Advanced Microprocessors

CONVOLUTIONS

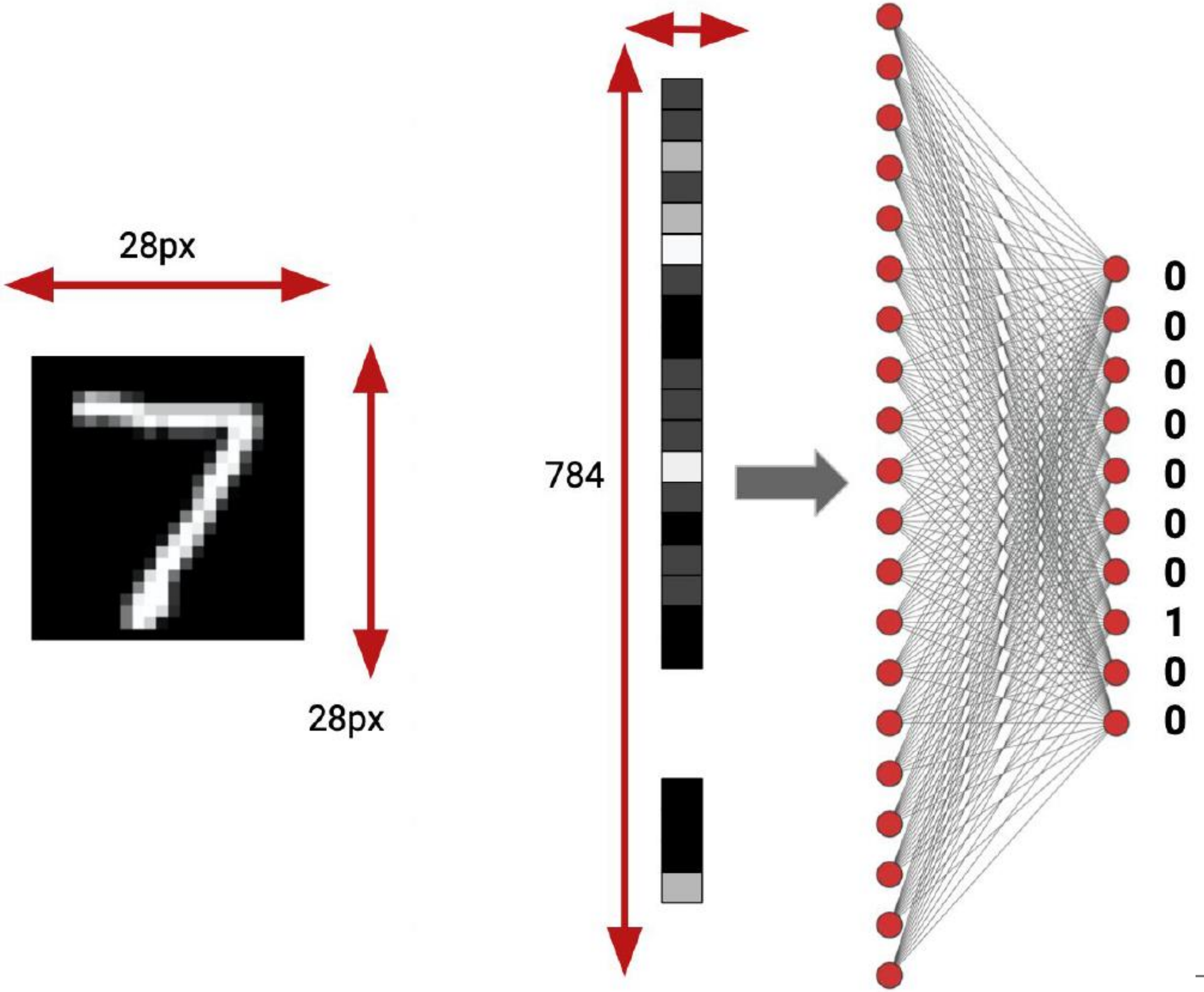
Dennis A. N. Gookyi

CONTENTS

- Convolutions

CONVOLUTIONS

- DNN



CONVOLUTIONS

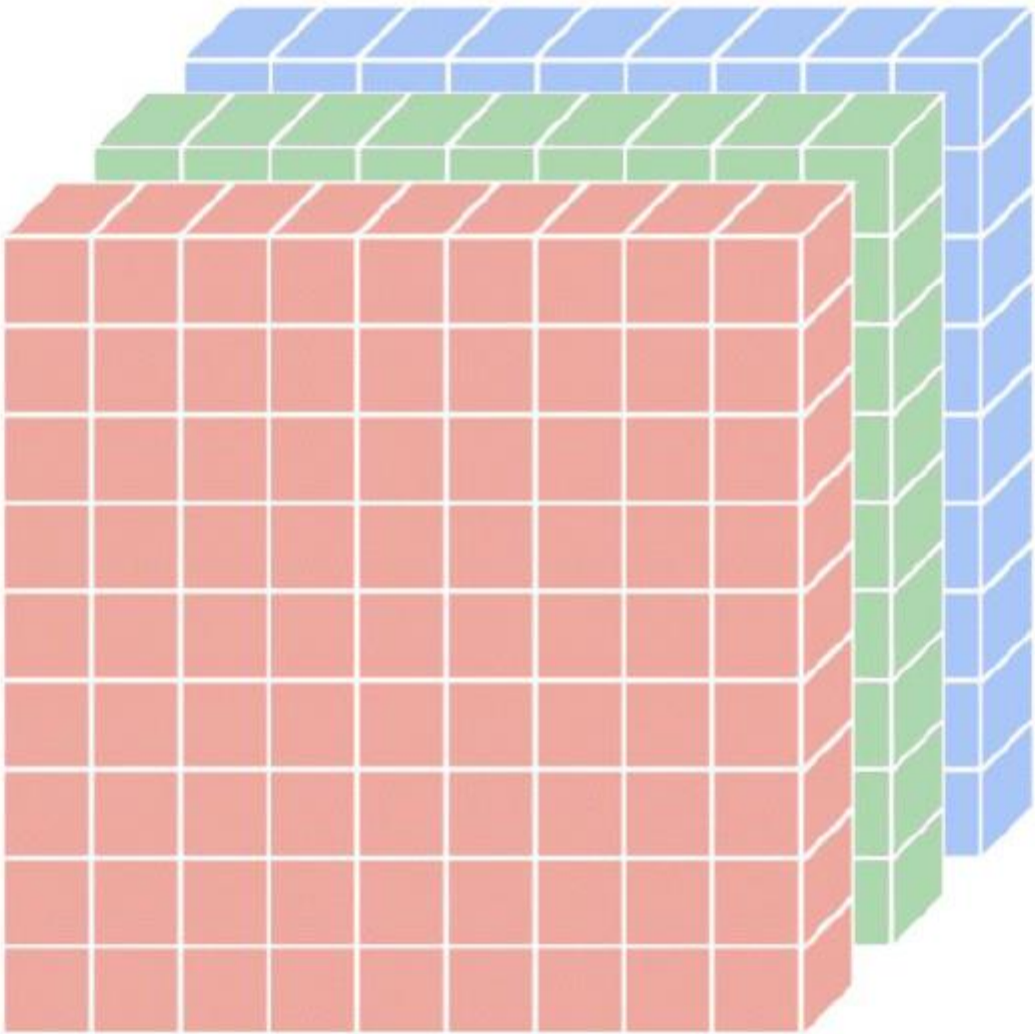
- DNN

CONVOLUTIONS

- Standard Convolution (1 Channel)

