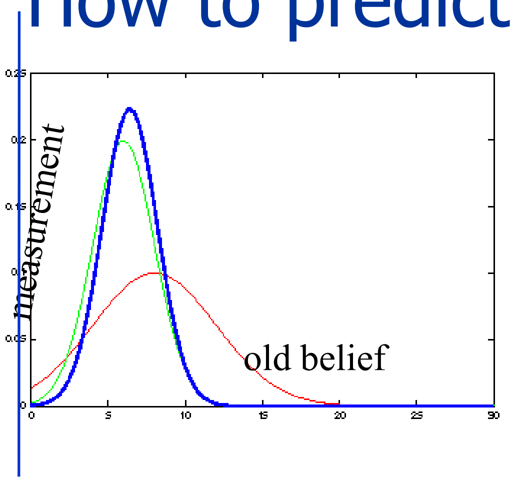
Recognition
Class 12

Kalman Filter

- Weighted sum: $\hat{x}_t = \hat{x}_t^- + K_t(z_t - H\hat{x}_t^-)$
- Questions:
  - How to set the weight between prediction and observations?
  - How to set the variance?
  - How to predict ?

$$k_G = \frac{E_{EST}}{E_{EST} + E_{MEA}}$$

$$E_{EST}(t) = (1 - k_G)E_{EST}(\text{t-1})$$



Time t-1 Time t

Simple Example

- Known system's linear dynamic model:
  - E.g., $x_t = \begin{pmatrix} p_t \\ \dot{p}_t \end{pmatrix}$,

    $\dot{p}_{t+1} = \dot{p}_t$ and $p_{t+1} = p_t + \Delta T \dot{p}_t + w_t$

$\dot{p}$ is velocity

- Known linear mapping between the state x_t and the observation, z_t:
  - E.g., $z_t = p_t + v_t$
- Measurement and estimation errors:
  - v_t and w_t are Gaussian noise

Assumptions

- Known system's linear dynamic model, with white noise $\sim G(0, Q)$

$$\boxed{x_t = Ax_{t-1} + Bu_t + w_{t-1}}$$

  - A is $n \times n$, B is $n \times \ell$
- Known linear transformation of the states to the measurements with white noise $\sim G(0, R)$: $\boxed{z_t = Hx_t + v_t}$
  - H is an $m \times n$ matrix

A, B, H, Q, R are assumed to be known

Goal

$$x_t = Ax_{t-1} + Bu_t + w_{t-1}, \quad w \in G(0, Q)$$
$$z_t = Hx_t + v_t, \quad v_t \in G(0, R)$$

- Compute x_t
- Given:
  - Initial state
  - Previous measurements
  - Known (or learned) linear models: A, B, H
- Minimize the error (its covariance) between the correct and the computed x_t

Notation

- A priori:
  - state estimation: $\hat{\boldsymbol{x}}_t^-$
  - error: $e_t^- = x_t - \hat{\boldsymbol{x}}_t^-$
  - covariance: $\Sigma_t^- = E(e_t^- \ e_t^{-T})$
- A posteriori, given z_t :
  - state estimation: $\hat{x}_t$
  - error: $e_t = (x_t - \hat{x}_t)$
  - covariance: $\Sigma_t = E(e_t \ e_t^T)$

We would like to minimize it

The Discrete Kalman Filter

Predict

(a priori estimate)

1. Predict the state ahead:

$$\hat{x}_t^- = A\hat{x}_{t-1}^- + Bu$$

2. Predict the error covariance ahead:

$$\Sigma_t^- = A\,\Sigma_{t-1}A^T + Q$$

Update

(a posteriori estimate)

1. Kalman gain K_t is:

$$k_t = \frac{E_{est}}{E_{est} + E_{mea}}$$

$$K_t = \Sigma_t^- H^T \left(H\Sigma_t^- H^T + R\right)^{-1}$$

2. Update the state estimate:

$$\hat{x}_t = \hat{x}_t^- + K_t(z_t - H\hat{x}_t^-)$$

3. Update the error covariance:

$$\Sigma_t = (\mathrm{I} - \mathrm{K}_t\mathrm{H})\Sigma_t^-$$

$$x_t = Ax_{t-1} + Bu_t + w_{t-1}, \quad w \in G(0, Q)$$

$$z_t = Hx_t + v_t, \quad v_t \in G(0, R)$$

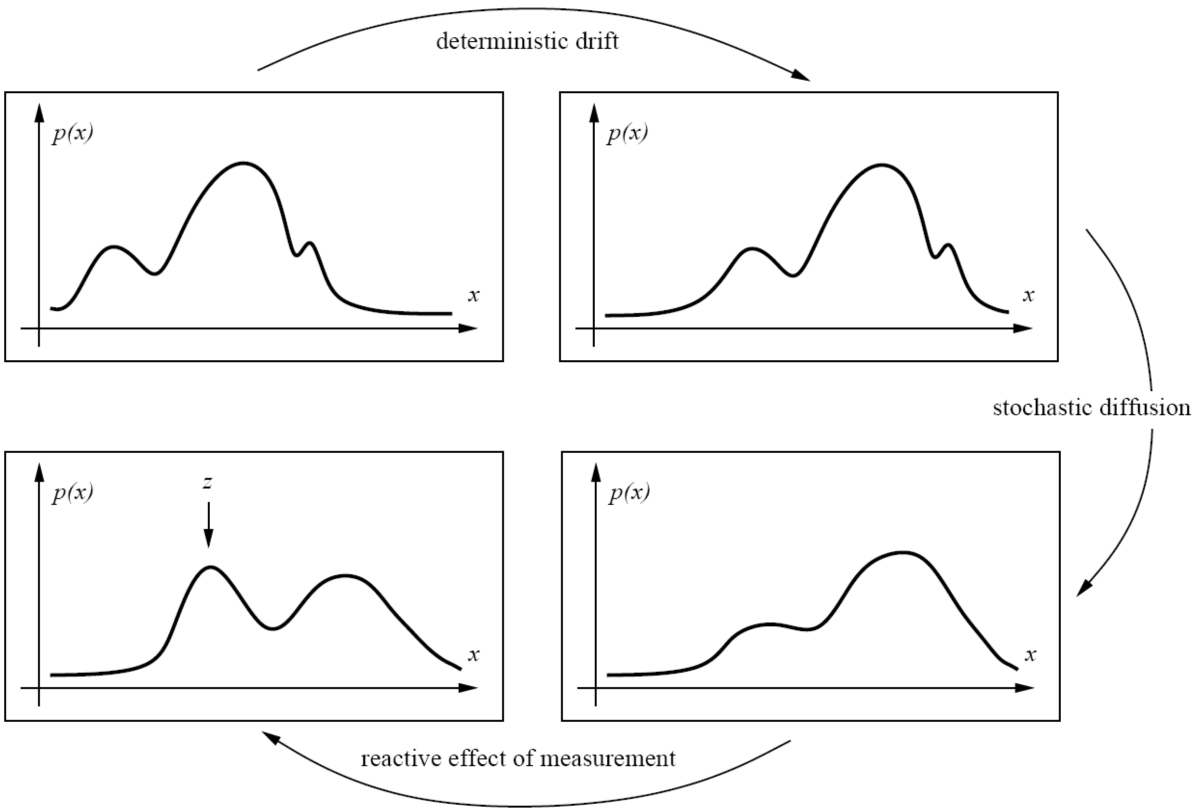
Up to Here

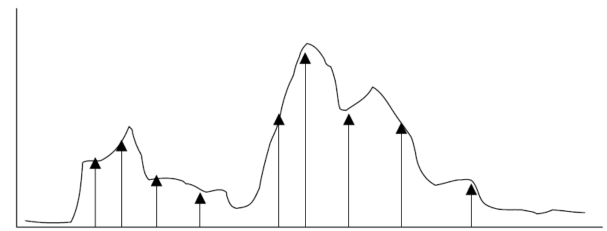
- Up to here: only brief outline of Kalman filter
- Many extensions exists
- Detailed: https://www.kalmanfilter.net/default.aspx
- See more references at the of the presentation

Non-Parametric Prediction

- Motivation: limitations of Kalman filter
- What are they?

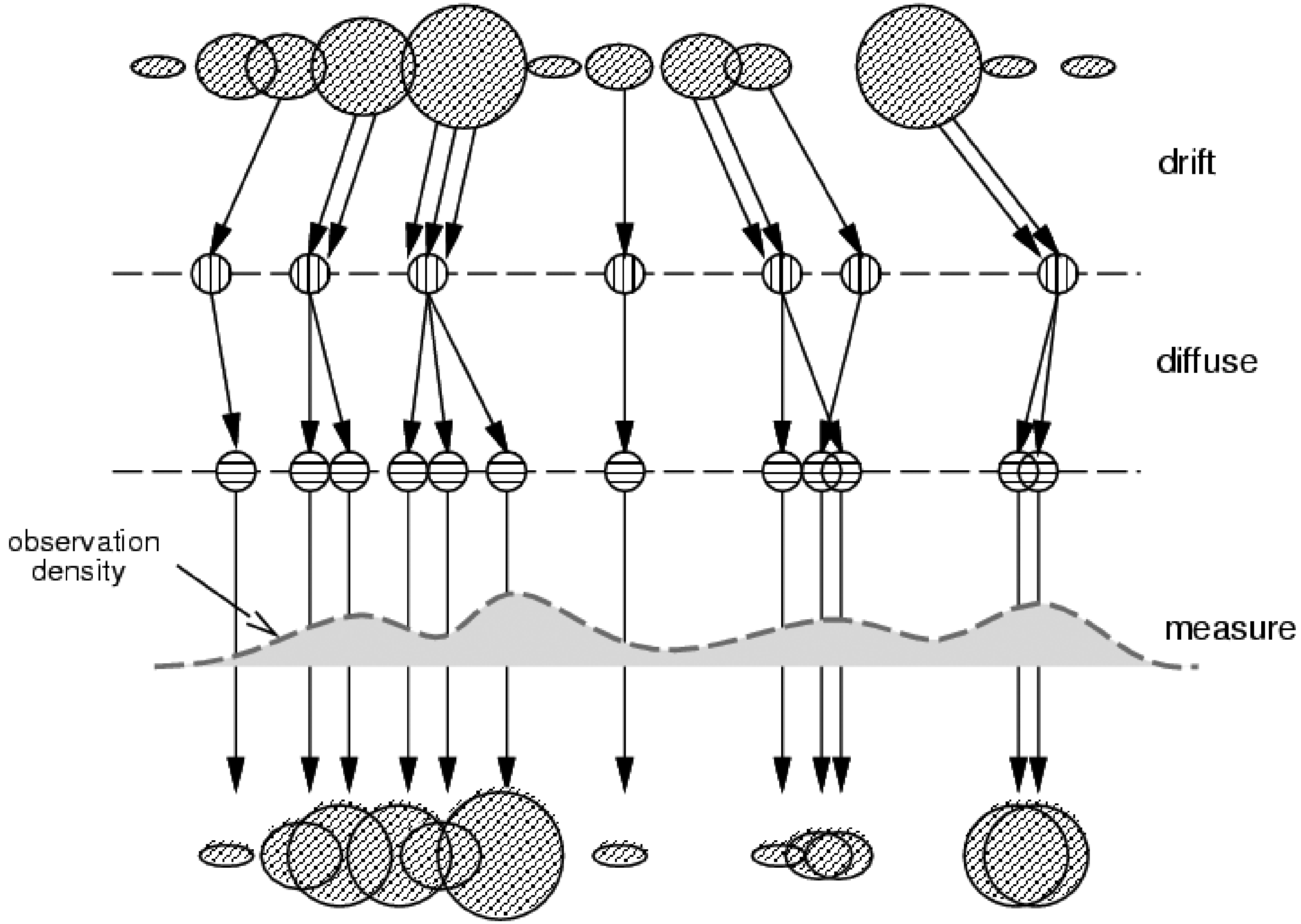
Particle Filters





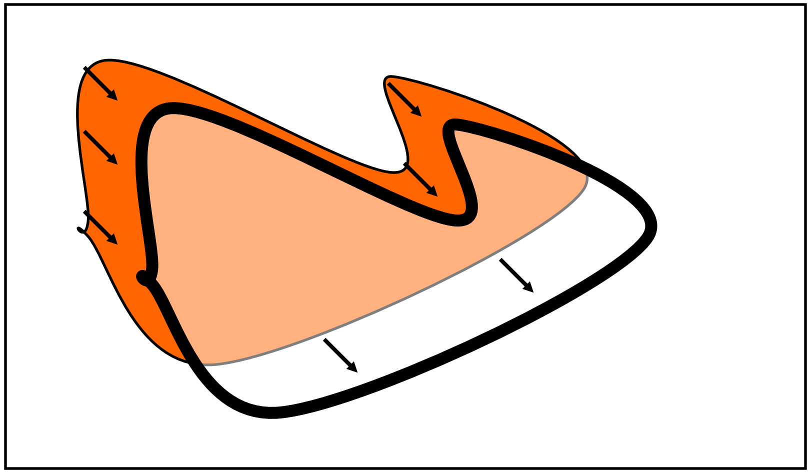
- Can represent distribution with set of weighted samples ("*particles*")
- Allows us to maintain **multiple hypotheses**

