



Image Analysis

Rasmus R. Paulsen

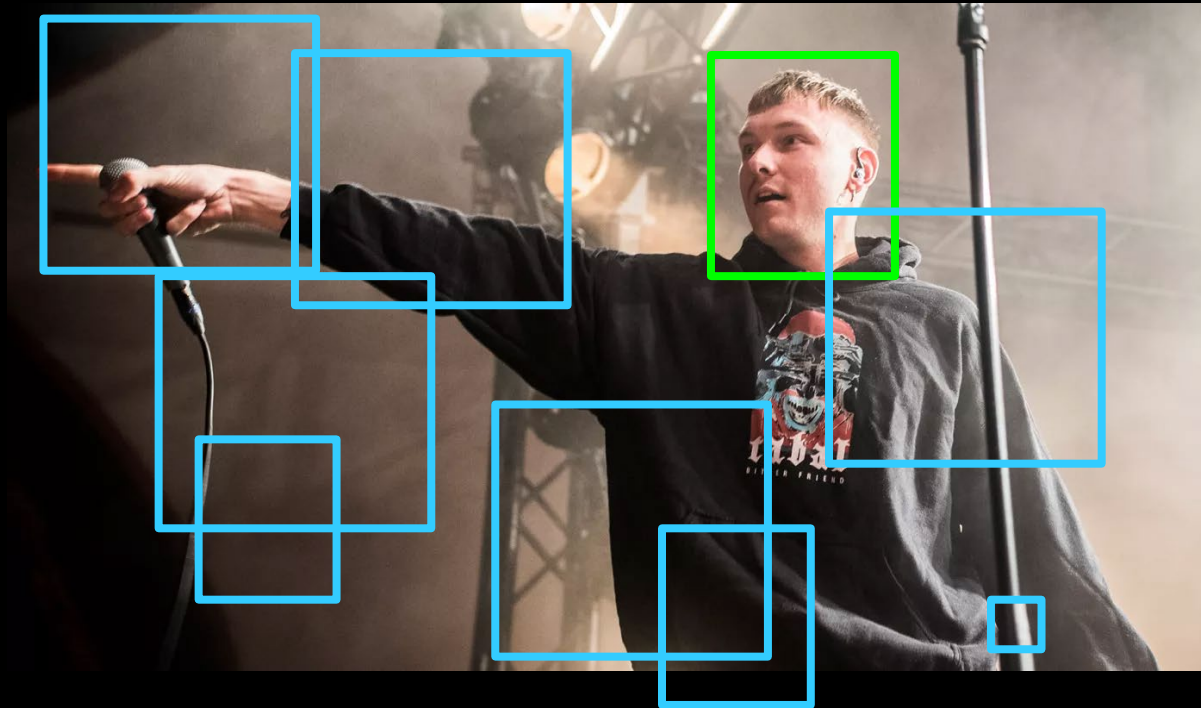
Tim B. Dyrby

DTU Compute

rapa@dtu.dk

<http://www.compute.dtu.dk/courses/02502>

Lecture 12 – Face detection using the Viola Jones method





What can you do after today?

- Describe the concept of face detection
- Describe the concept of Haar features
- Compute the values of 2, 3 and 4 rectangle Haar features
- Describe the integral image
- Compute the sum of pixels values in a rectangle using an integral image
- Describe the concept of a weak classifier
- Describe how several weak classifiers can be combined into a strong classifier
- Describe the attentional cascade
- Describe how faces can be detected using a moving window

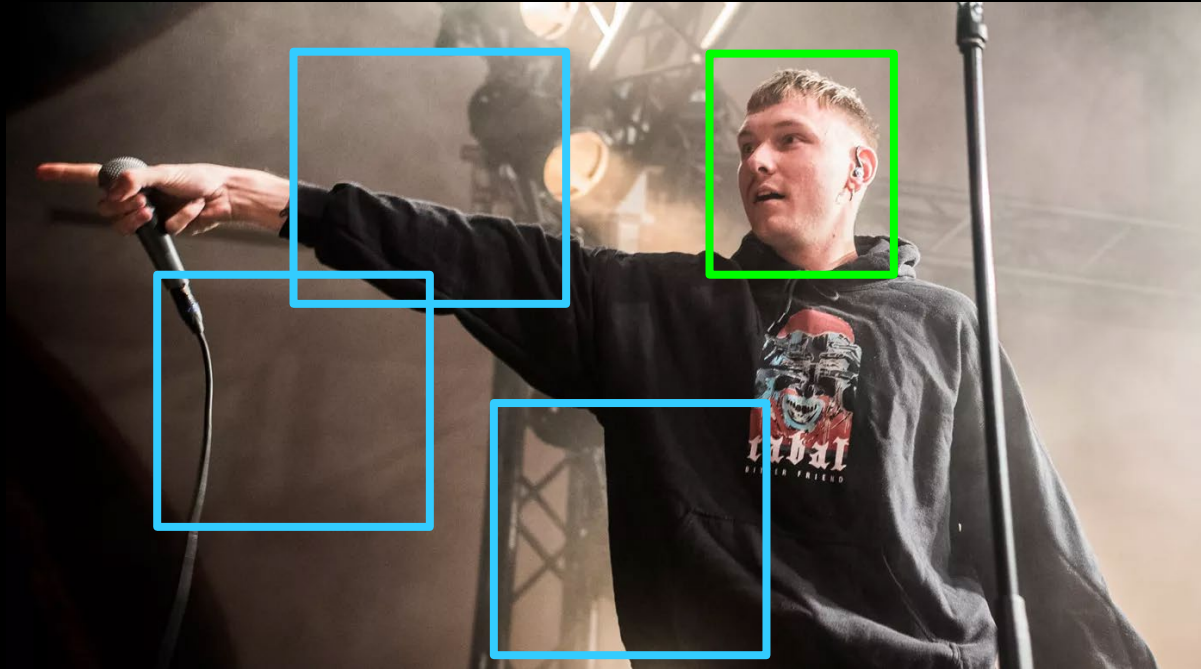
Face detection



■ First problem

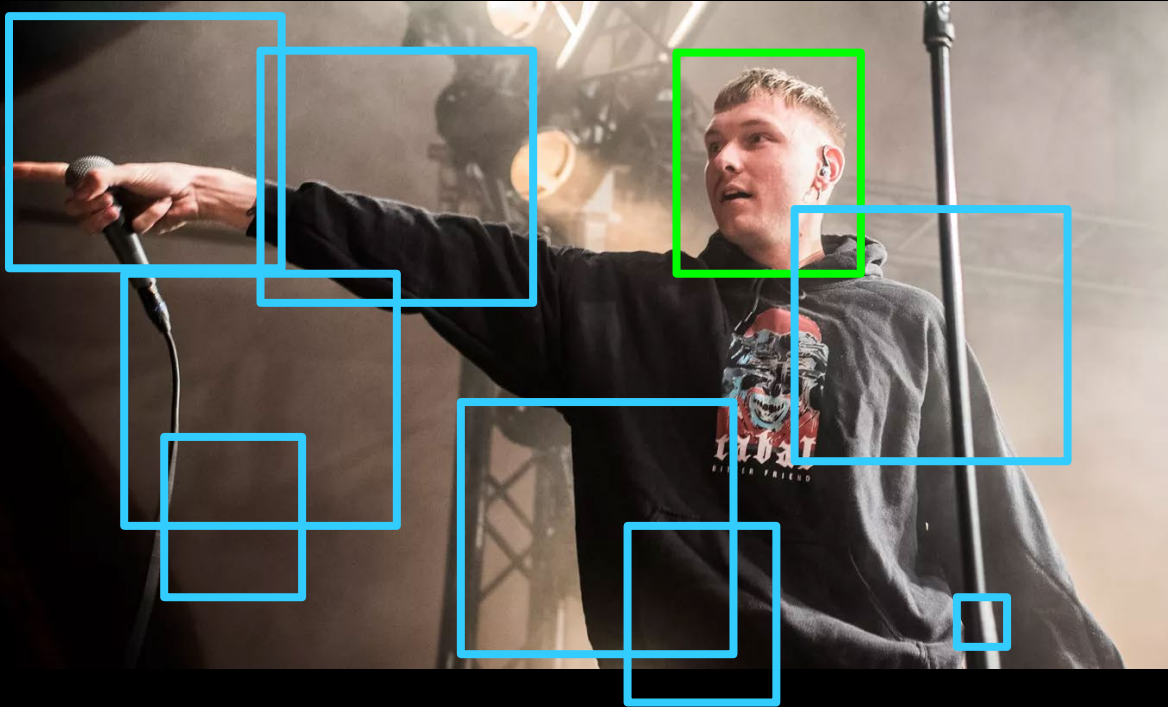
- Analyze a **window** in an image
- Is there a face in that window?

Face detection



- Slightly more advanced
 - Analyze many **windows** in an image
 - How many (if any) **windows** contain faces?

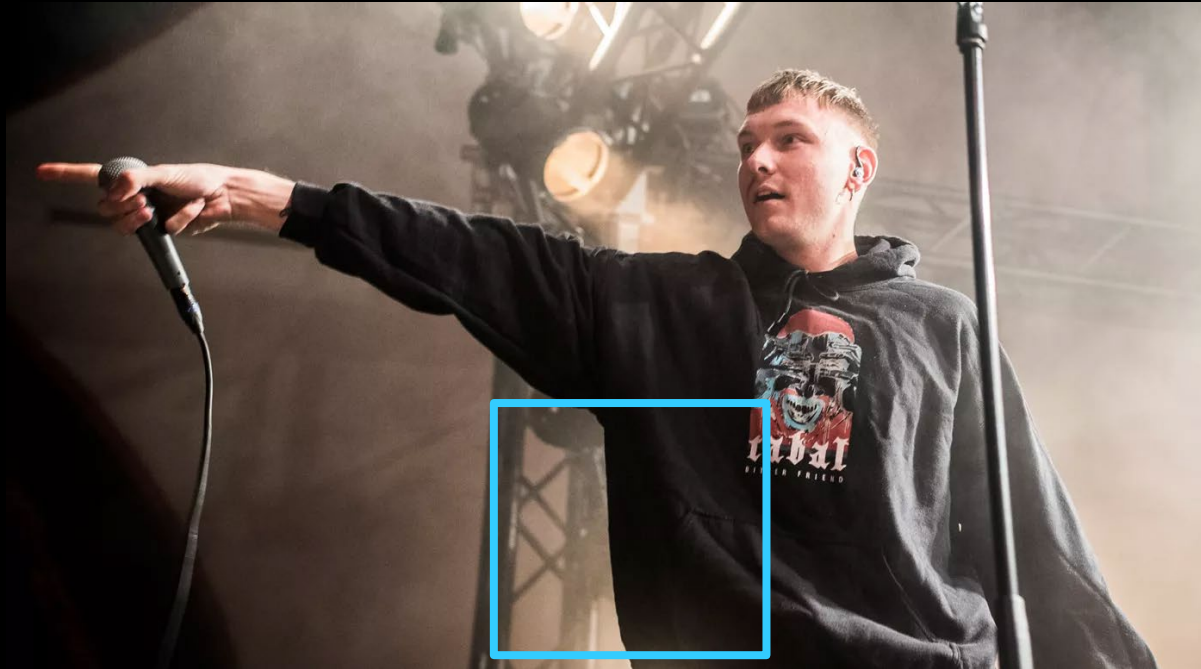
Face detection



■ Ideal

- Analyze (almost) all possible **windows** in an image
- How many (if any) **windows** contain faces?

