CS9840
Learning and Computer Vision
Prof. Olga Veksler

Lecture 2

Some Concepts from Computer Vision

Some Slides are from Cornelia, Fermüller, Mubarak Shah,

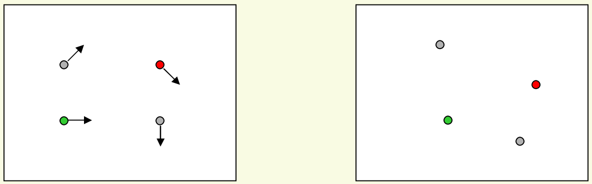
Gary Bradski,

Sebastian Thrun

Outline

- Some Concepts in Image Processing/Vision
  - Optical Flow Field (related to motion field)
  - Correlation
- Next time:
  - *“Recognizing Action at a Distance” by A. Efros, A.Berg, G. Mori, Jitendra Malik*
    - *Also maybe: "80 million tiny images: a large dataset for non-parametric object and scene recognition", A. Torralba, R. Fergus, W. Freeman*
  - there should be a link to PDF file on our web site
  - Discuss the paper and watch video

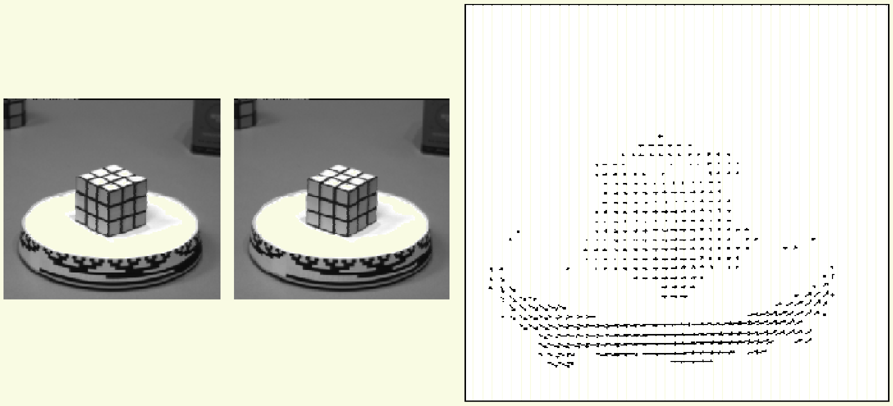
Optical flow



first image I_1 ***second image I_2***

- How to estimate pixel motion from image I_1 to image I_2?
  - Solve pixel correspondence problem
    - given a pixel in I_1, look for **nearby** pixels of the **same** color in I_2
- Key assumptions
  - **color constancy:** a point in I_1 looks the same in I_2
    - For grayscale images, this is **brightness constancy**
  - **small motion**: points do not move very far
- This is called the **optical flow** problem

Optical Flow Field



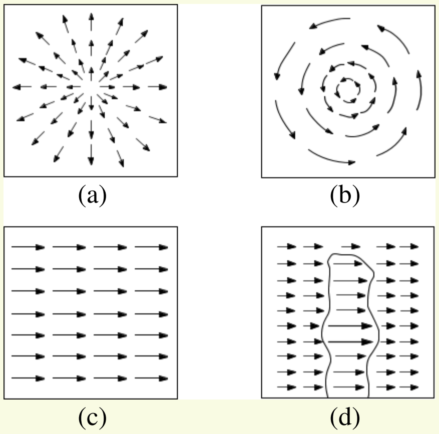
Optical Flow and Motion Field

- Optical flow field is the apparent motion of brightness patterns between 2 (or several) frames in an image sequence
- Why does brightness change between frames?
- Assuming that illumination does not change:
  - changes are due to the RELATIVE MOTION between the scene and the camera
  - There are 3 possibilities:
    - Camera still, moving scene
    - Moving camera, still scene
    - Moving camera, moving scene

Motion Field (MF)

- The MF assigns a velocity vector to each pixel in the image
- These velocities are INDUCED by the RELATIVE MOTION between the camera and the 3D scene
- The MF is the *projection* of the 3D velocities on the image plane

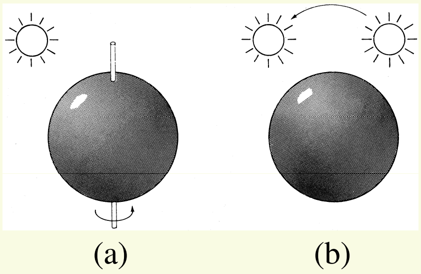
Examples of Motion Fields



(a) Translation perpendicular to a surface. (b) Rotation about axis perpendicular to image plane. (c) Translation parallel to a surface at a constant distance. (d) Translation parallel to an obstacle in front of a more distant background.

