

# Document Image Analysis (lab session)

July 2015

Alicia Fornés  
afornes@cvc.uab.es

VISUM Summer School



VISion Understanding and Machine intelligence

# Contents

- Word Spotting using Dynamic Time Warping (DTW)
- Writer Identification using Bag of Words

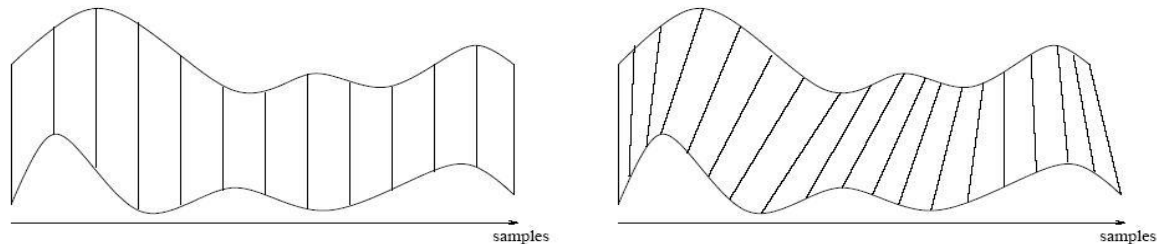
# DYNAMIC TIME WARPING

# Time Sequences Alignment

## Dynamic Time Warping (DTW)

DTW computes the distance between two time series optimizing the alignment

DTW can distort (warp) the time axis, compressing or expanding when necessary



**MATLAB:** `f_compare_vectors_DTW.m`

# DTW: Example with 1-D signals

Y X	1	4	7	9	3	6	9	7	2	1
2	1	5	30	79	80	96	145	170	170	171
5	17	2	6	22	26	27	43	47	56	72
7	53	11	2	6	22	23	27	27	52	88
3	57	12	18	38	6	15	51	43	28	32
2	58	16	37	67	7	22	64	68	28	29
8	107	32	17	18	32	11	12	13	49	77
8	156	48	18	18	43	15	12	13	49	98
4	165	48	27	43	19	19	37	21	17	26
2	166	52	52	76	20	35	68	46	17	18

$$D(i,j) = \min \left\{ \begin{array}{l} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{array} \right\} + d(x_i, y_j)$$

Distance = 18

Normalize by the length  
of the path

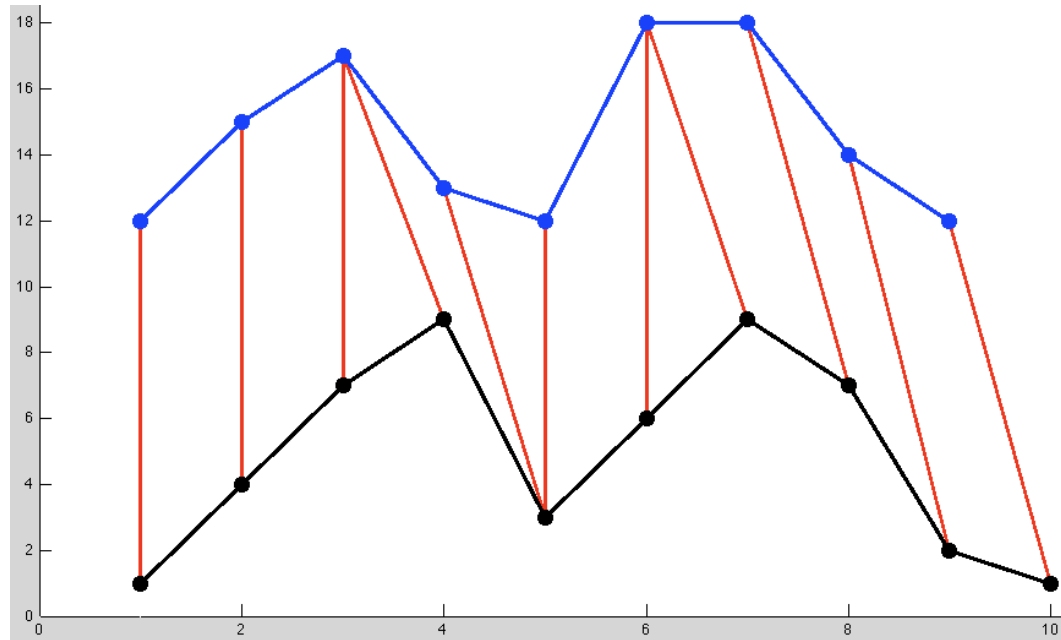
$$\text{dist}(\mathbf{X}, \mathbf{Y}) = D(M, N)/K$$

