Computational Systems Biology
Deep Learning in the Life Sciences

6.802 6.874 20.390 20.490 HST.506

Guest Lecturer: Brandon Carter

Prof. David Gifford

Lecture 5

February 20, 2020

Deep Learning Model Interpretation

Massachusetts Institute of Technology

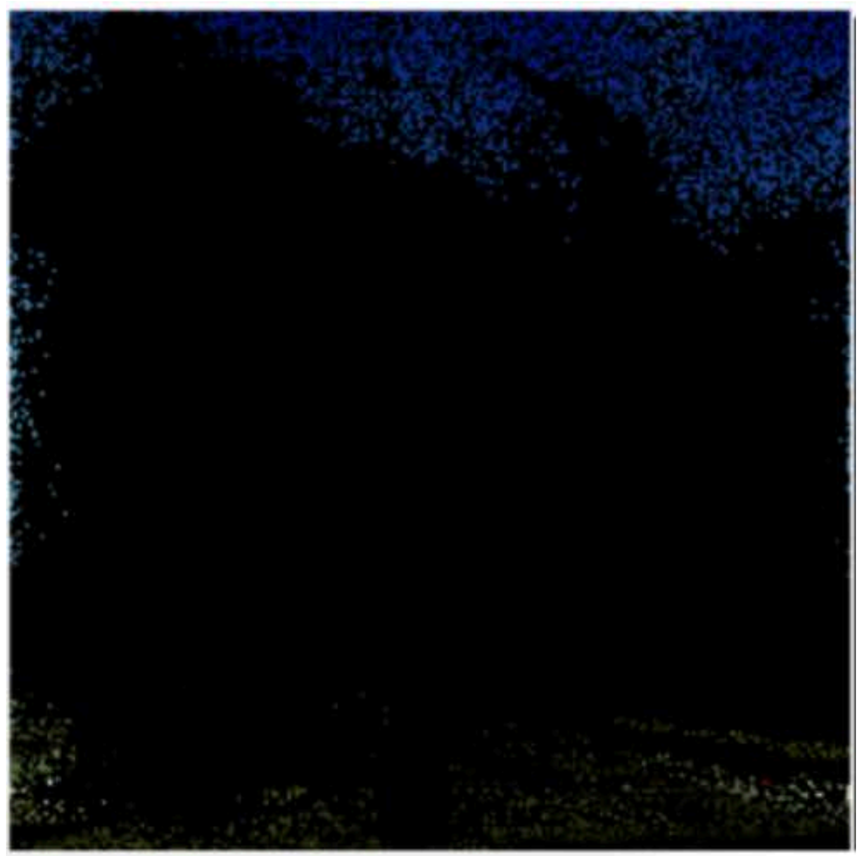
http://mit6874.github.io

What's on tap today!

- The interpretation of deep models
  - Black box methods (test model from outside)
  - White box methods (look inside of model)
  - Input *dependent* vs. input *independent* interpretations

Guess the image…

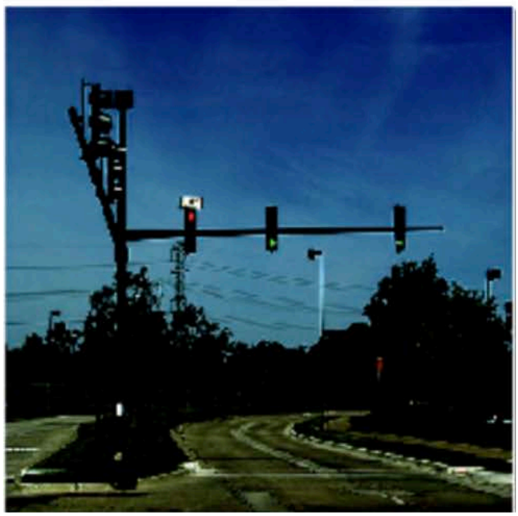
?



Guess the image…

traffic light

Guess the image…



traffic light
90% confidence

(InceptionResnetV2)

Why Interpretability?

- Adoption of deep learning has led to:
  - Large increase in predictive capabilities
  - Complex and poorly-understood black-box models

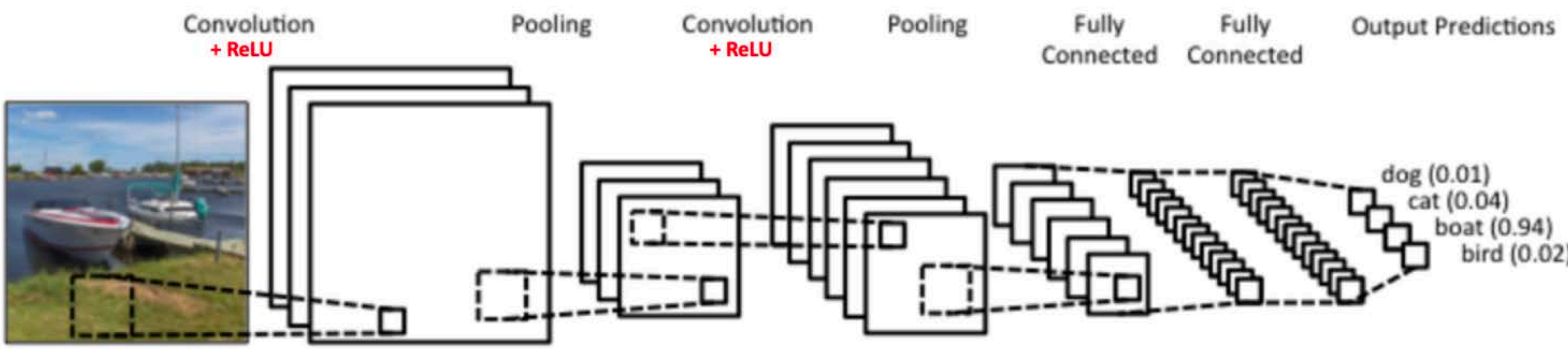
- Imperative that certain model decisions can be interpretably rationalized
  - Ex: loan-application screening, recidivism prediction, medical diagnoses, autonomous vehicles

- Explain model failures and improve architectures

- Interpretability is also crucial in scientific applications, where goal is to identify general underlying principles from accurate predictive models

How can we interpret deep models?

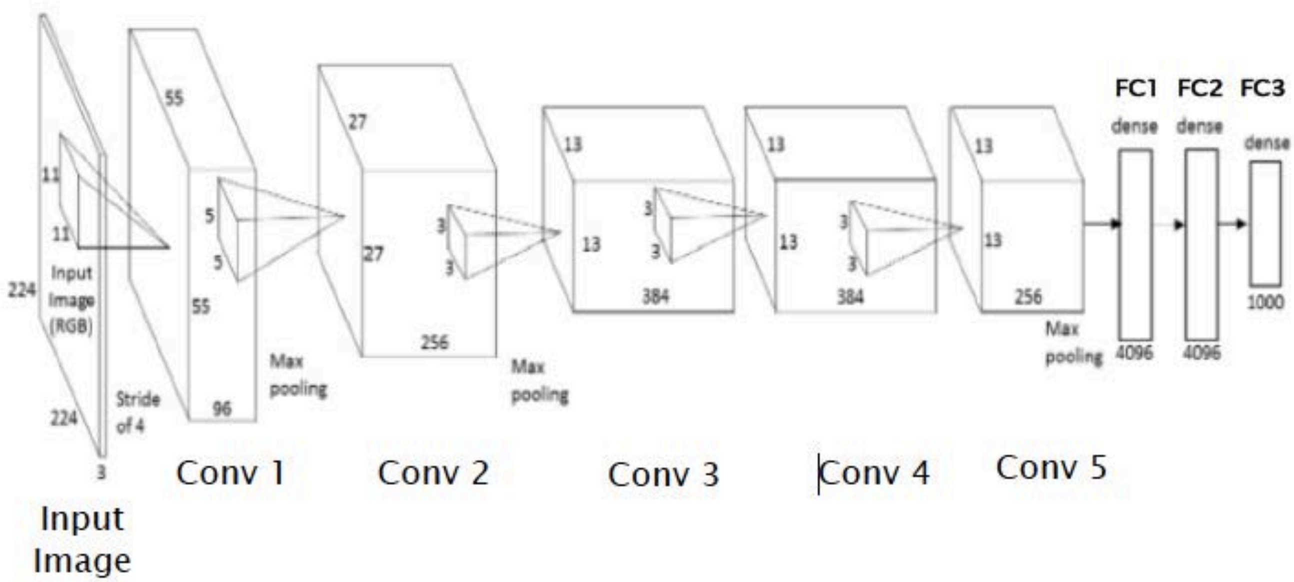
White Box Methods
(Look inside of model)

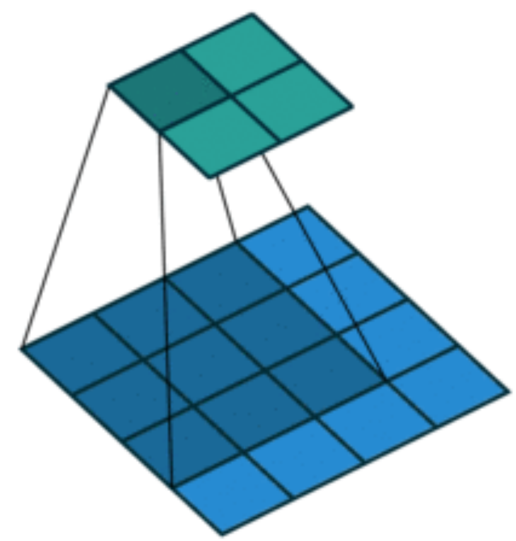


from https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Recall the ConvNet

AlexNet (Krizhevsky et al. 2012)



