greatlearning
Learning for Life

Deep Learning (for Computer Vision)

Arjun Jain

Visualizing and Understanding ConvNets

Understanding ConvNets

- Visualize patches that maximally activate neurons
- Visualize the weights
- Visualize the representation space
- Occlusion experiments

Visualize patches that maximally activate neurons

one-stream AlexNet

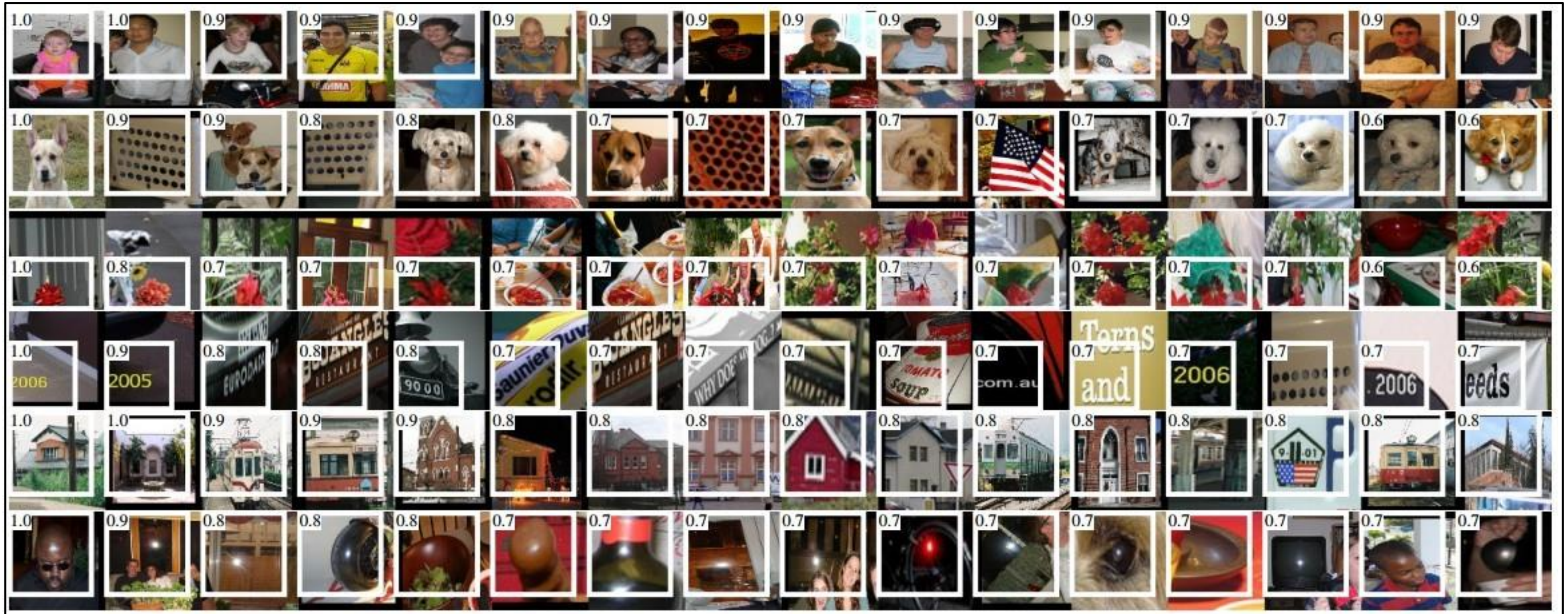
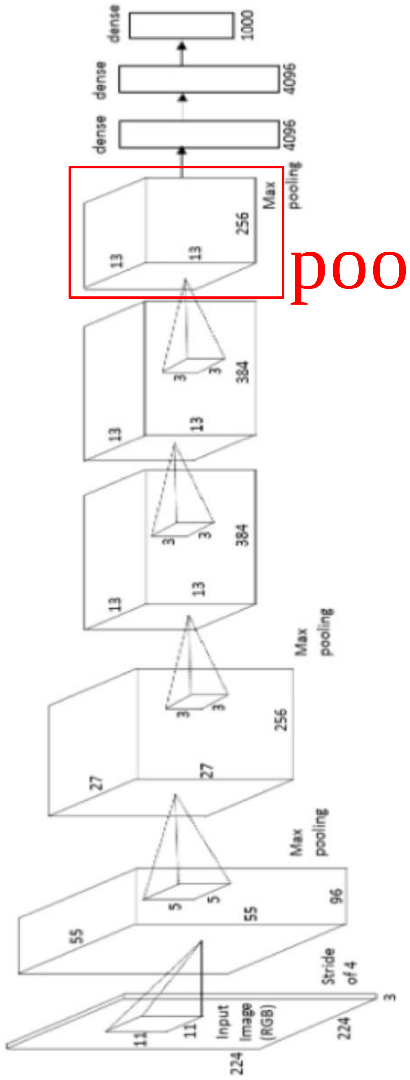