What is needed?



- A fast method to determine if a *window* contains a face

Primary task – image feature based classification

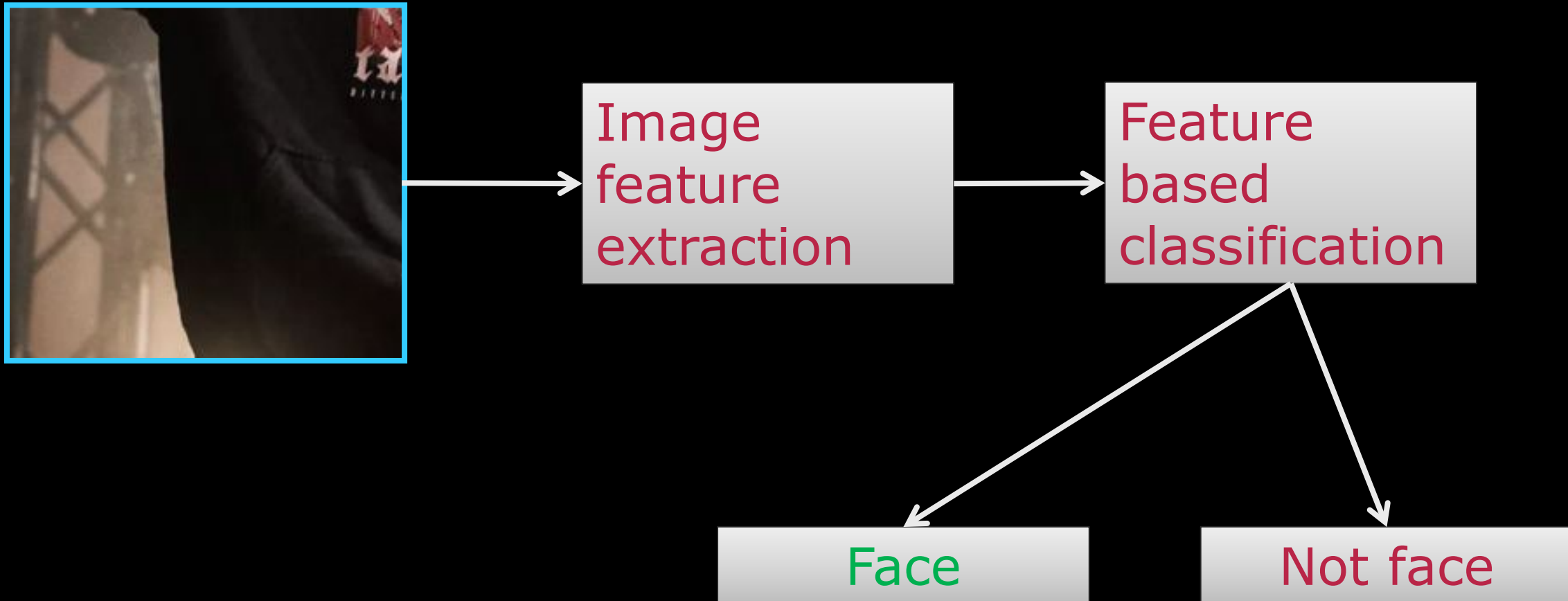


Image based features - what features can you think of?

The word cloud includes the following words: edges, symmetry, eyebrows, pattern, detector, size, chin, relation, something, area, lines, placement, contrast, circularity, shape, nose, eye, kevin, neckshape, outline, path, eyes, brows, mouth, color, magnus, 4-0, face, and face. The words are arranged in a dense cluster, with 'color' and 'magnus' being the largest and most prominent.

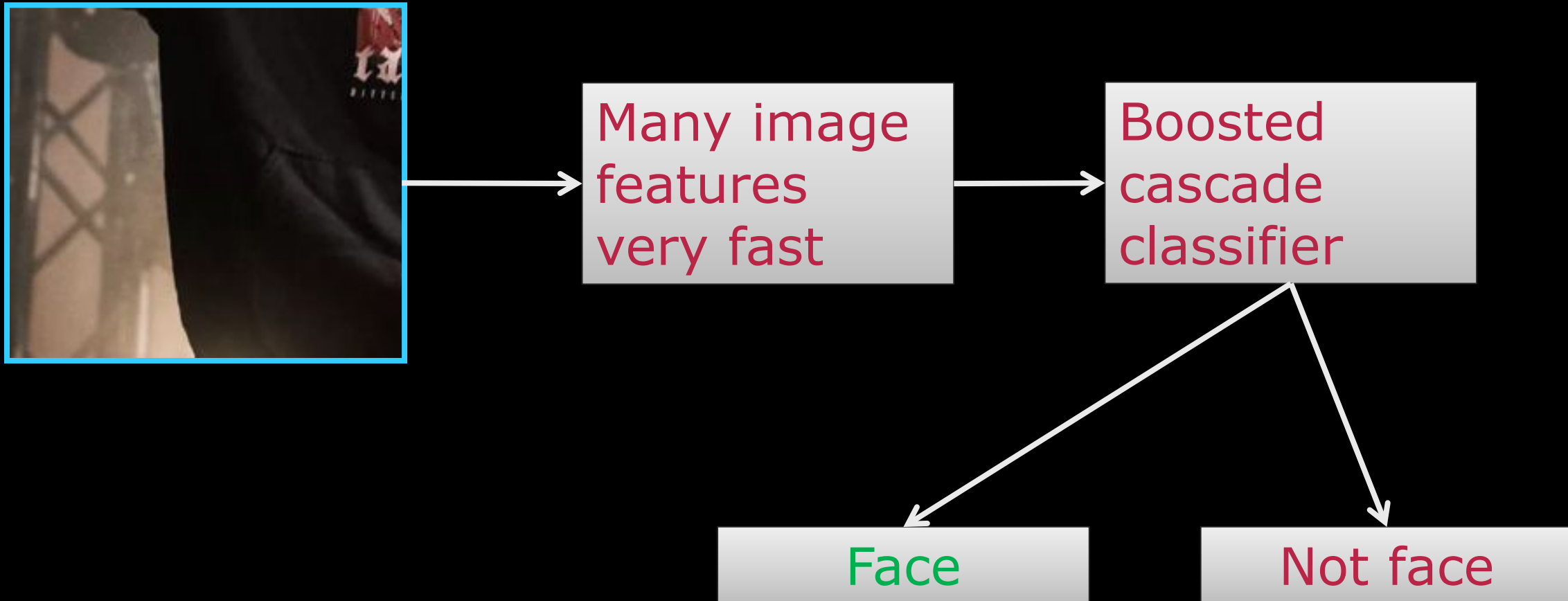
edges symmetry
eyebrows pattern
detector size
chin relation
something area
lines placement
contrast circularity
shape nose
eye kevin
neckshape outline
path eyes
brows brows
mouth color
magnus 4-0
face face

Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app



A word cloud featuring various terms related to facial features and image analysis. The most prominent words are 'color' and 'magnus' in large, bold, green and purple fonts respectively. Other visible words include 'edges', 'mouth', 'pattern', 'detector', 'size', 'chinese', 'relation', 'face', 'something', 'area', 'lines', 'placement', 'contrast', 'circularity', 'shape', 'nose', 'eye', 'eye', 'path', 'outline', 'neckshape', 'kevin', 'brows', 'eyebrows', 'symmetry', and '4-0'. The words are arranged in a dense, overlapping manner with various colors and orientations.

Viola Jones – fast features and smart classification



Training data



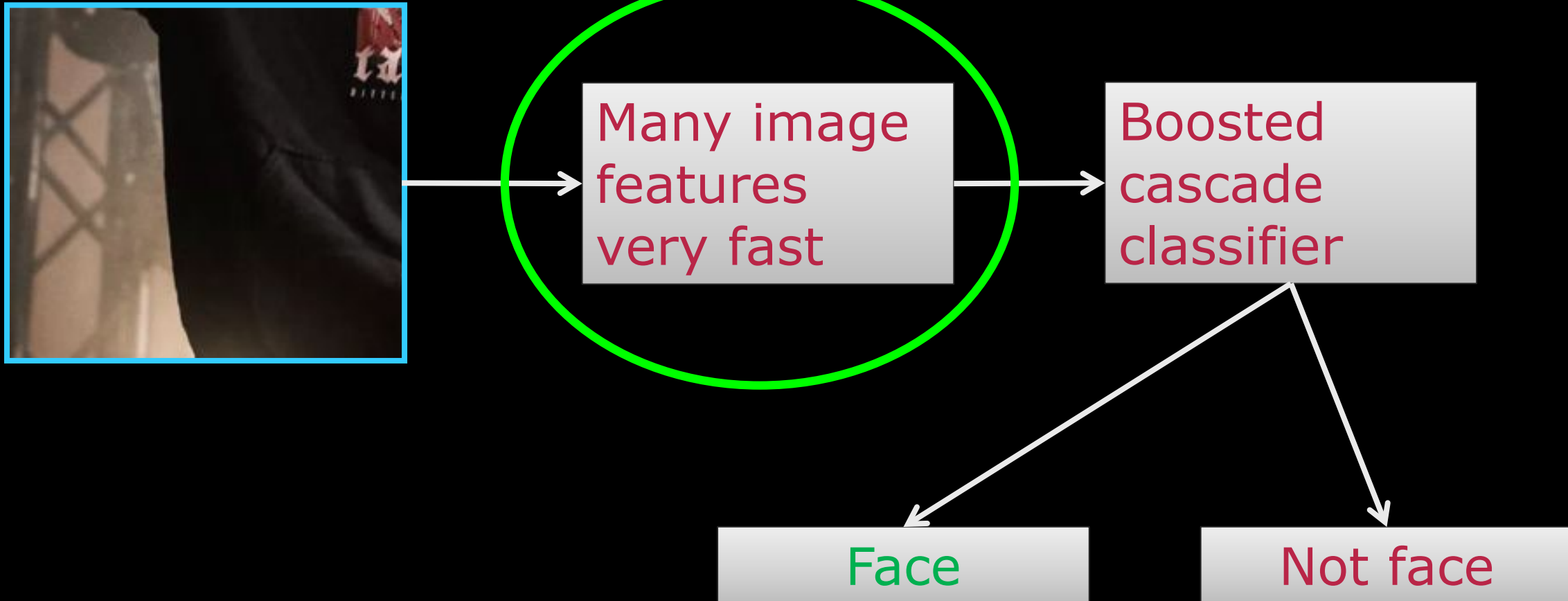
■ Face images:

