CS688: Web-Scale Image Search

Convolutional Neural Networks

Sung-Eui Yoon
(윤성의)

Course URL:
http://sgvr.kaist.ac.kr/~sungeui/IR

KAIST

Schedule

- Apr-28 (Tue): mid-term exam (can be changed)
- May 12, 14: Students Presentation I (2 talks per each class)
- May 19,
- May 21 (reserved)
- May 26,28: Mid-term project presentations
- June 2 (reserved)
- June 4: Students Presentation II
  - ICRA 20
- June 9, 11
- June 16, 18: No class (CVPR 20)
- June 23, 25: Final project presentations

KAIST

Announcements

- There are only 5 students in the class
  - You can do a single-man project
  - You can bring your own research as long as it is clearly related to the course theme
- Each student
  - Give two talks; each talk time is 25 min
  - Each talk covers one main paper with related papers
- Each team
  - Give a mid-term review presentation for the project
  - Give the final project presentation
- For the online case
  - You can capture your presentation and share them with us through KLMS or youtube

Deadlines

- Declare project team members
  - By 4/6 at KLMS
- Confirm schedules of paper talks and project talks at 4/7
- Declare two papers for student presentations
  - by 4/20 at KLMS
  - Discuss them at the class time of 4/21

KAIST

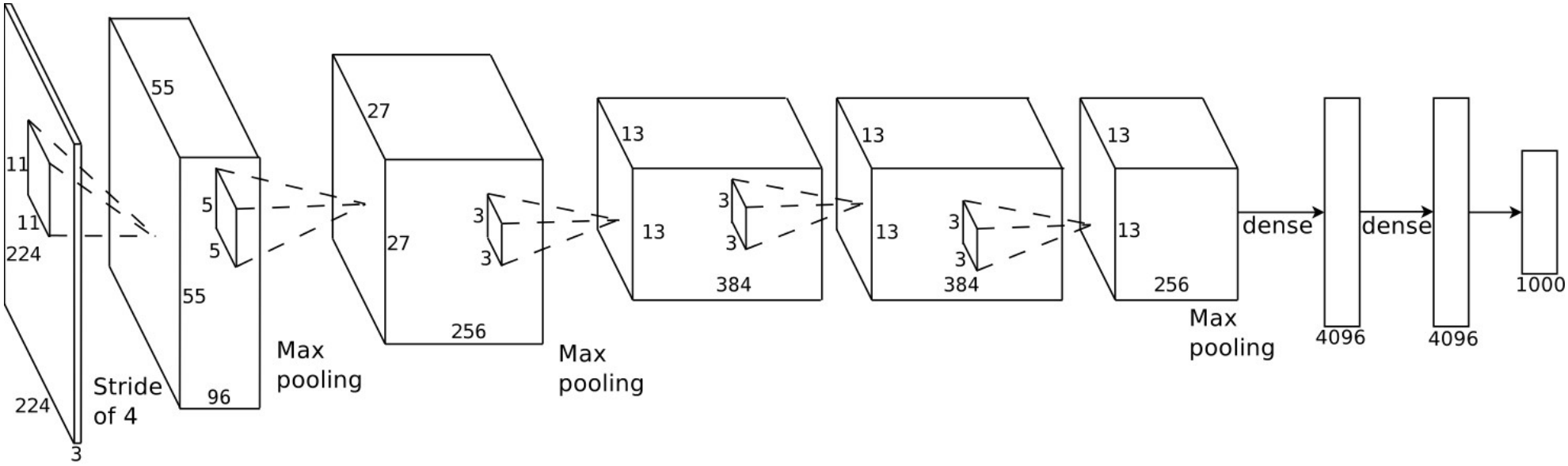
Class Objectives

- **Review basics of convolution neural nets (CNNs)**
- **At the prior class:**
  - **Browsed main components of deep neural nets**

KAIST

Convolution Neural Nets (CNNs)

- **Deep neural nets, especially, CNNs, provide low-level and high-level features**
  - **We can use those features for image search**
- **Achieve the best results in many computer vision related problems**



Krizhevsky et al., NIPS 2012

KAIST

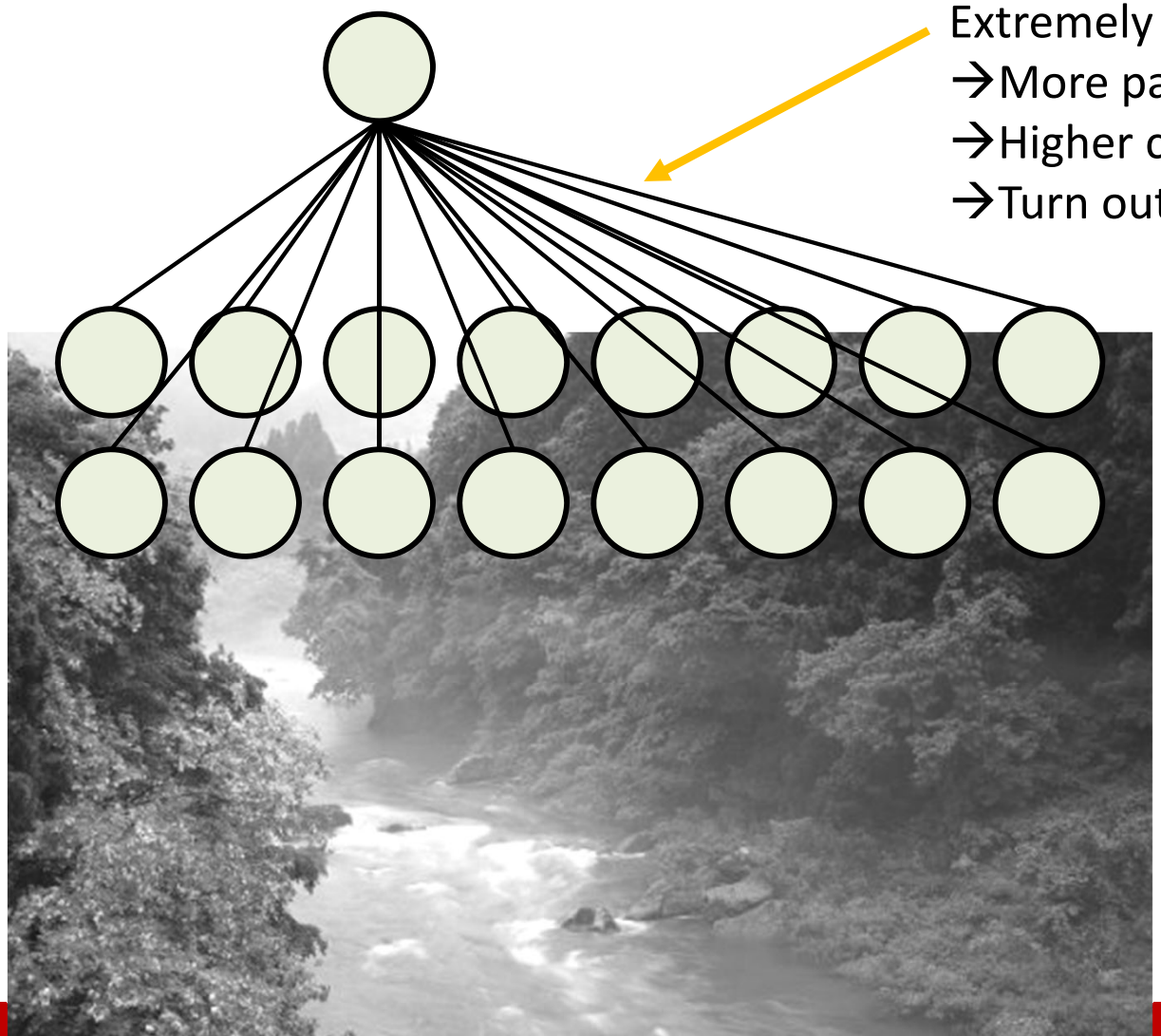
Working with images

- Major factors for features:
  - Want to have “selective” and “invariant” features.
  - Try to exploit knowledge of images to accelerate training or improve performance.

- Generally try to avoid wiring detailed visual knowledge into system --- prefer to learn.