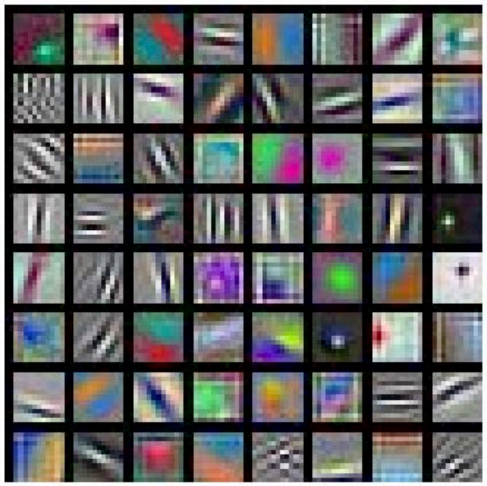


3x3 filter
4x4 input
2x2 output

https://srdas.github.io/DLBook/ConvNets.html

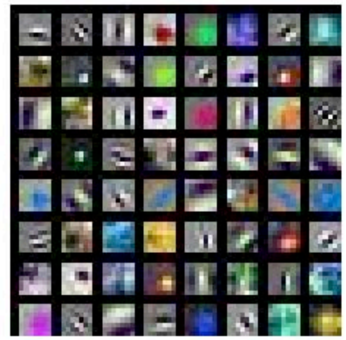
Visualizing filters

Only first layer filters are interesting and interpretable



AlexNet:
64 x 3 x 11 x 11

ResNet-18:
64 x 3 x 7 x 7



ResNet-101:
64 x 3 x 7 x 7

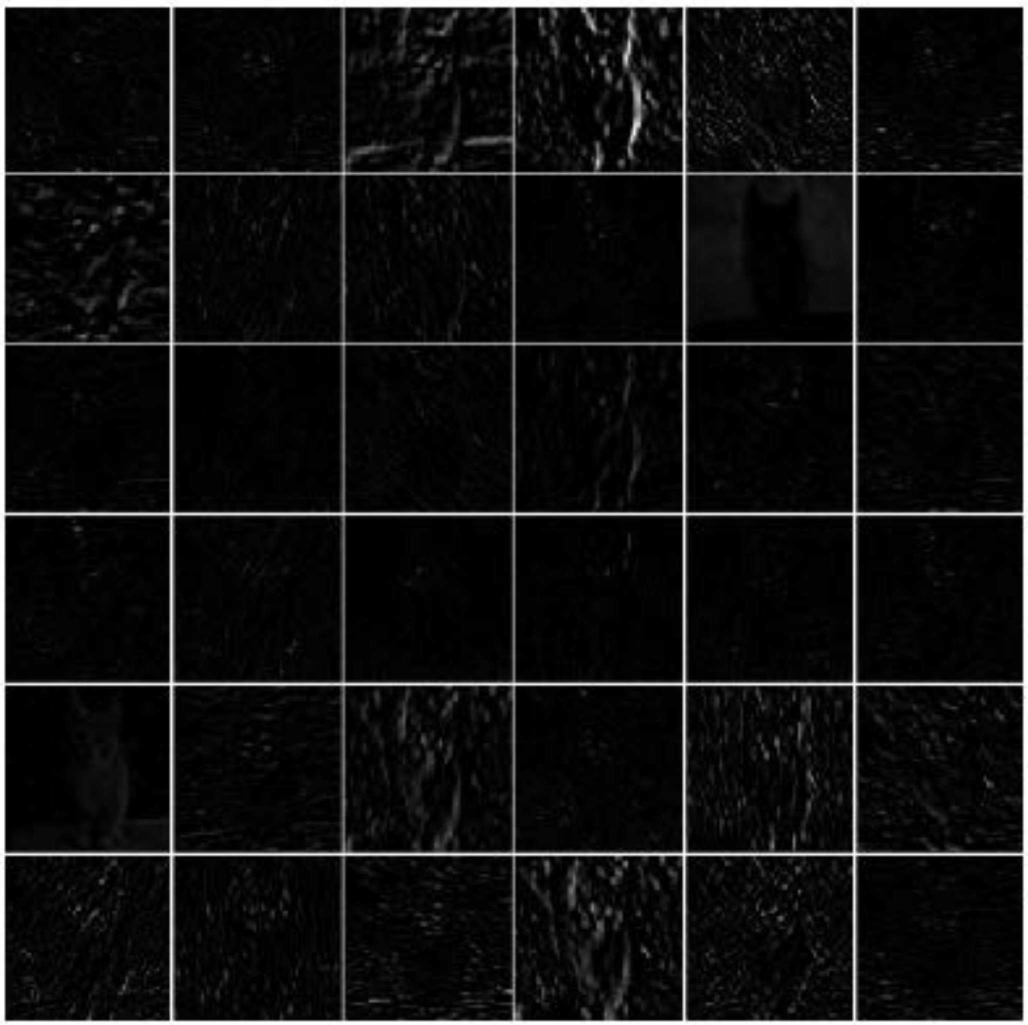
layer 1 weights

layer 3 weights

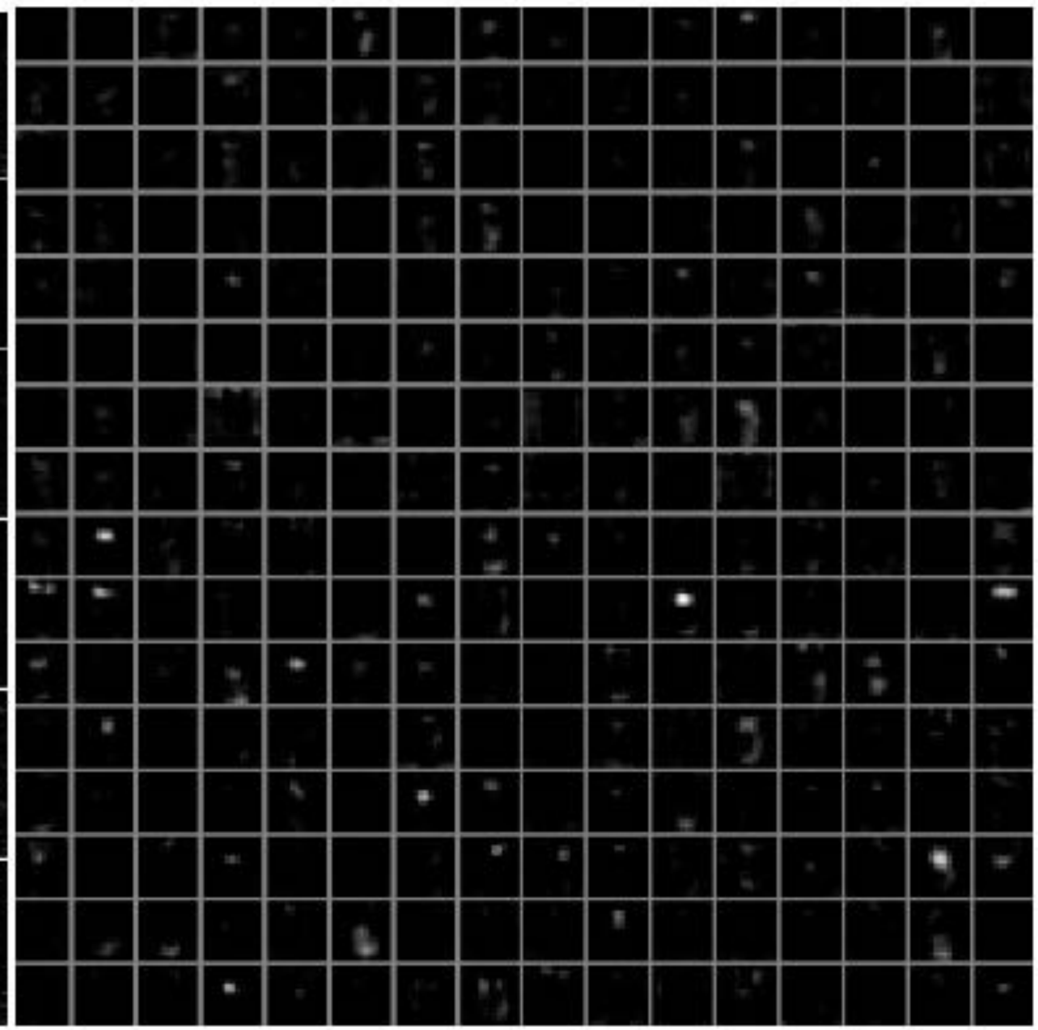
20 x 20 x 7 x 7

from ConvNetJS CIFAR-10 demo

Visualizing activations



First layer

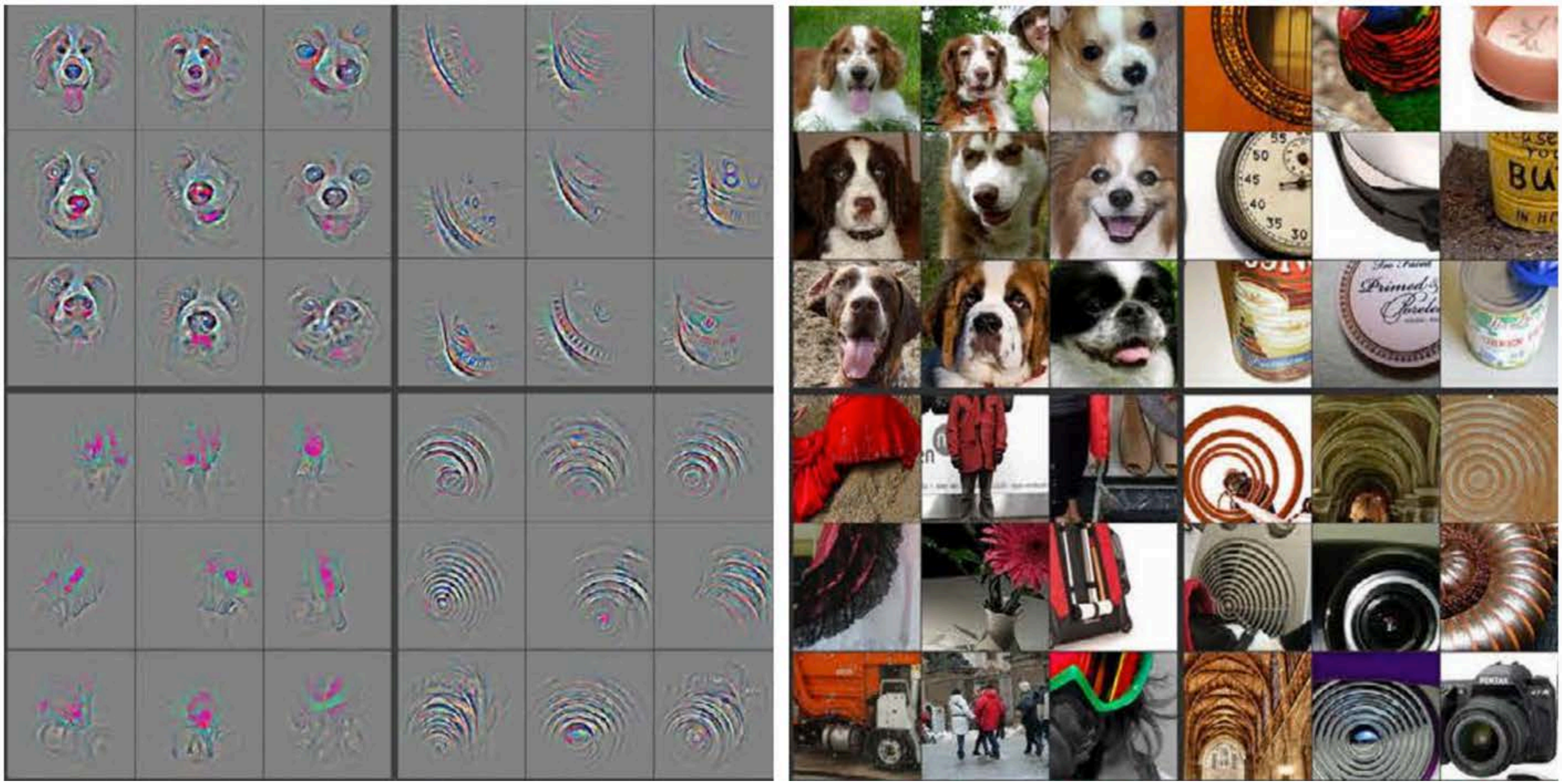


5th conv layer

Yosinski et al, "Understanding Neural Networks Through Deep Visualization", ICML DL Workshop 2014

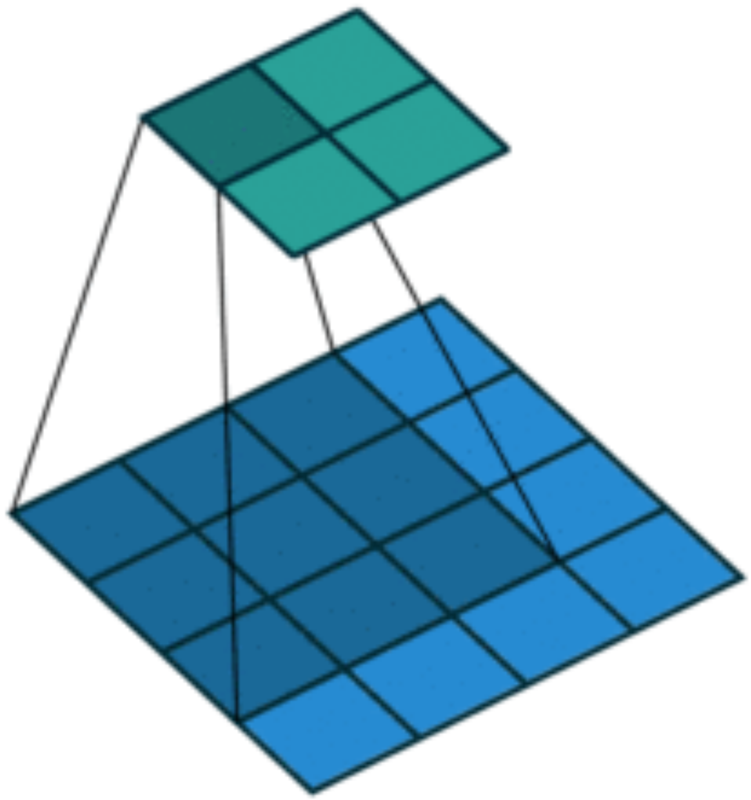
Deconvolute node activations

Deconvolutional neural net: A novel way to map high level activities back to the input pixel space, showing what input pattern originally caused a given activation in the feature maps



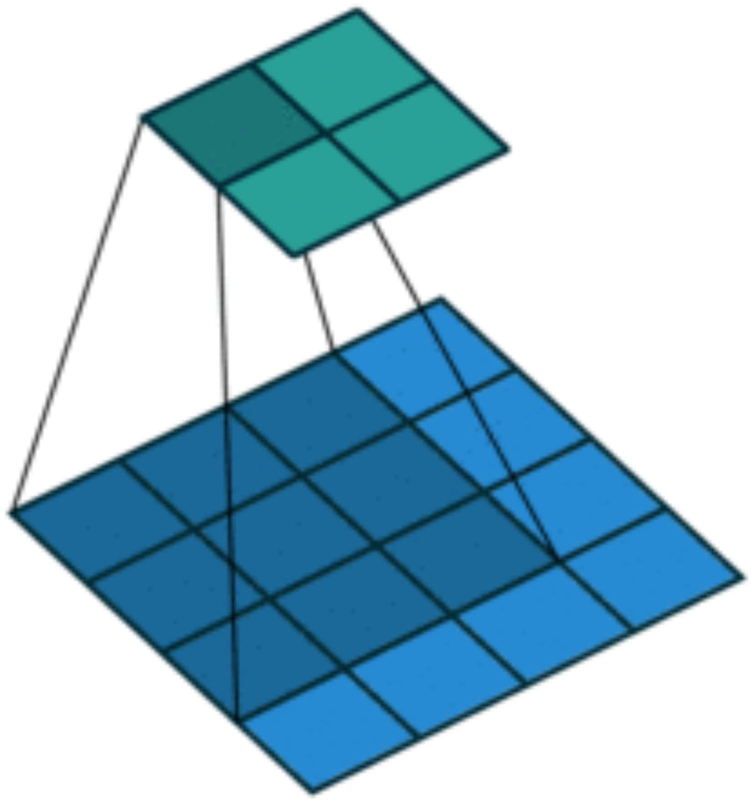
Zeiler et al., *Visualizing and Understanding Convolutional Networks*
Zeiler et al., *Adaptive Deconvolutional Networks for Mid and High Level Feature Learning*

