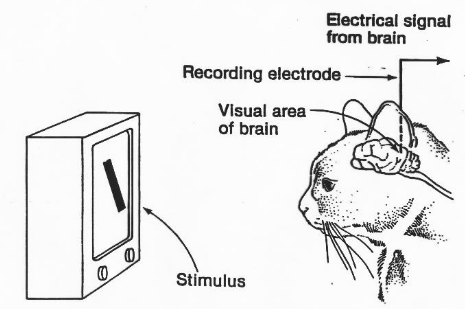
Convolutional Neural Networks

Dit-Yan Yeung

Department of Computer Science and Engineering
Hong Kong University of Science and Technology

COMP 4211: Machine Learning (Fall 2022)

Inspirations from the Visual Cortex

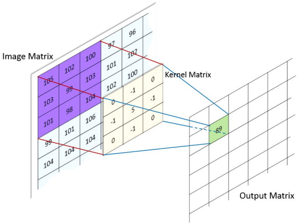


- The neocognitron model and later the convolutional neural network (CNN or ConvNet) model were inspired by Hubel and Wiesel's study on the structure of the visual cortex of cats and monkeys in the 1950s and 1960s.
- Hubel and Wiesel's work led to a Nobel Prize in 1981.

Inspirations from the Visual Cortex (2)

- Many neurons in the visual cortex have a small local receptive field reacting only to visual stimuli from a small region.
- Some neurons react to images of lines with different orientations.
- Some neurons have larger receptive fields and react to more complex patterns that are combinations of the simpler, lower-level patterns.
- These observations led to the layered model of the visual cortex in that higher-level neurons are based on the outputs of neighboring lower-level neurons.
- The CNN model is a special type of feedforward neural networks with two new building blocks: convolutional layers and pooling layers.

2D Convolution



- Convolution is a linear operation which applies a kernel to an image:
  - Image matrix: a channel in the previous layer
  - Output matrix: a feature map in a convolutional layer
  - Kernel matrix: connection weights between layers corresponding to the convolution kernel or filter of the feature map (each filter also has a bias term which is not shown here for simplicity).
- In order for a layer (output matrix) to have the same size as the previous layer (image matrix), zero padding is applied to add zeros around the inputs.

2D Convolution (2)

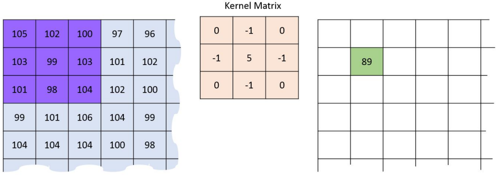


Image Matrix

Output Matrix

$$105*0+102*-1+100*0 \\ +103*-1+99*5+103*-1 \\ +101*0+98*-1+104*0 = 89$$

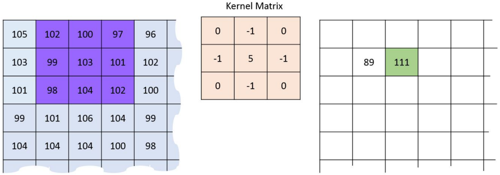


Image Matrix

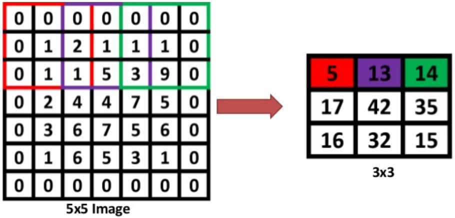
Output Matrix

$$102*0+100*-1+97*0 \\ +99*-1+103*5+101*-1 \\ +98*0+104*-1+102*0 = 111$$

Convolutional Layer vs. Fully Connected Layer

- Each unit in a convolutional layer is only connected to units in a local receptive field of the previous layer;
  Each unit in a fully connected layer is connected to all units in the previous layer.
- The weights of a convolution kernel are shared by all units in a feature map of a convolutional layer;
  The weights between two fully connected layers are not shared.
- Implications of weight sharing:
  - The number of model parameters is dramatically reduced.
  - Once the network has learned to recognize a pattern in one location, it can recognize it in any other location.

Stride



- We can connect a larger layer to a smaller layer by spacing out the receptive fields.
- The distance between two consecutive receptive fields is called the stride.
- The stride can be different in the two directions.

