

Face Recognition III

CSE 40537/60537 Biometrics

Daniel Moreira
Spring 2022

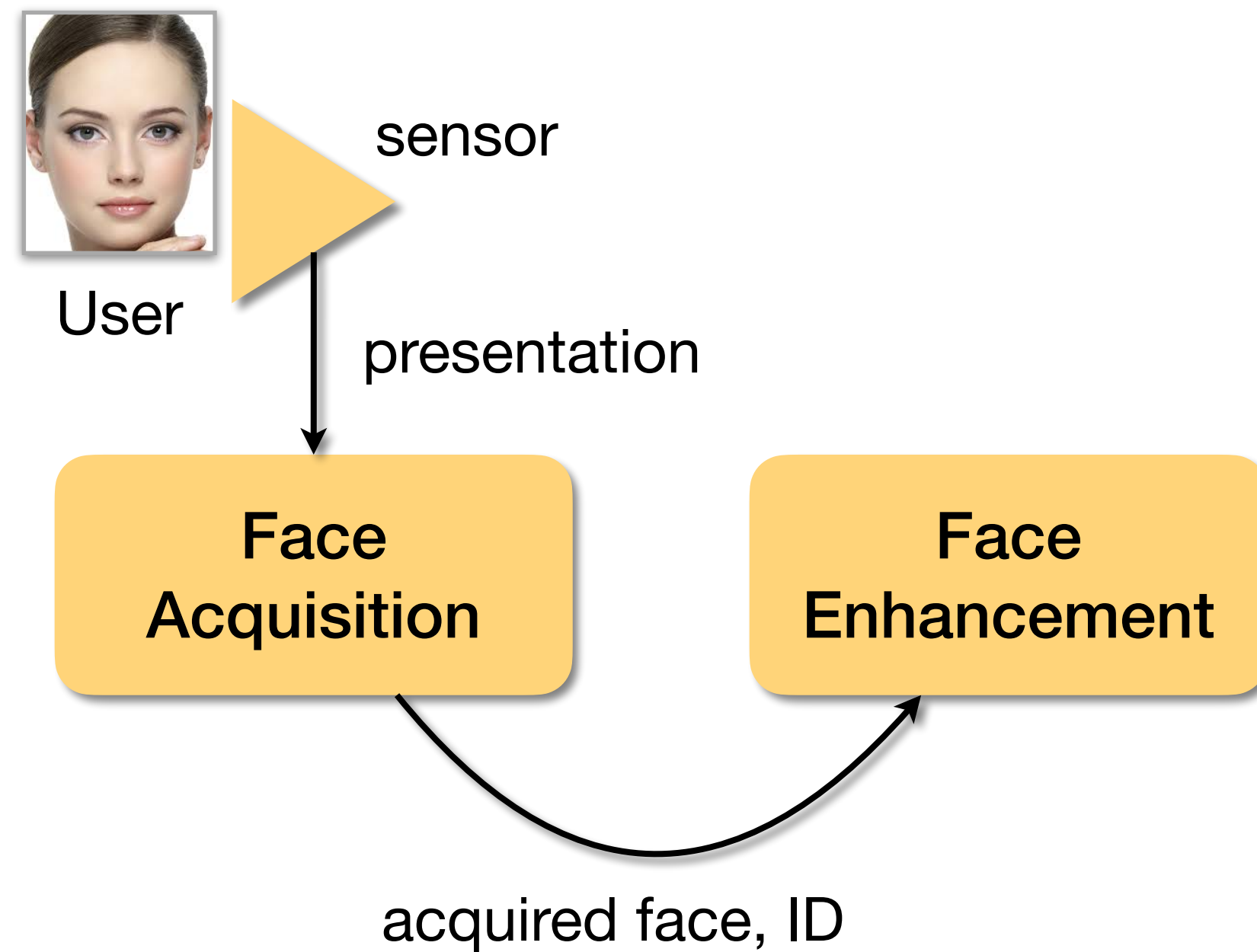


Today you will...

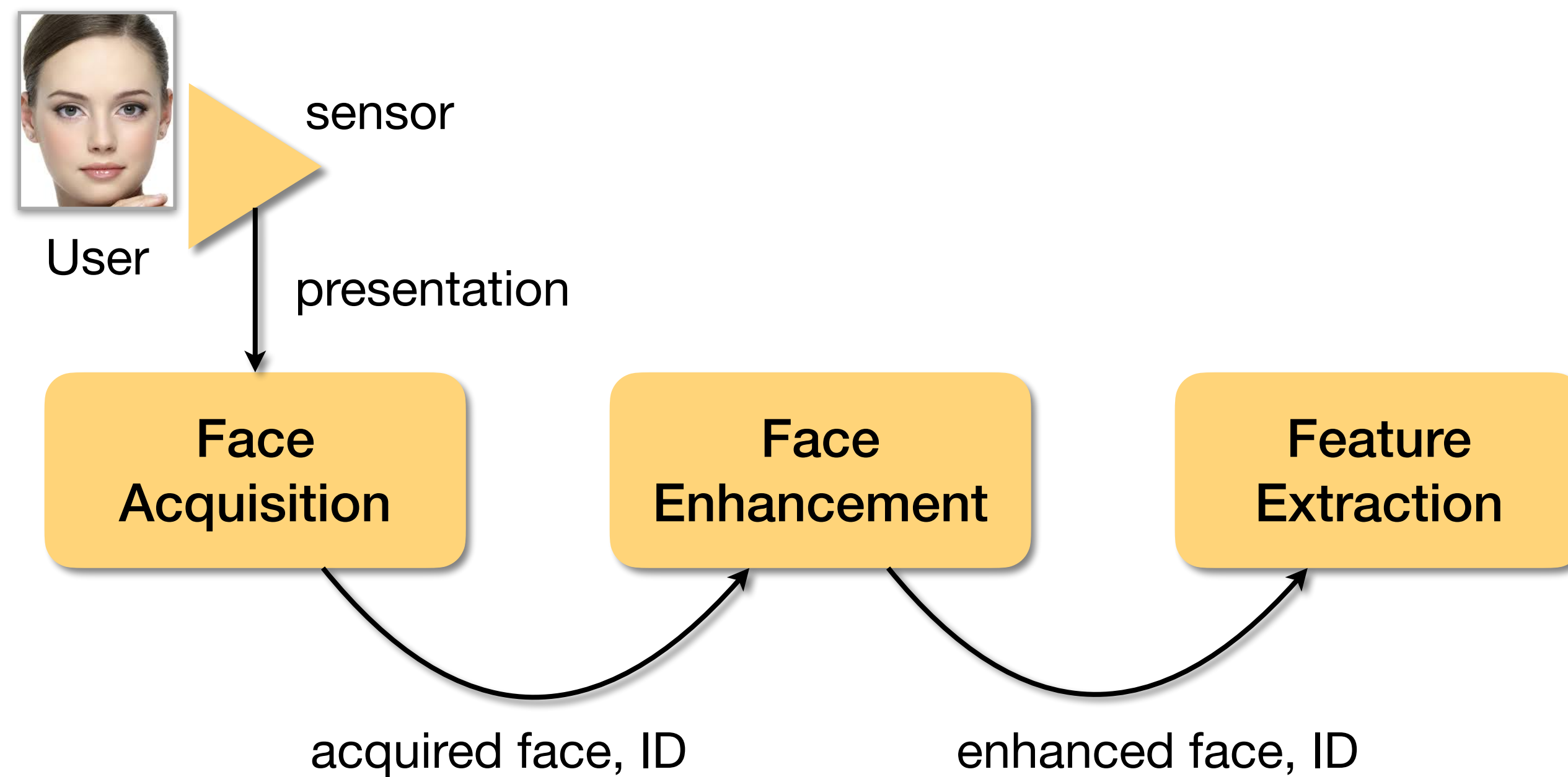
Get to know

Face description and matching.

Face Recognition



Face Recognition



Feature Extraction

Focus

2D-appearance-based methods.

Types

Handcrafted features from Computer Vision.

Data-driven learned features from Machine Learning.



Feature Extraction

Focus

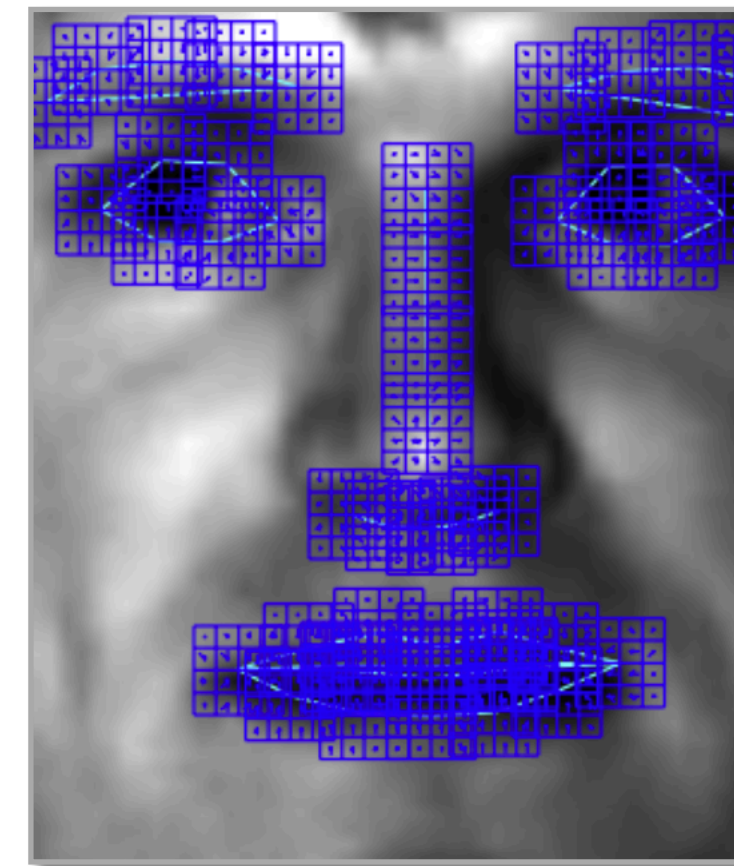
2D-appearance-based methods.

Types

Handcrafted features from Computer Vision.

Data-driven learned features from Machine Learning.

Déniz et al.
*Face recognition using
histograms of oriented gradients.*
Pattern recognition letters, 2011.



Source: Domingo Mery



Handcrafted

An expert designs what and how facial regions should be used.

Feature Extraction

Handcrafted Features

Examples

Based on Gabor filters, interest points (e.g., SIFT¹, SURF², HOG³), or texture descriptors (e.g., LBP⁴).



Geng and Jiang.
SIFT features for face recognition.
ICCSIT, 2009.

1 - Lowe. *Distinctive image features from scale-invariant keypoints*. IJCV, 2004.

2 - Bay et al. *SURF: Speeded up robust features*. ECCV, 2006.

3 - Dalal and Triggs. *Histograms of oriented gradients for human detection*. CVPR 2005.

4 - Ojala et al. *Performance evaluation of texture measures(...)*. ICPR, 1994.

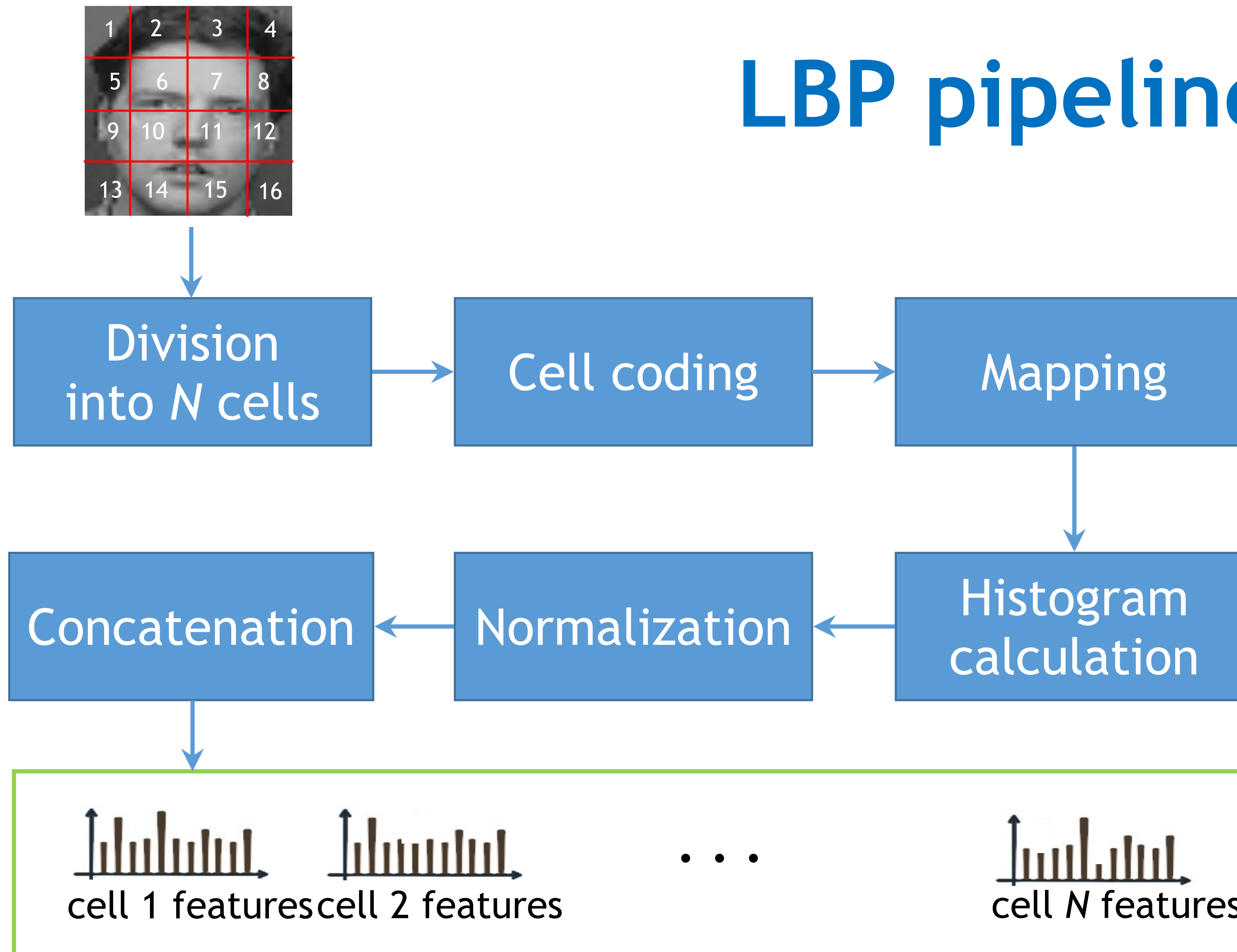
Local Binary Patterns

Selected Solution

Local Binary Patterns to describe face texture.

Next slides provided by Dr. Domingo Mery.
(<http://domingomery.ing.puc.cl/>)

LBP pipeline



Division
into N cells

Cell coding

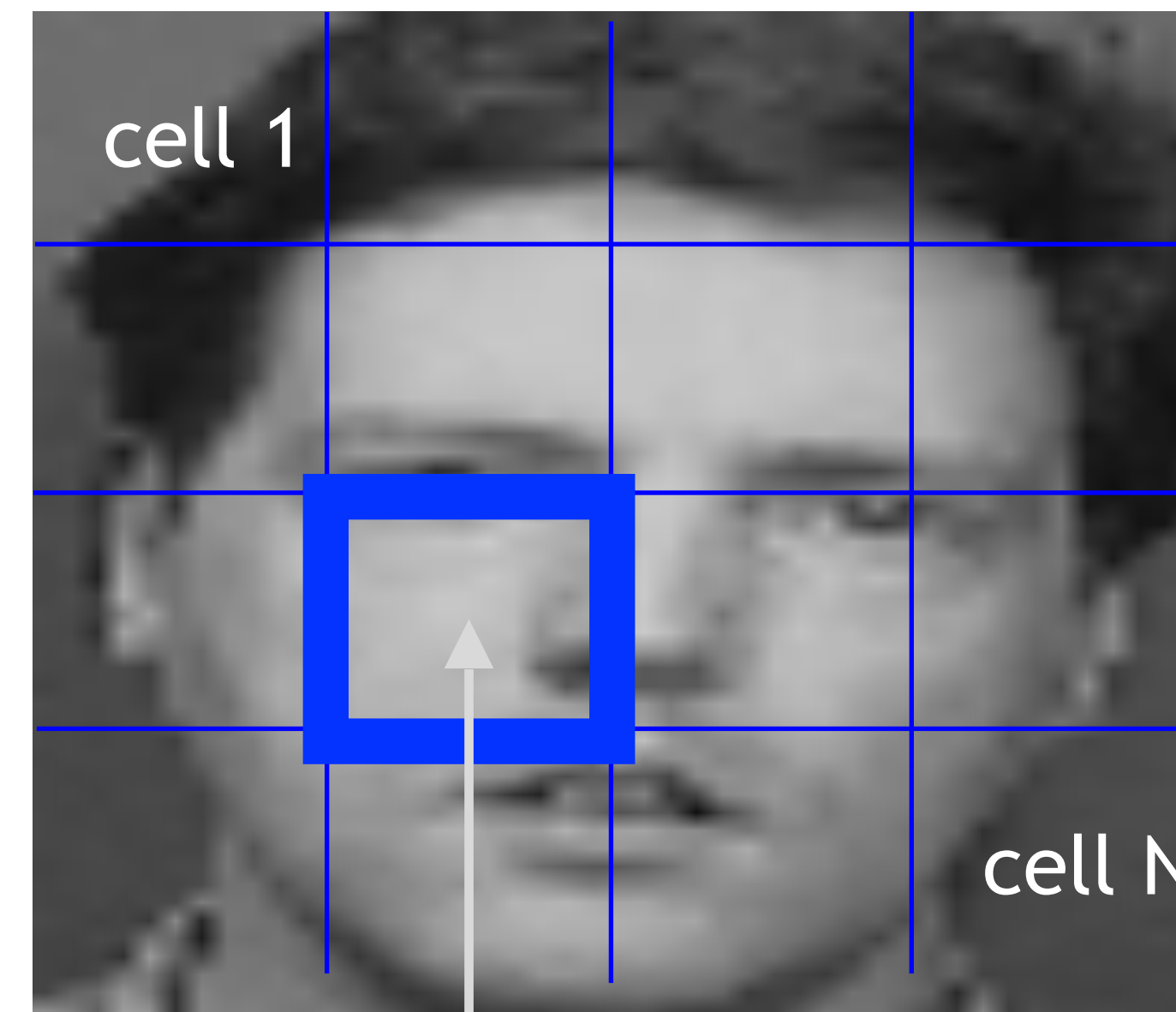
Mapping

Histogram
calculation

Normalization

Concatenation

- LBP descriptors are calculated in image sub-regions (cells)
- Number and size of cells cannot be arbitrary (note space-scale considerations)



example cell

Division
into N cells

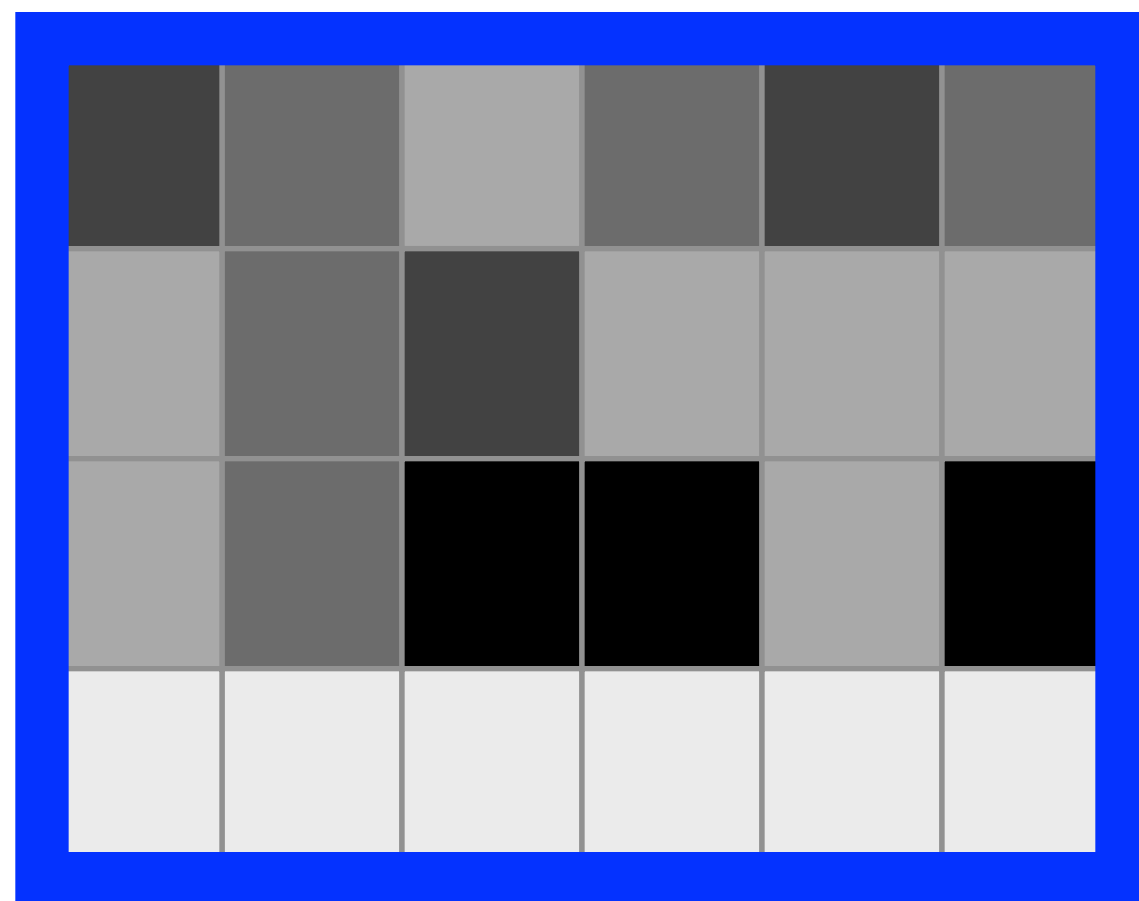
Cell coding

Mapping

Histogram
calculation

Normalization

Concatenation



Division
into N cells

Cell coding

Mapping

Histogram
calculation

Normalization

Concatenation

4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

Division
into N cells

Cell coding

Mapping

Histogram
calculation

Normalization

Concatenation

4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

Division
into N cells

Cell coding

Mapping

Histogram
calculation

Normalization

Concatenation

4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

<		



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: $\geq$

0		



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: ≥

0	≥	



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: ≥

0	1	



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: ≥

0	1	1



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: $\geq$

0	1	1
		0



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: $\geq$

0	1	1
		0
		0



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: $\geq$

0	1	1
		0
	1	0



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: $\geq$

0	1	1
		0
1	1	0



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: $\geq$

0	1	1
1		0
1	1	0



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: ≥

0	1	1
1		0
1	1	0

x

1	2	4
128	+	8
64	32	16

25

$= 0 + 2 + 4 + 0 + 0 + 32 + 64 + 128 = 230$



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10

4	6	9
9	6	4
9	6	2

0: <
1: ≥

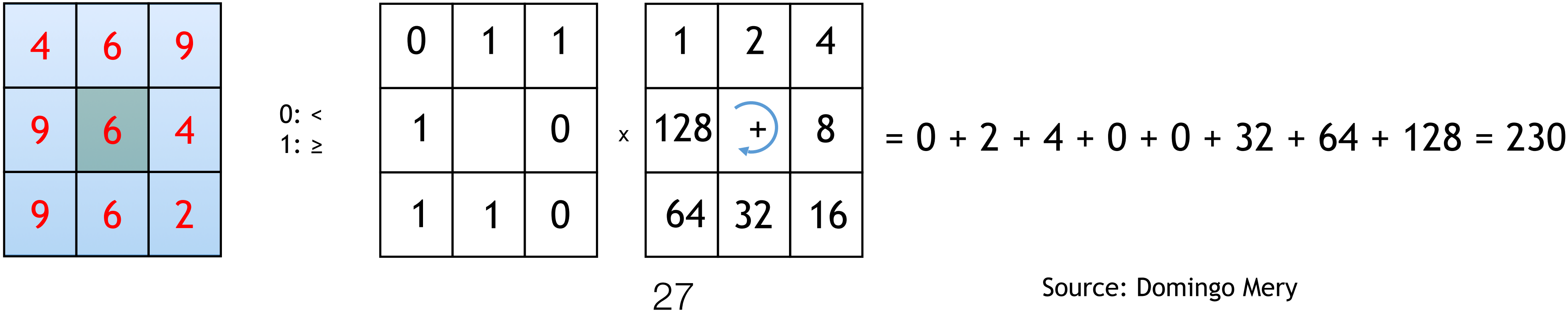
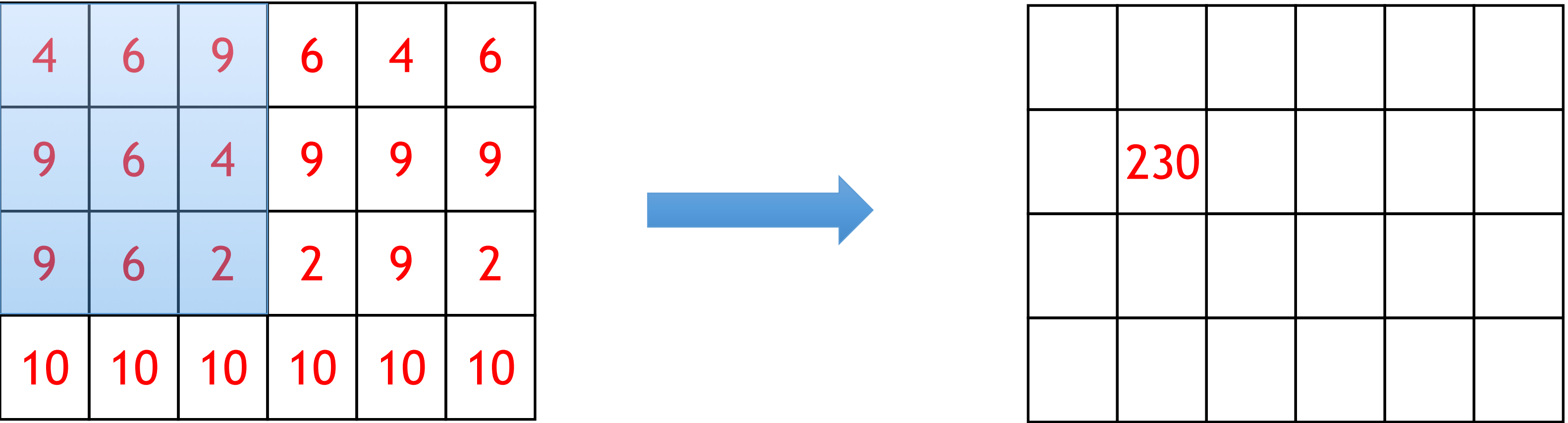
0	1	1
1		0
1	1	0

