Insight
Centre for Data Analytics

DEEP LEARNING WORKSHOP

Dublin City University
21-22 May 2018

DCU

Day 2 Lecture 6

Content-based Image Retrieval

Eva Mohedano
eva.mohedano@insight-centre.org

Postdoctoral Researcher
Insight Centre for Data Analytics
Dublin City University

Overview

- What is content-based image retrieval?
- The classic SIFT retrieval pipeline
- Using off the shelf CNN features for retrieval
- Learning representations for retrieval

Overview

- **What is content-based image retrieval?**
- The classic SIFT retrieval pipeline
- Using off the shelf CNN features for retrieval
- Learning representations for retrieval

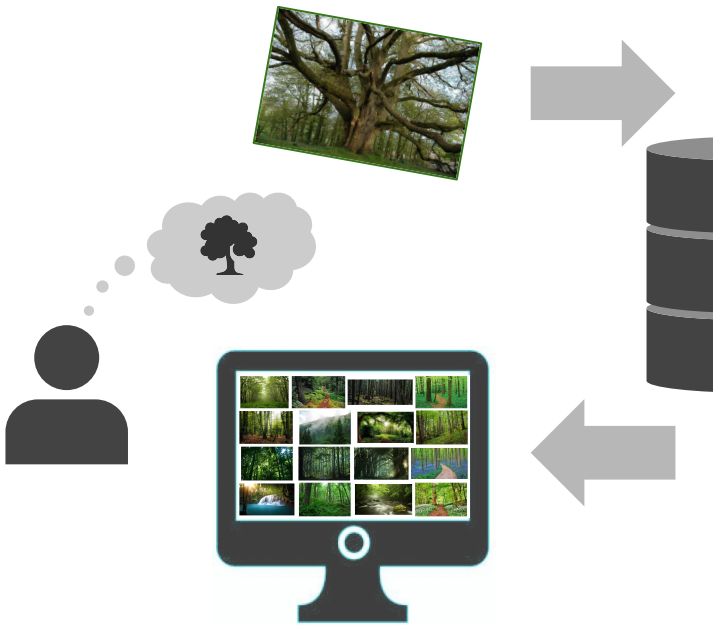
The problem: query by example

Given:

- An example query image that illustrates the user's information need
- A very large dataset of images

Task:

- Rank all images in the dataset according to how likely they are to fulfil the user's information need



Classification

Query: This chair

Results from dataset classified as “chair”

Retrieval

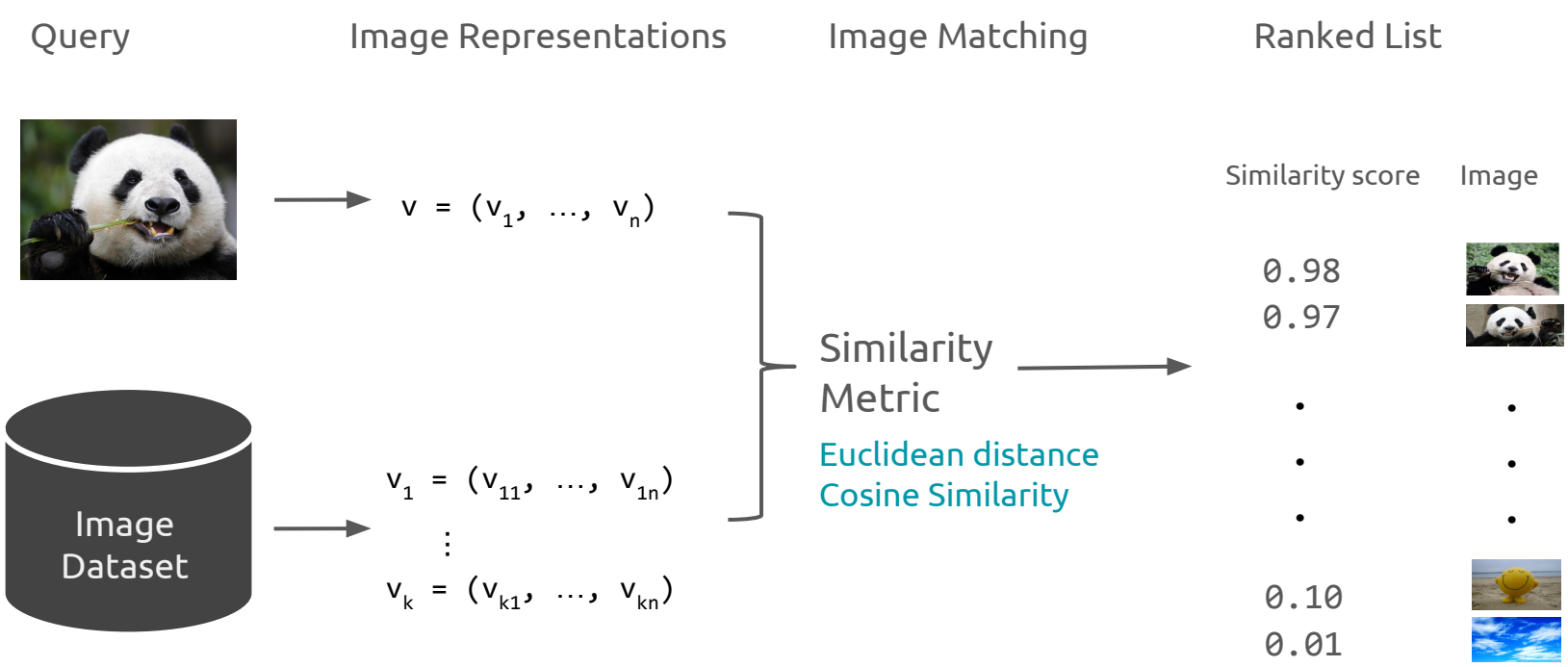
Query: This chair

Results from dataset ranked by similarity to the query

Overview

- What is content-based image retrieval?
- **The classic SIFT retrieval pipeline**
- Using off the shelf CNN features for retrieval
- Learning representations for retrieval

The retrieval pipeline



The classic SIFT retrieval pipeline

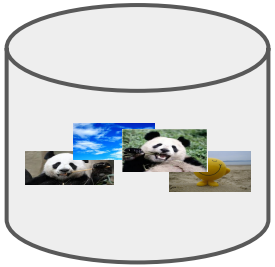
variable number of
feature vectors per image

$v_1 = (v_{11}, \ldots, v_{1n})$

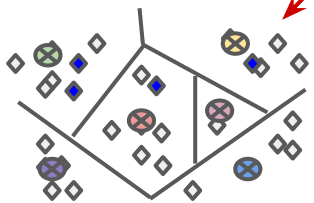
$\vdots$

$v_k = (v_{k1}, \ldots, v_{kn})$

Bag of Visual Words



N-Dimensional
feature space



M visual words
(M clusters)

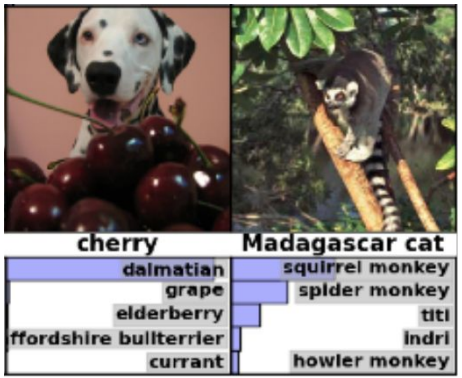
INVERTED FILE

word	Image ID
1	1, 12,
2	1, 30, 102
3	10, 12
4	2,3
6	10

Large vocabularies (50k-1M)
Very fast!
Typically used with SIFT features

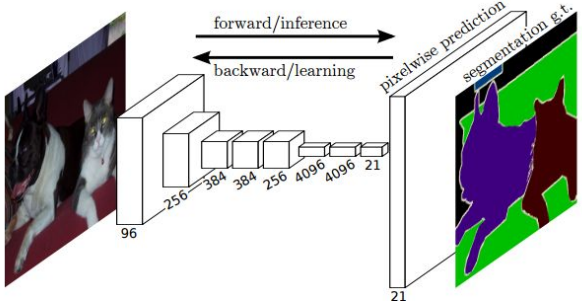
Convolutional neural networks and retrieval

Classification



Object Detection

Segmentation

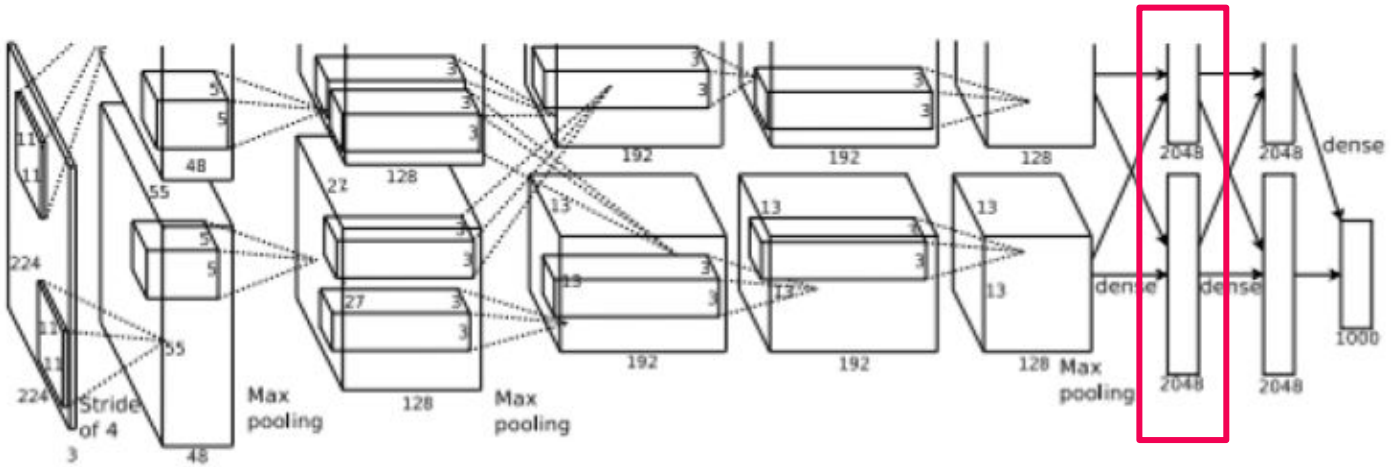


Overview

- What is content-based image retrieval?
- The classic SIFT retrieval pipeline
- **Using off the shelf CNN features for retrieval**
- Learning representations for retrieval

Off-the-shelf CNN representations

FC layers as global feature representation



Razavian et al. CNN features off-the-shelf: an astounding baseline for recognition. In CVPR
Babenko et al. Neural codes for image retrieval. ECCV 2014

Off-the-shelf CNN representations

Neural codes for retrieval [1]

- FC7 layer (4096D)
- L^2 norm + PCA whitening + L^2 norm
- Euclidean distance
- Only better than traditional SIFT approach after fine tuning on similar domain image dataset.