Path? → Backtracking

Final distance = 18/10

# DTW: Example with 1-D signals

DTW Matching:



# **DYNAMIC TIME WARPING-BASED WORD SPOTTING**

# DTW-based Word Spotting

- Word Segmentation
- Features: Profiles
- Matching: Dynamic Time Warping

*T.M. Rath, R. Manmatha: Word Image Matching Using Dynamic Time Warping. CVPR (2), pp. 521-527, 2003.*

**MATLAB:** `main_wordSpotting.m`



- Feature Extraction

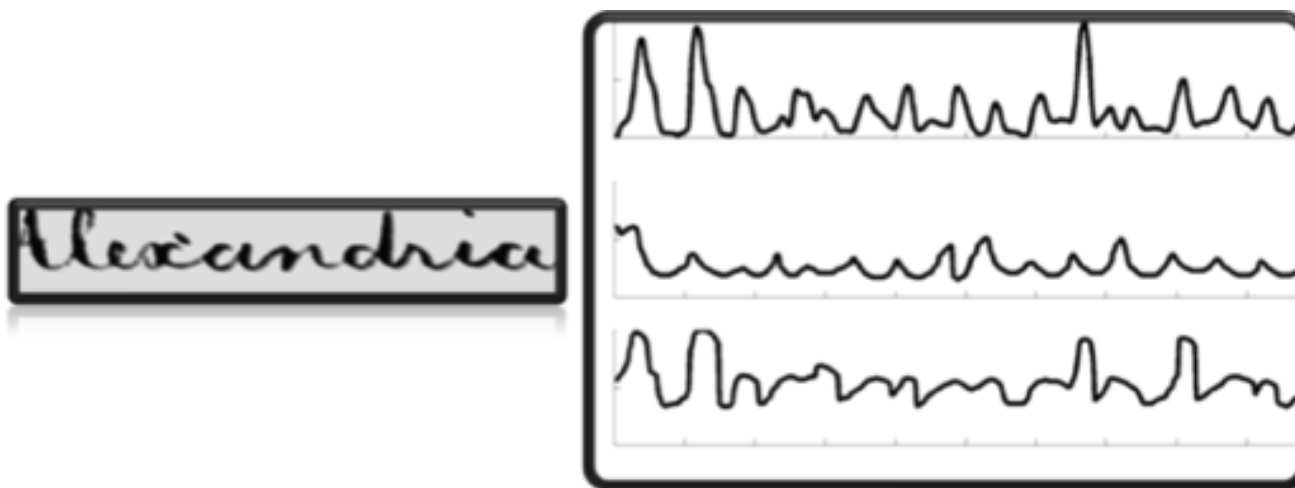
For each column of the word, extract 4 features

f1: upper profile

f2: lower profile

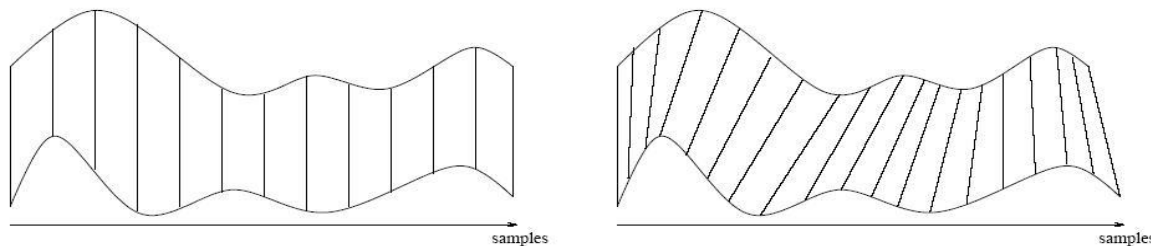
f3: number of foreground pixels

f4: number of transitions (gaps)



