Image classification and intro to neural networks

Vlad Shakhuro










8 October 2026

Outline

1. Image classification task and datasets
2. Linear classification and MLPs
3. Convolutional neural networks
4. Milestone: AlexNet

Binary classification



Does this image contain a pedestrian?

Binary answer $y \in \{\underbrace{0}_{\text{no}}, \underbrace{1}_{\text{yes}}\}$

Alternatively, the estimated probability of the positive answer $p_{\text{yes}} \in [0, 1]$

Multiclass classification



Which object is in this image?

The set of *allowed* classes
is fixed in advance

Integer answer $y \in \{\underbrace{1}_{\text{car}}, \underbrace{2}_{\text{sign}}, \ldots, \underbrace{K}_{\text{bike}}\}$

Alternatively, estimated class probabilities:

$$p_k \in [0,1] \quad k = 1, \ldots, K \quad \sum_{k=1}^{K} p_k = 1$$

Multi-label classification and attributes



Which objects are in this image?

Binary vector answer $y \in \{0,1\}^K$:

$$y = \left(\underbrace{1}_{\text{person}}, \underbrace{1}_{\text{bike}}, \underbrace{1}_{\text{dog}}, \underbrace{1}_{\text{car}}, \ldots, \underbrace{0}_{\text{bird}} \right)$$

Alternatively, K independent $p_k \in [0,1]$, sum need not be 1

Metrics

N samples, $\hat{y}_n = \arg\max_k p_k(x_n)$

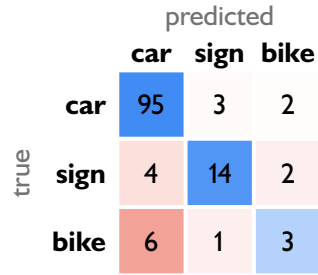
Accuracy: share of correct predictions

$$\text{acc} = \frac{1}{N} \sum_n \mathbb{1}[\hat{y}_n = y_n]$$

Top-5 accuracy: $T_5(x_n)$ are 5 classes with largest $p_k(x_n)$

$$\text{acc}_5 = \frac{1}{N} \sum_n \mathbb{1}[y_n \in T_5(x_n)]$$

More stable for images that may have several valid labels

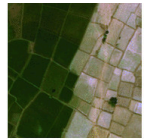


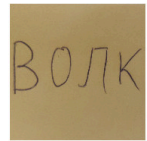
Confusion matrix for imbalanced classes

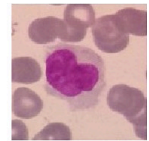
Always "car": $\text{acc} = 77\%$

Data domains and modalities









Every computer vision algorithm is designed to operate on images sampled from some *statistical population*. This population is described by an unknown probability distribution over the set of all "valid" (for that algorithm) images:

$$x \sim P(\mathcal{X}) \qquad \mathcal{X} \subseteq \mathbb{R}^{C \times H \times W}$$

These algorithms work by exploiting the inherent properties and invariants of the *statistical population* they support

CIFAR-10 and CIFAR-100



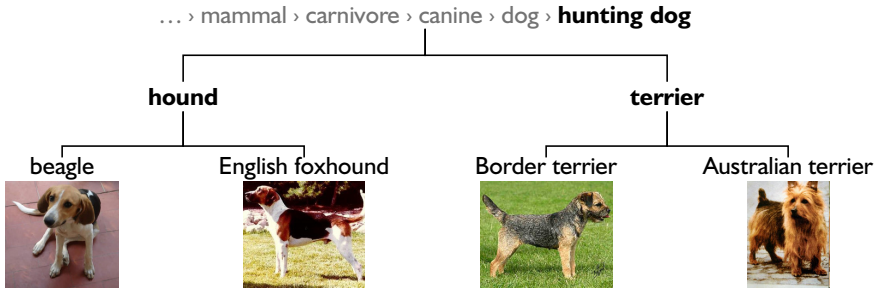
Labelled subsets of 80M Tiny Images

- 32 × 32 color images
- 50k train + 10k test images
- 10 and 100 classes

Krizhevsky. Learning Multiple Layers of Features from Tiny Images. U. Toronto tech. report 2009

ImageNet

- Classes are WordNet *synsets*: sets of synonyms naming one concept
- Goal: 500–1000 photos for each of 80k noun synsets
- Photos from image search, verified by crowd workers
- 14.2M images in 21.8k synsets, 1M of them with bounding boxes

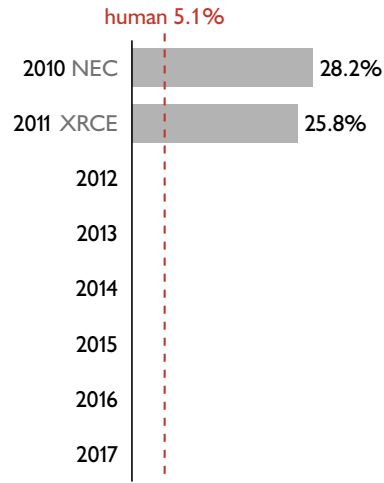


Deng et al. ImageNet: A large-scale hierarchical image database. CVPR 2009

ImageNet-1k (ILSVRC)

Subset for the yearly ImageNet challenge, 2010–2017

- 1000 classes, 120 dog breeds
- 1.28M train, 50k val, 100k test images
- Ranked by top-5 error



