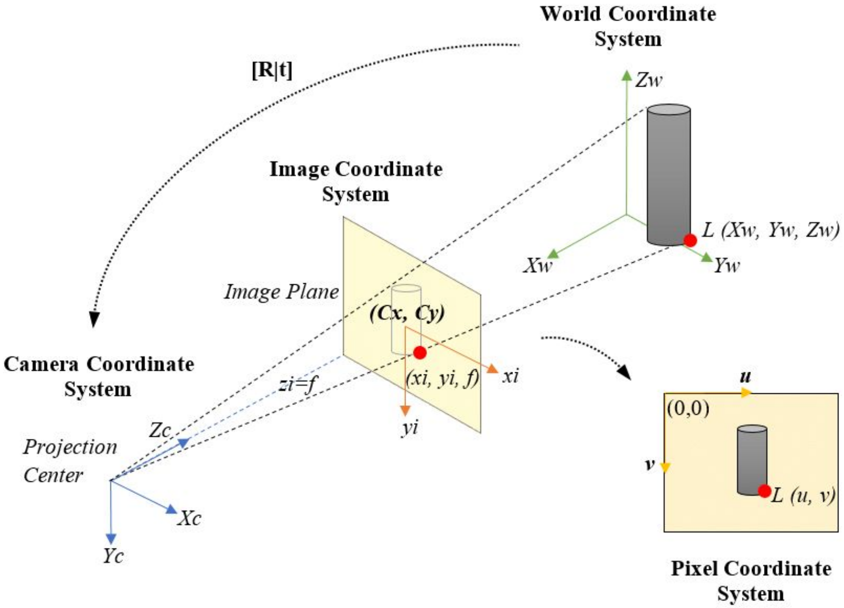
Vector search #3 – Image embeddings

Princeton COS579A – Edo Liberty and Matthijs Douze – 2023-09-22

Images

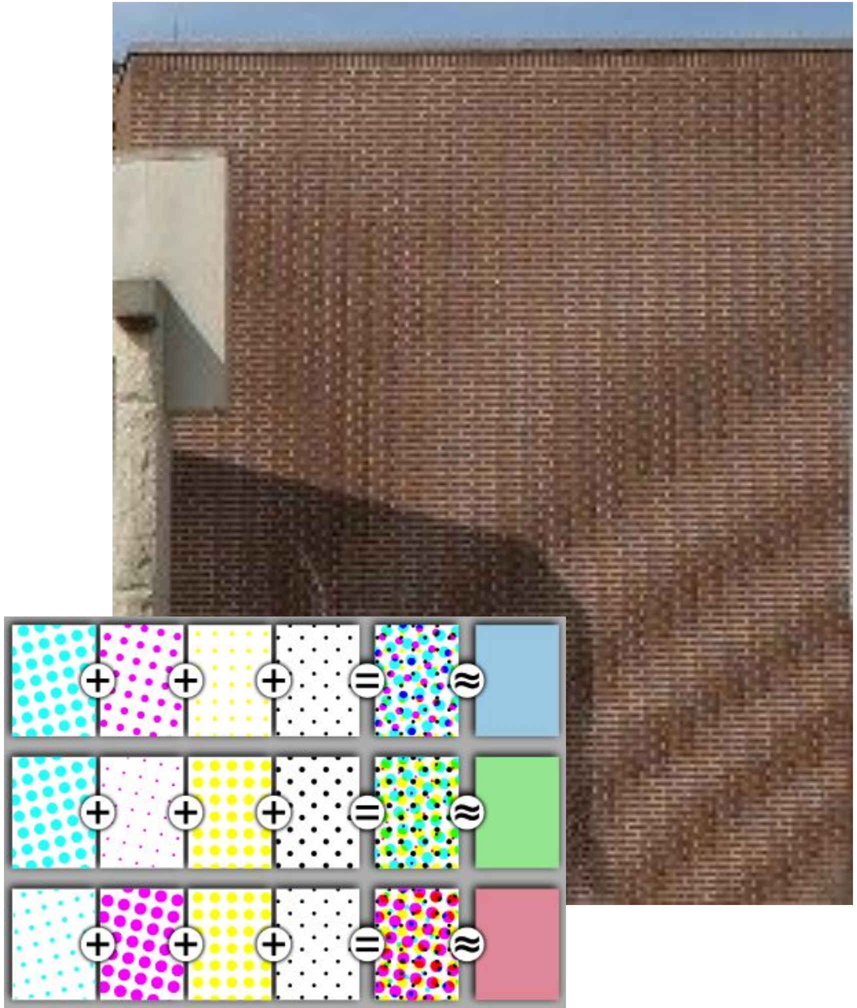
Acquisition process

- Image = approximation of Continuous signal
  - Unlike text
  - Low semantic level
- Convert to digital representation
  - After optics…
- Discretization:
  - Rasterization sampling
  - quantization



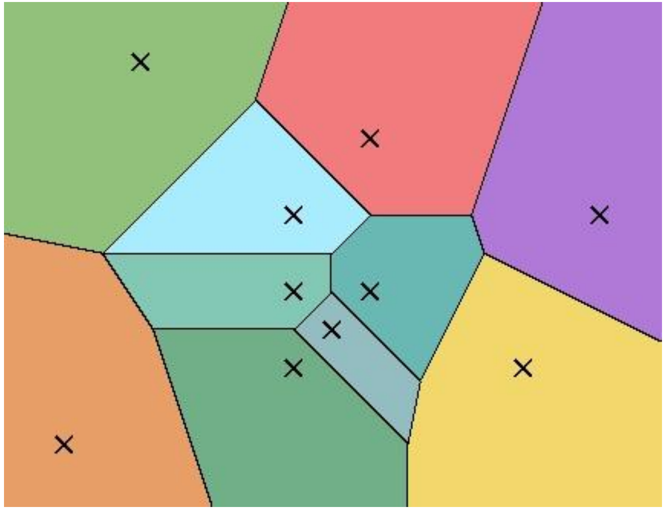
Rasterization

- Transfering a continuous 2D signal to a table
- Usually regular grid
- The sampling frequency has to be twice the maximum frequency in the image
  - Otherwise moiré pattern… then aliasing
- Techniques to avoid this
  - Blurring layer in front of CCD sensors
  - Halftoning patterns on images



Quantization

- Converting a continuous point (or vector) to an integer in {1…k}
  - Reproduction value
- Quantization in a vector space defines a Voronoi diagram
- Quantization of scalars
  - Example: sound is typically 44.1kHz, 16bit



Exercise: quantization of uniform scalars

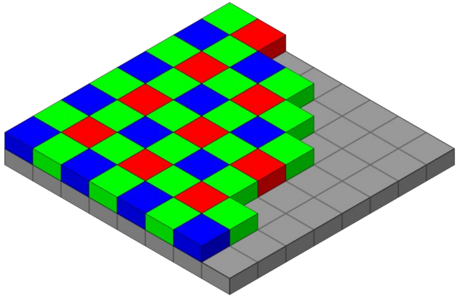
- Quantizer of [0, 1) to {0..k-1}
  - Q(x) = floor(x * k)
- What is the reconstruction?
- Compute expected quantization error
  - Mean squared error (MSE)
- Peak Signal to Noise Ratio (PSNR)
  - In decibel (dB)
  - Max = max value of the signal = 1 in this case
  - Higher = better

$$\mathrm{PSNR} = 10 \log_{10}\left(\frac{\mathrm{Max}^2}{\mathrm{MSE}}\right)$$

- How does PSNR depend on k ?

Colors

- 3 color channels
- RGB color space
  - Bayer pattern
  - 8 bit per channel
- HSV
  - Color picker
- YUV (Y: luminance, U et V: chrominance)
  - Used for compression
  - Higher resolution for luminance
- CMYK:
  - for print
  - Subtractive
- CIELAB
  - Perceptually uniform space



Comparing image pixels

- We have a digital representation of the images
- How to compare them?
  - Assuming images are of the same size
  - Just serialize into a vector and compute MSE on that…
  - How compression is evaluated

Pixel-wise comparison of images

①

Gaussian noise

①

Gaussian noise

①

Gaussian noise
PSNR = 19.82 dB

②

Crop + scaling

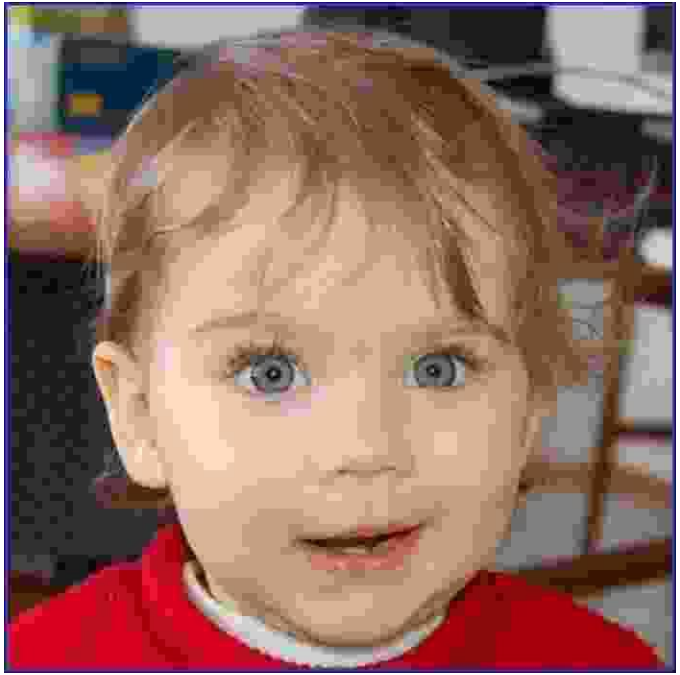
②

Crop + scaling

②

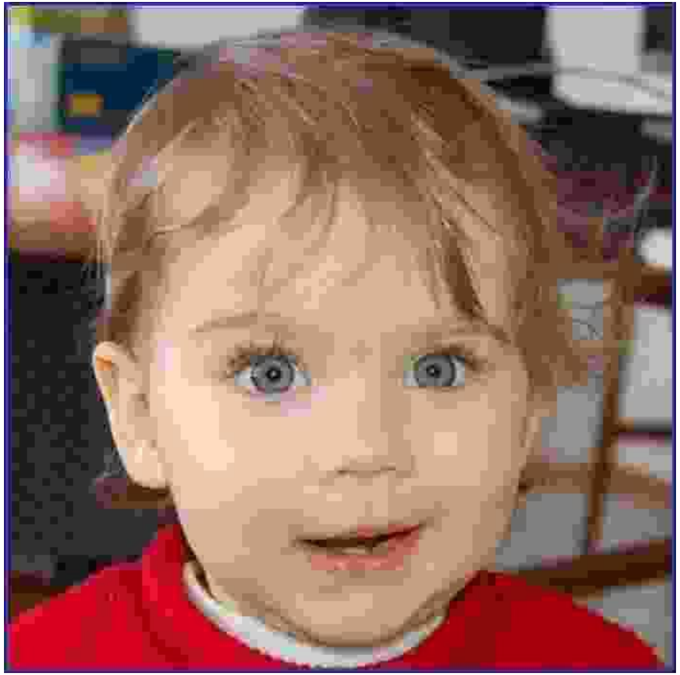
Crop + scaling
PSNR = 15.63 dB

③



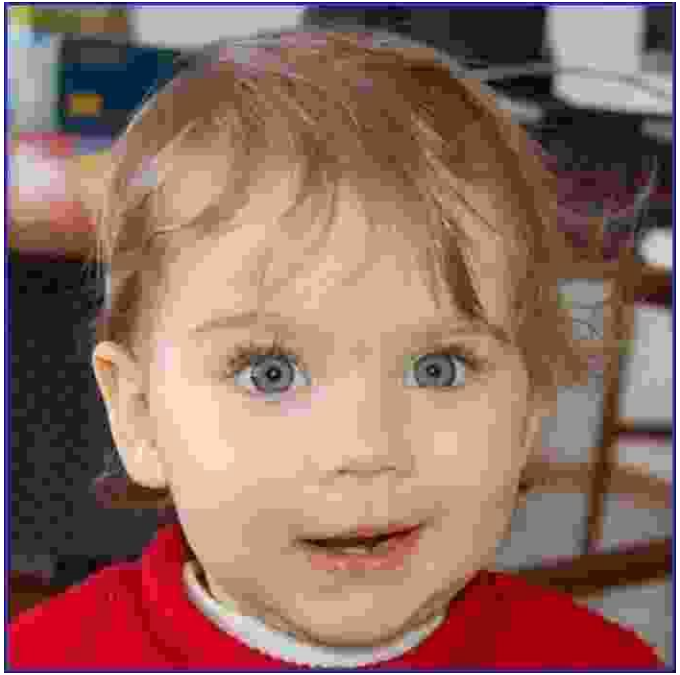
JPEG compression
(quality 5)

③



JPEG compression
(quality 5)

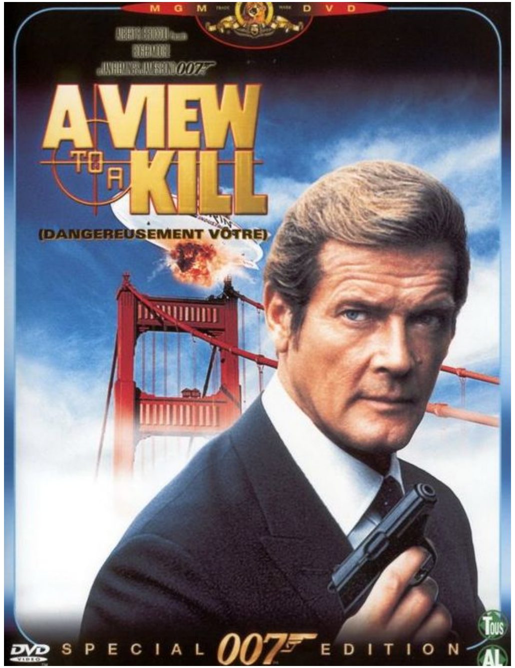
③



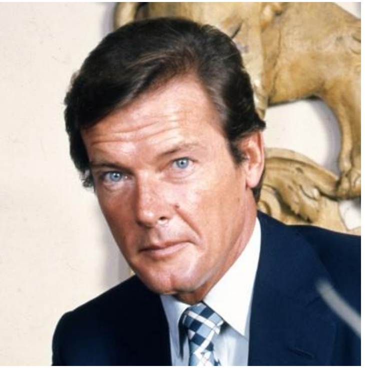
JPEG compression
PSNR = 25.84 dB

Levels of image recognition

Similarity search: what kind?



