Computer Vision IT416

Dinesh Naik.
Dept of Information Technology

Special Thanks to: Eminent Scholars of Computer Vision and Other Resource base whose slides are directly referred.

Course Description

Introduction

I. low-level vision

- Computer and Image Processing Introduction
- History of Photography
- Applications and Usage
- Concept of Dimensions
- Image Formation on Camera
- Camera Mechanism
- Concept of Pixel
- Perspective Transformation
- Concept of Bits per Pixel
- Types of Images
- Color Codes Conversion
- Gray scale to RGB Conversion
- Concept of Sampling
- Pixel Resolution
- Concept of Zooming
- Zooming methods
- Spatial Resolution
- Pixels Dots and Lines per inch
- Gray Level Resolution
- Concept of Quantization
- Concept of Dithering
- Histograms Introduction
- Brightness and Contrast
- Image Transformations
- Histogram Sliding
- Histogram Stretching
- Histogram Equalization
- Gray Level Transformations
- Concept of convolution
- Concept of Masks
- Concept of Blurring
- Concept of Edge Detection
- Prewitt Operator
- Sobel operator
- Robinson Compass Mask
- Krisch Compass Mask
- Laplacian Operator
- Fourier series and Transform
- Convolution theorem
- Linear filters
- Edges and contours
- Binary image analysis
- Background subtraction
- Texture
- Motion and optical flow

II. Mid-level vision

- Segmentation and clustering algorithms
- Hough transform
- Fitting lines and curves
- Robust fitting, RANSAC
- Deformable contours
- Interactive segmentation

III. Multiple views

- Local invariant feature detection and description
- Image transformations and alignment
- Planar homography
- Epipolar geometry and stereo
- Object instance recognition

IV. Recognition: high-level vision

- Basics of Object detection and recognition.
- Supervised classification algorithms
- Dimensionality reduction
- Deep learning, Convolutional neural networks

Reference Books

The course textbook is:

Computer Vision: Algorithms and Applications, by Rick Szeliski.

- Computer Vision: A Modern Approach, David A. Forsyth and Jean Ponce
- Computer Vision, Linda G. Shapiro and George C. Stockman
- Introductory Techniques for 3-D Computer Vision, Emanuele Trucco and Alessandro Verri.
- Multiple View Geometry in Computer Vision, Richard Hartley and Andrew Zisserman.
- Pattern classification, Richard O. Duda, Peter E. Hart, and David G. Stork
- Pattern Recognition and Machine Learning. Christopher M. Bishop
- Visual Object Recognition. K. Grauman and B. Leibe

Journals:

- IEEE-T-PAMI (Transactions on Pattern Analysis and Machine Intelligence)
- IEEE-T-IP (Transactions on Image processing)
- PR (Pattern Recognition)
- PRL (Pattern Recognition Letters)
- CVGIP (Computer Vision, Graphics & Image Processing)
- IJCV (International Journal of Computer Vision)

Evaluation Scheme

- **Mid Sem—20%**
- **End Sem—35%**
- **Assignment—45%**

Assignment Guidelines

Projects should be done GroupWise.

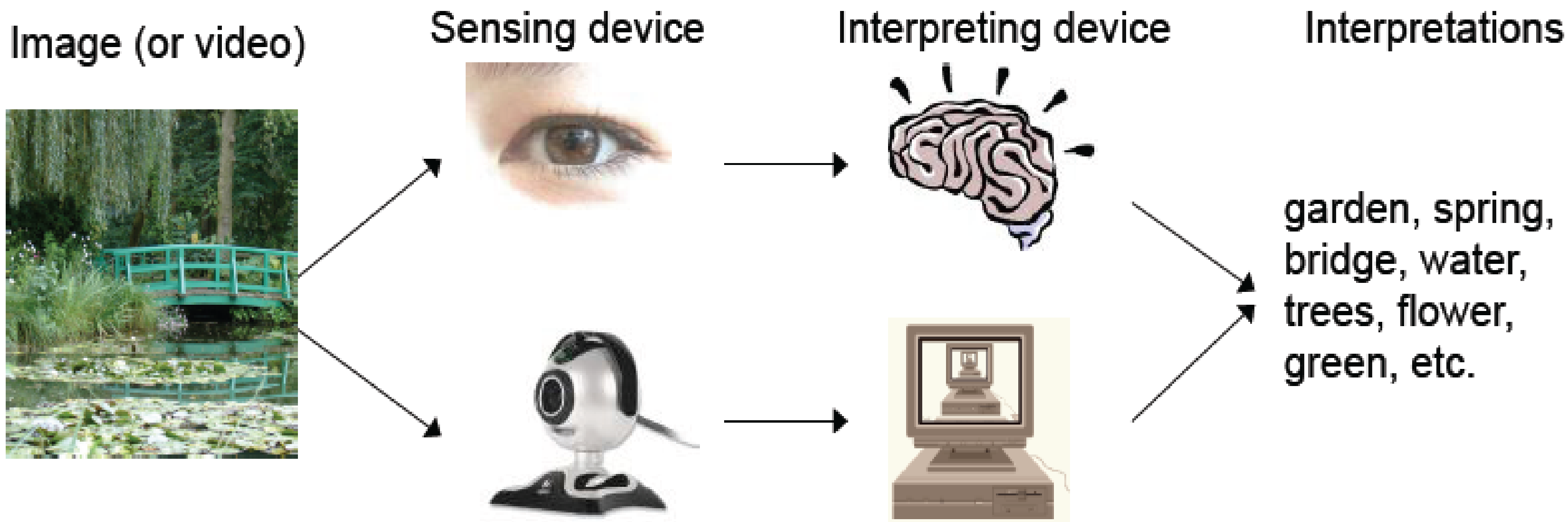
All group members should have individual unique contributions to the project (which must be stated clearly in your final report), and yet, be aware of what the other group members did. Note that this will be tested during the viva.

Your project work will be a research paper implementation[given by CI], or your own idea. In any case, if you wish to have some feedback about the triviality or difficulty of your project, please speak to me .

You will required to submit a report and appear for a viva during which you will demo your project, and answer questions about it. The final report should clearly but briefly describe the problem statement, a description of the main algorithm(s) you implemented, a description of the datasets on which they were tested, a detailed description of the results followed by a conclusion including an analysis of the good and bad aspects of your implementation or the algorithm.

You may use MATLAB/C/C++/Java/Python + any packages (OpenCV,ITK, etc) for your project. But merely invoking calls to someone's else's software is not substance enough. You should have your own non-trivial coding component. If software for the research paper you implement is already available, you should use it only for comparison sake - you will be expected to implement the paper on your own. Please discuss with me if you need any clarifications for your specific case.

What is (computer) vision?



The goal of computer vision

- To bridge the gap between pixels and “meaning”

What we see

0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

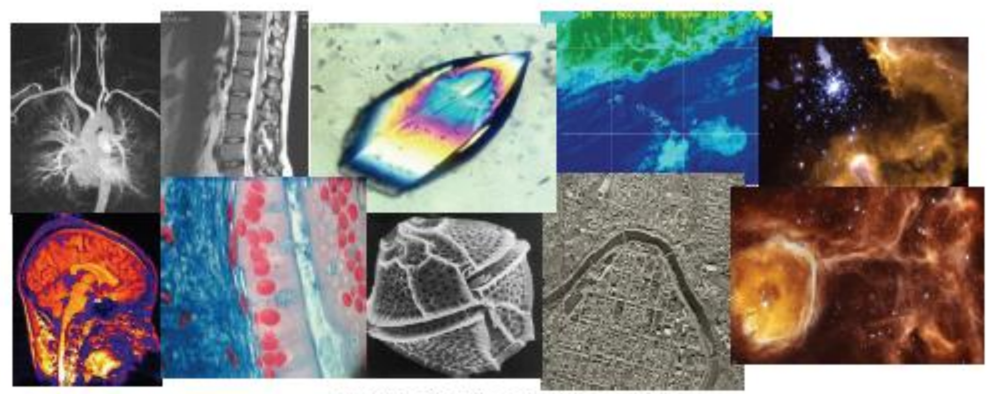
What a computer sees

Why study computer vision?

- Vision is useful: Images and video are everywhere!

Google Image Search | Picasa | flickr | webshots | picsearch | YouTube Broadcast Yourself

Surveillance and security



Medical and scientific images

To do Research on Computer vision areas.

- Inherently **interdisciplinary** subject: numerous application areas - remote sensing, photography, visual psychology, archaeology, surveillance, etc.
- Fast becoming a popular field of study in India: scope for R&D work in numerous research labs (In India: Samsung, GE, Phillips, Siemens, Microsoft, HP, TI, Google; DRDO, ICRISAT, ISRO, etc.)

What is vision?

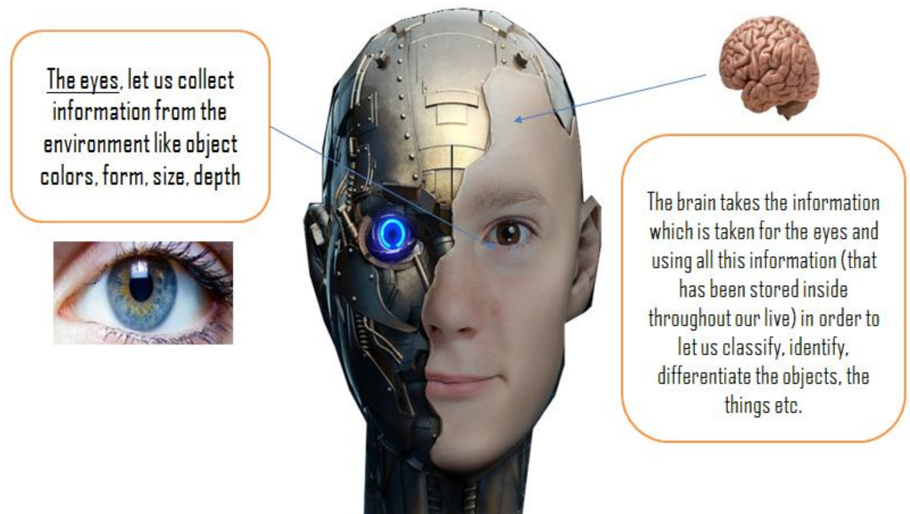
- Recognize objects
  - people we know
  - things we own
- Locate objects in space
  - to pick them up
- Track objects in motion
  - catching a baseball
  - avoiding collisions with cars on the road
- Recognize actions
  - walking, running, pushing

What is vision?

- What does it mean to see ?

“ The plain man’s answer (and Aristotle’s too) would be, to know what is where by looking. In other words, vision is the process of discovering from images what is present in the world, and where it is ” David Marr, Vision 1982

Human vision System

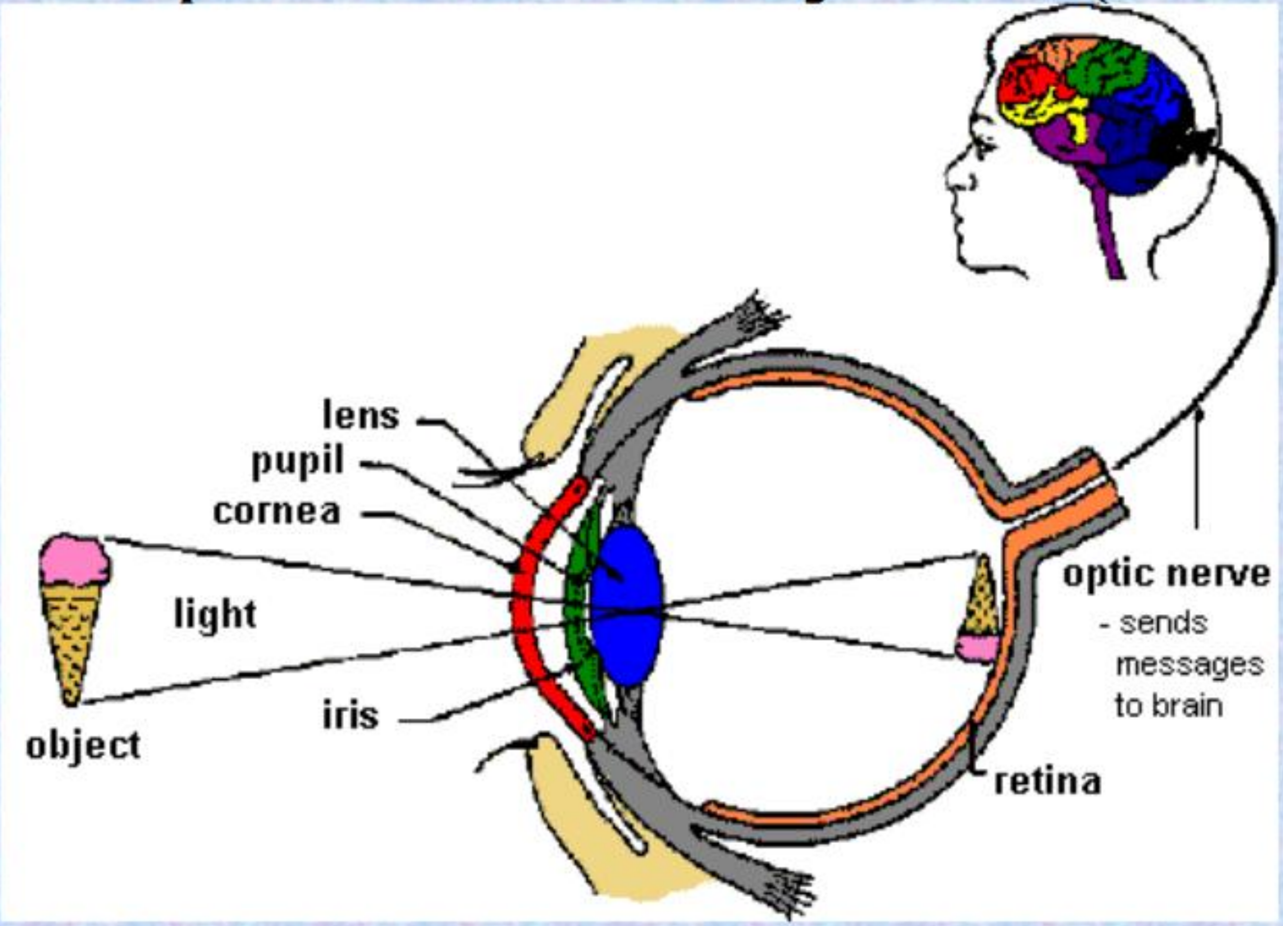


What is Computer Vision?

- Make computers understand images and videos.

- What kind of scene?
- Where are the cars?
- How far is the building?

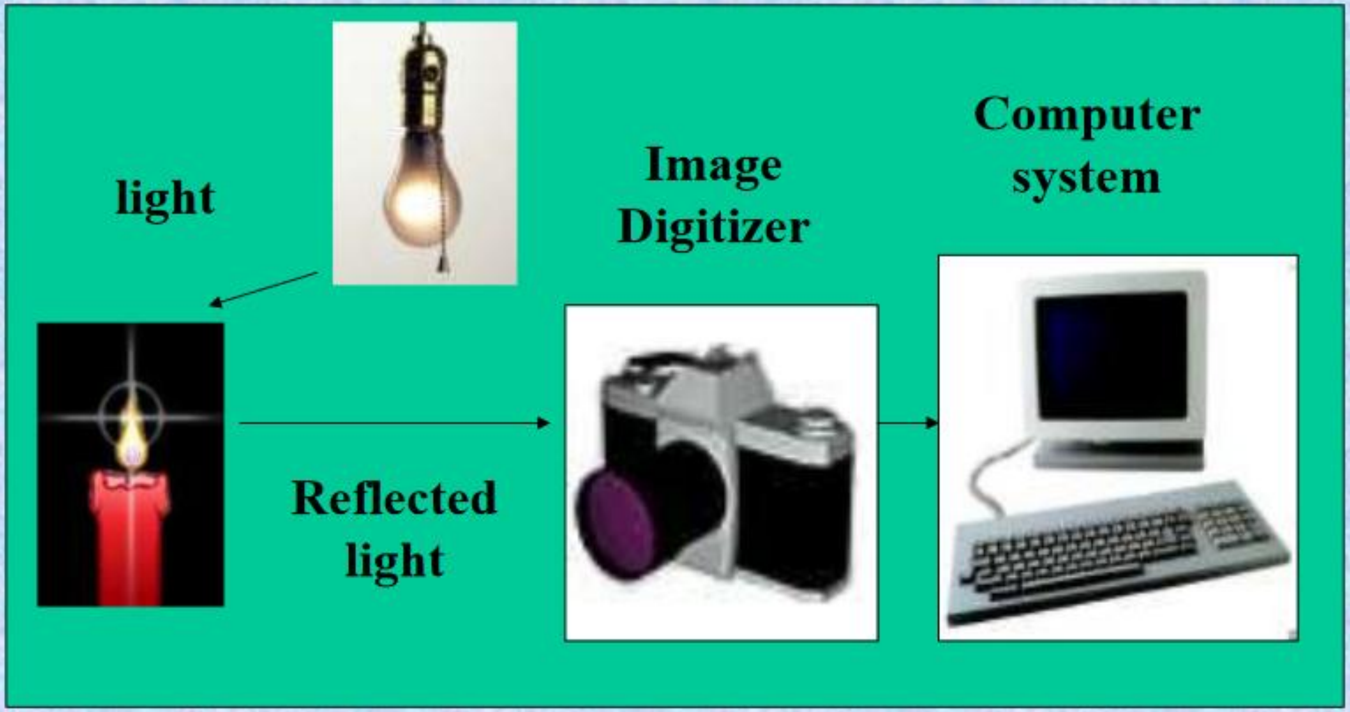
Human Vision System (HVS) Vs. Computer Vision System (CVS)



The Optics of the eye

http://www.cse.iitm.ac.in/~vplab/courses/CV_DIP/PDF/INTRO_CV.pdf

A computer Vision System (CVS)



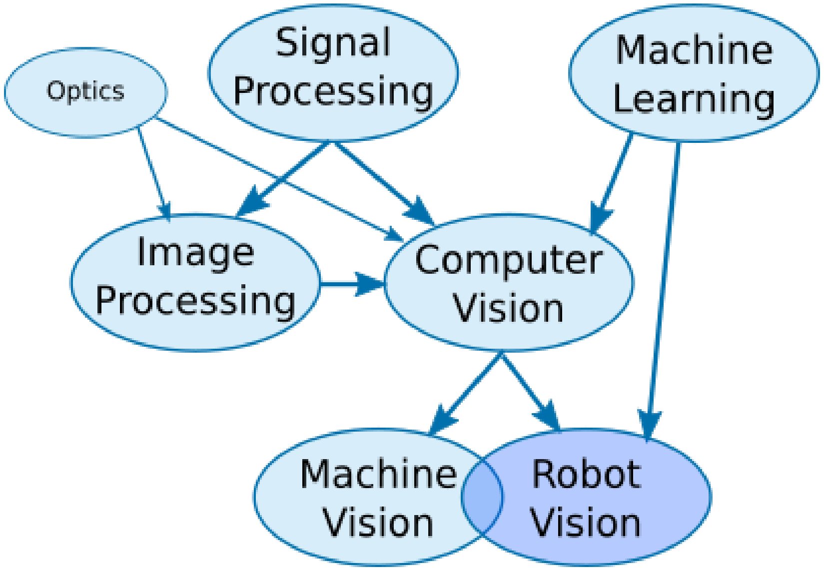
http://www.cse.iitm.ac.in/~vplab/courses/CV_DIP/PDF/INTRO_CV.pdf

What's the difference between

- Robot Vision,
- Computer Vision,
- Image Processing,
- Machine Vision
- And Pattern Recognition.

What is Robot Vision?

- Robot Vision involves using a combination of camera hardware and computer algorithms to allow robots to process visual data from the world.
- Robot Vision is closely related to Machine Vision
- They are both [closely related to Computer Vision].



Signal Processing involves processing electronic signals to either clean them up (e.g. removing noise), extract information, prepare them to output to a display or prepare them for further processing

Machine Vision

- Machine Vision refers to the industrial use of vision for automatic inspection, process control and robot guidance.
- Machine Vision is an engineering domain.
- In some ways, you could think of it as a child of Computer Vision.

Technique	Input	Output
Signal Processing	Electrical signals	Electrical signals
Image Processing	Images	Images
Computer Vision	Images	Information/features
Pattern Recognition/Machine Learning	Information/features	Information
Machine Vision	Images	Information
Robot Vision	Images	Physical Action

Computer Vision and Image Processing: What's the difference?

- Difference is **blurry**
- "Image processing" typically involves processing/analysis of (2D) images without referring to underlying 3D structure
- Computer vision – typically involves inference of **underlying 3D structure** from 2D images
- Many computer vision techniques also aim to infer properties of the scene directly – without 3D reconstruction.
- Computer vision – direct opposite of computer graphics

•In a way, Computer Vision can be considered the inversion of Computer Graphics.

•A computer graphics systems receives as its input the formal description of a visual scene, and its output is a visualization of that scene.

•A computer vision system receives as its input a visual scene, and its output is a formal description of that scene with regard to the system's task.

•Unfortunately, while a computer graphics task only allows one solution, computer vision tasks are often ambiguous, and it is unclear what the correct output should be.

Image Processing vs Computer Vision

- Computer Vision and Image Processing are like cousins, but they have quite different aims

Image Processing techniques are primarily used to improve the quality of an image, convert it into another format (like a histogram) or otherwise change it for further processing

- Computer Vision, on the other hand, is more about extracting information from images to make sense of them.
- we see that both of these domains are heavily influenced by the domain of Physics, specifically Optics

- If we include Pattern Recognition into the family tree, or more broadly Machine Learning. This branch of the family is focused on recognizing patterns in data, which is quite important for many of the more advanced functions required of Robot vision.

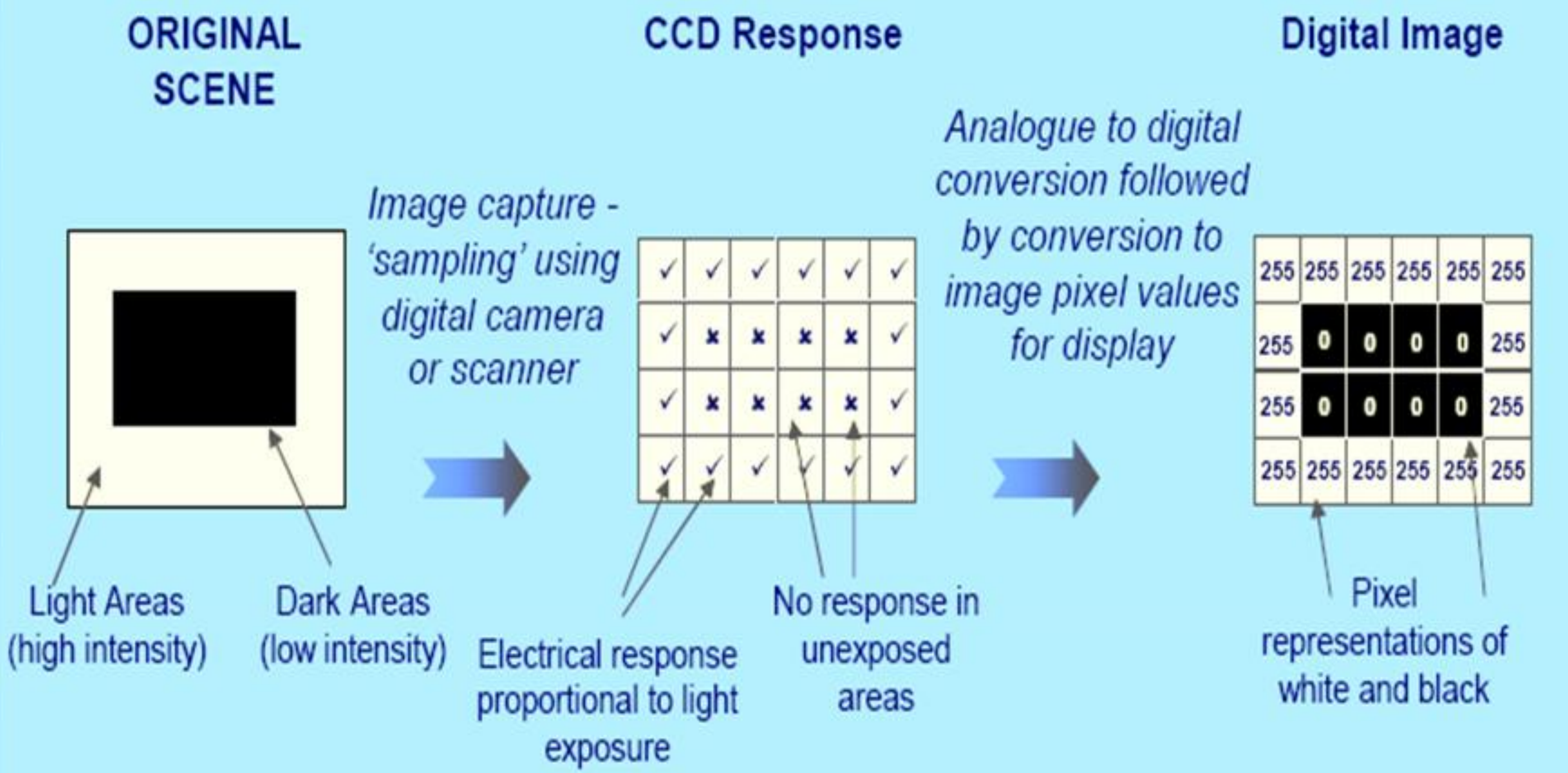
Digital Image processing **is in many cases concerned with taking one array of pixels as input and producing another array of pixels as output which in some way represents an improvement to the original array.**

Purpose:

1. **Improvement of Pictorial Information**
   - **improve the contrast of the image,**
   - **remove noise,**
   - **remove blurring caused by movement of the camera during image acquisition,**
   - **it may correct for geometrical distortions caused by the lens.**

2. **Automatic Machine perception (termed Computer Vision, Pattern Recognition or Visual Perception) for intelligent interpretation of scenes or pictures.**

Image acquisition using a CCD camera



A ***digital image*** is a two-dimensional (3-D image is called range data) array of intensity values, f(x, y), which represents 2-D intensity function discretized both in spatial coordinates (**spatial sampling**) and brightness (**quantization**) values.

