

Introduction to Convolutional Neural Networks

FPT Technology Research Institute

FPT University

Pham Quang Nhat Minh

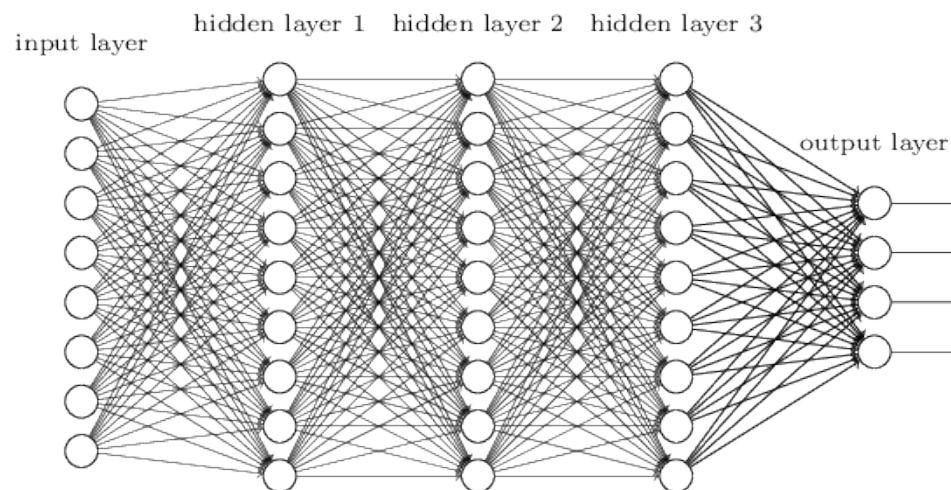
December 4, 2016

Contents

- Ý tưởng cơ bản của Convolutional Neural Networks
- Convolutional Neural Networks trong thực hành
- Các thư viện lập trình hỗ trợ CNN
- Các ứng dụng cơ bản của CNN
- Tài liệu tham khảo

Ý tưởng cơ bản của Convolutional Neural Networks

- Nhược điểm khi dùng mạng neural với fully-connected layers cho bài toán phân lớp ảnh
 - Không khai thác được thông tin không gian về cấu trúc của ảnh (ví dụ các điểm ảnh ở cách rất xa nhau cũng được xử lý giống với các điểm ảnh ở cạnh nhau)
- Convolutional Neural Networks (CNN) với cấu trúc đặc biệt có thể tận dụng được thông tin về không gian trong ảnh
 - Huấn luyện mạng Neural nhanh hơn nên có thể huấn luyện được mạng Neural sâu với nhiều tầng hơn
 - CNN rất hiệu quả trong phân lớp ảnh và được ứng dụng nhiều trong lĩnh vực thị giác máy tính (Computer Vision)

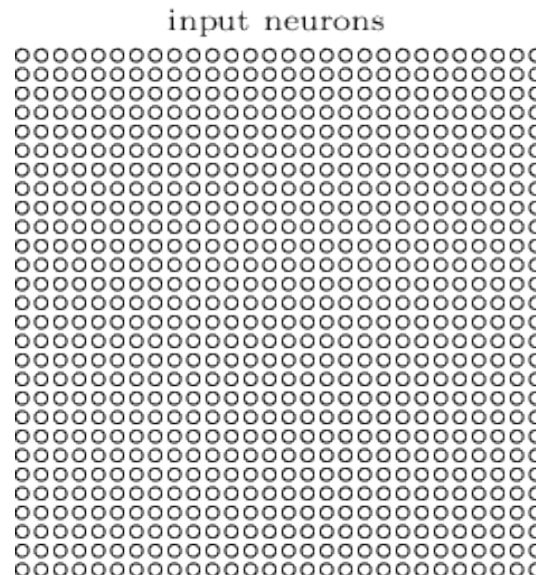


Ý tưởng cơ bản của Convolutional Neural Networks

- Có 3 ý tưởng cơ bản trong Convolutional Neural Networks
 - *local receptive fields*
 - *shared weights*
 - *pooling*