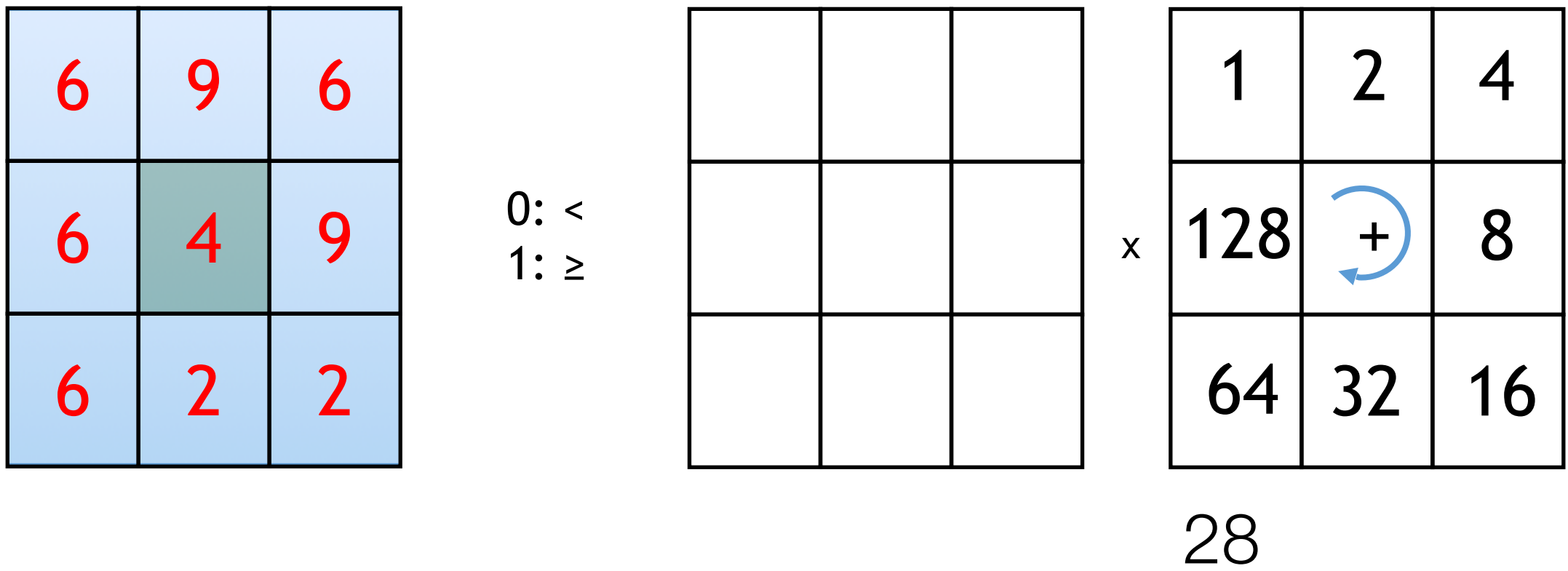
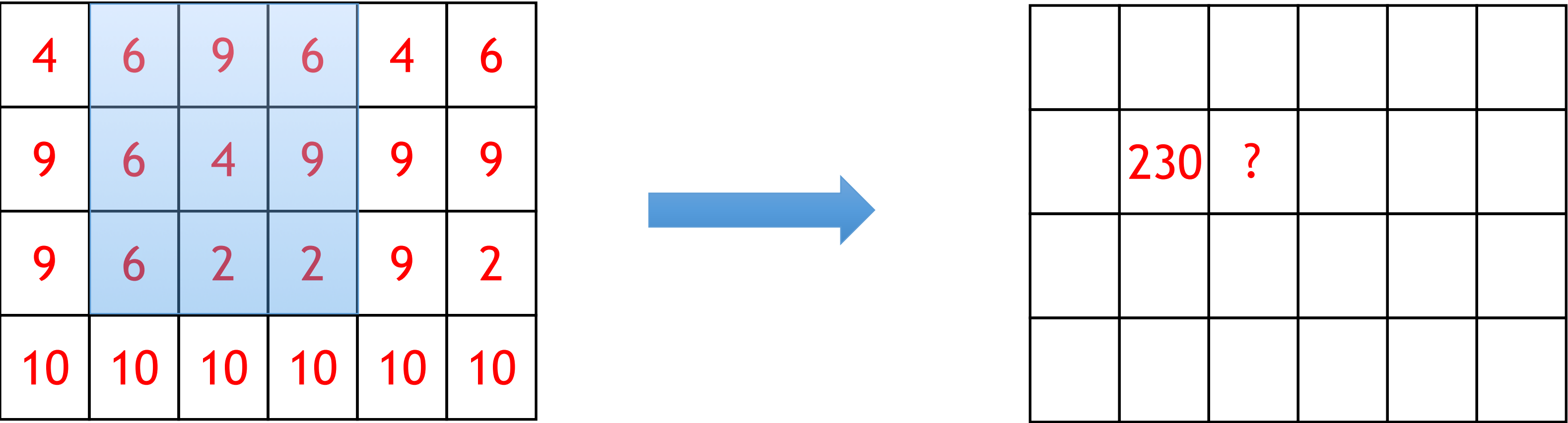
x

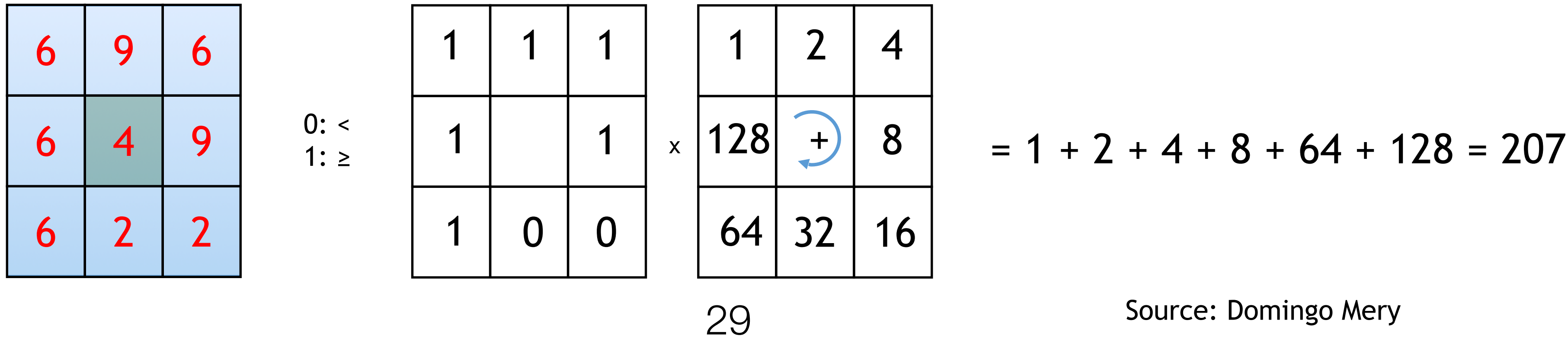
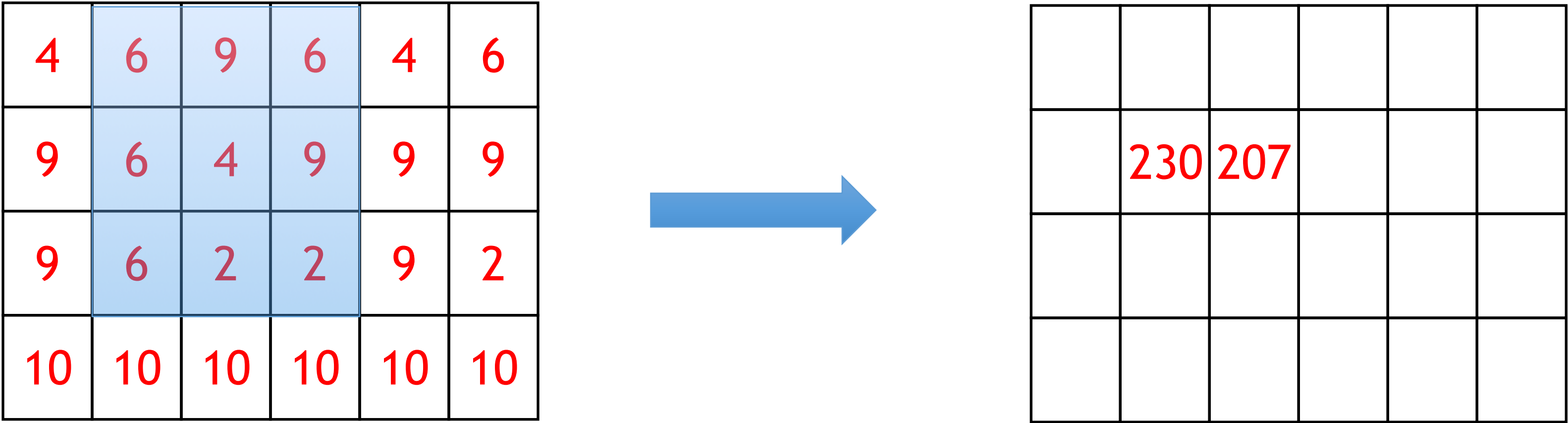
1	2	4
128	+	8
64	32	16

26

$= 0 + 2 + 4 + 0 + 0 + 32 + 64 + 128 = 230$





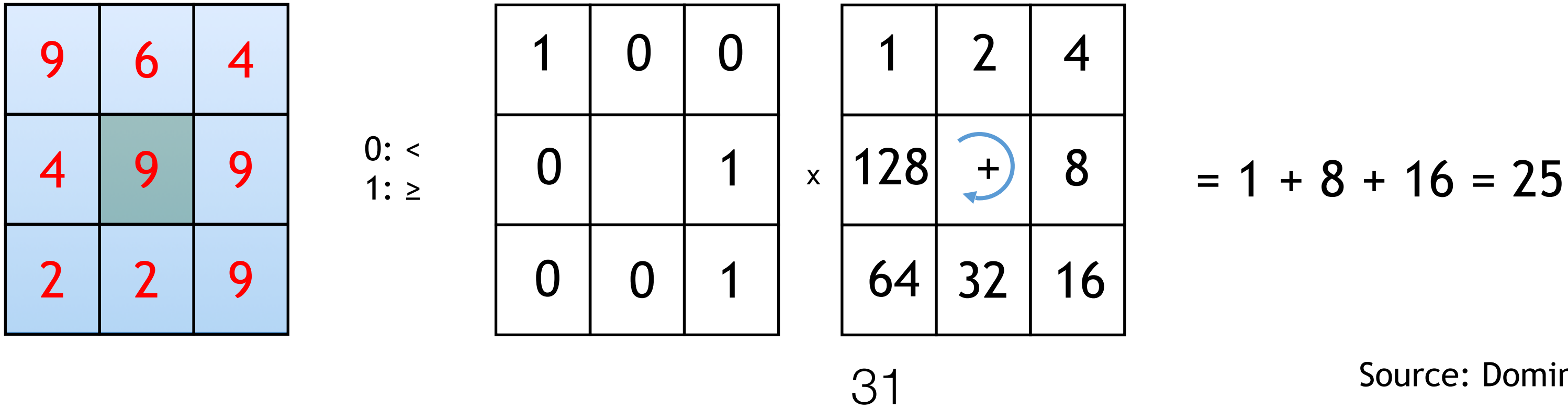
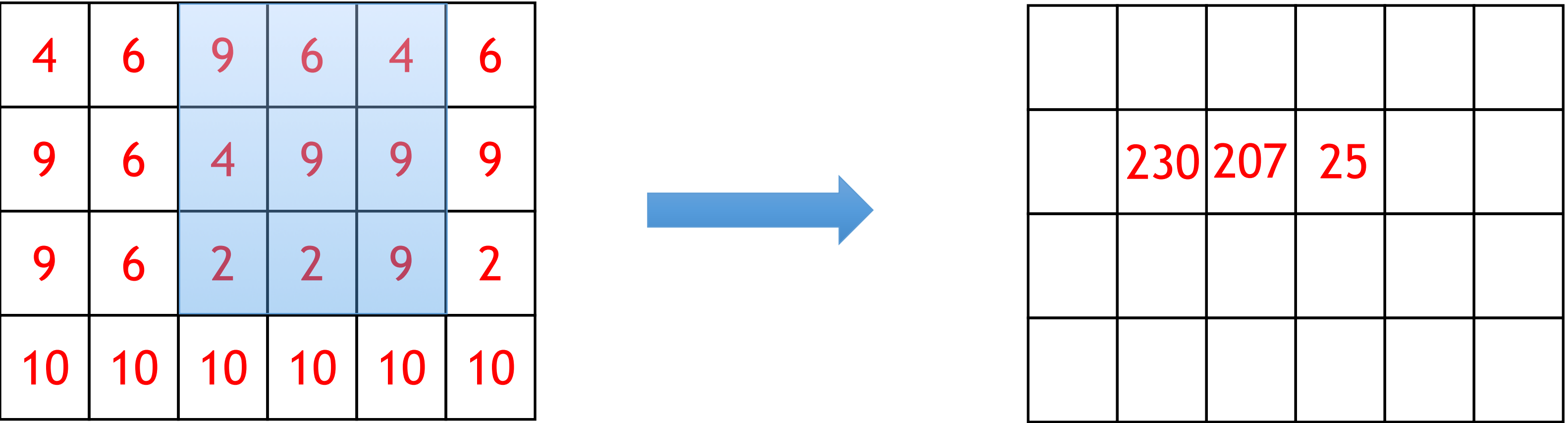




4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



	230	207	?		

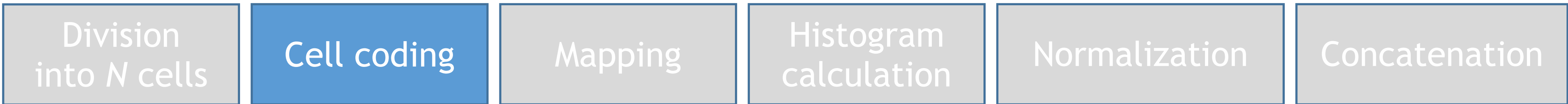




4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



	230	207	25	168	



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



	230	207	25	168	
	243				



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



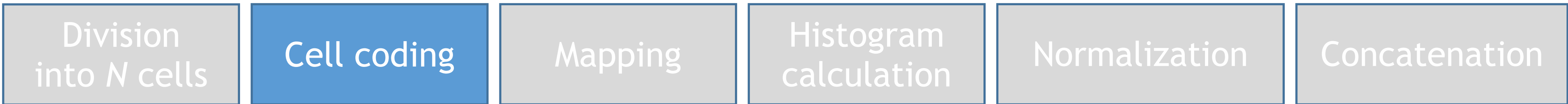
	230	207	25	168	
	243	255			



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



	230	207	25	168	
	243	255	255		



4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



	230	207	25	168	
	243	255	255	119	

Division
into N cells

Cell coding

Mapping

Histogram
calculation

Normalization

Concatenation

Note on neighborhood definition

- Original algorithm uses 3x3 pixel neighborhood
- Further extensions (Ojala, 2002) introduced arbitrary neighborhood with interpolation

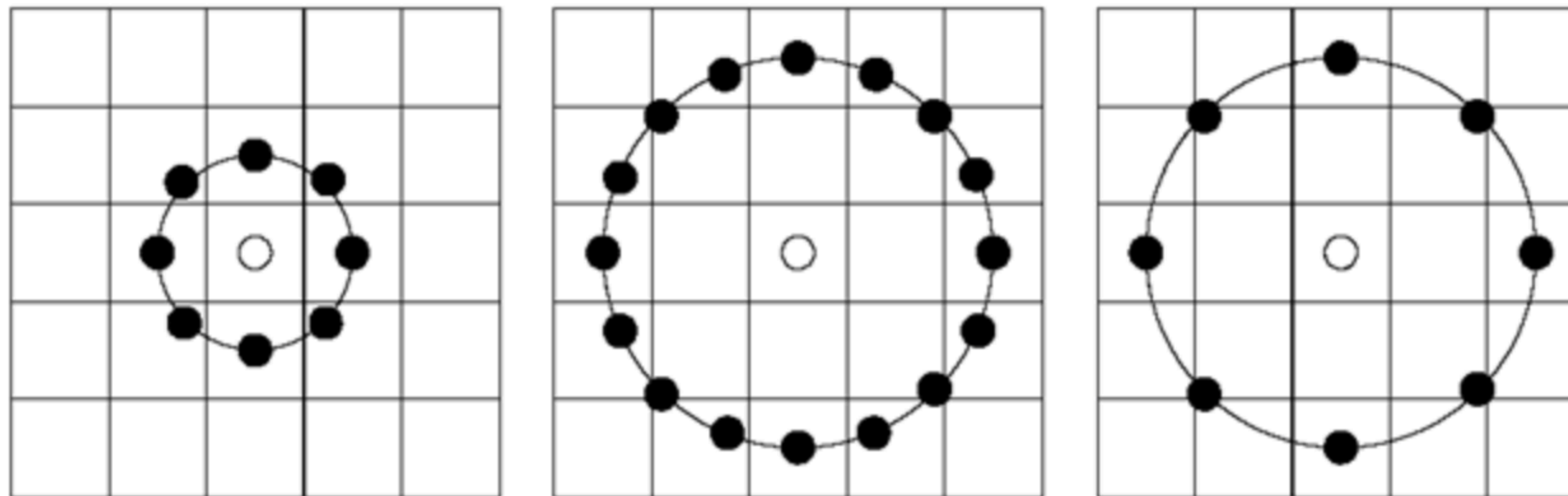


Image source: <http://what-when-how.com/face-recognition/local-representation-of-facial-features-face-image-modeling-and-representation-face-recognition-part-1/>

Division
into N cells

Cell coding

Mapping

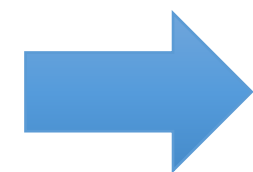
Histogram
calculation

Normalization

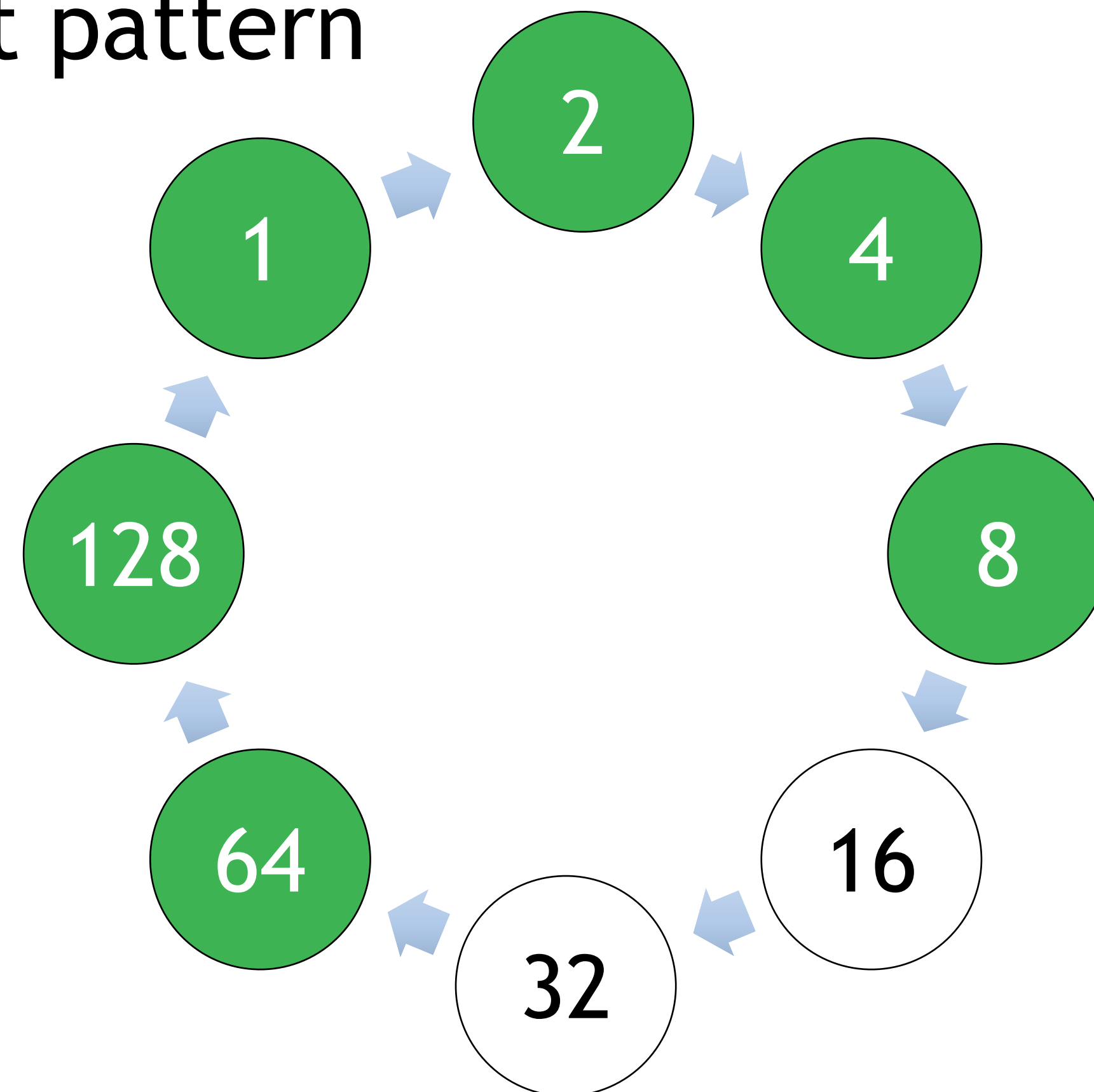
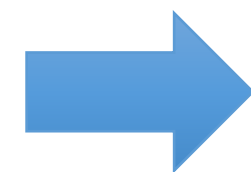
Concatenation