Particle Filters



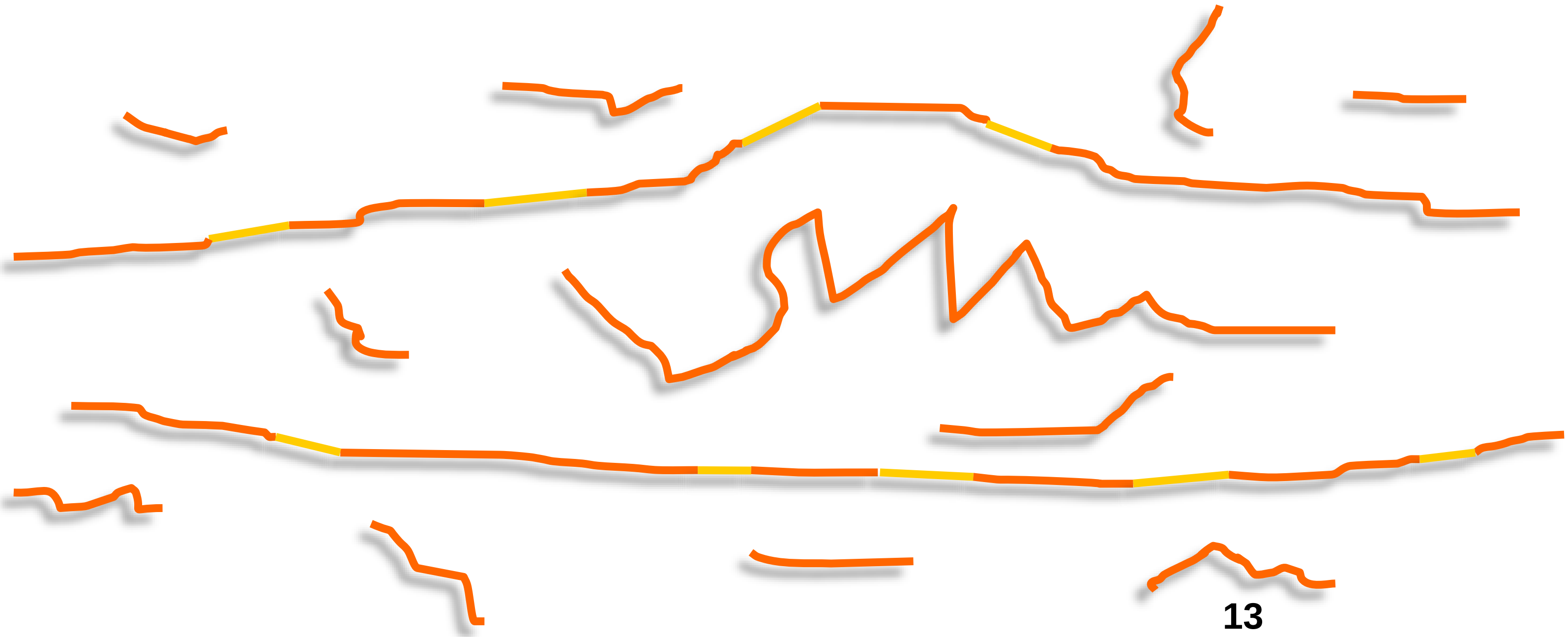
Contour Prediction

- Object model
- Object shape
- Video

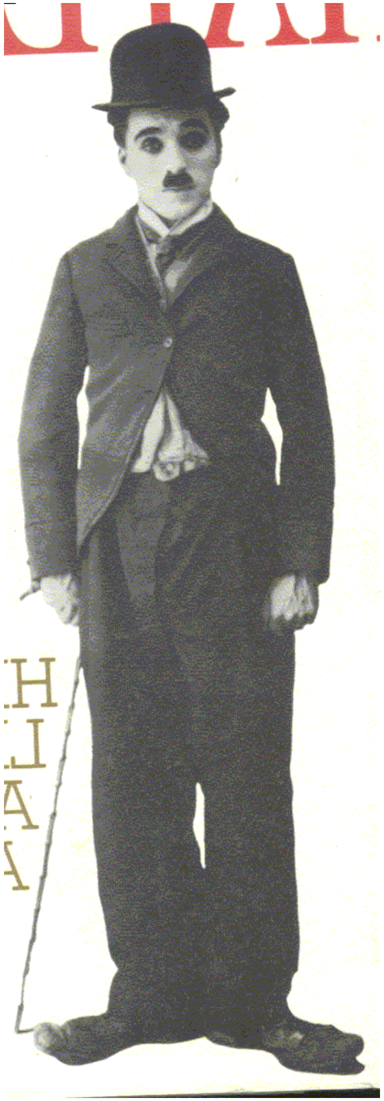


More on Tracking

- Tracklets
- Evaluation



Recognition



IDC HERZLIYA הבינתחומי הרצליה

What is Recognition

Learning:

See one or more images of a given object: Building a model

Recognition:

Recognize the object in a **novel** image: Determine the object (model) in the image

Simple Correlation

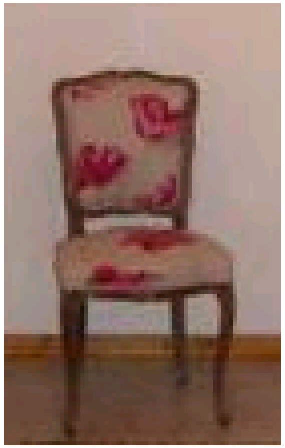
The model:

- An image of the object
- A set of images of the object

Recognition: compare a new image to each of the models' images

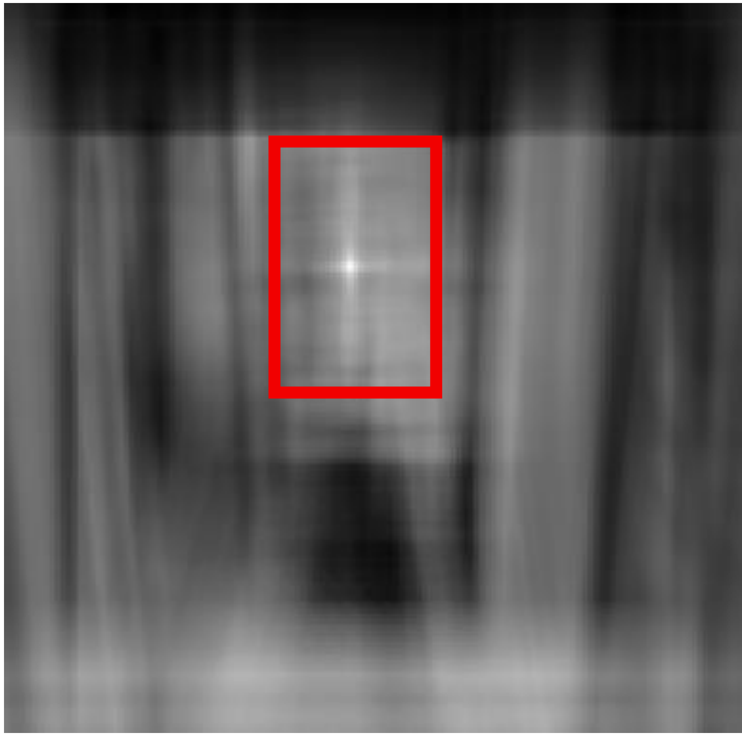
- Use nearest neighbor
- Use k-nearest neighbors

