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

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

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 (ReLUs)
- Faster computers (CPUs and GPUs)

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

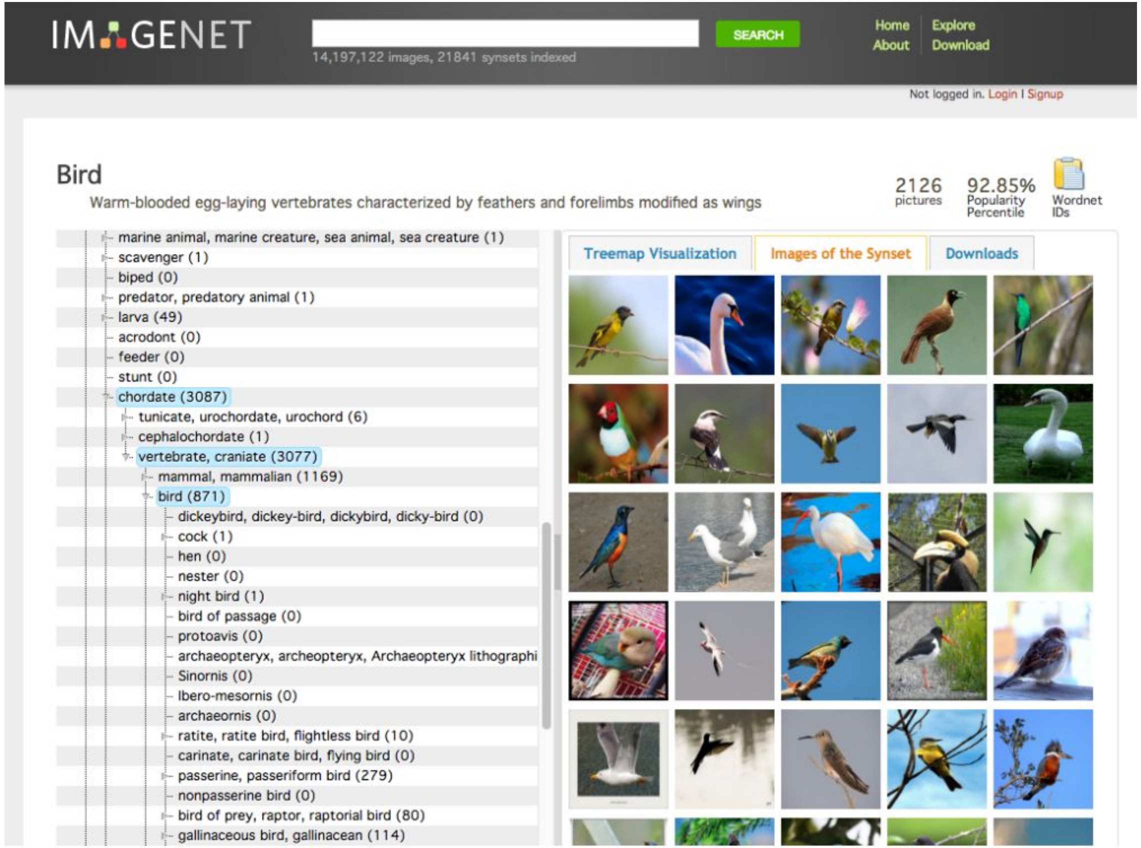
Convolutional neural network

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/

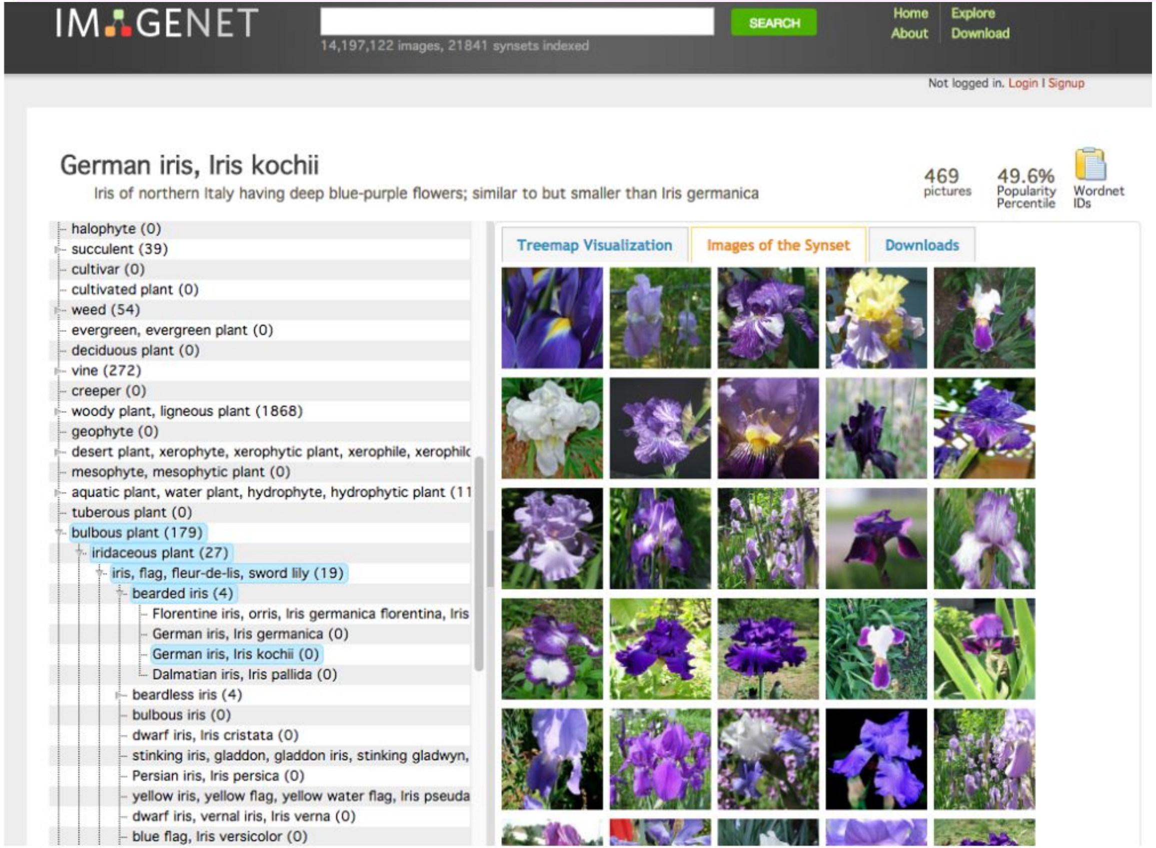
Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network



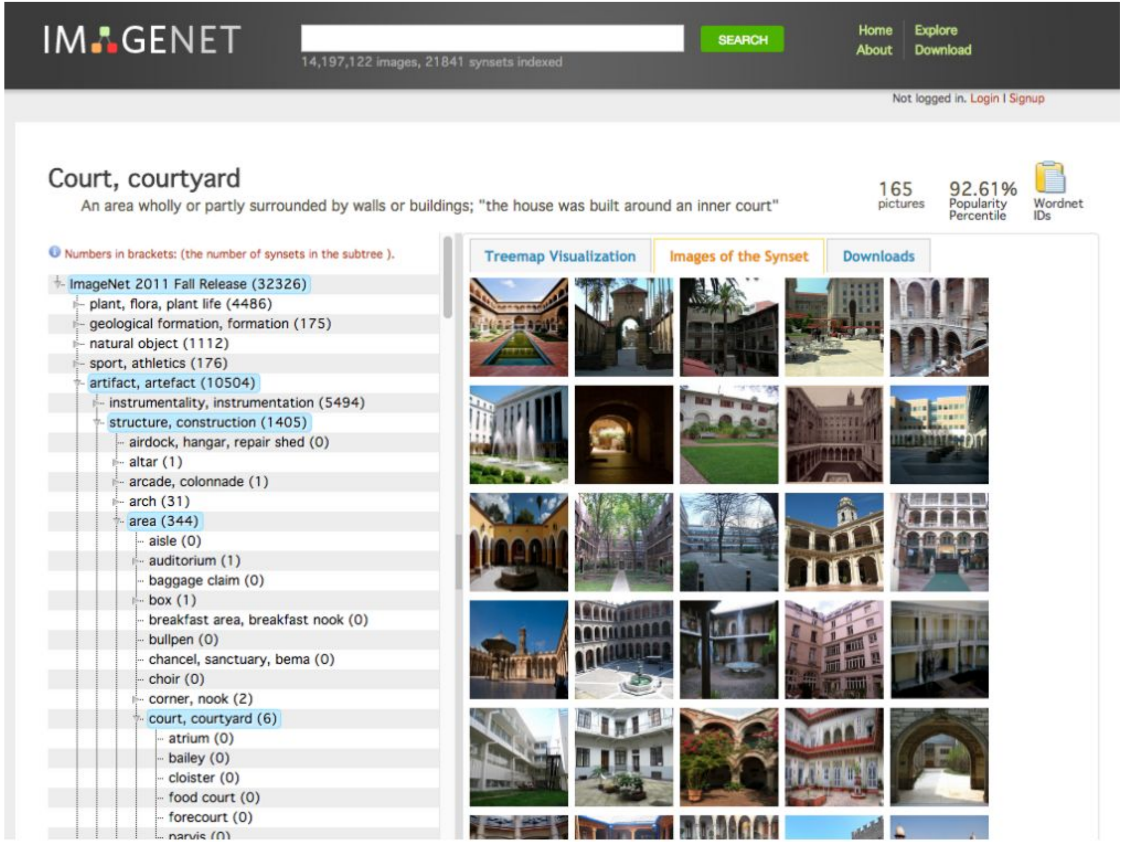
Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network