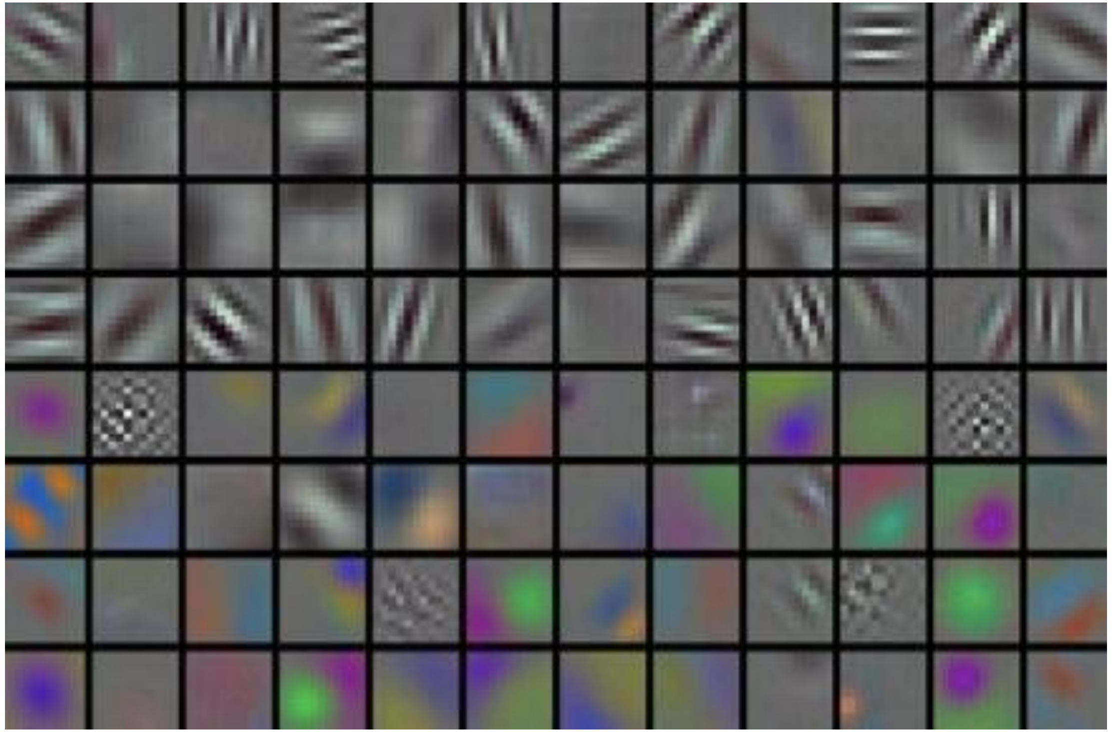
Figure 4: Top regions for six pool$_5$ units. Receptive fields and activation values are drawn in white. Some units are aligned to concepts, such as people (row 1) or text (4). Other units capture texture and material properties, such as dot arrays (2) and specular reflections (6).

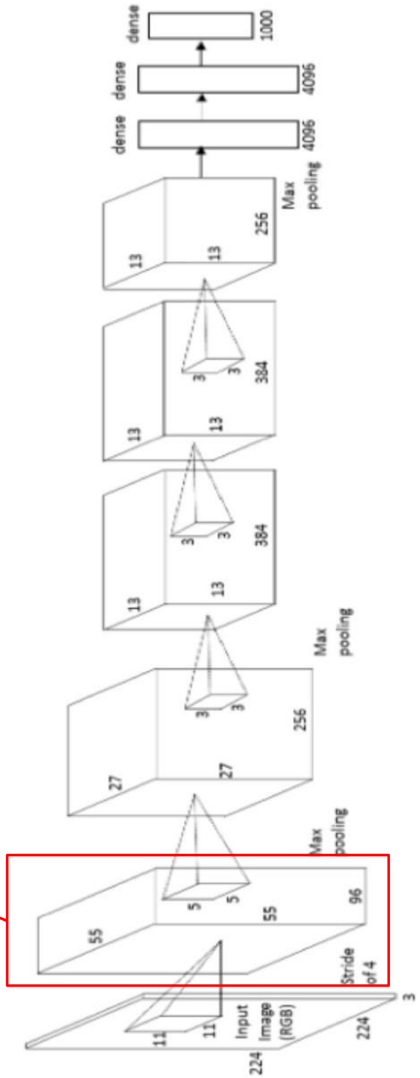


Source: Rich feature hierarchies for accurate object detection and semantic segmentation [Girshick, Donahue, Darrell, Malik]

