Kernel Square-Loss Exemplar Machine for Image Retrieval

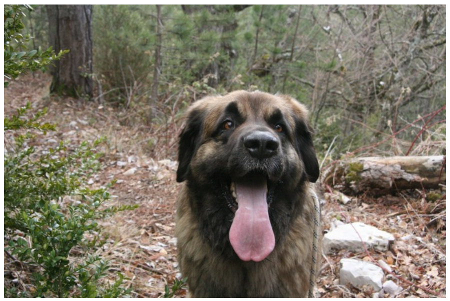
Rafael Rezende, Joaquin Zepeda, Jean Ponce, Francis Bach, Patrick Pérez

Paris Big-Data
2017-05-09

Image retrieval task

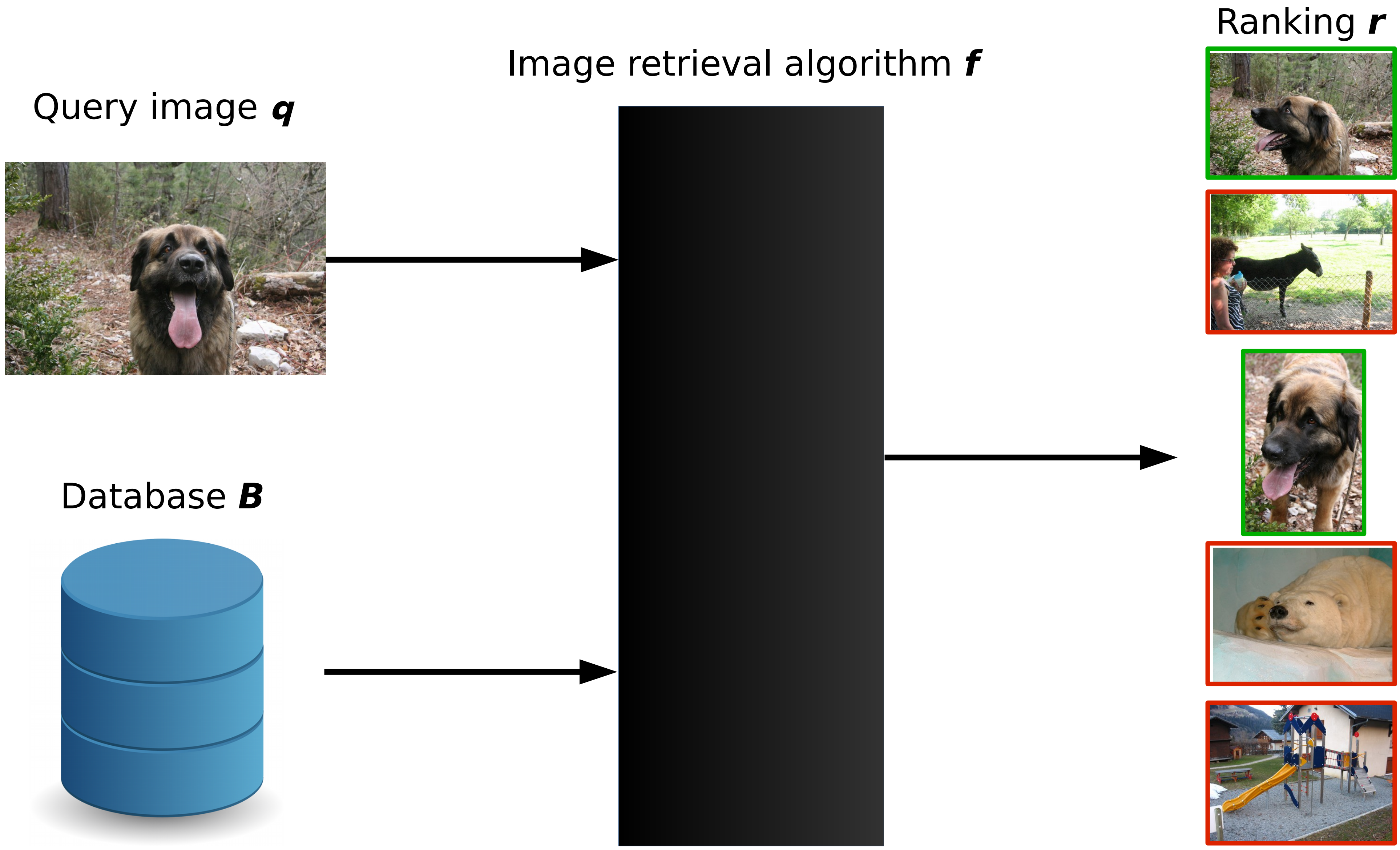
Image retrieval algorithm $\boldsymbol{f}$

