Building Features from Image Data

REPRESENTING IMAGES AS FEATURES FOR MACHINE LEARNING

Janani Ravi
CO-FOUNDER, LOONYCORN

www.loonycorn.com

Overview

Represent images as matrices

Model color information using channels

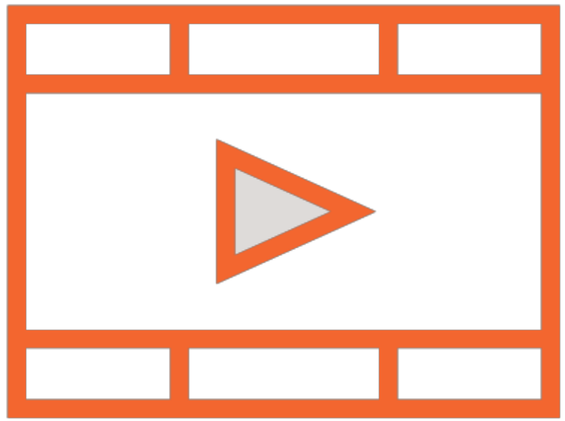
Working with grayscale and RGB images

Understand the need for pre-processing of images

Implement common operations on images

Prerequisites and Course Outline

Prerequisites



Basic Python programming

Basic understanding of ML algorithms

Understanding of image processing in ML will be helpful for some topics

Course Outline

Images as features

Detecting features and text

Dimensionality reduction with images

Working with Images in Machine Learning

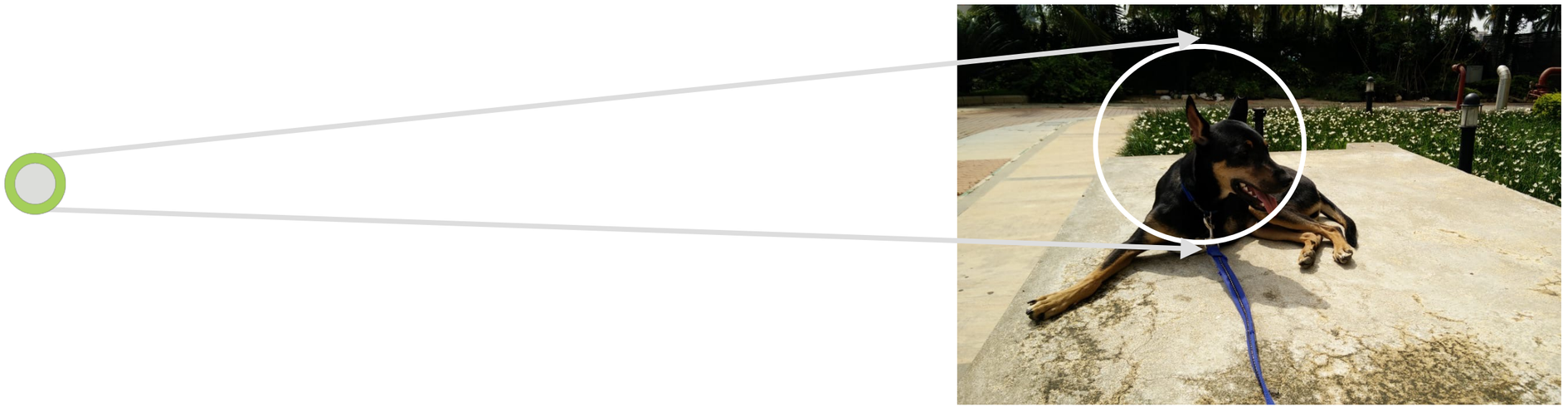
Viewing an Image

All neurons in the eye don’t see the entire image

Viewing an Image

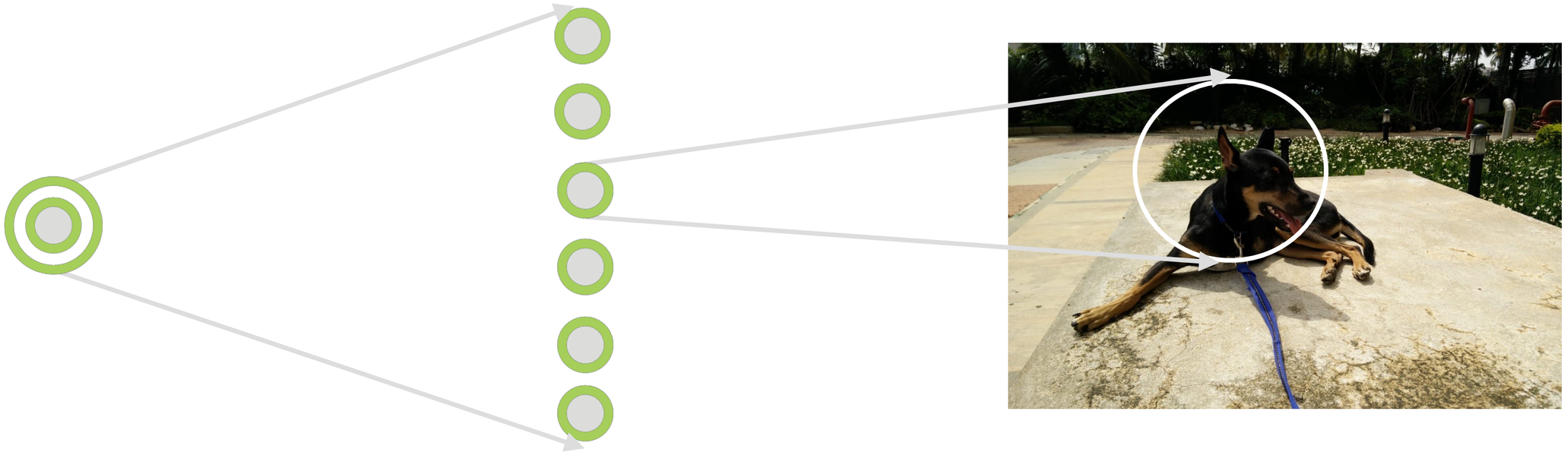
Each neuron has its own local receptive field