Transposed convolution times received gradient is layer gradient

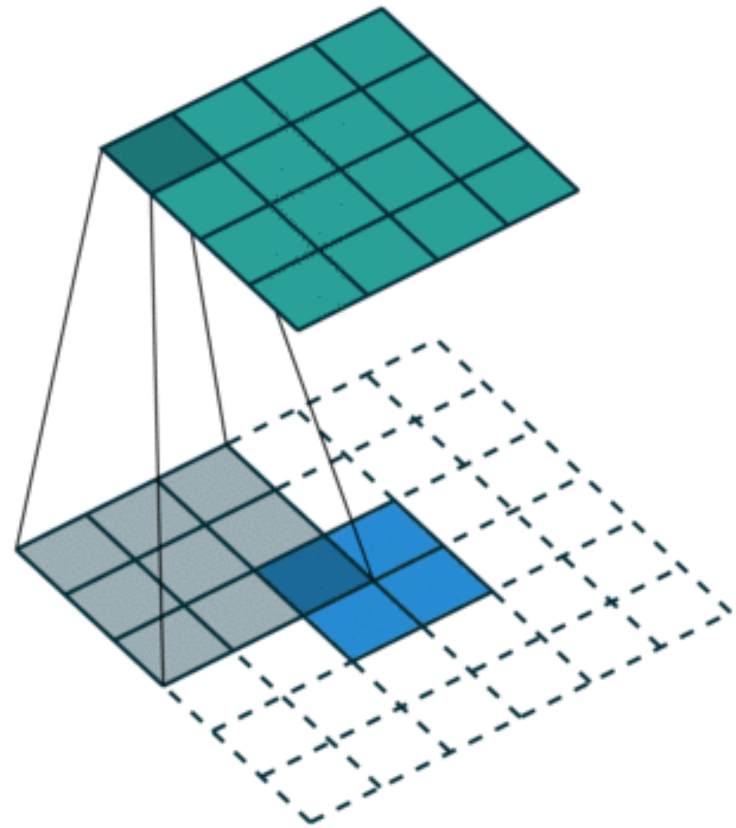


Convolution
3x3 filter on 4x4 input
2x2 output

Transposed convolution times received gradient is layer gradient

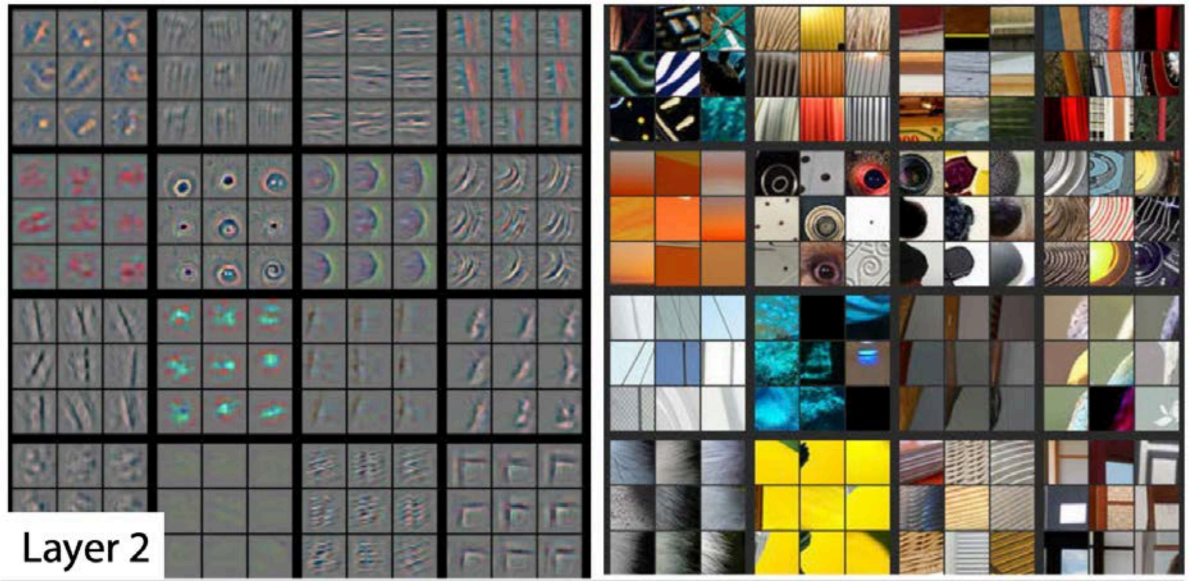


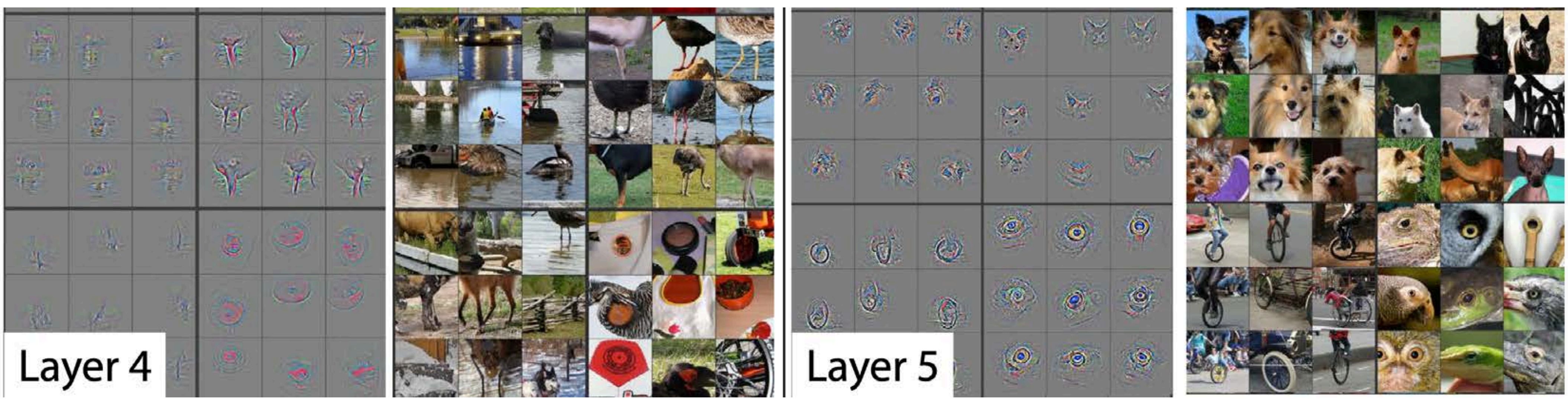
Convolution
3x3 filter on 4x4 input
2x2 output



Transposed Convolution
3x3 filter on 2x2 input
4x4 output

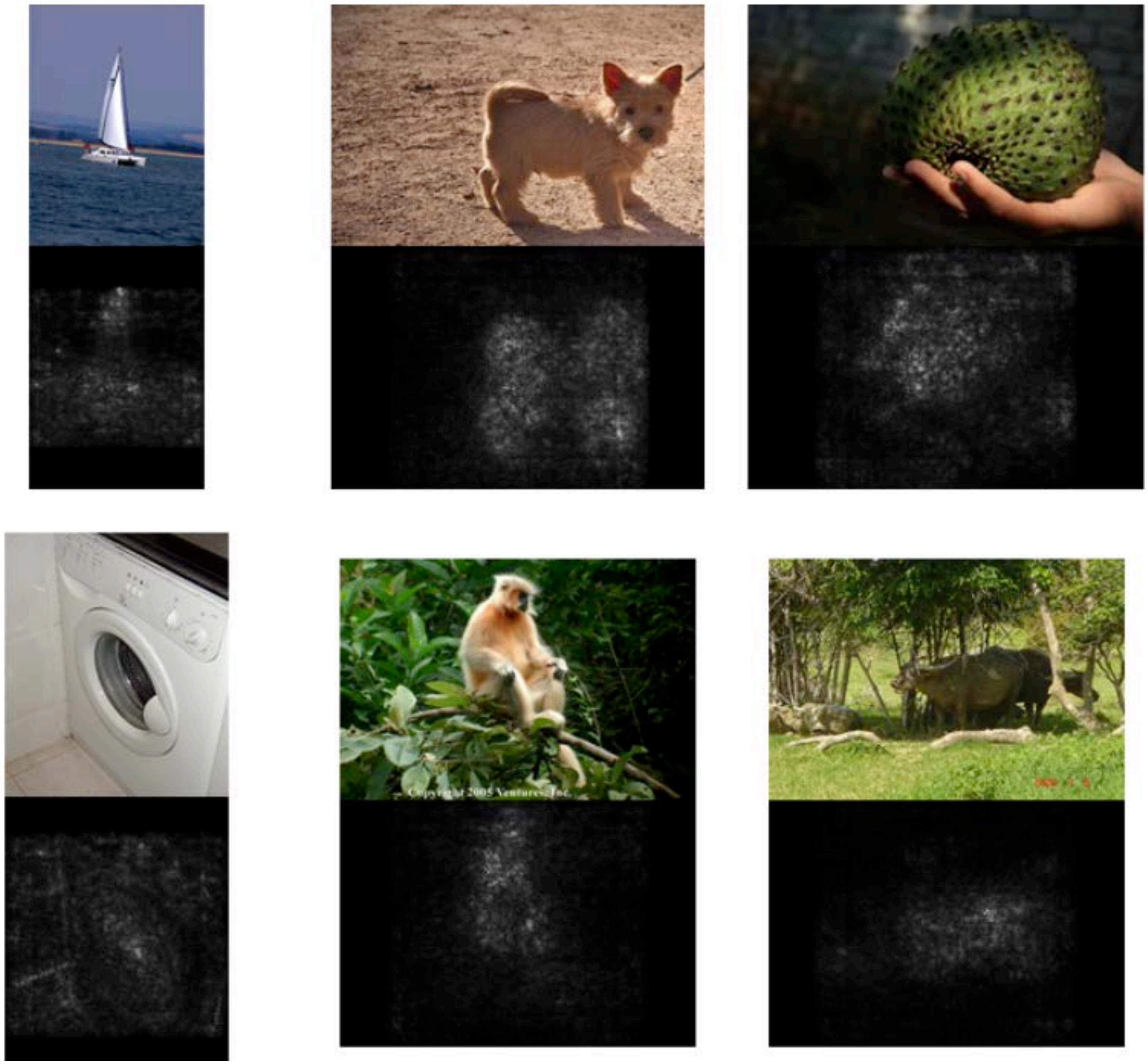
Deconvolute node activations





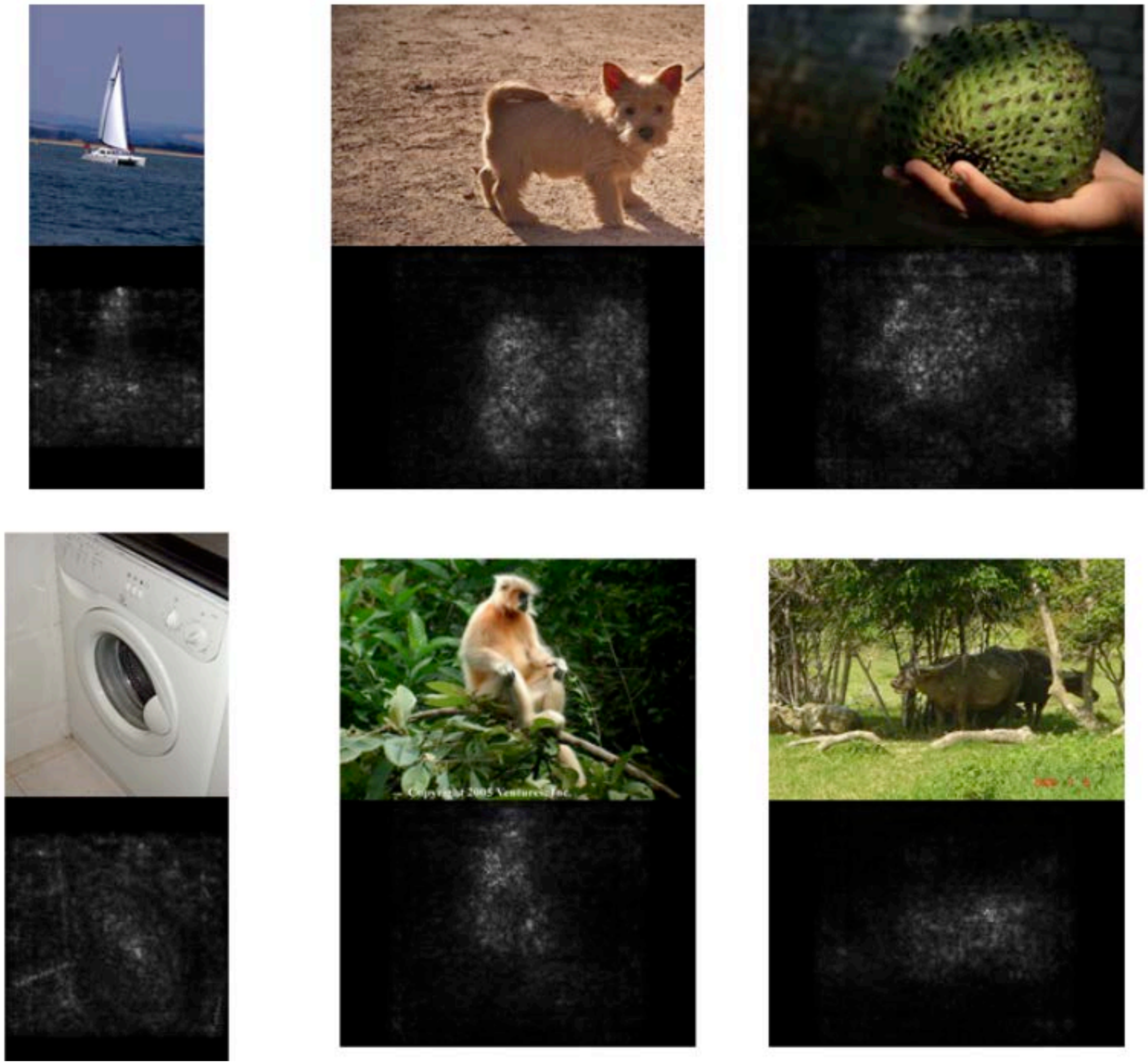
Zeiler et al., *Visualizing and Understanding Convolutional Networks*
Zeiler et al., *Adaptive Deconvolutional Networks for Mid and High Level Feature Learning*

Visualizing gradients: Saliency map



Simonyan et al., *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*

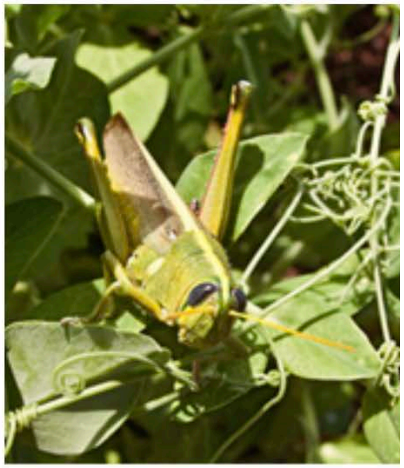
Visualizing gradients: Saliency map

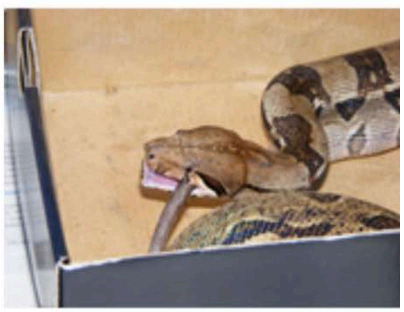


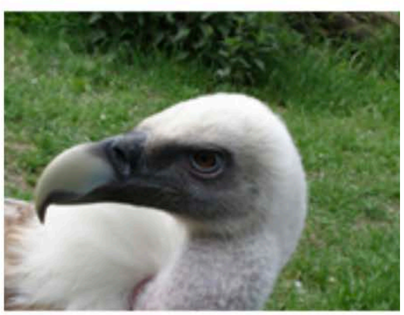
$$w = \left.\frac{\partial S_c}{\partial I}\right|_{I_0}$$

