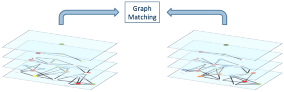
Error-tolerant coarse-to-fine matching model for hierarchical graphs

Pau Riba   Josep Lladós   Alicia Fornés

Computer Vision Center, Univ. Autònoma de Barcelona

Graph-based Representations, May 17th, 2017.



Outline

Introduction

Hierarchical Attributed Graph Representation

- Hierarchy construction
- Hierarchy by community detection
- Splitting of articulation points

Error-tolerant hierarchical matching

Experiments

- Datasets
- Results

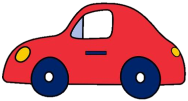
Conclusion and Future Work

Introduction

Introduction

Motivation

1. Increasing relevance in **visual object recognition and retrieval**, beyond classical pure appearance-based approaches.



Introduction

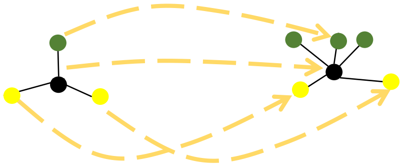
Motivation

1. Increasing relevance in **visual object recognition and retrieval**, beyond classical pure appearance-based approaches.

Introduction

Motivation

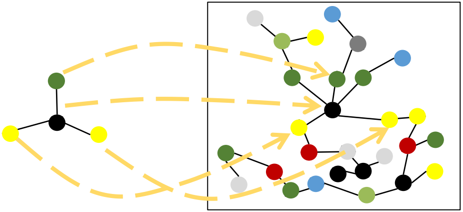
1. Increasing relevance in **visual object recognition and retrieval**, beyond classical pure appearance-based approaches.
2. Visual object detection using graphs involves an **inexact subgraph matching formulation**.



Introduction

Motivation

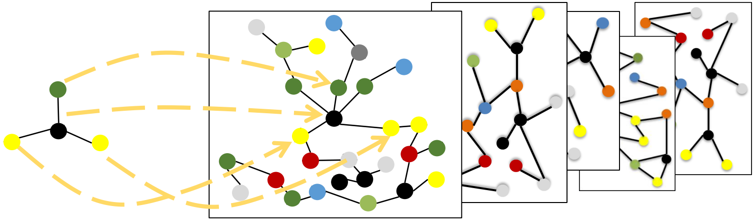
1. Increasing relevance in **visual object recognition and retrieval**, beyond classical pure appearance-based approaches.
2. Visual object detection using graphs involves an **inexact subgraph matching formulation**.



Introduction

Motivation

1. Increasing relevance in **visual object recognition and retrieval**, beyond classical pure appearance-based approaches.
2. Visual object detection using graphs involves an **inexact subgraph matching formulation**.
3. It is unavoidable in **large scale retrieval** (i.e. subgraph matching).



Introduction

Idea

Hypothesis

Hierarchical graph-based representations codify information at different detail levels. Therefore, a more efficient retrieval can be achieved.

Objective

To design a hierarchical graph representation and exploit its properties in matching time.

Hierarchical Attributed Graph Representation

Attributed graph

- $G = (V, E, \mathrm{L_V}, \mathrm{L_E})$.
- V is the set of nodes;
- $\mathrm{L_V} : V \rightarrow \Sigma_V \times A_V^k$ labelling function for nodes;
  - Σ_V set of symbolic labels for vertices;
  - A_V set of attributes for vertices
  - $k \in \mathbb{N}$;
- $E \subseteq V \times V$ is the set of edges;
- $\mathrm{L_E} : E \rightarrow \Sigma_E \times A_E^l$ labelling function for edges.
  - Σ_E set of symbolic labels for edges;
  - A_E set of attributes for edges
  - $l \in \mathbb{N}$;

Hierarchical graph construction

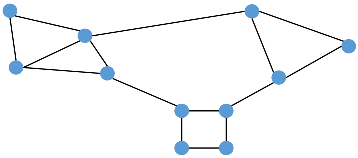
1. New type of edges: **Hierarchical edges**.
2. Two functions are defined:
   - **Contraction:** Creates node clusters. Cluster to Node.
   - **Embedding:** Vector representing a subgraph.
3. **Graph clustering/Community detection** as contraction function.

Contraction function

1. **Girvan-Newman based** algorithm.
   - Used for community detection.
   - Based on **betweenness centrality**(BC).
   - Smallest clusters without single node.