Query image $\boldsymbol{q}$



Database $\boldsymbol{B}$

Image retrieval task



Evaluation: measuring the ranking of the correct matches.

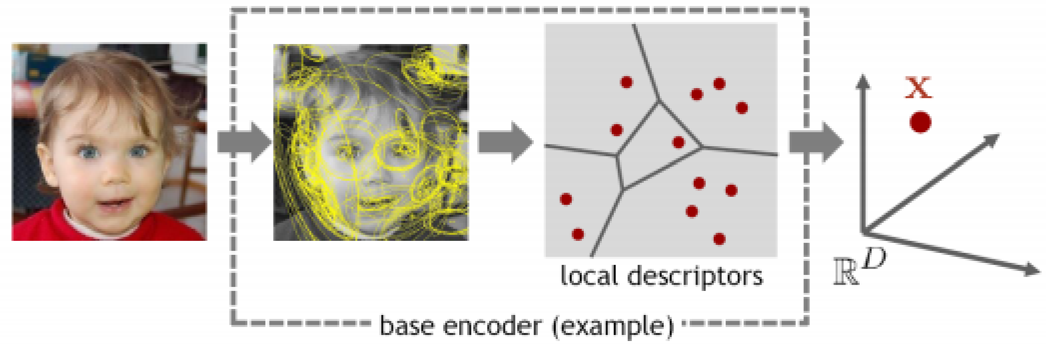
What makes a good retrieval algorithm?

