平成 30 年度　修士論文

添削者を困らせることのない
修士論文の書き方の研究

名古屋大学大学院理学研究科
素粒子宇宙物理学専攻宇宙地球物理系
宇宙線物理学研究室

博士課程（前期課程）2 年
学籍番号 123456

奥村 曉

2019 年 1 月 18 日

概要

ここには論文の概要（abstract）を書きます。論文の先頭なので早い時期に書き始める人がいますが、論文の結論や論理展開はなかなか執筆終盤まで固まりません。そのため、論文の流れや結論がかなり明確になった最終段階で書くようにしましょう。

概要は論文全体の内容を短文で説明するものですので、研究の背景と目的、研究内容、結果と結論などが全て網羅されている必要があります。ここを読んだだけで、論文の中身が大雑把に把握できるようにすることが大切です。原則として改行せずに 1 段落で書きますが、これは複数段落に分けて書くような文章を無理やり 1 段落に合体させろということではありません。1 段落で流れるように書いてください。

目次

図目次

表目次

第 1 章

深層学習による病理画像の診断支援

病理画像をデジタルで保存することが始まったのは数十年前になる．これによって遠隔地でも診断することができるようになったり，情報を共有することができるようになり，複数の医師で診断しミスを防止するセカンド・オピニオンが容易になった．計算機科学の分野の側面ではデータを収集することができるようになり，研究が盛んに行われることになった．その後は，様々な病理データでより改良されたアルゴリズムの提案が行われている．

細胞組織の形態を観察するための病理染色ではヘマトキシリン・エオジン染色 (HE 染色) が一般的に用いられる．細胞核を青紫色に染色し，細胞質をピンク色に染色する．正常から異常に変化していくと，細胞核が過度に増殖したり，細胞質の形が崩れたりすることで，その特徴を機械学習によって精度よく検出するための研究が行われている．

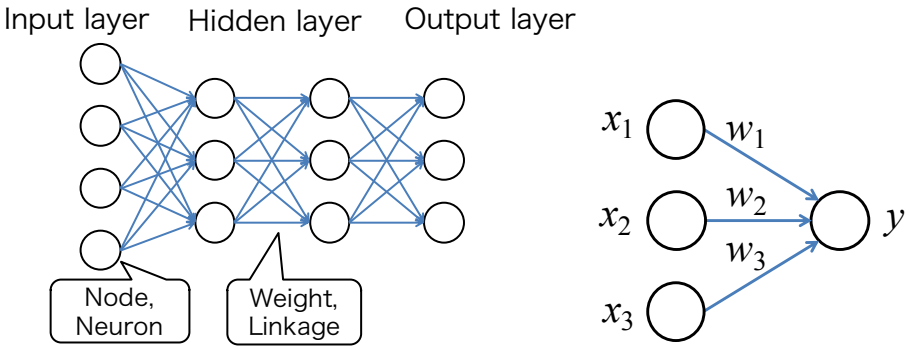
これまでは，核の形やテキスチャーからパターンマッチングなどの画像処理によって腫瘍を検出する研究されてきたが，近年になって画像処理に大きなブレークスルーが起きたことをきっかけに，新しい手法で解析するようになってきた．そのブレークスルーがディープラーニングである．

1.1　ニューラルネットワーク

人間の脳にはニューロンと呼ばれる神経細胞が 1000 億個以上あり，それぞれが複数のニューロンが電気信号によって情報を伝達している．また脳にはシナプスという場所があり，ここで電気信号を細胞体へ受け渡す．細胞体はある閾値以上の電気信号がきた場合に他のニューロンへ電気信号を伝播させる (これを発火と呼ぶ)．このようなニューロンとシナプスで行われる演算を模倣したアルゴリズムを作ることができれば，人間のような思考や認識をコンピュータを使って再現できると考えた．そのアルゴリズムがニューラルネットワーク (Neural Network: NN) である．

1.1.1　多層パーセプトロン

ニューラルネットワークは入力層，出力層，隠れ層から構成され，層と層の間にはニューロン同士のつながりの強さを示す重みがある．非線形問題を扱うために 1986 年 Rumelhart によって考案されたのが，パーセプトロンを複数つなぎ合わせ入力と出力以外に隠れた層を持つ多層パーセプトロン (Multi-layer perceptron: MLP) である (Figure 1.1(a))．ニューラルネットワークで多層パーセプトロンの層を全結合 (fully connected: FC) 層とも呼ぶ．