Russakovsky et al. ImageNet Large Scale Visual Recognition Challenge. IJCV 2015

ImageNet: one label per image

NN top-5 predictions: photos hold several objects, classes overlap



grille

convertible
grille
pickup
beach wagon
fire engine

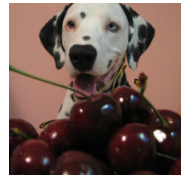
the whole car is in view



mushroom

agaric
mushroom
jelly fungus
gill fungus
dead-man's-fingers

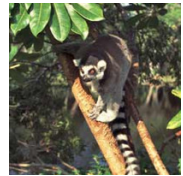
an agaric is a mushroom



cherry

dalmatian
grape
elderberry
Staffordshire bullterrier
currant

the dog takes up the frame



Madagascar cat

squirrel monkey
spider monkey
titi
indri
howler monkey

WordNet name of a lemur

Labels are noisy too: at least 6% of validation labels are wrong

Krizhevsky et al. ImageNet Classification with Deep Convolutional Neural Networks. NIPS 2012

Open Images



+ Person
+ Animal
+ Kangaroo
+ Plant
− Dog
− Deer
− Tree
− Carnivore

Largest *open* dataset:
9M Flickr photos, CC BY 2.0

- Multi-label: 61M image-level labels, 20k classes
- Labels are *verified*: present (+) or absent (−)
- Unverified classes are unknown, not negative
- Also boxes, masks and narratives

Kuznetsova et al. The Open Images Dataset V4: Unified Image Classification, Object Detection, and Visual Relationship Detection at Scale. IJCV 2020

Fine-grained classification

Different species, look alike



American crow

≠



Fish crow

small inter-class difference

Same species, look different



Painted bunting ♂

=



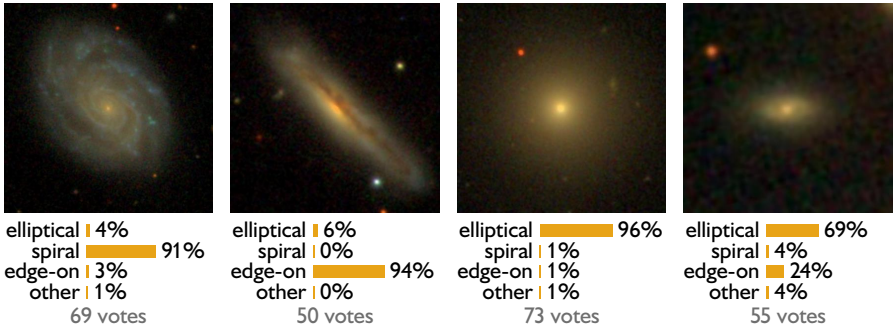
Painted bunting ♀

large intra-class variation

- Classes are subcategories:
  bird species, car models, aircraft
- Labels need experts:
  4.4% of CUB labels are wrong

Dataset	Classes	Images
CUB-200-2011	200	11,788
Stanford Cars	196	16,185
FGVC-Aircraft	100	10,000
iNaturalist 2017	5,089	859,000

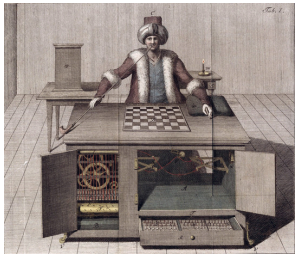
Galaxy Zoo



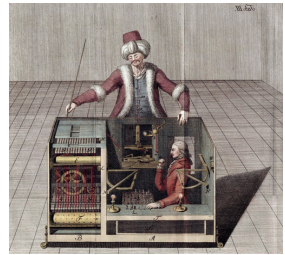
- 2007: 150K volunteers, 50M+ votes in a year, **for free**
- Label of a galaxy is a *distribution of votes*
- Galaxy Zoo 2: a hard question is split into a tree of simple ones

Lintott et al. Galaxy Zoo: Morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey. MNRAS 2008

Mechanical Turk (1770)



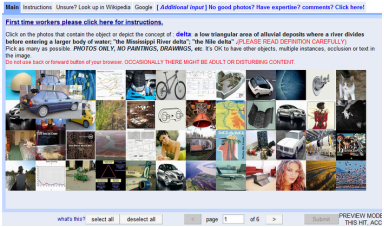
what the audience saw



how it might work (Racknitz, 1789)

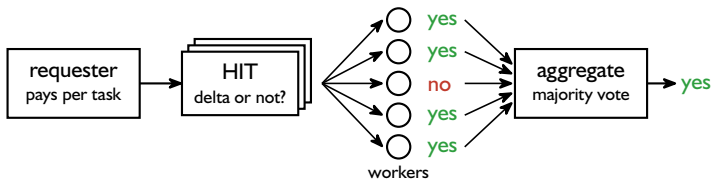
- Chess automaton by Wolfgang von Kempelen for Maria Theresa
- Toured for 84 years, played Napoleon and Benjamin Franklin
- A **hidden chess master** moved the pieces; revealed in 1857

Amazon Mechanical Turk (2005–2026)



ImageNet task:
pick the photos of a river delta

- "Artificial artificial intelligence", humans behind an API
- ImageNet: 49K workers, 167 countries, 2008–2010
- Closed on 30 September 2026
- Important topics: label ambiguity, cheating, unclear instructions, efficiency



