Gradient Based Learning Applied to Document Recognition

備考

要約

最初の優勢CNNモデルであるLeNetについて書かれた論文.

元論文

著者

Yann LeCun, Leon Bottou, Yoshua Bengio, Patrick Haffner

掲載

Proceedings of the IEEE, Vol. 86, No. 11, pp. 2278-2324, 1998.

Abstract

バックプロパゲーションアルゴリズムで学習させた多層ニューラルネットは、勾配学習の成功例である。適切なネットワークアーキテクチャがあれば、勾配に基づく学習アルゴリズムを用いて複雑な決定面を合成し、最小限の前処理で手書き文字のような高次元パターンを分類することが可能である。本論文では、手書き文字認識に適用される様々な手法をレビューし、標準的な手書き数字認識タスクで比較する。特に、2次元形状のばらつきを扱うために設計された畳み込みニューラルネットワークが、他のすべての手法よりも優れていることが示されている。

現実の文書認識システムは、フィールド抽出、セグメンテーション、認識、言語モデリングなど複数のモジュールで構成されている。グラフ変換ネットワーク（GTN）と呼ばれる新しい学習パラダイムにより、このようなマルチモジュールシステムを勾配ベースの手法で全体的な性能指標を最小化するようにグローバルに学習させることができる。

ここでは、オンライン手書き文字認識のための2つのシステムについて述べる。実験により、グローバルな学習の利点と、グラフ変換ネットワークの柔軟性が示された。

また、銀行小切手を読み取るためのグラフ変換ネットワークについても説明する。このシステムでは、畳み込みニューラルネットワークの文字認識器とグローバルな学習技術を組み合わせることで、企業や個人の小切手に対して記録的な精度を実現する。これは商業的に展開され、一日あたり数百万枚の小切手を読み取っている。

Introduction

ここ数年、機械学習技術、特にNNに適用される技術は、パターン認識システムの設計においてますます重要な役割を果たすようになってきている。実際、連続音声認識や手書き文字認識などのパターン認識アプリケーションの最近の成功は、学習技術の利用が決定的な要因であったと言える。

本論文では、より優れたパターン認識システムを構築するためには、自動学習への依存度を高め、手作業で設計されたヒューリスティックを減らすことが重要であることを伝えたい。これは、近年の機械学習とコンピュータ技術の進歩により可能となった。文字認識を例にとり、手作業による特徴抽出を、ピクセル画像を直接操作する注意深く設計された学習機械に置き換えることで、有利に認識できることを示す。また、文書認識を例にとり、個別に設計されたモジュールを手動で統合して認識システムを構築する従来の方法は、大域的な性能基準を最適化するためにすべてのモジュールを学習できるGTNと呼ばれる統一的でよく原理的な設計パラダイムに置き換えることができることを示す。

パターン認識の初期段階から、音声、文字、その他の種類のパターンなど、自然データの多様性と豊かさのために、完全に手作業で正確な認識システムを構築することはほとんど不可能であることが知られてきた。そのため、ほとんどのパターン認識システムは、自動学習技術と手作業で作成されたアルゴリズムを組み合わせて構築されている。通常のパターン認識では、システムを図1のように大きく2つのモジュールに分割している。最初のモジュールは特徴抽出器と呼ばれ、入力パターンを低次元ベクトルまたは短い記号列で表現できるように変換する。1) 照合や比較が容易であること、2) 入力パターンの変形や歪みに対して比較的不変であり、入力パターンの性質を変えないこと。特徴抽出器は、事前知識のほとんどを含み、むしろタスクに特有である。また、完全に手作業で作成されることが多いため、設計労力の大半を占めるのも特徴である。一方、分類器は汎用的で学習可能であることが多い。このアプローチの主な問題点の1つは、認識精度が、設計者が適切な特徴量のセットを考え出す能力によって大きく左右されることである。これは、残念ながら、新しい問題のたびにやり直さなければならない困難な作業であることが判明した。パターン認識に関する多くの文献は、特定のタスクに対する異なる特徴セットの相対的な利点を説明し、比較することに費やされている。

