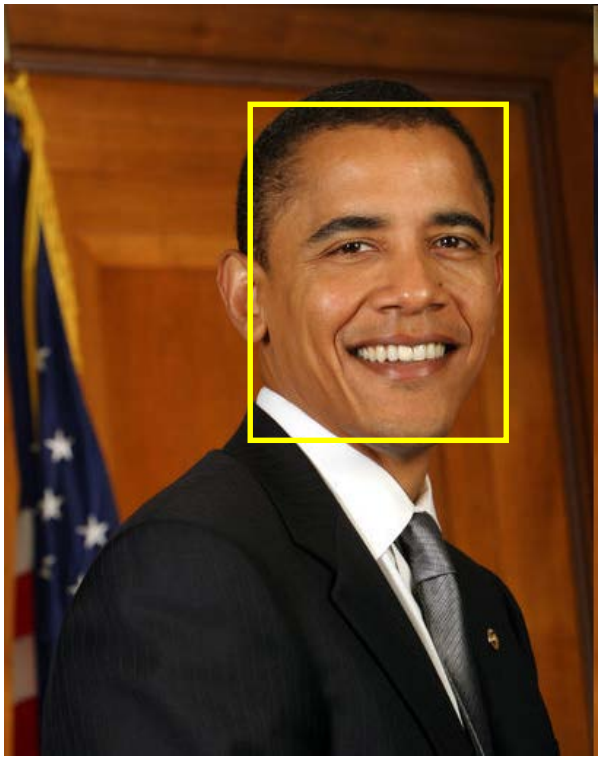
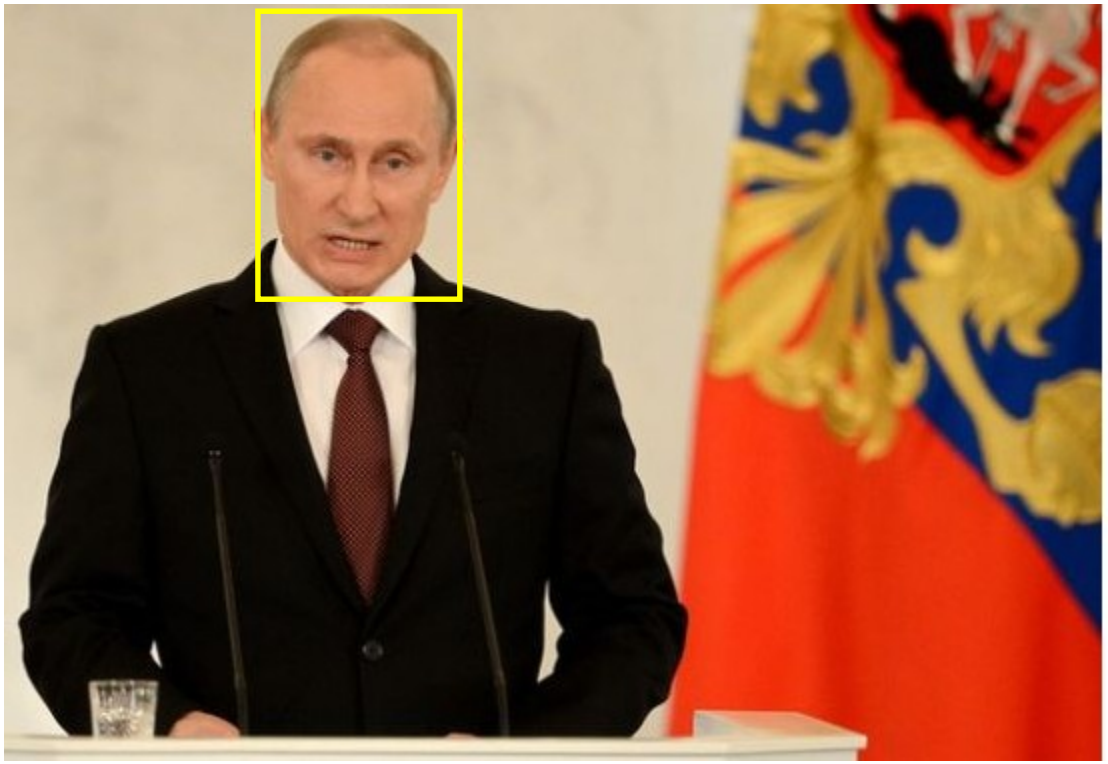
Face Detection



Barack Hussein Obama II, Aug. 04, 1961~



Vladimir Putin, Oct. 07, 1952~

Any faces contained in the image?

