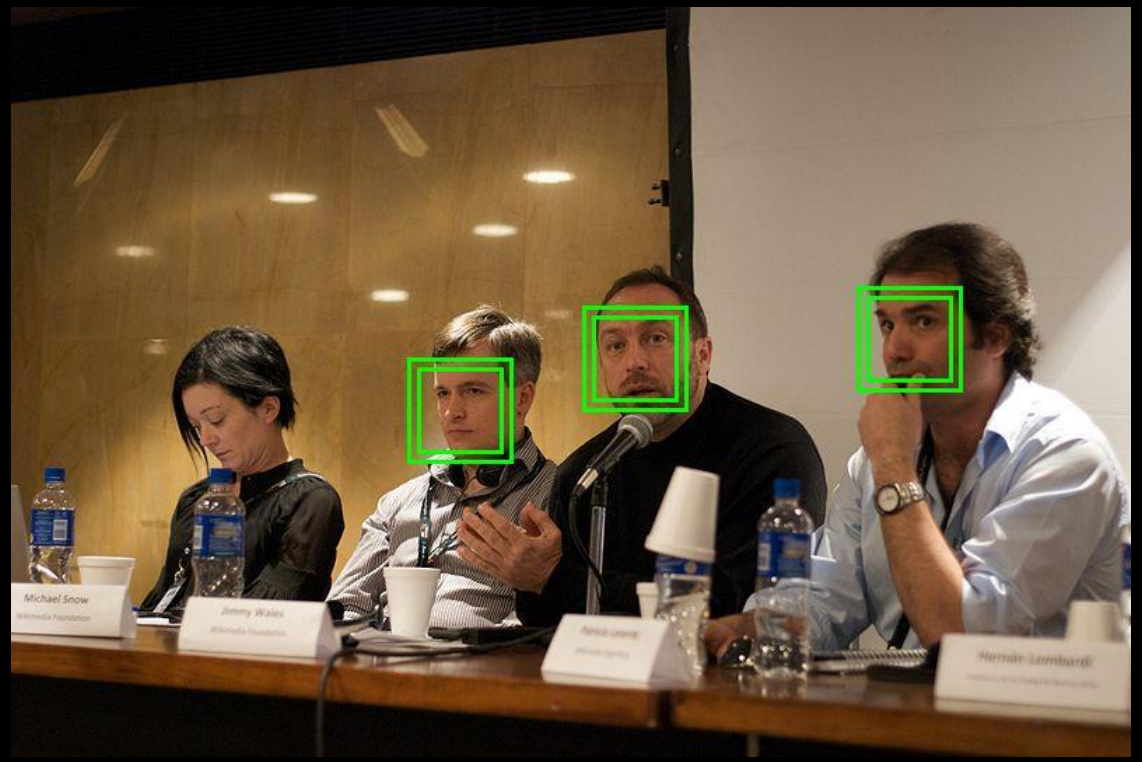
FACE DETECTION

Human face detection



Applications

- Automatic selection of camera settings
- Personalized advertising campaigns
- User Authentication
- Law Enforcement
- Healthcare

Human face detection

- Haar features for face detection
- Integral image
- Nearest Neighbour classifier
- Support Vector Machine

Face detection process

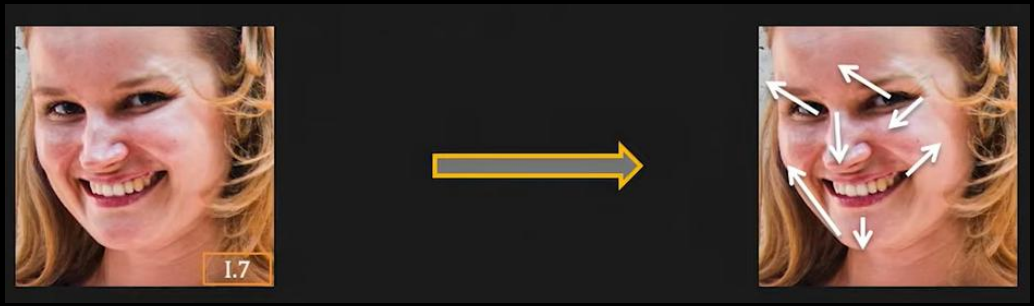
Face detection framework

- For each window:

- Features?
- Classifier?

