Visualising convolutional networks

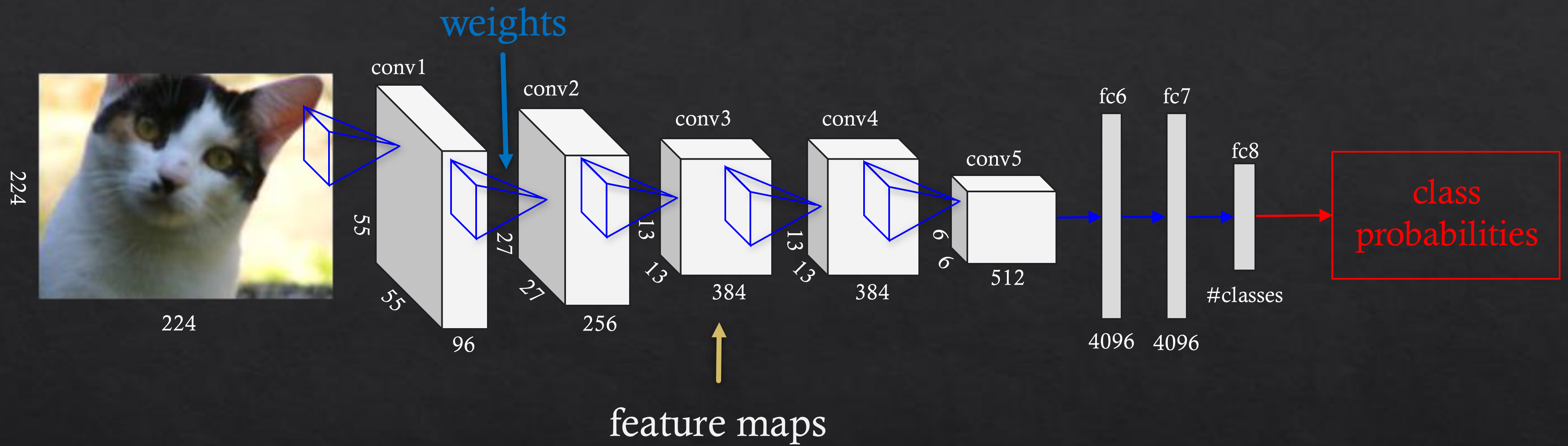
Hakan Bilen

Machine Learning Practical - MLP Lecture 12
16 January 2019
http://www.inf.ed.ac.uk/teaching/courses/mlp/

Lectures in second semester

- Understanding convolutional networks
- Generative adversarial networks
- Domain adaptation and transfer learning
- Convolutional network design and compression (Dr Elliot Crowley)
- Object detection and semantic segmentation
- Language and vision models
- Video analytics

Recap: Convolutional Neural Networks (CNNs)



What is inside the black box (filters and feature maps)?

Image credit: http://farm1.static.flickr.com/1/3375151_bf4e904a5d.jpg

Why does it matter?

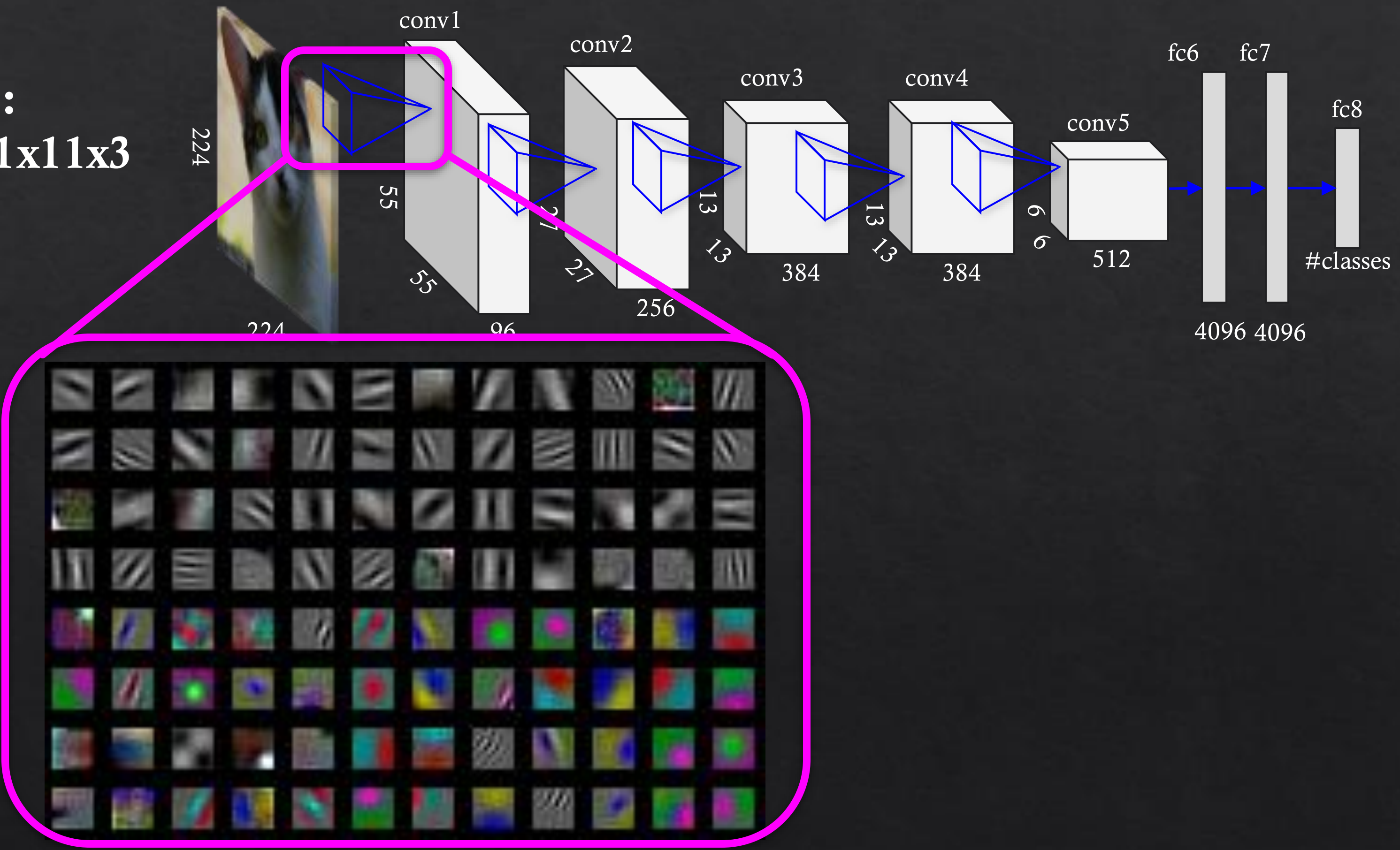
- Interpretability: understand what they learn and why they work
- Monitor training process (evolution of training)
- Gain intuitions to develop better models
- Diagnose potential problems

Today

1. Visualize filters / weights
2. Analyze activations
3. Deconvolutional networks
4. Saliency deconvolutional networks
5. Adversarial noise