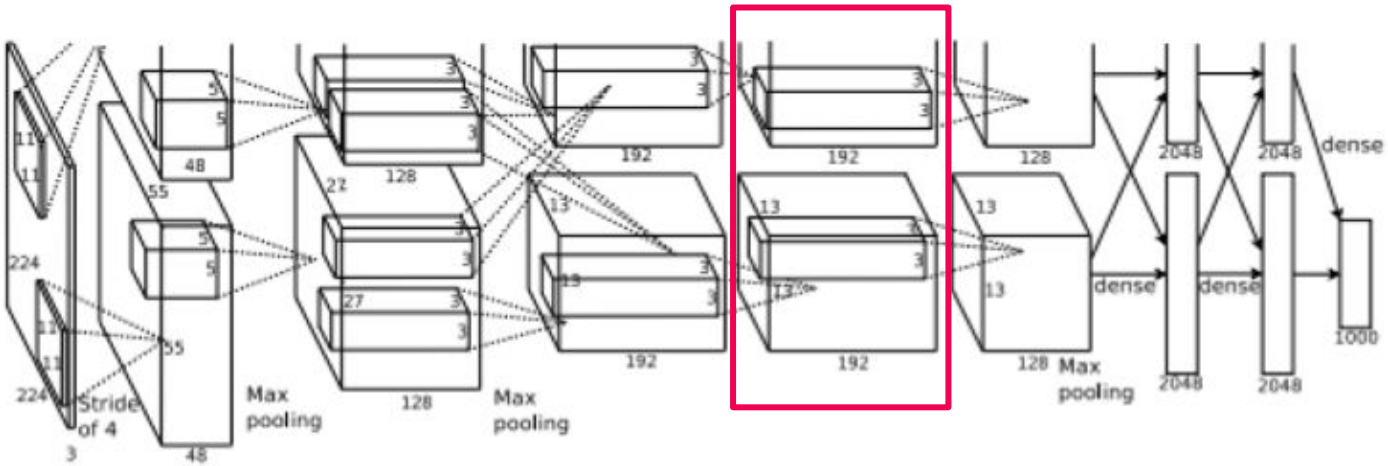
CNN features off-the-shelf: an astounding baseline for recognition [2]

- Extending Babenko's approach with spatial search
- Several features extracted by image (sliding window approach)
- Really good results but too computationally expensive for practical situations

[1] Babenko et al, Neural codes for image retrieval CVPR 2014
[2] Razavian et al, CNN features off-the-shelf: an astounding baseline for recognition, CVPR 2014

Off-the-shelf CNN representations

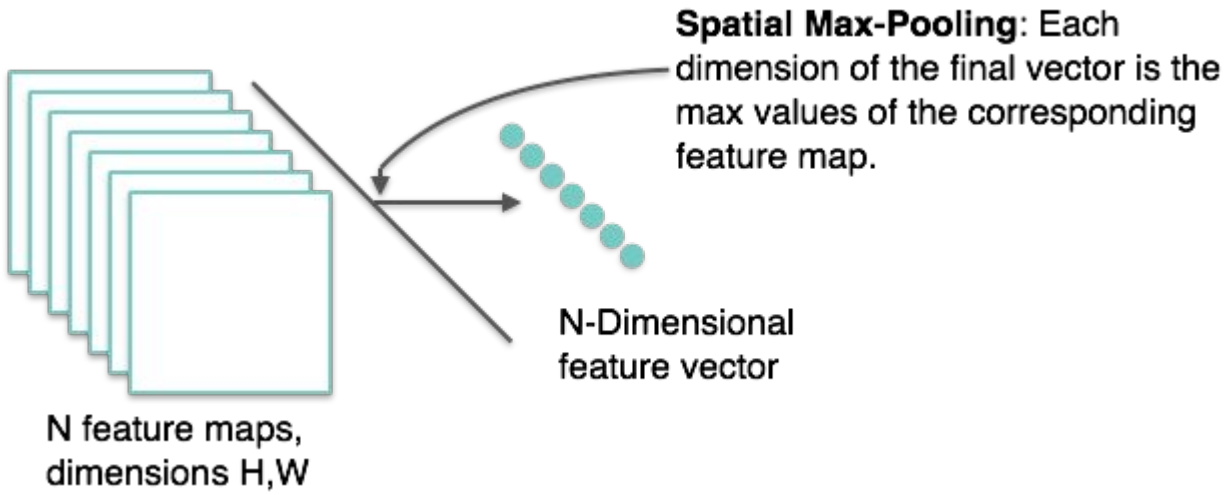
sum/max pool conv features across filters



Babenko and Lempitsky, Aggregating local deep features for image retrieval. ICCV 2015
Tolias et al. Particular object retrieval with integral max-pooling of CNN activations. *arXiv:1511.05879*.
Kalantidis et al. Cross-dimensional Weighting for Aggregated Deep Convolutional Features. *arXiv:1512.04065*.

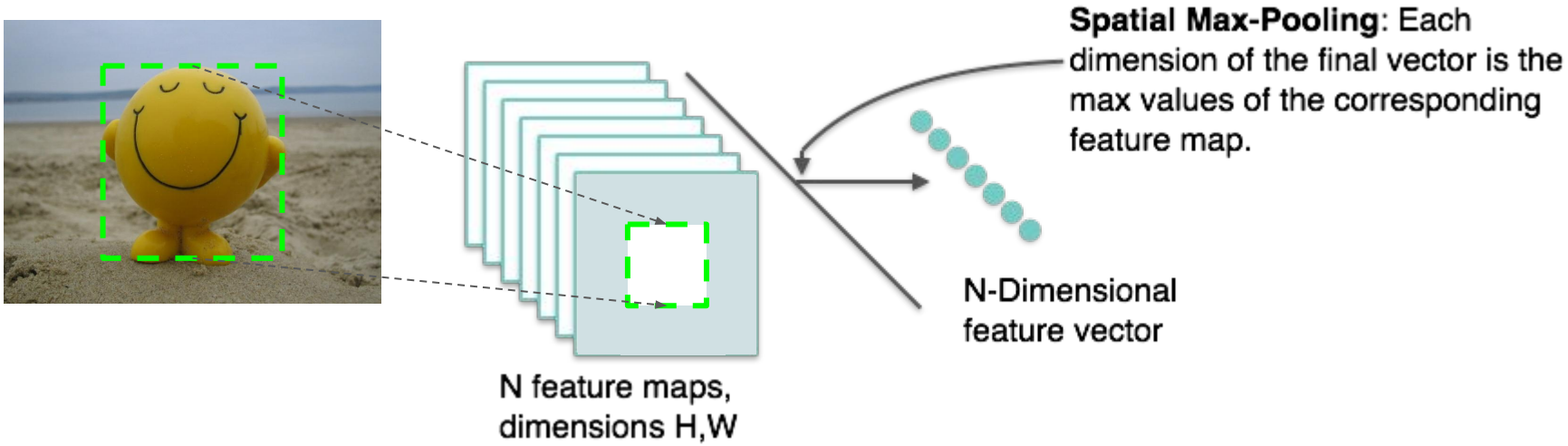
Off-the-shelf CNN representations

Descriptors from convolutional layers



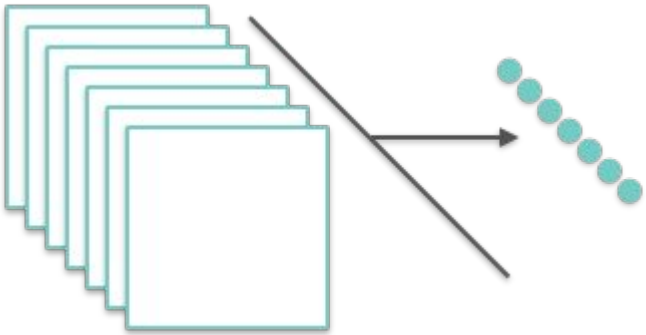
Off-the-shelf CNN representations

Pooling features on conv layers allow to describe specific parts of an image

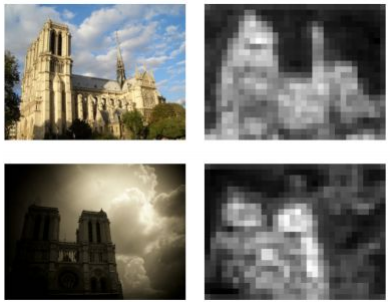


Off-the-shelf CNN representations

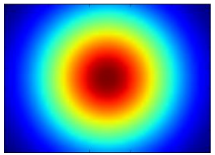
Sum/max pooling operation of a conv layer



Apply spatial weighting on the features before pooling them



[1] weighting based on ‘strength’ of the local features



[2] weighting based on the distance to the center of the image

[2] Babenko and Lempitsky, Aggregating local deep features for image retrieval. ICCV 2015
[1] Kalantidis et al. Cross-dimensional Weighting for Aggregated Deep Convolutional Features. ECCV 2016

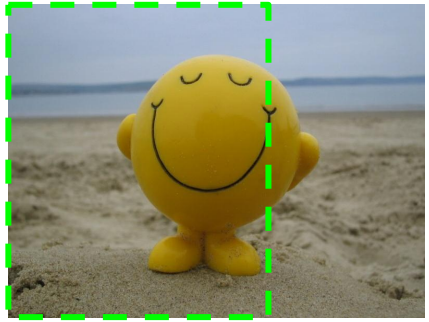
R-MAC

Regional Maximum Activation of Convolution.

