

14.1 視覚野のアーキテクチャ

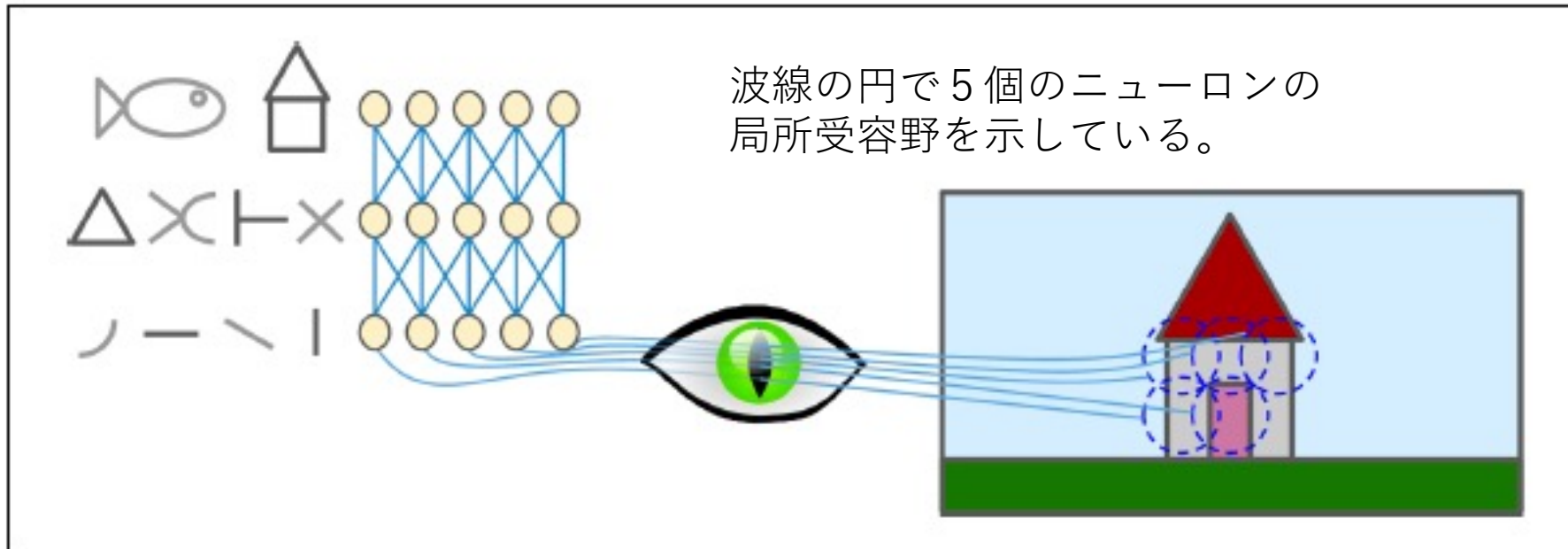


図14-1 視覚野のなかの生物学的ニューロンは、視野のなかの受容野と呼ばれる小さな領域に含まれる特定のパターンだけに反応する。視覚信号は脳内の連続したモジュールを通過し、ニューロンは次第に大きな受容野のなかの複雑なパターンに反応するようになる

