Face Recognition III

COMP 388-002/488-002 Biometrics

Daniel Moreira

Fall 2024

LOYOLA
UNIVERSITY CHICAGO

Today we will...

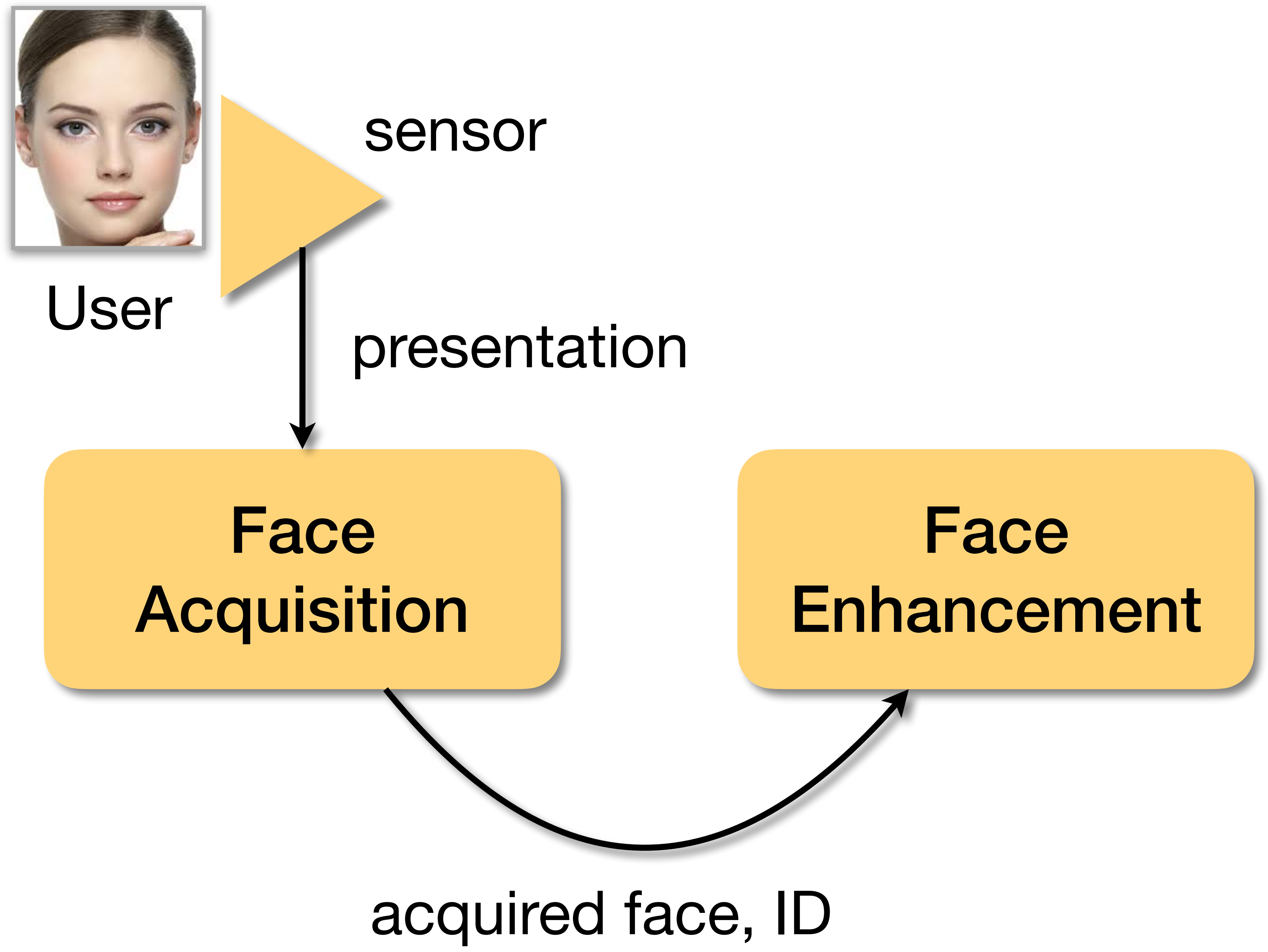
Get to know
Face description and matching.

Today's Attendance

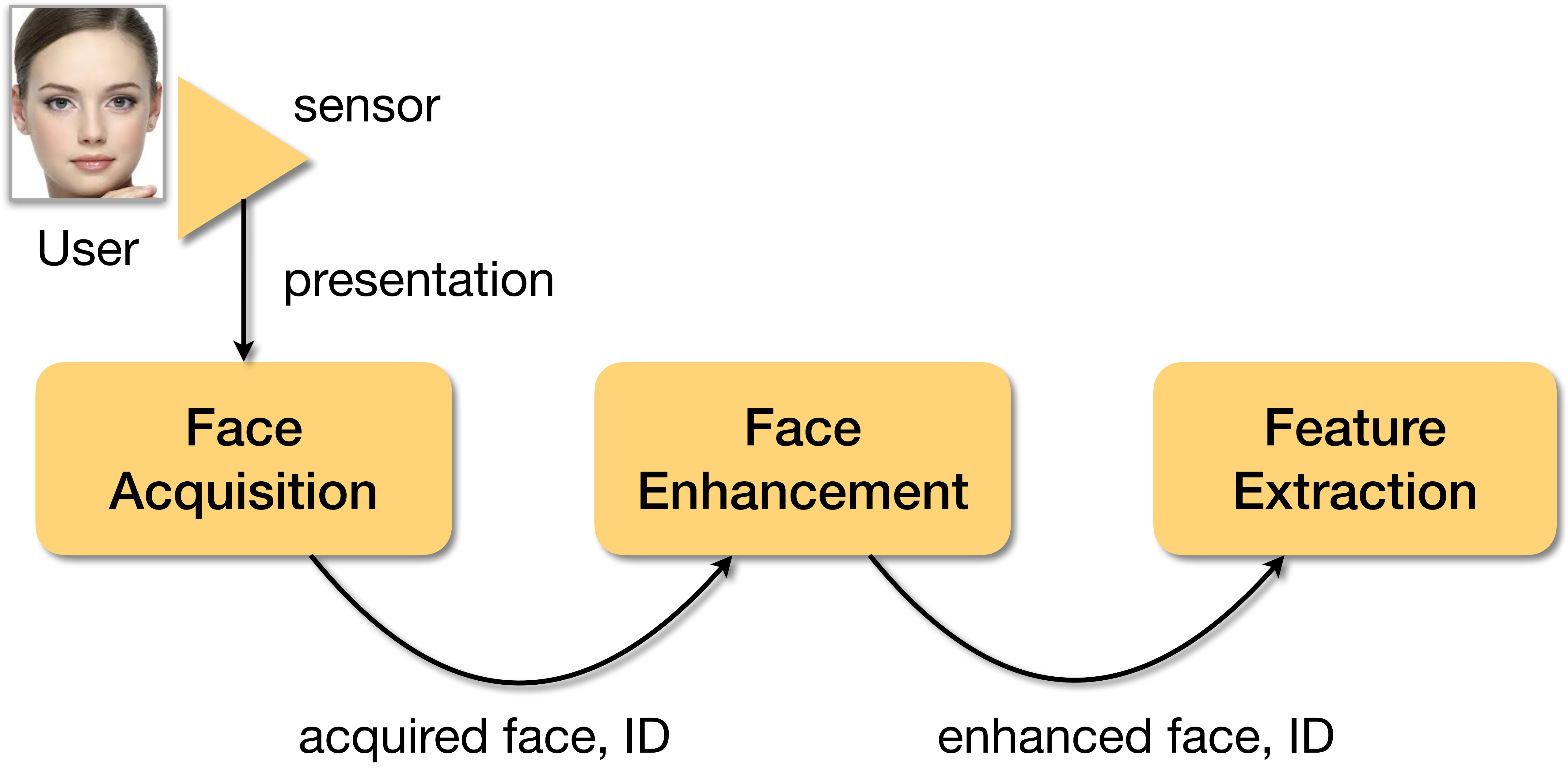
Please fill out the form

https://forms.gle/EwGr1RHf6Sf9ic2c9

Face Recognition



Face Recognition



Feature Extraction

Focus
2D-appearance-based methods.

Types
Handcrafted features from Computer Vision.

Data-driven learned features from Machine Learning.

LOYOLA UNIVERSITY CHICAGO

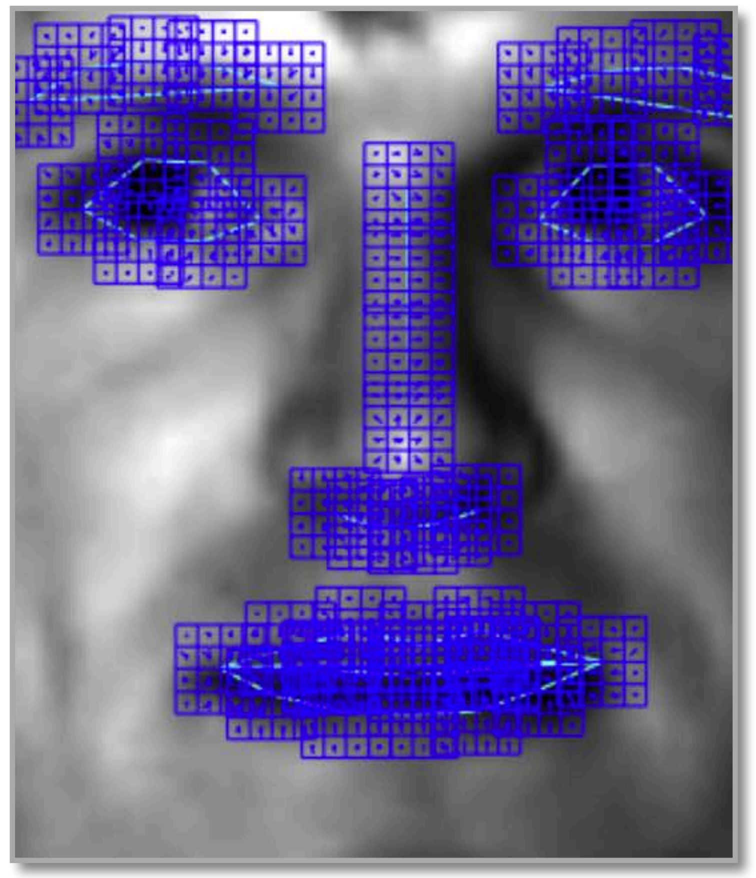
Feature Extraction

Focus
2D-appearance-based methods.

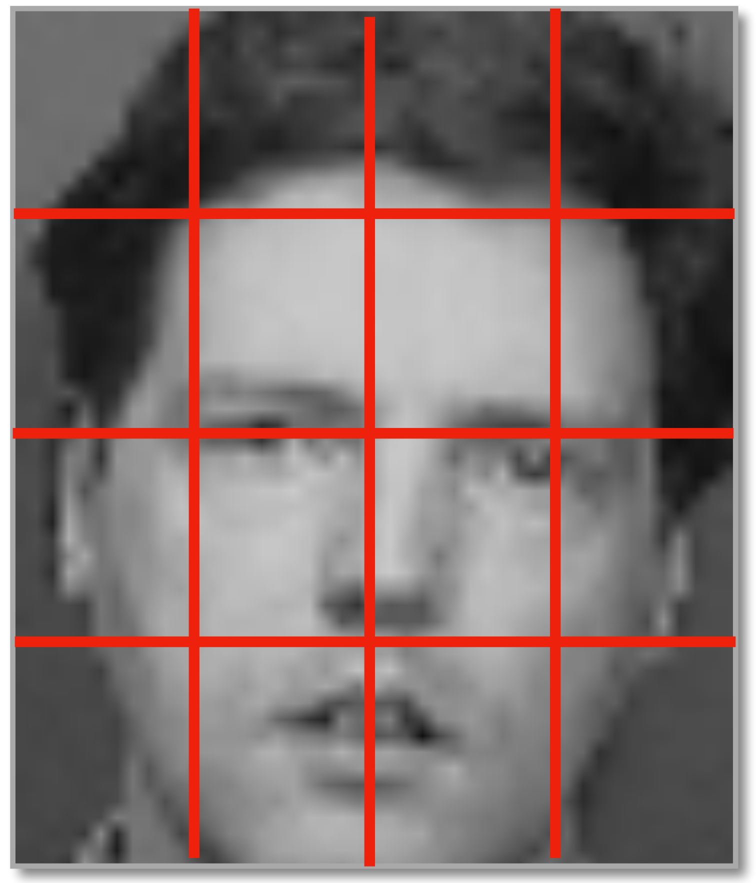
Types
Handcrafted features from Computer Vision.

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.

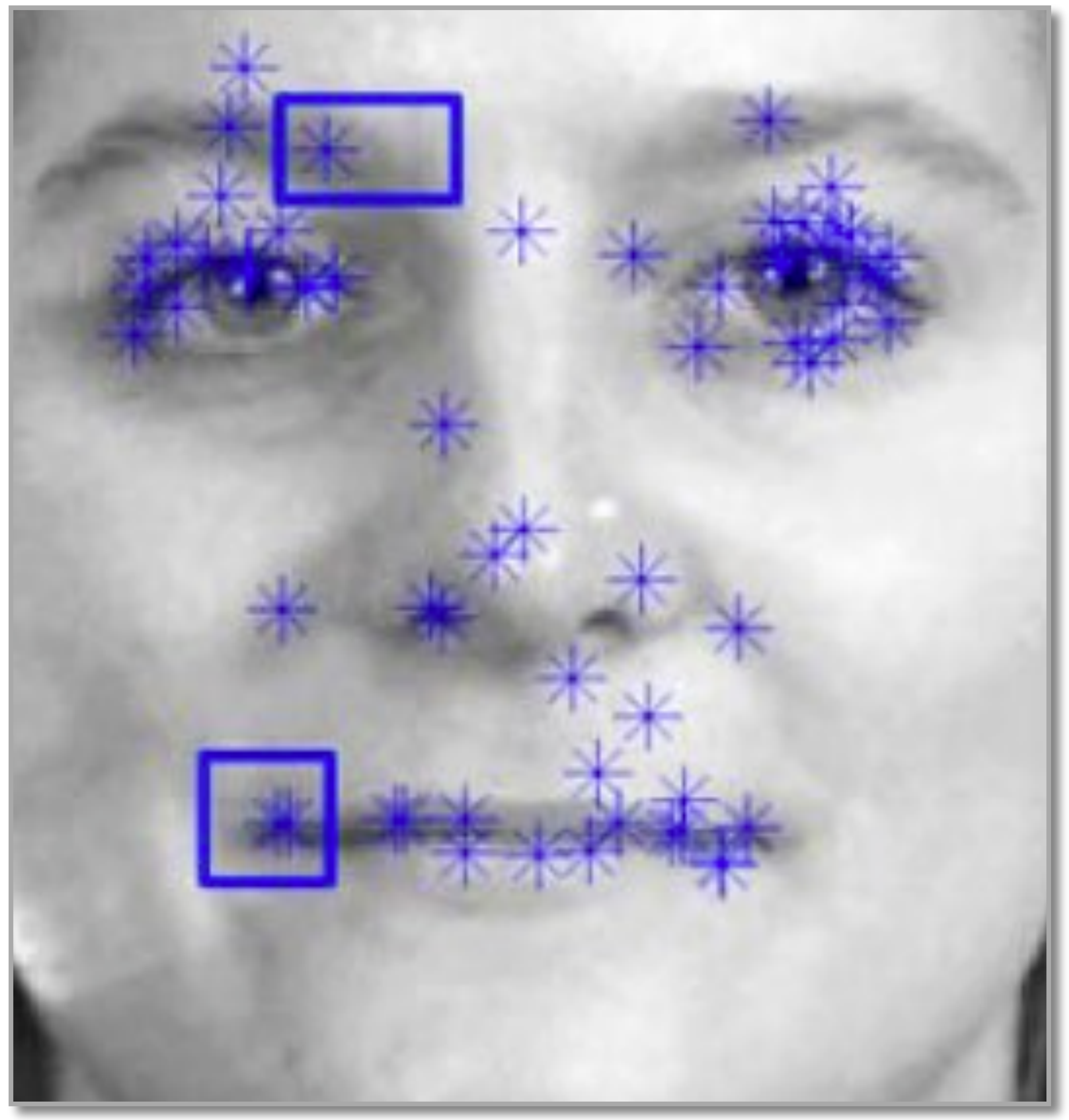
Feature Extraction

Handcrafted Features

Examples

Based on Gabor filters, interest points (e.g., SIFT[1], SURF[2], HOG[3]), or texture descriptors (e.g., LBP[4]).

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.



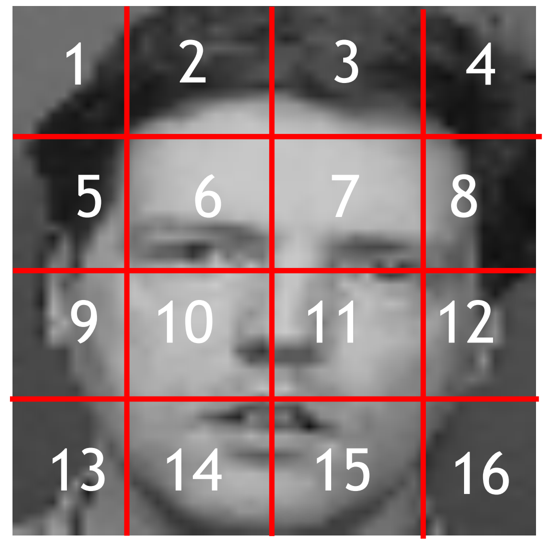
Geng and Jiang.
SIFT features for face recognition.
ICCSIT, 2009.