Same text



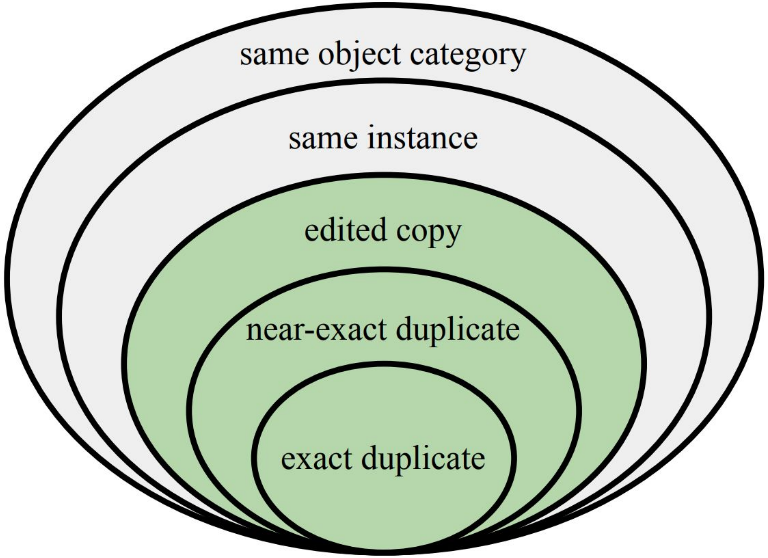
Same face

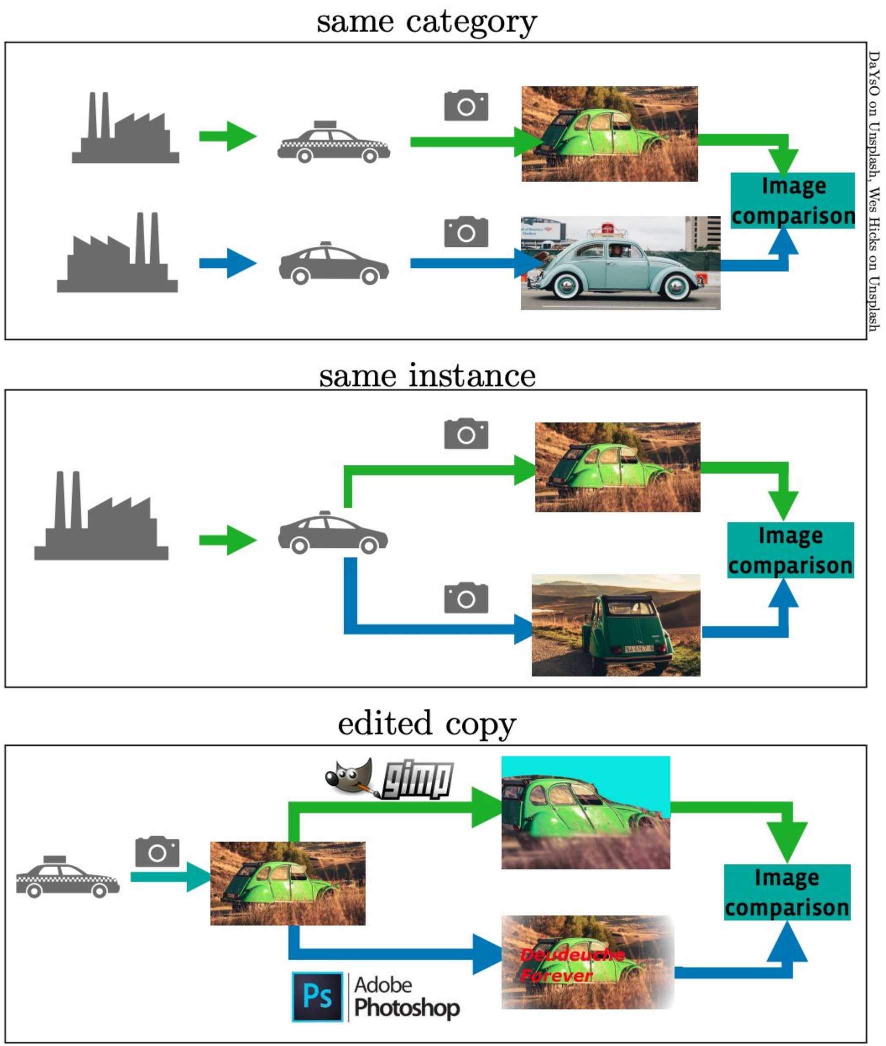
Same object

[The 2021 Image Similarity Dataset and Challenge, Douze et al, ArXiV'21]

Our focus

- General natural image recognition
  - Face / OCR are specific tasks
  - Medical imaging, satellite, etc.
- Nesting of image similarity levels

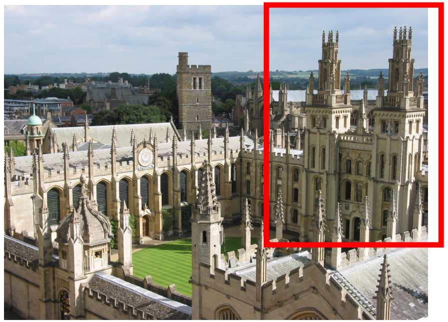




“Same instance” level

queries

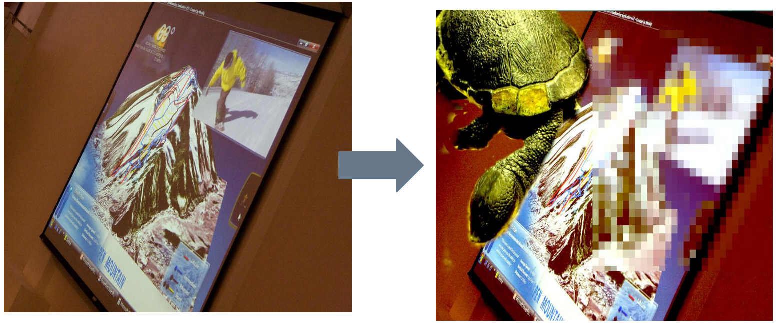
Correct results



http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/index.html

“Edited copy” level

qno 95619 bno 141163



Flickr / roland

Ambiguity….

- Visually close images that are **not** copies

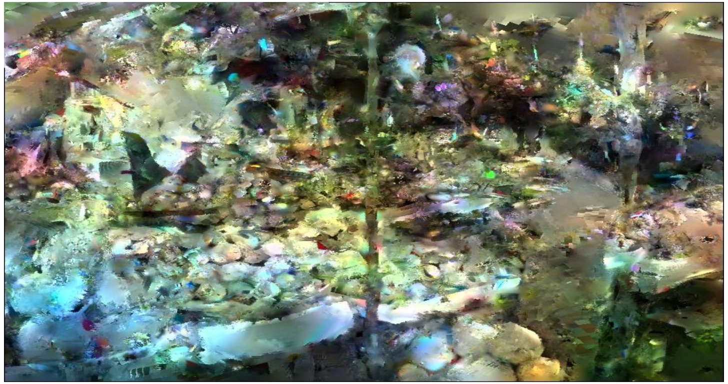
- Visually close images that are **not** the same object

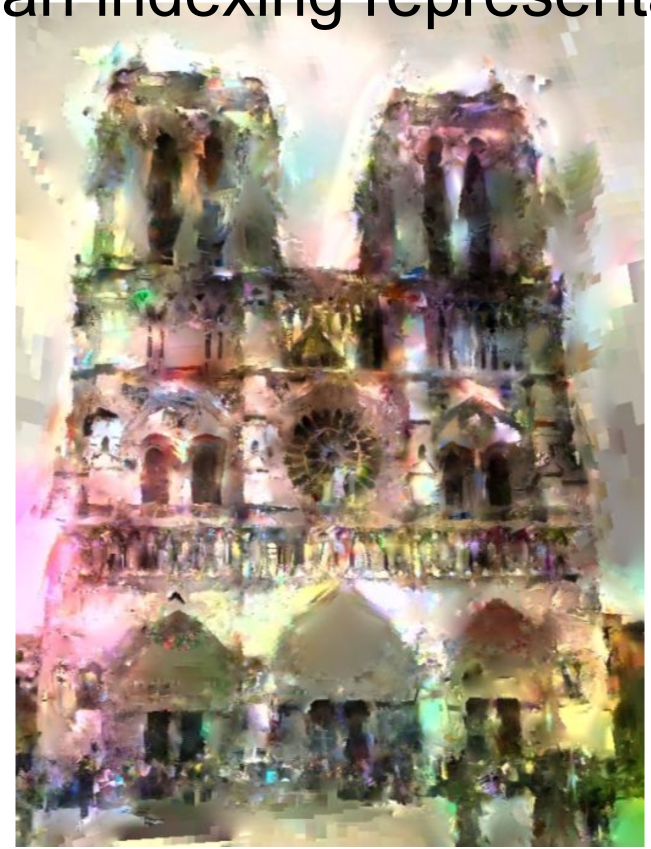
Image representation: what for?

- For image compression

- The only lossy step is quantization
  - Usually…
- Very different problem from search
  - Reconstruction contains lots of useless info for search

What can we reconstruct from an indexing representation?





[P. Weinzaepfel, H. Jégou, P. Pérez, “reconstructing an image from its local descriptors”, CVPR 2011]

Visual cues
for image similarity

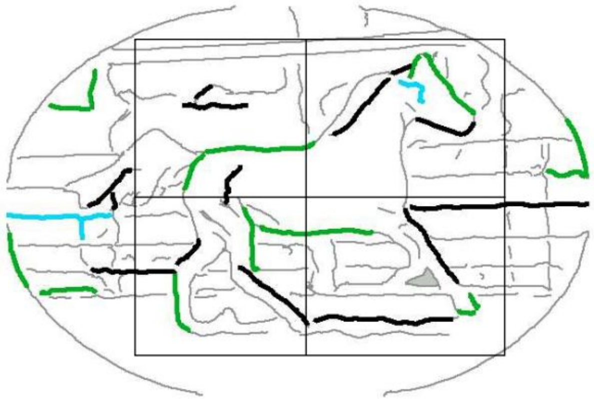
Lots of redundant information

Low-level visual cues: colors

- Easy to extract
  - eg. color histogram
- Invariant to geometrical layout of image
- Not very discriminant in isolation

Low-level visual cues: shapes

- Extract edges
- Recognize n-uplets of edges
- Works for some distinctive shapes
- Difficult to have perfect edge recognition



[V. Ferrari, L. Février, F. Jurie, C. Schmid, *Groups of Adjacent Contour Segments for Object Detection,* PAMI 2008]

Invariance vs. discriminative power

- For a certain set of transformations,
- Visual cues are more or less invariant

Very invariant:
high recall

Very discriminative:
high precision

Global color histogram

Pixel-wise comparison

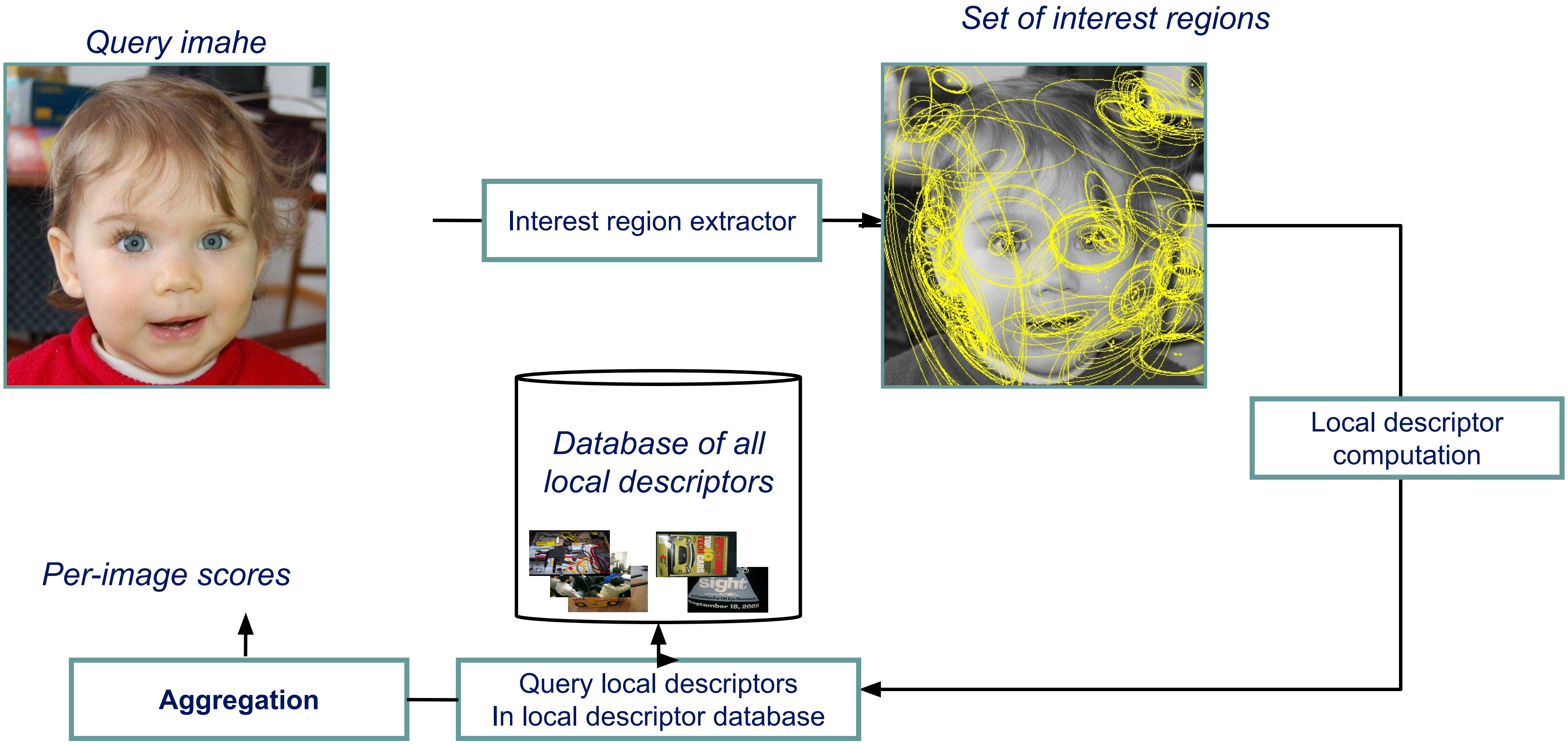
Local / global image descriptors

- Descriptor = embedding

- Local descriptors
  - Descriptors located on parts of the image
  - Image = set of descriptors + localization
  - Matched and compared across images
  - Robust to
- Global descriptors
  - One descriptor per image
  - Easy to index

