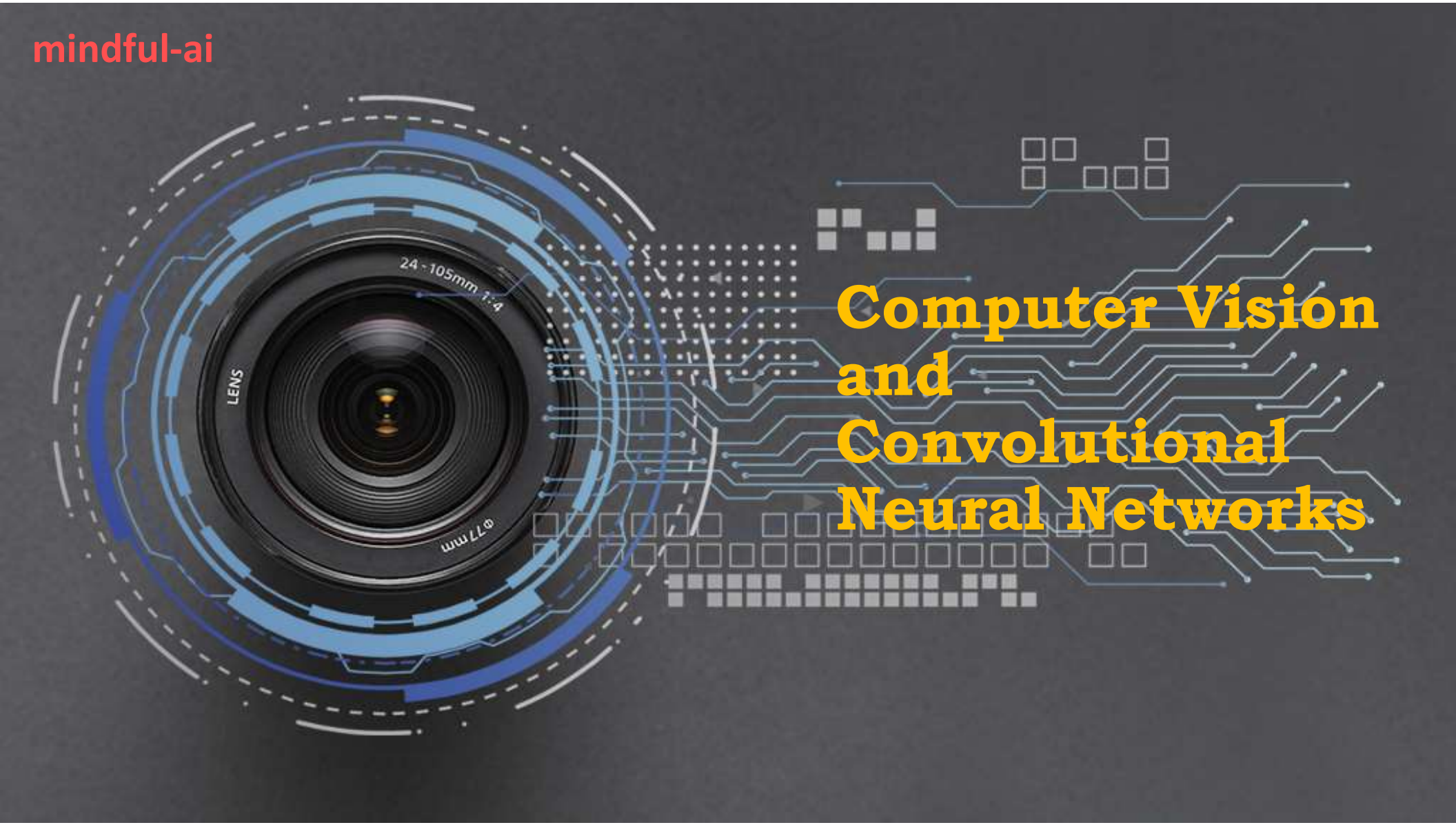
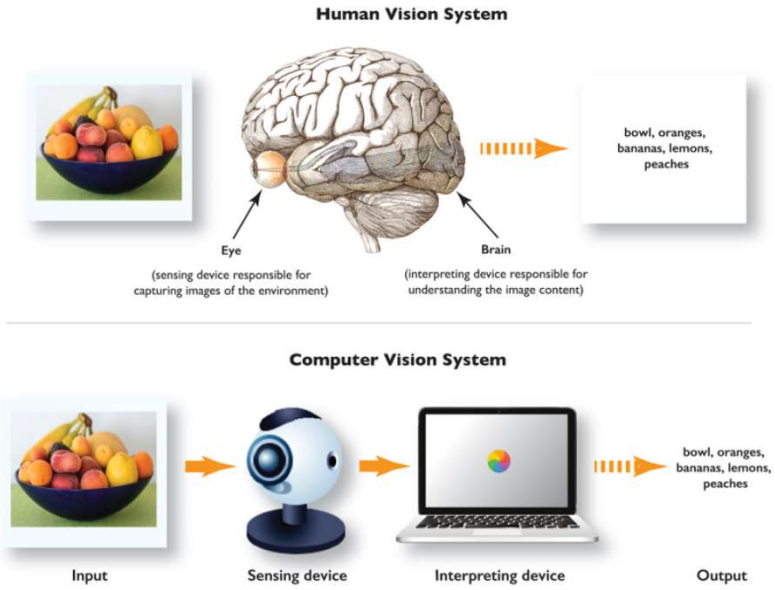
mindful-ai



Computer Vision and Convolutional Neural Networks

Computer Vision

- Computer Vision is a subfield of Deep Learning and Artificial Intelligence where humans teach computers to see and interpret the world around them.



A Brief History of Computer Vision

- Like all great things in the world of technology, computer vision started with a cat.
- Two Swedish scientists, Hubel and Wiesel, placed a cat in a restricting harness and an electrode in its visual cortex.
- The scientists showed the cat a series of images through a projector, hoping its brain cells in the visual cortex would start firing.

