Computer vision - Retrieval

Ronan Sicre
Credits to Yannis Avrithis https://sif-dlv.github.io/

background

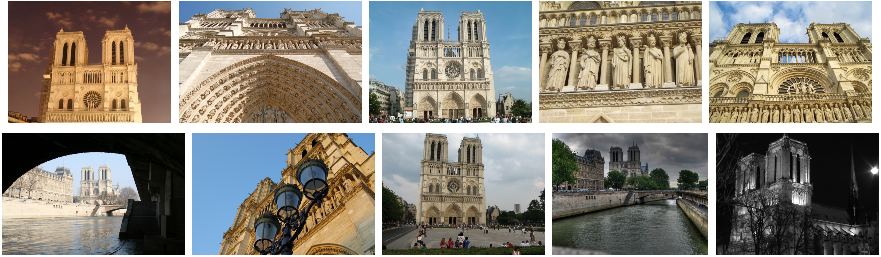
image classification challenges

- scale
- viewpoint
- occlusion
- clutter
- lighting
- number of instances
- texture/color
- pose
- deformability
- intra-class variability

image classification challenges

- scale
- viewpoint
- occlusion
- clutter
- lighting
- number of instances
- texture/color
- pose
- deformability
- intra-class variability

image retrieval challenges



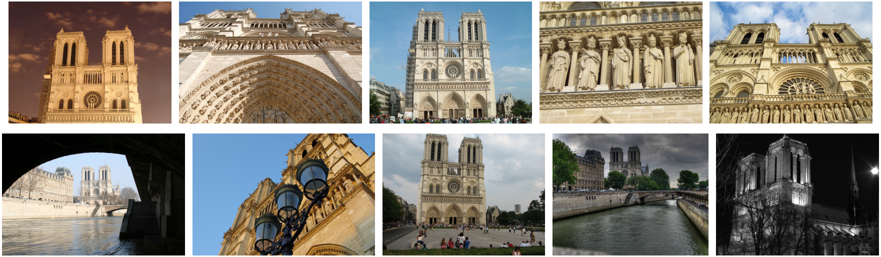
- scale
- viewpoint
- occlusion
- clutter
- lighting

- distinctiveness
- distractors

main difference to classification:

- no intra-class variability

image retrieval challenges



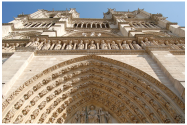
- scale
- viewpoint
- occlusion
- clutter
- lighting

- distinctiveness
- distractors

main difference to classification:

- no intra-class variability

image retrieval challenges







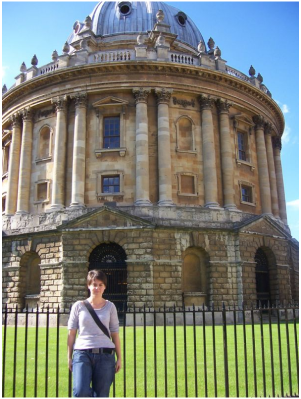
- scale
- viewpoint
- occlusion
- clutter
- lighting

- distinctiveness
- distractors

main difference to classification:

- no intra-class variability

vector quantization → visual words



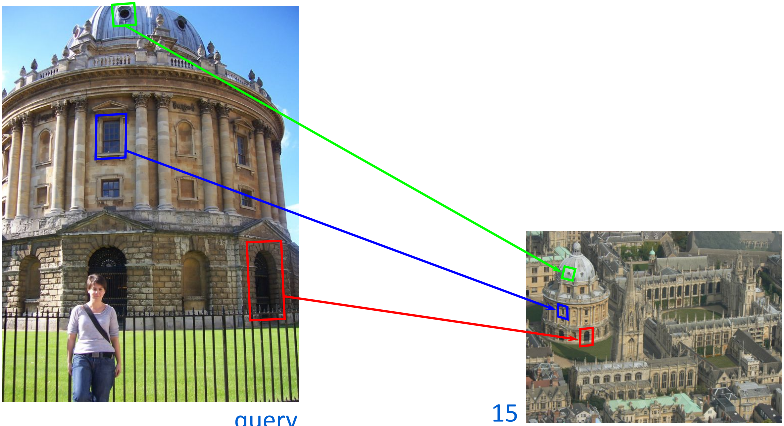
query

15 

- query *vs.* dataset image

Sivic and Zisserman. ICCV 2003. Video Google: A Text Retrieval Approach to Object Matching in videos.

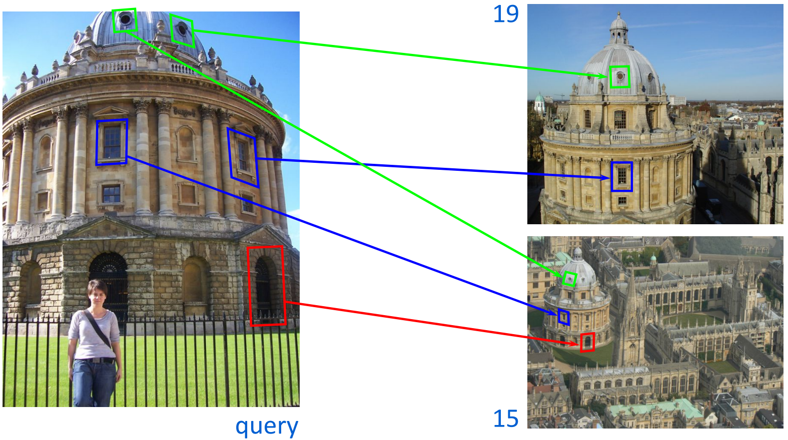
vector quantization $\rightarrow$ visual words



- pairwise descriptor matching

Sivic and Zisserman. ICCV 2003. Video Google: A Text Retrieval Approach to Object Matching in videos.

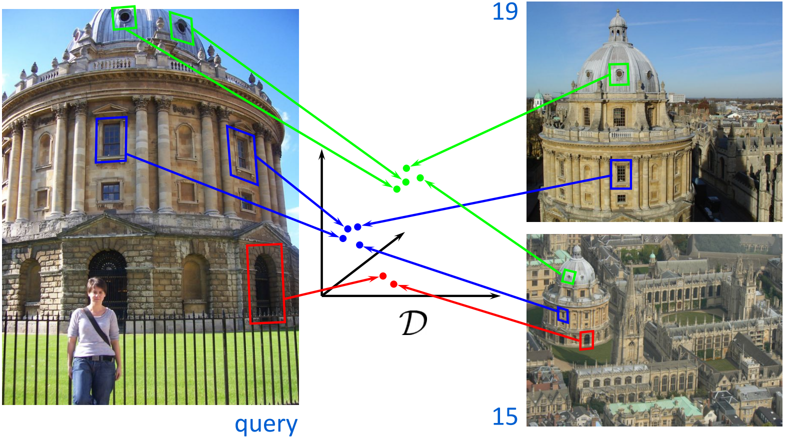
vector quantization → visual words



- pairwise descriptor matching for every dataset image

Sivic and Zisserman. ICCV 2003. Video Google: A Text Retrieval Approach to Object Matching in videos.

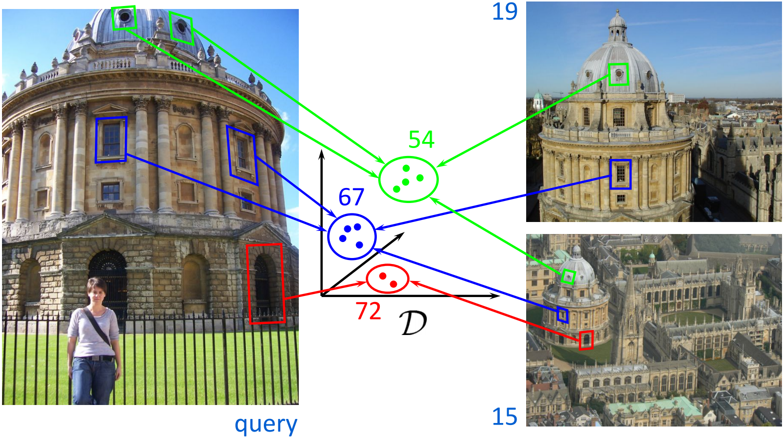
vector quantization → visual words



- similar descriptors should all be nearby in the descriptor space

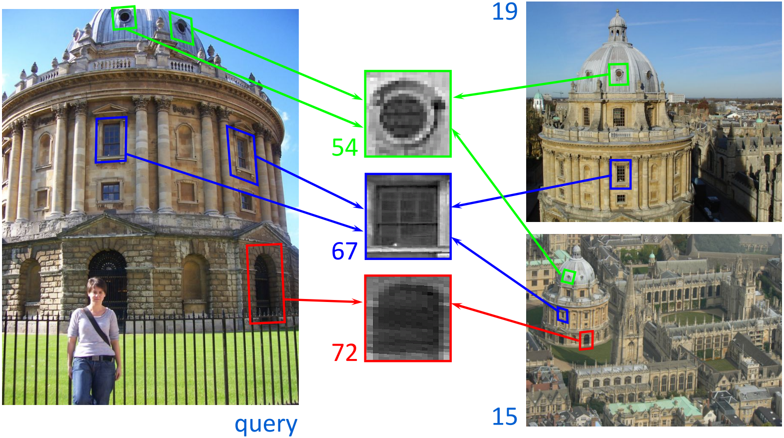
Sivic and Zisserman. ICCV 2003. Video Google: A Text Retrieval Approach to Object Matching in videos.

vector quantization → visual words



- let's quantize them into visual words

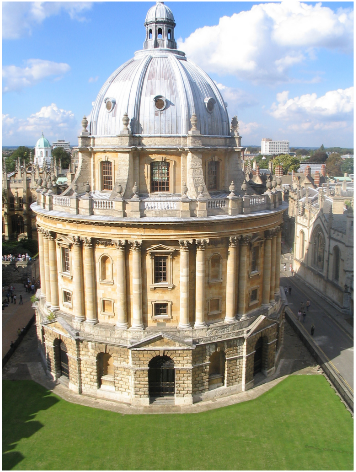
vector quantization → visual words



- now visual words act as a proxy; no pairwise matching needed

Sivic and Zisserman. ICCV 2003. Video Google: A Text Retrieval Approach to Object Matching in videos.

