Image Processing : a DSP application

Dr Rozenn Dahyot

http://www.mee.tcd.ie/~sigmedia

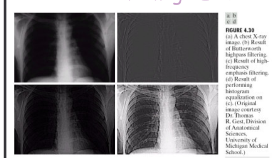
Edge detection

Image understanding

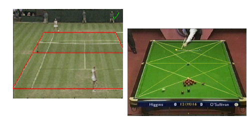
Segmentation

Object detection/recognition

Restoration/Image Enhancement



Video Indexing



Contents

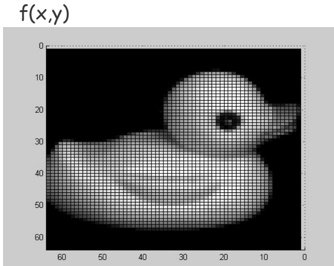
- Image as a 2D continuous signal
- Image as a 2D digital signal
  - standard methods for derivation
  - smoothing before derivating
- Applications

Image as a 2D Continuous Signal

Image: a surface

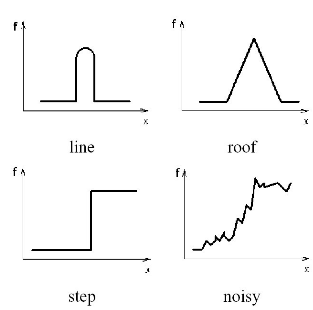
f(x,y)

y

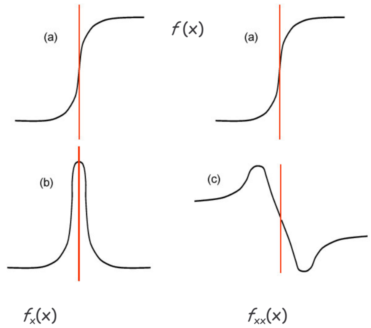


x

Type of Edges



Differentiation



The Gradient

First derivatives

$$\nabla f(x,y) = \left| \begin{array}{l} \boxed{f_x}(x,y) = \dfrac{\partial f(x,y)}{\partial x} \\[2ex] \boxed{f_y}(x,y) = \dfrac{\partial f(x,y)}{\partial y} \end{array} \right.$$

The Laplacian

2nd derivatives

$$\boxed{\Delta f(x,y)} = \boxed{f_{xx}}(x,y) + \boxed{f_{yy}}(x,y)$$

Laplacian

Edge detection

1. Compute $|\nabla f(x,y)|$
2. Select $|\nabla f(x,y)| > \boxed{s}$ threshold
3. Thinning:
   1. Local maxima of $|\nabla f(x,y)|$
   2. Zero-crossing of $\Delta f(x,y)$

Image as a 2D Digital Signal

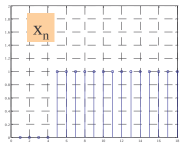
Image: an array of numbers

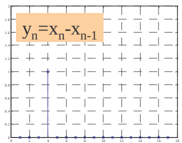
The Gradient

- First derivatives

$$f_x(x,y) \approx f(x,y) - f(x-1,y)$$

$$f_y(x,y) \approx f(x,y) - f(x,y-1)$$





(x,y) are now discrete !

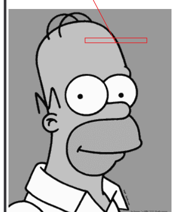
Computed by Convolution

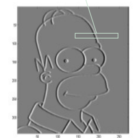
212 212 212 212 212 212 10 10 150 150 150 150

! Convolution

$\otimes H_x = [1 \quad -1]$

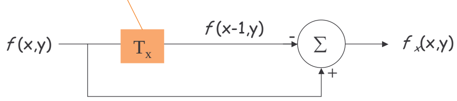
0 0 0 0 0 202 0 −140 0 0 0 0





Block diagram

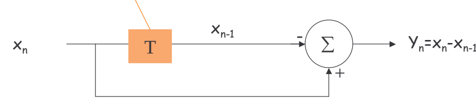
Shift in x operator



Impulsive response (f(0,0)=1, f(x,y)=0 ∀(x,y)≠(0,0))
cf. handout 8, p6:
$H_x(0,0)=1$
$H_x(1,0)=-1$
$H_x(x,y)=0$ otherwise