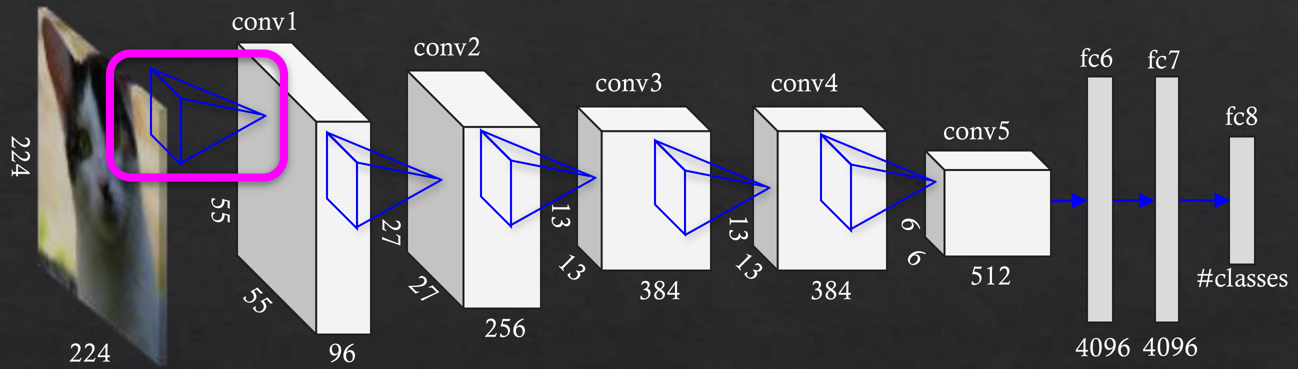
1. Visualize filters

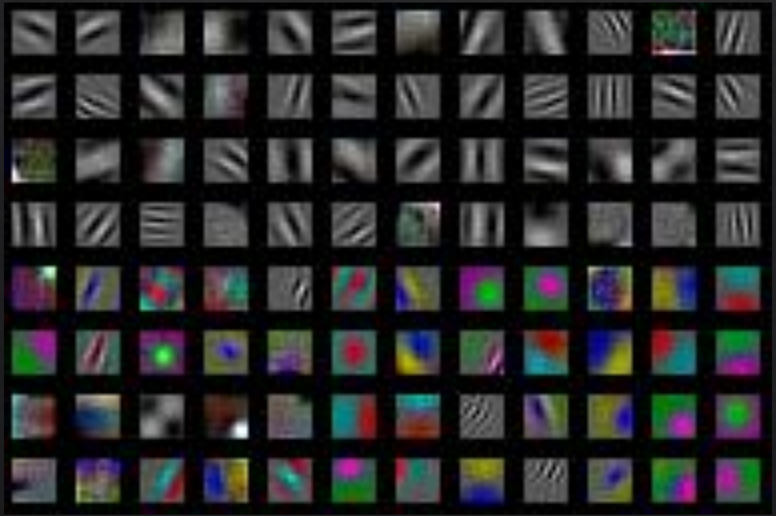
conv1:
96 filters 11x11x3



1. Visualize filters

conv1:
96 filters 11x11x3



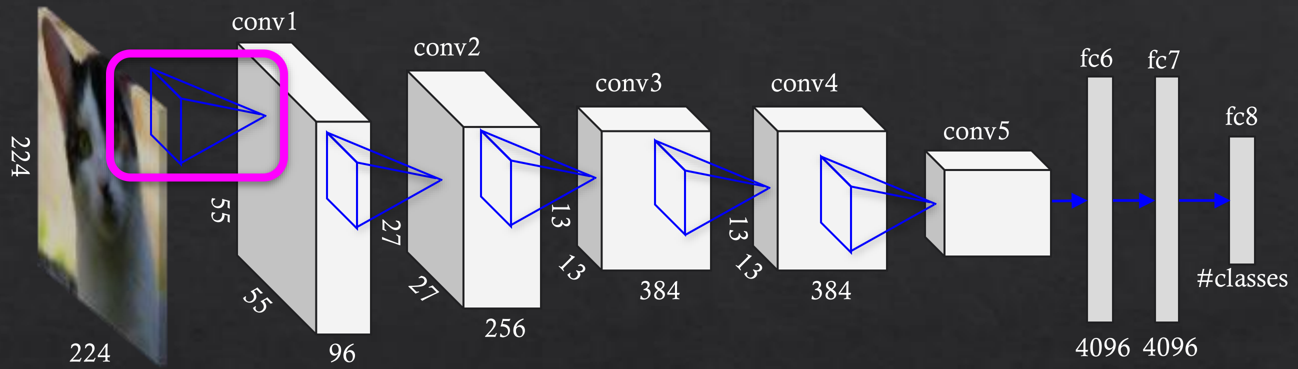


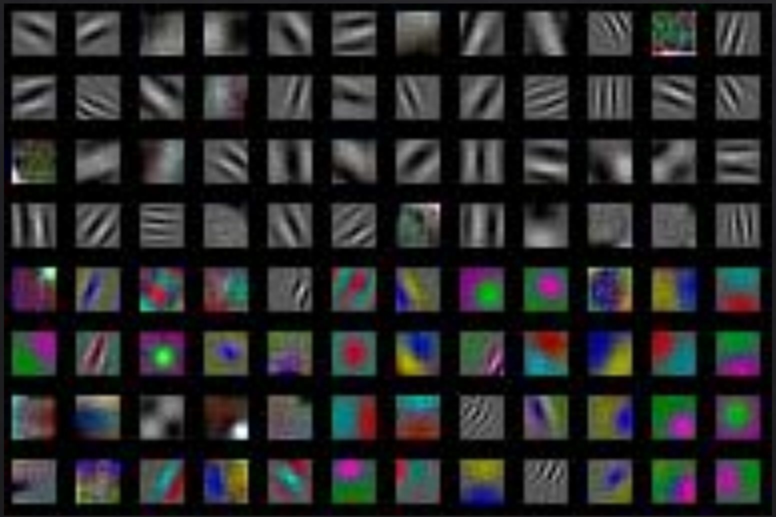
Question:

What do these filters detect?

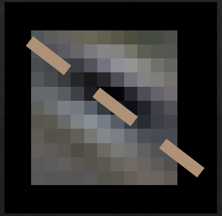
1. Visualize filters

conv1:
96 filters 11x11x3

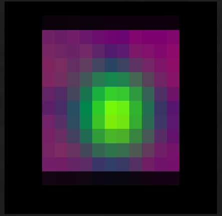




- Oriented edge filters (similar to Gabor filters)

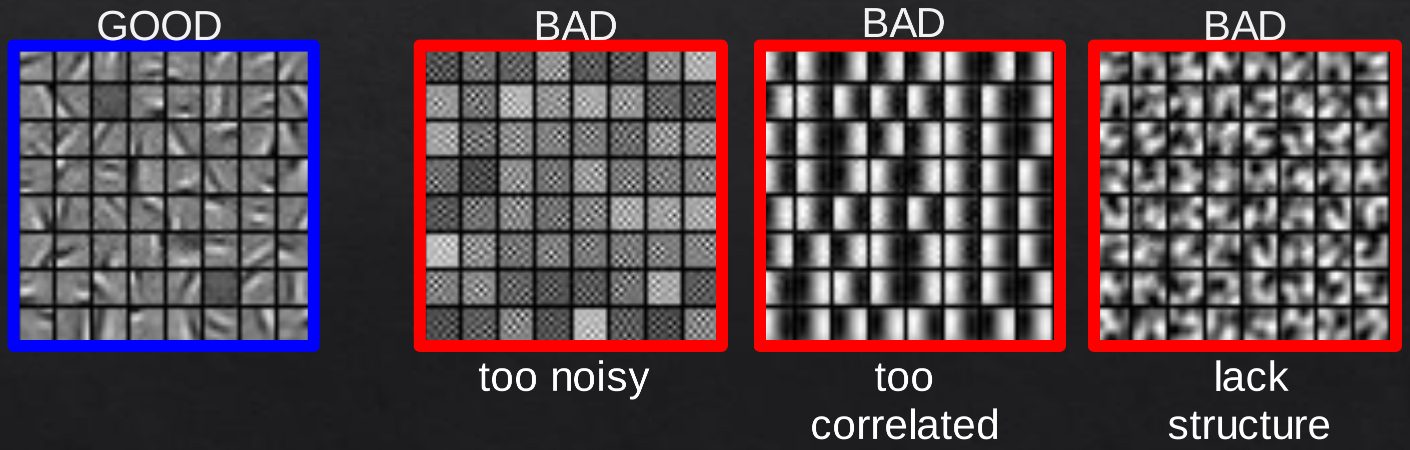


- Coloured blob detectors



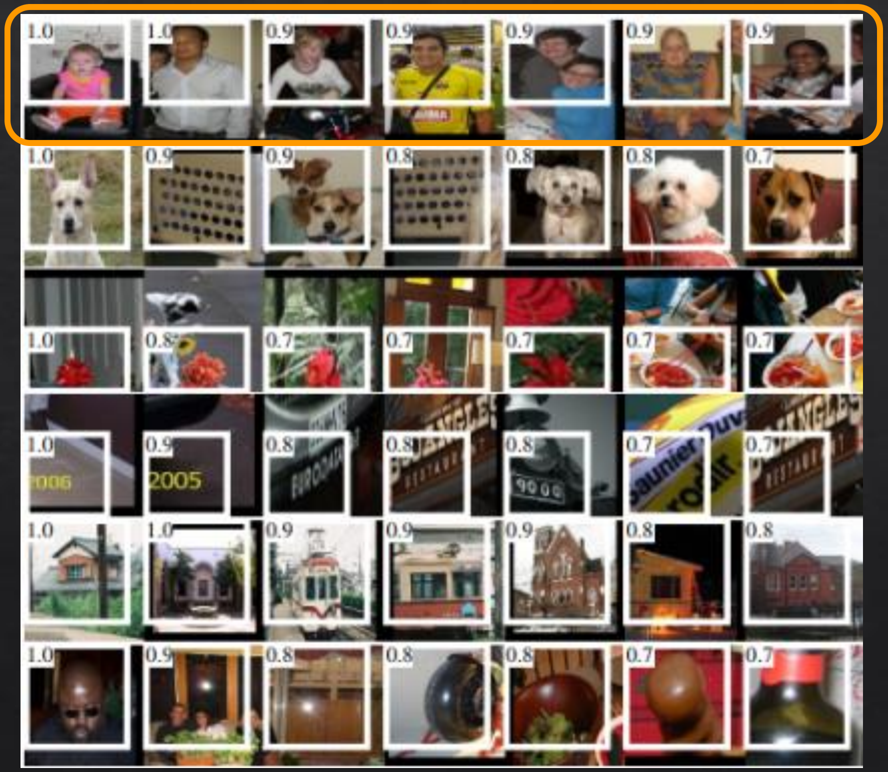
Monitoring filters during training

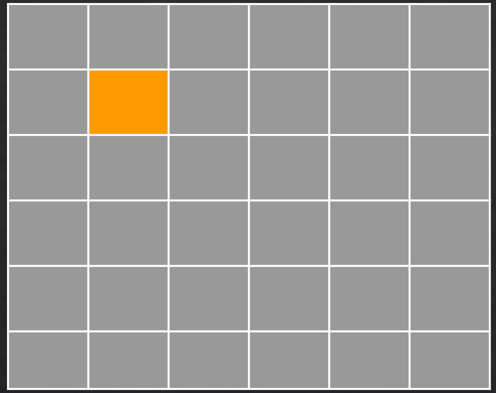
Good training: learned filters should exhibit structure and are uncorrelated



Slide credits: Ranzatto & Lecun

2. Analyze activations



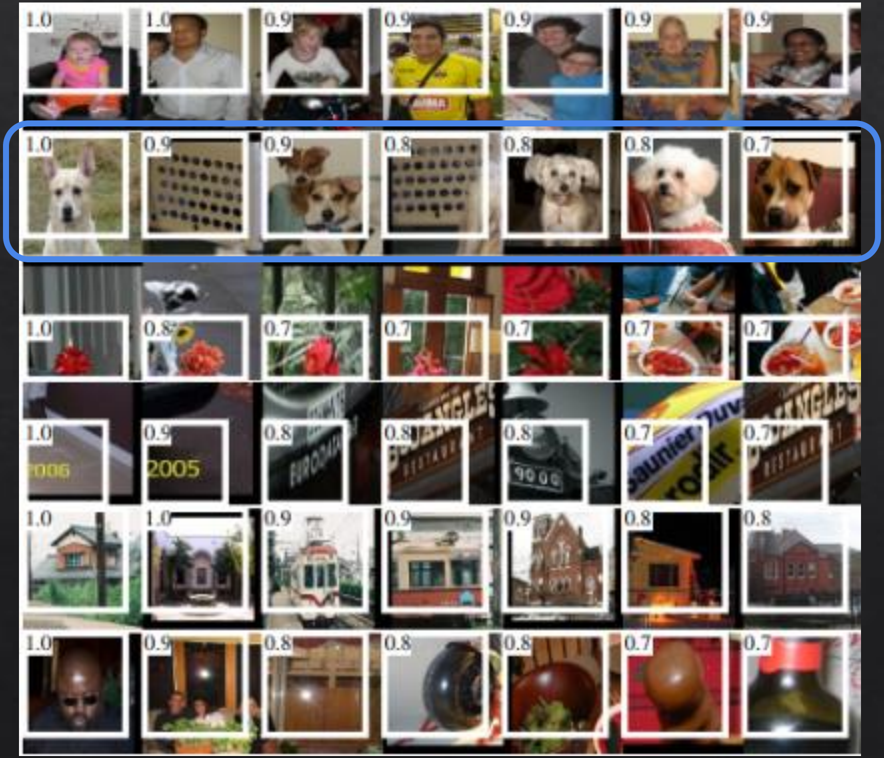


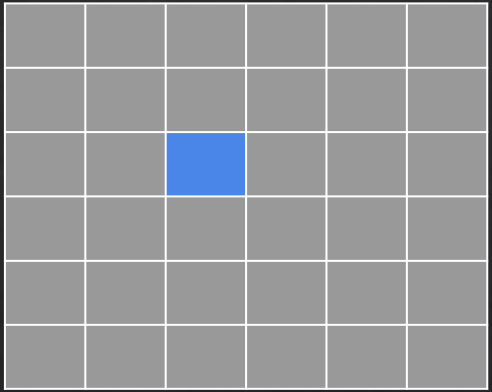
Pool5

1. Pick a neuron at a layer
2. Record it for multiple images
3. Show the images with highest activation value
4. See whether the images correspond to a common concept

Girshick et al. Rich feature hierarchies for accurate object detection and semantic segmentation, CVPR'14

