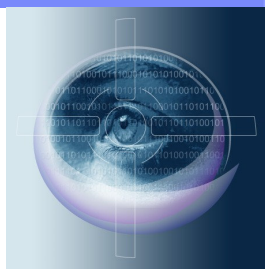


Face Detection and Tracking for Video Surveillance

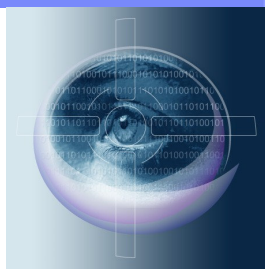
Rogerio Feris

IBM TJ Watson Research Center

rsferis@us.ibm.com

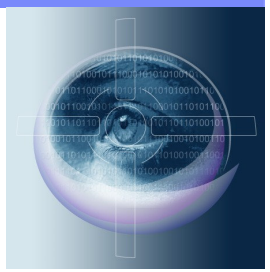
http://rogerioferis.com

Objectives for this Class



- Survey recent face detection/tracking methods
(with emphasis on detection, as Andrew already covered different tracking techniques)
- Pros and Cons of each approach
- State-of-the-art and Future Directions
- Practical Implementation Aspects

Outline



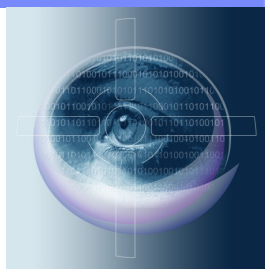
- Motivation
- Face Detection
  - Appearance-Based Learning
  - Other Modalities
- Face Tracking
- The IBM Face Capture System

Motivation

Capturing People in Surveillance Video [Feris et al, VS'07]

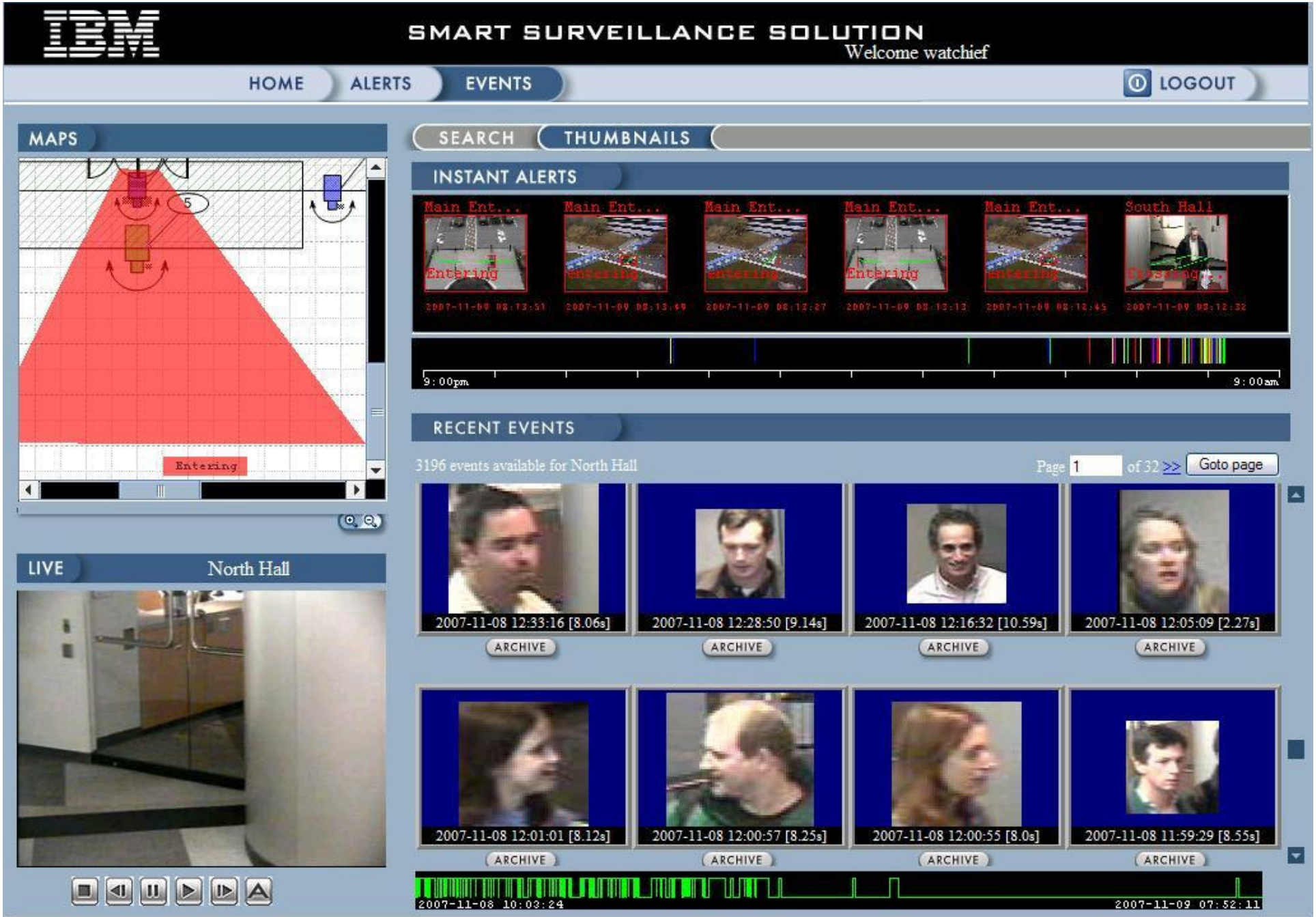
Face Tracker and Capture

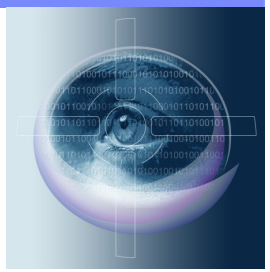
IBM Smart Surveillance Solution



Searching for People

Query: Show me all the **people** who entered IBM from **11am to 5pm**, in **November 8th, 2007**.


IBM
SMART SURVEILLANCE SOLUTION
Welcome watchief
HOME
ALERTS
EVENTS
LOGOUT
MAPS
Entering
SEARCH
THUMBNAILS
INSTANT ALERTS
Main Ent...
Entering
South Hall
9:00pm
9:00am
RECENT EVENTS
3196 events available for North Hall
Page 1 of 32 >>
Goto page
2007-11-08 12:33:16 [8.06s]
2007-11-08 12:28:50 [9.14s]
2007-11-08 12:16:32 [10.59s]
2007-11-08 12:05:09 [2.27s]
ARCHIVE
2007-11-08 12:01:01 [8.12s]
2007-11-08 12:00:57 [8.25s]
2007-11-08 12:00:55 [8.0s]
2007-11-08 11:59:29 [8.55s]
LIVE
North Hall
2007-11-08 10:03:24
2007-11-09 07:52:11



Face Detection and Tracking

- First step for other face analytics, like face recognition, gender, age, and race classification, facial expression analysis, etc.

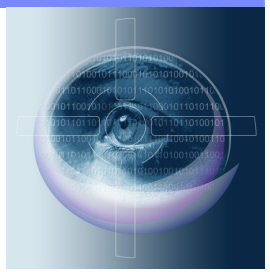
More Sophisticated People Search:

Query: Show me all **Asian** **Women** who entered the store wearing a **red jacket** **last month**.

Motivation

Face Mask Detection

Outline



- Motivation
- **Face Detection**
  - Appearance-Based Learning
  - Other Modalities
- Face Tracking
- The IBM Face Capture System

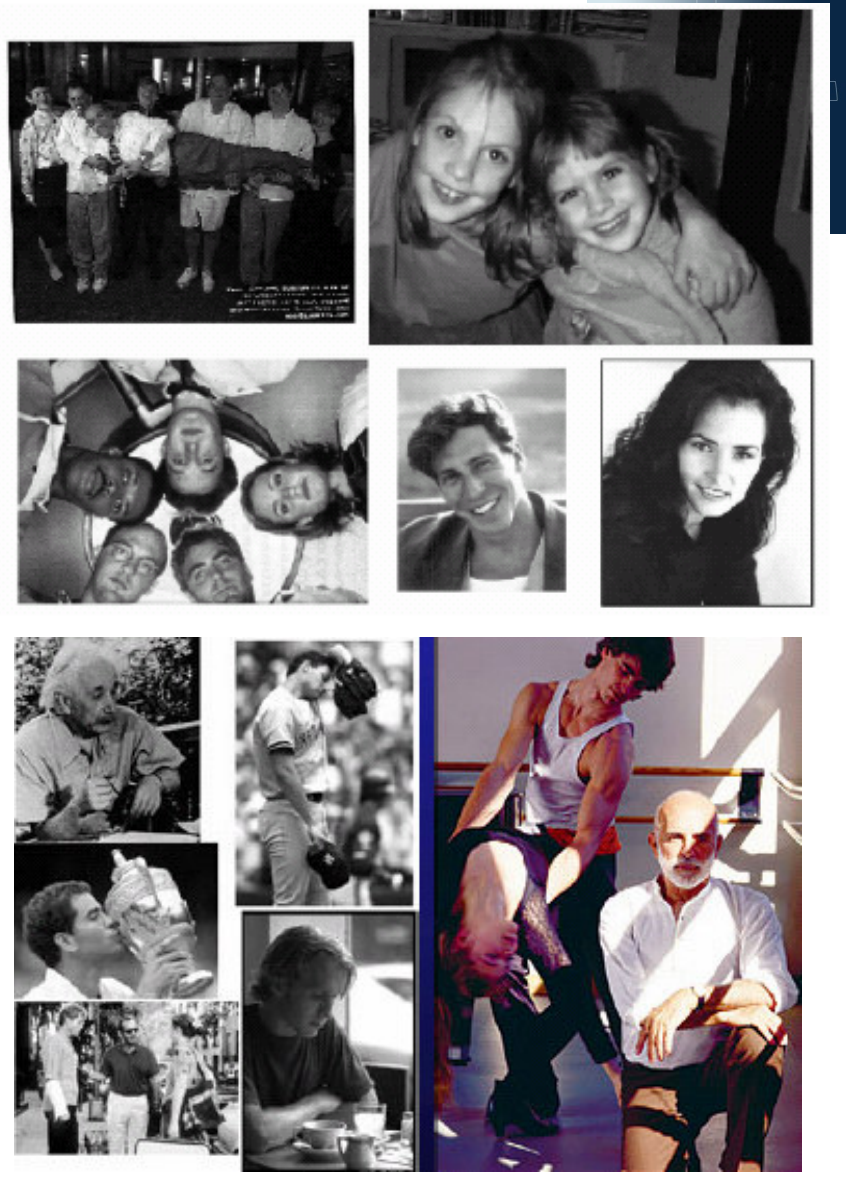
Problem Definition

Given an arbitrary image, the goal of face detection is to determine whether or not there are any faces in the image and, if present, return the image location and extent of each face. [Yang et al, Detecting Faces in Images: a Survey, Pami 2002]

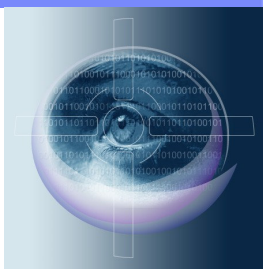
Image from Bo Wu – Real Adaboost for Multi-view face detection

Why this is a Hard Problem?

- Pose Variation
- Beards, Glasses, …
- Facial Expression
- Occlusion
- Image Orientation
- Lighting, Noise, …

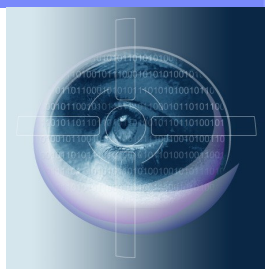


Related Problems



- Facial Feature Extraction (like eyes, nose, mouth localization)
- Face Recognition (who is the person)
- Facial Expression Analysis
- Head Pose Estimation
- Gender/Age/Race Classification
- Face Tracking

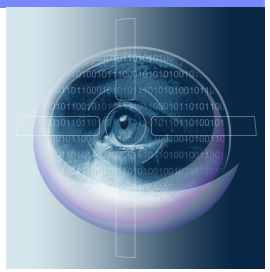
Early Work in Face Detection



- Template-based Approaches [Brunneli & Poggio, 1993]
  - Intra-class variation is difficult to represent with few templates
- Edge-Based / Deformable Templates [Yuille, 1992]
  - Sensitive to edge noise and template initialization.
- See face detection survey [Yang et al, PAMI 2002]
  - Comprehensive survey of methods till year 2000

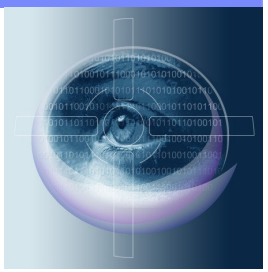
Early approaches don't work well for complex scenes!!!

