
Lecture Tema 3: Filtros e características nas imagens, Detecção e correspondência de características, Texturas e Segmentação de imagens

Erica Yurie Saito, João Vitor Verona Biazibetti
Department of Computer Science
Federal University of Technology - Paraná / UTFPR
Campo Mourão, Paraná, Brazil
ericasaito@alunos.utfpr.edu.br
joaaoverona@gmail.com

1 Introdução

Esse trabalho tem como objetivo fazer uma breve apresentação sobre filtros e características nas imagens, detecção e correspondência de características, texturas e segmentação de imagens.

Na seção 2 será visto sobre alguns tipos de filtros e suas funcionalidades e alguns operadores utilizados. Na seção 3, sobre detecção e alguns detectores de características. Na seção 4, sobre texturas e suas aplicações, como identificar uma textura, análise de textura e exemplos de medidas. Na seção 5, uma definição de segmentação, tipos de estratégias de segmentação, limiar e métodos para definir um bom limiar.

2 Filtros e características

Nessa seção será apresentado uma definição sobre filtragem e alguns tipos de filtros.

2.1 Filtragem

Filtragem é um processo de transformações da imagem pixel a pixel, que não dependem apenas do nível de cinza de um determinado pixel, mas também dos níveis de cinza dos pixels vizinhos.

O processo de filtragem é feito utilizando matrizes (máscaras) que são aplicadas sobre a imagem.

Há dois tipos de filtros lineares e não lineares.

2.1.1 Filtros lineares

Os filtros lineares suavizam e realçam detalhes da imagem e minimizam efeitos de ruído, sem alterar a média da imagem.

A seguir é apresentado alguns tipos de filtros lineares.

- Passa-baixa: Suaviza a imagem atenuando as altas frequências, que correspondem às transições abruptas. Tende a minimizar ruídos e apresenta o efeito de borramento da imagem. Quanto maior a máscara, maior o efeito de borramento.
- Passa-alta: - Realça detalhes, produzindo uma "agudização" da imagem, isto é, as transições entre regiões diferentes tornam-se mais nítidas. Realçam certas características presentes na imagem, tais como bordas, linhas curvas ou manchas, mas enfatizam o ruído existente na imagem.
- Realce não-direcional de bordas: Realçam bordas, independente da direção. Podem ser utilizados 3 tipos de máscaras:
 - Máscara alta: Deixa passar menos os baixos níveis de cinza, deixando a imagem mais clara.
 - Máscara baixa: Produz uma imagem mais escura que a Máscara alta.
 - Máscara média: Apresenta resultados intermediários.
- Realce de imagens: Utiliza máscaras apropriadas ao realce de características de imagens obtidas por um sensor específico.

A Figura 1 mostra um exemplo da aplicação do filtro passa-baixa e a Figura 2, um exemplo da aplicação do filtro passa-alta.

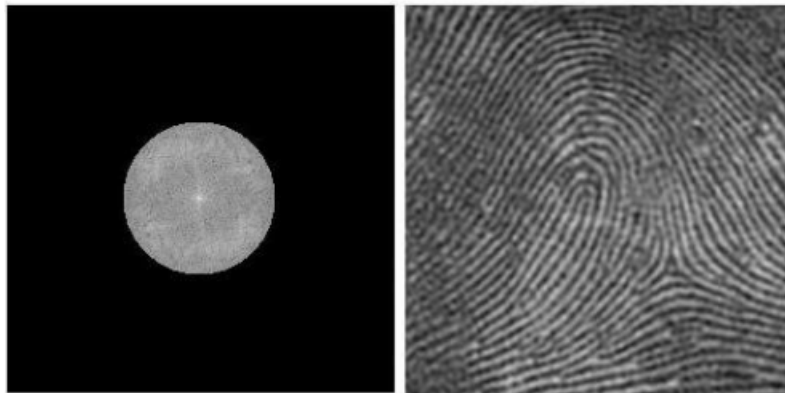


Figure 1: Exemplo da aplicação do filtro passa-baixa.



Figure 2: Exemplo da aplicação do filtro passa-alta.