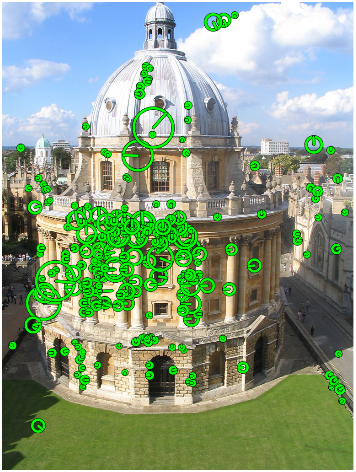
back to geometry: re-ranking




original images

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

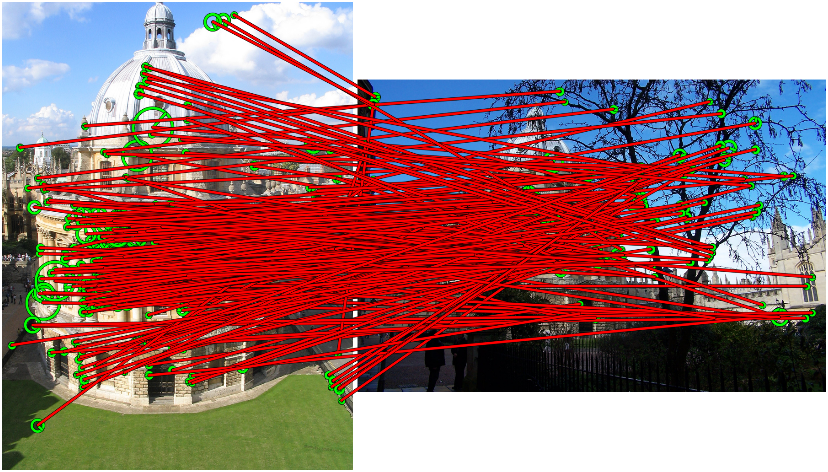
back to geometry: re-ranking




local features

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

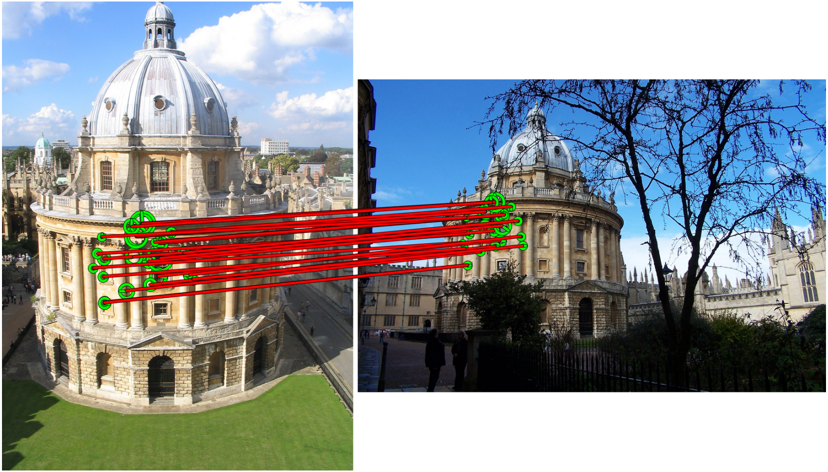
back to geometry: re-ranking



tentative correspondences: too many

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

back to geometry: re-ranking



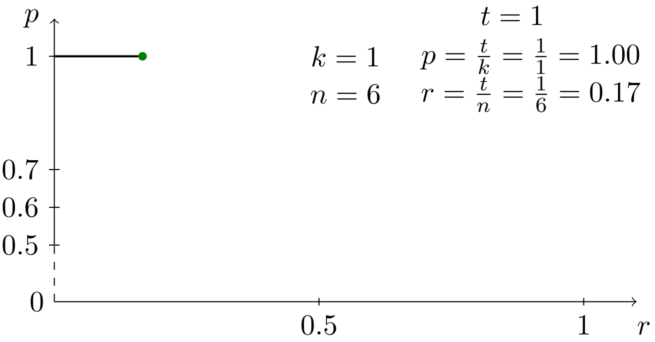
inliers: now more expensive to find

Fischler and Bolles. CACM 1981. Random Sample Consensus: A Paradigm for Model Fitting With Applications to Image Analysis and Automated Cartography.

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

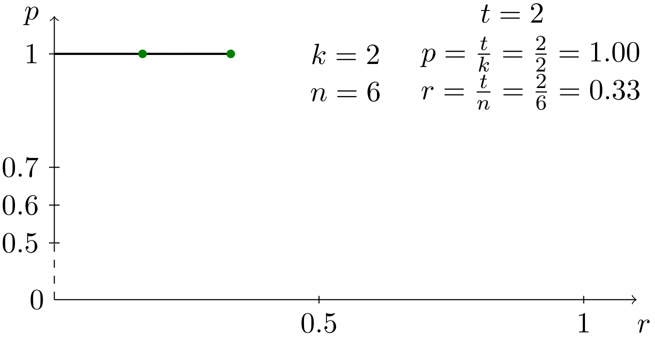


- \# total ground truth n, current rank k, \# true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

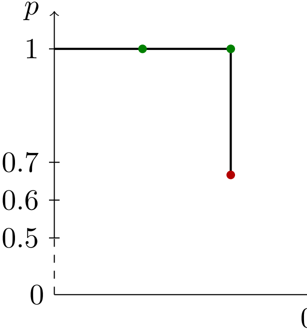


- # total ground truth n, current rank k, # true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F



$$t = 2$$

$$k = 3 \qquad p = \frac{t}{k} = \frac{2}{3} = 0.67$$

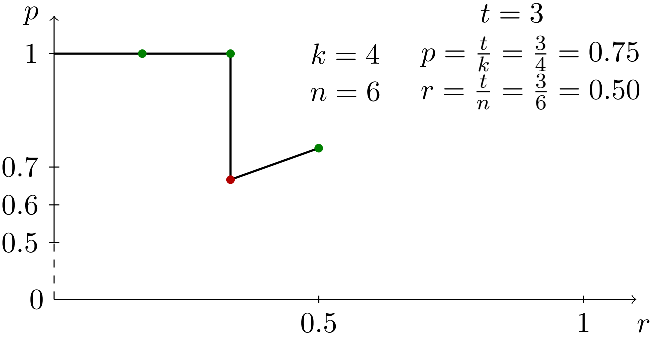
$$n = 6 \qquad r = \frac{t}{n} = \frac{2}{6} = 0.33$$

- $\#$ total ground truth n, current rank k, $\#$ true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

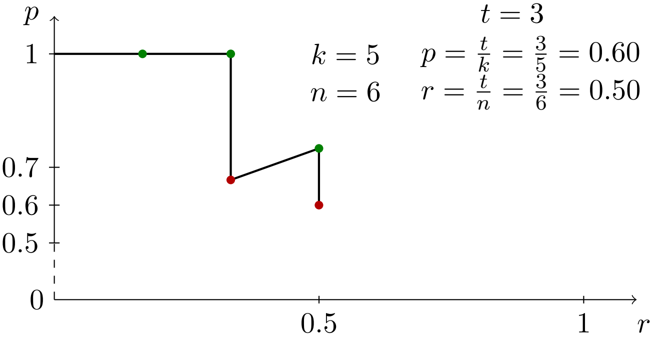


- \# total ground truth n, current rank k, # true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

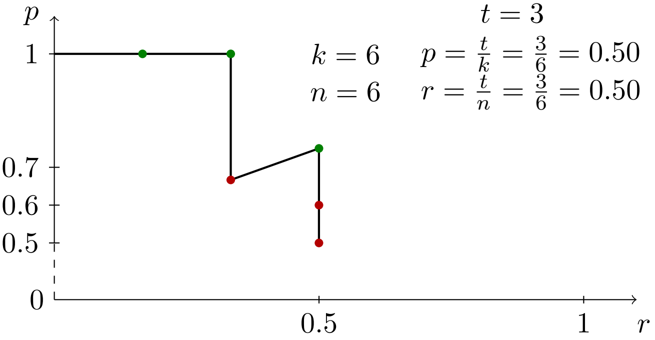


- \# total ground truth n, current rank k, \# true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

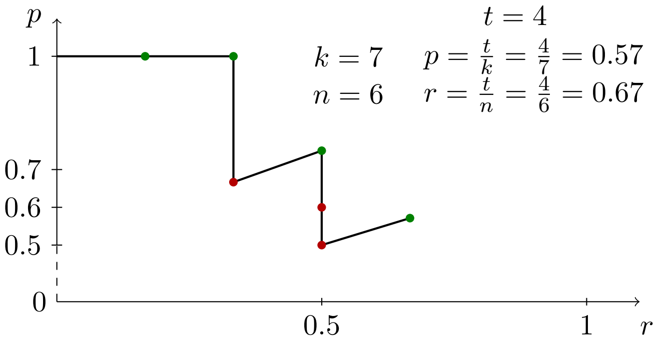


- $\#$ total ground truth n, current rank k, $\#$ true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

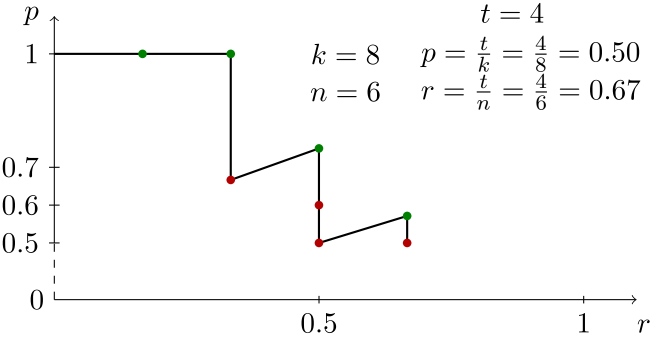


- # total ground truth n, current rank k, # true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

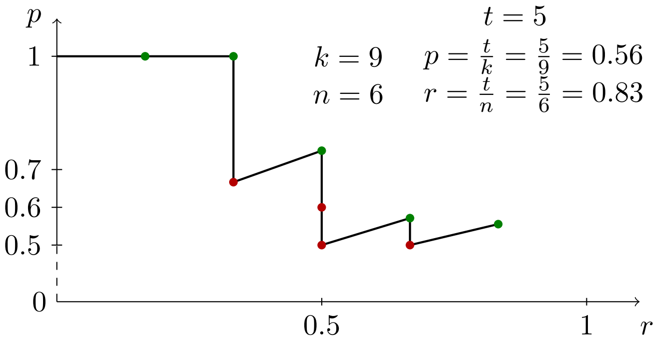


- \# total ground truth n, current rank k, # true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

