

Detección de Rostros con OpenCV

UTEM - Arquitectura de Computadores 2017

Detector de Rostros

Robust Real-Time Face Detection

PAUL VIOLA

Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA

viola@microsoft.com

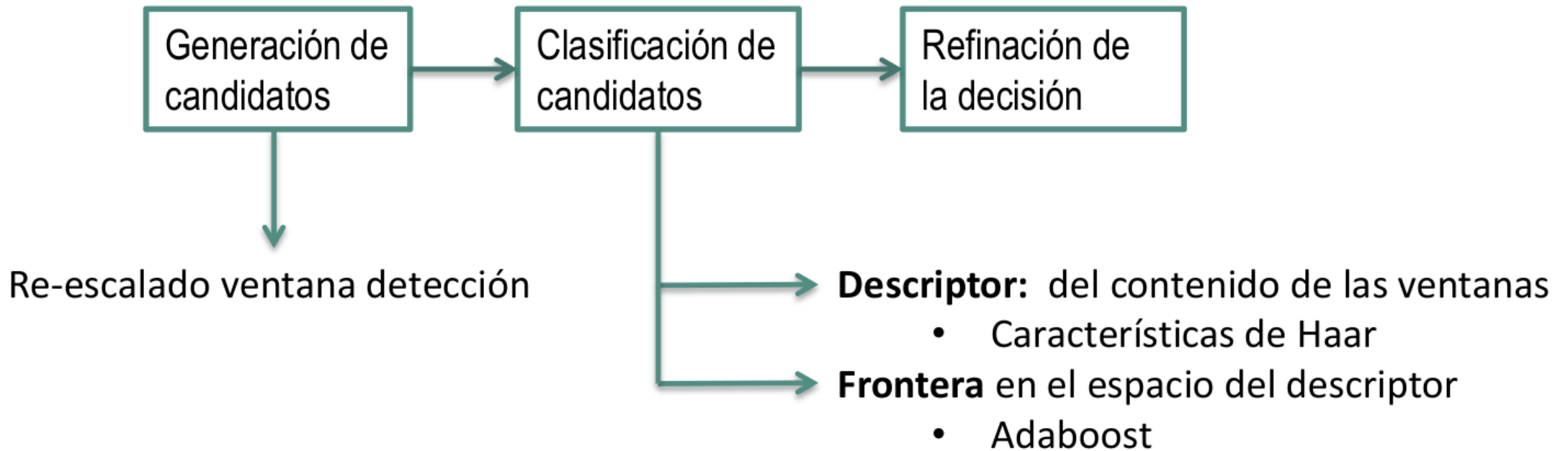
MICHAEL J. JONES

Mitsubishi Electric Research Laboratory, 201 Broadway, Cambridge, MA 02139, USA

mjjones@merl.com

Received September 10, 2001; Revised July 10, 2003; Accepted July 11, 2003

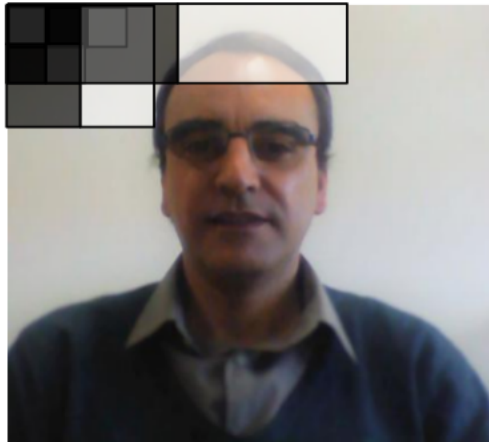
Abstract. This paper describes a face detection framework that is capable of processing images extremely rapidly while achieving high detection rates. There are three key contributions. The first is the introduction of a new image representation called the “Integral Image” which allows the features used by our detector to be computed very quickly. The second is a simple and efficient classifier which is built using the AdaBoost learning algorithm (Freund and Schapire, 1995) to select a small number of critical visual features from a very large set of potential features. The third contribution is a method for combining classifiers in a “cascade” which allows background regions of the image to be quickly discarded while spending more computation on promising face-like