Visualize the filters/kernels(raw weights)

one-stream AlexNet





conv1

only interpretable on the first layer :(

Source: Rich feature hierarchies for accurate object detection and semantic segmentation [Girshick, Donahue, Darrell, Malik]

Visualize the filters/kernels(raw weights)

layer 1 weights

layer 2 weights

layer 3 weights

one can still do it for higher layers, but it's just not that interesting

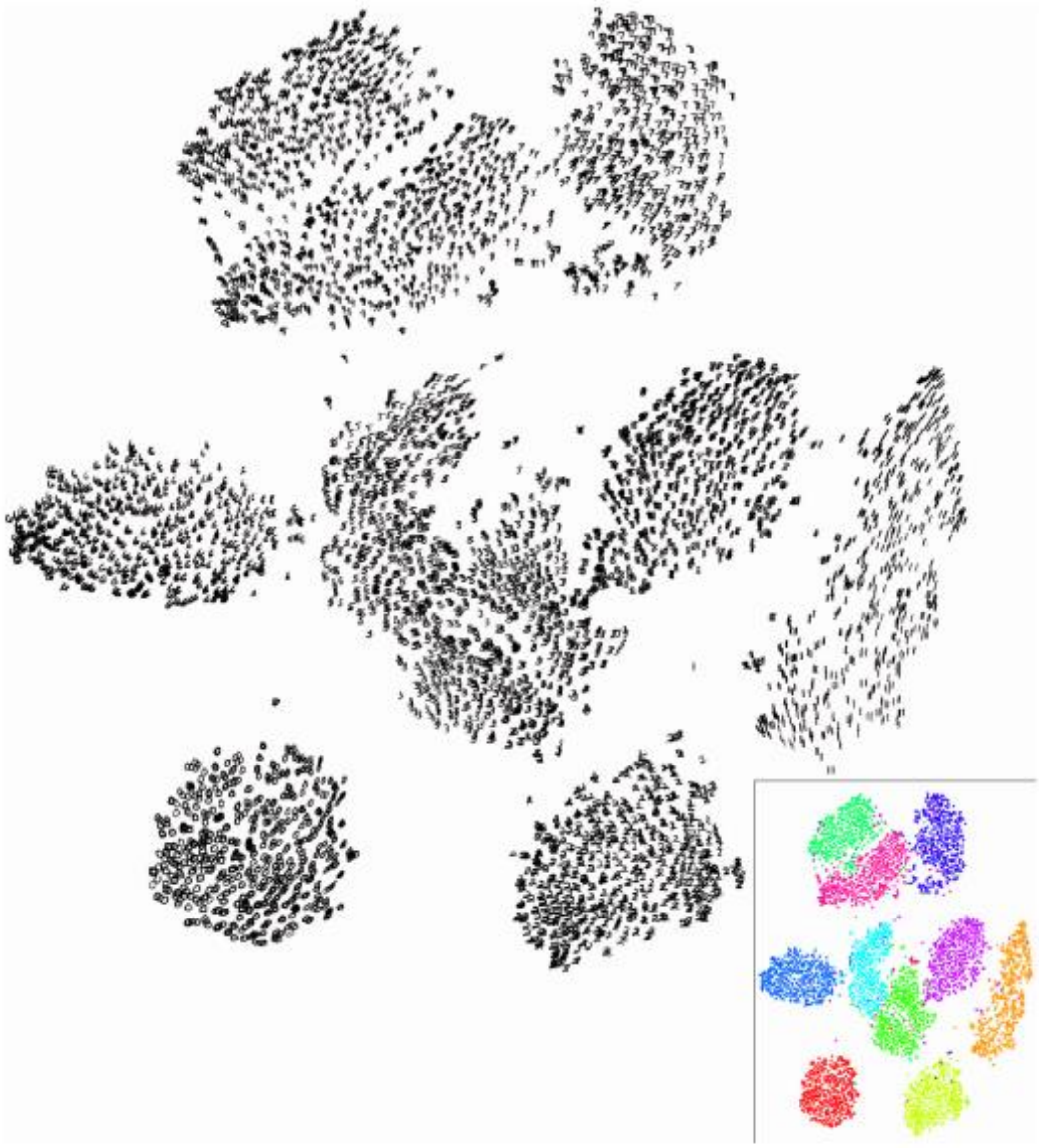
Source: ConvNetJS CIFAR-10 demo

Visualizing the representation

Embed high-dimensional points so that locally, pairwise distances are conserved

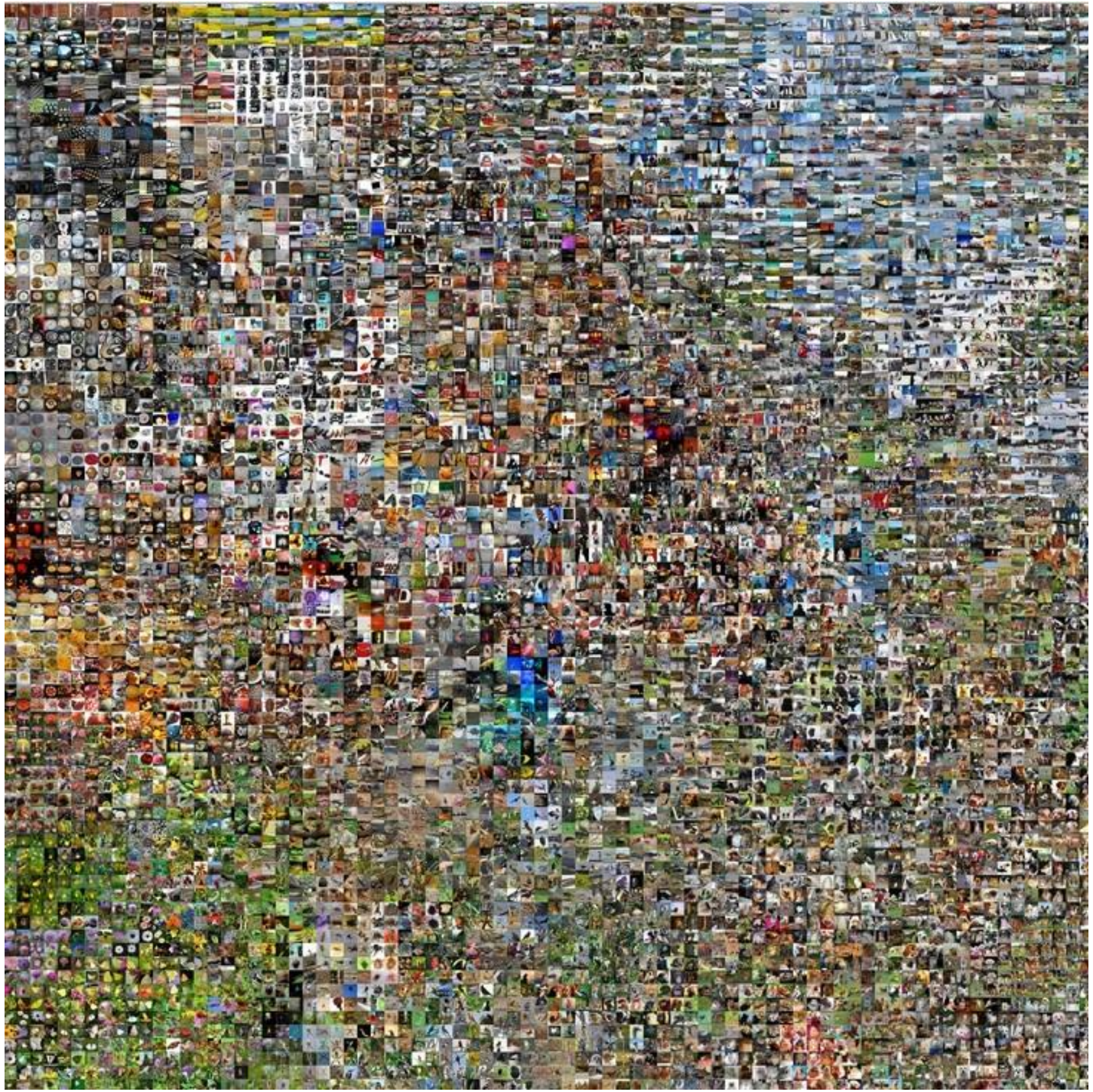
i.e. similar things end up in similar places and dissimilar things end up wherever

Right: Example embedding of MNIST digits (0-9) in 2D



Source: Visualizing data using t-SNE [van der Maaten &Hinton]