歴史的に、適切な特徴抽出器が必要とされたのは、分類器が用いる学習技術が、容易に分離可能なクラスを持つ低次元空間に限定されていたためである[1]。この10年間、3つの要因が重なり、このビジ

ョンは変化した。まず，高速な演算装置を備えた低コストなマシンが利用可能になったことで，アルゴリズムの改良よりも，よりブルートフォースな「数値的」手法に依存することができるようになった．第二に、手書き文字認識のように、市場が大きく、多くの人が関心を寄せる問題に対して大規模なデータベースが利用可能になったことで、設計者は認識システムを構築するために、手作業による特徴抽出を減らし、実データに依存することができるようになった。第三に、高次元の入力を処理できる強力な機械学習技術が利用可能になり、これらの大規模データセットが与えられたときに複雑な決定関数を生成できるようになったことが非常に重要な要因である。近年の音声・手書き文字認識システムの精度向上は、学習技術と大規模な学習データセットへの依存度が高まったことに大きく起因していると言える。その証拠に、最近の商用OCRシステムの多くは、バックプロパゲーションで学習させた多層NNを使用しています。

本研究では、手書き文字認識の課題を検討し（第I節、第II節）、手書き数字認識のベンチマークデータセットにおいて、いくつかの学習手法の性能を比較する（第III節）。より自動的な学習は有益であるが、どのような学習手法もタスクに関する最低限の事前知識なしには成功しない。多層NNの場合、知識を取り入れる良い方法は、そのアーキテクチャをタスクに合わせて調整することである。第II節で紹介した畳み込みNN[2]は、局所的な接続パターンを用い、重みに制約を与えることによって、2次元形状の不変性に関する知識を取り入れた特殊なNNアーキテクチャの一例である。第III部では、孤立手書き文字認識のためのいくつかの手法の比較について述べる。さらに、文字単位での認識から文書中の単語や文の認識に至るまで、複数のモジュールを組み合わせて全体の誤差を小さくするという考え方を第IV節で紹介する。手書き文字のような可変長のオブジェクトを複数モジュールで認識する場合、モジュールが有向グラフを操作することができれば、最適な認識が可能となる。このことは、同じく第IV章で紹介する学習可能なGTNという概念につながる。セクションVでは、単語や他の文字列を認識するための、今や古典的なHOSの方法について説明する。第VI節では、手動によるセグメンテーションとラベリングを必要とせずに、単語レベルで認識器を学習するための識別および非識別勾配に基づく技術を紹介する。第VII節では、入力上の全ての可能な位置で認識器をスキャンすることにより、セグメンテーションヒューリスティックの必要性を排除する、有望なスペースディスプレースメントNNアプローチを紹介する。第VIII節では、学習可能なGTNは一般的なグラフ構成アルゴリズムに基づく複数の一般化された変換として定式化できることが示される。また、音声認識でよく用いられるGTNとHMMの関連についても扱う。第IX節では、ペン型コンピュータに入力された手書き文字を認識するためのグローバルに学習されたGTNシステムについて説明する。この問題は、ユーザーが書いたものを機械が即座にフィードバックする必要があるため、「オンライン」手書き認識として知られている。このシステムの中核は畳み込みNNである。この結果は、認識器を事前にセグメント化され、手でラベル付けされた孤立文字で訓練するのではなく、単語レベルで訓練することの利点を明確に示している。第X節では、手書きと機械印刷の銀行小切手を読むためのGTNベースの完全なシステムについて説明する。このシステムの中核は、第II節で述べたLeNet-5と呼ばれる畳み込み型NNである。このシステムは、NCR社の銀行業界向け小切手認識システムで商用利用されている。このシステムは、全米のいくつかの銀行で、毎月数百万枚の小切手を読み取っている。

