CSC4007 Advanced Machine Learning

Lesson 08: Convolutional Neural Networks

by Vien Ngo
EEECS / ECIT / DSSC

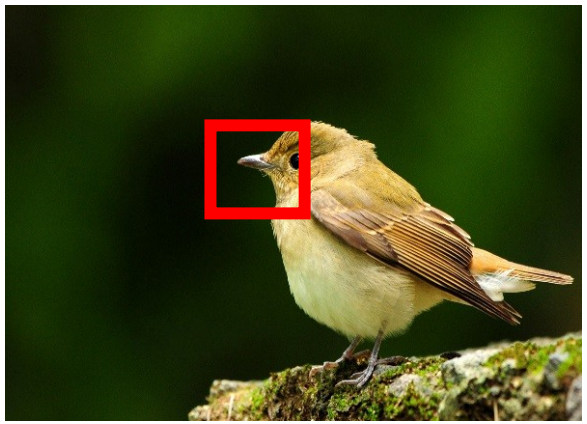
Some slides adapted from Barry Devereux, Andrej Karpathy's CS231N lectures

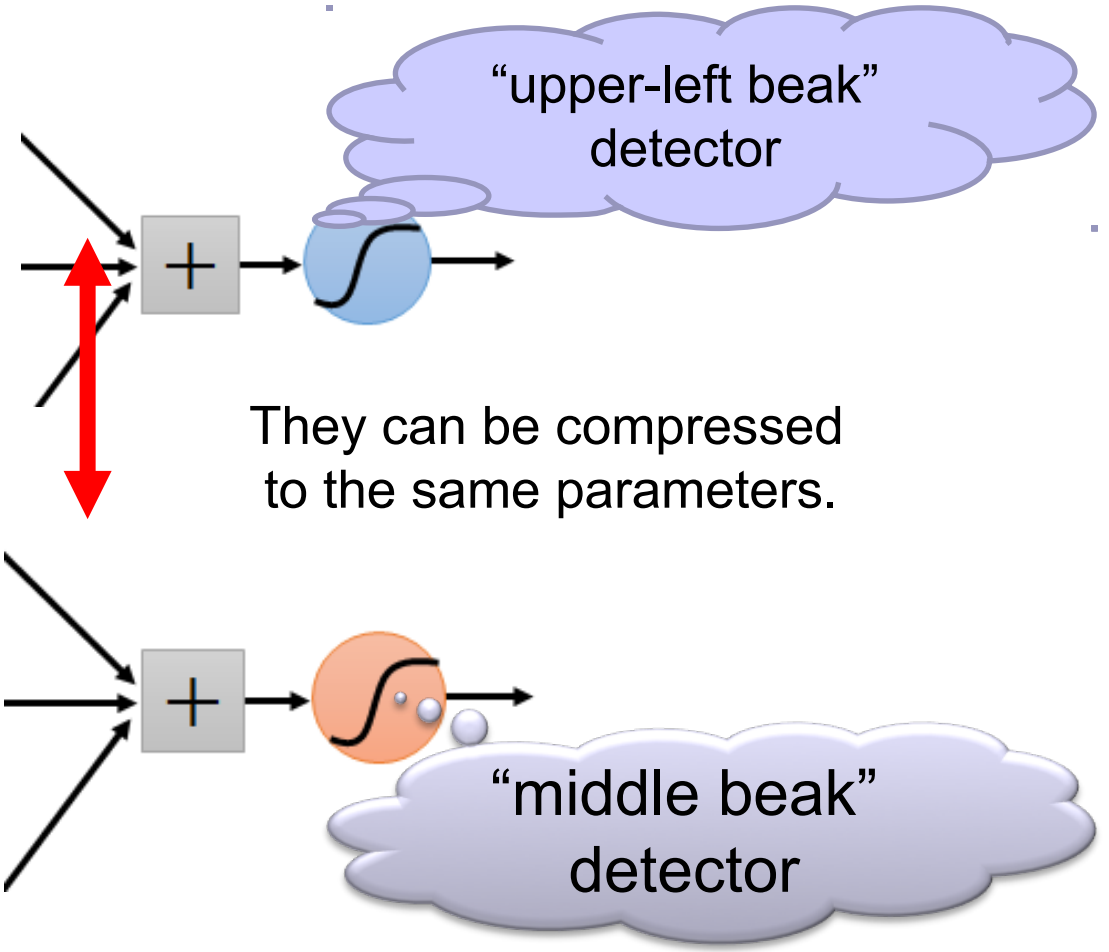
Outline

- Neural network basics and representation
- Perceptron learning, multi-layer perceptron
- Neural network training: Backpropagation
- Modern neural network architecture (a.k.a Deep learning):
  - **Convolutional neural network (CNN)**
  - Recurrent neural network (RNN), long-short term memory network (LSTM)

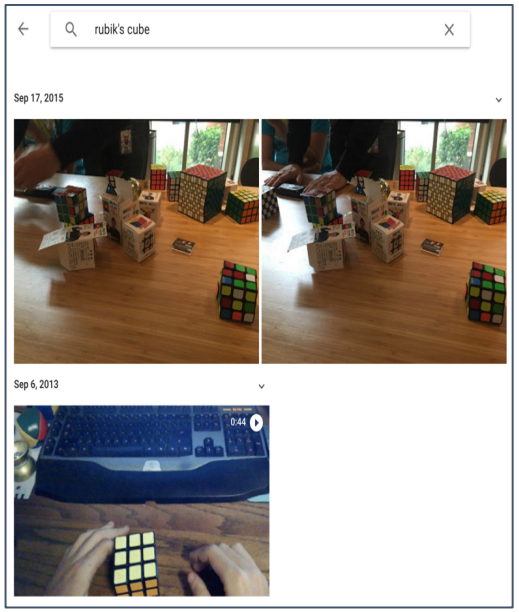
Consider learning an image

- Some patterns are much smaller than the whole image
- Same pattern appears in different places: They can be compressed!
- What about training a lot of such “small” detectors and each detector must “move around”.

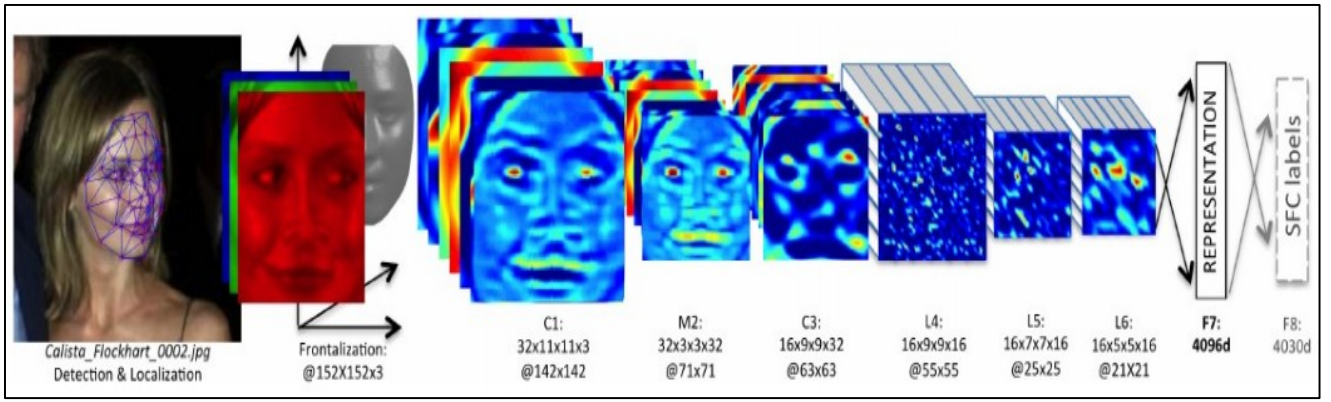




CNNs are everywhere…



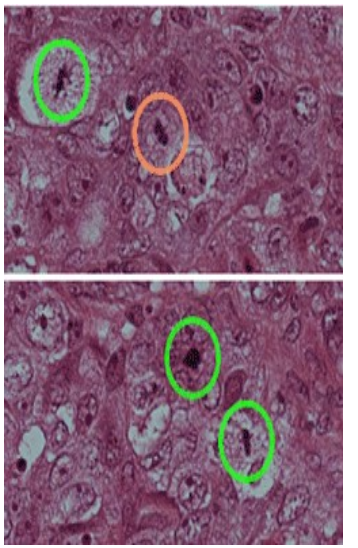
e.g. Google Photos search



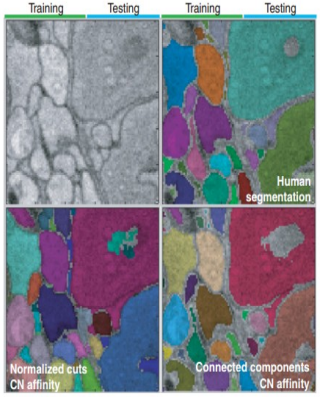
Face Verification, Taigman et al. 2014 (FAIR)