Identifiaction: Simple Correlation



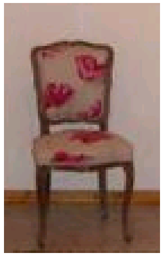
A chair

Find the chair

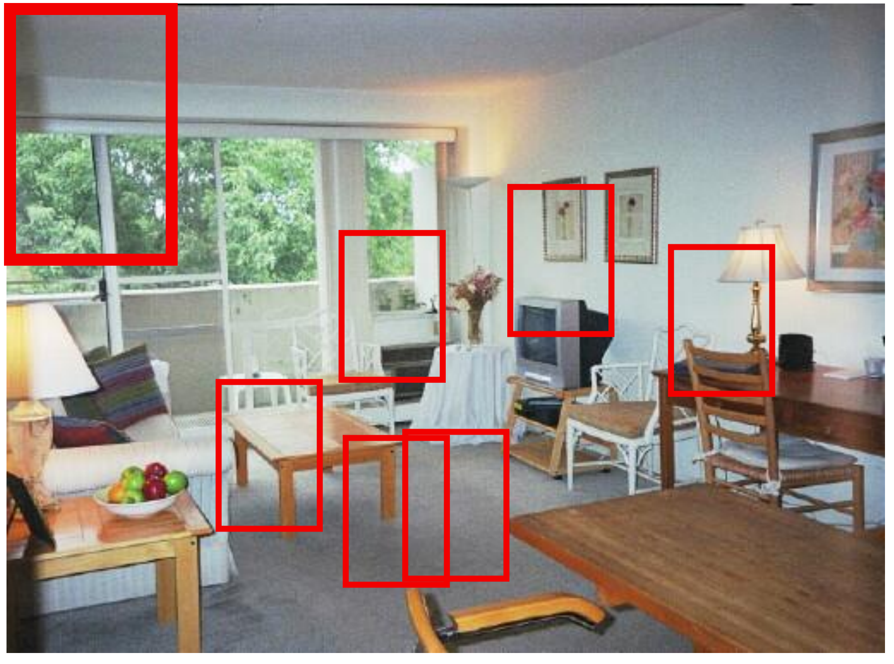


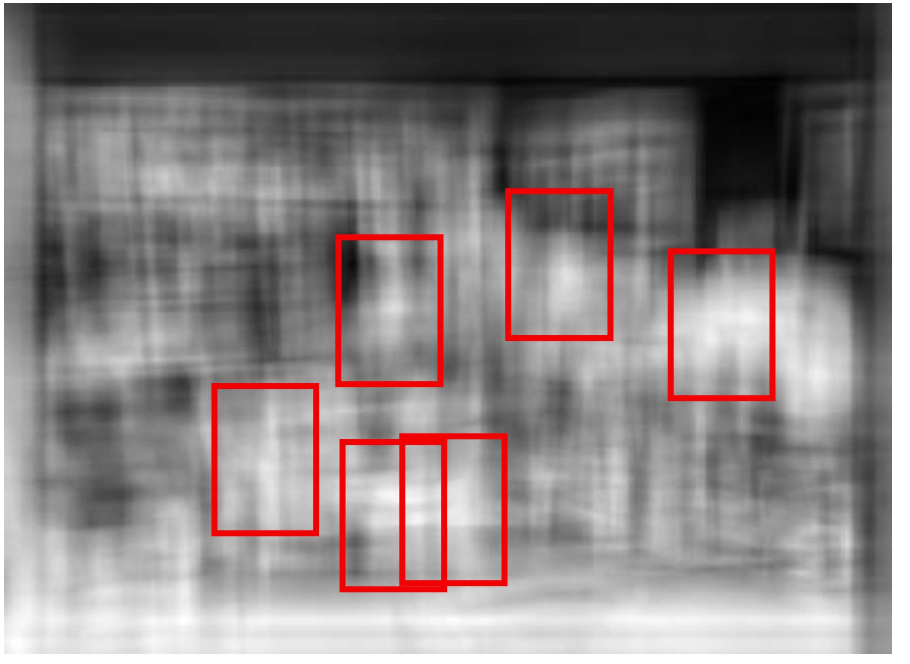
Output of normalized correlation

Adapted from Torralba



Classification: Simple Correlation





Simple template matching is not going to make it

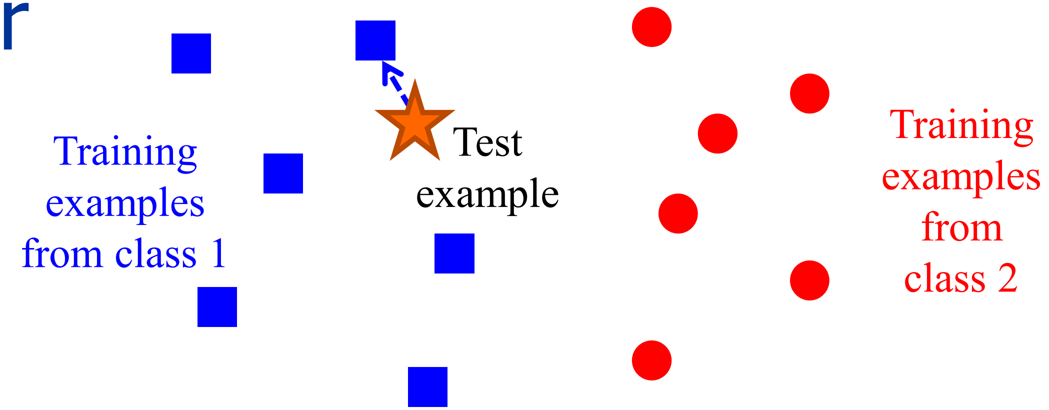
Adapted from Torralba

Classification Challenges

Simple Classification

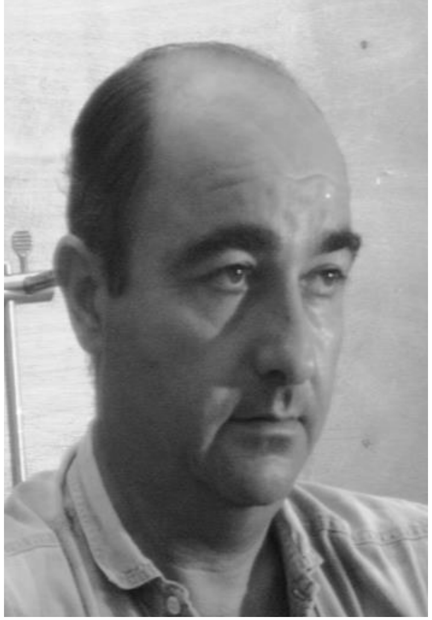
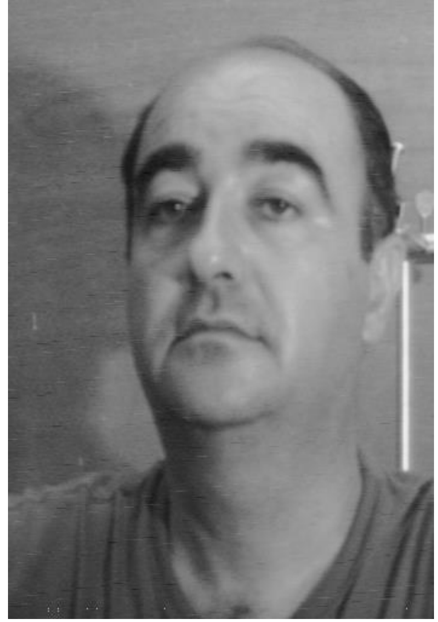
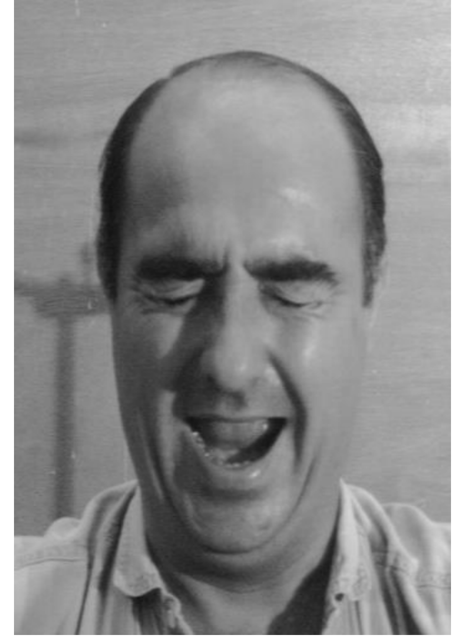
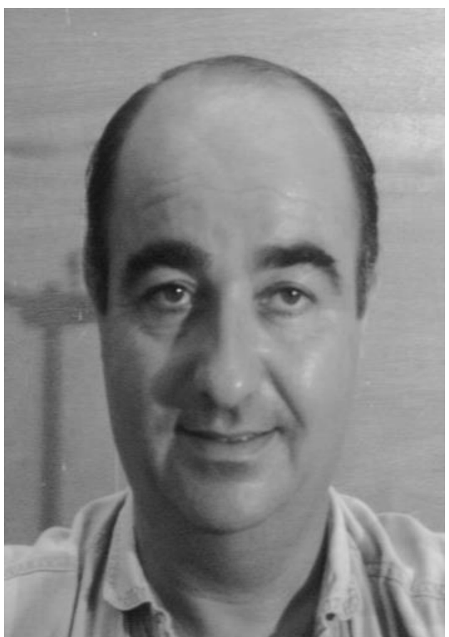
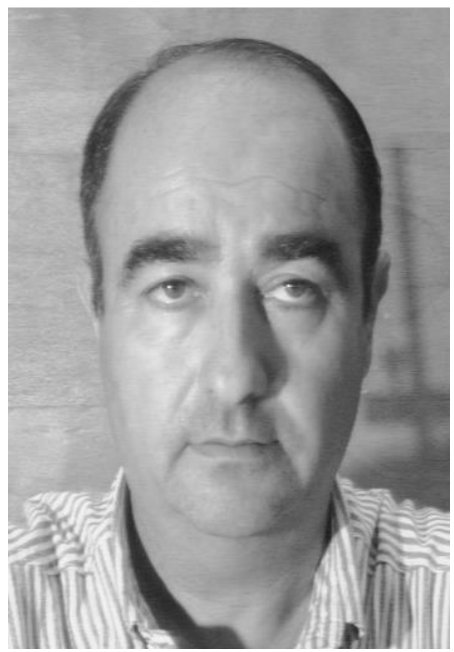
- Nearest neighbor

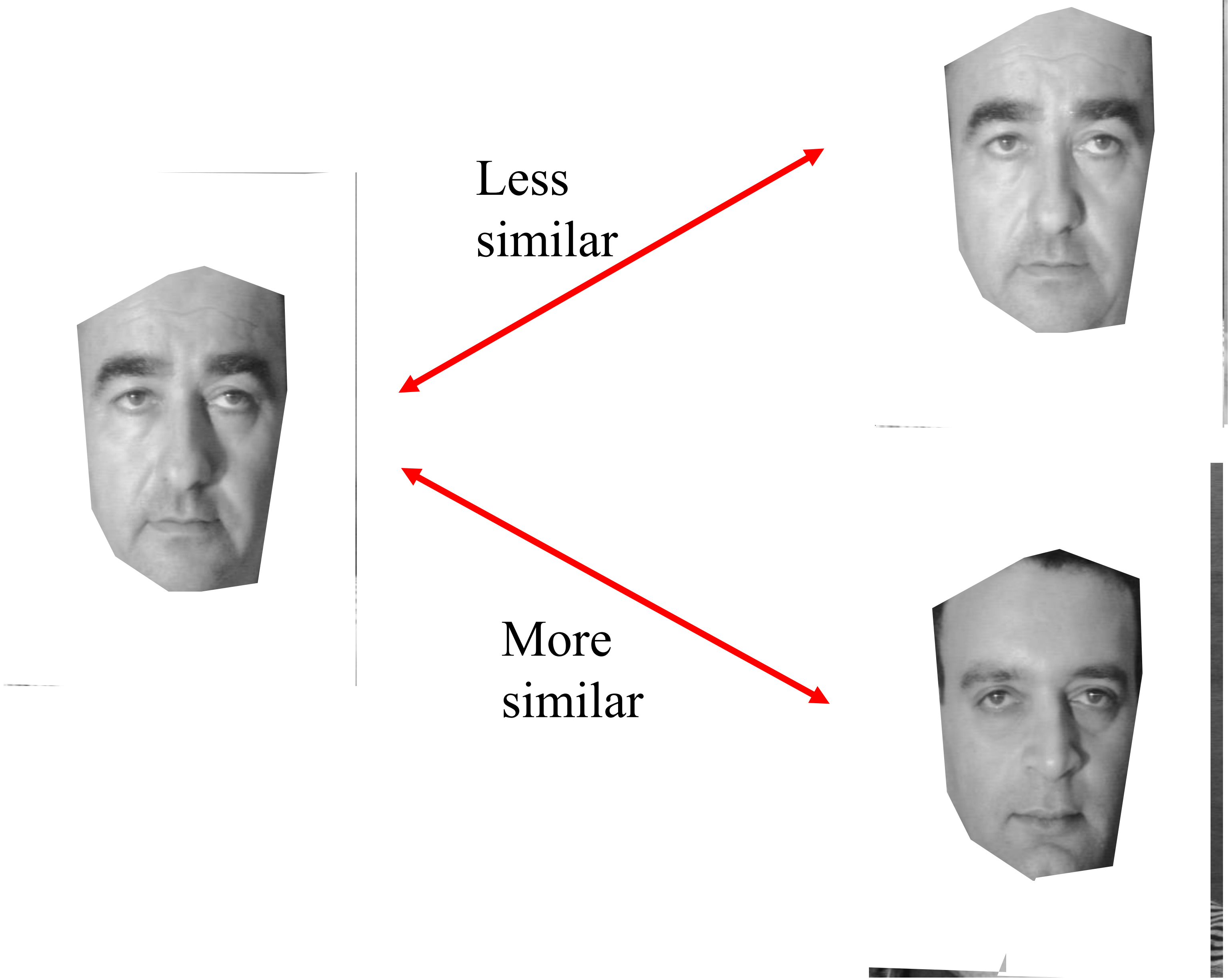
No training required!



- Pros:
  + Simple to implement
  + Can be any distribution
  + Works for any number of classes
  + *Nonparametric* method

- NN cons:
  - Need good distance function
  - Slow at test time



Less
similar
More
similar

הבינתחומי הרצליה
IDC HERZLIYA

Simple Correlation

- Expected to fail to generalize to new
  - Views
  - Illumination
  - Non rigid transformation
  - Occlusion
  - …

Main Challenges

- Generalization:
  Recognition in unseen images
- Robustness
- Scalability:
  - Space and time

Object Recognition Evolution

- Tailored end-to-end
  - Features (e.g., SIFTS, HOGS, RGB histograms,…)
  - Mapping (e.g., nearest neighbor, majority, …)
- Learning Half way
  - Features (e.g., BOW)
  - Mapping (e.g., SVM)
- Learning end-to-end
  - Features & mapping (e.g., neural networks)

Examples
Tailored End-to-End

- Simple image correlation using nearest neighbor or k-nearest neighbors

- Tailored facial feature representation (Yuillle 1991)

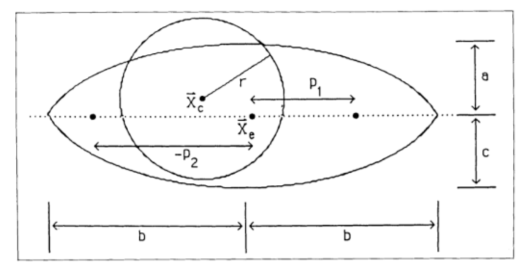
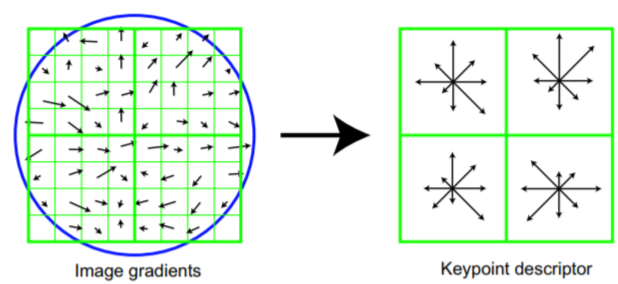


Figure 3. The eye template.