A. Learning from Data

機械学習の自動化にはいくつかのアプローチがあるが、近年NNコミュニティで普及した最も成功したアプローチの1つは、「数値」または「勾配に基づく学習」と呼ぶことができる。学習機械は関数$Y^p = F(Z^p, W)$を計算する。ここでZ_pはp番目の入力パターンであり、Wはシステムの調整可能なパラメータの集合を表している。パターン認識の場合、出力Y^pはパターンZ^pの認識されたクラスラベルとして、あるいは各クラスに関連するスコアや確率として解釈することができる。損失関数$E^p = D(D^p, F(W, Z^p))$は、パターンZ^pの「正しい」または望ましい出力と、システムが生成した出力の間の不一致を測定する。平均損失関数$E_{train}(W)$は、学習集合$(Z^1, D^1), \ldots, (Z^p, D^p)$というラベル付き例集合に対する誤差$E^p$の平均値である。最も単純な設定では、学習問題は$E_{train}(W)$を最小化する$W$の値を求めることからなる。 実際には、学習集合に対するシステムの性能はあまり興味がない。より適切な指標は、実際に使用されるであろう現場でのシステムのエラー率である。この性能は，テスト集合と呼ばれる訓練集合から切り離されたサンプル集合に対する精度を測定することによって推定される．多くの理論的・実験的研究[3]-[5]により，テスト集合の期待誤差E_{test}と学習集合の誤差E_{train}の差は，学習サンプル数とともに，およそ次のように減少することが分かっている．

$$E_{test} - Etrain = k(h/P)^{\alpha} \tag{1}$$

ここで、Pは学習サンプル数、hは「有効容量」または機械の複雑さの尺度[6],[7]，αは0.5から1.0までの数、kは定数である。このギャップは、学習サンプル数が増加すると必ず減少する。さらに、容量hが増加するとE_{train}は減少する。したがって、容量hを増加させる場合、E_{train}の減少とギャップの増加はトレードオフの関係にあり、最も低い汎化誤差E_{test}を達成する容量hが最適値となる。ほとんどの学習アルゴリズムでは、E_{train}を最小化すると同時に、ギャップのある推定値を最小化しようとする。これの正式なバージョンは構造的リスク最小化[6]、[7]と呼ばれ、各サブセットが前のサブセットのスーパーセットであるようなパラメータ空間のサブセットのシーケンスに対応する、容量増加の学習機のシーケンスを定義することに基づいている。実用的には、構造リスク最小化は$E_{train} + \beta H(W)$を最小化することで実現される。ここで、$H(W)$は正則化関数、βは定数と呼ばれる関数である。$H(W)$は、パラメータ空間の高容量部分集合に属するパラメータW上で大きな値をとるように選ぶ。$H(W)$を最小化することで、アクセス可能なパラメータ空間の部分集合の容量を制限し、それにより、学習誤差の最小化と学習誤差とテスト誤差の期待ギャップの最小化とのトレードオフを制御することができる。

B. Gradient-Based Learning

ある関数をあるパラメータで最小化するという一般的な問題は、コンピュータサイエンスにおける多くの問題の根底にある。勾配に基づく学習は，一般に離散（組合せ）関数よりも適度に滑らかな連続関数の方がはるかに容易に最小化できることを利用している．パラメータ値の小さな変化が損失関数に与える影響を推定することにより、損失関数を最小化することができる。これは、パラメータに対

する損失関数の勾配によって測定される。勾配ベクトルが（摂動による数値計算ではなく）解析的に計算できるようになると、効率的な学習アルゴリズムが考案される。これが連続値のパラメータを持つ多くの勾配に基づく学習アルゴリズムの基礎となっている。本稿では、パラメータWの集合は実数値のベクトルであり、それに対して$E(W)$は連続であり、かつほぼどこでも微分可能であることを示す。このような設定における最も単純な最小化手順は、Wが以下のように反復的に調整される勾配降下アルゴリズムである。

$$W_k = W_{k-1} - \epsilon \frac{\partial E(W)}{\partial W} \tag{2}$$

最も単純な場合、ϵはスカラー定数である。より洗練された方法では、変数ϵを使用するか、対角行列に置き換えるか、ニュートン法または準ニュートン法のように逆ヘシアン行列の推定値に置き換えます。また、共役勾配法[8]も用いることができる。しかし、付録Bによれば、文献上では反対の主張が多いが、これらの2次法の大規模学習機に対する有用性は非常に限られたものであることがわかる。

