Announcements

- HW 4 due **Wednesday, November 2 at 8pm**
- Quiz 7 due **Thursday, October 27 at 8pm**
- I am not holding office hours for today

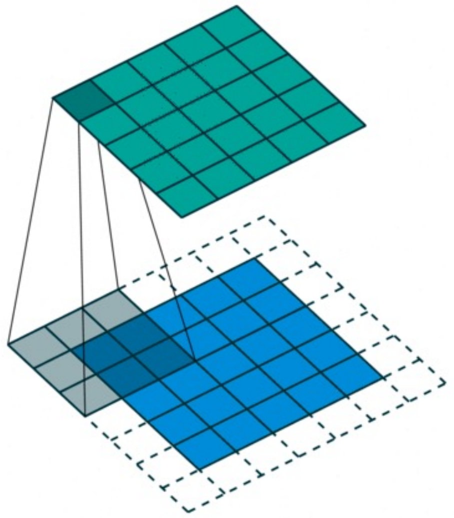
Lecture 15: Computer Vision (Part 3)

CIS 4190/5190

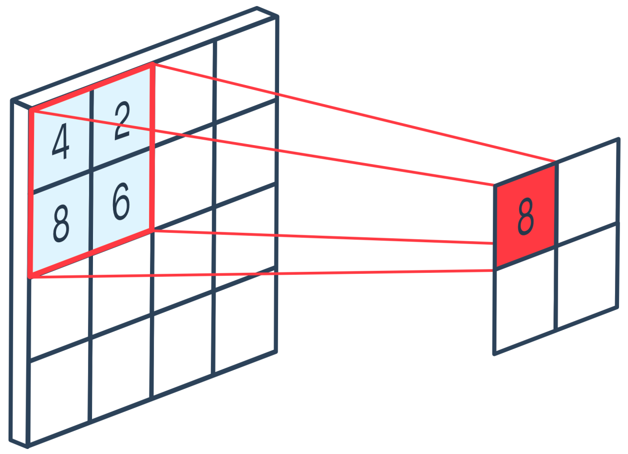
Fall 2022

Recap: Pooling & Convolution

- Use layers that capture structure



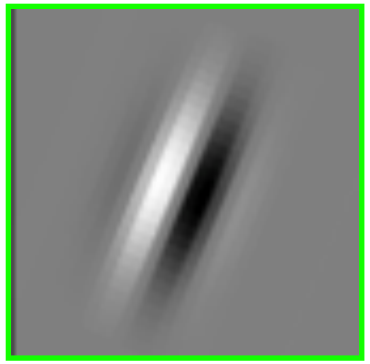
Convolution layers
(Capture equivariance)

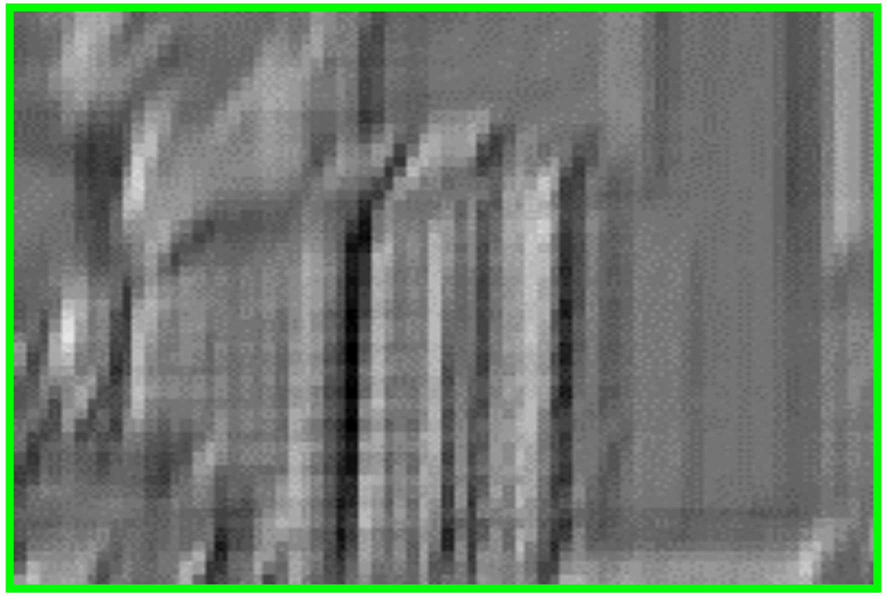


Pooling layers
(Capture invariance)

https://towardsdatascience.com/types-of-convolutions-in-deep-learning-717013397f4d
https://peltarion.com/static/2d_max_pooling_pa1.png

Recap: Convolution Layers

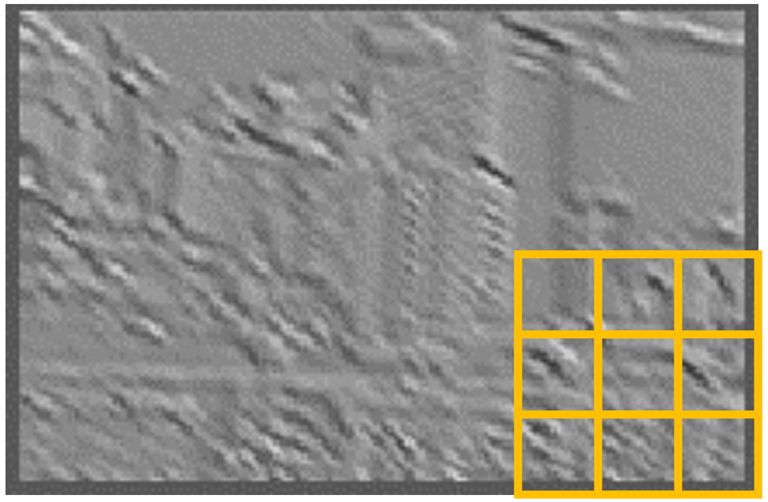


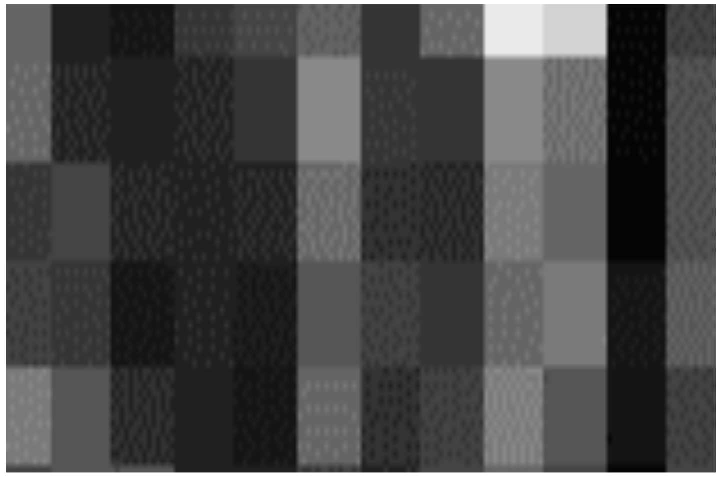


$$\text{output}[i,j] = \sum_{\tau=0}^{k-1} \sum_{\gamma=0}^{k-1} \text{filter}[\tau,\gamma] \cdot \text{image}[i+\tau, j+\gamma]$$

graphic credit: S. Lazebnik

Recap: Pooling Layers



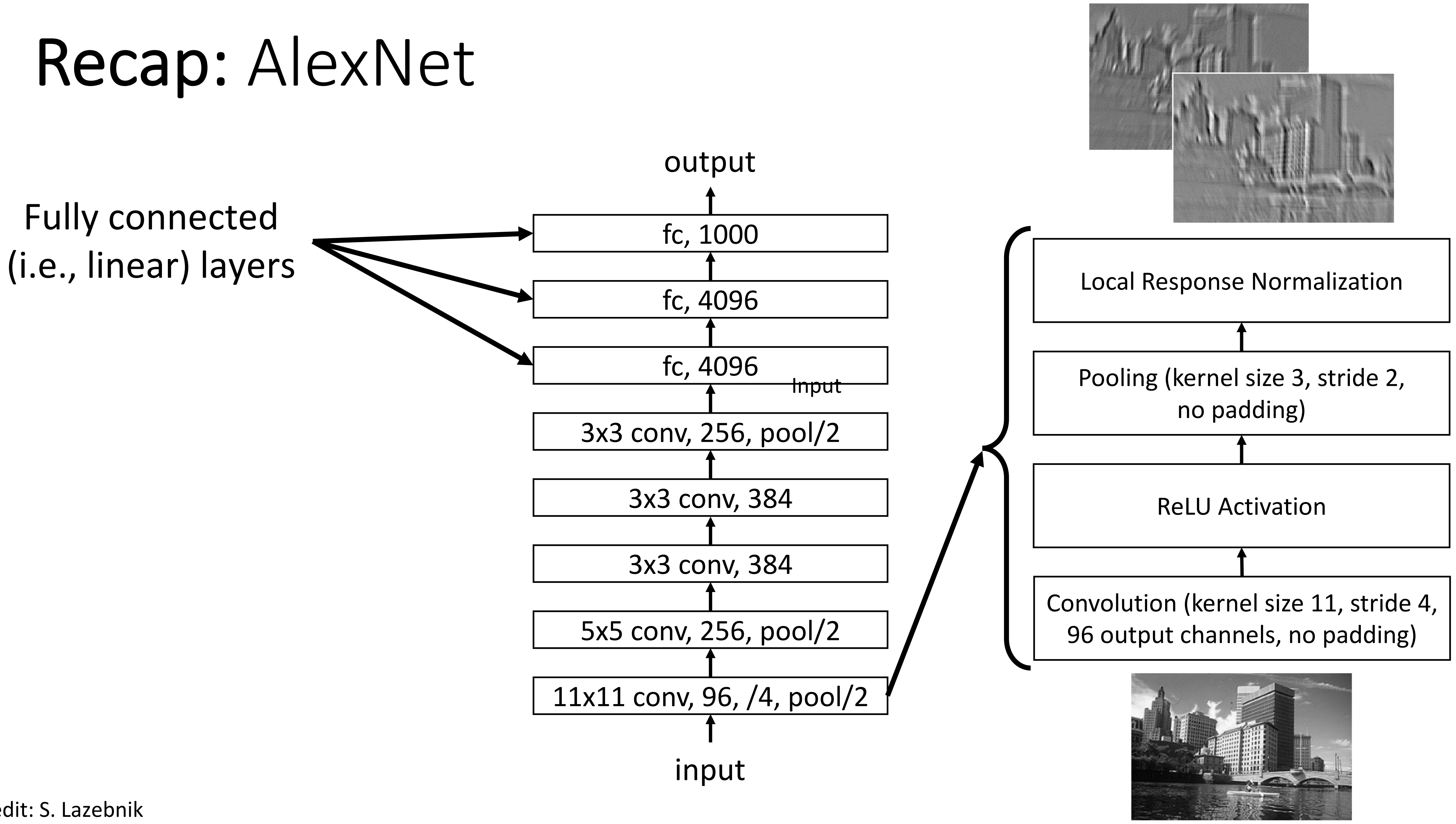


$$\text{output}[i, j] = \max_{0 \le \tau < k} \max_{0 \le \gamma < k} \text{image}[i + \tau, j + \gamma]$$

Recap: Convolution vs. Pooling

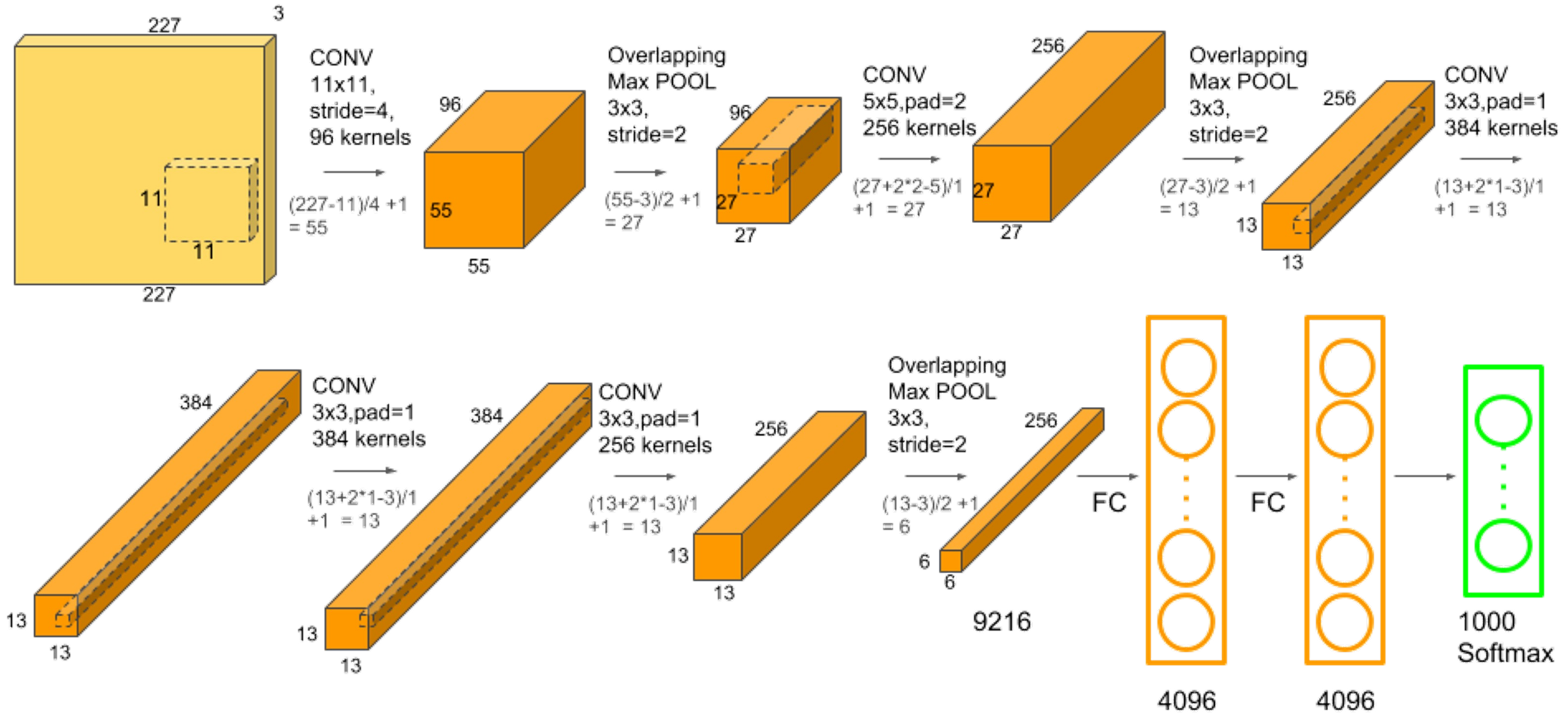
- **Convolution layers:** Translation equivariant
  - If object is translated, convolution output is translated by same amount
  - Produce “image-shaped” features that retain associations with input pixels
- **Pooling layers:** Translation invariant
  - Binning to make outputs insensitive to translation
  - Also reduces dimensionality
- Combined in modern architectures
  - Convolution to construct equivariant features
  - Pooling to enable invariance

Recap: AlexNet


Fully connected (i.e., linear) layers
output
fc, 1000
fc, 4096
fc, 4096
Input
3x3 conv, 256, pool/2
3x3 conv, 384
3x3 conv, 384
5x5 conv, 256, pool/2
11x11 conv, 96, /4, pool/2
input
Local Response Normalization
Pooling (kernel size 3, stride 2, no padding)
ReLU Activation
Convolution (kernel size 11, stride 4, 96 output channels, no padding)

slide credit: S. Lazebnik

Recap: AlexNet



Recap: Residual Connections

- **Challenges with deep networks**
  - Overfitting?
  - No, 56 layer network underfits!
- **Optimization/representation**
  - Difficulty representing the identity function!
- **Solution:** "Skip" connections
  - Facilitate direct feedback from loss
  - Easy to represent identity function

