



PROCESSAMENTO DE NUVEM DE PONTOS 3D
CGEO- 7028

SfM/SIFT

UFPR – Departamento de Geomática
Prof. Jorge Centeno
2023
copyright@ centenet

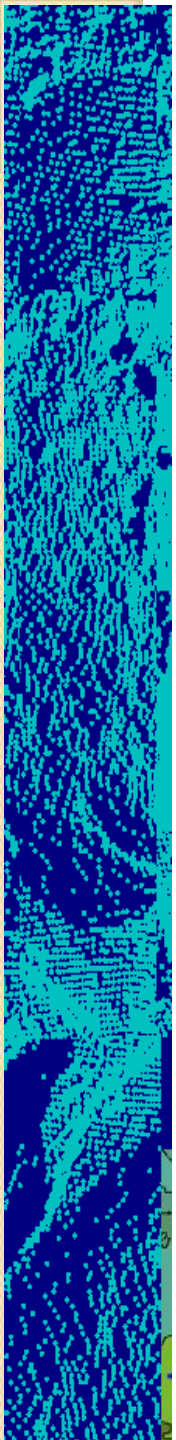
Problema

Dado um par de imagens (fotografias), detectar pares de pontos homólogos de forma automática.

Porém, podem existir diferenças de contraste, rotação, translação e escala.

Com estes pontos, e usando técnicas de fotogrametria, é possível orientar as fotos e depois calcular coordenadas 3D de outros pontos na imagem.





- Inicialmente, vamos revisar conceitos de filtragem espacial
- A filtragem é o resultado da convolução de uma máscara (kernel/ Moving window) e a imagem.

Filtro passa-baixas (suavização)

Atenua as altas frequências, aquelas associadas a detalhes na imagem, e deixa apenas as baixas frequências. O efeito deste filtro é a remoção de detalhes da imagem e sua suavização.

Este efeito é atingido substituindo o pixel central pela média da janela. A média pode ser uma média simples ou uma média ponderada, onde diferentes pesos são atribuídos aos vizinhos em função de sua proximidade ao pixel central.

Exemplo:

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Filtro

1	1	1
1	1	1
1	1	1

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Passa baixas



• Original

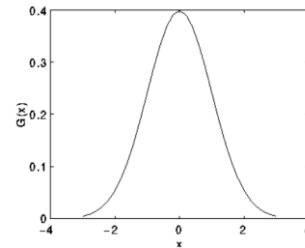


passa-baixas (suavização)

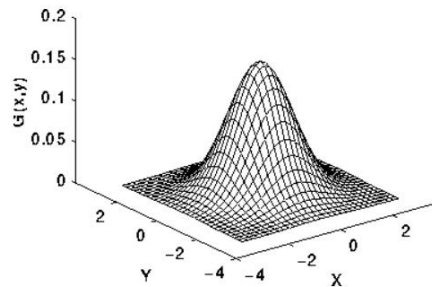
FILTRO GAUSSIANO

O Filtro Gaussiano é um tipo de filtro passa-baixas que usa uma função Gaussiana para calcular os pesos do filtro e, conseqüentemente, a transformação linear. Assim, maior peso é dado ao central e o peso diminui com a distância ao pixel central da janela.

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$



$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

Gaussiano

$$\begin{aligned} &= \sum_i \sum_j f(x-i, y-j) \exp \left\{ \frac{-(i^2 + j^2)}{2\sigma^2} \right\} \\ &= \sum_i \left[\sum_j f(x-i, y-j) \exp \left\{ \frac{-j^2}{2\sigma^2} \right\} \right] \exp \left\{ \frac{-i^2}{2\sigma^2} \right\} \\ &= [f(x, y) * G(y)] * G^T(x). \end{aligned}$$

Porém, um filtro Gaussiano 2D pode ser substituído por dois filtros Gaussianos 1D, que são mais rápidos de calcular.

Exemplo de Gaussiano

$$f * \left(\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \right) = \left(f * \frac{1}{4} [1 \ 2 \ 1] \right) * \left(\frac{1}{4} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \right)$$

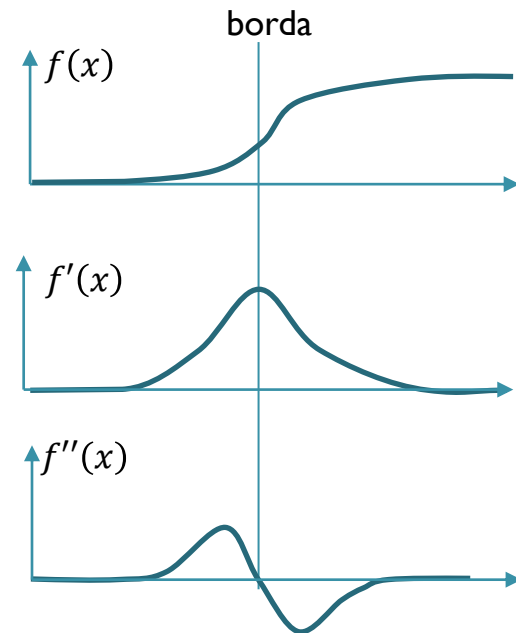
$$\frac{1}{4} [1 \ 2 \ 1] * \frac{1}{4} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}.$$

Laplaciano

- Um filtro Laplaciano calcula a segunda derivada de uma imagem.

$$\nabla^2 f(x, y) = \frac{\delta^2 f}{\delta x^2} + \frac{\delta^2 f}{\delta y^2}$$

Ele é usado como detector de bordas, pois mede a variação da primeira derivada da imagem. Esta variação é grande em locais onde a variação dos valores digitais de pixels vizinhos é grande (bordas).