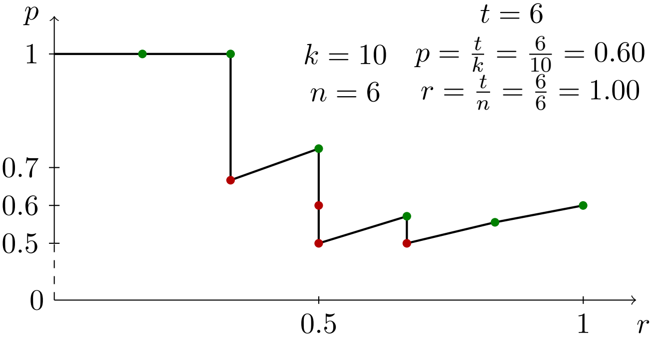


- $\#$ total ground truth n, current rank k, $\#$ true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

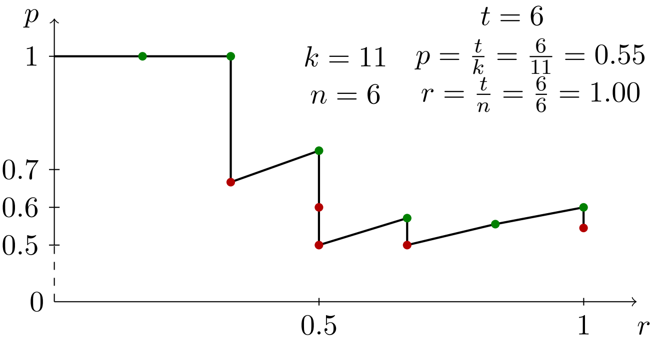


- \# total ground truth n, current rank k, # true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

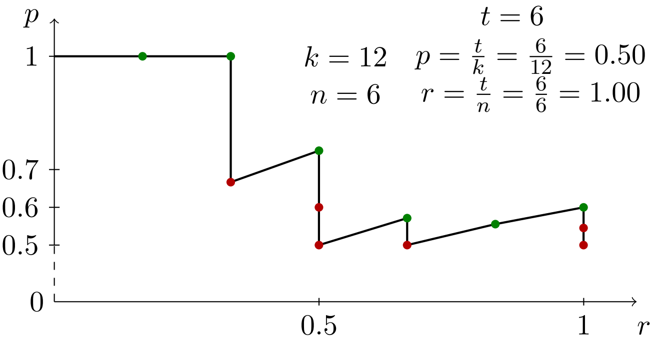


- \# total ground truth n, current rank k, \# true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

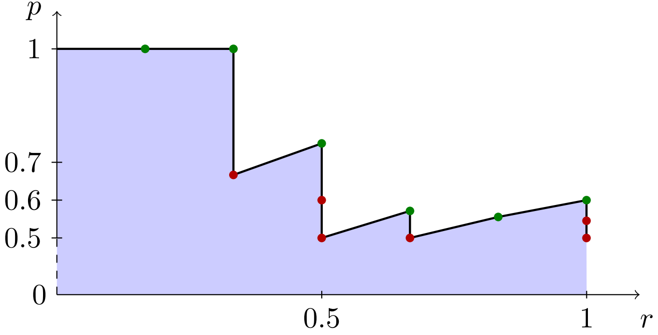


- $\#$ total ground truth n, current rank k, $\#$ true positives t
- precision $p = \frac{t}{k}$, recall $r = \frac{t}{n}$

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F

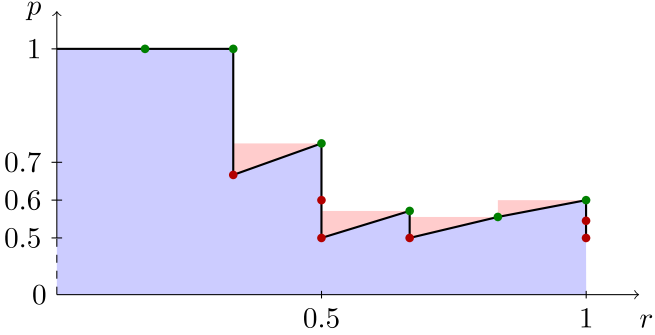


- average precision = area under curve
- the mean average precision (mAP) is the mean over queries

average precision (AP)

- ranked list of items with true/false labels

1	2	3	4	5	6	7	8	9	10	11	12
T	T	F	T	F	F	T	F	T	T	F	F



- average precision = area under curve (filled-in curve)
- the mean average precision (mAP) is the mean over queries

Oxford buildings dataset

[Philbin et al. 2007]


All Souls


Ashmolean


Balliol


Bodleian


Christ Church


Cornmarket


Hertford


Keble


Magdalen


Pitt Rivers

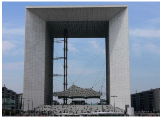

Radcliffe Camera

- Oxford5k: 5k images, 11 landmarks, $5 \times 11 = 55$ queries, $10 \sim 200$ positives/query
- Oxford105k: 100k additional distractor images

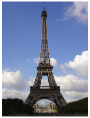
Philbin, Chum, Isard, Sivic and Zisserman. CVPR 2007. Object Retrieval With Large Vocabularies and Fast Spatial Matching.

Paris dataset

[Philbin et al. 2008]



Defense



Eiffel

Invalides

Louvre

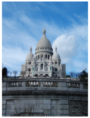
Moulin Rouge

Musée d'Orsay

Notre Dame

Pantheon

Pompidou



Sacré-Cœur

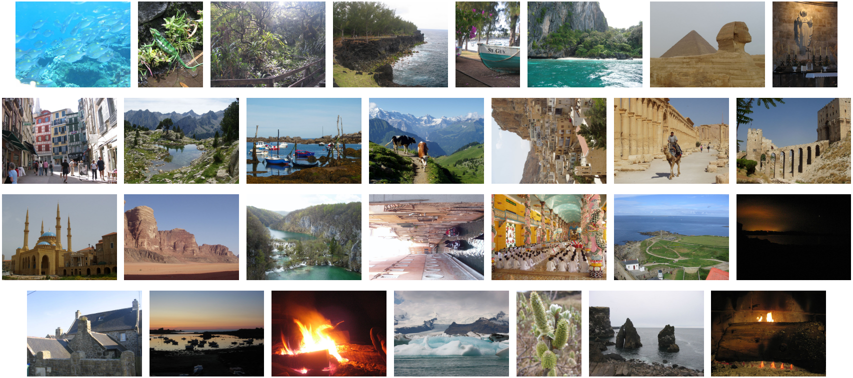
Triomphe

- Paris6k: 6k images, 11 landmarks, $5 \times 11 = 55$ queries, $50 \sim 300$ positives/query
- Paris106k: same 100k distractor images as Oxford

Philbin, Chum, Isard, Sivic and Zisserman. CVPR 2008. Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases.

Holidays dataset

[Jégou et al. 2008]



- personal holiday photos, natural and man-made scenes
- 1.5k images, 500 groups, 1 query/group, 1000 positives, $1 \sim 12$ positives/query

Jégou, Douze and Schmid. ECCV 2008. Hamming Embedding and Weak Geometric Consistency for Large Scale Image Search.

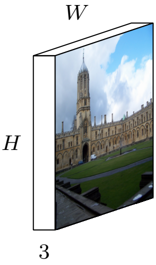
neural codes for image retrieval

- fine-tuning by softmax on 672 classes of 200k landmark photos
- outperforms VLAD and Fisher vectors on standard retrieval benchmarks, but still inferior to SIFT local descriptors

Babenko, Slesarev, Chigorin, Lempitsky. ECCV 2014. Neural Codes for Image Retrieval.

regional CNN features

[Razavian et al. 2015]

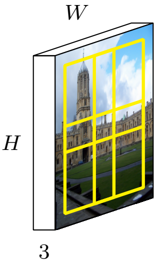


- 3-channel RGB input, largest square region extracted
- fixed multiscale overlapping regions, warped into $w \times h = 227 \times 227$
- each region yields a $w' \times h' \times k = 36 \times 36 \times 256$ dimensional feature at the last convolutional layer of AlexNet
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening of each descriptor

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

regional CNN features

[Razavian et al. 2015]

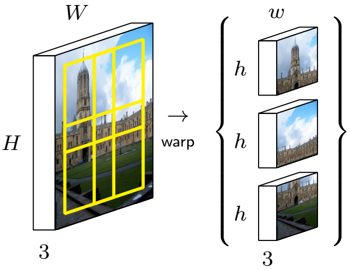


- 3-channel RGB input, largest square region extracted
- fixed multiscale overlapping regions, warped into $w \times h = 227 \times 227$
- each region yields a $w' \times h' \times k = 36 \times 36 \times 256$ dimensional feature at the last convolutional layer of AlexNet
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening of each descriptor

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

regional CNN features

[Razavian et al. 2015]

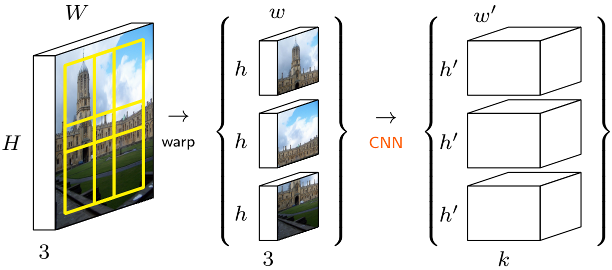


- 3-channel RGB input, largest square region extracted
- fixed multiscale overlapping regions, warped into $w \times h = 227 \times 227$
- each region yields a $w' \times h' \times k = 36 \times 36 \times 256$ dimensional feature at the last convolutional layer of AlexNet
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening of each descriptor

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

regional CNN features

[Razavian et al. 2015]

