Case Study: ImageNet

Forest Agostinelli
University of South Carolina

Outline

- Background
- 2009-2011
- AlexNet
- GoogLeNet
- Residual Neural Networks
- Impact

ImageNet

- ImageNet is a dataset of millions of annotated images, each of which falls into one of 1,000 classes
  - The classes are primarily subsets of the following mammal, bird, fish, reptile, amphibian, vehicle, furniture, musical instrument, geological formation, tool, flower, fruit
- Introduced in 2009
- WordNet was used to decide on categories based on semantic hierarchies
  - Getting data from these categories helps them get diverse images for the 1,000 classes

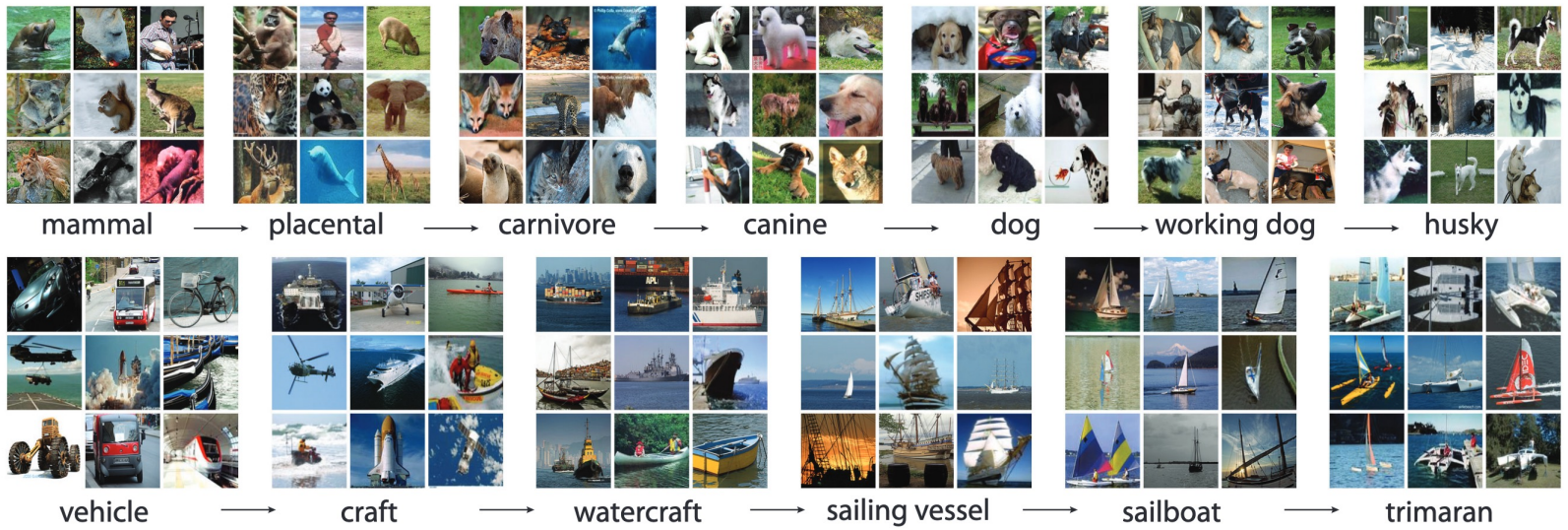


Figure 1: A snapshot of two root-to-leaf branches of ImageNet: the **top** row is from the mammal subtree; the **bottom** row is from the vehicle subtree. For each synset, 9 randomly sampled images are presented.

Fellbaum, Christiane. "WordNet: An electronic lexical database." *MIT Press google schola* 2 (1998): 678-686.

Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. IEEE, 2009.