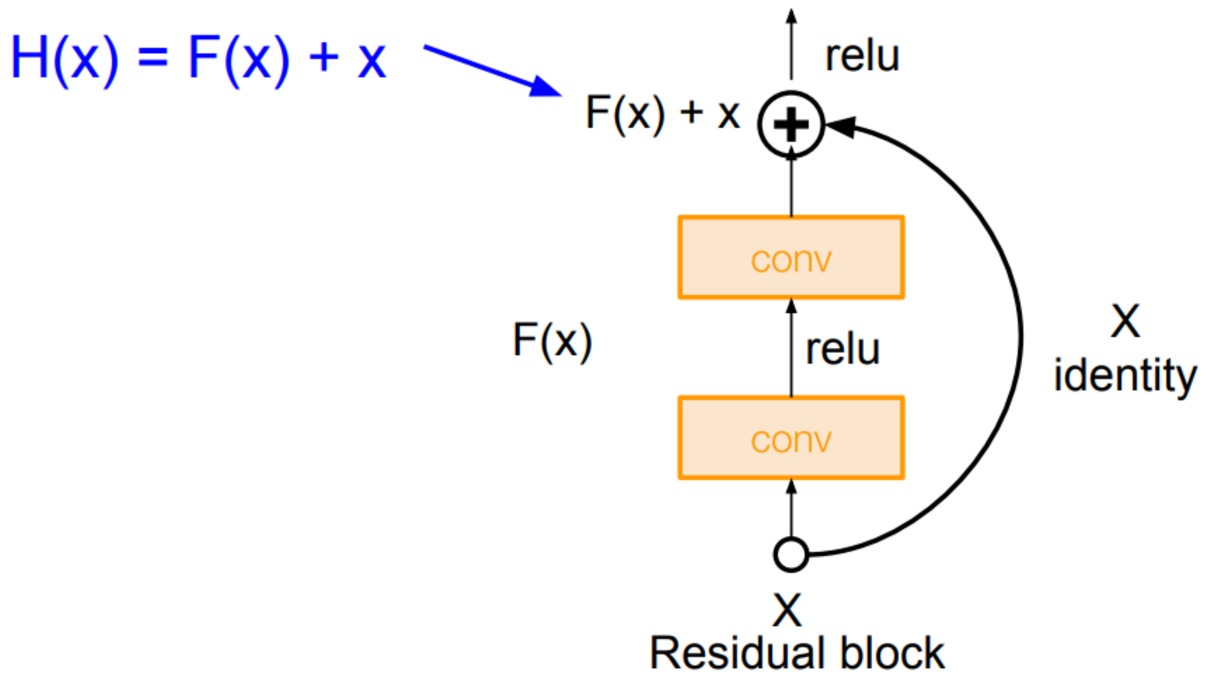


Image credit: Fei-Fei Li, Justin Johnson, Serena Yeung

Recap: Residual Networks

- Stack lots of residual blocks!
  - Kernel size 3, no padding, stride 1, no pooling
  - Reduce feature dimensions by using stride 2 once every K blocks
  - Maintains feature size to build very deep nets

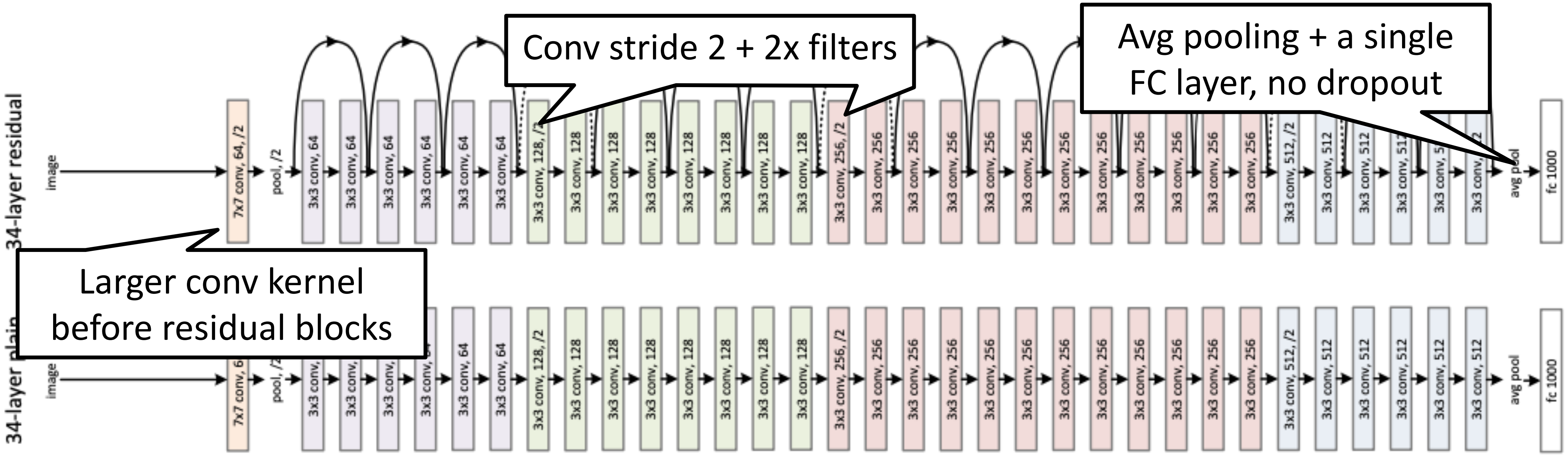


Image credit: He et al, Residual Nets, 2015

Recap: Transfer Learning/Finetuning

- **Transfer learning:** We can reuse trained concepts!
  - Since CNNs trained on ImageNet appear to learn general features
  - We can reuse these models in some way to perform new tasks

- **Strategy 1:** Feature extraction
  - Remove final (softmax) layer and replace with a new one
  - Train only the new layer

- **Strategy 2:** Finetuning
  - Do the same thing but train end-to-end

Agenda

- **Interpretability & Explainability**
- **Robustness to distribution shift**
- **Robustness to adversarial attacks**

Interpretability & Explanability

- **Interpretability:** How does the model make predictions?
  - Useful for debugging issues with the model
  - Not feasible for deep neural networks

- **Explainability:** How did the model make a specific prediction?
  - “Local” interpretation that can still be very useful for debugging

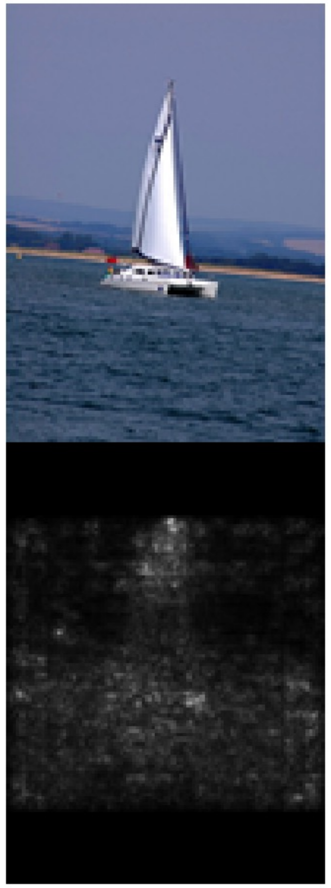
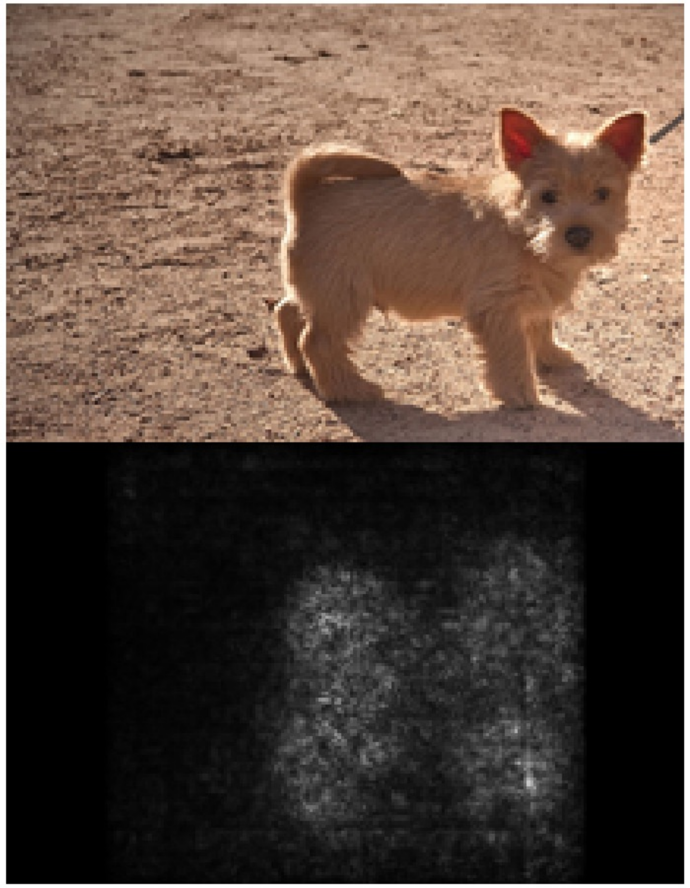
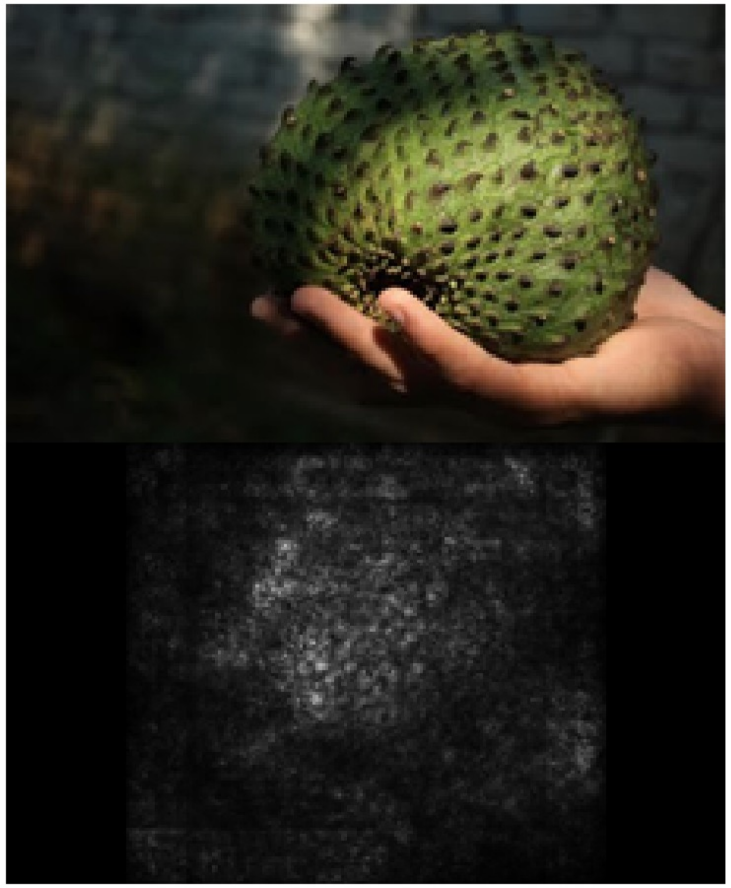
Input Gradients

- Consider the gradient of the loss with respect to the input:

$$s = \nabla_x \tilde{L}\left(f_\beta(x), y\right)$$

- **Intuition**
  - The gradient $s_{i,j}$ captures the effect of perturbing input $x_{i,j}$ on the loss when assuming the true label is y
  - Larger gradients → more “important” feature
  - **Note:** y does not need to be the true label!

Saliency Maps

Simonyan et al., Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. 2013

Lots of Modifications

- **Guided backpropagation:** Zero out negative signals in backward pass
- **Integrated gradients:** Average over range of gradients
- **Local explanations:** Use sampling + fit model instead of gradient

Local Explanations

- Construct dataset

$$Z = \left(x + \epsilon, f_{\beta}(x + \epsilon)\right)$$

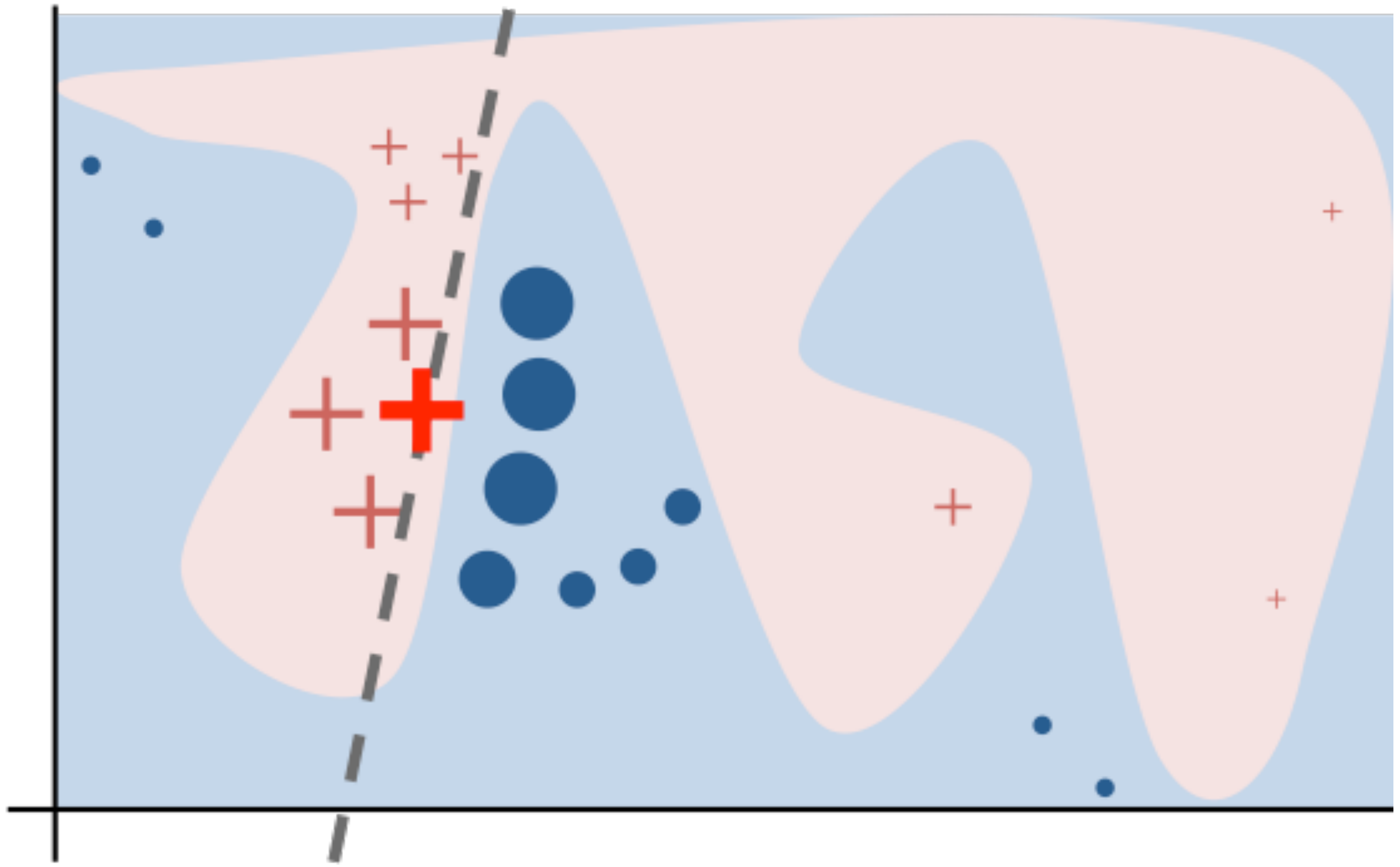
  - Here, $\epsilon \sim N(0, \sigma^2)$ is i.i.d. Gaussian noise

