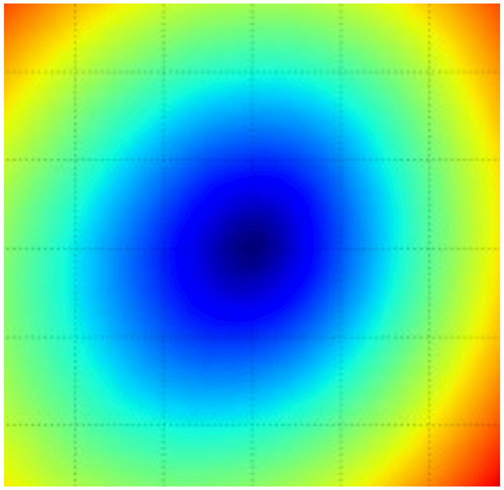
CS4670/5670: Computer Vision

Kavita Bala

Lecture 34: Recognition Basics 2



Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Administrative

- My office hours: moved, and then unmoved
  - So still right after class till 3pm
- Vote for PA3 artifact

Today

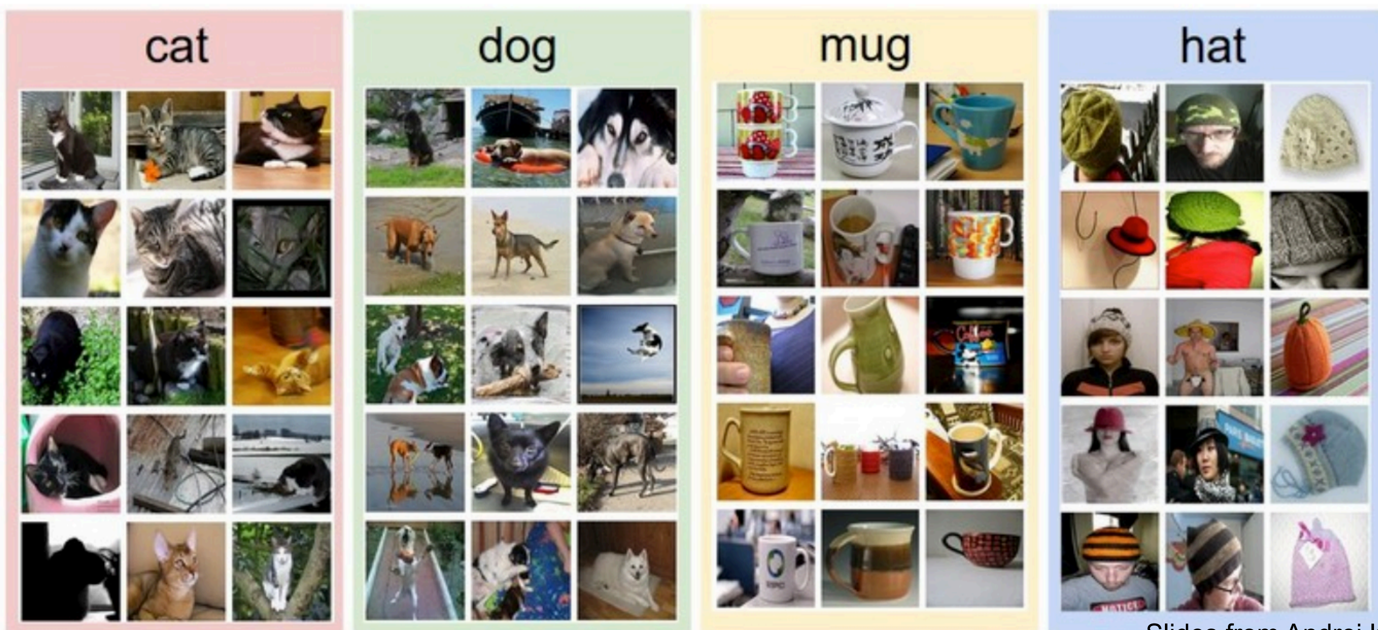
- Score function and loss function

- Then: Deep learning

Data-driven approach

- Collect a database of images with labels
- Use ML to train an image classifier
- Evaluate the classifier on test images

Example training set



Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Score function

class scores

Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Linear Classifier

(assume CIFAR-10 example so
32 x 32 x 3 images, 10 classes)

data (image)
[3072 x 1]

$$f(x_i, W, b) = Wx_i + b$$

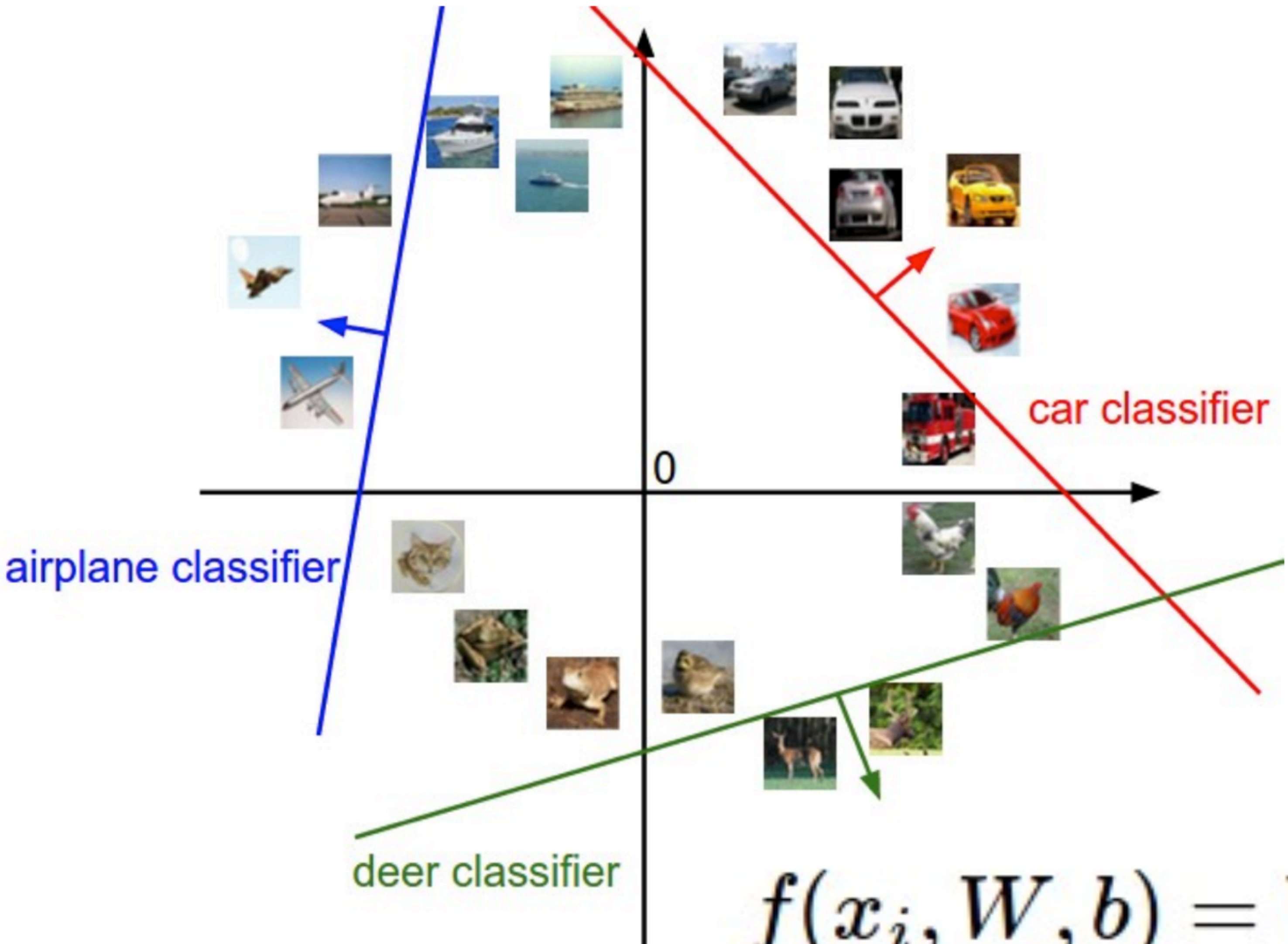
class scores
[10 x 1]