Simonyan et al., *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*

Application: Saliency maps can be used for object detection

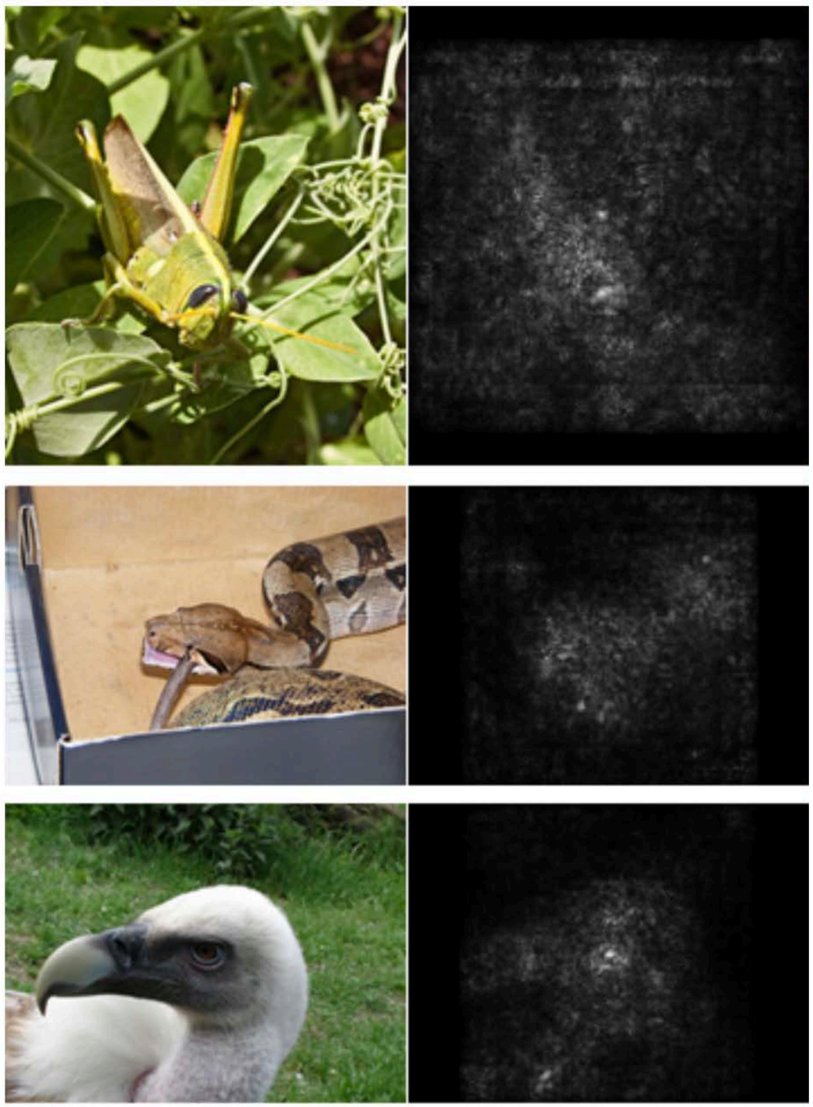






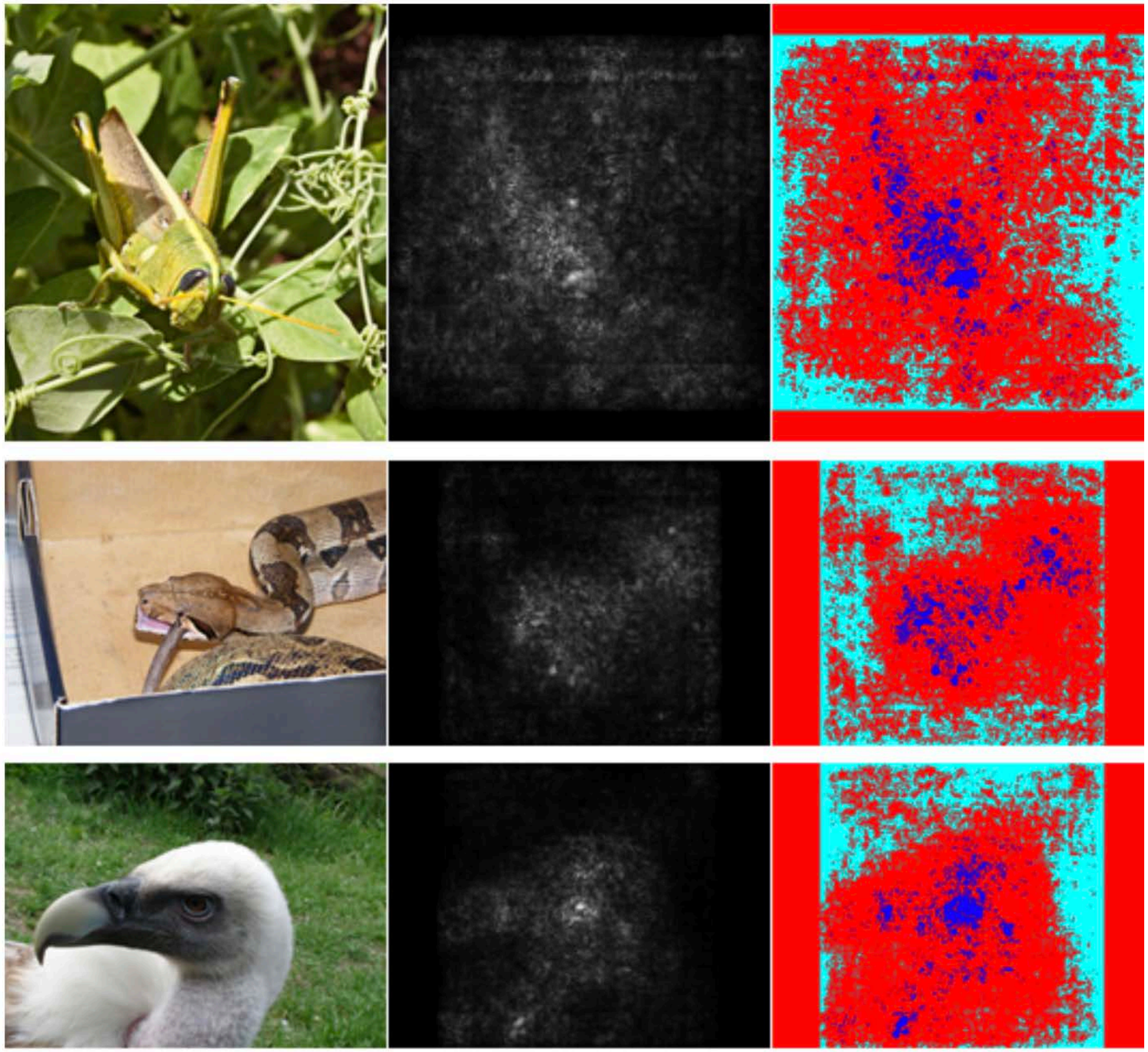
Simonyan et al., *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*

Application: Saliency maps can be used for object detection



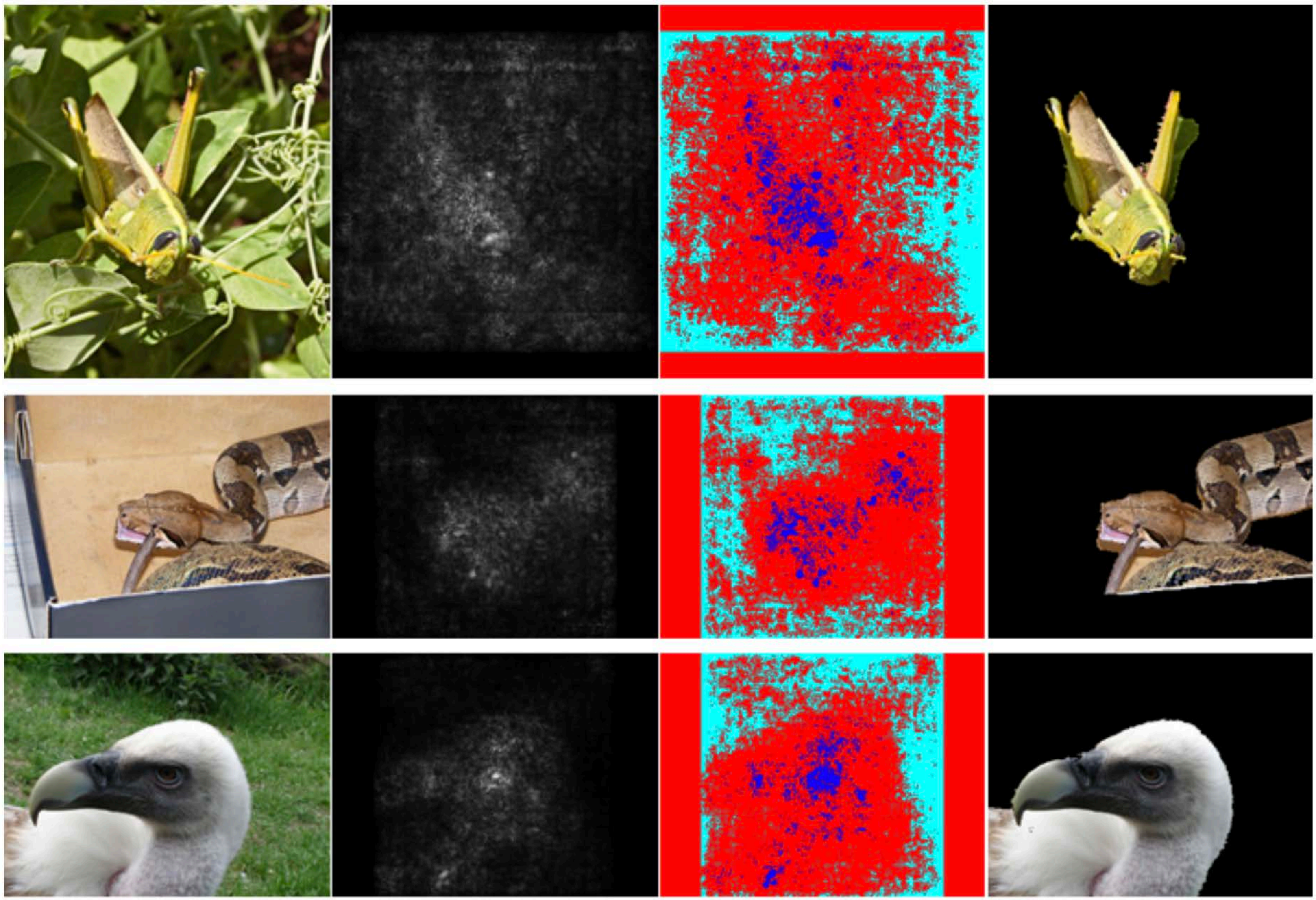
Simonyan et al., *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*

Application: Saliency maps can be used for object detection



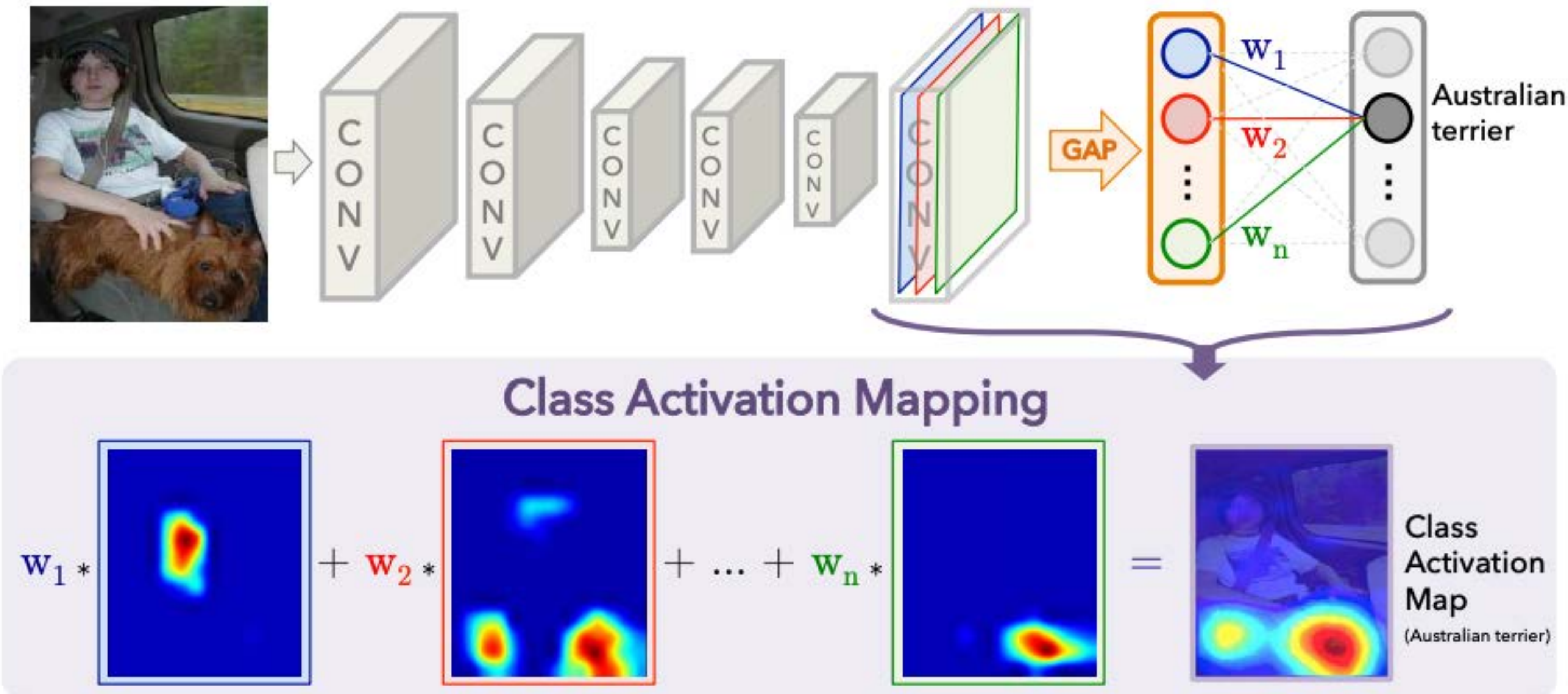
Simonyan et al., *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*

Application: Saliency maps can be used for object detection



Simonyan et al., *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*

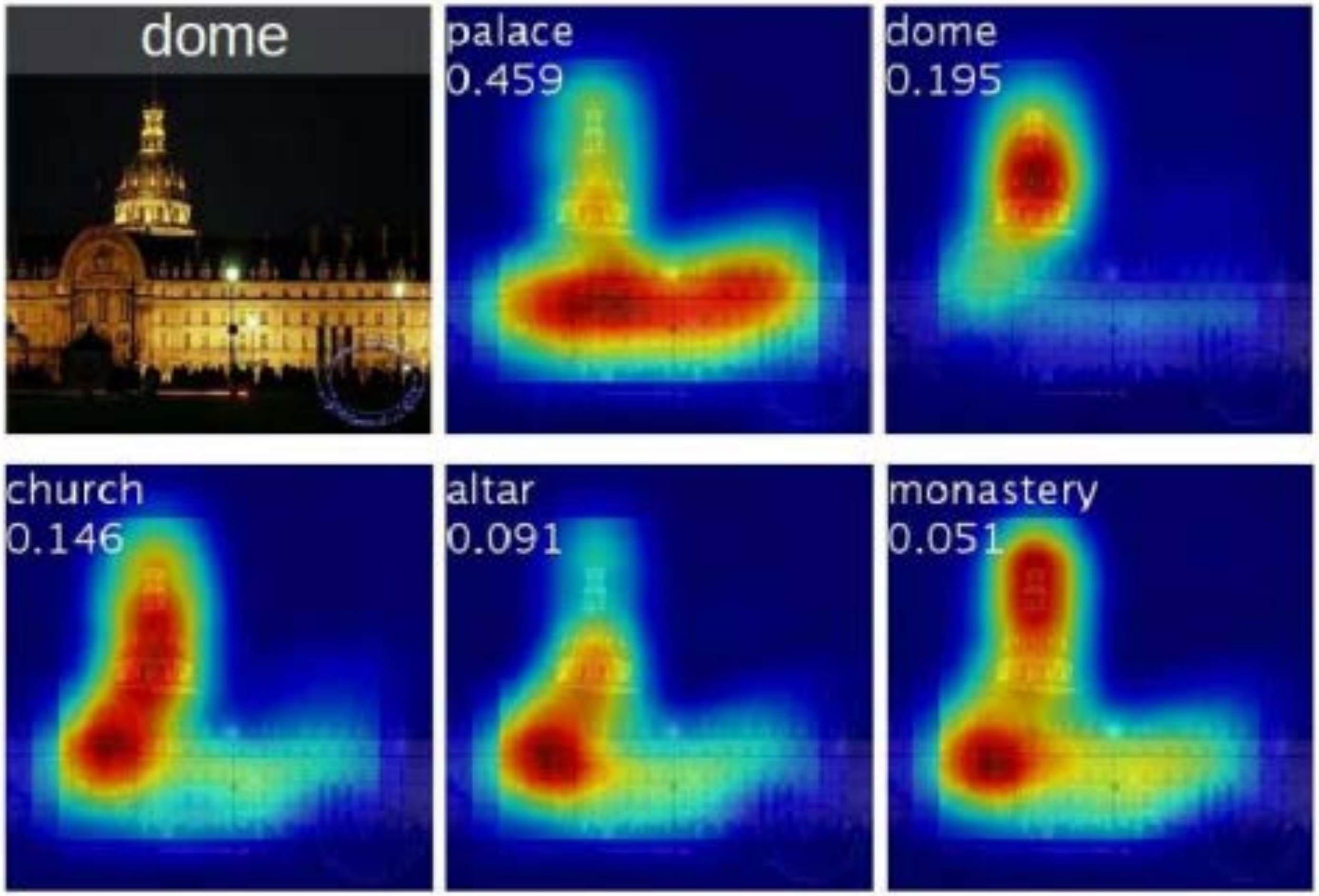
CAM: Class Activation Mapping



Use additional layer on top of the GAP (Global activation pooling) to learn **class specific** linear weights for each high level feature map and use them to weight the activations mapped back into input space.