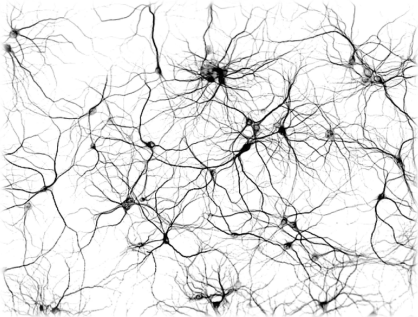
Label quality control

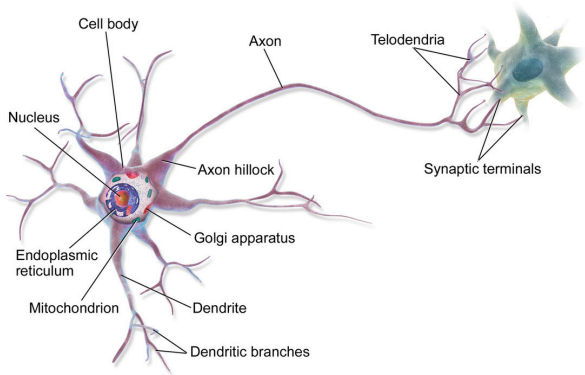
- Redundancy: several people label each image. Answers are merged by majority vote, weighted by annotator accuracy (Dawid–Skene) or kept as a vote distribution, as in Galaxy Zoo
- Gold questions: items with known answers catch careless annotators
- Clear instructions: “delta” may be river delta, an airline or a Greek letter
- Disagreement between annotators flags ambiguous items for experts
- Model in the loop: models pre-label, humans verify and fix

Outline

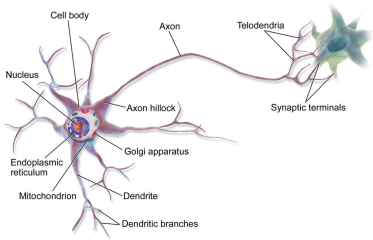
1. Image classification task and datasets
2. Linear classification and MLPs
3. Convolutional neural networks
4. Milestone: AlexNet

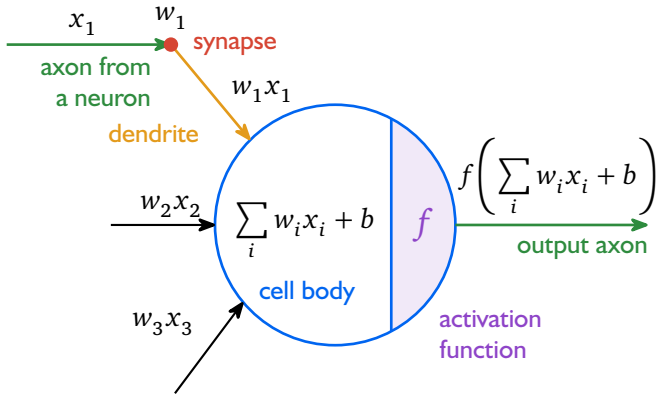
Biological neurons





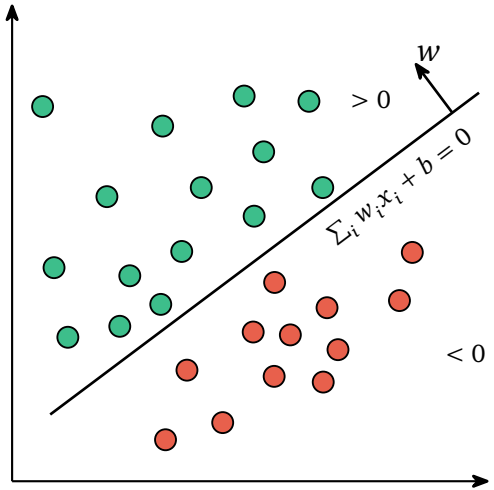
McCulloch-Pitts neuron model





$$a = f\left(\sum_{i=1}^{d} w_i x_i + b\right)$$

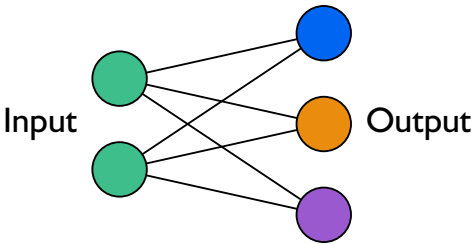
Neuron as linear classifier



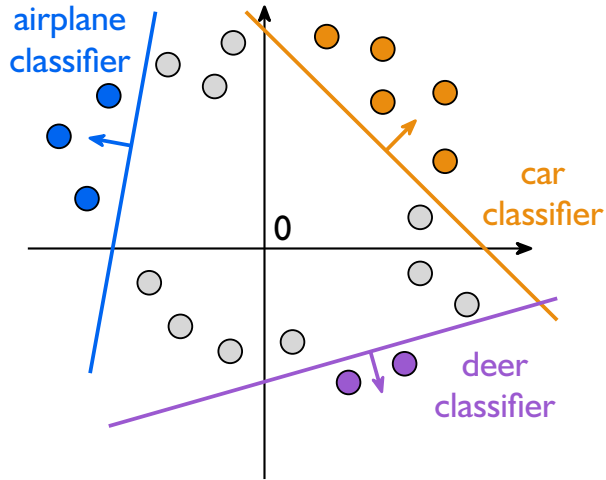
$$a = f\left(\sum_{i=1}^{d} w_i x_i + b\right)$$