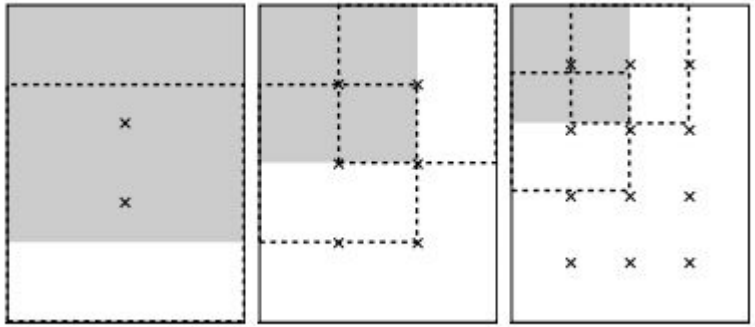
Settings

- Fully convolutional off-the-shelf VGG16
- Pool5
- Spatial Max pooling
- High Resolution images
- Global descriptor based on aggregating region vectors
- Sliding window approach

R-MAC

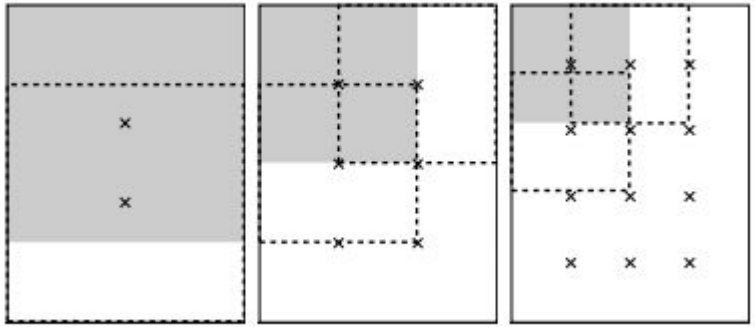


Region1

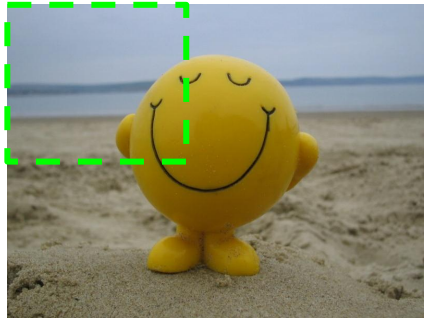


R-MAC

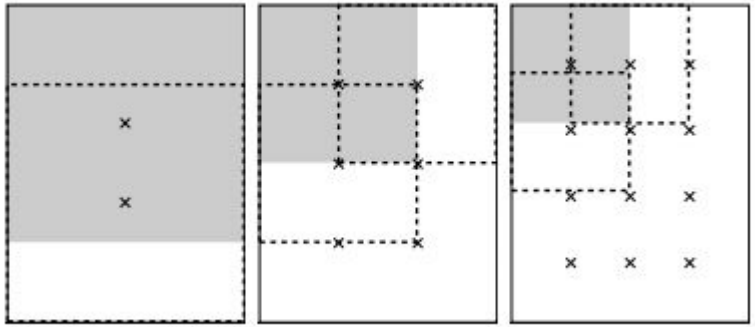
Region1
Region2



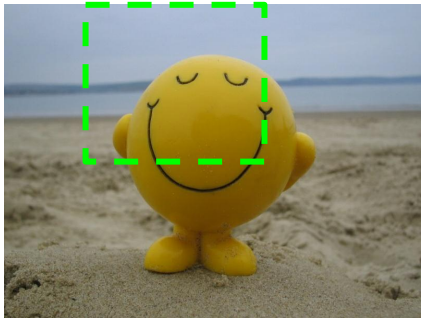
R-MAC



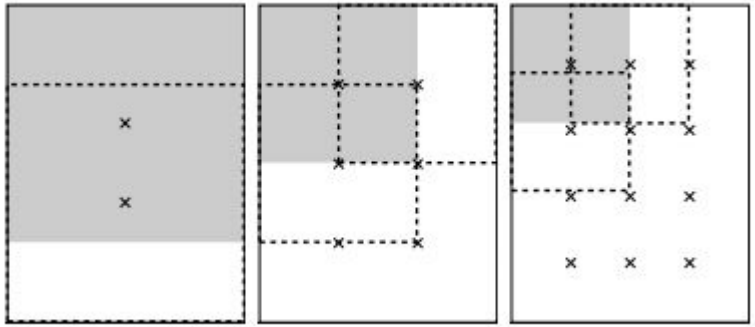
Region1
Region2
Region3



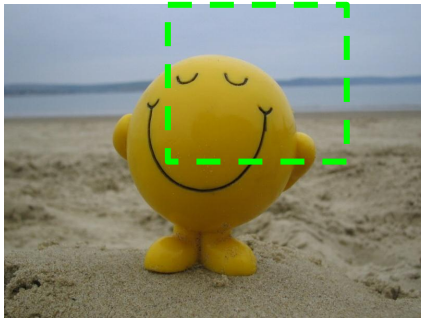
R-MAC



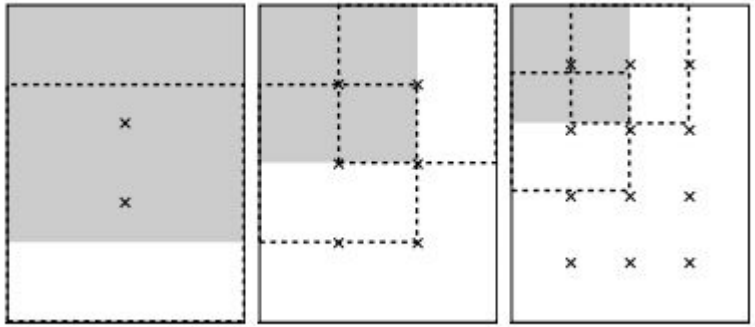
Region1
Region2
Region3
Region4



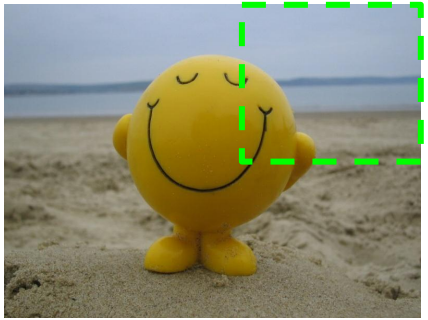
R-MAC



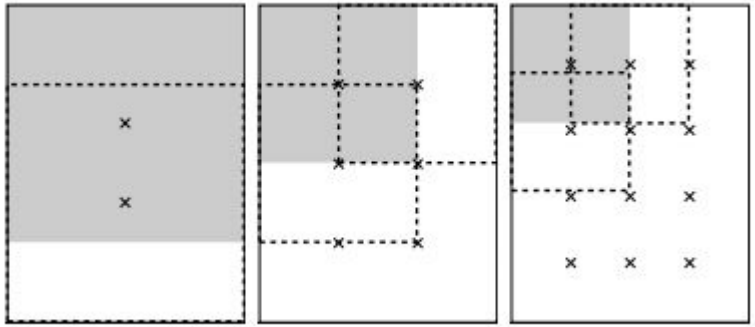
Region1
Region2
Region3
Region4
...



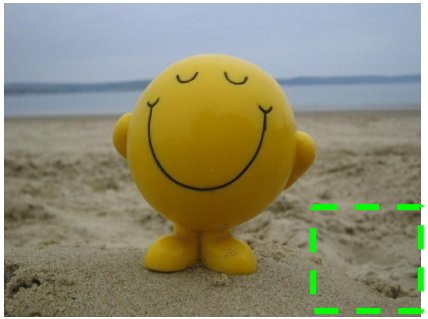
R-MAC



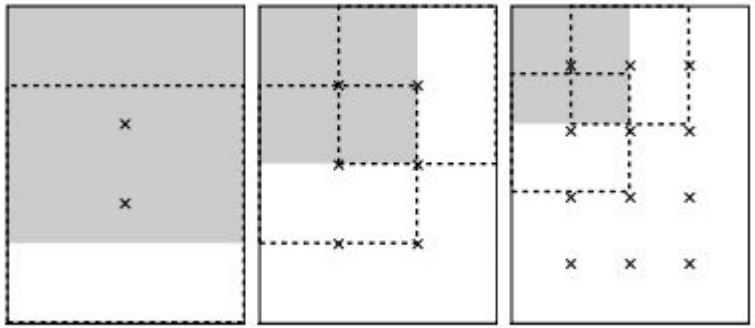
Region1
Region2
Region3
Region4
...



R-MAC



Region1
Region2
Region3
Region4
...
RegionN



R-MAC

Region vectors
512D

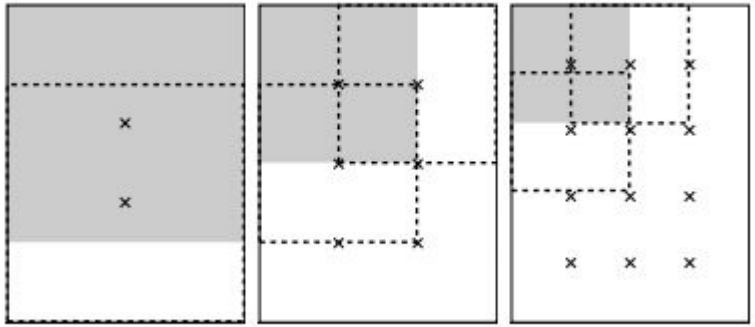
Region1
Region2
Region3
Region4
...
RegionN