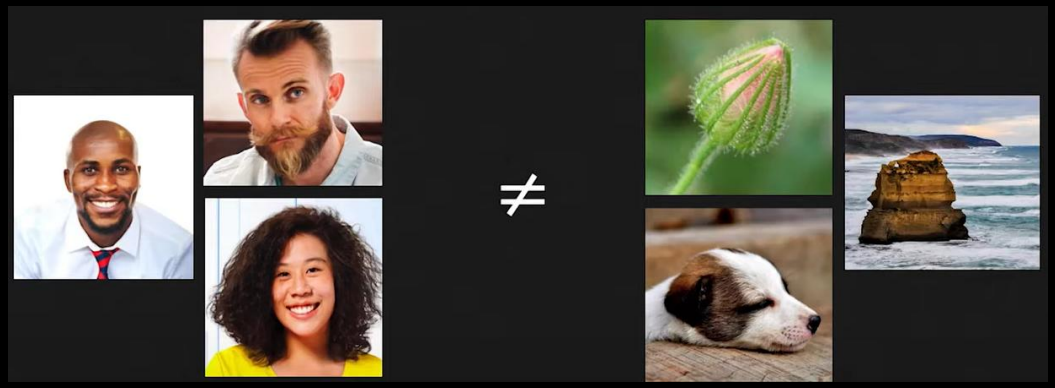
What are Good Features?

- Interest points (Edges, Corners, SIFT)?



- Facial Components (Templates)?