Viewing an Image



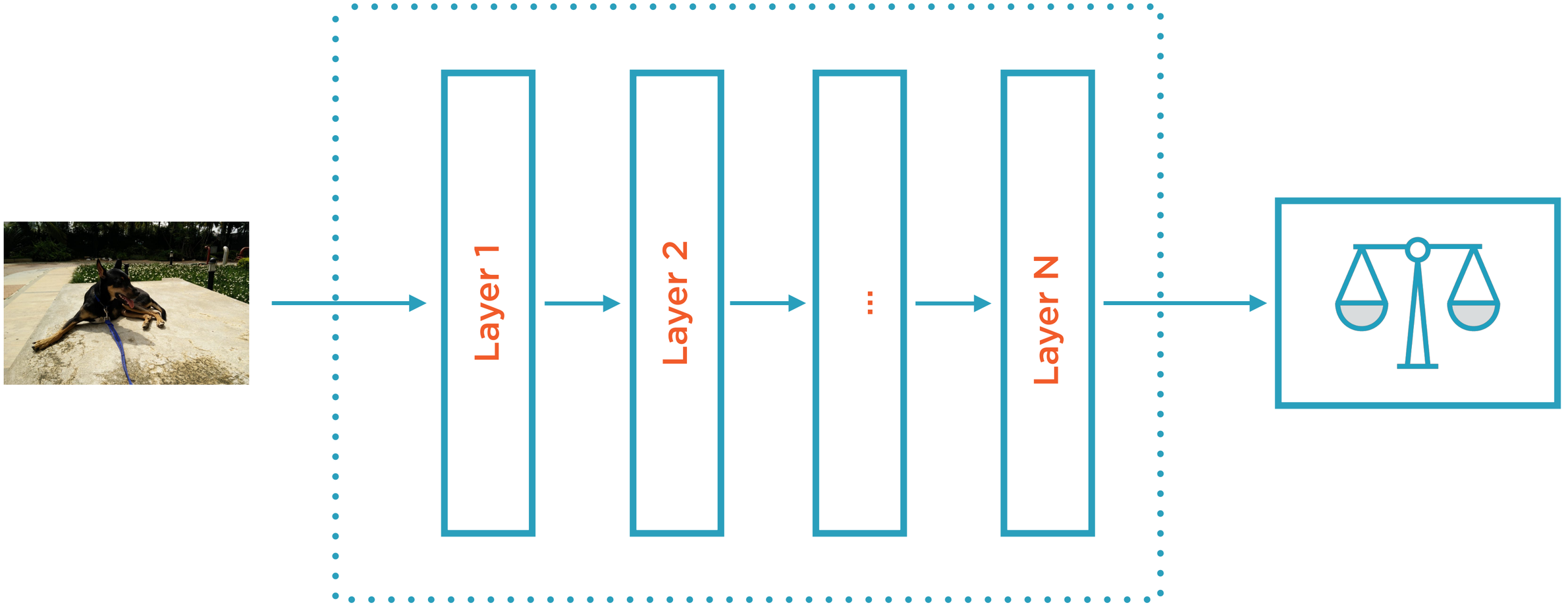
It reacts only to visual stimuli located in its receptive field

Viewing an Image



Some neurons react to more complex patterns that are combinations of lower level patterns