- 4916 hand labelled faces
- Aligned and scaled to 24x24 pixels

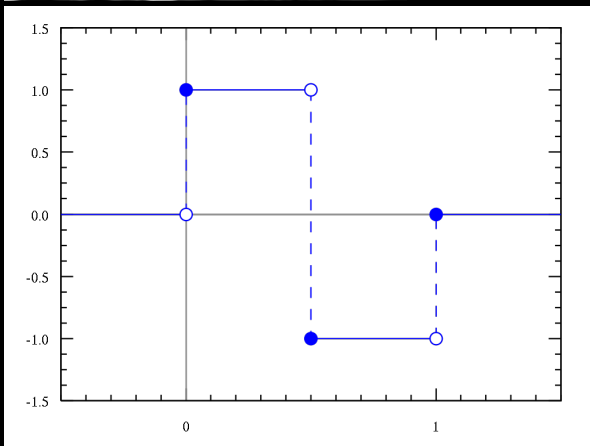
■ Non-face images:

- 9544 images with no faces
- 350 million sub-windows sampled from these

Viola Jones – fast features and smart classification



Haar features



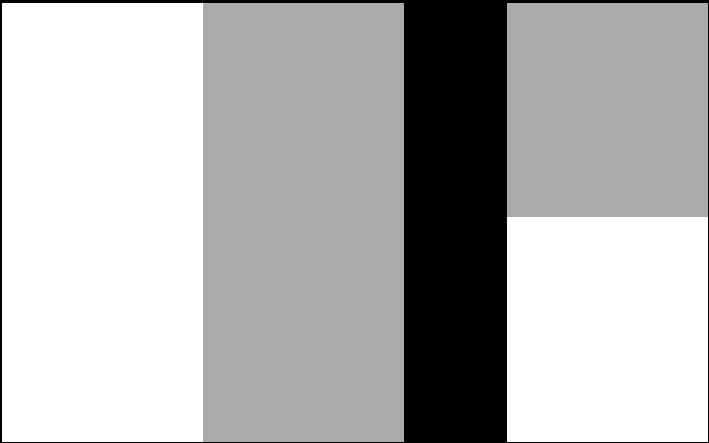
https://en.wikipedia.org/wiki/Haar_wavelet

- Alfred Haar (1885-1933)
 - Hungarian Mathematician
- Introduced the Haar wavelet in 1909
- *A wavelet is a wave-like oscillation with an amplitude that begins at zero, increases or decreases, and then returns to zero one or more times.*
- Simplest possible wavelet

<https://en.wikipedia.org/wiki/Wavelet>

Haar features

Two rectangle features



A

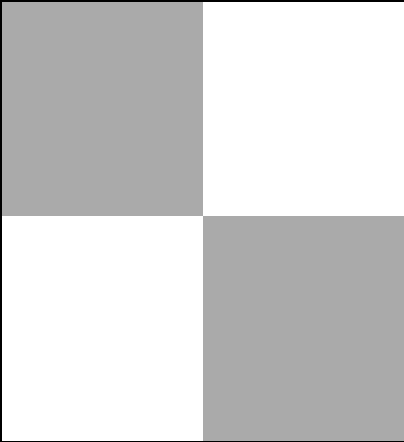
B

Three rectangle feature



C

Four rectangle feature



D

Feature =

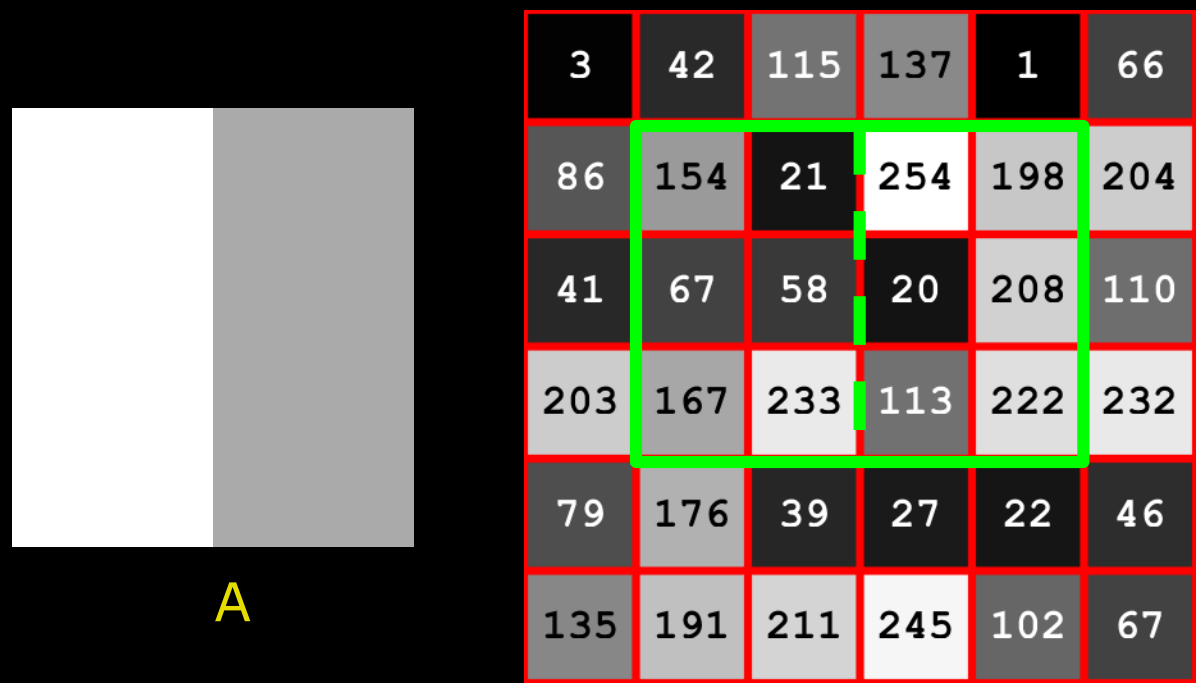
Sum of pixel values in image

 -

Sum of pixel values in image



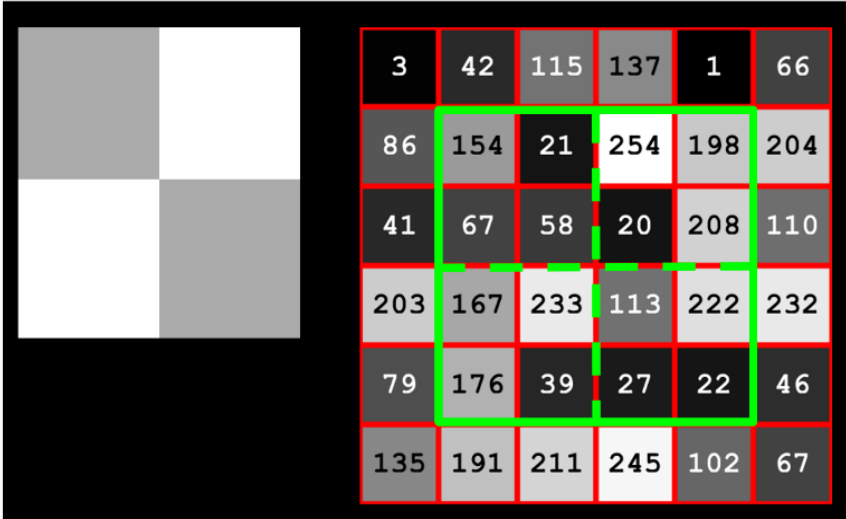
One Haar feature



Feature = 254+198+20+208+113+222-154-21-67-58-167-233 = 1015-700 = 315



Four rectangle Haar feature - what is the feature value?

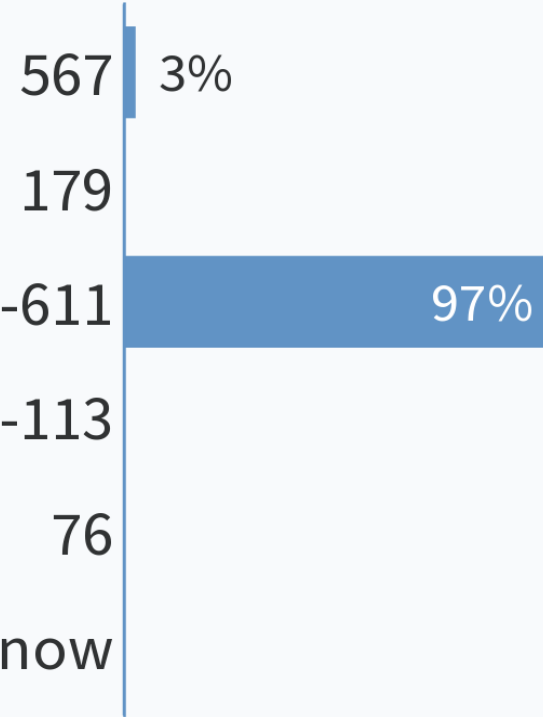
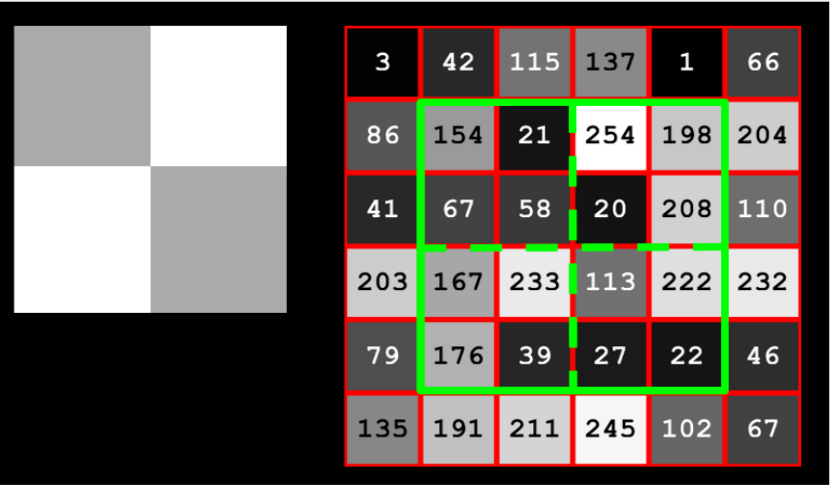


3	42	115	137	1	66
86	154	21	254	198	204
41	67	58	20	208	110
203	167	233	113	222	232
79	176	39	27	22	46
135	191	211	245	102	67

Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app



Four rectangle Haar feature - what is the feature value?