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

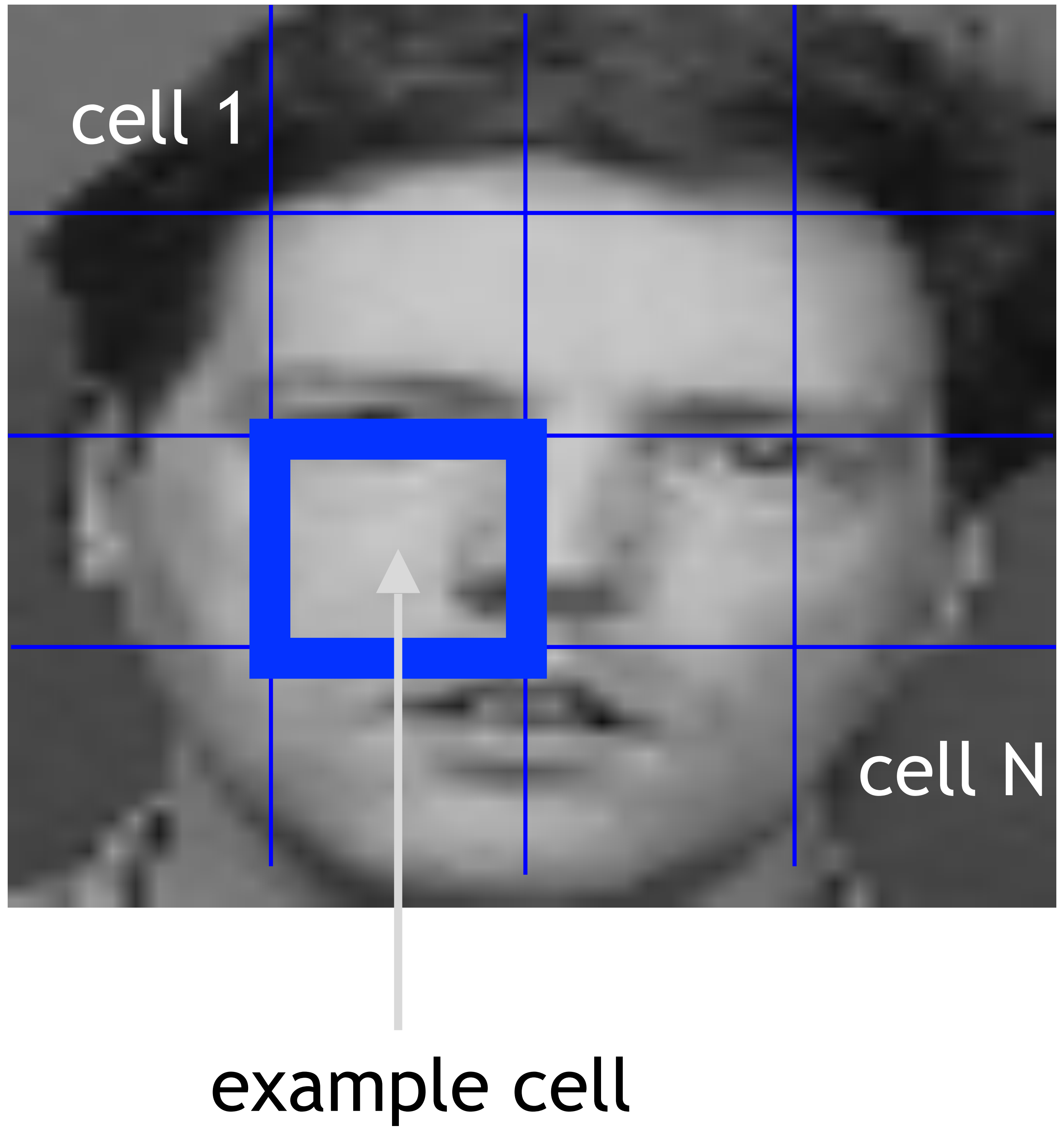
Concatenation

Source: Domingo Mery

Division into *N* cells | Cell coding | Mapping | Histogram calculation | Normalization | Concatenation

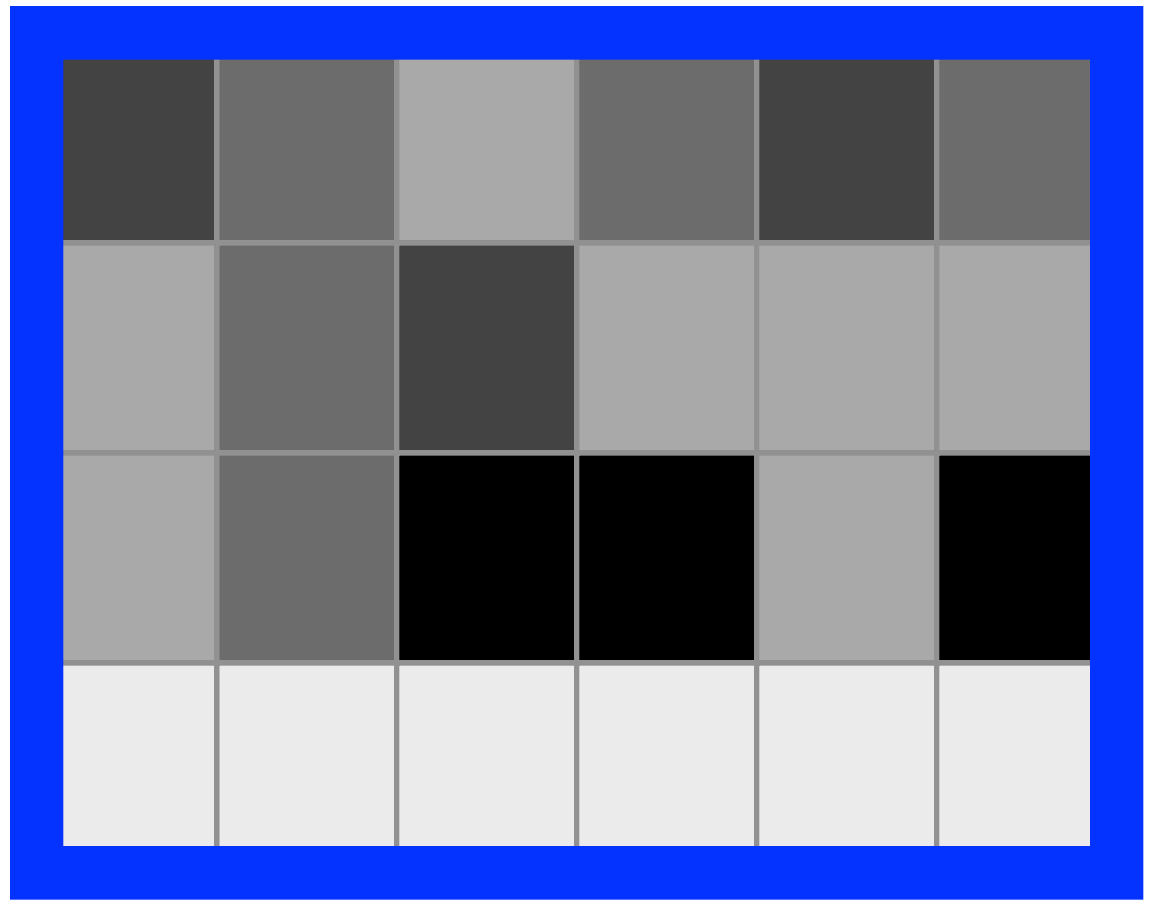
- LBP descriptors are calculated in image sub-regions (cells)

- Number and size of cells cannot be arbitrary (note space-scale considerations)



Source: Domingo Mery

Division into *N* cells	**Cell coding**	Mapping	Histogram calculation	Normalization	Concatenation



Source: Domingo Mery

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

Source: Domingo Mery

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

Source: Domingo Mery

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

Source: Domingo Mery

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
9	6	4
9	6	2

<		

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0		

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	≥	

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	1	

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	1	1

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	1	1
		0

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	1	1
		0
		0

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	1	1
		0
	1	0

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	1	1
		0
1	1	0

Source: Domingo Mery

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
9	6	4
9	6	2

0: <
1: ≥

0	1	1
1		0
1	1	0

Source: Domingo Mery

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
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

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

Source: Domingo Mery

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

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

Source: Domingo Mery

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

$\rightarrow$

	230				

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

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

Source: Domingo Mery

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	?			

6	9	6
6	4	9
6	2	2

0: <
1: ≥

x

1	2	4
128	+	8
64	32	16

Source: Domingo Mery

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

$\Rightarrow$

	230	207			

6	9	6
6	4	9
6	2	2

0: <
1: ≥

1	1	1
1		1
1	0	0

x

1	2	4
128	+	8
64	32	16

$= 1 + 2 + 4 + 8 + 64 + 128 = 207$

Source: Domingo Mery

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

→

	230	207	?		

Source: Domingo Mery

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

→

	230	207	25		

9	6	4
4	9	9
2	2	9

0: <
1: ≥

1	0	0
0		1
0	0	1

x

1	2	4
128	+	8
64	32	16

= 1 + 8 + 16 = 25

Source: Domingo Mery

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

→

	230	207	25	168	

Source: Domingo Mery

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

	230	207	25	168	
	243				

Source: Domingo Mery

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

→

	230	207	25	168	
	243	255			

Source: Domingo Mery

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

	230	207	25	168	
	243	255	255		

Source: Domingo Mery

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	

Source: Domingo Mery

Note on neighborhood definition

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

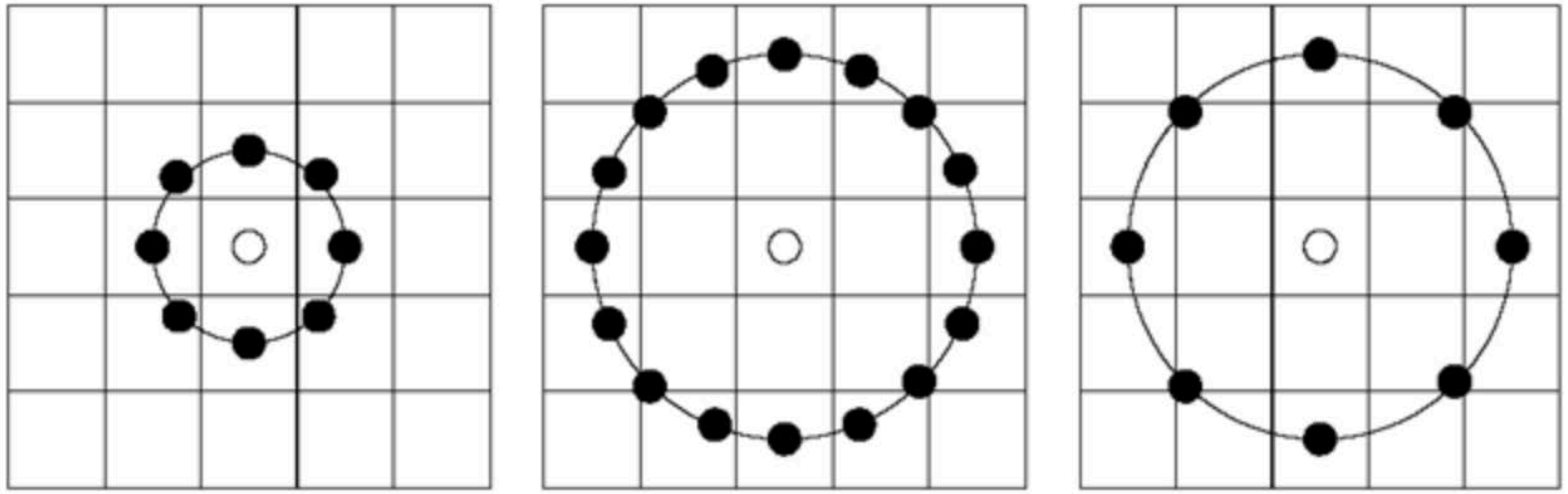
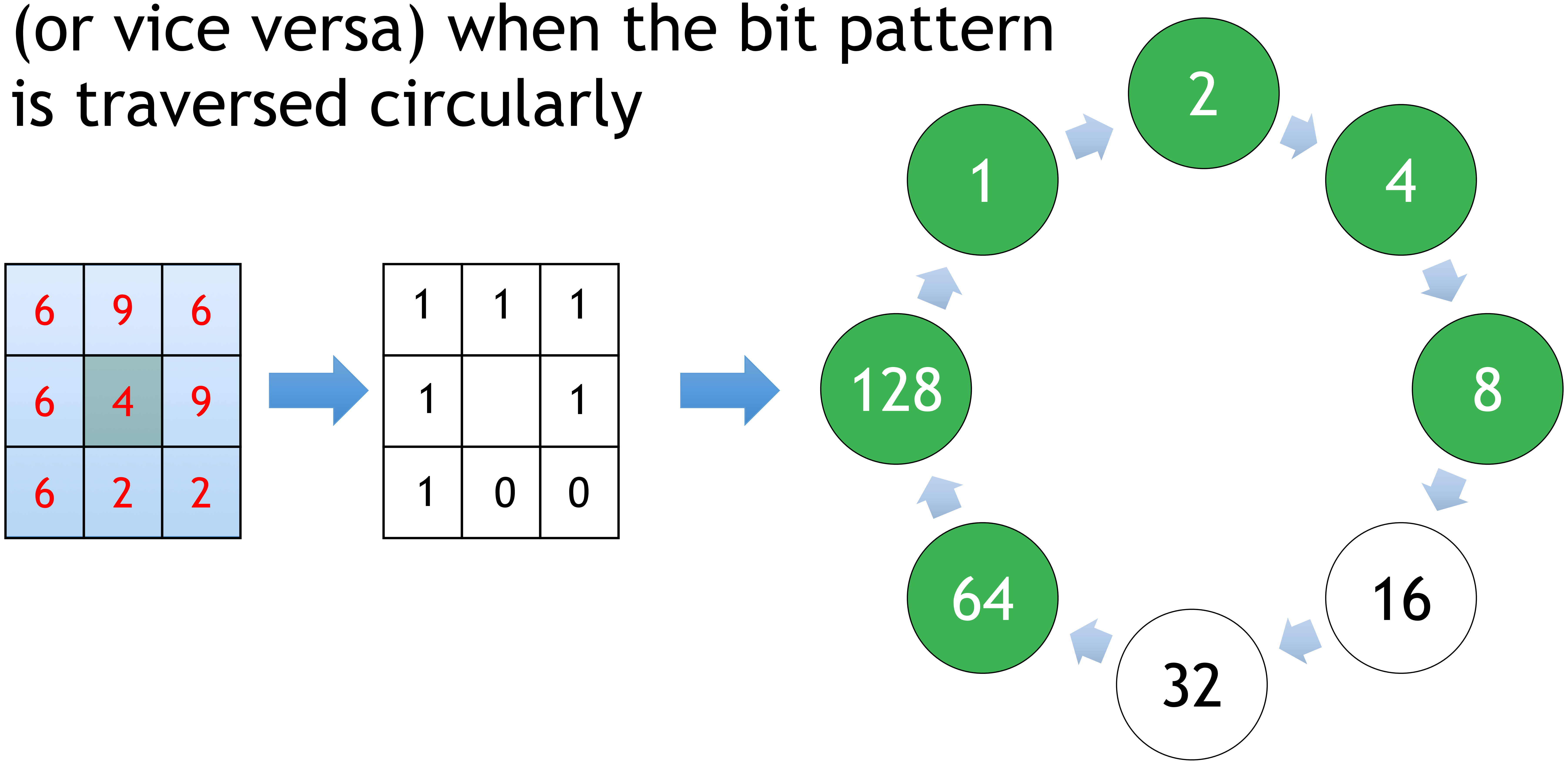


Image source: http://what-when-how.com/face-recognition/local-representation-of-facial-features-face-image-modeling-and-representation-face-recognition-part-1/

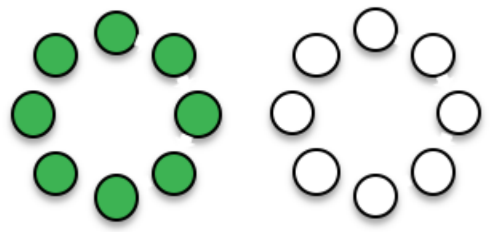
Source: Domingo Mery

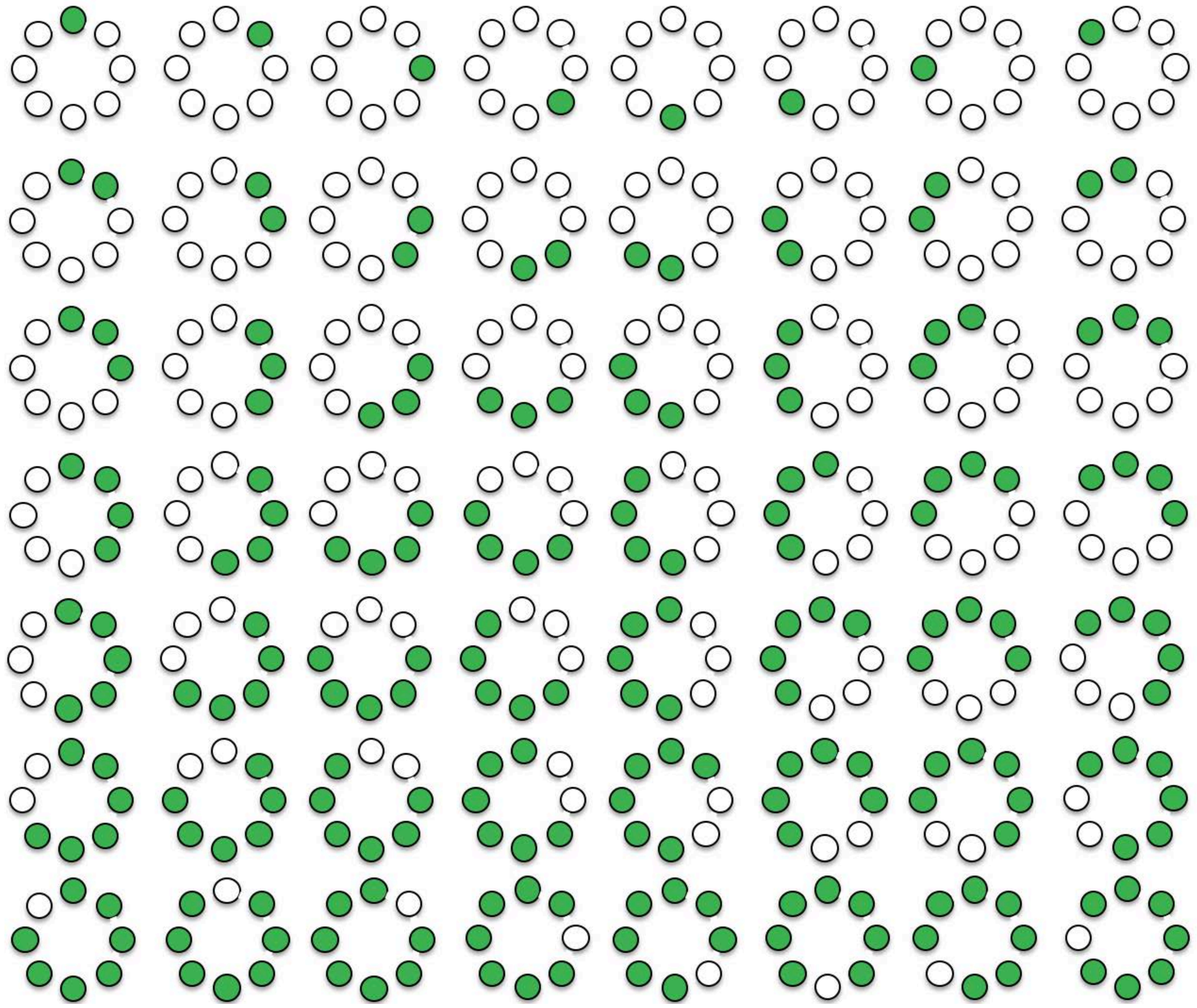
Uniform pattern: contains **at most two bitwise transitions (U) from 0 to 1** (or vice versa) when the bit pattern is traversed circularly