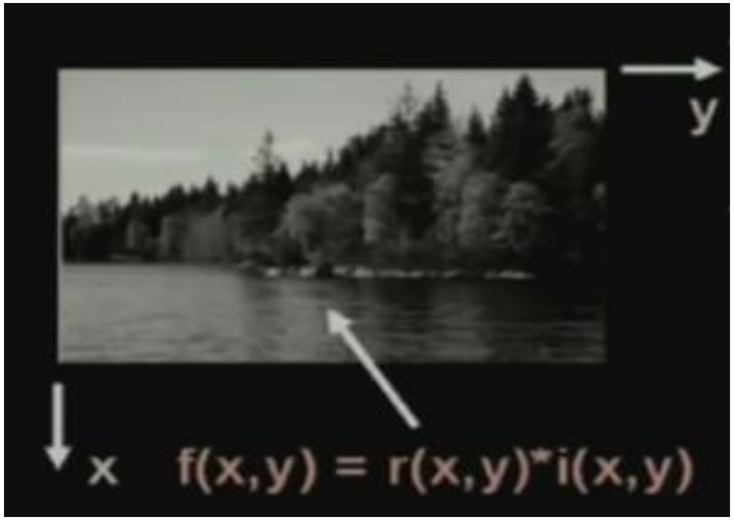
The elements of such an array are called **pixels** (picture elements).

The storage requirement for an image depends on the **spatial resolution** and number of bits necessary for **pixel quantization**.

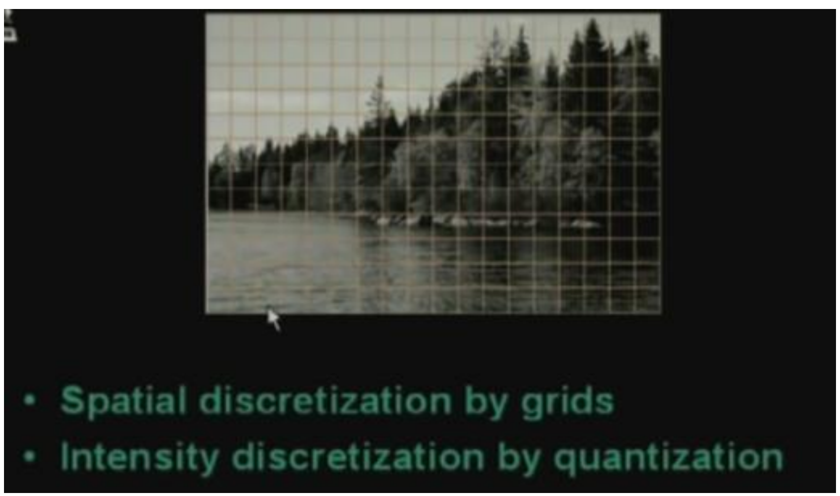
The processing of an image depends on the application domain and the methodology used to solve a problem. There exists four broad categories of tasks in digital image processing:

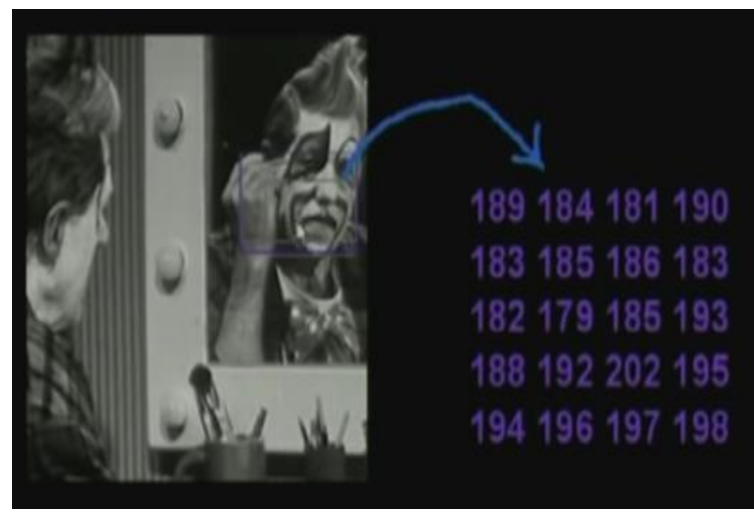
(i) Compression, (ii) Segmentation,
(iii) Recognition and (iv) motion.

Image Representation

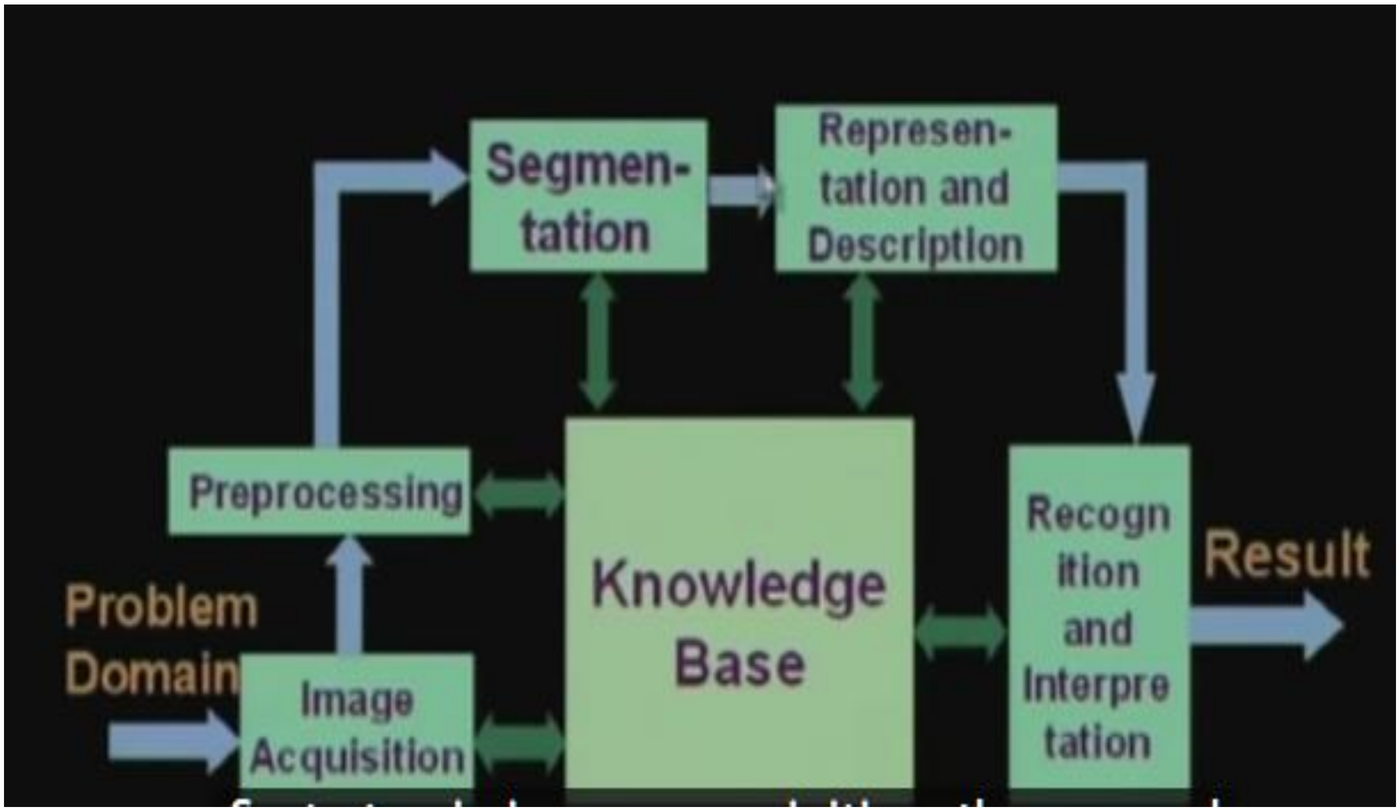


- An image is a 2-D light intensity function f(x,y)
- A digital image f(x,y) is discretized both in spatial coordinates and brightness
- It can be considered as a matrix whose row, column indices specify a point in the image and the element value identifies gray level value at that point





Steps in Digital Image processing



What is digitization

An image to be represented in the form of a finite 2-D matrix

$$\begin{bmatrix} f(0,0) & f(0,1) & f(0,2)\ldots\ldots\ldots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2)\ldots\ldots\ldots & f(1,N-1) \\ f(2,0) & f(2,1) & f(2,2)\ldots\ldots\ldots & f(2,N-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M,1) & f(M,2)\ldots & f(M-1,N-1) \end{bmatrix}$$

Image representation by 2-D finite matrix – Sampling

Each Matrix element represented by one of the finite set of discrete values- Quantization

There are generally three main categories of tasks involved in a complete computer vision system. They are:

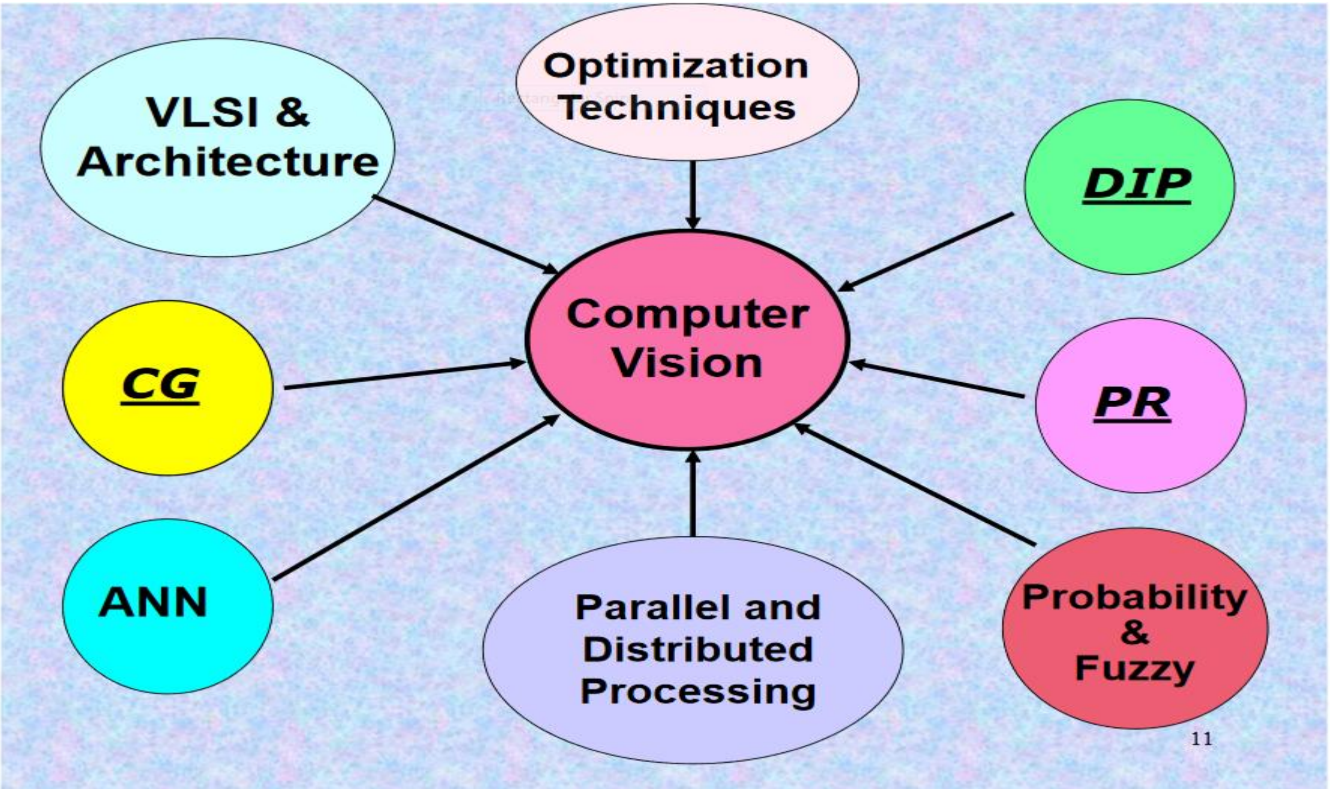
• *Low level processing*: Involves image processing tasks in which the quality of the image is improved for the benefit of human observers and higher level routines to perform better.

• *Intermediate level processing*: Involves the processes of feature extraction and pattern detection tasks. The algorithms used here are chosen and tuned in a manner as may be required to assist the final tasks of high level vision.

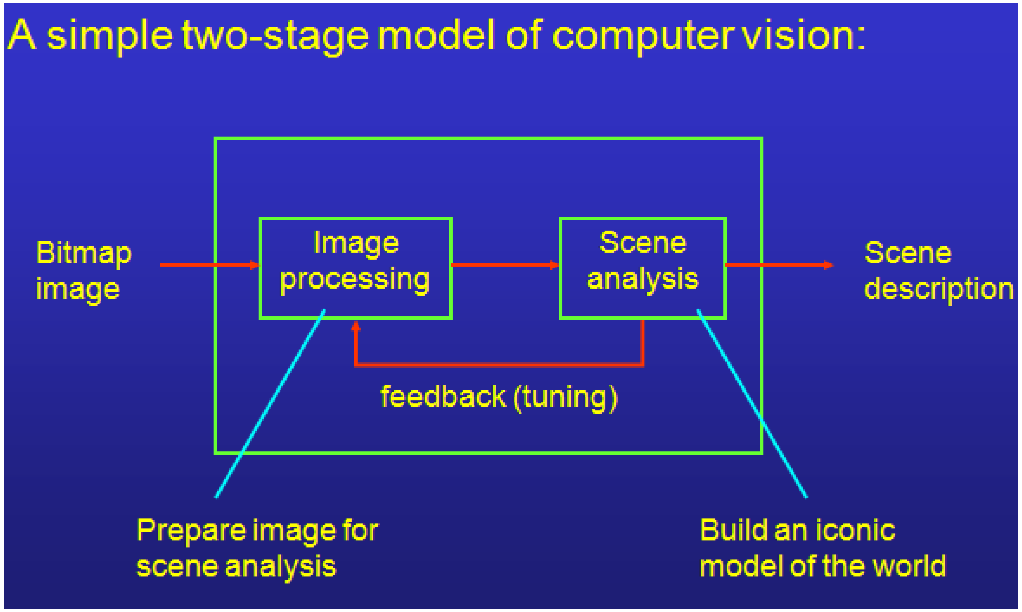
• *High level vision*: Involves autonomous interpretation of scenes for pattern classification, recognition and identification of objects in the scenes as well as any other information required for human understanding.

A top down approach, rather than a bottom-up approach is used in the design of these systems in many applications. The methods used to solve a problem in digital image processing depends on the application domain and nature of data being analyzed.

Computer Vision is an area of work, which is a combination of concepts, techniques and ideas from Digital Image Processing, Pattern Recognition, Artificial Intelligence and Computer Graphics



http://www.cse.iitm.ac.in/~vplab/courses/CV_DIP/PDF/INTRO_CV.pdf



- A simple two-stage model of computer vision:

The **image processing** stage **prepares** the input image for the subsequent scene analysis.

Usually, image processing results in one or more **new images** that contain specific information on relevant features of the input image.

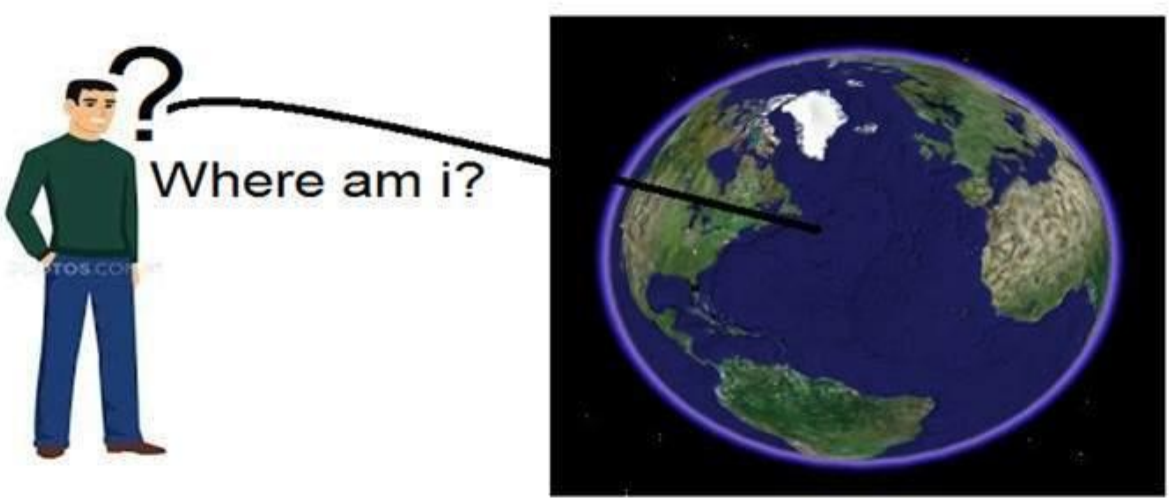
The information in the output images is **arranged in the same way** as in the input image. For example, in the upper left corner in the output images we find information about the upper left corner in the input image.

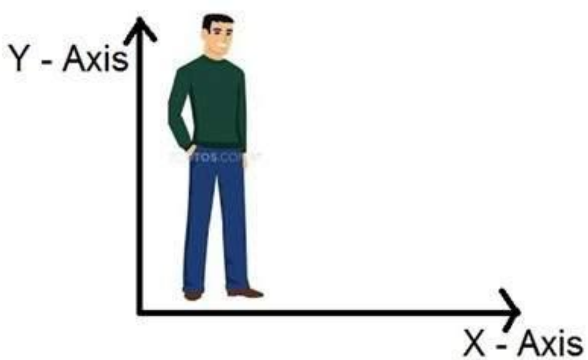
The **scene analysis** stage interprets the results from the image processing stage.

Its output completely depends on the problem that the computer vision system is supposed to solve.

For example, it could be the **number of bacteria** in a microscopic image, or the **identity of a person** whose retinal scan was input to the system.

Concept of Dimensions





History of Computer Vision

Marvin Minsky, MIT
Turing award, 1969

"In 1966, Minsky hired a first-year undergraduate student and assigned him a problem to solve over the summer:

connect a camera to a computer and get the machine to describe what it sees."

Crevier 1993, pg. 88

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Group July 7, 1966
Vision Memo. No. 100.

Half a century late
we're still working

THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

1960's: interpretation of synthetic worlds

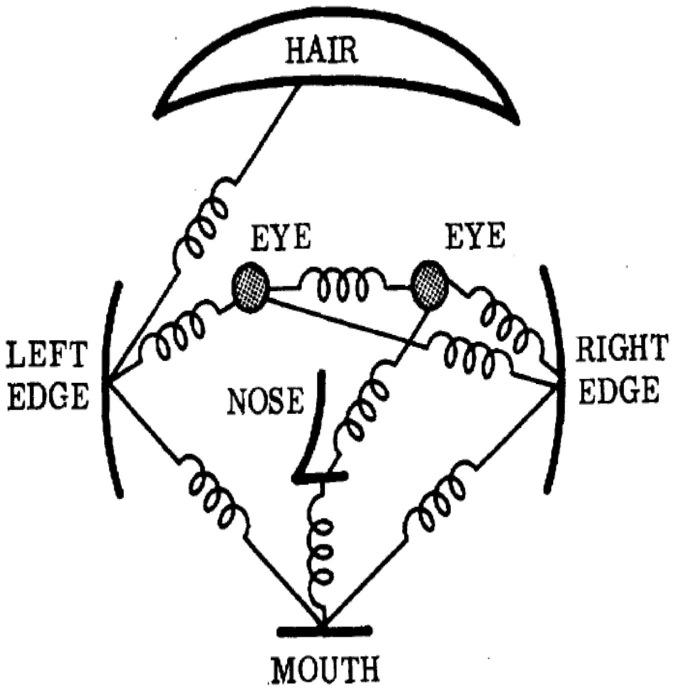
Larry Roberts
"Father of Computer Vision"

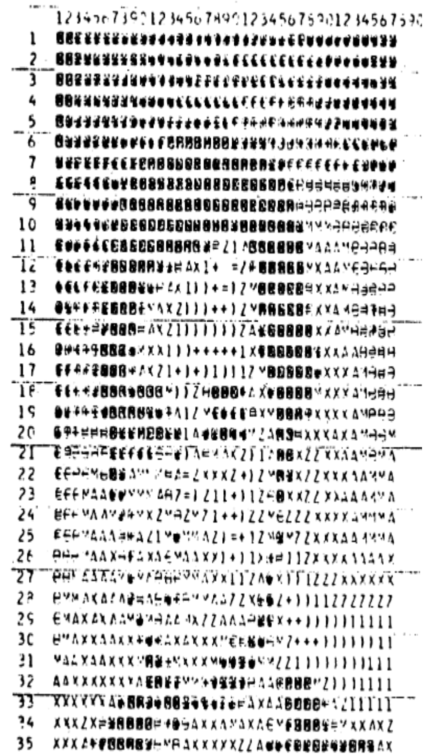
Larry Roberts PhD Thesis, MIT, 1963,
Machine Perception of Three-Dimensional Solids

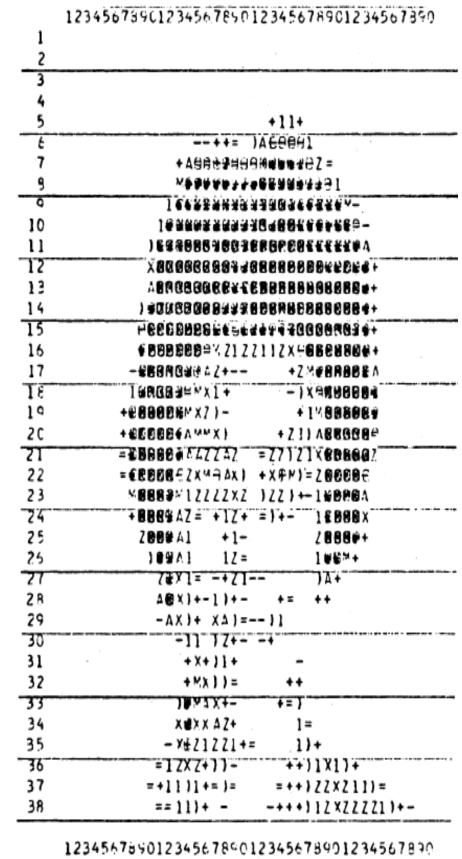
Extracting 3D information about solid objects from 2D photographs of line drawings.