Contraction function

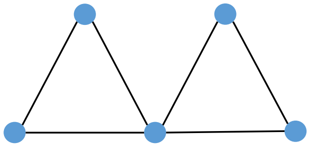
1. **Girvan-Newman based** algorithm.
   - Used for community detection.
   - Based on **betweenness centrality**(BC).
   - Smallest clusters without single node.



1. BC for all edges.
2. Remove edge with highest BC.
3. Recalculate BC.
4. GOTO 2 until no edge
5. **OUTPUT:** Dendrogram.

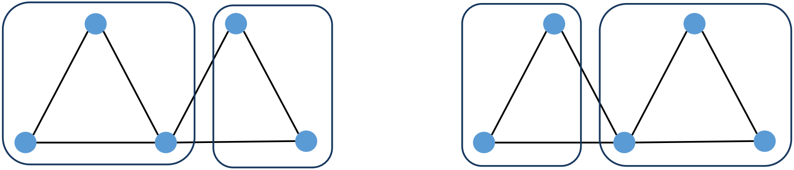
Contraction function

1. **Girvan-Newman based** algorithm.
   - Used for community detection.
   - Based on **betweenness centrality**(BC).
   - Smallest clusters without single node.
2. **Splitting of articulation points**



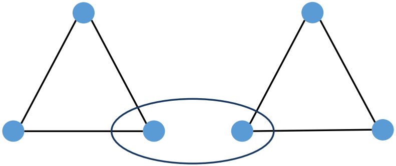
Contraction function

1. **Girvan-Newman based** algorithm.
   - Used for community detection.
   - Based on **betweenness centrality**(BC).
   - Smallest clusters without single node.
2. **Splitting of articulation points**



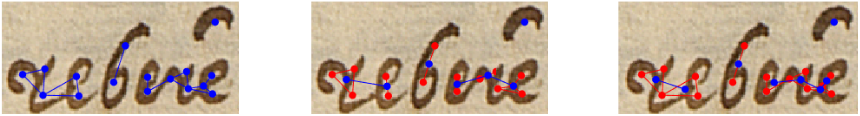
Contraction function

1. **Girvan-Newman based** algorithm.
   - Used for community detection.
   - Based on **betweenness centrality**(BC).
   - Smallest clusters without single node.
2. **Splitting of articulation points**



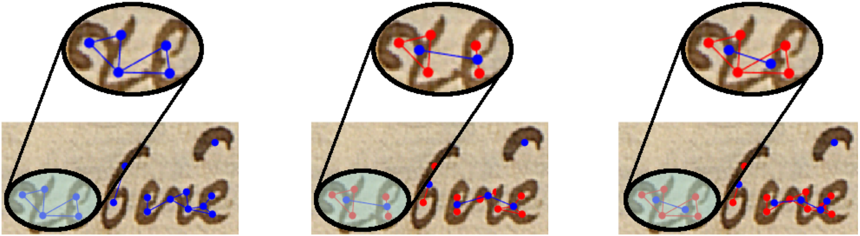
Contraction function

1. **Girvan-Newman based** algorithm.
   - Used for community detection.
   - Based on **betweenness centrality**(BC).
   - Smallest clusters without single node.
2. **Splitting of articulation points**

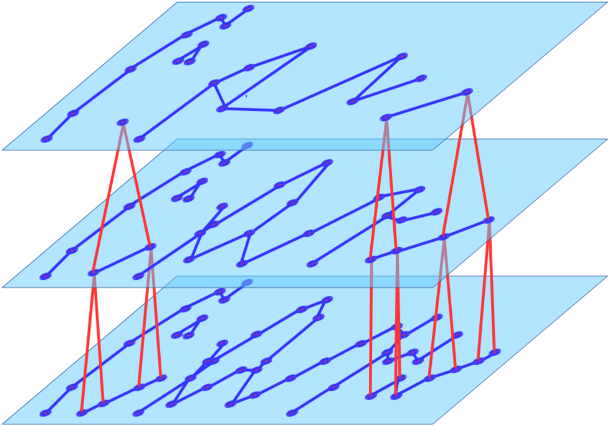


Contraction function

1. **Girvan-Newman based** algorithm.
   - Used for community detection.
   - Based on **betweenness centrality**(BC).
   - Smallest clusters without single node.
2. **Splitting of articulation points**



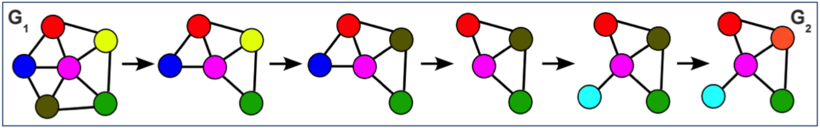
Hierarchical graph



Error-tolerant hierarchical matching

Graph-edit Distance

1. Idea of **String Edit Distance**.
2. Minimum cost of operations to transform one graph into the other.
   - **Insertion, deletion and substitutions**.
3. **Bipartite Graph Matching**: sub-optimal approximation (local edge structure).