視覚野の研究（1958）→ネオコグニトロン（1980）→Le Net-5アーキテクチャ（1988）
（畳み込み層を導入）

14.2 畳み込み層

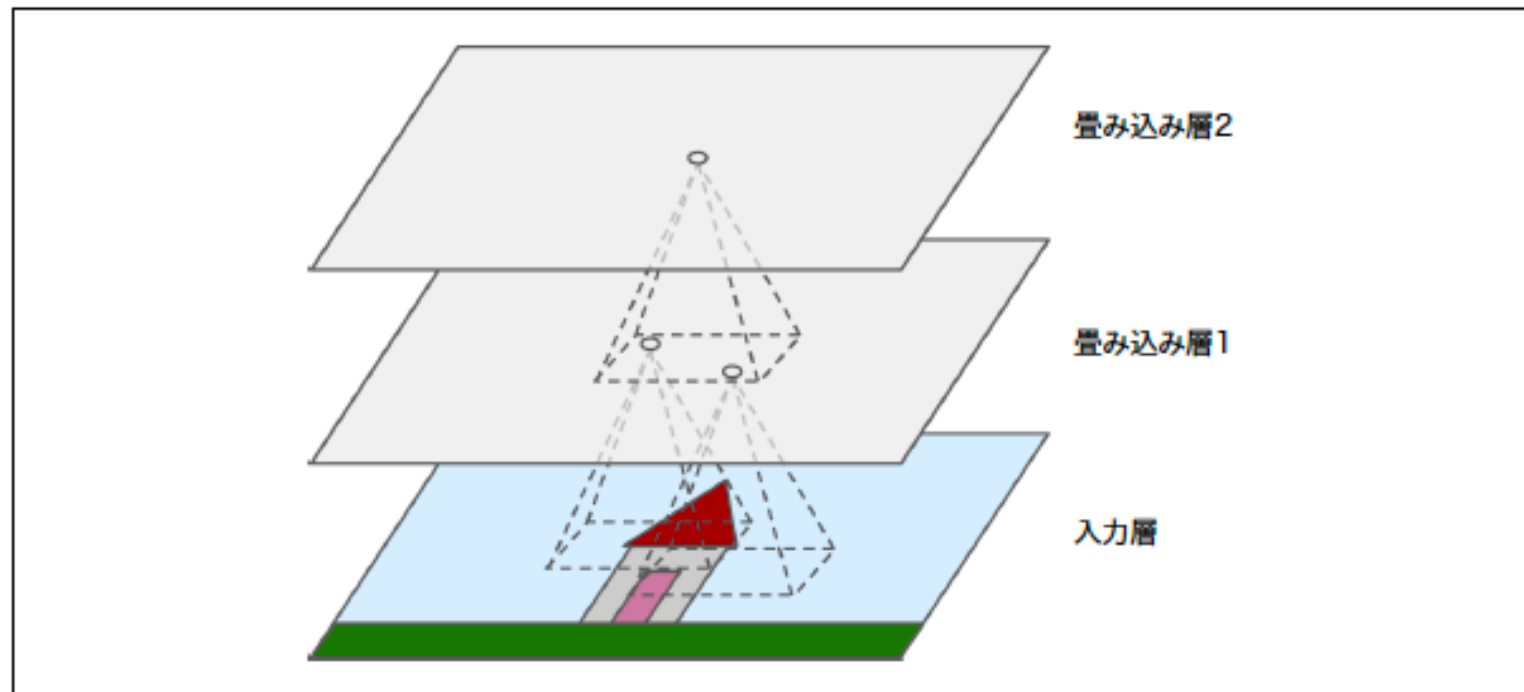


図14-2 長方形の局所受容野を持つCNNの階層構造

畳み込み層（**convolutional layer**）は、**CNN**でもっとも重要な部品だ。最初の畳み込み層のニューロンは、入力画像のすべてのピクセルと接続されているわけではなく、受容野に含まれるピクセルとだけ接続されている（図14-2）。同じように、第2の畳み込み層のニューロンは、最初の畳み込み層の小さな長方形に含まれるニューロンとだけ接続されている。このようなアーキテクチャにすると、最初の隠れ層は下位レベルの特徴量に集中し、次の隠れ層ではそれらをより高水準の特徴量に組み立てることができる