- Fit a linear model to this dataset Z

- “Smoothed” saliency maps (recover saliency maps as $\sigma \to 0$)

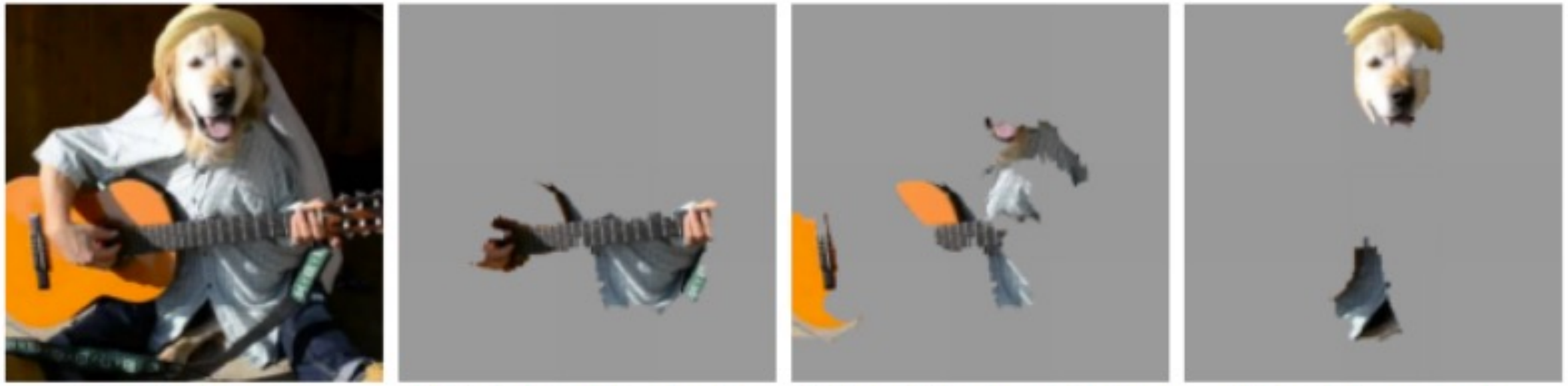
Ribeiro et al., “Why Should I Trust You? Explaining the Predictions of Any Classifier”, 2016

Local Explanations



Ribeiro et al., “Why Should I Trust You? Explaining the Predictions of Any Classifier”, 2016

Local Explanations



(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

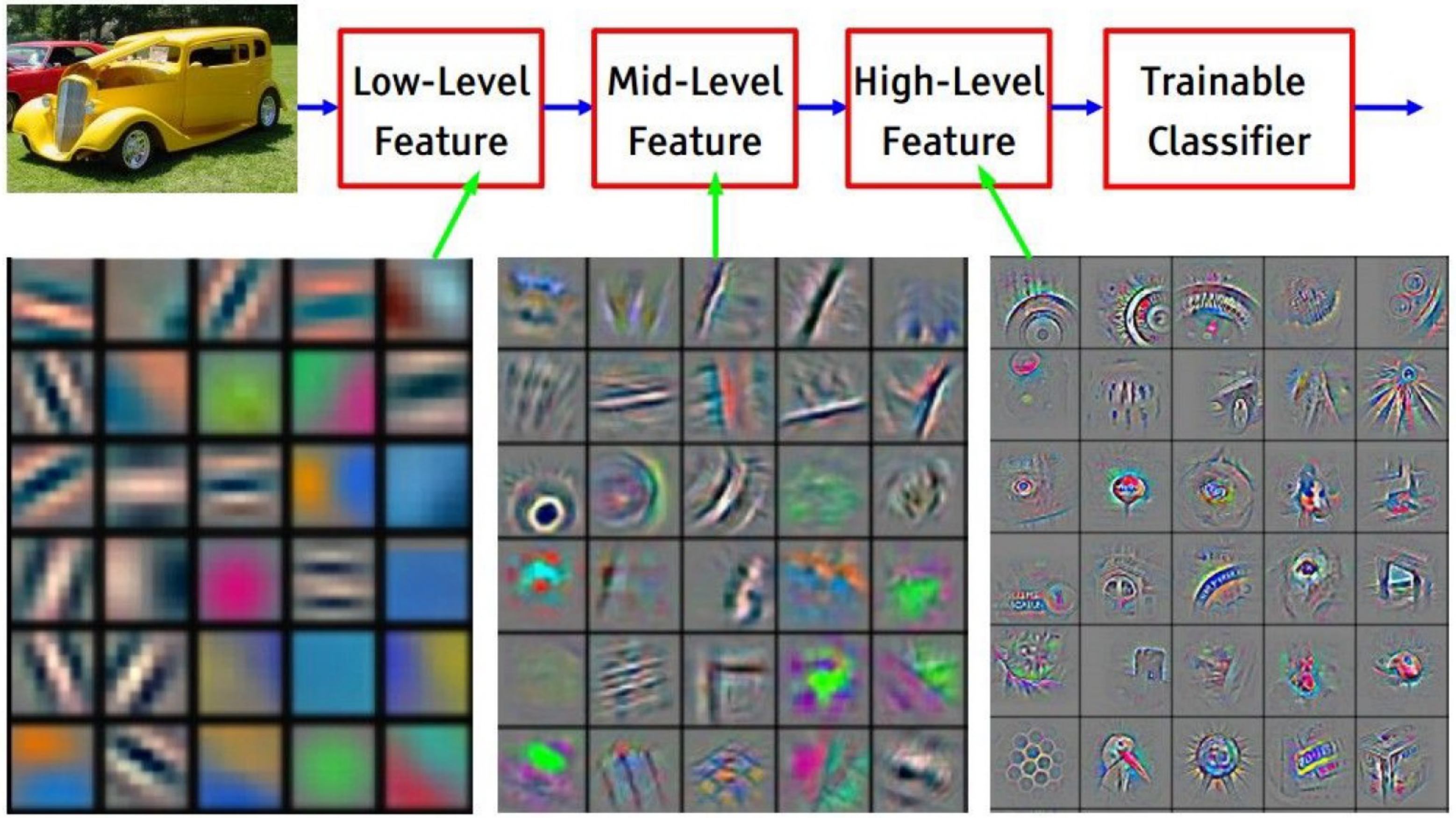
Figure 4: Explaining an image classification prediction made by Google's Inception network, highlighting positive pixels. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$)**, "Acoustic guitar"** ($p = 0.24$) **and "Labrador"** ($p = 0.21$)

Ribeiro et al., "Why Should I Trust You? Explaining the Predictions of Any Classifier", 2016

Neuron Visualization

- **Neuron visualization:** Look at $\nabla_x g_\beta(x)$ for an intermediate layer g_β
- **Network dissection:** Look at groups of pixels corresponding to objects

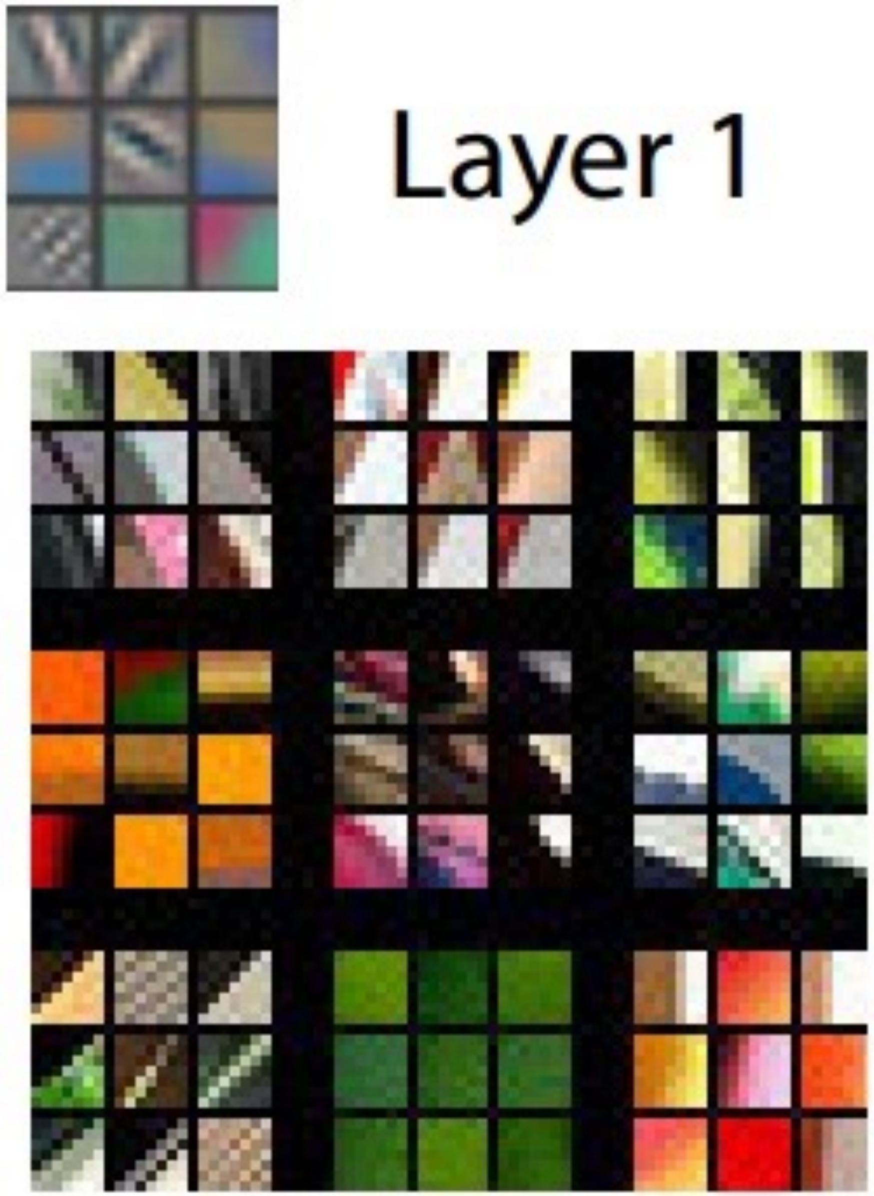
Neuron Visualization



Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Slide credit: Yann LeCun

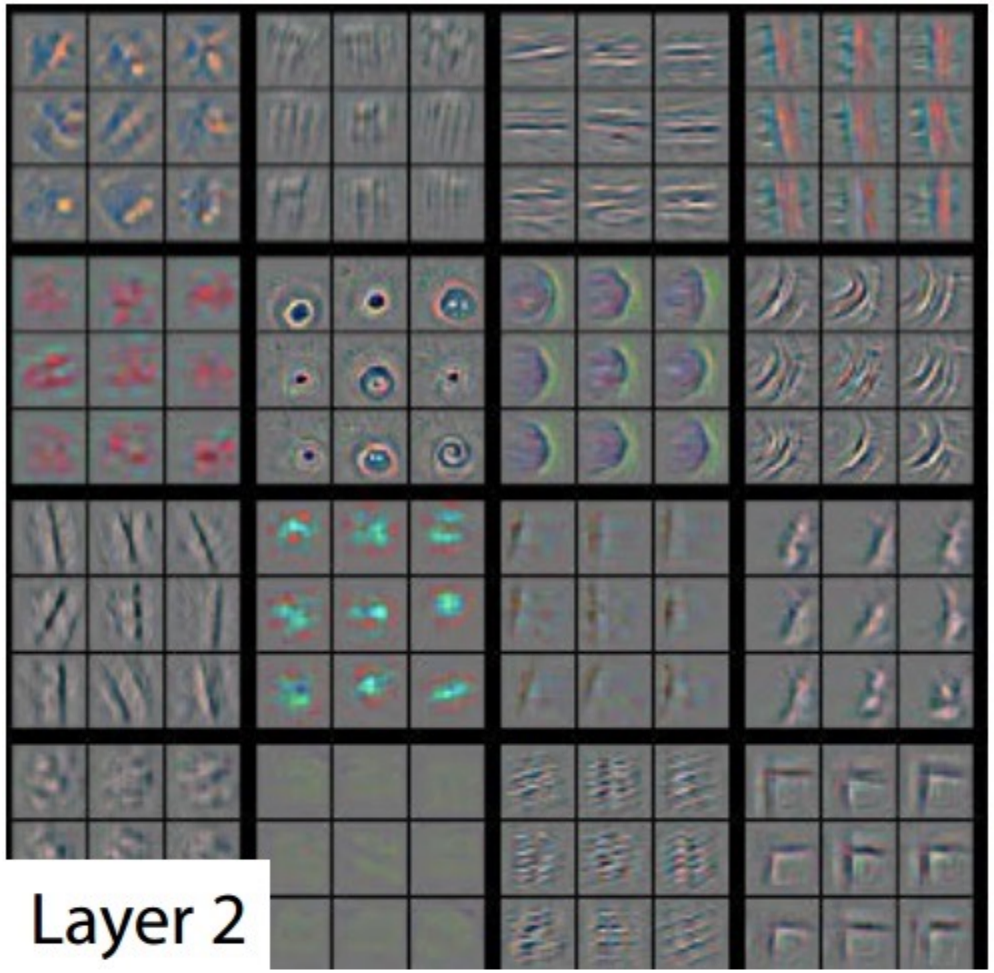
Layer 1

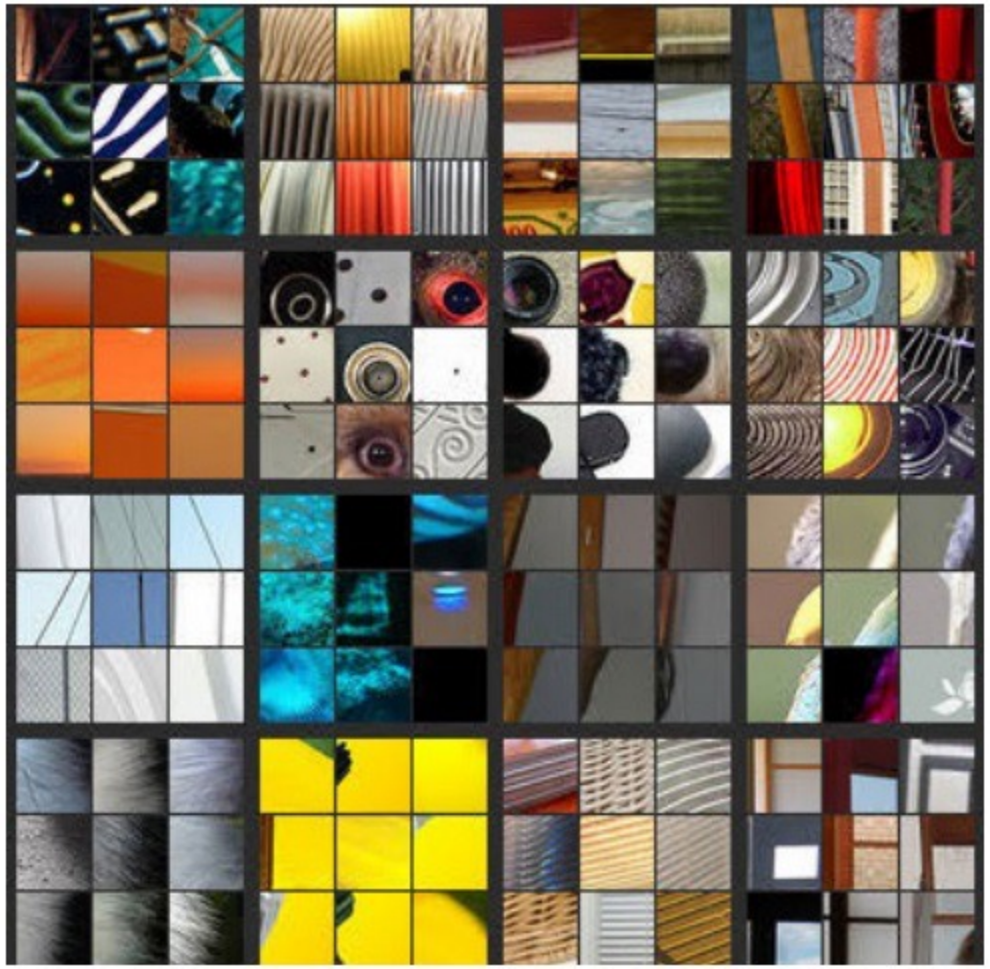


Visualizing and Understanding Convolutional Networks [Zeiler and Fergus, ECCV 2014]