2. Analyze activations





☺ Easy to implement
☹ Only qualitative analysis

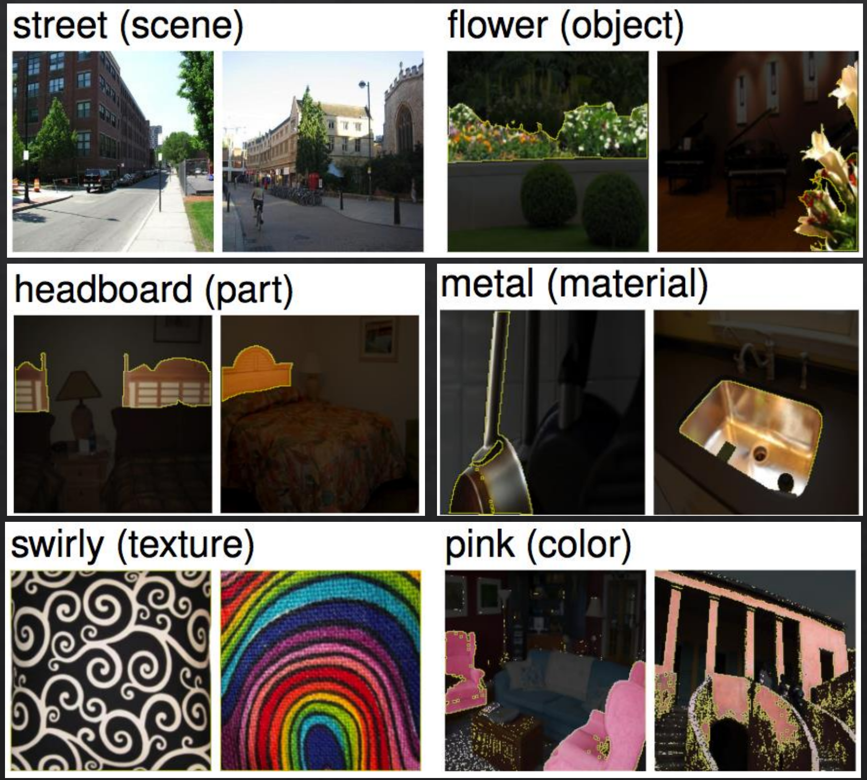
Girshick et al. Rich feature hierarchies for accurate object detection and semantic segmentation, CVPR’14

2. Analyze activations quantitatively

Q. How can we quantify alignment with visual concepts?

I. Collect images and label all the pixels with various concepts

- objects, parts, scenes, textures, colours and materials



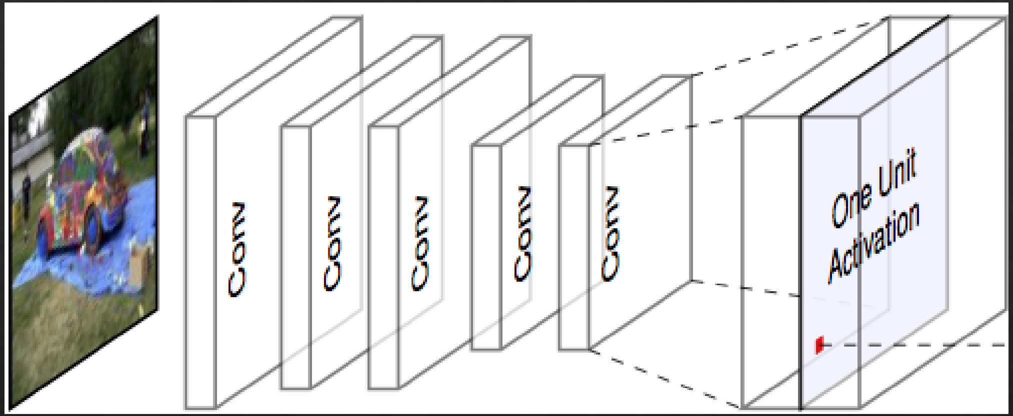
Bau et al., Network Dissection: Quantifying Interpretability of Deep Visual Representations, CVPR'17

2. Analyze activations quantitatively

II. Gather responses of neurons to known concepts

- Input image x to CNN
- Take an activation map $A_k(x)$ at layer l
- Threshold $P(A_k(x) > T)$
- Upscale to image size

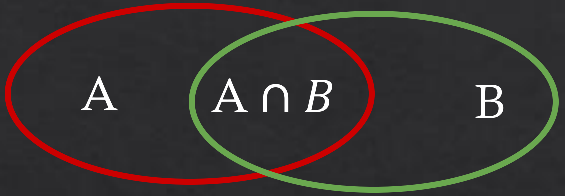
Forward pass



Bau et al., Network Dissection: Quantifying Interpretability of Deep Visual Representations, CVPR’17

2. Analyze activations quantitatively

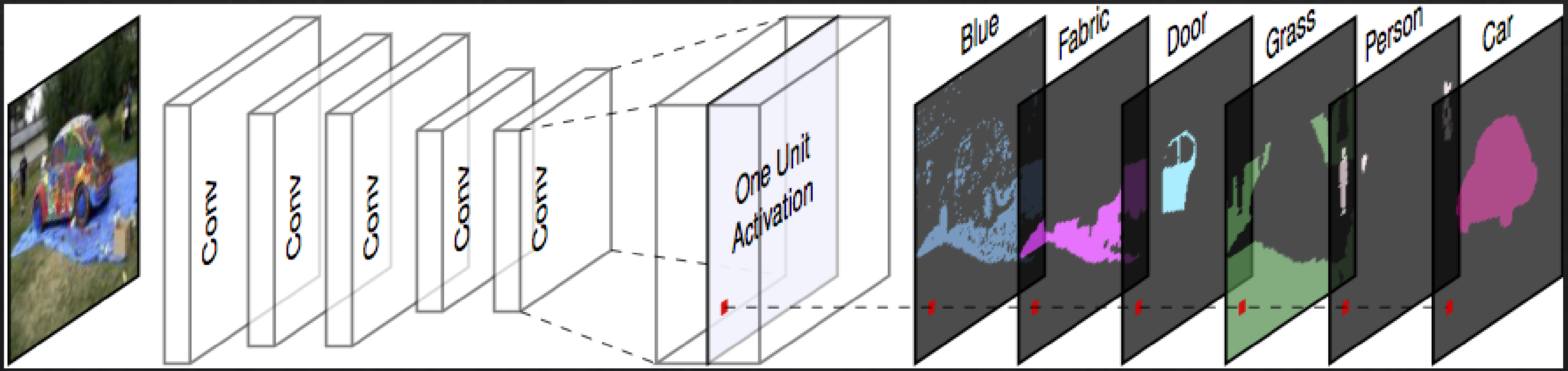
III. Measure overlap with human labelled concepts



Intersection over union

$$\text{IoU} = \frac{A \cap B}{A \cup B}$$

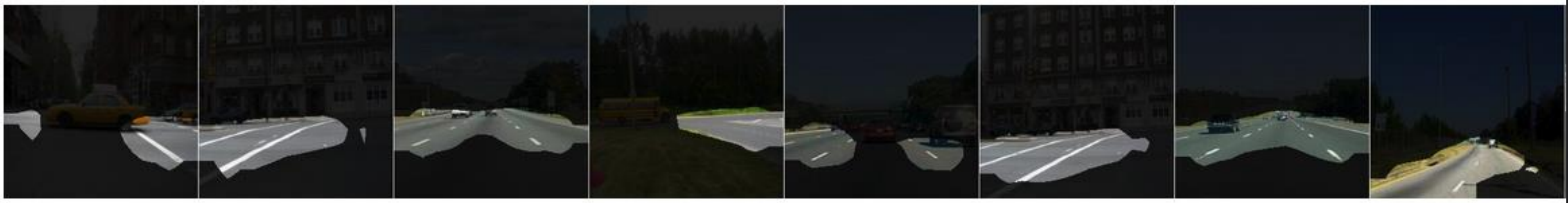
Forward pass



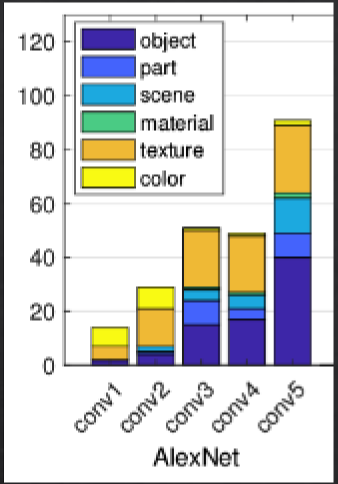
Bau et al., Network Dissection: Quantifying Interpretability of Deep Visual Representations, CVPR'17

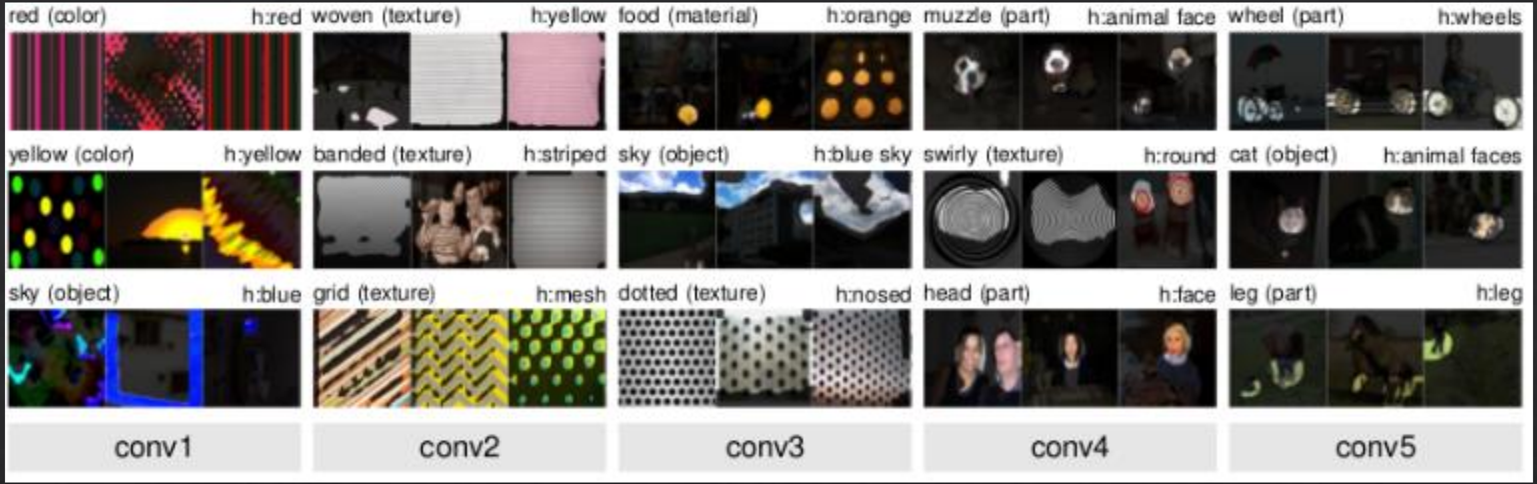
conv5 unit 79 car (object) IoU=0.13

conv5 unit 107 road (object) IoU=0.15



Bau et al., Network Dissection: Quantifying Interpretability of Deep Visual Representations, CVPR’17