Optimal parameters w_i can be found using classical iterative methods

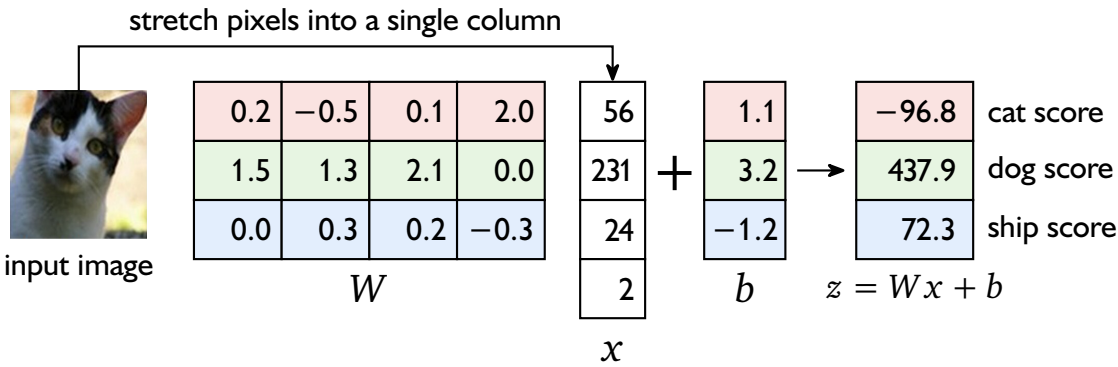
Linear multiclass classifier



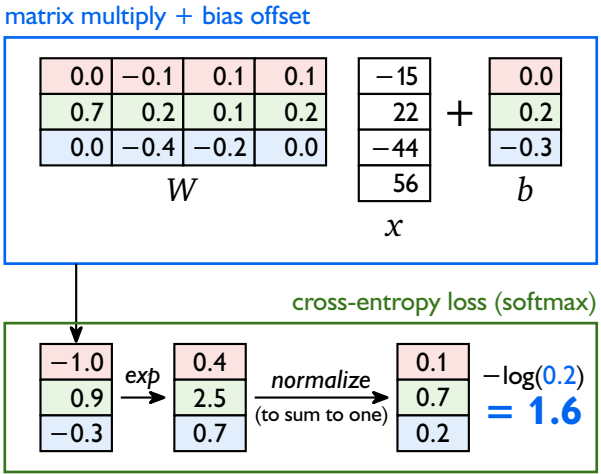
Perceptron



Linear classifier for images



Loss function



Softmax turns scores into probabilities:

$$p_k = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}$$

Categorical cross-entropy:

$$L(z, y) = -\sum_{k=1}^{K} y_k \log p_k$$

Training

Find weights w that minimize the mean loss over N training samples:

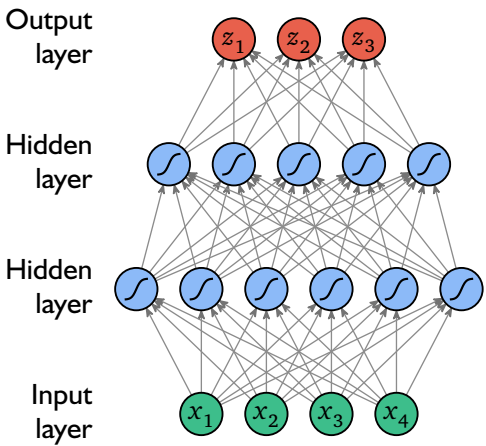
$$\mathscr{L}(w) = \frac{1}{N} \sum_{i=1}^{N} L\big(z(x_i; w), y_i\big) \to \min_w$$

- **Gradient descent:** step against the gradient, η is the learning rate

$$w \leftarrow w - \eta \nabla_w \mathscr{L}(w)$$

- **Minibatch SGD:** gradient on a random batch of 32–256 samples
- **Epoch:** one pass over the training set; training takes many epochs
- **Validation set:** held-out samples to tune η, batch size, architecture

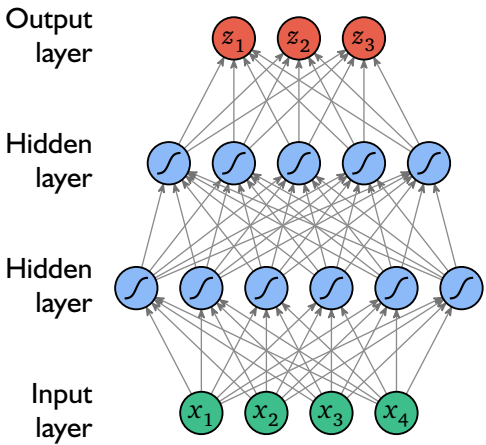
Multilayer perceptron (MLP)



Chained perceptrons may be called **deep neural networks**. Hidden layer neurons usually have nonlinear activation function (e.g. sigmoid). Number of outputs depends on task

Layers in NN may have two meanings: a set of neuron activations (also called representations) and a set of connections with weights

Multilayer perceptron (MLP)

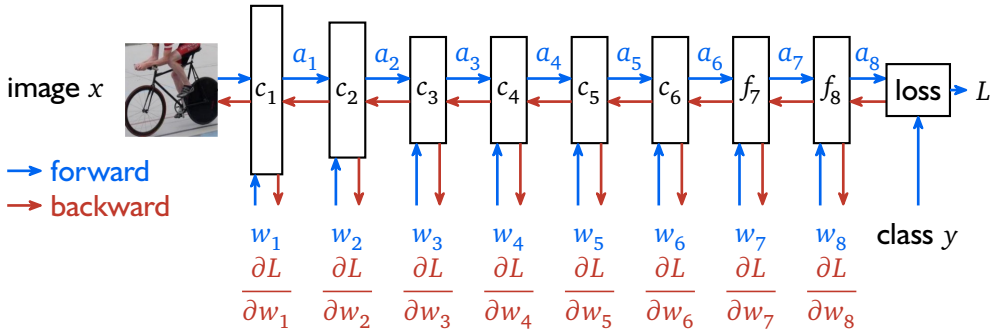


Chained perceptrons may be called **deep neural networks**. Hidden layer neurons usually have nonlinear activation function (e.g. sigmoid). Number of outputs depends on task

Layers in NN may have two meanings: a set of neuron activations (also called representations) and a set of connections with weights

How can we define architecture?