CONVOLUTIONS

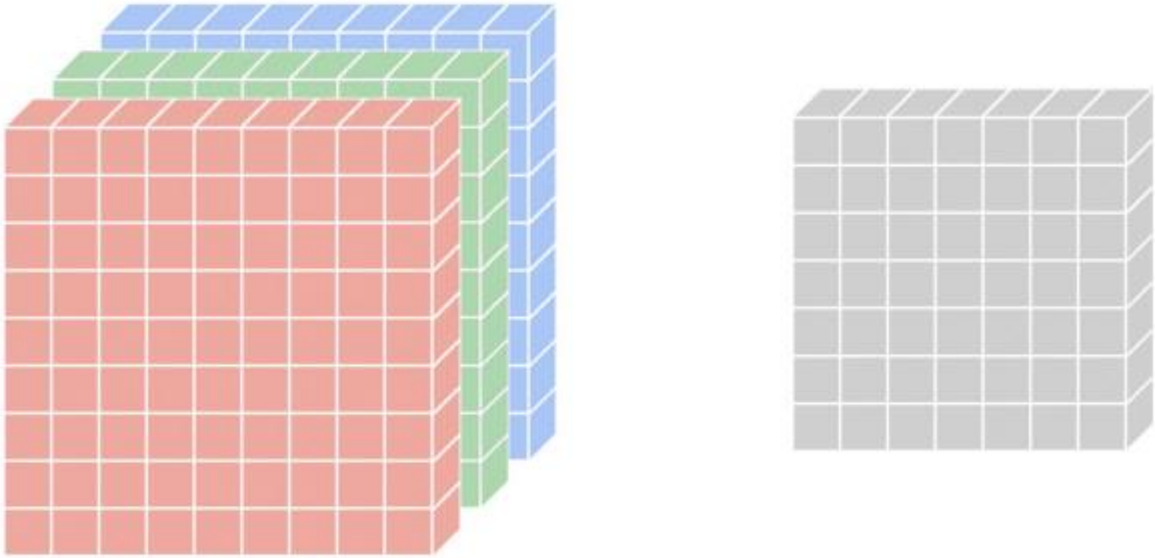
- Standard Convolution (3 Channel - RGB)



CONVOLUTIONS

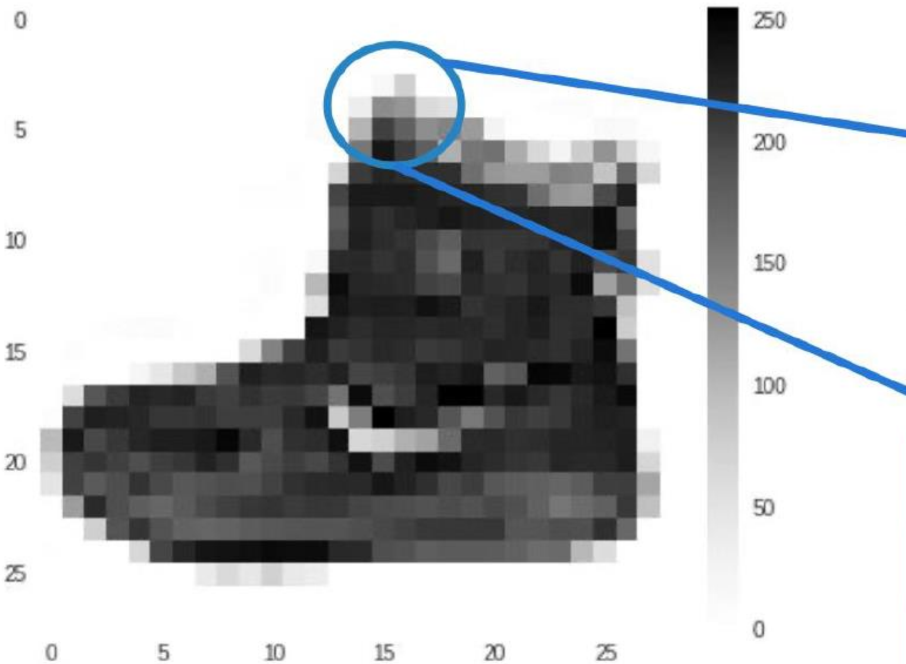
❖ Standard Convolution (3 Channel - RGB)

- Input Feature Map
  - 8 X 8 X 3
  - Width X Height X Channels
- Kernel (*1 Filter*)
  - 3 X 3 X 3



CONVOLUTIONS

❖ Standard Convolution (3 Channel - RGB)



0	64	128
48	192	144
142	226	168

Current Pixel Value is 192

Consider neighbor Values

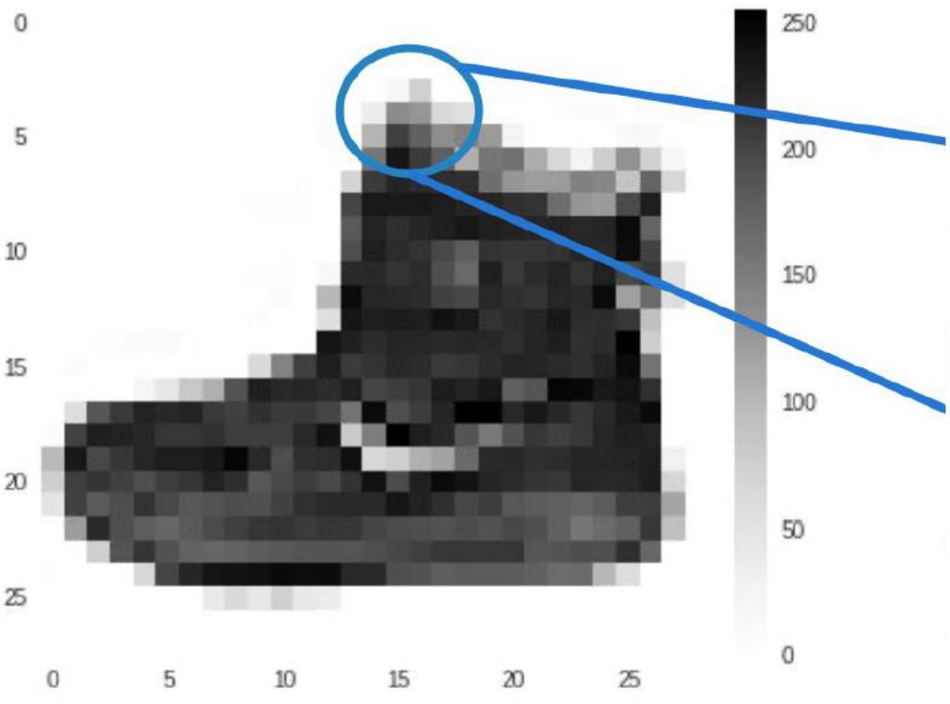
-1	0	-2
.5	4.5	-1.5
1.5	2	-3

Filter Definition

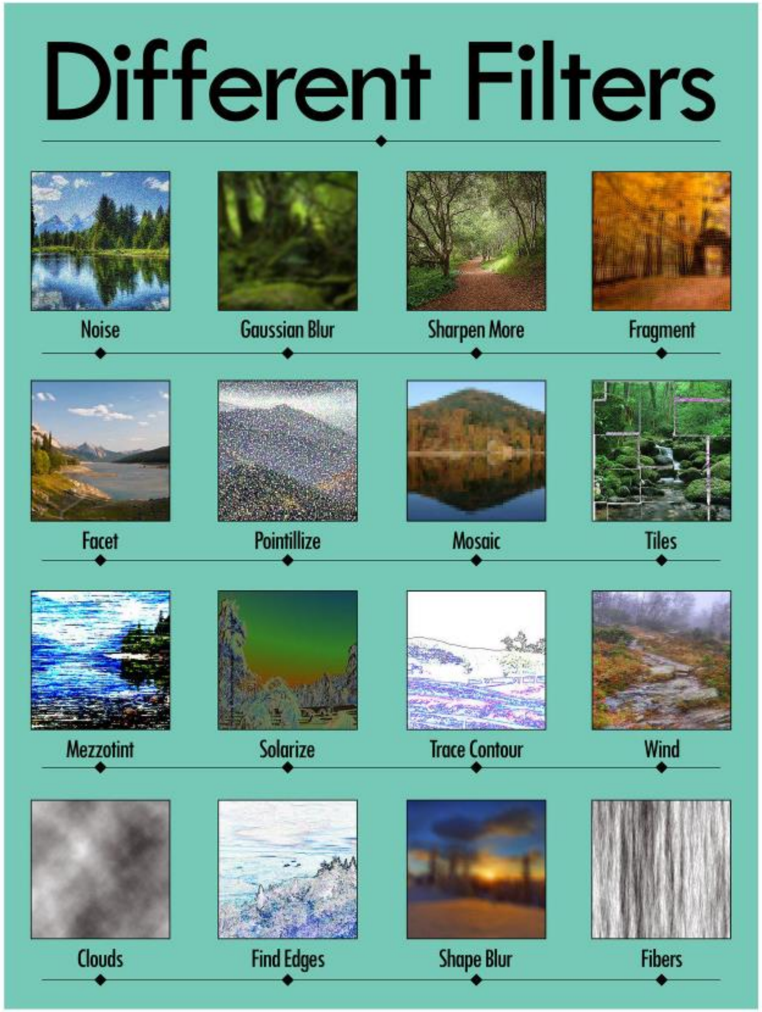
```
CURRENT_PIXEL_VALUE = 192
NEW_PIXEL_VALUE = (-1 * 0) + (0 * 64) + (-2 * 128) +
                  (.5 * 48) + (4.5 * 192) + (-1.5 * 144) +
                  (1.5 * 42) + (2 * 226) + (-3 * 168)
```