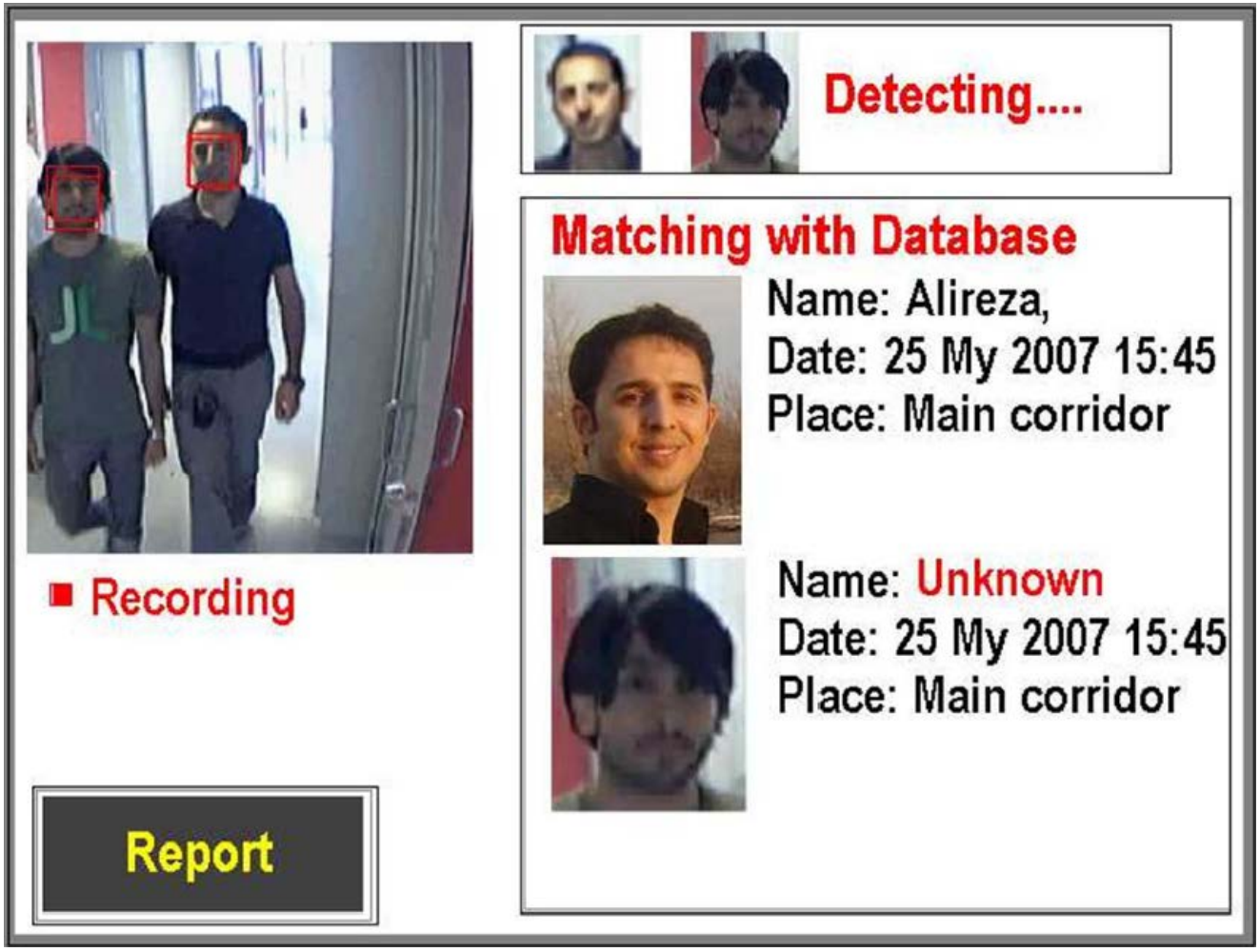
Who are they?

