Conceitos básicos de processamento digital de imagens

Augusto de Holanda B. M. Tavares

Universidade Federal da Paraíba
Centro de Informática
Departamento de Sistemas de Computação

20 de junho de 2024

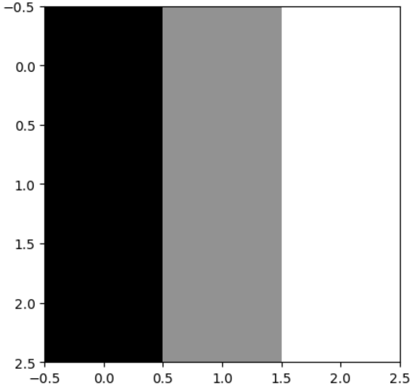
Imagens digitais

- Uma **imagem digital monocromática** (ou em **escala de cinza**) pode ser representada como um arranjo matricial em que o valor de cada um dos seus elementos representa a **intensidade luminosa**.
- Logo, uma imagem pode ser representada como:

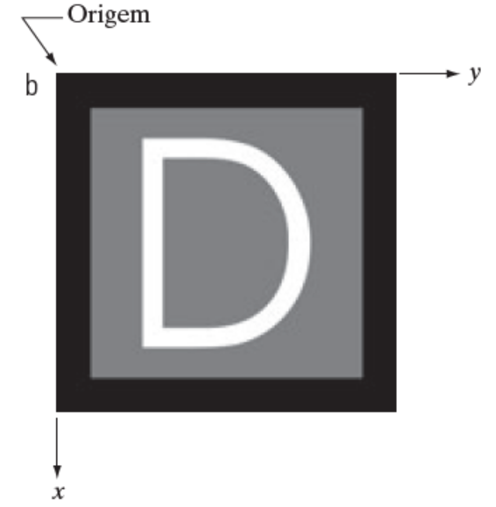
$$\boxed{f_a(x,y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}}$$

Imagens digitais

- A imagem equivalente ao arranjo apresentado anteriormente é:



Exemplo

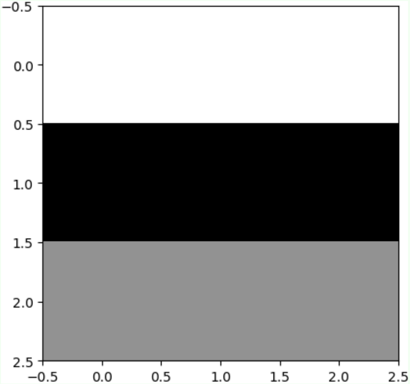


Imagens digitais

Exemplo

Exemplo

$$\boxed{f_b(x,y) = \begin{bmatrix} 7 & 7 & 7 \\ 0 & 0 & 0 \\ 4 & 4 & 4 \end{bmatrix}}$$



Imagens digitais

- As imagens abaixo são arranjos matriciais com 3 linhas e 3 colunas.

$$\boxed{f_a(x,y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}}$$

$$\boxed{f_b(x,y) = \begin{bmatrix} 7 & 7 & 7 \\ 0 & 0 & 0 \\ 4 & 4 & 4 \end{bmatrix}}$$

Imagens digitais

- Um elemento da imagem pode ser especificado por suas coordenadas no arranjo matricial, ou seja[1]:

$$\boxed{f_a(0,1) = 0;\ f_a(2,2) = 7}$$

$$\boxed{f_b(0,1) = 7;\ f_b(2,2) = 4}$$

- Os elementos da imagem são denominados **pixels**.
- As imagens digitais são definidas em um domínio discreto, de modo que as coordenadas de um *pixel* sempre serão números inteiros.

[1] É considerada uma indexação em 0, com o eixo x correspondendo ao eixo vertical orientado positivo para baixo e o eixo y correspondendo ao eixo horizontal orientado positivo para a direita.

Imagens digitais

- O número de pixels ao longo de um determinado eixo é a **resolução horizontal** ou a **resolução vertical** da imagem.
- Sendo M a resolução vertical e N a resolução horizontal, uma imagem $f(x, y)$ terá resolução $M \times N$.
- Em geral, quanto maior a resolução, melhor a qualidade da imagem.
- Efetivamente, a qualidade da imagem depende da **resolução espacial**: quantos *pixels* são utilizados por unidade de medida de distância.

Imagens digitais

- A relação entre os valores dos elementos do arranjo e a intensidade luminosa apresentada na imagem é determinada por convenção.
- Em uma imagem analógica, tem-se que um elemento de valor 0 corresponde ao preto e um elemento de valor 1 corresponde ao branco. Assim, os valores de intensidade estarão no intervalo $[0, 1] \in \mathbb{R}$.
- Para uma imagem digital qualquer, os elementos do arranjo tem os valores dentro de um intervalo $[0, 1, 2, \ldots, L - 1] \in \mathbb{N}$.
- A convenção determina que o 0 corresponda a ausência de luminosidade, ou seja, o preto, enquanto o valor máximo $L - 1$ é equivalente ao máximo de luminosidade, ou seja, o branco.

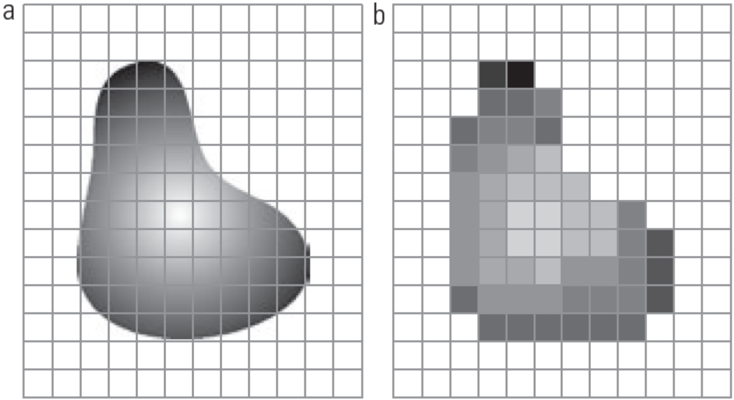
- A nível prático, o número de níveis de intensidade possível para cada um dos *pixels* é determinado por quantos *bits* são utilizados para armazenar esta informação.
- Por exemplo, com 3 *bits* por *pixel*, há $2^3 = 8$ níveis de intensidade possíveis por *pixel*. Neste caso $L = 8$, ou seja, os níveis de intensidade estarão no intervalo $[0, 7] \in \mathbb{N}$.
- De modo geral, as imagens são classificadas como **imagens de** k **bits**, tal que os níveis de intensidade possíveis são dados por:

$$\boxed{L - 1 = 2^k - 1}$$

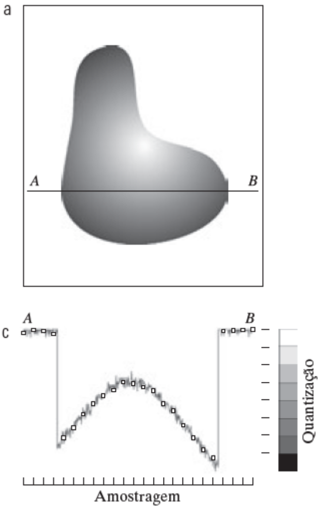
Imagens digitais

- O processo de converter uma imagem analógica para uma imagem digital necessita, então, de dois processos: a **amostragem** e a **quantização**.
- A amostragem é a definição do número e do espaçamento dos elementos do arranjo matricial correspondente de dimensão $M \times N$.
- A quantização é a atribuição do nível de intensidade luminosa medido originalmente ao intervalo de níveis de intensidade discreto $[0, L-1] \in \mathbb{N}$.

Imagens digitais



Imagens digitais



Imagens digitais

- Algumas propriedades são recorrentemente utilizadas ao se falar de processamento de imagens.
- A **faixa dinâmica** é a razão entre a intensidade máxima mensurável pela intensidade mínima detectável para um determinado sistema.
- O limite superior da faixa dinâmica é o limiar de **saturação**, enquanto o limite inferior é definido pelo **ruído**.
- O **constraste** de uma imagem é a diferença entre o maior e o menor nível de intensidade presentes.

Processamento digital de imagens

- Tendo definido o que é uma imagem digital, é possível definir o que é o **processamento digital de imagens** de maneira mais formal.
- Processar uma imagem equivale a realizar alguma operação matemática com os valores dos *pixels* no arranjo matricial de modo a:
  - Obter uma nova imagem, que é uma versão modificada da imagem original.
  - Extrair uma ou mais características da imagem original.
  - Atribuir algum significado à imagem original.

