

# MATLABで始める ディープラーニング

# ディープラーニングとは？

ディープラーニングは機械学習の手法のひとつです。モデルは、画像、テキスト、音声から直接分類方法を学習します。ディープラーニングには通常、ニューラルネットワークの構造が使われます。「ディープ」という表現は、ネットワーク内の層の数を示しており、層が多いほどネットワークは深くなります。従来のニューラルネットワークでは2~3層程度でしたが、ディープニューラルネットワークは数百もの層を持つこともあります。



# ディープラーニングの適用分野

ディープラーニングは実際に以下のような分野で活用されています。

- 横断歩道に近づくと減速する自動運転車
- 偽装された紙幣を拒否するATM
- 外国の道路標識を瞬時に翻訳するスマートフォンアプリ

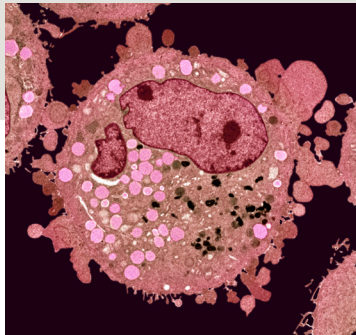
ディープラーニングは、顔認識、テキスト翻訳、音声認識のほか、車線区分や交通標識の認識など先進運転支援システム (ADAS) の高度な識別に特に適しています。



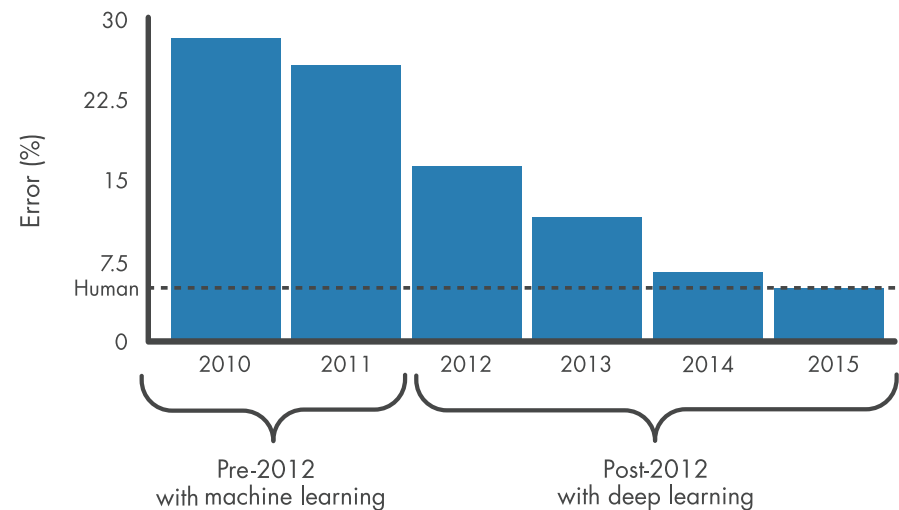
# ディープラーニングはなぜ最先端技術なのか

ひと言で言えば、精度です。高度なツールや技術には、飛躍的に向上したディープラーニングのアルゴリズムが使用されています。このアルゴリズムは人間よりも高い精度で画像を分類したり、世界最強の囲碁棋士に勝利するほど性能が高く、ユーザの好みに合った新しい曲を見つけてダウンロードするAmazon Echo®やGoogle Homeといった音声制御アシスタントにも活用されています。

UCLAの研究者たちは先端的な顕微鏡を構築し、この顕微鏡から高次元のデータセットを生成しました。このデータセットをディープニューラルネットワークに学習させ、組織試料のがん細胞を検出しています。