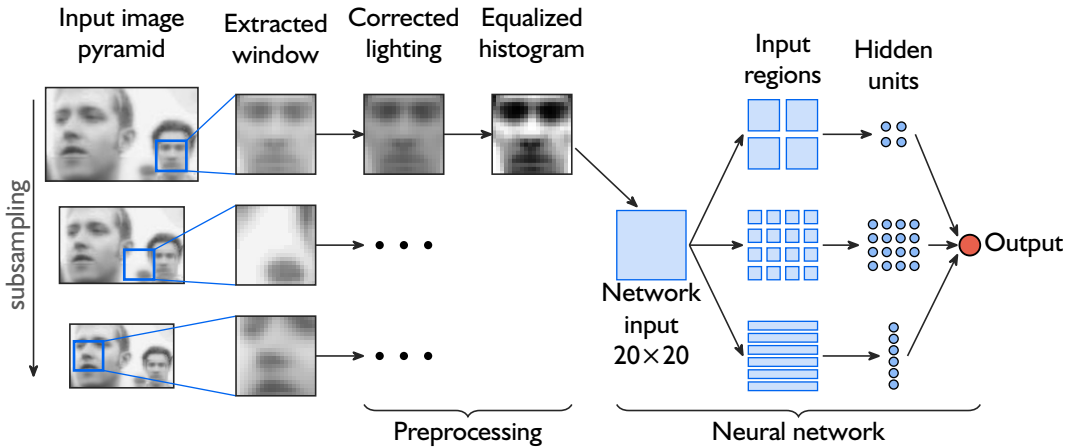
Backpropagation



Chain rule, layer k maps a_{k-1} to a_k ($a_0 = x$):

$$\frac{\partial L}{\partial w_k} = \frac{\partial L}{\partial a_k}\frac{\partial a_k}{\partial w_k} \qquad \frac{\partial L}{\partial a_{k-1}} = \frac{\partial L}{\partial a_k}\frac{\partial a_k}{\partial a_{k-1}}$$

Rowley face detector



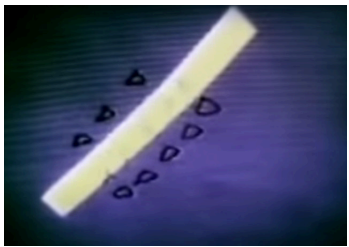
Rowley et al. Neural Network-Based Face Detection. PAMI 1998

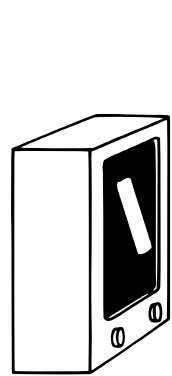
Outline

1. Image classification task and datasets
2. Linear classification and MLPs
3. Convolutional neural networks
4. Milestone: AlexNet

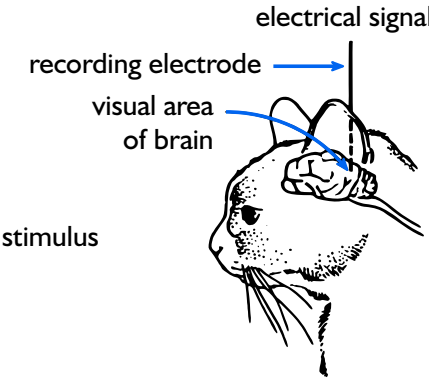
Hubel and Wiesel visual cortex experiments





