

Pattern Spotting in historical document images

Sovann EN, Caroline Petitjean, Stéphane Nicolas,
Frédéric Jurie, Laurent Heutte

LITIS, University of Rouen, France



Outline

- Introduction
- Commons Pipeline
- Our works
 - Similarity Distance
 - Feature Extraction
 - Scalability
- Conclusion & future works

Introduction

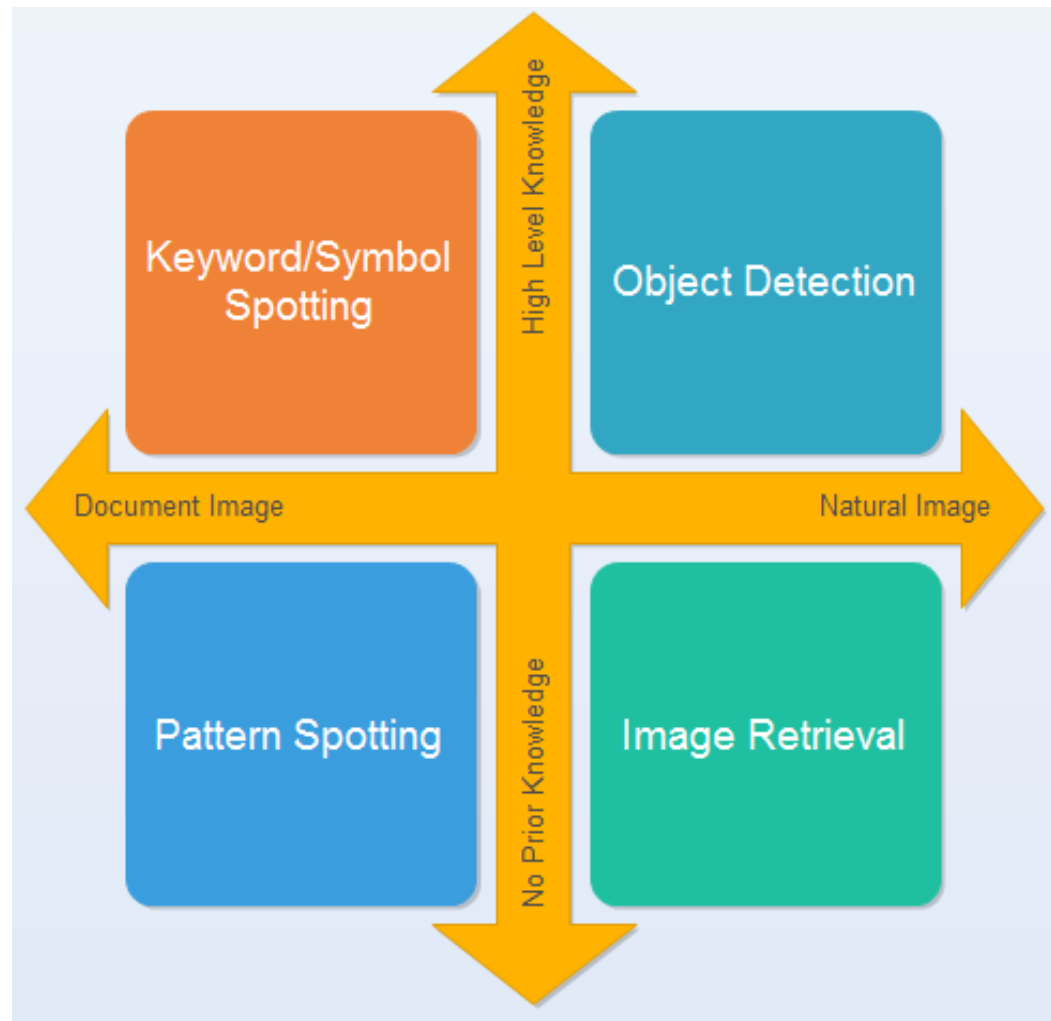
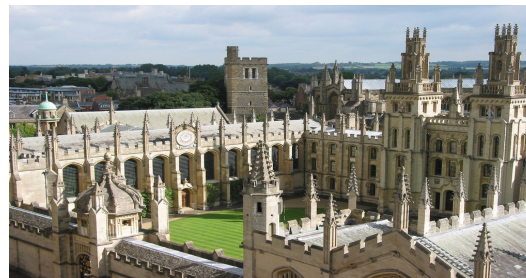
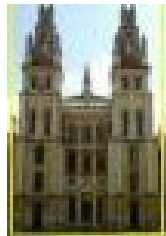


Image Retrieval

- Natural Image: high quality
- The query generally is big enough ($\sim 27\%$ of the image)
- No prior knowledge of the query nor the images
- The query can be any objects
- Nearest neighbor problem
- Challenges: changes in color, view point, scale, contrast, etc.



