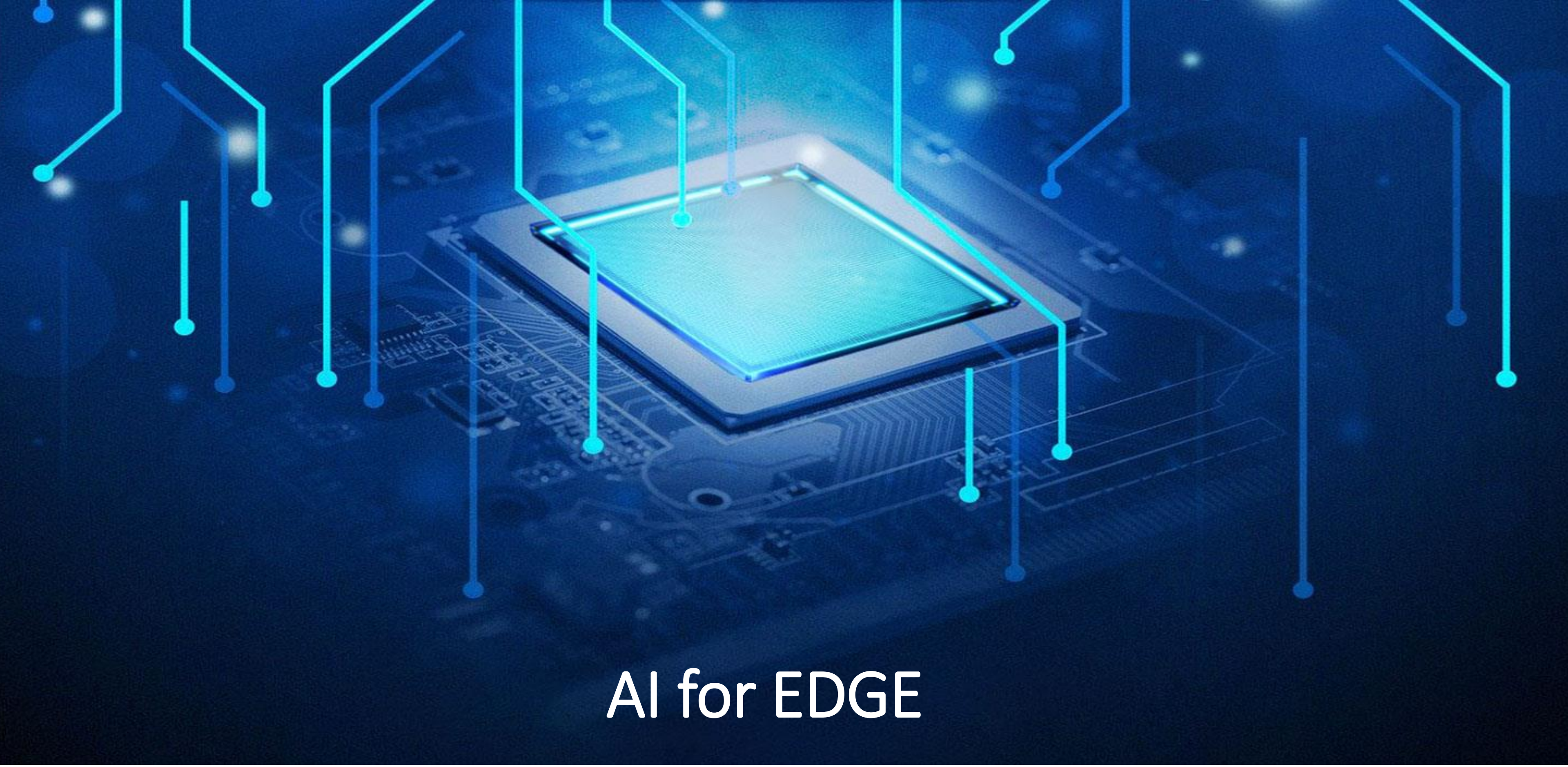


AI for EDGE

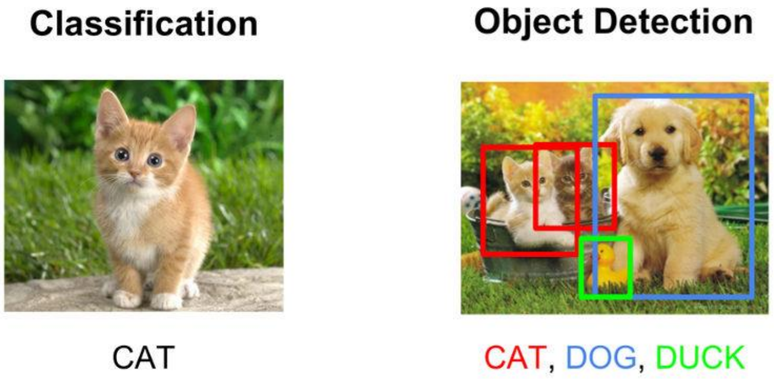
Image classification using CNN

Outline

- Computer vision - Image classification
- Convolution and Convolutional Neural Networks
- Image classification using CNN
- Classification use case: Visual Geometry Group CNN
- Receptor field
- Transfer learning