Zhou et al., *Learning Deep Features for Discriminative Localization*

CAM: Class Activation Mapping



Use additional layer on top of the GAP (Global activation pooling) to learn **class specific** linear weights for each high level feature map and use them to weight the activations mapped back into input space.

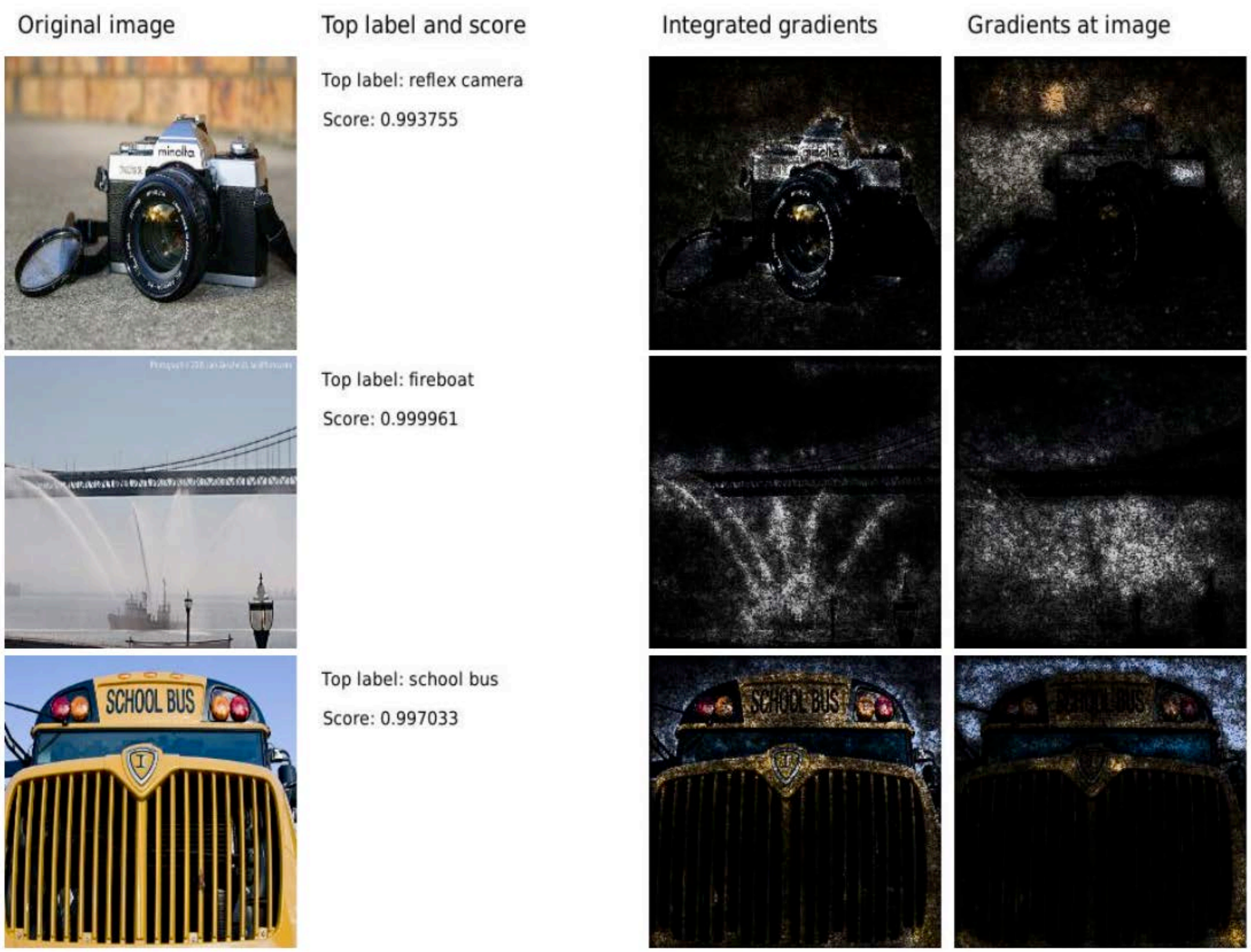
Zhou et al., *Learning Deep Features for Discriminative Localization*

Integrated Gradients

Given an input image x_i and a **baseline input** x_i' :

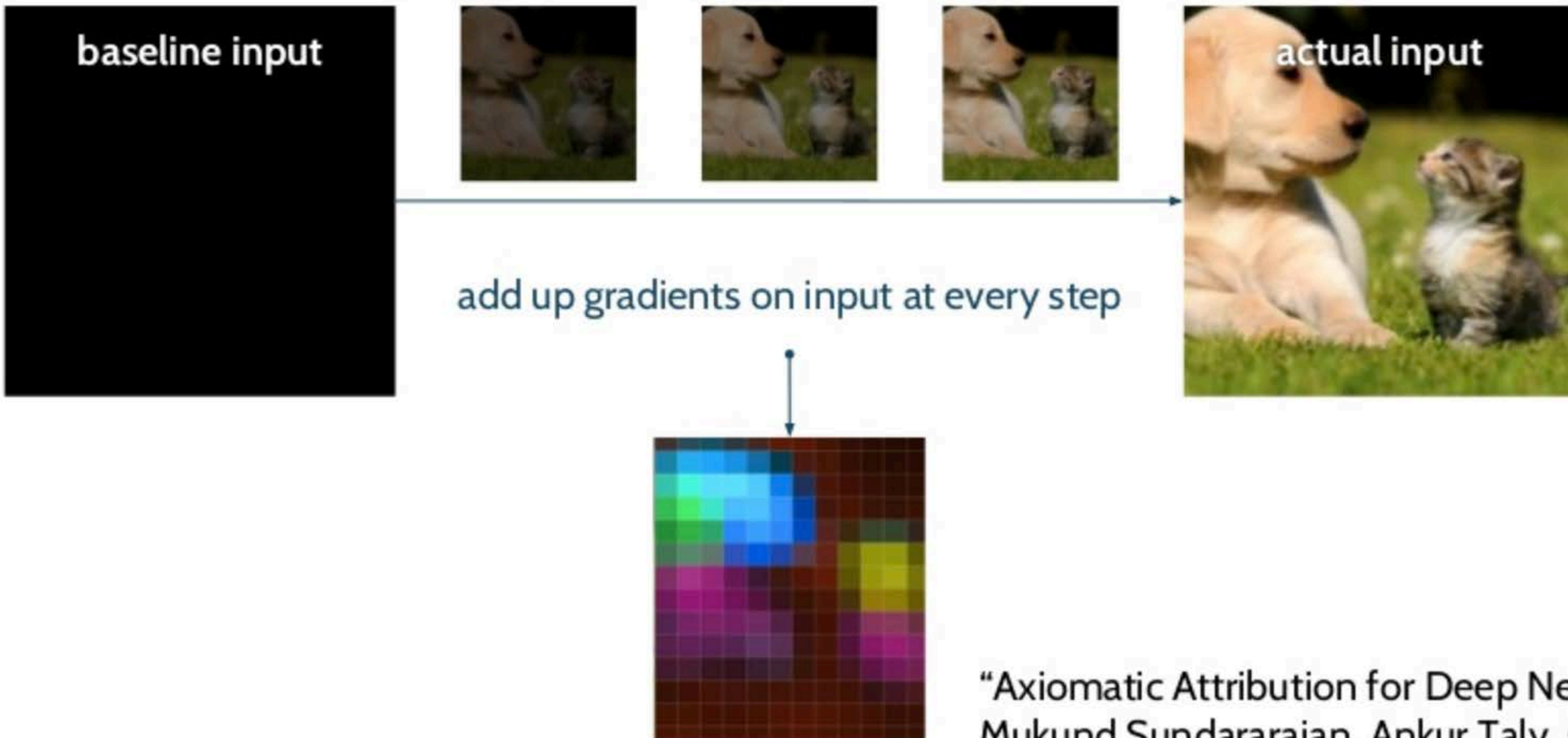
$$\text{IntegratedGrads}_i(x) ::= (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} \, d\alpha$$

$$\text{IntegratedGrads}_i^{approx}(x) ::= (x_i - x_i') \times \Sigma_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m} \times (x - x')))}{\partial x_i} \times \frac{1}{m}$$



Sundararajan et al., *Axiomatic Attribution for Deep Networks*

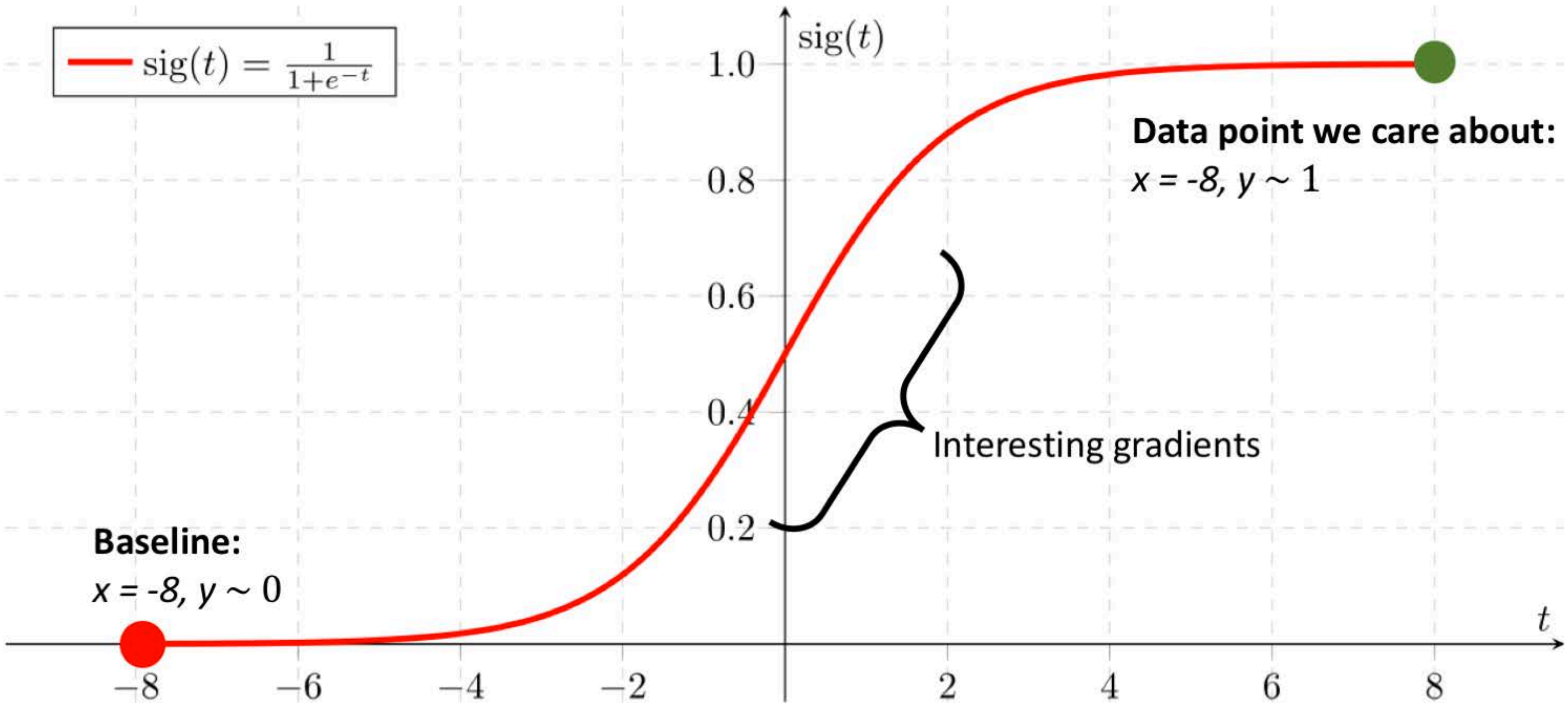
Integrated Gradients



"Axiomatic Attribution for Deep Networks"
Mukund Sundararajan, Ankur Taly, Qiqi Yan

https://www.slideshare.net/databricks/how-neural-networks-see-social-networks-with-daniel-darabos-and-janos-maginecz

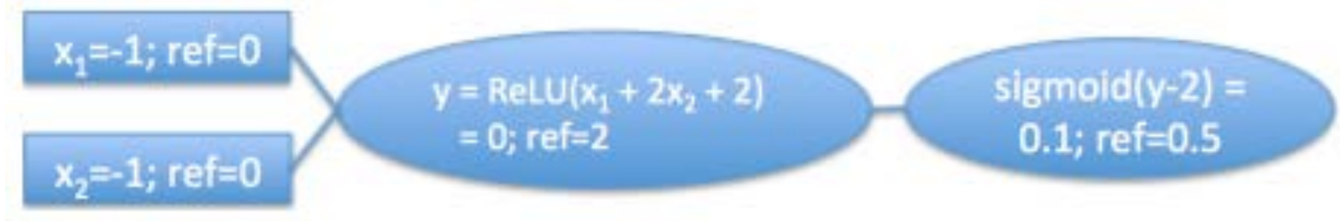
Integrated Gradients

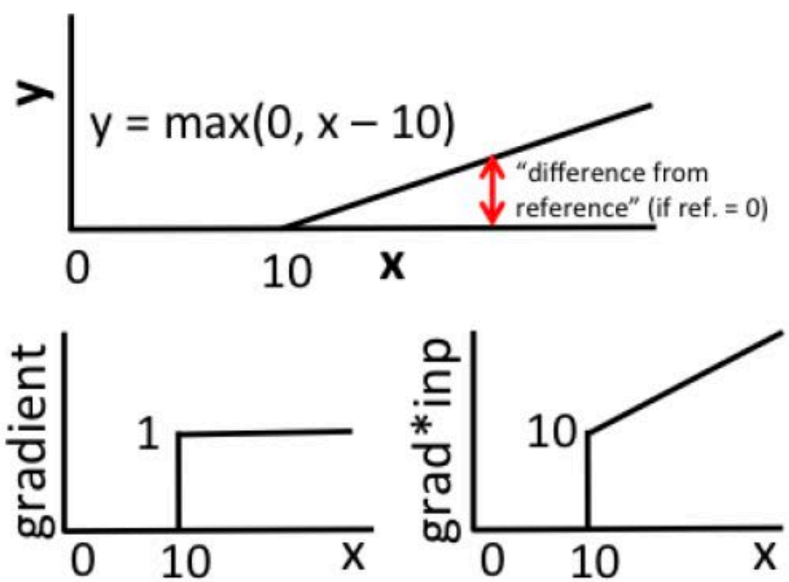


https://towardsdatascience.com/interpretable-neural-networks-45ac8aa91411

DeepLIFT

Compares the activation of each neuron to its **reference activation** and assigns contribution scores according to the difference