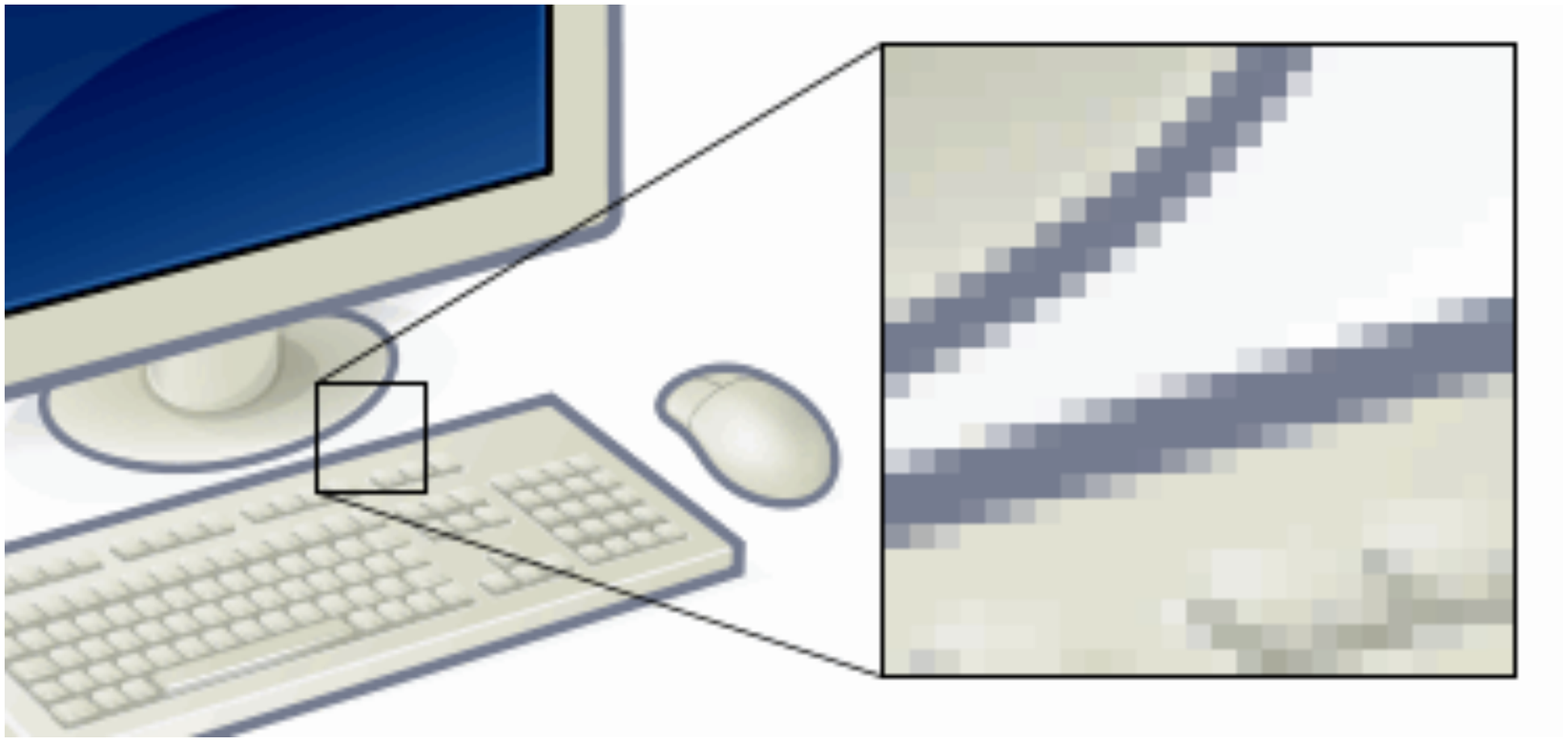
Neural Networks



Sounds like a classic neural network problem

Convolutional Neural Networks mimic the visual cortex and perceive images in 2 dimensions

Pixels in Images



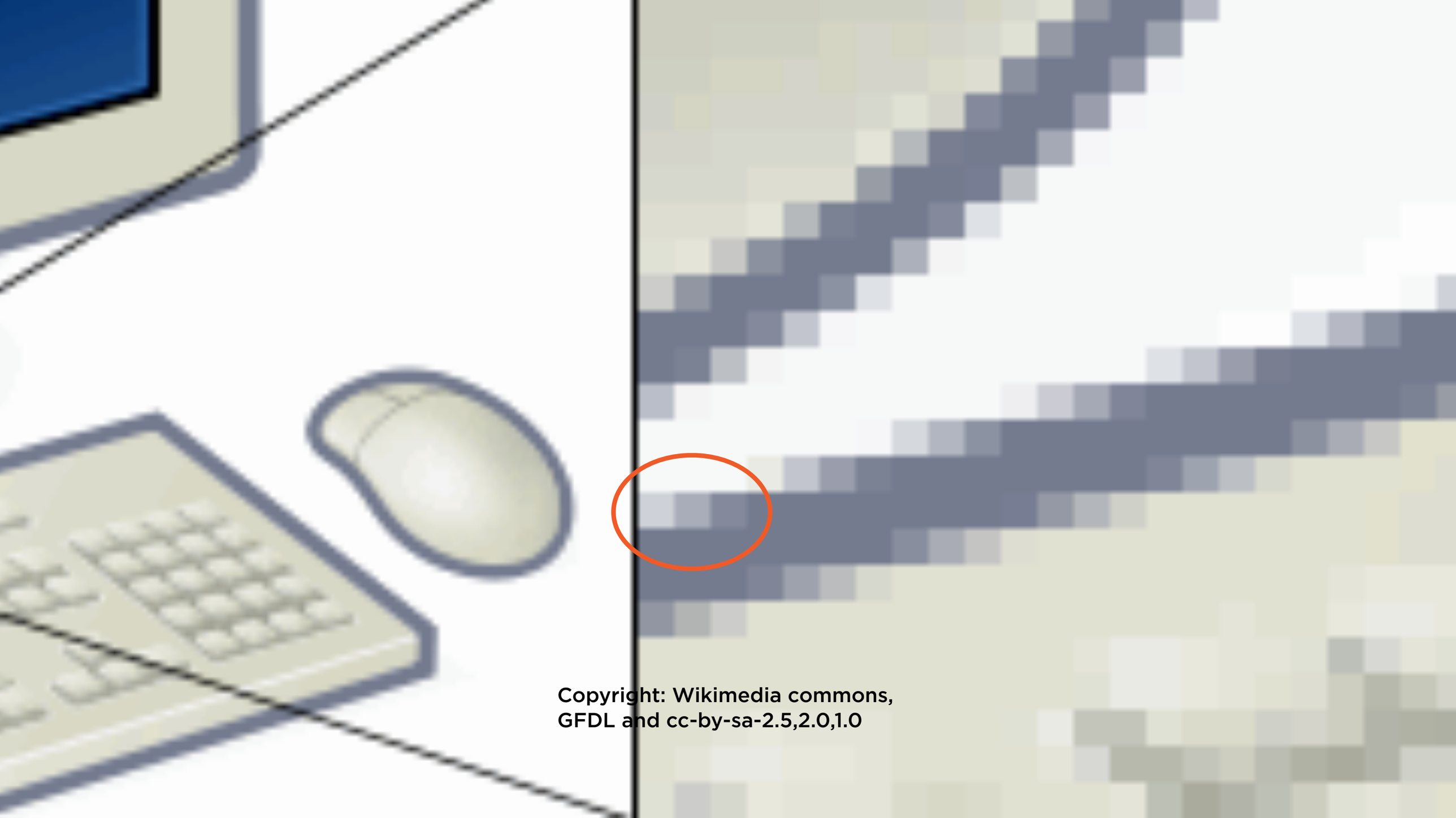
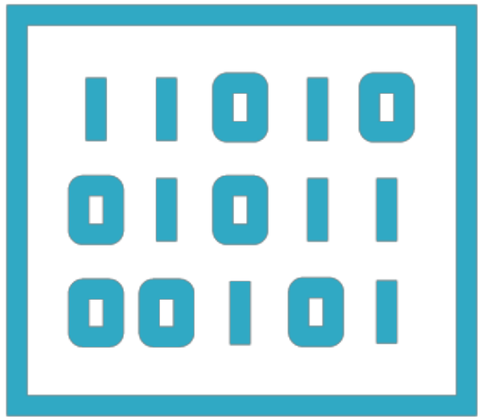
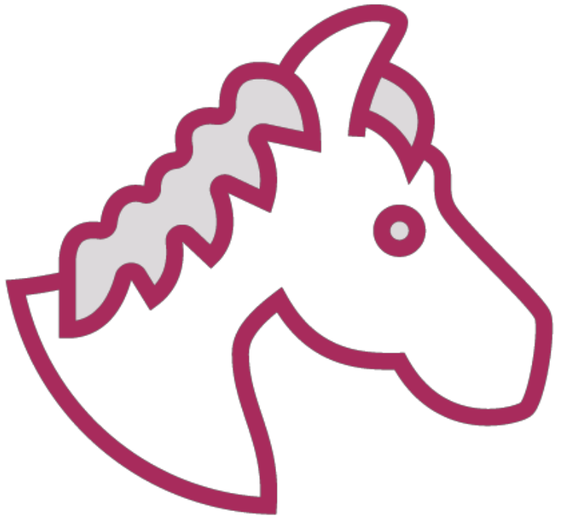


