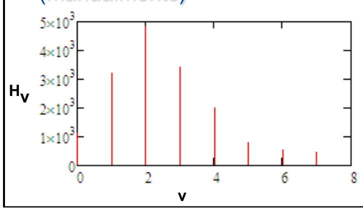
- Exercício – Proponha um algoritmo para equalizar imagens monocromáticas.
- Exercício – implementar (em Python p.ex.) esse algoritmo de equalização de imagens monocromáticas

Equalização de histograma

- Propriedade: a equalização de histograma é uma transformação idempotente (isto é, aplicar a transformação mais de uma vez não modifica o resultado)
  - Exercício:
    - **Demonstre isso.**

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	$r = v / v_{max}$	$H_v(r)$	$h_v(r)$	Σh_v	LUT	$s = T^{-1}(r)$	$h_v(s)$

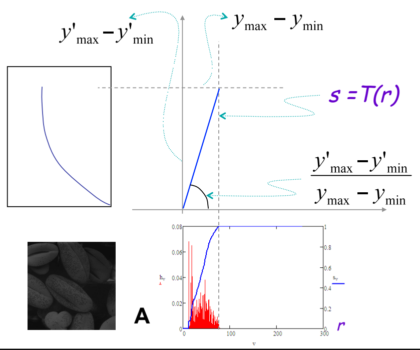
Normalização de Histograma

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

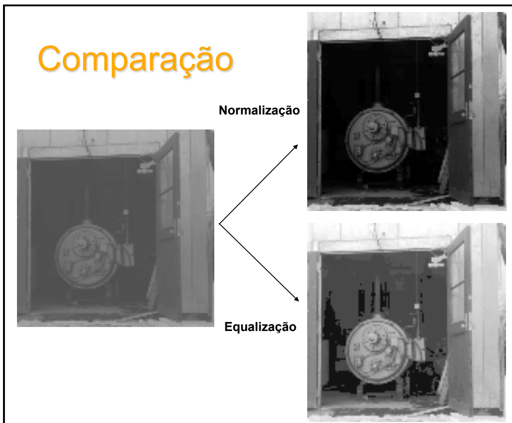
- A transformação é definida por:

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

$y_{max} - y_{min}$ = 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

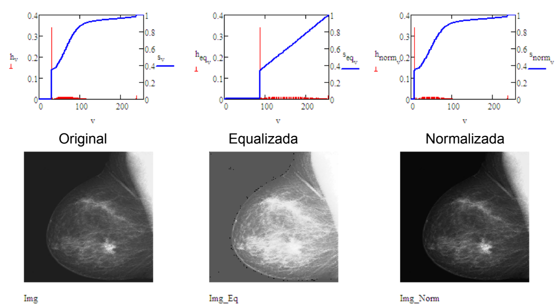
Comparação



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.

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

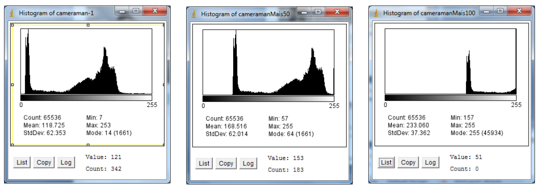
- onde $K_1(K_2)$ é o máximo valor possível da imagem de entrada (saída)

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.

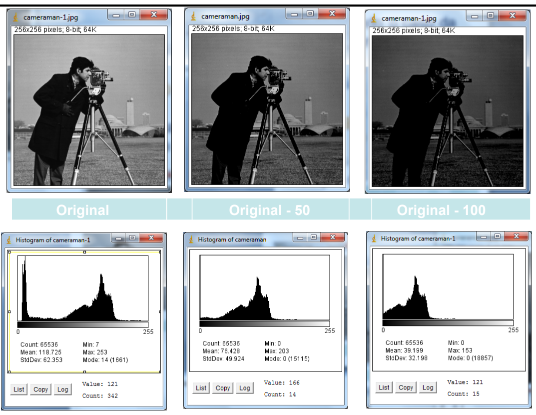
$$\dot{-}(a, b) = \begin{cases} a - b & \text{if } -K_1 - 1 \leq a - b \leq K_1 \\ -K_2 - 1 & \text{if } a - b < -K_1 - 1 \end{cases}$$

- 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



Operador Subtração

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

Operador Subtração

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

Operador Subtração

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

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