ILSVRC TOP-5 ERROR ON IMAGENET

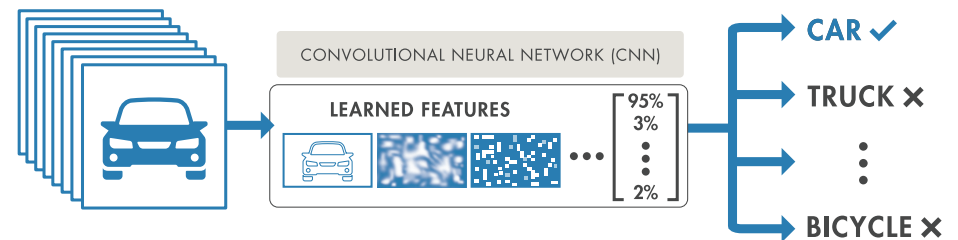


# ディープラーニングはなぜ最先端技術なのか (続き)

次の3つの技術がディープラーニングの高い認識精度を可能にしています。

## 大規模なラベル付きデータセットの入手し易さ

無償提供されているImageNetやPASCAL VoCなどのデータセットは、幅広い種類の物体の学習に活用できます。



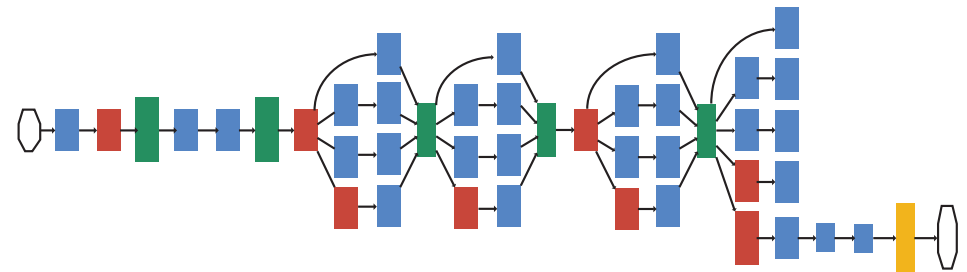
## 処理能力の向上

高性能なGPUにより、ディープラーニングに必要な大量データの学習を高速化でき、学習時間を数週間から数時間にまで短縮することができます。



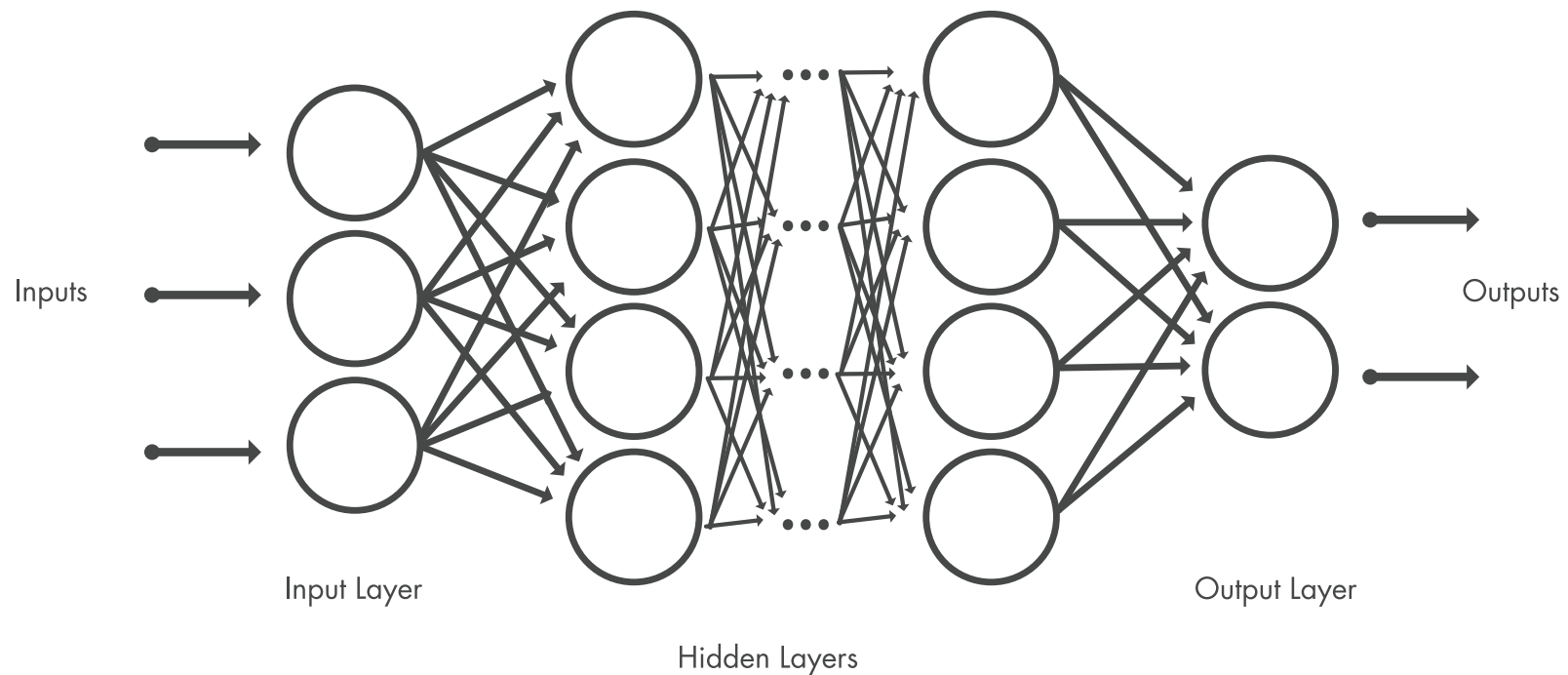