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

Convolutional neural network



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

Convolutional neural network

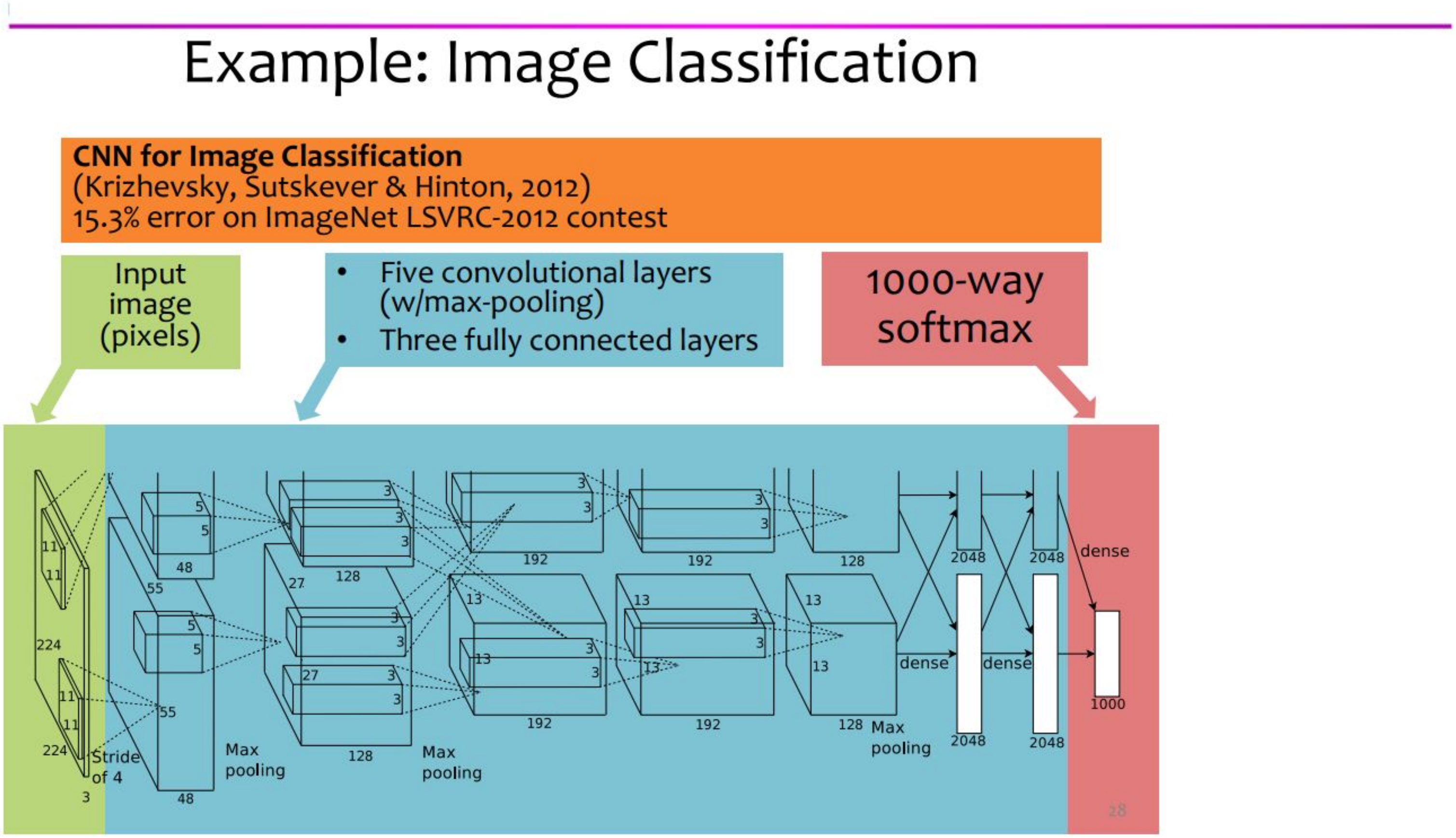
Example: Image Classification

Traditional Feature Extraction for Images:

- SIFT
- HOG

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

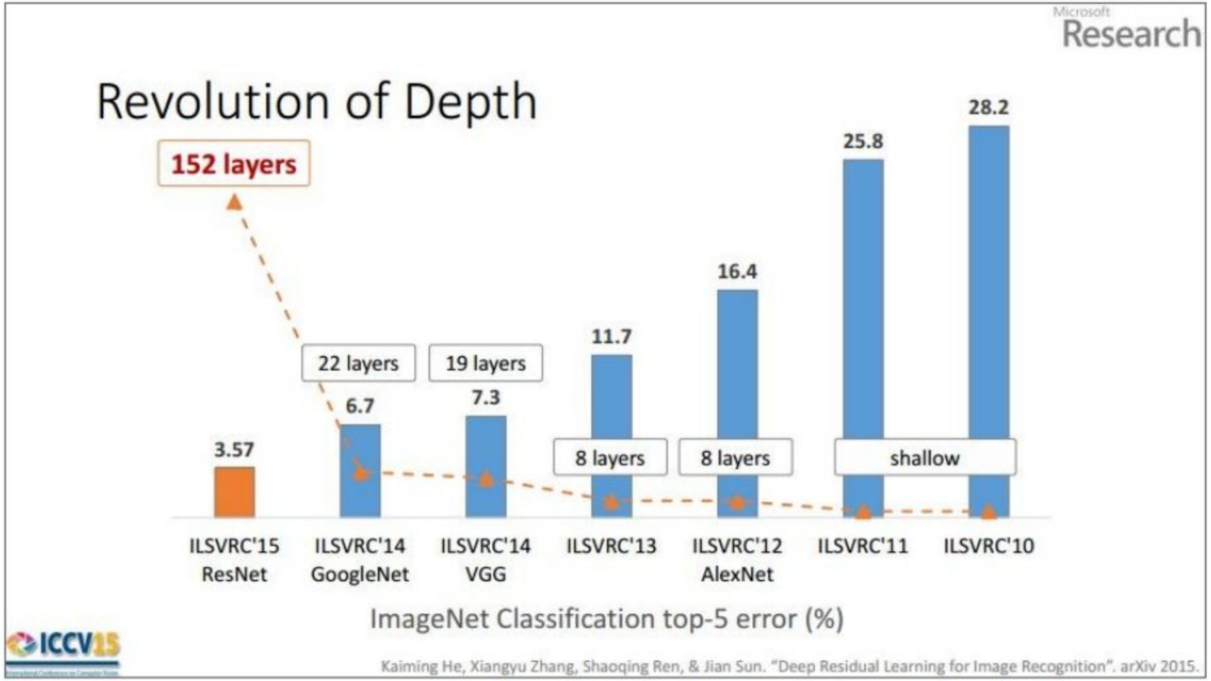
Convolutional neural network



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

Convolutional neural network

CNNs for Image Recognition



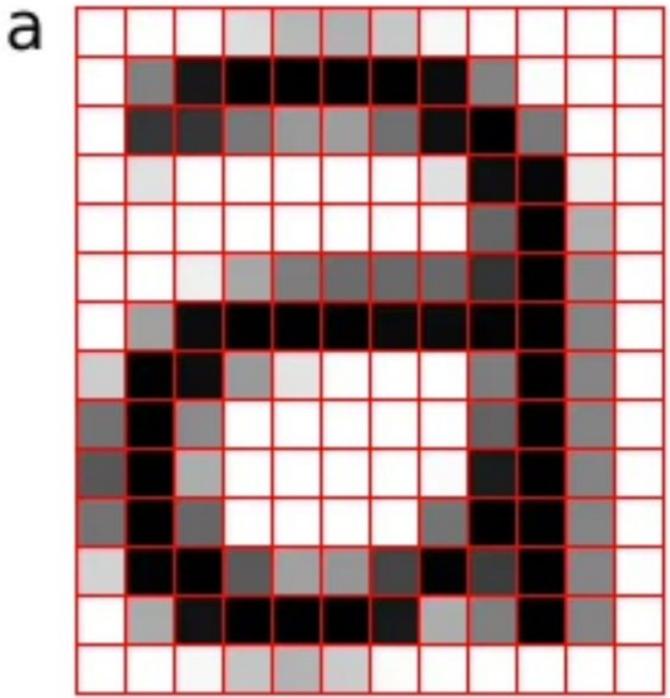
Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Year	CNN Architecture	Developed By
1998	LeNet	Yann LeCun *et al.*
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