weights
[10 x 3072]

bias vector
[10 x 1]

Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Geometric Interpretation



$$f(x_i, W, b) = Wx_i + b$$

Loss function, cost/objective function

- Given ground truth labels (yi), scores f (xi, W)
  - how unhappy are you with the scores?
- Loss function or objective/cost function
- Want to minimize the loss function

Loss function, cost/objective function

- Given ground truth labels (yi) and scores f(xi, W)
  - how unhappy are you with the scores?

$$f(x_i, W) = [\boxed{13}, -7, 11]$$

$$y_i = 0$$

Intuition

Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Loss fn: Multi-class SVM loss

- Given ground truth labels (yi) and scores f(xi, W)
  - how unhappy are you with the scores?

$$L_i = \sum_{j \neq y_i} \max(0, f(x_i, W)_j - f(x_i, W)_{y_i} + \Delta)$$

Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Loss fn: interpretation

$$\boxed{L_i} = \boxed{\sum_{j \neq y_i}} \max(0, \boxed{f(x_i, W)_j - f(x_i, W)_{y_i}} + \Delta)$$

loss due to
example i

sum over all
incorrect labels

difference between the correct class
score and incorrect class score

Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Example: loss

Example: $$L_i = \sum_{j \neq y_i} \max(0, f(x_i, W)_j - f(x_i, W)_{y_i} + \Delta)$$

e.g. 10

$$f(x_i, W) = [13, -7, 11]$$

$$y_i = 0$$

---

loss = ?

Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Example: loss

Example: $$L_i = \sum_{j \neq y_i} \max(0, f(x_i, W)_j - f(x_i, W)_{y_i} + \Delta)$$

e.g. 10

$$f(x_i, W) = [13, -7, 11]$$

$$y_i = 0$$

$$L_i = \max(0, -7 - 13 + 10) + \max(0, 11 - 13 + 10)$$

Need more..

- Regularization
  - Ambiguity: W is not unique
  - If loss is 0, k W also has 0 loss

  - Add a regularization penalty
    - Try to keep the weights low
    - Also, weights can blow up if you don’t have it

Important: regularization

$$R(W) = \sum_k \sum_l W_{k,l}^2$$

$$L = \frac{1}{N} \sum_i \sum_{j \neq y_i} \left[\max(0, f(x_i; W)_j - f(x_i; W)_{y_i} + \Delta) \right] + \boxed{\lambda R(W)}$$

Regularization strength

Determine by cross-validation

Example: regularization

$$x = [1, 1, 1, 1]$$

$$w_1 = [1, 0, 0, 0]$$

$$w_2 = [0.25, 0.25, 0.25, 0.25]$$

$$w_1^T x = w_2^T x = 1$$

Slides from Andrej Karpathy and Fei-Fei Li
http://vision.stanford.edu/teaching/cs231n/

Final loss function

$$L = \frac{1}{N}\sum_i \sum_{j \neq y_i} \left[\max(0, f(x_i; W)_j - f(x_i; W)_{y_i} + \Delta)\right] + \lambda R(W)$$

Can set delta to 1

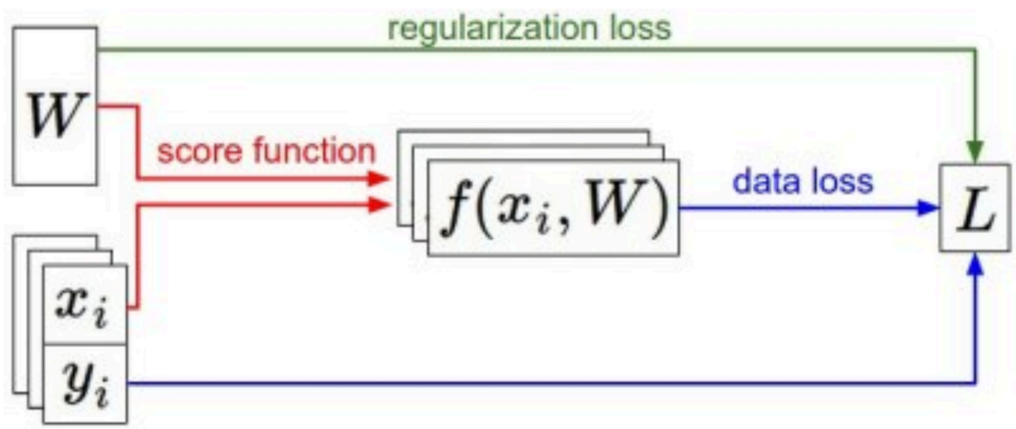
Summary

1. Score function

$$f(x_i, W, b) = Wx_i + b$$

2. Loss function

$$L = \frac{1}{N}\sum_i \sum_{j \neq y_i} \left[\max(0, f(x_i; W)_j - f(x_i; W)_{y_i} + \Delta)\right] + \lambda R(W)$$



http://vision.stanford.edu/teaching/cs231n/linear-classify-demo/

Summary

- Have score function and loss function
  - Will generalize the score function
- Find W and b to minimize loss
  - Minimize loss using gradient descent
- Now to CNNs

Lecture 34: Intro to Deep Learning

CS 4670/5670
Sean Bell

[Monroe 2014, xkcd]

code.flickr.com

Flickr | Flickr Blog | @flickr | @flickrapi | Developer Guidelines | API | Jobs

Search

Posted on October 20, 2014 by Rob Hess, Clayton Mellina, and Friends

← Previous

Introducing: Flickr PARK or BIRD

Zion National Park Utah by Les Haines

OR

Secretary Bird by Bill Gracey

Slide: Flickr

code.flickr.com

flickr.com Flickr jobs!

