Face Detection and Recognition: an Overview

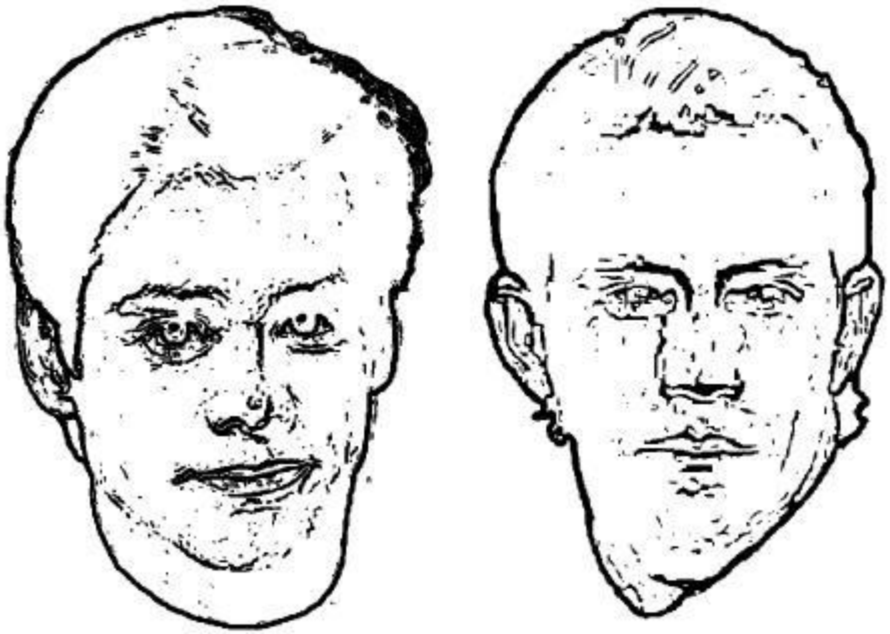
Face Recognition and Detection

The “Margaret Thatcher Illusion”, by Peter Thompson

Face detection and recognition

We can recognize FAMILIAR faces from extremely low resolution pictures.

How this is done? – We do not have clear idea – but it points to the minimization of processed information

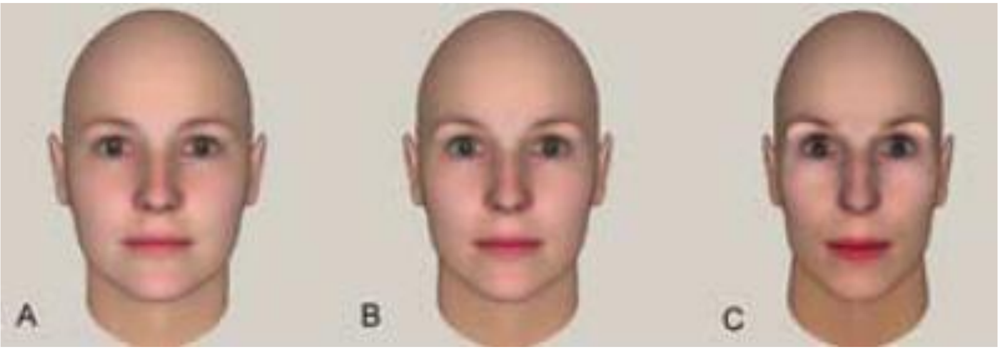


Contour information is not enough

Face is processed somehow as a ”whole” and not as composed by parts. From the combined picture on the left we see new face, when we split it we recognize other faces

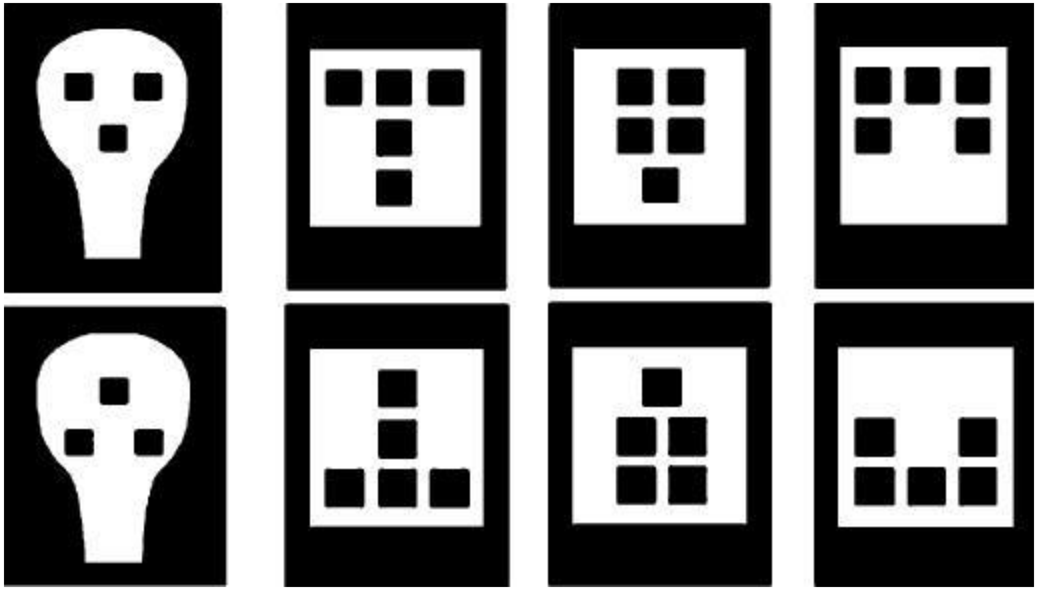
Eyebrows are very important for the identification of faces

Faces can be recognized despite extreme distortions

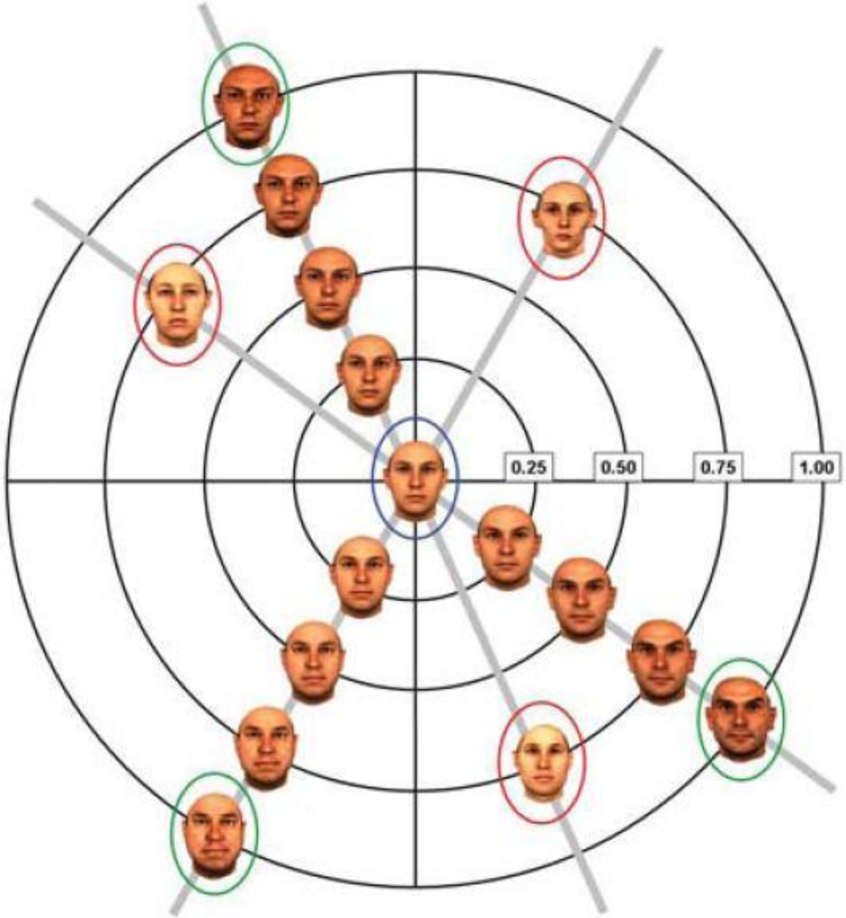


Faces seem to be encoded in memory in exaggerated. caricature way:

A) Average face (averaged from a number of persons
B) Some typical face
C) Face created by taking bid deviation from average
Such faces are recognized even better than typical ones



Newborn babies turn more attention to more face-like objects (upper row) than not face-like



Faces and antifaces: If face within green circle is observed for some time the center one will not be correctly recognized but as one in the red circle (more distance from the center means more differences) This means that there is some kind of prototype encoding and tuning to it

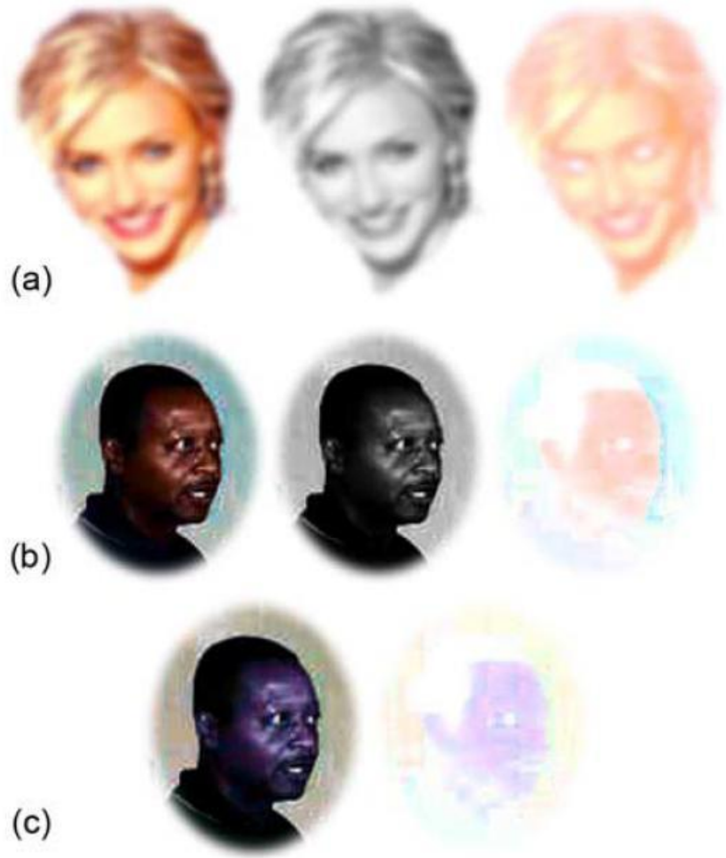
Impact of skin pigmentation

Row 1: Faces differ only in shape

Row 2: Faces differ only in skin pigmentation but not shape

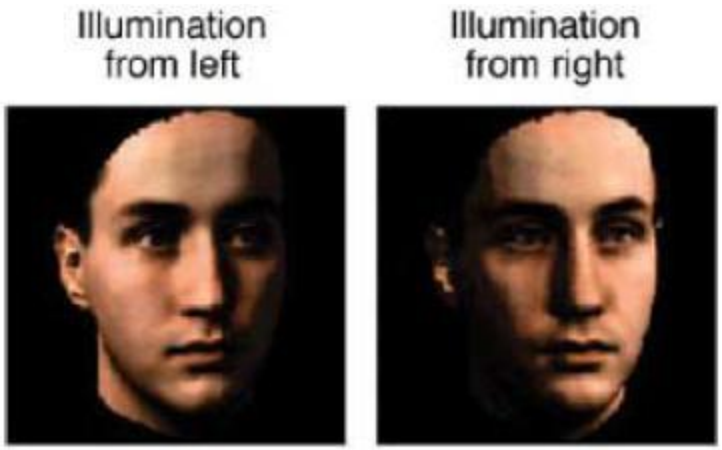
Row 3: Faces differ in shape and pigmentation

We see that pigmentation has significant impact (row 2)

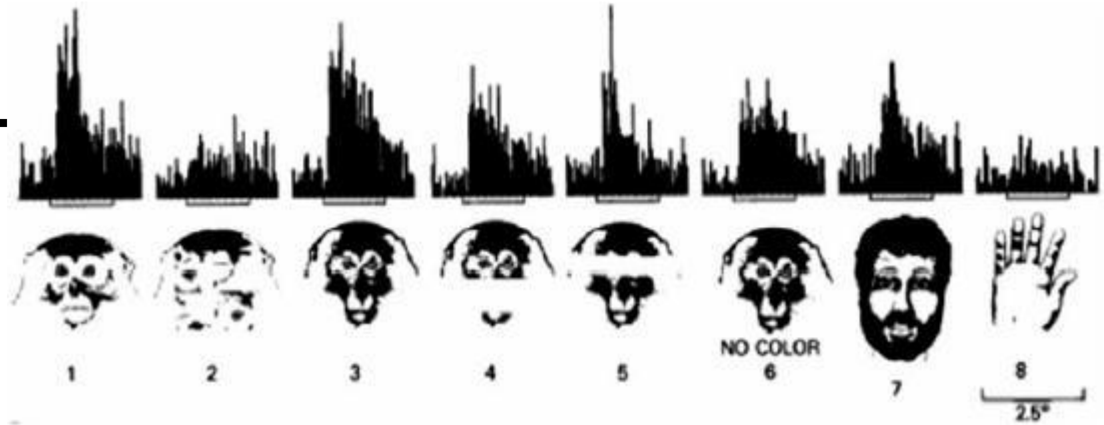


Color helps: Left original
Middle black and white
Right color only, eyes can be located more precisely

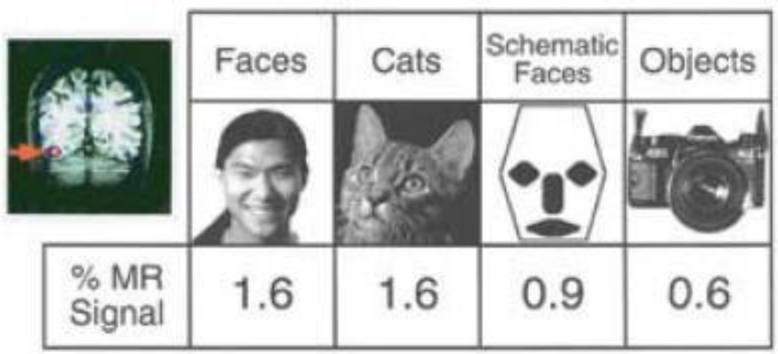
From negative picture it is impossible to identify faces



Face recognition is strongly compensated for the direction of ilumination, pictures above are easily recognized as same person



Response of neural cell of monkey in the face processing area of the brain. Response to something like face is much more stronger than for hand. (But remember that milions and milions of cells are processing at the same time)



Measurement from human brain: signal from face-like picture is much stronger than from other objects

The examples shown for faces indicate how sophisticated is information processing in biological systems.