Laplaciano

- Os filtros Laplacianos geralmente contêm valores negativos na vizinhança e um valor igual à soma destes valores negativos no centro, mas com sinal invertido.
- Na figura ao lado, os pixels acima, abaixo, à direita e à esquerda são negativos e somam -4, no centro corresponde um valor igual a 4 (4 diferenças).
- Outra opção é calcular a diferença entre o central e seus oito vizinhos.

0	-1	0
-1	4	-1
0	-1	0

0	-1	0
0	1	0
0	0	0

0	0	0
-1	1	0
0	0	0

0	0	0
0	1	-1
0	0	0

0	0	0
0	1	0
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

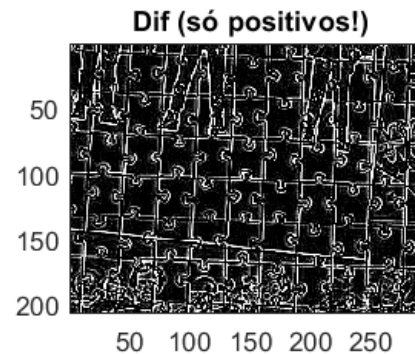
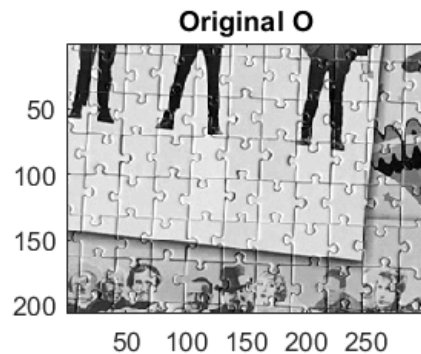
Diferença entre o central e seus vizinhos

```
-1 0 0 -1 0 0 -  
0 0 0 1  
0 1 0 1 0 1  
0 0 0  
0 0 0 0 0 0  
0 0 0 0  
-1 1 0 0 0 0 1 -  
0 1  
0 0 0 0 0  
0 0 0  
0 1 0 1 0 1  
0 0  
-1 0  
0
```



```
-1 -1 -1  
-1 8 -1  
-1 -1 -1
```

A diferença entre o central e seus oito vizinhos



Laplaciano na prática

$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

Na prática, o Laplaciano pode ser muito demorado para calcular e é sensível à presença de ruído. Por isso, não se usa diretamente sua formulação original.

Usa-se a diferença entre a imagem original e a imagem suavizada com um filtro Gaussiano.

Isto é conhecido como o Laplaciano do Gaussiano

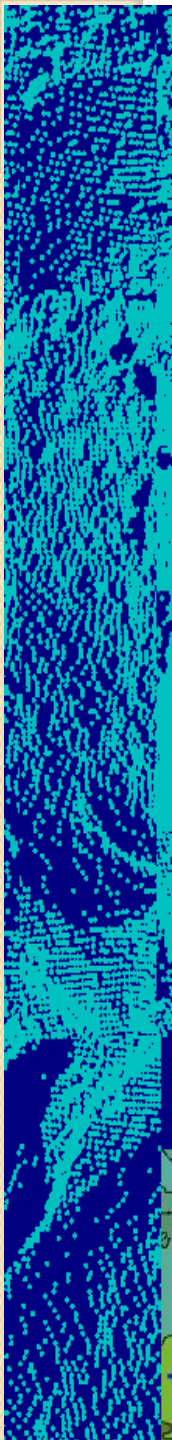
LoG

Filtros direcionais

- A convolução de uma janela e a imagem também é útil para salientar determinadas linhas ou bordas. Por exemplo, as técnicas de filtragem permitem salientar as bordas ou linhas que ocorrem numa determinada direção, fazendo a diferença dos valores na janela considerando sua posição em relação ao pixel central da janela. A seguir são mostrados alguns exemplos destes filtros.



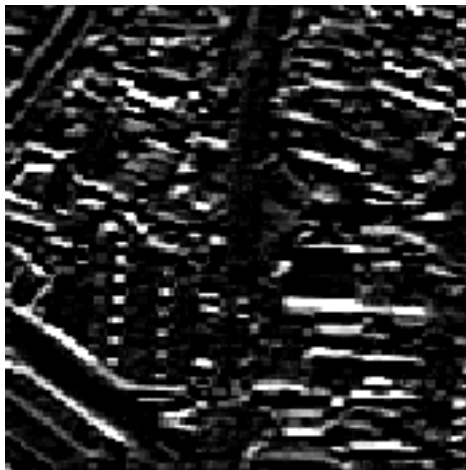
Exemplo: bordas horizontais



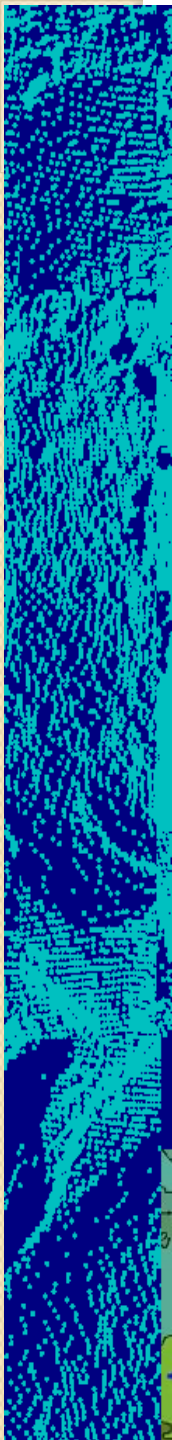
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$$

Norte

$$\begin{bmatrix} 1 \\ -1 \end{bmatrix}$$



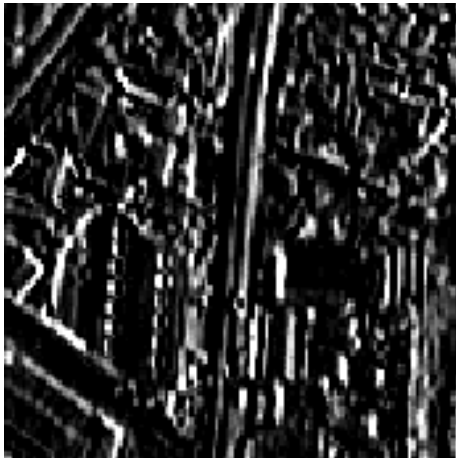
- Os contrastes na direção norte são salientados.
- Algumas linhas diagonais também são salientadas, pois possuem uma componente norte forte.



