CS 414 – Multimedia Systems Design

Lecture 4 – Digital Image Representation

Klara Nahrstedt

Spring 2009

Administrative

- Group Directories will be established hopefully today (or latest by Friday)
- MP1 will be out on 1/28 (today)
- Start by reading the MP1 and organizing yourself as a group this week, start to read documentation, search for audio and video files.

Images – Capturing and Processing

Capturing Real-World Images

- Picture – two dimensional image captured from a real-world scene that represents a momentary event from the 3D spatial world

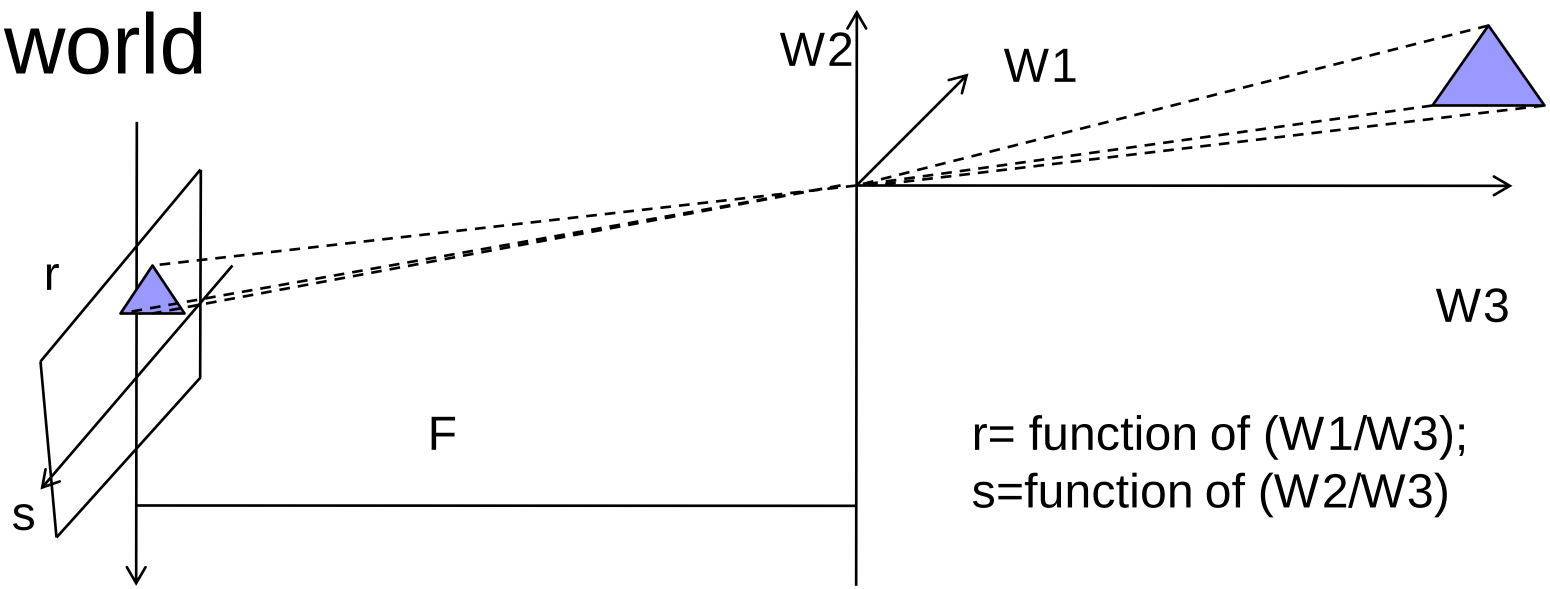


Image Concepts

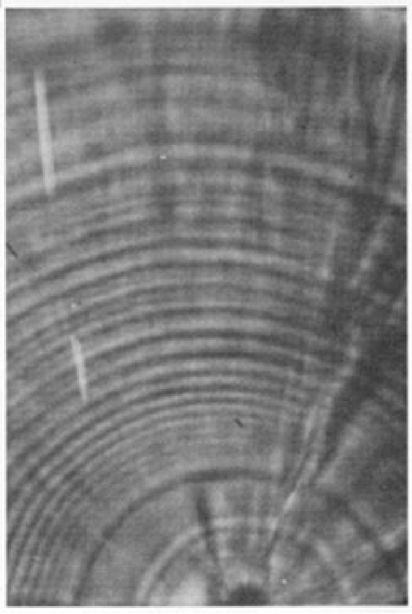
- An image is a function of intensity values over a 2D plane $I(r,s)$
- Sample function at discrete intervals to represent an image in digital form
  - matrix of intensity values for each color plane
  - intensity typically represented with 8 bits
- Sample points are called pixels