Filtros de Haar

Descriptor

- Aplicación de una base de filtros (filtros de Haar) a diferentes escalas y en múltiples posiciones de la imagen
- Problema: miles características potenciales por imagen



Clasificación Haar

Contribuciones del método

- Cálculo rápido de características utilizando el concepto de imagen integral
- Clasificación basada en Adaboost : selección de características + aprendizaje
- Combinación de varios clasificadores en cascada, cada uno de ellos más restrictivo

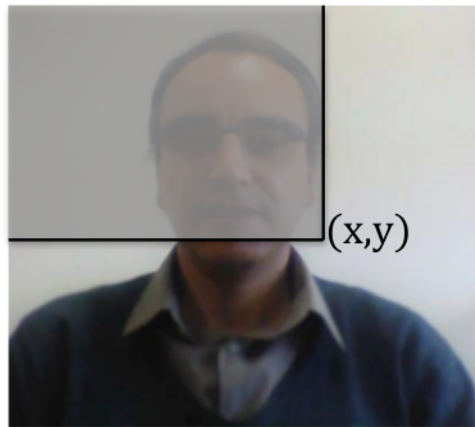
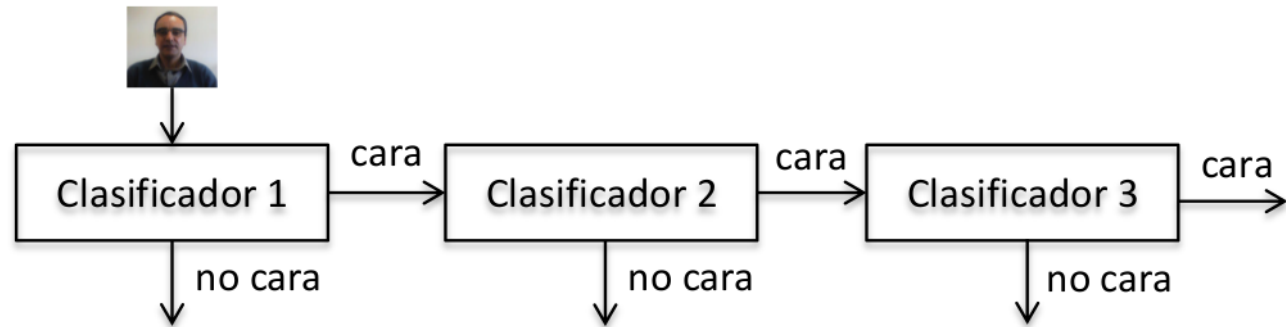