-1	1	1
-1	-2	1
-1	1	1

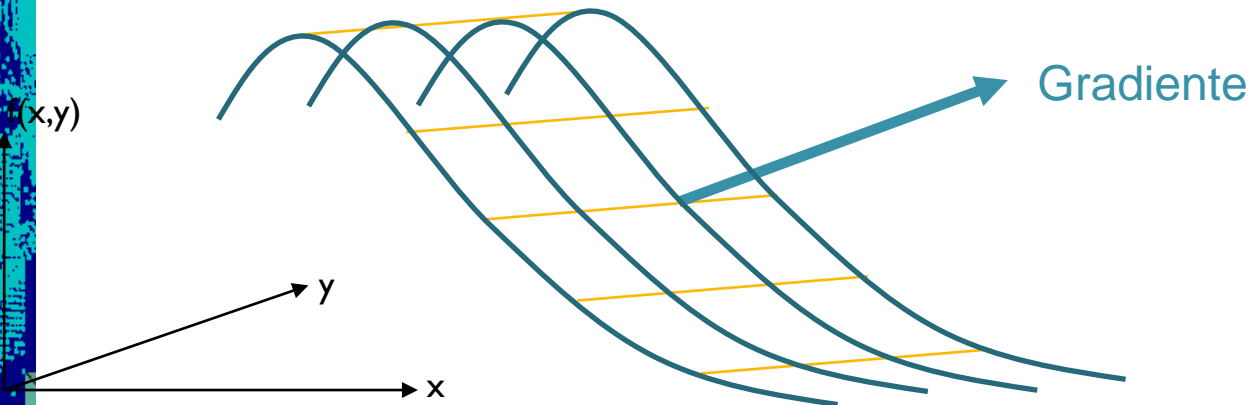
leste

-1	1
----	---



Filtros de Gradiente

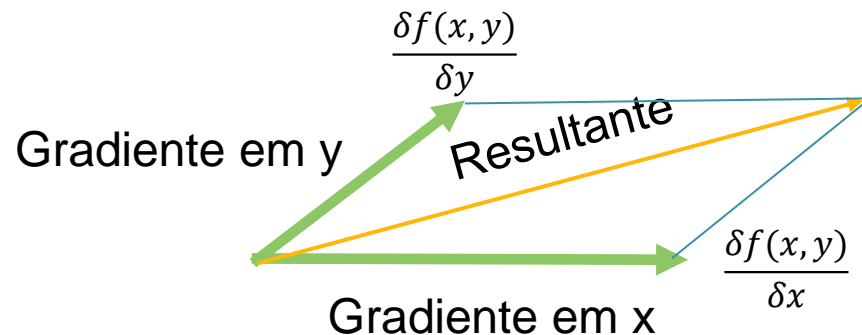
- O gradiente de uma função bidimensional descreve sua inclinação em relação ao sistema adotado.
- O gradiente é tradicionalmente representado perpendicular à superfície.



Filtros de Gradiente

O gradiente pode ser calculado a partir de suas duas componentes (Norte e Leste), ou seja, a derivada parcial da função da superfície em relação a cada uma das componentes.

$$\nabla(x, y) = \left[\frac{\delta f(x, y)}{\delta x}, \frac{\delta f(x, y)}{\delta y} \right]$$



Estimativa do Gradiente

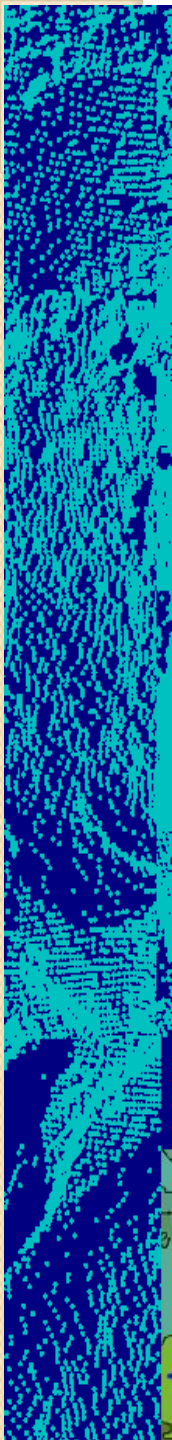
analisando a variação dos valores em sua vizinhança é possível calcular o gradiente para qualquer pixel, .

Para isto:

- Estima-se o gradiente em X
- Estima-se o gradiente em Y (Y perpendicular a X)
- Calcula-se a resultante da soma destes dois vetores.
- O pixel recebe um valor proporcional à magnitude do gradiente.

Armazena-se, na posição do pixel, o valor da Magnitude do gradiente

$$G(x,y) = \sqrt{\frac{\delta f(x,y)^2}{\delta x} + \frac{\delta f(x,y)^2}{\delta y}}$$



A diferença entre os filtros de gradiente radica na maneira de estimar as duas derivadas parciais.