- General image features, e.g., SIFTS (Lowe 2004)



הבינתחומי הרצליה
IDC HERZLIYA

Machine Learning

- Use a set of observation to uncover the relation between input and the desired output
  - The alternative: study the problem mathematically
- Types:
  - Supervised/Semi-supervised/unsupervised

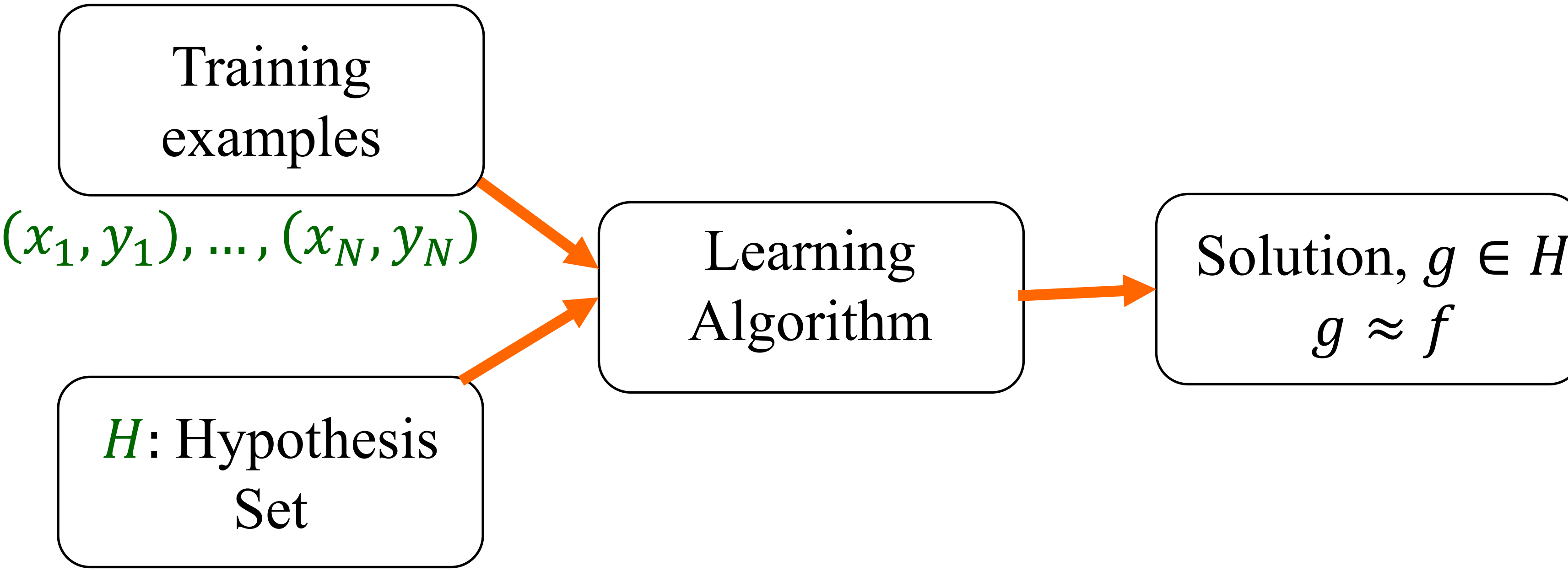
The Essences of Learning

- A pattern exists:
  - E.g., a picture of the same object has something in common
- Close mathematical solution is unknown
  - Otherwise – use it!
- Must have data
  - Labeled
  - Unlabeled

Components of General Learning

- Input: $x \in X$
- Output: $y \in Y$
- Target function: $f: X \to Y$
- Training examples: $(x_1, y_1), \dots, (x_N, y_N)$
- Hypothesis: $g: X \to Y, \quad$ s.t., $g \approx f$

Outline for Learning



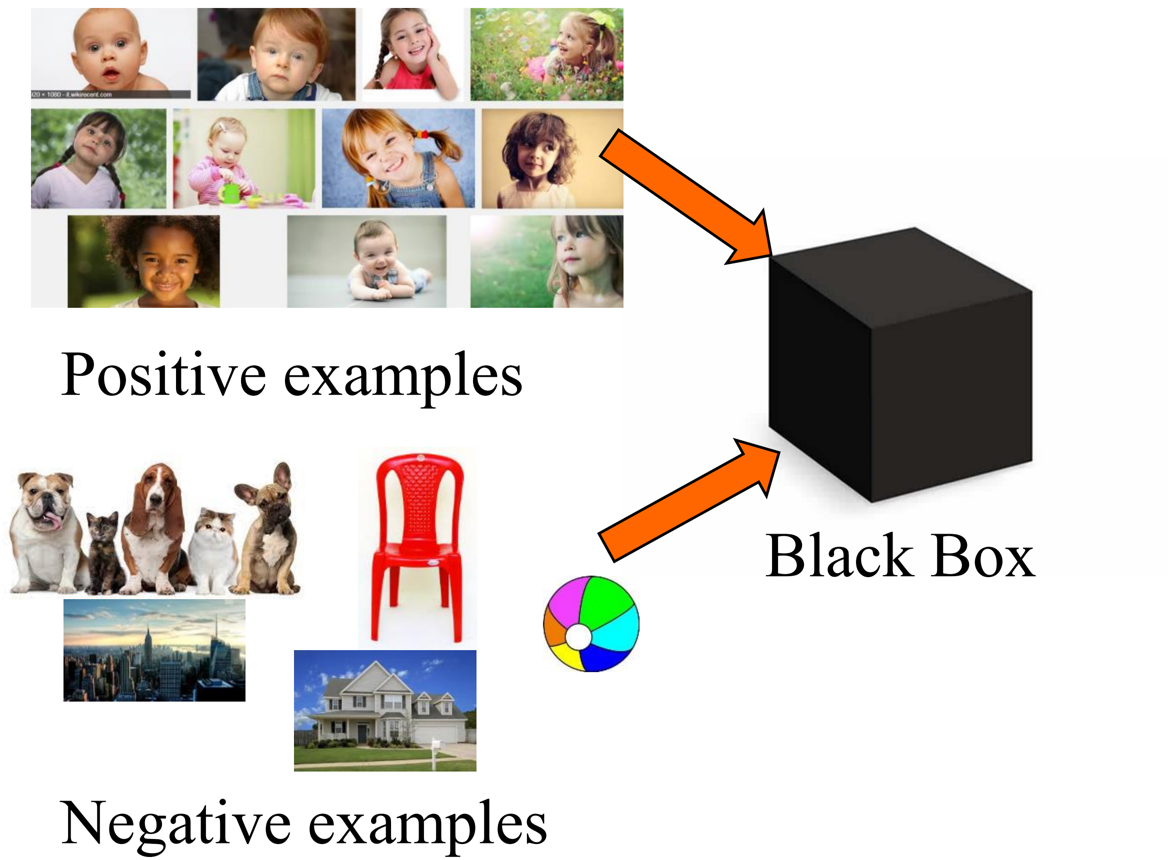
- Target function: $f: X \rightarrow Y$
- Find $g: X \rightarrow Y$, s.t., $g \approx f$

Object Recognition

- The function f: determine the object in the image
- Learn f from a set of labeled/unlabeled examples
- Use f to recognize objects in a new image

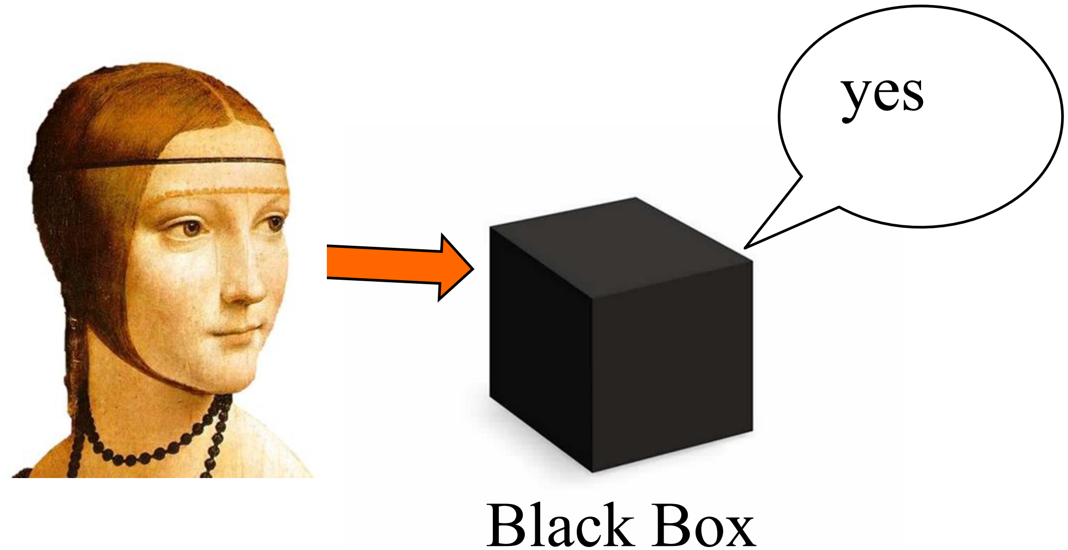
Supervised Learning

Training:



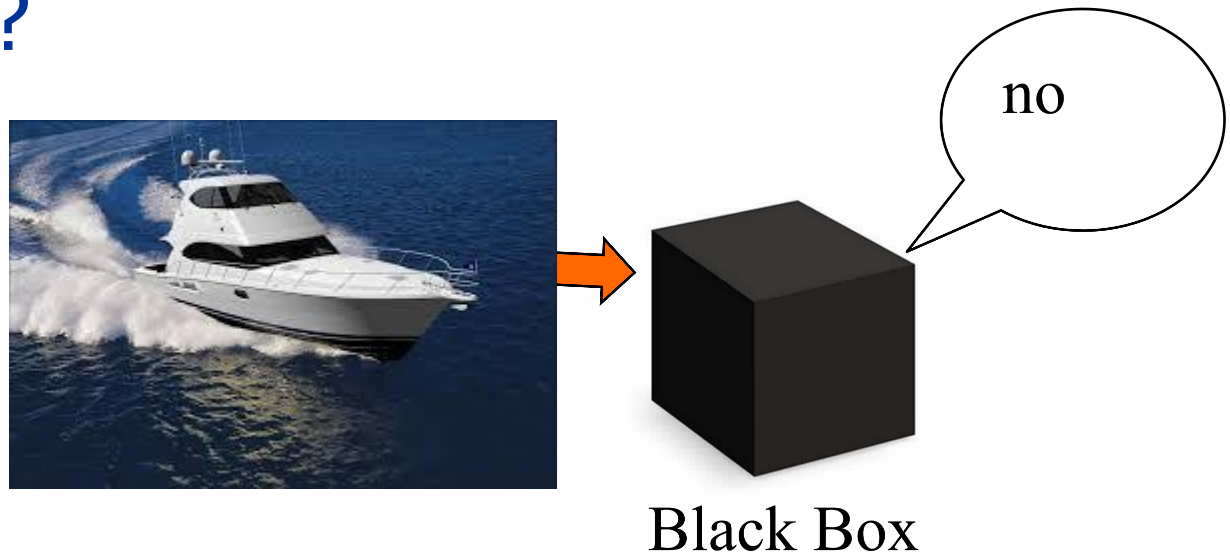
Supervised Learning

Testing: face?