Processamento digital de imagens

Exemplo

Exemplo

- Considere a imagem abaixo:

$$\boxed{f_a(x,y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}}$$

Exemplo

- As bordas da imagem são preenchidas com zeros, resultando em:

$$\boxed{f'_a(x,y) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}$$

Exemplo

- Para cada um dos elementos da imagem original, é tomada a média das intensidades em uma região de dimensão 3×3 centrada neste elemento.

$$f_a'(x,y) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Exemplo

- O resultado é o elemento correspondente na imagem processada:

$$\boxed{g_a(0,0) = \frac{1}{9}(0+0+0+0+0+0+0+4+4) = 0,88}$$

- Assim, a imagem processada $g_a(x,y)$ será:

$$\boxed{g_a(x,y) = \begin{bmatrix} 0,88 & 2,44 & 2,44 \\ 1,33 & 3,66 & 3,66 \\ 0,88 & 2,44 & 2,44 \end{bmatrix}}$$

Exemplo

- Note que o resultado anterior apresenta valores de intensidade não inteiros. Para corrigir isto, é realizada a aproximação para o inteiro mais próximo:

$$\boxed{g_a(x,y) = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 4 & 4 \\ 1 & 2 & 2 \end{bmatrix}}$$

Exemplo

- Considerando $g_a(x, y)$, deseja-se obter $h_a(x, y)$ tal que:

$$\boxed{h_a(x, y) = \log(1 + g_a(x, y))}$$

- Ou seja, o nível de intensidade em um ponto (x, y) qualquer da nova imagem será o logaritmo de 1 somado com o nível de intensidade naquele mesmo ponto na imagem original.

Exemplo

- Lembrando que:

$$\boxed{g_a(x, y) = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 4 & 4 \\ 1 & 2 & 2 \end{bmatrix}}$$

- Consequentemente:

$$\boxed{h_a(x, y) = \log(1 + g_a(x, y)) = \begin{bmatrix} 0,3 & 0,47 & 0,47 \\ 0,3 & 0,69 & 0,69 \\ 0,3 & 0,47 & 0,47 \end{bmatrix}}$$

Exemplo

- Aproximando para o inteiro mais próximo:

$$\boxed{h_a(x,y) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}}$$

Exemplo

- Considere a imagem abaixo:

$$\boxed{g_a(x,y) = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 4 & 4 \\ 1 & 2 & 2 \end{bmatrix}}$$

- Defina a posição de todos os *pixels* cujo valor é igual a 2.

Exemplo

```
# g é a imagem de entrada
for i in range M:
   for j in range N:
      if g(M,N) == 2:
         h(M,N) = 1
      else:
         h(M,N) = 0
```

$$\boxed{h_a(x,y) = \begin{cases} 1 & , g_a(x,y) = 2 \\ 0 & , c.c. \end{cases}}$$

- Os exemplos anteriores evidenciam alguns fatos relevantes para as operações de processamento digital de imagens.
- O primeiro é que embora as imagens aparentem ser matrizes, na verdade estas são **arranjos matriciais**.
- Consequentemente, as operações são realizadas termo a termo, e não de acordo com as regras de operações matriciais.
- Isto abre o leque de operações possíveis, como a expressão utilizando o logaritmo mostrada anteriormente:

$$\boxed{h_a(x,y) = \log(1 + g_a(x,y))}$$

- Operações termo a termo entre arranjos matriciais exigem dimensões idênticas!

Processamento digital de imagens

Exemplo

$$f_a(x, y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}$$

$$f_b(x, y) = \begin{bmatrix} 7 & 7 & 7 \\ 0 & 0 & 0 \\ 4 & 4 & 4 \end{bmatrix}$$

$$f_a f_b = \begin{bmatrix} 0 & 28 & 49 \\ 0 & 0 & 0 \\ 0 & 16 & 28 \end{bmatrix}$$

$$f_a^2 = \begin{bmatrix} 0 & 16 & 49 \\ 0 & 16 & 49 \\ 0 & 16 & 49 \end{bmatrix}$$

Processamento digital de imagens

- Outro ponto relevante são as restrições devido à natureza das imagens digitais.
- O nível de intensidade correspondente a um determinado *pixel* de uma imagem de k *bits* sempre será um número inteiro no intervalo $[0, L-1]$.
- Uma operação qualquer pode, então, gerar dois problemas:
  - Níveis de intensidade não inteiros.
  - Níveis de intensidade fora do intervalo $[0, L-1]$.

- Quanto o resultado é não inteiro, o mais comum é aproximar o valor para o inteiro mais próximo.
- Nos casos em que o resultado fica fora do intervalo de valores possíveis, há diferentes alternativas, com suas vantagens e desvantagens:
  - Truncar os valores abaixo de 0 para 0 e acima de $L-1$ para $L-1$. Isto provoca saturação e/ou escurecimento invevido da imagem.
  - Repetir o processo de normalização para o intervalo $[0, L-1]$ com os novos valores. Isto modifica a escala de intensidades utilizada.
- Em alguns casos, operações intermediarias podem ter valores fora do intervalo $[0, L-1]$.

- Os processos aplicados à uma imagem podem ser agrupados em duas categorias principais:
  - Processos lineares.
  - Processos não lineares.

Processos lineares

- Para os nossos propósitos, um processo é uma operação entre funções 2D:

$$\boxed{g(x, y) = h(f(x, y))}$$

- Um processo será linear se for respeitada a seguinte propriedade (k_1 e k_2 são constantes):

$$\boxed{\begin{aligned} g(x, y) =& h(k_1 f_1(x, y) + k_2 f_2(x, y)) \\ g(x, y) =& k_1 h(f_1(x, y)) + k_2 h(f_2(x, y)) \end{aligned}}$$

- Caso contrário, tem-se um processo não linear.

Processos lineares

- Considerando que as imagens digitais são objetos discretos, há duas operações lineares que são de particular interesse no contexto do processamento de imagens:
  - Convolução discreta.
  - Transformada de Fourier discreta.
- A convolução permite que seja calculado o resultado da aplicação de uma máscara de constantes a uma determinada imagem, produzindo uma imagem modificada como resultado. Como a entrada e a saída são definidos no plano (x, y), este processo opera no domínio do espaço.
- A Transformada de Fourier permite que sejam obtidos o espectro e a fase de uma imagem para a análise de suas características e a aplicação de filtros. Como a entrada é uma imagem no espaço, e a saída é a transformada em frequência, este processo opera no domínio da frequência.

Processos não-lineares

- São processos que manipulam a imagem de acordo com operações não lineares.
- Exemplos comuns são:
  - Filtros de ordem estatística.
  - Filtros adaptativos.
  - Transformações de intensidade logarítmica, exponencial, etc.
- Note que as transformações de intensidade também podem ser lineares, a depender da forma da função de transformação.

Processamento digital de imagens

- Para a visão computacional, este conjunto de processos lineares e não-lineares constituem um ferramental da etapa de pré-processamento.
- Ou seja, os algoritmos são empregados para garantir que os dados referentes às imagens sejam fornecidos em uma qualidade tal que os algoritmos de extração de características e classificação consigam operar de modo satisfatório.
- Assim, a seguir serão apresentados alguns dos algoritmos mais importantes de processamento digital de imagens.

Transformações de intensidade

- São processos em que a intensidade r de cada um dos *pixels* da imagem de entrada é o argumento de alguma função de transformação $T(\cdot)$ que resulta em um *pixel* com intensidade s na imagem de saída.

$$\boxed{s = T(r),\ \{s, r\} \in [0, L-1]}$$

- Podem resultar em aumento ou diminuição de contraste, alargamento de faixa dinâmica, etc.
- Como não há restrição sobre o formato da função, não é necessariamente um processo linear.
- A função pode incluir parâmetros constantes que modificam o resultado do processo.

Transformações de intensidade

Exemplo

$$\boxed{s = c\log(1+r)}$$

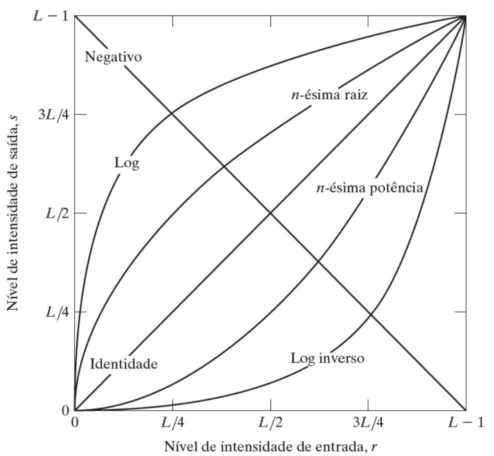
$$\boxed{s = cr^{\gamma}}$$

$$\boxed{s = (L-1) - r}$$

$$\boxed{s = \begin{cases} 0 & , r < \frac{L-1}{2} \\ L-1 & , r \geq \frac{L-1}{2} \end{cases}}$$

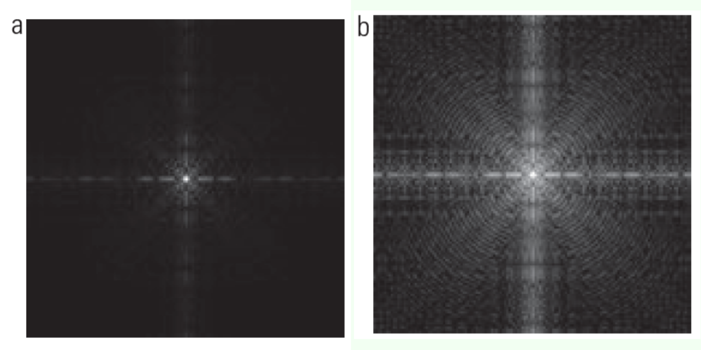
Transformações de intensidade

Exemplo



Transformações de intensidade

Exemplo

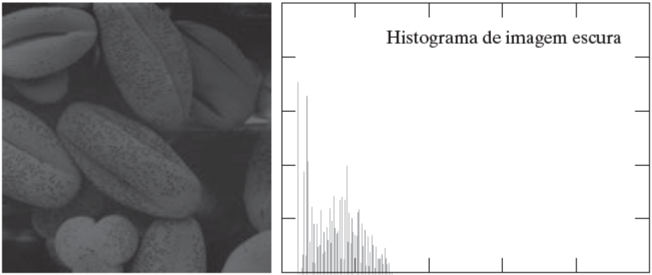


- a) Imagem original, b) Aplicação de transformada logarítmica.