ImageNet Images

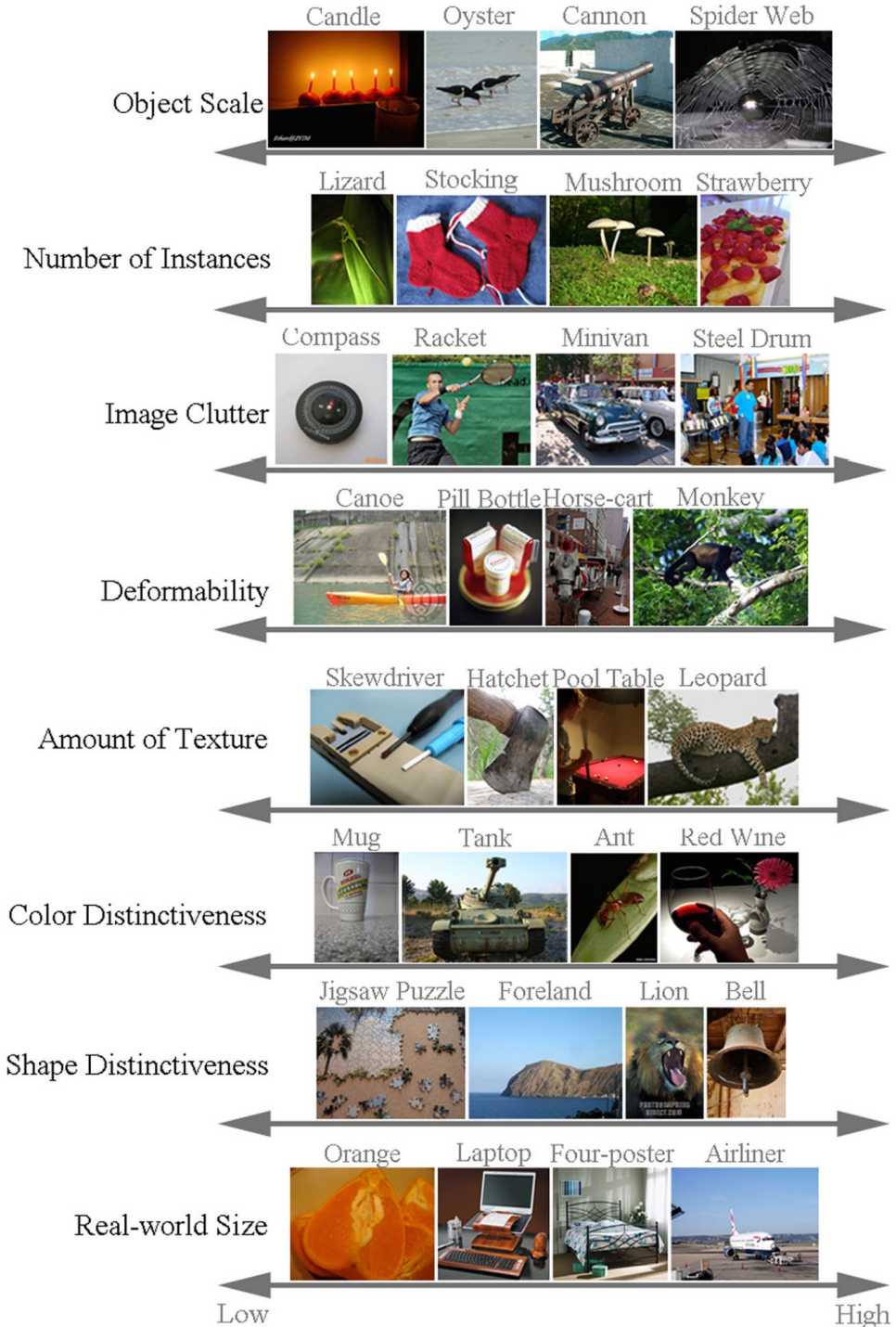


Fig. 1 The diversity of data in the ILSVRC image classification and single-object localization tasks. For each of the eight dimensions, we show example object categories along the range of that property. Object scale, number of instances and image clutter for each object category are computed using the metrics defined in Sect. 3.2.2 and in Appendix 1. The other properties were computed by asking human subjects to annotate each of the 1000 object categories (Russakovsky et al. 2013)

ImageNet Labeling

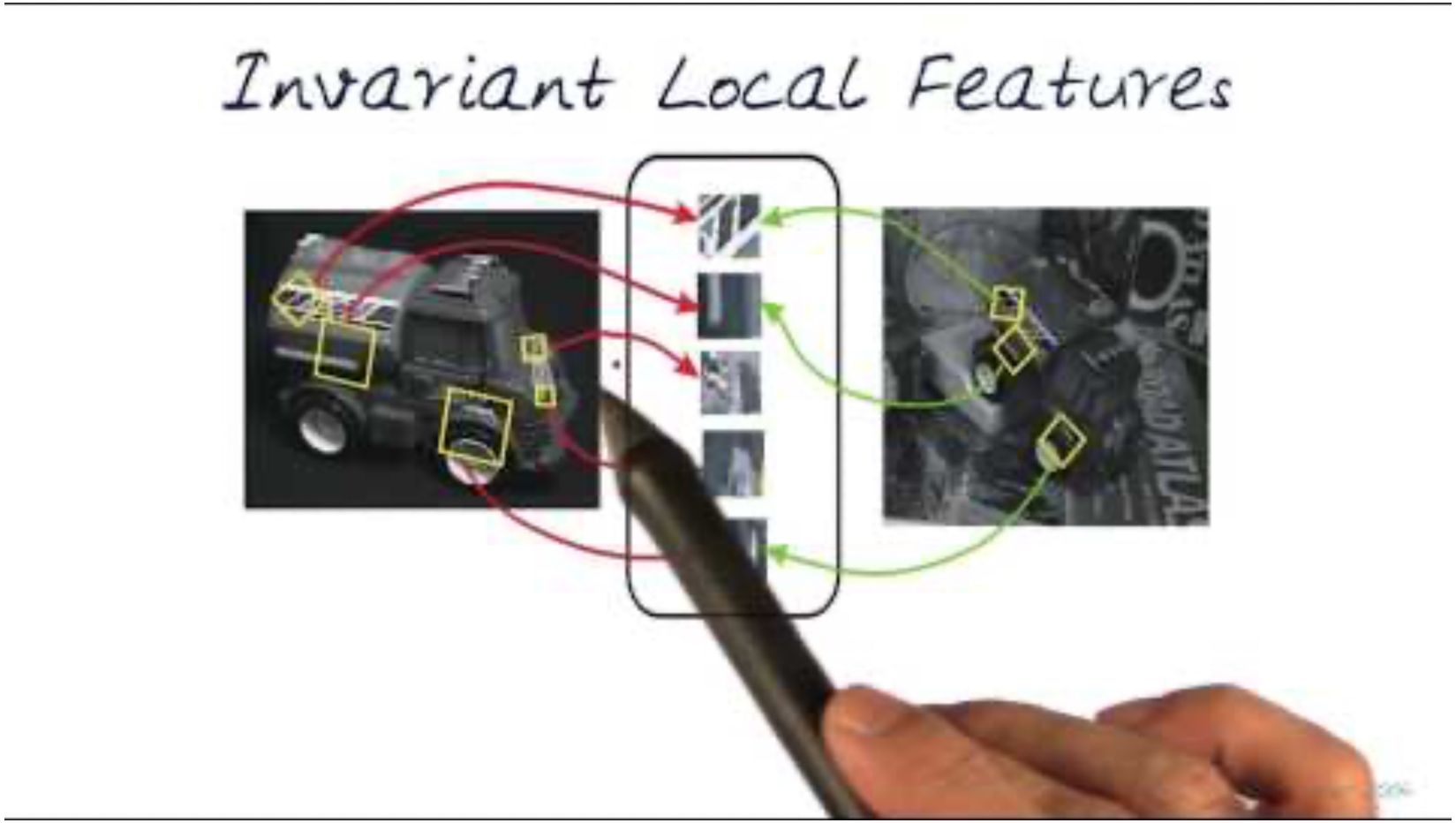
- Collecting candidate images
  - An image search is performed for each category
  - Image searches are performed on different search engines and in different languages
  - At the time, accuracy of image search results was around 10%
- Cleaning candidate images
  - Using Amazon Mechanical Turk, they sent the images to many different people to have them verify the label
  - Images for which a consensus is reached are integrated into the dataset

Outline

- Background
- 2009-2011
- AlexNet
- GoogLeNet
- Residual Neural Networks
- Impact