Supervised Learning

Testing: face?

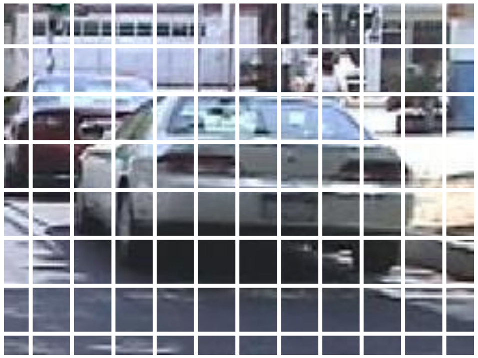


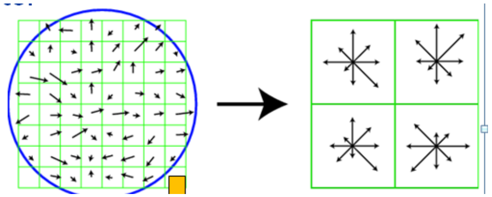
Learning Half Way

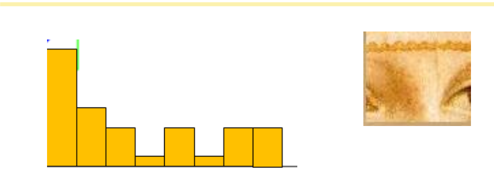
- Separate representation & matching
- Main Questions:
  - What is the representation
  - What is the "black box"

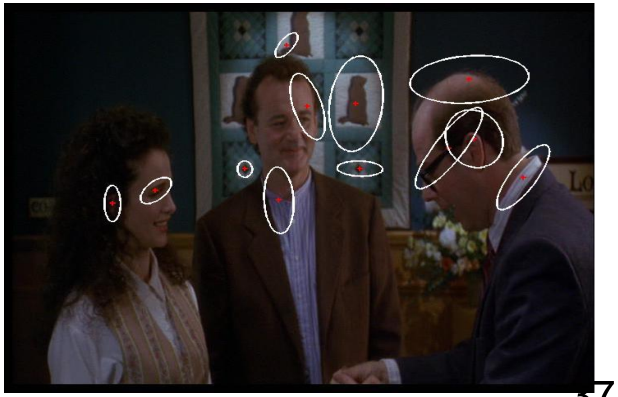
Tailored Representation

- The entire image
- Histogram
- Contours
- Features:
  - Regular grid
  - Local interest points
- Descriptors
  - Patches, Histograms, SIFT…









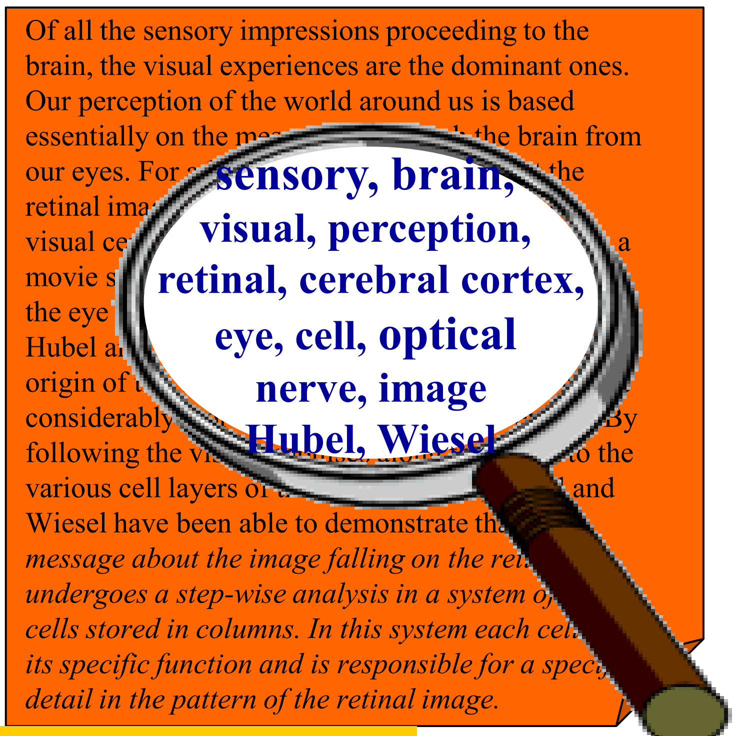
“Black Box”

- Examples:
  - Fisher classifier
  - SVM
  - Random Forest
  - …
  - Neural networks
  - Convolution neural network (CNN)

הבינתחומי הרצליה
IDC HERZLIYA

Bag of Words
(Sivic & Zisserman 2003)

- Use matching techniques from document analysis

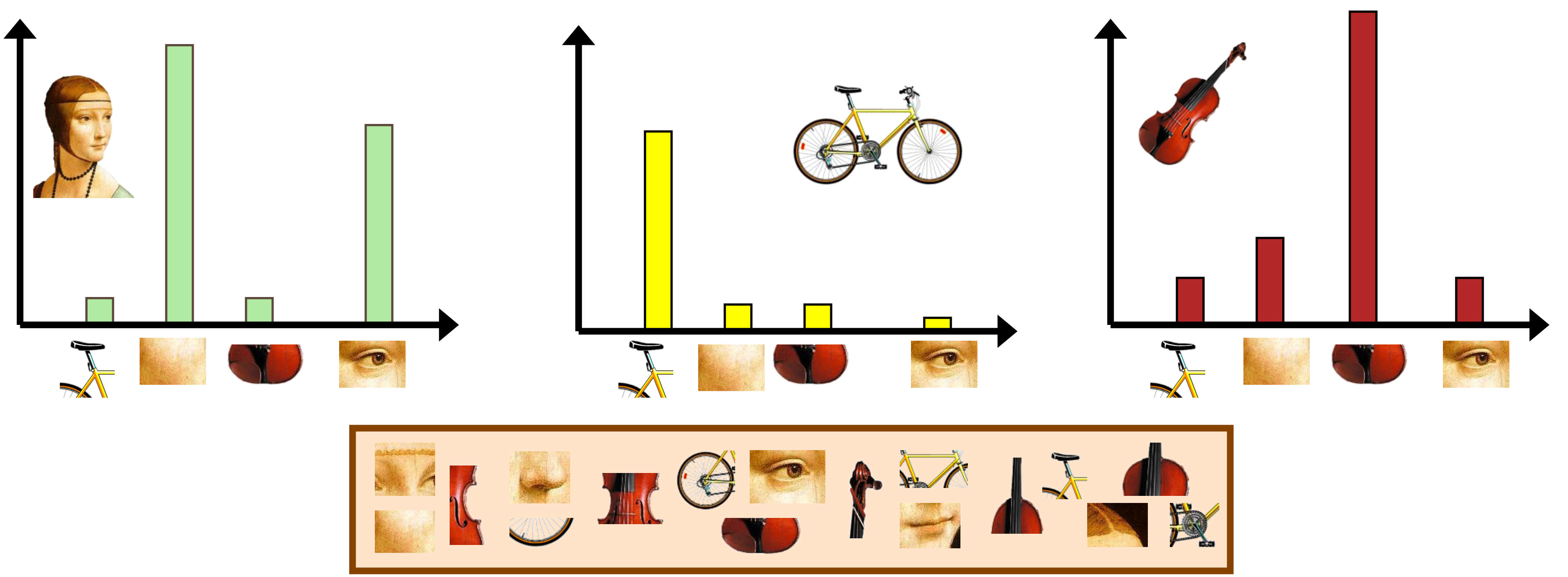


Object

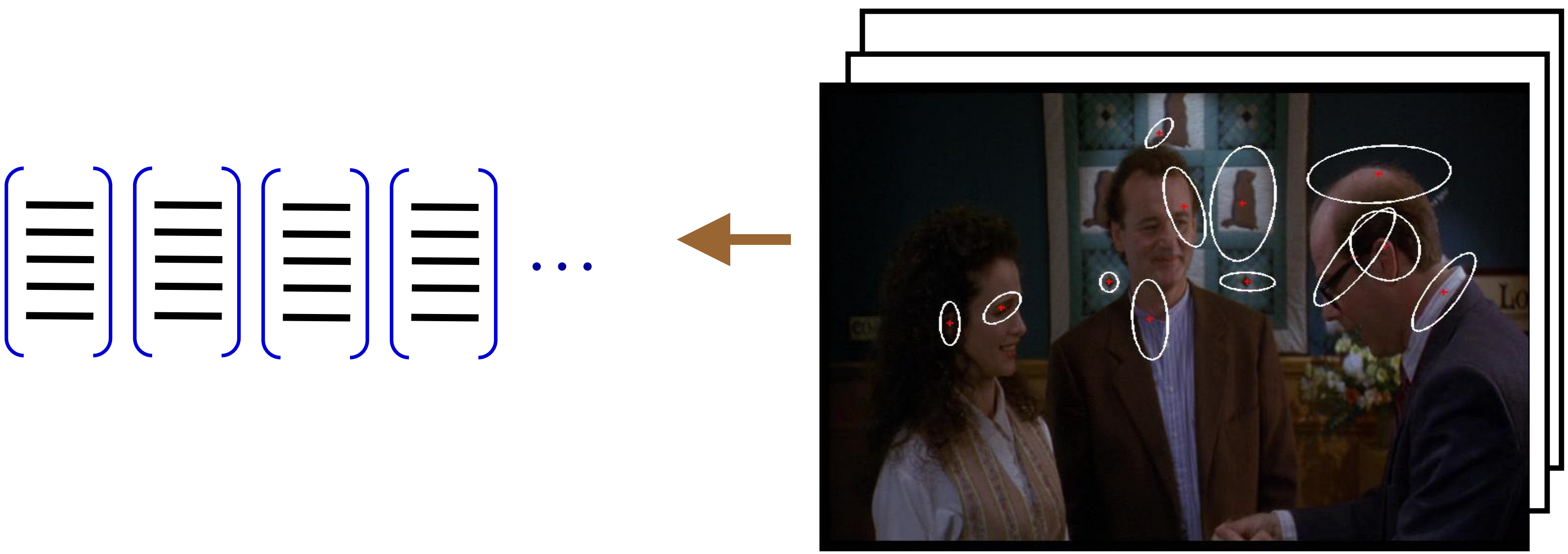
Bag of ‘words’

Bag of Words

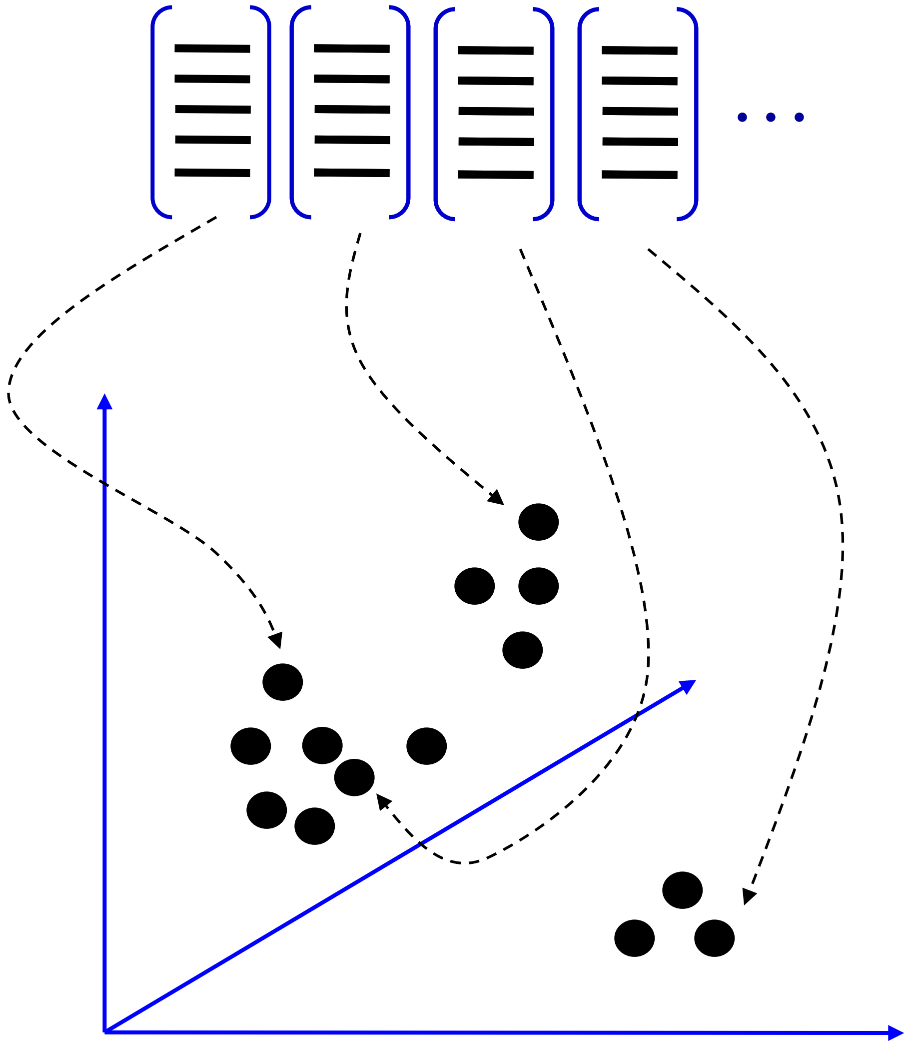
- Independent features
- Representation: histogram of features