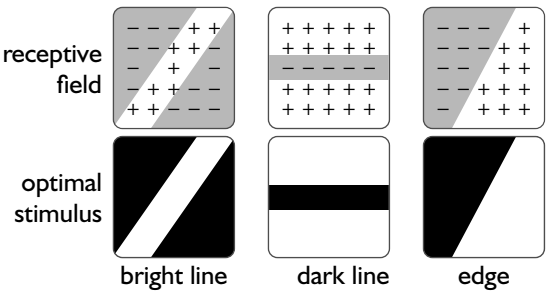


stimulus

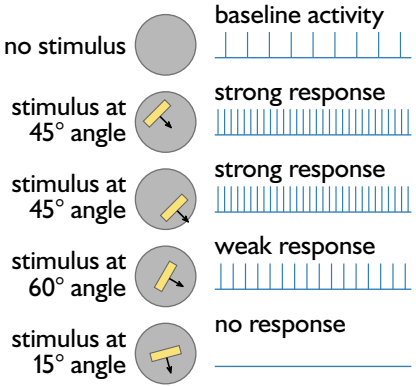


Simple and complex cells

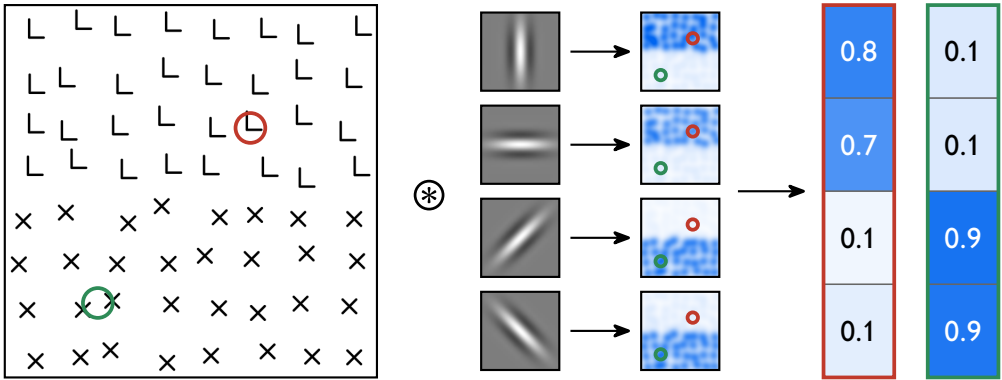
Simple cells



Complex cell



Modelling texture



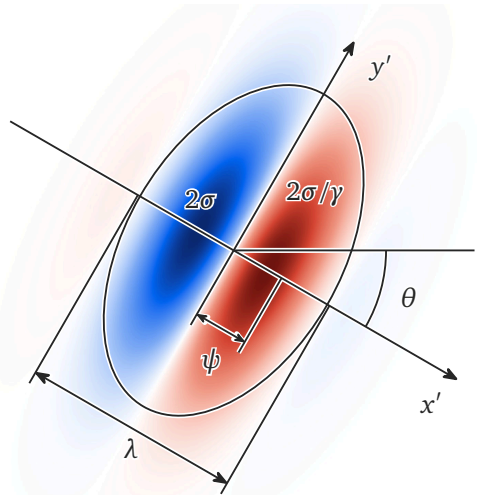
Texture may be described using a bank of filters. Every pixel convolved with filters will give vector of features

Gabor filter as a model for simple cells

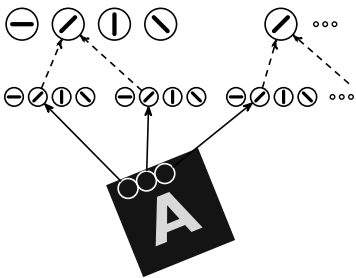
Bank of filters may be obtained using Gabor filters for different orientations

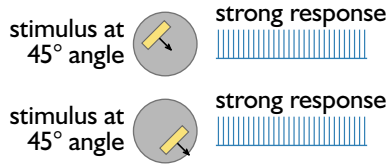
$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right)$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = R(-\theta) \begin{pmatrix} x \\ y \end{pmatrix}$$



Max operation as a model for complex cells

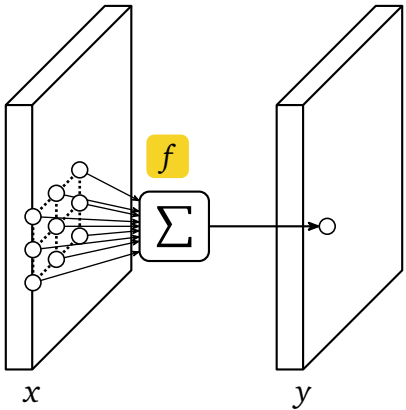




Position invariance (complex cells) may be obtained using MAX operation on top of simple convolutional cells

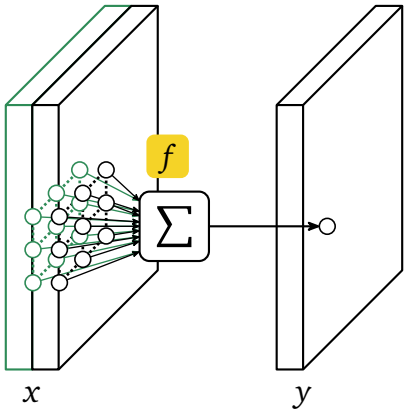
Riesenhuber, Poggio. Hierarchical models of object recognition in cortex. Nature Neuroscience 1999

Convolutional layer



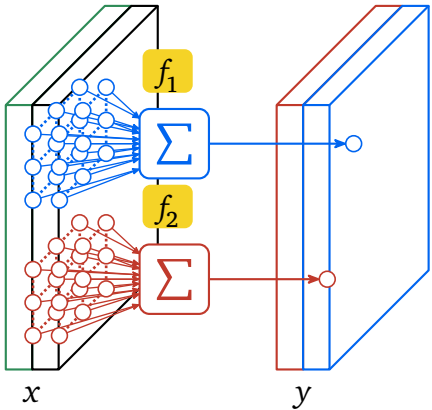
Convolution (linear filtering) for whole image may be modelled using a layer of neurons with shared weights.

Convolutional layer



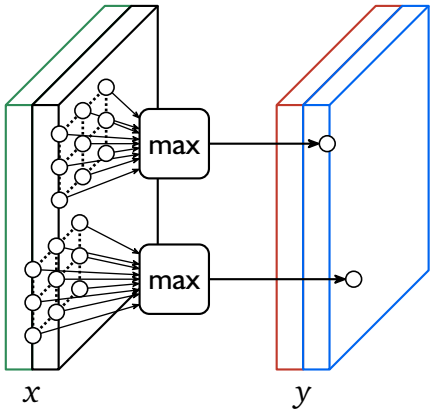
Convolution (linear filtering) for whole image may be modelled using a layer of neurons with shared weights.

