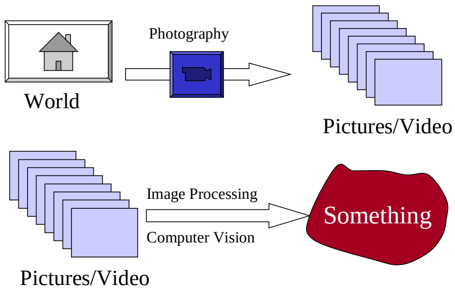
Image Processing (Computer Vision)

"Inverse Photography"



Stages in Computer Vision

- Physics: Image Formation (Light, Reflectance)
- Physics: Cameras: Optics (Lens), Sensors (CCD, CMOS)
- Image Processing: Coding (Transmission, Compression)
- Image Processing: Enhancement (Noise Cleaning, Colors)
- IP-CV: Feature Detection (Objects, Actions, Motion)
- Computer Vision: Scene recovery (3D, Reflectance)
- Computer Vision: Object Recognition
- Human and Machine Vision: Visual Perception
- Robotics: Control Action (autonomous driving)

Vision in Nature = Smart, Moving

- Only smart and moving organisms see!
  - Plants do not have eyes
- Visual recognition at early development
  - Babies recognize and track the mother very early
- Most of the brain is involved in vision processing

Application: Recognition

1984 2000

National Geographics: "Afghan Girl"

Panoramic Stereo Mosaics

(Last Exercise)

Image Processing: 2005/2006

Teacher: **Shmuel Peleg** <peleg@cs.huji.ac.il>
Assistant: **Yael Pritch** <yaelpri@cs.huji.ac.il>
Ozer Horaa: **??????** <???@cs.huji.ac.il>
Textbook:
Gonzalez & Woods, Digital Image Processing (2nd Ed.), Addison Wesley, 2002.
Jain, Pratt, Rosenfeld,….

Expected Work:
4 Written Exercises
4 individual computer exercises (MATLAB)
Grading:
Exam: 60-70%; Exercises: 30-40%

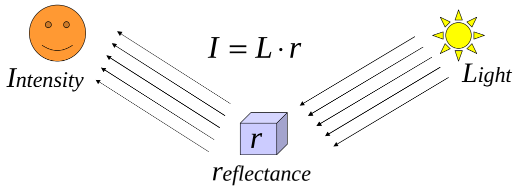
Relevant Computer Vision Courses

- Image Processing (Peleg)
- Computer Vision Seminar (Sundays 10-12)
- Computer Graphics (Sun 12-14, Wed 16-18, Lischinski)
- Introduction to Machine Learning (Mon 08-10, Tue 14-16, Shashua)
- Mathematical Methods in IP, CV, and CG (Thu 10-12)

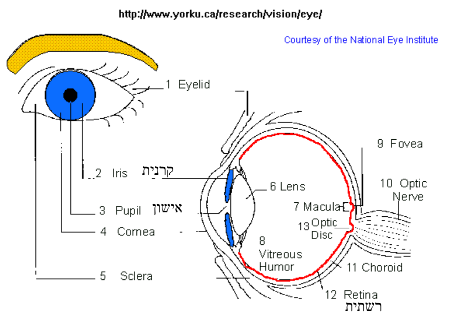
- Computer Vision (Peleg-Shashua-Weinshall)
- Image Sequence Analysis (Seminar, Peleg)
- Medical Image Processing & Robotics (Joskowicz)

Image Formation

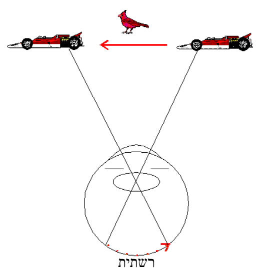
- Light is emitted by light source
- Light is reflected from objects
- Reflected light is sensed by eye or by camera



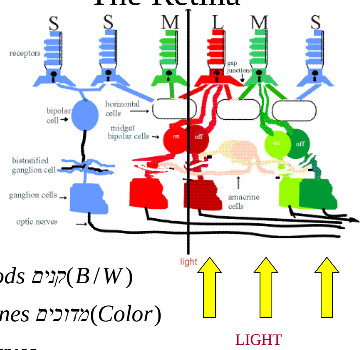
The Human Eye (2.1)



World to Retina Projection



The Retina

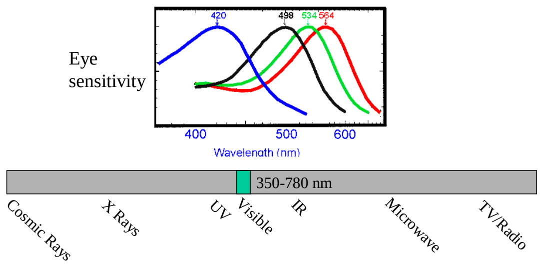


$120 \bullet 10^6$ *Rods* קנים(B/W)