(a) Muti-layer perceptron　　(b) Simple perceptron

Fig. 1.1: Architecture of Muti-layer perceptron

Figure 1.1(a) における丸や矢印はそれぞれノード (またはニューロン) と重み (または結合) と呼び，ともに数値である．例えば画像を分類しようと思えば，各ピクセルの画素数を各ノードに入力する．例えば 28×28pixel のグレースケール画像であれば，784 個のノードが必要となる．入力データ $\boldsymbol{x}$ が入力層に入ってくると，その値に重み $\boldsymbol{w}$ をかけ，活性化関数 H と呼ばれる関数に通し，結果 $\boldsymbol{y}$ を出力する．ここで，入力 $\boldsymbol{x}$，重み $\boldsymbol{w}$，出力 $\boldsymbol{y}$ を太字で表したが，これらは全てテンソルであり，1 つの層にあるノード $x_1, x_2, \cdots x_n$ を一括して $\boldsymbol{x}$ として表記している．

ここで，中間層の 1 つのノードについて考える．Figure 1.1(b) に MLP を構成する 1 ユニットである単純パーセプトロンを示した．この模式図を数式で表すと次のようになる．

$$y = H(\boldsymbol{w}\boldsymbol{x} + \boldsymbol{b}) \tag{1.1}$$

$$= H\left(\sum_{i=1}^{3} w_i x_i + b_i\right) \tag{1.2}$$

ここで $\boldsymbol{b}$ はバイアスと呼ばれ，発火のしやすさを表している．中間層における活性化関数は，Eq. (1.3) に示す正規化線形関数 (rectified linear unit: ReLU) と呼ばれる関数) がよく用いられる．

$$H(x) = \max\{0, x\} = \begin{cases} x & (x > 0) \\ 0 & (x \leq 0) \end{cases} \tag{1.3}$$

この演算を繰り返し出力層に書き出す．ここで，各層の重みの値によって出力結果は異なってくる．

出力層では，ノードの個数は区別したいクラス数分用意する．各ノードの出力値が各クラスに属している確率を表すように，活性化関数にはソフトマックス関数を用いる (ただし二値分類の場合はシグモイド関数を用いる)．ソフトマックス関数は Eq. (1.4) で表される．

$$y_i = \frac{\exp(x_i)}{\sum_{k=1}^{n} \exp(x_k)} \tag{1.4}$$

ここで y_i は，出力層が全部で n 個あるとして，i 番目の出力であることを示す．Eq. (1.4) からわかるように，入力の総和に対して 1 つのノードがどれくらいの値を持つかという割合で表されている．これにより各ノードの出力は確率として解釈できるため，値の一番大きいノードのインデックスを予測ラベルとして見ることができる．

1.1.2 畳み込みニューラルネットワーク

従来の画像認識では，画像から特徴を抽出しそれを識別器にかける手法が主流であった．古典的手法では画像から特徴を抽出するいわゆる特徴量設計が必要で，ここをいかにうまく設計するかがポイントであった．特徴抽出の方法として，HOG[?] や SIFT[?]，SURF[?] などがあり，これらによって抽出した特徴ベクトルを Support Vector Machine(SVM)[?] によって識別することが多かった．

しかし，1998 年に LeNet と呼ばれる畳み込みニューラルネットワーク (Convolutional Neural Network: CNN) が提案された [1]．CNN は畳み込み層とプーリング層からなっている．この畳み込みとプーリングの演算を通して，特徴量設計から識別までを end-to-end で行うことができる．

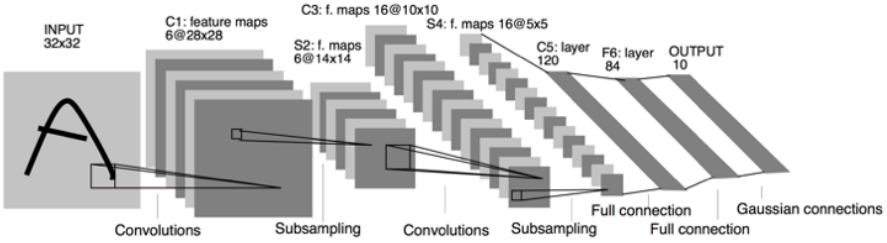


Fig. 1.2: Architecture of convolutional neural network[1]

畳み込み層では，入力に対してフィルター (カーネルとも呼ばれる) を用意し，Eq. (1.5) に示す計算を行う．

$$y_{i,j} = (\boldsymbol{K} * \boldsymbol{x})_{i,j} \tag{1.5}$$

$$= \sum_m \sum_n x_{i+m,j+n} K_{m,n} \tag{1.6}$$

ここで，$\boldsymbol{K}$ はフィルター，$\boldsymbol{x}$ は入力，y は出力である．CNN ではこの演算の後に活性化関数に通す．これを図で表すと Figure 1.3 のようになる．

