RoboCup 2010

2. Filter

Content

1. Introduction

Definition

- A filter is a process that removes from a signal some unwanted component or feature.

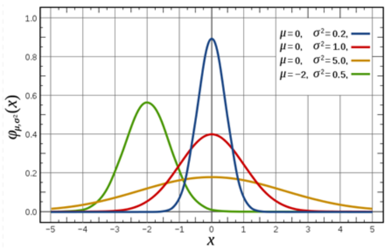
Typical filters

- Low-pass filter – low frequencies are passed, high frequencies are attenuated. (→ smooth image)
- High-pass filter – high frequencies are passed, low frequencies are attenuated. (→ detect edges and corners)

(from Wikipedia)

Noise removal: Gauss (1)

Gaussian distribution: $f(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

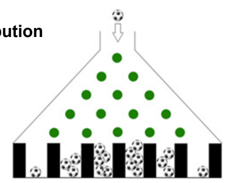


(from Wikipedia)

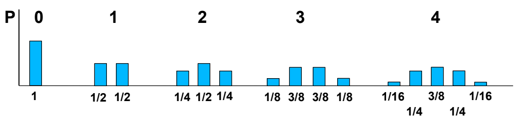
Noise removal: Gauss (2)

Approximation of the Gaussian distribution

1. Drop ball
2. Probability that ball moves left: P=0.5
3. Ball moves to next level
4. Continue with 2



Number of experiments (levels)



Noise removal: Gauss (3)

Approximation of the Gaussian distribution

- Binomial distribution ***B(n, p)***:
  The binomial distribution is the discrete probability distribution of the number of successes in a sequence of ***n*** independent yes/no experiments, each of which yields success with probability ***p***.
- The probability of getting exactly ***k*** successes in ***n*** trials is given by the probability mass function:

$$\mathrm{P}(K=k) = \binom{n}{k} p^k (1-p)^{n-k} \quad \text{where "n choose k" is} \quad \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

- Mean of the distribution: μ = n * P
- Variance: σ^2 = n * P * (1 - P)
- Standard deviation: σ = sqrt [n * P * (1 - P)]

Noise removal: Gauss (4)

Calculation of probabilities for P=0.5:

experiments		count
0	**1**	**1**
1	**1 1**	**2**
2	**1 2 1**	**4**
3	**1 3 3 1**	**8**
4	**1 4 6 4 1**	**16**
5	**1 5 10 10 5 1**	**32**

Binomial coefficients

$(a+b)^N$ = ?

$(a+b)^2 = a^2 + 2ab + b^2$

$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$

1. Noise removal: Gauss (5)

Application to images

- Apply 2D filter mask to all pixel positions
- Approximation of a 2D Gaussian function
- 1D binomial distribution ***B(n=2, p=0.5)***: (1, 2, 1) / 4
- 2D binomial distribution

$$\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} / 16$$

Noise removal: Gauss (6)

Complexity

- Linearly separable: Gaussian blur can be applied to a 2D image as two independent 1D filters.
- 2D filter: O (FW * FH * IW * IH)
- Two 1D filter: O (FW * IW * IH) + O (FH * IW * IH)
- Example
  Filter size: FW=15, FH=15
  Image size: IW=1000, IH=1000
  2D: 225 * 10e6
  1D: 30 * 10e6

→ Task Group 1: Implementation of a Gaussian filter, use masks (1,2,1)/4 and (1,3,3,1)/8

Noise removal: Median

Median filter

- For each pixel at position (x,y):
  1. Get list of pixels in the local neighborhood
  2. Sort list
  3. Replace pixel at position (x,y) with the median: value that separates the higher and lower half of the sample

→ Task Group 2: Implementation of a Median filter as neighborhood list use 9 and 25 pixels

Morphological Operators (1)

$$\begin{aligned} \textit{Dilatation}: \quad D_B(I(x)) &= \max\,\{I(x+r) \mid r \in B\}, \\ \textit{Erosion}: \quad E_B(I(x)) &= \min\,\{I(x+r) \mid r \in B\}. \end{aligned}$$

original erosion dilatation

Morphological Operators (2)

$$\begin{aligned} \textit{Opening}: \quad O_B(x) &= D_B\,[E_B\,(I(x))], \\ \textit{Closing}: \quad C_B(x) &= E_B\,[D_B\,(I(x))]. \end{aligned}$$

original erosion dilatation opening closing

- Opening: small objects and thin lines are removed
- Closing: holes within and between objects are filled