To play, drag an image from the examples or from your desktop.

PARK or BIRD

Want to know if your photo is from a U.S. national park? Want to know if it contains a bird? Just drag it into the box to the left, and we'll tell you. We'll use the GPS embedded in your photo (if it's there) to see whether it's from a park, and we'll use our super-cool computer vision skills to try to see whether it's a bird (which is a hard problem, but we do a pretty good job at it).

To try it out, just drag any photo from your desktop into the upload box, or try dragging any of our example images. We'll give you your answers below!

Want to know more about PARK or BIRD, including why the heck we did this? Just click here for more info →

PARK? BIRD?

EXAMPLE PHOTOS

Photo credits

Slide: Flickr

code.flickr.com   flickr.com   Flickr jobs!

PARK or BIRD

Want to know if your photo is from a U.S. national park? Want to know if it contains a bird? Just drag it into the box to the left, and we'll tell you. We'll use the GPS embedded in your photo (if it's there) to see whether it's from a park, and we'll use our super-cool computer vision skills to try to see whether it's a bird (which is a hard problem, but we do a pretty good job at it).

To try it out, just drag any photo from your desktop into the upload box, or try dragging any of our example images. We'll give you your answers below!

Want to know more about PARK or BIRD, including why the heck we did this? Just click here for more info → i

EXAMPLE PHOTOS




Photo credits

PARK?

YES

Ah yes, Bryce Canyon is truly beautiful.

BIRD?

NO

Beautiful clouds, but I don't see any birds flying up there.

Slide: Flickr

PARK or BIRD

Want to know if your photo is from a U.S. national park? Want to know if it contains a bird? Just drag it into the box to the left, and we'll tell you. We'll use the GPS embedded in your photo (if it's there) to see whether it's from a park, and we'll use our super-cool computer vision skills to try to see whether it's a bird (which is a hard problem, but we do a pretty good job at it).

To try it out, just drag any photo from your desktop into the upload box, or try dragging any of our example images. We'll give you your answers below!

Want to know more about PARK or BIRD, including why the heck we did this? Just click here for more info →

EXAMPLE PHOTOS

Photo credits

PARK?

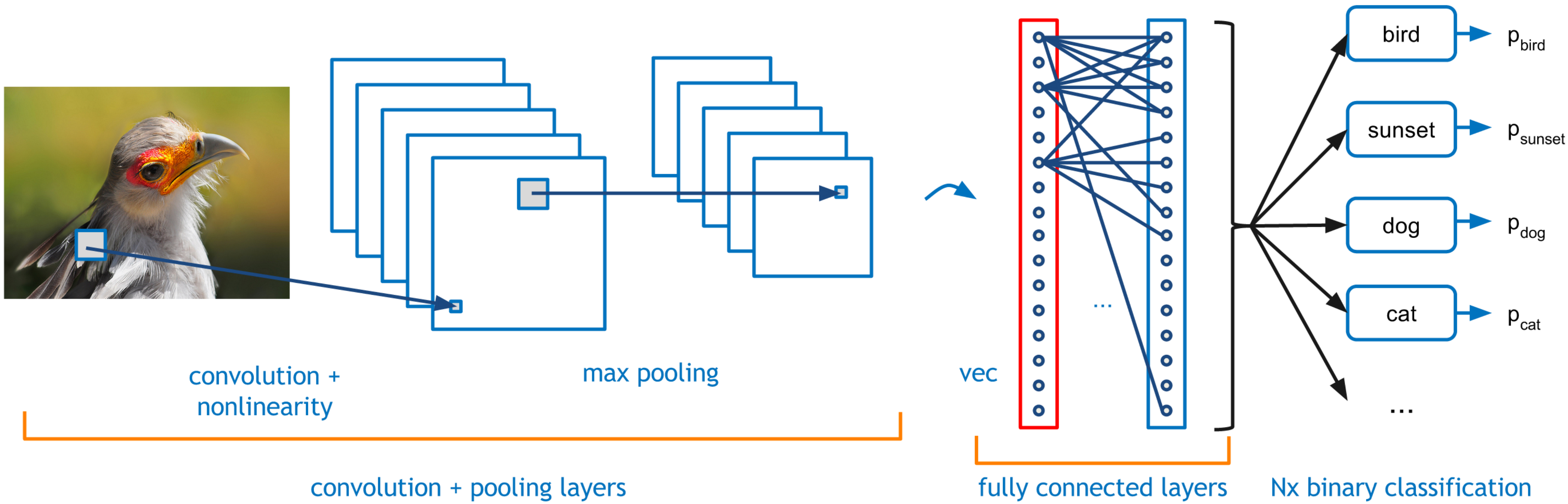
YES

Hey, yeah! I went to Everglades once!

BIRD?

YES

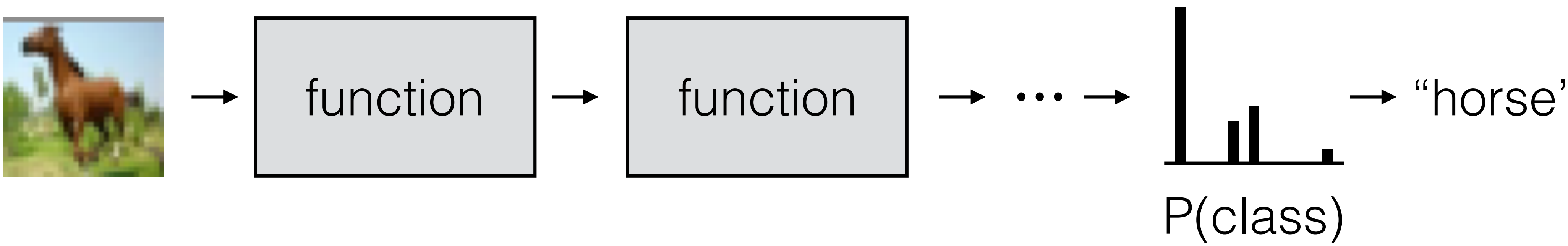
Hey! Nice bird shot!



In the next week, we'll learn what this is, how to compute it, and how to learn it

Slide: Flickr

What is a Convolutional Neural Network (CNN)?