Computer vision

- Scientific field that deals with how computers can gain high-level understanding from digital images
- Understanding in this context means the transformation of visual images into descriptions of the world
- Sub-domains of computer vision:
  - object detection,
  - image classification,
  - video tracking,
  - motion estimation,
  - scene reconstruction,
  - 3D scene modeling, and
  - image restoration



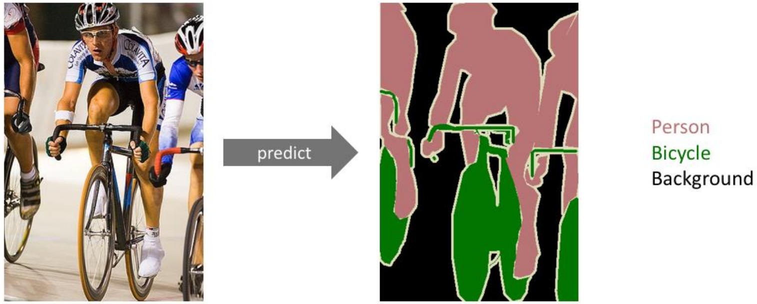


Image classification

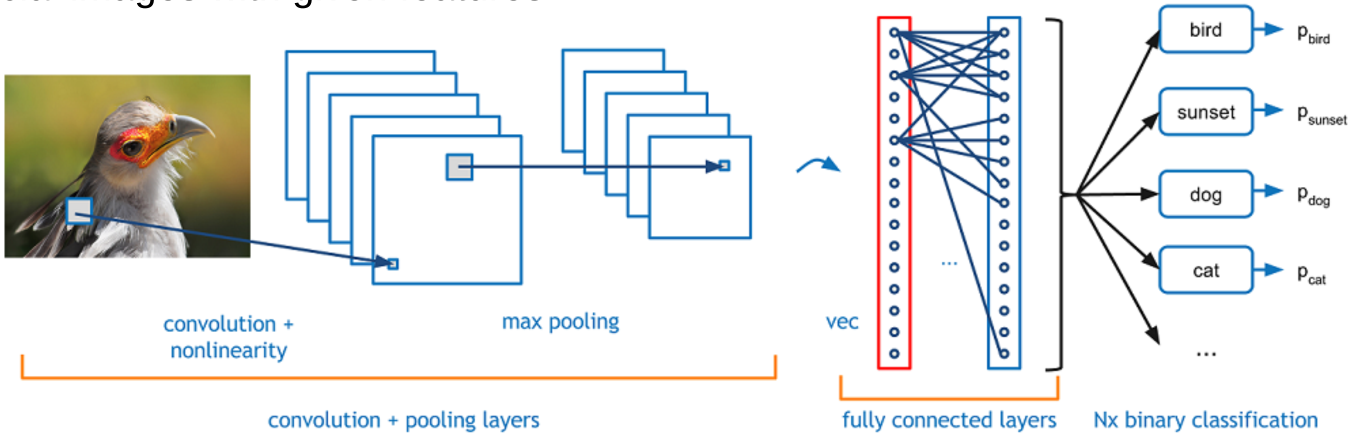
Image classification –

- ○ technology in the field of computer vision for finding and identifying objects in an image or video sequence
- ○ one or several pre-specified or learned objects or object classes can be recognized, usually together with their positions in the image

Image classification is probably the most important part of digital image analysis.

CNNs

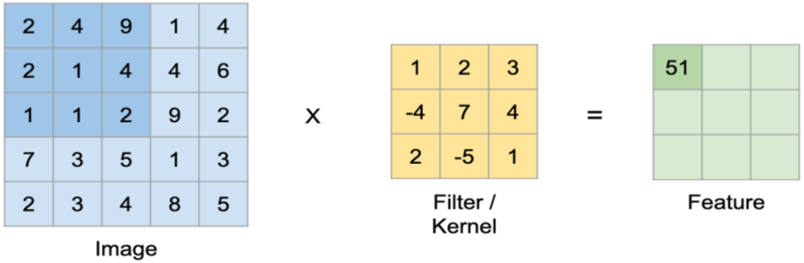
- The invention of convolutional networks caused a breakthrough in the field of deep learning and computer vision
- Convolutional neural networks are currently achieving state of the art results in image processing
- Convolutional architectures are currently used in:
  - image classification
  - image segmentation
  - object detection in the image
  - tracking objects over a series of camera frames
  - generating artificial images with given features



