

CS4670/5670: Computer Vision

Kavita Bala

Lecture 35: Recognition Wrapup



ConvNets breakthroughs for visual tasks

	Dataset	Performance	Score
<p>[Sermanet et al 2014]: OverFeat (fine-tuned features for each task) (tasks are ordered by increasing difficulty)</p>			
<ul style="list-style-type: none"> image classification 	ImageNet LSVRC 2013	competitive	13.6 % error
<ul style="list-style-type: none"> object localization 	Dogs vs Cats Kaggle challenge 2014	state of the art	98.9%
<ul style="list-style-type: none"> object detection 	ImageNet LSVRC 2013	state of the art	29.9% error
	ImageNet LSVRC 2013	competitive	24.3% mAP
<p>[Razavian et al, 2014]: public OverFeat library (no retraining) + SVM (simplest approach possible on purpose, no attempt at more complex classifiers) (tasks are ordered by “distance” from classification task on which OverFeat was trained)</p>			
<ul style="list-style-type: none"> image classification 	Pascal VOC 2007	competitive	77.2% mAP
<ul style="list-style-type: none"> scene recognition 	MIT-67	state of the art	69% mAP
<ul style="list-style-type: none"> fine grained recognition 	Caltech-UCSD Birds 200-2011	competitive	61.8% mAP
	Oxford 102 Flowers	state of the art	86.8% mAP
<ul style="list-style-type: none"> attribute detection 	UIUC 64 object attributes	state of the art	91.4% mAUC
	H3D Human Attributes	competitive	73% mAP
<ul style="list-style-type: none"> image retrieval (search by image similarity) 	Oxford 5k buildings	state of the art	68% mAP?
	Paris 6k buildings	state of the art	79.5% mAP?
	Sculp6k	competitive	42.3% mAP?
	Holidays	state of the art	84.3% mAP?
	UKBench	state of the art	91.1% mAP?

Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, Yann LeCun, **OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks**, <http://arxiv.org/abs/1312.6229>, ICLR 2014

Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, Stefan Carlsson, **CNN Features off-the-shelf: an Astounding Baseline for Recognition**, <http://arxiv.org/abs/1403.6382>, DeepVision CVPR 2014 workshop

ConvNets breakthroughs for visual tasks

	Dataset	Performance	Score
[Zeiler et al 2013] <ul style="list-style-type: none"> image classification 	ImageNet LSVRC 2013 Caltech-101 (15, 30 samples per class) Caltech-256 (15, 60 samples per class) Pascal VOC 2012	state of the art competitive state of the art competitive	11.2% error 83.8%, 86.5% 65.7%, 74.2% 79% mAP
[Donahue et al, 2014]: DeCAF+SVM <ul style="list-style-type: none"> image classification domain adaptation fine grained recognition scene recognition 	Caltech-101 (30 classes) Amazon -> Webcam, DSLR -> Webcam Caltech-UCSD Birds 200-2011 SUN-397	state of the art state of the art state of the art competitive	86.91% 82.1%, 94.8% 65.0% 40.9%
[Girshick et al, 2013] <ul style="list-style-type: none"> image detection image segmentation 	Pascal VOC 2007 Pascal VOC 2010 (comp4) ImageNet LSVRC 2013 Pascal VOC 2011 (comp6)	state of the art state of the art state of the art state of the art	48.0% mAP 43.5% mAP 31.4% mAP 47.9% mAP
[Oquab et al, 2013] <ul style="list-style-type: none"> image classification 	Pascal VOC 2007 Pascal VOC 2012 Pascal VOC 2012 (action classification)	state of the art state of the art state of the art	77.7% mAP 82.8% mAP 70.2% mAP

M.D. Zeiler, R. Fergus, **Visualizing and Understanding Convolutional Networks**, Arxiv 1311.2901 <http://arxiv.org/abs/1311.2901>

J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. **Decaf: A deep convolutional activation feature for generic visual recognition**. In ICML, 2014, <http://arxiv.org/abs/1310.1531>

R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. **Rich feature hierarchies for accurate object detection and semantic segmentation**. arxiv:1311.2524 [cs.CV], 2013, <http://arxiv.org/abs/1311.2524>

M. Oquab, L. Bottou, I. Laptev, and J. Sivic. **Learning and transferring mid-level image representations using convolutional neural networks**. Technical Report HAL-00911179, INRIA, 2013. <http://hal.inria.fr/hal-00911179>

ConvNets breakthroughs for visual tasks

	Dataset	Performance	Score
[Khan et al 2014] <ul style="list-style-type: none">shadow detection	UCF CMU UIUC	state of the art state of the art state of the art	90.56% 88.79% 93.16%
[Sander Dieleman, 2014] <ul style="list-style-type: none">image attributes	Kaggle Galaxy Zoo challenge	state of the art	0.07492

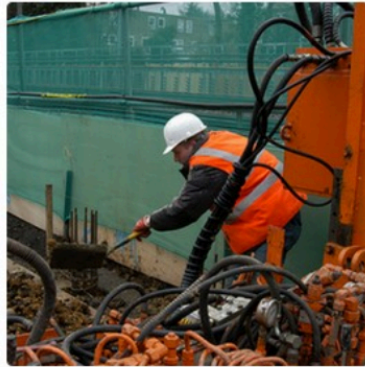
S. H. Khan, M. Bennamoun, F. Sohel, R. Togneri. **Automatic Feature Learning for Robust Shadow Detection**, CVPR 2014

Sander Dieleman, Kaggle Galaxy Zoo challenge 2014 <http://benanne.github.io/2014/04/05/galaxy-zoo.html>

Image Captioning



"man in black shirt is playing guitar."



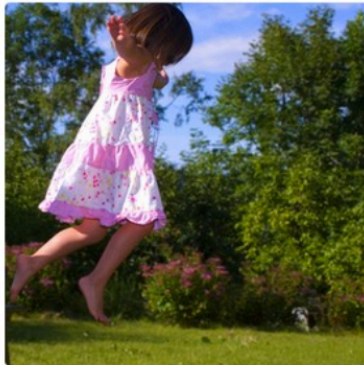
"construction worker in orange safety vest is working on road."



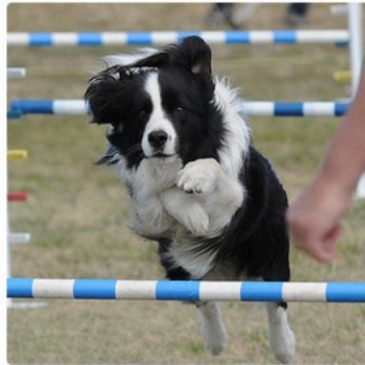
"two young girls are playing with lego toy."



"boy is doing backflip on wakeboard."



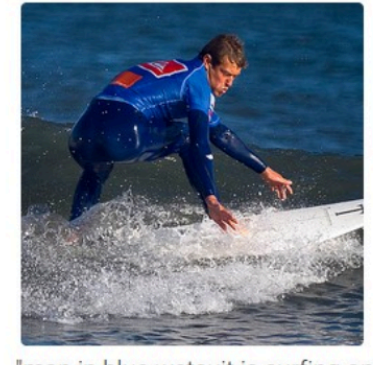
"girl in pink dress is jumping in air."



"black and white dog jumps over bar."



"young girl in pink shirt is climbing on swina."



"man in blue wetsuit is surfing on wave."

