

Data Mining

Lecture 14

Ananya Jana
CS360

Fall 2024



Convolutional neural network

- **Background: Computer Vision**
 - Image Classification
 - ILSVRC 2010 - 2016
 - Traditional Feature Extraction Methods
 - Convolution as Feature Extraction
- **Convolutional Neural Networks (CNNs)**
 - Learning Feature Abstractions
 - Common CNN Layers:
 - Convolutional Layer
 - Max-Pooling Layer
 - Fully-connected Layer (w/tensor input)
 - Softmax Layer
 - ReLU Layer
 - Background: Subgradient
 - Architecture: LeNet
 - Architecture: AlexNet
- **Training a CNN**
 - SGD for CNNs
 - Backpropagation for CNNs

Convolutional neural network

Motivation

Why is everyone talking about Deep Learning?



Deep learning:

- Has won numerous pattern recognition competitions
- Does so with minimal feature engineering

This wasn't always the case!

Since 1980s: Form of models hasn't changed much, but lots of new tricks...


- More hidden units
- Better (online) optimization
- New nonlinear functions (ReLU)
- Faster computers (CPUs and GPUs)

Convolutional neural network

Example: Image Classification

- ImageNet LSVRC-2011 contest:
 - **Dataset:** 1.2 million labeled images, 1000 classes
 - **Task:** Given a new image, label it with the correct class
 - **Multiclass** classification problem
- Examples from <http://image-net.org/>

Convolutional neural network

[SEARCH](#)

[Home](#)[About](#)[Explore](#)[Download](#)


Not logged in. [Login](#) [Signup](#)

Bird

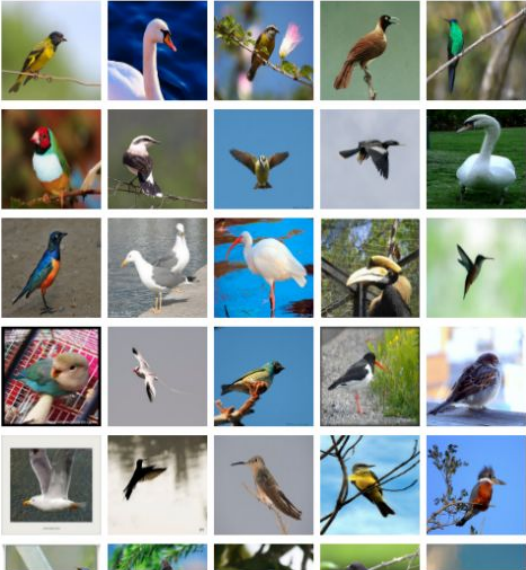
Warm-blooded egg-laying vertebrates characterized by feathers and forelimbs modified as wings

2126 pictures

92.85% Popularity Percentile


Wordnet IDs

[Treemap Visualization](#)[Images of the Synset](#)[Downloads](#)



- marine animal, marine creature, sea animal, sea creature (1)
- scavenger (1)
- biped (0)
- predator, predatory animal (1)
- larva (49)
- acrodont (0)
- feeder (0)
- stunt (0)
- chordate (3087)
 - tunicate, urochordate, urochord (6)
 - cephalochordate (1)
 - vertebrate, craniate (3077)
 - mammal, mammalian (1169)
 - bird (871)
 - dickeybird, dickey-bird, dickybird, dicky-bird (0)
 - cock (1)
 - hen (0)
 - nester (0)
 - night bird (1)
 - bird of passage (0)
 - protoavis (0)
 - archaeopteryx, archeopteryx, Archaeopteryx lithographi Sinornis (0)
 - Ibero-mesornis (0)
 - archaeornis (0)
 - ratite, ratite bird, flightless bird (10)
 - carinate, carinate bird, flying bird (0)
 - passerine, passeriform bird (279)
 - nonpasserine bird (0)
 - bird of prey, raptor, raptorial bird (80)
 - gallinaceous bird, gallinacean (114)

Convolutional neural network

IMAGENET

14,197,122 images, 21841 synsets indexed

SEARCH

Home
About

Explore
Download

Not logged in. [Login](#) | [Signup](#)

German iris, *Iris kochii*

Iris of northern Italy having deep blue-purple flowers; similar to but smaller than *Iris germanica*