Convolution

Convolution is the multiplication of the corresponding values of two functions over a certain interval and adding the results together

$$(f * g)[n] = \sum_{m}^{n} f[m]\, g[n-m]$$



2	4	9	1	4
2	1	4	4	6
1	1	2	9	2
7	3	5	1	3
2	3	4	8	5

Image

x

1	2	3
-4	7	4
2	-5	1

Filter / Kernel

=

51		

Feature

Sequence of actions in convolution:

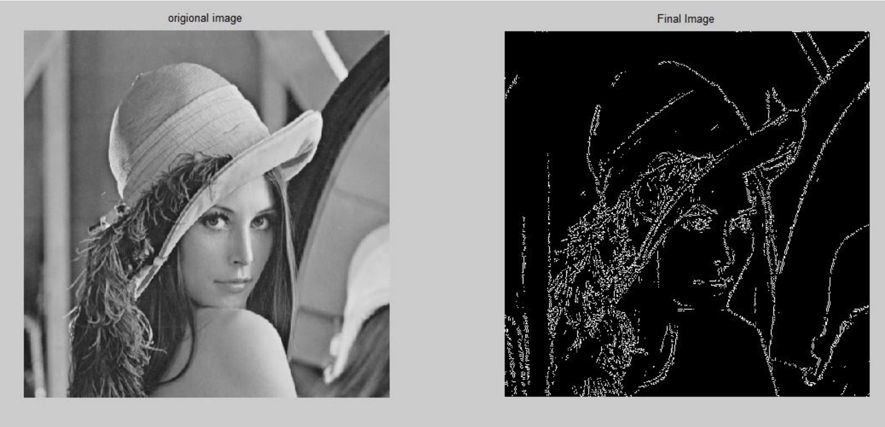
- We adjust the window (aka kernel/filter) with "certain" values to the image so that the anchor point (usually the center point of the kernel) coincides with the pixel whose value we will modify
- We multiply the corresponding values, and then add them, obtaining the resulting pixel value after transforming with a window
- The process is repeated by moving the window by a given step

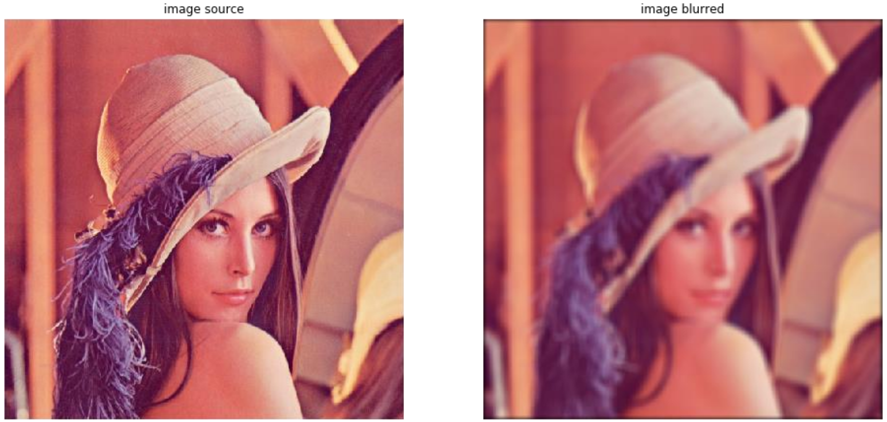
Convolution

Using convolution, you can transform images in various ways.

Examples of possible effects:

- smoothing
- sharpening
- noise reduction
- edge detection





Convolution

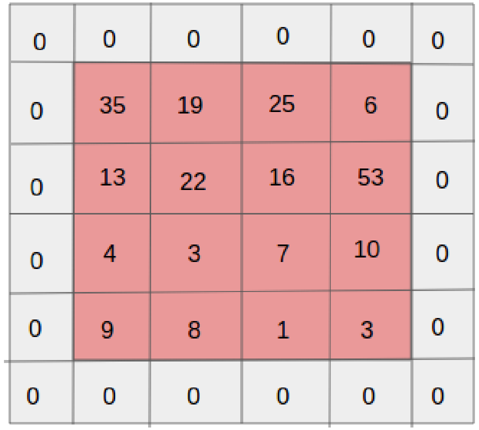
Summary:

- The transformation of the image depends on the value in the kernel.
- Window sizes may vary. The most common are 3x3, 5x5, 7x7 pixels.
- The larger the kernel, the greater the computational complexity of the convolution.
- Some filtering methods rely on window fitting and shifting similar to convolution, but do not use convolution (multiply and add) operations. We call such filters non-linear.
  An example is the median filter, which replaces a given pixel with its median from the vicinity of the window applied to the image.