A Brief History of Computer Vision

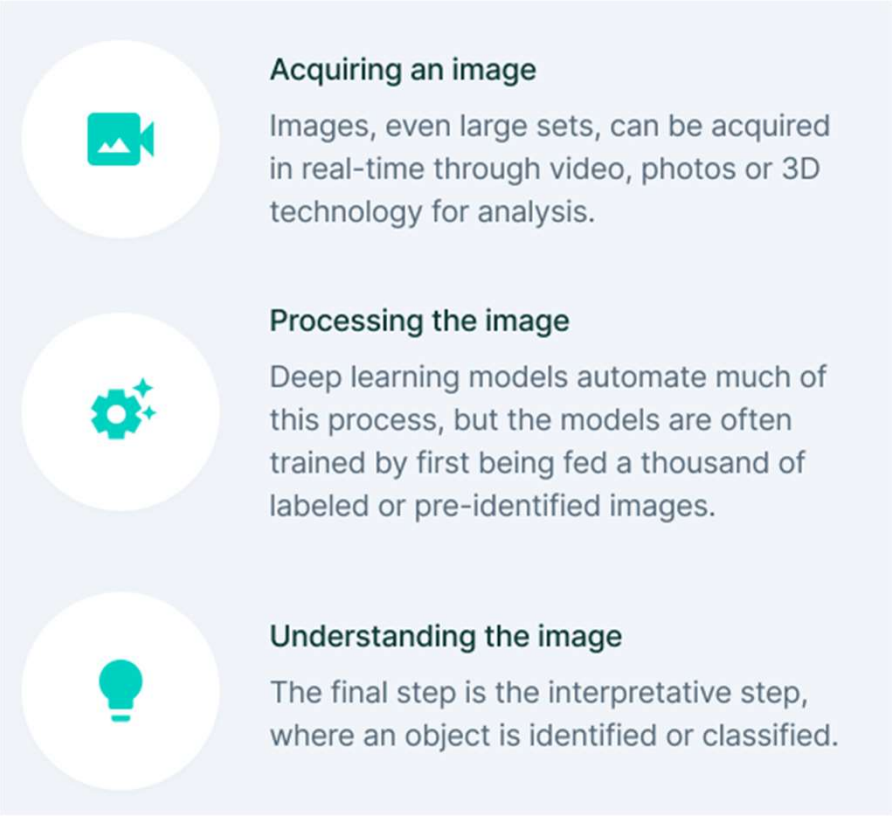
- With no avail with images, the eureka moment happened when a projector slide was removed, and a single horizontal line of light appeared on the wall
- **Neurons fired, emitting a crackling electrical noise**.
- The scientists had just realized that the early layers of the visual cortex respond to simple shapes, like lines and curves, much like those in the early layers of a deep neural network.
- They then used an oscilloscope to create these and observe the brain's reaction.

https://www.youtube.com/watch?v=Cw5PKV9Rj3o

Image Processing

- Digital Image Processing, or Image Processing, in short, is a subset of Computer Vision. It deals with enhancing and understanding images through various algorithms.
- More than just a subset, Image Processing forms the precursor of modern-day computer vision, overseeing the development of numerous rule-based and optimization-based algorithms that have led machine vision to what it is today.
- Image Processing may be defined as the task of performing a set of operations on an image based on data collected by algorithms to analyze and manipulate the contents of an image or the image data.

Practical Side of Computer Vision

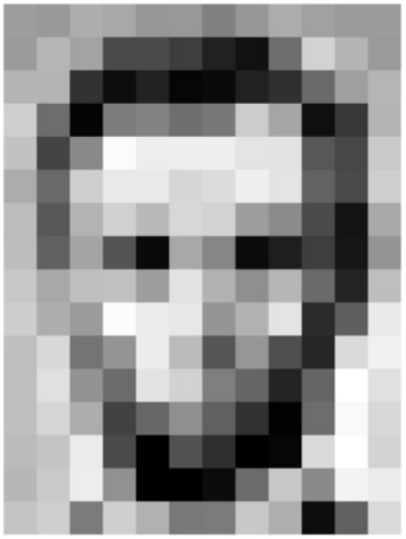


Pixels

- An image consists of several pixels, with a pixel being the smallest quanta in which the image can be divided into.
- Computers process images in the form of an array of pixels, where each pixel has a set of values, representing the presence and intensity of the three primary colors: red, green, and blue.
- All pixels come together to form a digital image.
- The digital image, thus, becomes a matrix, and Computer Vision becomes a study of matrices.

Pixels

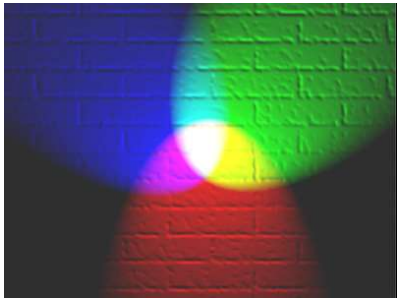
- The simplest computer vision algorithms use linear algebra to manipulate these matrices, complex applications involve operations like convolutions with learnable kernels and downsampling via pooling.



157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

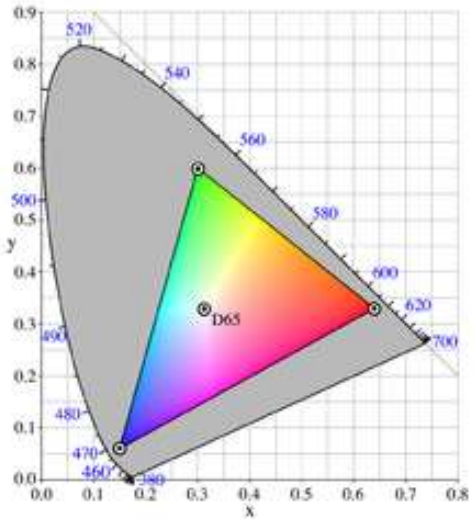
Color Space



- A color space is a specific organization of colors.
- A "color model" is an abstract mathematical model describing the way colors can be represented as tuples of numbers (e.g. triples in RGB or quadruples in CMYK)
- Example: RGB Color Model
  - The RGB color model is an additive color model[1] in which the red, green, and blue primary colors of light are added together in various ways to reproduce a broad array of colors.
  - The main purpose of the RGB color model is for the sensing, representation, and display of images in electronic systems, such as televisions and computers, though it has also been used in conventional photography.
  - It is Device Independent