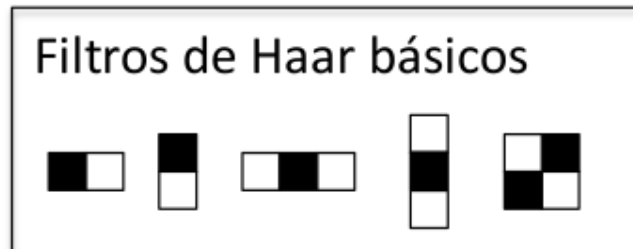
Imagen integral



Cascada de clasificadores

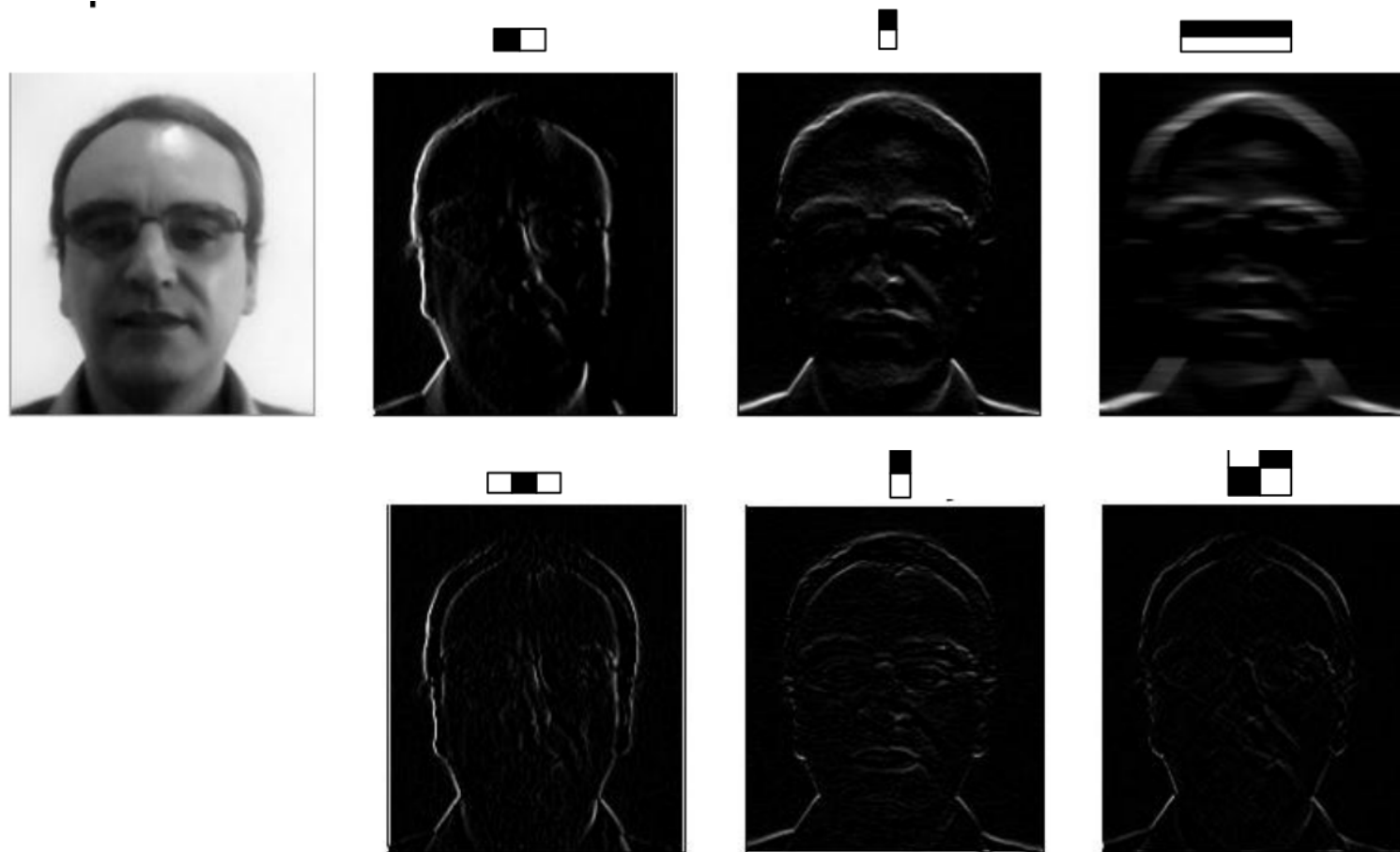
Filtros de Haar

- Combinación de rectángulos del mismo tamaño adyacentes horizontal o verticalmente
- Rectángulos en **negro** representan zonas con una **contribución positiva** al filtro
- Rectángulos en **blanco** representan zonas con una **contribución negativa** al filtro
- **Resultado del filtro:** diferencia en la suma de los valores de los píxeles entre zonas en negro y zonas en blanco



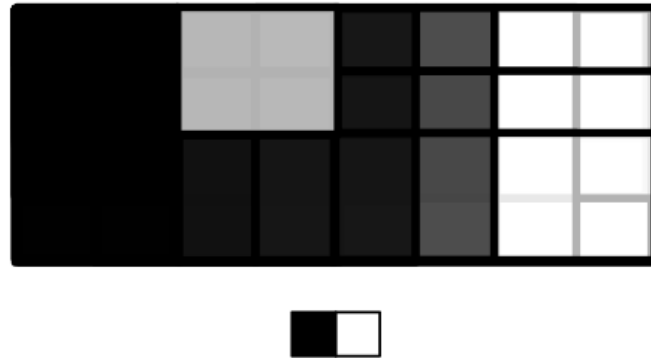
200	200	100	100	200	200	100	100
250	250	50	50	250	250	50	50
255	255	255	255	100	100	100	100
255	255	255	255	100	100	100	100
200	200	100	100	200	200	100	100
250	250	50	50	250	250	50	50
255	255	255	255	100	100	200	200
255	255	255	255	100	100	250	250