CNNs + CRFs

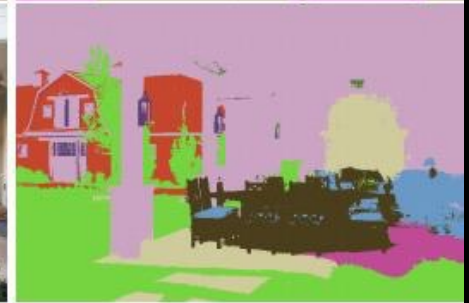
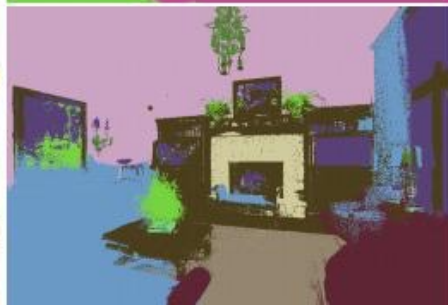
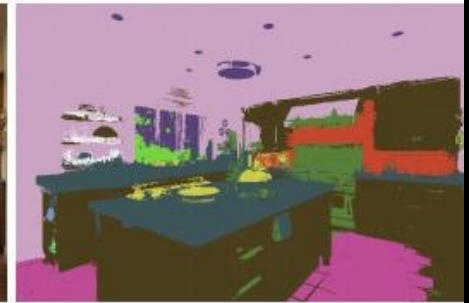
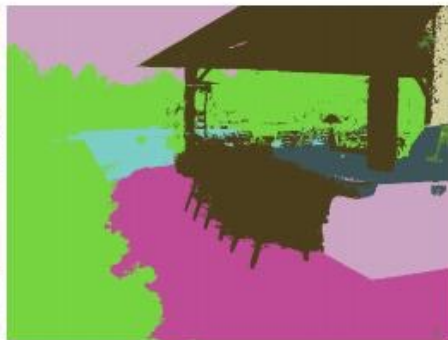


CRF Runtime: ~1s for 640x480 image

$$E(\mathbf{x}|\mathbf{I}, \boldsymbol{\theta}) = \sum_i \psi_i(x_i|\boldsymbol{\theta}) + \sum_{i<j} \psi_{ij}(x_i, x_j|\boldsymbol{\theta})$$

Material Segmentation[CVPR15]

brick	food	painted	tile
carpet	glass	paper	stone
ceramic	hair	plastic	water
fabric	leather	polishedstone	wood
foliage	metal	skin	



Bell, Upchurch, Snavely, Bala

“It can be concluded that from now on, deep learning with CNN has to be considered as the primary candidate in essentially any visual recognition task.”

[Razavian 2014]

CNNs at Google (as of 2014)



Applications

Google Image Search

The screenshot shows the Google Image Search interface. The search bar contains the word "cats". Below the search bar, there are tabs for "Web", "Images", "Videos", "News", "Shopping", and "More". The "Images" tab is selected. The search results are displayed in a grid of images. The first row shows three categories: "Cute", "Lots Of", and "Kittens". The "Cute" category shows a ginger kitten lying down. The "Lots Of" category shows a collage of various cats. The "Kittens" category shows a group of kittens. The second row shows four images: a ginger kitten lying on its back, a grey tabby kitten looking forward, a group of four kittens sitting together, and a ginger kitten looking up. The third row shows three images: a grey and white kitten, a grey tabby kitten with a speech bubble that says "A New York strip, it had a cherry hand of fat at its edge, as if the meat were wearing a protective latex sheath, and the meat was shot through with gristle.", and a ginger and white kitten with its mouth open.

Search by Image



Image size:
450 × 338

Find other sizes of this image:
[All sizes](#) - [Medium](#)

Best guess for this image: ***cats and kittens***

Funny Cats and Kittens Meowing Compilation 2013 - YouTube

www.youtube.com/watch?v=DXUAYRRkI6k

Nov 9, 2013 - [Cats Meowing](#) | [Cat Meowing](#) | [Kittens Meowing](#) | [Kitten Meowing](#) | [Meowing Cat](#) | [Funny Cats](#) | [Meowing Kittens](#) | [Cat Meowing Non Stop](#) | [Cats ...](#)

mama cat comes to rescue her little kitten - YouTube

www.youtube.com/watch?v=S5-D0f6nHSQ

Jul 7, 2007 - Pets animals [cats kittens](#) fun. Subscribe! <http://www.youtube.com/user/Epikneverdies>.

Visually similar images

[Report images](#)



CNNs at Google (as of 2014)




Applications - Photo Search

my photos of coffee

Web Images News Shopping Videos More Search tools

About 386,000,000 results (0.37 seconds)

Your photos
Only you can see these results



[View all Google+ results](#)


[Learn more - Give feedback](#)

my photos of cake

Web Images Shopping Videos News More Search tools

About 204,000,000 results (0.42 seconds)

Your photos
Only you can see these results



[View all Google+ results](#)


[Learn more - Give feedback](#)

my photos of waterfalls

Web Images Shopping Videos News More Search tools

About 27,000,000 results (0.39 seconds)

Your photos
Only you can see these results



[View all Google+ results](#)

[Learn more - Give feedback](#)

CNNs at Google (as of 2014)



Google Photos - Auto Awesome



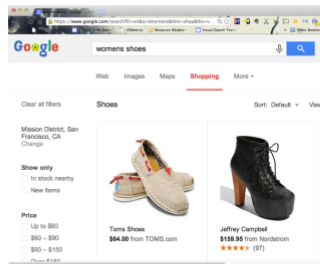
CNNs at Google (as of 2014)



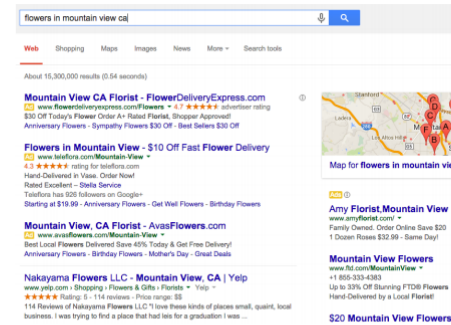
More Image Understanding at Google



YouTube



Google Shopping



Advertising

Much more...



StreetView / Maps



Self-Driving Cars

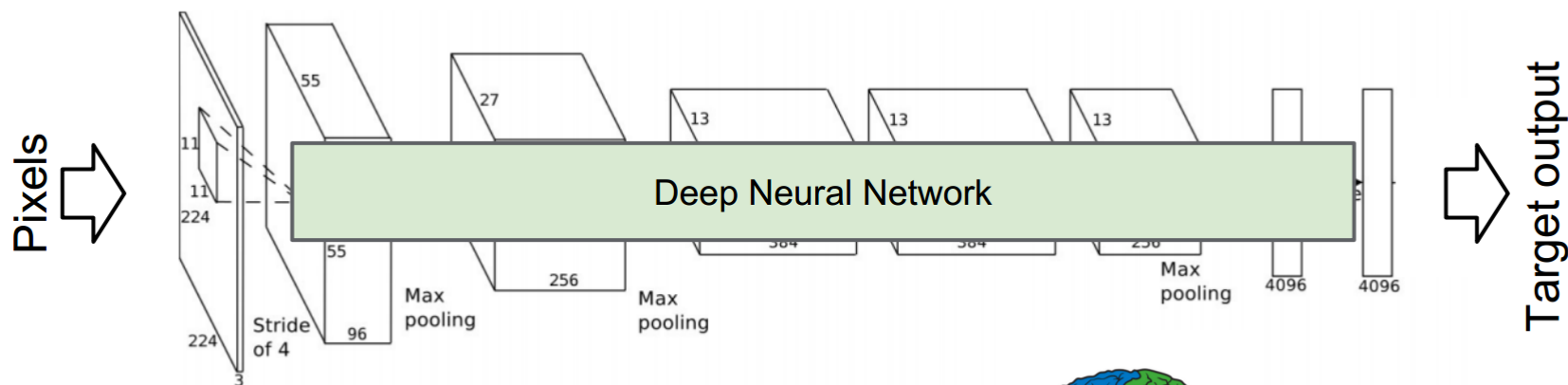


Robotics

CNNs at Google (as of 2014)