Digital Images

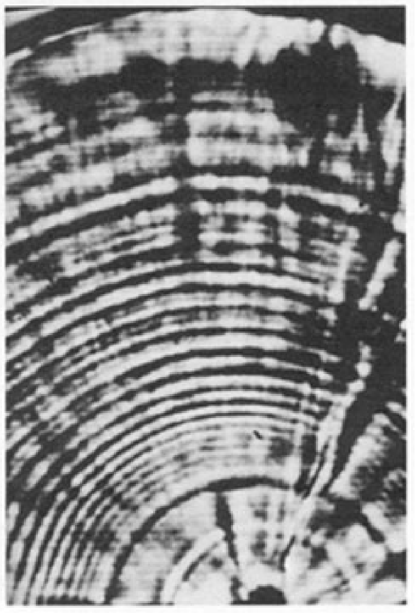
- Samples = pixels
- Quantization = number of bits per pixel
- Example: if we would sample and quantize standard TV picture (525 lines) by using VGA (Video Graphics Array), video controller creates matrix 640x480pixels, and each pixel is represented by 8 bit integer (256 discrete gray levels)

Image Representations

- Black and white image
  - single color plane with 2 bits
- Grey scale image
  - single color plane with 8 bits
- Color image
  - three color planes each with 8 bits
  - RGB, CMY, YIQ, etc.
- Indexed color image
  - single plane that indexes a color table
- Compressed images
  - TIFF, JPEG, BMP, etc.



4 gray levels

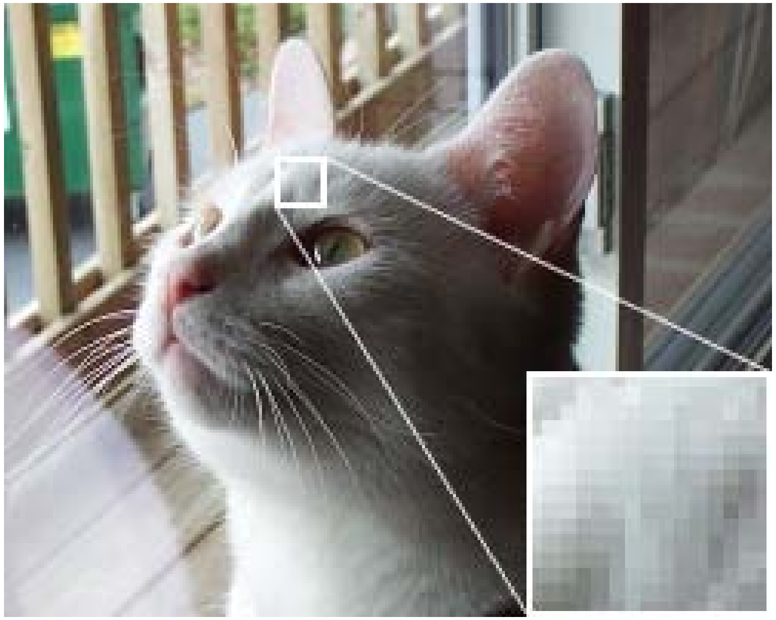


2gray levels

Digital Image Representation (3 Bit Quantization)

111	111	011	011	011	011	111	111
111	011	111	111	111	111	011	111
000	111	001	111	111	001	111	000
010	111	111	111	111	111	111	010
000	111	100	111	111	100	111	000
000	111	111	100	100	111	111	000
111	000	111	111	111	111	000	111
111	111	000	000	000	000	111	111

Color Quantization
Example of 24 bit RGB Image



24-bit Color Monitor

Image Representation Example

24 bit RGB Representation (uncompressed)