image source https://www.slideshare.net/slideshow/cnn-126271677/126271677

Convolutional neural network



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	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.5	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	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	0.5	0.0	0.5	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	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	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
0.5	0.0	0.6	1.0	1.0	1.0	1.0	1.0	0.5	0.0	0.5	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
0.6	0.0	0.6	1.0	1.0	1.0	1.0	0.5	0.0	0.0	0.5	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	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	0.8	0.8	0.9	1.0	1.0	1.0	1.0	1.0	1.0

Figure 1: Representation of image as a grid of pixels

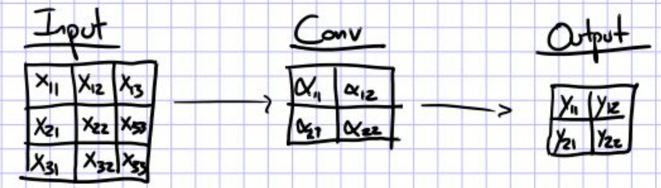
image source http://pippin.gimp.org/image_processing/images/sample_grid_a_square.png

Convolutional neural network

What's a convolution?

- Basic idea:
  - Pick a 3x3 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



$$y_{11} = \alpha_{11}x_{11} + \alpha_{12}x_{12} + \alpha_{21}x_{21} + \alpha_{22}x_{22} + \alpha_0$$
$$y_{12} = \alpha_{11}x_{12} + \alpha_{12}x_{13} + \alpha_{21}x_{22} + \alpha_{22}x_{23} + \alpha_0$$
$$y_{21} = \alpha_{11}x_{21} + \alpha_{12}x_{22} + \alpha_{21}x_{31} + \alpha_{22}x_{32} + \alpha_0$$
$$y_{22} = \alpha_{11}x_{22} + \alpha_{12}x_{23} + \alpha_{21}x_{32} + \alpha_{22}x_{33} + \alpha_0$$

Slide adapted from William Cohen

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

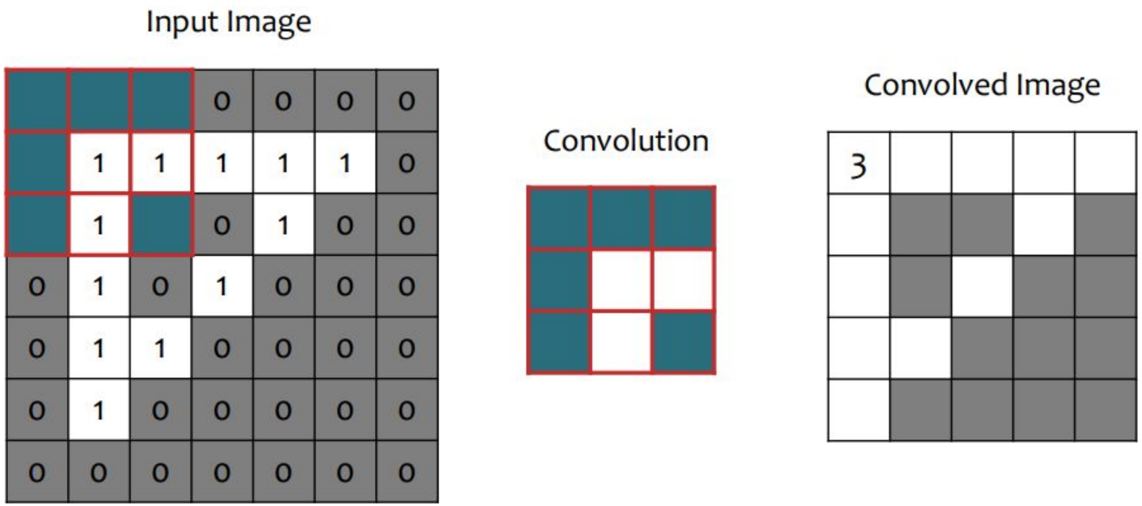
3	2	2	3	1
2	0	2	1	0
2	2	1	0	0
3	1	0	0	0
1	0	0	0	0

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

Convolutional neural network

Background: Image Processing

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

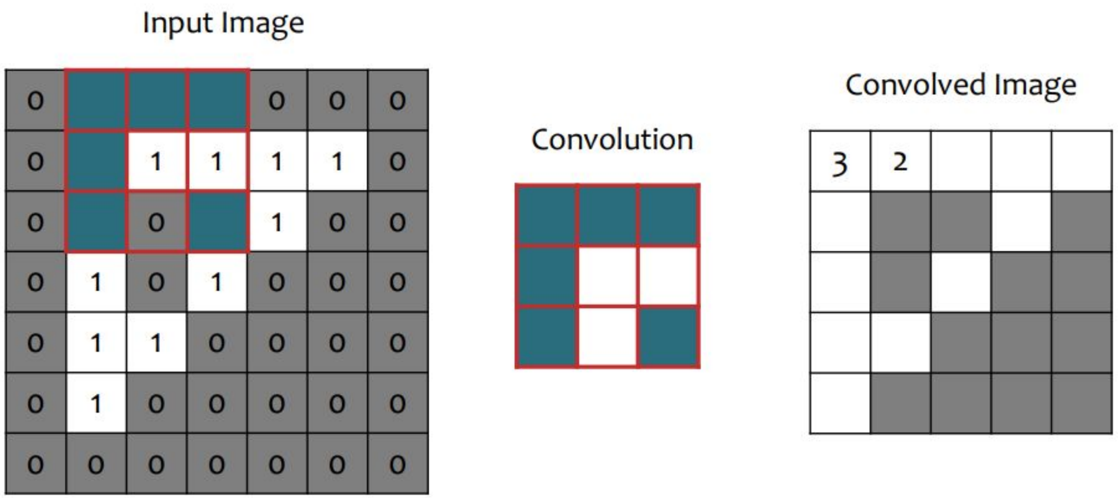


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Background: Image Processing

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

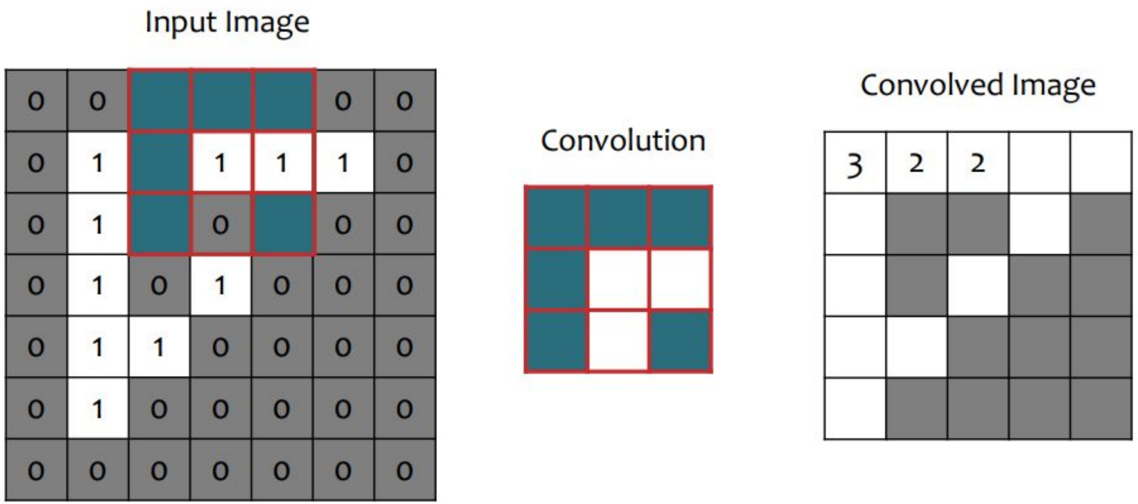


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Background: Image Processing