Slide credit: Steve Seitz

1970's: some progress on interpreting selected images

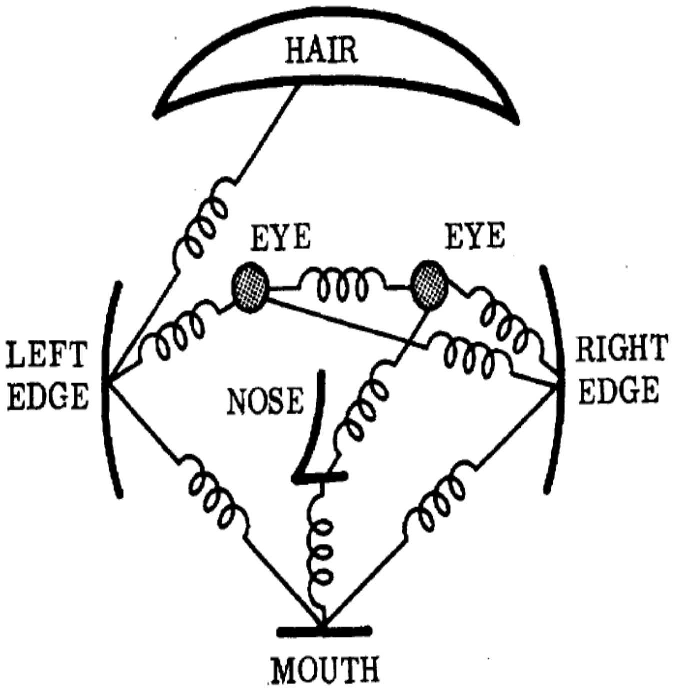


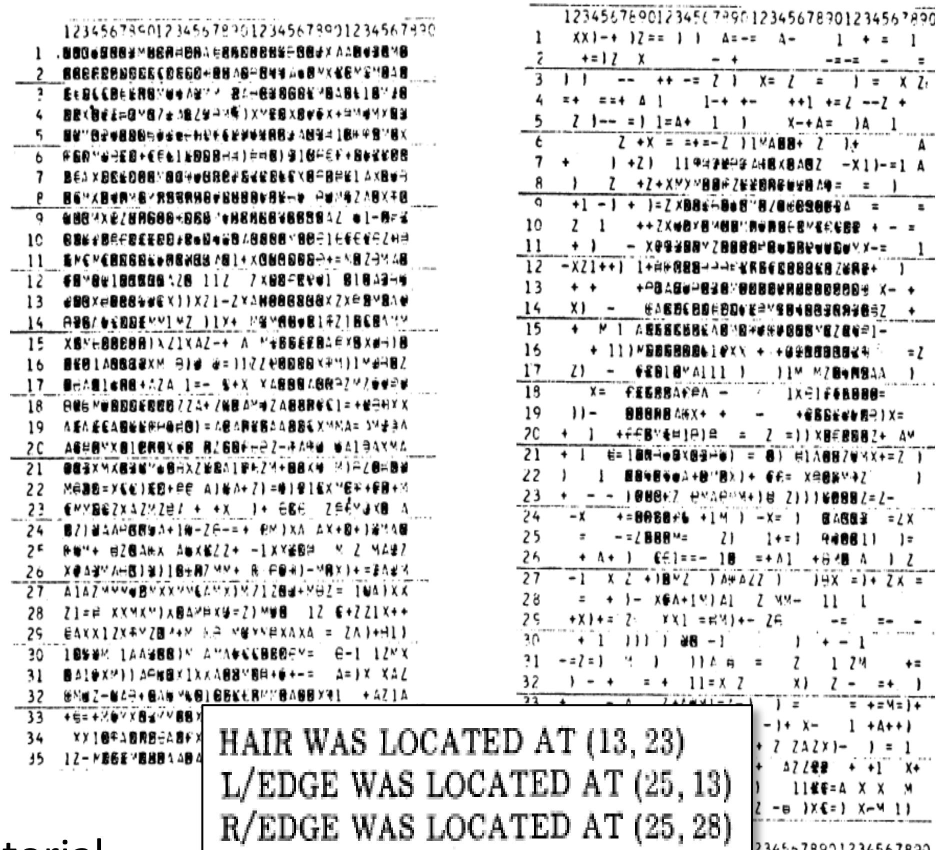




The representation and matching of pictorial structures
Fischler and Elschlager, 1973

1970's: some progress on interpreting selected images



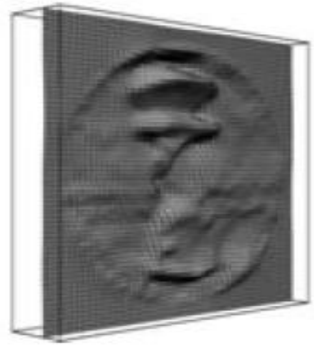


HAIR WAS LOCATED AT (13, 23)
L/EDGE WAS LOCATED AT (25, 13)
R/EDGE WAS LOCATED AT (25, 28)
L/EYE WAS LOCATED AT (22, 16)
R/EYE WAS LOCATED AT (22, 23)
NOSE WAS LOCATED AT (27, 20)
MOUTH WAS LOCATED AT (29, 19)

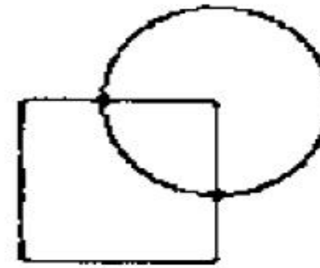
The representation and matching of pictorial structures
Fischler and Elschlager, 1973

1980's: ANNs come and go; shift toward geometry and increased mathematical rigor

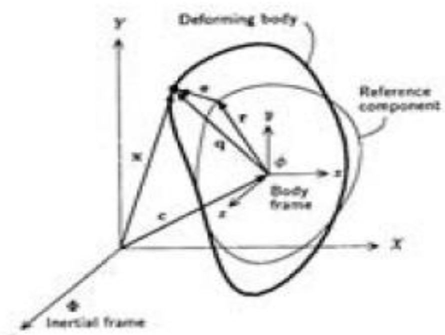
(a)



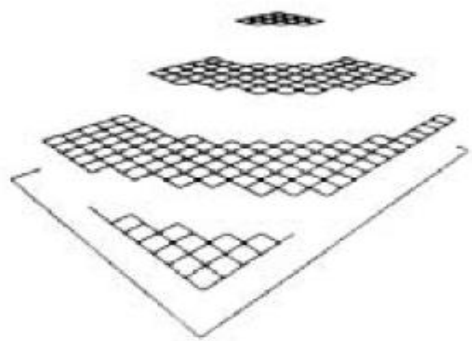
(b)



(c)



(d)



(e)

(f)

Image credit: Rick Szeliski

1990's: face recognition

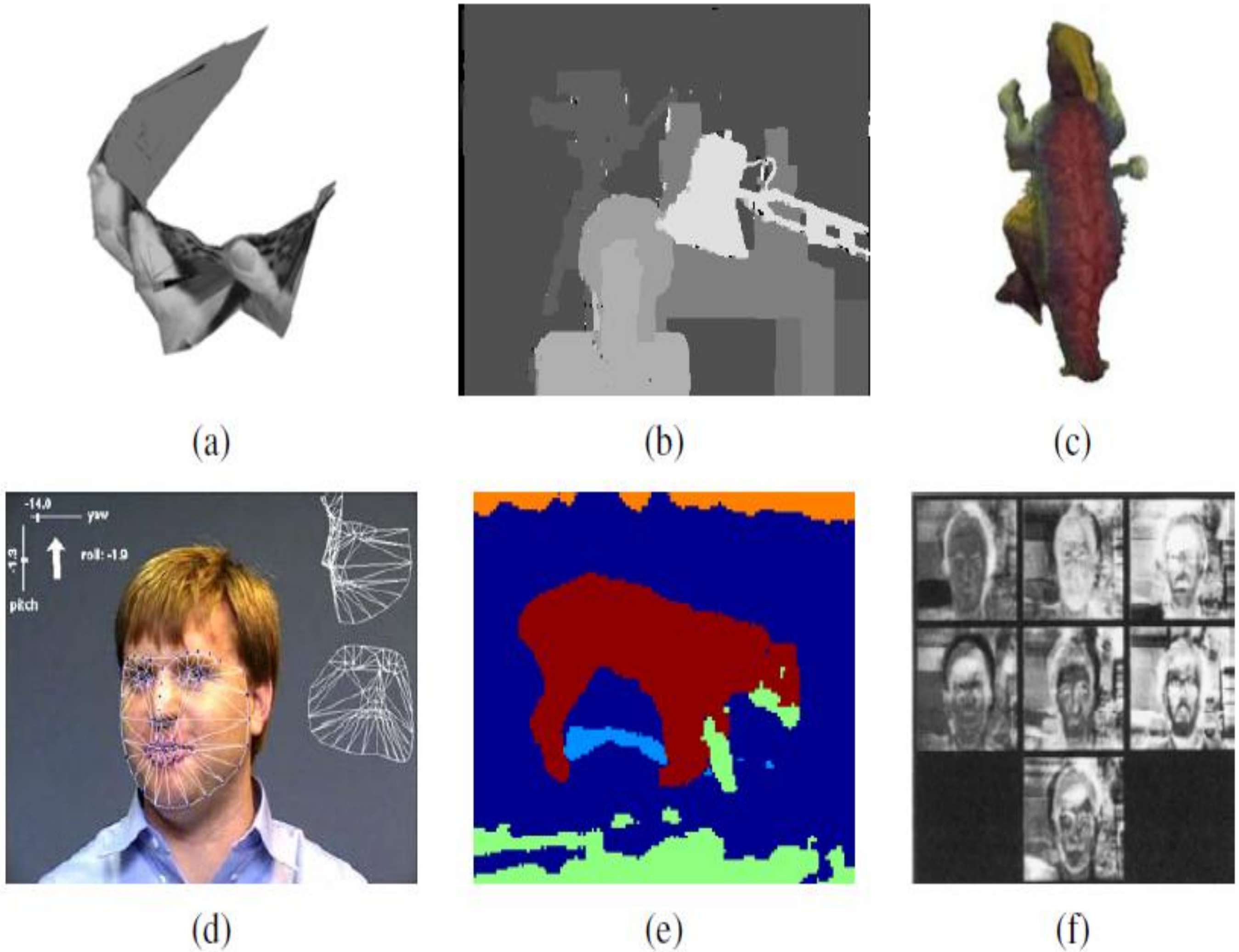


Image credit: Rick Szeliski

2000's: broader recognition; large annotated datasets available; video processing starts

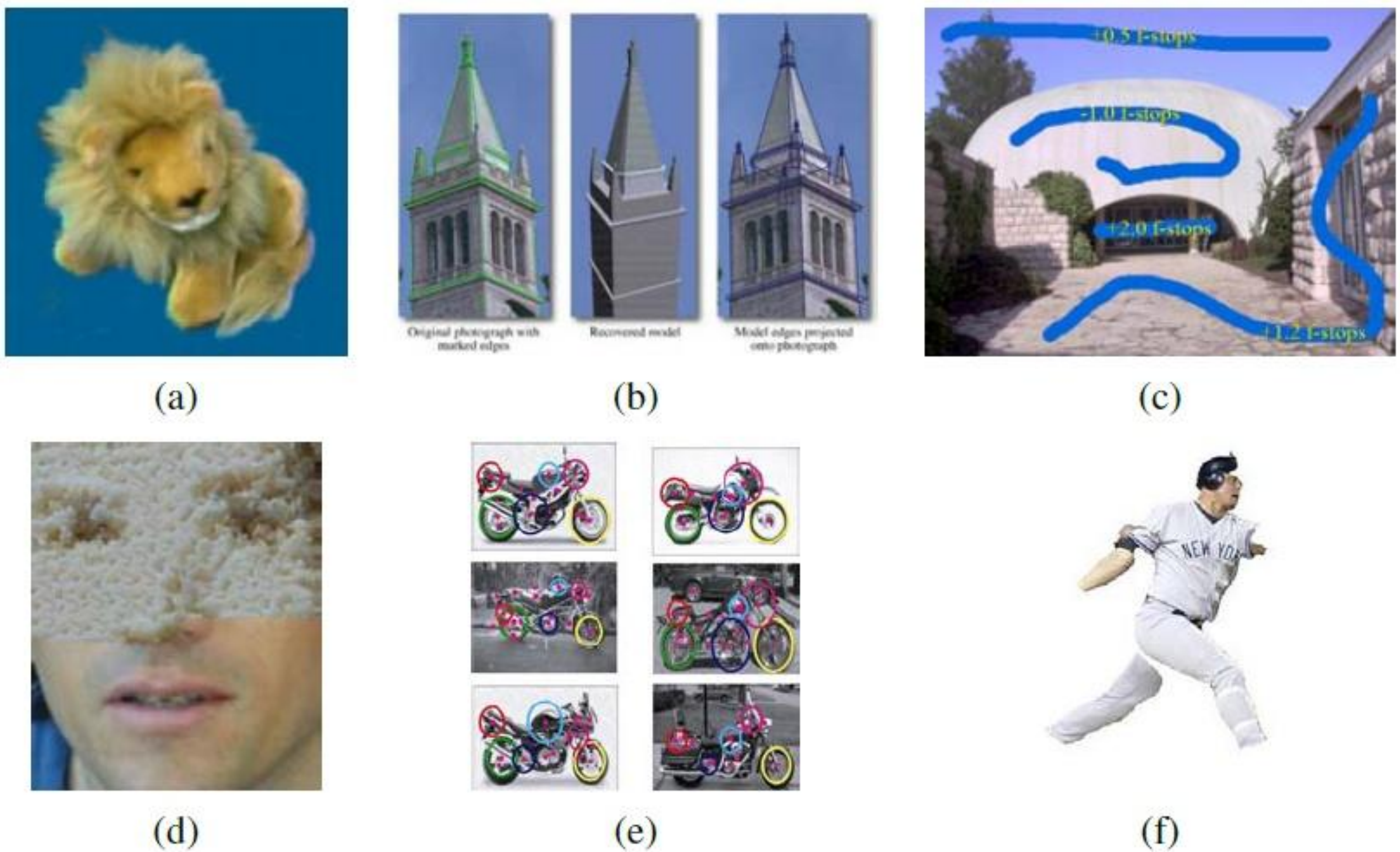
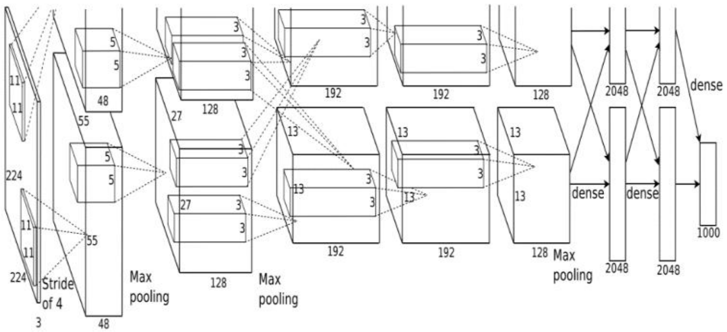
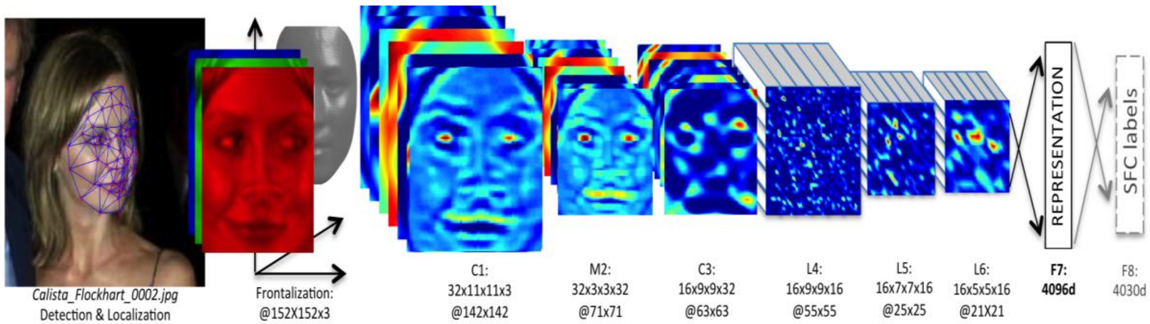


Image credit: Rick Szeliski

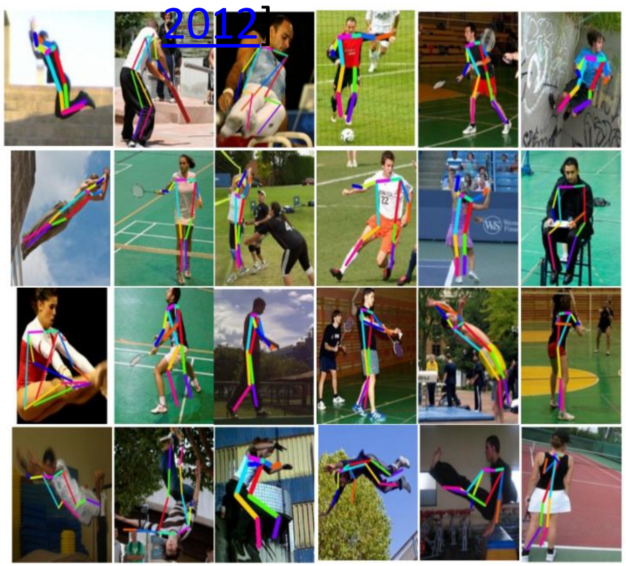
2010's: resurgence of deep learning



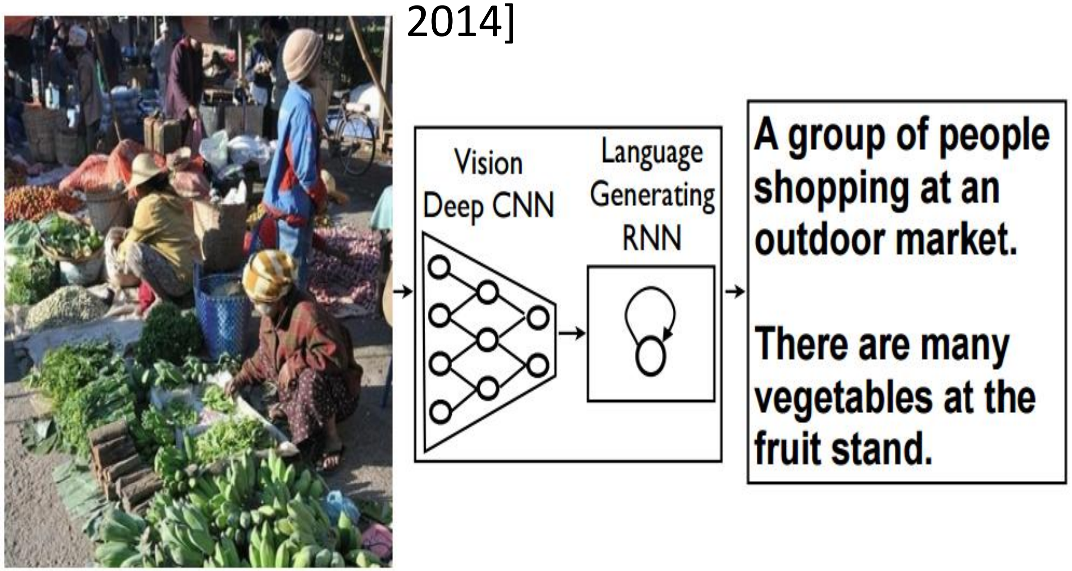
[AlexNet NIPS 2012]



[DeepFace CVPR 2014]



[DeepPose CVPR 2014]



[Show, Attend and Tell ICML 2015]

2020's: autonomous vehicles

2030's: robot uprising?

Different fields of applications include:

- **Character Recognition,**
- **Document processing,**
- **Commercial (signature & seal verification) application,**
- ***Biometry and Forensic* *(authentication: recognition and verification of persons using face, palm & fingerprint),***
- **Pose and gesture identification,**
- **Automatic inspection of industrial products,**
- **Industrial process monitoring,**
- **Biomedical Engg. (Diagnosis and surgery),**
- ***Military surveillance and target identification,***
- **Navigation and mobility (for robots and unmanned vehicles - land, air and underwater),**
- ***Remote sensing (using satellite imagery),***
- ***GIS***
- **Safety and security (night vision),**
- **Traffic monitoring,**
- ***Sports (training and incident analysis)***
- **VLDB (organization and retrieval)**
- **Entertainment and virtual reality.**

TARGETED INDUSTRIAL APPLICATIONS

Intelligent Traffic Control

Anti-forging Stamps

Card Counting Systems

Drive Quality Test

Camera Flame Detection

CCTV Fog Penetration

Key Image Search/Index

Security Monitoring

Robust Shadow Detection

Vehicle Segmentation

Visual Tracking Systems

Illegal content (adult) Filter

Scratch Detection

Smart Traffic Monitoring

Vehicle Categorization

Vehicle Wheel alignment

Number Plate Identification

Referrals for Line calls

Different categories of work being done in CV, to solve problems:

2-D image analysis –
segmentation, target detection,
matching, CBIR;

3-D multi-camera calibration;
Correspondence and stereo;
Reconstruction of
3-D Objects and surfaces;

Pattern Recognition
for Objects, scenes;

Video and motion analysis;
Video analytics; CBVR;
Compression;

Feature extraction:
Canny, GHT, Snakes,
DWT, Corners,
SIFT, GLOH, LESH;

Multi-sensor data,
Decision and feature fusion;

Image and Video-based
Rendering;

Steganography and
Watermarking;

The various sub-categories of technology in these related fields are:

- *image enhancement,*
- *image restoration and filtering,*
- *representation and description,*
- *feature extraction,*
- *image segmentation,*
- *image matching,*
- color image processing,
- image synthesis,
- image representation,

- *image reconstruction*
- *range data processing,*
- *stereo image processing*
- *computational geometry,*
- *image morphology,*
- *artificial neural networks,*
- *Neuro-fuzzy techniques,*
- *computational geometry,*
- *parallel architectures & algorithms.*

