Handwritten Word Spotting

July 2015

Alicia Fornés
afornes@cvc.uab.es

VISUM Summer School

VISion Understanding and Machine intelligence

Centre de Visió per Computador
UAB Universitat Autònoma de Barcelona
Generalitat de Catalunya
Parc de Recerca UAB

Contents

- Introduction
- Segmentation-based / Segmentation-free methods
- Learning-free / Learning-based methods
- Query-by-Example / Query-by-String
- Context aware Word Spotting
- Conclusions

INTRODUCTION

Word Spotting

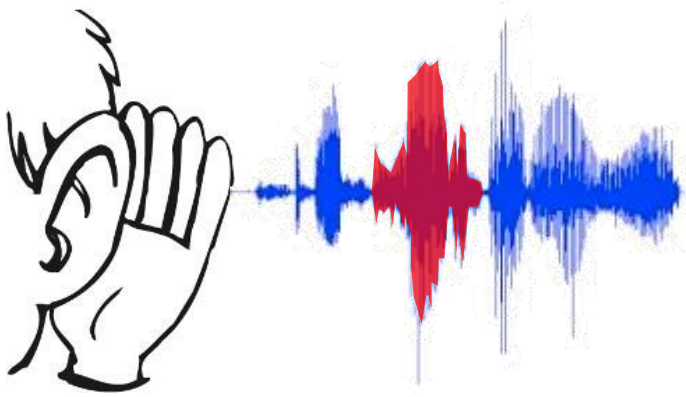
Spotting is the task of locating a particular element without explicitly recognizing the content

Word spotting is the task of locating a particular keyword without explicitly transcribing the content

Spoken Word Spotting

Word Spotting was introduced within the speech recognition field in the *70s*

- Continuous one-dimensional signal
- Unconstrained (words are not segmented, no grammar is considered in the sentence,…)
- Dynamic Time Warping, Hidden Markov Models, …



H. Sakoe and S. Chiba. “A Dynamic Programming Approach to Continuous Speech Recognition”. International Congress on Acoustics. 1971.

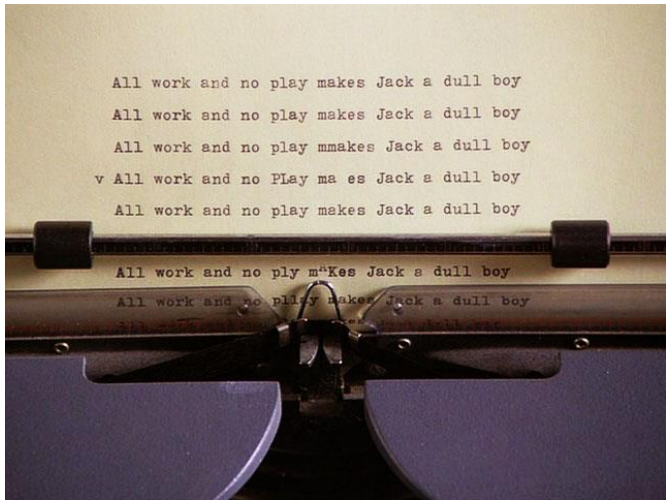
R.W. Christiansen and C.K. Rushforth. “Word Spotting in Continuous Speech using Linear Predictive Coding”. International Conference on Acoustics, Speech and Signal Processing. 1976

Typewritten Word Spotting

Then applied to typewritten documents in the early *90s*

Two-dimensional signal that can be “easily” segmented into words

Character Signatures, Hidden Markov Models,…

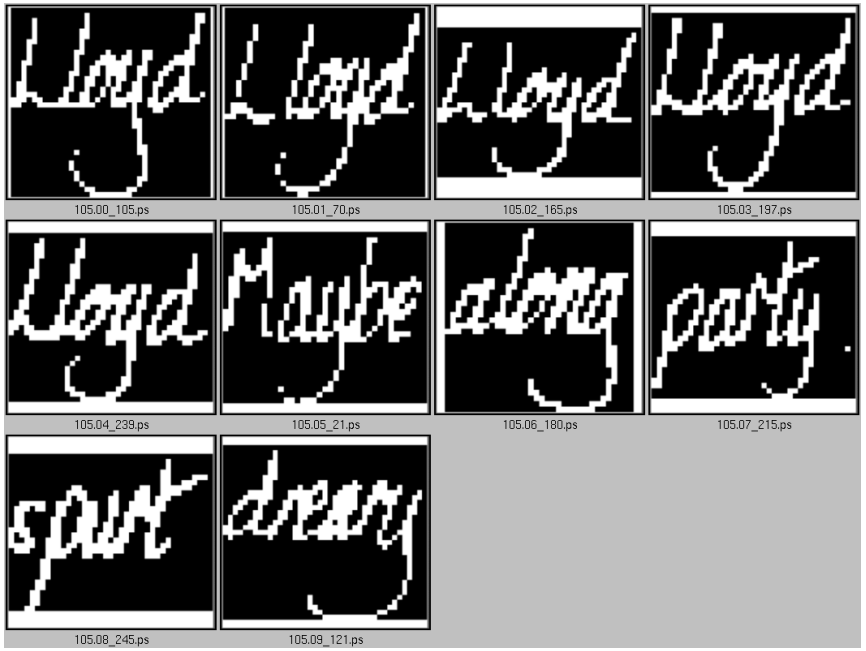


F.R. Chen et al. “Word Spotting in scanned images using hidden Markov models”. International Conference on Acoustics, Speech and Signal Processing. 1993

Handwritten Word Spotting

The first work on Handwritten Word Spotting was published in 1996…

Template matching between query and words in the DB allowing translation and affine transforms



R. Manmatha et al. “Word spotting: a new approach to indexing handwritting”. In Proc. of CVPR, 1996.

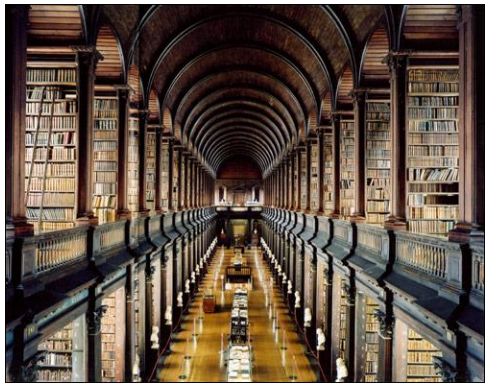
Introduction

Digitization in Archives

Preservation: Face the paper deterioration problem

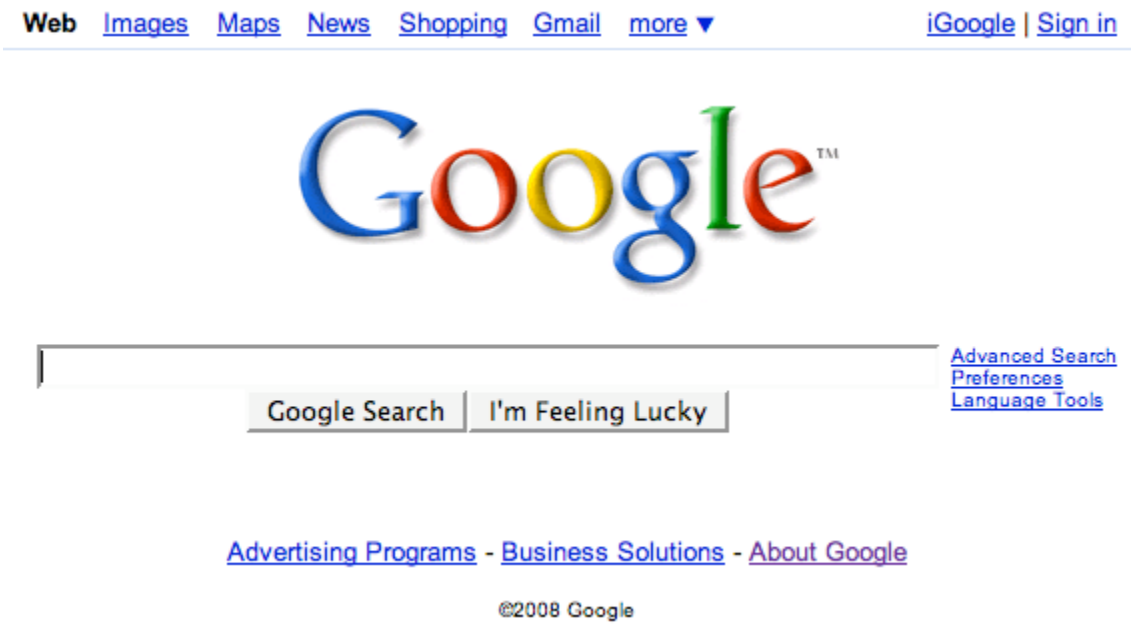
Storage: Avoid having kilometers of shelves

Accessibility: Allow users around the world access the cultural heritage



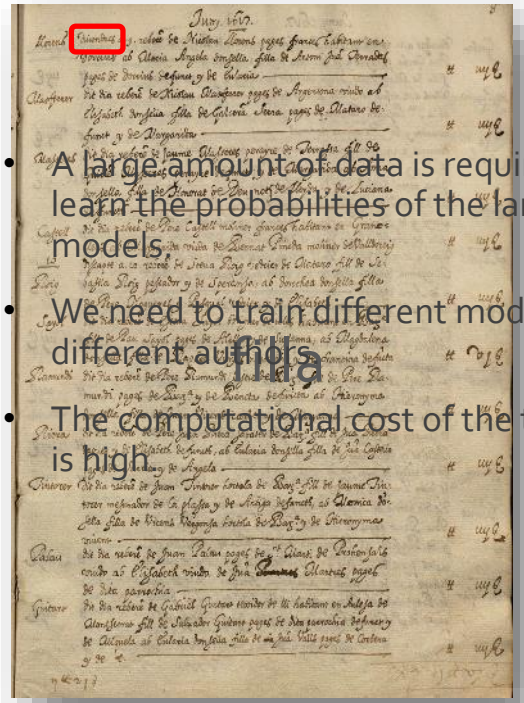
Introduction

Users would like to search information in document images like this!



Word Spotting Systems

Strategy 1: Full transcription + Search the words in textual transcriptions

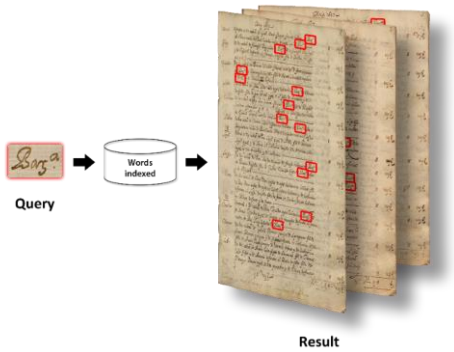


- A large amount of data is required to learn the probabilities of the language models.
- We need to train different models for different authors.
- The computational cost of the training is high.

filla

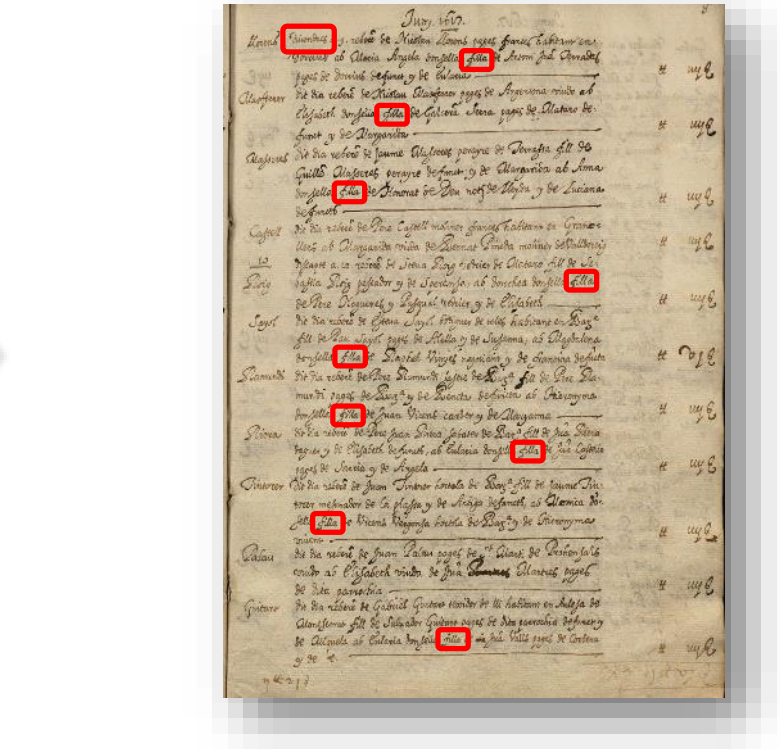
divendres a 9 rebere de Nicolau llorens pages frances habitant en dorrius ab Maria Angela donsella **filla** de Antoni jua Terrades pages de dorrius defunct y de Eularia
dit dia rebere de Nicolau Masferrer pages de Argentona viudo ab Elisabeth donsella **filla** de Galcera Serra pages de Mataro de funct y de Margarida
dit dia rebere de jaume Masseres perayre de Terrassa fill de guille Masseres perayre defunct y de Margarida ab Anna donsella **filla** de Honorat de Deu nog de Lleyda y de Luciana defuncts
dit dia rebere de Pere castell moliner frances habitant en Grano llers ab Margarida viuda de Bernat pineda moliner de Valldoreig
dissapte a 10 rebere de Steva Roig vedrier de Mataro fill de Se bastia Roig pescador y de Speransa ab dorochea donsella **filla** de Pere Nogueres y Pasqual vedrier y de Elisabeth
dit dia rebere de Esteva Sayol botiguer de teles habitant en Barca fill de Pau Sayol pages de Alella y de Susanna ab Magdalena donsella **filla** de Raphael Vinyes negociant y de francina defucta
dit dia rebere de Pere Ramundi sastre de Barca fill de Pere ra mundi pages de Barca y de Beneta defuncta ab Hieronyma donsella **filla** de Juan Vicens carder y de Maryanna
dit dia rebere de Pere juan Riera sabater de Barca fill de jua Riera daguer y de Elisabeth defuncts ab Eularia donsella **filla** de jua Castello pages de Sarria y de Angela
dit dia rebere de juan Tintorer hortola de Barca fill de jaume Tin torer mesurador de la plassa y de Antiga defuncts ab Monica do sella **filla** de Vicens Vergonsa hortola de Barca y de Hieronyma vivent
dit dia rebere de juan Palau pages de St Marti de Prohensals viudo ab Elisabeth viuda de jua xxxxx martres pages de dita parrochia
dit dia rebere de Gabriel Guitart texidor de lli habitant en Aulesa de Montserrat fill de Salvador Guitart pages de dita parrochia defunct y de Miquela ab Eularia donsella **filla** de xxxxx jua Valls pages de Corbera y de t

Word Spotting



Strategy 2: handwritten word spotting

- The recognition of words is formulated as a pattern recognition problem
- The transcription is not necessary because the search is done by similarity between the shapes.



Difficulties

Difficulties:

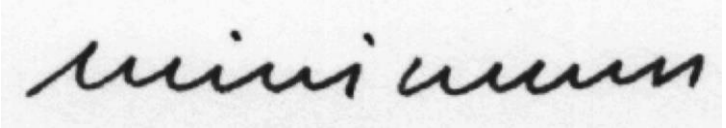
- Different Handwriting styles

- Different writing instruments

- Large vocabulary

- Segmentation problem

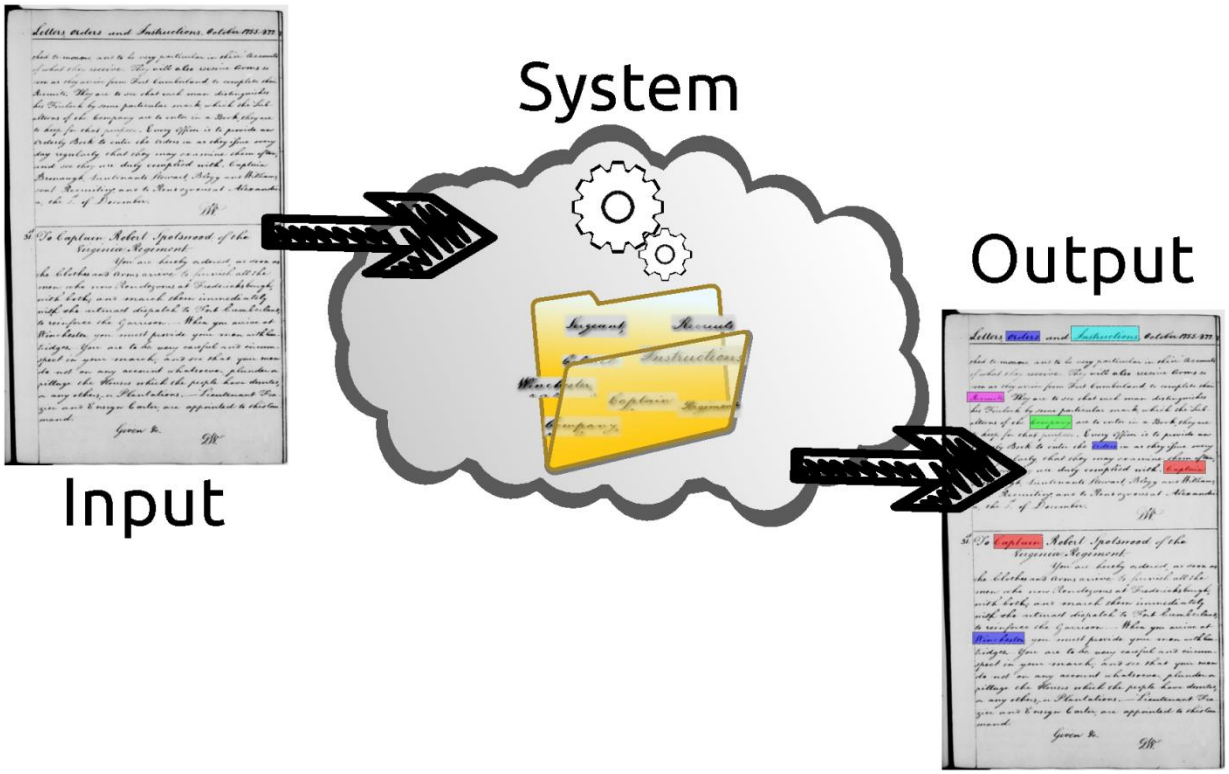
  chicken-egg



Handwritten Word Spotting

How to organize different families?

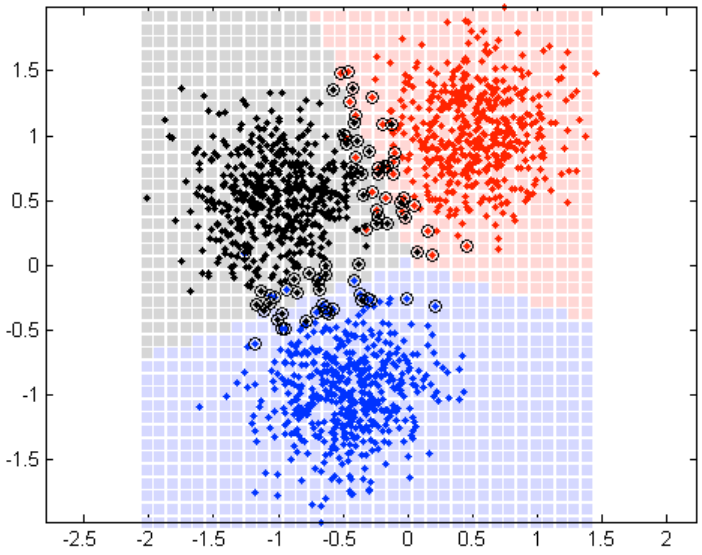
Retrieval application *vs.* **Filtering** application



Handwritten Word Spotting

How to organize different families?

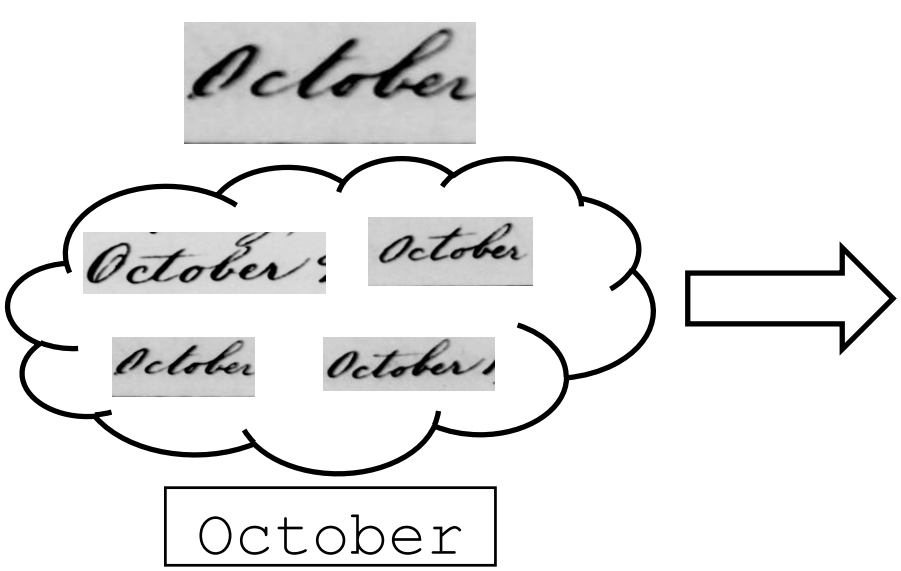
- **Retrieval** application *vs.* **Filtering** application
- **Learning-free** *vs.* **Learning-based** methods

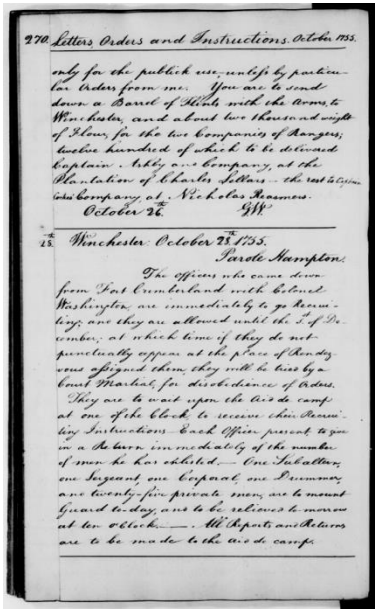


Handwritten Word Spotting

How to organize different families?

- **Retrieval** application *vs.* **Filtering** application
- **Learning-free** *vs.* **Learning-based** methods
- **Query-by-example** *vs.* **Query-by-class** *vs.* **Query-by-string**

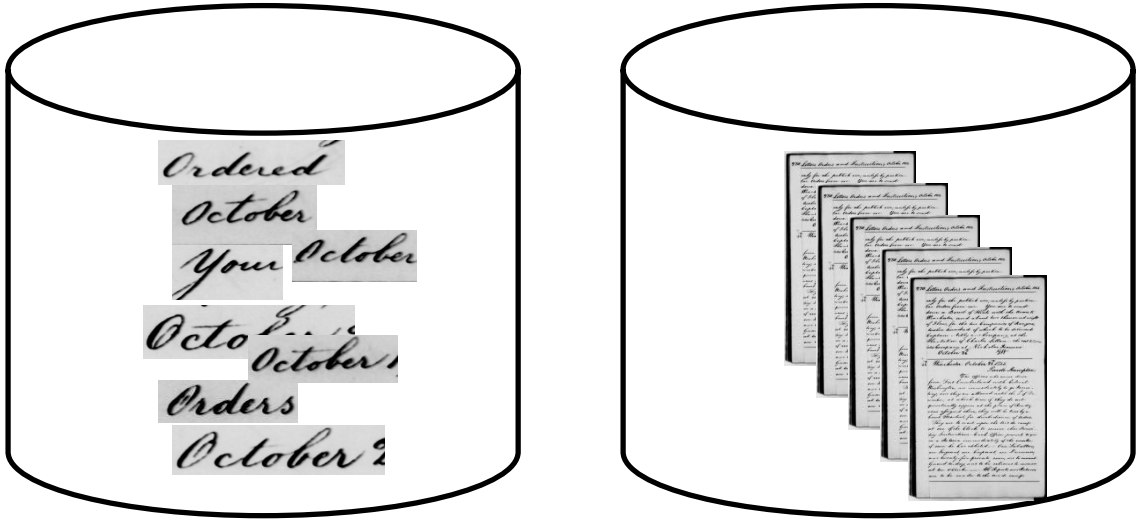




Handwritten Word Spotting

How to organize different families?