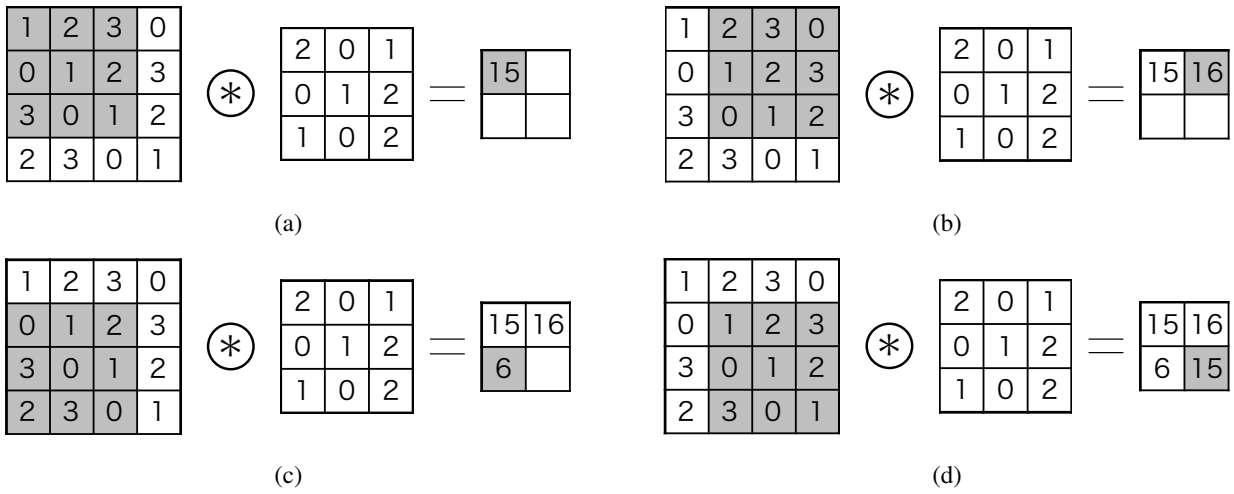


Fig. 1.3: Operation process of convolution

Figure 1.3 では，フィルターのサイズは 3×3 であるが，大きさは任意である (3×3 や $5 \times 5, 7 \times 7$ がよく用いられる)．また，フィルターは 1 マスずつ横にずらして計算を行っている．ずらし方をストライドといい，今回はス

トライド 1 である．CNN では多くの場合，ストライドは 1 である．このようなフィルターの畳み込み計算を行うと，フィルターごとに異なった画像の特徴を抽出して数値化することができる．

次にプーリングを行う．ここでは，画像認識で多く用いられる最大値プーリングについて述べる．Figure **??**に示すように，2×2 のプールサイズを用意した時，その範囲内にある最大値を取る演算である．ストライドはプールサイズと合わせ，プーリングを行った領域と被らないようにすることが一般的である．Figure **??**ではストライド 2 である．