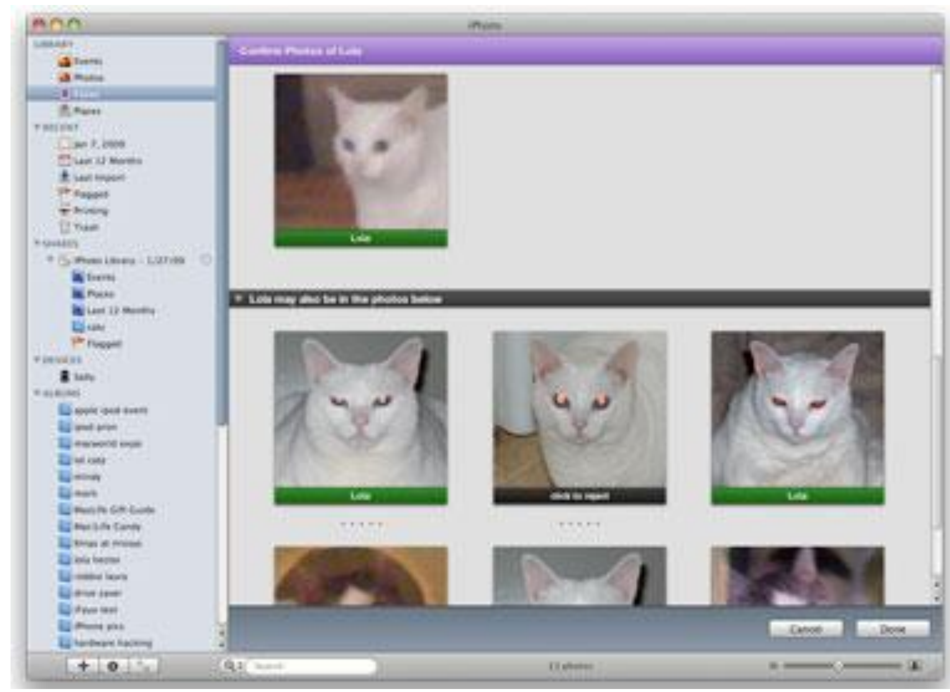
What is very amazing is getting correct results despite extreme distortions. For the most part, we do not know how this is done and we have difficulty in thinking how to develop algorithms which would have similar capabilities.

Consumer application: Apple iPhoto

http://www.apple.com/ilife/iphoto/

Consumer application: Apple iPhoto

Can be trained to recognize pets!

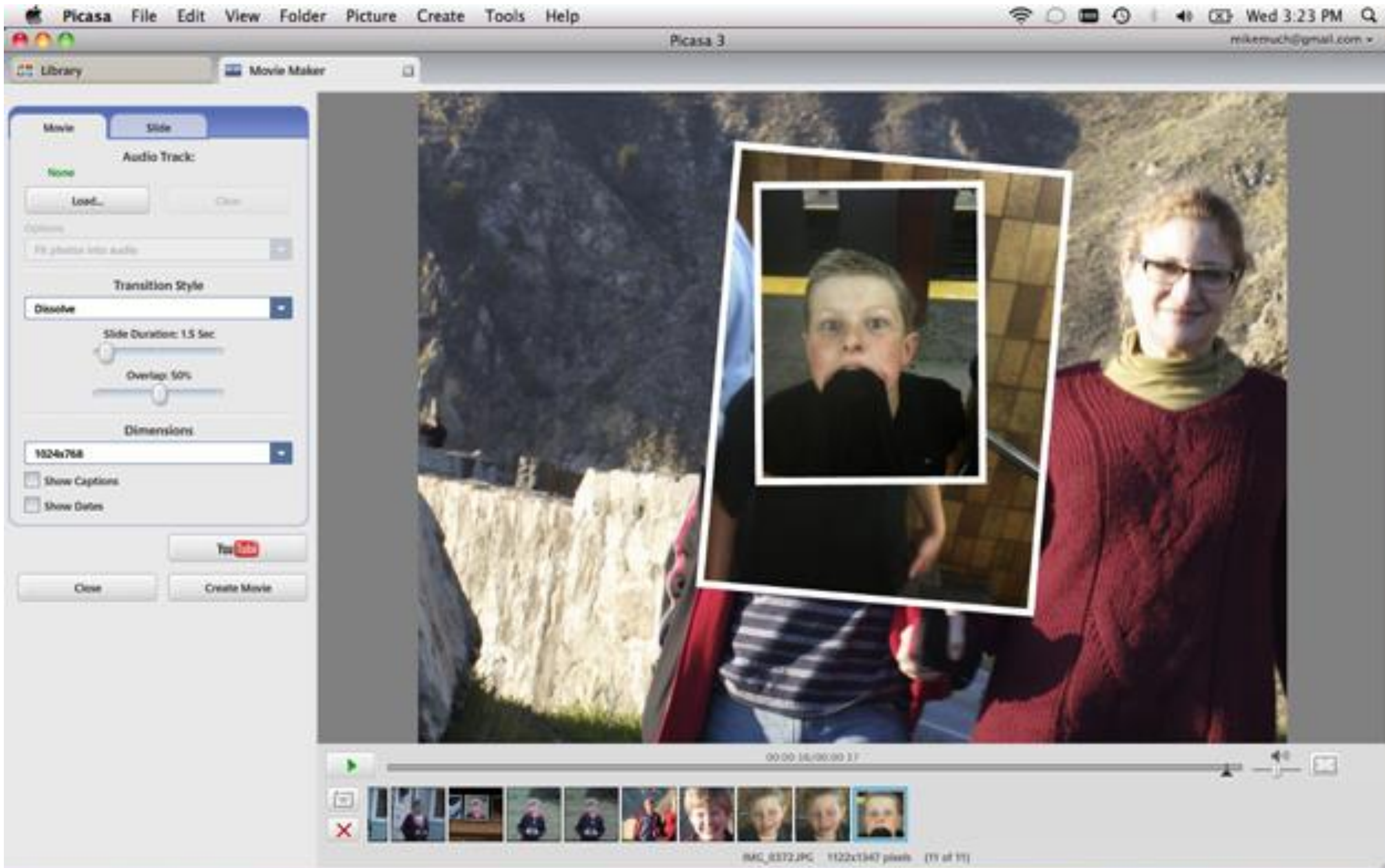


http://www.maclife.com/article/news/iphotos_faces_recognizes_cats

Consumer application: Apple iPhoto

[Things iPhoto thinks are faces]

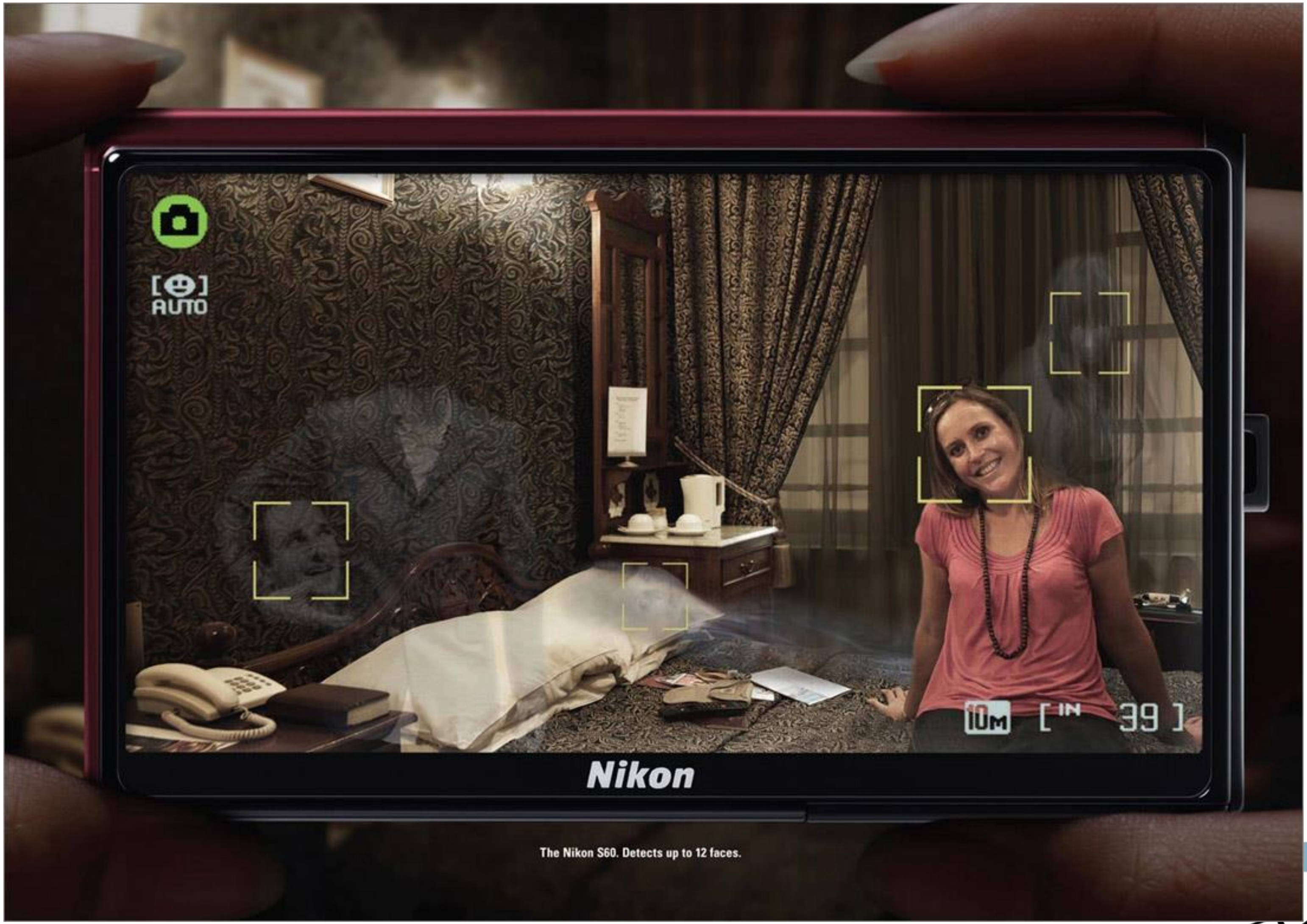
Consumer Application: Picasa



Demo FaceMovie
http://www.youtube.com/watch?feature=player_embedded&v=fLQtssJDMMc

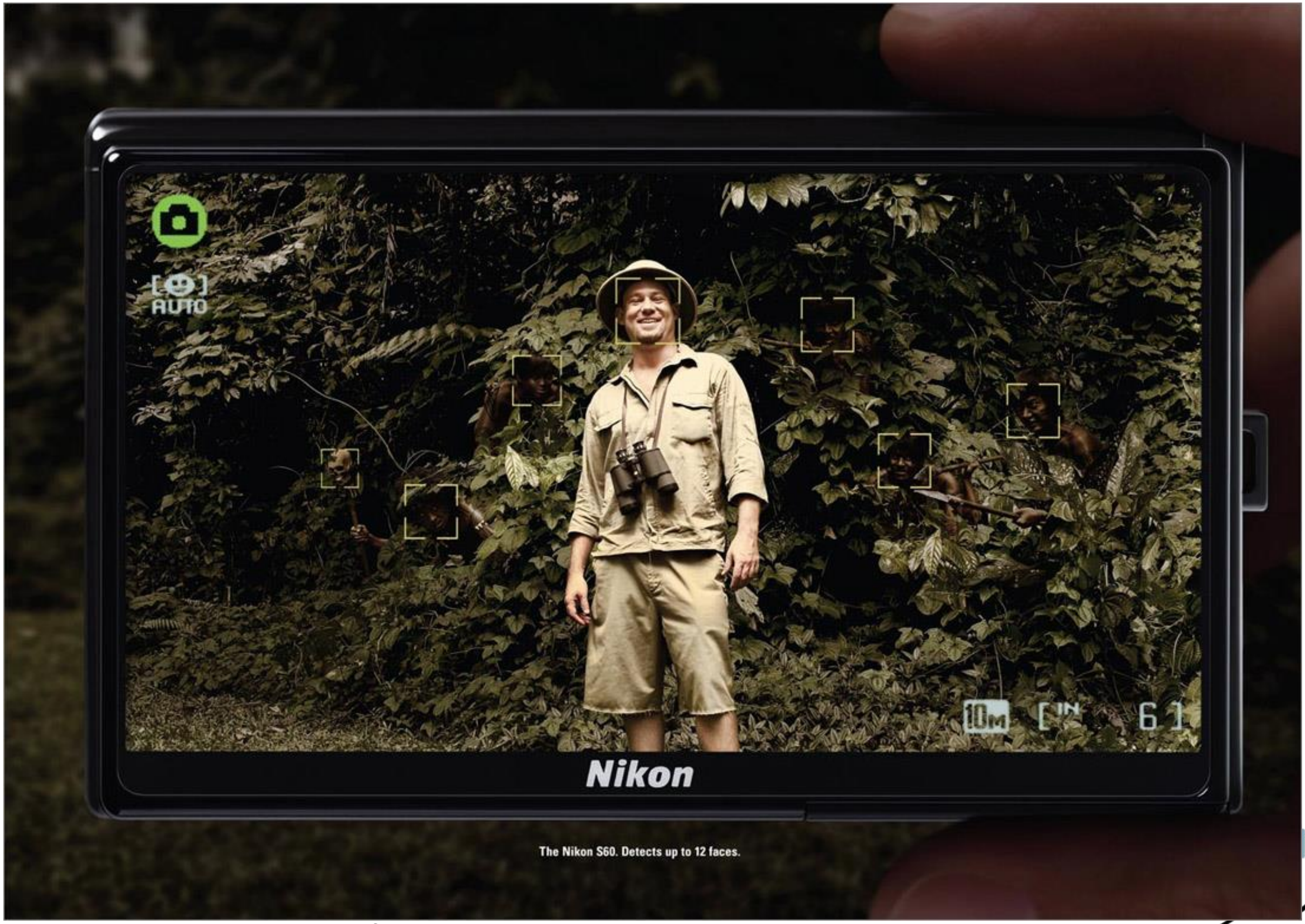
Funny Nikon ads

"The Nikon S60 detects up to 12 faces."