Optical Flow vs. Motion Field

- Recall that Optical Flow is the apparent motion of brightness patterns
- We equate Optical Flow Field with Motion Field
- Frequently works, but now always:

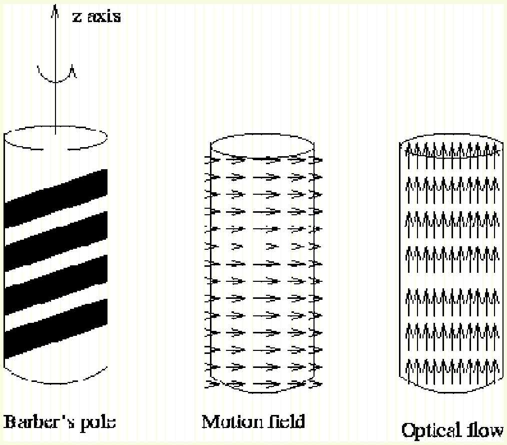


(a) A smooth sphere is rotating under constant illumination. Thus the optical flow field is zero, but the motion field is not

(b) A fixed sphere is illuminated by a moving source—the shading of the image changes. Thus the motion field is zero, but the optical flow field is not

Optical Flow vs. Motion Field

- Often (but not always) optical flow corresponds to the true motion of the scene



Computing Optical Flow: Brightness Constancy Equation

- Let ***P*** be a moving point in 3D:
  - At time ***t***, ***P*** has coordinates (***X***(***t***),***Y***(***t***),***Z***(***t***))
  - Let ***p***=(***x***(***t***),***y***(***t***)) be the coordinates of its image at time ***t***
  - Let ***E***(***x***(***t***),***y***(***t***),***t***) be the brightness at ***p*** at time ***t***.
- Brightness Constancy Assumption:
  - As ***P*** moves over time, ***E***(***x***(***t***),***y***(***t***),***t***) remains constant

Computing Optical Flow: Brightness Constancy Equation

$$E(x(t), y(t), t) = Constant$$

Taking derivative wrt time:

$$\frac{dE(x(t), y(t), t)}{dt} = 0$$

$$\frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial y}\frac{dy}{dt} + \frac{\partial E}{\partial t} = 0$$

Computing Optical Flow: Brightness Constancy Equation

1 equation with 2 unknowns

$$\frac{\partial E}{\partial x}\frac{dx}{dt} + \frac{\partial E}{\partial y}\frac{dy}{dt} + \frac{\partial E}{\partial t} = 0$$

Let

$$\nabla E = \begin{bmatrix} \frac{\partial E}{\partial x} \\ \frac{\partial E}{\partial y} \end{bmatrix}$$ (Frame spatial gradient)

$$v = \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix}$$ (optical flow)

and $$E_t = \frac{\partial E}{\partial t}$$ (derivative across frames)

Computing Optical Flow: Brightness Constancy Equation

- How to get more equations for a pixel?
  - Basic idea: impose additional constraints
    - most common is to assume that the flow field is smooth locally
    - one method: pretend the pixel's neighbors have the same (u,v)
      - If we use a 5x5 window, that gives us 25 equations per pixel!

$$E_t(p_i)+\nabla E(p_i)\cdot[u \quad v]=0$$

$$\begin{bmatrix} E_x(p_1) & E_y(p_1) \\ E_x(p_2) & E_y(p_2) \\ \vdots & \vdots \\ E_x(p_{25}) & E_y(p_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -\begin{bmatrix} E_t(p_1) \\ E_t(p_2) \\ \vdots \\ E_t(p_{25}) \end{bmatrix}$$