Visualizing the representation



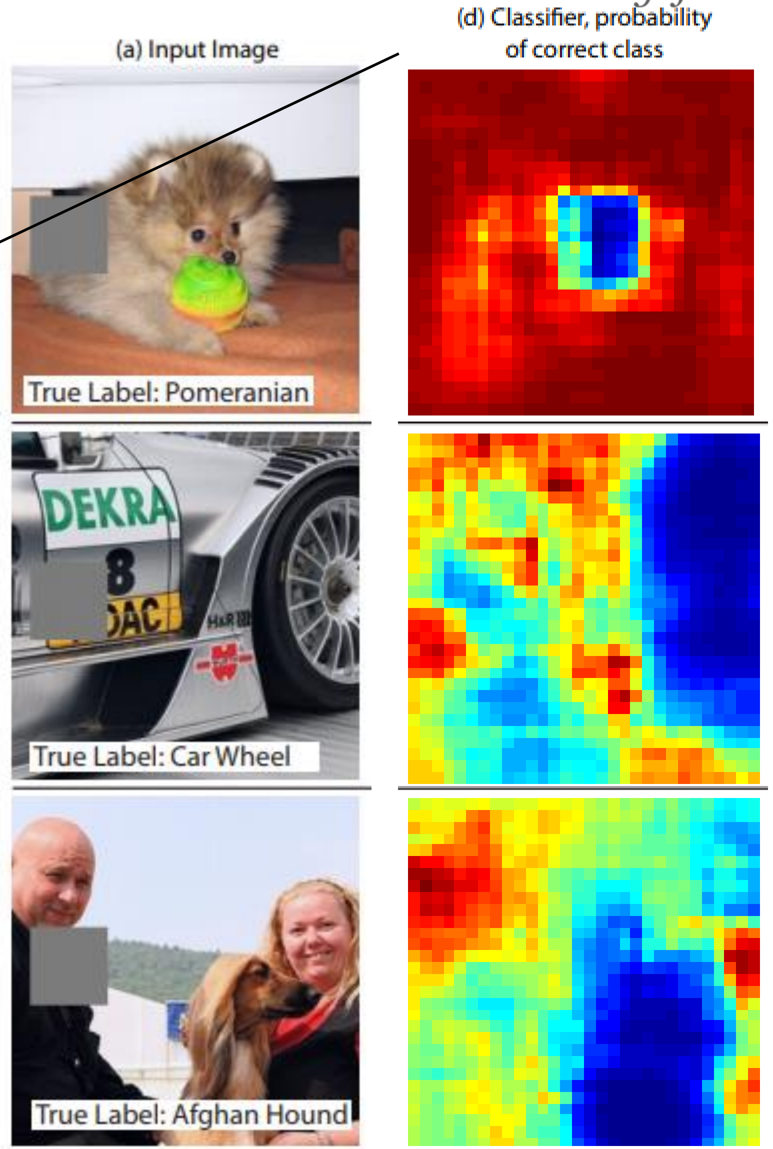
Two images are placed nearby

if their CNN codes are close.

Source: https://cs.stanford.edu/people/karpathy/cnnembed/

Occlusion experiments

(as a function of the position of the square of zeros in the original image)



Source: Visualizing and Understanding Convolutional Networks [Zeiler & Fergus 2013]

Universal approximation theorem

From Wikipedia, the free encyclopaedia

In the mathematical theory of artificial neural networks, the **universal approximation theorem** states that a feed-forward network with a single hidden layer containing a finite number of neurons can approximate continuous functions on compact subsets of $\mathbf{R}^n$, under mild assumptions on the activation function. The theorem thus states that simple neural networks can *represent* a wide variety of interesting functions when given appropriate parameters.