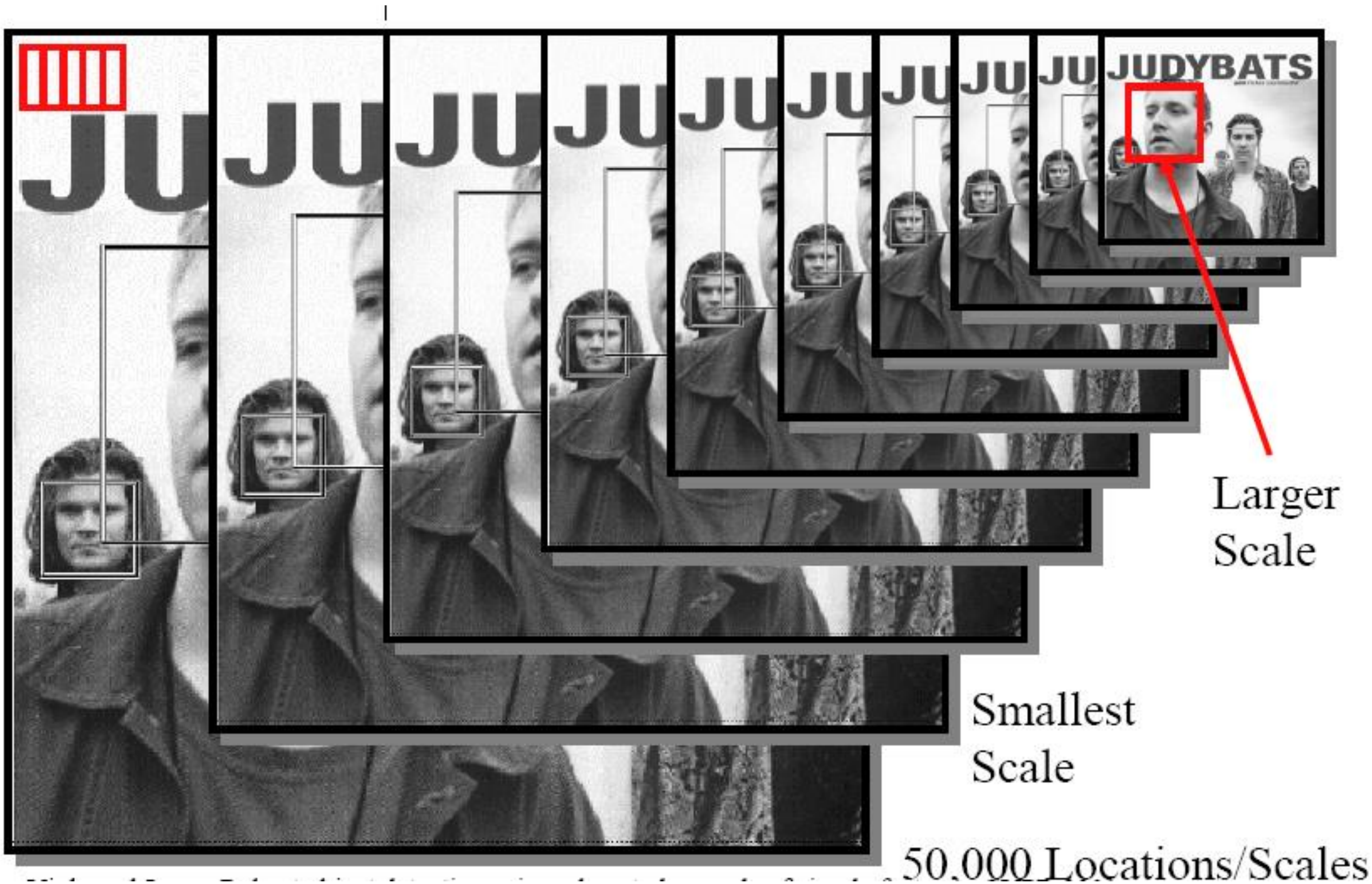


Funny Nikon ads

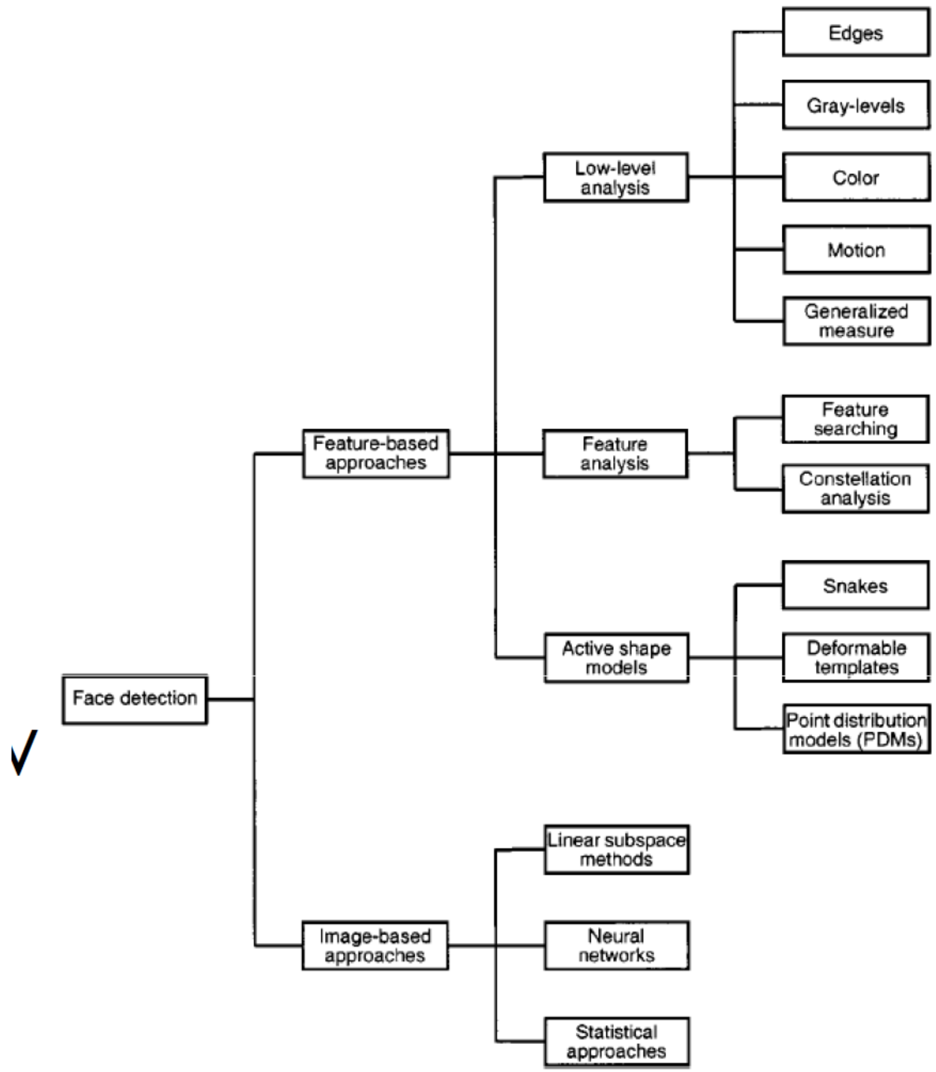
"The Nikon S60 detects up to 12 faces."



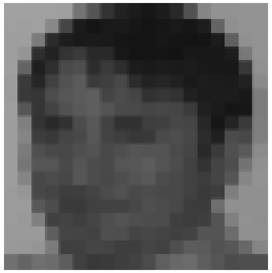
Scan classifier over locs. & scales



Face Detection: A computational perspective



Force Brute Approach



- Consideriamo un'immagine 19x19 pixel come un punto in uno spazio 361-dimensionale F_{raw}
- Ci sono 256^{361} possibili combinazioni di livelli di grigio (punti nello spazio)
  - $256^{361} = 2^{8x361} = 2^{2888}$
- Popolazione mondiale (tutte le facce!)
  - $6{,}400{,}000{,}000 \approx 2^{32}$
- Lo spazio F_{raw} contiene 87 volte tutte le facce del mondo !!!
  - F_{raw} **sparso** e ad **alta dimensionalità**

Parameters

- Posa (frontale, di profilo, intermedia)
- Orientazione
- Elementi (variabili) strutturali
  - barba, baffi
  - occhiali
  - trucco
  - …
- Espressione facciale
- Occlusioni
- Condizioni di acquisizione
  - illuminazione
  - risoluzione
  - caratteristiche della fotocamere

- Età
- Sesso
- Razza
- Apparenza (?)
- …

Approcci

- Knowledge-based
  - codificano la conoscenza umana di cosa costituisce una faccia tipica (di solito, si considerano le relazioni intercorrenti tra feature facciali)
- Feature invariant
  - si basano su feature strutturali invarianti a più fattori possibili
- Template matching
  - i template pattern di feature precalcolati che codificano feature caratterizzanti una faccia con cui ispezionare l'immagine
- Appearence-based
  - simile al template matching, ma qui i template sono esclusivamente avatar di facce

Challenges of face detection

- Sliding window detector must evaluate tens of thousands of location/scale combinations
- Faces are rare: 0–10 per image
  - For computational efficiency, we should try to spend as little time as possible on the non-face windows
  - A megapixel image has ~10^6 pixels and a comparable number of candidate face locations
  - To avoid having a false positive in every image, our false positive rate has to be less than 10^{-6}

The Viola/Jones Face Detector

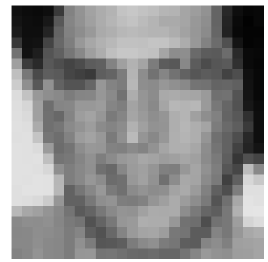
- A seminal approach to real-time object detection
- Training is slow, but detection is very fast
- Key ideas
  - *Integral images* for fast feature evaluation
  - *Boosting* for feature selection
  - *Attentional cascade* for fast rejection of non-face windows

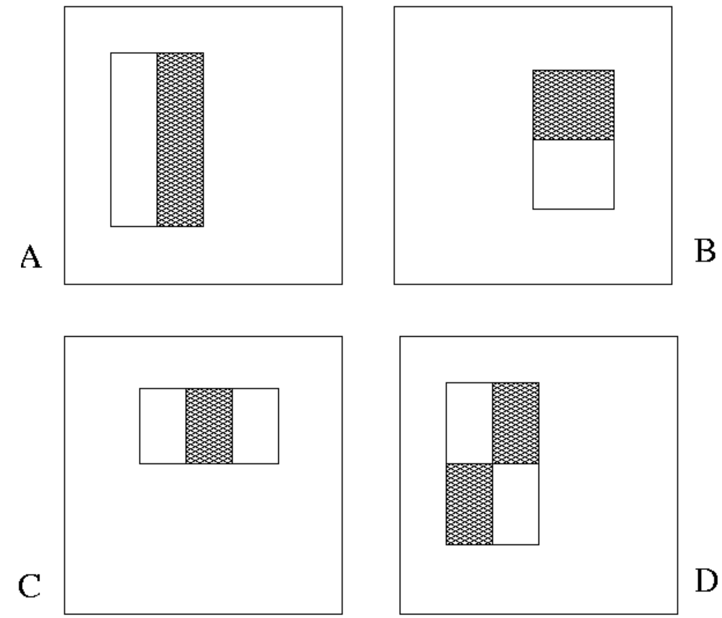
P. Viola and M. Jones. *Rapid object detection using a boosted cascade of simple features.* CVPR 2001.

P. Viola and M. Jones. *Robust real-time face detection.* IJCV 57(2), 2004.

Image Features

"Rectangle filters"

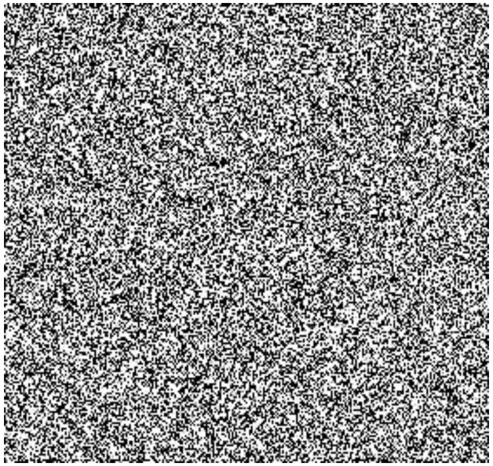




Value =

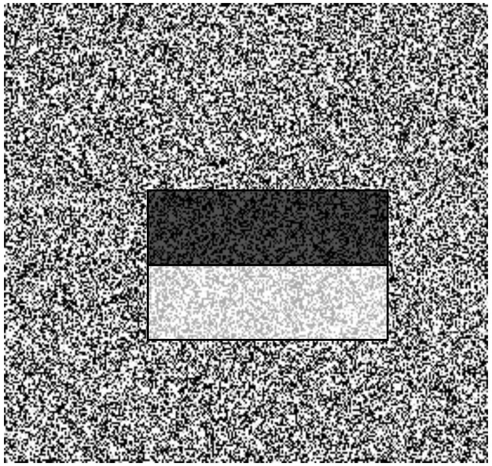
∑ (pixels in white area) –
∑ (pixels in black area)

Example



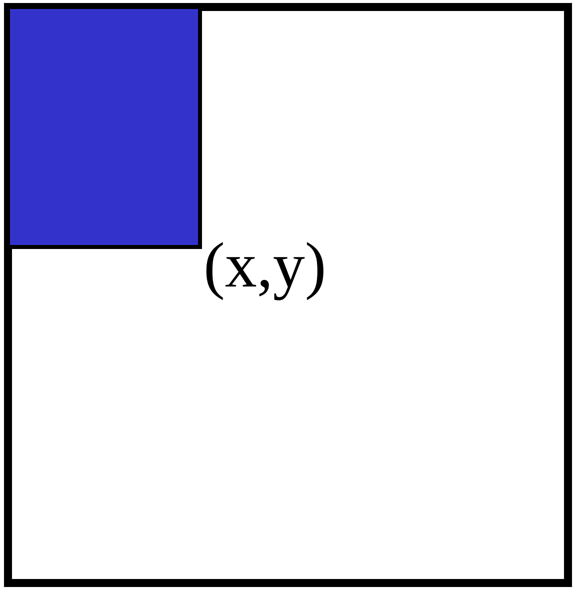
Source

Result

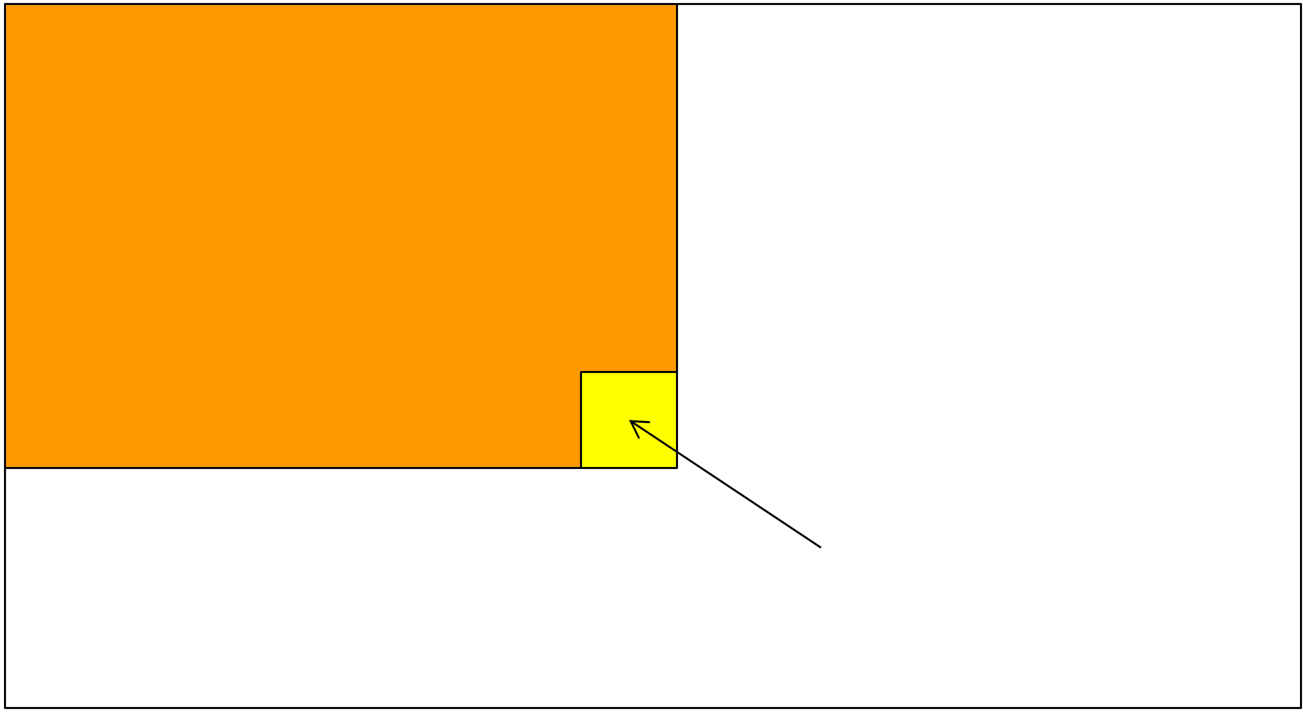


Fast computation with integral images

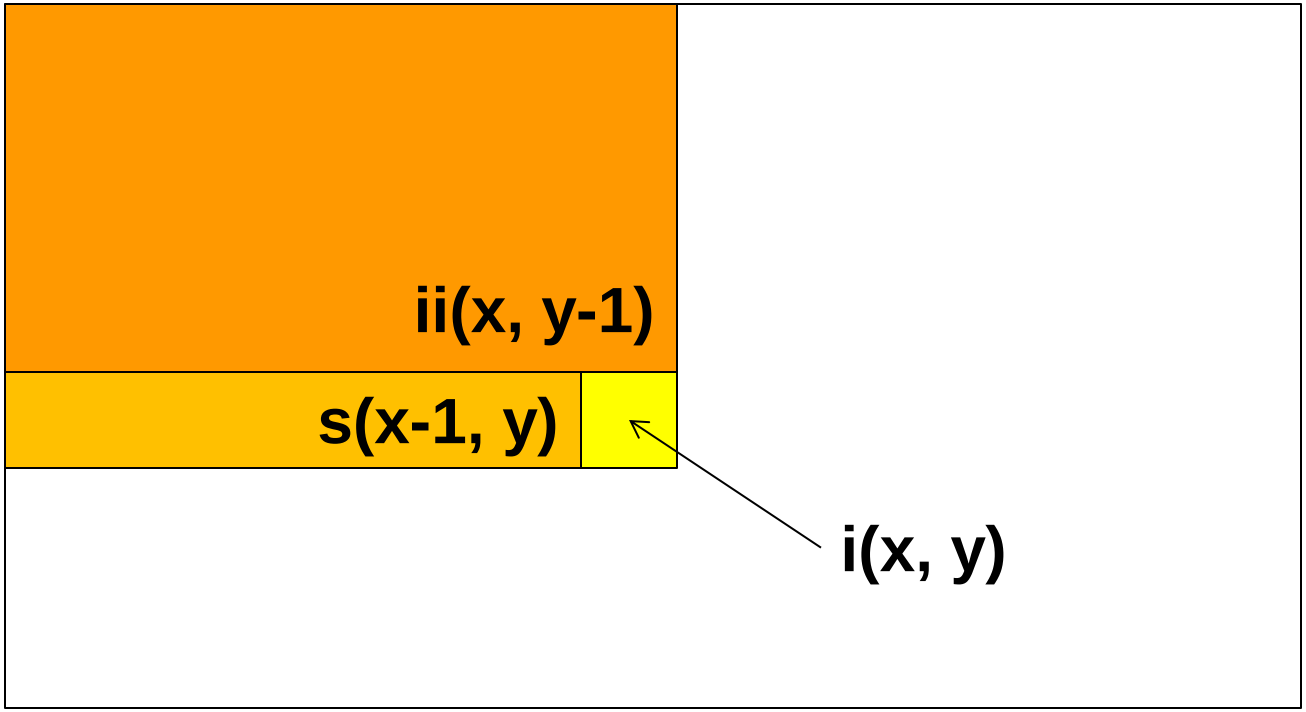
- The *integral image* computes a value at each pixel (*x*,*y*) that is the sum of the pixel values above and to the left of (*x*,*y*), inclusive
- This can quickly be computed in one pass through the image