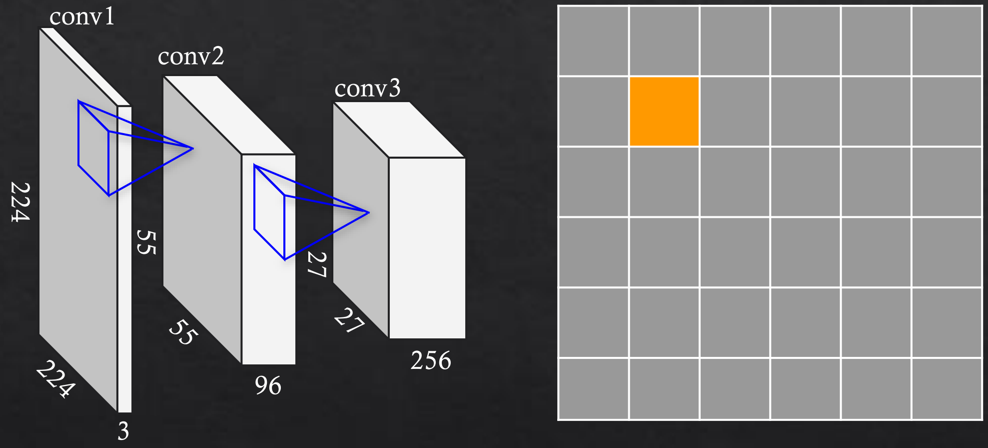


- More complex concepts emerge at the later layers
- Some low level concepts at the later layer are still useful for classification

Bau et al., Network Dissection: Quantifying Interpretability of Deep Visual Representations, CVPR'17

3. Deconvolutional networks

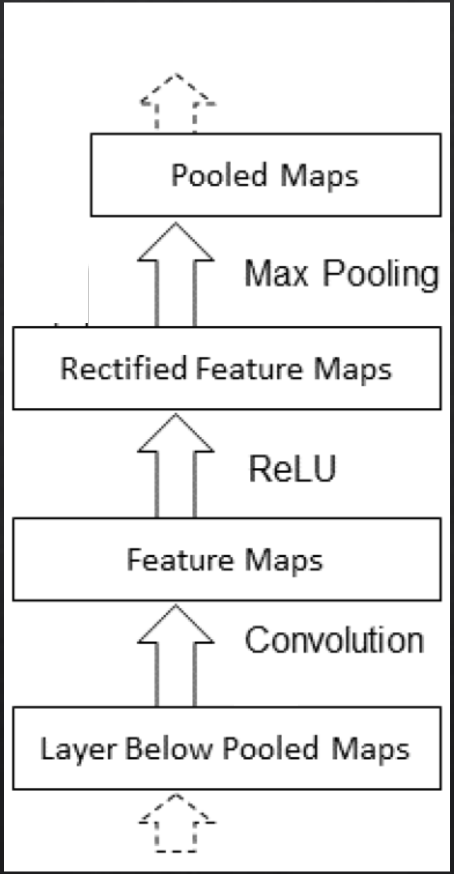
So far, finding correlations between a set of images and activations



What input pattern originally caused a given activation in the feature maps?

Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14

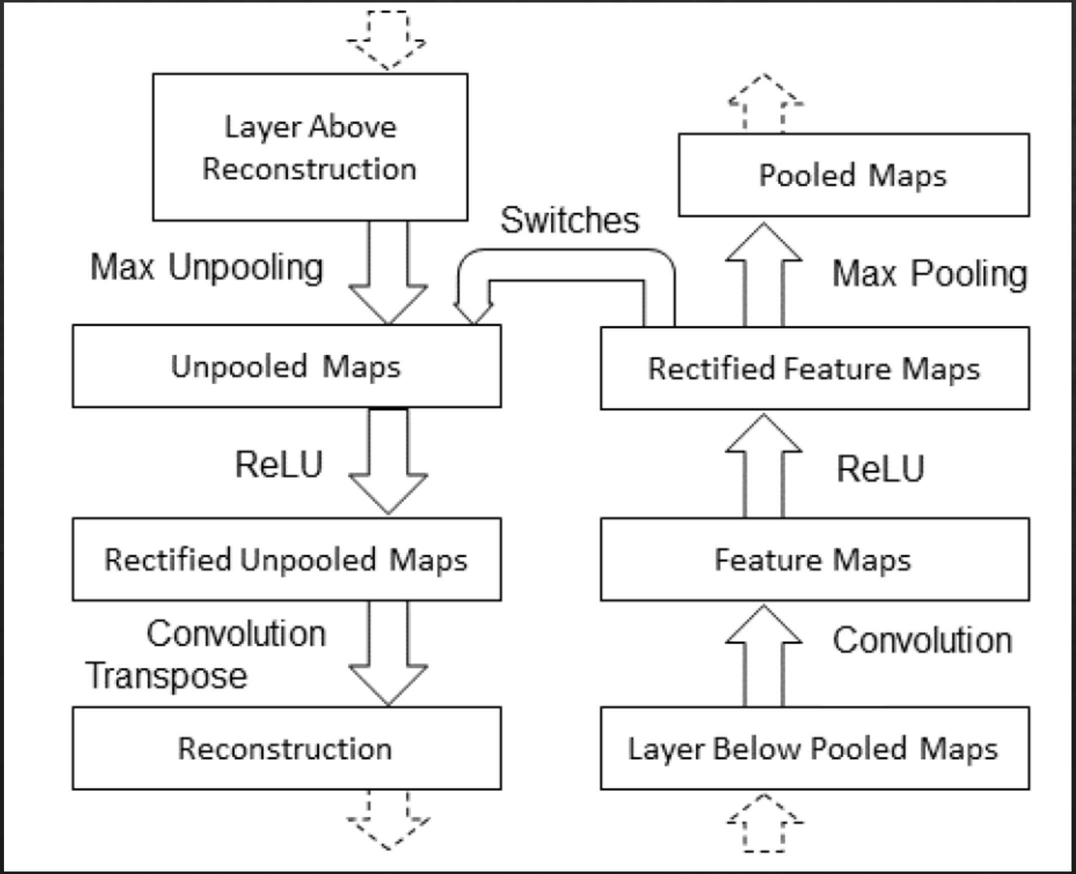
Convnet



How to project the activations back to the input pixel space?

Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14

Deconvnet Convnet



- Deconvnet aims to project the activations back to the input pixel space
- Invert convnet by
  - Unpooling
  - (Un)rectification
  - Convolution transpose

Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14

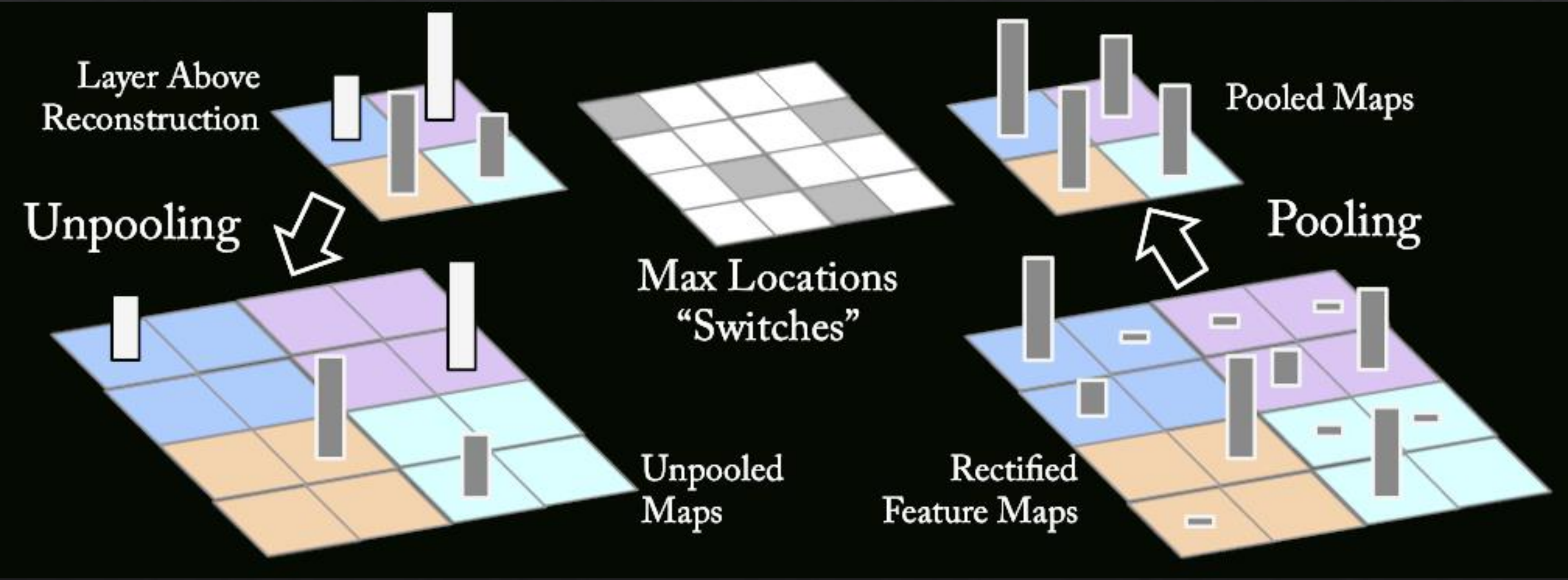
Question

Is max pool operation invertible?

$$y = \mathrm{maxpool}(x)$$

$$x ?= (\mathrm{maxpool})^{-1}(y)$$

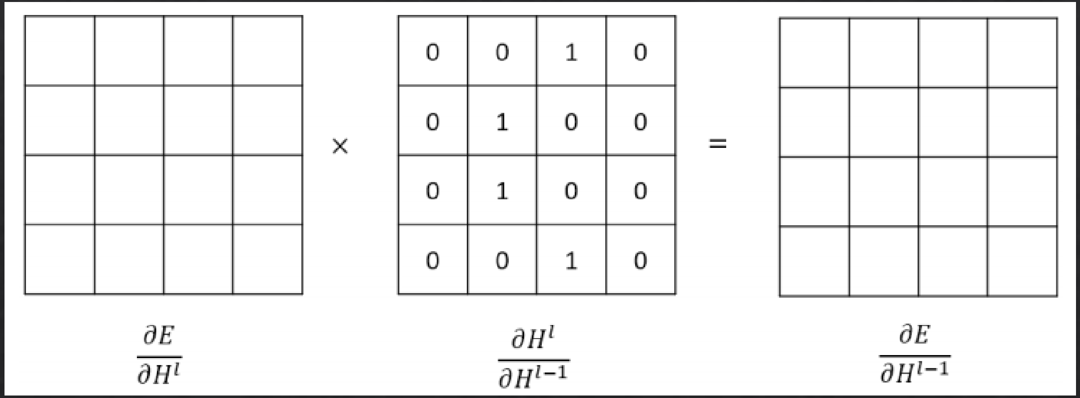
Unpooling



Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14

Unpooling

Relation to backprop (see lecture 8)