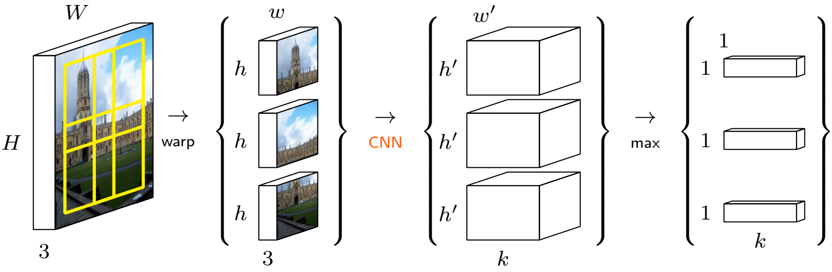


- 3-channel RGB input, largest square region extracted
- fixed multiscale overlapping regions, warped into $w \times h = 227 \times 227$
- each region yields a $w' \times h' \times k = 36 \times 36 \times 256$ dimensional feature at the last convolutional layer of AlexNet
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening of each descriptor

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

regional CNN features

[Razavian et al. 2015]

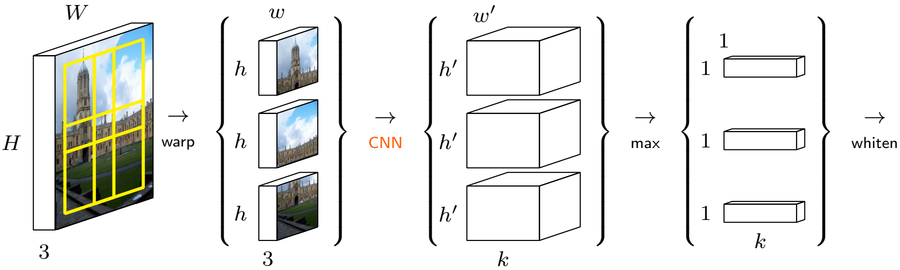


- 3-channel RGB input, largest square region extracted
- fixed multiscale overlapping regions, warped into $w \times h = 227 \times 227$
- each region yields a $w' \times h' \times k = 36 \times 36 \times 256$ dimensional feature at the last convolutional layer of AlexNet
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening of each descriptor

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

regional CNN features

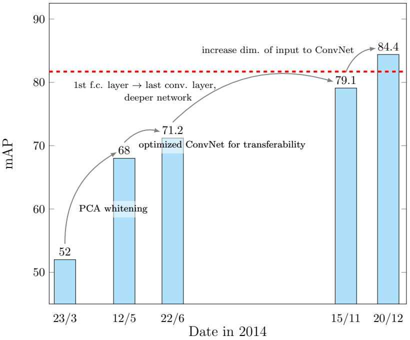
[Razavian et al. 2015]



- 3-channel RGB input, largest square region extracted
- fixed multiscale overlapping regions, warped into $w \times h = 227 \times 227$
- each region yields a $w' \times h' \times k = 36 \times 36 \times 256$ dimensional feature at the last convolutional layer of AlexNet
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening of each descriptor

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

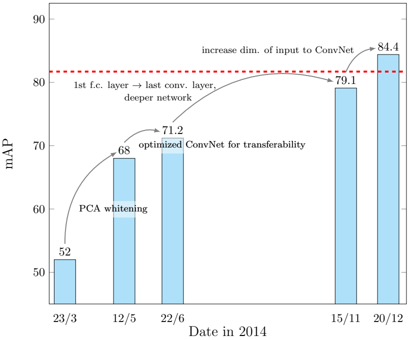
regional CNN features



- CNN visual representation jumps by more than 30% mAP to outperform standard SIFT pipeline in a few months
- however, this is based on multiple regional descriptors per image and exhaustive pairwise matching of all descriptors of query and all dataset images, which is not practical

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

regional CNN features

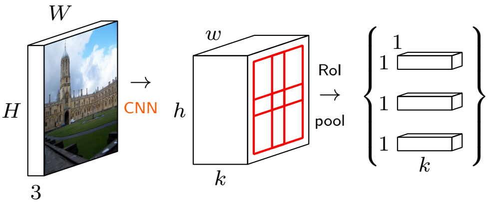


- CNN visual representation jumps by more than 30% mAP to outperform standard SIFT pipeline in a few months
- however, this is based on multiple regional descriptors per image and exhaustive pairwise matching of all descriptors of query and all dataset images, which is not practical

Razavian, Sullivan, Maki and Carlsson 2015. Visual Instance Retrieval with Deep Convolutional Networks.

regional max-pooling (R-MAC)

[Tolias et al. 2016]

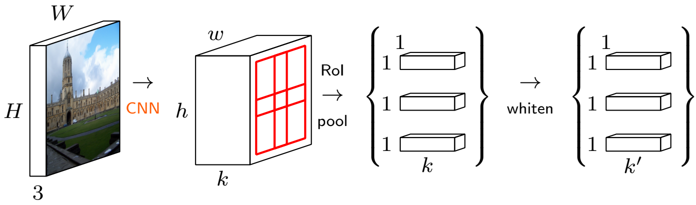


- VGG-16 last convolutional layer, $k = 512$
- fixed multiscale overlapping regions, spatial max-pooling
- ℓ_2-normalization, PCA-whitening, ℓ_2-normalization
- sum-pooling over all descriptors, ℓ_2-normalization

Tolias, Sicre and Jegou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

regional max-pooling (R-MAC)

[Tolias et al. 2016]

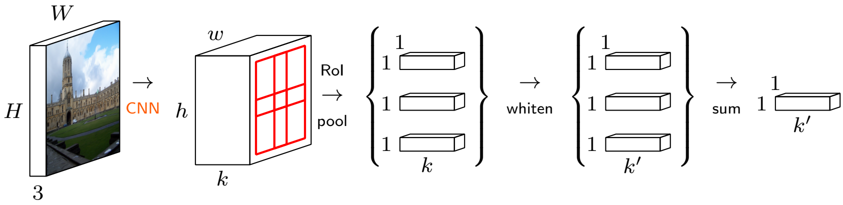


- VGG-16 last convolutional layer, $k = 512$
- fixed multiscale overlapping regions, spatial max-pooling
- ℓ_2-normalization, PCA-whitening, ℓ_2-normalization
- sum-pooling over all descriptors, ℓ_2-normalization

Tolias, Sicre and Jegou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

regional max-pooling (R-MAC)

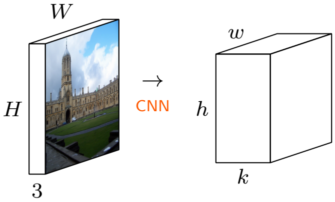
[Tolias et al. 2016]



- VGG-16 last convolutional layer, $k = 512$
- fixed multiscale overlapping regions, spatial max-pooling
- ℓ_2-normalization, PCA-whitening, ℓ_2-normalization
- sum-pooling over all descriptors, ℓ_2-normalization

Tolias, Sicre and Jegou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

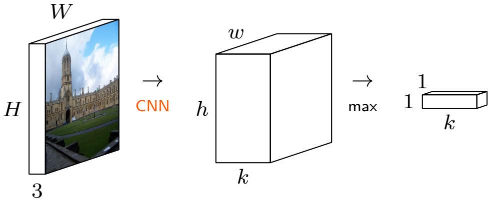
global max-pooling (MAC)



- VGG-16 last convolutional layer, $k = 512$
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening, ℓ_2-normalization
- MAC: maximum activation of convolutions

Tolias, Sicre and Jegou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

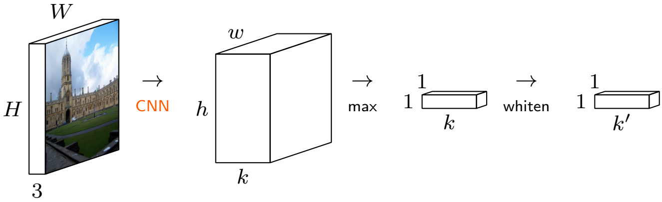
global max-pooling (MAC)



- VGG-16 last convolutional layer, $k = 512$
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening, ℓ_2-normalization
- MAC: maximum activation of convolutions

Tolias, Sicre and Jegou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

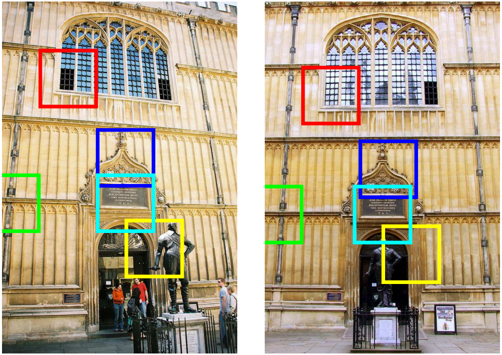
global max-pooling (MAC)



- VGG-16 last convolutional layer, $k = 512$
- global spatial max-pooling
- ℓ_2-normalization, PCA-whitening, ℓ_2-normalization
- MAC: maximum activation of convolutions

Tolias, Sicre and Jegou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

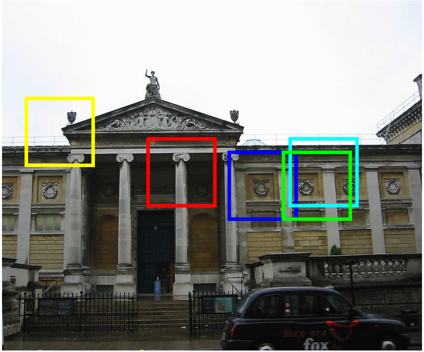
global max-pooling: matching

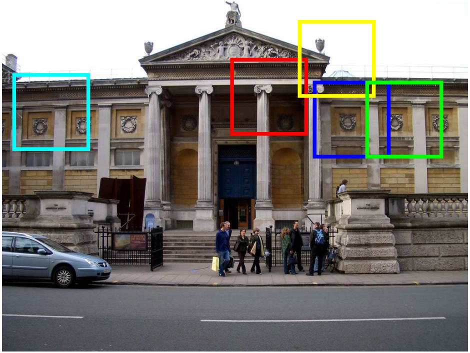


- receptive fields of 5 components of MAC vectors that contribute most to image similarity

Tolias, Sicre and Jégou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

global max-pooling: matching





- receptive fields of 5 components of MAC vectors that contribute most to image similarity

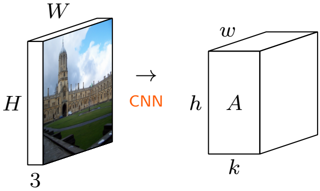
Tolias, Sicre and Jégou. ICLR 2016. Particular Object Retrieval with Integral Max-Pooling of CNN Activations.

global max-pooling: matching

- receptive fields of 5 components of MAC vectors that contribute most to image similarity

cross-dimensional weighting (CroW)