469
pictures

49.6%
Popularity
Percentile

Wordnet
IDs

Treemap Visualization

Images of the Synset

Downloads



- halophyte (0)
- succulent (39)
- cultivar (0)
- cultivated plant (0)
- weed (54)
- evergreen, evergreen plant (0)
- deciduous plant (0)
- vine (272)
- creeper (0)
- woody plant, ligneous plant (1868)
- geophyte (0)
- desert plant, xerophyte, xerophytic plant, xerophile, xerophilic
- mesophyte, mesophytic plant (0)
- aquatic plant, water plant, hydrophyte, hydrophytic plant (11)
- tuberous plant (0)
- bulbous plant (179)
- iridaceous plant (27)
 - iris, flag, fleur-de-lis, sword lily (19)
 - bearded iris (4)
 - Florentine iris, orris, *Iris germanica* florentina, *Iris*
 - German iris, *Iris germanica* (0)
 - German iris, *Iris kochii* (0)
 - Dalmatian iris, *Iris pallida* (0)
 - beardless iris (4)
 - bulbous iris (0)
 - dwarf iris, *Iris cristata* (0)
 - stinking iris, gladdon, gladdon iris, stinking gladwyn,
 - Persian iris, *Iris persica* (0)
 - yellow iris, yellow flag, yellow water flag, *Iris pseudacorus*
 - dwarf iris, vernal iris, *Iris verna* (0)
 - blue flag, *Iris versicolor* (0)

Slides taken from <https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf>

Convolutional neural network



14,197,122 images, 21841 synsets indexed

SEARCH

[Home](#)
[About](#)

[Explore](#)
[Download](#)

Not logged in. [Login](#) | [Signup](#)

Court, courtyard

An area wholly or partly surrounded by walls or buildings; "the house was built around an inner court"

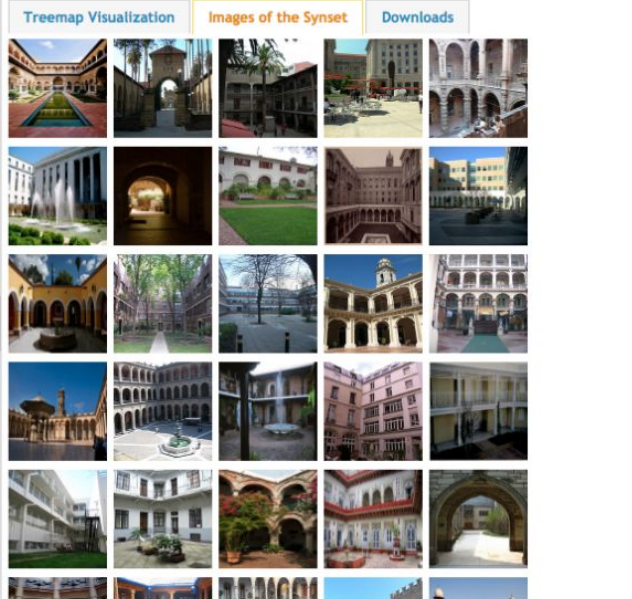
165
pictures

92.61%
Popularity
Percentile


Wordnet
IDs

Numbers in brackets: (the number of synsets in the subtree).

- ImageNet 2011 Fall Release (32326)
 - plant, flora, plant life (4486)
 - geological formation, formation (175)
 - natural object (1112)
 - sport, athletics (176)
 - artifact, artefact (10504)
 - instrumentality, instrumentation (5494)
 - structure, construction (1405)
 - airdock, hangar, repair shed (0)
 - altar (1)
 - arcade, colonnade (1)
 - arch (31)
 - area (344)
 - aisle (0)
 - auditorium (1)
 - baggage claim (0)
 - box (1)
 - breakfast area, breakfast nook (0)
 - bullpen (0)
 - chancel, sanctuary, bema (0)
 - choir (0)
 - corner, nook (2)
 - court, courtyard (6)
 - atrium (0)
 - bailey (0)
 - cloister (0)
 - food court (0)
 - forecourt (0)
 - narvis (0)



Convolutional neural network

Example: Image Classification

Traditional Feature Extraction for Images:

- SIFT
- HOG

Example: Image Classification

CNN for Image Classification

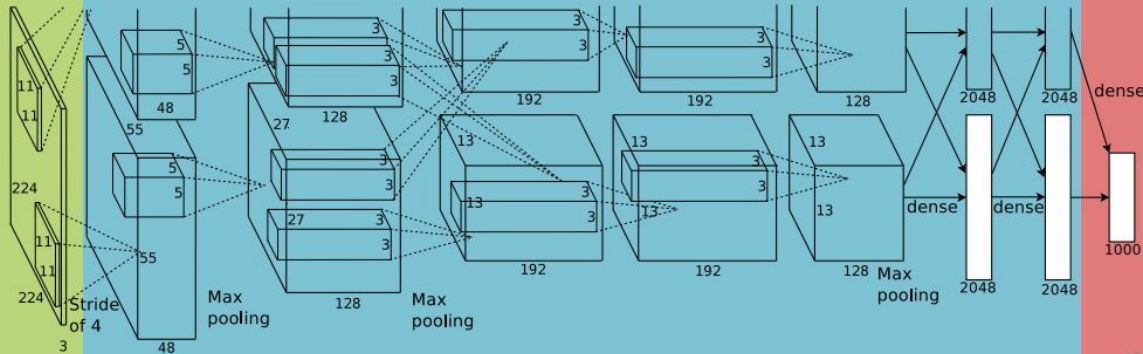
(Krizhevsky, Sutskever & Hinton, 2012)