> Time scalability

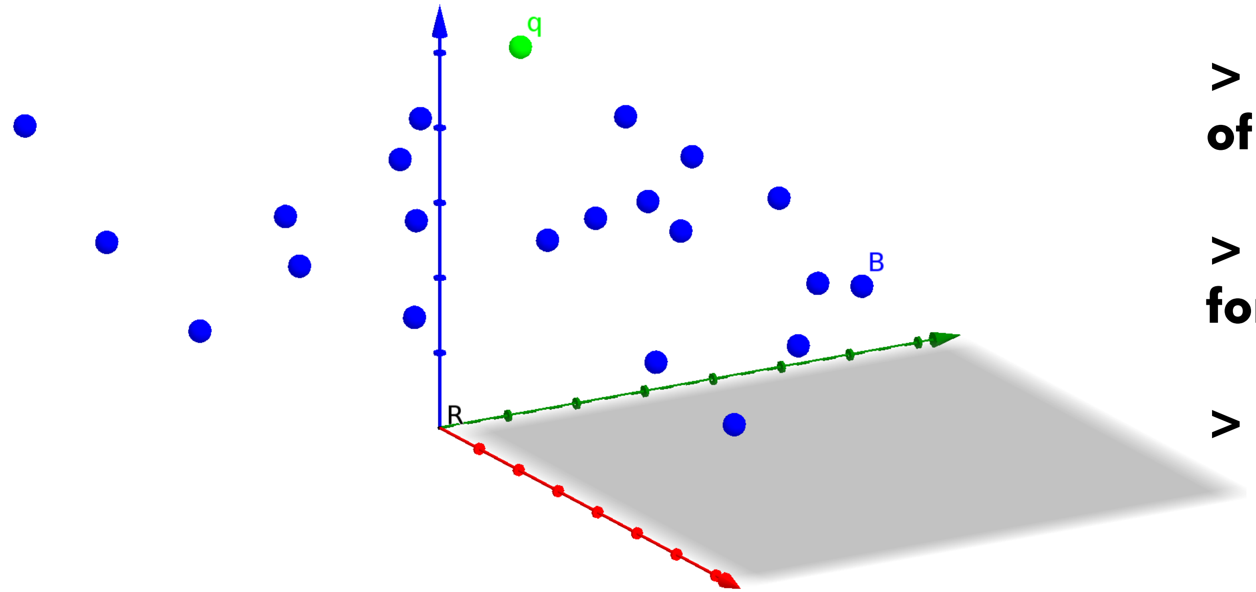
> Minimum storage

> Efficient evaluation

Global feature representation



Images are represented by a D-dimensional normalized feature vector.

