بنام خداوند جان و خرد

Interpretability
in (Convolutional) Neural Networks

Mohammad Taher Pilehvar

Machine Learning 01
https://teias-courses.github.io/ml01/

TeIAS | Tehran Institute for Advanced Studies

Image source

Many of the slides are borrowed from Stanfords's CS230 course and from the distill.pub website

Interpretation (explanation)

The process of **demystifying** the black box machine learning models

to improve **transparency** and interpretability

to make them more **trustworthy** and **reliable**

Why is interpretation important?

Accuracy is not enough!

- A model can achieve high accuracy by memorizing the unimportant features or patterns in your data set.
  - Robustness

Why is interpretation important?

- **Data scientists**: get insights on the model.
  - Analyze strengths and weaknesses, to improve the model
  - Communicate insights to the target audience
  - Avoid surprises, enhance robustness

- **End users**: how a models makes a certain prediction?
  - Are they being treated fairly?
  - Whether they can trust it (e.g., online shopping)?

- **Regulators**: are systems fair and transparent?
  - Protect customers

By Ilknur Kaynar Kabul, SAS.

Transparent, trustworthy, and explainable

- Transparent:
  - The system can explain how it works and/or why it gives certain predictions
- Trustworthy
  - The system can handle different scenarios in the real world without continuous control.
- Explainable
  - The system can convey useful information about its inner workings, for the patterns that it learns and for the results that it gives.

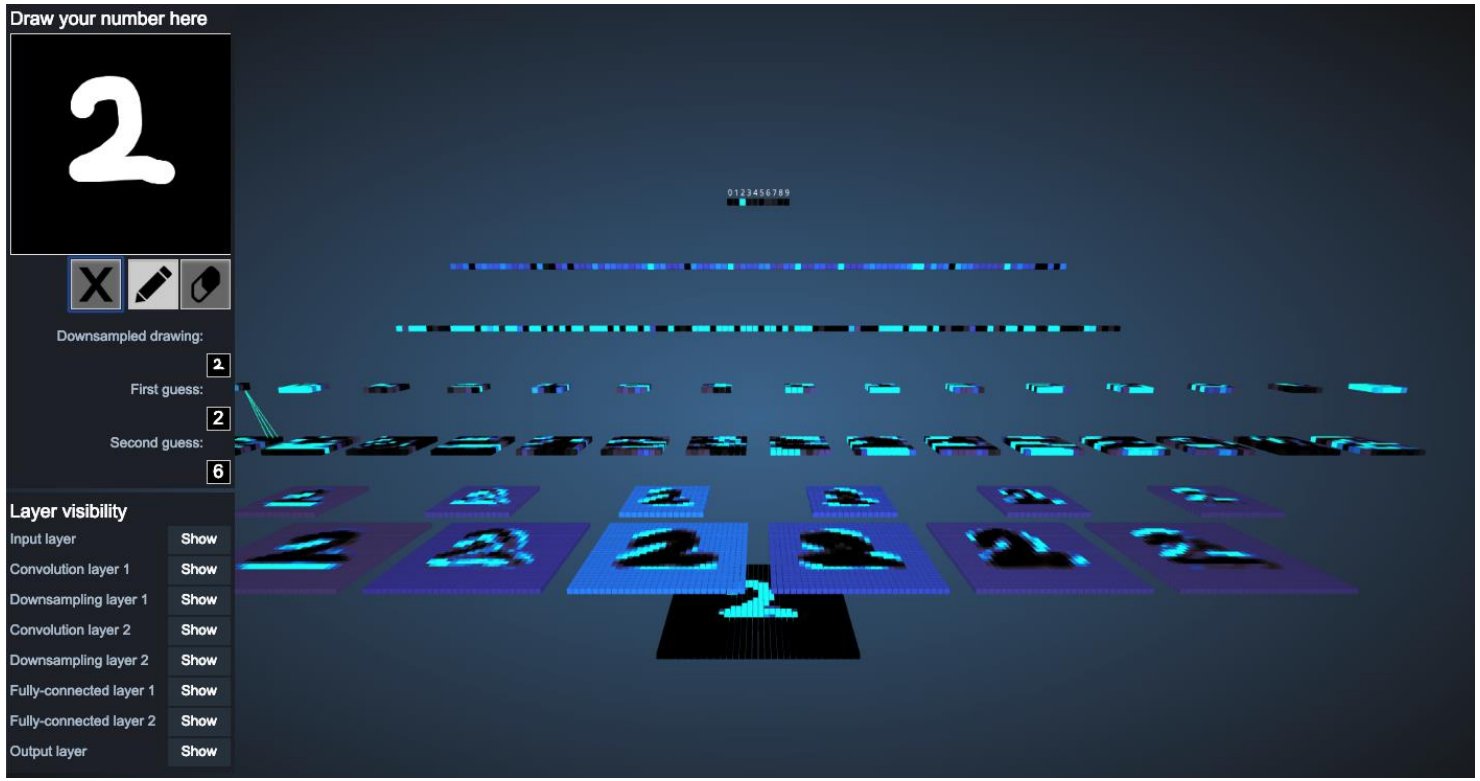
By Ilknur Kaynar Kabul, SAS.

Visualization in CNNs

- A way of interpreting the outputs of neural networks
- Sheds light on the reasons behind their decision making

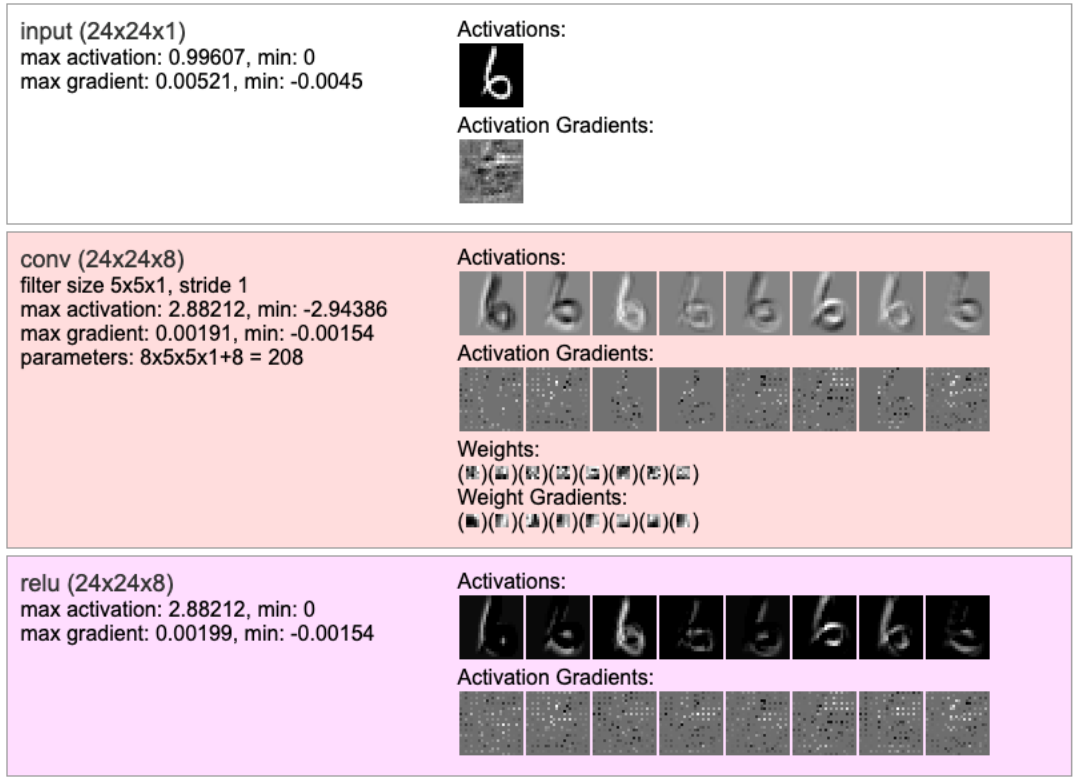
Visualizing intermediate activations

- http://scs.ryerson.ca/~aharley/vis/conv/flat.html

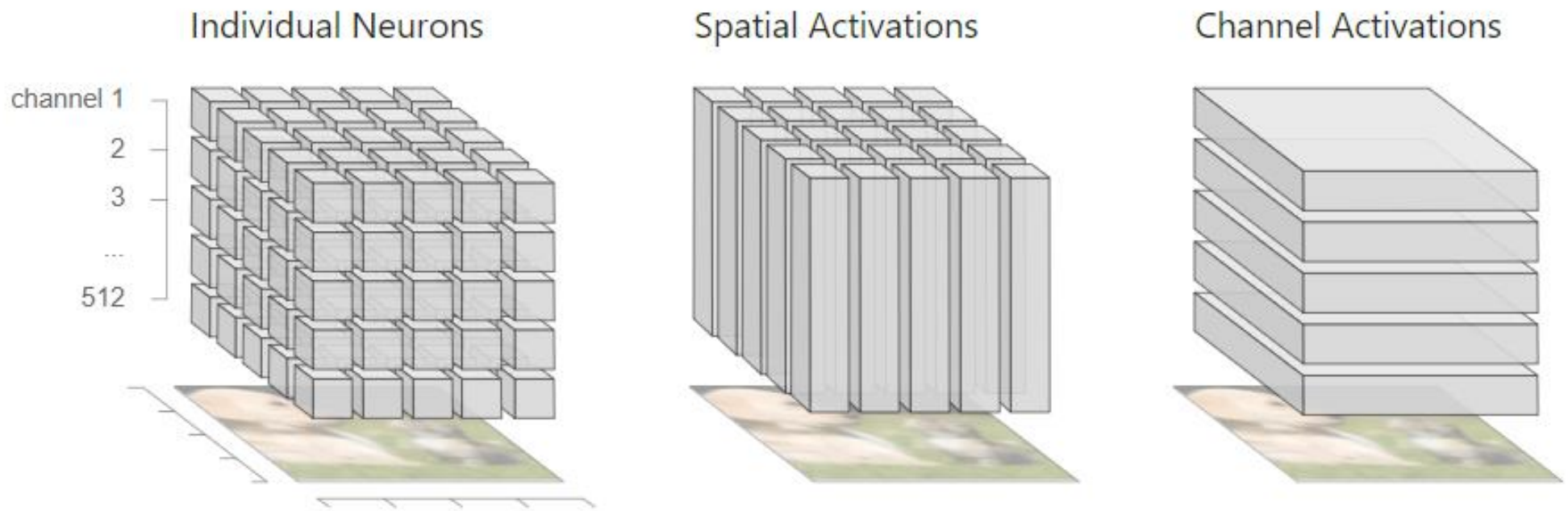


Visualizing intermediate activations

- https://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html



A (hidden) CNN layer

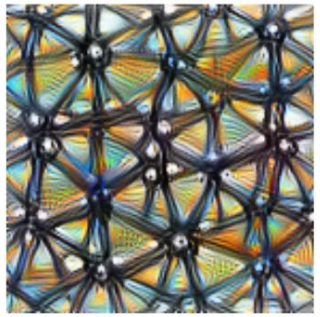


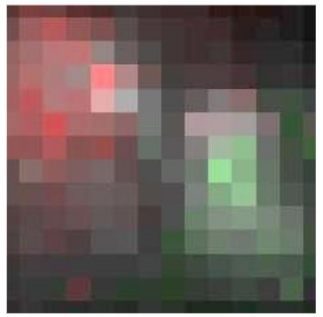
The cube of activations that a neural network for computer vision develops at each hidden layer. Different slices of the cube allow us to target the activations of individual neurons, spatial positions, or channels.

Visualization

Different ways to visualize or interpret NN representations.

- Retrieve from real images
- Feature visualization
- Attribution
- Dimensionality reduction
- Deconvolution




Visualization by Retrieving from a dataset

Girshick et al (2014): Rich feature hierarchies for accurate object detection and semantic segmentation

Deep Visualization Toolbox

yosinski.com/deepvis

#deepvis

Jason Yosinski

Jeff Clune

Anh Nguyen

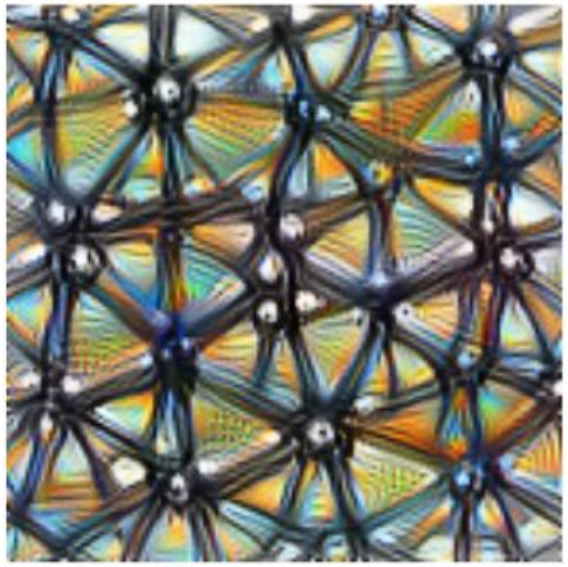
Thomas Fuchs

Hod Lipson

Cornell University

University of Wyoming

Jet Propulsion Laboratory
California Institute of Technology

Feature Visualization by Optimization

Feature Visualization

How neural networks build up their understanding of images

Edges (layer conv2d0) **Textures** (layer mixed3a) **Patterns** (layer mixed4a) **Parts** (layers mixed4b & mixed4c) **Objects** (layers mixed4d & mixed4e)

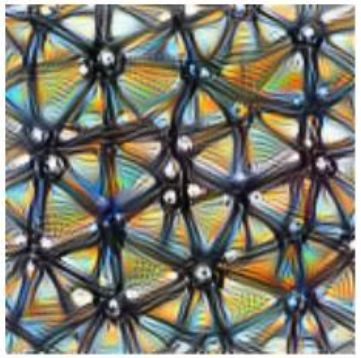
https://distill.pub/2017/feature-visualization/

Feature visualization by optimization

Different **optimization objectives** show what different parts of a network are looking for.

