第 6 回　知能システム学特論レポート

15344203　有田 裕太
15344206　緒形 裕太
15344209　株丹 亮
12104125　宮本 和

西田研究室，計算力学研究室

2015 年 7 月 6 日

進捗状況

理論研究の進捗

畳込みニューラルネットワークの理論について

プログラミングの進捗

プログラム実行環境の見直し
データセット作成

畳み込みの定義

- 濃淡地を各画素に格納したグレースケールの画像を考える.
- 画像サイズを $W \times W$ 画素, 画素をインデックス $(i, j)(i = 0, \cdots, W-1, j = 0, \cdots, W-1)$ 画素値を x_{ij}, フィルタサイズを $H \times H$ 画素, 画素をインデックス $(p, q)(p = 0, \cdots, H-1, q = 0, \cdots, H-1)$, 画素値を h_{pq} とすると,

$$u_{ij} = \sum_{p=0}^{H-1} \sum_{q=0}^{H-1} x_{i+p,j+q} h_{pq} \tag{1}$$

- この計算はフィルタが表す特徴的な濃淡構造を，画像から抽出する「特徴抽出」の働きがある.

畳み込みの働き

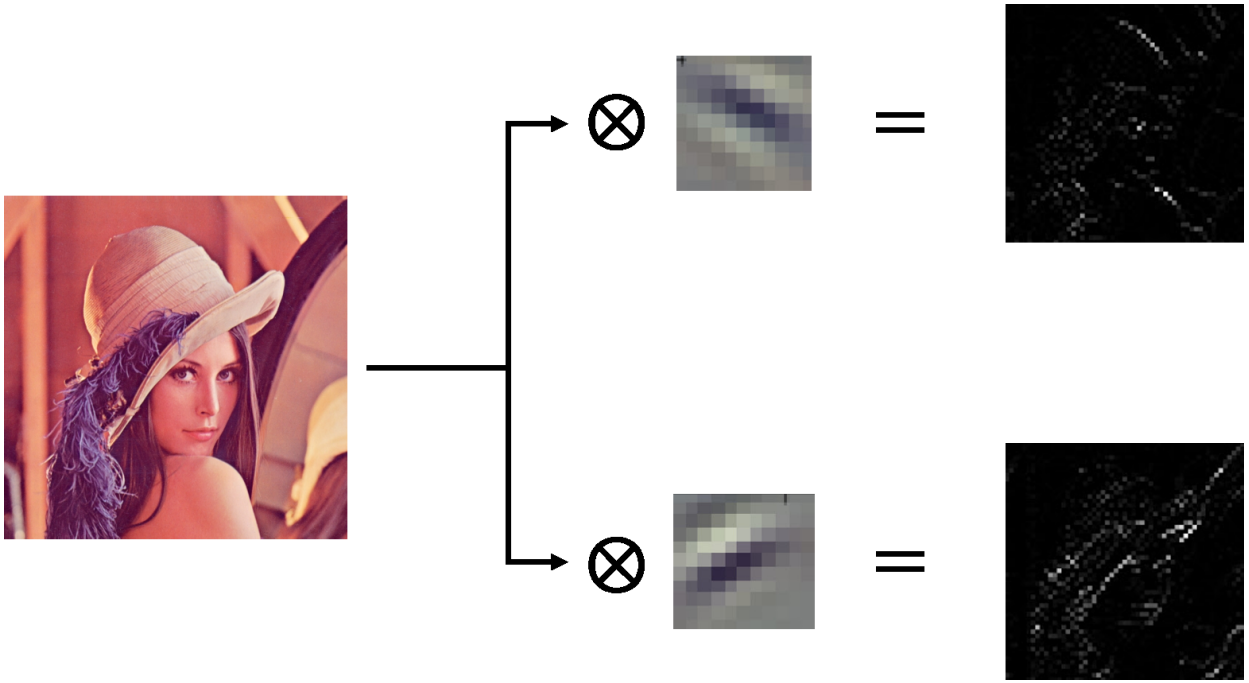


Figure: 画像の畳み込みの例

- 入力画像：227 × 227
- フィルタ：11 × 11
- 出力画像：55 × 55
- ストライド：4

パディング

- 畳込みは画像にフィルタを重なり合う画素どおしの積を求めて，フィルタ全体の和を求める.
- 画像内にフィルタ全体が収まる範囲内でフィルタを動かすと，畳込み処理を行った後の画像サイズは入力画像は小さくなる.

