“Hidden Face by Coffee Beans”

Class 8

Face Detection

Index

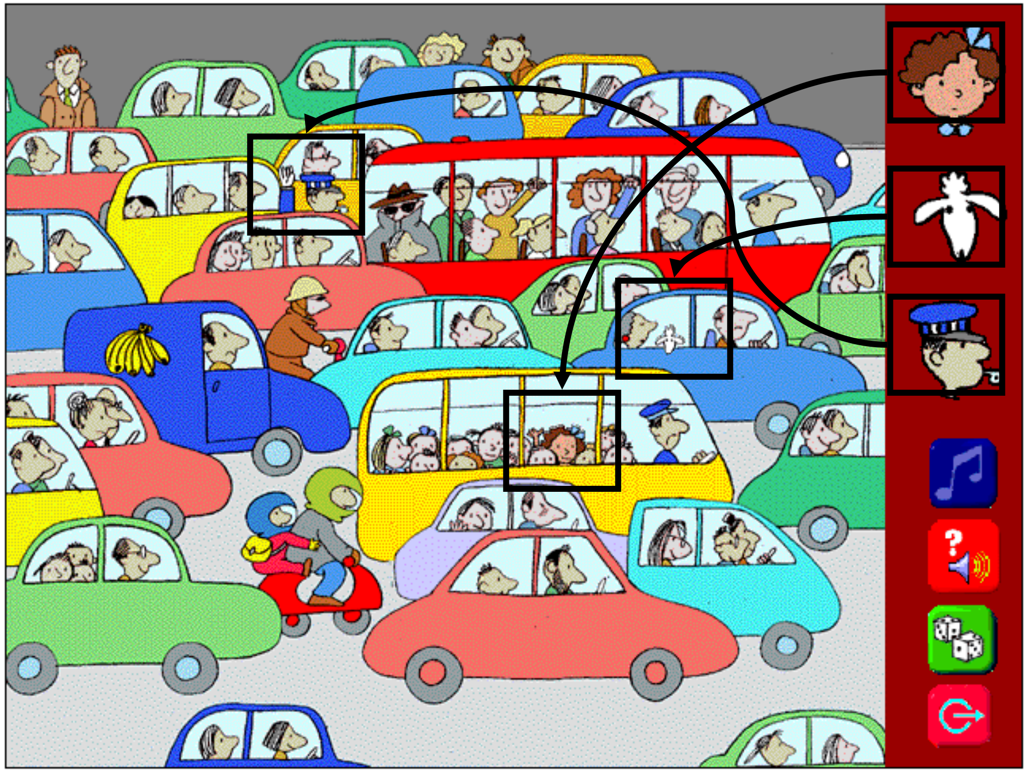
1) The problem of detection and object recognition

   1) Stimulus equivalence
   2) Classification vs Identification

2) Feature-based methods for face detection

   1. Adaboost
   2. Cascade of classifiers

What is the problem of object detection and recognition?

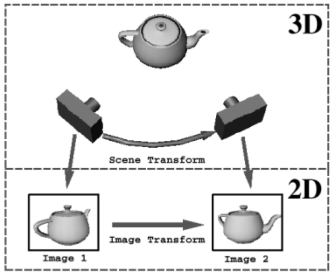


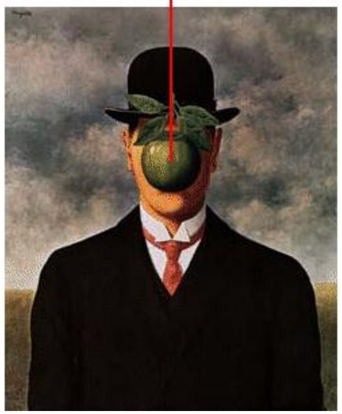
What's the difficulty of object detection and recognition?

Problem: stimulus equivalence – an infinite number of views correspond to the same object.

1 object↔ ∞ images

Reasons: scale, orientation, etc.





We need invariant features!

What's the problem of object detection and recognition?

Problem: stimulus equivalence.

In some cases, we can find mechanisms associated to simple recognition of image features that are unambiguous signs of the presence of the object.

But, in most cases there is no way to construct the **learning complex descriptions.**

What's the problem of object detection and recognition?

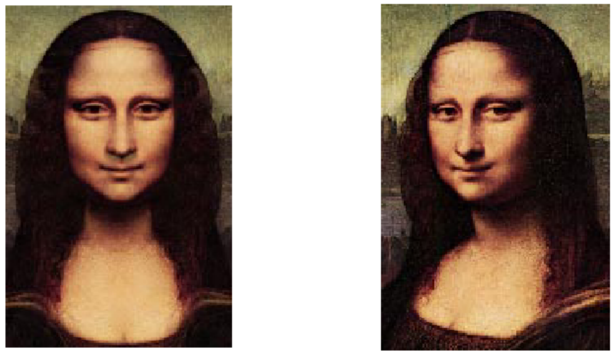
Problem: stimulus equivalence.

"The Cubist are destined... To give back to painting its true aim, which is to reproduce... objects as they are.
Lighting must be eliminated because ... it is the sign of a particular instant... As well, perspective must be eliminated because ... it is accidental a think like lighting"

J.Rivière. ***Present tendencies in painting***.
Revue d'Europe et Amérique, Paris, March 1912.

What’s the problem of object detection and recognition?

Difference between classify (categorize) and identify.



What's the problem of object detection and recognition?

Why is it useful to recognize classes?

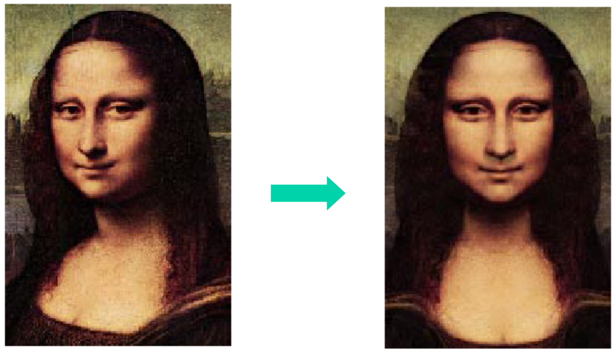
Recognition of new objects within a class:

- We can infer the properties (uses, dangers, ...) of things we've never seen! (Imagine the first European who saw a tiger!);

- Restricts the number of models for identifying (indexing);

- Allows the use of specific information to identify the class (eg neutralizing facial expressions);

- It enables generalization from very partial information (e.g. Mona Lisa!).



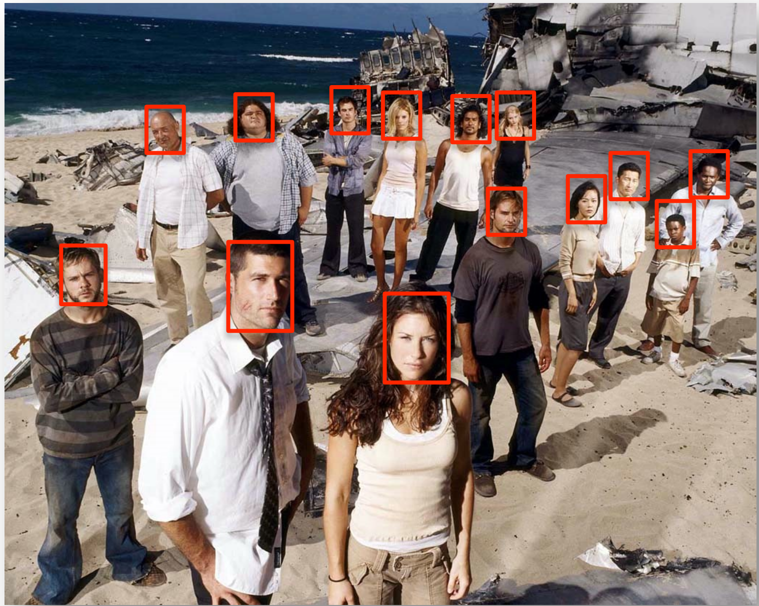
Face detection vs. recognition

THE PROBLEM

Given an image, automatically detect faces – i.e. detect the location of faces.

Detecion vs. Recognition

Detection



Recognition

?

Jack Kate Sawyer

Supervised classification method

Classification of image features

Goal: Define a space of image features allowing to represent objects based on their appearance (or through a set of local features) in the image.

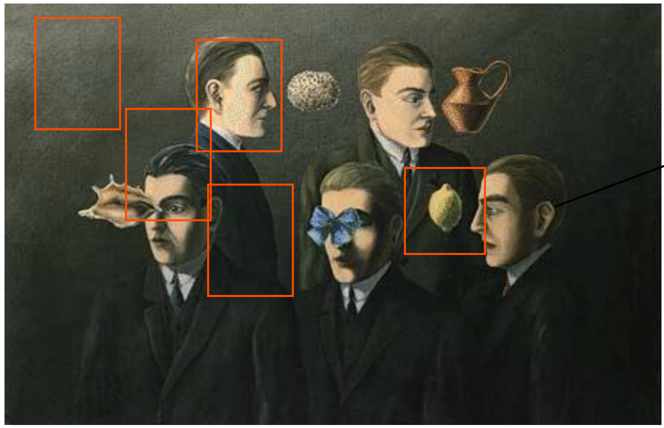
Three principal aspects:

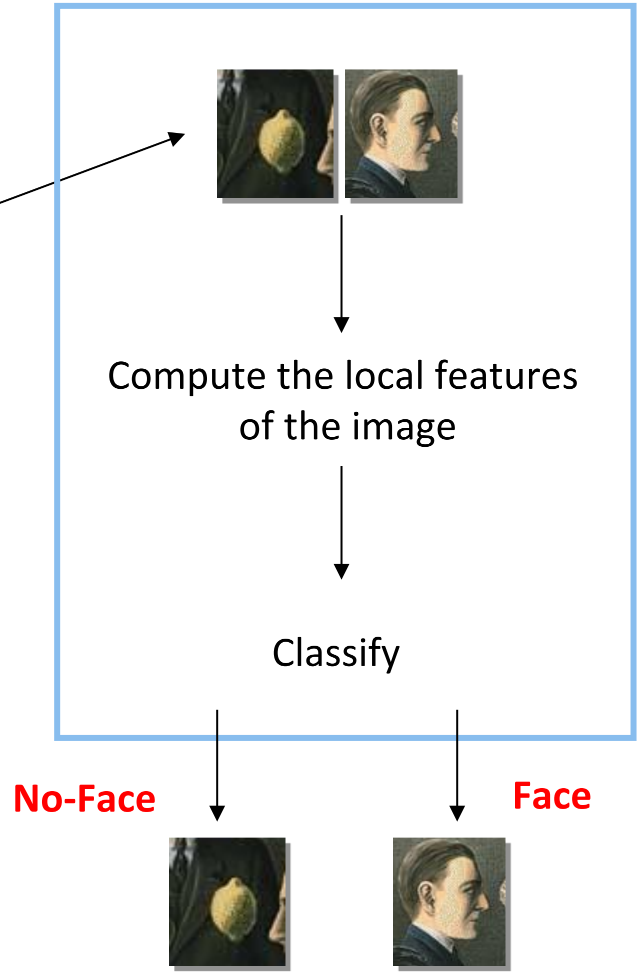
- **Adequate representation** (build descriptors of objects).
  - Normally, we try to reduce the size of the data so that the invariance is kept and the other dimensions are removed.

•**Training,** from a set of objects examples with their descriptors**.**

•**Detection or recognition** of a new object instance by using its descriptor and the learned model.

Face detection





The image is divided into a large set of windows (overlapped). Each window is classified as ”face" or ”no face", based on a set of local measures.

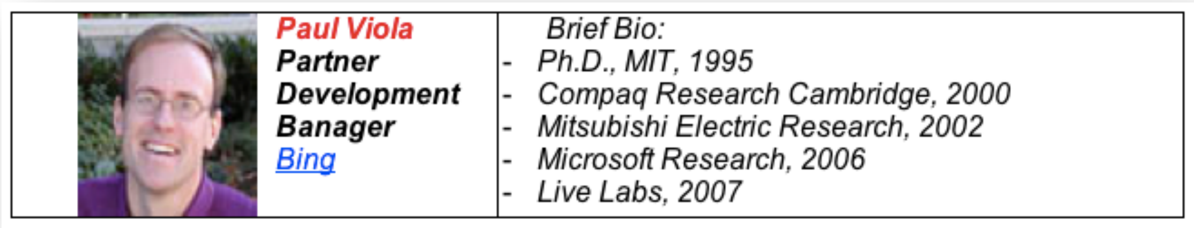
Face detection: Viola & Jones

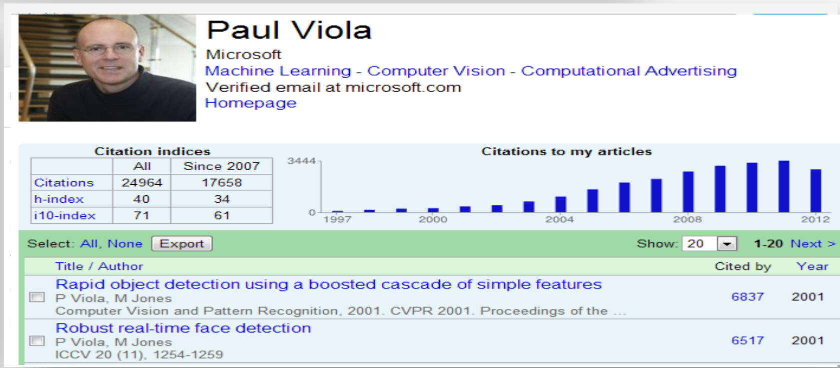
The main approach for face detection:

Robust Real-Time Face Detection

Paul Viola & Michael Jones

International Journal of Computer Vision, 2004.

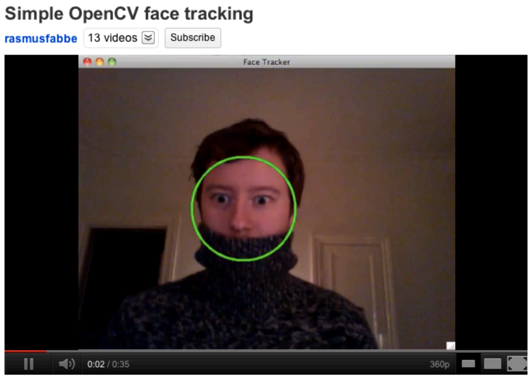




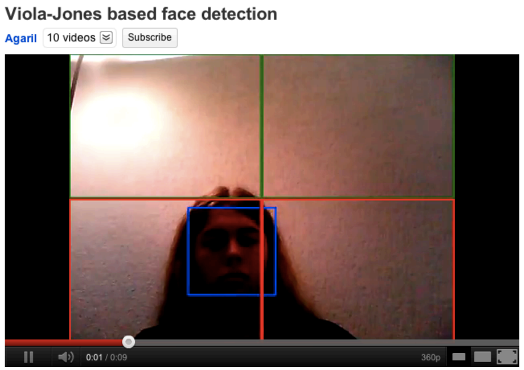
Face detection: Viola & Jones

LINKS

http://www.youtube.com/watch?NR=1&v=JyBMxeVCQkc

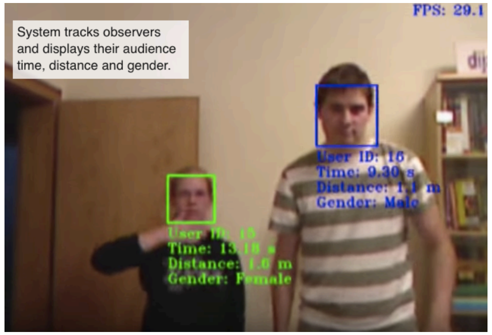


http://www.youtube.com/watch?NR=1&v=lvBvFHEX-CY



http://www.youtube.com/watch?v=jy8uwRK_zxU

https://www.youtube.com/watch?v=20AulLgWNR8



http://videolectures.net/lmcv04_verri_clafa1

Viola & Jones

Goals: of the face detector of Viola & Jones:

- Accurate detection of faces
- Fast algorithm!
- Real-time detection (video processing)

With a camera we can not wait long to take the photo!

Viola & Jones

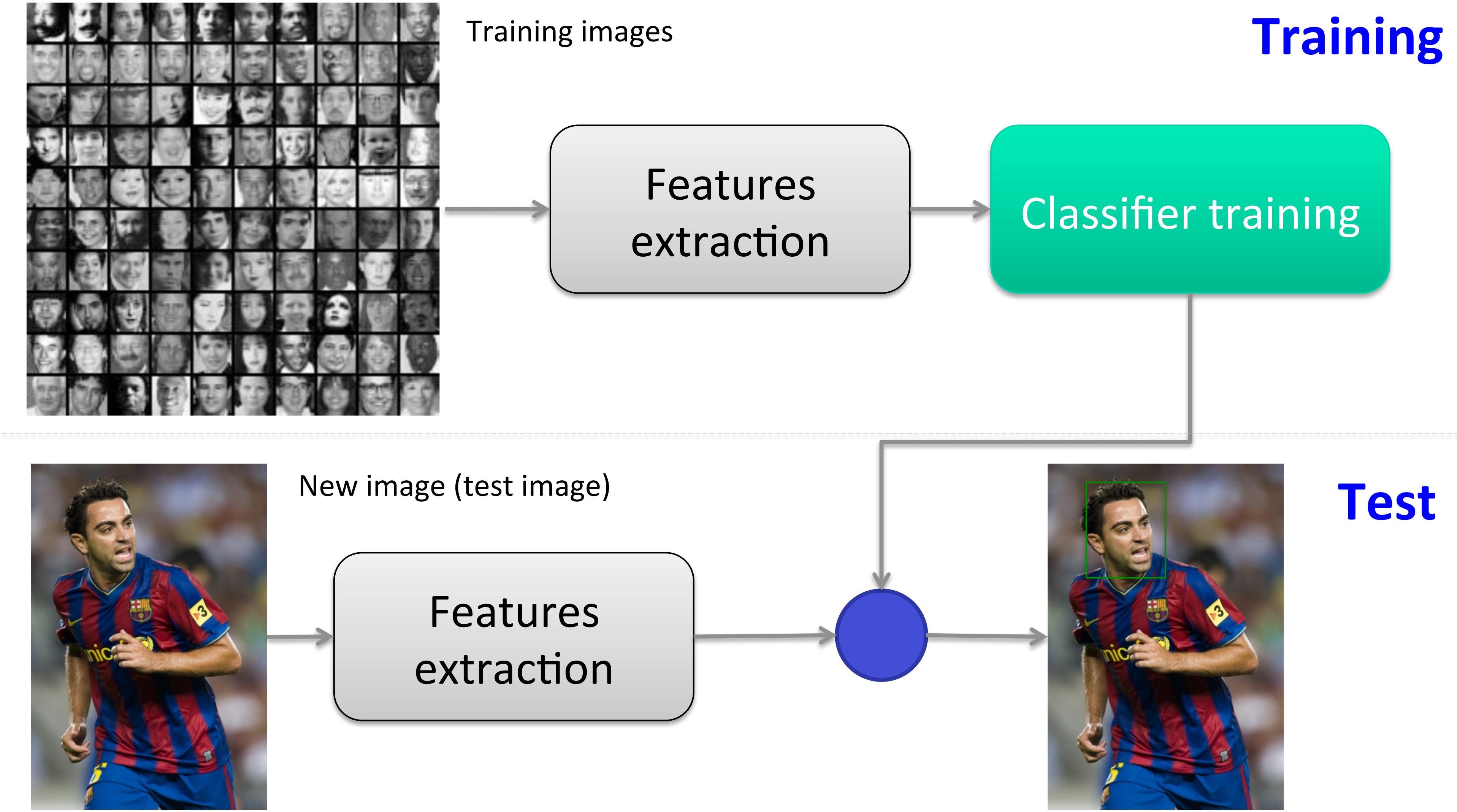
Main focus

The Viola & Jones' method is formulated as a standard supervised classification problem.