l2-PCAw-l2

Region1
Region2
Region3
Region4
...
RegionN

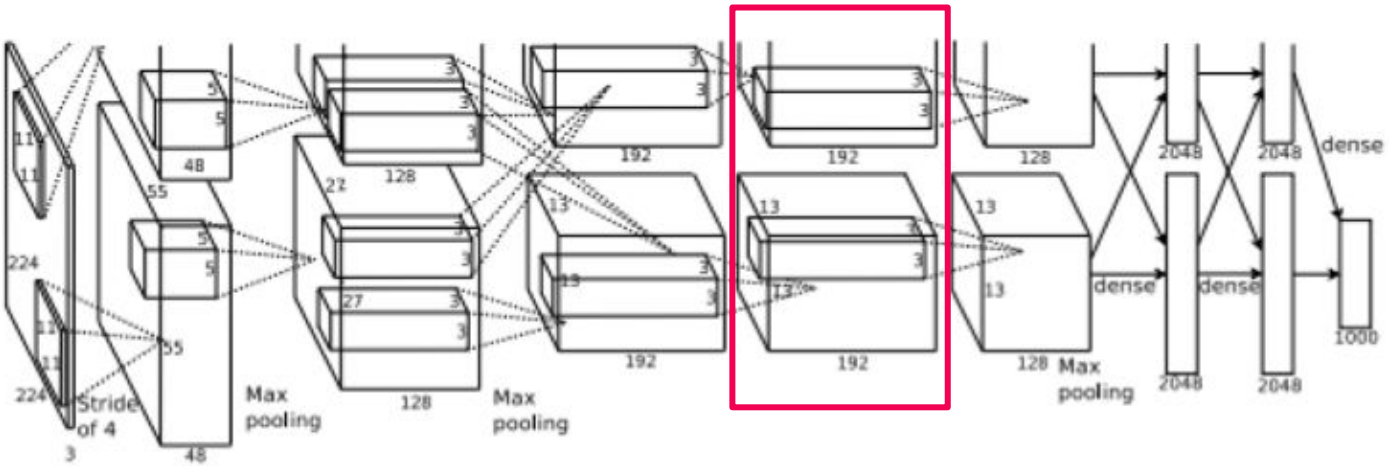
Σ

Global
vector
512D



Off-the-shelf CNN representations

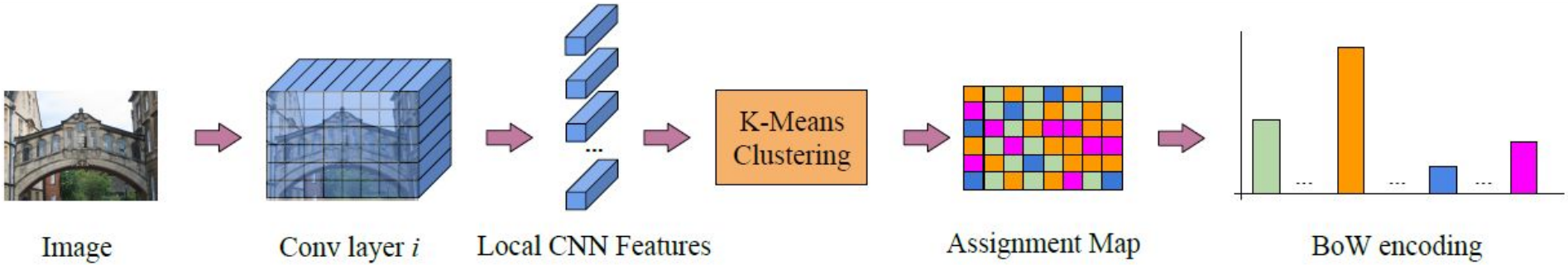
BoW, VLAD encoding of conv features



Ng et al. Exploiting local features from deep networks for image retrieval. CVPR Workshops 2015
Mohedano et al. Bags of Local Convolutional Features for Scalable Instance Search. ICMR 2016

Off-the-shelf CNN representations

Descriptors from convolutional layers



(336x256) Resolution

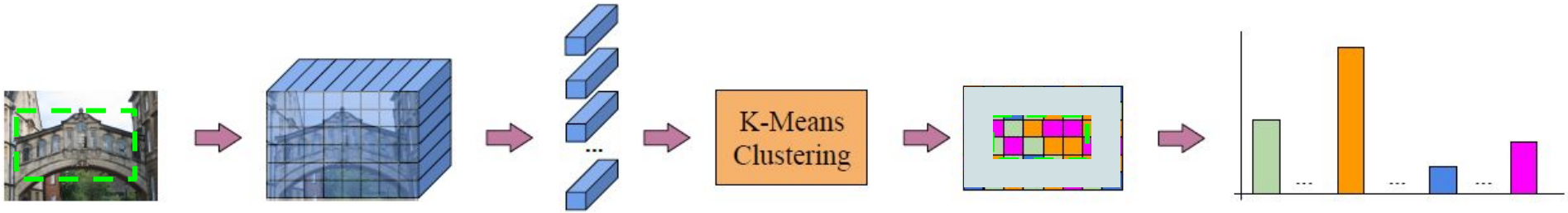
conv5_1 from VGG16 (42x32)

25K centroids

25K-D vector

Mohedano et al. Bags of Local Convolutional Features for Scalable Instance Search. ICMR 2016

Off-the-shelf CNN representations



(336x256)
Resolution

conv5_1 from
VGG16
(42x32)

25K centroids

25K-D vector

Off-the-shelf CNN representations

Paris Buildings 6k Oxford Buildings 5k

TRECVID Instance Search 2013
(subset of 23k frames)

		Oxford 5k	Paris 6k	INS 23k
BoW	GS	0.650	0.698	**0.323**
	LS	**0.739**	**0.819**	0.295
Sum pooling (as ours)	GS	0.606	0.712	0.156
	LS	0.583	0.742	0.097
Sum pooling (as in [7])	GS	0.672	0.774	0.139
	LS	0.683	0.763	0.120

[7] Kalantidis et al. Cross-dimensional Weighting for Aggregated Deep Convolutional Features. *arXiv:1512.04065*.

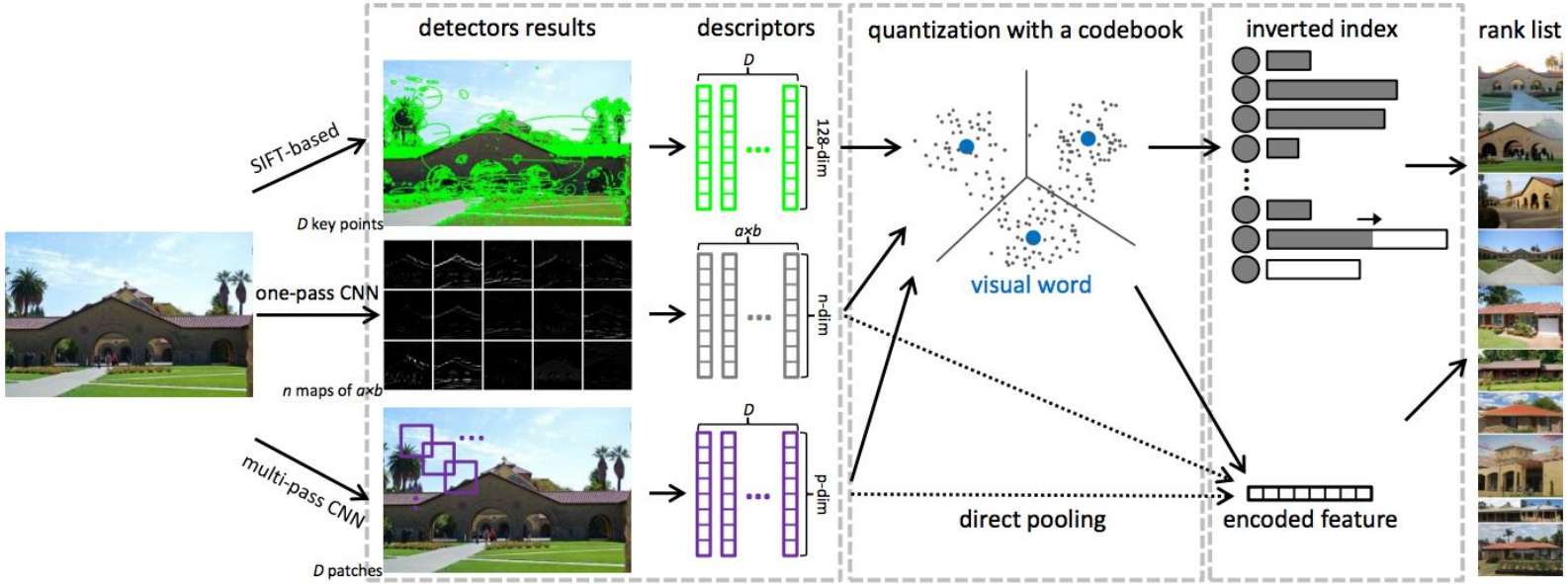
Mohedano et al. Bags of Local Convolutional Features for Scalable Instance Search. ICMR 2016

Off-the-shelf CNN representations

CNN representations

- L^2 Normalization + PCA whitening + L^2 Normalization
- Cosine similarity
- Convolutional features better than fully connected features
- Convolutional features keep spatial information ➔ retrieval + object location
- Convolutional layers allows custom input size.
- If data labels available, fine tuning the network to the image domain improves CNN representations.

CNN as a feature extractor



source：Zheng, SIFT Meets CNN: A Decade Survey of Instance Retrieval , 2016

Overview

- What is content-based image retrieval?
- The classic SIFT retrieval pipeline
- Using off the shelf CNN features for retrieval
- **Learning representations for retrieval**

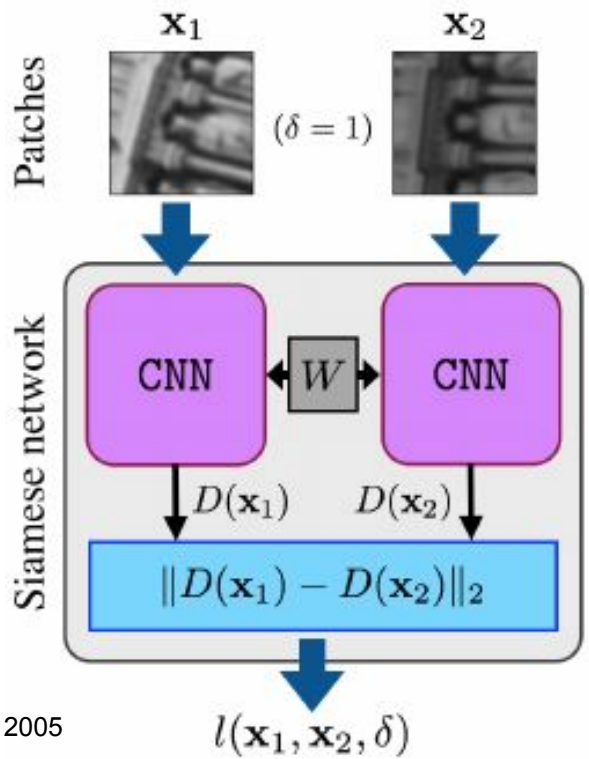
What loss function should we use?

Learning representations for retrieval

Siamese network: network to learn a function that maps input patterns into a target space such that L^2 norm in the target space approximates the semantic distance in the input space.