`n` layer index
`x,y` spatial position
`z` channel index
`k` class index

Neuron
$\text{layer}_n[x,y,z]$



Channel
$\text{layer}_n[:,:,z]$

Layer/DeepDream
$\text{layer}_n[:,:,:]^2$

Class Logits
`pre_softmax[k]`

Class Probability
`softmax[k]`

distill.pub

Feature visualization by optimization

- Repeat:
  - Forward propagate image x
  - Compute the objective L
  - Backpropagate to get dL/dx
  - Update x's pixels with gradient ascent

$$L = s_{dog}(x) - \lambda \|x\|_2^2$$

$$x = x + \alpha \frac{\partial L}{\partial x}$$

Class Logits

`pre_softmax[k]`

Feature visualization by optimization

Brown bear

Pug

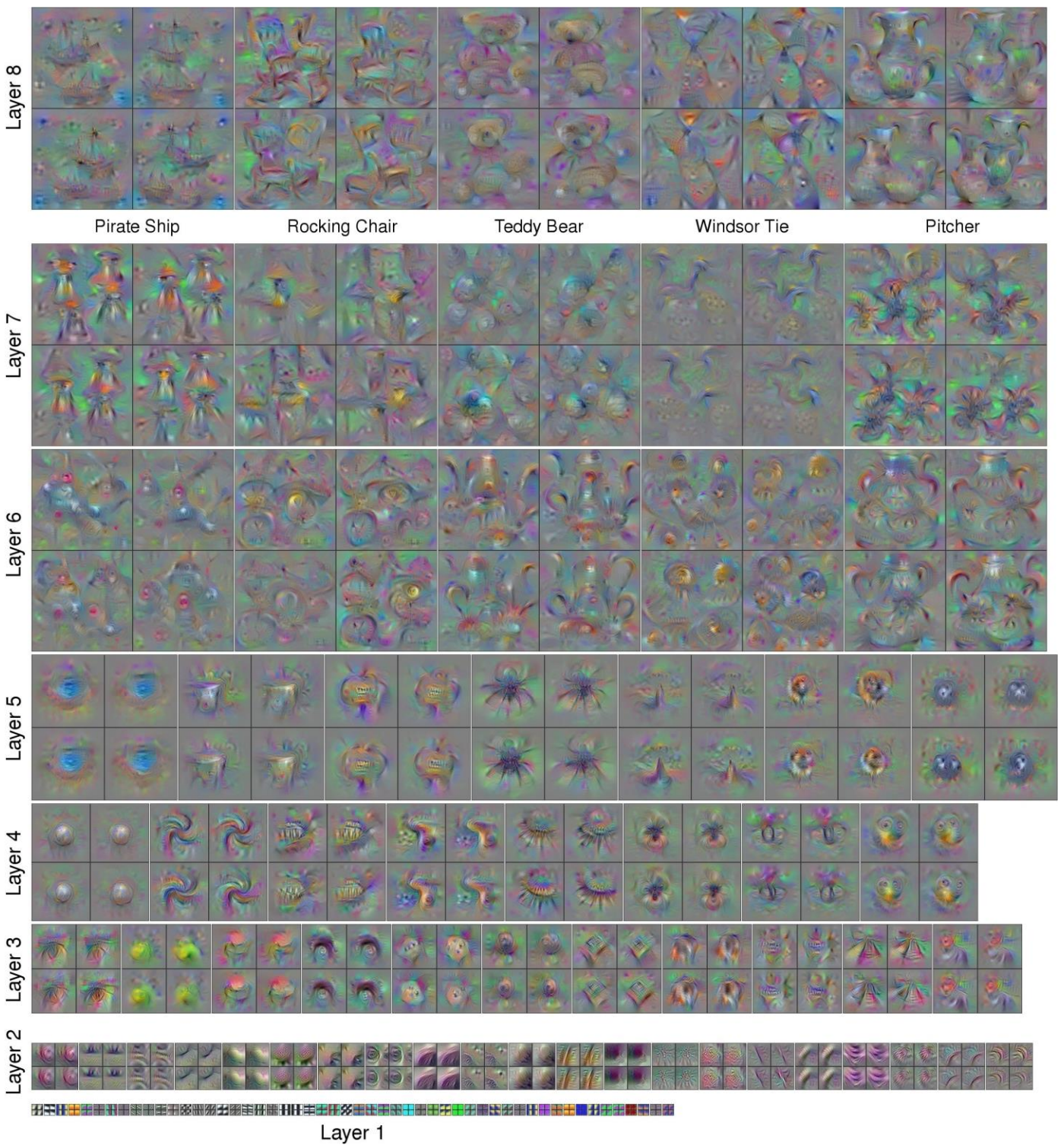
Saxophone

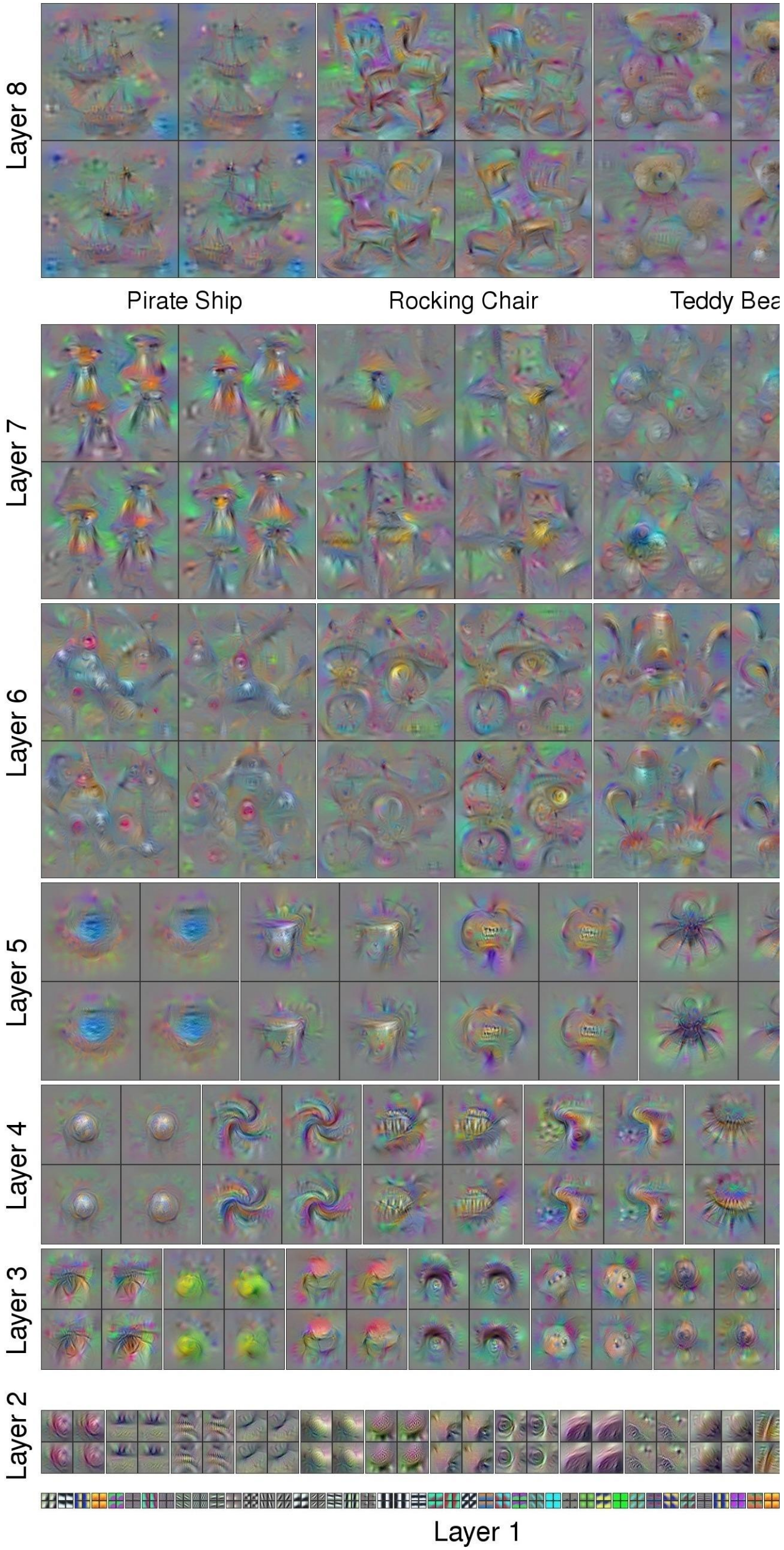
Source: [Visualizing GoogLeNet Classes]

Feature visualization by optimization

Source: Visualizing GoogLeNet Classes

https://yosinski.com/deepvis





https://yosinski.com/deepvis

Why visualize by optimization?

Dataset Examples show us what neurons respond to in practice

Optimization isolates the causes of behavior from mere correlations. A neuron may not be detecting what you initially thought.