Image Recognition



Images represented as pixels

Identify edges, colors, shapes

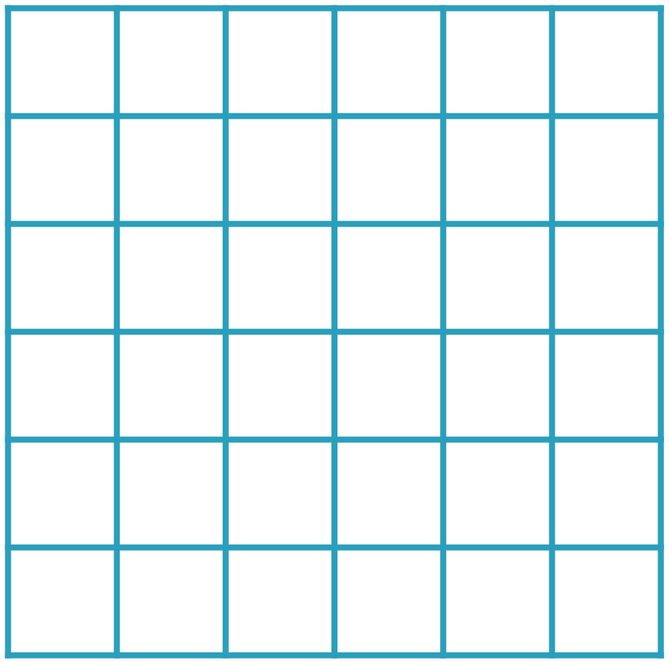


A photo of a horse

Neural networks, specifically convolutional neural networks (CNNs) work well for hard image recognition tasks