Uniform pattern: contains at most two bitwise transitions (U) from 0 to 1 (or vice versa) when the bit pattern is traversed circularly

6	9	6
6	4	9
6	2	2



1	1	1
1		1
1	0	0



Division
into N cells

Cell coding

Mapping

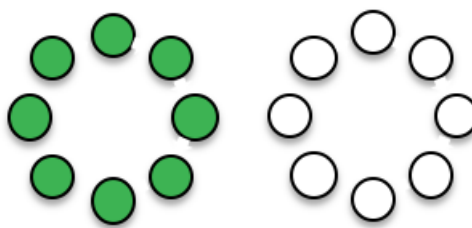
Histogram
calculation

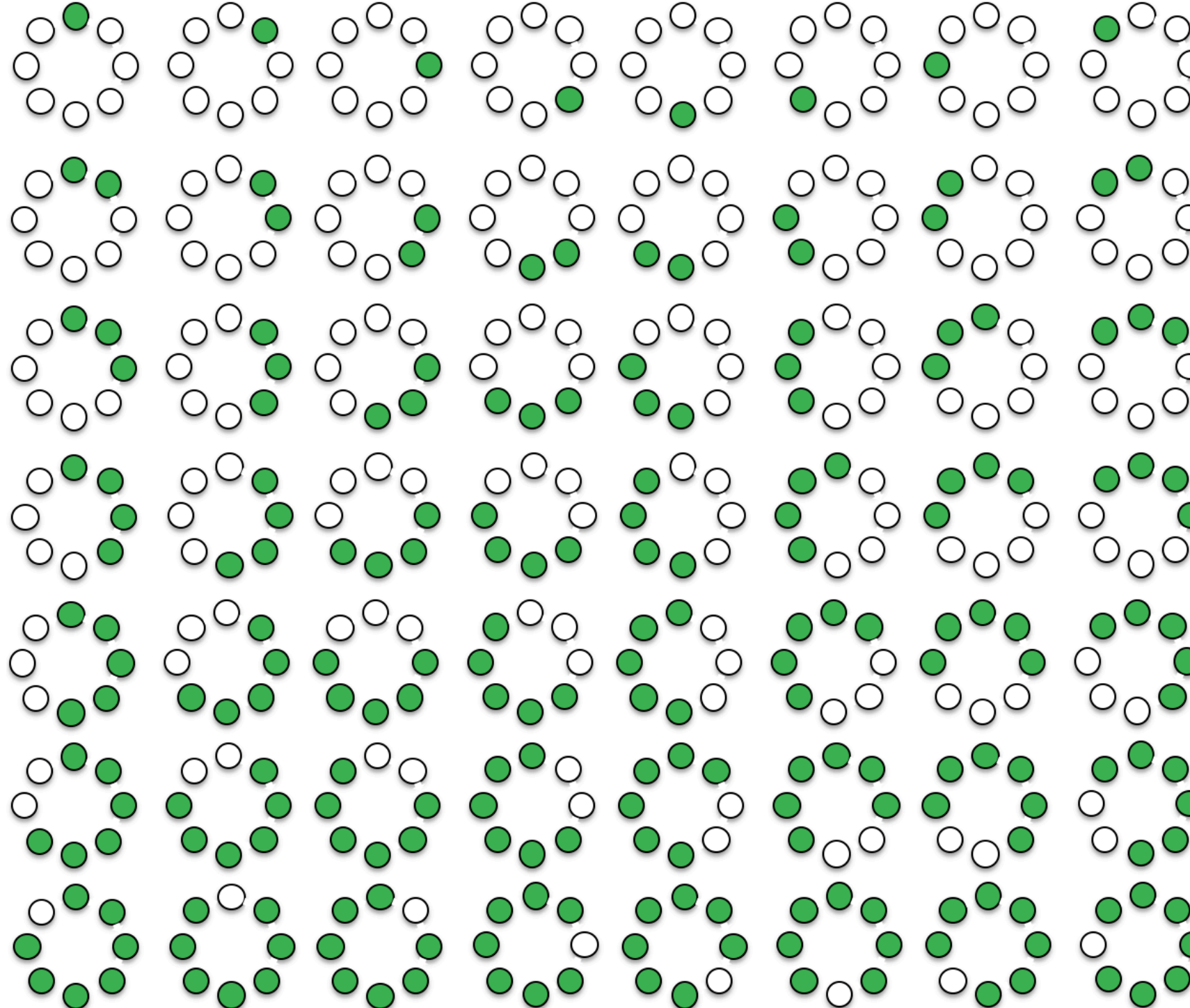
Normalization

Concatenation

Uniform patterns

Uniform patterns
account for almost
90% of all patterns.

$$U = 0$$


$$U = 2$$


Division
into N cells

Cell coding

Mapping

Histogram
calculation

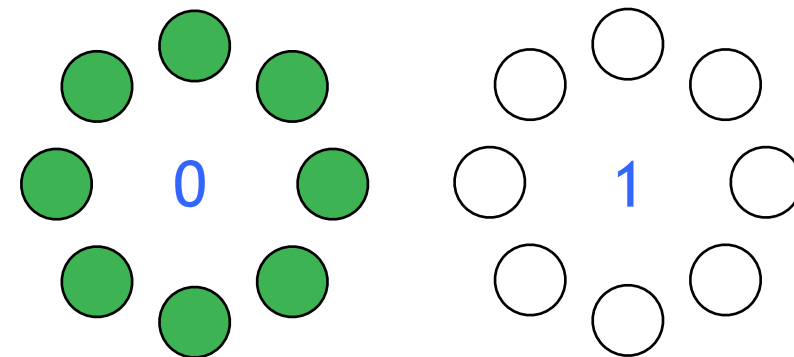
Normalization

Concatenation

$U = 0$

2
patterns

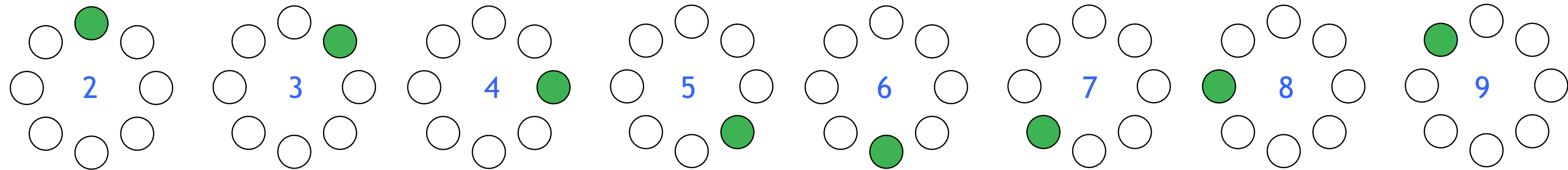
$\{0, 1\}$



$U = 2$

$8 \times 7 = 56$
patterns

$\{2, 3, \dots, 57\}$



Uniform patterns

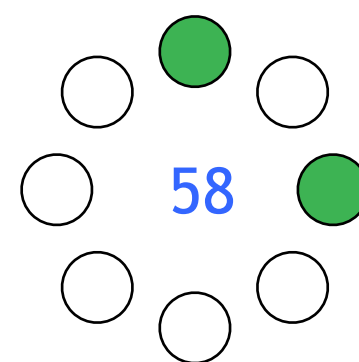
$2 + 56 = 58$
patterns

Non-uniform patterns

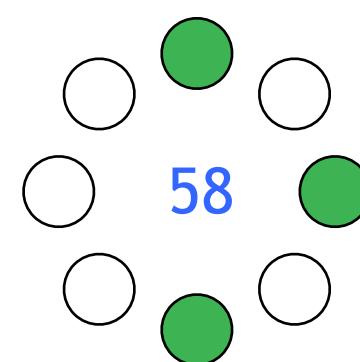
$256 - 58 = 198$
patterns

$\{58\}$

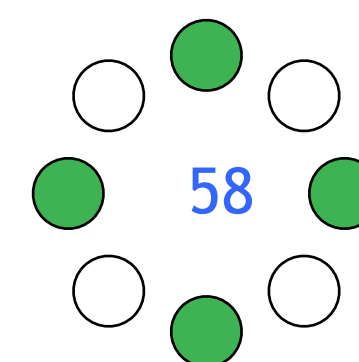
$U = 4$



$U = 6$

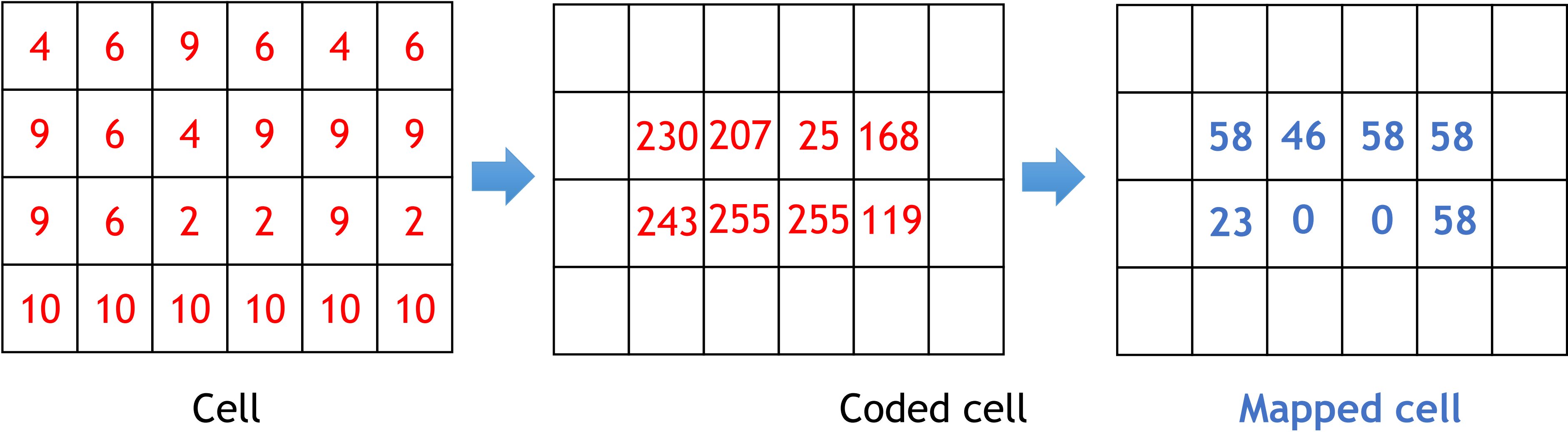


$U = 8$





Result of cell code mapping



Division
into N cells

Cell coding

Mapping

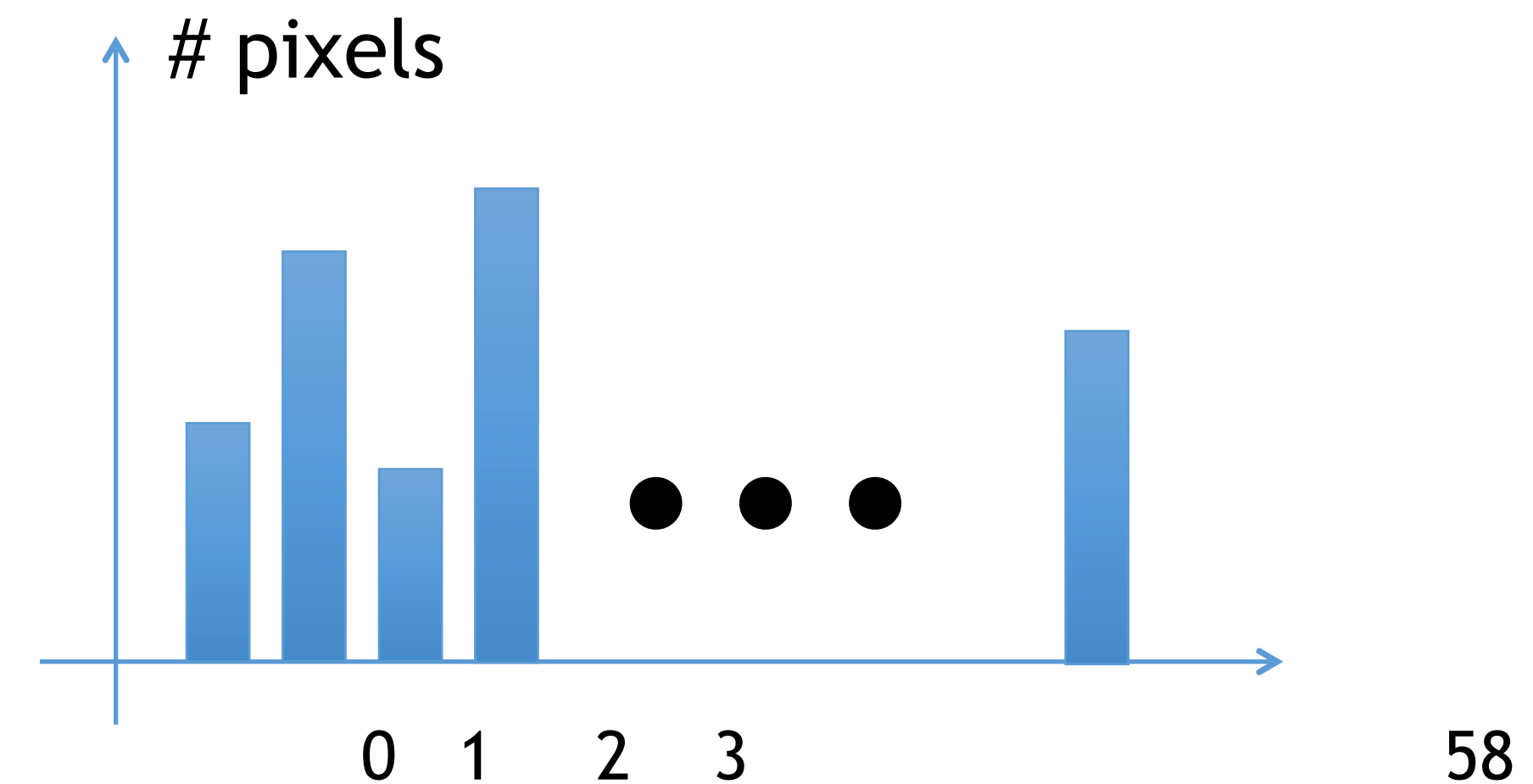
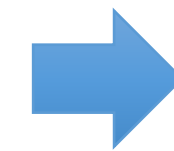
Histogram
calculation

Normalization

Concatenation

Mapped cell

	58	46	58	58	
	23	0	0	58	



- Each cell is represented as 59-digit LBP descriptor
- Similar textures have similar histograms.

Division
into N cells

Cell coding

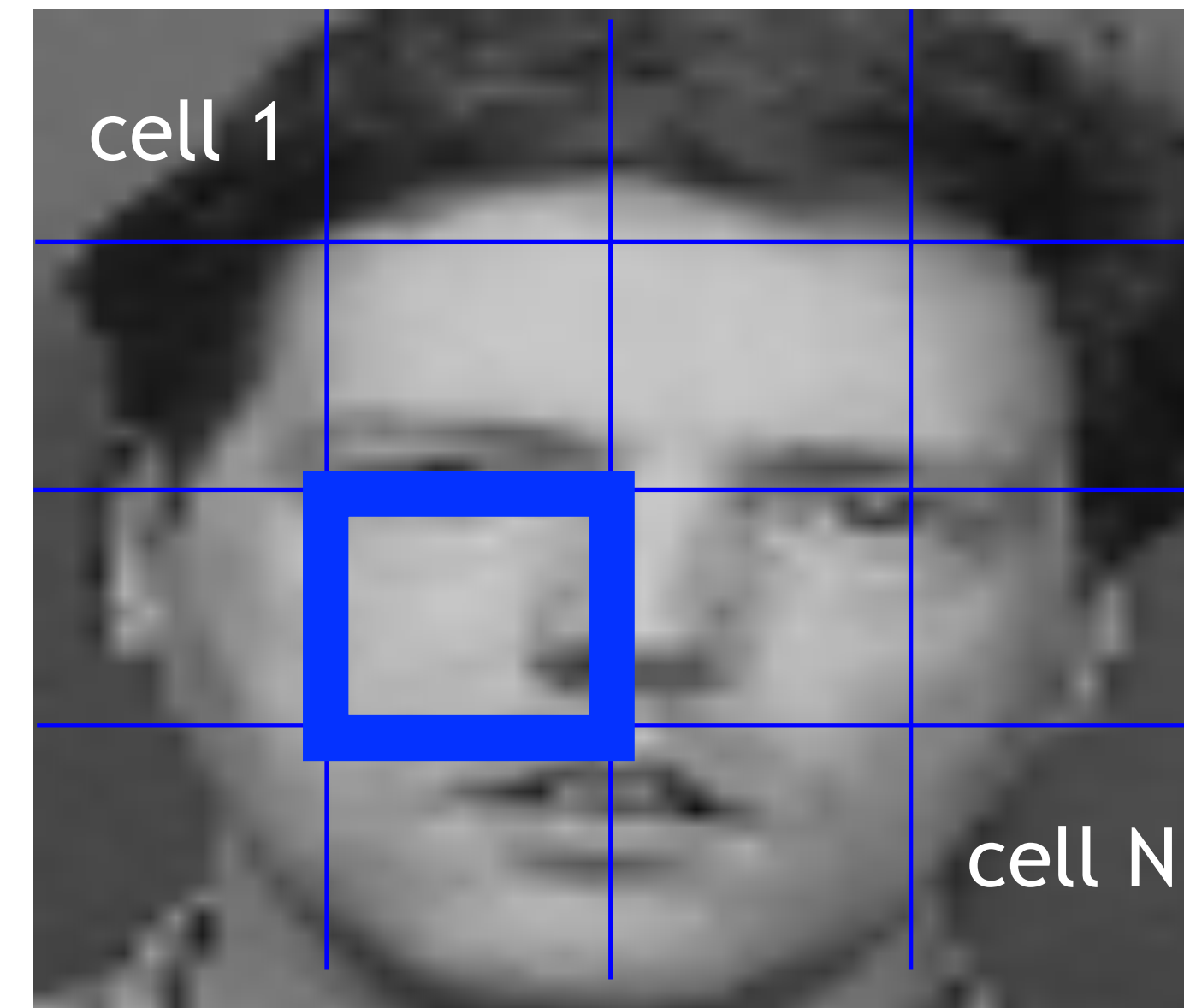
Mapping

Histogram
calculation

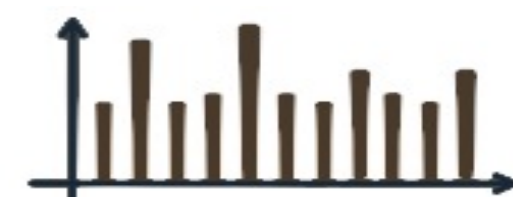
Normalization

Concatenation

- Normalization of histograms makes LBP descriptors **size-invariant**
- **Concatenation** of all cell histograms provides the image LBP descriptor



59 numbers



cell 1 features

59 numbers



...

