畳み込みニューラルネットワークによる学習とクラス分類

《学修項目》

- 畳み込みニューラルネットワーク(CNN)の構成
- 畳み込み層、プーリング層、全結合層
- 学習とクラス分類（クロスエントロピー誤差）

《キーワード》

> CNN、畳み込み層、フィルタ行列、画像特徴量、バイアス加算、活性化関数、プーリング層、アベレージ・プーリング、MAXプーリング、全結合層、誤差逆伝播法、識別、クラス分類

《参考文献，参考書籍》

- [1] 東京大学MIセンター公開教材 「AI基礎：3-4 深層学習の基礎と展望」 《利用条件CC BY-NC-SA》
- [2] 東京大学MIセンター公開教材 「AI基礎：3-5 認識」 《利用条件CC BY-NC-SA》

1. はじめに

ここでは、画像認識分野でよく使われている畳み込みニューラルネットワーク(CNN：Convolutional Neural Network)について、その構造と学習・クラス分類の方法について概略を説明する。

図35に一般的な畳み込みニューラルネットワークの構造を示す。ここで、画像をクラス分類するネットワークの構築を想定してみよう（例えば、犬と猫の画像を入力したら、犬の画像の確率と猫の画像の確率を出力してくれるようなネットワーク）。まず、入力された画像の情報は、畳み込み層と呼ばれる部分に入力され、その出力がプーリング層と呼ばれる部分に入力される。

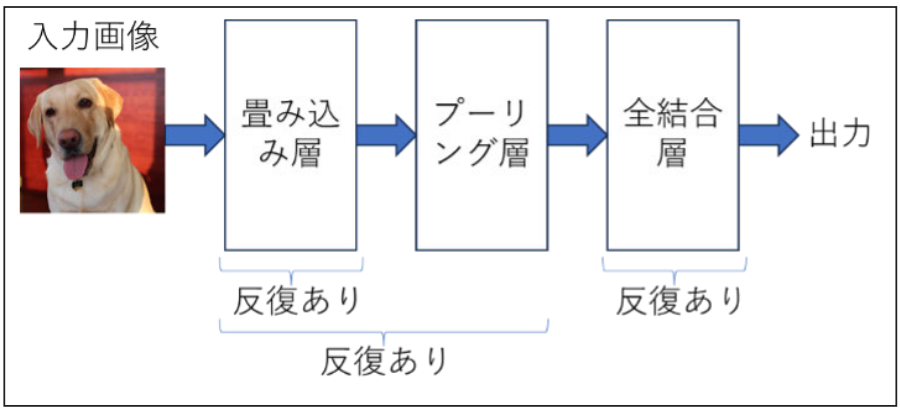


図35 畳み込みニューラルネットワークの構造

ここで、畳み込み層は複数回反復して適用する場合があり、畳み込み層とプーリング層のセットも複数回反復する場合がある。プーリング層からの出力は全結合層に入力され（この部分も通常複数の層で構成される）、その結果として画像をクラス分けした結果を出力する。ここで、全結合層の部分は、「3層型ニューラルネットワークによる学習と識別」で述べたような、隣合う層の全てのユニットが相互に全結合されている構造となっている。

以下、各部の構造や働きについて述べる。

2. 畳み込み層

畳み込み層は、画像の局所的な特徴を抽出する部分と考えることができる。これを実現するために、畳み込み層では、フィルタと呼ばれる行列を利用する。

図36に一般的な畳み込み層の構造を示す。まず、入力された画像がカラー画像であると想定するとR、G、Bで構成される3枚の画像と見ることができるので、これら3枚を入力画像と考えよう（このような入力枚数をチェンネルと呼ぶ。ここでは、チャンネル数が3となる）。