Local connectivity

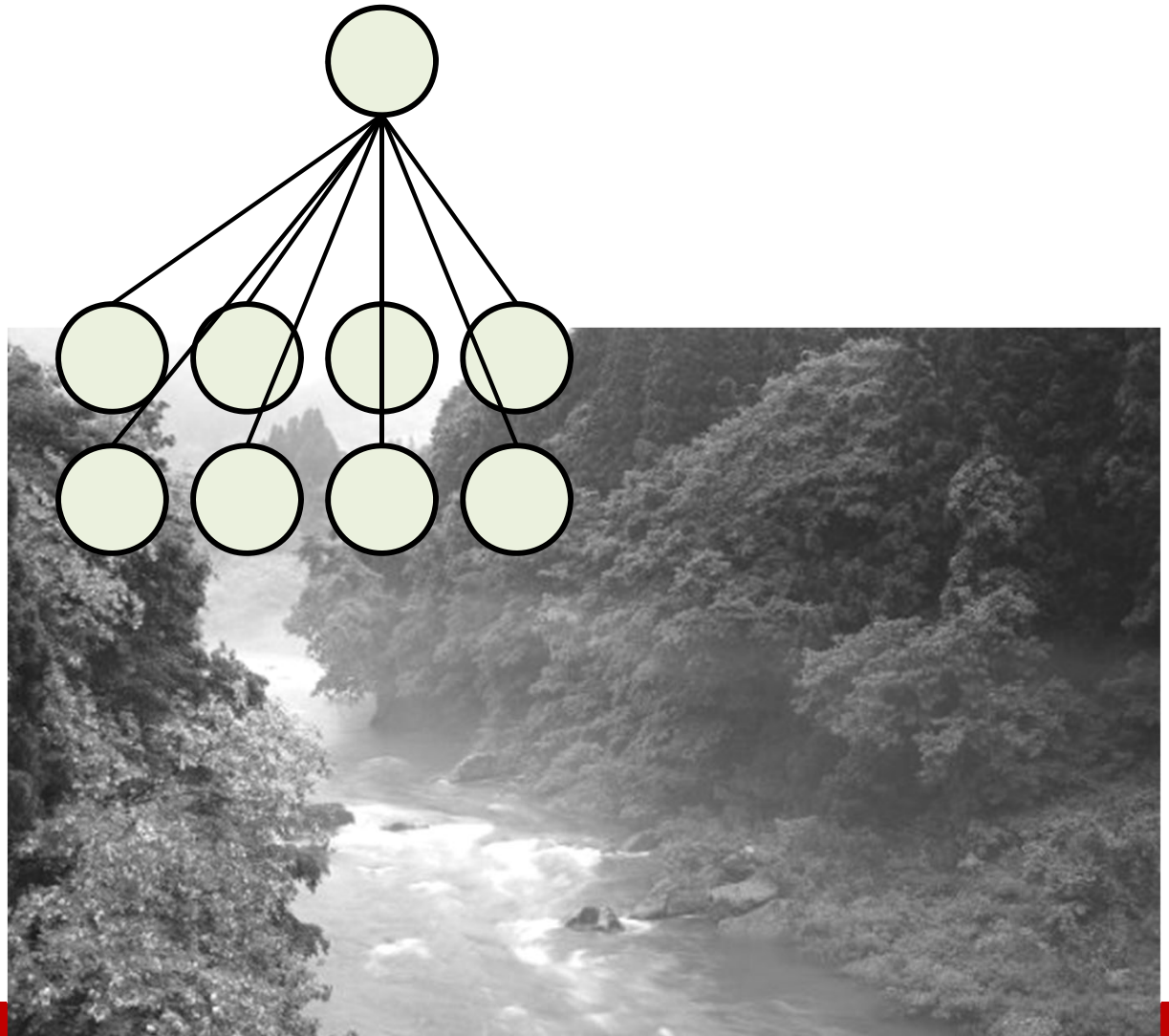
- Neural network view of single neuron:



Extremely large number of connections.
→More parameters to train.
→Higher computational expense.
→Turn out not to be helpful in practice.

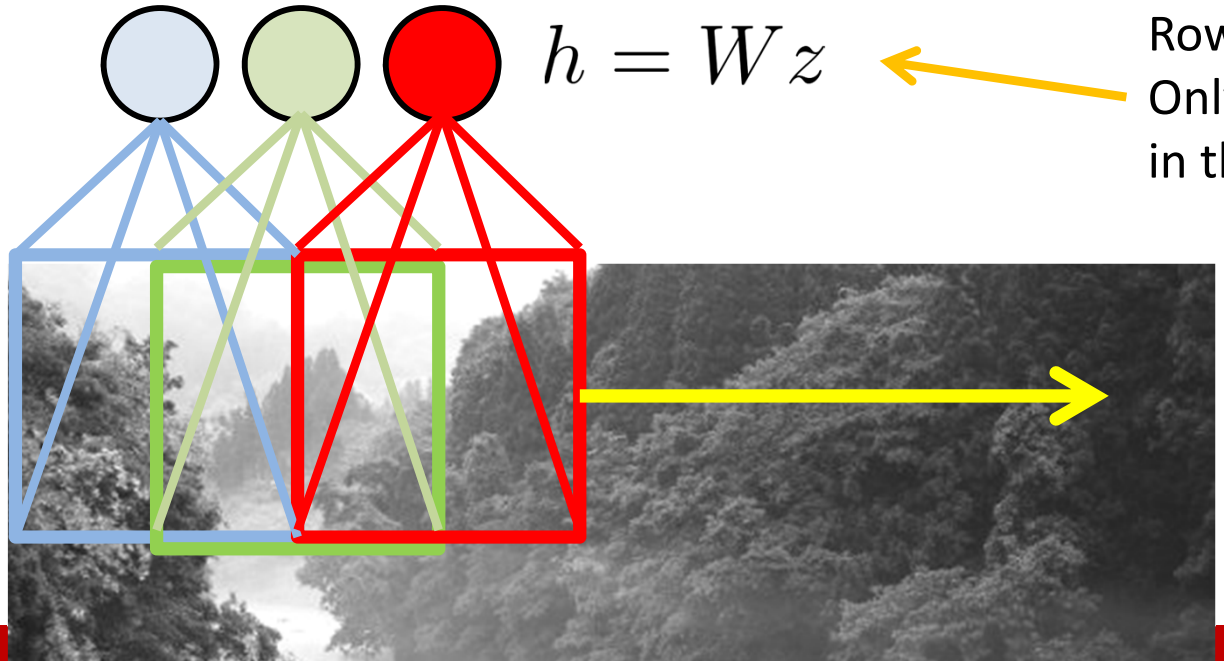
Local connectivity

- Reduce parameters with local connections.
  - Weight vector is a spatially localized “filter”.



Local connectivity

- Sometimes think of neurons as viewing small adjacent windows.
  - Specify connectivity by the size ("receptive field" size) and spacing ("step" or "stride") of windows.
    - Typical RF size = 5 to 20
    - Typical step size = 1 pixel up to RF size.



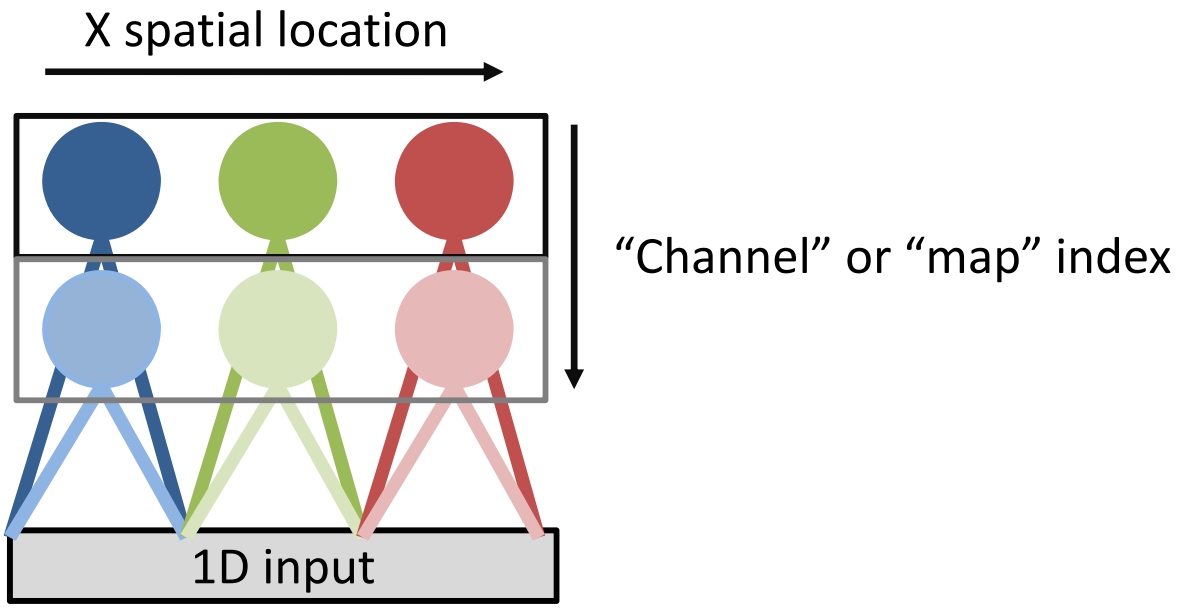
$$h = Wz$$

Rows of W are sparse.
Only weights connecting to inputs in the window are non-zero.