15.3% error on ImageNet LSVRC-2012 contest

Input
image
(pixels)

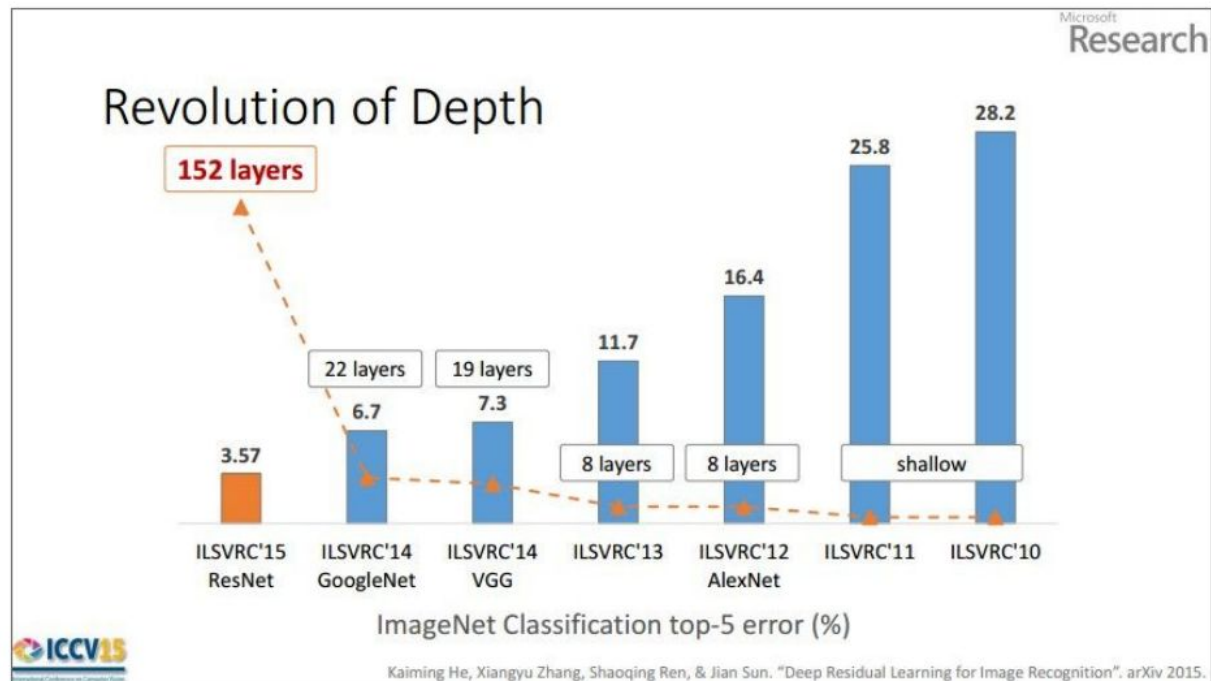
- Five convolutional layers (w/max-pooling)
- Three fully connected layers