Source: Domingo Mery

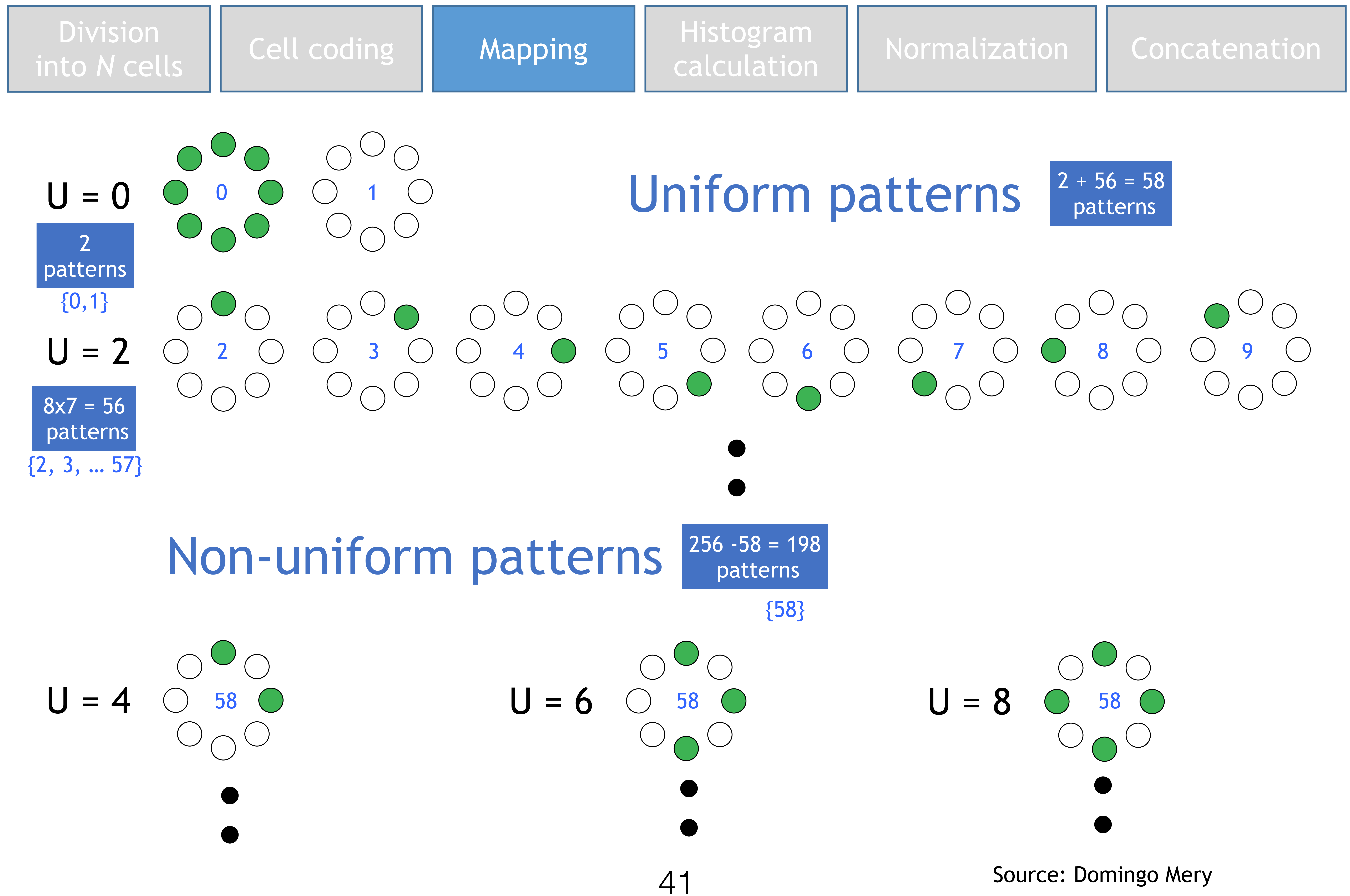
Uniform patterns

U = 0 

U = 2 

Uniform patterns account for almost 90% of all patterns.

Source: Domingo Mery



Division into N cells
Cell coding
Mapping
Histogram calculation
Normalization
Concatenation
U = 0
0
1
2 patterns
{0,1}
Uniform patterns
2 + 56 = 58 patterns
U = 2
2
3
4
5
6
7
8
9
8x7 = 56 patterns
{2, 3, … 57}
Non-uniform patterns
256 -58 = 198 patterns
{58}
U = 4
58
U = 6
58
U = 8
58
Source: Domingo Mery

Division into *N* cells | Cell coding | **Mapping** | Histogram calculation | Normalization | Concatenation

Result of cell code mapping

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

Cell

	230	207	25	168	
	243	255	255	119	

Coded cell

	58	46	58	58	
	23	0	0	58	

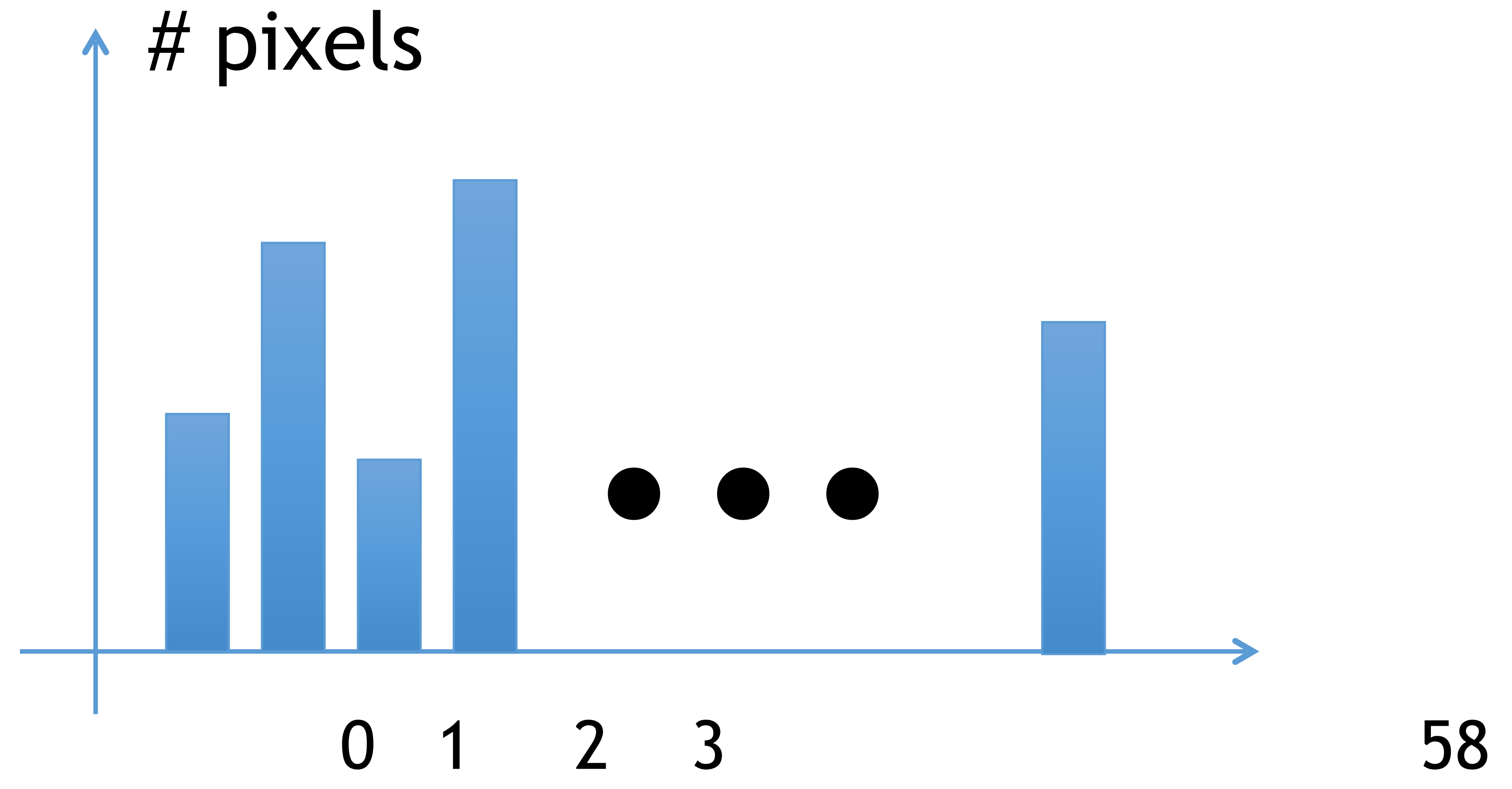
Mapped cell

Source: Domingo Mery

Division into *N* cells | Cell coding | Mapping | **Histogram calculation** | Normalization | Concatenation

Mapped cell

	58	46	58	58	
	23	0	0	58	

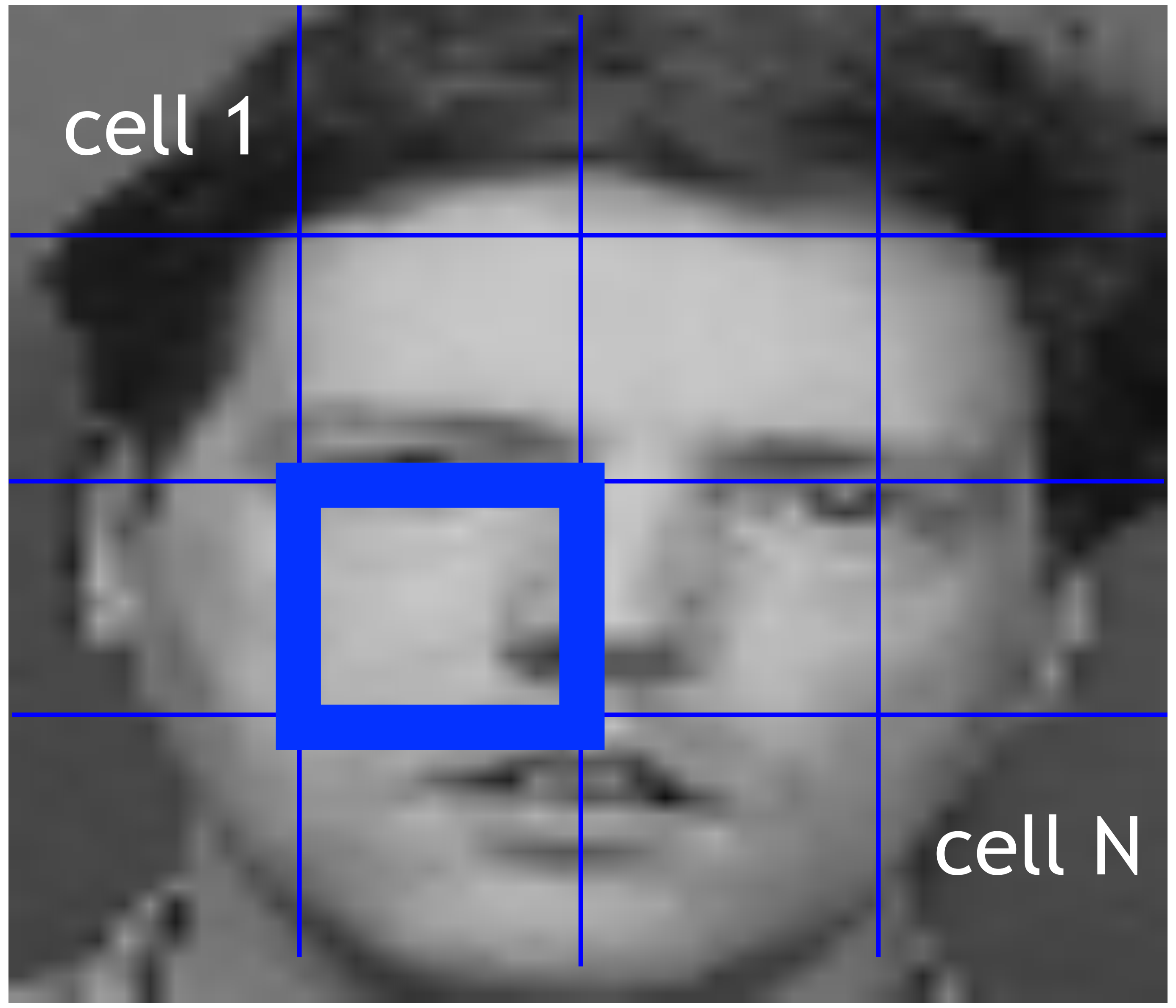


- Each cell is represented as 59-digit LBP descriptor
- Similar textures have similar histograms.

Source: Domingo Mery

Division into *N* cells | Cell coding | Mapping | Histogram calculation | **Normalization** | **Concatenation**

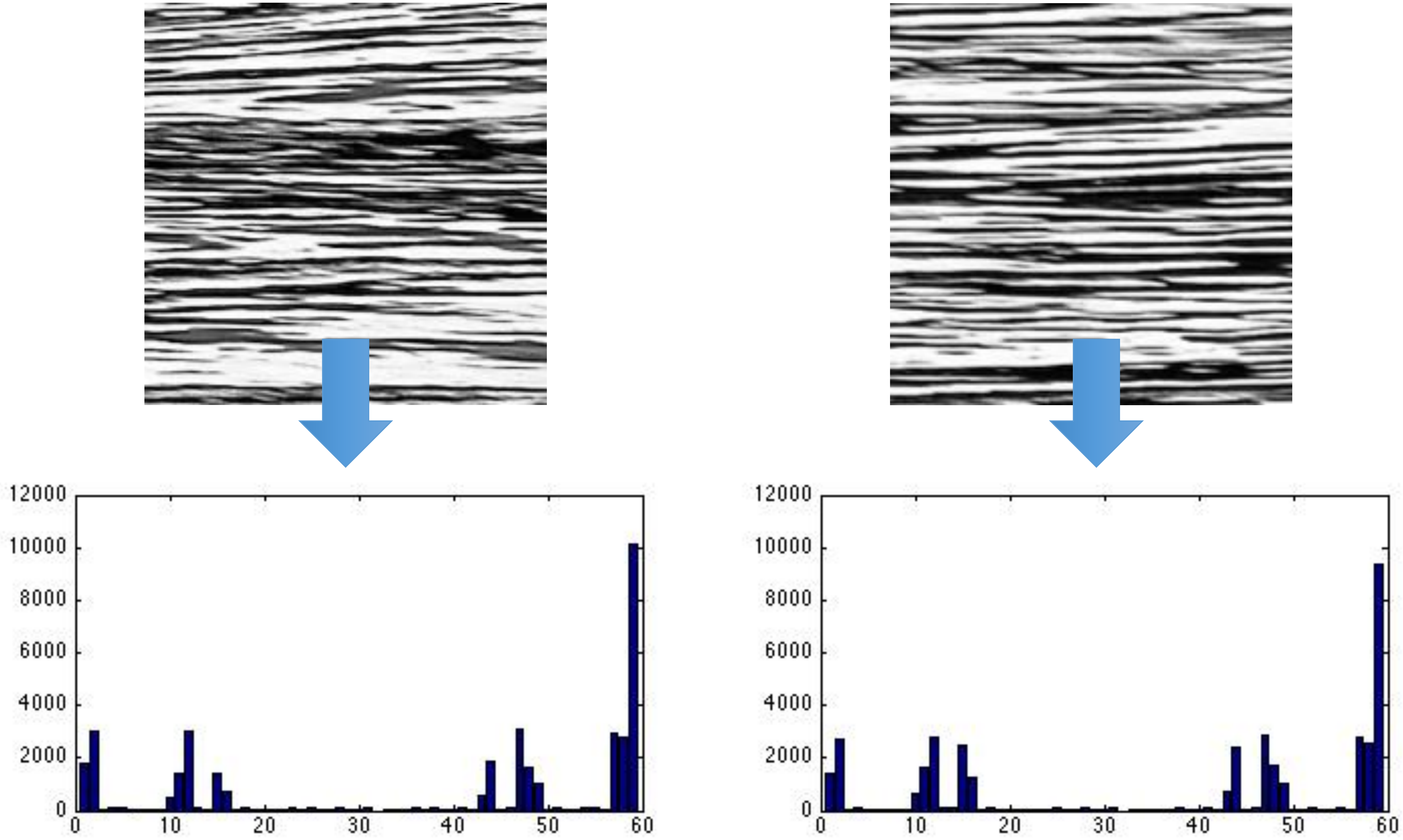
- Normalization of histograms makes LBP descriptors size-invariant
- Concatenation of all cell histograms provides the image LPB descriptor



Source: Domingo Mery

Local Binary Patterns

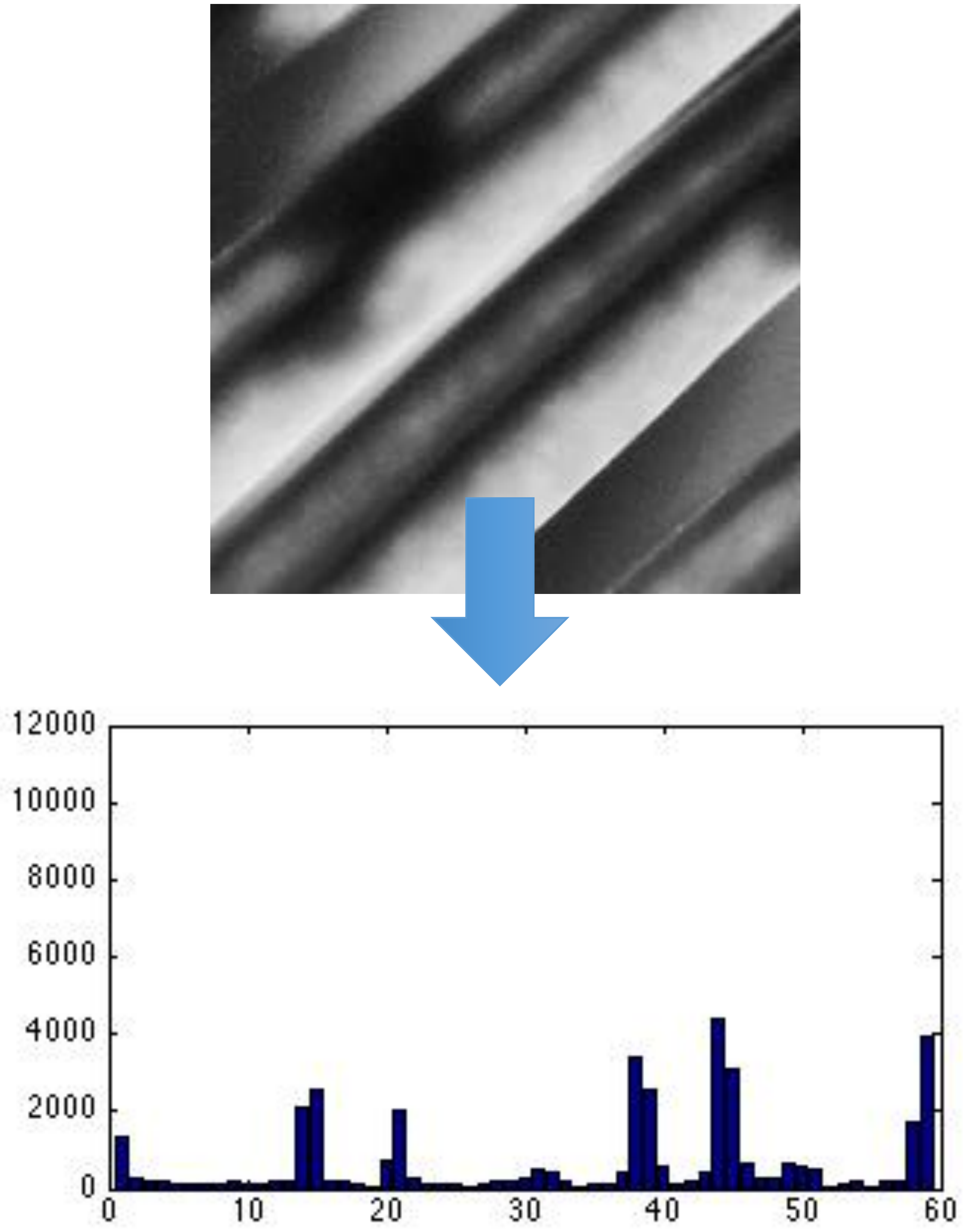
Similar textures have similar histograms

