SPATIAL REASONING (2)

IMAGE-BASED LOCALIZATION

Dr TIAN Jing

tianjing@nus.edu.sg

Module objective

Knowledge and understanding

- Understand the fundamentals of spatial reasoning: Image-based location and place recognition, including feature-based and learning-based methods.

Key skills

- Workshop on image-based location and place recognition

Use vision for localization

- Vision data can be used as a complement to
  - Wheel odometry
  - GPS
  - Inertial measurement units (IMU)
  - GPS-denied environments, such as underwater and aerial
- Localization
  - Refer to environment (where am I? focus of our course), useful for navigation
  - Refer to machine itself (what is camera's posture?), useful for display (e.g., Augmented Reality (AR)).

Major reference

- [Intermediate] CS 6476 Computer Vision, https://www.cc.gatech.edu/~hays/compvision/
- [Advanced] CSC2541 Visual Perception for Autonomous Driving, http://www.cs.toronto.edu/~urtasun/courses/CSC2541/CSC2541_Winter16.html
- [Survey]: N. Piasco, D. Sidibé, C. Demonceaux, V. Gouet-Brunet, “A survey on Visual-Based Localization: On the benefit of heterogeneous data,” *Pattern Recognition*, 2018, pp. 90-109.
- [Survey]: L. Zheng, Y. Yang, Q. Tian, “SIFT Meets CNN: A Decade Survey of Instance Retrieval,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 40, No. 5, May 2018, pp. 1224-1244.

Topics

- Introduction to image-based location and place recognition
- Place recognition pipeline
  - Feature extraction
  - Feature encoding
  - Feature indexing
- Workshop on place recognition

Motivation

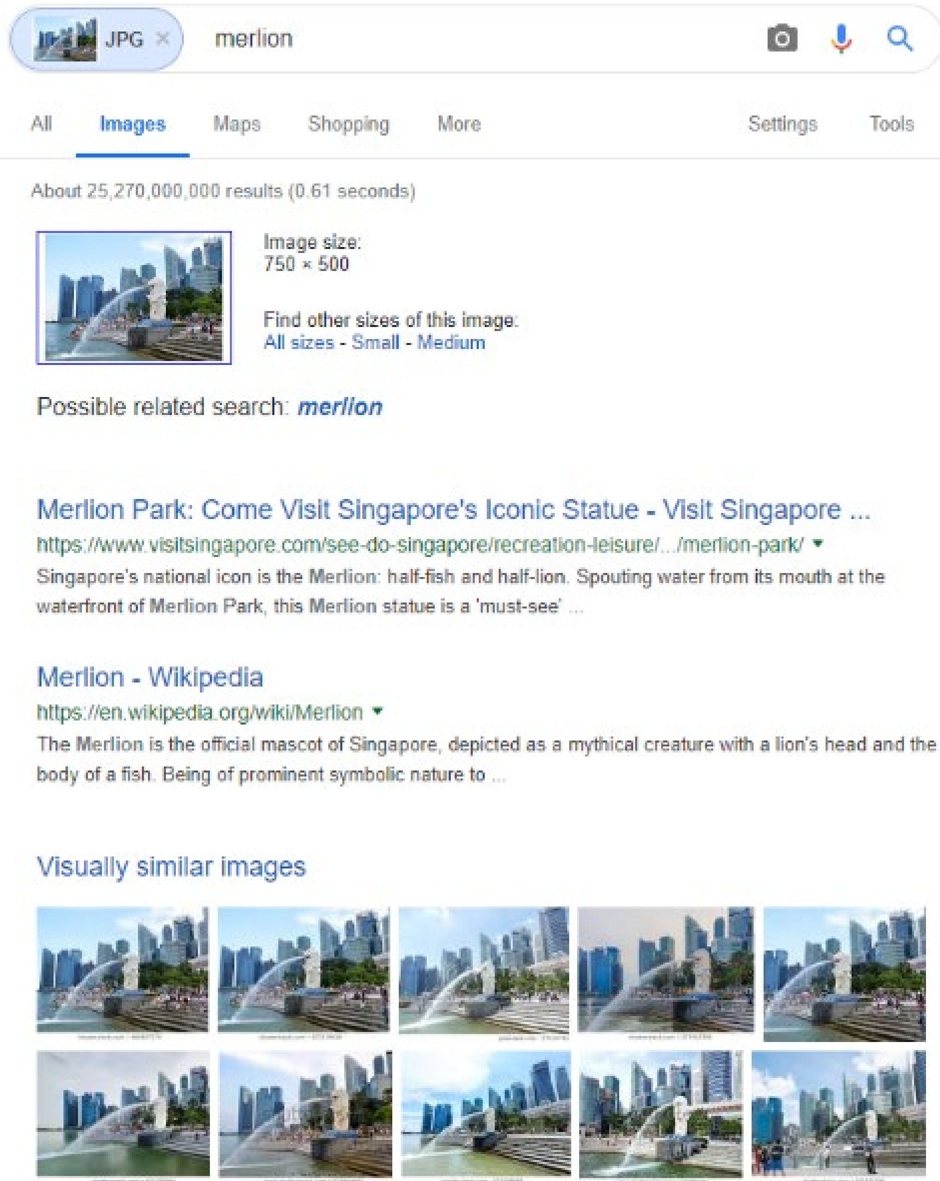


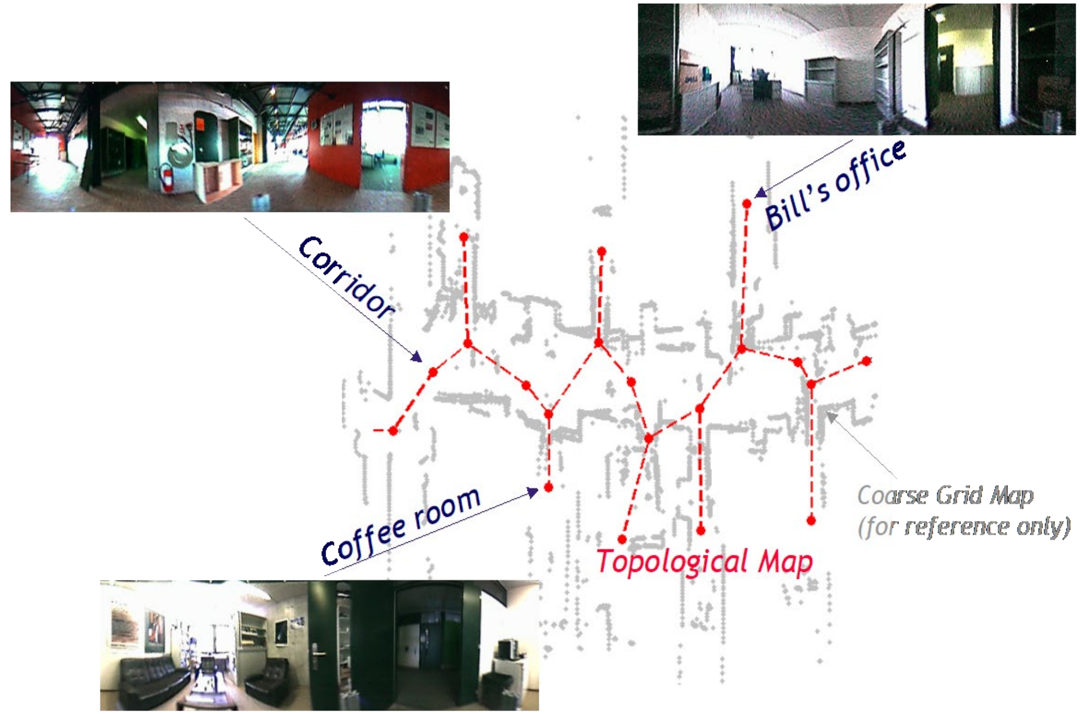
Image-based location and place recognition

- **Image retrieval**: Have I seen this image before? Which images in my database look similar to it?
- Example: Google Reverse Image Search

Motivation

Image-based location and place recognition

- **Robotics**: Has the robot been to this place before? Which images were taken around the same location?
- Example: SLAM (simultaneous localization and mapping), which is the backbone of spatial awareness of a robot.



- A map is necessary for localizing the robot
  - Pure localization with a known map.
  - SLAM: no a priori knowledge of the robot's workspace
- An accurate pose estimate is necessary for building a map of the environment
  - Mapping with known robot poses.
  - SLAM: the robot poses have to be estimated along the way

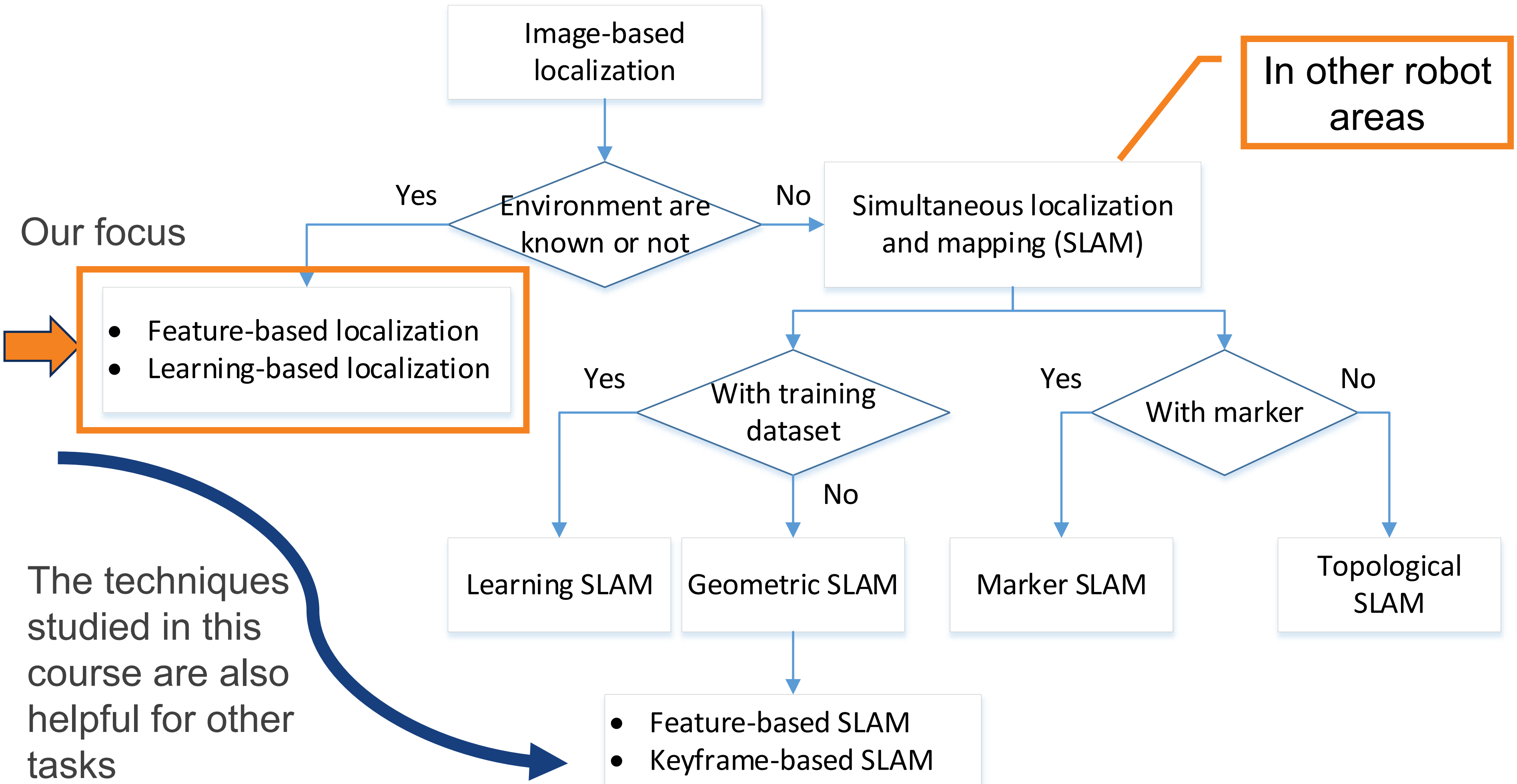
Source: Cornelia Fermüller, Path planning, CMSC498F, CMSC828K (Spring 2016), Robotics and Perception, http://users.umiacs.umd.edu/~fer/cmsc498F-828K/cmsc-498F-828K.htm

Challenge

- Lighting changes: Different time of day
- Changes in camera viewpoint
- Occlusions and ambiguous objects: People, cars, trees.

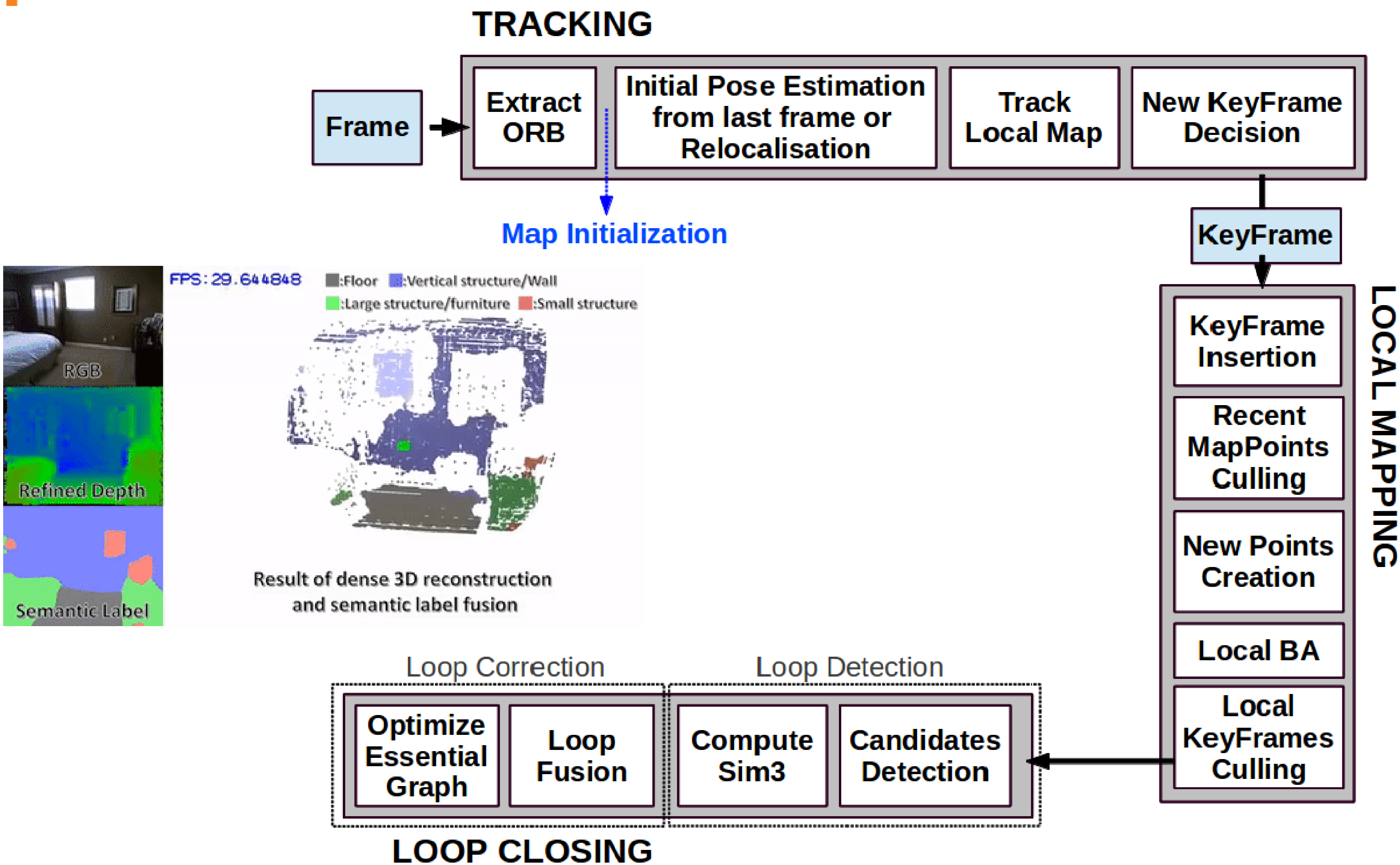
Reference: N. Piasco, D. Sidibé, C. Demonceaux, V. Gouet-Brunet, "A survey on Visual-Based Localization: On the benefit of heterogeneous data," Pattern Recognition, 2018, pp. 90-109.