1000-way
softmax



Convolutional neural network

CNNs for Image Recognition

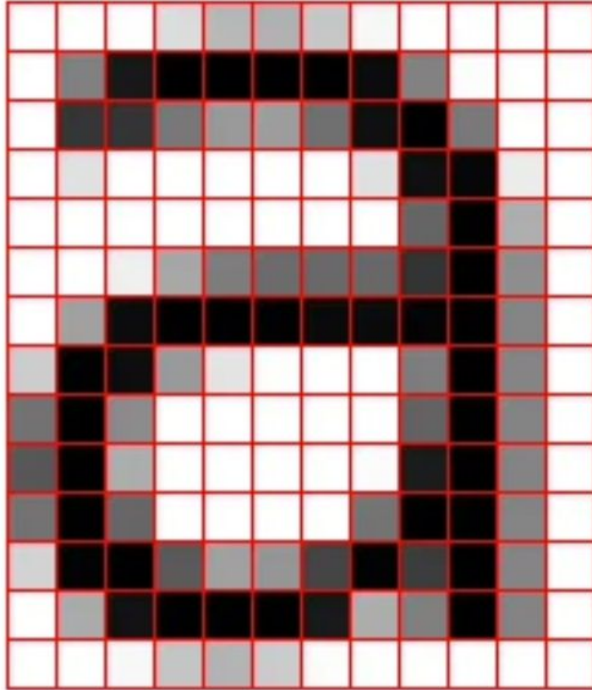


Convolutional neural network

| Year | CNN Architecture | Developed By |
|------|------------------|---|
| 1998 | LeNet | Yann LeCun <i>et al.</i> |
| 2012 | AlexNet | Alex Krizhevsky, Geoffrey Hinton, and Ilya Sutskever |
| 2013 | ZFNet | Matthew Zeiler and Rob Fergus |
| 2014 | GoogleNet | Google |
| 2014 | VGGNet | Simonyan and Zisserman |
| 2015 | ResNet | Kaiming He |
| 2017 | DenseNet | Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger |

Convolutional neural network

a



| | | | | | | | | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 1.0 | 1.0 | 1.0 | 0.9 | 0.6 | 0.6 | 0.6 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 0.5 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 0.2 | 0.2 | 0.5 | 0.6 | 0.6 | 0.5 | 0.0 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 0.9 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.9 | 0.0 | 0.0 | 0.9 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.5 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 | 0.4 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 0.4 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 0.9 | 0.0 | 0.0 | 0.6 | 1.0 | 1.0 | 1.0 | 1.0 | 0.5 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 0.5 | 0.0 | 0.6 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.5 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 0.5 | 0.0 | 0.7 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 0.0 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 0.6 | 0.0 | 0.6 | 1.0 | 1.0 | 1.0 | 1.0 | 0.5 | 0.0 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 0.9 | 0.1 | 0.0 | 0.6 | 0.7 | 0.7 | 0.5 | 0.0 | 0.5 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 0.7 | 0.1 | 0.0 | 0.0 | 0.0 | 0.1 | 0.9 | 0.8 | 0.0 | 0.5 | 1.0 | 1.0 | 1.0 | 1.0 |
| 1.0 | 1.0 | 1.0 | 0.8 | 0.8 | 0.9 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 |

Figure 1: Representation of image as a grid of pixels

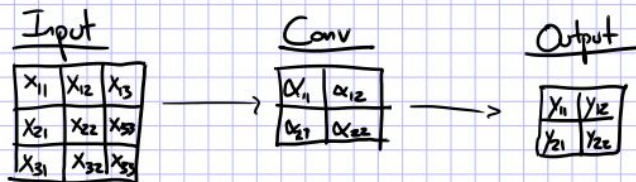
image source http://pippin.gimp.org/image_processing/images/sample_grid_a_square.png

Convolutional neural network

What's a convolution?

- Basic idea:
 - Pick a 3×3 matrix F of weights
 - Slide this over an image and compute the “inner product” (similarity) of F and the corresponding field of the image,
- Key point:
 - Different convolutions extract different types of low-level “features” from an image
 - All that we need to vary to generate these different features is the weights of F

Ex: 1 input channel, 1 output channel



$$y_{11} = \alpha_{11} x_{11} + \alpha_{12} x_{12} + \alpha_{21} x_{21} + \alpha_{22} x_{22} + \alpha_0$$

$$y_{12} = \alpha_{11} x_{12} + \alpha_{12} x_{13} + \alpha_{21} x_{22} + \alpha_{22} x_{23} + \alpha_0$$

$$y_{21} = \alpha_{11} x_{21} + \alpha_{12} x_{22} + \alpha_{21} x_{31} + \alpha_{22} x_{32} + \alpha_0$$

$$y_{22} = \alpha_{11} x_{22} + \alpha_{12} x_{23} + \alpha_{21} x_{32} + \alpha_{22} x_{33} + \alpha_0$$

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 0 | 1 | 0 |

Convolved Image

| | | | | |
|---|---|---|---|---|
| 3 | 2 | 2 | 3 | 1 |
| 2 | 0 | 2 | 1 | 0 |
| 2 | 2 | 1 | 0 | 0 |
| 3 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | 0 | 0 | 0 | 0 |
| | 1 | 1 | 1 | 1 | 1 | 0 |
| | 1 | | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Convolved Image

| | | | | |
|---|--|--|--|--|
| 3 | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | | | | 0 | 0 | 0 |
| 0 | | 1 | 1 | 1 | 1 | 0 |
| 0 | | 0 | | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Convolved Image

| | | | | |
|---|---|--|--|--|
| 3 | 2 | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | | | | 0 | 0 |
| 0 | 1 | | 1 | 1 | 1 | 0 |
| 0 | 1 | | 0 | | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Convolved Image

| | | | | |
|---|---|---|--|--|
| 3 | 2 | 2 | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | | | | 0 |
| 0 | 1 | 1 | | 1 | 1 | 0 |
| 0 | 1 | 0 | | 1 | | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Convolved Image

| | | | | |
|---|---|---|---|--|
| 3 | 2 | 2 | 3 | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | | | |
| 0 | 1 | 1 | 1 | | 1 | 0 |
| 0 | 1 | 0 | 0 | | 0 | |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Convolved Image

| | | | | |
|---|---|---|---|---|
| 3 | 2 | 2 | 3 | 1 |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 0 | 1 | 0 |