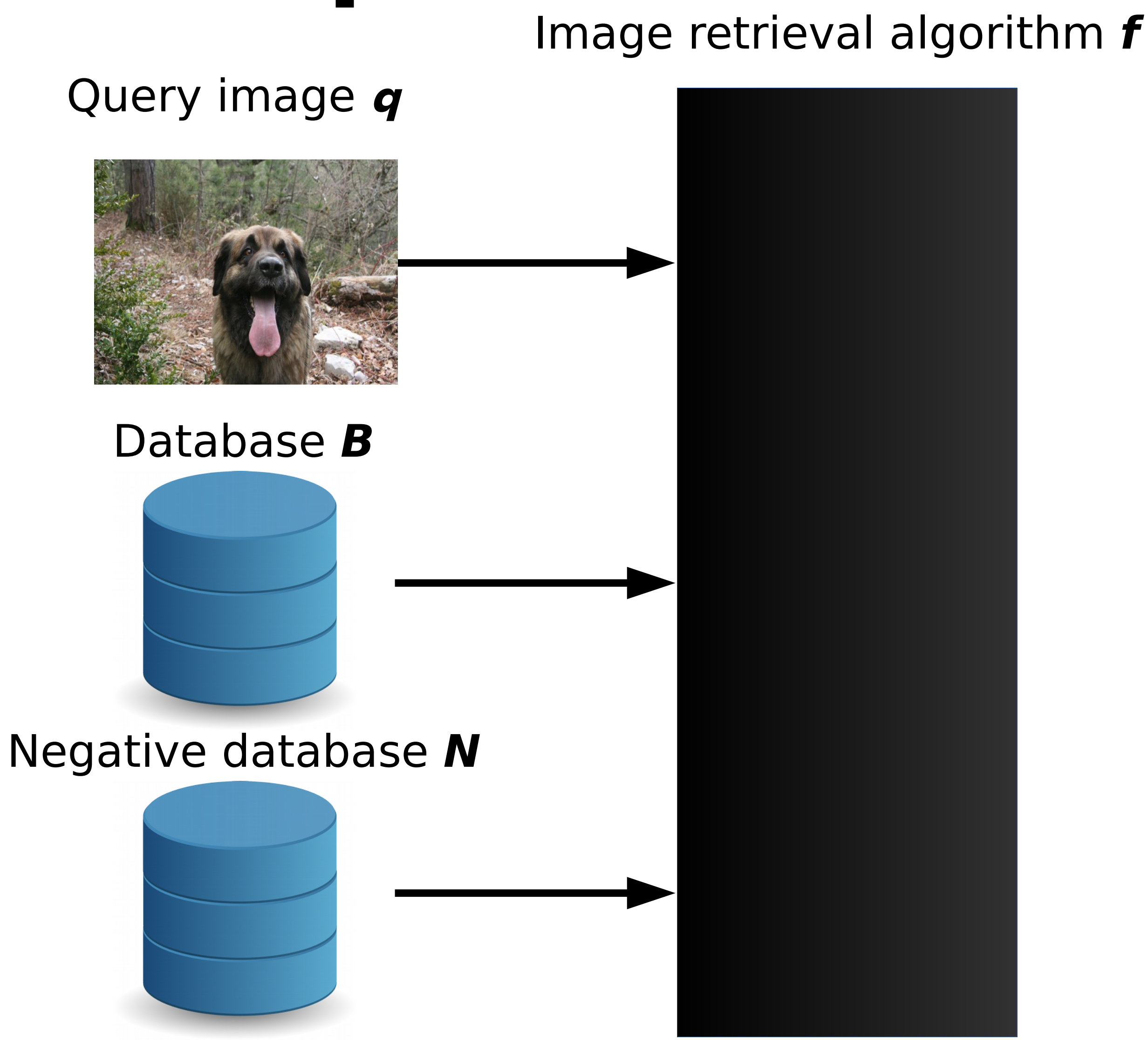


> Scalability: Precomputation of features of B.

> Storage: We store D bits for each image in B.

> Evaluation: Inner product.

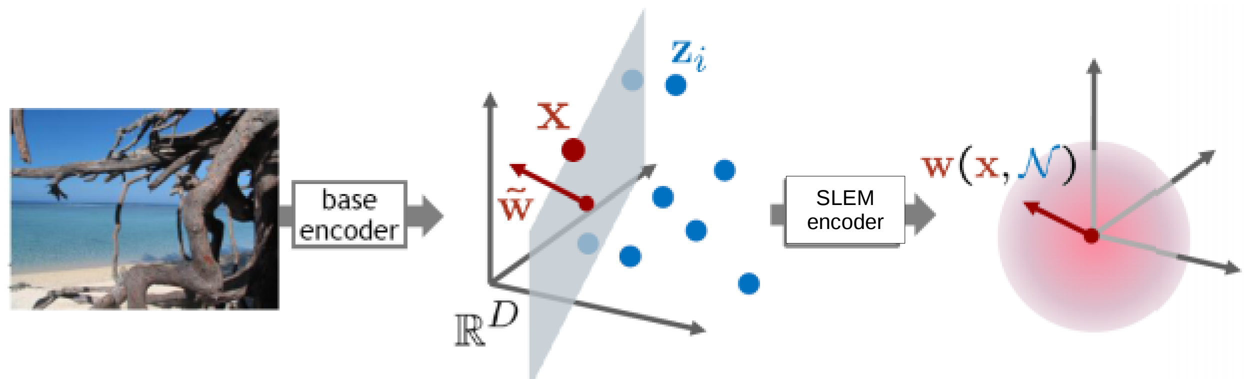
Exemplar classifier as feature representation



Square-loss exemplar machine (SLEM)

New feature representation is given the solution of an optimization problem.

$$J(\omega,\nu) = \theta\, l(1,\omega^T x+\nu)+\frac{1}{n}\sum_{i=1}^{n} l(-1,\omega^T z_i+\nu)+\frac{\lambda}{2}\|\omega\|^2$$



Offline (positive independent)

$$\begin{cases} \mu = & \frac{1}{n}\sum_{i=1}^{n} x_i, \\ U = & \frac{1}{n}XX^T - \mu\mu^T + \lambda \mathrm{Id}_d. \end{cases}$$

Online (positive dependent)

$$\begin{cases} \omega^\star & = \dfrac{2\theta}{\theta+1}U^{-1}(x - \mu), \\ \nu^\star & = \dfrac{\theta-1}{\theta+1} - \dfrac{1}{\theta+1}(\theta x + \mu)^T\omega^\star \end{cases}$$

Kernel feature representation

If the distance between two image representations can be measured by a "similarity score", we can generate a larger vectorial space by changing its similarity score.

Feature representation: $d(x, y) = \langle x, y \rangle,$

Kernel feature representation: $K(x, y) = \langle \varphi(x), \varphi(y) \rangle.$

By computing **{K(q, b), b** in **B}**, we compare feature representations in a larger Hilbert space (called kernel space) without augmenting storage.