Interpretación de los filtros de Haar



Filtros de Haar a una imagen

- Cada filtro se aplica a todas las posibles escalas en horizontal y vertical
- Cada escala se aplica en todas las posibles posiciones de la imagen
- El resultado de aplicar de cada filtro en cada escala y posición es una característica de Haar
- Imágenes de 24x24 píxeles: 162336 características



Filtros Haar Extendidos

- Rotaciones de los filtros básicos que permiten detectar características direccionales
- Nuevas configuraciones de los filtros para detectar otros tipos de características

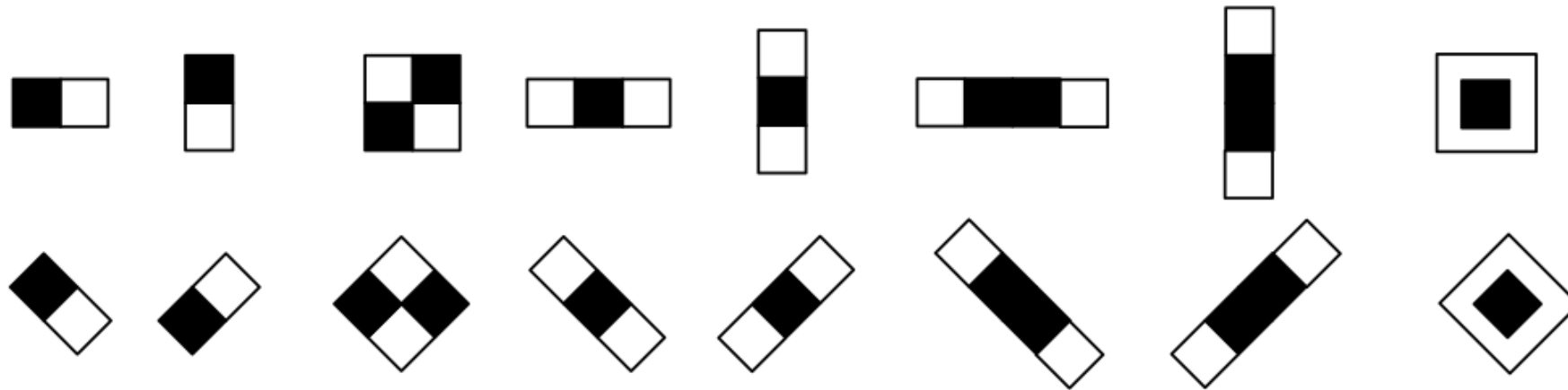
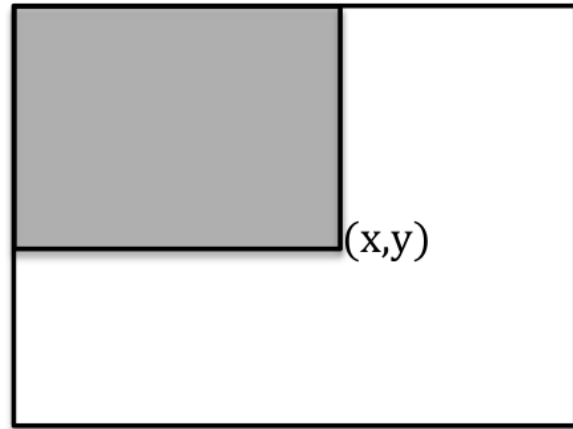