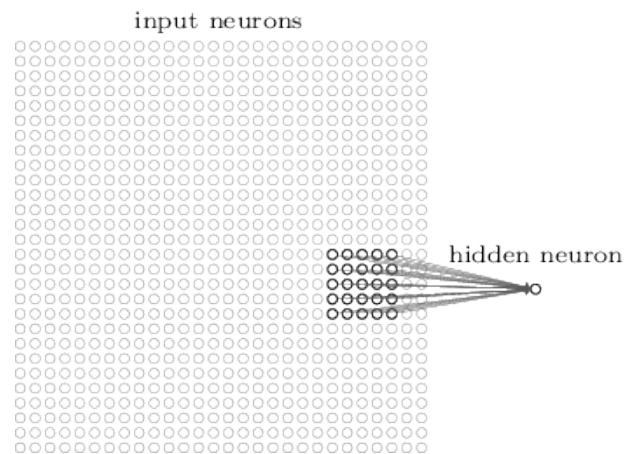
Local receptive fields

- Biểu diễn đầu vào của CNN là một hình vuông kích thước 28 x 28
- Chúng ta sẽ cần nối các điểm đầu vào (input pixels) tới một tầng ẩn trong mạng Neural
- Trong mạng neural thường, chúng ta sẽ kết nối mọi điểm đầu vào tới mọi neural trong tầng ẩn.
- Trong CNN, chúng ta sẽ kết nối mỗi khu vực nhỏ trong ảnh tới các neural trong tầng ẩn.



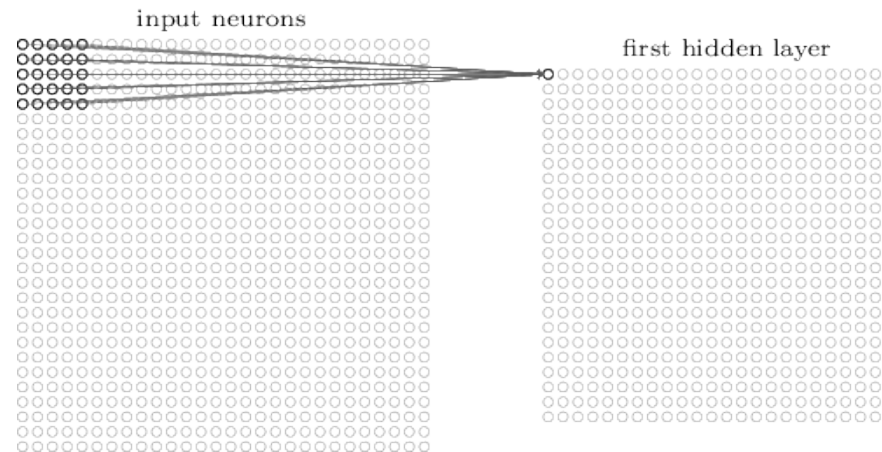
Local receptive fields

- Mỗi neural trong tầng ẩn đầu tiên sẽ kết nối với một vùng nhỏ của input neural.
- Vùng trong ảnh đầu vào được gọi là local receptive fields