Computing the integral image



Computing the integral image



Cumulative row sum: $s(x, y) = s(x–1, y) + i(x, y)$

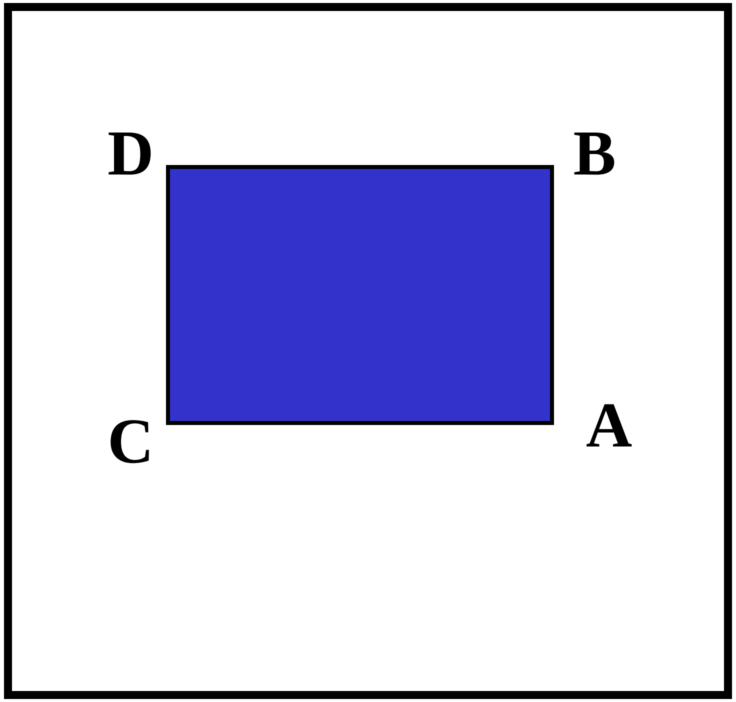
Integral image: $ii(x, y) = ii(x, y−1) + s(x, y)$

MATLAB: ii = cumsum(cumsum(double(i)), 2);

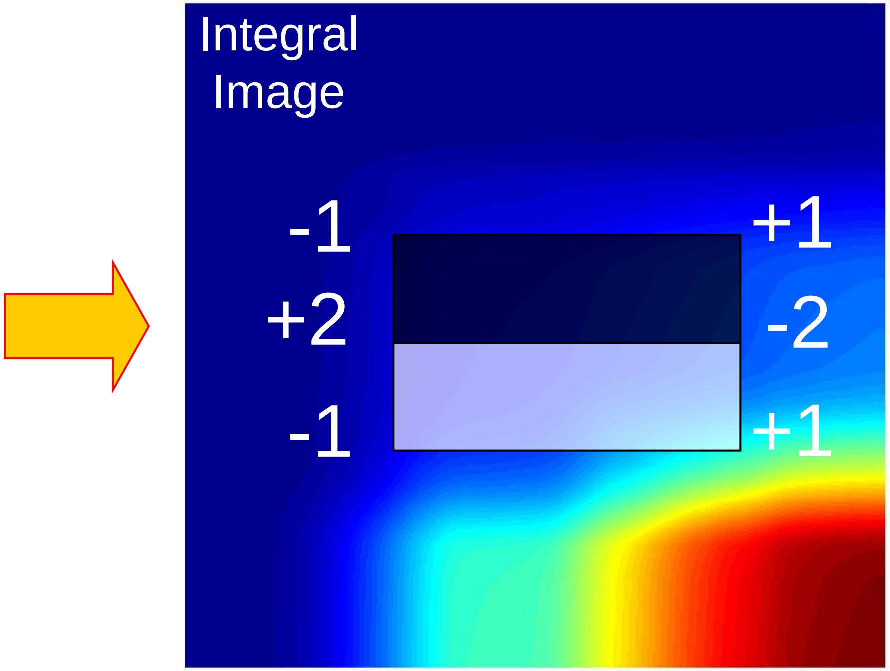
Computing sum within a rectangle

- Let A,B,C,D be the values of the integral image at the corners of a rectangle
- Then the sum of original image values within the rectangle can be computed as:

  sum = A – B – C + D
- Only 3 additions are required for any size of rectangle!

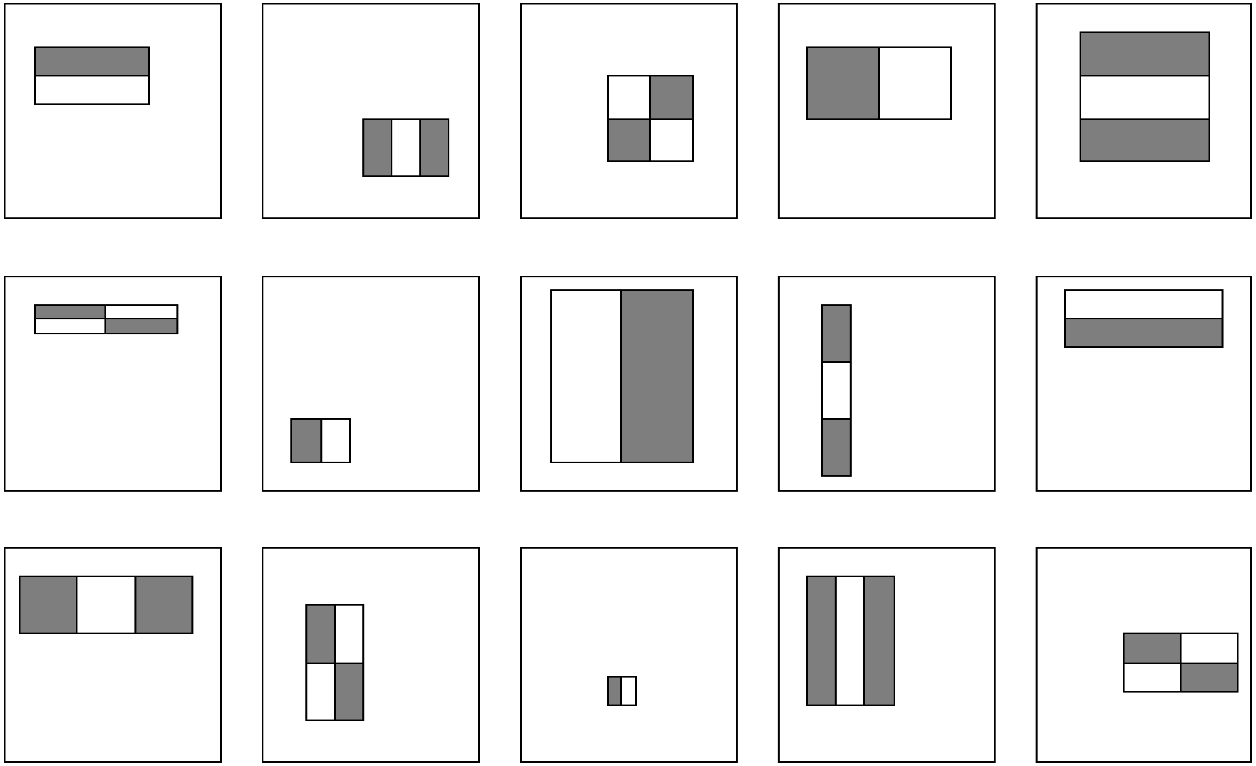


Example



Feature selection

- For a 24x24 detection region, the number of possible rectangle features is ~160,000!



Feature selection

- For a 24x24 detection region, the number of possible rectangle features is ~160,000!
- At test time, it is impractical to evaluate the entire feature set
- Can we create a good classifier using just a small subset of all possible features?
- How to select such a subset?

Boosting

- Boosting is a classification scheme that combines *weak learners* into a more accurate *ensemble classifier*
- Training procedure
  - Initially, weight each training example equally
  - In each **boosting** round:
    - Find the weak learner that achieves the lowest *weighted* training error
    - Raise the weights of training examples misclassified by current weak learner
  - Compute final classifier as linear combination of all weak learners (weight of each learner is directly proportional to its accuracy)
    - Exact formulas for re-weighting and combining weak learners depend on the particular boosting scheme (e.g., AdaBoost)

Y. Freund and R. Schapire, A short introduction to boosting, *Journal of Japanese Society for Artificial Intelligence*, 14(5):771-780, September, 1999.

Boosting for face detection

- Define weak learners based on rectangle features

value of rectangle feature

$$h_t(x) = \begin{cases} 1 & \text{if } p_t f_t(x) > p_t \theta_t \\ 0 & \text{otherwise} \end{cases}$$

window

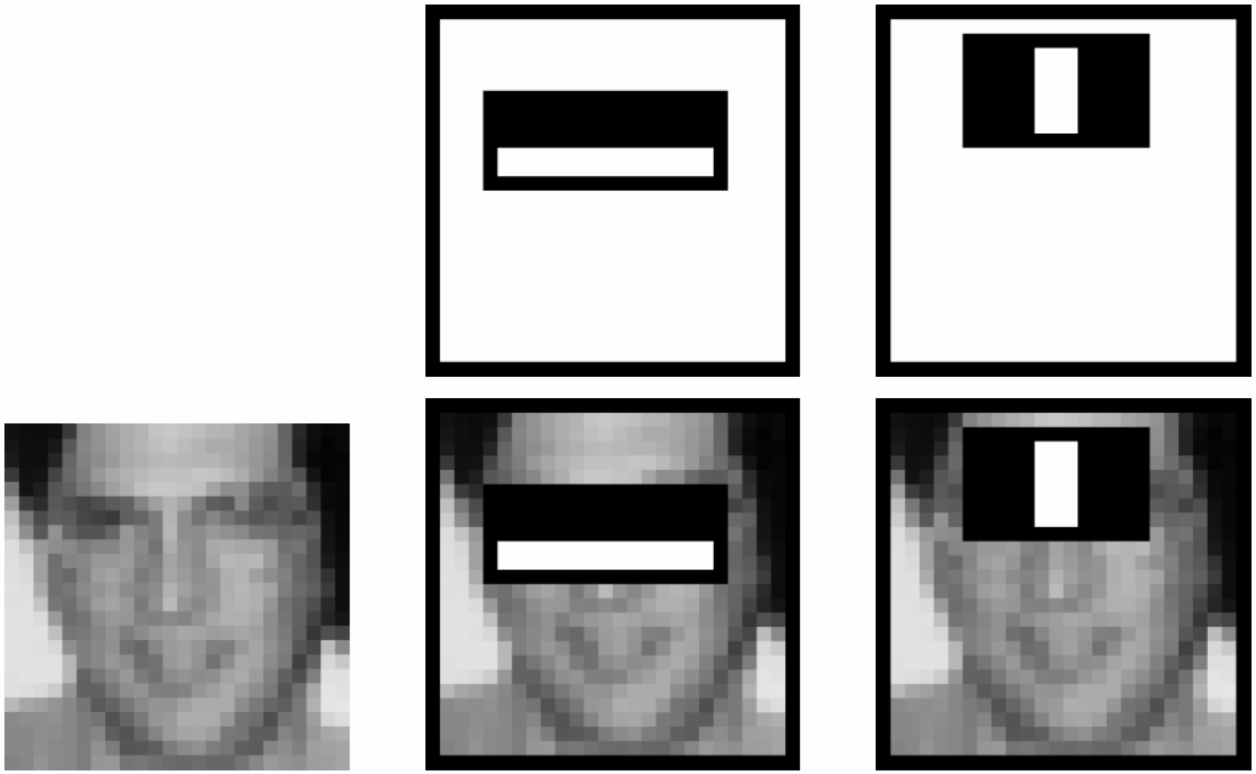
parity

threshold

- For each round of boosting:
  - Evaluate each rectangle filter on each example
  - Select best filter/threshold combination based on weighted training error
  - Reweight examples

Boosting for face detection

- First two features selected by boosting:



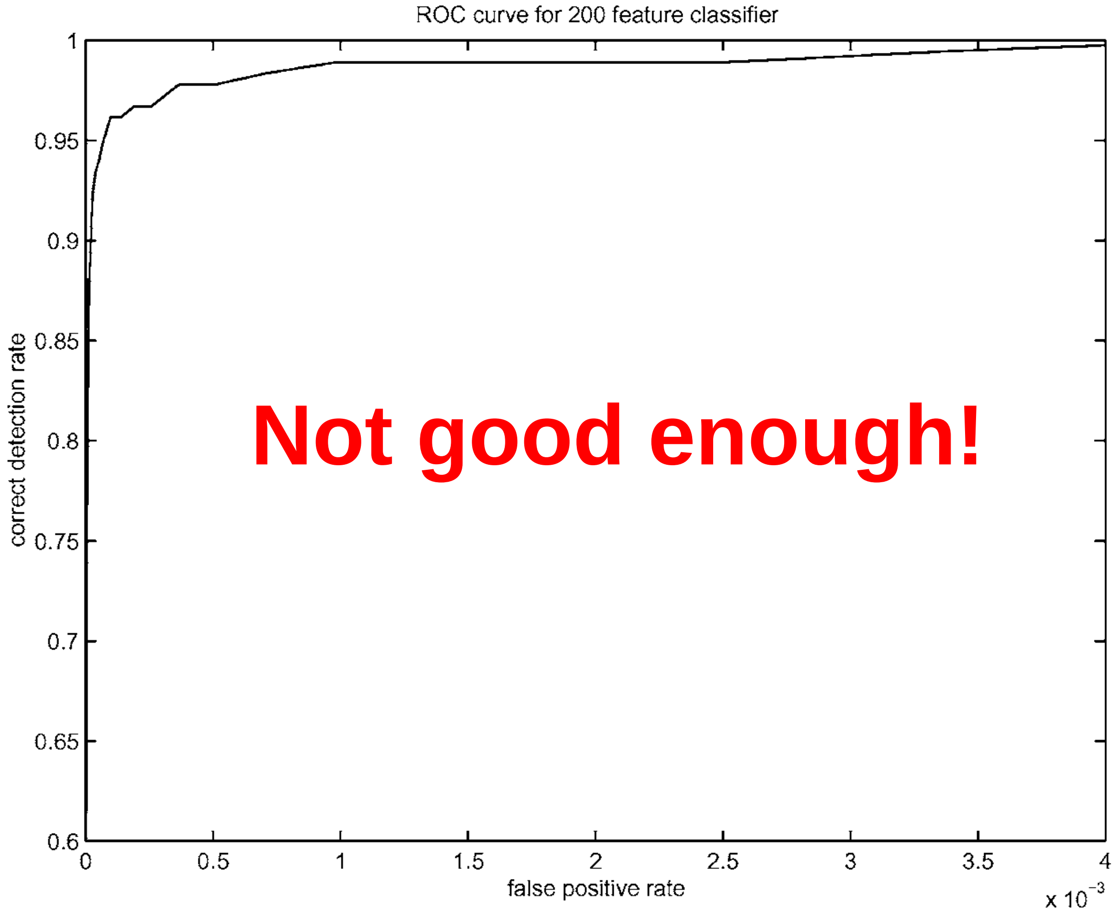
This feature combination can yield 100% detection rate and 50% false positive rate

