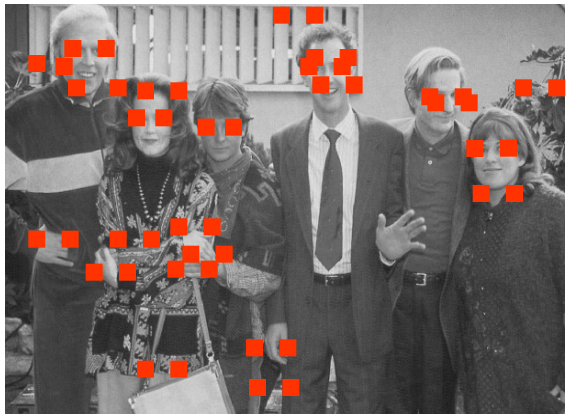
Introduction

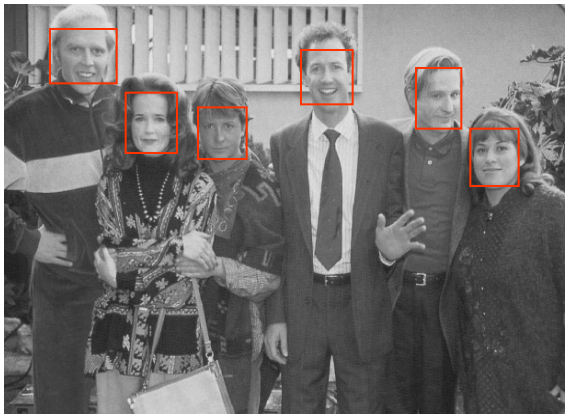
- #1 Face detection and recognition
- # 2 Image segmentation
- #3 Object classification
- #4 Stereo matching
- #5 Human computer interaction

Project 1

- Face detection
- Face recognition
- Specifications:
  - Study and present one method of face detection and one of face recognition
  - Using OpenCV to generate one face recognition system to recognize one face from web camera.

Face Detection System





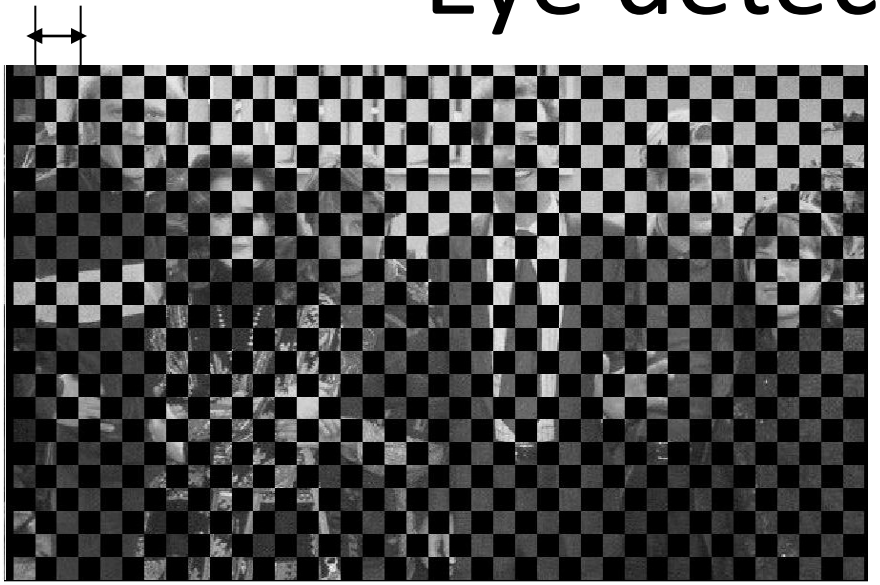
Input images → Eye detection → Face-candidate positions → Face detection → Result

Iris features

Adaboost + Haar basic functions

Reducing calculation time by decreasing the input number of the Adaboost algorithm considerably.

Eye detection



M x N : size of face

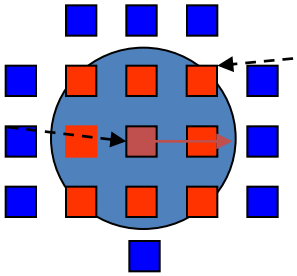
K = Max([M/4],[N/4])

K x K : size of eye region

ds : the golden ratio of K

$$(\approx 0.618 \times K)$$

Each block KxK → Iris feature detection → Positions of eye candidates

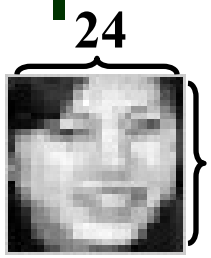


Selected candidate of left eye

Candidates for checking right eye

Candidate of face: both left and right candidates of eyes are selected

Face detection using AdaBoost



d_1

d_2

d_{1000}

AdaBoost method

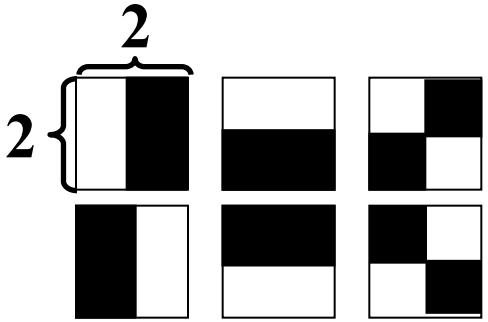
$$H(d_i) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(d_i)\right)$$