- Matching: Dynamic Time Warping (DTW)

The distance at each point is the square of the euclidean distance between 4-dimensional vector



$$D(i, j) = \min \left\{ \begin{array}{l} D(i, j-1) \\ D(i-1, j) \\ D(i-1, j-1) \end{array} \right\} + d(x_i, y_j)$$

$$d(x_i, y_j) = \sum_{k=1}^d (x_{i,k} - y_{j,k})^2$$

# Comparison with a fixed-length descriptor: Zoning

Figure 1 **DTW**

Query	RETRIEVED WORDS			
Orders	Orders	Orders	Orders	Orders
October	October	October	October	October
Orders	Orders	Orders	Orders	Orders
October	October	October	October	October
October	October	October	Orders	Orders
Orders	Orders	Orders	October	October
Setters	October	October	October	October
Orders	Orders	October	October	October
October	October	October	Orders	Orders
Order	Orders	Orders	October	October

Figure 2 **ZONING**

Query	RETRIEVED WORDS			
Orders	Orders	Orders	Order	Orders
October	October	October	October	October
Orders	Orders	October	Orders	Orders
October	October	October	Order	Orders
October	October	Orders	October	October
Orders	Orders	Orders	Orders	Orders
Setters	October	Orders	October	October
Orders	Orders	Orders	Orders	Orders
October	October	October	October	Orders
Order	Orders	October	October	October