CONVOLUTIONS

❖ Standard Convolution (3 Channel - RGB)



Kernels = Filters



CONVOLUTIONS

- Standard Convolution (3 Channel - RGB)

Image Kernels

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

custom

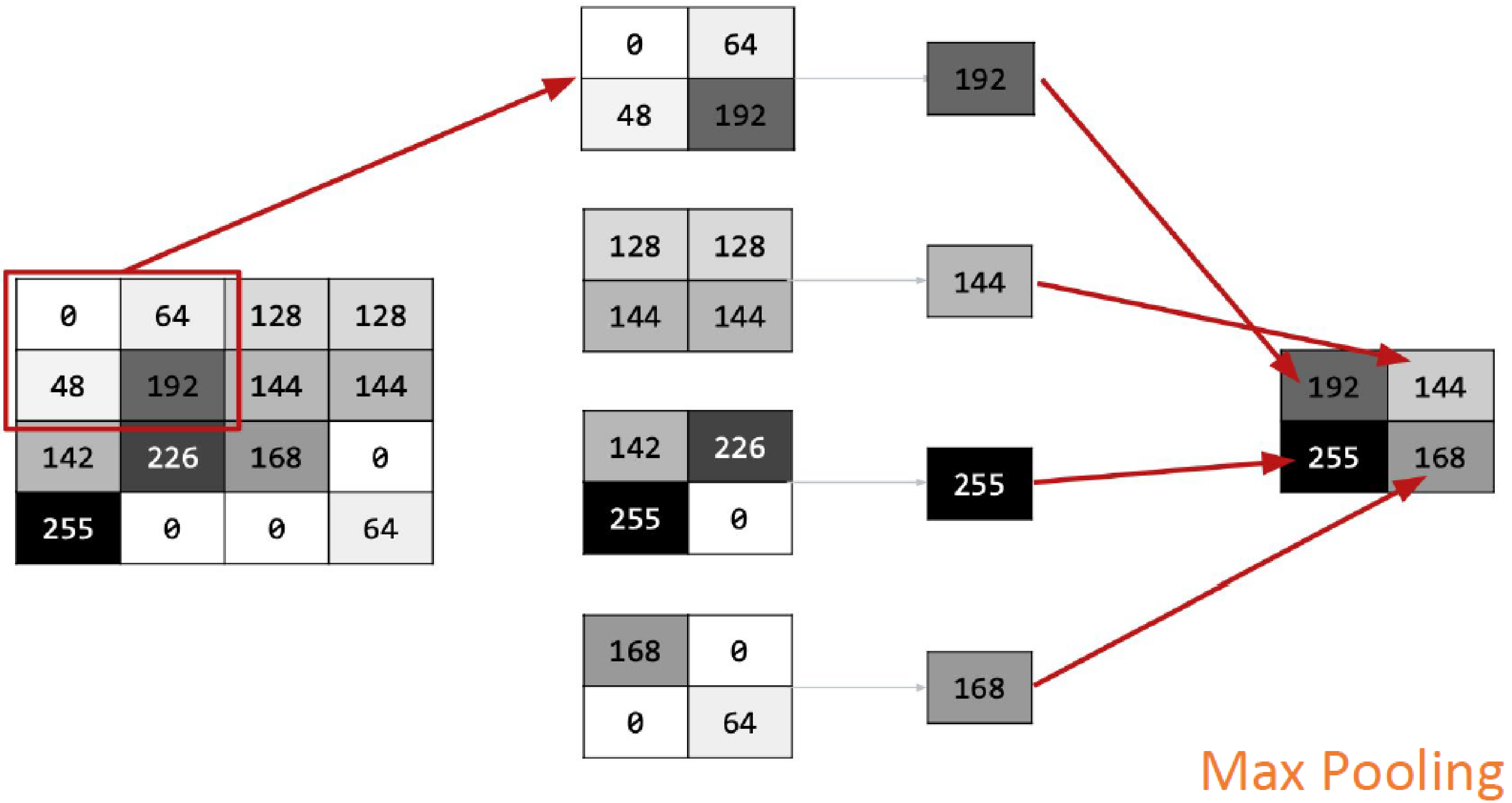
-1	-2	-1
0	0	0
1	2	1

custom

https://setosa.io/ev/image-kernels/

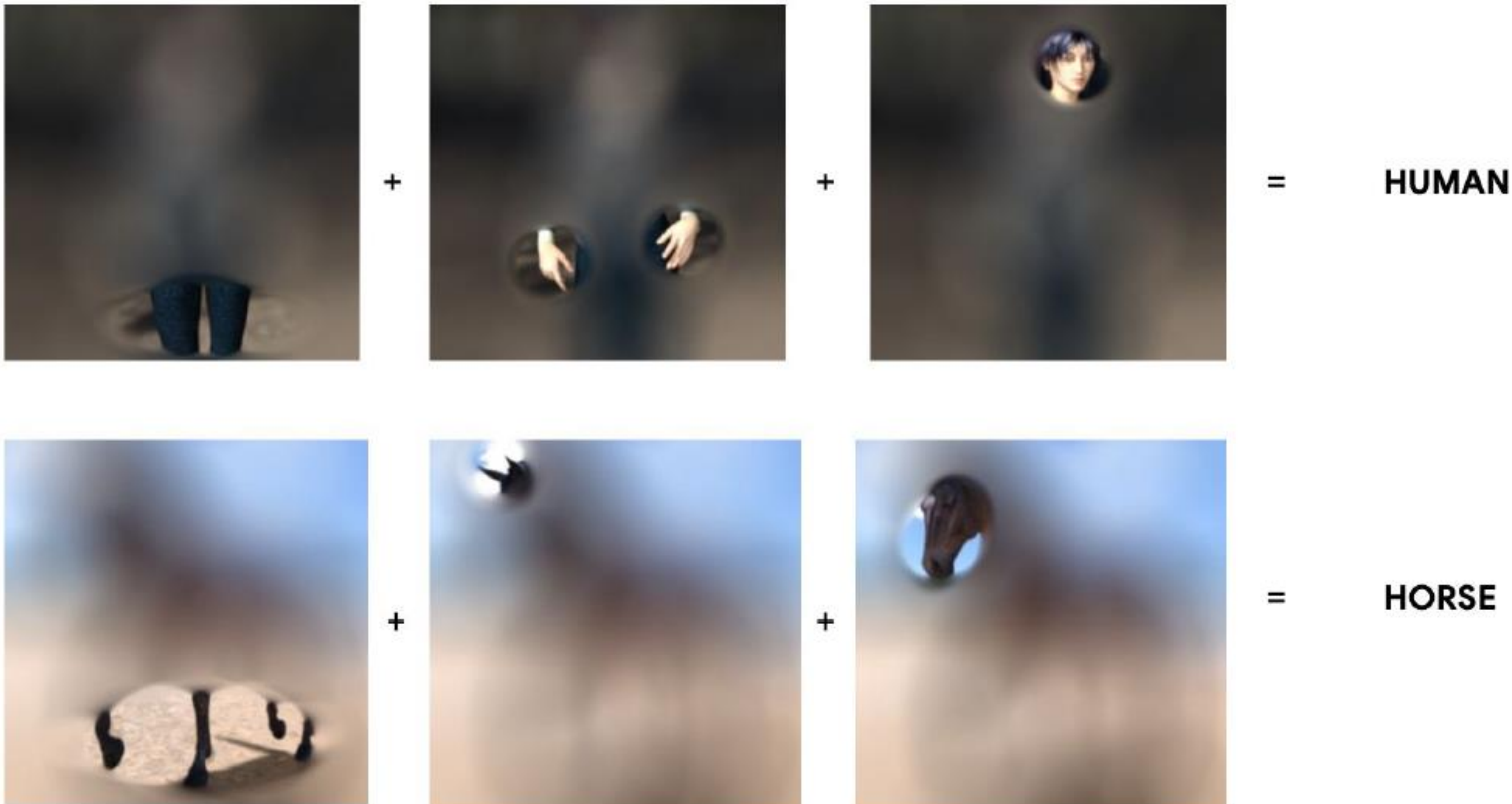
CONVOLUTIONS

- Standard Convolution (3 Channel - RGB)