Transformações de intensidade

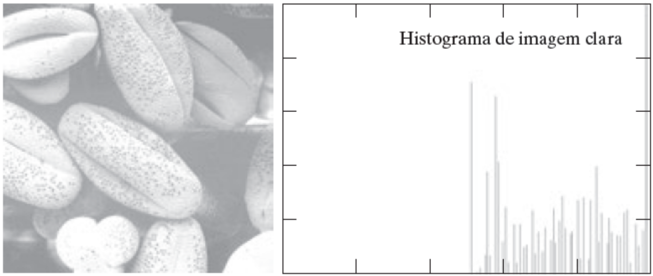
- Uma transformação de intensidade particularmente importante é a equalização de histograma.
- Um histograma é um gráfico onde para uma variável aleatória observada ao longo de múltiplos eventos é anotado o número de ocorrências de cada uma das suas possibilidades.
- Ao considerar o valor de intensidade de um *pixel* em uma imagem como uma variável aleatória, é construído um histograma com os 2^k valores de intensidade possíveis.
- Ao analisar a taxa de ocorrência de determinados níveis de intensidade, é possível intuir se a imagem correspondente a um determinado histograma é clara, escura, possui alto ou baixo contraste, etc.

Exemplo



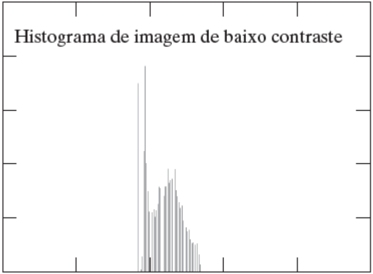
Transformações de intensidade

Exemplo



Transformações de intensidade

Exemplo



Transformações de intensidade

- A transformação abaixo equaliza o histograma de uma imagem, deixando-o o mais uniforme possível:

$$\boxed{s_k = T(r_k) = \frac{L-1}{MN}\sum_{j=0}^{k} n_j,\ k \in [0, L-1]}$$

- Onde:
  - k é o k-ésimo nível de intensidade presente na imagem.
  - M e N são as dimensões da imagem.
  - n_j é o número de ocorrências do nível de intensidade j no histograma da imagem.
  - Note que $\frac{n_j}{MN}$ é a probabilidade de um *pixel* qualquer ter o nível de intensidade j.

Transformações de intensidade

- O resultado da transformação será uma tabela da forma:

s_k	r_k
$s_0 = T(r_0)$	r_0
$s_1 = T(r_1)$	r_1
$\vdots$	$\vdots$
$s_{L-1} = T(r_{L-1})$	r_{L-1}

- Onde $s_k, r_k \in [0, L-1]$.
- Normalmente, o valor de s_k obtido pela transformação não será inteiro, devido a divisão $\frac{L-1}{MN}$.
- Para respeitar a natureza digital da imagem este será aproximado para o valor inteiro mais próximo. Esta aproximação é o motivo pelo qual a equalização não é perfeita.

Transformações de intensidade

- O histograma pode ser fornecido como um gráfico ou como uma tabela com o número de ocorrências de cada nível de intensidade:

r_k	n_k
$r_0 = 0$	n_0
$r_1 = 1$	n_1
$\vdots$	$\vdots$
$r_{L-1} = L-1$	n_{L-1}

- Sendo que necessariamente $\sum_{k=0}^{L-1} n_k = MN$.

Transformações de intensidade

Exemplo

- Considere uma imagem de 2 *bits* com dimensões $M = N = 200$ e*pixels* distribuídos de acordo com o seguinte histograma:

r_k	n_k
$r_0 = 0$	20.000
$r_1 = 1$	10.000
$r_2 = 2$	5.000
$r_3 = 3$	5.000

Transformações de intensidade

Exemplo

- Pela transformação de equalização, tem-se que ($L-1=3$):

$$\boxed{s_0 = T(r_0) = \frac{3}{40.000}\sum_{j=0}^{0} n_j = \frac{3}{40.000}(20.000) = 1,5}$$

$$\boxed{s_1 = T(r_1) = \frac{3}{40.000}\sum_{j=0}^{1} n_j = \frac{3}{40.000}(30.000) = 2,25}$$

Transformações de intensidade

Exemplo

$$\boxed{s_2 = T(r_2) = \frac{3}{40.000}\sum_{j=0}^{2} n_j = \frac{3}{40.000}(35.000) = 2,625}$$

$$\boxed{s_3 = T(r_3) = \frac{3}{40.000}\sum_{j=0}^{3} n_j = \frac{3}{40.000}(40.000) = 3}$$

Exemplo

- Assim, a tabela de transformação é dada por:

s_k	r_k
$s_0 = 1,5$	$r_0 = 0$
$s_1 = 2,25$	$r_1 = 1$
$s_2 = 2,625$	$r_2 = 2$
$s_3 = 3$	$r_3 = 3$

Transformações de intensidade

Exemplo

- Aproximando para o inteiro mais próximo:

s_k	r_k
$s_0 = 1$	$r_0 = 0$
$s_1 = 2$	$r_1 = 1$
$s_2 = 3$	$r_2 = 2$
$s_3 = 3$	$r_3 = 3$

Transformações de intensidade

Exemplo

- Abaixo, o histograma de entrada (a esquerda) e o histograma de saída (a direita):

r_k	n_k
$r_0 = 0$	20.000
$r_1 = 1$	10.000
$r_2 = 2$	5.000
$r_3 = 3$	5.000

$\Longrightarrow$

s_k	n_k
0	0
1	20.000
2	10.000
3	10.000

Convolução discreta

- A aplicação de um filtro no domínio do espaço é definida pela convolução de uma máscara, que é um arranjo matricial de constantes, com a imagem.
- A convolução é uma operação em que todos os valores de uma função são multiplicados ao menos uma vez por todos os valores da outra, sendo calculada a soma dos produtos em cada etapa.
- Esta operação fornece o resultado da aplicação de algum processo linear a uma função qualquer (no nosso caso, uma imagem).

Convolução discreta

- A convolução discreta 2D é apresentada abaixo:

$$\boxed{w(x,y) \circledast f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x-s, y-t)}$$

- Onde:
  - $w(x,y)$ é a máscara de convolução. Por praticidade, assume-se que suas dimensões são ímpares da forma $2a+1$ e $2b+1$ (tipicamente $a = b$).
  - $f(x,y)$ é a imagem.
  - s e t são variáveis locais.

Convolução discreta

- Considere o processo de obtenção de uma imagem com as médias das intensidades apresentado anteriormente.
- No caso utilizado no exemplo, era tomada a média aritmética da região 3×3 centrada em cada *pixel* da imagem.
- Isto é equivalente a convoluir a imagem com uma máscara da forma:

$$\boxed{w_a(x, y) = \begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}}$$

Convolução discreta

- Lembre-se que na convolução todos os elementos da imagem são multiplicados por todos os elementos da máscara ao menos uma vez.
- Isto gera um problema para máscaras com dimensões maiores que 1×1 nas bordas da imagem, já que a máscara iria "passar" dos limites da imagem.
- Isto evidencia a necessidade de preencher as bordas para realizar a convolução. Ao fim da operação, as bordas preenchidas são cortadas para que a imagem retorne às suas dimensões originais.
- No caso prático, considerando uma máscara de dimensão $m \times m$, onde m é ímpar, o preenchimento será de tamanho $m - 2$ em todas as direções. Estritamente falando, o correto seria $m - 1$, mas isto gera o cálculo de termos que serão descartados.

Convolução discreta

Exemplo

- Deseja-se calcular o resultado da convolução da imagem f_a pela máscara w_a.

$$\boxed{f_a(x, y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}}$$

$$\boxed{w_a(x, y) = \begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}}$$

Exemplo

- Como w_a tem dimensão $m = n = 3$, as bordas da imagem são preenchidas com zeros até a imagem ter suas dimensões aumentadas em $m - 2 = 1$, resultando em:

$$\boxed{f'_a(x,y) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}$$

Convolução discreta

Exemplo

- Em seguida, é tomado o o primeiro elemento da imagem original, $f_a(0,0) = 0$, agora $f_a'(1,1) = 0$, e é aplicada a fórmula da convolução discreta entre o arranjo centrado neste de mesma dimensão da máscara w_a e a máscara w_a.

$$\boxed{f_a'(x,y) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}$$

Exemplo

- Lembrando que $2a + 1 = m$, e que $a = b$ neste caso, tem-se:

$$\boxed{w_a(x, y) \circledast f'_a(1, 1) = \sum_{s=-1}^{1} \sum_{t=-1}^{1} w_a(s, t) f'_a(1 - s, 1 - t)}$$

- Esta fórmula se resume a calcular um produto de arranjos matriciais e tomar a soma dos elementos do resultado como sendo o elemento correspondente da imagem resultante.