Roadmap



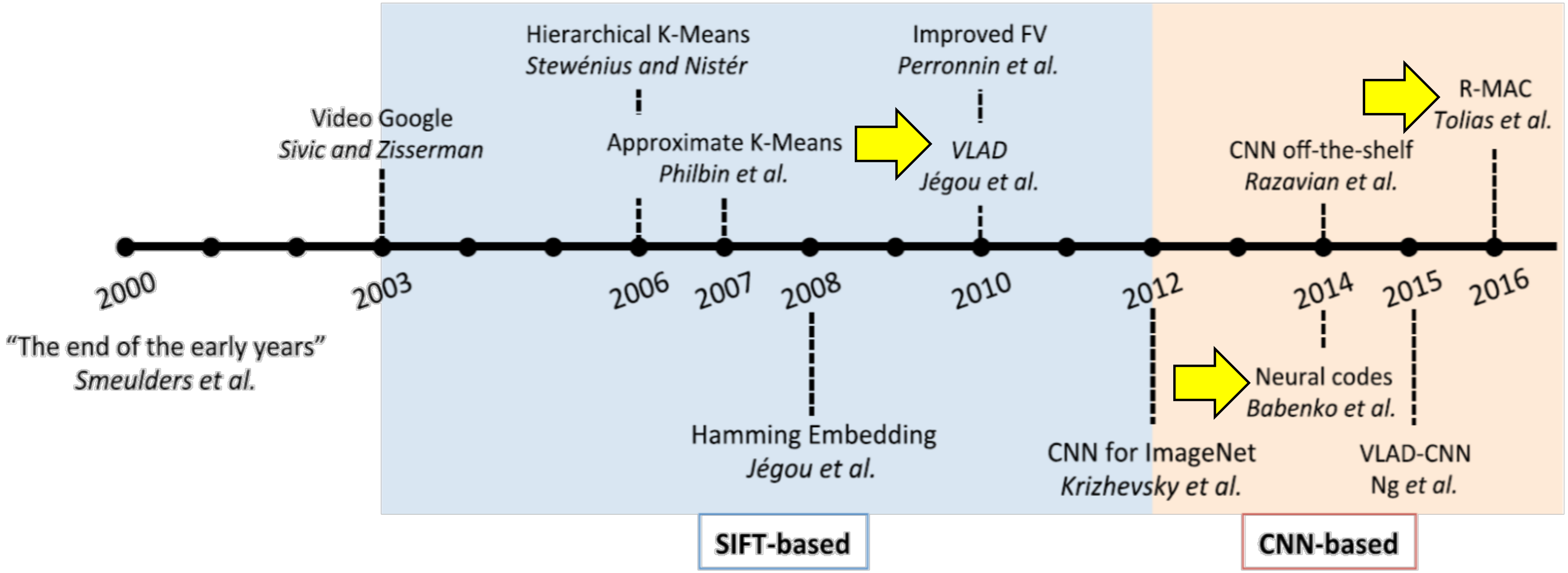
Modified from the reference: Yihong Wu, Fulin Tang, Heping Li, "Image Based Camera Localization: an Overview," Visual Computing for Industry, 2018, https://arxiv.org/abs/1610.03660

Appendix: Vision-based SLAM



Reference: ORB-SLAM: a Versatile and Accurate Monocular SLAM System, https://arxiv.org/pdf/1502.00956.pdf; http://www.luigifreda.com/2017/04/08/cnn-slam-real-time-dense-monocular-slam-learned-depth-prediction/

Place recognition



Milestones: After a survey of methods before the year 2000 [1], Video Google was proposed in 2003 [2], marking the beginning of the BoW model [3]. Although SIFT-based methods were still moving forward, CNN-based methods began to gradually take over, such as the fine-tuned CNN model for generic instance retrieval [4, 5].

Reference: L. Zheng, Y. Yang, Q. Tian, "SIFT Meets CNN: A Decade Survey of Instance Retrieval," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 40, No. 5, May 2018, pp. 1224-1244.

[1] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 12, pp. 1349-1380, Dec. 2000.

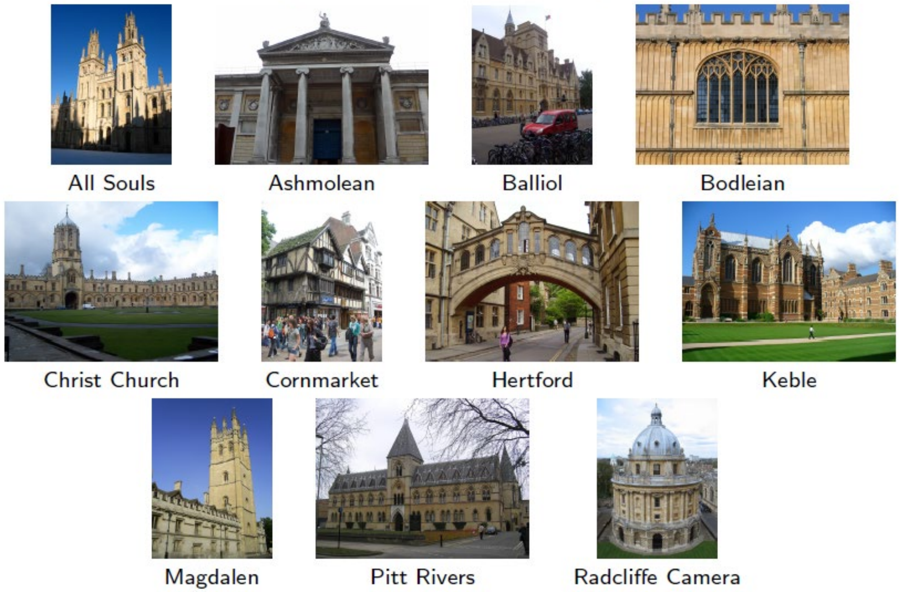
[2] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," ICCV 2003.

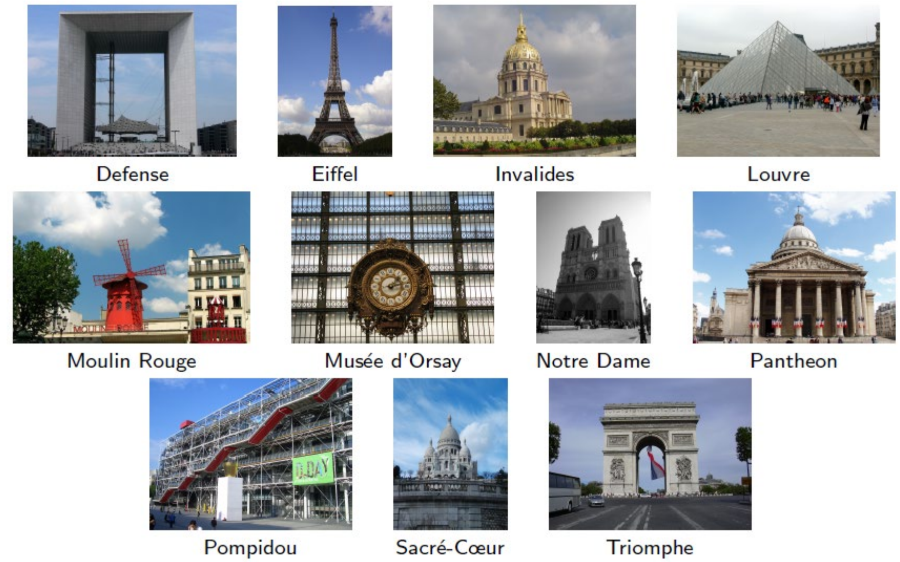
[3] H. Jegou, M. Douze, C. Schmid, and P. Perez, "Aggregating local descriptors into a compact image representation," CVPR 2010.

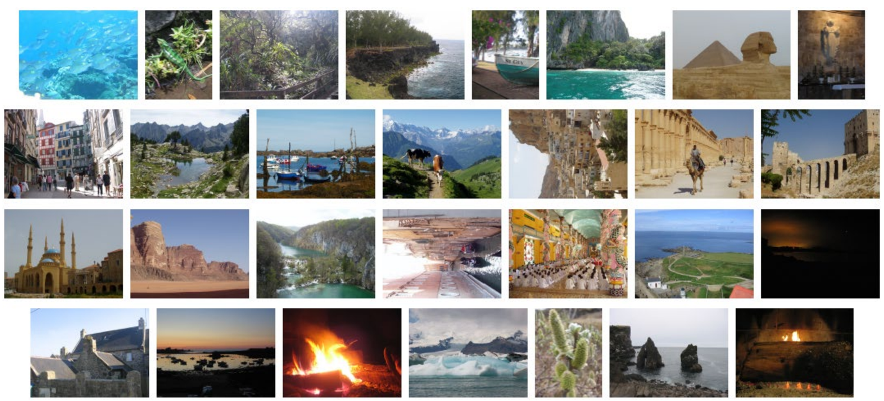
[4] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, "Neural codes for image retrieval," ECCV 2014.

[5] G. Tolias, R. Sicre, and H. Jegou, "Particular object retrieval with integral max-pooling of CNN activations," ICLR 2016.

Place recognition: Major dataset







Dataset	# image	# query	Content
Oxford5k	5,062	55	Buildings
Paris6k	6,412	55	Buildings
Holidays	1,491	500	Scene

Reference:

- J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” CVPR 2017.
- H. Jegou, M. Douze, C. Schmid, “Hamming embedding and weak geometric consistency for large scale image search,” ECCV 2008.
- J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, “Lost in quantization: Improving particular object retrieval in large scale image databases,” CVPR 2008.

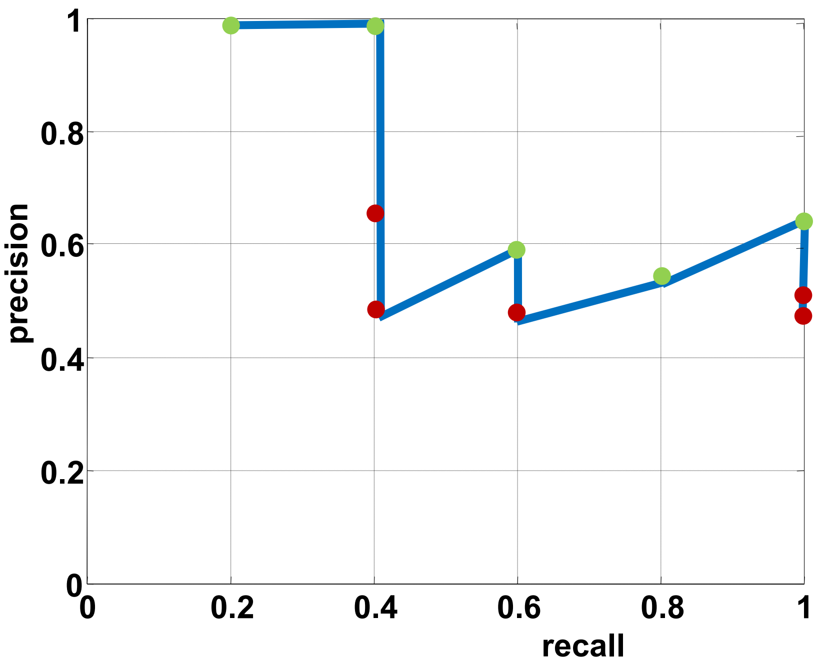
Place recognition: Performance metric

- The images in the query and the database represent scenes rather than objects (e.g. street view panorama, buildings images, indoor scenes).
- The performance of such system is evaluated according to the precision rate rather than the recall rate (i.e. a perfect place recognition system should recover in its top ranked candidates documents that display the exact location of the query).

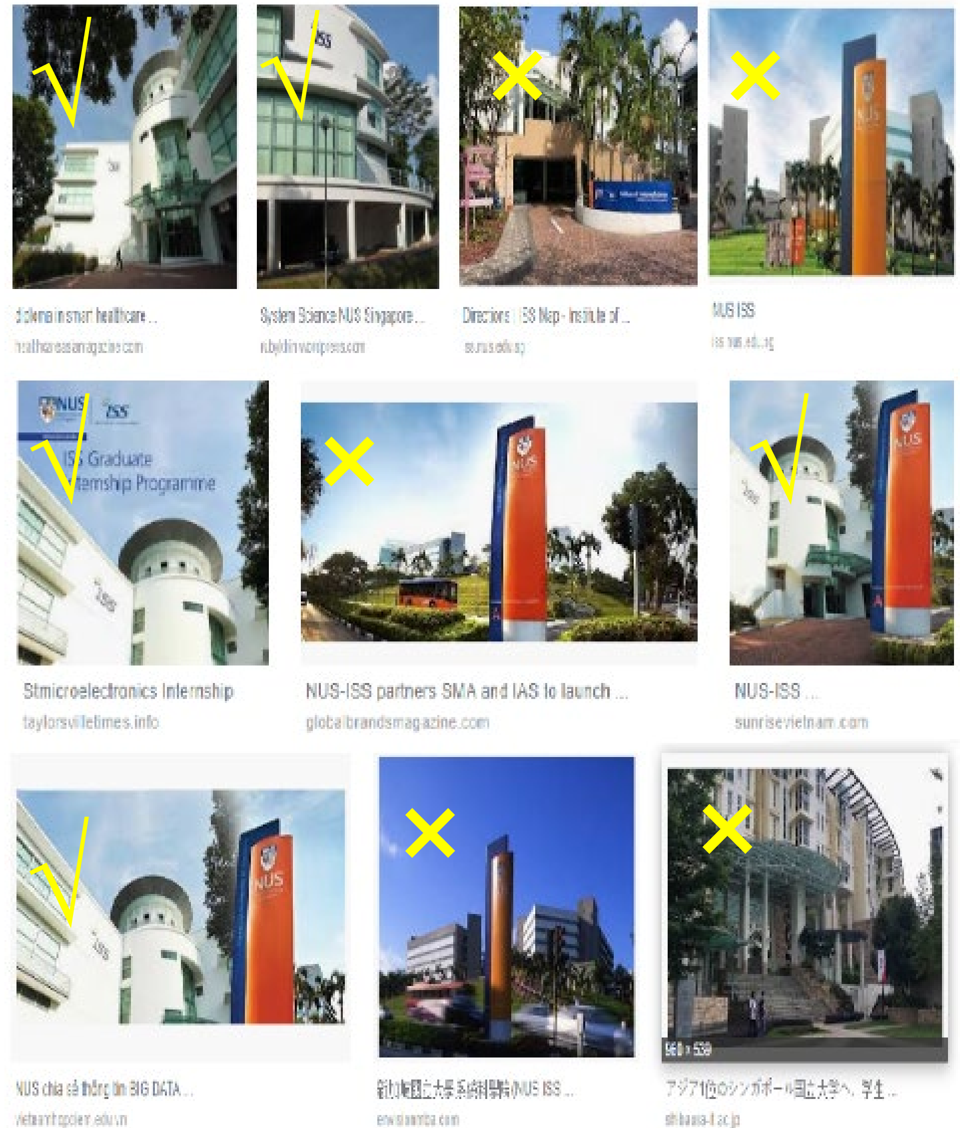
Performance metric

Query image (input)