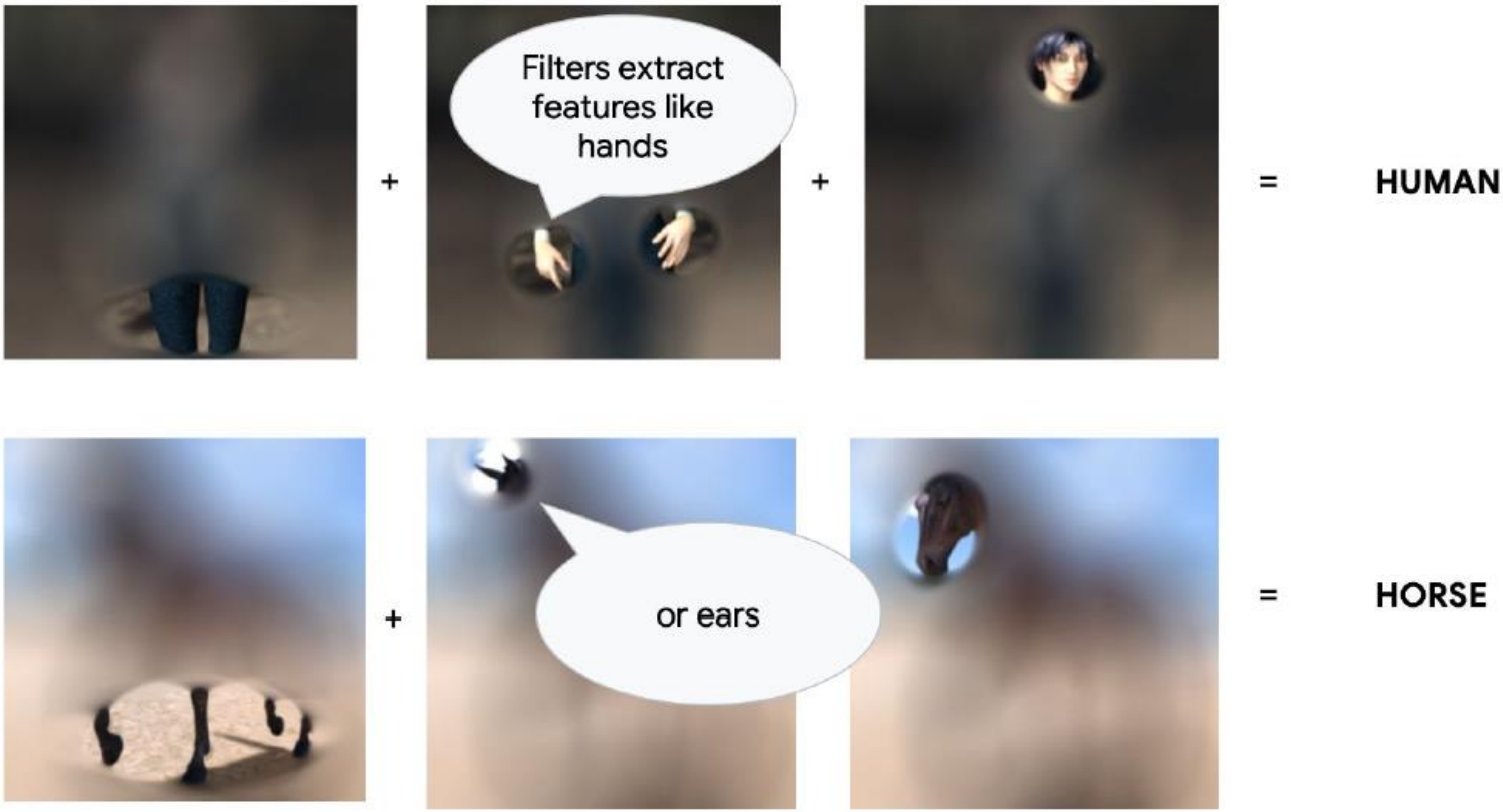
CONVOLUTIONS

❖ Standard Convolution (3 Channel - RGB)



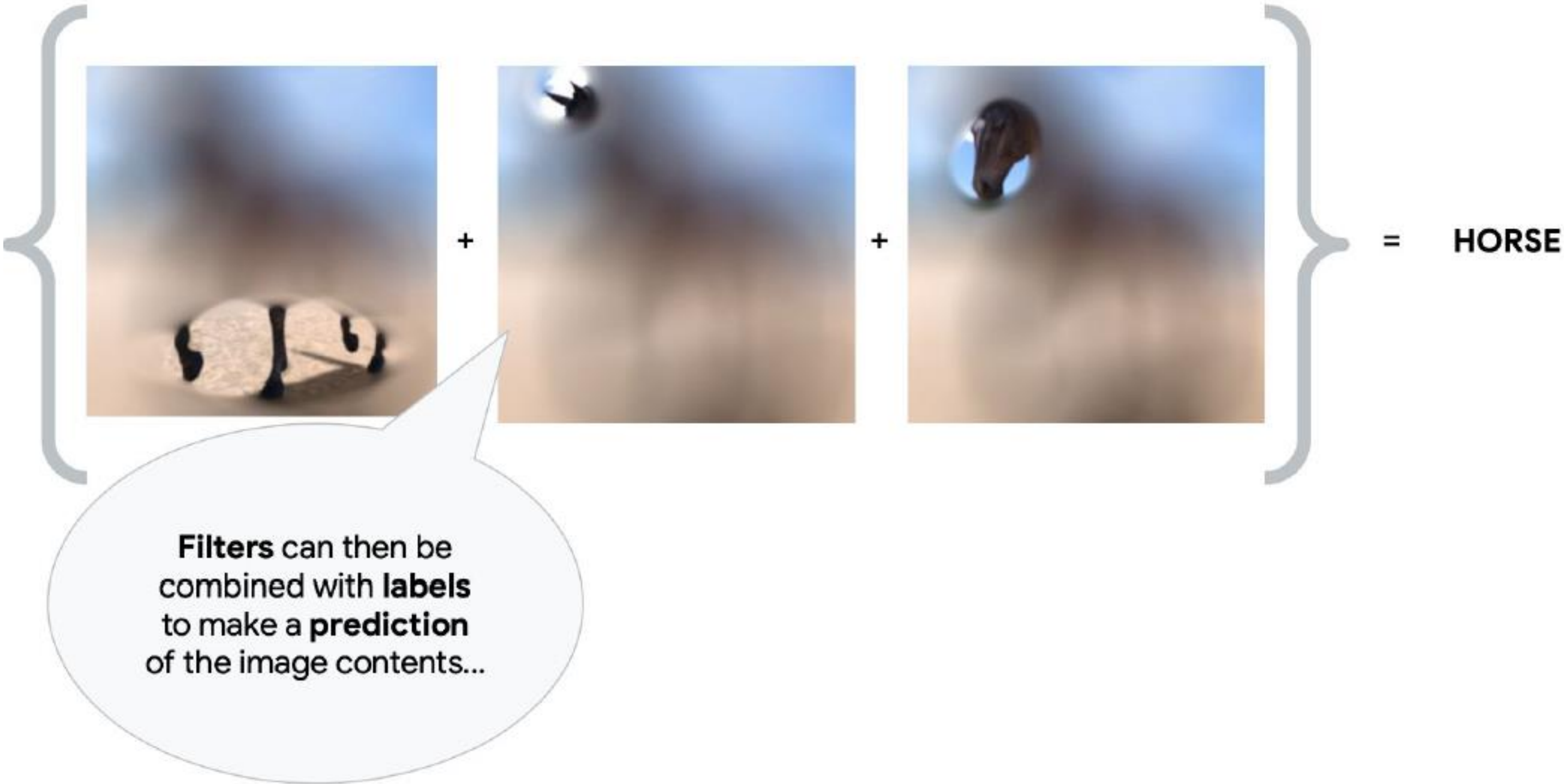
CONVOLUTIONS

- Standard Convolution (3 Channel - RGB)