Local image descriptors

Typical local descriptor indexing



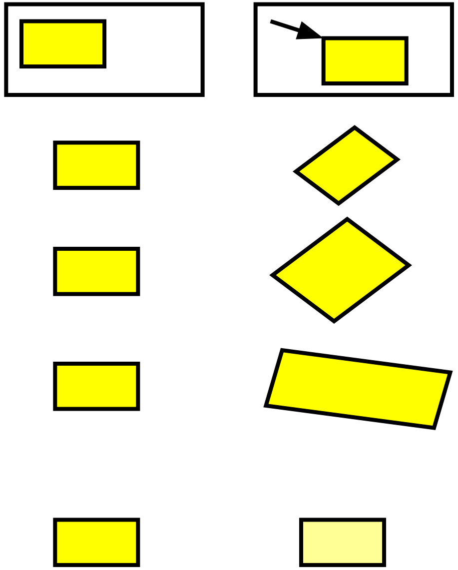
Typical applications

- Images of the same object with different viewpoints
  - Building matching
  - Different viewing conditions
- Planar image matching

- Stages:
  - Detection
  - Non-maximum suppression
  - Neighborhood normalization
  - Descriptor extraction

Local descriptor exrtractors: requirements

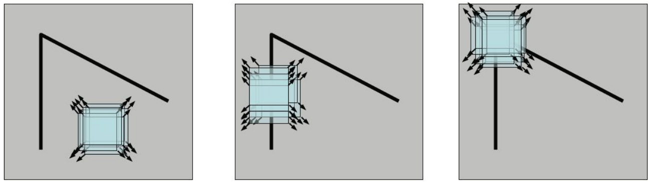
- Should be **invariant** to…
- Geometrical transformations
  - translation
  - rotation
  - rotation + scale
  - affine (local approximation of homography)
- Photometric transformations
  - Affine intensity change ($I \rightarrow a\,I + b$)



The Harris local detector

[A Combined Corner and Edge Detector, C. Harris et M. Stephens, 1988]

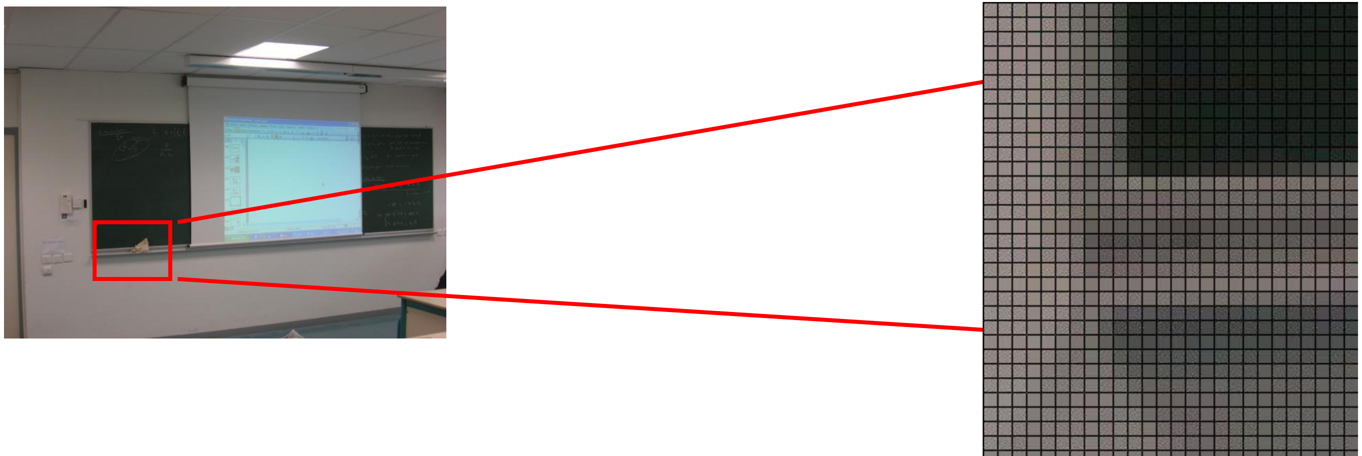
- Detect “corners”
  - Repeatable on images
  - Precisely localized
- Local analysis
  - Corner → strong image gradient in all directions



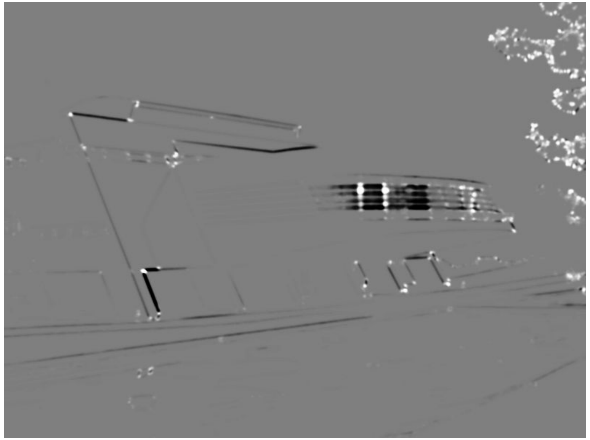
“flat” region: no change in all directions

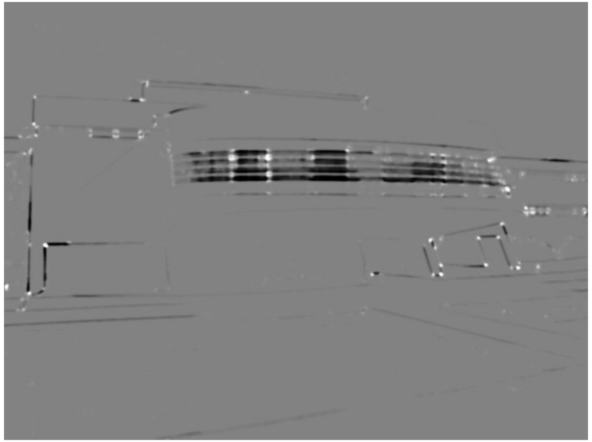
“edge”: no change along the edge direction

“corner”: significant change in all directions



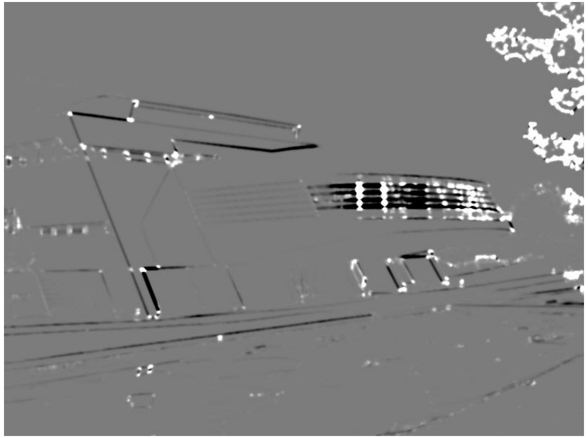
Harris : exemple

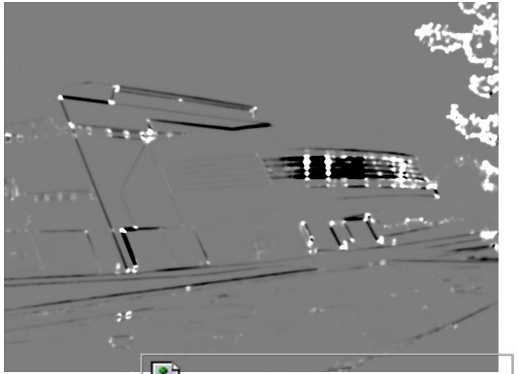


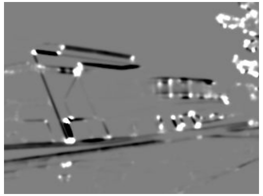


Scale invariance

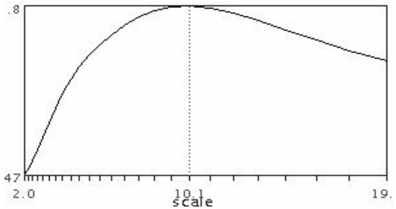
- Image pyramid
- Extraction at each scale

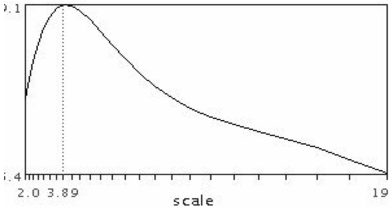






- Keep per-scale maximum

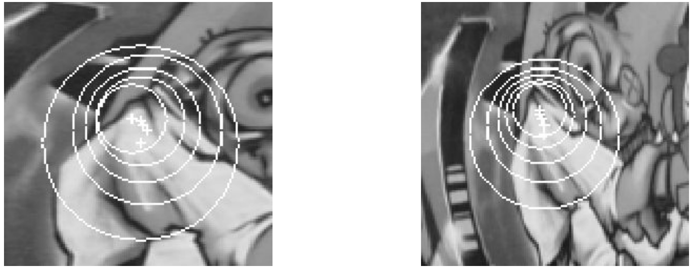




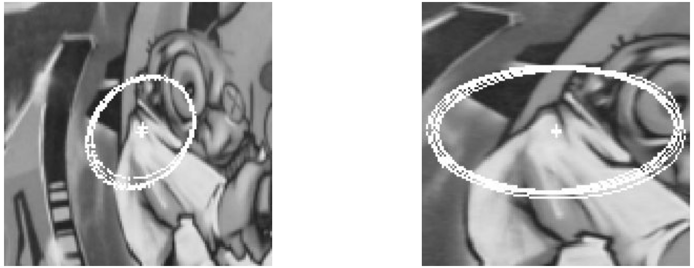
[A comparison of affine region detectors, K. Mikolajczyk et al., IJCV 2005]

Affine normalization

- initialization

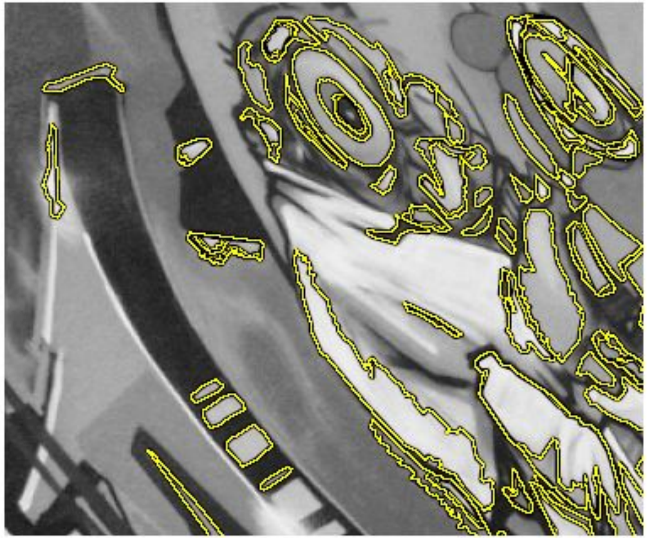


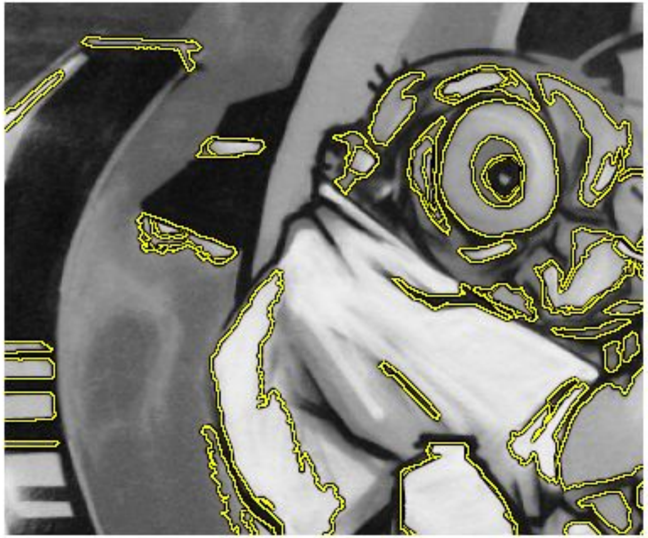
- Iterative estimation of neighborhood: circle → ellipse