Precision = #relevant / #returned
Recall = #relevant / #total relevant



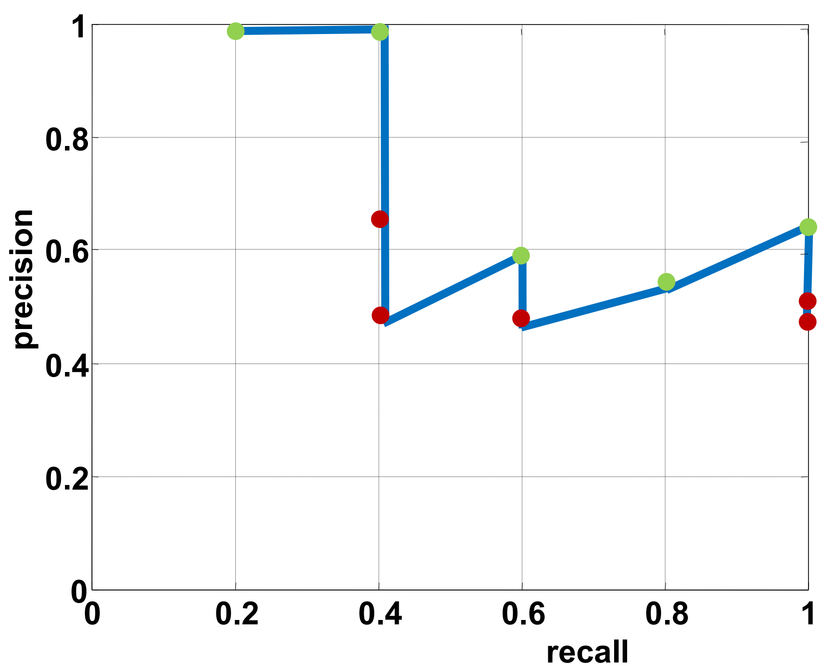
Returned results (ranked)



Performance metric

Ranked list of returned results with True/False labels (in previous slide example).

K	1	2	3	4	5	6	7	8	9	10
Label	T	T	F	F	T	F	T	T	F	F
TP	1	2	2	2	3	3	4	5	5	5
P	1	1	2/3	2/4	3/5	3/6	4/7	5/8	5/9	5/10
GTP	Supposed to be 5 for this query. It depends on dataset.									



- K: current rank
- TP: true positives
- P: precision $= {}^{TP}/_{K}$
- GTP: total number of ground truth positives in the dataset

Summation of precision values of correct results / GTP

$$\frac{\left(1+1+\frac{3}{5}+\frac{4}{7}+\frac{5}{8}\right)}{5}$$

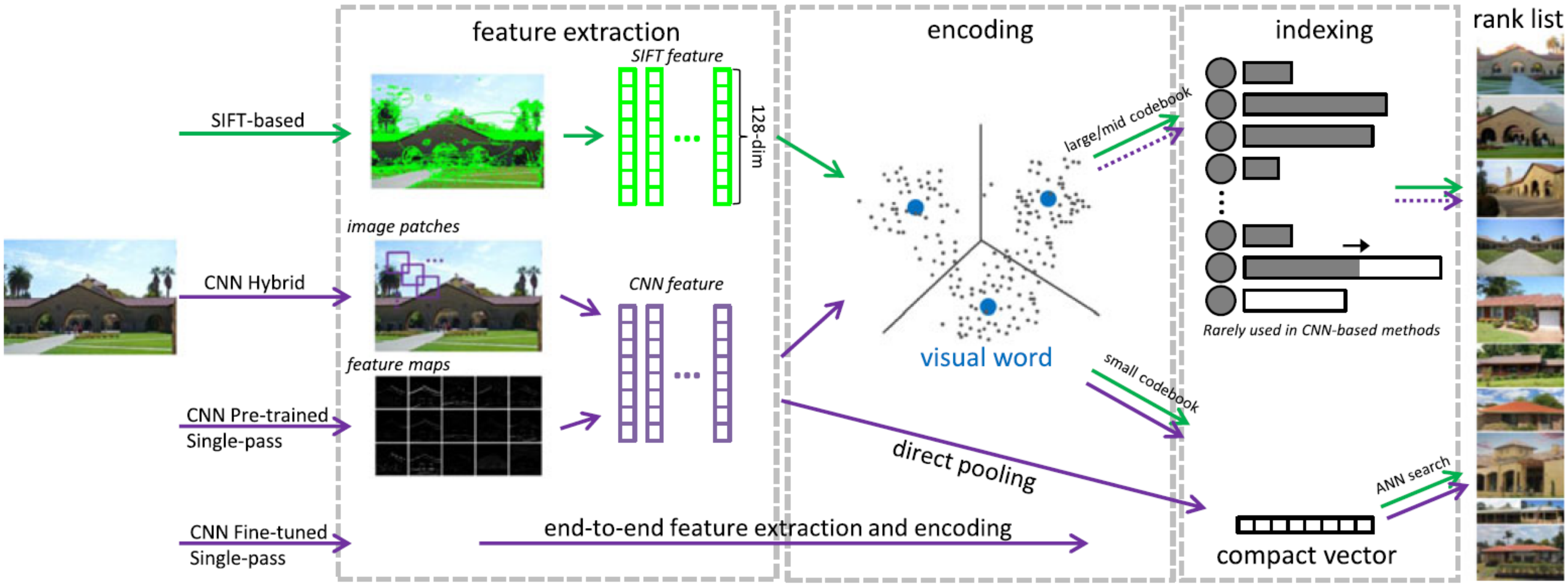
- Average precision = average precision (for a single query)
- Mean average precision (mAP) = mean of average precision over all queries

Topics

- Introduction to image-based location and place recognition
- Place recognition pipeline
  - Feature extraction
  - Feature encoding
  - Feature indexing
- Workshop on place recognition

Place recognition pipeline (1)

Feature extraction → Feature encoding (dictionary) → Feature indexing (search)

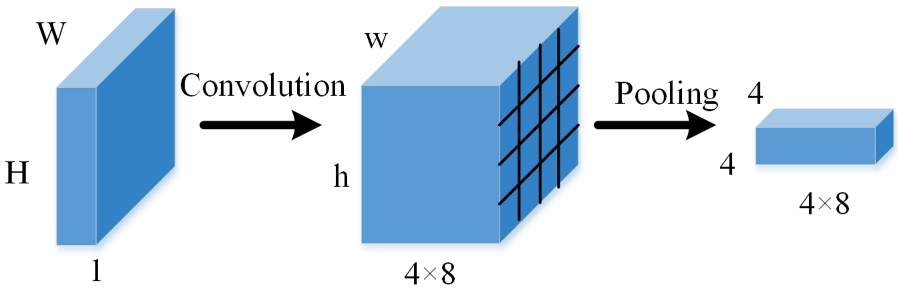


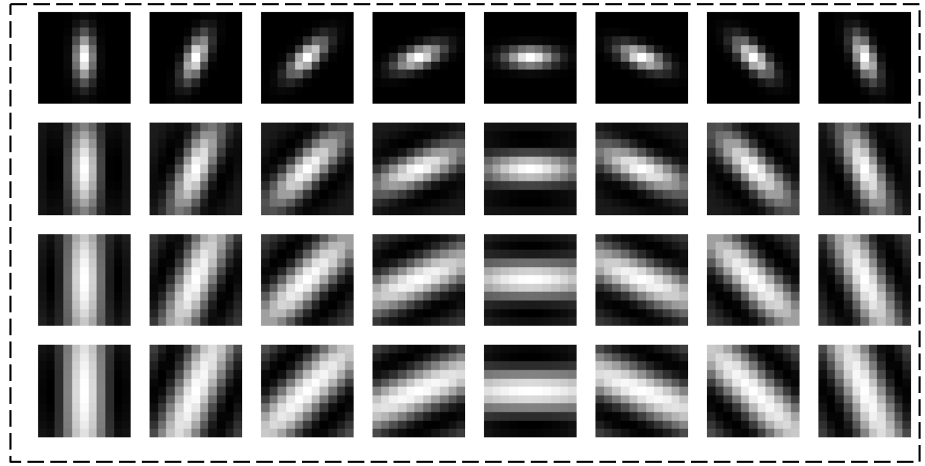
Reference: L. Zheng, Y. Yang, Q. Tian, “SIFT Meets CNN: A Decade Survey of Instance Retrieval,” IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 40, No. 5, May 2018, pp. 1224-1244.

Feature extraction

Features	Remark
Global feature: GIST	Following slides
Point feature: SIFT, SURF	Previous day course
Point feature: ORB	Following slides
Patch (blob) feature HoG, LBP	Vision Systems course
Learned feature: CNN-based	Following slides

Global feature: GIST





- Given an input image, a GIST descriptor is computed by <u>convolving</u> the image with 32 <u>Gabor filters</u> (at 4 scales, 8 orientations), producing 32 feature maps of the same size of the input image.
- Divide each feature map into 16 cells (by a 4×4 grid), and then <u>average</u> the feature values within each cell.
- <u>Concatenate</u> the 16 averaged values of all 32 feature maps, resulting in a $16 \times 32 = 512$ GIST descriptor.
- Intuitively, GIST summarizes the gradient information (scales and orientations) for different parts of an image.

8	orientations
4	scales
<u>x 16</u>	bins
512	dimensions

Reference: A. Olivia and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," IJCV, 2001.

Point feature: ORB (Oriented FAST and rotated BRIEF)

FAST (Features from accelerated segment test)

- Objective: Determine a pixel p (intensity value I_p) in the image as an interest point or not based on its neighboring pixels (say a circle of 16 pixels).
- Determine the pixel p is a corner, if there exists a set of n continuous pixels in the circle (of 16 pixels) which are all brighter than $I_p + t$, or all darker than $I_p - t$, with an appropriate threshold value t.
- Faster version: First compare the intensity of pixels 1, 5, 9 and 13 of the circle with I_p. At least three of these four pixels should satisfy the threshold criterion so that the interest point will exist.
  - If at least three of the four-pixel values I_1, I_5, I_9, I_{13} are not above or below $I_p + t$, then p is not an interest point (corner). In this case reject the pixel p as a possible interest point.
  - Else: check all 16 pixels and check if 12 contiguous pixels fall in the criterion.

Rotation calibration: It computes the intensity weighted centroid of the patch with located corner at center. The direction of the vector from this key point to centroid gives the orientation.

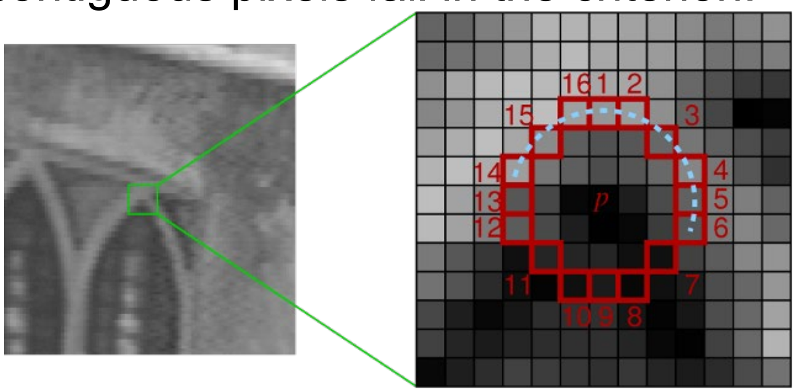
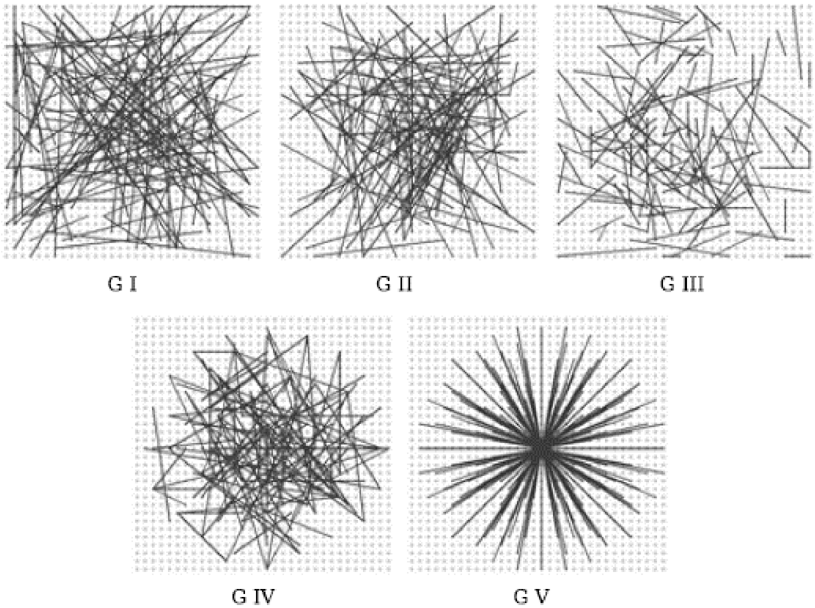


Photo: https://medium.com/software-incubator/introduction-to-orb-oriented-fast-and-rotated-brief-4220e8ec40cf

Point feature: ORB (Oriented FAST and rotated BRIEF)

Brief (Binary robust independent elementary feature)

- Sample a pair of pixels a and b, according to sampling geometry patterns (five figures at below).
- A vector of binary code: 1, if $a > b$, else 0.
- Dimension of this feature: Number of pairs



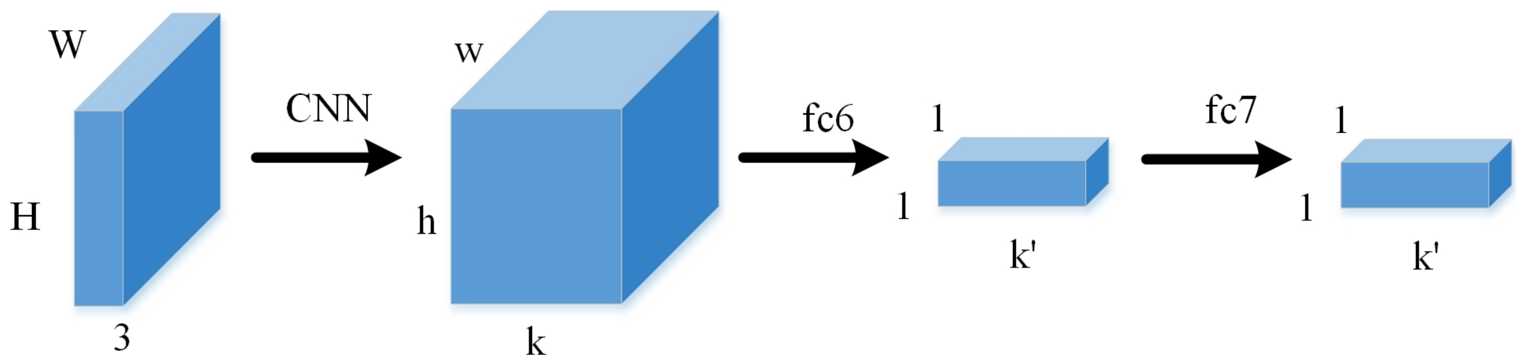
Reference:

- https://docs.opencv.org/3.4/d1/d89/tutorial_py_orb.html
- https://medium.com/@deepanshut041/introduction-to-orb-oriented-fast-and-rotated-brief-4220e8ec40cf
- E. Rublee, V. Rabaud, K. Konolige and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," ICCV 2011, pp. 2564-2571.