Convolução discreta

Exemplo

- Ou seja

$$
\boxed{
\begin{aligned}
w_a(x,y) \circledast f'_a(x,y) =& \frac{1}{9}0 + \frac{1}{9}0 + \frac{1}{9}0 \\
& + \frac{1}{9}0 + \frac{1}{9}0 + \frac{1}{9}4 \\
& + \frac{1}{9}0 + \frac{1}{9}0 + \frac{1}{9}4 \\
w_a(x,y) \circledast f'_a(x,y) =& \frac{8}{9}
\end{aligned}
}
$$

Convolução discreta

Exemplo

- O resultado do processo é deslocado para $f'_a(1, 2)$ e repetido na nova coordenada:

$$\boxed{f'_a(x, y) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}$$

- Isto será repetido até que todos os *pixels* da imagem original tenham um resultado correspondente.

Convolução discreta

- O preenchimento é necessário para atender à restrição das operações entre arranjos: os operandos devem ter as mesmas dimensões.
- O preenchimento pode ser realizado com 0, 1 ou por repetição. O preenchimento por 0 e 1 é mais simples de implementar, mas interfere significativamente no resultado.
- O preenchimento por repetição exige uma implementação mais complexa, mas tende a influir menos no resultado.
- As máscaras geralmente são quadradas com dimensões ímpares para apresentar um eixo de simetria, o que facilita a sua implementação.

Convolução discreta

- Como foi mencionado anteriormente, os resultados da operação podem ser números não inteiros. Neste caso, deve ser realizada a aproximação para o inteiro mais próximo.
- Uma máscara pode ter uma dimensão mínima de 1×1 e no máximo a dimensão $M \times N$ da imagem.

Convolução discreta

- A importância do processo de convolução se deve ao fato de que a máscara $w(x, y)$ pode ter valores arbitrários como os seus coeficientes.
- A imagem resultante é, então, uma versão modificada da imagem original, cuja natureza específica da modificação depende dos coeficientes das máscaras.
- Estas máscaras são chamadas de **filtros**.
- A seguir são apresentadas alguns dos filtros mais comuns:

Filtros de média ou suavização

- Uma máscara cuja soma dos coeficientes seja igual a 1 é um filtro de média.
- O filtro pode não ter viés (média aritmética) ou pode aplicar alguma ponderação, dando mais ênfase a alguns elementos em detrimento dos demais.

$$\boxed{\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}} \qquad \boxed{\begin{bmatrix} 0,0625 & 0,0625 & 0,0625 \\ 0,0625 & 0,5 & 0,0625 \\ 0,0625 & 0,0625 & 0,0625 \end{bmatrix}}$$

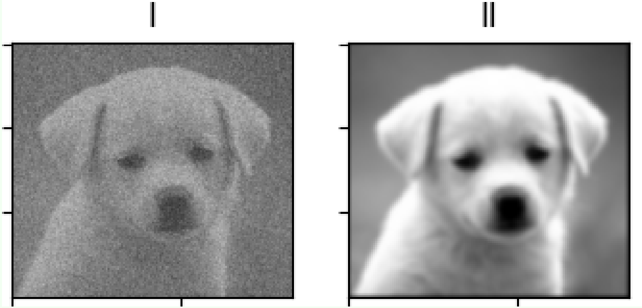
Filtros de média ou suavização

- Uma máscara maior será mais efetiva em tratar o ruído, mas irá borrar mais a imagem:

$$\boxed{\begin{bmatrix} \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} \\ \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} \\ \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} \\ \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} \\ \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} & \frac{1}{25} \end{bmatrix}}$$

Filtros de média ou suavização

Exemplo



- I) imagem submetida á ruído aleatório, II) imagem tratada com filtro de média aritmética

Filtros de média ou suavização

- Em imagens digitais, a presença de ruído é caracterizada como um ou mais *pixels* tendo o seu valor de intensidade alterado de maneira indesejada.
- Isto implica que ao analisar a variação de intensidade, haverá descontinuidades correspondentes aos *pixels* afetados pelo ruído, assumindo que este seja distribuído de maneira aleatória.
- Logo, os filtros de média são efetivos no tratamento do ruído ao uniformizar estas variações de intensidade.
- No entanto, variações abruptas de intensidade também podem corresponder a regiões de borda na imagem, que são os detalhes. Isto faz com que estes filtros causem um efeito indesejado de borramento.

Filtros de aguçamento

- Estes filtros cupem o papel oposto dos filtros de suavização: eles enfatizam as variações de intensidade em uma imagem.
- Isto realça os detalhes, que correspondem às regiões de borda.
- Por outro lado, caso haja ruído na imagem este também será enfatizado.
- Assim como há diferentes filtros de suavização, também há múltiplos filtros de aguçamento.
- O processo de aguçamento tipicamente passa por obter uma imagem onde apenas as bordas estão presentes a partir de um filtro e somar esta a imagem original.

Filtros de aguçamento

- Filtro laplaciano:

$$\boxed{w_L(x,y) = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}} \qquad \boxed{w_L(x,y) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}}$$

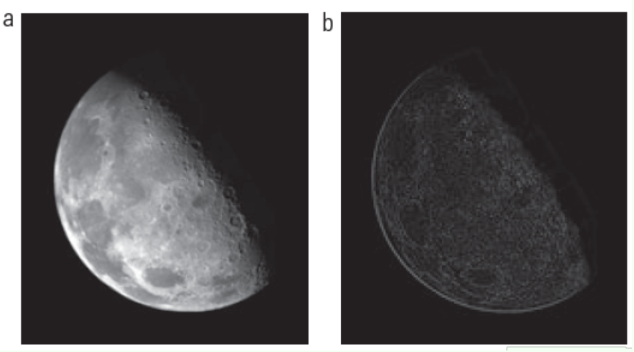
- A imagem com realce é obtida a partir de:

$$\boxed{g(x,y) = f(x,y) + c(w_L(x,y) \circledast f(x,y))}$$

- Sendo $c = -1$ se o elemento central de $w(x,y)$ for negativo e $c = 1$ se o elemento central de $w(x,y)$ for positivo.

Filtros de aguçamento

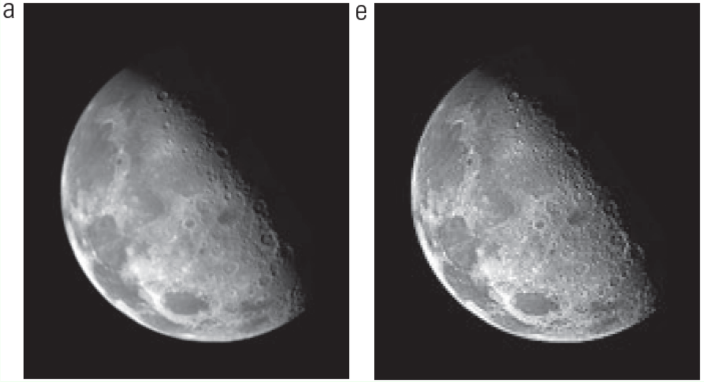
Exemplo



- a) Imagem original, b) Laplaciano.

Filtros de aguçamento

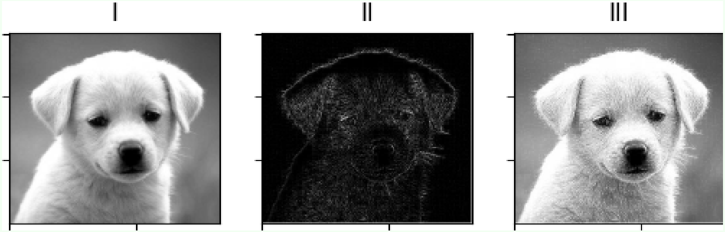
Exemplo



- a) Imagem original, e) Imagem aguçada com laplaciano (c=-1).

Filtros de aguçamento

Exemplo



- I) Imagem original, II) Laplaciano, III) Imagem aguçada com laplaciano (c=1).

Filtros de aguçamento

- Máscaras de Sobel:

$$\boxed{w_x(x,y) = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}} \qquad \boxed{w_y(x,y) = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}}$$