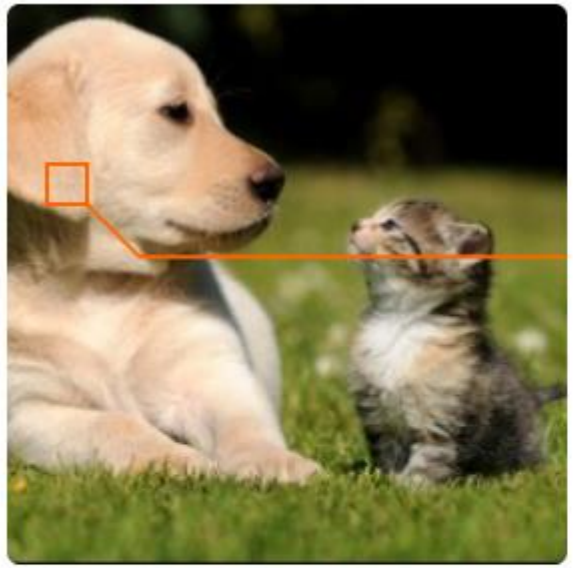
Baseball—or stripes?
mixed4a, Unit 6

Animal faces—or snouts?
mixed4a, Unit 240

Clouds—or fluffiness?
mixed4a, Unit 453

Buildings—or sky?
mixed4a, Unit 492

Semantic dictionary

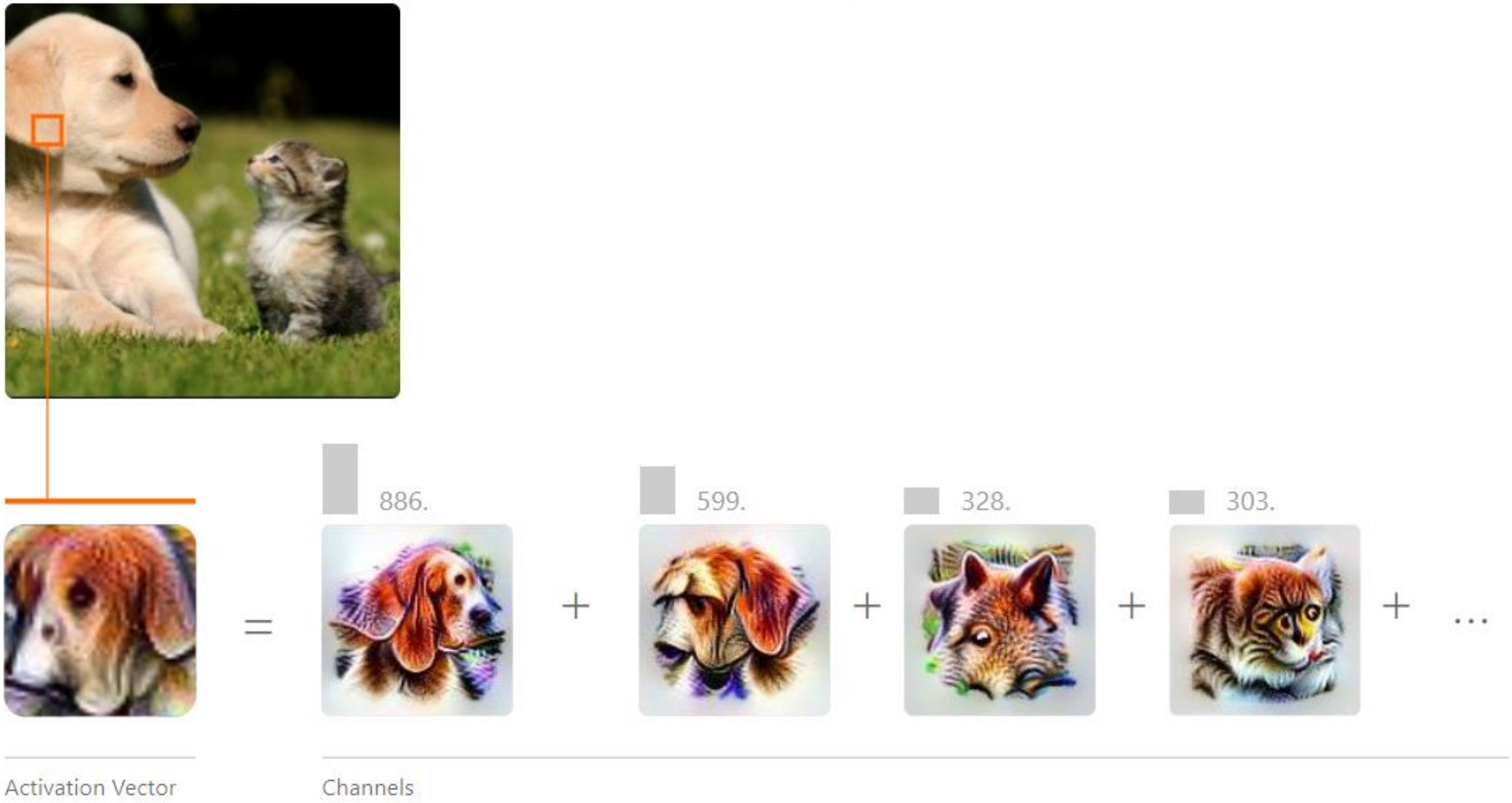


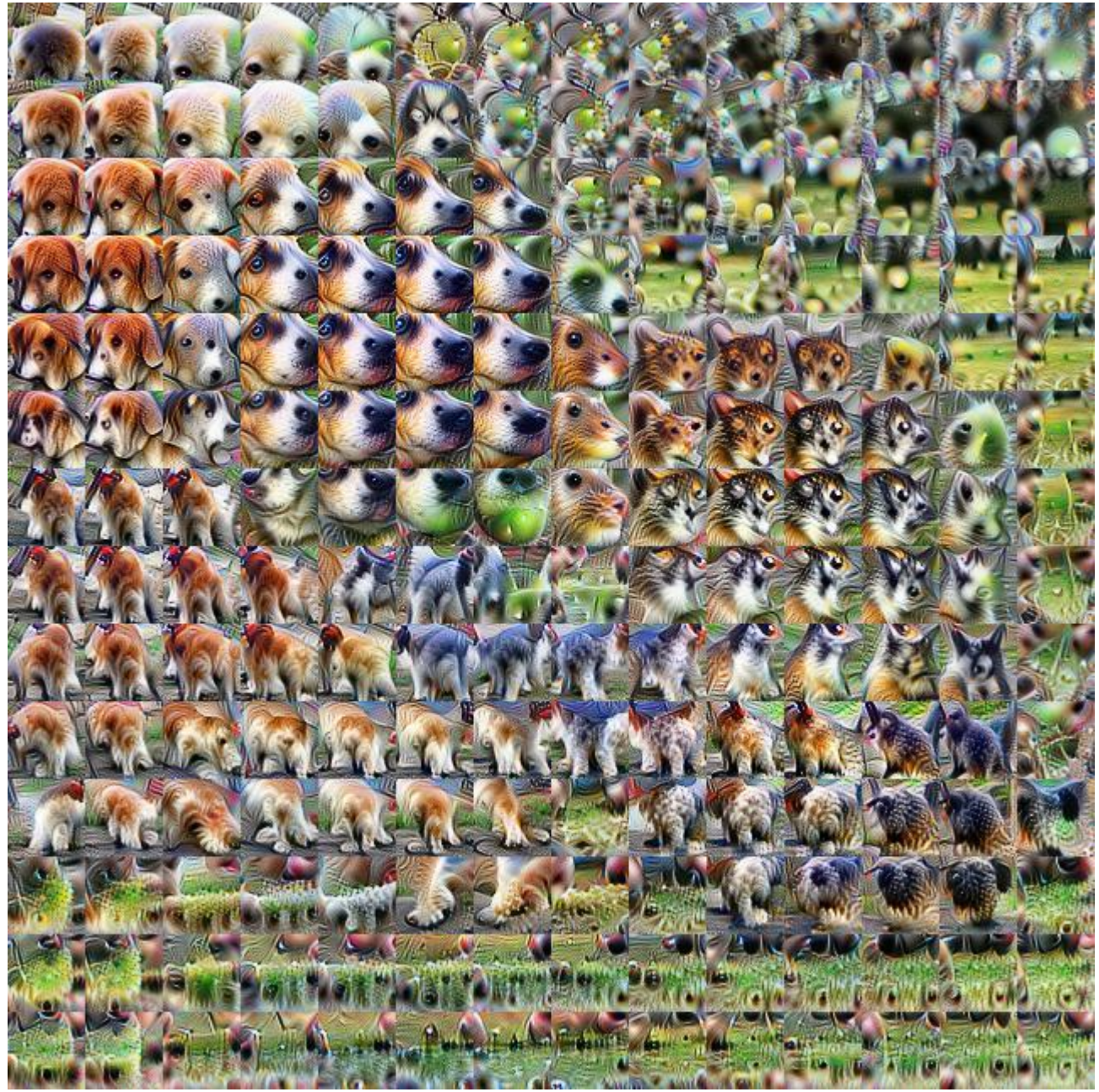
Making sense of these activations is hard because we usually work with them as abstract vectors:

> $a_{4,1}$ = [0, 0, 0, 25.2, 164.1, 0, 42.7, 4.51, 115.0, 51.3, 0, 0,...

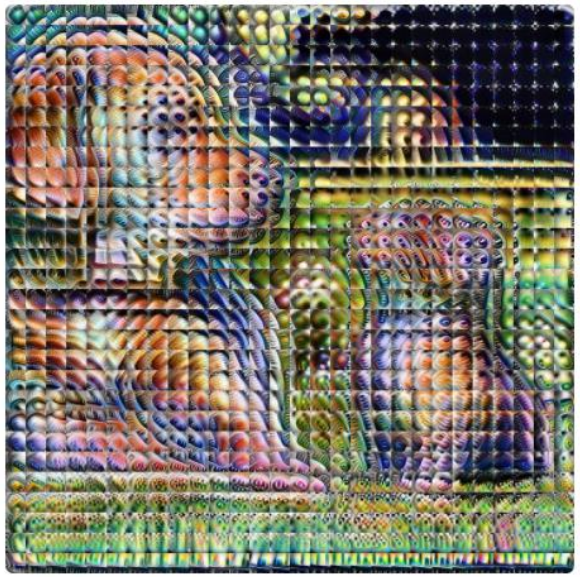
With feature visualization, however, we can transform this abstract vector into a more meaningful "semantic dictionary".