Ex: filtros de Roberts:

$$\begin{array}{|c|c|} \hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array} \frac{\delta f(x,y)}{\delta x}$$

$$\begin{array}{|c|c|} \hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array} \frac{\delta f(x,y)}{\delta y}$$

$$\begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

$$\frac{\delta f(x,y)}{\delta x}$$

$$\frac{\delta f(x,y)}{\delta y}$$

Ex: filtros de Prewitt

Filtros de Sobel

- Sobel

-1	0	1
-2	0	2
-1	0	1

$$\frac{\delta f(x, y)}{\delta x}$$

-1	-2	-1
0	0	0
1	2	1

$$\frac{\delta f(x, y)}{\delta y}$$

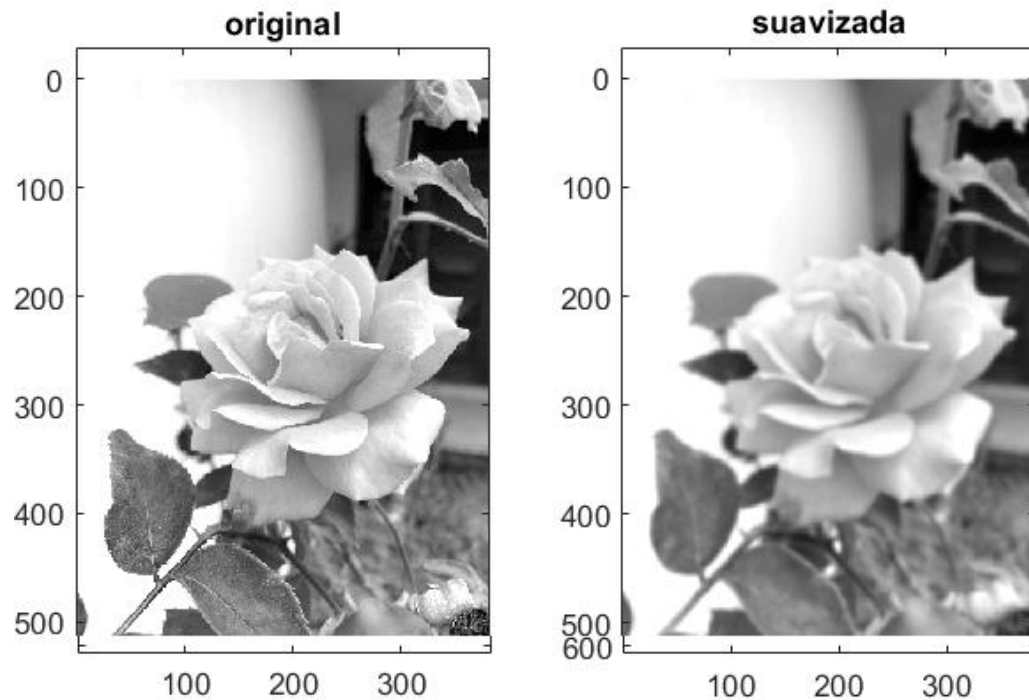




Detectar quinas por Harris

Redução de ruído:

Como o cálculo do gradiente é altamente sensível à presença de ruído, em uma primeira fase a imagem é suavizada, usando um filtro Gaussiano.



Localização de esquinas (HARRIS)

Para detectar uma esquina, Harris propõe analisar a auto-similaridade entre uma região e as regiões em volta.

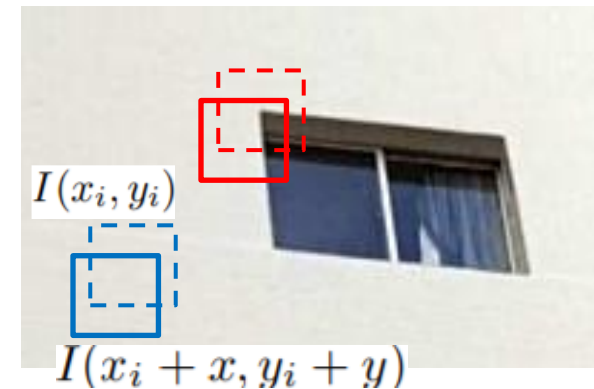
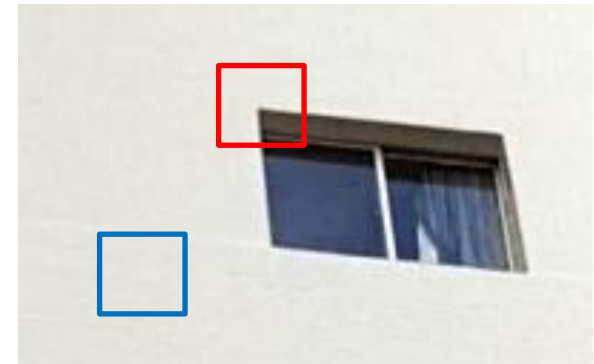
Por exemplo, analisando a diferença entre a região em questão e regiões próximas, resultantes do deslocamento da região.

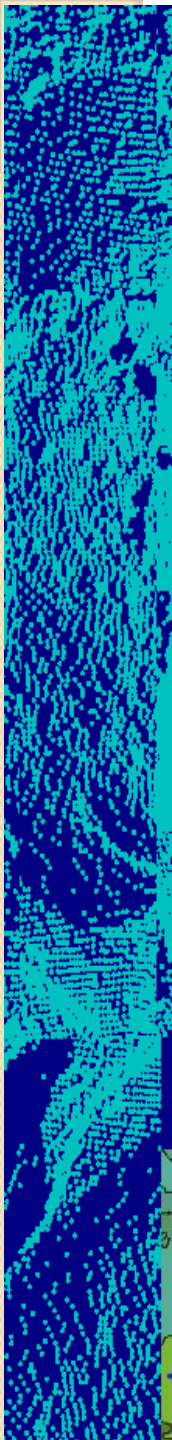
$$I(x_i, y_i)$$

$$I(x_i + x, y_i + y)$$

Se a diferença é pequena, trata-se de uma região uniforme.

$$[I(x_i + x, y_i + y) - I(x_i, y_i)]^2$$





Por isso, Harris propõe calcular a soma das diferenças ao quadrado, ponderando cada elemento (w)

$$S_w(x, y) = \sum_{x_i, y_i} w(x_i, y_i) [I(x_i + x, y_i + y) - I(x_i, y_i)]^2$$

A diferença pode ser aproximada pela série de Taylor como:

$$\begin{aligned} I(x_i + x, y_i + y) &= I(x_i, y_i) + \frac{\partial I}{\partial x_i} x + \frac{\partial I}{\partial y_i} y \\ &= I(x_i, y_i) + [G_x \quad G_y] \begin{bmatrix} x \\ y \end{bmatrix} \end{aligned}$$

onde G_x e G_y são as componentes do Gradiente nas direções x e y . Substituindo em “S”, devemos elevar ao quadrado.

$$S_w(x, y) = [x \quad y] H \begin{bmatrix} x \\ y \end{bmatrix}$$



A expressão quadrática pode ser escrita como:

A matriz H é quadrada 2×2 e simétrica.