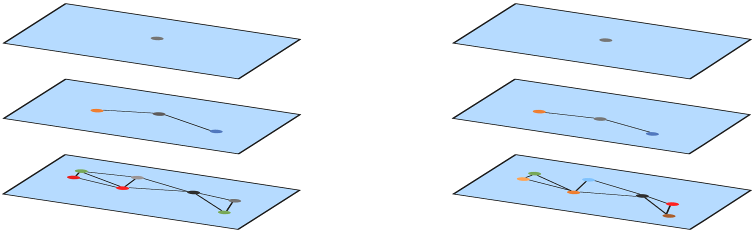


[1] K.Riesen, H.Bunke, Approximate graph edit distance computation by means of bipartite graph matching, Image and Vision Computing, 2009.

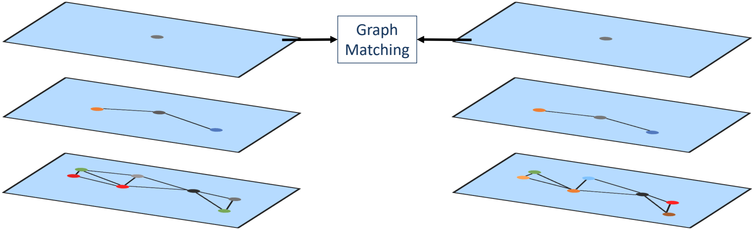
Coarse-to-fine

Coarse-to-fine

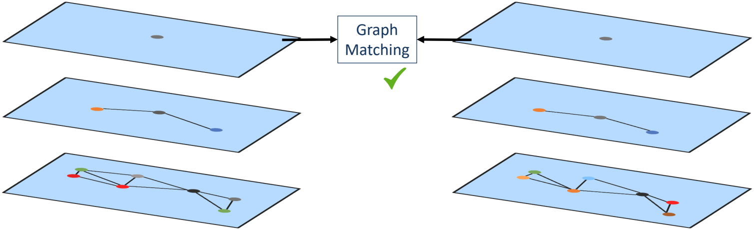
Coarse-to-fine



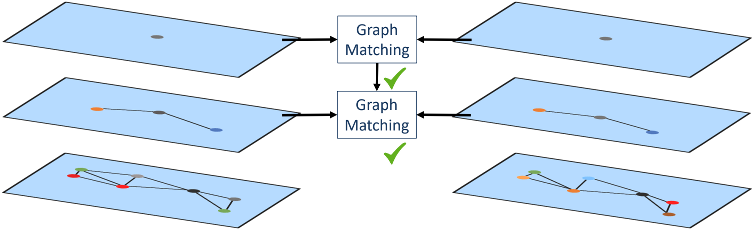
Coarse-to-fine



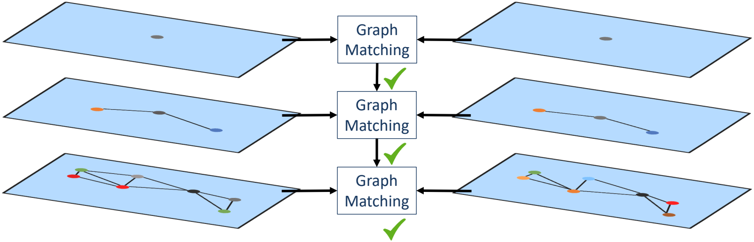
Coarse-to-fine



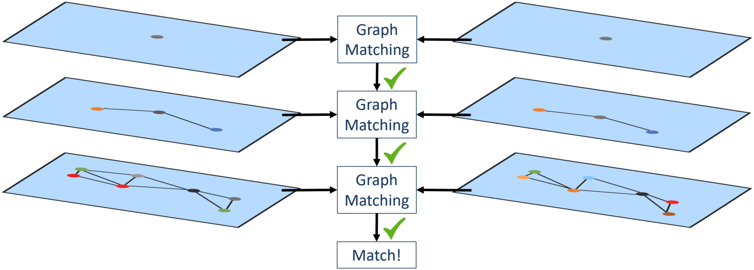
Coarse-to-fine



Coarse-to-fine



Coarse-to-fine



Experiments

Datasets

Object Classification

COIL-100

- 100 objects; 72 poses.
- 15 classes.
- 360, 75 and 150 (train, val, test).