Convolution

The problem of convolution on pixels from the edge of the image

- When convolution is used on edge pixels, a situation may arise in which part of the window will protrude beyond the image. In this case, we can either omit this pixel, or apply the so-called padding, i.e. adding an additional border of pixels to the image.
- If padding is omitted, the resulting convolution image will be smaller than the original, and if padding is applied it will be the same size.
- The most commonly used types of padding are zero padding and padding by mirroring pixel values.



0	0	0	0	0	0
0	35	19	25	6	0
0	13	22	16	53	0
0	4	3	7	10	0
0	9	8	1	3	0
0	0	0	0	0	0

Image classification using CNN

The convolutional neural network that performs the classification is divided into two parts:

- Feature extractor
- Classifier

- The feature extractor performs convolution operations on the input images using filters whose numerical parameters (weights) are selected during the training of the network. Each output pixel is passed through the activation function. In this part of the network, instead of feature vectors, we have feature maps.
- The classifier is a small feedforward network whose input is the flattened resulting image from the feature extractor.

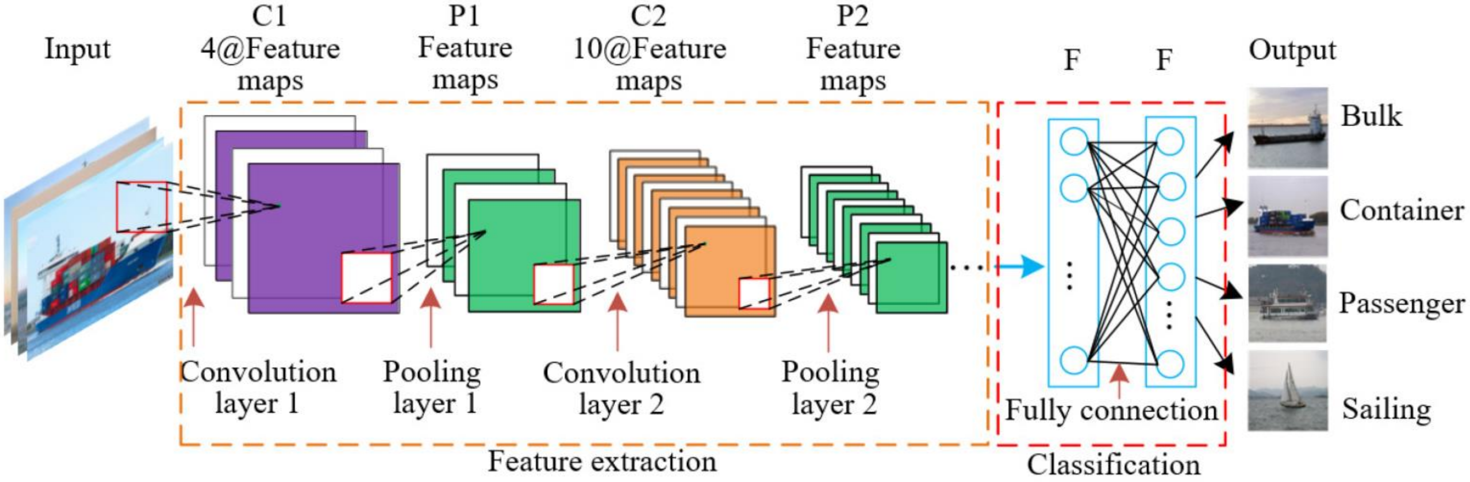


Image classification using CNN

Network convolution operations have the following basic parameters:

- filters: number of filters
- kernel_size: window size, e.g. 3x3, 5x5
- strides: step by which the window is moved in the x and y axes, e.g. (1, 1)
- padding: type of padding used. In Keras **'valid'** means no padding, **'same'** means padding with zeros, and **'reflect'** padding by mirroring.
- activation: the activation function that will be entered by the sum of the products from the applied window

The input/output tensor from a convolutional layer usually has two shape formats:

- NHWC: (batch, height, width, channels)
- NCHW: (batch, channels, height, width)
- Channels is the number of channels equal to the number specified in the filters parameter of the layer.

Image classification using CNN

Images entering convolutional layers usually have more than one channel (e.g. when we have an RGB image for input).

In this case, for each filter, the convolution is performed on all channels of the input image, and the resulting images are summed.

Operation sequence:

- ○ Carrying out convolution on each channel separately.
- ○ Summing the resulting images.
- ○ Adding bias.
- ○ Passing the resulting image through the activation function.

The weights of a single filter are [H x W x input_channels],
and the weights of the entire layer are [H x W x input_channels x output_channels].