[Goodfellow et al. 2014]

Self-driving cars



Ciresan et al. 2013



Turaga et al 2010

Convolution

1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

stride=1

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

Dot product

3 -1

6 x 6 image

Convolution

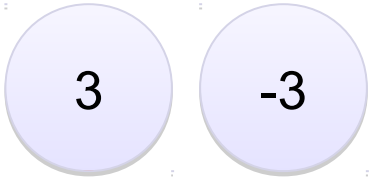
1	-1	-1
-1	1	-1
-1	-1	1

Filter 1

If stride=2

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

6 x 6 image



Convolution

-1	1	-1
-1	1	-1
-1	1	-1

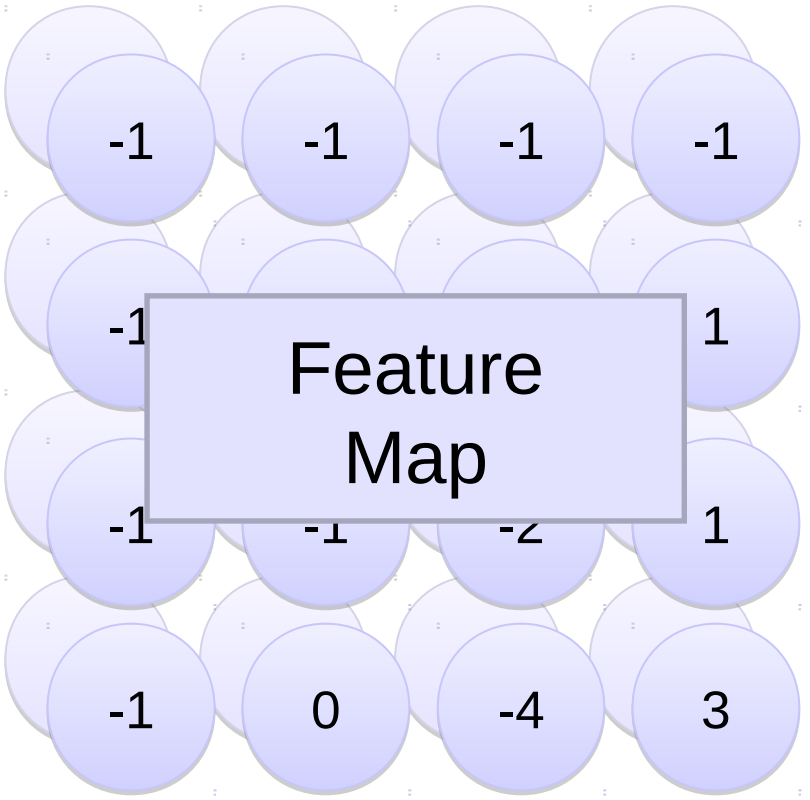
Filter 2

stride=1

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