Variants…

- MSER

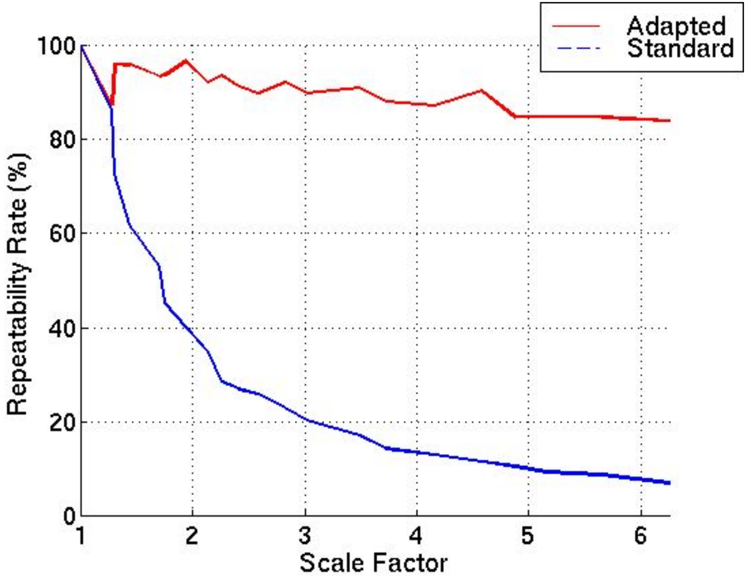


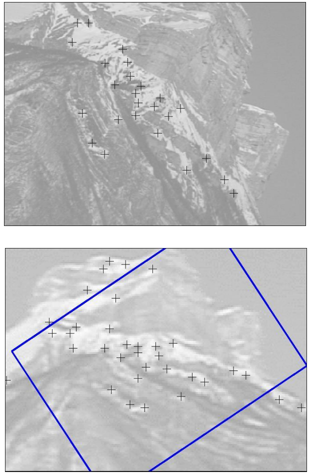


[Robust wide-baseline stereo from maximally stable extremal regions, J. Matas,, O. Chum, M. Urbana and T. Pajdlaa, Image and Vision Computing 22(10), 2004]

Repeatability of region detectors

- Scale change





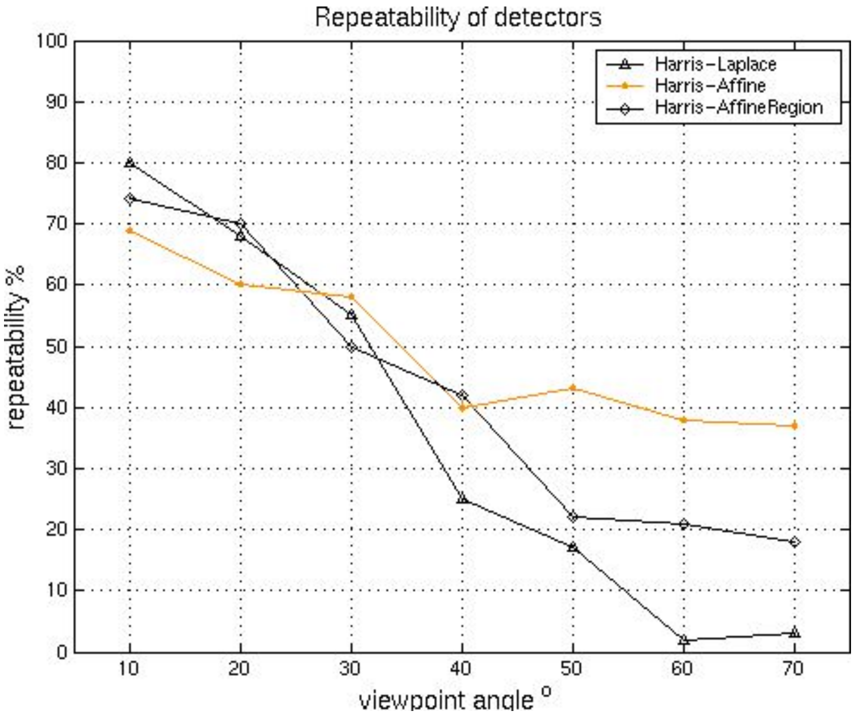
Repeatability – rotation

0

40

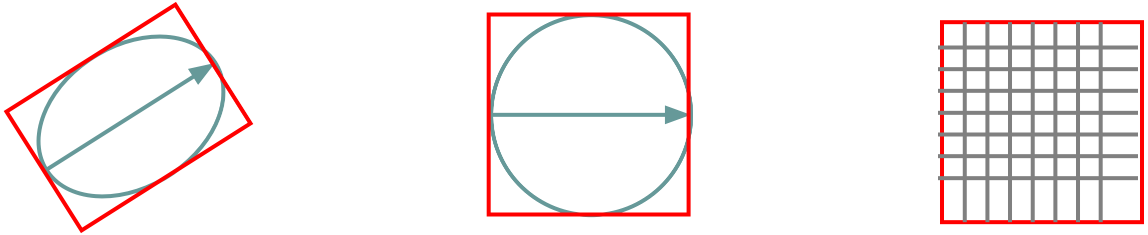
60

70



Descriptor extraction

- From patches



- Sampled on the image

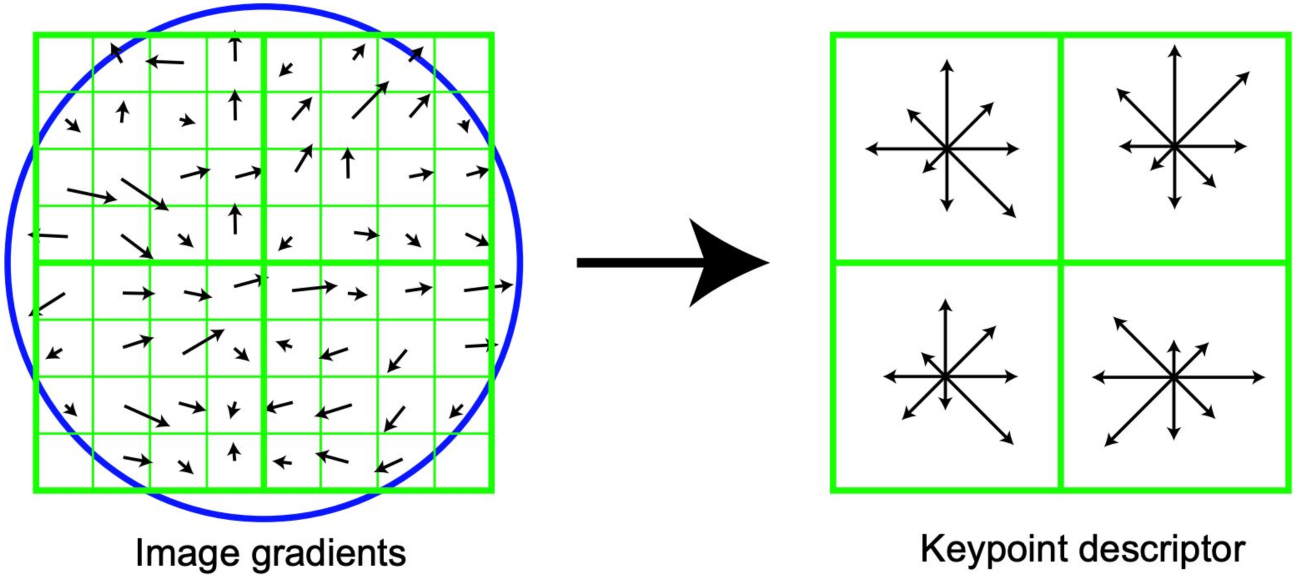


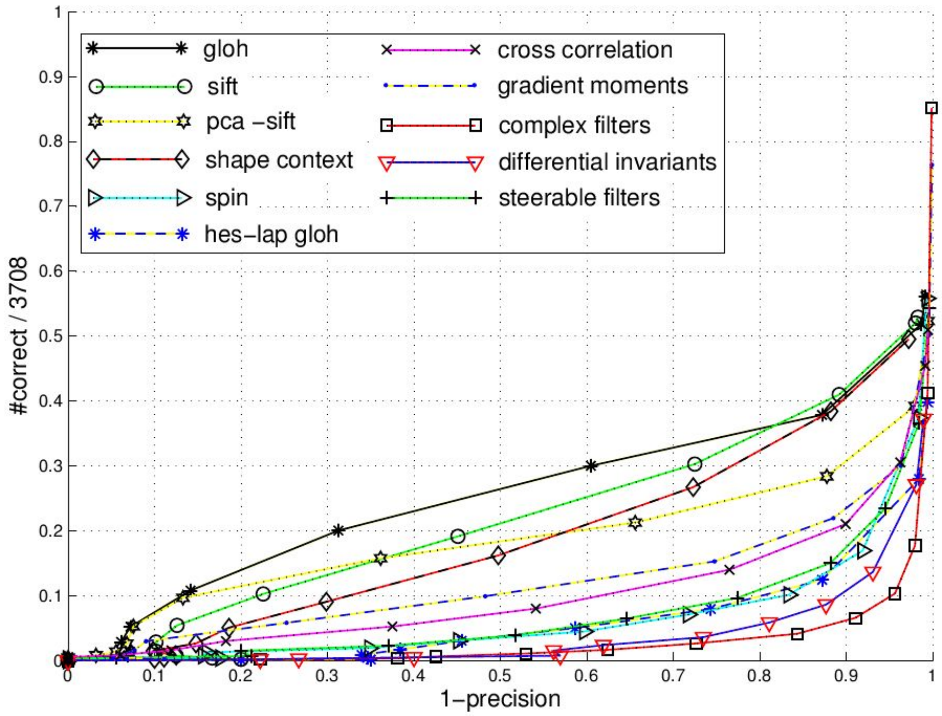
Figure 7: A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location, as shown on the left. These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 subregions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array.

[Lowe. "Distinctive Image Features from Scale-Invariant Keypoints", IJCV'04]

[Mikolajczyk & Schmid, A performance evaluation of local descriptors, PAMI'05]

Variants and evaluation

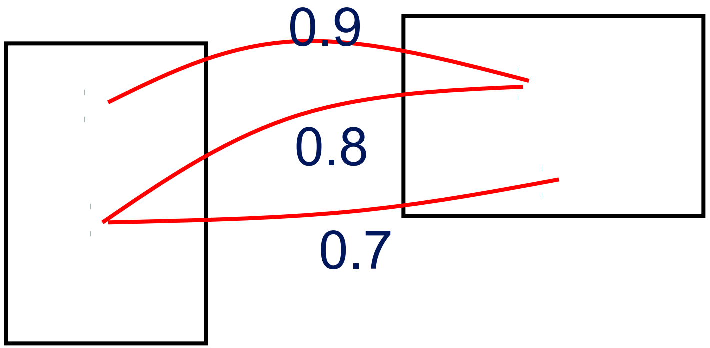
- PR plot



Matching images with local descriptors

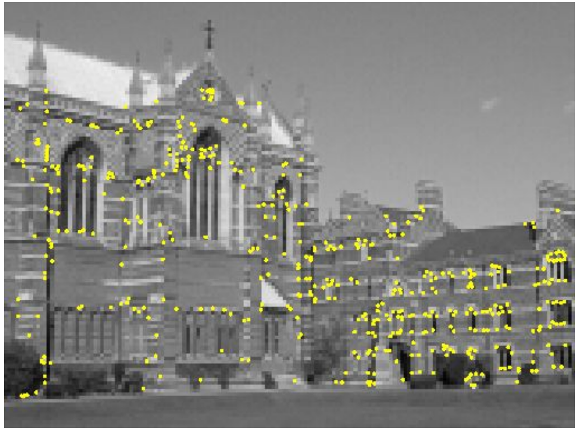
Geometric matching

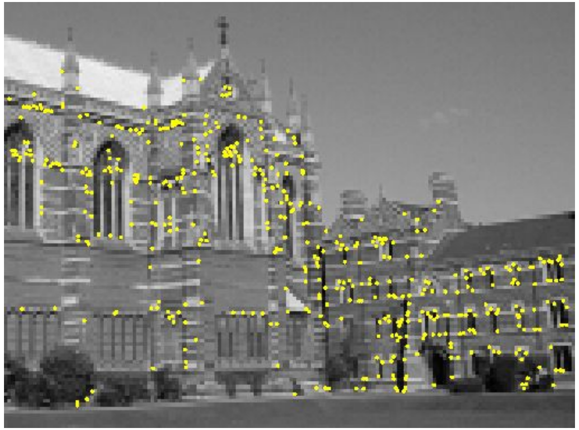
- Vector search gives matching keypoints
  - Lowe's criterion – contrast with background matches
- Sometimes ambiguous…
  - Winner takes all

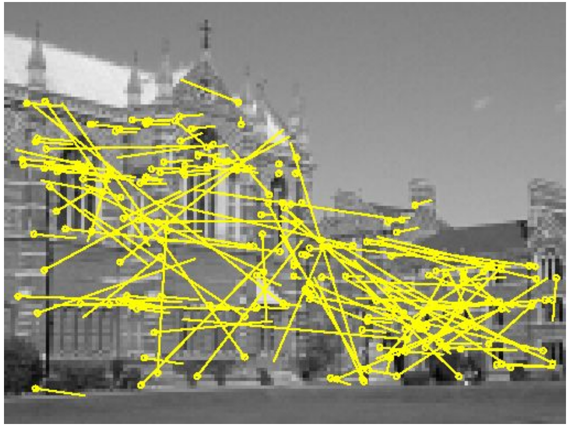


Outliers…

[Multiple view geometry in computer vision
R Hartley, A Zisserman, Cambridge univ press 2003]