128	135	166	138	190	132
129	255	105	189	167	190
229	213	134	111	138	187

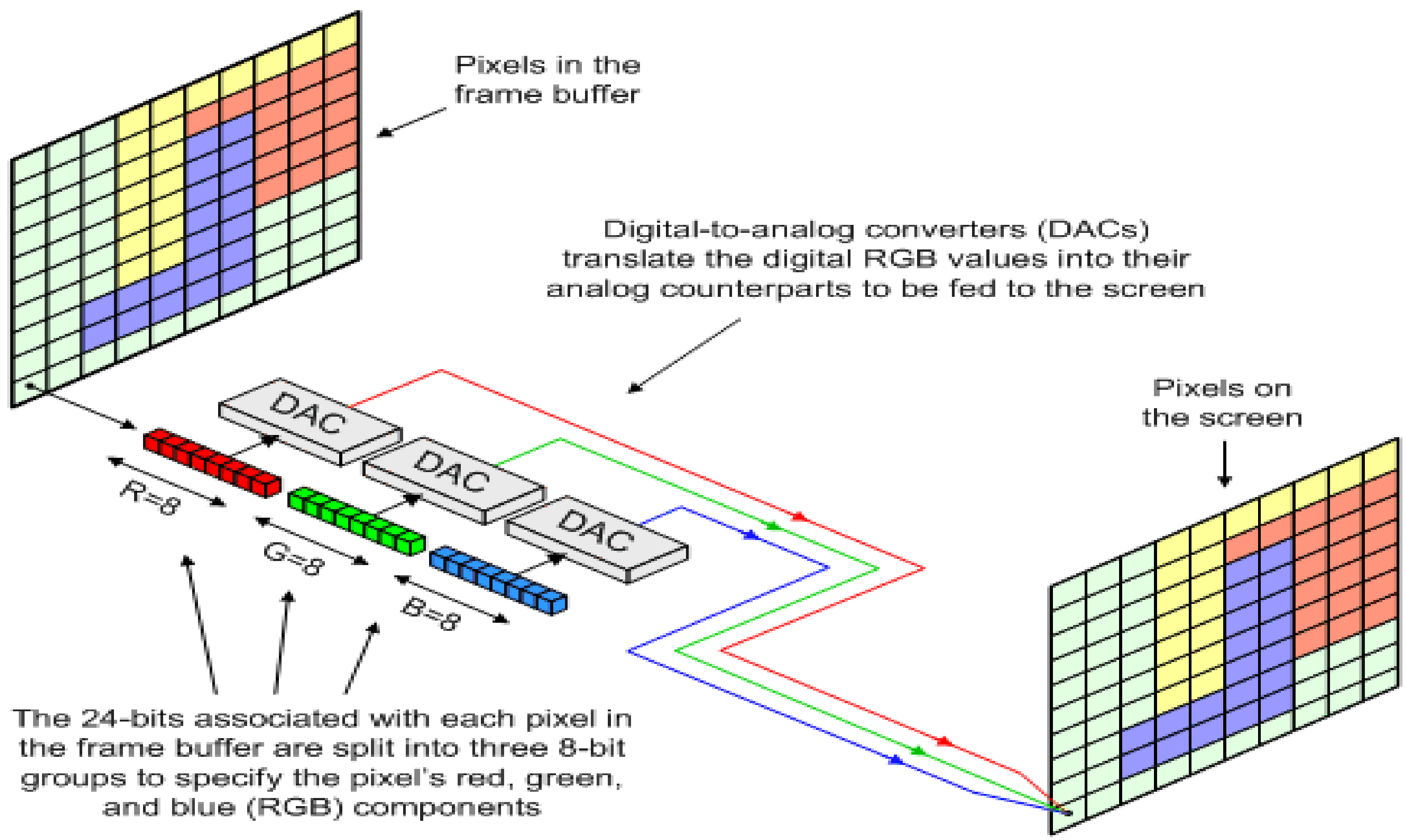
128	138
129	189
229	111

135	190
255	167
213	138

166	132