Images as Matrices

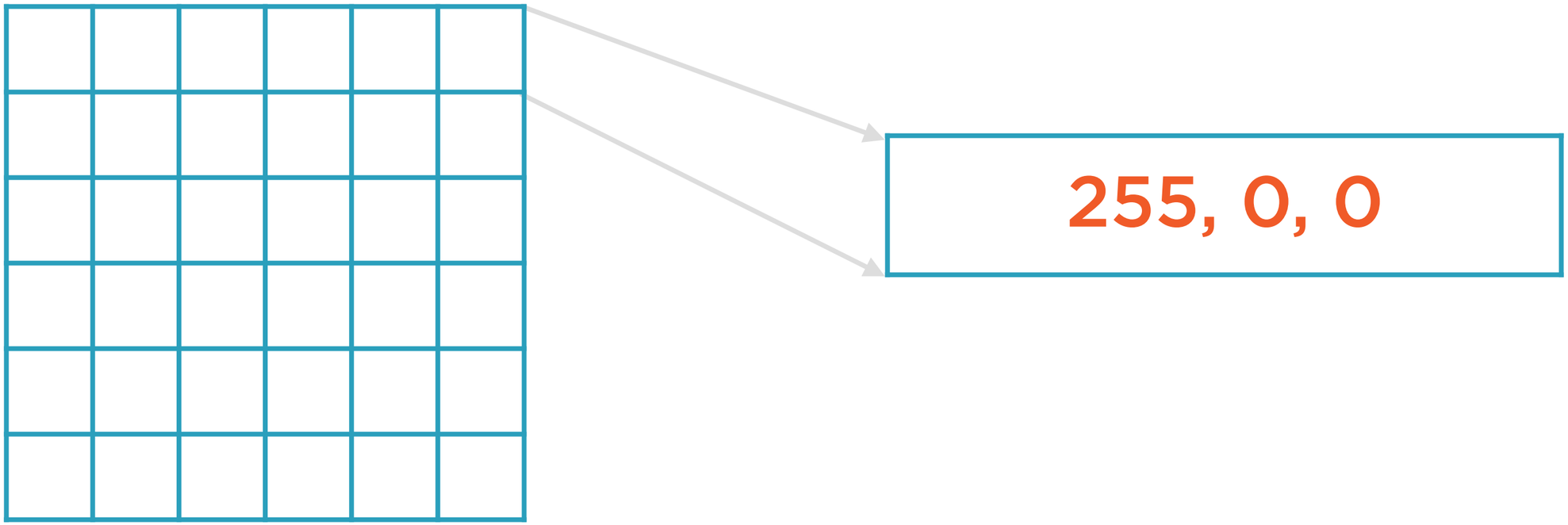
RGB Images



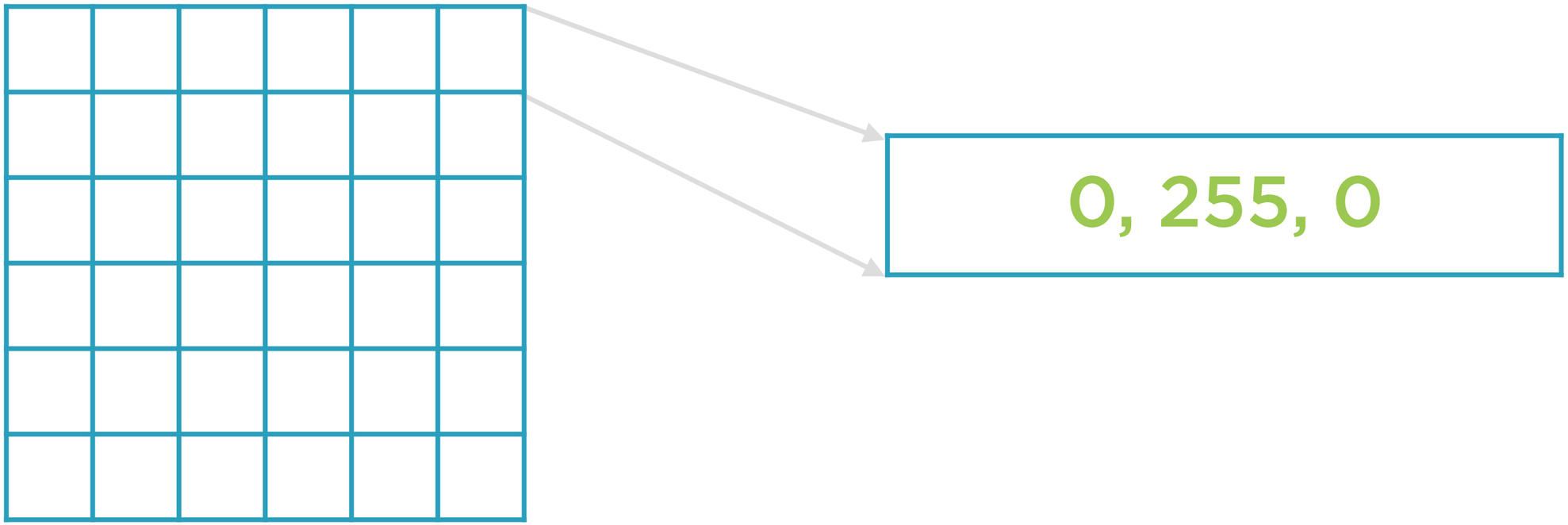
RGB values are for color images

R, G, B: 0-255

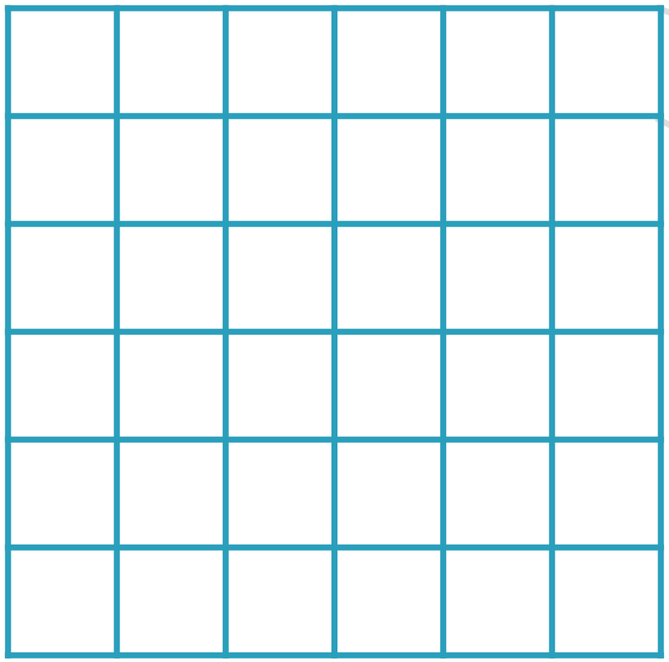
RGB Images