Outline



- Motivation
- Face Detection
  - **Appearance-Based Learning**
  - Other Modalities
- Face Tracking
- The IBM Face Capture System

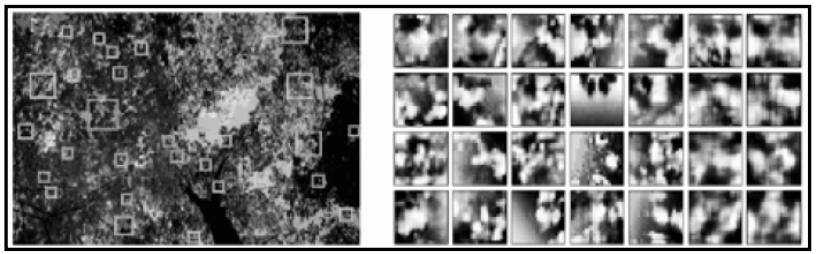
Apperance-based Learning



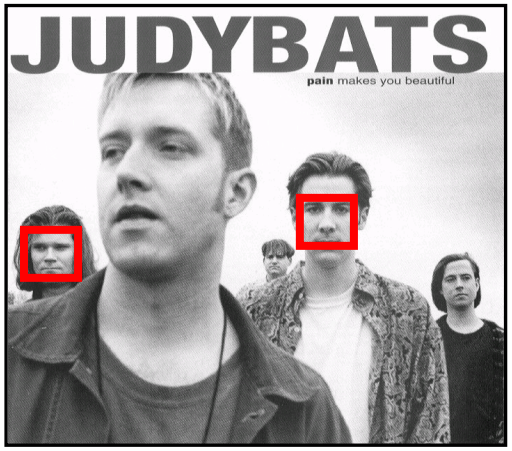
- General Idea:
  - Collect a large set of resized (e.g., 20x20) face and non-face images and learn a classifier to discriminate them.
  - Given a test image, detect faces by applying the classifier at each position and scale of the image.

Faces

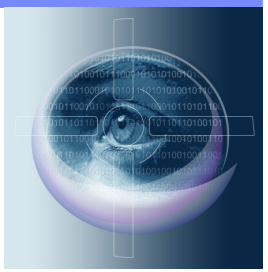
Non-Faces



Search over Space and Scale

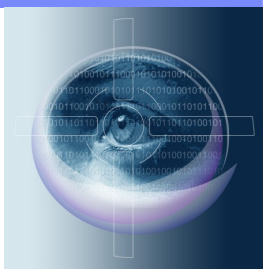


Research Questions



- How can we create a robust classifier to discriminate faces from non-faces?
- How can we handle multi-view face detection?
- How can we make this process more efficient?

Pioneer Work [Sung & Poggio, 1994]

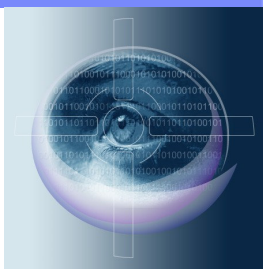


- Sung & Poggio, “Example-based Learning for View-based Human Face Detection”, AI Memo, 1994.

- **Face/Non-Face Classifier:** Neural Network (Multi-Layer Perceptron) trained on distances from image patches to face and non-face distributions.

- **How to model face and non-face distributions?**

Pioneer Work [Sung & Poggio, 1994]



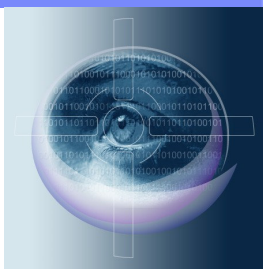
Modeling the Face and Non-Face Distributions

➢ They collected 1067 face patterns and for each one of this training images they did:

- resizing to 19x19 pixels
- Masking to avoid background pixels
- Lighting correction / Histogram Equalization

19x19

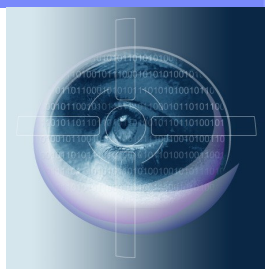
Pioneer Work [Sung & Poggio, 1994]



- In addition, for each one of the 1067 face patterns, new virtual samples were generated by rotation and mirror operations, totalizing **4150 face patterns**.

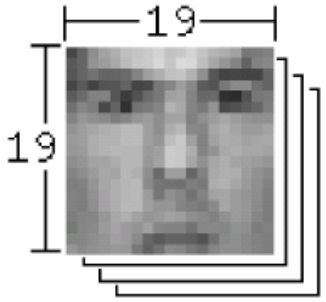
- **6189 Non-face data samples** were specially selected to be patterns that look like faces but are not faces. This is done by a process called bootstrap, which we will describe later.

Pioneer Work [Sung & Poggio, 1994]

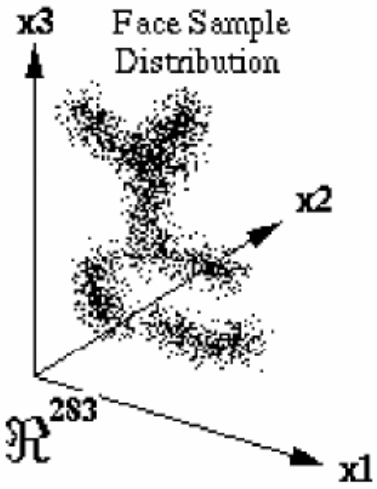


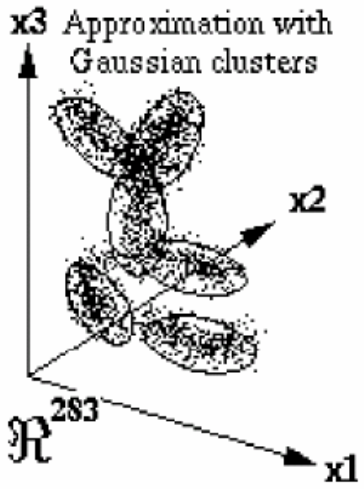
Modeling the Face and Non-Face Distributions

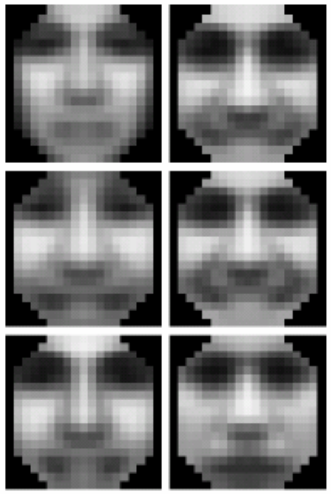
- Each 19x19 pattern can be seen as a point in a high dimensional space.
- The K-means algorithm is used to cluster face and non-face samples into 6 clusters.
- Each cluster is described by a multi-dimensional Gaussian with a centroid and covariance matrix.



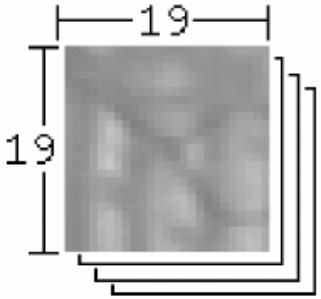
Frontal Face Pattern samples to approximate vector subspace of canonical face views



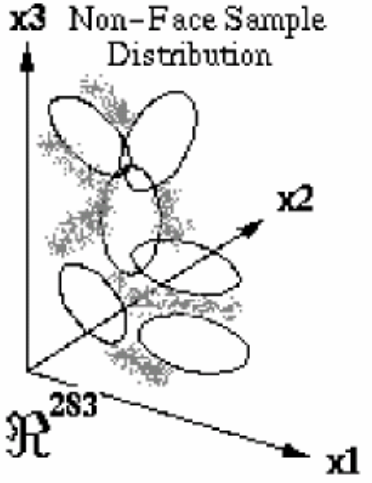


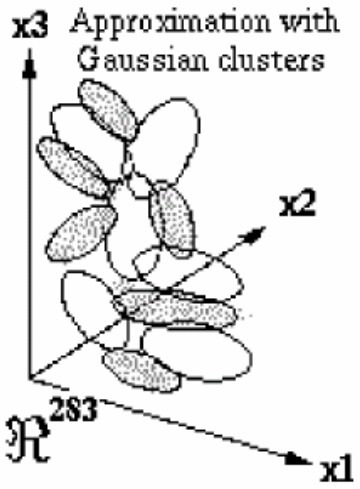


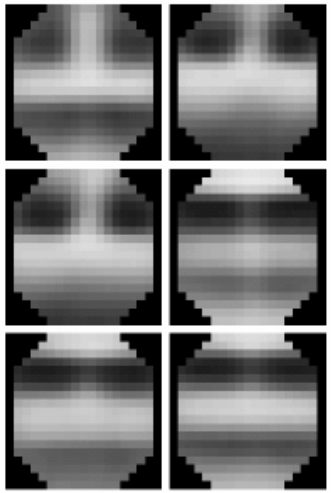
Face Centroids



Special Non-Face Pattern samples to refine vector subspace boundaries of canonical face views



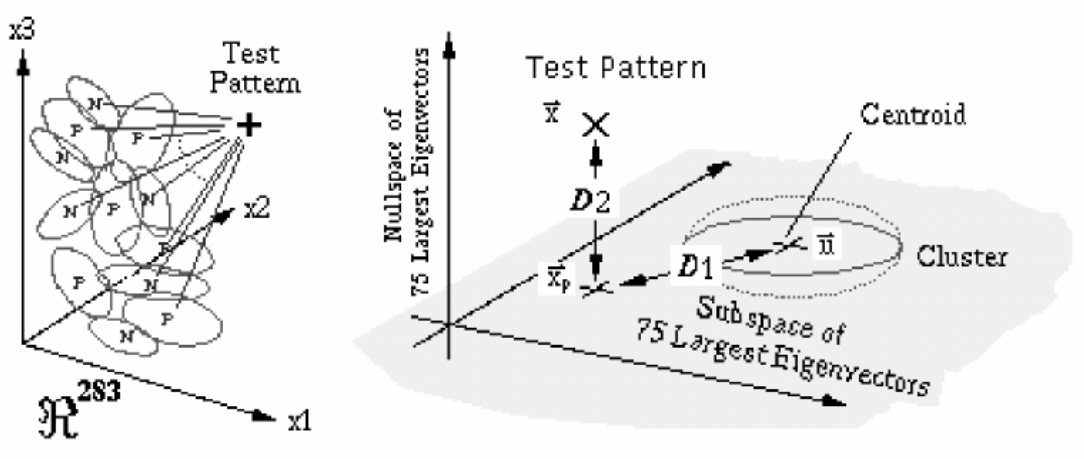




Non-Face Centroids

Pioneer Work [Sung & Poggio, 1994]

Distance Metrics



- Compute the distance of a sample to all face and non-face clusters

- **Final Classifier:** Multi-layer Perceptron is trained on the distance vector

Pioneer Work [Sung & Poggio, 1994]

Bootstrap:

1. Start with a small set of non non-face examples in the training set

2. Train a MLP classifier with the current training set

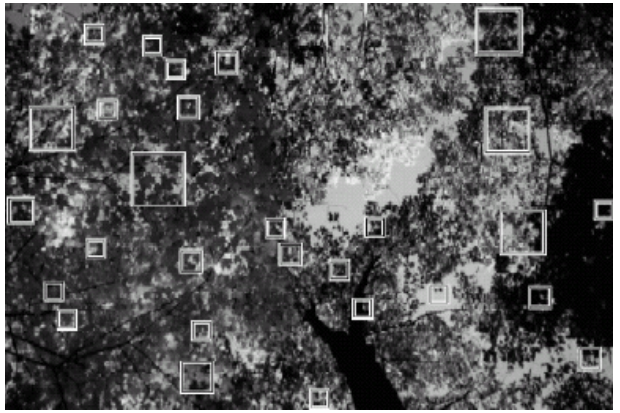
3. Run the learned face detector on a sequence of random images.