Convolved Image

| | | | | |
|---|---|---|---|---|
| 3 | 2 | 2 | 3 | 1 |
| 2 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | | | | 1 | 1 | 0 |
| 0 | | | 0 | 0 | 1 | 0 |
| 0 | | | 0 | | | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | | |
|--|--|--|
| | | |
| | | |
| | | |

Convolved Image

| | | | | |
|---|---|---|---|---|
| 3 | 2 | 2 | 3 | 1 |
| 2 | 0 | | | |
| | | | | |
| | | | | |
| | | | | |

Convolutional neural network

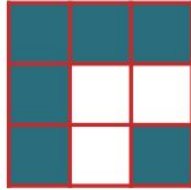
Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

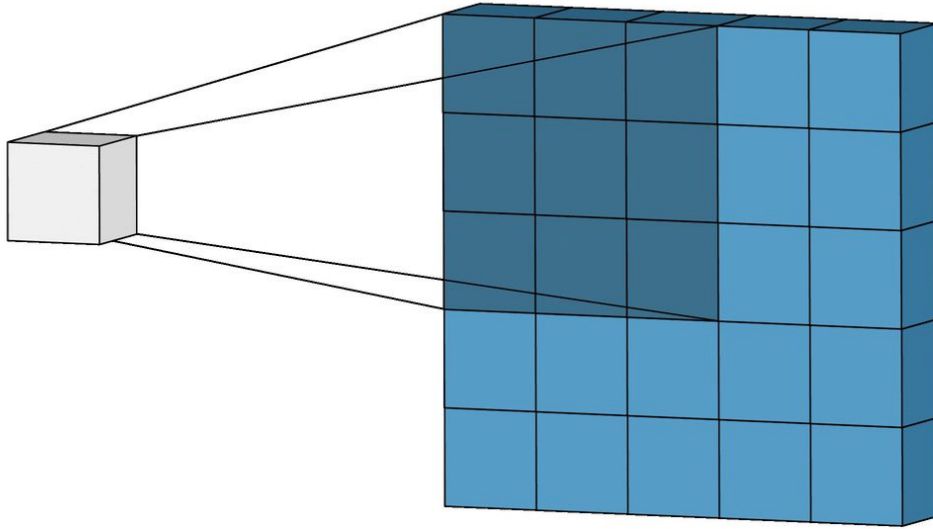
Convolution



Convolved Image

| | | | | |
|---|---|---|---|---|
| 3 | 2 | 2 | 3 | 1 |
| 2 | 0 | 2 | 1 | 0 |
| 2 | 2 | 1 | 0 | 0 |
| 3 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |

Convolutional neural network



Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Identity
Convolution

| | | |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

Convolved Image

| | | | | |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |

Convolutional neural network

Background: Image Processing

A **convolution matrix** is used in image processing for tasks such as edge detection, blurring, sharpening, etc.

Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Blurring
Convolution

| | | |
|----|----|----|
| .1 | .1 | .1 |
| .1 | .2 | .1 |
| .1 | .1 | .1 |

Convolved Image

| | | | | |
|----|----|----|----|----|
| .4 | .5 | .5 | .5 | .4 |
| .4 | .2 | .3 | .6 | .3 |
| .5 | .4 | .4 | .2 | .1 |
| .5 | .6 | .2 | .1 | 0 |
| .4 | .3 | .1 | 0 | 0 |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|--|--|
| 3 | | |
| | | |
| | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|--|
| 3 | 3 | |
| | | |
| | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| | | |
| | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| 3 | | |
| | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| 3 | 1 | |
| | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| 3 | 1 | 0 |
| | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| 3 | 1 | 0 |
| | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| 3 | 1 | 0 |
| 1 | | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| 3 | 1 | 0 |
| 1 | 0 | |

Convolutional neural network

Downsampling

- Suppose we use a convolution with stride 2
- Only 9 patches visited in input, so only 9 pixels in output