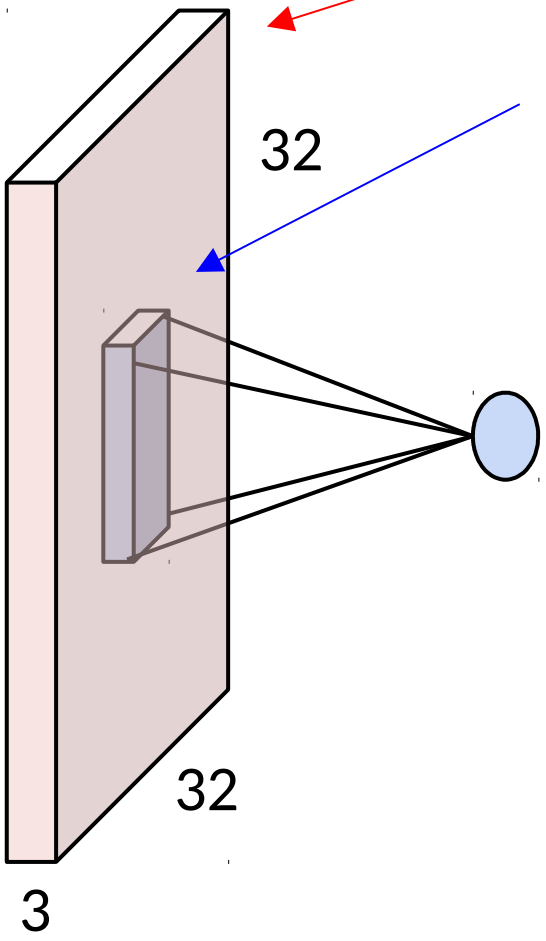
6 x 6 image

Repeat this for each filter



Two 4 x 4 images
Forming 2 x 4 x 4 matrix

Convolution Layer



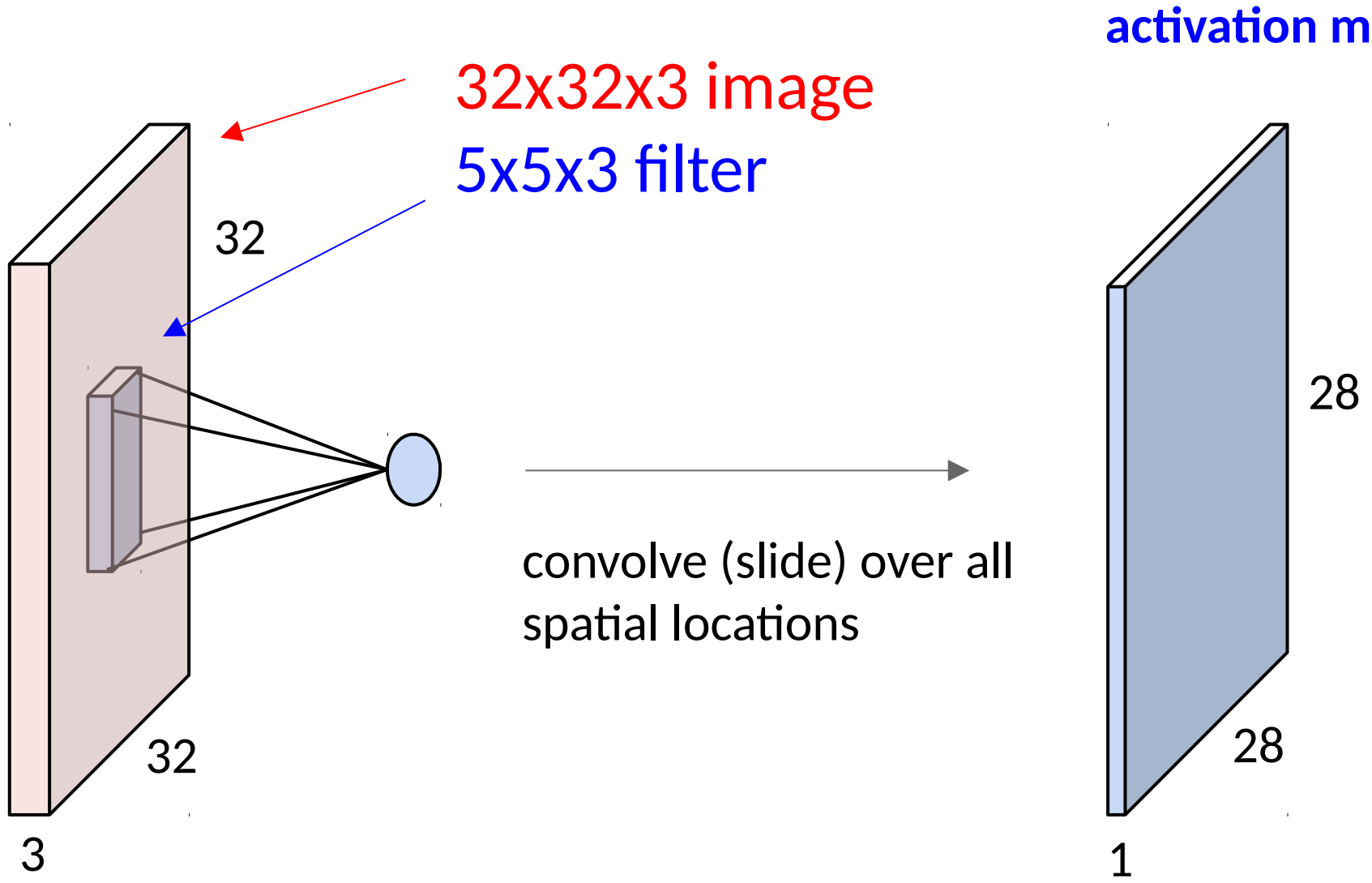
32x32x3 image

5x5x3 filter $w_{i,j,k}$

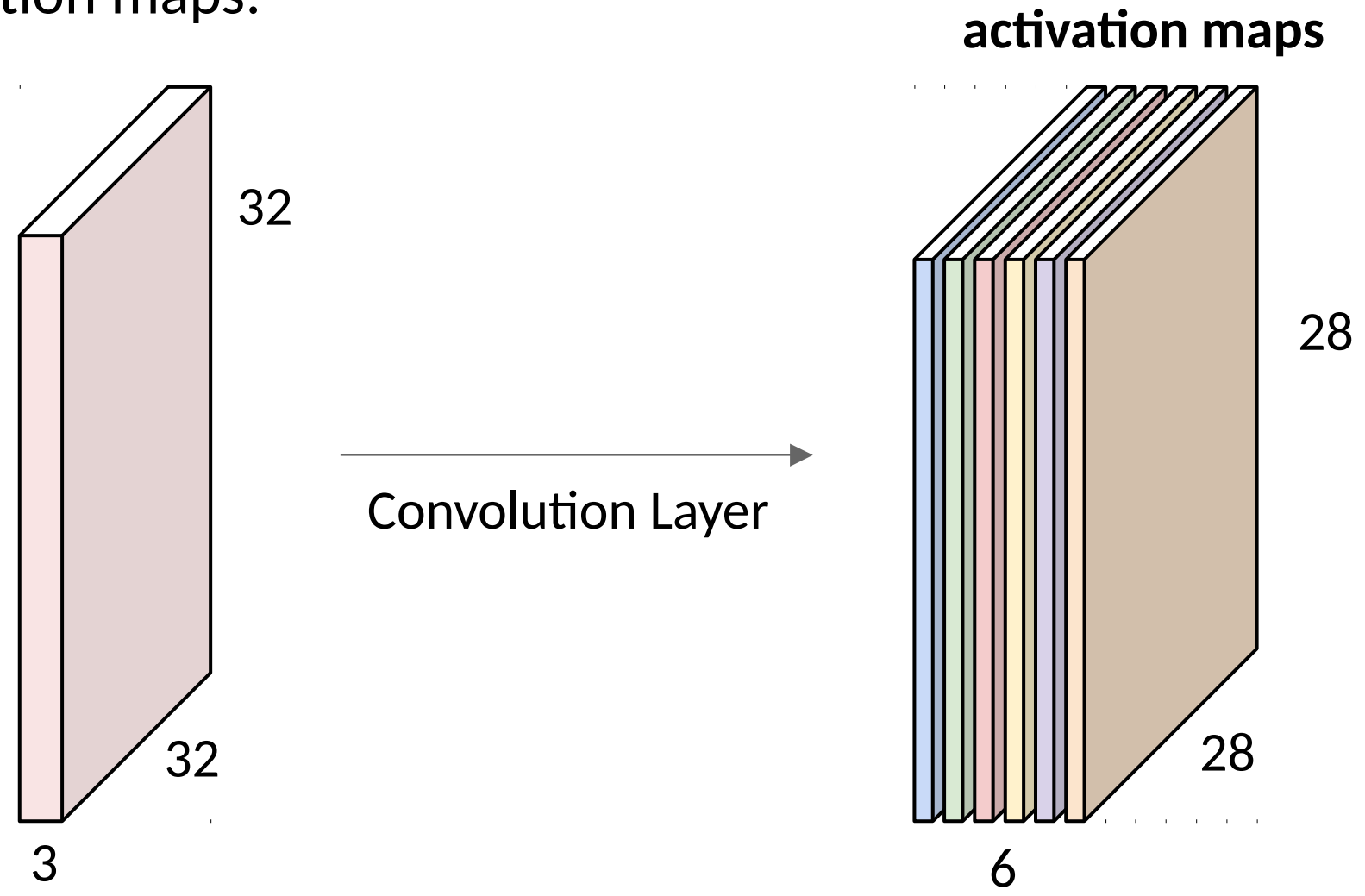
1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image
(i.e. 5*5*3 = 75-dimensional dot product + bias)

$$w^T x + b$$

Convolution Layer

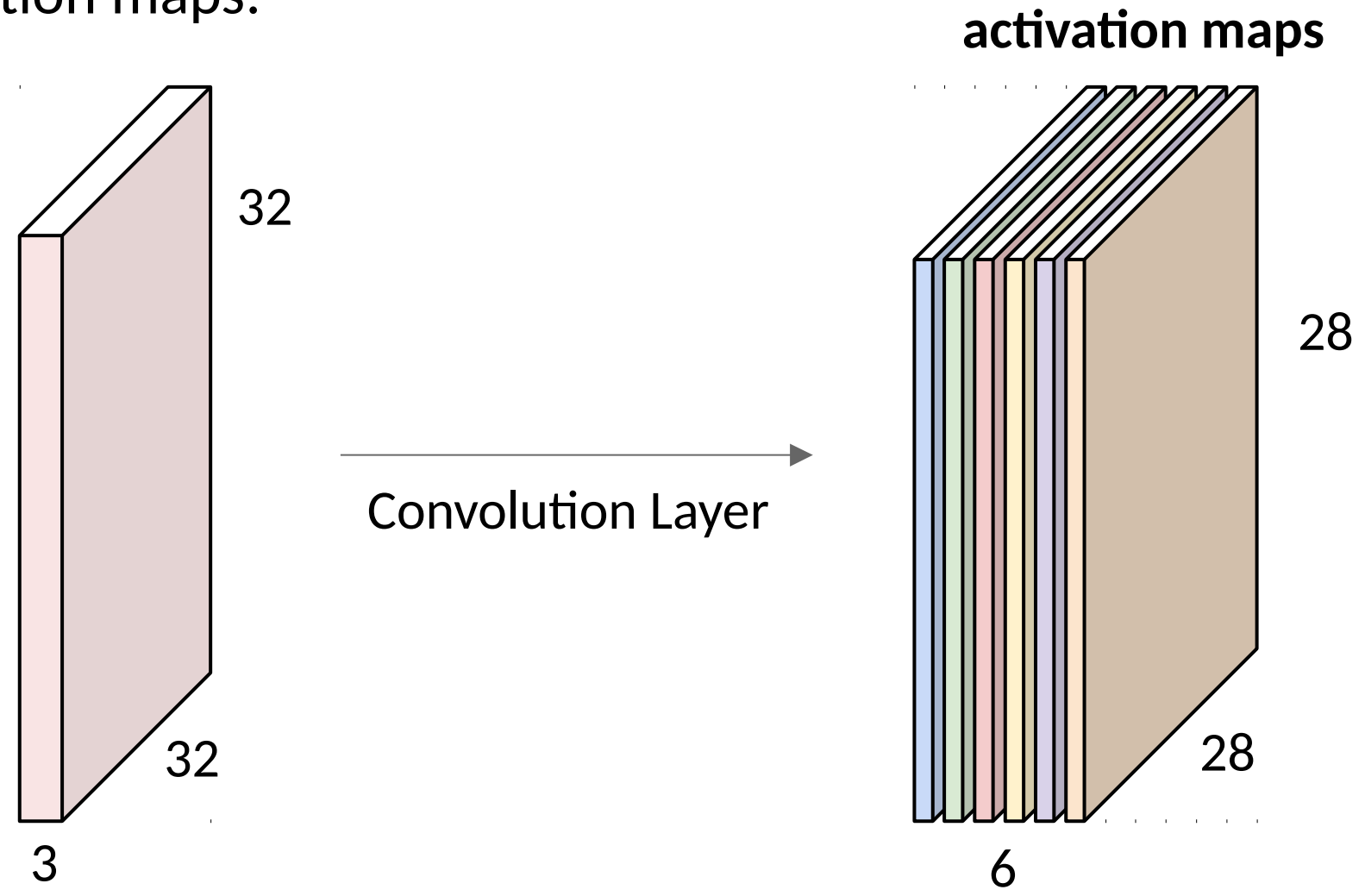


For example, if we had 6 (5x5x3) filters, we'll get 6 separate activation maps:



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

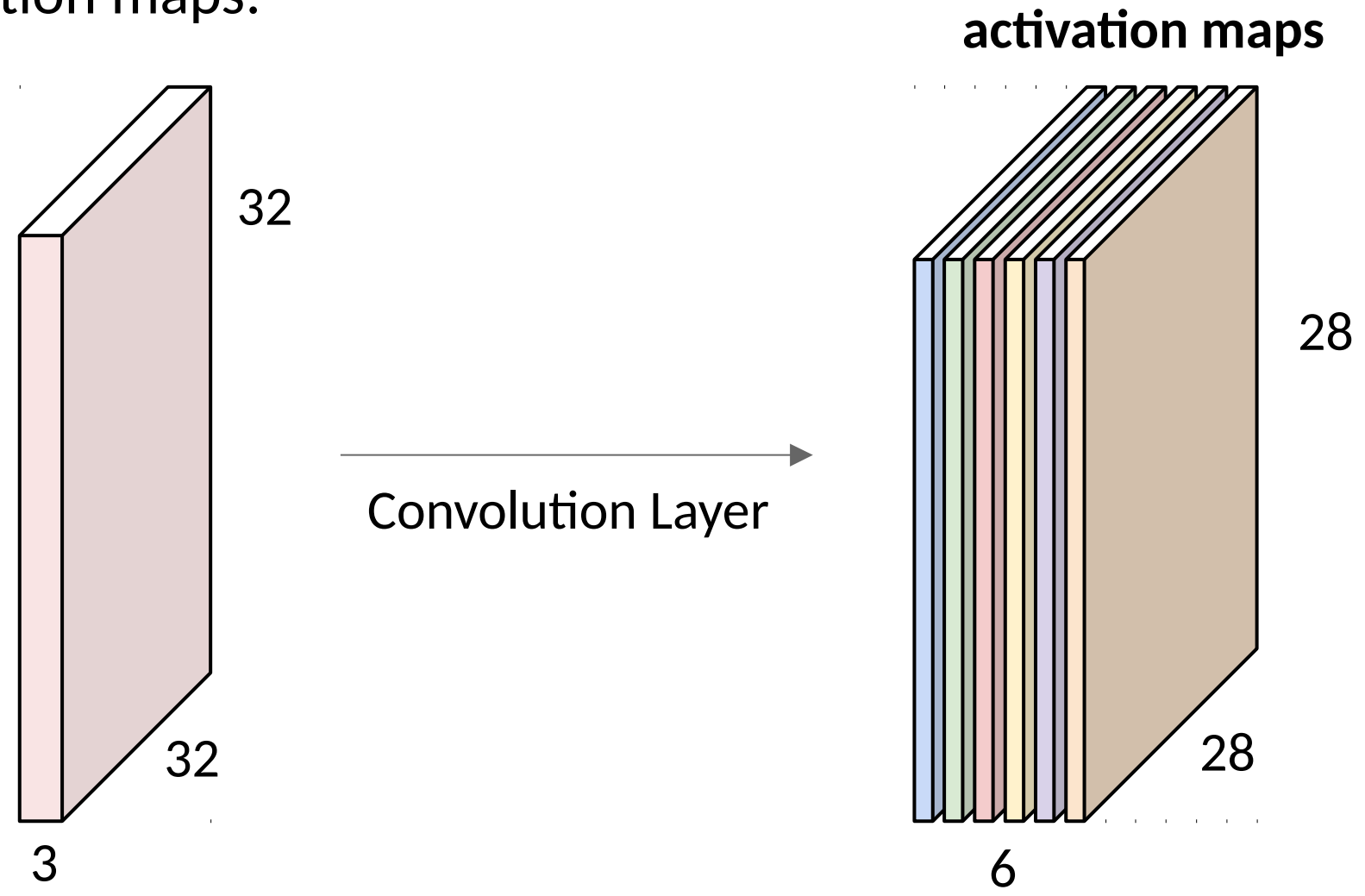
For example, if we had 6 (5x5x3) filters, we'll get 6 separate activation maps:



We processed [32x32x3] volume into [28x28x6] volume.

Q: how many parameters would this be if we used a fully connected layer instead?

For example, if we had 6 (5x5x3) filters, we'll get 6 separate activation maps:

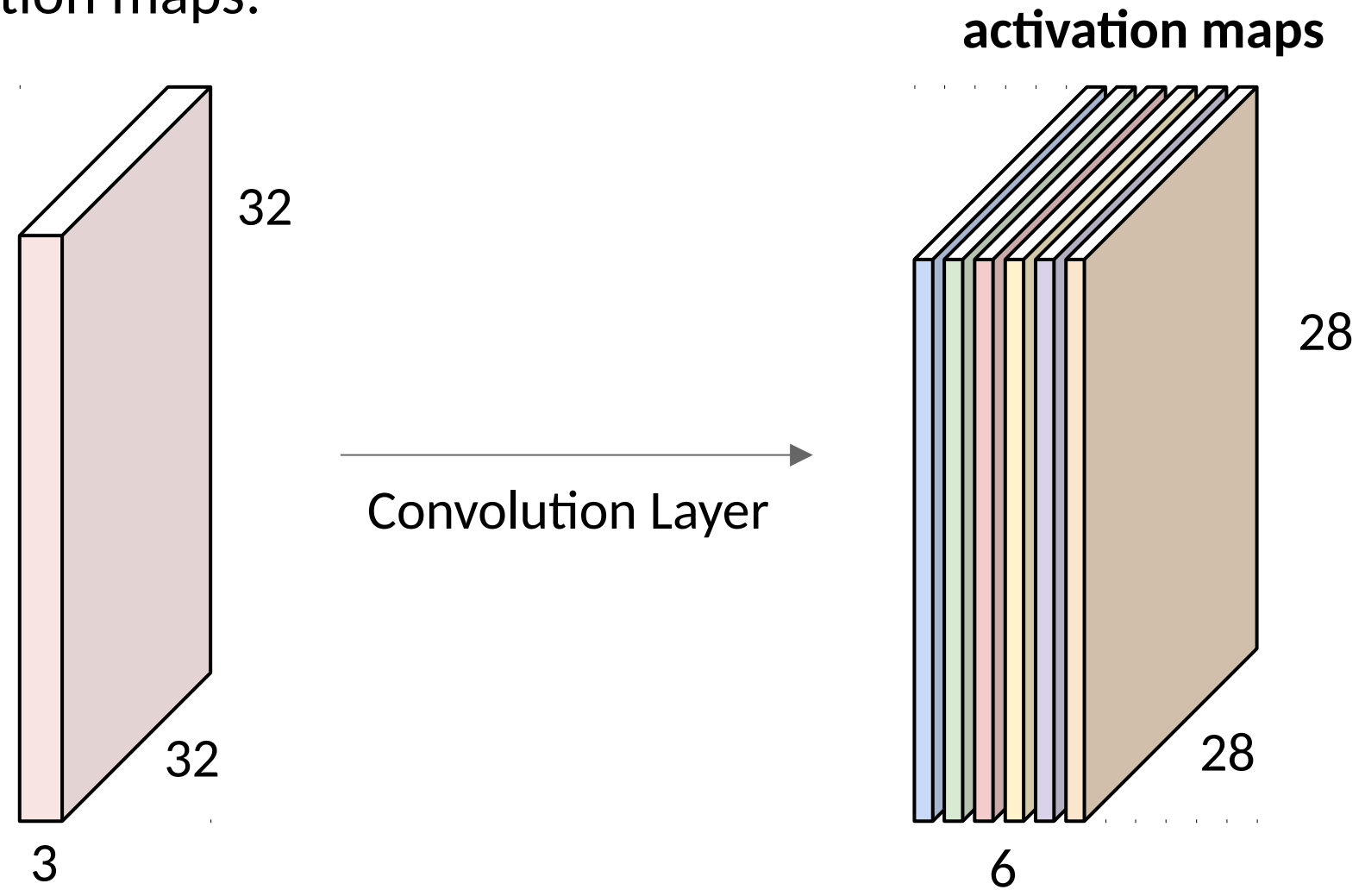


We processed [32x32x3] volume into [28x28x6] volume.

Q: how many parameters would this be if we used a fully connected layer instead?