## 専門家が構築した学習済みモデル

「転移学習」と呼ばれる手法で、AlexNetなどのモデルを再学習し新しい認識タスクを実行できます。AlexNetは130万枚もの高解像度画像で学習され、1000種類の物体を認識できるネットワークです。転移学習では、学習済みモデルを活用することで、はるかに少ないデータセットで高い精度に到達できます。



# ディープニューラルネットワークの仕組み

ディープニューラルネットワークは、非線形な処理を行う複数の層からなります。各層はシンプルな要素毎の演算を並列で実行しますが、これらの演算は生物の神経系にインスパイアされたものとなっています。このネットワークは、入力層、複数の隠れ層、出力層で構成され、各層はノードまたはニューロンを經由して相互接続されています。隠れ層はそれぞれ、前の層の出力をその入力として使用します。



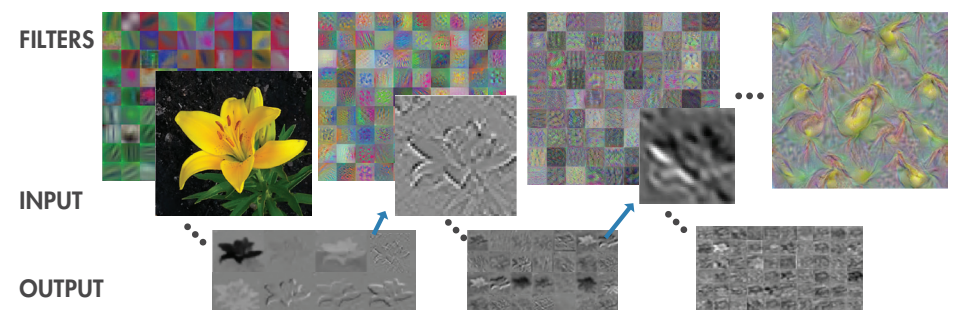
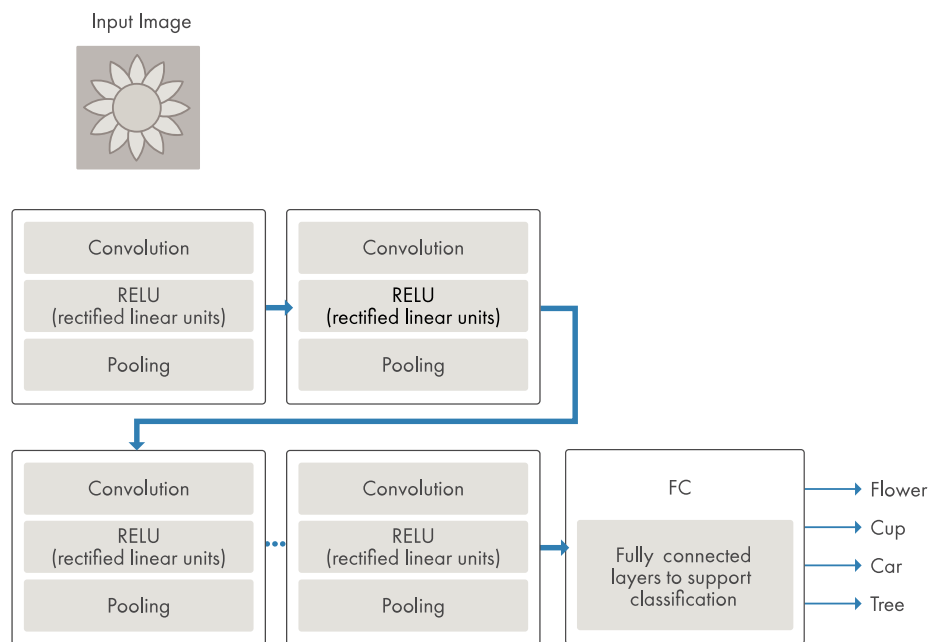
# ディープニューラルネットワークの学習方法