A First Attempt: SIFT Features

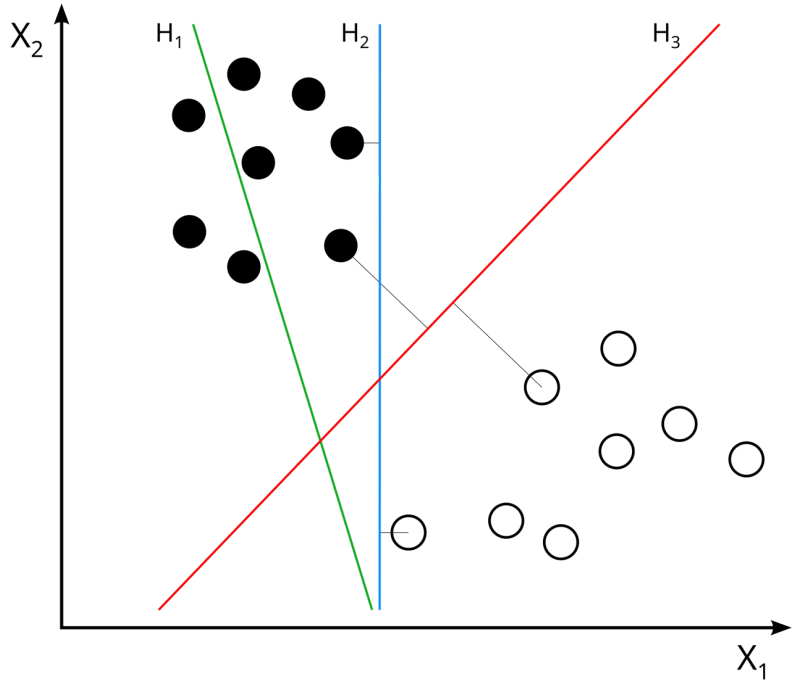
- Scale invariant feature transform (SIFT) features

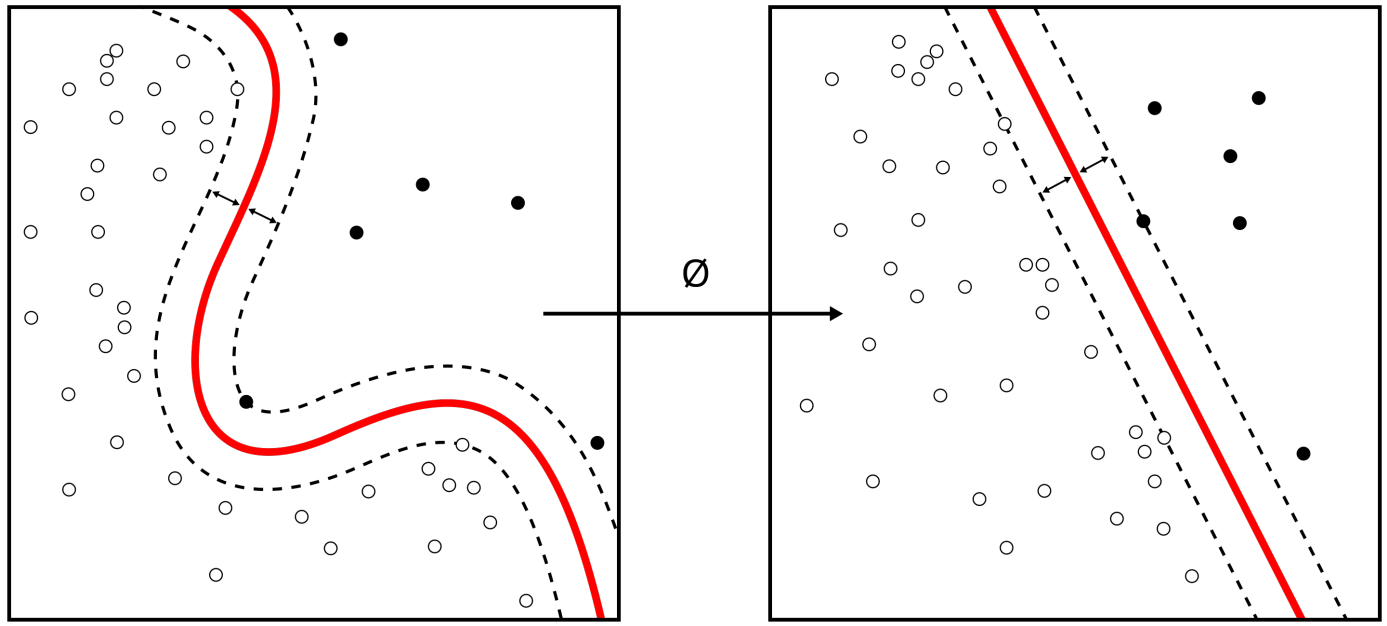


Lowe, David G. "Towards a computational model for object recognition in IT cortex." International Workshop on Biologically Motivated Computer Vision. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000.

A First Attempt: Support Vector Machines

- A support vector machine (SVM) attempts to find a line that maximizes the separations between classes
  - Linear classifier
- Can use a kernel to compare points in a different space to get a non-linear classifier





https://en.wikipedia.org/wiki/Support_vector_machine

A First Attempt

- Extract SIFT features from the image
- Processes them to create histograms of features
- Give those features as input to an SVM
- **Results:** Top-1 error rate of 80% (20% accuracy)

Deng, Jia, et al. "What does classifying more than 10,000 image categories tell us?." Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part V 11. Springer Berlin Heidelberg, 2010.