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

| | |
|---|---|
| 1 | 1 |
| 1 | 1 |

Convolved Image

| | | |
|---|---|---|
| 3 | 3 | 1 |
| 3 | 1 | 0 |
| 1 | 0 | 0 |

Convolutional neural network


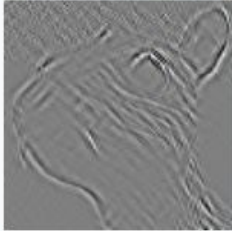


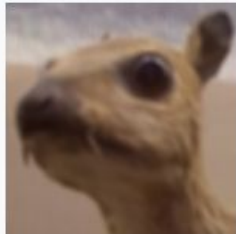
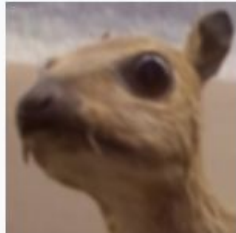
| Operation | Kernel ω | Image result $g(x,y)$ |
|-------------------------|---|---|
| Identity | $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ |  |
| Ridge or edge detection | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ |  |
| | $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ |  |

image source: [https://en.wikipedia.org/wiki/Kernel_\(image_processing\)](https://en.wikipedia.org/wiki/Kernel_(image_processing))

Convolutional neural network

| | | |
|---|--|---|
| Sharpen | $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ |  |
| Box blur (normalized) | $\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ |  |
| Gaussian blur 3 × 3 (approximation) | $\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ |  |

Convolutional neural network

| | | |
|---|---|--|
| Gaussian blur 5×5 (approximation) | $\frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$ |  |
| Unsharp masking 5×5 Based on Gaussian blur with amount as 1 and threshold as 0 (with no image mask) | $\frac{-1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & -476 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$ |  |

image source: [https://en.wikipedia.org/wiki/Kernel_\(image_processing\)](https://en.wikipedia.org/wiki/Kernel_(image_processing))

Convolutional neural network

Convolutional Layer

CNN key idea:
Treat convolution matrix as
parameters and learn them!



Input Image

| | | | | | | |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Learned
Convolution

| | | |
|---------------|---------------|---------------|
| θ_{11} | θ_{12} | θ_{13} |
| θ_{21} | θ_{22} | θ_{23} |
| θ_{31} | θ_{32} | θ_{33} |

Convolved Image

| | | | | |
|----|----|----|----|----|
| .4 | .5 | .5 | .5 | .4 |
| .4 | .2 | .3 | .6 | .3 |
| .5 | .4 | .4 | .2 | .1 |
| .5 | .6 | .2 | .1 | 0 |
| .4 | .3 | .1 | 0 | 0 |

Convolutional neural network

Downsampling by Averaging

- Downsampling by averaging **used to be** a common approach
- This is a special case of convolution where the weights are fixed to a uniform distribution
- The example below uses a stride of 2

Input Image

| | | | | | |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Convolution

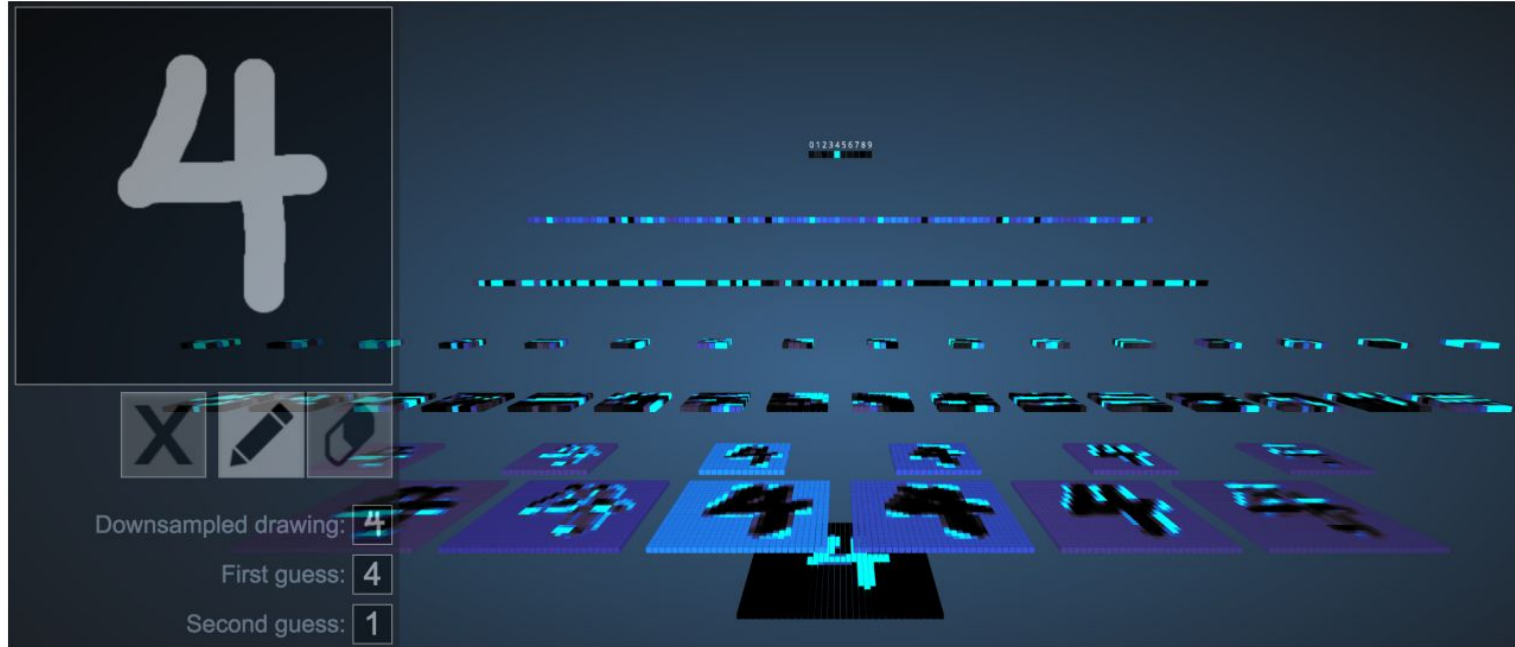
| | |
|---------------|---------------|
| $\frac{1}{4}$ | $\frac{1}{4}$ |
| $\frac{1}{4}$ | $\frac{1}{4}$ |