RGB Images



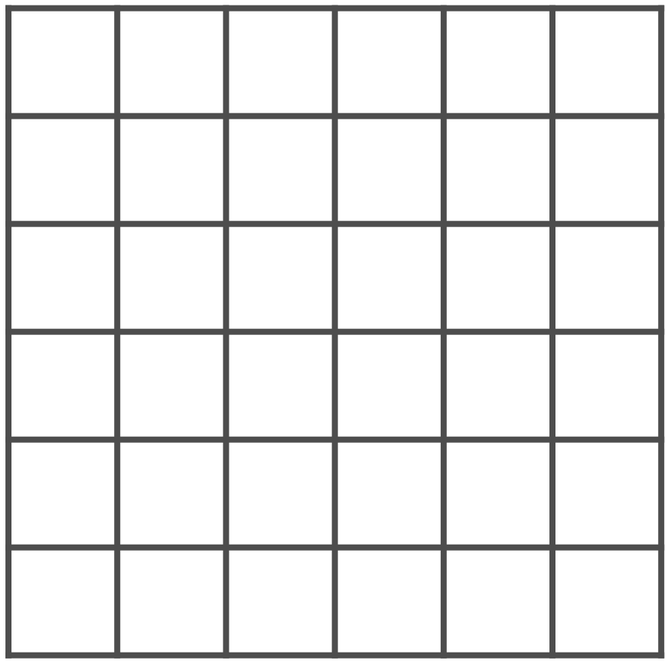
RGB Images

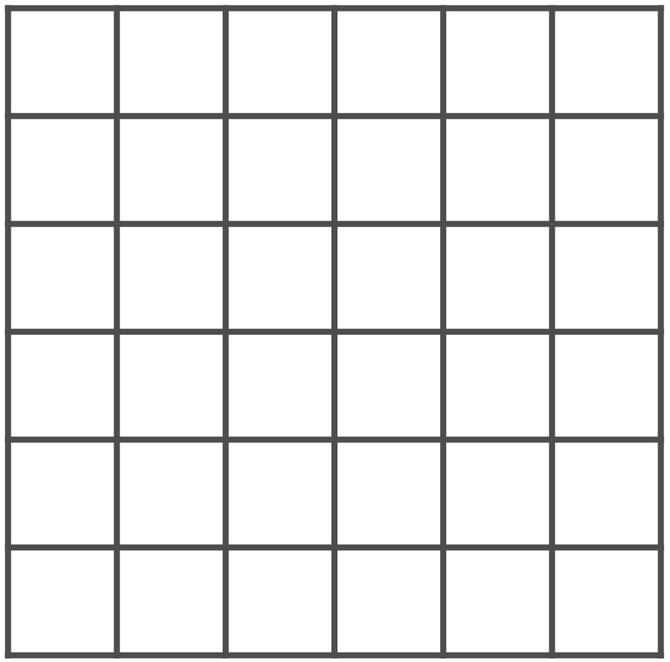


0, 0, 255

3 values to represent color, 3 channels

Grayscale Images



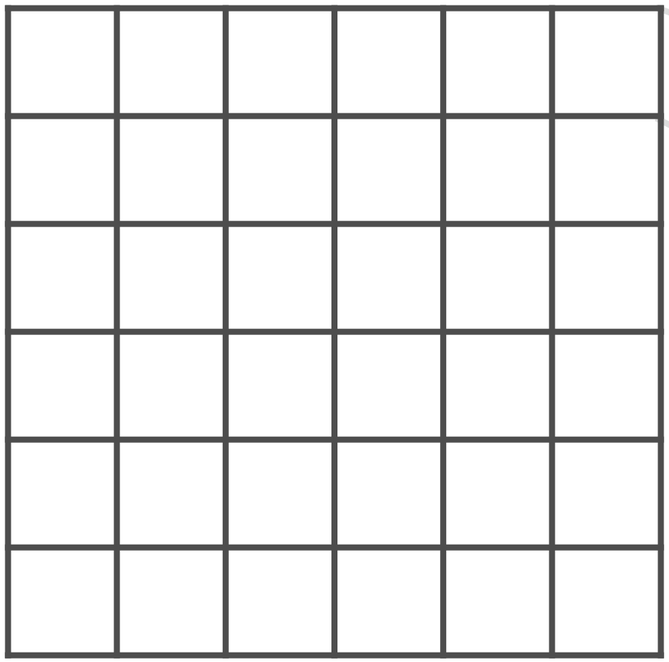