$7 \bullet 10^6$ *Cones* מדוכים$(Color)$

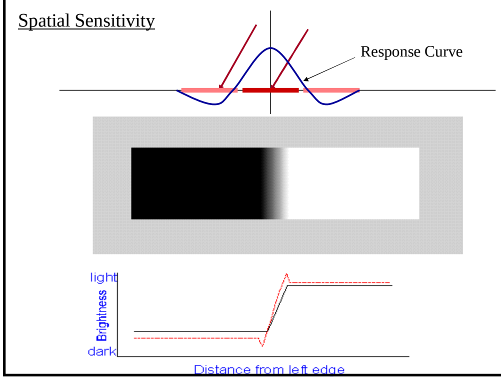
$\approx 10^4$ *Nerves*

Colors - Electromagnetic Radiation

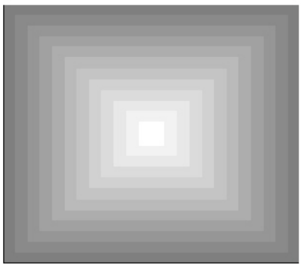


- Maximum Sun Energy: 450 nm
- Best Atmospheric Transmittance: Visible Range

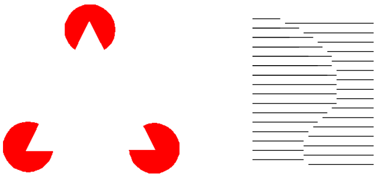
Mach Bands (1)



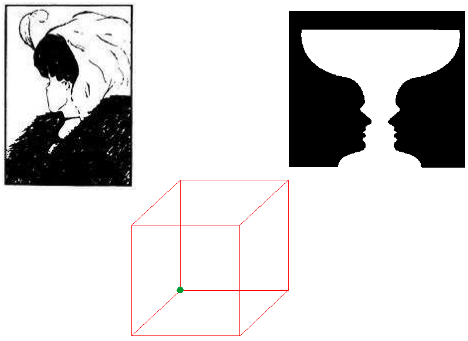
Mach Bands (2)



Visual Illusions



Dual Interpretations



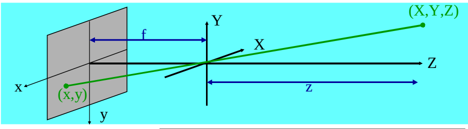
What is That?

Image Digitization (2.3)

- Transforming the 3D world into 2D image
  - Perspective Projection (Optics)
- Sampling the Image Plane
  - Finite number of **Pixels**
- Quantizing the color/gray-level
  - Finite number of colors

Perspective Projection (2.5.2)

- Transforming the 3D world into 2D image
  - Continuous Perspective Projection
  - optics



$$x = \frac{f}{Z} X$$

$$y = \frac{f}{Z} Y$$

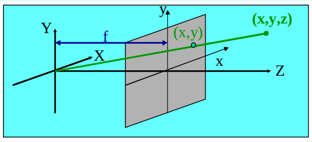


Image Sampling

- Sampling the Image Plane
  - Finite number of **Pixels**

Original | Reduce by 4 | Reduce by 16 | Reduce by 32

Color/Grey-level Quantization

- Quantizing the color/gray-level
  - Finite number of colors

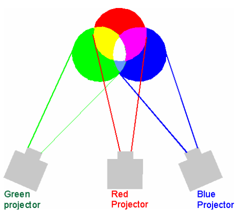
256 Levels | 16 Levels | 4 Levels | 2 Levels

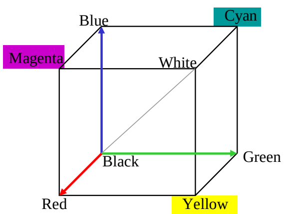
Digital Pictures

- A Matrix of numbers (B/W)
- A Matrix of triplets (RGB Color, etc.)