105	190
134	187

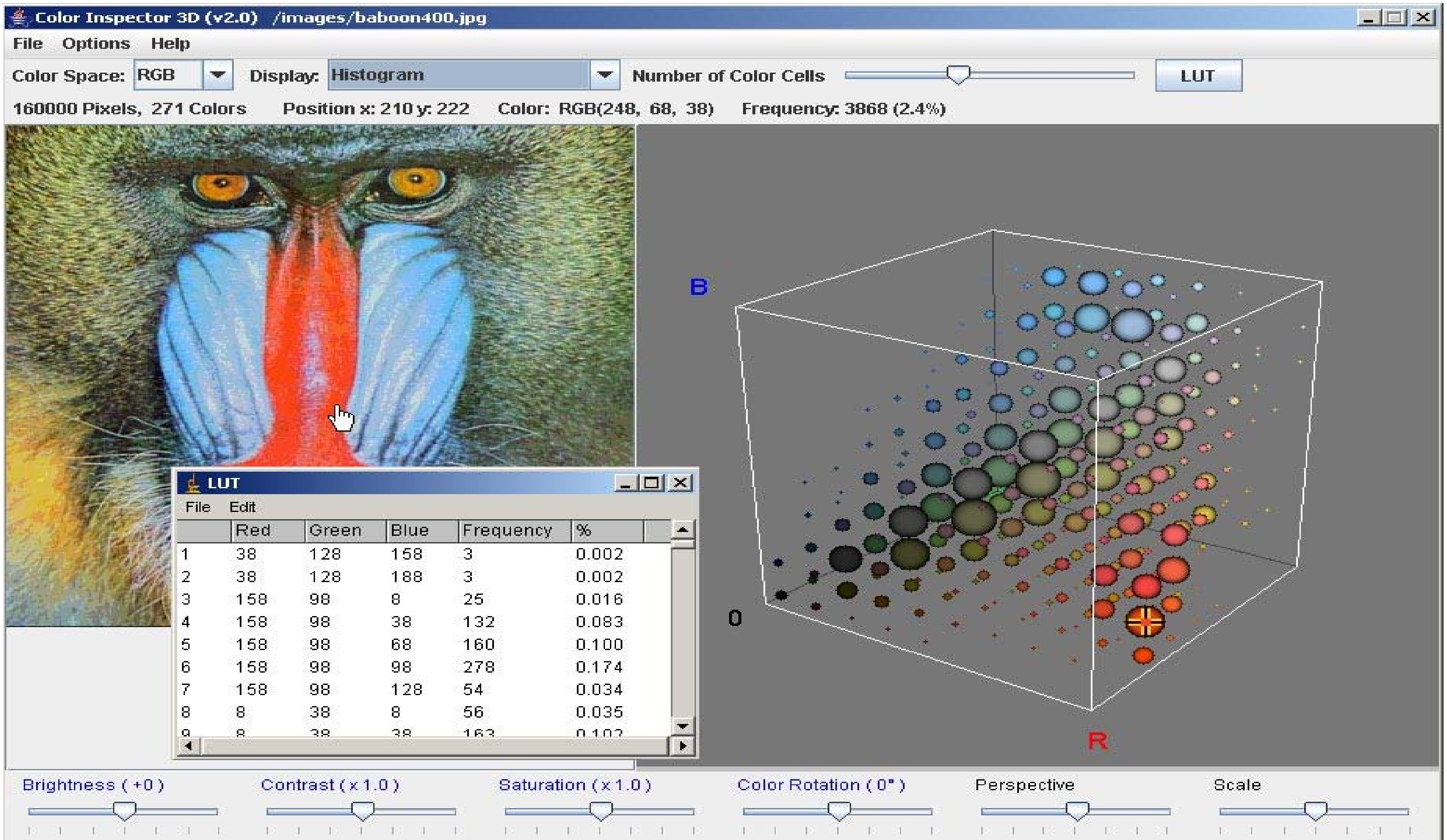
Color Planes

Graphical Representation



Graphical representation of 24-bit "true color"

Image Properties (Color)