Image classification using CNN

Feature maps - example

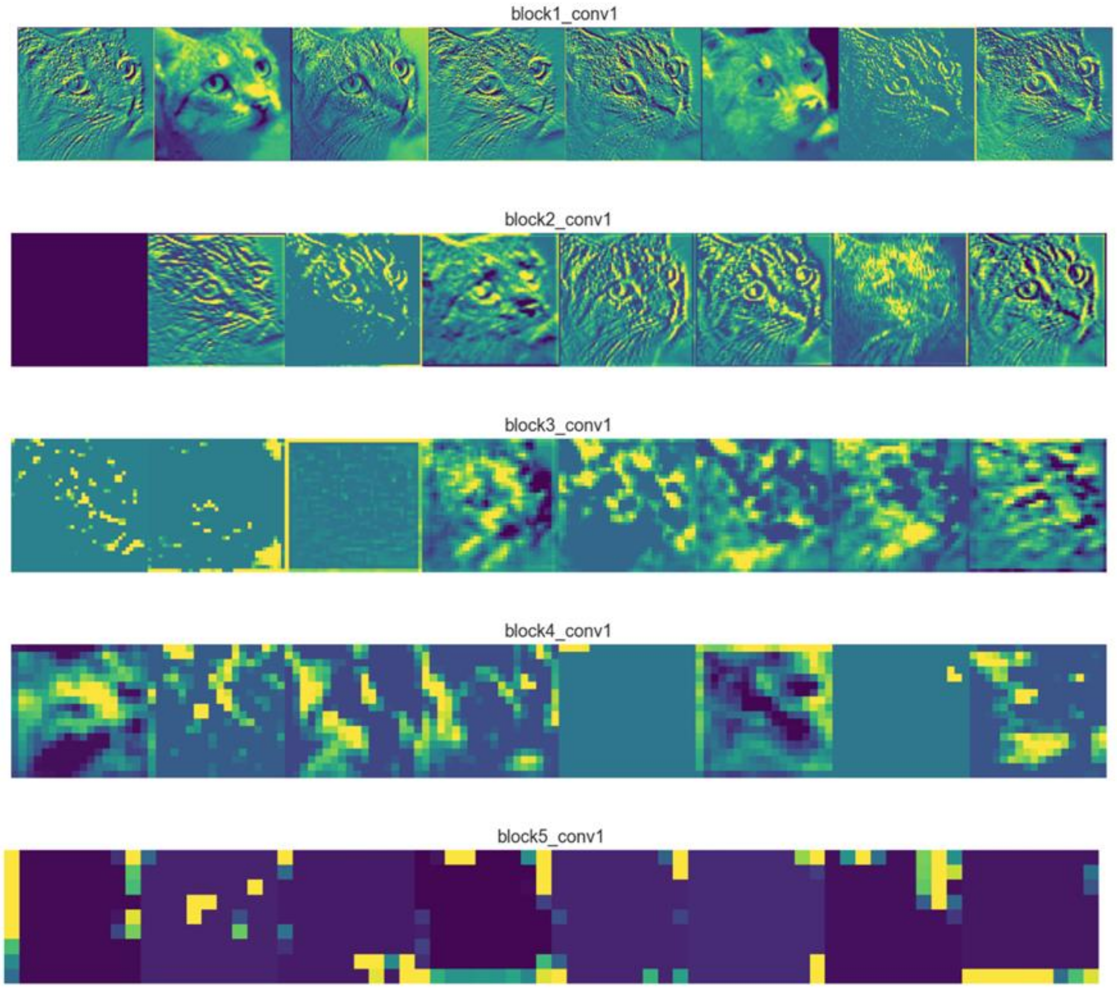


Image classification – case study

- VGG is a convolutional neural network that won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014, which consisted of multiclass classification for 1000 classes of different objects.
- This network serves as a role model for developing our own convolutional networks to this day.