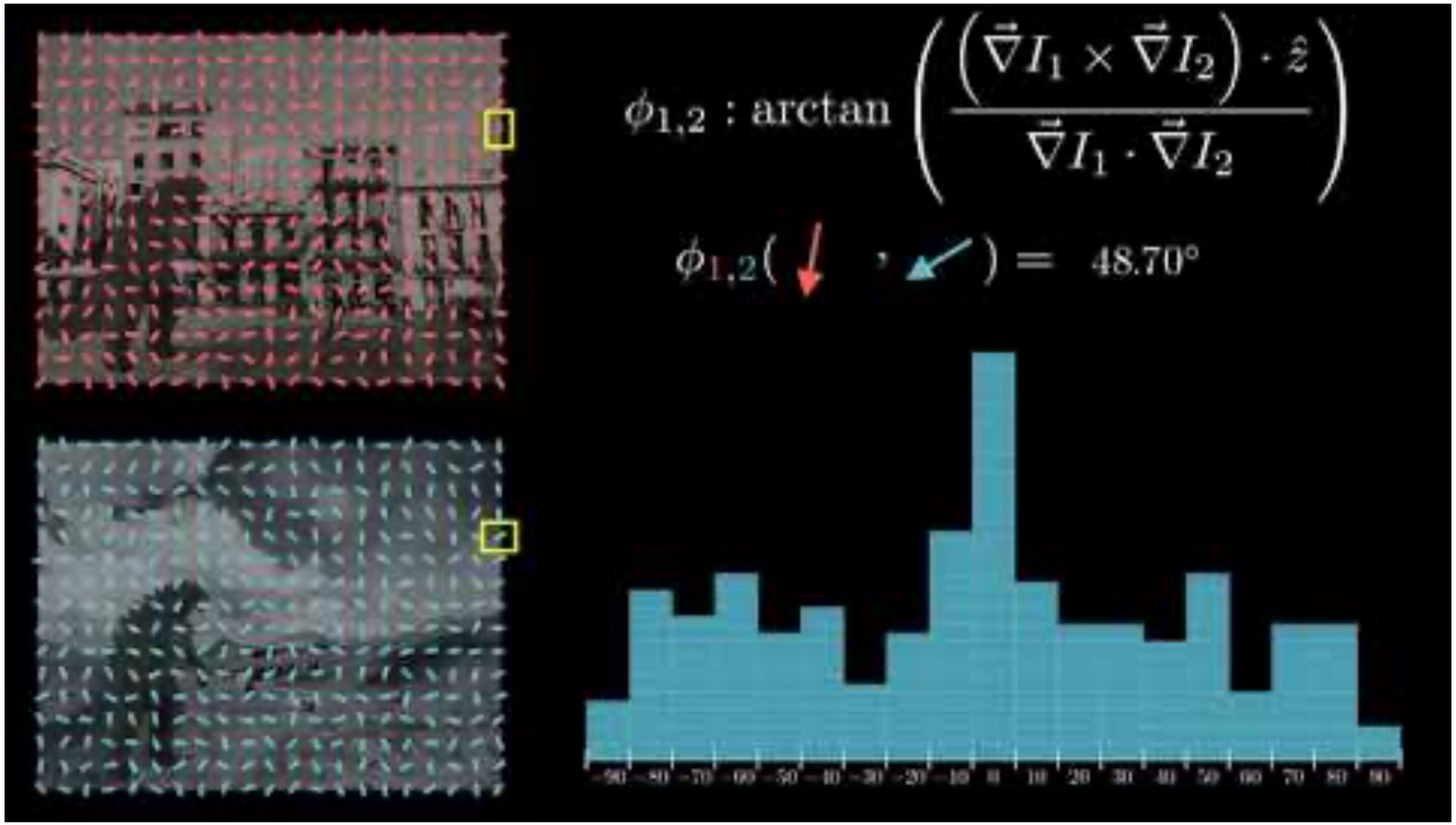
ILSVRC

- The ImageNet dataset has been used to create the ImageNet Large Scale Visual Recognition Challenge (ILSVRC)
- The ILSVRC was seen as a significantly challenging test for computer vision algorithms
- It led to many innovations in machine learning and deep learning, in particular
- The metric used is usually top-5 accuracy since some images that contain more than one class
  - If the top 5 classes in the model's output matches the true class, it is considered correct

https://www.image-net.org/challenges/LSVRC/index.php

ILSVRC 2010

- Combined histogram of oriented gradient (HOG) features and local binary pattern (LBP) features



Lin, Yuanqing, et al. "Large-scale image classification: Fast feature extraction and SVM training." CVPR 2011. IEEE, 2011.

ILSVRC 2010

- Note the manual hierarchy of features
- Results
  - 47.1% top-1 error rate
  - 28.2% top-5 error rate

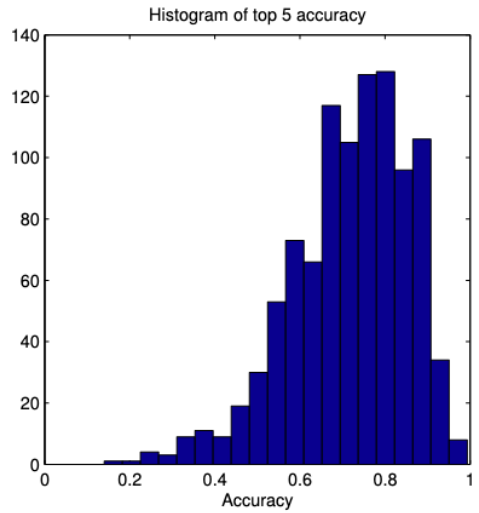
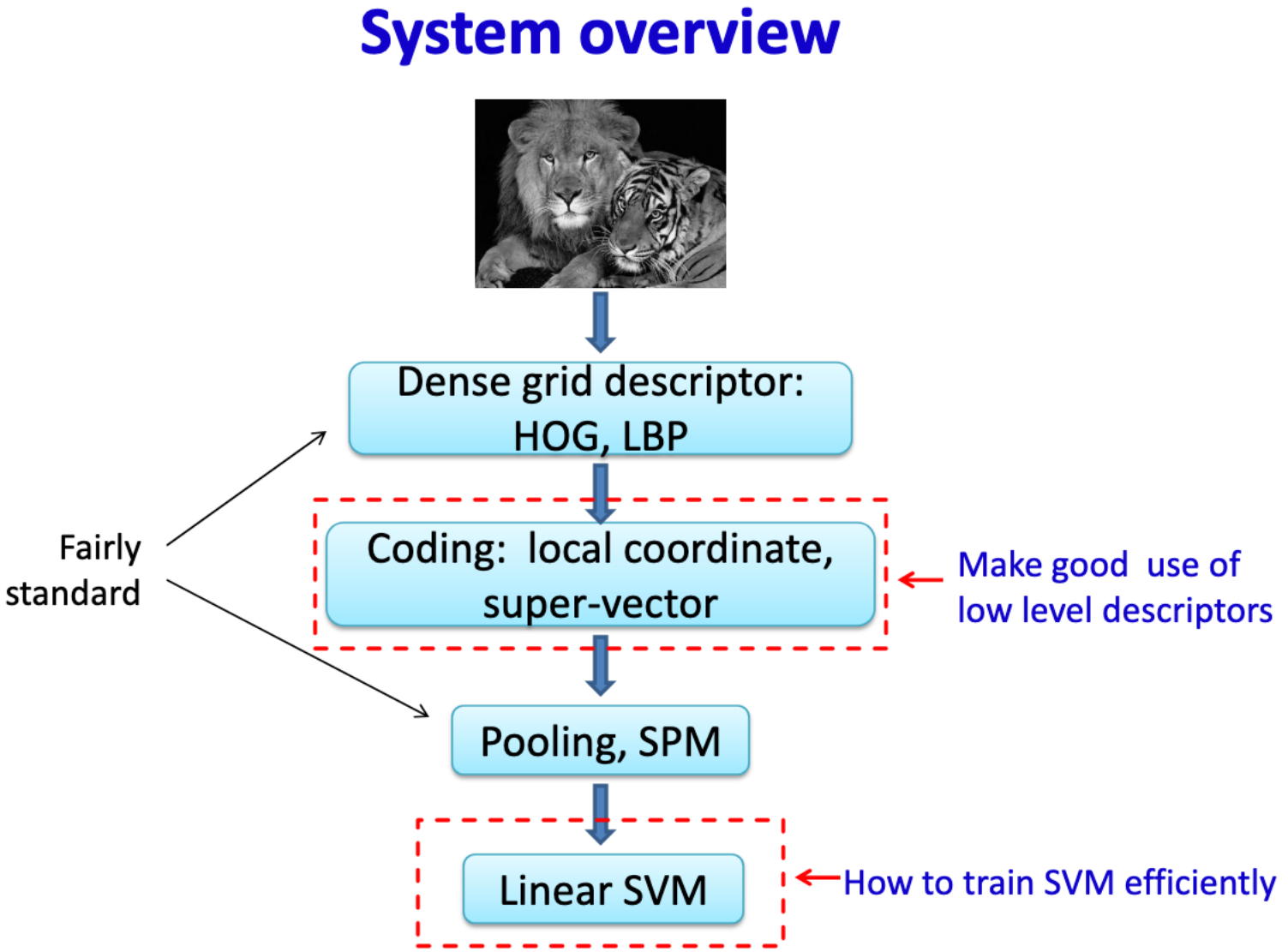