Collect Features



2. Codewords dictionary formation



128-D SIFT space

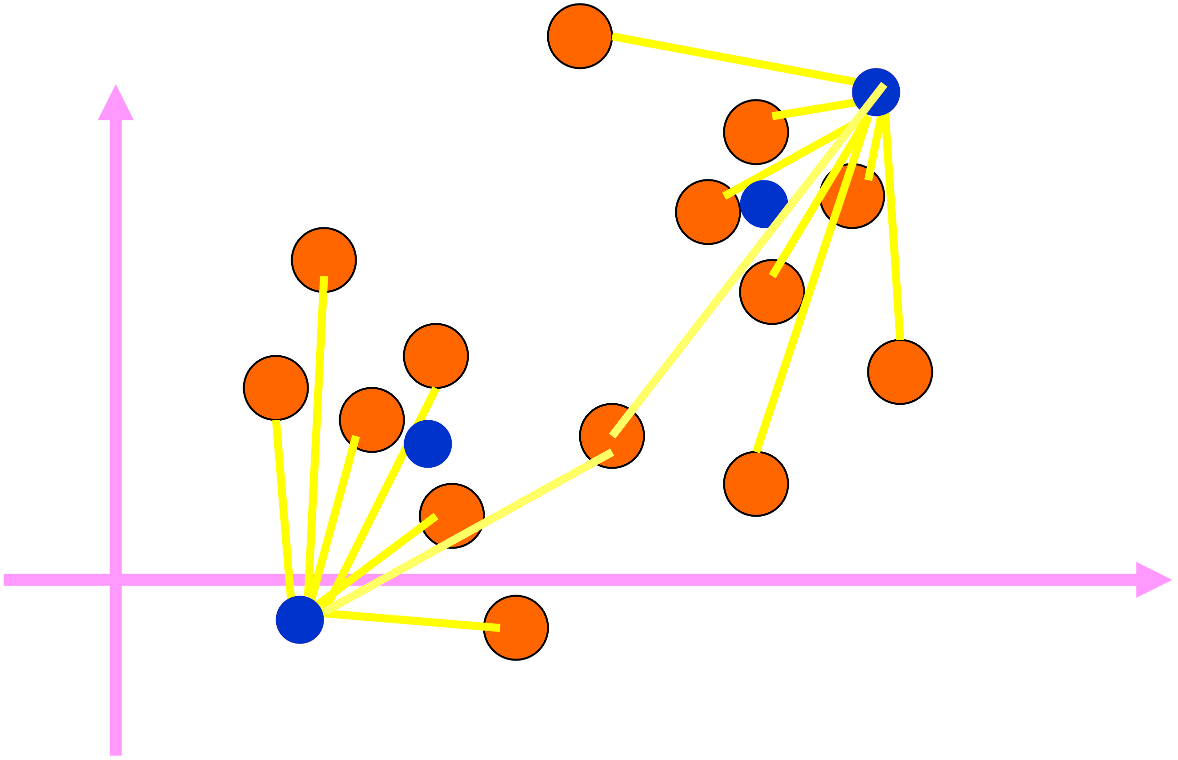
K-Means

- Choose a fixed number of clusters
- Find the centers of the clusters
- Minimize:

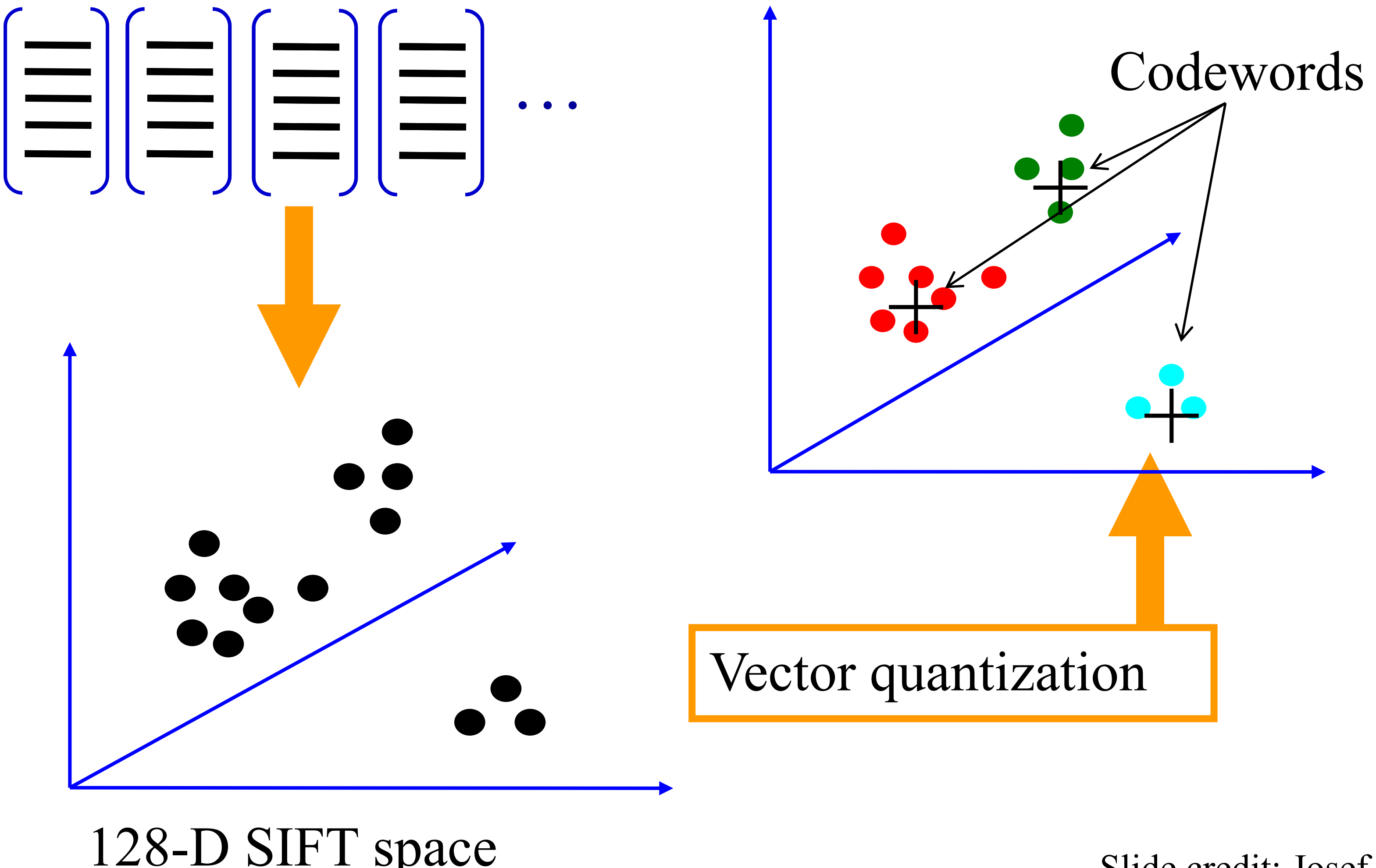
$$\sum_{i \in \text{clusters}} \left\{ \sum_{j \in \text{elements of i'th cluster}} \left\| x_j - \mu_i \right\|^2 \right\}$$

  where μ_i is the center of S_i
- Iterative Algorithm
  - Fix cluster centers; allocate points to closest cluster
  - Fix allocation; compute best cluster centers:

K-mean Clustering

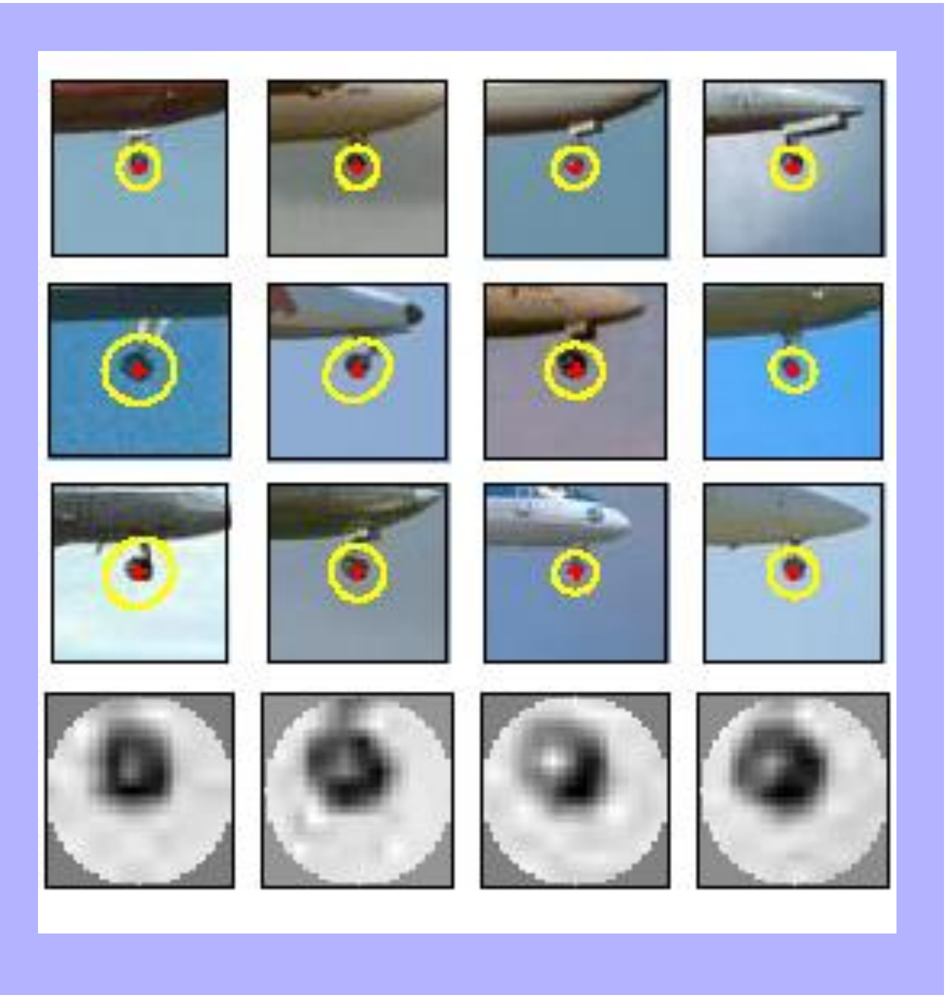


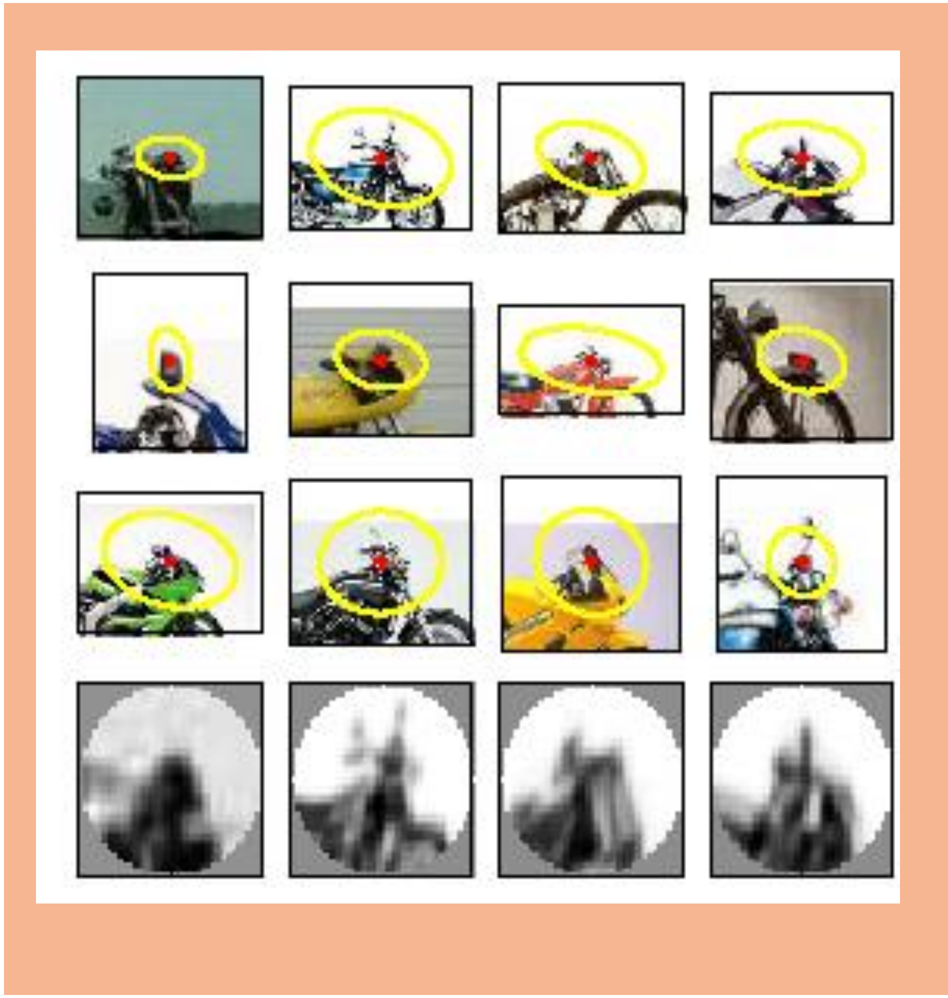
2. Codewords dictionary formation



Slide credit: Josef Sivic

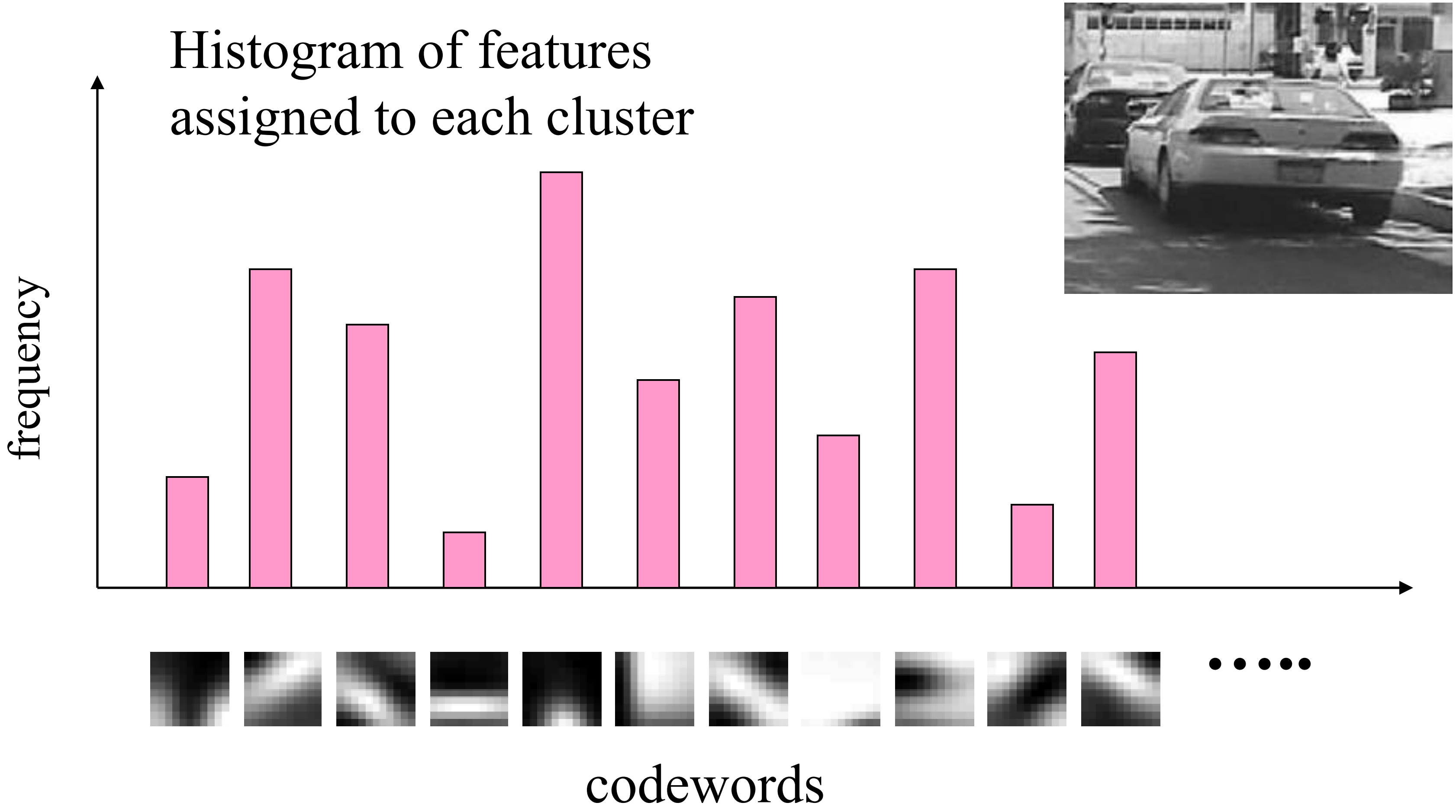
Image patch examples of codewords





Sivic et al. 2005

Image Representation



Weighted Histogram

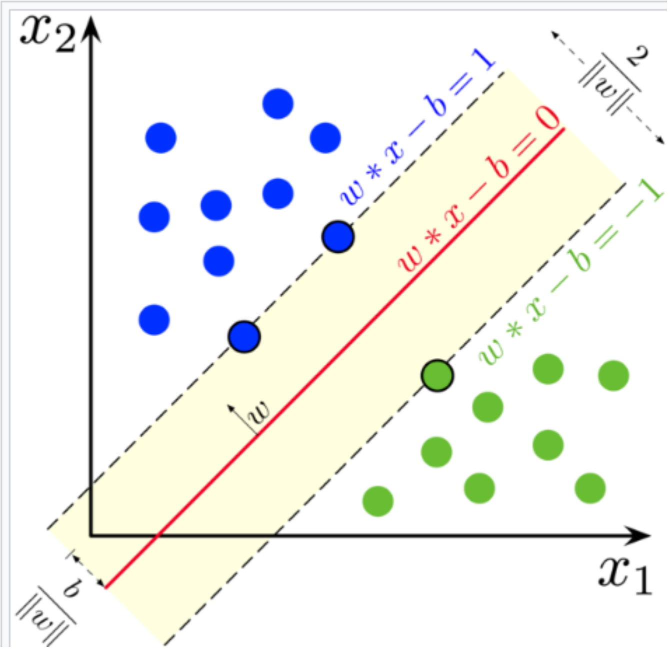
- Weight a word according to its frequency it all documents

Uses of BoW Representation

- Treat as feature vector for standard classifier
  - e.g., SVM
- Cluster BoW vectors over image collection
  - Discover visual themes
- Hierarchical models
  - Decompose scene/object

Support Vector Machin (SVM)

- Linear classifier



הבינתחומי הרצליה
IDC HERZLIYA

Limitations?

Other Classic Recognition Methods

- Geometry:
  - Invariance to view points
  - 3D alignment
  - Weak geometric constraints

Summary

- Goals of computer vision:
  - Infer the physical world from images
  - Impart human perceptual ability to machines
  - Improve over human perceptual ability

What we learned

- What are the challenges
- Tasks examples
- Example of solutions (algorithms)
- Principals of solutions
- Implementations: parallelism

Course Topics

- Image features:
  - edges, corners, other interest points
- Image formation: Geometry
- Geometry: Stereo, SFM, Homography, epipolar geometry
- Motion analysis:
  - Optical flow, change detection, tracking
- Object Recognition
  - BOW

Tools

- Convolution
- Algebra:
  - Projective Algebra, SVD, 2D/3D transformations, Use of EigenVectors,
- Gaussians and Mixture of Gaussians
- RANSAC
- Evaluation methods

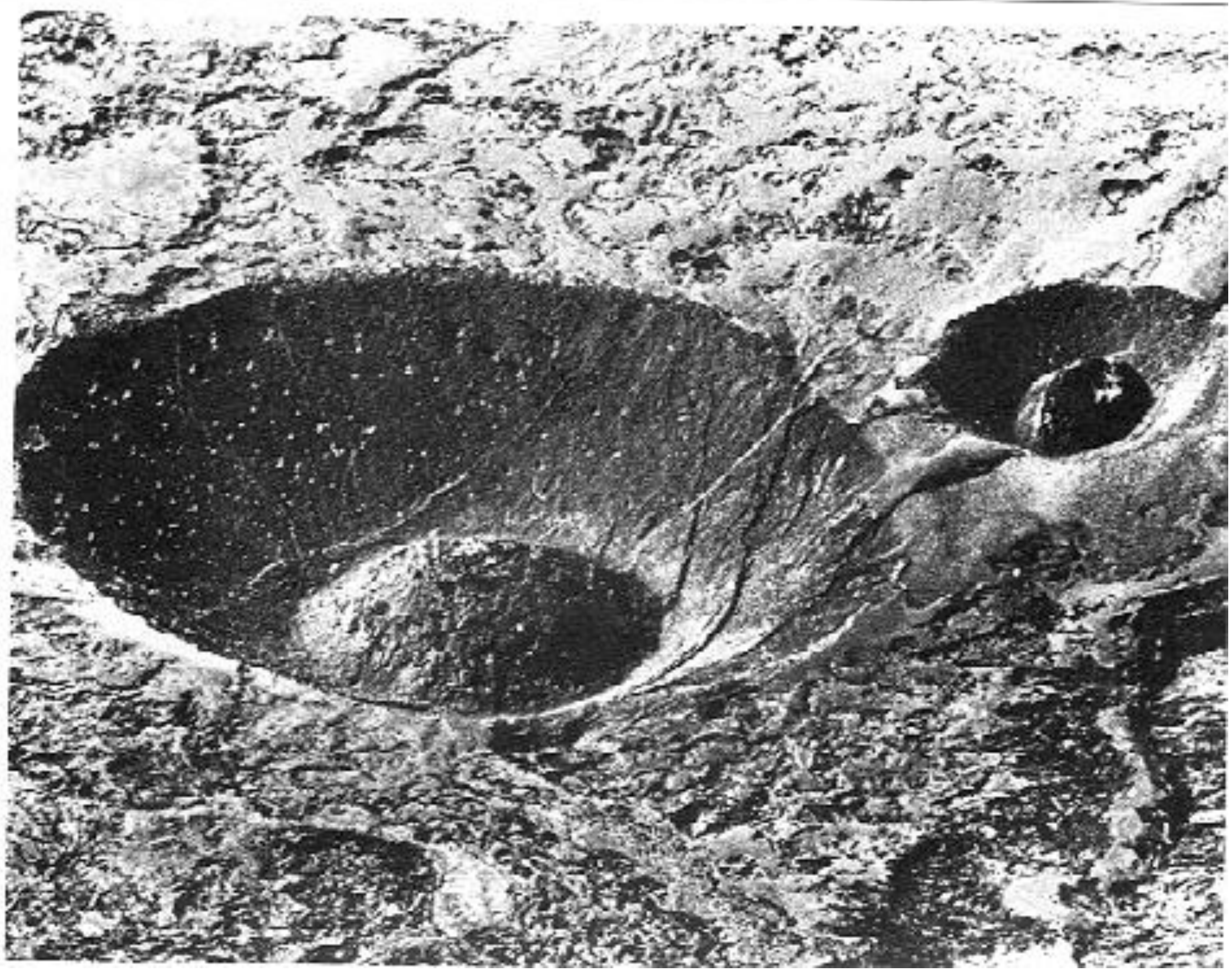
More Tools

- Double threshold
- RGB Histograms & histogram of gradients
- Nearest neighbor & Lowe distance
- Use vectorized solutions!
- …

