(7) Vision For Robotics

① Image Filtering : Accept/Reject certain frequency components in the frequency domain.

1. Filtering can be done in both frequency/spatial domains.
   Spatial domain : Filter $\Leftrightarrow$ mask/kernel

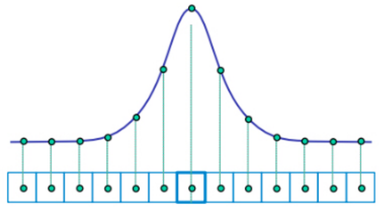
2. Spacial Filters: S_{xy} pixel surrounding point (x,y) in image.
   It generates a new value for corresponding pixel in output image J. eg. Average Filter $J(x,y) = \dfrac{\sum_{(r,c)\in S_{x,y}} I(r,c)}{(2M+1)(2N+1)}$

3. Simple Frequency Filter
   1. Low Pass : Reduce Noise ; Blurs resultant image
   2. High Pass : Edge Detection

4. Linear, Shift-Invariant Filters.
   1. Linear : each pixel is a linear combination of its neighbor
   2. Shift-Invariant : Same operation is performed on every point on the image
      1. Useful Operations: Correlation ; Convolution

Correlation : $J(x) = F \circ I(x) = \sum_{i=-N}^{N} F(i)\, I(x+i)$

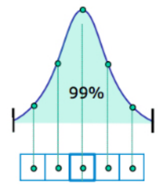
Smoothing filter : $F(i) = \begin{cases} 0.33, & i \in [-1,1] \\ 0, & i \notin [-1,1] \end{cases}$



Normalize filter so that values always add up to 1

$\mu = 0$

σ : controls the amount of smoothing



5. Filters Using Continous Functions

1. Use a Gaussian

$$G(x) = \frac{1}{6\sqrt{2\pi}} \cdot e^{-\frac{(x-\mu)^2}{2 6^2}}$$

2. Closer Pixel have larger influence than farther ones

3. Sigma controls the amount of smoothing

6. Derivatives With Correlation

1. Derivative of an image 对图像相关性求导,量化强度变化的

2. Quantify how quickly intensity changes 快慢

3. Approximate derivative operator

7. Template Matching

1. Find location in image similar to a template

2. SSD : Sum of Squared Differences

$$\sum_{i=-N}^{N} (F(i) - I(x+i))^2$$

2D Plane,

1	2	3
4	5	6
7	8	9

Σ □

1D Line Σ □

Find Minum One

$$= \sum_{i=-N}^{N} (F^2(i) + I^2(x+i) - 2F(i)I(x+i))$$

$$= \underbrace{\sum_{i=-N}^{N} (F^2(i))}_{1} + \underbrace{\sum_{i=-N}^{N} (I^2(x+i))}_{2} - 2\underbrace{\sum_{i=-N}^{N} (F(i)I(x+i))}_{3}$$

1. Same for all pixels

2. Overlap the filter

3. -2 times of Correlation ⇒ Find Value Low

High Correlation = good template match

SSD is better than Correlation

8. Normalized Cross Correlation

1. Correlation: Affected by magnitude of intensities 受强度影响

2. Solution: Normalize 归一化

$$\frac{\sum_{i=-N}^{N}(F(i)I(x+i))}{\sqrt{\sum_{i=-N}^{N}(I(x+i))^2}\cdot\sqrt{\sum_{i=-N}^{N}(F(i))^2}}$$

9. Convolution

Correlation $J(x)=F\circ I(x)=\sum_{i=-N}^{N}F(i)I(x+i)$

Convolution $J(x)=F*I(x)=\sum_{i=-N}^{N}F(i)I(x-i)$

$$J(x,y)=F*I(x,y)=\sum_{j=-M}^{M}\sum_{i=-N}^{N}F(i,j)\cdot I(x-i,y-j)$$

$$F*(G*I)=(F*G)*I$$

When, smooth an image and detect edges

$\Downarrow$

Convolve the guassian filter with a derivative filter

1

and convolve result with image.

2.

1. 选出特定范围的变化强度，并与原图卷积，凸出变化强度高的区域

类似 High Pass Filter) 特定范围以外为0.