Convolved Image

| | | |
|---------------|---------------|---------------|
| $\frac{3}{4}$ | $\frac{3}{4}$ | $\frac{1}{4}$ |
| $\frac{3}{4}$ | $\frac{1}{4}$ | 0 |
| $\frac{1}{4}$ | 0 | 0 |

Convolutional neural network

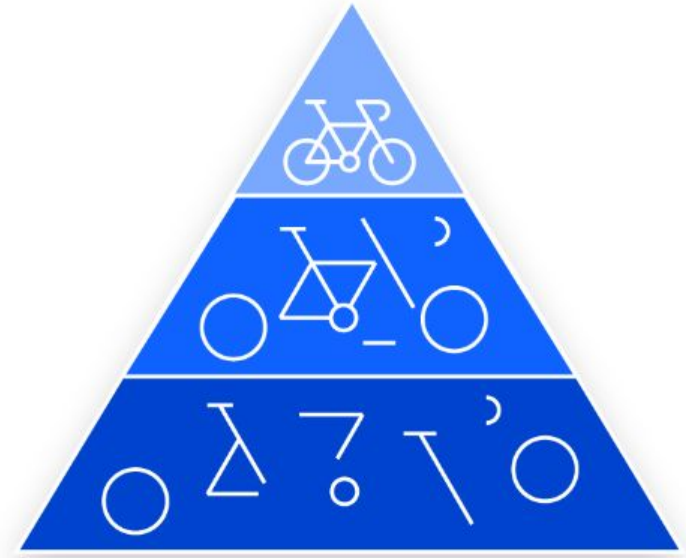
3D Visualization of CNN



Convolutional neural network

Why multiple layers?

Multiple convolution layer can follow the initial convolution layer. When this happens, the structure of the CNN can become hierarchical as the later layers can see the pixels within the receptive fields of prior layers. As an example, let's assume that we're trying to determine if an image contains a bicycle. You can think of the bicycle as a sum of parts. It is comprised of a frame, handlebars, wheels, pedals, et cetera. Each individual part of the bicycle makes up a lower-level pattern in the neural net, and the combination of its parts represents a higher-level pattern, creating a feature hierarchy within the CNN.