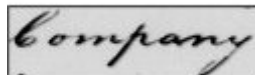
Object Detetion

- Natural Image: high quality
- The object is known prior to the detection
 - the model represents the object can be trained
- Classification problem
 - There are predefined classes of objects
- Challenges: variabilities of the objects

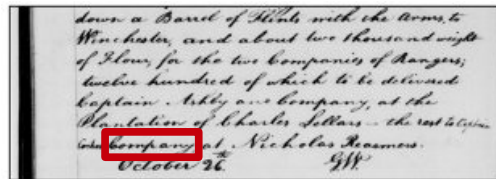


Keyword Spotting

- Document Image: low quality
- The query is not known prior to the search, but contains limited set of characters
- Words can be isolated (exhaustive sliding window can be avoided)
- Nearest neighbor approach
 - possibility to transform into a classification problem + NN
- Challenges: noise, writing style, changes in scale or background, etc.



company

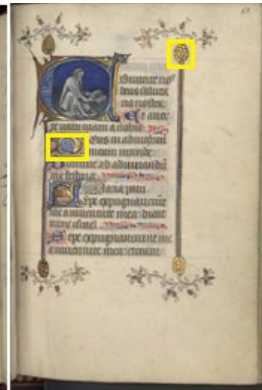


Pattern Spotting

- Document Image: low quality
- The query is not known prior to the search
 - the query size is small ($<2\%$ of the image size)
- Segmentation is hardly feasible
- Nearest neighbor approach
- Challenges: noise, changes in scale, background, and color, etc.



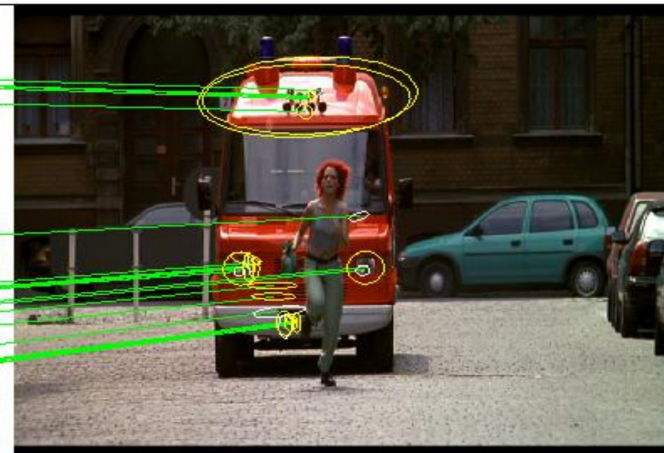
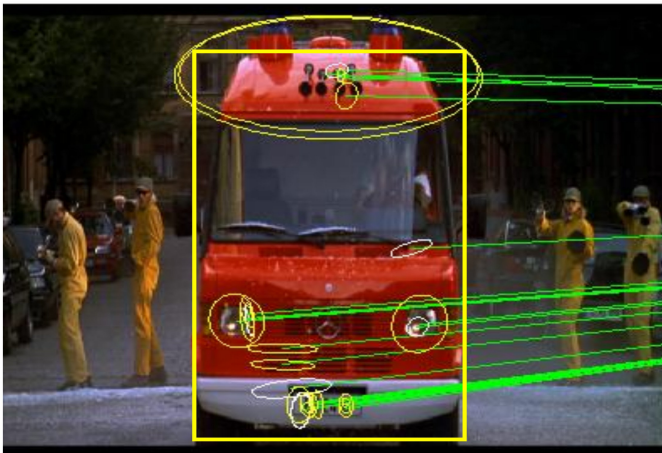
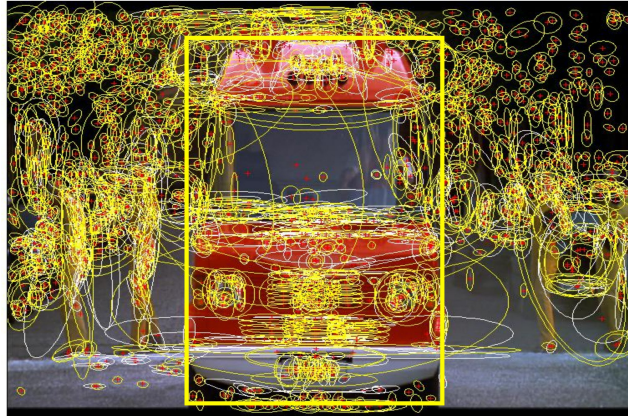
Pattern Spotting



Commons pipeline

- Image Retrieval System:
 - generally use BoVW (Bag of Visual Word)
 - VLAD, FV, GIST, HOG, Sparse coding ...
 - SIFT-like descriptor to characterize local patch
 - Need a codebook to quantize each SIFT descriptor
 - Use spatial consistency to localize the object
 - No sliding window
 - Spatial consistency can be approximated by: RANSAC-like algorithm