Universal approximation theorem

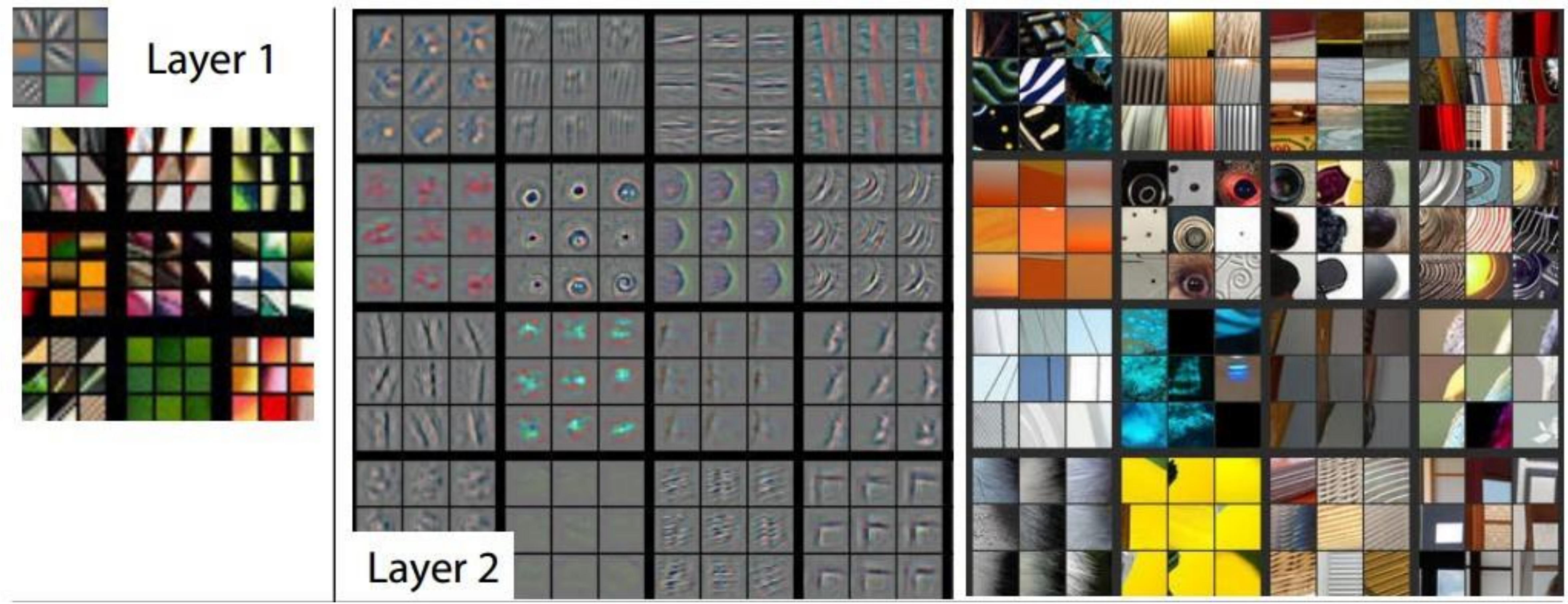
From Wikipedia, the free encyclopaedia

In the mathematical theory of artificial neural networks, the **universal approximation theorem** states that a feed-forward network with a single hidden layer containing a finite number of neurons can approximate continuous functions on compact subsets of $\mathbf{R}^n$, under mild assumptions on the activation function. The theorem thus states that simple neural networks can *represent* a wide variety of interesting functions when given appropriate parameters.

Then why so deep?

Visualizing and Understanding Convolutional Networks

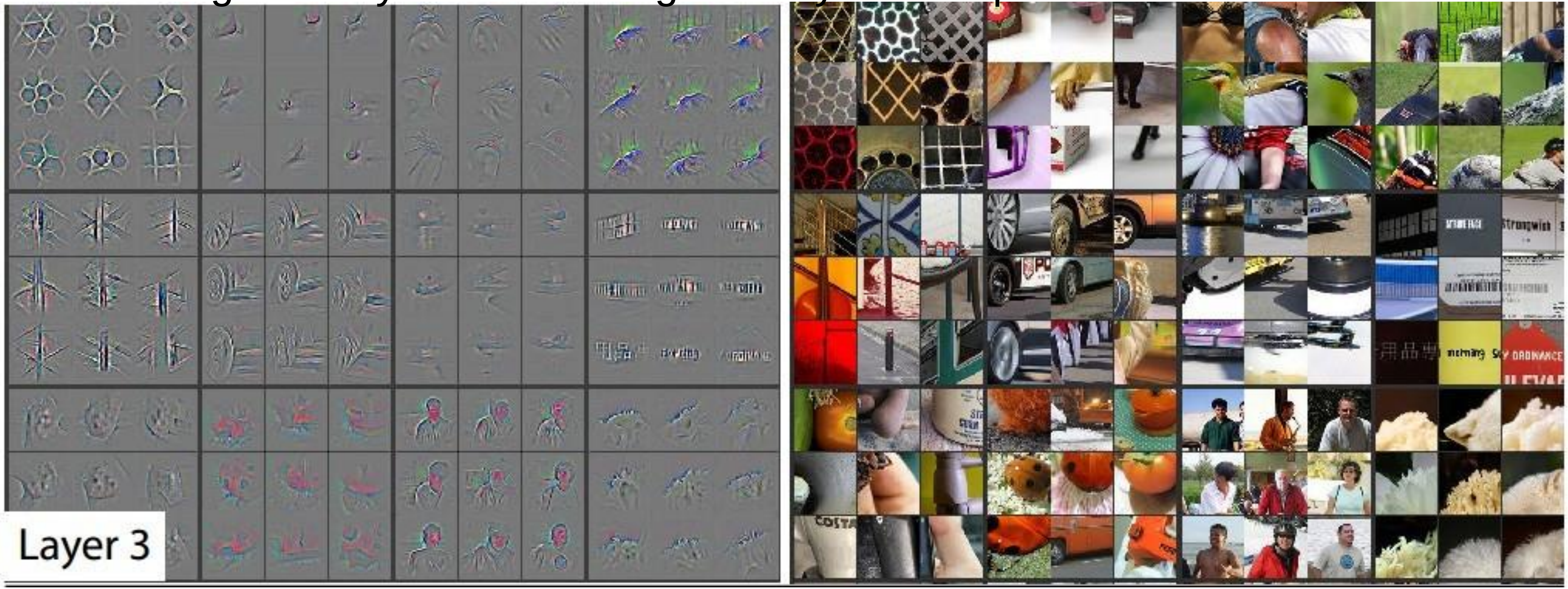
Visualizing arbitrary neurons along the way to the top...