Slide credit: Jia-Bin Huang

Layer 2

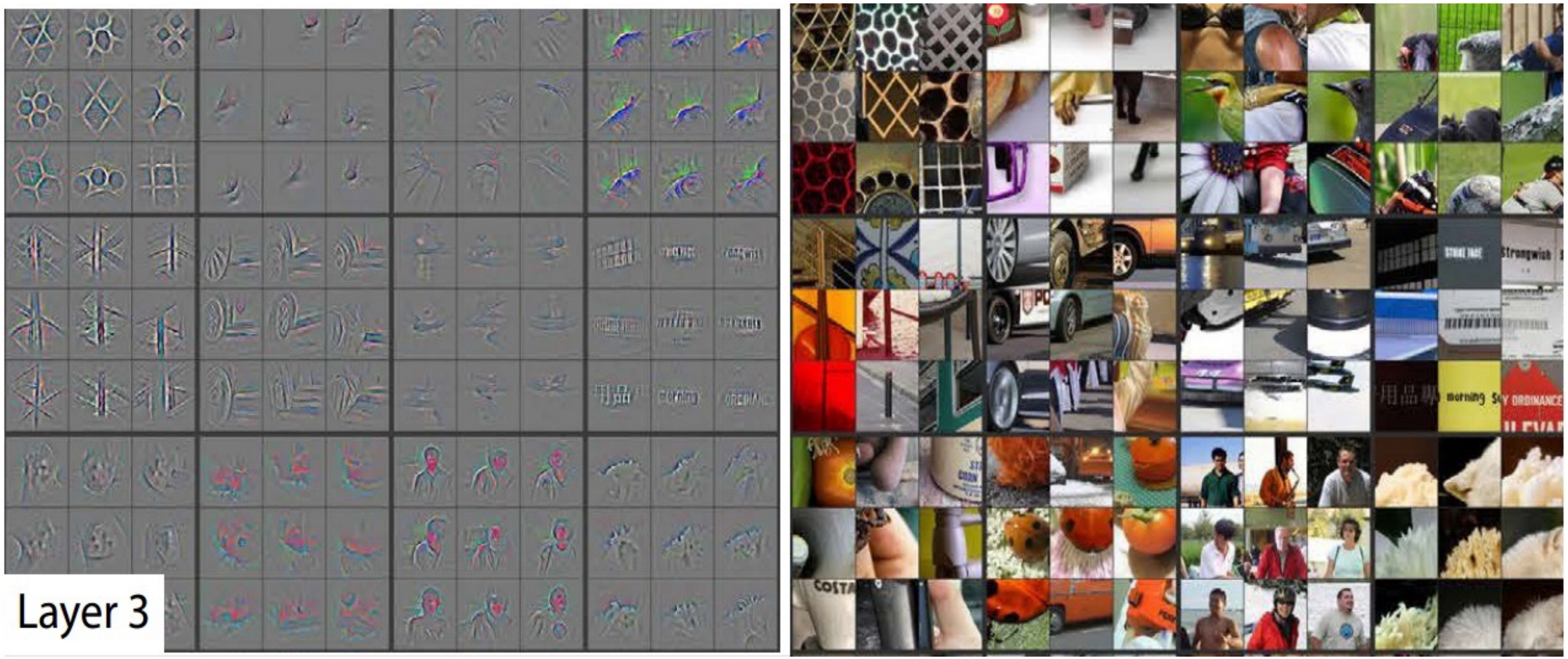




Visualizing and Understanding Convolutional Networks [Zeiler and Fergus, ECCV 2014]

Slide credit: Jia-Bin Huang

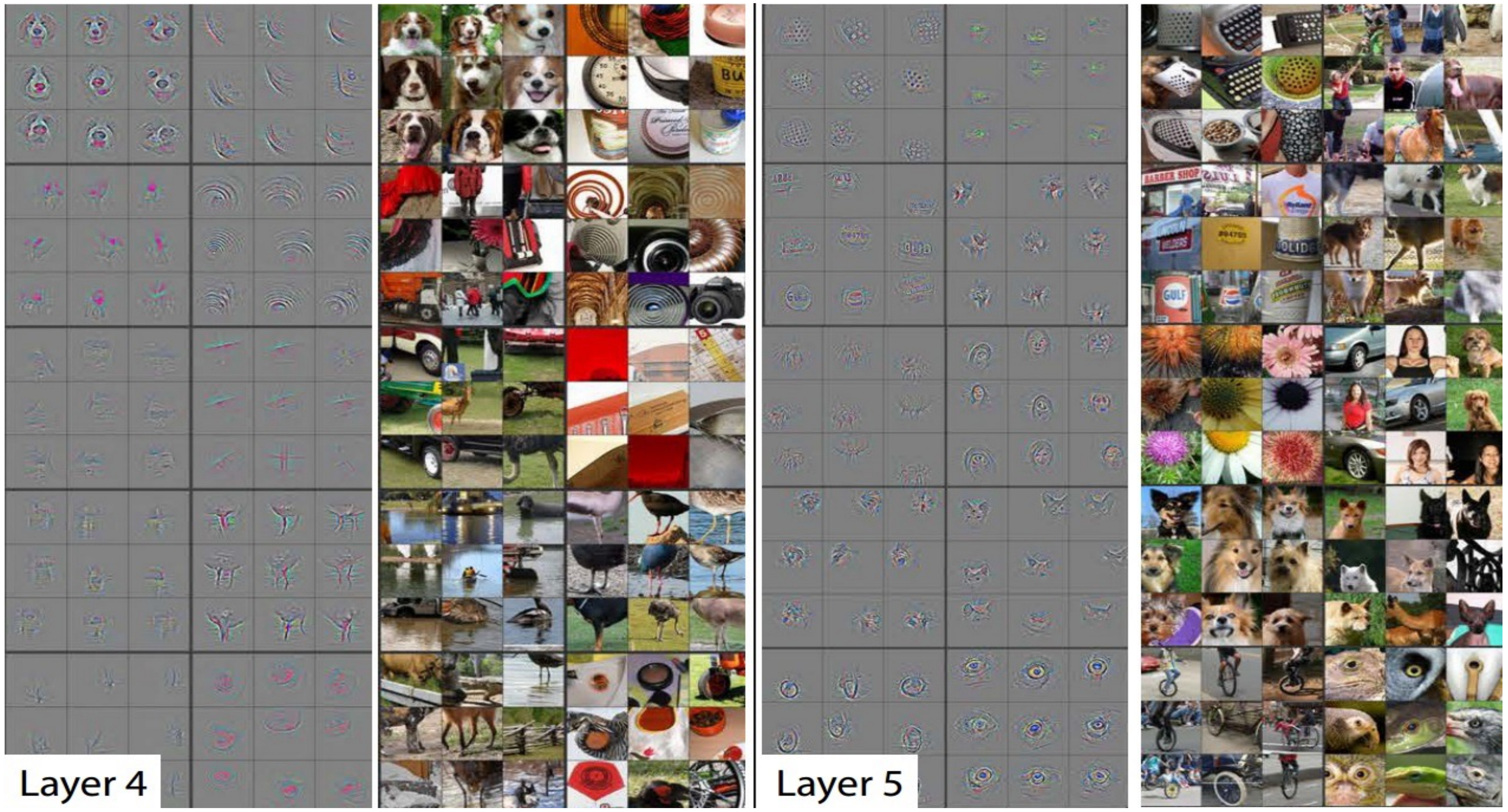
Layer 3



Visualizing and Understanding Convolutional Networks [Zeiler and Fergus, ECCV 2014]

Slide credit: Jia-Bin Huang

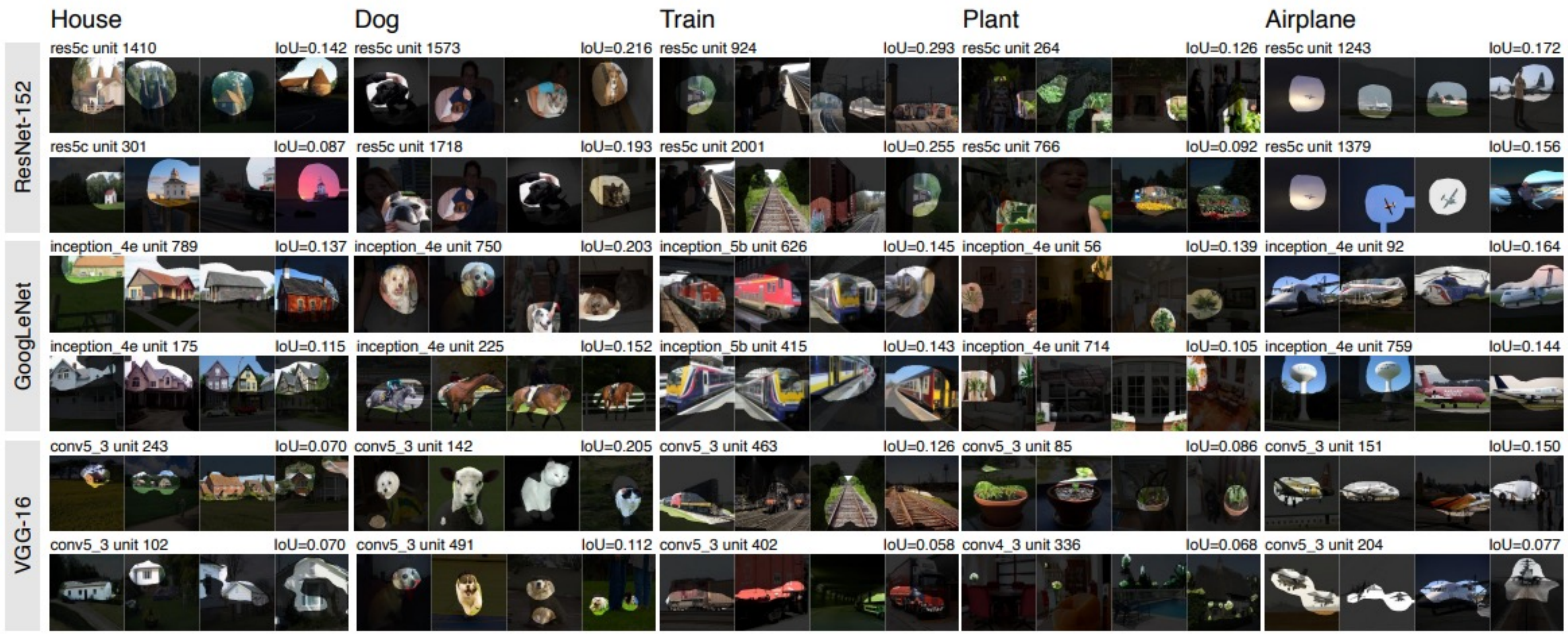
Layer 3



Visualizing and Understanding Convolutional Networks [Zeiler and Fergus, ECCV 2014]

Slide credit: Jia-Bin Huang

Neural Network Dissection

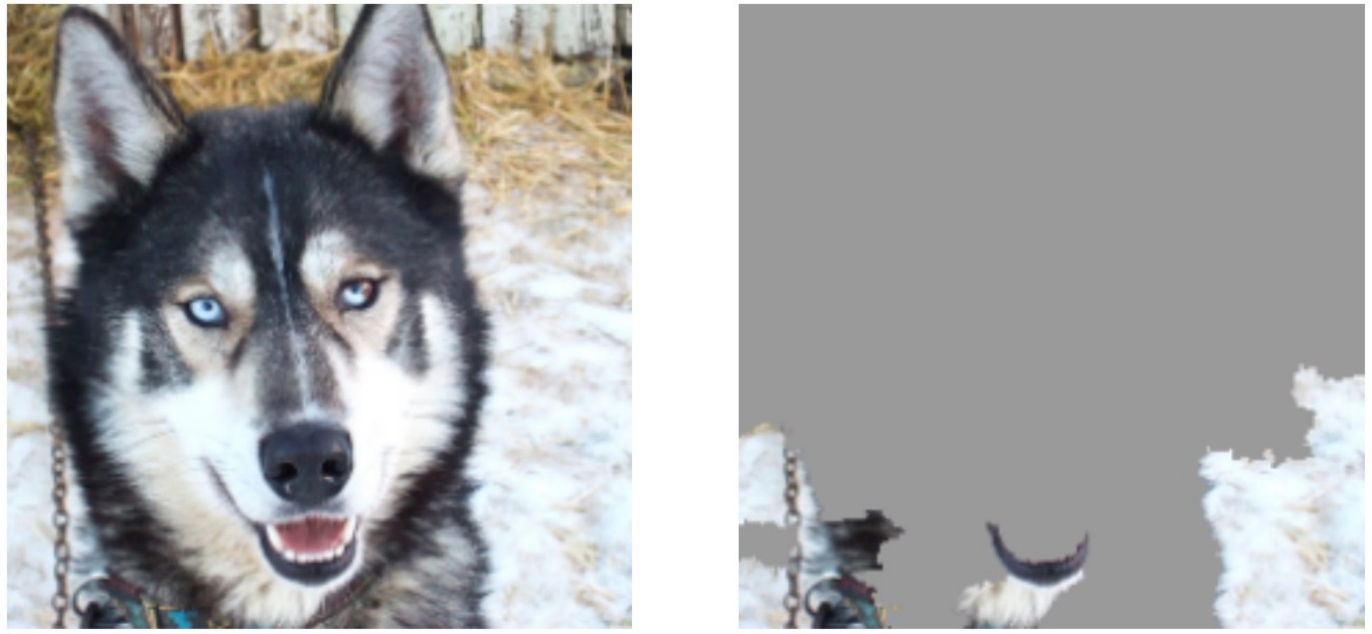

House
Dog
Train
Plant
Airplane
ResNet-152
GoogLeNet
VGG-16
res5c unit 1410 IoU=0.142
res5c unit 1573 IoU=0.216
res5c unit 924 IoU=0.293
res5c unit 264 IoU=0.126
res5c unit 1243 IoU=0.172
res5c unit 301 IoU=0.087
res5c unit 1718 IoU=0.193
res5c unit 2001 IoU=0.255
res5c unit 766 IoU=0.092
res5c unit 1379 IoU=0.156
inception_4e unit 789 IoU=0.137
inception_4e unit 750 IoU=0.203
inception_5b unit 626 IoU=0.145
inception_4e unit 56 IoU=0.139
inception_4e unit 92 IoU=0.164
inception_4e unit 175 IoU=0.115
inception_4e unit 225 IoU=0.152
inception_5b unit 415 IoU=0.143
inception_4e unit 714 IoU=0.105
inception_4e unit 759 IoU=0.144
conv5_3 unit 243 IoU=0.070
conv5_3 unit 142 IoU=0.205
conv5_3 unit 463 IoU=0.126
conv5_3 unit 85 IoU=0.086
conv5_3 unit 151 IoU=0.150
conv5_3 unit 102 IoU=0.070
conv5_3 unit 491 IoU=0.112
conv5_3 unit 402 IoU=0.058
conv4_3 unit 336 IoU=0.068
conv5_3 unit 204 IoU=0.077

http://netdissect.csail.mit.edu/

Why Are Explanations Useful?