A: (32*32*3)*(28*28*6) = **14.5M parameters**

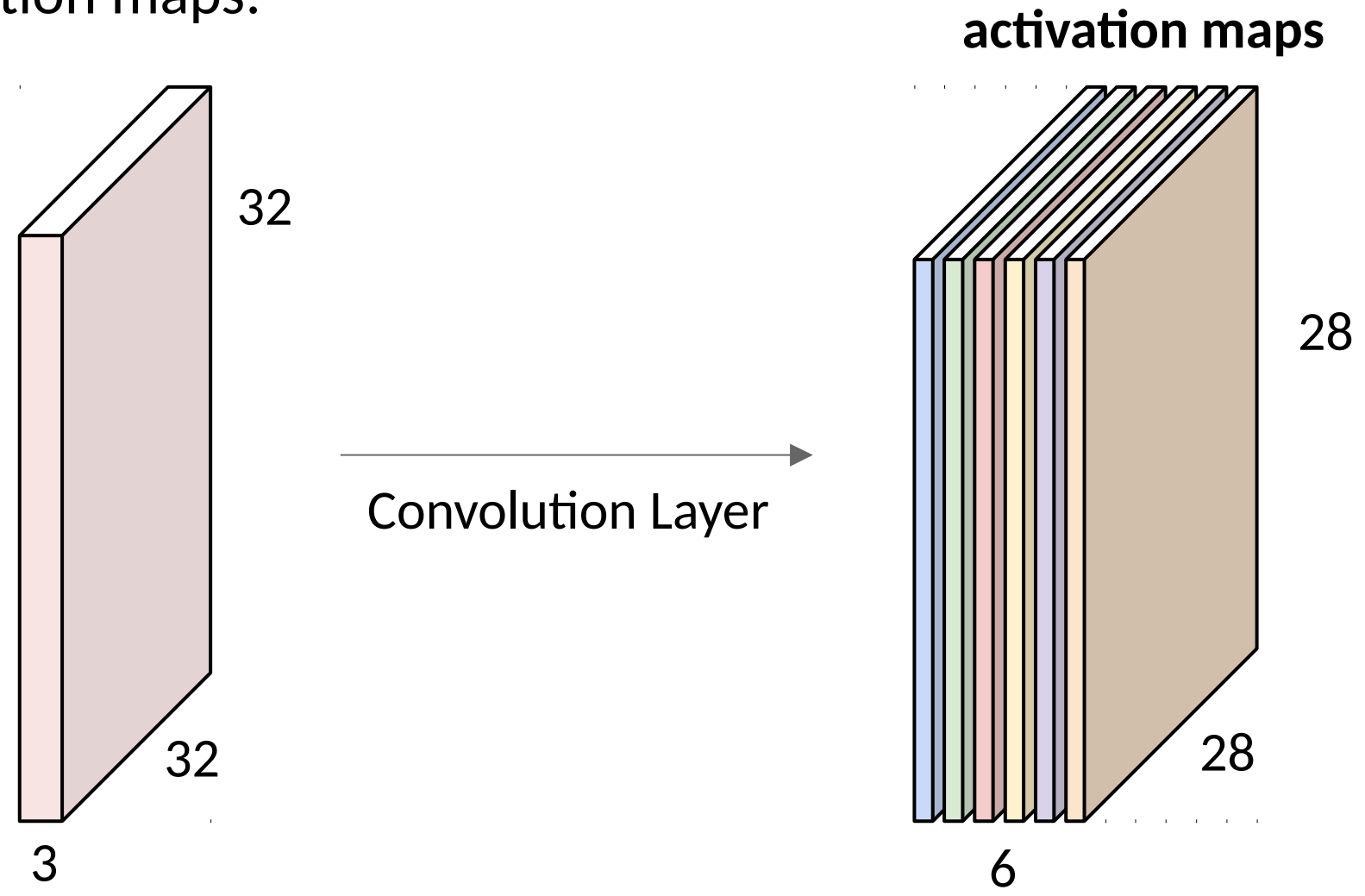
For example, if we had 6 (5x5x3) filters, we'll get 6 separate activation maps:



We processed [32x32x3] volume into [28x28x6] volume.

Q: how many parameters are used instead?

For example, if we had 6 (5x5x3) filters, we'll get 6 separate activation maps:



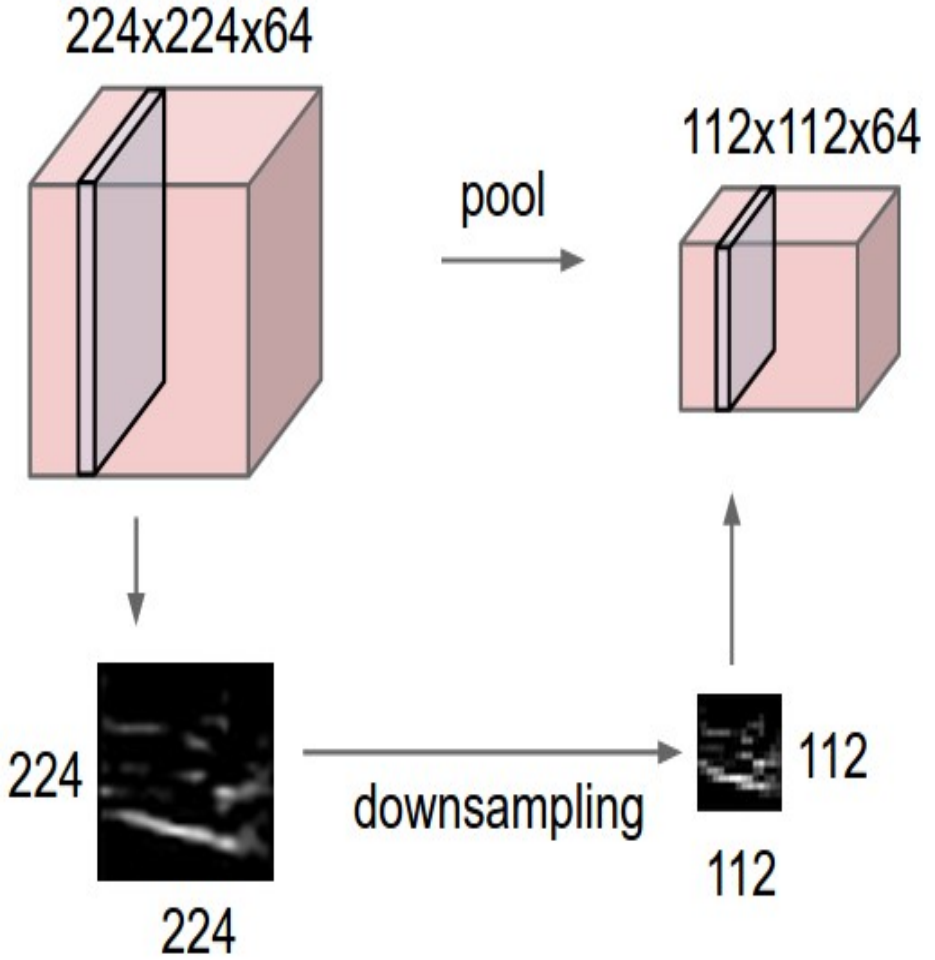
We processed [32x32x3] volume into [28x28x6] volume.

Q: how many parameters are used instead?

A: (5*5*3)*6 = **450 parameters**

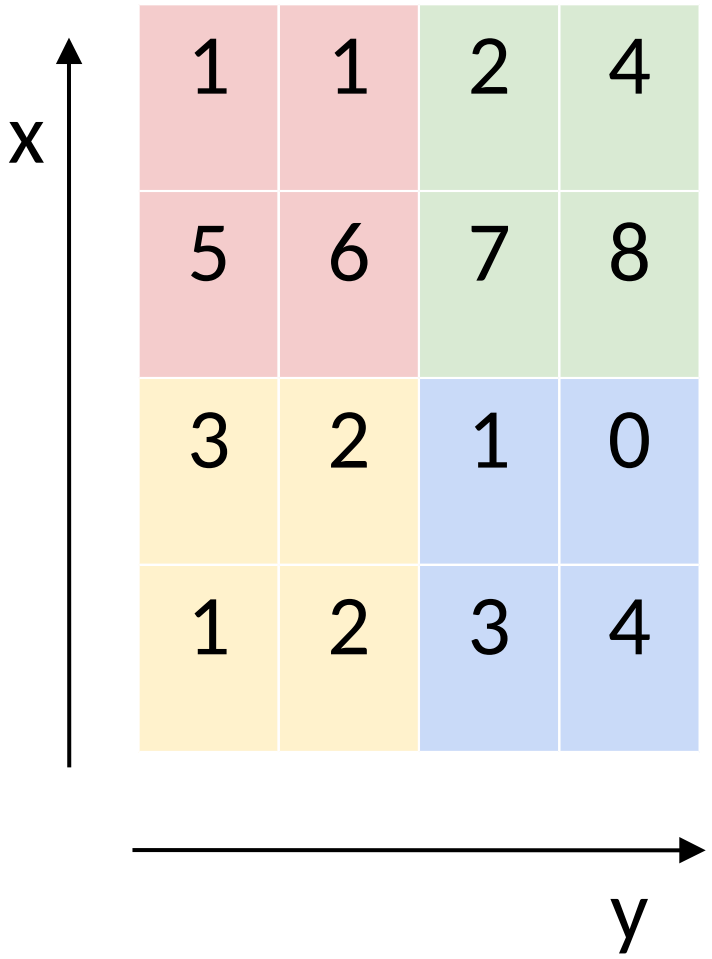
Pooling layer

- makes the representations smaller and more manageable
- operates over each activation map independently to reduce dimensionality