- O gradiente da imagem é dado por:

$$\boxed{g'(x,y) = w_x(x,y) \circledast f(x,y) + w_y(x,y) \circledast f(x,y)}$$

Filtros de aguçamento

- Um processo de aguçamento pode ser obtido a partir de uma aproximação utilizando um filtro de média qualquer. Esta é a **máscara de nitidez** ou **filtragem high-boost**.
- Seja $w_m(x, y)$ uma máscara correspondente a um filtro de média qualquer. A imagem suavizada $\bar{f}(x, y)$ é obtida a partir de $f(x, y)$ por:

$$\boxed{\bar{f}(x, y) = w_m(x, y) \circledast f(x, y)}$$

- E a máscara do processo será dada por:

$$\boxed{g'(x, y) = f(x, y) - \bar{f}(x, y)}$$

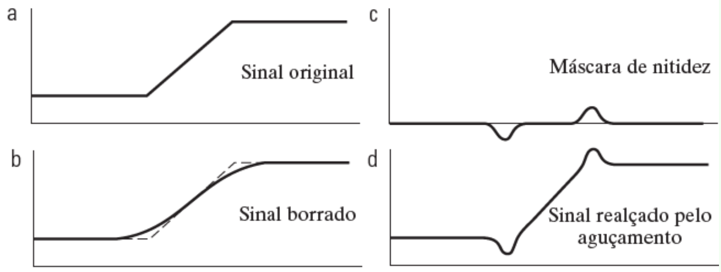
Filtros de aguçamento

- A imagem aguçada $g(x,y)$ é, então:

$$\boxed{g(x,y) = f(x,y) + k\,g'(x,y)}$$

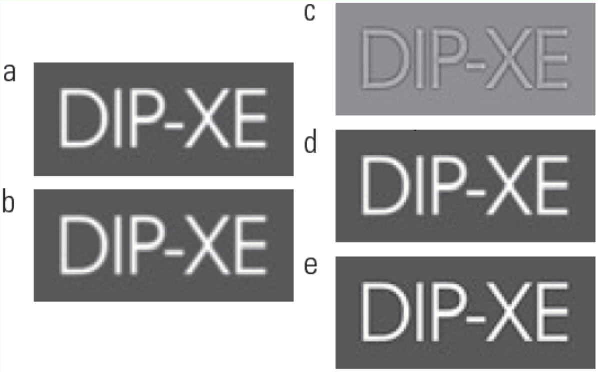
- $k \geq 0$ é uma constante de ajuste.
- Para $k = 1$ tem-se a máscara de nitidez simples.
- $k > 1$ é a filtragem *high-boost*, que enfatiza o aguçamento. $k < 1$ diminui a sua contribuição.
- A vantagem é obter uma implementação simples que emula o aguçamento de outros filtros a partir de um filtro de média.
- Ainda há outro ajuste possível ao alterar o tipo de filtro de média $w_m(x,y)$ utilizado.

Exemplo


a
Sinal original
b
Sinal borrado
c
Máscara de nitidez
d
Sinal realçado pelo aguçamento

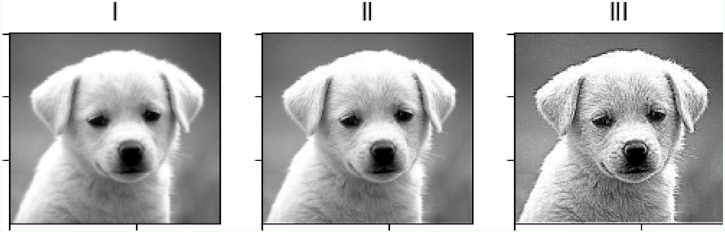
Filtros de aguçamento

Exemplo



Filtros de aguçamento

Exemplo



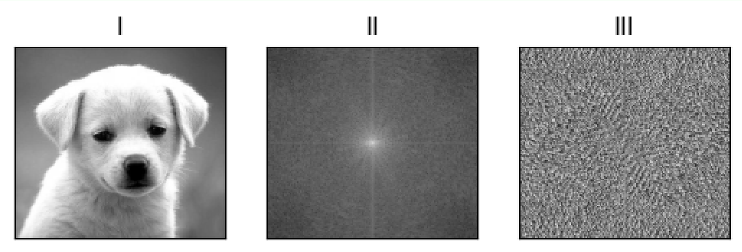
- I) Imagem original, II) Máscara de nitidez, III) Filtragem *high-boost*.

Transformada de Fourier

- A transformada de Fourier é uma operação matemática que transpõe uma imagem do domínio do espaço para o domínio da frequência.
- Sendo uma imagem um arranjo matricial 2D onde cada um dos *pixels* é a intensidade em um ponto, a transformada de Fourier da imagem será um arranjo de dimensões corresponentes onde cada um dos pontos é um **número complexo**, que é o valor da transformada naquele ponto.
- Como números complexos possuem duas dimensões, um gráfico da transformada teria 4 dimensões, o que não é representável.
- Sendo assim, a transformada é separada em **módulo**, ou **espectro**, e **fase**.

Transformada de Fourier

Exemplo



- I) Imagem original, II) Espectro, III) Fase.

- O espectro contém as informações de intensidade luminosa.
- A fase contém as informações de detalhes da imagem.

Transformada de Fourier

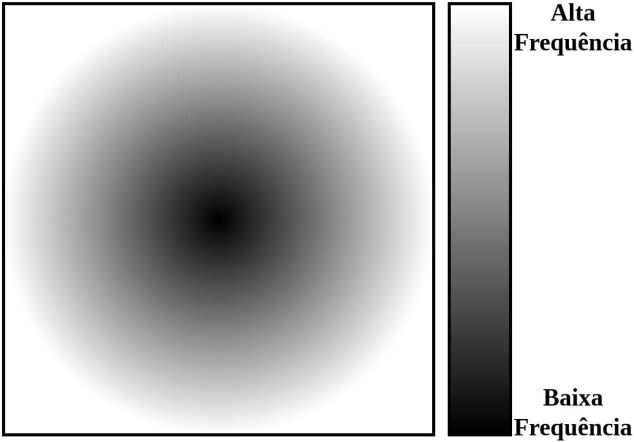
- Qual é, então, o propósito de utilizar a transformada?
- Para além de *insights* a partir da análise da fase e do espectro, a principal vantagem é o **teorema da convolução**.

Teorema da Convolução

- A convolução no domínio do espaço é equivalente ao produto no domínio da frequência, e vice-versa.

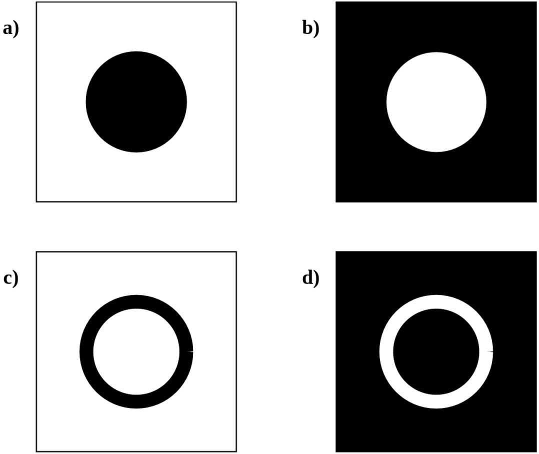
- Ou seja, troca-se uma operação complexa no espaço, que é a convolução, por um produto simples.
- Sendo assim, a filtragem no domínio da frequência envolve multiplicar a transformada de uma imagem por algum arranjo matricial desejado, ao invés de realizar uma convolução.

Transformada de Fourier



- As coordenadas de frequência são tipicamente centradas no plano da imagem (note que há 2 eixos).

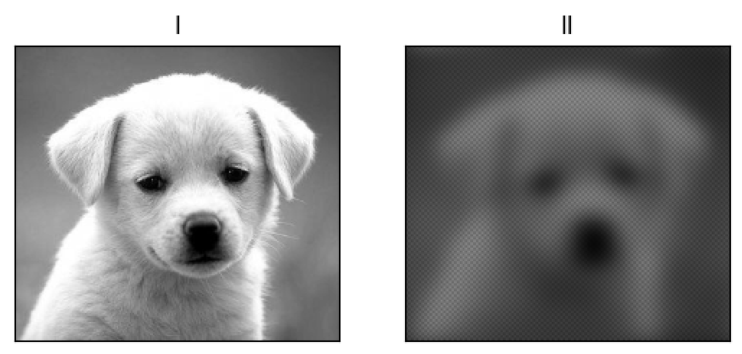
Transformada de Fourier



- a) Filtro passa-alta, b) Filtro passa-baixa, c) Filtro rejeita-faixa, d) Filtro passa-faixa.

Transformada de Fourier

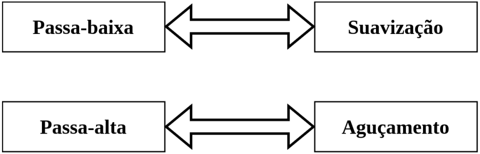
Exemplo



- I) Imagem original, II) Imagem submetida a um filtro passa-baixa gaussiano.

Transformada de Fourier

- Há uma equivalência entre os filtros em frequência e as máscaras utilizadas na filtragem espacial:



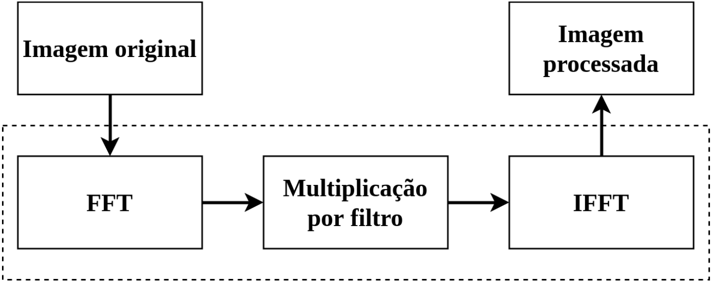
- Note que os filtros passa e rejeita-faixa não possuem um equivalente espacial direto.
- Efetivamente, como a forma geométrica do filtro em frequência é arbitrária, a filtragem em frequência apresenta maior flexibilidade do que a espacial.

Transformada de Fourier

- Embora importantes, a compreensão das minúcias analíticas e dos métodos de cálculo da transformada de Fourier para funções 2D não são objeto desta disciplina.
- Ainda, a implementação da transformada é por si só um problema complexo, visto que na sua forma básica o número de operações necessárias para imagens pequenas é da ordem de 10^{12}.
- Sendo assim, caso necessário serão utilizadas implementações da **Fast Fourier Transform (FFT)** e da **Inverse Fast Fourier Transform (IFFT)** disponíveis em bibliotecas.
- O número de operações destas implementações cresce com o logaritmo do tamanho da imagem, tornando-as práticas para uso em processamento de imagens.