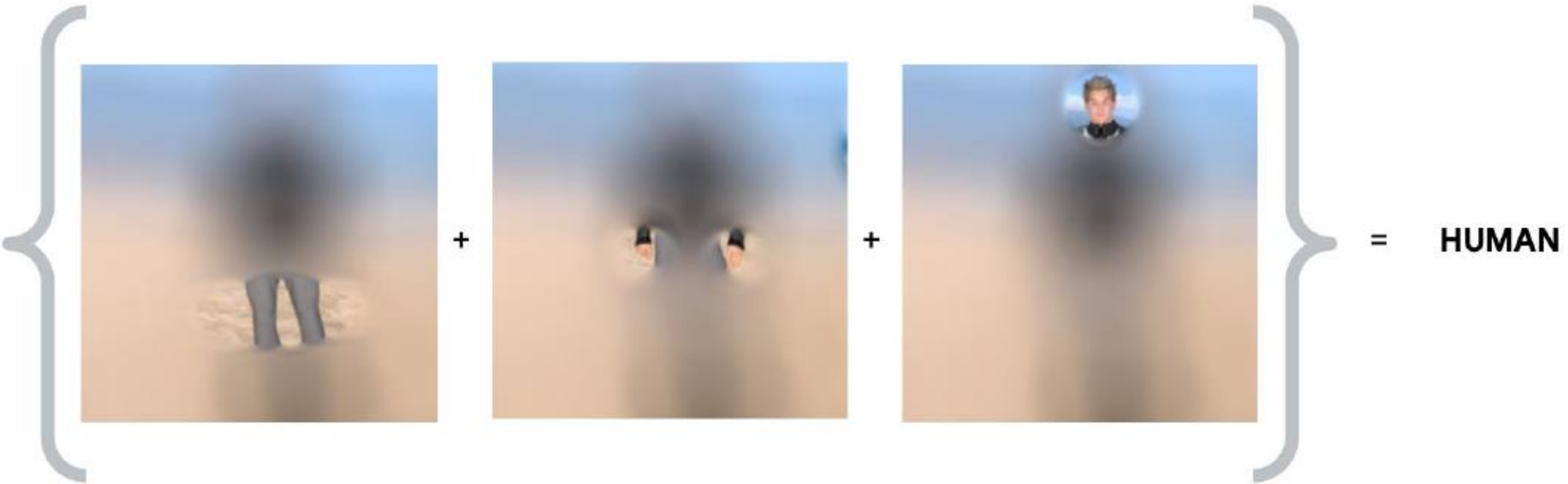
CONVOLUTIONS

❖ Standard Convolution (3 Channel - RGB)



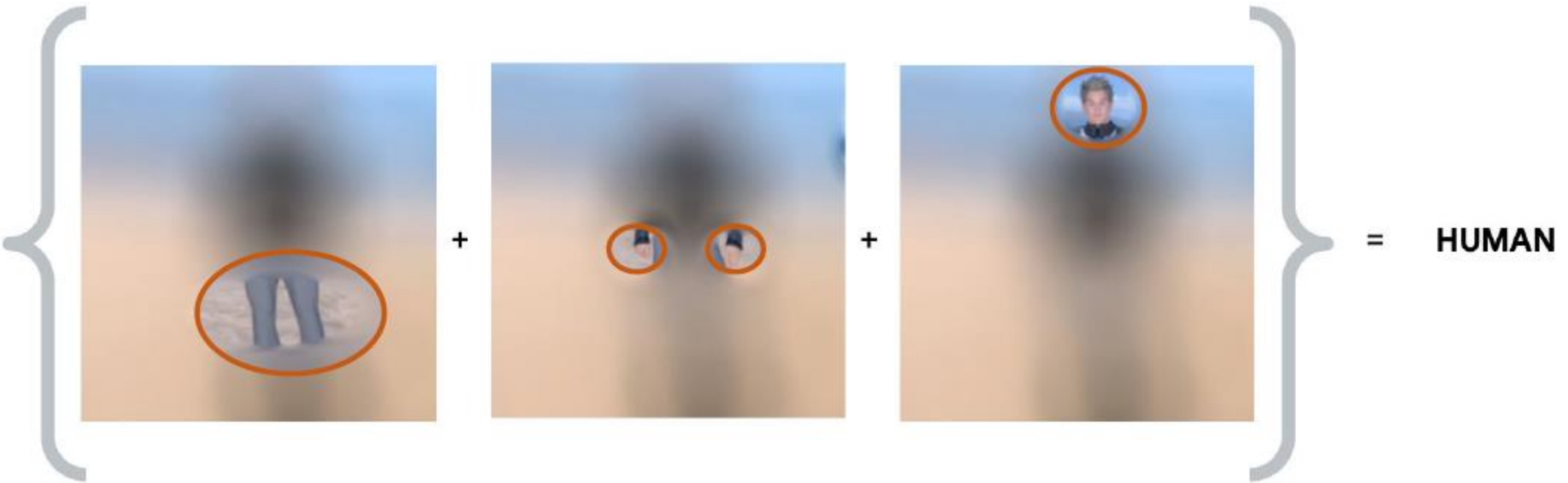
CONVOLUTIONS

- Standard Convolution (3 Channel - RGB)