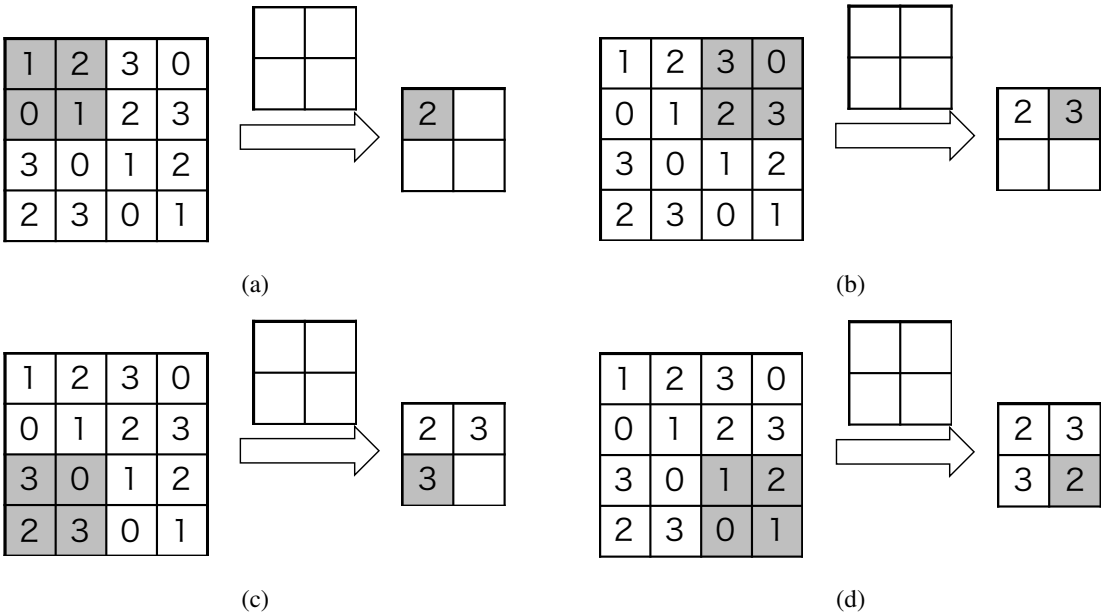


Fig. 1.4: Operation process of max pooling

プーリング層では画像のサイズを小さくして (コンピュータの) 計算コストを減らし，微小な変化に対してロバストになる．

この畳み込みとプーリングを繰り返して，入力からフィルターの数だけ特徴を抽出し，この抽出した特徴マップを FC 層へ繋げて識別を行う手法が CNN である．

1.1.3　再帰的ニューラルネットワーク

再帰的ニューラルネットワーク (Recurrent Neural Network: RNN) は時系列解析や自然言語処理に利用されるニューラルネットワークである．内部にループを持つことで過去の情報を保持しておくことができる．時系列の入力 $x = (x_1, \cdots, x_T)$ があった時に，出力 $y = (y_1, \cdots, y_T)$ と隠れ層のベクトル $h = (h1, \cdots, h_T)$ をそれぞれ以下の式で計算する．

$$h_t = H(W_{ih}x_t + W_{hh}h_{t-1} + b_h) \tag{1.7}$$
$$y_t = W_{ho}h_t + b_o \tag{1.8}$$

ここで，W は重み行列 (W_{ih} は入力と隠れ層間の重み行列) であり，b はバイアス項である．そして H が活性化関数である．

RNN は過去の情報をどこまでさかのぼって関連性を見つけるかを判断することができないため，時系列データが長くなるほど，その長期の依存性を学習するには人が慎重にパラメータを設計する必要があるなど，学習が難し

くなるという問題があった．この問題を解決するために Long-short term memory(LSTM) が提案された．LSTM も RNN の一種であるため繰り返し構造を持ち，3 つのゲートを持つ層からなっている．

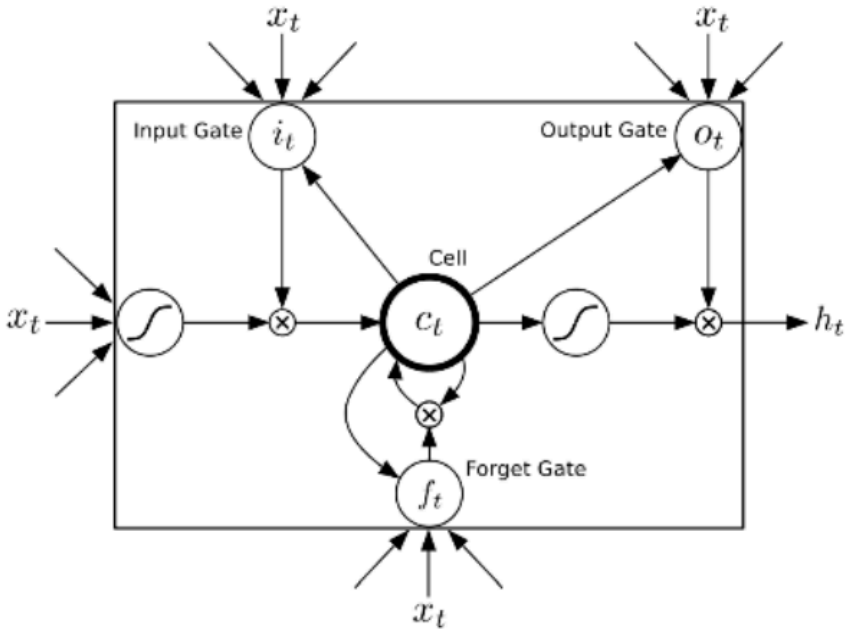


Fig. 1.5: Architecture of LSTM

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \tag{1.9}$$
$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{1.10}$$
$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{1.11}$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \tag{1.12}$$
$$h_t = o_t \tanh(c_t) \tag{1.13}$$

ここで σ はシグモイド関数であり，次の式で定義される．

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{1.14}$$

f_t で示される層は忘却ゲート層と呼ばれ，過去の情報で捨てるべき情報を判断する．これはシグモイド層によって行われ，0 と 1 の間の値を出力し，0 は完全に忘れる．1 は完全に維持するという意味である．i_t や c_t で示される層は，入力ゲート層と呼ばれ，i_t で新たに入力された情報から，どの情報を更新するかを判断し c_t で古い情報を落とし新しい情報を加え，値を更新する．最後が o_t や h_t で示される層で，出力ゲート層と呼ばれる．まず何を出力するべきかを o_t のゲートで判断して，c_t に tanh を適用して掛けることで出力が計算される．