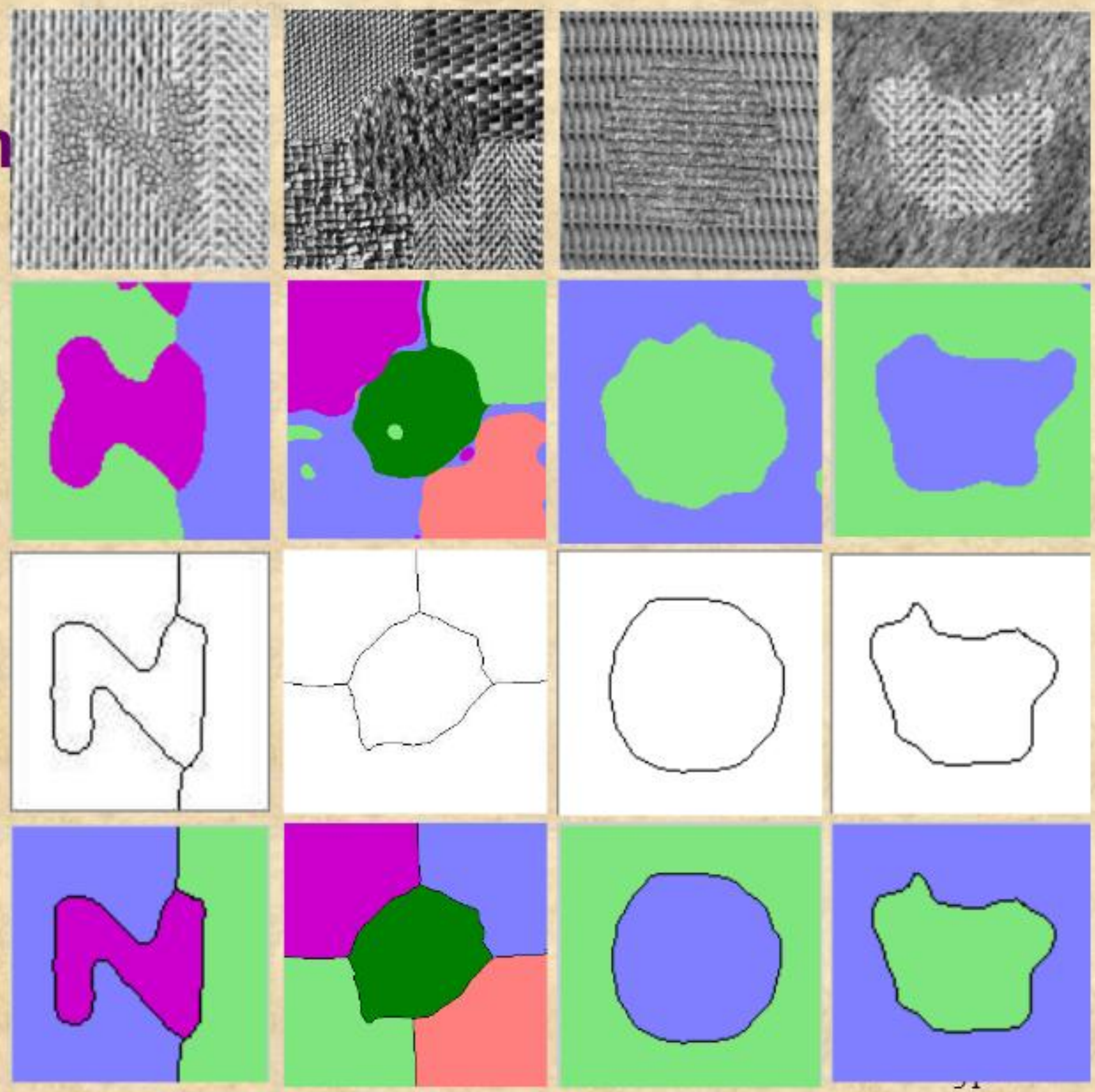
Results of Segmentation

Input Image

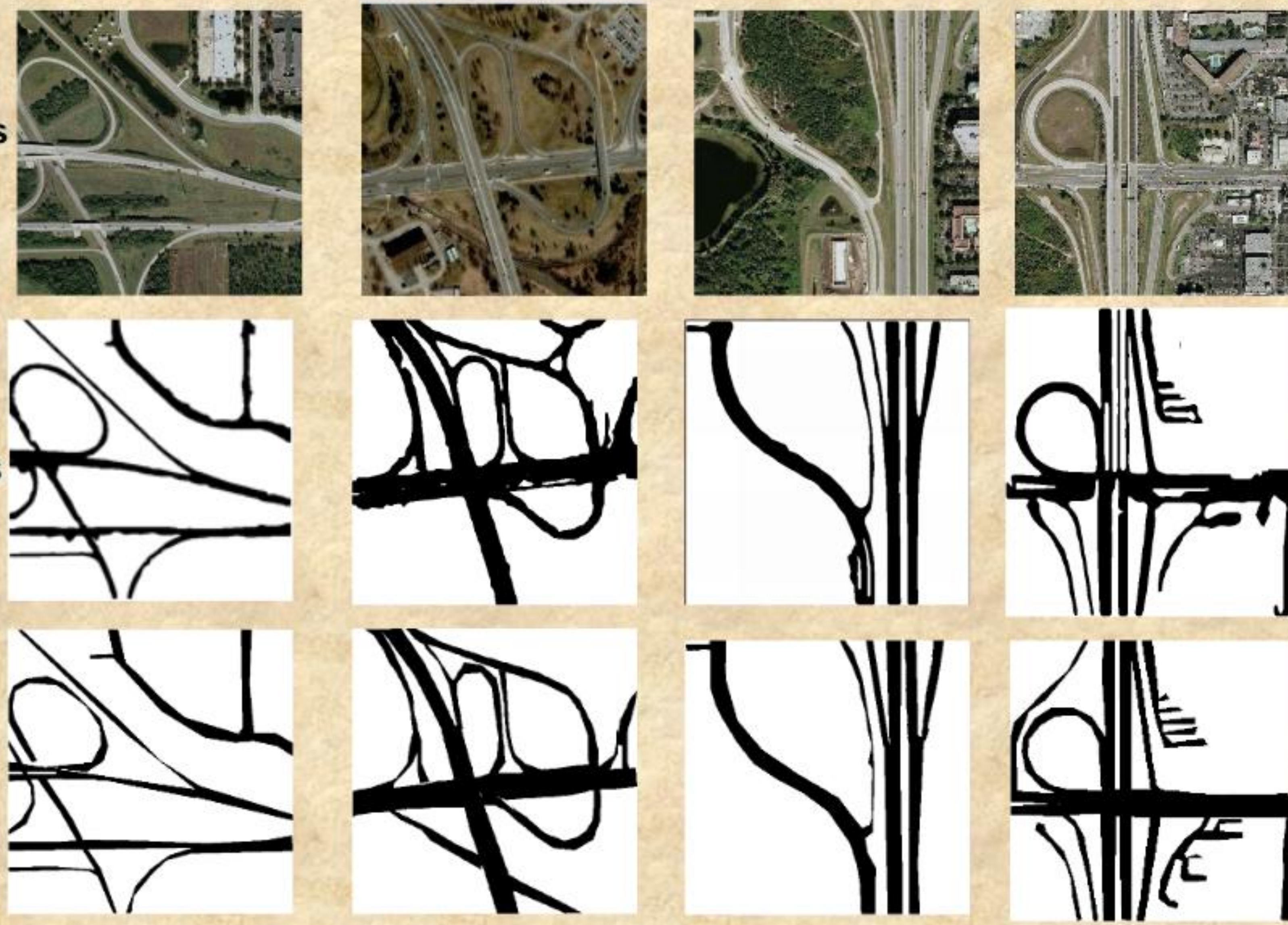
Segmented map before integration

Edge map before integration

Segmented map and Edge map after integration



Road extraction from Satellite Images



SAT Images

Results

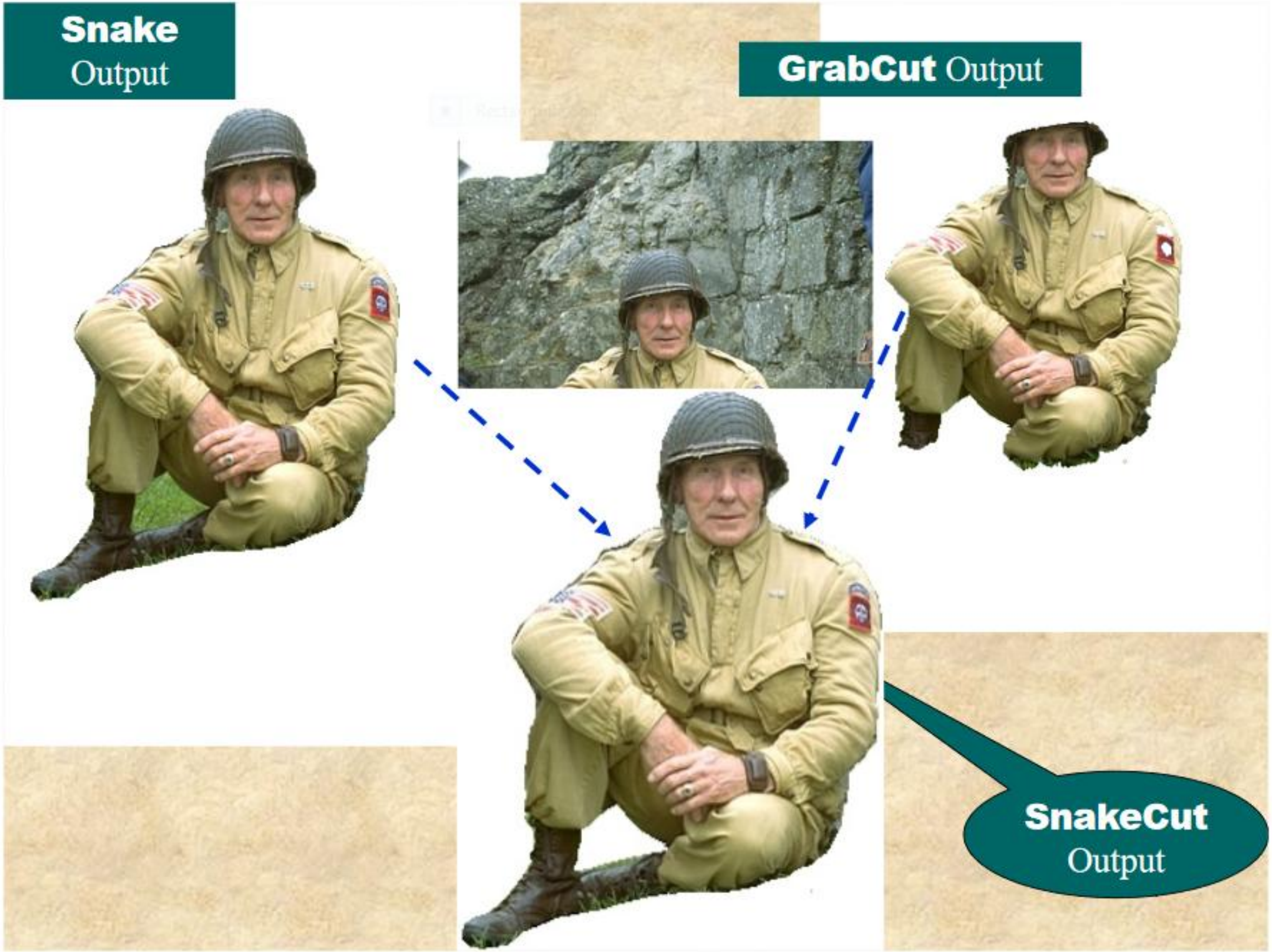
Hand-drawn

SNAKECUT – Extraction of a Foreground Object with holes

Our proposed approach for segmentation of an object with a hole, using a combination of (i) Active Contour and (ii) GrabCut

- **Here, objective is to crop the soldier from the input image**

- **Cropped image should not contain any part of the background**


Snake
Output
GrabCut Output
SnakeCut
Output

Object Extraction From an Image

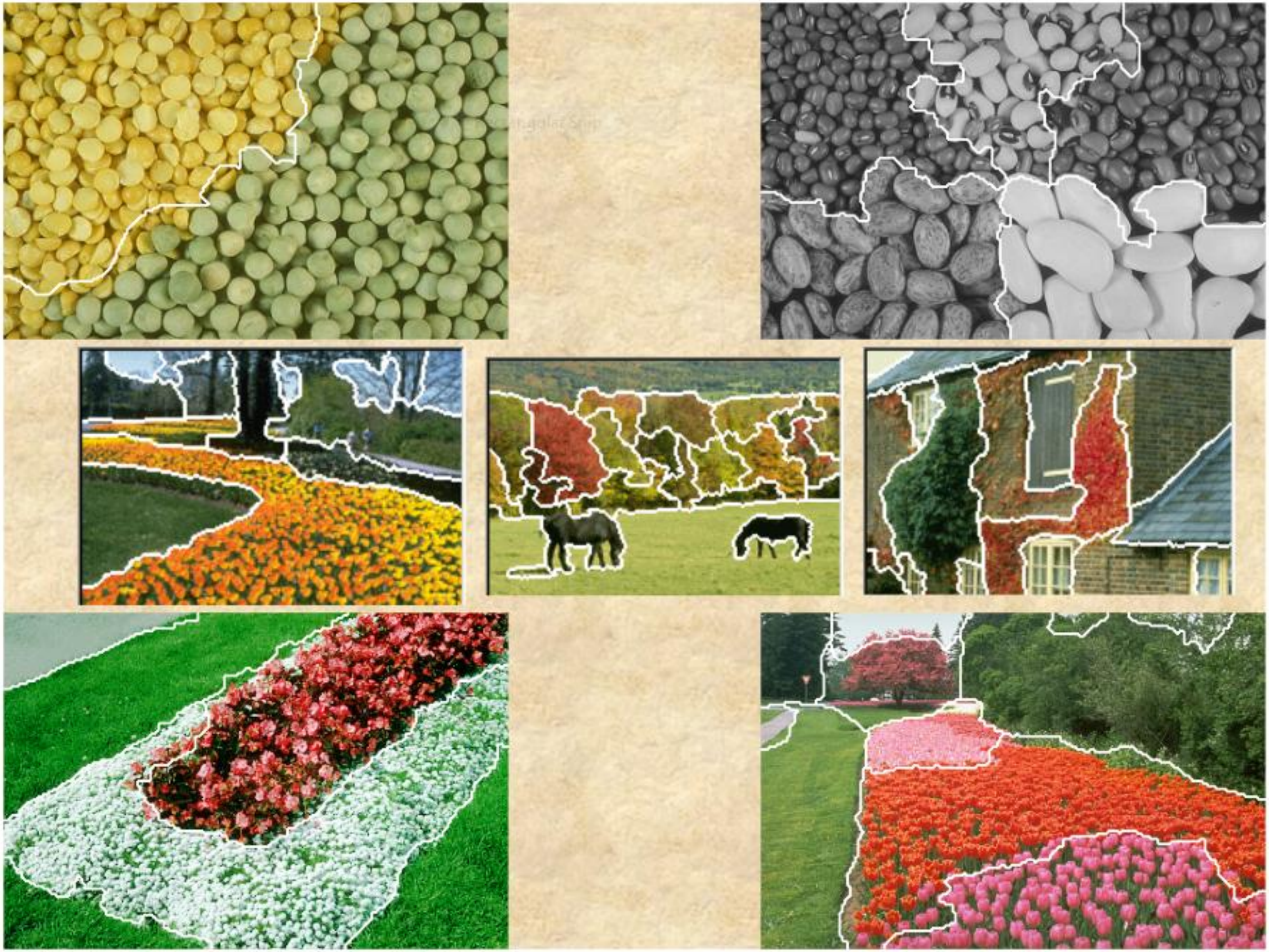
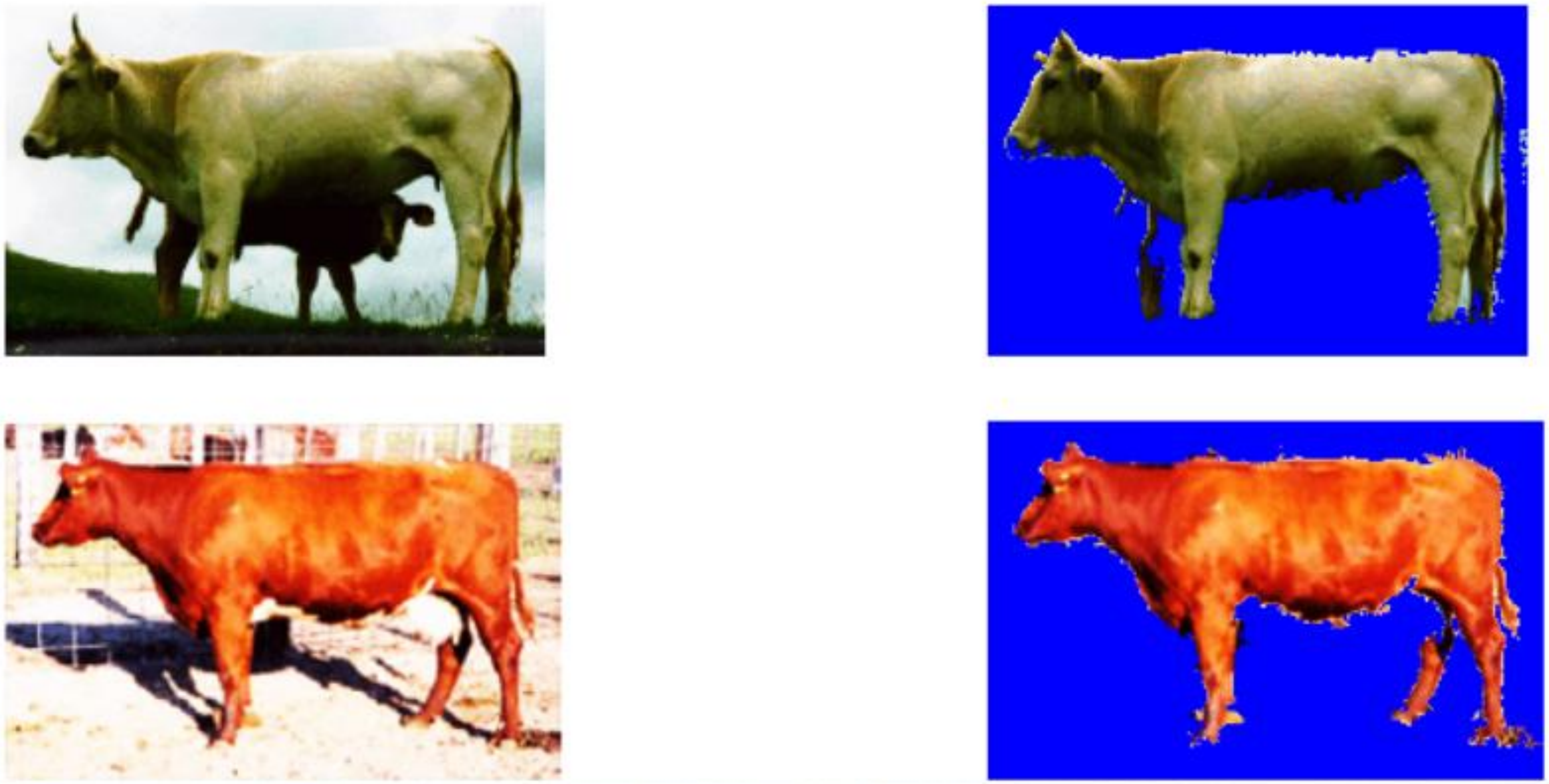
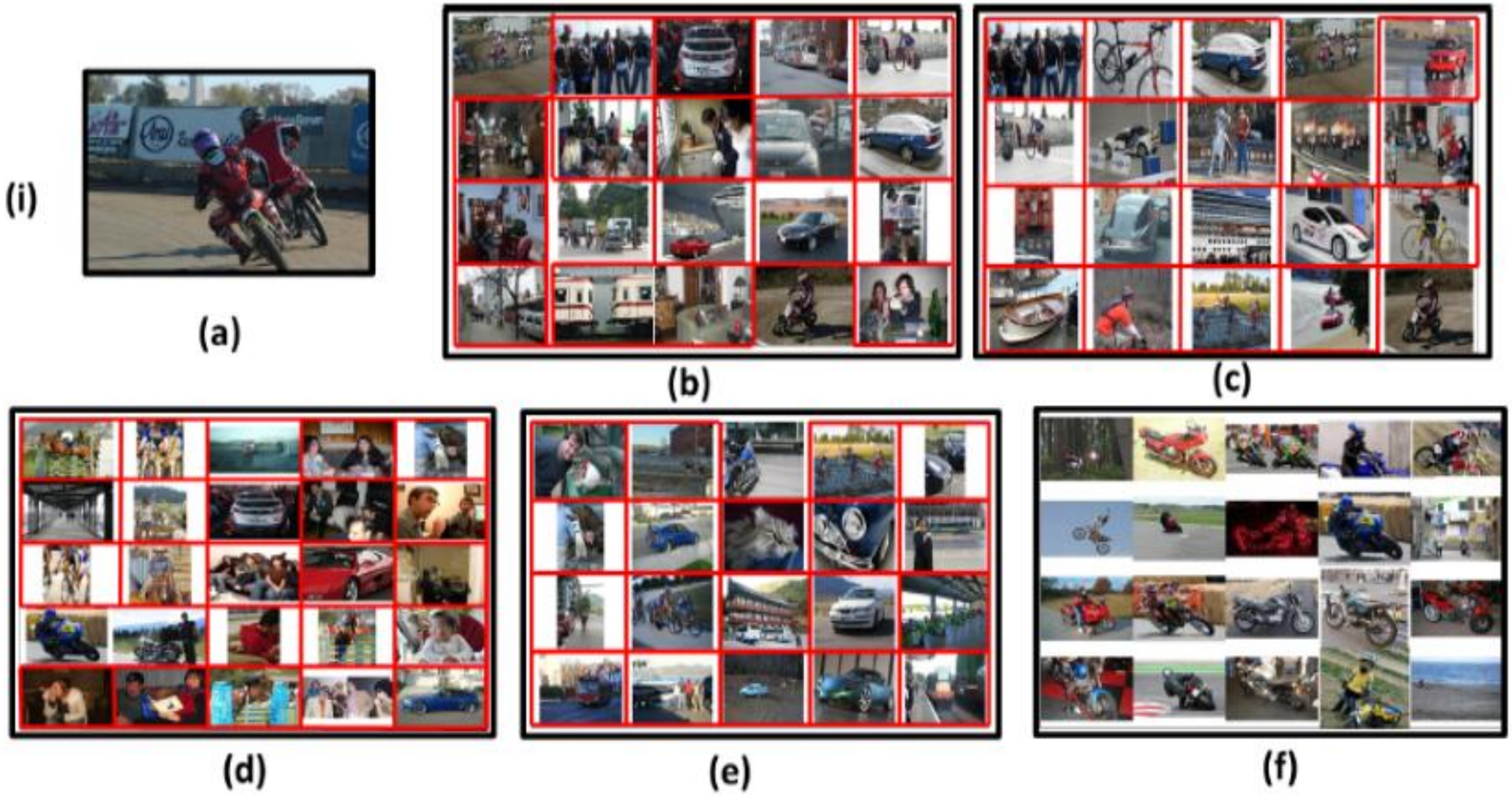


Image Segmentation



Object Detection or segmentation –
involves object Detection and recognition modules

Smart CBIR - Retrieval Results



Results of top 20 image retrievals (arranged in row-major order) shown for visual comparative study, using: (a) query image from the PASCAL datasets; (b) MTH (2010); (c) MSD (2011), (d) SLAR (2012); (e) CDH (2013); and (f) our proposed RADAR framework. Erroneous results are highlighted using a red template

Visual data on the Internet

- Flickr
  - 10+ billion photographs
  - 60 million images uploaded a month
- Facebook
  - 250 billion+
  - 300 million a day
- Instagram
  - 55 million a day
- YouTube
  - 100 hours uploaded every minute

CISCO 90% of net traffic will be visual!

Mostly about cats

Vision is Really Hard

- Vision is an amazing feature of natural intelligence
  - Visual cortex occupies about 50% of Macaque brain
  - More human brain devoted to vision than anything else

Why is Computer Vision Hard?

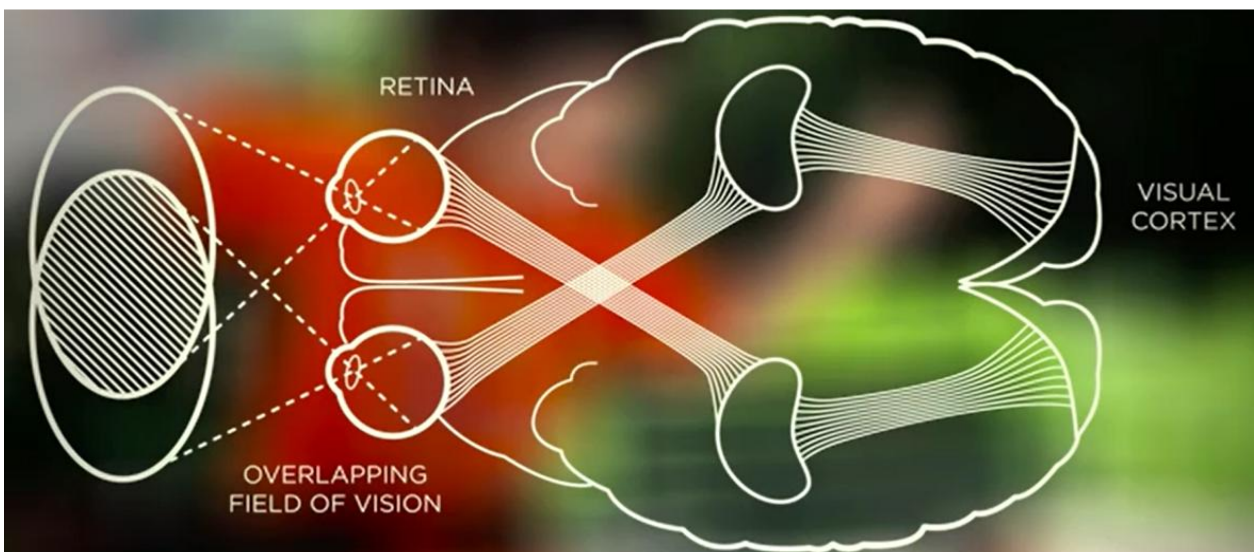
What did you see?

- Where this picture was taken?
- How many people are there?
- What are they doing?
- What object the person on the left standing on?
- Why this is a funny picture?