- Models do not always use the information we expect them to!

An Interesting Local Explanation



(a) Husky classified as wolf (b) Explanation

Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

Ribeiro et al., "Why Should I Trust You? Explaining the Predictions of Any Classifier", 2016

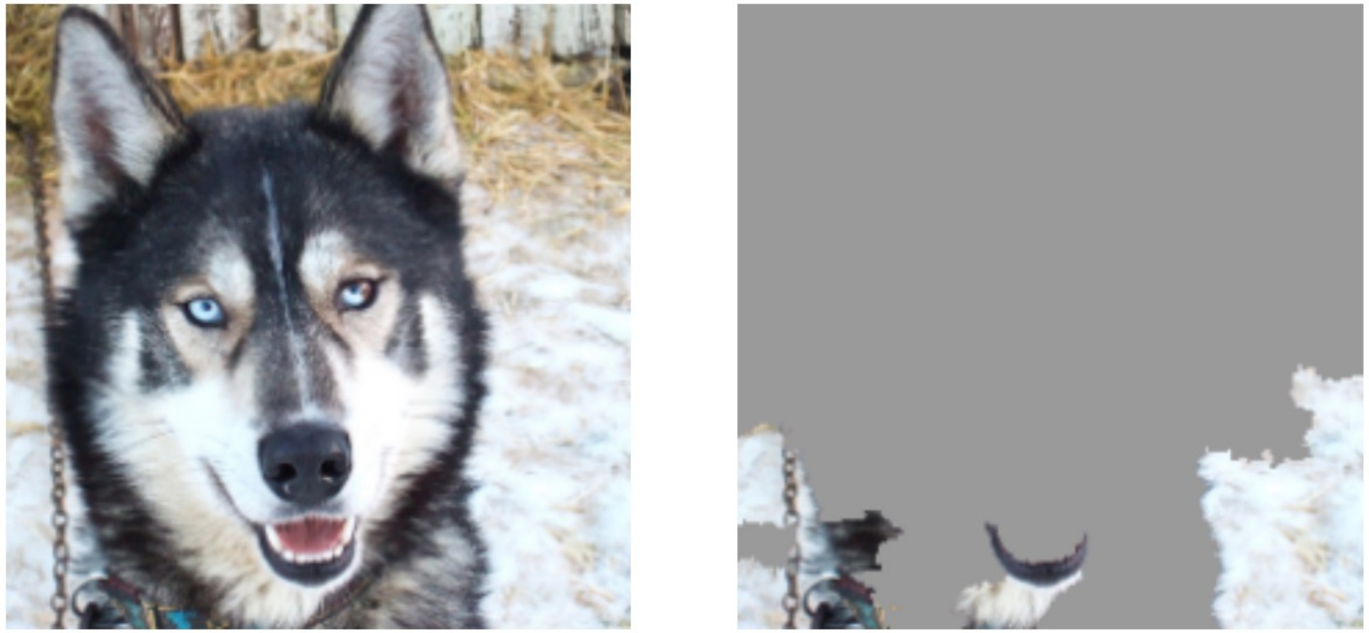
Correlated Inputs/Features

- Suppose two features x_1 and x_2 are highly correlated

- Which one should the model use to predict the label y?
  - Doesn't make a difference!

Ribeiro et al., "Why Should I Trust You? Explaining the Predictions of Any Classifier", 2016

Correlated Inputs/Features



(a) Husky classified as wolf (b) Explanation

Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

Ribeiro et al., "Why Should I Trust You? Explaining the Predictions of Any Classifier", 2016

Problematic Correlations

- In practice, unexpected features can be correlated with the output

- **Example**
  - Model predicts "has asthma" → "lower pneumonia risk"
  - Why?

- **Explanation**
  - A patient who has asthma is more careful and receives better medical care
  - **Patients with asthma have better outcomes for pneumonia!**
  - **Does not mean we should label asthma patients as lower risk!**

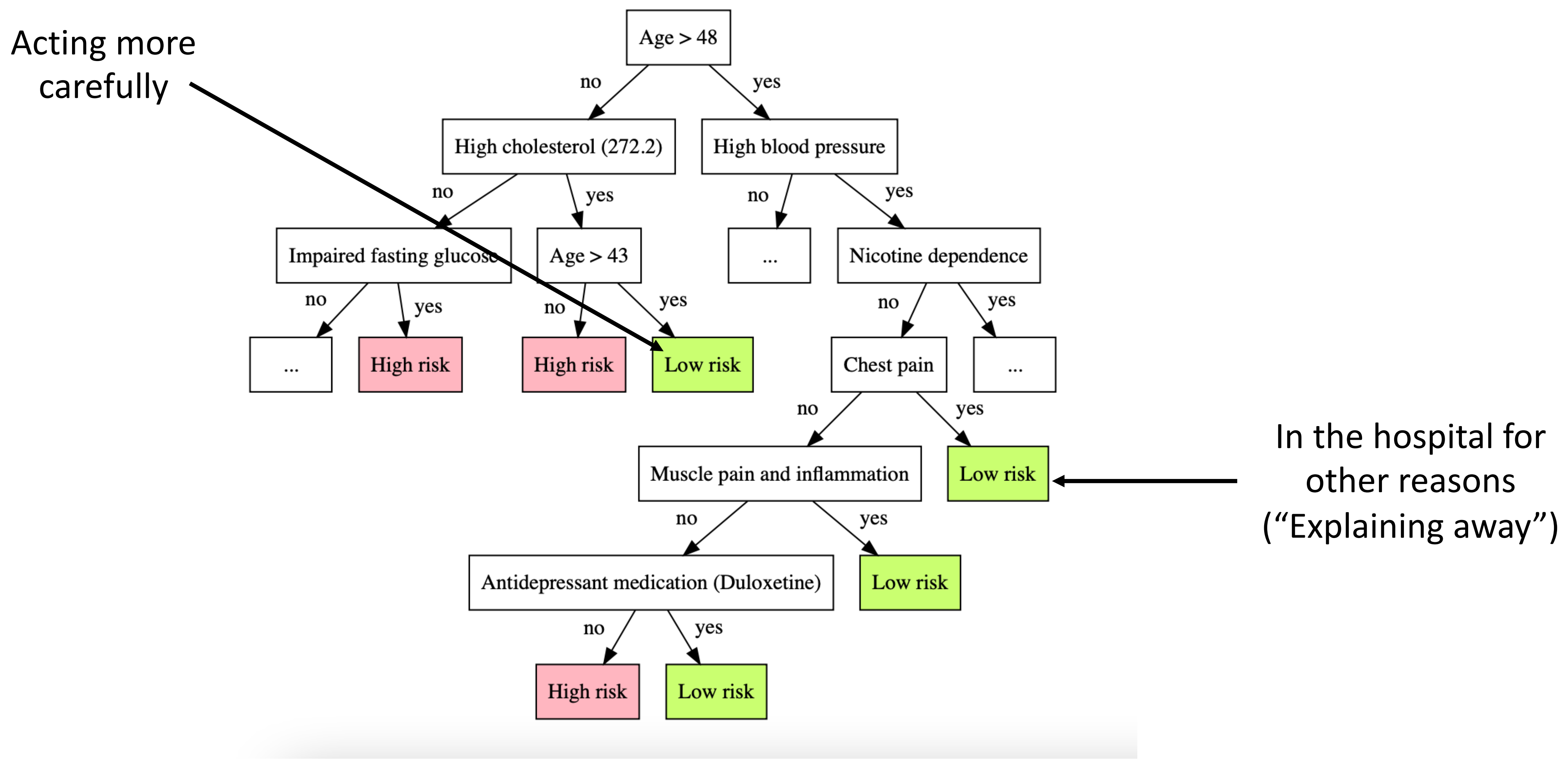
Caruana et al., "Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission", 2015

Example: Diabetes prediction

- **Input:** ~400 patient features (e.g., lab tests, current medications, etc.)
- **Label:** Does the patient have diabetes?
- Train a decision tree to solve this problem

Bastani et al., “Interpreting Blackbox Models via Model Extraction”, 2017

Example: Diabetes prediction



Bastani et al., “Interpreting Blackbox Models via Model Extraction”, 2017

Example: Chest X-Rays

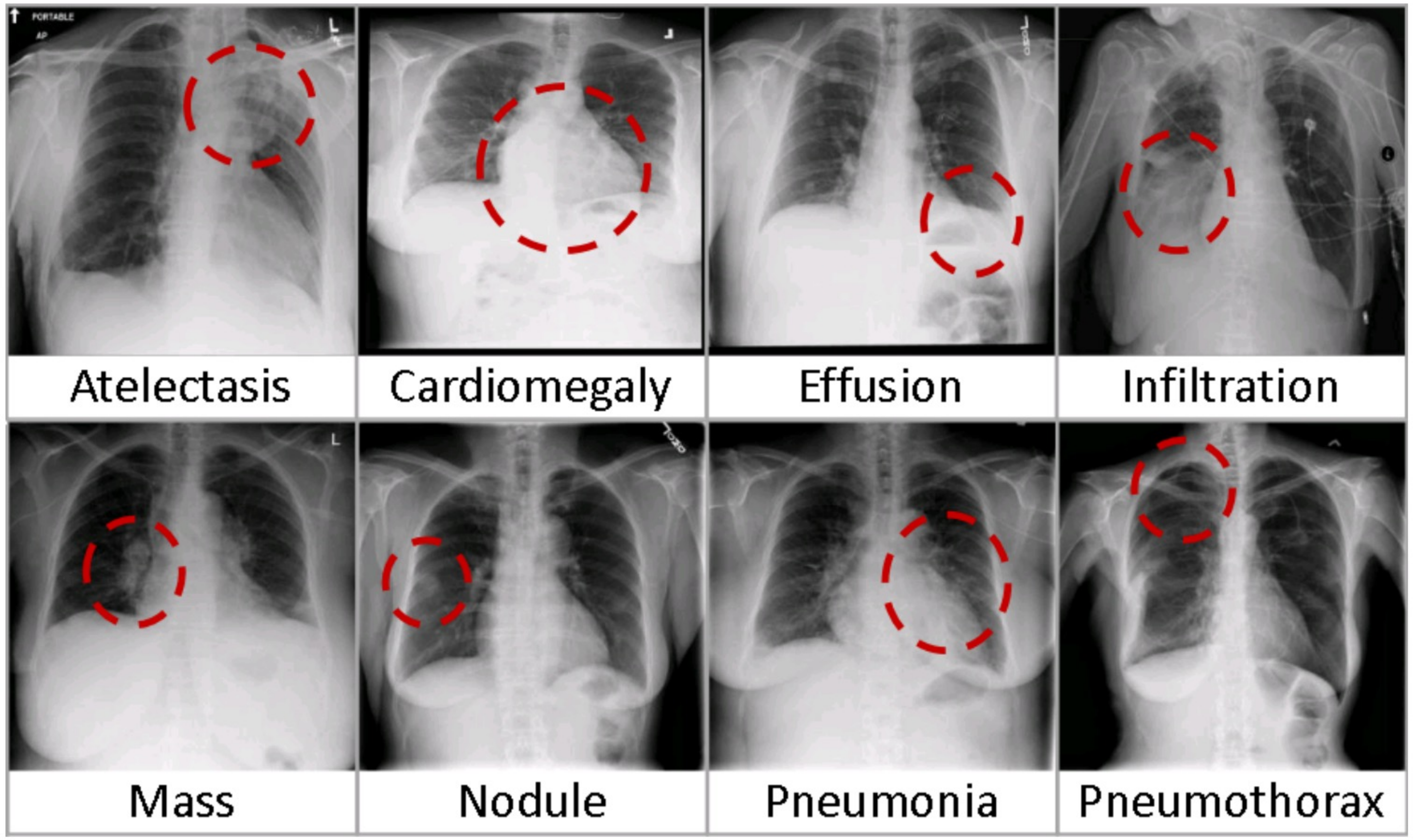


Figure 1. *Eight common thoracic diseases observed in chest X-rays that validate a challenging task of fully-automated diagnosis.*

Wang et al., ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases, 2017

Example: Chest X-Rays

- **Task:** Diagnose pneumothorax from chest x-ray
- **Problem:** Some of the patients were already treated!
  - Treatment is visible in chest x-ray
  - Deep neural network is predicting who was already treated!

Oakden-Rayner, Exploring large scale public medical image datasets, 2017

Potential Solutions

- **No general solutions (yet)**
- **Good practices**
  - Be very careful with data processing/cleaning
  - Use existing interpretability techniques to better understand model
  - Work closely with domain experts to examine potential data/model issues

Agenda

- **Interpretability & Explainability**
- **Robustness to distribution shift**
- **Robustness to adversarial attacks**

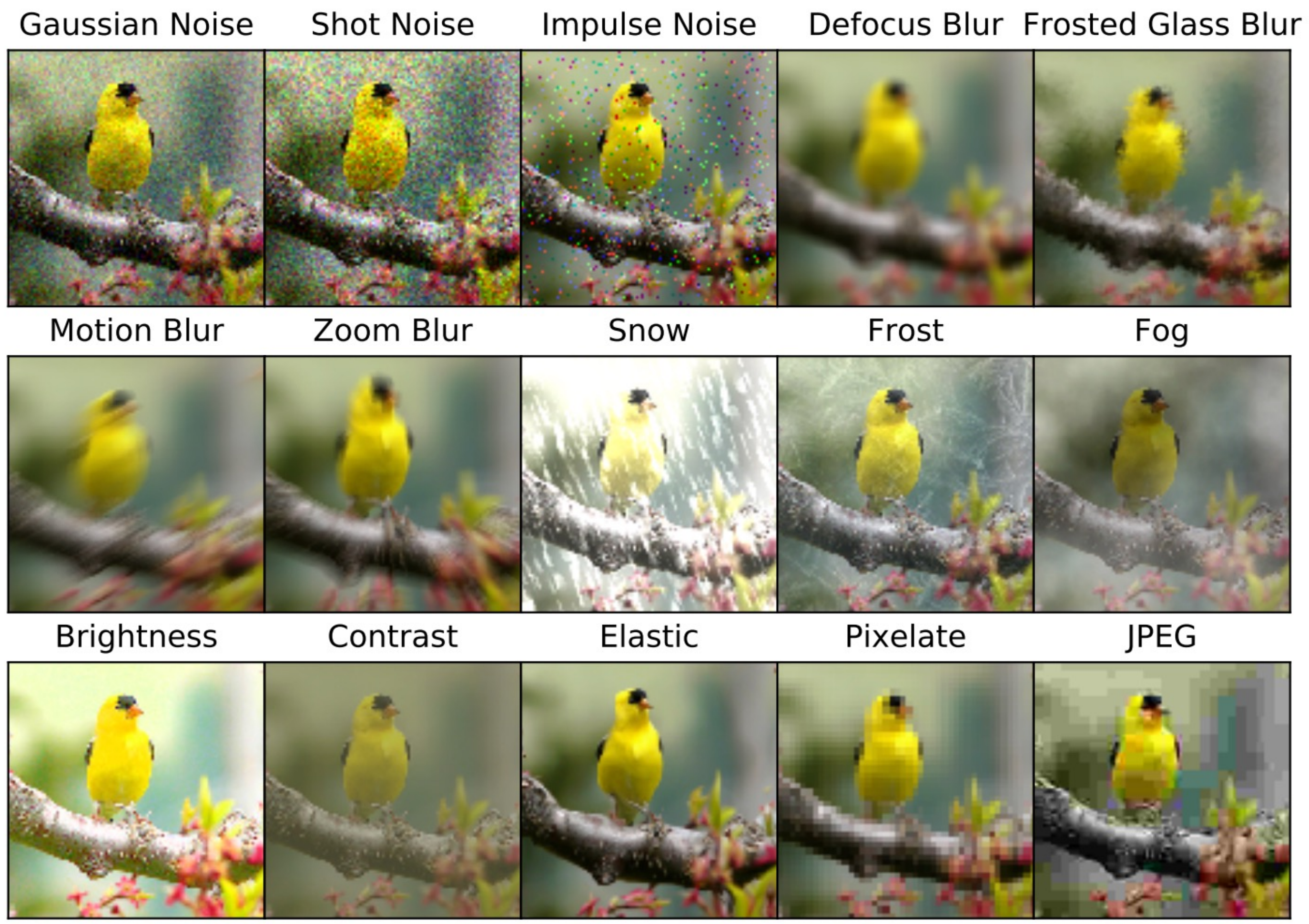
Robustness to Distribution Shift

- Neural networks generalize well **on distribution**
- **Ideal scenario**
  - Test set and training set are i.i.d. from the same distribution
  - **Equivalently:** Test set is obtained by shuffling entire dataset and then splitting
- **Often fails in practice! "Distribution shift"**

Robustness to Distribution Shift

- **Images/computer vision**
  - Added noise, color shifts, lighting changes, different resolution, etc.
- **Audio/speech-to-text**
  - Noisy background, changes in recording device, etc.
- **Natural language processing**
  - Substitute synonyms, add unrelated text, etc.