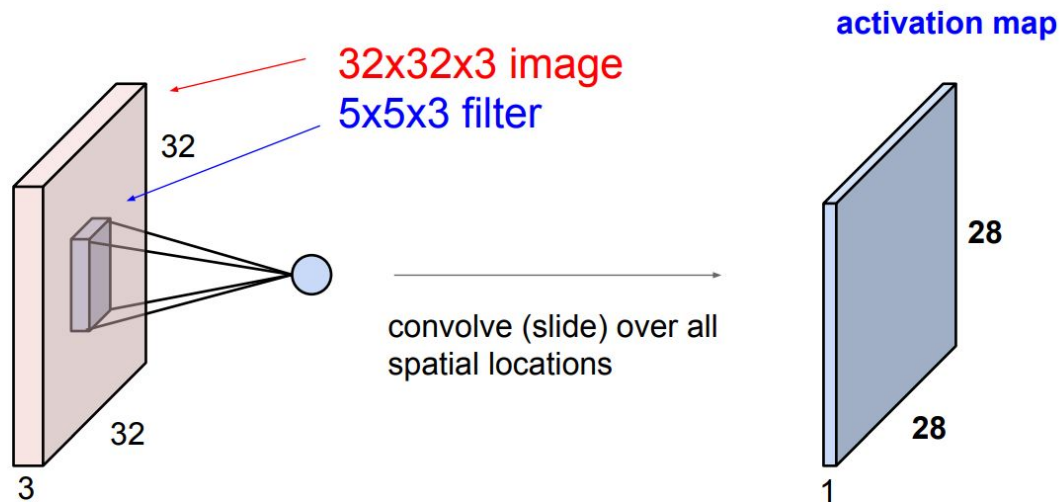


Source: <https://www.ibm.com/topics/convolutional-neural-networks>

Convolutional neural network

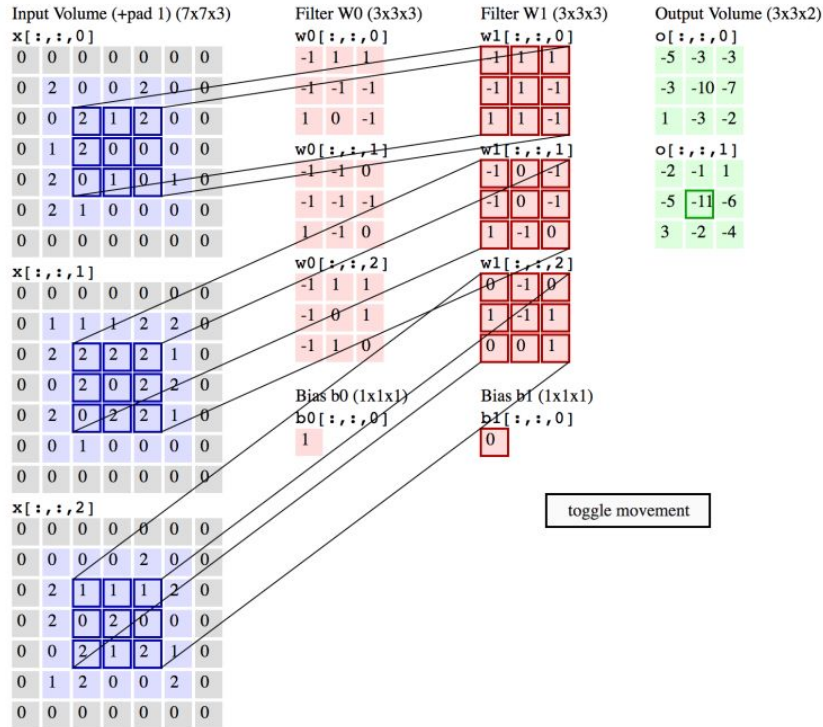
Convolution of a Color Image

- Color images consist of 3 floats per pixel for RGB (red, green blue) color values
- Convolution must also be 3-dimensional



Convolutional neural network

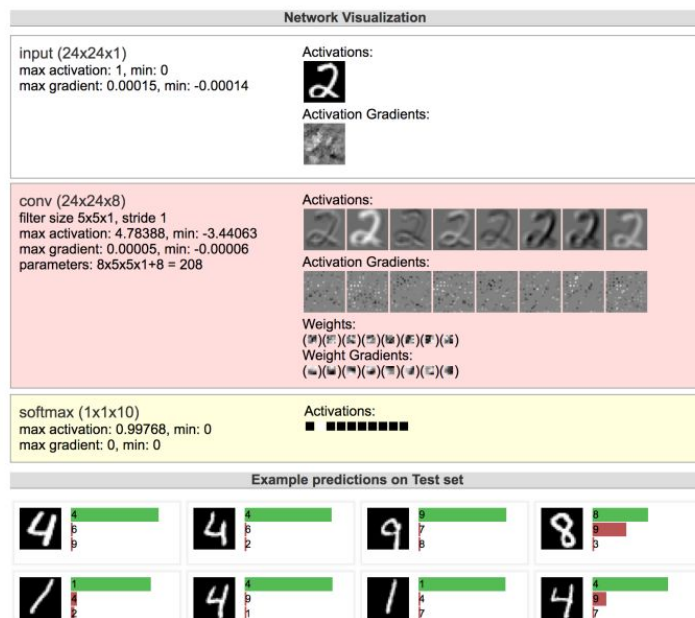
<http://cs231n.github.io/convolutional-networks/>



Convolutional neural network

MNIST Digit Recognition with CNNs (in your browser)

<https://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html>



Link: <https://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html>