The major steps are basically three:

- Extraction of image features
- Training a decision rule, called classifier
- Test for new images using the trained classifier.

Supervised classification scheme



Viola & Jones

PRINCIPAL CONCEPTS

In this class, the main concepts to be learned are:

A) Features

- "Rectangular" features (Haar features)
- Integral images

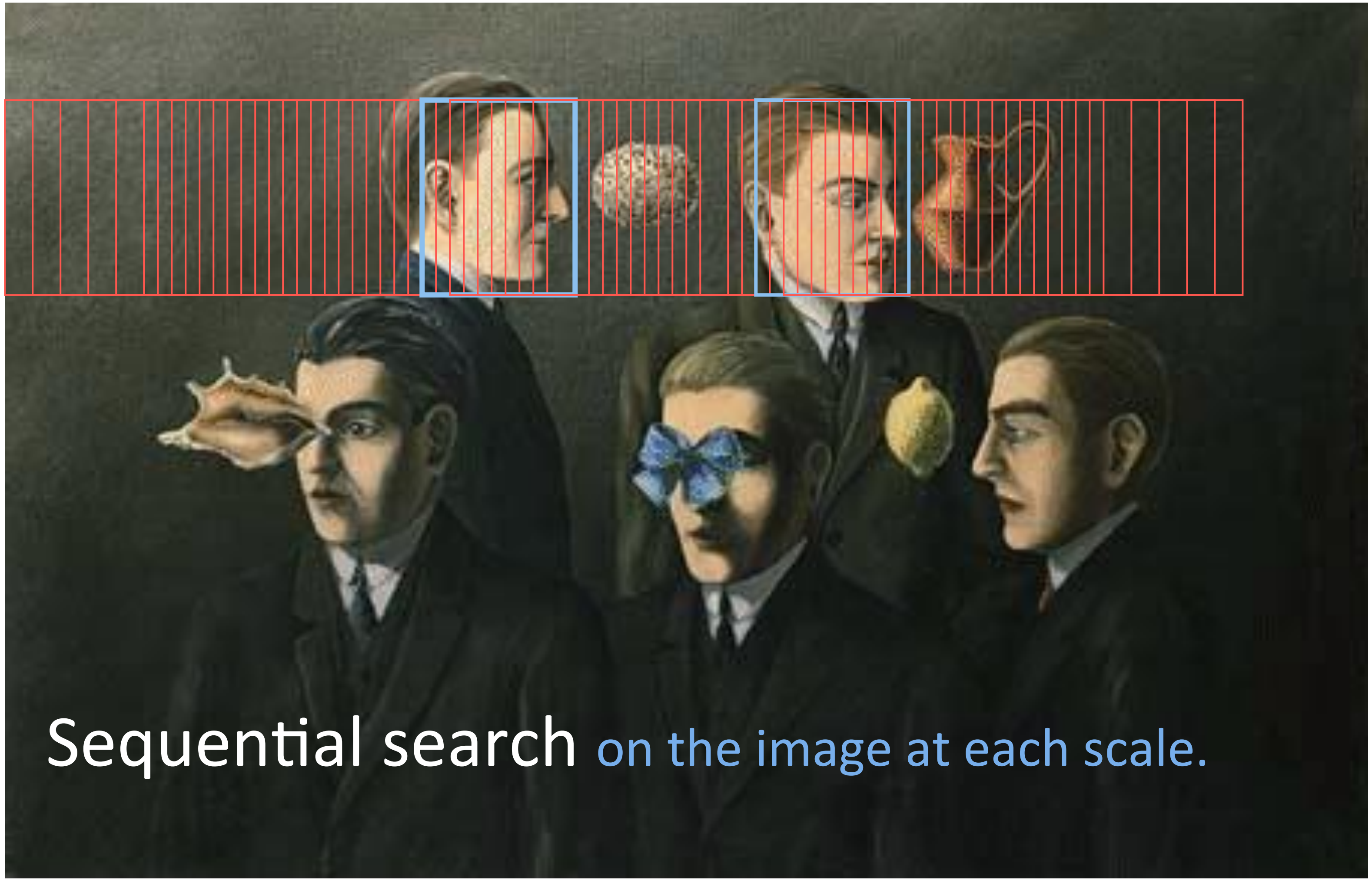
B) Classifier

- AdaBoost Classifier
- Multiple Classifier (Cascade classifiers)

These concepts are sufficient to understand the face detector of Viola & Jones!

Finding faces are always done through a **sliding window** across the image, so we are talking about **sequential search**.

Face detection



Given the sliding window for each pixel, the question is: is there a face?

Face detection

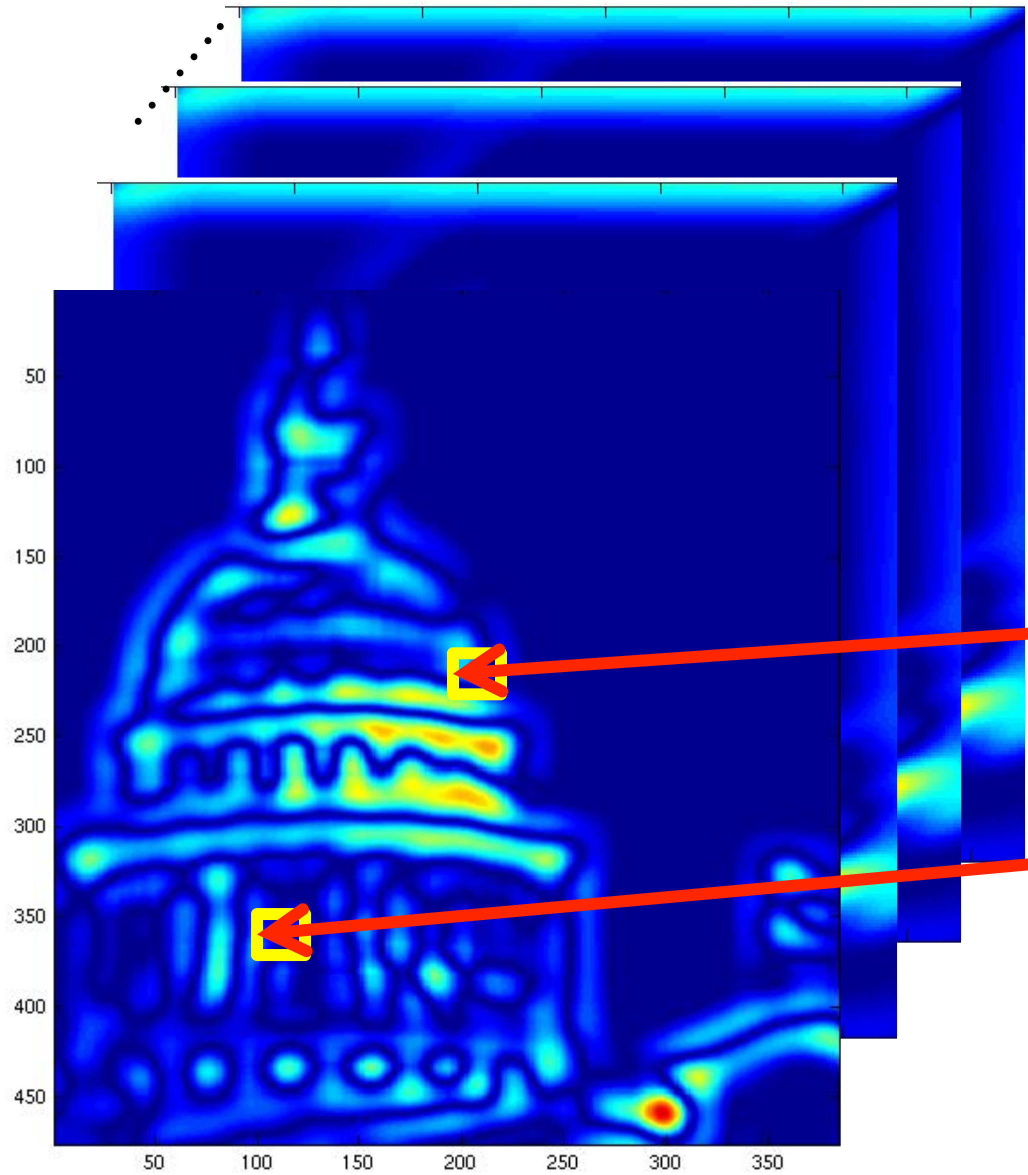
1) **Haar image features**

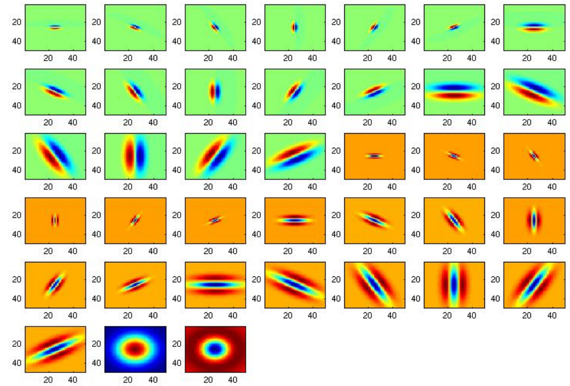
2) Integral images

3) AdaBoost

4) Classifiers cascade

Remember?





We can form a feature vector from the list of responses at each pixel.

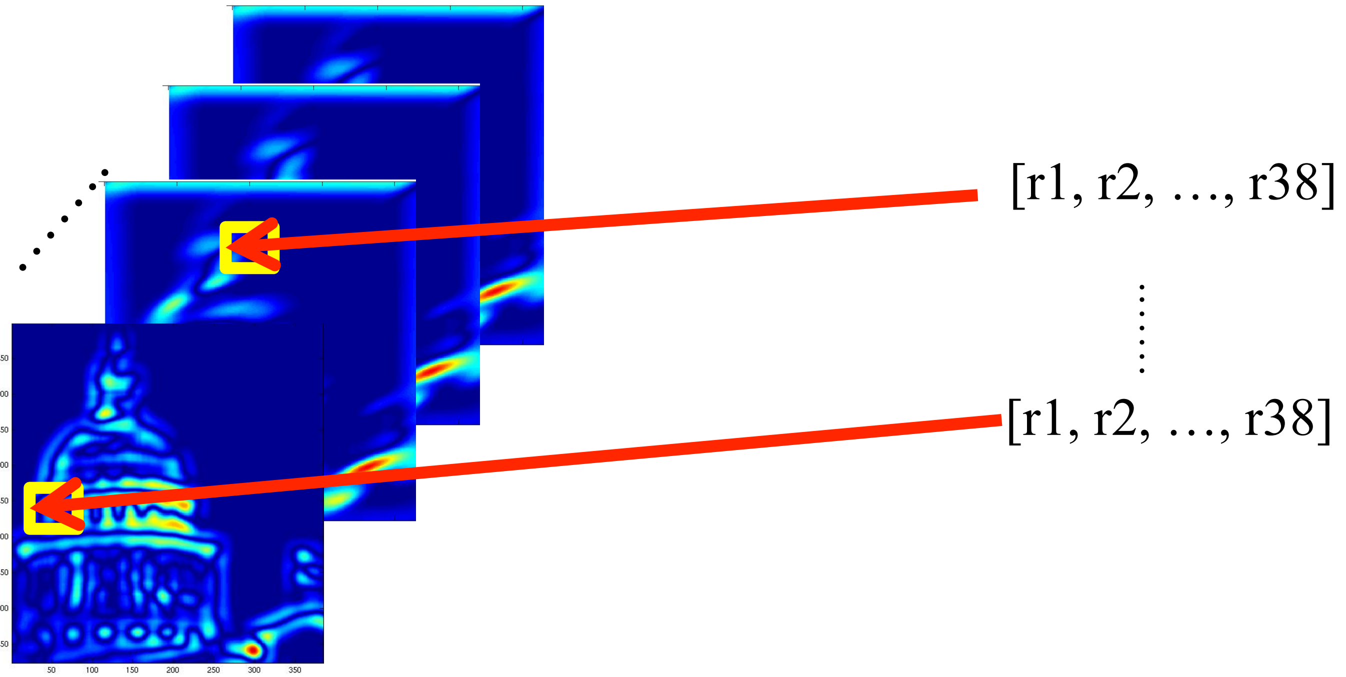
[r1, r2, …, r38]

⋮

[r1, r2, …, r38]

Kristen Grauman

Remember: Going from pixel representation to image representation?



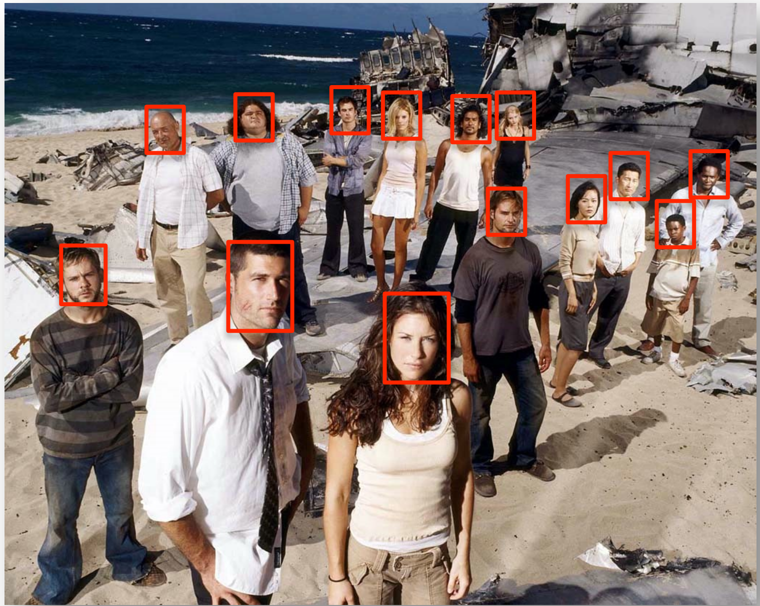
A simple way to represent the whole image is to get the mean abs value of each feature:

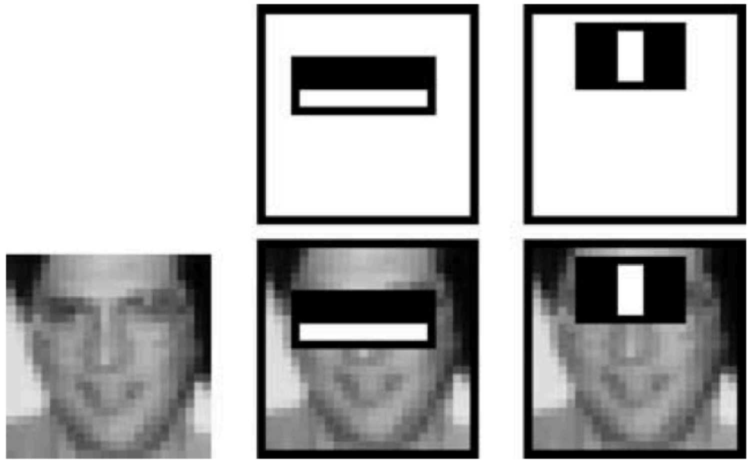
$$I \rightarrow f(I)=[mean_{all\ pixels}(|r1|),\ mean_{all\ pixels}(|r2|),\ \ldots,\ mean_{all\ pixels}(|r38|)]$$

Features

HAAR Features

What is the information they provide, when applied to face images?