ゼロパディングとストライドとは

ゼロパディング

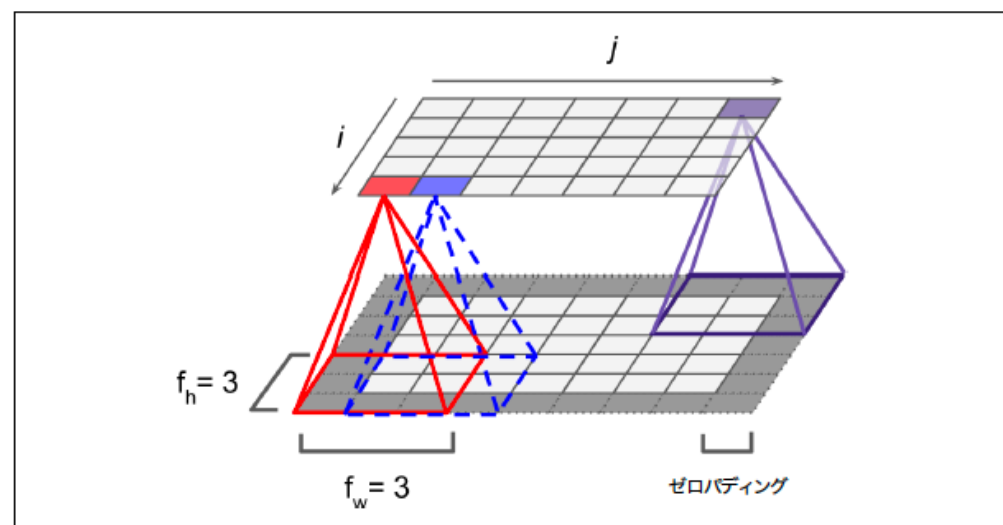


図14-3 2つの層の接続とゼロパディング

高さと幅が前の層と同じになるように、図に示すように、入力周囲に0を追加する。

ストライド

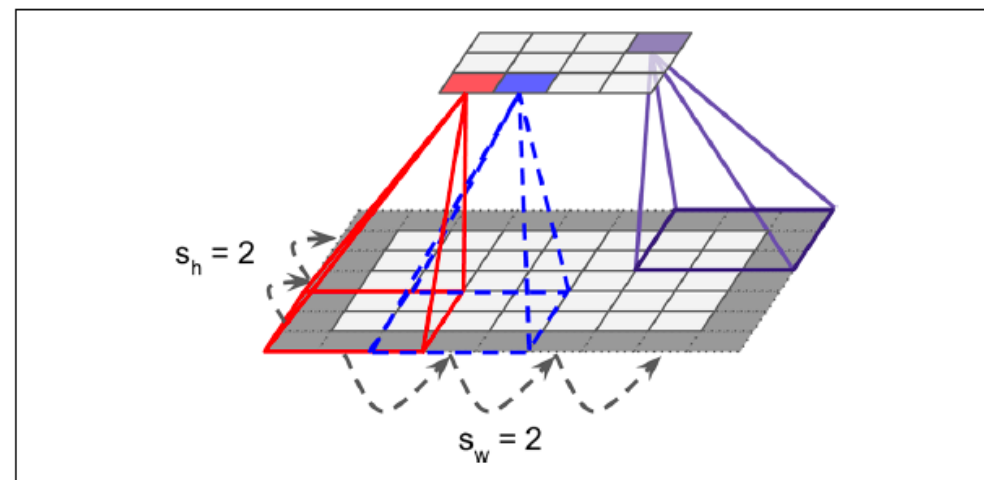


図14-4 ストライド幅2を使った次元削減

受容野を離すと大きな入力層をずっと小さな層に接続できる。こうすると、モデルの計算量が大幅に下がる。**2**つの連続する受容野の距離をストライドと呼ぶ。この図では、受容野と**2**というストライドを使って、**5 × 7**の入力層（およびゼロパディング）を**3 × 4**の層に接続している