Strong classifier $H(d_i)$

Weak classifier

$h_t(d_i)$

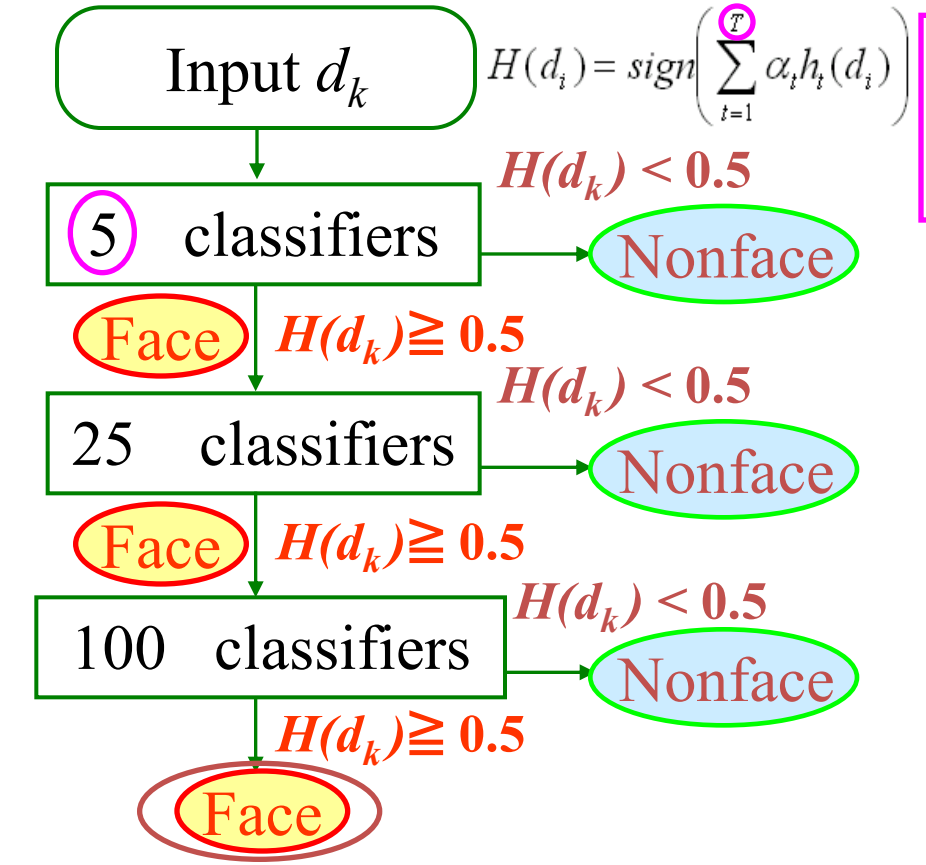
Basic Haar functions



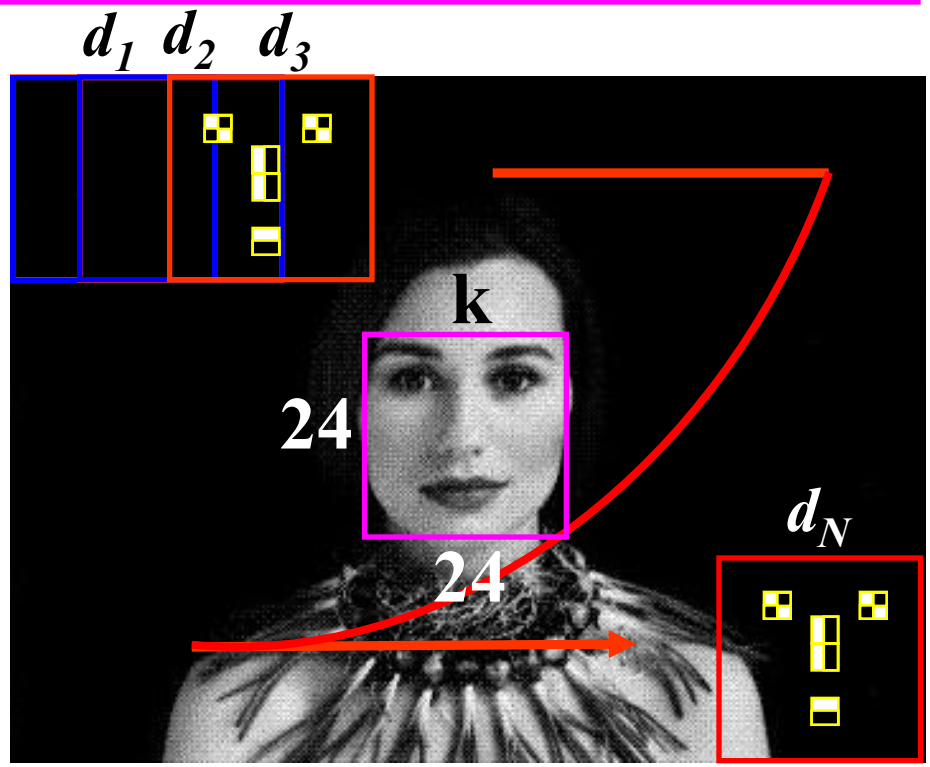
Test on the CMU facial database

AdaBoost cascade

Improve calculation speed by using cascade layers[1]



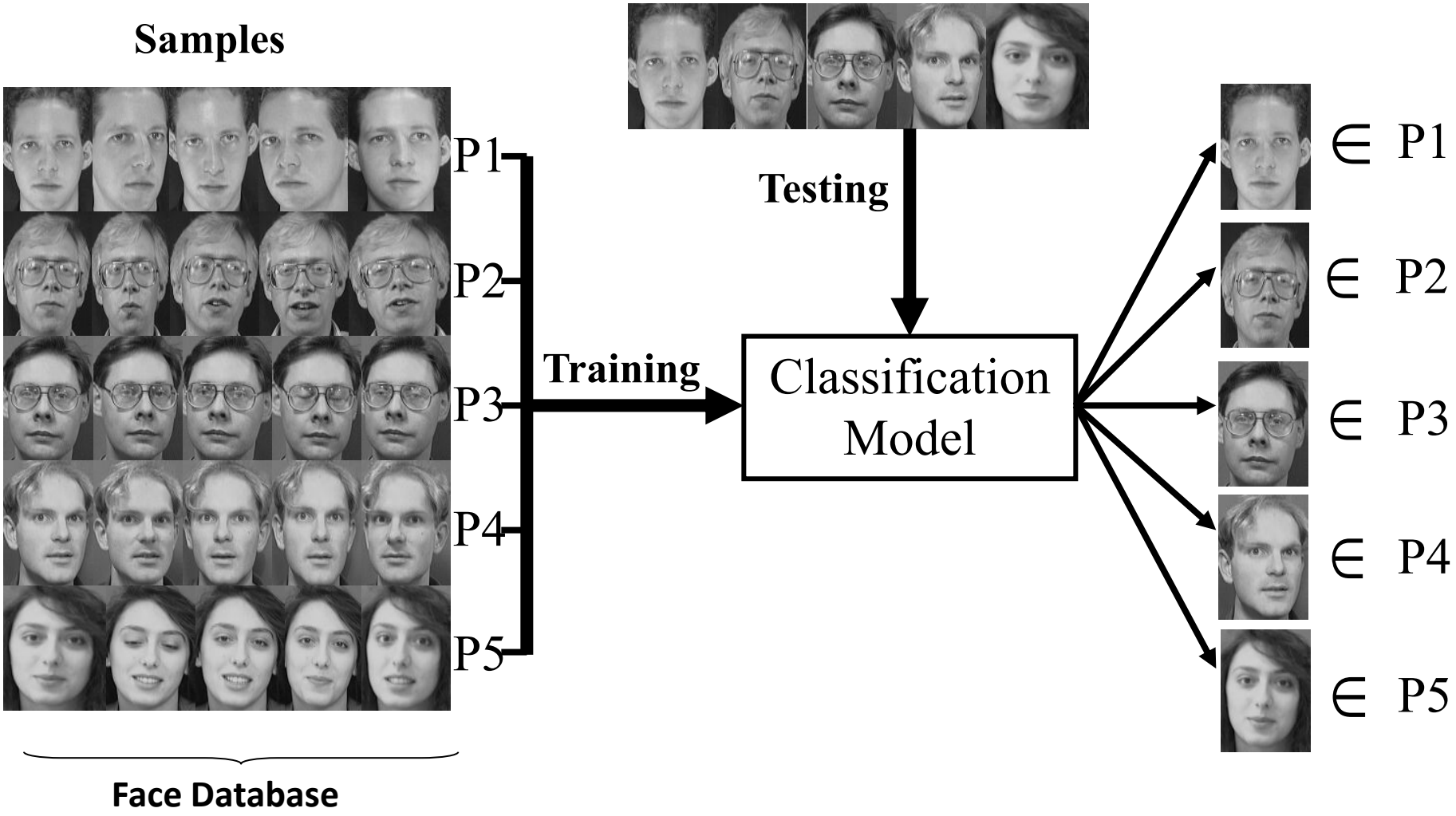
Face size 24 × 24[pixels]