Applied in:

- Dimensionality reduction [1]
- Face verification [2]
- Learning local image representations [3]



[1] Song et al.: Deep metric learning via lifted structured feature embedding. CVPR 2015
[2] Chopra et al. Learning a similarity metric discriminatively, with application to face verification CVPR' 2005
[3] Simo-Serra et al. Fracking deep convolutional image descriptors. CoRR, abs/1412.6537, 2014

Learning representations for retrieval

Siamese network: network to learn a function that maps input patterns into a target space such that L^2 norm in the target space approximates the semantic distance in the input space.

$$l(\mathbf{x}_1, \mathbf{x}_2, \delta) = \delta \cdot l_P(d_D(\mathbf{x}_1, \mathbf{x}_2)) + (1 - \delta) \cdot l_N(d_D(\mathbf{x}_1, \mathbf{x}_2))$$

$$l_P(d_D(\mathbf{x}_1, \mathbf{x}_2)) = d_D(\mathbf{x}_1, \mathbf{x}_2)$$

$$l_N(d_D(\mathbf{x}_1, \mathbf{x}_2)) = \max(0, m - d_D(\mathbf{x}_1, \mathbf{x}_2))$$

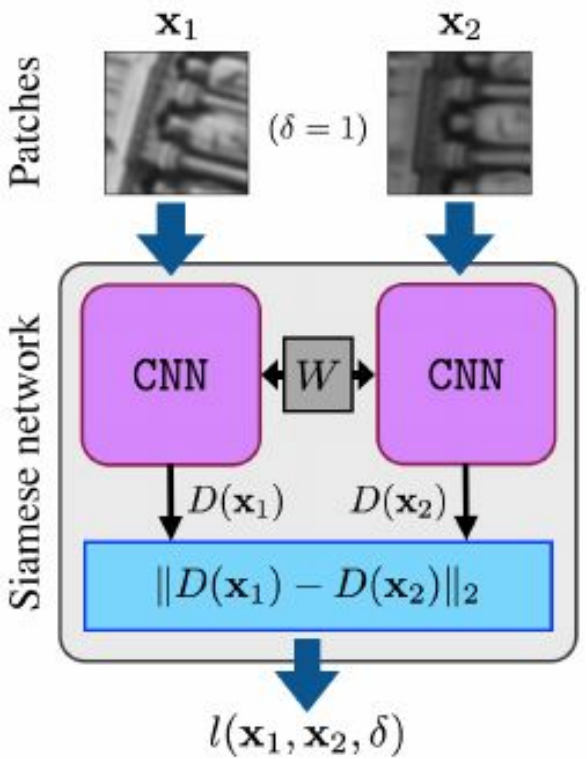
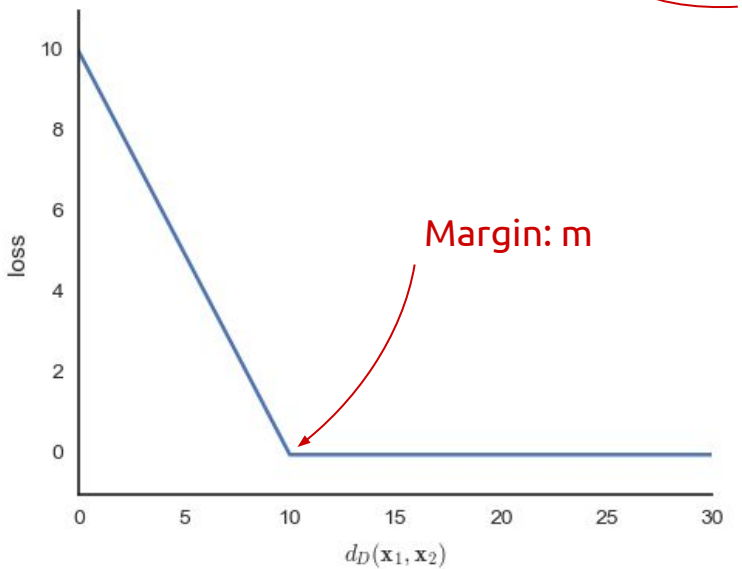


Image credit: Simo-Serra et al. Fracking deep convolutional image descriptors. CoRR 2014

$$l_N(d_D(\mathbf{x}_1, \mathbf{x}_2)) = \max(0, m - d_D(\mathbf{x}_1, \mathbf{x}_2))$$

Hinge loss

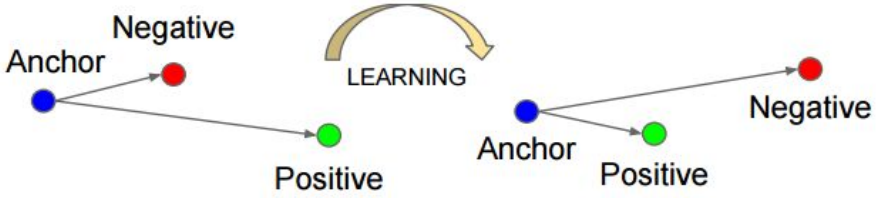
Negative pairs: if nearer than the margin, pay a linear penalty



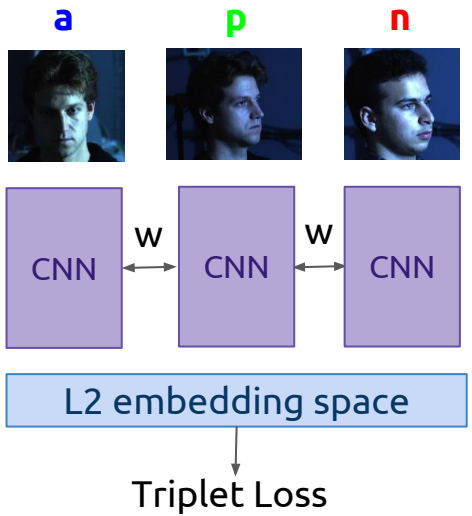
The margin is a **hyperparameter**

Learning representations for retrieval

Siamese network with triplet loss: loss function minimizes distance between query and positive and maximizes distance between query and negative



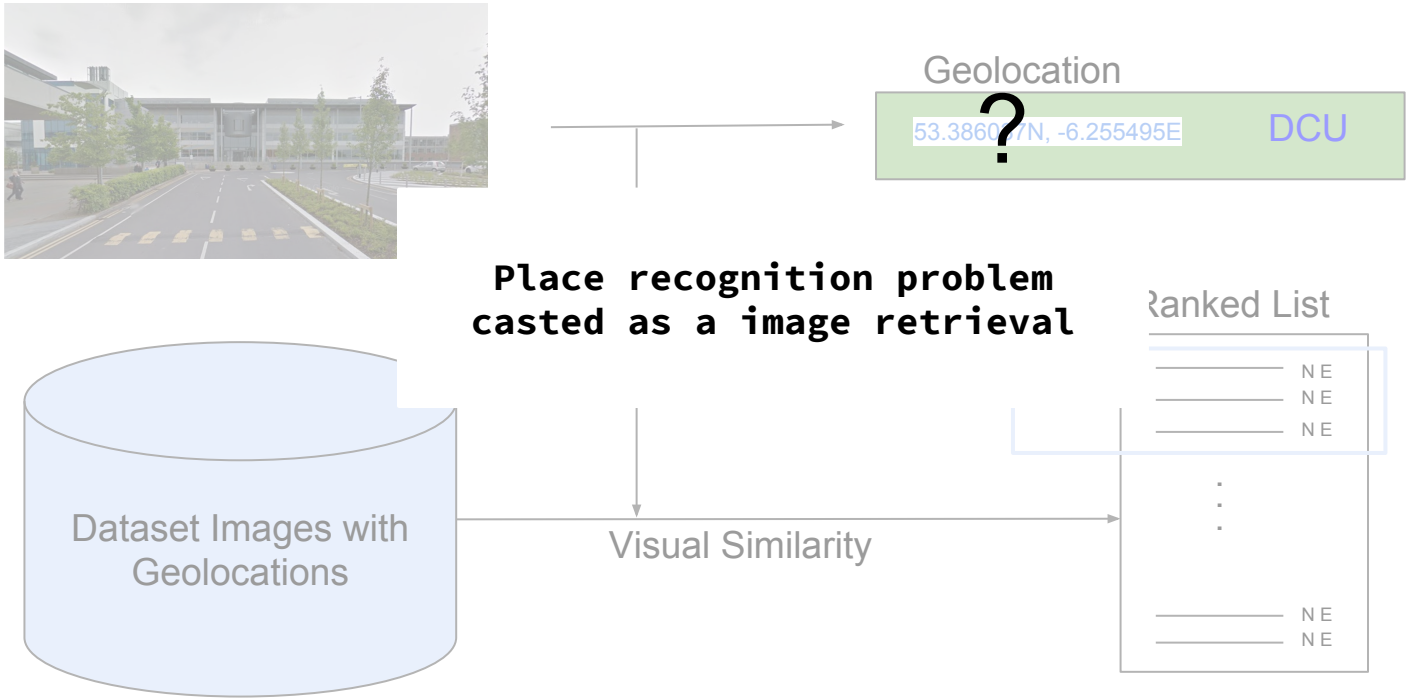
$$\|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2$$



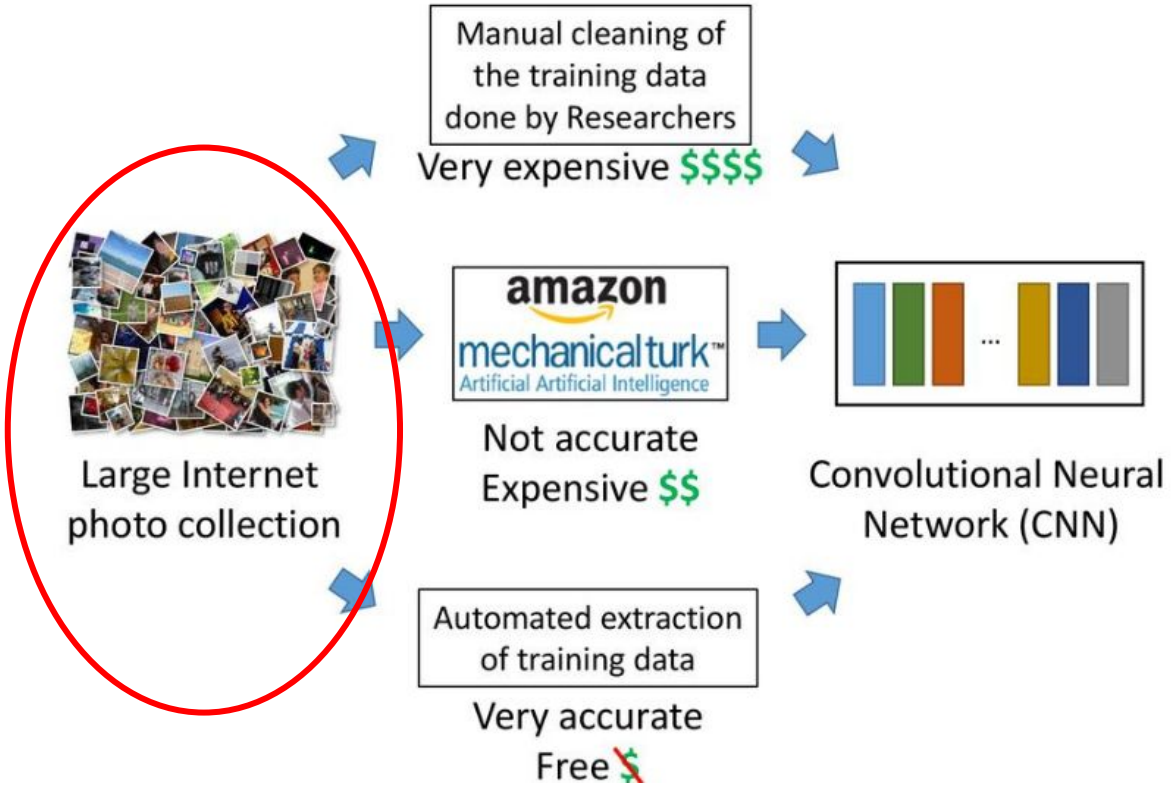
Schroff et al. FaceNet: A Unified Embedding for Face Recognition and Clustering, CVPR 2015

How do we get the training data?