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

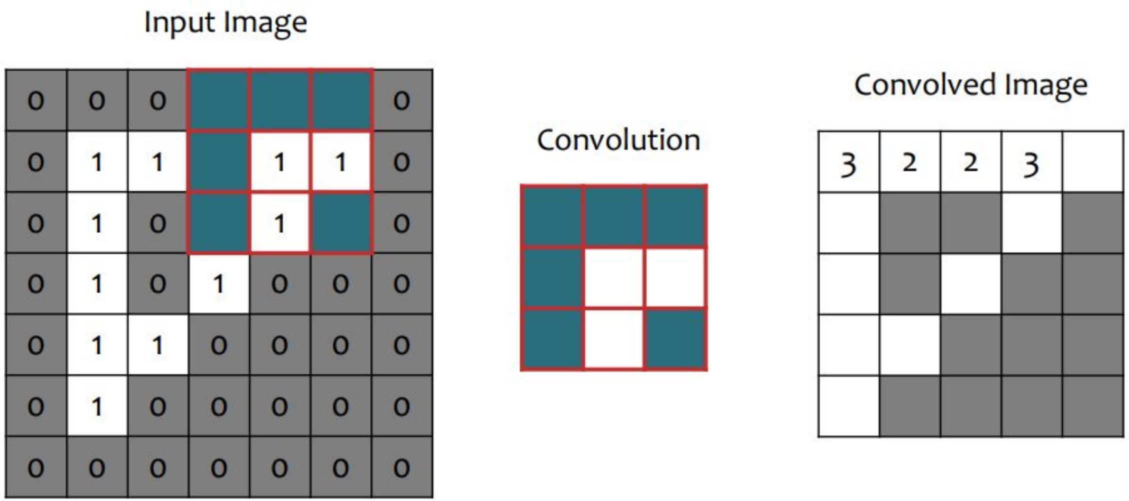


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Background: Image Processing

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

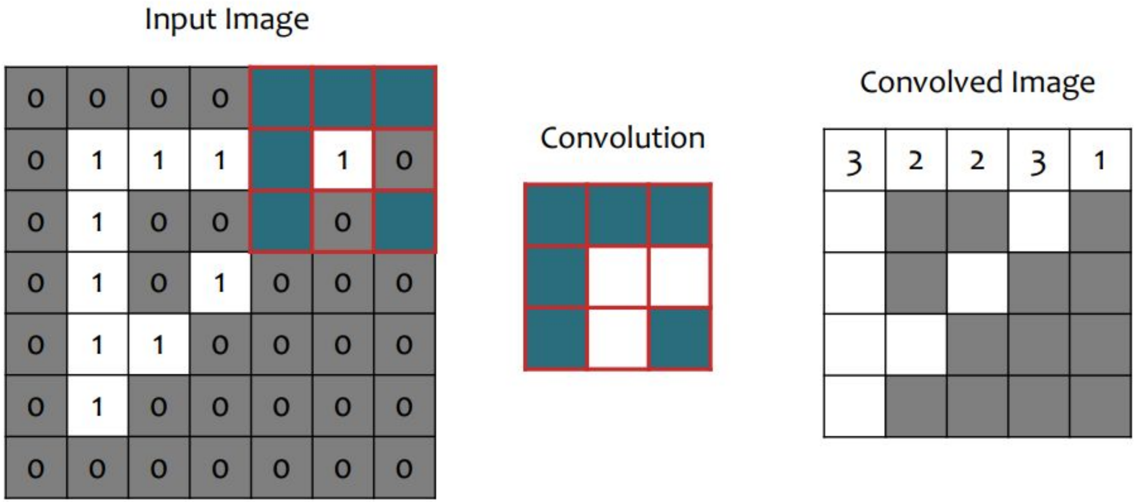


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Background: Image Processing

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

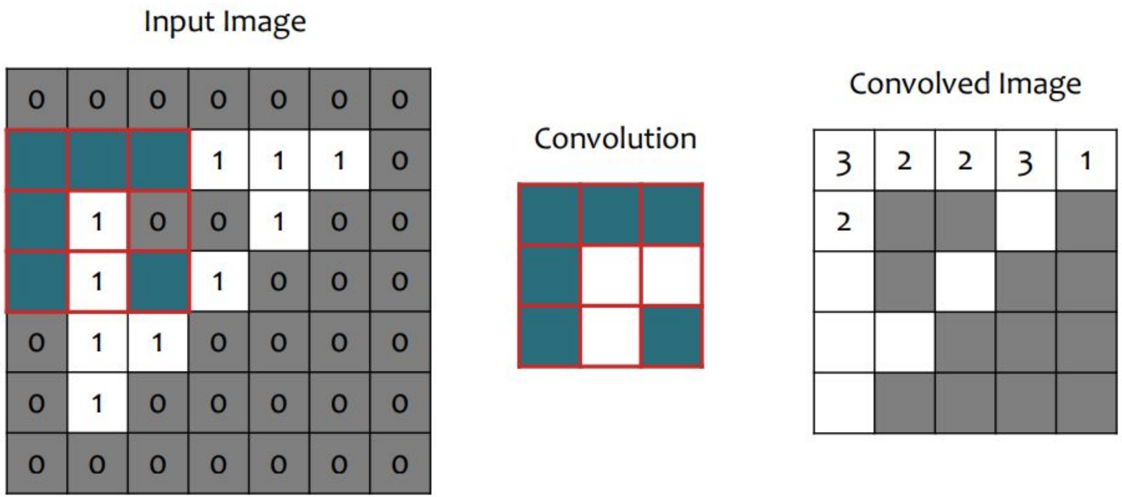


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Background: Image Processing

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

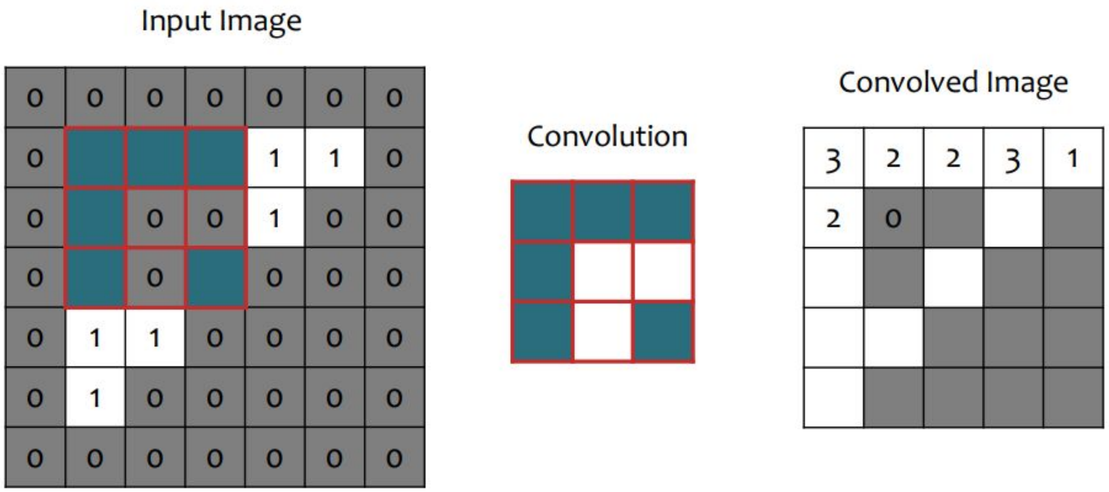


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Background: Image Processing