Color Histogram

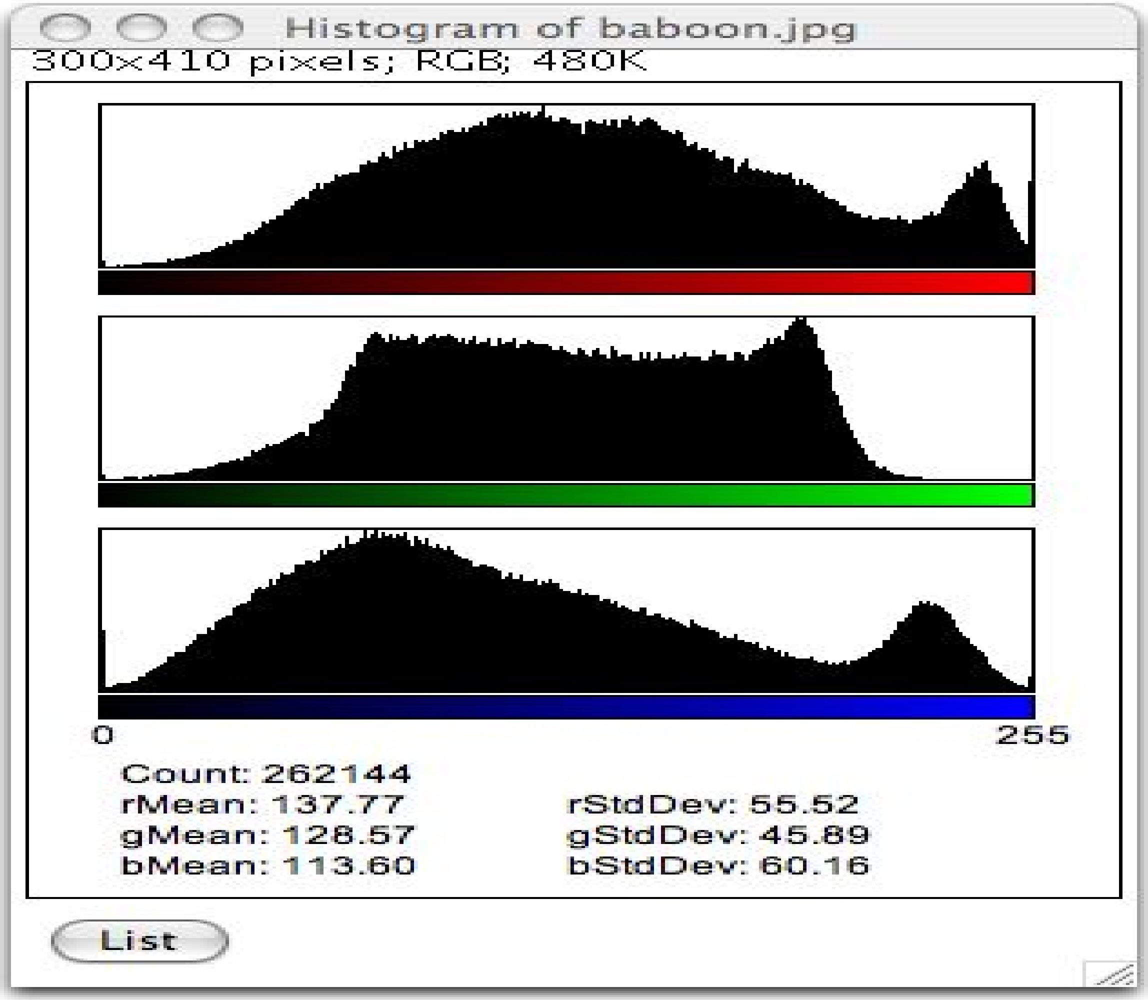
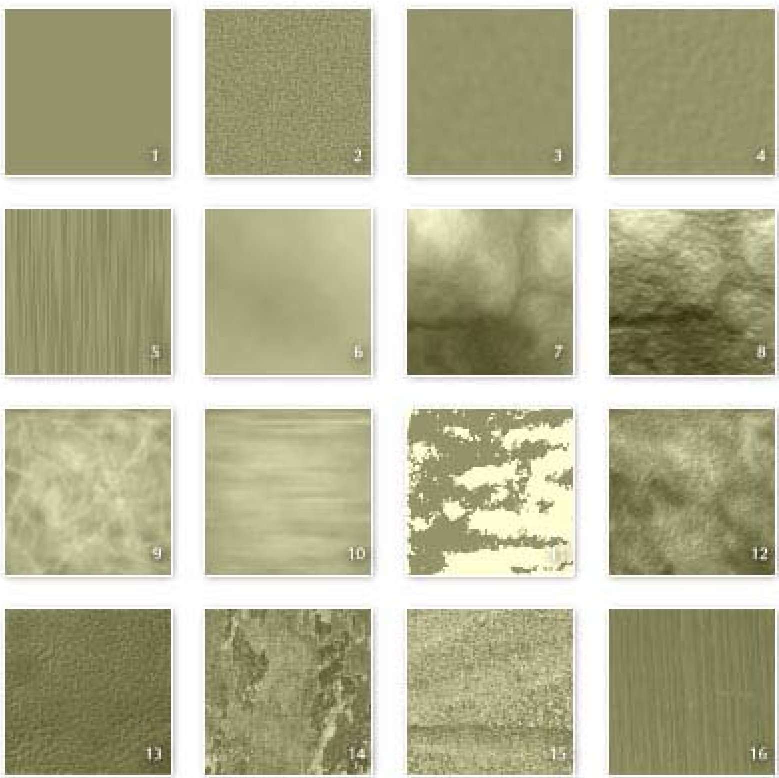