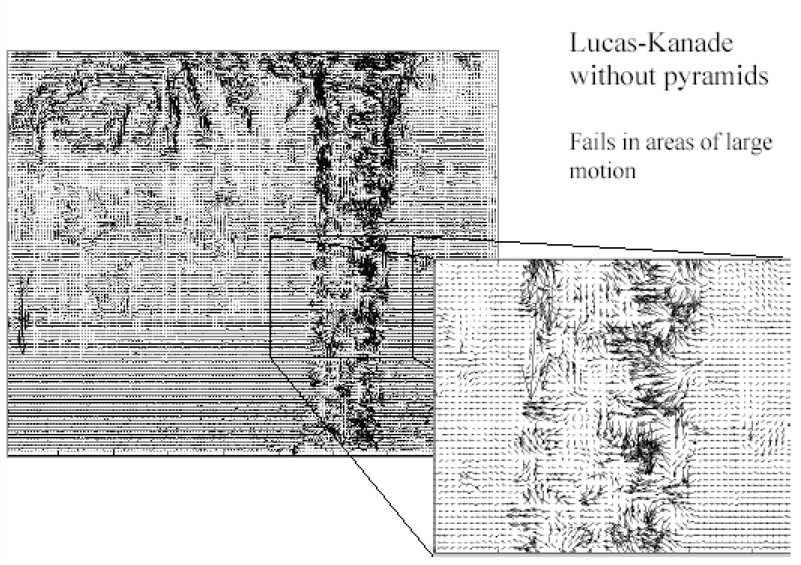
matrix ***E***	vector ***d***	vector ***b***
25x2	2x1	25x1

Video Sequence

*

* Picture from Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Optical Flow Results



Lucas-Kanade without pyramids

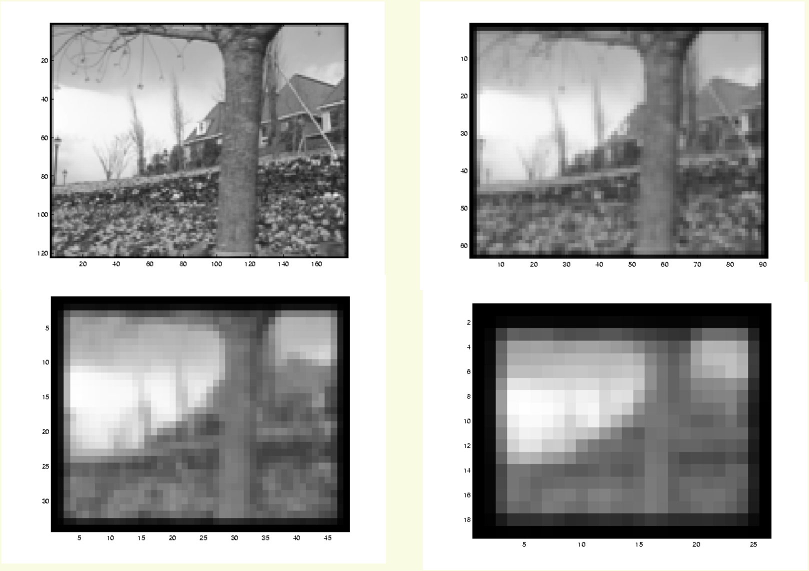
Fails in areas of large motion

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

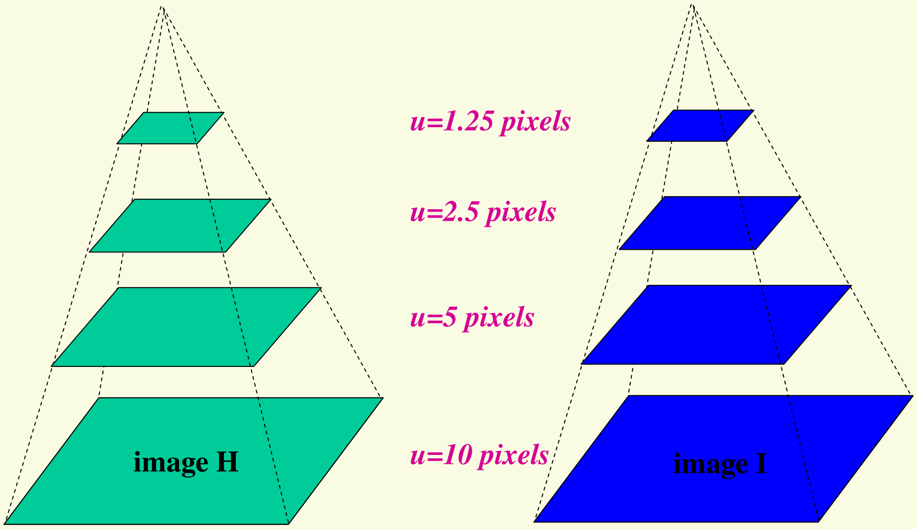
Revisiting the small motion assumption

- Is this motion small enough?
  - Probably not—it's much larger than one pixel (2nd order terms dominate)
  - How might we solve this problem?