- **Retrieval** application *vs.* **Filtering** application
- **Learning-free** *vs.* **Learning-based** methods
- **Query-by-example** *vs.* **Query-by-class** *vs.* **Query-by-string**
- **Segmentation-based** vs. **Segmentation-free** methods



Families of HWS methods

- **Segmentation-free** *vs.* **Segmentation-based** methods
- **Learning-free** *vs.* **Learning-based** methods
- **Query-by-example** *vs.* **Query-by-string**

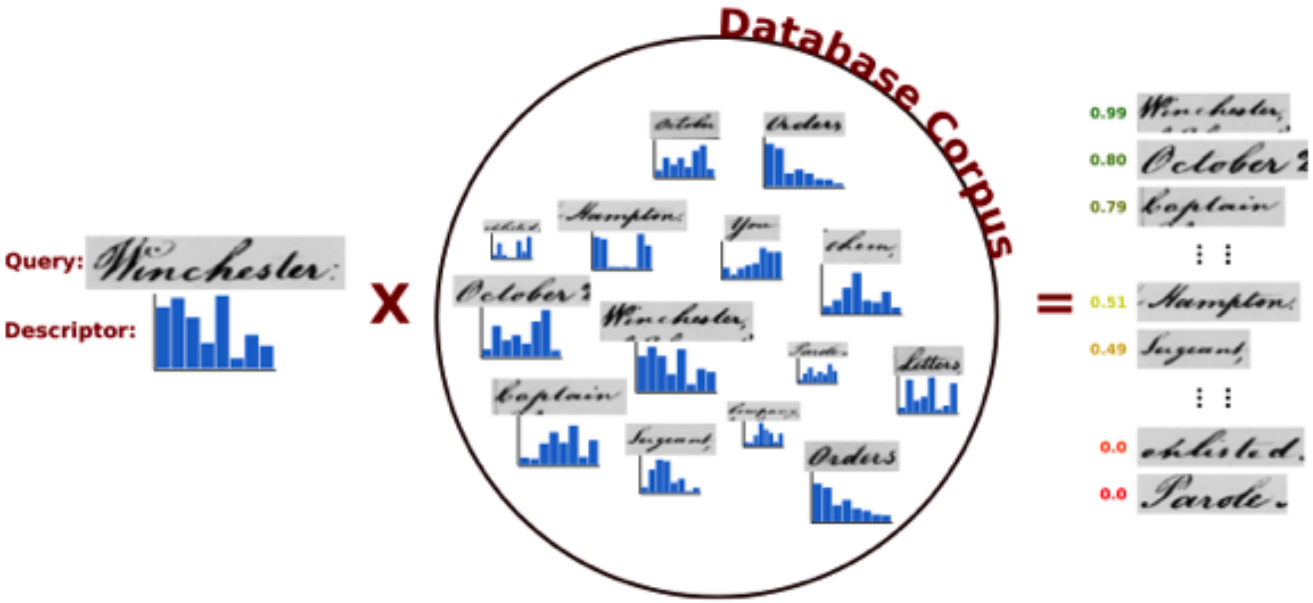
WORD SPOTTING LEARNING-FREE & SEGMENTATION-BASED

Classical Architecture

Layout analysis for word segmentation

Feature extraction: each word has a feature vector

Some distance is used for matching the query with the corpus



Classic Approach: Rath and Manmatha 2003

Break-through reference paper

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.

Classic Approach: Rath and Manmatha 2003

- Feature Extraction

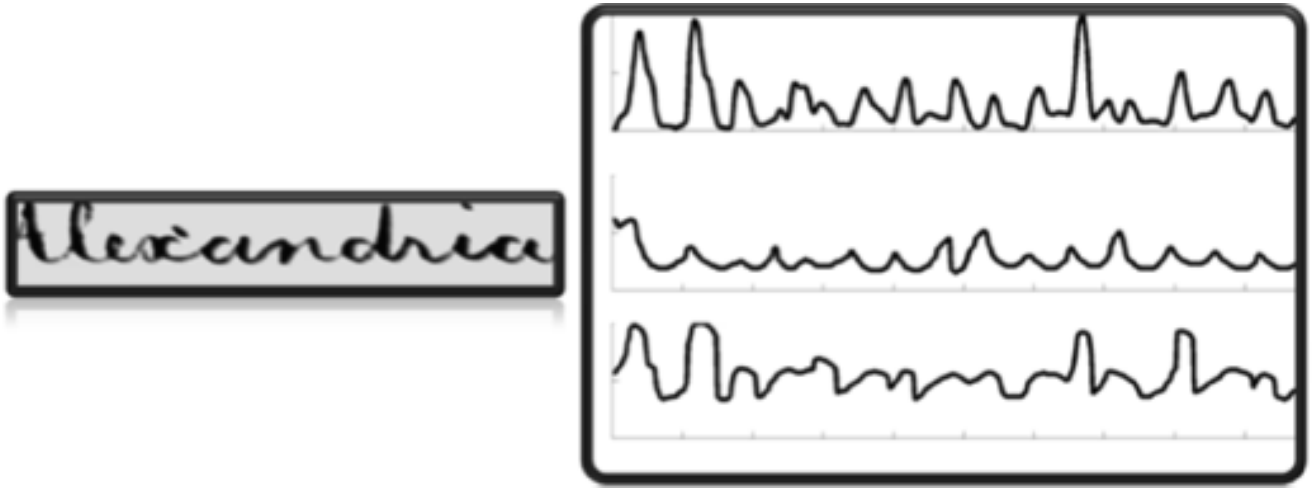
  For each column of the word, extract 4 features

  f1: upper profile

  f2: lower profile

  f3: number of foreground pixels

  f4: number of transitions (gaps)

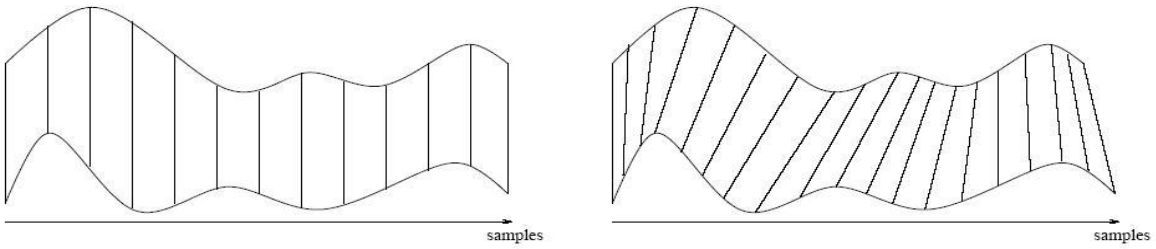


Classic Approach: Rath and Manmatha 2003

- Matching: Dynamic Time Warping (DTW)

  DTW computes the distance between two time series optimizing the alignment

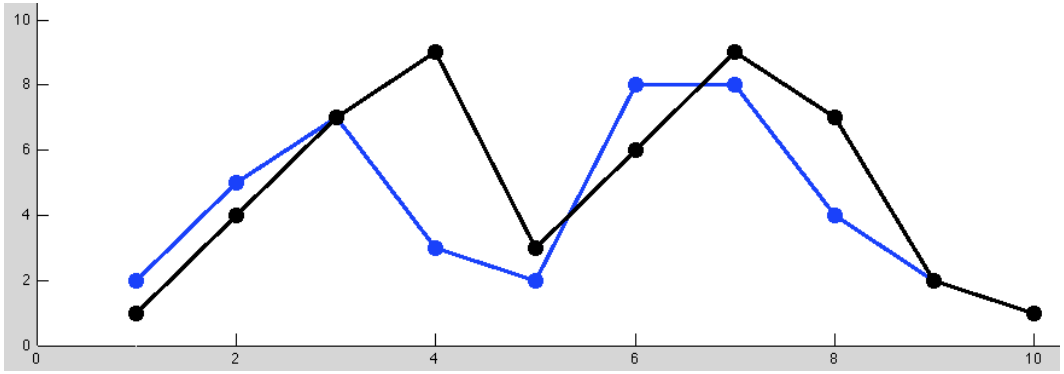
  DTW can distort (warp) the time axis, compressing or expanding when necessary



DTW: Example with 1-D signals

Two 1-Dimensional signals

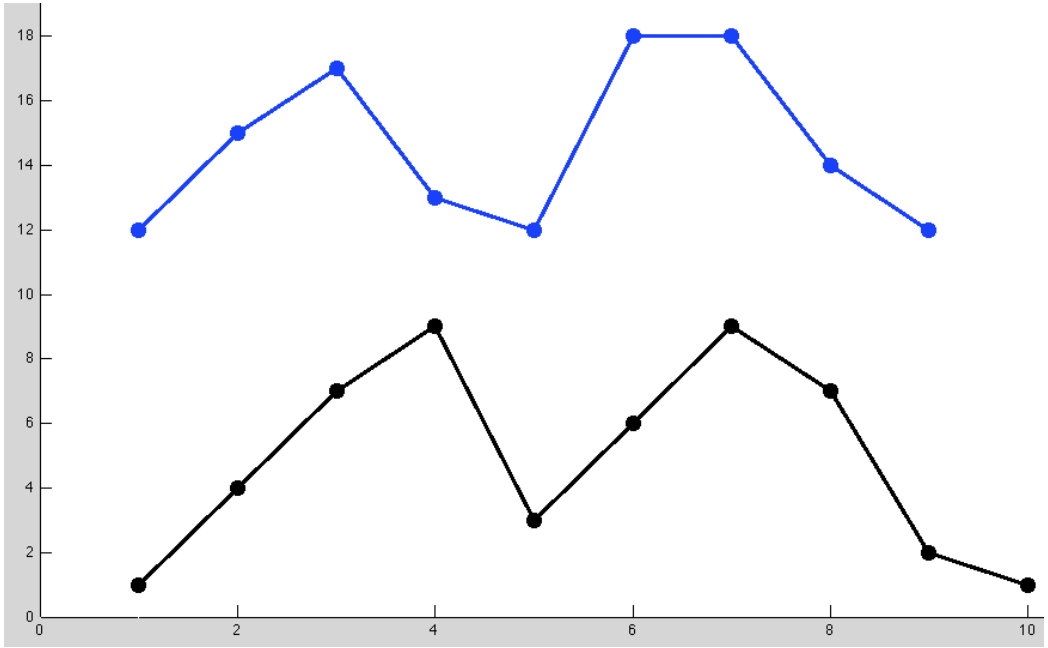
X =[2,5,7,3,2,8,8,4,2] ←→ Y = [1,4,7,9,3,6,9,7,2,1]



DTW: Example with 1-D signals

Two 1-Dimensional signals

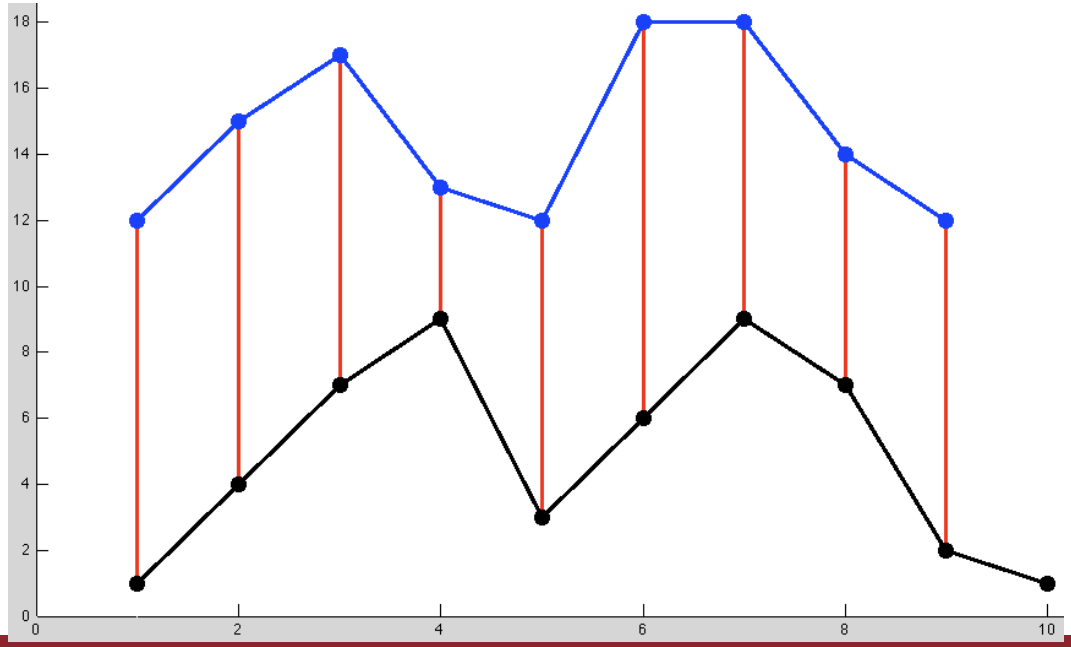
X =[2,5,7,3,2,8,8,4,2] ←→ Y = [1,4,7,9,3,6,9,7,2,1]



DTW: Example with 1-D signals

Matching:

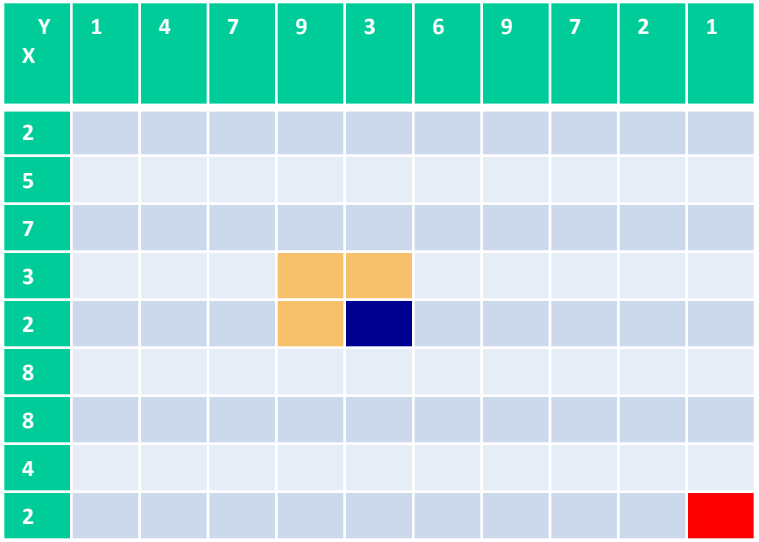
Euclidean distance → Undesired!



DTW: Example with 1-D signals

DTW is computed using Dynamic Programming

- The last cell contains the total cost (distance)
- The cost of a cell depends on 3 cells



$$D(i,j) = \min \begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_i)$$

DTW: Example with 1-D signals

X \ Y	1	4	7	9	3	6	9	7	2	1
2	1	5	30	79	80	96	145	170	170	171
5	17									
7	53									
3	57									
2	58									
8	107									
8	156									
4	165									
2	166									

$$D(i,j) = \min \begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_i)$$

First row:

$(1\text{-}2)^2 = 1$

$(4\text{-}2)^2 + 1 = 5$

$(7\text{-}2)^2 + 5 = 30$

First column:

$(5\text{-}1)^2 + 1 = 17$

$(7\text{-}1)^2 + 17 = 53$

DTW: Example with 1-D signals

X \ Y	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									
3	57									
2	58									
8	107									
8	156									
4	165									
2	166									

$$D(i,j) = \min \begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_i)$$

Second row:

$(4-5)^2 + 1 = 2$

$(7-5)^2 + 2 = 6$

$(9-5)^2 + 6 = 22$

…

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

$$D(i,j) = \min \begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_i)$$

Distance = 18

Normalize by the length of the path

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

DTW: Example with 1-D signals

X \ Y	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

$$D(i,j) = \min \begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_i)$$

Distance = 18

Normalize by the length of the path

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

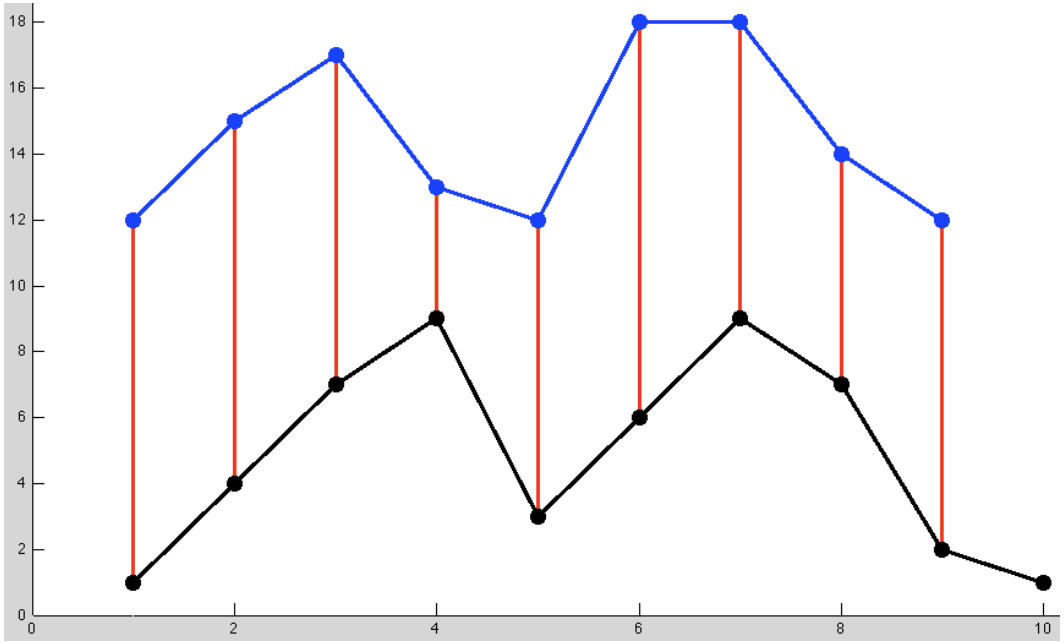
Path? → Backtracking

Final distance = 18/10

DTW: Example with 1-D signals

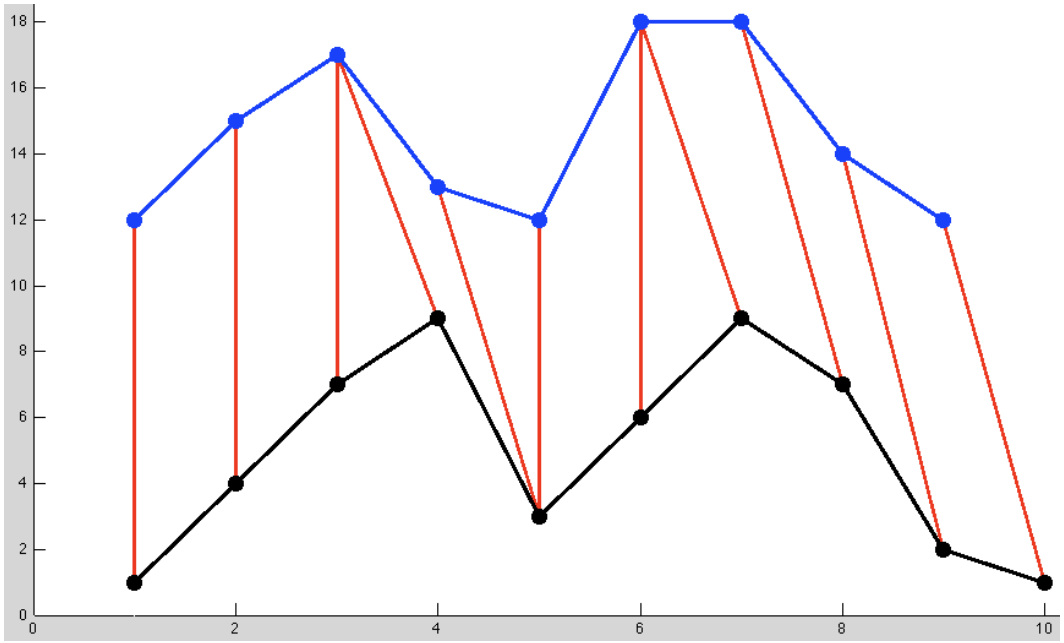
Matching:

Euclidean distance → Undesired!



DTW: Example with 1-D signals

DTW Matching:



Classic Approach: Rath and Manmatha 2003

- Matching: Dynamic Time Warping (DTW)

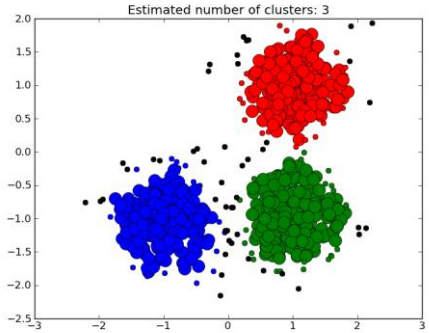
  The distance at each point is the square of the euclidean distance between 4-dimensional vector

$$D(i,j) = \min \begin{Bmatrix} D(i,j-1) \\ D(i-1,j) \\ D(i-1,j-1) \end{Bmatrix} + d(x_i, y_i)$$

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

Improvement: Rath and Manmatha 2007

- Same method + Clustering (groups)
  - DFT features from profiles → K-means clustering
  - Matrix with pairwise distances → k-means
    - Average linkage: average distance between all pair items

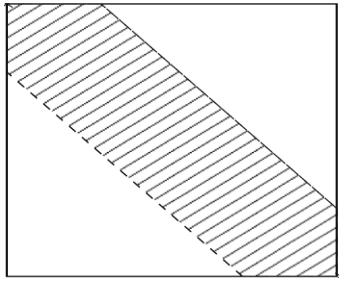


- Useful for speeding up the transcription of documents
  - The user transcribes only one word in each cluster
  - All words in the cluster are automatically transcribed

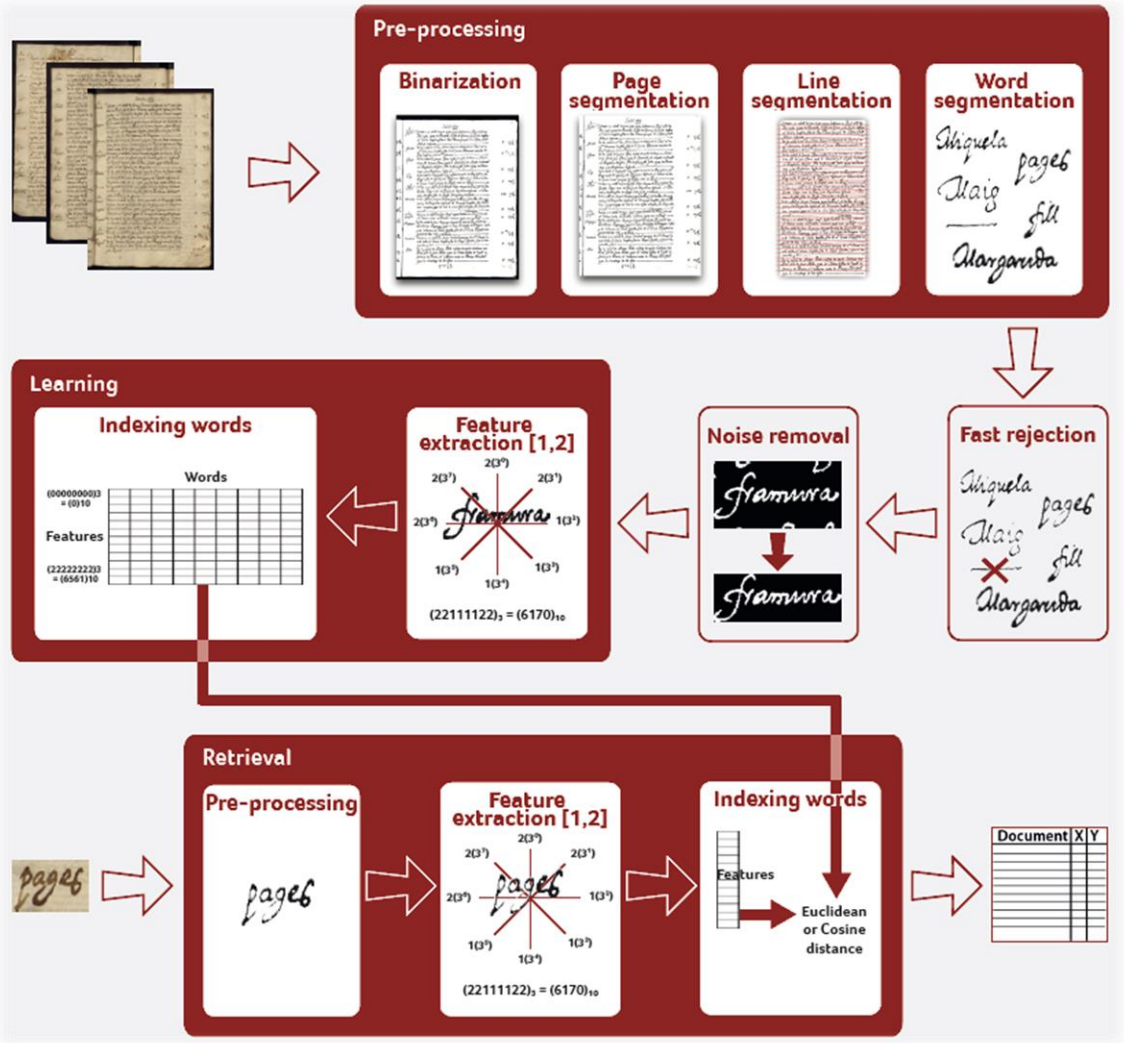
T.M. Rath, R. Manmatha: Word spotting for historical documents. IJDAR, 2007.