1.2 推論と学習

ニューラルネットワークでは推論フェーズと学習フェーズに分かれている．

推論フェーズでは，順伝播ニューラルネットを用いる．

学習フェーズでは，逆伝播ニューラルネットを用いる．

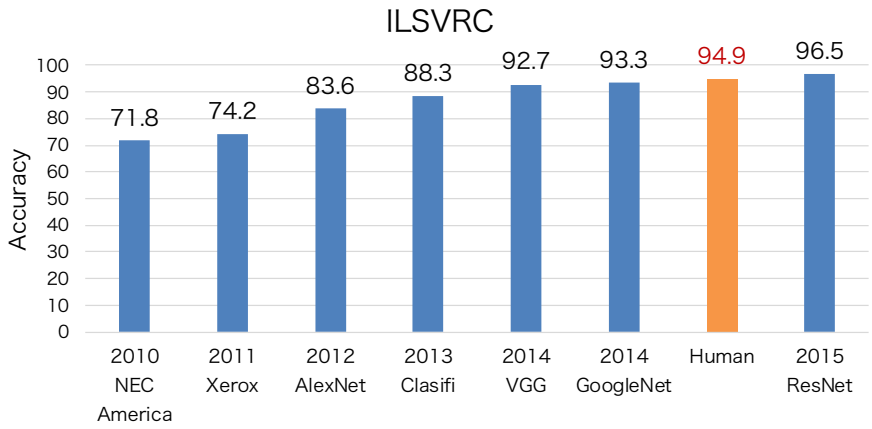


Fig. 1.6: Transition of accuracy of image recognition on ILSVRC

損失関数 (loss function)，コスト関数，目的関数とも呼ばれる．

平均二乗和誤差 (mean squared error: MSE)

$$E_{\mathrm{MSE}} = \frac{1}{2}\sum_{k}(y_k - t_k)^2 \tag{1.15}$$

交差エントロピー誤差 (cross entropy error)

$$E_{\mathrm{crossentropy}} = -\sum_{k} t_k \ln y_k \tag{1.16}$$

誤差逆伝播法 (Backpropergation)

1.2.1　最適化手法

確率的勾配降下法 (Stochastic gradient decay: SGD)

Adam

1.2.2　学習のテクニック

Dropout

Batch Normalization

1.3　画像認識におけるディープラーニング

Deep Learning とは Deep Neural Network(DNN) を指すことが多い．この"Deep"とは，ニューラルネットワークの層が深いことに由来している．

Figure 1.6 に画像認識タスクの精度の近年の推移を示す．これは ImageNet Large Scale Visual Recognition Challenge (ILSVRC) と呼ばれる世界的な画像認識のコンペティションである (2010 年から始まった)．カテゴリ数は 1000 クラスで，画像枚数は 120 万枚の訓練データと 15 万枚のテストデータが用意されている．2011 年と 2012 年は約 10% もの大差で AlexNet[2] が優勝している．これがディープラーニングの始まりである．AlexNet は 5 つの畳み込み層と 3 つの全結合層を持っている．2014 年には VGGNet[3] や GoogLeNet[4] が 9 割の精度を超えた．VGGNet は AlexNet(8 層) よりさらに深い構造 (19 層) であり，GoogLeNet は 22 層もある．そして 2015 年には ResNet[5] が人間の精度をも超える認識精度を達成した．ResNet は GoogLeNet よりもさらに深く 152 層もあ

る．CNN を複数回かけて検出を行う場合，CNN の浅い側では空間分解能はあるが抽象的な情報が少ない．深い側では意味論的な情報は取得できる（ポーズ，変形など）が空間分解能が小さいため幾何学的な情報が失われる．

アーキテクチャの進化の方向は大きく 3 つある．1 つ目は層を深くすることである．2 つ目は FC 層の使用を避ける，または Inception モジュールの使用することである．これにより学習するパラメータ数を削減することができる．3 つ目は ResNet などのショートカット接続の利用や，事前学習・転移学習を行うことである．これによって学習効率を向上させ，最終的にモデルの精度向上へと繋がる．ここで，事前学習のデータセットと適用データとの間には類似性があると良い．

画像処理におけるディープラーニングでは大きく 3 つのタスクがあり，それぞれ，クラス分類，物体検出，セグメンテーションである．以下に詳細を述べる．

クラス分類

与えられた画像をカテゴリごとに分ける手法である．

物体検出

物体検出とは Bounding Box で物体の位置とその物体の種類を特定する方法である．歴史的には幾何的情報，手動特徴量，そしてそのカスケードを利用していた．その後，HOG や SIFT など局所特徴量を抽出する方法を設計するようになったが，これは深い専門知識を必要とした．また広い範囲でオブジェクトを正確に検出する方法は，メモリ容量と処理時間に課題がある．現在は Deep Neural Network になりデータのみから抽象的な特徴量を複数得ることができる．Figure 1.7 に物体検出で有名はアルゴリズムである SSD と YOLO のアーキテクチャを示す．クラス分けの場合は数 1000 のカテゴリを学習して Top Error Rate が 2% 以下と人間よりも認識精度が高いが，物体検出においては，現状ではカテゴリが数 100 程度くらいまででも認識精度が人間よりも低くなってしまう．また物体検出は精度を上げるために処理に時間がかかることが多いため，リアルタイムに物体検出を行う時は，速度と精度のトレードオフが生じてしまう．