Imagen integral

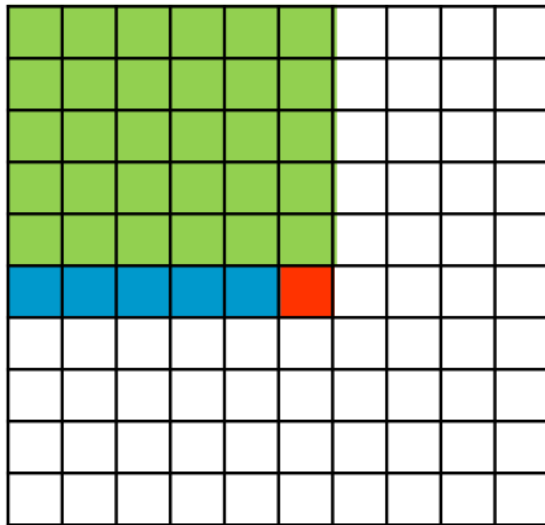
- Cada píxel de la imagen integral tiene un valor que corresponde a la suma de todos los píxeles situados a la izquierda y arriba en la imagen original
- Permite calcular muy rápidamente la suma de todos los píxeles de cualquier rectángulo en la imagen



$$I(x, y) = \sum_{\substack{x' \leq x \\ y' \leq y}} I(x', y')$$

Cálculo de la imagen integral

- Cálculo muy eficiente con sólo un recorrido por toda la imagen
- A partir de la suma acumulada de la fila actual y el valor de la imagen integral en la fila anterior



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

$$s(x, y) = \sum_{x' \leq x} I(x', y) = s(x - 1, y) + I(x, y)$$

I

5	0	10	20
10	10	5	0
0	0	5	20
15	10	10	0



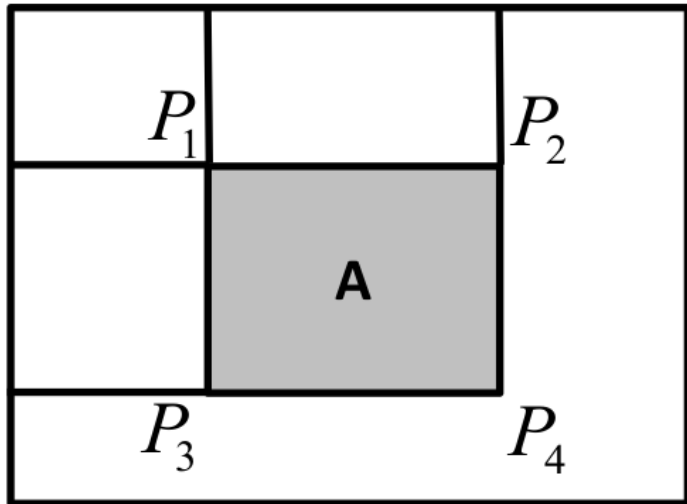
s

--	--	--	--

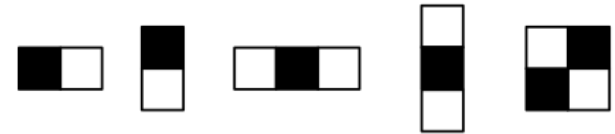
II

Utilización de la imagen integral

- Cálculo en tiempo constante de la suma de la intensidad de los píxeles de cualquier subventana de la imagen
- Sólo 3 sumas por rectángulo