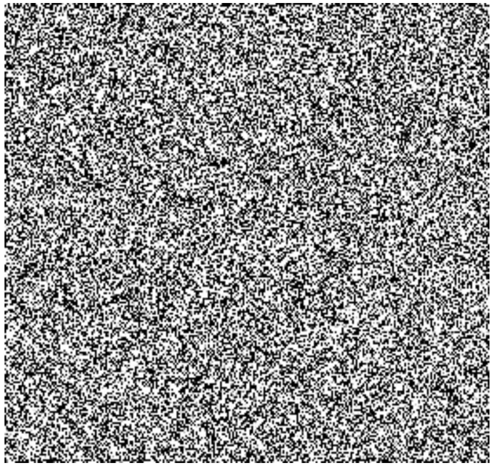




Difference in intensity between eyes and cheeks

Difference in intensity between eyes and nose

Example



Source

Result

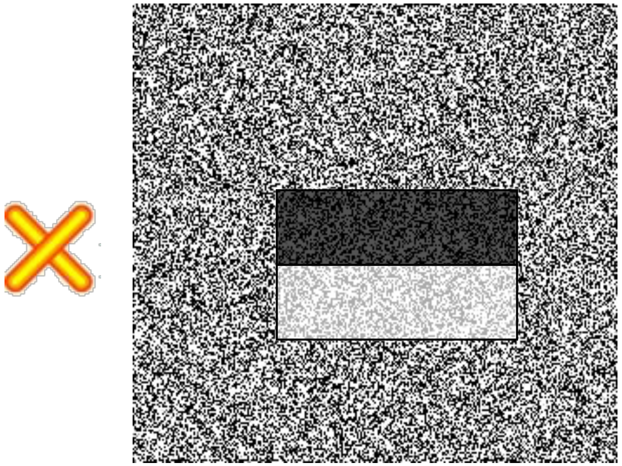


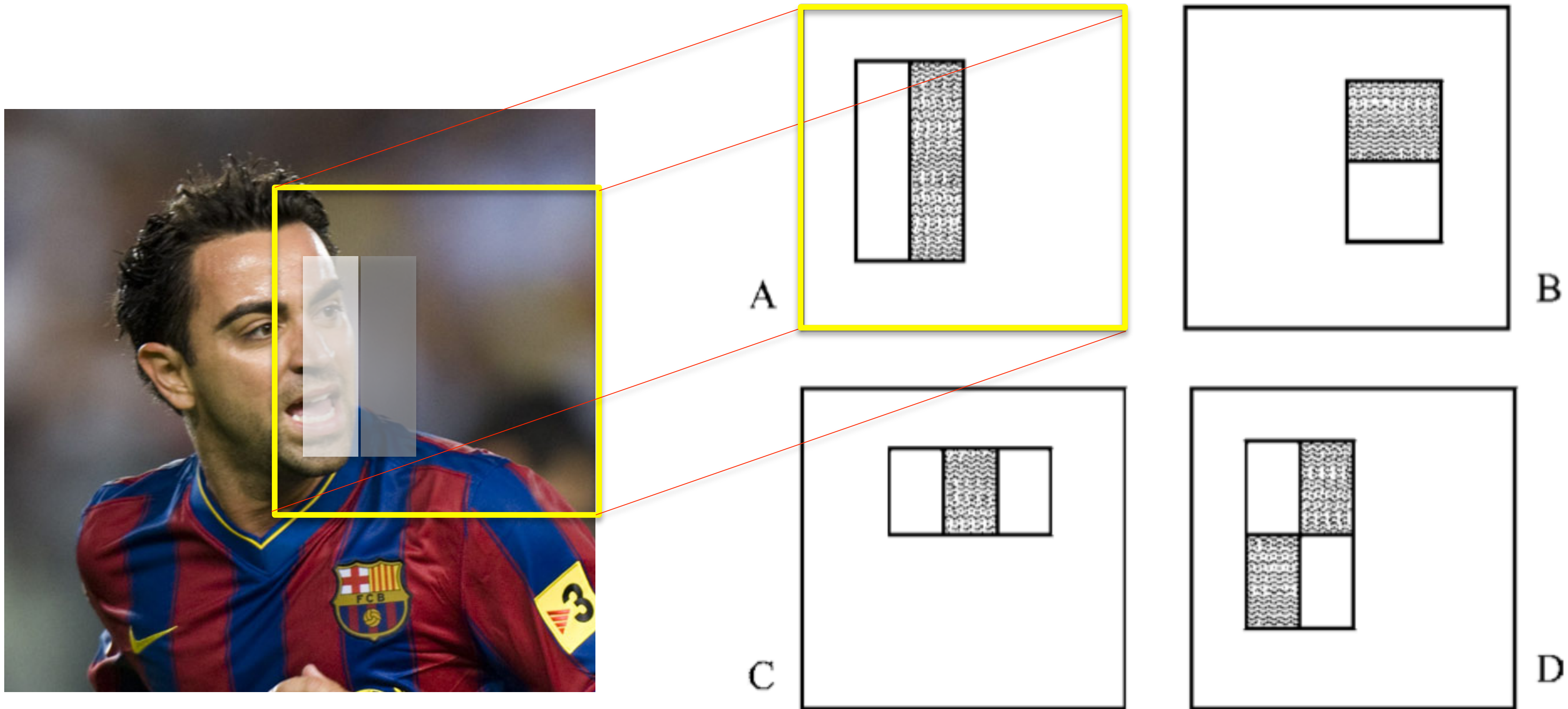
Image features

Haar features

Considering an image, we consider a square window in a given position: this window apply the Haar-type features. Repeat for each posible position.

- How Haar features are interpreted?
- What image structures do they show?

1st order: Vertical and horitzontal changes



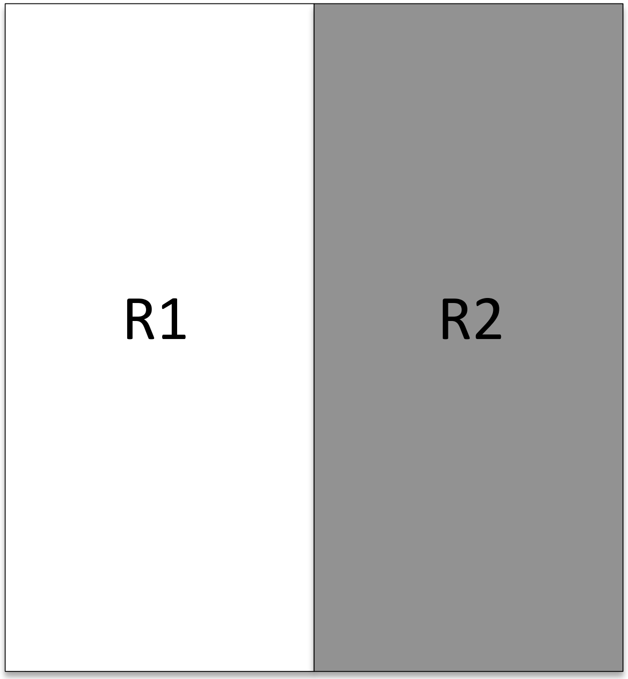
2nd order

1st order: Diagonal changes

Features

Haar features effect

Each feature mask, applied to the image I (x, y) indicates two regions:



The F value of the image feature corresponding to the mask that is applied in the particular pixel is the sum of the pixels in the white region minus the sum of the pixels in the dark region:

$$F_k = \sum_{(i,j)\in R1} I(i,j) - \sum_{(i,j)\in R2} I(i,j)$$

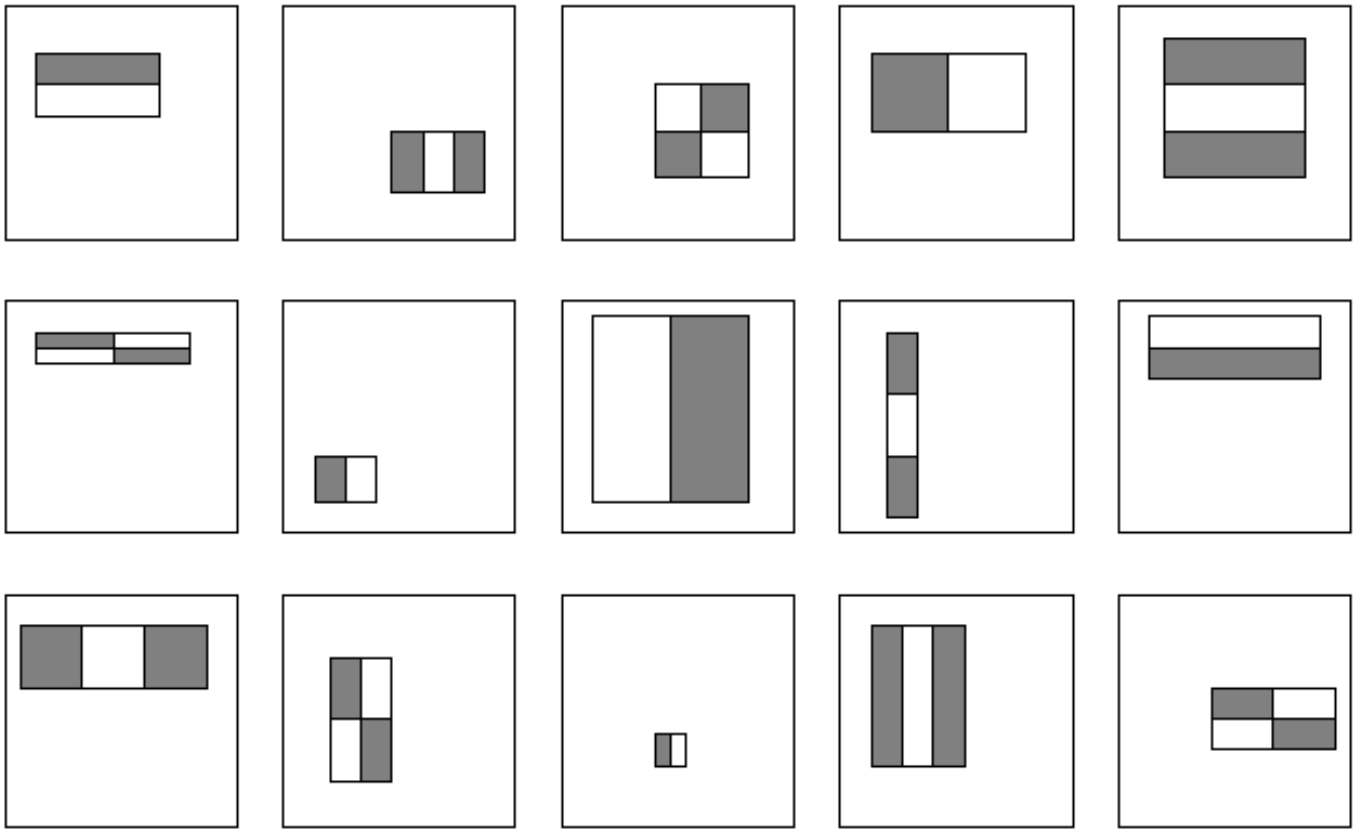
Typically, a set of K masks is used:

- F_k is the feature that corresponds to the k^{th} mask.

Features

Feature set

The masks have different size, shape, and position with regard to the square window



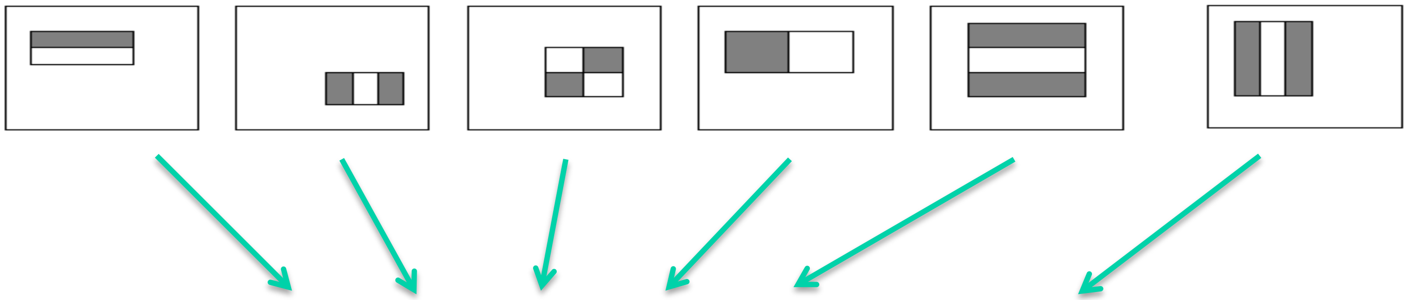
You can define up to **16 million masks** using:

- Different shape
- Different scale
- Different location of the squares.

Image features

Features extraction

Given the set of masks, how is the feature vector of each window constructed?



$$\mathbf{X} = [\, F_1,..\ F_i,..\ F_j,..\ F_k,...\ F_l,\ ...\ ,\ F_n]$$

The feature vector describes the content of the window, and is used to train the classifier to detect the face and later, given the trained classifier to detect a face in new images.

Face detection

1) Haar image features

2) **Integral images**

3) AdaBoost

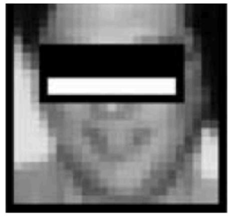
4) Classifiers cascade

Integral Images

The main requirement of Viola & Jones detector is SPEED!

How can we compute Haar features quickly?

Convolution?

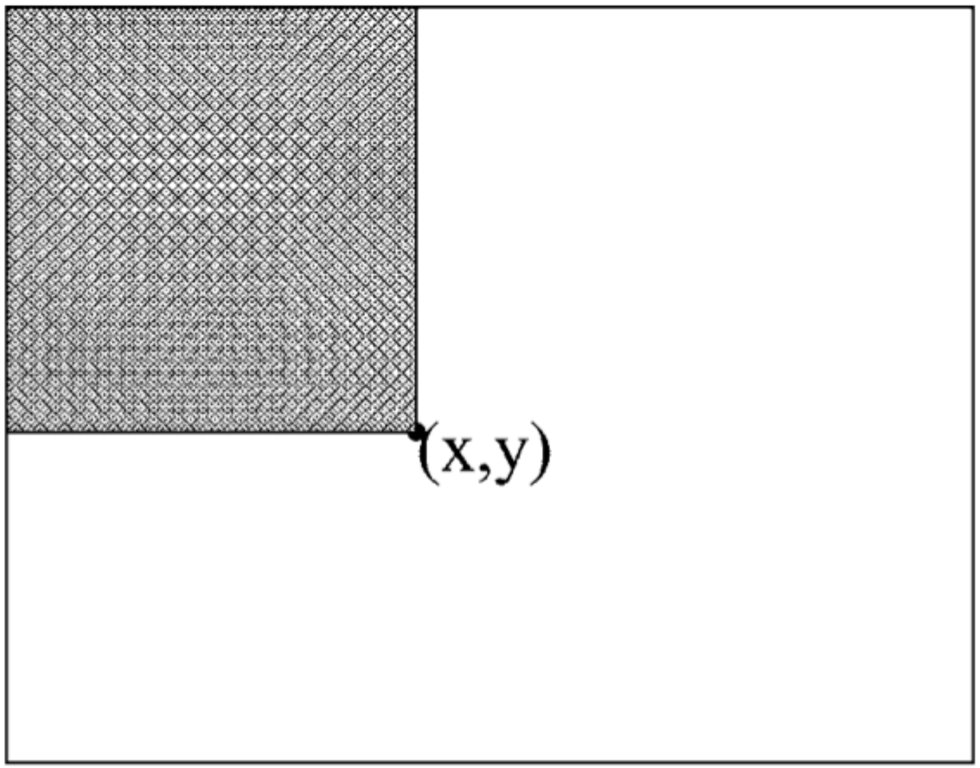


$$\cancel{I(x,y)\otimes K(x,y)=\iint I(x,y)K(x-\tau_x,y-\tau_y)d\tau_x d\tau_y}$$

INTEGRAL IMAGES

Integral Images

Df: The integral image S is constructed as follows: the value at position (x, y) is the sum of all pixels in the image I above and left of point (x, y):

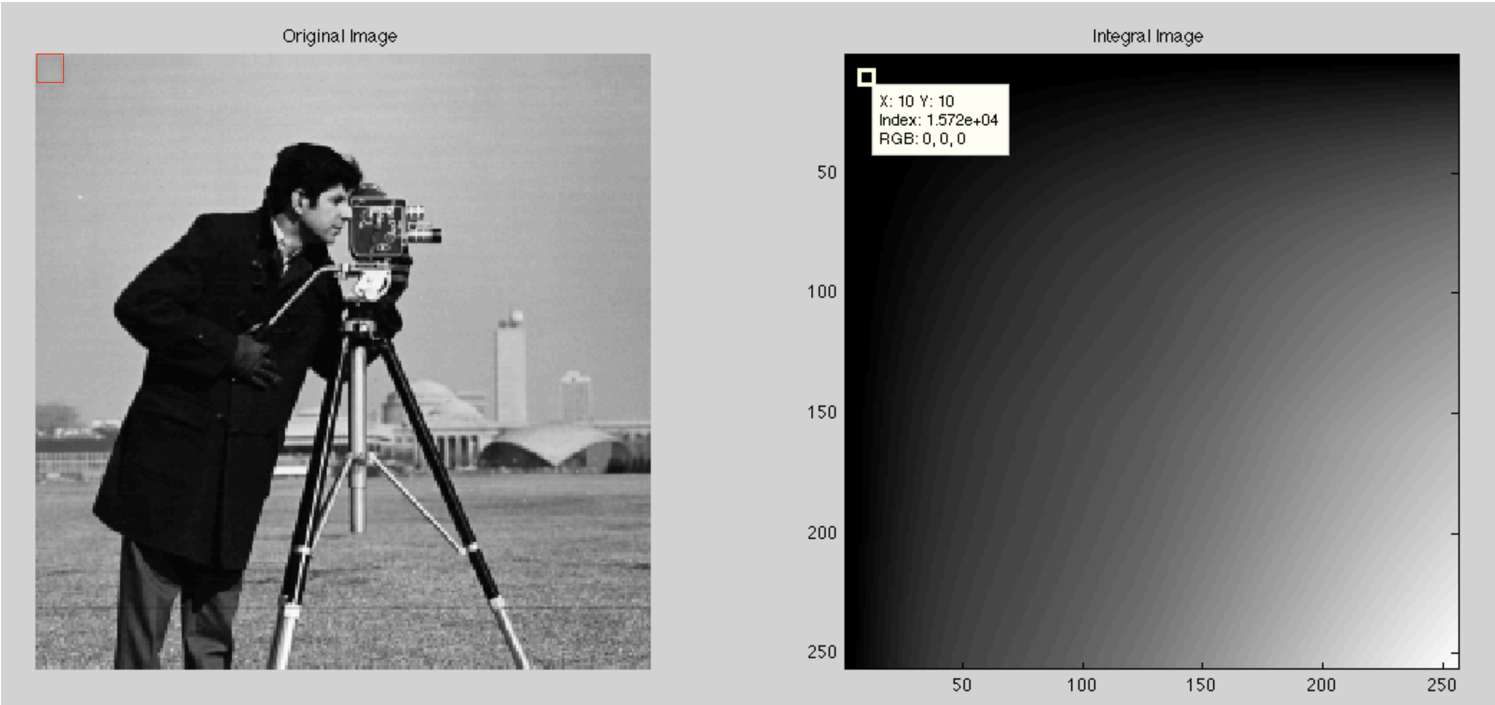


$$S(x,y) = \sum_{i=1}^{x} \sum_{j=1}^{y} I(i,j)$$

Integral Images

In **MATLAB**:

```
I = imread('cameraman.tif');
S = cumsum(cumsum(double(I),2));
```



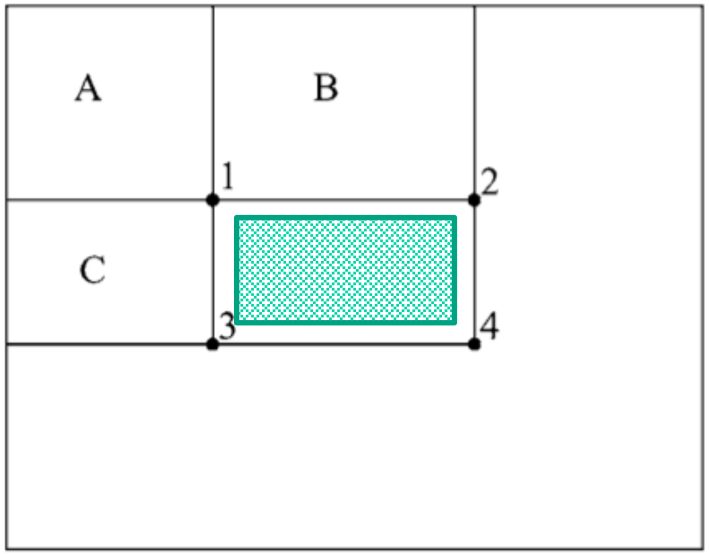
How much is: sum(sum(I(1:10,1:10))) ?