Outline

- Overview
- Face detection
  - Introduction
  - Viola-Jones method

Overview

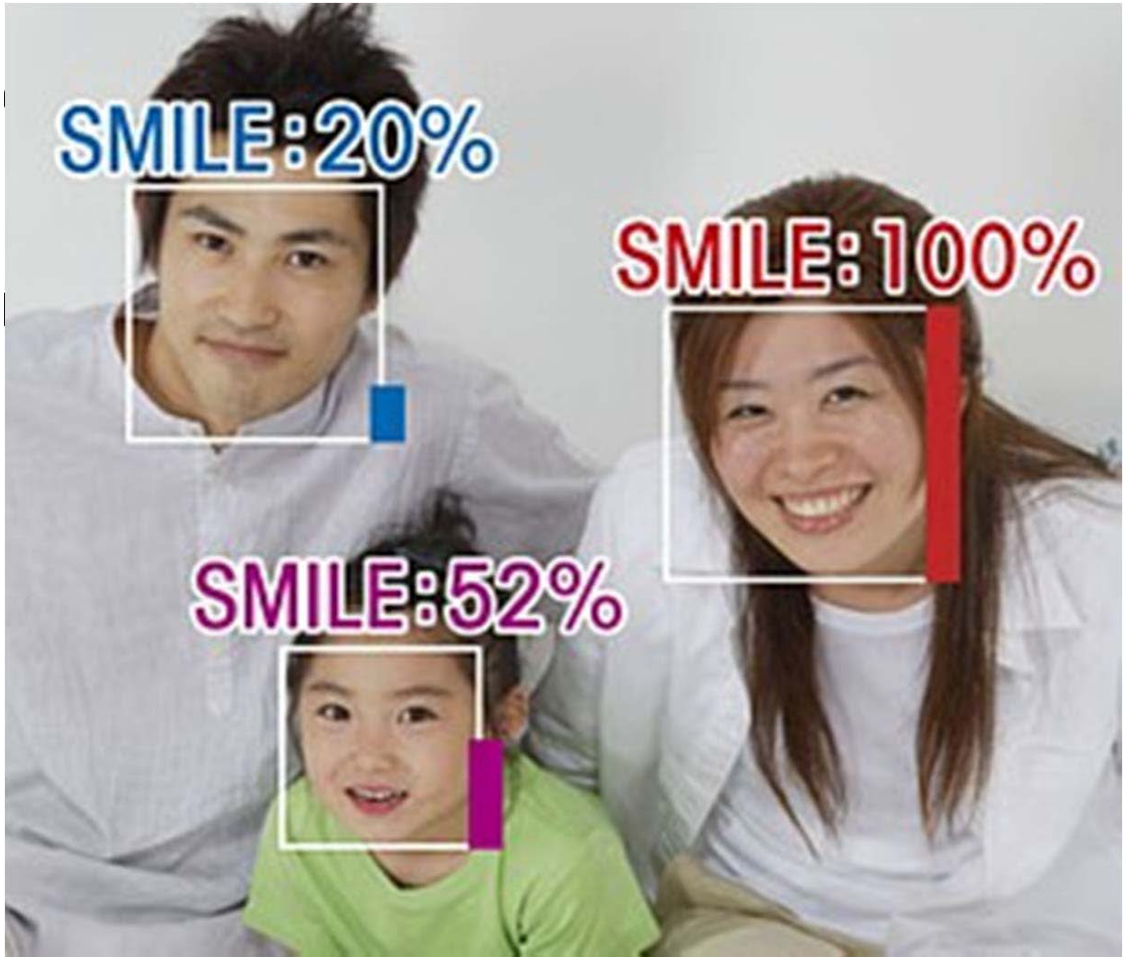
- Applications of face detection



Intelligent surveillance

Overview

- Applications of face detection



Smile detection: embedded in most modern cameras

Outline

- Overview
- Face detection
  - Introduction
  - AdaBoost
  - Viola-Jones method

Introduction

- Identify and locate human faces in an image regardless of their
  - Position
  - Scale
  - Orientation
  - pose (out of-plane rotation)
  - illumination

Introduction

Where are the faces, if any?

Introduction

- Why face detection is so difficult?

Introduction

- Appearance based methods
  - Train a classifier using positive (and usually negative) examples of faces
  - Representation: different appearance based methods may use different representation schemes
  - Most of the state-of-the-art methods belong to this category

The most successful one: Viola-Jones method!