Improvement: Rath and Manmatha 2007

- Advantages
  - Features and matching robust to deformations
    - They can handle the variability in the handwriting style
- 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

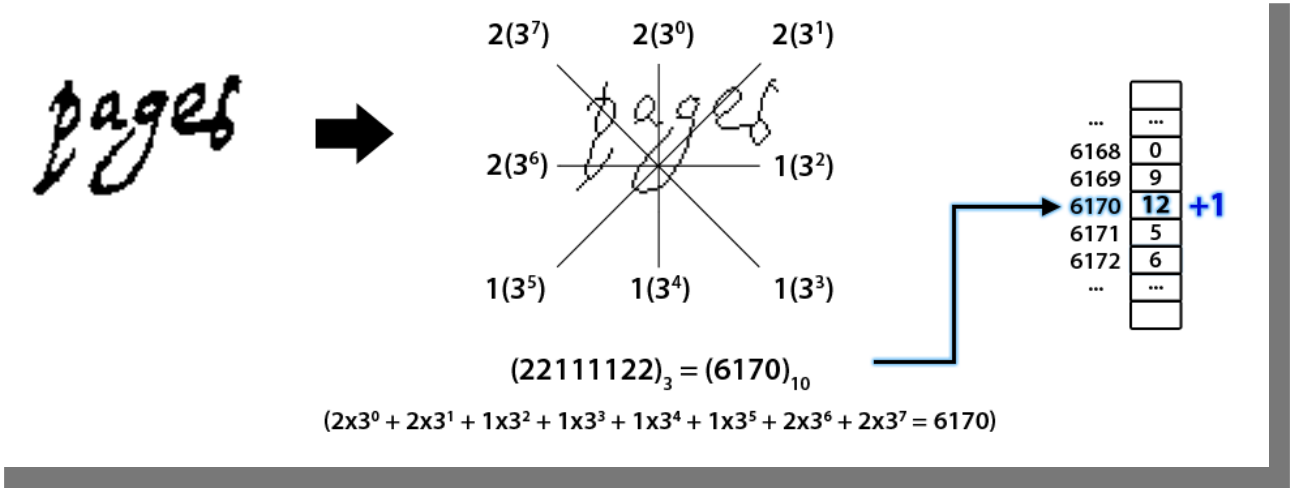


Word Spotting using a pseudo-structural descriptor



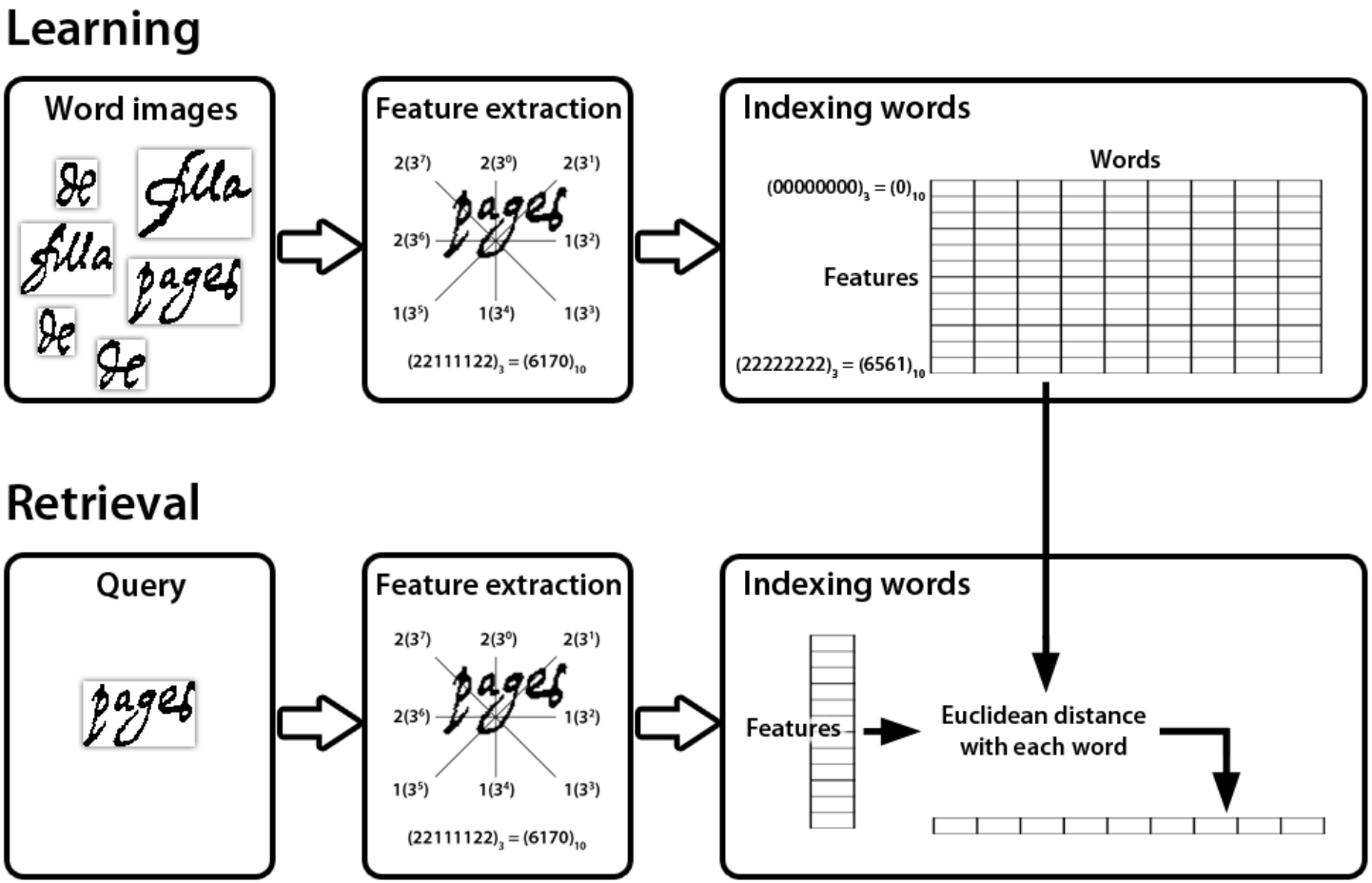
Word Spotting using a pseudo-structural descriptor

Characteristic Loci Features: A characteristic Loci feature is composed by the number of the intersections in four directions (up, down, right and left).



	Values		
direction	**0**	**1**	**2**
Vertical	{0}	[1, 2]	[3, $+\infty$]
Horizontal	{0}	[1, 4]	[5, $+\infty$]
Diagonal	{0}	[1, 3]	[4, $+\infty$]

Word Spotting using a pseudo-structural descriptor

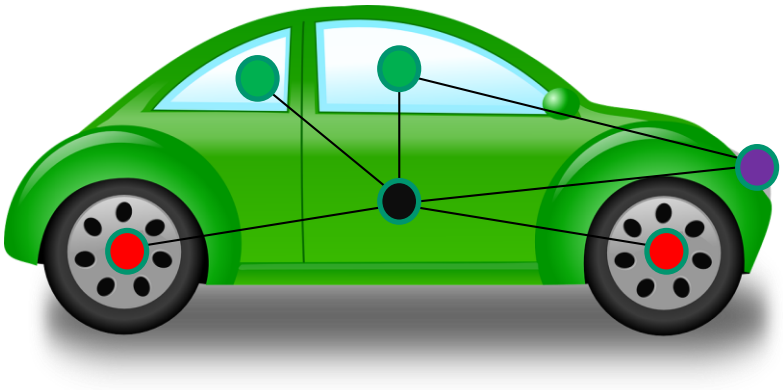


D. Fernandez, J. LLados, and A. Fornes. Handwritten Word Spotting in Old Manuscript Images using a Pseudo-Structural Descriptor Organized in a Hash Structure. Iberian Conference on Pattern Recognition and Image Analysis (IBPRIA), June 2011.

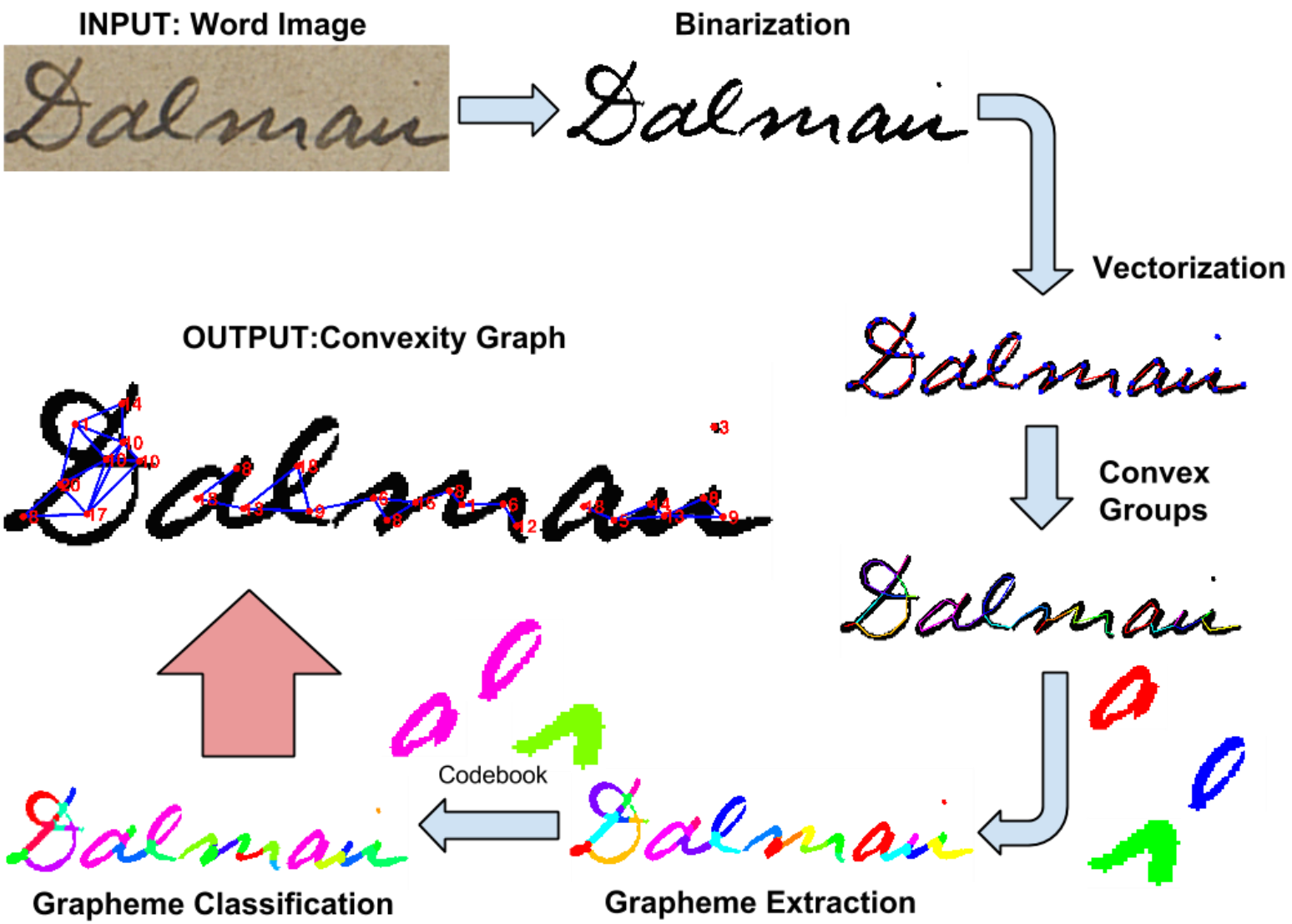
Structural Representations

Graph-based representation.

A **graph** is a set of objects called vertices that can be connected with an edge representing their relation.



Word Spotting using a graph representation



Robust Graph-based Representation

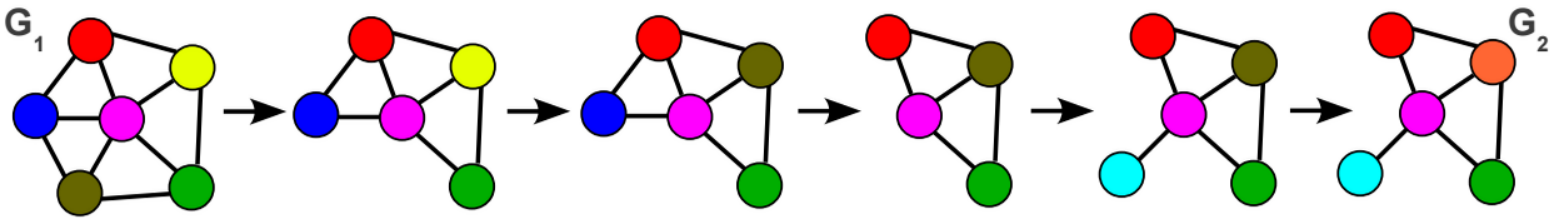
- **Robust graph representation**, tolerant to the inherent deformation of handwritten strokes, and variations among writers.
- Representation based on **convex groups** of the skeleton image (nodes), called **graphemes**, and adjacency relationships (edges).
- **Grapheme** or **Glyph** is the smallest unit used in describing the writing system of a language.

P.Riba, J.Lladós, A.Fornés. Handwritten Word Spotting by Inexact Matching of Grapheme Graphs. International Conference on Document Analysis and Recognition (ICDAR), 2015.

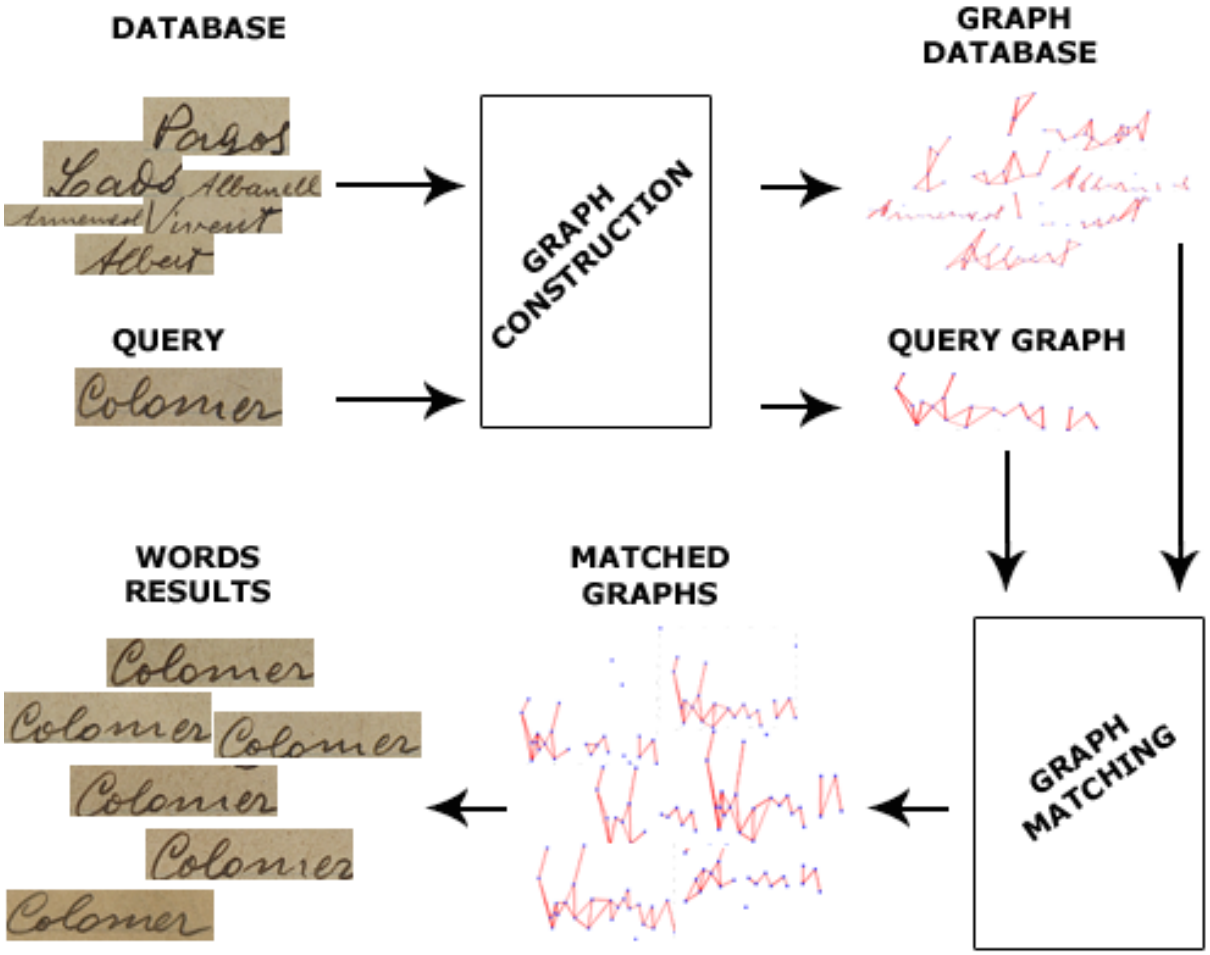
Word Spotting using a graph representation

Inexact Graph Matching

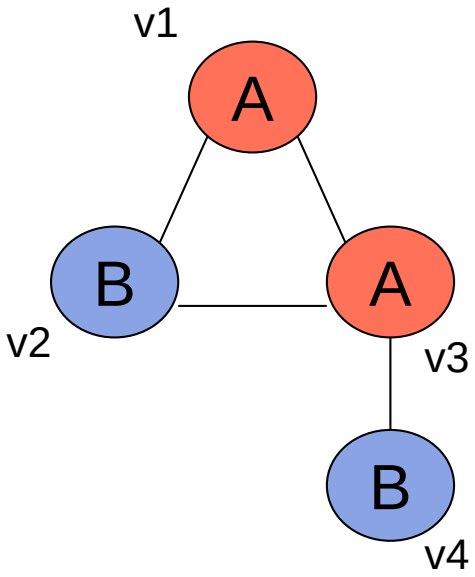
- **Graph Edit Distance** follows the idea of string edit distance.
- The minimum cost of operations required to transform one graph into the other. (vertices and edges).
  - **Insertion, deletion and substitutions.**
- **Bipartite Graph Matching**, suboptimal approximation to graph edit distance.