MAX Pooling Layer

Single depth slice

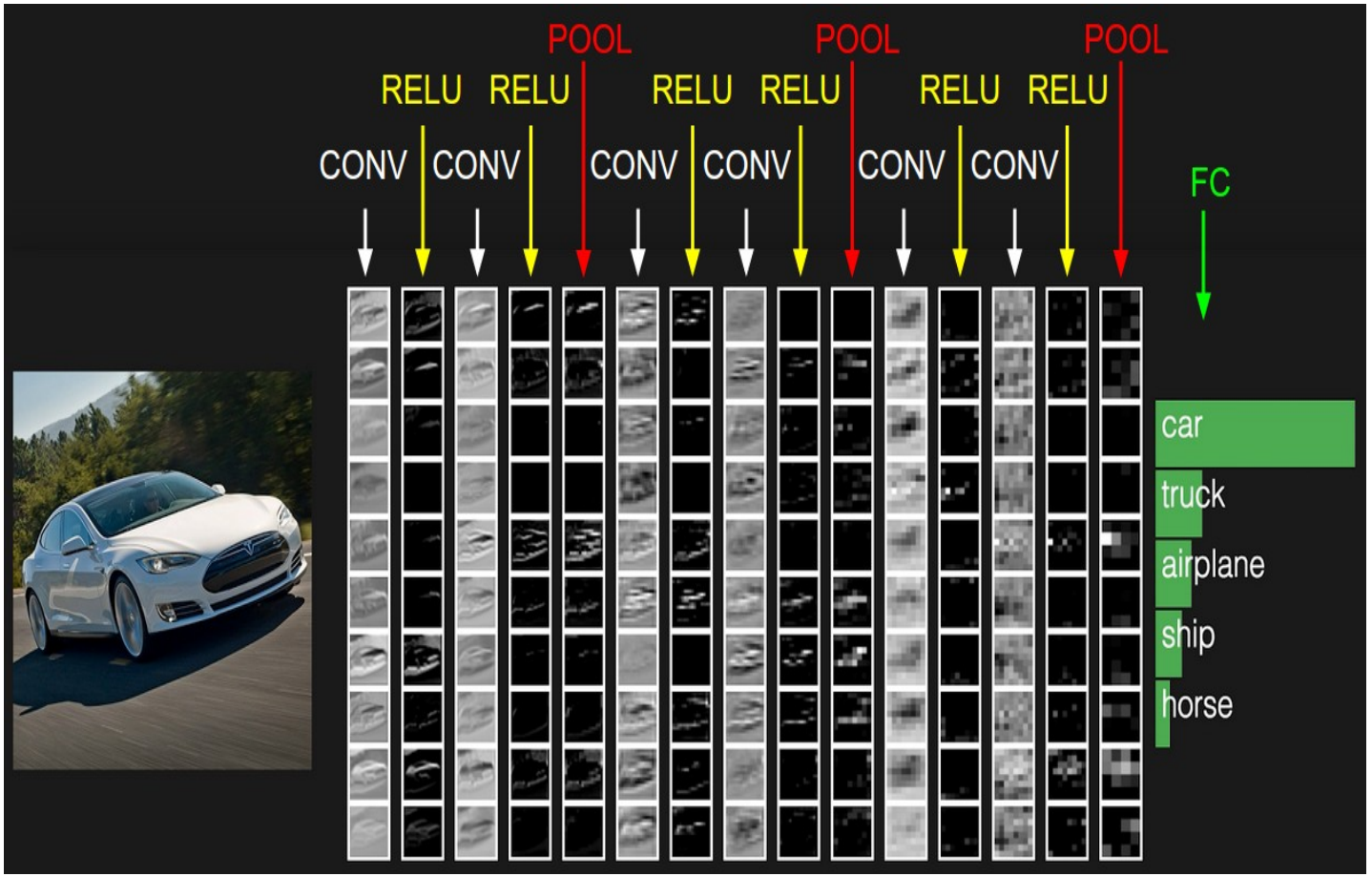


max pool with 2x2 filters and stride 2

6	8
3	4

Fully Connected Layer (FC layer)

- FC Layer is often used after CNNs to go classification or regression.



Examples of CNN architectures

AlexNet (2012): the model that popularised the use of deep convolutional neural networks for image classification tasks.

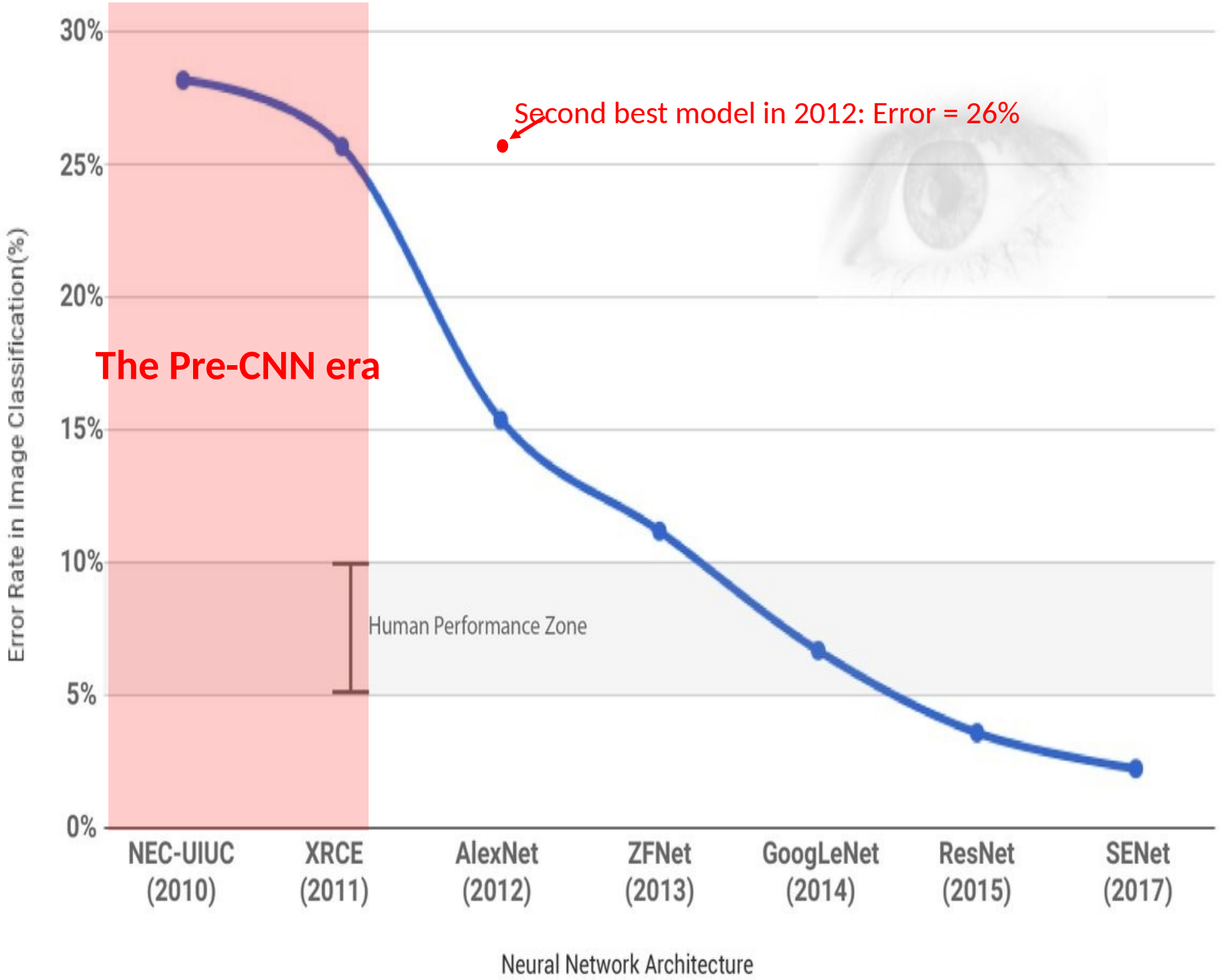
Part of the reason this model won is that the authors developed sophisticated techniques to exploit **GPUs** in model training.
All modern large-scale neural network training happens on GPUs or TPUs

AlexNet (2012). Won the **ILSVRC challenge:** a competition where researchers submit image classifiers and evaluate their accuracy.