Commons pipeline

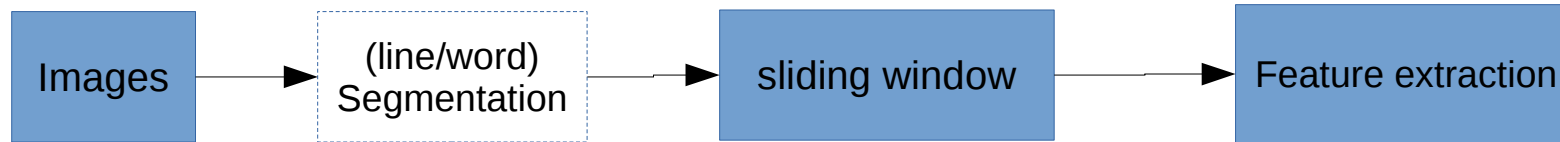


Commons pipeline

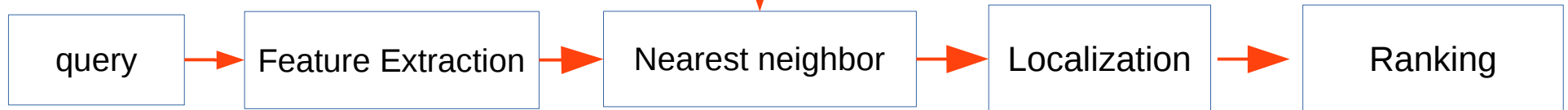
- Keyword spotting:
 - Dominated by BoVW (Bag of Visual Word)
 - HMM, Deep learning, ...
 - Use SIFT-like descriptor to characterize local patch
 - Need a codebook to quantize each SIFT descriptor
 - Use spatial consistency to localize the object
 - sliding window can be used
 - The similarity is done between the query and each subwindow

Commons pipeline

Offline processing



Online processing



Commons pipeline

- Preprocessing
- Feature Extraction
- Similarity Distance measure
- Postprocessing: localization

Our Works

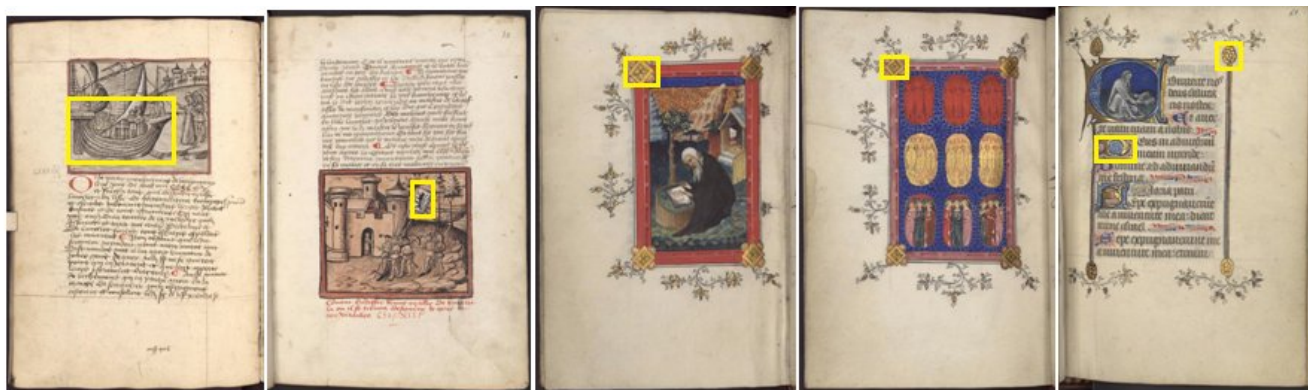
- Similarity Distance Measure: nearest neighbor approach
 - Simple, but will not cope well with different variabilities
 - turn NN problem into a classification problem
 - learn an adapted distance function for each query
- Keyword spotting is dominated by BoVW
 - Many recent feature extractions are proposed in CV
 - Would those features outperform the traditional BoVW?