Local connectivity

- Spatial organization of filters means output features can also be organized like an image.
  - X,Y dimensions correspond to X,Y position of neuron window.
  - “Channels” are different features extracted from same spatial location. (Also called “feature maps”, or “maps”.)

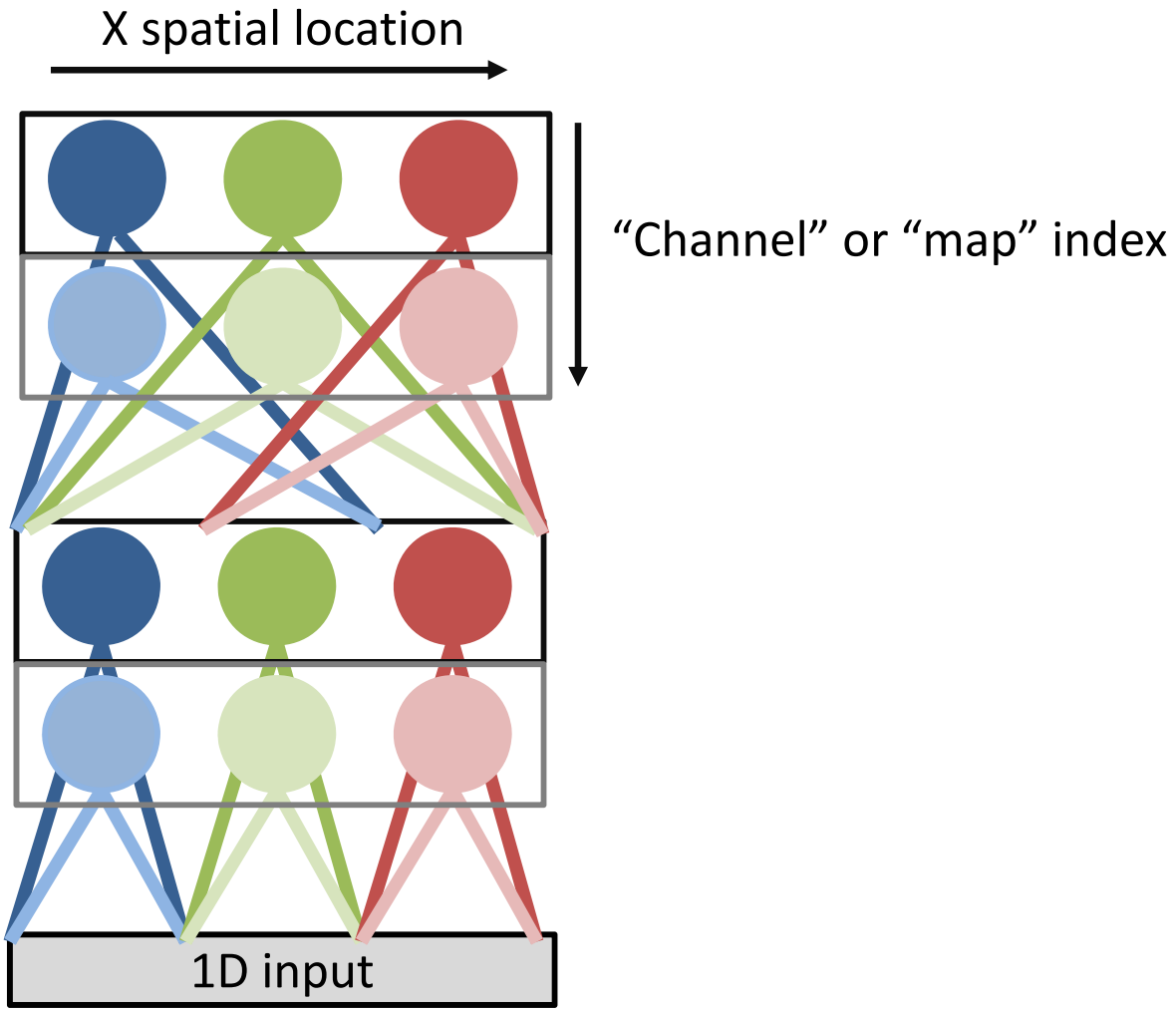
1-dimensional example:



Local connectivity

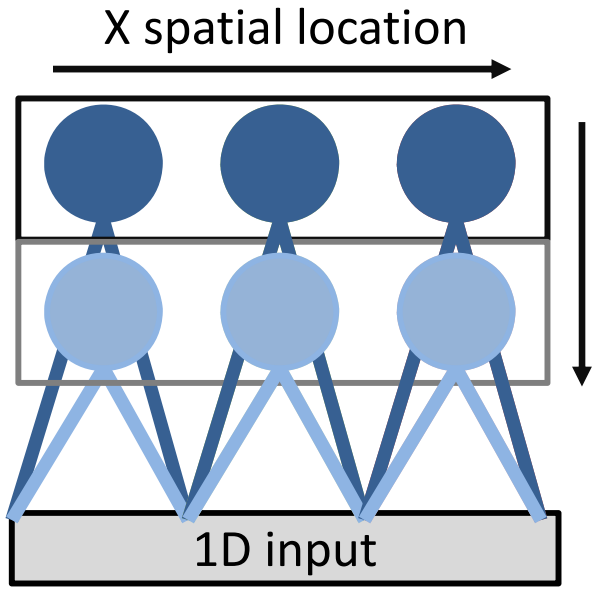
- We can treat output of a layer like an image and re-use the same tricks.

1-dimensional example:



Weight-Tying or Convolutional Network

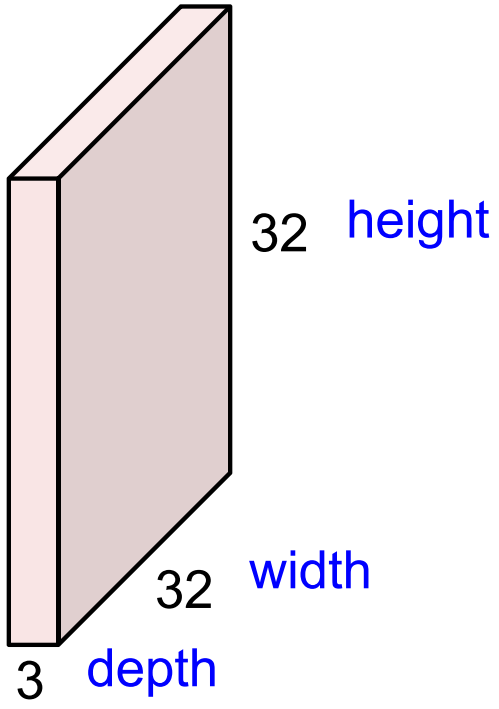
- Even with local connections, may still have too many weights.
  - Images tend to be “stationary”: different patches tend to have similar low-level structure.