Image Properties (Texture)

- Texture – small surface structure, either natural or artificial, regular or irregular
- Texture Examples: wood barks, knitting patterns
- Statistical texture analysis describes texture as a whole based on specific attributes: regularity, coarseness, orientation, contrast, …

Texture Examples



Spatial and Frequency Domains

- Spatial domain
  - refers to planar region of intensity values at time t

- Frequency domain
  - think of each color plane as a sinusoidal function of changing intensity values
  - refers to organizing pixels according to their changing intensity (frequency)

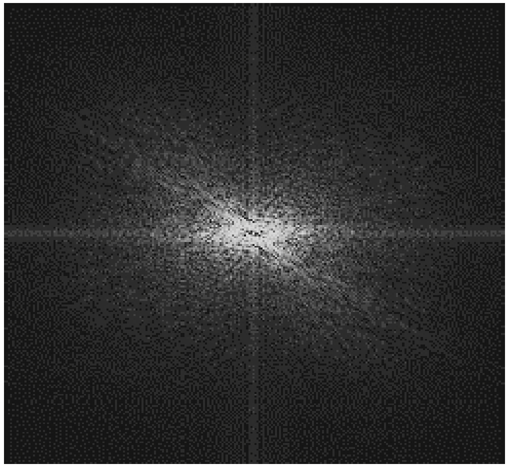
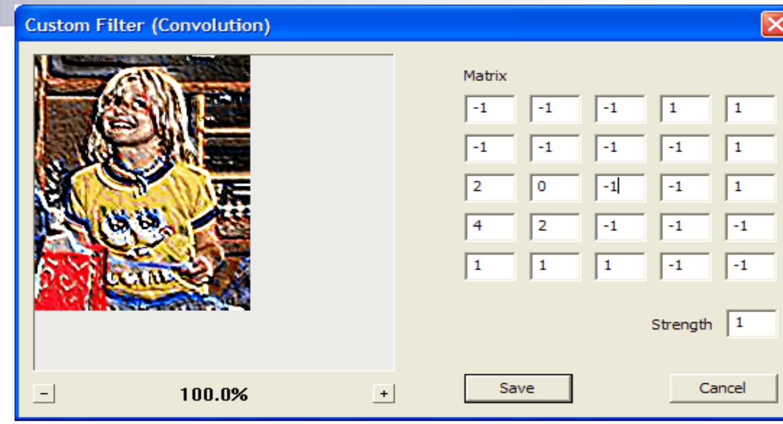


Image Processing Function: 1. Filtering

- Filter an image by replacing each pixel in the source with a weighted sum of its neighbors
- Define the filter using a *convolution mask*, also referred to as a *kernel*
  - non-zero values in small neighborhood, typically centered around a central pixel
  - generally have odd number of rows/columns

Convolution Filter



100	100	100	100	100
100	100	50	50	100
100	100	100	100	100
100	100	100	100	100
100	100	100	100	100

X

	0	1	0	
	0	0	0	
	0	0	0	

=

100	100	100	100	100
100	100	50	50	100
100	100	50	100	100
100	100	100	100	100
100	100	100	100	100