Boosting vs. SVM

- Advantages of boosting
  - Integrates classifier training with feature selection
  - Complexity of training is linear instead of quadratic in the number of training examples
  - Flexibility in the choice of weak learners, boosting scheme
  - Testing is fast
  - Easy to implement
- Disadvantages
  - Needs many training examples
  - Training is slow
  - Often doesn't work as well as SVM (especially for many-class problems)

Boosting for face detection

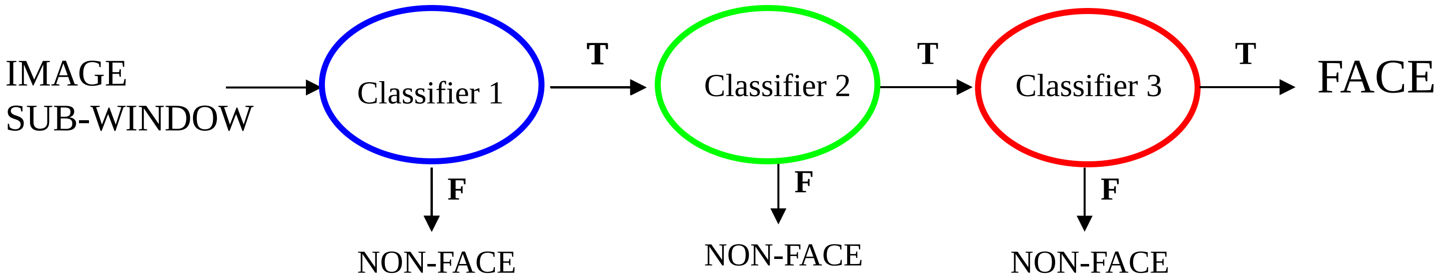
- A 200-feature classifier can yield 95% detection rate and a false positive rate of 1 in 14084



Receiver operating characteristic (ROC) curve

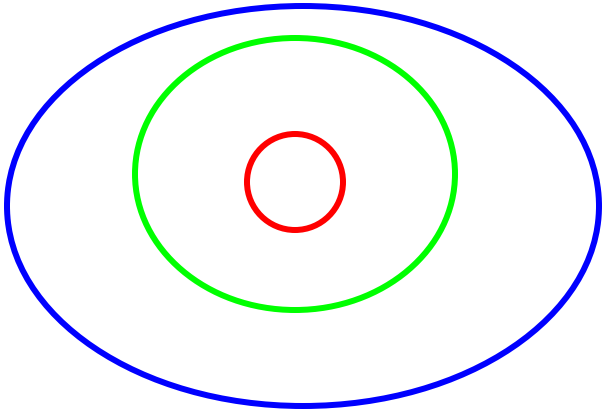
Attentional cascade

- We start with simple classifiers which reject many of the negative sub-windows while detecting almost all positive sub-windows
- Positive response 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

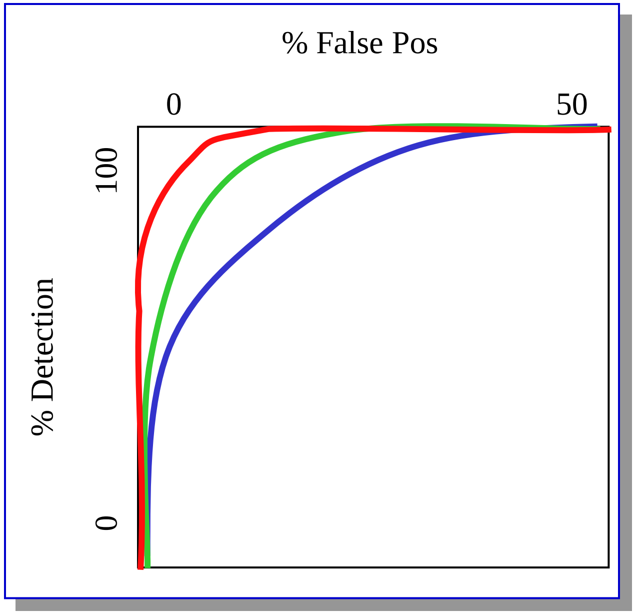


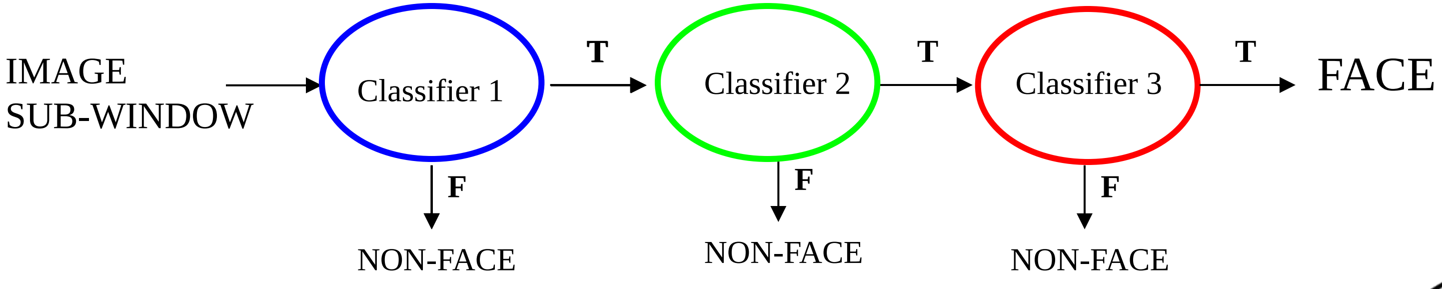
Attentional cascade

- Chain classifiers that are progressively more complex and have lower false positive rates:



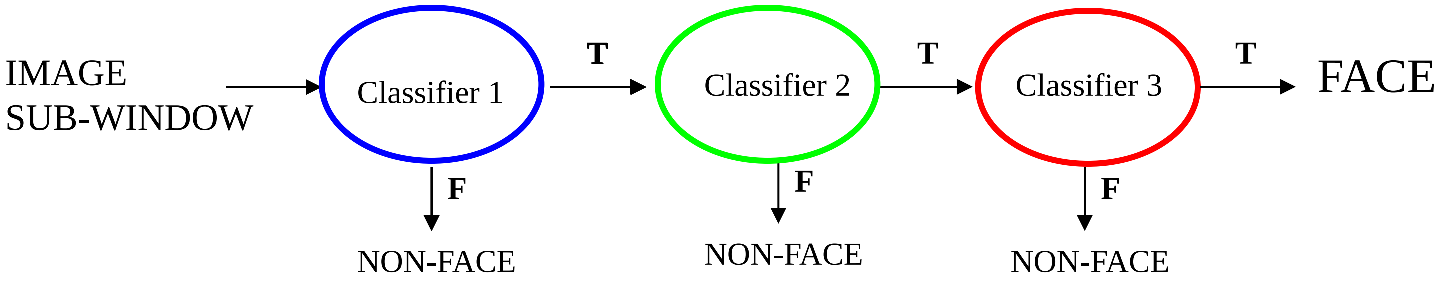
Receiver operating characteristic



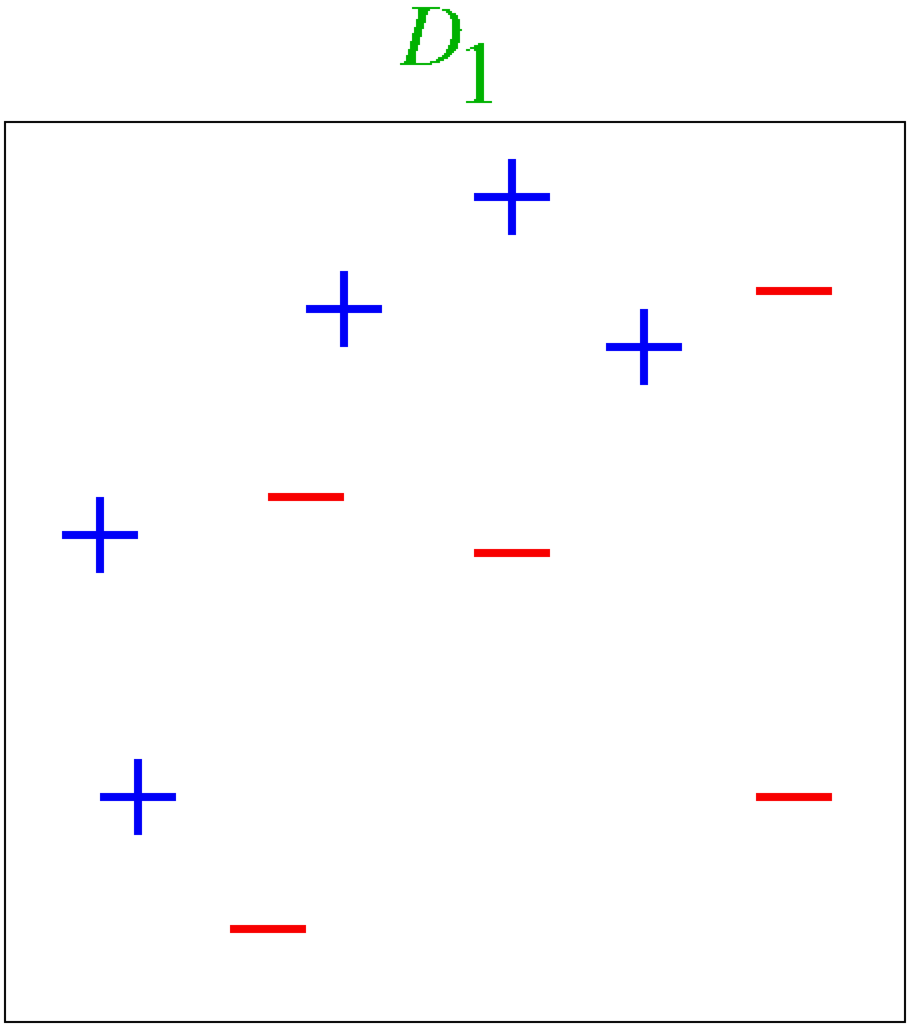


Attentional cascade

- The detection rate and the false positive rate of the cascade are found by multiplying the respective rates of the individual stages
- A detection rate of 0.9 and a false positive rate on the order of 10^{-6} can be achieved by a 10-stage cascade if each stage has a detection rate of 0.99 ($0.99^{10} \approx 0.9$) and a false positive rate of about 0.30 ($0.3^{10} \approx 6\times10^{-6}$)



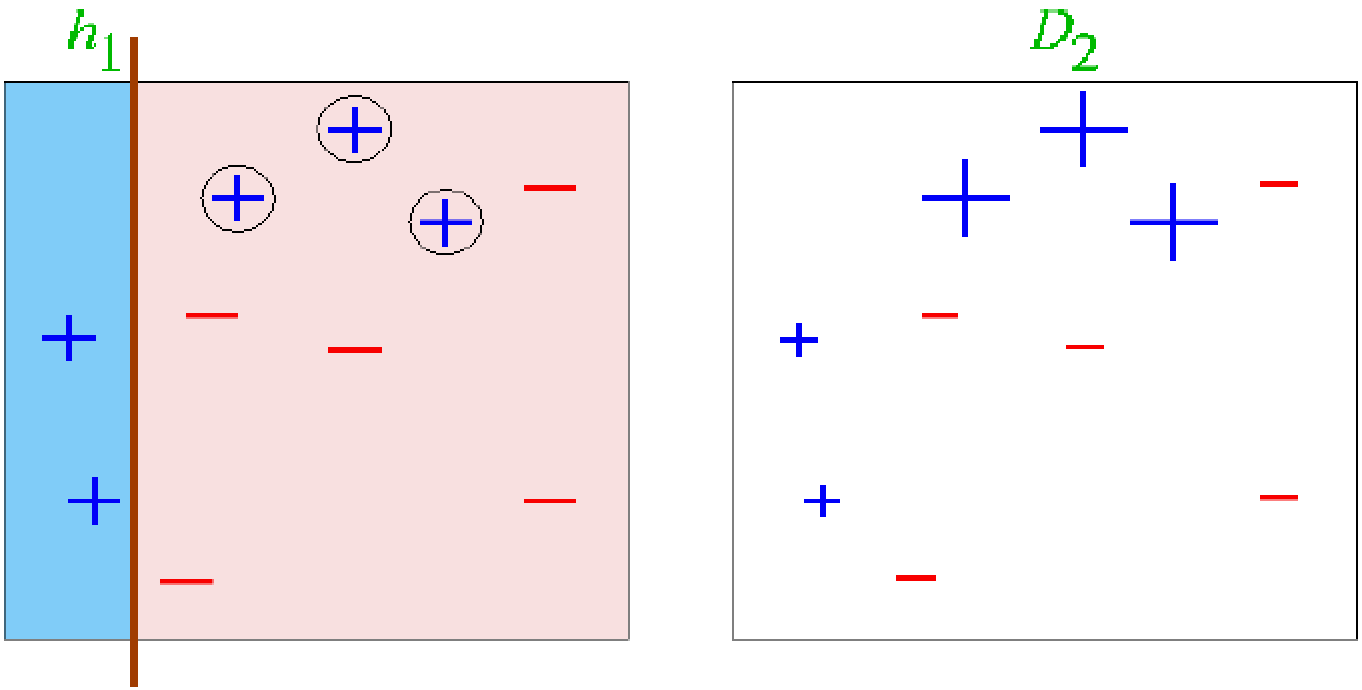
Un esempio



Il training set di partenza e composto da due popolazioni con probabilita a priori simili tra loro.