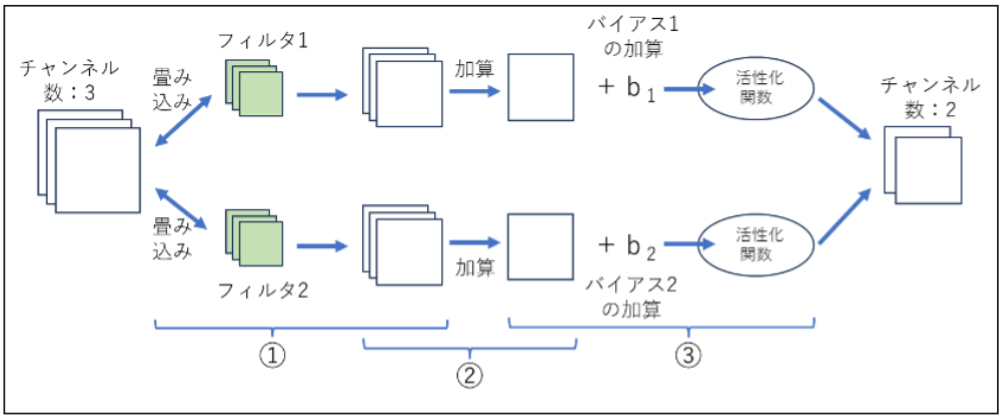


図36 畳み込み層の構造（文献[75]p.240上図を参考に作成）

次に図36①の部分に示すように、各チャンネル画像についてフィルタを畳み込んでいる。この畳み込みの処理は図37のように行われる。

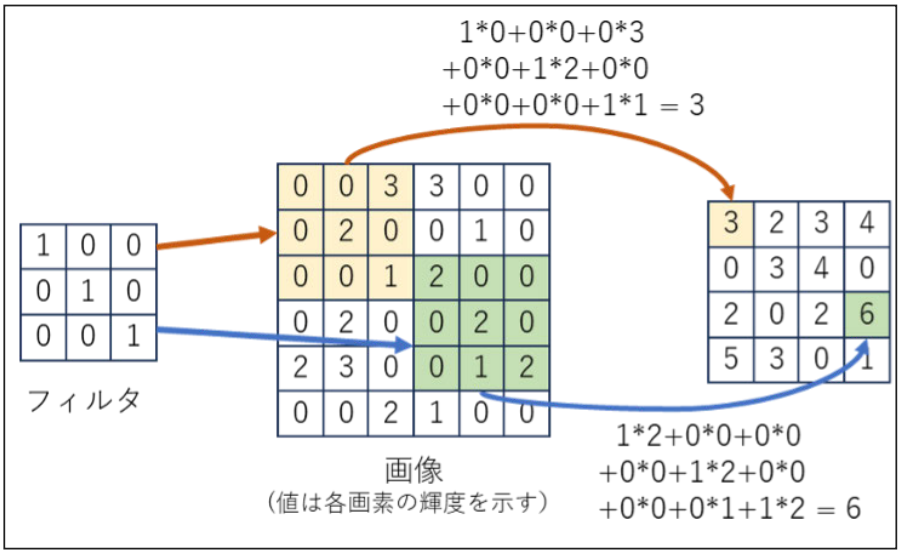


図37 画像へのフィルタの畳み込み（文献[75]p.238 上図を参考に作成）

まず、フィルタが図37の左側の3×3の行列で表現されている場合、これを画像の左上に重ね合わせ（クリーム色のハッチング部分）、同じ位置の要素同士を掛け算してから全体の総和を計算して一つの値を求める操作を行う（左上の場合は3の値が得られる）。

この操作を図38のように画像上の左上から右下に向かってフィルタを重ね合わせて計算した結果が図37の右側の4×4の行列である。このような操作を画像へのフィルタの畳み込み処理と呼ぶ。