2.1.2 Filtros não-lineares

Os filtros não-lineares minimizam ruídos e suavizam bordas, alterando a média da imagem. Eles alteram a imagem sem diminuir a sua resolução. Há algumas técnicas como:

- Operadores para detecção de bordas: Detecta características, como bordas, linhas, curvas e manchas. Apresentando alguns exemplo de operadores:

– Operador de Roberts:

- * Denominado pela função:

$$a' = \sqrt{(a - d)^2 + (c - b)^2}$$

- * Onde a' é o nível de cinza correspondente à localização a .

$$\begin{bmatrix} a & b & \dots \\ c & d & \dots \\ \dots & \dots & \dots \end{bmatrix}$$

- * Há o problema que algumas bordas podem ser mais realçadas que as outras dependendo da direção, mesmo possuindo a mesma magnitude.

– Operador de Sobel:

- * Mais sofisticado que o Operador de Roberts.
- * Realça linhas verticais e horizontais mais escuras que o fundo, sem realçar pontos isolados.
- * Utiliza duas máscaras:
 - Máscara (a) = detecta as variações no sentido horizontal.

$$\begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

- Máscara (b) = detecta as variações no sentido vertical.

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

- Em cada pixel, o resultado será dado pela equação:

$$a' = \sqrt{a^2 + b^2}$$

- Onde a' é o nível de cinza corresponde à localização do elemento central da máscara.

● Filtros morfológicos

- Exploram as propriedades geométricas dos sinais (níveis de cinza da imagem).
- As máscaras são chamadas elementos estruturantes e os valores são 0 ou 1.
- Há 3 filtros morfológicos básicos:
 - * Filtro morfológico de mediana:
 - Utilizado para suavização e eliminação de ruídos
 - Mantém a dimensão da imagem.
 - * Filtro morfológico de erosão:
 - Provoca efeitos de erosão das partes claras da imagem (altos níveis de cinza), gerando imagens mais escuras.
 - * Filtro morfológico de dilatação:
 - Provoca efeitos de dilatação das partes escuras da imagem (baixos níveis de cinza), gerando imagens mais claras.

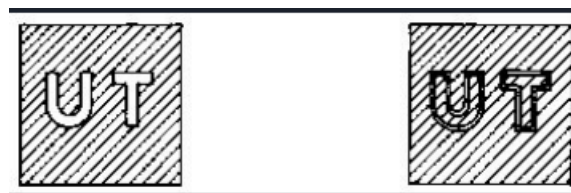


Figure 3: Exemplo da aplicação do operador de Roberts

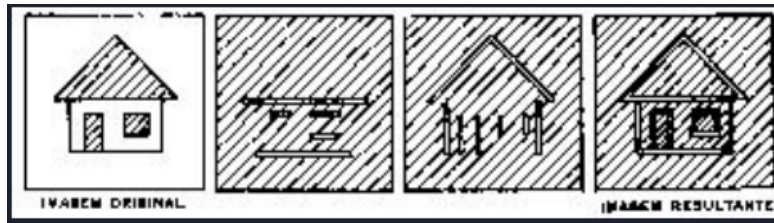


Figure 4: Exemplo da aplicação do operador de Sobel

A figura 3 mostra um exemplo de aplicação do Operador de Roberts e a figura 4 mostra um exemplo do operador de Sobel.

Para explicar os filtros morfológicos, assumamos as seguintes matrizes como o elemento estruturante e a imagem.

Elemento estruturante:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Imagem:

$$\begin{bmatrix} 3 & 6 & 5 \\ 2 & 8 & 3 \\ 2 & 6 & 5 \end{bmatrix}$$

No caso do filtro morfológico da mediana, o pixel central seria alterado para o valor 6 (valor mediano da sequência [2,3,6,6,8]). No filtro morfológico de erosão, o valor central seria o menor valor da ordenação, sendo o valor 2 e no filtro morfológico de dilatação, o maior valor, sendo o valor 8.

3 Detecção e correspondência de características

Nessa seção, será apresentado sobre características, detecção de características e alguns exemplos de detectores de características.

3.1 Características e detecção de características

Características são estruturas específicas em uma imagem. Cantos, arestas, curvas, podem ser exemplos de características.

Já detecção de características são métodos que detectam se para cada ponto de uma imagem existe uma informação de característica previamente abstraída. Há diversos tipos de detecção de características:

- Arestas: Canny, Sobel, Transformada de Hough
- Cantos: Operador de Harris
- Curvas
- Específicos por aplicação: k-curvatura

3.2 Detector de características

Será apresentado alguns exemplos de detector de características.