CNN: Neural code

- Use of feature activation from the top layers of CNN network as high level descriptor
- 3-channel RGB input, 227×227
- AlexNet last pooling layer, global descriptor of dimension $w \times h \times k = 6 \times 6 \times 256 = 9216$
- Alternatively, fully connected layers fc_6, fc_7, global descriptors of dimension $k' = 4096$

Appendix, full (simplified) AlexNet architecture:
[227x227x3] INPUT
[55x55x96] CONV1: 96 11x11 filters at stride 4, pad 0
[27x27x96] MAX POOL1: 3x3 filters at stride 2
[27x27x96] NORM1: Normalization layer
[27x27x256] CONV2: 256 5x5 filters at stride 1, pad 2
[13x13x256] MAX POOL2: 3x3 filters at stride 2
[13x13x256] NORM2: Normalization layer
[13x13x384] CONV3: 384 3x3 filters at stride 1, pad 1
[13x13x384] CONV4: 384 3x3 filters at stride 1, pad 1
[13x13x256] CONV5: 256 3x3 filters at stride 1, pad 1
[6x6x256] MAX POOL3: 3x3 filters at stride 2
[4096] FC6: 4096 neurons
[4096] FC7: 4096 neurons
[1000] FC8: 1000 neurons (class scores)



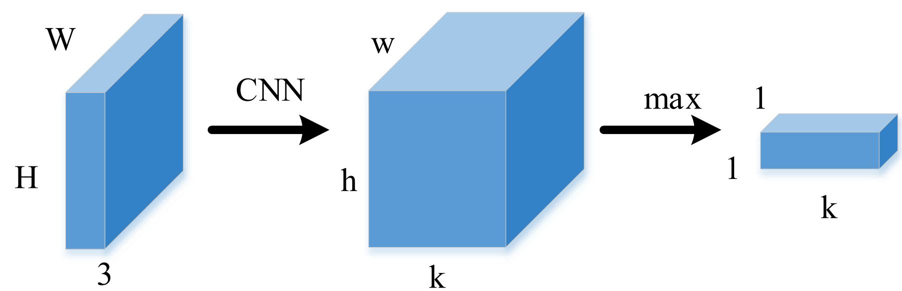
Reference: A. Babenko, A. Slesarev, A. Chigorin, V. Lempitsky, "Neural Codes for Image Retrieval," ECCV 2014, https://arxiv.org/abs/1404.1777

CNN: Maximum activations

Maximum activations of convolutions (MAC)

- Given a set of 2D convolutional feature channel responses $X = \{X_i\}, i = 1, 2, \cdots k$, spatial max-pooling over all location is given as $f = [f_{\Omega,1}, \cdots, f_{\Omega,k}]$, where $f_{\Omega,i} = \max_{p\in\Omega} X_i(p)$, Ω is the set of valid spatial locations, $X_i(p)$ is the response at the particular position p, k is the number of feature channels

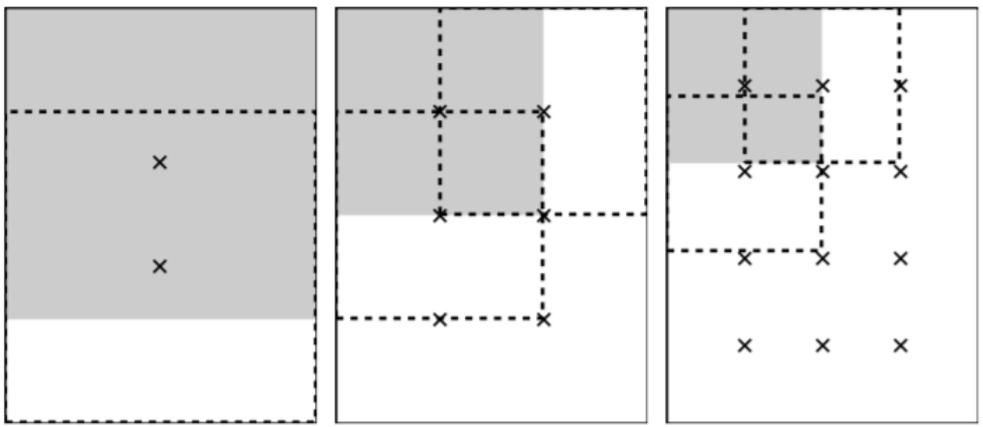
Global feature vector
(max-pooling per activation map)

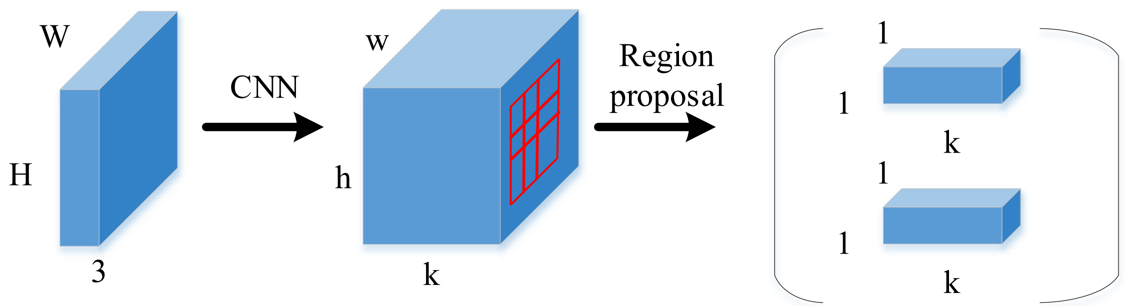


Reference: G. Tolias, R. Sicre, H. Jégou, "Particular object retrieval with integral max-pooling of CNN activations," ICLR 2016, https://arxiv.org/abs/1511.05879

CNN: Maximum activations

- Sampling region: Sample regions extracted at 3 different scales. We show the top-left region of each scale (gray colored region) and its neighbouring regions towards each direction (dashed borders). The cross indicates the region centre.
- Regional feature vector: Fixed multi-scale overlapping spatial region pooling.

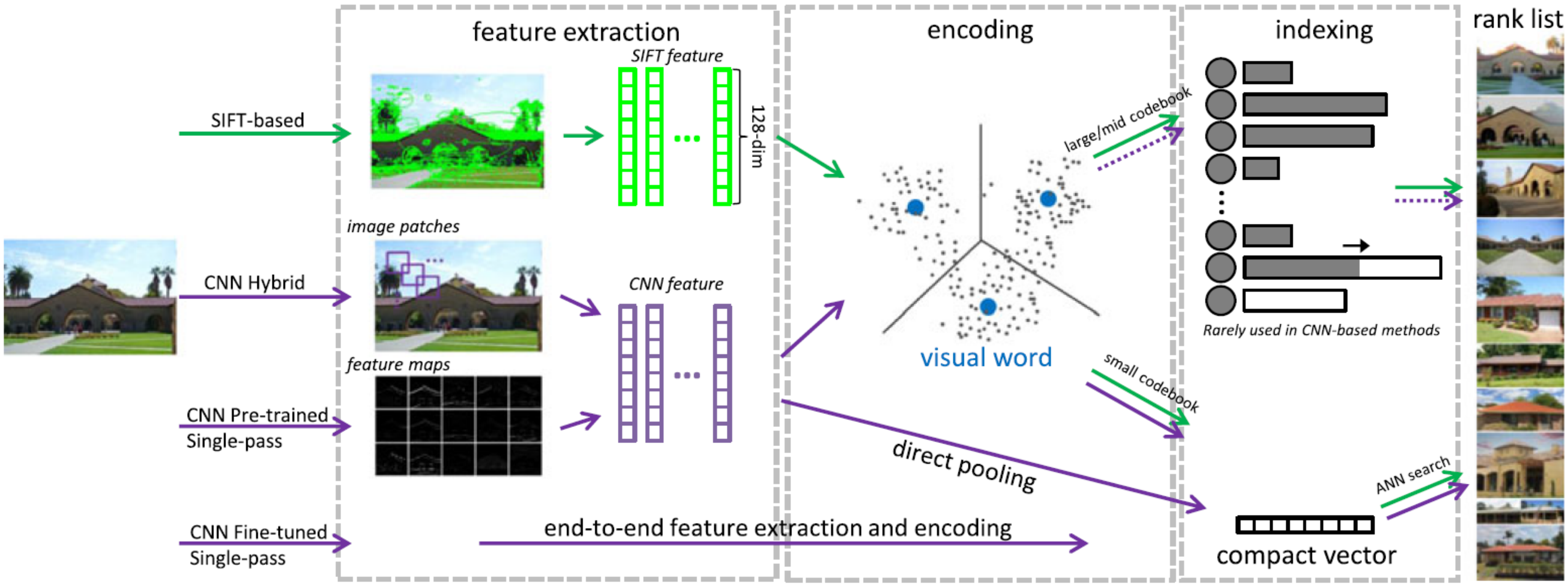




Reference: G. Tolias, R. Sicre, H. Jégou, "Particular object retrieval with integral max-pooling of CNN activations," ICLR 2016, https://arxiv.org/abs/1511.05879

Place recognition pipeline (2)

Feature extraction → Feature encoding (dictionary) → Feature indexing (search)

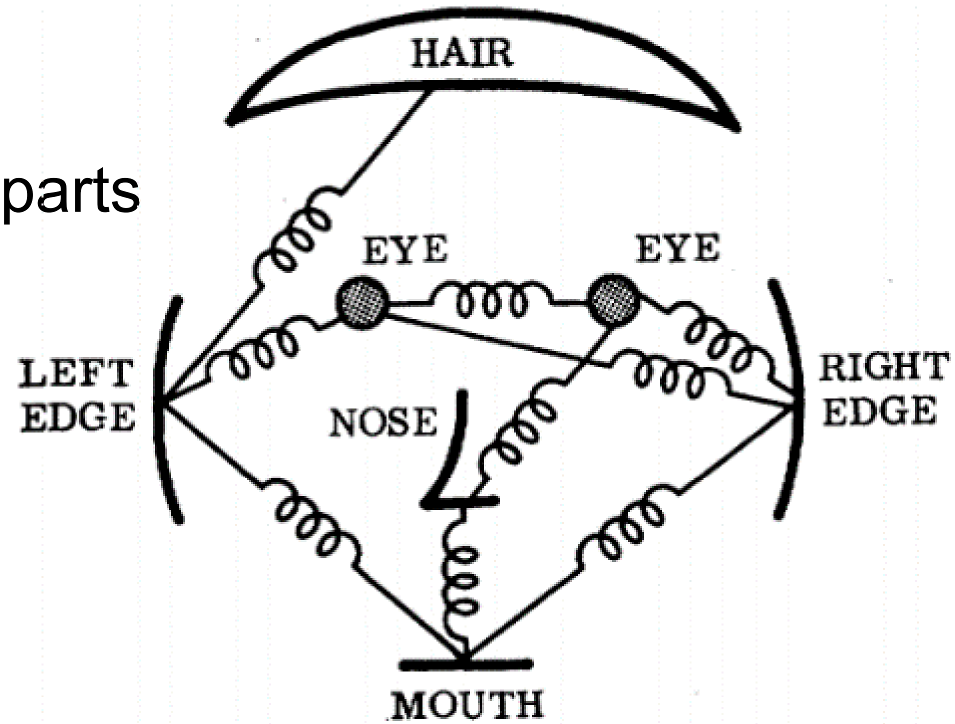


Reference: L. Zheng, Y. Yang, Q. Tian, “SIFT Meets CNN: A Decade Survey of Instance Retrieval,” IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 40, No. 5, May 2018, pp. 1224-1244.

Intuition: Part model

Model

- Object as a set of parts
- Relative locations between parts
- Appearance of part



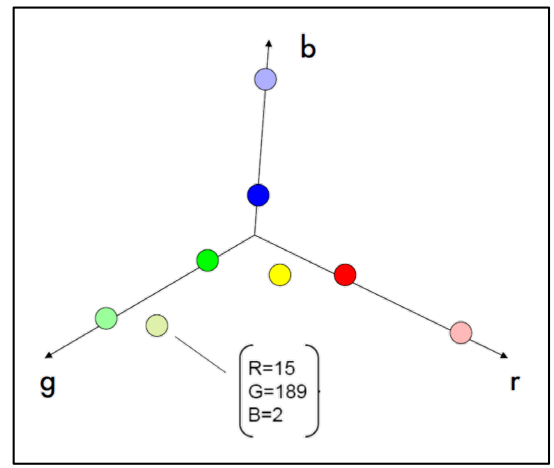
Reference: M. A. Fischler, and R. A. Elschlager, "The representation and matching of pictorial structures," IEEE Trans. on Computer, Vol. 22, No. 1, 1973, pp. 67-92, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.118.7951&rep=rep1&type=pdf

Intuition: Histogram

- Consider a histogram h over integers $C = \{0,1,2,3,4\}$, computed from the following samples.
- Each sample is encoded (hard assigned into one vector, all such vectors are pooled (averaged) into one vector.
- C is a codebook or vocabulary.