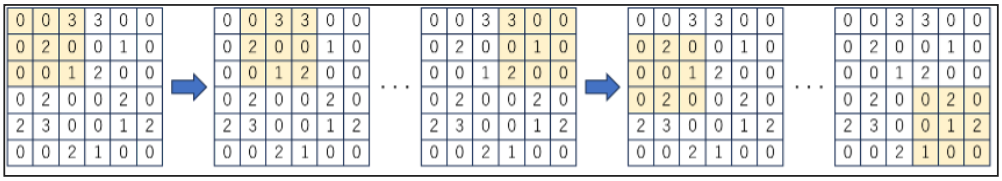


図38 画像へのフィルタの重ね合わせ順序（ストライドが1の場合）

ここで、フィルタの畳み込みが何を意味しているのか考えてみよう。図37のフィルタのように左斜めに1の要素、他の要素が0のようなフィルタの場合、画像に重ね合わせて要素の積和計算を行うと、その重ね合わせた画像部分に左斜め成分がある場合は、大きな値を取ることになる。このようにフィルタは、重ね合わせた局所的な画像の特徴を検出していることに相当する。もちろん、異なるフィルタを適用した場合は、別の特徴を抽出することができる。

なお、画像にフィルタを畳み込んだ結果の行列は元の画像より小さくなる。図37の例の場合、元の画像が6×6行列だったが、3×3のフィルタを畳み込んだ結果、4×4行列になっている。このように、フィルタの畳み込みや後述するプーリングを繰り返すと画像が小さくなってしまうため、元画像を取り囲むように0の値を付加してから、フィルタを畳み込み、画像の縮小を防ぐ場合がある。このような処理をパディングと呼ぶ（0の値を付加する方法をゼロパディングと呼ぶ）。畳み込みの際、パディングは元画像の端の画素の影響が減少してしまうという欠点を改善する効果もある。

また、図38では、フィルタを画像上で1画素ずつ移動しながら走査していたが、N画素(N>1)ずつ移動して走査する場合もある。ただし、移動幅が大きいと画像の局所特徴を取り漏らしてしまう恐れがあるため、画像認識では、1画素ずつ移動させる場合が多い。なお、この移動幅のことをストライドと呼んでいる。

それでは次に図36②の部分について説明する。上記の方法でRGBの各画像についてフィルタ1（各チャンネル用に3種類ある）とフィルタ2（3種類）をそれぞれ畳み込んだ場合、フィルタ1の結果として3つの行列、フィルタ2の結果として3つの行列がそれぞれ得られる。ここで、それぞれの3つの画像特徴量（行列）を統合するために、3つの行列の要素ごとに総和を求め、1つの行列にまとめる。

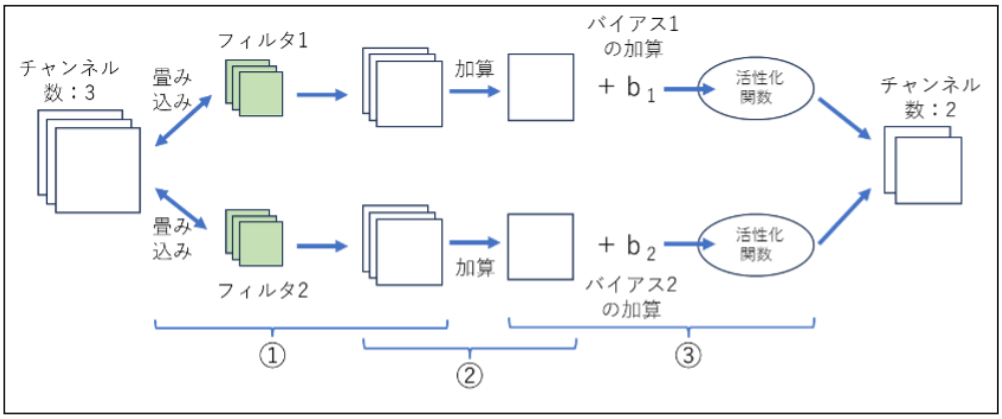


図36 畳み込み層の構造（文献[75]p.240上図を参考に作成）（再掲）

次に図36③に示すように、得られた行列の各要素にバイアス値（図のb1とb2）を加算した後、活性化関数を適用し、最終的な要素値を決定する。この結果、フィルタ数と同じ数の行列（図の場合、行列の数は2）が得られ、各行列の大きさは畳み込み後の行列の大きさと一致したものが得られる。