- Each unique filter is spatially convolved with the input to produce responses for each map. [LeCun et al., 1989; LeCun et al., 2004]

Convolution Layer

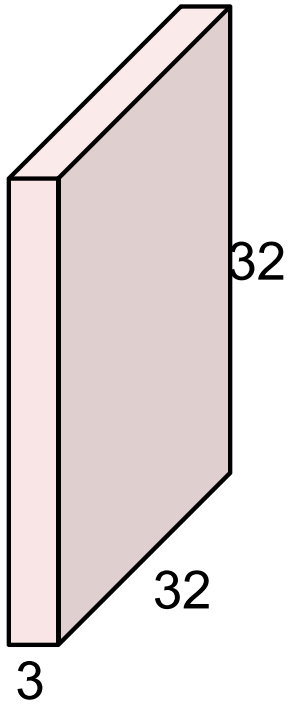
32x32x3 image -> preserve spatial structure



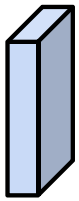
Convolution Layer

Filters always extend the full depth of the input volume

32x32x3 image

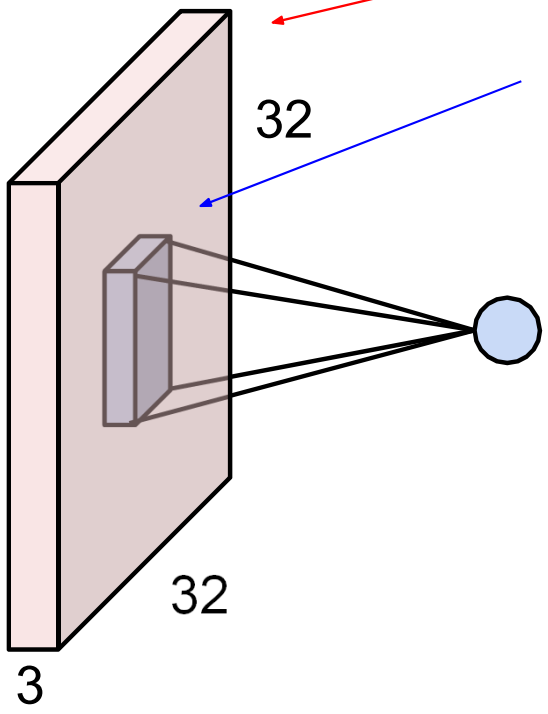


5x5x3 filter



Convolve the filter with the image
i.e. “slide over the image spatially, computing dot products”

Convolution Layer



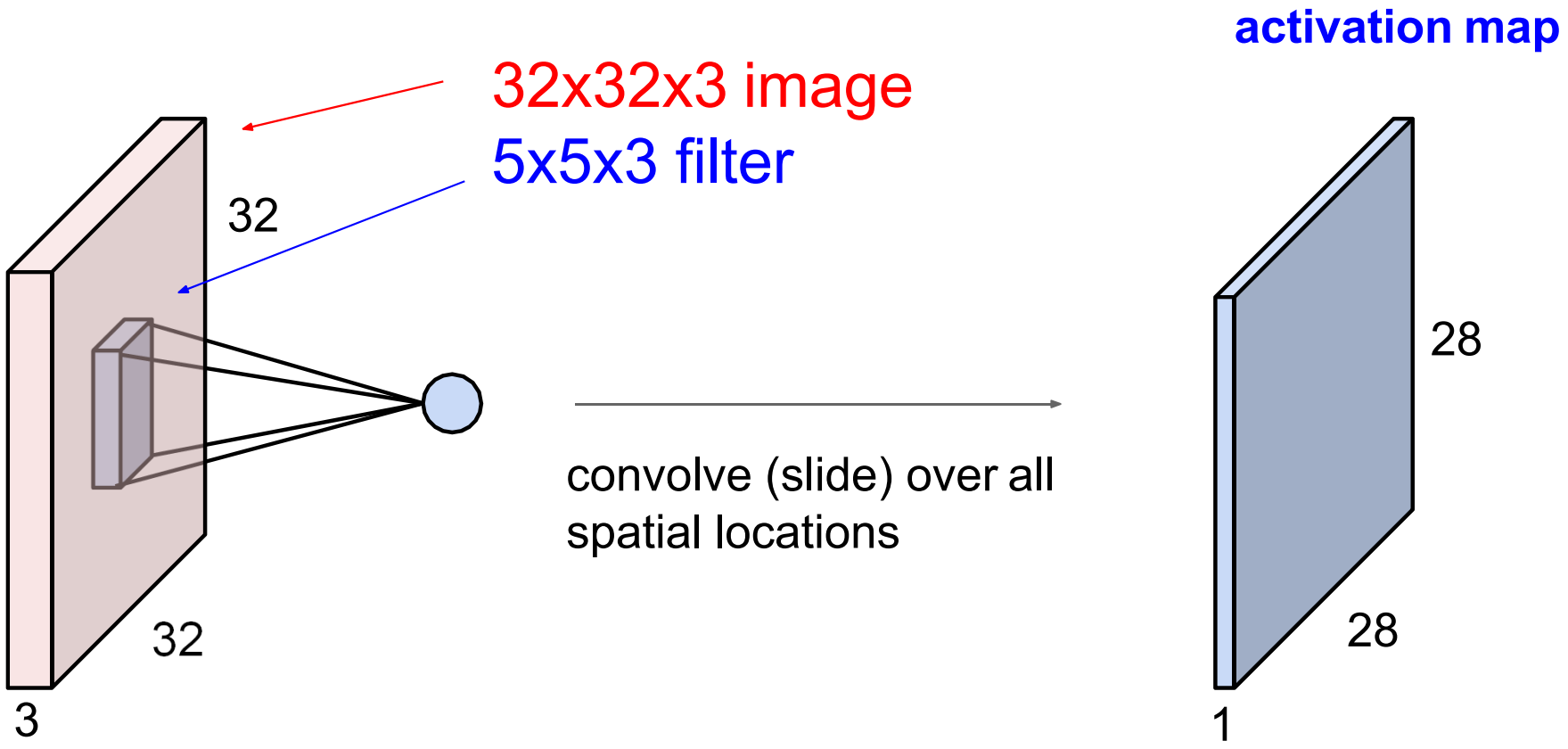
32x32x3 image

5x5x3 filter w

1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image
(i.e. 5*5*3 = 75-dimensional dot product + bias)

