Neural Network Architectures

Convolutional Neural Networks

Nayel Bettache

Introduction to CNNs

- Convolutional Neural Networks (CNNs) are specialized neural networks designed for image and spatial data.
- They consist of three main types of layers: convolutional layers, pooling layers, and fully connected layers.
- Key applications include image classification, object detection, and natural language processing.

Key Components of CNNs

- Convolutional Layers: Extract spatial features using filters.
- Activation Functions: Introduce non-linearity (e.g., ReLU).
- Pooling Layers: Reduce spatial dimensions while preserving features.
- Fully Connected Layers: Perform classification or regression tasks.

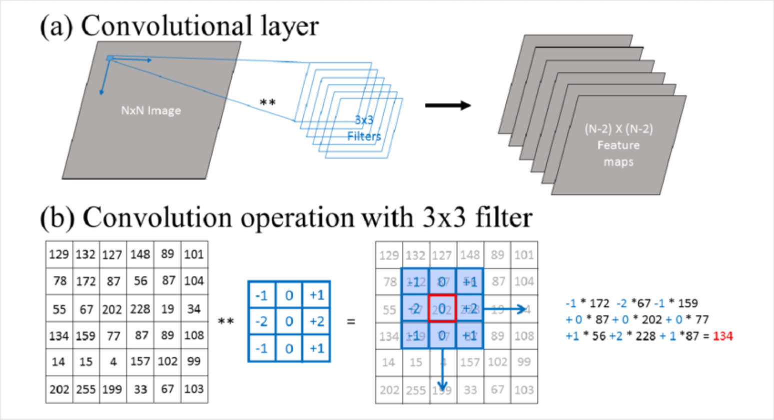
Convolutional Layers - Overview

- The convolutional layer applies a set of filters (kernels) to the input image.
- Each filter extracts a specific feature (e.g., edges, textures).
- Output: A set of feature maps that highlight spatial patterns in the data.

$$Y(i,j) = \sum_{m=1}^{M} \sum_{n=1}^{N} X(i+m, j+n) \cdot K(m,n) \tag{1}$$

Convolutional Layers - Example

- Input: 6x6 image. Filter: 3x3 kernel
- Stride: step size with which the conv. filter moves.
- Padding: adding extra pixels (typically zeros) around the edges of the input data before applying the convolution filter.



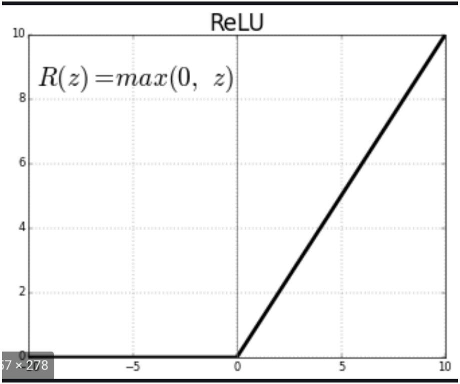
Padding and Stride

- Padding: Adds a border of zeros around the input to preserve spatial dimensions.
- Stride: Controls the step size of the convolution operation.
- Larger strides result in smaller output dimensions.

$$\text{Output size} = \frac{\text{Input size} - \text{Filter size} + 2 \cdot \text{Padding}}{\text{Stride}} + 1 \tag{2}$$

Activation Functions

- Commonly used activation function: ReLU (Rectified Linear Unit)
- Formula: $f(x) = \max(0, x)$
- Introduces non-linearity, enabling CNNs to learn complex patterns.

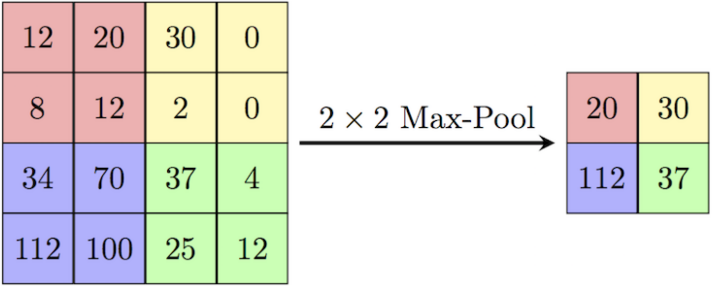


Max Pooling Layers - Overview

- Pooling layers reduce the spatial dimensions of feature maps.
- Max pooling selects the maximum value in a region (e.g., 2x2).
- Helps reduce computation and overfitting by preserving dominant features.

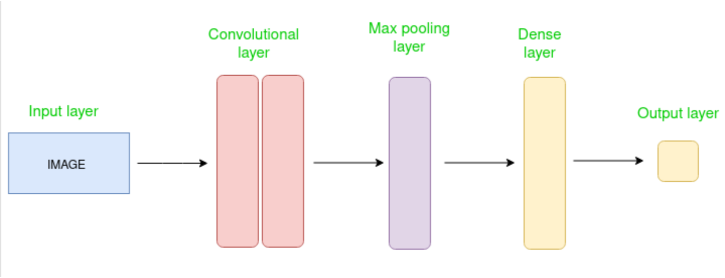
Max Pooling - Example

- Input: 4x4 feature map
- Pool size: 2x2
- Stride: 2



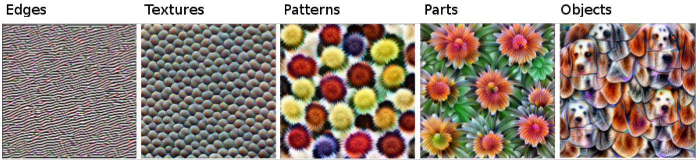
CNN Architectures - Stacking Layers

- Convolutional layers and pooling layers are stacked to form deep networks.
- Example: VGG16, ResNet, Inception.
- Each layer extracts increasingly complex features.



Visualization of Features

- Early layers learn simple features like edges and corners.
- Deeper layers capture complex patterns like textures and objects.



Conclusion

- CNNs leverage convolutional and pooling layers to efficiently process spatial data.
- Max pooling helps reduce dimensions while preserving key features.
- Future work includes exploring advanced architectures like transformers in vision tasks.