Improve calculation speed

[1] Paul Viola, Michael Jones, "Robust Real-time Object Detection", July 2001

Face recognition



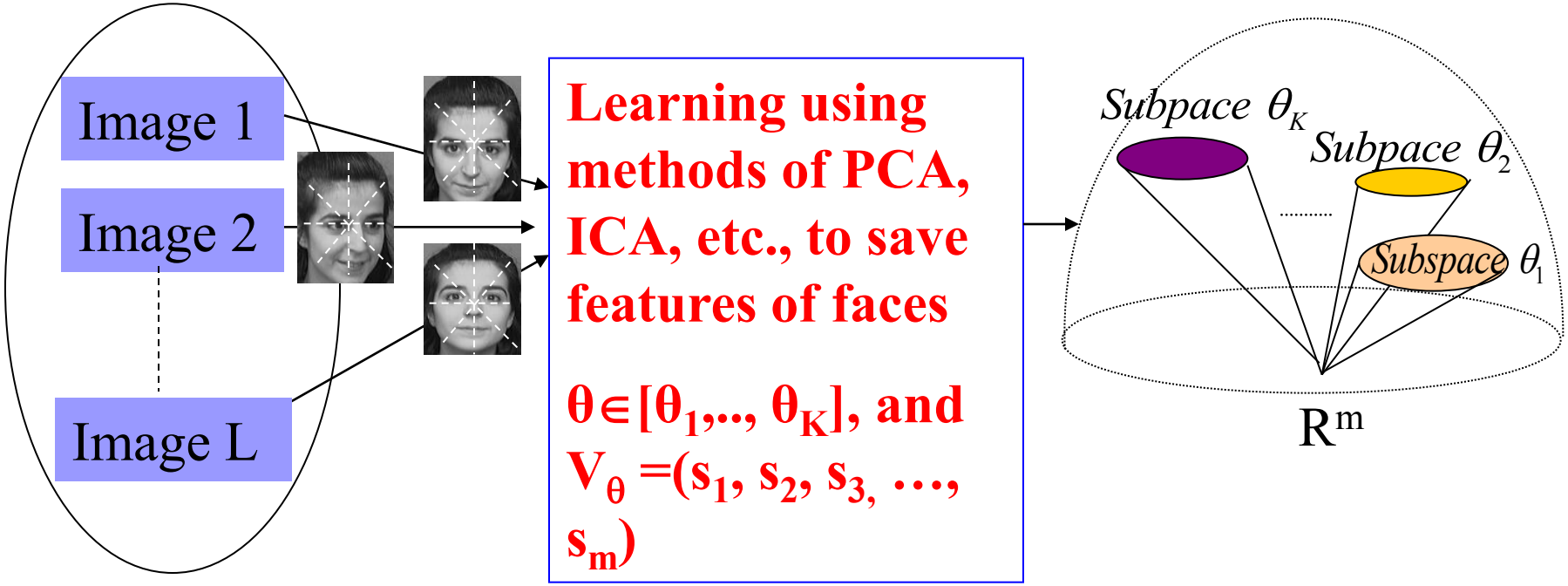
Static environment : a little or not difference between test and training environments.

Training Step

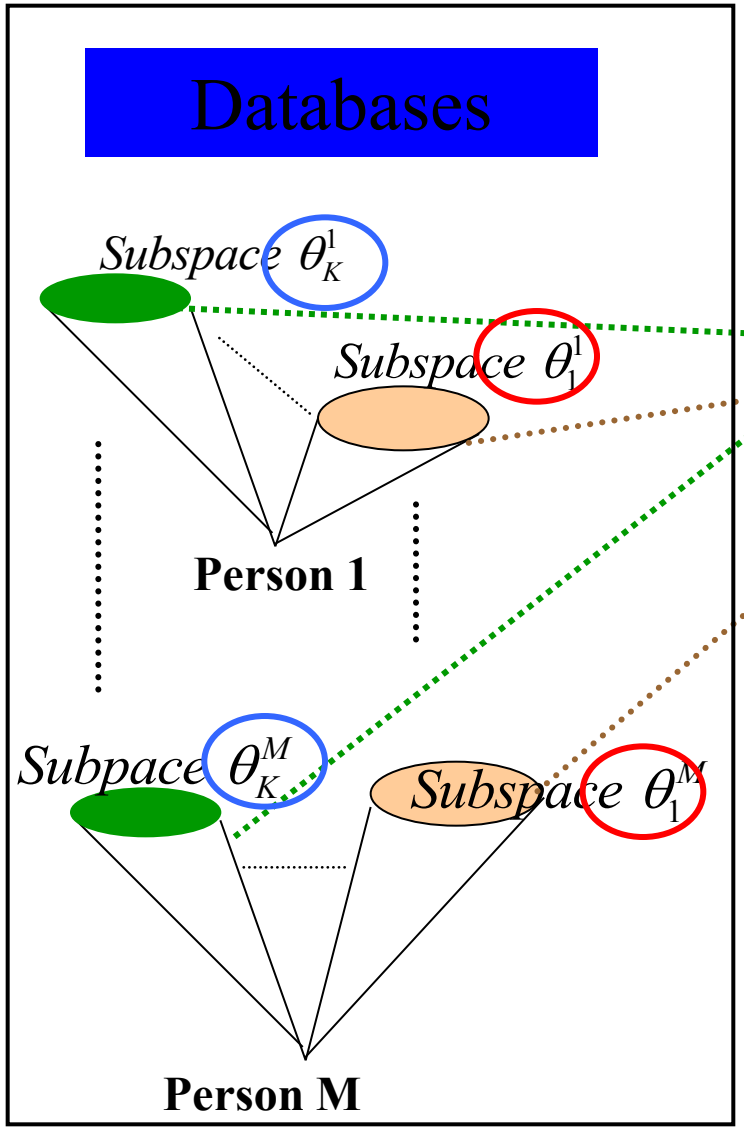
Sample image set of each person → Sample vector set of each person

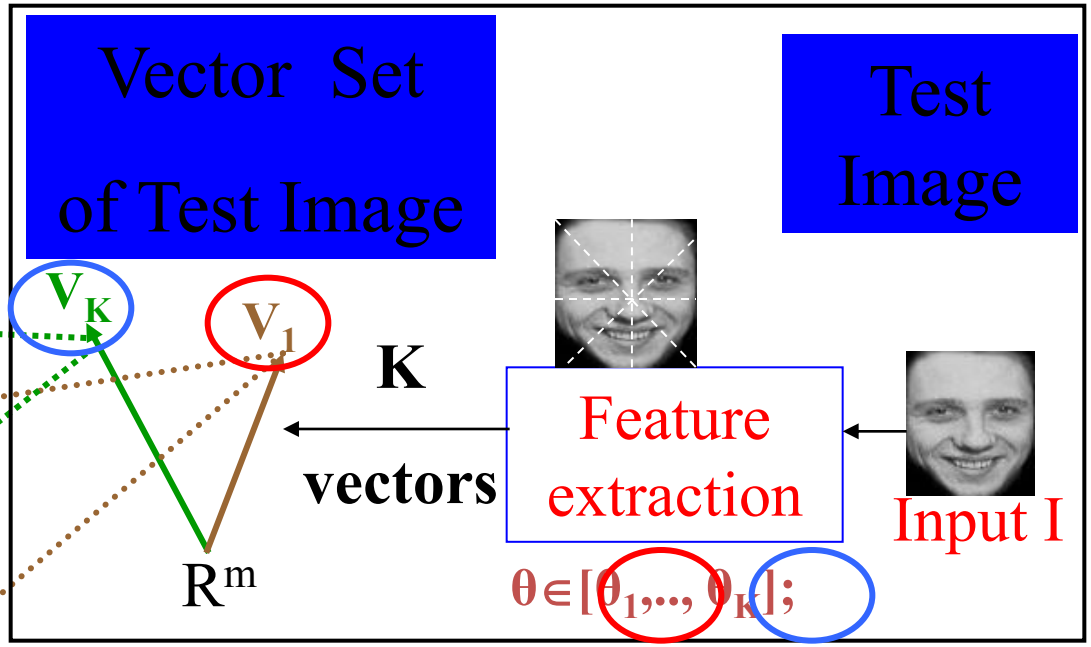
Person $i^{th} \in$ M people

Sample vector set i^{th}



Testing Step





$$\mathrm{Rec}(I_k) = \arg\max_{j} \left\{ \sum_{i=1}^{L} \delta(\{\mathrm{Class}(V_i)\text{-}j\}) \right\}_{j=1,..,M}$$

$$\text{where } \delta(t) = \begin{cases} 1, \text{if } t \text{ is equal to } 0 \\ 0, \text{otherwises} \end{cases}$$

$$\text{Recognition accuracy} = \frac{\text{Number of correct Rec}(I_k)}{\text{Number of test images}}$$

References

- http://www.face-rec.org/interesting-papers/
- OpenCV

Project 2

- Image segmentation
- Specification:

  - Study and present state-of-the art of the segmentation algorithms

  - Select one method to implement such that your program can group one image of one specified object (horse, dog, car, etc.) in a given image.

Regions and Edges

- Ideally, regions are bounded by closed contours
  - We could “fill” closed contours to obtain regions
  - We could “trace” regions to obtain edges
- Unfortunately, these procedures rarely produce satisfactory results.