1	2	3	4	5	6	7	8	9	10	9	8	7	6	5
2	4	5	6	7	8	9	10	11	12	11	10	9	8	7
3	5	6	7	8	9	10	11	12	13	12	11	10	9	8
4	6	7	8	9	10	11	12	13	14	13	12	11	10	9
5	7	8	9	10	11	12	13	14	15	14	13	12	11	10
6	8	9	10	11	12	13	14	15	16	15	14	13	12	11
7	9	10	11	12	13	14	15	16	17	16	15	14	13	12
8	10	11	12	13	14	15	16	17	18	17	16	15	14	13
9	11	12	13	14	15	16	17	18	19	18	17	16	15	14
10	12	13	14	15	16	17	18	19	20	19	18	17	16	15
9	11	12	13	14	15	16	17	18	19	18	17	16	15	14
8	10	11	12	13	14	15	16	17	18	17	16	15	14	13
7	9	10	11	12	13	14	15	16	17	16	15	14	13	12
6	8	9	10	11	12	13	14	15	16	15	14	13	12	11
5	7	8	9	10	11	12	13	14	15	14	13	12	11	10
4	6	7	8	9	10	11	12	13	14	13	12	11	10	9
3	5	6	7	8	9	10	11	12	13	12	11	10	9	8
2	4	5	6	7	8	9	10	11	12	11	10	9	8	7
1	3	4	5	6	7	8	9	10	11	10	9	8	7	6

Color Spaces (4.6)

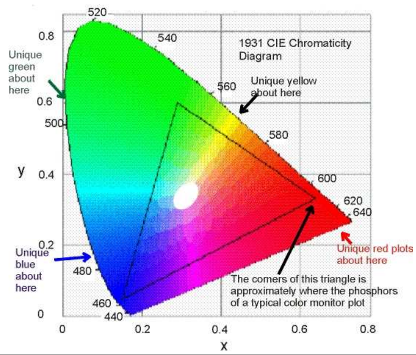




Y - Luminance

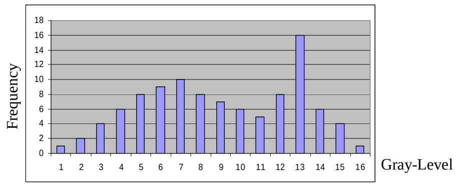
$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

CIE Chromaticity Diagram (1931)



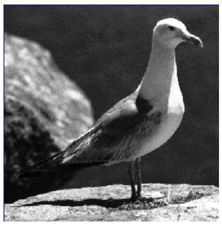
The Histogram

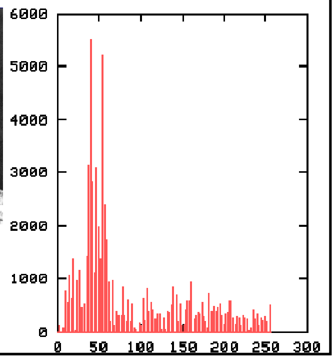
- Frequency counting of gray levels



- In continuous intensities:
  a continuous probability distribution *p(g)*

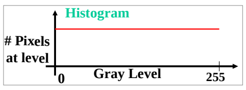
Histogram - Example

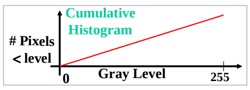




Histogram Equalization

- Equal usage of all gray levels



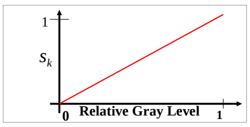


- Normalizing to range 0..1

N Pixels, Range *0 .. K-1*

n_i = # pixels at *i*

$s_k = \frac{1}{N}\sum_{i=1}^{k} n_i$



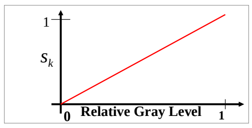
$r_k = \frac{k}{K}$

Histogram Equalization

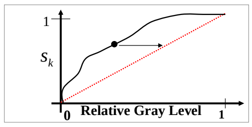
N Pixels, Range *0 .. K-1*

n_i = # pixels at *i*

$s_k = \frac{1}{N}\sum_{i=1}^{k} n_i$



$r_k = \frac{k}{K}$



$r_k = \frac{k}{K}$

Histogram Equalization (cont.)

N Pixels, Range *0 .. K-1*

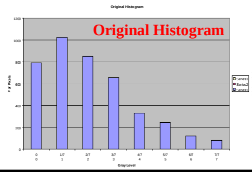
n_i = # pixels at *i*

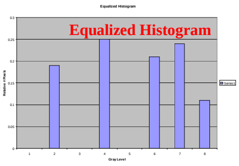
Normalized Cumulative Occurrence: $s_j = \frac{1}{N}\sum_{i=1}^{j} n_i$

- For every original level j:
  Change its gray level to $s_j \cdot (K-1)$
- Stretch gray levels back to [*0.. K-1*]

Equalization Example

Grey Level (*k*)	Normalized (*r*)	# Pixels (*n*)	Normalized (*n/N*)	Cumulative	Approx	Result
0	0	790	0.19	0.19	1/7	1
1	1/7	1023	0.25	0.44	3/7	3
2	2/7	850	0.21	0.65	5/7	5
3	3/7	656	0.16	0.81	6/7	6
4	4/7	329	0.08	0.89	6/7	6
5	5/7	245	0.06	0.95	1	7
6	6/7	122	0.03	0.98	1	7
7	7/7	81	0.02	1.00	1	7
Total:		4096	1			