3. プーリング層

畳み込み層で得られたチャンネル数分の行列に対して、通常はプーリングと呼ばれる処理を適用する。プーリングは、画像特徴の局所的な位置ずれを許容し、行列のサイズを小さくする効果がある。

それでは、プーリングの具体的な処理手順を見ていこう。プーリングには、局所領域の平均値を求めるアベレージプーリングと最大値を求めるMAXプーリングがあるが、ここではCNNでよく使われているMAXプーリングについて説明する。

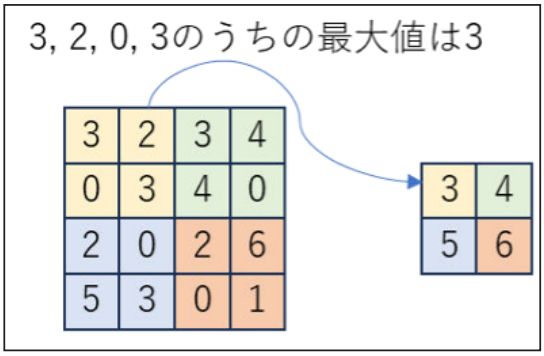


図39 MAXプーリングの例

畳み込み層の結果、図39左の行列が得られた場合、これに2×2の領域でMAXプーリングを適用した結果が右の行列である（ストライドは2）。このように、図のハッチングされた2×2の小領域を見て、その中での最大値を抽出して、新たな行列を生成する。

プーリングは小領域の代表値を計算しているため、前述のように、画像内で特徴を示す部分の位置が多少変動したとしても、その位置ずれを吸収できる効果がある。また、図39のように、プーリングを適用すると適用前の行列より結果の行列サイズが小さくなる効果があり、計算時間の削減にも効力を発揮する。

4. 全結合層と出力

図35に示すように、プーリング層の結果は、最終的に全結合層への入力となり、全結合層の出力が最終結果となる。

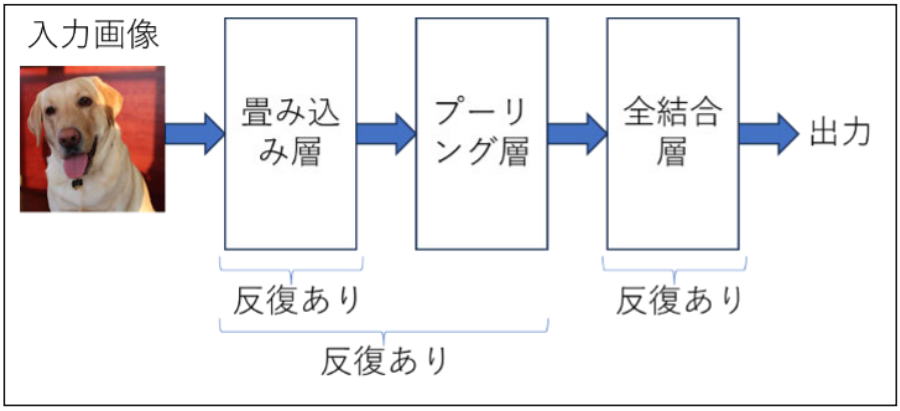


図35 畳み込みニューラルネットワークの構造（再掲）

ここでは、図40を使って、その様子を説明する。まずチャンネル数分あるプーリングの結果（図ではチャンネル数が2）を1列の数値列（ベクトル）に変換し、この数値を全結合層の入力層の入力とする。