Convolutional layer

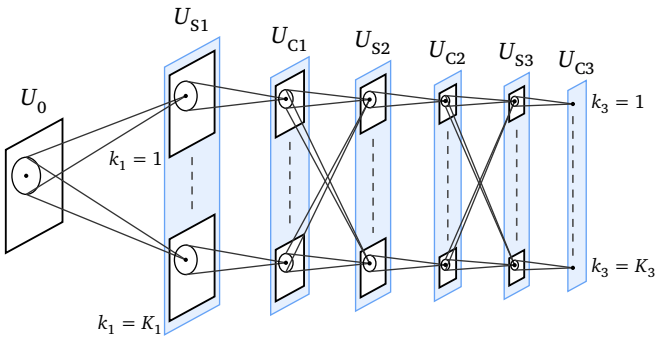


Convolution (linear filtering) for whole image may be modelled using a layer of neurons with shared weights. Convolutional layer is a set of convolutions over the same input

Max pooling layer



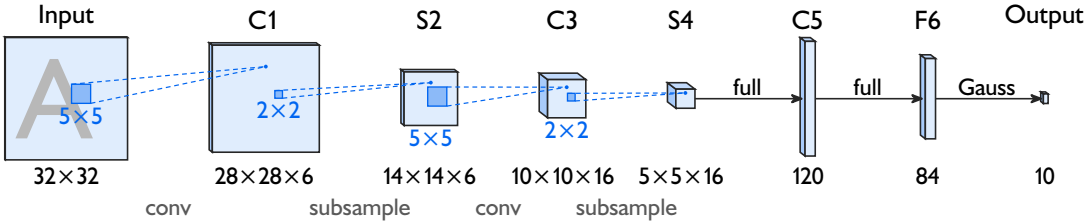
Neocognitron



Multilayer network with interleaved S and C layers. Last layer neurons are invariant to shifts in image

Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 1980

LeNet



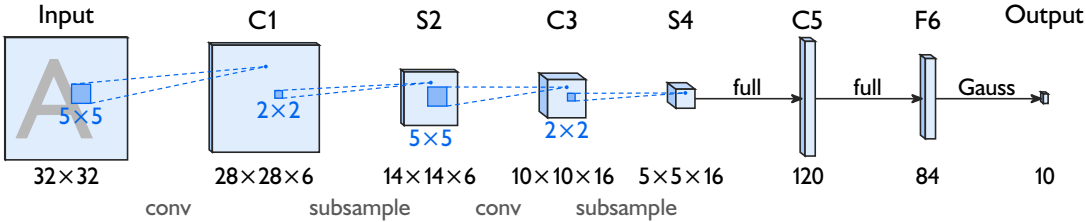
Neocognitron idea + error backpropagation method

→ Convolutional Neural Network (CNN)

Since convolutional neurons share parameters and look at a small neighbourhood, convolutional networks are very effective

LeCun et al. Gradient-based learning applied to document recognition. Proc. IEEE 1998

LeNet



Neocognitron idea + error backpropagation method
→ Convolutional Neural Network (CNN)

Since convolutional neurons share parameters and look at a small neighbourhood, convolutional networks are very effective

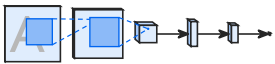
How many trained weights are there in different layers? (C1, S2, …, F6, Output)?

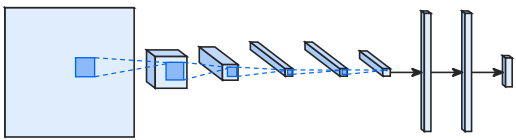
LeCun et al. Gradient-based learning applied to document recognition. Proc. IEEE 1998

Outline

1. Image classification task and datasets
2. Linear classification and MLPs
3. Convolutional neural networks
4. Milestone: AlexNet

LeNet and AlexNet comparison





1998:

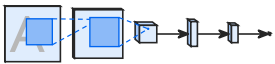
- 2 conv layers
  (6, 16 filters)
- 2 fully connected layers
  (120, 84 neurons)

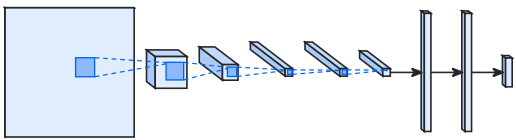
2012:

- 5 conv layers
  (96, 256, 384, 384, 256 filters)
- 2 fully connected layers
  (4096, 4096 neurons)

Krizhevsky et al. ImageNet Classification with Deep Convolutional Neural Networks. NIPS 2012

LeNet and AlexNet comparison





1998:

- 2 conv layers
  (6, 16 filters)
- 2 fully connected layers
  (120, 84 neurons)

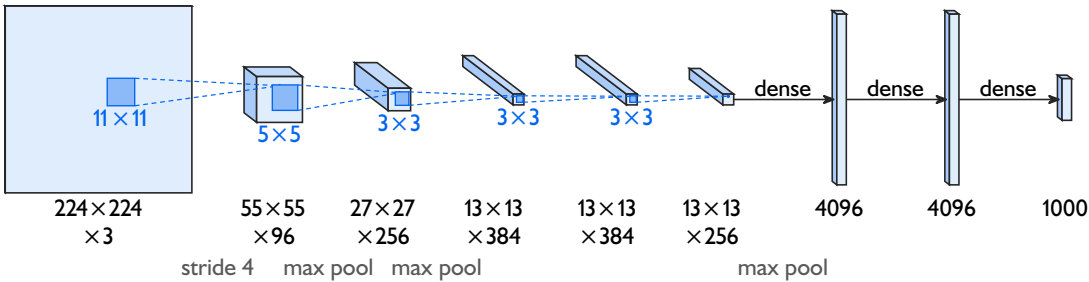
2012:

- 5 conv layers
  (96, 256, 384, 384, 256 filters)
- 2 fully connected layers
  (4096, 4096 neurons)

What else has changed?