たとえば、ある画像セットの各画像に4種類の物体のいずれか1つが含まれていて、私たちはディープラーニングネットワークを使用して各画像にどの物体が含まれるかを自動的に認識したいとします。ネットワークの学習データとして各画像にラベル付けを行います。

ネットワークはこの学習データを使用して、物体の具体的な特徴を解釈し、対応するカテゴリに関連付けることができます。

ネットワークの各層は前の層からデータを取り込み、変換して、次の層に渡します。ネットワークの複雑度と学習される内容の詳細度は、層から層へと渡される度に増していきます。

ここで注目すべきポイントは、ネットワークが直接データから学習している点です。我々はどのような特徴量が学習されるかについて一切影響を与えることはできません。



# 畳み込みニューラルネットワークについて

畳み込みニューラルネットワーク (CNNまたはConvNet) は、静止画や動画向けのディープラーニングで最も一般的なアルゴリズムです。

その他のニューラルネットワークと同様に、CNNは入力層、出力層、その間にある多くの隠れ層で構成されます。

## 特徴抽出のための層

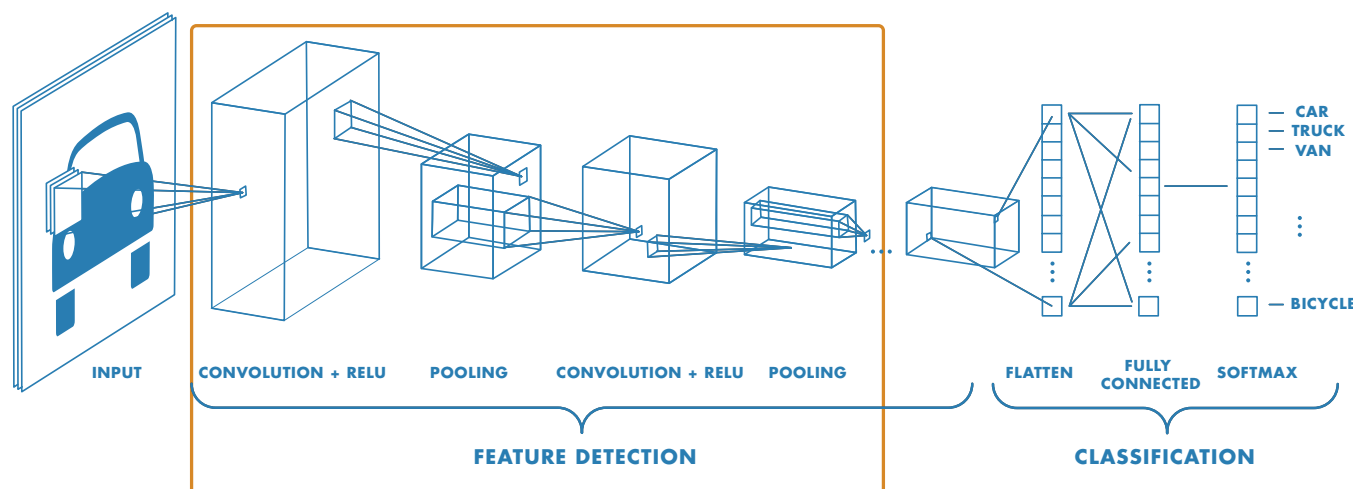
各層は、データに対して畳み込み、プーリング、正規化線形ユニット (ReLU)のうち、いずれか1つの操作を実行します。