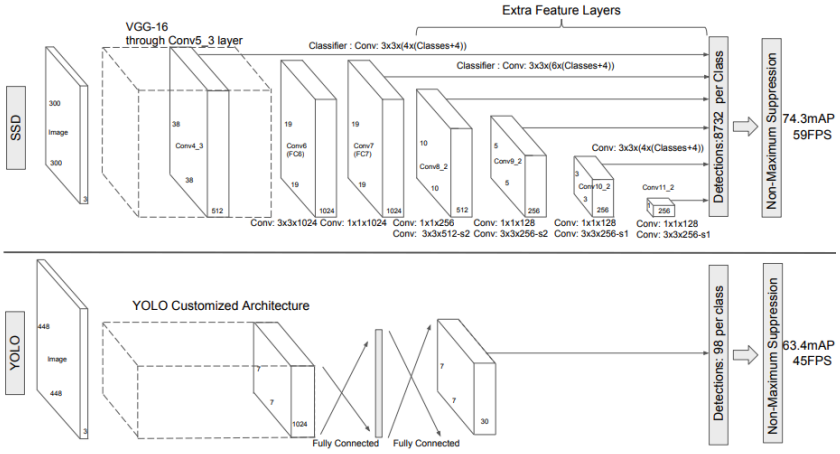


Fig. 1.7: Network Architecture of SSD and YOLO

セグメンテーション

セマンティックセグメンテーションとは，画像を画素レベルで認識することである．画像内の各画素をオブジェクトクラスに割り当てる手法である．セマンティックセグメンテーションの手法についてディープラーニング以前では，Texton Forests や，Random Forests に基づいた分類を行っていたが，物体検出と同様に CNN が登場してからは，高精度なセグメンテーションが実現するようになった．CNN を使ったセグメンテーションの手法で一般的に使われるようになったものが Unet である (Figure 1.8)．この Unet は文字通り U の形をしたネットワークであることが特徴で，2 つのアーキテクチャーからできている．1 つ目がエンコーダーのアーキテクチャーで CNN とプーリングで特徴を抽出しながら次元を削減していき，2 つ目のデコーダーのアーキテクチャーで画像をセグメンテーションの結果になるように復元する．ここで問題になることが，プーリングをすることで位置情報を消してしまっているので，この位置情報を利用して画像を復元するためには，エンコーダーとデコーダーで画像サイズが同じところ同士をショートカットで接続することが Unet 構造の優れている点である．

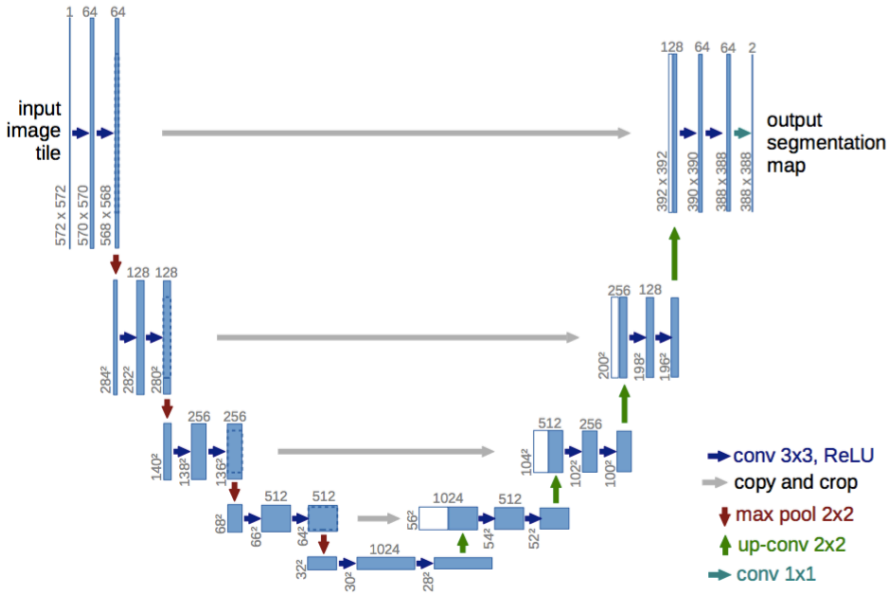


Fig. 1.8: Artchitecture of Unet

1.4　深層学習による 3 次元画像解析

ディープラーニングを医療画像に応用するコンペティションが世界で行われているが，その半数が 3D 医療画像の解析になっているほど需要が高まっている．その理由は，現在解決しなくてはならない課題があるからである．まずは 2 次元画像と違って，処理するべきデータが大きいということである．そのため学習するパラメータをなるべく少なくする工夫がされている．また 3 次元画像には，動画または，ボリューム画像があるが，2 次元画像とその深さ方向 (動画であれば，時間方向) には異方性があることから，機械学習の方法に工夫が必要になる．今まで考案されている手法として，2DCNN を拡張した 3DCNN，また CNN と時系列解析でよく用いられる LSTM を組み合わせた手法と，LSTM 内部に CNN を組み込んだ手法，それらをすべて組み合わせた手法が考案されいる．LSTM の研究も盛んに行われているため，その改良モデルが数多く存在する．特に，LSTM の学習効率を上げた