[Kalantidis et al. 2016]

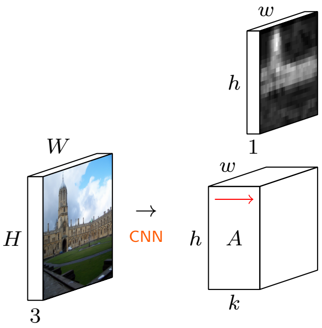


- VGG-16 feature map A, last pooling layer, $k = 512$
- spatial weights F, channel weights $\mathbf{w}$, weighted feature map
- global spatial sum-pooling
- ℓ_p-normalization, PCA-whitening, ℓ_2-normalization

Kalantidis, Mellina, Osindero. ECCVW 2016. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features.

cross-dimensional weighting (CroW)

[Kalantidis et al. 2016]

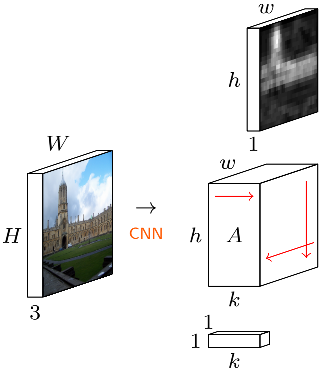


- VGG-16 feature map A, last pooling layer, $k = 512$
- spatial weights F, channel weights $\mathbf{w}$, weighted feature map
- global spatial sum-pooling
- ℓ_p-normalization, PCA-whitening, ℓ_2-normalization

Kalantidis, Mellina, Osindero. ECCVW 2016. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features.

cross-dimensional weighting (CroW)

[Kalantidis et al. 2016]

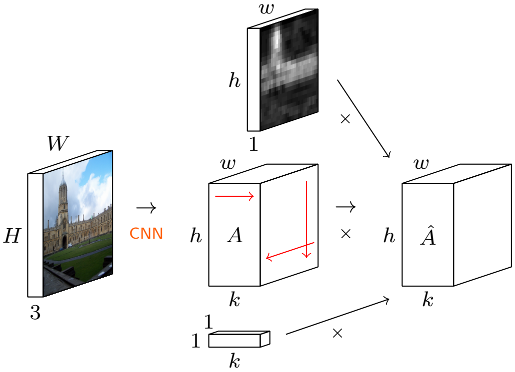


- VGG-16 feature map A, last pooling layer, $k = 512$
- spatial weights F, channel weights $\mathbf{w}$, weighted feature map
- global spatial sum-pooling
- ℓ_p-normalization, PCA-whitening, ℓ_2-normalization

Kalantidis, Mellina, Osindero. ECCVW 2016. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features.

cross-dimensional weighting (CroW)

[Kalantidis et al. 2016]

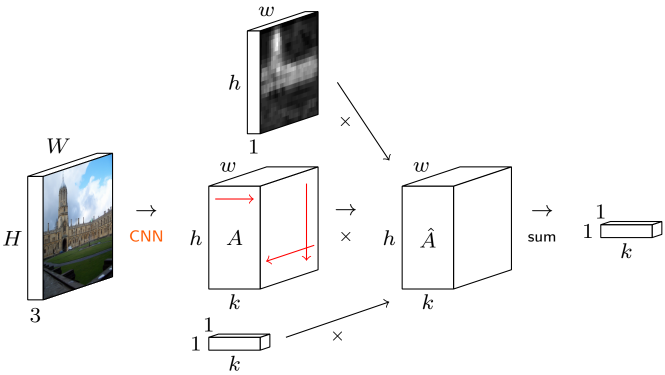


- VGG-16 feature map A, last pooling layer, $k = 512$
- spatial weights F, channel weights $\mathbf{w}$, weighted feature map
- global spatial sum-pooling
- ℓ_p-normalization, PCA-whitening, ℓ_2-normalization

Kalantidis, Mellina, Osindero. ECCVW 2016. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features.

cross-dimensional weighting (CroW)

[Kalantidis et al. 2016]

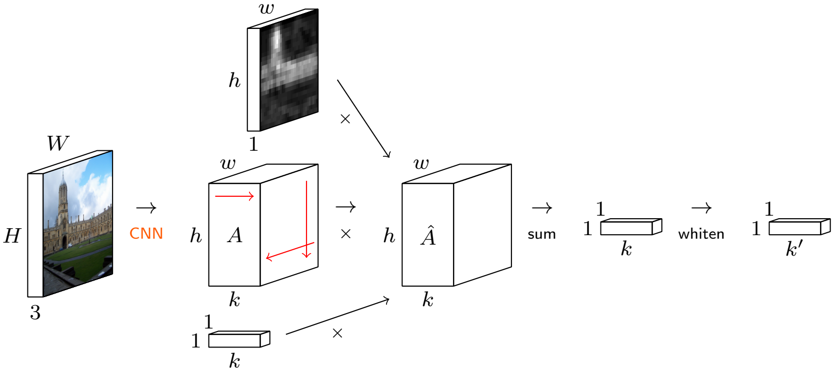


- VGG-16 feature map A, last pooling layer, $k = 512$
- spatial weights F, channel weights $\mathbf{w}$, weighted feature map
- global spatial sum-pooling
- ℓ_p-normalization, PCA-whitening, ℓ_2-normalization

Kalantidis, Mellina, Osindero. ECCVW 2016. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features.

cross-dimensional weighting (CroW)

[Kalantidis et al. 2016]



- VGG-16 feature map A, last pooling layer, $k = 512$
- spatial weights F, channel weights $\mathbf{w}$, weighted feature map
- global spatial sum-pooling
- ℓ_p-normalization, PCA-whitening, ℓ_2-normalization

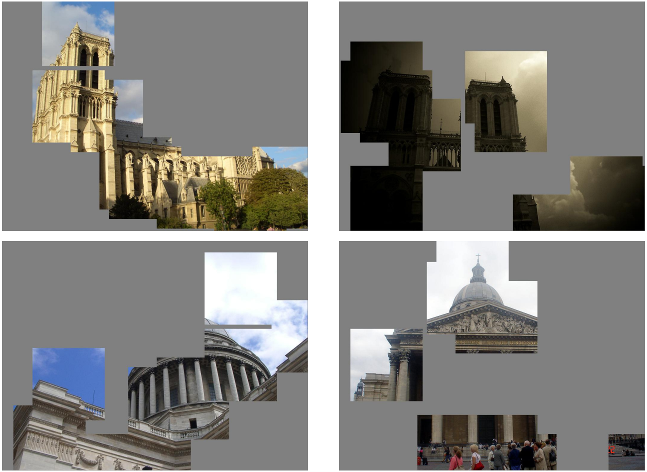
Kalantidis, Mellina, Osindero. ECCVW 2016. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features.

cross-dimensional weighting (CroW)

- input image

Kalantidis, Mellina, Osindero. ECCVW 2016. Cross-Dimensional Weighting for Aggregated Deep Convolutional Features.

cross-dimensional weighting (CroW)



- receptive fields of nonzero elements of the 10 channels with the highest sparsity-sensitive weights

manifold learning

siamese architecture

[Chopra et al. 2005]

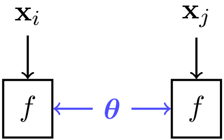
$\mathbf{x}_i$ $\mathbf{x}_j$

- an input sample is a pair $(\mathbf{x}_i, \mathbf{x}_j)$
- both $\mathbf{x}_i, \mathbf{x}_j$ go through the same function f with shared parameters $\boldsymbol{\theta}$
- loss ℓ_{ij} is measured on output pair $(\mathbf{y}_i, \mathbf{y}_j)$ and target t_{ij}

Chopra, Hadsell, Lecun, CVPR 2005. Learning a Similarity Metric Discriminatively, with Application to Face Verification.

siamese architecture

[Chopra et al. 2005]

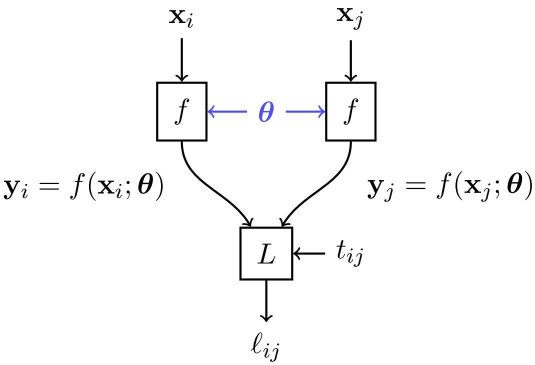


- an input sample is a pair $(\mathbf{x}_i, \mathbf{x}_j)$
- both $\mathbf{x}_i, \mathbf{x}_j$ go through the same function f with shared parameters $\boldsymbol{\theta}$
- loss ℓ_{ij} is measured on output pair $(\mathbf{y}_i, \mathbf{y}_j)$ and target t_{ij}

Chopra, Hadsell, Lecun, CVPR 2005. Learning a Similarity Metric Discriminatively, with Application to Face Verification.

siamese architecture

[Chopra et al. 2005]

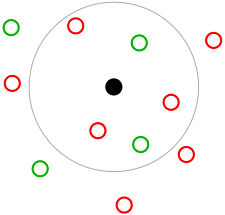


- an input sample is a pair $(\mathbf{x}_i, \mathbf{x}_j)$
- both $\mathbf{x}_i, \mathbf{x}_j$ go through the same function f with shared parameters $\boldsymbol{\theta}$
- loss ℓ_{ij} is measured on output pair $(\mathbf{y}_i, \mathbf{y}_j)$ and target t_{ij}

Chopra, Hadsell, Lecun, CVPR 2005. Learning a Similarity Metric Discriminatively, with Application to Face Verification.

contrastive loss

[Hadsel et al. 2006]



- input samples $\mathbf{x}_i$, output vectors $\mathbf{y}_i = f(\mathbf{x}_i; \boldsymbol{\theta})$
- target variables $t_{ij} = \mathbb{1}[\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j)]$
- contrastive loss is a function of distance $\|\mathbf{y}_i - \mathbf{y}_j\|$ only

$$\ell_{ij} = L((\mathbf{y}_i, \mathbf{y}_j), t_{ij}) = \ell(\|\mathbf{y}_i - \mathbf{y}_j\|, t_{ij})$$