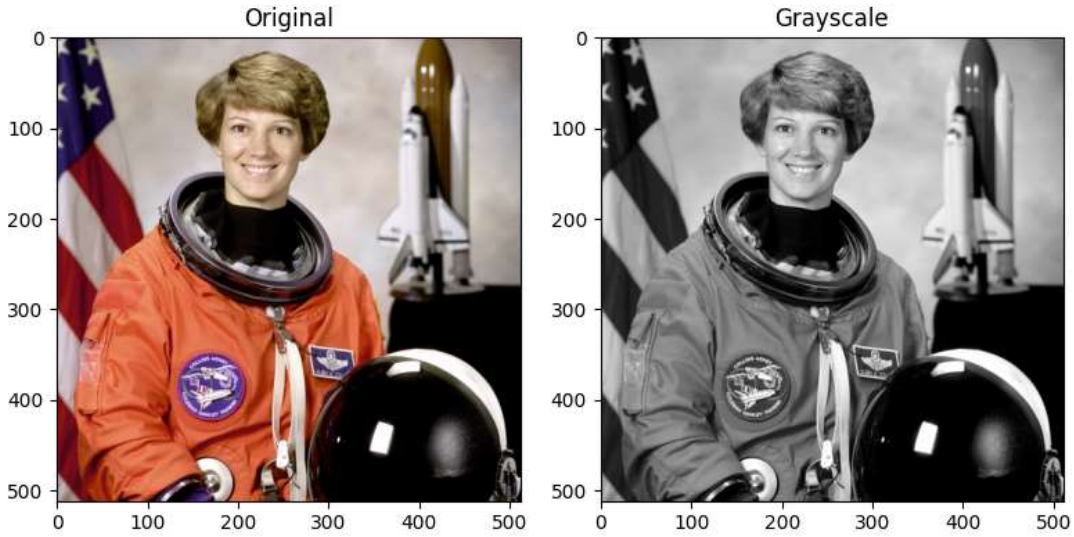
Color Triangle

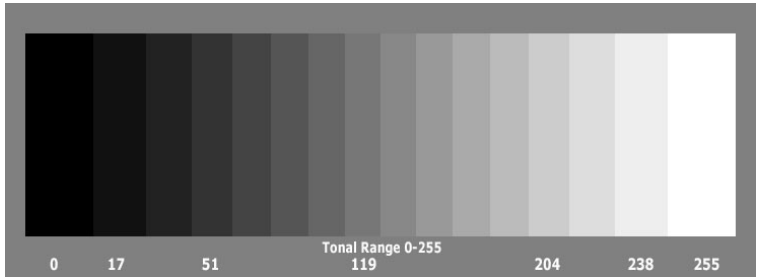
- The choice of primary colors is related to the physiology of the human eye; good primaries are stimuli that maximize the difference between the responses of the cone cells of the human retina to light of different wavelengths, and that thereby make a large color triangle

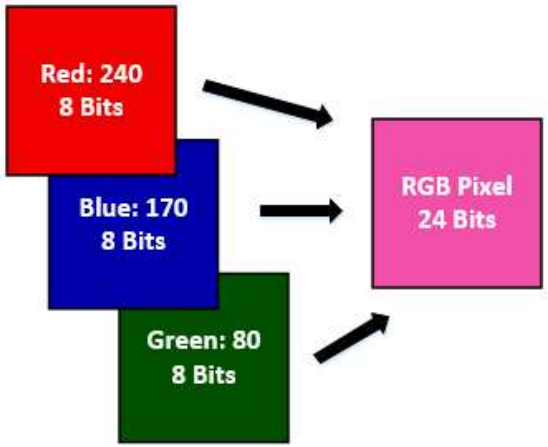


A set of primary colors, such as the sRGB primaries, define a color triangle; only colors within this triangle can be reproduced by mixing the primary colors. Colors outside the color triangle are therefore shown here as gray.

RGB Vs Grey Scale





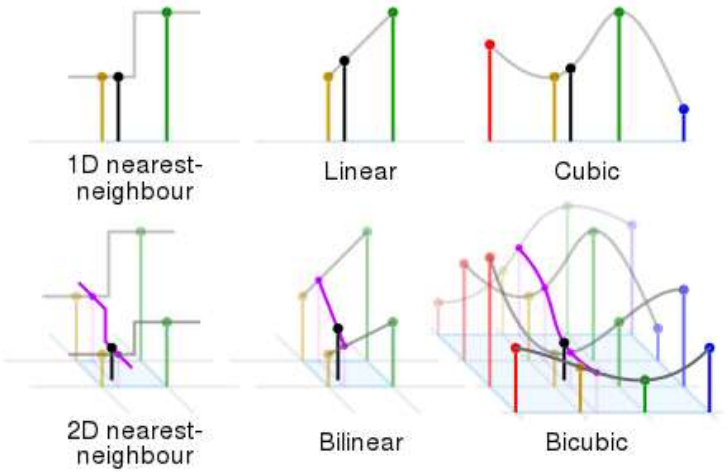


Basic Operations

- Resizing
- Color space conversion
- Erosion
- Dilation
- Filtering
- Segmentation

Resizing

- Nearest Neighbour Interpolation
- Bi-Linear Interpolation
- Bi-Cubic Interpolation
- Box Sampling



300x200px image
9KB

200x200px image
7.1KB (21% smaller)

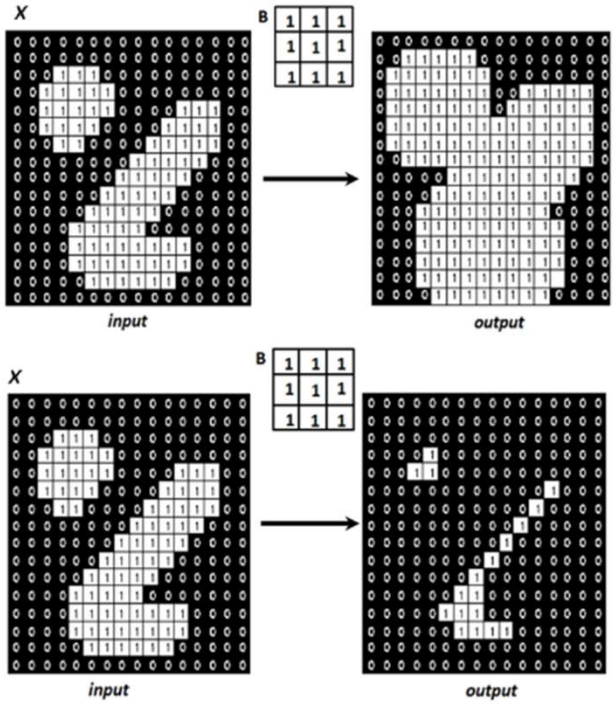
Box sampling considers every pixel in the downscaled image as a box in the original. And its color is the average of the colors inside the box.

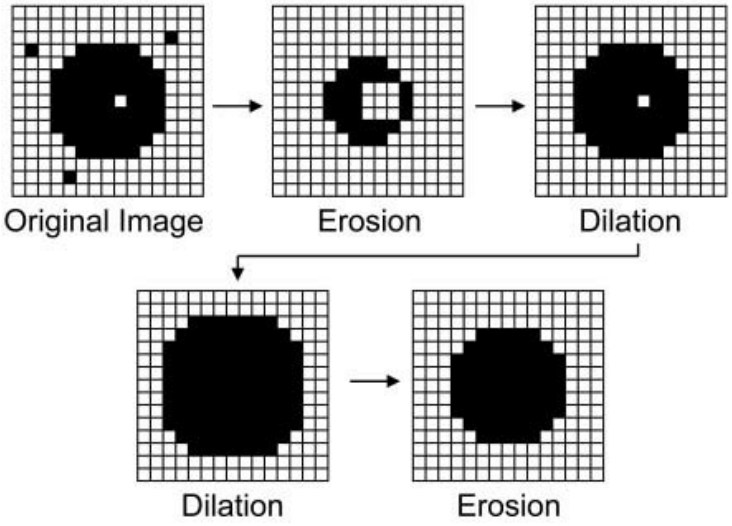
Color Space Conversion

- Color space conversion is the translation of the representation of a color from one basis to another.
- This typically occurs in the context of converting an image that is represented in one color space to another color space, the goal being to make the translated image look as similar as possible to the original.