4. Collect all the non non-face patterns that the current system wrongly classifies as faces (i.e., false positives)

5. Add these non non-face patterns to the training set

6. Got to Step 2 or stop if satisfied

Test Image



False Detections

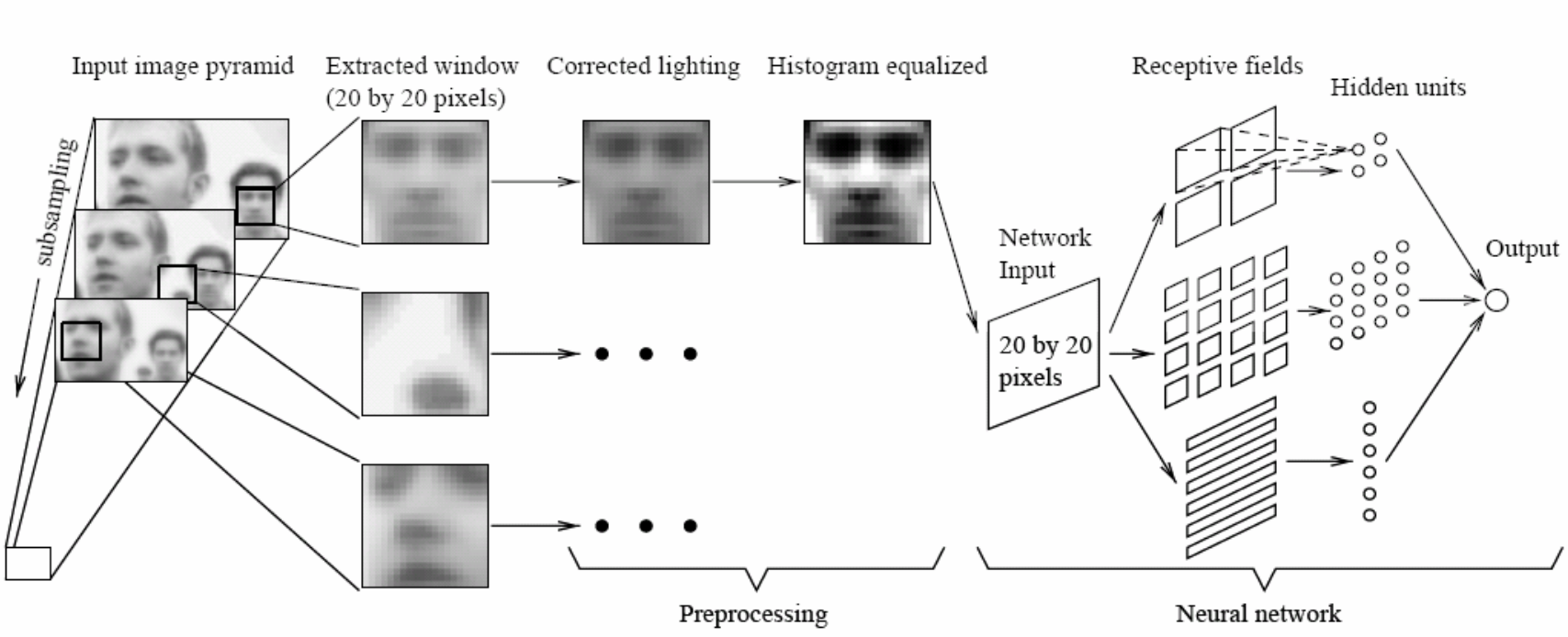
Pioneer Work [Sung & Poggio, 1994]

Results

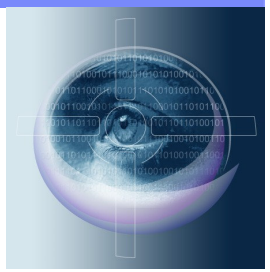
- Able to detect upright frontal faces in complex scenes.

Pioneer Work [Rowley et al, 1996]

- Rowley et al, "Neural Network-Based Face Detection, CVPR 1996



Pioneer Work [Rowley et al, 1996]

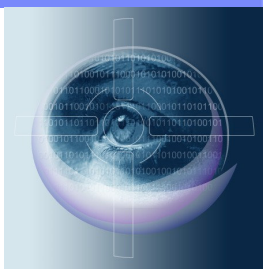


- Same techniques for bootstrap, preprocessing, etc.
- Neural Network applied directly into the image.
- Different heuristics (like multiple neural networks and arbitration)
- Faster than Sung & Poggio (but still not real-time)
- You can play with the **source code** at:

  http://vasc.ri.cmu.edu/NNFaceDetector/

Pioneer Work [Rowley et al, 1996]

Pioneer Work – Pros and Cons



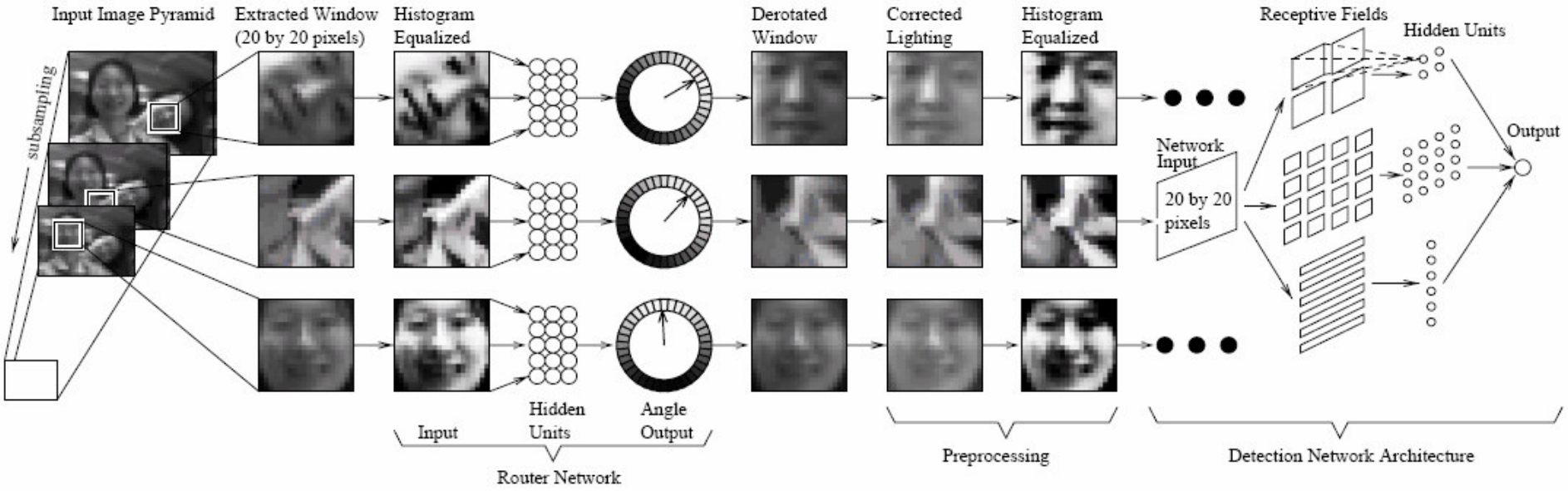
- Pros:
  - Reliable results in complex scenes

- Cons:
  - Can not handle multi-view faces
  - Computationally expensive

Rotation-Invariant Face Detection

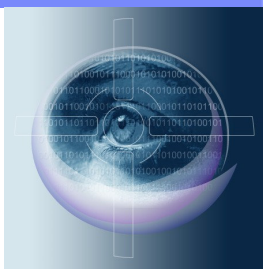
Rowley et al, 1997

- A router network is applied to determine the angle of the input window
- The de-rotated window is then applied to a frontal face detector



Rotation-Invariant Face Detection

Local Statistics of Parts [Schneiderman, 2000]

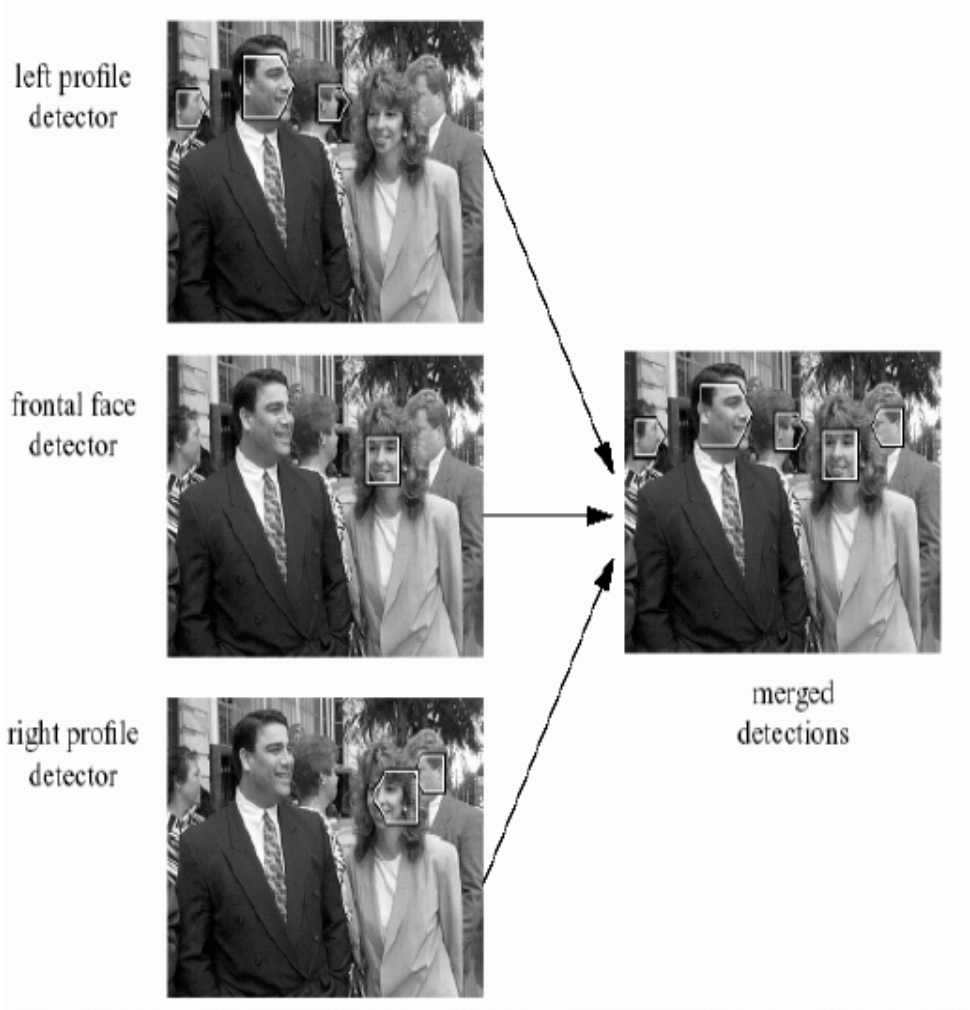


- H. Schneiderman, T. Kanade. "A Statistical Method for 3D Object Detection Applied to Faces and Cars". IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2000)

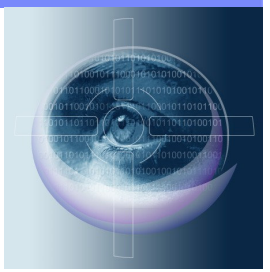
- Approach based on probabilistic modeling of local appearance. **First face detector system that works for multiple views!**

- Three different classifiers are created: frontal face detector, left profile, and right profile. The right profile classifier is the same as the left profile, but take as input mirrored images.

Local Statistics of Parts [Schneiderman, 2000]



Local Statistics of Parts [Schneiderman, 2000]



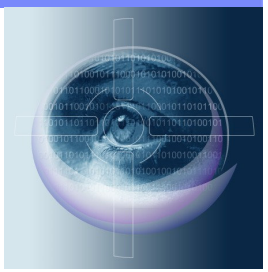
- Scan image at multiple positions and scales as in previous approaches. Apply Bayesian classifier to decide whether the search window contains a face or not:

$$\frac{P(image|\text{object})}{P(image|\text{non-object})} > \lambda \qquad \left(\lambda = \frac{P(\text{non-object})}{P(\text{object})}\right)$$

- Distribution of faces and non-faces are represented by “parts” of the object:

$$\prod_r \frac{P_r(part_r|object)}{P_r(part_r|non-object)} > \lambda$$

Local Statistics of Parts [Schneiderman, 2000]



- Each part has a position (x,y) inside the window and a local operator output $f_k(x,y)$

$$\prod_r \frac{P_r(part_r|object)}{P_r(part_r|non-object)} = \prod_k \prod_{x,y} \frac{P_k(f_k(x,y),x,y|object)}{P_k(f_k(x,y),x,y|non-object)}$$