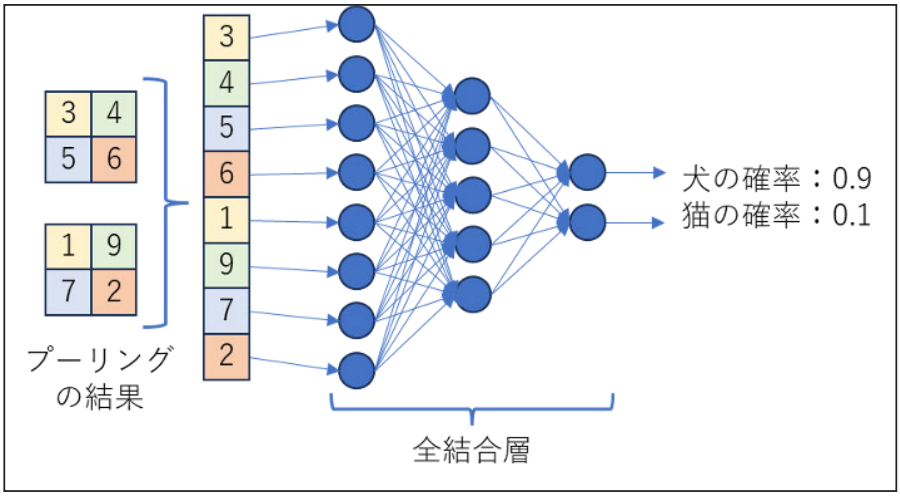


図40 全結合層とその出力

前述のように、全結合層は、「3層型ニューラルネットワークによる学習と識別」で説明した構成と同じであり（つまり隣り合う層のユニット間が全て結合されており、そこに重みが付加されている）、各層の出力の計算方法も同じである。ただし、最終結果として、画像のクラス分類をする場合（例えば、画像が犬のクラスか、猫のクラスかを分類するような場合）、各クラスの画像である確率を出力させたい。

このような場合は、全結合層の出力層の値にソフトマックス関数と呼ばれる活性化関数を適用することが多い。ソフトマックス関数 $f(x)$ は、識別クラスの数が n（出力ユニットの数がn）だった場合、ある出力ユニットの値xについて、下記の値を返す。

$$f(x) = \frac{exp(x)}{\sum_{i=1}^{n} exp(x_i)}$$

ここで、x_iは出力ユニットのi番目の値を示す。これにより、出力ユニットの各値を0から1の値に正規化され、かつ出力ユニット全体の総和を1とすることができる。よって、各ユニットの出力値を各クラス画像となる確率値とみることができる（値1に近づくほど、そのクラス画像である確率が高いとみる）。

5. 学習と識別

ネットワークの学習では、上記の手法で得られた出力値について、教師信号を与えて誤差を計算し、ネットワークの各種パラメータの値を更新する。

それでは、まず教師信号について考えてみよう。教師信号は、入力画像に該当するユニットの確率値を1として、他のユニット値を0に設定すれば良い。図40の例では、犬の画像を入力した場合は、最終結果を示す出力ユニットに教師信号として「1, 0」（上のユニットに1、下のユニットに0）を与え、猫の場合は、「0, 1」を与えることになる。