Regions and Edges

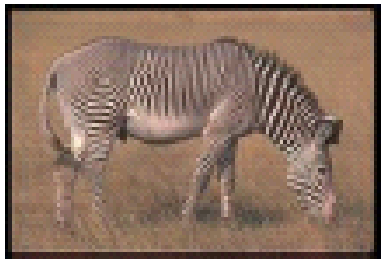
- **Edges** are found based on **DIFFERENCES** between values of adjacent pixels.
- **Regions** are found based on **SIMILARITIES** between values of adjacent pixels.
- Goal associate some higher level – more meaningful units with the regions of the image

- Grouping (or clustering)
  - collect together tokens that “belong together”
- Fitting
  - associate a model with tokens
  - Issues : model, token goes to which element, elements in the model

Clustering by connected elements

Clustering by color

Segmentation with Model EM



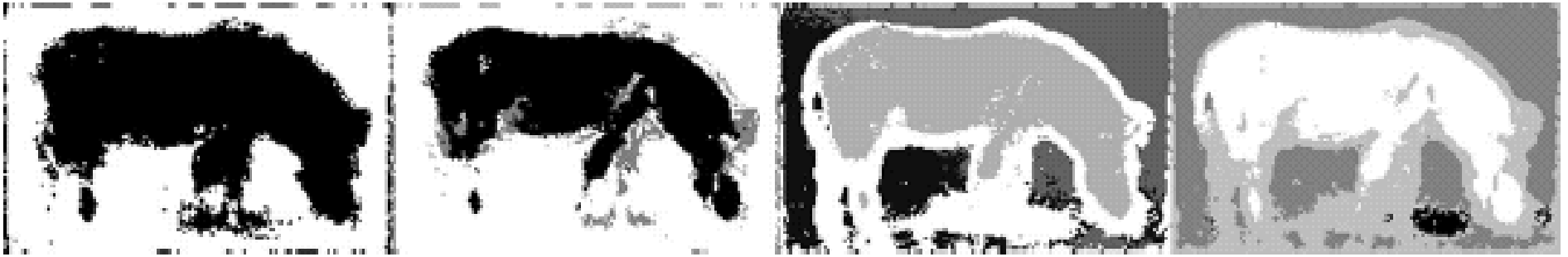


Figure from “Color and Texture Based Image Segmentation Using EM and Its Application to Content Based Image Retrieval”,S.J. Belongie et al., Proc. Int. Conf. Computer Vision, 1998, c1998, IEEE

Implementation

References

- http://www.dam.brown.edu/people/eitans/
- http://www.cvpapers.com/cvpr2009.html
- http://www.cvpapers.com/cvpr2010.html
- http://www.cvpapers.com/cvpr2008.html
- OpenCV

Project 3

- Object detection or classification
- Specification:

  - Study and present state-of-the art for one object detection (pedestrian, car)

  - Demo

  or

  - Study and present state-of-the art for object classification (PASCAL challenges)

  - Demo on one object

Object Classification

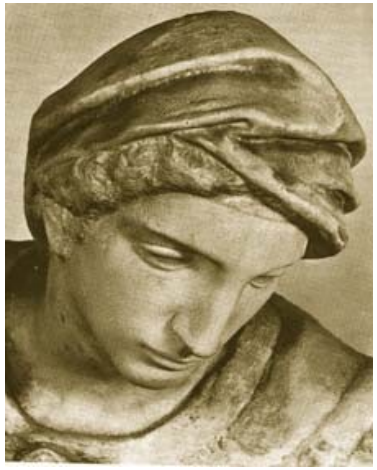
Image Classification (Object Categorization)

- An interesting problem.
- Application: Content based image retrieval system.

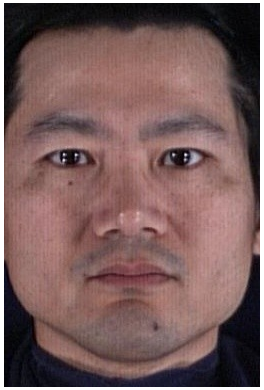
- Classify and label one specified-object image from a set of object images