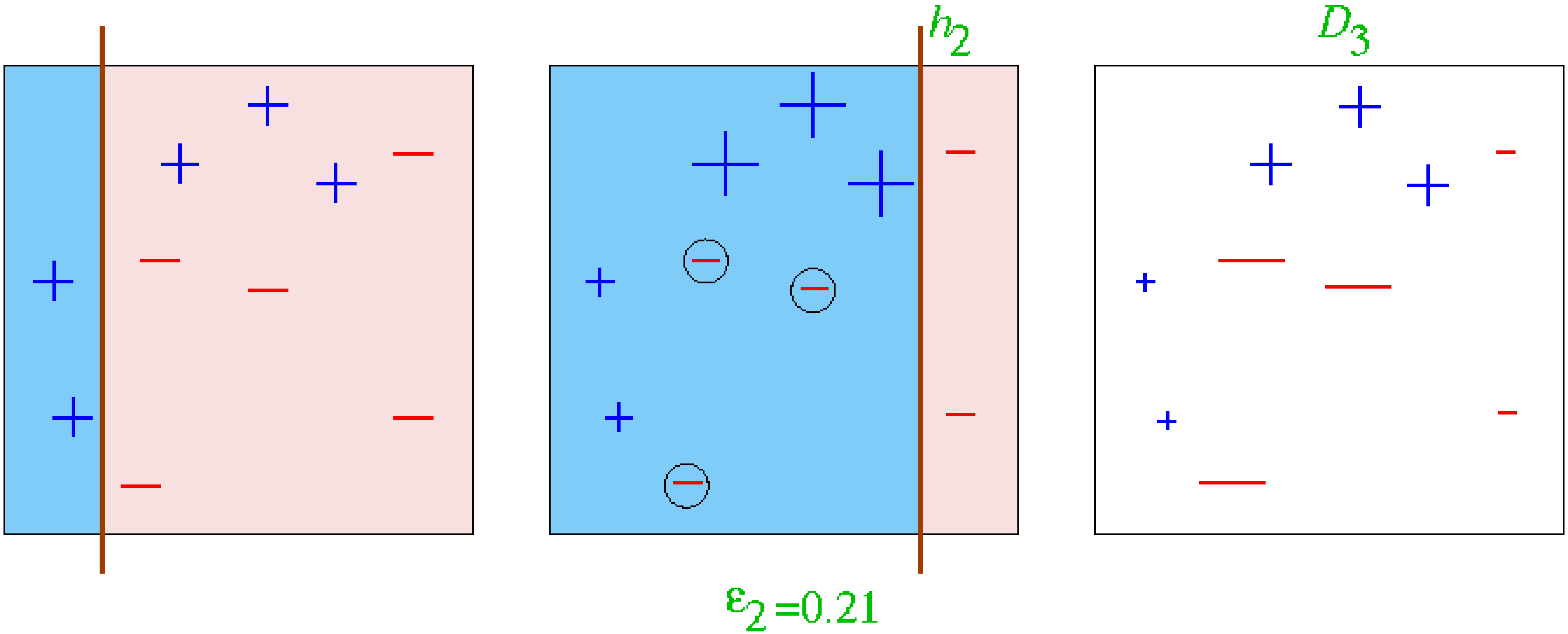
La “frontiera” tra le due popolazioni e ben definita ma non appare essere “lineare”.

First Result: una sola feature



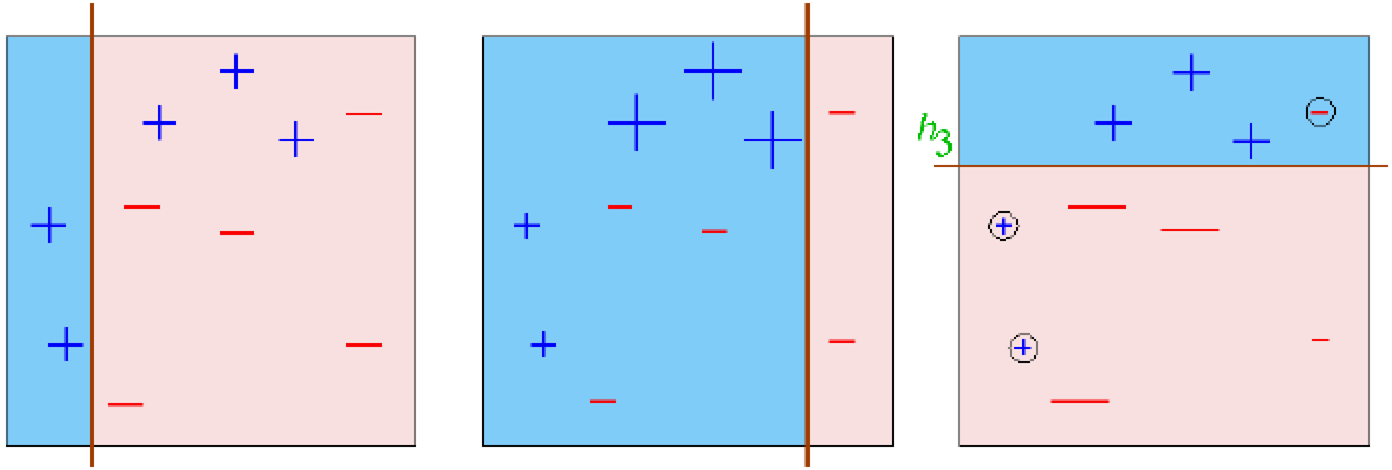
Un primo tentativo di classificatore cerca di ottimizzare il riconoscimento dei “-” basandosi su una sola feature.

Second classifier



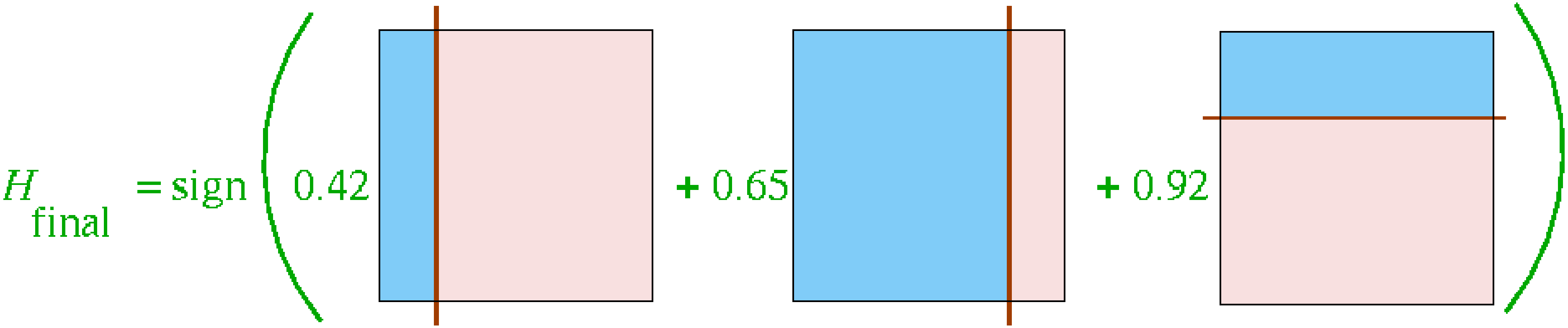
Un nuovo classificatore esamina il training set aggiornato, sempre su una unica feature. Questa volta cerca di ottimizzare sui "+".

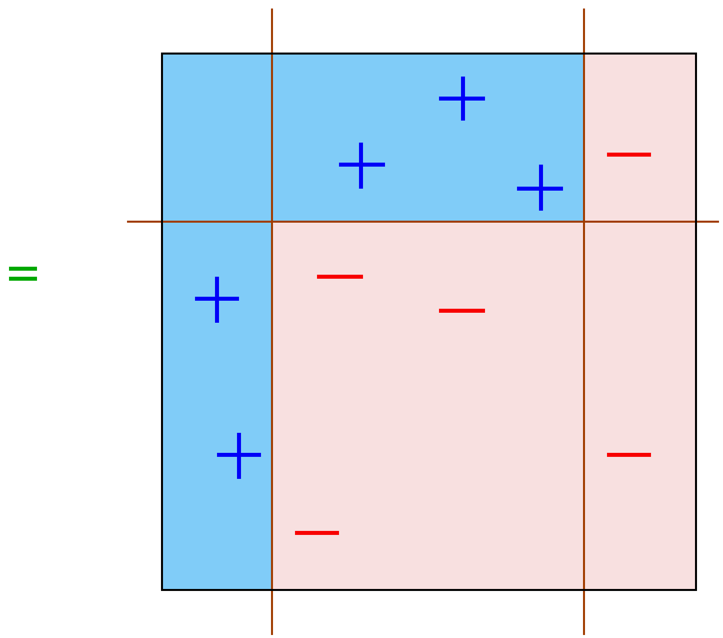
Third classifier



I pesi dei campioni nel Training Set sono stati ormai molto cambiati. Il terzo classificatore opera come soglia sulla seconda feature.

Final Classifier





Il classificatore finale risultante ha costruito una frontiera lineare a tratti che si adatta perfettamente al Training Set.

Training the cascade

- Set target detection and false positive rates for each stage
- Keep adding features to the current stage until its target rates have been met
  - Need to lower AdaBoost threshold to maximize detection (as opposed to minimizing total classification error)
  - Test on a *validation set*
- If the overall false positive rate is not low enough, then add another stage
- Use false positives from current stage as the negative training examples for the next stage

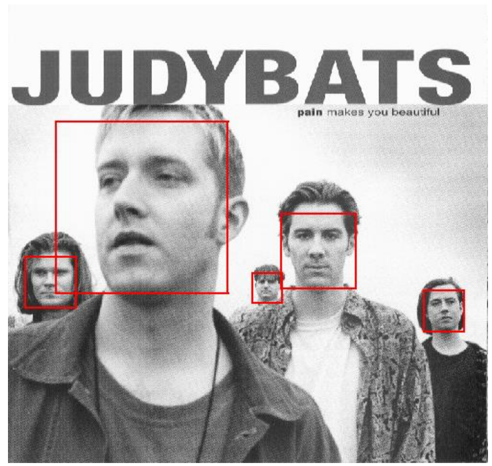
The implemented system

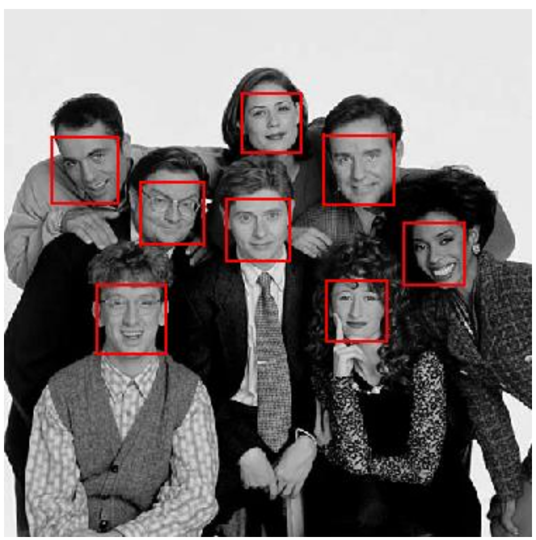
- Training Data
  - 5000 faces
    - All frontal, rescaled to 24x24 pixels
  - 300 million non-faces
    - 9500 non-face images
  - Faces are normalized
    - Scale, translation
- Many variations
  - Across individuals
  - Illumination
  - Pose

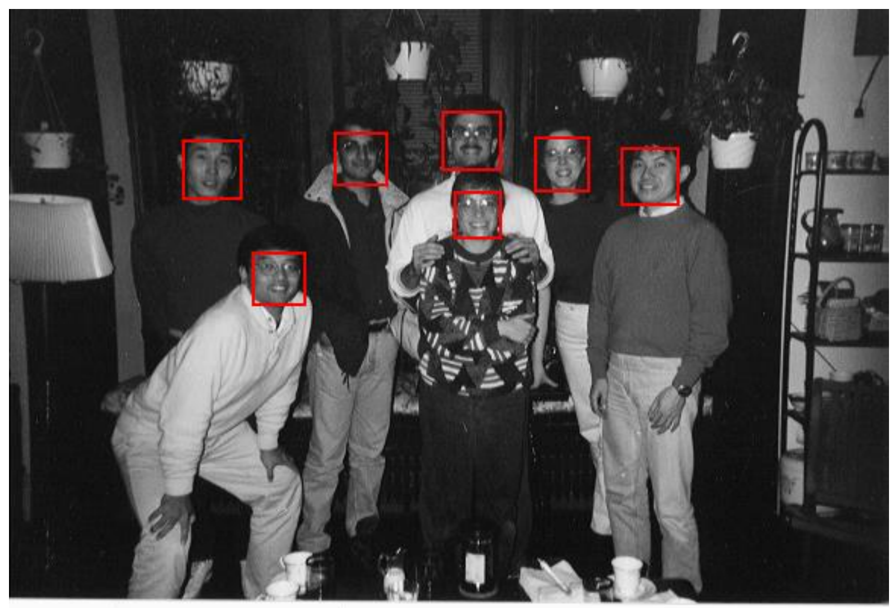
System performance

- Training time: “weeks” on 466 MHz Sun workstation
- 38 layers, total of 6061 features
- Average of 10 features evaluated per window on test set
- “On a 700 Mhz Pentium III processor, the face detector can process a 384 by 288 pixel image in about .067 seconds”
  - 15 Hz
  - 15 times faster than previous detector of comparable accuracy (Rowley et al., 1998)