$$w^T x + b$$

Convolution Layer

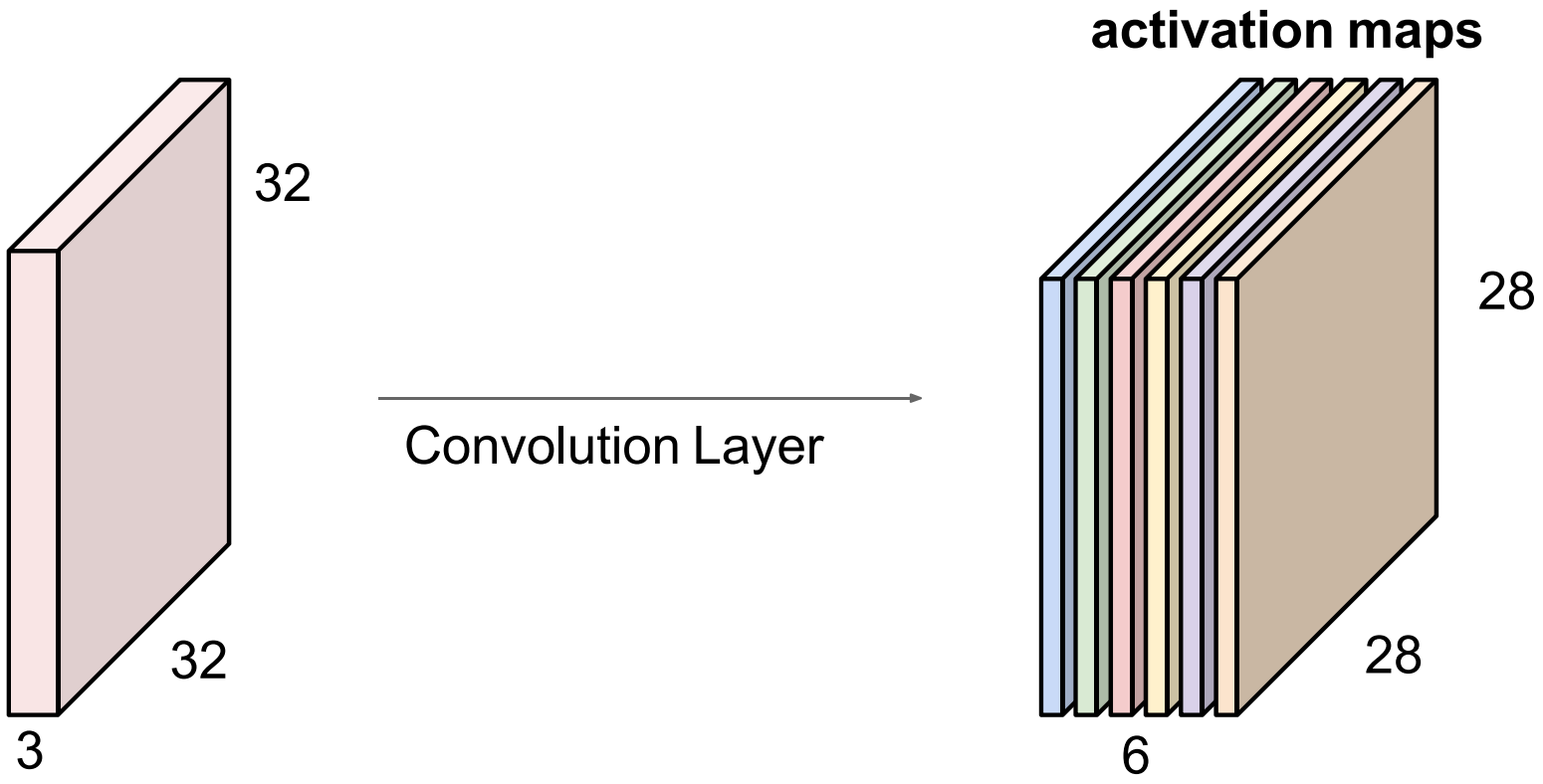


Convolution Layer

consider a second, green filter

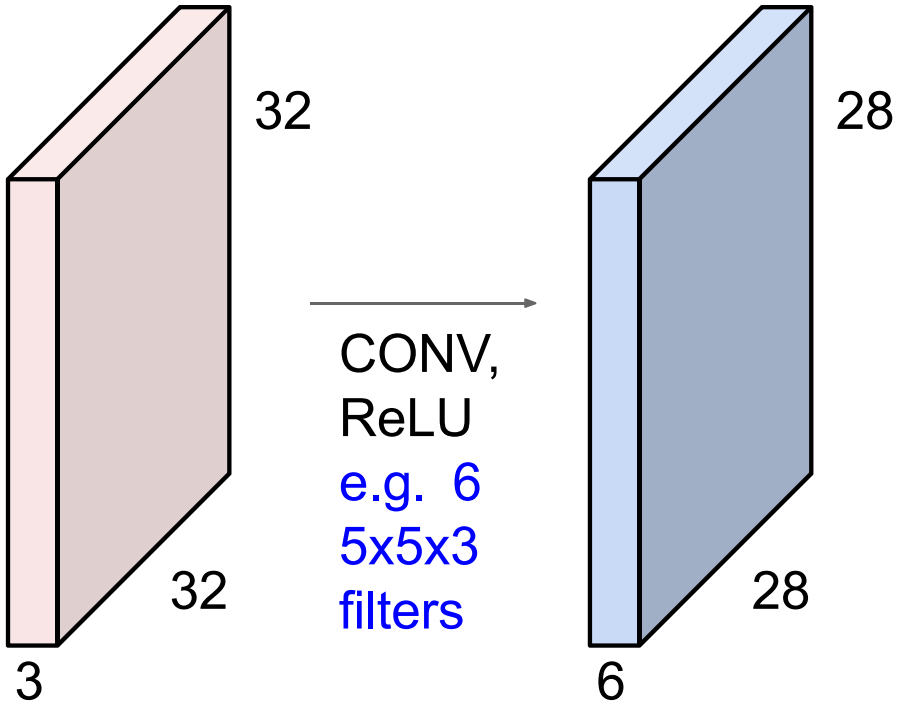
Ack. Fei-Fei Li & Justin Johnson & Serena Yeung

For example, if we had 6 5x5 filters, we'll get 6 separate activation maps:

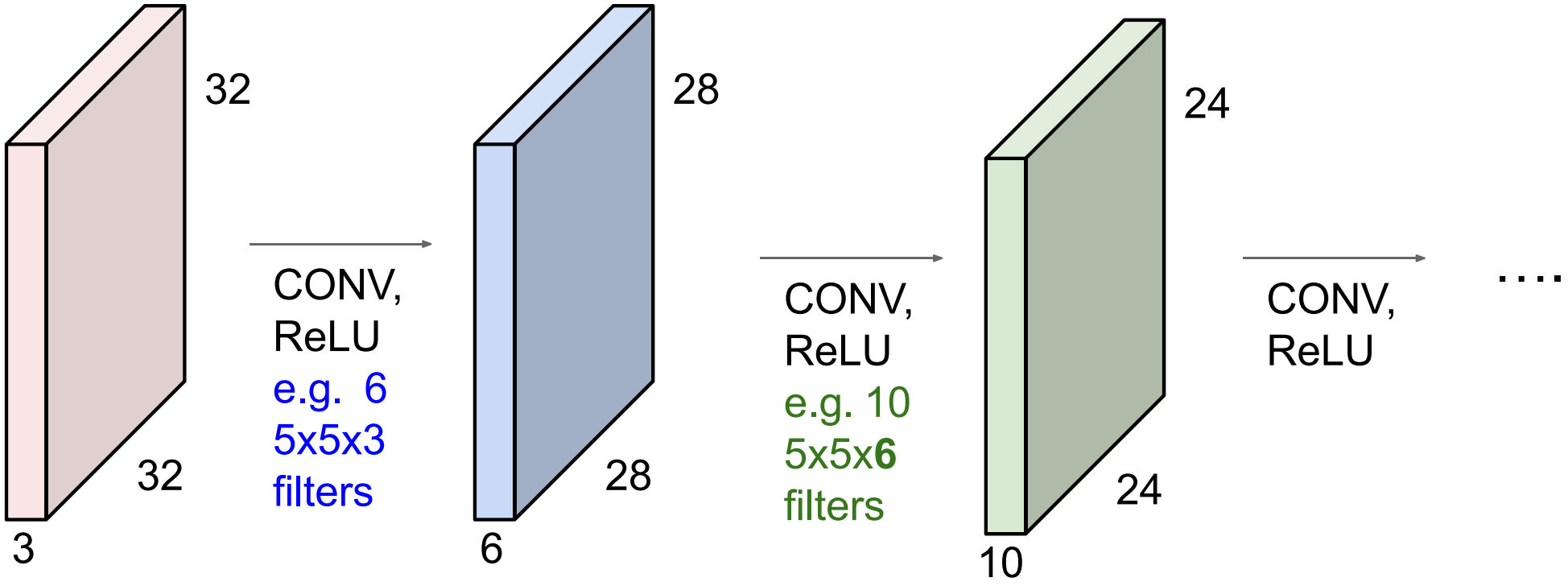


We stack these up to get a "new image" of size 28x28x6!