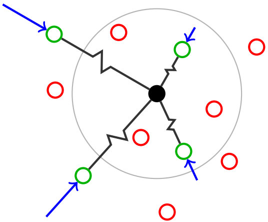
- similar samples are attracted

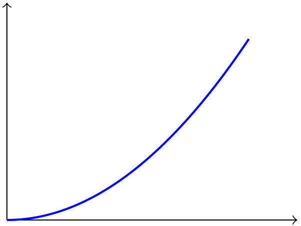
$$\ell(x,t) = t\ell^+(x) + (1-t)\ell^-(x) = tx^2 + (1-t)[m-x]_+^2$$

Hadsell, Chopra, Lecun. CVPR 2006. Dimensionality Reduction By Learning an Invariant Mapping.

contrastive loss

[Hadsel et al. 2006]





- input samples $\mathbf{x}_i$, output vectors $\mathbf{y}_i = f(\mathbf{x}_i; \boldsymbol{\theta})$
- target variables $t_{ij} = \mathbb{1}[\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j)]$
- contrastive loss is a function of distance $\|\mathbf{y}_i - \mathbf{y}_j\|$ only

$$\ell_{ij} = L((\mathbf{y}_i, \mathbf{y}_j), t_{ij}) = \ell(\|\mathbf{y}_i - \mathbf{y}_j\|, t_{ij})$$

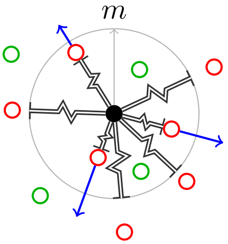
- similar samples are attracted

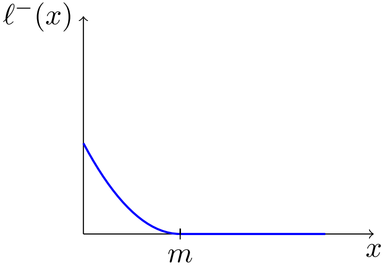
$$\ell(x, t) = \boxed{t\ell^+(x)} + (1-t)\ell^-(x) = \boxed{tx^2} + (1-t)[m-x]_+^2$$

Hadsell, Chopra, Lecun. CVPR 2006. Dimensionality Reduction By Learning an Invariant Mapping.

contrastive loss

[Hadsel et al. 2006]





- input samples $\mathbf{x}_i$, output vectors $\mathbf{y}_i = f(\mathbf{x}_i; \boldsymbol{\theta})$
- target variables $t_{ij} = \mathbb{1}[\mathrm{sim}(\mathbf{x}_i, \mathbf{x}_j)]$
- contrastive loss is a function of distance $\|\mathbf{y}_i - \mathbf{y}_j\|$ only

$$\ell_{ij} = L((\mathbf{y}_i, \mathbf{y}_j), t_{ij}) = \ell(\|\mathbf{y}_i - \mathbf{y}_j\|, t_{ij})$$

- dissimilar samples are repelled if closer than margin m

$$\ell(x, t) = t\ell^+(x) + (1-t)\boxed{\ell^-(x)} = tx^2 + (1-t)\boxed{[m-x]_+^2}$$

Hadsell, Chopra, Lecun. CVPR 2006. Dimensionality Reduction By Learning an Invariant Mapping.

triplet architecture

[Wang et al. 2014]

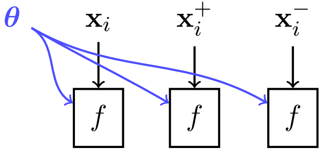
$\mathbf{x}_i$ $\mathbf{x}_i^+$ $\mathbf{x}_i^-$

- an input sample is a triplet $(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-)$
- $\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-$ go through the same function f with shared parameters $\boldsymbol{\theta}$
- loss ℓ_i measured on output triplet $(\mathbf{y}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)$

Wang, Song, Leung, Rosenberg, Wang, Philbin, Chen, Wu. CVPR 2014. Learning Fine-Grained Image Similarity with Deep Ranking.

triplet architecture

[Wang et al. 2014]

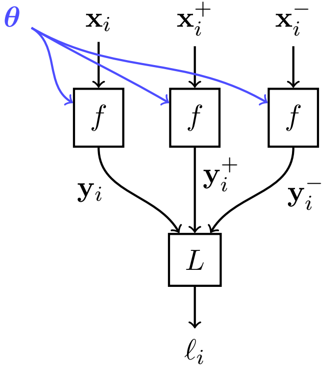


- an input sample is a triplet $(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-)$
- $\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-$ go through the same function f with shared parameters $\boldsymbol{\theta}$
- loss ℓ_i measured on output triplet $(\mathbf{y}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)$

Wang, Song, Leung, Rosenberg, Wang, Philbin, Chen, Wu. CVPR 2014. Learning Fine-Grained Image Similarity with Deep Ranking.

triplet architecture

[Wang et al. 2014]



- an input sample is a triplet $(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-)$
- $\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-$ go through the same function f with shared parameters $\boldsymbol{\theta}$
- loss ℓ_i measured on output triplet $(\mathbf{y}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)$

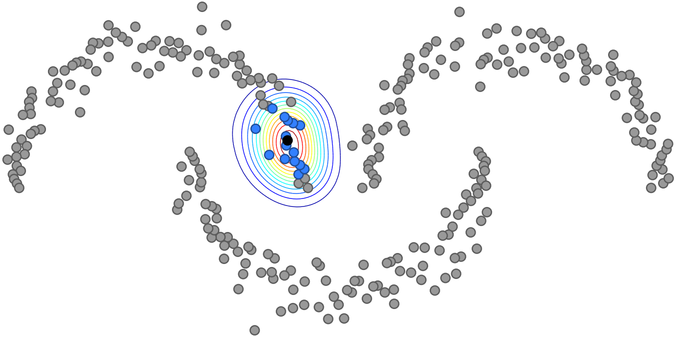
Wang, Song, Leung, Rosenberg, Wang, Philbin, Chen, Wu. CVPR 2014. Learning Fine-Grained Image Similarity with Deep Ranking.

graph-based methods

ranking on manifolds: single query

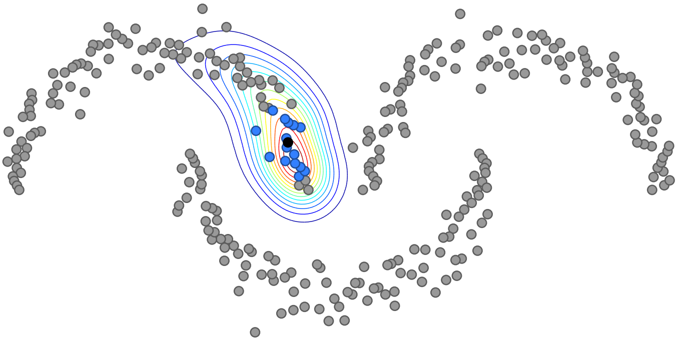
- data points (•), query point (•), nearest neighbors (•)
- iteration $\times 30$

ranking on manifolds: single query



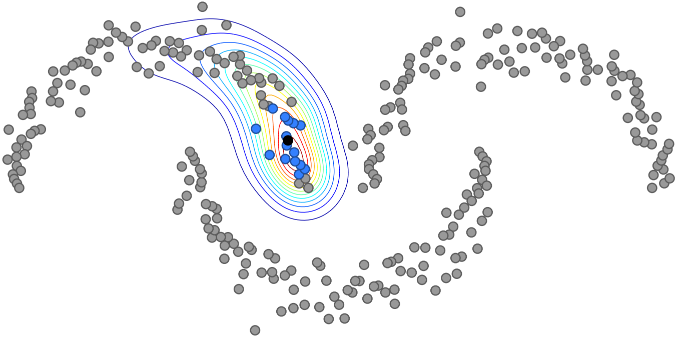
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 0×30

ranking on manifolds: single query



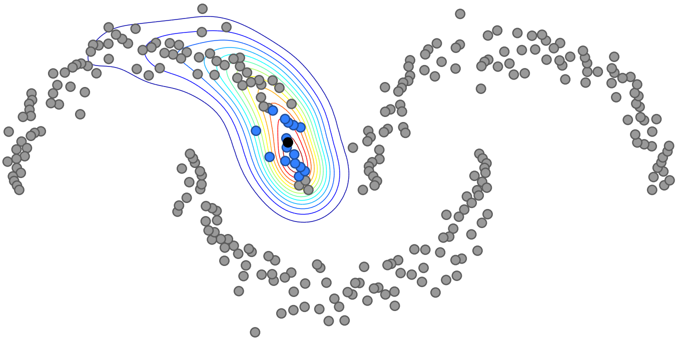
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 1×30

ranking on manifolds: single query



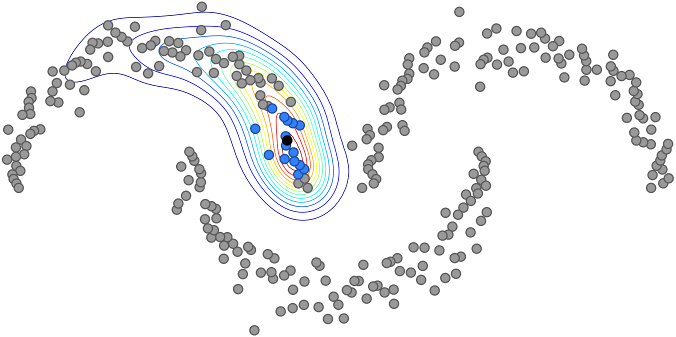
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 2×30

ranking on manifolds: single query



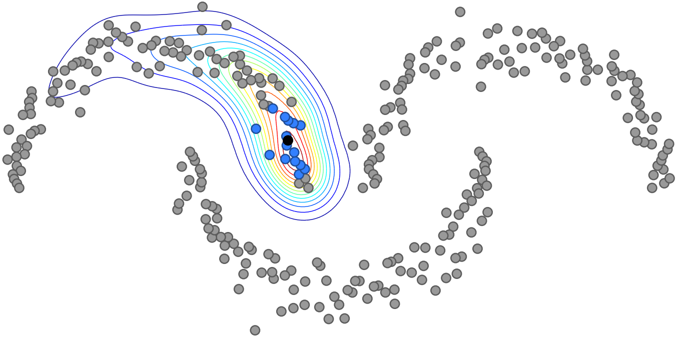
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 3×30