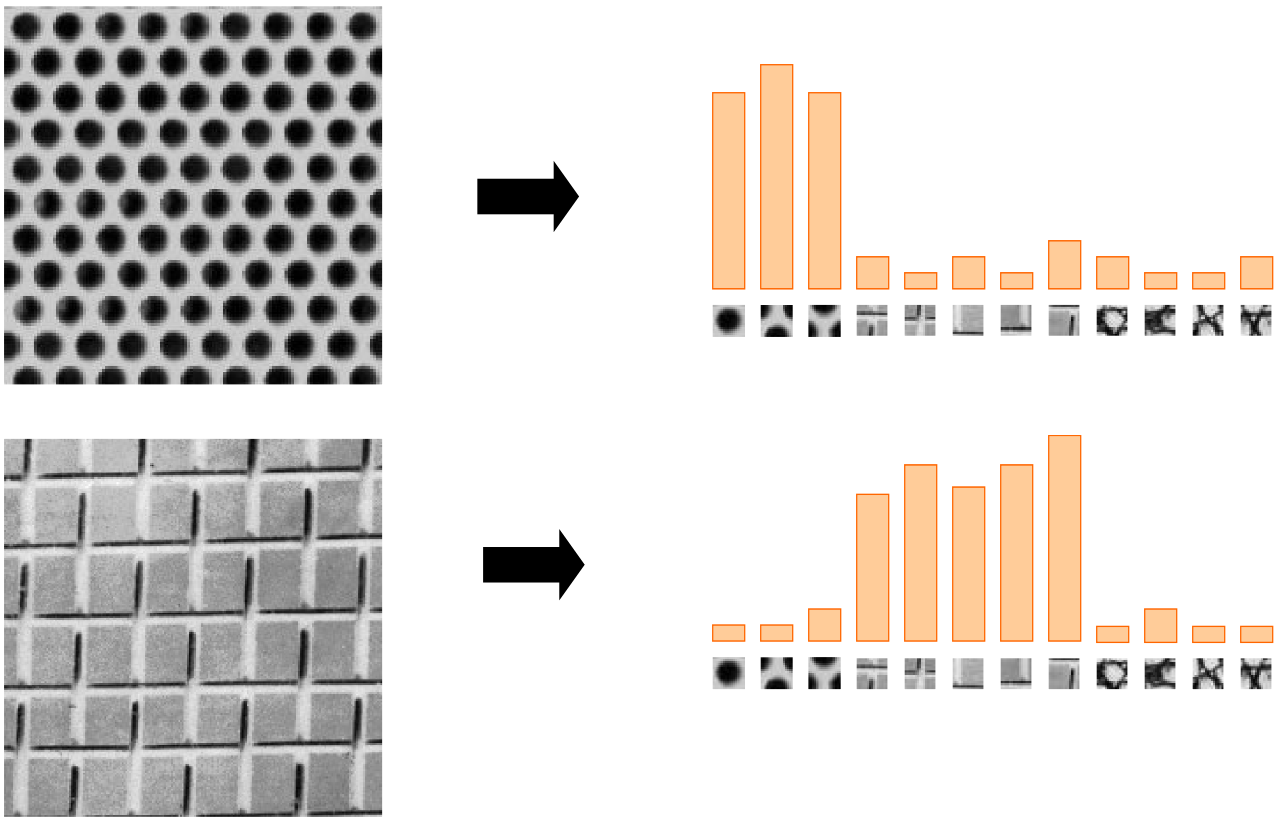
$$
\begin{array}{ccccccccccc}
C & = & \{ & 0 & 1 & 2 & 3 & 4 & \} & & \\
\hline
3 & \rightarrow & (& 0 & 0 & 0 & 1 & 0 &) & & \\
2 & \rightarrow & (& 0 & 0 & 1 & 0 & 0 &) & & \\
0 & \rightarrow & (& 1 & 0 & 0 & 0 & 0 &) & & \\
3 & \rightarrow & (& 0 & 0 & 0 & 1 & 0 &) & & \\
2 & \rightarrow & (& 0 & 0 & 1 & 0 & 0 &) & & \\
2 & \rightarrow & (& 0 & 0 & 1 & 0 & 0 &) & + & \\
\hline
h & = & (& 1 & 0 & 3 & 2 & 0 &) & / & 6
\end{array}
$$

An example on color space



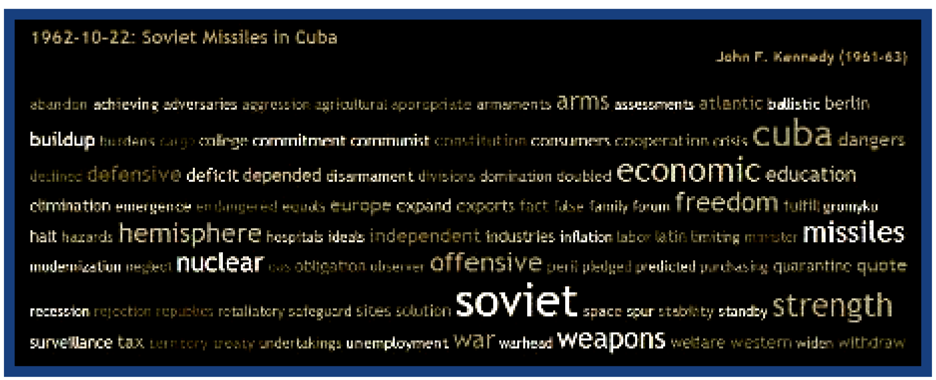
Intuition: Texture recognition

- Texture is characterized by the repetition of basic elements or *textons*. For stochastic textures, it is the identity of the textons, not their spatial arrangement.



Intuition: Bag-of-words models

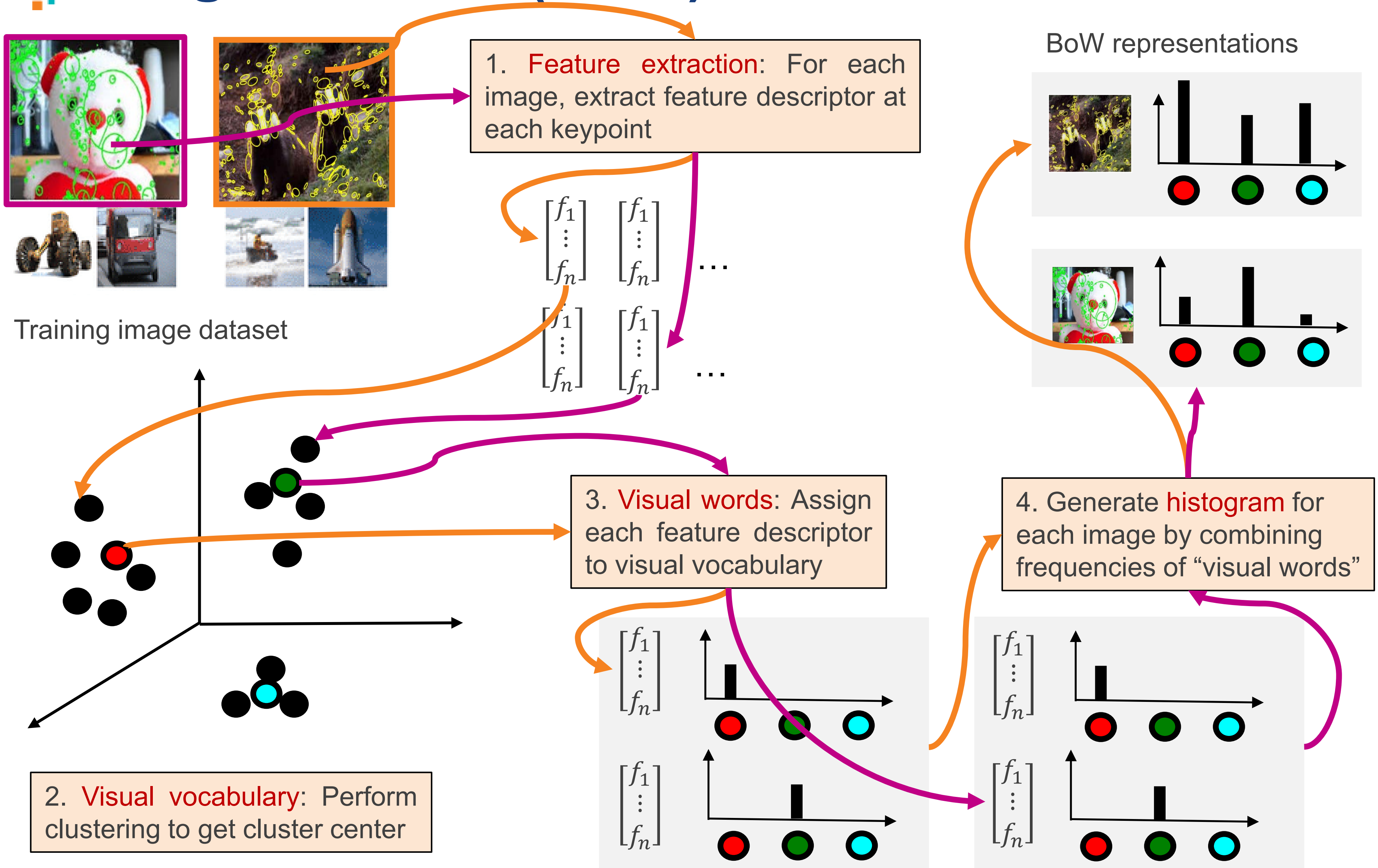
- Orderless document representation: frequencies of words from a dictionary.



Reference:

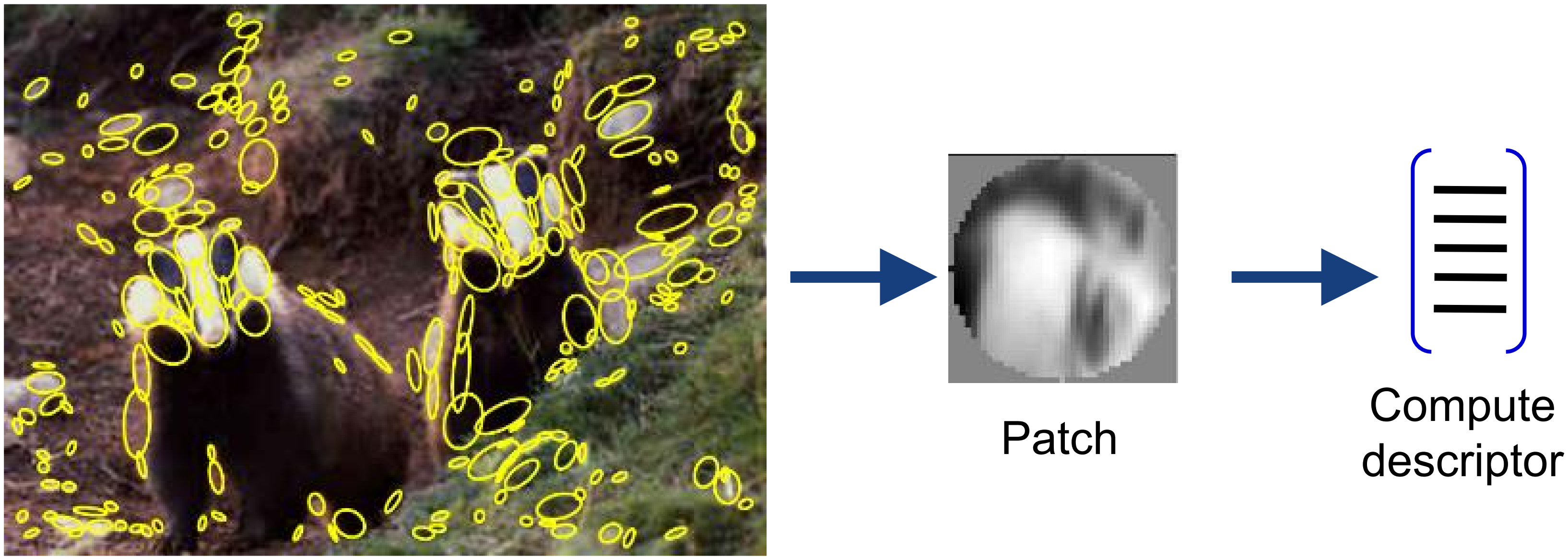
1. G. Salton and M. J. McGill, Introduction to Modern Information Retrieval, 1986
2. US Presidential Speeches Tag Cloud, http://chir.ag/phernalia/preztags/

Bag-of-words (BoW): Overview

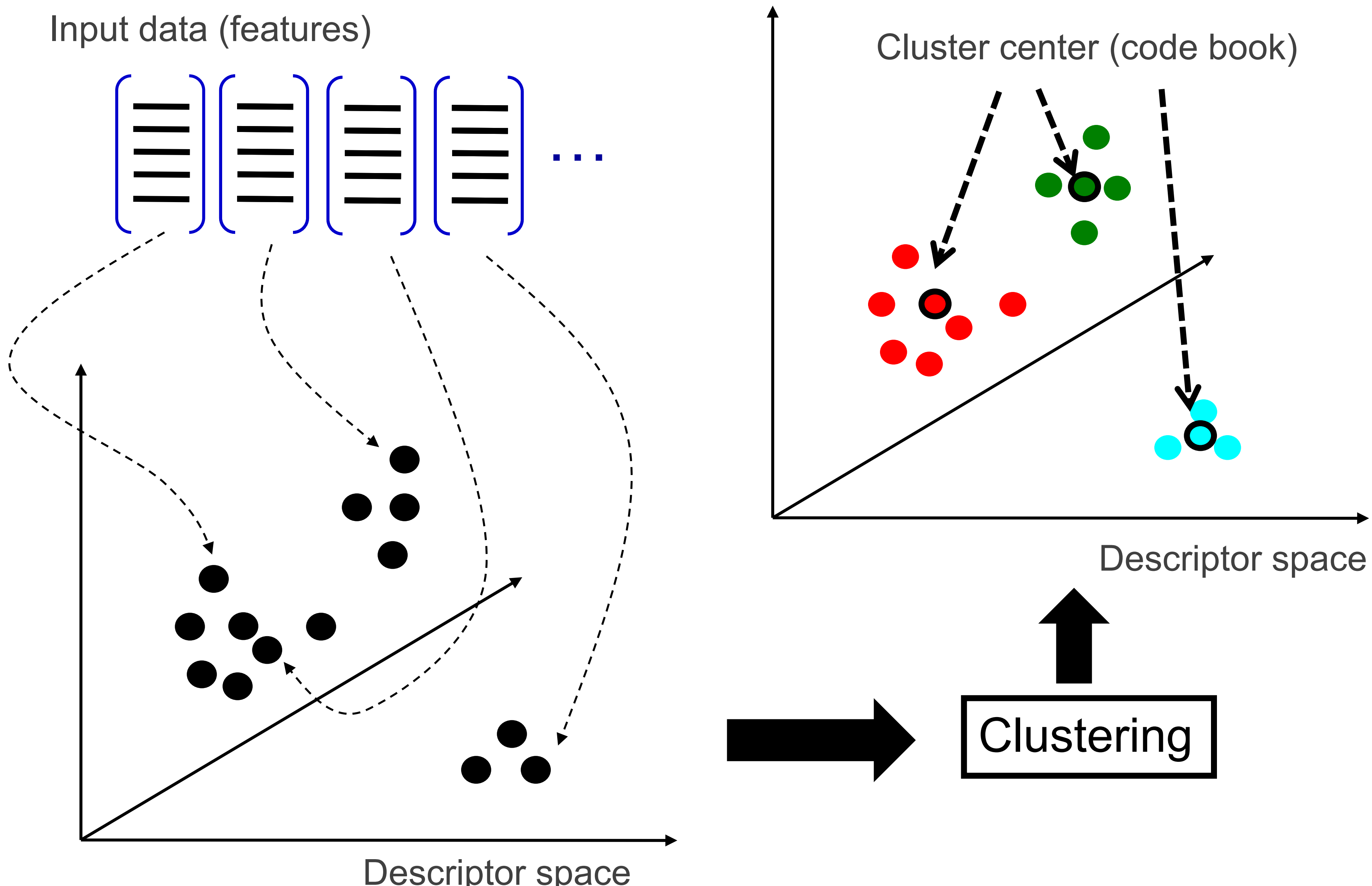


BoW: Feature extraction

- Interest points



BoW: Learn visual vocabulary



BoW: Clustering

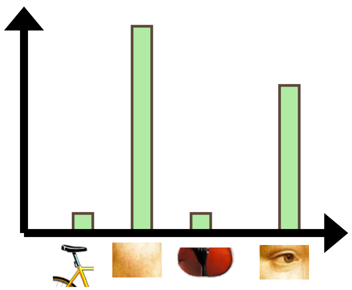
- Clustering is a common method for learning a visual vocabulary or codebook
  - Unsupervised learning process
  - Each cluster center produced by k-means becomes a codevector
  - Codebook can be learned on separate training set

- The codebook is used for quantizing features
  - A vector quantizer takes a feature vector and maps it to the index of the nearest codevector in a codebook
  - Codebook = visual vocabulary
  - Codevector = visual word