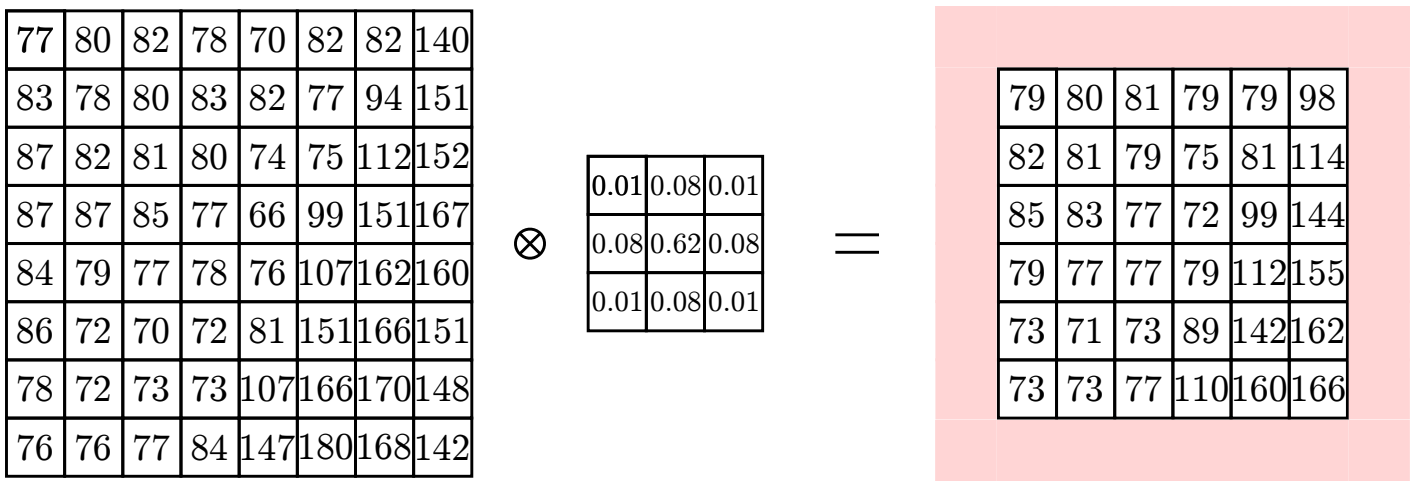


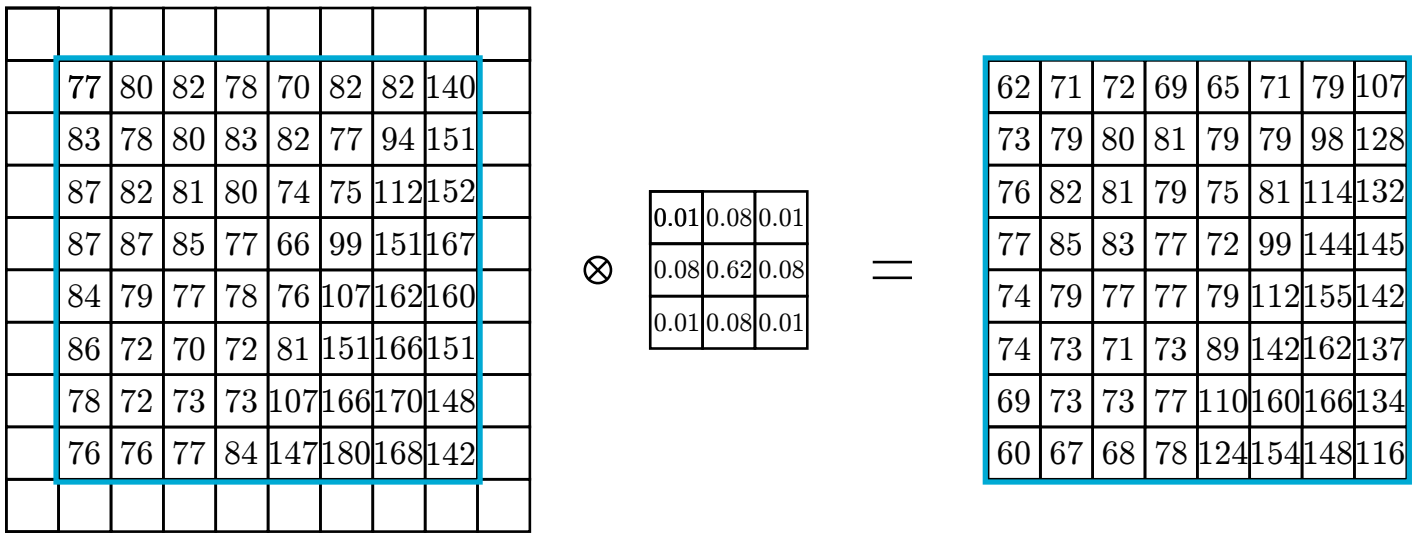
Figure: 8×8 の入力画像を畳込み処理した場合の出力画像のサイズ縮小の様子

パディング

このときの画像サイズは

$$\left(W - 2\left\lfloor\frac{H}{2}\right\rfloor\right) \times \left(W - 2\left\lfloor\frac{H}{2}\right\rfloor\right) \tag{2}$$

- 一方で畳込みの結果が入力画像と同サイズに出力する場合，入力画像の外側に幅 $\lfloor H/2 \rfloor$ の余剰を設け，出力画像が元の入力画像と同サイズになるようにする.
- 余分に設けた部分の画素値を 0 とする方法をゼロパディング（zero-padding）と呼ぶ.



ストライド

ストライド

フィルタの適用位置を 1 画素ずつではなく，数画素ずつずらして計算する．このずらす間隔をストライド (stride) という．

ストライドを s とするとき，出力画像の画素値は

$$u_{ij} = \sum_{p=0}^{H-1} \sum_{q=0}^{H-1} x_{si+p,sj+q} h_{pq} \tag{3}$$

サイズは

$$([(W-1)/s]+1) \times ([(W-1)/s]+1) \tag{4}$$

- 畳み込み層の出力側のユニット数が大きくなりすぎるのを防ぐために，2 以上のストライドが使われることがある
- ストライドを大きくすることは画像特徴を取りこぼすことを意味し，性能を悪化させる可能性

GPU を用いた学習実行時の演算処理高速化

- 学習データがあまり多くない場合でも学習を行うことができるファインチューニングを試験的に行った
- ファインチューニングは大規模なデータセットで学習済みの状態から目的とする別のデータセットへ学習し直す方法
- CPU のみの演算だと時間がかかりすぎる

そこで...

GPU による並列計算

- caffe がサポートしている NVIDIA の CUDA を使う（導入済み）
- Deep Learning 用の CUDA ライブラリ cuDNN を使用する（デベロッパー登録申請中）

今後の課題

理論研究

CNN の詳細な調査
プーリングの理論

プログラミング

データセットの作成