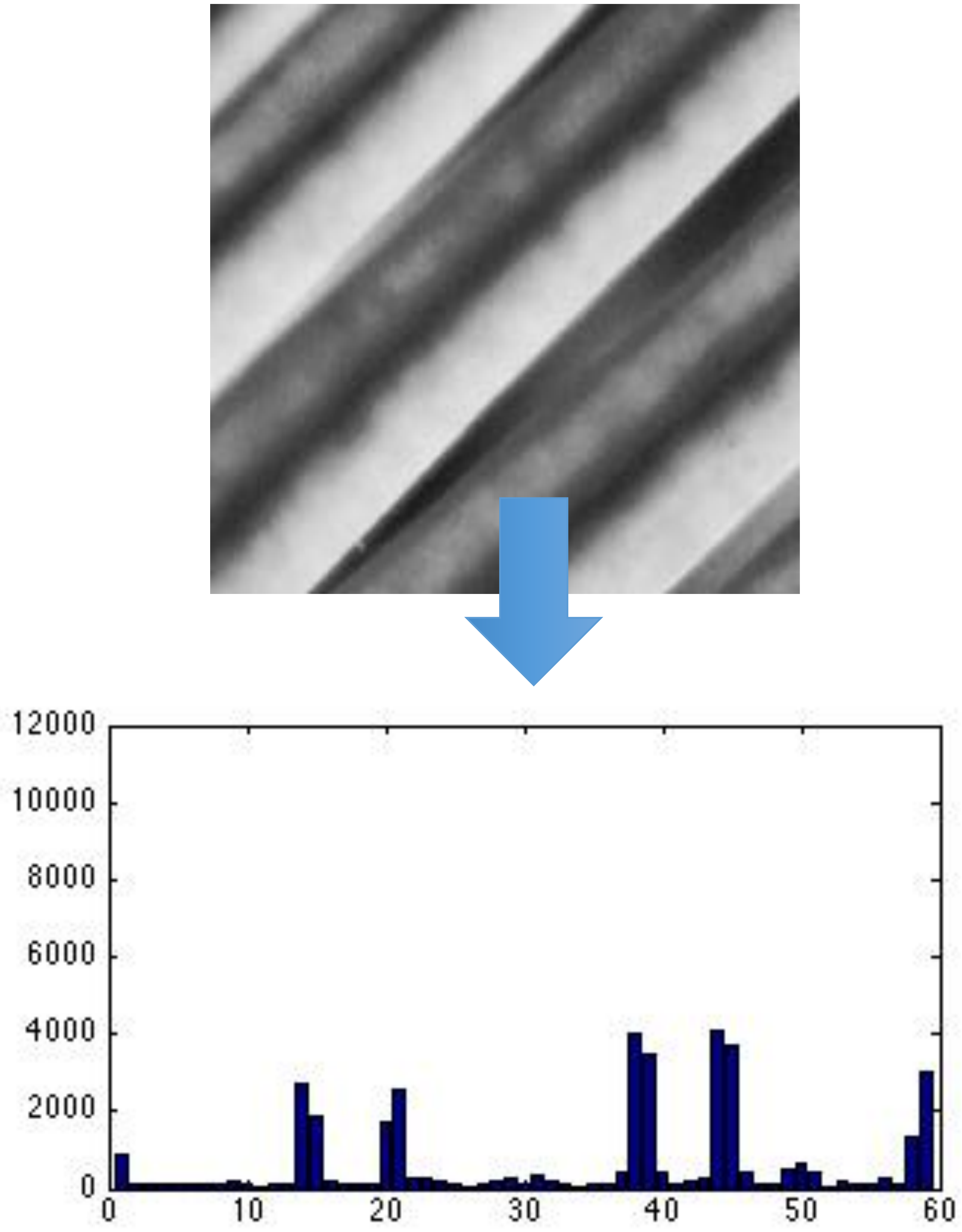


Source: Domingo Mery

Local Binary Patterns

Similar textures have similar histograms

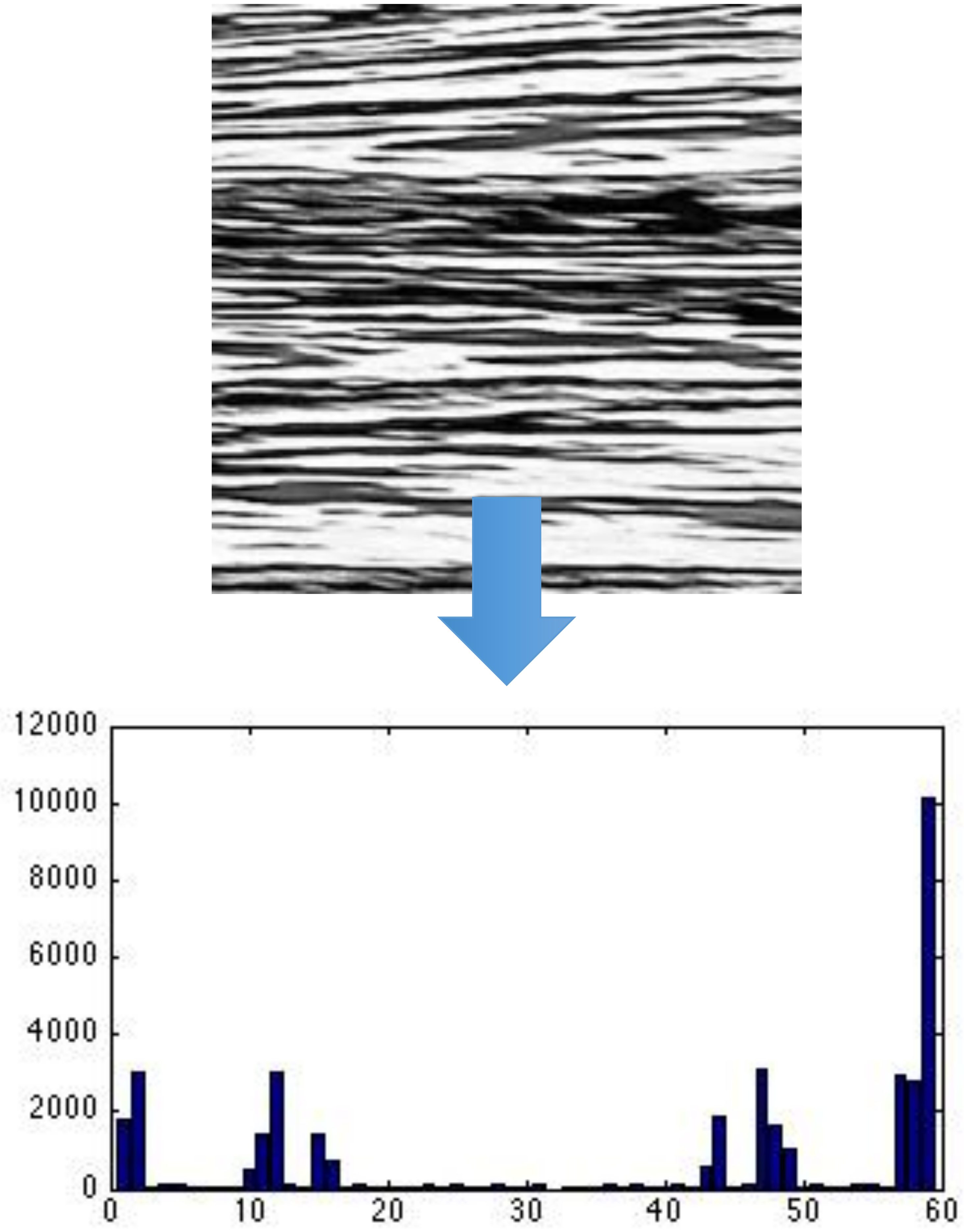


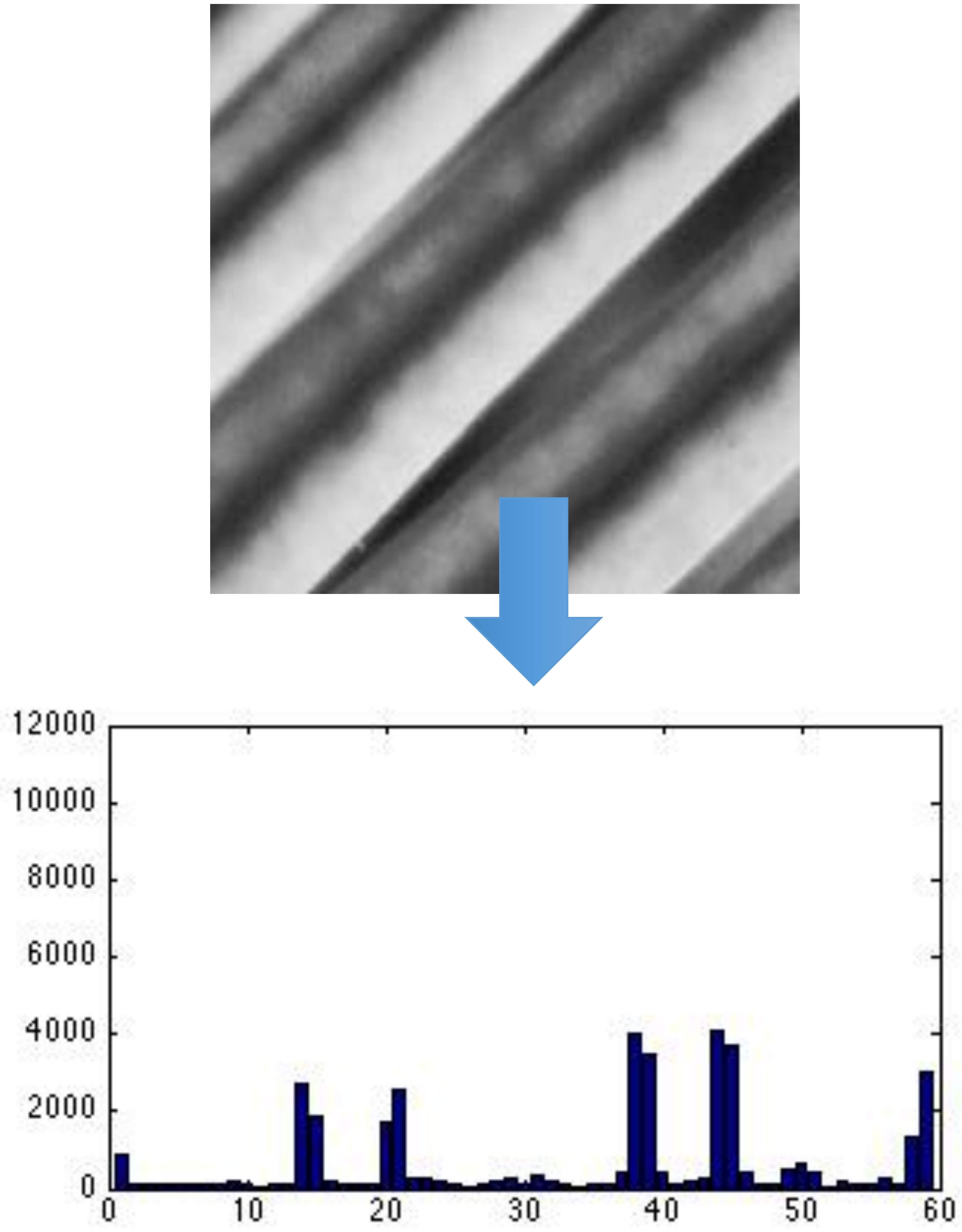


Source: Domingo Mery

Local Binary Patterns

Similar textures have similar histograms

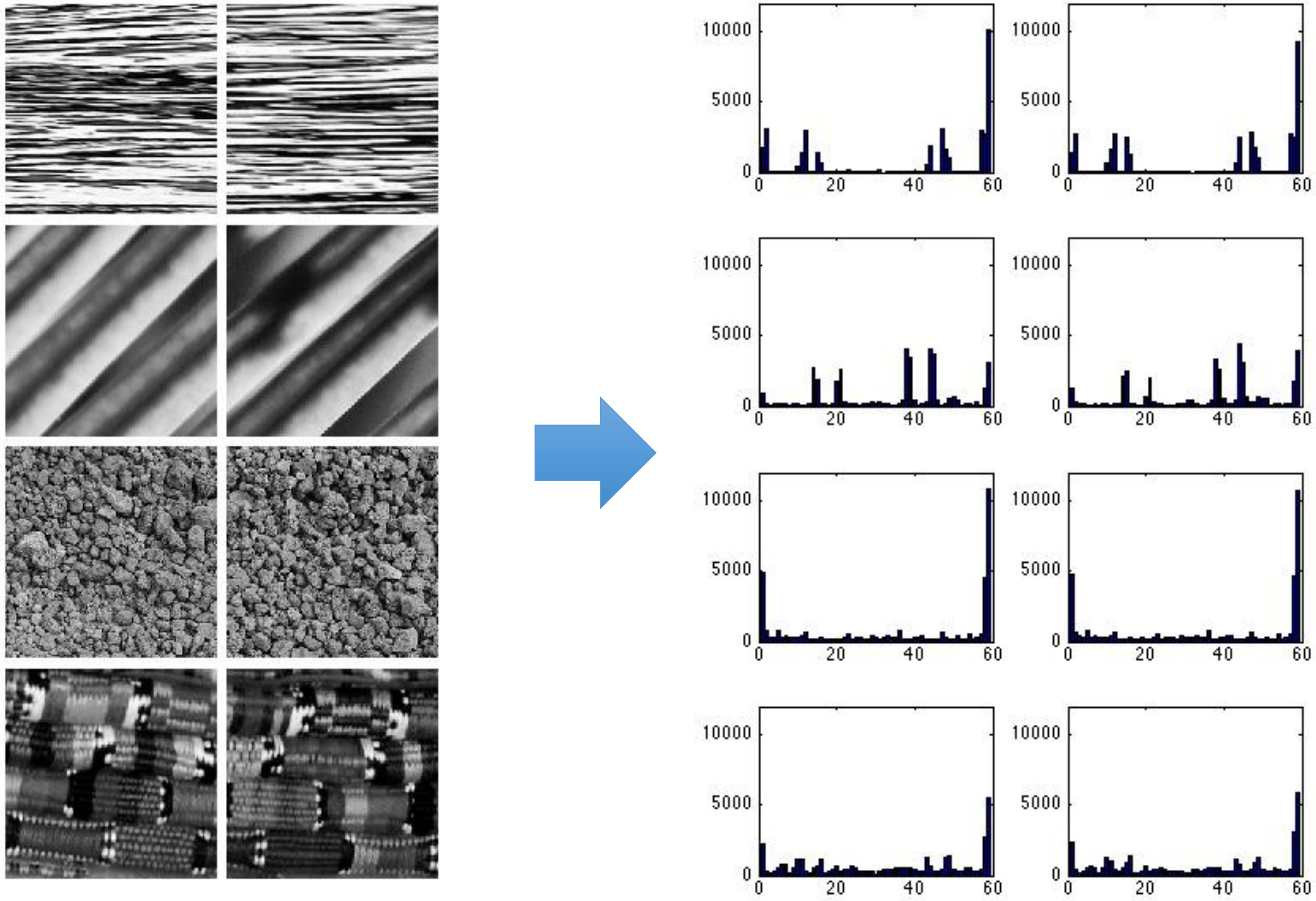




Source: Domingo Mery

Local Binary Patterns

Similar textures have similar histograms



Source: Domingo Mery

LBP for face recognition

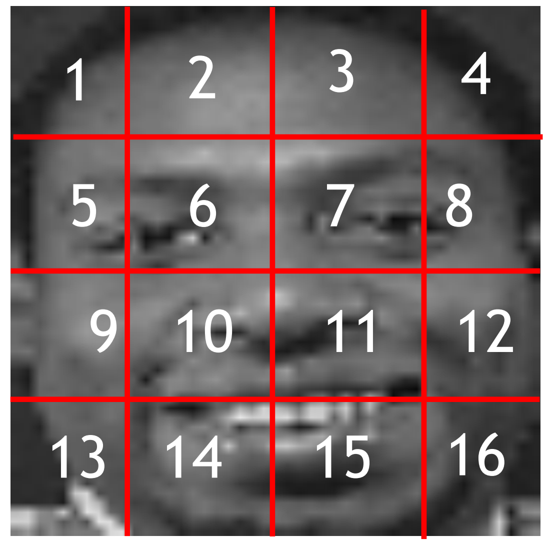
In the training set there are *k* classes.

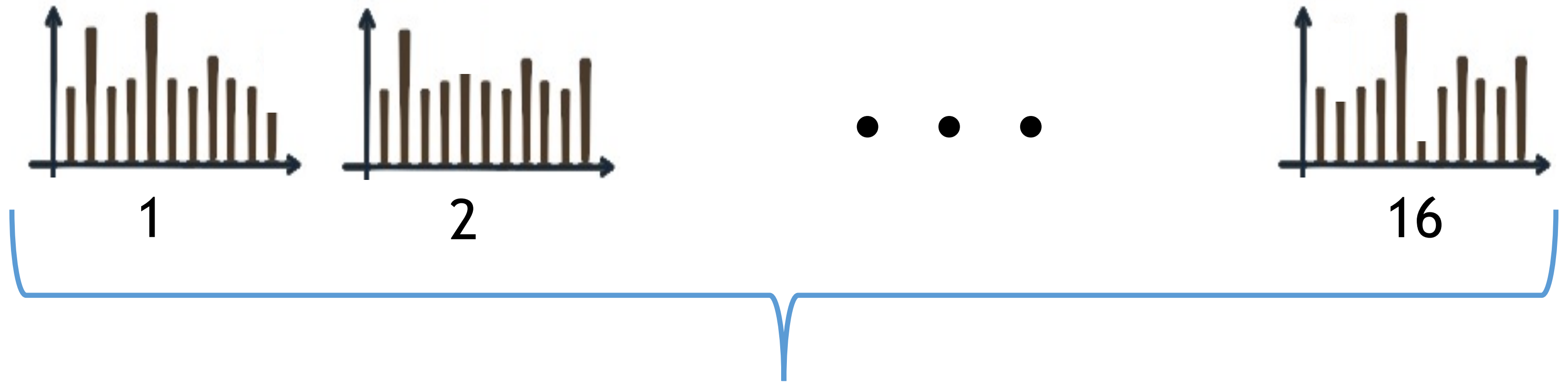
For each class we have *n* training images.