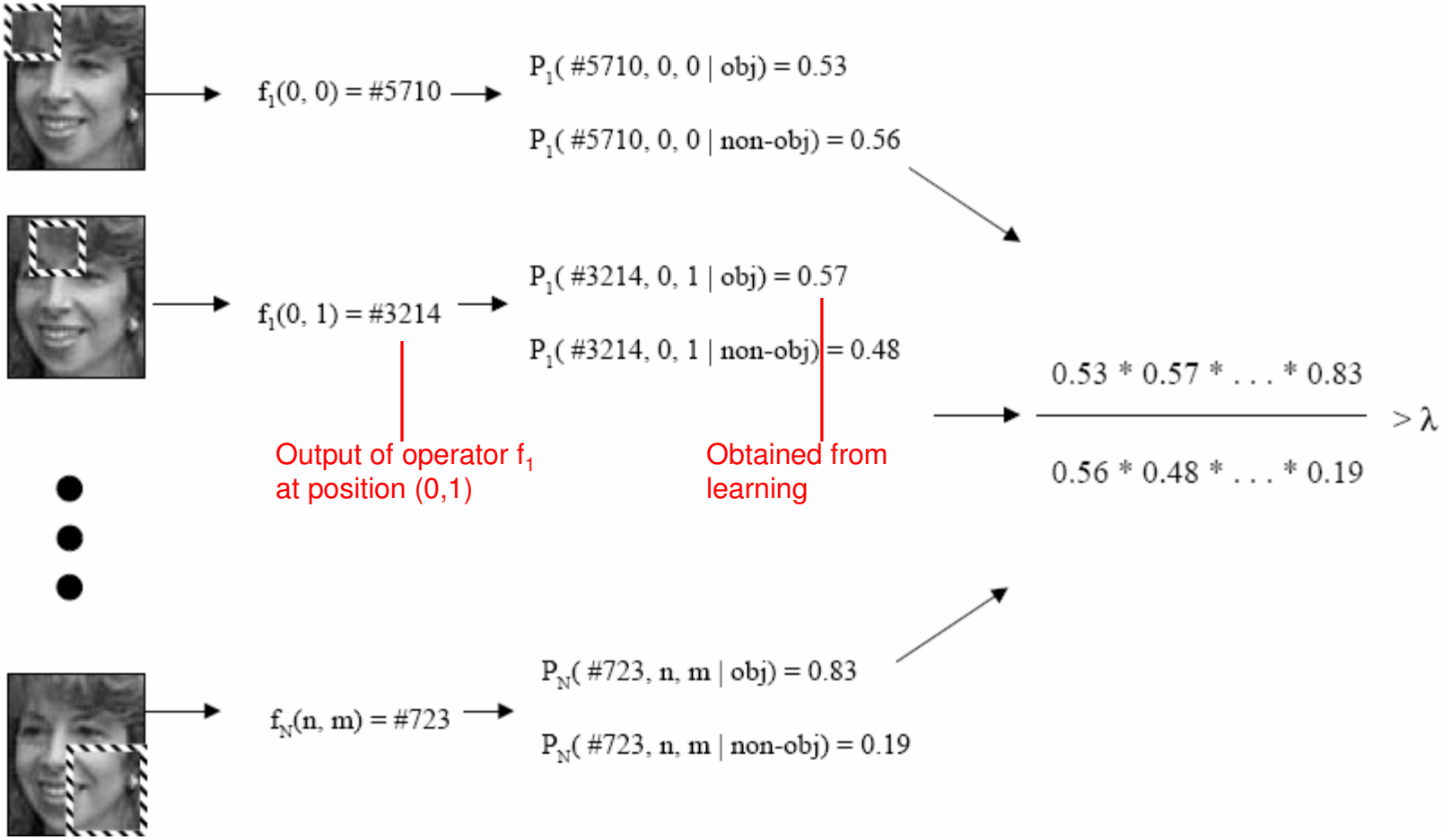
where each $part_r$ corresponds to a unique combination of k, x, and y.

- The local operators $f_k(x,y)$ are obtained through a wavelet transform of the image, encoding information like frequency and orientation of the local image patch.

- Histograms are used to model the probability distributions P_k.

- Learning consists in using a large set of face and non-face images to estimate those histograms.

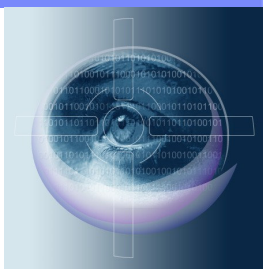
Local Statistics of Parts [Schneiderman, 2000]



Local Statistics of Parts [Schneiderman, 2000]

Also used for car detection!

Local Statistics of Parts [Schneiderman, 2000]

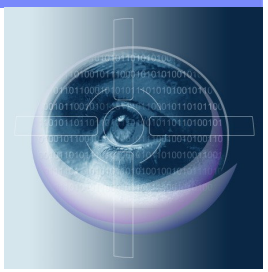


- Training: 2000 original images for each face viewpoint. For each original image they generated 400 synthetic variations by altering background, and slightly changing aspect ratio, orientation, frequency content, and position.

- Non-face training images were obtained through bootstrap

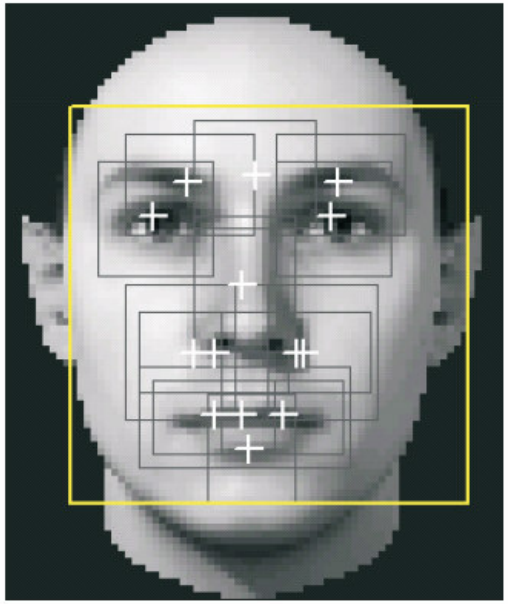
- Pros: Accurate multi-view face results
- Cons: Very Slow – 5 seconds for a 320x240 image

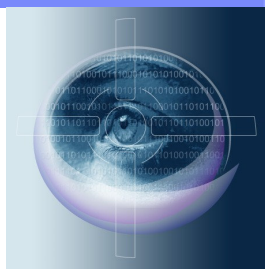
Other Appearance-based Methods



- Support Vector Machines [Osuna, 1997]
- Component-based SVMs [Heisele, 2001]
- Snow-based detector [Yang, 2002]
- Mixture of Factor Analyzers [Yang, 2000]
- Many others …

[Heisele, 2001]





Limitation

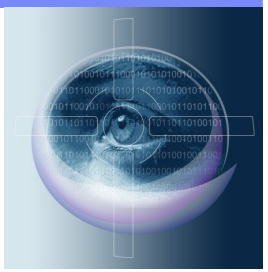
- **A common limitation of the approaches presented so far is that they are slow and can not work in real-time.**

Solution

- Viola and Jones, “Robust Real-time Face Detection”, 2001

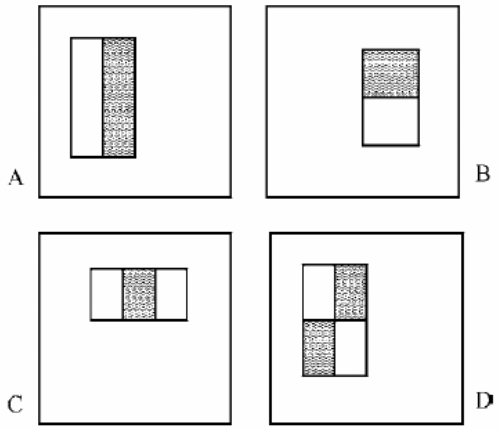
This is a breakthrough work which allowed appearance-based methods to run in real-time, while keeping the same or improved accuracy.

Real-Time Face Detection [Viola & Jones, 2001]

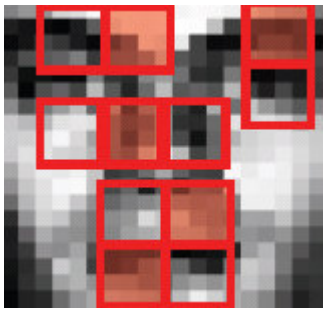


- "Rectangle Features"

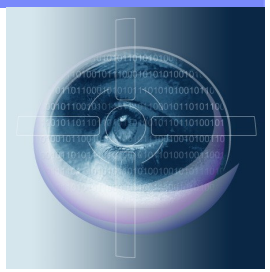
  The sum of the pixels which lie within the white rectangles are subtracted from the sum of pixels in the grey rectangles



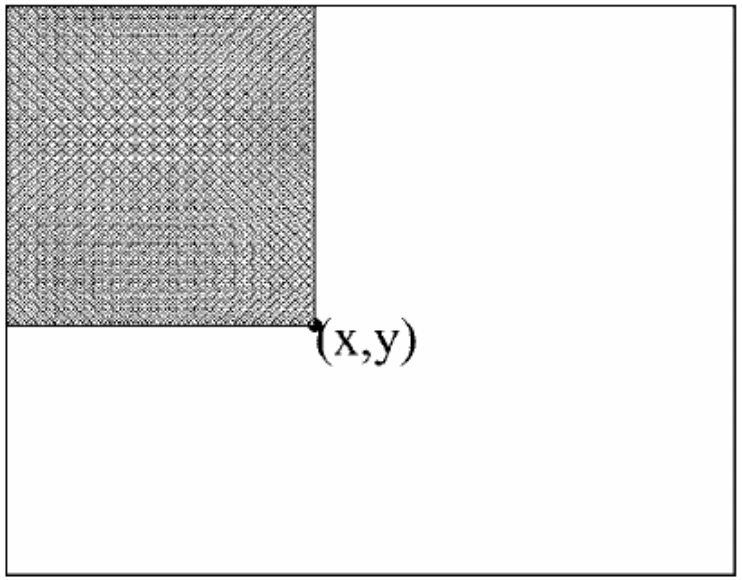
24x24



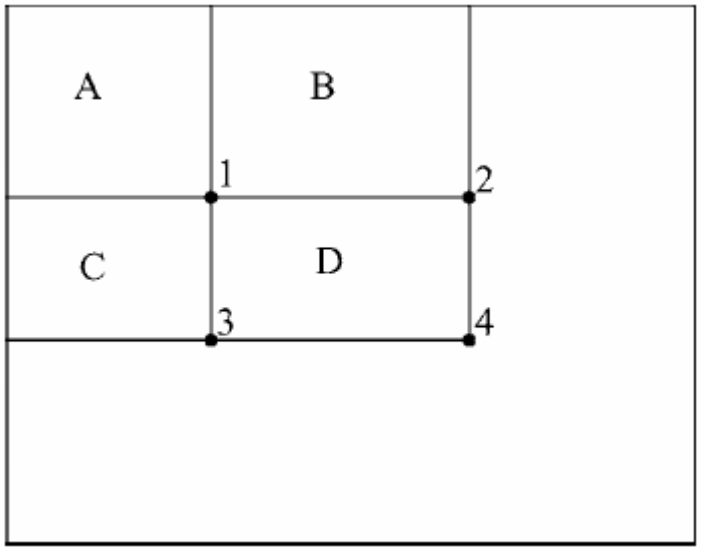
Real-Time Face Detection [Viola & Jones, 2001]



➢ Rectangle Features can be computed very rapidly using an intermediate representation called integral image.

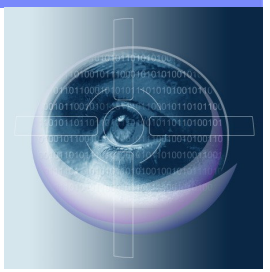


The value of the integral image at point (x,y) is the sum of all the pixels above and to the left.



The sum of pixels within D can be computed as 4+1-(2+3)

Real-Time Face Detection [Viola & Jones, 2001]



- Scan image at multiple positions and scales as in previous approaches.

Apply Adaboost strong classifier (which is based on Rectangle Features) to decide whether the search window contains a face or not:

- **Adaboost Strong Classifier**: linear combination of weak classifiers.

H(x) = 1 if window x has a face and 0 otherwise

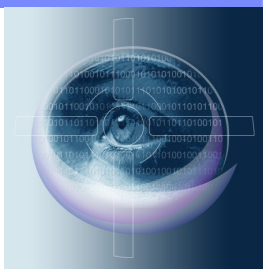
$$H(x) = \begin{cases} 1 \text{ if } \sum_{t=1}^{T} \alpha_t h_t(x) \geq \phi \\ 0 \text{ otherwise} \end{cases}$$

ϕ → Threshold

$h_t(x)$ → Weak Classifier

α_t → Weights

Real-Time Face Detection [Viola & Jones, 2001]



- **Each week classifier corresponds to a single Rectangle Feature:**

$$h_j(x) = \begin{cases} 1 \text{ if } p_j f_j(x) < p_j \theta_j \\ 0 \text{ otherwise} \end{cases}$$

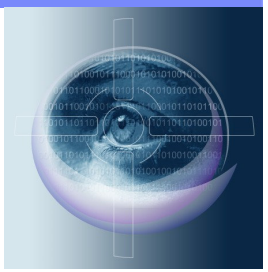
Threshold

Sign

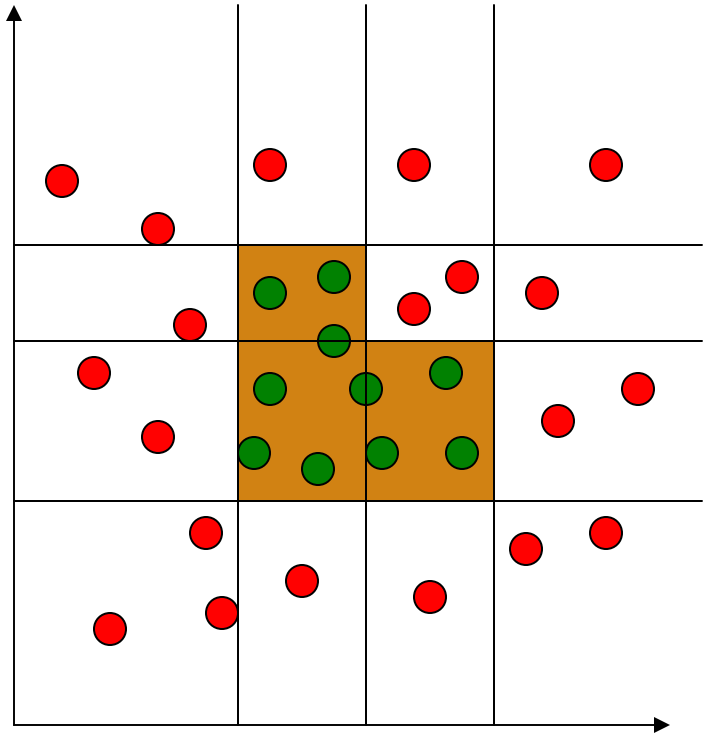
Rectangle Feature

- There are so many configurations of Rectangle Features. **How can we choose the features (weak classifiers) to form the Adaboost Strong Classifier?**