Figure 4. The histogram of the top 5 hit rate of the 1000 classes in ImageNet dataset.



ILSVRC 2011

- Uses a Fischer vector to encode images which were then given to an SVM for classification
- Two lessons learned
  - The larger the feature vectors, the better the classification accuracy
  - The more training data the better the classification accuracy
- Results
  - 45.7% top-1 error rate
  - 25.7% top-5 error rate

Outline

- Background
- 2009-2011
- AlexNet
- GoogLeNet
- Residual Neural Networks
- Impact

AlexNet

- Instead of a more manual approach of designing feature extractors and processing the features they extract, deep convolutional neural networks (CNNs) were used
- CNNs automated much of the process of feature design and processing through gradient descent
- The manual component was limited to the design of the CNN architecture which, as future research will show, can often be easily reused for new tasks
  - Furthermore, a CNN trained on ImageNet can be used as a quick feature extractor with little to no retraining of the CNN
- Accomplishing this required several improvements to CNN training: better non-linearities, parallelization of training, data augmentation, dropout

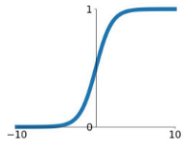
Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012).

AlexNet: Rectified Non-linearities

- Switching from tanh to ReLU non-linearities shows a significant improvement in performance

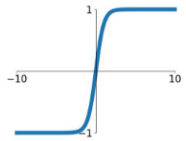
Sigmoid

$\sigma(x) = \frac{1}{1+e^{-x}}$



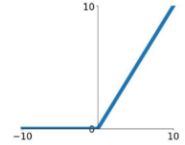
tanh

$\tanh(x)$



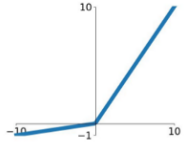
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

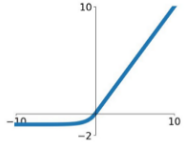


Maxout

$\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



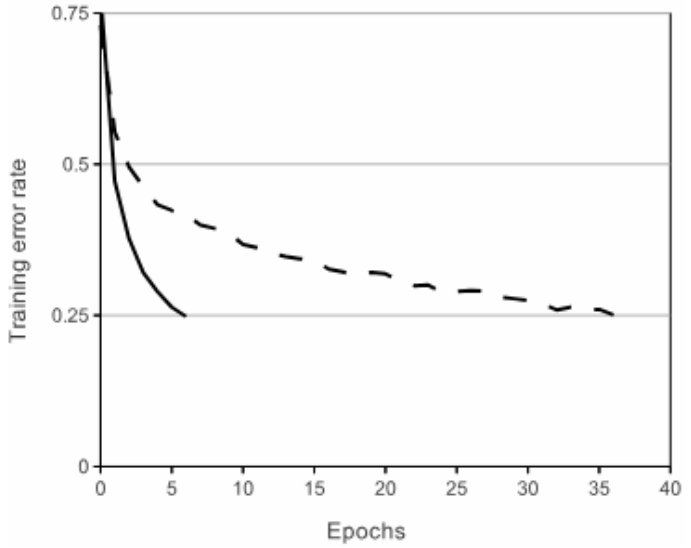


Figure 1: A four-layer convolutional neural network with ReLUs **(solid line)** reaches a 25% training error rate on CIFAR-10 six times faster than an equivalent network with tanh neurons **(dashed line)**. The learning rates for each network were chosen independently to make training as fast as possible. No regularization of any kind was employed. The magnitude of the effect demonstrated here varies with network architecture, but networks with ReLUs consistently learn several times faster than equivalents with saturating neurons.