An Illustrative Example

- Consider two layers of a CNN:
  - Convolutional layer: 200 feature maps of size 150×100; 5×5 filters; stride 1
  - Input layer: 3 channels of RGB image of size 150×100
- Total number of parameters:

$$(5 \times 5 \times 3 + 1) \times 200 = 15{,}200.$$

- As a comparison, a fully connected layer with 150×100 units, each connected to all $150 \times 100 \times 3$ inputs, would have many more parameters:

$$(150 \times 100 \times 3 + 1) \times 150 \times 100 = 675{,}015{,}000.$$

A General Formula

- Given:
  - Kernel size of a convolutional layer: $f \times f$
  - Length of one side of the input feature map: n
  - Padding: p
  - Stride: s
- Length of corresponding side of the convolutional layer:

$$\left\lfloor \frac{n + 2p - f}{s} + 1 \right\rfloor .$$

- To ensure that the feature map in a convolutional layer is of the same size as the input feature map, we can choose p such that

$$\left\lfloor \frac{n + 2p - f}{s} + 1 \right\rfloor = n.$$

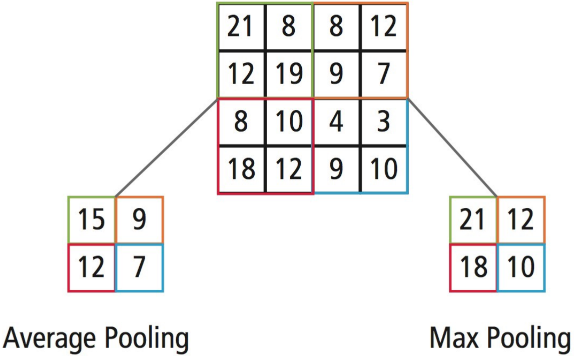
Multiple Feature Maps

- Each feature map, which corresponds to a convolution kernel or filter, highlights the areas in an image that are most similar to the filter.
- Each convolutional layer has multiple feature maps of the same size corresponding to different filters.
- There is only weight sharing within a feature map but not between different feature maps.
- A convolutional layer simultaneously applies multiple filters to all the feature maps of the previous layer.
- During training, the network learns the most useful filters by learning their weights and learns to combine them into more complex patterns in higher layers.

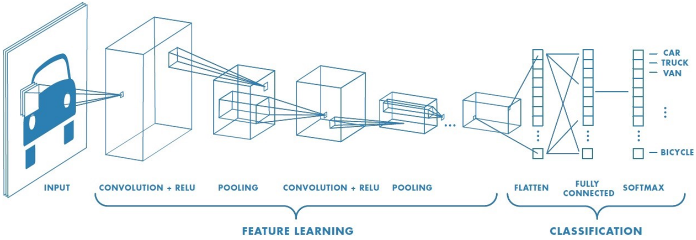
Pooling

- A pooling layer subsamples the input image to give a smaller image.
- Consequences of the pooling operation:
  - It reduces the computational load, memory usage, and number of parameters.
  - It can tolerate local distortion and translation of patterns of interest.
- Like the convolutional layers, each unit in a pooling layer is connected to some units in the previous layer within a local receptive field, with specified size and stride.
- Unlike the convolutional layers, a pooling layer has no weights but it has an aggregation function such as the maximum (for max pooling) or mean (for average pooling).
- A pooling layer typically works on every input channel independently.

Average Pooling and Max Pooling



Typical CNN Architectures



- Typical CNN models for classification tasks stack different types of layers as follows:
  - A few convolutional layers (each usually followed by a ReLU layer)
  - Then a pooling layer
  - Then a few convolutional layers (+ReLU) and a pooling layer, and so on
  - Then one or more fully connected layers (+ReLU)
  - Then a final fully connected softmax layer.

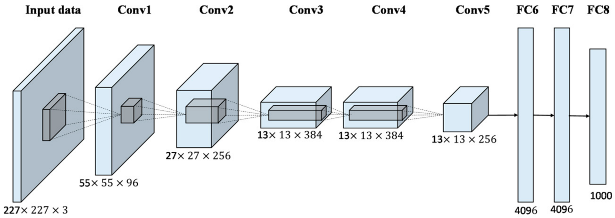
Typical CNN Architectures (2)

- In general, the convolutional and pooling layers are for feature learning and the fully connected layers are for classification.
- Going through the convolutional layers, the feature maps typically get smaller but deeper (i.e., more feature maps per layer).
- The fully connected layers form an ordinary feedforward neural network to which the transformed inputs are fed.
- Instead of using a convolutional layer with a large kernel size (e.g., 9×9 with 81 parameters), it is better to stack two convolutional layers with smaller kernels (e.g., 3×3 with 18 parameters for the two layers together).