Adaboost Learning

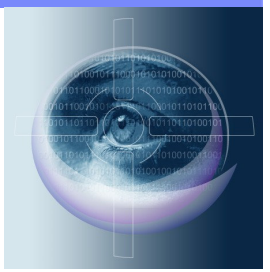


➢ Adaboost ensembles many weak classifiers into one single strong classifier



- Initialize sample weights
- For each cycle:
  - ➢ Find a classifier/rectangle feature that performs well on the weighted samples
  - ➢ Increase weights of misclassified examples
- Return a weighted combination of classifiers

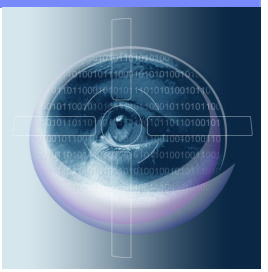
Real-Time Face Detection [Viola & Jones, 2001]



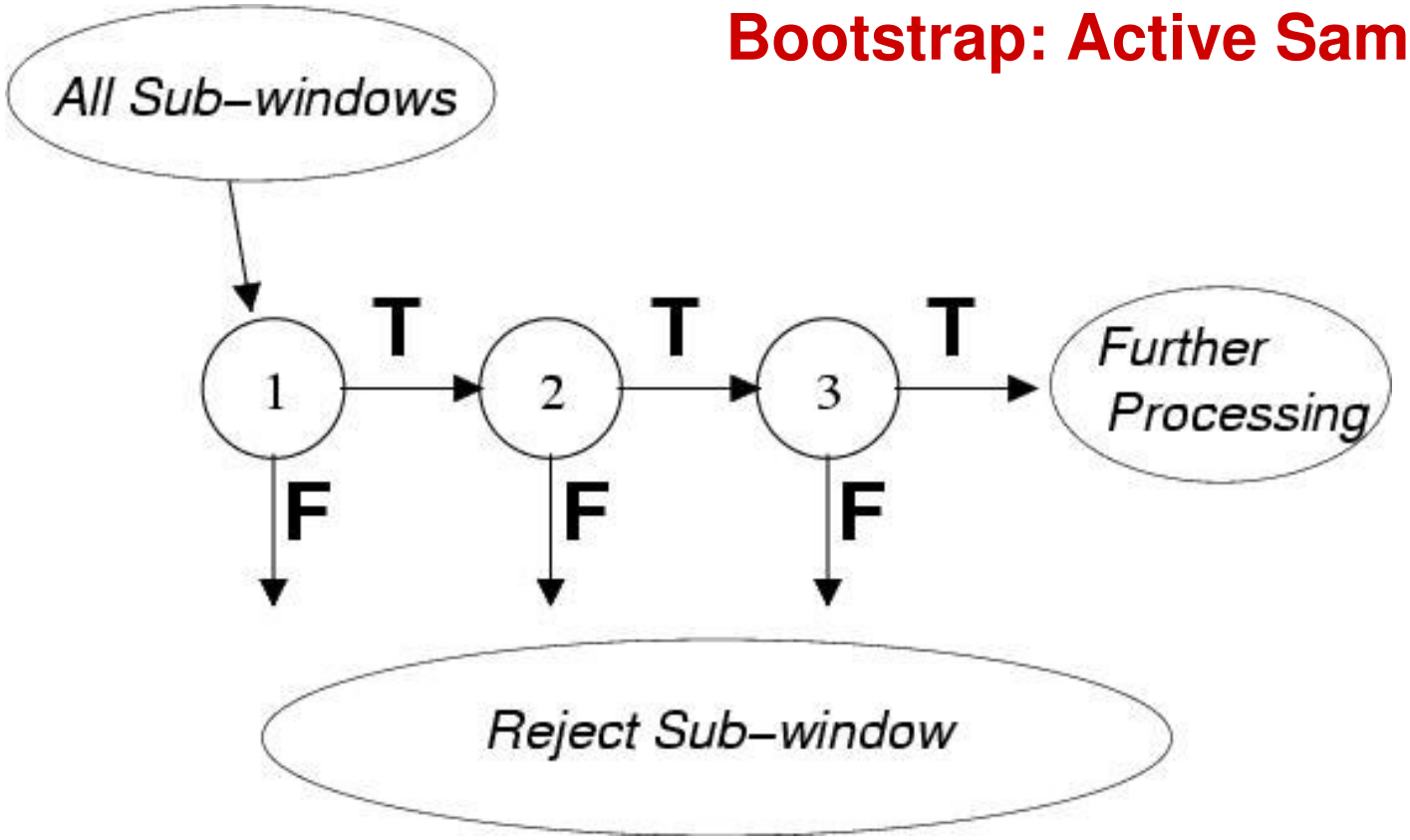
Attentional Cascade

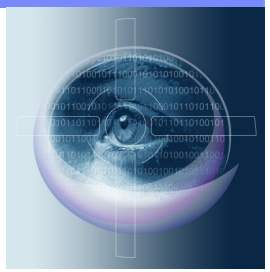
- We **start with simple classifiers** which reject many of the negative sub-windows while detecting almost all positive sub-windows
- Positive results from the first classifier triggers the evaluation of a second (more complex) classifier, and so on
- A negative outcome at any point leads to the **immediate rejection** of the sub-window

Real-Time Face Detection [Viola & Jones, 2001]



Bootstrap: Active Sampling

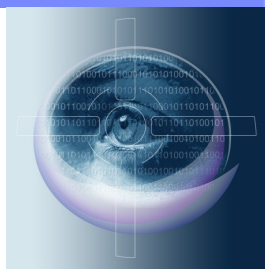




The attentional cascade is based on the assumption that within an image, most sub-images are non-face instances.

Is this assumption always valid???

Real-Time Face Detection [Viola & Jones, 2001]



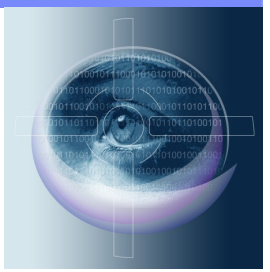
Summary

- Rectangle Features
- Adaboost for feature selection
- Attentional Cascade

Pros and Cons

- Pros:
  - Reliable face detector that runs in real-time
- Cons:
  - Requires thousands of training samples to learn a robust classifier
  - Training may take order of weeks!

Real-Time Face Detection [Viola & Jones, 2001]



Source Code: OPENCV Implementation

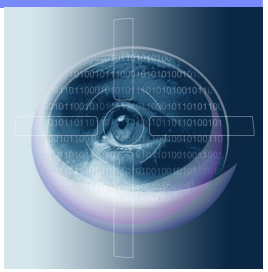
➢ Rainer Lienhart, Alexander Kuranov, Vadim Pisarevsky. Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. *DAGM'03, 25th Pattern Recognition Symposium*, Madgeburg, Germany, pp. 297-304, Sep. 2003.

Guide for Opencv training of boosted classifiers:

- Opencv documentation

- Clik here for more detailed implementation aspects!

Variations and Recent Methods



- Real-time multi-view face detection [Jones, TR 2003]
- Occlusion Handling [Lin et al, ECCV 2004]

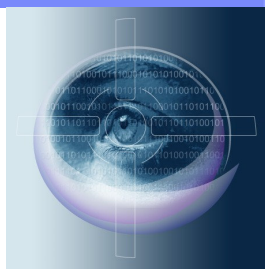
Improvements - Learning Process:

- Real Adaboost [Bo Wu, FG 2004]
- Kullback-Leibler Adaboost [Liu, CVPR 2003]
- Vector Boosting [Huang ICCV 2005]
- FloatBoost, Gentle Adaboost, etc.

Improvements – Feature Level

- Fisher Discriminant Analysis [Wang and Ji, CVPR 2005]
- Edge Orientation Histograms [Levi and Weiss, CVPR 2004]
- Sparse Feature Set [Huang, FG 2006]
- etc.

Benchmark Datasets



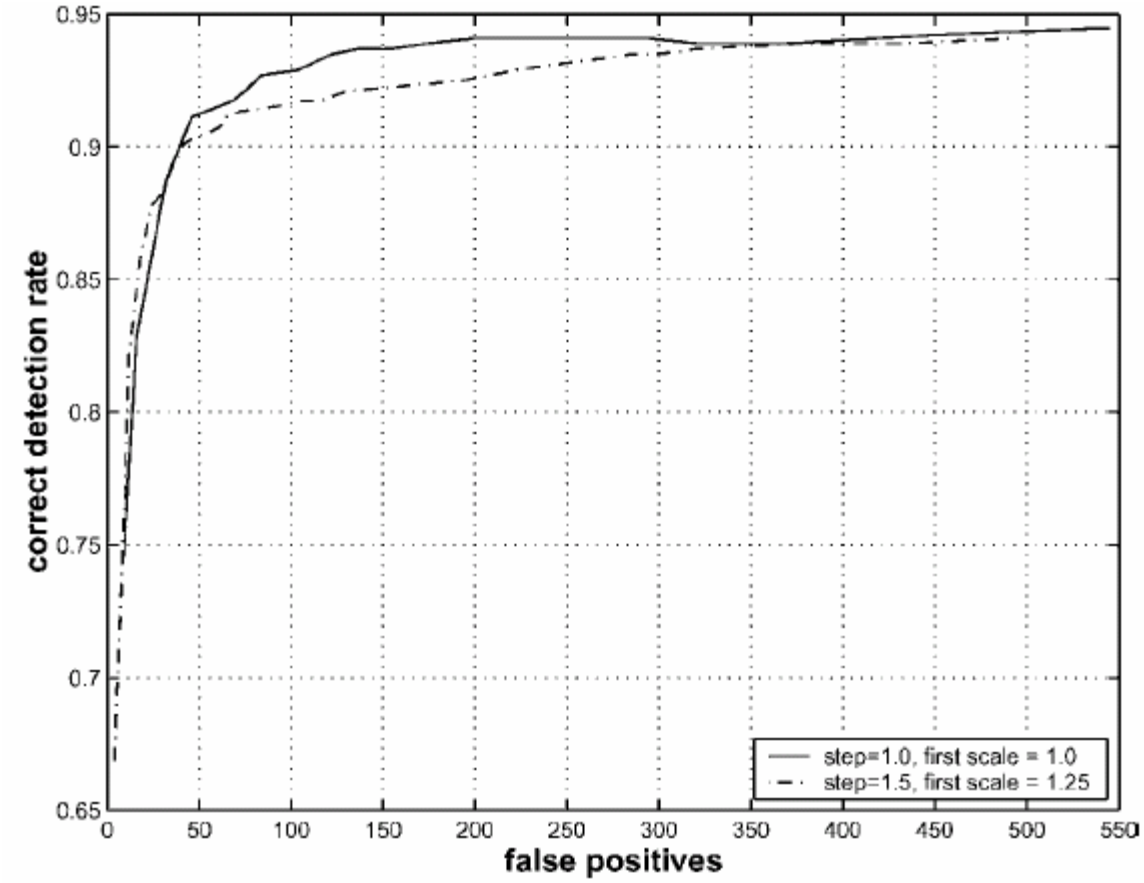
Every face detector paper compares results using the following datasets:

- CMU+MIT dataset (http://vasc.ri.cmu.edu/NNFaceDetector/): 130 gray scale images with a total of 507 frontal faces.

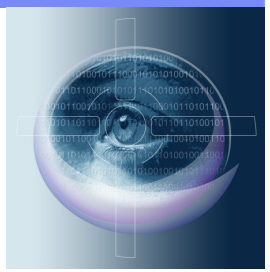
- CMU profile face test set (http://vasc.ri.cmu.edu/NNFaceDetector/): 208 images with faces in profile views.