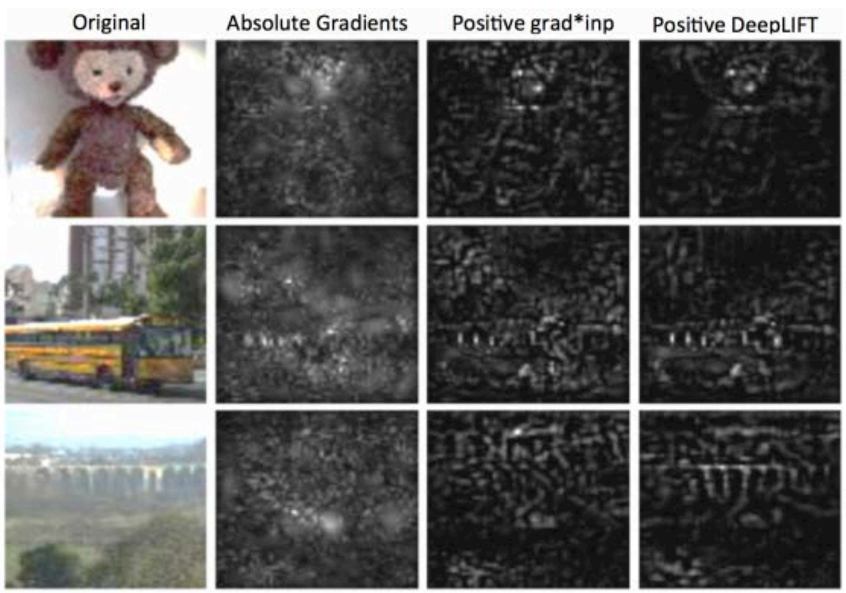


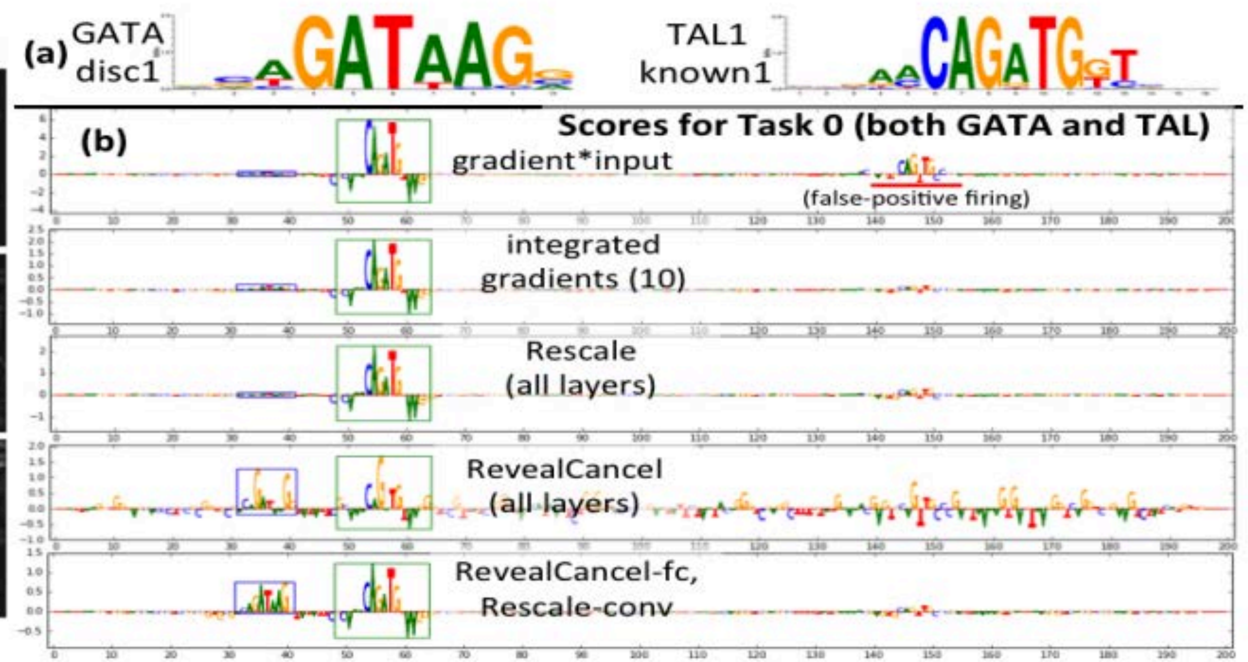


Shrikumar et al., *Learning Important Features Through Propagating Activation Differences*
Shrikumar et al., *Not Just A Black Box: Learning Important Features Through Propagating Activation Differences*

DeepLIFT

Compares the activation of each neuron to its **<u>reference activation</u>** and assigns contribution scores according to the difference





Shrikumar et al., *Learning Important Features Through Propagating Activation Differences*
Shrikumar et al., *Not Just A Black Box: Learning Important Features Through Propagating Activation Differences*

Other input dependent attribution score approaches:

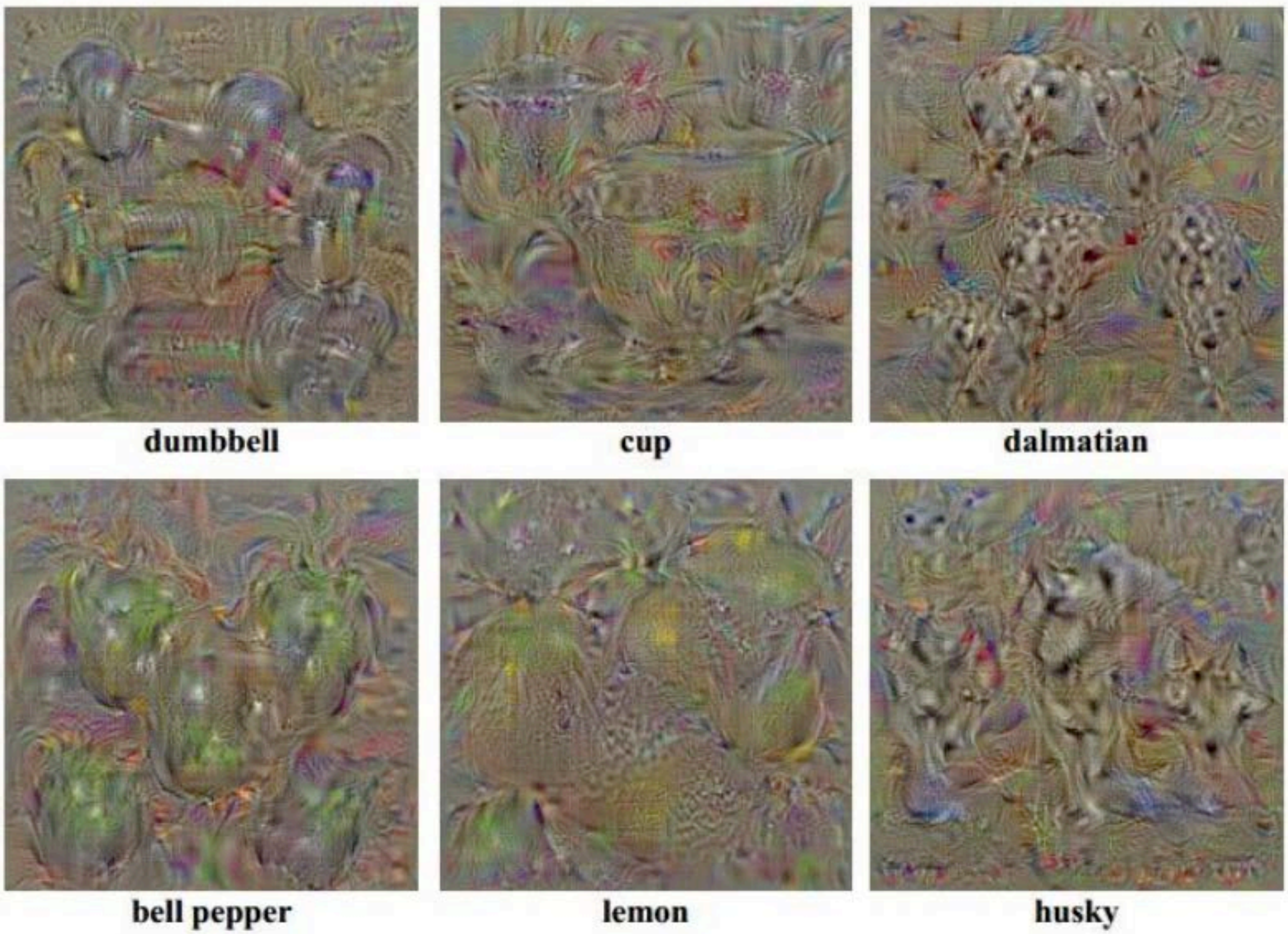
- LIME (Local Interpretable Model-agnostic Explanations)
  - Identify an interpretable model over the representation that is locally faithful to the classifier by approximating the original function with linear (interpretable) model
- SHAP (SHapley Additive explanation)
  - Unified several additive attribution score methods by using definition of Shapley values from game theory
  - Marginal contribution of each feature, averaged over all possible ways in which features can be included/excluded
- Maximum entropy
  - Locally sample inputs that maximize the entropy of predicted score

Input independent visualization: gradient ascent

Generate input that maximizes activation of certain neuron or final activation of the class

$$\arg\max_I S_c(I) - \boxed{\lambda \|I\|_2^2}$$

Simple regularizer: Penalize L2 norm of generated image



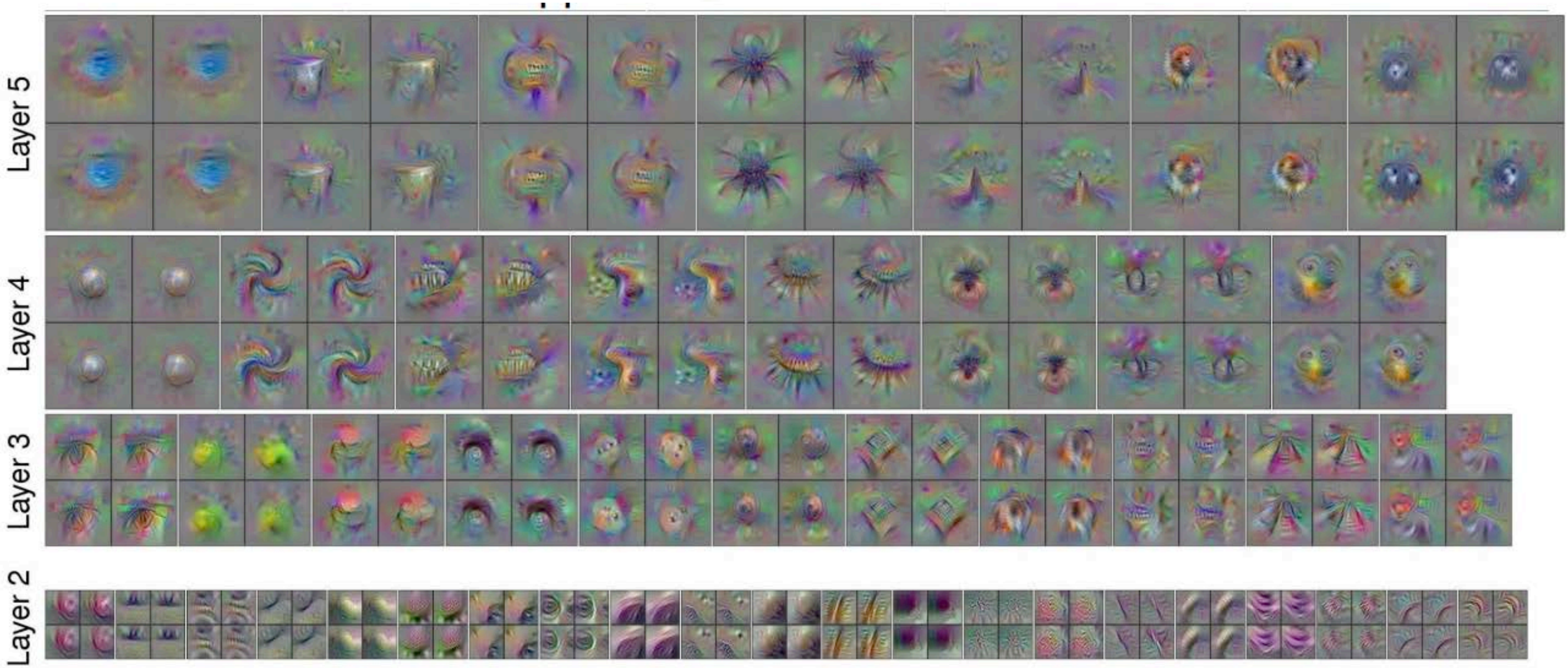
Simonyan et al., *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*

Input independent visualization: gradient ascent

Generate input that maximizes activation of certain neuron or final activation of the class

$$\arg\max_I S_c(I) - \boxed{\lambda \|I\|_2^2}$$

Simple regularizer: Penalize L2 norm of generated image



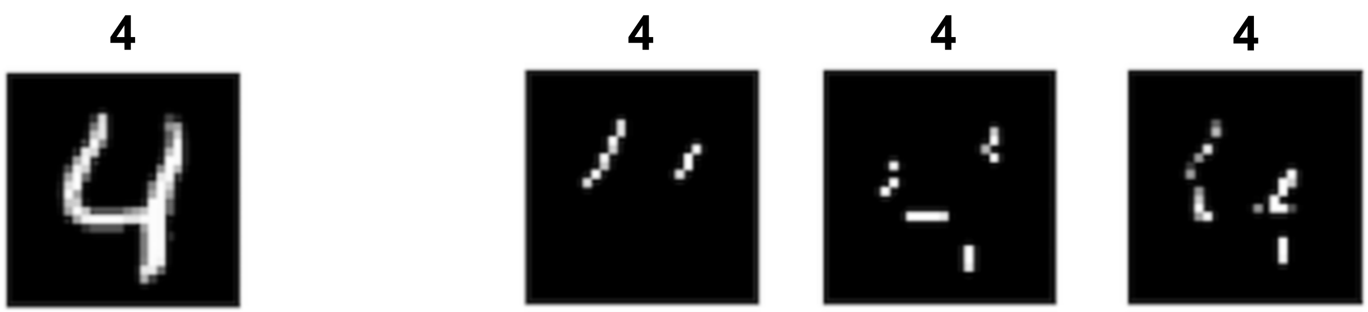
Yosinski et al., *Understanding Neural Networks Through Deep Visualization*