3.2.1 Método de Canny

O método de Canny é utilizado para a detecção de bordas. Diferente dos outros métodos de detecção de bordas, esse método possui dois refinamentos adicionais, detecta borda com largura de um pixel e fornece bordas fracas em regiões da imagem de menor contraste. O passo a passo do método de Canny é:

1. Suavização da imagem com uma Gaussiana $G\sigma$
2. Calcular o contraste da borda (E_s) é a magnitude do vetor gradiente, que é obtida dos gradientes vertical e horizontal da imagem, como nos outros métodos.
3. Calcular a orientação da borda: O gradiente da imagem é um vetor composto por:

$$\nabla I(x, y) = \frac{\partial I(x, y)}{\partial x}i + \frac{\partial I(x, y)}{\partial y}j$$

A orientação do vetor gradiente em cada pixel é obtida por:

$$\theta(x, y) = \arctan \frac{\partial I(x, y)}{\partial y} / \frac{\partial I(x, y)}{\partial x}$$

4. Eliminar pontos não-maximais da imagem de gradiente, calculando com as 4 direções (0° , 45° , 90° e 135°). Um pixel é contorno se a sua magnitude é máxima na direção do gradiente.
5. Limiarização com histerize:
 - (a) Definir dois limiares, L_{baixo} e L_{alto}
 - (b) . Se um pixel da imagem $I_{N(i,j)} \leq L_{alto}$ então marcar esse pixel como sendo uma borda
 - (c) Checar seus vizinhos na direção perpendicular ao gradiente $\theta(i, j)$. Se eles forem maiores do que L_{baixo} , então marcá-los como sendo bordas

Esse procedimento permite um limiar mais alto para eliminar ruídos ao mesmo tempo que não elimina bordas reais existentes na imagem.

3.2.2 Método de Sobel

O método de Sobel é um operador que calcula diferenças finitas, dando uma aproximação do gradiente da intensidade dos pixels da imagem.

Em cada ponto da imagem, o resultado da aplicação do filtro Sobel devolve o gradiente da intensidade em cada ponto, dando a direção da maior variação de claro para escuro e a quantidade de variação nessa direção, obtendo uma noção de como varia a luminosidade em cada ponto.

Sendo A a imagem inicial então, G_x e G_y serão duas imagens que em cada ponto contém uma aproximação às derivadas horizontal e vertical de A:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * A$$
$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A$$

A magnitude G e a direção do gradiente θ são obtidas por:

$$G = \sqrt{G_x^2 + G_y^2}$$

$$\theta = \arctan\left(\frac{G_y}{G_x}\right)$$

3.2.3 SIFT (*Scale Invariant Feature Transform*)

O SIFT é uma técnica de processamento de imagens que permite a detecção e extração de descritores locais. Ele não varia em mudanças de iluminação, ruídos, rotação, escala. De forma simplificada, as 4 etapas são:

1. Detecção dos máximos locais = *Keypoints*
2. Localização dos *keypoints* = Refinamento
3. Cálculo da orientação dos *keypoints* considerando o gradiente = Invariante à rotação
4. Gera descritor de acordo com orientação do *keypoint*

4 Texturas

Texturas são facilmente perceptíveis pela visão humana, mas pela visão computacional é uma tarefa complexa.

As texturas contém diversas informações. Entre elas há informações sobre a distribuição espacial, variação de luminosidade, suavidade, rugosidade, regularidade e além disso, descreve o arranjo estrutural das superfícies e as relações vizinha.

As texturas podem ser aplicadas em:

- Segmentação ou divisão de uma imagem em regiões
- Descrição de regiões
- Classificação e rotulação de uma região
- Análise de forma
- Réplica para caracterizar superfícies (síntese de imagens)

4.1 Identificação de texturas

Há diversas maneiras de identificar as texturas:

- Entropia de uma imagem: Quantidade de cada um dos a_j tons (histograma) ou a_j texturas.

$$H(Pa) = \sum_{i=1}^J p(a_j) \log p(a_j)$$

- Coeficiente de Hurst: Aproximação da DF(dimensão fractal) para imagens.

$$D = \frac{\ln N}{\ln(1/r)}$$

N = Quantidade de partes que a imagem foi dividida, r = fator de escala

- Coeficientes de variação espacial: Medida de dispersão (coeficiente de variação) do conjunto de pixels pertencentes à região da imagem.

$$CV = \frac{DP}{\bar{x}} * 100$$
$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

\bar{x} =medida de posição (média)

- Outros
 - Dimensão Fractal

- Momentos de Intensidades
- Matrizes de Co-ocorrência
- Descritores de Textura de Haralick
- Funções de Autocorrelação
- LZW
- Matriz de comprimentos corridos
- Spatiograms
- Descritores de Textura baseados nos Histogramas de Soma e Diferenças ...