Challenges

- The view point variation


- Illumination

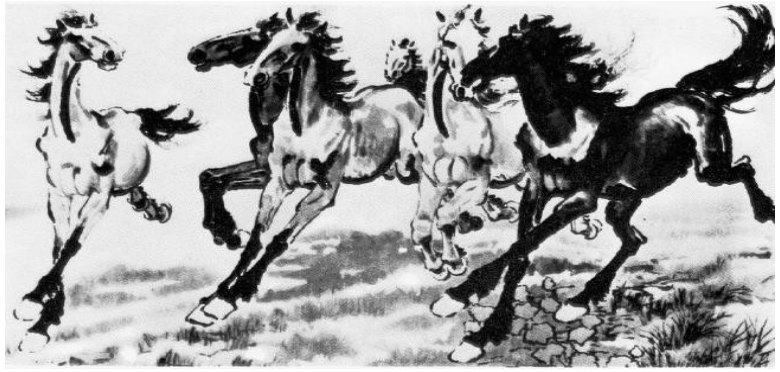



Challenges

- Occlusion

- Scale

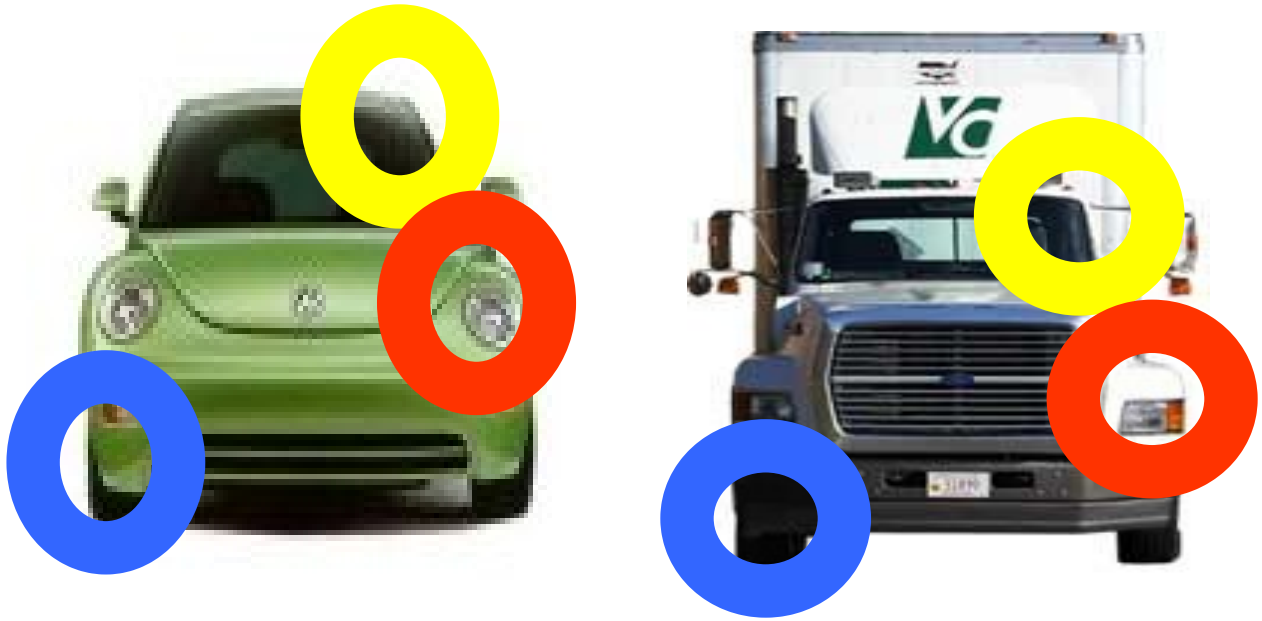
- Deformation



- Intra-class variation

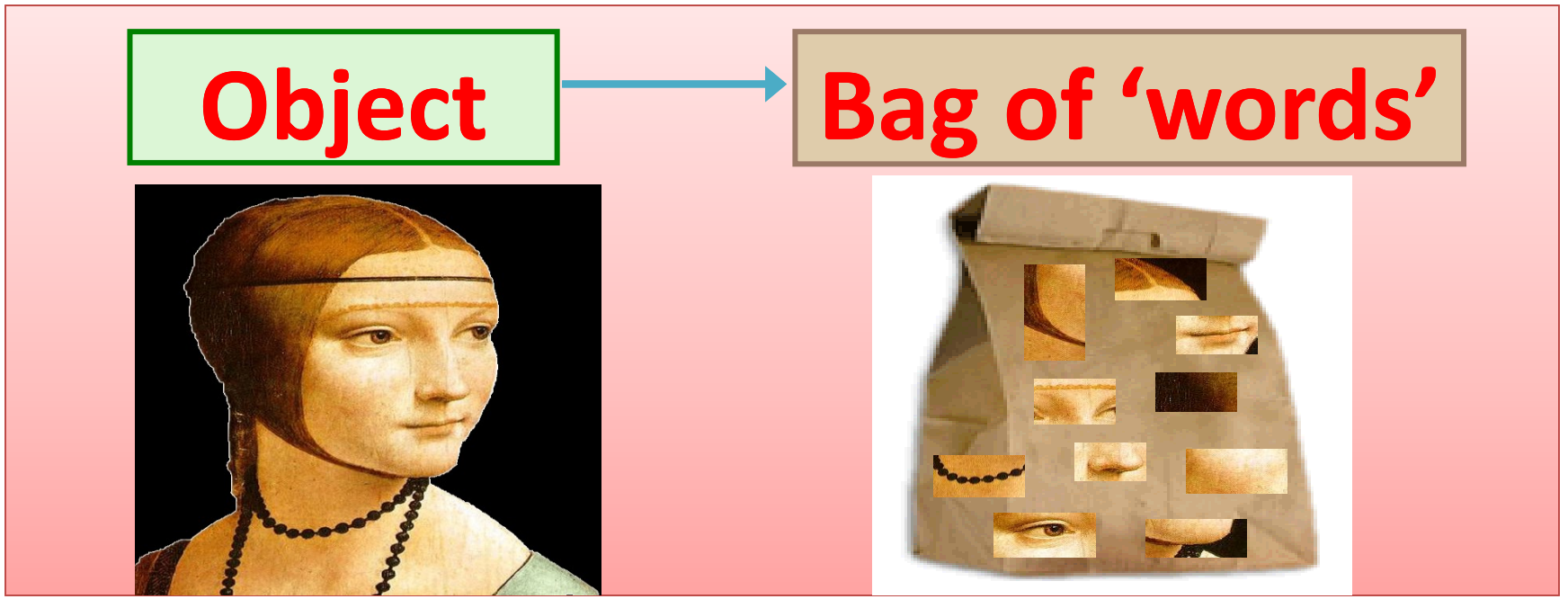
Generative Models

- Constellation model [Fergus et al, 2005]
  - Based on pictorial model [Fischler & Elschlager 1973]
  - Keep only parts which are distinctive to separate from other classes → Avoid modeling global variability



Generative Models

- Bag of words [in Sivic et al, 2005 – FeiFei Li & Petrona, 2005]
  - Local features – words
  - Object in an image – topic of a document



Challenge Datasets

- Caltech 256 : 44% [Bosch et al, 2007]
- Caltech 101 : 80% [Bosch et al, 2007]
- VOC : 66.4% [NEC-UIUC team 2009]
- MIT Pedestrian data: http://cbcl.mit.edu/cbcl/software-datasets/PedestrianData.html
- UIUC Car data : http://pascallin.ecs.soton.ac.uk/challenges/VOC/databases.html#UIUC

References

• N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in CVPR'05.
• A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," in IJCV’01.
• D. G. Lowe, “Distinctive Image Features from Scale-Invariant keypoints” in IJCV’04.
• http://pascallin.ecs.soton.ac.uk/challenges/VOC/
• http://www.csie.ntu.edu.tw/~cjlin/libsvm/
• OPenCV

Project 4

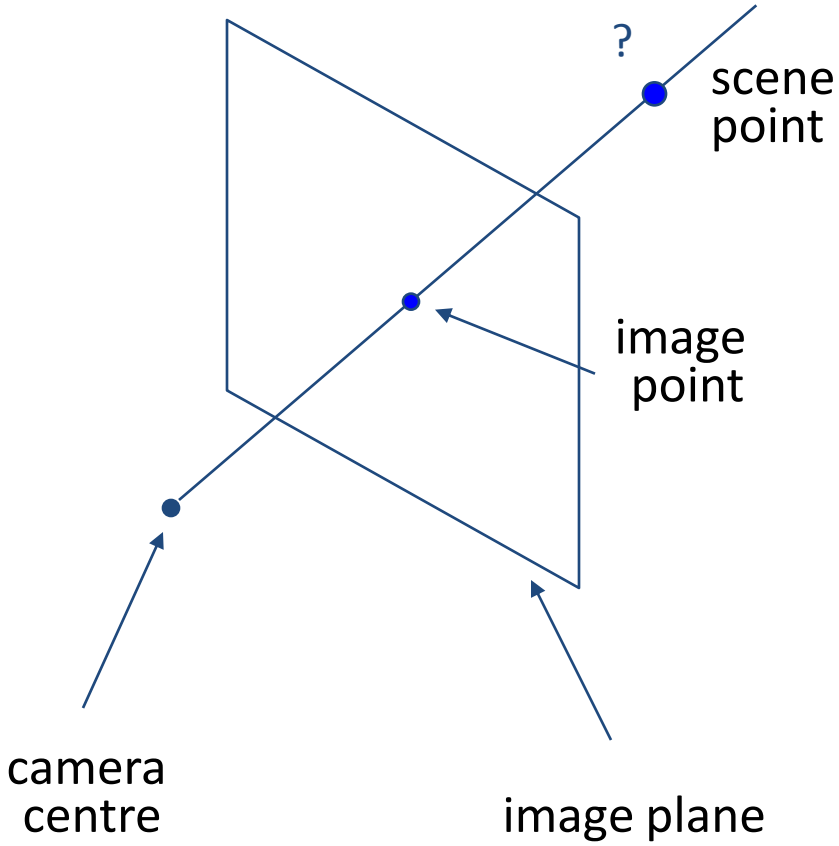
- Stereo Matching
- Specification

  - Study and present state-of-the art of the stereo matching.

  - Select one method to implement such that your program can generate the depth map and show it in 3D coordinate, where we can rotate and scale that 3D object.

Imaging geometry

- central projection
- camera centre, image point and scene point are collinear
- an image point back projects to a ray in 3-space
- depth of the scene point is unknown



Objective of Stereo Problem

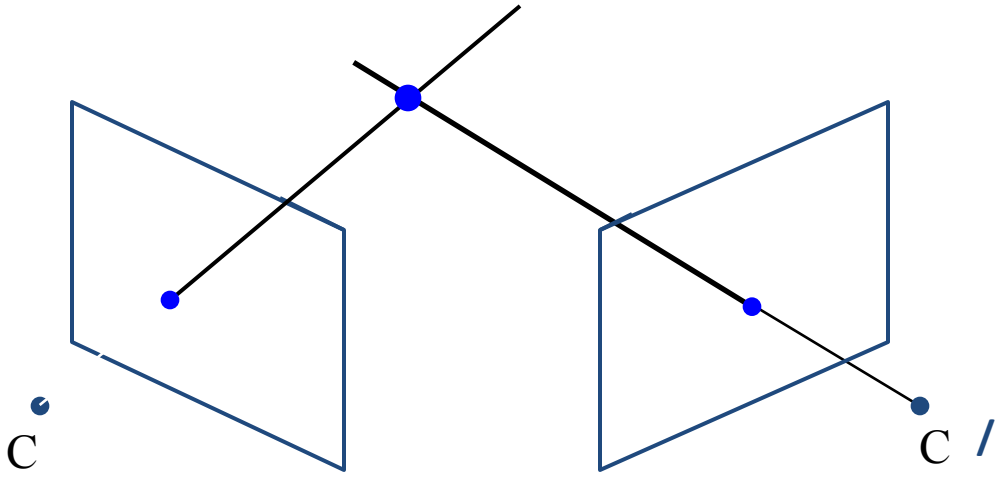
Given two images of a scene acquired by known cameras compute the 3D position of the scene (structure recovery)