**畳み込み** は、入力画像を複数の畳み込みフィルターに通します。それぞれのフィルターが画像の持っている特定の特徴に対して活性化します。

**プーリング** では、非線形のダウンサンプリングを実行して出力を単純化していきます。同時にネットワークの学習に必要なパラメータ数も削減します。

**正規化線形ユニット (ReLU)** では、負の値をゼロにマッピングし、正の値を保持するように動作することで、より高速かつ効率的な学習を実現します。

これら3つの動作は数十層または数百層にわたって繰り返され、それぞれの層が異なる特徴量を検出するように学習が行われます。





# 畳み込みニューラルネットワークについて (続き)

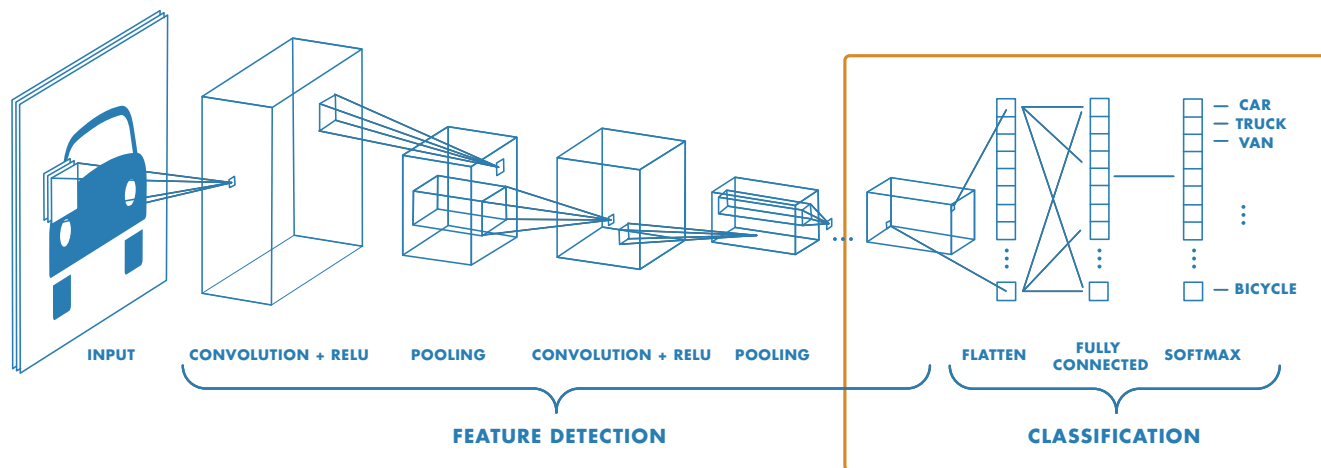
## 分類のための層

特徴量を抽出した後、CNNは分類タスクにシフトします。

最後から2番目の層は、K次元のベクトルを出力する**全結合層 (FC)**です。Kはネットワークで予測可能なクラスの数を示します。このベクトルには、分類されるすべての画像の各クラスに対する確率が含まれています。

CNNの最終層では、**ソフトマックス関数**を使用して分類結果を出力します。

どのような層をどう積み上げるかについては、確実な公式のようなものではありません。一番よい方法は、いくつか試してその精度を確認する、または事前学習済みのネットワークを使用することです。



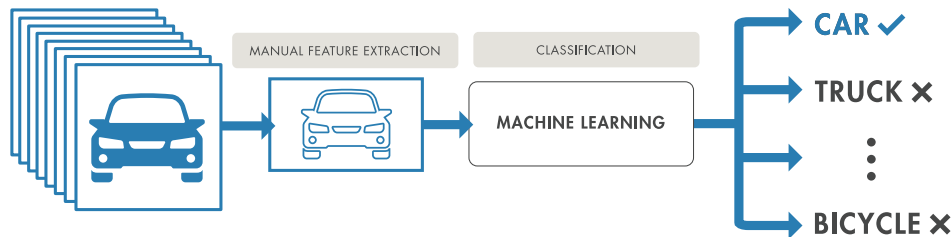
≫ CNNの仕組みを詳しく見る

# ディープラーニングと機械学習の違いとは？

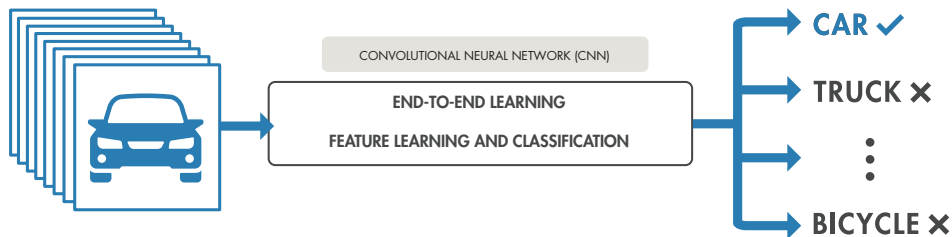
ディープラーニングは機械学習の特殊形と言うべきものです。機械学習では、画像に含まれる特徴を手作業で抽出しますが、ディープラーニングでは、生の画像データをディープニューラルネットワークに直接入力することで、ネットワークが自動的に特徴量の抽出方法を学び取ります。