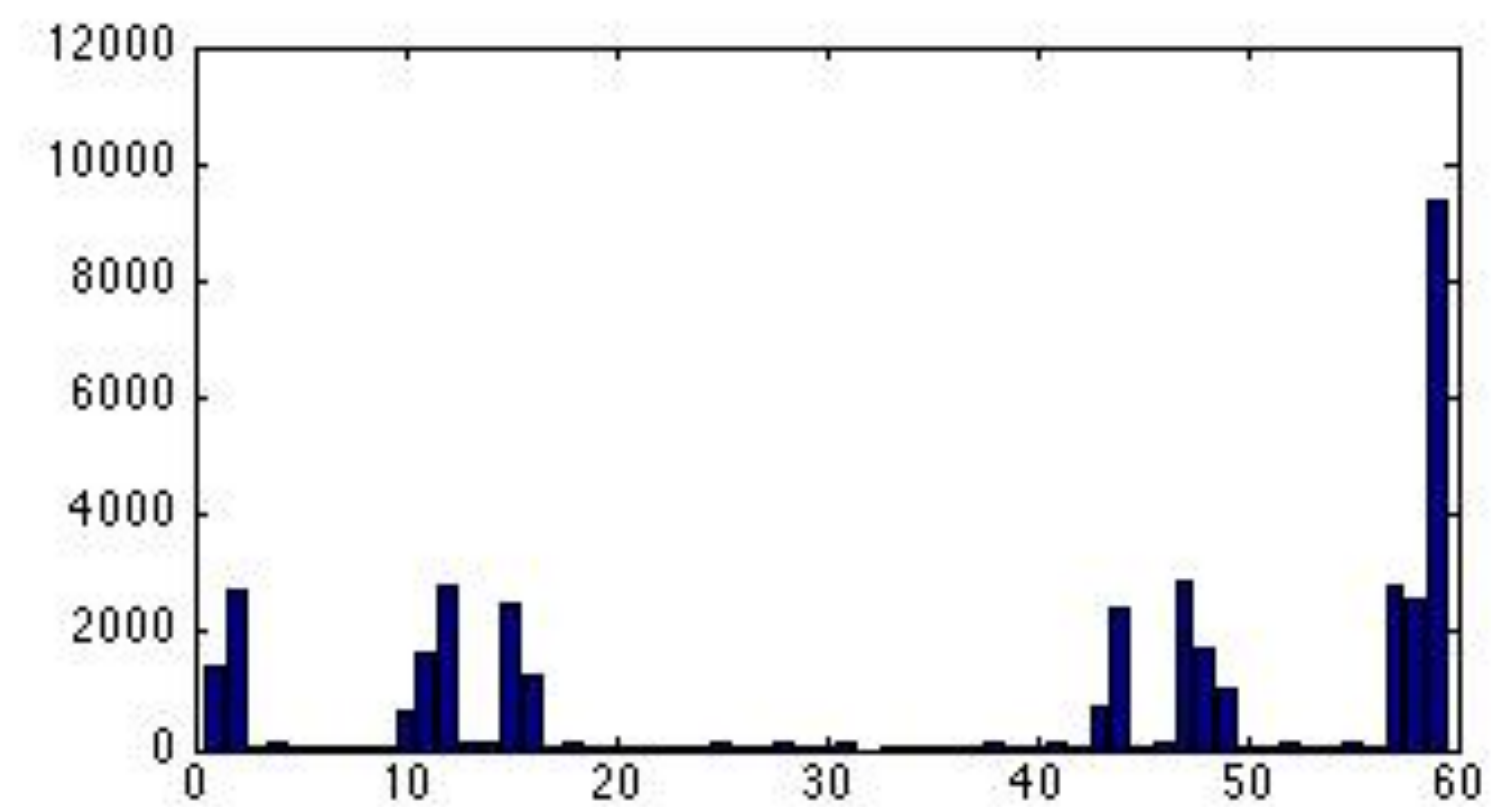
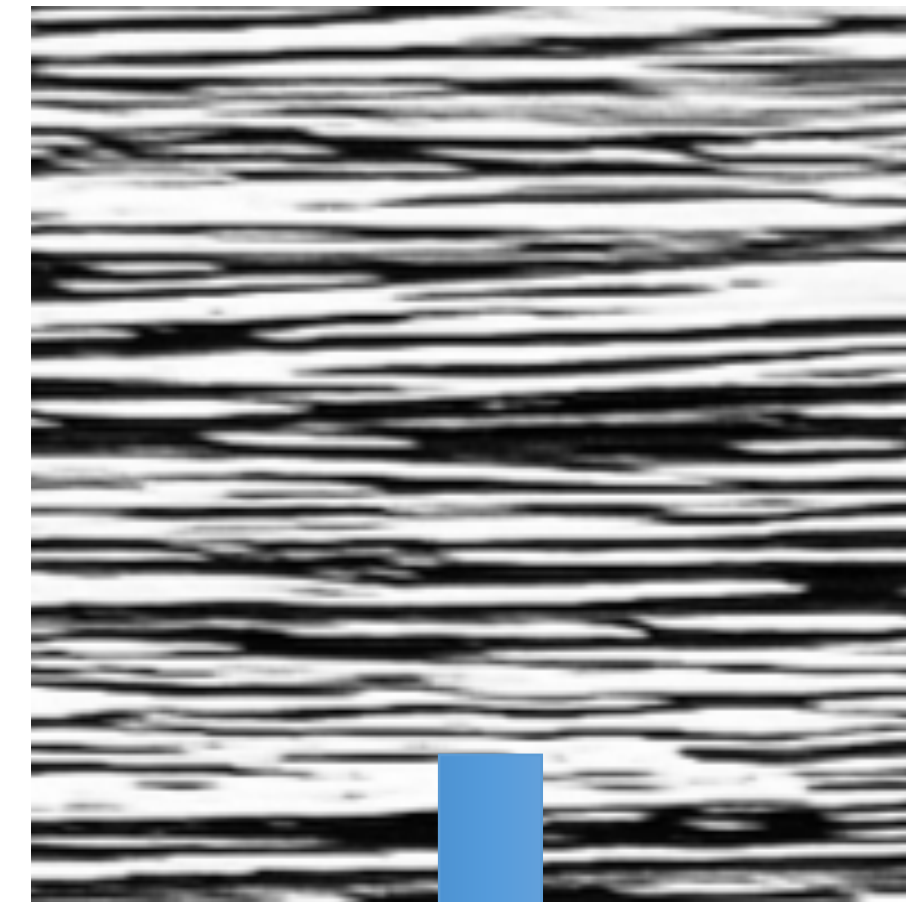
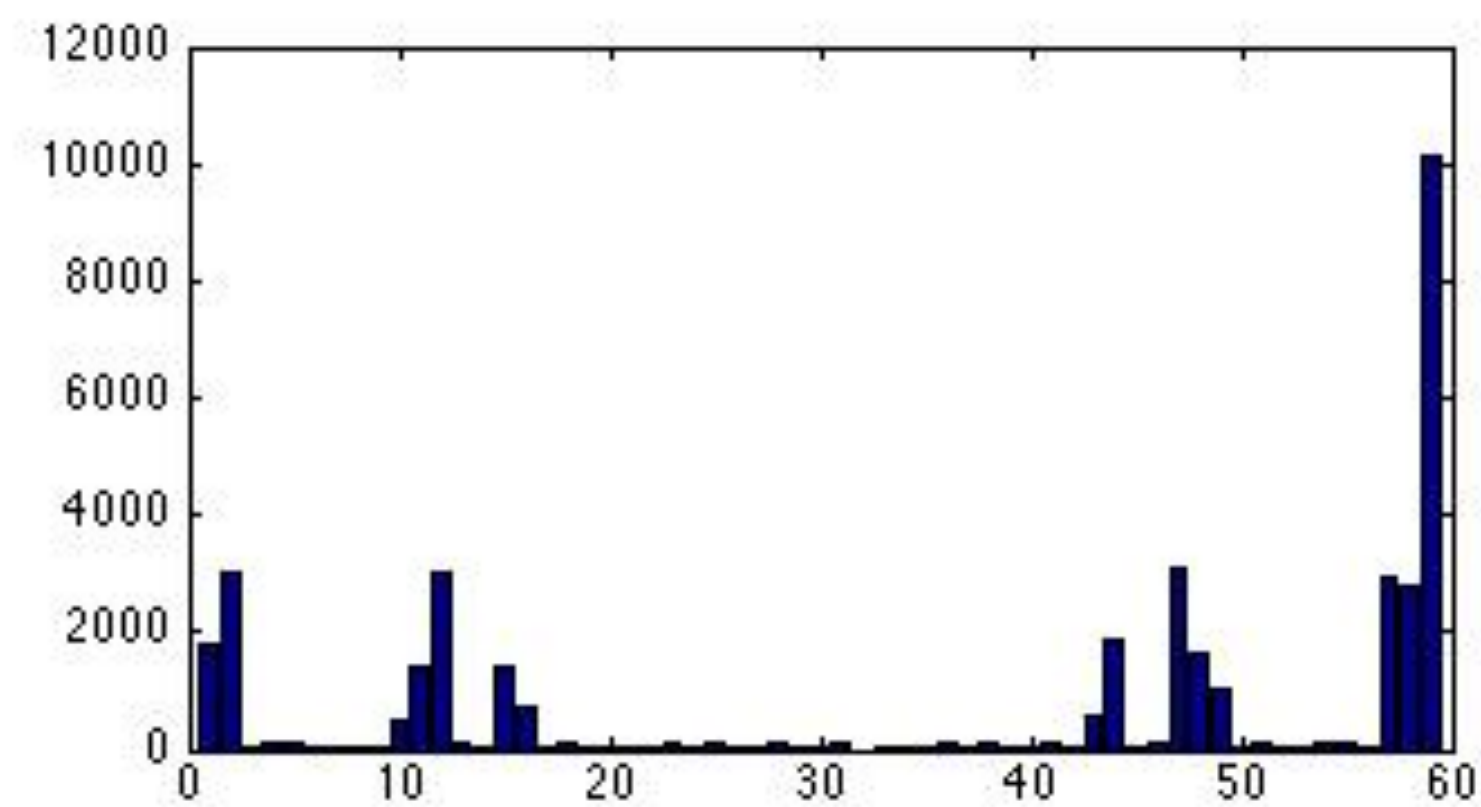
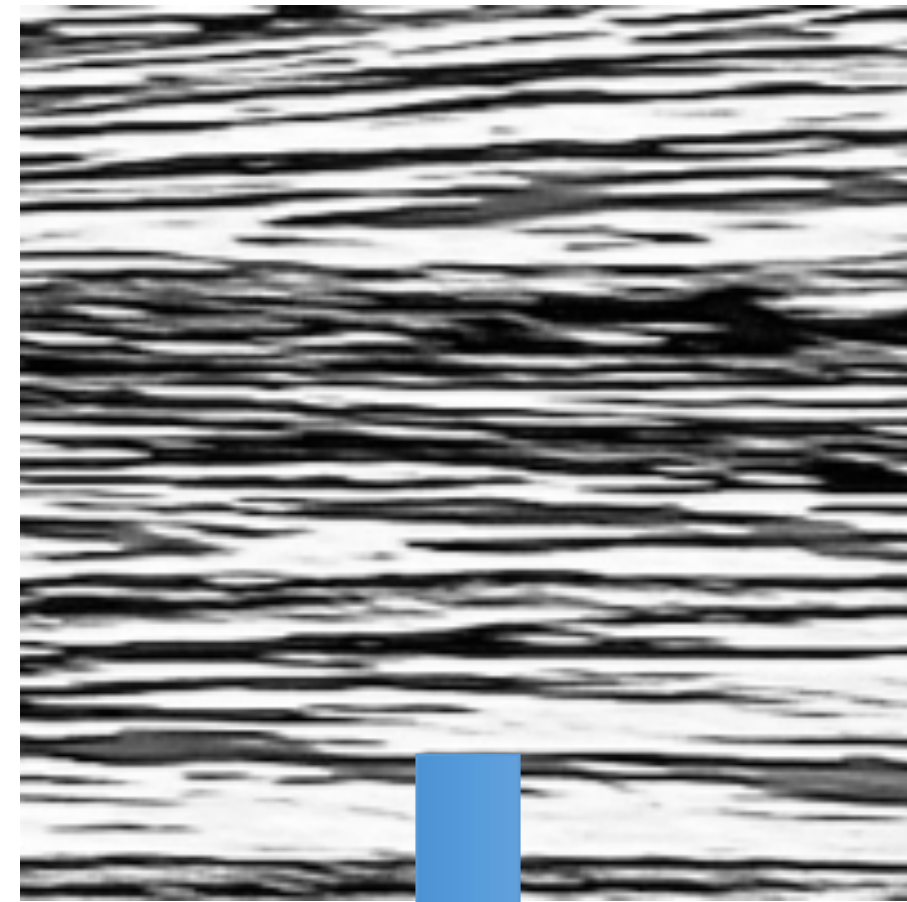
59 numbers



cell N features

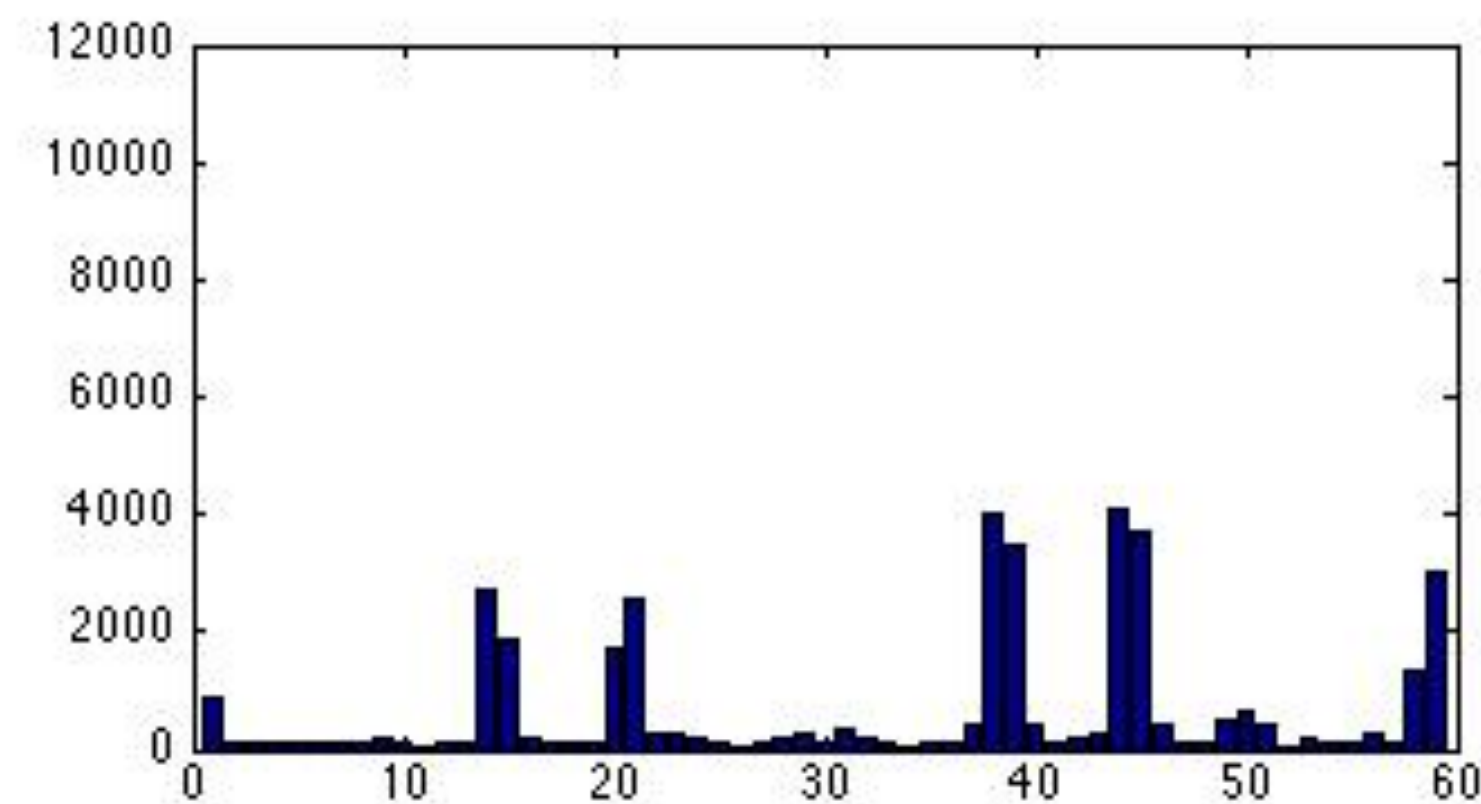
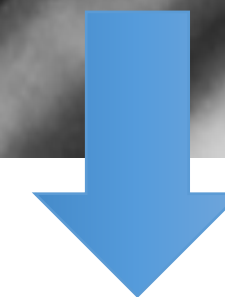
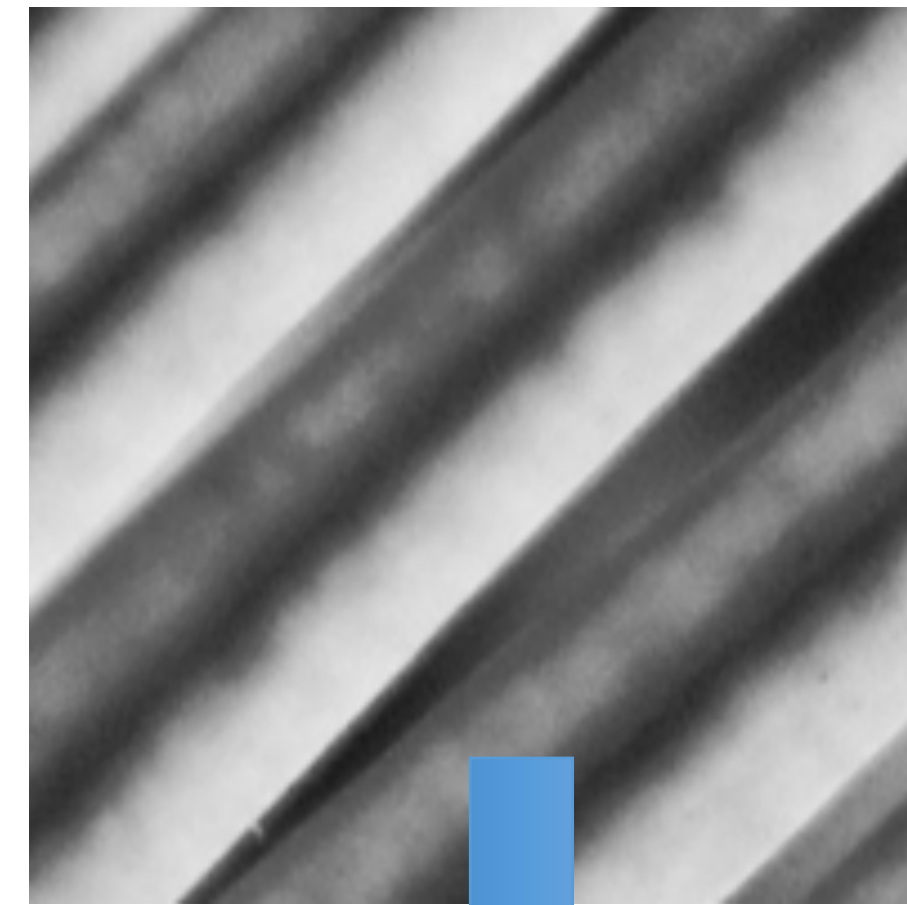
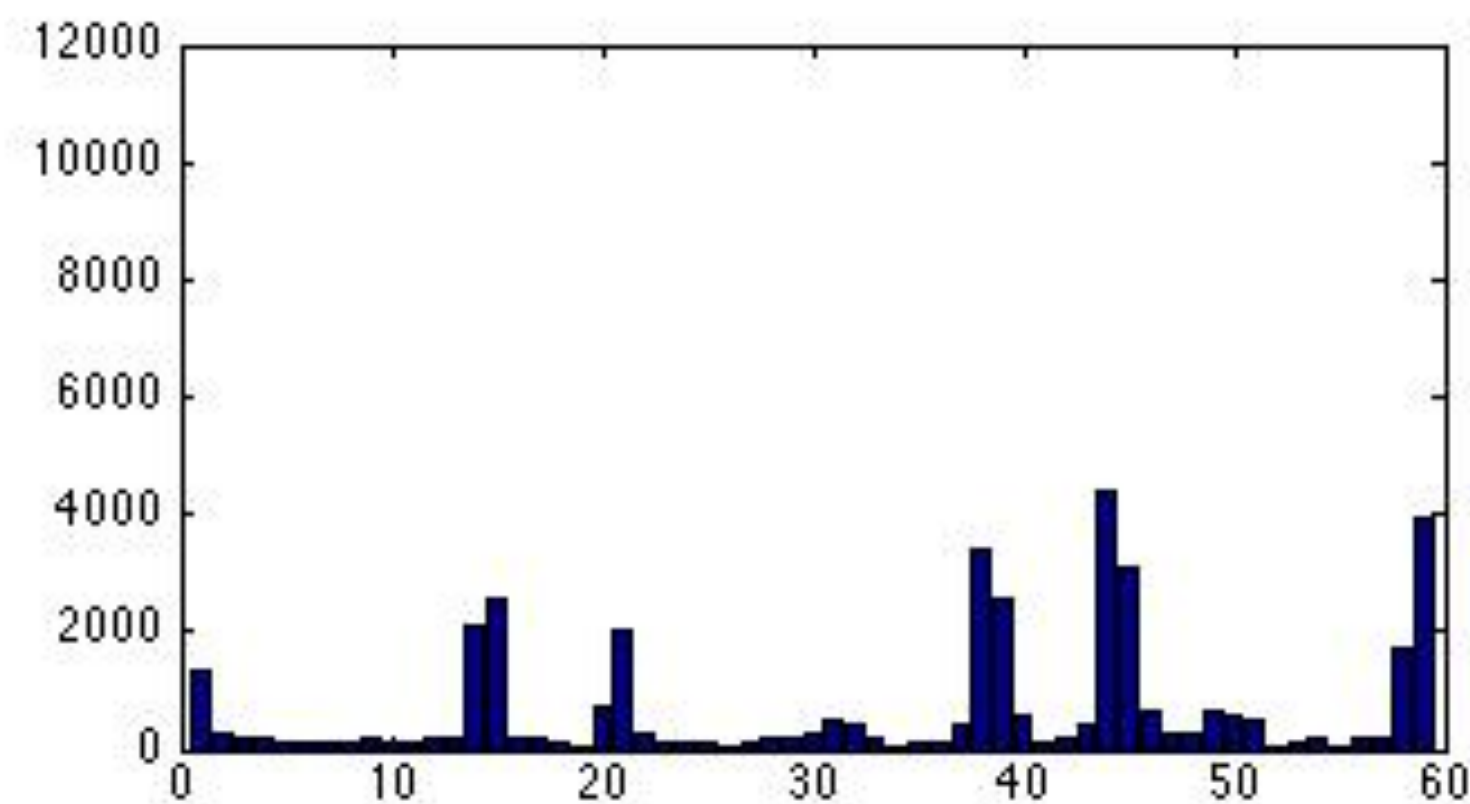
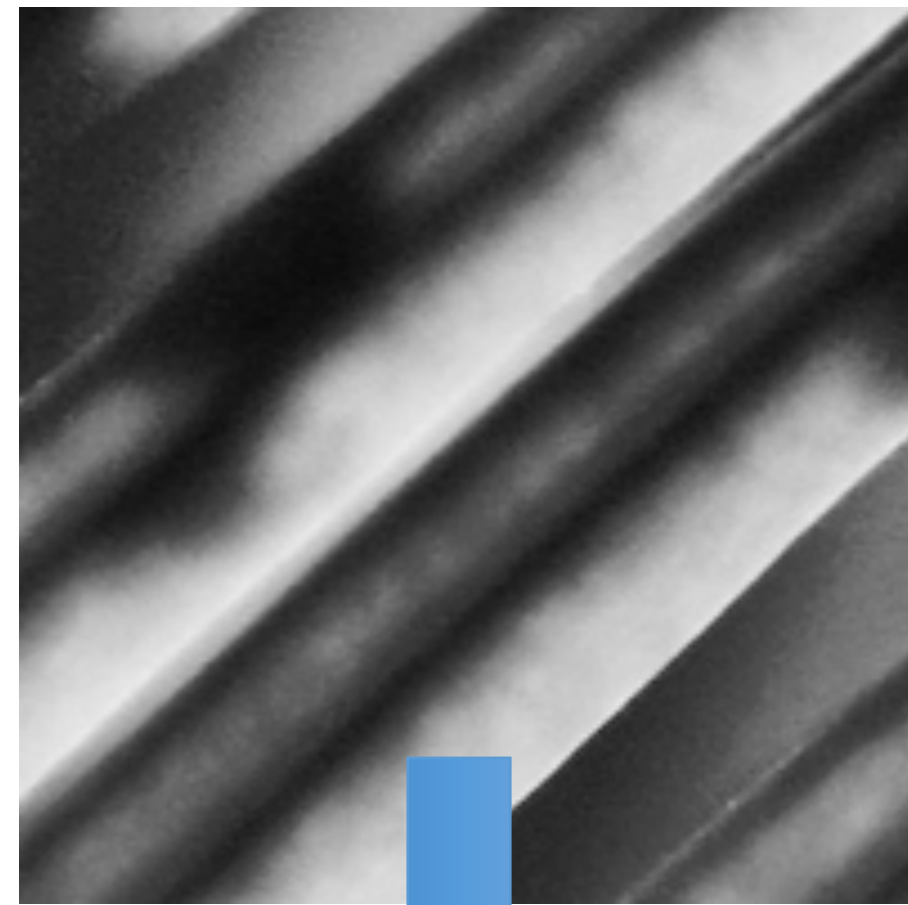
Local Binary Patterns