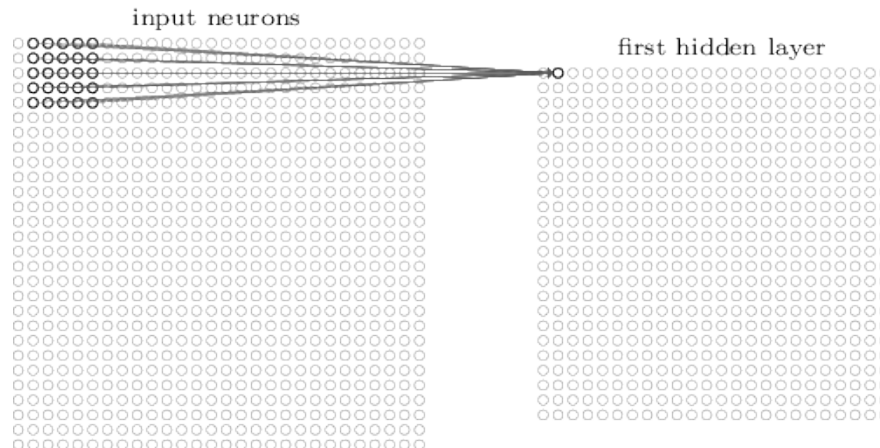
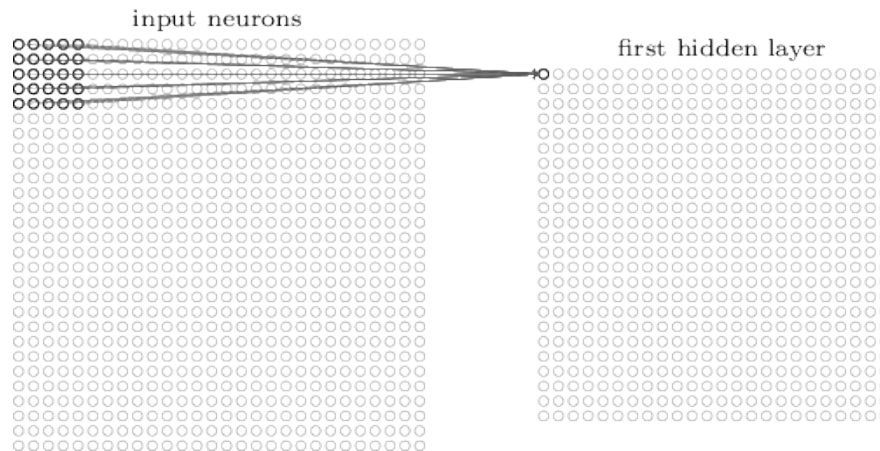
Local receptive fields

- Mỗi neural trong tầng ẩn đầu tiên sẽ kết nối với một vùng nhỏ của input neural.
- Vùng trong ảnh đầu vào được gọi là local receptive fields
- Mỗi kết nối sẽ học một trọng số
- Sau đó chúng ta sẽ dịch chuyển các local receptive fields dọc theo toàn bộ ảnh



Local receptive fields

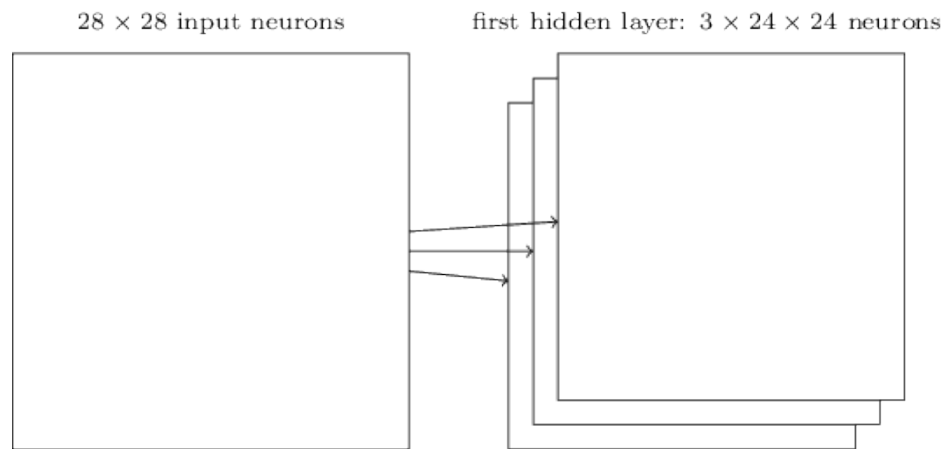
- Vùng trong ảnh đầu vào được gọi là local receptive fields
- Mỗi kết nối sẽ học một trọng số
- Sau đó chúng ta sẽ dịch chuyển các local receptive fields dọc theo toàn bộ ảnh



Shared weights and biases

- Each hidden neuron has a bias and 5 x 5 weights
- We use the same weights and bias for each of the 24 x 24 hidden neurons.