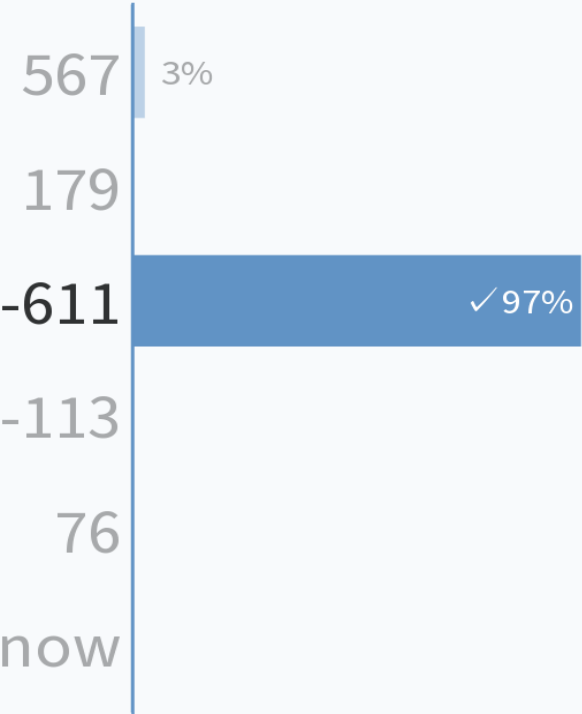
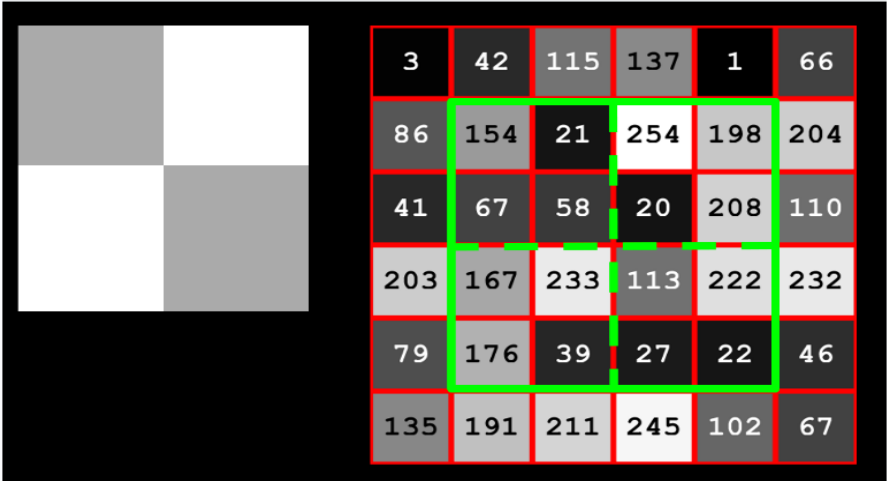
I do not know

Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app





Four rectangle Haar feature - what is the feature value?



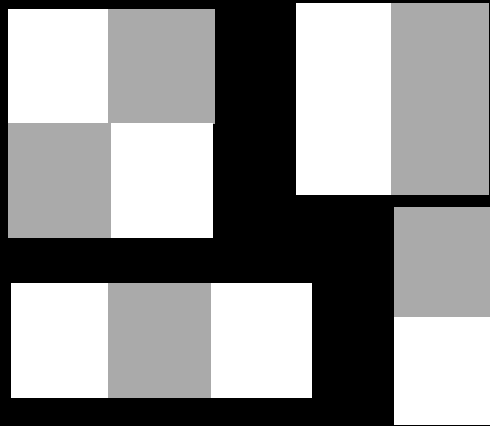
Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app



Fast computing of Haar features

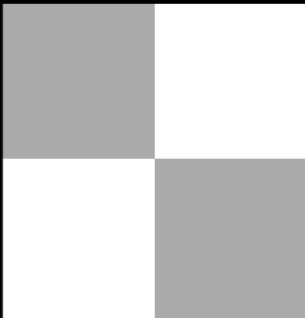


24 x 24 pixels



- Even for small Haar features, there are quite a lot of basic operations
- The larger the Haar feature, the more operations
- We need a fast way to compute Haar features

How many basic operations (plus and minus) are needed to compute the feature?

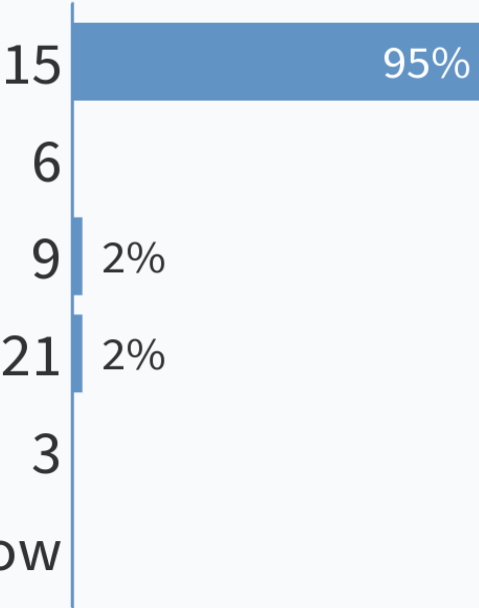
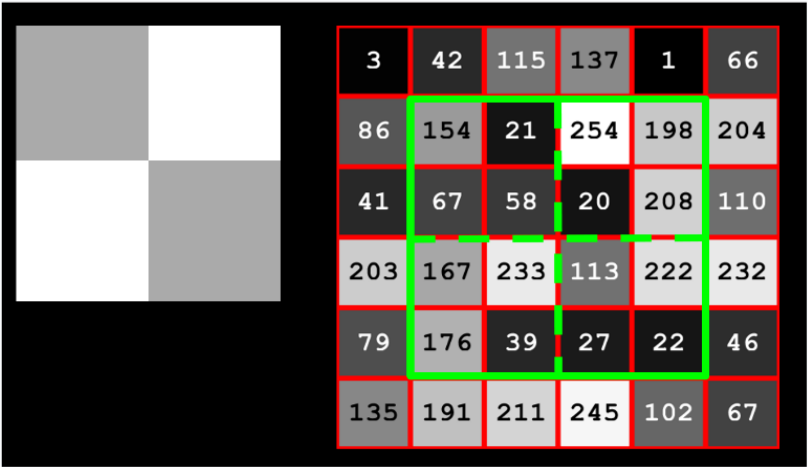


3	42	115	137	1	66
86	154	21	254	198	204
41	67	58	20	208	110
203	167	233	113	222	232
79	176	39	27	22	46
135	191	211	245	102	67

Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app



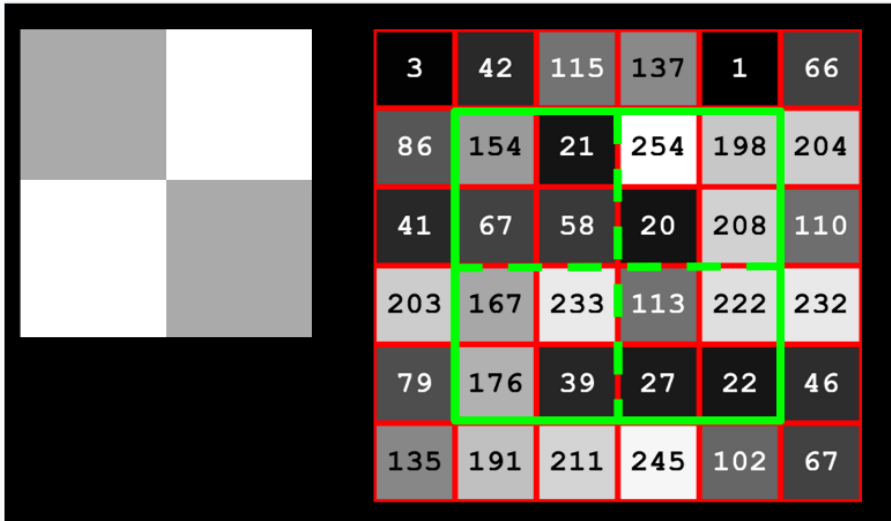
How many basic operations (plus and minus) are needed to compute the feature?



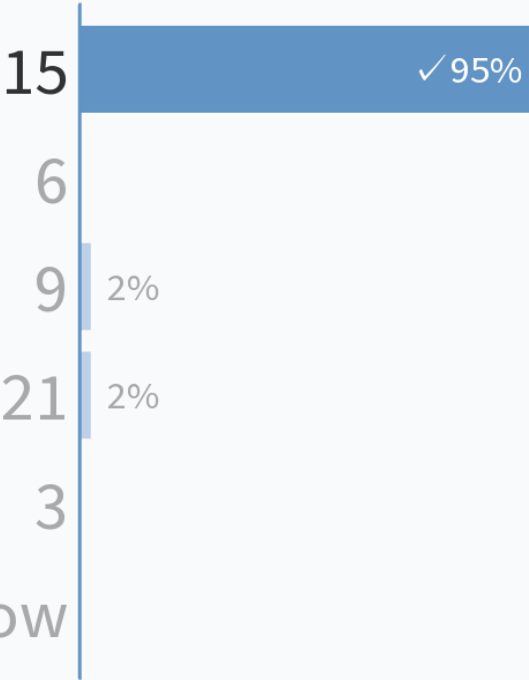
I do not know



How many basic operations (plus and minus) are needed to compute the feature?



I do not know



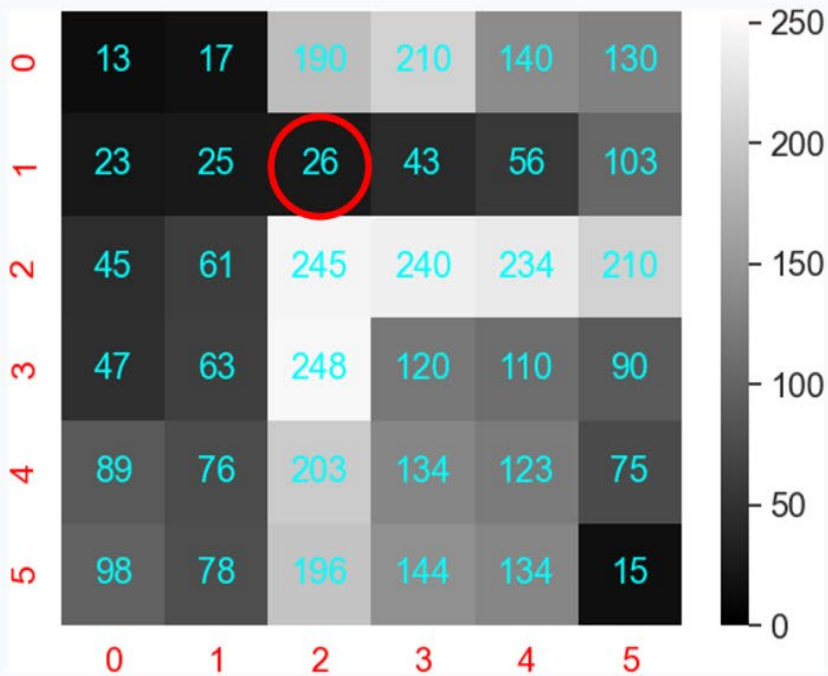
Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app