Visualizing spatial activations

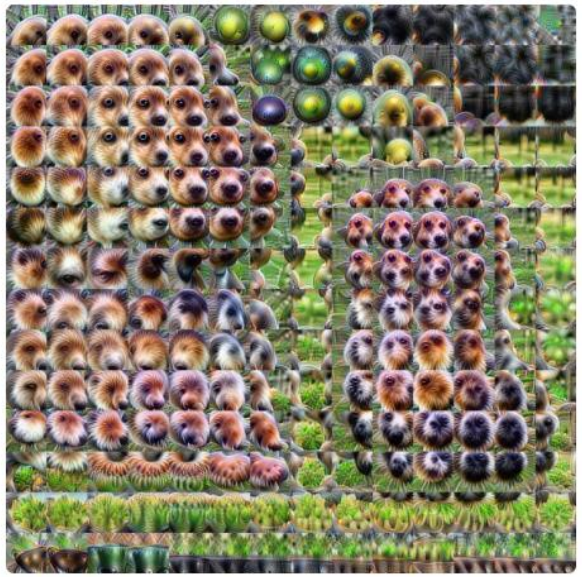




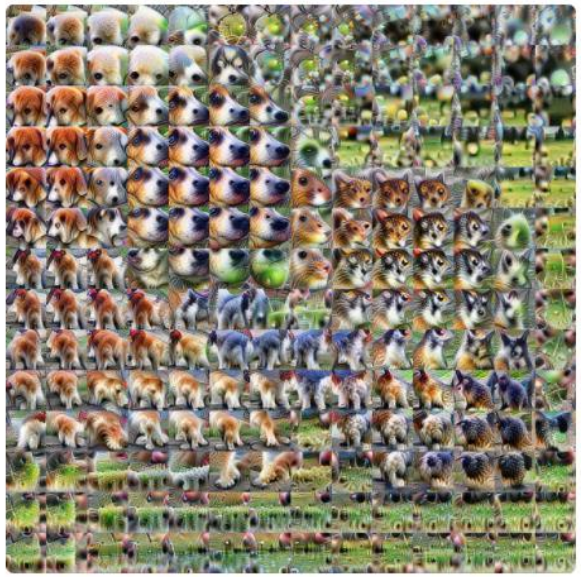
Evolving understanding of the network



MIXED3A



MIXED4A

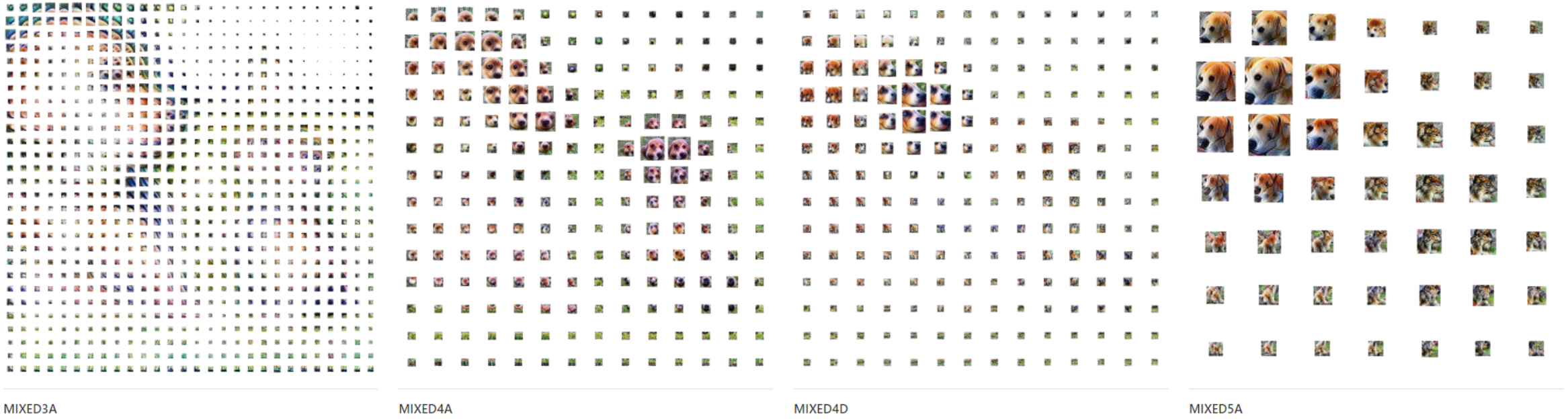


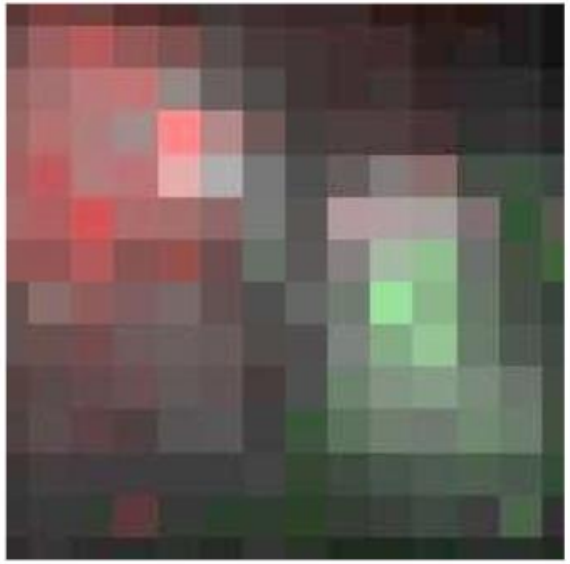
MIXED4D

MIXED5A

Evolving understanding of the network

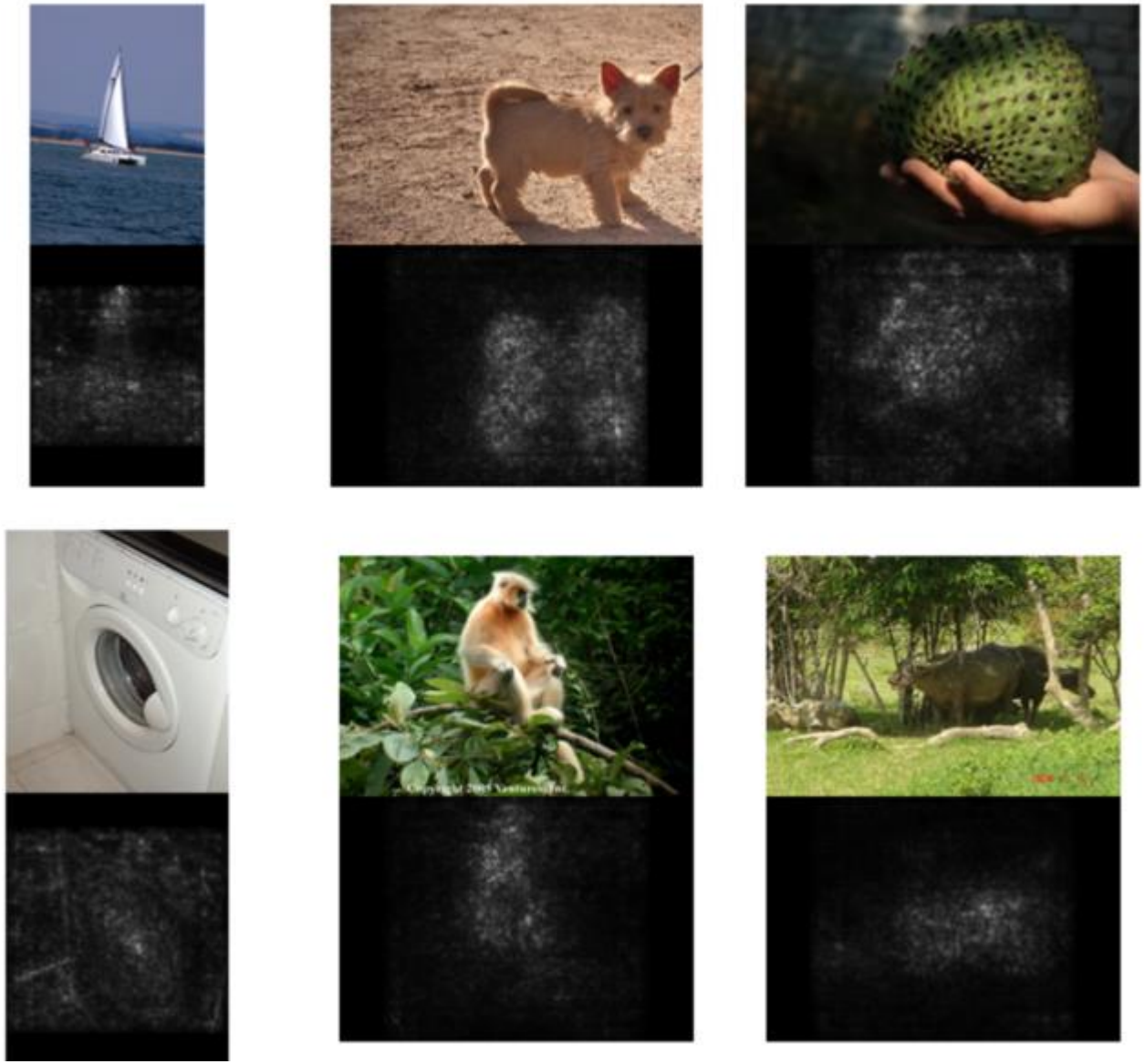
Normalized by the magnitude of the activations





Attribution

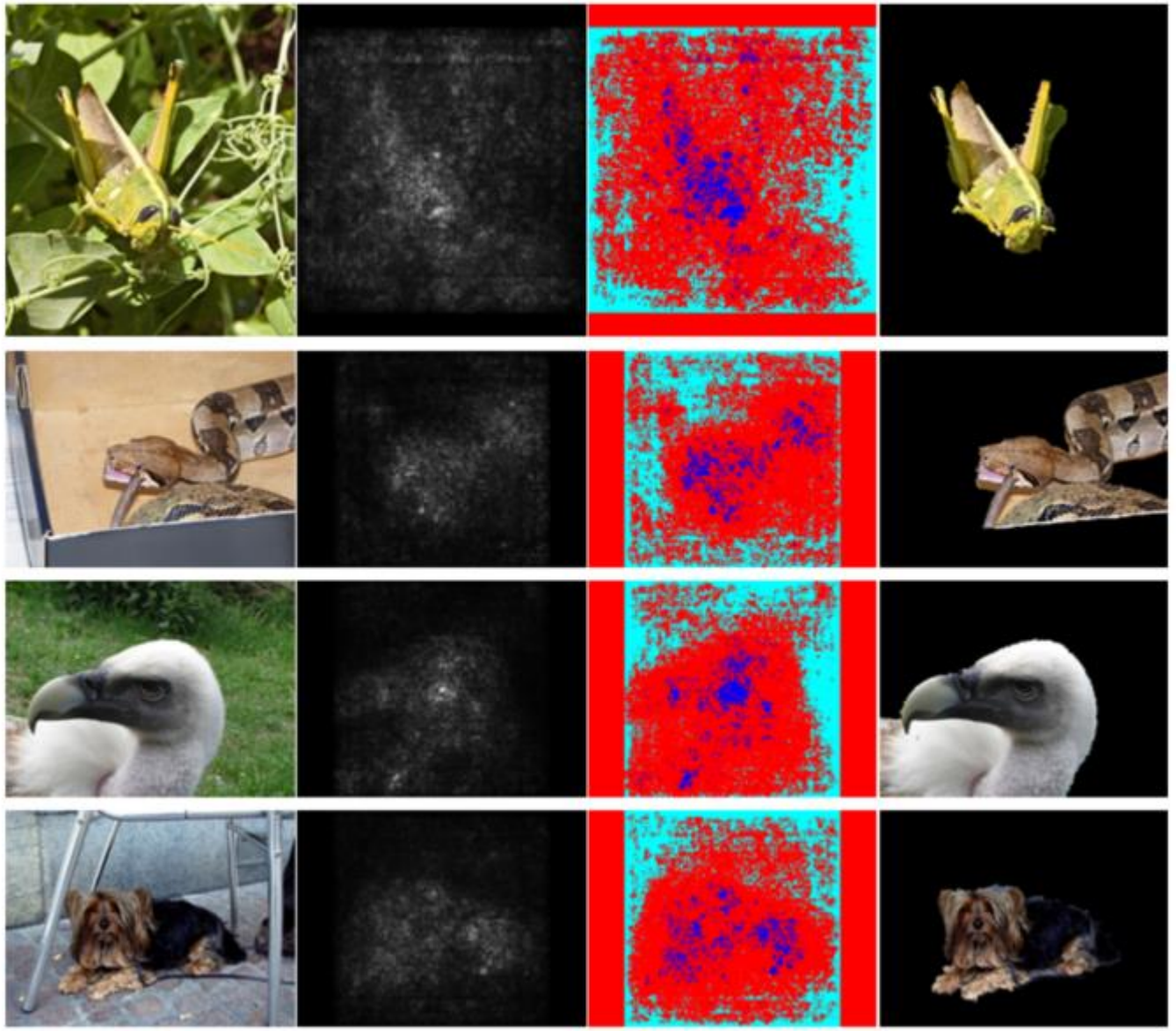
Saliency map



$$w = \left. \frac{\partial S_c}{\partial I} \right|_{I_0}$$

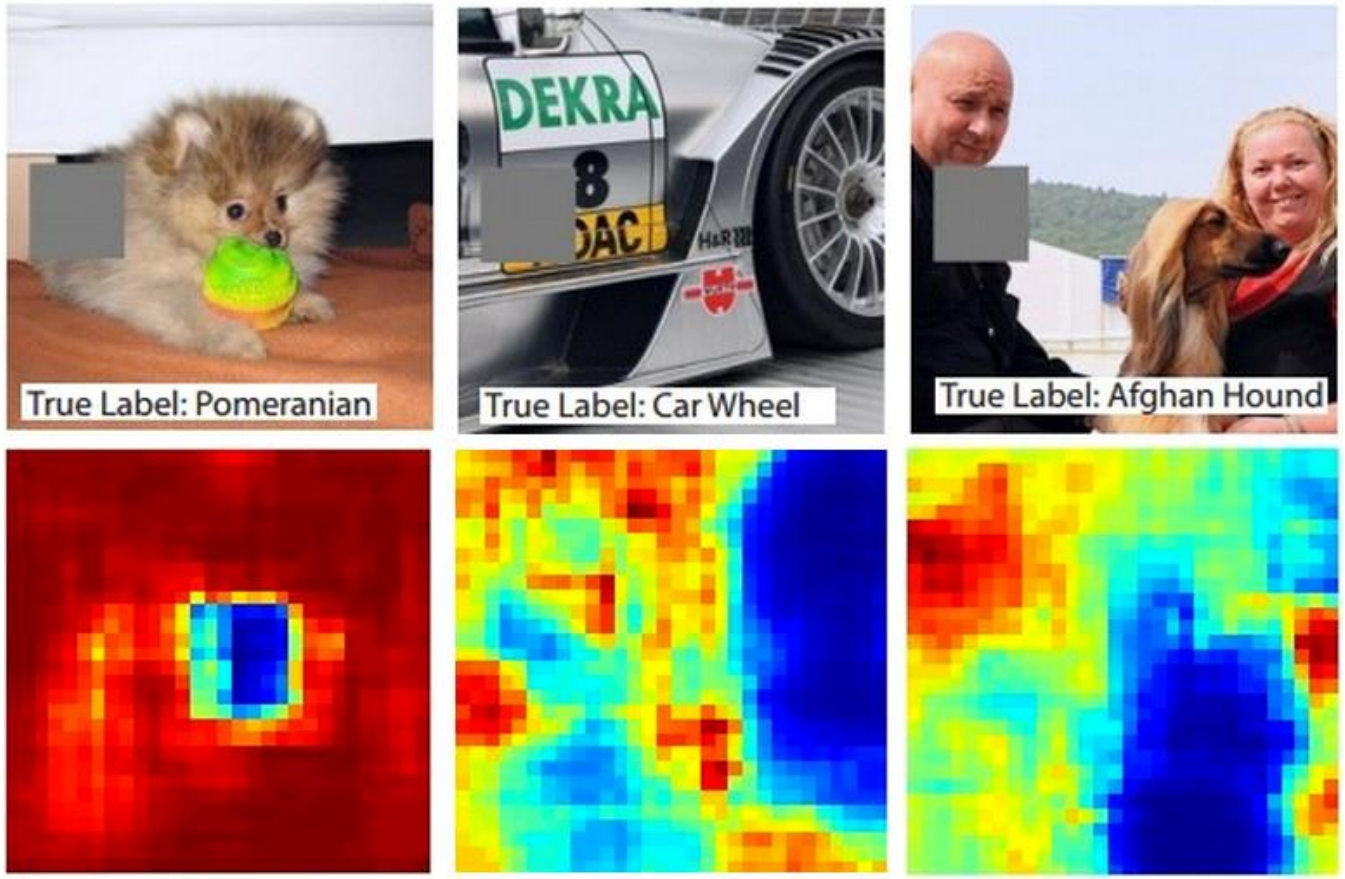
Simonyan et al (2014): Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps

Saliency map



Simonyan et al (2014): Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps

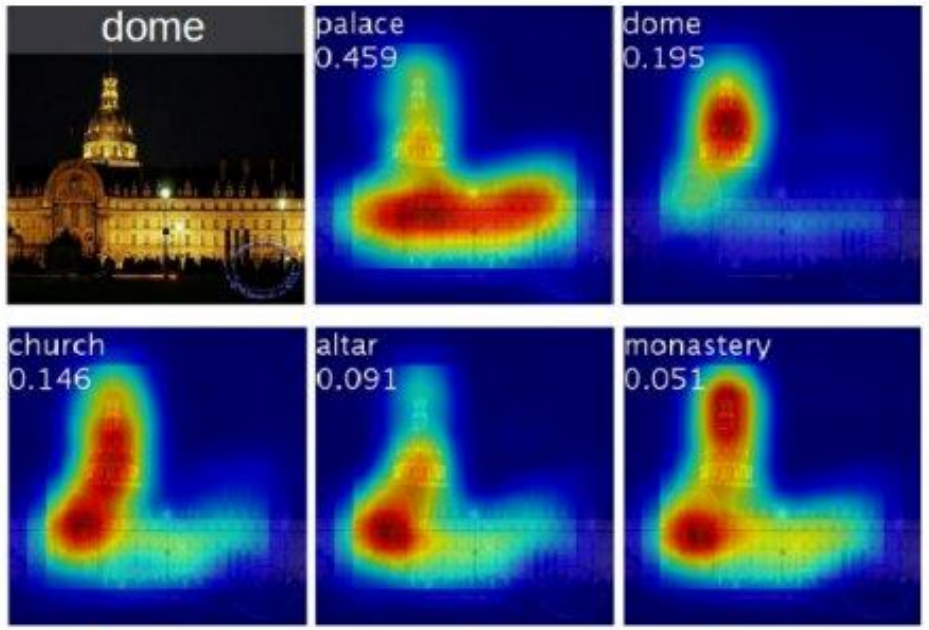
Occlusion-based saliency map



Zeiler and Fergus (2013): Visualizing and Understanding Convolutional Networks

Heatmaps of class activation

- Class activation map (CAM) visualization

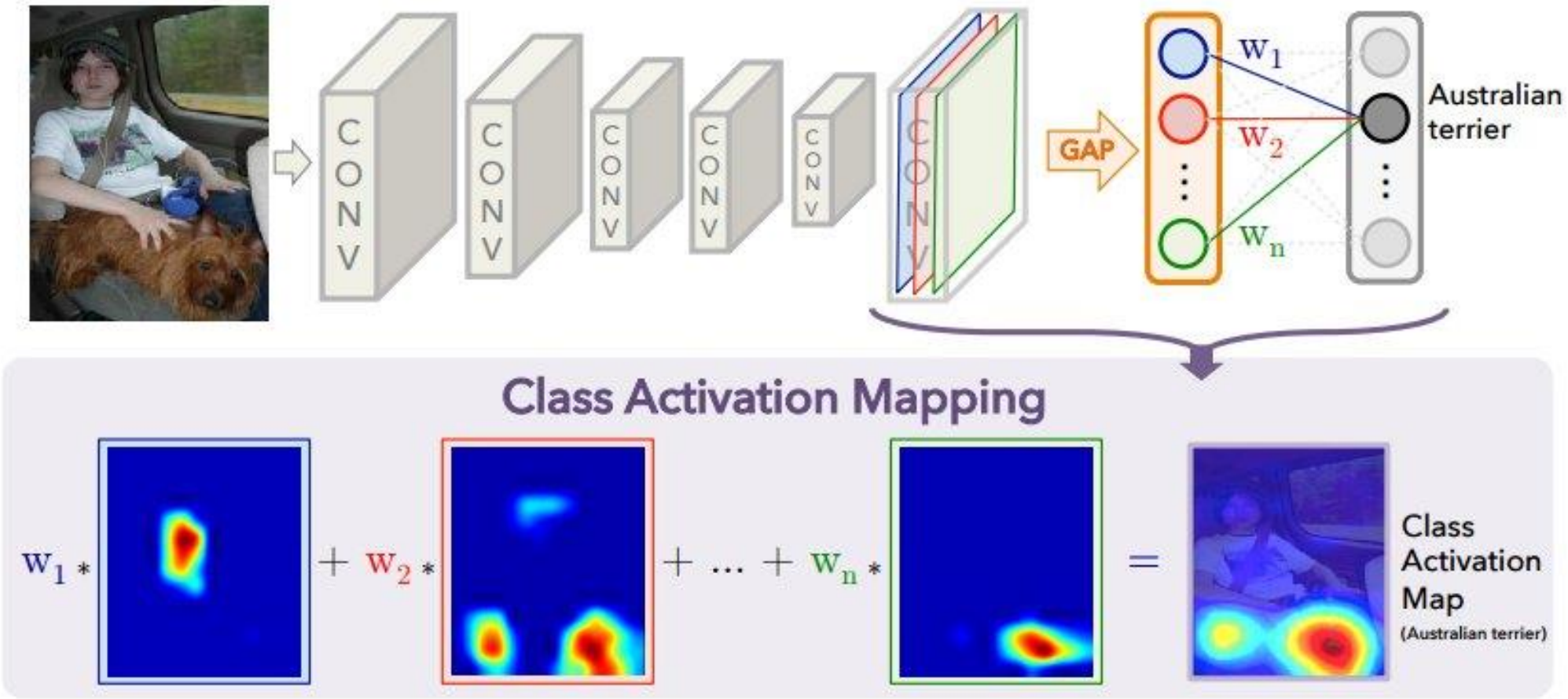


Class activation maps of top 5 predictions

Class activation maps for one object class

https://github.com/metalbubble/CAM

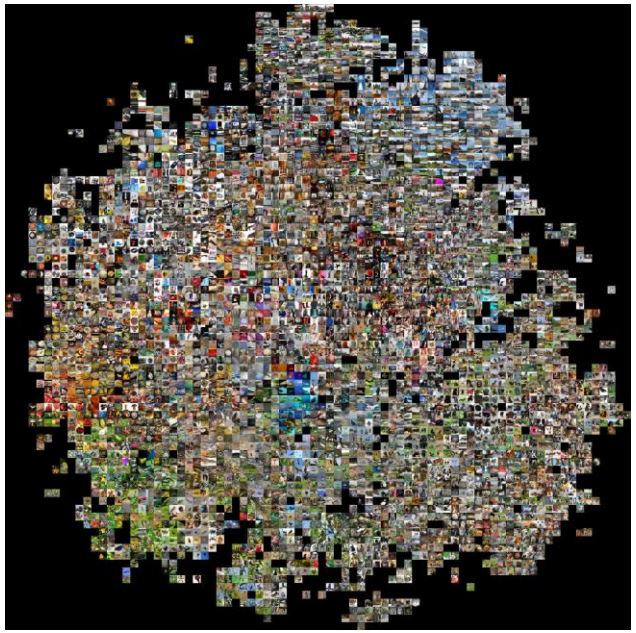
Heatmaps of class activation



https://github.com/metalbubble/CAM

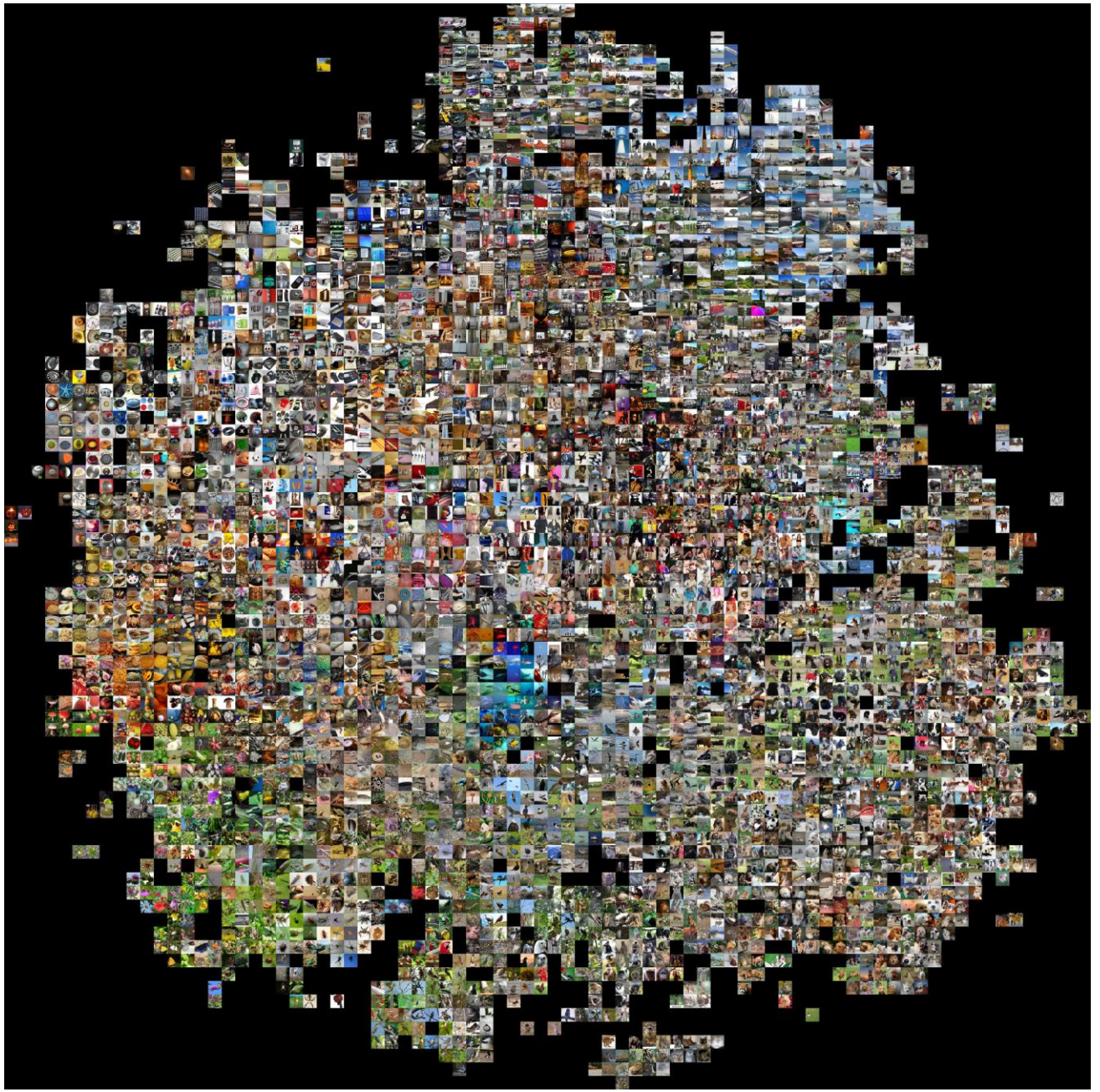
Watch more

