Machine Learning for Economists

Class 15: Convolutional Neural Network

葛雷

中国人民大学 - 数量经济

2025 年 5 月 15 日

中國人民大學
RENMIN UNIVERSITY OF CHINA

Why CNN of Images

CNN Model Framework

Key elements

Appendix

Why CNN of Images

CNN Model Framework

Key elements

Appendix

How Human Visual Neurons Works

- Previously we reduced [28,28] dimension picture to 28×28 features, then used a ANN for item prediction
- Certainly, it is not efficient
- We human perceive the item on the picture by shapes and patterns

Convolutional Neural Network

- Convolutional Neural Network (CNN) follow the same way as human visual neurons and can successfully perceive pictures
- Then what is the Convolutional Neural Network ?

Lower level patterns to high level patterns

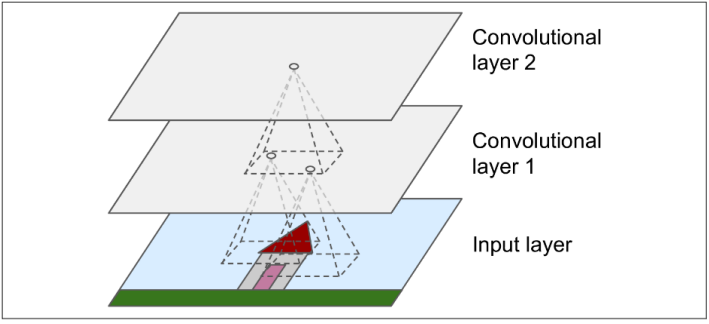


Figure 14-2. CNN layers with rectangular local receptive fields

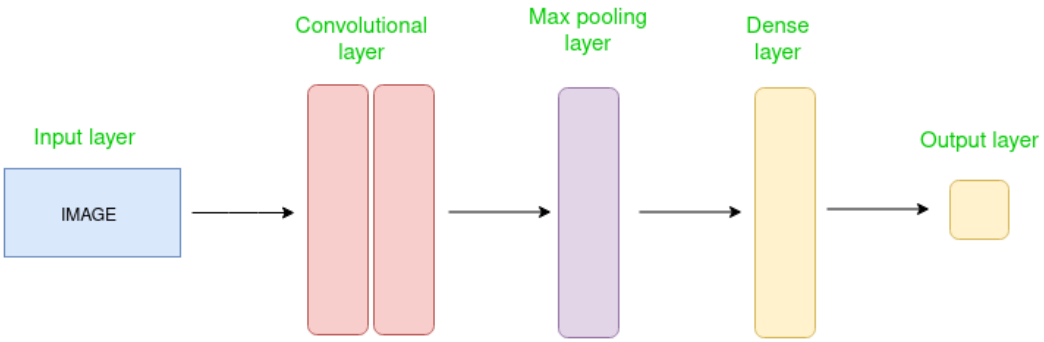
Why CNN of Images

CNN Model Framework

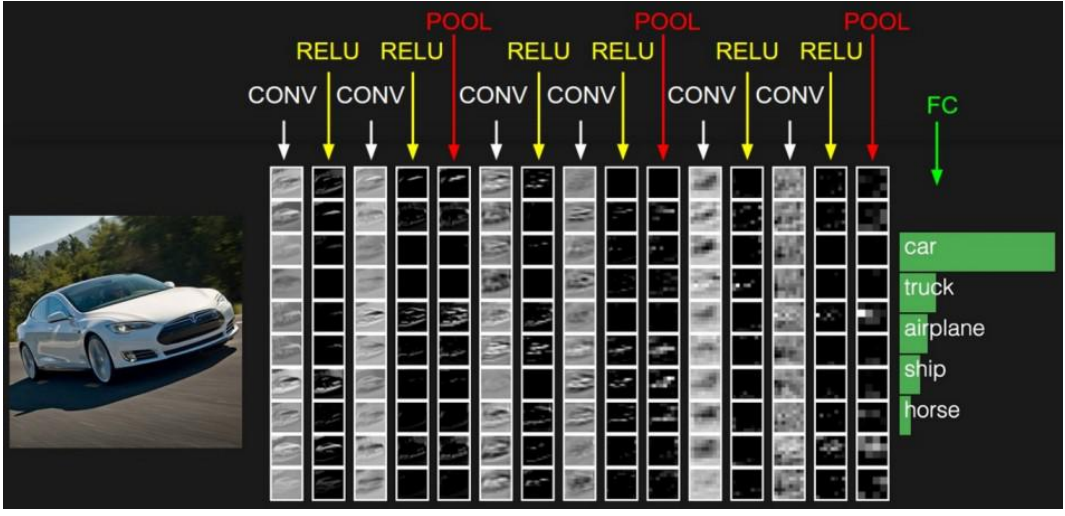
Key elements

Appendix

Whole Picture of CNN model



Whole Picture of CNN model



Codes in Keras

```
model = keras.models.Sequential([
    keras.layers.Conv2D(64, 7, activation="relu", padding="same",
                        input_shape=[28, 28, 1]),
    keras.layers.MaxPooling2D(2),
    keras.layers.Conv2D(128, 3, activation="relu", padding="same"),