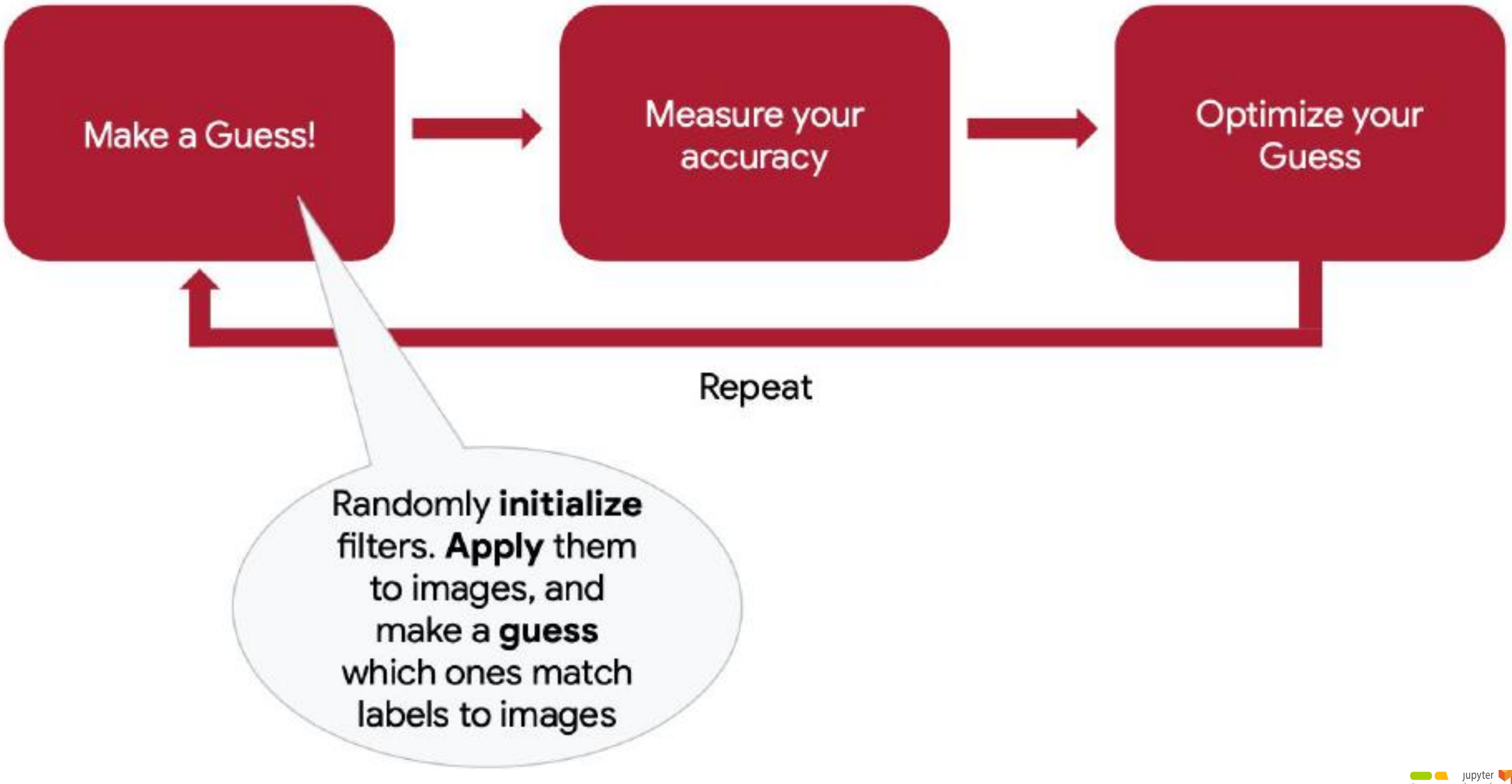
CONVOLUTIONS

- Standard Convolution (3 Channel - RGB)