Exploring Image datasets with GPS coordinates (Google Time Machine Data)



Relja Arandjelović et al, NetVLAD: CNN architecture for weakly supervised place recognition, CVPR 2016



Automatic data cleaning

Strong baseline (SIFT) + geometric verification + 3D camera estimation[1]

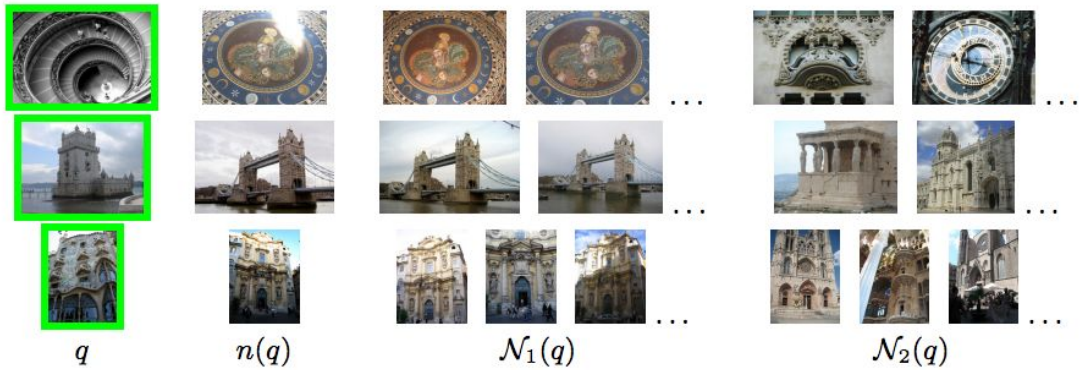
Further manual inspection

Further manual inspection

Radenović, CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples, CVPR 2016
Gordo, Deep Image Retrieval: Learning global representations for image search, CVPR 2016

[1] Schonberger From single image query to detailed 3D reconstruction

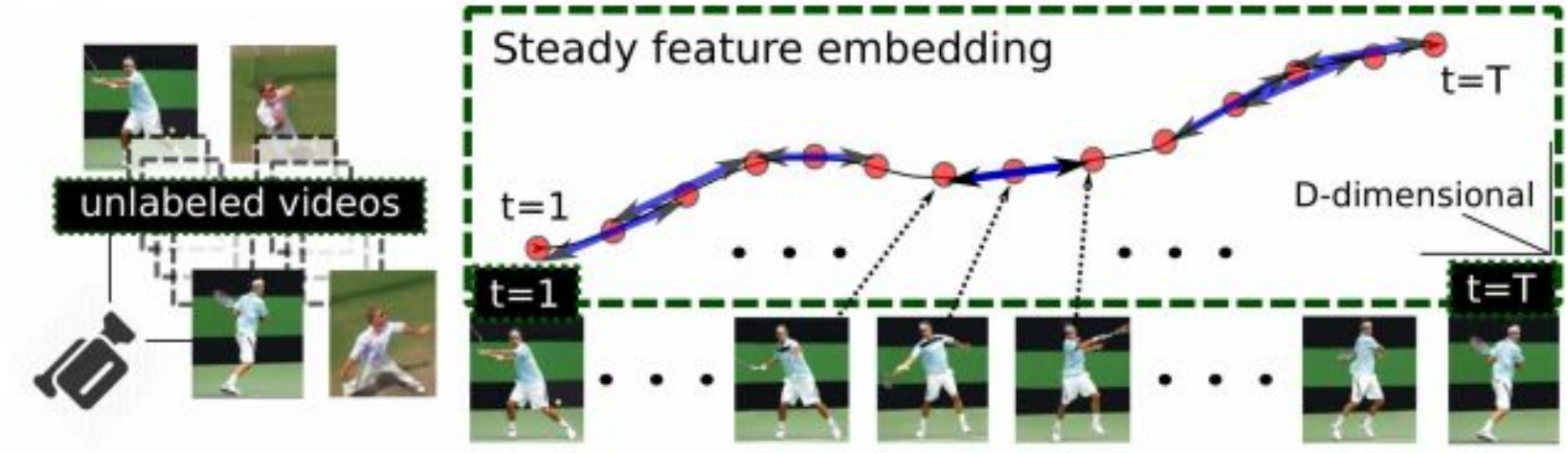
Generating training pairs



Select the hardest negative

Select triplet which generates larger loss

Radenović, CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples, CVPR 2016



Jayaraman and Grauman, Slow and steady feature analysis: higher order temporal coherence in video, ECCV 2016

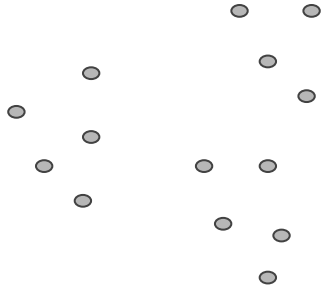
NetVLAD

Relja Arandjelović et al, NetVLAD: CNN architecture for weakly supervised place recognition, CVPR 2016

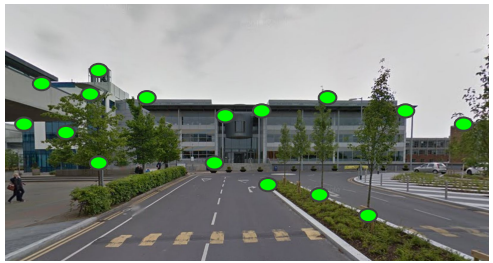
VLAD

$\{\mathbf{x}_i\}$ Local features
Vectors dimension *D*

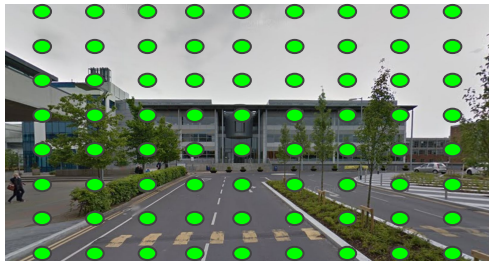
Feature Space D-dim

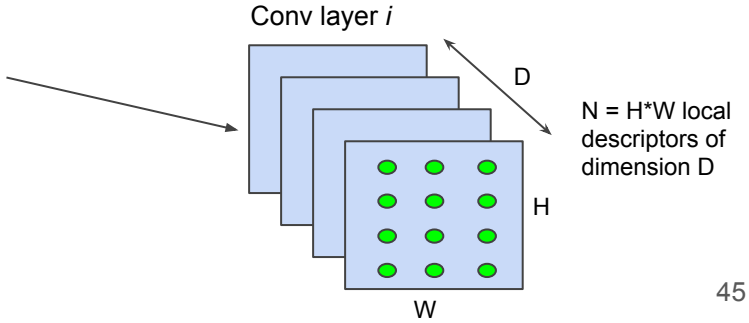


Sparse local feature extraction



Dense Local Feature extraction



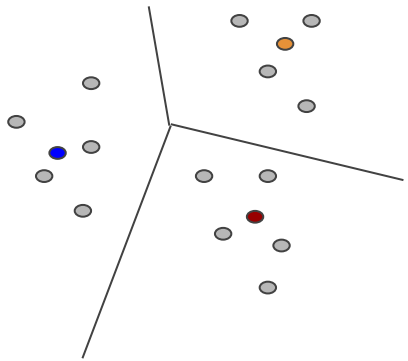


VLAD

$\{\mathbf{x}_i\}$ Local features
Vectors dimension *D*

$\{\mathbf{c}_k\}$ *K* cluster centers
Visual Vocabulary
(K vectors with dimension *D*)
Learnt from data with K-means

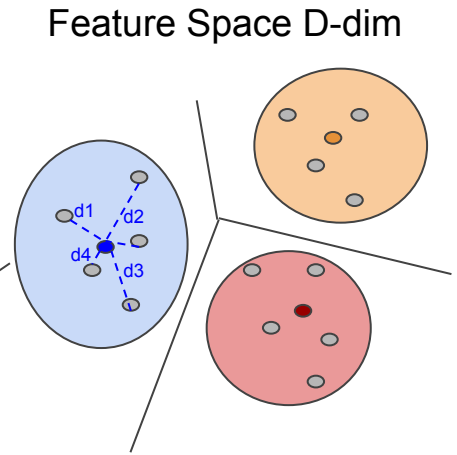
Feature Space D-dim



VLAD

$\{\mathbf{x}_i\}$ Local features
Vectors dimension *D*

$\{\mathbf{c}_k\}$ *K* cluster centers
Visual Vocabulary
(K vectors with dimension *D*)
Learnt from data with K-means