Integral Images

Why are we interested in the integral images?

The Haar features are based on sums and subtractions of image rectangles.

>> How to compute the area of a rectangle by an integral image?



1 = area of A

2 = area of A + B

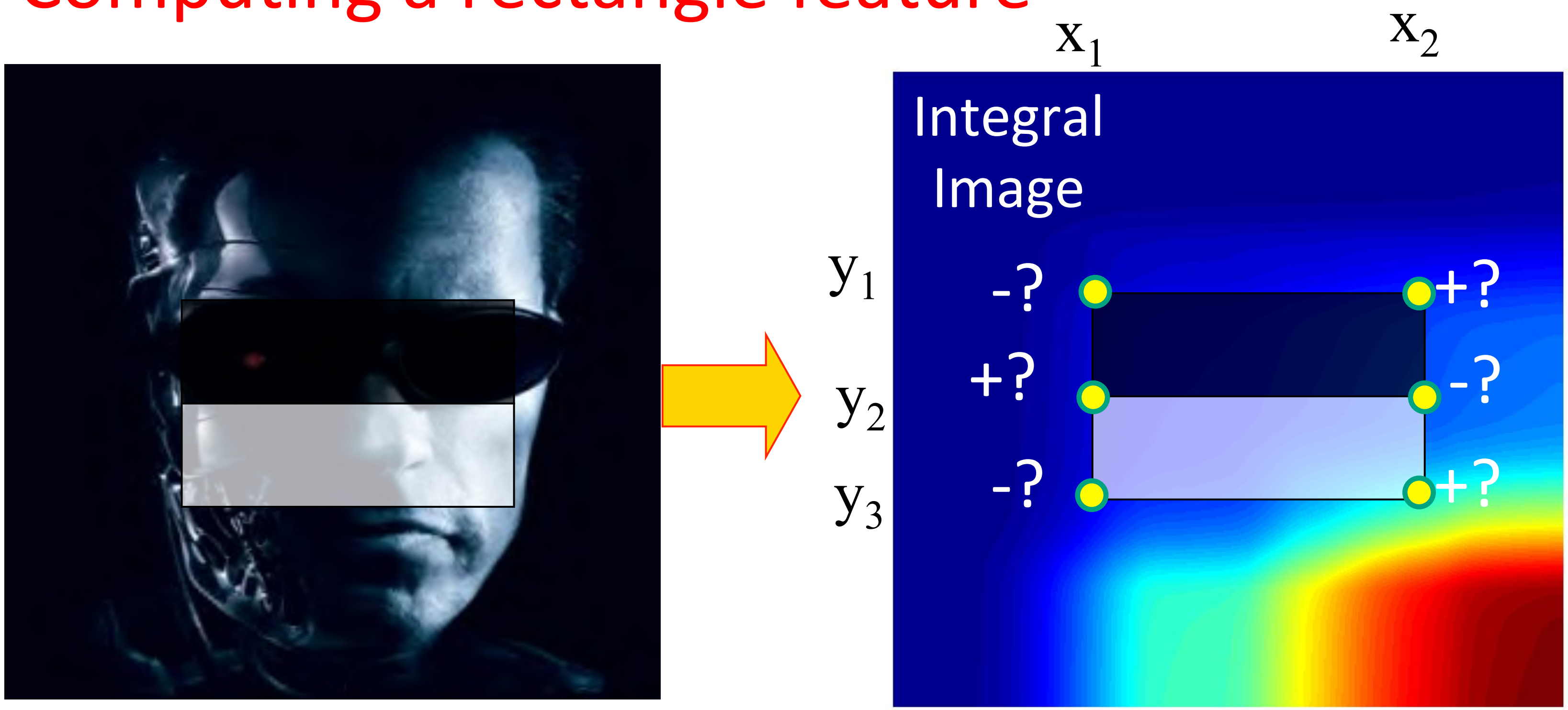
3 = area of A + C

4 = area of A+B+C+D

Area of **D = 4-2-(3-1)**

The computation of the features through integral images, is reduced to computing a set of 3 **sums and subtractions on the integral image points**!!! VERY FAST!

Computing a rectangle feature



Convolution with a mask is substituted by a subtraction of pixels of the integral image.

- What is each pixel of the integral image representing?
- If I is the integral image, what would be the result of the convolution:

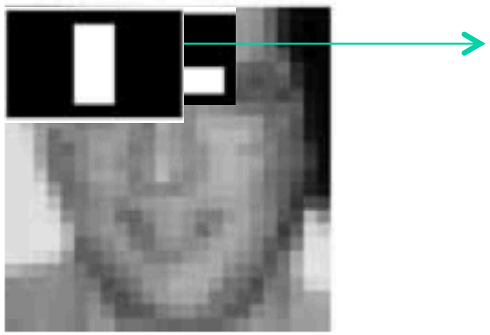
$$I(x2,y3) - I(x1,y3) - I(x2,y2) + I(x1,y2) - (I(x2,y2) - I(x1,y2) - I(x2,y1) + I(x1,y1))$$

$$I(x2,y3) - I(x1,y3) - 2 * I(x2,y2) + 2 * I(x1,y2) + I(x2,y1) - I(x1,y1)?$$

Features

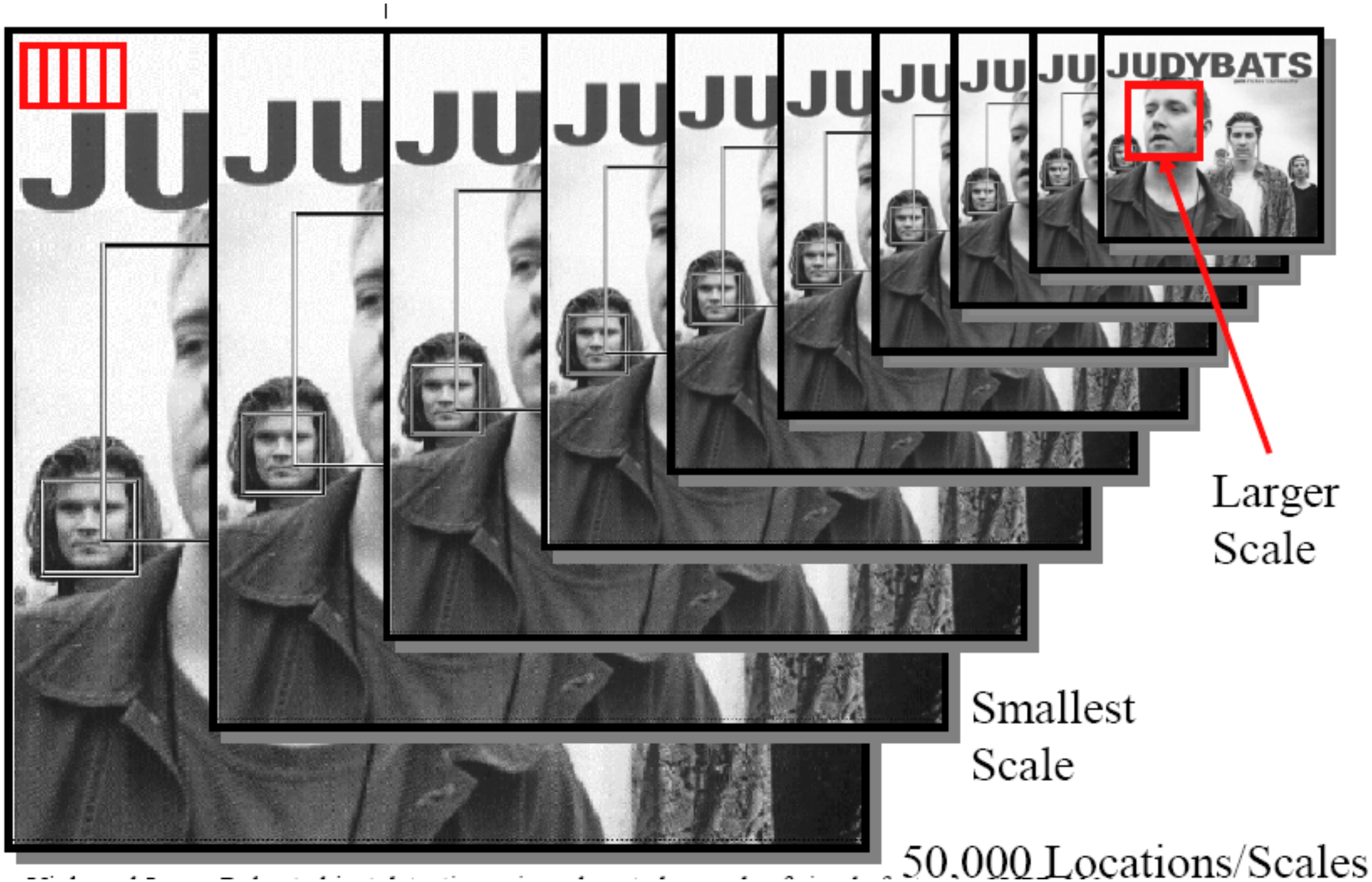
FEATURES FOR FACE DETECTION:

Computing values of the I and II order Haar features at different scale and at different points of the image of the face => thousands of features.



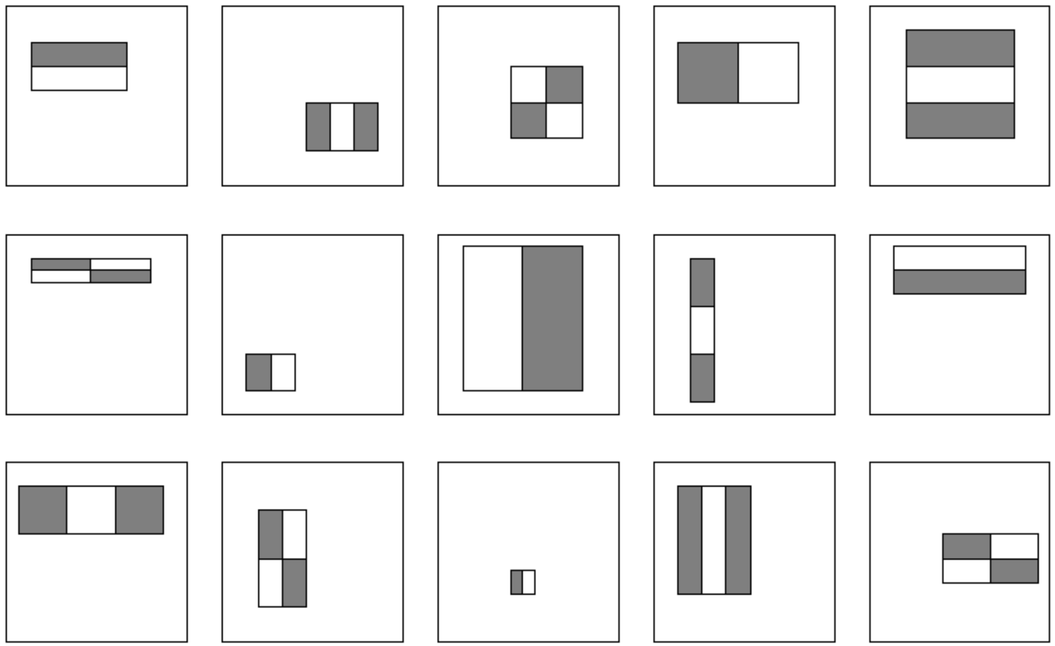
$x_1, x_2, x_3, \ldots x_n \quad x_{n+1} \quad \ldots x_m$

The scale



Re-scaling of the image instead of generating filters of different window.

Viola-Jones detector: features



Considering all possible filter parameters: position, scale, and type:

180.000+ possible features associated with each 24 x 24 window

Which subset of these features should we use to determine if a window has a face?

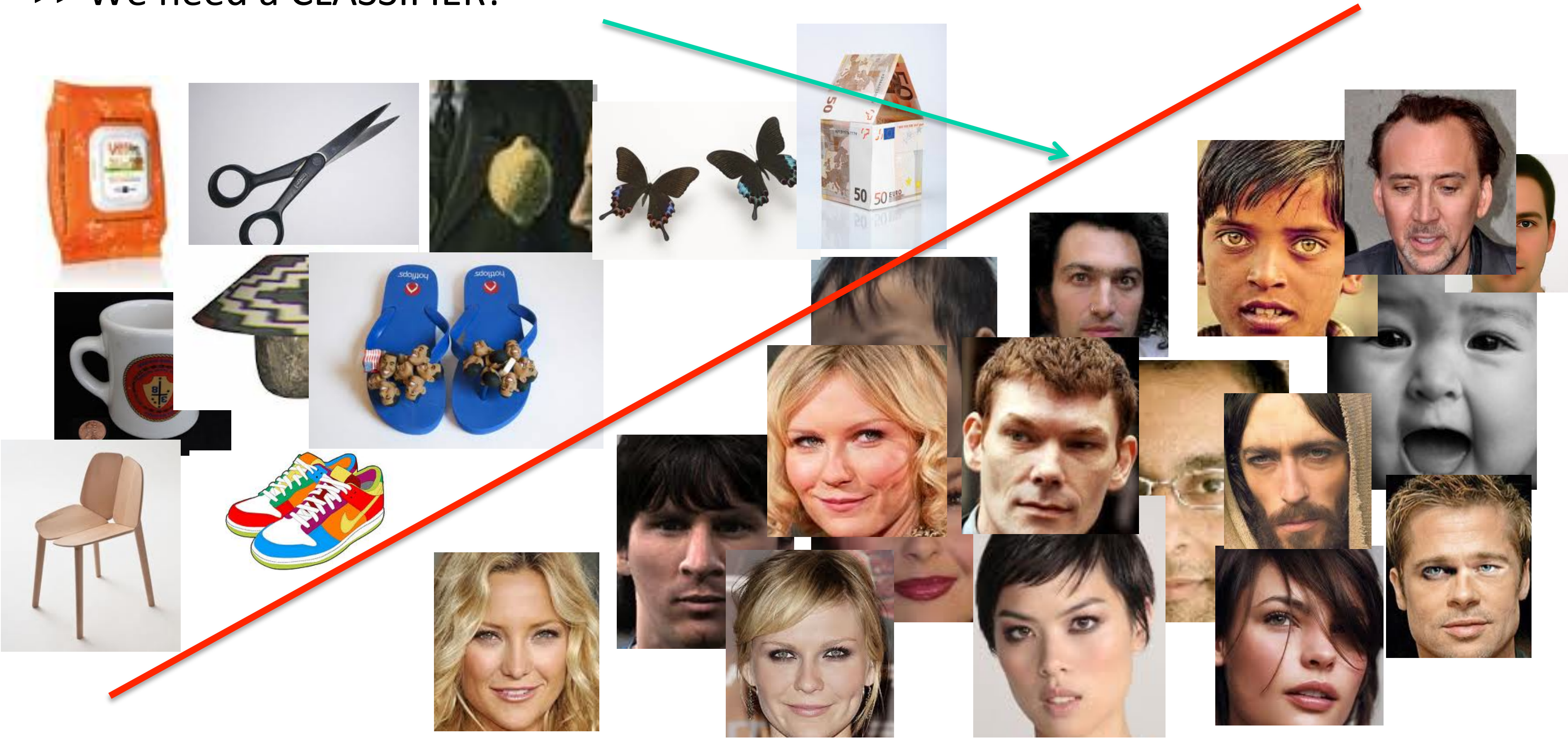
Face detection

1) Haar image features
2) Integral images
3) **AdaBoost**
4) Classifiers cascade

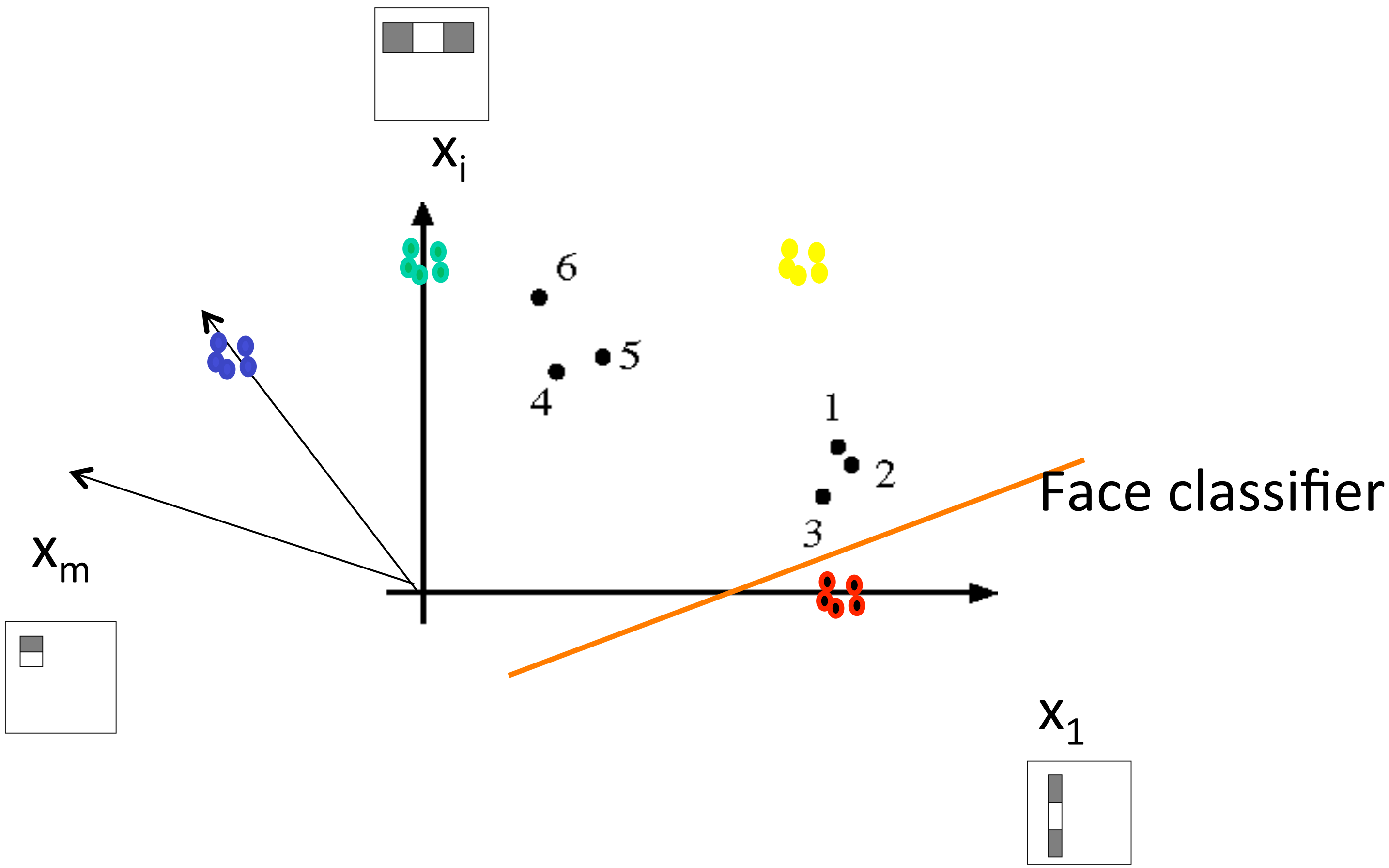
Face classification

In the feature space, we need to separate windows containing faces from those that do not contain them.