Basic principle: triangulate from corresponding image points

- Determine 3D point at intersection of two back-projected rays

Corresponding points are images of the same scene point

Triangulation



The back-projected points generate rays which intersect at the 3D scene point

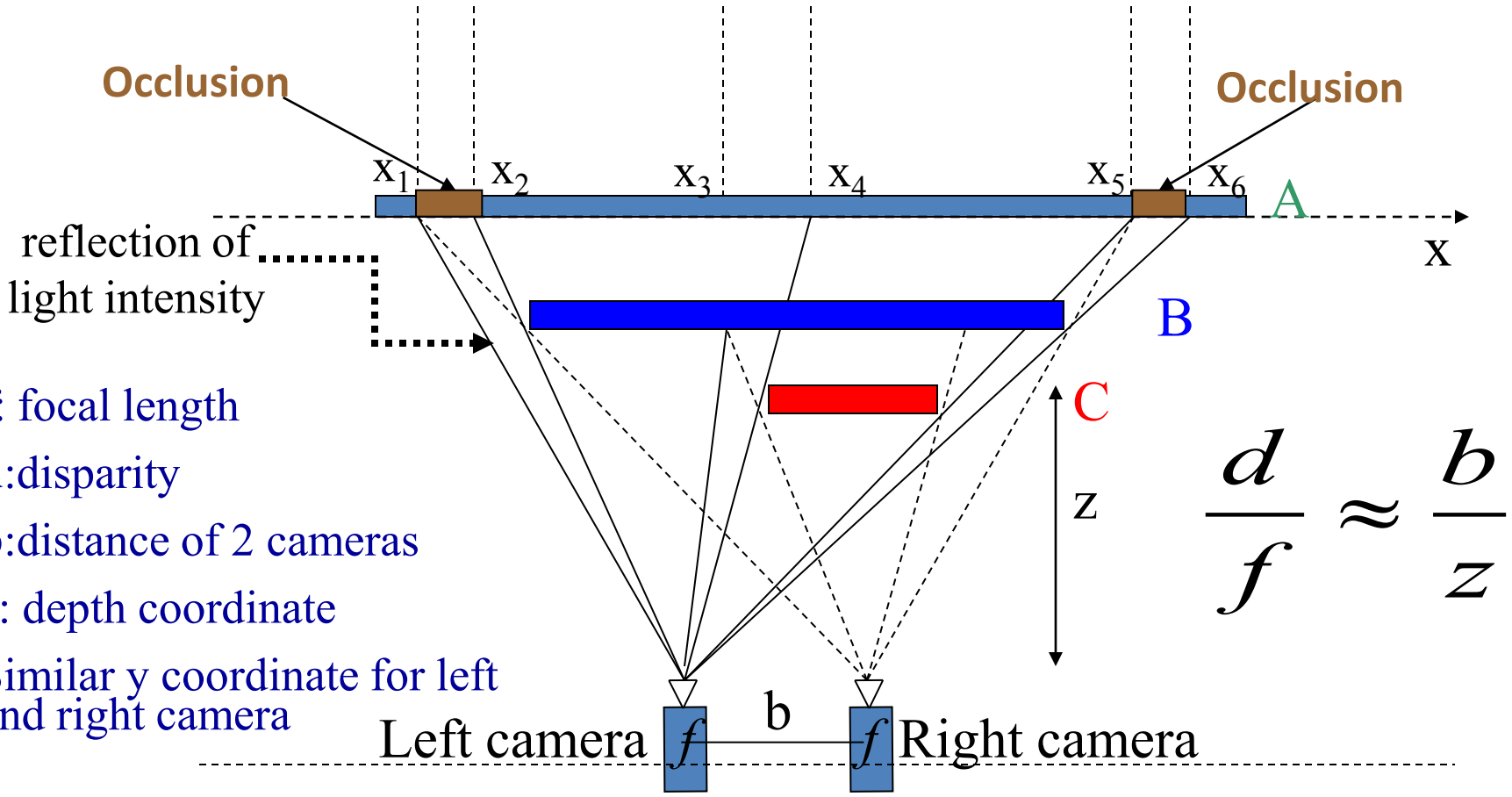
Stereo Correspondence Problem

Given a point x in one image find the corresponding point in the other image

This appears to be a 2D search problem, but it is reduced to a 1D search by the epipolar constraint