AlexNet: Parallelization of Training

- Input is of size 224x224x3 (150,528 total features)
- To speed up training, the network is split up amongst multiple GPUs

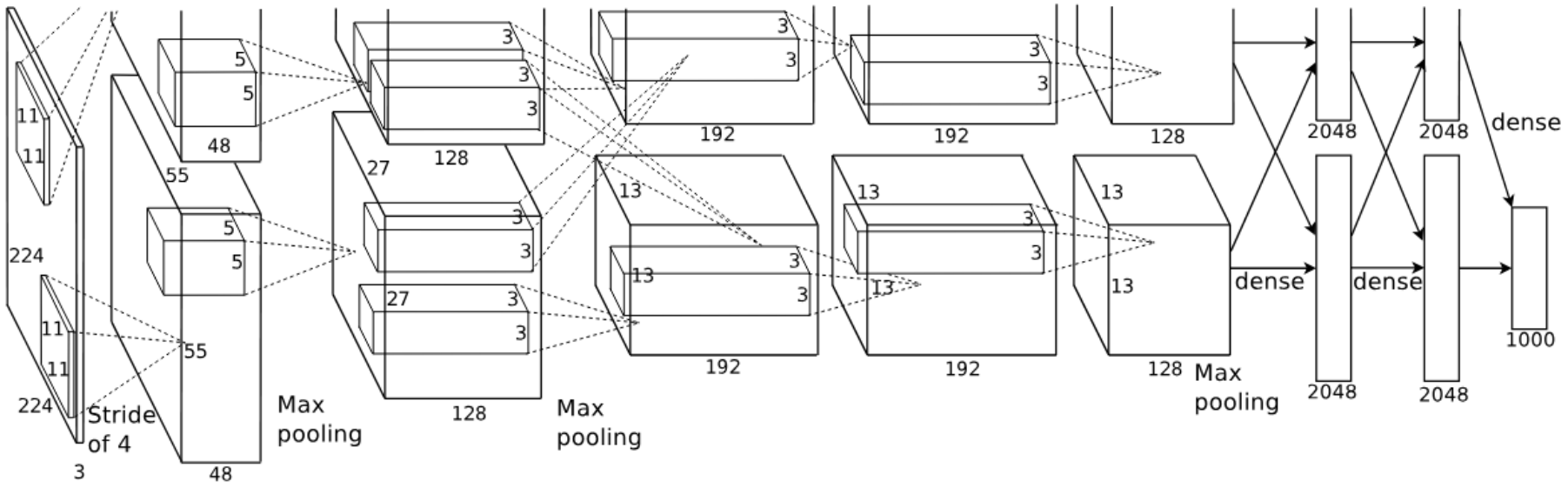


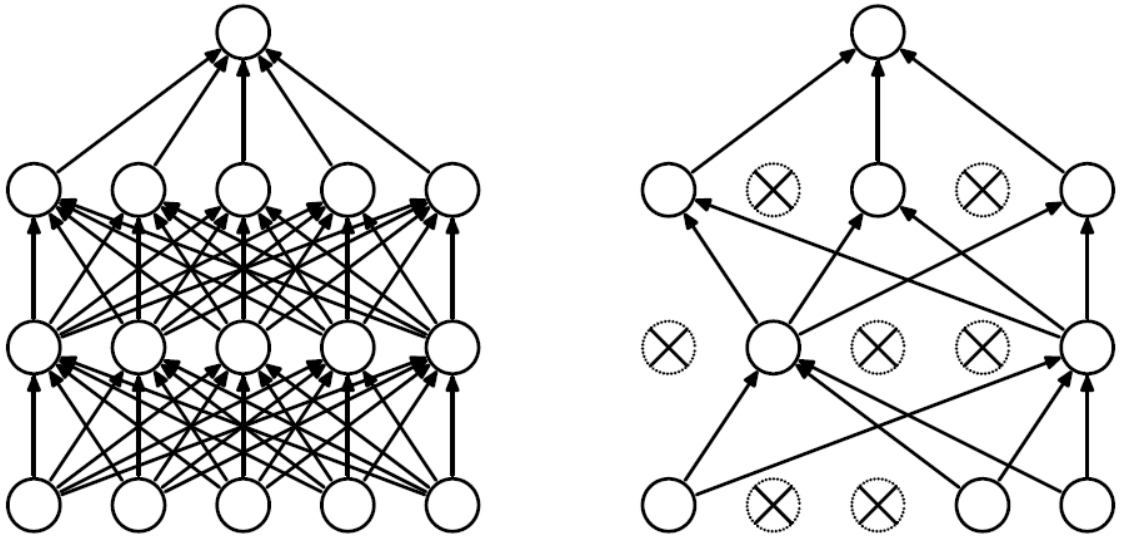
Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network's input 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.

AlexNet: Data Augmentation

- Previous ImageNet competitions showed the benefit of a plethora of data
- Adding random translation and reflections
  - Increases the dataset by a factor of 2048
  - At test time, the average CNN output of 5 patches and their reflections (10 patches total) are obtained to give the overall output
  - Without this data augmentation, overfitting occurs and they have to use a smaller CNN to combat it
- Altering RGB intensity
  - Decreases top-1 error rate by over 1%

AlexNet: Dropout

- To further reduce overfitting, dropout was used in the first two fully connected layers with a rate of 0.5
- Significantly reduced overfitting, but also doubles the number of iterations required to achieve good performance



(a) Standard Neural Net

(b) After applying dropout.

AlexNet: Results

- Results
  - 37.5% top-1 error rate
  - 17.0% top-5 error rate
- This was a significant improvement in performance compared to previous methods

Model	Top-1	Top-5
Sparse coding [2]	*47.1%*	*28.2%*
SIFT + FVs [24]	*45.7%*	*25.7%*
CNN	**37.5%**	**17.0%**

Table 1: Comparison of results on ILSVRC-2010 test set. In *italics* are best results achieved by others.

Outline

- Background
- 2009-2011
- AlexNet
- GoogLeNet
- Residual Neural Networks
- Impact