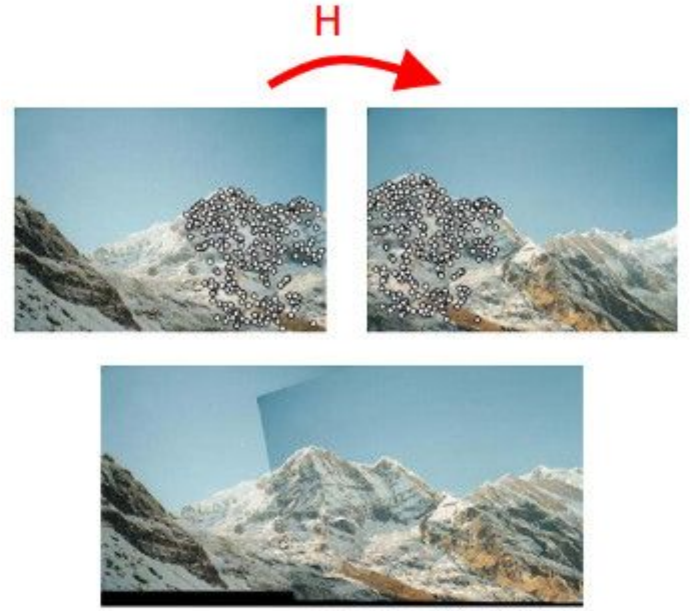
Hierarchy of 2D planar transformations

	DOF	Geometrical invariants	Mathematical expression
translation	2	tout, sauf les positions absolues	
Rigid transformation	3	Lengths, angles, surfaces	
Similarity	4	Length ratios	
Affine transformation	6	Parallelism, surface ratios	
Homography	8	cross-ratio	

\+ Epipolar geometry

Estimating transformation parameters and finding outliers

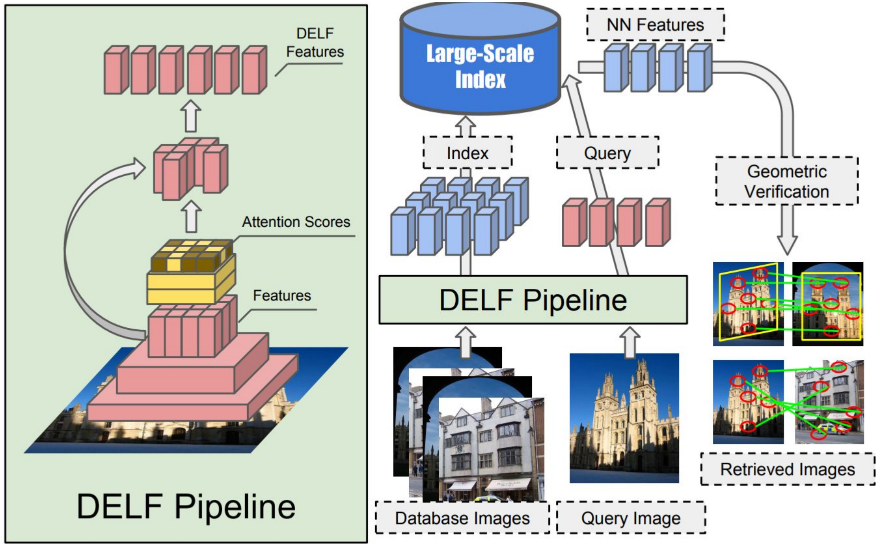
- All variants are linear in their parameters
- RANdom SAmple Consensus
  - Sample enough points
  - Estimate parameters
  - Count inliers
  - Iterate….
- Tradeoff between
  - accurate geometric model
  - ease of parameter estimation
- What for
  - Number of inliers as an image matching metric
  - Remap image to superpose with another image



[OpenCV documentation]

DELF: deep image descriptor

- Dense local descriptors
  - Standard neural net (resnet50)
- A neural net that predicts important features
- Training with image-level supervision only



[Large-Scale Image Retrieval With Attentive Deep Local Features, Noh et al, ICCV'17]

Global image descriptors

[The earth mover's distance, multi-dimensional scaling, and color-based image retrieval Y Rubner, LJ Guibas, C Tomasi - Proceedings of DARPA Image, 1997]

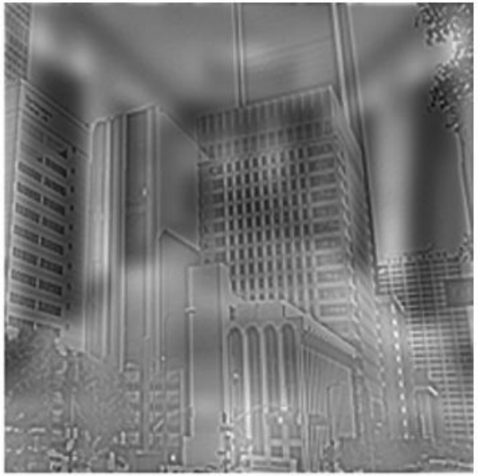
Simple global image descriptors: color histogram

- Adaptive color palette
- Compare color palettes with earth mover's distance
    - Slow!
- Invariant to shape…

Simple global image descriptors: GIST

- Global version of the SIFT Descriptor: image = patch
- General layout of image
- Easy to extract…

(a)

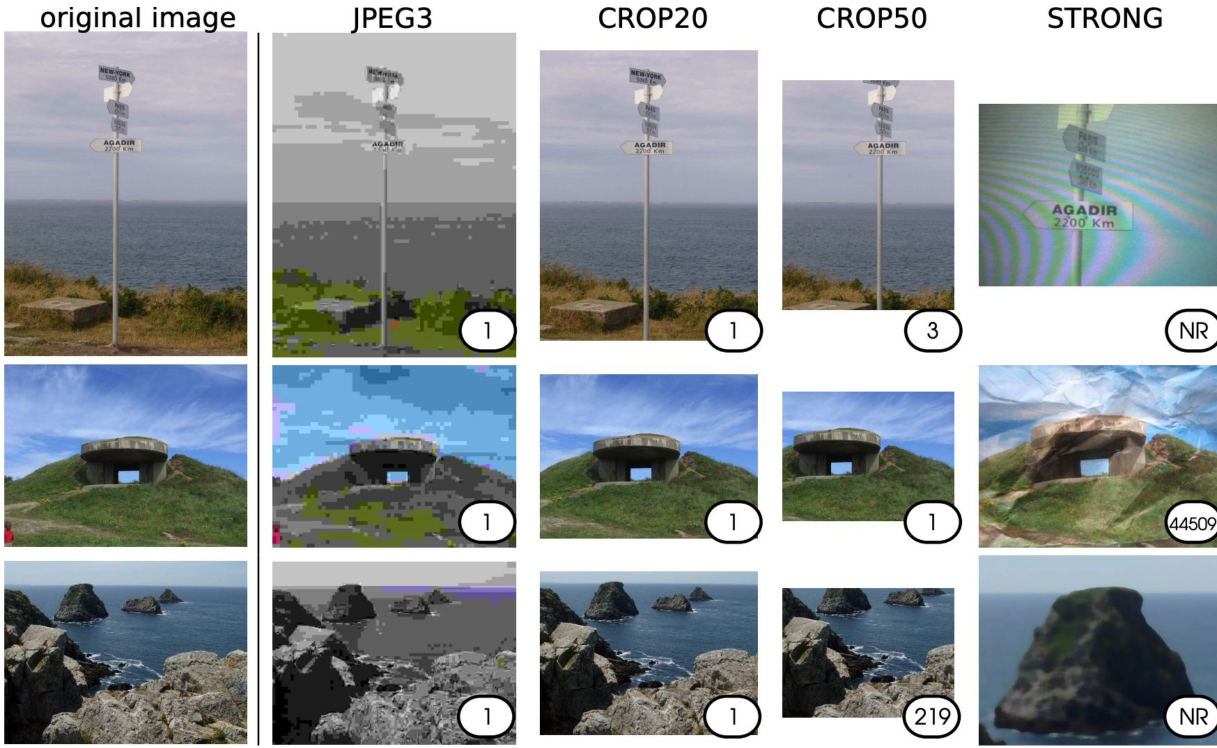


(b)

[Evaluation of gist descriptors for web-scale image search, Douze et al, ICVR’09]

Value of cheap global descriptors

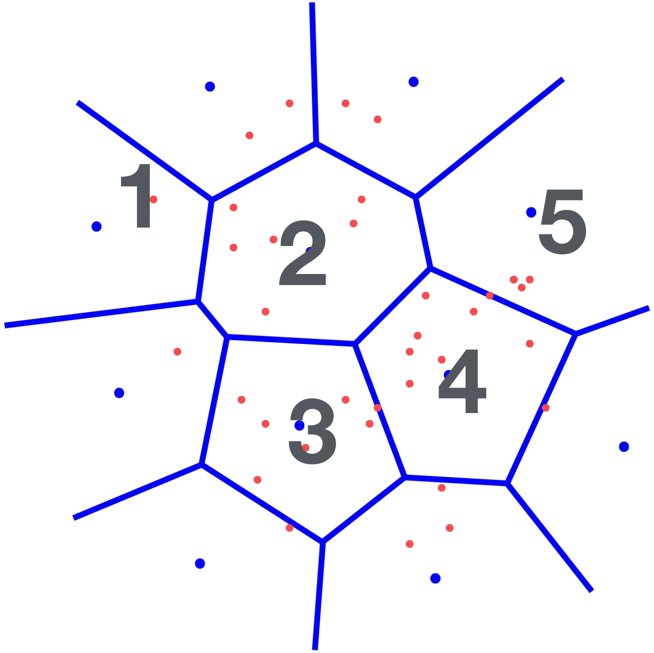
- Results of searching in 100M vectors
- Works for small changes
- Pre-filtering for more accurate second stage.



Bag of visual words

- Summarize local descriptors into a global descriptor

$$\mathbb{R}^d \rightarrow \{1, ..., k\}$$



- Count vectors assigned to each cell
- $\rightarrow$ bag of words
- inverted index

Bag of visual words

[Sivic & Zissermann, Video Google: A Text Retrieval Approach to Object Matching in Videos, ICCV'03]

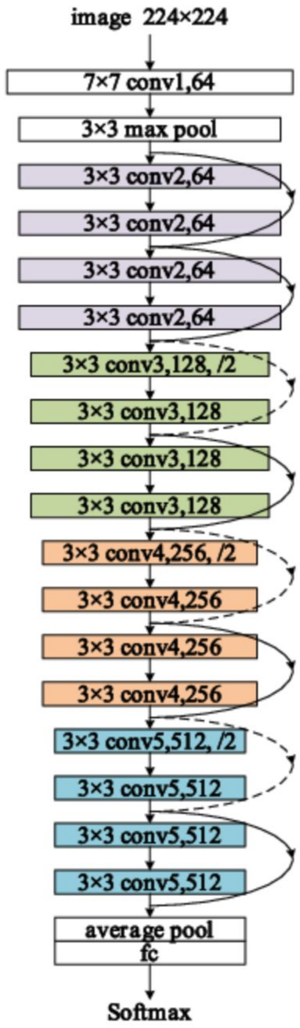
- Import tricks from text processing
  - Stop words
  - TF-IDF
- Post-ranking is useful
- First large-scale local descriptor based indexing
- Many improvements:
  - Add binary signature (Hamming Embedding)
  - Accumulate differences w.r.t. Centroids (VLAD)

Deep learned methods

Neural networks for images

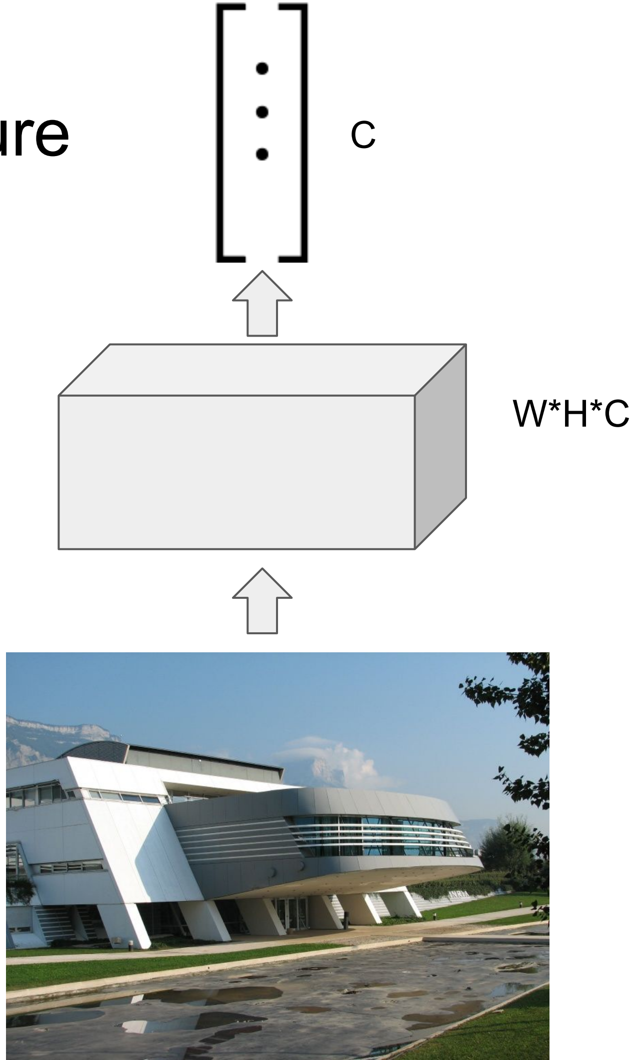
- Typical architecture: resnet
  - Family of models
  - Clear scaling rules
- Start from image of fixed size
- Intermediate representation: tensor
  - Width * height * nb channels
  - Initially nb channels = 3
- Stack of convolutional layers
  - Convolution involves all channels → all channels
  - Trainable parameters
- Applied as residuals (add to previous value)
- Resolution reductions – increase nb channels

[Deep residual learning for image recognition, Kaiming He et al, CVPR’16]