FEI FEI LI'S CONCEPT OF IMAGENET

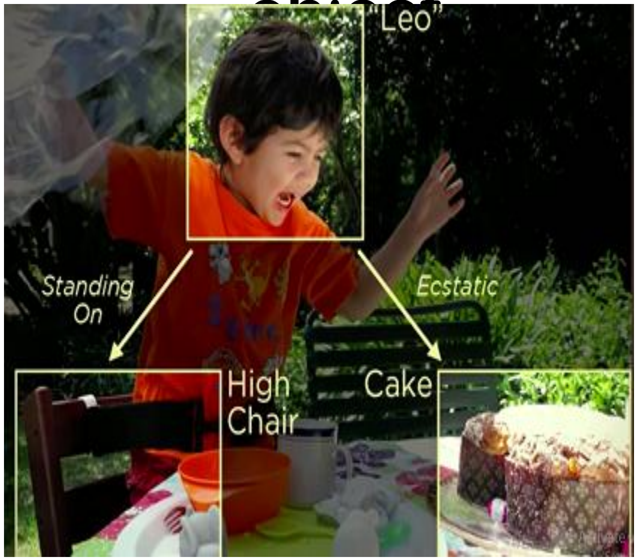
"If We Want Machines to Think, We Need to Teach Them to See"

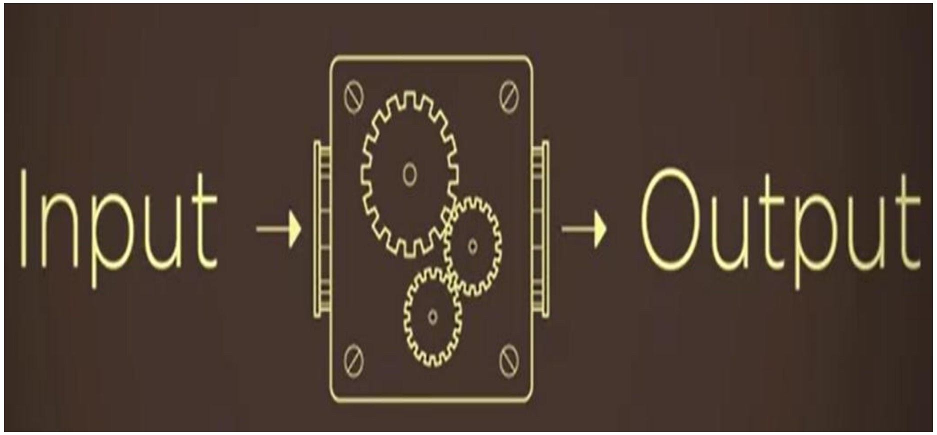
Vision begins with the eyes, but truly takes place in the brain.

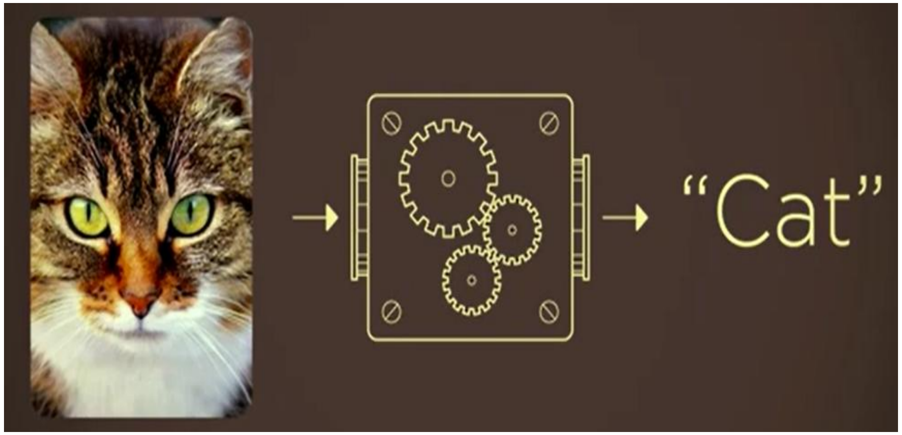


AIM OF IMAGENET

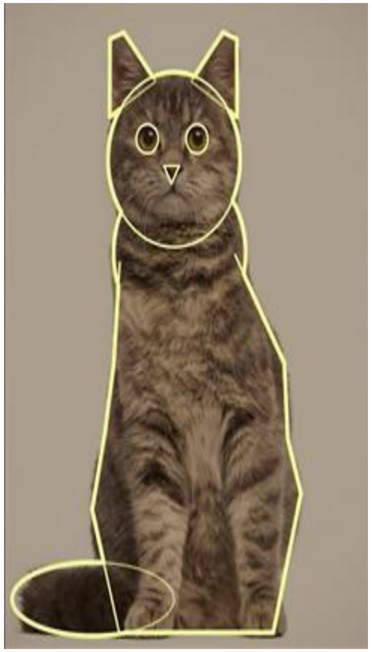
- To train the machine to recognize the input object

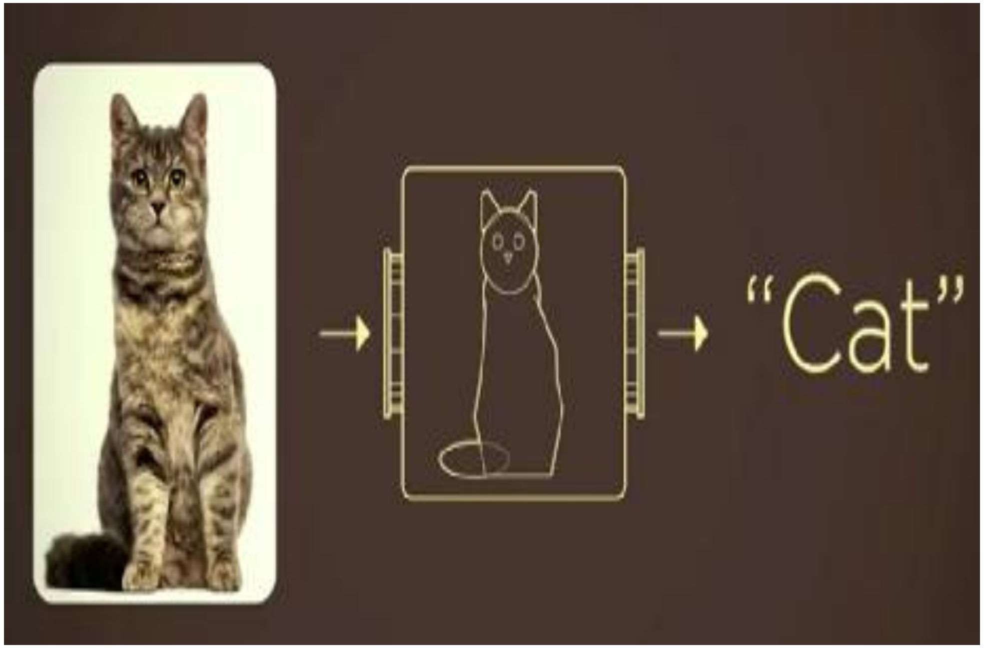




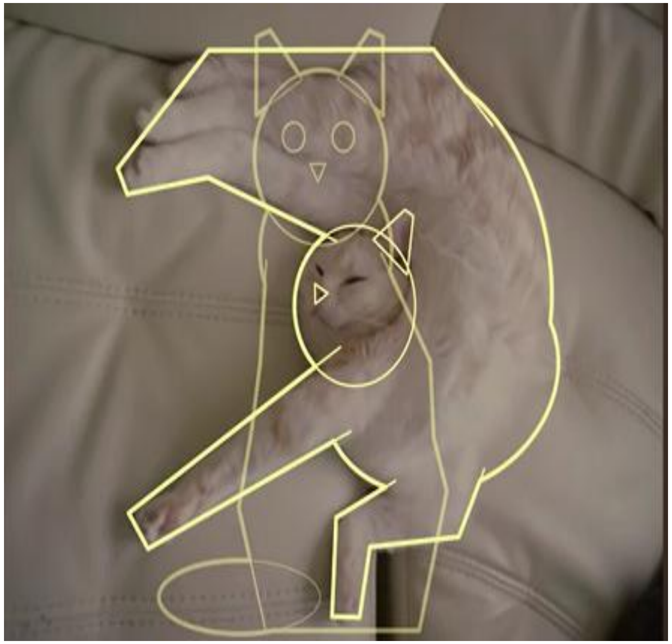


- **To train the objects a mathematical model is used identifying the face as a circular object with circular eyes and body structure related to cat.**

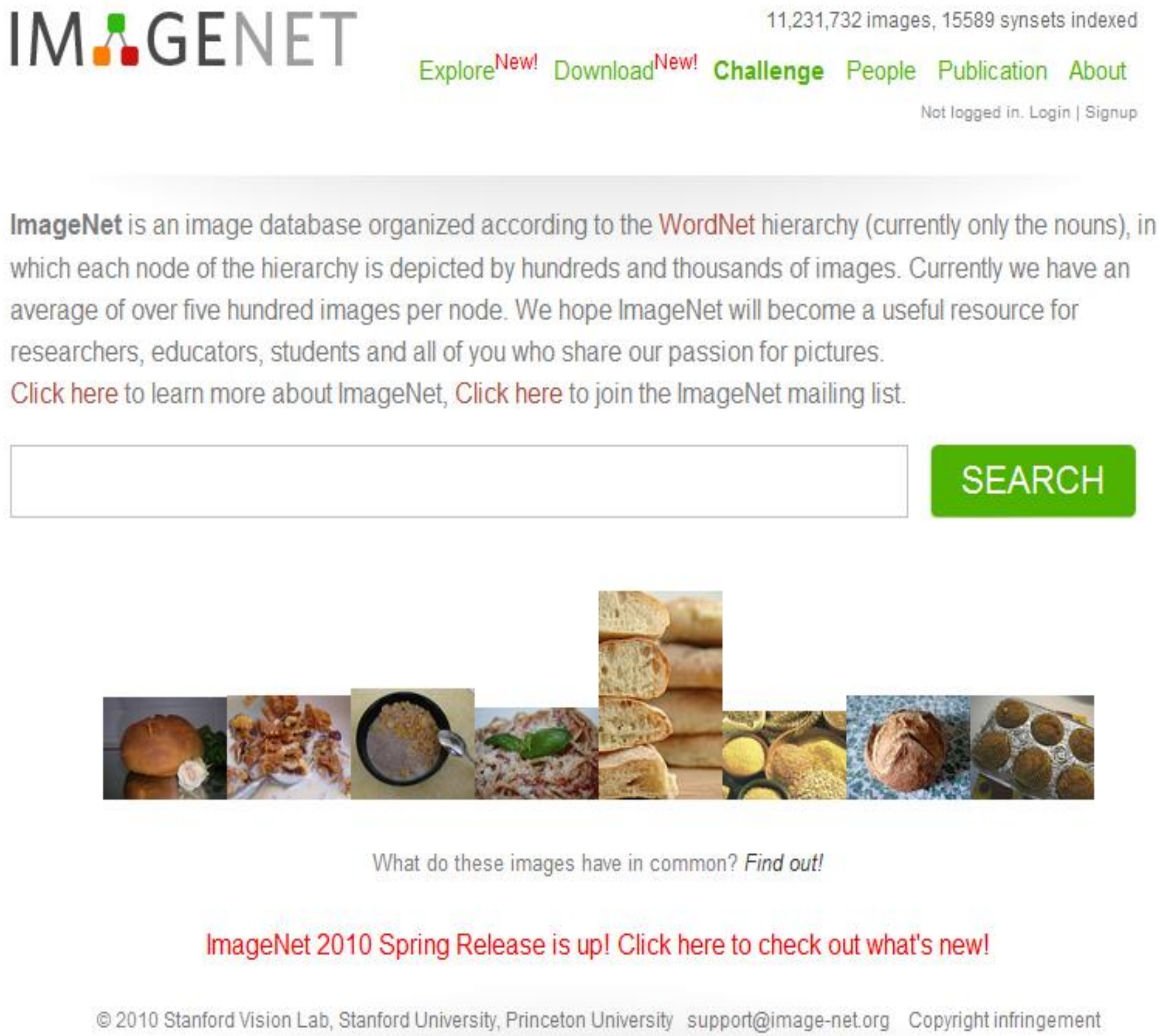




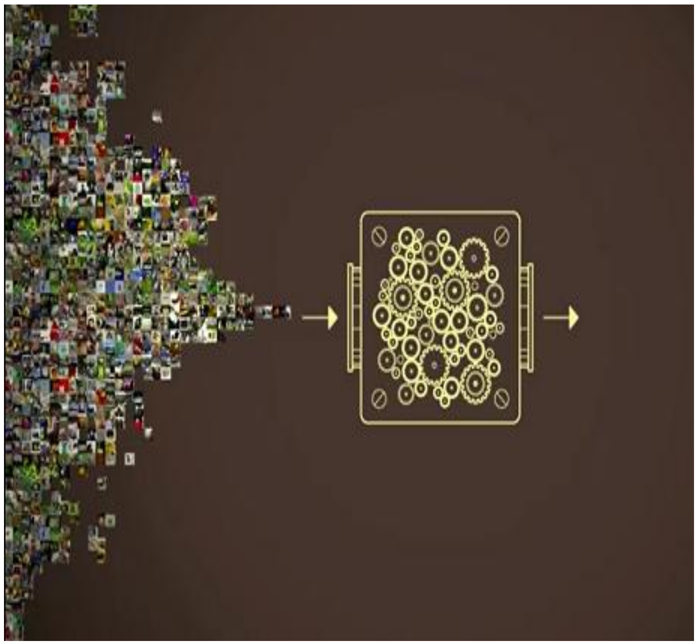
- **But if this happens… then model will fail….**

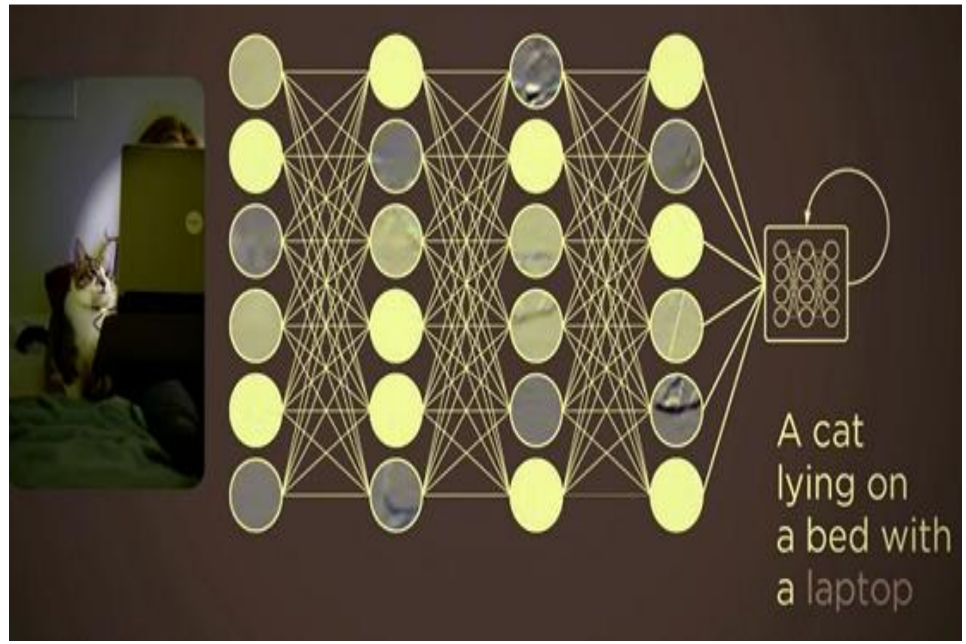


- **This gave rise to a large classified data set called IMAGENET**



IMAGENET + MACHINE + CNN ALGO = CORRECT RESULT





Some more results of IMAGENET so obtained are:

- **But it still fails a no. of times… so there are chances of improvement yet….**

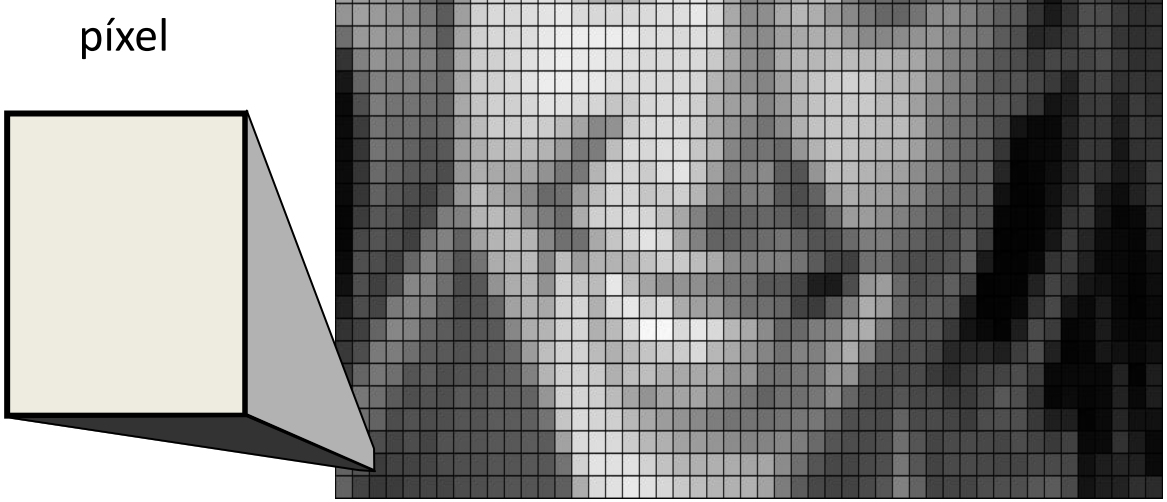
Digital Representation of Images

- A digital image is a matrix or two-dimensional array of numbers.
- Each cell in the matrix represents a pixel.Sample: Image of 20x15.

90	67	68	75	78	98	185	180	153	139	132	106	70	80	81	69	69	67	35	34
92	87	73	78	82	132	180	152	134	120	102	106	95	75	72	63	75	42	19	29
63	102	89	76	98	163	166	164	175	159	120	103	132	96	68	42	49	46	17	22
45	83	109	80	130	158	166	174	158	134	105	71	82	121	80	51	12	50	31	17
39	69	92	115	154	122	144	173	155	105	98	86	82	106	83	76	17	29	41	19
34	80	73	132	144	110	142	181	173	122	100	88	141	142	111	87	33	18	46	36
37	93	88	136	171	164	137	171	190	149	110	137	168	161	132	96	56	23	48	49
66	117	106	147	188	202	198	187	187	159	124	151	167	158	138	105	80	55	59	54
127	136	107	144	188	197	188	184	192	172	124	151	138	108	116	114	84	46	67	54
143	134	99	143	188	172	129	127	179	167	106	118	111	54	70	95	90	46	69	52
141	137	96	146	167	123	91	90	151	156	121	93	78	82	97	91	87	45	66	39
139	137	80	131	162	145	131	129	154	161	158	149	134	122	115	99	84	35	52	30
137	133	56	104	165	167	174	181	175	169	165	162	158	142	124	103	67	19	31	23
135	132	65	86	173	186	200	198	181	171	162	153	145	135	121	104	53	14	15	33
132	132	88	50	149	182	189	191	186	178	166	157	148	131	106	78	28	10	15	44

Digital Representation of Images

- A more common way to display an image …



The Three Stages of Computer Vision

- low-level

  image → image

- mid-level

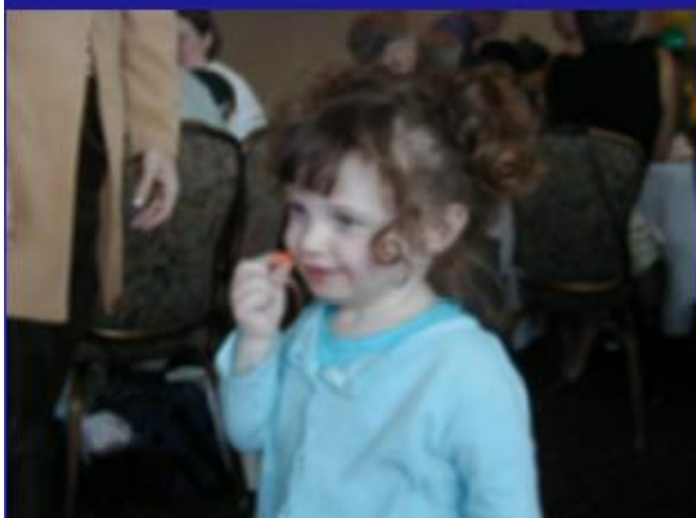
  image → features

- high-level

  features → analysis

Low-Level

sharpening



blurring

Low-Level

original image

Canny

edge image

Mid-Level

edge image

ORT

data structure

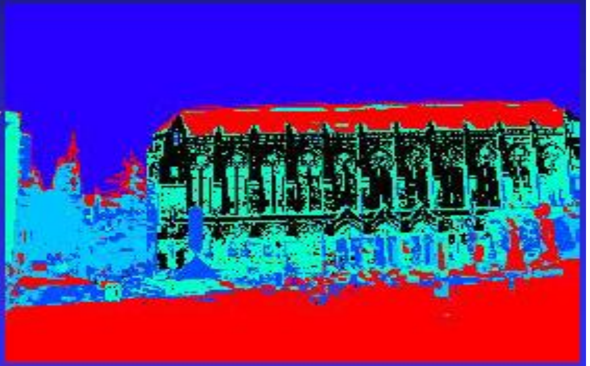
circular arcs and line segments

Mid-level

original color image

K-means clustering

(followed by connected component analysis)



regions of homogeneous color

data structure

Low- to High-Level

low-level →

edge image

↓ mid-level

consistent line clusters

← high-level

Building Recognition

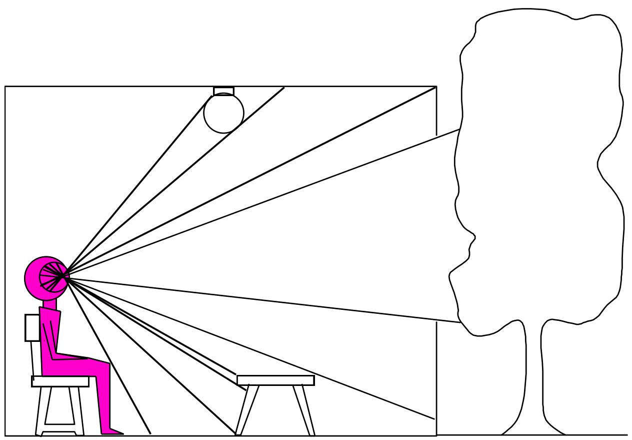
Image

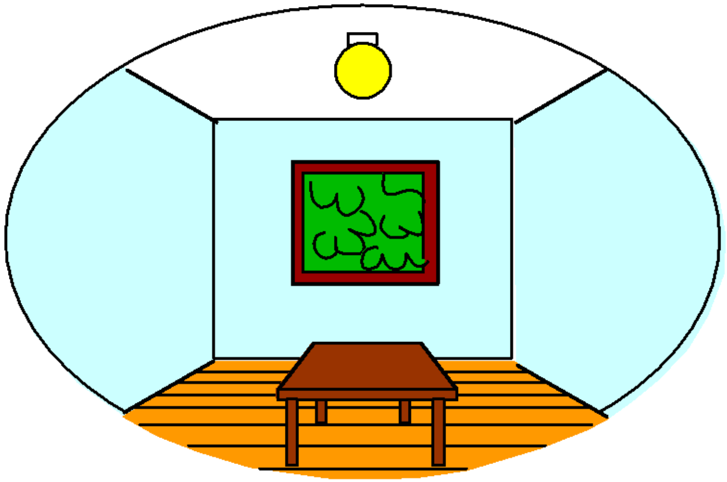
- 2-D array of numbers (intensity values, gray levels)
- Gray levels 0 (black) to 255 (white)
- Color image is 3 2-D arrays of numbers
  - Red
  - Green
  - Blue
- Resolution (number of rows and columns)
  - 128X128
  - 256X256
  - 512X512