$$\sigma \left(b + \sum_{l=0}^4 \sum_{m=0}^4 w_{l,m} a_{j+l,k+m} \right).$$

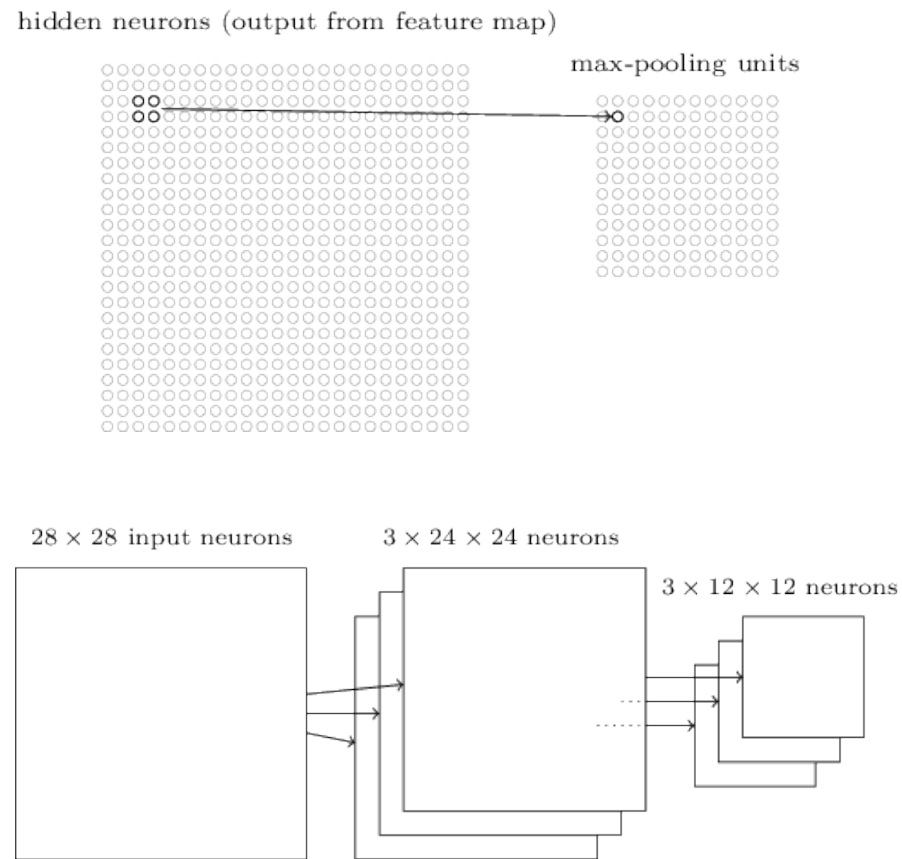


Pooling Layers

- Pooling Layers thường được dùng ngay sau convolutional layers
- Pooling Layers đơn giản hoá thông tin trong output của convolutional layers
- Input của Pooling Layers là output của mỗi feature map ở tầng convolutional
- Output của Pooling Layers là một feature map “cô đặc” hơn
- Chẳng hạn Pooling Layers có thể sử dụng phương pháp **max-pooling**

Max-Pooling

- Ví dụ: Mỗi unit trong Pooling layer sẽ “tóm tắt” (summarize) một khu vực (chẳng hạn các neurons 2×2 ở tầng trước đó).
- Từ đầu ra bao gồm 24×24 neurons của tầng trước, sau bước pooling chúng ta sẽ có 12×12 neurons.



Ý nghĩa của tầng Pooling & Tại sao tầng Pooling hữu ích

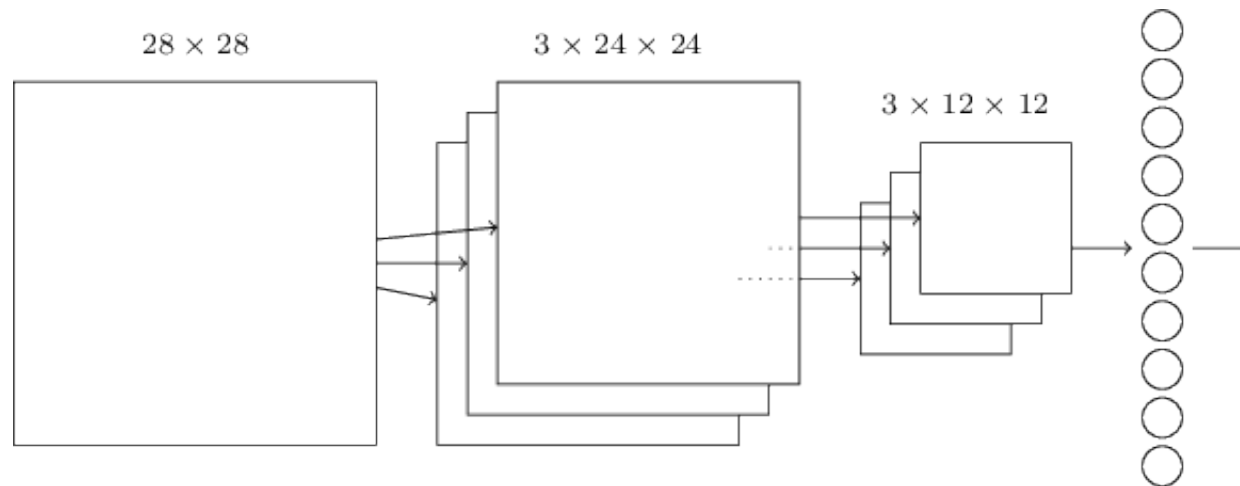
- Có thể tưởng tượng phương pháp max-pooling giống như một cách để xác định xem một feature cho trước được tìm thấy hay không ở vùng nào đó của ảnh.
- Phương pháp Pooling sẽ giúp giảm đi số parameters cần thiết cho các tầng tiếp theo.