Deep descriptors: general architecture

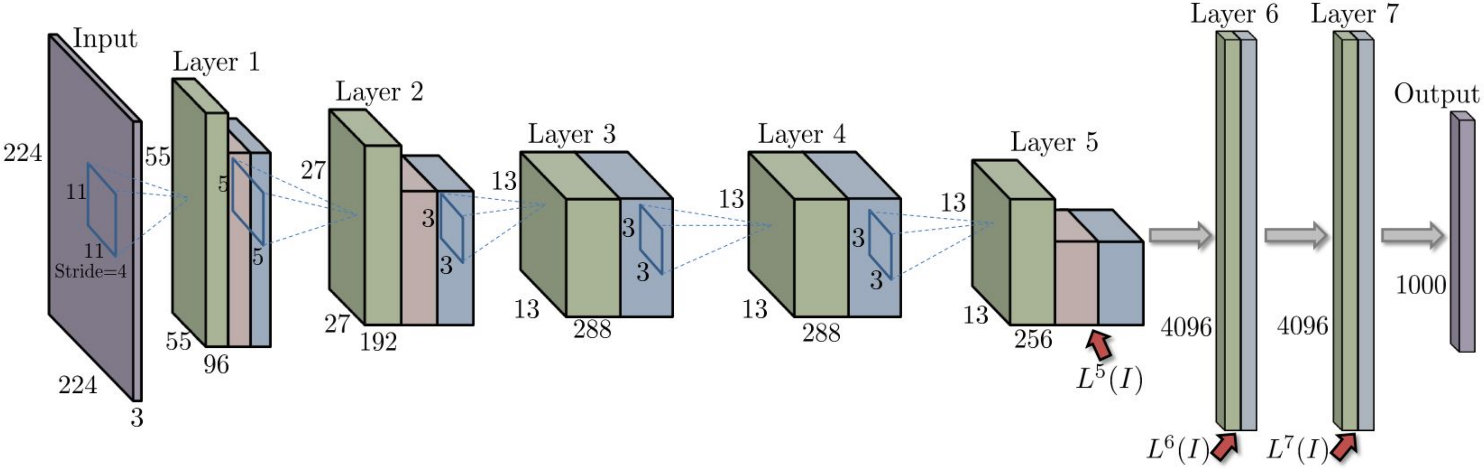
- Convolutional (or transformer) trunk
  - Eg. resnet50
- Generates an activation map
  - Dense set of vectors, localized geometrically
- Pooling function
  - → to an embedding vector
  - Simplest: average pooling (used for classification)



Simplest approach

[Babenko et al, Neural Codes for Image Retrieval, CVPR'14]

- Use CNN trained for classification between buildings
- Embedding = representation from one of the classification layers



Results

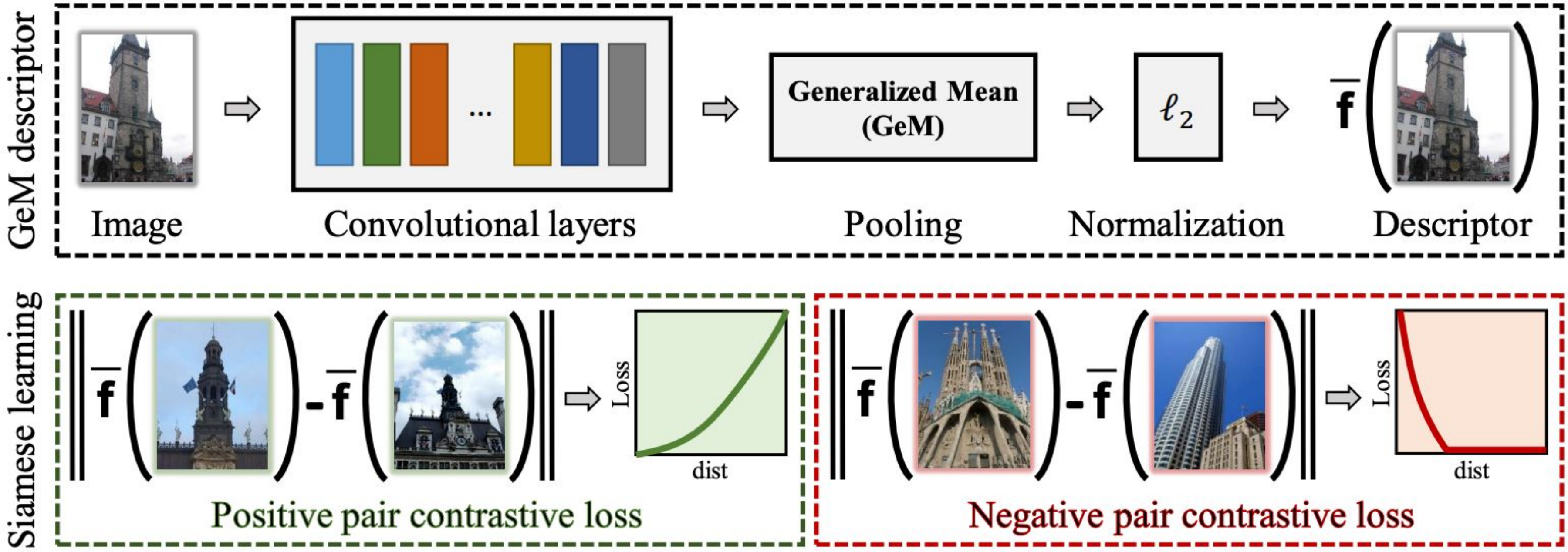
- Competitive with handcrafted descriptors
- Benefits from re-training
  - But still on classification dataset

Descriptor	Dims	Oxford	Oxford 105K	Holidays	UKB
Fisher+color[7]	4096	—	—	**0.774**	3.19
VLAD+adapt+innorm[2]	32768	0.555	—	0.646	—
Sparse-coded features[6]	11024	—	—	0.767	**3.76**
Triangulation embedding[9]	8064	**0.676**	**0.611**	0.771	3.53
Neural codes trained on ILSVRC					
Layer 5	9216	0.389	—	0.690*	3.09
Layer 6	4096	0.435	0.392	0.749*	3.43
Layer 7	4096	0.430	—	0.736*	3.39
After retraining on the Landmarks dataset					
Layer 5	9216	0.387	—	0.674*	2.99
Layer 6	4096	0.545	0.512	**0.793***	3.29
Layer 7	4096	0.538	—	0.764*	3.19
After retraining on turntable views (Multi-view RGB-D)					
Layer 5	9216	0.348	—	0.682*	3.13
Layer 6	4096	0.393	0.351	0.754*	3.56
Layer 7	4096	0.362	—	0.730*	3.53

Table 1. Full-size holistic descriptors: comparison with state-of-the-art (holistic descriptors with the dimensionality up to 32K). The neural codes are competitive with the state-of-the-art and benefit considerably from retraining on related datasets (Land-

Training for retrieval

[Radenovic et al, Fine-tuning CNN Image Retrieval with No Human Annotation, PAMI'18]



“Tricks” for precise image matching

$$f_k^{(g)} = \left(\frac{1}{|\mathcal{X}_k|} \sum_{x \in \mathcal{X}_k} x^{p_k} \right)^{\frac{1}{p_k}}.$$

- GeM pooling
- Contrastive loss training

$$\mathcal{L}(i,j) = \frac{1}{2} \left(Y(i,j) ||\bar{\mathbf{f}}(i) - \bar{\mathbf{f}}(j)||^2 + (1 - Y(i,j)) \left(\max\{0, \tau - ||\bar{\mathbf{f}}(i) - \bar{\mathbf{f}}(j)||\} \right)^2 \right)$$

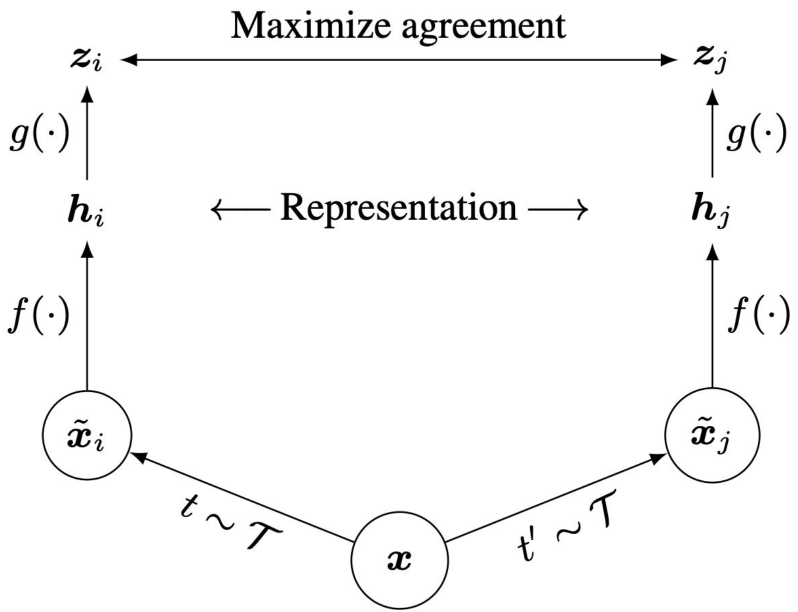
- Positives = BOW verified image matches
- Negatives = images from other buildings that are close for current state of the CNN
- Whitening
- Works well for rigid objects
  - Buildings

[Chen et al, A Simple Framework for Contrastive Learning of Visual Representations, ICML'20]

SimCLR: unsupervised training

- In the context, unsupervised =
  - train a representation on images without labels
- Batches of 2 transformations of an image
  - Image should be recognizable
- NCE loss
- Large batch sizes

$$\ell_{i,j} = -\log \frac{\exp(\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)},$$



SimCLR: augmentations

(a) Original

(b) Crop and resize

(c) Crop, resize (and flip)

(i) Gaussian blur

Results

- “linear evaluation”
  - Train a linear classifier for imagenet on top of the features
- Not evaluated for retrieval

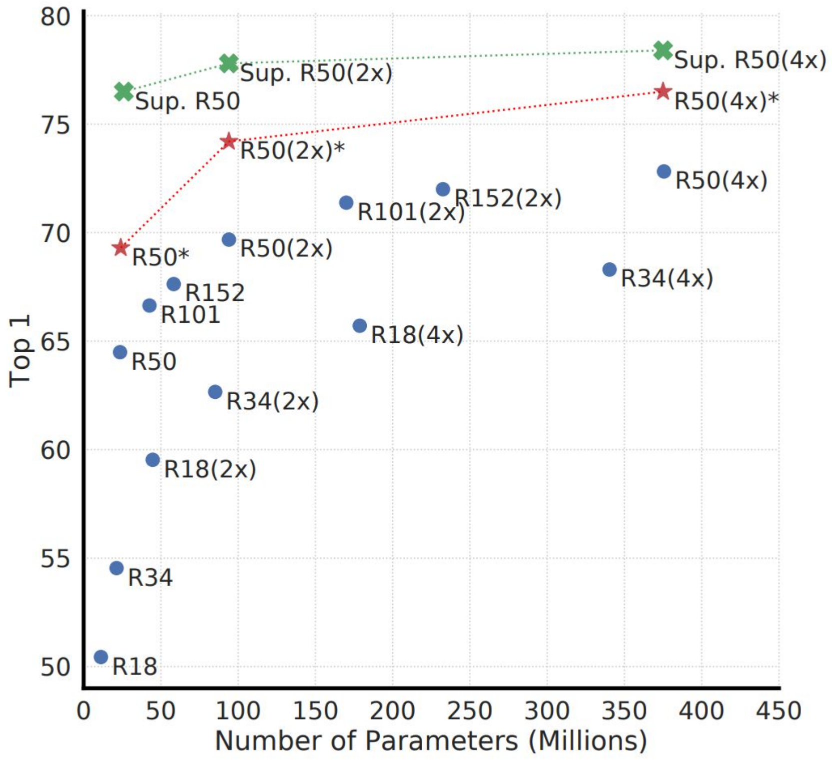


Figure 7. Linear evaluation of models with varied depth and width. Models in blue dots are ours trained for 100 epochs, models in red stars are ours trained for 1000 epochs, and models in green crosses are supervised ResNets trained for 90 epochs[7] (He et al., 2016).

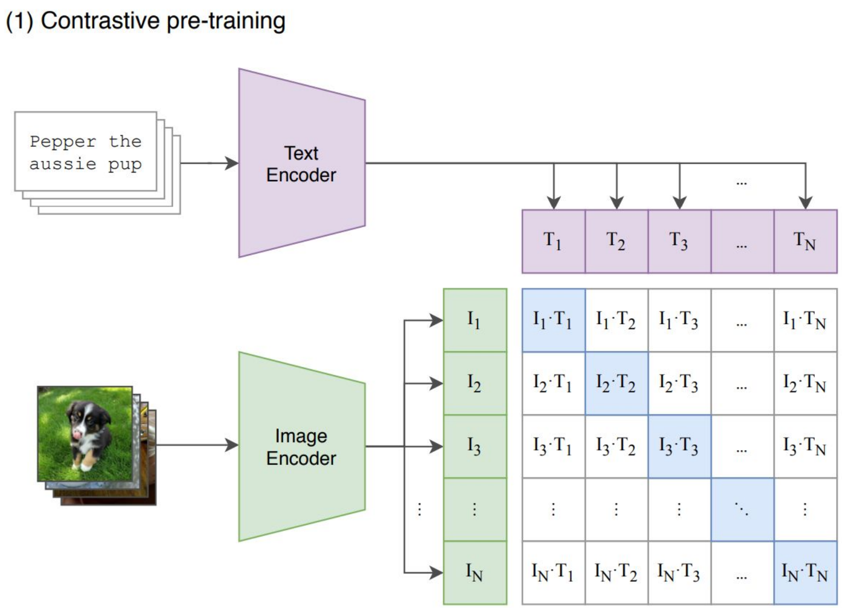
LPIPS perceptual metric

Mixed image-text embeddings

[Learning Transferable Visual Models From Natural Language Supervision, Radford et al, ICML’21]