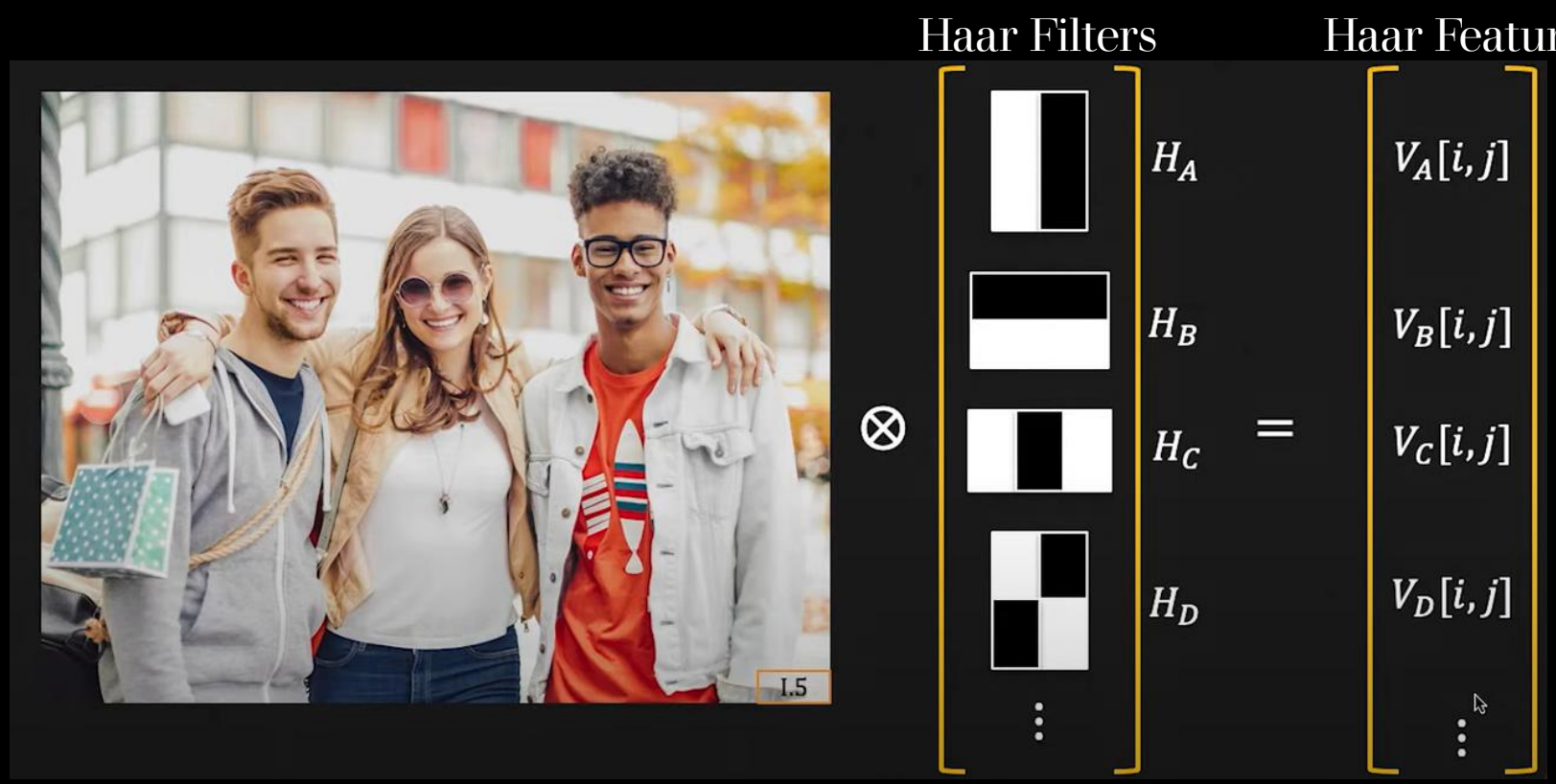
Characteristics of Good features



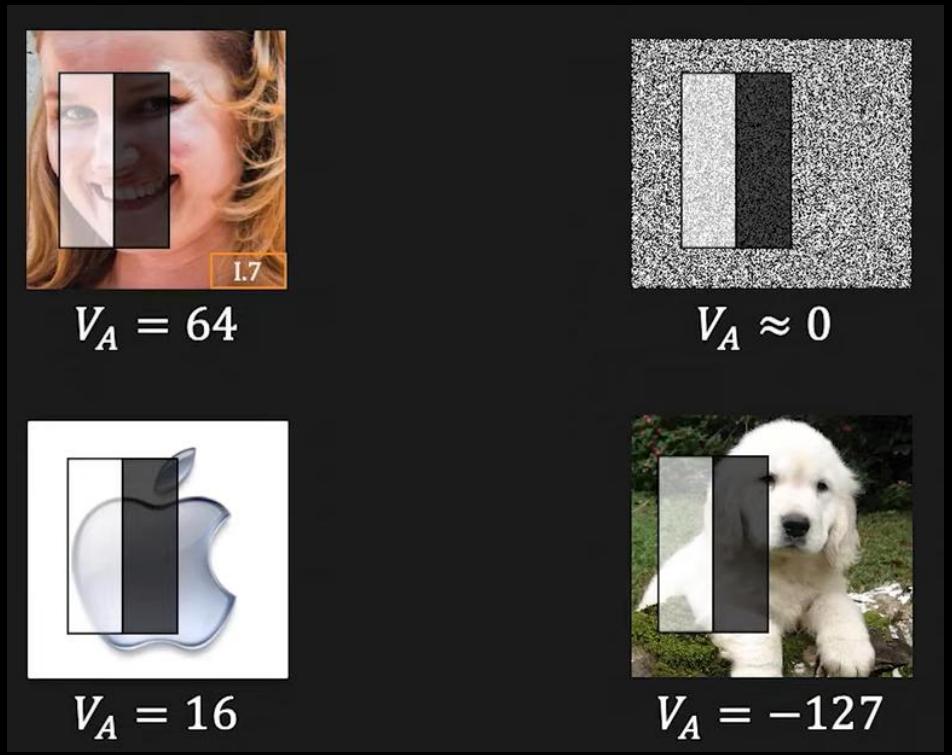
- Discriminate Face/Non-face

- Extremely fast to compute

Haar features

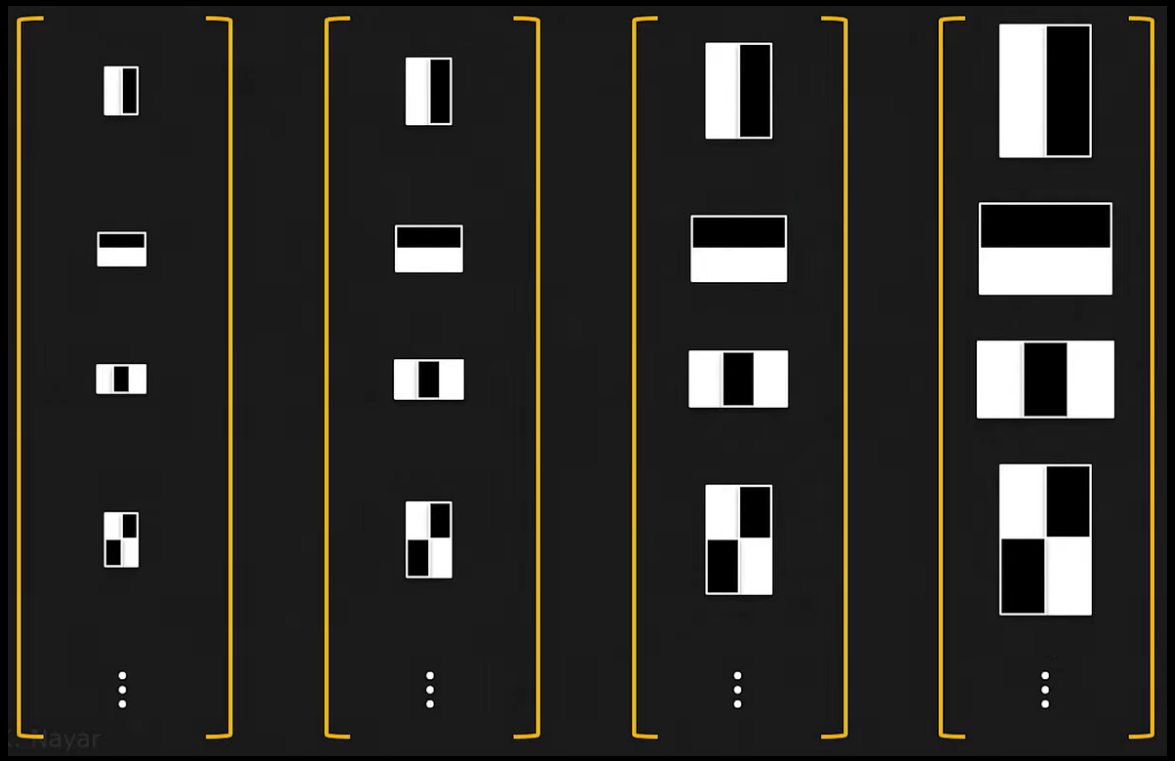


Discriminative ability of Haar features



- Sensitive to Directionality of patterns

Detecting faces of different size



Computing Haar features

$\otimes$  H_A

White = 1, Black = -1

Response to Filter H_A at location (i, j):

$$V_A[i,j] = \sum_m \sum_n I[m-i, n-j] H_A[m,n]$$

$V_A[i,j] =$ $\sum$ (pixel intensities in white area)

$-$ $\sum$ (pixels intensities in black area)

Haar feature: Computation cost

- Value = Σ(pixel intensities in white) – Σ(pixels intensities in black)