>> We need a CLASSIFIER!



Face classification



AdaBoost

Viola & Jones uses AdaBoost classifier to detect faces.

- AdaBoost: Adaptive Boosting
  Introduced by Freund & Schapire in 1999

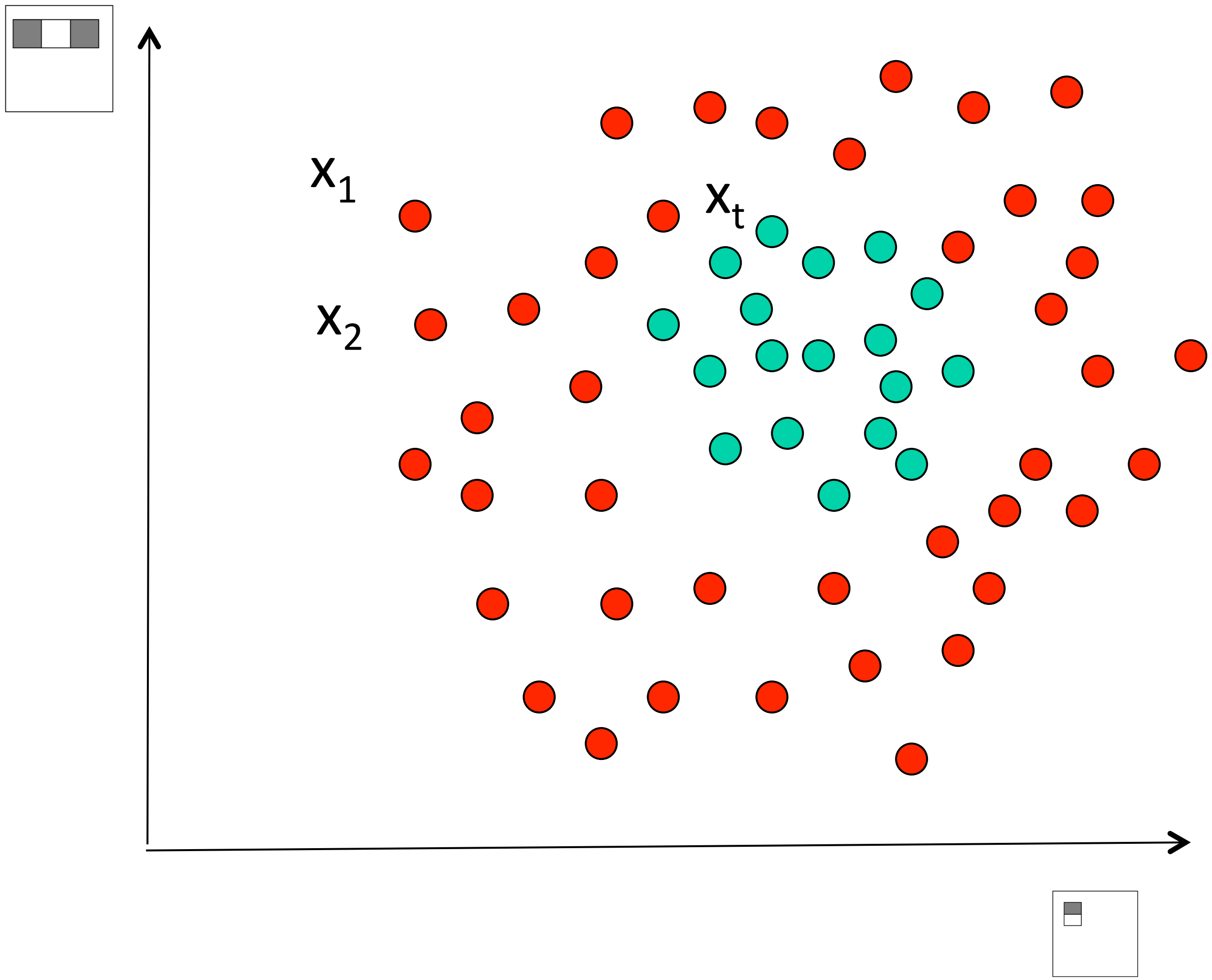
- It is a classification algorithm that joins several weak classifiers (weak) to form a single strong classifier (strong)
  BOOSTING >>

- The weak classifiers are defined at each iteration, and each is devoted to examples that are misclassified in the previous classifier:
  ADAPTIVE >>

From the Oxford Dictionary:
boost |bo͞ost|
verb [with obj.]
help or encourage (something) to increase or improve: *a range of measures to boost tourism*.

AdaBoost

ADABOOST EXAMPLE



Each point (image) is labeled with a class label:

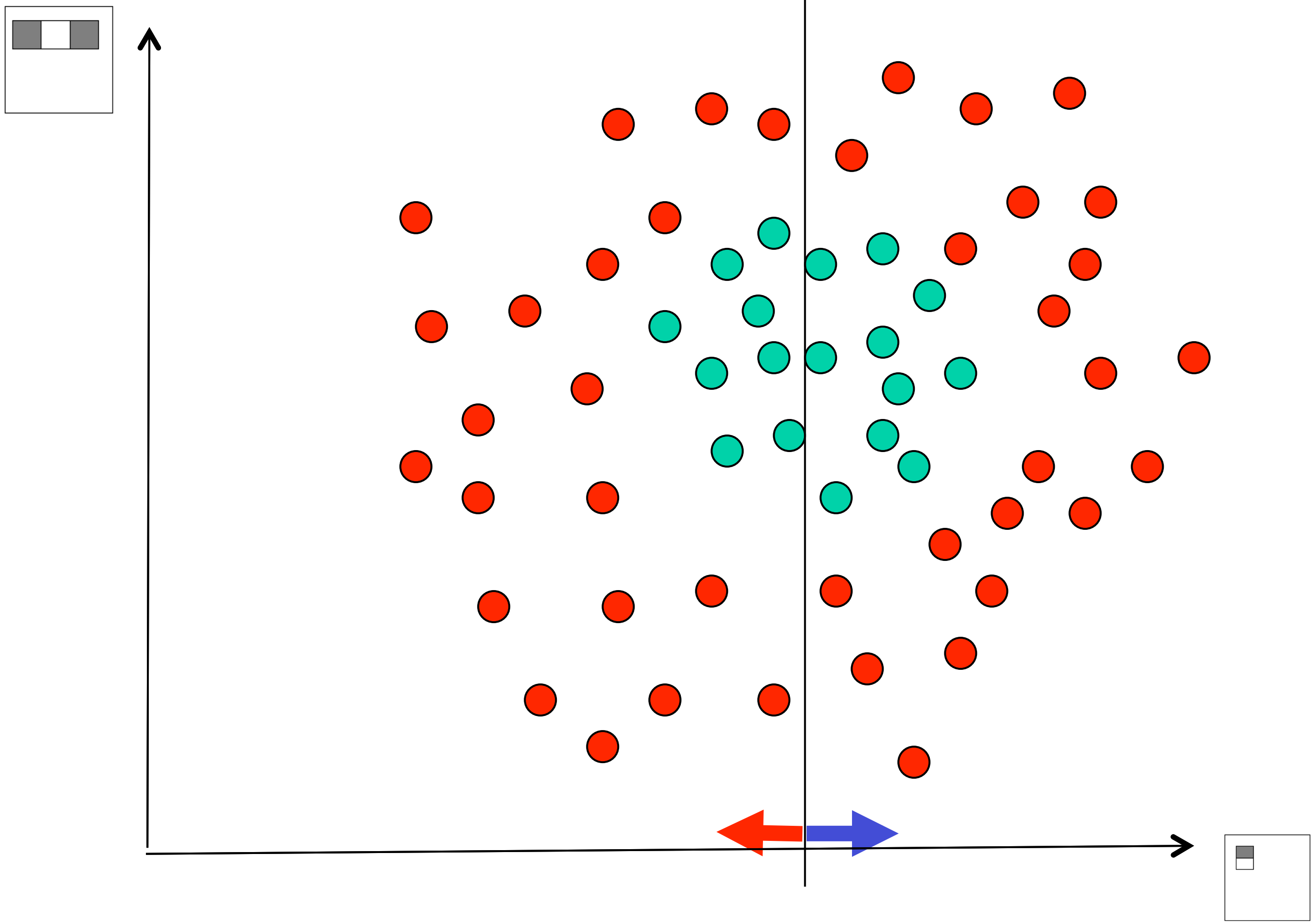
$$y_t = \begin{cases} +1 \; (\bullet) \\ -1 \; (\bullet) \end{cases}$$

and weight $w_t = 1/N$

N = points number.

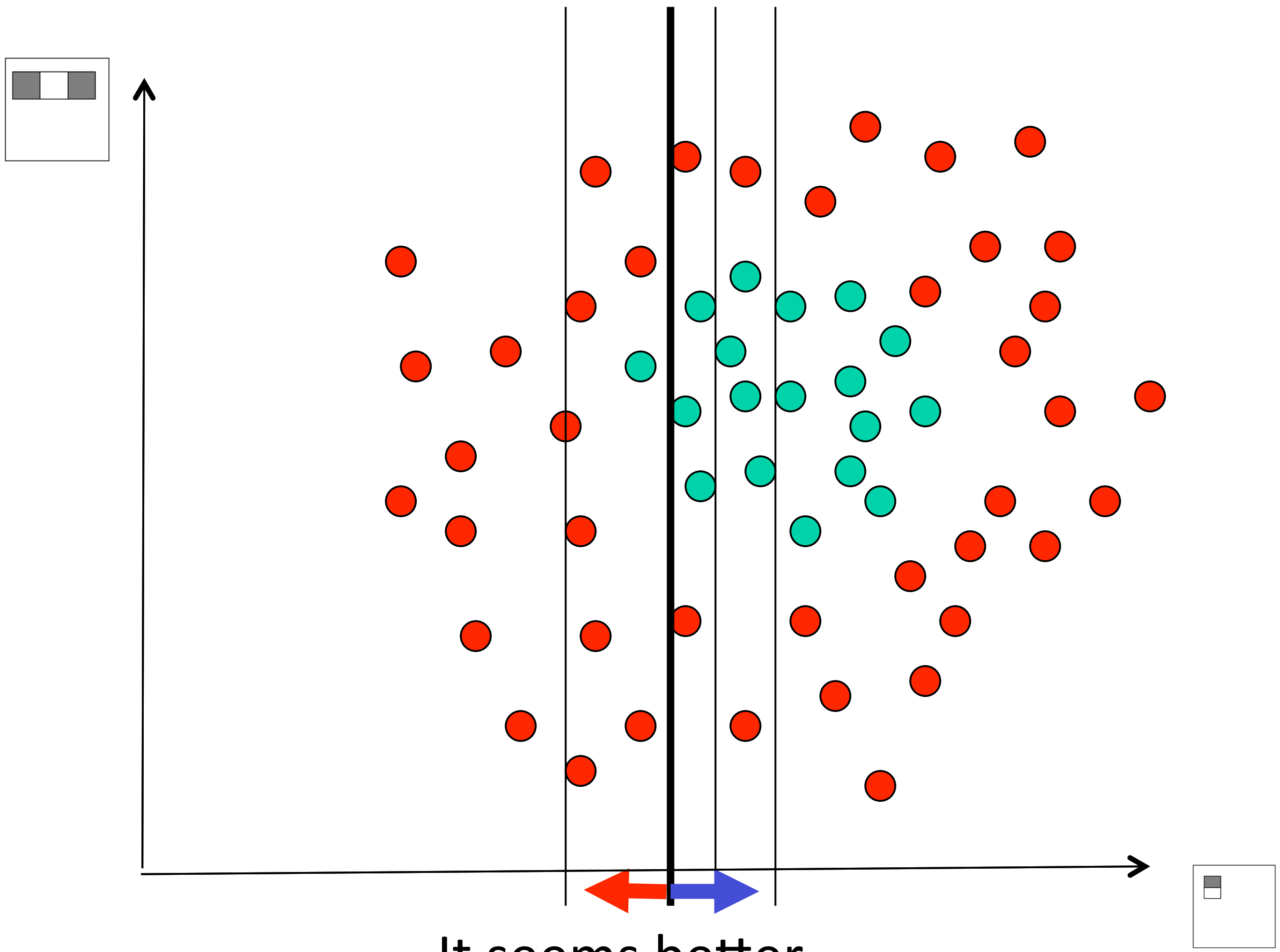
AdaBoost

Let's consider a weak classifier (e.g. a line)