Our Works

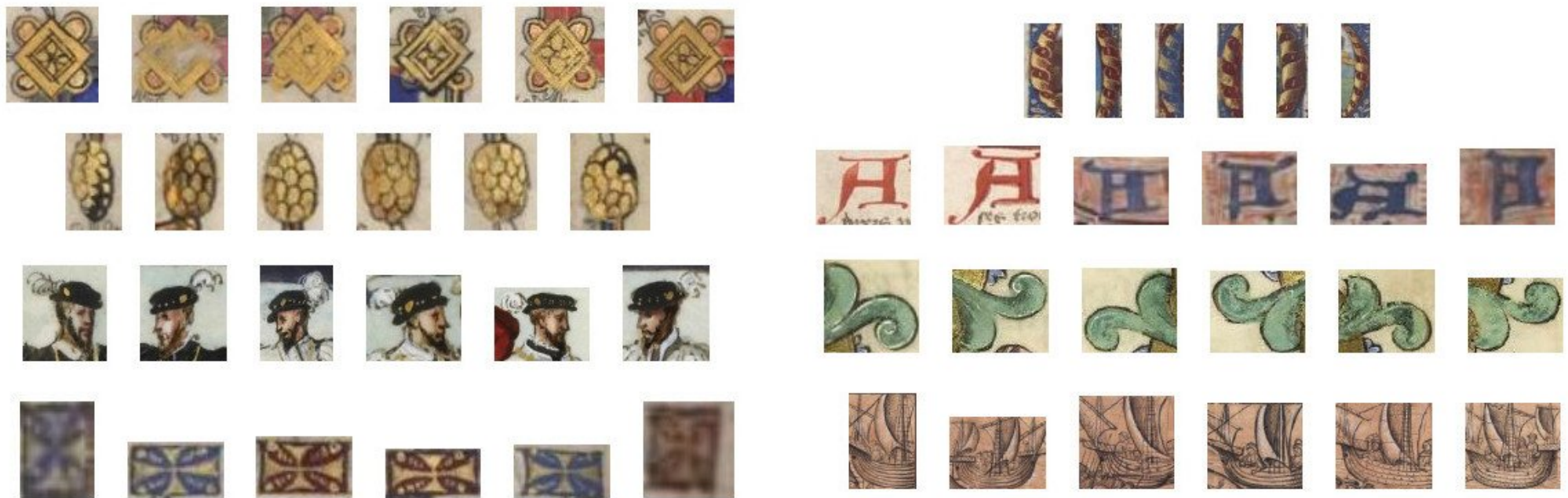
- Sliding window makes Pattern spotting become unscalable
 - an image of 1024×1024 needs 1M subwindows
 - Non-exhaustive search
 - reduce the number of subwindows

Dataset

- DocExplore (<http://www.docexplore.eu/>) Dataset:
 - 1597 medieval manuscript images
 - 1094 queries with groundtruth annotations
 - varying size: 30*30 pixels to 600*600 pixels
 - variability includes: scale, color, contrast, noise, writing style

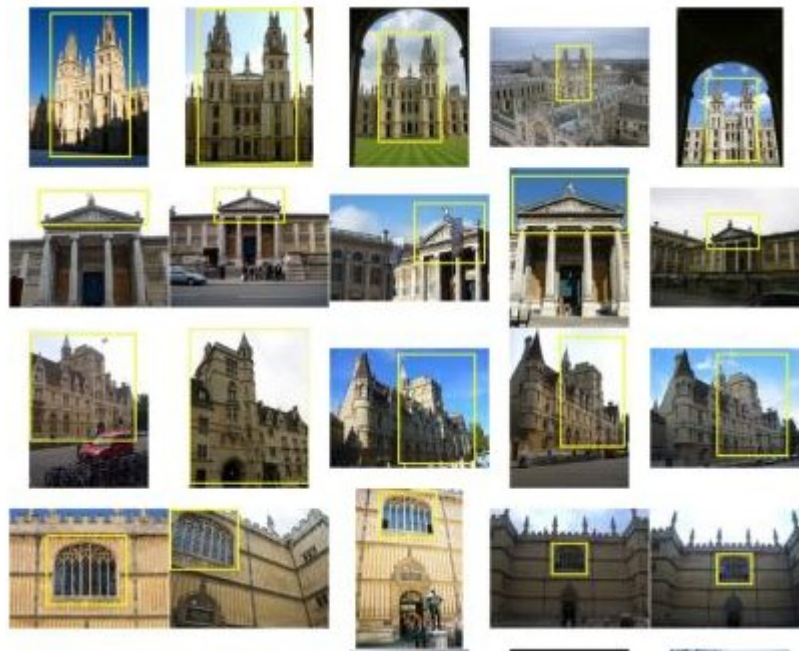


Dataset



Dataset

- Oxford dataset:
 - 5k images taken from Flickr
 - 55 queries with groundtruth data