一般的な最小化手法として、オンラインアップデートとも呼ばれる確率的勾配アルゴリズムがある。これは，平均勾配のノイズの多い（または近似された）バージョンを用いてパラメータベクトルを更新するものです．このアルゴリズムの最も一般的な例では，Wは1つのサンプルに基づいて更新される

$$W_k = W_{k-1} - \epsilon \frac{\partial E^{pk}(W)}{\partial W} \tag{3}$$

この方法では、パラメータベクトルは平均的な軌跡を描いて変動するが、通常、冗長なサンプルを含む大規模な学習セット（音声認識や文字認識のようなもの）では、通常の勾配降下法や2次法よりもかなり高速に収束することができる。この理由については付録Bで説明する。このようなアルゴリズムの学習への応用は1960年代から理論的に研究されてきたが [9]-[11] 、非自明な課題に対する実用的な成功は80年代中頃までなかった。

C. Gradient Back Propagation

勾配に基づく学習法は1950年代後半から用いられてきたが、そのほとんどが線形システムに限られたものであった[1]。このような単純な勾配降下法が複雑な機械学習課題に対して意外に有用であることは、以下の3つの出来事が起こるまで広く認識されていなかった。最初の出来事は、初期の警告[12]に反して、損失関数のローカルミニマムの存在は実際には大きな問題ではないようだ、ということに気づいたことであった。これは、ボルツマンマシンのような初期の非線形勾配に基づく学習技術[13]、[14]において、ローカルミニマムが大きな障害になっていないように思われることに気づいたときに明らかになったものである。第二の出来事は、Rumelhartら[15]によって、幾層もの処理からなる非線形システムの勾配を計算する簡単で効率的な方法、すなわちバックプロパゲーションアルゴリズムが一般化されたことである。第三の出来事は、シグモイド単位を持つ多層NNにバックプロパゲーショ

ンを適用することで、複雑な学習課題を解決できることを実証したことである。逆伝播法の基本的な考え方は、出力から入力への伝搬によって勾配を効率的に計算することである。この考え方は1960年代初頭の制御理論の文献に記載されていたが[16]、機械学習への応用は当時は一般に認識されていなかった。興味深いことに、NNの学習における逆伝播法の初期の導出は、勾配ではなく、中間層のユニットに対する「仮想目標」[17]、[18]、あるいは最小外乱論[19]を用いていた。制御理論の文献で用いられているラグランジュ形式は、逆伝播の導出[20]や、リカレントネットワーク[21]や異種モジュールのネットワーク[22]に対する逆伝播の一般化の導出に、おそらく最も厳密な方法を提供する。一般的な多層システムに対する簡単な導出はセクションI-Eに示す。

多層NNの場合、ローカルミニマムが問題にならないようだが、これは理論的にやや謎である。これは、ネットワークがタスクに対して大きすぎる場合（実際には通常そうである）、パラメータ空間に「余分な次元」が存在することで、到達できない領域のリスクが減少するためと推測される。逆伝播法は、ニューラルネットワークの学習アルゴリズムとして最も広く用いられており、おそらくあらゆる形式の学習アルゴリズムの中で最も広く用いられている。

D. Learning in Real Handwriting Recognition Systems

孤立した手書き文字の認識は、文献上広く研究されており（レビューとしては[23]と[24]を参照）、NNの初期の成功例の1つである[25]。第III章では、手書き文字の認識に関する比較実験を行った。その結果、勾配に基づく学習で学習したNNの性能が、同じデータでテストした他の全ての手法よりも優れていることが示された。最適なNNは畳み込みネットワークと呼ばれ、ピクセル画像から直接関連する特徴を抽出するよう学習するように設計されている（セクションIIを参照）。