Similar textures have similar histograms



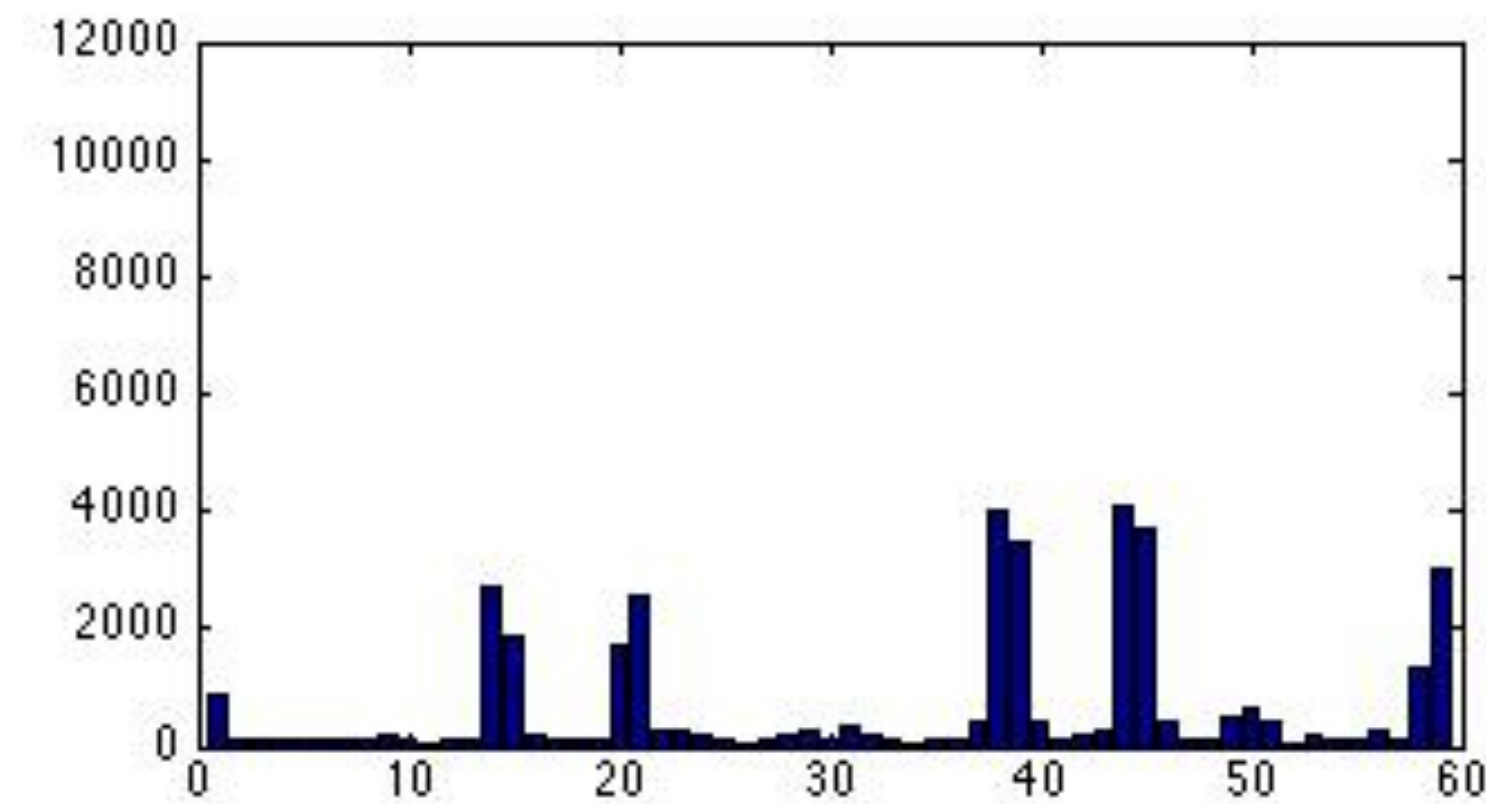
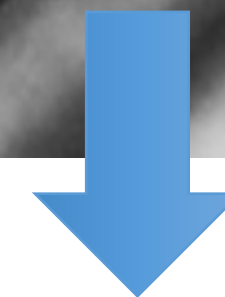
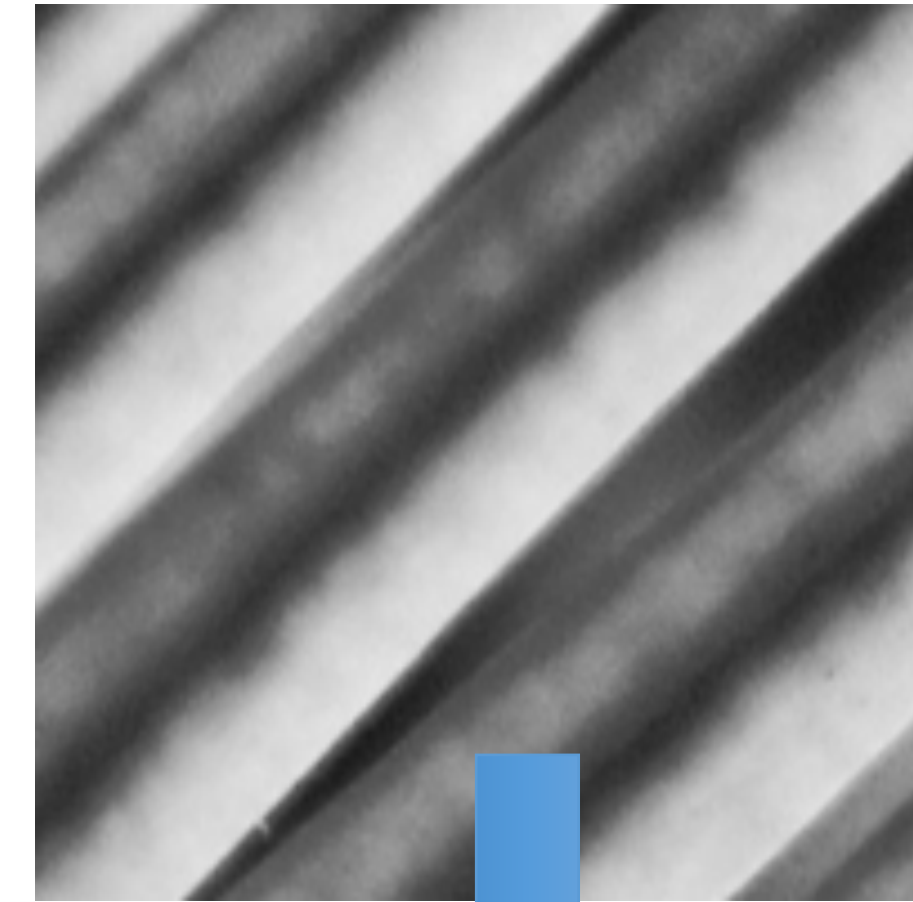
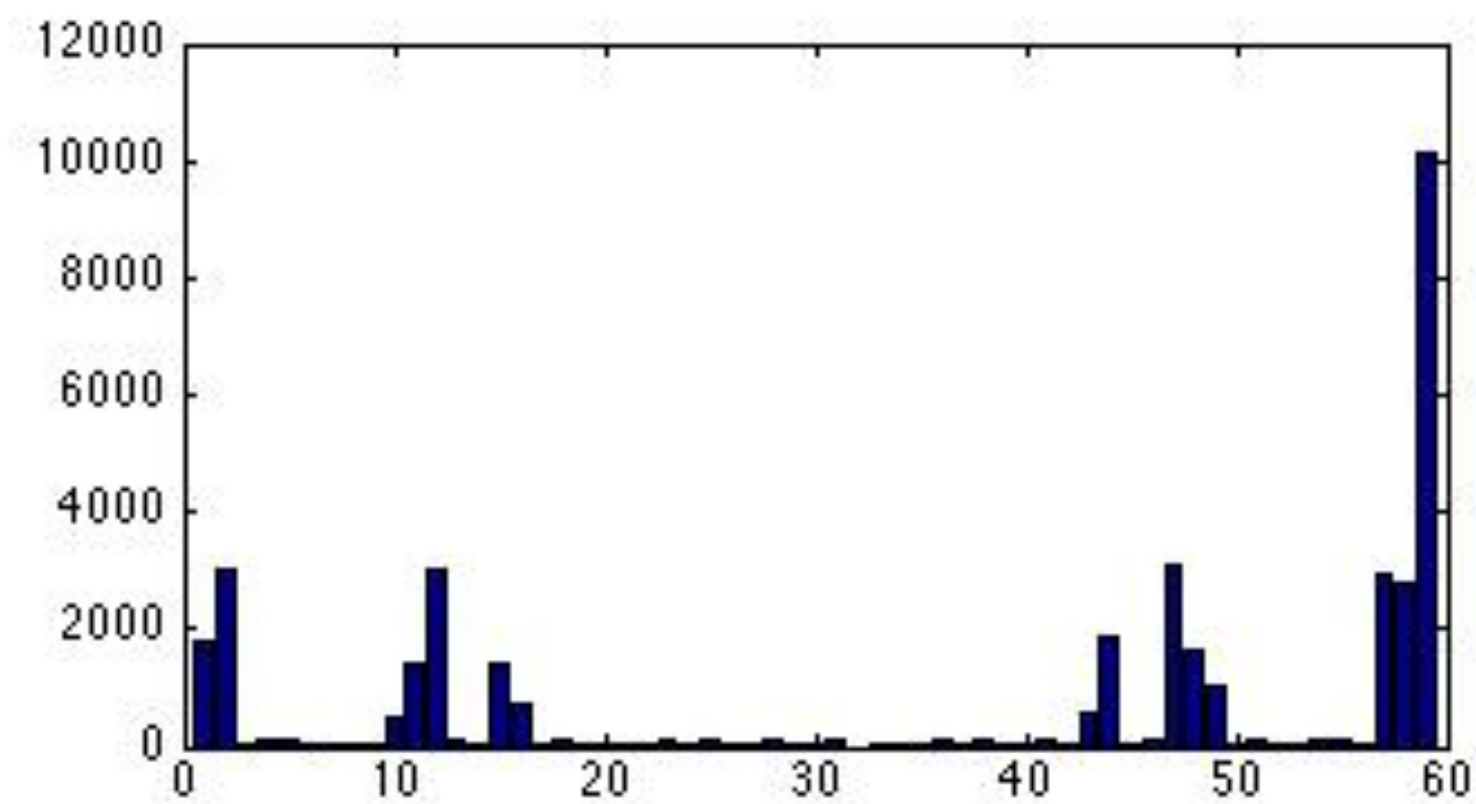
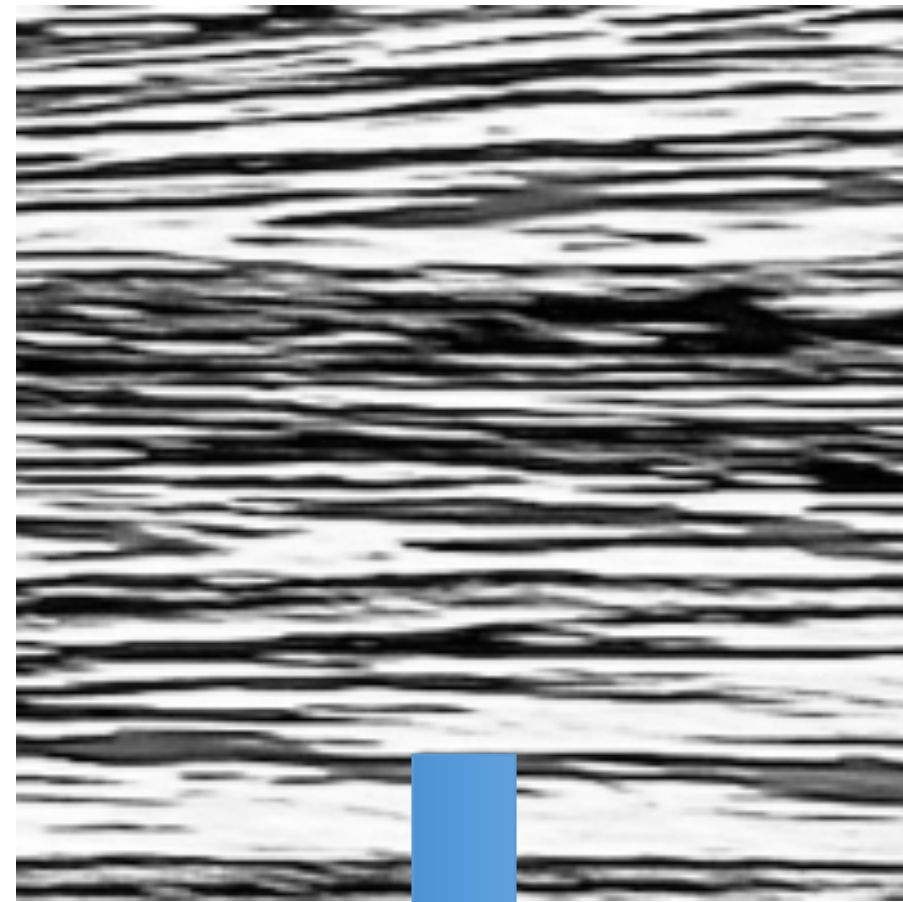
Local Binary Patterns

Similar textures have similar histograms



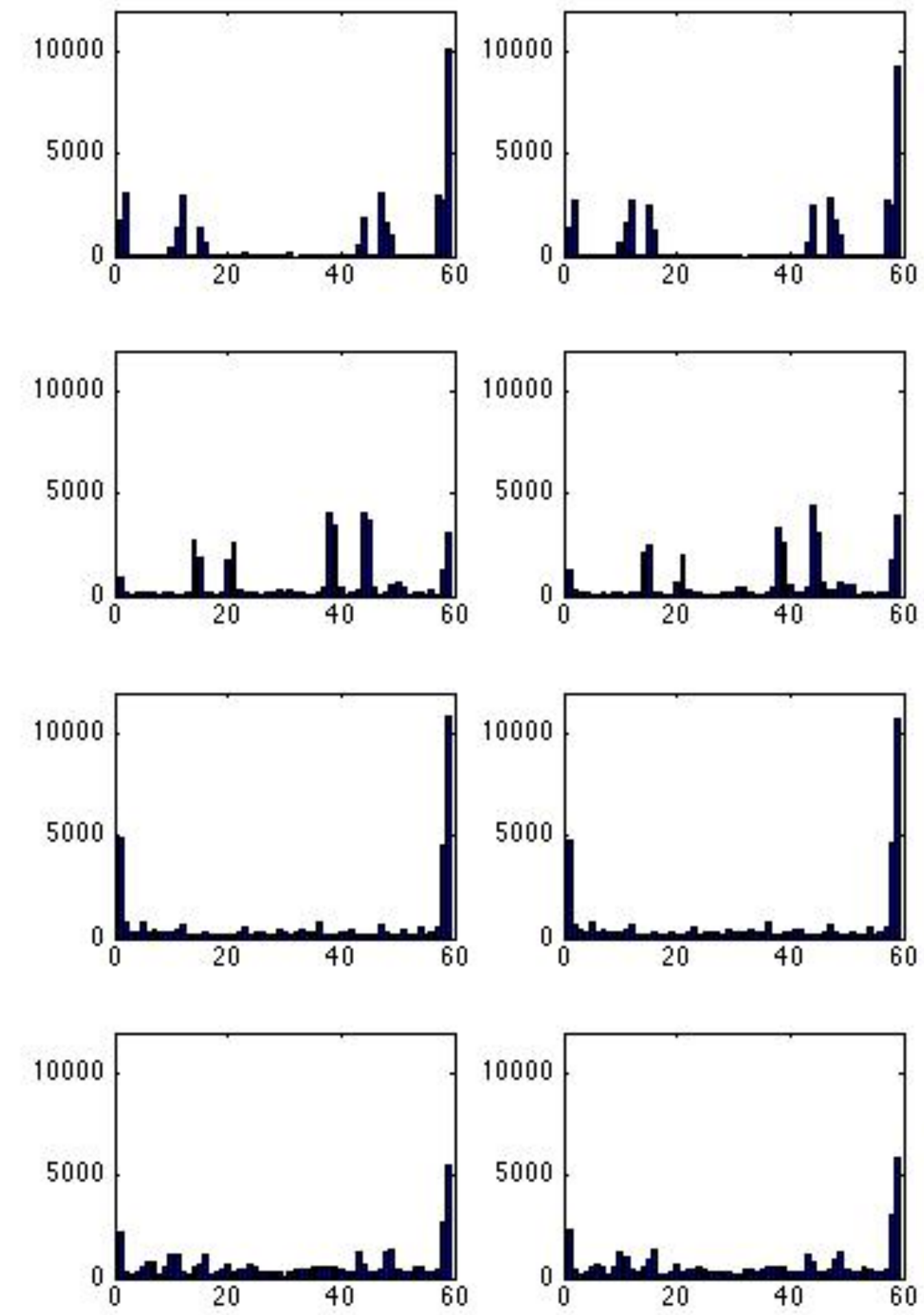
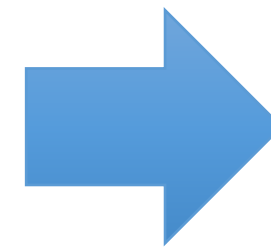
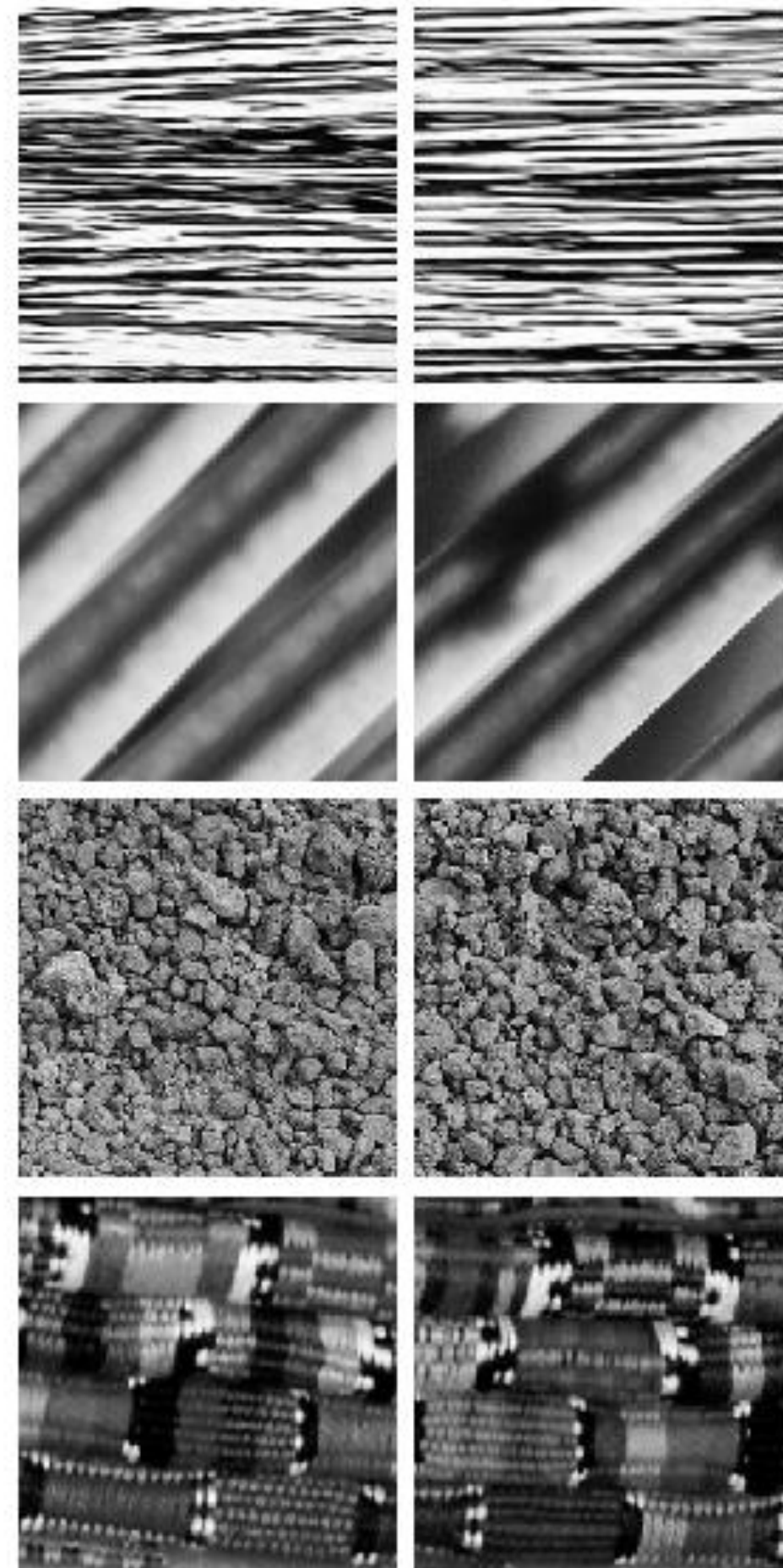
Local Binary Patterns

Similar textures have similar histograms



Local Binary Patterns

Similar textures have similar histograms



LBP for face recognition



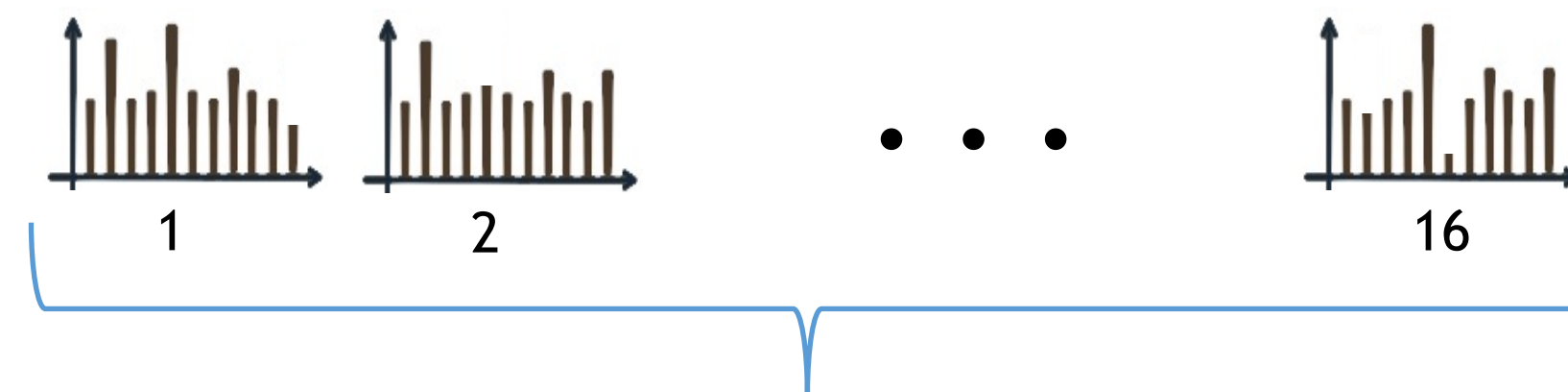
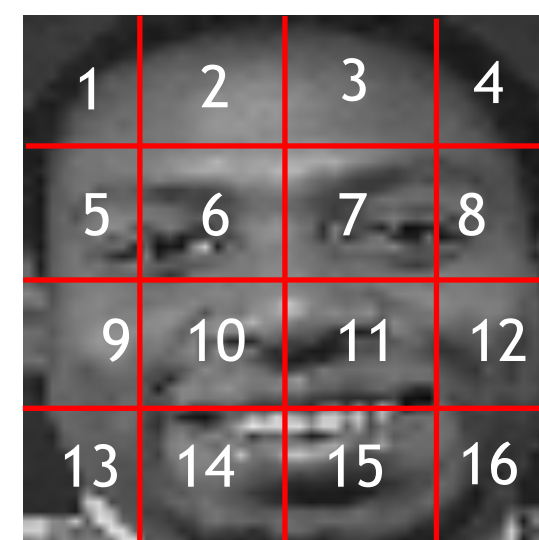
In the training set there are k classes.

For each class we have n training images.

In this example there are 40 classes with 9 images in each class.