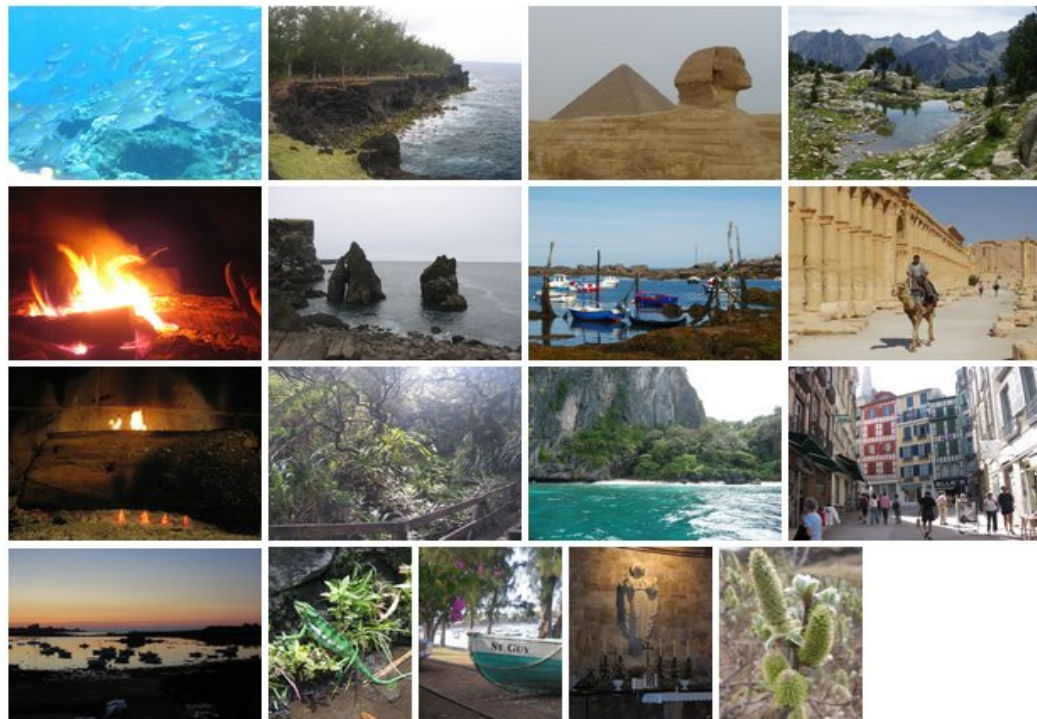
Dataset

- Paris dataset:
 - 6k images taken from Flickr
 - 55 queries with groundtruth data



Dataset

- Holiday dataset:
 - 1k images of landscape
 - 500 queries with groundtruth data



Adapted distance function

- Learn an adapted function for every single query
 - No training corpus → zero shot learning
- Assuming there are two distributions: relevant and irrelevant.
- X is positive (relevant) if $P(y=1|x) > P(y=0|x)$

$$P(y = k|x) = \frac{f(x; \mu_{y=k}, \Sigma_{y=k})P(y = k)}{\sum_{l \in \{0,1\}} f(x; \mu_{y=l}, \Sigma_{y=l})P(y = l)}$$

f is a multivariate normal distribution

→ find μ and Σ to build the model

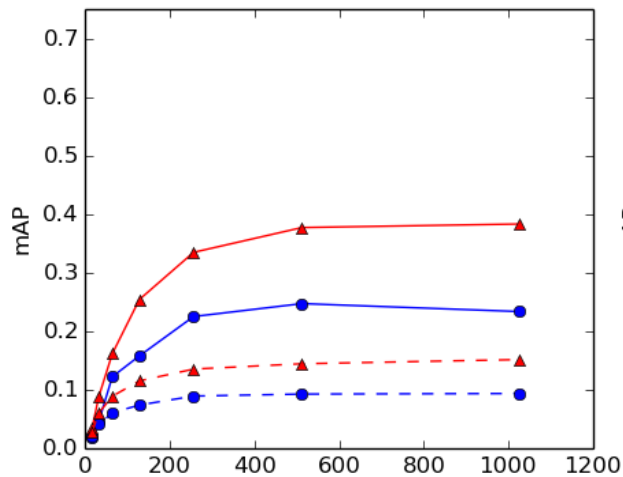
S. EN, F. Jurie, S. Nicolas, C. Petitjean and L. Heutte, Linear discriminant analysis for zero-shot learning image retrieval. VISAPP 2015.

Adapted distance function

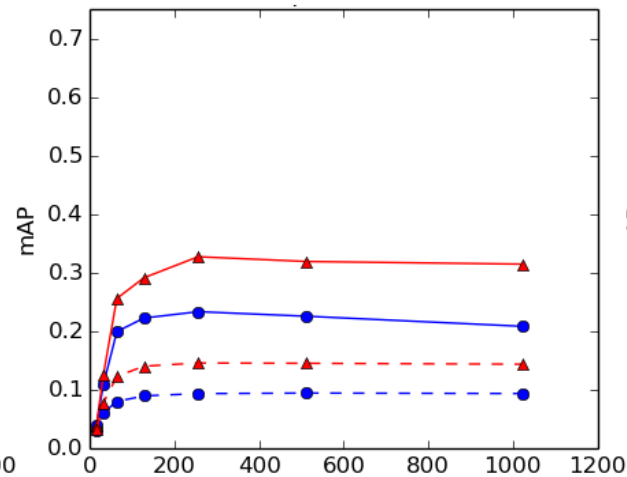
- **Irrelevant Class:** (μ_o, Σ_o) can be easily approximated using the dataset (true if it is big enough and contains only small amount of relevant image)
- **Relevant Class:**
 - μ_1 can be approximated by the query vector
 - As we do not have enough images to calculate
 - assume $\Sigma = \Sigma_1 = \Sigma_o$
- $P(y=1|x) > P(y=0|x) \Leftrightarrow x^t \cdot \Sigma^{(-1)} (\mu_1 - \mu_o) > 0$
- the cost of learning a new model for every single query ~ 0
- the ranking costs as much as a simple dot product distance