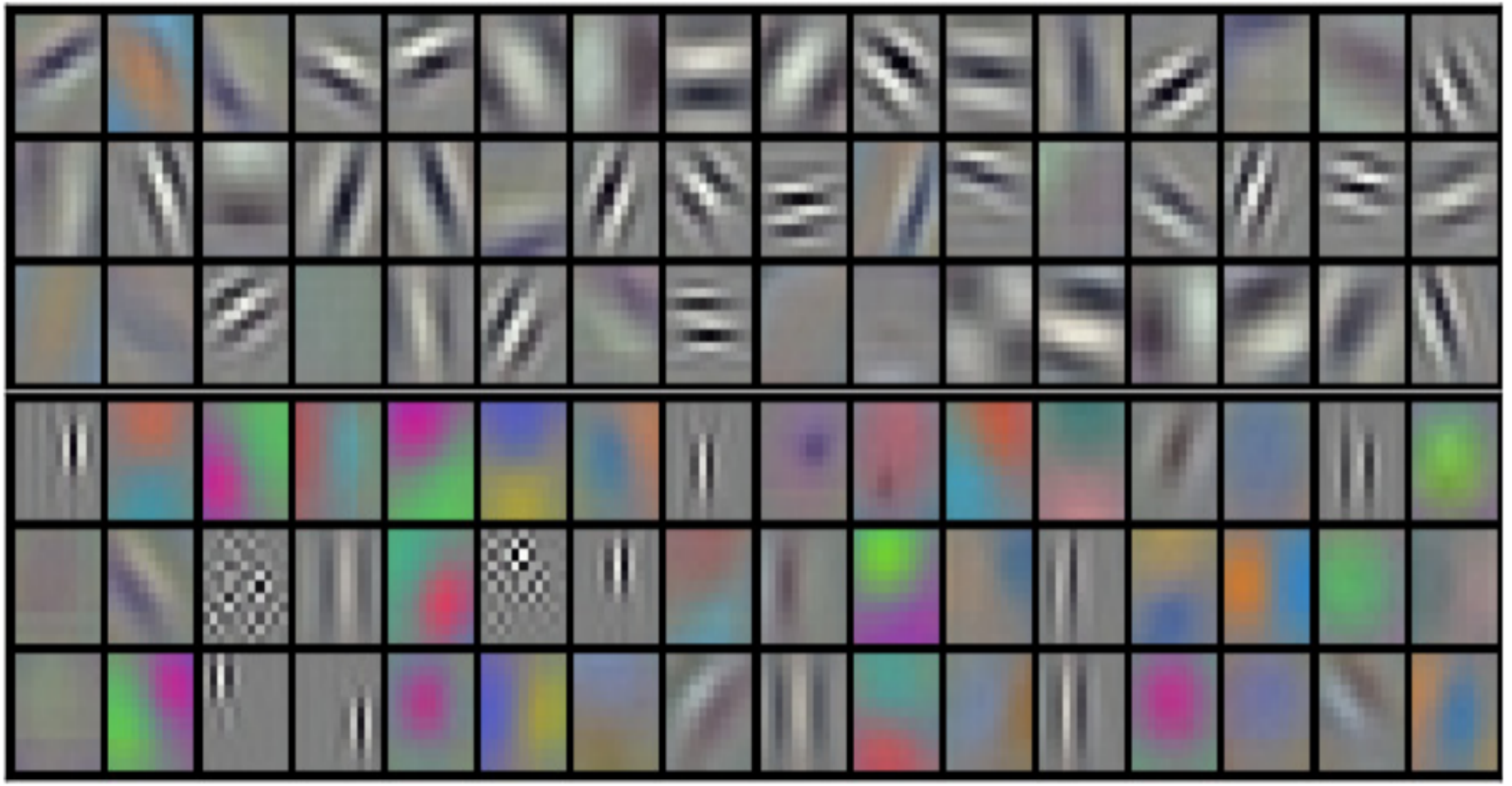
~1 thousand object categories.
~10 million training images

Some classifications are very difficult:

Which of ~120 different breeds of dog is this?


Second best model in 2012: Error = 26%
The Pre-CNN era
Human Performance Zone
Error Rate in Image Classification(%)
30%
25%
20%
15%
10%
5%
0%
NEC-UIUC (2010)
XRCE (2011)
AlexNet (2012)
ZFNet (2013)
GoogLeNet (2014)
ResNet (2015)
SENet (2017)
Neural Network Architecture

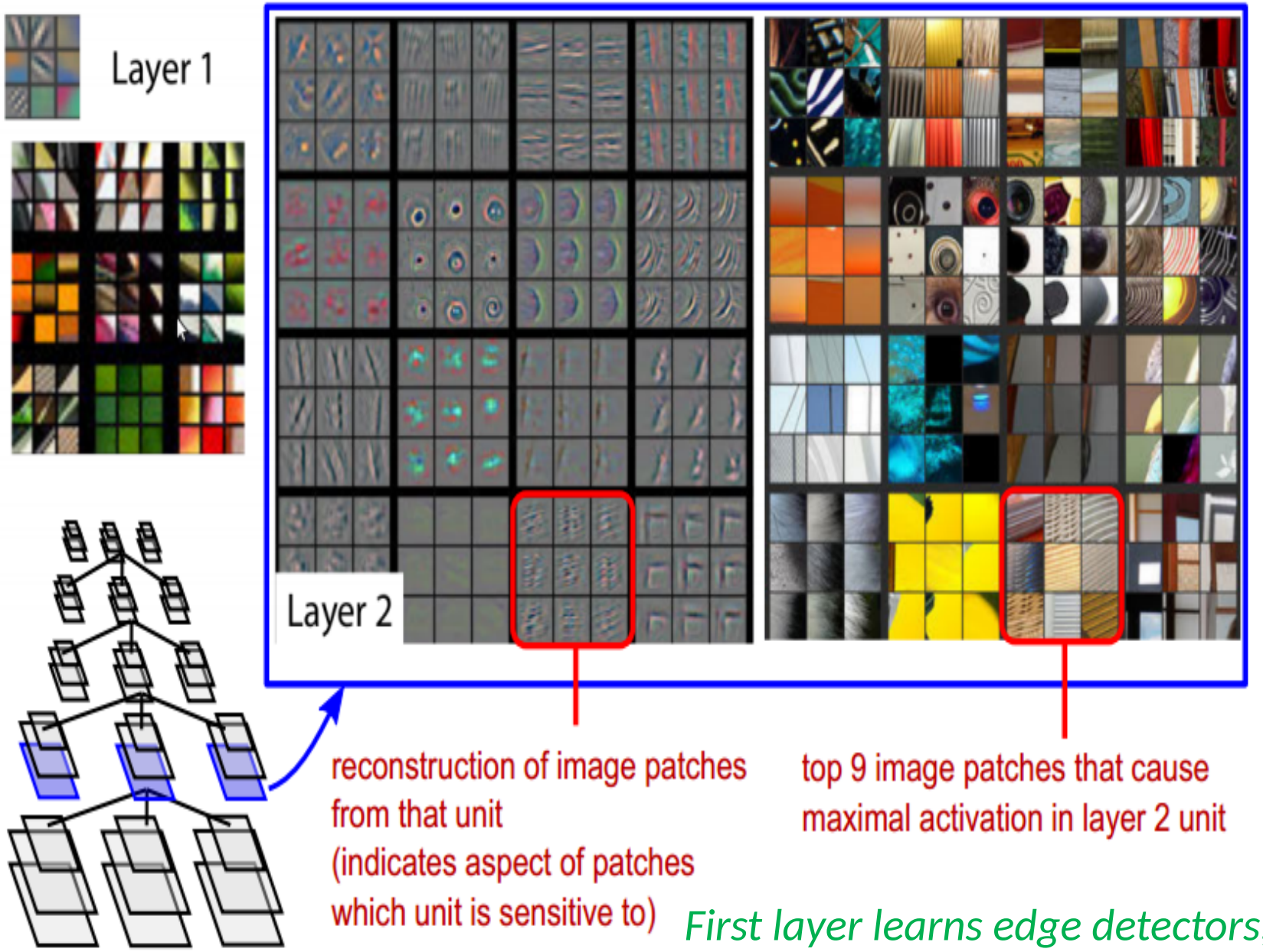
AlexNet: features learned on first convolutional layer



Emergent modularity for GPUs 1 and 2: Combining different orientation features makes more sense than combining orientation and colour information.