Image formation

3D world

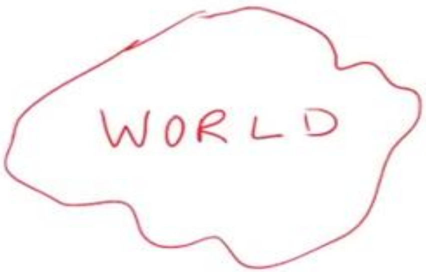
2D image





Point of observation

A camera creates an image …



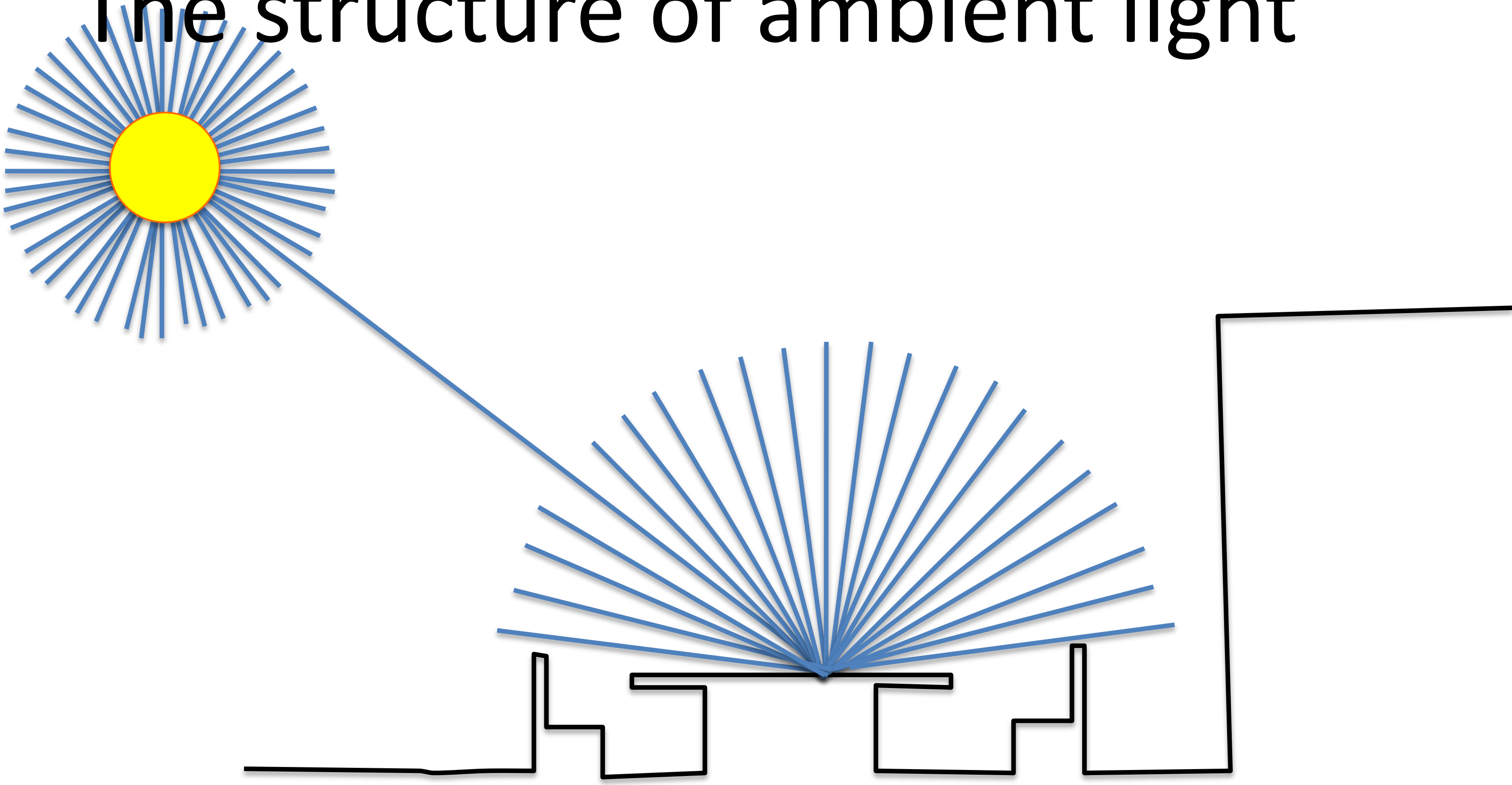
The image I(x,y) measures how much light is captured at pixel (x,y)

Cameras, lenses, and calibration

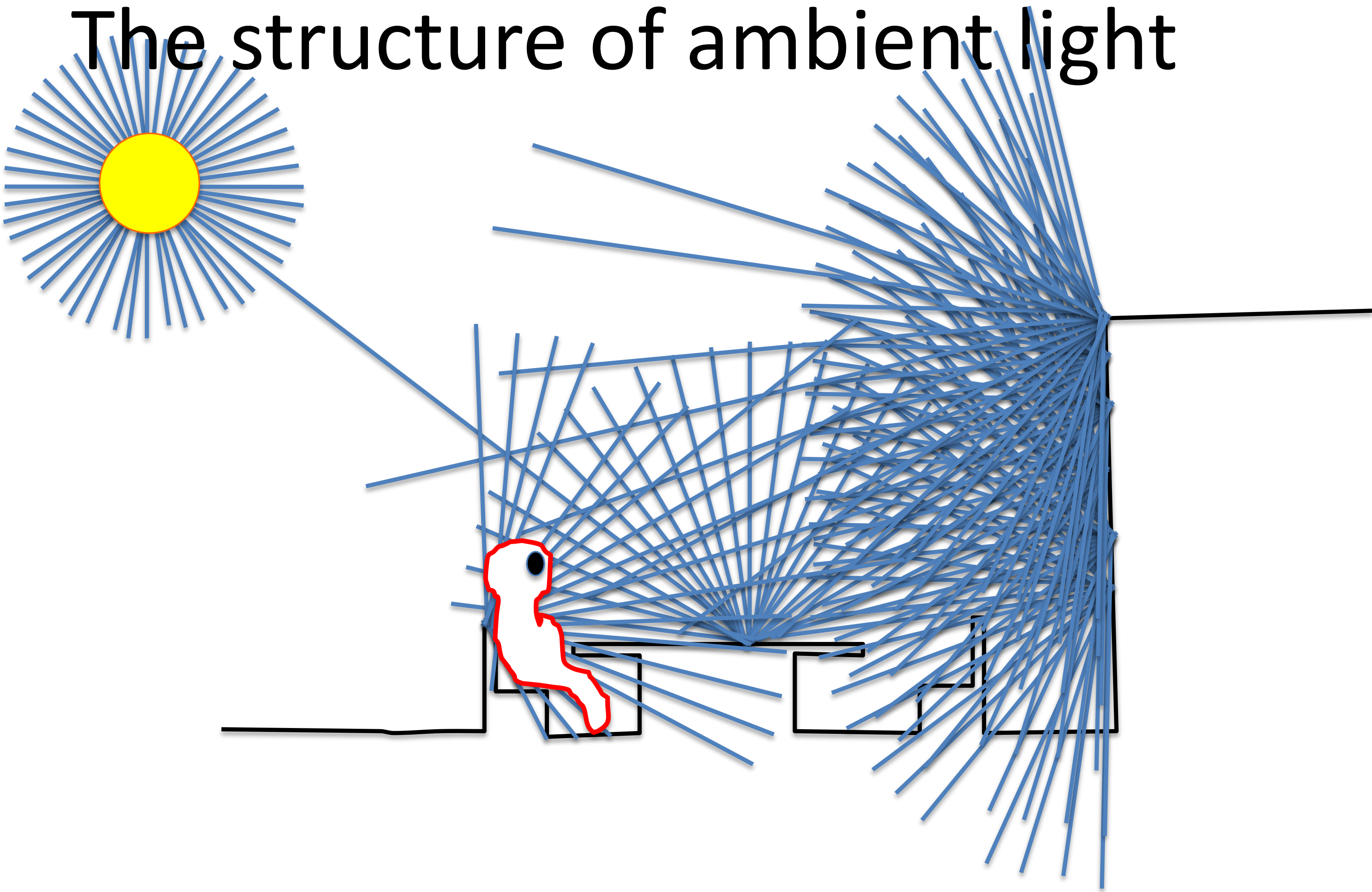
- Camera models
- Projection equations

Images are projections of the 3-D world onto a 2-D plane…

The structure of ambient light

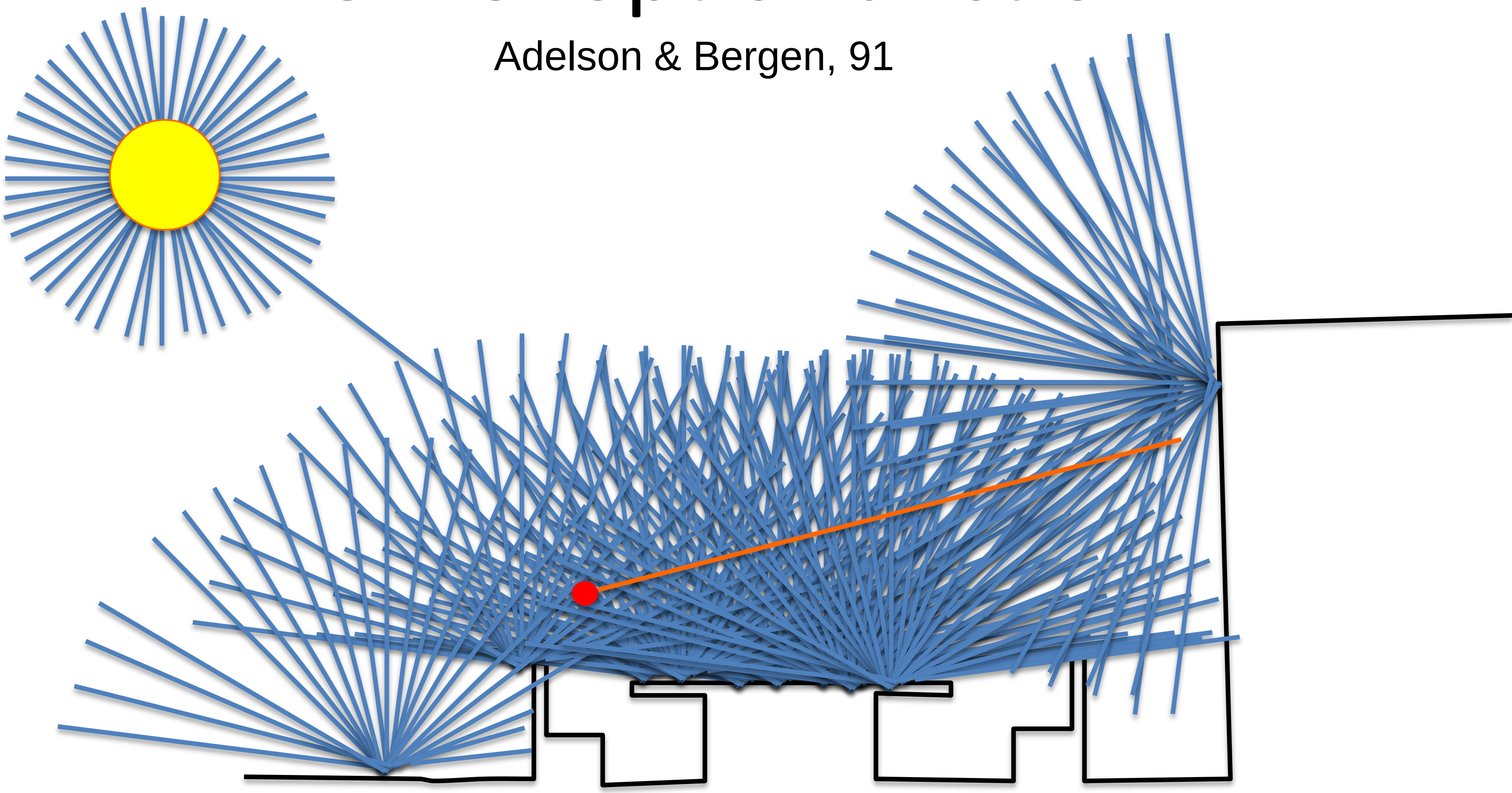


The structure of ambient light



The Plenoptic Function

Adelson & Bergen, 91

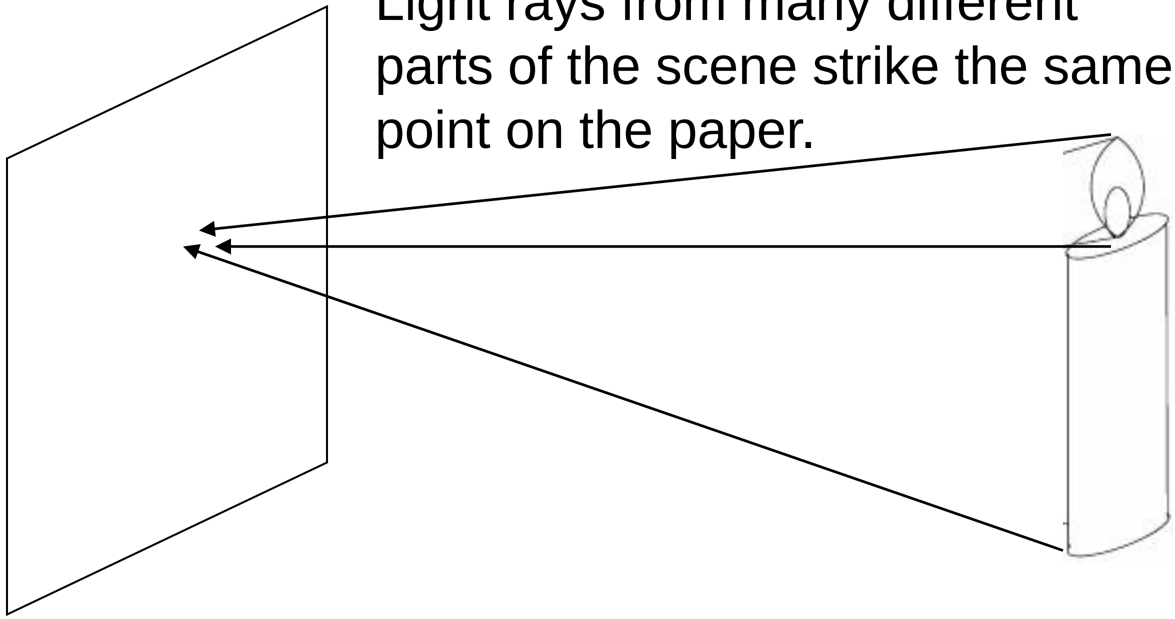


The intensity P can be parameterized as:

$$P(\theta, \phi, \lambda, t, X, Y, Z)$$

“The complete set of all convergence points constitutes the permanent possibilities of vision.” Gibson

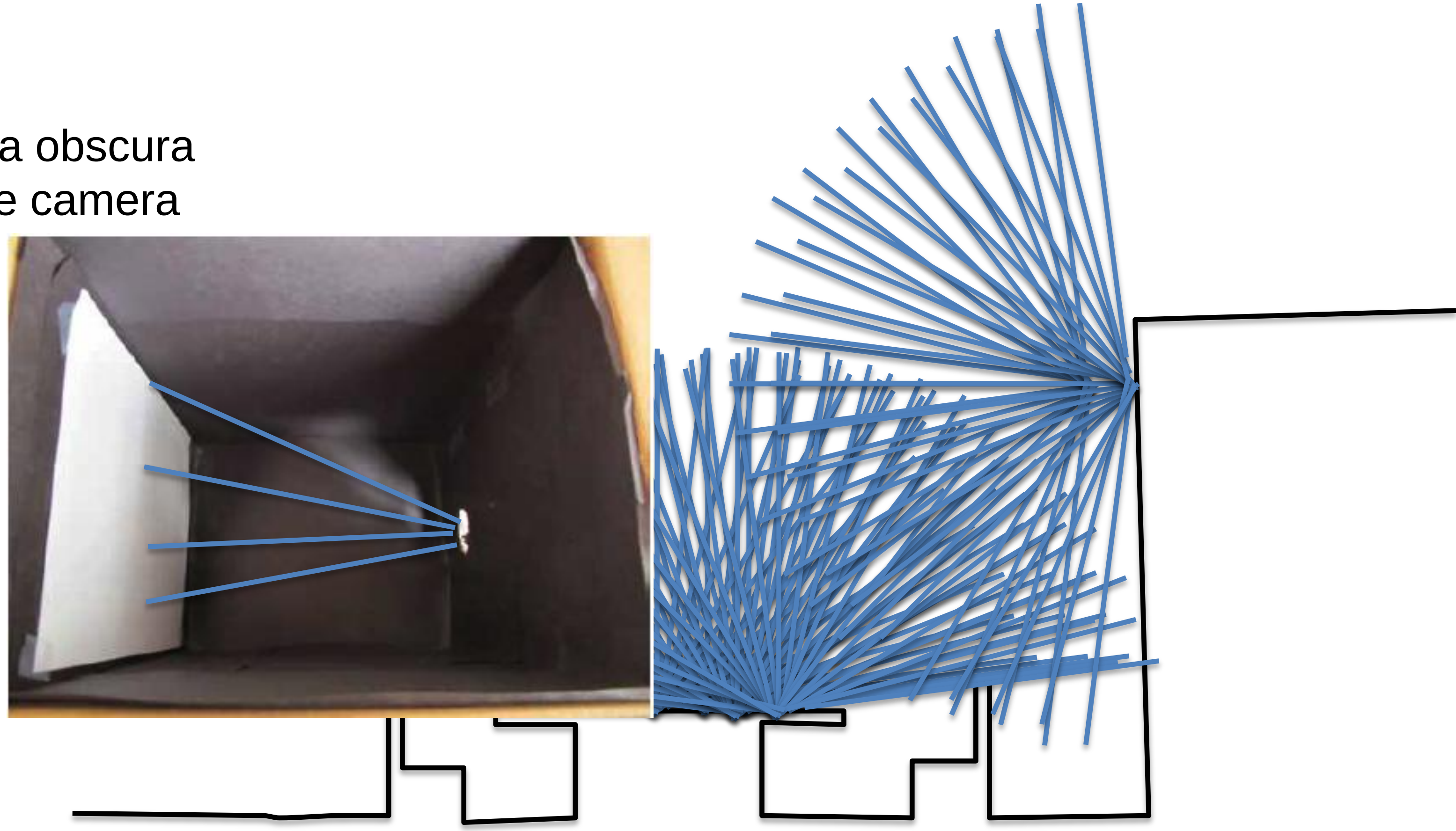
Light rays from many different parts of the scene strike the same point on the paper.



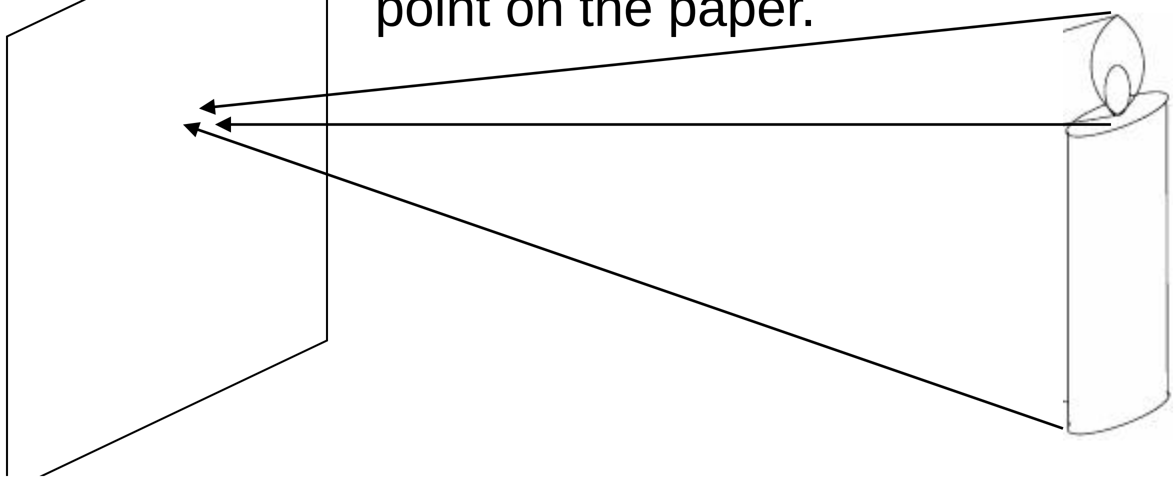
Forsyth & Ponce

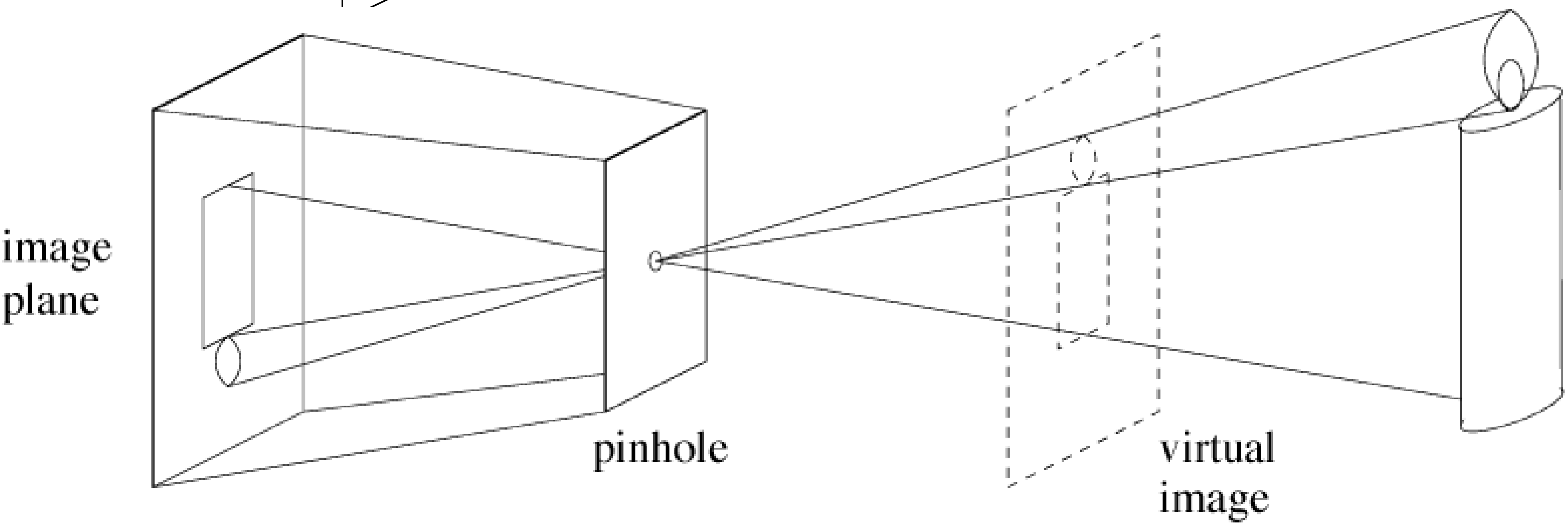
Measuring the Plenoptic function

The camera obscura
The pinhole camera



Light rays from many different parts of the scene strike the same point on the paper.





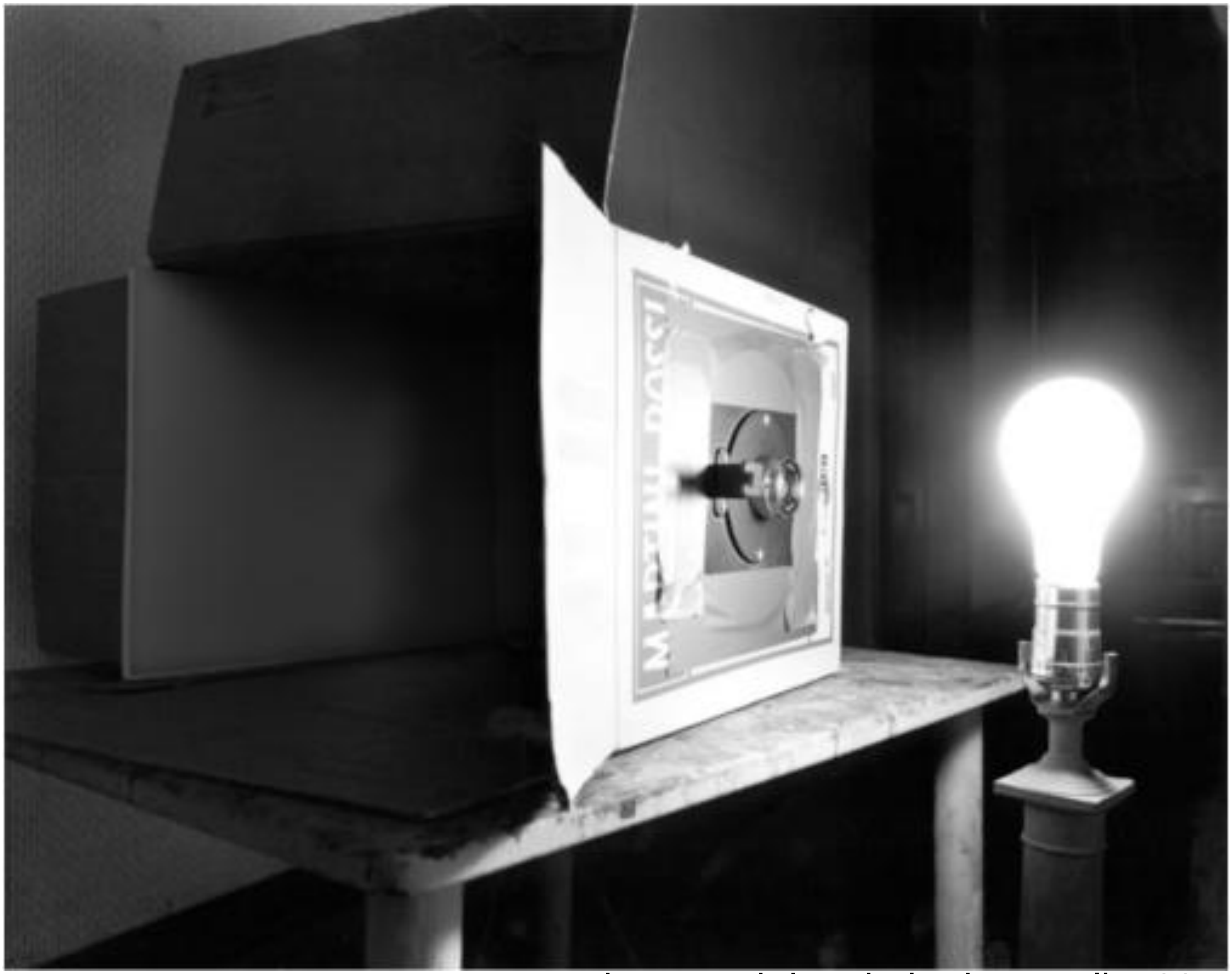
The pinhole camera only allows rays from one point in the scene to strike each point of the paper.