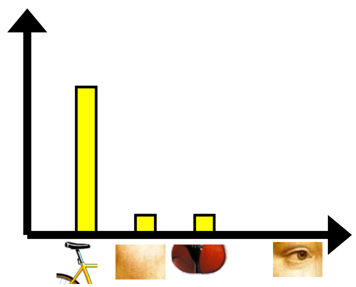
BoW: Similarity evaluation

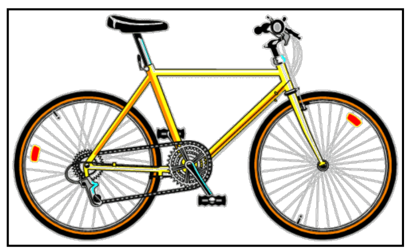
- Evaluate similarity of two images based on their BoW representations

$$\mathbf{p} = [1, 8, 1, 4]$$



$$\mathbf{q} = [5, 1, 1, 0]$$





Histogram Intersection

$$\mathbf{H}_1 = (10, 0, 0, 0, 100, 10, 30, 0, 0)$$
$$\mathbf{H}_2 = (0, 40, 0, 0, 0, 6, 0, 110, 0)$$
$$S = \sum_{i=1}^{N} \min(H_1(i), H_2(i)) = 6$$

Euclidean distance

$$\mathbf{H}_1 = (10, 0, 0)$$
$$\mathbf{H}_2 = (0, 40, 0)$$
$$S = \sqrt{\sum_{i=1}^{N} (H_1(i) - H_2(i))^2} = 41.23$$

Manhatten distance

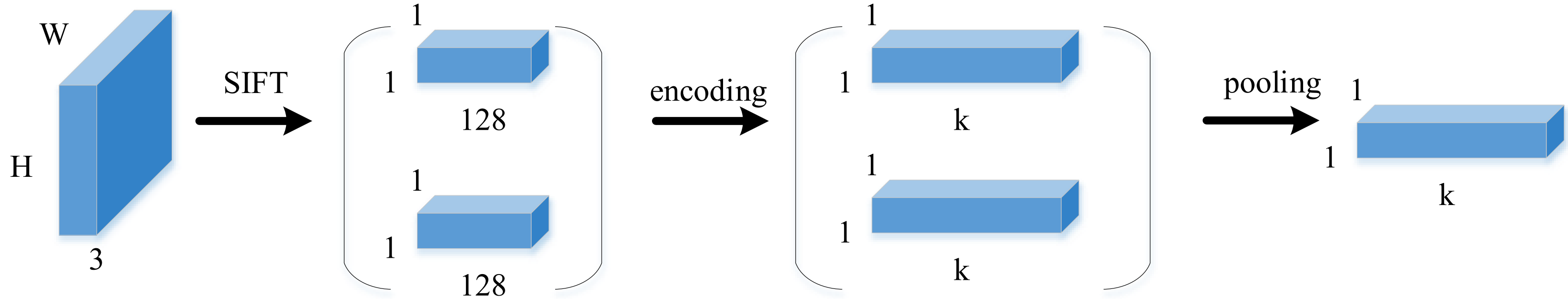
$$\mathbf{H}_1 = (10, 0, 0)$$
$$\mathbf{H}_2 = (0, 40, 0)$$
$$S = \sum_{i=1}^{N} |H_1(i) - H_2(i)| = 50$$

Reference: http://vision.cs.utexas.edu/376-spring2018/slides/lecture18-spring2018.pdf

BoW: Example using SIFT or CNN

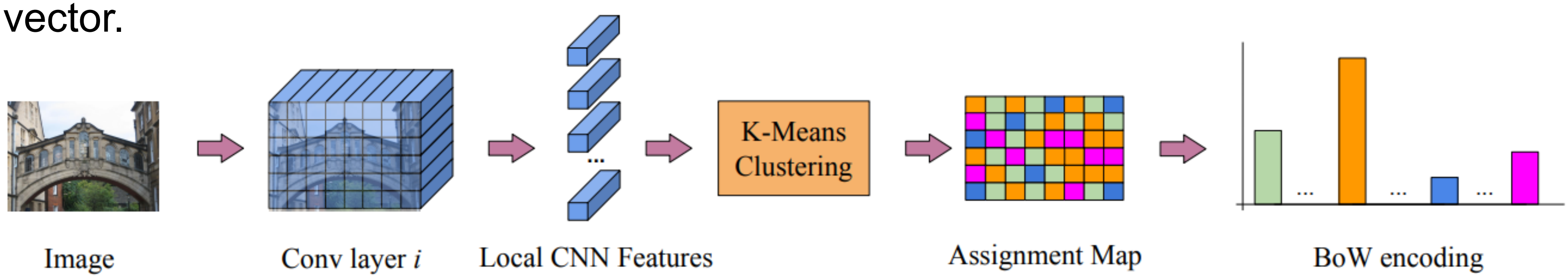
Example: SIFT

Given a gray–scale image input. $N \times 128$ descriptors (N is number of key points, 128 is the SIFT dimensions). Clustering/encoding (hard assignment) on k visual words). Note that N and k are user defined.



Example: CNN

Use bag of words encode the local convolutional features of an image into a single vector.



Reference: E. Mohedano, K. McGuinness, N. O'Connor, A. Salvador, F. Marques, "Bags of Local Convolutional Features for Scalable Instance Search," ICMR 2016, https://arxiv.org/abs/1604.04653

VLAD: Vector of Locally Aggregated Descriptors

Given a codebook $X = \{x_t, t = 1, \cdots, T\}$, $\{\mu_i, i = 1, \cdots, N\}$, learned with K-means, and a set of local descriptors

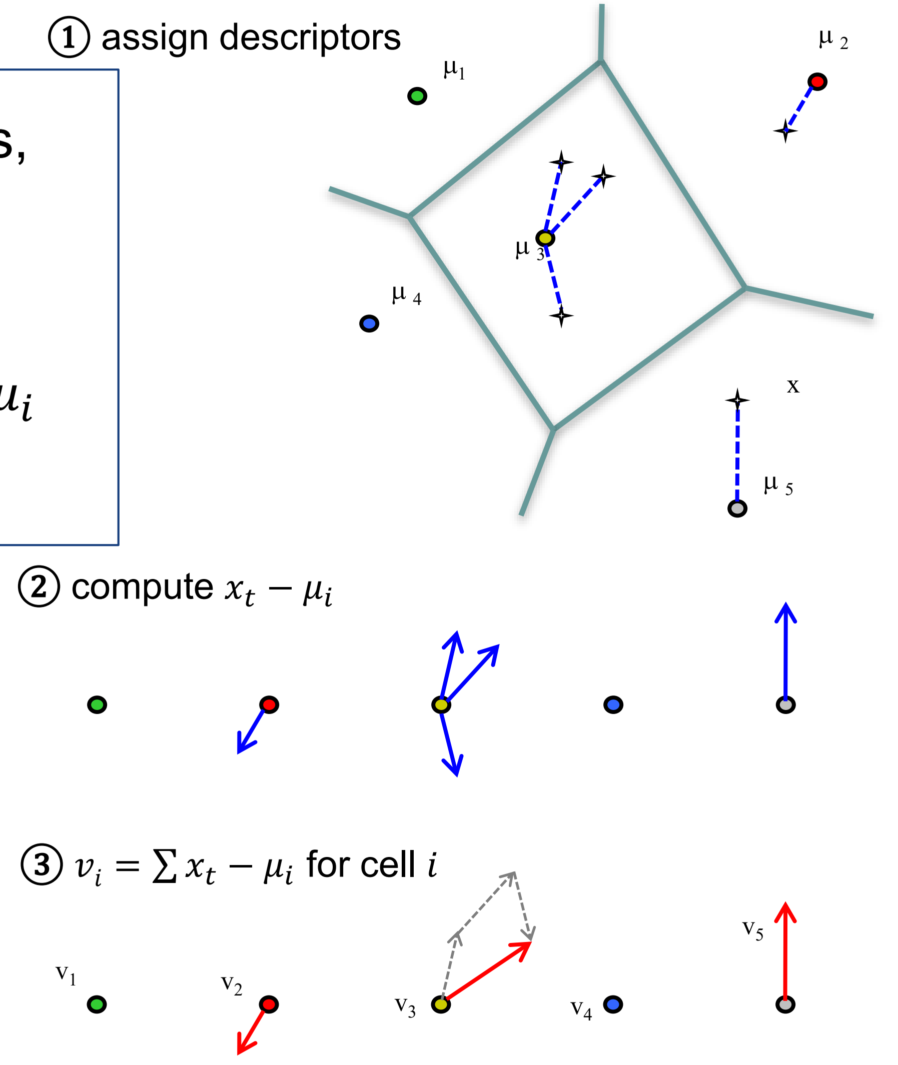
- ① assign: $NN(x_t) = \arg\min_{\mu_i} \|x_t - \mu_i\|$
- ②③ compute: $v_i = \sum_{x_t : NN(x_t) = \mu_i} x_t - \mu_i$
- concatenate v_i

0/1 assignment of x_t to cluster i

$$v_i = \sum_t \overbrace{a_i(x_t)} \underbrace{(x_t - c_i)}$$

Residual vector

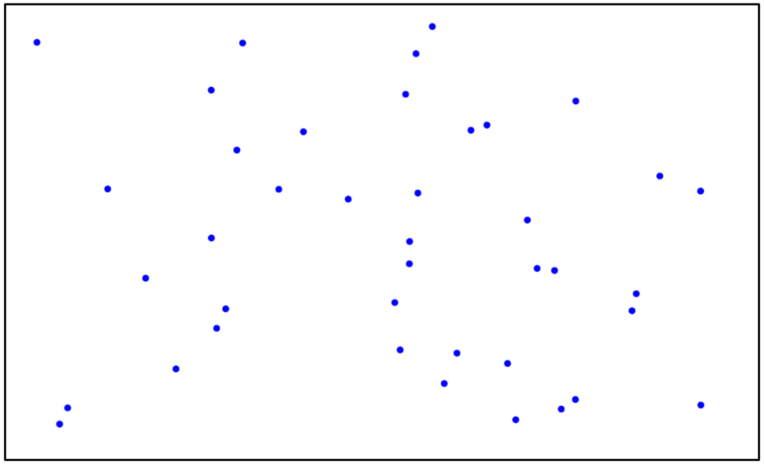
Sum over all (blue) descriptors in each cell. Then, all (red) residual vectors are normalized.



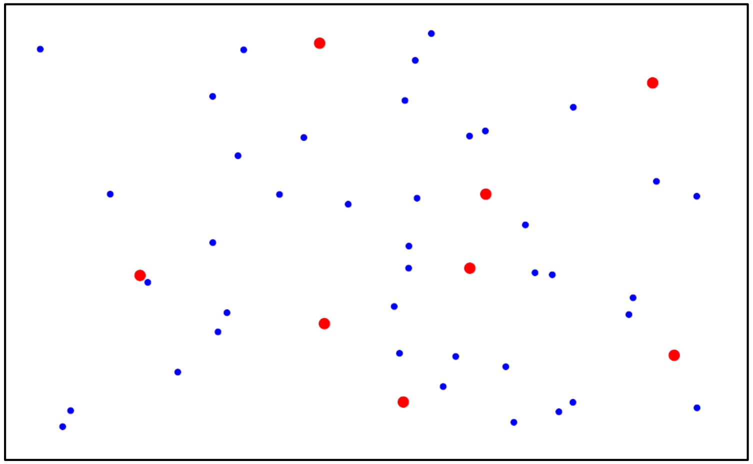
Reference: H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, "Aggregating local image descriptors into compact codes," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 34, No. 9, 2012, pp.1704-1716. https://hal.inria.fr/inria-00633013/document/

VLAD: Example

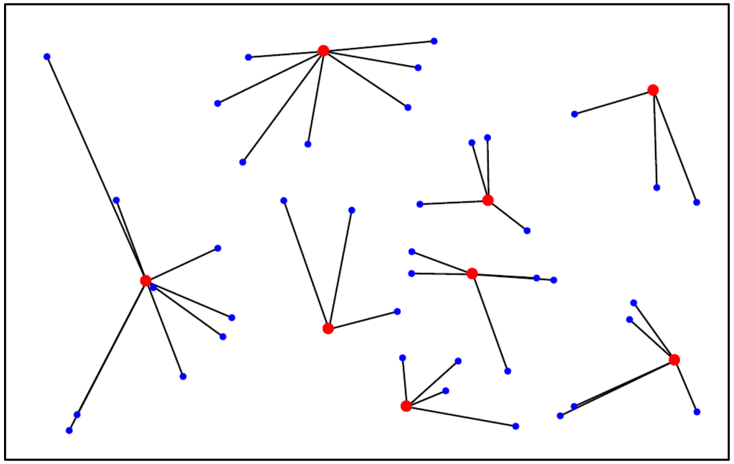
Input vectors



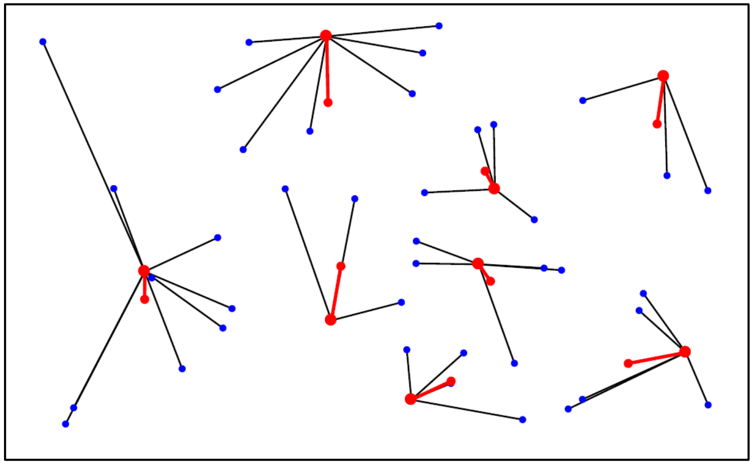
Codebook



Residuals

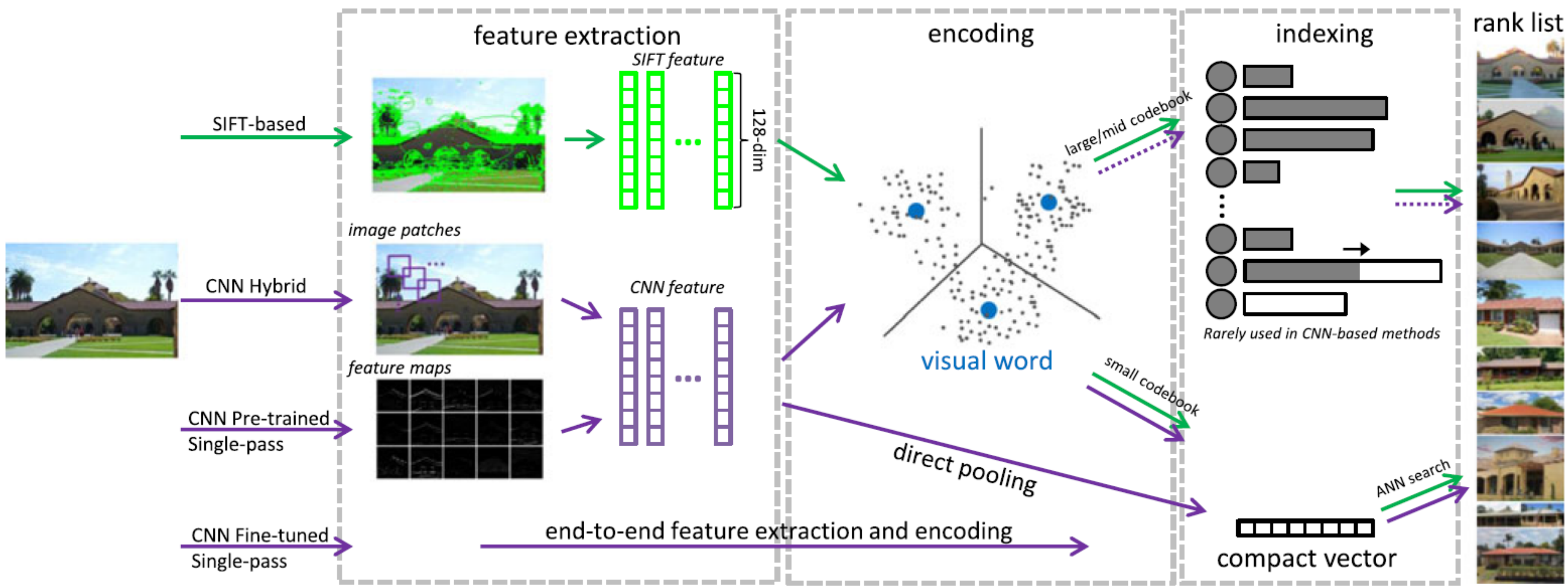


Pooling



Place recognition pipeline (3)