Graph Matching for Word Spotting



Segmentation-free? → Graph Embedding
Binary topological node feature



Label A Label B

ν_{v1}=[1,3,6,1,2,4] $\hat{\nu}_{v1}$=[0,1,1,0,0,1]

ν_{v2}=[2,2,7,0,3,3] $\hat{\nu}_{v2}$=[1,0,1,0,1,0]

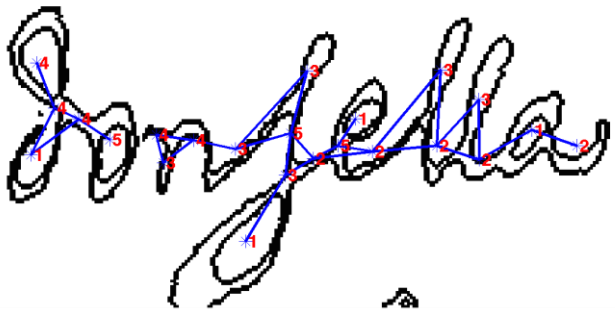
ν_{v3}=[1,4,6,2,1,7] $\hat{\nu}_{v3}$=[0,1,1,1,0,1]

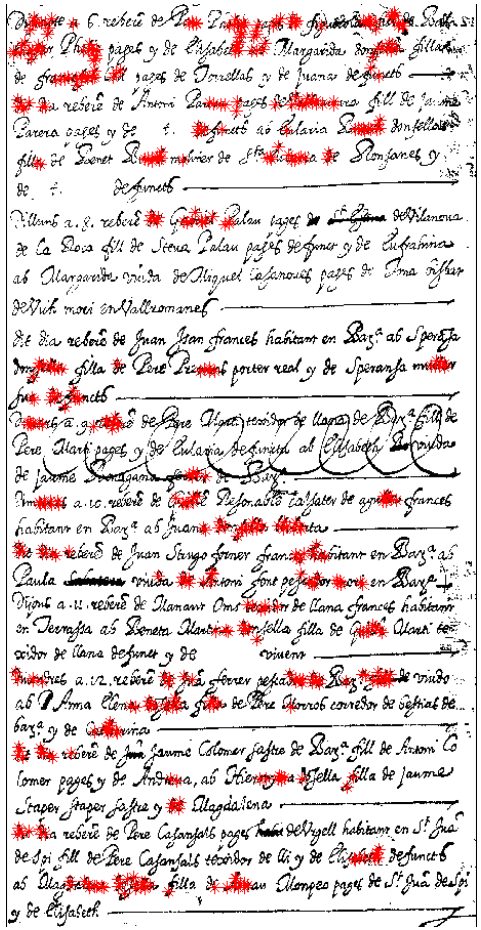
ν_{v4}=[1,1,4,0,2,1] $\hat{\nu}_{v4}$=[0,0,0,0,0,0]

4 paths of length 3 incident in v4 coming from nodes labeled with A. (tottering is allowed)

[v1, v2, v3, v4]
[v3, v1, v3, v4]
[v3, v2, v3, v4]
[v3, v4, v3, v4]

Experimental Results

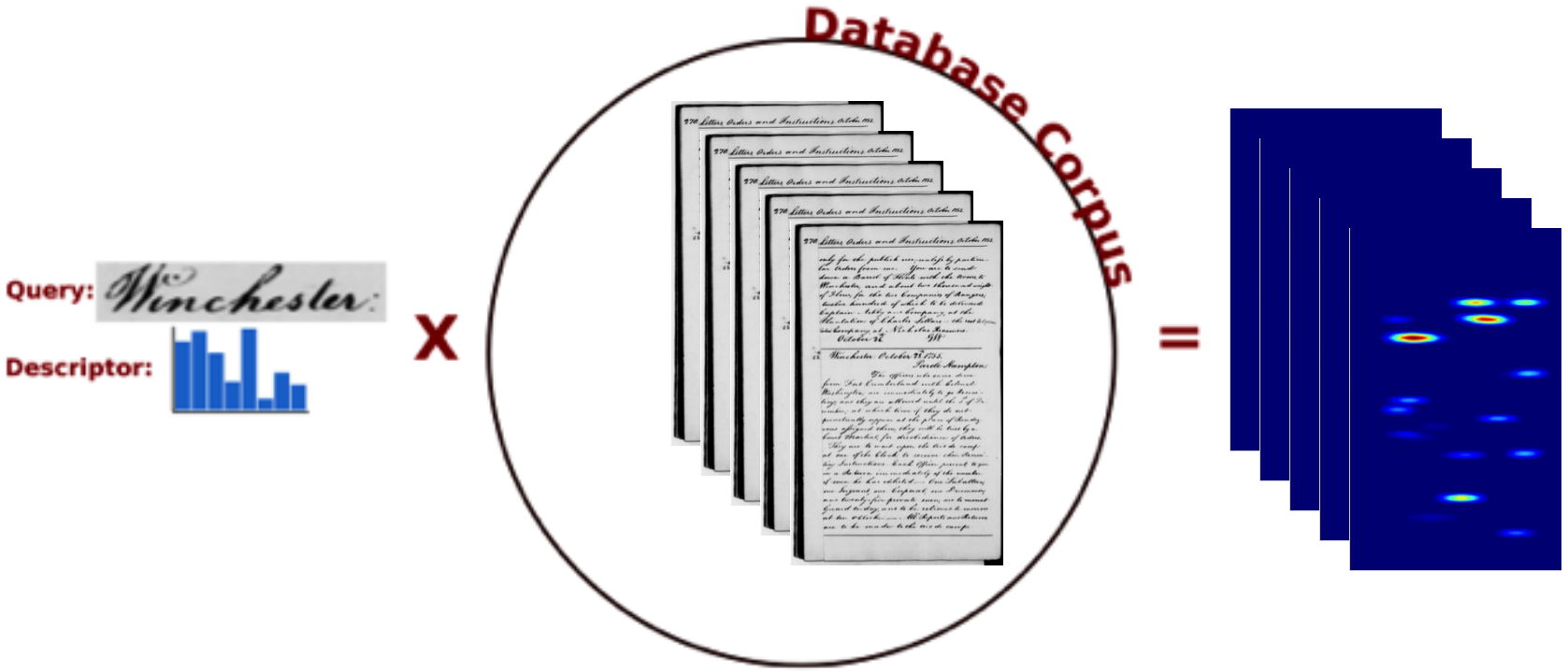




WORD SPOTTING LEARNING-FREE & SEGMENTATION-FREE

Segmentation-free methods

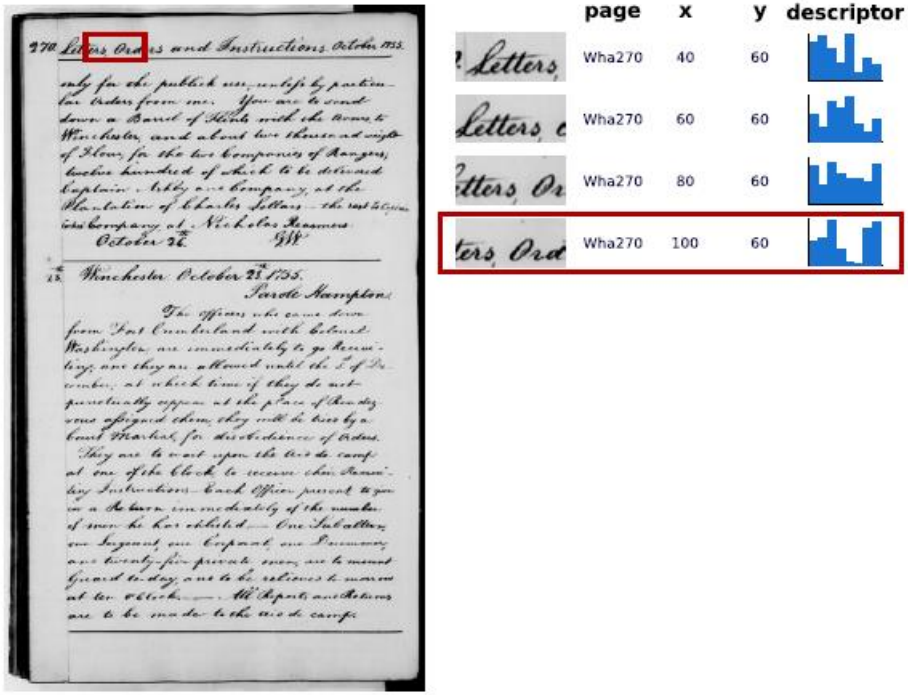
No segmentation of lines nor words



Word Spotting using Bag of Visual Words

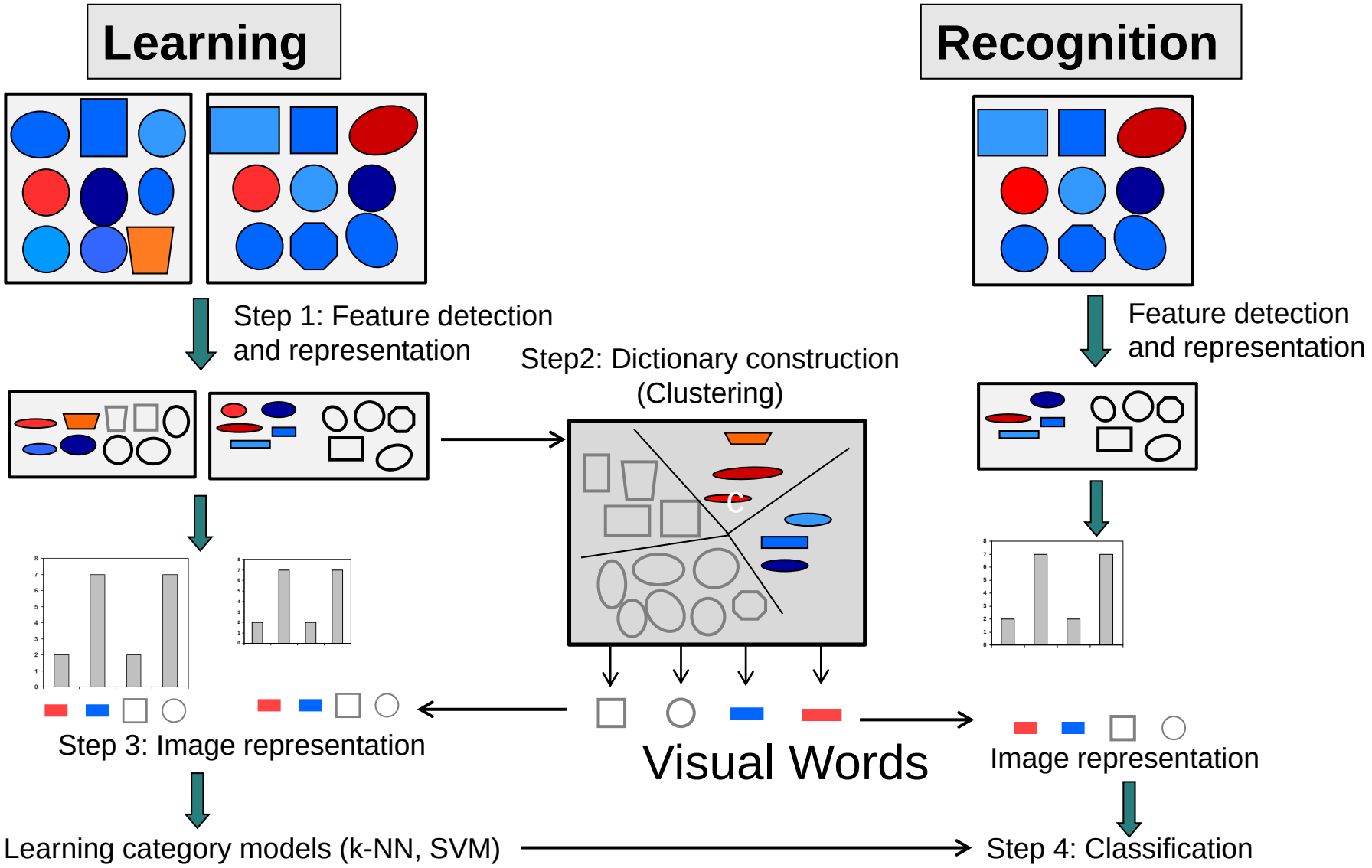
The method does not need any word nor line segmentation

Patch-based framework



M. Rusiñol, D. Aldavert, R. Toledo and J. Lladós. Efficient segmentation-free keyword spotting in historical document collections. Pattern Recognition, 2015

Bag of Words


Learning
Recognition
Step 1: Feature detection and representation
Step2: Dictionary construction (Clustering)
Feature detection and representation
Step 3: Image representation
Visual Words
Image representation
Learning category models (k-NN, SVM)
Step 4: Classification

Problem with Bag-of-Words

integral

Histogram= [a,e,g,i,l,n,r,t]

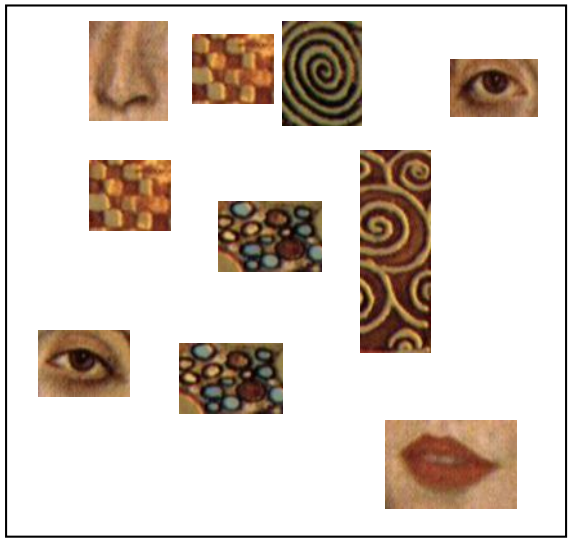
relating

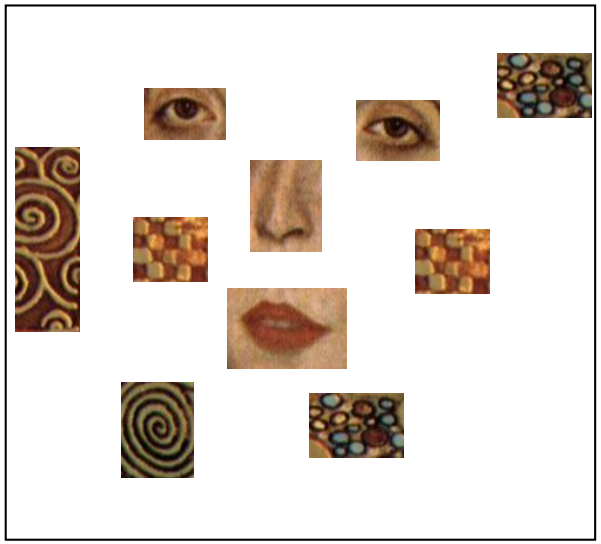
Histogram = [a,e,g,i,l,n,r,t]

triangle

Histogram = [a,e,g,i,l,n,r,t]

Problem with Bag-of-Words





All have equal probability for bag-of-words methods

Location information is important

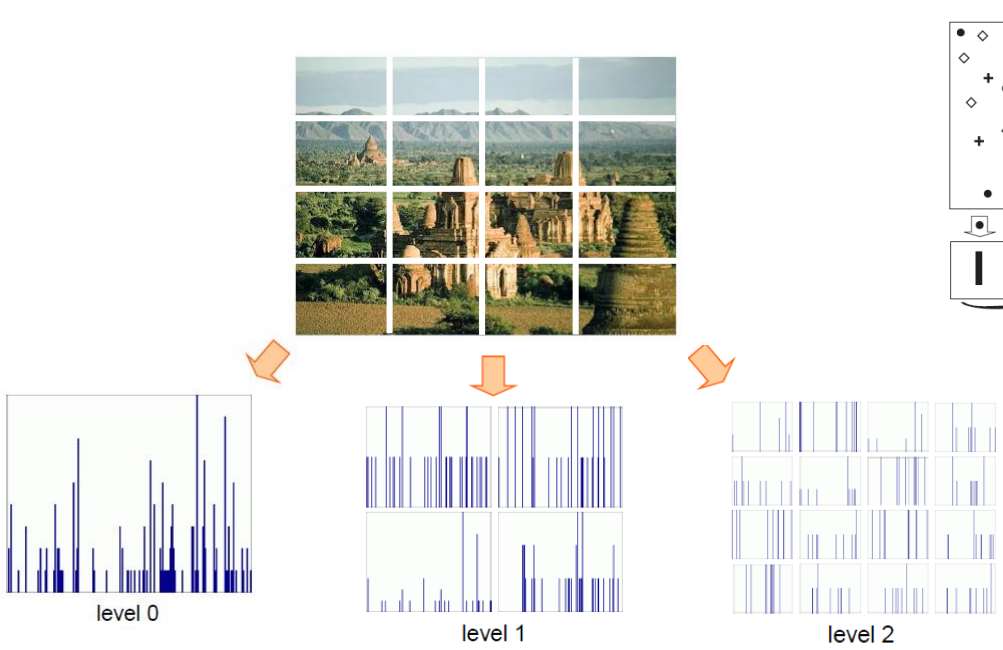
BoW + location still doesn't give correspondence

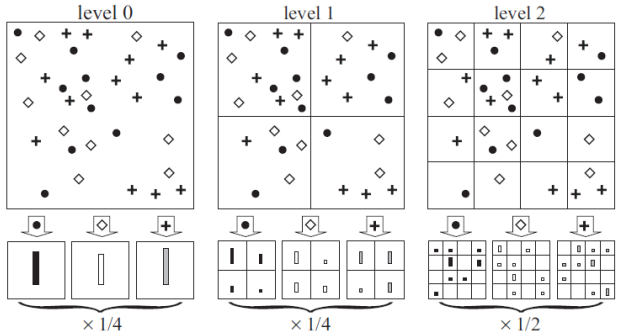
Slide credit: Li Fei-Fei

Improving BoW: Adding spatial information

Spatial Pyramids (Lazebnik, Schmid, Ponce, 2006)

- Hierarchical computation of histograms at different spatial image decompositions
- Combination of histograms at all levels of decomposition
- More weight to finer grids



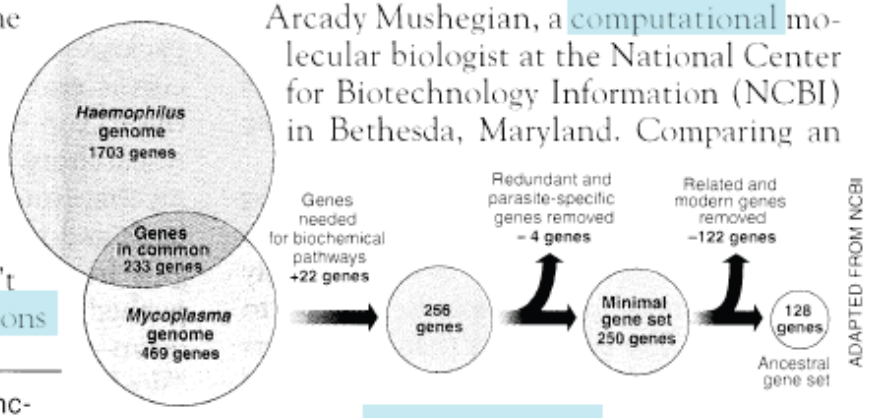


Improving BoW: Topic Models

Seeking Life's Bare (Genetic) Necessities

L: COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an



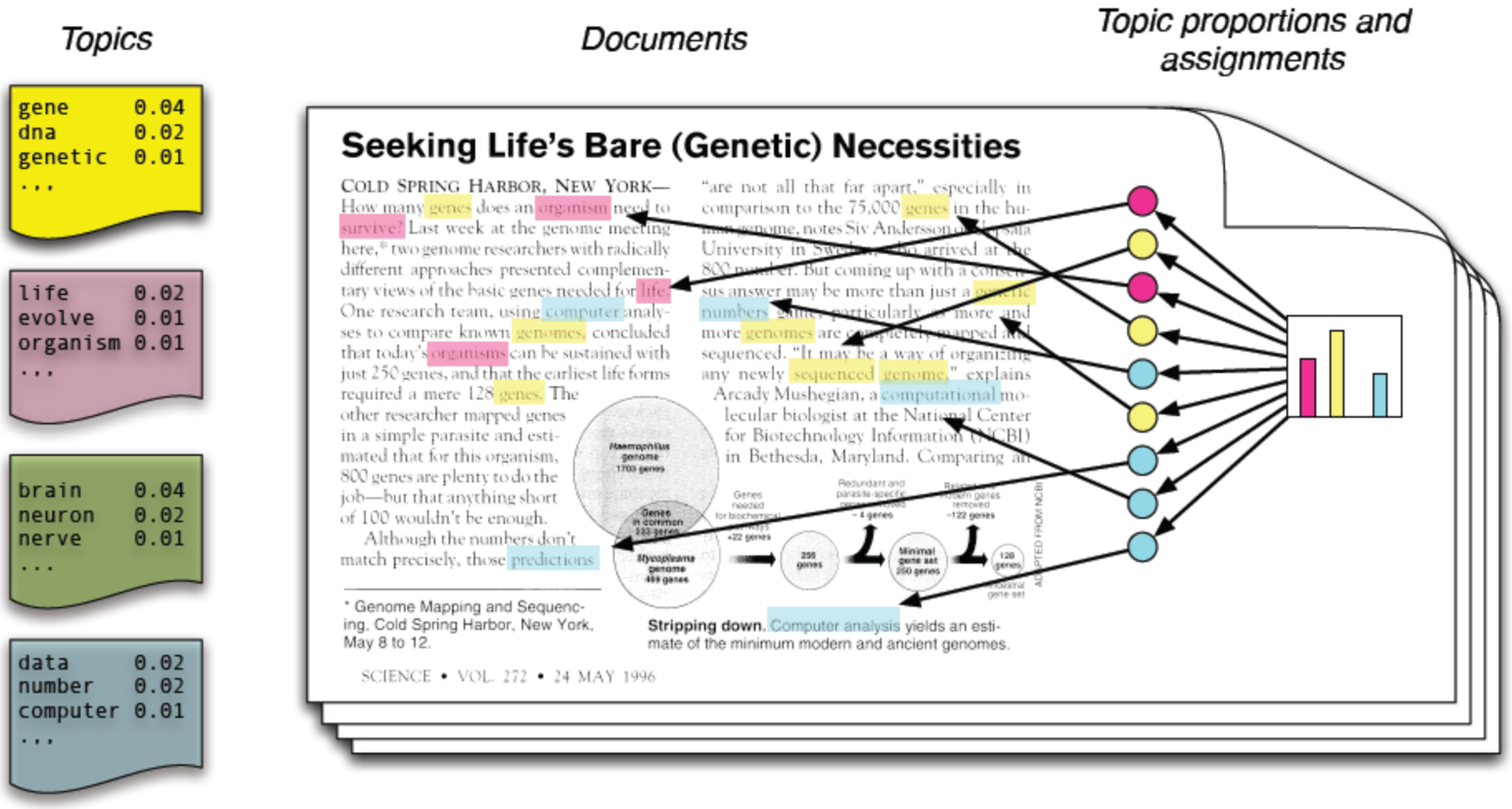
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 24 MAY 1996

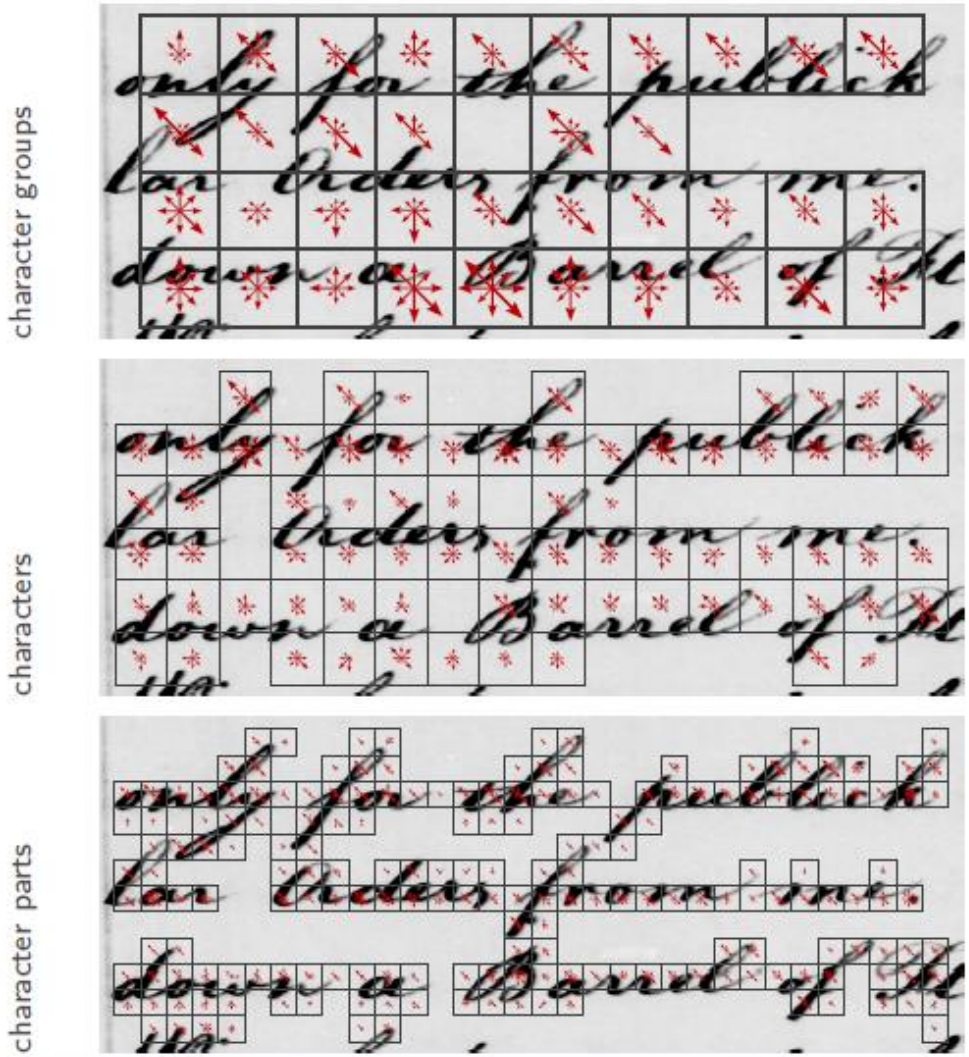
* D.M. Blei. Topic Models, (2009)