    keras.layers.Conv2D(128, 3, activation="relu", padding="same"),
    keras.layers.MaxPooling2D(2),
    keras.layers.Conv2D(256, 3, activation="relu", padding="same"),
    keras.layers.Conv2D(256, 3, activation="relu", padding="same"),
    keras.layers.MaxPooling2D(2),
    keras.layers.Flatten(),
    keras.layers.Dense(128, activation="relu"),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(64, activation="relu"),
    keras.layers.Dropout(0.5),
    keras.layers.Dense(10, activation="softmax")
])
```

Why CNN of Images

CNN Model Framework

Key elements

Appendix

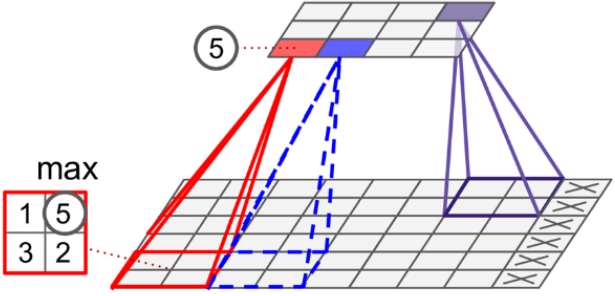
Key elements in CNN

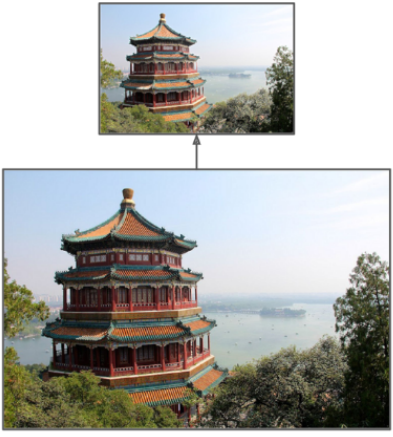
- Dense Layers
- RELU activation function
- Softmax in output for the classification
- Cross Entropy Loss

Key elements in CNN

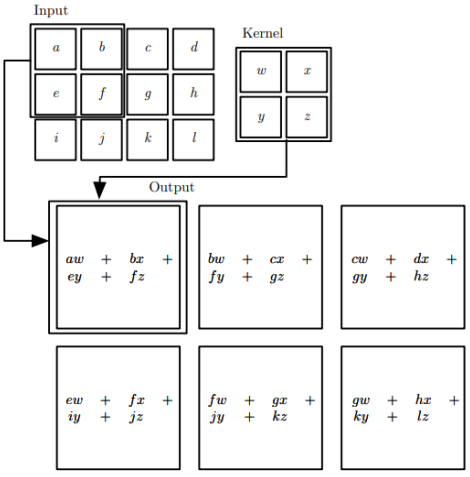
- Pooling Layers (new)
- Convolution Layers (new)
- Data Dimension 3D or 4D (batch_size, channels, height, width)

Pooling layers





Convolution Layers



Convolution Layers

- one filter (kernel) $\rightarrow$ one feature map
- However, each convolution layer has K filter, so it has K feature maps
- the weights on the filters are the parameters model needs to train in gradient descent
- Math details please refer to Equation 14-1 on Textbook