Các phương pháp Pooling khác

- Max-Pooling không phải là một phương pháp duy nhất được dùng trong Pooling Layers
- L2 Pooling cũng là một cách phổ biến
- Thay vì tính max trong mỗi vùng (2 x 2) trong output của tầng convolutional, lấy căn bậc 2 của tổng bình phương của các giá trị activation trong vùng đó.
- Việc dùng phương pháp pooling nào tốt hơn được xác định thông qua thực nghiệm

Kết hợp các thành phần với nhau thành mạng CNN

- Mô hình mạng CNN cho nhận dạng chữ viết tay (MNIST) có thể như ở dưới đây.
- Tầng cuối của mạng CNN là tầng kết nối đầy đủ (fully-connected layer)
 - Kết nối mọi neuron trong tầng pooling với các neuron trong tầng output (10 output neurons)



Backpropagation trong convolutional neural networks

- Thuật toán backpropagation để tính gradient chúng ta đã học áp dụng cho mạng neural với các tầng được kết nối đầy đủ (fully-connected layers)
- Đối với mạng CNN, chúng ta cần chỉnh sửa lại thuật toán backpropagation
- Chi tiết về thuật toán backpropagation xem tại:
<http://neuralnetworksanddeeplearning.com/chap6.html>

Convolutional neural networks trong thực hành

- Xem phần “Convolutional neural networks in practice” trong chương 6 của cuốn sách “Neural Networks and Deep Learning” (Michael Nielsen)
 - http://neuralnetworksanddeeplearning.com/chap6.html#convolutional_neural_networks_in_practice
- Trang Github tại:
 - <https://github.com/mnielsen/neural-networks-and-deep-learning/blob/master/src/network3.py>

Zero Padding

- Thêm các giá trị 0 xung quanh border (của ảnh)
- Zero-padding cho phép chúng ta kiểm soát kích thước của output
- Zero-padding cũng cho phép chúng ta “bảo vệ” (preserve) kích thước của input.

Các thư viện lập trình hỗ trợ CNN

- DyNet: <https://github.com/clab/dynet>
- Caffe: <http://caffe.berkeleyvision.org/>
- Keras: <https://keras.io>
- Chainer: <http://chainer.org/>

Các ứng dụng của CNN

- CNN chủ yếu được sử dụng trong Computer Vision (nhận dạng ảnh, etc)
 - Nhận dạng chữ viết tay (LeNet)
 - Nhận dạng ảnh (ImageNet)
 - ...
- CNN cũng được dùng trong xử lý ngôn ngữ tự nhiên, nhưng không phổ biến bằng Recurrent Neural Networks (RNN)
 - Do tính chất của mạng CNN hỗ trợ dữ liệu dạng grid (2D hay 3D) trong khi RNN hỗ trợ tốt hơn dữ liệu dạng chuỗi (thích hợp hơn cho text)

Tài liệu tham khảo

- Chương 6 “Deep Learning”, sách “Neural Networks and Deep Learning”
 - <http://neuralnetworksanddeeplearning.com/chap6.html>
- Chương 9 - Convolutional Networks, sách Deep Learning
 - www.deeplearningbook.org/contents/convnets.html