Output of Face Detector on Test Images

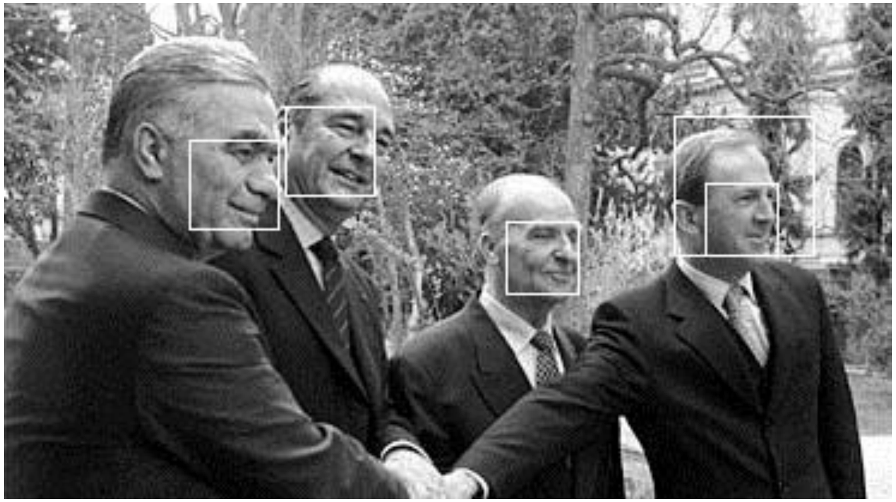






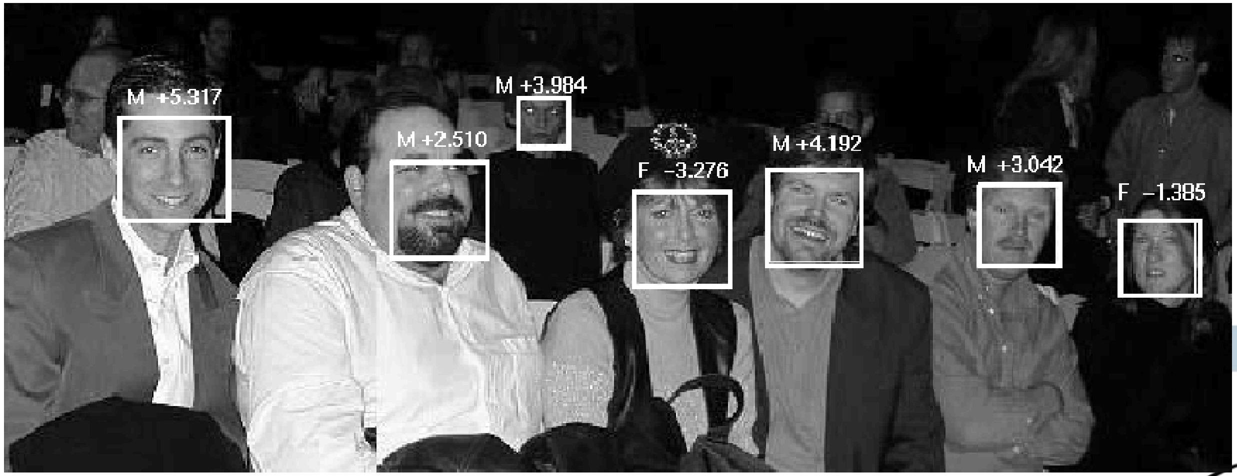
Other detection tasks

Facial Feature Localization

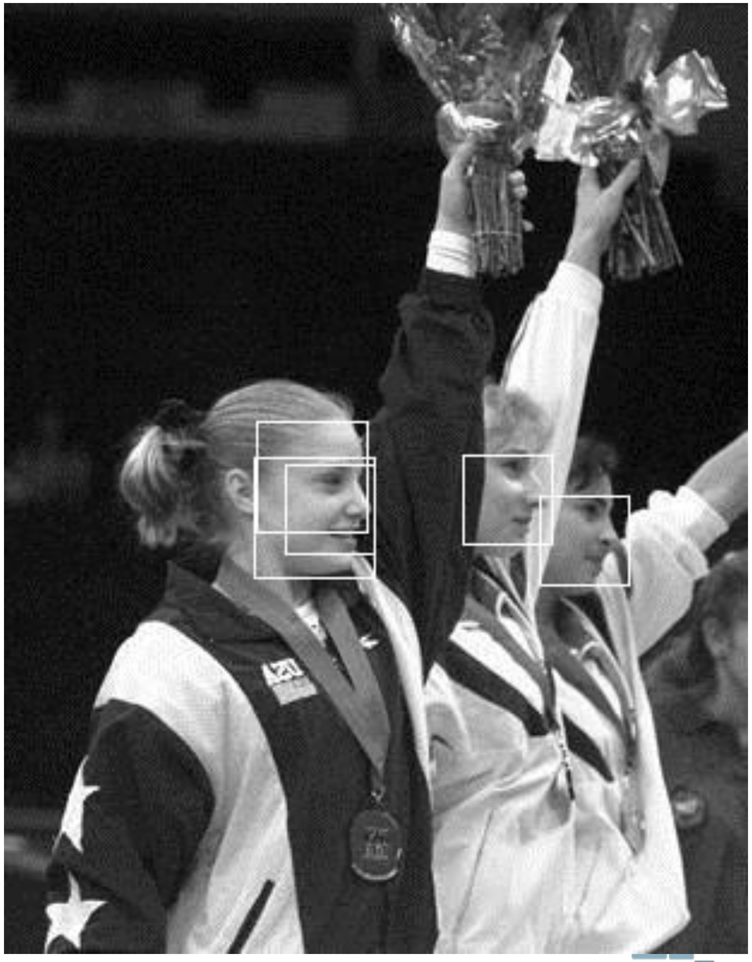


Profile Detection

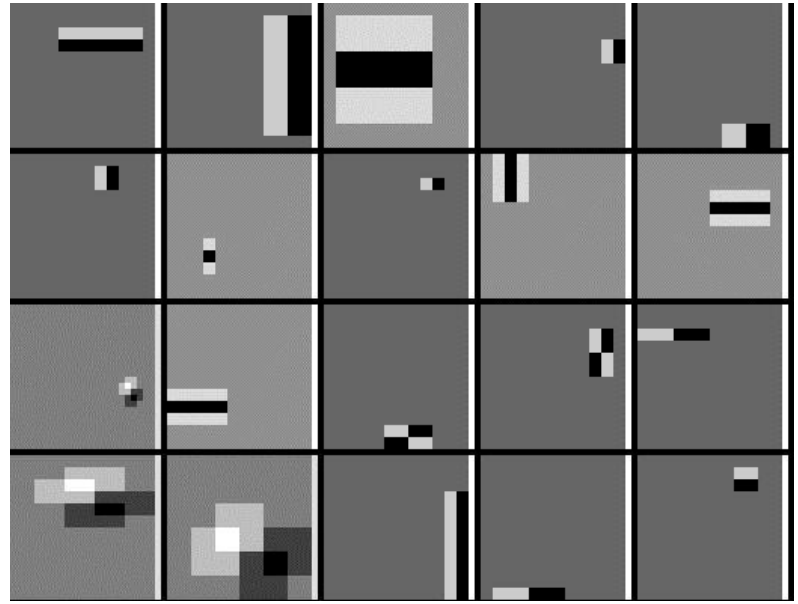
Male vs. female



Profile Detection



Profile Features



Face Detection: OpenCV

- La libreria OpenCV include una implementazione dell'algoritmo di Viola-Jones
- Un buon punto di partenza: http://note.sonots.com/SciSoftware/haartraining.html
- 3 utili tool
  - "createsamples"
  - "haartraining"
  - "performance"

- createsamples
  - Tool di OpenCV per la creazione automaticadi esempi di addestramento
  - 4 funzionalità
    - crea esempi di addestramento da una singola immagine applicando delle deformazioni
    - crea esempi di addestramento da un insieme di immagini senza introdurre deformazioni
    - crea esempi di addestramento con riferimento assoluto (ground truth) da una singola immagine applicando deformazioni
    - Visualizza immagini del formato interno .vec che contiene collezioni di immagini
  - L'utilizzo migliore è di usare una combinazione delle funzionalità per creare molte immagini con deformazioni e creare un unico insieme

Summary: Viola/Jones detector

- Rectangle features
- Integral images for fast computation
- Boosting for feature selection
- Attentional cascade for fast rejection of negative windows

What is Face Recognition?

A set of two tasks:

- Face Identification: Given a face image that belongs to a person in a database, tell whose image it is.
- Face Verification: Given a face image that might not belong to the database, verify whether it is from the person it is claimed to be in the database.

Difference between Face Detection and Recognition

Detection – two-class classification

- Face vs. Non-face

Recognition – multi-class classification

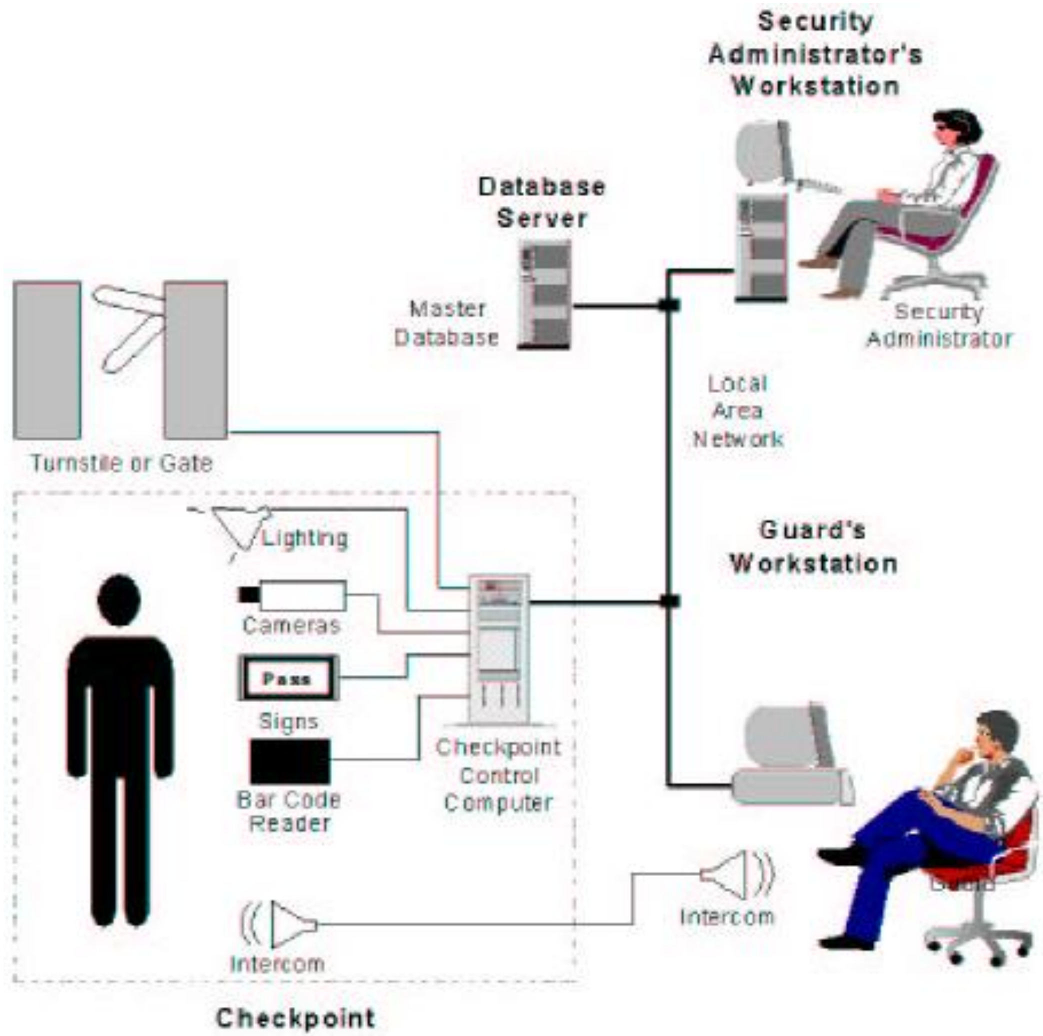
- One person vs. all the others

Applications of Face Recognition

- Access Control
- Face Databases
- Face ID
- HCI - Human Computer Interaction
- Law Enforcement

Applications of Face Recognition

- Multimedia Management
- Security
- Smart Cards
- Surveillance
- Others



Different Approaches

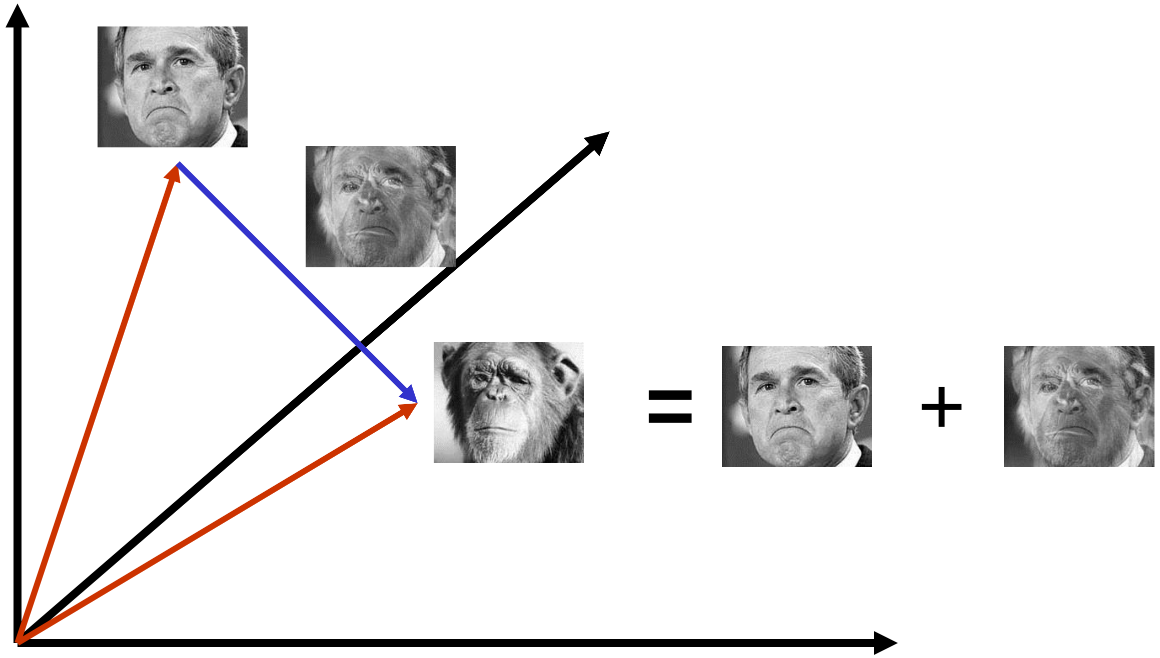
Features:

- Features from global appearance
  - Principal Component Analysis(PCA)
  - Independent Component Analysis(ICA)
- Features from local regions
  - Local Feature Analysis(LFA)
  - Gabor Wavelet

Similarity Measure

- Euclidian Distance
- Neural Networks
- Elastic Graph Matching
- Template Matching
- …

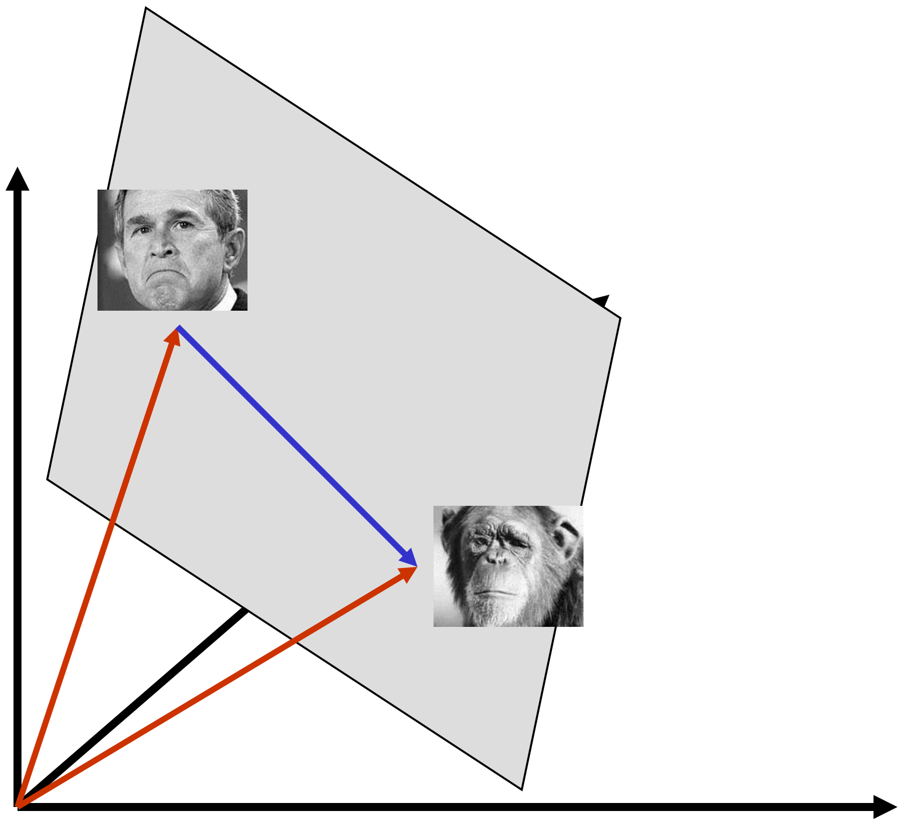
The space of faces



An image is a point in a high dimensional space

- An N x M image is a point in R^{NM}
- We can define vectors in this space as we did in the 2D case

Dimensionality reduction



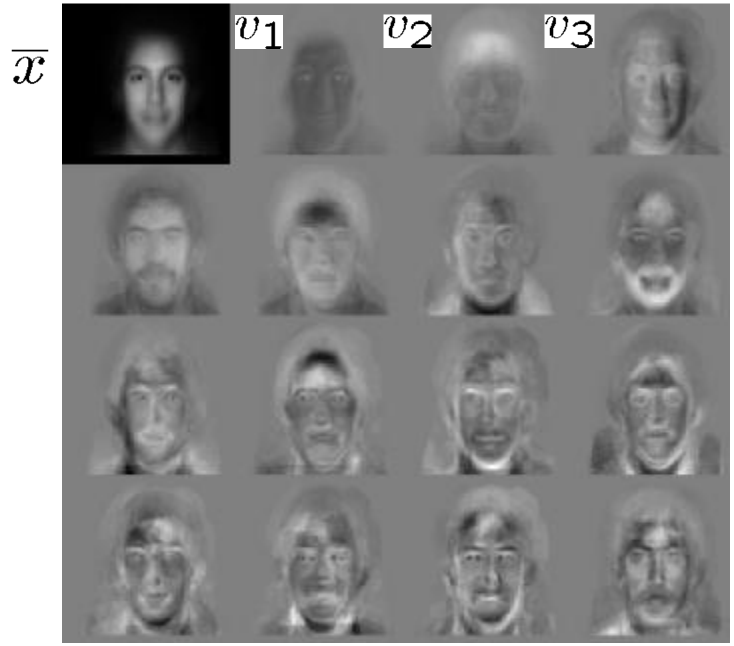
The set of faces is a "subspace" of the set of images

- We can find the best subspace using PCA
- This is like fitting a "hyper-plane" to the set of faces
  - spanned by vectors $\mathbf{v_1}, \mathbf{v_2}, ..., \mathbf{v_K}$
  - any face $\mathbf{x} \approx \overline{\mathbf{x}} + a_1\mathbf{v_1} + a_2\mathbf{v_2} + \ldots + a_k\mathbf{v_k}$

Eigenfaces

PCA extracts the eigenvectors of **A**

- Gives a set of vectors $\mathbf{v_1}$, $\mathbf{v_2}$, $\mathbf{v_3}$, ...
- Each vector is a direction in face space
  - what do these look like?