ranking on manifolds: single query



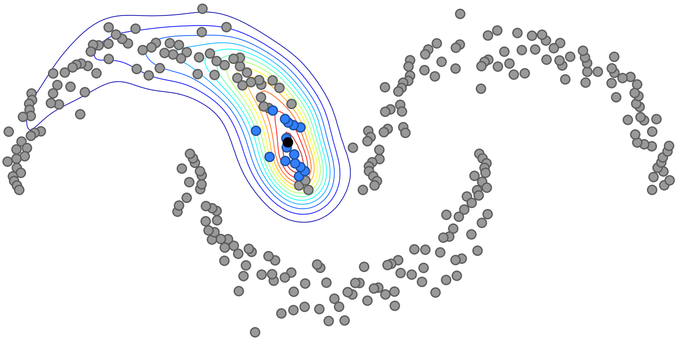
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 4×30

ranking on manifolds: single query



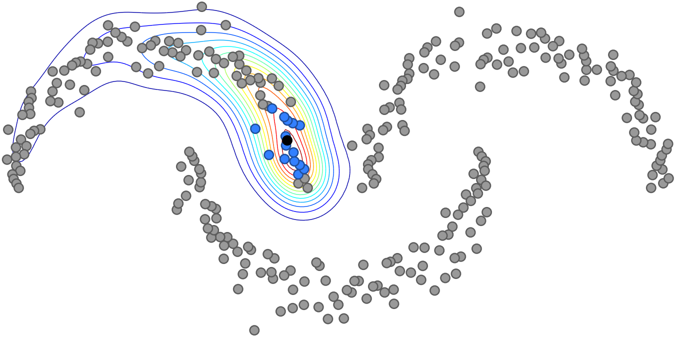
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 5×30

ranking on manifolds: single query



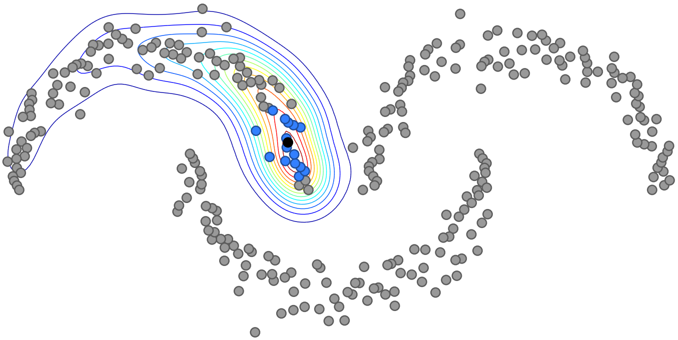
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 6×30

ranking on manifolds: single query



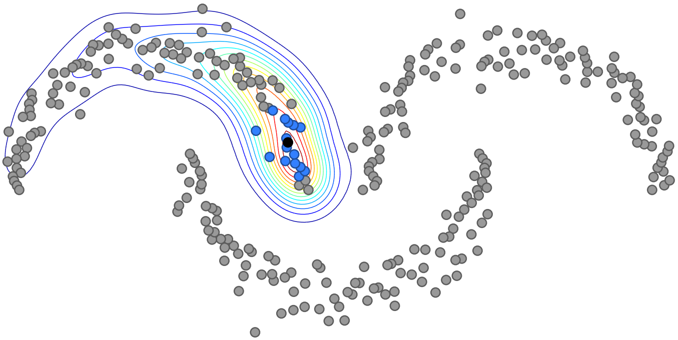
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 7×30

ranking on manifolds: single query



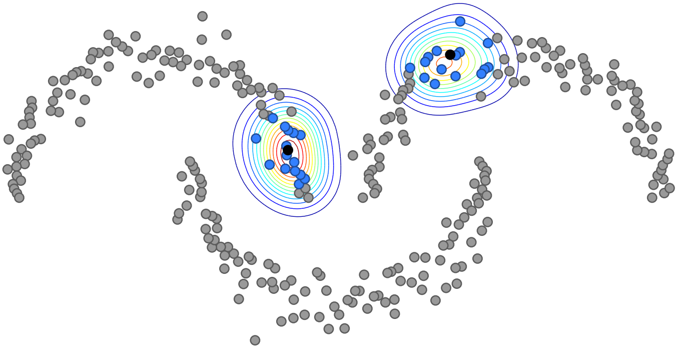
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 8×30

ranking on manifolds: single query



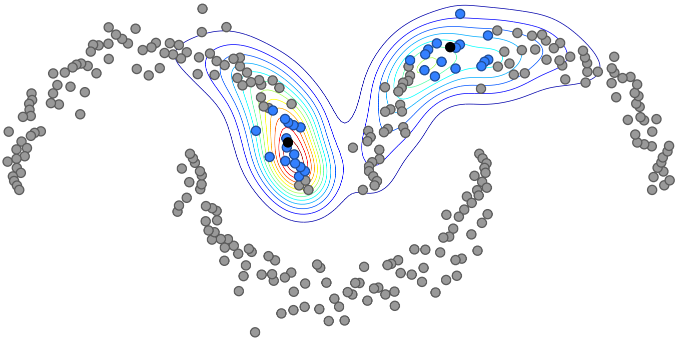
- data points (∘), query point (•), nearest neighbors (∘)
- iteration 9×30

ranking on manifolds: multiple queries



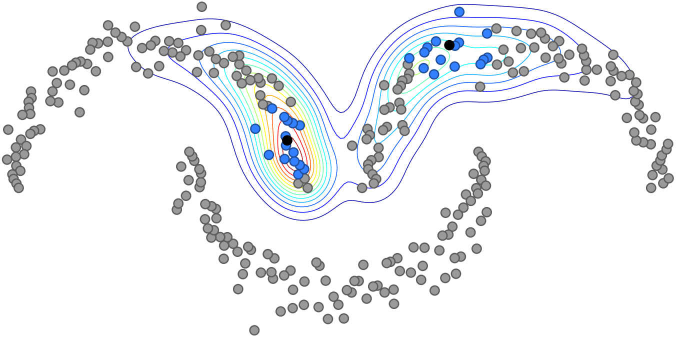
- data points (∘), query points (•), nearest neighbors (∘)
- iteration 0×30

ranking on manifolds: multiple queries



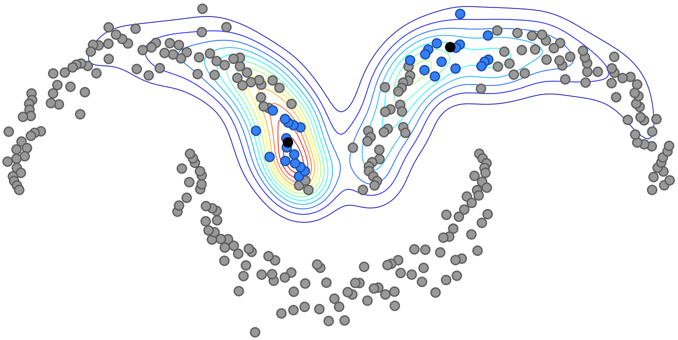
- data points (∘), query points (•), nearest neighbors (∘)
- iteration 1×30

ranking on manifolds: multiple queries



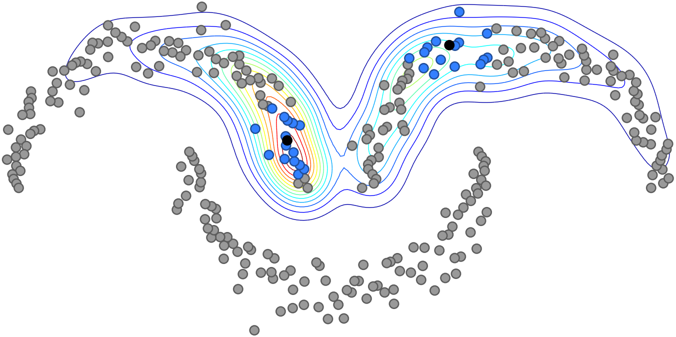
- data points (∘), query points (•), nearest neighbors (∘)
- iteration 2×30

ranking on manifolds: multiple queries



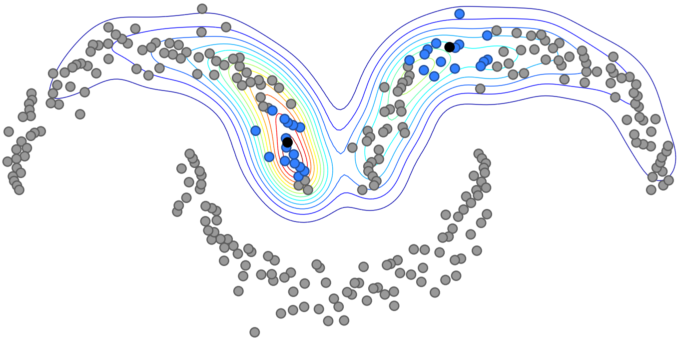
- data points (∘), query points (•), nearest neighbors (•)
- iteration 3×30

ranking on manifolds: multiple queries



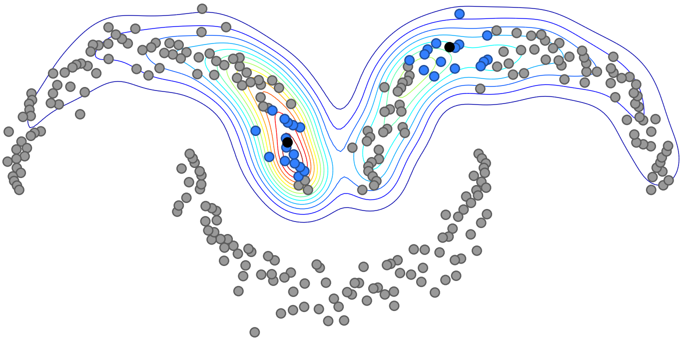
- data points (∘), query points (•), nearest neighbors (∘)
- iteration 4×30

ranking on manifolds: multiple queries



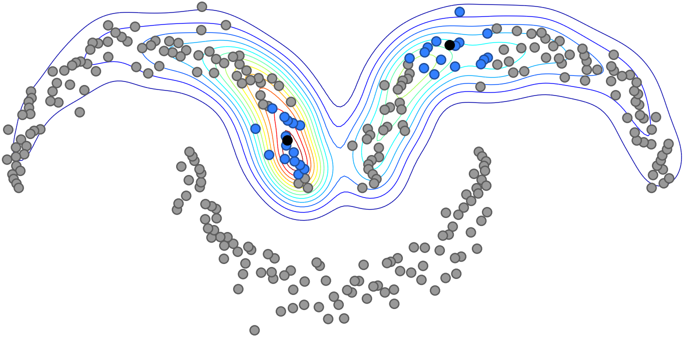
- data points (∘), query points (•), nearest neighbors (∘)
- iteration 5×30