ディープラーニングで優れた結果を得るには、何十万、あるいは何百万もの画像がしばしば必要となります。また、計算量が多いため高性能なGPUも欠かせません。

TRADITIONAL MACHINE LEARNING



DEEP LEARNING

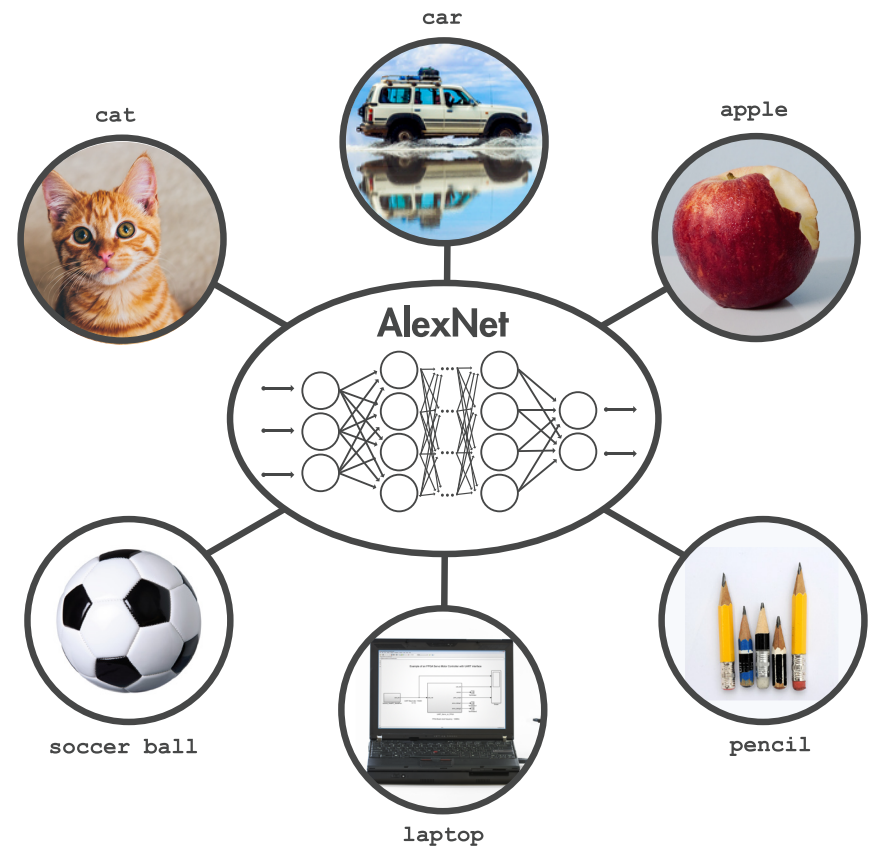


機械学習	ディープラーニング
+ 小さなデータセットでも良い結果が得られる	- 非常に大規模なデータセットが必要
+ モデルの学習が高速	- 計算量が多い
- 最良の結果を得るには、さまざまな特徴量と分類器を試す必要がある	+ 特徴量と分類器を自動的に学び取る
- 精度に上限がある	+ 精度に上限がない

# ディープラーニングを始めるには

もしディープラーニングにあまり慣れていない場合には、AlexNetのような学習済みのネットワークを使うことで、素早く簡単に作業に取り掛かることができます。AlexNetとは、100万を超える画像で学習が行われた学習済みのネットワークのことで、画像の分類タスクにおいて最もよく使われているネットワークといえます。AlexNetは、キーボード、コンピューターのマウス、鉛筆、その他の事務用品のほか、犬、猫、馬などの動物品種といった1000種類もの異なるカテゴリに画像を分類することができます。

AlexNetは2012年にはじめて発表されて以来、リサーチコミュニティにおいては非常によく知られるモデルとなりました。



≫ [学習済みネットワークに関する詳細はこちら](#)