Black box methods
(Do not look inside of model)

$[x_1, x_2, \ldots x_n] \longrightarrow F \longrightarrow y$

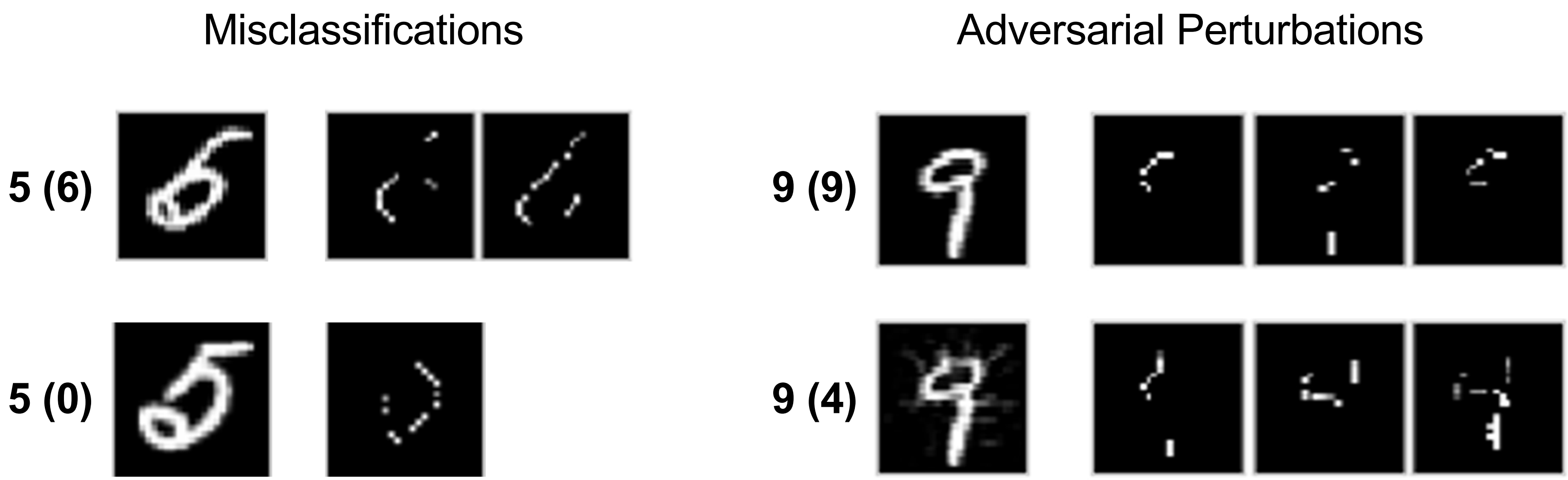
Sufficient Input Subsets

- One simple rationale for ***why*** a black-box decision is reached is a sparse subset of the input features whose values form the basis for the decision
- A ***sufficient input subset*** (SIS) is a minimal feature subset whose values alone suffice for the model to reach the same decision (even without information about the rest of the features' values)



Carter et al., *What made you do this? Understanding black-box decisions with sufficient input subsets*

SIS help us understand misclassifications



Formal Definitions – Sufficient Input Subset

- Black-box model that maps inputs $\mathbf{x} \in \mathcal{X}$ via a function $f : \mathcal{X} \to \mathbb{R}$
- Each input has indexable features $\mathbf{x} = [x_1, \ldots, x_p]$ with each $x_i \in \mathbb{R}^d$

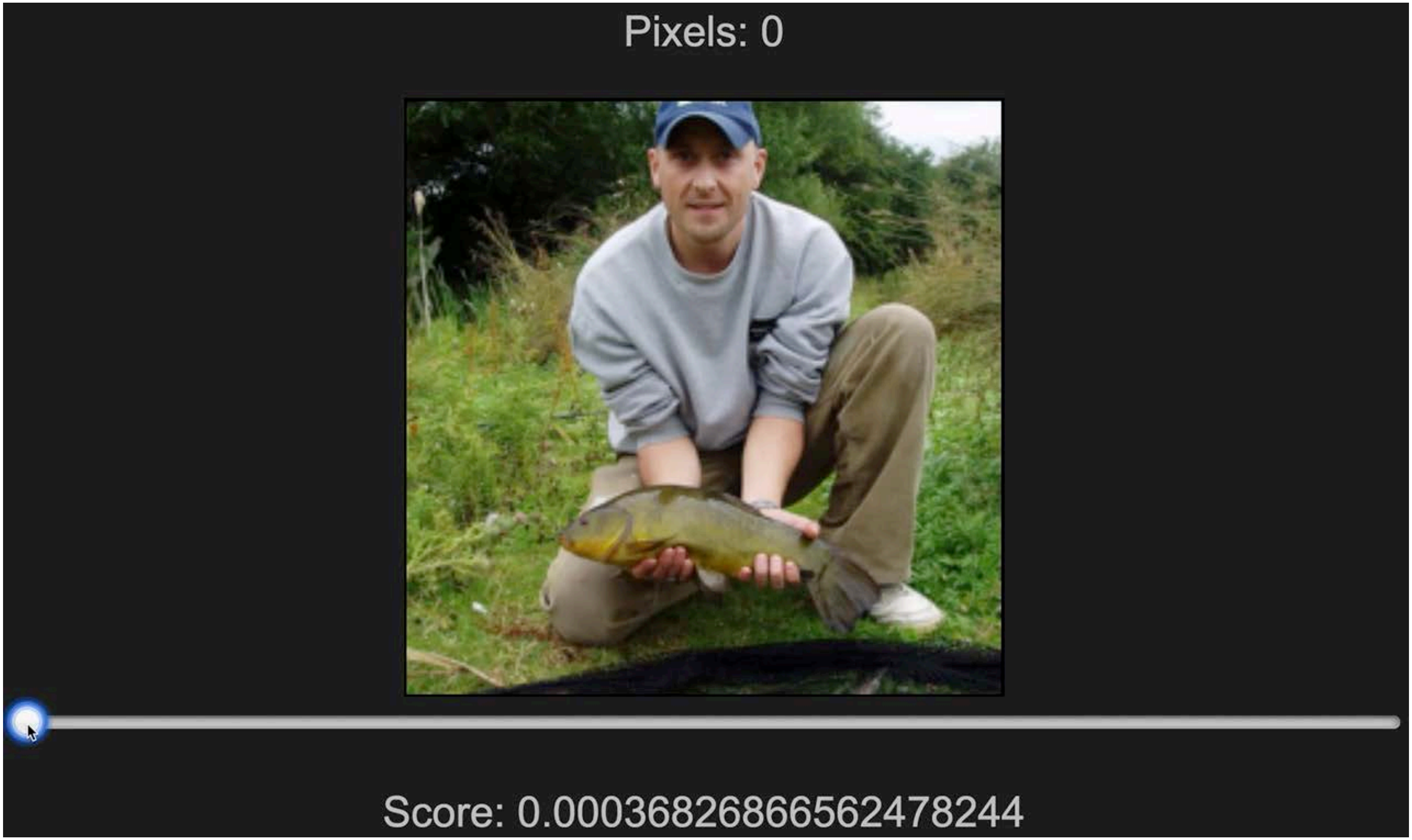
Formal Definitions – Sufficient Input Subset

- Black-box model that maps inputs $\mathbf{x} \in \mathcal{X}$ via a function $f : \mathcal{X} \to \mathbb{R}$
- Each input has indexable features $\mathbf{x} = [x_1, \ldots, x_p]$ with each $x_i \in \mathbb{R}^d$
- A **SIS** is a subset of the input features $S \subseteq [p]$ (along with their values)
- Presume decision of interest is based on $f(\mathbf{x}) \geq \tau$ (pre-specified threshold)
- Our goal is to find a **complete** collection of **minimal-cardinality subsets** of features S, each satisfying $f(\mathbf{x}_S) \geq \tau$
- $\mathbf{x}_S$ = input where values of features outside of S have been masked

SIS Algorithm

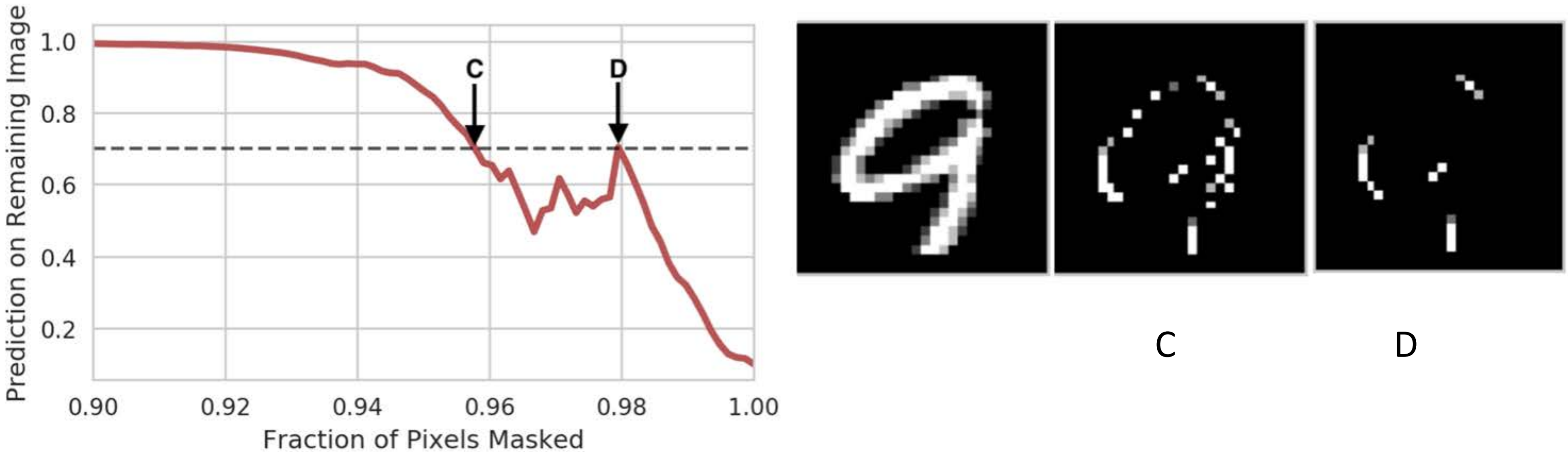
- From a particular input: we extract **SIS-collection** of disjoint feature subsets, each of which alone suffices to reach the same model decision
- Aim to quickly identify each sufficient subset of minimal cardinality via **backward selection** (preserves interaction between features)
- Aim to identify all such subsets (under disjointness constraint)
- Mask features outside of SIS via their average value (mean-imputation)
- Compared to existing interpretability techniques, SIS is **faithful to any type of model** (sufficiency of SIS is guaranteed), and does **not** require: gradients, additional training, or an auxiliary explanation model

Backward Selection Visualized



Courtesy of Zheng Dai

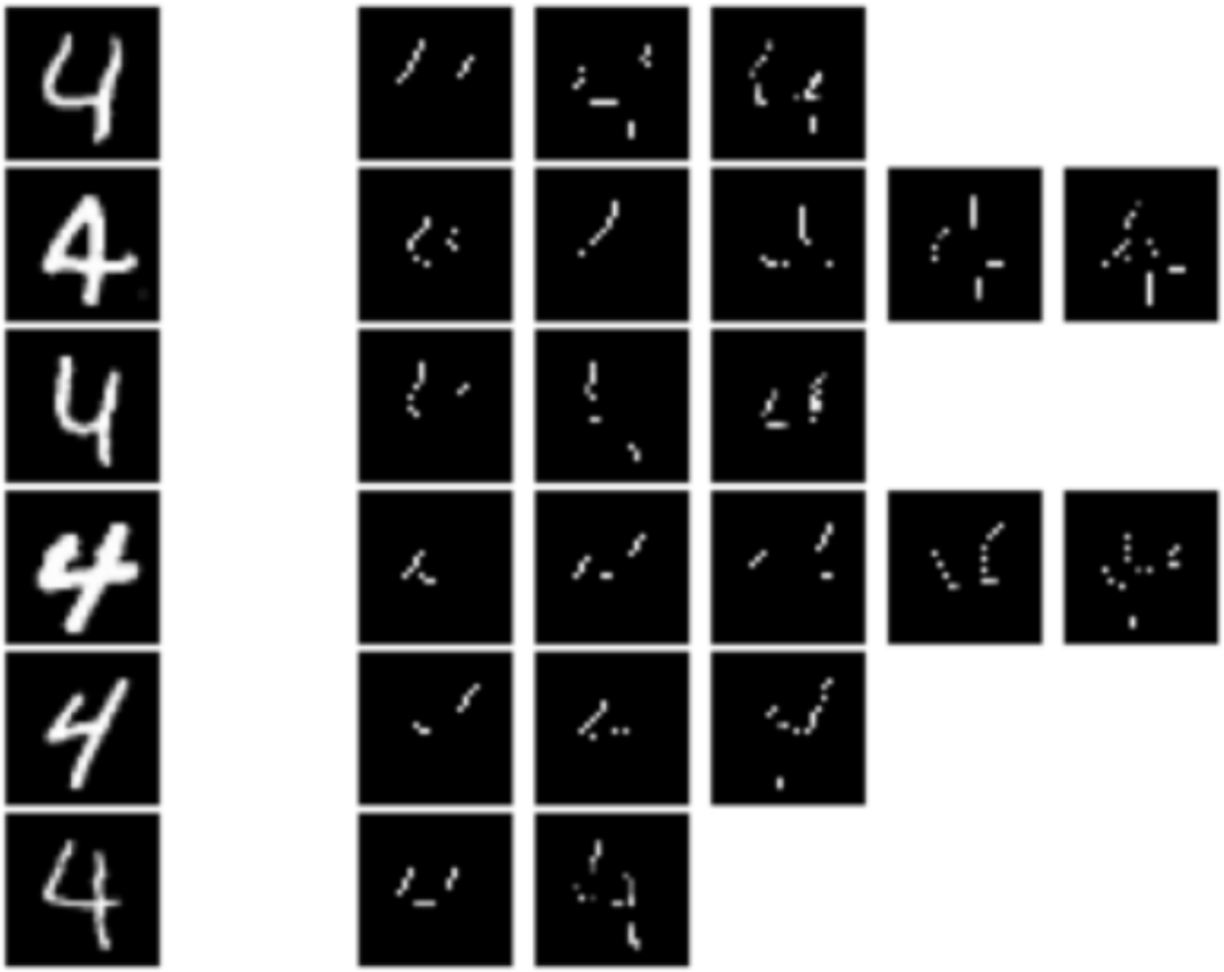
SIS avoids local minima by using backward selection