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

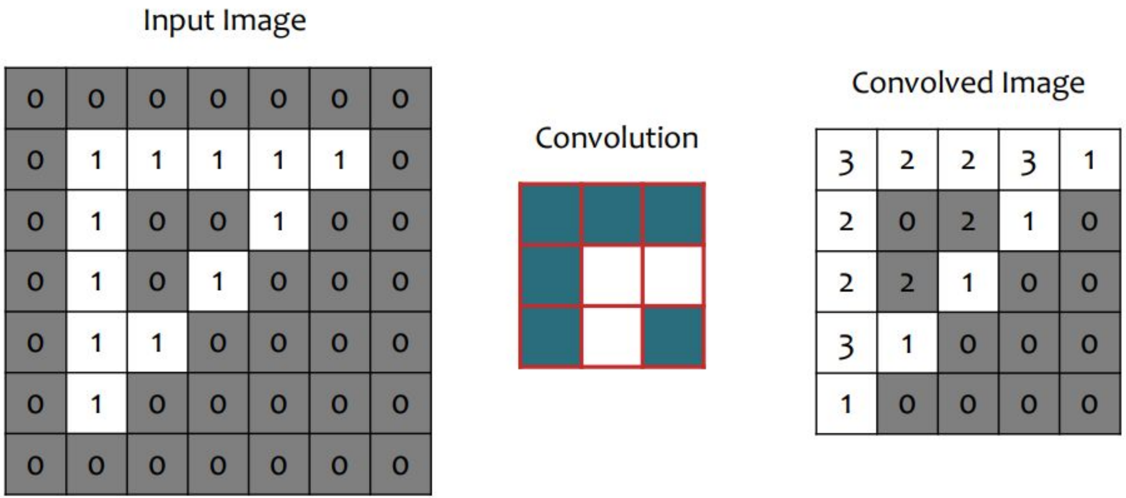


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Background: Image Processing

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



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

Convolutional neural network

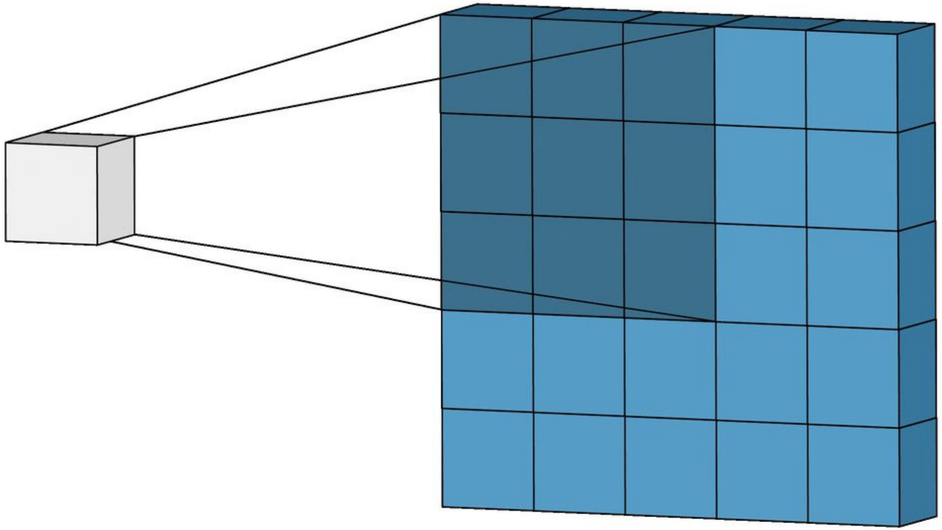


Image source https://towardsdatascience.com/convolutional-neural-networks-explained-9cc5188c4939

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

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

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

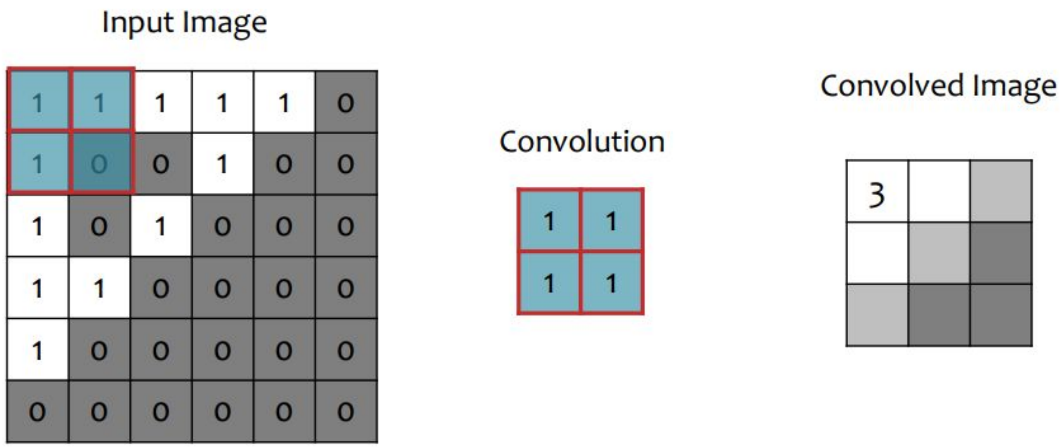
.4	.5	.5	.5	.4
.4	.2	.3	.6	.3
.5	.4	.4	.2	.1
.5	.6	.2	.1	0
.4	.3	.1	0	0