Erosion and Dilation

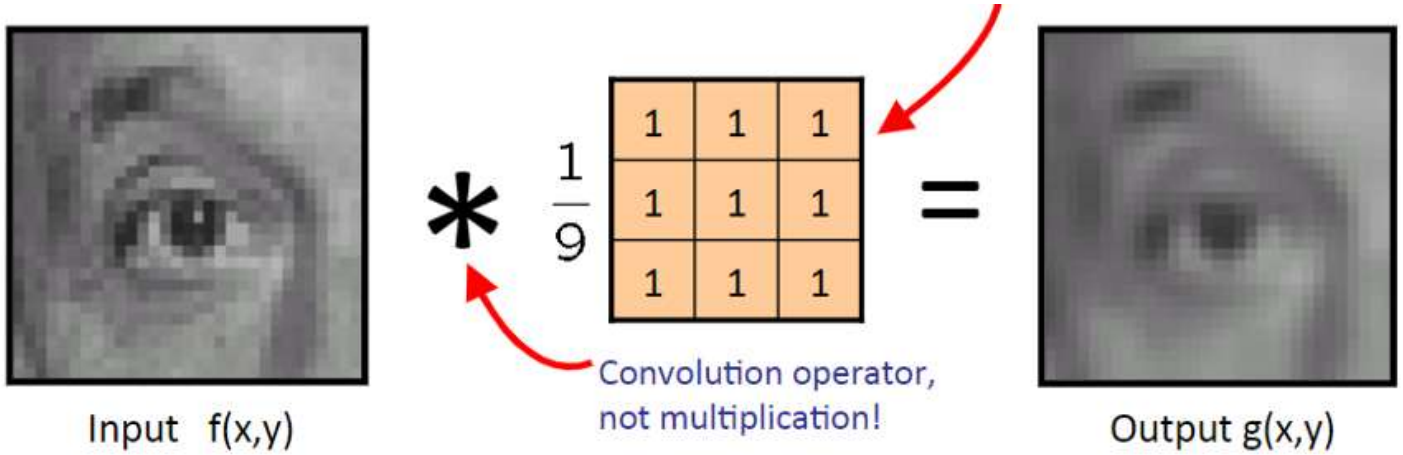
- Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries





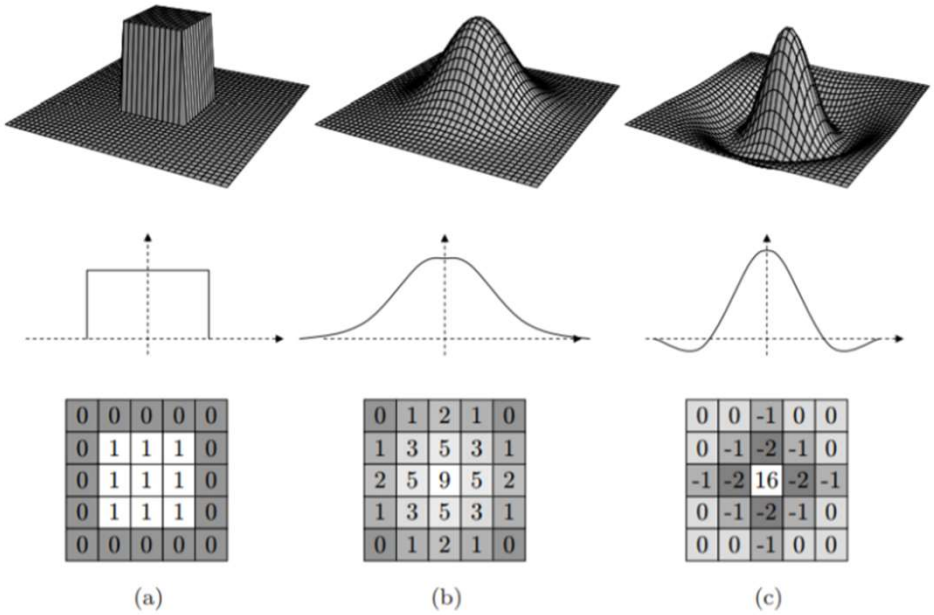
Averaging Filters

- Average filters take the mean value of the pixels in a neighborhood, which is defined by the size of a mask (m-columns and n-rows). It is important to divide by the sum of the values in the neighborhood to normalize output values.



Gaussian Filter (Smoothening)

- Gaussian Filter blurs an image with a bell shape represented by its normal distribution. Used for smoothening images



0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

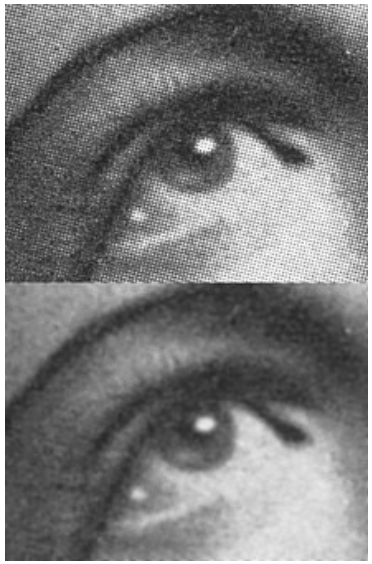
(a)

0	1	2	1	0
1	3	5	3	1
2	5	9	5	2
1	3	5	3	1
0	1	2	1	0

(b)

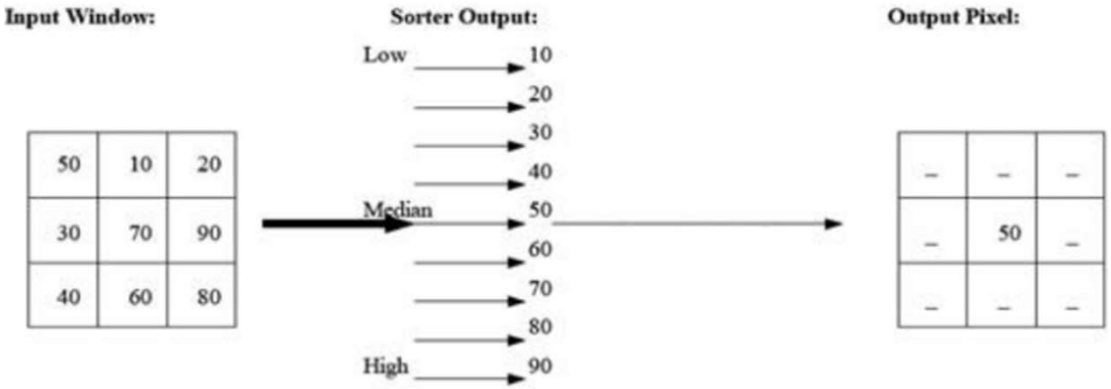
0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