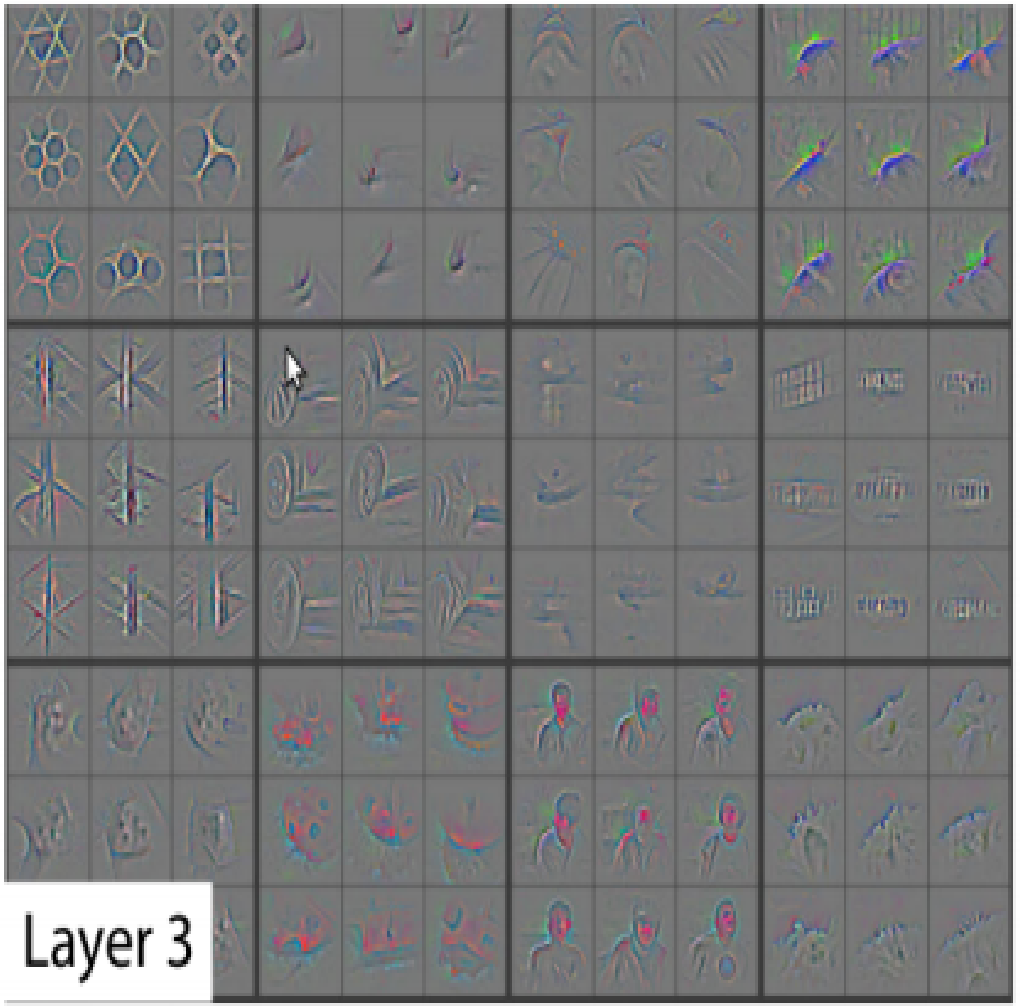
Krizhevsky et al., 2012

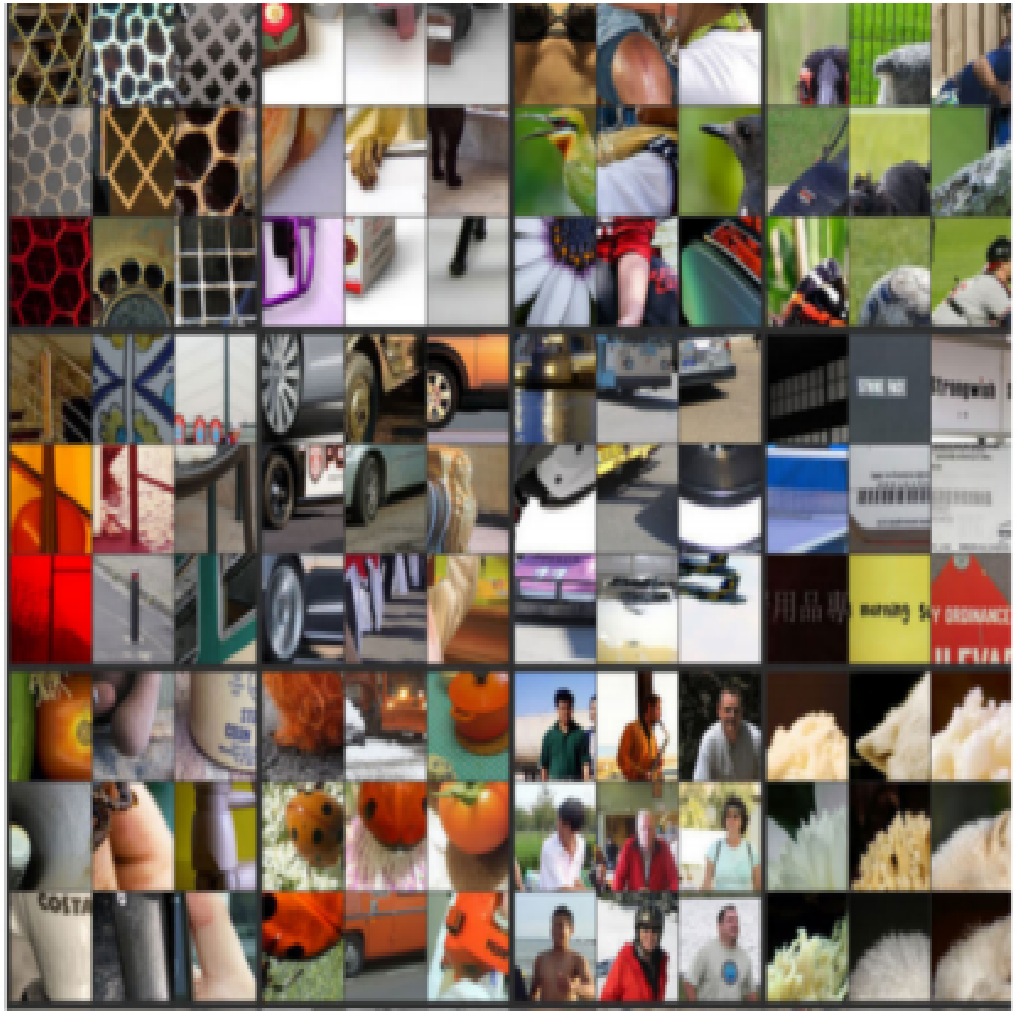
AlexNet: first and second convolutional layers



First layer learns edge detectors, colour gradients.
Subsequent layers learn more complex features
(2nd layer: corners)

AlexNet: convolutional layer 3

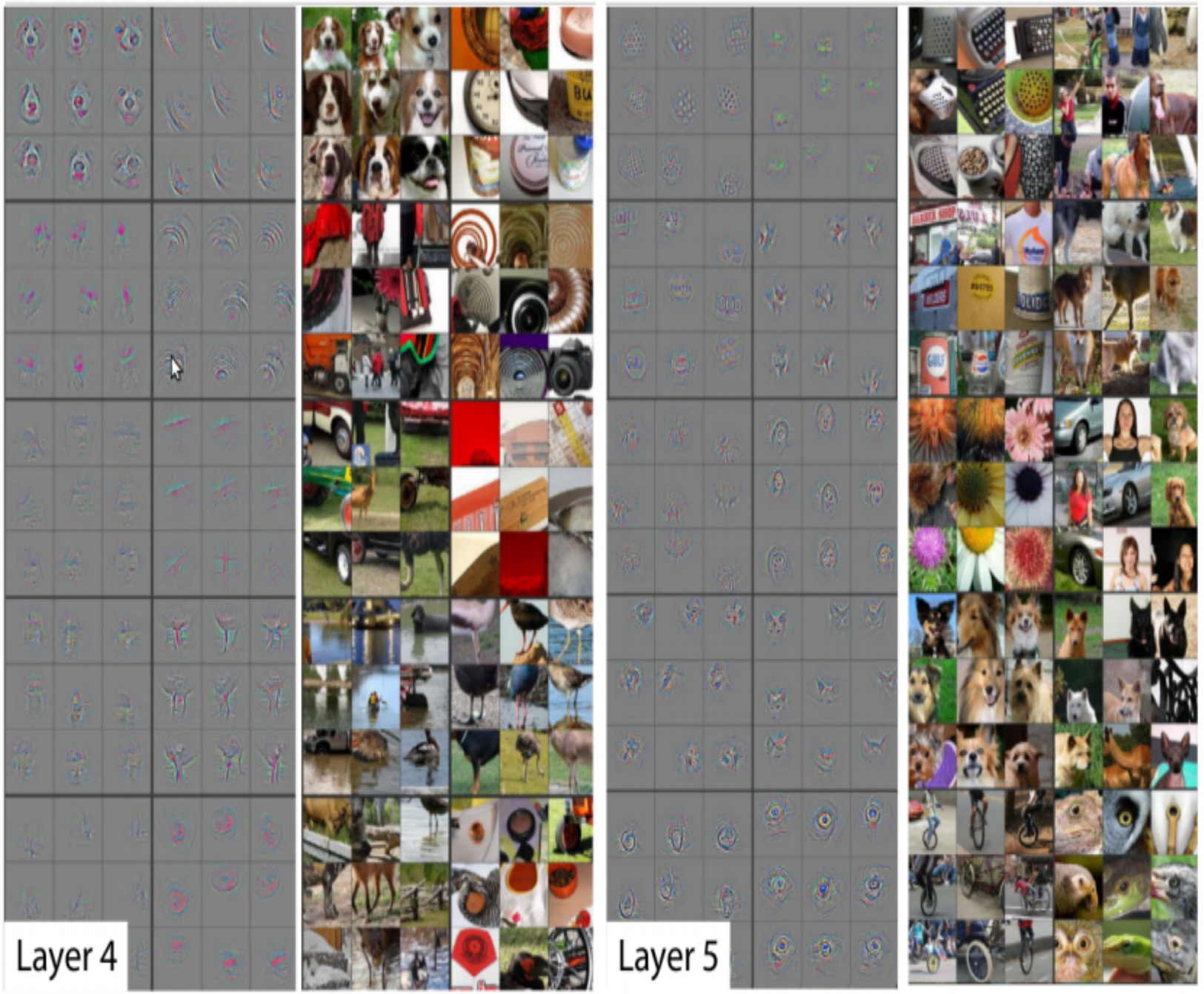




Even more complex features (lattices of holes, round orange objects)

Note that high-level category or semantic information is still mostly absent
Ladybird != tomato != casserole pot

AlexNet: convolutional layers 4 & 5



"Eyes of birds"

"Legs of dogs"

Etc.

CNN: Summary

- Higher level layers show more invariance to:
  - ✔ *Translation*
  - ✔ *Rotation*
  - ✔ *Lighting changes*
- CNN layer is often used with max-pooling layers to downsample/reduce dimensionality

- CNN output can be flatten, then fed into fully connected layers