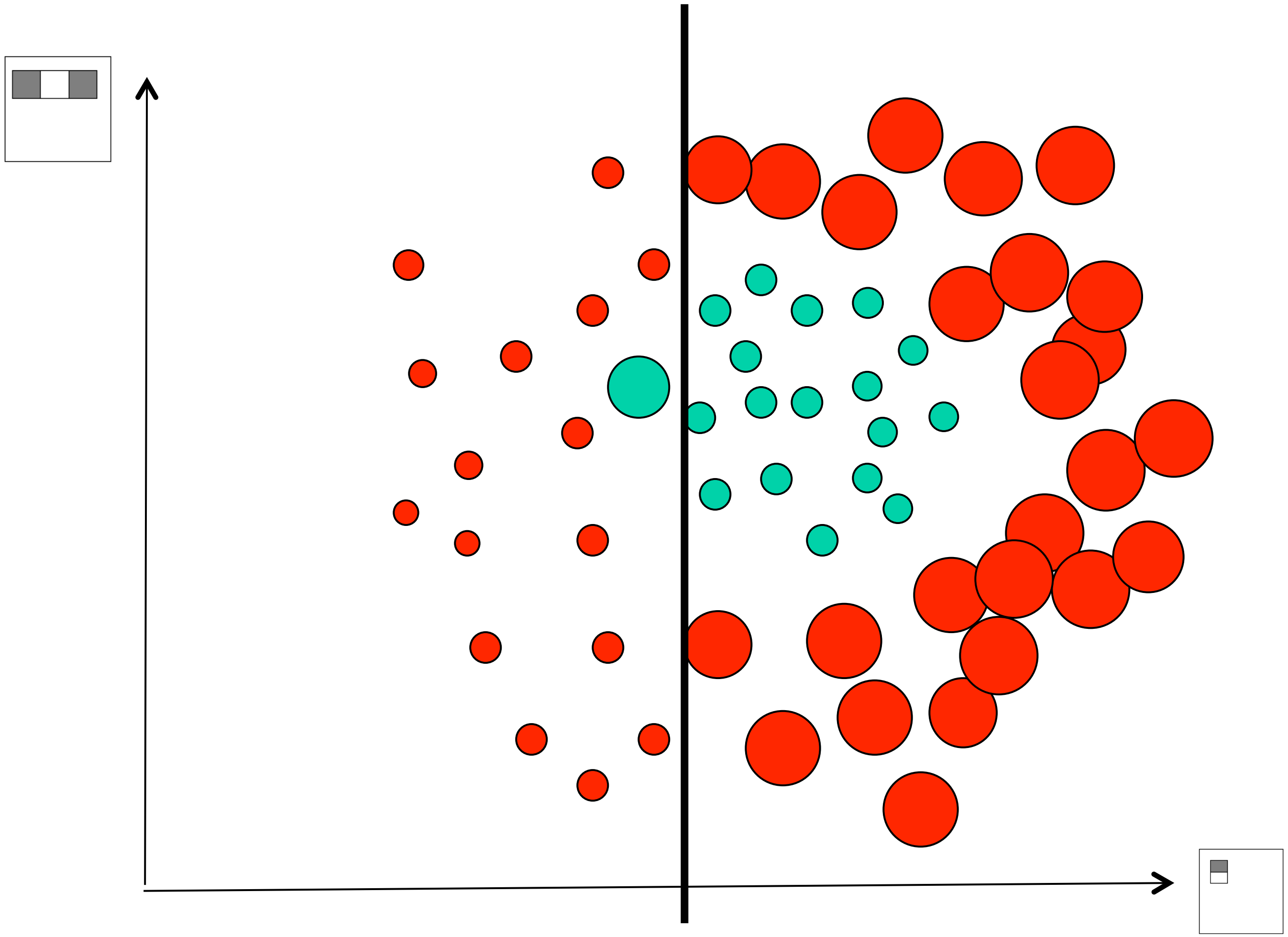
h => p(error) = 0.5 random

AdaBoost



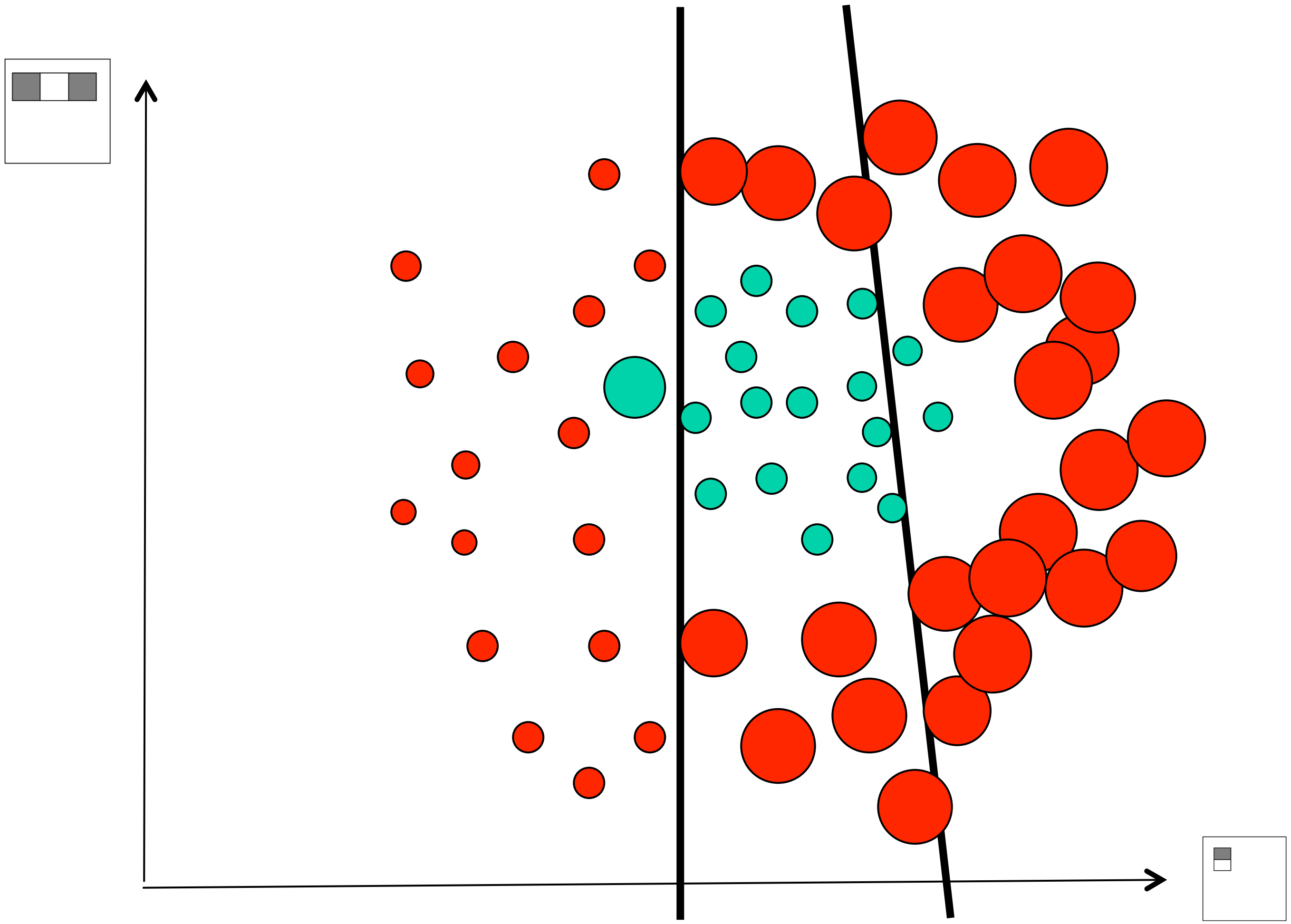
It seems better

AdaBoost



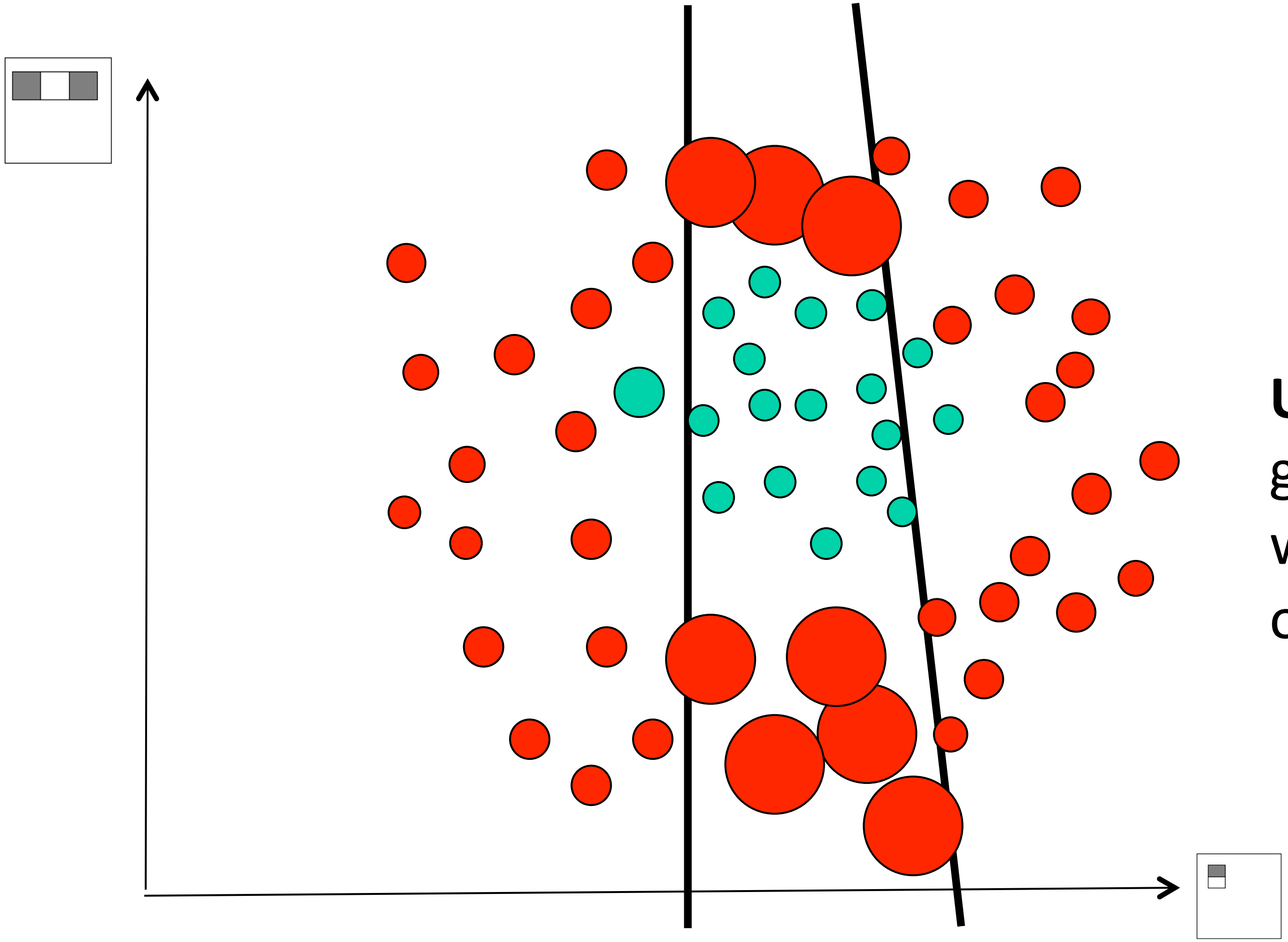
We consider the new problem and seek the best weak classifier.

AdaBoost



We consider the new problem and seek the best weak classifier.

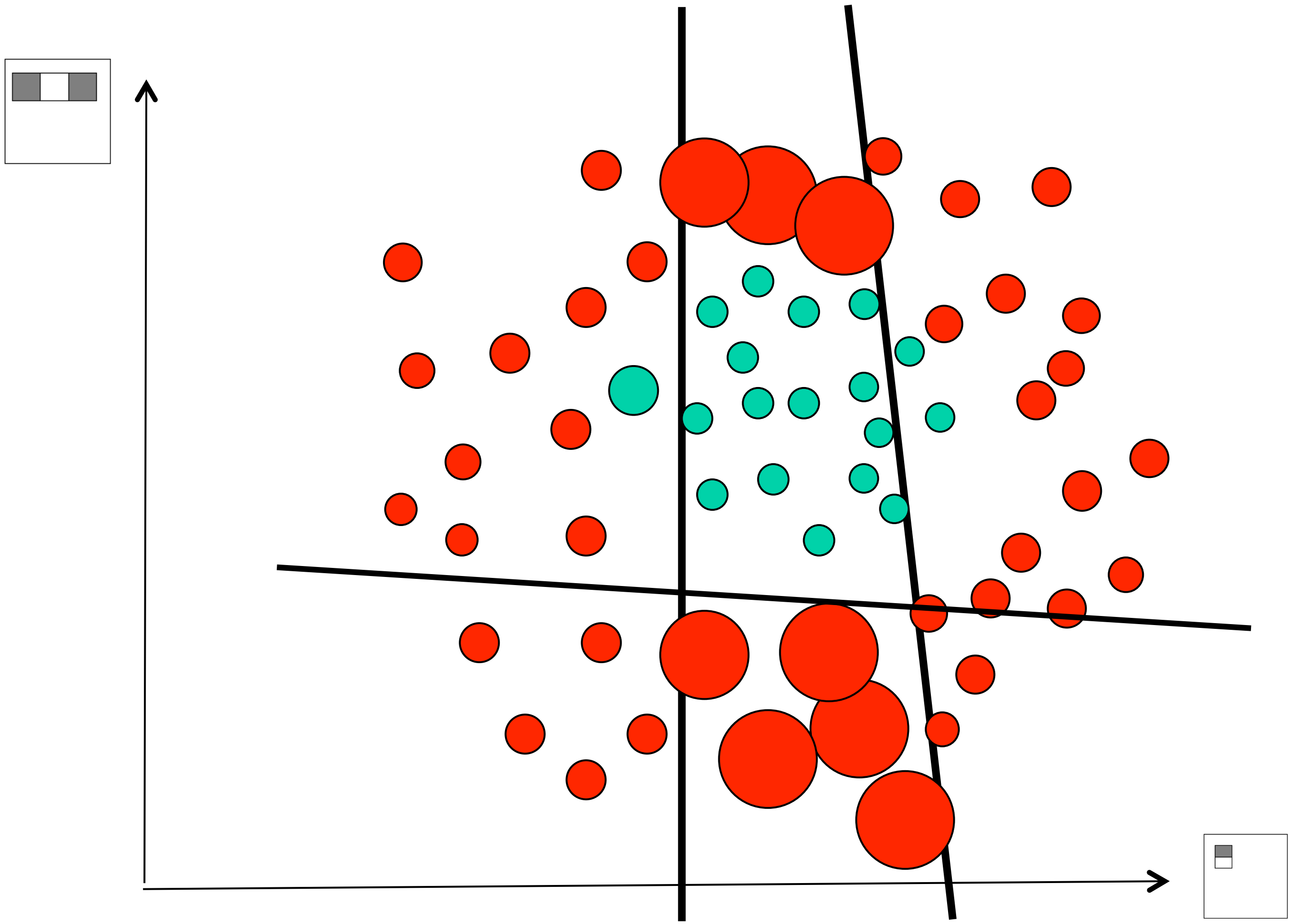
AdaBoost



Update the weights: giving priority to wrongly classified ones.

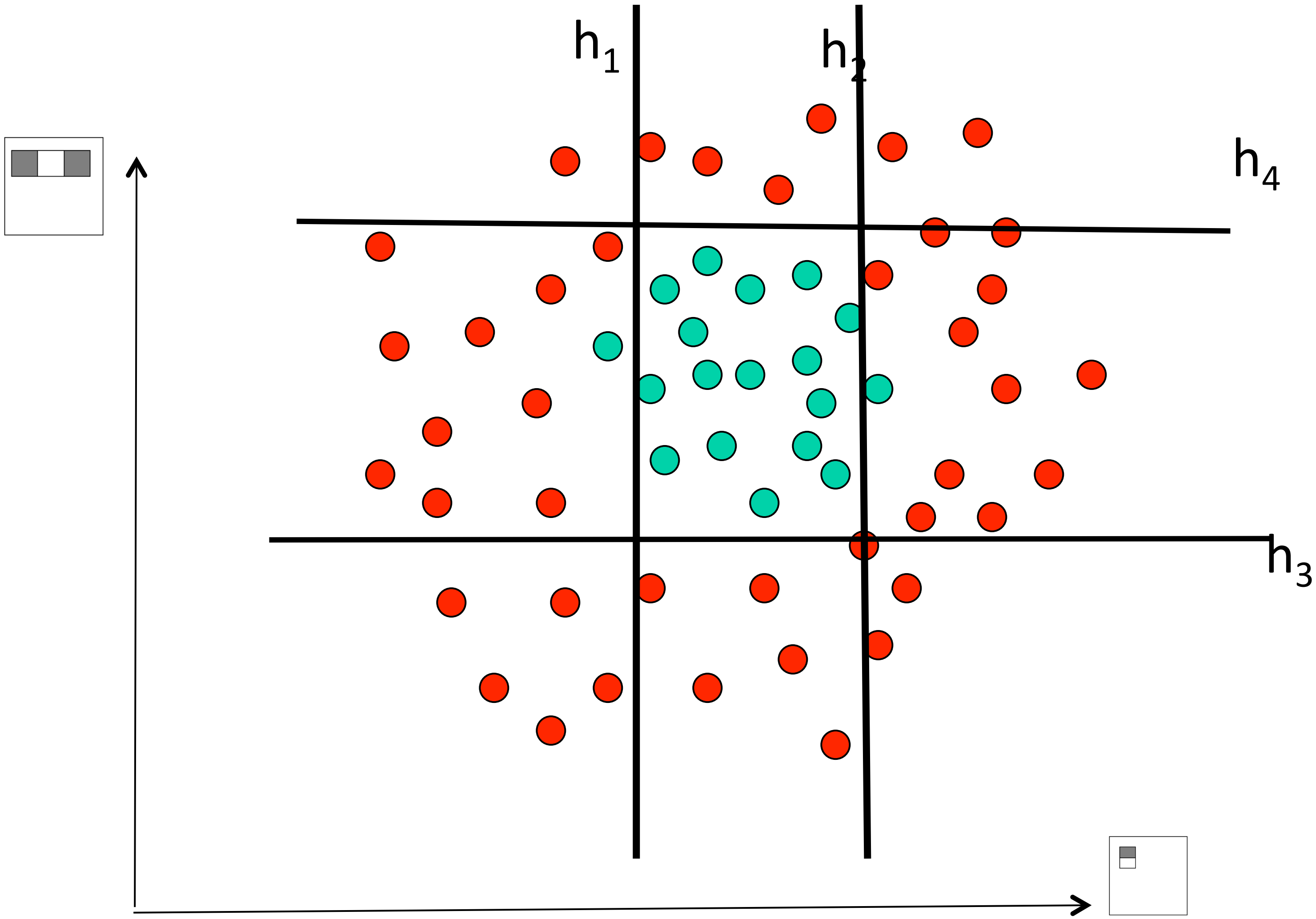
We consider the new problem and seek the best weak classifier.

AdaBoost



We consider the new problem and seek the best weak classifier.

AdaBoost

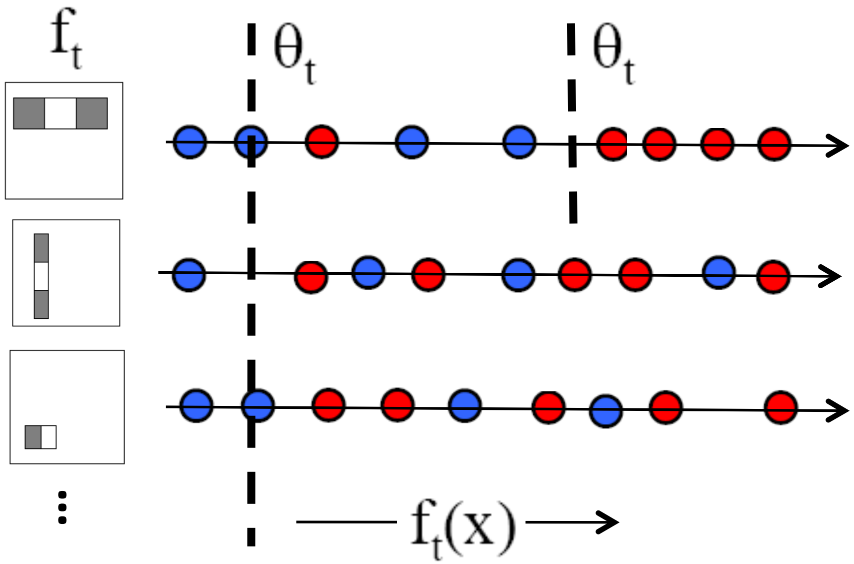


The strong (final) classifier C is composed as a sum of all weak classifiers. Once trained C(x), and given a new instance x', we compute where C(x') "falls”.

Viola-Jones detector: AdaBoost

WEAK CLASSIFIER

- Want to select the single rectangle feature and threshold that best separates positive (faces) and negative (non-faces) training examples, in terms of *weighted* error.



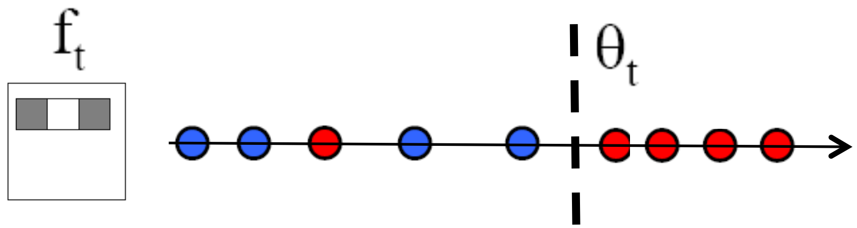
Outputs of a possible rectangle feature on faces and non-faces.

Resulting weak classifier:

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

The Adaboost algorithm

- **Step 1**: Initialize i=1.
- **Step 2**: If i==T (max number of iterations), go to Step 6, else go to Step 3:
- **Step 3**: Given the descriptors $f(x) = (f_1, f_2, \ldots f_t, \ldots)$ of a set of training objects (for which we know their class labels – face or not face), for each feature i look for the threshold Θ that separates the two classes with minimum error.



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

- **Step 4**: We choose the feature f_t (x) that separates with the smallest error the training examples.
  - The chosen classifier h_t (x) is called weak classifier, t is the index of the feature.