(c)



Median Filter

- ***Median filters*** are the most popular because of the ability to reduce impulse noise aka salt-and-pepper noise. In order to perform median filtering at a point of an image, we first sort the values of the pixels in the neighborhood, determine the median and then assign that value to the corresponding pixel in the filtered image.



median filtering algorithm

Median Filter

IMAGE WITH SALT AND PEPPER NOISE

IMAGE AFTER MEDIAN FILTERING. WINDOW SIZE 3x3

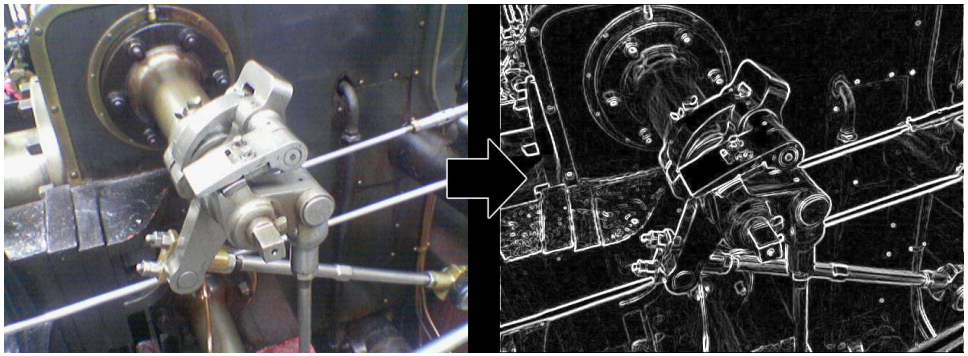
IMAGE AFTER MEDIAN FILTERING. WINDOW SIZE 5x5

IMAGE AFTER MEDIAN FILTERING. WINDOW SIZE 7x7

Sobel Filter

- This sharpening filter is using a coefficient to smooth the output image while enhancing edges. It uses a weight value of 2 in the center. Sobel filter emphasizes the edges.

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$



Laplacian Filter

- Laplacian is a derivative operator; its uses highlight gray level discontinuities in an image and try to deemphasize regions with slowly varying gray levels.
- This operation in result produces such images which have grayish edge lines and other discontinuities on a dark background. This produces inward and outward edges in an image

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

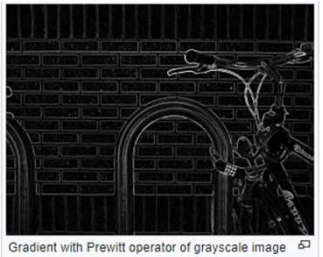
−1	2	−1
2	−4	2
−1	2	−1

Prewitt Filter

- The Prewitt operator is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function.
- At each point in the image, the result of the Prewitt operator is either the corresponding gradient vector or the norm of this vector.

$$\mathbf{G_x} = \begin{bmatrix} +1 & 0 & -1 \\ +1 & 0 & -1 \\ +1 & 0 & -1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G_y} = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} * \mathbf{A}$$

Grayscale image of a brick wall & a bike rack



Gradient with Prewitt operator of grayscale image of a brick wall & a bike rack

Interactive Example

- Let’s explore and interactive example:
  - **setosa.io/ev/image-kernels/**

Convolution

- In the context of CNN's, these filters are referred to as **Convolution Kernels**
- The process of passing them over an image is known as **convolution**

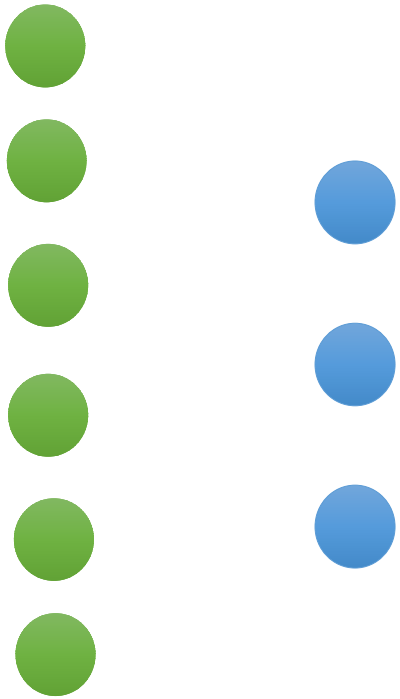
CNN

- Recall that running an ANN for the MNIST data set resulted in a network with relatively goof accuracy
- However, there are some issues with always using ANN models for the image data
- ANNs
  - Large amounts of paramters (over 100,000 for tiny 28x28 images)
  - We lose all the 2D information due to flattening
  - We only work on very similar well cantered images

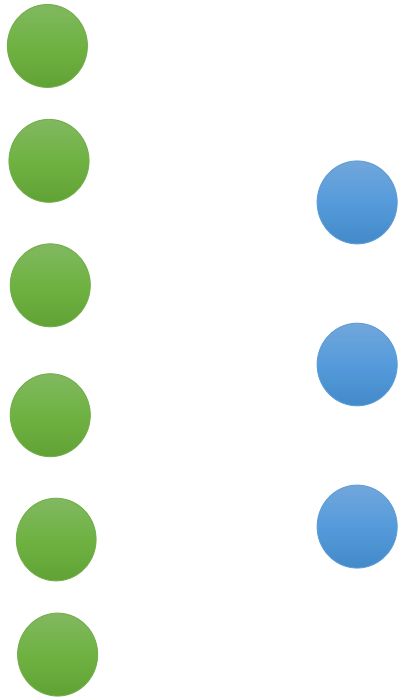
CNN

- CNN uses convolutional layers to help alleviate these issues
- A convolutional layer is created when we apply multiple image filters to the input images
- The layer will then be trained to figure out the best filter weight values
- The CNN also helps reduce paramters by focussing on local connectivity
- Not all neurons will be fully connected
- Instead, neurons are only connected to a subset of local neuraon in the next later (these end up being the filters)