The Deep and now Deeper Hammer



Deep learning infrastructure by the
Google Brain team



“ImageNet Classification with Deep Convolutional Neural Networks”,
Krizhevsky, Sutskever, Hinton, NIPS 2012

CNNs at Google (as of 2014)



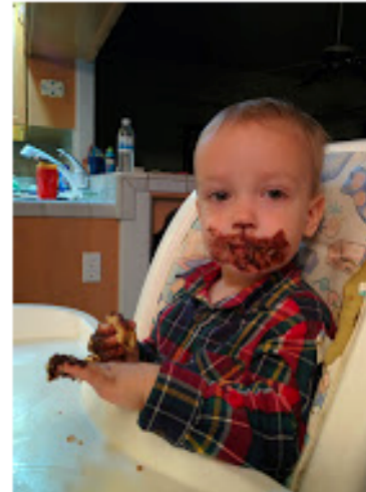
Personal Photos - Example Annotations



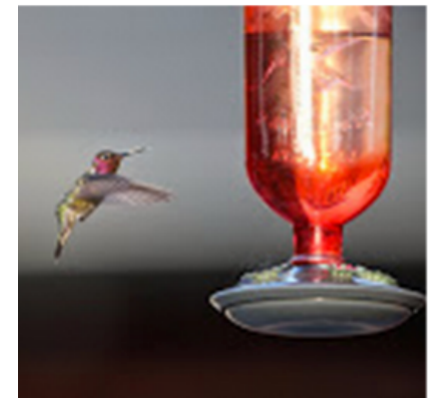
Christmas tree
Red
Christmas decoration
Christmas



Crowd
Cheering
People
Stadium

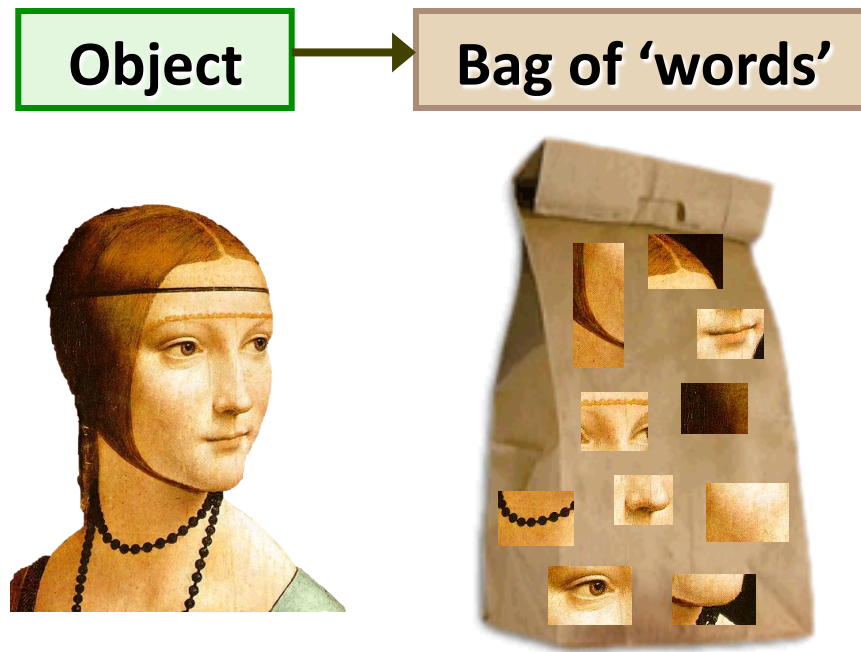


Play
Meal
Cake
Child



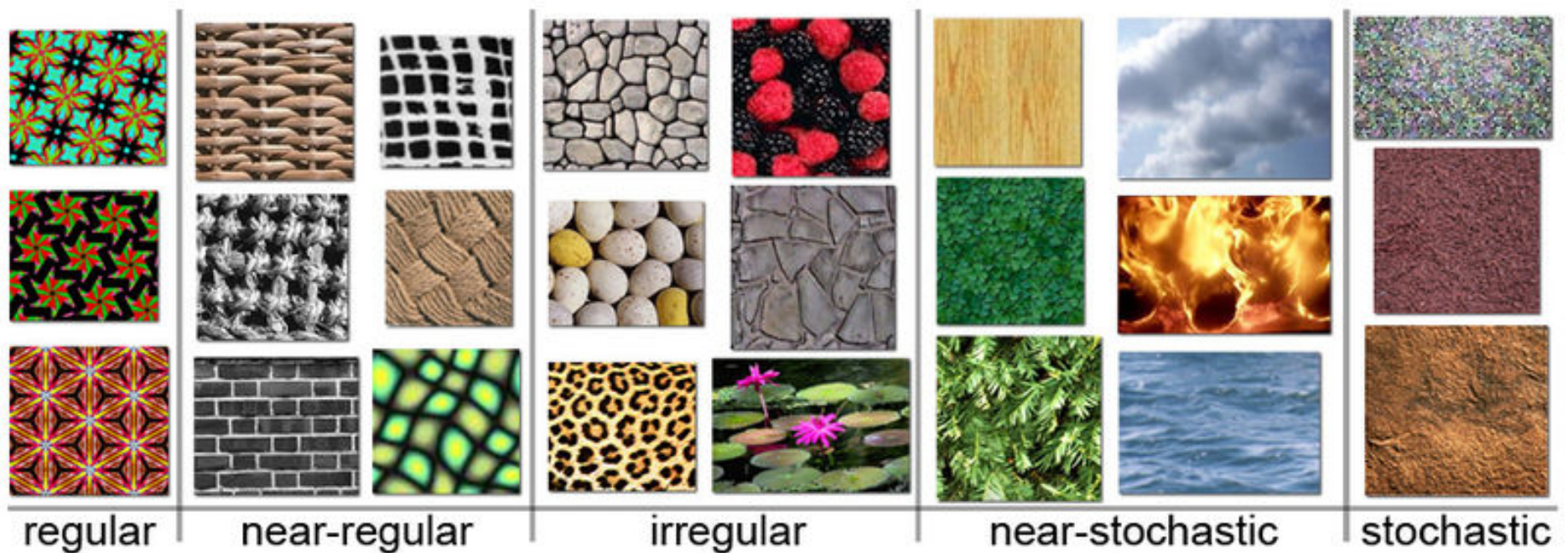
Hummingbird
Macro photography
Reflection
Red