# Improvements

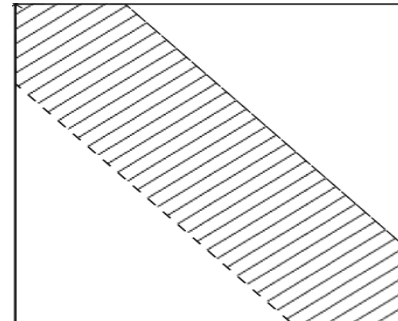
## Disadvantages

Depends on a good word segmentation

Complexity  $O(n^2)$  → Slow method, hardly scalable

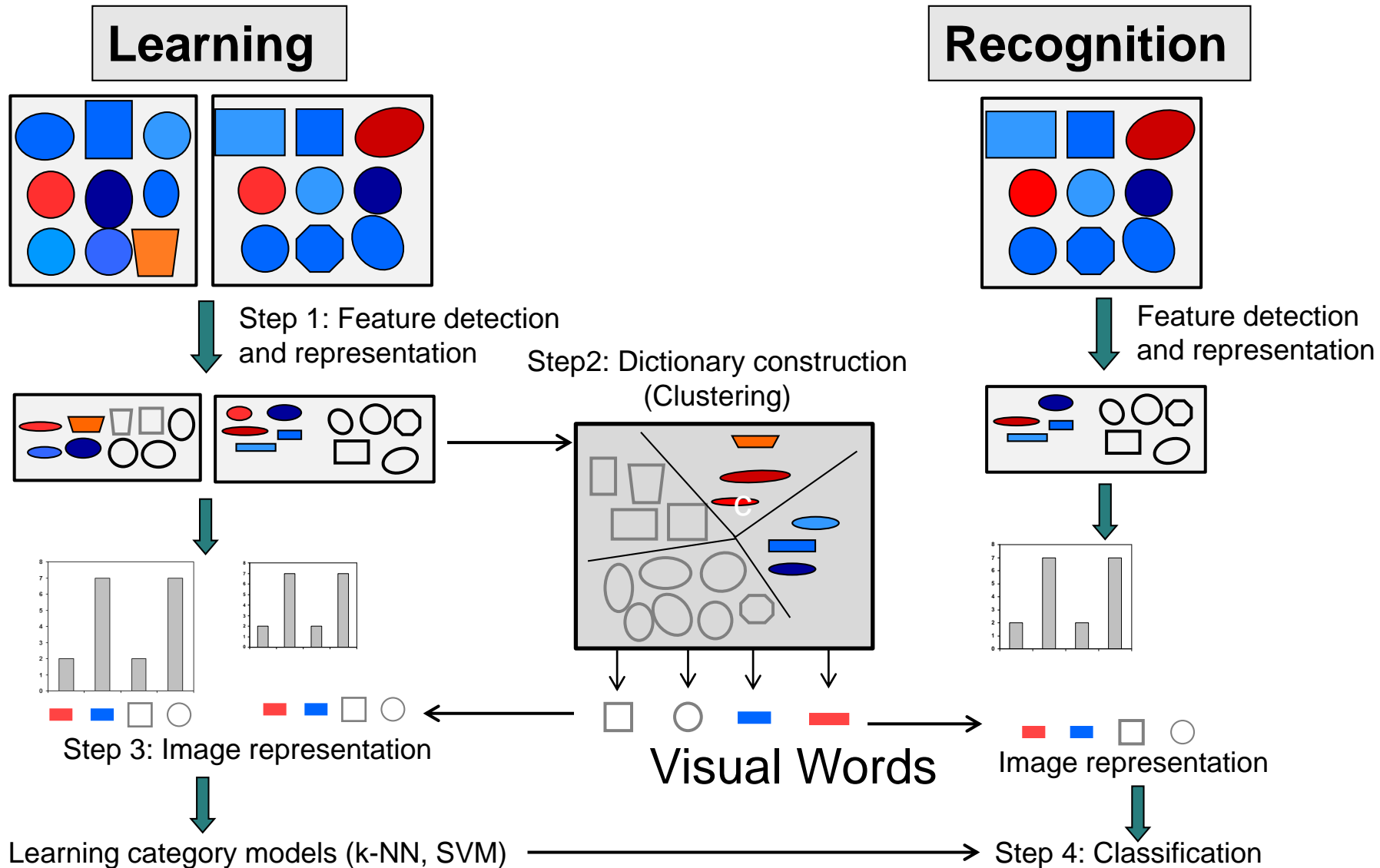
All distances between words have to be computed

Optimizations → Sakoe-Chiba band



# WRITER IDENTIFICATION WITH BAG OF WORDS

# Bag of Words

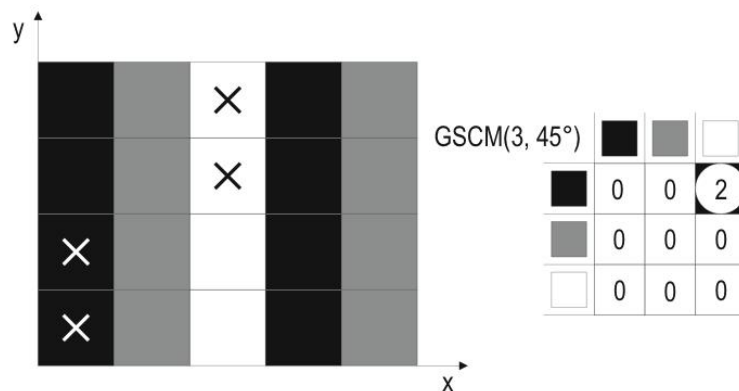


# Textural features

## Grey-Scale (Gray-Level) Co-occurrence Matrices

$GSCM_{d,\alpha}(a,b)$  = number of pairs (Pixel1,Pixel2)  
with a distance  $d$  and angle  $\alpha$ , with color  $a$  and  $b$

In binary images we have only 2 levels (black-white)