しかし、手書き文字認識の最も難しい問題の一つは、個々の文字を認識するだけでなく、単語や文の中で隣接する文字と文字を分離することである（セグメンテーションと呼ばれる処理）。これを行うための技術として、「標準」となっているのがHOSと呼ばれるものである。これは、ヒューリスティックな画像処理技術を使って文字間のカットの候補を多数生成し、その後、認識器によって各候補文字に与えられたスコアに基づいてカットの最適な組み合わせを選択するものである。このようなモデルでは、システムの精度は、ヒューリスティックにより生成されたカットの品質と、認識器が正しく分割された文字と文字の断片、複数の文字、またはその他の正しく分割されていない文字を区別する能力に依存する。このタスクを実行するための認識器のトレーニングは、正しく分割されていない文字のラベル付きデータベースを作成することが困難であるため、大きな課題となる。最も簡単な方法は、文字列の画像をセグメンテーションソフトで処理し、すべての文字仮説に手作業でラベルを付けることである。しかし、この作業は非常に面倒でコストがかかるだけでなく、一貫したラベリングが困難である。例えば、切り取った4の右半分を1としてラベル付けすべきか、それとも非文字としてラベル付けすべきか。8の右半分は3と書くべきか？

第V章で説明する最初の解決策は、文字レベルではなく、文字列全体のレベルでシステムを学習させることである。この目的のためには、勾配に基づく学習の概念を用いることができる。システムは、

誤答の確率を測定する全体的な損失関数を最小化するように学習される。セクションVでは、損失関数が微分可能であるため、勾配に基づく学習法の使用に適していることを確認するための様々な方法について検討する。セクションVでは、代替仮説を表現する方法として、弧が数値情報を持つ有向無サイクルグラフの使用を紹介し、GTNの考え方を導入している。

第VII章で説明する2つ目の解決策は、セグメンテーションを完全に排除することである。このアイデアは、入力画像上のあらゆる可能な場所に認識器を掃引し、認識器の「文字スポット」特性、すなわち、入力フィールドに他の文字が存在する場合でも、中心のない文字を含む画像を拒否しながら、中心のある文字を正しく認識する能力に頼ることである [26], [27]. 認識器を入力に対してスイープすることによって得られる認識器出力のシーケンスは、次に、言語的制約を考慮に入れ、最終的に最も可能性の高い解釈を抽出するGTNに供給される。このGTNはHMMに似ているため、古典的な音声認識[28], [29]を彷彿とさせるアプローチである。この手法は一般にかなり高価であるが、畳み込みNNの使用により、計算コストの大幅な削減が可能となり、特に魅力的である。

E. Globally Trainable Systems

前述したように、実用的なパターン認識システムの多くは、複数のモジュールで構成されている。例えば、文書認識システムは、フィールドロケータ（関心領域の抽出）、フィールドセグメンテータ（入力画像を文字候補の画像に切り出す）、認識器（各文字候補の分類とスコアリング）、文脈ポストプロセッサ（一般に確率文法に基づく）（認識器が生成した仮説の中から文法的に正しい答えを選択する）により構成されている。ほとんどの場合、モジュールからモジュールへ伝達される情報は、アークに数値情報が付加されたグラフとして表現するのが最適である。例えば、認識モジュールの出力は、各アークに候補文字のラベルとスコアが含まれ、各パスが入力文字列の代替解釈を表す非循環グラフとして表現することができる。通常、各モジュールは、その文脈の外で、手動で最適化され、時には訓練される。例えば、文字認識器はあらかじめ分割された文字のラベル付き画像で学習される。その後、システム全体を組み立て、モジュールのパラメータのサブセットを手動で調整し、全体のパフォーマンスを最大化する。この最後のステップは非常に退屈で、時間がかかり、ほぼ確実に最適とは言えない。