Example: Synthetic Perturbations



Example: Synthetic Perturbations

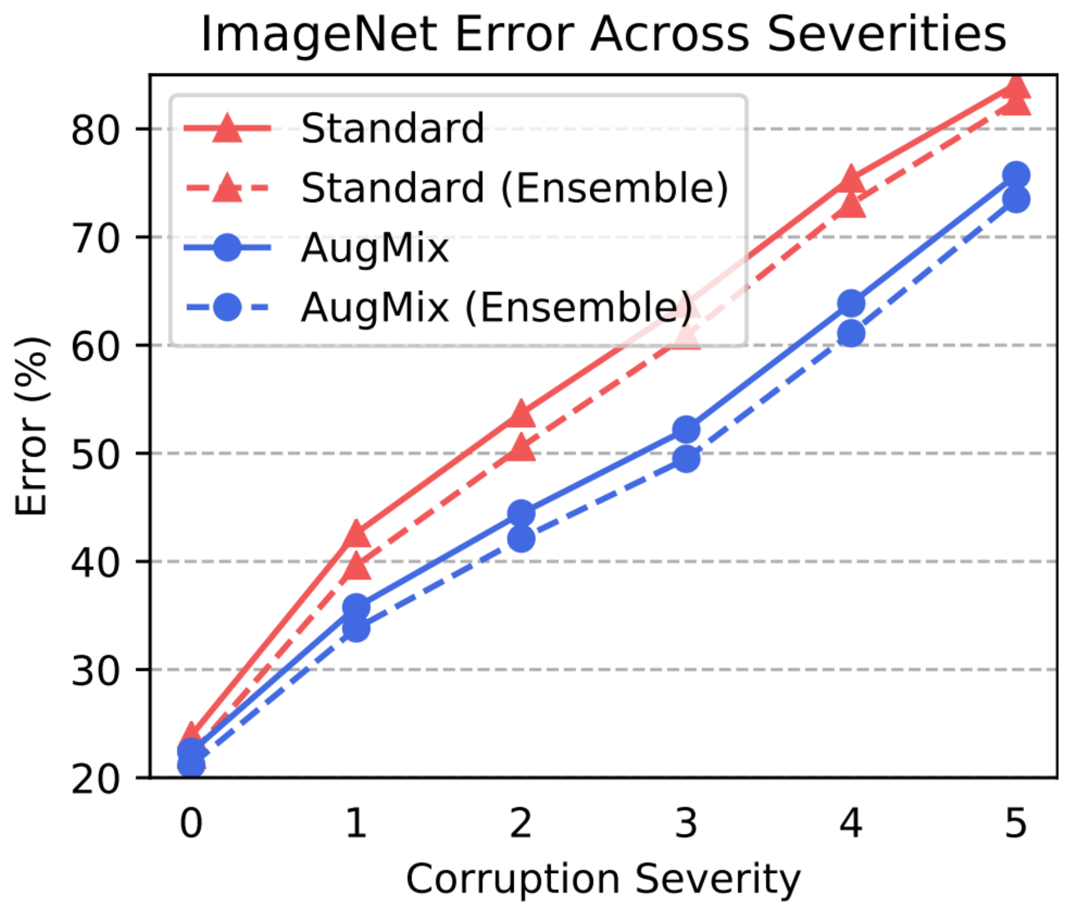
- **Question:** Why should the model be robust?
- **Answer:** Humans are robust!

Example: Synthetic Perturbations

- **Significantly reduces performance**
  - 20% error rate → 80% error rate

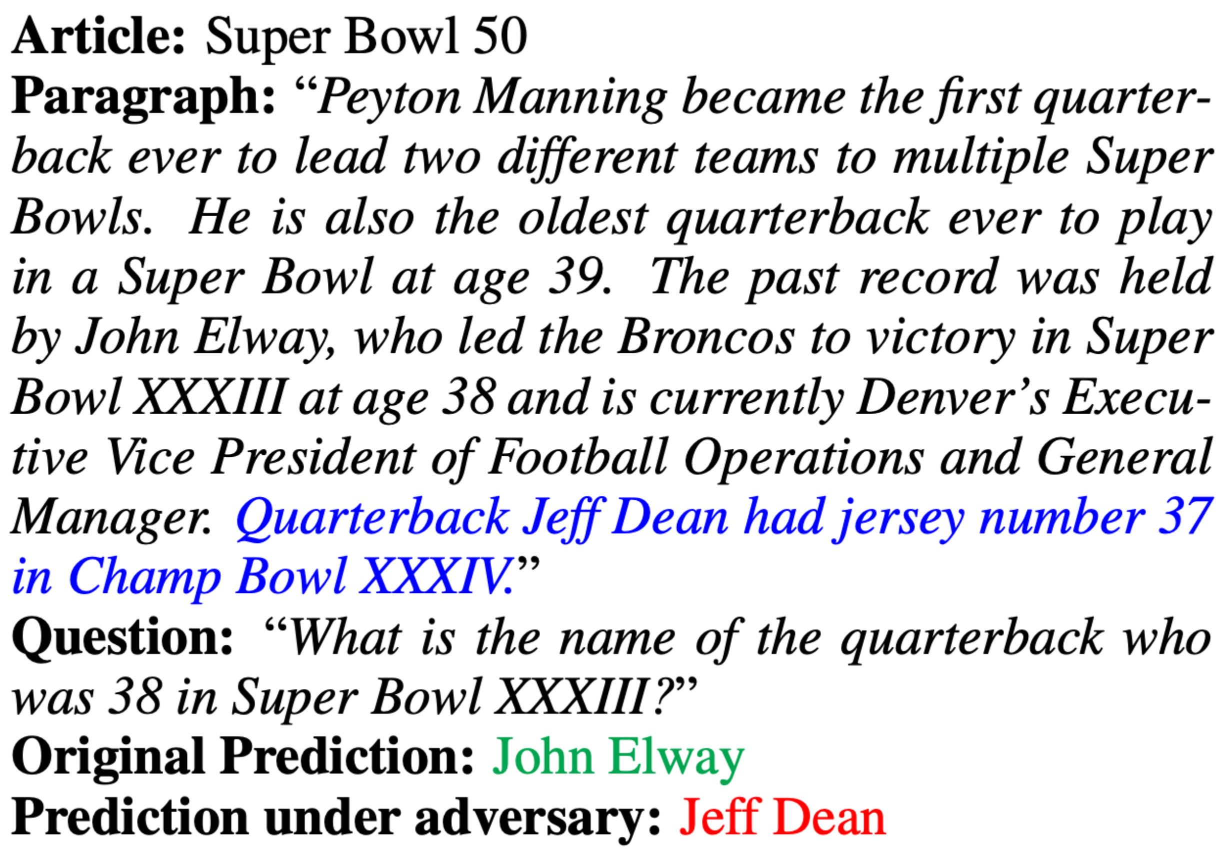
- **Data augmentation can help (but not 100% solution)**

Example: Synthetic Perturbations



Hendrycks et al., AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty, 2020

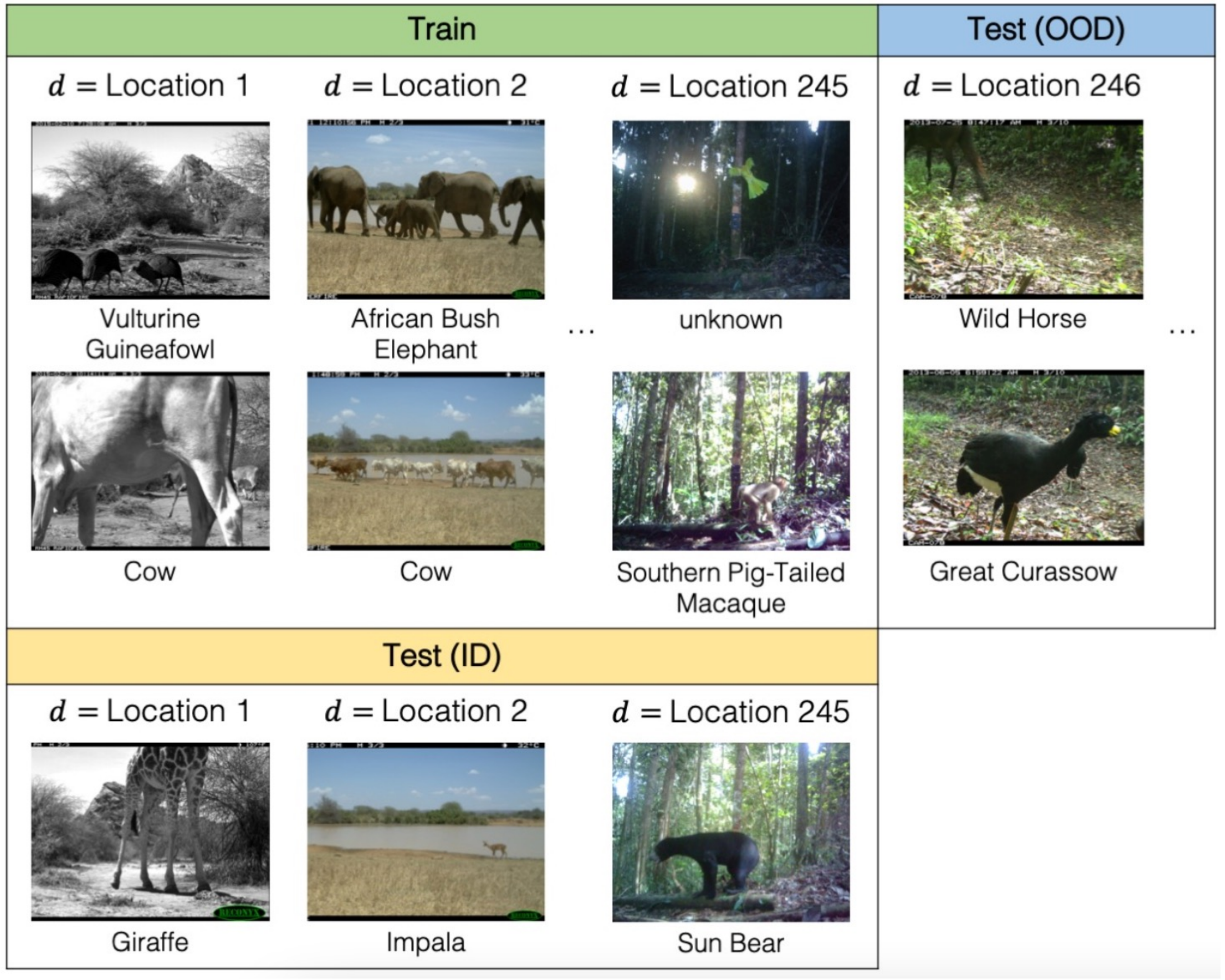
Example: Natural Language Processing



Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

Jia & Liang, Adversarial Examples for Evaluating Reading Comprehension Systems, 2021

Example: Real Perturbations



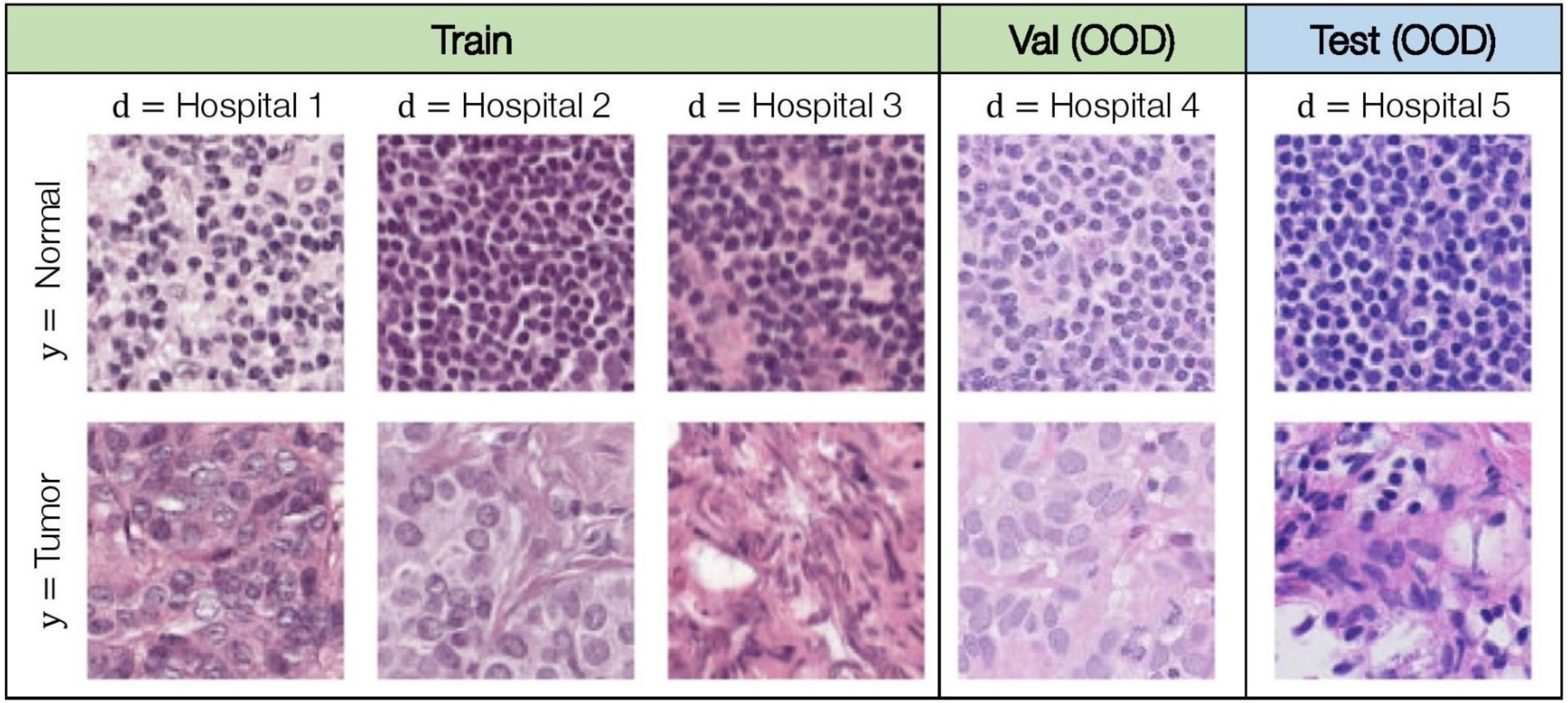
Koh et al., WILDS: A Benchmark of in-the-Wild Distribution Shifts, 2020