Benchmark Datasets

ROC Curves



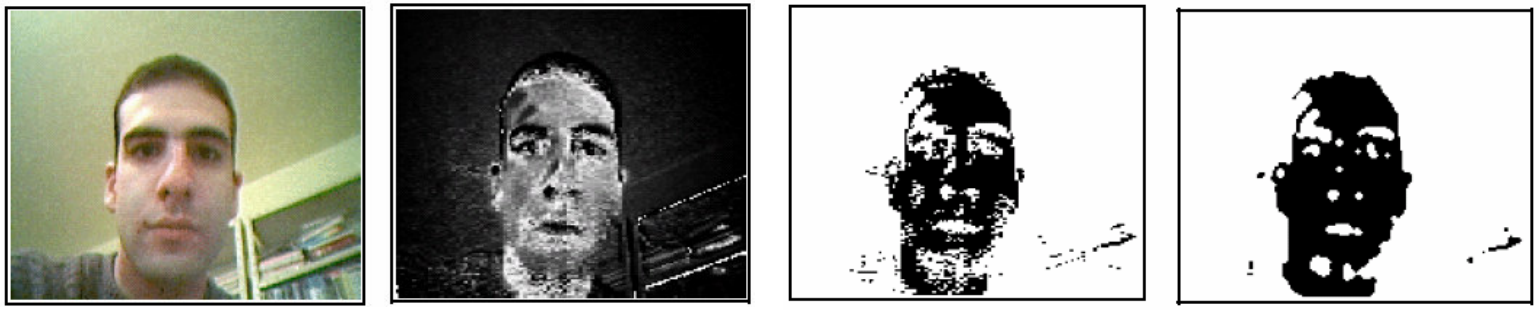
Outline



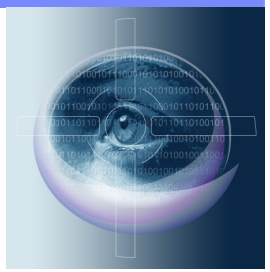
- Motivation
- Face Detection
  - Appearance-Based Learning
  - **Other Modalities**
- Face Tracking
- The IBM Face Capture System

Color Information

- Skin-Color Detection



- Pros: Simple to Implement, View-Invariant
- Cons: Sensitive to Lighting Variations, other skin-color regions
- See [Jones and Rehg, 2002]



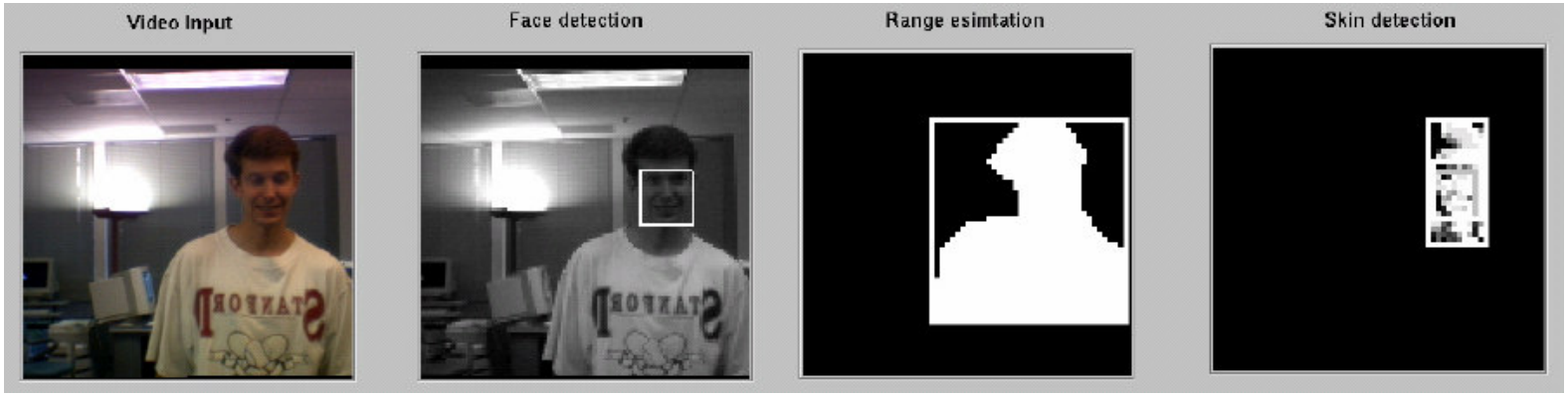
Motion Information

- Background Subtraction
- Reduce Search Space Dramatically

Depth Information

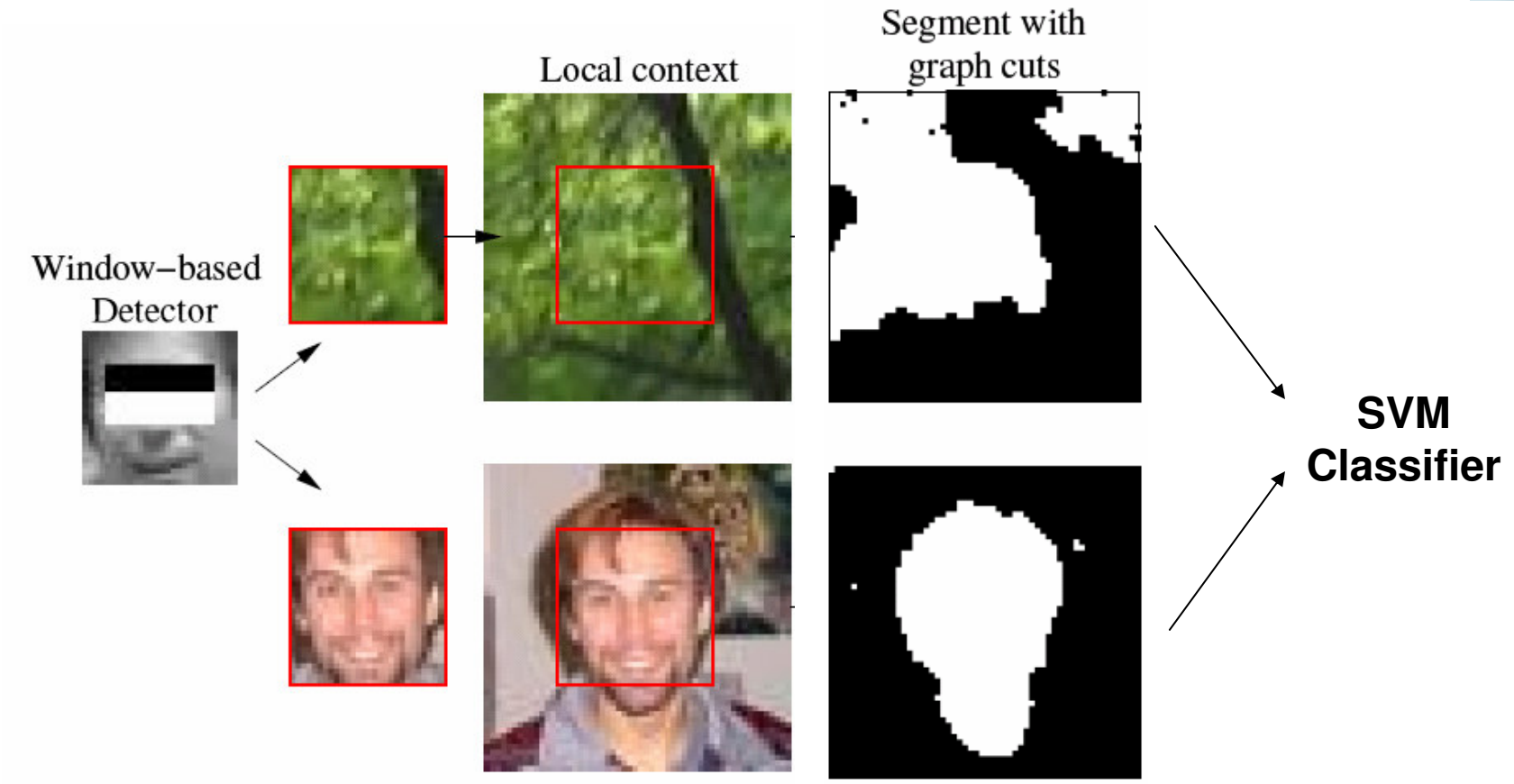
- e.g., from stereo cameras
- also help to reduce search space

[Trevor Darrell et al, FG 1998]



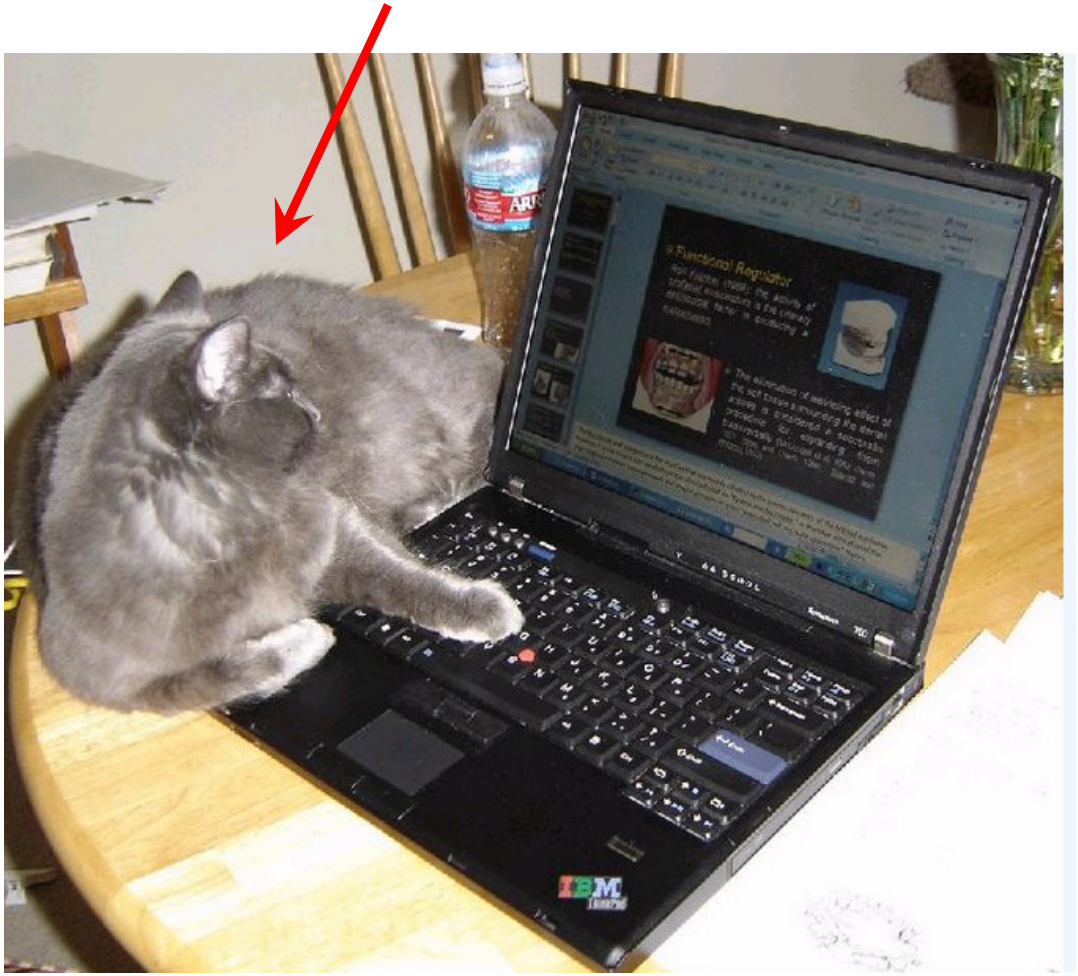
Context Information

- Using Segmentation to Verify Object Hypothesis [Ramanan, CVPR 2007]

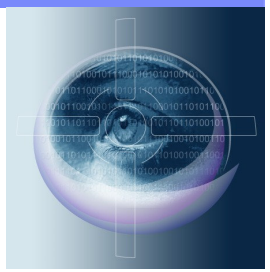


Sometimes context information may not be helpful…

Is this a cat or an IBM researcher?

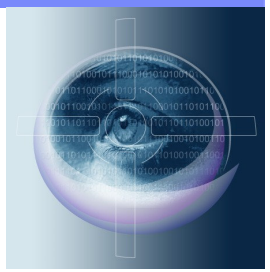


Outline



- Motivation
- Face Detection
  - Appearance-Based Learning
  - Other Modalities
- **Face Tracking**
- The IBM Face Capture System

Face Tracking

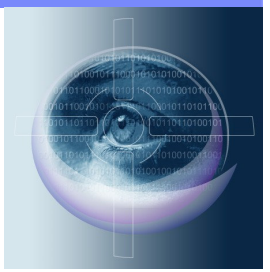


- Once the face is detected, a variety of tracking methods can be used to track the face along the video: **Condensation**, **Mean-Shift**, etc.

- Problems: Drifting, Low frame rate, how to decide the track is finished, etc.

- Recent approaches: **Face tracking by means of continuous detection** (see e.g., Froba 2004)

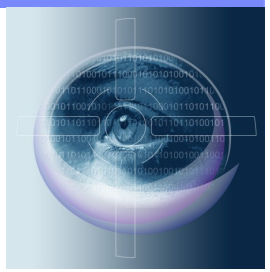
Face Tracking



- Apply face detector and tracker at every frame and use the output of the detector to guide tracking and avoid drifting problems.