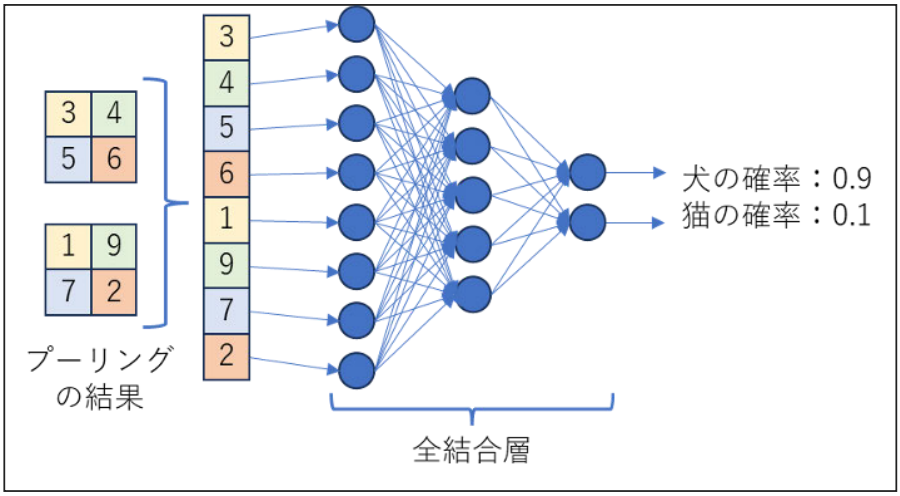


図40 全結合層とその出力（再掲）

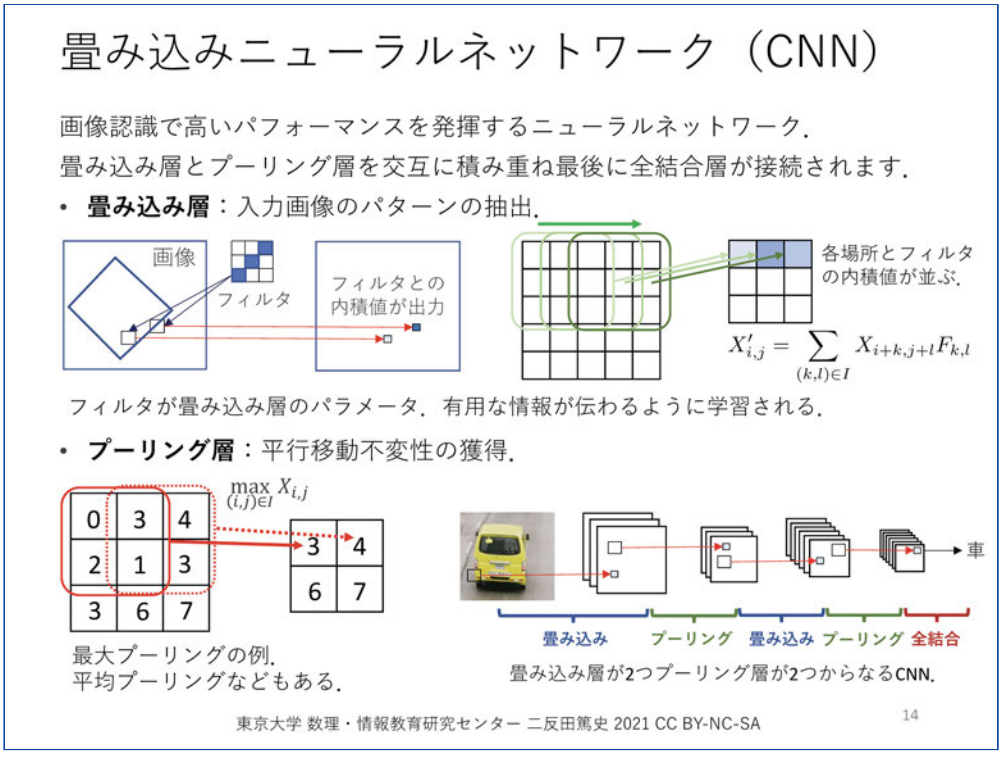
次に出力値と教師信号の誤差について説明する。画像のクラス分類を行う場合は、誤差としてクロスエントロピー誤差を採用する場合が多い。クロスエントロピー誤差は、2つの分布の差（この場合、ネットワークが計算して出力する値の分布と教師信号が与える値の分布との差）を計算することができる。

いま、k番目の出力ユニットの出力値をy_k、対応する教師信号を $\widehat{y_k}$ とした場合、クロスエントロピー誤差は、下記の値で計算できる。

$$-\sum_k \widehat{y_k} log(y_k)$$

学習では、この誤差の値を最小にする方向に、畳み込み層のフィルタの値・バイアスの値、全結合層の重みの値を誤差逆伝播法によって修正する。そして、大量の教師付き画像データで学習を行ったネットワークは、画像を入力すると、その画像がどのクラスにどれだけの確率で属しているかを出力するようになる。