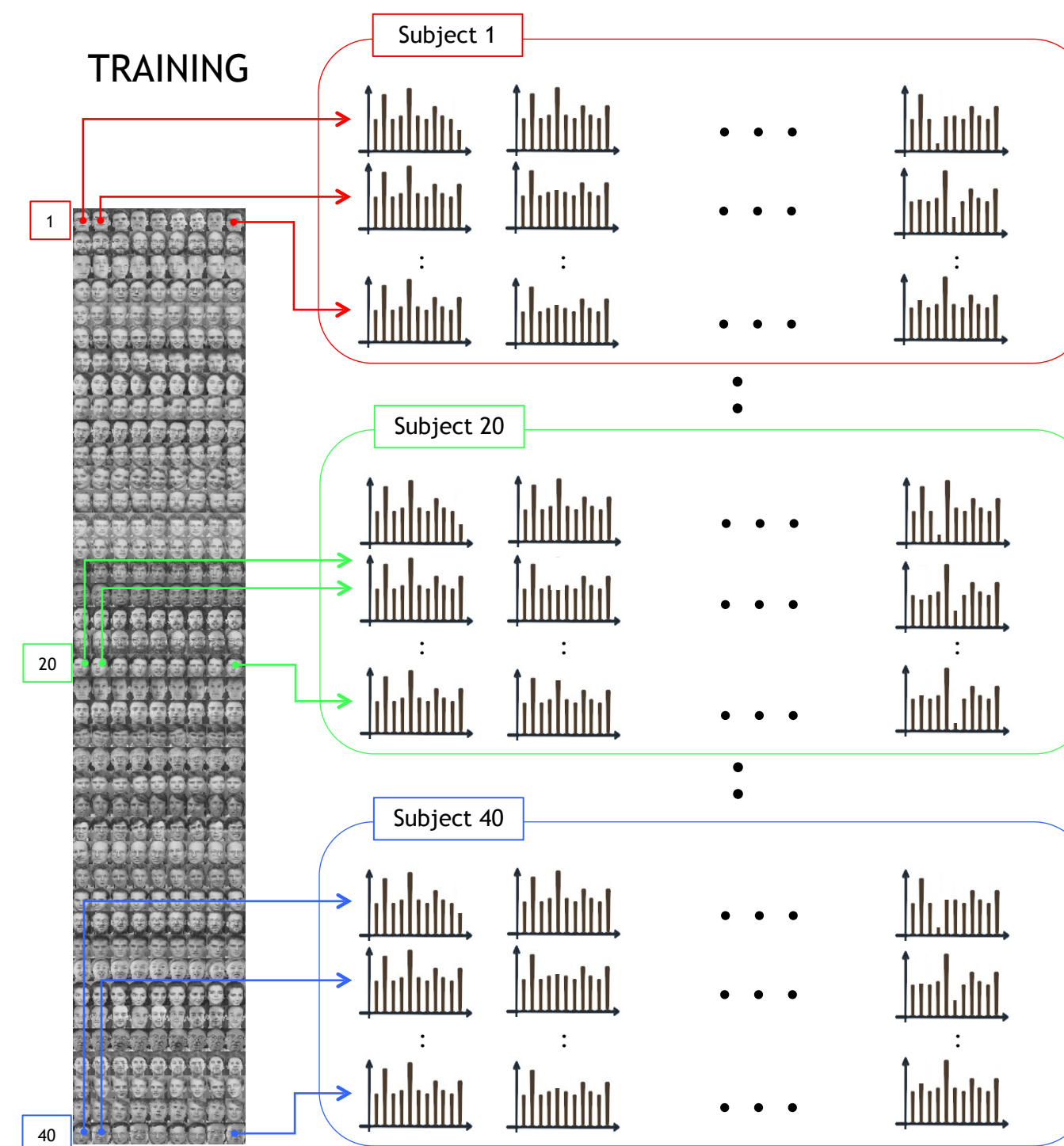
Each image is partitioned into 16 cells.

In each cell we extract LBP features.



A face is described using a feature of $16 \times 59 = 944$ elements

LBP for face recognition



Training Data

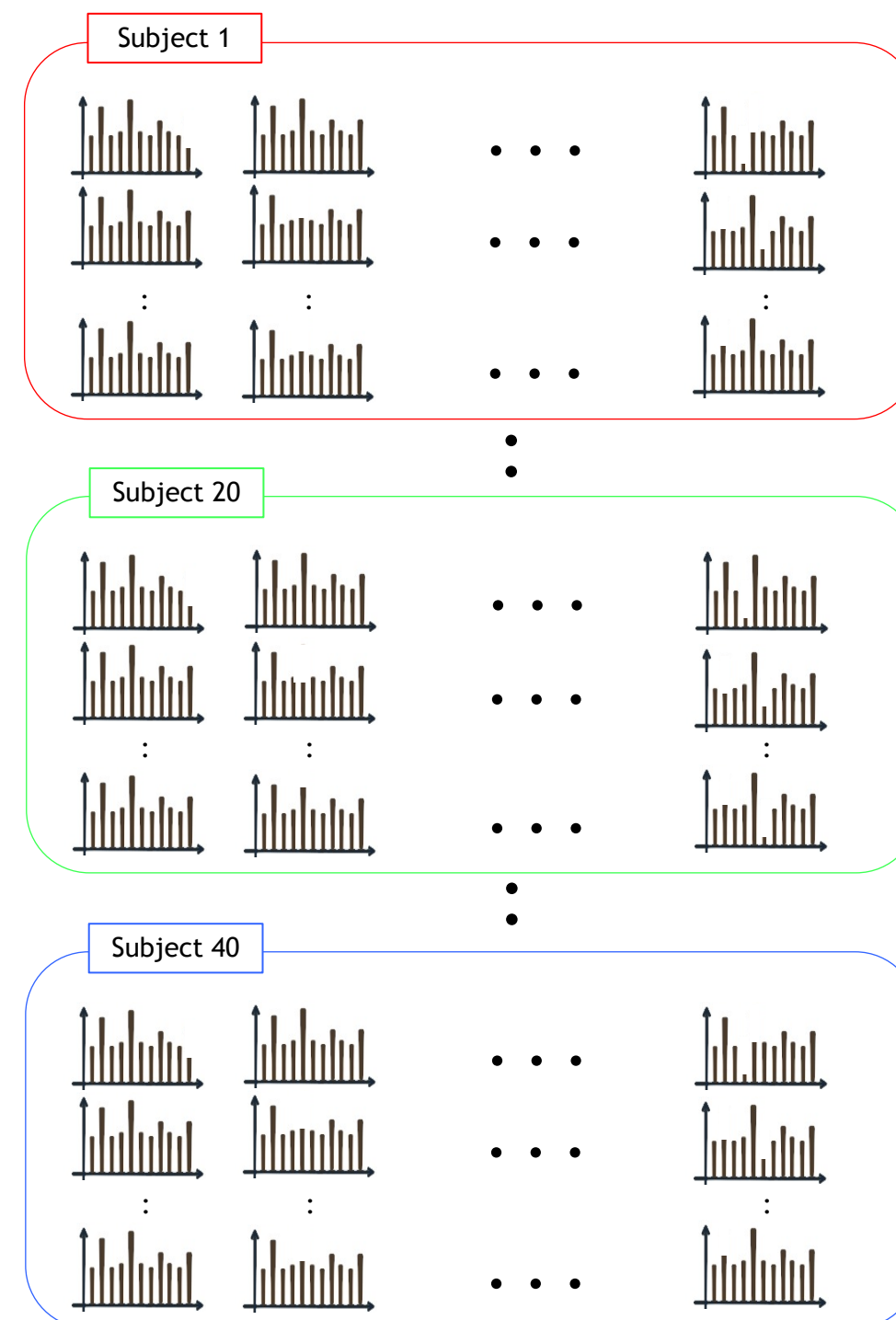
Table with:

$9 \times 40 = 360$ rows

and

$16 \times 59 = 944$ columns

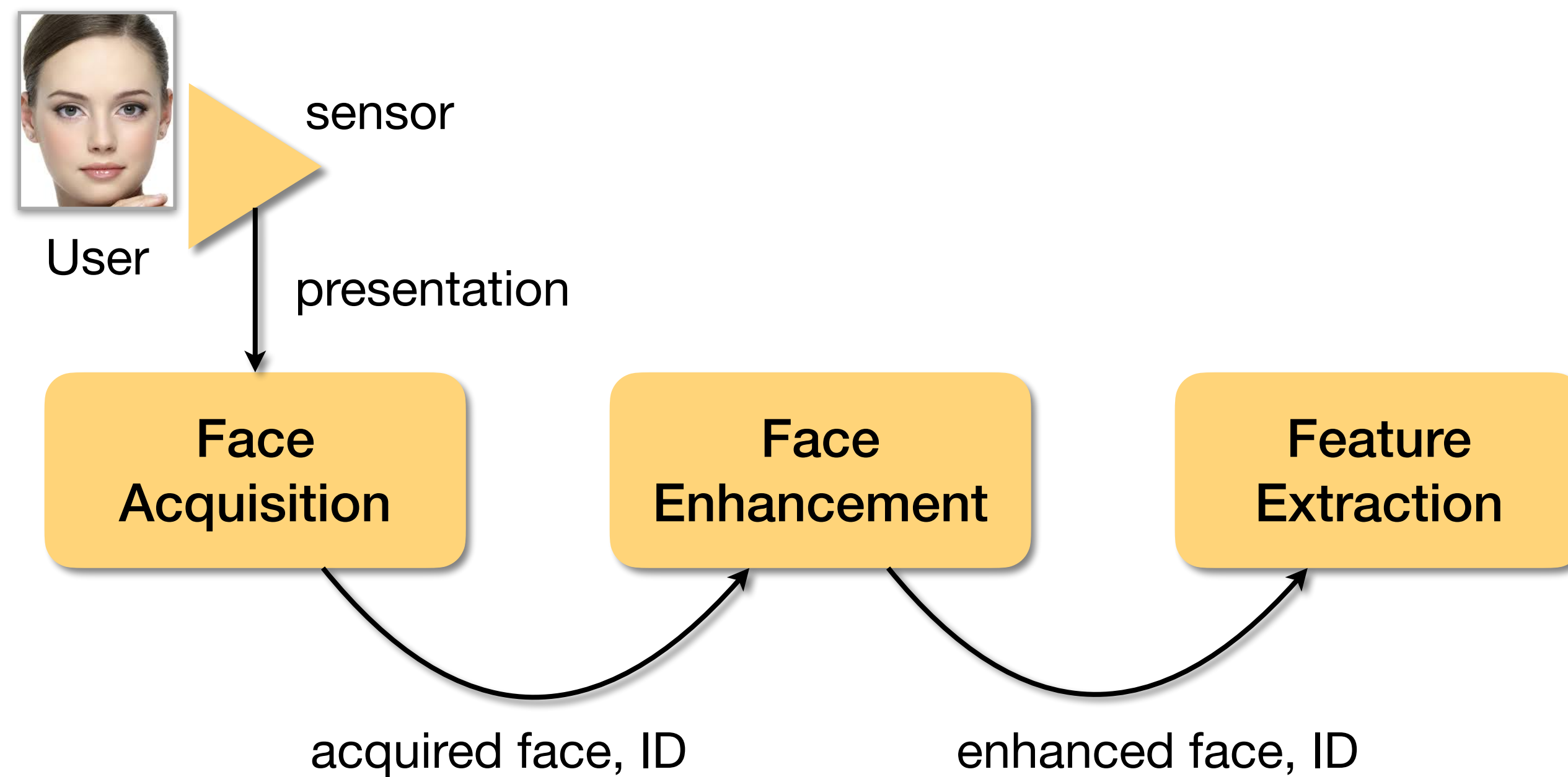
LBP for face recognition



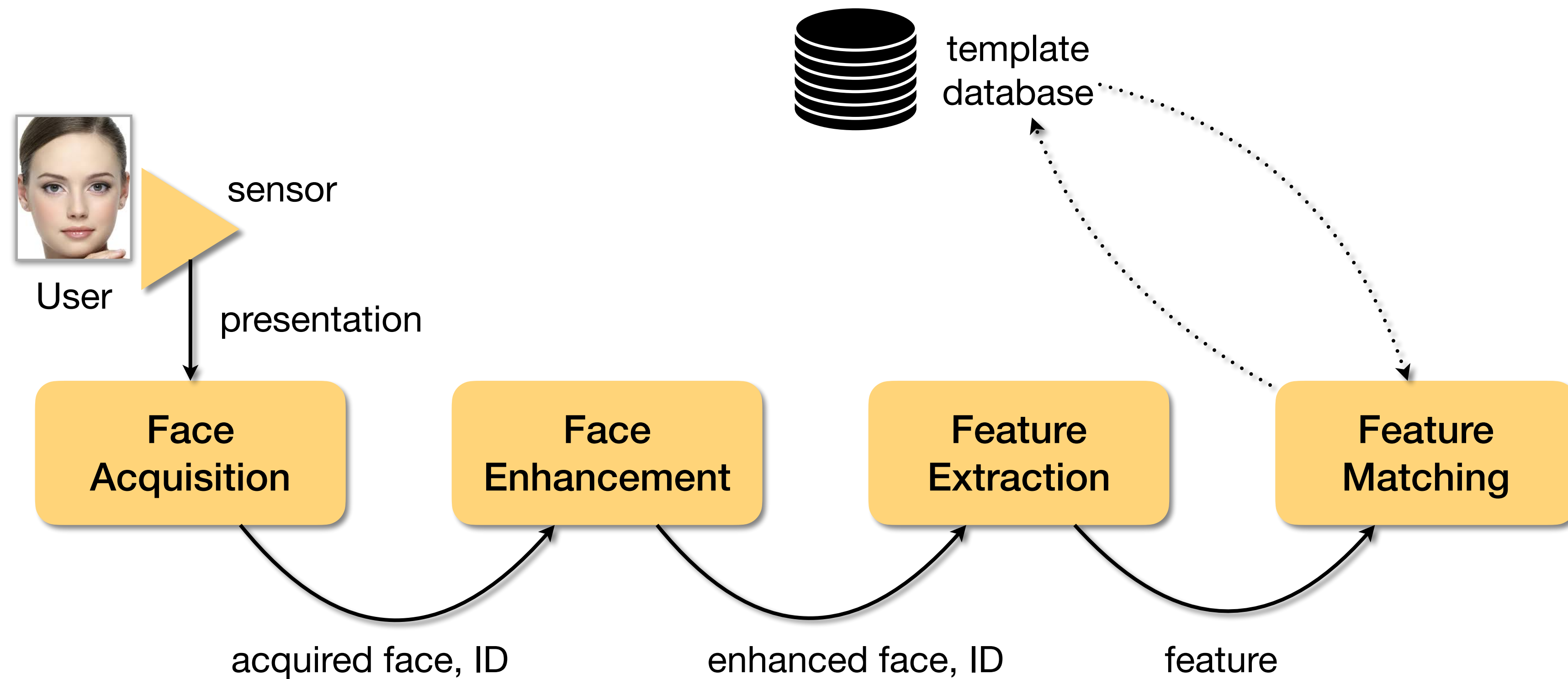
LBP for face recognition



Face Recognition



Face Recognition



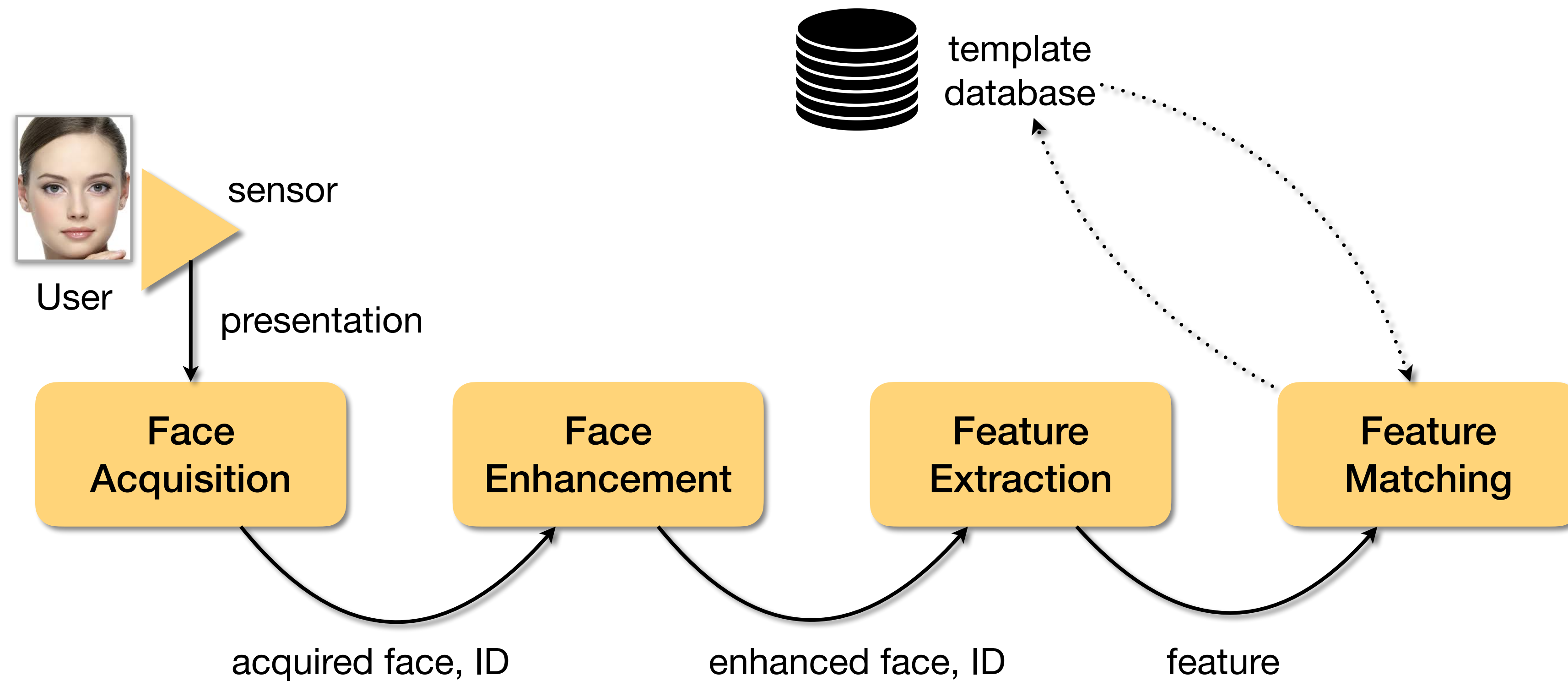
LBP for face recognition (Feature Matching)



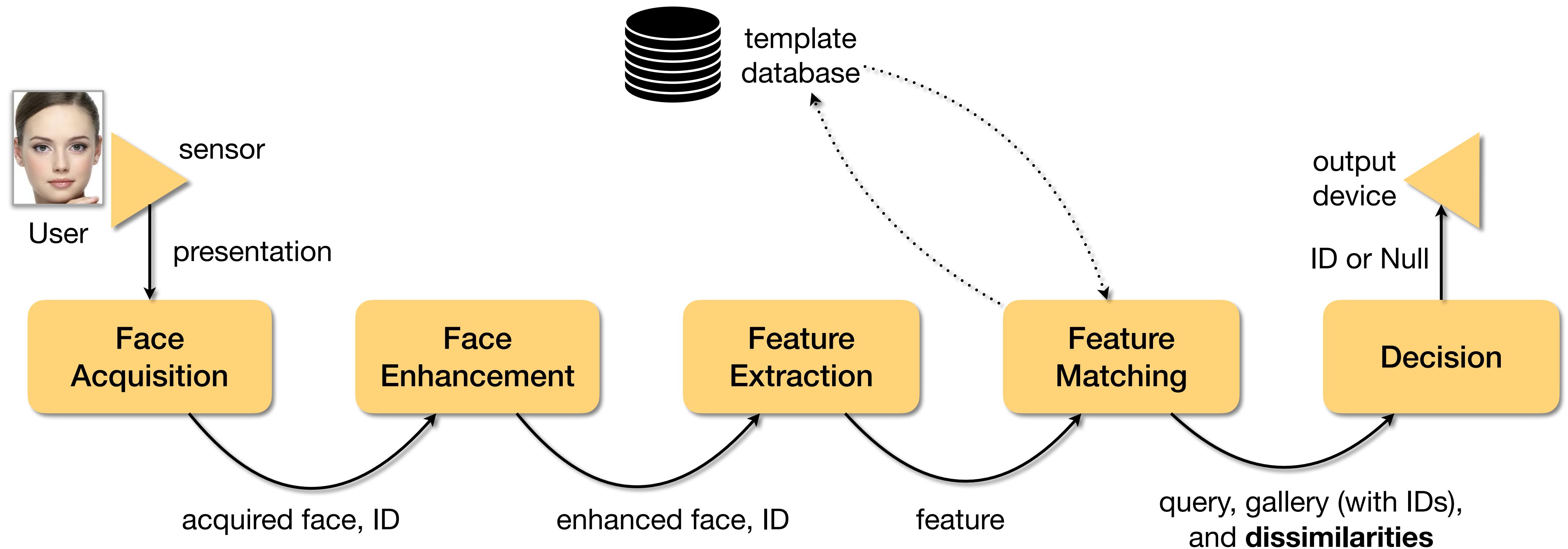
LBP for face recognition (Feature Matching)



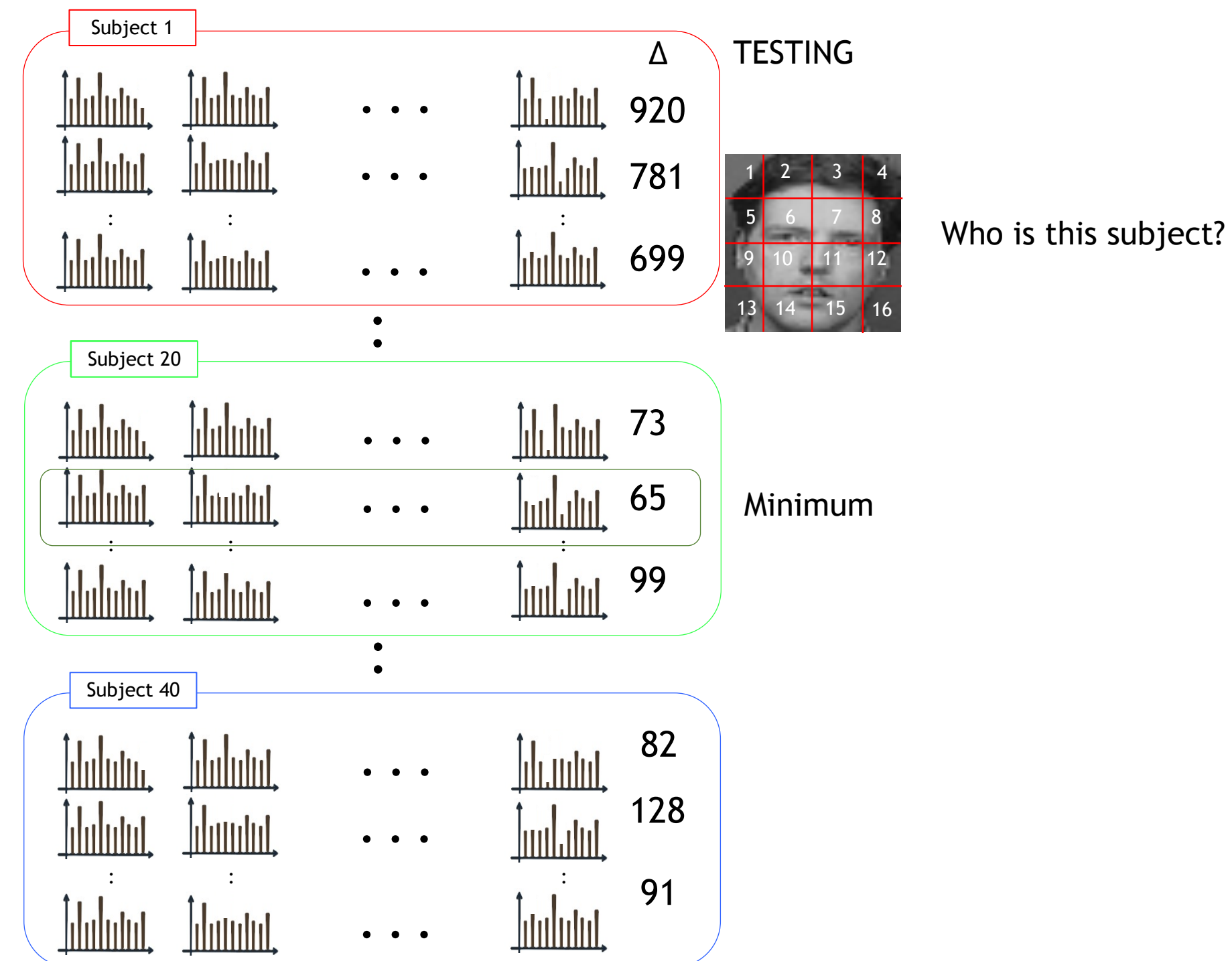
Face Recognition



Face Recognition



LBP for face recognition (Decision)



Feature Extraction

Focus

2D-appearance-based methods.