Block diagram

Shift in x operator



Impulsive response ($x_0=1$, $x_n=0$ ∀n>0)
cf. handout 8, p6:
$H_0=1$
$H_1=-1$
$H_n=0$ otherwise

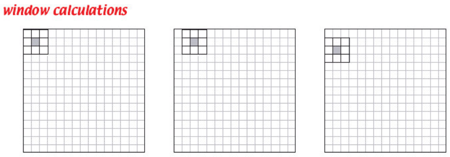
A one dimensional filter applied to rows and columns of the image

Computed by Convolution

Other filters
averaging in the orthogonal direction : $H_x = \begin{bmatrix} 0 & 1 & -1 \\ 0 & 1 & -1 \\ 0 & 1 & -1 \end{bmatrix}$

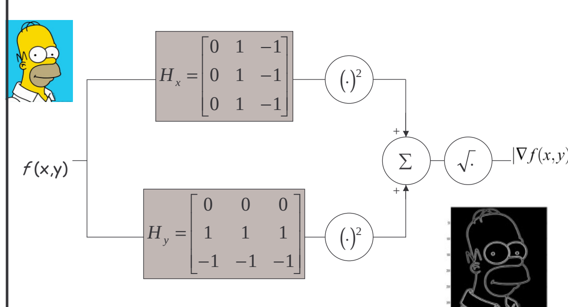
- ***Local processing within a moving window***

moving window calculations



output at: pixel 2, row2 *pixel 3, row 2* *pixel 2, row 3*

Magnitude of the Gradient



The Laplacian

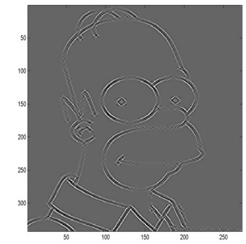
- Second derivatives

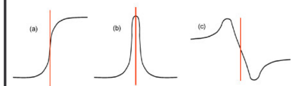
$$\Delta f(x,y) \approx$$
$$f(x+1,y) - f(x-1,y) +$$
$$f(x,y+1) - f(x,y-1) - 4f(x,y)$$

$$L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

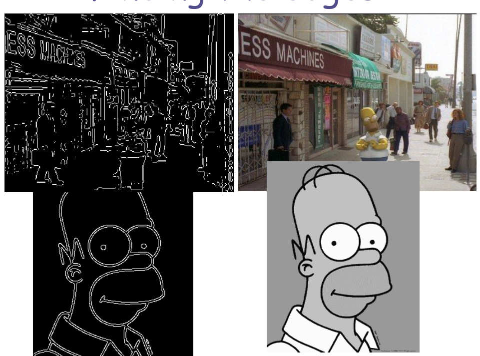
The Laplacian

$$L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$





Finding the edges



Remarks

- Enhancing the edge J
- Enhancing the noise L

Remember handout 2, p4 :

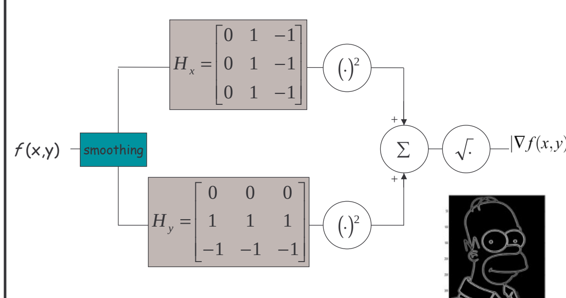
f'(x) à jw F(jw)-f(0)

f''(x) à (jw)² F(jw)- (jw) f(0)-f'(0)

High Frequencies are enhanced by differentiation

Smoothing before gradient Calculation using Gaussian filters

Smoothing before gradient calculation for noise removal



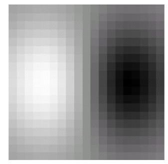
Gaussian filter

$$G(x,y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right) \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right) = G(x) \times G(y)$$