Put Conv,Pool,Output together

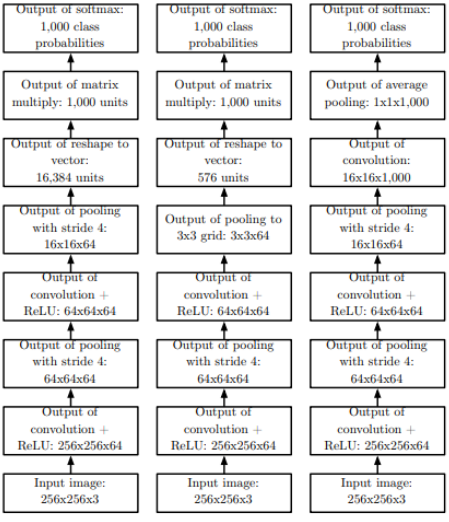


Figure 9.11

(Goodfellow 2016)

ANN dense connections, CNN sparse connection

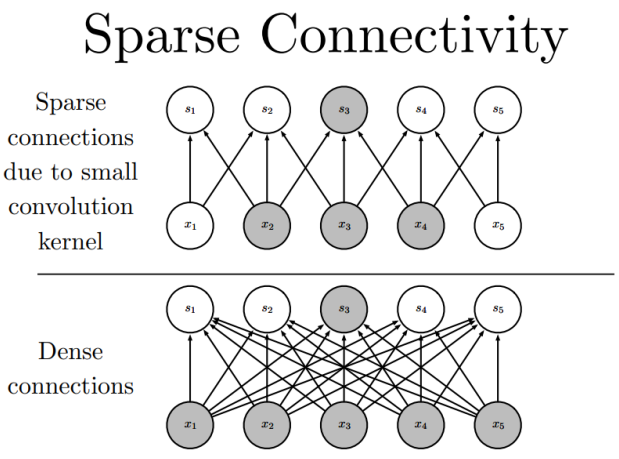
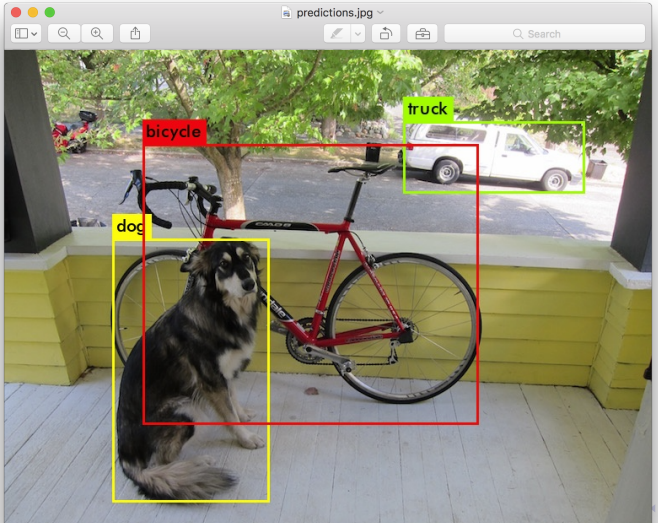


Figure 9.3

(Goodfellow 2016)

Popular package Yolo



Reference

1. Hands-on Machine Learning with Scikit-Learn, Keras and TensorFlow (3rd edition)
2. Deep Learning (2016), Ian Goodfellow and Yoshua Bengio and Aaron Courville
3. Wikipedia
4. geeksforgeeks
5. Kaggle
6. Wikipedia
7. ChatGPT
8. DeepSeek

Why CNN of Images

CNN Model Framework

Key elements

Appendix

Number of parameters in CNN

The total number of trainable parameters in a convolutional layer is given by:

$$\text{Total Parameters} = (K_H \times K_W \times C_{\text{in}} + 1) \times C_{\text{out}}$$

Where:

- K_H = Kernel height
- K_W = Kernel width
- C_{in} = Number of input channels
- C_{out} = Number of output channels (i.e., number of filters)
- The $+1$ accounts for the bias term per filter (optional)

Number of parameters in CNN (Why?)

- Why?
- Each input channel has its own kernel weights for a each output channel in a CNN layer.
- Optional HW: Check Equation 14-1 and understand why