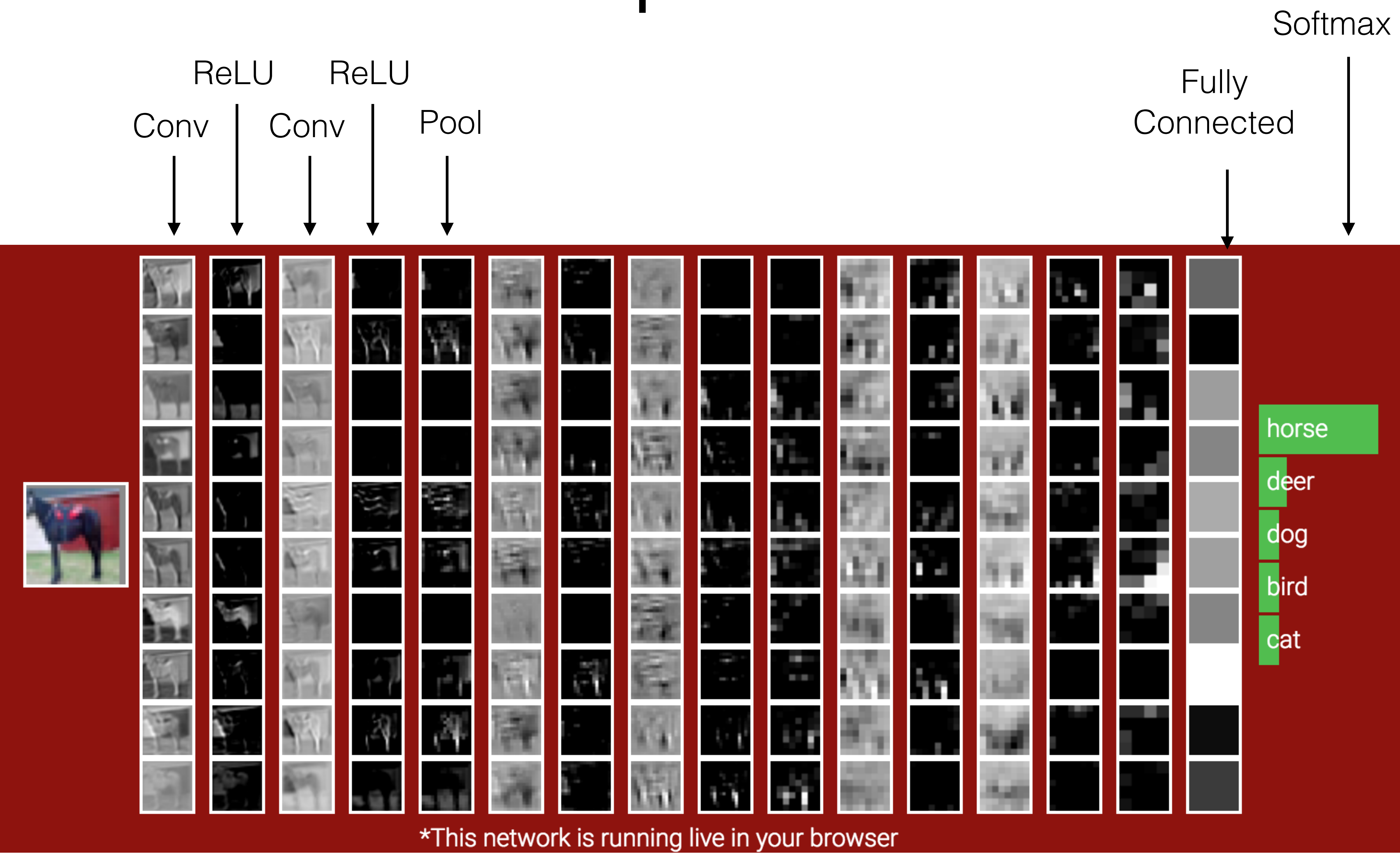


Key questions:

- What kinds of of functions should we use?

- How do we learn the parameters for those functions?

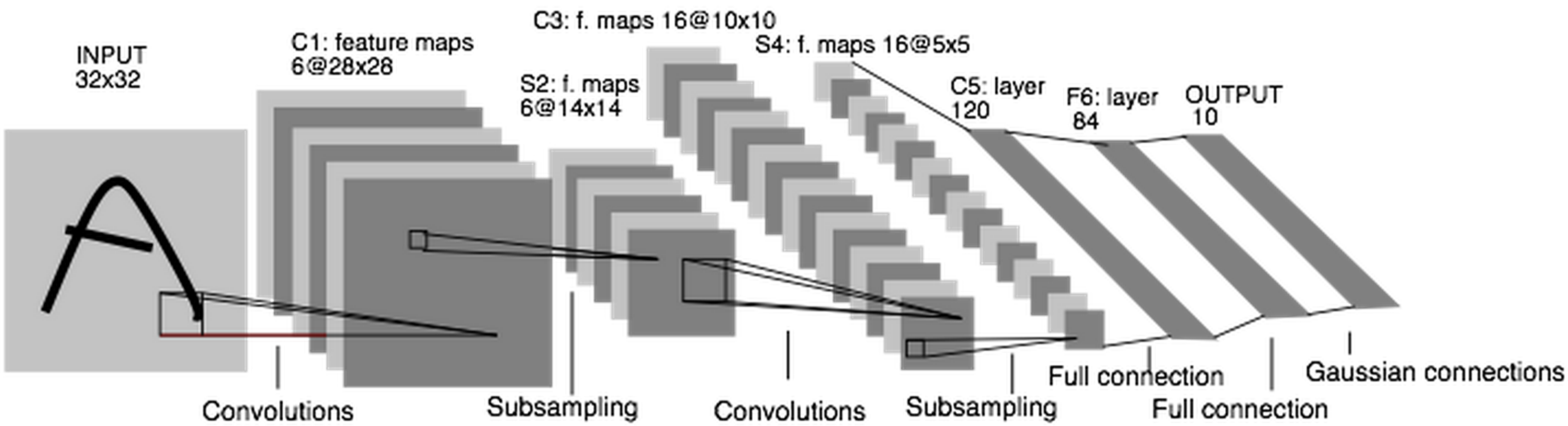
Example CNN



[Andrej Karpathy]

CNNs in 1989: “LeNet”