より良い代替案は、文書レベルでの文字誤判定確率のようなグローバルなエラー指標を最小化するように、何らかの方法でシステム全体を学習させることである。理想的には、システムのすべてのパラメータに関して、このグローバルな損失関数の良い最小値を見つけたいと思う。性能を測定する損失関数Eがシステムの調整可能なパラメータWに対して微分可能であれば、勾配に基づく学習を用いてEの局所最小値を見つけることができる。しかし、一見したところ、システムの大きさと複雑さによって、この作業は困難であるように思われる。

グローバルな損失関数$E^p(Z^p, W)$が微分可能であることを保証するために、システム全体は微分可能なモジュールのフィードフォワードネットワークとして構築されている。各モジュールが実装する関数は、モジュールの内部パラメータ（例えば、文字認識モジュールの場合はNN文字認識器の重

み）に関して、またモジュールの入力に関して、ほぼどこでも連続的で微分可能である必要がある。この場合、よく知られたバックプロパゲーション法の単純な一般化により、システム内のすべてのパラメータに関する損失関数の勾配を効率的に計算することができる[22]。例えば、モジュールのカスケードとして構築されたシステムを考えてみよう。各モジュールは関数$X_n = F_n(W_n, X_{n-1}$を実装し、X_nはモジュールの出力を表すベクトル、W_nはモジュールの調整可能なパラメータ(Wのサブセット)、X_{n-1}はモジュールの入力ベクトル(および前のモジュールの出力ベクトル)である。最初のモジュールへの入力X_0は、入力パターンZ^pである。X_nに関するE^pの偏導関数が既知であれば、W_nとX_{n-1}に関するE^pの偏導関数は後方回帰を用いて計算することができる。

$$\frac{\partial E^p}{\partial W_n} = \frac{\partial F}{\partial W}(W_n, X_{n-1})\frac{\partial E^p}{\partial X_n} \tag{4}$$

$$\frac{\partial E^p}{\partial X_{n-1}} = \frac{\partial F}{\partial X}(W_n, X_{n-1})\frac{\partial E^p}{\partial X_n} \tag{5}$$

ここで、$(\frac{\partial F}{\partial W})(W_n, X_{n-1})$は点$(W_n, X_{n-1}$で評価した$F$の$W$に関するヤコビアン、$(\frac{\partial F}{\partial X})(W_n, X_{n-1})$は$F$の$X$に関するヤコビアンとします。ベクトル関数のヤコビアンは、すべての入力に対するすべての出力の偏導関数を含む行列である。最初の式は$E^p(W)$の勾配のいくつかの項を計算し、2番目の式はNNのよく知られた逆伝播法のように、後方回帰を生成する。この勾配を学習パターンにわたって平均化することで、完全な勾配を得ることができる。多くの場合、ヤコビアン行列を明示的に計算する必要がないことに注目すると面白い。上式はヤコビアンと偏微分のベクトルの積を用いるが、ヤコビアンをあらかじめ計算せずに、この積を直接計算した方が簡単な場合が多い。通常の多層NNになぞらえて、最後のモジュール以外は出力が外部から観測できないため、隠れ層と呼ばれる。上記のような単純なモジュールのカスケードよりも複雑な状況では、偏微分の表記がやや曖昧で厄介になる。より一般的な場合における完全に厳密な導出は，ラグランジュ関数 [20]-[22] を用いて行うことができる.

従来の多層NNは、状態情報が固定サイズのベクトルで表現され、モジュールが行列の乗算（重み）と成分単位のシグモイド関数（ニューロン）の交互積層である特殊なケースであった。しかし、前述のように複雑な認識系では、状態情報を円弧に数値情報を付加したグラフで表現するのが最適である。この場合、GTと呼ばれる各モジュールは、1つ以上のグラフを入力として受け取り、グラフを出力として生成する。このようなモジュールのネットワークは GTN と呼ばれる。第IV、VI、VIII章では、GTNの概念を展開し、勾配に基づく学習により、大域的損失関数を最小化するように、すべてのモジュールのパラメータを学習することができることを示す。状態情報がグラフのような本質的に離散的なオブジェクトで表現されるときに勾配が計算できることは逆説的に見えるかもしれないが、後で示すようにその困難は回避できる。