Typical CNN Architectures (3)

- Like other feedforward neural networks, the BP algorithm based on SGD or variants can be used for training CNNs.
- Some popular CNN architectures:
  - LeNet-5 (1998)
  - AlexNet (winner of ILSVRC 2012)
  - VGGNet (winner of ILSVRC 2014)
  - GoogLeNet (winner of ILSVRC 2014)
  - ResNet (winner of ILSVRC 2015)

AlexNet



- AlexNet won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012 by a large margin: 15.3% top-5 error rate compared to 26.1% for the second best.
- Feature learning layers ('conv' for convolutional layer):
  conv – max-pooling – conv – max-pooling – conv – conv – conv
- Classification layers ('fc' for fully connected layer):
  fc – fc – fc

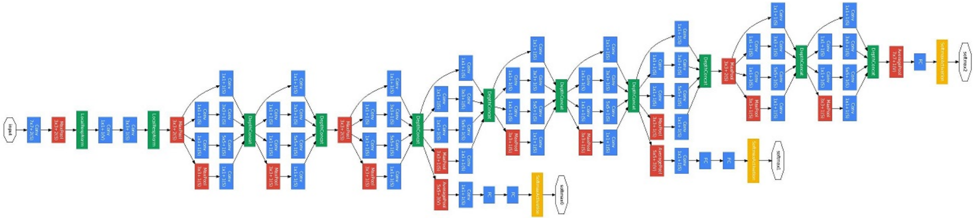
AlexNet (2)

- Two regularization techniques are used:
  - Dropout regularization (on first two fully connected layers)
  - Data augmentation
- A variant of AlexNet, called ZF Net (developed by Zeiler and Fergus), also performed well in ILSVRC 2013.

VGGNet

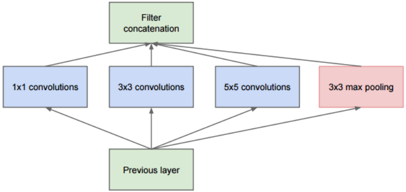
- In ILSVRC 2014, VGGNet had the lowest localization error and second lowest classification error.
- Compared with AlexNet, VGGNet has a simpler architecture but is deeper (16 or 19 layers).
- VGGNet has many parameters and is very slow to train.
- Pre-trained VGGNet models are commonly used as feature extractors, with or without fine-tuning, for many computer vision tasks.

GoogLeNet

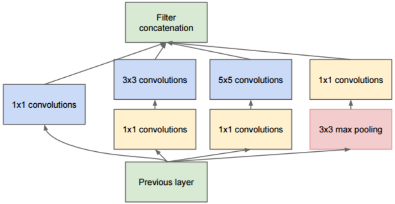


- GoogLeNet won ILSVRC 2014 by pushing the top-5 error rate down to 6.7%.
- It has a much deeper CNN architecture consisting of nine subnetworks called inception modules that can use parameters more efficiently (10 times fewer parameters than AlexNet).

Inception Module



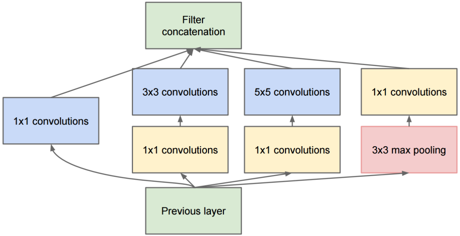
(a) Inception module, naïve version



(b) Inception module with dimension reductions

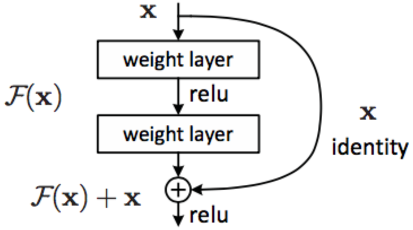
- Kernels of different sizes (1×1, 3×3, 5×5) are used to capture patterns at different scales.
- All four layers are of the same size, so they can be concatenated in the concatenation layer.

1×1 Kernels in Inception Modules



- They output many fewer feature maps than their inputs to reduce dimensionality, which is particularly useful before applying the computationally expensive 3×3 and 5×5 convolution operations.
- Each pair of convolutional layers ($[1 \times 1, 3 \times 3]$ or $[1 \times 1, 5 \times 5]$) acts like a single, powerful convolutional layer capable of capturing more complex patterns.

ResNet



- The residual network (ResNet) won ILSVRC 2015 with a top-5 error rate under 3.6%, which is better than human performance (about 5%).
- It has an extremely deep architecture with 152 layers.
- Skip connections (a.k.a. shortcut connections) are introduced to make it possible to train such a deep network.

ResNet (2)

- A small neural network with a skip connection is called a residual unit.
- Each residual unit is composed of two convolutional layers using 3×3 kernels with batch normalization and ReLU.
- ResNet's architecture is quite simple, starting and ending like GoogLeNet, with a deep stack of simple residual units in between.

CNN Architectures after ResNet

- Xception
- ResNeXt
- DenseNet
- EfficientNet
- NFNet (Normalizer-Free ResNet, proposed by Google DeepMind in Feb 2021)
- ...

To Learn More...

- Graph convolutional networks