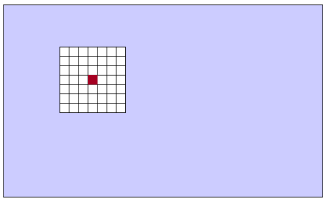
Examples for Equalization




Adaptive Histogram Equalization

- Different regions in a single image
  - Example: Coin on white paper
- Poor result for Histogram Equalization
  - Do the coins and paper separately
  - How to segment?
- Compute histogram in local regions around each pixel

Adaptive Equalization



- For each pixel
  - Compute Histogram in Neighborhood
  - Transform only the center pixel
- Go to next pixel

Color Quantization

- 24 bits/pixel - 8 bits/color -256^3 Colors
- 8 bits/pixel - 256 colors
  - 3-3-2 bits for R-G-B
  - *General* Quantization - Look Up Table (**LUT**)
  - **LUT** can be for **RGB** or for **YUV**

LUT	0	1	2	k	254	255
R	⋯	R_1	⋯	R_k	⋯	R_{255}
G	⋯	G_1	⋯	G_k	⋯	G_{255}
B	⋯	B_1	⋯	B_k	⋯	B_{255}

Quantization Error

LUT	0	1	2	k	254	255
R	⋯	R_1	⋯	R_k	⋯	R_{255}
G	⋯	G_1	⋯	G_k	⋯	G_{255}
B	⋯	B_1	⋯	B_k	⋯	B_{255}

- If pixel ***p*** with color ***(r,g,b)*** is coded by ***k***, a possible quantization error **for *p*** is:

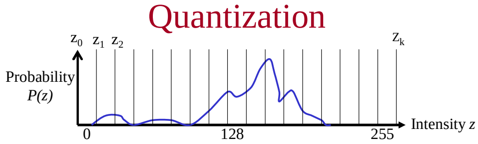
$$E_p^2 = (r - R_k)^2 + (g - G_k)^2 + (b - B_k)^2$$

- The total error introduced by a LUT is:

$$E^2 = \sum_p E_p^2$$

- Unknowns: $(r,g,b) \rightarrow k$, $k \rightarrow (R_k, G_k, B_k)$

Quantization

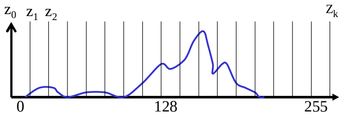


- Map the continuous intensities to $\{q_0, \ldots, q_{k-1}\}$
  - Borders of segments: $z_0, z_1, z_2, \ldots, z_k$
  - Represent each segment $[z_{i-1}, z_i]$ by intensity q_{i-1}
- Uniform Quantization: $q_i = (z_i + z_{i+1})/2$

$$z_{i+1} - z_i = (z_k - z_0)/k$$

- Prior Mappings (e.g. Gamma Correction)

Optimal Quantization (6.5.1)



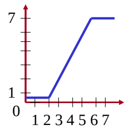
- Minimize the error: $\sum_{i=0}^{k-1} \int_{z_i}^{z_{i+1}} (q_i - z)^2 p(z)dz$
- Solution (Prove!):

$$q_i = \frac{\int_{z_i}^{z_{i+1}} z \cdot p(z)dz}{\int_{z_i}^{z_{i+1}} p(z)dz} \qquad z_i = \frac{q_{i-1} + q_i}{2}$$

Operation with LUT (4.2)

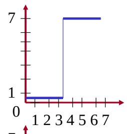
- Stretch

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	0	0	2	4	6	7	7



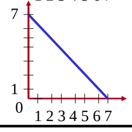
- Threshold

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	0	0	0	7	7	7	7



- Negative

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
7	6	5	4	3	2	1	0



1-D Discrete Convolution

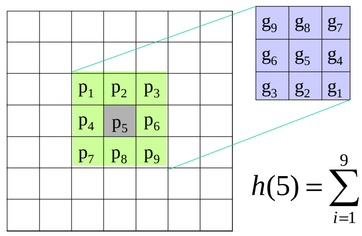
$$h(i) = (f * g)(i) = \sum_{k=1}^{n} f(k)g(i-k)$$

$f = (0 \quad 0 \quad 1 \quad 0 \quad 0)$ $\quad g = (0 \quad 0 \quad 1 \quad -1 \quad 0)$

$$h(6) = \sum_{k=1}^{5} f(k) \cdot g(6-k) = f(3) \cdot g(3) = 1$$

$$h(7) = \sum_{k=1}^{5} f(k) \cdot g(7-k) = f(3) \cdot g(4) = -1$$

2D Discrete Convolution



$$h(5) = \sum_{i=1}^{9} p(i)g(i)$$

2-D Discrete Convolution

$$h = f * g$$

$$h(i,j) = \sum_{k=1}^{n} \sum_{l=1}^{m} f(k,l)g(i-k, j-l)$$

Question: What is the complexity of convolution

Convolutions: Smoothing

Q: What is the average gray level after convolution?