Before CNNs: Bag of words



Adapted from slides by Rob Fergus and
Svetlana Lazebnik

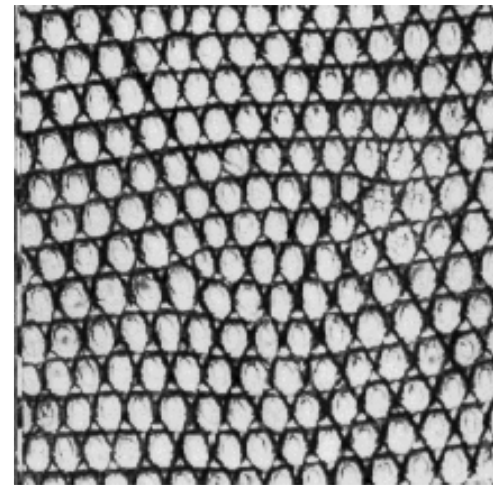
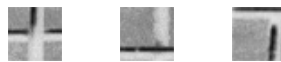
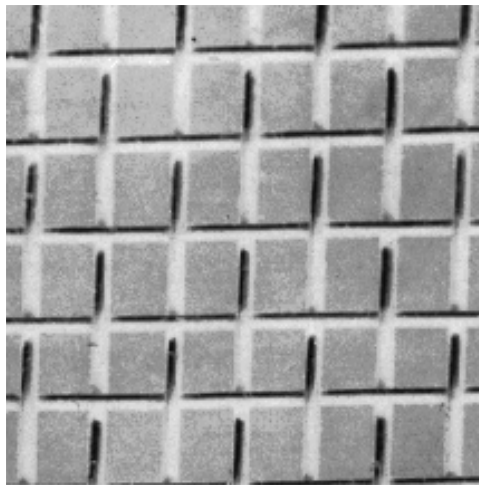
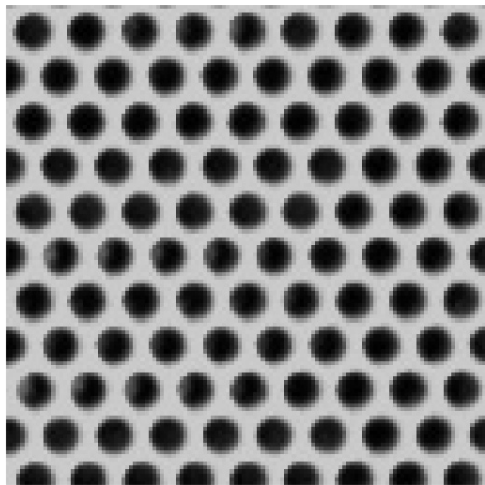
Origin 1: Texture Recognition



Example textures (from Wikipedia)

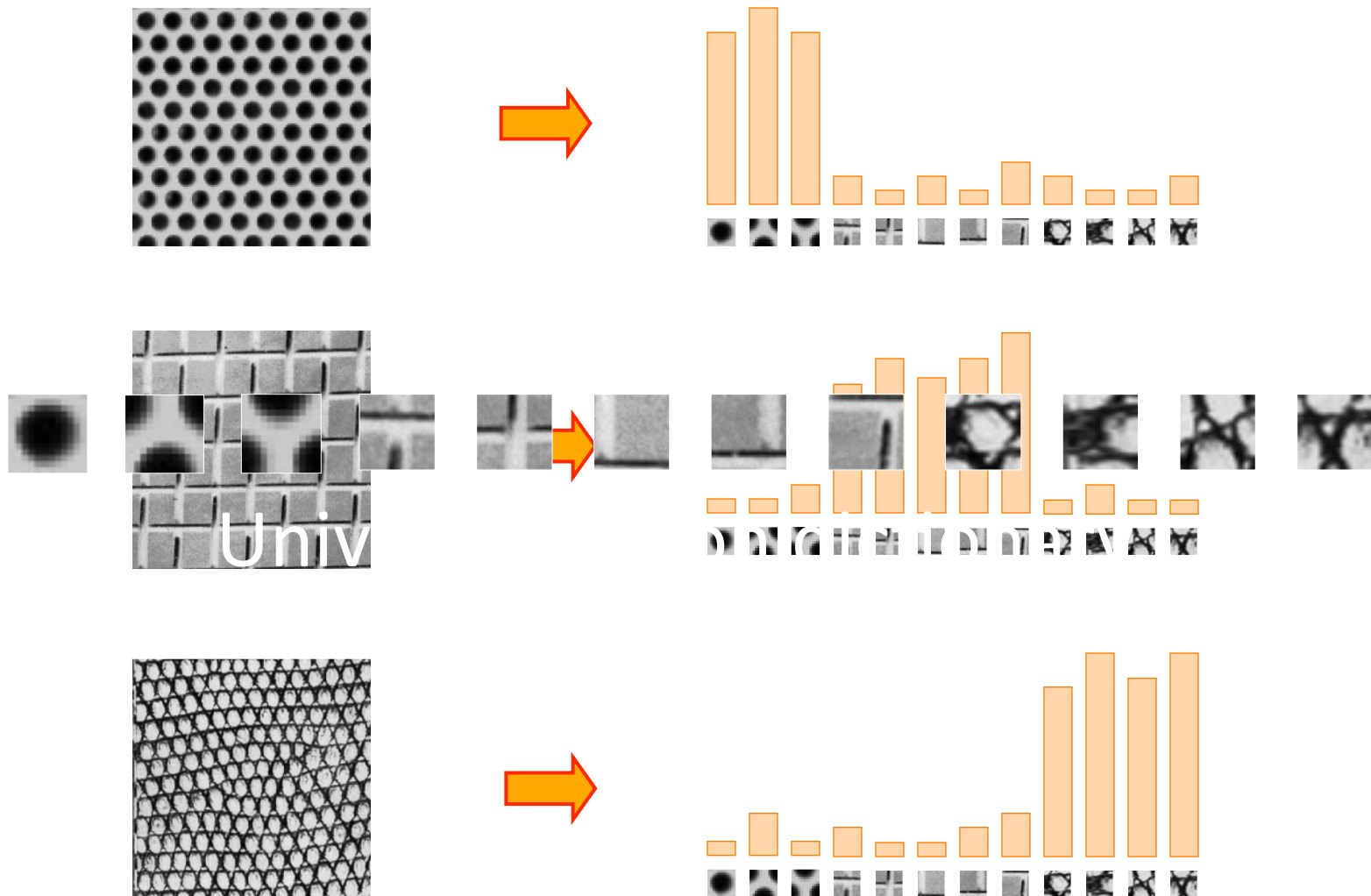
Origin 1: Texture recognition

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



Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Origin 1: Texture recognition



Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

2007-01-23: State of the Union Address George W. Bush (2001-)

abandon accountable affordable afghanistan africa aided ally anbar armed army **baghdad** bless **challenges** chamber chaos
choices civilians coalition commanders **commitment** confident confront congressman constitution corps debates deduction
deficit deliver **democratic** deploy dikembe diplomacy disruptions earmarks **economy** einstein **elections** eliminates
expand **extremists** failing faithful families **freedom** fuel funding god haven ideology immigration impose

insurgents iran **iraq** islam julie lebanon love madam marine math medicare moderation neighborhoods nuclear offensive
palestinian payroll province pursuing **qaeda** radical regimes resolve retreat rieman sacrifices science sectarian senate

september **shia** stays strength students succeed sunni **tax** territories **terrorists** threats uphold victory
violence violent **war** washington weapons wesley

Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

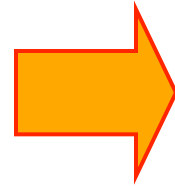


Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)



Bags of features for object recognition



face, flowers, building

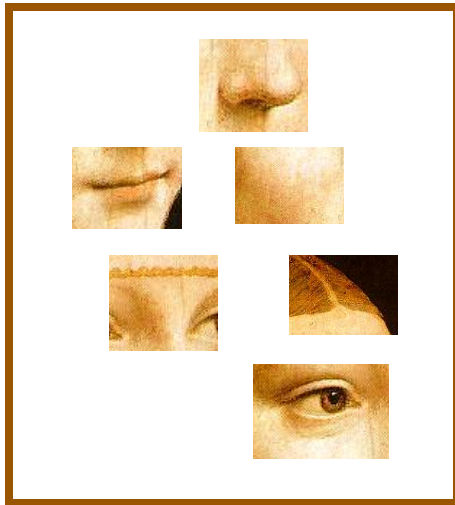
- Works pretty well for image-level classification and for recognizing object *instances*

Bag of features

- First, take a bunch of images, extract features, and build up a “dictionary” or “visual vocabulary” – a list of common features
- Given a new image, extract features and build a histogram – for each feature, find the closest visual word in the dictionary