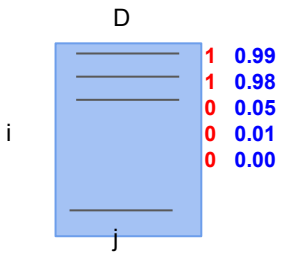


Sum of all residuals of the assignments to the cluster 1

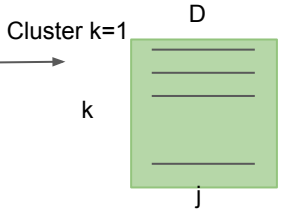
VLAD vector

VLAD

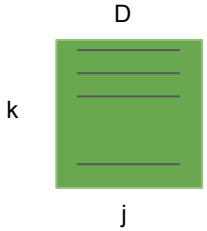
N Local features {Xi}



K Clusters features {Ck}



V Matrix VLAD (residuals)



$$V(j,k) = \sum_{i=1}^{N} a_k(\mathbf{x}_i)\,(x_i(j) - c_k(j))\,,$$

Hard Assignment

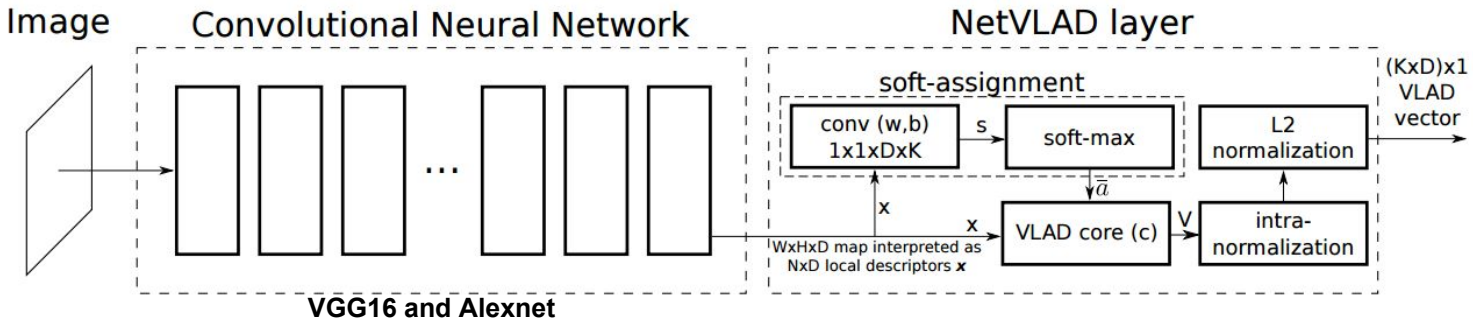
Not differentiable !

Soft Assignment

$$\bar{a}_k(\mathbf{x}_i) = \frac{e^{-\alpha\|\mathbf{x}_i - \mathbf{c}_k\|^2}}{\sum_{k'} e^{-\alpha\|\mathbf{x}_i - \mathbf{c}_{k'}\|^2}},$$

Softmax function :)

$$\bar{a}_k(\mathbf{x}_i) = \frac{e^{\mathbf{w}_k^T \mathbf{x}_i + b_k}}{\sum_{k'} e^{\mathbf{w}_{k'}^T \mathbf{x}_i + b_{k'}}},$$



VGG16 and Alexnet

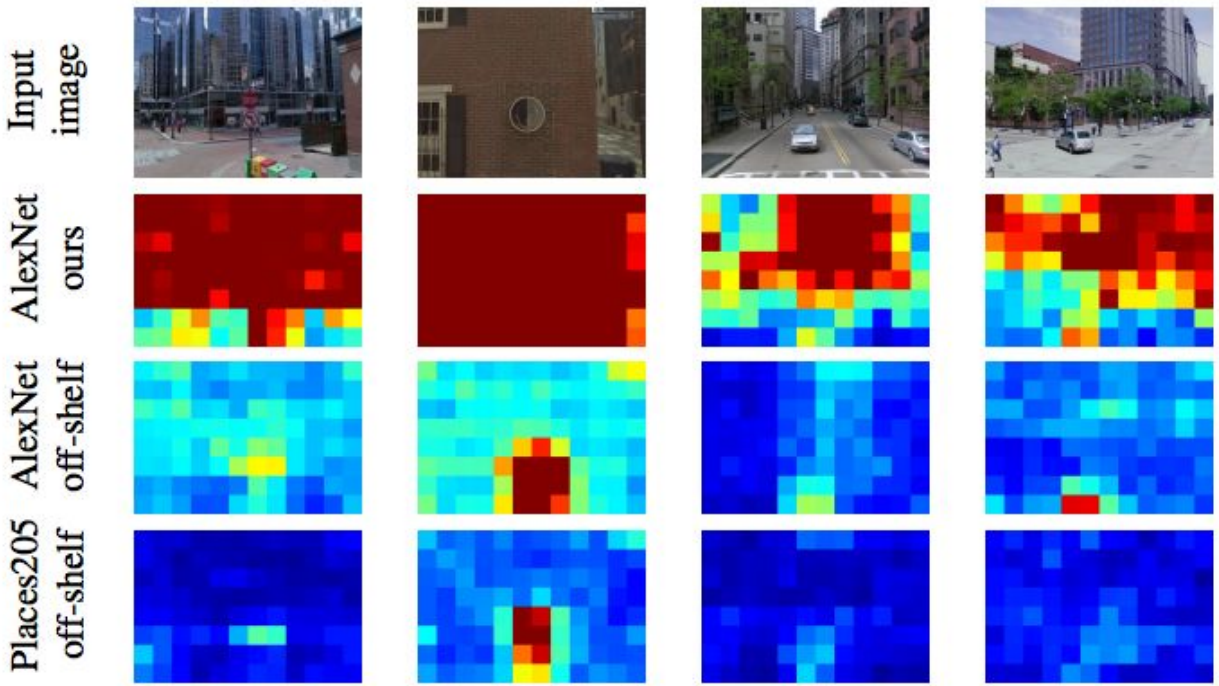
$$L_\theta = \sum_j l\left(\min_i d_\theta^2(q, p_i^q) + m - d_\theta^2(q, n_j^q)\right)$$

$$l(x) = \max(x, 0),$$

Consider the positive example closer to the query

Consider all negative candidates

With distance inferior to a certain margin



Relja Arandjelović et al, NetVLAD: CNN architecture for weakly supervised place recognition, CVPR 2016

Method	Oxford 5k (full)	Oxford 5k (crop)	Paris 6k (full)	Paris 6k (crop)	Holidays (orig)	Holidays (rot)
Jégou and Zisserman [32]	–	47.2	–	–	65.7	65.7
Gordo *et al.* [23]	–	–	–	–	78.3	–
Razavian *et al.* [62]	53.3†	–	67.0†	–	74.2	–
Babenko and Lempitsky [7]	58.9	53.1	–	–	–	80.2
a. Ours: NetVLAD off-the-shelf	53.4	55.5	64.3	67.7	**82.1**	**86.0**
b. Ours: NetVLAD trained	**62.5**	**63.5**	**72.0**	**73.5**	79.9	84.3

Relja Arandjelović et al, NetVLAD: CNN architecture for weakly supervised place recognition, CVPR 2016

Fine-tuned R-MAC

R-MAC

Region vectors
512D

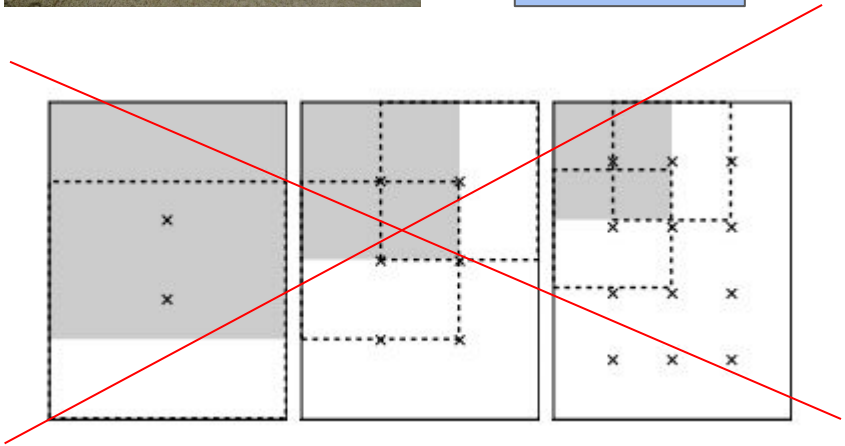
Region1
Region2
Region3
Region4
...
RegionN