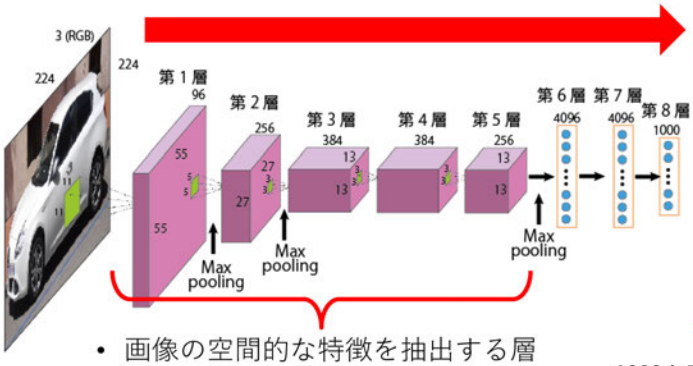
6. まとめ [1][2]



畳み込みニューラルネットワーク（CNN; Convolutional Neural Network）

- 代表的な物体識別モデル
- 実際には様々な実装がある（下図はAlexNet）

最終層にsoftmaxを適用した結果



0.00 goldfish
0.00 great white shark
0.00 tiger shark
0.00 hammerhead
0.01 electric ray
...
0.89 car
...
0.00 stinkhorn
0.00 earthstar
0.00 hen-of-the-woods
0.02 bolete
0.00 ear, spike
0.00 toilet tissue

- 画像の空間的な特徴を抽出する層
- 途中の層ではどうなっているか？

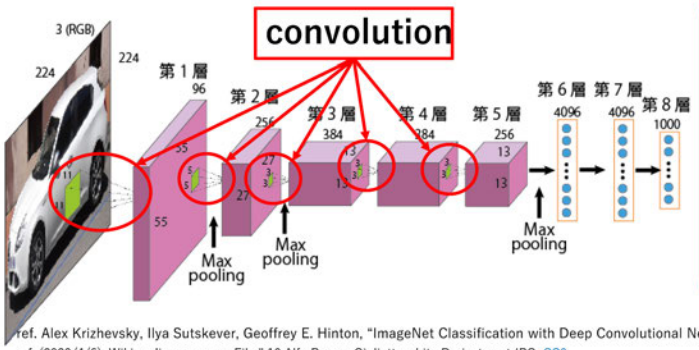
1000クラスのうちcarだけが高く、残りがほぼ0のようなベクトルが出力

ref. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS2012

ref. (2020/4/6): Wikimedia commons: File:" 10 Alfa Romeo Giulietta white Derivate cut.JPG CC0
https://commons.wikimedia.org/wiki/Category:CC-Zero?uselang=ja#/media/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG

畳み込みニューラルネットワークにおける畳み込み演算

- 第1層から第5層までは、2次元デジタルフィルタで説明した畳み込み演算（convolution）を行っている
- ただし、2次元デジタルフィルタではフィルタは人がデザインしていたのに対し、CNNではデータから学習により取得する



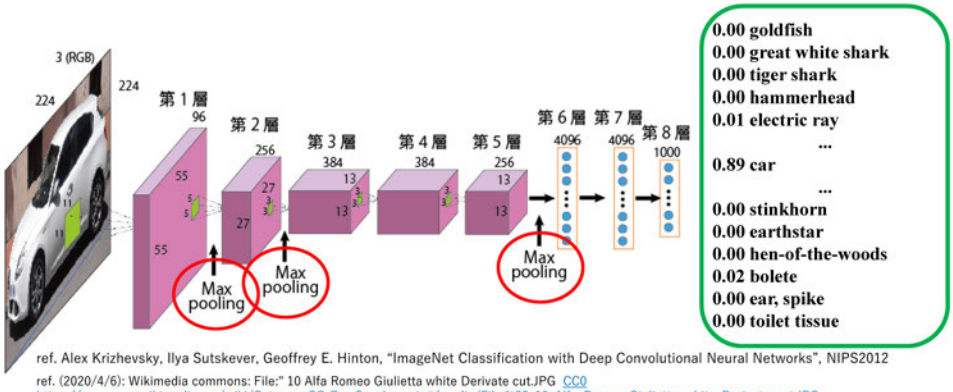
0.00 goldfish
0.00 great white shark
0.00 tiger shark
0.00 hammerhead
0.01 electric ray
...
0.89 car
...
0.00 stinkhorn
0.00 earthstar
0.00 hen-of-the-woods
0.02 bolete
0.00 ear, spike
0.00 toilet tissue

ref. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS2012

ref. (2020/4/6): Wikimedia commons: File:" 10 Alfa Romeo Giulietta white Derivate cut.JPG CC0
https://commons.wikimedia.org/wiki/Category:CC-Zero?uselang=ja#/media/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG