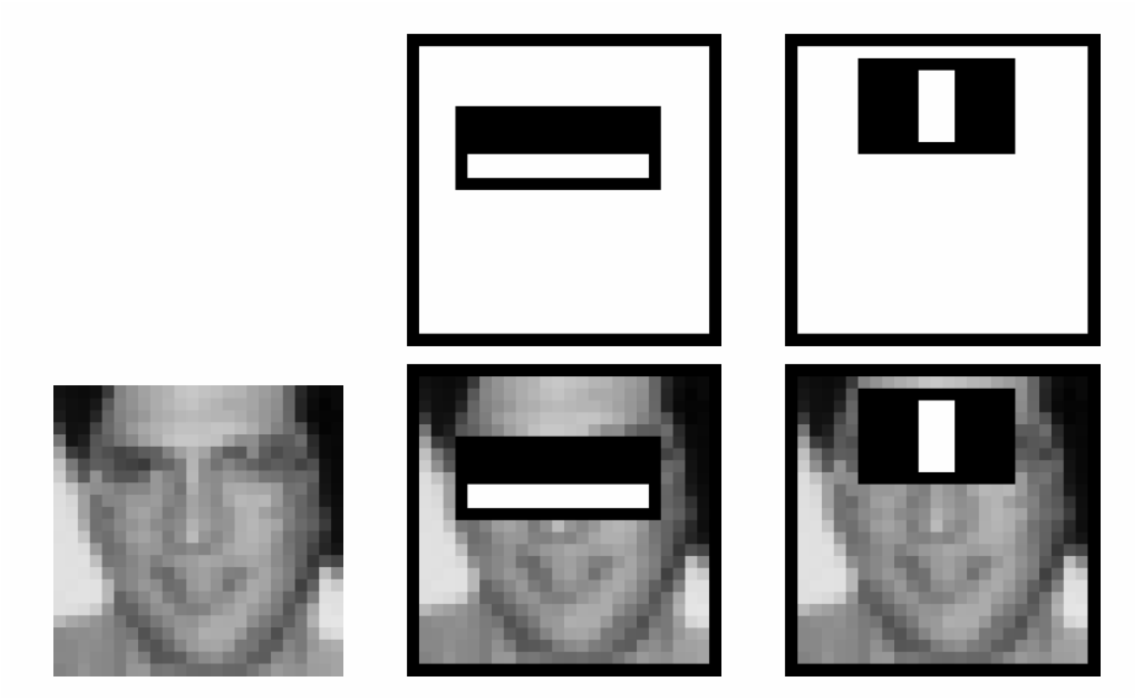
- **Step 5**: Sum the new single classifier h_t (x) which corrects the error of the sum of the previous classifiers, to the previous ones. Give more priority to the wrongly classified samples (faces and no faces).

- **Step 6:** The ultimate classifier is the sum of the weak classifiers added to each iteration:

$$F(x)=F(x)+h_t(x),\ t=1,2,\ldots T$$

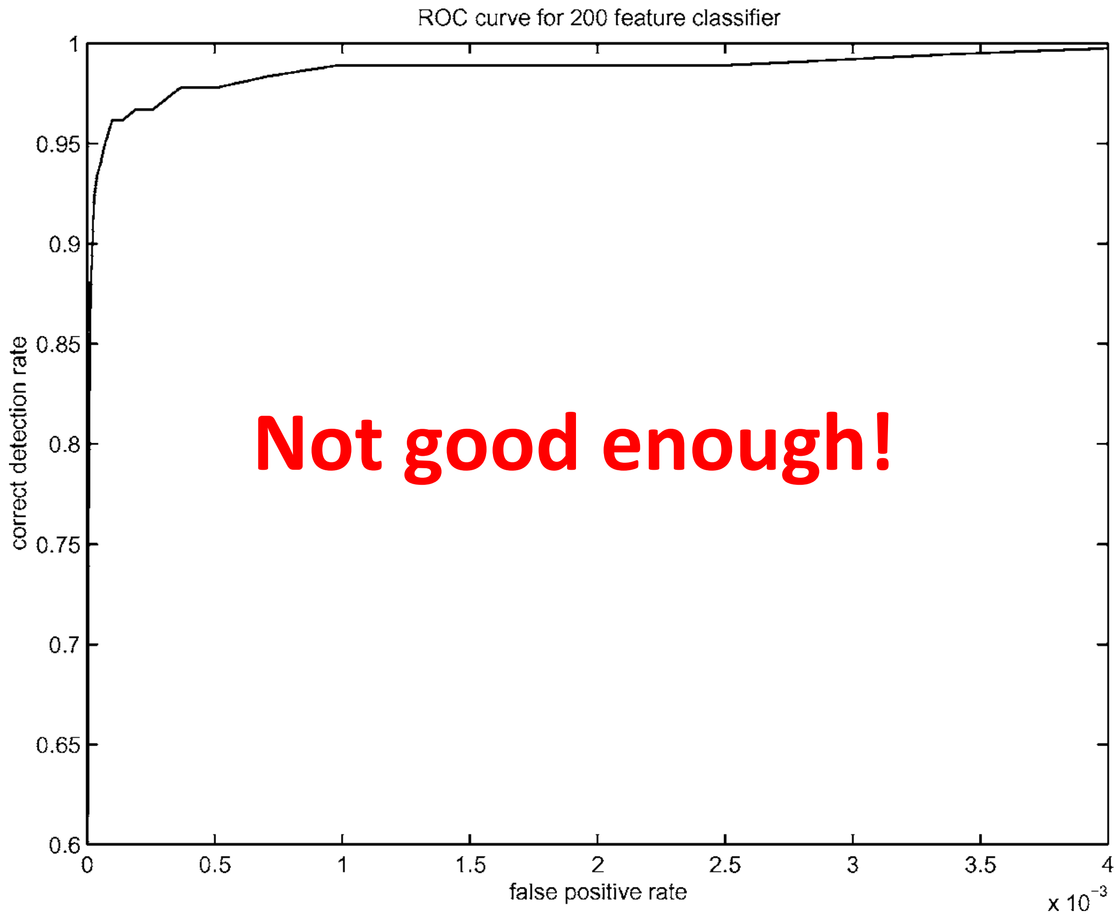
Boosting for face detection

- First two features selected by boosting:



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

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



- Even if the filters are fast to compute, each new image has a lot of possible windows to search.

- How to make the detection more efficient?

Face detection

1) Haar image features

2) Integral images

3) AdaBoost

4) **Cascade of classifiers**

Cascade of classifiers

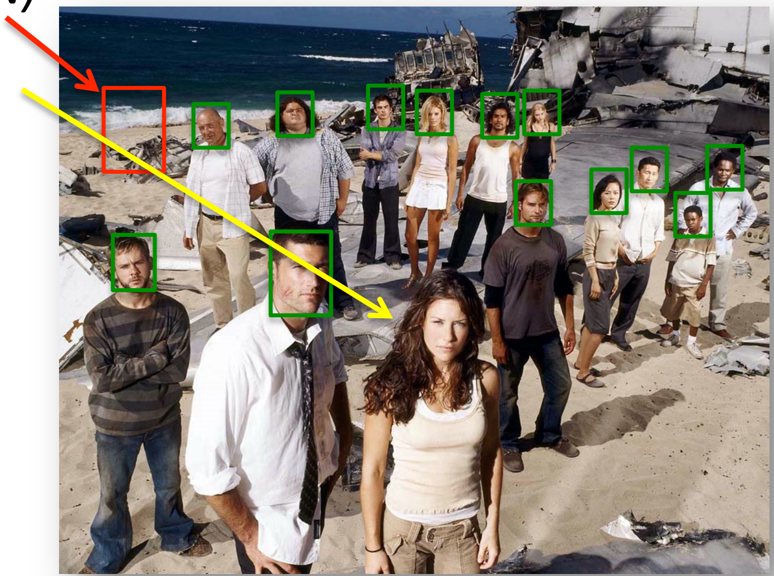
Now we have to design an appropriate classifier for the problem of face detection.

We recall that the main objective is to obtain an algorithm:

- Fast and
- Reliable.

In general, we need to accurately estimate the error detection by:

- *False Positives (FP)*
- *False Negatives (FN)*



True Positive Rate = TPr

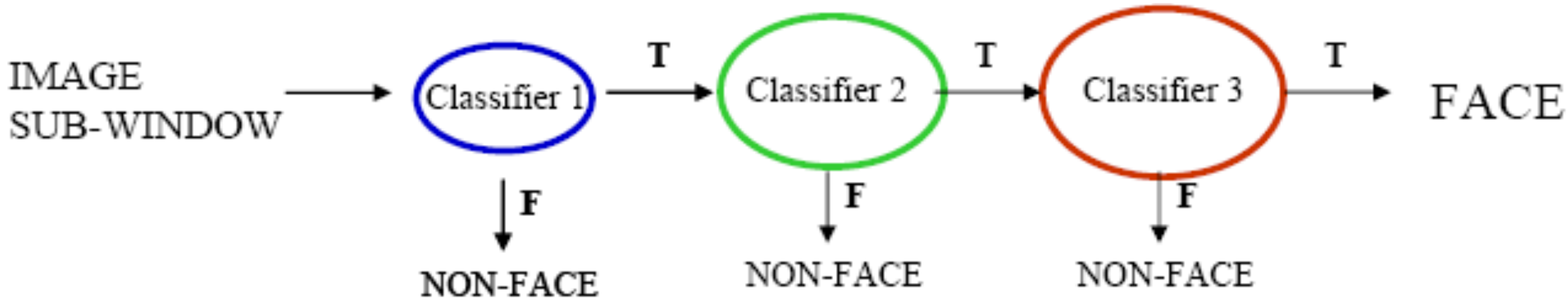
False Positive Rate = FPr

Cascade of classifiers

The idea of Viola & Jones is that each iteration of the cascade can accept a False Positive, but no False Negative, that is not to be missed any face!!!

What about the False Positives?

- The idea is that you can refine them with another classifier!
- Whence CASCADE of classifiers



Cascade of classifiers

CRITERIA TO DESIGN A CASCADE

1) Each classifier is an AdaBoost classifier.

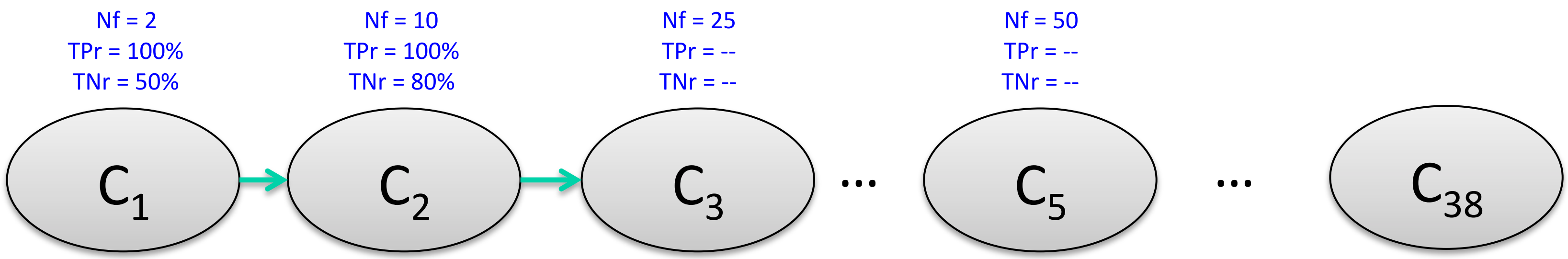
2) The first classifier C_1 is the simplest of all, and only classifies based on one feature.

3) The following classifiers are more complex and use more features to refine the results of previous classifiers.

4) Each subsequent classifier C_i is **trained with the error** of the classifier $C_{i\text{-}1}$.

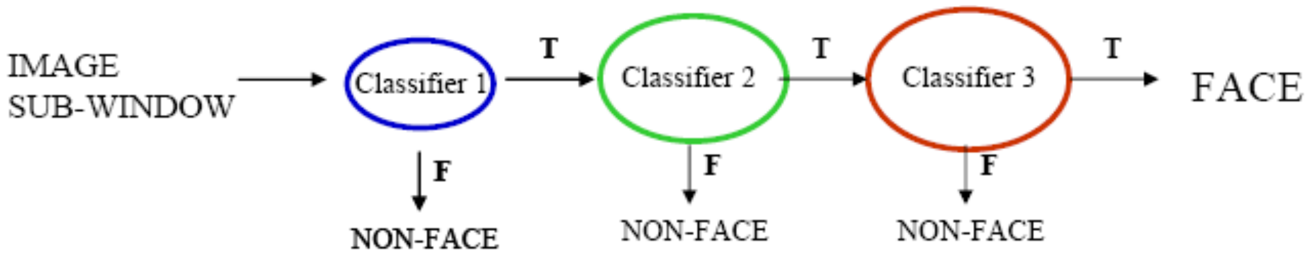
5) For each classifier C_i (during the training phase), we decide the FPr value we want to obtain, and add features until the decided value is not achieved! (N_f = number of features).

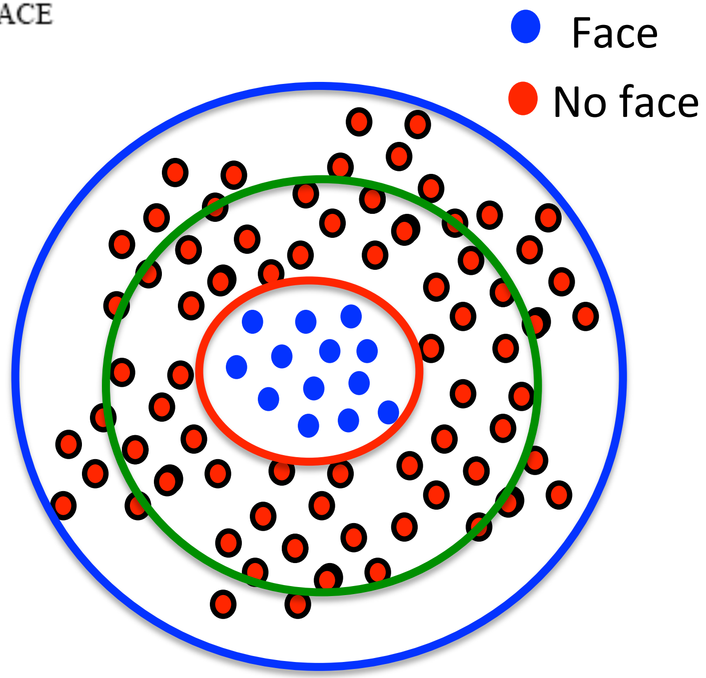
6) If FPr of C_i is not enough, we add another classifier, C_{i+1}.



Cascade of classifiers

¿How to interpret the cascade in the feature space?





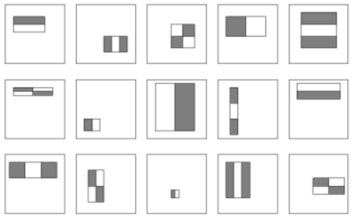
Each classifier in the cascade reduces the number of false negatives, keeping the number of successes (TP)!

Viola-Jones detector: summary

Faces

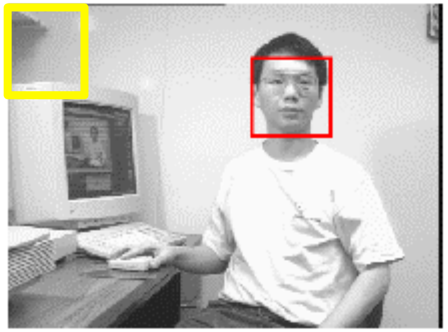
Non-faces

Train cascade of classifiers with AdaBoost



Selected features, thresholds, and weights

Apply to each subwindow



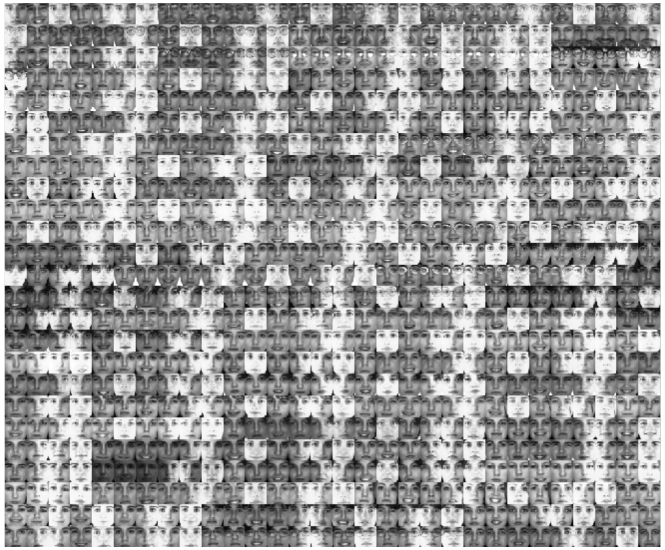
New image

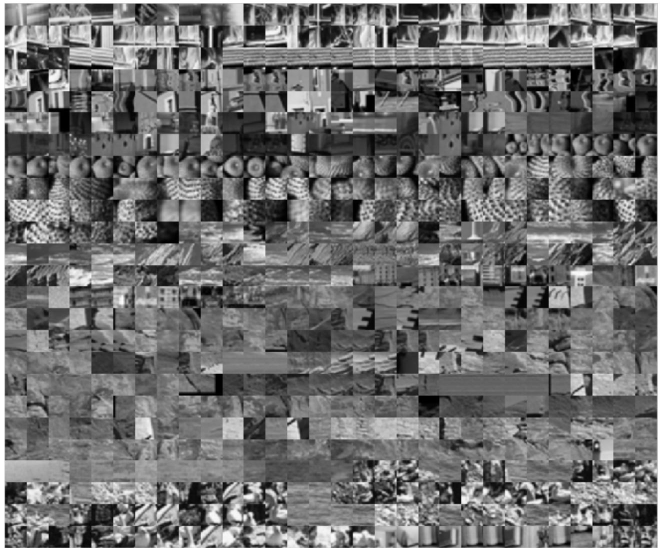
The classifier "learns" from the data

Training data

- 5000 faces (front) samples
- 108 no faces samples
- Many variations among individuals
- Different lighting, position, rotation

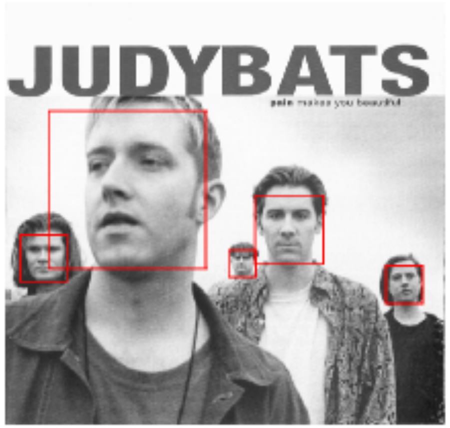
- Number of classifiers: 38.
- Number of features: 6060.
- Base Detector Resolution: 24x24 pixels
- Number of scales: 12.
- Scale factor for each scale: 1.25 (re-scaled features, not the image).
- Detection time: 0.067 seconds on a 384x288 px image on a 700MHz Pentium III.