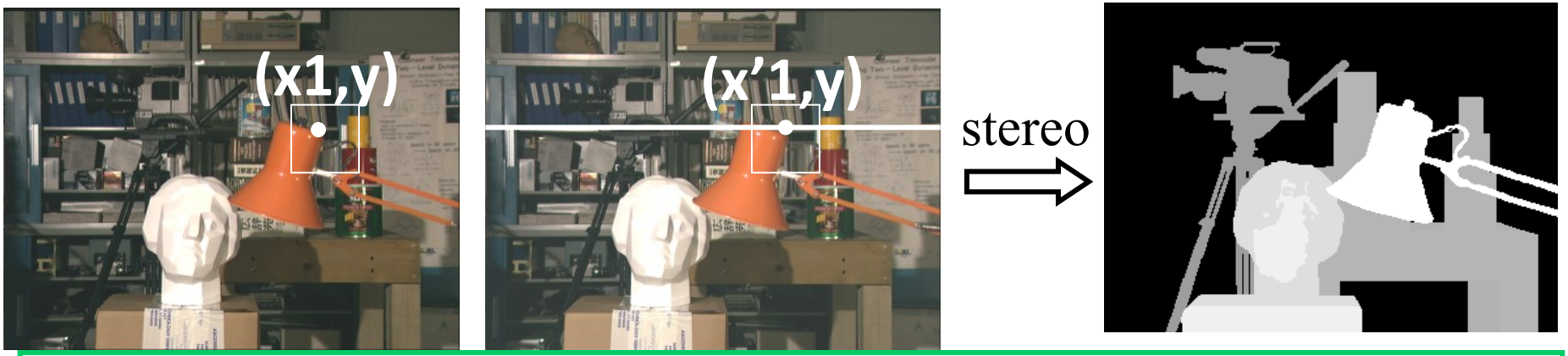
Stereo Correspondence Problem

Disparity: Difference of pixel positions in left and right images

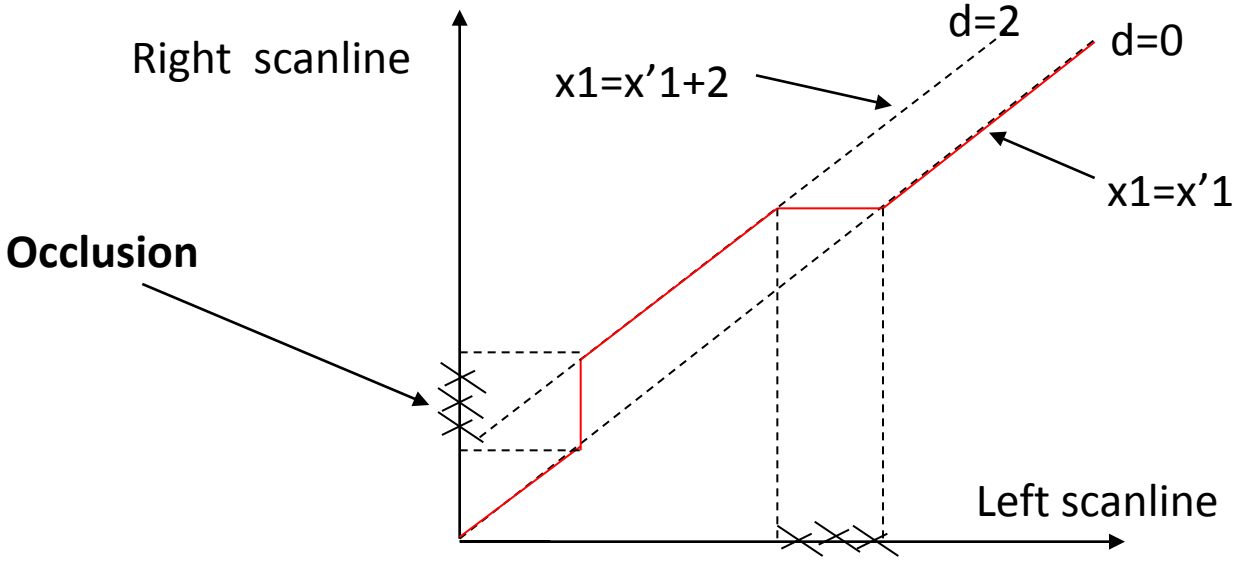


Difficulty of Stereo Problem

left image **right image** **disparities**



disparity = x1-x'1 is inversely proportional to depth



Approach for Stereo Reconstruction

1. For each point in the first image determine the corresponding point in the second image

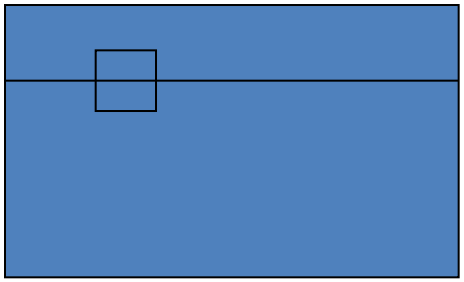
   (this is a search problem)

2. For each pair of matched points determine the 3D point by triangulation

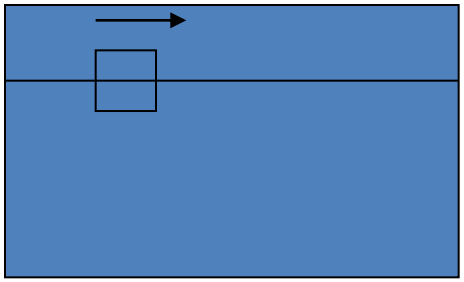
   (this is an estimation problem)

Previous Methods

- Window-based method

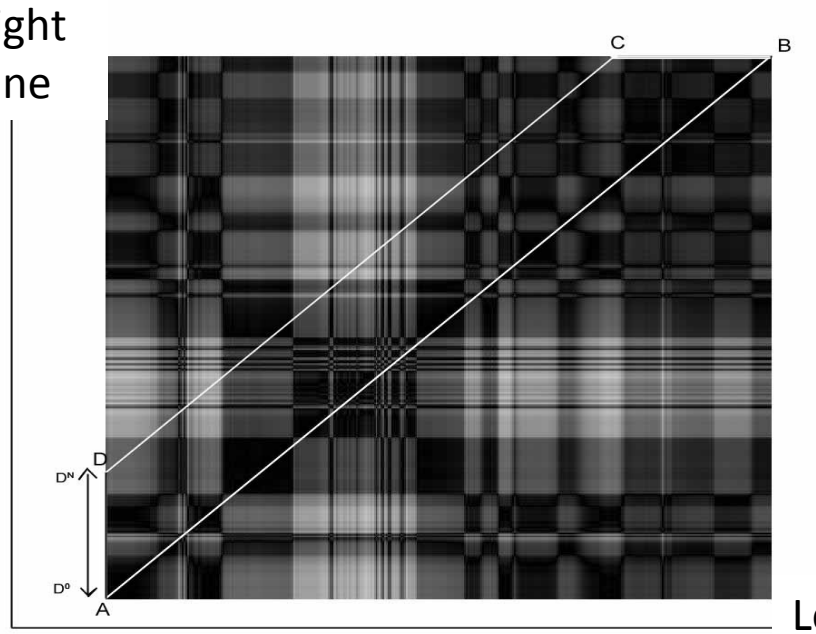


Left Image I^L



Right Image I^R

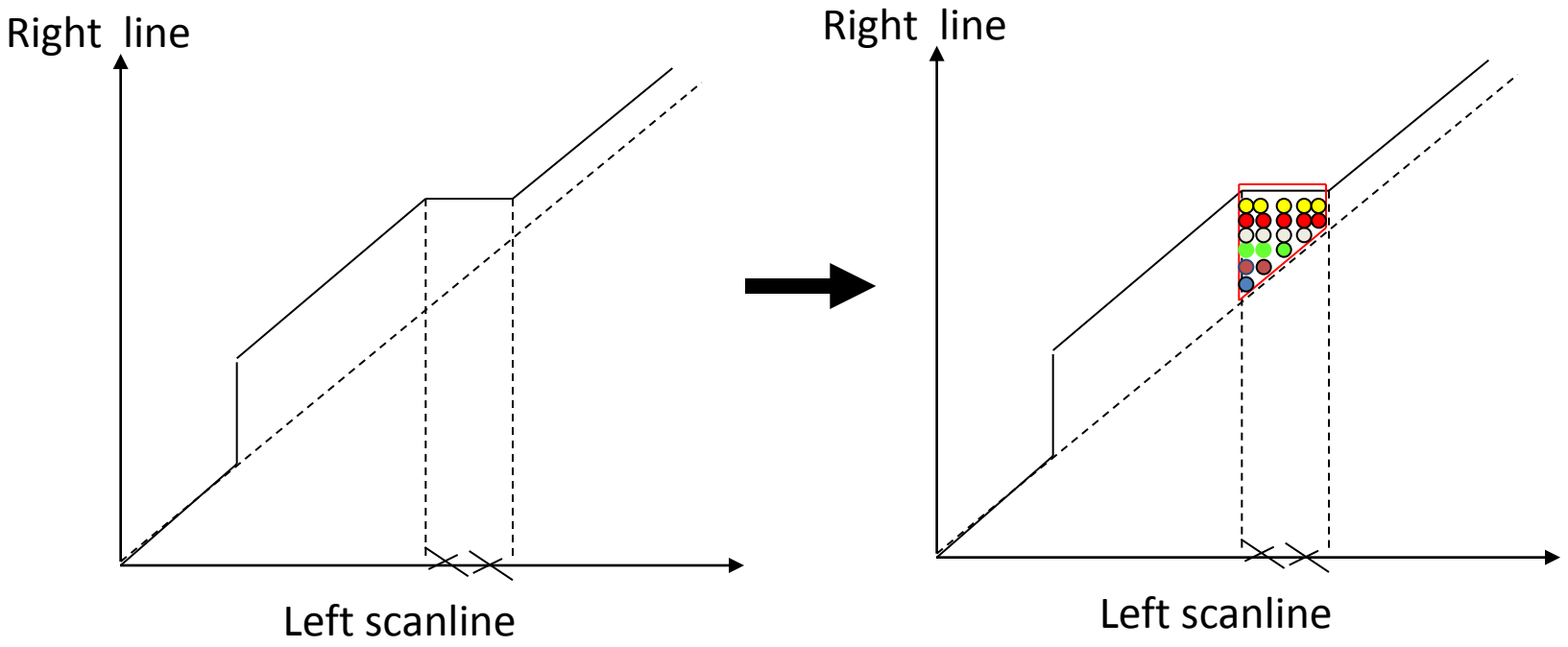
- Dynamic programming



Matching space:

$$M(l,r) = | I^L(x_l, y_j) - I^R(x_r, y_j) |$$

Problem of dynamic programming



Number of states should be increased considerably to solve this problem

References

- OpenCV
- Three Dimension Computer Vision, O. Faugeras, MIT press, 1993.