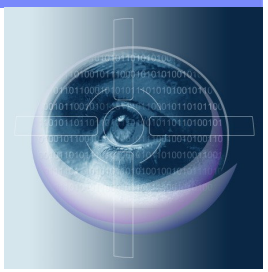
- Problem: this process can be slow!

Face Tracking

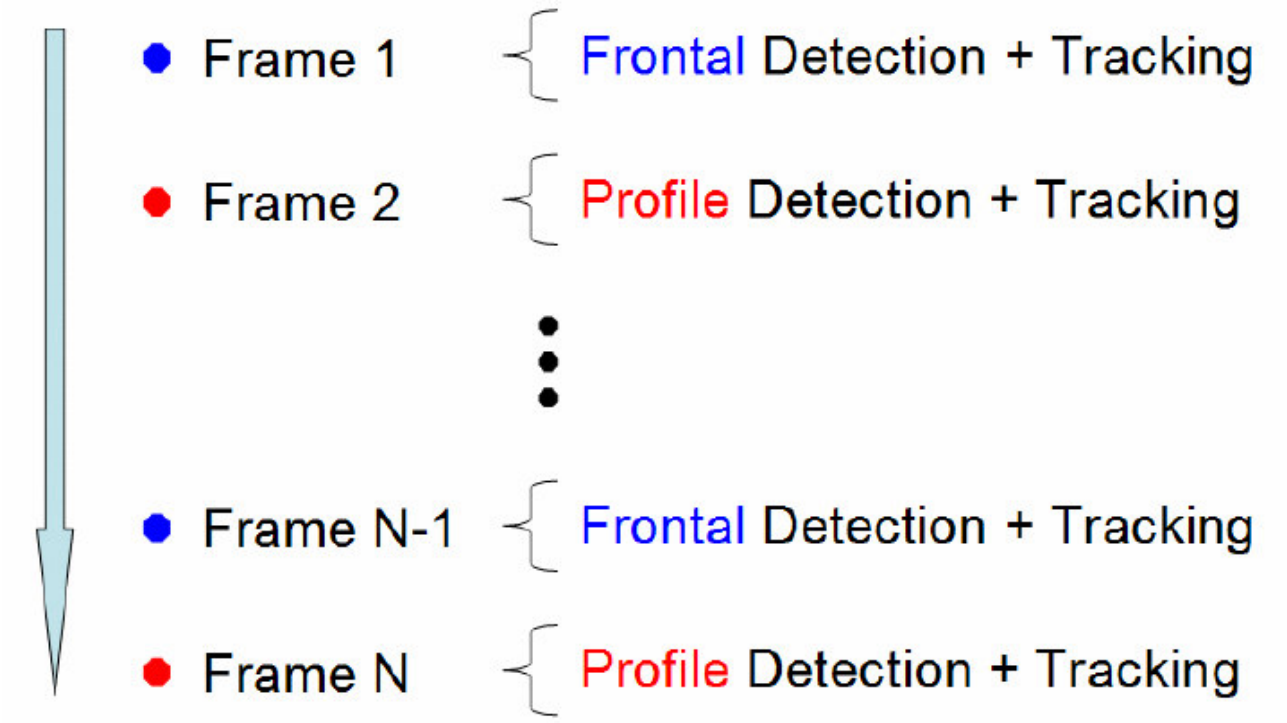


- Making the tracker faster (see Feris, VS'07):
  - Apply face detector only at a specific range of scales.
  - Constrain detection and tracking only to foreground regions detected by Background Subtraction.
  - Interleave frontal and profile detectors along the video sequence.

Face Tracking

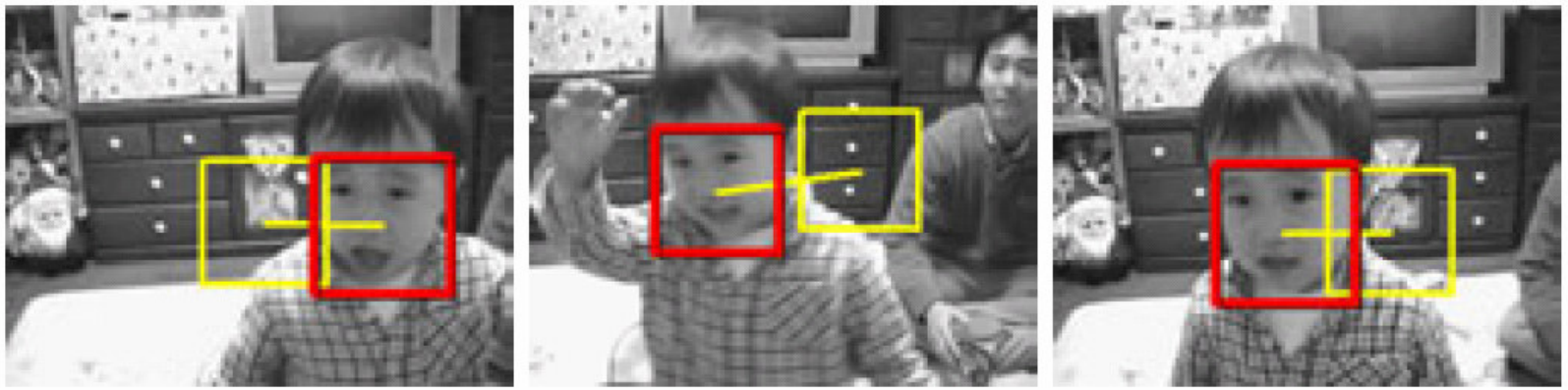


➢ Interleaving view-based classifiers along the video to save frame rate.

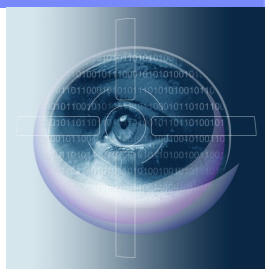


Face Tracking

- Recent Tracking by Detection Approaches:
  - Ensemble Tracking [Avidan, CVPR 2005]
  - Tracking in Low Frame Rate Video: A Cascade Particle Filter with Discriminative Observers of Different Lifespans [Li, CVPR 2007]

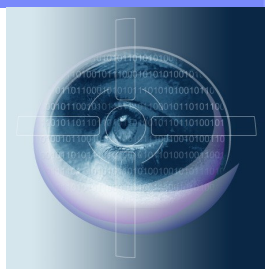


Outline



- Motivation
- Face Detection
  - Appearance-Based Learning
  - Other Modalities
- Face Tracking
- **The IBM Face Capture System**

See [Feris et al, Capturing People in Surveillance Video, VS'07]

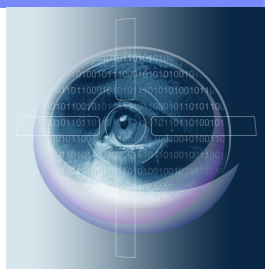


Viola & Jones Face Detector:

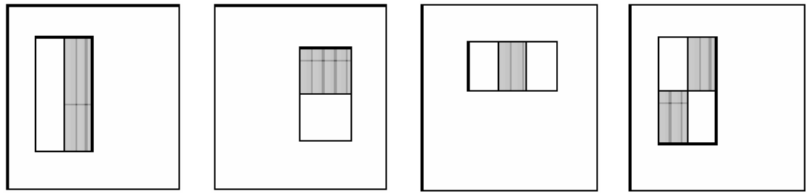
- Great for frontal views, but limited to handle profile views
- Requires thousands of samples to learn a robust classifier

- Choice of features is critical.

How to Choose the Best Features?



- Pool of features with all possible configurations



- Feature Selection Using Adaboost

Problems:

- Very large set: training slow (160000 features for 24x24)
- Discrete-domain features
- Limited to Integrate multiple types of features

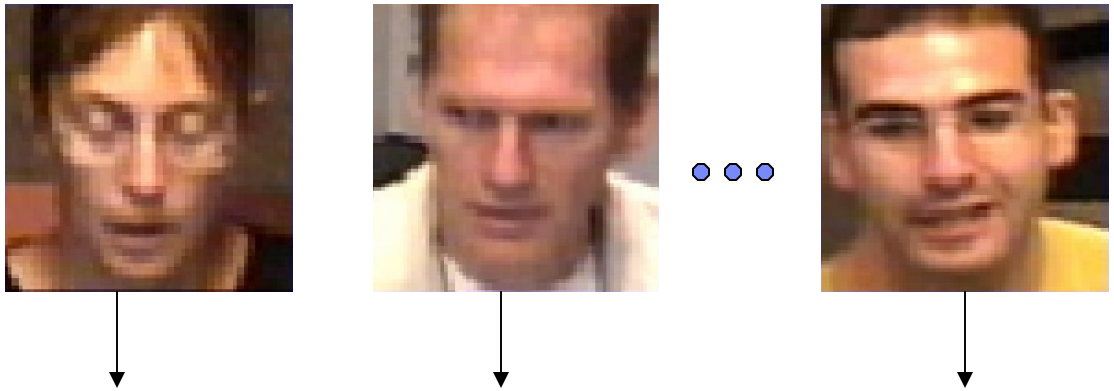
Observation: Selected Features “fit” the face appearance/geometry

Main Idea:

Rather than considering all possible configurations in feature pool…

Learn ***specific features*** that match the structure of each sample

IBM Face Detector: Big Picture



Adaptive,
Continuous Domain

Specific Features

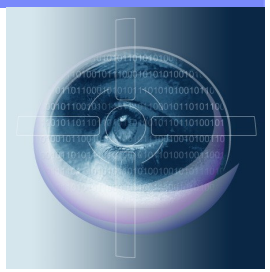
Specific Features

Specific Features

Stage 1
Wavelet Networks

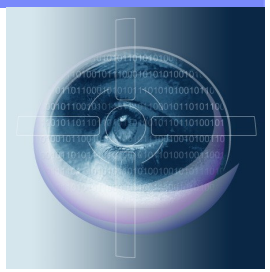
General Features

Stage 2
Adaboost Learning



Learning Specific Features

Wavelet Networks



Given N wavelet features distributed along the face image:

$$\Psi = \{\psi_{\mathbf{n}_1}, \psi_{\mathbf{n}_2}, \ldots, \psi_{\mathbf{n}_N}\}$$

Problem: Determine parameters of each wavelet (position, scale and orientation) that "fits" the face structure