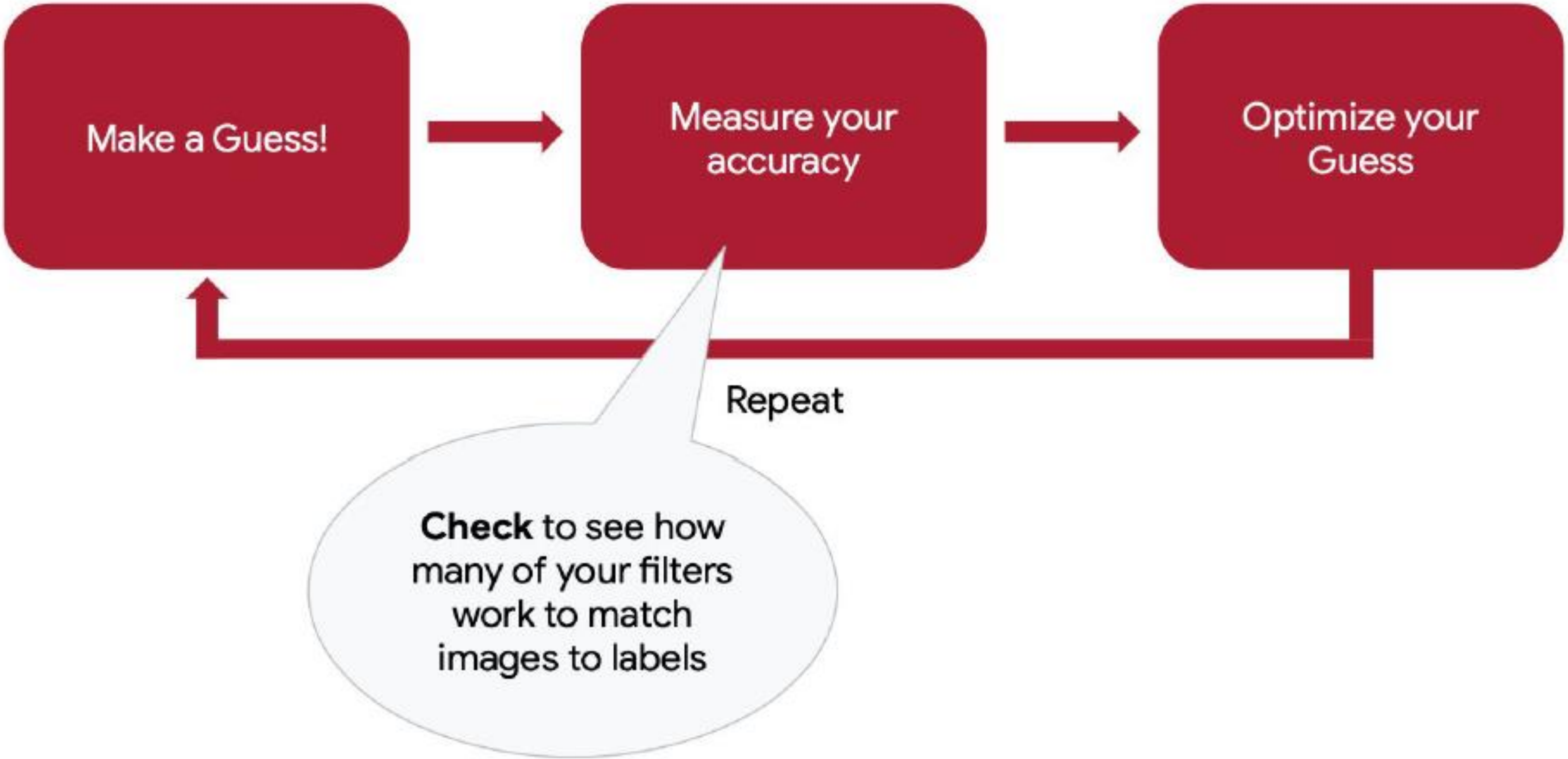
CONVOLUTIONS

- The machine learning paradigm



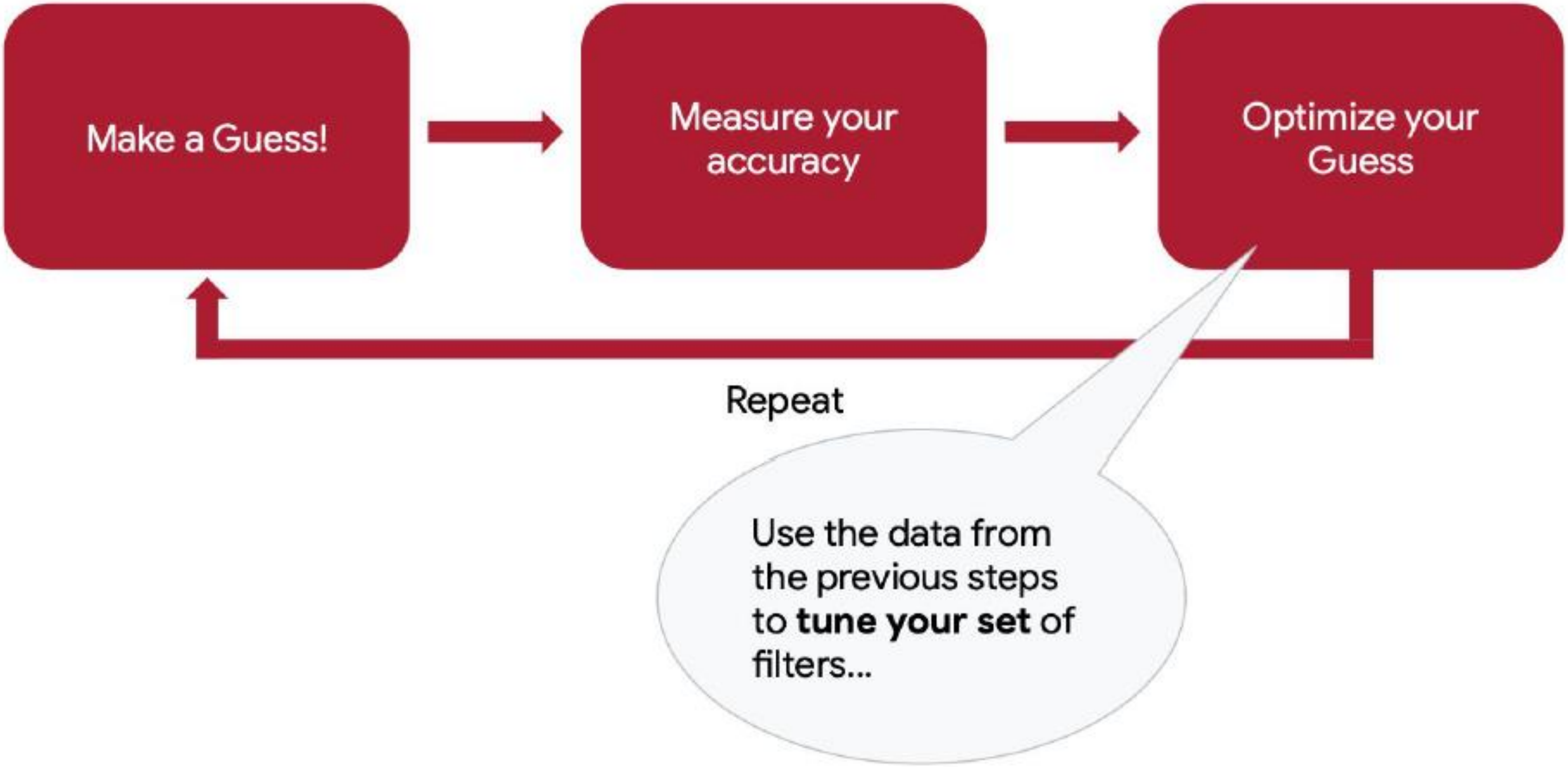
CONVOLUTIONS

- The machine learning paradigm



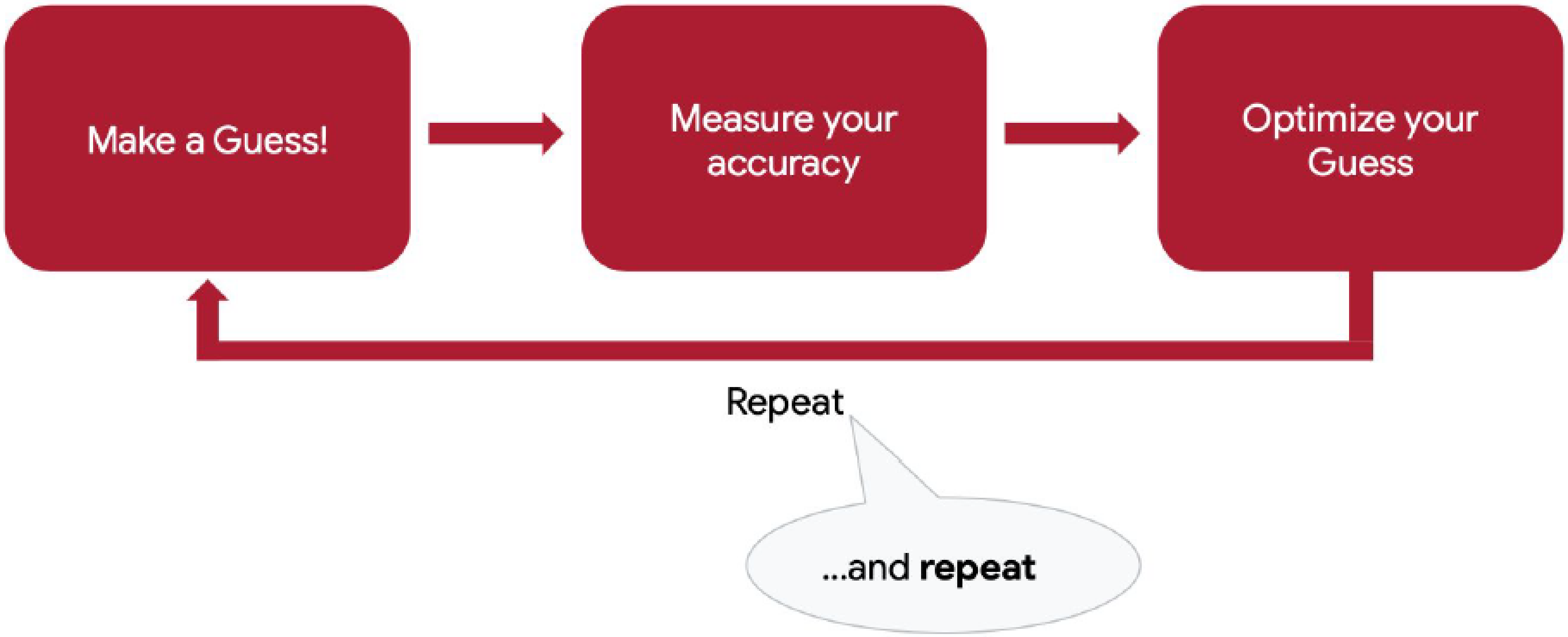
CONVOLUTIONS

❖ The machine learning paradigm



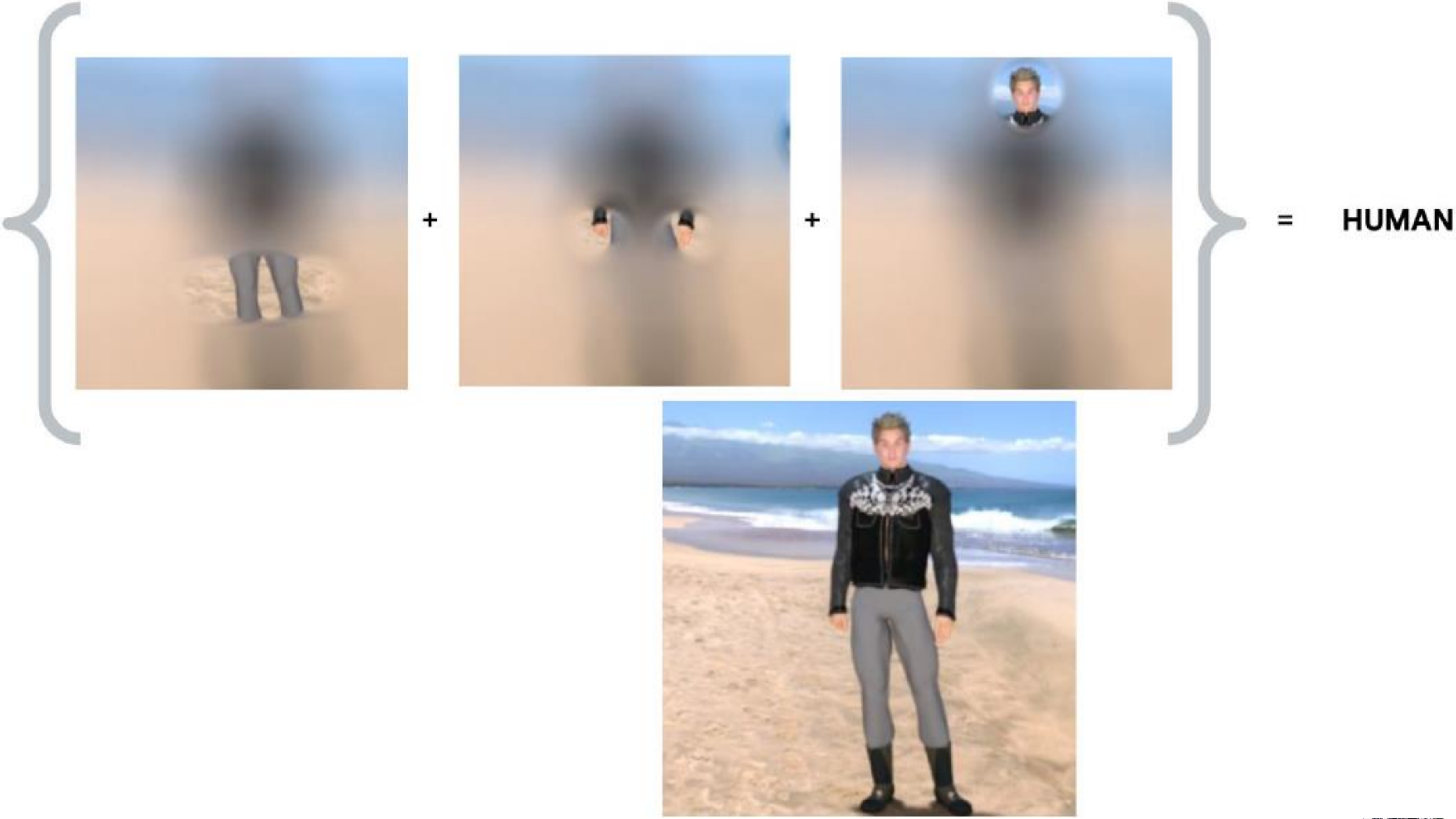
CONVOLUTIONS

❖ The machine learning paradigm



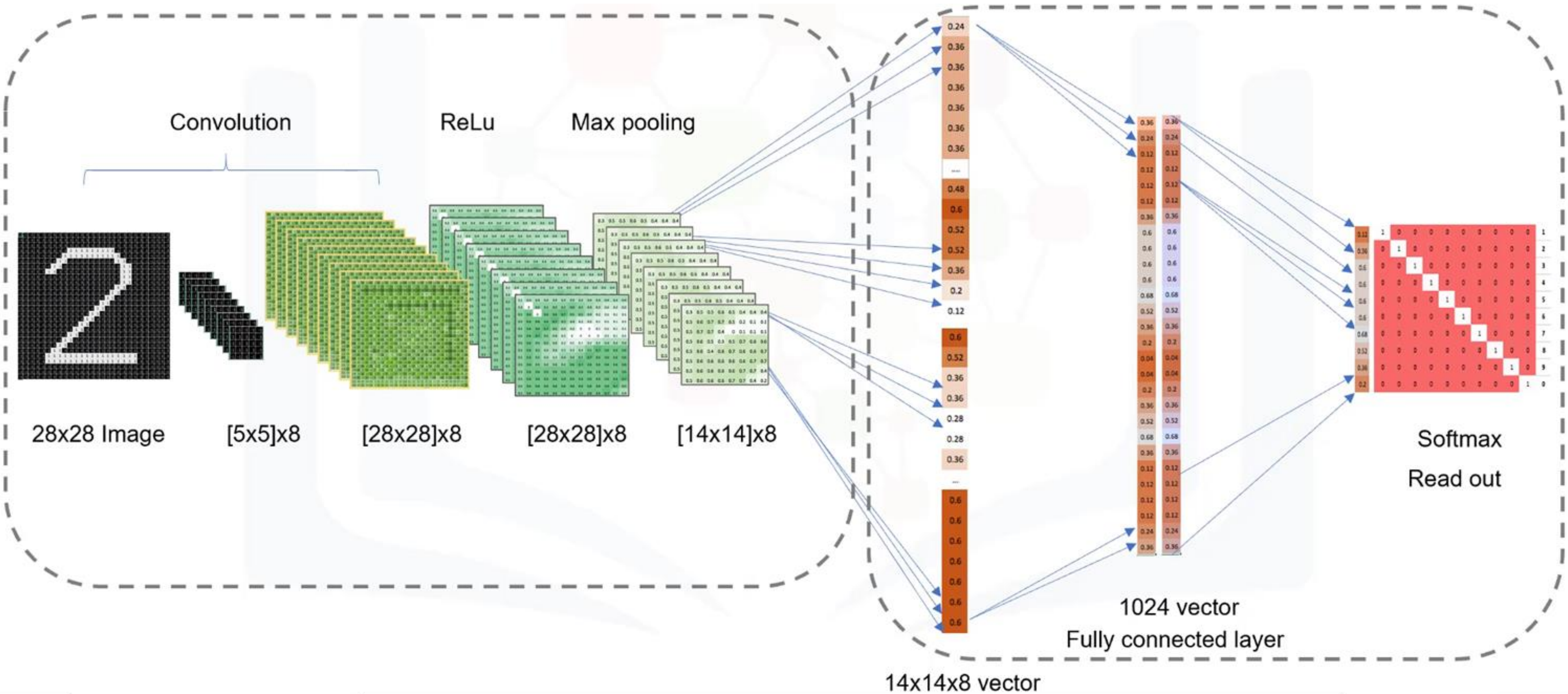
CONVOLUTIONS

- The machine learning paradigm



CNN ARCHITECTURE

❖ CNN Architecture



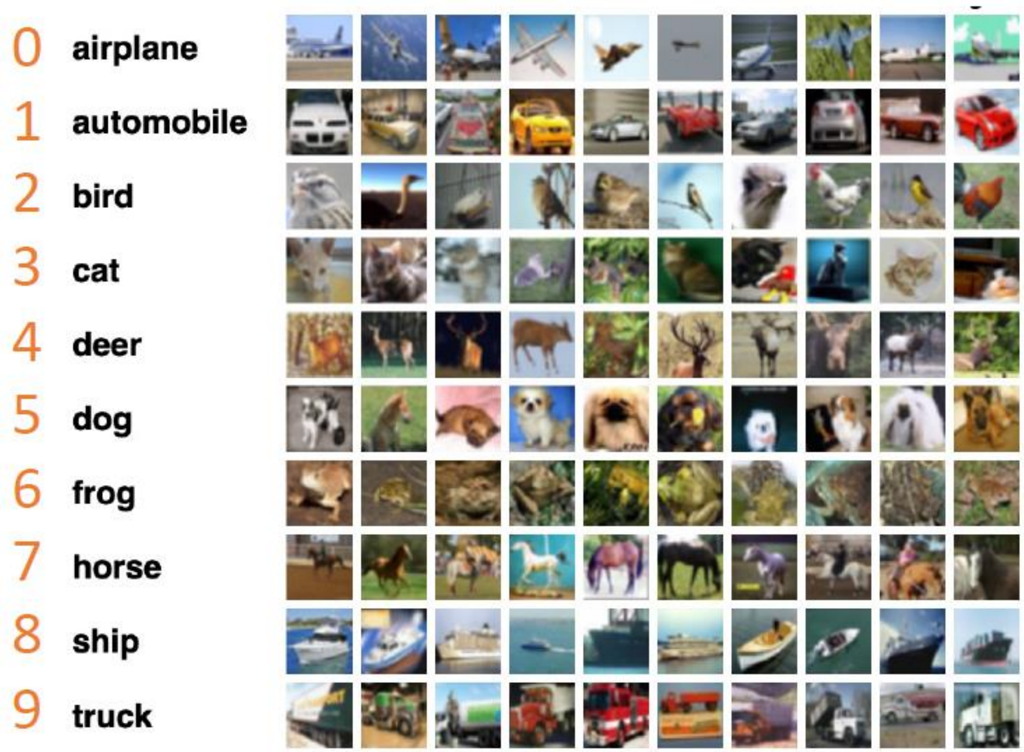
CONVOLUTIONS

- Exploring CNN
  - CNN Explainer
    - https://poloclub.github.io/cnn-explainer/
  - ConvNetJS MNIST demo
    - https://cs.stanford.edu/people/karpathy/convnetjs/demo/mnist.html
  - ConvNetJS CIFAR-10 demo
    - https://cs.stanford.edu/people/karpathy/convnetjs/demo/cifar10.html

CONVOLUTIONS

❖ Image classification using CNN

Cifar-10



https://www.tensorflow.org/datasets/catalog/cifar10

CONVOLUTIONS

- Image classification using CNN
  - We saw how to build Neural Networks (DNN and CNN) that classify images of digits (MNIST)
  - Now we will instead, recognize the 10 classes of CIFAR ('airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', and 'truck')
  - There are some key differences between these image datasets that we need to take into account:
    - While MNIST has 28x28 monochrome images (1 color channel), CIFAR has 32x32 color images (3 color channels)
    - Besides, MNIST images are simple, containing just the object centered in the image, with no background
    - Conversely, CIFAR ones are not centered and can have the object with a background, such as airplanes that might have a cloudy sky behind them
  - Those differences are the main reason to use a CNN instead of a DNN

CONVOLUTIONS

❖ Image classification using CNN

Image Classification using CNN

Code Time!

CNN_Cifar-10.ipynb