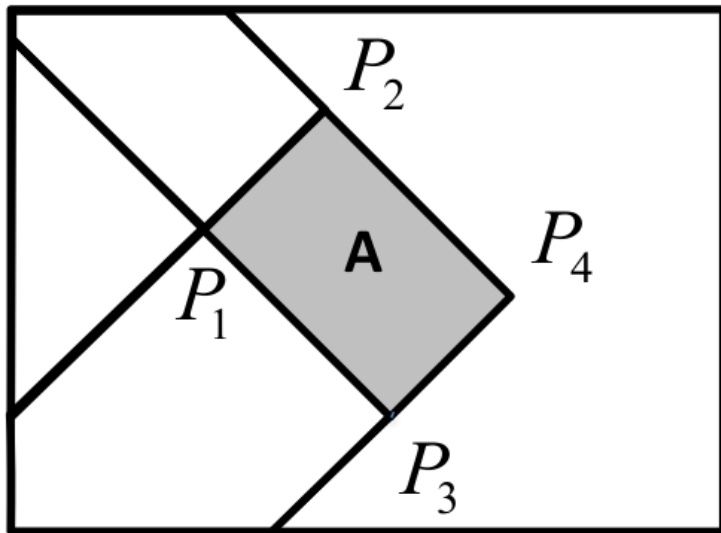
Filtros de Haar básicos



$$A = II(P_4) - II(P_2) - II(P_3) + II(P_1)$$

Filtros Haar Extendidos

- Imagen integral en 45°
- Calculo en dos pasos por imagen.



Primer paso (de arriba abajo y de izquierda a derecha)

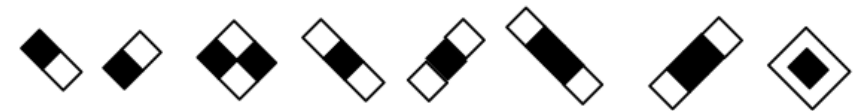
$$RII(x, y) = RII(x-1, y-1) + RII(x-1, y) - RII(x-2, y-1) + I(x, y)$$

Segundo paso (de abajo arriba y de derecha a izquierda)

$$RII(x, y) = RII(x, y) + RII(x-1, y+1) - RII(x-2, y)$$

$$A = II(P_4) - II(P_2) - II(P_3) + II(P_1)$$

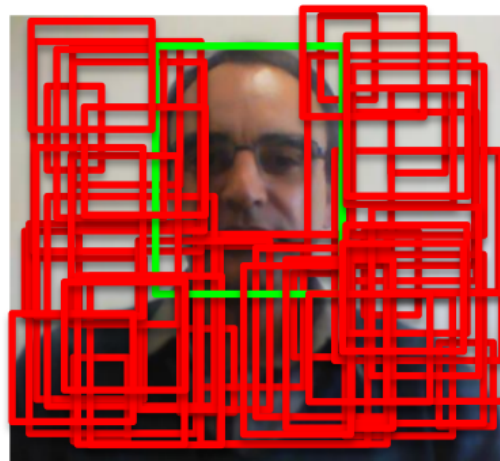
Filtros de Haar extendidos



Clasificadores de Cascada

Cascada de clasificadores: motivación

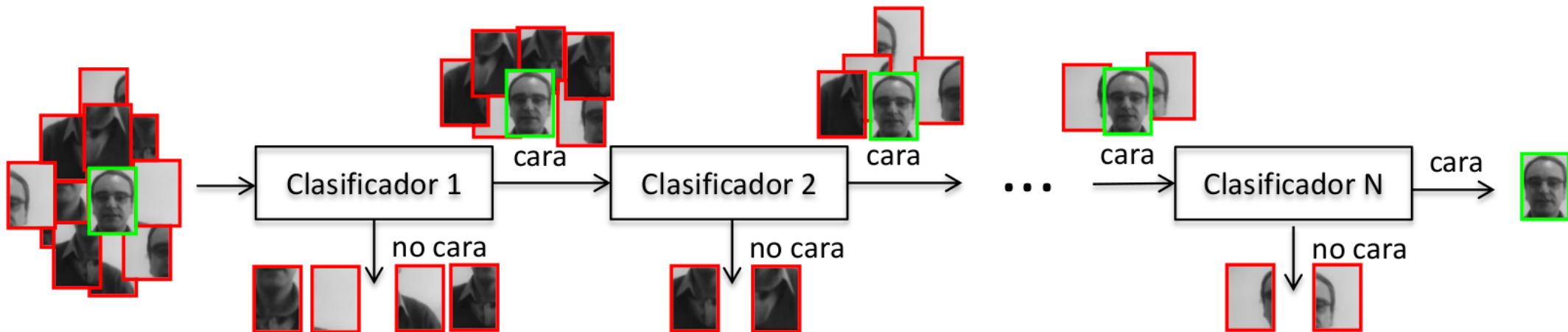
- La inmensa mayoría de las ventanas en una imagen no corresponden a una cara
- Objetivo: descartar muchas de ellas con el mínimo coste computacional posible (pocas características) y concentrar el esfuerzo en las que se corresponden a una cara