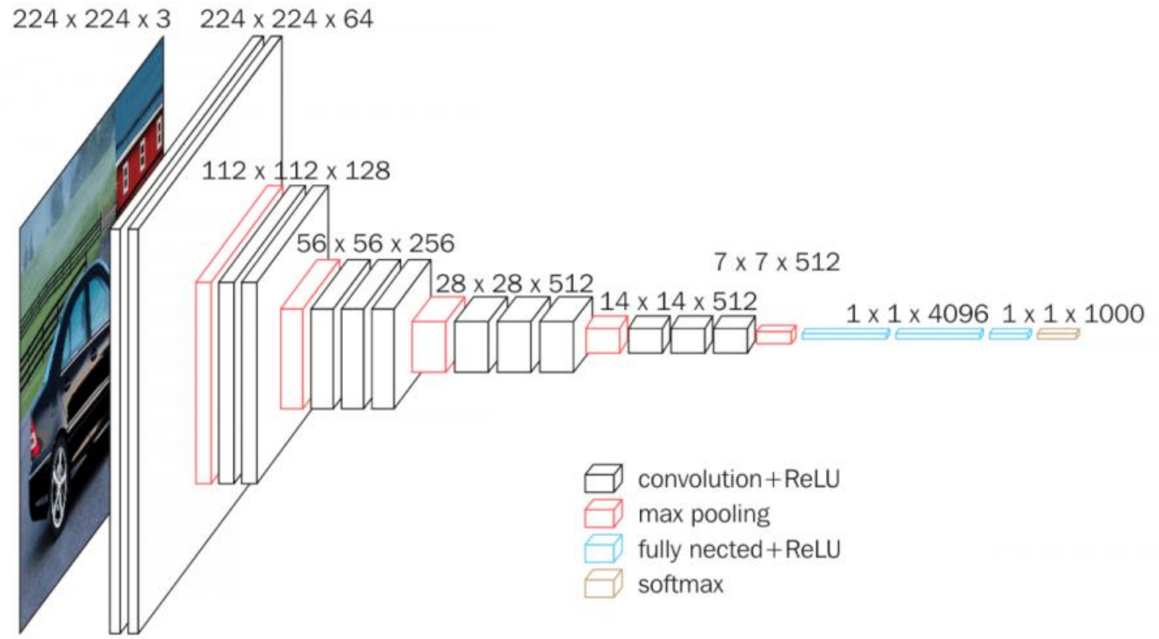
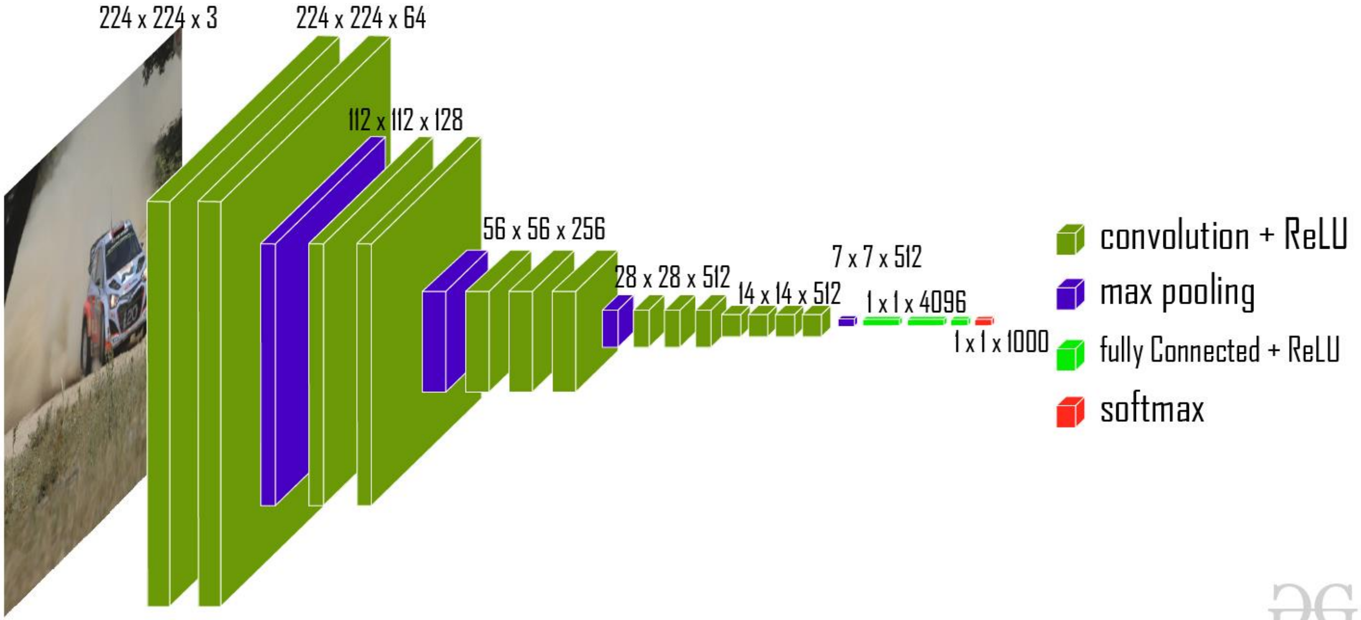