Improving BoW: Topic Models

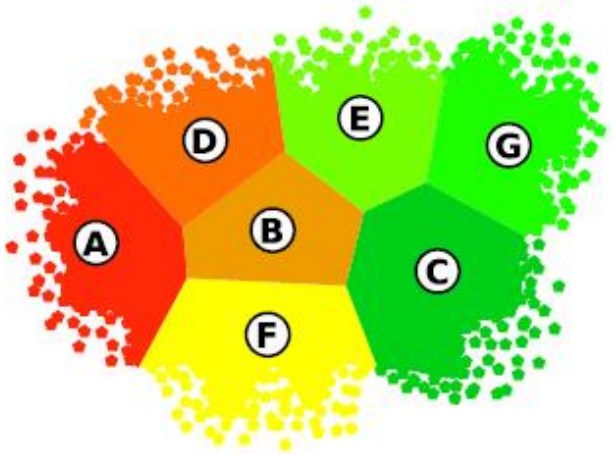


* D.M. Blei. Topic Models, (2009)

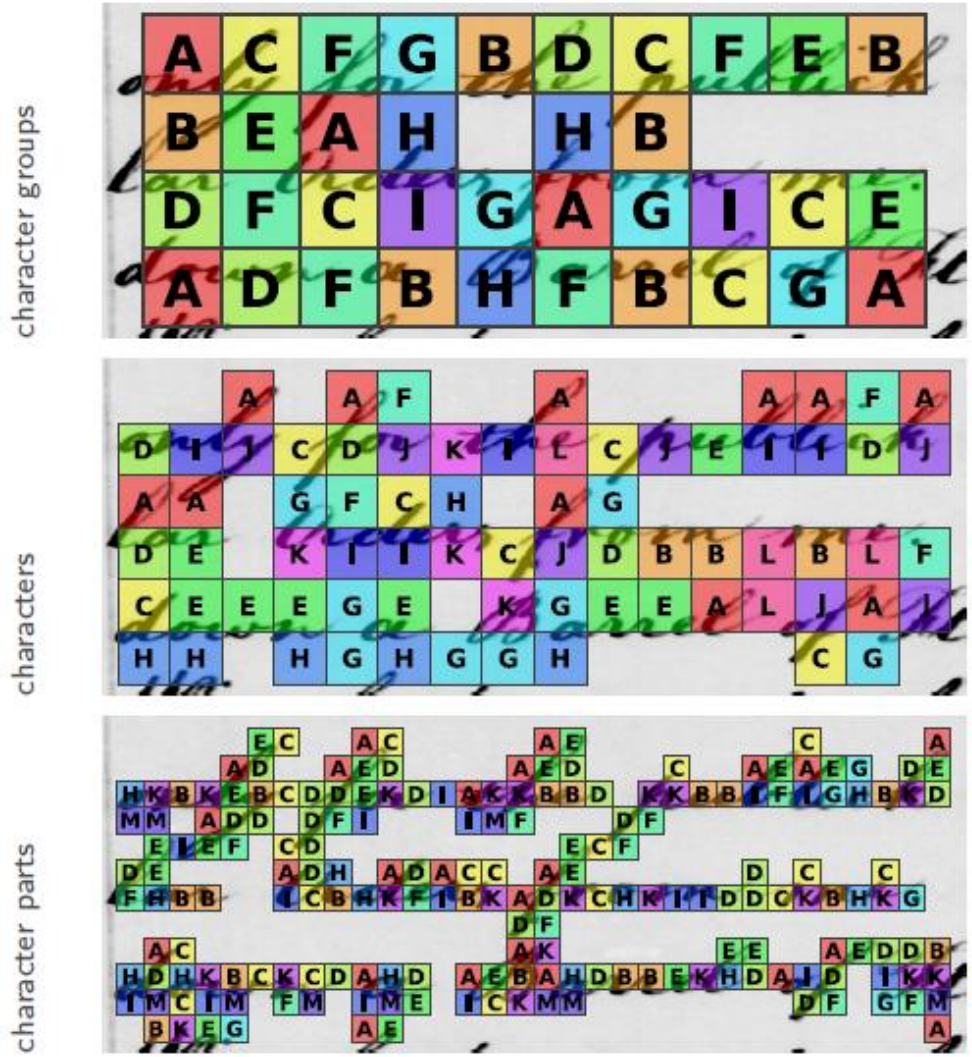
Bag of Visual Words for Word Spotting



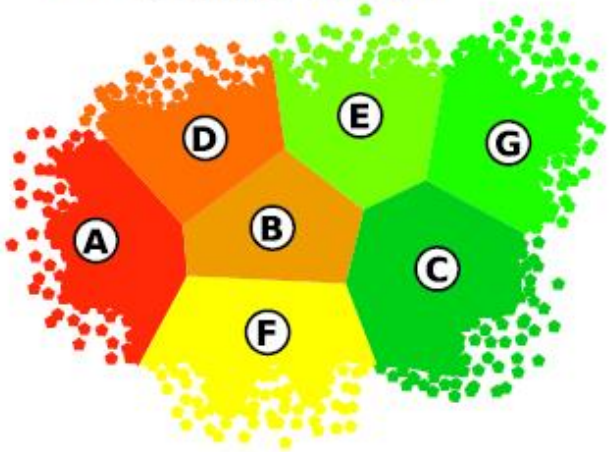
- SIFT features
- Densely extracted at 3 different scales
- Remove low-gradient features
- Build a codebook with k-means



Bag of Visual Words

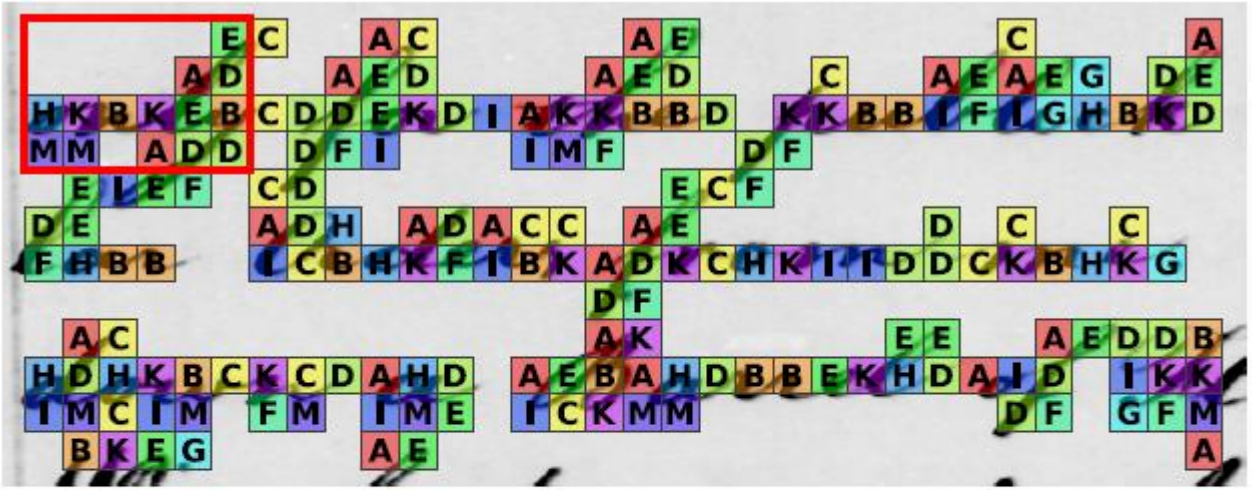


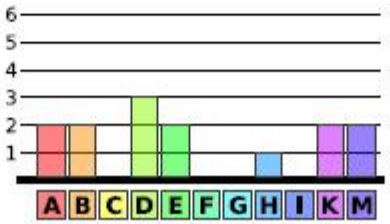
- SIFT features
- Densely extracted at 3 different scales
- Remove low-gradient features
- Build a codebook with *k*-means
- Quantization



Bag of Visual Words

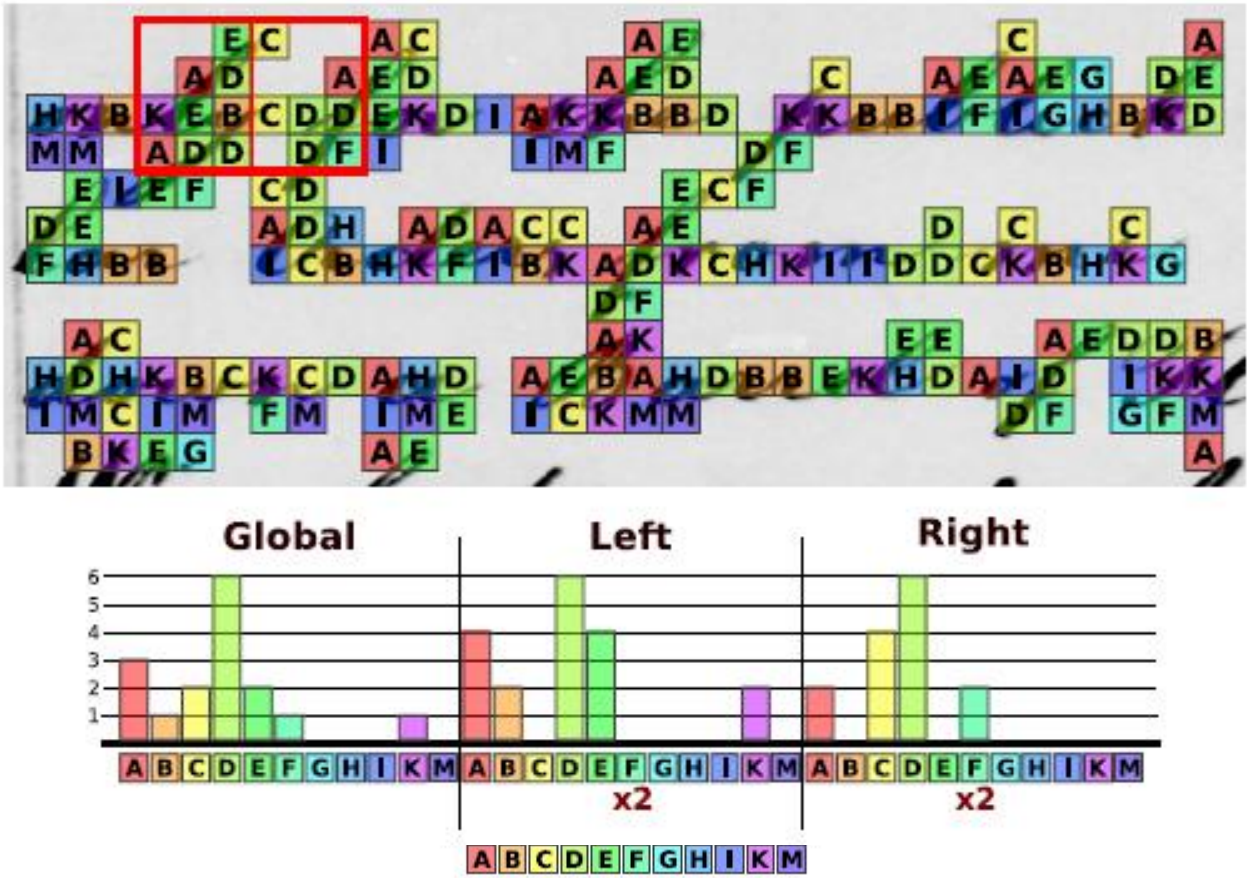
The document is splitted into overlapping local patches





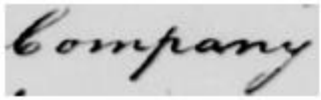
Bag of Visual Words

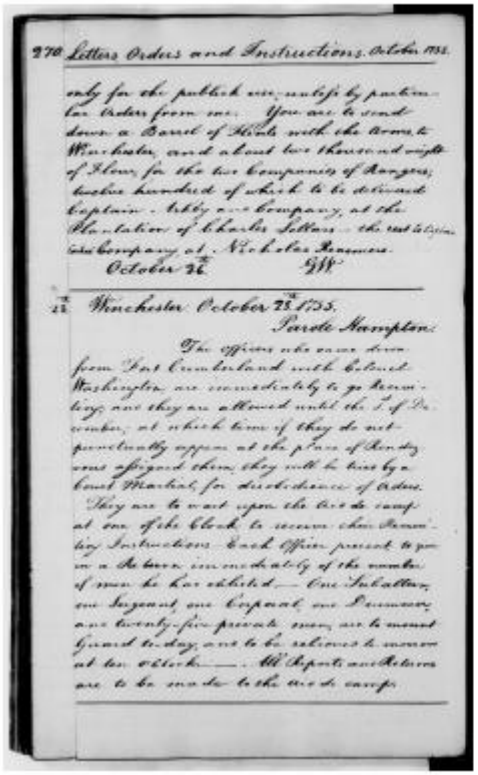
The document is splitted into overlapping local patches

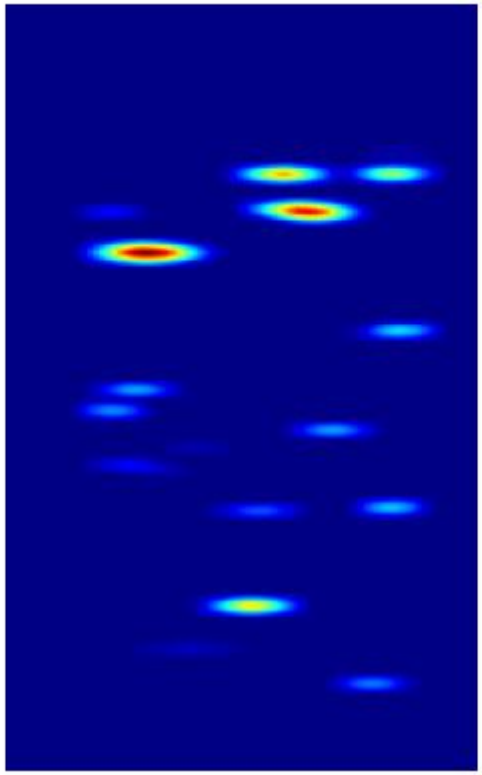


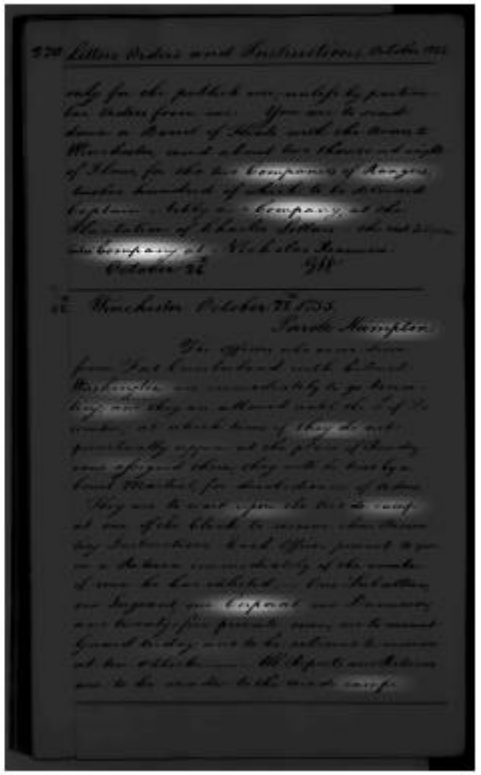
Bag of Visual Words

- A voting scheme provides the probability map where to find the queried word









Discussion: Representations

Table 4. Pros and Cons Summary.

	Size	Time	Indexability	Preprocessing	Performance	Scalability
BoVW	−−	+	+	+	++	− (size)
DTW	+	−−	−	−	+	− (time)
Pseudo-structural	++	++	++	−	+	−(discriminative power)
Structural	++	++	++	−−	−	− (discriminative power)

J. Lladós, M. Rusiñol, A. Fornés, D. Fernández and A. Dutta. On the Influence of Word Representations for Handwritten Word Spotting in Historical Documents. International Journal of Pattern Recognition and Artificial Intelligence, 2012.

Discussion: Techniques

Techniques

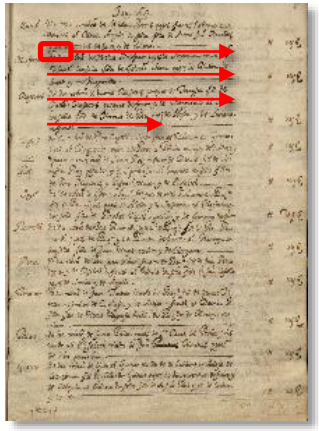
	Complexity Scalability	Performance	Generalizable Other scenarios	Room for improvements
Hashing /Indexing	++	-	+	++
Alignments (DTW / Graph / Trees)	-	++	-	-
Template matching (Euclidean distance)	+	+	--	-

WS - segmentation

Segmentation-free

✓ The image is divided into patches (e.g. using a sliding window).

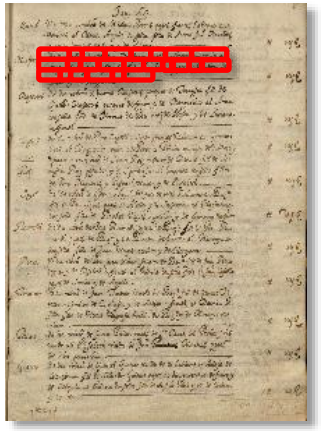
It is computationally costly in terms of time.



Segmentation-based

✓ The computational cost is low.

- It requires each document image to be segmented at word level.
- The word classification is strongly influenced by over or under-segmentations



Discussion: Segmentation-free

Segmentation-free methods

Advantages:

Do not depend on any segmentation method

Disadvantages:

Find parts of words (defender = end)

Discussion: Learning-free methods

- Learning-free methods
  - Advantages ☺
    - No training data required
    - Independent to alphabet and language
    - They are an image retrieval case
      - fast with indexing-hashing techniques
  - Disadvantages ☹
    - Difficult Multiple writer
      - How to deal with handwriting variations?
    - Difficult query-by-string
      - How to generate a “realistic” word from the ASCII text?

QUERY-BY-EXAMPLE
QUERY-BY-STRING

Classification of Word Spotting

Query-by-String

- ✓ Flexibility to search any kind of keyword.

It has to build a writing model and to map this model to the words images.

Labelled datasets are required in order to train the recognition engine.

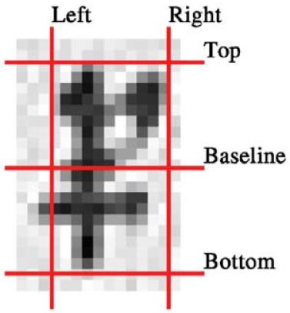
Query-by-Example

- ✓ It does not require learning.
- ✓ It only requires collecting one or several examples of the keyword.
- ✓ It does not need labeled training data.
- • It is sensitive to different writing styles.

Query-by-String (& Segmentation-free) approach

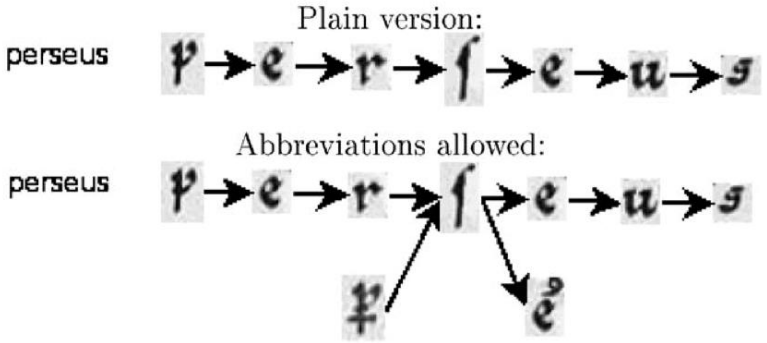
Writing Model is constructed

- For each letter in the alphabet, extract one image, and provide information about the linking behavior



When the user types the query word

- Generates different grammatical variations
- Links the letters according to the rules



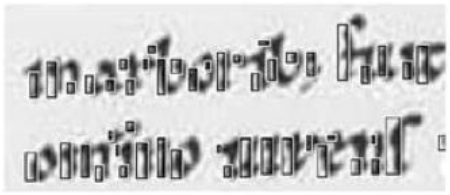
Y.Leydier, A.Ouji, F.LeBourgeois, H.Emptoz; "Towards an omnilingual word retrieval system for ancient manuscripts". PR(42), pp. 2089-2105, 2009.

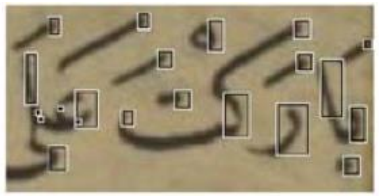
Query-by-String approach

- Features: Gradients

- Extract guides

  Inverse direction to the writing order

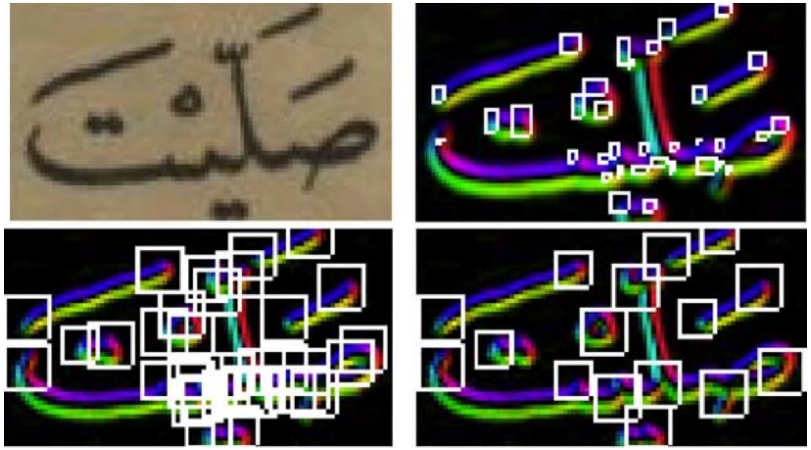




- ZOI location

  Expand while variation in angle gradients

  Reduce overlapping regions



Query-by-String approach

- Matching
  - ZOIs are stored in a tree
  - Search the tree through the lowest score path
  - Prune branches with high scores

Advantages / Disadvantages

- Query-by-string ☺
- Model has to be created ☹

LEARNING-BASED WORD SPOTTING

Difficulties

Introduction: word spotting

Learning-based Keyword Spotting

Advantages:

- Higher precisions and recall rates can be expected.
- Flexible systems, can be used for Document filtering and database search.
- More sophisticated system allow the inclusion of more, external data (e.g. language models) to further increase the recognition

Disadvantages:

- A (large) training set is needed
- Fixed costs for training the system have to be taken into account, whenever new writing styles appear.