Fast computation of Haar features – the integral image



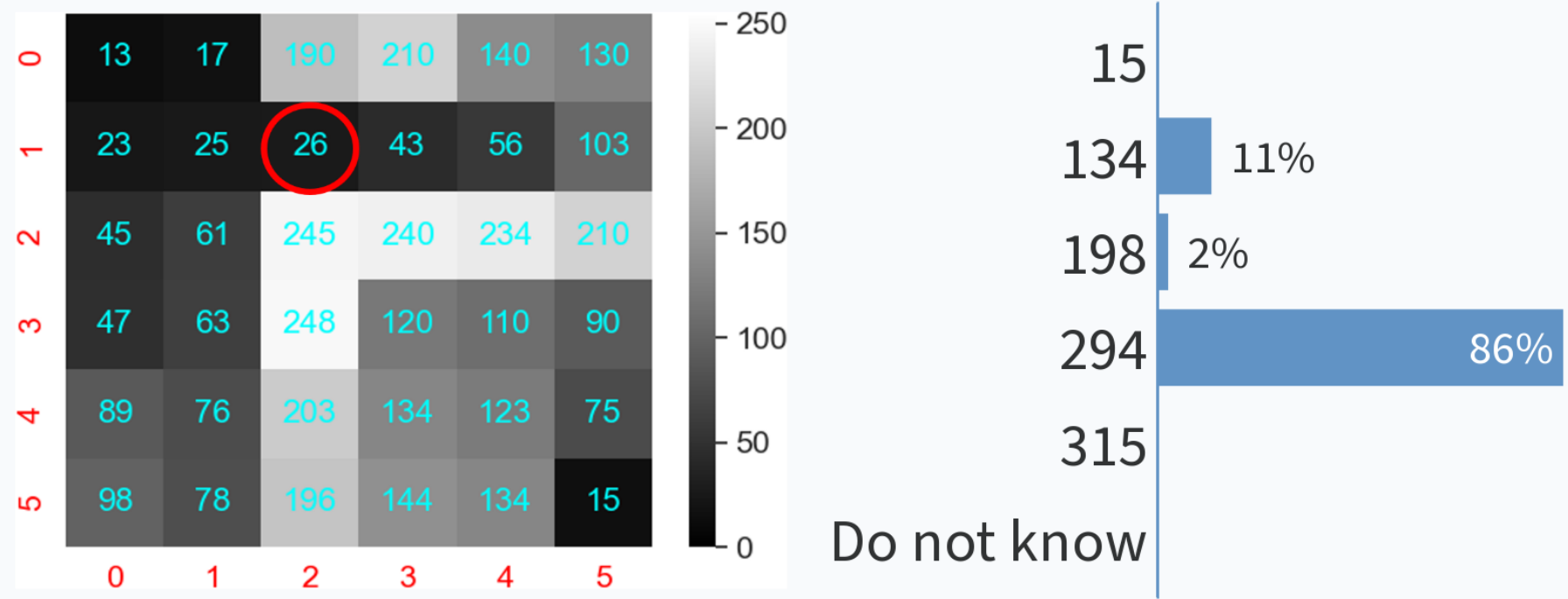
- In an integral image the pixel value is:
 - The sum of pixel above it and to the left of it in the original image
 - Including the pixel itself
- Can be computed very fast

Computing the integral image - what is the value in the marked pixel?



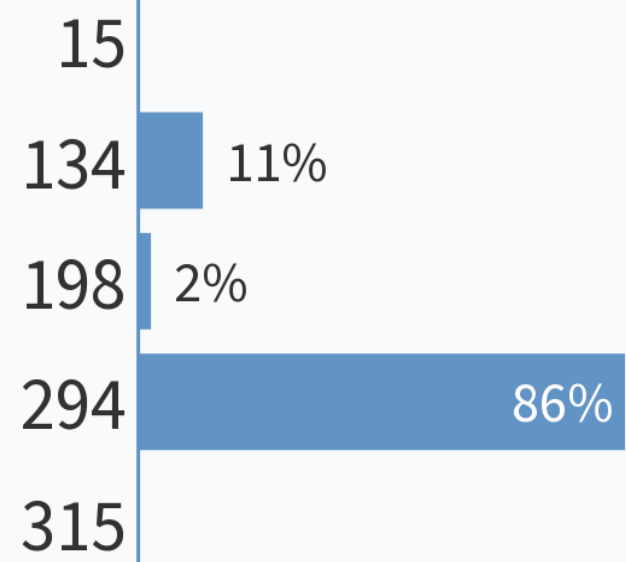
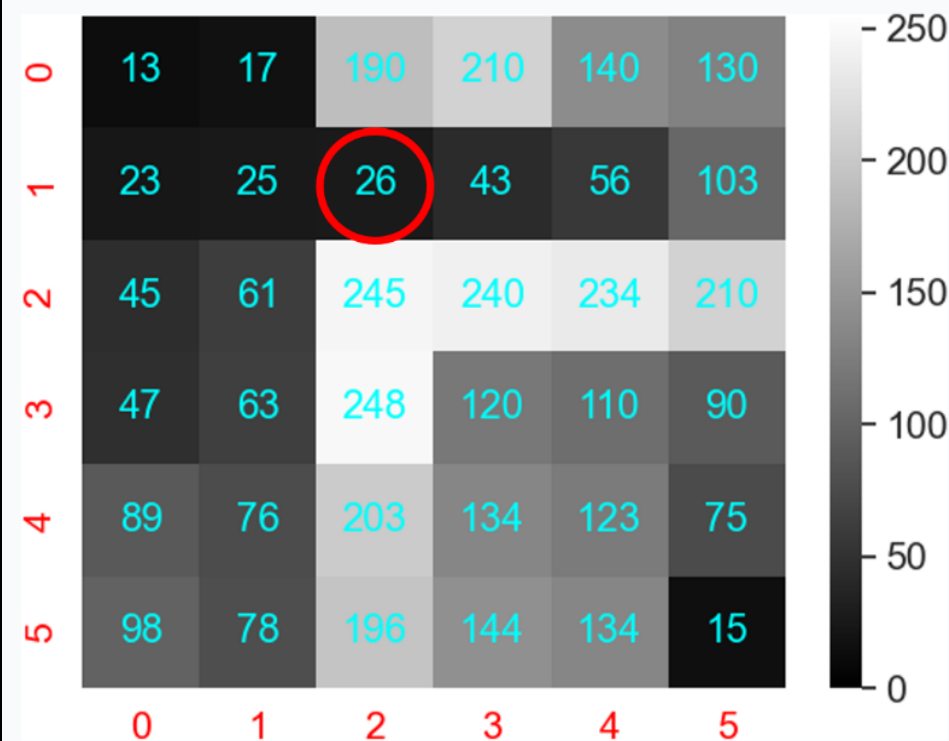
- 15
- 134
- 198
- 294
- 315
- Do not know

Computing the integral image - what is the value in the marked pixel?





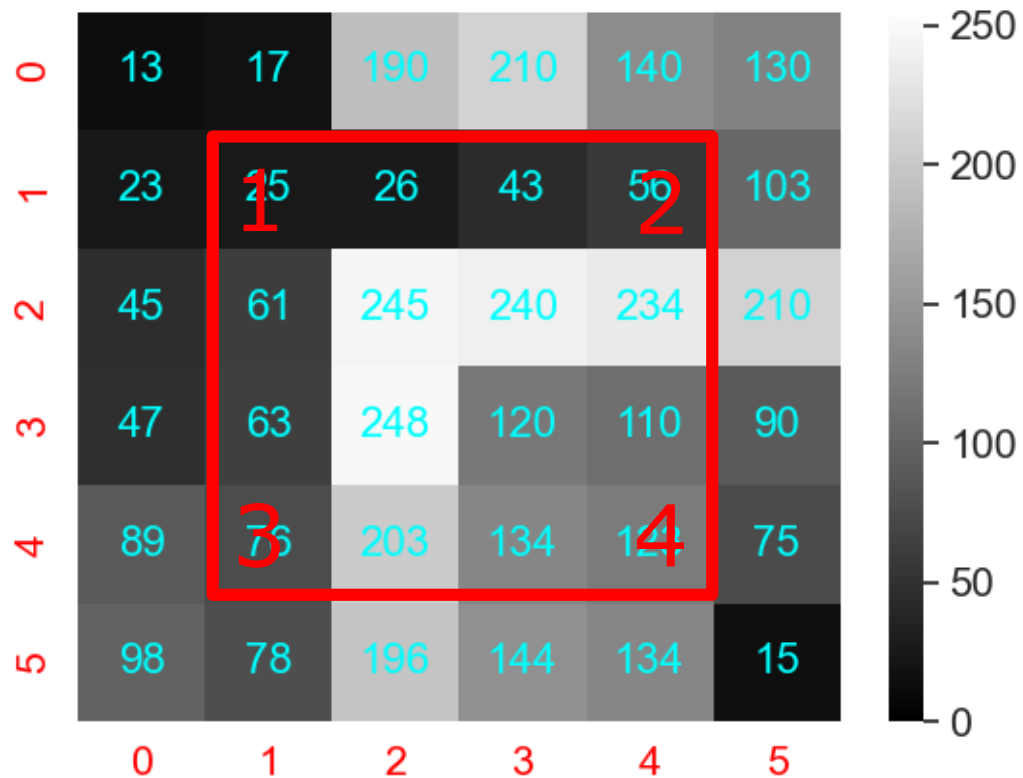
Computing the integral image - what is the value in the marked pixel?



Do not know

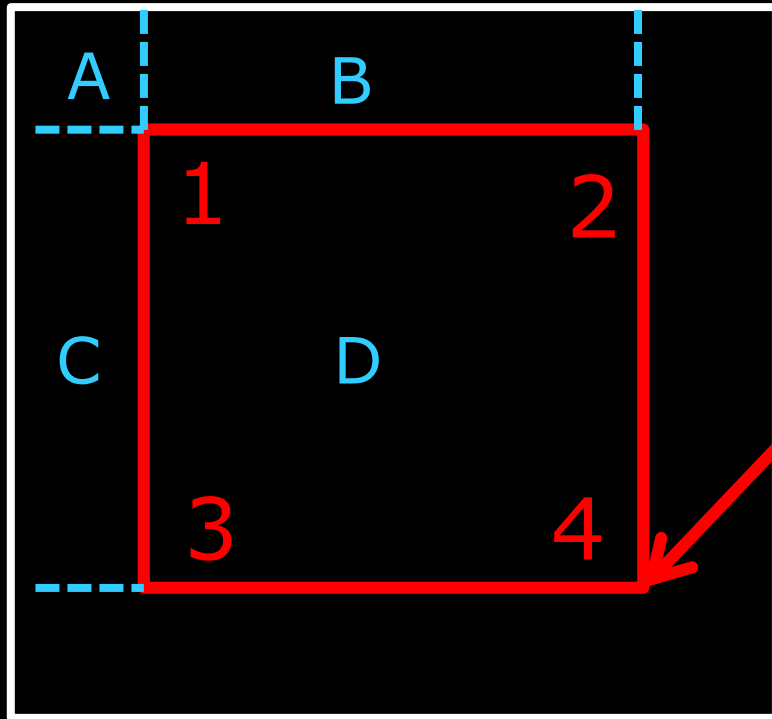


Using the integral image



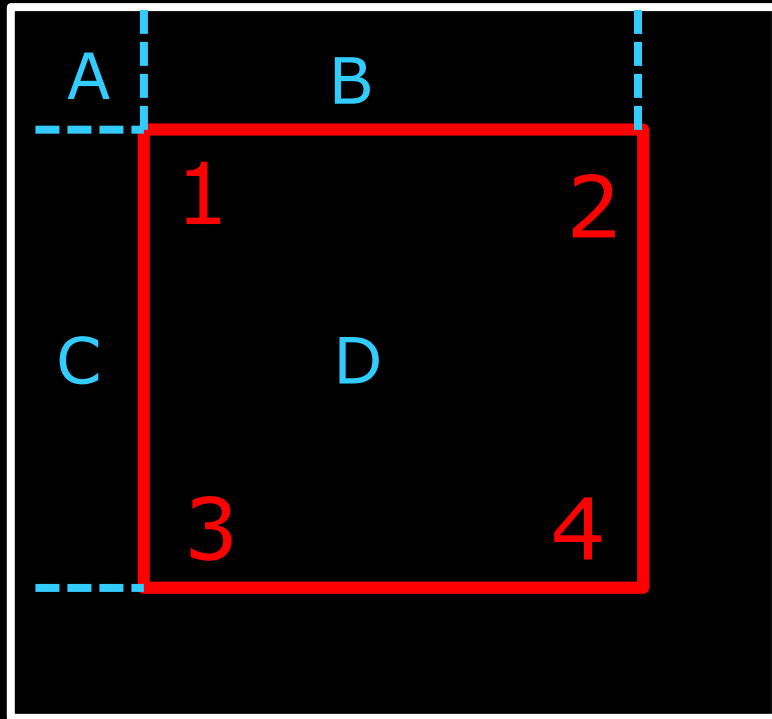
- We want to compute the pixel sum in the rectangle
- Defined by four corners: 1, 2, 3, 4