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

Each image in partitioned into 16 cells.

In each cell we extract LBP features.





A face is described using a feature of 16 x 59 = 944 elements

Source: Domingo Mery

LBP for face recognition

Training Data

Table with:

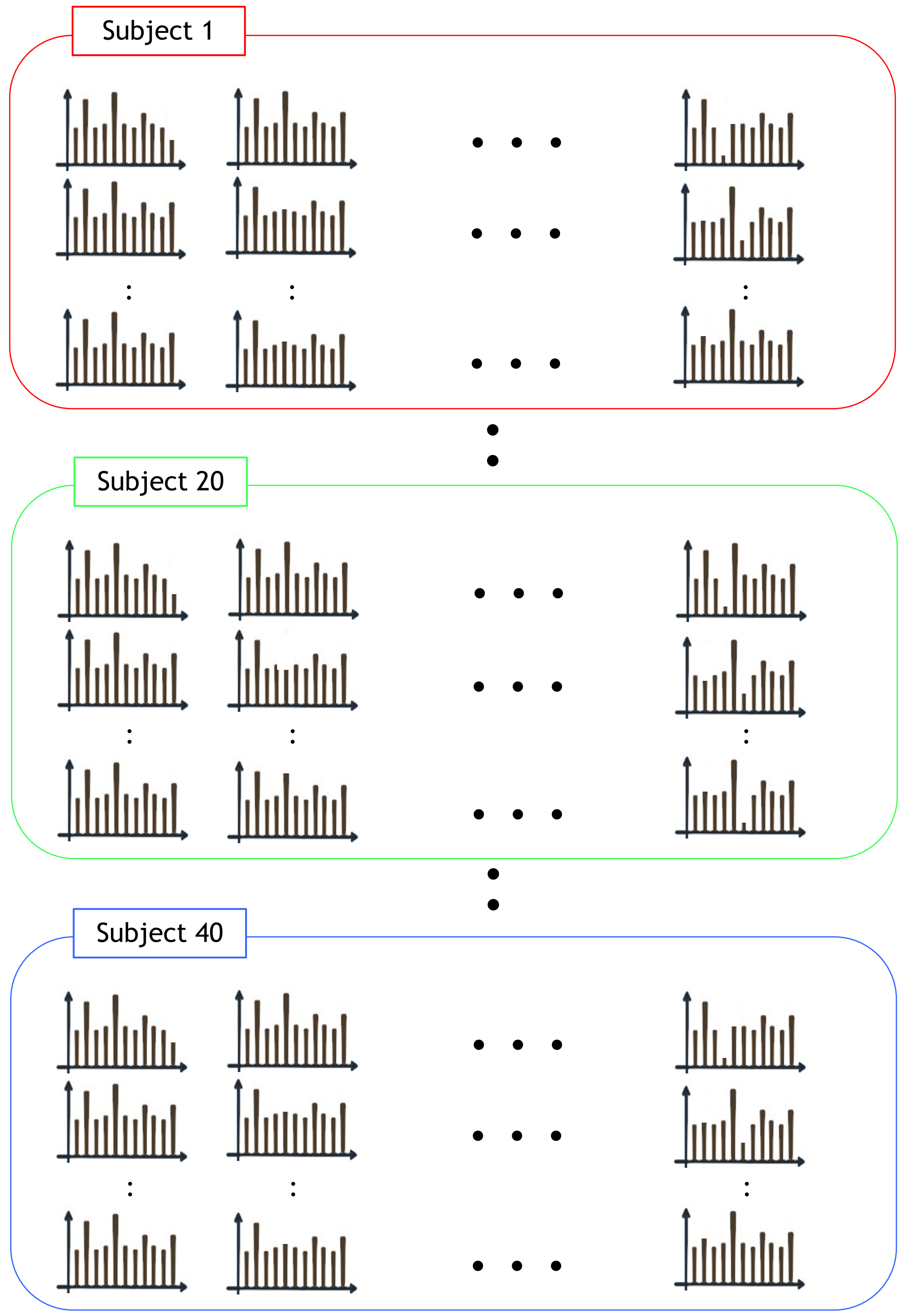
9x40 = 360 rows

and

16 x 59 = 944 columns

Source: Domingo Mery

LBP for face recognition

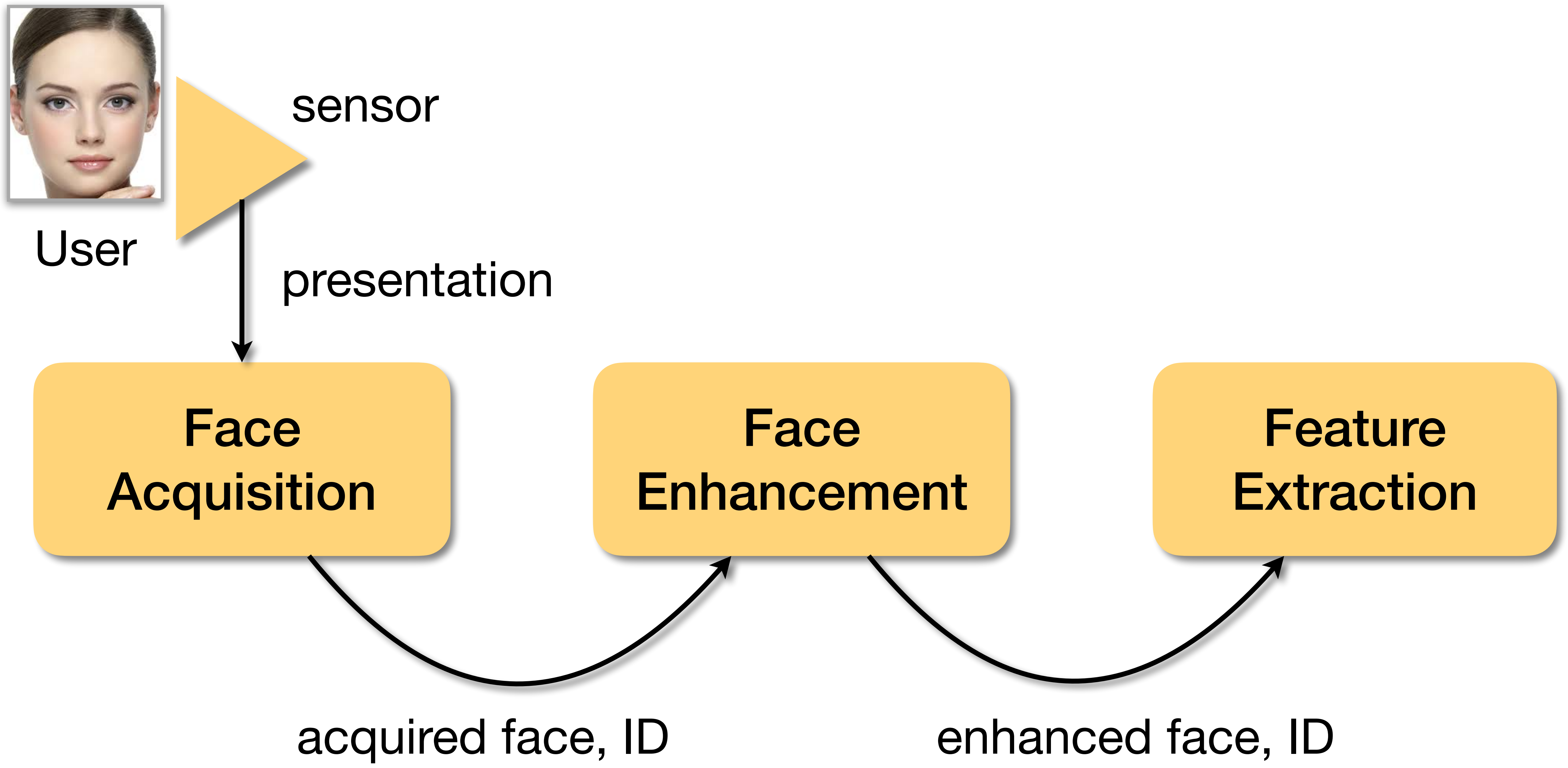


Source: Domingo Mery

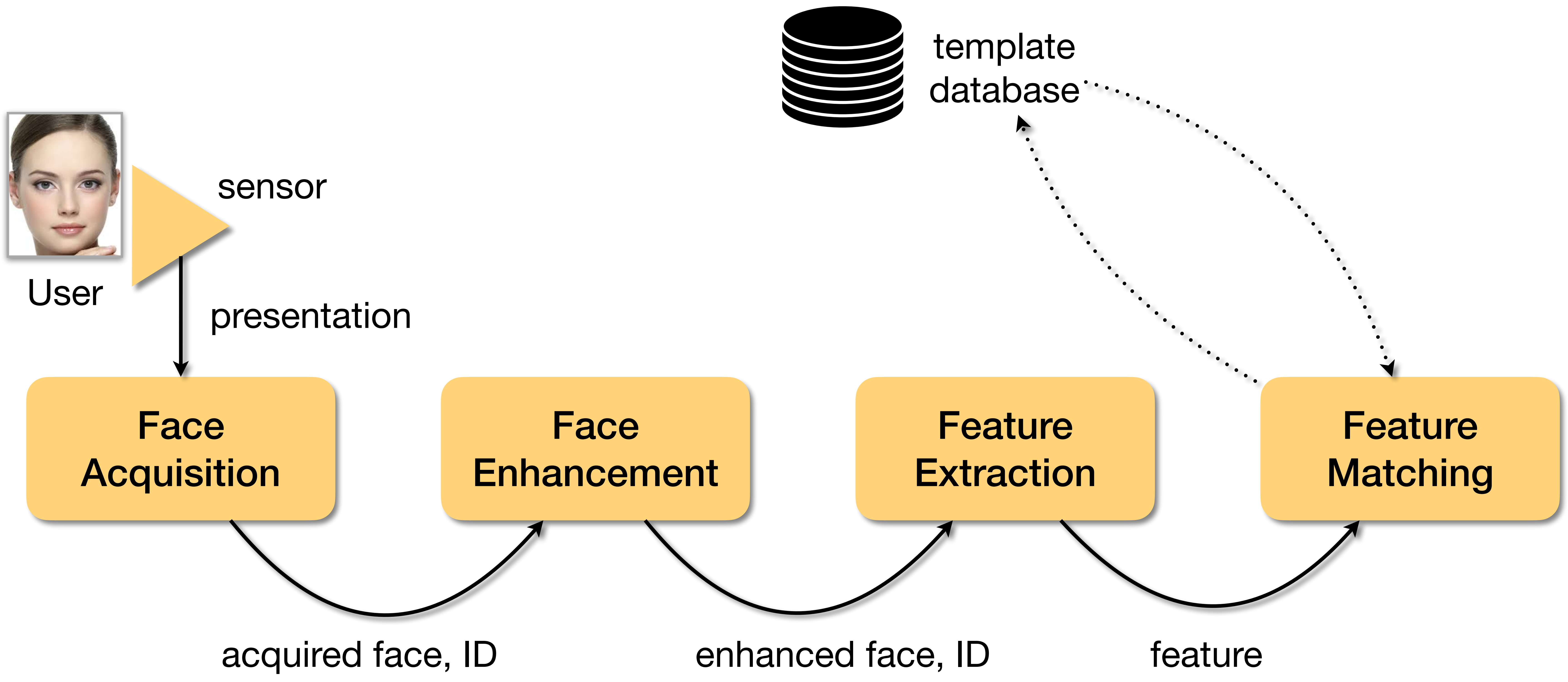
LBP for face recognition

Source: Domingo Mery

Face Recognition

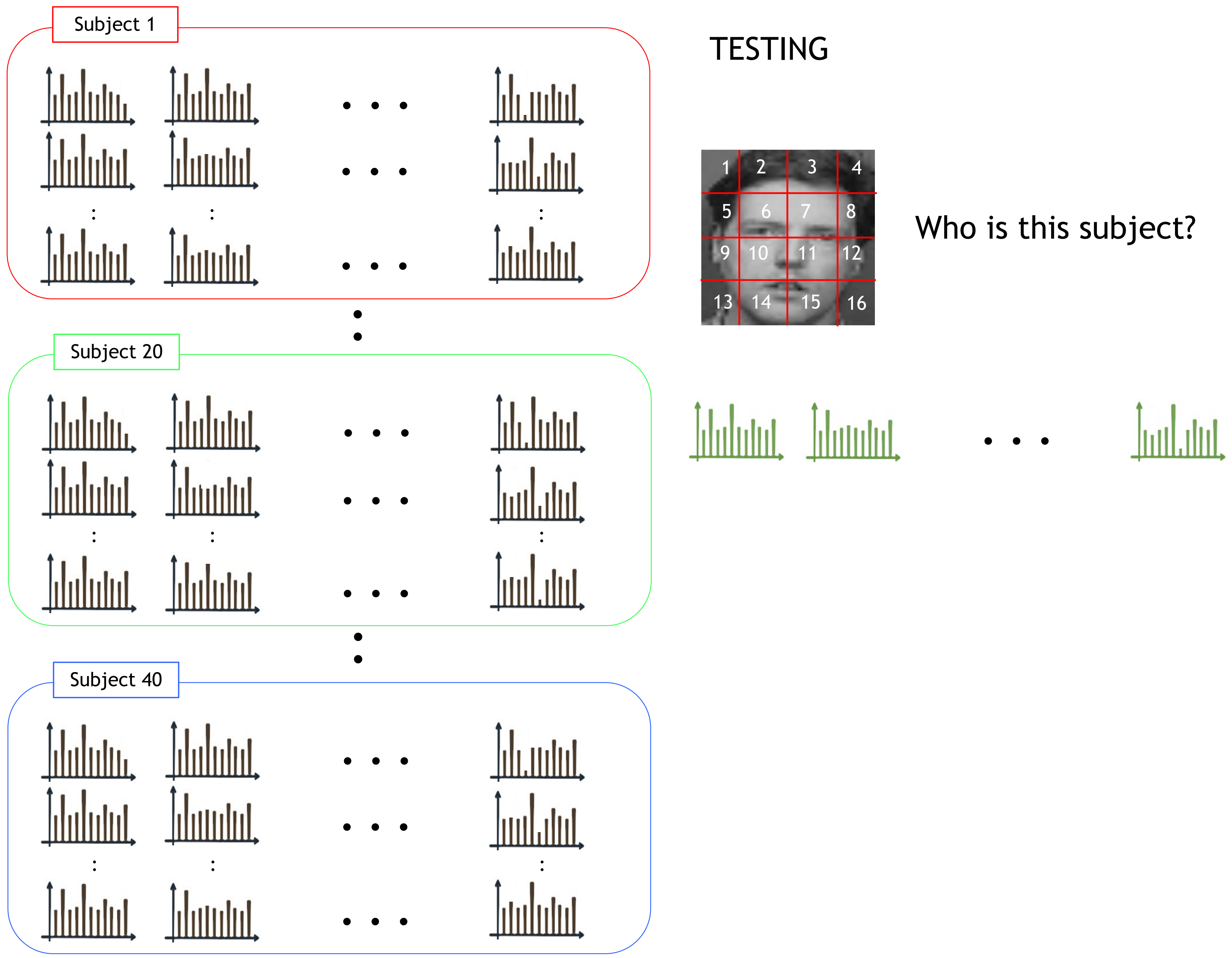


Face Recognition



LOYOLA
UNIVERSITY CHICAGO

LBP for face recognition (Feature Matching)