https://www.youtube.com/watch?v=fZvOy0VXWAI

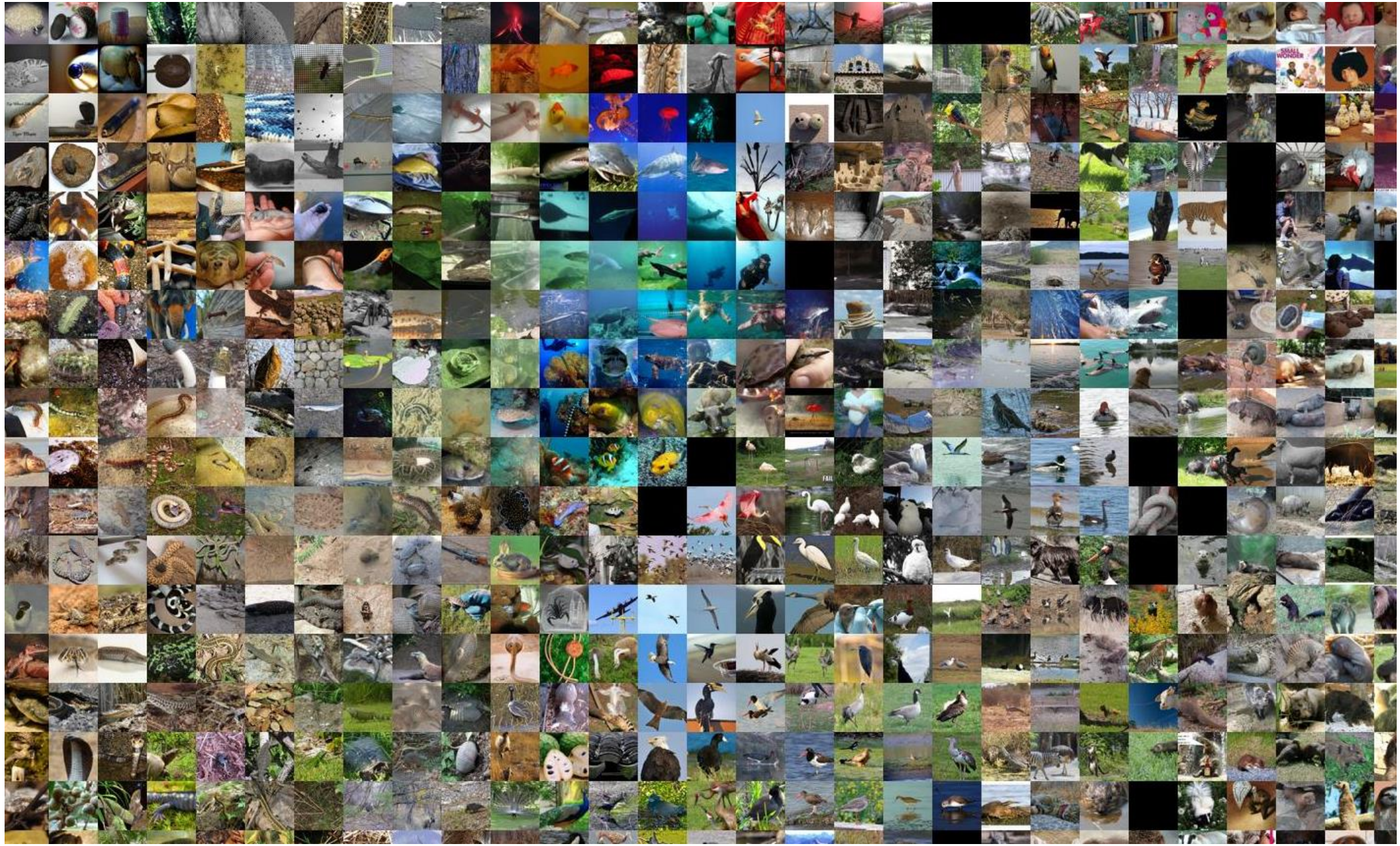


Dimensionality Reduction

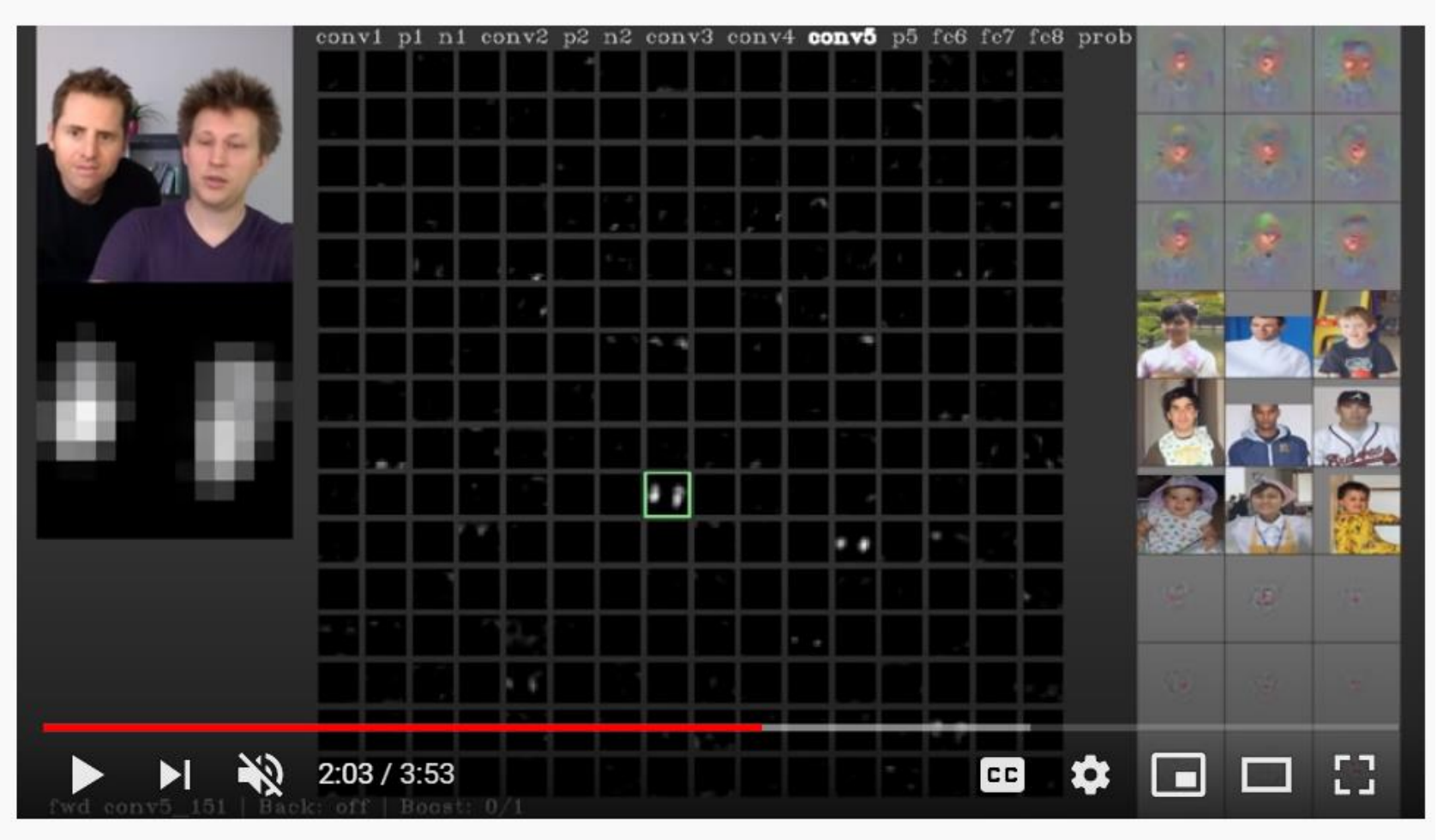
Embedding the codes with t-SNE



source



Visualizing intermediate activations



https://www.youtube.com/watch?v=AgkfIQ4IGaM

Deep Visualization Toolbox

yosinski.com/deepvis

#deepvis

Jason Yosinski

Jeff Clune

Anh Nguyen

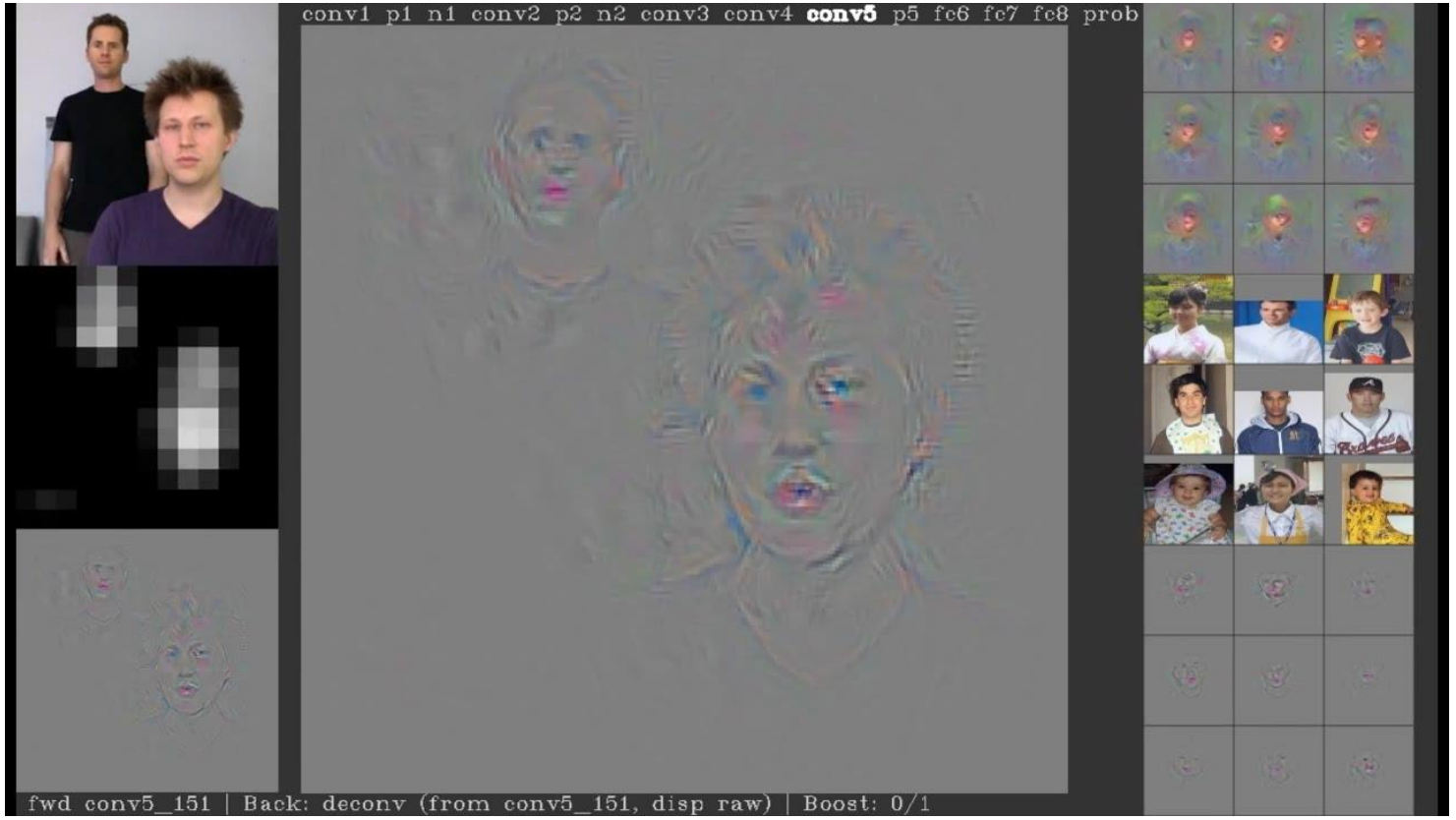
Thomas Fuchs

Hod Lipson

Cornell University

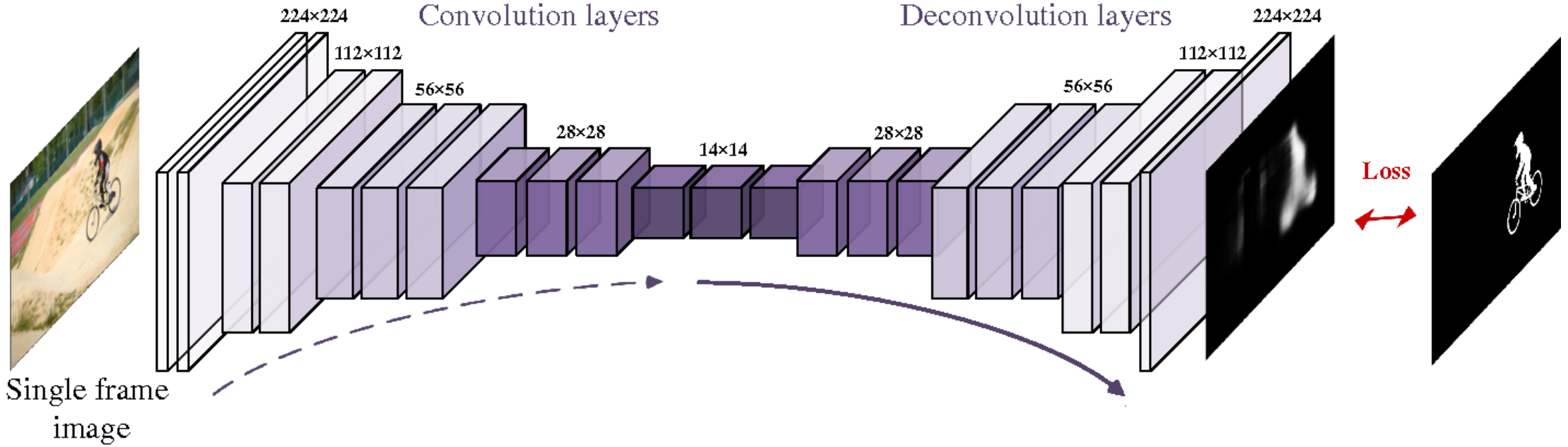
University of Wyoming

Jet Propulsion Laboratory
California Institute of Technology



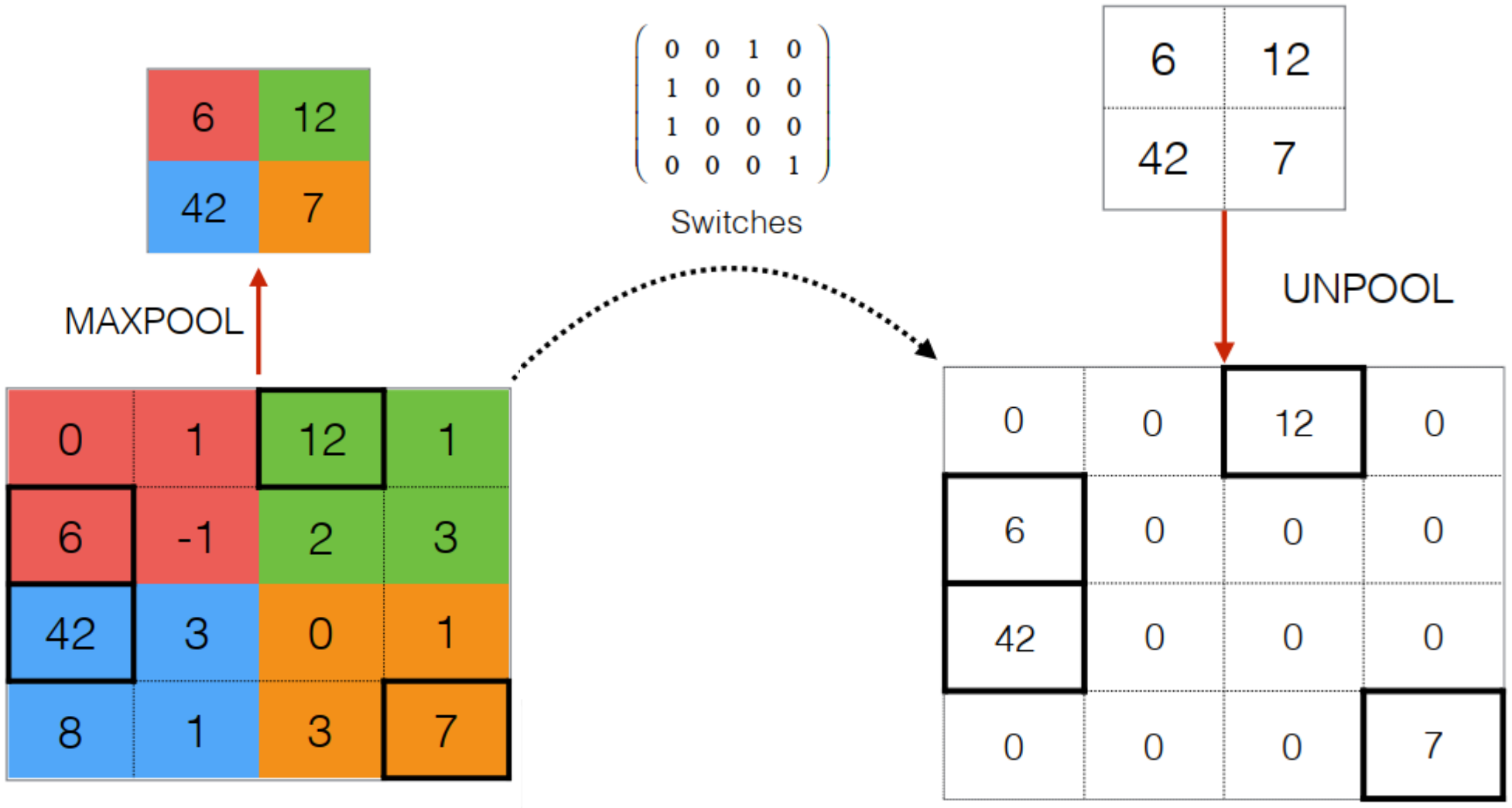
Deconvolution for activation visualization