Adapted distance function

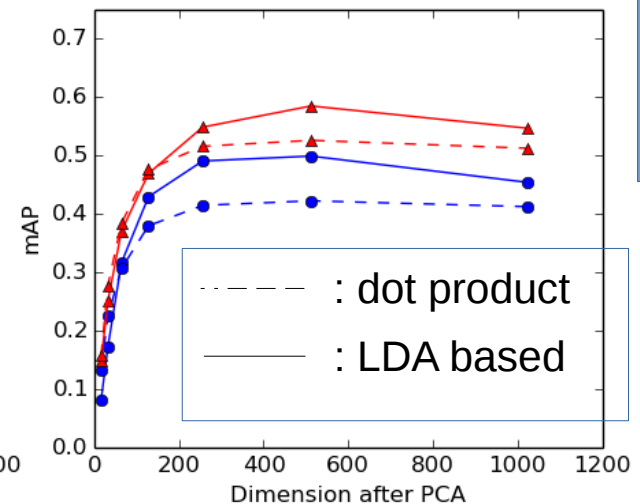
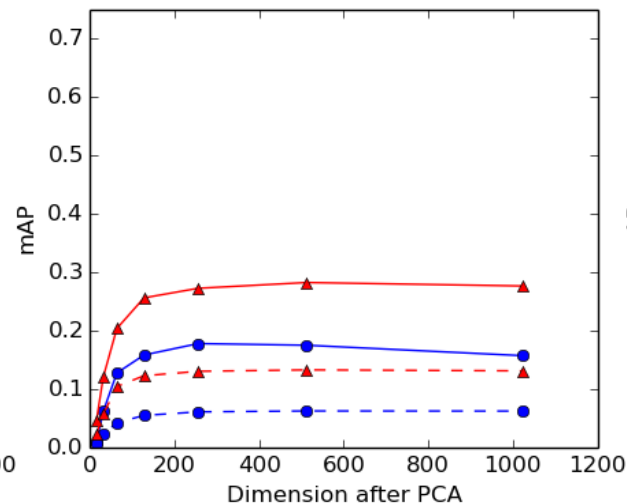
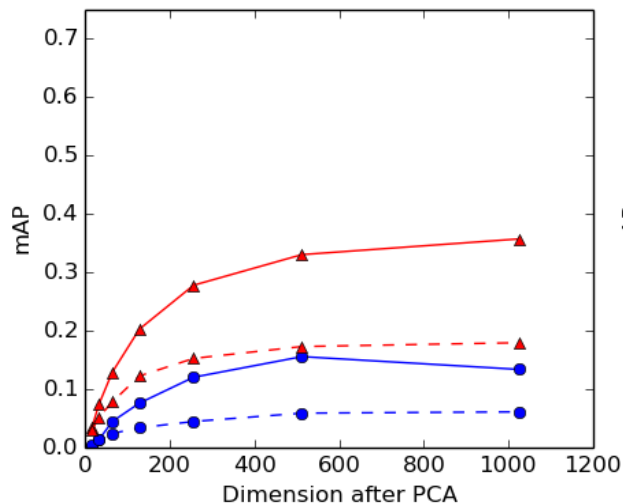
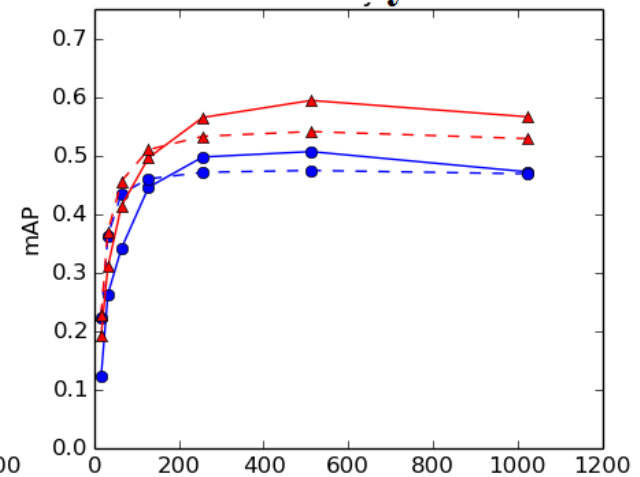
Oxford1M



Paris1M



Holiday1M



K=16(blue)

K=64(red)

--- : dot product
— : LDA based

Feature Extraction

- Quantization is a lossy process: we can not get back to the original descriptor
 - Vocabulary size: the larger the better, but storage & computational cost ?
 - Image retrieval follows NN search: large vocabulary, 1M codeword
 - Storage intracable with 1 million image and 1 million codeword
- VLAD and Fisher Vector: dense, compact, easy to compress ...