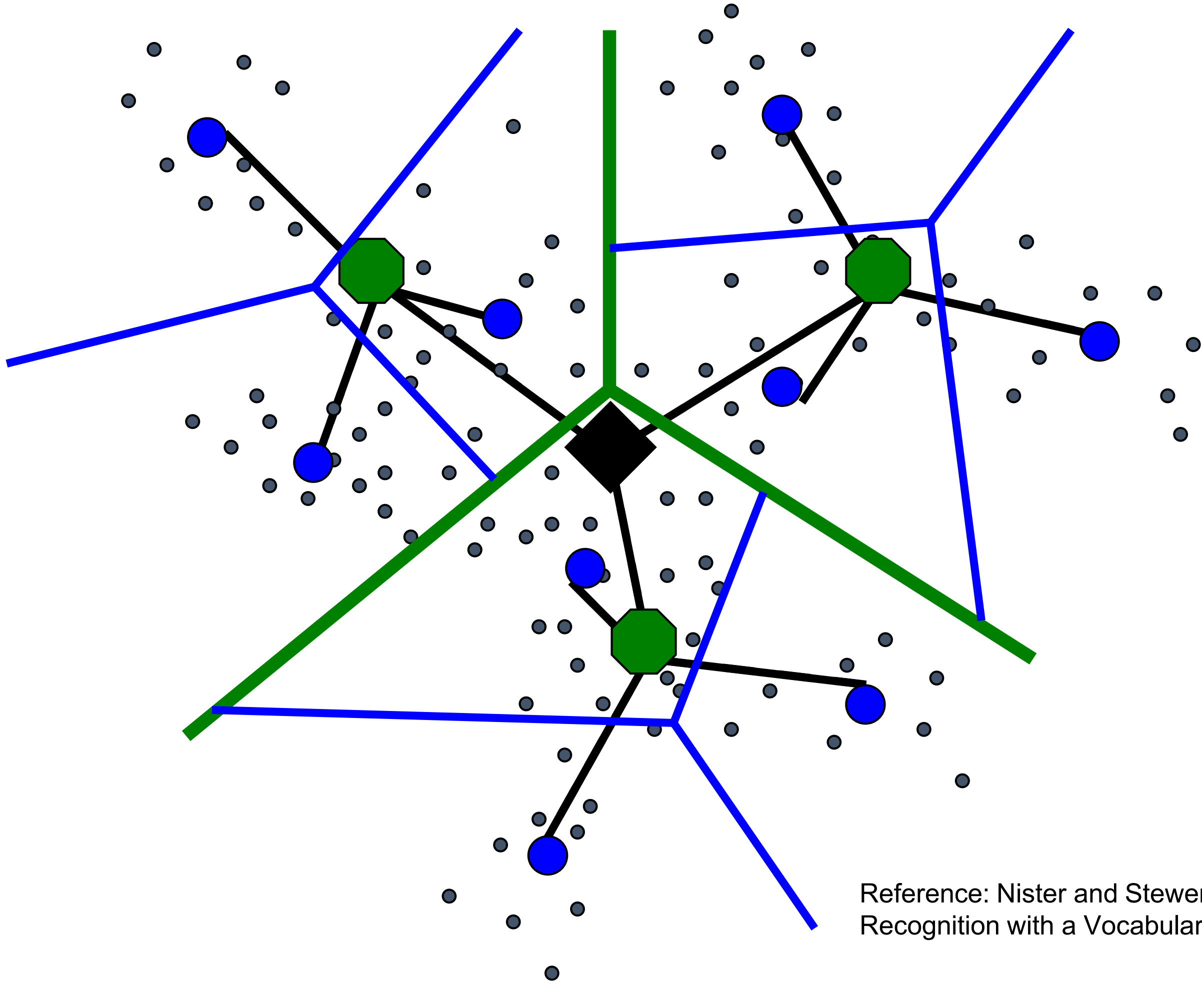
Feature extraction → Feature encoding (dictionary) → Feature indexing (search)



Reference: L. Zheng, Y. Yang, Q. Tian, “SIFT Meets CNN: A Decade Survey of Instance Retrieval,” IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 40, No. 5, May 2018, pp. 1224-1244.

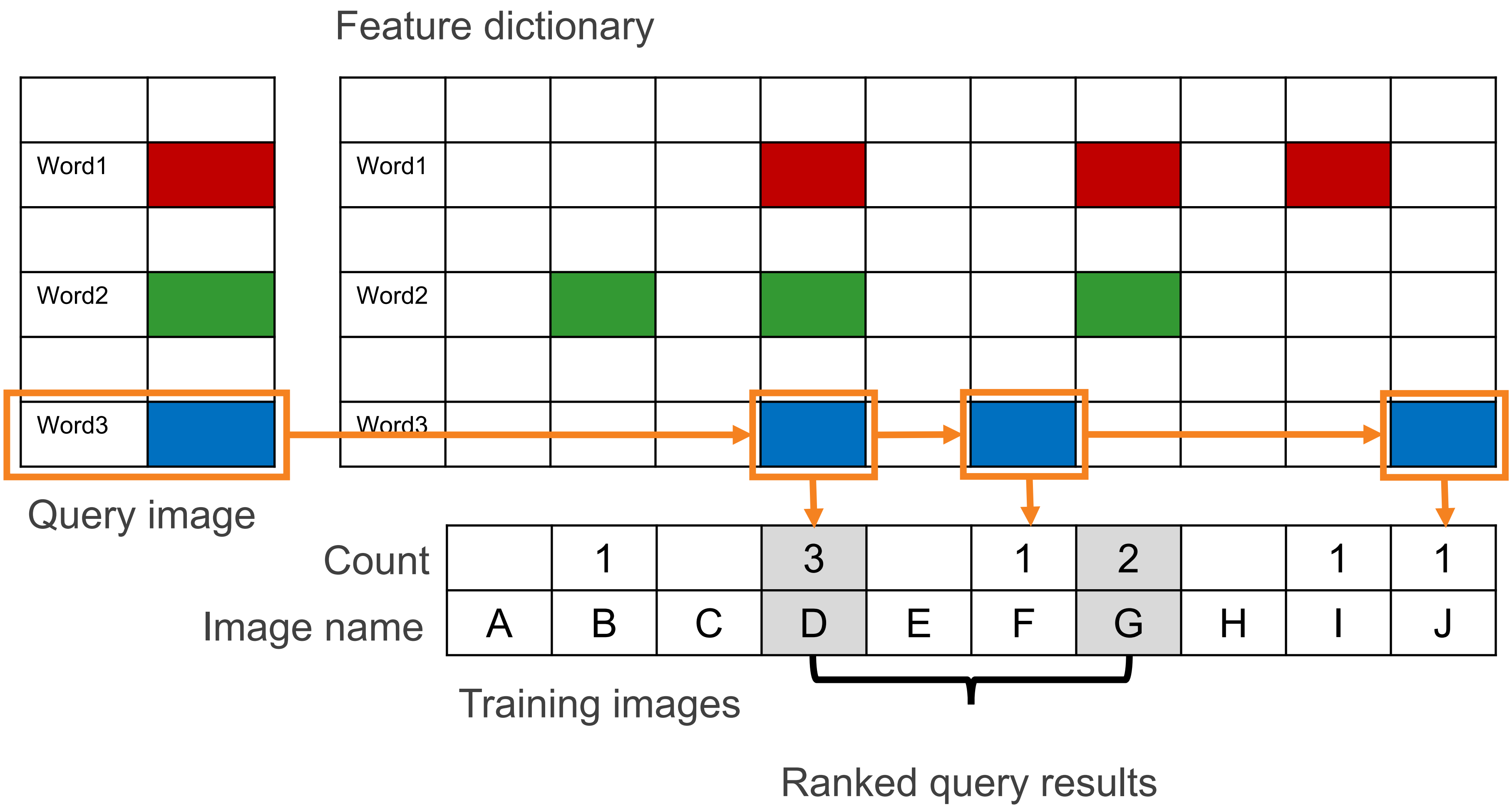
Vocabulary trees: Hierarchical clustering for large vocabularies

- Tree construction:



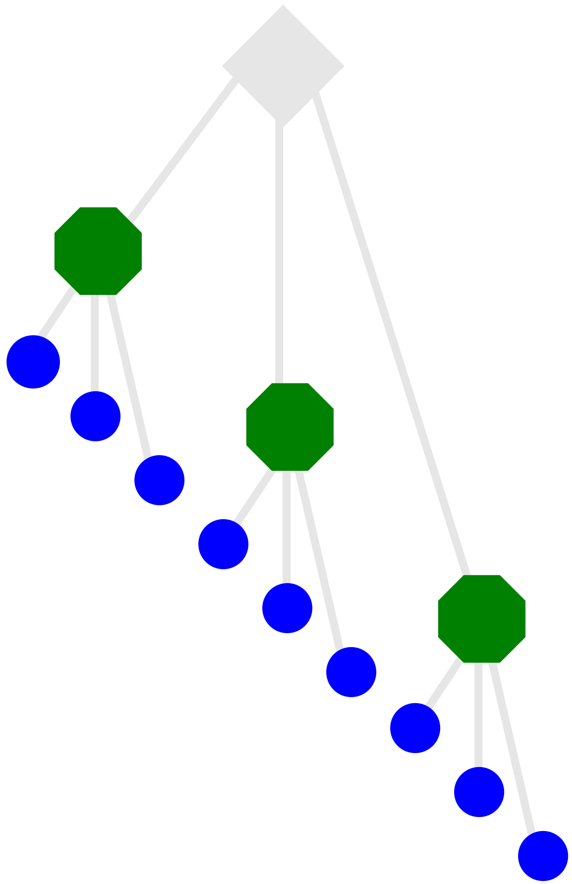
Reference: Nister and Stewenius, "Scalable Recognition with a Vocabulary Tree", CVPR 2006

Vocabulary trees: Inverted file index



Vocabulary tree

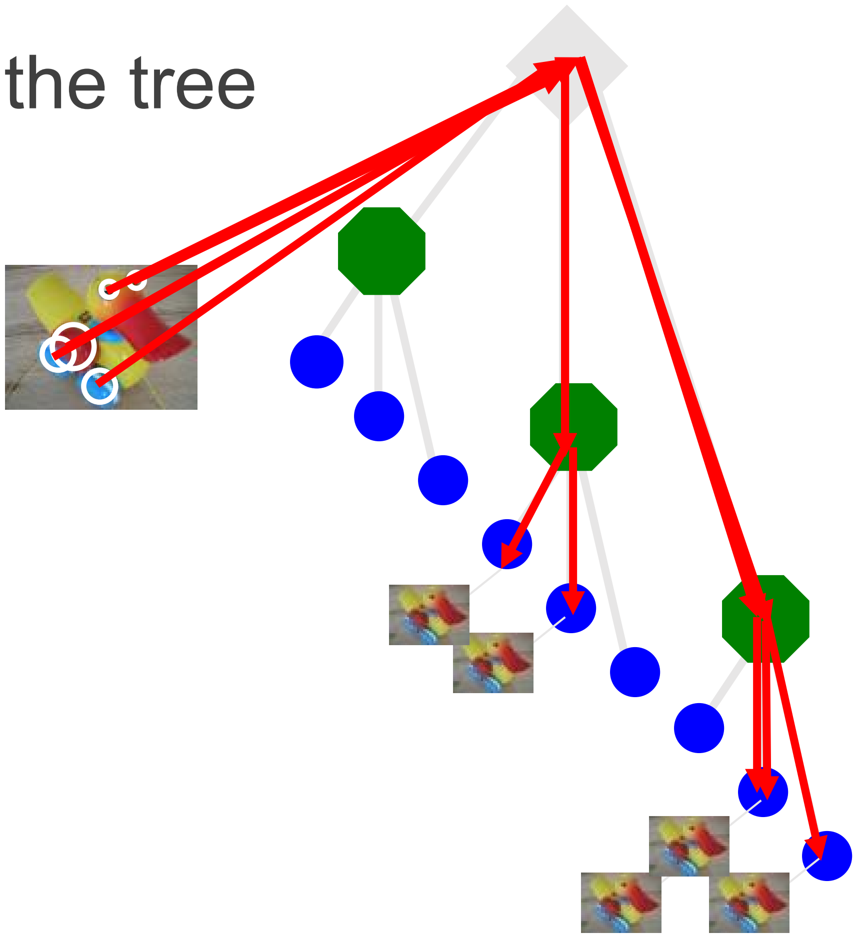
- Training: Filling the tree



Reference: Nister and Stewenius, "Scalable Recognition with a Vocabulary Tree”, CVPR 2006

Vocabulary tree

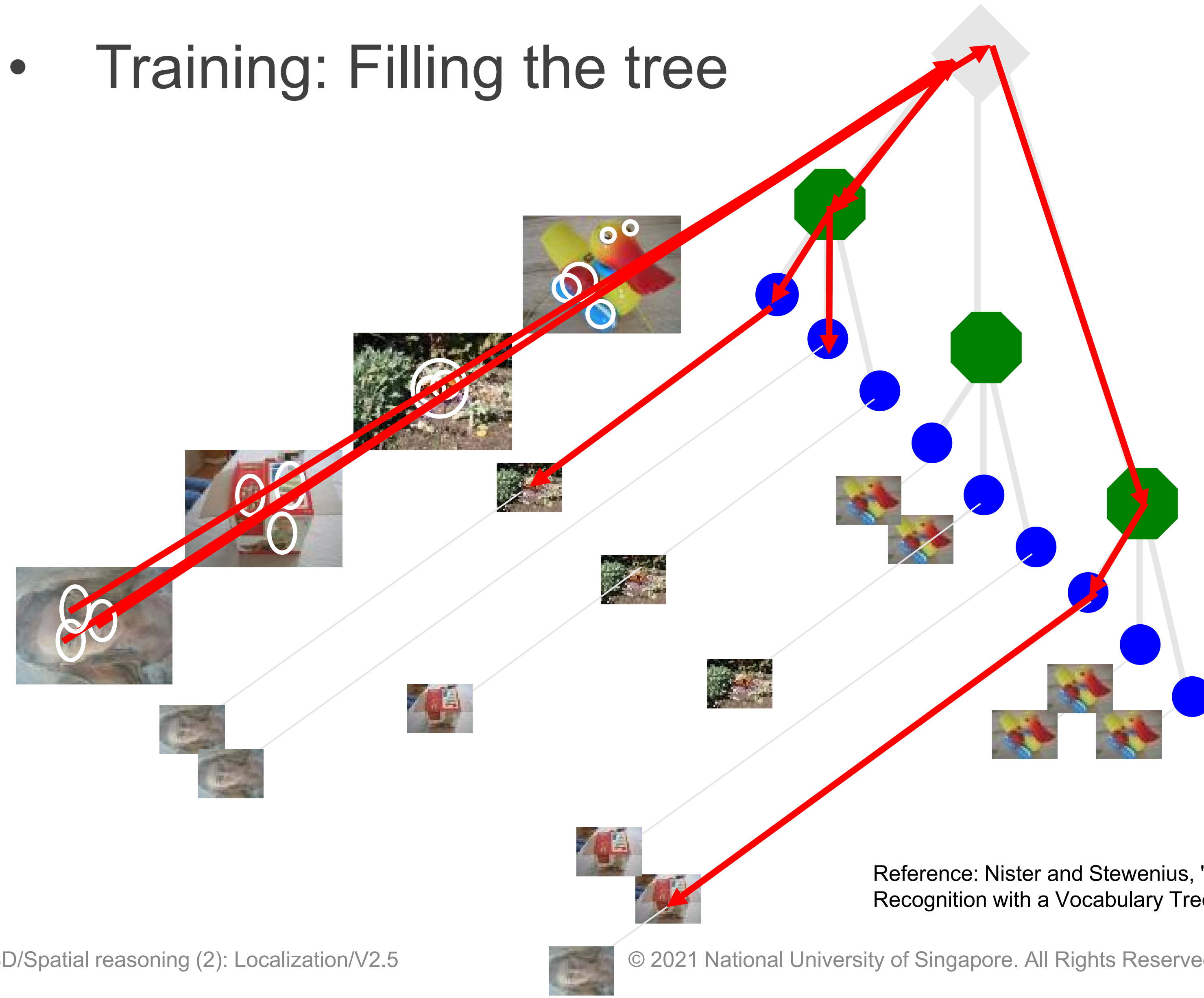
- Training: Filling the tree



Reference: Nister and Stewenius, "Scalable Recognition with a Vocabulary Tree", CVPR 2006