Preview: ConvNet is a sequence of Convolution Layers, interspersed with activation functions



Preview: ConvNet is a sequence of Convolution Layers, interspersed with activation functions

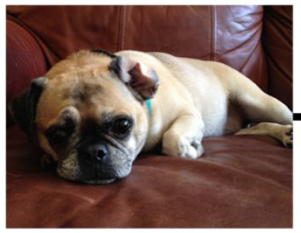


Ack. Fei-Fei Li & Justin Johnson & Serena Yeung

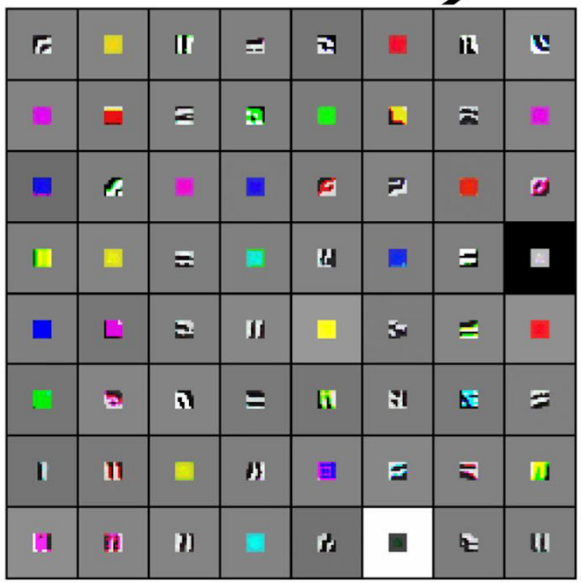
Preview

[Zeiler and Fergus 2013]

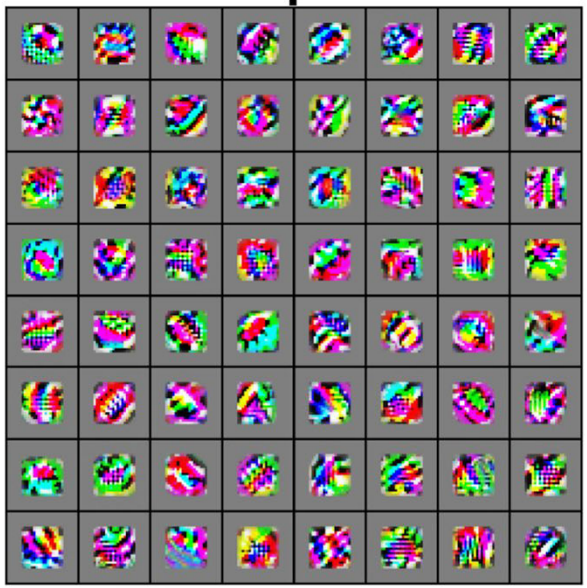
Visualization of VGG-16 by Lane McIntosh. VGG-16 architecture from [Simonyan and Zisserman 2014].



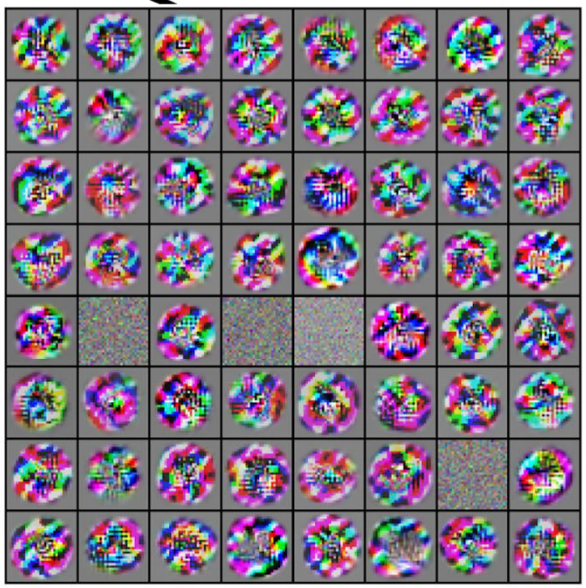
Low-level features → Mid-level features → High-level features → Linearly separable classifier



VGG-16 Conv1_1



VGG-16 Conv3_2

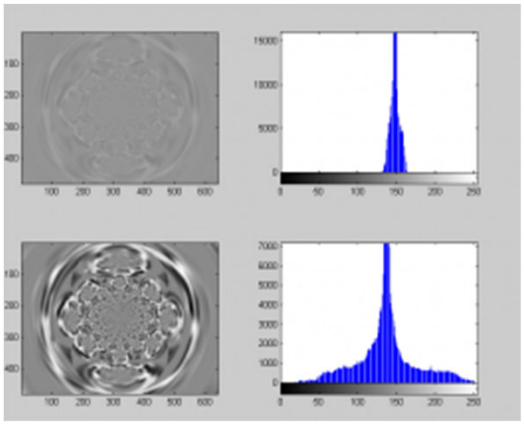


VGG-16 Conv5_3

Ack. Fei-Fei Li & Justin Johnson & Serena Yeung

Contrast Normalization

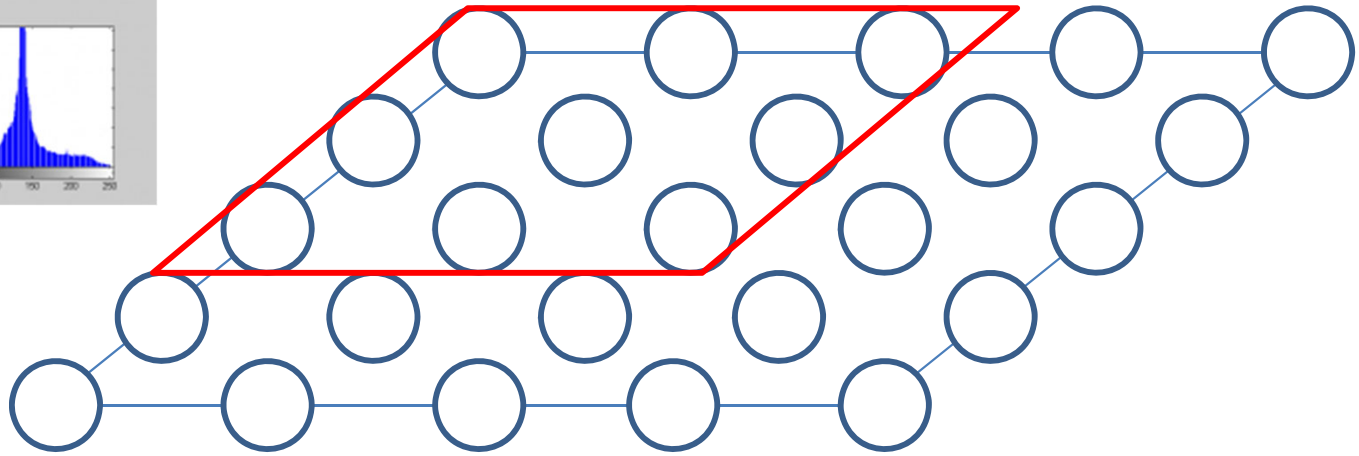
- Empirically useful to soft-normalize magnitude of groups of neurons.
  - Subtract out the local mean first.



Giassa.net

$$h = \frac{z}{\sqrt{W z^2 + \epsilon}}$$

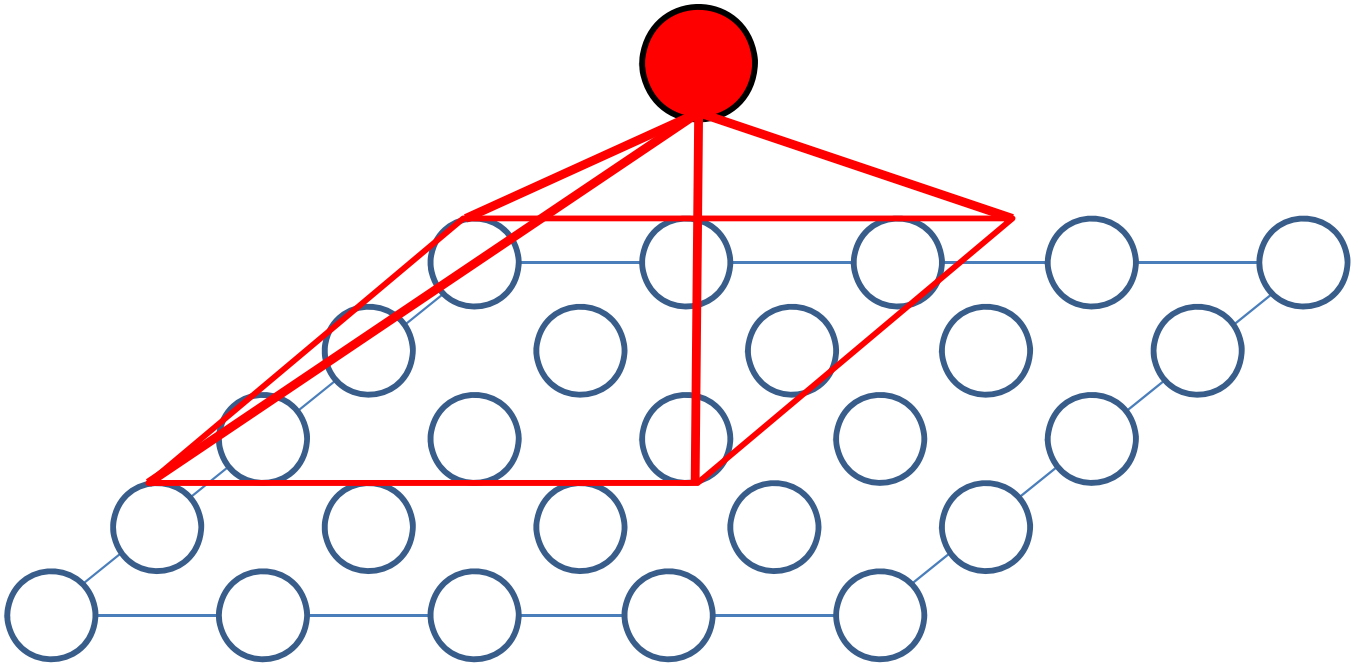
std.