畳み込みニューラルネットワークにおける プーリング (pooling)

- 画面を小さく区切り、各区間ごとに画素をまとめて1つの値にする
  - データのサイズを小さくする役割
  - 物体が写っている位置や角度の違いに頑健にする役割
- 最大値を取る場合 max pooling、平均値をとる場合 average poolingと呼ぶ

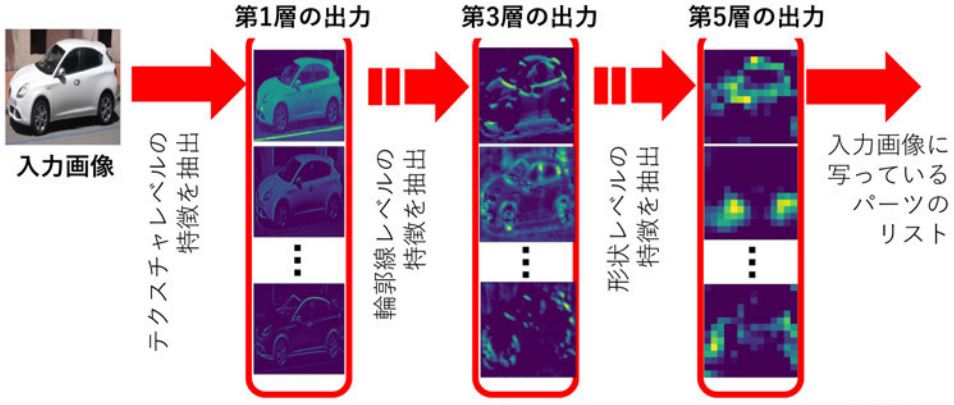


ref. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS2012

ref. (2020/4/6): Wikimedia commons: File:" 10 Alfa Romeo Giulietta white Derivate cut.JPG CC0
https://commons.wikimedia.org/wiki/Category:CC-Zero?uselang=ja#/media/File:%22_10_Alfa_Romeo_Giulietta_white_Derivate_cut.JPG

CNNにおける画像のフィルタリング

- 層を経るごとにより具体的な形状の特徴を取り出していく
- 第5層になると、「入力画像にどんなパーツが写りこんでいるか（実際には尤度分布）」がわかってくる
- 「入力画像に写っているパーツのセット」が「他のクラスに比べ、車にありがちなパーツのセット」であるならば「車」と判別



モデルはKeras VGG16 pretrainedを使用

ref. (2020/4/3): Wikimedia commons: File:Red Apple.jpg CC BY 2.0
https://commons.wikimedia.org/wiki/File:Red_Apple.jpg

深層学習はそれまでの画像処理と何が違ったのか？

- AlexNet (2012)登場より前の画像認識では、CNNにおける第1~2層の出力に相当する情報を使って認識していた
  → Deep Networkと対比してShallow networkと呼ばれる
- 2000年前後にいくつかの技術革新
  - 数学的解法：勾配消失問題（層を深くすると学習が進まなくなる現象）に対する効率的な解決法の提案（1990年代後半）
  - GPGPUの発展：コンピュータグラフィックスの描画に用いられていたGPUをベクトル計算機とみなして気象や地震シミュレーション等、数値計算に利用（2006年NVIDIAがCUDA提供開始）
  - Big Data時代の到来：Webで画像やテキストなどが大量に収集できるようになり、モデルの学習に使えるデータが爆発的に増加
- 様々な画像処理タスクに対する学習データセットが公開
  - 学習データを使わない自己教師あり学習（self-supervised learning)の研究も進められている

memo