Kernel SLEM

$$\min_{\alpha\in\mathbb{R}^n,\ \nu\in\mathbb{R}} \left(\frac{1}{n}\sum_{i=1}^{n} l(y_i, k_i^T\alpha + \nu) + \frac{\lambda}{2}\alpha^T K \alpha \right)$$

Offline: Cholesky decomposition of kernel matrix K.

$$K = BB^T \qquad \beta = B^T\alpha$$

Online: Add one new row to the decomposition B, corresponding to positive exemplar.

$$B' = \begin{bmatrix} u & v^T \\ 0 & B \end{bmatrix},\ v = B^\dagger k_0,\ u = \sqrt{k_{00} - ||v||^2}$$

$$J'(\beta,\nu) = \theta\ l(1, b_0'^T\beta + \nu) + \frac{1}{n}\sum_{i=1}^{n} l(-1, b_i'^T\beta + \nu) + \frac{\lambda}{2}||\beta||^2$$

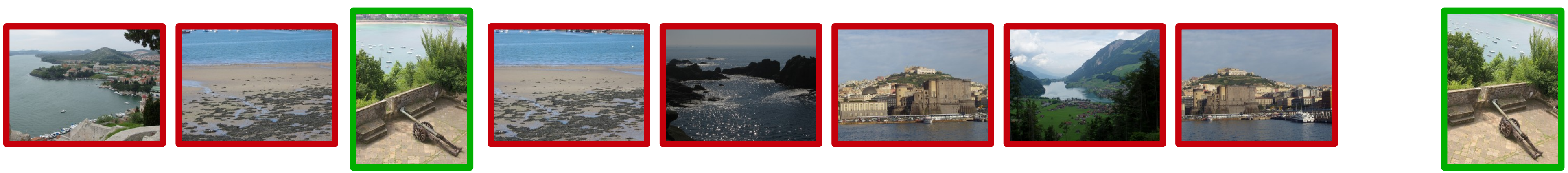
Solve linear problem similar to non-kernelized case.

Quantitative results

Query Image:

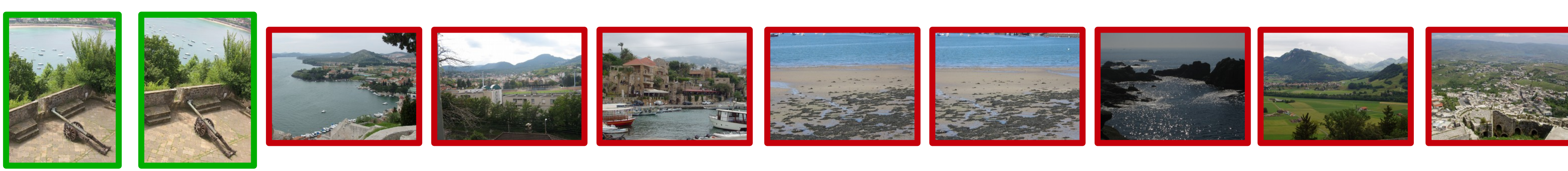
VLAD [1]

1. 2. 3. 4. 5. 6. 7. 8. … 225.