[Jarrett et al., ICCV 2009]

Pooling

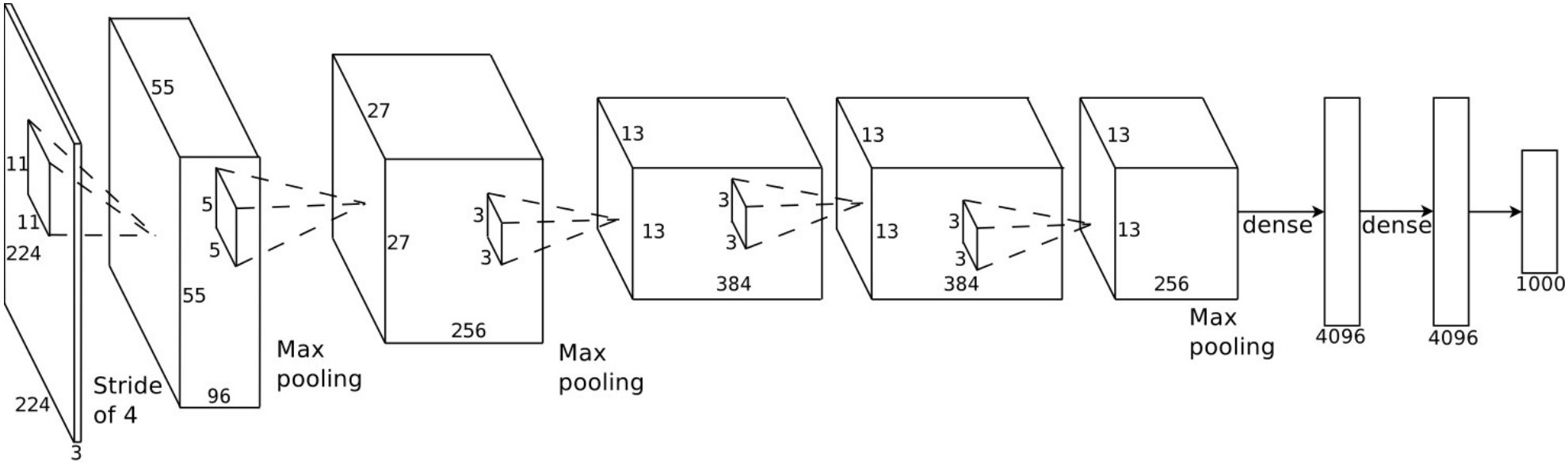
- Functional layers designed to represent invariant features.
  - Combined with convolution, corresponds to hard-wired translation invariance.
- Usually fix weights to local box or Gaussian filter.
  - Easy to represent max-, average-, or 2-norm pooling.



[Scherer et al., ICANN 2010]
[Boureau et al., ICML 2010]

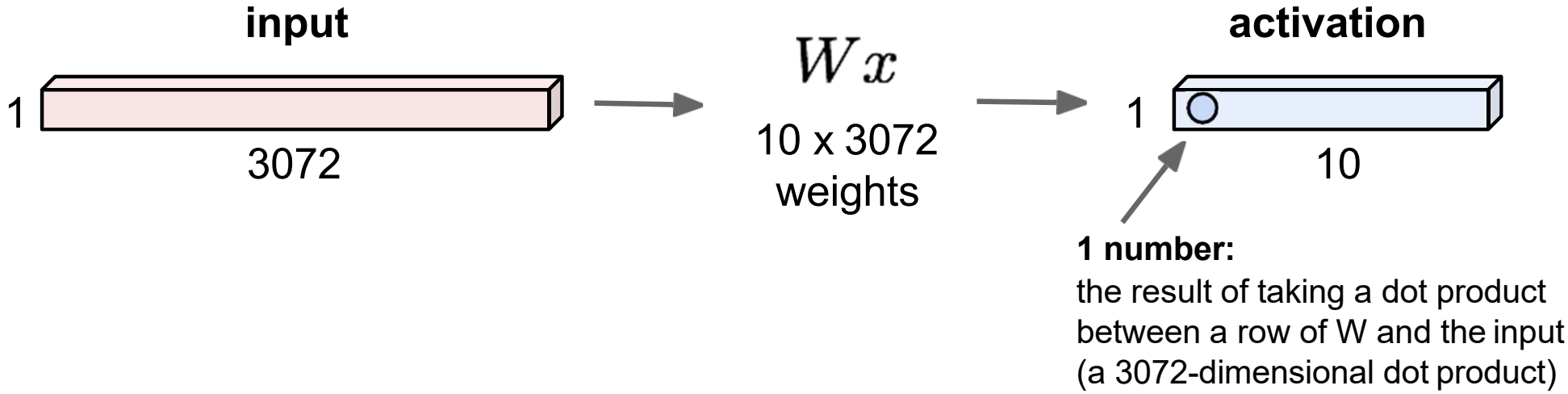
Application: Image-Net

- System from Krizhevsky et al., NIPS 2012:
  - Convolutional neural network.
  - Local connectivity.
  - Max-pooling.
  - Rectified linear units (ReLu).
  - Contrast normalization.