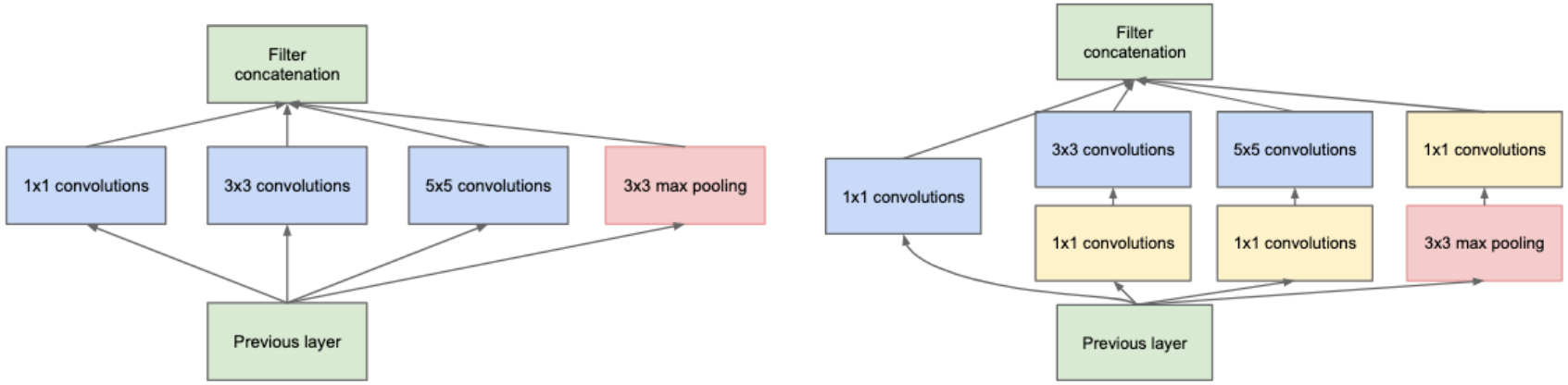
GoogLeNet: Motivation

- Inspired by the LeNet architecture for MNIST
- While performance can be improved by increases network size, this also risks overfitting
- How can we use fewer parameters while improving performance

Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

GoogLeNet: Inception Architecture

- Use feature detectors as multiple resolutions in parallel
- Use 1x1 convolutions to reduce number of feature maps to reduce number of parameters



(a) Inception module, naïve version (b) Inception module with dimension reductions

Figure 2: Inception module

Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

GoogLeNet

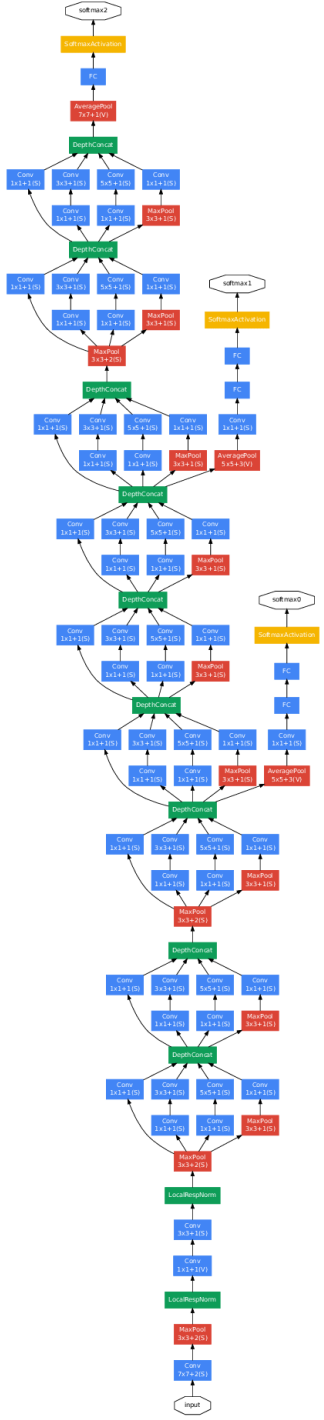
- Adds a loss to intermediate layers in addition to the output layer
  - Helps with gradient propagation
- 12x fewer parameters than AlexNet
- Averages over several crops and several models

Team	Year	Place	Error (top-5)	Uses external data
SuperVision	2012	1st	16.4%	no
SuperVision	2012	1st	15.3%	Imagenet 22k
Clarifai	2013	1st	11.7%	no
Clarifai	2013	1st	11.2%	Imagenet 22k
MSRA	2014	3rd	7.35%	no
VGG	2014	2nd	7.32%	no
GoogLeNet	2014	1st	6.67%	no

Table 2: Classification performance

Number of models	Number of Crops	Cost	Top-5 error	compared to base
1	1	1	10.07%	base
1	10	10	9.15%	-0.92%
1	144	144	7.89%	-2.18%
7	1	7	8.09%	-1.98%
7	10	70	7.62%	-2.45%
7	144	1008	6.67%	-3.45%

Table 3: GoogLeNet classification performance break down

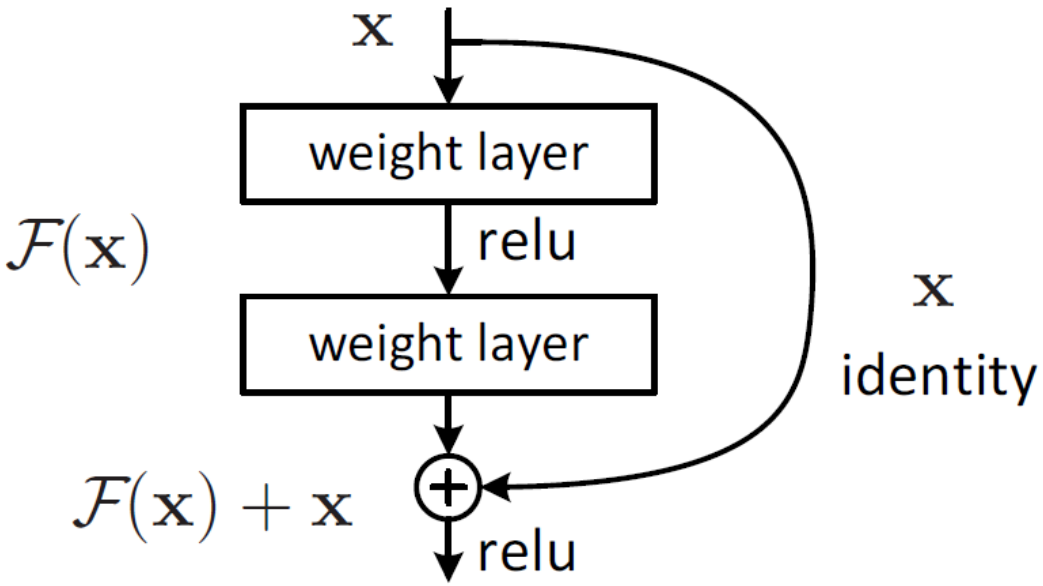


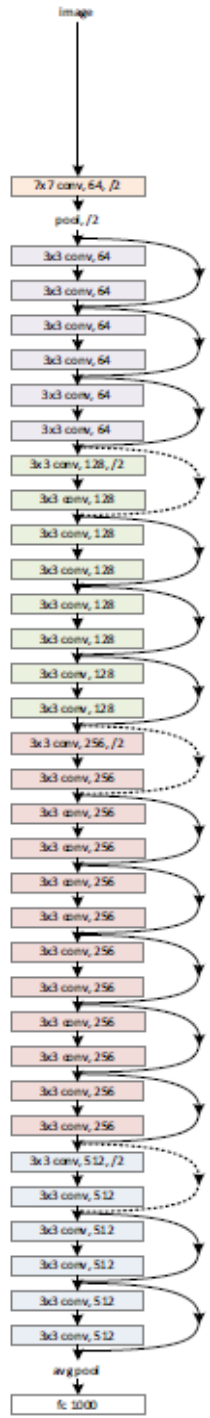
Outline

- Background
- 2009-2011
- AlexNet
- GoogLeNet
- Residual Neural Networks
- Impact