**MATLAB:** `main_writer_identification.m`

# Results

## \*\*\* CLASSIFICATION \*\*\*

- Test Image 1, TestPages\wA\_p002.png is **writer: 05**, min distance to page: TrainPages/w-05/p002.png
- Test Image 2, TestPages\wA\_p010.png is writer: 01, min distance to page: TrainPages/w-01/p008.png
- Test Image 3, TestPages\wB\_p003.png is writer: 02, min distance to page: TrainPages/w-02/p005.png
- Test Image 4, TestPages\wB\_p004.png is writer: 02, min distance to page: TrainPages/w-02/p007.png
- Test Image 5, TestPages\wC\_p008.png is writer: 03, min distance to page: TrainPages/w-03/p010.png
- Test Image 6, TestPages\wC\_p009.png is **writer: 02**, min distance to page: TrainPages/w-02/p005.png
- Test Image 7, TestPages\wD\_p001.png is writer: 04, min distance to page: TrainPages/w-04/p004.png
- Test Image 8, TestPages\wD\_p005.png is writer: 04, min distance to page: TrainPages/w-04/p003.png
- Test Image 9, TestPages\wE\_p006.png is writer: 05, min distance to page: TrainPages/w-05/p008.png
- Test Image 10, TestPages\wE\_p007.png is writer: 05, min distance to page: TrainPages/w-05/p009.png

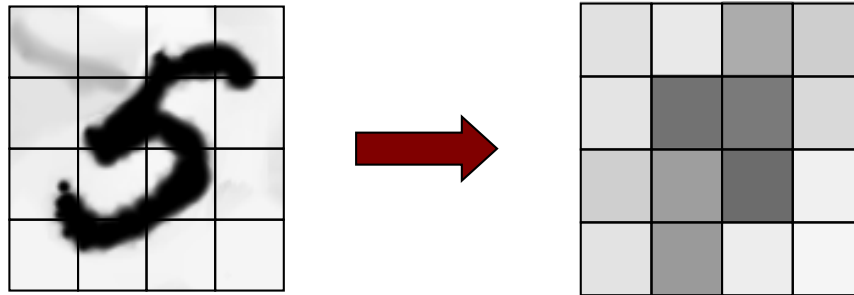
Ground-truth:

wA= writer 1, wB= writer 2, wC= writer 3, wD= writer 4, wE= writer 5



# Shape Descriptor: Zoning

- The image is divided in  $n \times m$  cells.
- For each cell the mean of gray levels is computed and all these values are joined in a feature vector of length  $n \times m$ .



**MATLAB:** `f_compute_glcmm` → `compute_glcmm = 0;` % if 0, compute Zoning

# Comparison with a shape-based descriptor: Zoning

## \*\*\* CLASSIFICATION \*\*\*

- Test Image 1, TestPages\wA\_p002.png is **writer: 05**, min distance to page: TrainPages/w-05/p002.png
- Test Image 2, TestPages\wA\_p010.png is **writer: 04**, min distance to page: TrainPages/w-04/p010.png
- Test Image 3, TestPages\wB\_p003.png is writer: 02, min distance to page: TrainPages/w-02/p009.png
- Test Image 4, TestPages\wB\_p004.png is writer: 02, min distance to page: TrainPages/w-02/p001.png
- Test Image 5, TestPages\wC\_p008.png is writer: 03, min distance to page: TrainPages/w-03/p005.png
- Test Image 6, TestPages\wC\_p009.png is writer: 03, min distance to page: TrainPages/w-03/p003.png
- Test Image 7, TestPages\wD\_p001.png is **writer: 01**, min distance to page: TrainPages/w-01/p001.png
- Test Image 8, TestPages\wD\_p005.png is writer: 04, min distance to page: TrainPages/w-04/p008.png
- Test Image 9, TestPages\wE\_p006.png is **writer: 03**, min distance to page: TrainPages/w-03/p004.png
- Test Image 10, TestPages\wE\_p007.png is writer: 05, min distance to page: TrainPages/w-05/p009.png

Ground-truth:

wA= writer 1, wB= writer 2, wC= writer 3, wD= writer 4, wE= writer 5

# Tired of Toy-examples?

Try with more challenging datasets:

- Word Spotting:

<http://transcriptorium.eu/~icdar15kws/>

- Writer Identification at the ICDAR/GREC 2011 Competition:

<http://www.cvc.uab.es/cvcmuscima/>

Still want more? There are many competitions at ICDAR conferences:

- <http://2015.icdar.org/program/competitions/>
- <http://www.icdar2013.org/program/competitions>
- etc.