VJ is based on AdaBoost classifier

Outline

- Overview
- Face detection
  - Introduction
  - AdaBoost
  - Viola-Jones method

AdaBoost (Adaptive Boosting)

- It is a machine learning algorithm[1]
- AdaBoost is adaptive in the sense that subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers
- The classifiers it uses can be weak, but as long as their performance is slightly better than random they will improve the final model

[1] Y. Freund and R.E. Schapire, "A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting", 1995

AdaBoost (Adaptive Boosting)

- AdaBoost is an algorithm for constructing a ”strong” classifier as a linear combination of simple weak classifiers,

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$$

- Terminology
  - $h_t(x)$ is a weak or basis classifier
  - $H(x)=\text{sgn}(f(x))$ is the final strong classifier

AdaBoost (algorithm for binary classification)

Given:

- Training set $(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$, where $y_i \in \{-1, +1\}$
- number of iterations T

Initialize weights for samples $D_1(i) = 1/m$

For $t = 1{:}T$

find $h_t = \arg\min_{h_j \in \mathcal{H}} \varepsilon_j, \varepsilon_j = \sum_{i=1}^{m} D_t(i)\left[h_j(x_i) \neq y_i\right]$

if $\varepsilon_t \geq 0.5$, stop;

set $\alpha_t = 0.5 \ln\left((1-\varepsilon_t)/\varepsilon_t\right)$

update weights for samples $D_{t+1}(i) = \dfrac{D_t(i)\exp\left(-\alpha_t y_i h_t(x_i)\right)}{Denom}$

Outputs the final classifier,

$$H(x) = \operatorname{sgn}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$

Outline

- Overview
- Face detection
  - Introduction
  - AdaBoost
  - Viola-Jones method

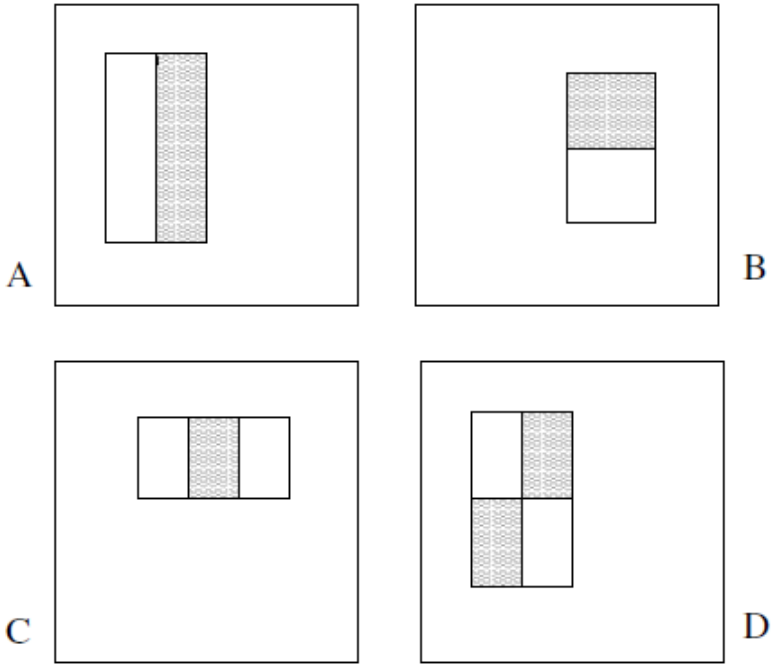
Viola-Jones face detection

- VJ face detector[1]
  - Harr-like features are proposed and computed based on *integral image*
  - A simple and efficient classifier is built by selecting a small number of important features by using AdaBoost
  - Classifiers are combined in a cascade structure which dramatically increases the detection speed

[1] P. Viola and M.J. Jones, “Robust real-time face detection", IJCV, 2004

Viola-Jones face detection

- Haar-like features
  - Compute the difference between the sums of pixels within two (or more) rectangular regions



Viola-Jones face detection

- Integral image
  - The integral image at location (x, y) contains the sum of all the pixels above and to the left of x, y, inclusive:

$$ii(x,y) = \sum_{x' \le x, y' \le y} i(x', y')$$

where $i(x, y)$ is the original image

  - By the following recurrence, the integral image can be computed in one pass over the original image