Seus elementos são calculados a partir das componentes do Gradiente nas direções x e y .

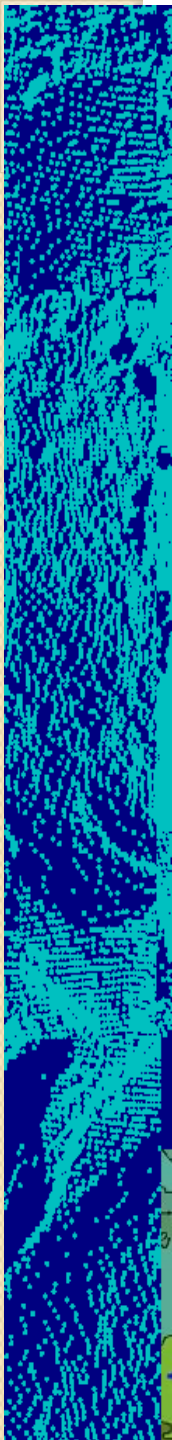
$$A = \sum_{x_i y_i} w(x_i, y_i) I_x^2(x_i, y_i)$$

$$B = \sum_{x_i y_i} w(x_i, y_i) I_{xy}(x_i, y_i)$$

$$C = \sum_{x_i y_i} w(x_i, y_i) I_y^2(x_i, y_i)$$

$$S_W(x, y) = [xy]H \begin{bmatrix} x \\ y \end{bmatrix}$$

$$H(x, y) = \begin{bmatrix} A & B \\ B & C \end{bmatrix}$$



A intenção é determinar os locais onde a diferença é máxima. Para isto se calcula um “score” da diferença, como:

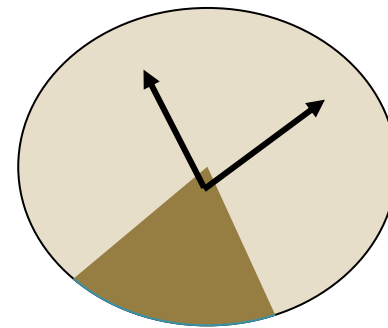
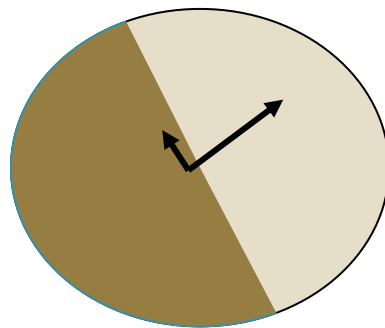
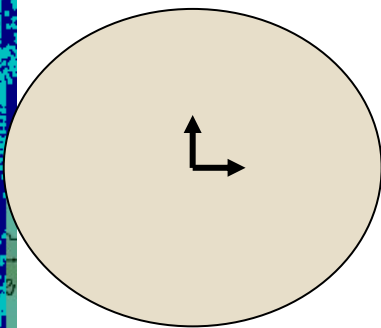
$$R = \det(H) - k(\text{trace}(H))^2$$

E as componentes desta equação se calculam a partir dos autovalores da matriz H .

$$\det(H) = \lambda_1 \lambda_2$$

$$\text{trace}(H) = \lambda_1 + \lambda_2$$

- Se os dois autovalores são pequenos, R é pequeno, logo não se trata de uma borda e sim de uma região uniforme.
- Se o primeiro autovalor for muito grande ($\lambda_1 \gg \lambda_2$) caracteriza-se a existencia de uma direção preponderante, logo trata-se de uma borda (linha)
- Se λ_1 e λ_2 são grandes e $\lambda_1 \sim \lambda_2$, R também é grande, e isto ocorre em quinas.





RESUMO

1. Aplicar um filtro de suavização (Gaussiano) para reduzir ruído.
2. Aplicar o operador de Sobel para estimar o gradiente em cada pixel nas direções x e y
3. Com esses valores, calcular o valor do índice “R” para cada pixel da imagem.
4. Aplicar um limiar para detectar os maiores valores que sejam máximos locais.

Introdução a SIFT

É possível detectar feições (ex: quinas) e uma imagem e identificar a mesma feição em outra.
Problemas: diferenças de rotação, escala e cor.



SIFT Scale-Invariant Feature Transform

Detectores de quinas (como Harris), são invariantes à rotação, mas não em relação à escala.