Grayscale Images

Each pixel represents only intensity information

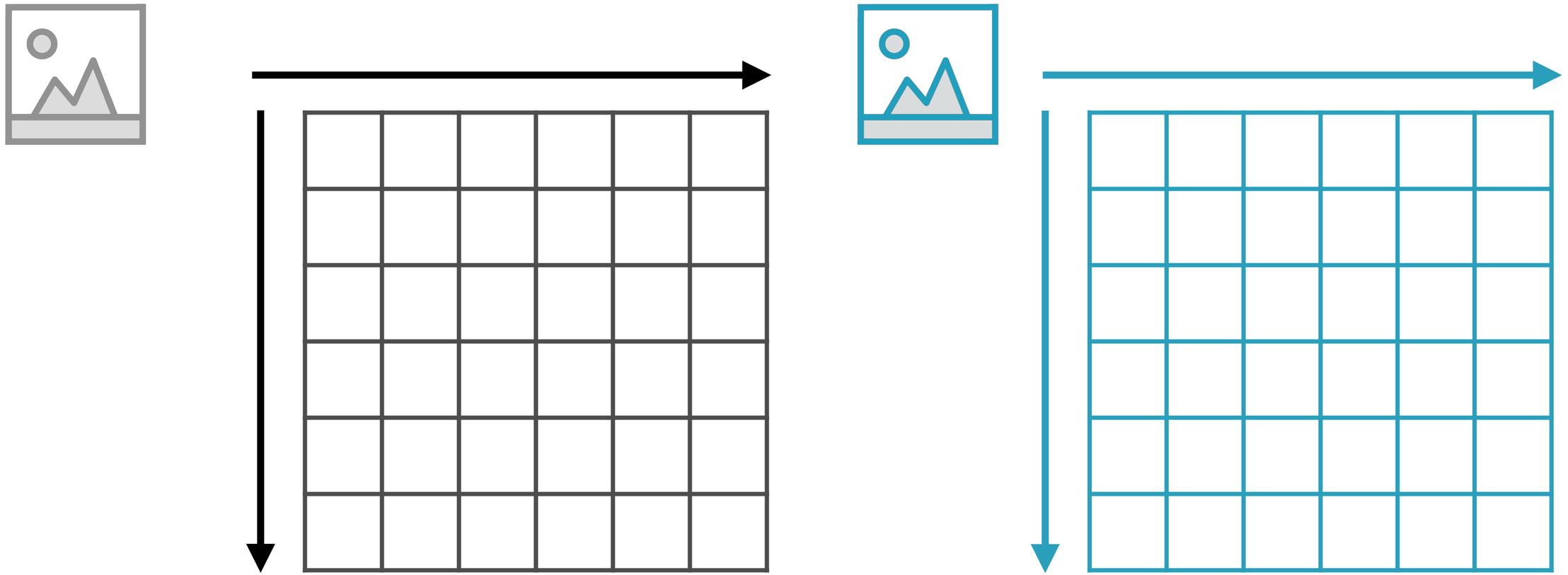
0.0 - 1.0

Grayscale Images



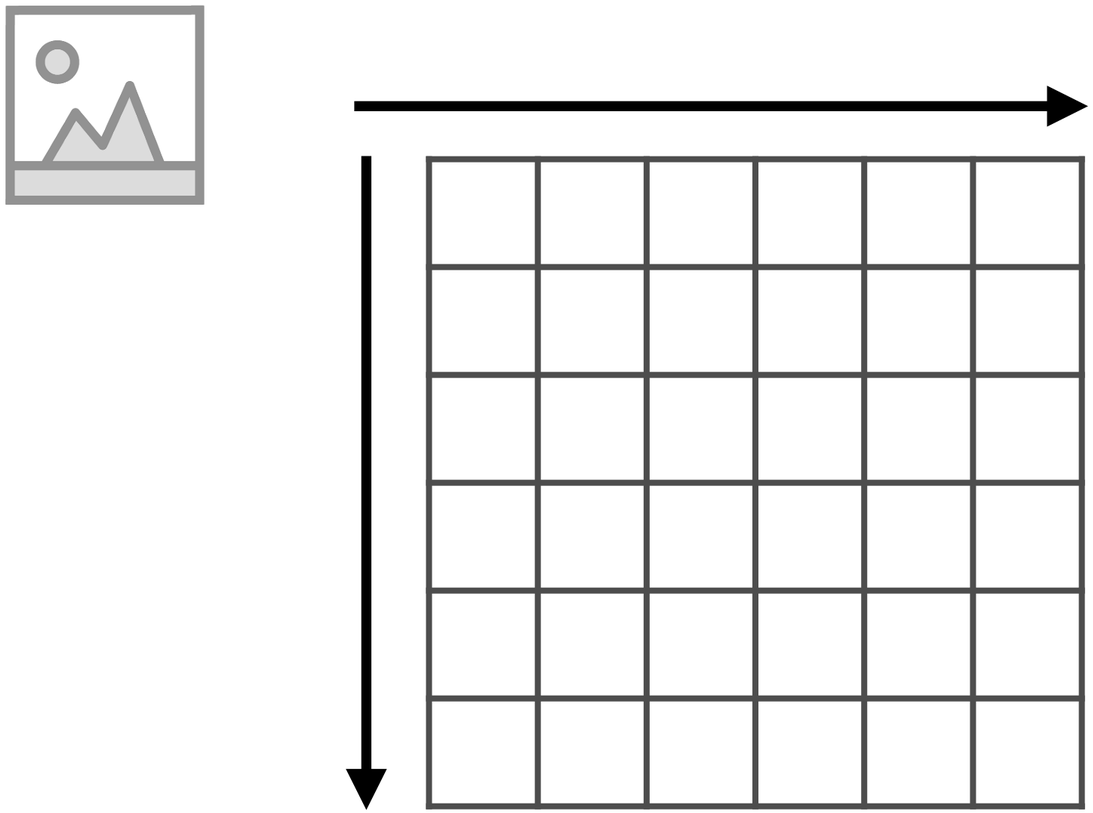
1 value to represent intensity, 1 channel