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

Convolutional neural network

Downsampling

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

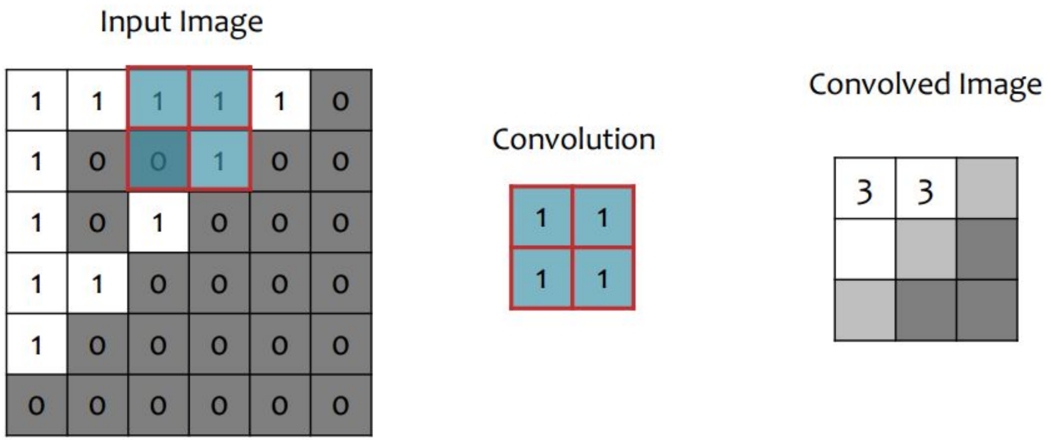


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

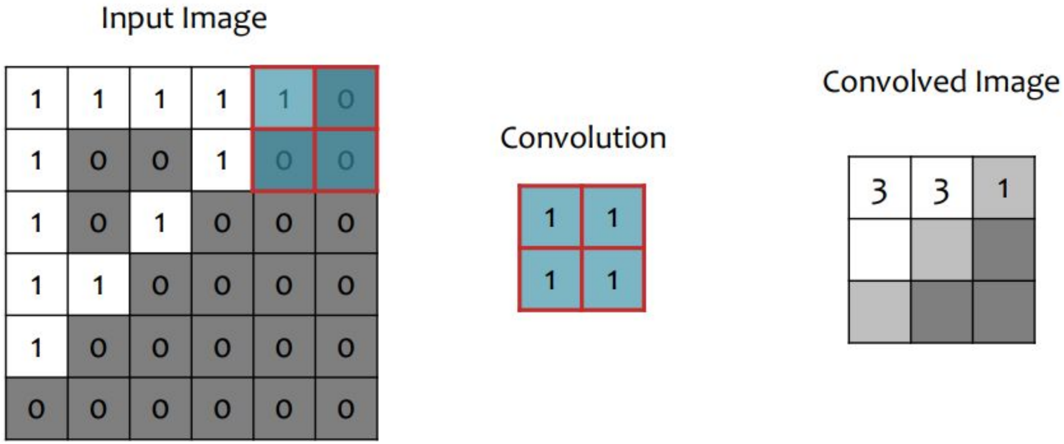


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

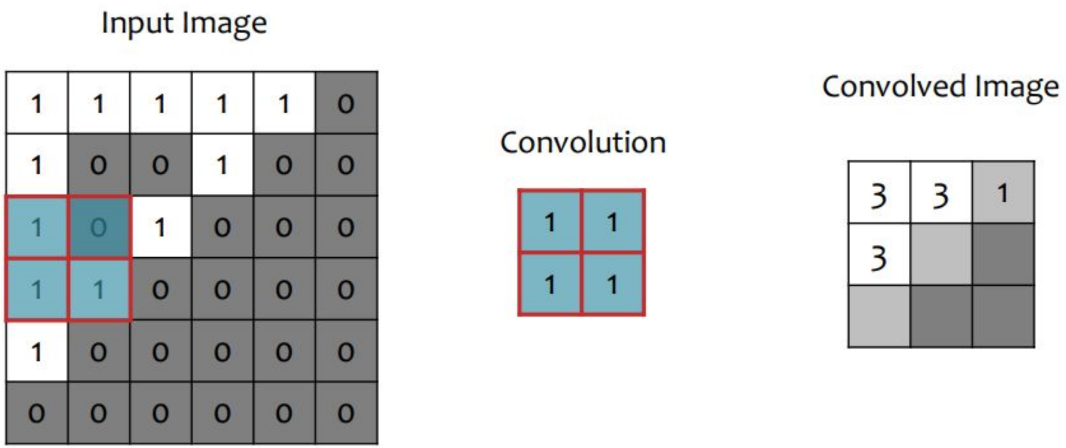


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

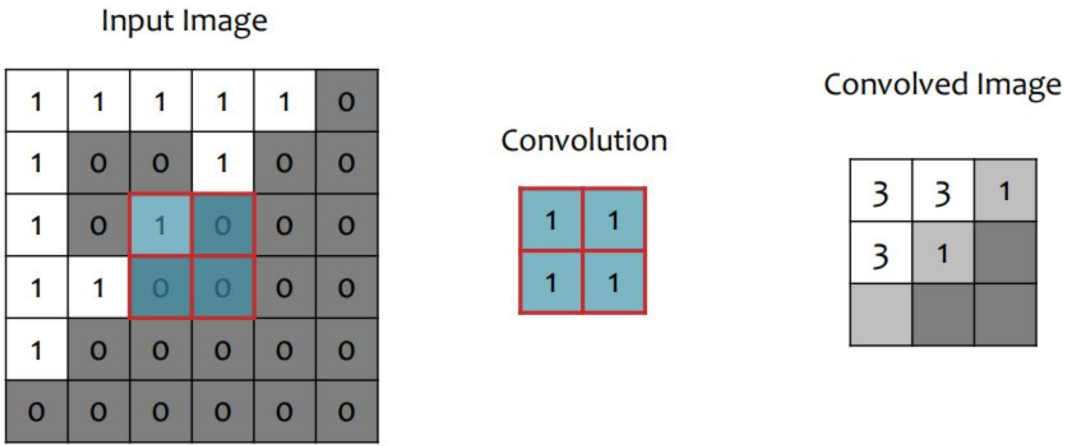


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

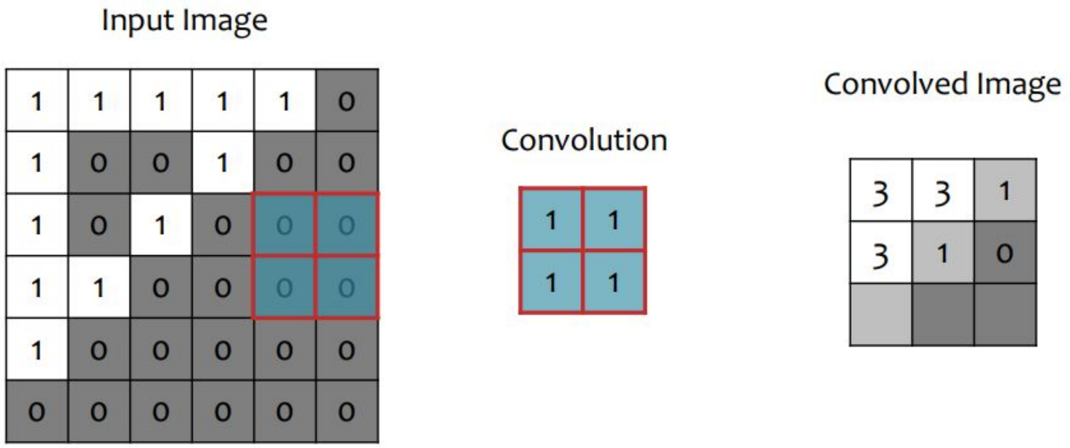


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

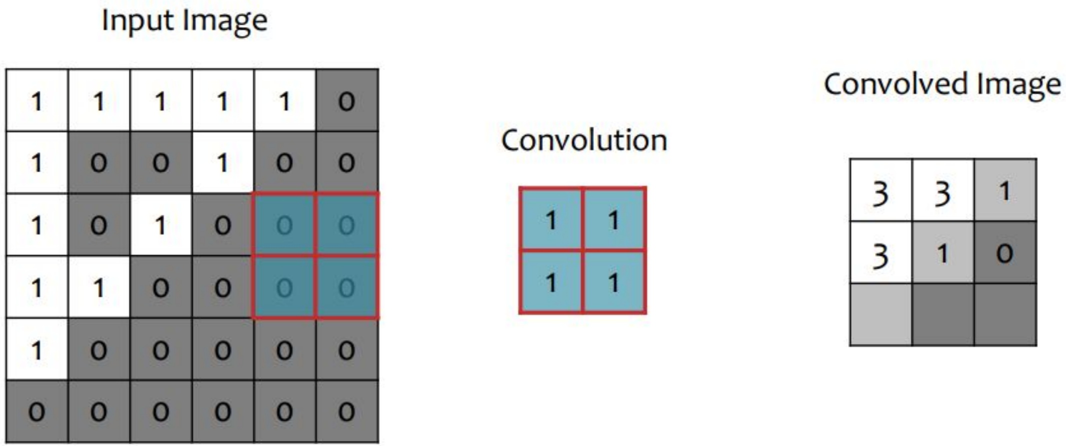


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

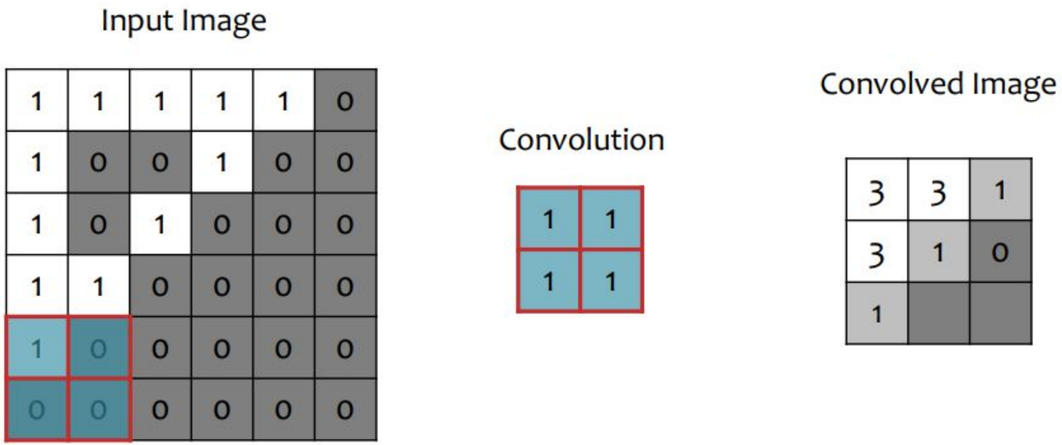


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

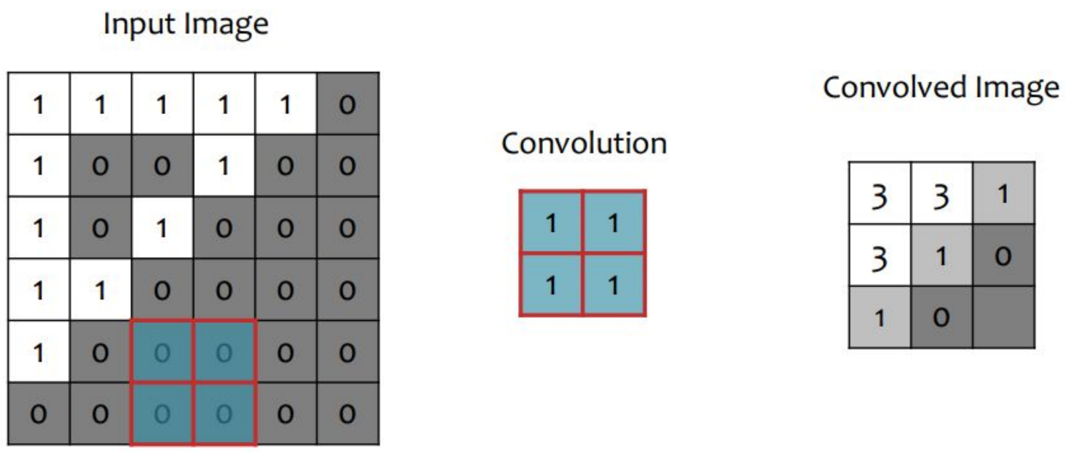


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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