1D Analysis

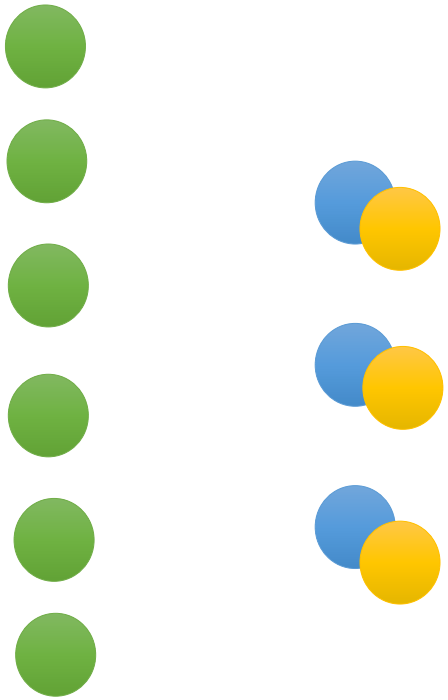


Fully Connected: Lot’s of parameters



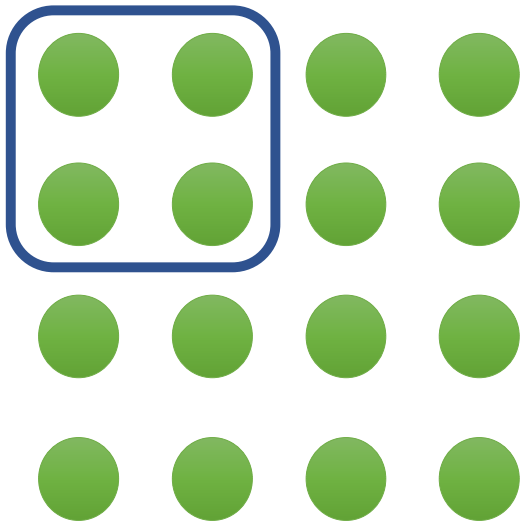
Locally Connected: Reduced parameters

1D Analysis

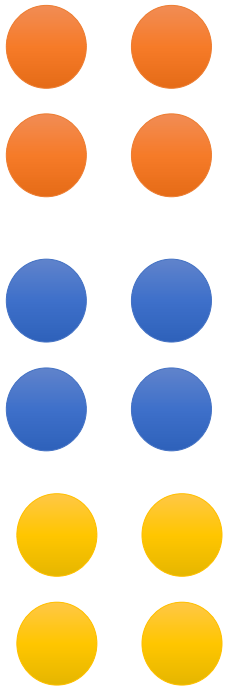


Locally Connected: Reduced parameters

2D Analysis

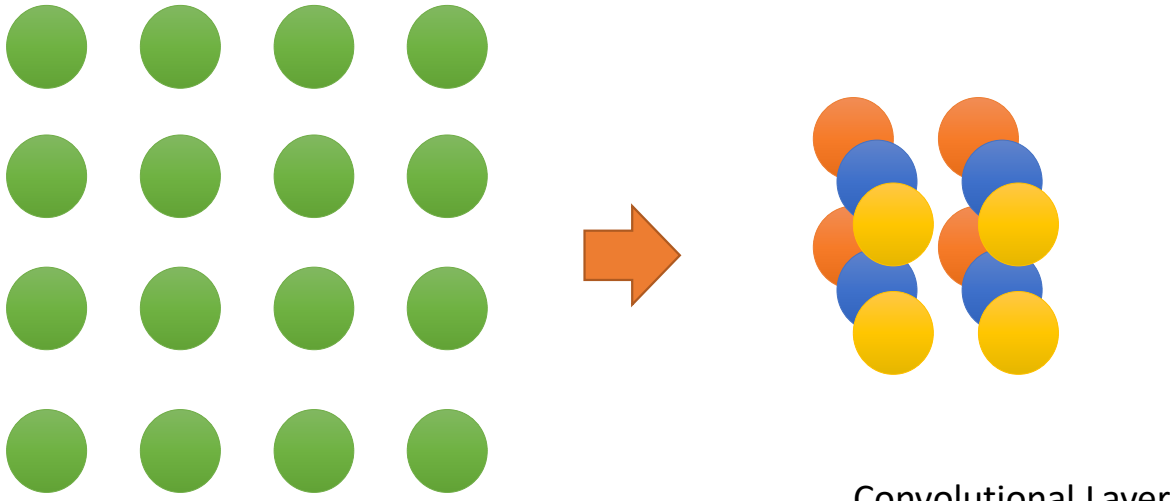


Input Image



Essentially a kernel

2D Analysis



Input Image

Convolutional Layer

3D Analysis

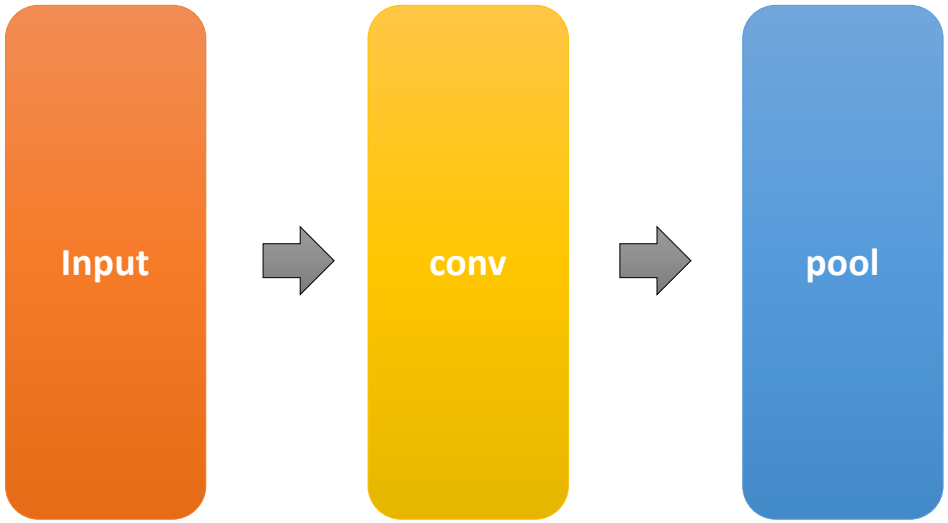
- But that was just for 2D gray scale images
- What about colour images?
- Colour images can be thought of as 3d Tensors consisting of Red, Green and Blue colour channels
- We have already see that additive colour mixing allows us to represent a wide variety of colours by simply combining different amount of R, G and B
- This means that a shape of the colour array has 3 dimensions: height, width and color channels
- So the filter we need to use should have 3 dimensions

Pooling Layers

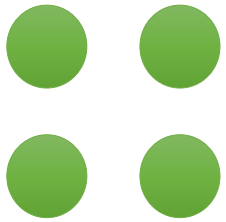
- Even with local connectivity, when dealing with color images and possibly 10s and 100s of filters we will have a large about of parameters
- We can use pooling layers to reduce this

Pooling Layers

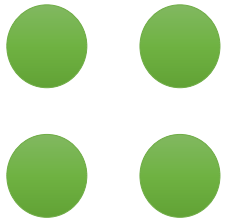
- Pooling Layers accept convolutional layers as inputs