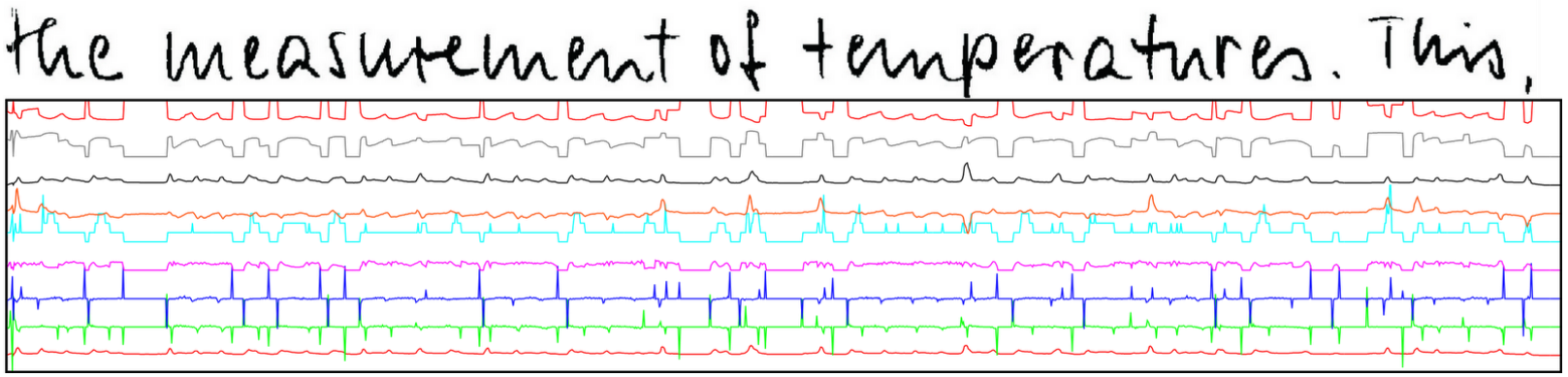
Approaches

Idea: evaluate efficiently a matching score between the query and the text line

Common Approaches:

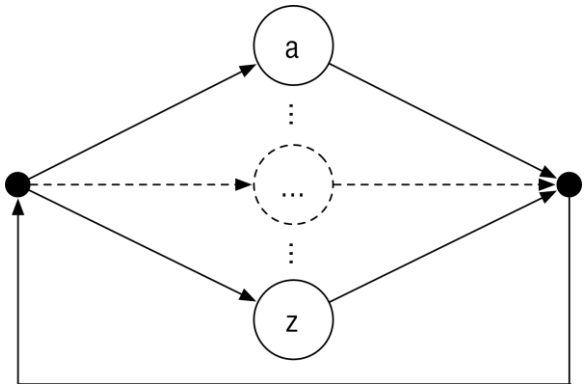
- Hidden Markov Models
- Neural Networks: BLSTM NN

Sequential Representation of a Text Line

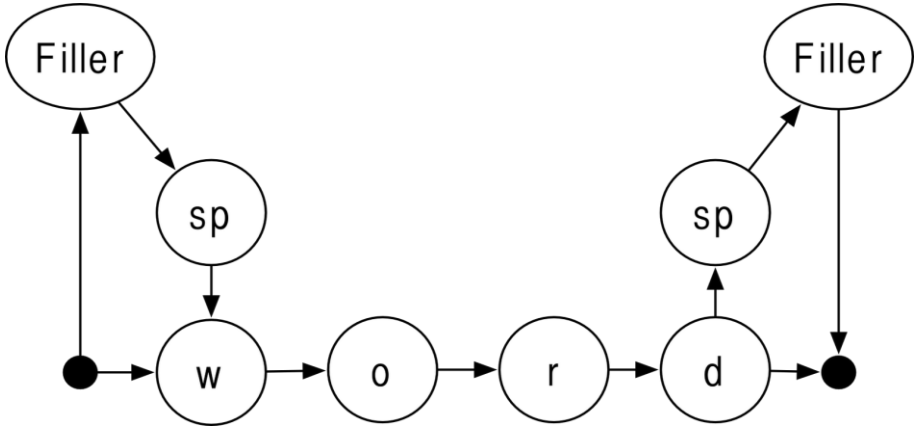


HMM for Word Spotting

Given a Keyword, construct 2 HMMs



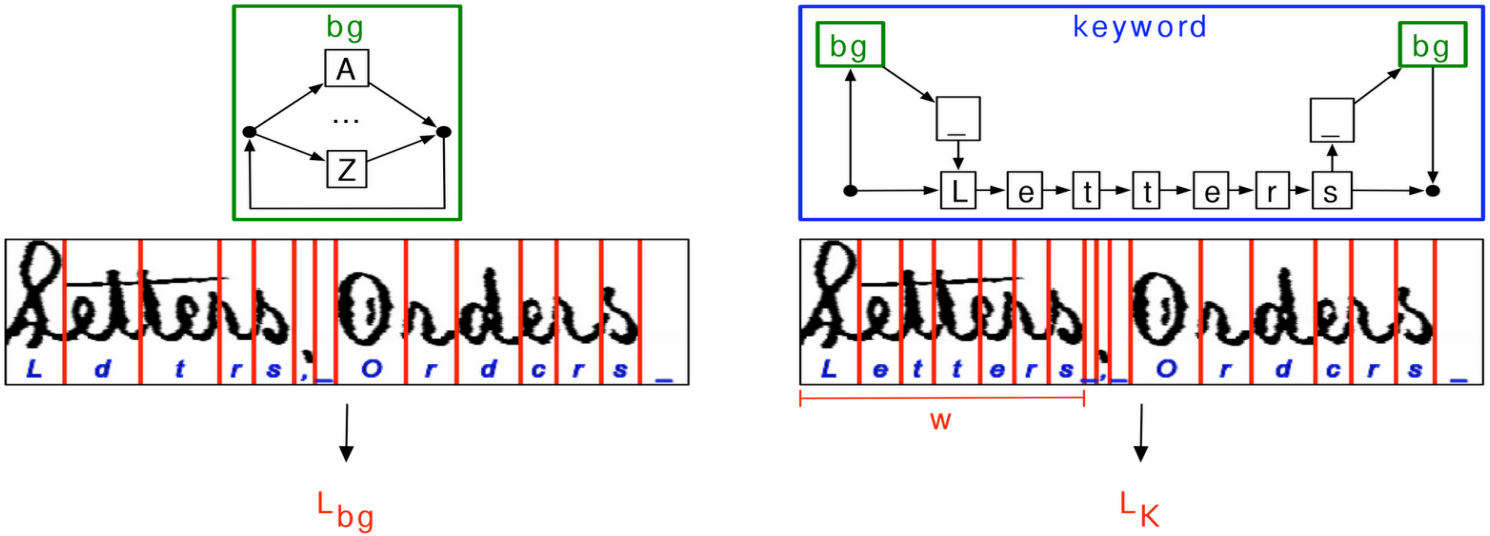
The Background-HMM



The Keyword-HMM

HMM for Word Spotting

Perform Viterbi recognition using both models



Consider likelihood ratio $R=L_K/L_{bg}$ $(0 \leq R \leq 1$, since $L_{bg} \geq L_K)$

Return positive match, iff $R/w > T$, with

keyword width w and threshold T

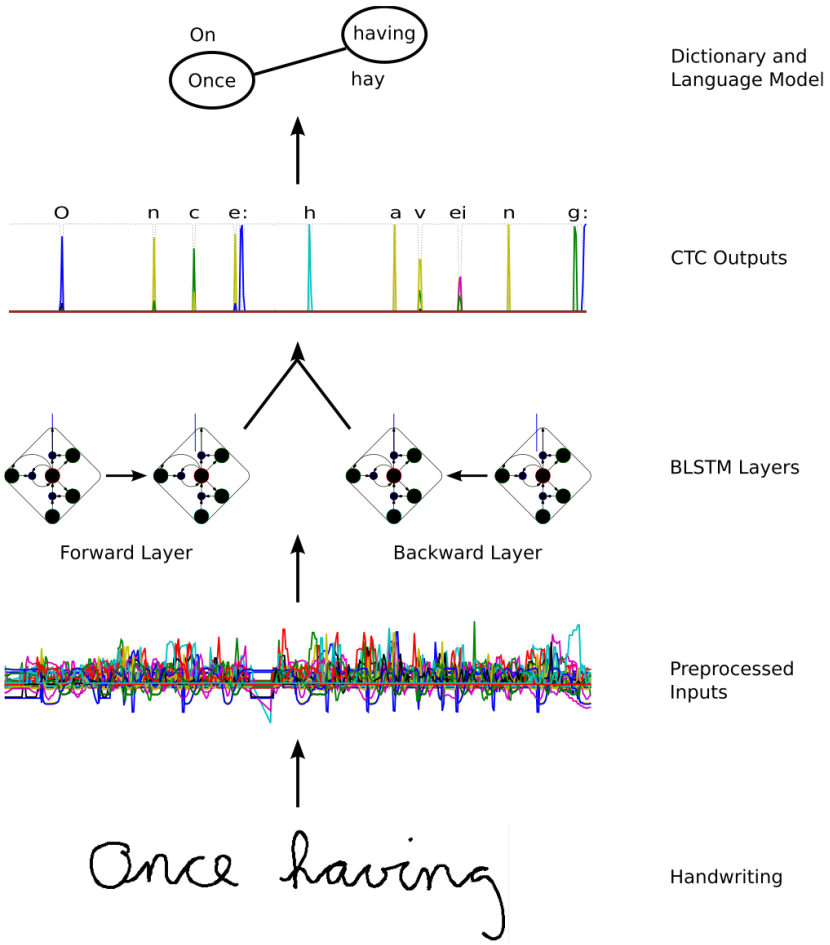
Handwriting Recognition using BLSTM NN

Bidirectional Neural Networks + n-grams

Graves et al., "A novel connectionist system for unconstrained handwriting recognition", PAMI 2009

- Recurrent Neural Networks
- Bidirectional (BLSTM blocks)
- CTC: Dynamic Programming
- Word n-grams + dictionary

- Best results so far



Bidirectional Long-Short Term Memory Neural Networks

BLSTM NN are bidirectional recurrent NN with a specialized LSTM hidden layer

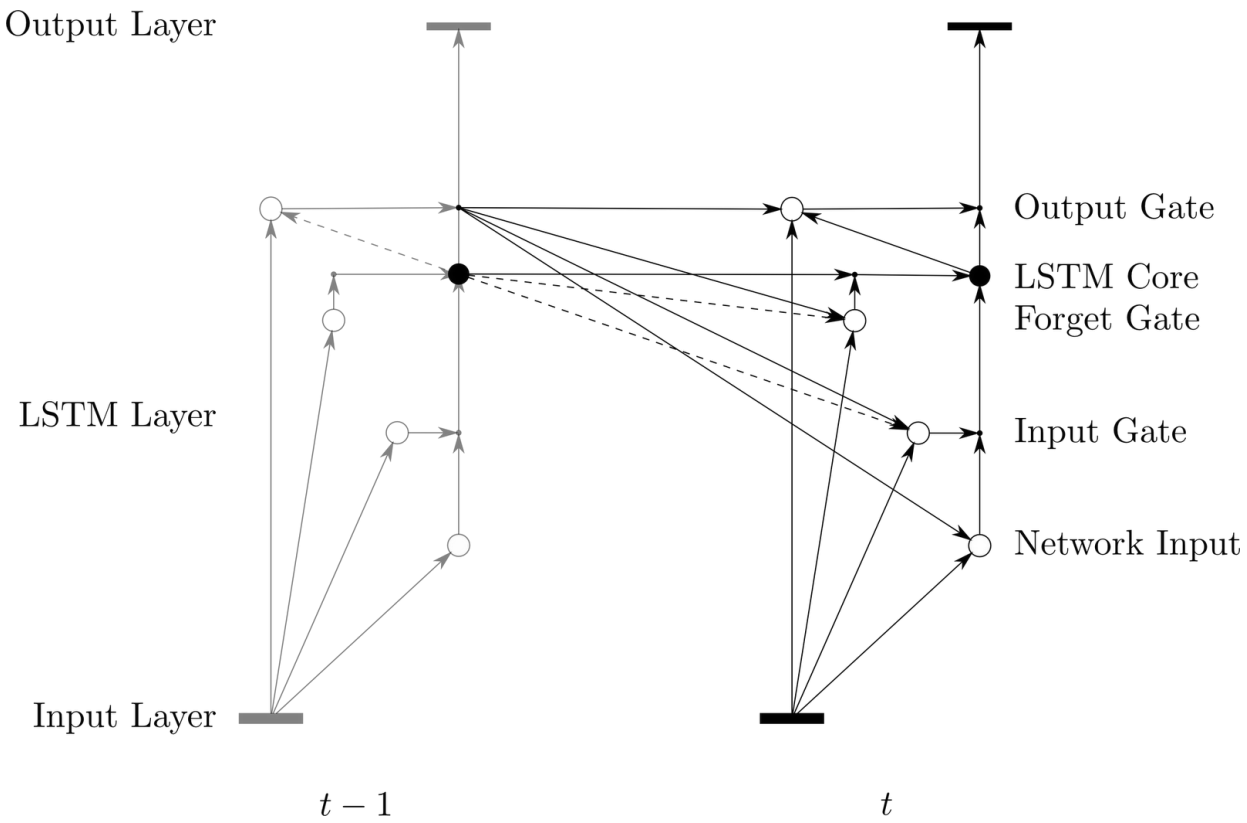
LSTM: Long Short-Term Memory

- A core of the cell stores a value
- Information flow into and out of each cell is controlled by gates
- A forget gate can reset the core's value to 0
- LSTM cells are differentiable memory cells
- LSTM cells overcome the vanishing gradient problem:
  - > Backpropagation for several time-steps is hard
  - > Recurrent NN can therefore not be trained to consider more than just a few time steps

Frinken V, Fischer A, Manmatha R, Bunke H, A Novel Word Spotting Method based on Recurrent Neural Networks, IEEE Trans Pattern Analysis and Machine Intelligence (PAMI), 34(2), pp. 211-24, 2012

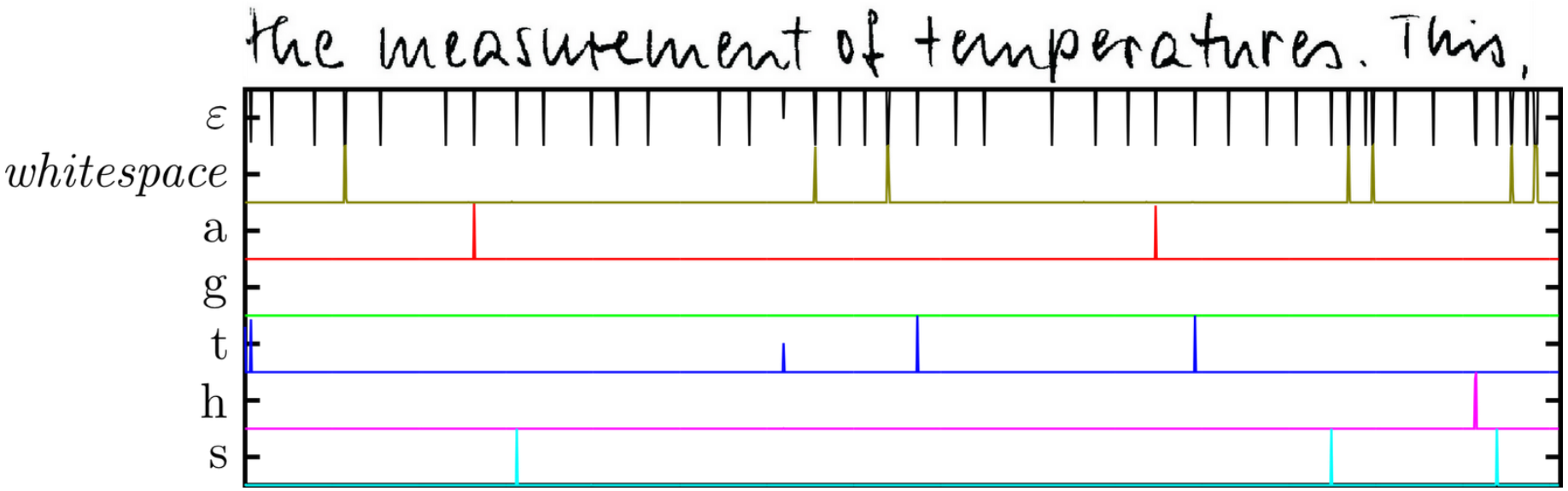
Bidirectional Long-Short Term Memory Neural Networks

The information flow within a LSTM neural network

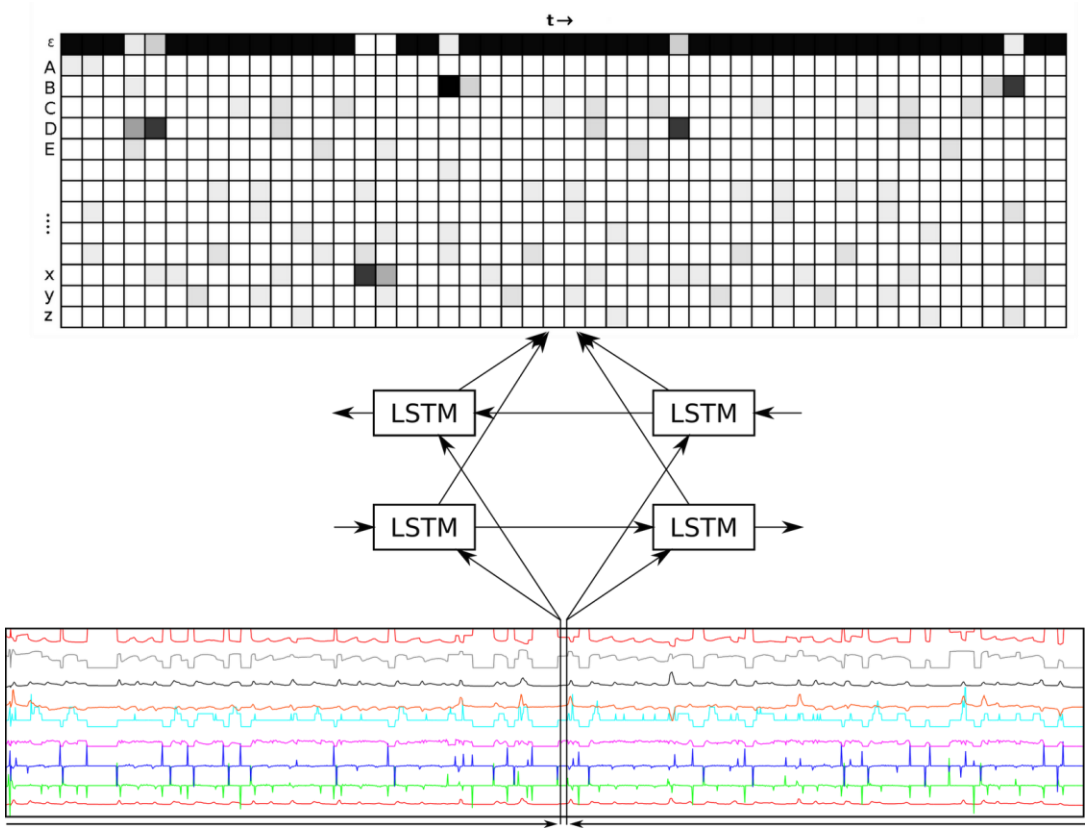


BLSTM NN with CTC Output Layer

The neural network can be trained to detect characters.



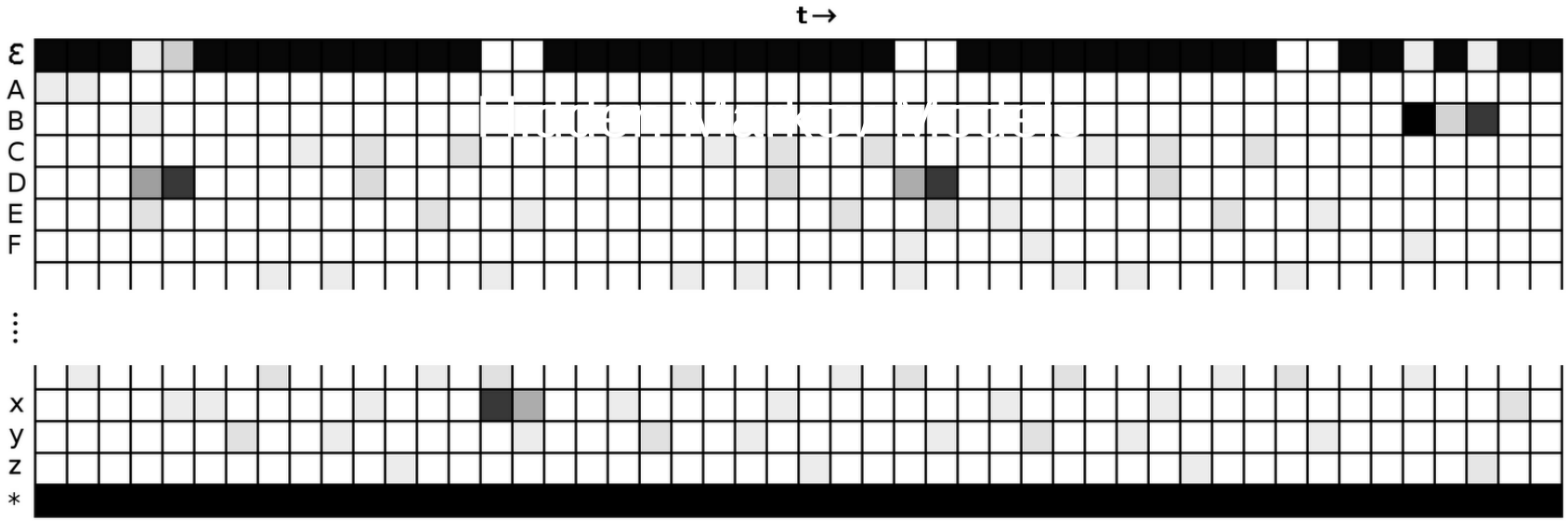
BLSTM NN for Word Spotting



Take home message: For each position, a character probability vector is returned

BLSTM NN for Word Spotting

1. Add a new row to the matrix with a constant activation of 1.0



BLSTM NN for Word Spotting

1. Add a new row to the matrix with a constant activation of 1.0
2. transform the keyword, e.g., **word**:

2. a) Add the ε-character before and after each character of the search phrase:

2. b) Add the *-character before and after the new search phrase

word → * ε w ε o ε r ε d ε *

BLSTM NN for Word Spotting

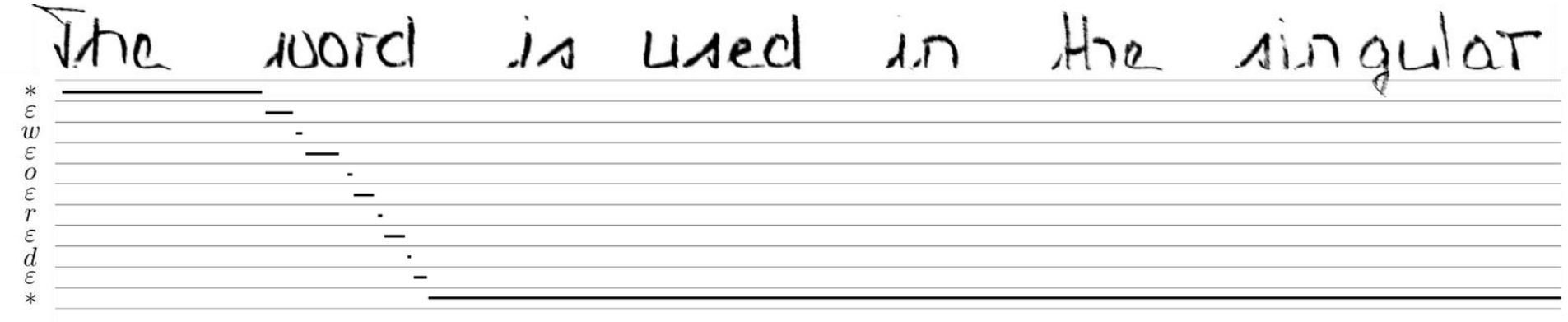
1. Add a new row to the matrix with a constant activation of 1.0
2. transform the keyword, e.g., **`word`**:

2. a) Add the ε-character before and after each character of the search phrase:

2. b) Add the $*$-character before and after the new search phrase

3. Use dynamic programming to search for the best path in the matrix that follows the characters of the search word

4. The matching score is then the product of the path's nodes



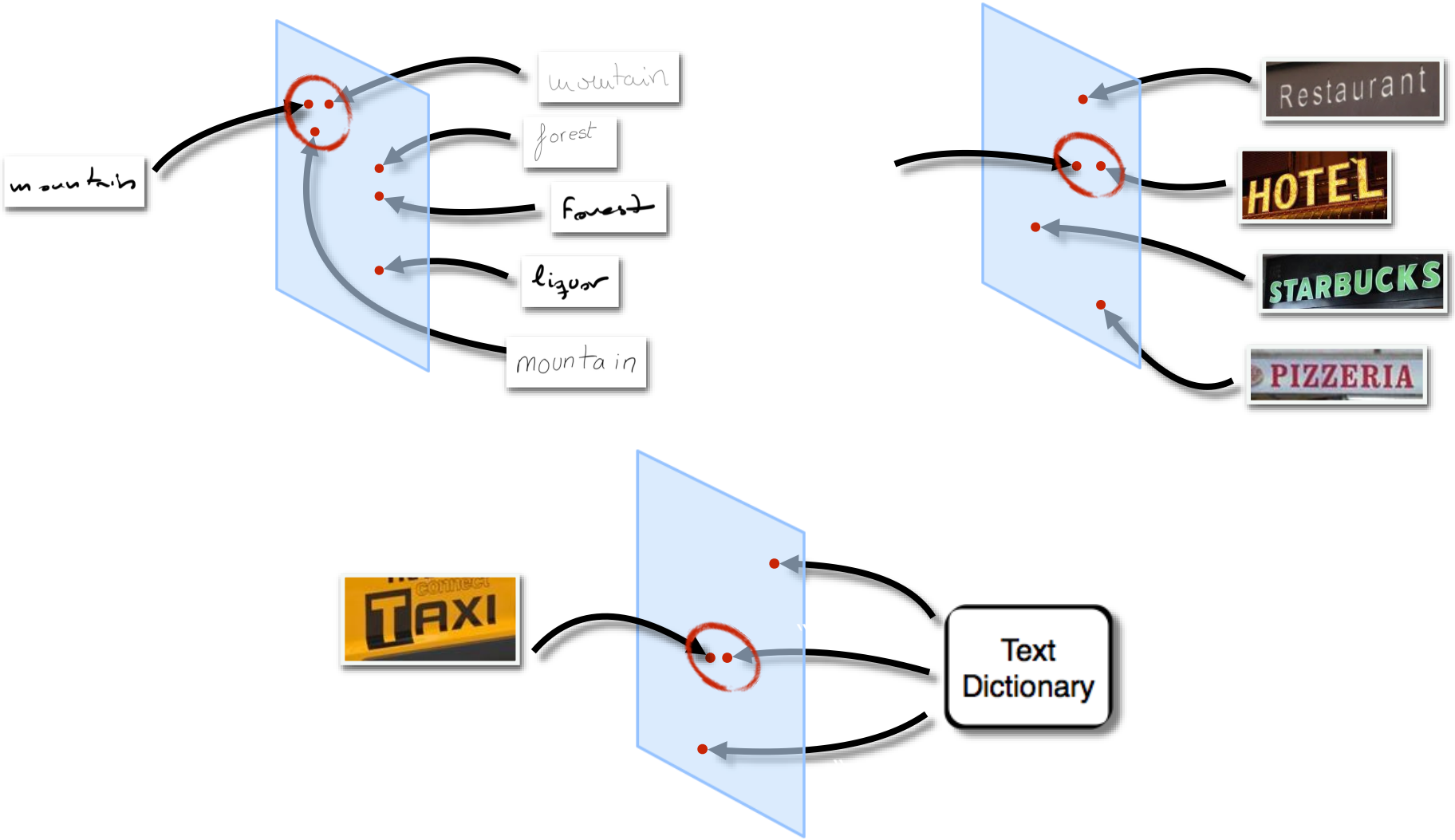
Query-by-Example + Query-by-String approach

Segmentation-based approach

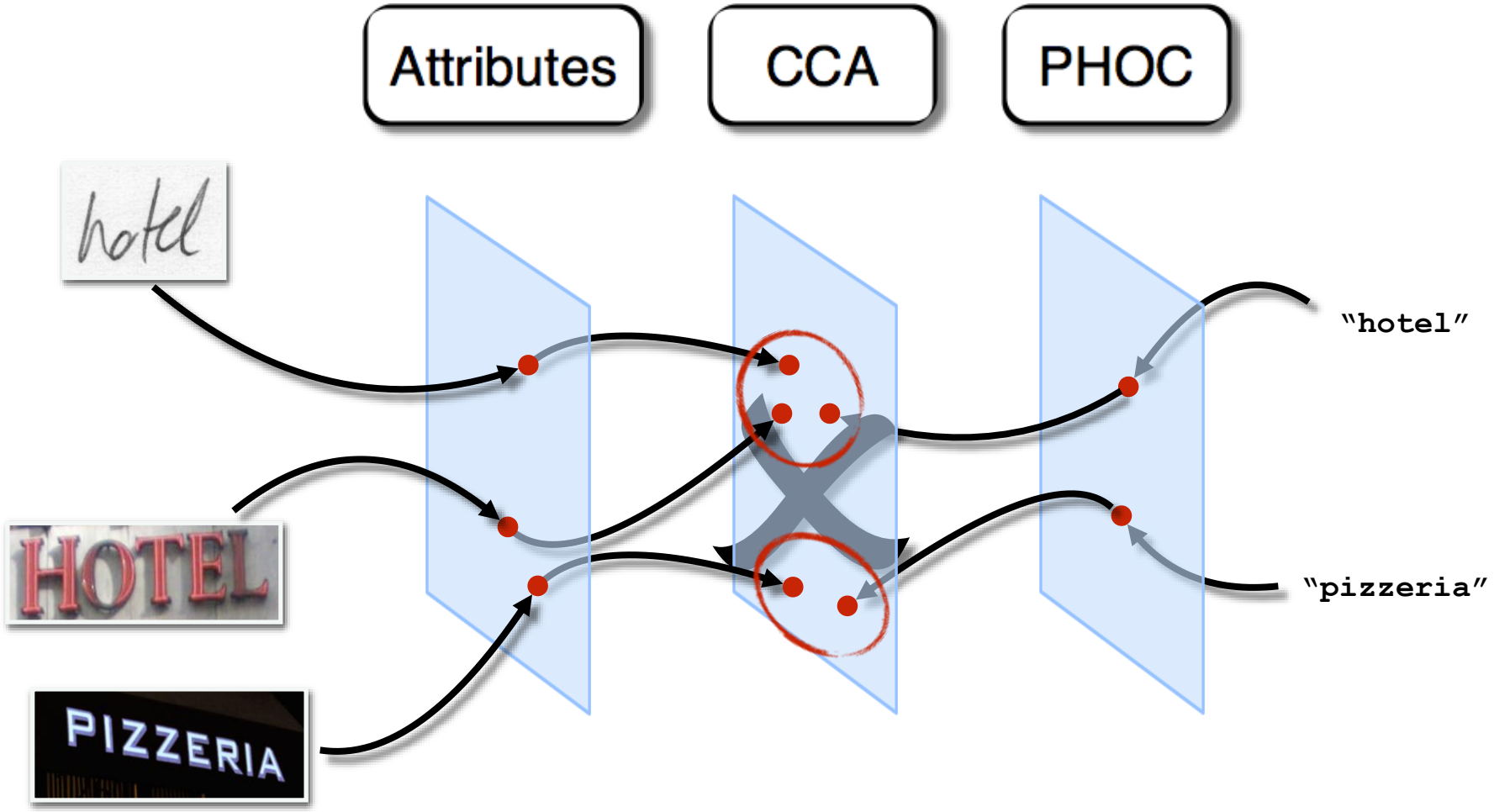
Query-by-string and Query-by-example

J.Almazán, A.Gordo, A.Fornés, E.Valveny. Word Spotting and Recognition with Embedded Attributes. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), volume 36, issue 12, pages 2552-2566, 2014.

Unified Framework



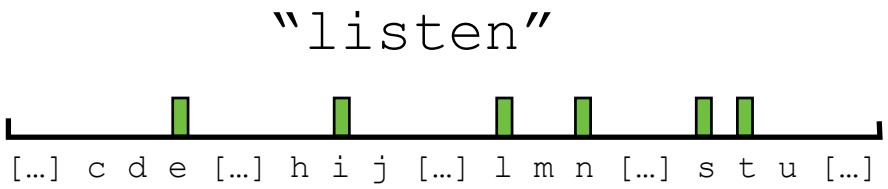
Proposal



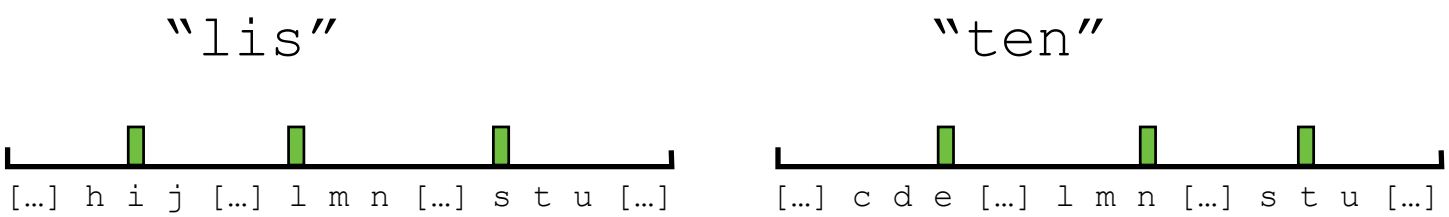
Label Embedding

Proposal: Pyramidal Histogram of Characters (PHOC)

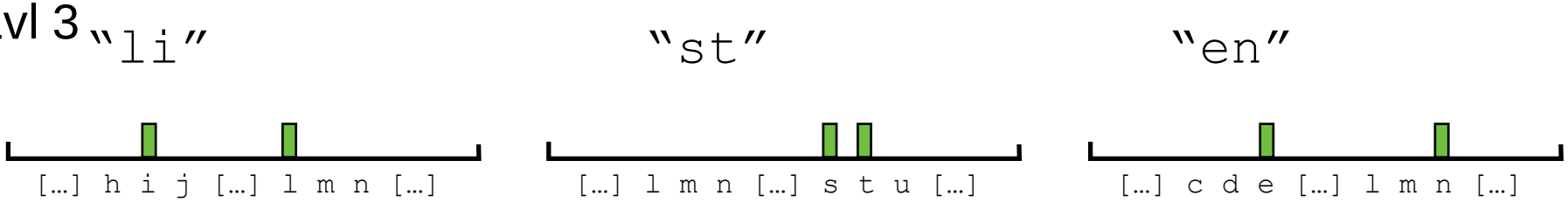
Lvl 1



Lvl 2



Lvl 3



Attribute Embedding

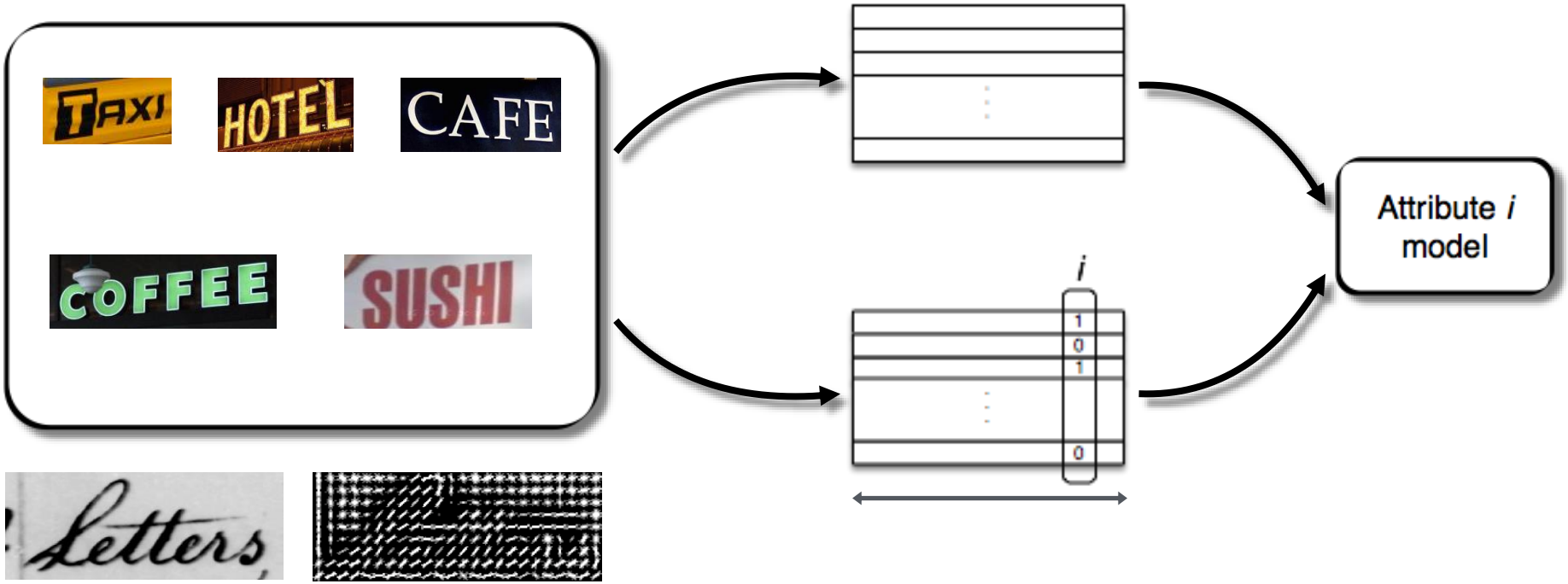


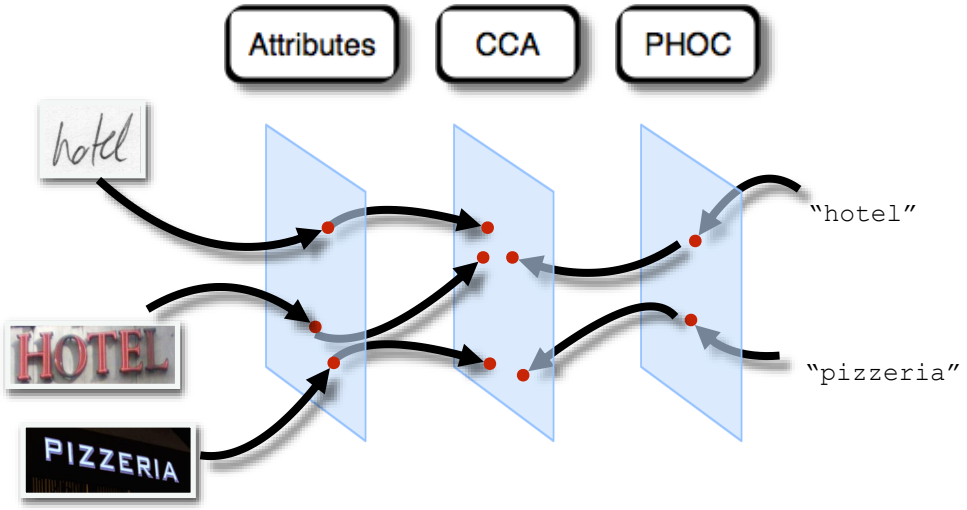
Image representation: Fisher Vector (FV) over SIFT+XY and 2x6 SP (FV can be seen as a bag of words encoding higher-order statistics)

Attribute model: linear SVM

Common Subspace

Canonical Correlation Analysis (CCA) to learn a common subspace between attribute scores and PHOCs

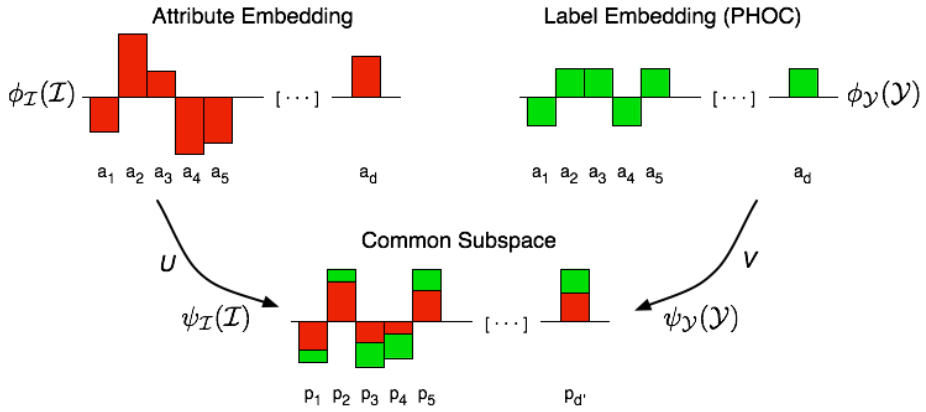
CCA finds a common space where the correlation of the projected views is maximal



Common Subspace

Canonical Correlation Analysis (CCA) to learn a common subspace between attribute scores and PHOCs

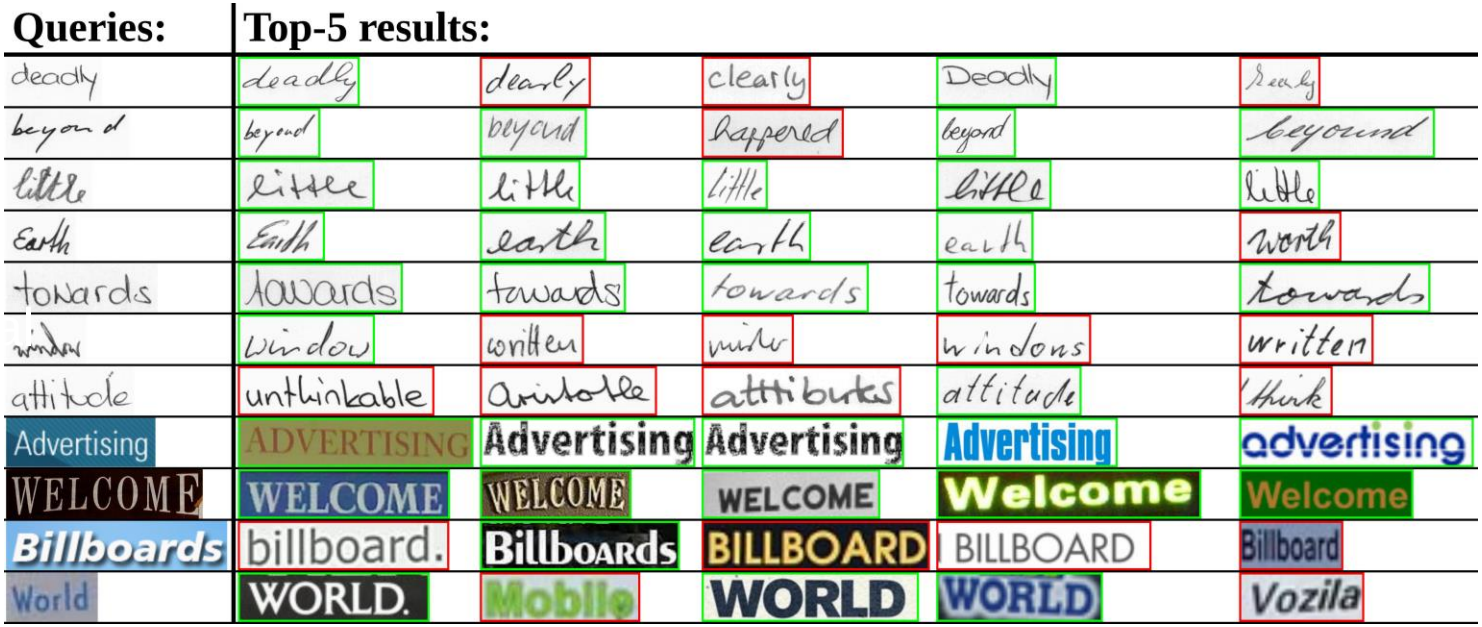
CCA finds a common space where the correlation of the projected views is maximal

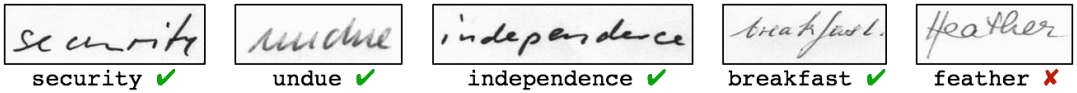


Three advantages:

- Comparison between image and text embeddings is meaningful
- Attributes scores of images of the same word are brought together
- Dimensionality reduction **(96 dimensions)**

Qualitative Results





Demo application

Segmentation-based WS approach

Query-by-string and Query-by-example

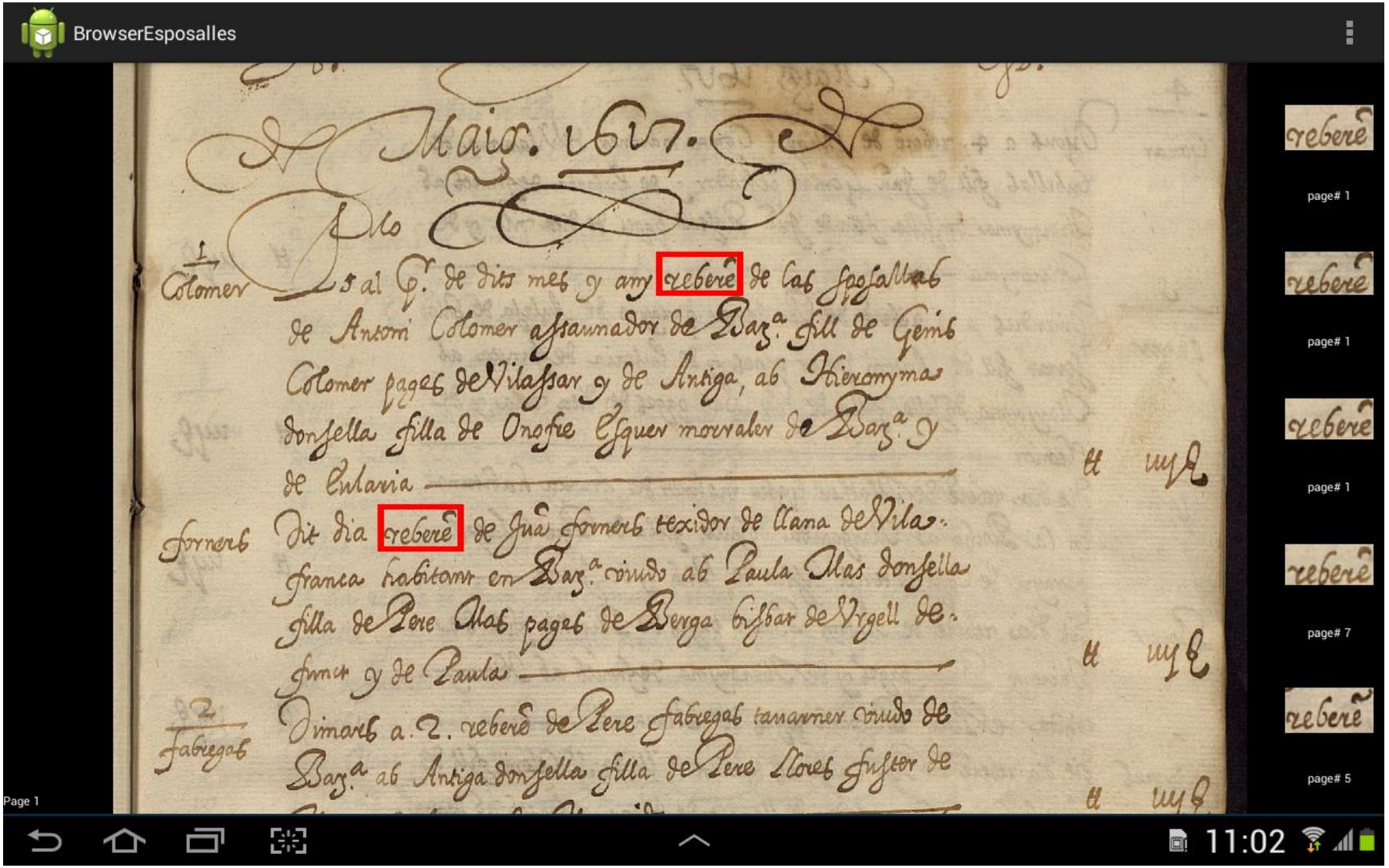
Combined searches

- *AND*
- *OR*

https://www.youtube.com/watch?v=1wUq51t5_tc

P.Riba, J.Almazán, A.Fornés, D.Fernández, E.Valveny, J.Lladós. e-Crowds: a mobile platform for browsing and searching in historical demography-related manuscripts. International Conference on Frontiers in Handwriting Recognition (ICFHR), 2014.