Para contornar este problema Lowe (2004), criou um algoritmo, Scale Invariant Feature Transform (SIFT)

"Distinctive Image Features from Scale-Invariant Keypoints",

- O algoritmo trata de
 - a) Detectar pontos significativos (quinas)
 - b) Calcular descritores de contraste para caracterizar cada ponto
 - c) Encontrar pares de pontos baseado nos descritores.

https://docs.opencv.org/3.2.0/da/df5/tutorial_py_sift_intro.html



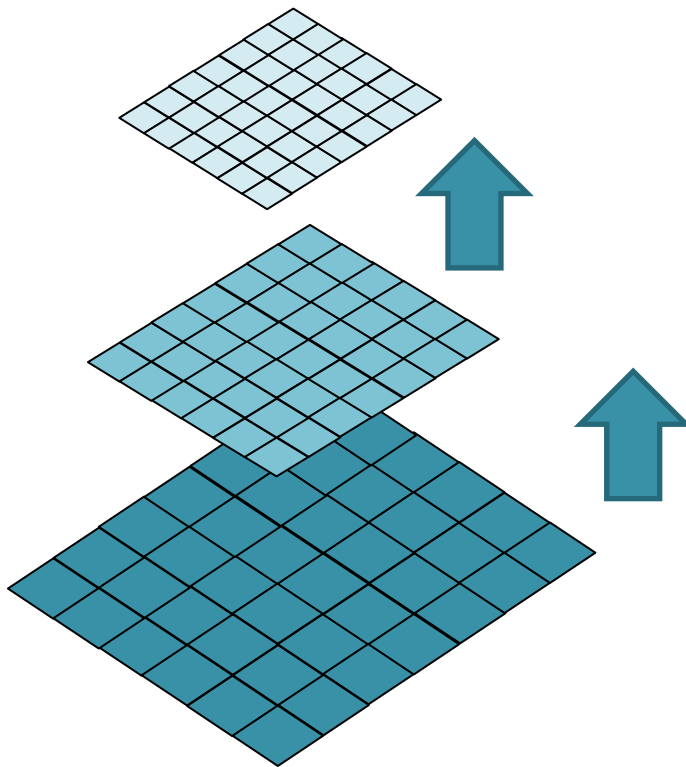
etapas

1. Detectar bordas (aplicar detector de bordas)
2. Selecionar bordas fortes
3. Buscar quinas, não bordas
4. Descrever essas quinas em função de sua vizinhança.
5. Classificar pares de pontos por semelhança.

Scale-space Extrema Detection

Não é possível usar a mesma janela móvel para detectar quinas em diferentes escalas.

Por isso, Lowe propõe buscar quinas aplicando a filtragem variando a resolução da imagem



Variando a escala, a mesma janela pode ser aplicada a cada amostra (oitavas)

Problema da escala

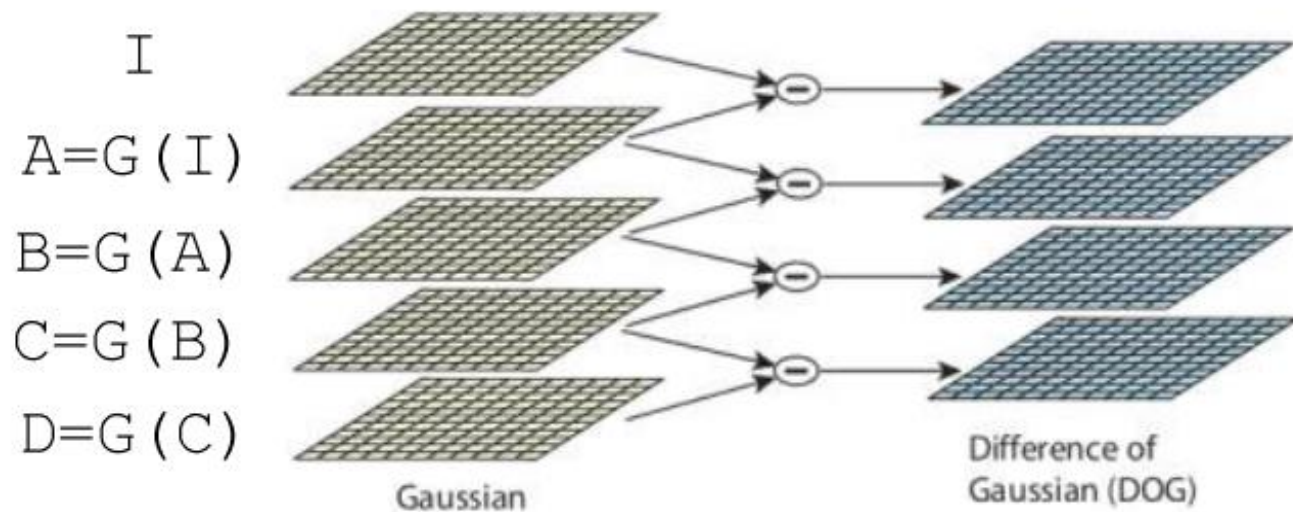
Em cada amostra (nível de amostragem) o filtro Laplaciano de Gaussiano (LoG) é calculado para diferentes valores de σ .

- LoG atua como um detector de pontos que detecta estes detalhes em vários tamanhos devido à mudança de σ . Ou seja, σ atua como um parâmetro de escala.
- Aplicando o filtro a representações da imagem em diferentes resoluções, é obtida uma lista de prováveis posições de quinas (detalhes) na imagem. Cada ponto é um potencial ponto de interesse (keypoint)

DoG

$$FL = I - FG$$

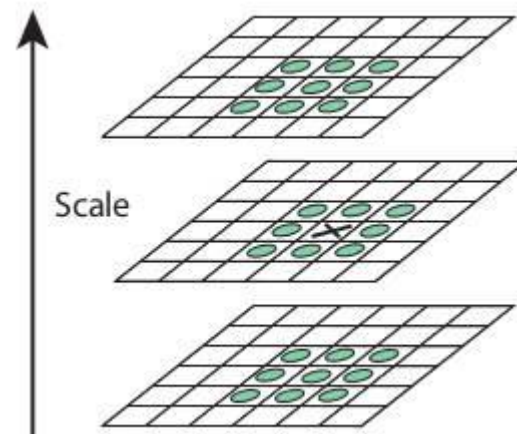
- Como o cálculo do LoG é caro, então usa-se a Diferença de Gaussianas, uma aproximação do LoG.
- A diferença de Gaussianas é obtida como a diferença de suavização Gaussiana de uma imagem com dois σ diferentes. Este processo é feito para diferentes oitavas da imagem na Pirâmide Gaussiana.