Bag of features: outline

1. Extract features



Bag of features: outline

1. Extract features
2. Learn “visual vocabulary”

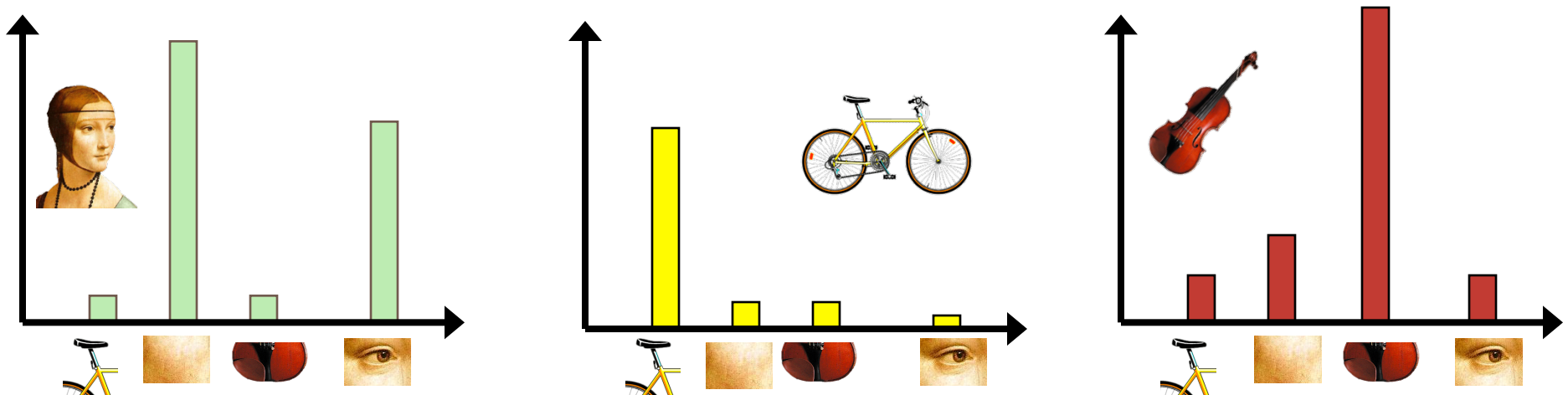


Bag of features: outline

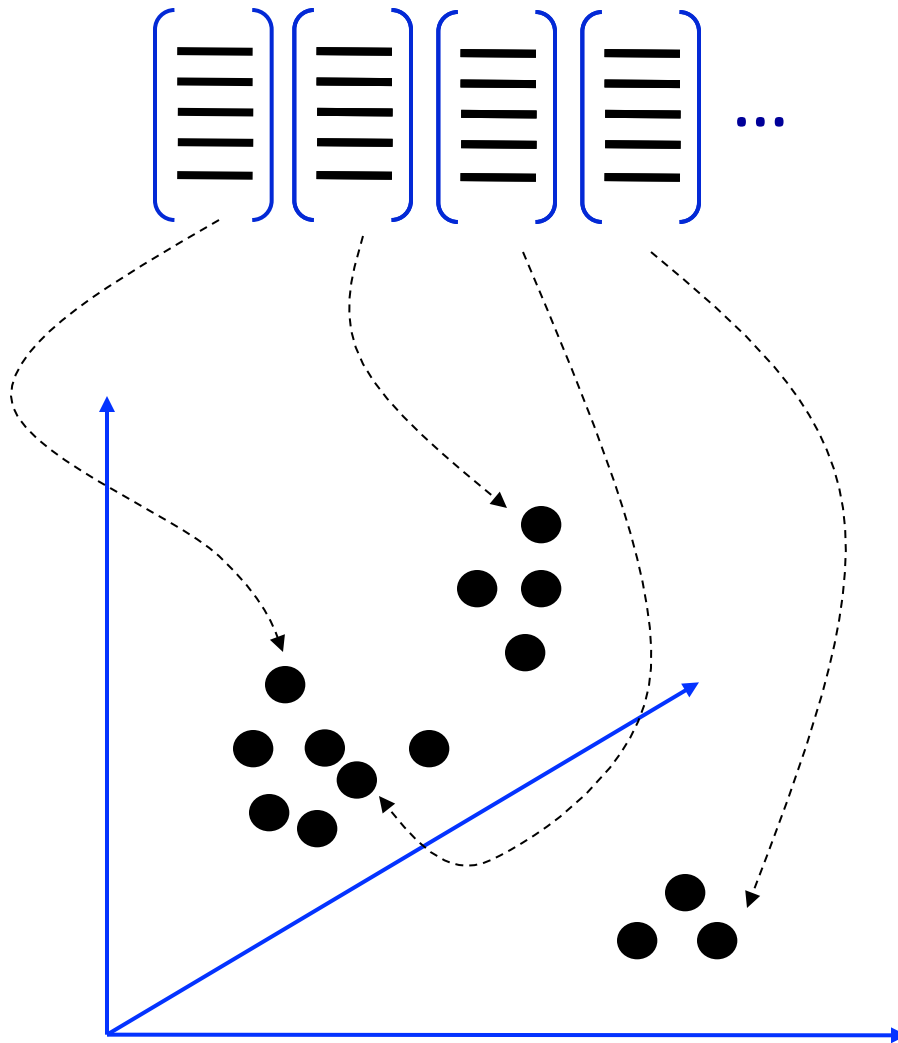
1. Extract features
2. Learn “visual vocabulary”
3. Quantize features using visual vocabulary

Bag of features: outline

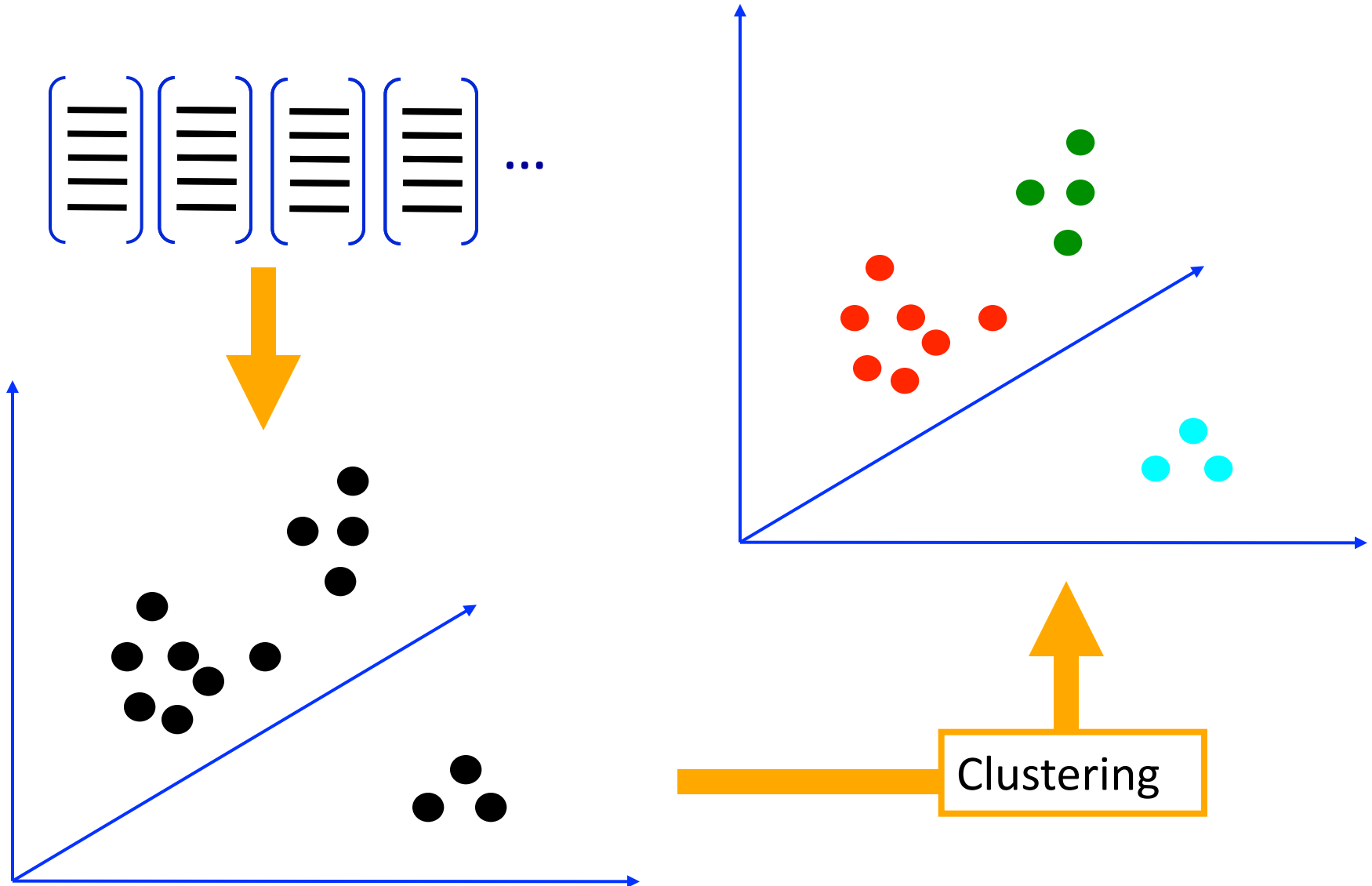
1. Extract features
2. Learn “visual vocabulary”
3. Quantize features using visual vocabulary
4. Represent images by frequencies of “visual words”