Source: Visualizing and Understanding Convolutional Networks [Zeiler & Fergus 2013]

Visualizing and Understanding Convolutional Networks

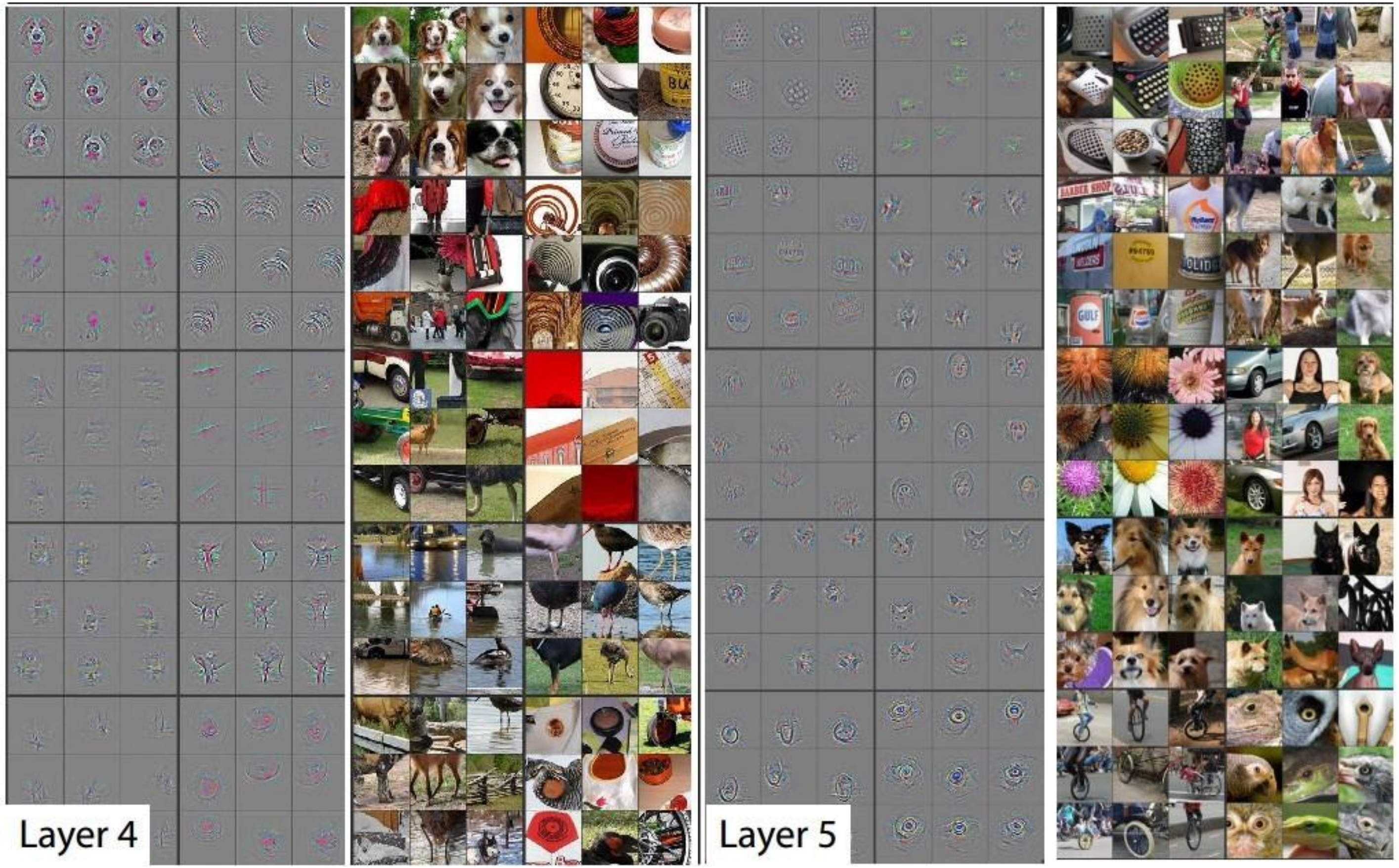
Visualizing arbitrary neurons along the way to the top...