Detecção de extremos (máximos)

Uma vez encontrados pontos candidatos, é verificado se esta hipótese se confirma na região, tanto na resolução em questão, como nas outras resoluções.

Um pixel em uma imagem é comparado com seus 8 vizinhos, para verificar se é um máximo local, e os 9 pixels na próxima escala e 9 pixels nas escalas anteriores. Se for um local de máximo, é um potencia ponto de interesse (keypoint). Ainda, conclui-se que este ponto é melhor representado nessa escala.



refinamento

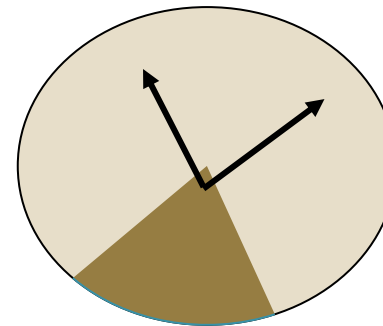
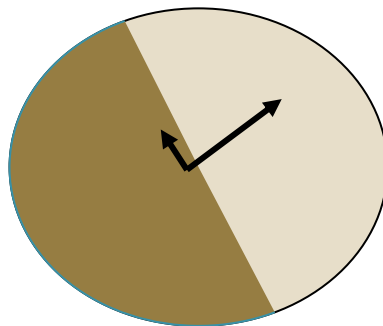
A posição dos pontos detectados é refinada para obter resultados mais acurados.

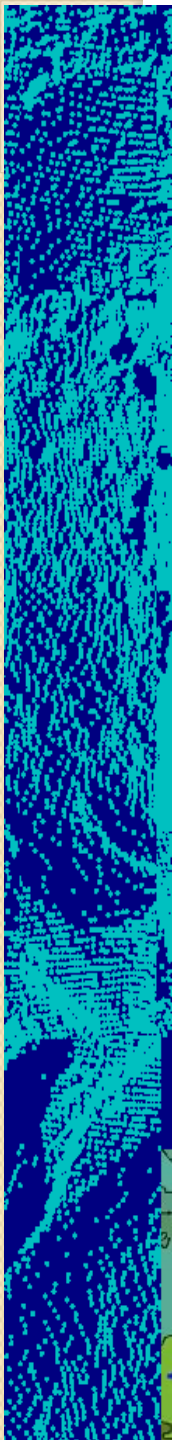
Para isto usa-se uma série de Taylor para interpolar uma superfície com os valores da vizinhança e assim calcular a posição exata do máximo.

Ainda, se a intensidade do pico ficar abaixo de um limiar, o ponto é descartado, por ser considerado “fraco”

O DoG detecta pontos e bordas. As bordas precisam ser removidas, pois não fornecem localização precisa. Para isso, usa-se o princípio do detector de Harris, analisando os autovalores. Como nas bordas o primeiro autovalor é muito maior que o segundo, então as bordas podem ser eliminadas.

Assim, elimina-se pontos-chave de baixo contraste e localizados em bordas, restando pontos de interesse fortes.



- 
- Até aqui, foram localizados os pontos significativos (key points) nas duas imagens.
 - Agora devemos procurar os pares homólogos. Para isto:
 1. Calcula-se a orientação da região
 2. Calcula-se descritores espaciais do ponto
 3. Classifica-se pares por semelhança
 - ...

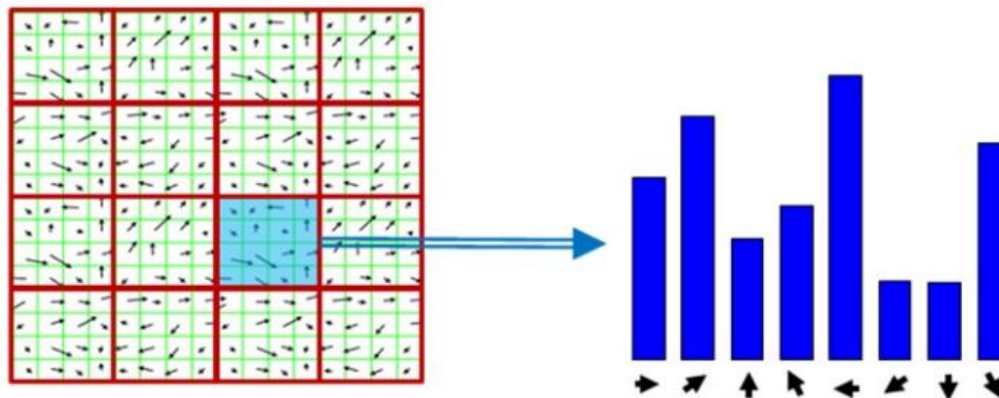
Atribuição de Orientação

Uma vez detectado o local da quina, analisa-se a variação do Gradiente nas imediações do ponto para estimar a orientação predominante na região.

Uma vizinhança de $14 \times 6 \times 16$ pixels em torno do ponto é analisada. A vizinhança é dividida em 4 sub- vizinhanças, cada uma com 16 pixels.

Em cada sub- vizinhança, o gradiente dos 16 pixels é calculada e o histograma das direções do gradiente computado. Para isto, as direções são arredondadas (ex 36 direções)

O pico mais alto do histograma, junto com outros picos com valores maiores que 80% do pico são usados para calcular a direção predominante.



Descritores

Uma vez encontrada a direção predominante, a região pode ser rotacionada em relação a esta direção, suprimindo assim o efeito da rotação.

Os vetores do gradiente são então rotacionados em relação à direção predominante.

A seguir, as direções são simplificadas em 8 valores e todos estes valores armazenados no vetor de características.