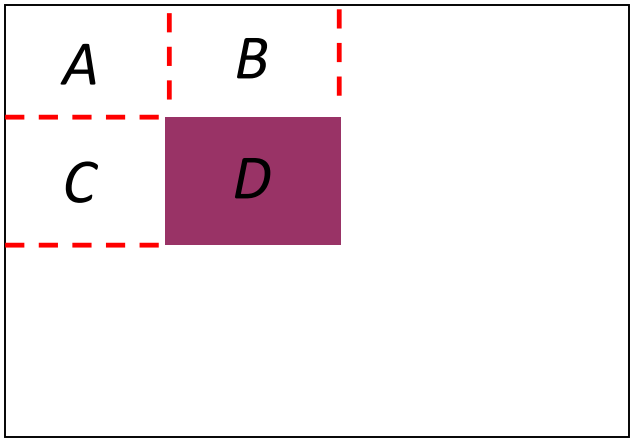
$$s(x,y) = s(x,y-1) + i(x,y)$$

$$ii(x,y) = ii(x-1,y) + s(x,y)$$

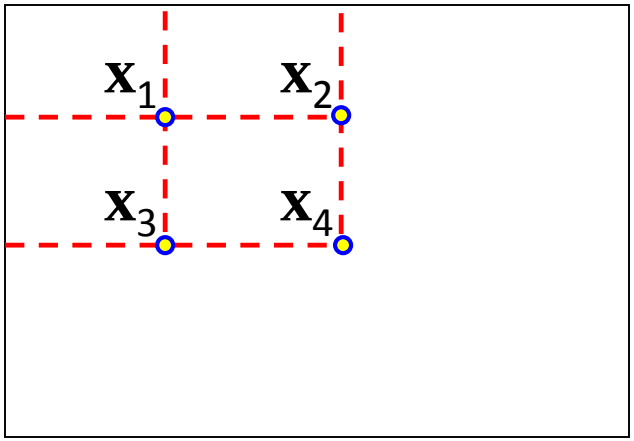
where $s(x, y)$ is the cumulative row sum, $s(x, -1) = 0$, and $ii(-1, y) = 0$

Viola-Jones face detection

- Haar-like feature can be efficiently computed by using integral image



original image $i(x, y)$



integral image $ii(x, y)$

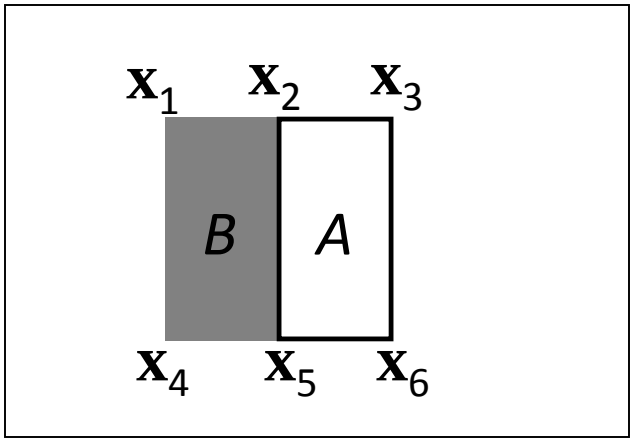
Actually,

$$ii(\mathbf{x}_1) = A$$
$$ii(\mathbf{x}_2) = A + B$$
$$ii(\mathbf{x}_3) = A + C$$
$$ii(\mathbf{x}_4) = A + B + C + D$$

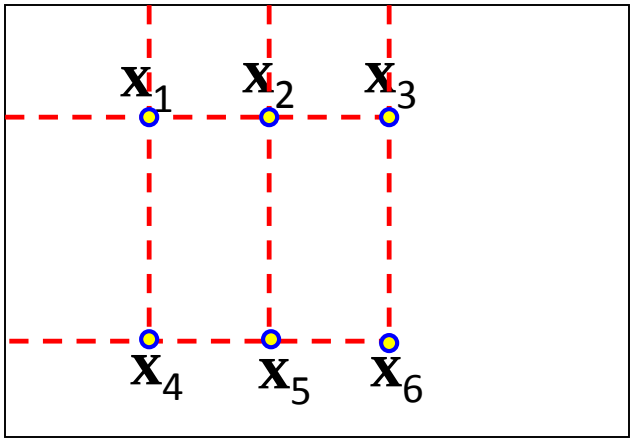
$$\Rightarrow \quad D = ii(\mathbf{x}_4) + ii(\mathbf{x}_1) - ii(\mathbf{x}_2) - ii(\mathbf{x}_3)$$