Reduce the resolution!



Coarse-to-fine optical flow estimation



u=1.25 pixels

u=2.5 pixels

u=5 pixels

u=10 pixels

image H

image I

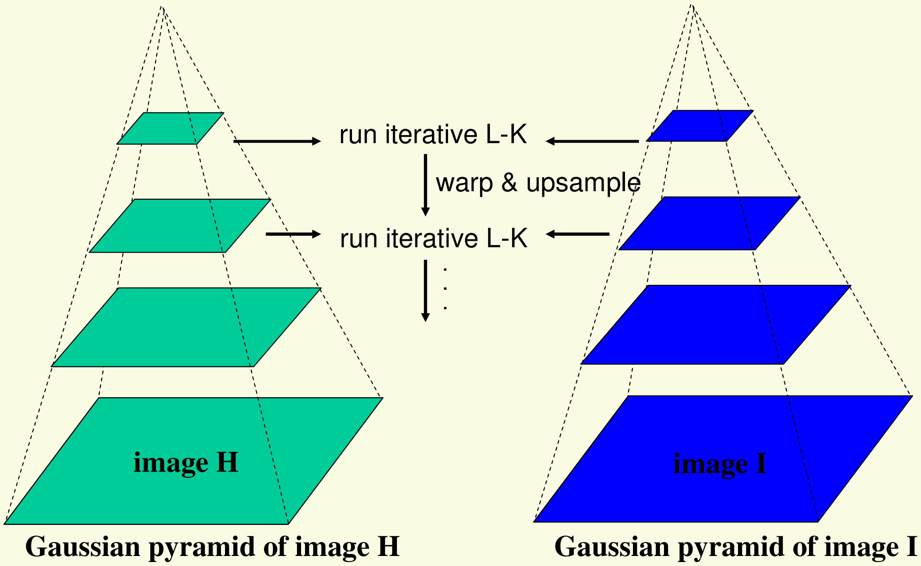
Gaussian pyramid of image H

Gaussian pyramid of image I

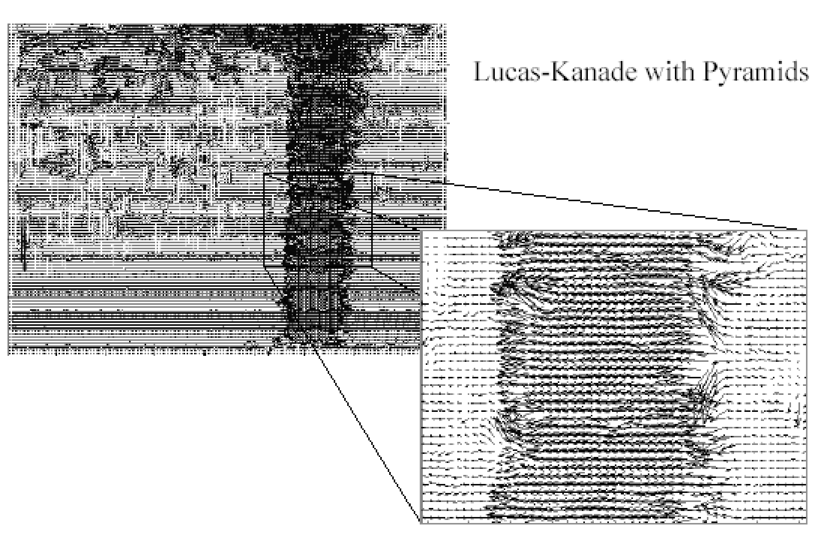
Iterative Refinement

- Iterative Lukas-Kanade Algorithm
  1. Estimate velocity at each pixel by solving Lucas-Kanade equations
  2. Warp H towards I using the estimated flow field
     - use image warping techniques
  3. Repeat until convergence