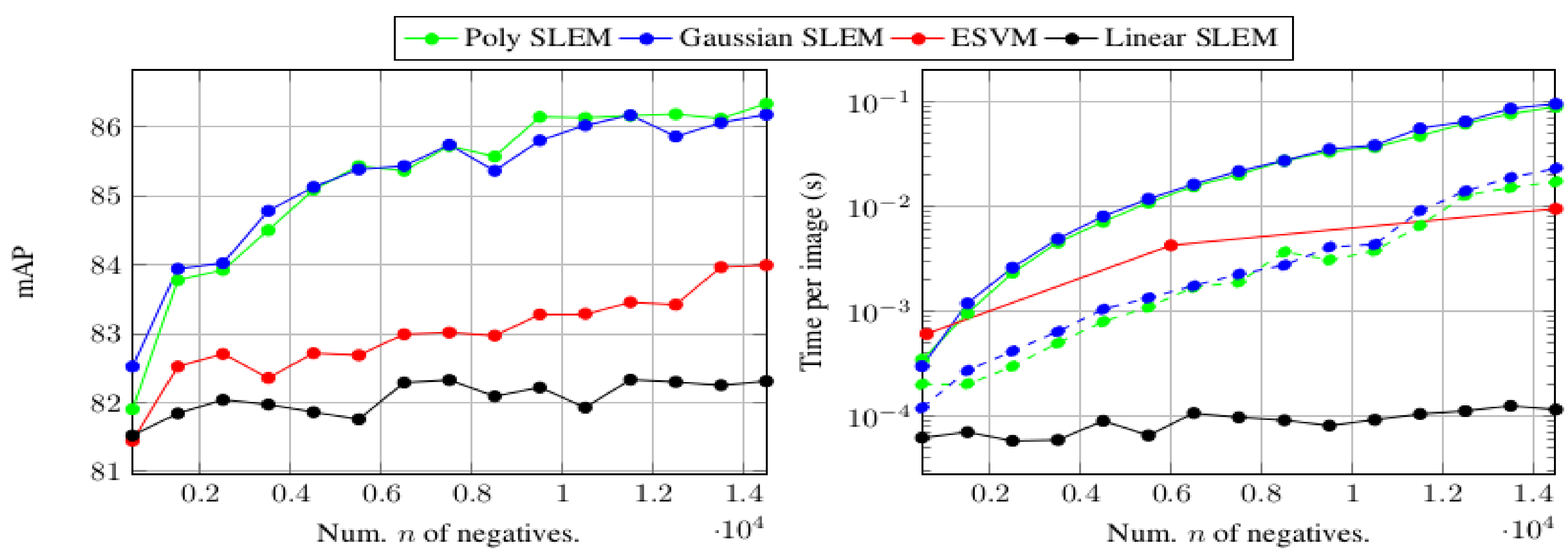
VLAD + PolySLEM

1. 2. 3. 4. 5. 6. 7. 8. 9. 10.



Qualitative results

Dataset	Holidays				Oxford 5k				Oxford 105k	
Model, features	VLAD	SPoC	AlexNet	NetVLAD	VLAD	SPoC	AlexNet	NetVLAD	SPoC	NetVLAD
Baseline	72.7	76.5	68.2	85.4	46.3	54.4	40.6	67.5	50.1	65.6
PCAW	75.5	81.7	69.2	88.3	50.9	63.7	45.0	69.1	55.5	66.1
LDA	54.7	82.2	64.1	74.3	29.6	62.2	42.5	72.7	52.4	40.7
ESVM [37]	77.5^3	84.0^3	71.3	91.4^2	57.2^3	62.1	43.9	72.5	56.5	67.5
Linear SLEM	78.0^2	82.3	72.1	91.3^3	**59.3**	64.1^3	46.2^3	72.9^3	56.7^3	68.0^3
Gaussian SLEM (16)	76.8	80.3	71.2	91.4^2	52.8	63.0	43.5	71.9	55.8	67.4
Gaussian SLEM (32)	77.4	81.7	72.0^3	91.4^2	54.9	63.1	44.0	71.1	56.0	67.8
Gaussian SLEM (fr)	**78.1**	86.2^2	**72.9**	**91.7**	59.0^2	**64.9**	47.0^2	**74.4**	59.5^2	70.0^2
Poly SLEM (16)	76.9	82.3	71.4	91.3^3	53.0	63.6	43.6	71.4	56.1	67.5
Poly SLEM (32)	77.3	82.4	72.1^2	**91.7**	54.9	63.6	44.1	71.6	56.3	67.9
Poly SLEM (fr)	**78.1**	**86.3**	**72.9**	**91.7**	**59.3**	64.8^2	**47.3**	74.1^2	**62.5**	**70.2**



Kernel Square-Loss Exemplar Machine for Image Retrieval

Thank you!

Rafael Rezende, Joaquin Zepeda, Jean Ponce, Francis Bach, Patrick Pérez

Paris Big-Data

2017-05-09