Transformada de Fourier

- Sendo assim, o processo de filtragem em frequência é resumido em especificar a geometria do filtro para obter o efeito desejado e utilizar a FFT e a IFFT para transitar para e de o domínio da frequência.



Filtros de ordem estatística

- Outro método de filtragem espacial envolve o uso de vetores ordenados.
- Considere, novamente, uma imagem com as bordas preenchidas:

$$\boxed{f'_a(x,y) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}$$

Filtros de ordem estatística

- Adotando a mesma lógica da convolução, define-se uma janela de dimensões ímpares centrada em um *pixel*:

$$\boxed{f_a'(x,y) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 4 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}$$

- Os elementos desta sub-região podem ser expressos como um vetor:

$$\boxed{\begin{bmatrix} 0 & 0 & 0 & 0 & 4 & 7 & 0 & 4 & 7 \end{bmatrix}}$$

Filtros de ordem estatística

- O vetor é, então, ordenado:

$$\boxed{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 4 & 4 & 7 & 7 \end{bmatrix}}$$

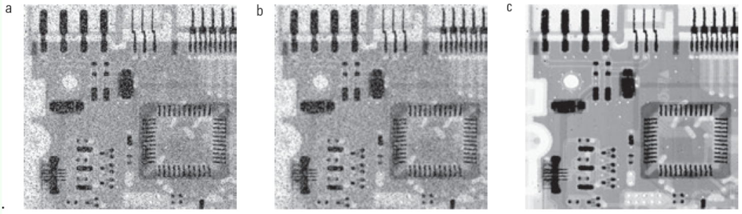
- Tendo o vetor ordenado, é definido algum critério relativo a um percentil (0%, 30%, 83%, etc.) que seleciona um dos elementos do vetor.
- Por exemplo, no vetor acima os elementos para os percentis 0% (mínimo), 50% (mediana) e 100% (máximo) seriam 0, 0 e 7, respectivamente.
- O valor do *pixel* correspondente na imagem processada será o valor deste elemento.
- Este processo é aplicado a todos os *pixels* da imagem.

Filtros de ordem estatística

- Os filtros de ordem estatística possuem a capacidade de tratar o ruído impulsivo, que não é bem tratado por filtros de suavização.
- Ainda, ao tratar este tipo de ruído, estes filtros não apresentam o efeito colateral de borramento.
- No entanto, filtros de ordem estatística mal aplicados podem resultar artefatos indesejados na imagem processada.
- Outros graus de flexibilidade envolvem gerar recortes dos vetores ordenados antes de selecionar algum dos elementos.
- As operações envolvidas são não lineares por natureza.

Filtros de ordem estatística

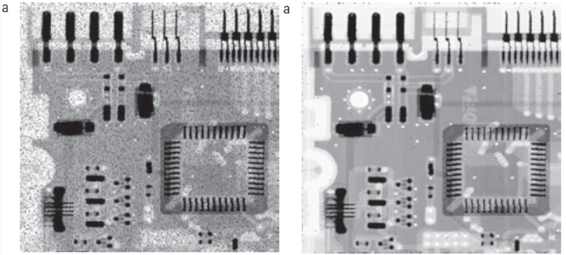
Exemplo



- a) Imagem submetida a ruído sal e pimenta, b) Imagem tratada com filtro de média, c) Imagem tratada com filtro de mediana.

Filtros de ordem estatística

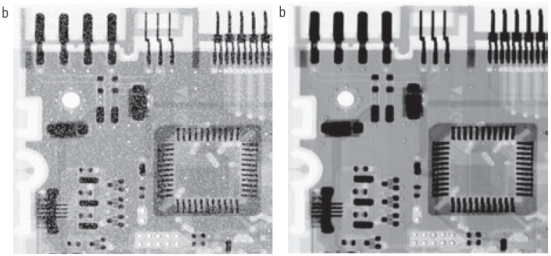
Exemplo



- a) Imagem submetida a ruído pimenta, b) Imagem tratada com filtro de máximo.

Filtros de ordem estatística

Exemplo



- a) Imagem submetida a ruído sal, b) Imagem tratada com filtro de mínimo.

Filtros adaptativos

- Os processos apresentados anteriormente são aplicados de maneira indiscriminada a uma imagem qualquer. Ou seja, todos os *pixels* passam pelo mesmo processo.
- Esta abordagem tem a vantagem de ser simples, mas também leva a efeitos indesejados.

Exemplo

- Um filtro de média age sobre toda a imagem para atenuar o ruído e causando borramento. Este efeito ocorre independentemente de uma região estar afetada pelo ruído ou não.

Filtros adaptativos

- Um **filtro adaptativo** é um processo cujas características variam de acordo com os parâmetros da região local sendo processada.
- Isto permite uma maior granularidade sobre como os efeitos do processo aplicados a uma imagem qualquer, ao custo de uma formulação mais complexa.
- Em termos práticos, isto envolve a inclusão de instruções condicionais no código do filtro.

Filtro adaptativo de mediana

Exemplo

- O filtro de mediana é efetivo apenas para densidades de ruído impulsivo relativamente baixas.
- O filtro de mediana também substitui todos os *pixels* pela mediana da região local, mesmo que o *pixel* no centro da janela de filtragem não tenha sido afetado pelo ruído. Isso gera distorções na imagem.
- É possível, então, formular um filtro adaptativo de mediana que consegue atuar em ruídos impulsivos de maior densidade, além de ser melhor em preservar os detalhes.

Filtro adaptativo de mediana

Exemplo

- Sejam:
  - z_{max} a maior intensidade na região local $S_{x,y}$.
  - z_{min} a menor intensidade em $S_{x,y}$.
  - z_{med} a mediana em $S_{x,y}$.
  - $z_{x,y}$ a intensidade em (x, y).
  - S_{max} o maior tamanho permitido para $S_{x,y}$.

Filtro adaptativo de mediana

Exemplo

- A execução do filtro é dividida em duas etapas:
  - Etapa A:
    1. Calcule $A_1 = z_{\text{med}} - z_{\text{min}}$ e $A_2 = z_{\text{med}} - z_{\text{max}}$. Se $A_1 > 0$ e $A_2 < 0$, vá para a etapa B.
    2. Senão, aumente o tamanho de $S_{x,y}$. Se $S_{x,y} \leq S_{\text{max}}$, volte para o início. Caso contrário, a saída é z_{med}.
  - Etapa B:
    1. Calcule $B_1 = z_{x,y} - z_{\text{min}}$ e $B_2 = z_{x,y} - z_{\text{max}}$. Se $B_1 > 0$ e $B_2 < 0$, a saída é $z_{x,y}$.
    2. Caso contrário, a saída é z_{med}
- O algoritmo acima tem 3 finalidades: remover o ruído sal e pimenta (impulsivo), proporcionar a suavização de ruídos não impulsivo e reduzir distorções devido ao processo de filtragem.

Filtro adaptativo de mediana

Exemplo

- Sob o ponto de vista do algoritmo z_{min} e z_{max} são componentes "similares a impulsos" na região local.
- Sendo assim, a etapa A verifica se a mediana z_{med} é um impulso ou não. Caso se cumpra que $z_{\text{min}} < z_{\text{med}} < z_{\text{max}}$, a mediana não é um impulso, e passa-se para a etapa B.
- Na etapa B verifica-se se $z_{x,y}$ é um impulso. Caso se cumpra que $z_{\text{min}} < z_{x,y} < z_{\text{max}}$, o valor central não é um impulso. Neste caso, a saída é $z_{x,y}$. Ou seja, pontos intermediários não são alterados, o que reduz as distorções.

Exemplo

- Se não se cumpre que $z_{\text{min}} < z_{x,y} < z_{\text{max}}$, tem-se que $z_{x,y} = z_{\text{min}}$ ou $z_{x,y} = z_{\text{max}}$. Isso significa que o ponto central é um valor extremo, e a saída será z_{med}, que se sabe não ser um ruído impulsivo, já que a condição da etapa A foi cumprida. Isso equivale ao filtro de mediana padrão, que trata o ruído impulsivo mas leva a perda de detalhes.

Filtro adaptativo de mediana

Exemplo

- Caso na etapa A não se cumpra que $z_{\text{min}} < z_{\text{med}} < z_{\text{max}}$, isso significa que a mediana é um impulso. Neste caso, a janela centrada em $z_{x,y}$ é aumentada, e o processo de verificação é retomado caso o tamanho máximo não tenha sido extrapolado, numa tentativa de chegar na etapa B. Caso contrário, a saída será z_{med}, que pode ser um impulso.