4.2 Análise de textura

Há dois tipos de análise, a análise estrutural e a análise estatística.

4.2.1 Análise estrutural

A análise é baseada em repetição dos padrões primitivos básicos com uma certa regra de posicionamento. Não são muitos aplicados, pois as texturas mais comuns não são tão regulares para utilizar esse método.

4.2.2 Análise estatística

Há dois tipos:

- Primeira ordem:
 - Define com que frequência um pixel aparece na imagem
 - Depende somente de valores de pixel individuais e não de interação ou co-ocorrência de pixels vizinhos, não levando em consideração a distribuição espacial dos níveis de cinza da imagem.
 - Exemplo: média, variância, desvio padrão, simetria, energia, entropia...
- Segunda ordem:
 - Observação de um par de pixels, em uma distância randômica numa imagem, numa orientação e posição randômica
 - Obtém um resultado mais preciso, pois leva o posicionamento em consideração
 - Exemplo: SGLDM - *Spatial Grey Level Dependence Level*, GLDM - *Grey Level Difference Method*, RLM - *Run Length Method*



Figure 5: Análise estatística de primeira ordem e segunda ordem

A principal diferença entre essas duas análises é que a segunda leva em consideração a posição dos pixels. Isso significa que se a análise de primeira ordem analisasse os dois casos da figura 5, os histogramas dos dois casos seriam iguais, mas a análise de segunda ordem resultaria em diferentes histogramas.

4.3 Exemplo de medidas de análise

- Primeira ordem:

- n-ésimo momento do histograma de uma imagem:

$$\mu_n(z) = \sum_{i=1}^L (z_i - m)^n p(z_i)$$

$$m = \sum_{i=1}^L z_i p(z_i)$$

- Suavidade relativa R da textura

$$R = 1 - \frac{1}{1 + \omega^2(z)}$$

- Obliquidade (*skewness*)

$$V = \frac{\mu_3}{\omega^3}$$

- Curtose (achamento da distribuição)

$$K = \frac{\mu_4}{\omega^4(z)} - 3$$

- Segunda ordem

- Matrizes de co-ocorrência (Matriz de ocorrência simultânea)

- * Faz uma relação entre dois pixels por vez, um chamado de pixel referência e o outro de pixel vizinho
- * Representa a distância e as relações espaciais angulares sobre uma sub-região de uma imagem de tamanho especificado
- * Cada elemento da matriz é uma medida de probabilidade de ocorrência de valores de níveis de cinza separados por uma dada distância numa dada direção
- * A matriz de co-ocorrência de uma imagem I quantizada em N níveis de cinzas para dado um ângulo e uma distância é representada por uma matriz M onde M(p, q) indica a quantidade de ocorrências onde um pixel de intensidade q é vizinho de um pixel de intensidade p a uma distância e formando um ângulo X com o mesmo

p \ q	1	2	3	4	5	6
1	1	0	1	0	0	0
2	1	1	1	0	0	0
3	0	2	1	2	0	0
4	0	0	0	1	2	1
5	0	0	1	1	1	2
6	0	0	0	0	1	2

Figure 6: Exemplo de Matriz de ocorrência

- Matrizes Run-Length

- * Codifica a textura levando em consideração a quantidade de vezes que um mesmo nível de cinza i aparece em uma sequência j a uma determinada direção
- * Produz medidas baseado no número de sequências de tons de cinza para um certo comprimento (*Run length*)
- * A sequência forma conjunto de pixels consecutivos e co-lineares com a mesma intensidade

i \ j	1	2	3	4	5	6
1	1	8	0	0	0	0
2	2	4	1	0	0	0
3	3	4	1	0	0	0

Figure 7: Exemplo de Matriz Run Length

5 Segmentação de imagens

A segmentação de uma Imagem I é uma partição em regiões homogêneas R_1, R_2, \dots, R_n tal que:

1. Todo pixel pertence a uma região $I = \bigcup_{i=1}^n R_i$
2. R_i é uma região conexa para todo $i = 1, 2, \dots, n$
3. Nenhum pixel pertence a mais de uma região
 $R_i \cap R_j = \emptyset, \forall i \neq j$

5.1 Estratégias de segmentação

Há dois tipos de estratégias:

- Descontinuidade dos níveis de cinza
 - A partição da imagem é feita com as alterações bruscas de intensidade
 - Exemplo: Detecção de linhas, bordas, pontos isolados
- Similaridade dos níveis de cinza
 - A partição da imagem é feita através da similaridade dos pixels, seguindo um critério
 - Exemplo: Binarização, crescimento de regiões, divisão, junção de regiões

5.1.1 Detecção de descontinuidades

Procura por pontos isolados.