Pooling Layer



Max Pooling

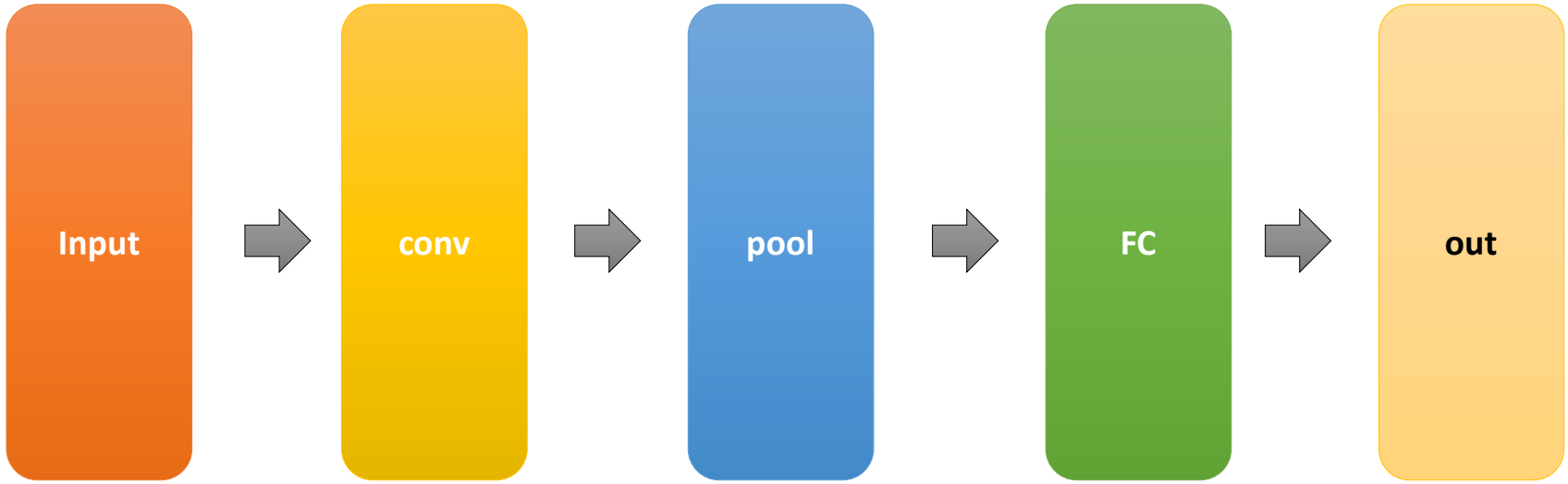


Average Pooling

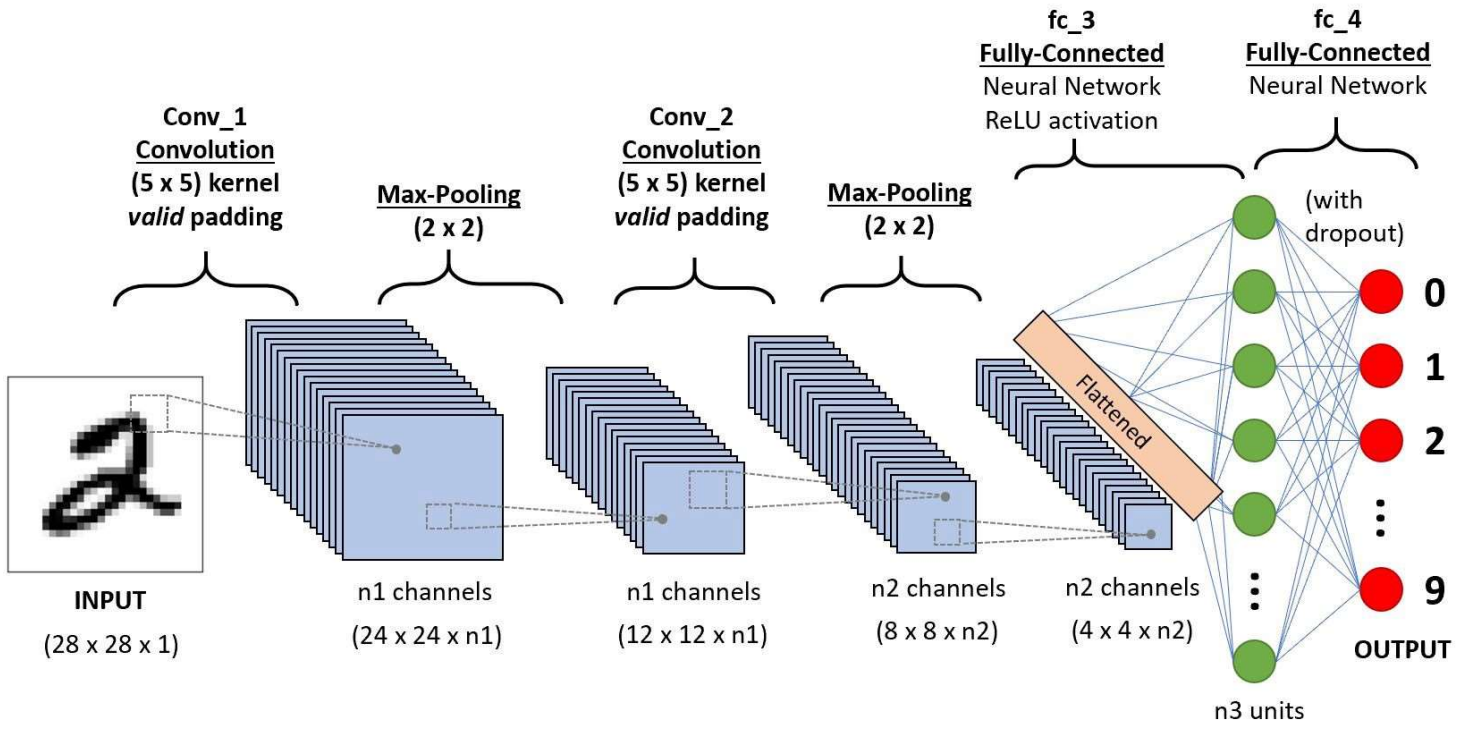
We loose quite a lot of information. But, the general trend remain the same