ranking on manifolds: multiple queries



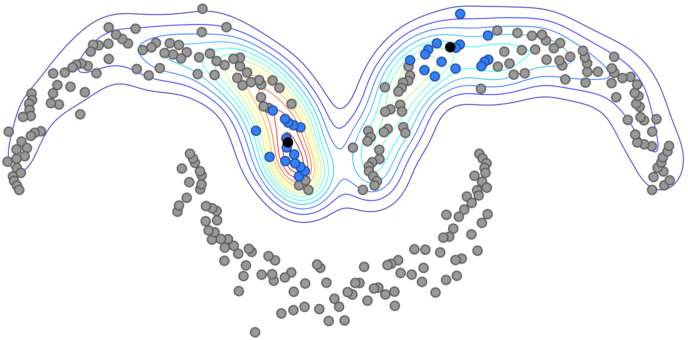
- data points (∘), query points (•), nearest neighbors (∘)
- iteration 6×30

ranking on manifolds: multiple queries



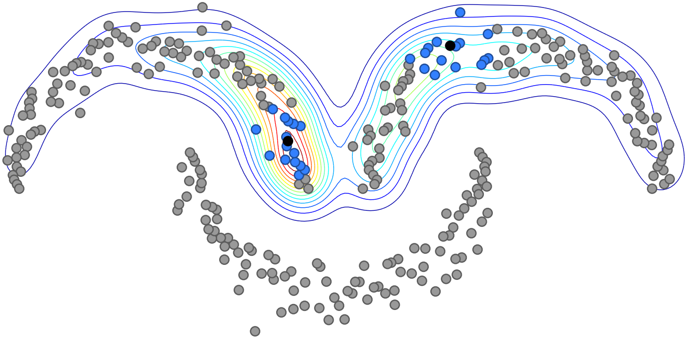
- data points (∘), query points (•), nearest neighbors (∘)
- iteration 7×30

ranking on manifolds: multiple queries



- data points (∘), query points (•), nearest neighbors (∘)
- iteration 8×30

ranking on manifolds: multiple queries



- data points (∘), query points (•), nearest neighbors (∘)
- iteration 9×30

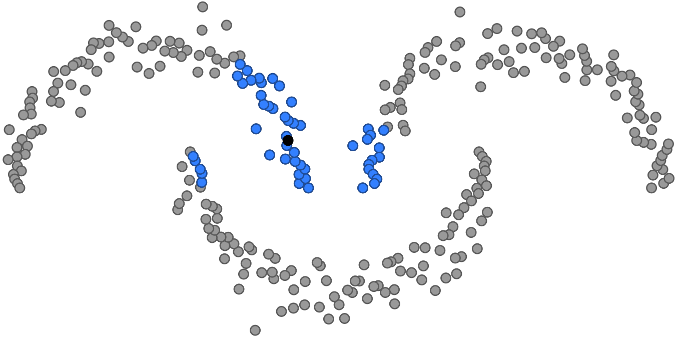
mining on manifolds

[Iscen et al. 2018]

- data points (∘), query point $\mathbf{x}$ (•)

mining on manifolds

[Iscen et al. 2018]



- data points (∘), query point $\mathbf{x}$ (•)
- Euclidean nearest neighbors $E(\mathbf{x})$ (∘)

mining on manifolds

[Iscen et al. 2018]

- data points ($\circ$), query point $\mathbf{x}$ ($\bullet$)
- manifold nearest neighbors $M(\mathbf{x})$ ($\circ$)

mining on manifolds

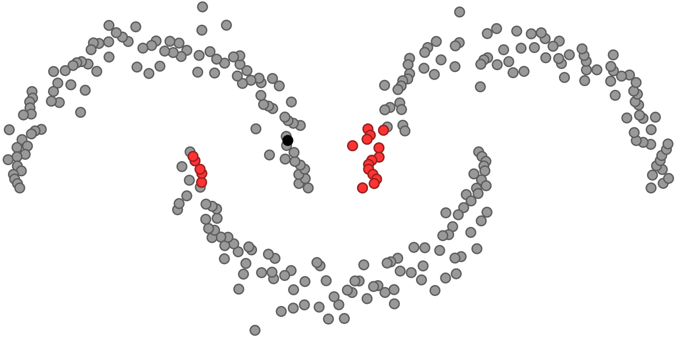
[Iscen et al. 2018]

- data points (∘), query point $\mathbf{x}$ (•)
- hard positives $S^+ = M(\mathbf{x}) \setminus E(\mathbf{x})$ (∘)

Iscen, Tolias, Avrithis and Chum. 2018 (unpublished). Mining on Manifolds: Metric Learning without Labels.

mining on manifolds

[Iscen et al. 2018]



- data points (∘), query point $\mathbf{x}$ (•)
- hard negatives $S^{-} = E(\mathbf{x}) \setminus M(\mathbf{x})$ (•)

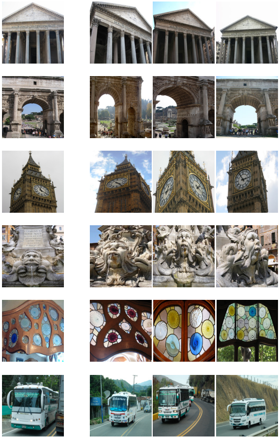
Iscen, Tolias, Avrithis and Chum. 2018 (unpublished). Mining on Manifolds: Metric Learning without Labels.

mining on manifolds

- query (anchor) $(\mathbf{x})$
- positives $S^+(\mathbf{x})$ *vs.* Euclidean neighbors $E(\mathbf{x})$
- negatives $S^-(\mathbf{x})$ *vs.* Euclidean non-neighbors $X \setminus E(\mathbf{x})$

Iscen, Tolias, Avrithis and Chum. 2018 (unpublished). Mining on Manifolds: Metric Learning without Labels.

mining on manifolds



- query (anchor) $(\mathbf{x})$
- positives $S^+(\mathbf{x})$ *vs.* Euclidean neighbors $E(\mathbf{x})$
- negatives $S^-(\mathbf{x})$ *vs.* Euclidean non-neighbors $X \setminus E(\mathbf{x})$

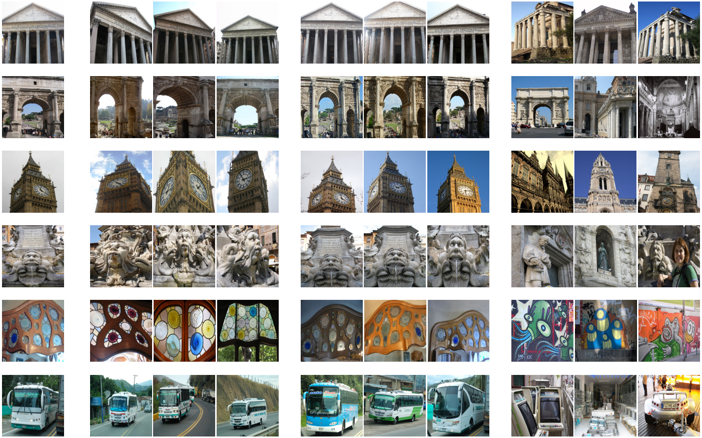
Iscen, Tolias, Avrithis and Chum. 2018 (unpublished). Mining on Manifolds: Metric Learning without Labels.

mining on manifolds

- query (anchor) $(\mathbf{x})$
- positives $S^+(\mathbf{x})$ *vs.* Euclidean neighbors $E(\mathbf{x})$
- negatives $S^-(\mathbf{x})$ *vs.* Euclidean non-neighbors $X \setminus E(\mathbf{x})$

Iscen, Tolias, Avrithis and Chum. 2018 (unpublished). Mining on Manifolds: Metric Learning without Labels.

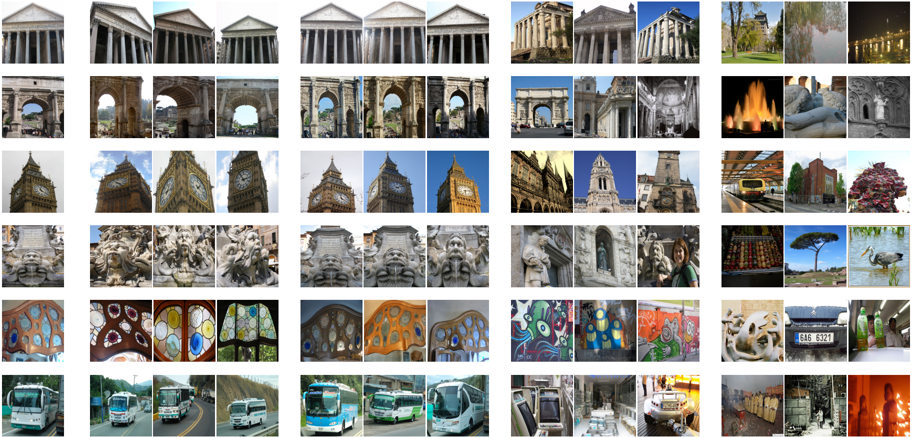
mining on manifolds



- query (anchor) $(\mathbf{x})$
- positives $S^{+}(\mathbf{x})$ *vs.* Euclidean neighbors $E(\mathbf{x})$
- negatives $S^{-}(\mathbf{x})$ *vs.* Euclidean non-neighbors $X \setminus E(\mathbf{x})$

Iscen, Tolias, Avrithis and Chum. 2018 (unpublished). Mining on Manifolds: Metric Learning without Labels.

mining on manifolds



- query (anchor) $(\mathbf{x})$
- positives $S^+(\mathbf{x})$ *vs.* Euclidean neighbors $E(\mathbf{x})$
- negatives $S^-(\mathbf{x})$ *vs.* Euclidean non-neighbors $X \setminus E(\mathbf{x})$

Iscen, Tolias, Avrithis and Chum. 2018 (unpublished). Mining on Manifolds: Metric Learning without Labels.

Conclusion

Features and embeddings
Feature matching, geometric verification
mean Average Precision
Indexing, and approximate neighbor search
deep representation
contrastive loss
manifold learning