Using the integral image



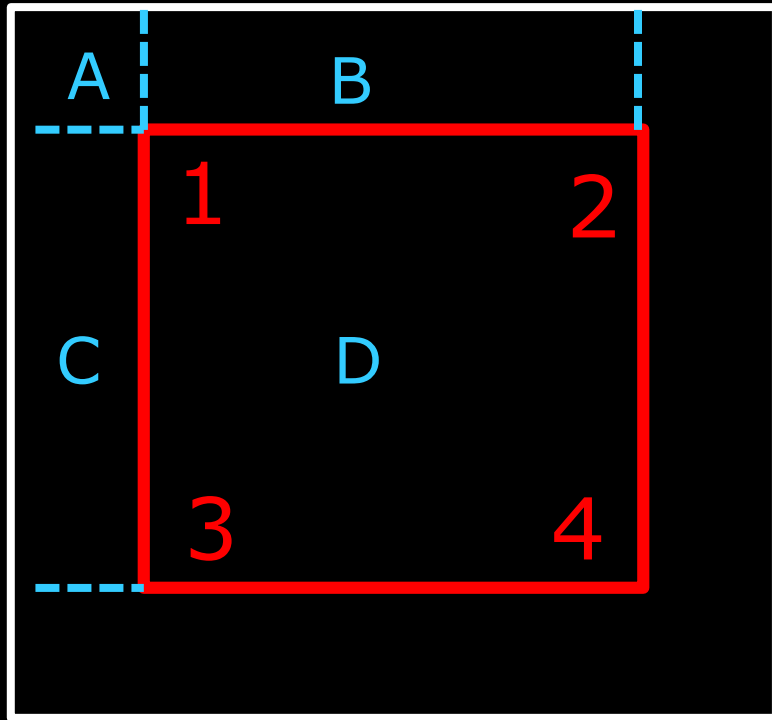
- Define four regions:
 - A, B, C, D
- The sum of pixels in the area
 - $A+B+C+D$ is the value of the integral image at point 4

Using the integral image



- The sum of pixels in the area
 - $A+B$ is the value of the integral image at point 2
 - $A+C$ is the value of the integral image at point 3

Using the integral image – short notation



■ The sum of pixels in the area

- $ii(2) = A+B$

- $ii(3) = A+C$

- $ii(4) = A+B+C+D$

- $ii(1) = A$

- $ii(4)-ii(3)-ii(2) = D - A$

■ $ii(4)-ii(3)-ii(2)+ii(1) = D$



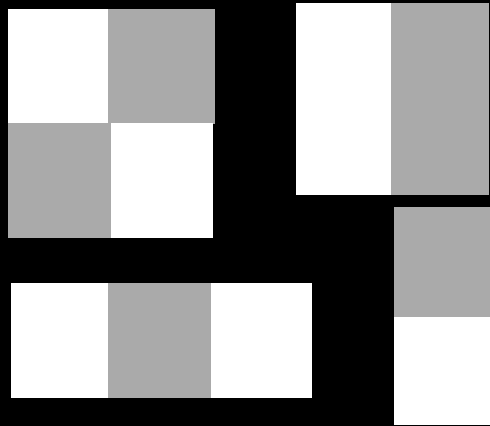
Course evaluation

- Very important to get your feedback on the course
- Please do it now – log into DTU Inside and fill in the evaluation
- What works well – so we should keep it and strengthen that part
- What can be improved and how?
- The question about “The teacher gave me feedback on my progress”
 - Very hard with large courses
 - We try with quizzes, TAs, exercise solutions (these can be improved)

Haar features in an image window



24 x 24 pixels



- Image window of 24 x 24 pixels
- All possible sizes and shapes of Haar features
- More than 180.000 features according to Viola and Jones
- They are *overcomplete* – meaning there is a very high redundancy
- We need *feature selection*



Possible features

$f_1 =$

$f_2 =$

$f_3 =$

$f_4 =$

$f_5 =$

$f_6 =$

$f_7 =$

$f_8 =$

...

$f_{180000} =$



Feature selection – from the article



- There are over 180,000 rectangle features associated with each image sub-window, a number far larger than the number of pixels.
- Even though each feature can be computed very efficiently, computing the complete set is prohibitively expensive.
- Our hypothesis, which is borne out by experiment, is that a very small number of these features can be combined to form an effective classifier.
- The main challenge is to find these features



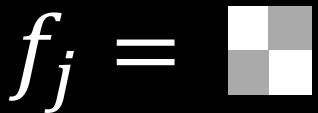
Learning Classification Functions

Weak classifier

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$



24 x 24 sub-
window



Feature value computed on the sub-window

$$p_j \in [-1, 1]$$

Parity – determines if the feature value should be positive or negative

$$\theta_j$$

Feature threshold

Weak classifier

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

$$x = \text{[Image of a person's arm]} \quad f_j(\text{[Crop of image]}) = \text{[2x2 Pixel Grid]} = 2049$$

Learnt by training: $p_j = 1 \quad \theta_j = 456$

$$\rightarrow 1 * 2049 < 1 * 456 \rightarrow h_j(\text{[Crop of image]}) = 0$$



What is this parity?

Weak classifier

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) < p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

$$x = \text{[Image of a person's arm]} \quad f_j(\text{[Crop of image]}) = \text{[4x4 Grayscale Feature Map]} = 2049$$

Learnt by training: $p_j = -1$ $\theta_j = 456$

$$\rightarrow -1 * 2049 < -1 * 456 \rightarrow h_j(\text{[Crop of image]}) = 1$$

Creating a strong classifier from weak classifiers

$$h(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{otherwise} \end{cases}$$

$$h_1(\text{img}) = \text{img}$$

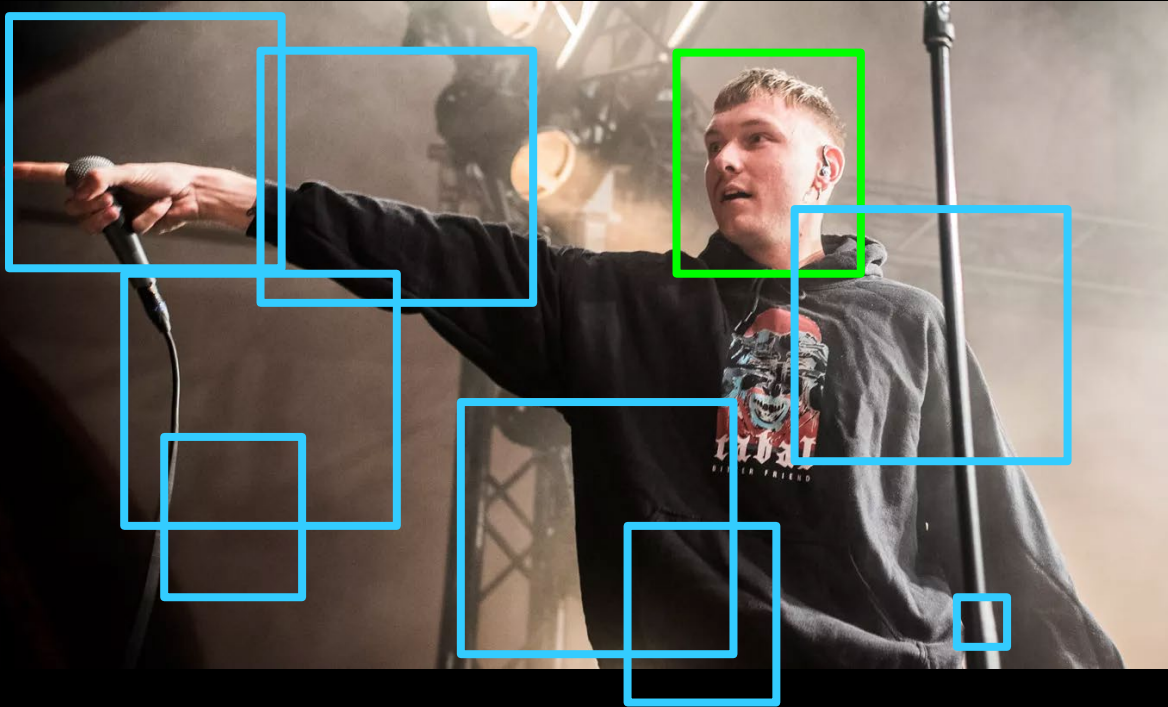
$$h_2(\text{img}) = \text{img}$$

...

$$h(\text{img}) = \alpha_1 h_1 + \alpha_2 h_2 + \dots + \alpha_T h_T$$

Learnt using AdaBoost

Boosted features – good performance but not enough



- Frontal face classifier with
 - T=200 features
 - Detection rate 95%
 - False positives 1 in 14084
 - 0.7 seconds for a 384 x 288

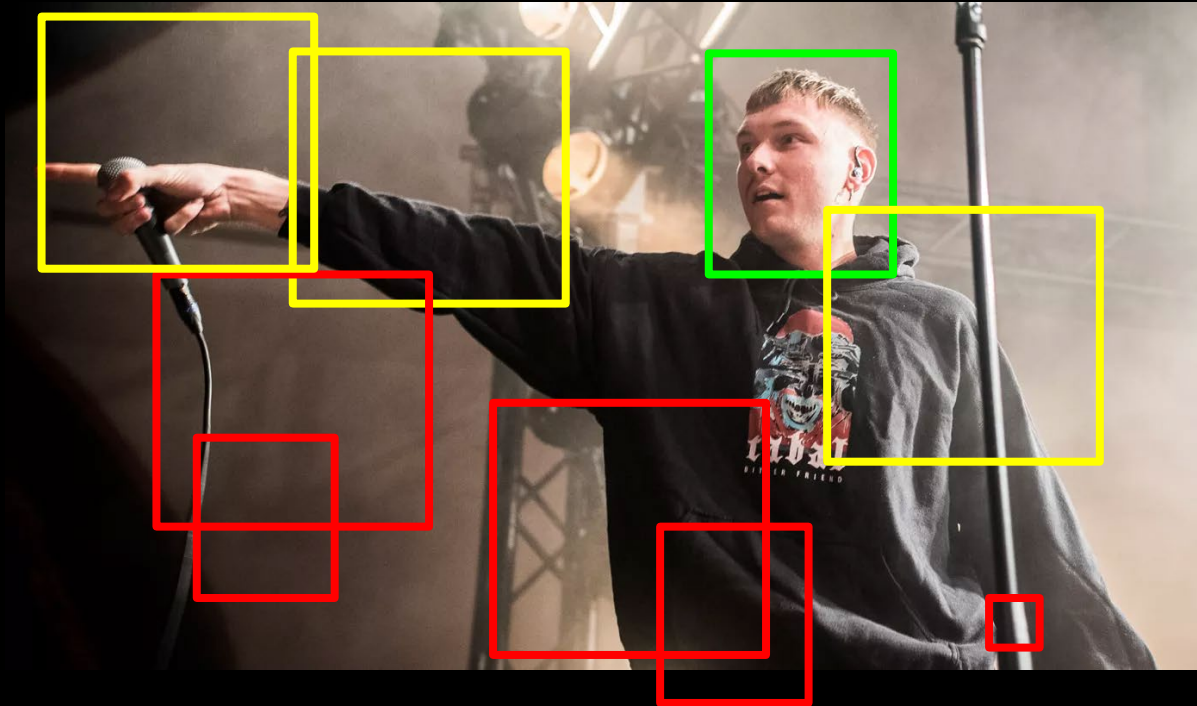
$$h_1(\text{image}) = \alpha_1 h_1 + \alpha_2 h_2 + \dots + \alpha_T h_T$$