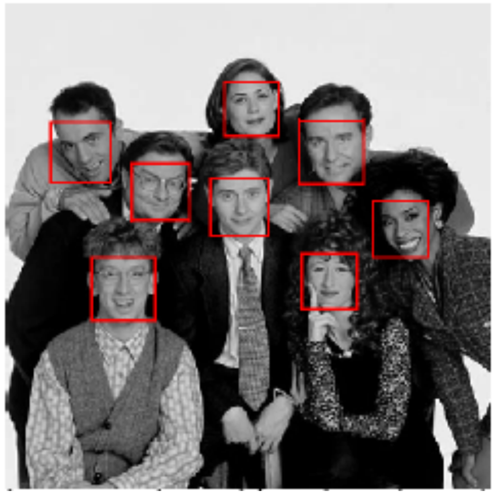


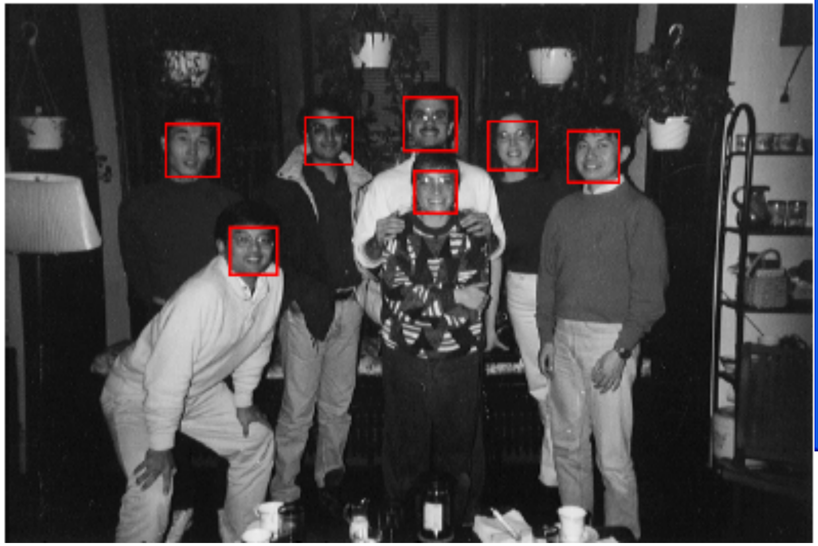


Face detection

RESULTS

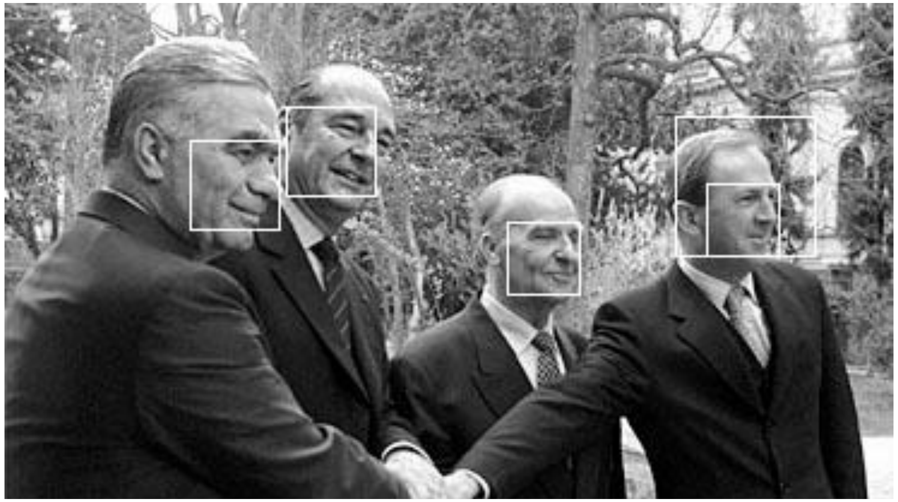






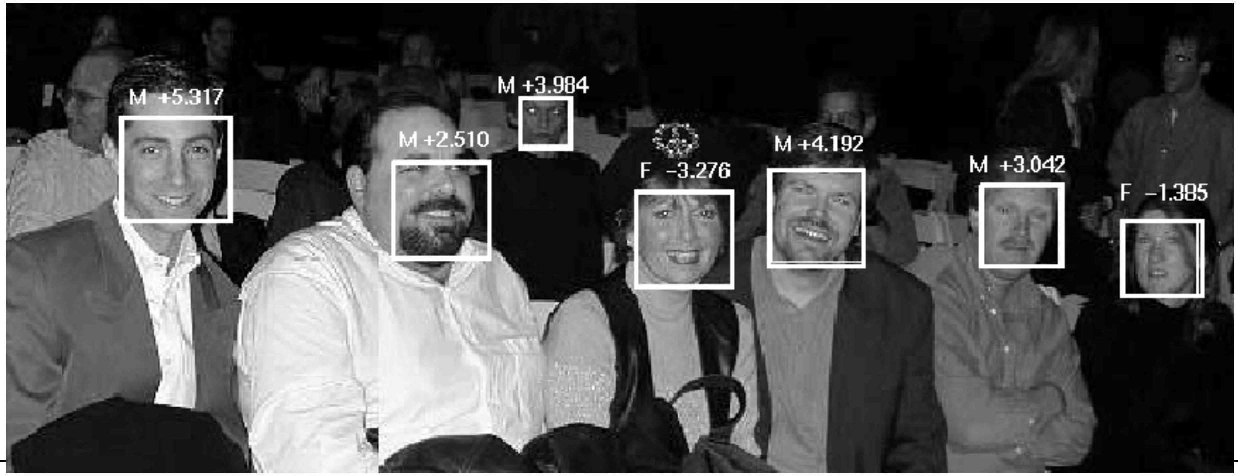
Other detection tasks

Facial Feature Localization

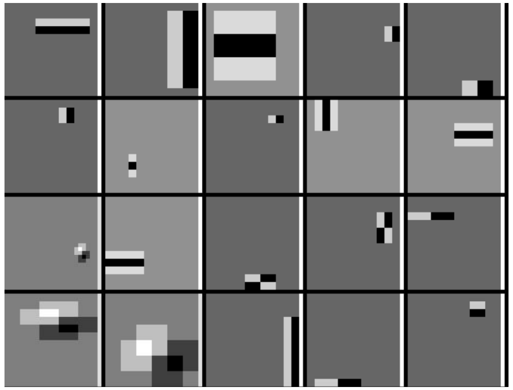


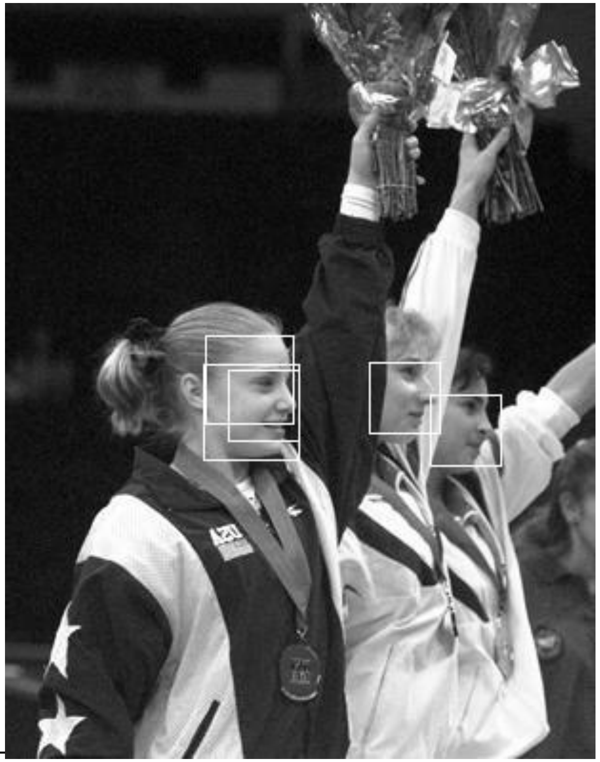
Profile Detection

Male vs. female



Profile Features

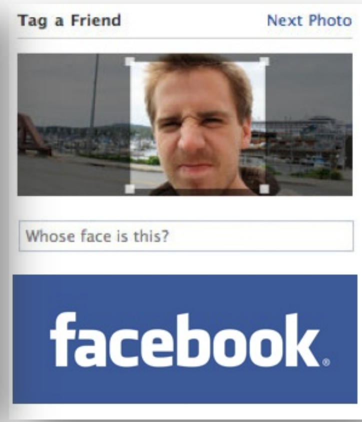




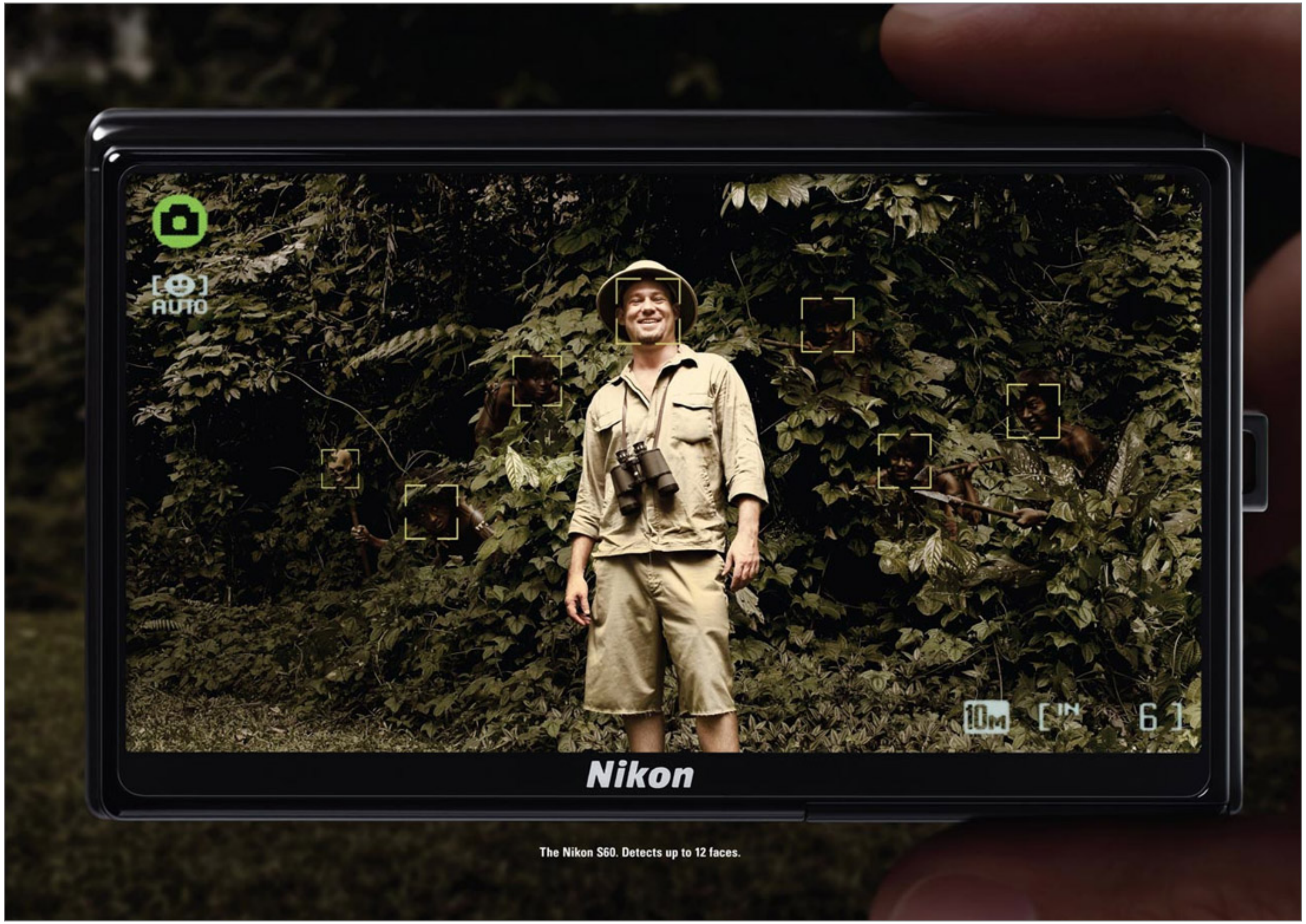
Face detection

APPLICATIONS

Auto-focus in digital cameras (+ smile detection).
Detection / Recognition of people (Video surveillance).
Data collection for cataloging / tagging photos (Facebook, iPhoto, etc.).



Funny Nikon ads



"The Nikon S60 detects up to 12 faces."

Consumer application: Apple iPhoto

Photos for Mac

Applications

Lexus LS600 Driver Monitor System

Google now erases faces, license plates on Map Street View

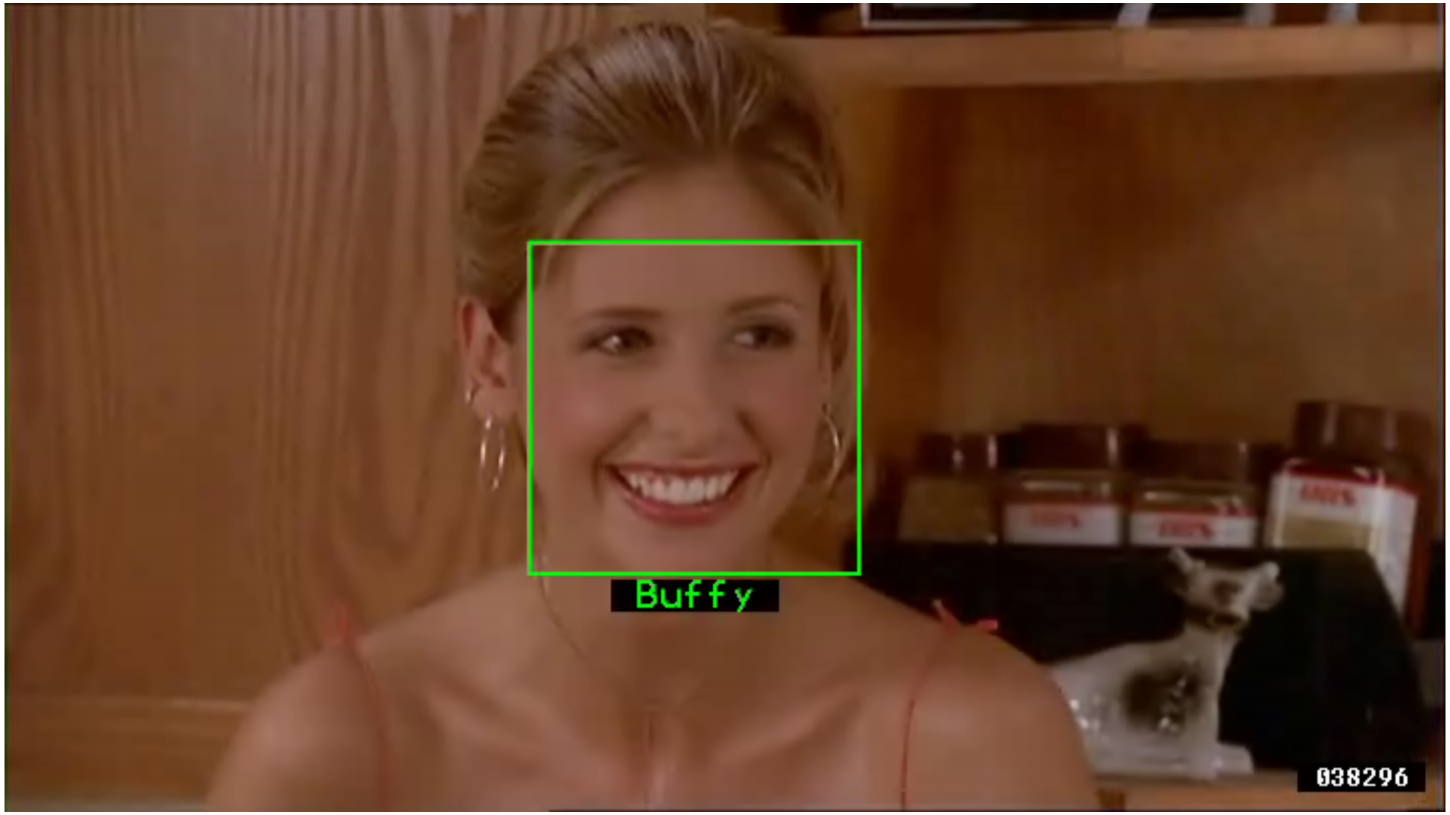
By Elinor Mills, CNET News.com
Friday, August 24, 2007 01:37 PM

Google has gotten a lot of flack from privacy advocates for photographing faces and license plate numbers and displaying them on the Street View in Google Maps. Originally, the company said only people who identified themselves could ask the company to remove their image.

But Google has quietly changed that policy, partly in response to criticism, and now anyone can alert the company and have an image of a license plate or a recognizable face removed, not just the owner of the face or car, says Marissa Mayer, vice president of search products and user experience at Google.

"It's a good policy for users and also clarifies the intent of the product," she said in an interview following her keynote at the Search Engine Strategies conference in San Jose, Calif., Wednesday.

The policy change was made about 10 days after the launch of the product in late May, but was not publicly announced, according to Mayer. The company is removing images only when someone notifies them and not proactively, she said. "It was definitely a big policy change inside."



Frontal faces detected and then tracked, character names inferred with alignment of script and subtitles.

Everingham, M., Sivic, J. and Zisserman, A., "Hello! My name is... Buffy" - Automatic naming of characters in TV video, BMVC 2006. http://www.robots.ox.ac.uk/~vgg/research/nface/index.html

Conclusions

- The Viola & Jones method is a method for automatic detection of faces in images.
- The Haar features provide an optimal description of the features of an image window that approximate the first and second derivatives of the image.
  - A huge set of features (... approx. 16M Features)
- Using integral images, the Haar features can be computed very quickly!
  - Efficient representation of the whole image with images
- The cascade of classifiers enables a very low rate of false negatives by detecting faces in real time!
  - Efficient selection of the features due to Adaboost.
- Fastest detector faces the literature.