Vocabulary tree

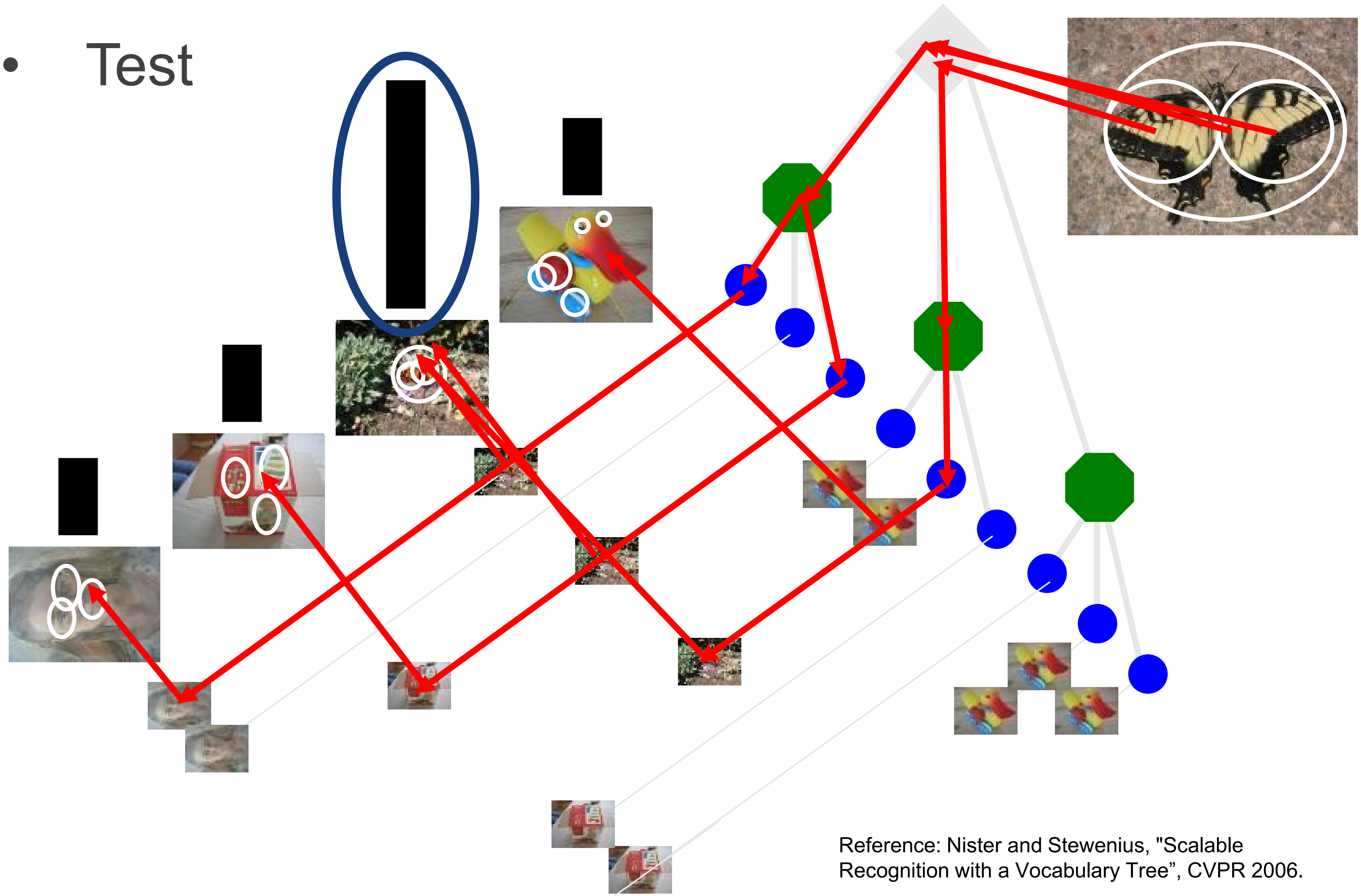
- Training: Filling the tree



Reference: Nister and Stewenius, "Scalable Recognition with a Vocabulary Tree", CVPR 2006

Vocabulary tree

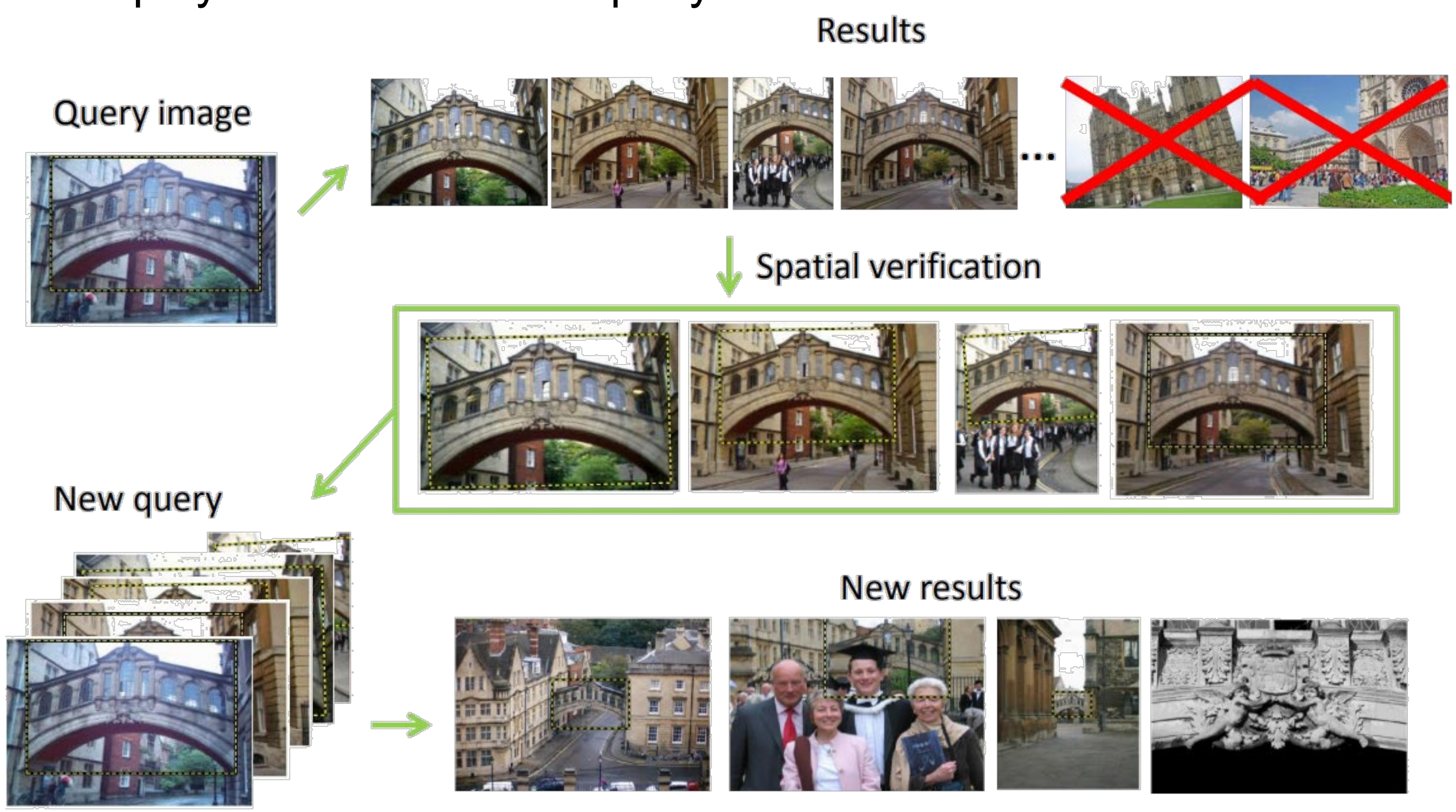
- Test



Reference: Nister and Stewenius, "Scalable Recognition with a Vocabulary Tree", CVPR 2006.

Post-processing

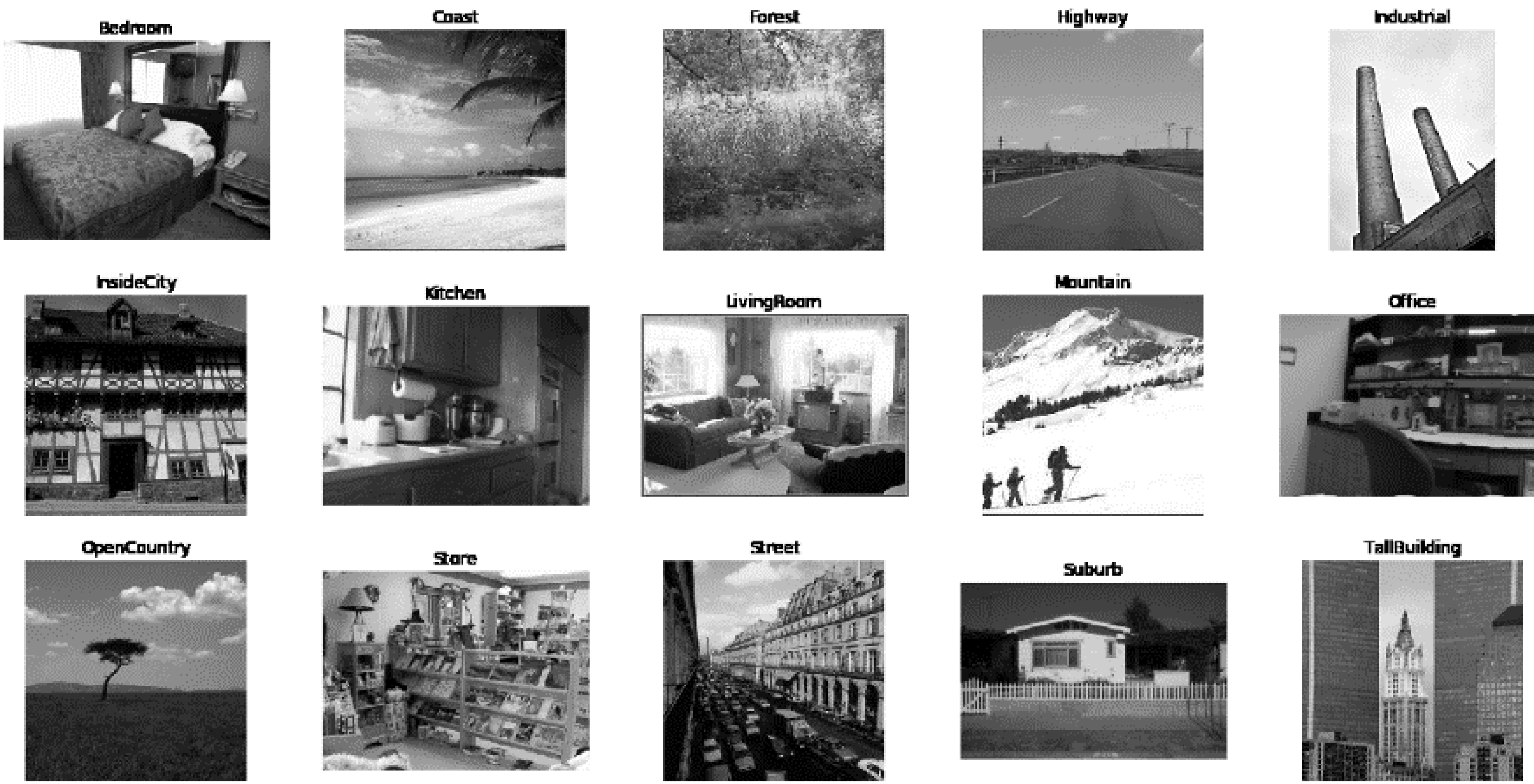
- Re-ranking: Perform spatial matching only on top-ranking images, and re-ranking according to a score based on geometry, e.g. number of inliers.
- Query expansion (QE): A number of top-ranked images from the original rank list are employed to issue a new query which is in turn used to obtain a new rank list.



Reference: O. Chum, J. Philbin, J. Sivic, M. Isard, A. Zisserman, "Total Recall: Automatic Query Expansion with a Generative Feature Model for Object Retrieval," ICCV 2007.

Workshop

- Objective: Perform image-based place recognition.
- Dataset: Scene recognition, https://www.cc.gatech.edu/~hays/compvision/proj4/



Workshop

Evaluate following methods in workshop

- VLAD: Hervé Jegou, Florent Perronnin, Matthijs Douze, Jorge Sanchez, Patrick Perez, "Aggregating local image descriptors into compact codes," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 34, No. 9, 2012, pp.1704-1716. https://hal.inria.fr/inria-00633013/document/
- Neural code: Artem Babenko, Anton Slesarev, Alexandr Chigorin, Victor Lempitsky, "Neural Codes for Image Retrieval," ECCV 2014, https://arxiv.org/abs/1404.1777
- Global sum-pooling: Artem Babenko, Victor Lempitsky, "Aggregating Deep Convolutional Features for Image Retrieval," ICCV 2015, https://arxiv.org/abs/1510.07493
- Global max-pooling: Giorgos Tolias, Ronan Sicre, Hervé Jégou, "Particular object retrieval with integral max-pooling of CNN activations," ICLR 2016, https://arxiv.org/abs/1511.05879

Rename your *.ipynb file to be your name and upload it into LumiNUS.

What we have learnt

Knowledge	• Feature extraction: Keypoint-based features, CNN-based features • Feature encoding: BoW, VLAD • Feature indexing: Inverted file search
Application	• Image-based location and place recognition • Other similar applications such as image retrieval

Thank you!

Dr TIAN Jing
Email: tianjing@nus.edu.sg