C

D

Example SIS for different instances of ”4”



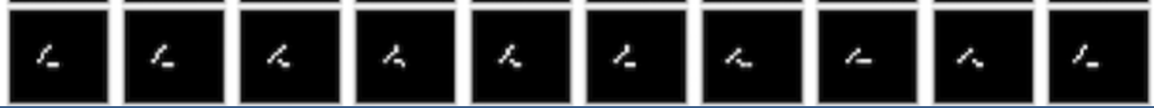
SIS Clustered for General Insights

- Identifying the input patterns that justify a decision across many examples helps us better understand the general operating principles of a model

- We cluster all SIS identified across a large number of examples that received the same model decision

- Insights revealed by our SIS-clustering can be used to compare the global operating behavior of different models

SIS Clustering Shows CNN vs. Fully Connected Network Differences (digit 4)

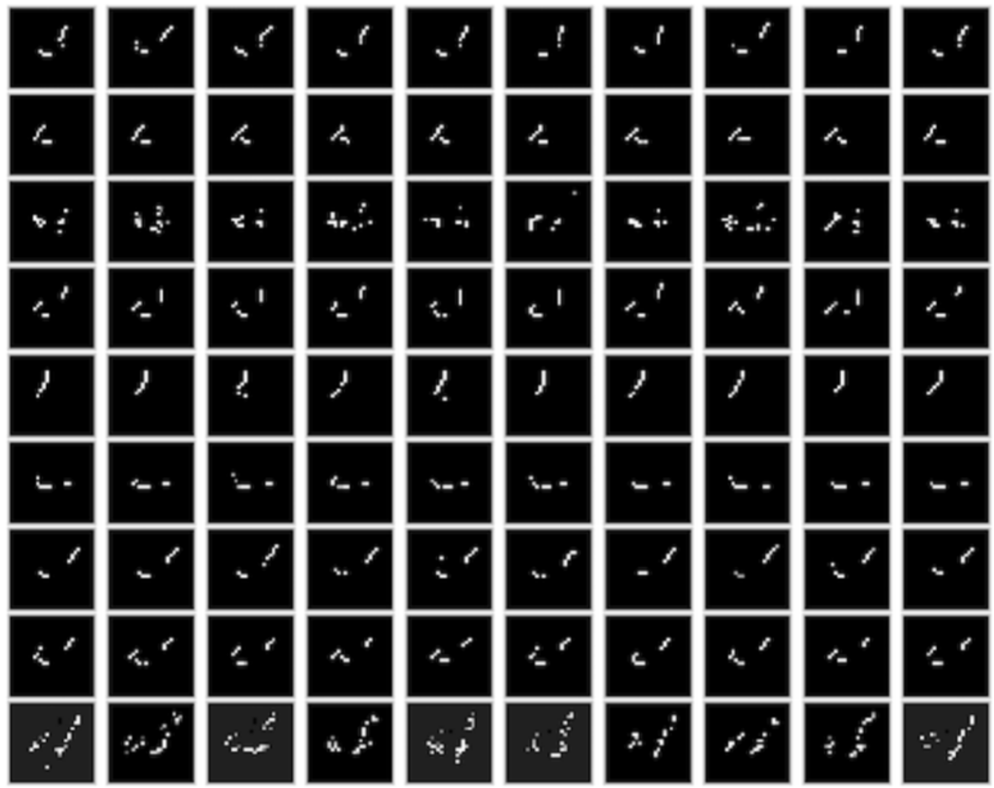
Cluster	**% CNN SIS**	
C_1	100%	
C_2	100%	
C_4	100%	
C_5	100%	
C_6	100%	
C_7	100%	
C_8	100%	

SIS Clustering Shows CNN vs. Fully Connected Network Differences (digit 4)

Cluster	% CNN SIS	
C_3	5%	
C_9	0%	

SIS Clustering Shows CNN vs. Fully Connected Network (MLP) Differences

Cluster	% CNN SIS
C_1	100%
C_2	100%
C_3	5%
C_4	100%
C_5	100%
C_6	100%
C_7	100%
C_8	100%
C_9	0%



- CNN: spatially-contiguous strokes comprising small portion of digit
- MLP: decision based on pixels throughout digit, relies on global shape
- CNN is more susceptible to mistaking other (non-digit) handwritten characters for 4 if they happen to share some of the same strokes

Applying SIS to Natural Language

- We use a dataset of beer reviews from BeerAdvocate [McAuley et al. 2012]

- Different LSTM networks are trained to predict user-provided numerical ratings of aspects like **aroma**, **appearance**, and **palate**

LSTMs Learn Aspect-Specific Features

on tap at the brewpub december 27 2010 pours a dark brown color with a good tan head that leaves behind a bit of lacing and sticks around for awhile the nose is really nice and chocolatey really love the level they 've used under that a bit of roasted malt but this was mostly about the chocolate the taste is n't quite as nice though the chocolate notes really still stand out the feel was quite nice with a full body pretty viscous for what it is drinks quite well i 'm a big fan

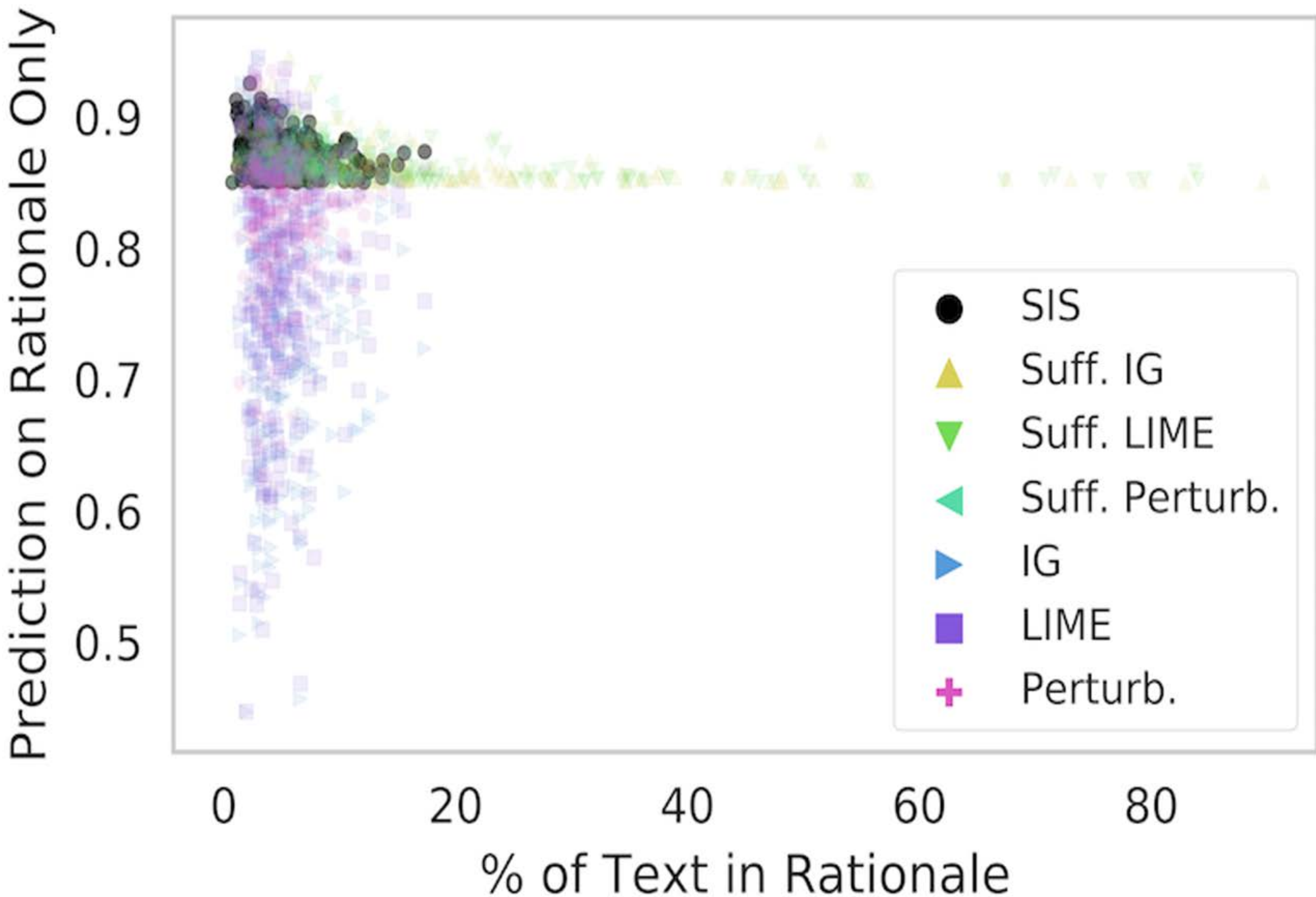
Appearance Aroma Palate

Multiple SIS in Aroma Review

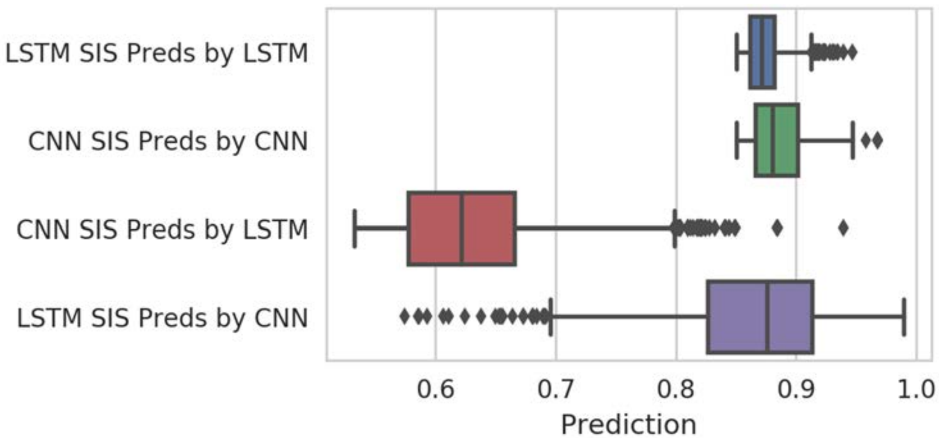
on tap at a the pour is a dark amber color bordering on mahogany with a finger 's worth of slightly off white head s wow the nose on this beer is phenomenal tons of vanilla bourbon maple syrup brown sugar caramel and toffee provide a wonderful sweetness some dark fruit notes and chocolate fill in the background of the aroma t the flavor is similarly impressive lots of sweet rich vanilla bourbon and oak accompanied by toffee caramel brown sugar and maple syrup the finish is all that prevents this from a perfect score as there is a bit of alcohol and heat but there are some nice hints of chocolate m the mouthfeel is smooth creamy rich and full bodied a light but nearly perfect level of carbonation d i was told this beer was good but i had to see for myself this is one of if not the best barrel aged barleywines i 've come across i might go back again soon to have some more

Aroma SIS 1 Aroma SIS 2 Aroma SIS 3

SIS Produces Minimal Sufficient Subsets



SIS Clustering Shows LSTM/CNN Differences



Clu.	% LSTM	SIS #1	SIS #2	SIS #3	SIS #4
C1	0%	delicious	-	-	-
C2	0%	very nice	-	-	-
C3	20%	rich chocolate	very rich	chocolate complex	smells rich
C4	33%	oak chocolate	chocolate raisins raisins oak bourbon	chocolate oak	raisins chocolate
C5	70%	complex aroma	aroma complex peaches complex	aroma complex interesting cherries	aroma complex

Example sufficient input subsets for MAFF binding

Two DNA sequences that receive positive TF (MAFF) binding predictions (SIS is shaded):

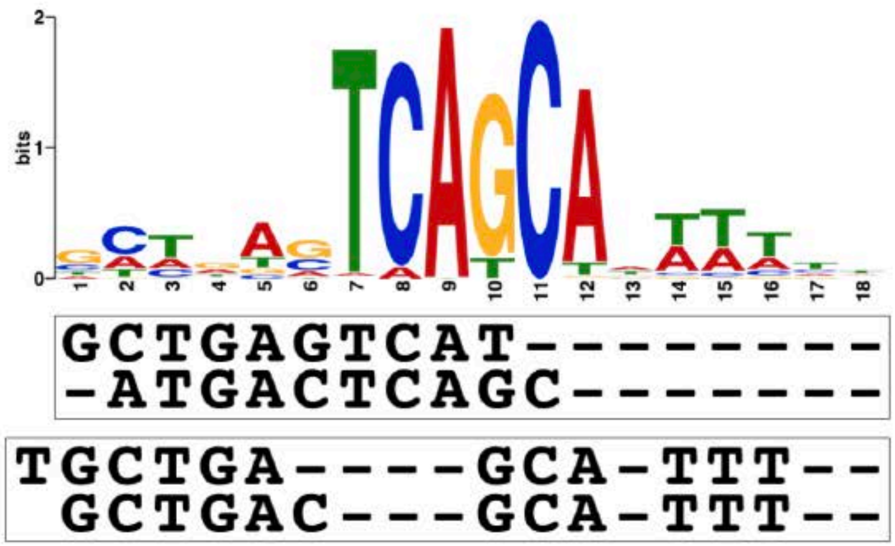
```
CACTGTCATTCTCTTGGTCAGCCCTGGACATCCCTGGAAAGGATGACTCAGCTGTCCGTTTTAAACAGGGTAGTTCAGAAGAATACATTCCTGGTTATTCA
TTTTTTTCTCCCTTCGATTTCCACTATGATTTGTATTTCCTTTGTTCTGCTGACTTTGCAATTTCGGTTGTTTTTTCTAAATTTCTTAGGGTGAAAACTGA
```

Example clustered SIS for a transcription factor (MAFF factor)

Clustering results for a particular TF (MAFF), two clusters were found:

SIS	Freq.
GCTGAGTCAT	197
ATGACTCAGC	185
GCTGAGTCA-C	83
GCTGAGTCAC	53
GCTGACTCAGCA	42

SIS	Freq.
TGCTGA--GCA-TTT	12
GCTGAC--GCA-TTT	8
TGCTGAC--GCA-TT	6
TGCTGAC--GCA-AA	5
TGCTGAC--GCA-AT	4



Right image: known JASPAR motif (top) and alignment with cluster modes (bottom)

FIN - Thank You

SIS Resources

SIS paper:
https://arxiv.org/abs/1810.03805

Code for open-source SIS library and tutorial:
https://github.com/google-research/google-research/tree/master/sufficient_input_subsets