Demo application in Tablet device (Android)



Query-by-String vs Query-by-Example discussion

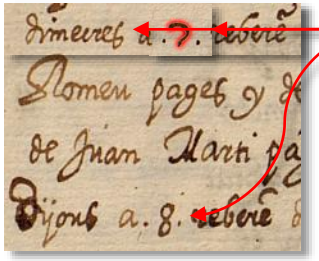
Query-by-string → *USER LOVES IT!* ☺

Query-by-Example	Query-by-String
Need to find one occurrence manually	User types the text query (keyboard)
Does not handle different handwriting styles	Can handle different handwriting stlyes
Does not handle grammatical alternatives	Handles grammatical (spelling) alternatives
No prior knowledge need	Need to build a writing model

CONTEXT AWARE WORD SPOTTING

Word Spotting Difficulties

- Efficient strategy in the access by content to historical manuscripts when explicit recognition is not possible.
- **Problem:** Word spotting is usually built based solely on the statistics of local terms which make it **difficult** the recognition in some cases:



Whether it says "7" or "1", it should be "7".

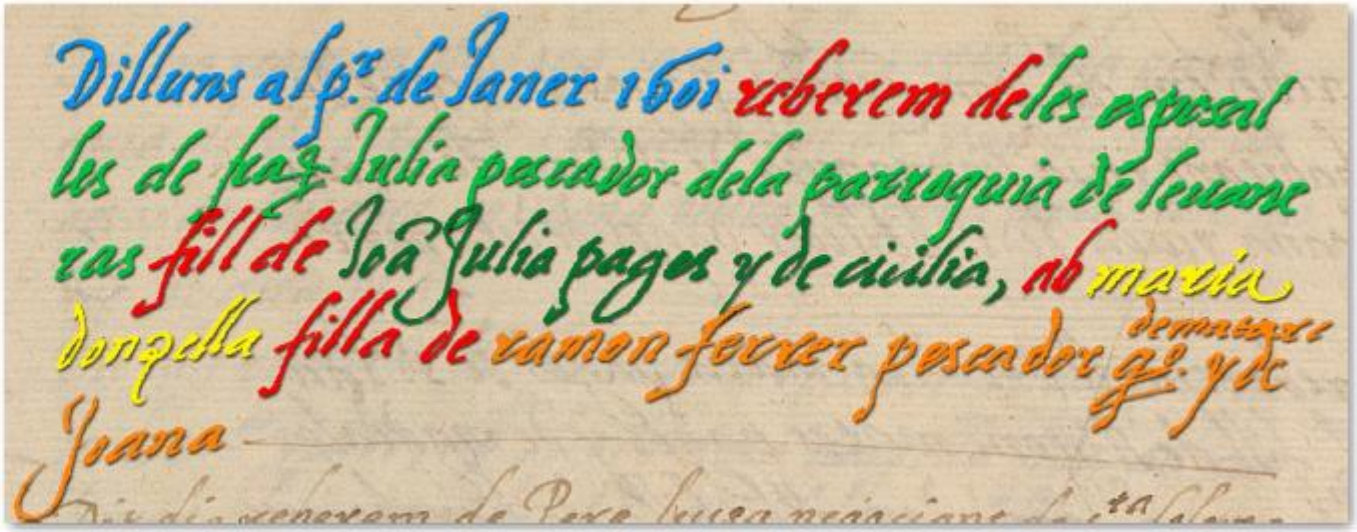
- **Solution: Contextual information** can provide more relevant information for the recognition of a query word than intrinsic word image information.

The role of the context in visual object recognition

- Classical word spotting is based on the statistics of local terms.
- **Contextual information** can provide more relevant information for the recognition of a query word than intrinsic word image information.
- **Context** of a word in the document can be defined in terms of the other recognized words and their mutual dependences.

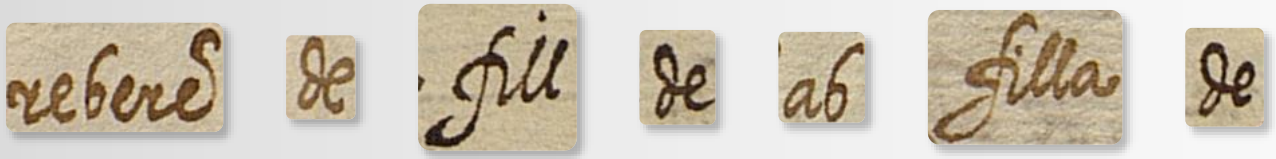
Context: The structure of a marriage record

- Date (DI)
- Husband (HI)
- Husband's Parents (HPI)
- Wife (W)
- Wife's Parents (WPI)
- Key-words

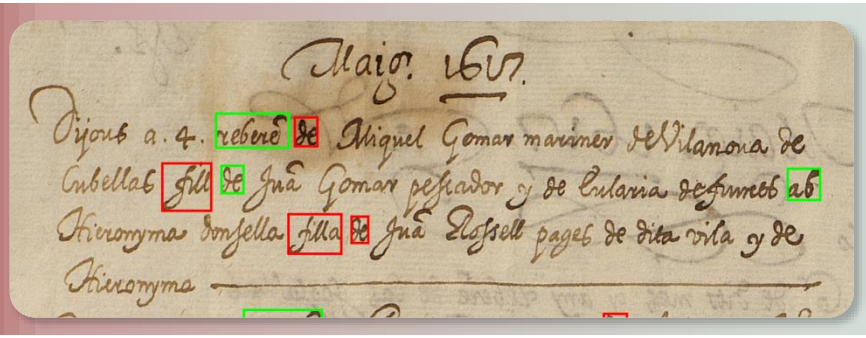


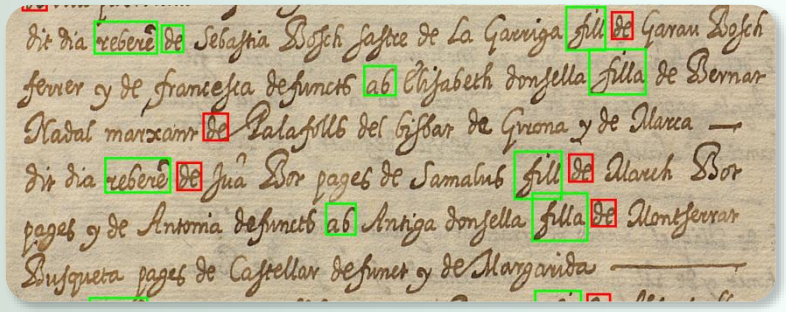
Context-aware Word Spotting

Key-words

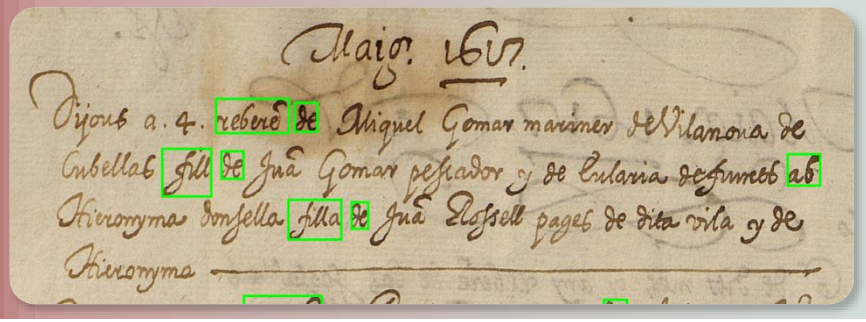


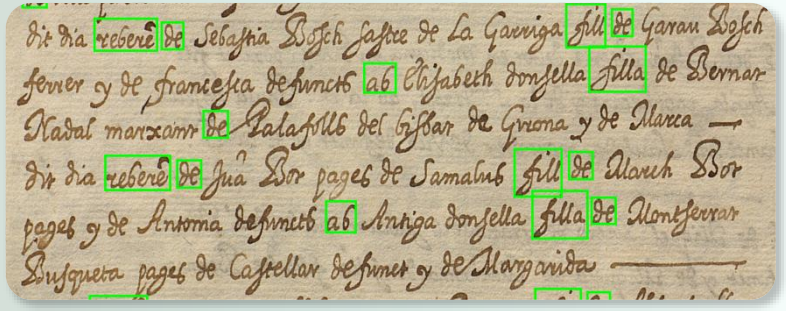
Word-Spotting





Contextual W-S

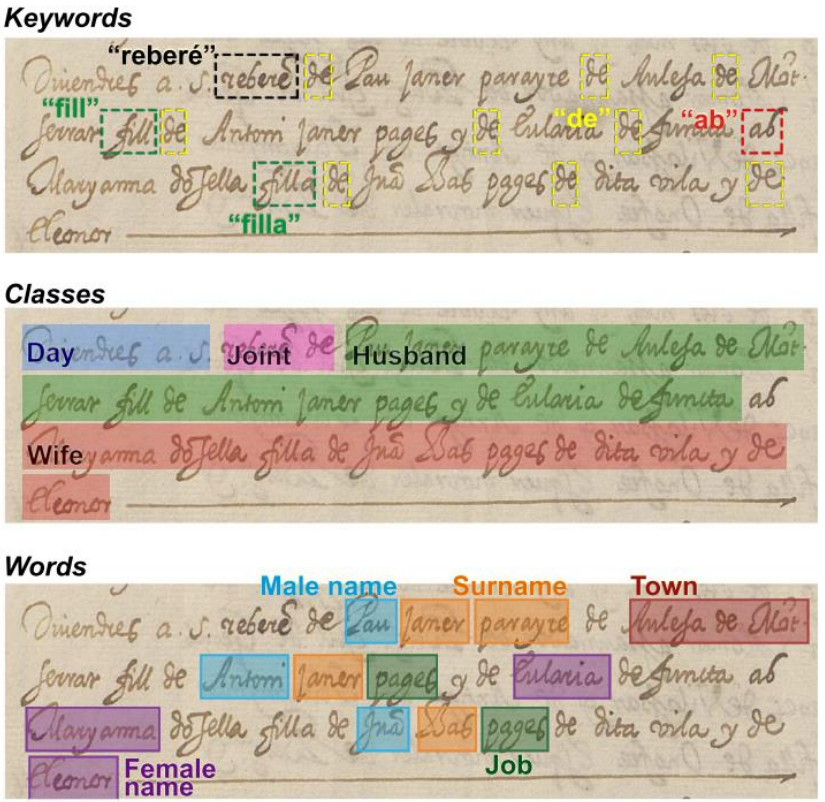




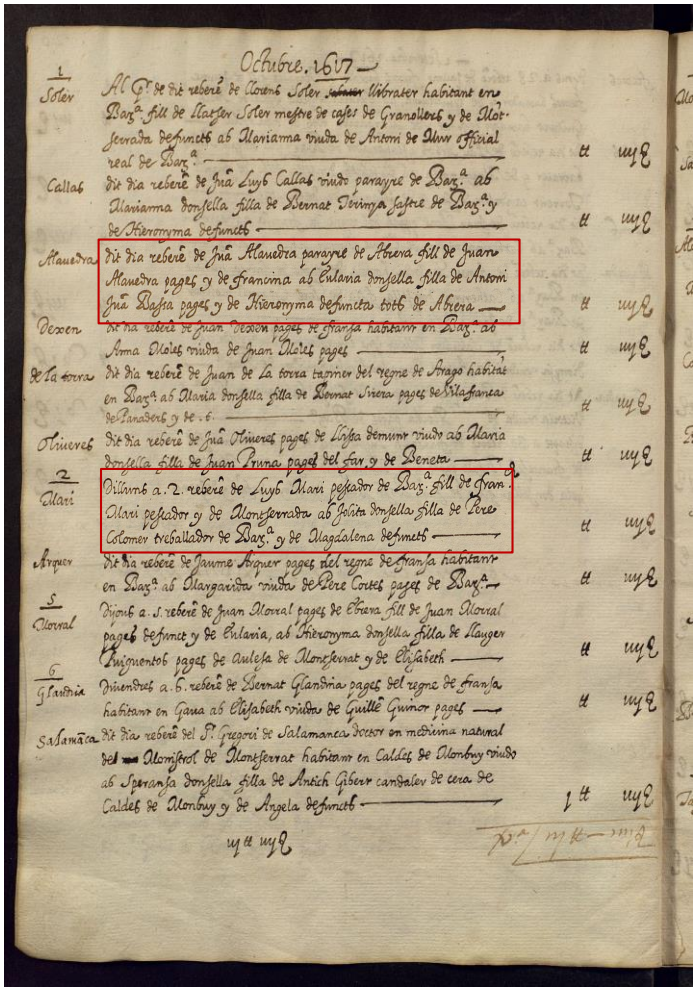
Context-aware Word Spotting

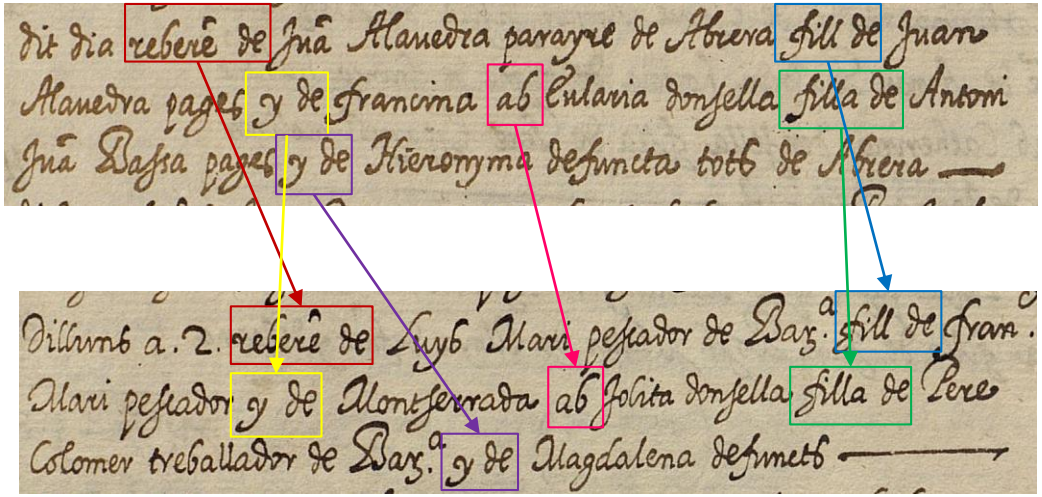
- Applied in documents that presents a **repetitive structure** in the words with text lines along pages.

- Objective: **Word-Spotting using structural information**
  - Discover the words that can be used as **keywords**.
  - Classify the rest of words by **Classes**.
  - Structural **Word** Spotting: find words using the example/string and its class.
  - Speed up the transcription

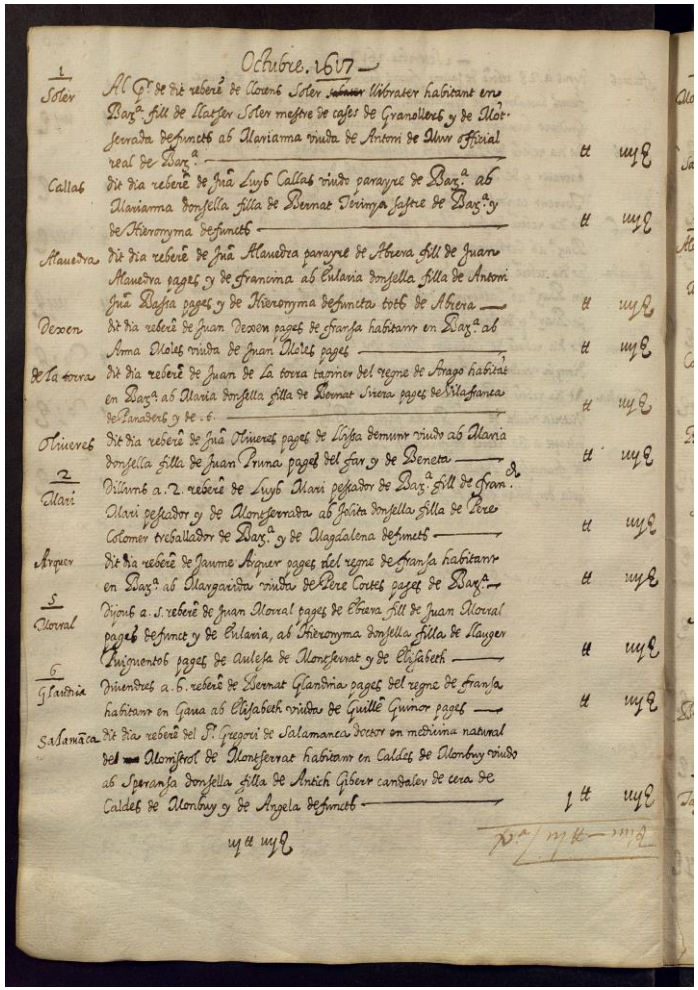


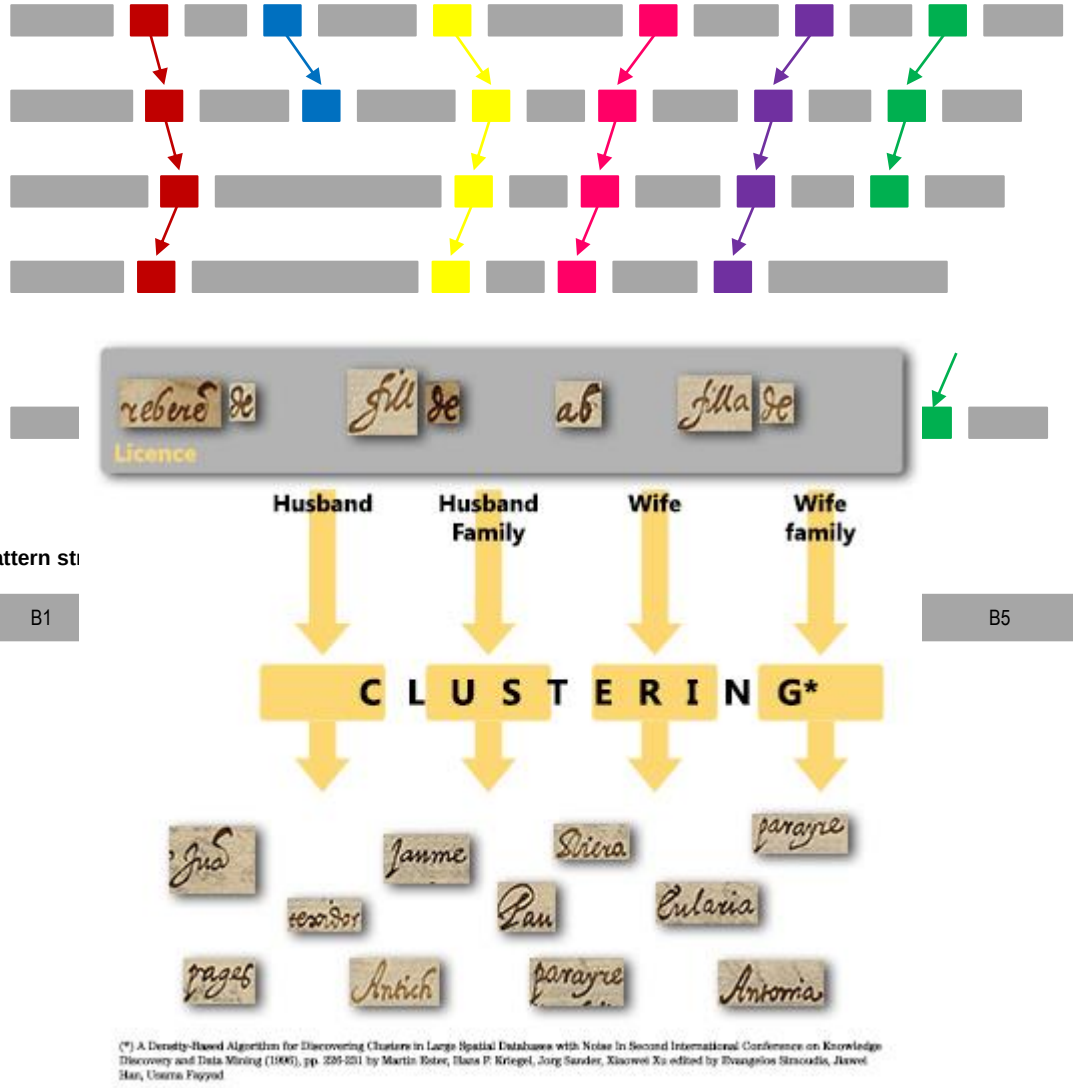
Alignment of records





Extraction of key and frequent words





(*) A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise In Second International Conference on Knowledge Discovery and Data Mining (1996), pp. 226-231 by Martin Ester, Hans P. Kriegel, Jorg Sander, Xiaowei Xu edited by Evangelos Simoudis, Jiawei Han, Usama Fayyad

CONCLUSIONS

Conclusions

- **Segmentation-based** *vs.* **Segmentation-free** methods
  - Accuracy of word segmentation methods?

- **Learning-free** *vs.* **Learning-based** methods (closer to recognition)
  - Do we have enough transcribed data?

- **Query-by-example** *vs.* **Query-by-class** *vs.* **Query-by-string**
  - Can we learn the appearance of words?

- **Context aware Word Spotting**
  - Discovers the contextual information (structure).
  - Tool for semi-automatic transcription (most frequent words)
  - Context can improve the methods (controlled conditions)

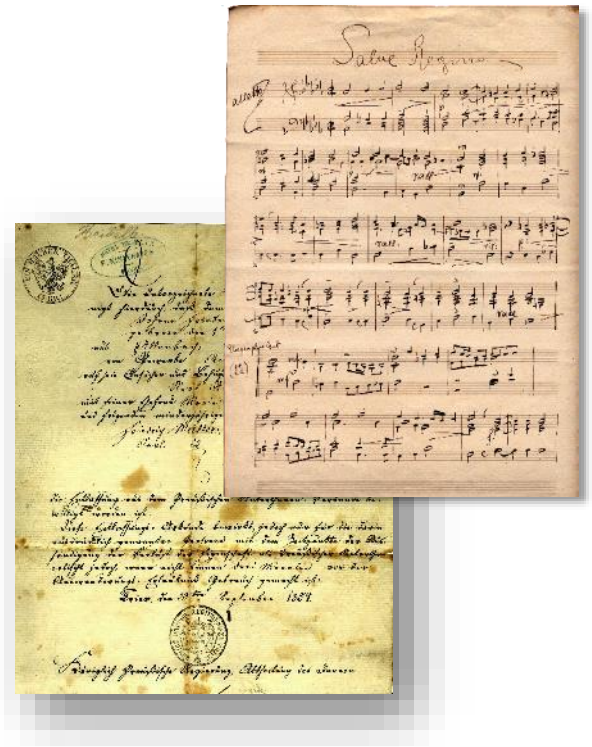
Achievements

Such scenarios are kind of solved…

- Single writter + Query-by-example + presegmented small collections
- Multiple writters + large amount of training data
- Query-by-string + large amount of training data

Challenges

- Having multiple writers in query-by-example scenarios
- Query-by-string when few or no training data is available
- Segmentation-free methods
- The use of context
- Scalability of the methods
- Non-latin scripts / shapes / signatures?



- Use of relevance feedback ?
  - Role of the User?

Centre de Visió per Computador

Obrigada!

July 2015

Alicia Fornés
afornes@cvc.uab.es

VISUM Summer School

VISion Understanding and Machine intelligence