Feature Extraction

- Bag of Visual Word
 - count the number of occurrence of the codeword
- Vector of Locally Aggregated Descriptor
 - Accumulate the difference between descriptors and its closest cluster center
 - Aggregate the accumulation
- Fisher Vector
 - an aggregated version (like VLAD) of local descriptors
 - use GMM model (vs k-means clustering): follow soft assignment principle
 - use gradient function (derived from Fisher Kernel) on each components (posterior probability, means and covariance) for aggregation

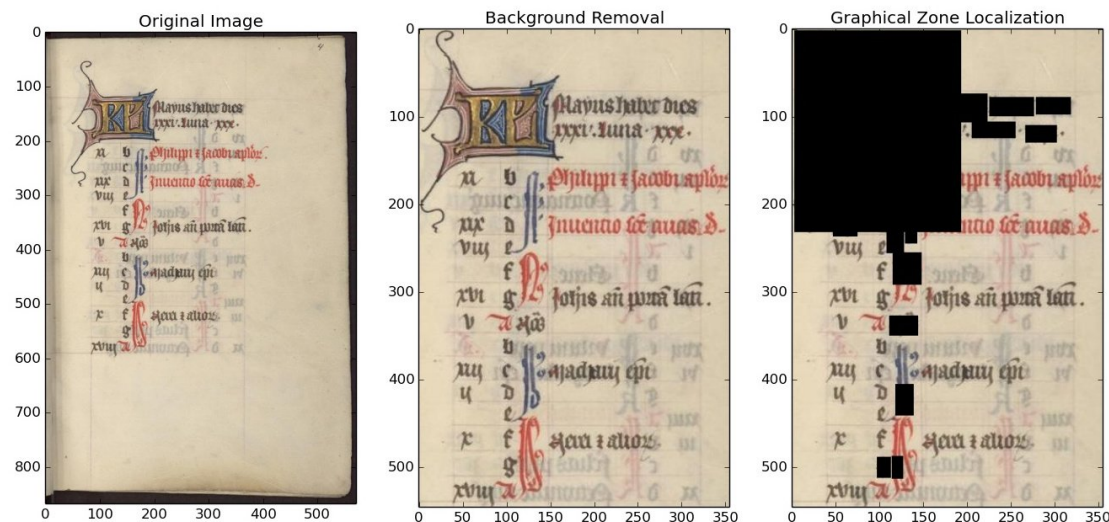
Feature Extraction: Results

- Experimental results running on docexplore dataset

K	BoVW		K	VLAD		FV	
	mAP	Memory		mAP	Memory	mAP	Memory
500	0.195	4GB					
1k	0.203	8GB	4	0.251	2GB	0.216	2GB
5k	0.213	40GB	16	0.319	8GB	0.261	8GB
10k	0.212	80GB	64	0.295	32GB	0.348	32GB

Scalability

- Reduce the #subwindows:
 - Background removal
 - Estimate where the object is (reject textual zone)
 - Scribo Module: cover 72% of the groundtruth bbox



Scalability

- Compress Feature:
 - Product Quantization

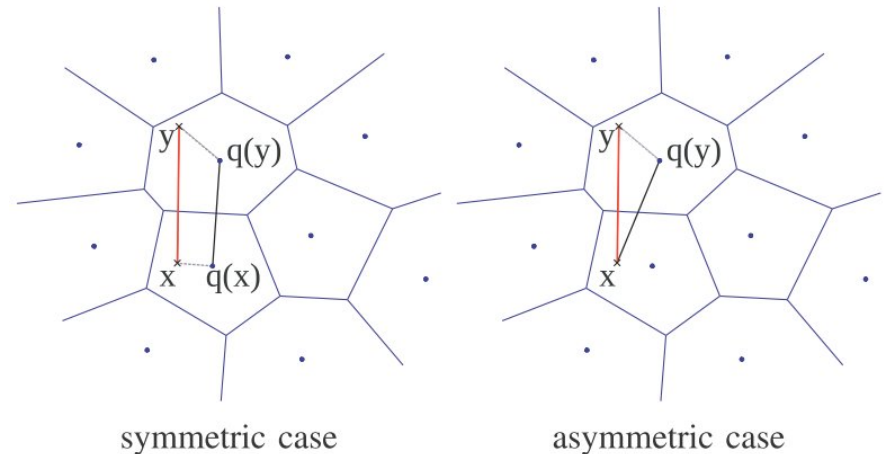
32D feature vector:



4D code words:



- Non-exhaustive search
 - Inverted File Structure (IVF)
- Approximate distance computation
 - Asymmetric distance computation



Scalability

- Results comparison:
 - BR: use only background removal
 - SB: use Scribo module to estimate the graphical zone

K		500	1k	5k	10k	#subw.	M/T
BR		0.189	0.190	0.186	0.171	14.5M	58/3.5
SB		0.195	0.204	0.212	0.212	2.1M	8.4/1.8