Image classification – case study

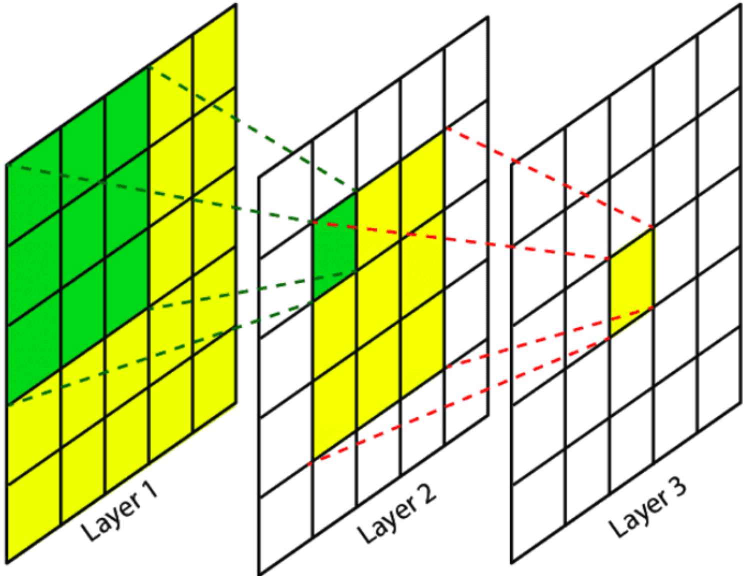
- VGG-Net architecture



- Max pooling is the so-called downsampling (crossing the image with a window of a certain size and selecting only pixels with the maximum value). This results in a reduction in the size of the image.
- In the case of the VGG network, the window size was 2x2.
- There are also other types of pooling, e.g. average pooling or min pooling.

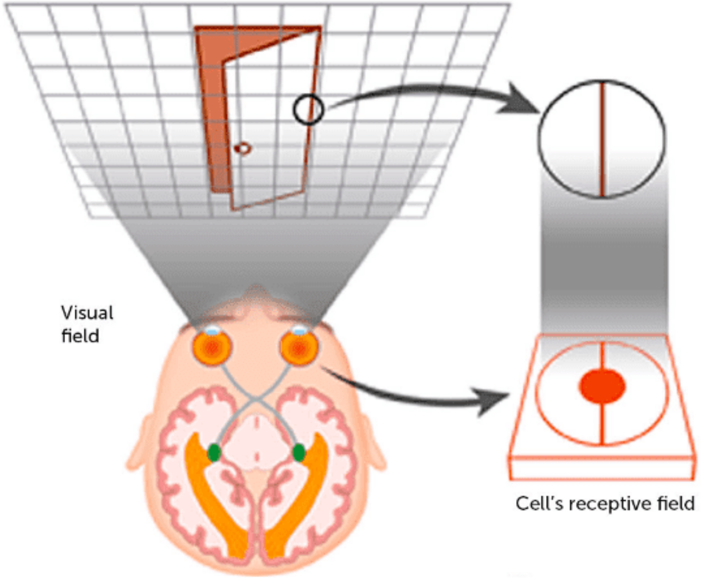
Receptive field

- In feedforward neural networks, the connections of neurons are peer-to-peer.
  In the case of convolutional networks, they are implemented in a slightly different way - by means of convolution (in the feature extractor).
- Convolutions are limited by the size of the window.