Images as Matrices

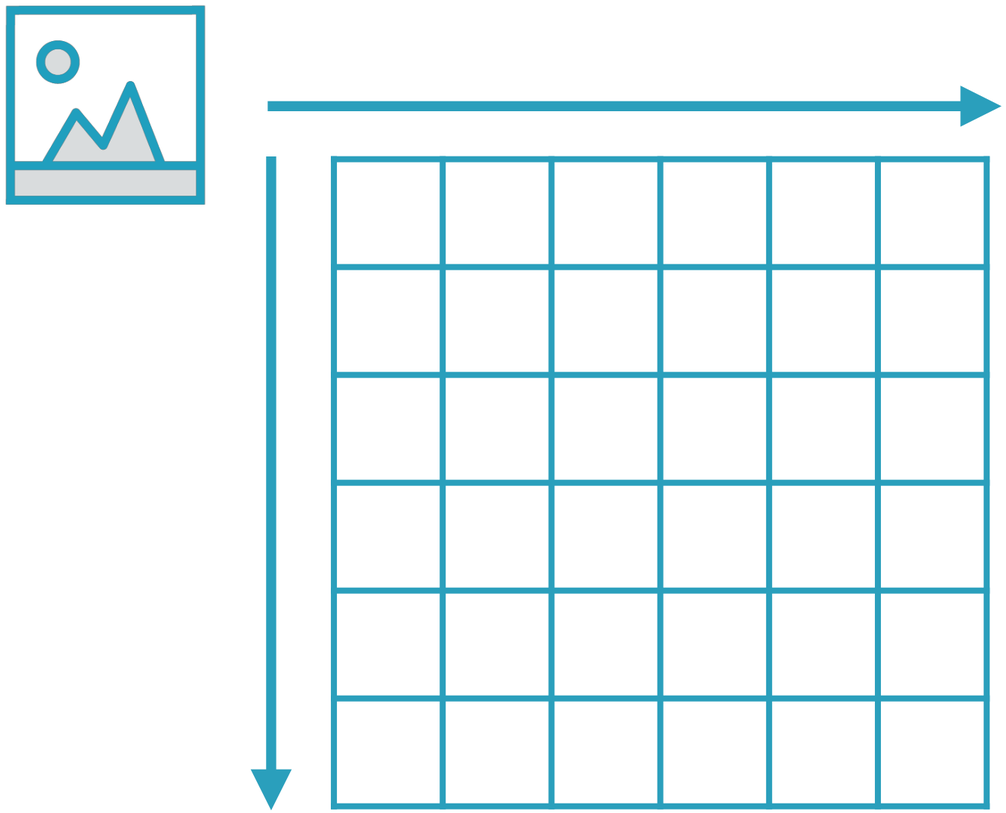


Images can be represented by a 3-D matrix

Images as Tensors



(6, 6, 1)



(6, 6, 3)

List of Images

ML frameworks (e.g. TensorFlow) usually deal with a list of images in one 4-D Tensor

List of Images

The images should all be the same size

List of Images

(10, 6, 6, 3)

The number of channels

List of Images

(10, 6, 6, 3)

The height and width of each image in the list

List of Images

(10, 6, 6, 3)

The number of images

Need for Image Pre-processing

Image Pre-processing Methods

Uniform Aspect Ratio	Uniform Image Size	Mean and Perturbed Images
Normalized Image Inputs	Dimensionality Reduction	Data Augmentation

Common techniques to improve CNN performance

Image Pre-processing Methods

Uniform Aspect Ratio	Uniform Image Size	Mean and Perturbed Images
Normalized Image Inputs	Dimensionality Reduction	Data Augmentation

Common techniques to improve CNN performance

Uniform Aspect Ratio

Most models assume square shape

Crop images to be square

Usually, center of image most important

Makes aspect ratio constant

Image Pre-processing Methods

Uniform Aspect Ratio	**Uniform Image Size**	Mean and Perturbed Images
Normalized Image Inputs	Dimensionality Reduction	Data Augmentation

Common techniques to improve CNN performance

Uniform Image Size

Fit image size to CNN feature maps

250 x 250 image to 100 x 100 image

Downscaling factor of 0.4

Up-scaling and down-scaling

Image Pre-processing Methods

Uniform Aspect Ratio	Uniform Image Size	**Mean and Perturbed Images**
Normalized Image Inputs	Dimensionality Reduction	Data Augmentation

Common techniques to improve CNN performance

Mean and Perturbed Images

Mean image: average pixel across entire training dataset

Insights often emerge

E.g. faces usually in center of image

Mean and Perturbed Images

Perturbed image: intentionally distort pixels by varying them from mean image

E.g. to prevent CNN from only focusing on center

Image Pre-processing Methods

Uniform Aspect Ratio	Uniform Image Size	Mean and Perturbed Images
Normalized Image Inputs	Dimensionality Reduction	Data Augmentation

Common techniques to improve CNN performance

Normalized Image Inputs

“Normalize” each pixel

Subtract mean

Divide by standard deviation

Ensures each pixel has similar data distribution

Normalized Image Inputs

Converts pixels to N(0,1) distribution

Then scale to be in [0,1] or [0,255]

Helps neural networks converge faster

Image Pre-processing Methods

Uniform Aspect Ratio	Uniform Image Size	Mean and Perturbed Images
Normalized Image Inputs	**Dimensionality Reduction**	Data Augmentation

Common techniques to improve CNN performance

Dimensionality Reduction

RGB data has 3 channels

Can reduce to grayscale (just 1 channel)

Reduces dimensionality of all image tensors

Reduce the size of the problem so training completes faster

Image Pre-processing Methods

Uniform Aspect Ratio	Uniform Image Size	Mean and Perturbed Images
Normalized Image Inputs	Dimensionality Reduction	**Data Augmentation**

Common techniques to improve CNN performance

Data Augmentation

Perturbed images are a form of data augmentation

Scaling, rotation, affine transforms

Makes CNN training more robust

Reduces risk of overfitting

Demo

Working with image arrays for single channel and multichannel images

Demo

Manipulating image colors

Demo

Exploring image pre-processing and transformation techniques

Demo

Applying image de-noising techniques

Demo

Normalization to center pixel values

ZCA whitening to de-correlate image features

Demo

Exploring image augmentation using Open CV and the Albumentations Python libraries

Summary

- Represent images as matrices
- Model color information using channels
- Working with grayscale and RGB images
- Understand the need for pre-processing of images
- Implement common operations on images