Example: Real Perturbations

	Train			Test	
Satellite Image (x)					
Year / Region (d)	2002 / Americas	2009 / Africa	2012 / Europe	2016 / Americas	2017 / Africa
Building / Land Type (y)	shopping mall	multi-unit residential	road bridge	recreational facility	educational institution

Koh et al., WILDS: A Benchmark of in-the-Wild Distribution Shifts, 2020

Example: Real Perturbations



Koh et al., WILDS: A Benchmark of in-the-Wild Distribution Shifts, 2020

Potential Solutions

- **No general strategy (yet)**
- **Good practices**
  - Train on as large & diverse of a dataset as possible
  - Use data augmentation when possible
  - If available, finetune on location-specific dataset (transfer learning)

Agenda

- **Interpretability & Explainability**
- **Robustness to distribution shift**
- **Robustness to adversarial attacks**

Robustness to Adversarial Attacks

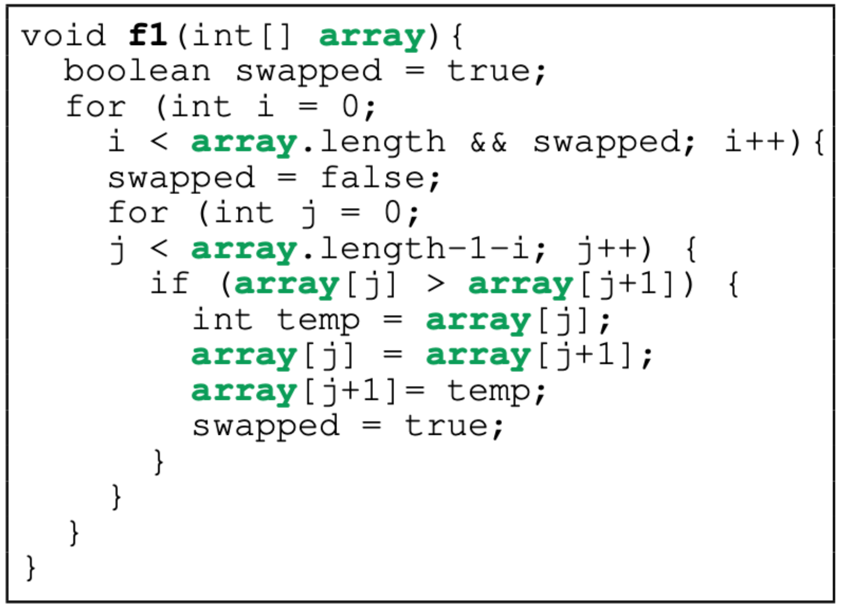
- **Example:**
  - Want to reject email attachment if it contains malicious code
  - Use machine learning to predict if code is malicious
- **What can go wrong?**
  - Attacker perturbs code (e.g., add random lines of dead code) until it is labeled benign by the machine learning model!
  - **Strong form of robustness is needed**

Example: Function Name Prediction

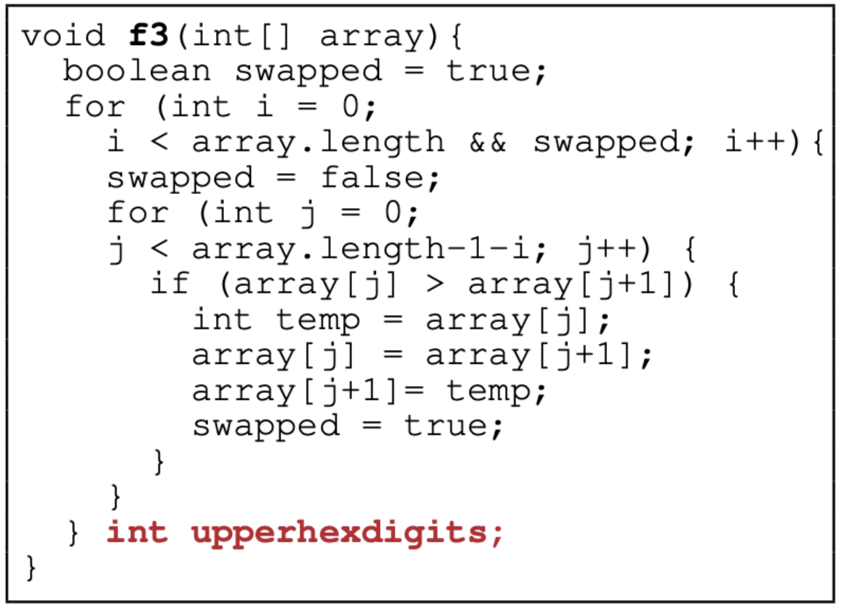
- **Task:** Given a function (e.g., as a string), predict its name
- **Attack:** Add a random line of irrelevant code

Yefet et al., Adversarial examples for models of code, 2019

Example: Function Name Prediction



```
void f1(int[] array){
  boolean swapped = true;
  for (int i = 0;
    i < array.length && swapped; i++){
    swapped = false;
    for (int j = 0;
    j < array.length-1-i; j++) {
      if (array[j] > array[j+1]) {
        int temp = array[j];
        array[j] = array[j+1];
        array[j+1]= temp;
        swapped = true;
      }
    }
  }
}
```



```
void f3(int[] array){
  boolean swapped = true;
  for (int i = 0;
    i < array.length && swapped; i++){
    swapped = false;
    for (int j = 0;
    j < array.length-1-i; j++) {
      if (array[j] > array[j+1]) {
        int temp = array[j];
        array[j] = array[j+1];
        array[j+1]= temp;
        swapped = true;
      }
    }
  } int upperhexdigits;
}
```

Prediction: **sort** (98.54%)

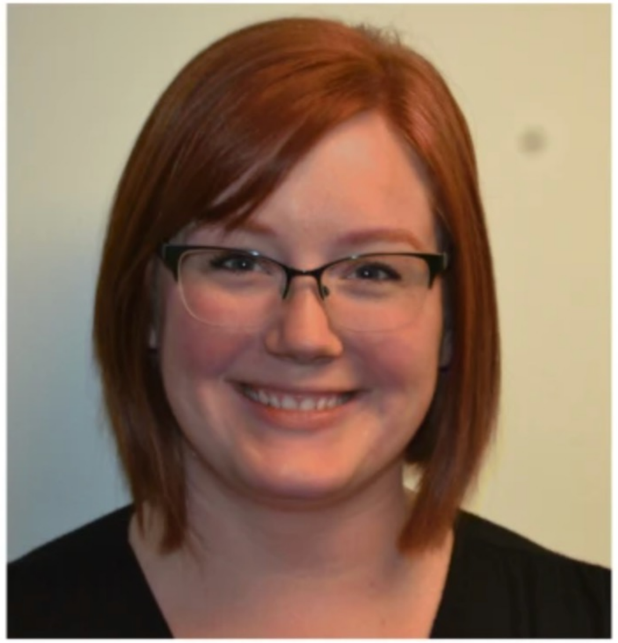
Prediction: **escape** (100%)

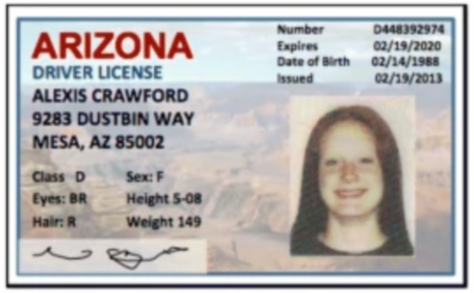
Yefet et al., Adversarial examples for models of code, 2019

Robustness to Adversarial Perturbations

- **Task:**
  - Photo ID verification
  - Goal is to check whether uploaded photo matches a photo ID

- **Attack:**
  - User perturbs their image to match the photo in the ID
  - Challenge for machine learning in online identity verification!





ARIZONA
DRIVER LICENSE
ALEXIS CRAWFORD
9283 DUSTBIN WAY
MESA, AZ 85002

Number	D448392974
Expires	02/19/2020
Date of Birth	02/14/1988
Issued	02/19/2013

Class D Sex: F
Eyes: BR Height 5-08
Hair: R Weight 149

(Valid photo ID from Papesh 2018)

Robustness to Adversarial Perturbations

- **Robustness:** Similar images ⇒ same label
- **Goal:** Robust to **any** small perturbation in **some family**
  - **Note:** Very far from solving this problem
- **Key question:** What is “some family”?

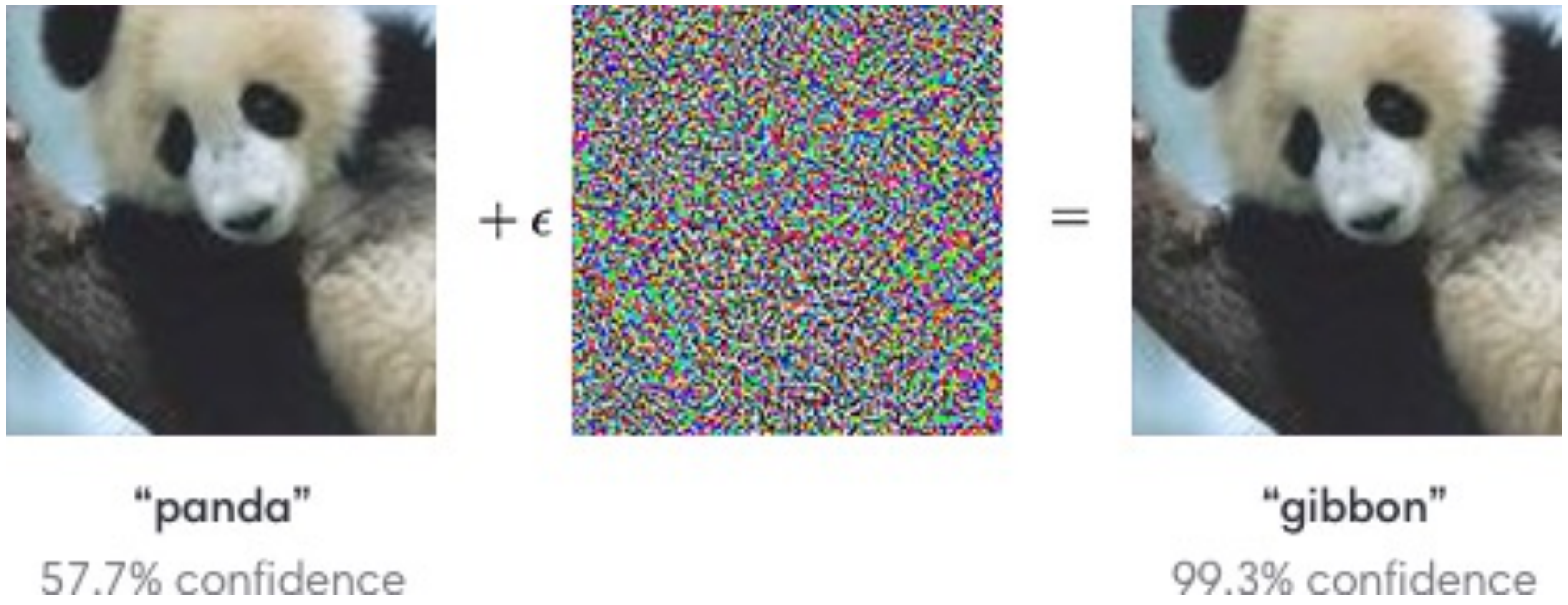
Robustness to Adversarial Perturbations

- **(Very limited) example for images:**

$$\|x - x'\|_\infty \leq \epsilon \Rightarrow \text{same label}$$

- **Question:** Why should the model be robust to these perturbations?
  - Should not change the label
  - Humans are robust!

Robustness to Adversarial Perturbations



Szegedy et al., Intriguing Properties of Neural Networks, 2014

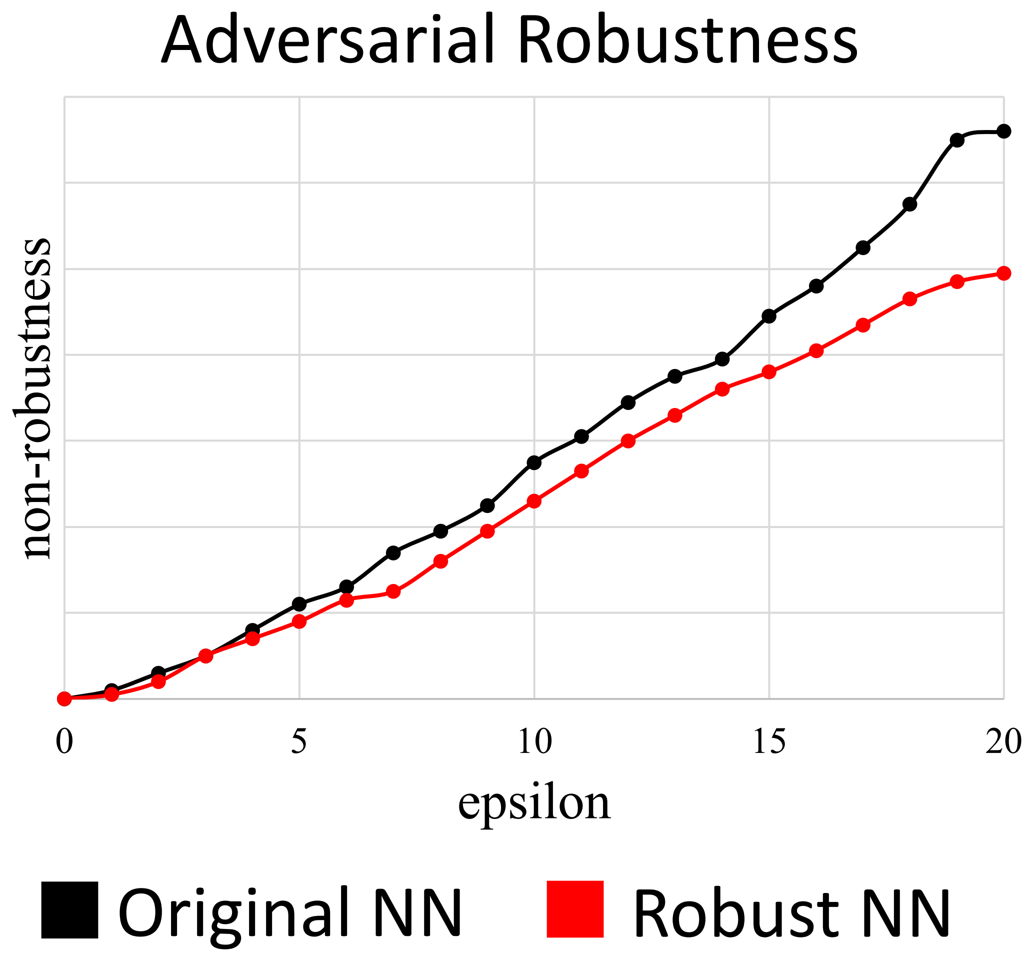
Robustness to Adversarial Perturbations

- **Strategy for improving adversarial robustness**
  - Data augmentation!
  - **Adversarial training:** Use adversary to generate new examples for training

- Does it work?

Goodfellow et al., Explaining and harnessing adversarial examples, 2015

Improving Robustness?