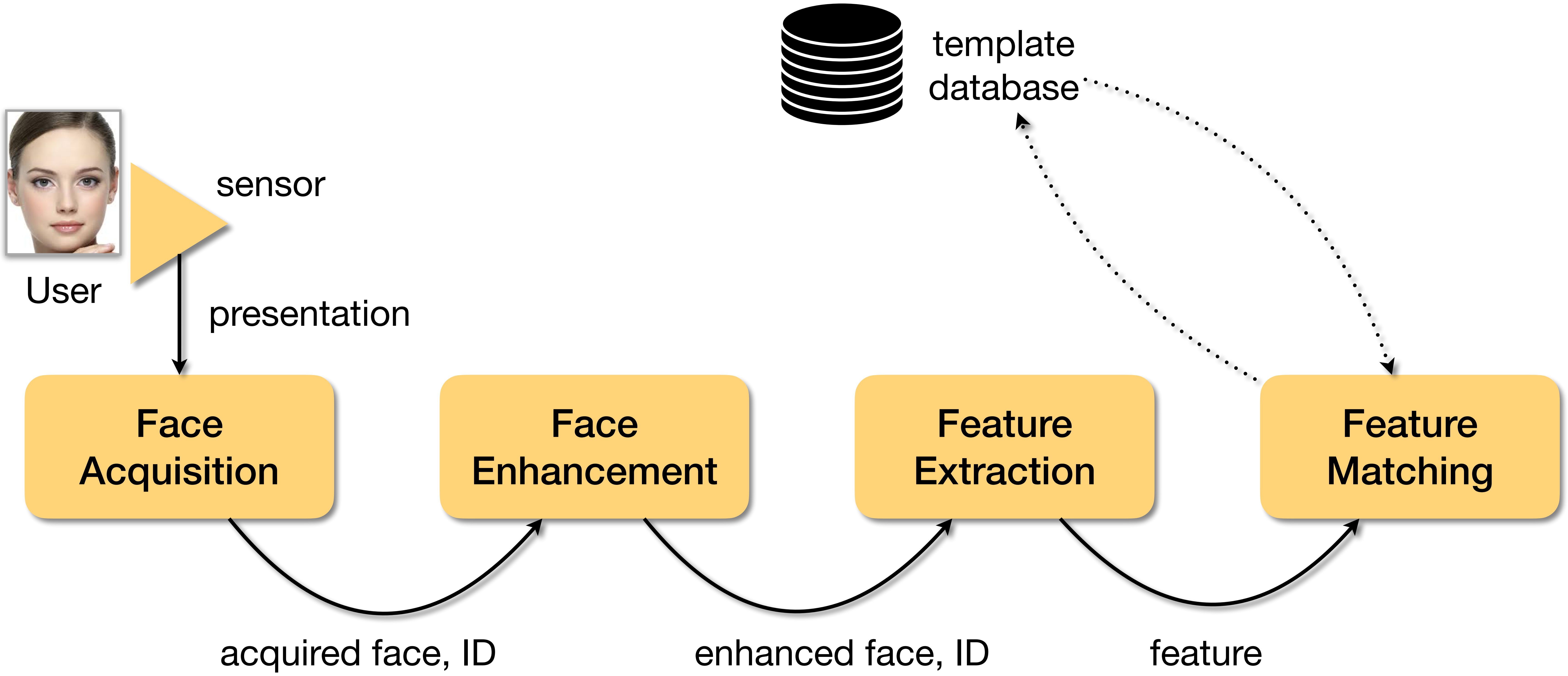


Source: Domingo Mery

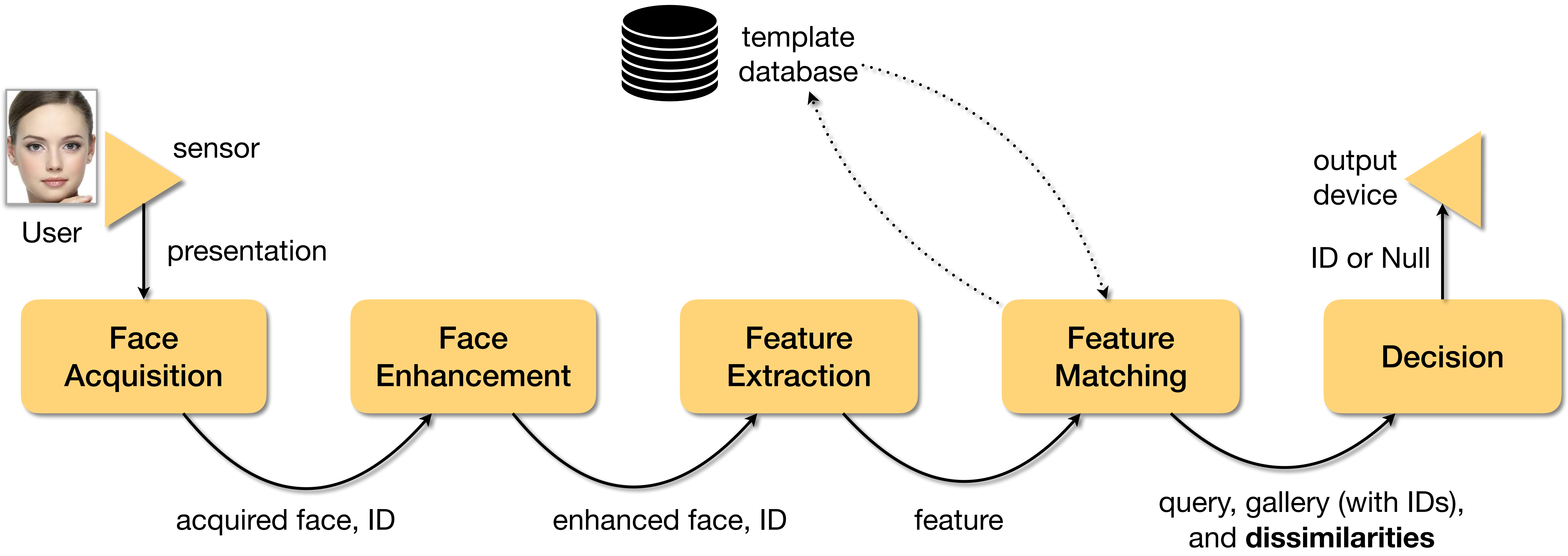
LBP for face recognition (Feature Matching)

Source: Domingo Mery

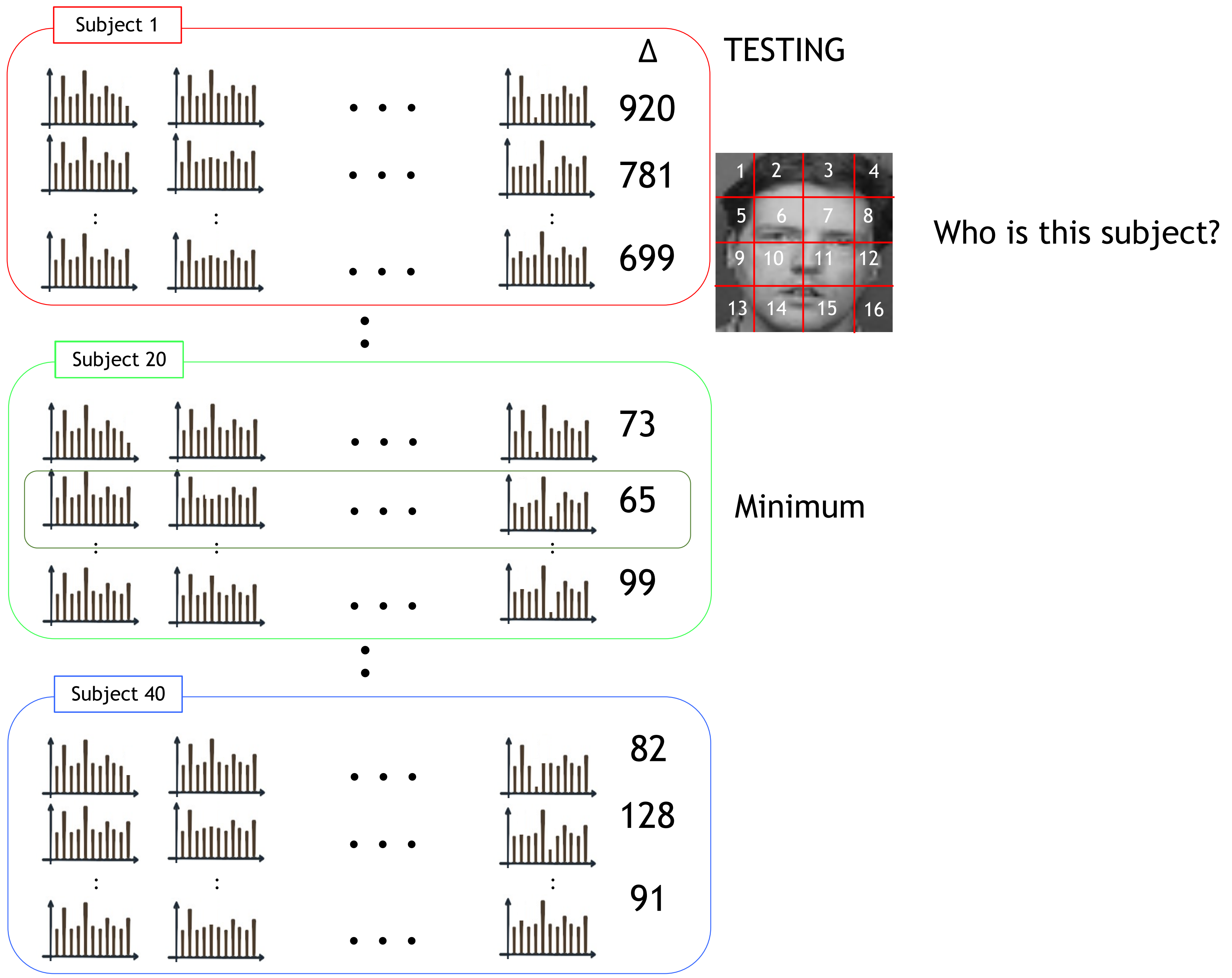
Face Recognition



Face Recognition



LBP for face recognition (Decision)



Source: Domingo Mery

Feature Extraction

Focus
2D-appearance-based methods.

Types
Handcrafted features from Computer Vision.

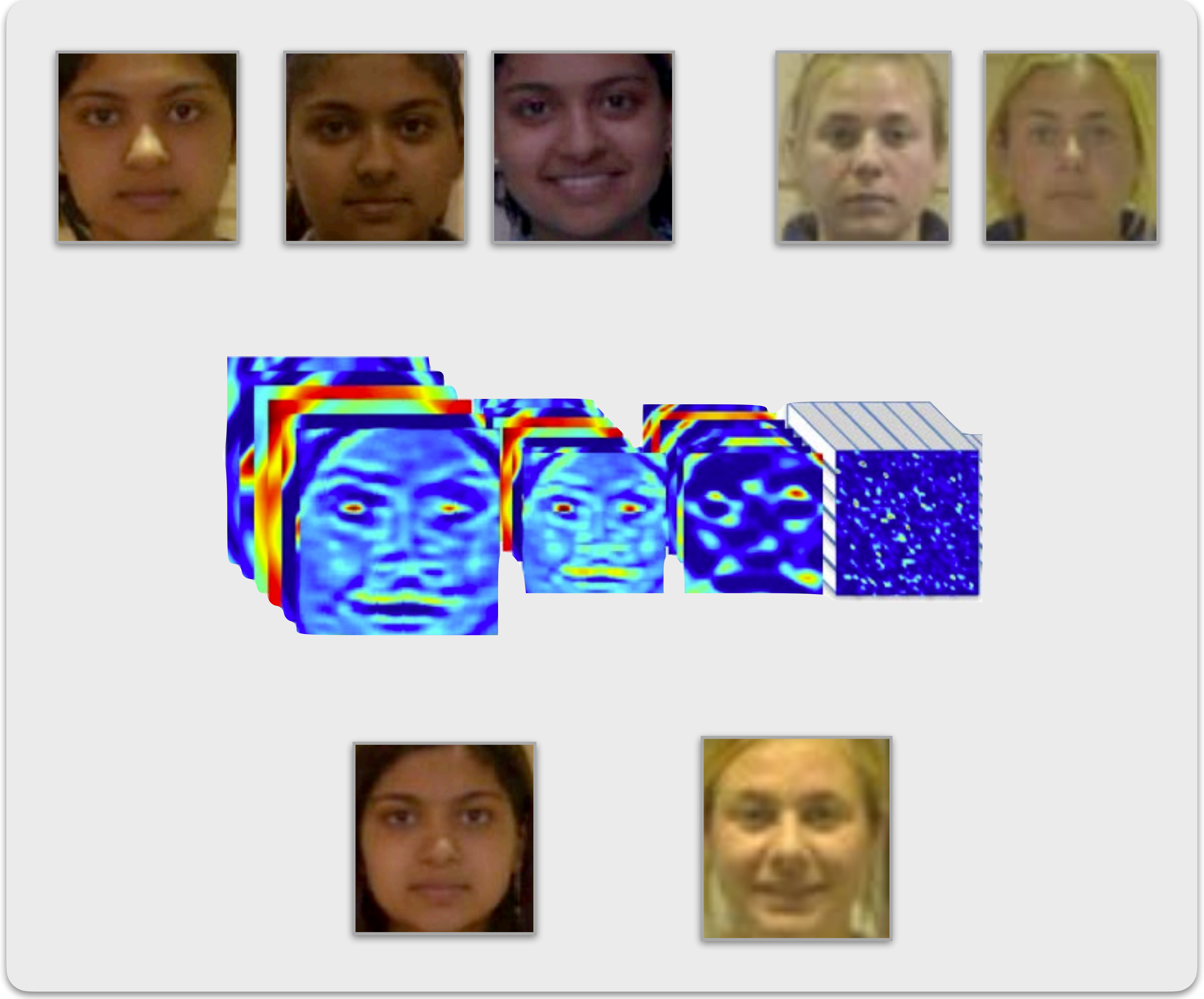
Data-driven learned features from Machine Learning.

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

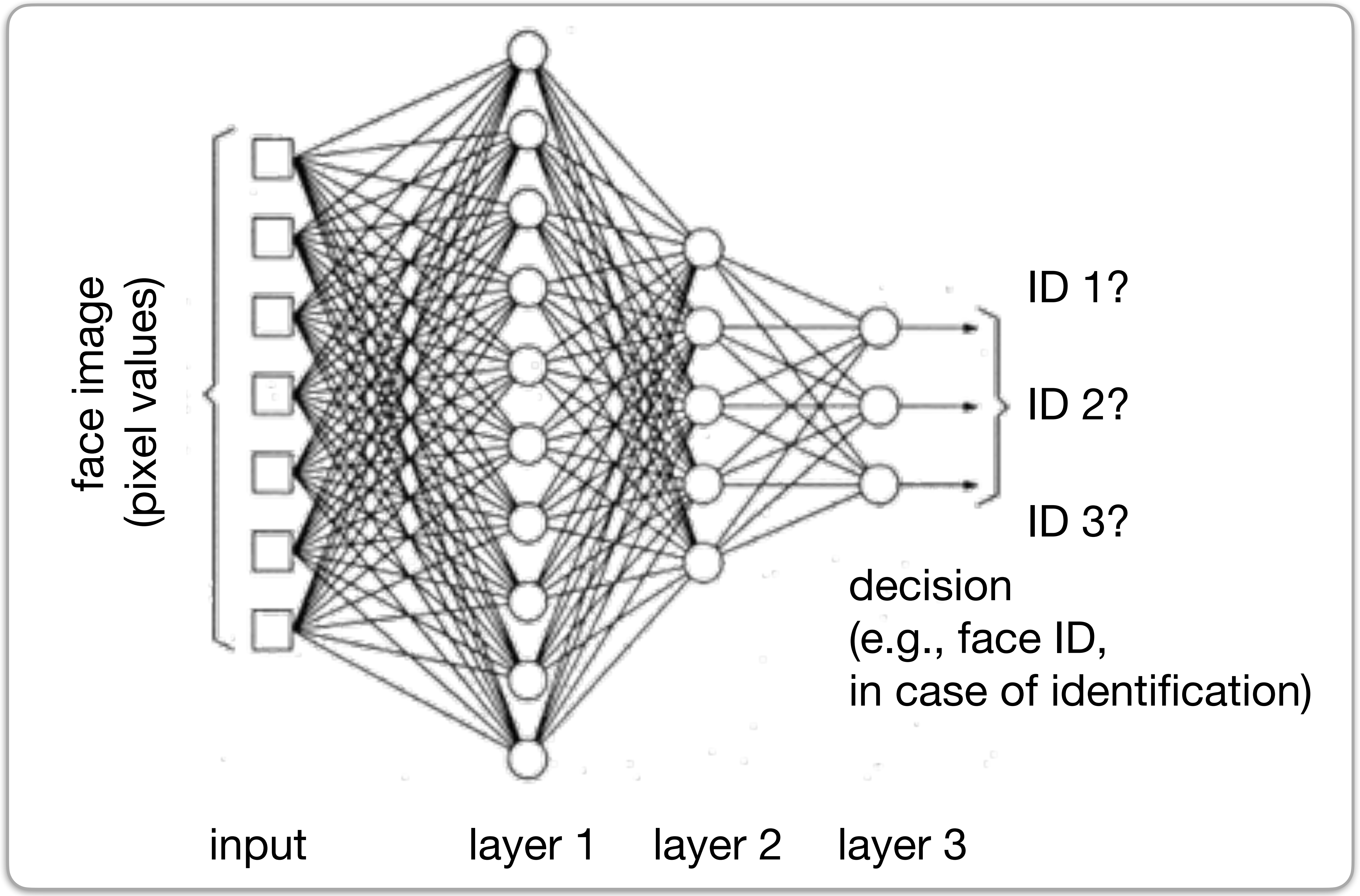
Feature Extraction

Deep Convolutional Neural Networks

From pixels to classification decision.

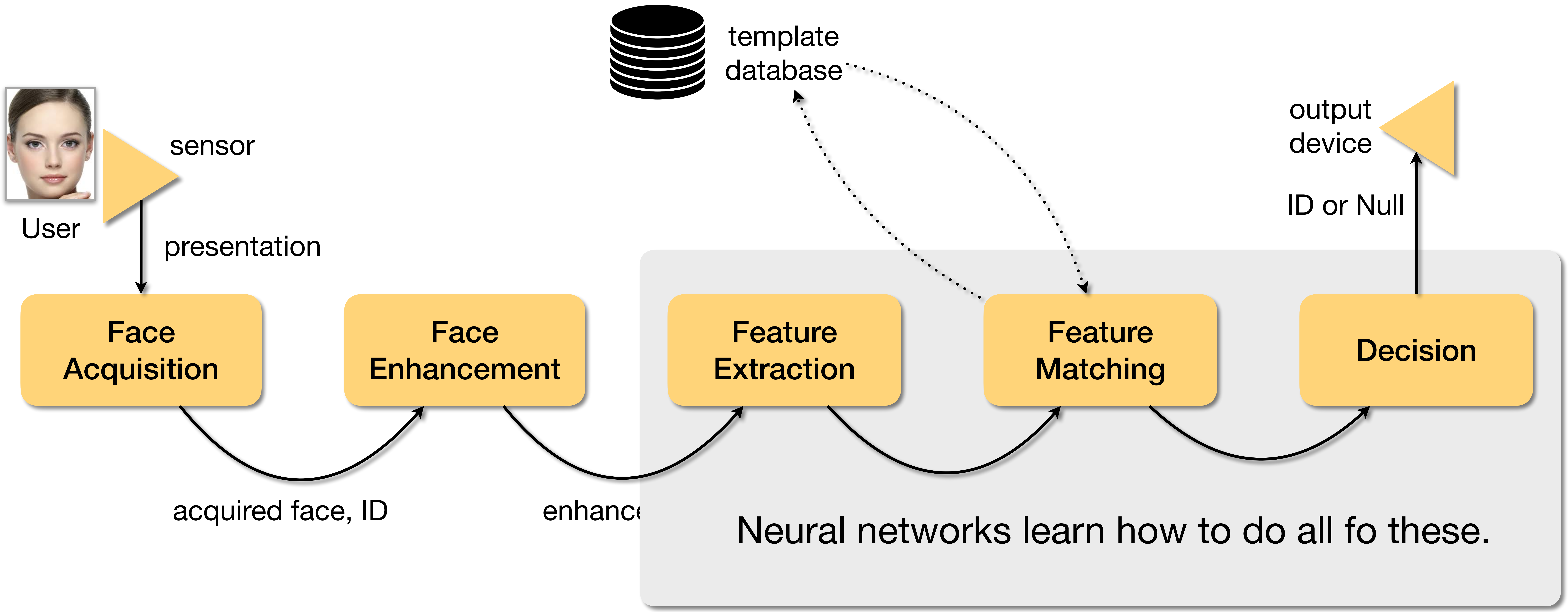
Hierarchy of feature extractors.

Each layer extracts features from previous layer.