ODBK

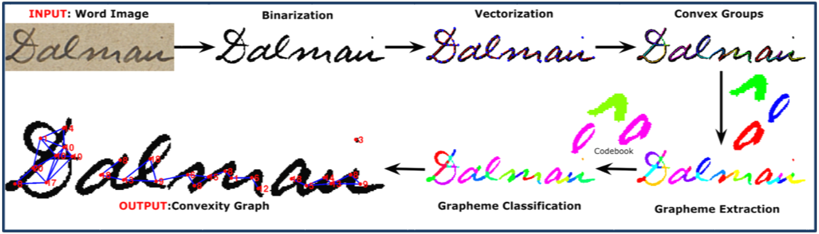
- 209 3D objects; 14 views.
- 50 classes.
- 300, 150 and 150 (train, val, test).

- Harris corner detector
- Delaunay triangulation.
- **Embedding**: Mean Morgan Index of length 1 and 2.

Datasets

Barcelona Historical Handwritten Marriages database

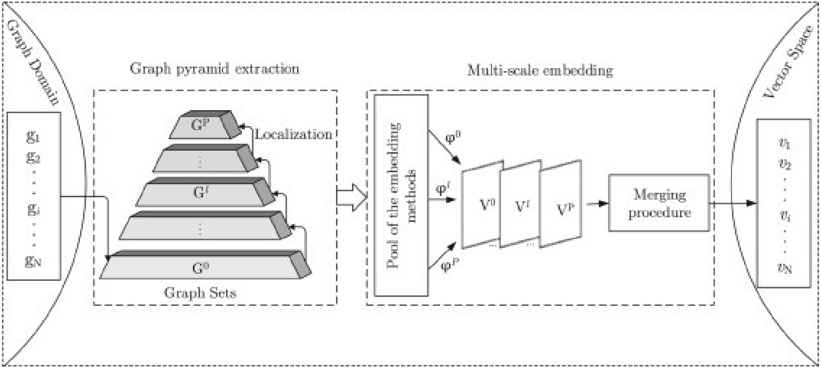
- 174 Handwritten pages.
- Cropped words.
- Graph based on convexities (graphemes).
- Codebook as labels (BSM).
- **Embedding**: Histogram of node codebooks.



[2] P.Riba, A.Fornés, J.Lladós. Handwritten word spotting by inexact matching of grapheme graph, International Conference on Document Analysis and Recognition, 2015.

Results

Object classification



(Image credit to S.F.Mousavi.)

[3] S.F.Mousavi, M.Safayani, A.Mirzaei, H.Bahonar. Hierarchical graph embedding in vector space by graph pyramid, Pattern Recognition, 2017.

Results
Object classification

Comparison:

COIL database						
	Thresh.	K-NN(%) 1	3	5	AC[1](%)	t(s)
Original	-	100.00	100.00	98.00	-	2010
1st abst.	-	72.67	74.67	72.67	-	**167**
2nd abst.	-	38.00	39.33	44.67	-	**13**
Original	-	100.00	97.00	90.00	Mousavi *et al.* [3]	
1st abst.	-	98.17	94.83	88.83		
2nd abst.	-	87.00	81.67	78.17		

ODBK database						
	Thresh.	K-NN(%) 1	3	5	AC[1](%)	t(s)
Original	-	79.33	76.00	74.00	-	34959
1st abst.	-	58.67	58.00	54.67	-	**1954**
2nd abst.	-	42.00	41.33	46.00	-	**141**
Original	-	66.67	65.33	63.33	Mousavi *et al.* [3]	
1st abst.	-	66.67	62.67	62.00		
2nd abst.	-	60.00	55.33	53.33		

[3] S.F.Mousavi, M.Safayani, A.Mirzaei, H.Bahonar. Hierarchical graph embedding in vector space by graph pyramid, Pattern Recognition, 2017.