Viola-Jones face detection

- Haar-like feature can be efficiently computed by using integral image

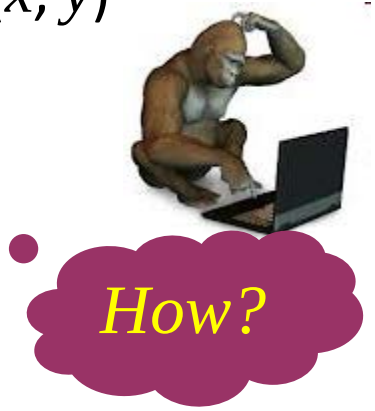


original image $i(x, y)$



integral image $ii(x, y)$

How to calculate A-B in integral image?

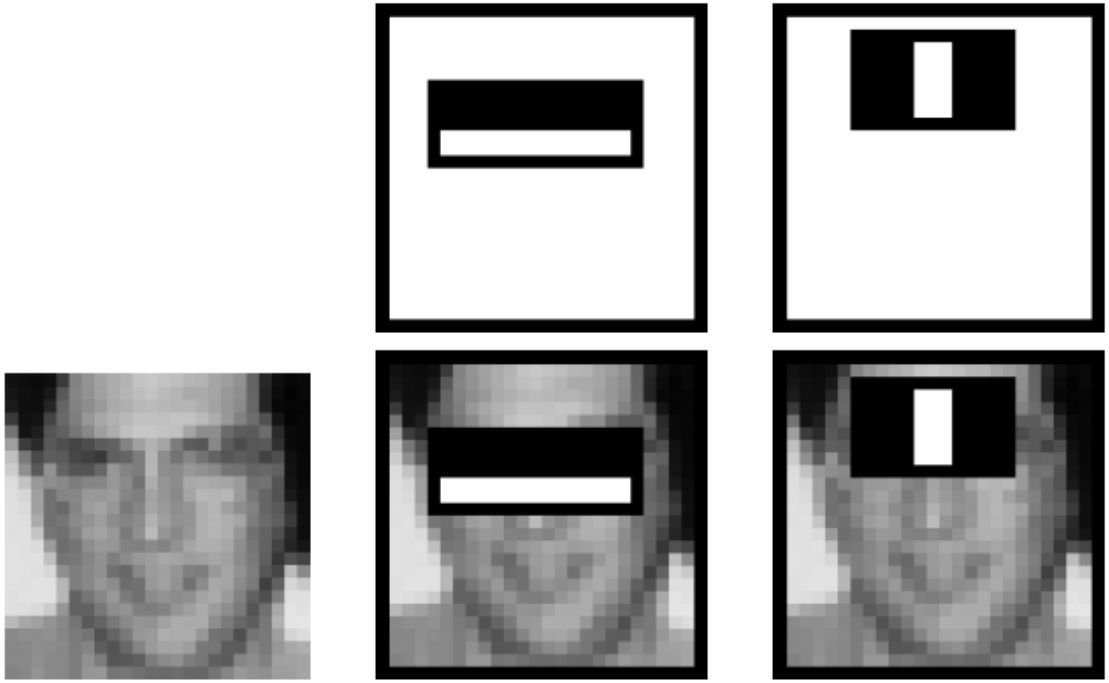


Viola-Jones face detection

- VJ face detector
  - Main idea
    - Feature selection: select important features
    - Focus of attention: focus on potential regions
    - Use an integral graph for fast feature evaluation
  - Use AdaBoost to learn
    - A set of important features (feature selection); sort them in the order of importance; each feature can be used as a simple (weak) classifier
    - A cascade of classifiers that combine all the weak classifiers to do a difficult task; filter out the regions that most likely do not contain faces