- E is loss function
- $\frac{\partial E}{\partial H_{l-1}} = \frac{\partial E}{\partial H_l} \frac{\partial H_l}{\partial H_{l-1}}$

- Unpooling corresponds to backprop of maxpooling

Unrectification (UnReLU)

$$H_{l-1} \longrightarrow \boxed{\text{ReLU}} \longrightarrow H_l$$

$$H_l = \max(H_{l-1}, 0)$$

$$R_{l-1} \longleftarrow \boxed{\text{Un-ReLU}} \longleftarrow R_l$$

$$R_{l-1} = \max(R_l, 0)$$

Relation to backpropagation

$$\frac{\partial E}{\partial H_{l-1}} = \frac{\partial E}{\partial H_l} \cdot \mathbf{1}(H_l > 0)$$

UnReLU does not utilise $R_l \cdot \mathbf{1}(R_l > 0)$ but $\max(R_l, 0)$

Transpose convolution (deconvolution?)

$$H_{l-1} \xrightarrow{\;conv\;} H_l$$

<u>Convolution</u>

$$H_l = conv(H_{l-1}, W_l)$$

$$R_{l-1} \xleftarrow{\;t.\,conv\;} R_l$$

<u>Transpose convolution</u>

$$R_{l-1} = conv(R_l, W_l^T)$$

- It is not inverse convolution!
- Usually $conv(H_{l-1}, W_l) \neq conv(H_{l-1}, W_l^T)$

Relation to backprop (see lecture 8)

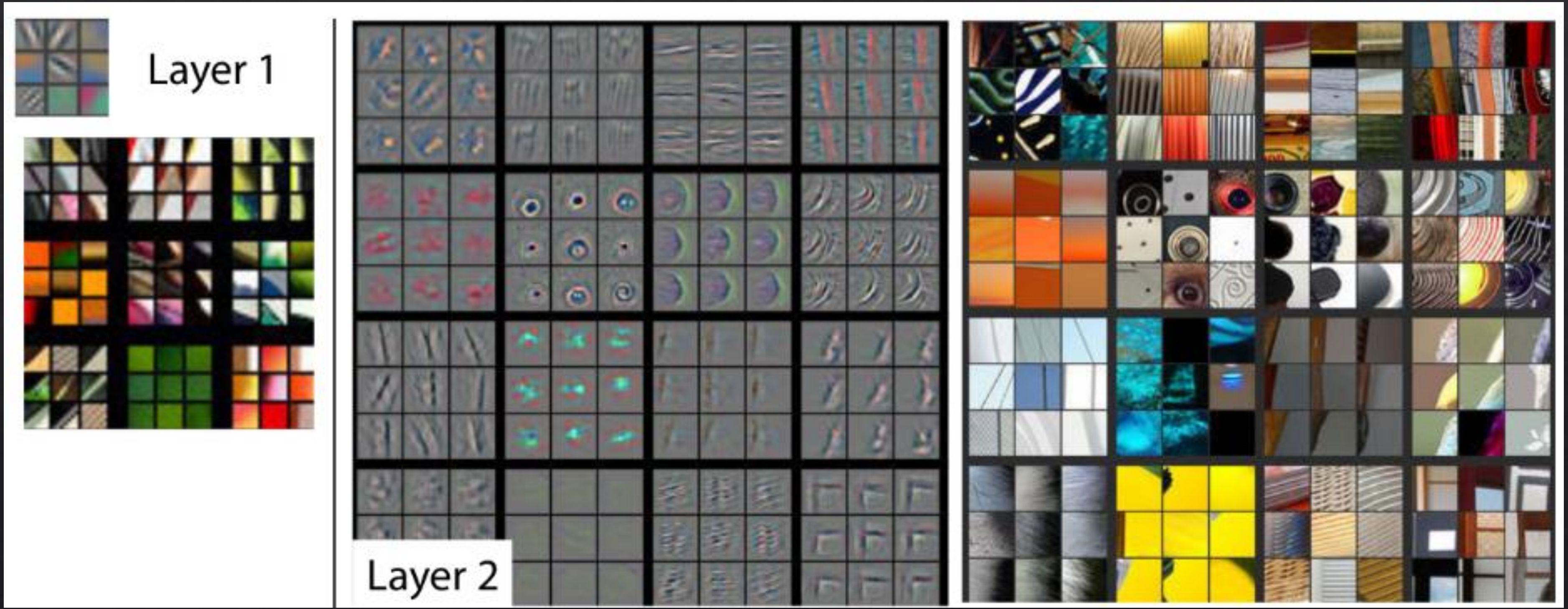
$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & \frac{\partial E}{\partial h_{11}^l} & \frac{\partial E}{\partial h_{12}^l} & 0 \\ 0 & \frac{\partial E}{\partial h_{21}^l} & \frac{\partial E}{\partial h_{22}^l} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} w_{22}^l & w_{21}^l \\ w_{12}^l & w_{11}^l \end{bmatrix} = \begin{bmatrix} \frac{\partial E}{\partial h_{11}^{l-1}} & \frac{\partial E}{\partial h_{12}^{l-1}} & \frac{\partial E}{\partial h_{13}^{l-1}} \\ \frac{\partial E}{\partial h_{21}^{l-1}} & \frac{\partial E}{\partial h_{22}^{l-1}} & \frac{\partial E}{\partial h_{23}^{l-1}} \\ \frac{\partial E}{\partial h_{31}^{l-1}} & \frac{\partial E}{\partial h_{32}^{l-1}} & \frac{\partial E}{\partial h_{33}^{l-1}} \end{bmatrix}$$

Padded $\partial E/\partial H^l$ — Rotated W^l — $\partial E/\partial H^{l-1}$

Layer 1-2: Top-9 Patches

Top 9 activations are projected down to pixel space using deconvolutional net

Patches from validation images that give maximal activation of a given feature map

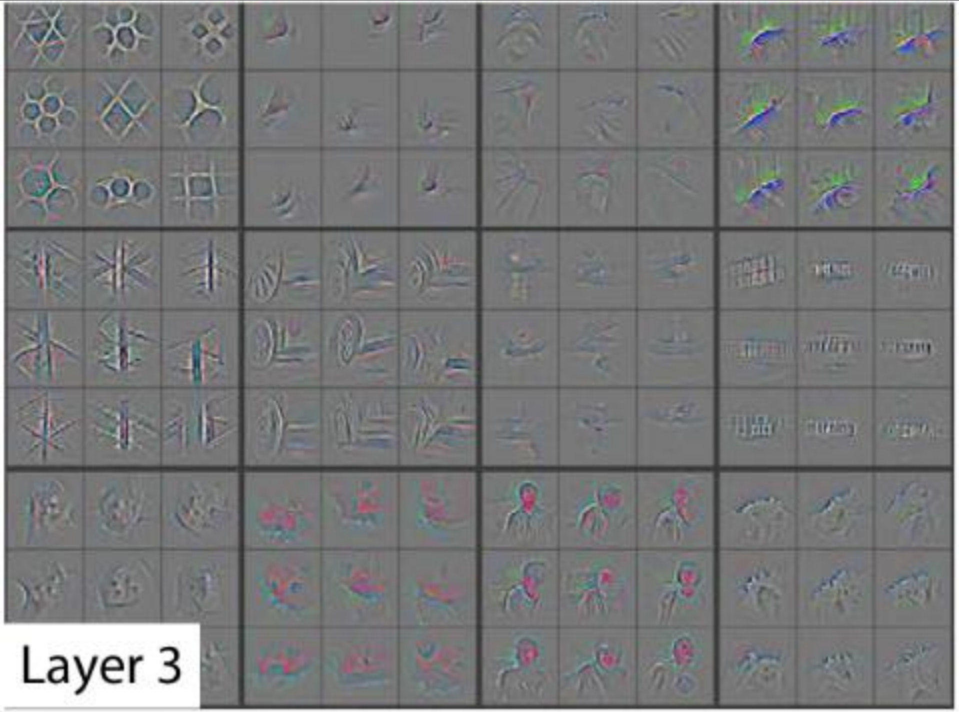


[Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14]()

Layer 3: Top-9 Patches

Top 9 activations are projected down to pixel space using deconvolutional net

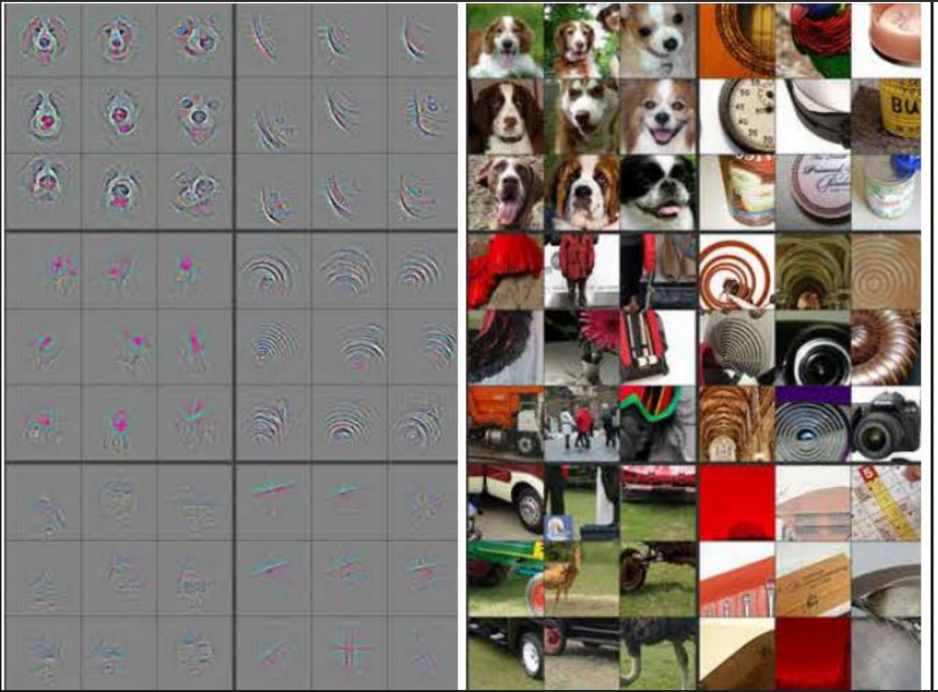
Patches from validation images that give maximal activation of a given feature map