Left for Additional Lectures

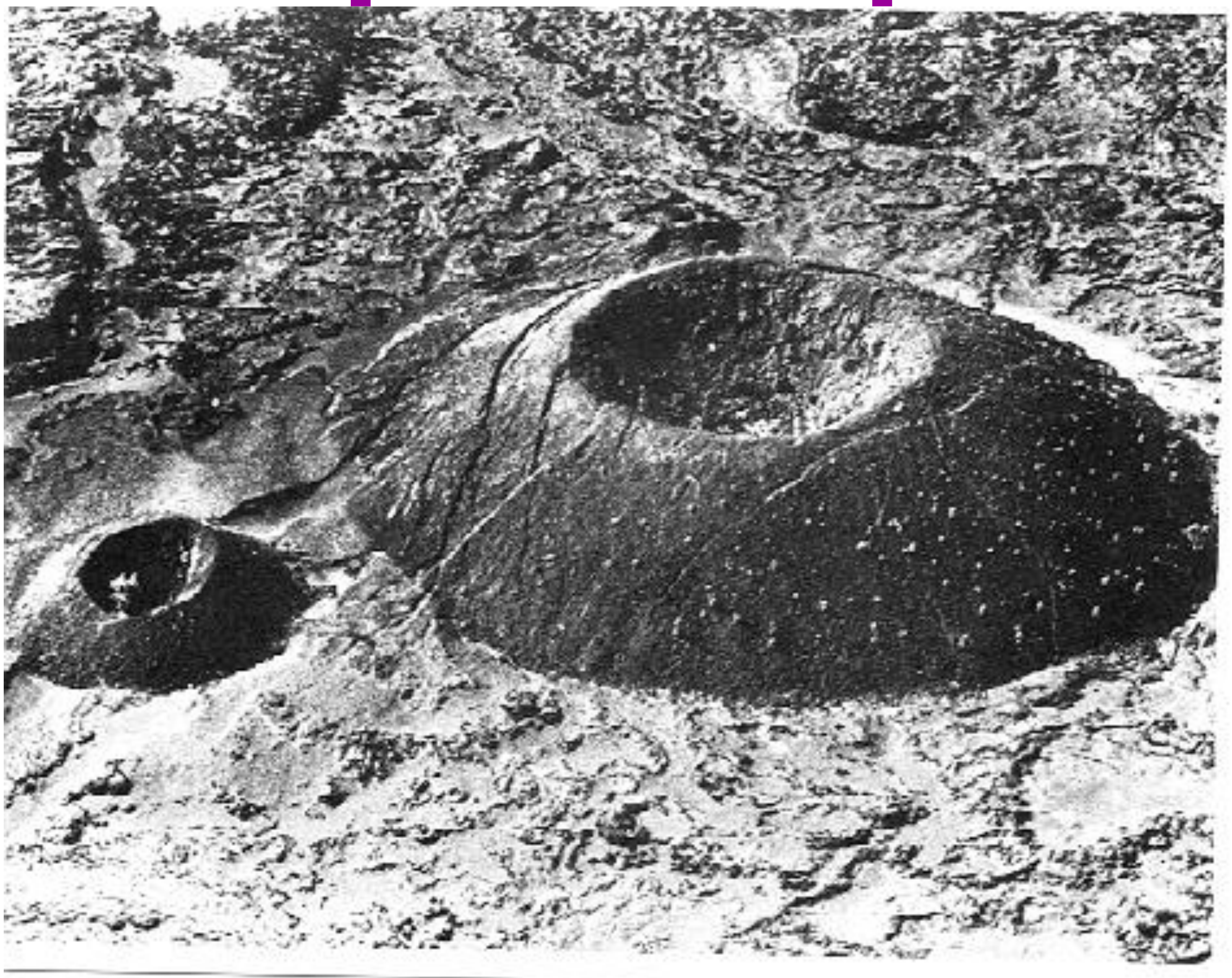
- More on object recognition
  - Supervised/unsupervised, AdaBoost, Cascade
- More on multi camera
- Segmentation
- Photometry:
  - Image formation
  - Photometric Stereo
- Introduction to CNN
  - state-of-the-art compared to classic methods

Depth Perception

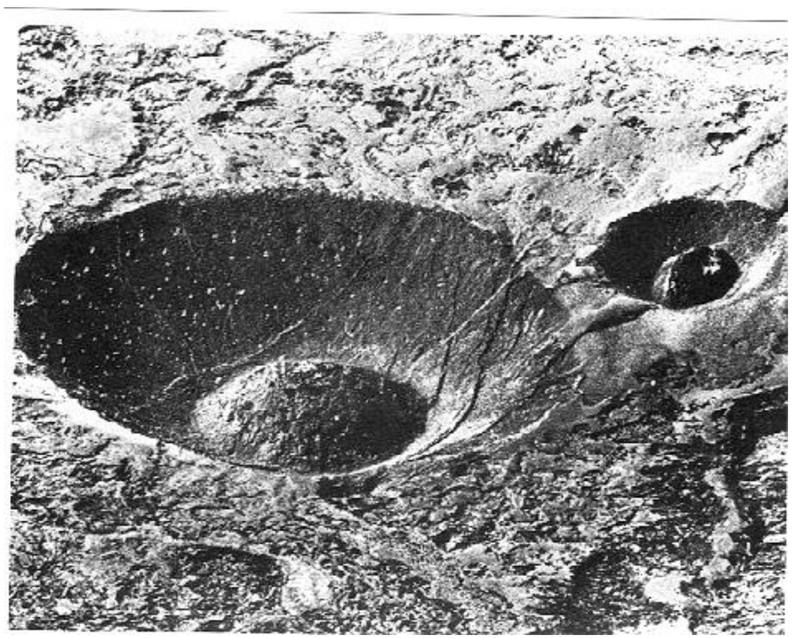


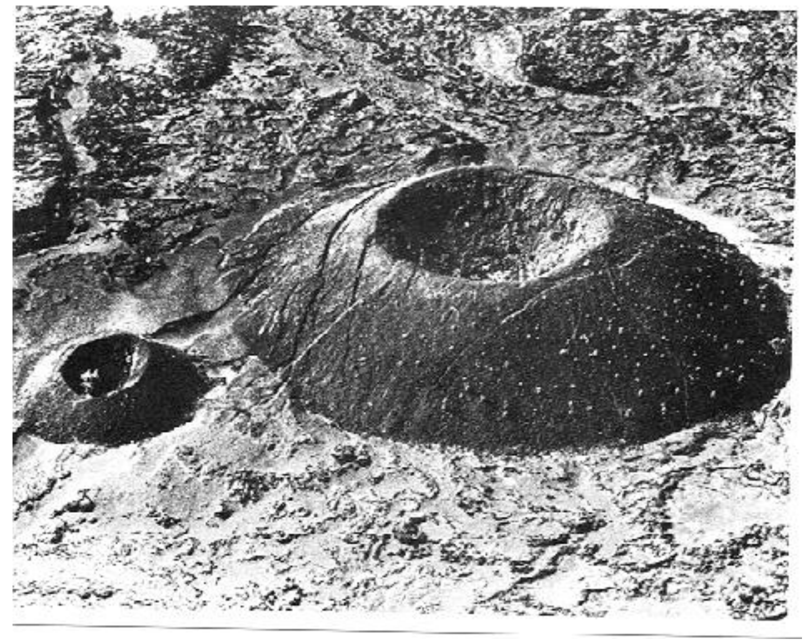
הבינתחומי הרצליה
IDC HERZLIYA

Depth Perception

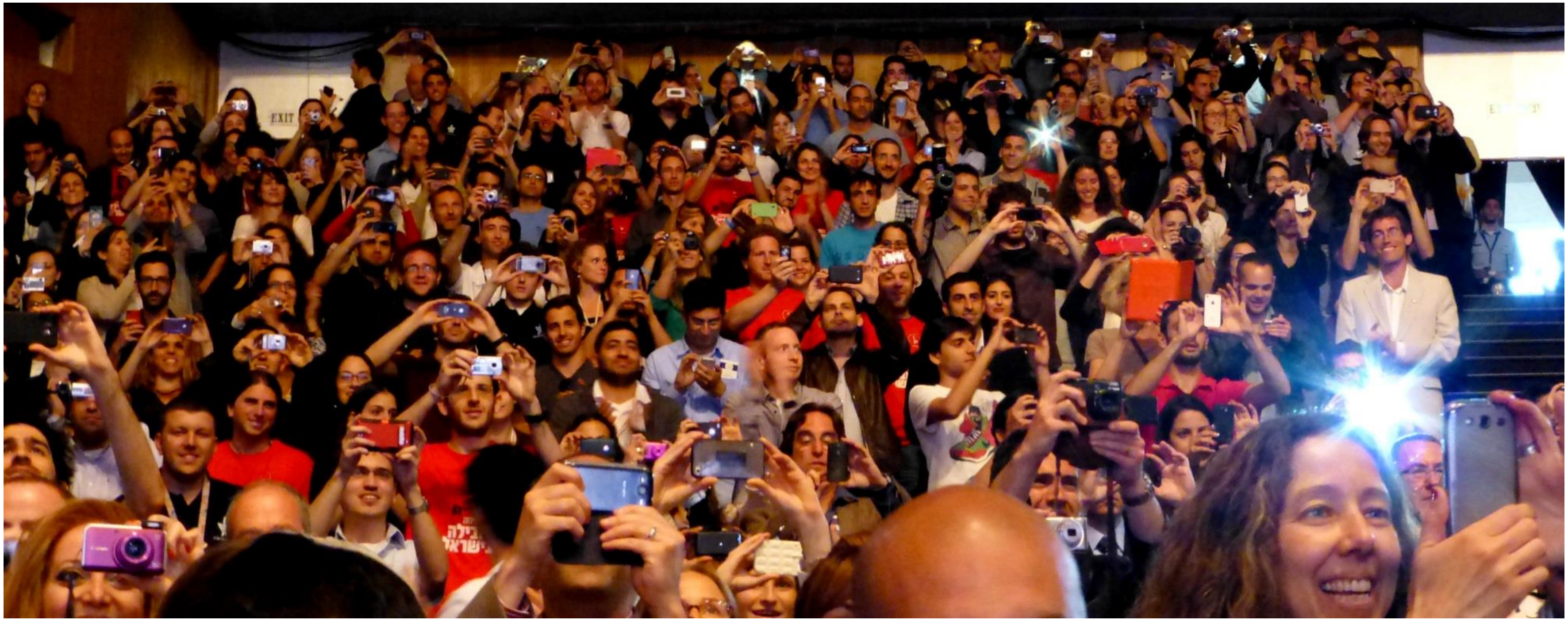


Shape from Shading is ill-pose





CrowdCam



Papal coronation

CrowdCam: Moving Object Segmentation

Depth Ambiguity

- Illusion

See you in the one-on-one meeting