Viola-Jones face detection

- VJ face detector



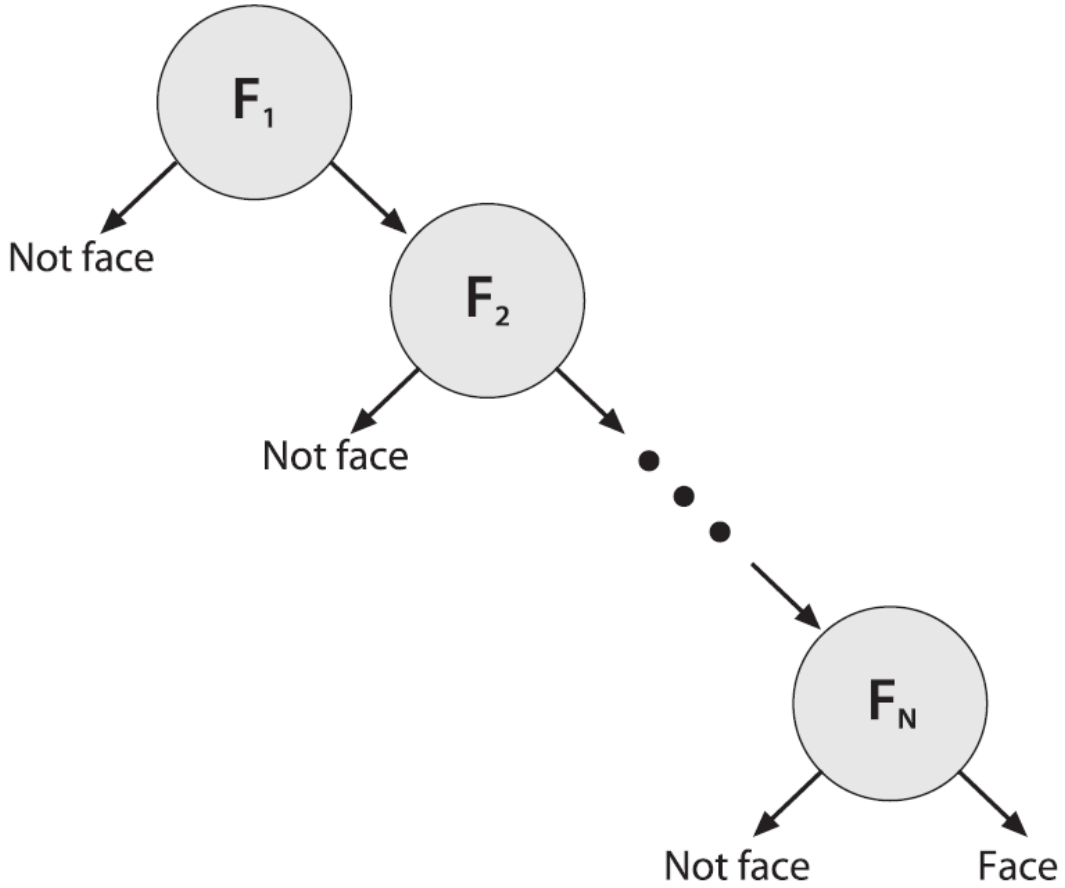
The first and second features selected by AdaBoost. The first feature measures the difference in intensity between the region of the eyes and a region across the upper cheeks. The feature capitalizes on the observation that the eye region is often darker than the cheeks. The second feature compares the intensities in the eye regions to the intensity across the bridge of the nose.

Viola-Jones face detection

- Rejection cascade
  - Within an image, most sub-images are non-face instances
  - Use smaller and efficient classifiers to reject many negative examples at early stage while detecting almost all the positive instances
  - Simpler classifiers are used to reject the majority of sub-windows
  - More complex classifiers are used at later stage to examine difficult cases
  - Learn the cascade classifier using Adaboost, i.e., learn an ensemble of weak classifiers

Viola-Jones face detection

- Rejection cascade



Rejection cascade: each node represents a multitree boosted classifier ensemble tuned to rarely miss a true face while rejecting a possibly small fraction of nonfaces

Viola-Jones face detection

- Implementation
  - VJ face detector has been implemented in OpenCV
  - OpenCV has also provided the training result from a frontal face dataset and the result is contained in “haarcascade_frontalface_alt2.xml”
  - A demo program has been provided on our course website: FaceDetectionEx

Viola-Jones face detection

- Demo time: some examples

Viola-Jones face detection

- Summary
  - Three main components
    - Integral image: efficient convolution
    - Use Adaboost for feature selection
    - Use Adaboost to learn the cascade classifier
  - Pros:
    - Fast and fairly robust; runs in real time
  - Cons:
    - Very time consuming in training stage (may take days in training)
    - Requires lots of engineering work