$$\mathbf{n}_i = (c_x, c_y, \theta, s_x, s_y)$$

Position Orientation Scale

Wavelet Networks

$$E = \min_{n_i, w_i \forall i} \| f - (\sum_i w_i \psi_{n_i}) \|^2$$

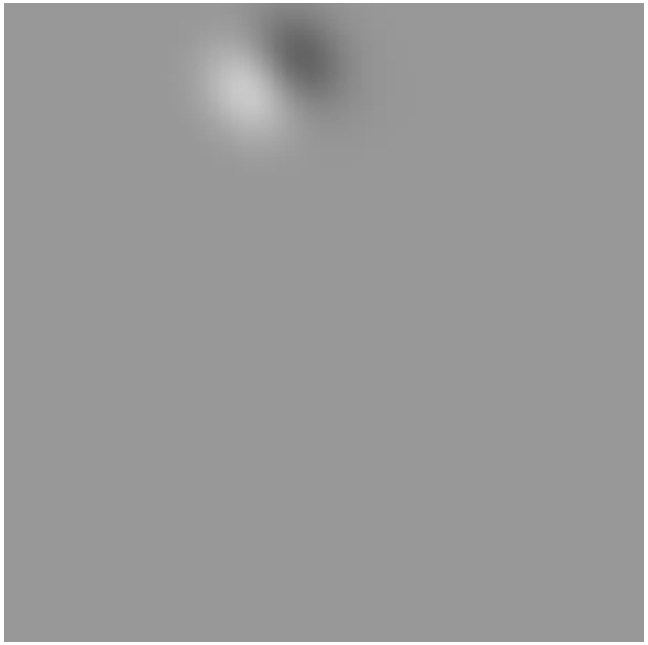
Original Image

Wavelet Representation

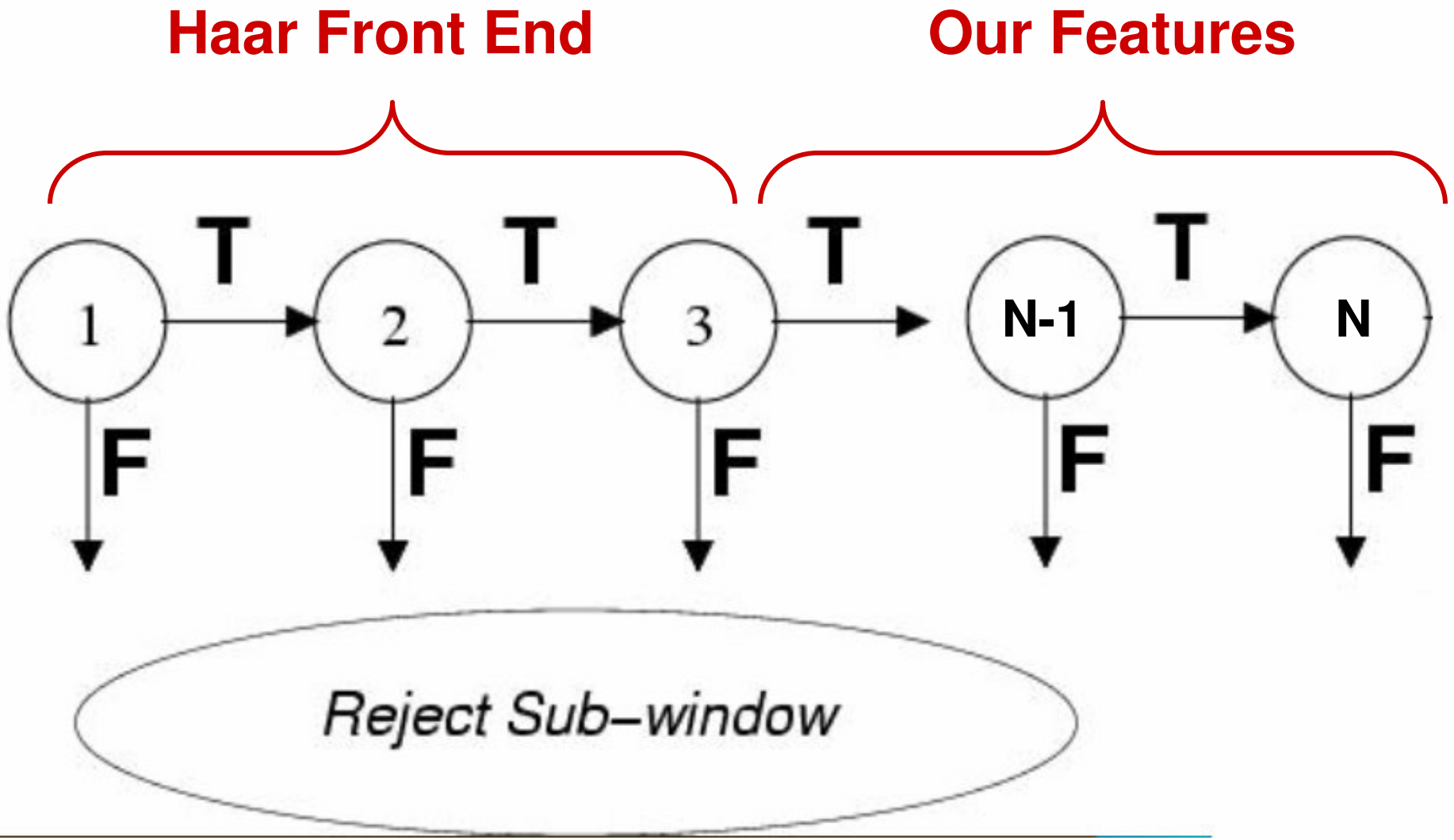
Original Image Wavelet Representation Final Feature Positions

Levenberg-Marquardt Optimization – Continuous-domain features

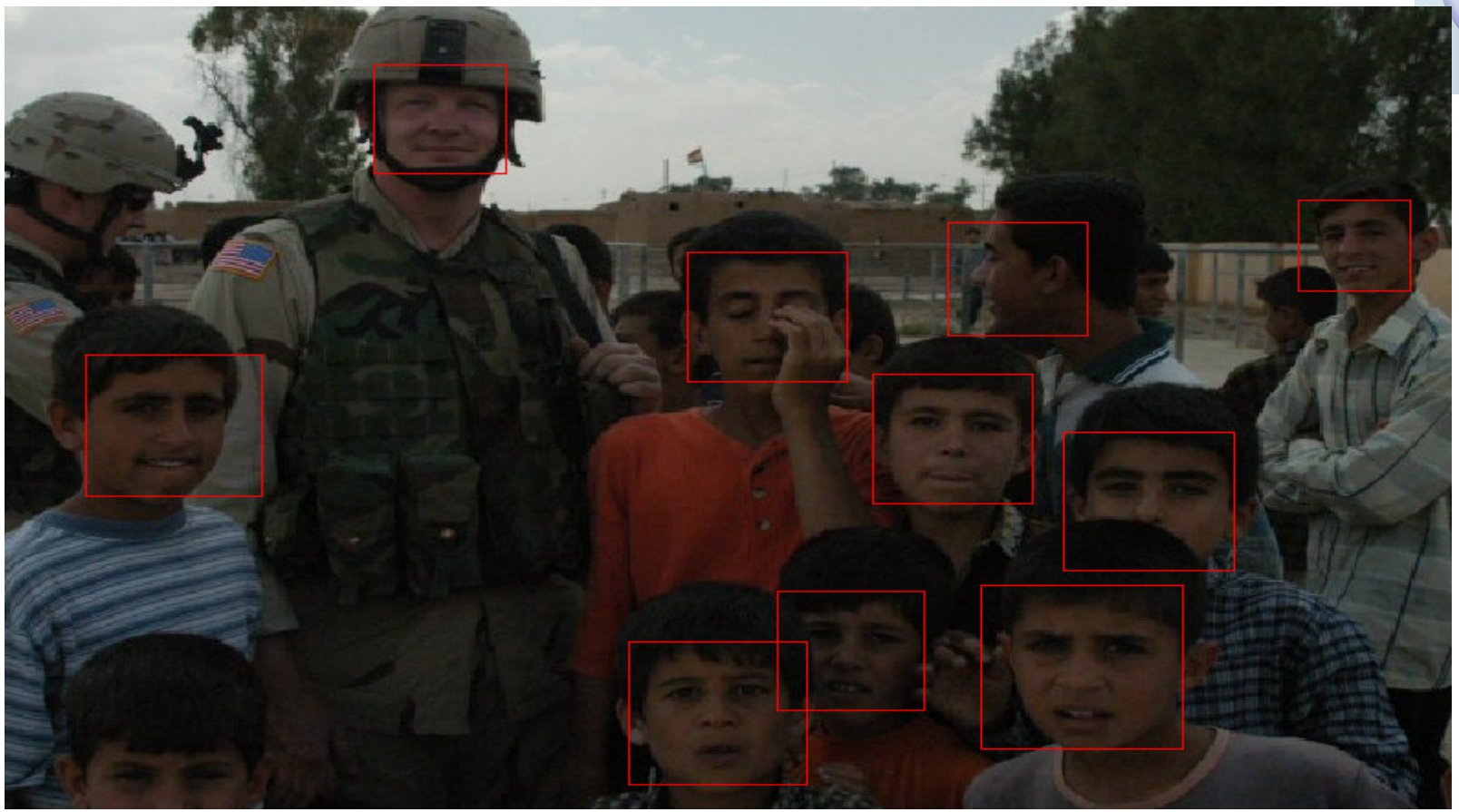
Wavelet Networks

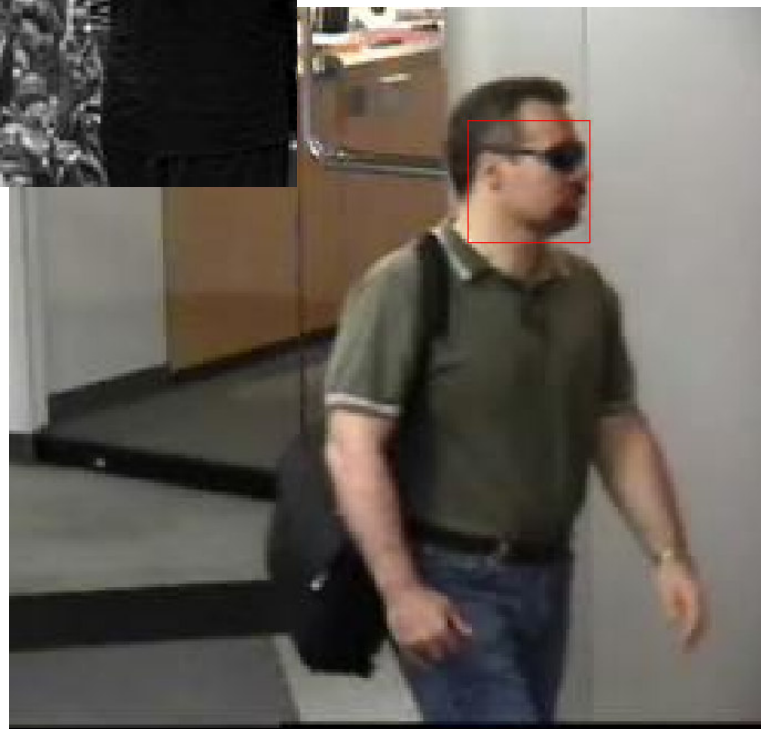


Efficient Detector

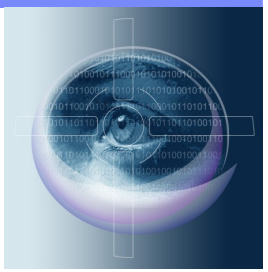


Several Demo Images



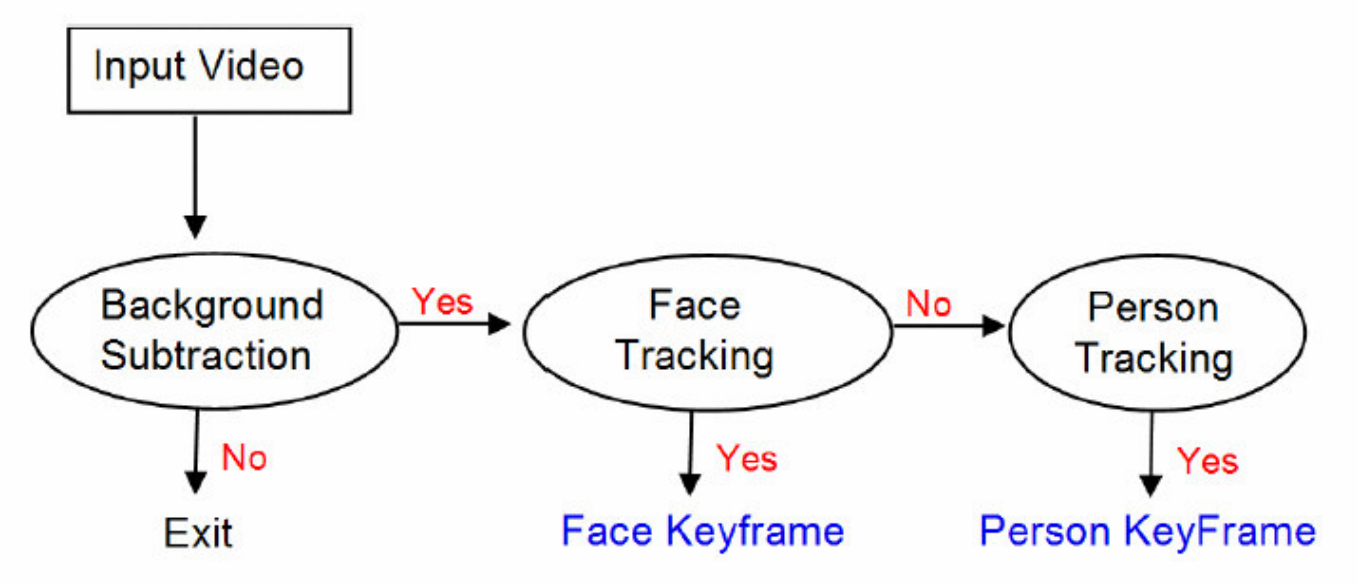


IBM Face Tracking

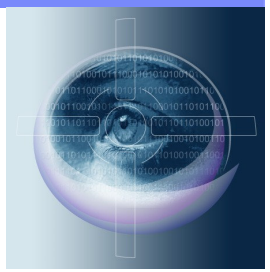


- Correlation-based tracker by continuous detection.
- Interleaving of frontal and profile detectors along the video sequence

Reducing False Negatives



Future Directions



- Adaptation and Online Learning:

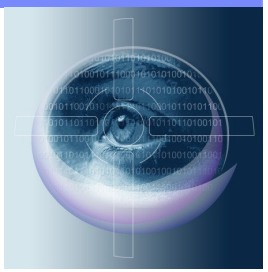
Chang Huang, Haizhou Ai, Takayoshi Yamashita, Shihong Lao, Masato Kawade, **Incremental Learning of Boosted Face Detector**, ICCV 2007

- Faster Learning

Minh-Tri Pham, Tat-Jen Cham, **Fast training and selection of Haar features using statistics in boosting-based face detection,** ICCV 2007.

- Training from millions of images?

HOMEWORK



- Review Paper : Chang HUANG, Haizhou AI, Yuan LI, Shihong LAO, **Vector Boosting for Rotation Invariant Multi-View Face Detection**, The IEEE International Conference on Computer Vision (ICCV-05), pp.446-453, Beijing, China, Oct 17-20, 2005

- Due Date: March 24th, 2008