Esquema de clasificadores

Cascada de clasificadores: esquema

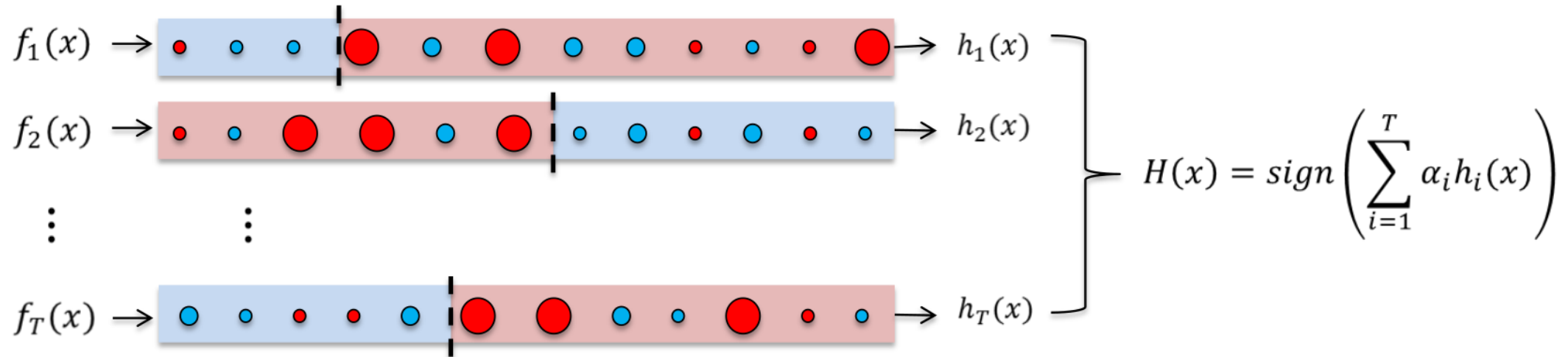
- Combinación secuencial de clasificadores
- Una imagen sólo es reconocida como cara si todos los clasificadores la aceptan
- Objetivo de cada nivel: alcanzar un índice de rendimiento (falsas detecciones vs. detecciones correctas) determinado con el mínimo número de características posibles



Aprendizaje Cascada

Aprendizaje de un nivel de la cascada

- Cada nivel de la cascada será un clasificador “fuerte” entrenado con Adaboost
- Adaptado para cumplir con un objetivo determinado de falsos positivos (falsas detecciones de caras) y falsos negativos (caras no detectadas)



Adaboost

- La idea inicial es determinar características basadas en sumas y restas de los niveles de intensidad de la imagen, a través de filtros de Haar de cierto tamaño y calculados para las posiciones concretas de la sub-imagen que se quiere clasificar. Estas características son evaluadas por un clasificador débil, para decidir si la sub-imagen podría corresponder a una cara o no; si este valor está por encima de cierto valor umbral, la ventana se clasificara como cara.

Adaboost

- ADABOOST es un sistema que construye una regla de clasificación final usando varios clasificadores menores, denominados “débiles” por su sencillez y escasa precisión. En solitario estos clasificadores débiles no constituirían un sistema de clasificación usable debido a su alta inexactitud, pero al usarlos en conjunto es posible construir un clasificador muchísimo más preciso. Los clasificadores débiles corresponden a reglas de clasificación simples y que entregan como resultado un valor de confianza, o confidencia, respecto a la predicción que están haciendo.

Aprendizaje Global

Aprendizaje global de la cascada

Objetivo de FP y FN
Nº máximo características

repetir

Entrenar un clasificador fuerte

Adaptar margen al objetivo de FP y FN

si *objetivo no cumplido*

Incrementar número de características

hasta *objetivo cumplido o número máximo características*



Clasificador 1

Clasificador 2

...

Clasificador N

Buscar nuevos ejemplos negativos
Falsos positivos del paso anterior

Nº determinado de niveles
Objetivo de rendimiento