The Attentional Cascade



Image Attention

- The process of focusing on specific parts of an image
 - Followed by fine grained analysis of selected windows



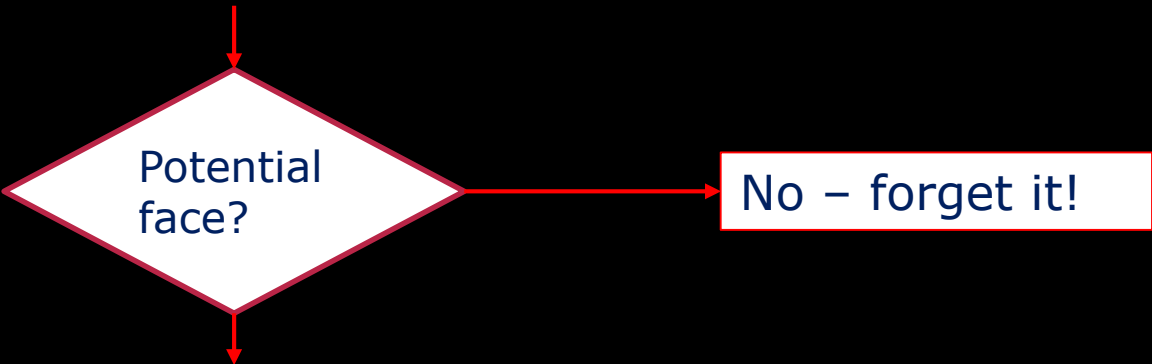
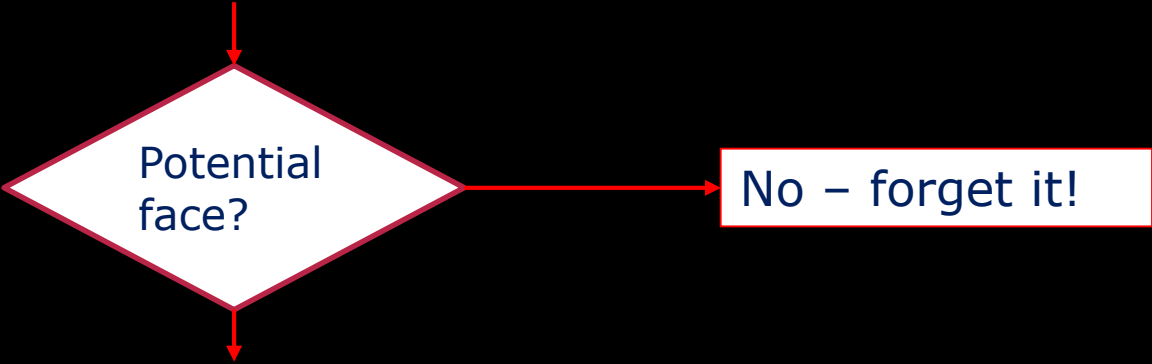
Focusing on potential face regions



Cascaded classifier



Also called a *degenerate decision tree*



What is a false negative?

A face window classified as face window

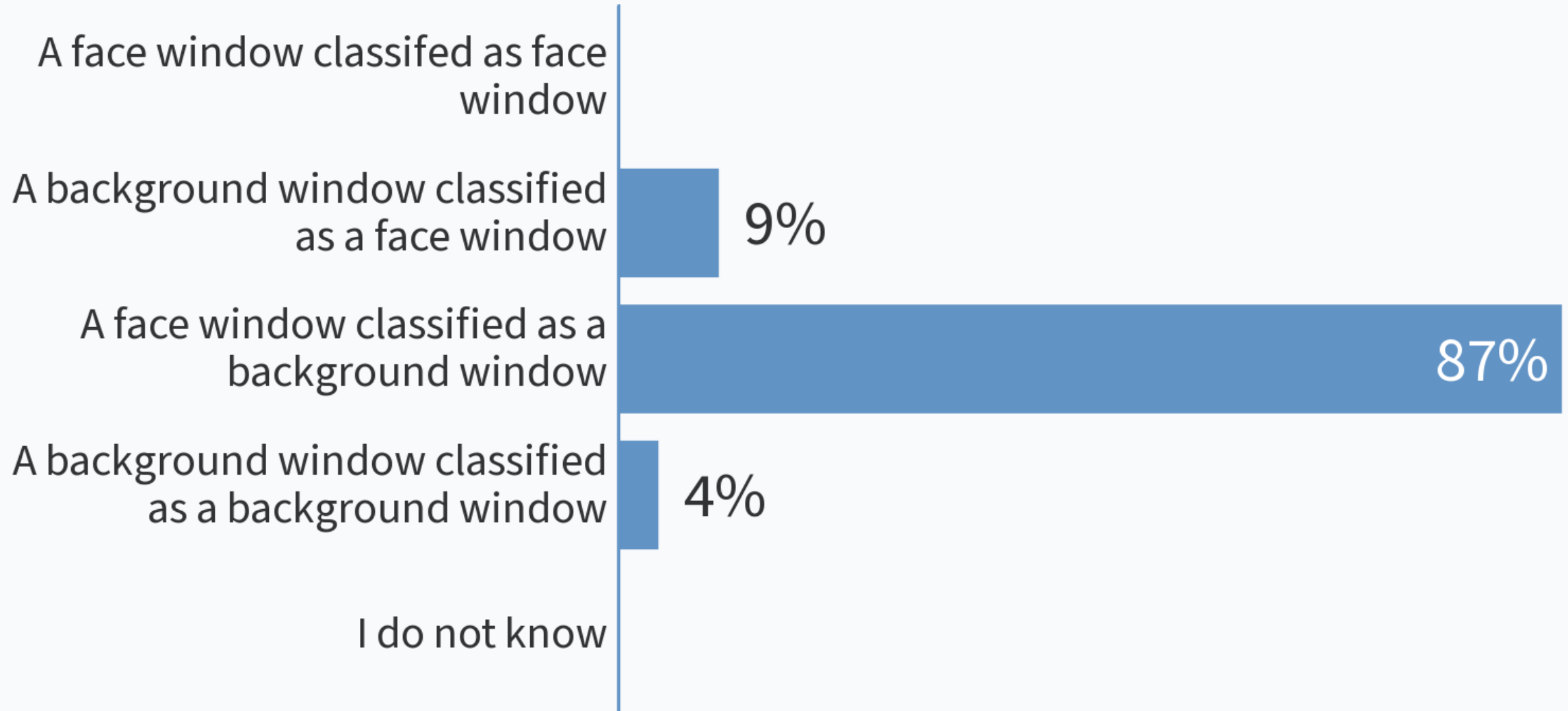
A background window classified as a face window

A face window classified as a background window

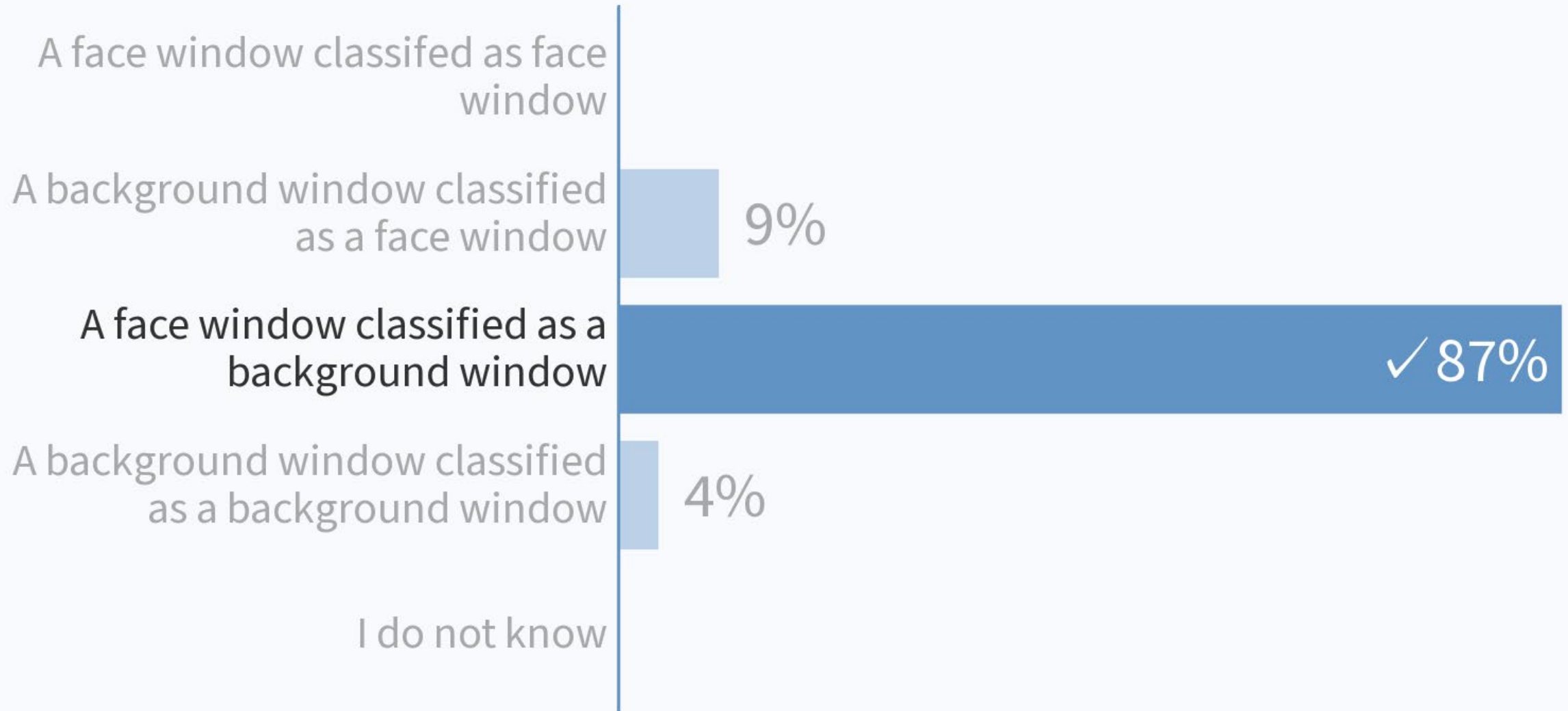
A background window classified as a background window

I do not know

What is a false negative?