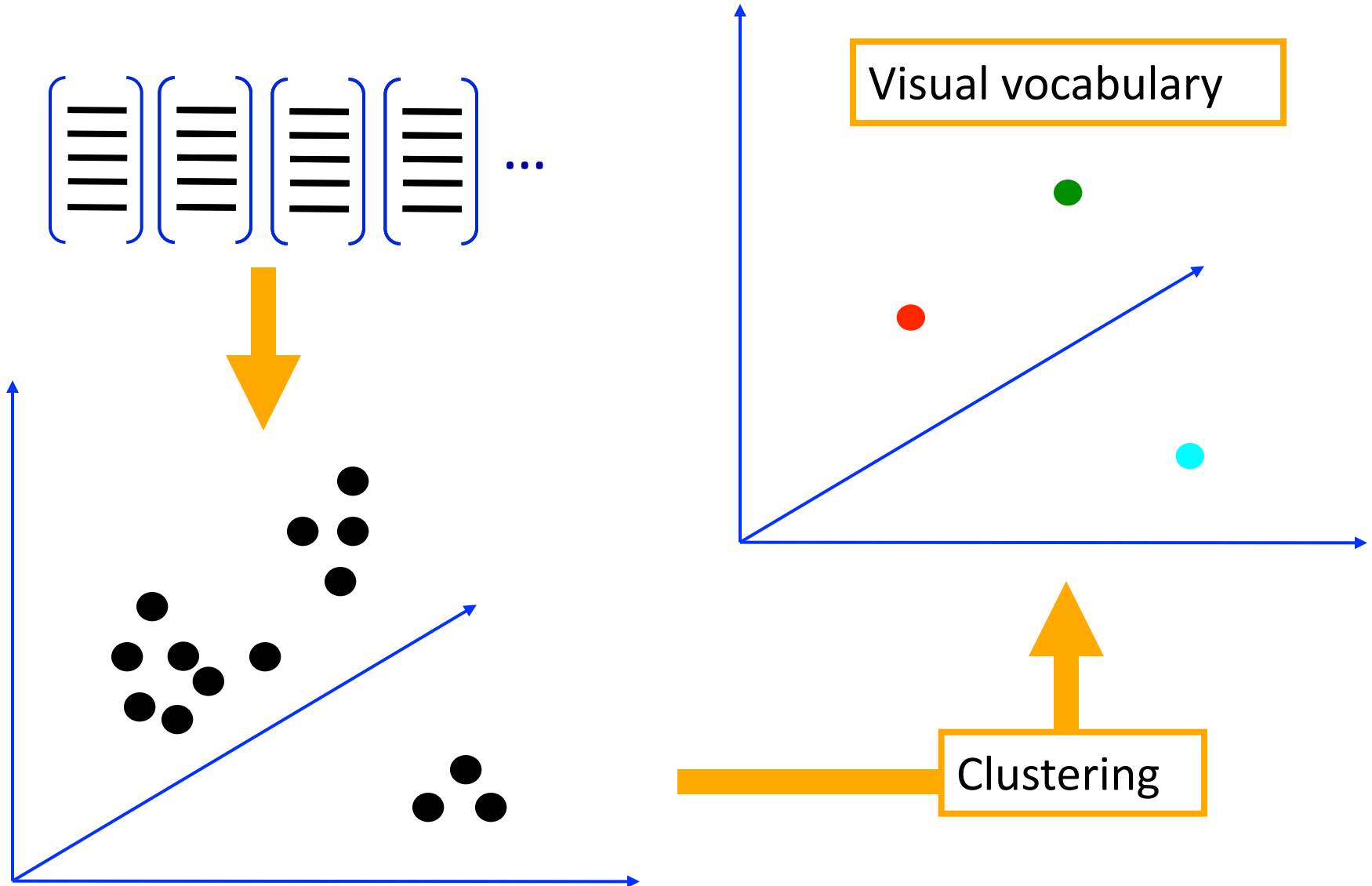
2. Learning the visual vocabulary



2. Learning the visual vocabulary



2. Learning the visual vocabulary



K-means clustering

- Want to minimize sum of squared Euclidean distances between points x_i and their nearest cluster centers m_k