Transposed convolution

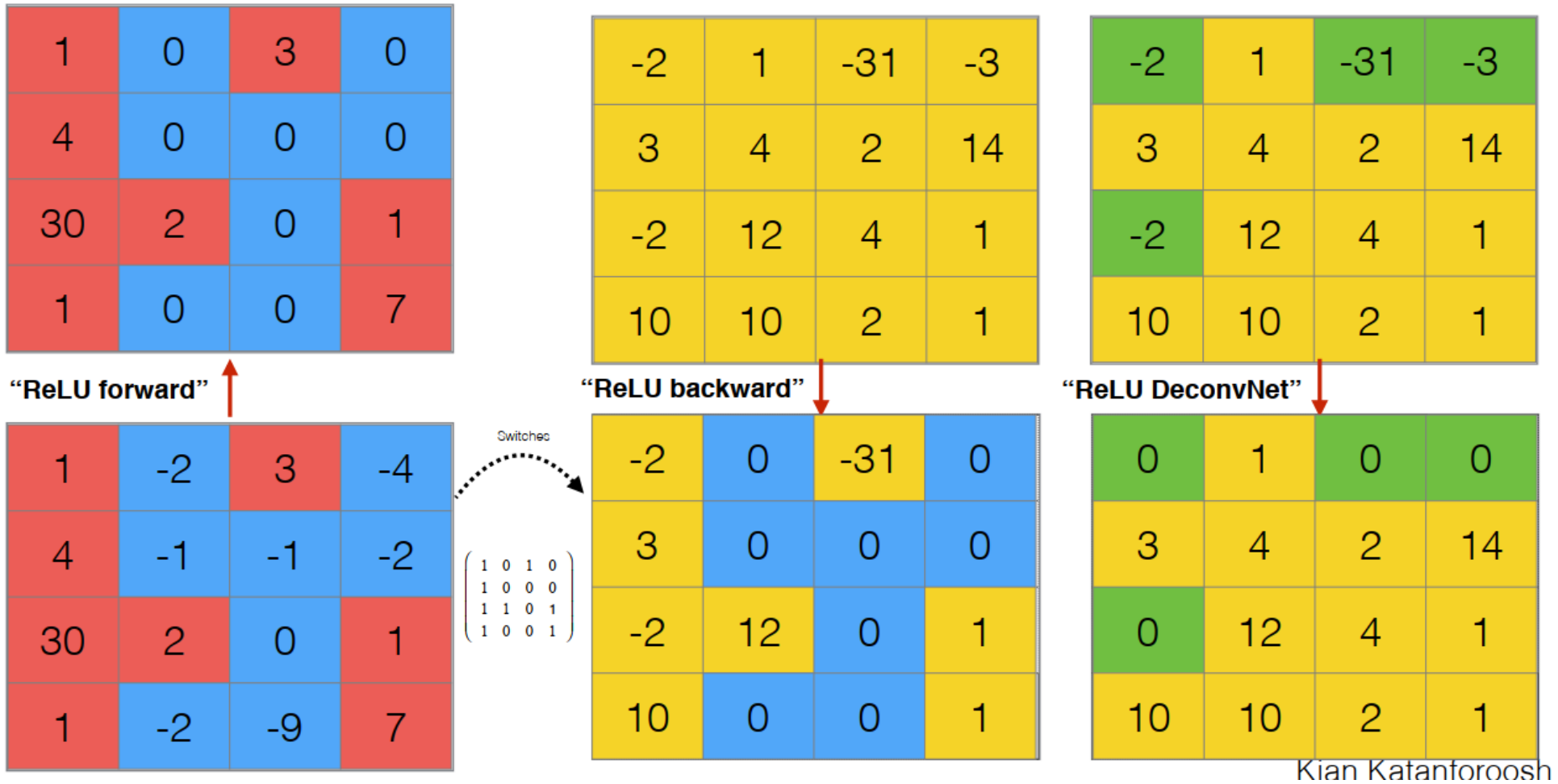


source

Unpooling


6 12
42 7
MAXPOOL
0 1 12 1
6 -1 2 3
42 3 0 1
8 1 3 7
0 0 1 0
1 0 0 0
1 0 0 0
0 0 0 1
Switches
6 12
42 7
UNPOOL
0 0 12 0
6 0 0 0
42 0 0 0
0 0 0 7

Backward ReLU



Kian Katanforoosh

Transposed convolution

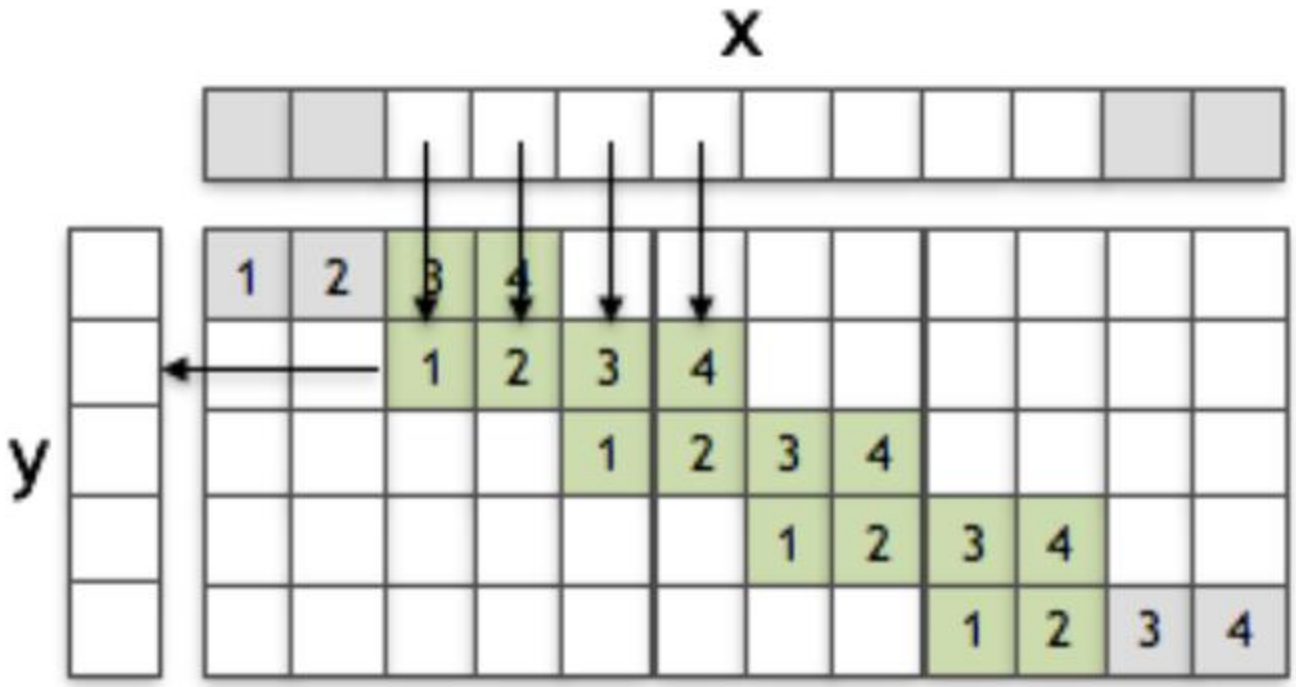


Figure 2: Convolution with stride 2 in 1D

Transposed convolution

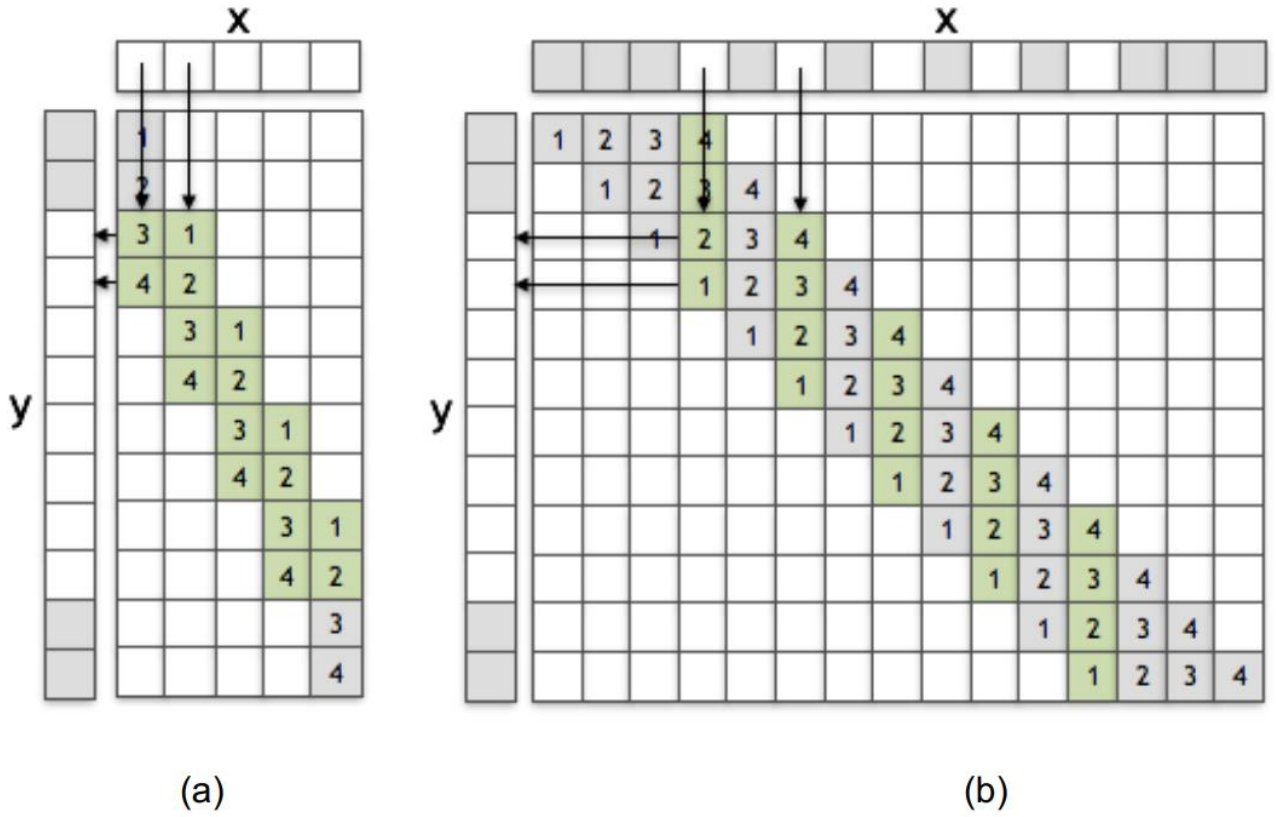
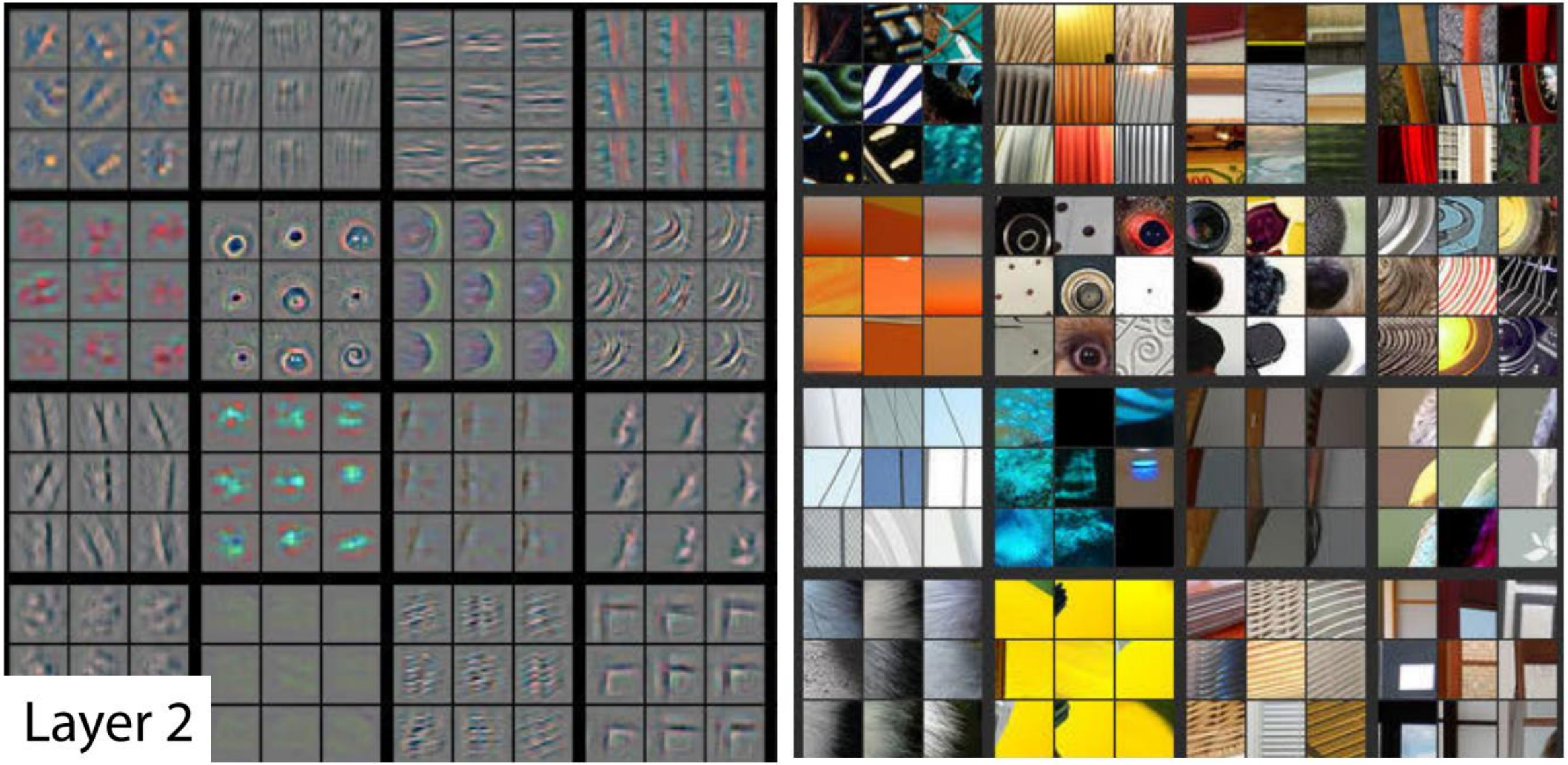