Goodfellow et al., Explaining and harnessing adversarial examples, 2015

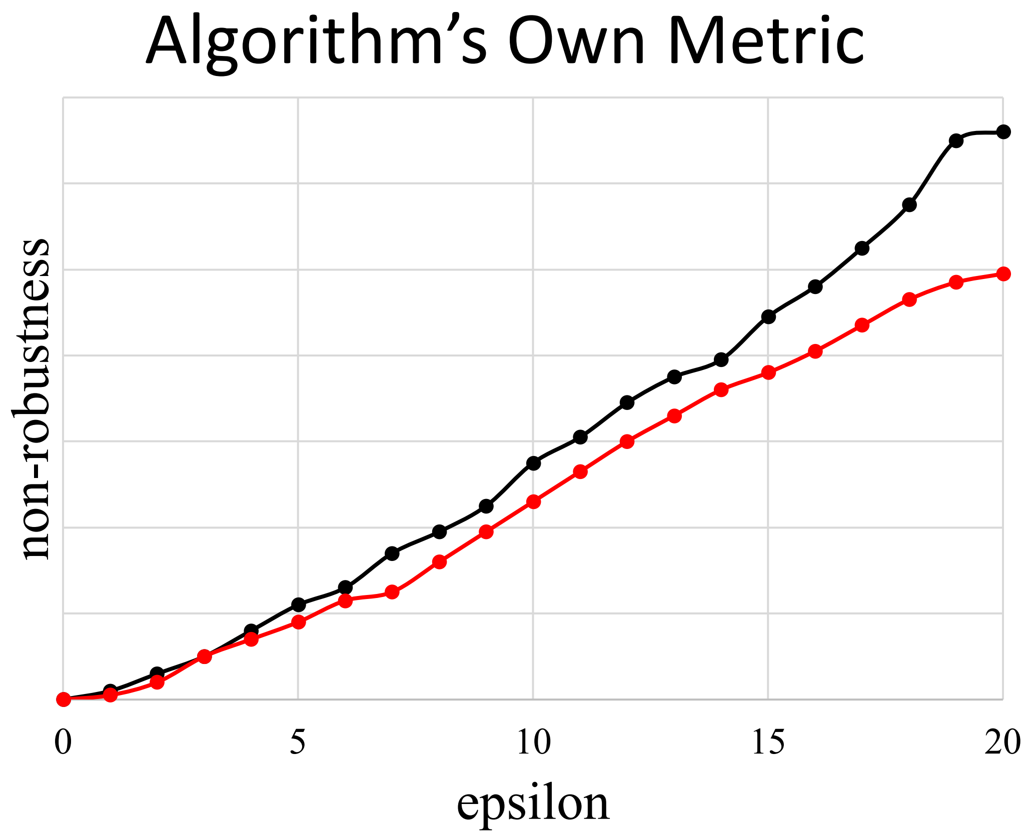
Improving Robustness?

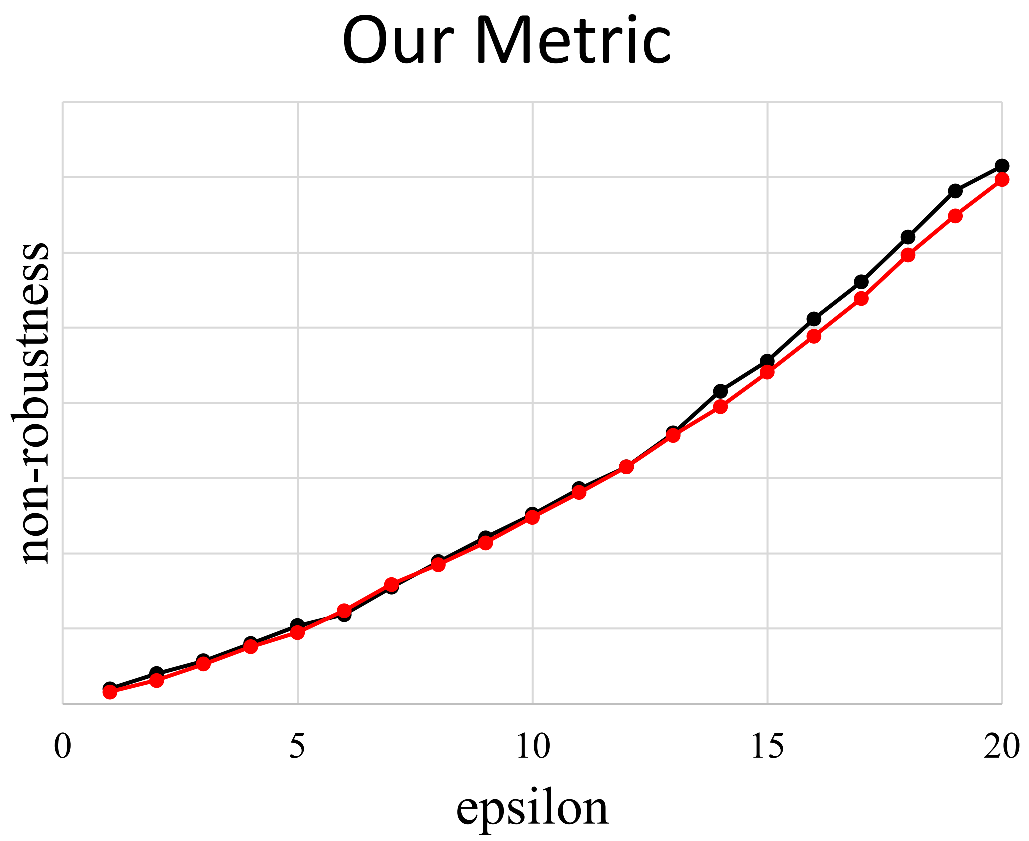
- **Problem**
  - Only robust to the current adversary
  - What if the adversary changes? **Distribution shift!**

- **Example**
  - Adversarial training using one adversary
  - Test against a more powerful adversary

Bastani et al., Measuring robustness of neural networks via constraints, 2016

Improving Robustness?





■ Original NN 🟥 Robust NN

Bastani et al., Measuring robustness of neural networks via constraints, 2016

Potential Solutions

- **No general strategy (yet)**

- **Good practices**
  - Use the strongest adversary you can design
  - Use variety of different adversaries

Can Uncertainty Help?

- **Recall:** Most neural networks predict an uncertainty

$$p_\beta(y \mid x)$$

- **Idea:** Can we use uncertainty to detect adversarial attacks?

- **Answer:** No!
  - Adversarial examples can have very high confidence
  - Probabilities can be overconfident even for normal test examples!

Potential Solutions

- **General solutions for non-adversarial setting:** Calibrated prediction
- **Intuition:** Among examples where neural network predicts it is correct with probability p, it is correct for a fraction $\approx p$
- **Algorithms:** Temperature scaling, isotonic regression, etc.

Guo et al., On the calibration of modern neural networks, 2017

Potential Solutions

- **No general solutions for adversarial setting**

- **Good practices**
  - Don’t blindly trust predicted probabilities!

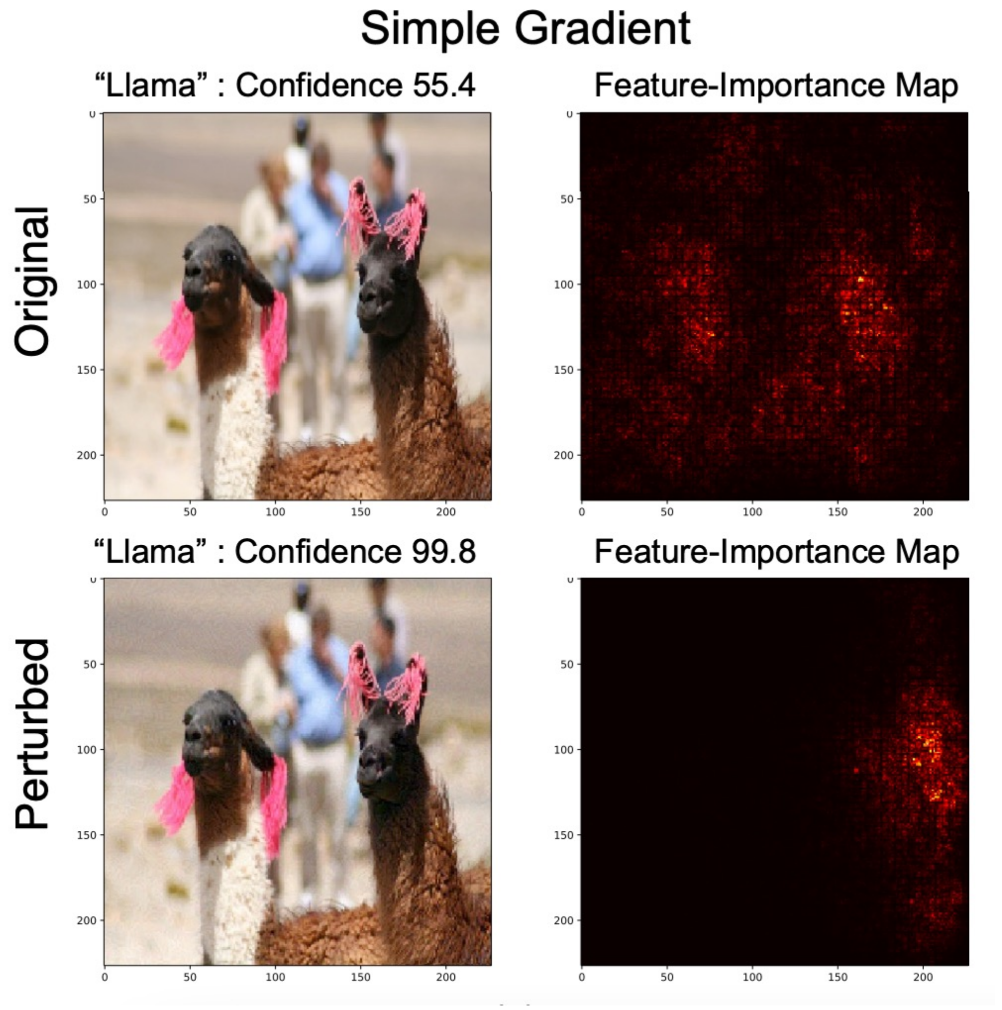
Can Explanations Help?

- **Idea:** Check if explanation makes sense
- **Question:** Are explanations of neural networks robust?

$$\mathrm{Explain}(x + \epsilon) \approx \mathrm{Explain}(x)$$

- **Answer:** No!
- Not even robust to distribution shift

Fragility of Explanations



Ghorbani et al., Interpretation of neural networks is fragile, 2017

Fragility of Explanations

- Not just a problem for neural networks!

```
If  Race ≠ African American:
        If Prior-Felony = Yes and Crime-Status = Active, then Risky
        If Prior-Convictions = 0, then Not Risky

If Race = African American:
        If Pays-rent = No and Gender = Male, then Risky
        If Lives-with-Partner = No and College = No, then Risky
        If Age ≥35 and Has-Kids = Yes, then Not Risky
        If Wages ≥70K, then Not Risky

Default: Not Risky
```

```
If  Current-Offense = Felony:
        If Prior-FTA = Yes and Prior-Arrests ≥ 1, then Risky
        If Crime-Status = Active and Owns-House = No and Has-Kids = No, then Risky
        If Prior-Convictions = 0 and College = Yes and Owns-House = Yes, then Not Risky

If Current-Offense = Misdemeanor and Prior-Arrests > 1:
        If Prior-Jail-Incarcerations = Yes, then Risky
        If Has-Kids = Yes and Married = Yes and Owns-House = Yes, then Not Risky
        If Lives-with-Partner = Yes and College = Yes and Pays-Rent = Yes, then Not Risky

If Current-Offense = Misdemeanor and Prior-Arrests ≤ 1:
        If Has-Kids = No and Owns-House = No and Prior-Jail-Incarcerations = Yes, then Risky
        If Age ≥ 50 and Has-Kids = Yes and Prior-FTA = No, then Not Risky

Default: Not Risky
```

Lakkaraju & Bastani, “How do I fool you?": Manipulating User Trust via Misleading Black Box Explanations, 2020

Misleading Explanations

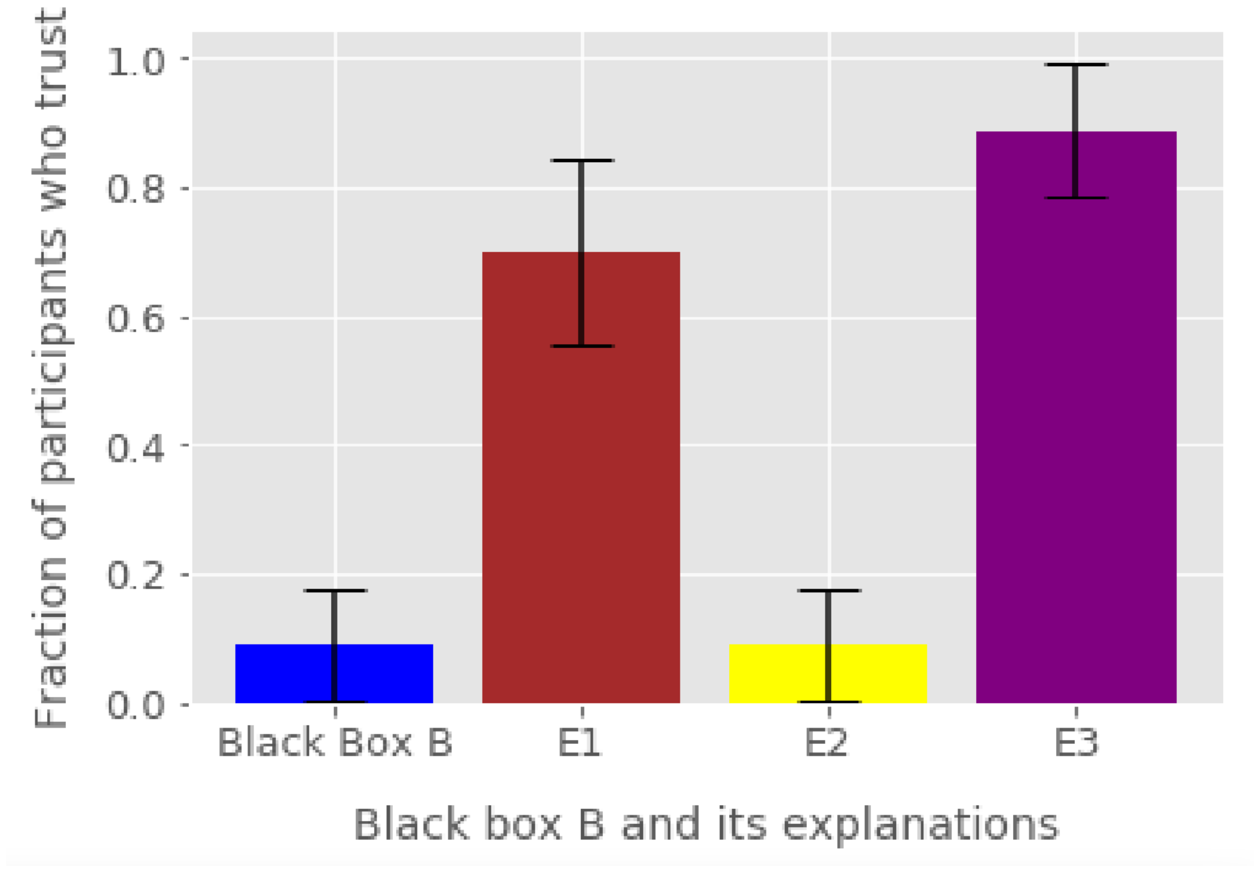
- Can construct explanations to mislead users into trusting a model

- **Strategy**
  - Design a set of features that users believe are trustworthy
  - Generate an explanation that highlights these features as important

- Users believe the model is using trustworthy features even if it is not

Lakkaraju & Bastani, “How do I fool you?": Manipulating User Trust via Misleading Black Box Explanations, 2020

Misleading Explanations



E1 & E3 are misleading explanations

Lakkaraju & Bastani, “How do I fool you?": Manipulating User Trust via Misleading Black Box Explanations, 2020

Potential Solutions

- **No general strategy (yet)**

- **Good practices**
  - Be careful when interpreting explanations!

Conclusion

- Robustness and interpretability remain key challenges for neural networks (and machine learning more broadly)

- **Good practices**
  - Use variety of techniques to try and understand what models are doing (interpretation, extensive testing on different examples, etc.)
  - Be careful when training models!
  - **Monitor performance of models running in production**

- Lots of ongoing research!