Edge detection: Sobel operator

- **Idea:** Edges emerge from brightness variations between adjacent pixels. Calculate the gradient of the image intensity function.
- Calculates the rate of change and the direction of the largest possible increase from light to dark. The result shows how "abruptly" the image changes at that point.
- Filter mask (Sobel operator):

$$G_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \qquad G_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

horizontal derivative — vertical derivative

Edge detection: Gradient Magnitude

Gradient magnitude (edge strength)

$$\mathbf{G} = \sqrt{\mathbf{G}_x^{\,2} + \mathbf{G}_y^{\,2}}$$

Direction of gradient

$$\Theta = \arctan\left(\frac{\mathbf{G}_y}{\mathbf{G}_x}\right)$$

Edge detection: Canny (1)

Calculation of an edge image: Canny edge detector

1) Smooth image with a Gaussian filter
2) Calculate gradients based on the *Sobel* operator
3) Get edge strength (gradient magnitude)
4) Calculate gradient orientation

$$\text{gradient orientation} = \begin{cases} \arctan\left(\frac{Grad_Y}{Grad_X}\right) & \text{if} \quad Grad_X \neq 0 \\ 0^0 & \text{if} \quad Grad_X = 0, Grad_Y = 0 \\ 90^0 & \text{if} \quad Grad_X = 0, Grad_Y \neq 0 \end{cases}$$

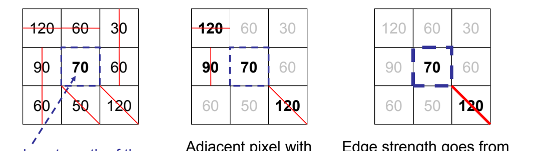
(use 4 edge orientations: 0˚, 45˚, 90˚, and 135˚)

Edge detection: Canny (2)

- Edges should have a width of 1 pixel:
  → Step 5: Non maxima suppression
  (search for local maxima)
- Edge pixel is **preserved** if:
  - Current edge pixel has higher edge strength than all adjacent pixels **or**
  - there is at least on adjacent pixel with a higher edge strength than the current pixel for which:
    the direction of the edge strength goes from the adjacent pixel with the higher edge strength to the current pixel.

Edge detection: Canny (3)

Example: Search for local maxima



edge strength of the current pixel

Edge direction

Adjacent pixel with higher edge strength

Edge strength goes from the adjacent pixel to the current pixel
→ edge pixel is not removed

Edge detection: Canny (4)

Step 6: Hysteresis (selection of edge pixels)

- 2 threshold values are defined: T_{low} and T_{high}
- in the case of edge strength > T_{high}
  → apply pixel as edge pixel
- in the case of T_{low} < edge strength < T_{high} and pixel adjoins to an edge pixel:
  → apply pixel as edge pixel

Result: edge image

Example Canny (1)

Example: Canny-edge detector

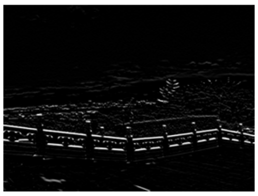
original image

smoothed image

1. Smooth image

Example Canny (2)

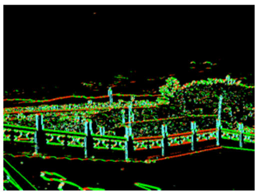
gradient in x-direction



gradient in y-direction

2. Images of gradients

Example Canny (3)



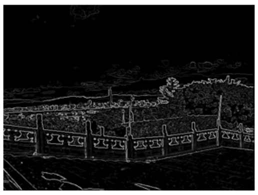
0 degree, 45 degree, 90 degree, 135 degree

3. Edge strength

4. Edge direction

Example Canny (4)

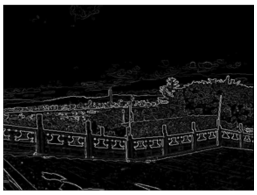
without search for local maxima



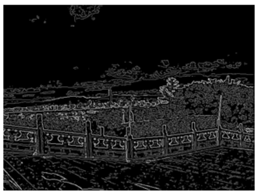
search for local maxima

5. Search for local maxima

Example Canny (5)



edge image without threshold values



weak and sharp edges

6a. Edge image before hysteresis

Example Canny (6)

selected weak and sharp edges

edge image

6b. Edge image after hysteresis