Source: Visualizing and Understanding Convolutional Networks [Zeiler & Fergus 2013]

Visualizing and Understanding Convolutional Networks

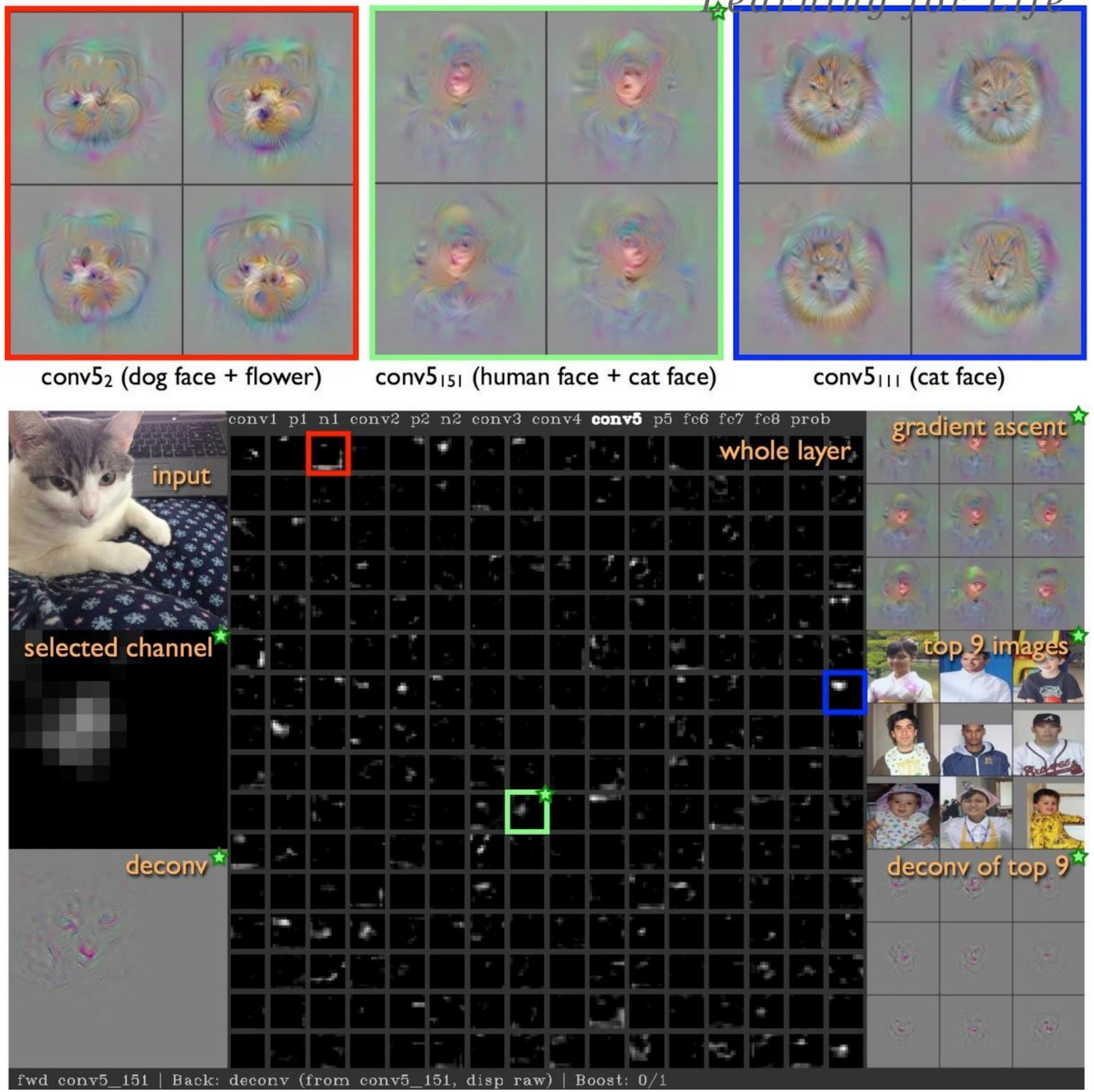
Visualizing arbitrary neurons along the way to the top...



Source: Visualizing and Understanding Convolutional Networks [Zeiler & Fergus 2013]

Deep Visualization Toolbox

http://yosinski.com/deepvis



Source: Understanding Neural Networks Through Deep Visualization [Yosinski, Clune et.al.] [(http://yosinski.com/deepvis)](http://yosinski.com/deepvis)

Thank you!