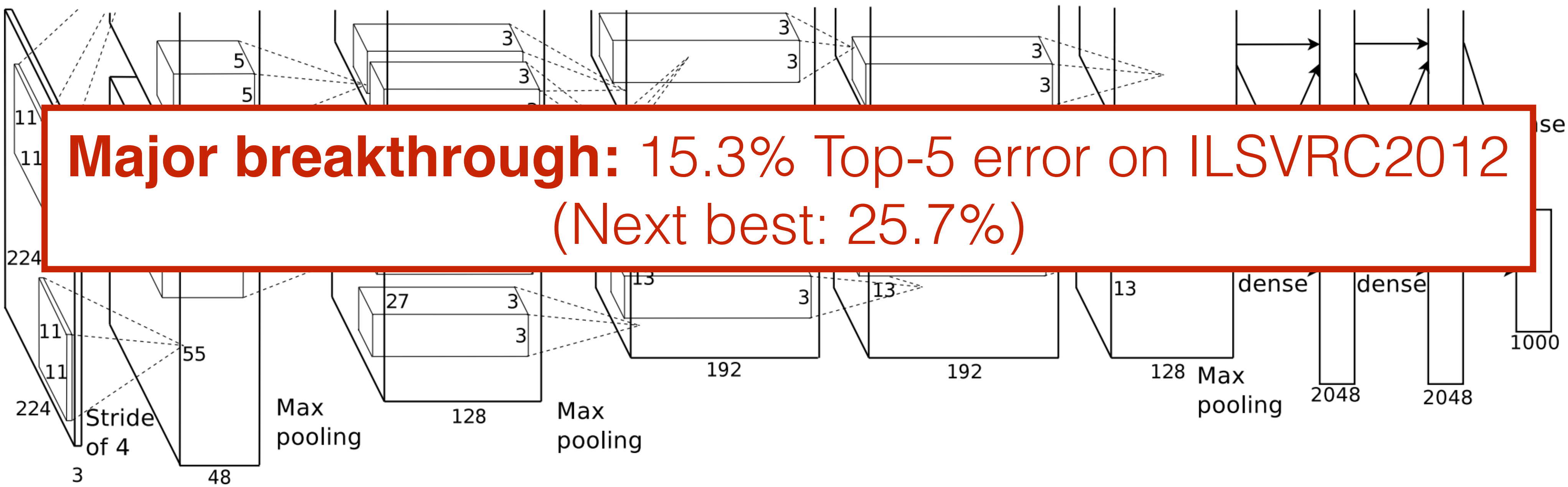
CNNs were *not* invented overnight



LeNet: a classifier for handwritten digits. [LeCun 1989]

CNNs in 2012: “SuperVision” (aka “AlexNet”)

“AlexNet” — Won the ILSVRC2012 Challenge



Major breakthrough: 15.3% Top-5 error on ILSVRC2012 (Next best: 25.7%)

Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network’s input is 150,528-dimensional, and the number of neurons in the network’s remaining layers is given by 253,440–186,624–64,896–64,896–43,264–4096–4096–1000.

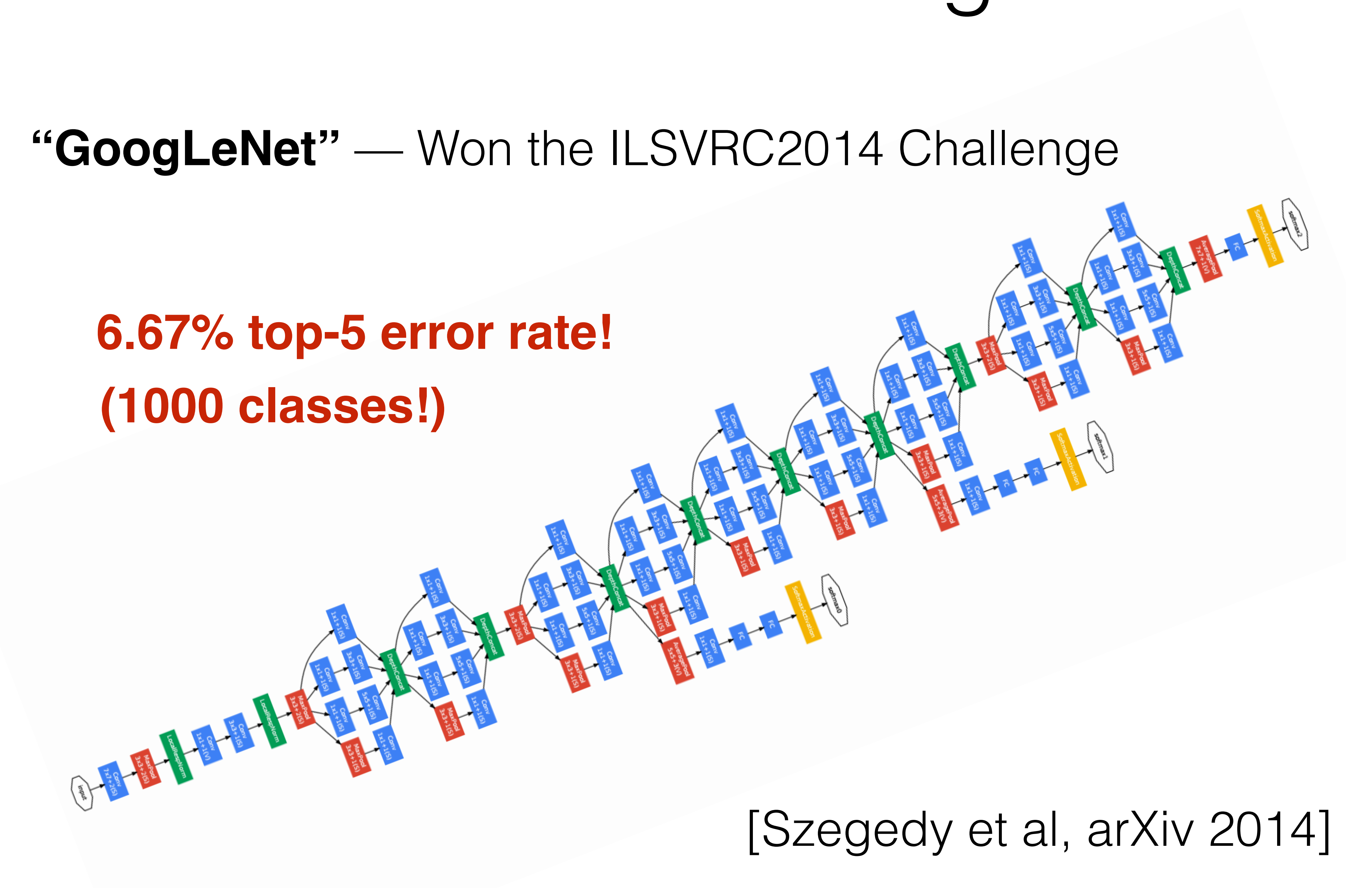
[Krizhevsky, Sutskever, Hinton. NIPS 2012]

CNNs in 2014: “GoogLeNet”

“GoogLeNet” — Won the ILSVRC2014 Challenge

6.67% top-5 error rate!

(1000 classes!)