Forsyth & Ponce

Pinhole camera

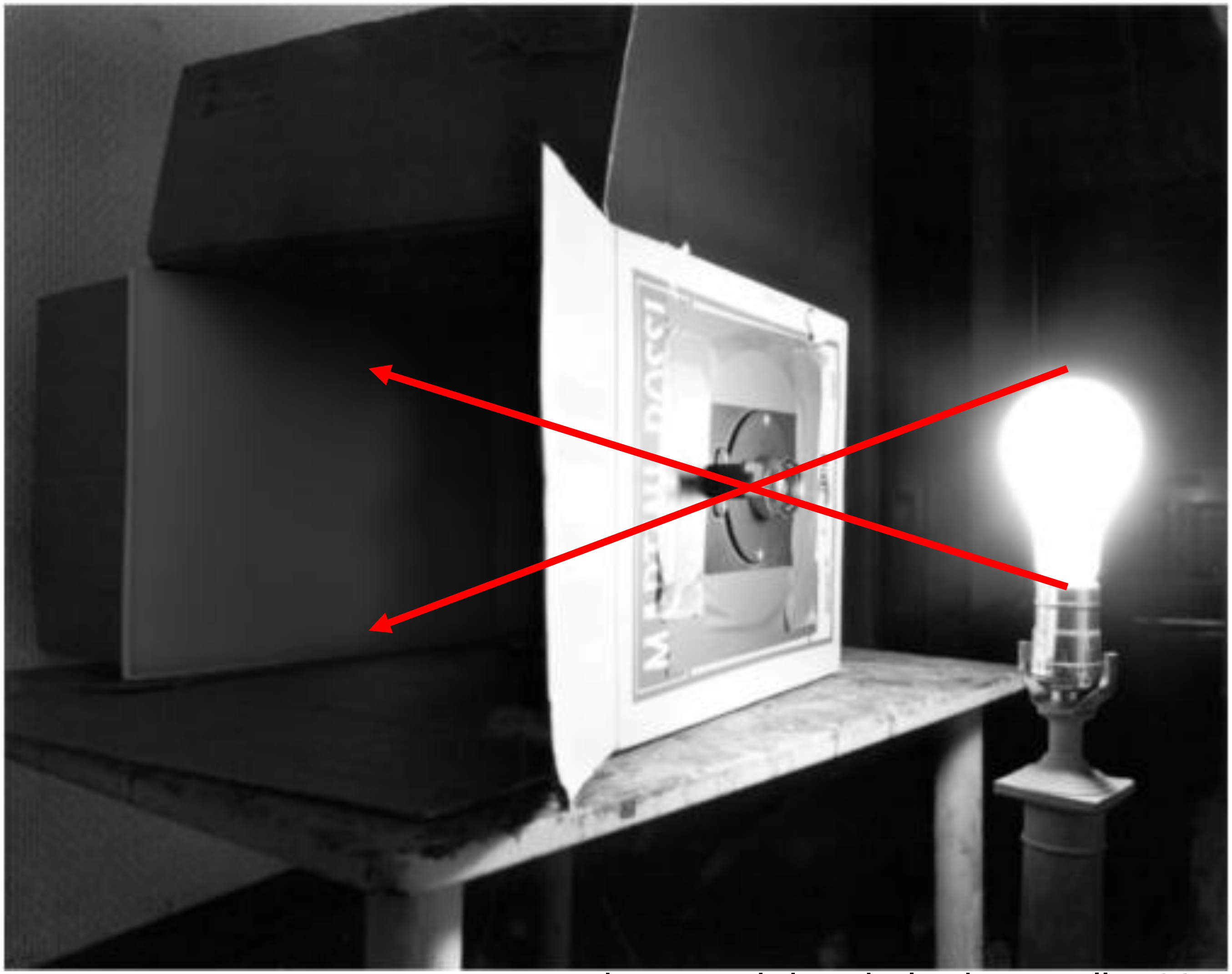
Photograph by Abelardo Morell, 1991

Pinhole camera



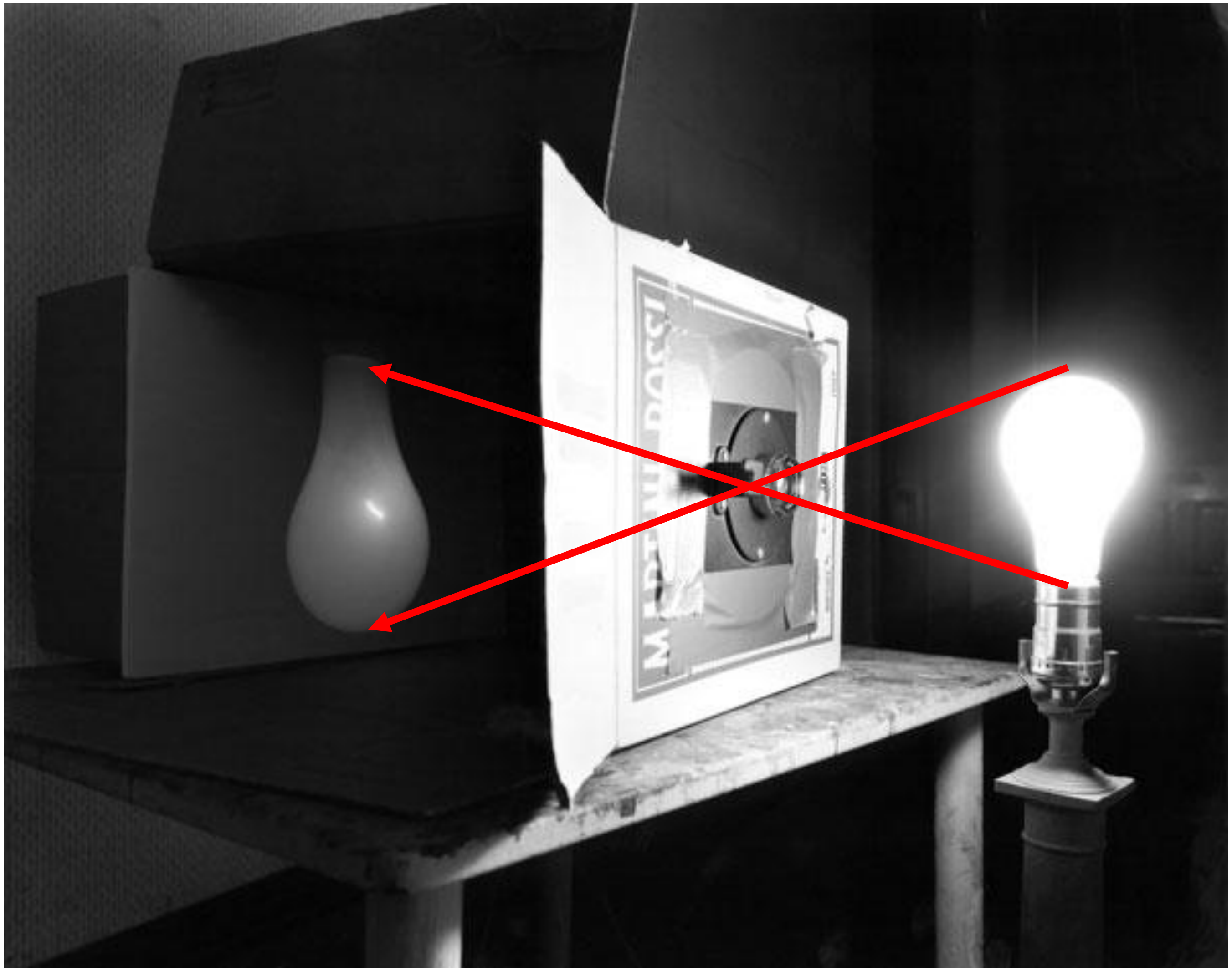
Photograph by Abelardo Morell, 1991

Pinhole camera



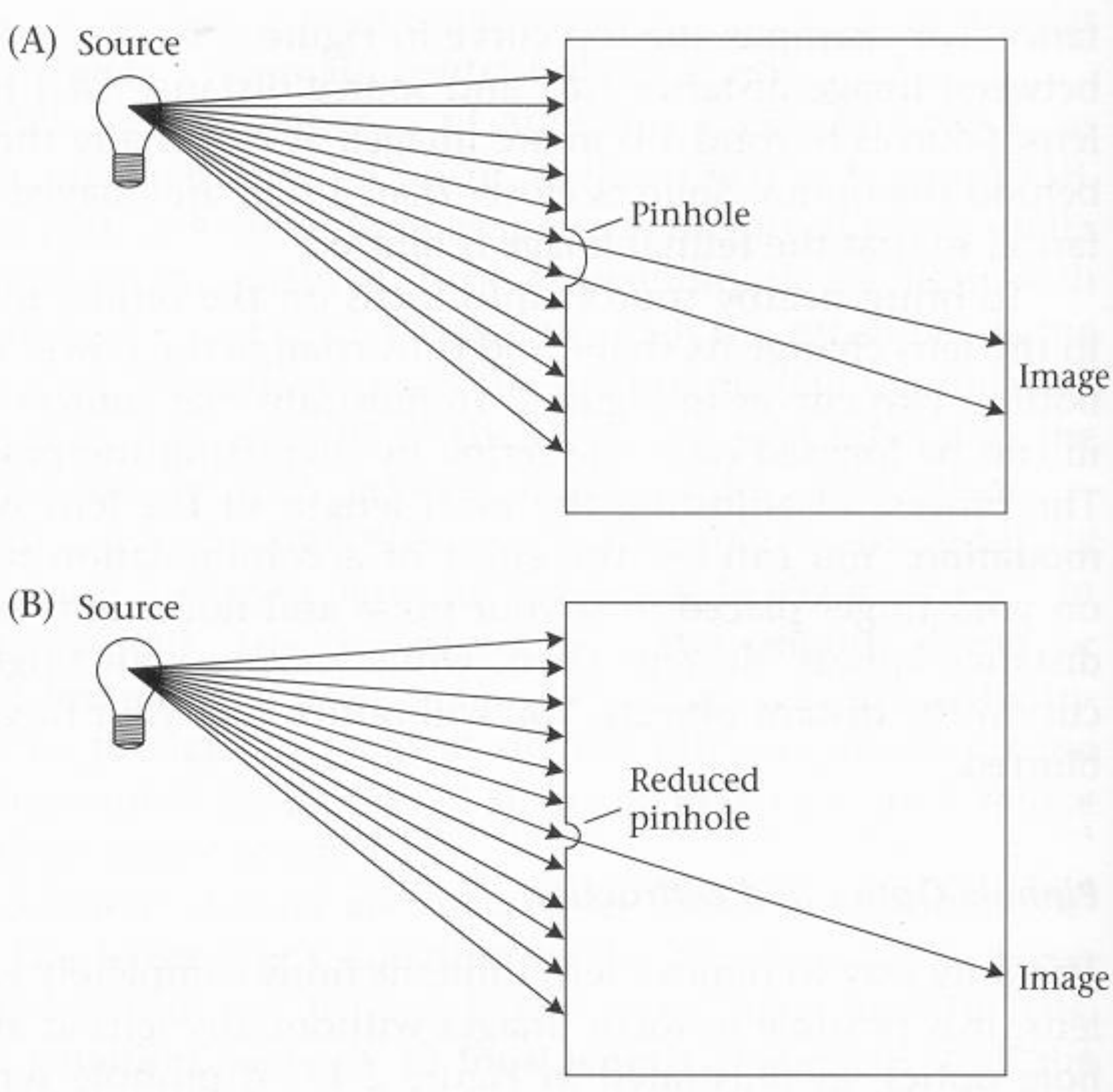
Photograph by Abelardo Morell, 1991

Pinhole camera



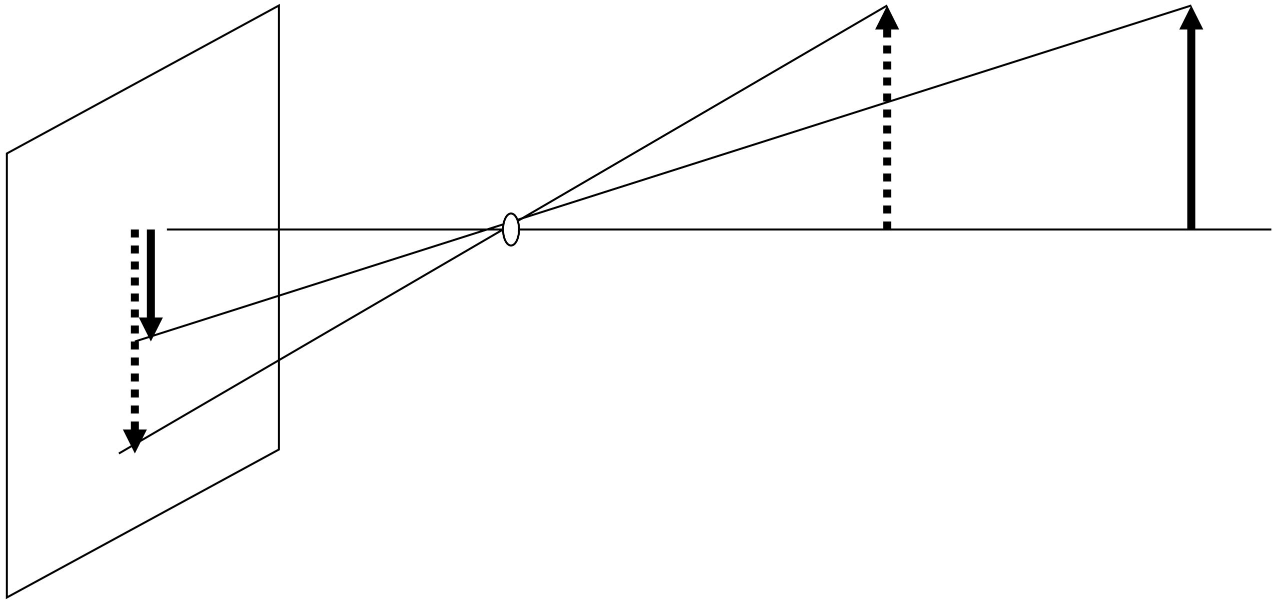
Photograph by Abelardo Morell, 1991

Effect of pinhole size



Wandell, Foundations of Vision, Sinauer, 1995

Measuring distance

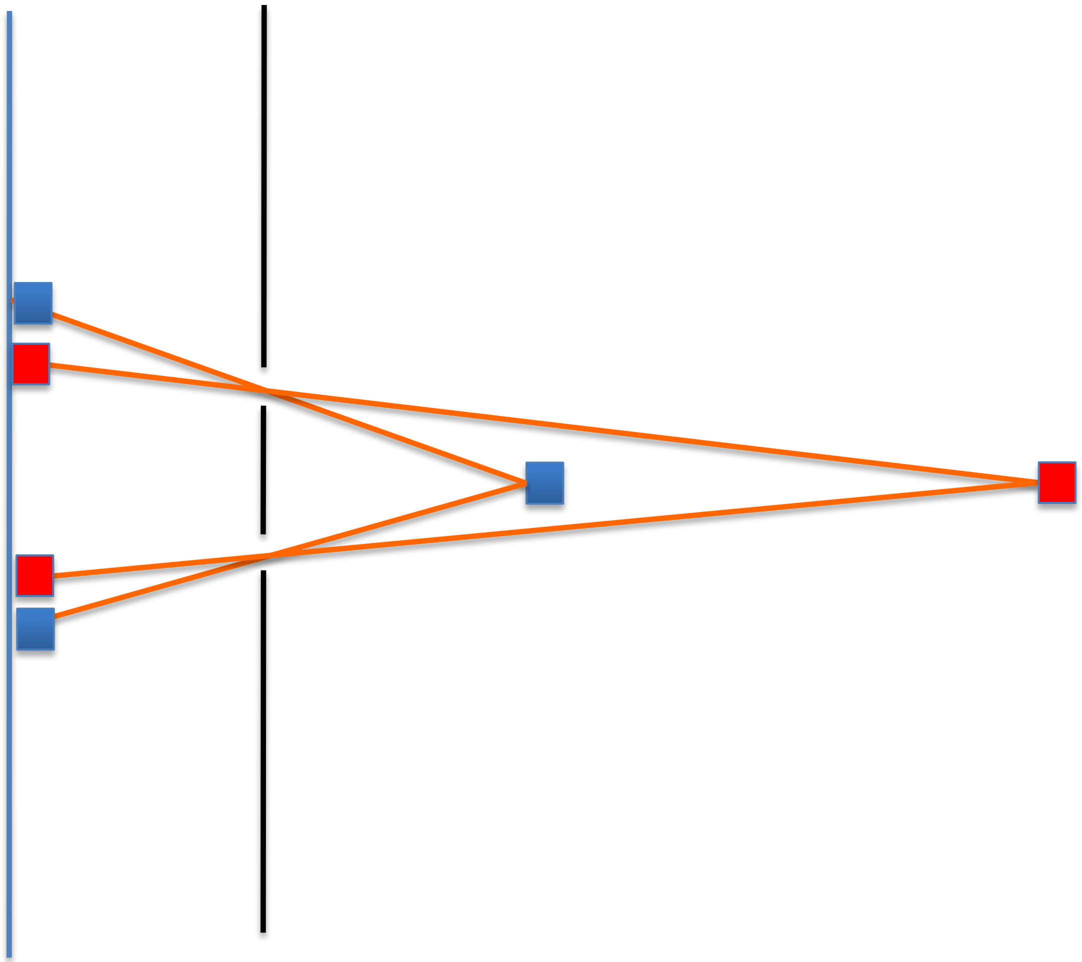


• Object size decreases with distance to the pinhole

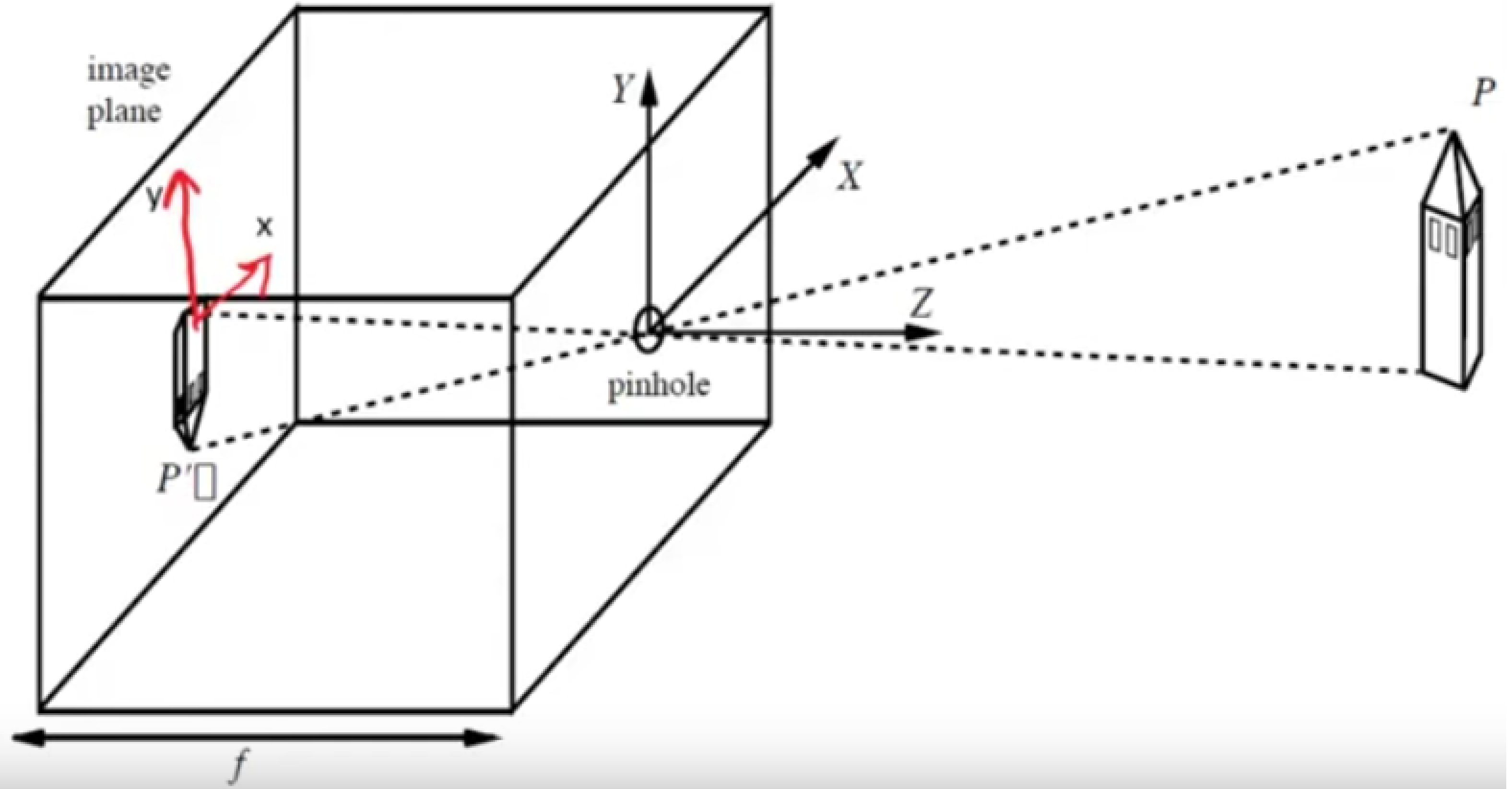
• There, given a single projection, if we know the size of the object we can know how far it is.

• But for objects of unknown size, the 3D information seems to be lost.