Coarse-to-fine optical flow estimation

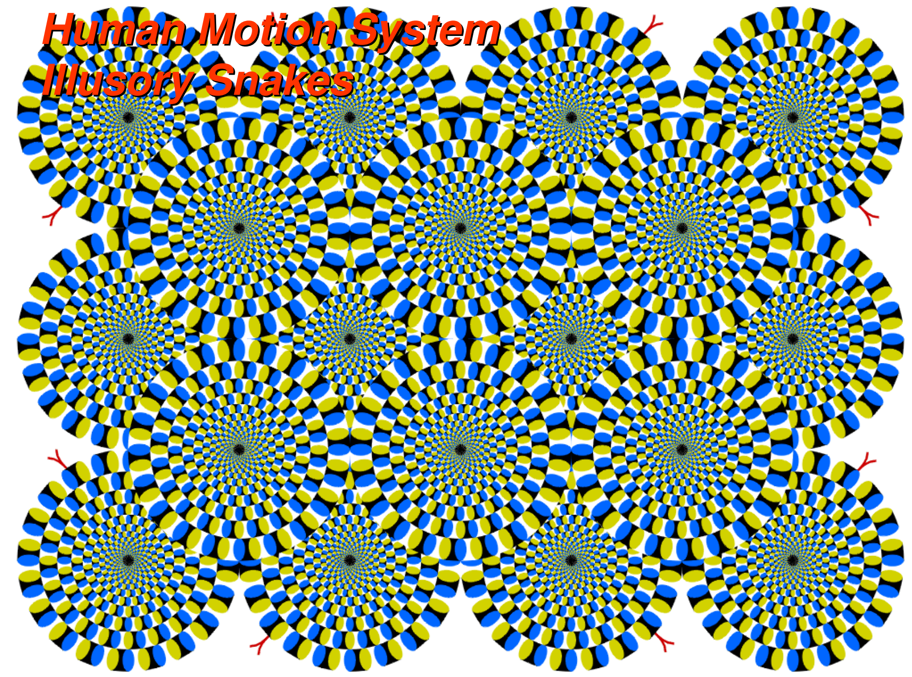


Optical Flow Results



* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

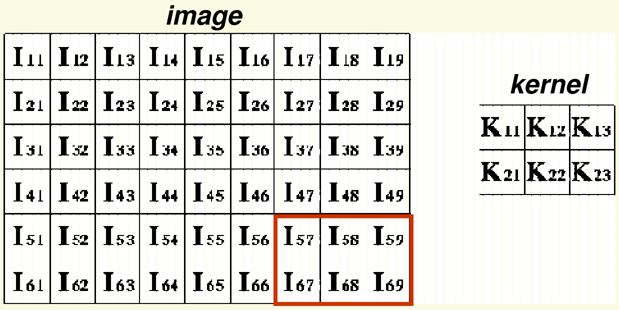
Human Motion System
Illusory Snakes



from Gary Bradski and Sebastian Thrun

Other Concepts to Review

- Convolution is the operation of applying a “kernel” to each pixel of an image



image

I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	I_{16}	I_{17}	I_{18}	I_{19}
I_{21}	I_{22}	I_{23}	I_{24}	I_{25}	I_{26}	I_{27}	I_{28}	I_{29}
I_{31}	I_{32}	I_{33}	I_{34}	I_{35}	I_{36}	I_{37}	I_{38}	I_{39}
I_{41}	I_{42}	I_{43}	I_{44}	I_{45}	I_{46}	I_{47}	I_{48}	I_{49}
I_{51}	I_{52}	I_{53}	I_{54}	I_{55}	I_{56}	I_{57}	I_{58}	I_{59}
I_{61}	I_{62}	I_{63}	I_{64}	I_{65}	I_{66}	I_{67}	I_{68}	I_{69}

kernel

K_{11}	K_{12}	K_{13}
K_{21}	K_{22}	K_{23}

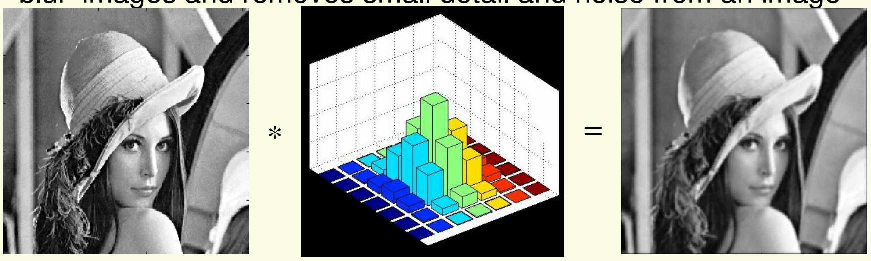
- Result of convolution has the same dimension as the image
- For example:

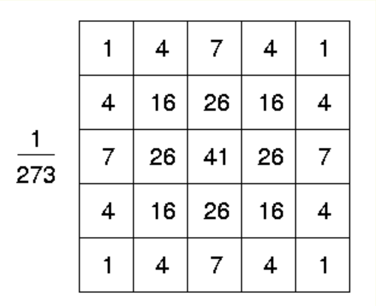
$$O_{57} = I_{57}K_{11} + I_{58}K_{12} + I_{59}K_{13} + I_{67}K_{21} + I_{68}K_{22} + I_{69}K_{23}$$

- Convolution is frequently denoted by *, for example I*K

Other Concepts to Review

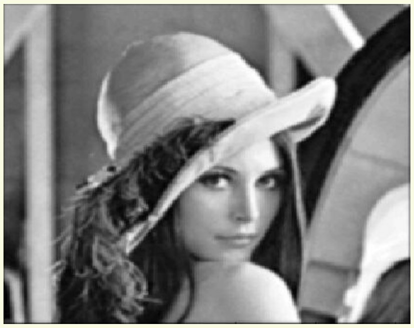
- Gaussian smoothing (blurring): convolution operator that is used to `blur' images and removes small detail and noise from an image





$$\frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

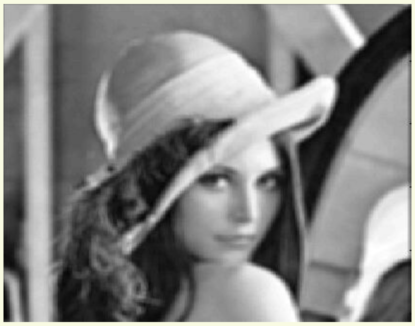
Gaussian vs. Smoothing



Gaussian Smoothing

$\frac{1}{273}$

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



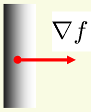
Smoothing by Averaging

$\frac{1}{25}$

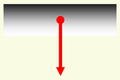
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Other Concepts to Review

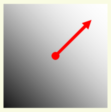
- Image gradient: points in the direction of the most rapid increase in intensity of image ***f***



$\nabla f = \left[\frac{\partial f}{\partial x}, 0\right]$



$\nabla f = \left[0, \frac{\partial f}{\partial y}\right]$



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

- Sobel operator to compute gradient:

$\frac{1}{8}$

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

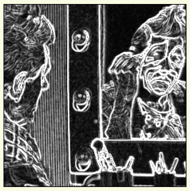
$\frac{\partial f}{\partial x}$

$\frac{1}{8}$

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

$\frac{\partial f}{\partial y}$

- Results:

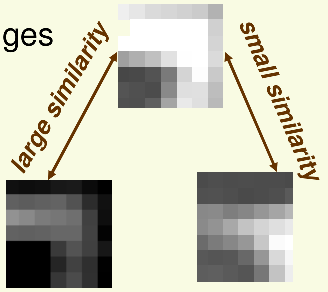


Other Concepts to Review

- Cross-correlation

$$c(f,g)=\sum_{i=1}^{d} f(i)g(i)$$

  - measures similarity between images (or image regions) f and g
  - works OK if there is no change in intensity
- Normalized cross correlation, more popular in image processing
  - Insensitive to linear intensity changes between image patches f and g



$$NCC(f,g)=\frac{\sum_{i=1}^{d}\left(f(i)-\bar{f}\right)\left(g(i)-\bar{g}\right)}{\left[\sum_{i=1}^{d}\left(f(i)-\bar{f}\right)^{2}\sum_{k=1}^{d}{}^{2}\left(g(i)-\bar{g}\right)\right]^{1/2}}$$

Next Time

- Paper:*“Recognizing Action at a Distance” by A. Efros, A.Berg, G. Mori, Jitendra Malik*
  - *Also maybe: "80 million tiny images: a large dataset for non-parametric object and scene recognition", A. Torralba, R. Fergus, W. Freeman*
- When reading the paper, think about following:
  - What is the problem paper tries to solve
  - What makes this problem difficult?
  - What is the method used in the paper to solve the problem
  - What is the contribution of the paper (what new does it do)?
  - Do the experimental results look “good” to you?