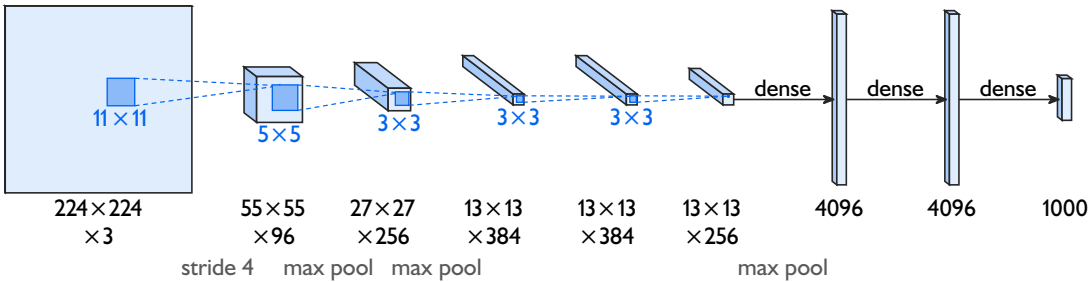
Krizhevsky et al. ImageNet Classification with Deep Convolutional Neural Networks. NIPS 2012

AlexNet



- **Data:** ImageNet, 1.2M images, 20 times more than CIFAR
- **Compute:** 2 GTX 580 GPUs, 1 week to train, CUDA interface
- **Scale:** 60M parameters, 1000 times LeNet
- **Prev. overfitting:** ReLU, image augmentations, dropout

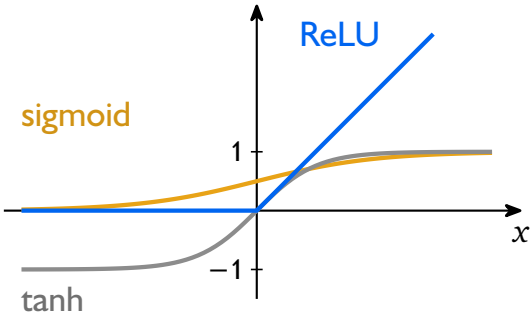
AlexNet



- **Data:** ImageNet, 1.2M images, 20 times more than CIFAR
- **Compute:** 2 GTX 580 GPUs, 1 week to train, CUDA interface
- **Scale:** 60M parameters, 1000 times LeNet
- **Prev. overfitting:** ReLU, image augmentations, dropout

HW: compute *manually* number of parameters for AlexNet

ReLU activation



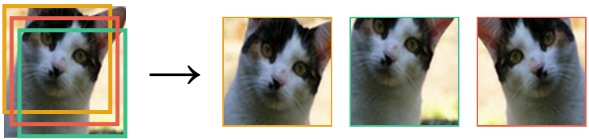
$$\mathrm{ReLU}(x) = \max(0, x)$$

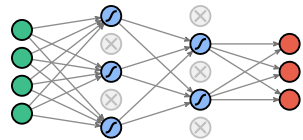
- Sigmoid and tanh *saturate*: for large $|x|$ the gradient is near 0 and vanishes through many layers
- ReLU has gradient 1 for every $x > 0$ and is cheap to compute
- A 4-layer CNN with ReLU reached 25% training error on CIFAR-10 6 times faster than with tanh

Nair, Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. ICML 2010

Fighting overfitting

60M parameters, 1.2M images: easy to memorize the training set





Data augmentation

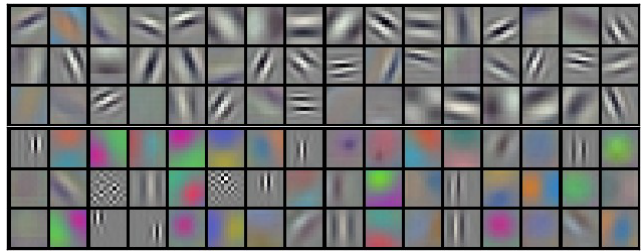
- Random 224 × 224 crops of 256 × 256 images, mirrored: 2048 times more samples
- Color jitter

Dropout

- In training, zero each dense neuron with probability 0.5
- Neurons cannot rely on each other

Srivastava et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. JMLR 2014

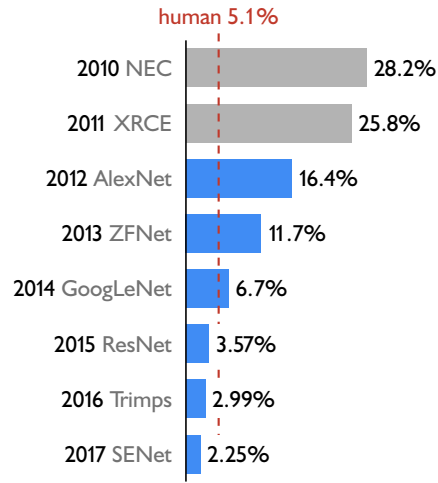
AlexNet: learned first-layer filters



96 kernels with size $11 \times 11 \times 3$ learned from data look like Gabor filters and color blobs, as in the simple cells of the primary visual cortex (V1)

ILSVRC 2012

- AlexNet: 16.4% top-5 error, runner-up: 26.2%
- Every winner since is a CNN
- 2015: below human error



Russakovsky et al. ImageNet Large Scale Visual Recognition Challenge. IJCV 2015

Conclusion

We reviewed following topics:

- image classification tasks and datasets
- how to obtain and label data
- classification with single neuron and MLP, training with backpropagation
- main biological principles behind convolutional neural networks
- AlexNet: data, compute and training tricks that made CNNs win in 2012