Types

Handcrafted features from Computer Vision.

Data-driven learned features from Machine Learning.



Feature Extraction

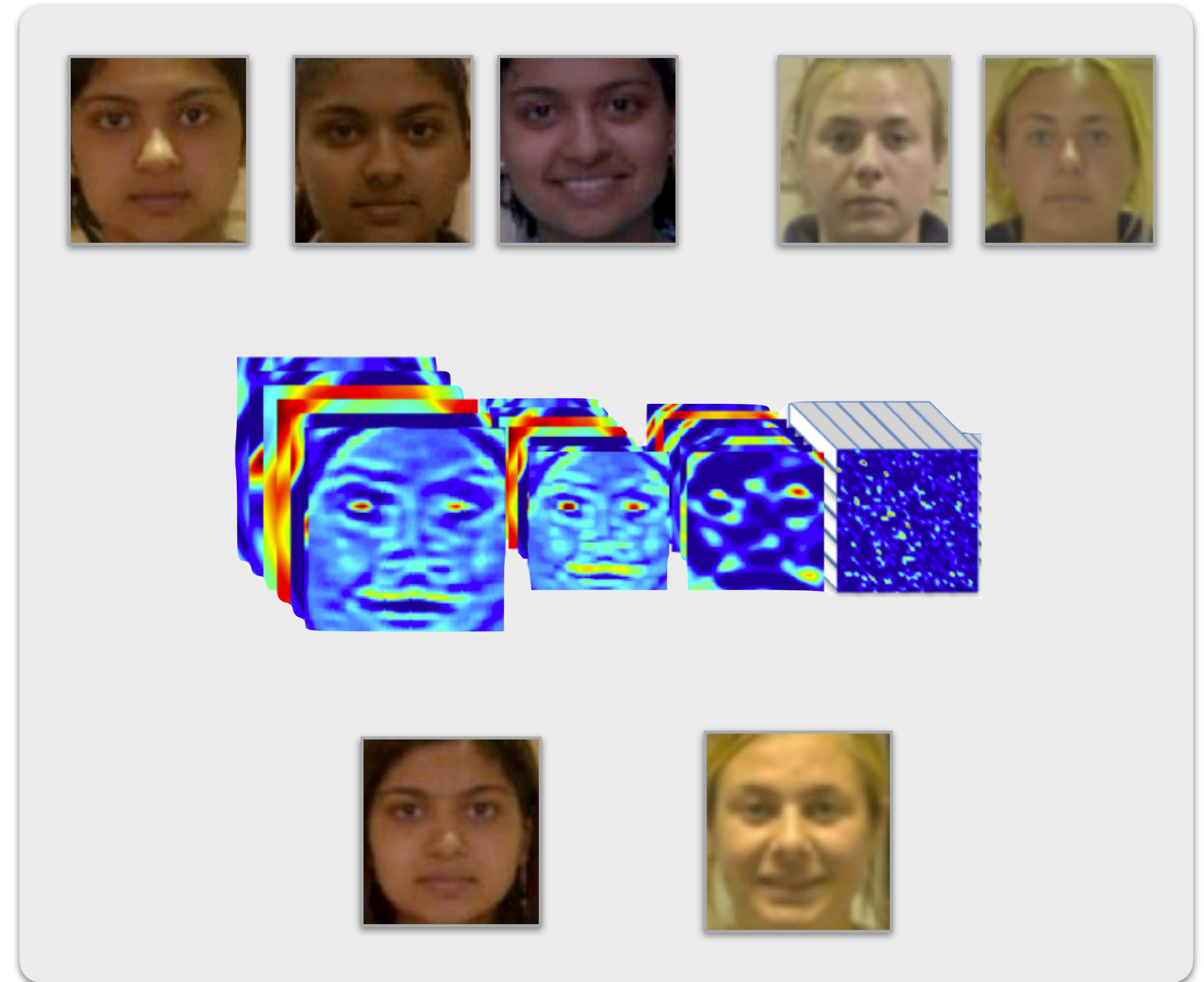
Focus

2D-appearance-based methods.

Types

Handcrafted features from Computer Vision.

Data-driven learned features from Machine Learning.



Feature Extraction

Deep Convolutional Neural Networks

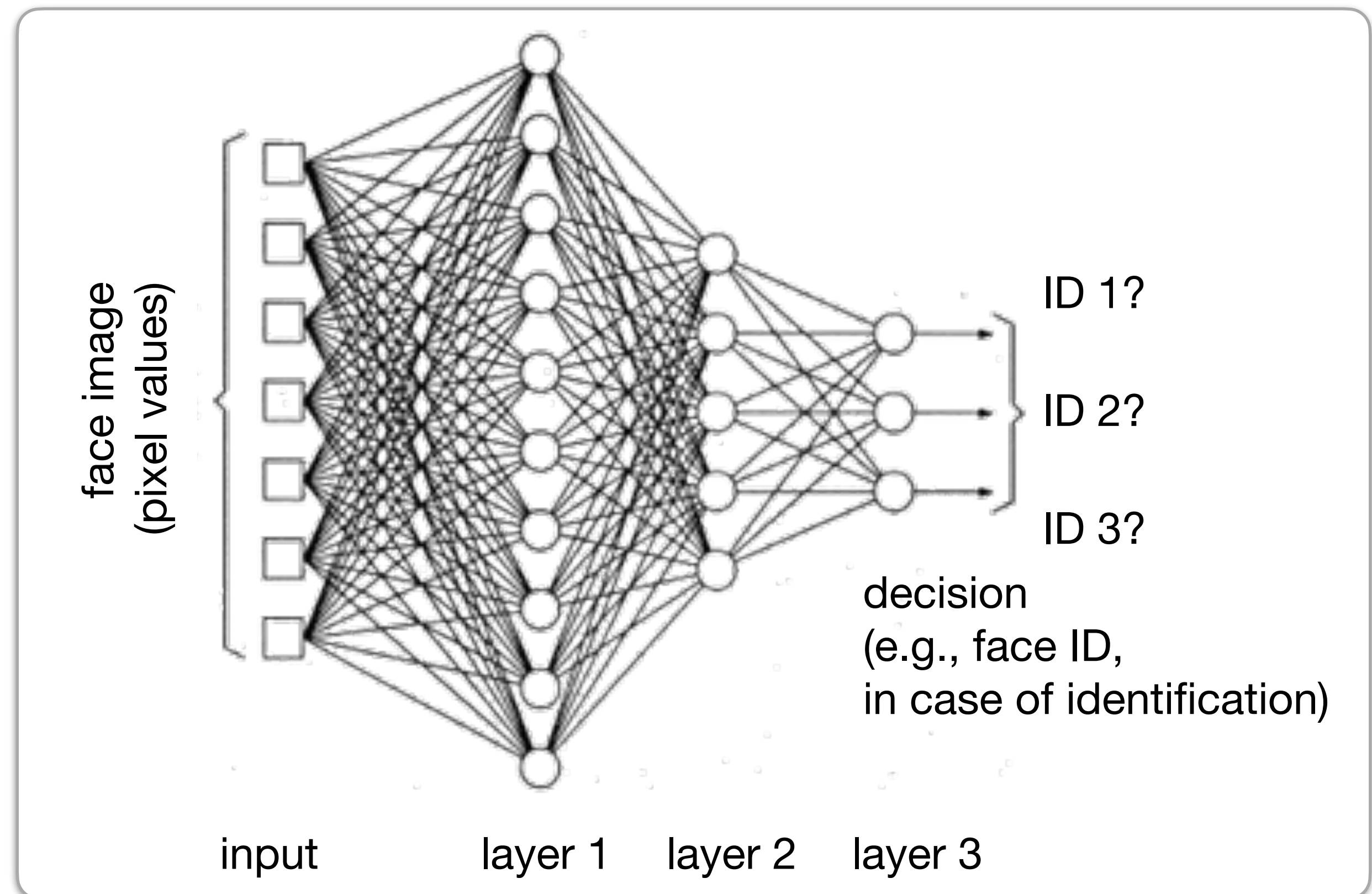
Feature Extraction

Deep Convolutional **Neural Networks**

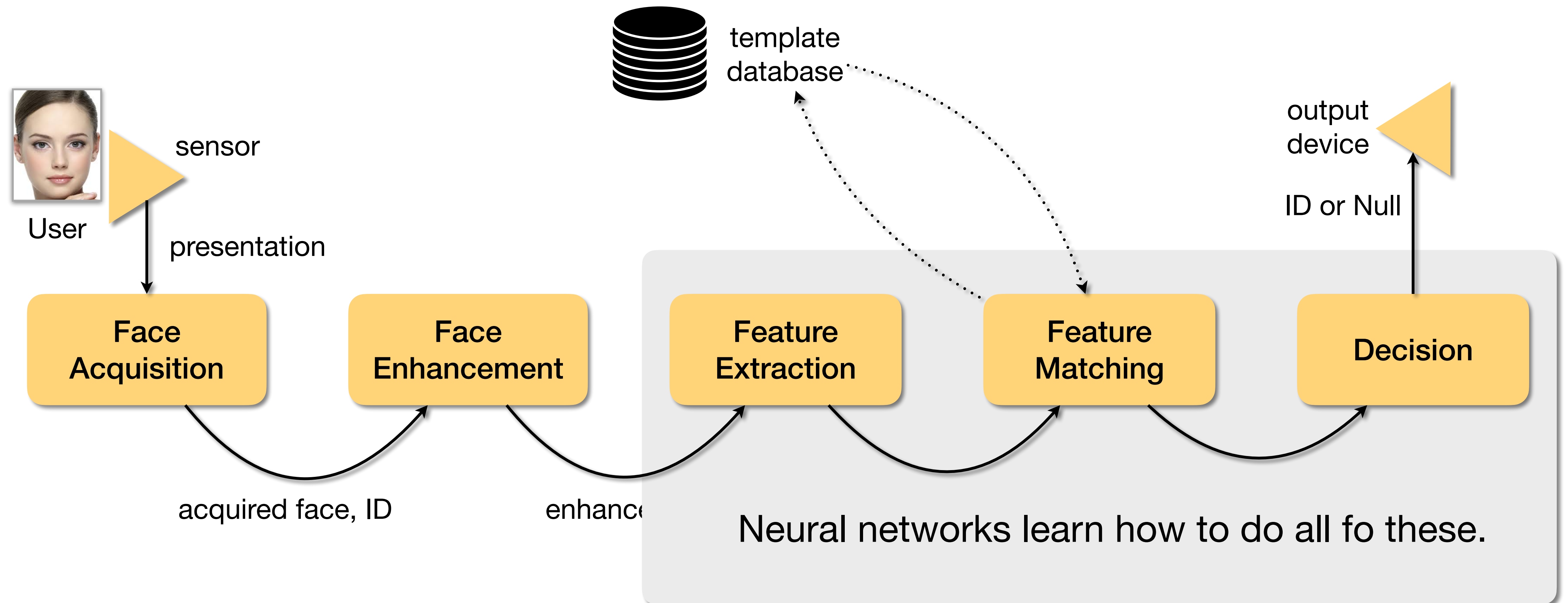
From pixels to
classification decision.

Hierarchy of feature
extractors.

Each layer extracts features
from previous layer.



Face Recognition



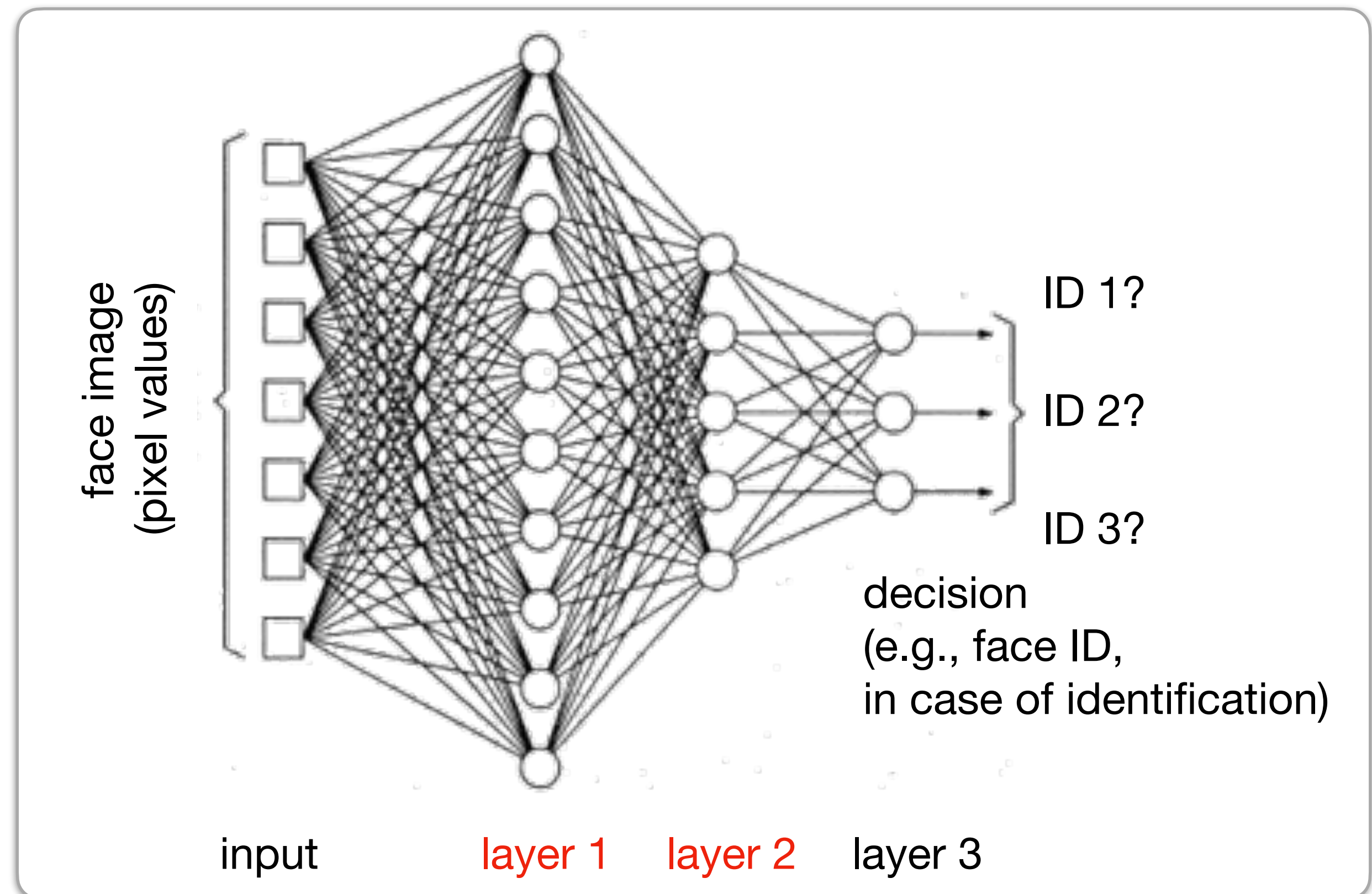
Data-Driven Face Recognition

Deep **Convolutional** Neural Networks

Convolutional Layers

E.g., layers 1 and 2.

Feature extractors are convolutional operations which are performed on the output of the previous layer.



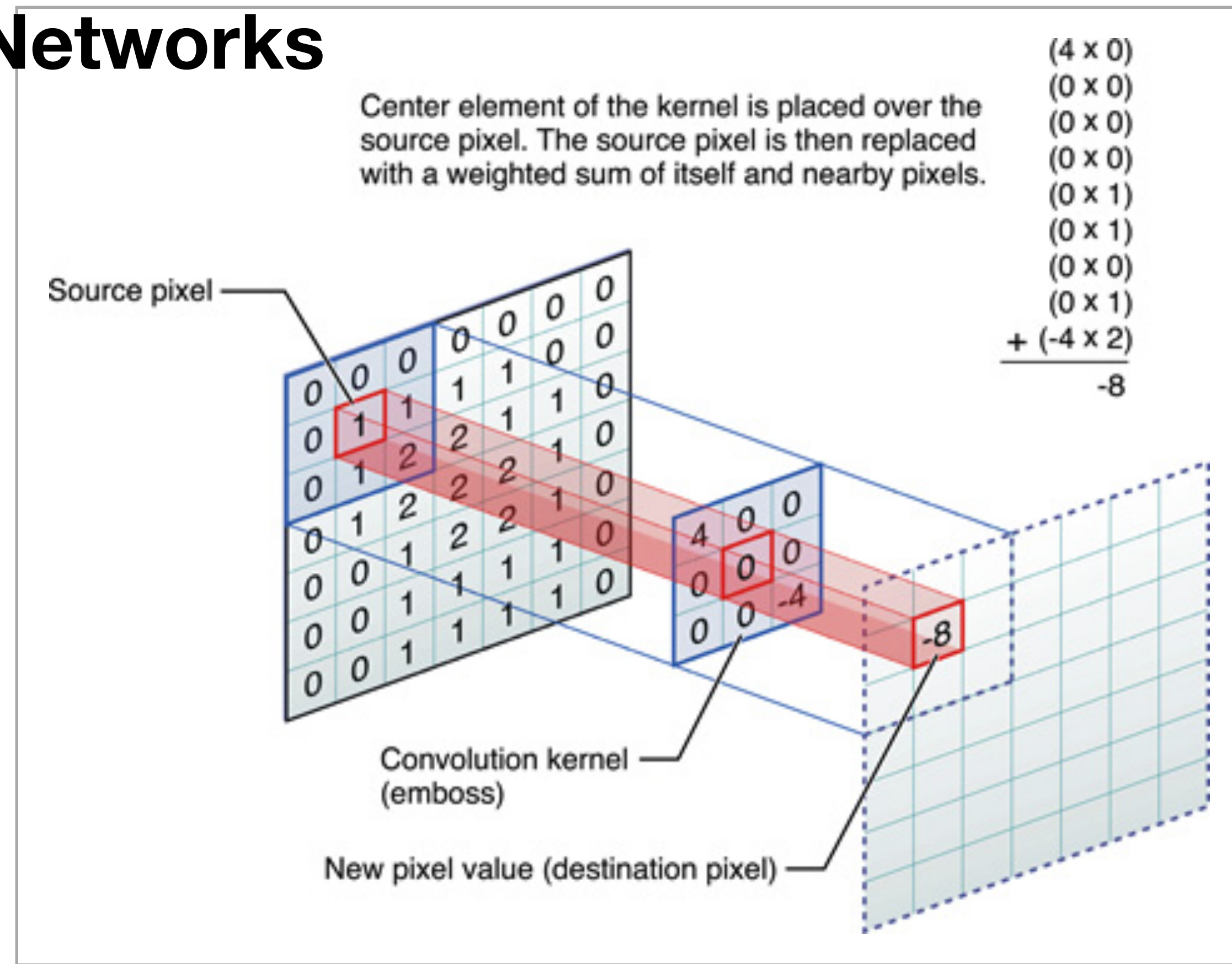
Data-Driven Face Recognition

Deep **Convolutional** Neural Networks

Convolutional Layers

E.g., layers 1 and 2.

Feature extractors are convolutional operations which are performed on the output of the previous layer.



Source: <https://developer.apple.com/library/archive/documentation/Performance/Conceptual/vimage/ConvolutionOperations/ConvolutionOperations.html>

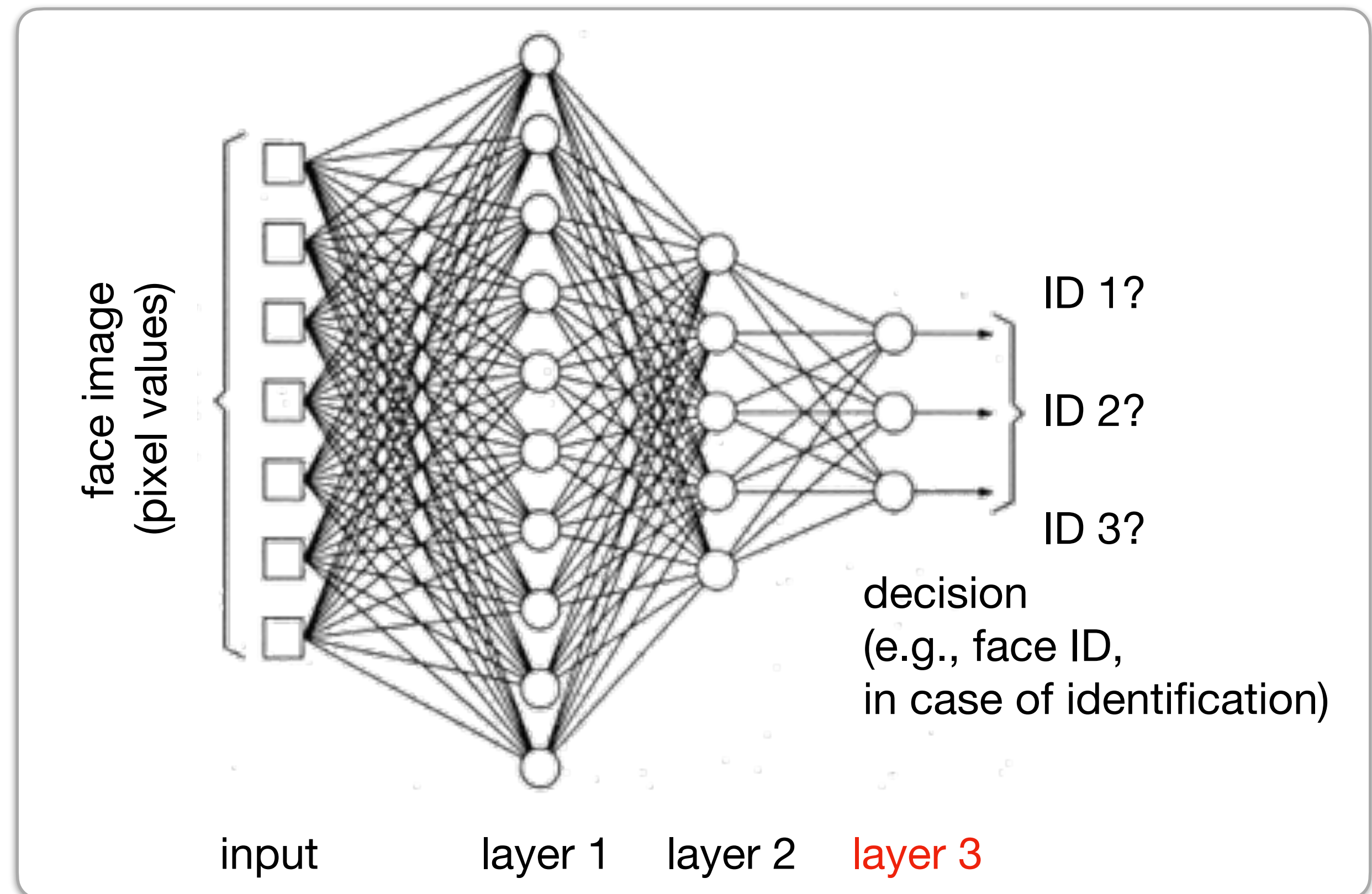
Data-Driven Face Recognition

Deep **Convolutional** Neural Networks

Fully Connected Layer

E.g., layer 3.

It performs the classification, presenting one score output for each class (identity, in the case of Biometrics).