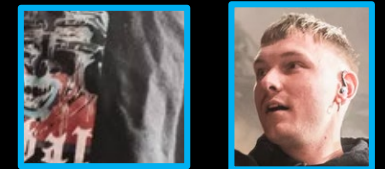
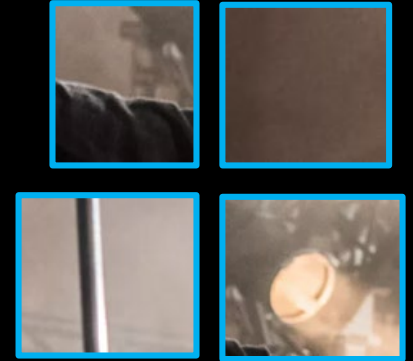
What is a false negative?



The attentional cascade



- Quickly reject negative sub-windows
 - Detect almost all positive sub-windows
 - False-negatives close to zero
 - Keep all potential face windows
 - Using the training set to find weights that fulfils this criterion
- Later more complex classifier
 - Low false positive rate



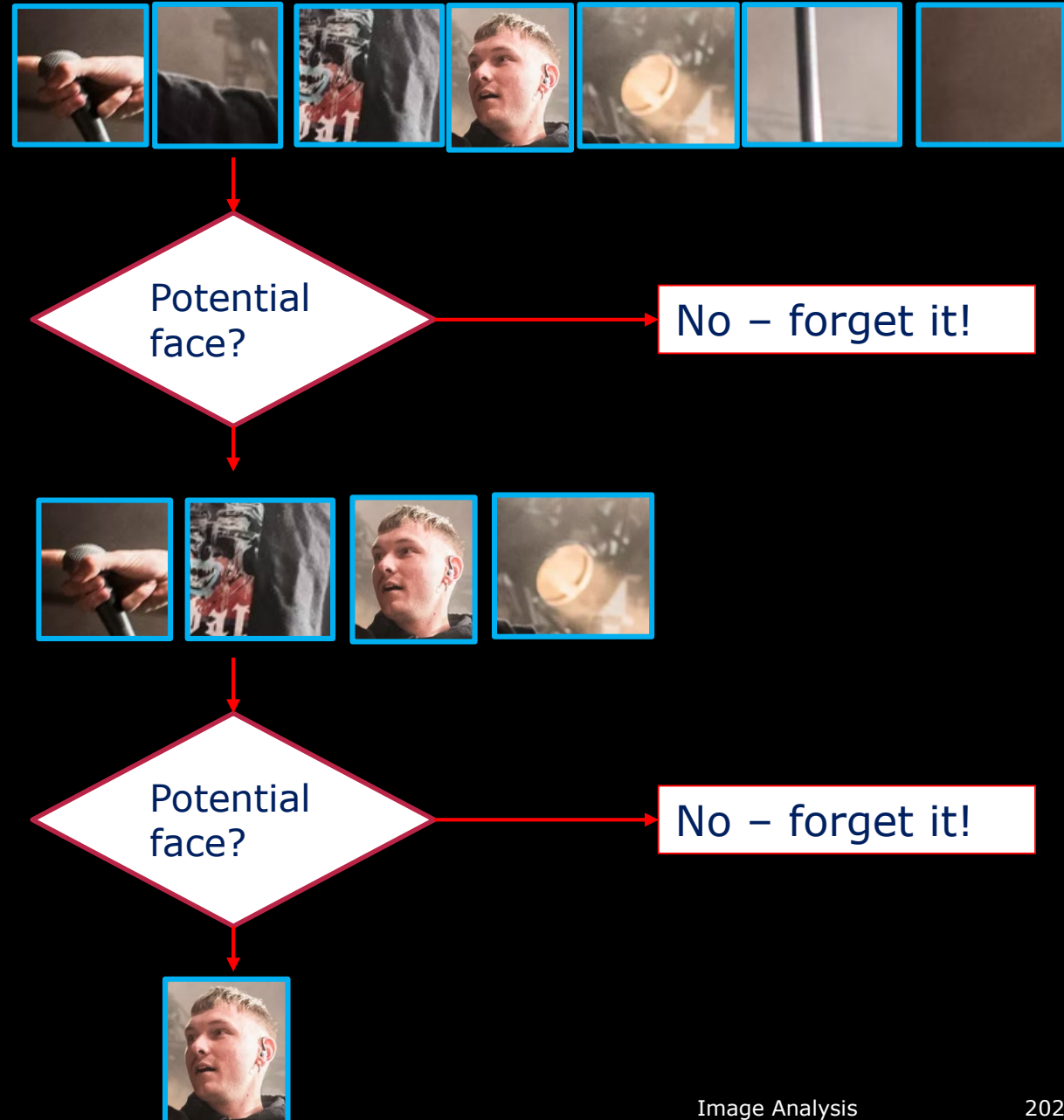
Training a cascade

$$h(\text{img}) = \alpha_1 h_1 + \alpha_2 h_2 + \dots + \alpha_T h_T$$

Learnt using AdaBoost

$$h(\text{img}) = \alpha_1 h_1 + \alpha_2 h_2 + \dots + \alpha_T h_T$$

Learnt using AdaBoost



First stage classifier

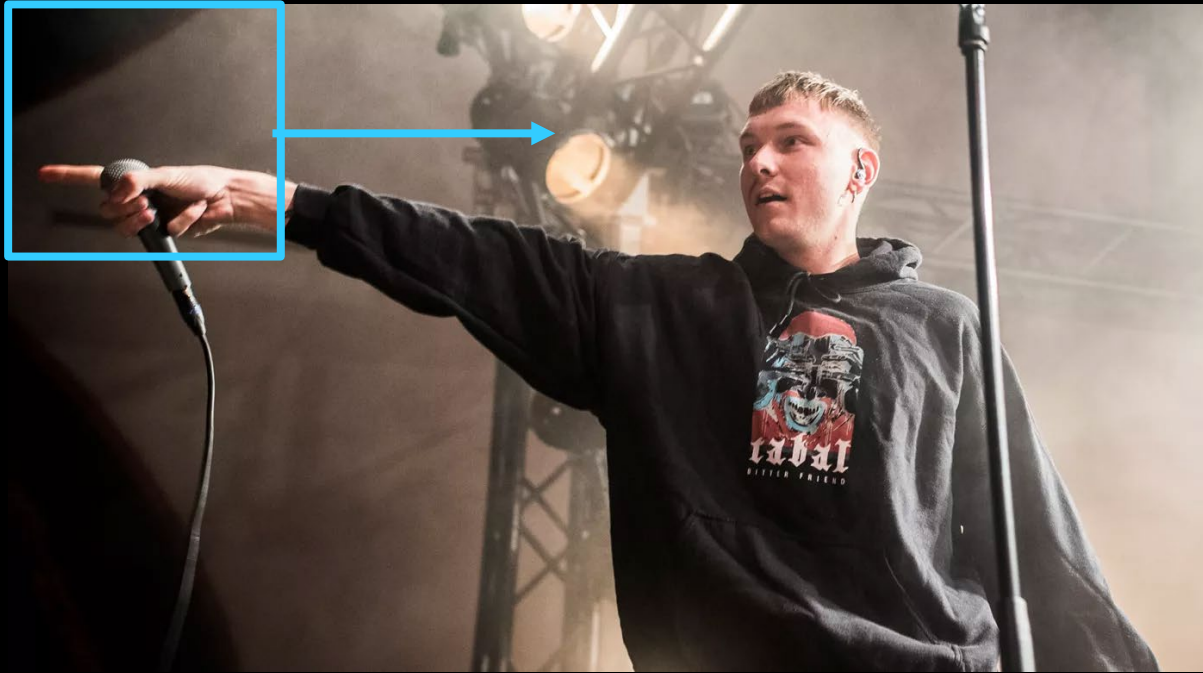


Final classifier



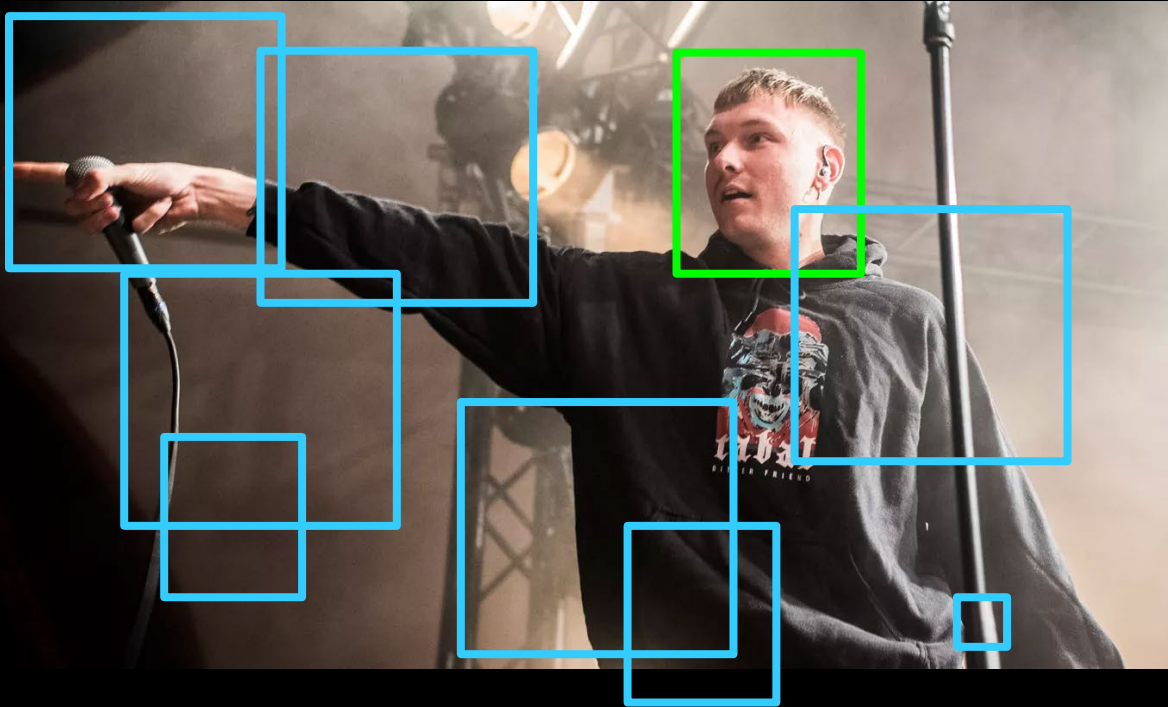
- 38 stages (step in the cascade)
- Total 6000 features (over the entire cascade)
- Faces are detected using on average 10 features per sub-window

Finding all faces in an image



- Slide a sub-window over the entire image
- Do a face detection for all positions
- Scale the features in a certain interval
 - To find faces of different sizes

Conclusion



- One of the most important algorithms before deep learning
- Uses many interesting concepts
 - Attention
 - Boosted weak classifiers
 - Very fast feature computation



Demo



Next week

- Advanced topics