Mean Filter

20	12	14	23
45	15	19	33
55	34	81	22
8	64	49	95

Subset of image

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Convolution filter

Mean Filter

20	12	14	23
45	15	19	33
55	34	81	22
8	64	49	95

Subset of image

$$\frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}$$

Convolution filter

Common 3x3 Filters

- Low/High pass filter

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

- Blur operator

$$\frac{1}{13}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

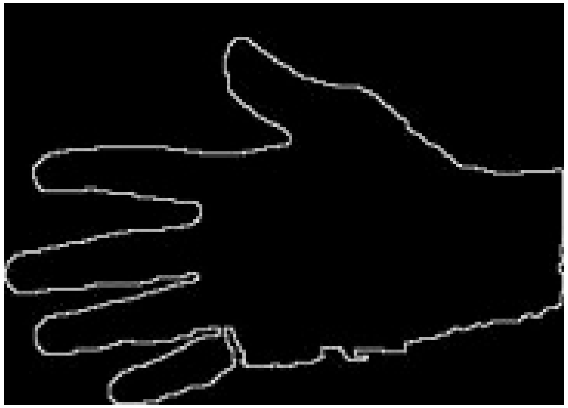
- H/V Edge detector

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \qquad \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Example

Image Function: 2. Edge Detection

- Identify areas of strong intensity contrast
  - filter useless data; preserve important properties
- Fundamental technique
  - e.g., use gestures as input
  - identify shapes, match to templates, invoke commands



Edge Detection

Simple Edge Detection

- Example: Let assume single line of pixels

5	7	6	4	152	148	149

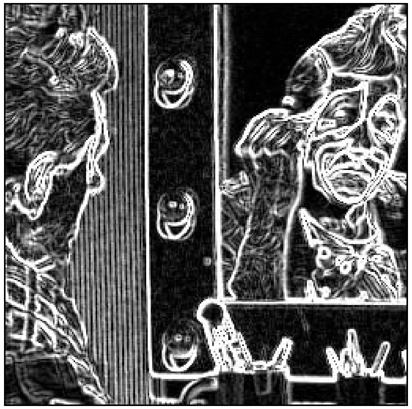
- Calculate 1st derivative (gradient) of the intensity of the original data
  - **Using gradient, we can find peak pixels in image**
  - *I(x)* represents intensity of pixel *x* and
  - *I'(x)* represents gradient (in 1D),
  - Then the gradient can be calculated by convolving the original data with a mask (-1/2 0 +1/2)
  - $I'(x) = -1/2 * I(x-1) + 0 * I(x) + ½ * I(x+1)$

Basic Method of Edge Detection

- Step 1: filter noise using mean filter
- Step 2: compute spatial gradient
- Step 3: mark points > *threshold* as edges

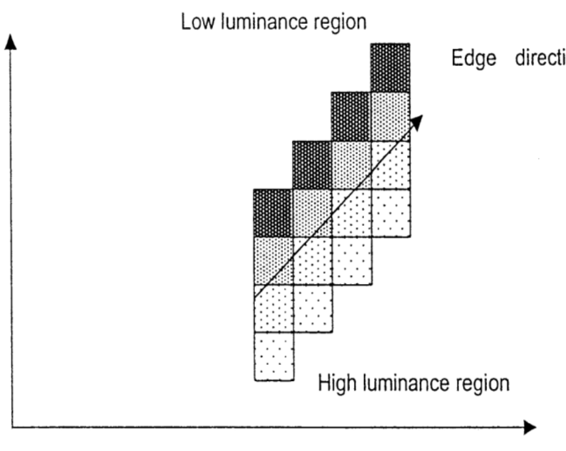
Mark Edge Points

- Given gradient at each pixel and threshold
  - mark pixels where *gradient > threshold* as edges



Compute Edge Direction

- Calculation of Rate of Change in Intensity Gradient
- Use 2nd derivative
- Example: (**5 7 6 4 152 148 149)**
- Use convolution mask (+1 -2 +1)
- I''(x) = 1*I(x-1) -2*I(x) + 1*I(x+1)
- **Peak detection in 2nd derivate is a method for line detection**.



Summary

- Other Important Image Processing Functions
  - Image segmentation
  - Image recognition
    - Formatting
    - Conditioning
    - Marking
    - Grouping
    - Extraction
    - Matching
  - Image synthesis