Filtro adaptativo de mediana

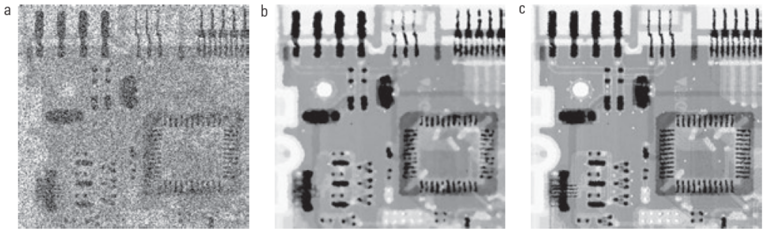
Exemplo

- Uma abordagem possível para a implementação em código seria:

```
def adaptive_median_filter(S_xy,S_max,
                           z_min,z_max,z_med):
   while S_xy <= S_max:
      if z_med - z_min > 0 && z_med - z_max < 0:
         if z_xy - z_min > 0 && z_xy - z_max < 0:
            return z_xy
         else:
            return z_med
      else:
         S_xy = S_xy + 1
   return z_med
```

Filtro adaptativo de mediana

Exemplo



- a) Imagem submetida a ruído impulsivo de alta intensidade, b) restauração com filtro de mediana, c) restauração com filtro adapativo de mediana.

Imagens coloridas

- Os conceitos explorados anteriormente tratam de imagens monocromáticas, ou imagens em escala de cinza.
- Nas imagens monocromáticas a única informação atrelada a cada um dos *pixels* é o nível de intensidade luminosa.
- Visto que o ser humano é capaz de visualizar uma faixa (estreita) do espectro eletromagnético como sinais luminosos de diferentes cores, é importante definir como os conceitos vistos para imagens monocromáticas se estendem para imagens coloridas.

Imagens coloridas

- Várias das decisões relativas ao processamento de imagens são informadas pelas características do sistema visual humano, visto que este é o público alvo.
- Sendo assim, as subjetividades e limitações deste sistema definem parte dos padrões e sistemas utilizados na manipulação de imagens coloridas.
- O principal destes fatores é que os cones, que são as células do olho sensíveis a cor, são sensíveis aos comprimentos de onda do espectro eletromagnético correspondentes ao vermelho, verde e azul.

Imagens coloridas

- Tendo este fato como base, a CIE determinou um sistema com três fontes luminosas de comprimento de onda padrão:
  - Vermelho $= 435,9\ nm$.
  - Verde $= 546,1\ nm$.
  - Azul $= 700\ nm$.
- Note que estas fontes primárias são um conceito ideal, visto que qualquer fonte real terá espalhamento espectral.
- A CIE estabelece um espaço de cores que cuja formação é possível a partir da combinação destas fontes.
- Este espaço **não** engloba todas as cores possíveis. Estes valores são apenas um padrão para garantir a consistência entre diferentes dispositivos de visualização.

Imagens coloridas

- Seria possível, então, especificar qualquer cor como uma combinação linear em determinada proporção das fontes de cores padrão. No entanto, isto é pouco intuitivo.
- Consequentemente, costuma-se distinguir as cores a partir de sua **matiz (hue)**, **saturação (saturation)** e **brilho (brightness/intensity)**.
- A matiz é a cor dominante percebida, associada ao comprimento de onda da cor.
- Ou seja, ao falar que um objeto é vermelho, estamos declarando que o seu matiz é vermelho. Um objeto laranja possui matiz laranja, e assim por diante.

Imagens coloridas

- A saturação é o grau de branco presente em uma cor percebida, também interpretado como o seu grau de pureza.
- Cores muito saturadas são mais vivas e intensas (verde limão, vermelho sangue), enquanto as cores menos saturadas tendem aos tons pastéis (lilás, rosa claro).
- Quanto mais branco compõe a cor, menor a sua saturação.
- O brilho é a intensidade luminosa percebida.

Imagens coloridas

- Se definem os **valores triestímulo** como as quantidades X, Y e Z de vermelho, verde e branco, respectivamente, utilizadas para combinar uma cor qualquer.
- Os **coeficientes tricromáticos** são os valores tri-estímulo normalizados:

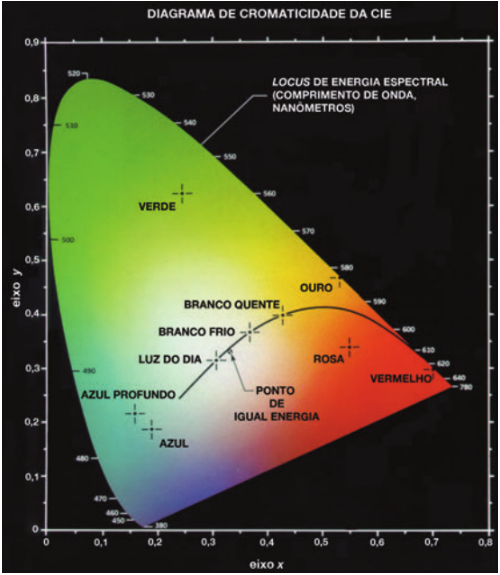
$$\boxed{x = \frac{X}{X+Y+Z}} \qquad \boxed{y = \frac{Y}{X+Y+Z}} \qquad \boxed{z = \frac{Z}{X+Y+Z}}$$

- Consequentemente:

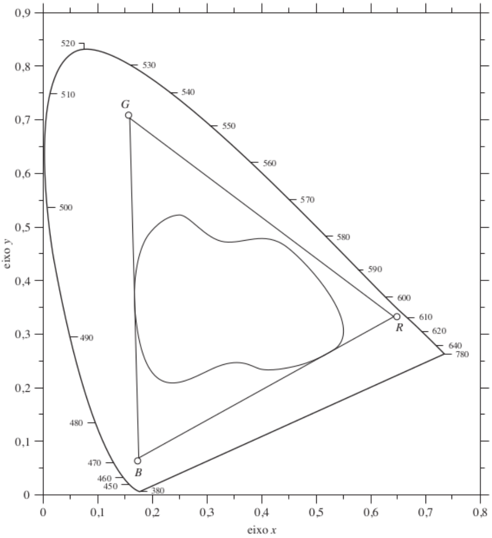
$$\boxed{x + y + z = 1}$$

- Ou seja, uma cor pode ser definida por apenas dois coeficientes, permitindo encontrar o terceiro.

- O diagrama de cromaticidade da CIE especifica as cores do espectro visível em termos dos coeficientes x e y.



- É possível, então, comparar a capacidade de representação de cores de um monitor com o padrão da CIE.
- Note que não há como definir três fontes primárias que englobem todas as cores possíveis.



Imagens coloridas

- A formulação de cores refletida anteriormente define um fato importante: uma cor é completamente especificada por três parâmetros.
- Sendo assim, são definidos modelos que permitem representar uma cor como um ponto no espaço
- Dois dos modelos mais comuns para imagens digitais são os modelos RGB e HSI.
- Para efeitos práticos, uma imagem colorida é um arranjo onde cada um dos pontos contém três valores, um para cada componente do modelo.
- Alternativamente, pode-se dizer que uma imagem é representada por três arranjos em paralelo.

Imagens coloridas

- No **modelo RGB**, cada um dos canais da imagem contém a informação correspondente a intensidade de uma das fontes de cor primárias.
- Ao lado é apresentada uma imagem de 3 *bits* com *pixels* pretos, cinzas e brancos no modelo RGB.
- Para $f_r = f_g = f_b$ uma imagem no modelo RGB se equipara a uma imagem monocromática.

$$\boxed{f_r(x,y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}}$$

$$\boxed{f_g(x,y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}}$$

$$\boxed{f_b(x,y) = \begin{bmatrix} 0 & 4 & 7 \\ 0 & 4 & 7 \\ 0 & 4 & 7 \end{bmatrix}}$$

Imagens coloridas

- Imagem de 3 *bits* com *pixels* azuis no modelo RGB.

$$f_r(x, y) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$f_g(x, y) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$f_b(x, y) = \begin{bmatrix} 7 & 7 & 7 \\ 7 & 7 & 7 \\ 7 & 7 & 7 \end{bmatrix}$$

Imagens coloridas

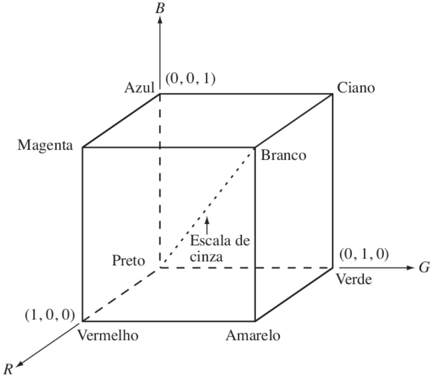
- Há uma relação entre o nível de intensidade de um dos canais de cores e a matiz, saturação e valor da cor observada, mas ela nem sempre é intuitiva.
- Por exemplo, qual é a cor ao lado?
- O problema é maior ainda para o caso inverso: intuir os valores RGB a partir de uma cor qualquer.

$$\boxed{f_r(x,y) = \begin{bmatrix} 3 & 3 & 3 \\ 3 & 3 & 3 \\ 3 & 3 & 3 \end{bmatrix}}$$

$$\boxed{f_g(x,y) = \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}}$$

$$\boxed{f_b(x,y) = \begin{bmatrix} 7 & 7 & 7 \\ 7 & 7 & 7 \\ 7 & 7 & 7 \end{bmatrix}}$$

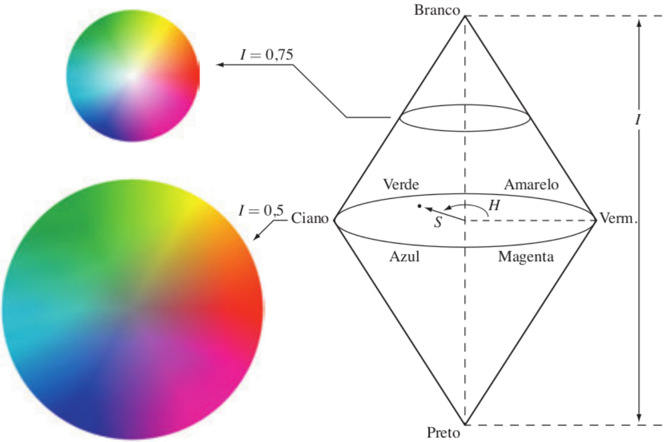
- Cubos de cores do espaço RGB.