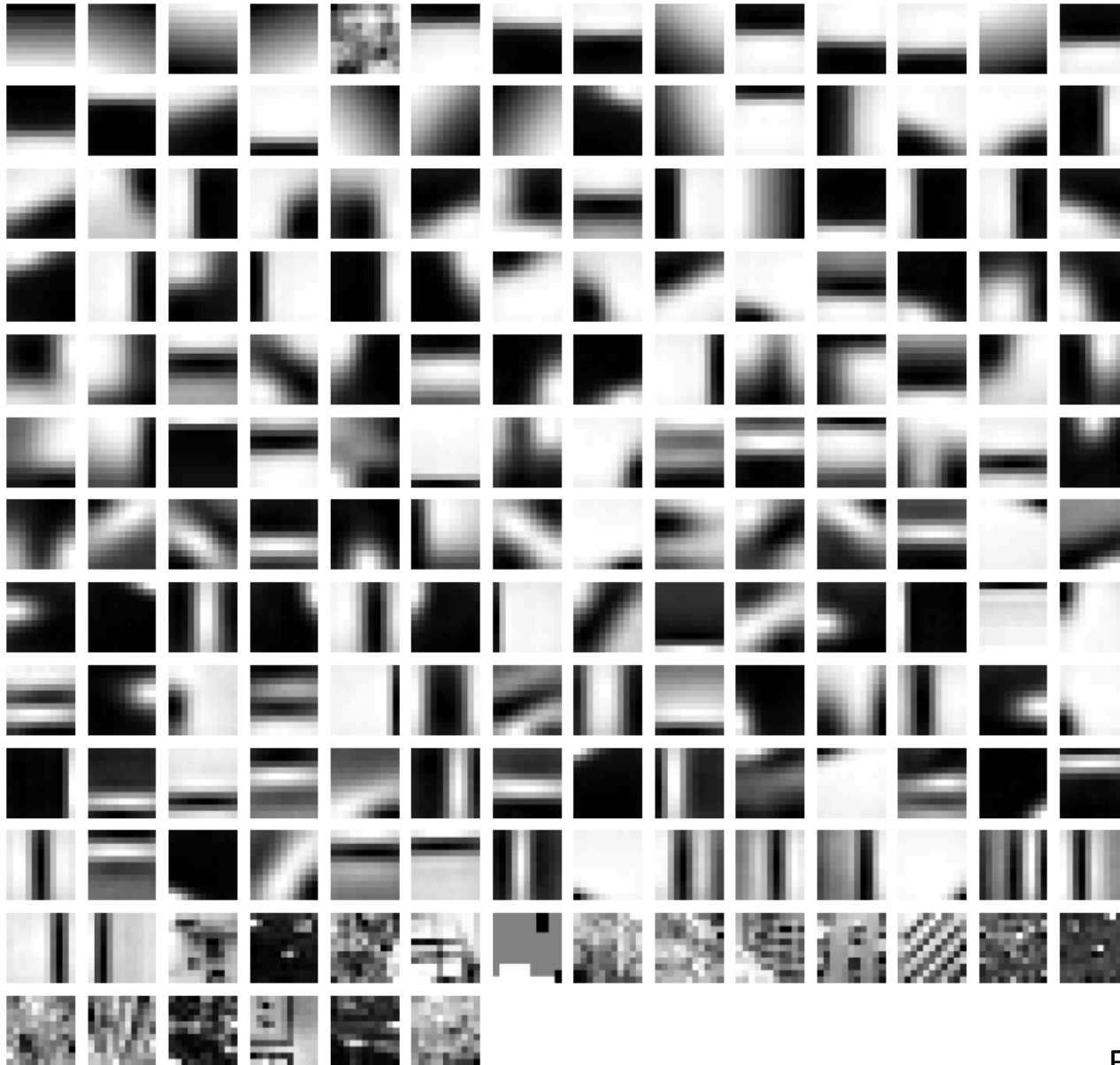
$$D(X, M) = \sum_{\text{cluster } k} \sum_{\text{point } i \text{ in cluster } k} (x_i - m_k)^2$$

- Algorithm:
- Randomly initialize K cluster centers
- Iterate until convergence:
 - Assign each data point to the nearest center
 - Recompute each cluster center as the mean of all points assigned to it

From clustering to vector quantization

- 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
 - Provided the training set is sufficiently representative, the codebook will be “universal”
- 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

Example visual vocabulary



3. Image representation

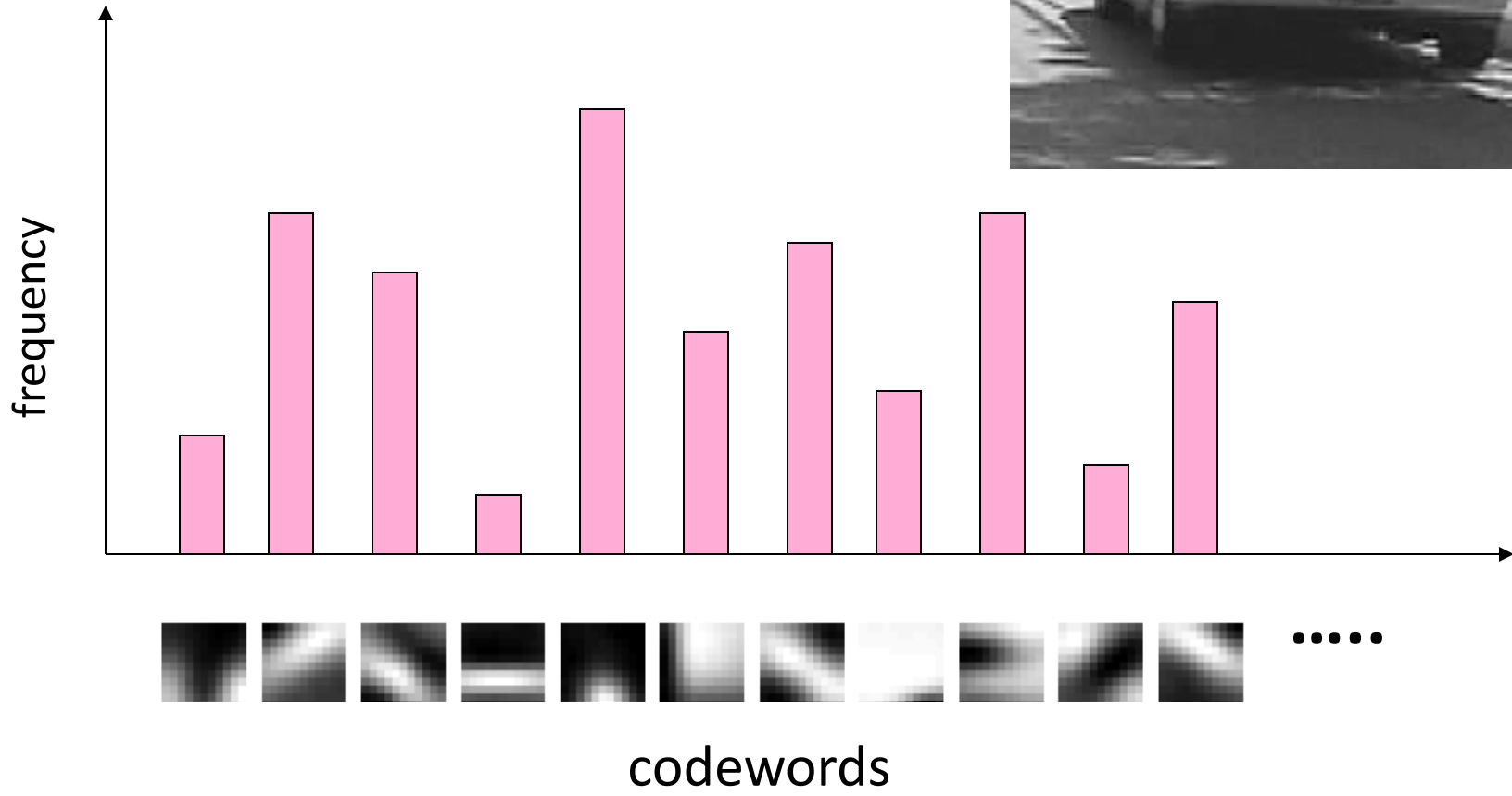
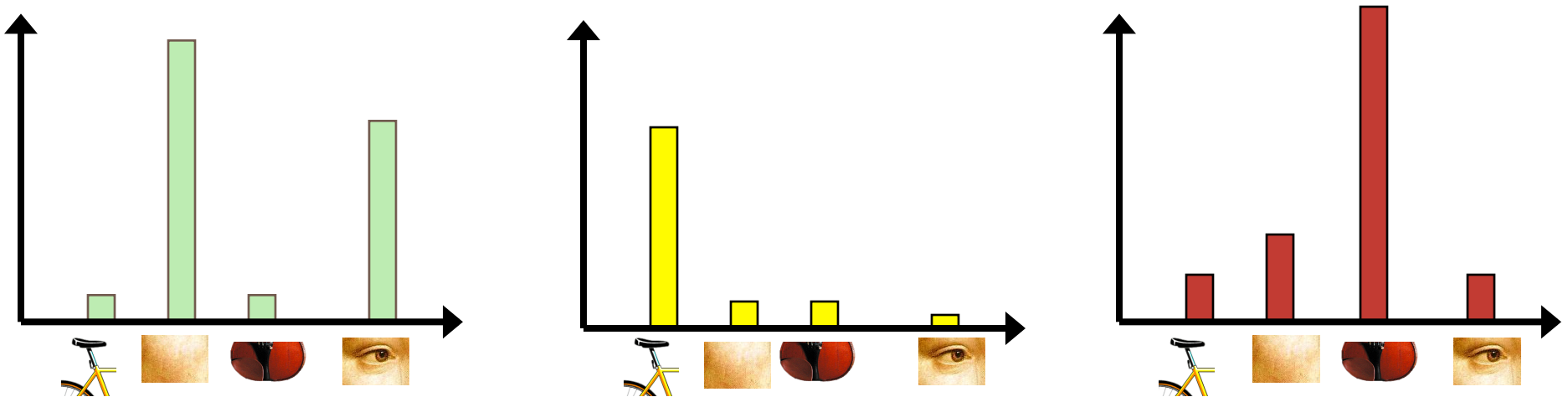