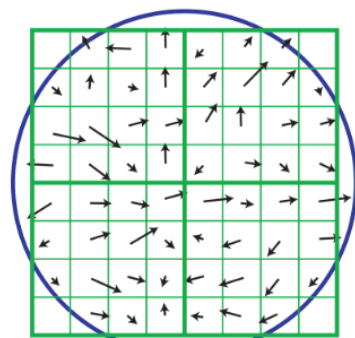
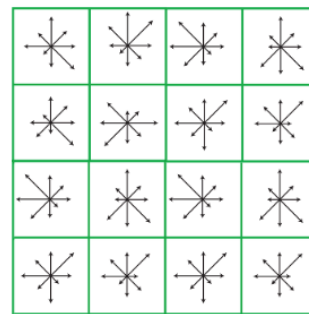


Image gradients



Keypoint descriptor

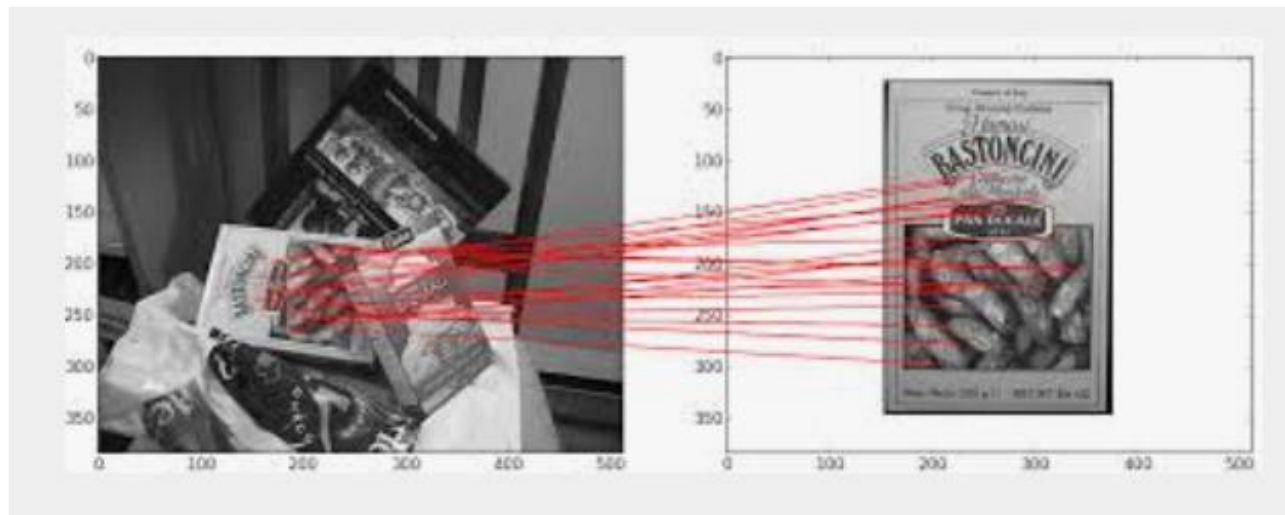
$$a = \begin{bmatrix} a1 \\ a2 \\ a3 \\ a4 \\ a5 \\ a6 \\ \dots \\ a16 \end{bmatrix}$$

$$v = \begin{bmatrix} a1 \\ a2 \\ \dots \\ b1 \\ b2 \\ \dots \\ c1 \\ \dots \\ c16 \end{bmatrix}$$

Matching/Correspondência

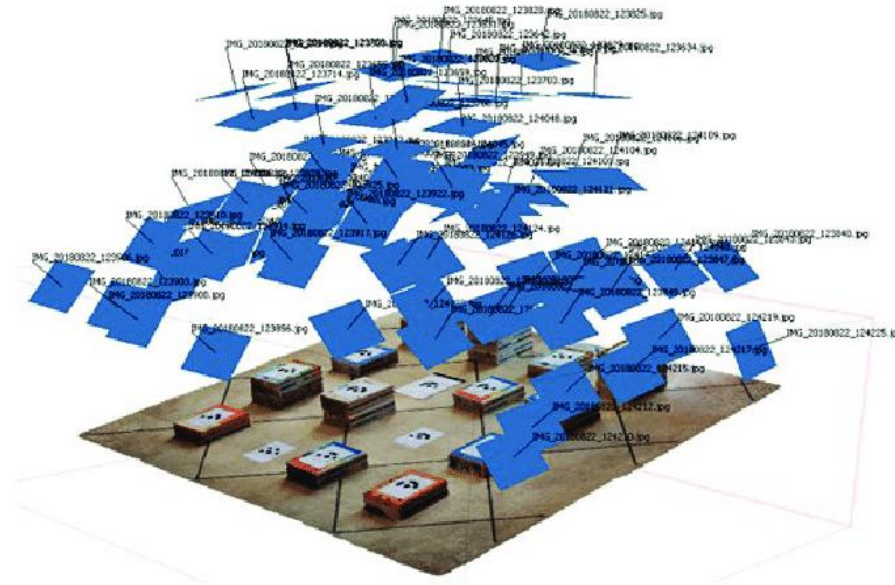
Como cada ponto possui um vetor descritor que não sofre o efeito da escala nem da rotação, então, o seguinte passo é identificar na outra imagem um ponto com o vetor descritor mais parecido. Isto é feito classificando os pares pelo critério de distância mínima Euclidiana.

Em algumas situações, podem ocorrer dois potenciais candidatos com distâncias muito parecidas. Para identificar o melhor par, seleciona-se as situações onde a segunda maior distância é relativamente menor (abaixo de 80% da maior).



SfM

- Uma vez detectados os pares homólogos, calcula-se a orientação exterior de cada imagem por bundle-block adjustment.



SfM

Com a posição e orientação das imagens pode-se calcular as coordenadas 3D de qualquer outro ponto.

Para isto, novos pares de pontos são detectados e a interseção dos pontos no espaço 3D é calculada (densificação)

- Veja suas aulas de fotogrametria...