# AlexNetの使用例

AlexNetを使用してあらゆる画像の物体を分類できます。この例では、デスクトップパソコンにインストールしたWebカメラの画像に写る物体を分類するのにAlexNetを使います。MATLAB®本体に加えて、以下を使用します。

- Deep Learning Toolbox™
- MATLABでWebカメラを使用するためのサポートパッケージ
- AlexNetを使用するためのサポートパッケージ

AlexNetを読み込んだ後、Webカメラに接続し、ライブ画像をキャプチャします。

```
camera = webcam; % カメラに接続する  
  
nnet = alexnet; % ニューラルネットワークを読み込む  
  
picture = camera.snapshot;  
% 写真を撮る
```

次に、画像サイズをAlexNetのサイズ要件に合わせて227 x 227 ピクセルにサイズ変更します。

```
picture = imresize(picture,[227,227]);  
% 写真サイズを変更する
```

これで、AlexNetで画像を分類できるようになります。

```
label = classify(nnet, picture);  
% 写真を分類する  
  
image(picture); % 写真を表示する  
  
title(char(label)); % ラベルを表示する
```

マグカップ



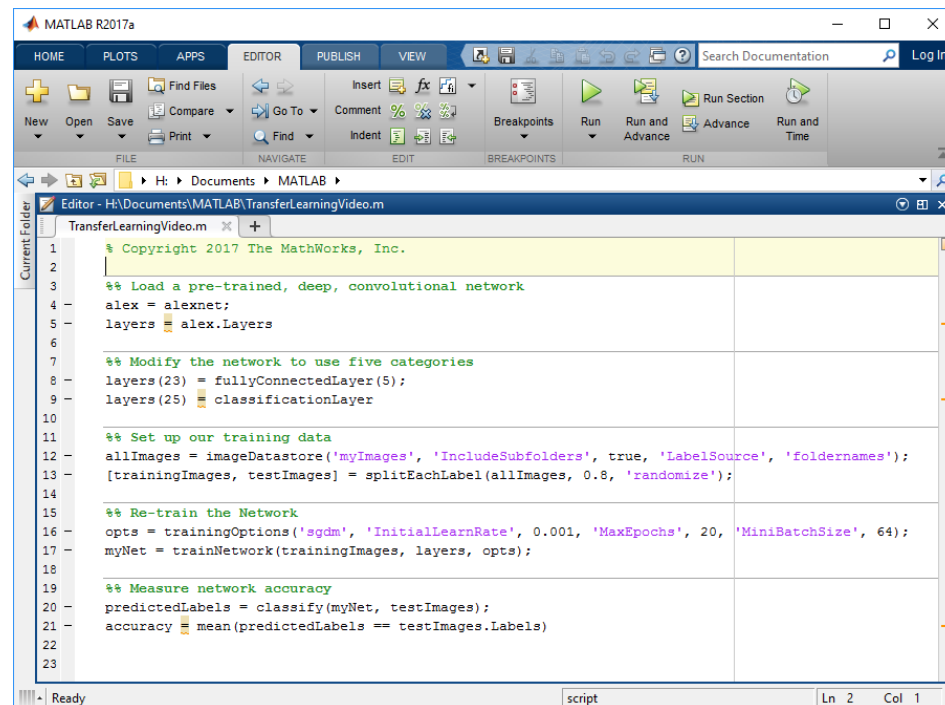
≫ 入門ビデオを見る: 11行のMATLABコードでディープラーニング

# 既存のネットワークの再学習

前の例では、ネットワークに変更を加えないでそのまま使いました。何も修正しなかったのは、私たちが分類したいと思っていた画像に似た画像で AlexNet が学習されていたからです。

これに対して、もともとのネットワークで学習されていない物体に対して AlexNet を使用する場合は、転移学習により再学習させることができます。転移学習は、ある特定の課題に関する知識を、関連する別の課題に適用する手法です。この例では、ネットワークの最後の3層を再定義して、手持ちの画像を使用して再学習を行います。