LOYOLA UNIVERSITY CHICAGO

Face Recognition

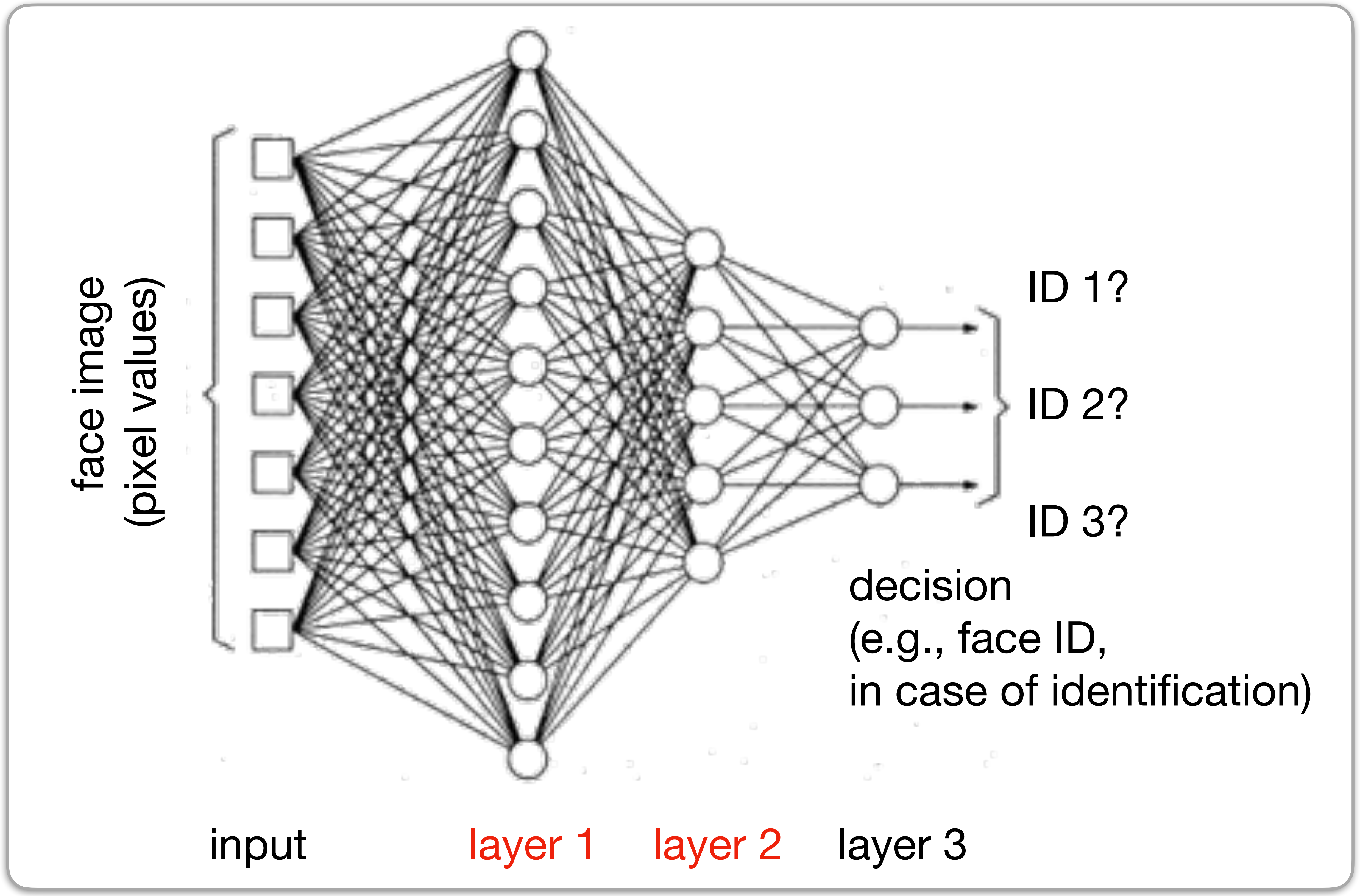


LOYOLA
UNIVERSITY CHICAGO

Data-driven Face Recognition

Deep Convolutional Neural Networks

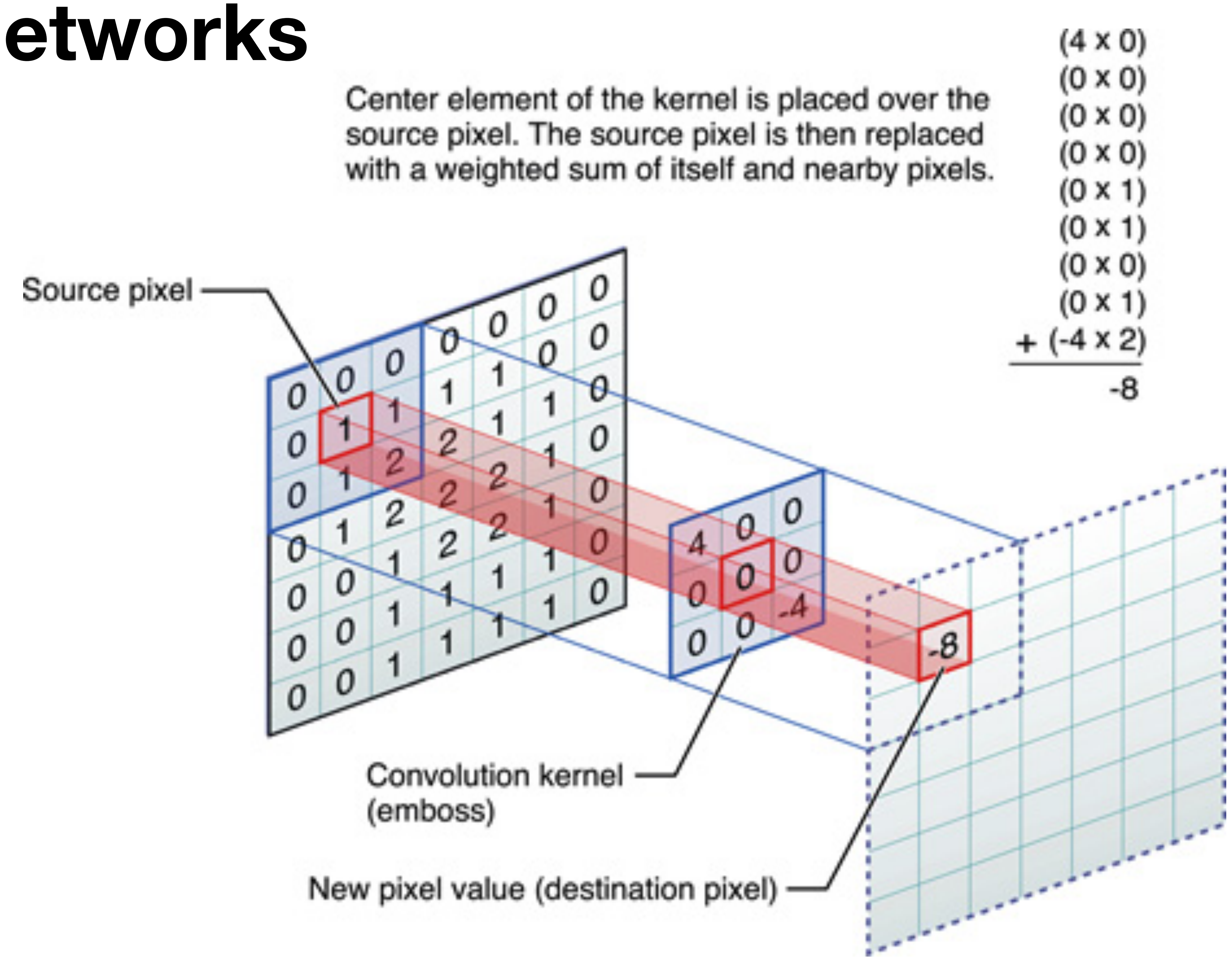
Convolutional Layers
E.g., layers 1 and 2.
Feature extractors are convolutional operations which are performed on the output of the previous layer.



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.

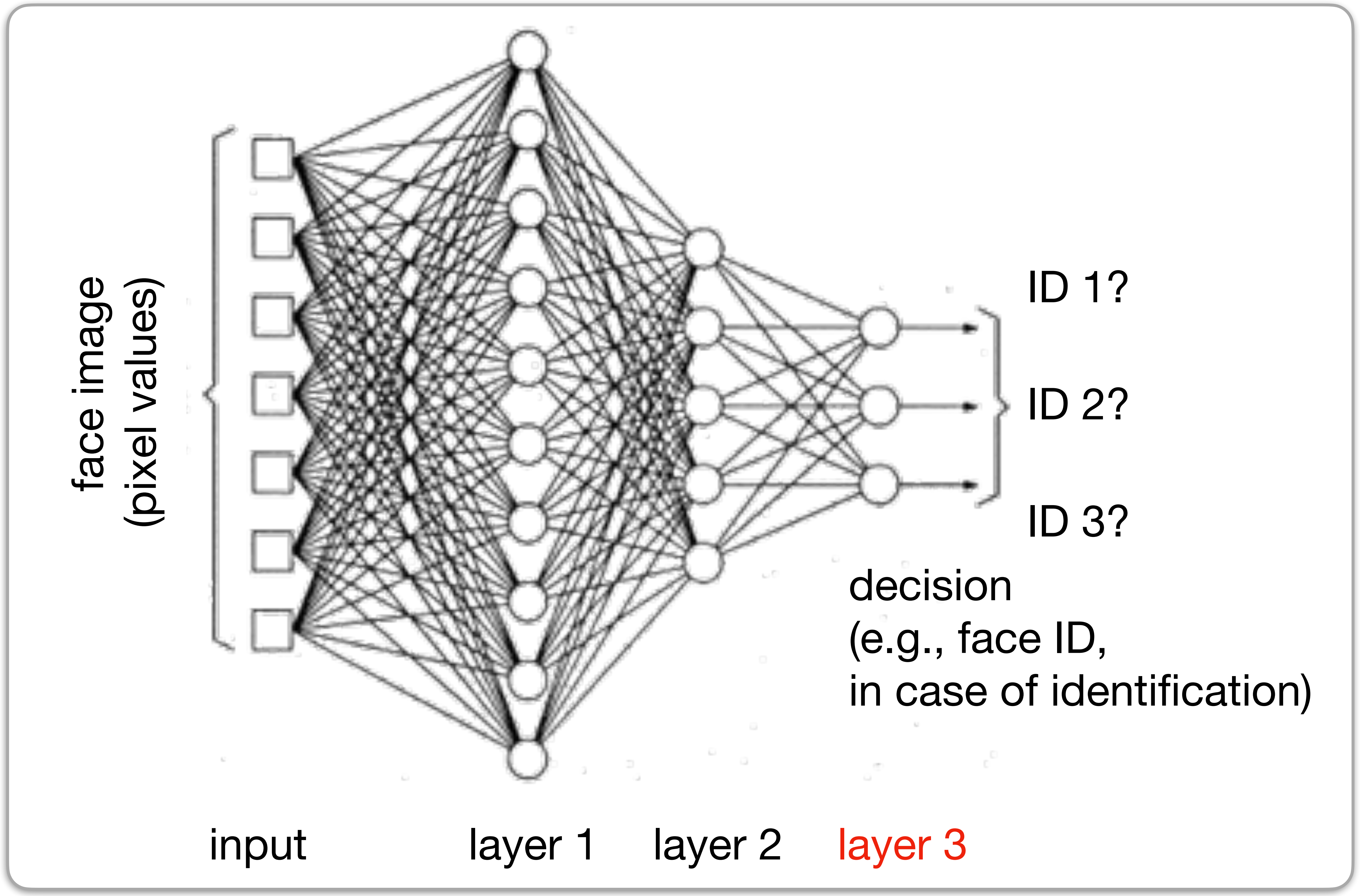


Source:https://developer.apple.com/library/archive/documentation/Performance/Conceptual/vImage/ConvolutionOperations/ConvolutionOperations.html

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).

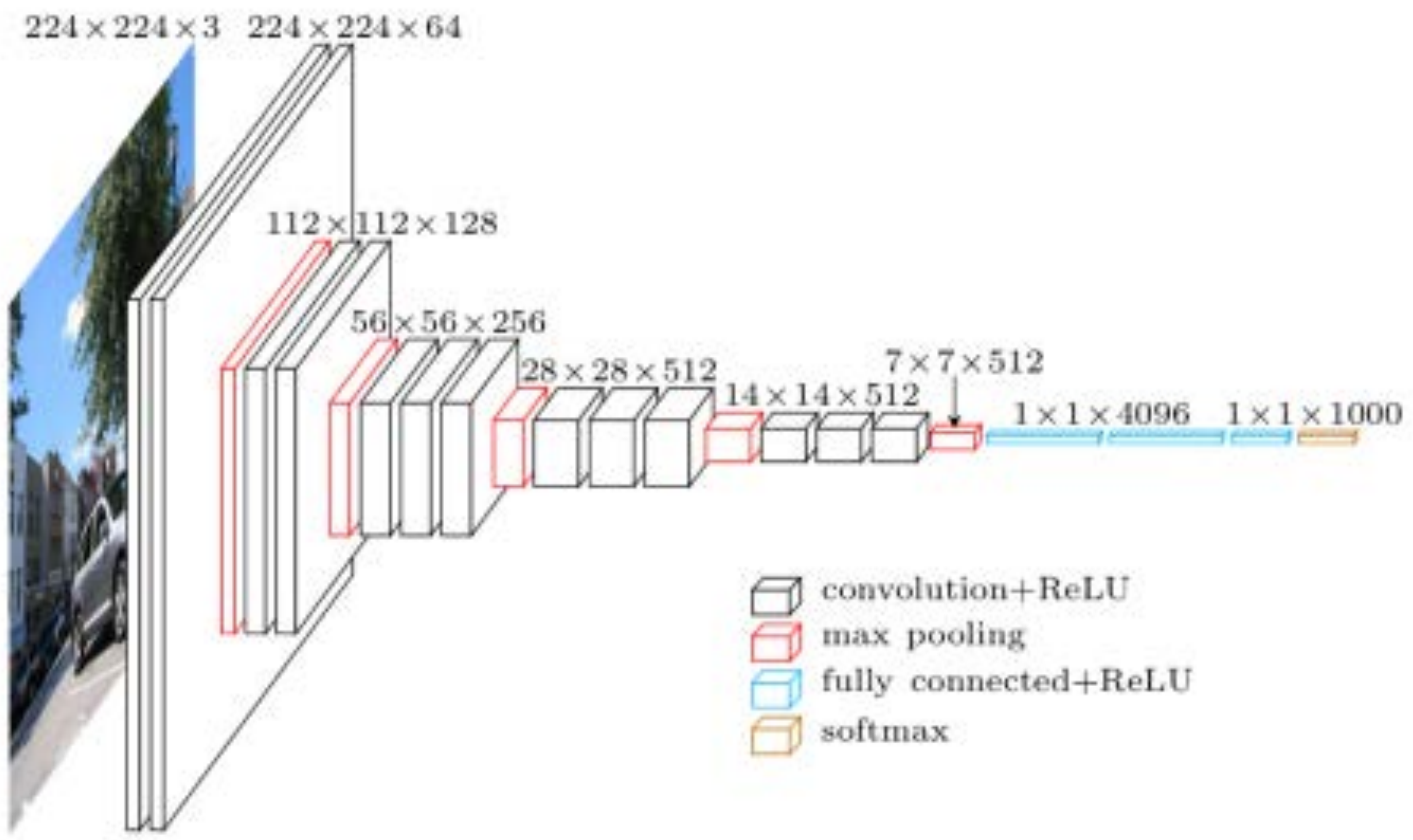


Data-driven Face Recognition

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

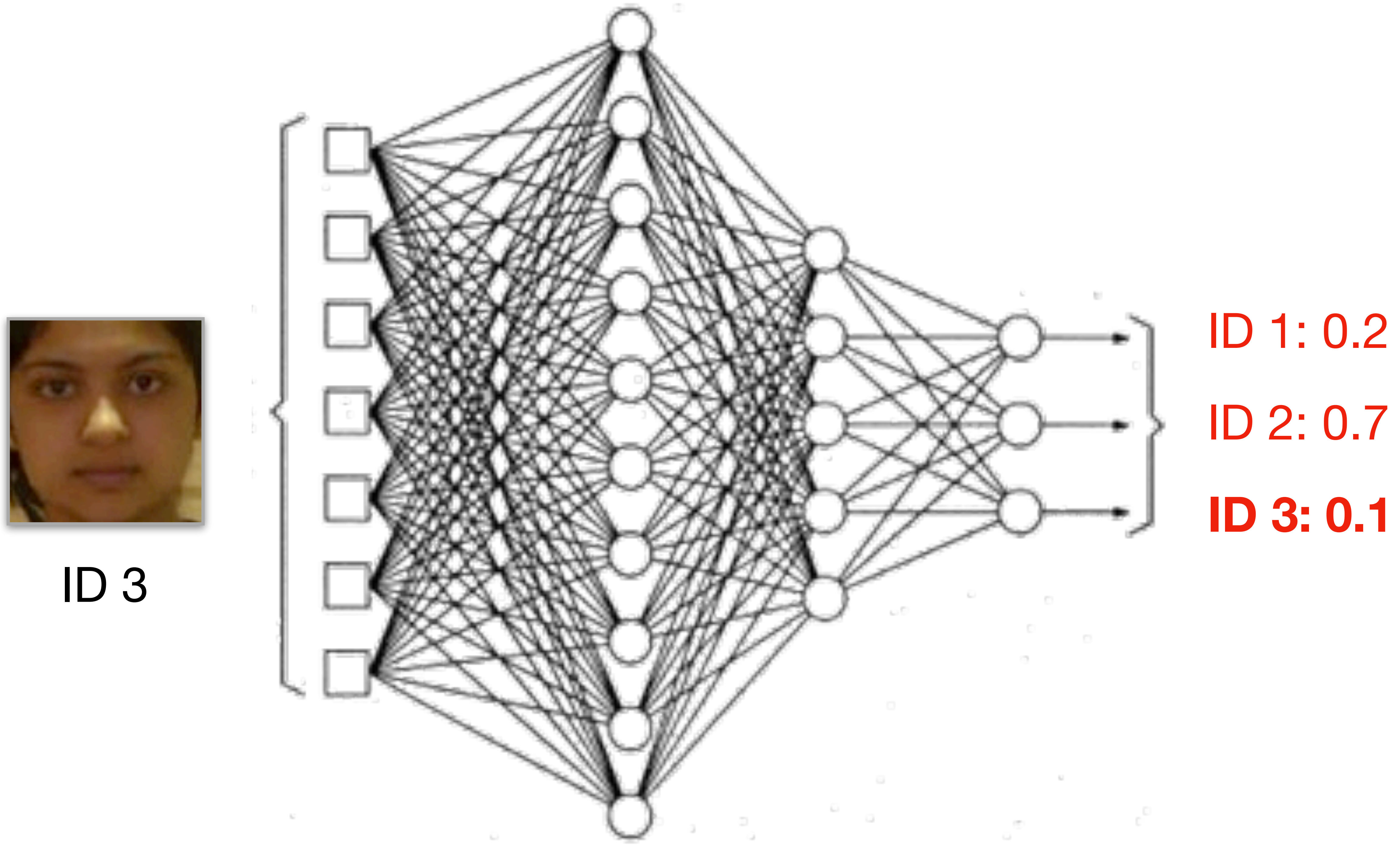


16 to 19 convolutional layers

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.