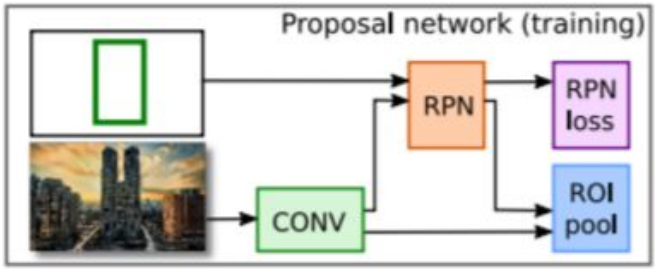
l2-PCAw-l2

Region1
Region2
Region3
Region4
...
RegionN

Σ

Global
vector
512D



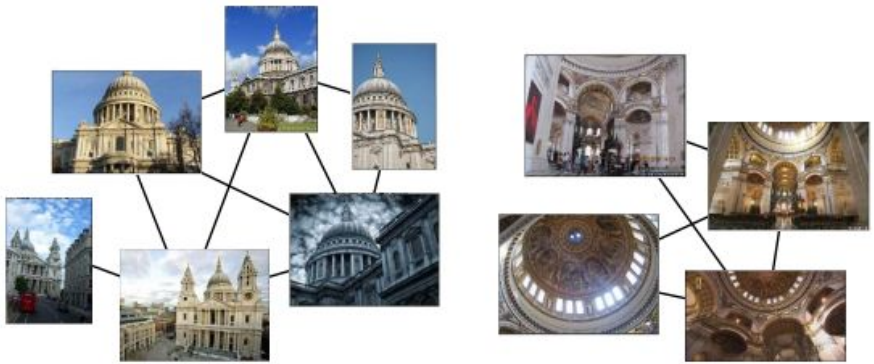


Learning representations for retrieval

Generated dataset to create the training triplets

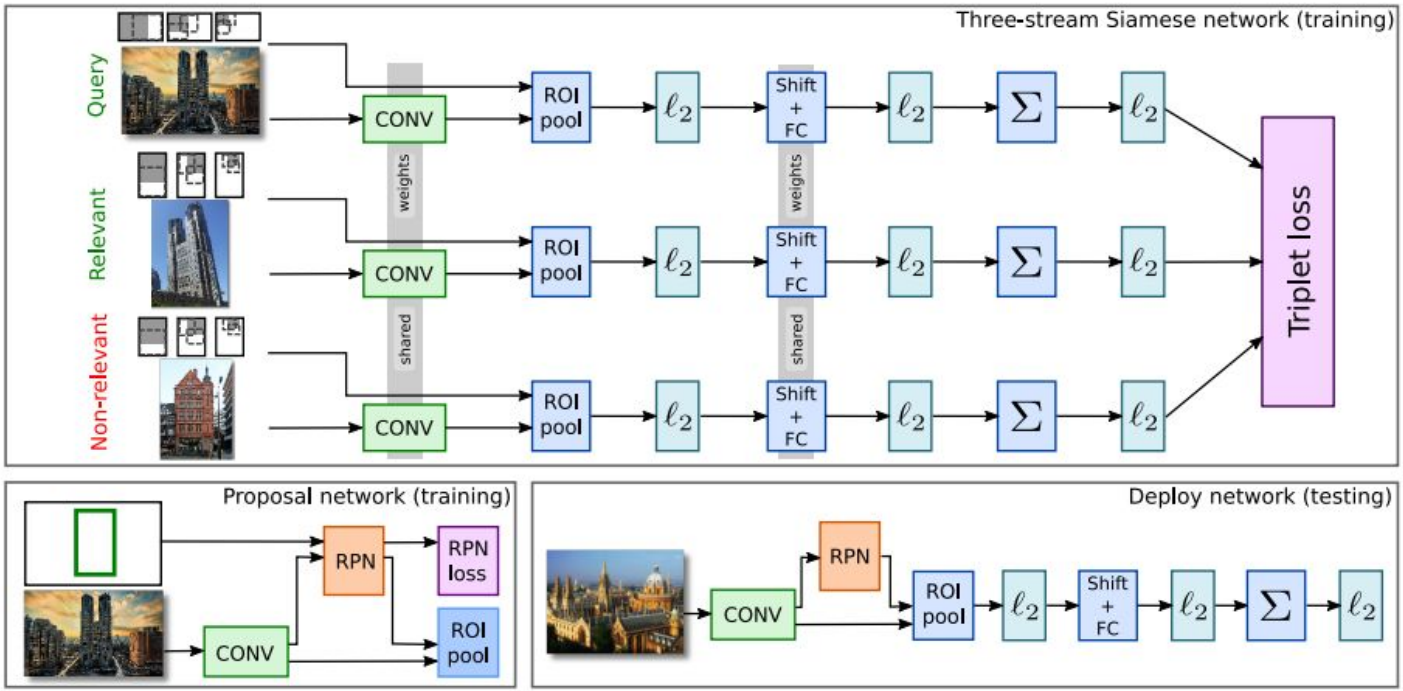
Dataset: Landmarks dataset:

- 214K images of 672 famous landmark site.
- Dataset processing based on a matching baseline: SIFT + Hessian-Affine keypoint detector.
- Important to select the “useful” triplets.



Gordo et al, Deep Image Retrieval: Learning global representations for image search. ECCV 2016

Learning representations for retrieval



Gordo et al, Deep Image Retrieval: Learning global representations for image search. ECCV 2016

Learning representations for retrieval

Comparison between R-MAC from off-the-shelf network and R-MAC retrained for retrieval

		R-MAC		Learned R-MAC		
Dataset	PCA	[14]	Reimp.	C-Full	C-Clean	R-Clean
Oxford 5k	PCA Paris	66.9	66.1	-	-	-
	PCA Landmarks	-	64.7	75.3	75.9	**78.6**
Paris 6k	PCA Oxford	83.0	82.5	-	-	-
	PCA Landmarks	-	81.6	82.2	83.7	**84.5**

Gordo et al, Deep Image Retrieval: Learning global representations for image search. ECCV 2016

Summary

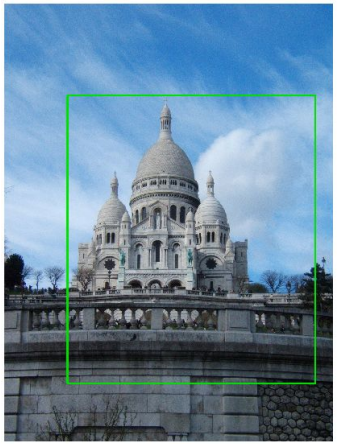
- Pre-trained CNN are useful to generate image descriptors for retrieval
- Convolutional layers allow us to encode local information
- Knowing how to rank similarity is the primary task in retrieval
- Designing CNN architectures to learn how to rank

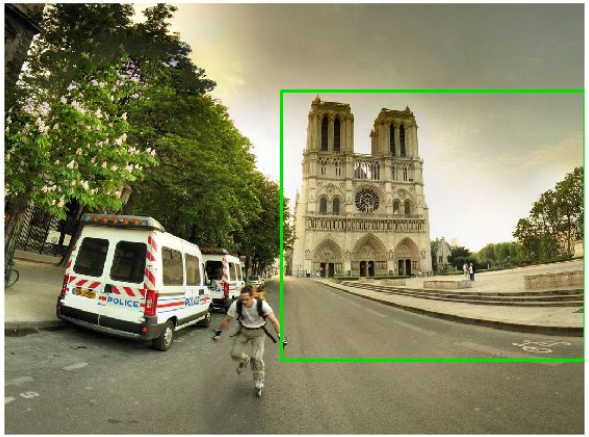
Thank you!

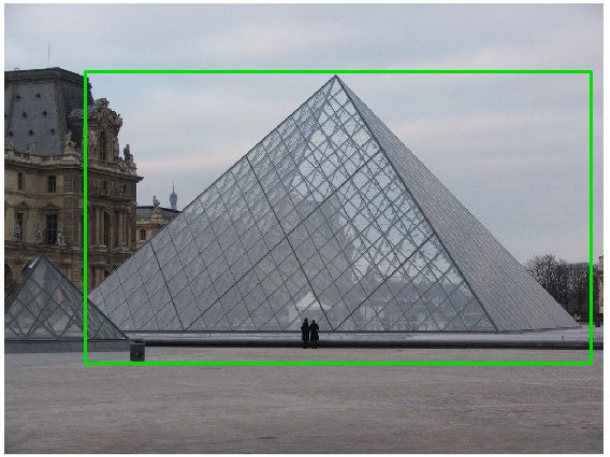
Summary state-of-the-art

		Layer	Method	Dim	Oxford 5k	Paris 6k
Off-the-shelf CNN representations	Global representation	fc	Neural Codes	128	0.433	-
		conv	SPoC	256	0.657	-
			CroW	256	0.684	0.765
			R-MAC	256	0.561	0.729
		conv	NetVLAD	256	0.555	0.643
			BoCF	25k(*)	0.739	0.82
	Multiple representations	fc	CNN-astounding	4-15k	0.68	0.795
		conv	CNN-astounding2	32k	0.843	0.879
Learning representations for retrieval	Global representation	conv	MAC	512	0.774	0.824
			regionProposal+MAC	512	**0.813**	**0.855**
			NetVLAD	512	0.676	0.749

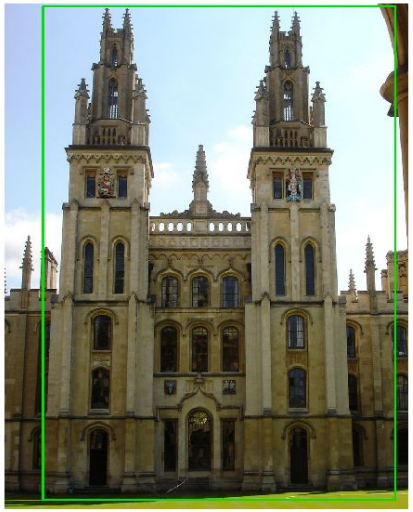
PARIS dataset

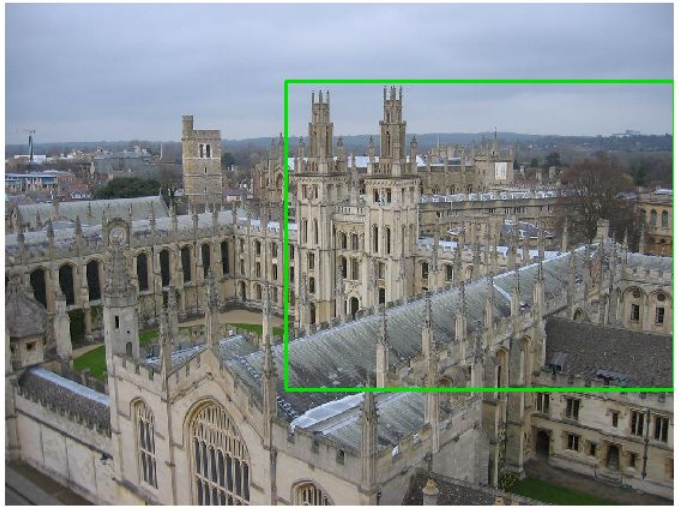


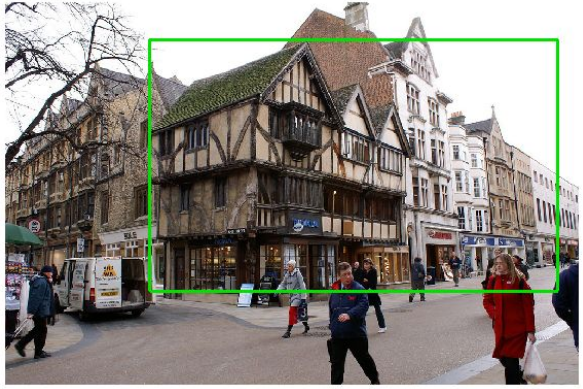


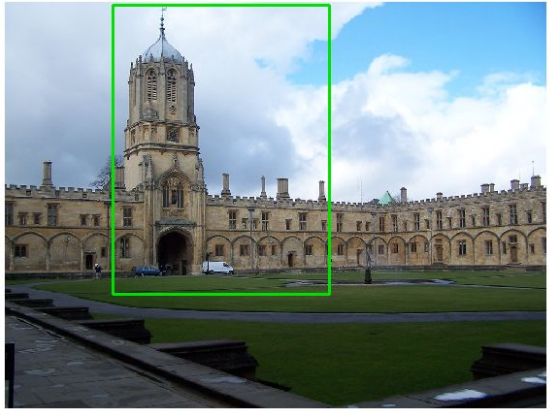


Oxford dataset

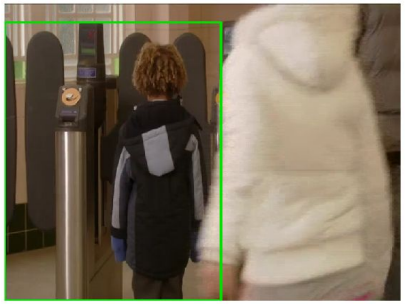








TRECVID dataset



INSTRE dataset