転移学習があなたの課題にうまく当てはまらない場合は、独自のネットワークをゼロから学習する必要があります。この方法では最も精度の高い結果が得られますが、通常、何十万ものラベル付き画像と、非常に多くの計算リソースが必要になります。



```
1 % Copyright 2017 The MathWorks, Inc.
2
3 %% Load a pre-trained, deep, convolutional network
4 alex = alexnet;
5 layers = alex.Layers
6
7 %% Modify the network to use five categories
8 layers(23) = fullyConnectedLayer(5);
9 layers(25) = classificationLayer
10
11 %% Set up our training data
12 allImages = imageDatastore('myImages', 'IncludeSubfolders', true, 'LabelSource', 'foldernames');
13 [trainingImages, testImages] = splitEachLabel(allImages, 0.8, 'randomize');
14
15 %% Re-train the Network
16 opts = trainingOptions('sgdm', 'InitialLearnRate', 0.001, 'MaxEpochs', 20, 'MiniBatchSize', 64);
17 myNet = trainNetwork(trainingImages, layers, opts);
18
19 %% Measure network accuracy
20 predictedLabels = classify(myNet, testImages);
21 accuracy = mean(predictedLabels == testImages.Labels)
22
23
```

## >> 転移学習を始める

# ディープラーニング向けの計算資源

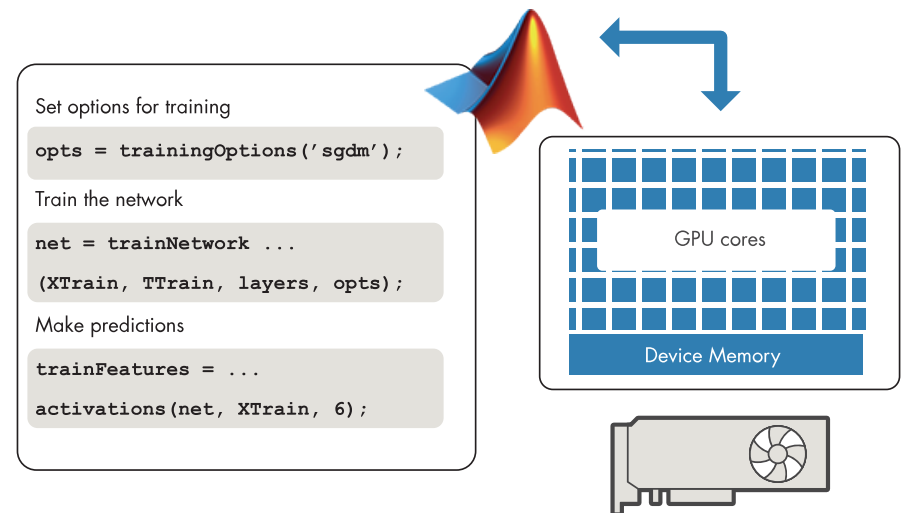
データサイズや利用可能な処理能力によっては、ディープラーニングモデルの学習には、数時間、数日、数週間といった長い時間を要する場合があります。ワークフローを設定する際にどのような計算資源を選択するかは、非常に重要なポイントとなります。

現在、計算資源として使える選択肢としては、CPU、GPU、クラウドの3つの選択肢があります。

**CPUの計算**は、最もシンプルで入手しやすい選択肢です。前のセクションで説明した例はCPUで動作しますが、CPUベースの計算は学習済みのネットワークを使用した単純なケースでのみ使用することをおすすめします。

**GPUを使用すると**、ネットワークの学習時間を数日から数時間に短縮することができます。MATLABを使うことで、追加のプログラミングを行わずにGPUを使用できます。NVIDIA® 3.0の計算に対応したGPUをおすすめします。複数のGPUを使用することで、処理速度をさらに高速化できます。

**クラウドでのGPU計算**は、ハードウェアを購入してセットアップする必要がないことを意味します。ローカルGPUを使うように書かれたMATLABコードは、いくつかの設定を変更するだけでクラウドリソースを使うように拡張してやることができます。



》 GPUでビッグデータを使用したディープラーニングについてさらに詳しく

# その他のディープラーニングのリソース

[ディープラーニングの基礎を学ぶ](#)

[MATLABによるディープラーニング: クイックスタートビデオ](#)

[転移学習でディープラーニングを素早く始める](#)

[AlexNetを使用した転移学習](#)

[畳み込みニューラルネットワークの基礎を学ぶ](#)

[分類を行う単純なディープニューラルネットワークの作成](#)

[ディープラーニングと光学タイムストレッチによるがん診断](#)