Projecting onto the eigenfaces

The eigenfaces $\mathbf{v_1}, \ldots, \mathbf{v_K}$ span the space of faces

- A face is converted to eigenface coordinates by

$$\mathbf{x} \rightarrow (\underbrace{(\mathbf{x} - \overline{\mathbf{x}}) \cdot \mathbf{v_1}}_{a_1}, \underbrace{(\mathbf{x} - \overline{\mathbf{x}}) \cdot \mathbf{v_2}}_{a_2}, \ldots, \underbrace{(\mathbf{x} - \overline{\mathbf{x}}) \cdot \mathbf{v_K}}_{a_K})$$

$$\mathbf{x} \approx \overline{\mathbf{x}} + a_1\mathbf{v_1} + a_2\mathbf{v_2} + \ldots + a_K\mathbf{v_K}$$

$\mathbf{x}$

$a_1\mathbf{v_1}$ $a_2\mathbf{v_2}$ $a_3\mathbf{v_3}$ $a_4\mathbf{v_4}$ $a_5\mathbf{v_5}$ $a_6\mathbf{v_6}$ $a_7\mathbf{v_7}$ $a_8\mathbf{v_8}$

Recognition with eigenfaces

Algorithm

1. Process the image database (set of images with labels)
   - Run PCA—compute eigenfaces
   - Calculate the K coefficients for each image
2. Given a new image (to be recognized) **x**, calculate K coefficients

$$\mathbf{x} \rightarrow (a_1, a_2, \ldots, a_K)$$

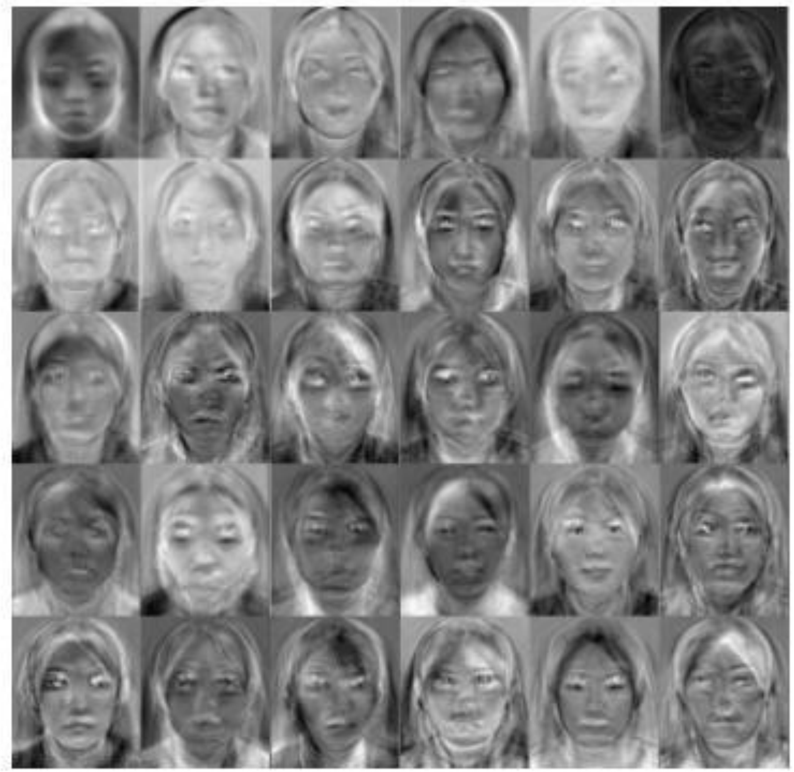
3. Detect if x is a face

$$\|\mathbf{x} - (\overline{\mathbf{x}} + a_1\mathbf{v_1} + a_2\mathbf{v_2} + \ldots + a_K\mathbf{v_K})\| < \text{threshold}$$

4. If it is a face, who is it?
   – Find closest labeled face in database
      » nearest-neighbor in K-dimensional space

The PCA Approach - Eigenface

Eigenfaces – an example



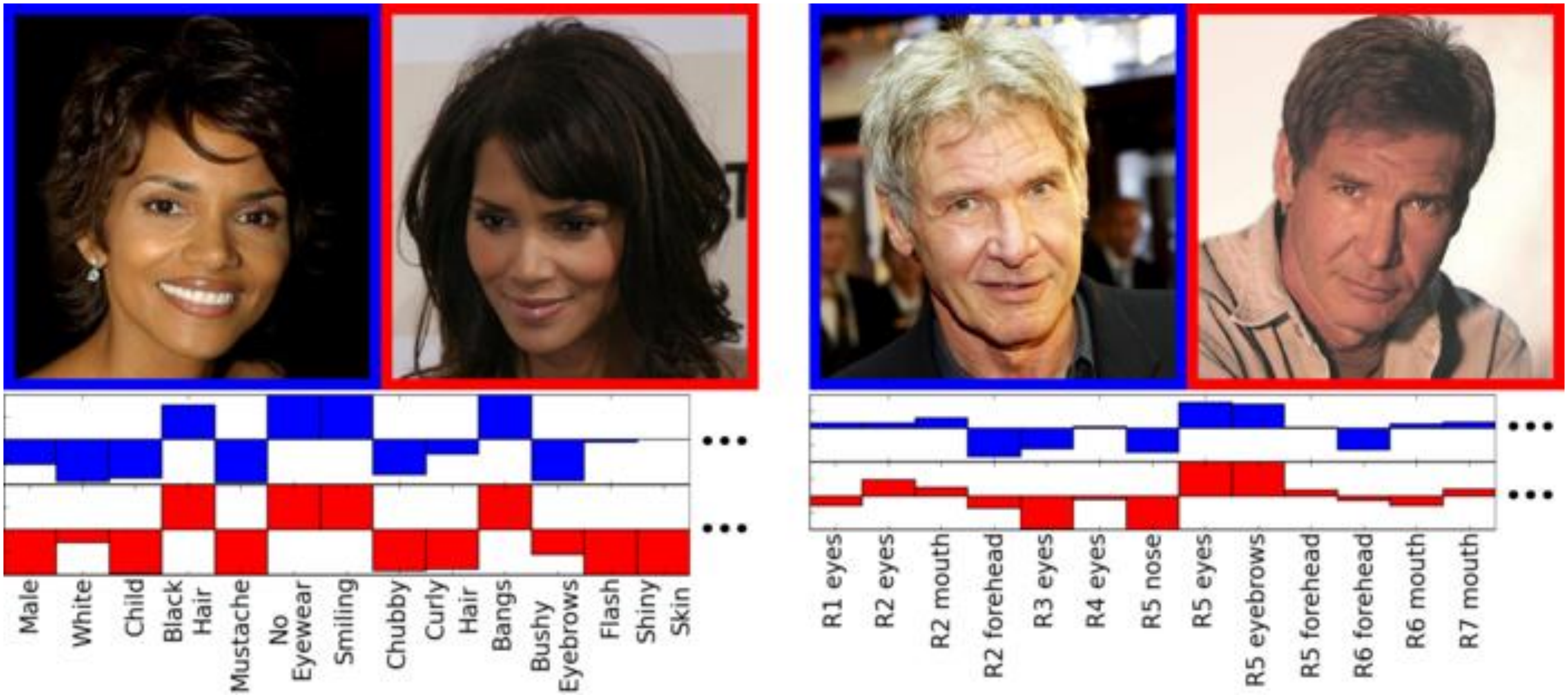
Face Detection + Recognition

Detection accuracy affects the recognition stage

Key issues:

- Correct location of key facial features(e.g. the eye corners)
- False detection
- Missed detection

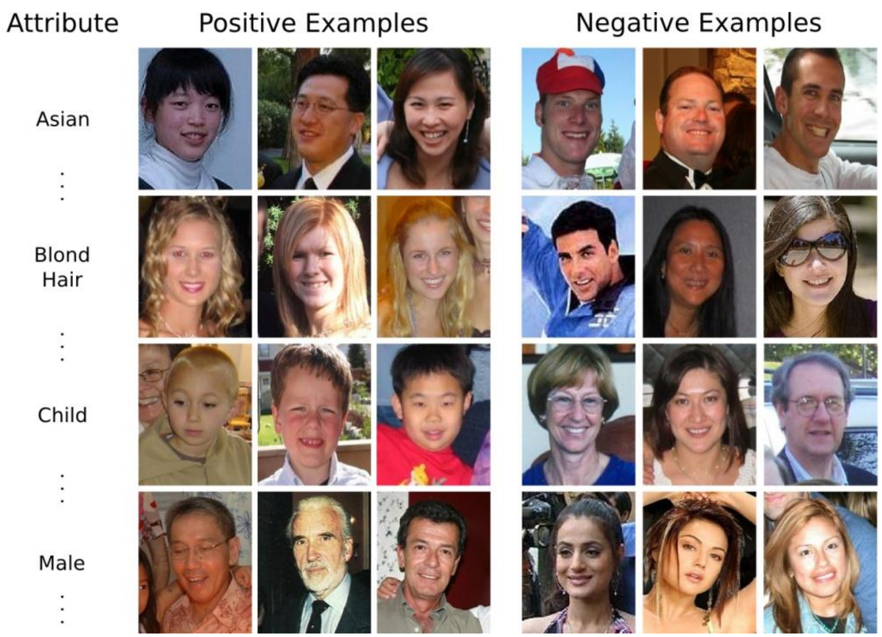
Face Recognition



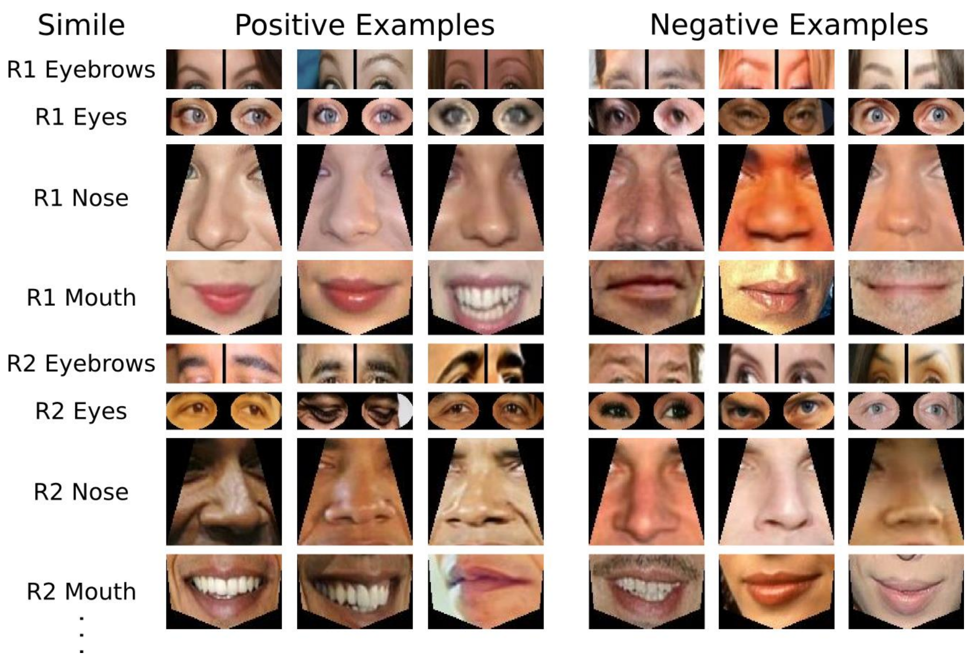
N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, **"Attribute and Simile Classifiers for Face Verification,"** ICCV 2009.

Face Recognition

Attributes for training



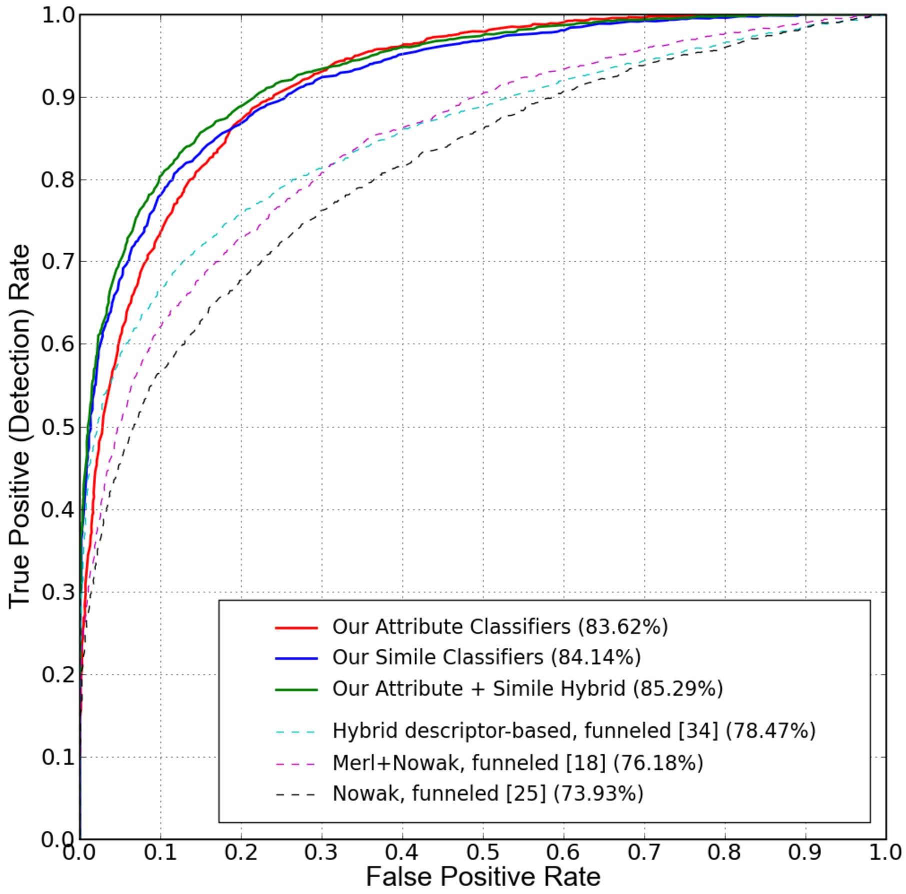
Similes for training



N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, **"Attribute and Simile Classifiers for Face Verification,"** ICCV 2009.

Face Recognition

Results on Labeled Faces in the Wild Dataset



N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, **"Attribute and Simile Classifiers for Face Verification,"** ICCV 2009.

BUT….

We have to remember that biological systems are able to deal with extreme variations of signals and still extract right information from them. This will be illustrated now by the example of face recognition

Faces can be distorted in many ways and still recognized. We can guess something about PRINCIPLES OF FACE PROCESSING

Face detection: State-of-the-art

(Courtesy Boris Babenko)
http://vision.ucsd.edu/~bbabenko/

TLD simultaneously Tracks the object, Learns its appearance and Detects it whenever it appears in the video.

http://info.ee.surrey.ac.uk/Personal/Z.Kalal/tld.html