CLIP

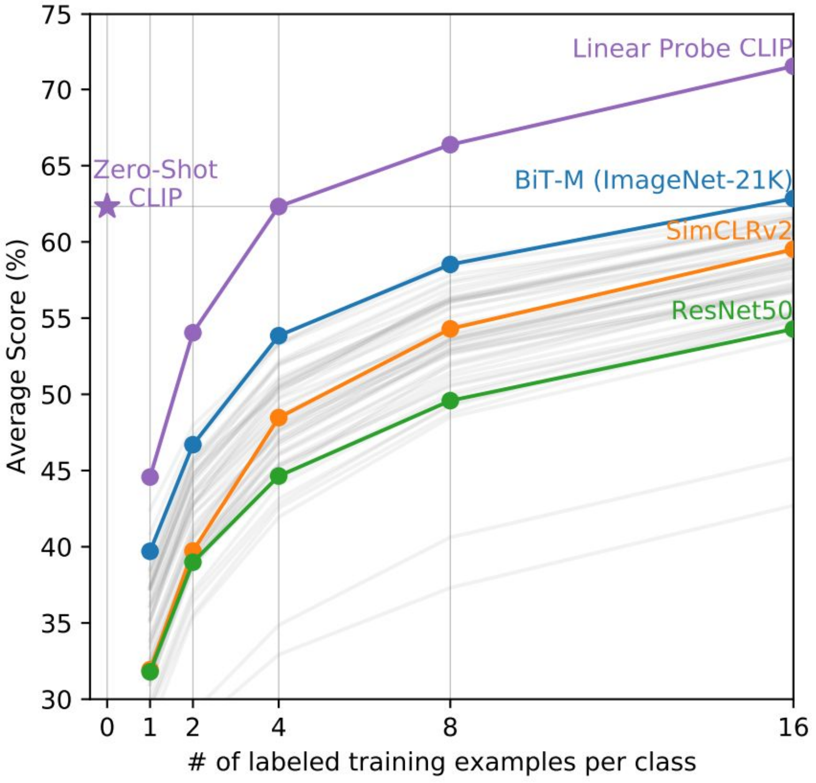
- Text encoder
  - Transformer model
- Image encoder
  - Resnet50 (with adaptations)
- Later → both replaced with transformers
- Distinguish the correct caption in the training minibatch



CLIP training + results

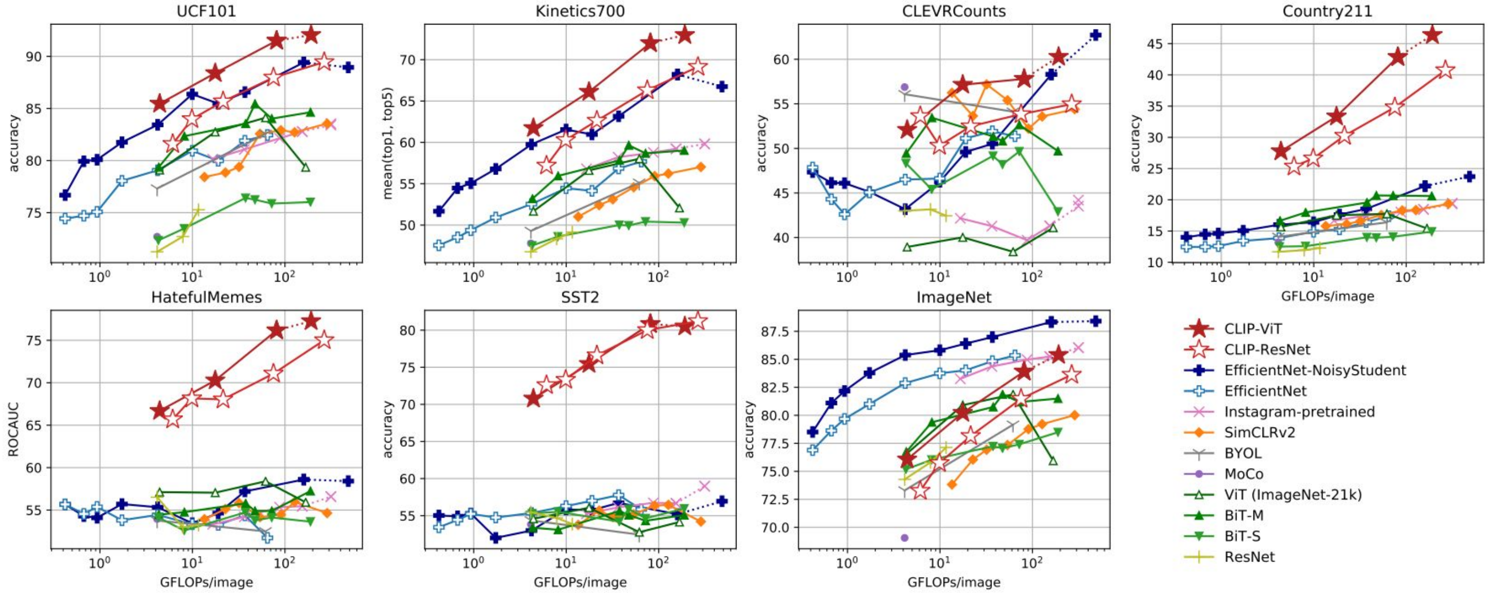
- 400M text-image pairs
- Start without per-modality pre-training
- Large mini-batches (32k)
  - To have enough negatives
- Biggest 18 days on 500 GPUs

- 11 downstream tasks
- Flagship: “0-shot imagenet”
  - Does not use the imagenet training data
  - Works better with a prompt “photo of a XXX”



Results on 12+15 image classification datasets

- Different models
  - x=gflops
- Shows better generalization

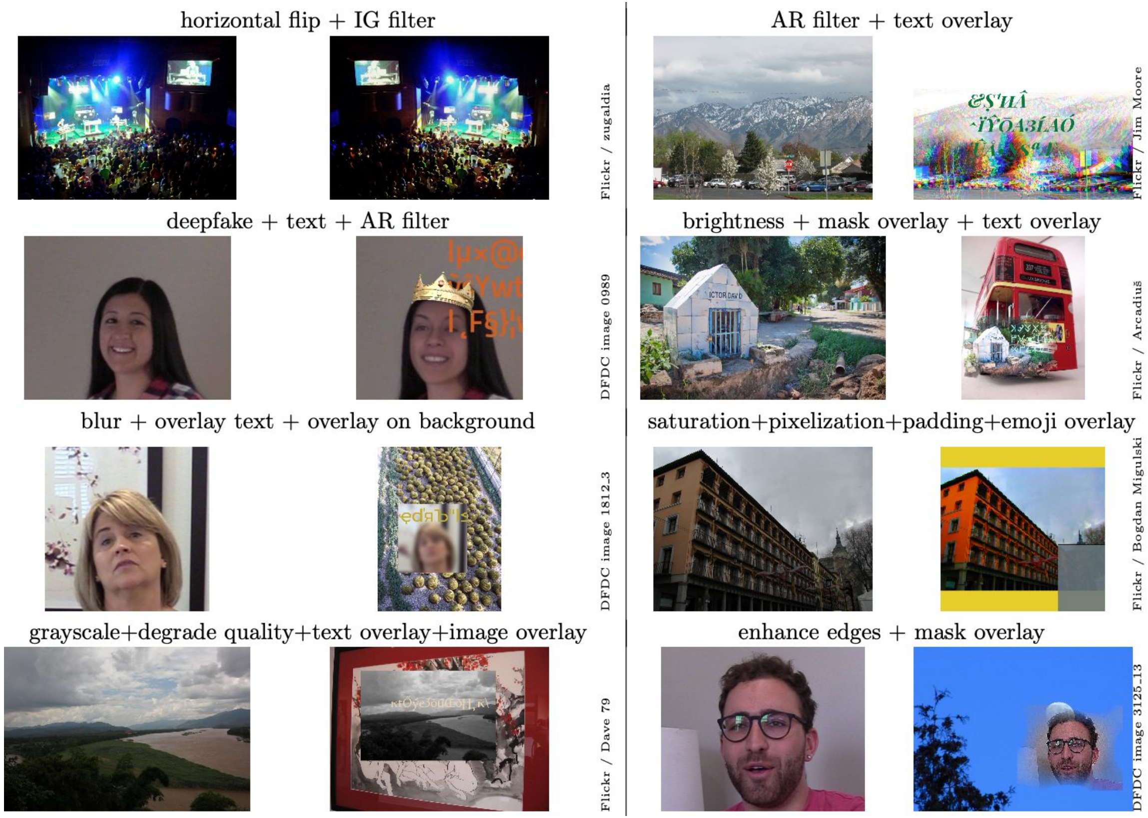


SSCD : training embeddings for image copy detection

[A Self-Supervised Descriptor for Image Copy Detection, Pizzi et al, CVPR'22]

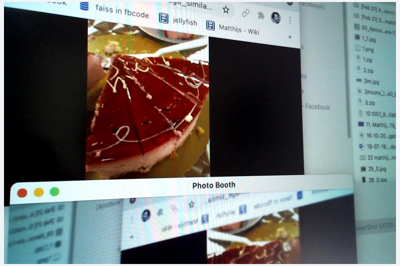
Motivation: the Image Similarity Challenge (DISC2021)

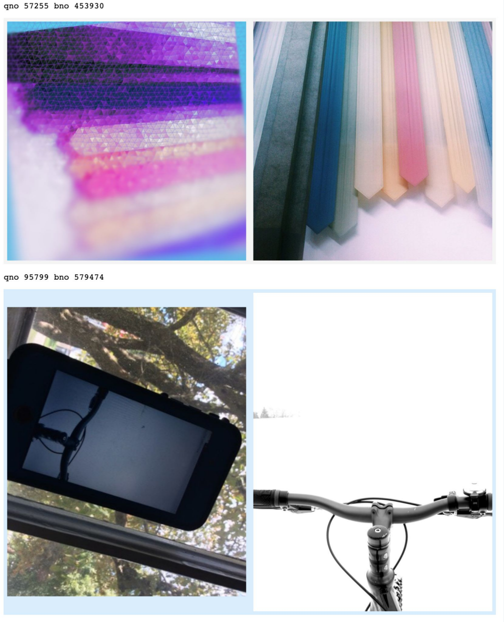
- Detect image copies
  - Dataset scale 1M images
  - Strong image transformations

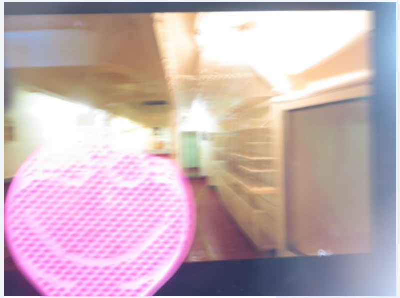


Douze et al. *The 2021 Image Similarity Dataset and Challenge*. Arxiv 2021

Real-world transformations





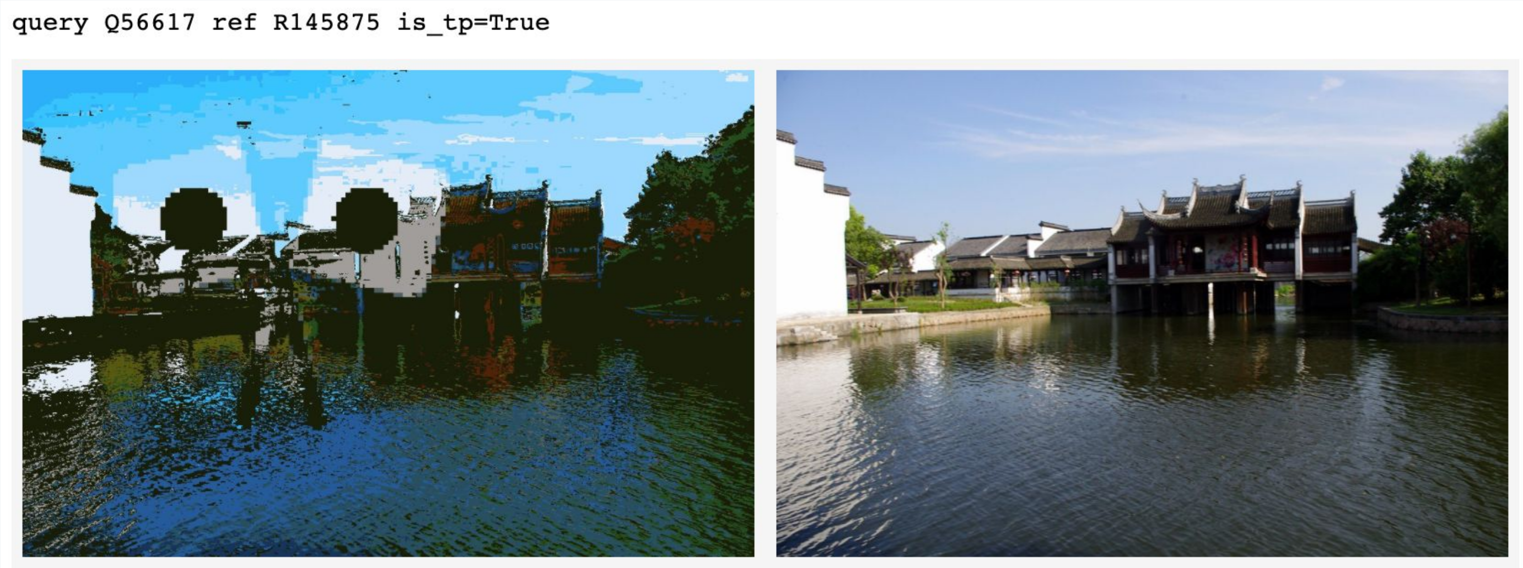


Manual transform example

- Manual example, found > 90% precision, VisionForce / matching
- Editors did an amazing job but
  - it is hard to calibrate the strength of the transformations