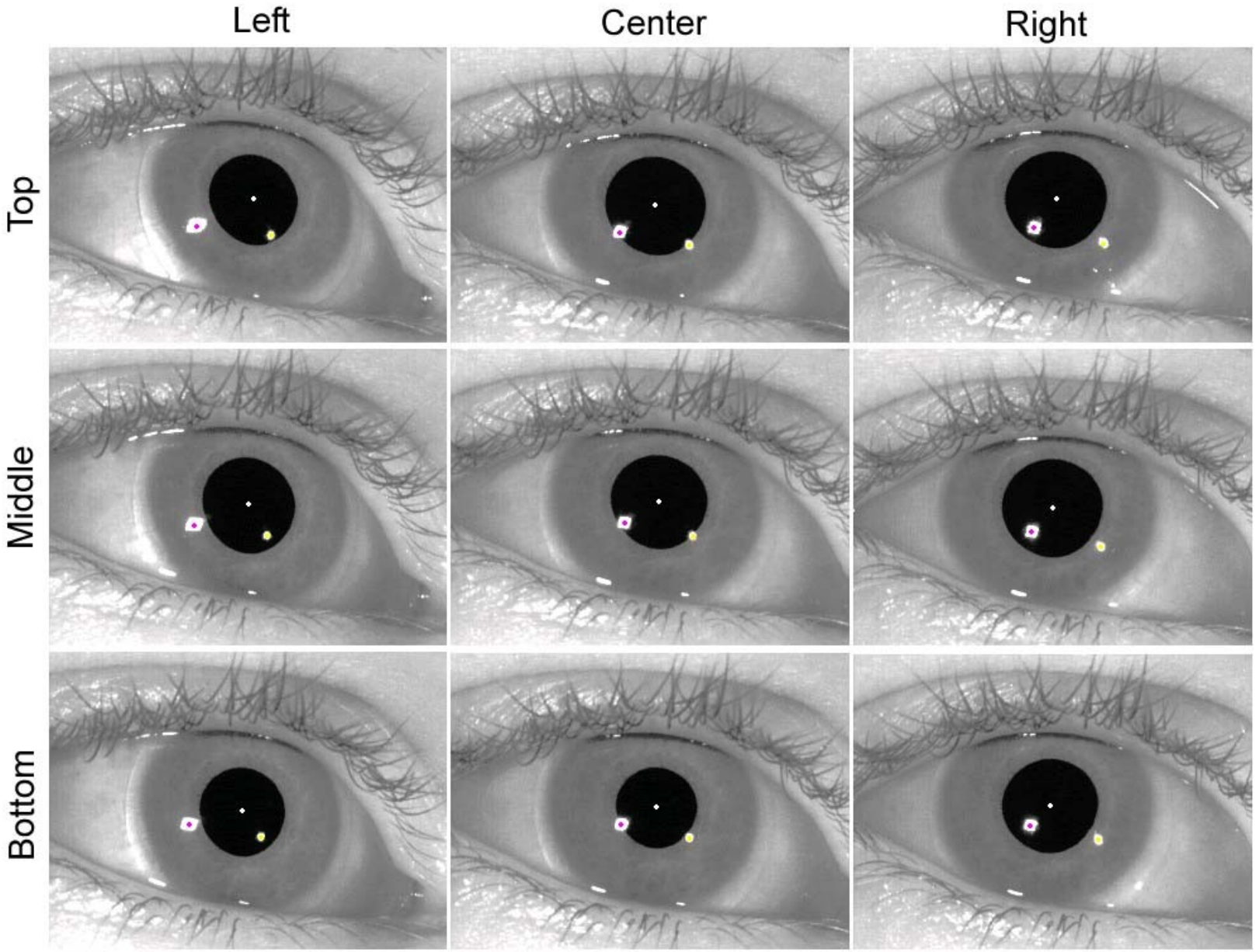
Project 5

- Human Computer Interaction (HCI)
- Specification:

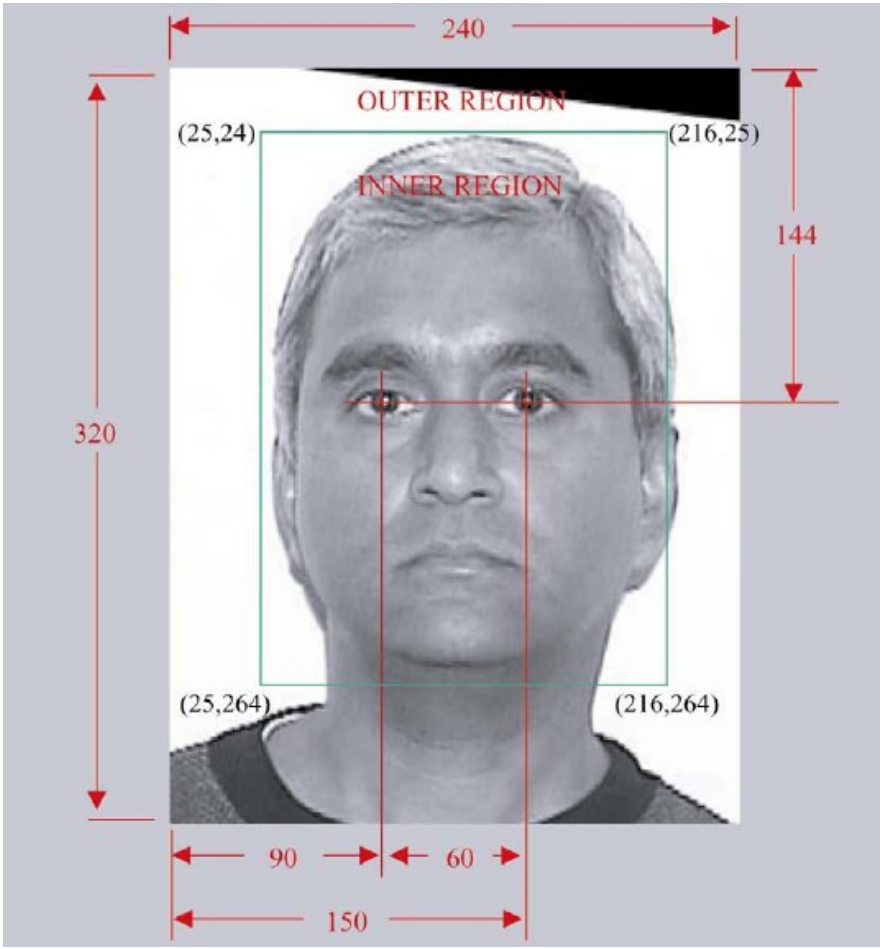
  - Study and present method of eye tracking and eye blink detection.

  - Build a demo where we can use eye movement and eye blink to move and control mouse click.

Corneal Reflections/Calibration

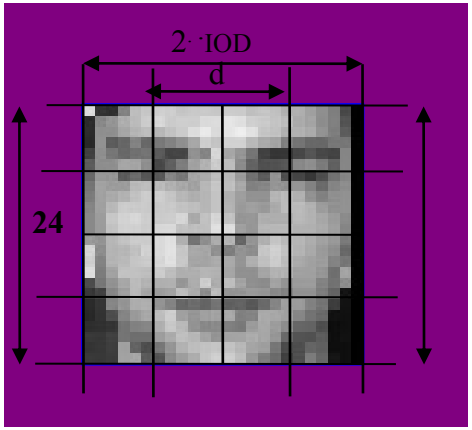


Eye-centered face models



Canonical face model suitable for Face Recognition in documents [Identix'02]

Canonical face model suitable for on-line Face Memorization and Recognition in video



Procedure: after the eyes are located, the face is extracted from video and resized to the canonical 24x24 form, in which it is memorized or recognized.

Tracking

- Mean-shift method
- Kalman filter
- Particle filter

References

- OpenCV
- http://mplab.ucsd.edu/grants/project1/free-software/mptwebsite/introduction.html