Figure 3: (a) Transposed convolution with stride 2 and (b) sub-pixel convolution with stride ½ in 1D

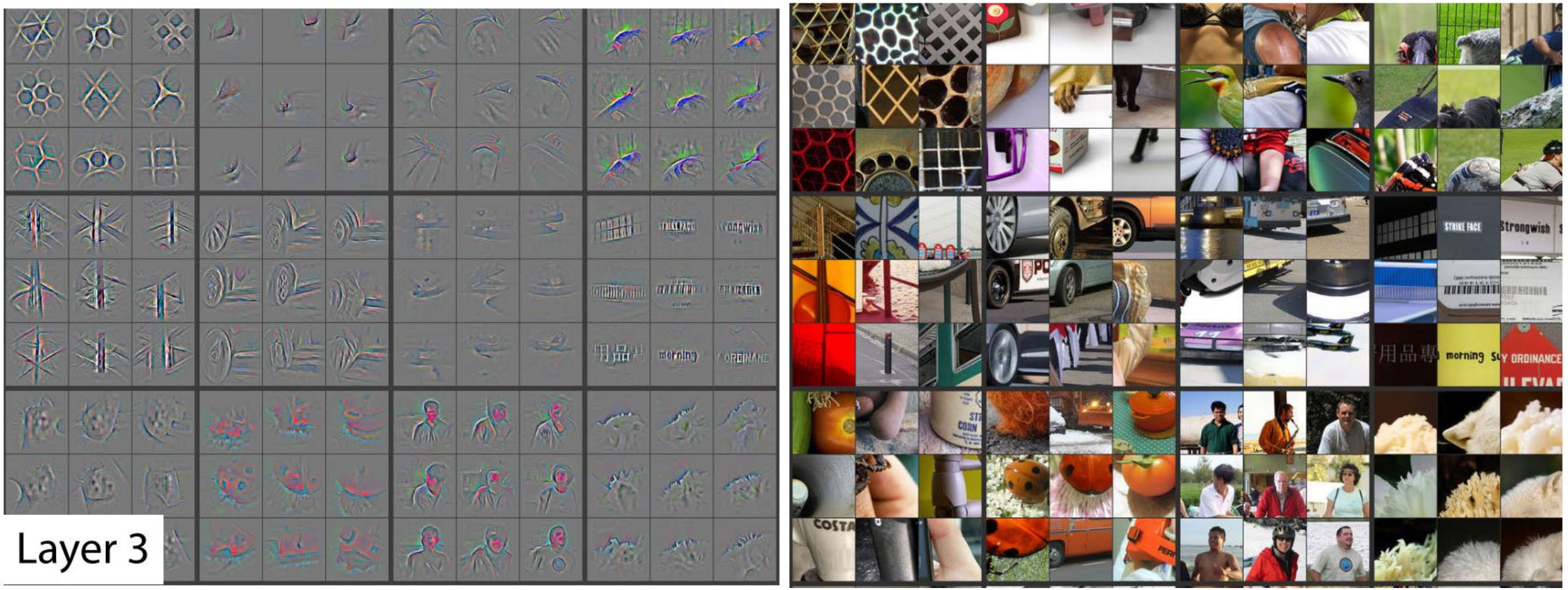
Shi et al: Is the deconvolution layer the same as a convolutional layer?

Layer 1

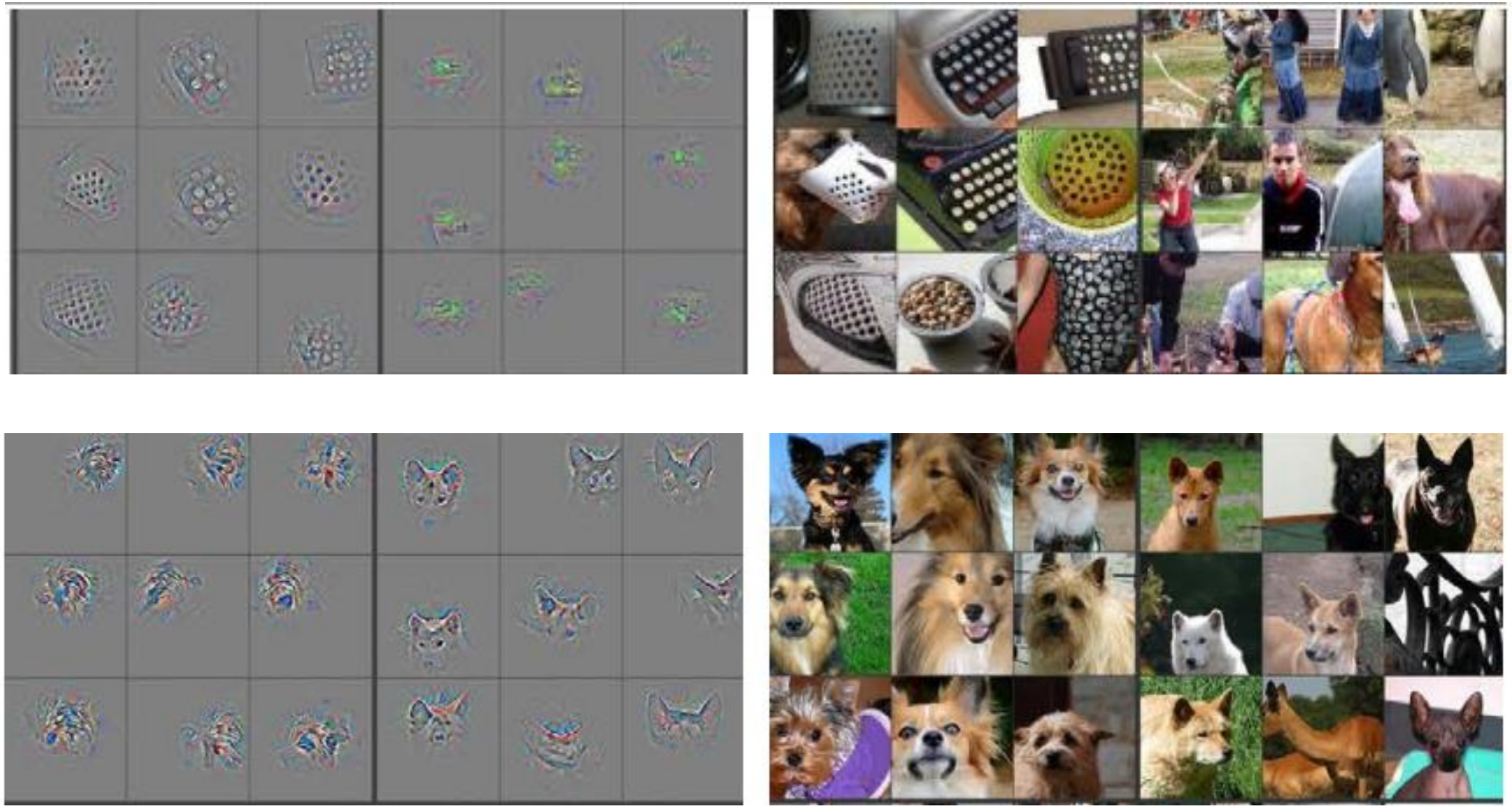
Zeiler and Fergus (2013): Visualizing and Understanding Convolutional Networks



Zeiler and Fergus (2013): Visualizing and Understanding Convolutional Networks



Zeiler and Fergus (2013): Visualizing and Understanding Convolutional Networks



Zeiler and Fergus (2013): Visualizing and Understanding Convolutional Networks

Visualizing intermediate activations

1. The first layer acts as a collection of various edge detectors.

    - At that stage, the activations retain almost all of the information present in the initial picture.

Visualizing intermediate activations

2. As you go higher, the activations become increasingly abstract and less visually interpretable.

    - They begin to encode higher-level concepts such as “cat ear” and “cat eye.” Higher presentations carry increasingly less information about the visual contents of the image, and increasingly more information related to the class of the image.

Visualizing intermediate activations

3. The sparsity of the activations increases with the depth of the layer

   - In the first layer, all filters are activated by the input image; but in the following layers, more and more filters are blank. This means the pattern encoded by the filter isn't found in the input image.

Representations – a universal characteristic

The features extracted by a layer become increasingly abstract with the depth of the layer.

- The activations of higher layers carry less and less information about the specific input being seen, and more and more information about the target

Representations – a universal characteristic

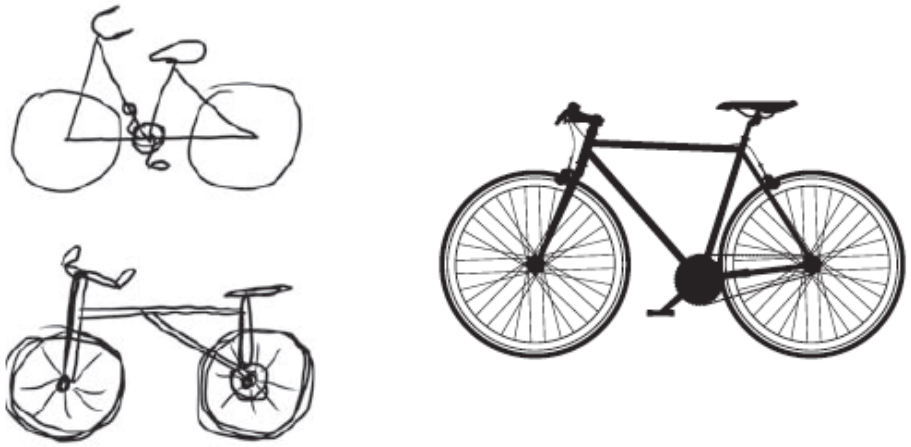
Information distillation pipeline

Raw data going in (in this case, RGB pictures) and being repeatedly transformed so that irrelevant information is filtered out (for example, the specific visual appearance of the image), and useful information is magnified and refined (for example, the class of the image).

Representations – a universal characteristic

- Analogous to the way humans and animals perceive the world

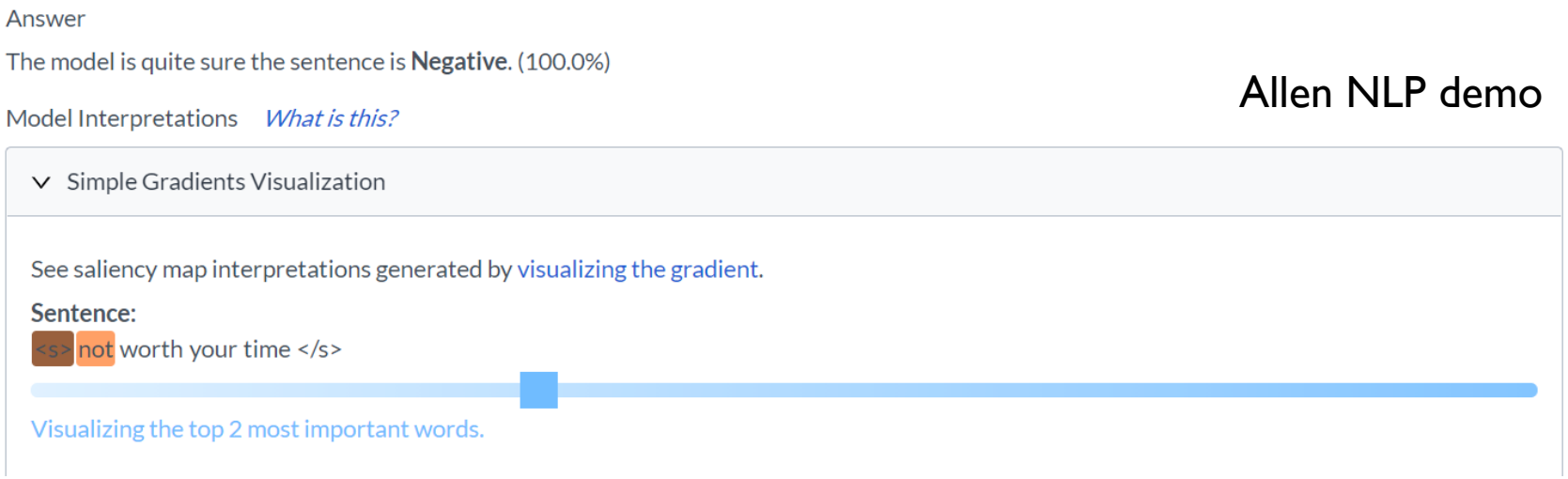
Brain has learned to completely abstract its visual input (to transform it into high-level visual concepts while filtering out irrelevant visual details)



Interpretability in NLP

Mostly borrowed from Vision

- Visualizing the gradients
- Integrated gradients
- SmoothGrad



Allen NLP demo