Imagens coloridas

- O **modelo HSI** se coloca como uma alternativa mais prática para a descrição de cores.
- Seus parâmetros são justamente a matiz, a saturação e o brilho, que se adequam a como as cores costumam ser descritas.
- A contraposição é que embora o modelo RGB seja prático para implementação de imagens digitais e corresponda ao comportamento do sistema visual humano, utilizar o modelo HSI pode ser mais intuitivo.

Imagens coloridas



- Modelo HSI: A matiz é o ângulo, a saturação o raio e a intensidade a altura do plano.

Imagens coloridas

- De maneira análoga ao modelo RGB, o imagens coloridas no modelo HSI podem ser representadas por um conjunto de três arranjos
- A imagem de 8 *bits* ao lado possui matiz vermelha e baixa saturação, mas como a intensidade é máxima ela será observada como branco.

$$\boxed{f_h(x,y) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}$$

$$\boxed{f_s(x,y) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}$$

$$\boxed{f_i(x,y) = \begin{bmatrix} 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \end{bmatrix}}$$

Imagens coloridas

- Por sua vez, a imagem ao lado apresentará matiz azul com saturação máxima e intensidade intermediária.

$$f_h(x,y) = \begin{bmatrix} 160 & 160 & 160 \\ 160 & 160 & 160 \\ 160 & 160 & 160 \end{bmatrix}$$

$$f_s(x,y) = \begin{bmatrix} 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \end{bmatrix}$$

$$f_i(x,y) = \begin{bmatrix} 128 & 128 & 128 \\ 128 & 128 & 128 \\ 128 & 128 & 128 \end{bmatrix}$$

- Sejam R, G e B os valores de intensidade das componentes de uma imagem RGB, e H, S e I os componentes de uma imagem HSI.
- A matiz H pode ser obtida a partir dos valores RGB utilizando:

$$\boxed{H = \begin{cases} \frac{\theta}{360}, & B \leq G \\ \frac{360-\theta}{360}, & B > G \end{cases}}$$

- Onde,

$$\boxed{\theta = \cos^{-1}\left(\frac{0,5\left[(R-G)+(R-B)\right]}{\sqrt{(R-G)^2+(R-B)(R-G)}+\epsilon}\right),\ 0 < \epsilon << 1}$$

- A saturação S é obtida a partir dos valores RGB por:

$$\boxed{S = 1 - \frac{3\min(R, G, B)}{R + G + B}}$$

- E a intensidade I será dada por:

$$\boxed{I = \frac{R + G + B}{3}}$$

- As expressões acima presumem que os valores RGB foram normalizados para o intervalo $[0, 1]$. Os resultados estarão no mesmo intervalo.

- O caminho inverso, ou seja, ir de uma imagem HSI para uma imagem RGB possui conjuntos de fórmulas diferentes de acordo com o valor de H.
- Setor RG ($0^\circ \leq H < 120^\circ$):

$$\boxed{B = I(1 - S)}$$

$$\boxed{R = I\left[1 + \frac{S \cos H}{\cos(60^\circ - H)}\right]}$$

$$\boxed{G = 3I - (R + B)}$$

Imagens coloridas

- Setor RG ($120^\circ \leq H < 240^\circ$):
  - Faça $H = H - 120^\circ$.

$$\boxed{R = I(1 - S)}$$

$$\boxed{G = I\left[1 + \frac{S\cos H}{\cos(60^\circ - H)}\right]}$$

$$\boxed{B = 3I - (R + G)}$$

- Setor RG ($240^\circ \leq H < 360^\circ$):
  - Faça $H = H - 240^\circ$.

$$\boxed{G = I(1 - S)}$$

$$\boxed{B = I\left[1 + \frac{S\cos H}{\cos(60^\circ - H)}\right]}$$

$$\boxed{R = 3I - (G + B)}$$

Imagens coloridas

- Em geral, todos os processos apresentados anteriormente podem ser aplicados a imagens coloridas RGB ou HSV.
- O que irá diferir é que na aplicação podem ser feitas combinações:
  - Um mesmo processo pode ser aplicado nos três canais.
  - Processos diferentes podem ser aplicados em diferentes canais.
  - Apenas um canal pode ser processado.
  - etc.
- Além disso, o efeito obtido irá depender que qual canal foi alterado.

Exemplo

- Realizar uma operação de negativo no canal I de uma imagem HSI irá inverter o brilho percebido, mantendo a matiz e a saturação.

Exemplo

- Uma operação de negativo nos três canais RGB irá gerar um efeito de negativo colorido na imagem.

Imagens coloridas

Exemplo

- É possível aplicar um filtro de suavização em um, dois ou três canais de uma imagem colorida, borrando os detalhes apenas nos canais processados.

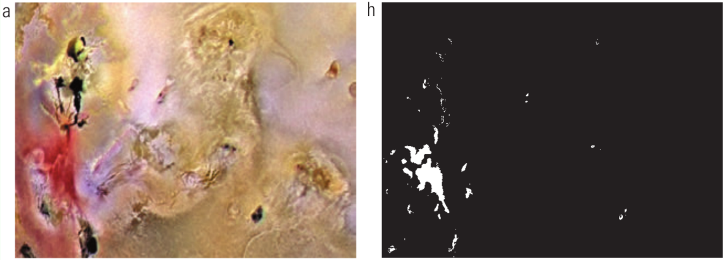
Exemplo

- De maneira análoga, um filtro de aguçamento pode ser aplicado para realçar os detalhes correspondentes a apenas um dos canais de uma imagem RGB.

Imagens coloridas

Exemplo

- Uma transformação de limiarização pode ser aplicada no canal H para isolar apenas os *pixels* de cor avermelhada em uma imagem.



Vídeo

- No contexto de visão computacional, raramente há aplicações em que uma imagem é processada independente do tempo.
- Traduzindo o grego acima: trabalha-se com sinais de vídeo, formados por uma sequência de imagens ao longo do tempo.
- Isto parte da necessidade dos sistemas de visão computacional de detectar, compreender e agir a partir da mudança nas informações contidas no vídeo captado pelas câmeras.

Vídeo

- Em geral, as técnicas de processamento de imagens digitais se aplicam ao vídeo digital, visto que este é meramente uma sequência de quadros ao longo do tempo.
- Por outro lado, há restrições e vantagens que se aplicam exclusivamente nos casos em que se trabalha com vídeo.

Vídeo

- Em uma imagem isolada, os *pixels* podem ser **correlacionados espacialmente**.
- Isto é, em determinadas partes da imagem, como o céu ou o mar representados em uma fotografia, há uma correlação das propriedades de *pixels* próximos.
- Via de regra, imagens naturais e/ou esteticamente agradáveis possuem regiões contínuas de intensidade e cores similares, com transições relativamente suaves.
- Imagens artificiais ou submetidas a ruído podem ter menor grau de correlação espacial entre os *pixels*.
- Esta correlação pode ser explorada para algoritmos de compressão, por exemplo.

Vídeo

- Por outro lado, os *pixels* de uma imagem isolada sempre serão **temporalmente descorrelacionados**.
- Não há um conjunto de *pixels* anterior ou posterior correspondente com o qual os *pixels* da imagem em questão poderiam ser correlacionados.
- Isto não é verdade para um vídeo. Neste caso, é possível falar de **correlação temporal**.

Vídeo

- Na maioria dos vídeos, um conjunto de *pixels* que aparece em um quadro tem uma probabilidade maior de aparecer no próximo.
- Esta propriedade pode ser explorada em aplicações de compressão de vídeo e rastreio de objetos.

Exemplo

- Ao buscar um objeto em um quadro de um vídeo, caso este tenha sido encontrado no quadro anterior a busca pode começar por esta posição antiga, confiando que no quadro atual o objeto estará na mesma posição ou na região próxima.

Vídeo

- Outra diferença extremamente importante entre o processamento de vídeo e o processamento de imagens é a restrição temporal para o processamento em si.
- As imagens sempre são **processadas após a gravação/captura**, isto é, não há um limite de tempo prático para que um algoritmo qualquer seja executado para manipular os *pixels*.
- É evidente que um algoritmo com melhor resultado e mesmo tempo de execução quando comparado a outro será preferível, mas *a priori* não há problema em implementar um algoritmo mais lento com maior qualidade quando comparado a um algoritmo mais rápido com menor qualidade.

Vídeo

- Este cenário pode se repetir para os casos com vídeo, ou seja, o vídeo inteiro é gravado e o processamento ocorre após.
- Porém, na maioria dos sistemas de visão computacional o processamento é feito em **tempo real**.
- Isto gera uma restrição no tempo disponível para a execução dos algoritmos de processamento, visto que as ações do sistema podem ser atrasadas pela espera de um processo.