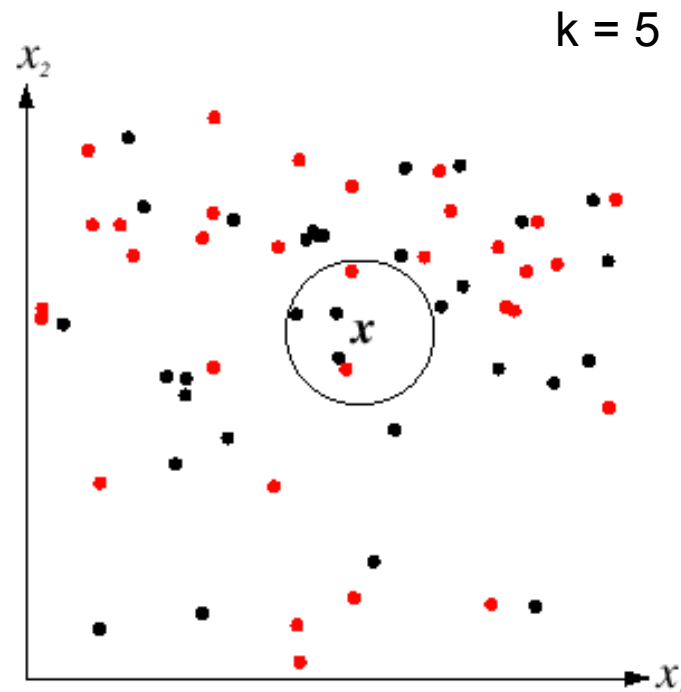
Image classification

- Given the bag-of-features representations of images from different classes, classify image.



K nearest neighbors

- For a new point, find the k closest points from training data
- Labels of the k points “vote” to classify
- Works well provided there is lots of data and the distance function is good



Uses of BoW representation

- Treat as feature vector for standard classifier
 - e.g k-nearest neighbors, support vector machine
- Cluster BoW vectors over image collection
 - Discover visual themes

Large-scale image matching



11,400 images of game covers
(Caltech games dataset)

- Bag-of-words models have been useful in matching an image to a large database of object *instances*



how do I find this image in the database?

Large-scale image search



- Build the database:
 - Extract features from the database images
 - Learn a vocabulary using k-means (typical k: 100,000)
 - Compute *weights* for each word
 - Create an inverted file mapping words → images

Weighting the words

- Just as with text, some visual words are more discriminative than others

the, and, or vs. ***cow, AT&T, Cher***

- the bigger fraction of the documents a word appears in, the less useful it is for matching
 - e.g., a word that appears in *all* documents is not helping us

TF-IDF weighting

- Instead of computing a regular histogram distance, we'll weight each word by its *inverse document frequency*
- inverse document frequency (IDF) of word j =

$$\log \frac{\text{number of documents}}{\text{number of documents in which } j \text{ appears}}$$

TF-IDF weighting

- To compute the value of bin j in image l :

term frequency of j in l **X** *inverse document frequency of j*

Inverted file

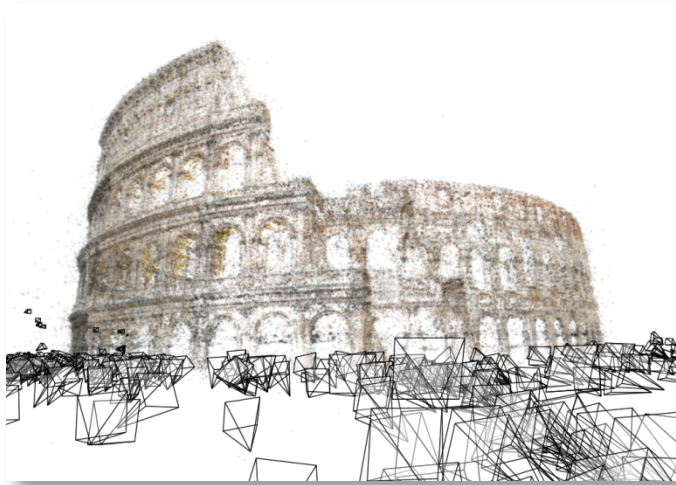
- Each image has ~1,000 features
- We have ~100,000 visual words
 - each histogram is extremely sparse (mostly zeros)
- Inverted file
 - mapping from words to documents

```
"a": {2}
"banana": {2}
"is": {0, 1, 2}
"it": {0, 1, 2}
"what": {0, 1}
```

Inverted file

- Can quickly use the inverted file to compute similarity between a new image and all the images in the database
 - Only consider database images whose bins overlap the query image

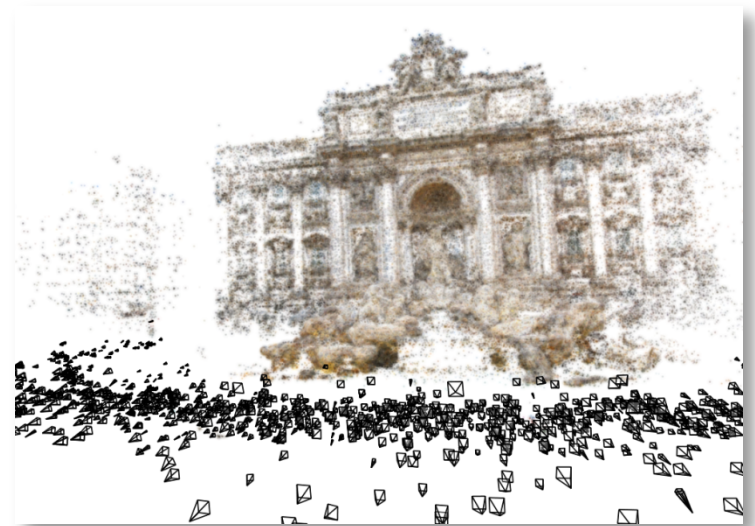
...into 3D models



Colosseum



St. Peter's Basilica



Trevi Fountain

Large-scale image matching

- How can we match 1,000,000 images to each other?
- Brute force approach: 500,000,000,000 pairs
 - won't scale
- Better approach: use bag-of-words technique to find *likely* matches
- For each image, find the top M scoring other images, do detailed SIFT matching with those

Example bag-of-words matches



Example bag-of-words matches



Matching Statistics

Dataset	Size	Matches possible	Matches Tried	Matches Found	Time
Dubrovnik	58K	1.6 Billion	2.6M	0.5M	5 hrs
Rome	150K	11.2 Billion	8.8M	2.7M	13 hrs
Venice	250K	31.2 Billion	35.5M	6.2M	27 hrs

Quiz 4