Automatic transform example

- Automatic example, found > 90% precision, VisionForce / matching



Baseline: SimCLR

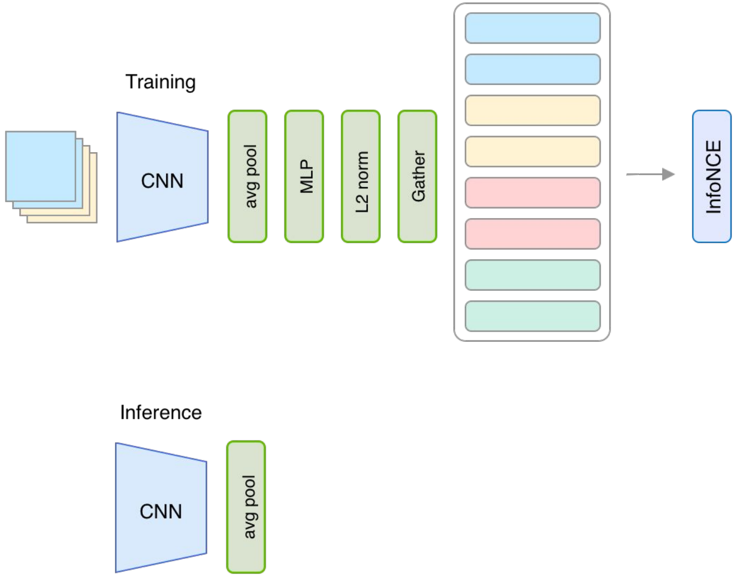
- Contrastive learning objective:
  Learns by training on matching image copies

- Embedding MLP for matching copies is discarded for inference

- Contrastive InfoNCE loss

$$\ell_{i,j} = -\log \frac{\exp(s_{i,j})}{\sum_{k \neq i} \exp(s_{i,k})}$$

$$\mathcal{L}_{\text{InfoNCE}} = \frac{1}{|P|} \sum_{i,j \in P} \ell_{i,j}.$$



Chen et al. *A simple framework for contrastive learning of visual representation.* Arxiv 2020

Part 1:
Contrastive learning for copy detection

- Surprisingly, SimCLR is not especially strong at copy detection.
- Intuitively, it seems it should be. Our work follows this intuition.
- In the first part of this work, we optimize SimCLR for copy detection.

	dimensions	DISC µAP	DISC µAPSN
Multigrain (supervised)	2048	**20.5**	**41.7**
SimCLR	2048	13.1	33.9
SimCLR (with MLP)	128	9.4	17.3

SimCLR for copy detection

SimCLR for copy detection adaptations:

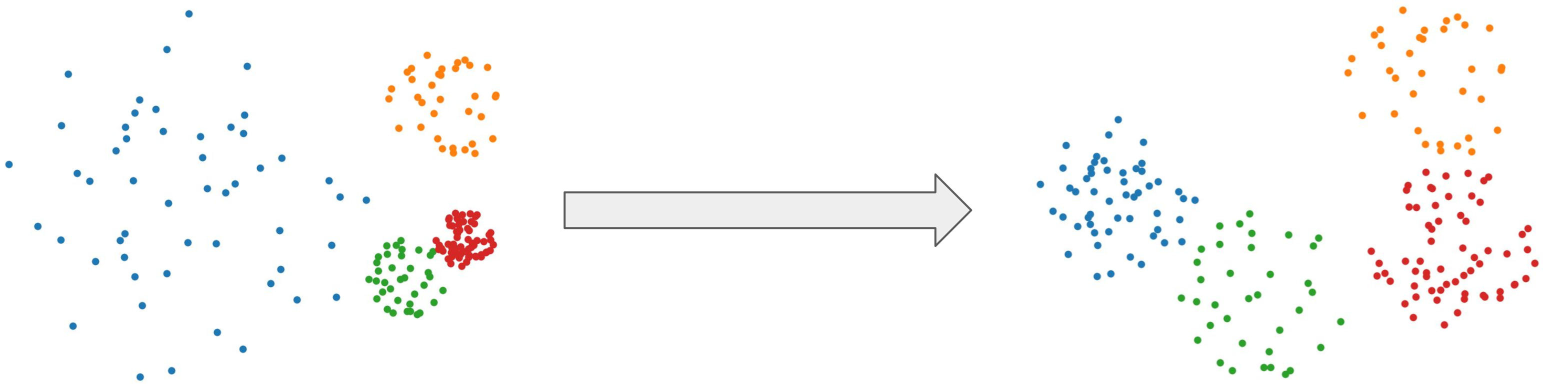
- generalized mean (GeM) pooling
- strengthening the blur augmentation
- using a lower InfoNCE softmax temperature
- using a simple linear projection to 512d

We call this SimCLR_{CD}.

name	method	dimensions	μAP	μAPSN
SimCLR	trunk features	2048	13.1	33.9
	+ GeM pooling	2048	21.5	45.3
SimCLR	projection	128	9.4	17.3
	+ GeM pooling	128	11.1	18.8
	+ strong blur	128	14.1	26.0
	+ low temp	128	26.0	41.5
	+ 512d	512	27.5	43.5
SimCLR_{CD}	+ linear proj	512	33.0	51.6

Part 2: Calibrated descriptor distance

- Descriptor spaces vary in density.
- The meaning of descriptor distance varies based on local density.
- A calibrated descriptor would provide a uniform notion of distance.
  - Can use range search



Differential entropy regularization

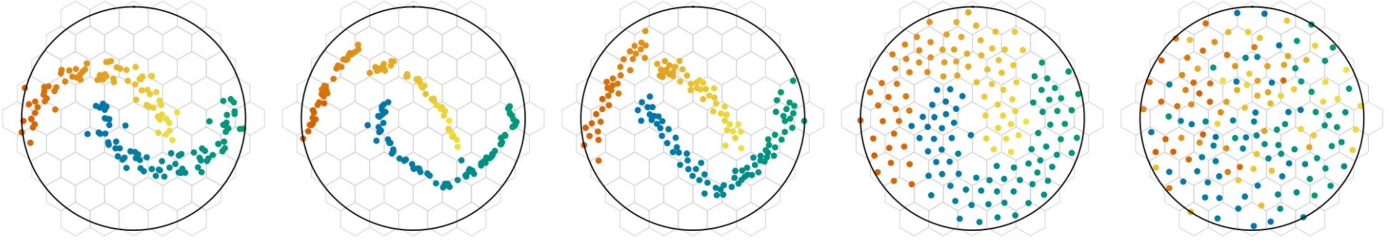
KoLeo loss [1] based on the Kozachenko-Leonenko differential entropy estimator.

Promotes a uniform distribution by maximizing distance to the nearest non-match.

$$\mathcal{L}_{\text{KoLeo}} = -\frac{1}{N} \sum_{i=1}^{N} \log \left(\min_{j \notin \hat{P}_i} \| z_i - z_j \| \right)$$

$$\mathcal{L}_{\text{basic}} = \mathcal{L}_{\text{InfoNCE}} + \lambda \mathcal{L}_{\text{KoLeo}}$$

where P_i is the set of positives (matches) for image i, and λ is a regularization weight.

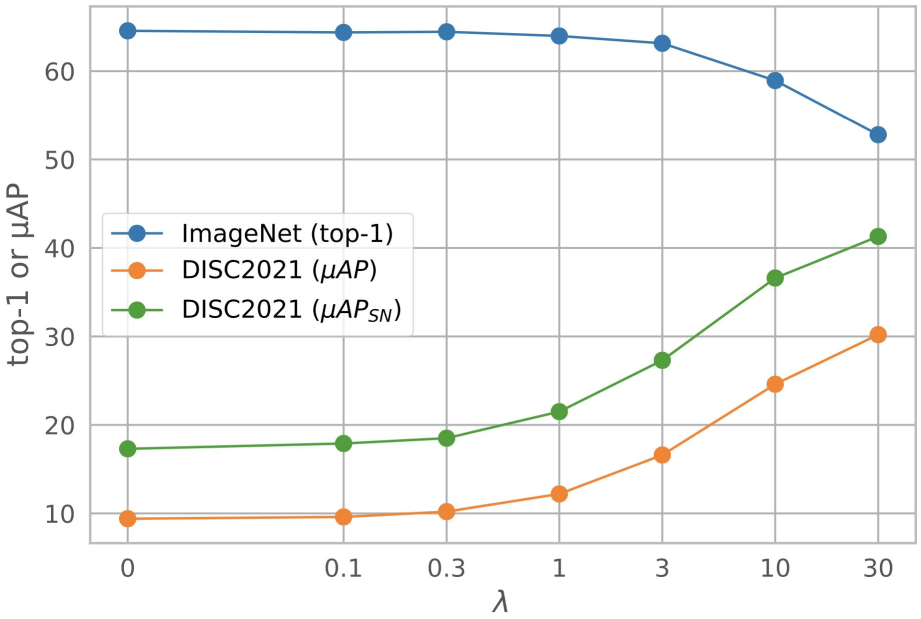


input $\lambda = 0$ $\lambda = 0.01$ $\lambda = 0.1$ $\lambda \to \infty$

[1] Sablayrolles et al. *Spreading Vectors for Similarity Search*. ICLR 2019

SimCLR + differential entropy

SimCLR with varying differential entropy regularization strengths λ (and no other changes)



Resolving the dimensional collapse

Entropy regularization also resolves a collapse described by [1]

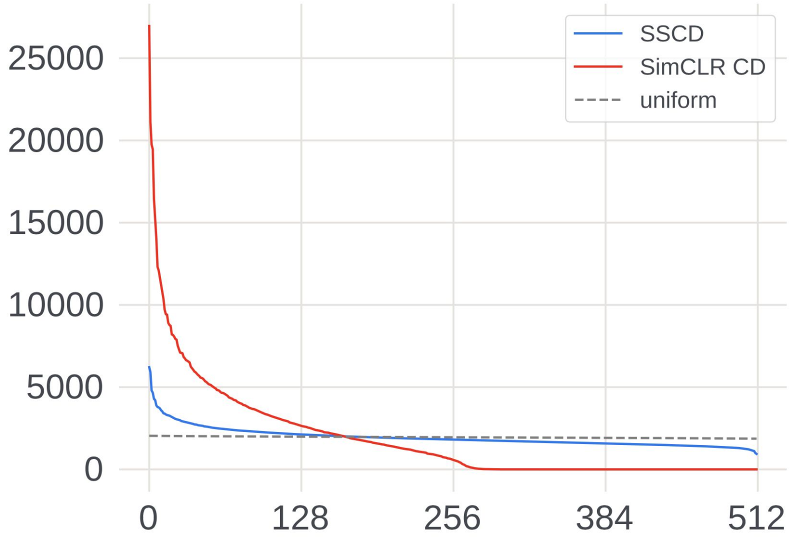
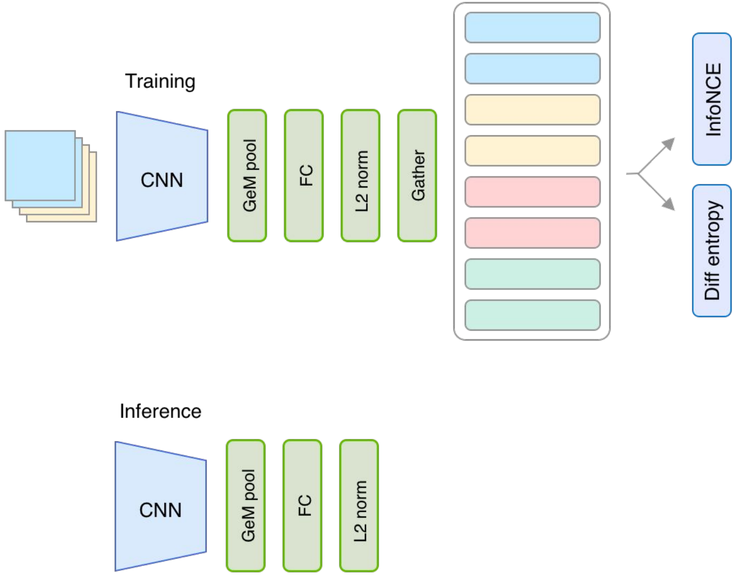


Figure 5. Descriptor principal values on the DISC2021 reference set: SSCD ($\lambda = 30$) and SimCLR_{CD} ($\lambda = 0$), compared to a reference uniform distribution.

[1] Jing et al. *Understanding dimensional collapse in contrastive self-supervised learning*. ICLR 2022

SSCD: SimCLR$_{CD}$ + differential entropy

SSCD combines SimCLR$_{CD}$ optimizations with differential entropy regularization



model	μAP	μAP_{SN}	recall@1	MRR
SimCLR$_{CD}$	33.0	51.6	58.6	60.5
$\lambda = 1$	33.1	51.9	58.7	60.9
$\lambda = 3$	38.0	56.1	62.9	65.1
$\lambda = 10$	45.3	61.5	67.7	69.5
$\lambda = 30$	50.4	64.5	69.8	71.4

Additional experiments

- Additional augmentations
  - Rotations, Emoji, Text
  - MixUp and CutMix to model collages
- Datasets
  - Training on DISC dataset (reduce domain shift)
  - Evaluate on Copydays dataset
- Larger trunk model

method	trained on	transforms	dims	μAP	μAP_{SN}
Multigrain [7]	ImageNet*		2048	20.5	41.7
DINO [9] †	ImageNet		1500	32.2	53.8
SimCLR [10] trunk	ImageNet	SimCLR	2048	13.1	33.9
SimCLR [10] proj	ImageNet	SimCLR	128	9.4	17.3
$\text{SimCLR}_{\text{CD}}$ trunk	ImageNet	strong blur	2048	39.8	56.8
SSCD	ImageNet	strong blur	512	50.4	64.5
SSCD	ImageNet	advanced	512	55.5	71.0
SSCD	ImageNet	adv.+mixup	512	56.8	72.2
SSCD	DISC	strong blur	512	54.8	63.6
SSCD	DISC	advanced	512	60.4	71.1
SSCD	DISC	adv.+mixup	512	61.5	72.5
$\text{SSCD}_{\text{large}}$ †	DISC	adv.+mixup	1024	**63.7**	**75.3**

Example matches

DISC2021 examples where SSCD's first result is correct, and SimCLR's is not.

SSCD	SimCLR	queries
✓	✓	38.9 %
✓	✗	39.0 %
✗	✓	0.3 %
✗	✗	21.8 %