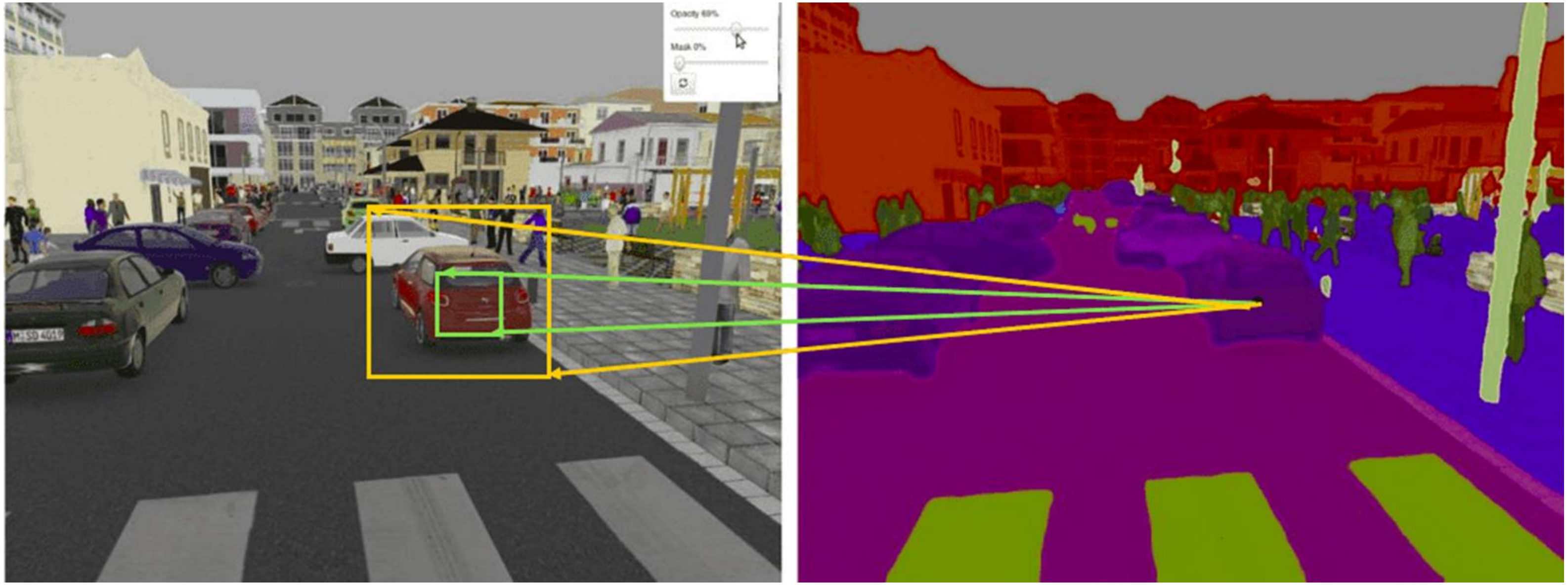
Receptive field

- In biology, the part of the stimulus to which a single neuron responds is called a receptive field.
- In convolutional layers, one filter of a given size equals one neuron. The size of its receptor field is proportional to the size of the window and inversely proportional to the size of the input image.



Receptive field

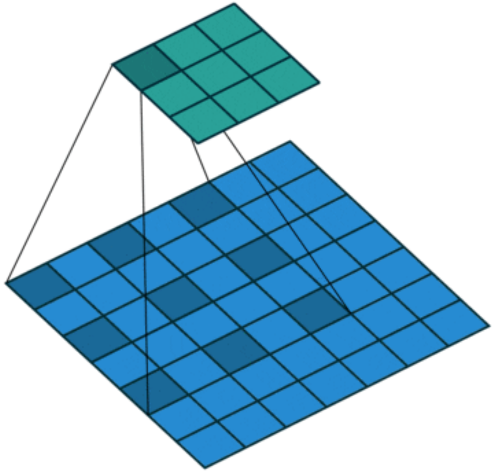
- The receptive field in the convolutional layers determines the size of the feature in the image, on the basis of which a single neuron will be activated!



Receptive field

- How to increase the receptive field in the CNN network?
  - Through larger windows in convolutions.
    This solution is not optimal due to the significant increase in computation time.
  - By increasing the number of convolutional layers.
    In this case, features of higher orders have a greater range, because they are "features of features".
  - By adding pooling layers or increasing strides in convolutions (image downsampling).
  - Through the use of so-called dilated convolutions

Dilated convolutions are convolutions with empty spaces inside that form a grid



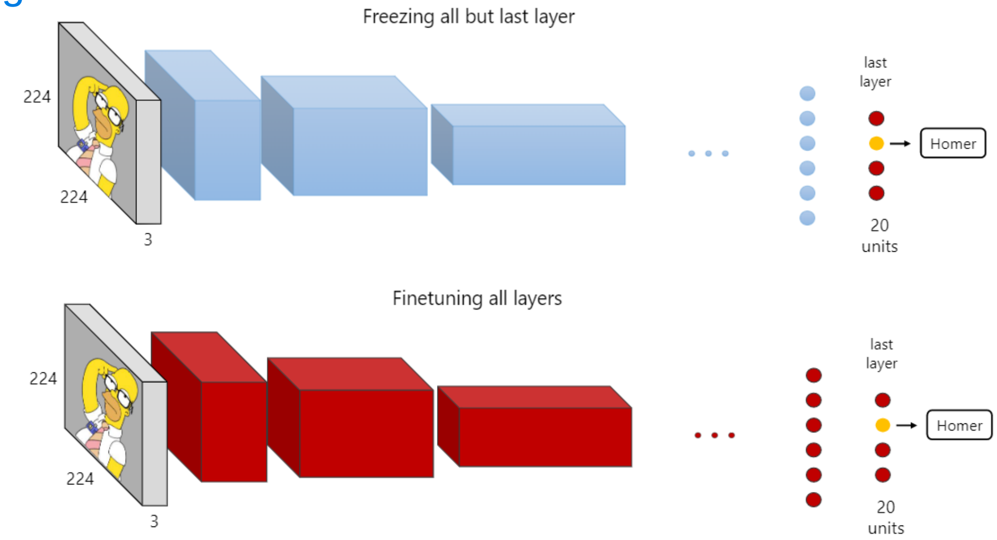
Transfer learning

- Resnet, vgg, and other ILSVRC competition-winning models are available off-the-shelf in Tensorflow Keras with the appropriate parameters.
- Assuming that some visual features that these models have learned to extract are common to many images, we can try to use them as a starting point for other tasks.
- Using the finished model from one task as a starting point for another task is known as transfer learning.

Transfer learning

This is usually done by:

- Selection of several convolutional layers from the finished network
- Setting trainable parameter to False.
- Adding new fully connected layers at the end.
- Training the prepared network.
- (Optional) Unlocking learning in convolutional layers and training the entire network with a low learning rate.
  This step is referred to as fine tuning.



Related topics

TensorFlow Core: Image classification

https://www.tensorflow.org/tutorials/images/classification

Image Recognition and Object Detection

https://learnopencv.com/image-recognition-and-object-detection-part1/