But more accurate

② Edge Detection

1. Convert a 2D image into a set of curves
    1. Extract salient features of the scene
    2. More compact than pixels.

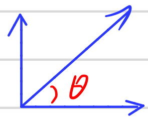
2. Origin of Edge
    1. Surface 2. Depth 3. Surface Color 4. Illumination

3. Image Gradient 图像梯度
    1. Gradient of an image

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

2. $\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$ 向右变淡 $\nabla f = \left[0, \frac{\partial f}{\partial x}\right]$ 向下变淡

$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$ 沿某特定角度的梯度下降



① The gradient points in the direction of most rapid increase in intensity

② The gradient direction is given by $\theta = \tan^{-1}\left(\frac{\partial f}{\partial y} \Big/ \frac{\partial f}{\partial x}\right)$

③ The edge strength is given by the gradient magnitude

边缘强度 梯度幅度

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

4. The Discrete Gradient 离散梯度

1. Differentiate a digital image $F[x,y]$
    1. Reconstruct a continous image, then take gradient
    2. Take discrete derivative ("Finite Difference")

$$\frac{\partial f}{\partial x}[x,y] \approx F[x+1,y] - F[x,y]$$

5. The Sobel Operator

1. Better approximation of the derivatives exist
2. Definition: Omits the $\frac{1}{8}$ term

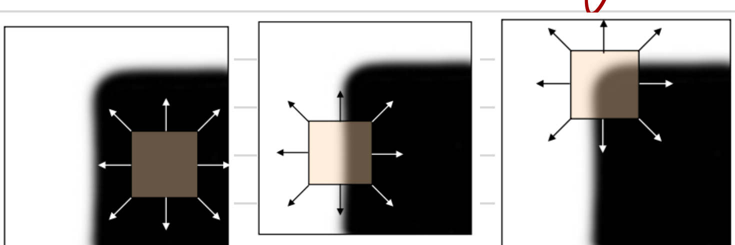
6. 2D Edge Detection

Usually use a separable filter such that: $G_\sigma(x,y) = G_\sigma(x)G_\sigma(y)$

$$\nabla S = \nabla(G_\sigma * I) = \begin{bmatrix} \frac{\partial(G_\sigma * I)}{\partial x} \\ \frac{\partial(G_\sigma * I)}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial G_\sigma}{\partial x} * I \\ \frac{\partial G_\sigma}{\partial y} * I \end{bmatrix} = \begin{bmatrix} G'_\sigma(x)G_\sigma(y) * I \\ G_\sigma(x)G'_\sigma(y) * I \end{bmatrix}$$

7. Corner Detection

1. Corner = Large change in intensity in at least 2 directions.
2. Shift a window in any direction to identify "corners"



"flat" region: no intensity change

"edge": no change along the edge direction

"corner": significant change in at least 2 directions

$$SSD(\Delta x, \Delta y) = \sum_{x,y \in P} \left(I(x,y) - I(x+\Delta x, y+\Delta y)\right)^2$$

8. Harris Corner Dectector

① $$SSD(\Delta x,\Delta y)=\sum_{x,y\in P}\left(I(x,y)-I(x+\Delta x,y+\Delta y)\right)^2$$

Let $I_x=\dfrac{\partial I(x,y)}{\partial x}$, $I_y=\dfrac{\partial I(x,y)}{\partial y}$

· Approximating by first order Taylor Series expansion

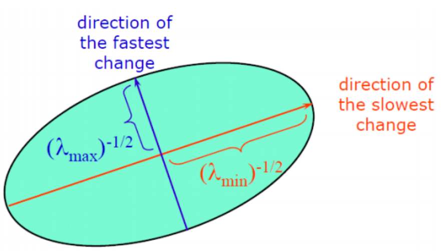
② $$I(x+\Delta x,y+\Delta y)\approx I(x,y)+I_x(x,y)\Delta x+I_y(x,y)\Delta y$$

Take ② into ①:

$$SSD(\Delta x,\Delta y)=\sum_{x,y\in P}\left(I(x,y)-I(x,y)-I_x(x,y)\Delta x-I_y(x,y)\Delta y\right)^2$$

$$\Rightarrow SSD(\Delta x,\Delta y)=\sum_{x,y\in P}\left(I_x(x,y)\Delta x+I_y(x,y)\Delta y\right)^2$$

9. Corner Detection



(1) Eigenvalues of M represent principal signal changes in two orthogonal direction in the neighborhood of (x,y)

M的特征值表示(x,y)周围两个正交方向的主要信号变化

(2) Therefore, corners can be found in locations where both eigenvalue are large.