14.2.1 フィルタ

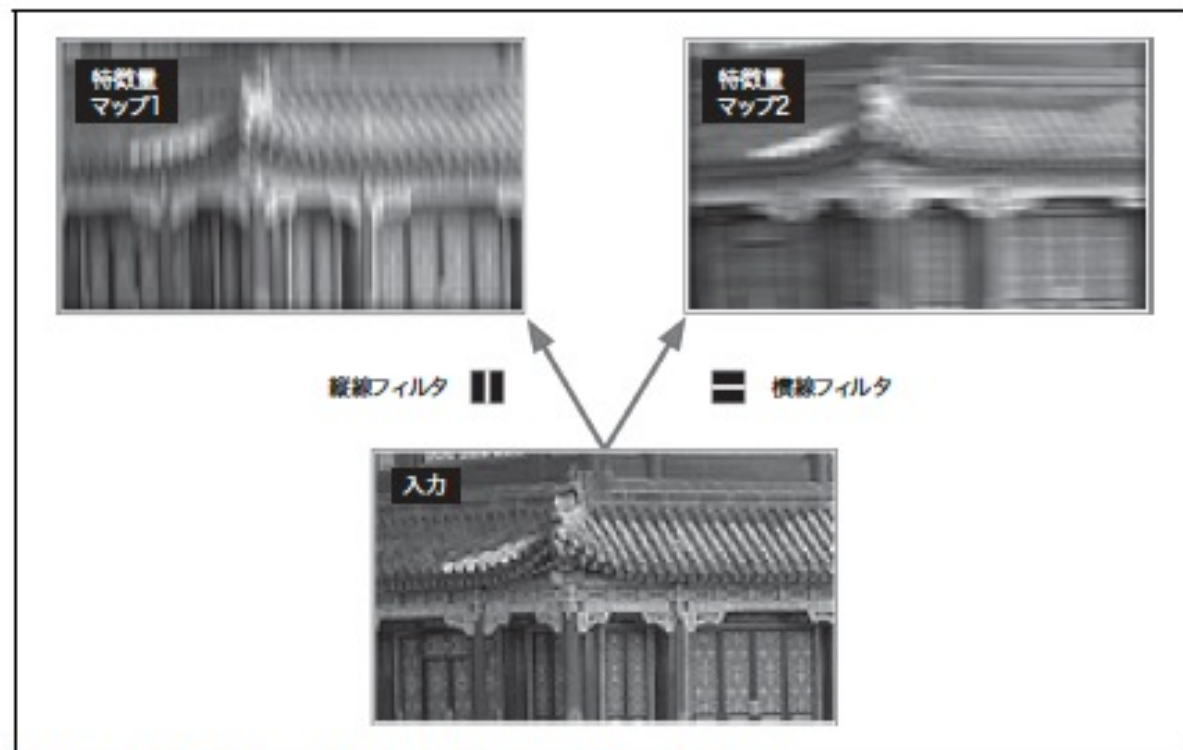


図14-5 2つの異なるフィルタを適用すると2つの特徴量マップが得られる

ニューラルネットワークに図14-5 の下部に示した入力画像を渡し、最初の層のすべてのニューロンが同じ縦線フィルタ（および同じバイアス項）を使ったとすると、その層は左上の画像を出力する。縦の白線に対応する部分が強調され、それ以外はぼやけることがわかる。同様に、右上の画像は、すべてのニューロンが横線フィルタを使ったときに得られるものである。横の白線に対応する部分が強調され、それ以外はぼやける。このように、同じフィルタを使うニューロンで満たされた層を使うと、画像のなかでフィルタにもっとも近い部分が強調される特徴量マップ（**featuremap**）が得られる。

14.2.2 複数の特徴量マップの積み上げ

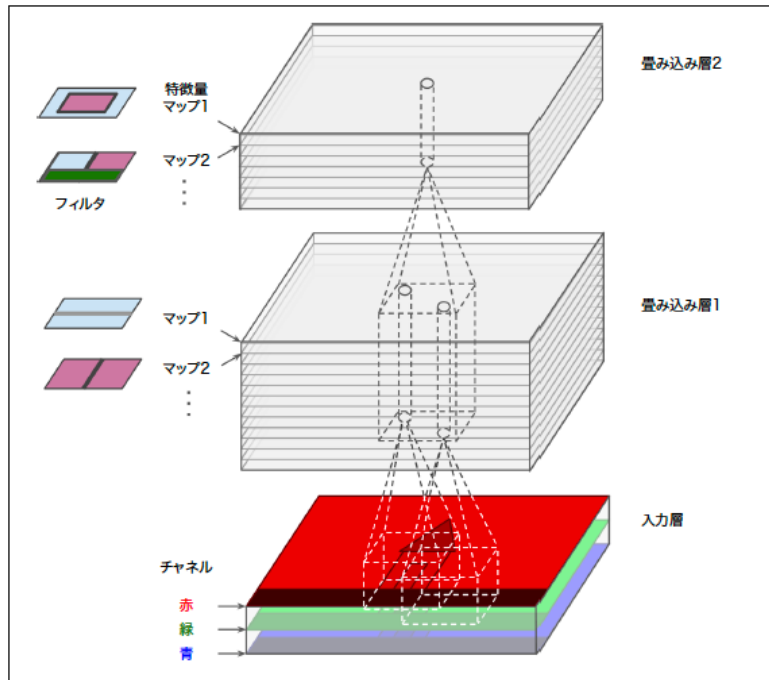


図14-6 複数の特徴量マップを持ち、チャンネルの入力画像を受け付ける畳み込み層

今までは、話を単純にするために、畳み込み層の出力を2次元レイヤとして表現してきたが、実際の畳み込み層は複数のフィルタを持ち、フィルタごとに1つの特徴量マップを出力するので、3次元の方が正確に表現できる

式 14-1 特定の畳み込み層に含まれる特定のニューロンの出力の計算

$$z_{i,j,k} = b_k + \sum_{u=0}^{f_h-1} \sum_{v=0}^{f_w-1} \sum_{k'=0}^{f_c-1} x_{i',j',k'} \cdot w_{u,v,k',k} \text{ with } \begin{cases} i' = i \times s_h + u \\ j' = j \times s_w + v \end{cases}$$