- Computation cost = (N*M – 1) additions per pixel per filter per scale

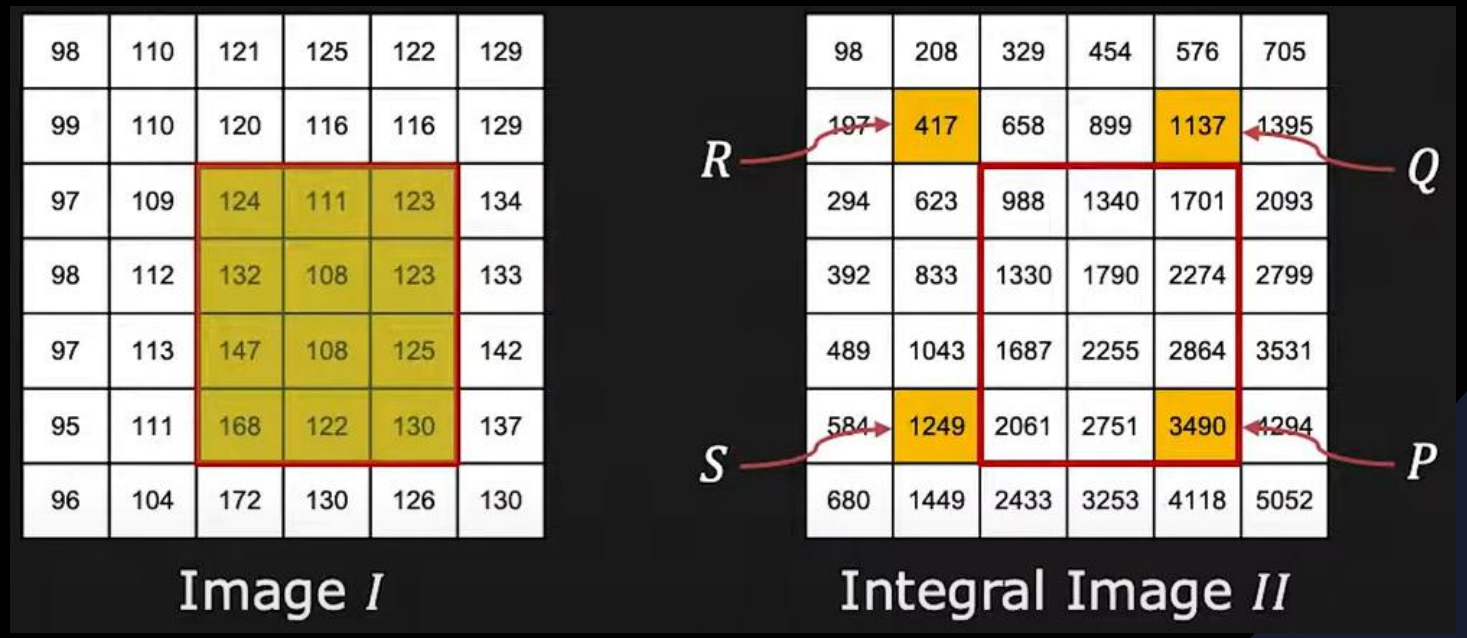
Integral image

- A table that holds sum of all pixel values to the left and top of a given pixel including itself

Summation within a Rectangle

$$\text{Sum} = II_P - II_Q - II_S + II_R$$
$$= 3490 - 1137 - 1249 + 417$$
$$= 1521$$

Computation Cost: Only 3 additions



Haar response using Integral image

Value = Σ(pixel intensities in white) – Σ(pixels intensities in black)

$= (II_O - II_T + II_R - II_S) - (II_P - II_Q + II_T - II_O)$

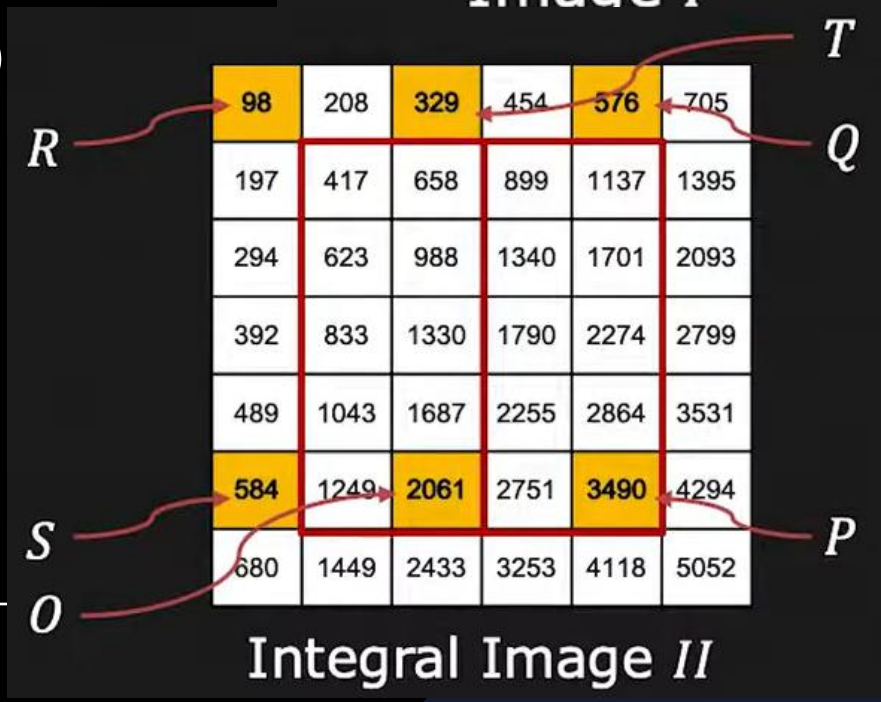
$= (2061 - 329 + 98 - 584) - (3490 - 576 + 329 - 2061)$