Data-Driven Face Recognition

Deep Convolutional Neural Networks

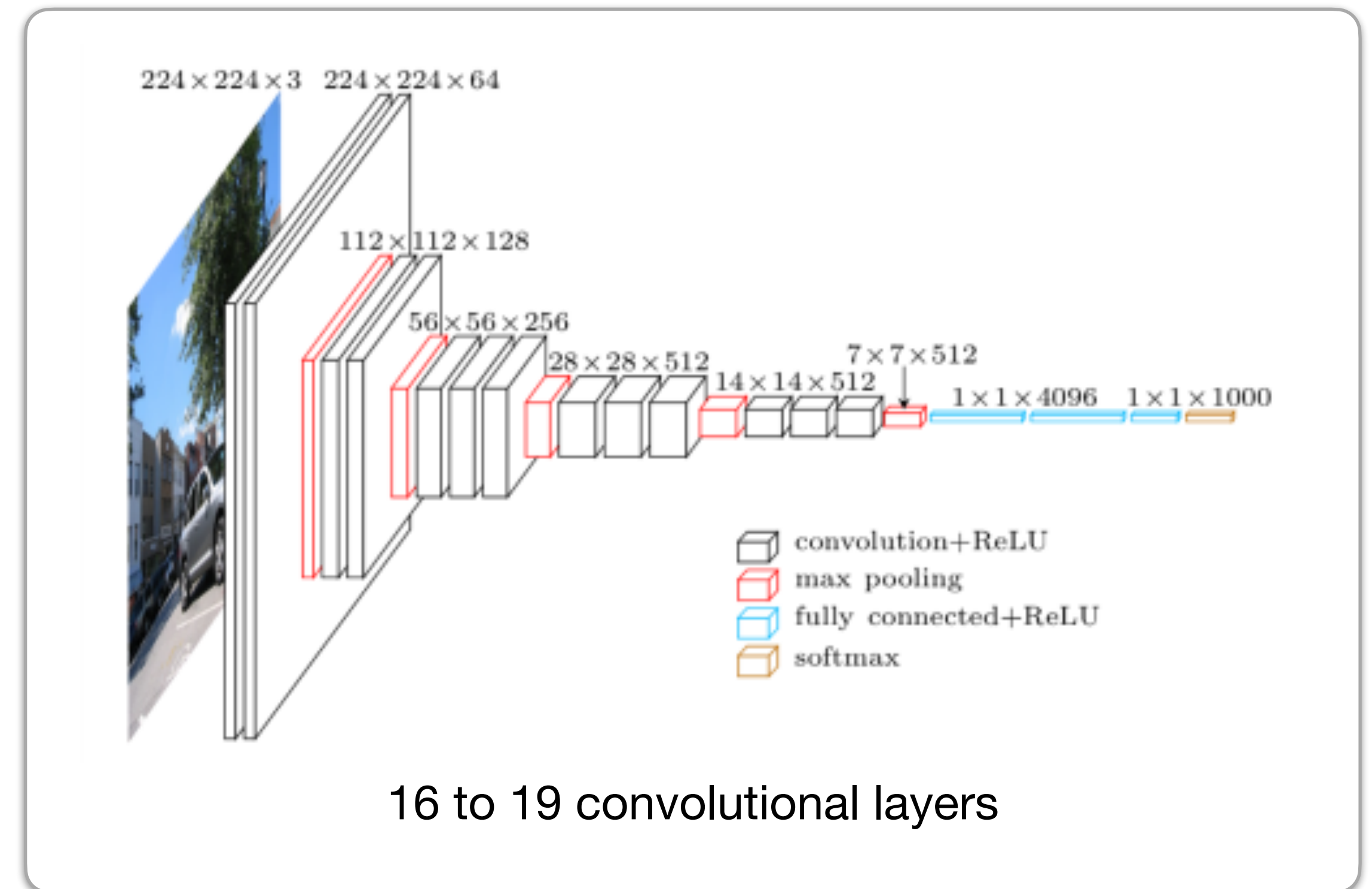
How deep can they be?

“Deep” refers to the number of layers.

E.g., VGG16

Simonyan and Zisserman

Very Deep Convolutional Networks for Large-Scale Image Recognition

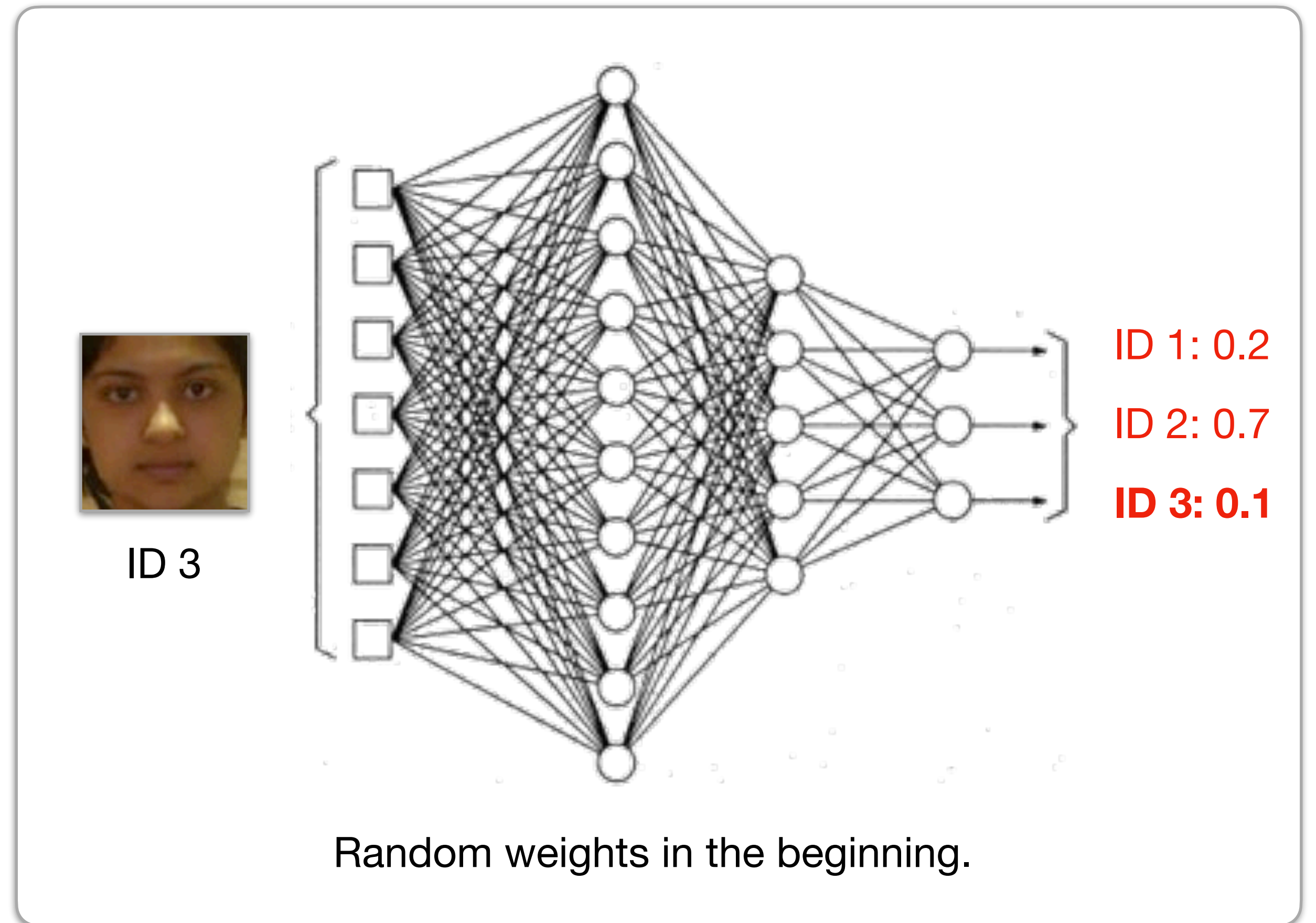


Data-Driven Face Recognition

Deep Learning

Training

Labeled examples
(e.g., faces and expected IDs)
are used to teach the network
to classify them correctly.

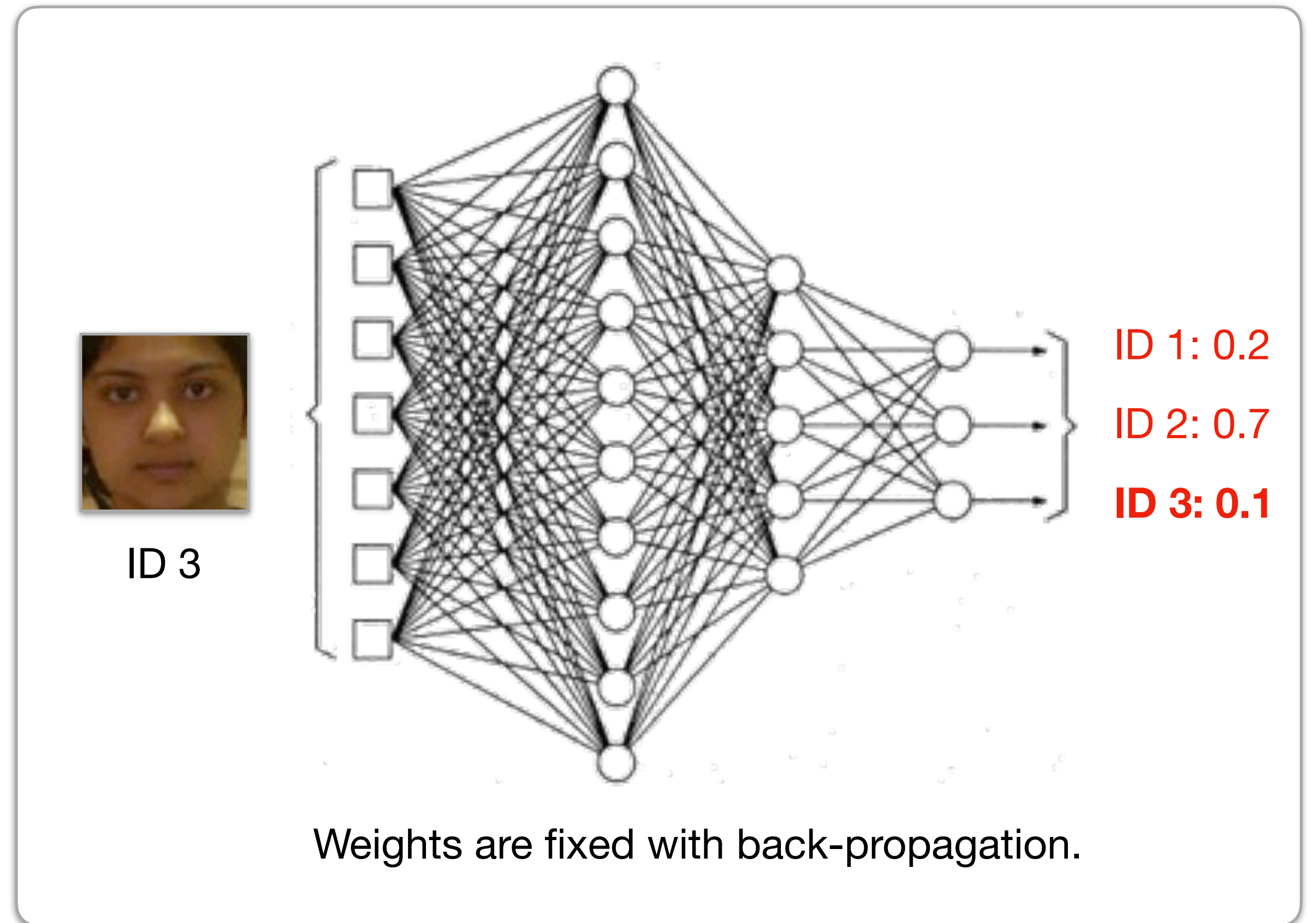


Data-Driven Face Recognition

Deep Learning

Training

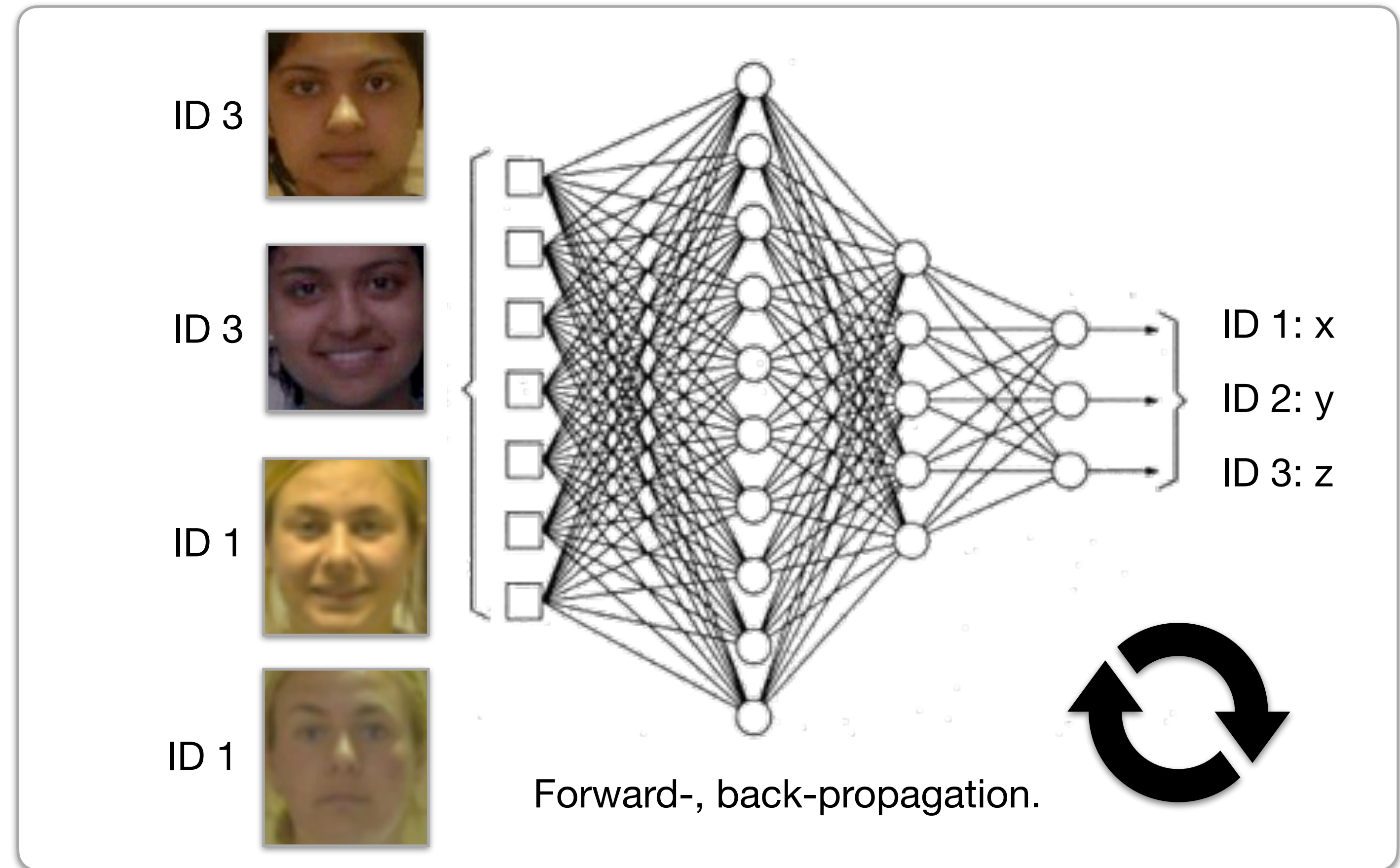
Back-propagation is used to fix the weights of the convolutions within the network.



Data-Driven Face Recognition

Deep Learning

Present various examples of each class and perform forward-, back-propagation.



Data-Driven Face Recognition

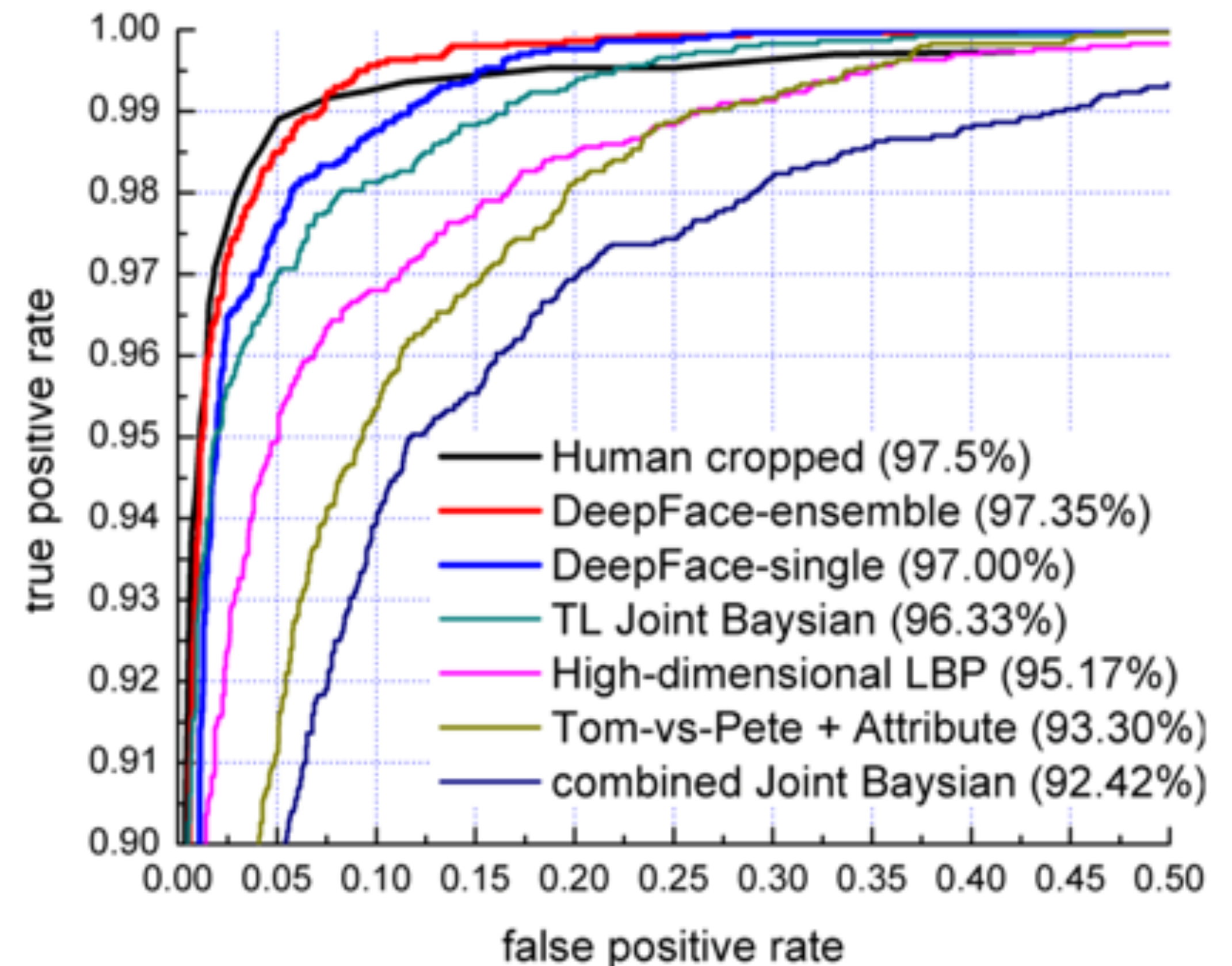
How good can it be?

E.g., DeepFace (Facebook)

Taigman *et al.*

DeepFace: Closing the Gap to Human-Level Performance in Face Verification

CVPR, 2014



Data-Driven Face Recognition

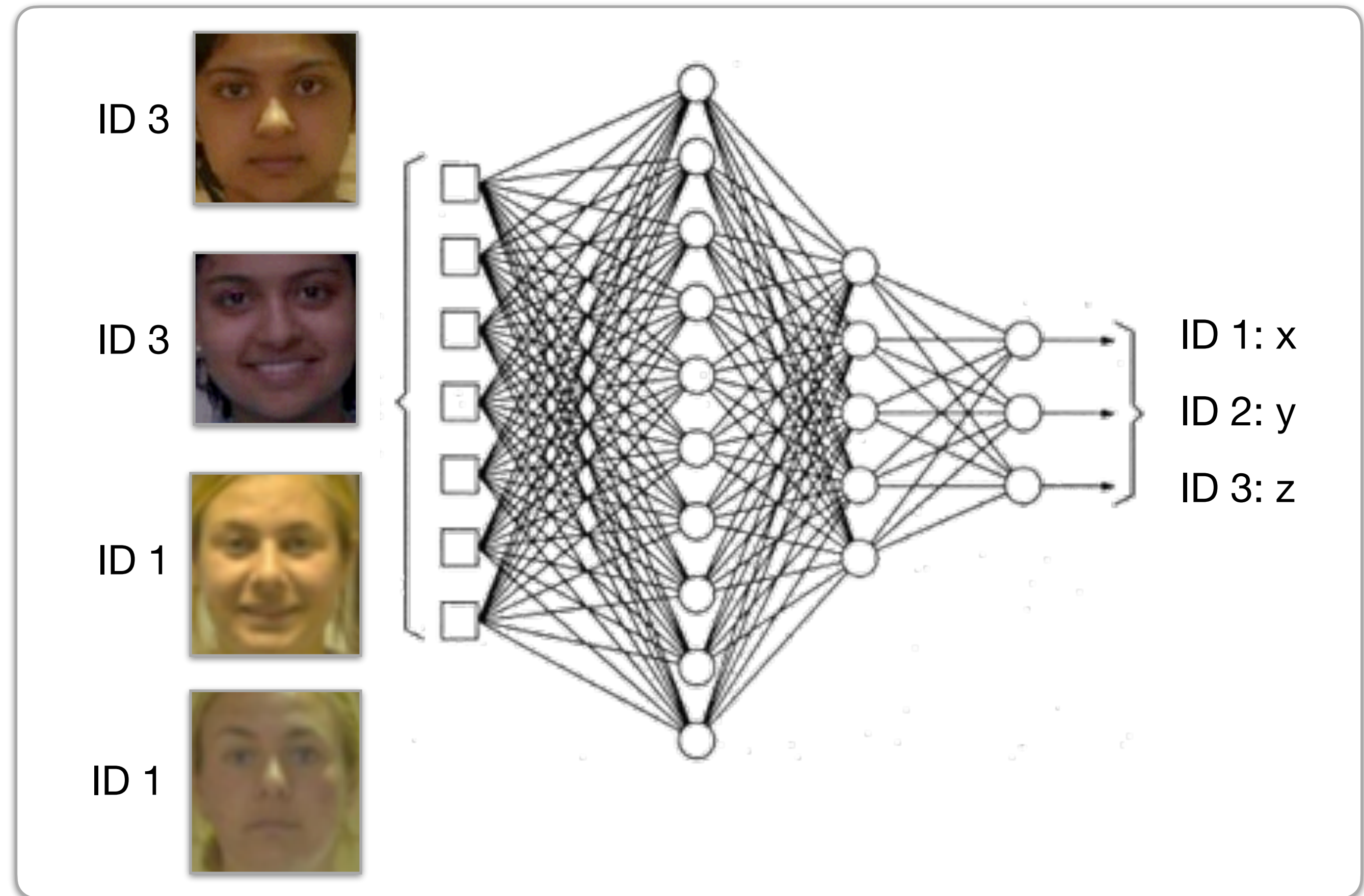
Deep Learning

What are the cons here?

How to enroll a new person?

Fixed number of classes (i.e., persons).

Need for large training dataset (thousands of sample per class).



S'up Next?

Improving Deep Learning

ArcFace

Additive Angular Margin Loss for Deep Face Recognition

Deng et al., CVPR 2019.

<https://bit.ly/3qsQmch>