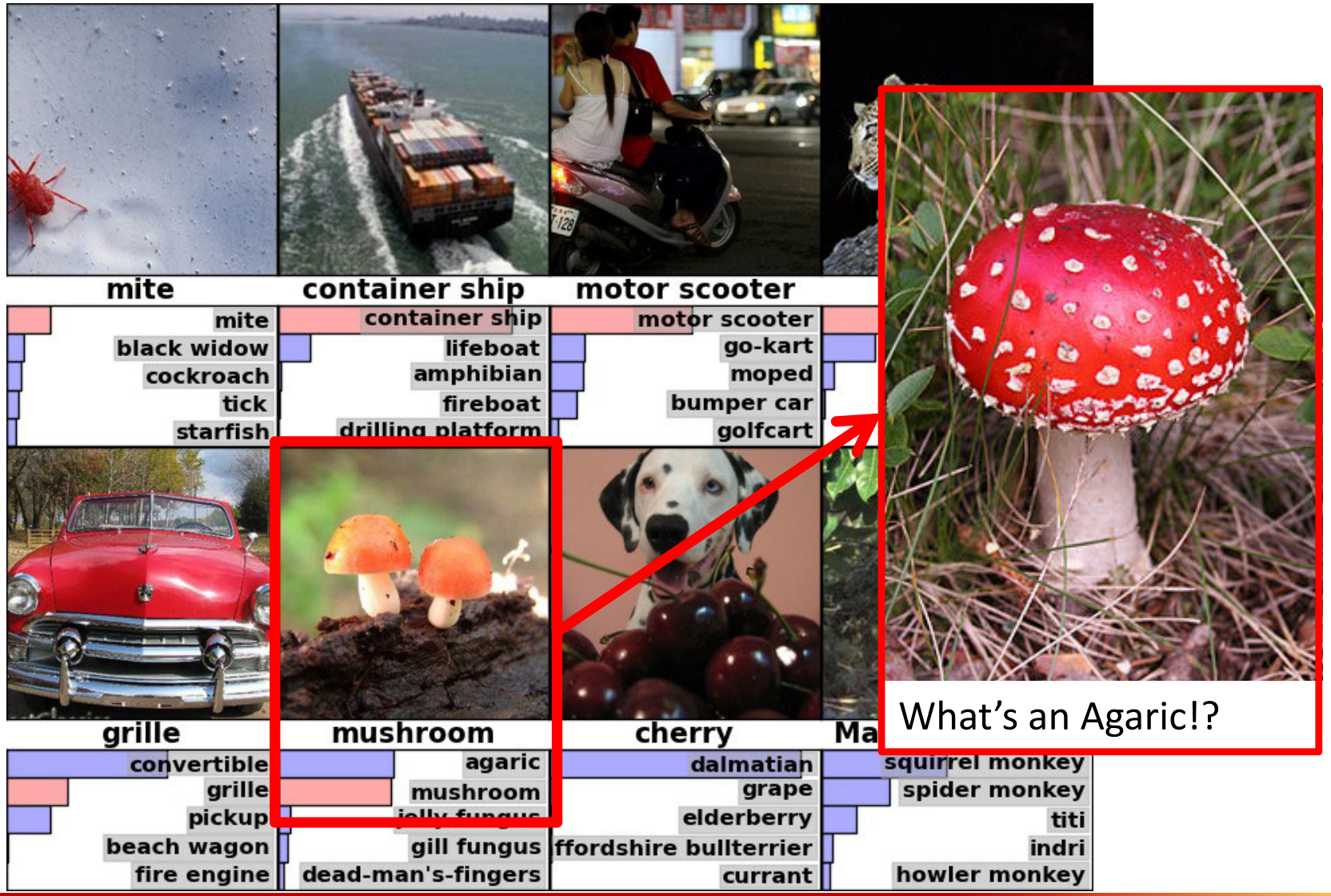
Fully Connected Layer

32x32x3 image -> stretch to 3072 x 1



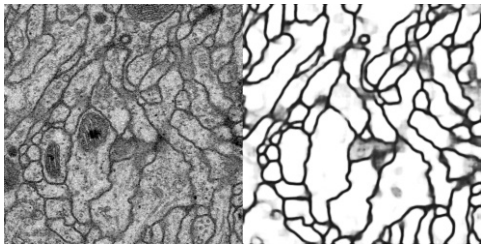
Application: Image-Net

- Top result in LSVRC 2012: ~85%, Top-5 accuracy.



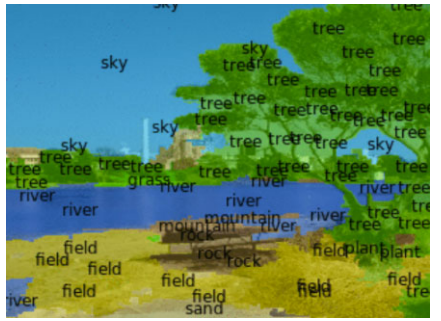
More applications

- Segmentation: predict classes of pixels / super-pixels.



← Ciresan et al., NIPS 2012

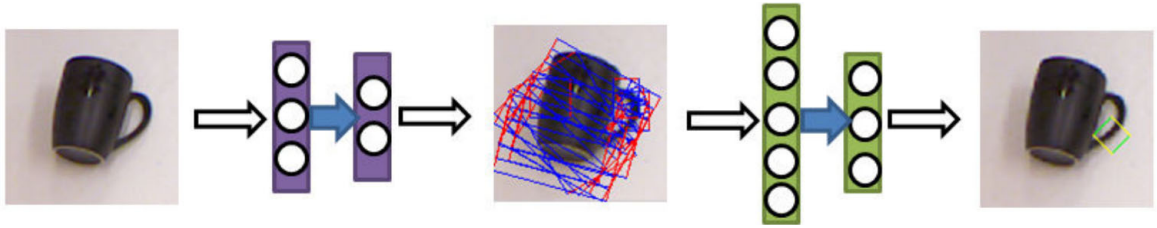
Farabet et al., ICML 2012 →



- Detection: combine classifiers with sliding-window architecture.
  - Economical when used with convolutional nets.

Pierre Sermanet (2010) →

- Robotic grasping. [Lenz et al., RSS 2013]



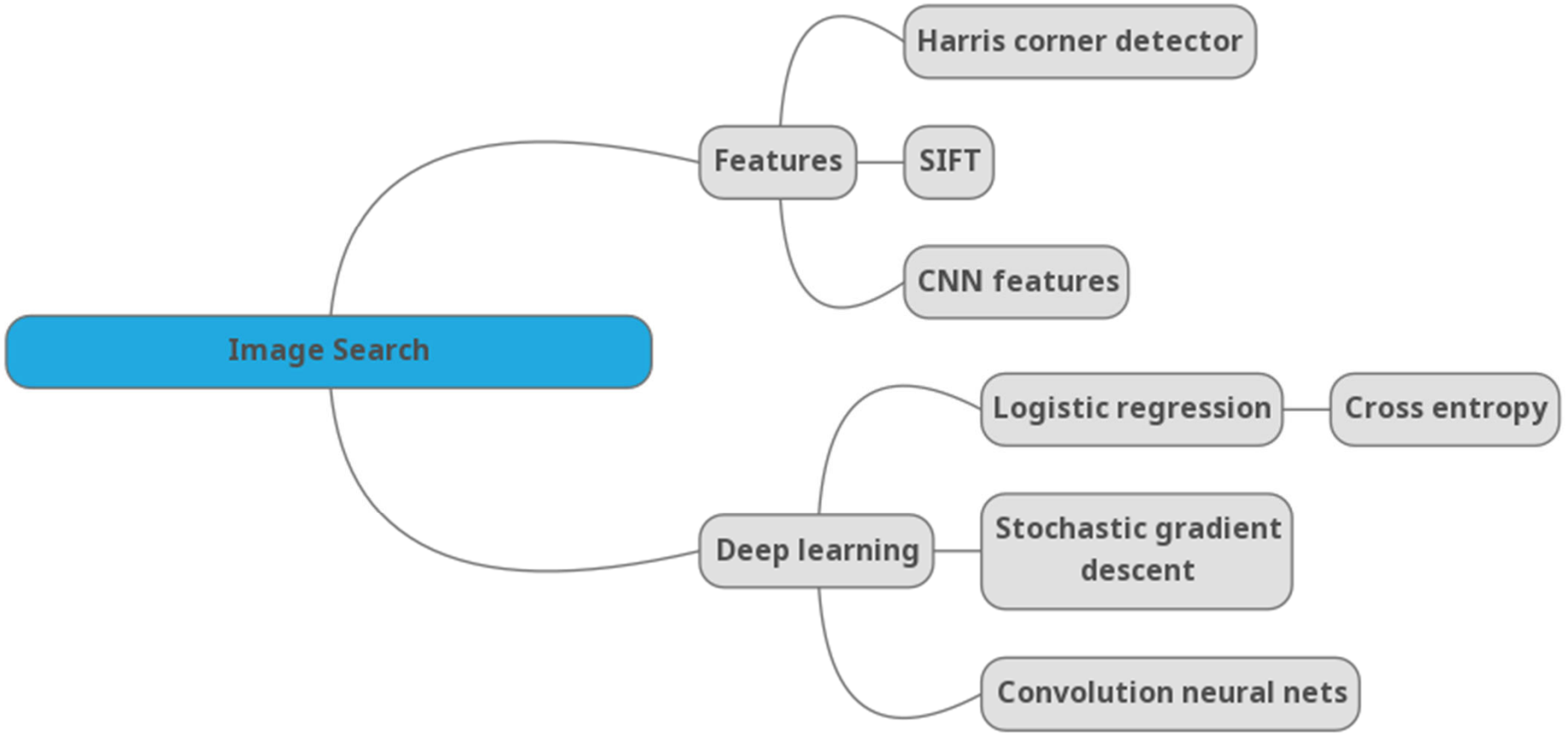
http://www.youtube.com/watch?v=f9CuzqI1SkE

Class Objectives were:

- **Study convolution neural nets (CNNs)**
- **At the prior class:**
  - **Browsed main components of deep neural nets**

KAIST

Summary up to Now



KAIST

Next Time

- **Bag-of-visual-Words (BoW) model**

KAIST

Homework for Every Class

- Go over the next lecture slides
- Come up with one question on what we have discussed today
  - 1 for typical questions (that were answered in the class)
  - 2 for questions with thoughts or that surprised me
- Write questions 3 times before the mid-term exam
  - Write a question about one out of every four classes
  - Multiple questions in one time will be counted as one time
- Common questions are compiled at the Q&A file
  - Some of questions will be discussed in the class
- If you want to know the answer of your question, ask me or TA on person