(3) $\lambda_1\approx\lambda_2\approx 0$ · Image is uniform

$\lambda_1\approx 0$; λ_2 is large   Image varies in one direction (Edge)

λ_1,λ_2 are both large   Variations in all directions (Corner)

10. Cornerness Function

1. Compute eigenvalues expensive 计算量大

2. Cornerness function $C = \lambda_1\lambda_2 - k(\lambda_1+\lambda_2)^2 = det(M) - k \cdot trace^2(M)$

   k: sensitivity parameter determined empirically (0.04~0.15)

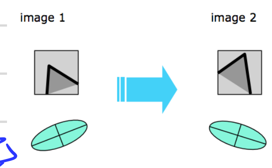
   Extract local maxima from the corness function

通过计算角点函数,取局部最大值,意味着此处就是Corner

11. Harris Dector: Properties

(1) Rotation:

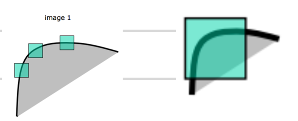
Ellipse rotates but keep shape
⇒ Cornerness invariant to image rotation



角的转动会引起椭圆转动 & 形状不变 ⇒ 图像旋转不会引起角的性质发生变化

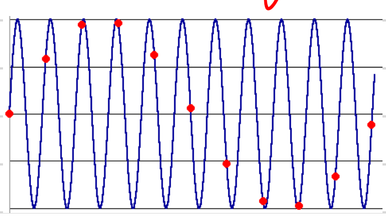
(2) Scale: Point classified as edge

⇒ Not invariant to scale



对角的放大操作会引起其他点被认为 edge

12. Image Scaling



1. Image sub-sampling

Per cycle ≥ 2 sample

Doesn't cause aliasing — wrong signal/image

2. Subsampling with Gaussian Pre-filtering

Solution: filter the image, then sub-sample

先使图像平滑,再对图像进行缩放

13. Gaussian Pyramid Construction

Repeat: (1) Repeat

(2) Subsample: Until minium resolution reached

(3) Can specify desired number of level

14. SIFT (Scale Invariant Feature Transform)

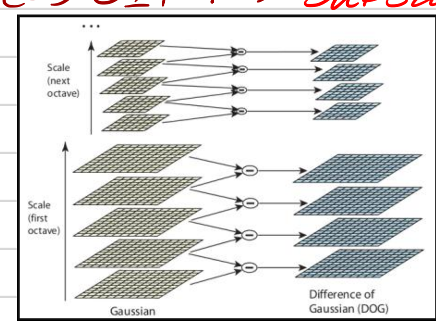
(1) SIFT features invariant to

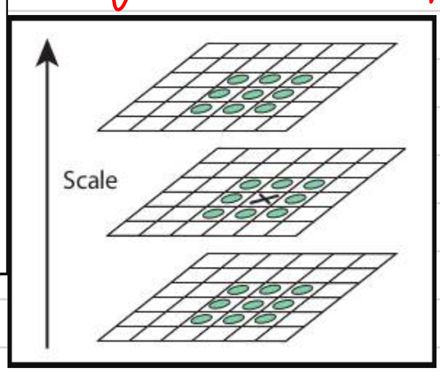
• Resolution • Scaling • Small changes in view

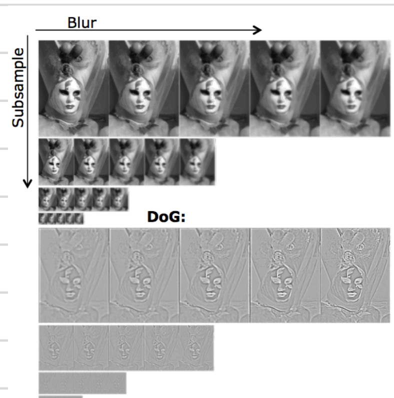
• Illumination

(2) Computionally expensive

(3) SIFT ⇒ Subsampling and bluring







1. Scale-space pyramid: Repeated blurring and subsampling
2. DoG ⇒ Difference of Gaussian Pyramid

⇓

Subtract successive smoothed images.

3. Keypoint: Local extreme in DoG pyramid

DOG:将一个原始灰度图像的模糊图像由另一幅灰度图像进行增强,通过DOG降低模糊图像的模糊度