Scalability

K	D	D'=512		D'=256		D'=128	
		No	PQ,IVF	No	PQ,IVF	No	PQ,IVF
BoVW							
500	0.195	-	-	0.198	0.157	0.193	0.168
1k	0.203	0.195	0.138	0.191	0.154	0.189	0.164
5k	0.213	0.181	0.100	0.177	0.145	0.171	0.149
10k	0.212	0.172	0.096	0.170	0.137	0.165	0.138
VLAD							
4	0.251	-	-	0.269	0.224	0.261	0.218
16	0.319	0.359	0.282	0.353	0.306	0.340	0.286
64	0.295	0.357	0.305	0.346	0.306	0.332	0.288
FV							
4	0.216	-	-	0.243	0.186	0.240	0.190
16	0.261	0.287	0.243	0.283	0.242	0.275	0.229
64	0.348	0.374	0.342	0.364	0.330	0.350	0.303

Keyword spotting

- With very few modifications
 - ICDAR 2015 keyword spotting competition

Method	mAP	# subwindows	Representation	Dimension	spatial information
Baseline	0.10				No
PGR	0.27		BoVW	56k	SPM
CVC	0.08		BoVW	172k	SPM
BoVW5k	0.04	980k	BoVW	5k	No
VLAD16	0.10	980k	VLAD	1k	No
FV16	0.09	900k	FV	1k	No

Retrieval Efficiency

- Timing and Memory requirement
 - BoVW: codebook size of 1k
 - VLAD, FV: codebook size of 16
 - BR = Background removal, SB: Scribo module

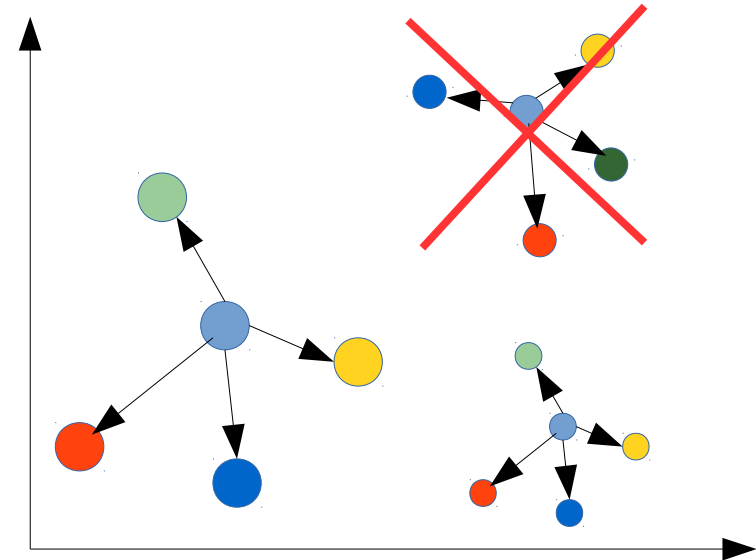
Method	#subw.	Time	RAM
BR	14.5M	3.5m	58GB
SB, D	2.1M	1.8m	8.4GB
SB, D'=128	2.1M	7.2s	1.1GB
SB, D'=128+PQ	2.1M	5.2s	33.5MB
SB, D'=128+IVFPQ	2.1M	0.51s	33.5MB

Conclusion

- Number of subwindows is reduced by 7 times using Scribo Module
- Feature compression achieved with 1:32 ratio with little decrease in performance
- IVF helps to avoid exhaustive distance computation without lose in performance
- Retrieval is done with less than 1 second on 14M subwindows
- A unified benchmarking of 3 feature extractions:
 - VLAD and FV could be good alternative representation
 - Performance increases by almost 2 times with VLAD and FV

Future works

- Possible directions:
 - Better graphical part detection system
 - reduce #nb of sub-windows
 - Increase the precision
 - Objectness: cover 90% of the bbox
 - Spatial verification reinforcement
 - Identify the discriminant regions on the image
 - Use its position to matches with the discriminant regions on the retrieved image
 - Propose a final ranking function based on spatial information (DPM, Metric learning etc)
 - Color feature or deep feature



Thanks for your attention !

Publications:

1. S. EN, F. Jurie, S. Nicolas, C. Petitjean and L. Heutte, Linear discriminant analysis for zero-shot learning image retrieval. VISAPP 2015. Accepted.
2. S. EN, C. Petitjean, S. Nicolas and L. Heutte, Segmentation-free pattern spotting in historical document images. ICDAR 2015. Accepted.
3. S. EN, C. Petitjean, S. Nicolas and L. Heutte, Détection des motifs graphiques dans les images de documents anciens. Colloque de GRETSI 2015. Accepted.
4. S. EN, C. Petitjean, S. Nicolas and L. Heutte, A scalable pattern spotting system for historical documents. Pattern Recognition Journal. Being reviewed.