[Szegedy et al, arXiv 2014]

CNNs in 2014: "VGGNet"

"VGGNet" — Second Place in the ILSVRC2014 Challenge

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

No fancy picture, sorry

7.3% top-5 error rate

(and 1st place in the detection challenge)

[Simonyan et al, arXiv 2014]

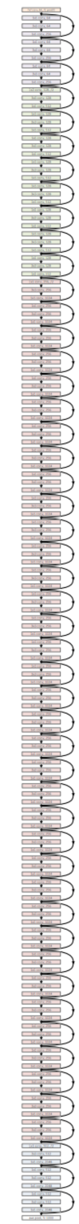
CNNs in 2015: “ResNet”

AlexNet, 8 layers
(ILSVRC 2012)

VGG, 19 layers
(ILSVRC 2014)

GoogLeNet, 22 layers
(ILSVRC 2014)

ResNet, 152 layers
(ILSVRC 2015)



Note: Despite its massive depth, ResNet has a lower runtime complexity than VGG

https://www.youtube.com/watch?v=1PGLj-uKT1w&feature=youtu.be&t=4m40s

[Kaiming et al, “Deep Residual Learning for Image Recognition” 2015]

CNNs in 2015: "ResNet"

- **1st places in all five main tracks**
  - ImageNet Classification: "*Ultra-deep*" (quote Yann) 152-layer nets
  - ImageNet Detection: 16% better than 2nd
  - ImageNet Localization: 27% better than 2nd
  - COCO Detection: 11% better than 2nd
  - COCO Segmentation: 12% better than 2nd

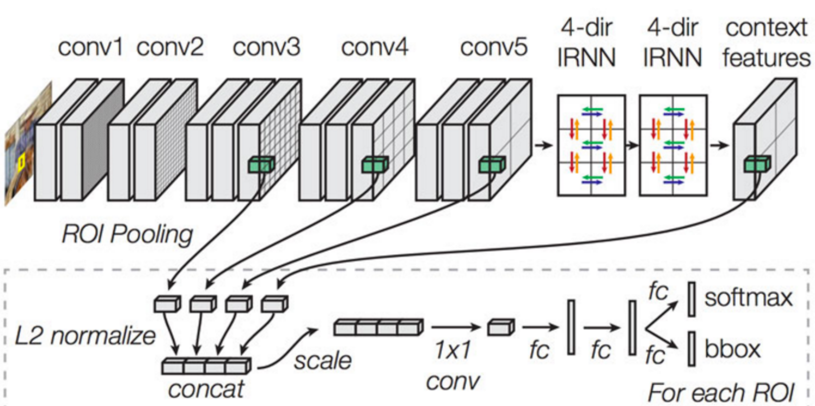
[Kaiming et al, "Deep Residual Learning for Image Recognition" 2015]

Aside: I placed in this year's competition!

Challenges Ranking

Team	BBox	Segmentation
MSRA	1st	1st
FAIRCNN	2nd	2nd
ION	3rd	-

"Best Student Entry", 3rd Overall



Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks

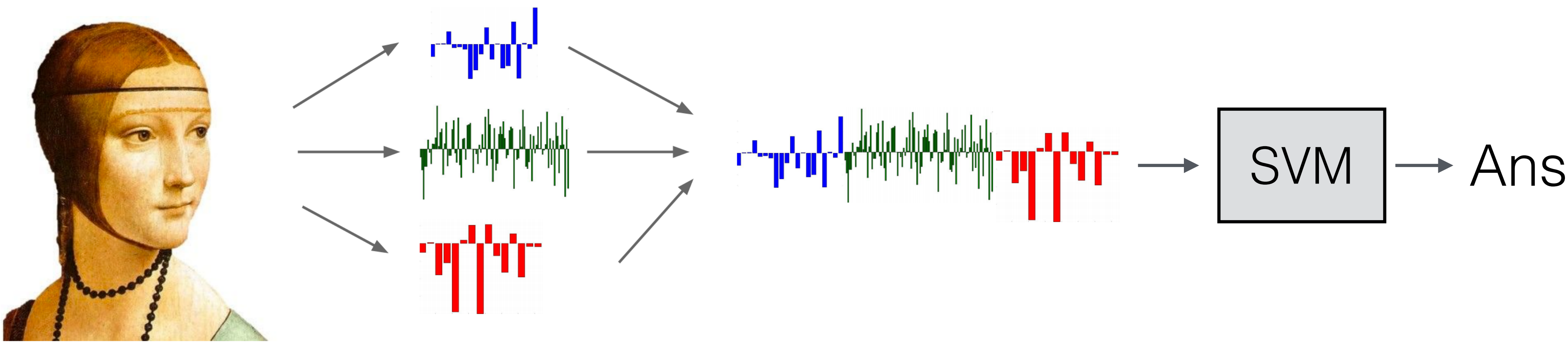
Sean Bell, C. Lawrence Zitnick, Kavita Bala, Ross Girshick

CVPR 2016

Best Student Entry on the 2015 MS COCO Challenge!

PDF » | Bibtex » | Slides (11M PDF) » | Video »

Aside: Before Deep Learning



Input Pixels

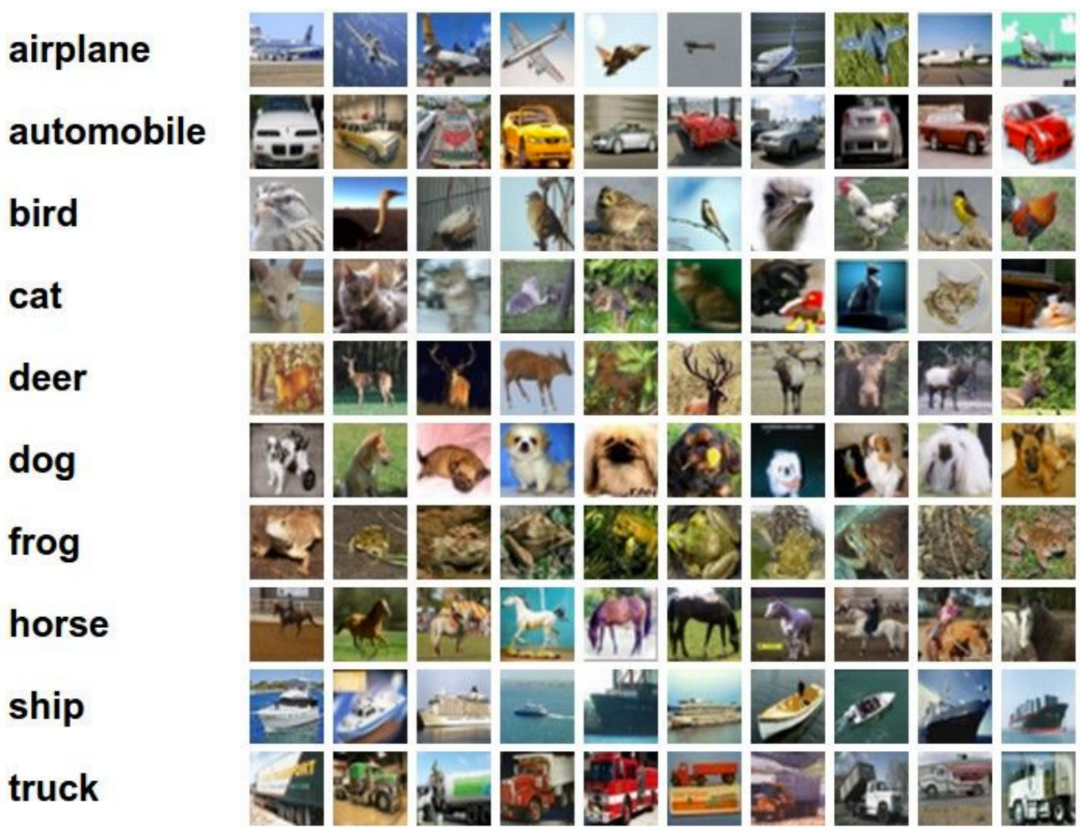
Extract Features

Concatenate into a vector ***x***

Linear Classifier

Figure: Karpathy 2016

Why use features?
Why not pixels?