$= 64$

Computation cost: Only 7 additions

98	110	121	125	122	129
99	110	120	116	116	129
97	109	124	111	123	134
98	112	132	108	123	133
97	113	147	108	125	142
95	111	168	122	130	137
96	104	172	130	126	130

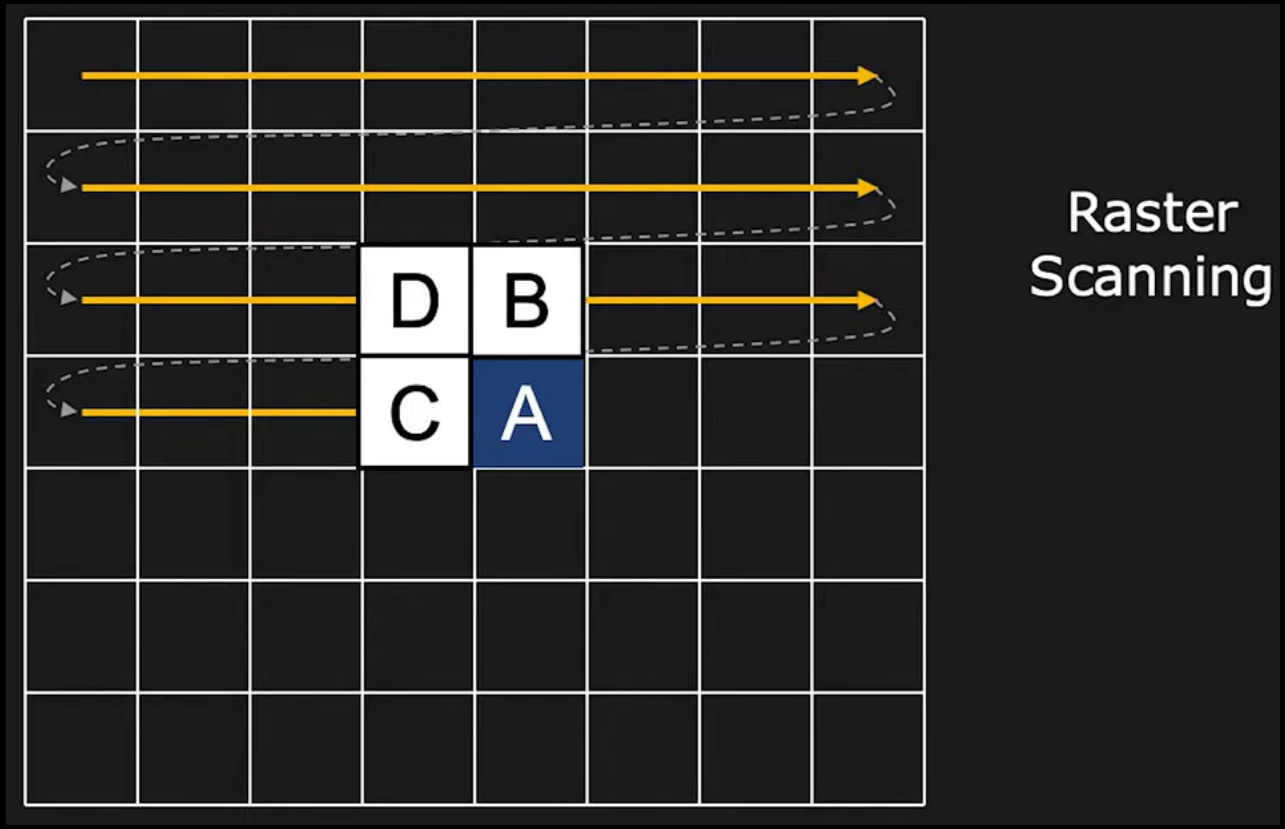
Image *I*



Integral Image *II*

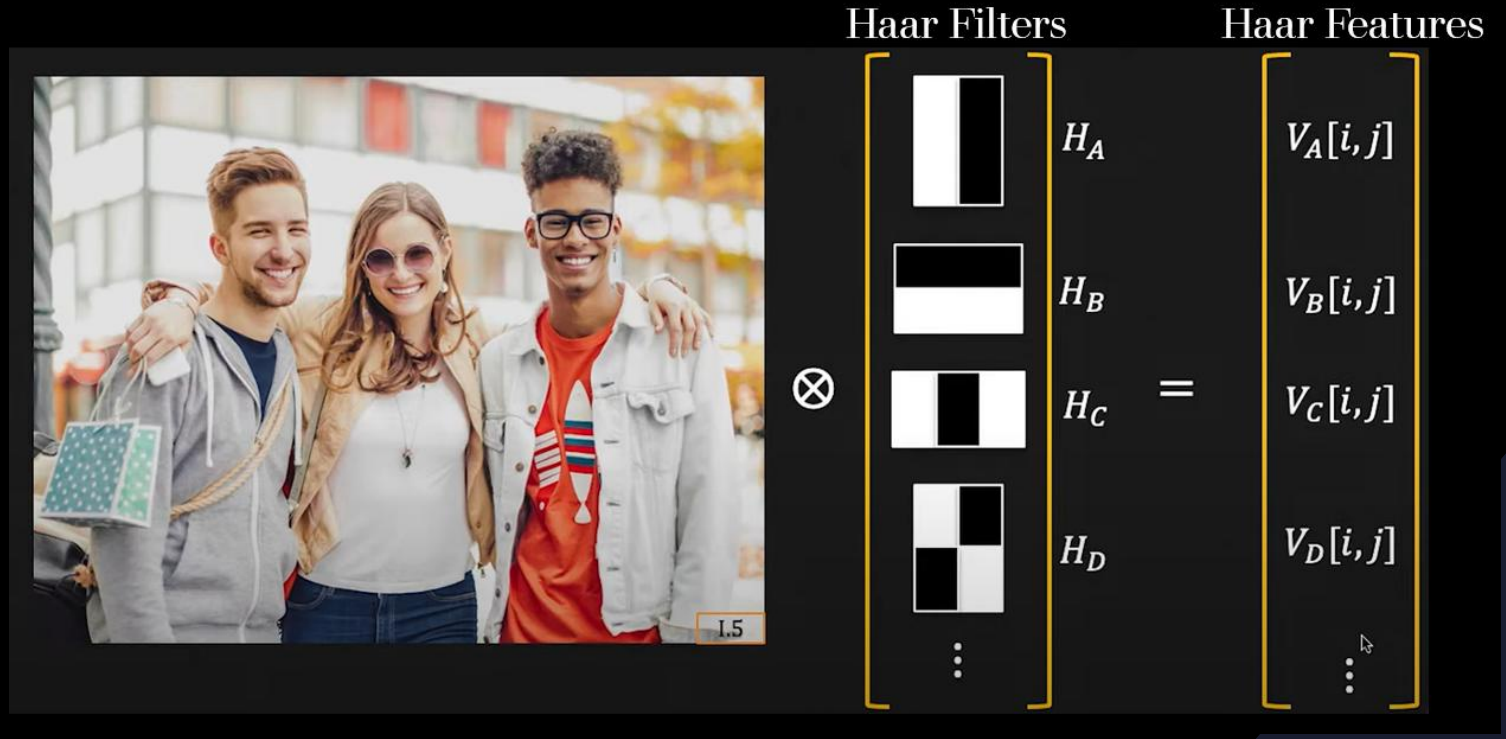
Computing Integral image

- $II_A = II_B + II_C - II_D + I_A$



Haar features using Integral image

- Integral image needs to be computed once per test image
- Allows fast computations for Haar features



Programs

1. Design Haar filters of different scales
2. Form Integral Image from test image
3. Calculate Haar features using Integral image & Haar filters of different scales