- Detecção de linhas: Feita através do uso de um filtro passa altas direcionais
- Detecção de contornos
 - Mais utilizado entre os métodos de detecção de descontinuidades
 - Geralmente efetuada a partir do cálculo da primeira e da segunda derivada da imagem
 - Bordas = normalmente são borradas
 - Existência de imperfeições no processo de aquisição da imagem podem causar “rampas” no contorno da imagem
 - A magnitude da primeira derivada e as passagens por zero da segunda derivada podem ser utilizadas para detectar os contornos

5.1.2 Detecção de similaridade

Procura por pontos semelhantes.

- Limiarização

- Consiste em separar regiões de uma imagem quando esta apresenta fundo e objeto (*background, foreground*)
- Extrai objetos selecionando um limiar T que separa o fundo e o objeto
- Há 3 tipos de limiares:
 - * Global:
 - Para cada ponto (s, y) tal que $f(s, y) > T$ denominado ponto do objeto, caso contrário fundo
 - * Adaptativo:
 - Seleciona um limiar individual para cada pixel baseado no alcance da intensidade estimado em sua vizinhança local
 - Melhor limiarização quando não existem cumes bem definidos
 - Passo a passo:
 1. Dividir a imagem original em sub-imagens
 2. Determinar um limiar independentemente para cada região
 3. Cada imagem R_i é então processada usando um limiar local
 4. Uma nova imagem R' é definida por $R' = U * R_i$
 - * Local:
 - Semelhante ao global, mas utiliza propriedades locais
 - Além das intensidades, utiliza-se uma propriedade local
 - $T(x, y) = T(l(x, y), P(x, y))$

$$g(x, y) = \begin{cases} 1, & \text{se } f(x, y) > L(\text{forma}) \\ 0, & \text{se } f(x, y) \leq L(\text{fundo}) \end{cases}$$

Um bom limiar é aquele que causa um erro mínimo.
Ele pode ser achado pelos seguintes métodos:

- Valor obtido por testes
- Valor médio dos tons de cinza
- Valor mediano entre o tom máximo e o tom mínimo
- Algoritmos automáticos (Otsu, Kittler...)

5.1.3 Algoritmos de limiarização de Otsu

Permite a seleção automática do valor do limiar T. Passo a passo do algoritmo:

1. Normalizar o histograma de acordo com:

$$p_r(r_k) = \frac{n_k}{n}$$

Onde:

- $0 \leq r_k \leq 1$ e $k = 0, 1, 2, \dots, L - 1$
 - L = número de níveis de cinza da imagem
 - n = número total na imagem
 - $p_r(r_k)$ = percentual do k-ésimo nível de cinza
 - n_k = número de pixels cujo nível de cinza corresponde k
2. Para um determinado limiar T, têm-se dois grupos de pixels
 - Grupo C0 formado pelos valores 0, 1, 2, ..., T-1
 - Grupo C1 formado pelos valores T, T+1, ..., L

3. Encontrar a variância

$$\sigma_B^2 = w_0(\mu_0 - \mu_T)^2 + w_1(\mu_1 - \mu_T)^2$$

Onde:

- w_0 = probabilidade de c0
- w_1 = probabilidade de c1
- μ_0 = média de c0
- μ_1 = média de c1
- μ_T = média total
- k = nível de cinza

4. Aplicar a equação da variância para todos possíveis limiares e escolher o de maior valor

6 References

- Material da Divisão de Processamento de Imagens (DPI), Teoria: Processamento de Imagens, disponível em: <http://www.dpi.inpe.br/spring/teoria/filtrage/filtragem.htm>
- Material do Professor Eduardo L.L. Cabral, disponível em: https://edisciplinas.usp.br/pluginfile.php/4112482/mod_resource/content/0/V10%20-Deteccao%20de%20bordas.pdf
- Material do Departamento de Informática da PUC-RJ, Trabalho de Imagem, disponível em: <http://www.inf.puc-rio.br/~asouza/FCG/imagem.html>
- Material do Professor Kamel Bensebaa, disponível em: <http://slideplayer.com.br/slide/10104678/>