$$f(x_i, W, b) = Wx_i + b$$

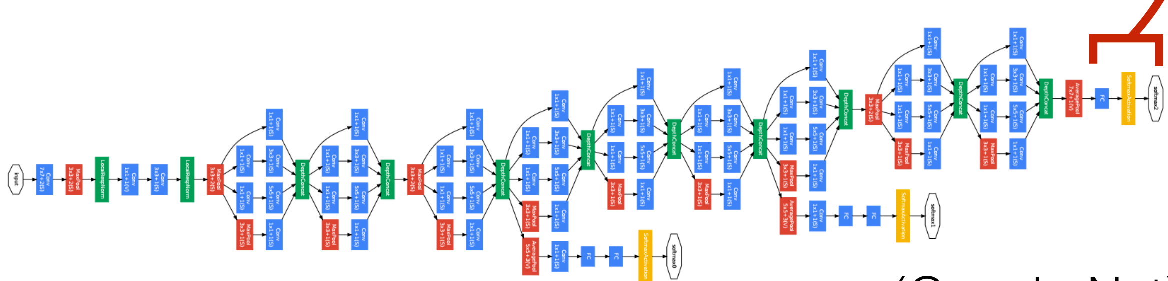
Q: What would be a very hard set of classes for a linear classifier to distinguish?

(assuming x = pixels)

Slide from Karpathy 2016

The last layer of (most) CNNs are linear classifiers

This piece is just a linear classifier



(GoogLeNet)

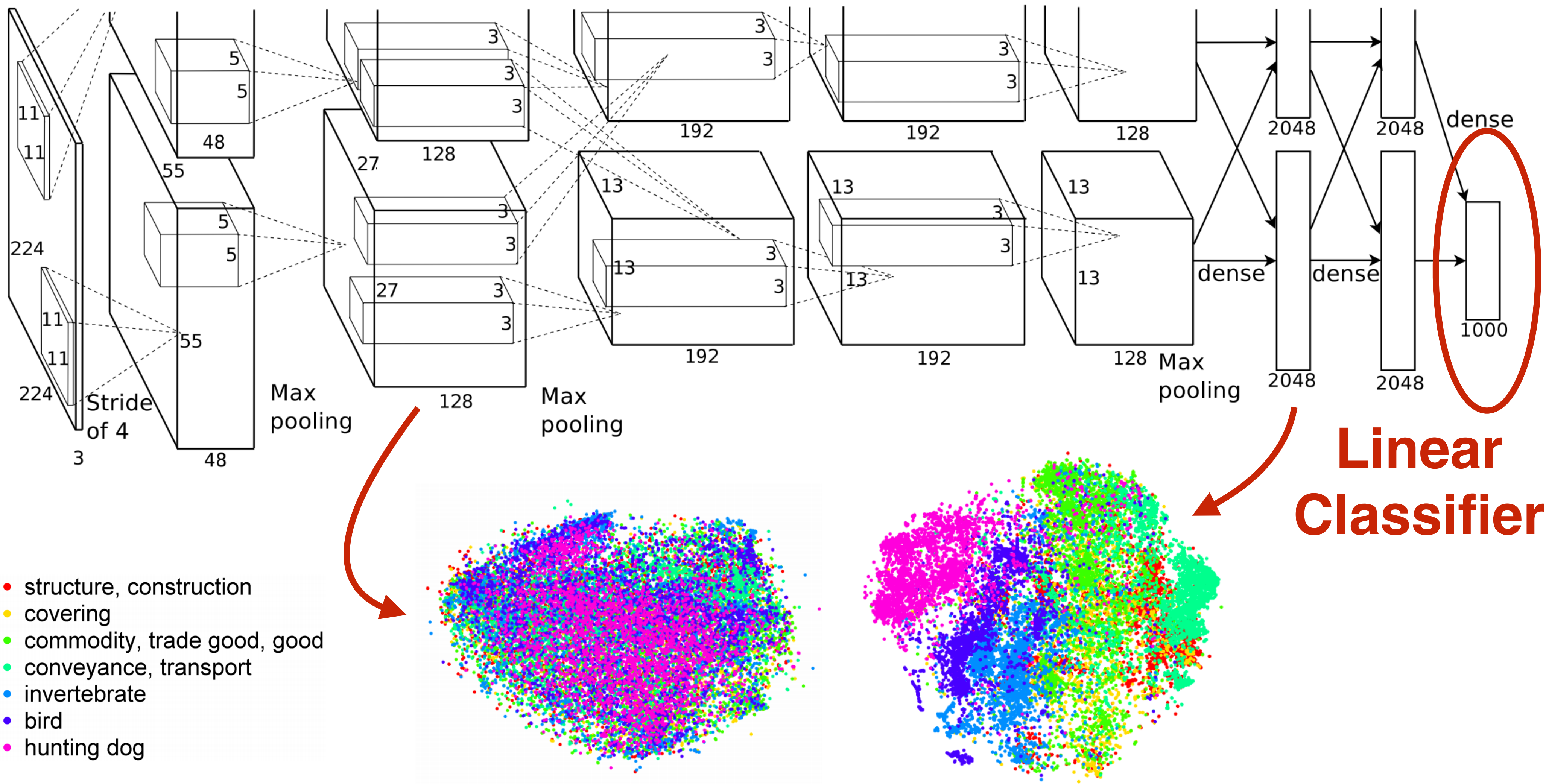
Ans

Input Pixels

Perform everything with a big neural network, trained end-to-end

Key: perform enough processing so that by the time you get to the end of the network, the classes are linearly separable

Example: Visualizing AlexNet in 2D with t-SNE



(c) DeCAF$_1$ (d) DeCAF$_6$

(2D visualization using t-SNE)

[Donahue, “DeCAF: DeCAF: A Deep Convolutional …”, arXiv 2013]

Recall: Linear Classifiers

- We have some dataset of (x,y)
- We have a **score function:** $s = f(x; W) \overset{\text{e.g.}}{=} Wx$
- We have a **loss function**:

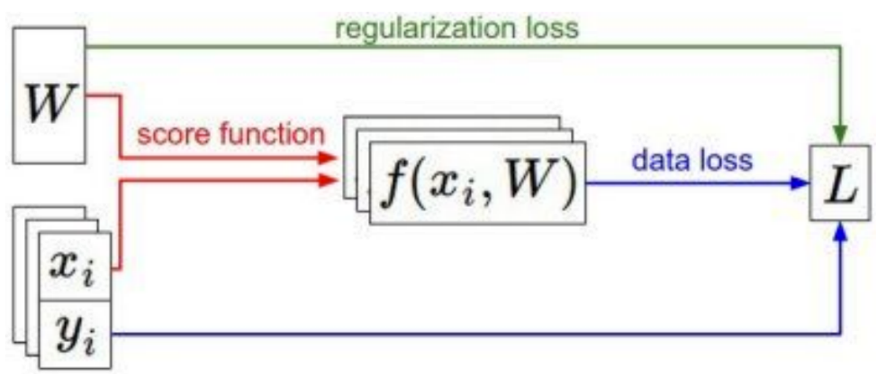
Softmax

$$L_i = -\log\left(\frac{e^{s_{y_i}}}{\sum_j e^{s_j}}\right)$$

SVM

$$L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$$

$$L = \frac{1}{N} \sum_{i=1}^{N} L_i + R(W)$$ Full loss



Slide from Karpathy 2016

Softmax Classifier

$\mathbf{s_{i,j}}$: Score that example **i** is is class **j**

$\mathbf{p_{i,j}}$: "Probability" that example **i** is class **j**

$$p_{i,j} = \frac{e^{s_{i,j}}}{\sum_k e^{s_{i,k}}}$$ (called the "Softmax" function)

$\mathbf{L_i}$: Loss for example **i:**

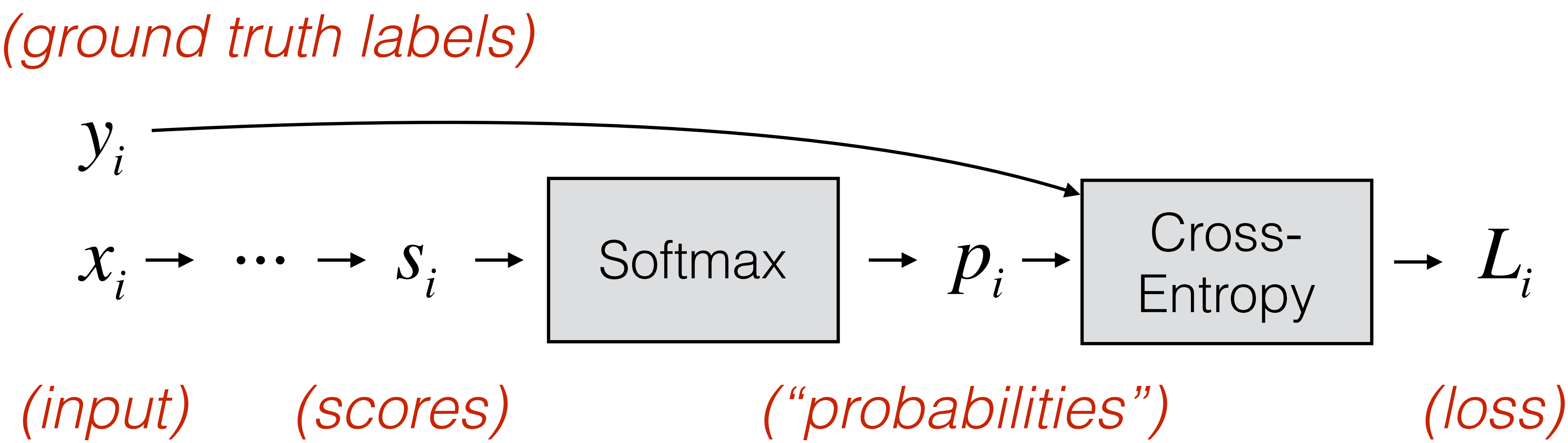
$$L_i = -\log p_{i,y_i}$$ (Cross-entropy)

$$L = \frac{1}{N}\sum_i L_i$$ (Average over examples)

Note: Softmax and cross-entropy loss used together is often called "Softmax loss"

Softmax Classifier

Block diagram for one example (x_i, y_i):



Softmax Classifier

$$p_{i,j} = \frac{e^{s_{i,j}}}{\sum_k e^{s_{i,k}}} \qquad L_i = -\log p_{i,y_i}$$

cat	**3.2**
car	5.1
frog	-1.7

unnormalized log probabilities

Q1: What is the min/max possible values for L?

Slide from Karpathy 2016

Softmax Classifier

$$p_{i,j} = \frac{e^{s_{i,j}}}{\sum_k e^{s_{i,k}}} \qquad L_i = -\log p_{i,y_i}$$

unnormalized probabilities

cat	**3.2**		**24.5**		**0.13**	→ L_i = -log(0.13) = **0.89**
car	5.1	exp →	164.0	normalize →	0.87	
frog	-1.7		0.18		0.00	

unnormalized log probabilities

probabilities

Q2: At initialization, W is small and thus s ~= 0.
What is the loss L?

Slide from Karpathy 2016

Softmax vs SVM Loss

Softmax:

$$L_i = -\log\left(\frac{e^{s_{y_i}}}{\sum_j e^{s_j}}\right)$$

SVM:

$$L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + 1)$$

assume scores:
[10, -2, 3]
[10, 9, 9]
[10, -100, -100]
and $\boxed{y_i = 0}$

Q: Suppose I take a datapoint and I jiggle a bit (changing its score slightly). What happens to the loss in both cases?

Slide from Karpathy 2016

Softmax Classifier

Let's code this up in NumPy:

$$p_{i,j} = \frac{e^{s_{i,j}}}{\sum_k e^{s_{i,k}}}$$

```
def softmax(s):
    exp_s = np.exp(s)
    probs = exp_s / np.sum(exp_s, axis=1, keepdims=True)
    return probs
```

Doesn't work — what's the problem?

- What if there is the value 1000 appears in "f"?

  Overflow —> we get inf/inf = NaN

- What if the largest value is -1000?

  Underflow —> we get 0/0 = NaN

This expression is numerically unstable

Softmax Classifier

Observation: subtracting a constant does not change “p”:

$$p_{i,j} = \frac{e^{s_{i,j}-C}}{\sum_k e^{s_{j,k}-C}} = \frac{e^{-C} e^{s_{i,j}}}{\sum_k e^{-C} e^{s_{i,k}}} = \frac{e^{s_{i,j}}}{\sum_k e^{s_{i,k}}}$$

If we choose “C” to be the max, then it works:

- If a large value appears in “f”, then that value will become 1 and all others will be 0 (avoiding overflow)
- If all values in “f” are large negative, then they will be shifted up towards 0 (avoiding underflow)