Residual Neural Networks

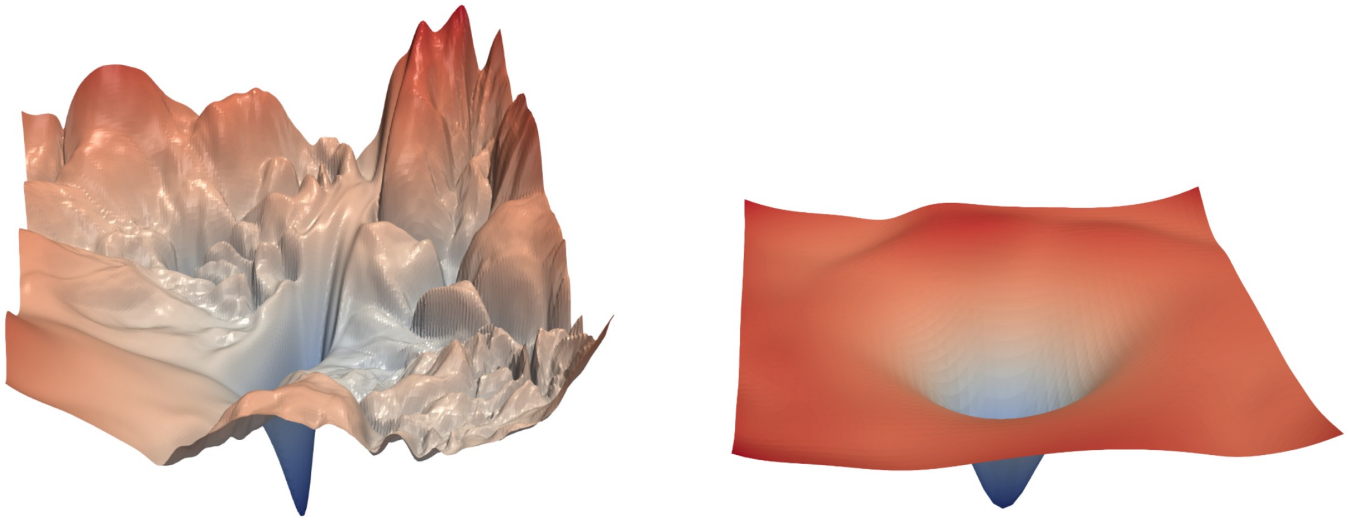
- Training can become more difficult as the number of layers increases
- Adding skip connections allows us to train networks with hundreds of layers
- The intuition is, if a residual block is not useful, its weights can be pushed to zero and it will have no effect





Residual Neural Networks: Loss Surface Interpretation

- When visualizing the loss landscape, we see that residual neural networks have a much smoother loss landscape
- This could make optimization a lot easier



(a) without skip connections

(b) with skip connections

Loss surface

Residual Neural Networks

- It was shown that deeper residual neural networks generally improved performance

method	top-1 err.	top-5 err.
VGG [40] (ILSVRC'14)	-	8.43†
GoogLeNet [43] (ILSVRC'14)	-	7.89
VGG [40] (v5)	24.4	7.1
PReLU-net [12]	21.59	5.71
BN-inception [16]	21.99	5.81
ResNet-34 B	21.84	5.71
ResNet-34 C	21.53	5.60
ResNet-50	20.74	5.25
ResNet-101	19.87	4.60
ResNet-152	**19.38**	**4.49**

Table 4. Error rates (%) of **single-model** results on the ImageNet validation set (except † reported on the test set).

method	top-5 err. (**test**)
VGG [40] (ILSVRC'14)	7.32
GoogLeNet [43] (ILSVRC'14)	6.66
VGG [40] (v5)	6.8
PReLU-net [12]	4.94
BN-inception [16]	4.82
ResNet (ILSVRC'15)	**3.57**

Table 5. Error rates (%) of **ensembles**. The top-5 error is on the test set of ImageNet and reported by the test server.

Outline

- Background
- 2009-2011
- AlexNet
- GoogLeNet
- Residual Neural Networks
- Impact

Recap

- HOG and LBP features with SVMs (2010)
  - 47.1% top-1 error rate
  - 28.2% top-5 error rate
- Fischer features with SVMs (2011)
  - 45.7% top-1 error rate
  - 25.7% top-5 error rate
- AlexNet (2012)
  - 37.5% top-1 error rate
  - 17.0% top-5 error rate
- GoogLeNet (2014)
  - 6.66% top-5 error rate
- Residual Neural Networks (2015)
  - 3.57% top-5 error rate

Foundation Models

- AlexNet, GoogleNet, and ResNets have been used as foundation models for computer vision
- These architectures were first trained on ImageNet or an even larger corpus of data
- They were then used on other downstream tasks by practitioners that may not have had the time or the resources to use them, otherwise

Computer Vision Foundation Models Uses

- Image classification can be done for a completely different task by
  - Extracting activations of intermediate layers
  - Using them to train a simple linear model
- Image classification can also be done by
  - Finetuning the model for a limited number of training iterations with the new data
- Clustering can be done by
  - Extracting activations of intermediate layers
  - Using them with a simple clustering algorithm, such as PCA
- Many other computer vision tasks have used these foundation models
  - Object localization, image segmentation, image understanding, etc.

Reflection on Evaluation Metrics

- The ImageNet dataset had a significant impact on AI
- However, limiting our evaluation to top-1 or top-5 accuracy can blind us to more fundamental AI questions
  - Semantic understanding and description
  - Disambiguation
  - Human-in-the-loop image processing