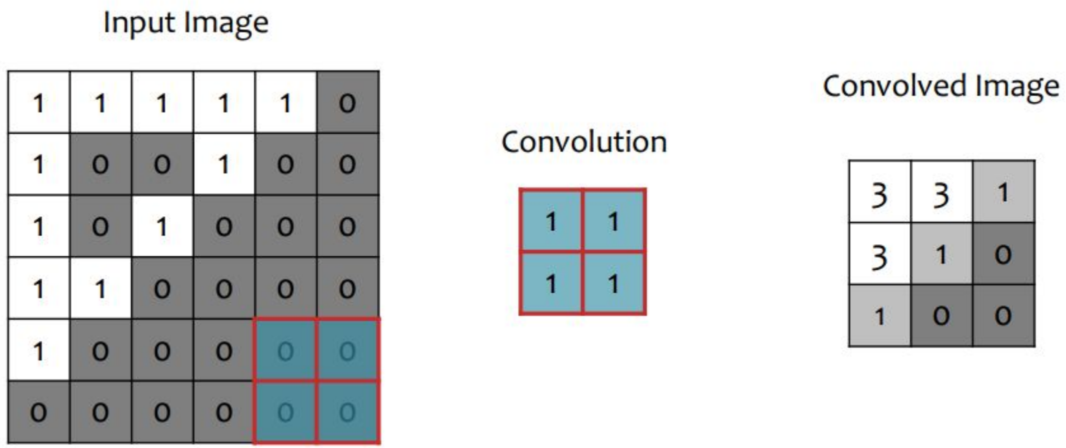


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

Downsampling

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



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

Convolutional neural network

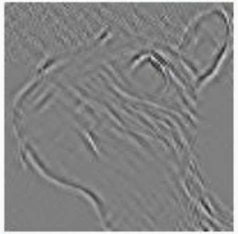
Operation	Kernel ω	Image result g(x,y)
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Ridge or edge detection	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	

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

Convolutional neural network

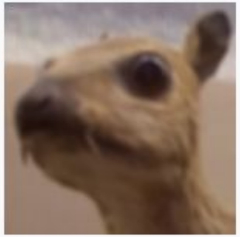
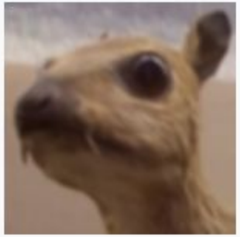
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur 3 × 3 (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

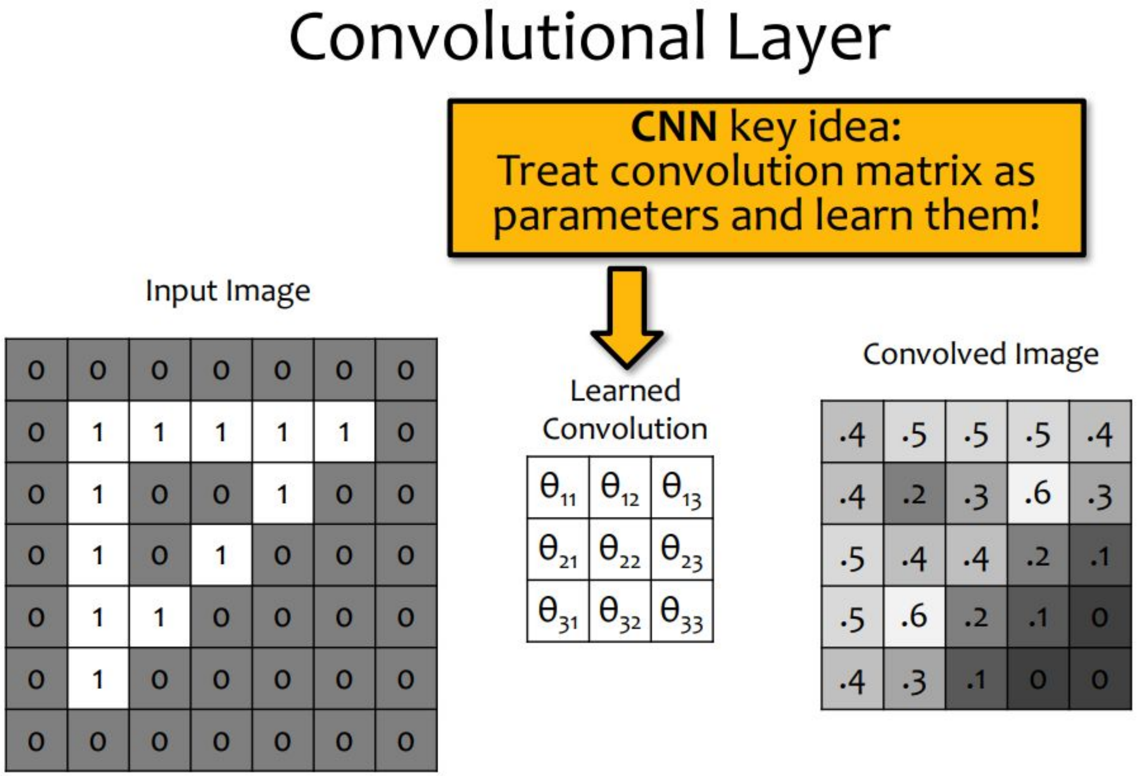
image source: https://en.wikipedia.org/wiki/Kernel_(image_processing)

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)