Results

Object classification

COIL database						
	Thresh.	K-NN(%) 1	3	5	AC[1](%)	t(s)
Original	-	100.00	100.00	98.00	-	2010
1st abst.	-	72.67	74.67	72.67	-	**167**
2nd abst.	-	38.00	39.33	44.67	-	**13**

ODBK database						
	Thresh.	K-NN(%) 1	3	5	AC[1](%)	t(s)
Original	-	79.33	76.00	74.00	-	34959
1st abst.	-	58.67	58.00	54.67	-	**1954**
2nd abst.	-	42.00	41.33	46.00	-	**141**

Results

Object classification

COIL database						
	Thresh.	K-NN(%) 1	3	5	AC[1](%)	t(s)
Original	-	100.00	100.00	98.00	-	2010
1st abst.	-	72.67	74.67	72.67	-	**167**
2nd abst.	-	38.00	39.33	44.67	-	**13**
1st abst.	0.1982	98.00	97.33	93.33	67.37	977
	0.1680	90.00	89.33	82.67	**95.41**	289
2nd abst.	0.2153	100.00	99.33	96.67	33.68	1444
	0.1895	97.33	94.67	93.33	58.99	937

ODBK database						
	Thresh.	K-NN(%) 1	3	5	AC[1](%)	t(s)
Original	-	79.33	76.00	74.00	-	34959
1st abst.	-	58.67	58.00	54.67	-	**1954**
2nd abst.	-	42.00	41.33	46.00	-	**141**
1st abst.	0.2396	79.33	76.00	74.00	48.18	22501
	0.2130	78.67	75.33	72.00	**79.10**	10496
2nd abst.	0.2973	78.67	74.67	72.67	33.76	26111
	0.2573	76.67	71.33	68.67	68.49	12228

Results

Object classification

COIL database						
	Thresh.	K-NN(%) 1	3	5	AC^1(%)	t(s)
Original	-	100.00	100.00	98.00	-	2010
1st abst.	-	72.67	74.67	72.67	-	**167**
2nd abst.	-	38.00	39.33	44.67	-	**13**
1st abst.	0.1982	98.00	97.33	93.33	67.37	977
	0.1680	90.00	89.33	82.67	**95.41**	289
2nd abst.	0.2153	100.00	99.33	96.67	33.68	1444
	0.1895	97.33	94.67	93.33	58.99	937
1st abst. 2nd abst.	0.1982 0.2153	98.67	98.00	92.67	**71.63**	**893**

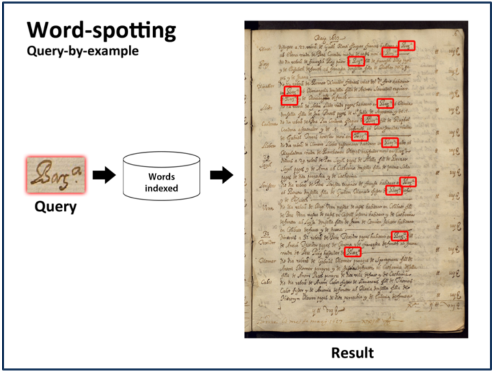
ODBK database						
	Thresh.	K-NN(%) 1	3	5	AC^1(%)	t(s)
Original	-	79.33	76.00	74.00	-	34959
1st abst.	-	58.67	58.00	54.67	-	**1954**
2nd abst.	-	42.00	41.33	46.00	-	**141**
1st abst.	0.2396	79.33	76.00	74.00	48.18	22501
	0.2130	78.67	75.33	72.00	**79.10**	10496
2nd abst.	0.2973	78.67	74.67	72.67	33.76	26111
	0.2573	76.67	71.33	68.67	68.49	12228
1st abst. 2nd abst.	0.2130 0.2973	78.00	74.00	70.67	**79.23**	**10292**

Results

Word Spotting

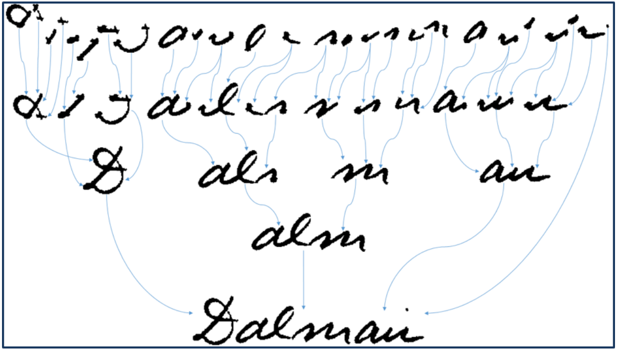
1. **Word spotting**: Retrieve all the instances of a query using visual object retrieval.