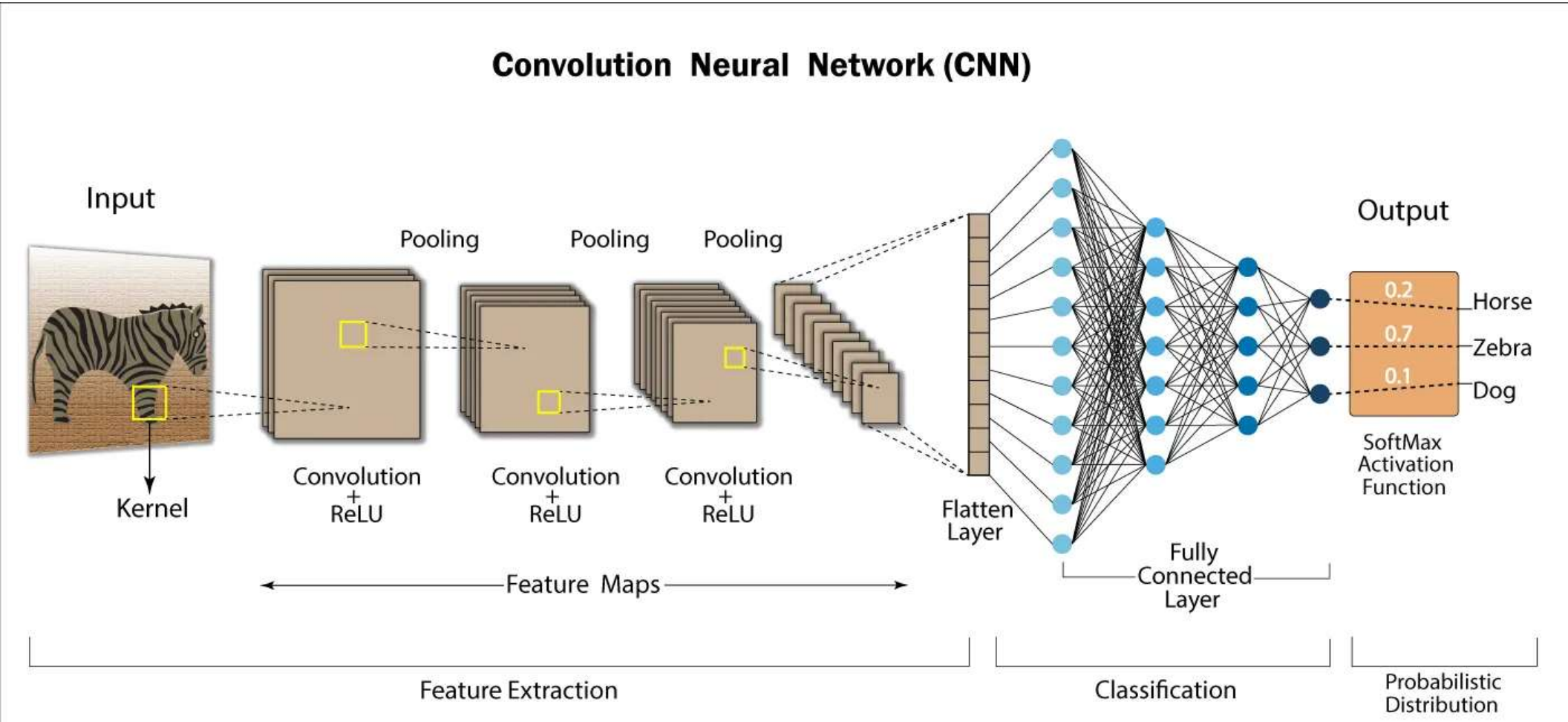
Convolutional Networks

- There is nearly no wrong way. Try to use any combination of convolutional layer, pooling layers
- But finally connect with a fully connected layer and also an output layer

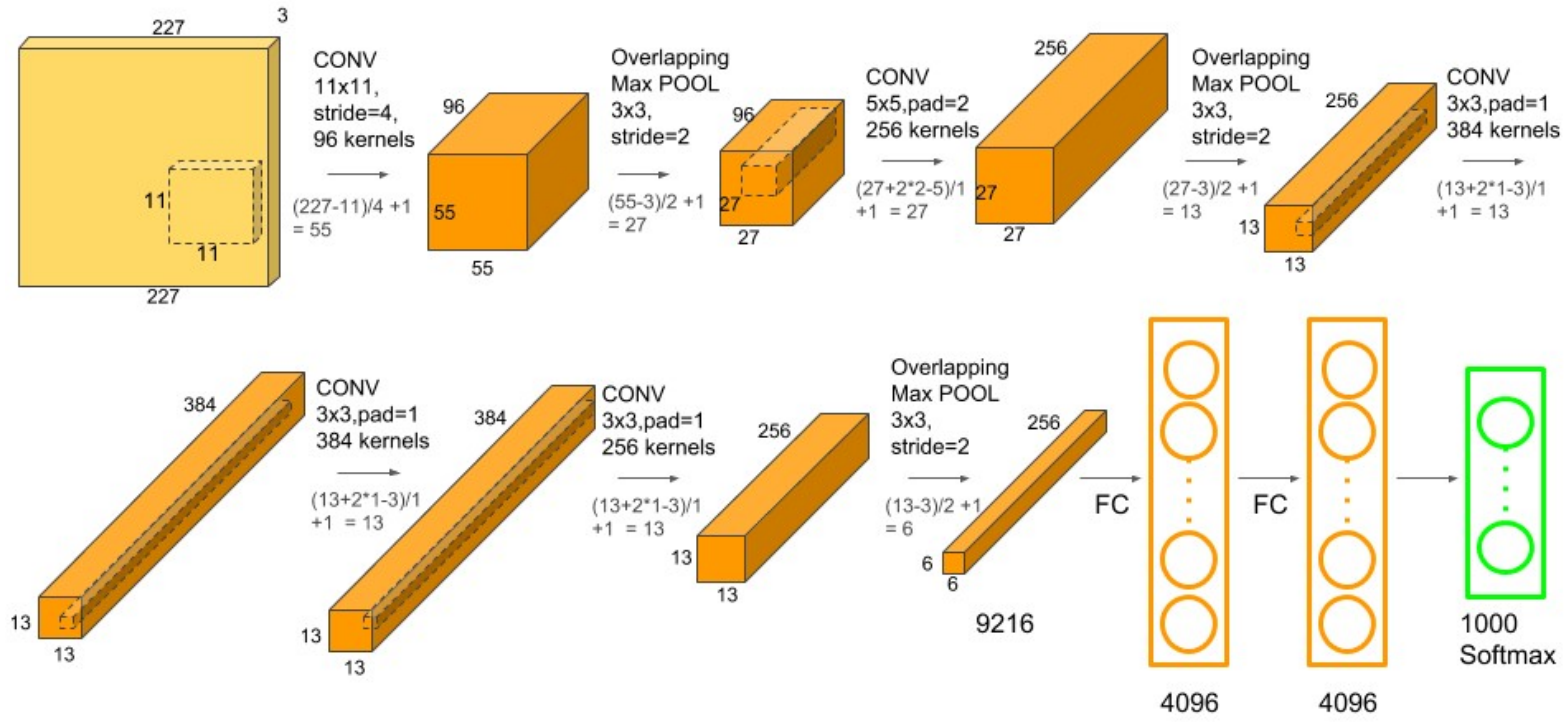


Typical CNN Architecture




Convolution Neural Network (CNN)
Input
Pooling
Pooling
Pooling
Output
0.2
Horse
0.7
Zebra
0.1
Dog
Kernel
Convolution + ReLU
Convolution + ReLU
Convolution + ReLU
Flatten Layer
SoftMax Activation Function
Fully Connected Layer
Feature Maps
Feature Extraction
Classification
Probabilistic Distribution

Popular CNN Architectures: AlexNet



Famous CNN Architectures