GRU(Gated Linear Unit) や，順方向だけでなく逆方向の時系列も計算に入れる BiLSTM が時系列解析の精度向上になっている報告がある．

1.4.1　3DCNN と Stacked Convolution

2 次元画像が深さ方向に連続している 3 次元画像の特徴を抽出するために，2 次元の CNN を拡張して，3 次元のカーネルを使って畳み込みを行う，3DCNN を利用した手法がある．

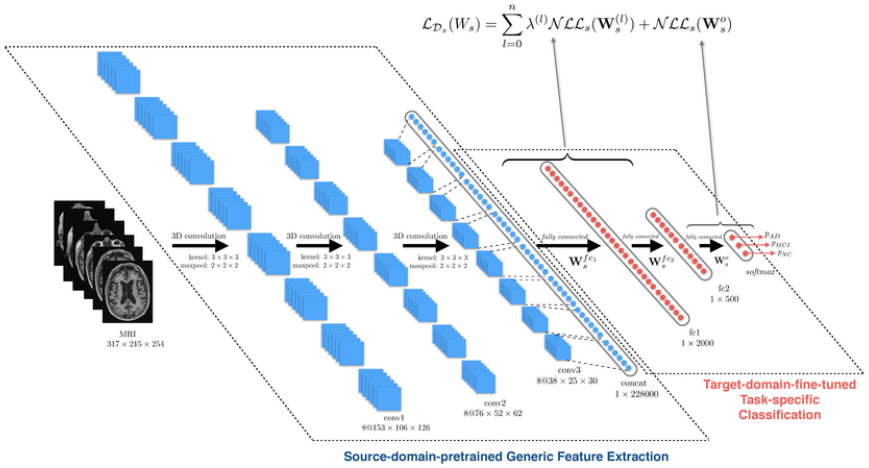


Fig. 1.9: Artchitecture of 3DCNN

1.4.2　LSTM と 2DCNN の組み合わせ

時系列解析に使われる LSTM を用いて 3 次元の画像を解析することができる．これはよく動画の解析で行われることがある．つまりフレームごとの画像の特徴を 2 次元の CNN で計算してから時系列情報を LSTM で解析することで，画像の時系列解析を行うことができる．これを 3 次元の医療画像で CT や MRI で適用する研究も行われている．3DCNN のデメリットであったパラメータの増大を 2DCNN と LSTM の組み合わせで解決することができる．

1.5　教師なし学習

機械学習の手法には，上記で説明したように，ラベルの貼られているデータセットを用いて学習することを教師あり学習と呼び，その反対で，データセットはあっても，そのデータセットの特性を示したラベルが与えられていない場合のデータセットを用いて学習することを教師なし学習と呼ぶ．

1.5.1　Autoencoder

教師なし学習で画像の特徴を抽出する方法としてオートエンコーダがある．画像の場合におけるオートエンコーダの手法とは，ある画像から情報を圧縮する「エンコーダ」と言われる部分と，その圧縮した情報から画像を復元する「デコーダ」の二つからなる．入力とデコーダから復元された画像が同じ画像になるようにニューラルネットワークで学習させる．この学習の結果，潜在変数は似てる画像どうしで近い値になるように変化し，この分布を見れば画像の分類を教師ラベルがなくても，学習を行うことができる．