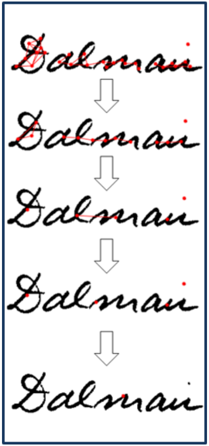
2. Shape dissimilarity functions avoiding an **explicit transcription**.



Results

Word Spotting

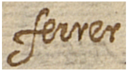




Results

Word Spotting

Query:

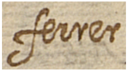


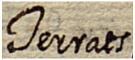
Original:

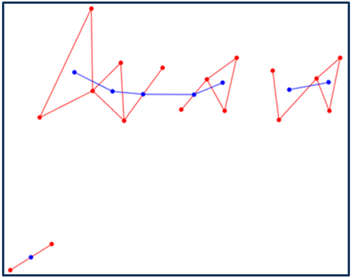
1st Abstract:

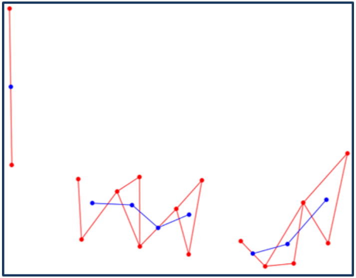
Results

Word Spotting









Results

Word Spotting

	mAP (%)	R (%)	SPC (%)	Time/query[1] (s)
Original	69.45	100.00	0.00	19.58
+abst. (t=0.30)	**68.27**	90.91	69.98	**12.46**
+abst. (t=0.25)	**61.71**	67.93	97.91	**3.94**
+[4] (t=0.20)	66.13	92.54	46.13	16.34
+[4] (t=0.30)	61.15	83.55	63.04	12.74

[4] P.Riba, J.Lladós, A.Fornés, A.Dutta, Large-scale graph indexing using binary embeddings of node contexts for information spotting in document image databases, Pattern Recognition Letters, 2017

[1] 1000 queries selected randomly against 13098 graphs.

Conclusion and Future Work

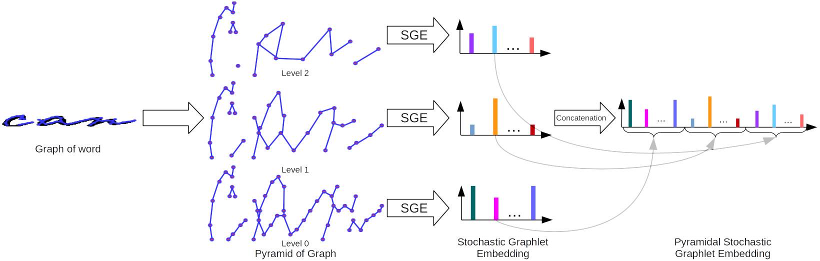
Final thoughts

Conclusions

1. **Graph based representations** suffer from the sensitivity to noise.
2. **Hierarchical graph representation** has been proved to codify abstract information increasing the representation power.
3. Graph matching can be speeded up by pruning comparisons at abstract levels.

Future work

- Pyramidal Stochastic Graphlet Embedding:



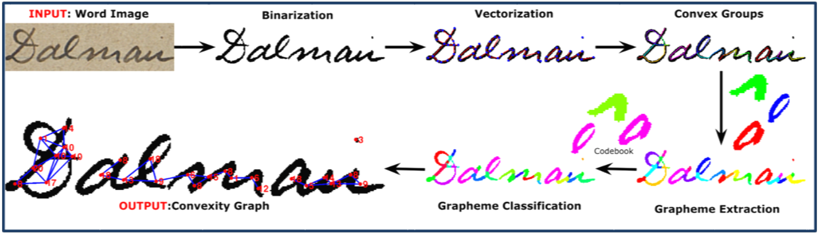
Submitted to ICDAR, 2017

- Matching algorithms using the whole hierarchy at once.

Thank you for your attention!

Pau Riba
Computer Vision Center
priba@cvc.uab.cat

Graph construction



Graph-based Word Spotting