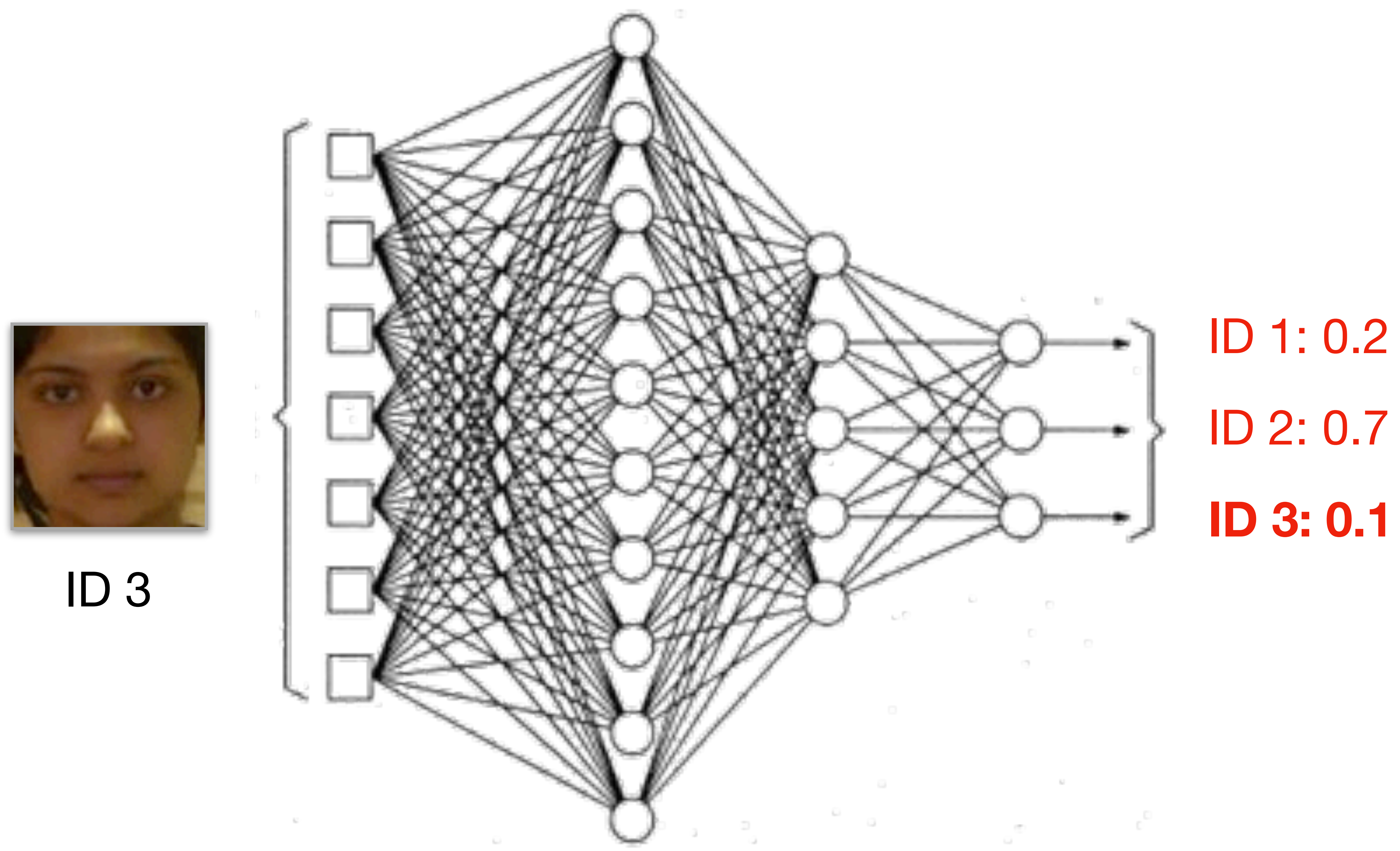


Random weights in the beginning.

Data-driven Face Recognition

Deep Learning

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

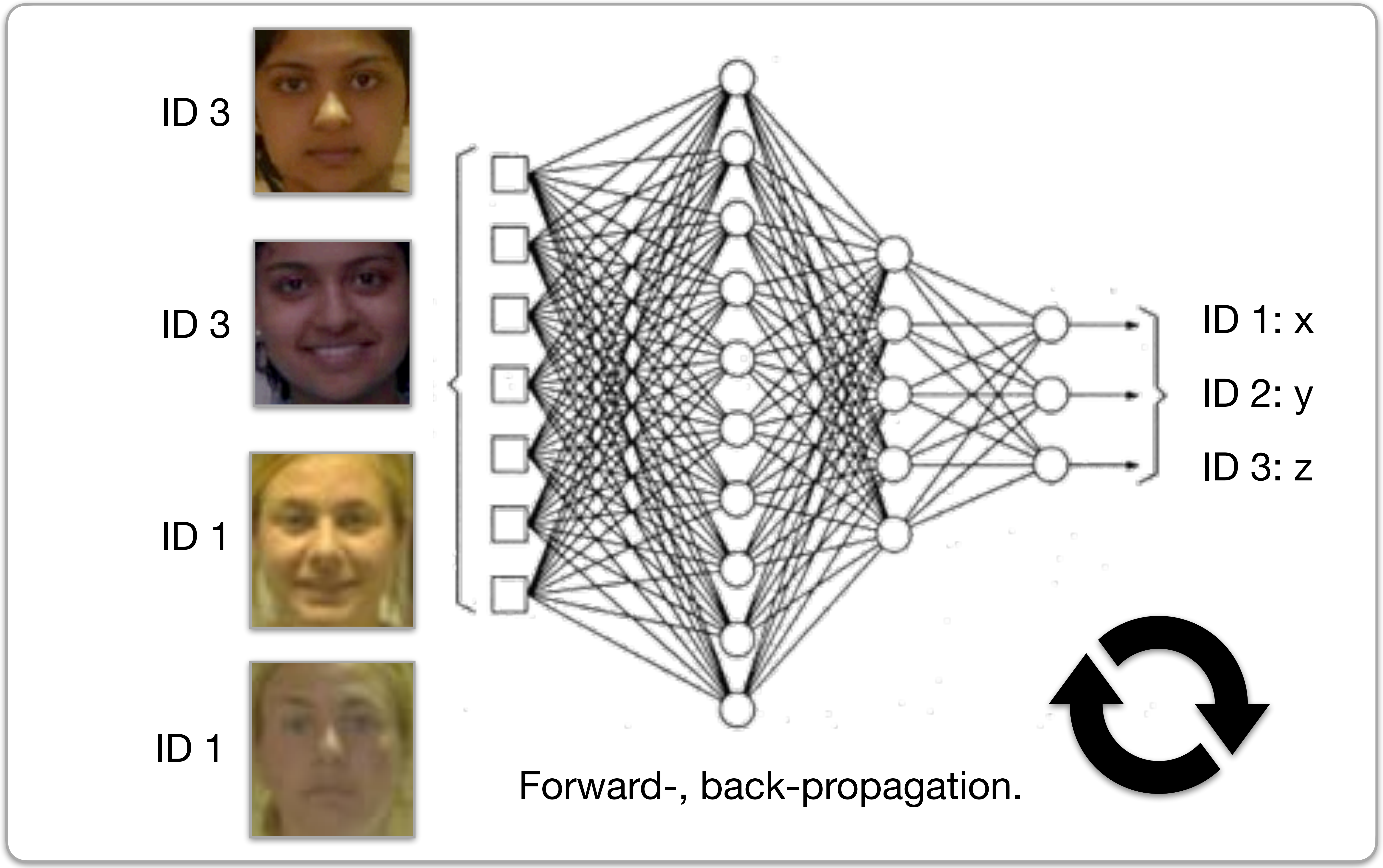


Weights are fixed with back-propagation.

Data-driven Face Recognition

Deep Learning

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

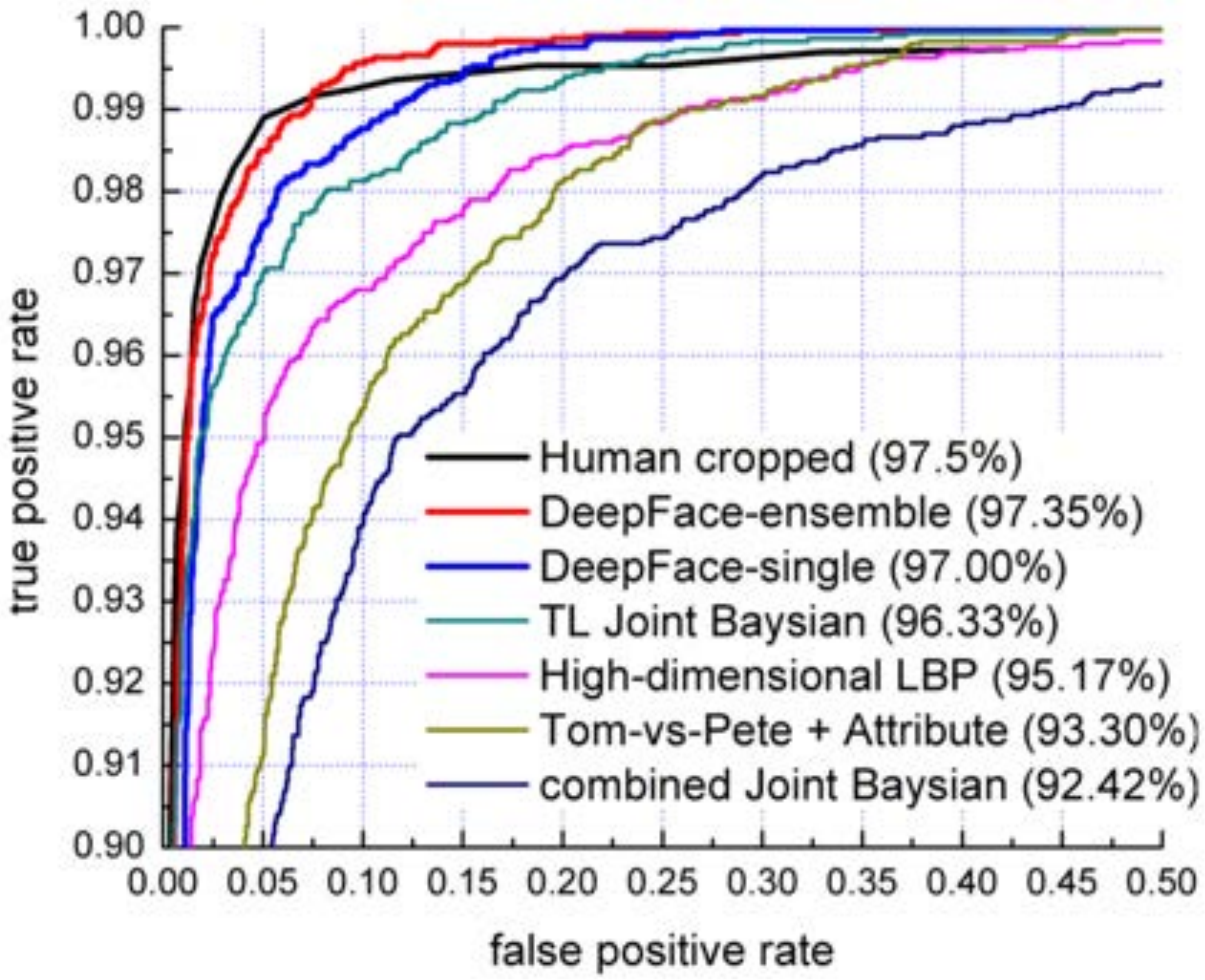


LOYOLA UNIVERSITY CHICAGO

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



LOYOLA UNIVERSITY CHICAGO

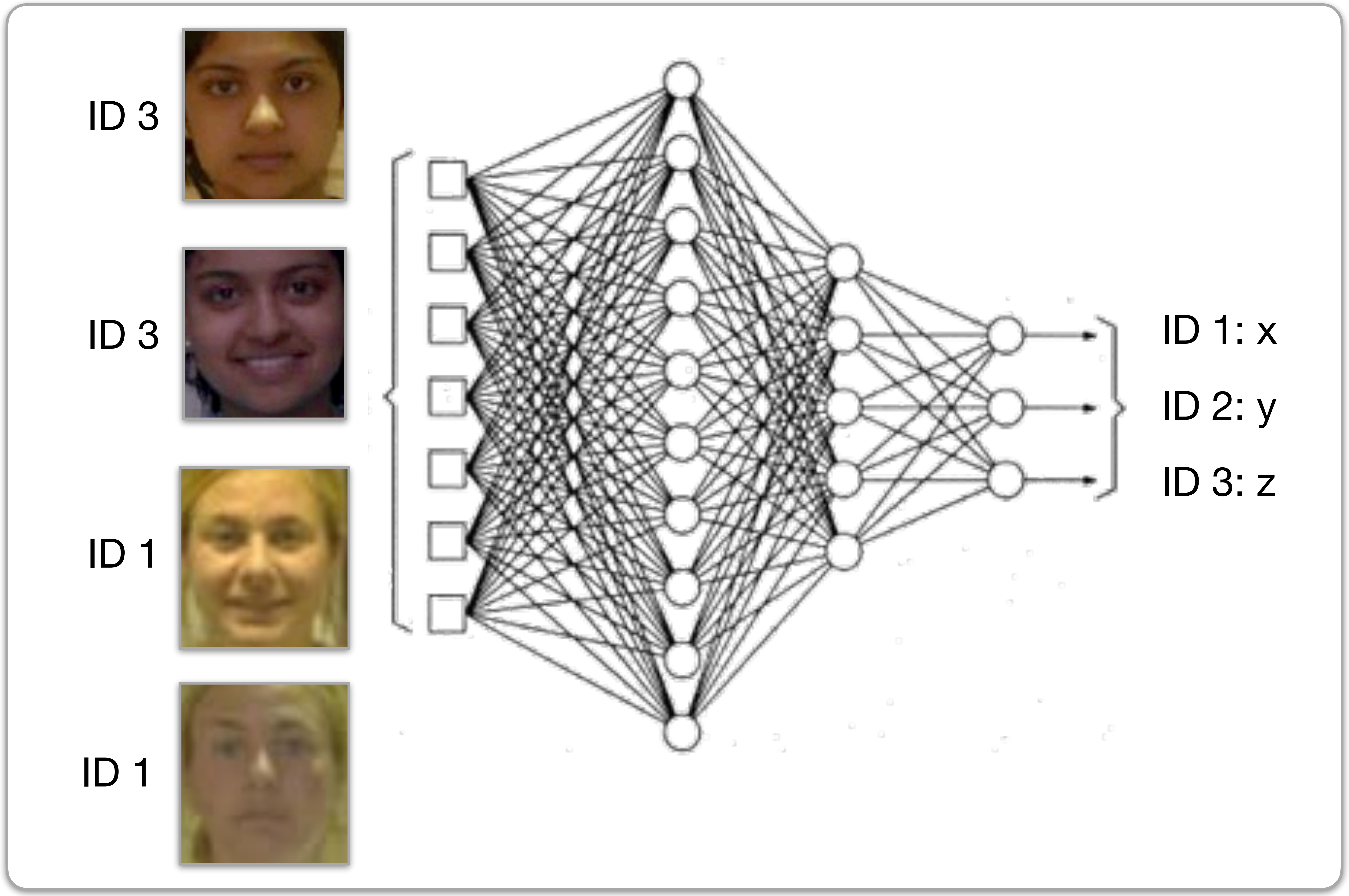
Data-driven Face Recognition

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).

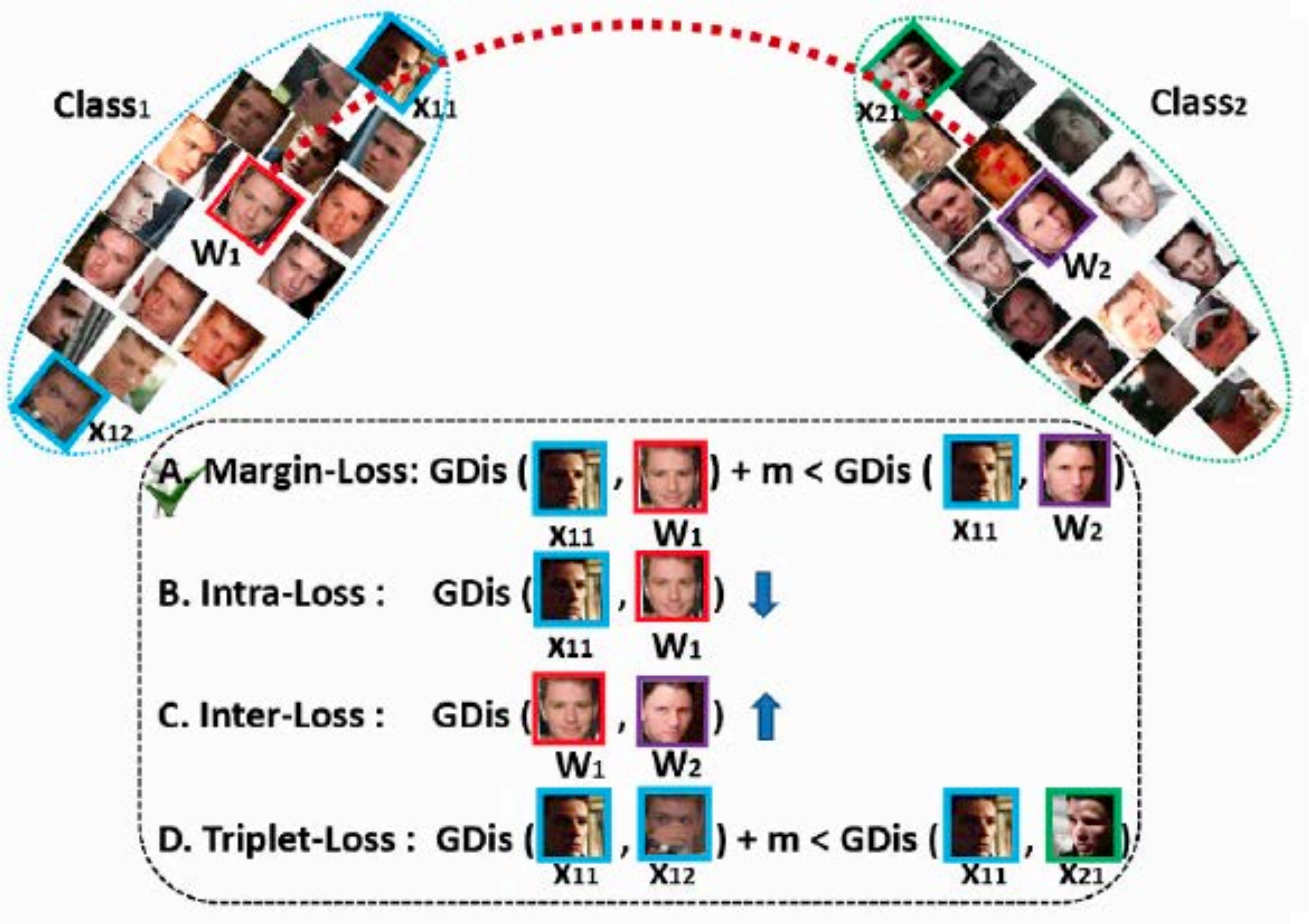


What's Next?

Improving Deep Learning

ArcFace
Additive Angular Margin Loss for Deep Face Recognition
Deng et al., CVPR 2019.
https://bit.ly/3qsQmch

Fill out your
***Today-I-missed* Statement**
Please visit
https://sakai.luc.edu/x/HAZC1P.



LOYOLA UNIVERSITY CHICAGO