Smoothing $\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ $\quad \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$

Original Image

Corrupted Image

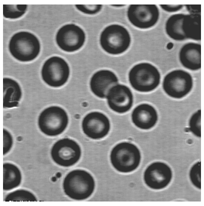
Filtered Image

Convolutions: Edge-Detection

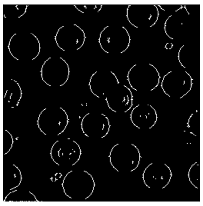
Q: What is the average gray level after convolution?

Edge Detection

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



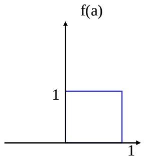
Original Blood Image

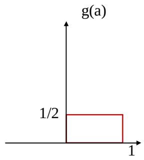


Edge Map

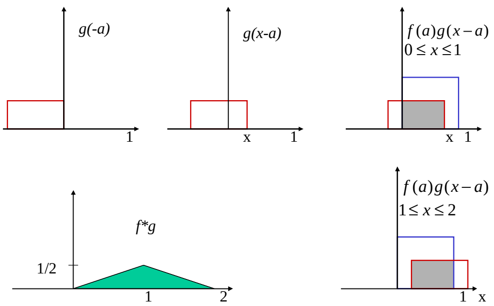
Convolution (Continuous)

$$(f * g)(x) = \int_{-\infty}^{\infty} f(a)g(x-a)da$$





$$(f * g)(x) = \int_{-\infty}^{\infty} f(a)g(x-a)da$$



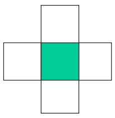
Properties of Convolution

Commutative: $f * g = g * f$

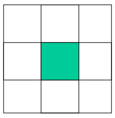
Transitive: $f * (g * h) = (f * g) * h$

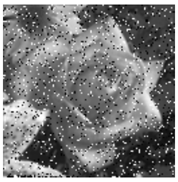
Associative: $f * (g + h) = f * g + f * h$

Median Filtering



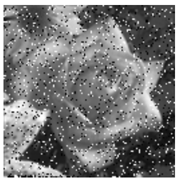
- Replace the value of a pixel with the MEDIAN of its neighborhood
- Depends on the definition of "neighborhood"

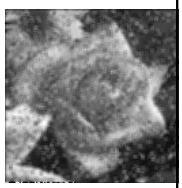




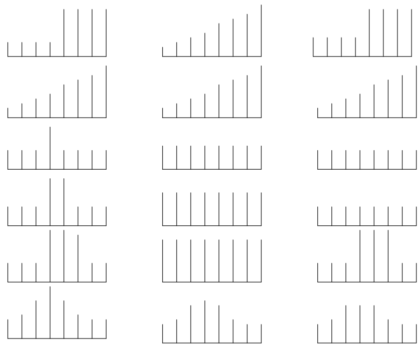
Noise Cleaning

- Averaging / Smoothing - Loss of Detail
- Median - Blockiness
- Min, Max, Min/Max, etc.





Compare: Average Median



An Approximation to Derivatives

$$\frac{\partial}{\partial x} f(i,j) \cong f(i,j) - f(i-1,j) \qquad \begin{pmatrix} 1 & -1 \end{pmatrix}$$

$$\frac{\partial}{\partial y} f(i,j) \cong f(i,j) - f(i,j-1) \qquad \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

$$\frac{\partial^2}{\partial x^2} f(i,j) \cong \frac{\partial}{\partial x} f(i+1,j) - \frac{\partial}{\partial x} f(i,j) =$$
$$= f(i-1,j) + f(i+1,j) - 2f(i,j)$$

$$\begin{pmatrix} 1 & -2 & 1 \end{pmatrix}$$

Laplacian

Equation: $\nabla^2 f = \frac{\partial}{\partial x^2} f + \frac{\partial}{\partial y^2} f$

Convolution: $\begin{pmatrix} 1 & -2 & 1 \end{pmatrix} + \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$

Subtracting the Laplacian:

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

Subtracting The Laplacian