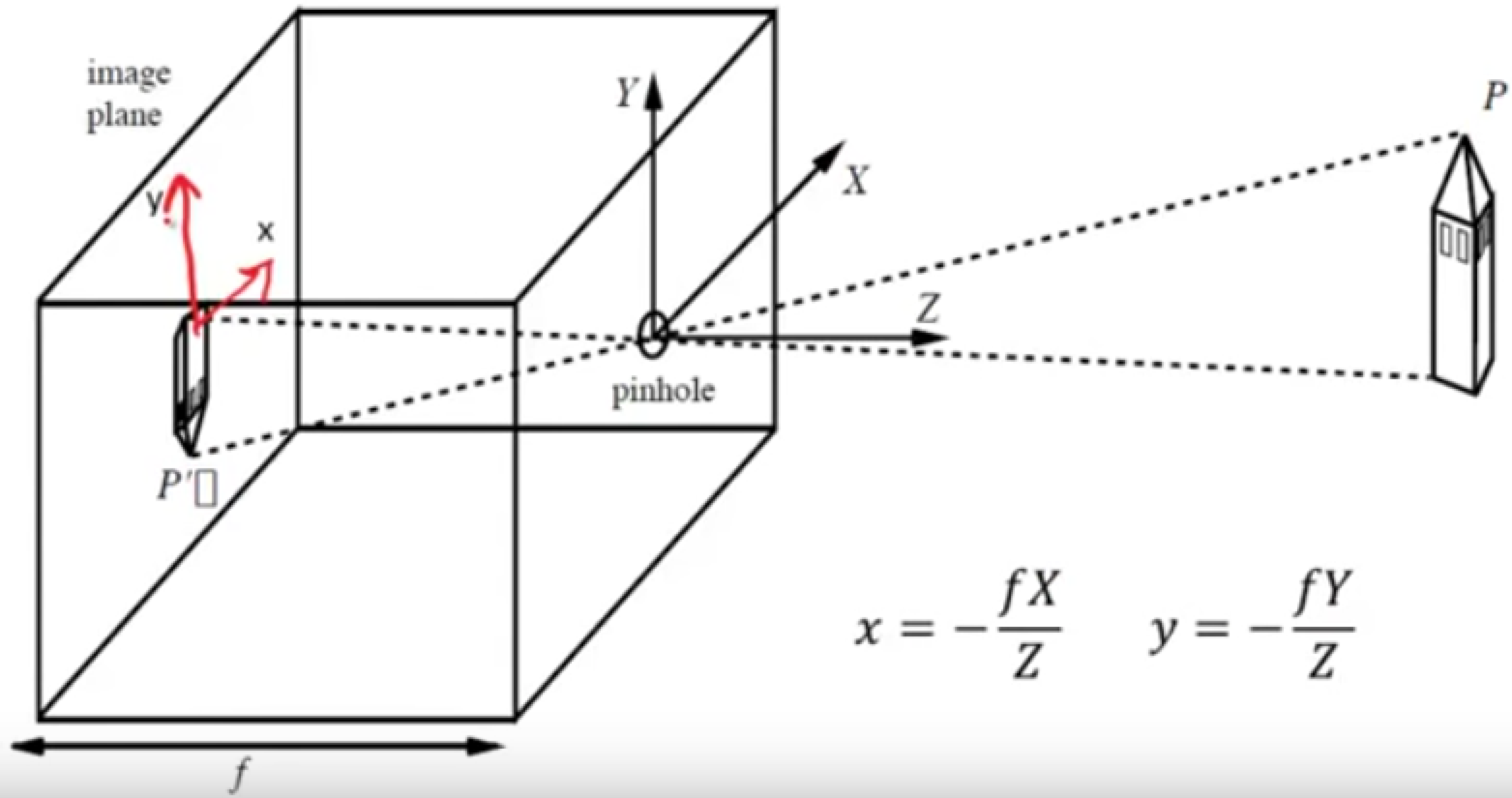
Two pinholes



The Pinhole Camera



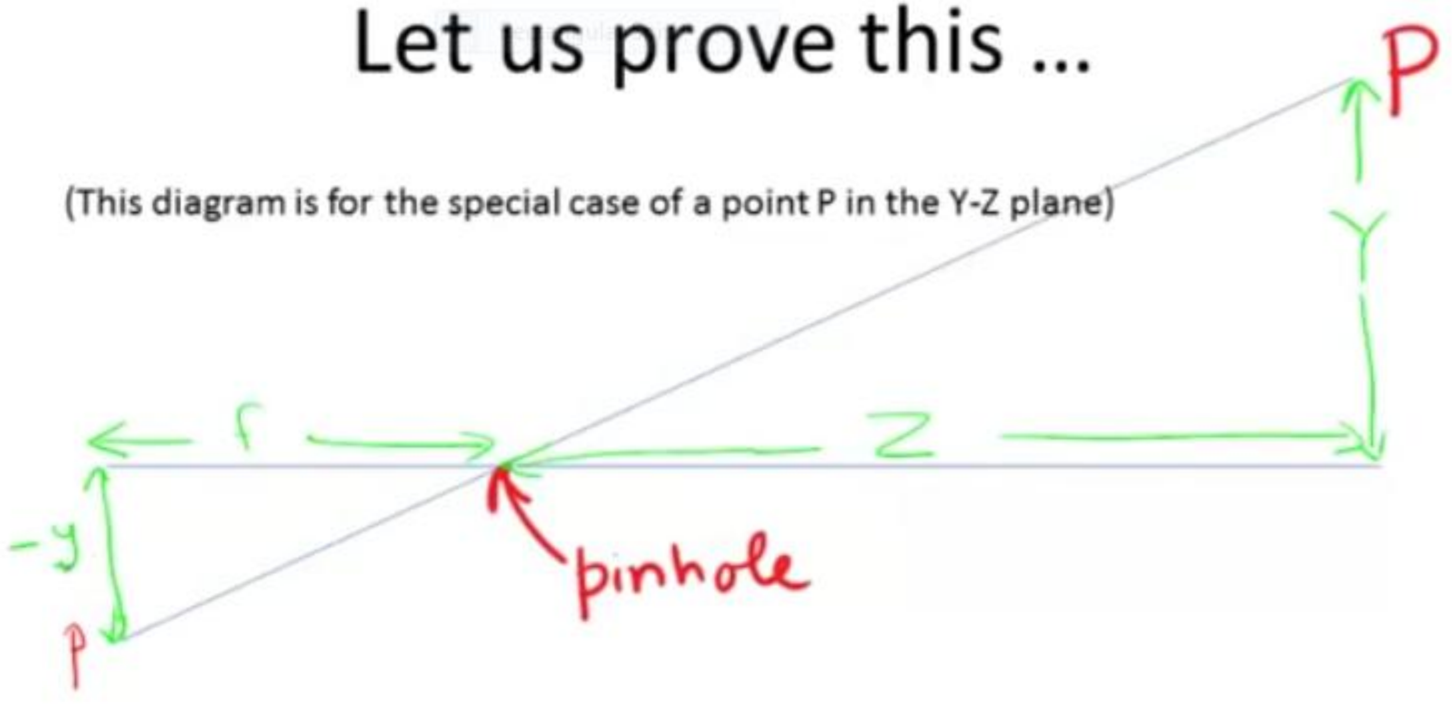
The Pinhole Camera



$$x = -\frac{fX}{Z} \qquad y = -\frac{fY}{Z}$$

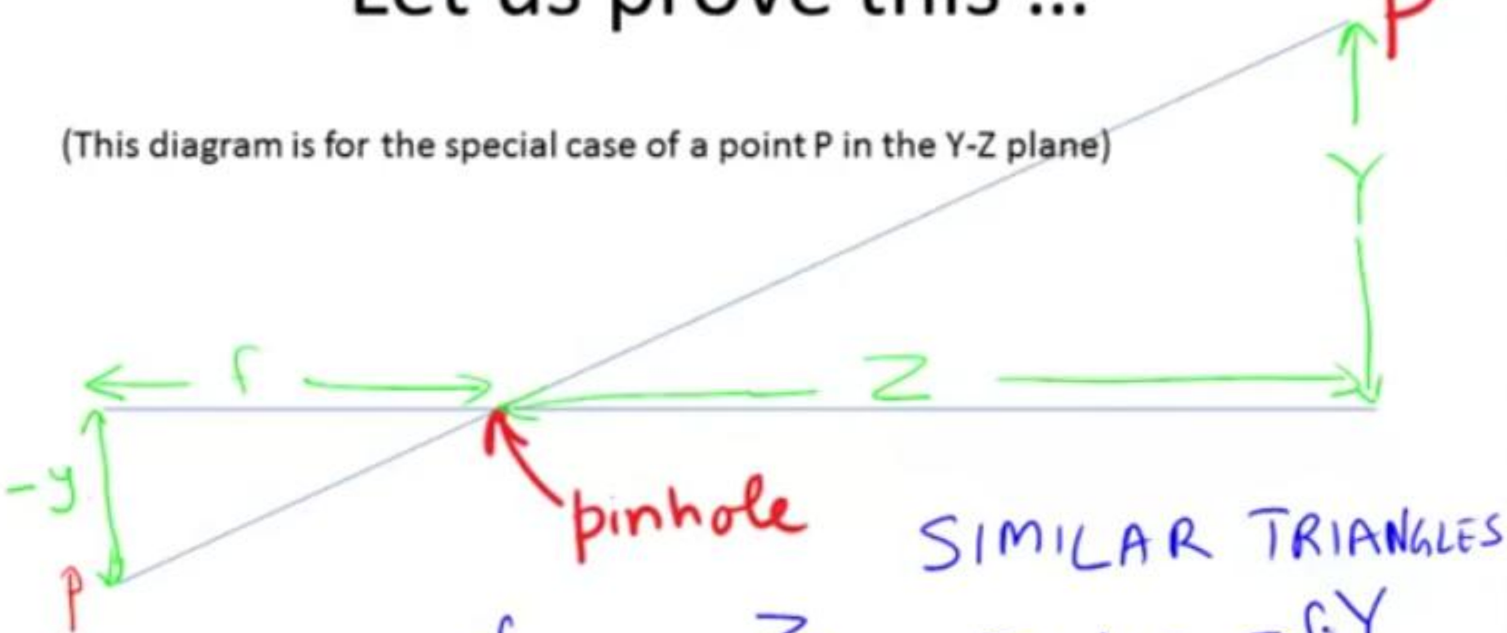
Let us prove this ...

(This diagram is for the special case of a point P in the Y-Z plane)



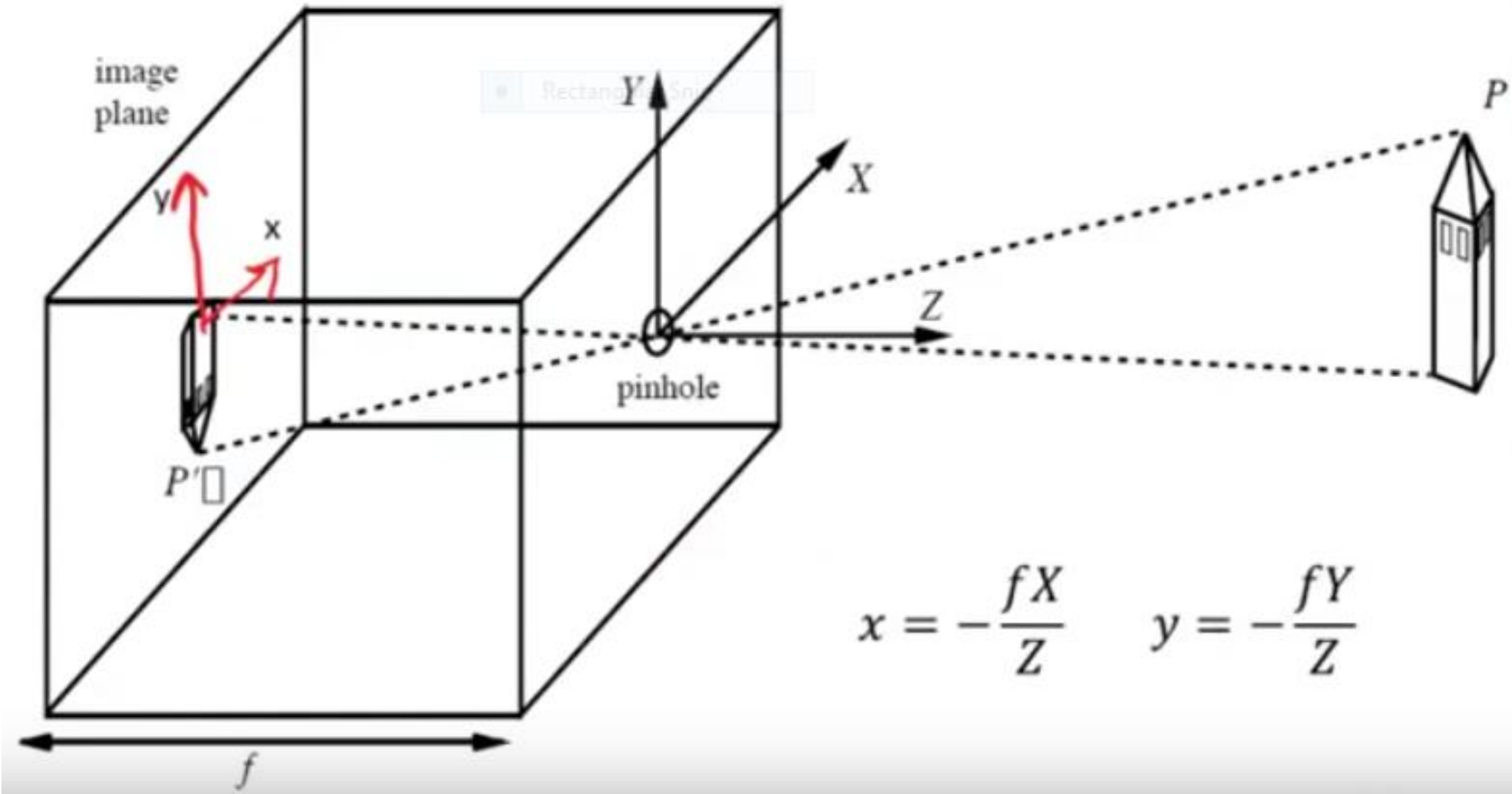
Let us prove this ...

(This diagram is for the special case of a point P in the Y-Z plane)



SIMILAR TRIANGLES

$$\frac{f}{-y} = \frac{Z}{Y} \Rightarrow y = \frac{-fY}{Z}$$



$$x = -\frac{fX}{Z} \qquad y = -\frac{fY}{Z}$$

Basic Relationships Between Pixels

- Neighborhood
- Adjacency
- Connectivity
- Paths
- Regions and boundaries

Neighbors of a Pixel

- Any pixel $p(x, y)$ has two vertical and two horizontal neighbors, given by

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1)$$

- This set of pixels are called the 4-neighbors of P, and is denoted by $N_4(P)$.

- Each of them are at a unit distance from P.

Neighbors of a Pixel

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & f(0,3) & f(0,4) & ----- \\ f(1,0) & f(1,1) & f(1,2) & f(1,3) & f(1,4) & ----- \\ f(2,0) & f(2,1) & f(2,2) & f(2,3) & f(2,4) & ----- \\ f(3,0) & f(3,1) & f(3,2) & f(3,3) & f(3,4) & ----- \\ | & | & | & | & | & ----- \\ | & | & | & | & | & ----- \end{bmatrix}$$

- A Pixel p at coordinates (x, y) has 4 horizontal and vertical neighbors.
- Their coordinates are given by:

$(x+1, y)$	$(x-1, y)$	$(x, y+1)$	&	$(x, y-1)$
f(2,1)	f(0,1)	f(1,2)		f(1,0)

- This set of pixels is called the ***4-neighbors*** of p denoted by $N_4(p)$.
- Each pixel is unit distance from (x ,y).

Neighbors of a Pixel (Contd..)

- The four diagonal neighbors of p(x,y) are given by,

$(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)$

- This set is denoted by $N_D(P)$.

- Each of them are at Euclidean distance of 1.414 from P.

Neighbors of a Pixel

$$
f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & f(0,3) & f(0,4) & \text{-----} \\ f(1,0) & f(1,1) & f(1,2) & f(1,3) & f(1,4) & \text{-----} \\ f(2,0) & f(2,1) & f(2,2) & f(2,3) & f(2,4) & \text{-----} \\ f(3,0) & f(3,1) & f(3,2) & f(3,3) & f(3,4) & \text{-----} \\ | & | & | & | & | & \text{-----} \\ | & | & | & | & | & \text{-----} \end{bmatrix}
$$

- A Pixel p at coordinates (x, y) has 4 diagonal neighbors.
- Their coordinates are given by:

(x+1, y+1)	(x+1, y-1)	(x-1, y+1)	&	(x-1, y-1)
f(2,2)	f(2,0)	f(0,2)		f(0,0)

- This set of pixels is called the ***diagonal-neighbors*** of p denoted by $N_D(p)$.
- diagonal neighbors + 4-neighbors = 8-neighbors of p.
- They are denoted by $N_8(p)$. So, $N_8(p) = N_4(p) + N_D(p)$

Neighbors of a Pixel (Contd..)

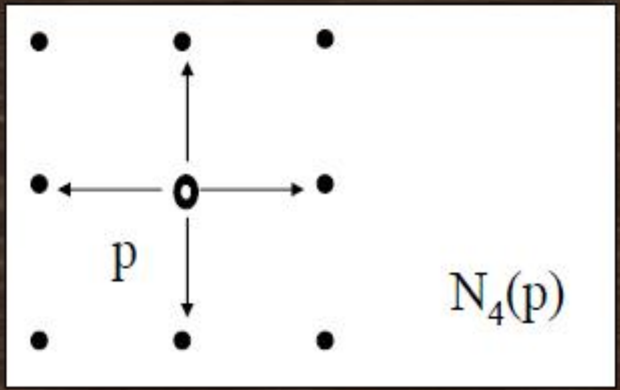
- The points $N_D(P)$ and $N_4(P)$ are together known as 8-neighbors of the point P, denoted by $N_8(P)$.

- Some of the points in the N_4, N_D and N_8 may fall outside image when P lies on the border of image.

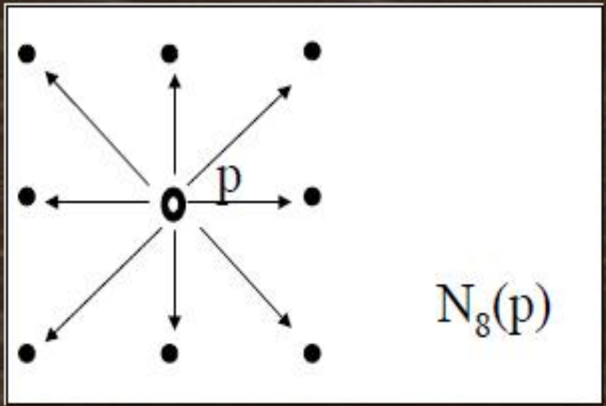
Neighbors of a Pixel (Contd..)

Neighbors of a pixel

a. 4-neighbors **of a pixel p are its vertical and horizontal neighbors denoted by $N_4(p)$**



b. 8-neighbors **of a pixel p are its vertical horizontal and 4 diagonal neighbors denoted by $N_8(p)$**



Neighbors of a Pixel (Contd..)

- N_4 - 4-neighbors
- N_D - diagonal neighbors
- N_8 - 8-neighbors (N_4 U N_D)

Adjacency

- Two pixels are connected if they are neighbors and their gray levels satisfy some specified criterion of similarity.

- For example, in a binary image two pixels are connected if they are 4-neighbors and have same value (0/1).

Adjacency, Connectivity

Adjacency: Two pixels are adjacent if they are neighbors and their intensity level 'V' satisfy some specific criteria of similarity.

e.g. V = {1}

V = { 0, 2}

Binary image = { 0, 1}

Gray scale image = { 0, 1, 2, ------, 255}

In binary images, 2 pixels are adjacent if they are neighbors & have some intensity values either 0 or 1.

In gray scale, image contains more gray level values in range 0 to 255.

Adjacency (contd.)

- Let V be set of gray levels values used to define adjacency.
- **4-adjacency**: Two pixels *p* and *q* with values from V are 4-adjacent if q is in the set $N_4(p)$.
- **8-adjacency**: Two pixels *p* and *q* with values from V are 8-adjacent if q is in the set $N_8(p)$.
- **m-adjacency**: Two pixels *p* and *q* with values from V are m-adjacent if,
  - q is in $N_4(P)$.
  - q is in $N_D(p)$ and the set [$N_4(p) \cap N_4(q)$] is empty (has no pixels whose values are from V).

Adjacency, Connectivity

4-adjacency: Two pixels p and q with the values from set 'V' are 4-adjacent if q is in the set of $N_4(p)$.

e.g. V = { 0, 1}

1	1	0
1	1	0
1	0	1

p in RED color

q can be any value in GREEN color.

Adjacency, Connectivity

8-adjacency: Two pixels p and q with the values from set 'V' are 8-adjacent if q is in the set of $N_8(p)$.

e.g. V = { 1, 2}

0	1	1
0	2	0
0	0	1

p in RED color

q can be any value in GREEN color

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

(ii) q is in $N_D(p)$ & the set $N_4(p) \cap N_4(q)$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(i) b & c

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(i) b & c

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 I

Soln: b & c are m-adjacent.

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(ii) b & e

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$

e.g. V = { 1 }

(ii) b & e

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Soln: b & e are m-adjacent.

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

e.g. V = { 1 }

(iii) e & i

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_D(p)$ & the set $N_4(p) \cap N_4(q)$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iii) e & i

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_D(p)$ & the set $\underline{N_4(p) \cap N_4(q)}$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iii) e & i

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Soln: e & i are m-adjacent.

Adjacency, Connectivity

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

(ii) q is in $N_D(p)$ & the set $N_4(p) \cap N_4(q)$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iv) e & c

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

m-adjacency: Two pixels p and q with the values from set 'V' are m-adjacent if

(i) q is in $N_4(p)$ OR

(ii) q is in $N_D(p)$ & the set $N_4(p) \cap N_4(q)$ have no pixels whose values are from 'V'.

e.g. V = { 1 }

(iv) e & c

0 a	1 b	1 c
0 d	1 e	0 f
0 g	0 h	1 i

Soln: e & c are NOT m-adjacent.

Adjacency, Connectivity

Connectivity: 2 pixels are said to be connected if their exists a path between them.

Let 'S' represent subset of pixels in an image.

Two pixels p & q are said to be connected in 'S' if their exists a path between them consisting entirely of pixels in 'S'.

For any pixel p in S, the set of pixels that are connected to it in S is called a **connected component of S**.

Paths & Path lengths

- A *path* from pixel p with coordinates *(x, y)* to pixel *q* with coordinates *(s, t)* is a sequence of distinct pixels with coordinates:

  $(x_0, y_0), (x_1, y_1), (x_2, y_2) \dots (x_n, y_n)$,

  where $(x_0, y_0)=(x, y)$ and $(x_n, y_n)=(s, t)$;

  (x_i, y_i) is adjacent to (x_{i-1}, y_{i-1}) $\quad 1 \le i \le n$

- Here n is the *length* of the path.

- We can define 4-, 8-, and m-paths based on type of adjacency used.

Paths

Example # 1: Consider the image segment shown in figure. Compute length of the ***shortest-4, shortest-8 & shortest-m paths*** between pixels p & q where,

V = {1, 2}.

	4	2	3	2 q
	3	3	1	3
	2	3	2	2
p	2	1	2	3

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 →	1	2	3	

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 →	1 →	2	3	

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 →	1 →	2 ↑	3	

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-4 path:

V = {1, 2}.

$$\begin{array}{ccccccccc}
 & 4 & & 2 & & 3 & \times & 2 & q \\
 & 3 & & 3 & & 1 & & 3 & \\
 & & & & & \uparrow & & & \\
 & 2 & & 3 & & 2 & & 2 & \\
 & & & & & \uparrow & & & \\
p & 2 & \rightarrow & 1 & \rightarrow & 2 & & 3 &
\end{array}$$

So, Path does not exist.

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-8 path:

$V = \{1, 2\}$.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 $\rightarrow$	1	2	3	

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-8 path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

So, shortest-8 path = 4

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

$$
\begin{array}{cccccc}
 & 4 & 2 & 3 & 2 & q \\
 & 3 & 3 & 1 & 3 & \\
 & 2 & 3 & 2 & 2 & \\
p & 2 \rightarrow & 1 & 2 & 3 &
\end{array}
$$

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2 →	1 →	2	3	

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
			↑		
p	2 →	1 →	2	3	

Paths

<u>Example # 1</u>:

<u>Shortest-m path:</u>

V = {1, 2}.

$$\begin{array}{cccccc}
 & 4 & 2 & 3 & 2 & q \\
 & 3 & 3 & 1 & 3 & \\
 & & & \uparrow & & \\
 & 2 & 3 & 2 & 2 & \\
 & & & \uparrow & & \\
p & 2 \rightarrow & 1 \rightarrow & 2 & 3 &
\end{array}$$

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

So, shortest-m path = 5

Paths

Example # 1:

Shortest-m path:

V = {1, 2}.

	4	2	3	2	q
	3	3	1	3	
	2	3	2	2	
p	2	1	2	3	

So, shortest-m path = 5

Connected Components

- If *p* and *q* are pixels of an image subset S then *p* is ***connected*** to *q* in S if there is a path from *p* to *q* consisting entirely of pixels in S.
- For every pixel *p* in S, the set of pixels in S that are connected to *p* is called a ***connected component*** of S.
- If S has only one connected component then S is called ***Connected Set.***

Regions and Boundaries

- A subset R of pixels in an image is called a *Region* of the image if R is a connected set.
- The *boundary* of the region R is the set of pixels in the region that have one or more neighbors that are not in R.

Distance measures

Given pixels p, q and **z** with coordinates (x, y), (s, t), (u, v) respectively, the distance function D has following properties:

a. $D(p, q) \geq 0$ [D(p, q) = 0, iff p = q]
b. $D(p, q) = D(q, p)$
c. $D(p, z) \leq D(p, q) + D(q, z)$

The following are the different Distance measures:

- **Euclidean Distance :**
  $D_e(p, q) = [(x-s)^2 + (y-t)^2]$

b. City Block Distance: →
$D_4(p, q) = |x-s| + |y-t|$

		2		
	2	1	2	
2	1	0	1	2
	2	1	2	
		2		

c. Chess Board Distance: →
$D_8(p, q) = max(|x-s|, |y-t|)$

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2