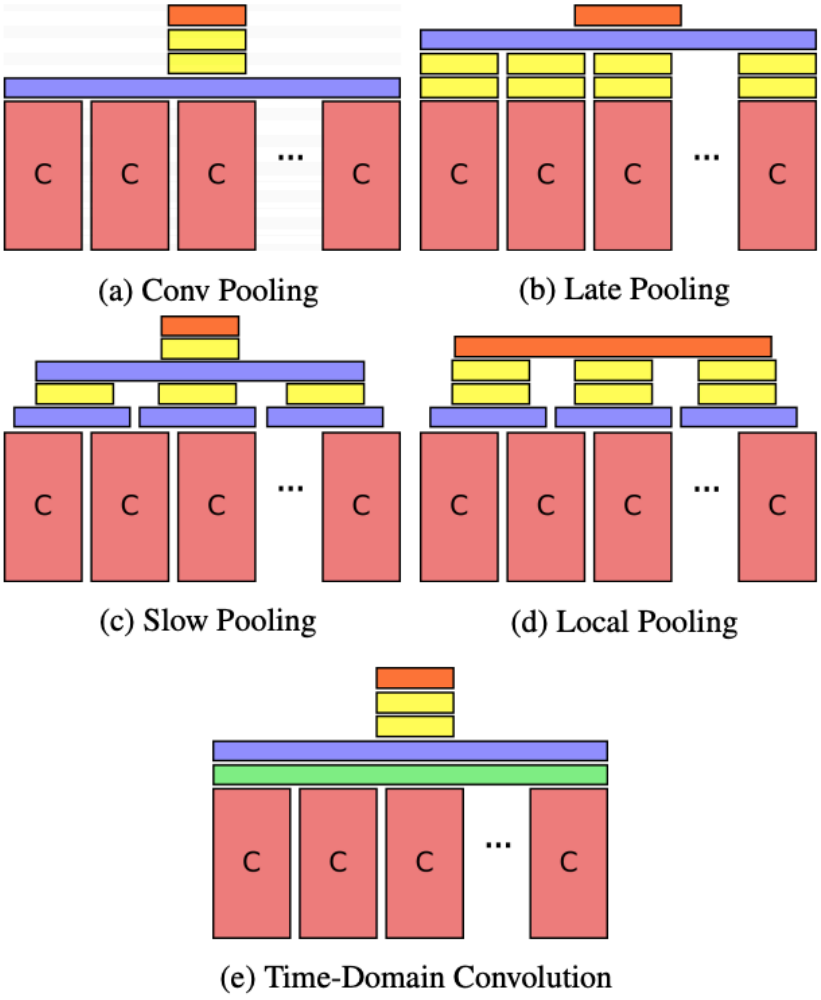


Fig. 1.10: Architecture of stacked convolution

1.5.2　Variational Autoencoder

本研究では，このオートエンコーダの派生である．Variational Autoencoder(VAE) を利用した．これはオートエンコーダの「エンコーダ」と「デコーダ」は同じネットワーク構造であるが，データセットの潜在変数の分布が，正規分布になるような制約を加えて学習を行う手法である．こうすることで Autoencoder の潜在変数では分布の距離に意味ないが，VAE では正規分布に埋め込まれるため，画像の類似度を分布が表現することができるところが特徴である．

1.5.3　敵対的生成ネットワーク

敵対的生成ネットワーク (Generative Advarsarial Network: GAN) は 2014 年に提案された手法である．Figure 1.11 のように GAN では Generator と Discriminator の 2 つのネットワークがある．Generator は訓練データと同じような画像を生成するネットワークで Discriminator は，入力されたデータが訓練データから来たものか Generator で生成されたものかを識別するように学習する．VAE よりも GAN の方が細部まで鮮明に画像を生成す

ることができる．しかし GAN は計算時間がかかるという問題や，Discriminator か Generator のどちらかか強くなってしまうなど，学習が安定しない問題があるため，これについての多くの研究報告がされている．

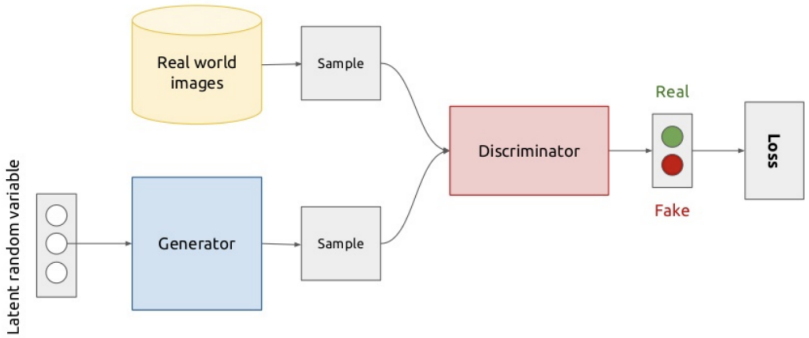


Fig. 1.11: Diagram of GAN

1.6　半教師あり学習

弱教師あり学習とも呼ばれる．これは，教師あり学習と教師なし学習を組み合わせて学習する方法である．こうすることでデータに教師ラベルをつけているものが少数であっても，データの特徴を学習しながら少量のラベルで識別境界を決めることができる．GAN や VAE の考え方を発展させてネットワークを構築することが考えられる．

引用文献

[1] Y. Lecun *et al*., Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (1998) 2278–2324.

[2] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, p. 2012.

[3] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, CoRR abs/1409.1556.

[4] C. Szegedy *et al*., Going deeper with convolutions, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2015.

[5] K. He *et al*., Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.