Convolutional neural network

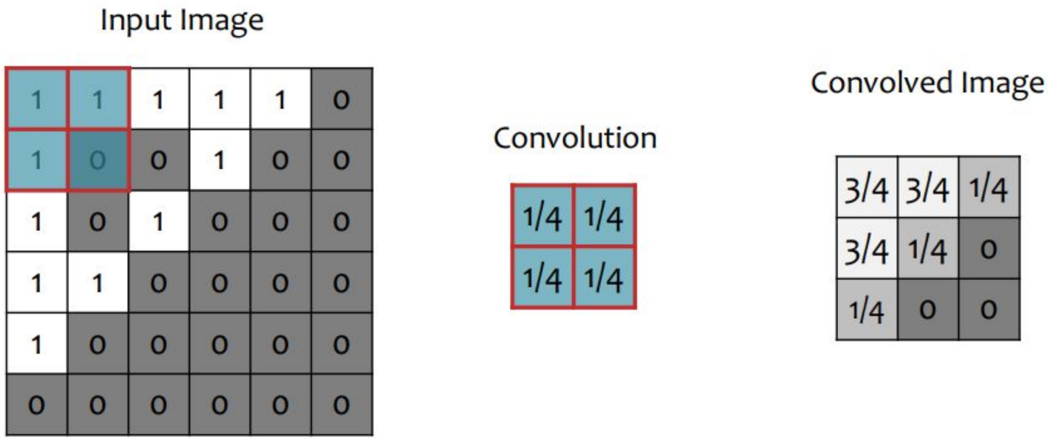


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

Convolutional neural network

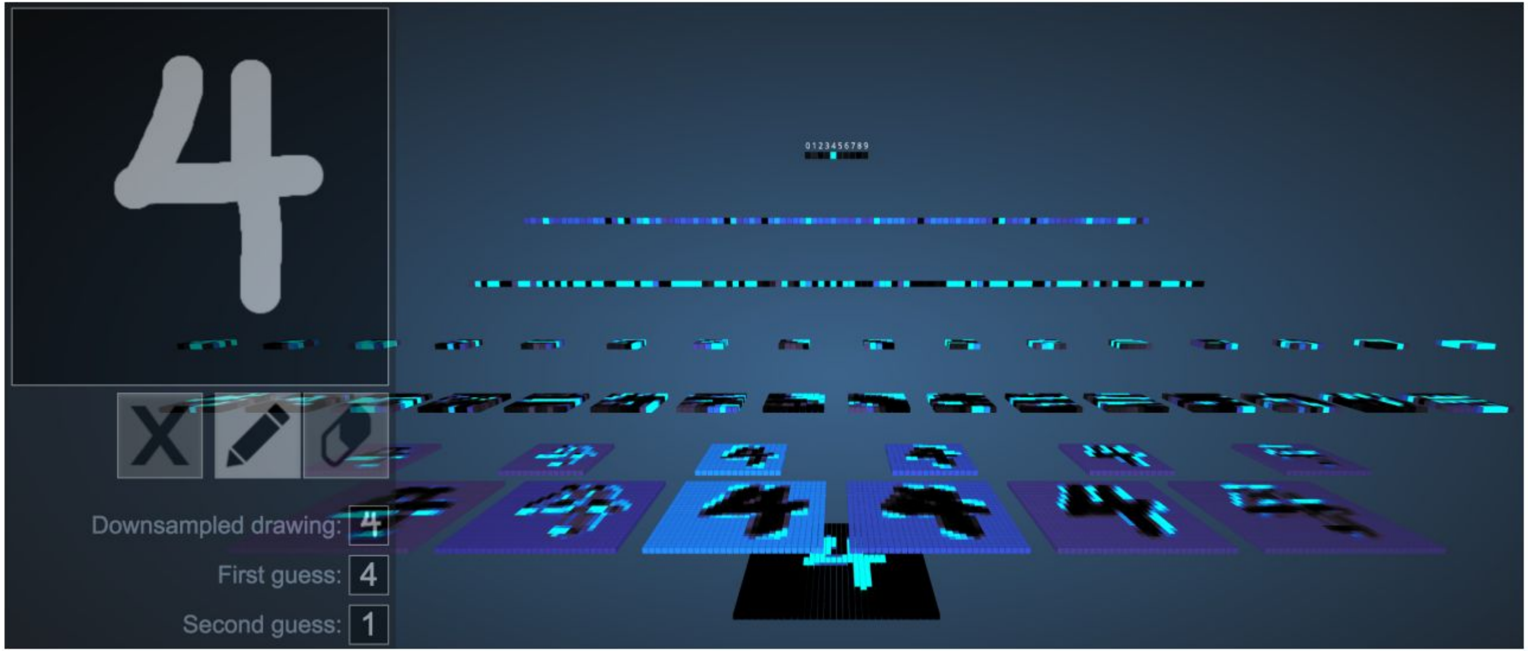
Downsampling by Averaging

- 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



Convolutional neural network

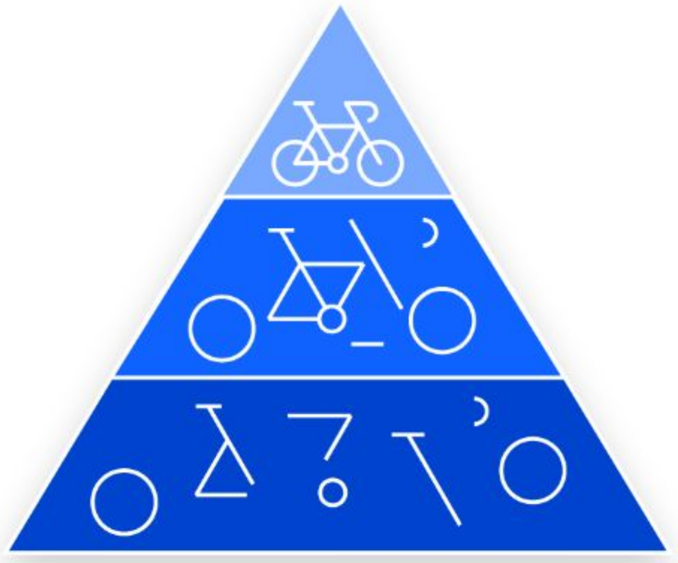
3D Visualization of CNN



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

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.

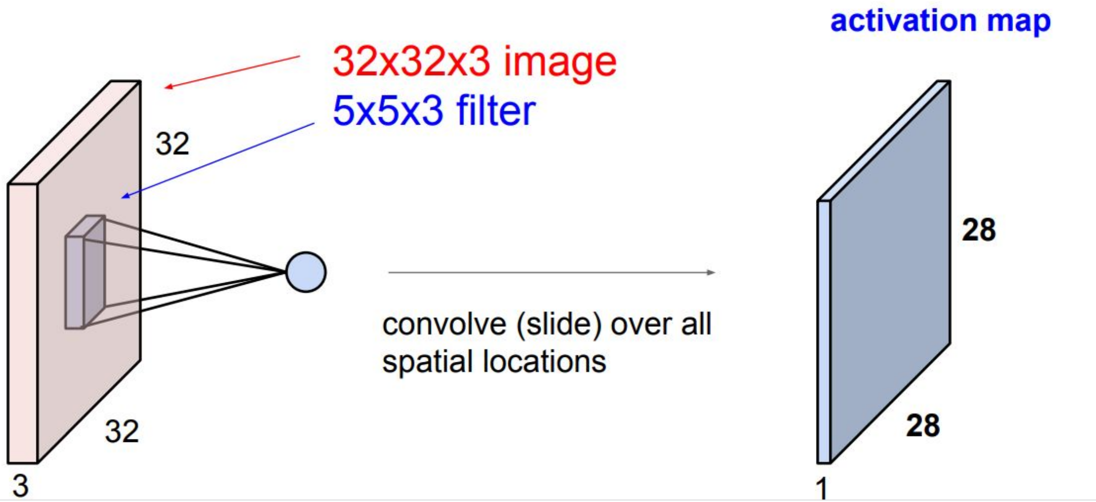


Source: https://www.ibm.com/topics/convolutional-neural-networks

Convolutional neural network

Convolution of a Color Image

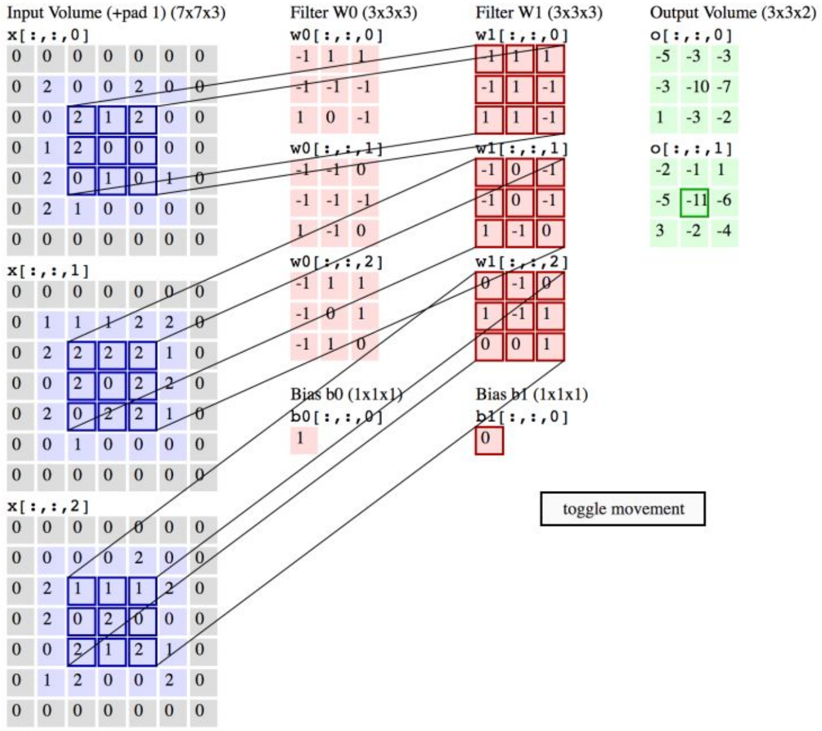
- Color images consist of 3 floats per pixel for RGB (red, green blue) color values
- Convolution must also be 3-dimensional



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

Convolutional neural network

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

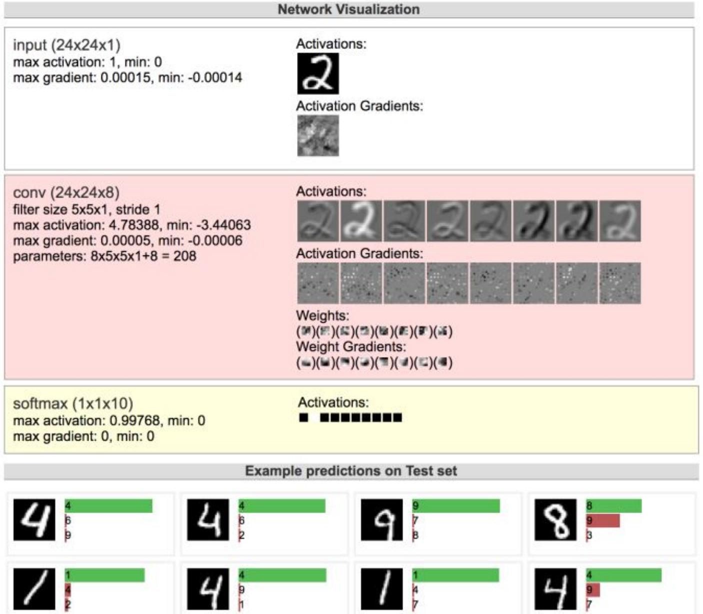


Slides taken from https://www.cs.cmu.edu/~mgormley/courses/10601-s17/slides/lecture21-cnn.pdf

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