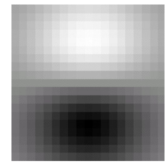
- Smoothing (low-pass) filter
- Separable filter (faster processing)

Derivative of Gaussian (DroG)

$$\nabla(\tilde{f} = G \otimes f) = \left| \begin{array}{l} \tilde{f}_x = G_x(x) \otimes G(y) \otimes f \\ \tilde{f}_y = G(x) \otimes G_y(y) \otimes f \end{array} \right.$$



$G_x(x) \otimes G(y)$

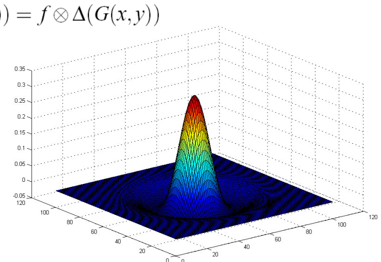


$G(x) \otimes G_y(y)$

Laplacian of Gaussian (LoG)

$$\Delta(f \otimes G(x,y)) = f \otimes \Delta(G(x,y))$$

Mexican hat



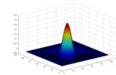
$$\Delta(G(x,y)) = \frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] \exp\left(-\frac{x^2}{2\sigma^2}\right) \cdot \exp\left(-\frac{y^2}{2\sigma^2}\right)$$

Laplacian of Gaussian (LoG)

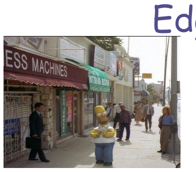
$f(x,y)$ → G → Δ → $f \otimes \Delta(G(x,y))$

Equivalent to :

$f(x,y)$ → ΔG → $f \otimes \Delta(G(x,y))$



Edge detection



Using Pixel difference

Using DroG

Applications

Unsharp Masking

$$\hat{f}(x,y) = f(x,y) - (k \cdot \Delta f(x,y))$$

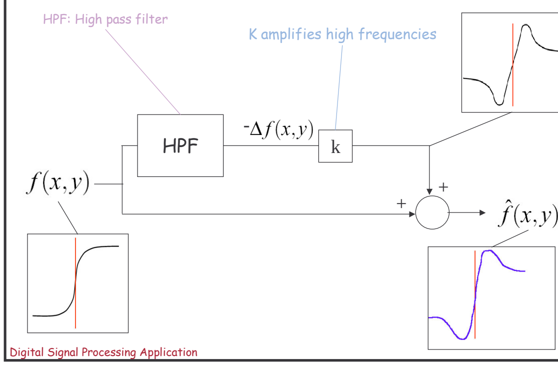
Original image

Laplacian of the image

Enhanced edge image

Amplifying term

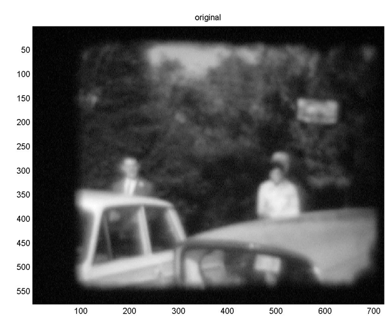
Unsharp Masking



For image enhancement

Application to astronomical images, medical images (Xray photo), etc.

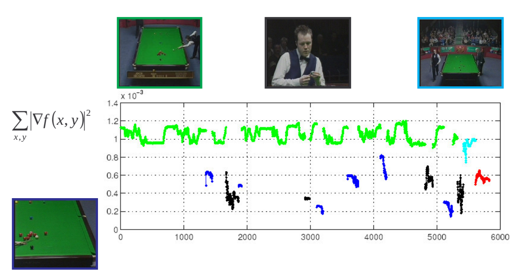
For Image Restoration



For Image Restoration

Sport Video Indexing

Inter-Shot Analysis in Sport Videos using the “sum of edges” in images



Sport Video Indexing

Intra-Shot Analysis in Sport Videos using the "Hough transform" (looking at the straight lines)

Sport Video Indexing

Any questions ?