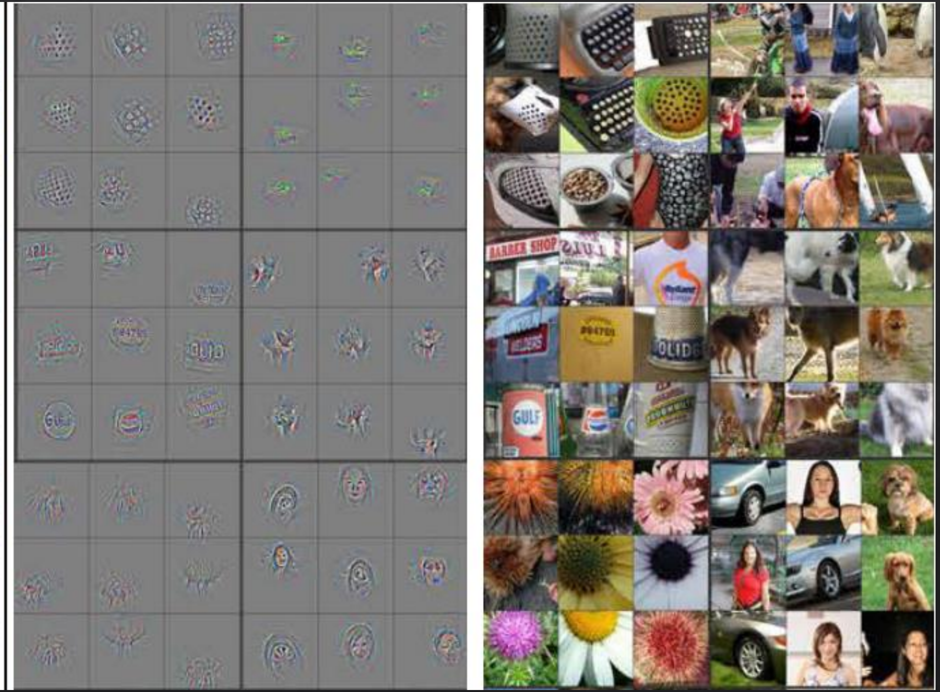
Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14

Layer 4-5: Top-9 Patches

Layer 4

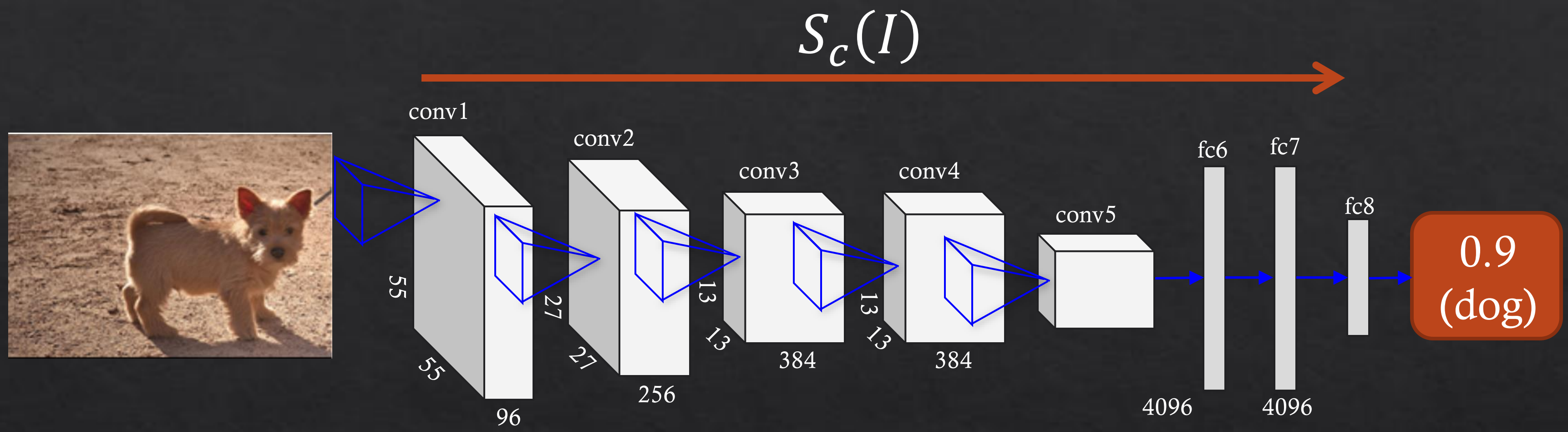


Layer 5



[Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14]()

4. Image specific saliency

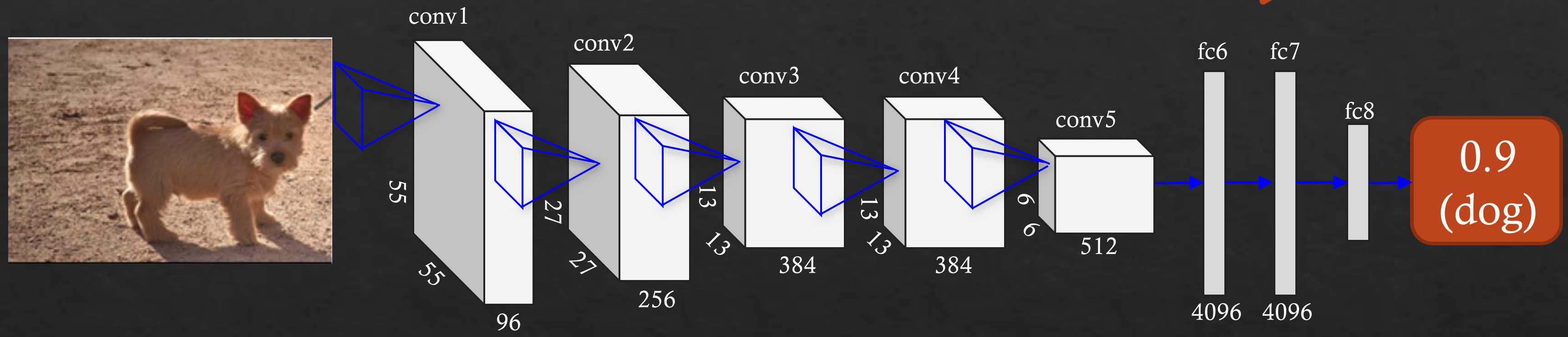


Which pixels matter most for the prediction?

Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS'14

4. Image specific saliency

$S_c(I)$



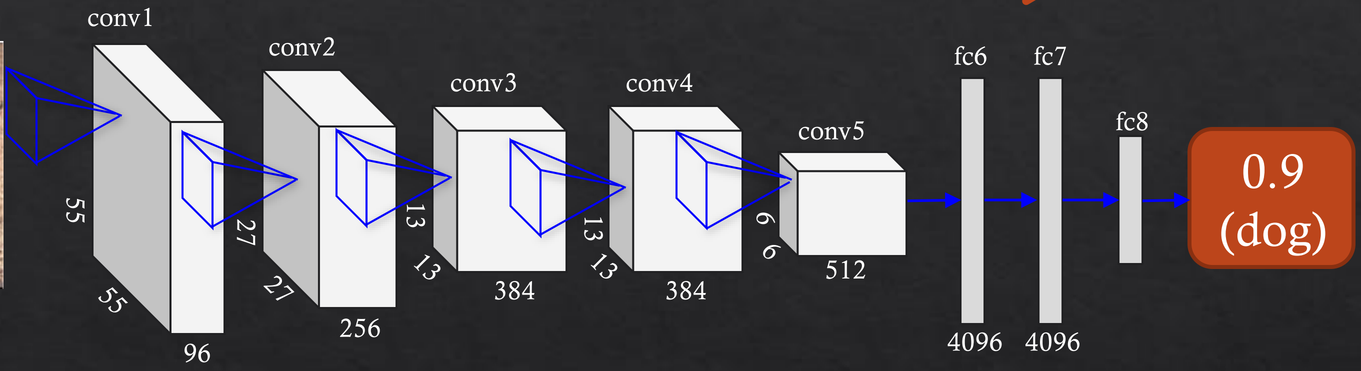
Question

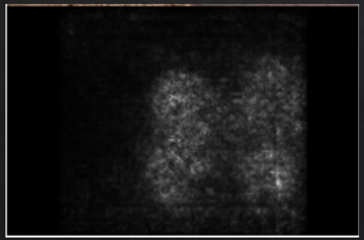
Can we calculate influence of each pixel on the class probability?

Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS'14

4. Image specific saliency

$S_c(I)$



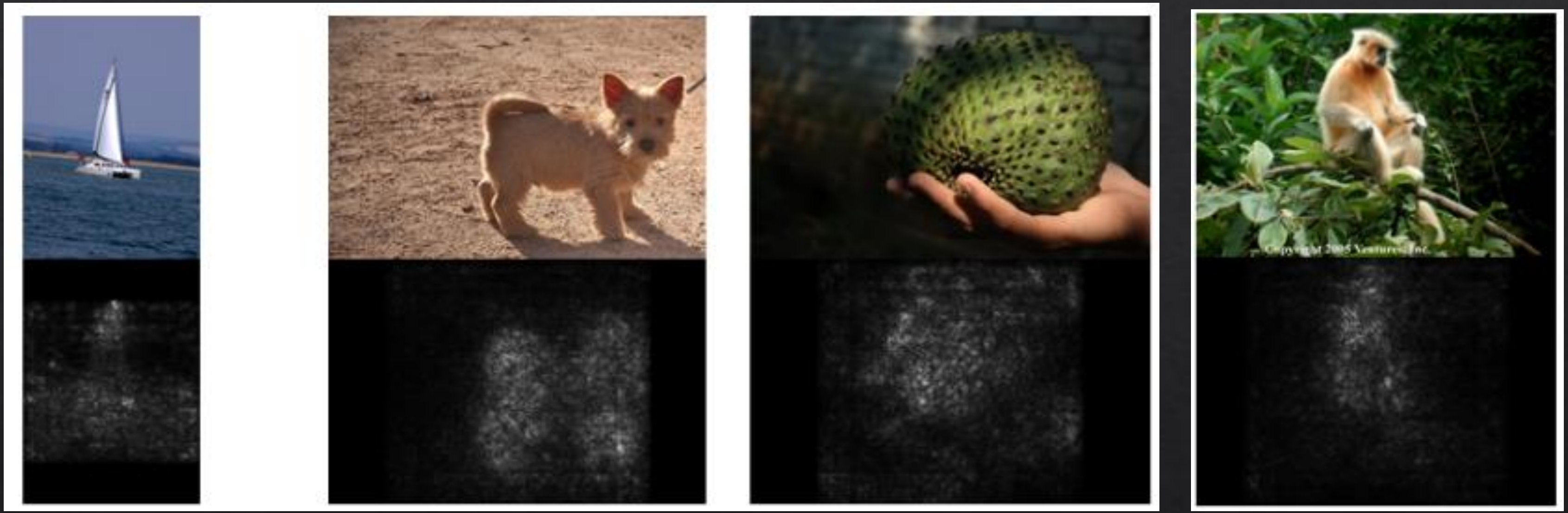


$$\frac{\delta S_c}{\delta I}$$

$|\delta S_c/\delta I|$

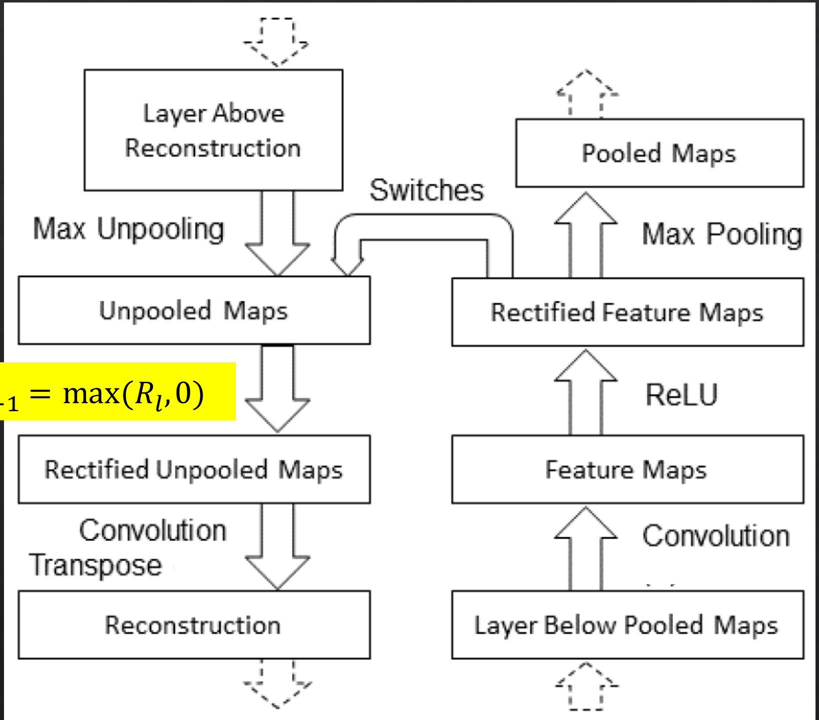
Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS'14

4. Image specific saliency

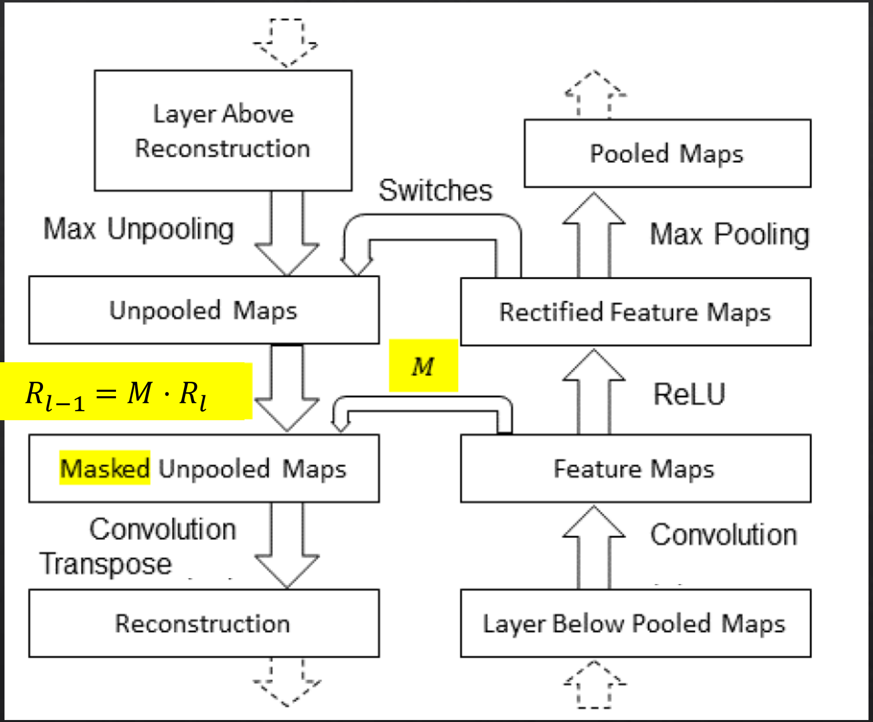


Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS’14

Deconv net vs Saliency net



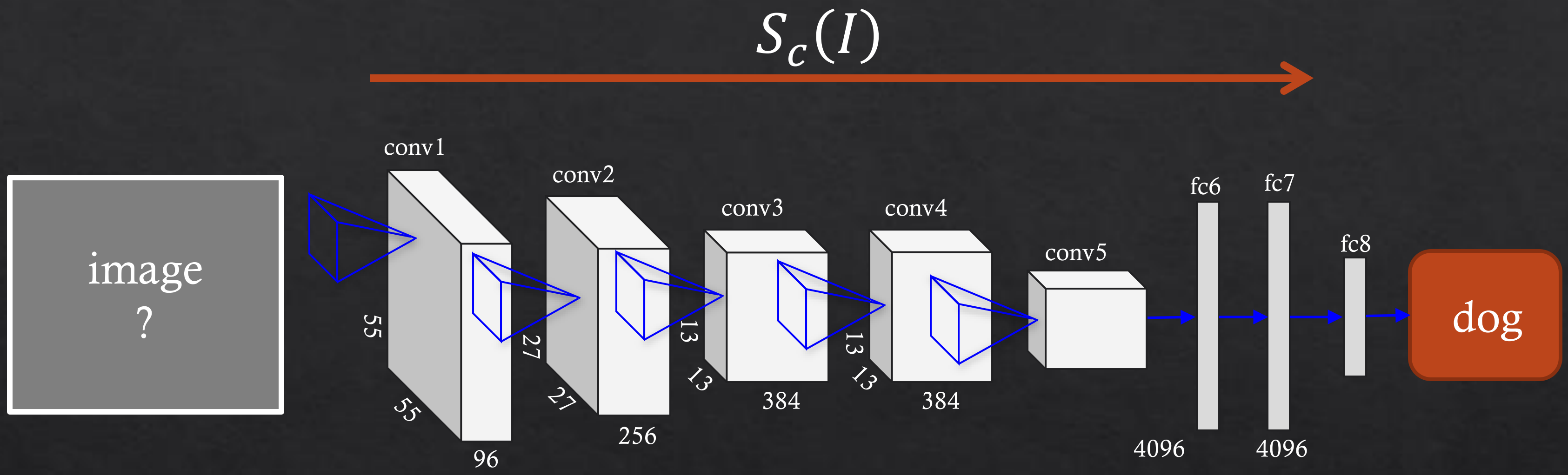
$R_{l-1} = \max(R_l, 0)$



$R_{l-1} = M \cdot R_l$

$$\boxed{M = \mathbf{1}(R_l > 0)}$$

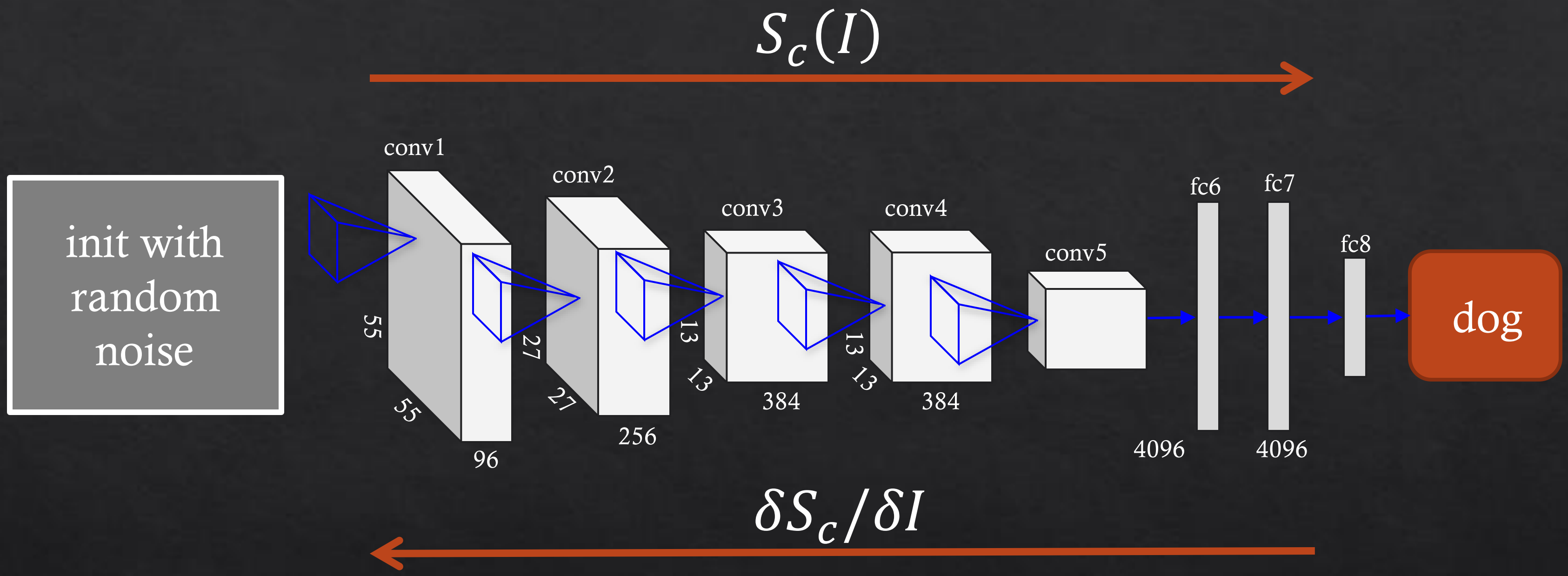
4. Generic class saliency maps



Can we generate an image that outputs high score for dog?

Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS'14

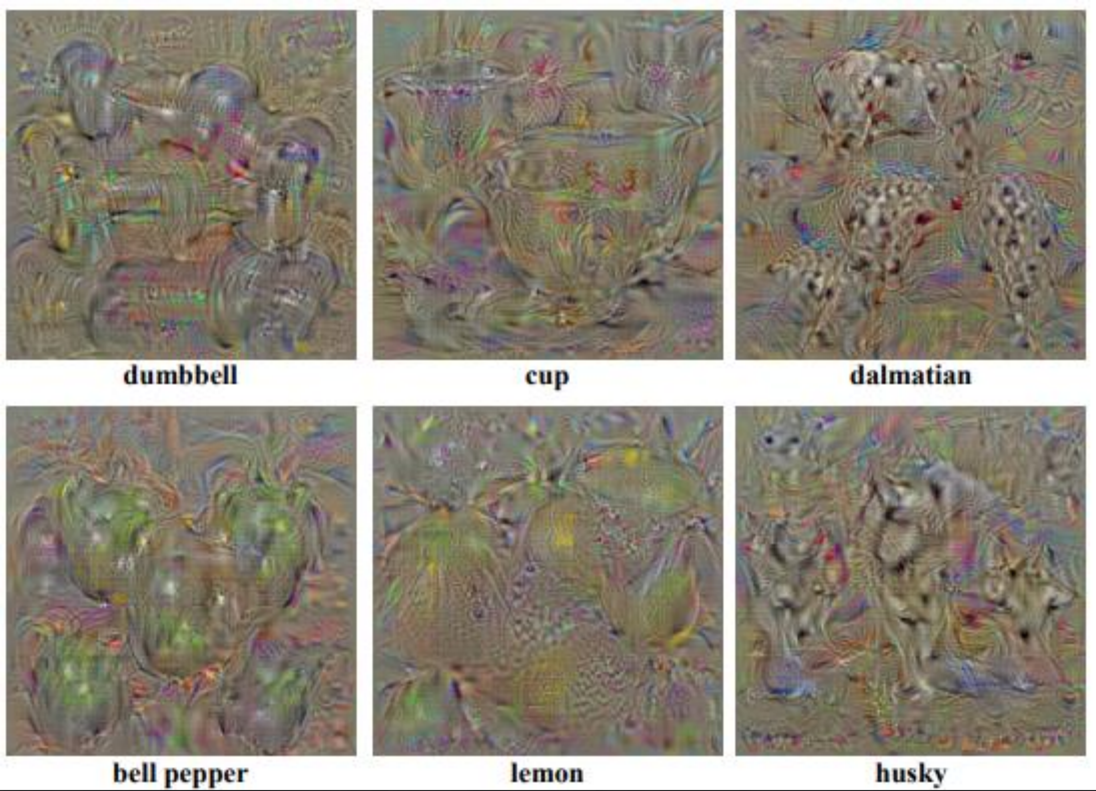
4. Generic class saliency maps



- $argmax_I \; S_c(I) - \lambda \parallel \mathrm{I} \parallel_2^2$
- Maximize "dogness" by modifying pixel values

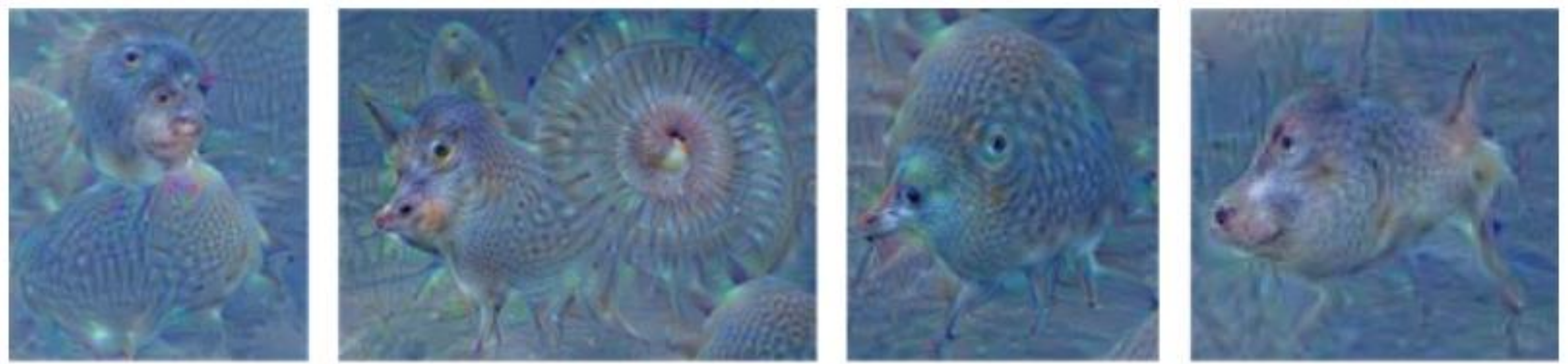
Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS'14

4. Generic class saliency maps



Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS'14

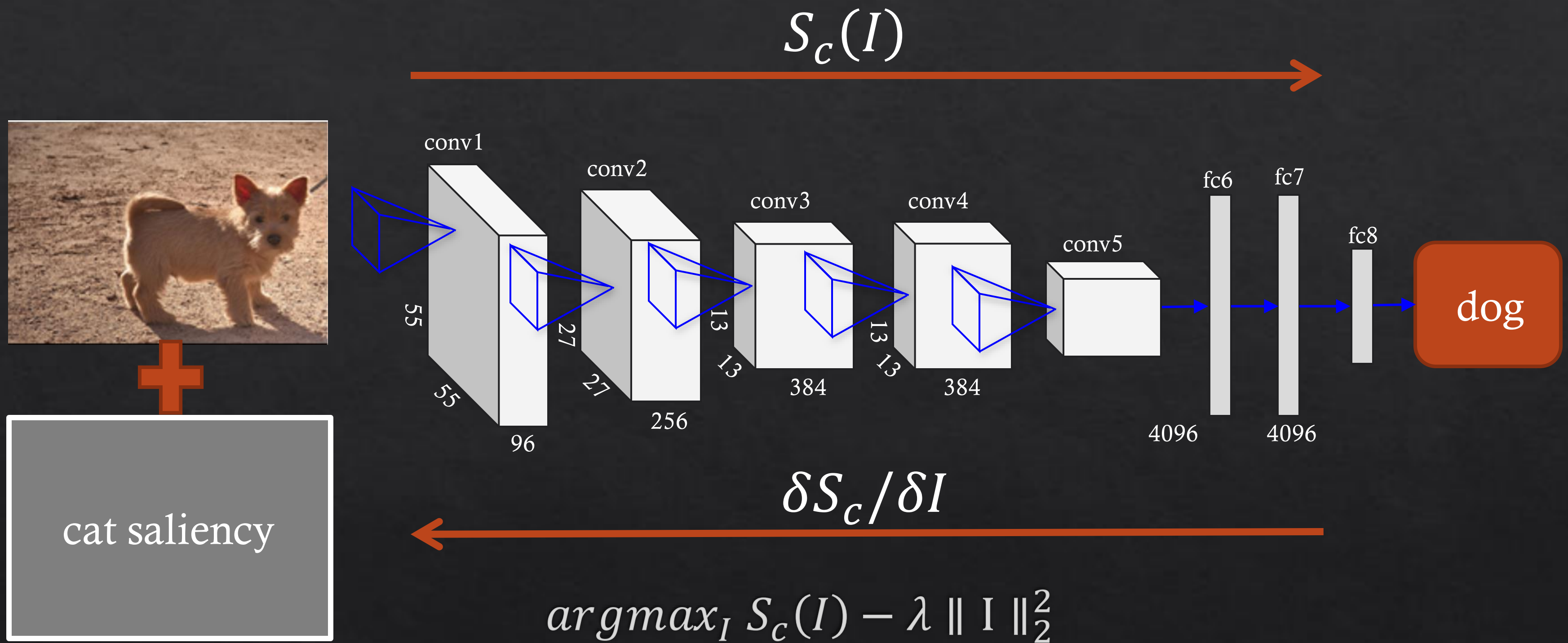
4. Image and generic class saliency (Deep dream)



"Admiral Dog!" "The Pig-Snail" "The Camel-Bird" "The Dog-Fish"

Mordvintsev et al, Inceptionism: Going Deeper into Neural Networks

4. Image and generic class saliency maps

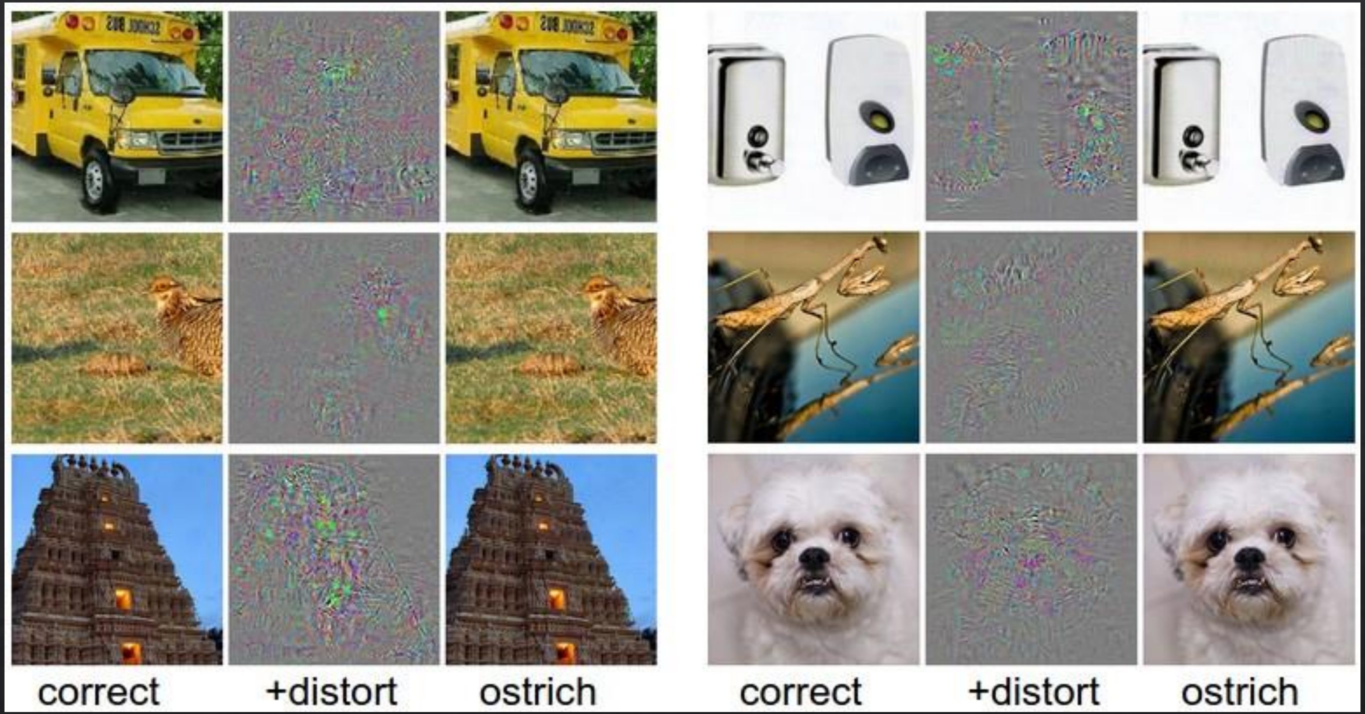


$$argmax_I \; S_c(I) - \lambda \parallel I \parallel_2^2$$

What happens if we add saliency for another class?

Simonyan, Vedaldi and Zisserman, Deep Inside Convolutional Networks, NIPS'14

Adversarial Examples



Problem common to any discriminative method!

[Szegedy et al., Intriguing properties of neural networks, ICLR'14]

Summary

Visualize CNNs

- Filters
- Highest activations
- Deconv network
- Saliency network
- Generating adversarial samples

Reading material

Recommended

- Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV'14

Extra

- Simonyan, Vedaldi, Zisserman, Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, ICLR'14
- Szegedy et al. Intriguing properties of neural networks, ICLR'14
- Nice summary of adversarial techniques by Karpathy
- Try to generate adversarial examples or interesting pictures!