$z_{i,j,k}$

- ・畳み込み層（l層）の特徴量マップ^kのi行j列にあるニューロンの出力
- ・加重総和にバイアス項を加える。

14.2.3 TensorFlowとKerasによる実装

Paddingは"same"か"valid"で書く

paddingは"SAME"か"VALID":でなければならない。

– "SAME"にすると、畳み込み層は必要に応じてゼロパディングを使う。出力ニューロンの数は、入力ニューロンをストライドで割って切り上げた数と同じになる。

入力サイズが13、ストライドが5なら、出力サイズは13になる ($13/5 = 2.6$ 、切り上げて3)。それから、必要に応じて入力の前後にできる限り均等に0を加えていく。stride=1の場合、レイヤの出力の空間次元(幅と高さ)は入力と同じで、だからsameという名前になっている。

– "VALID"にすると、畳み込み層は0パディングを使わず、図14-7に示すように、ストライド次第では入力画像の下の方と右の列の一部を無視する(単純化のために、ここでは横方向しか示していないが、もちろん縦方向でも同じことが行われる) そのため、すべてのニューロンの受容野が入力内の有効な位置のものだけになり(境界の外に出ていかない)、だからvalidという名前になっている。

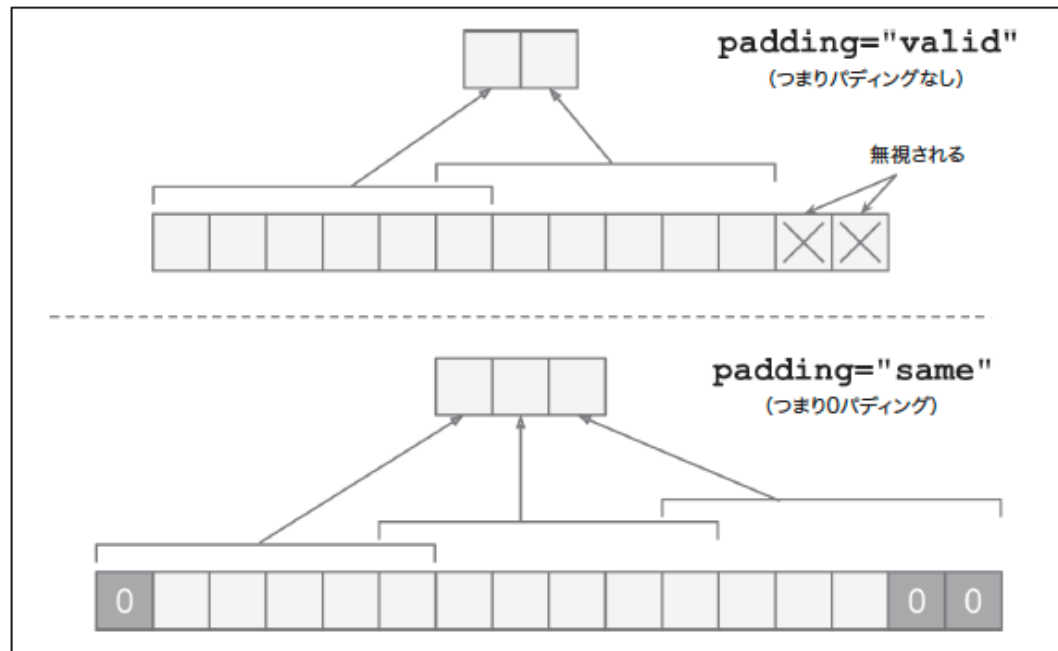


図14-7 padding="VALID"と"SAME"の違い(入力幅13、フィルタ幅6、ストライド5)

14.2.4 メモリ要件

- CNNには、RAMが必要になる

CNNには、特に訓練中に多くのRAMが必要になるという問題もある。これは、バックプロパゲーションのパスで、前進パス中に計算した中間値がすべて必要になるから。

メモリを使い切って訓練がクラッシュしたときには、ミニバッチサイズを小さくしてみるとよい。また、ストライドを使ったり層を削減したりして次元削減する方法もある。さらに、32ビット浮動小数点数の代わりに16ビット浮動小数点数を試すのもよいだろう。そして、複数のデバイスにCNNを分散する方法も残っている。