2. 孤立文字認識のための畳み込みニューラルネットワーク

勾配降下法で学習した多層ネットワークは、大量の例から複雑な高次元の非線形マッピングを学習できるため、画像認識タスクの候補として明らかである。従来のパターン認識では、手作業で設計された特徴抽出器が入力から関連情報を収集し、無関係な変数を排除する。次に、学習可能な分類器が、結果として得られる特徴ベクトルをクラスに分類する。この方式では、標準的な完全連結型多層ネットワークを分類器として用いることができる。より興味深いのは、特徴抽出器自身の学習にできるだけ依存する方式である。文字認識の場合、ほぼ生の入力（例えば、サイズ正規化された画像）をネットワークに与えることができる。これは通常の完全連結フィードフォワードネットワークで可能であり、文字認識のようなタスクではある程度成功するが、問題もある。

まず、一般的な画像は大きく、数百の変数（ピクセル）を持つことが多い。**完全接続された第1層の隠れユニットが例えば100個であれば、すでに数万個の重みが含まれていることになる。このような多数のパラメータはシステムの容量を増加させるため、より大きな学習セットを必要とする。** また、これほど多くの重みを保存するために必要なメモリが、ある種のハードウェア実装を除外する可能性もある。しかし、**画像や音声アプリケーション用の非構造化ネットの主な欠点は、入力の並進や局所的な歪みに対する不変性が組み込まれていないことである。文字画像やその他の2次元、1次元の信号をNNの固定サイズ入力層に送る前に、サイズをほぼ正規化し、入力フィールドの中央に配置する必要がある。** しかし、手書き文字は単語単位で正規化されるため、個々の文字の大きさ、傾き、位置がバラバラになってしまうことがある。そのため、手書き文字が単語単位で正規化され、個々の文字の大きさや傾き、位置のばらつきが生じ、さらに書き方のばらつきも加わって、入力オブジェクトの特徴的な位置がばらばらになってしまう。原理的には、十分な大きさの完全連結ネットワークがあれば、このようなばらつきに対して不変な出力を生成するように学習することができる。しかし、そのような学習を行うと、入力のどこに特徴があっても検出できるように、入力の様々な位置に同じような重みパターンを持つユニットを複数配置することになるだろう。このようなウェイトパターンを学習するためには、可能なバリエーションを網羅するために非常に多くの学習インスタンスが必要となる。**後述の畳み込みネットワークでは、空間的に重みの配置を強制的に複製することで、自動的にシフト不変性を得ることができる。**

第二に、**完全連結型アーキテクチャの欠点は、入力のトポロジーが完全に無視されることである。** 入力変数は、学習の結果に影響を与えることなく、どのような（固定された）順序で提示されてもよい。これに対して、画像（あるいは音声の時間周波数表現）は強い2次元局所構造を持っており、空間的あるいは時間的に近接する変数（あるいはピクセル）は高い相関を持つ。局所相関は、空間的または時間的オブジェクトを認識する前に局所特徴を抽出・結合することのよく知られた利点の理由であり、近傍変数の構成は少数のカテゴリ（例えば、エッジ、コーナーなど）に分類できるためである。畳み込みネットワークは、隠れユニットの受容野を局所的に制限することで、局所特徴の抽出を強制している。

A. 畳み込みニューラルネットワーク

畳み込みネットワークは、1) 局所受容野、2)重みの共有(または重みの複製)、3) 空間的または時間的サブサンプリング、という3つのアーキテクチャのアイデアを組み合わせて、ある程度のシフト、スケール、歪みの不変性を保証している。図2にLeNet-5と呼ばれる典型的な文字認識用畳み込みネットワークの例を示す。入力面には、大きさがほぼ正規化され、中心が定まった文字の画像が入力される。各層のユニットは、前の層の小さな近傍に位置するユニット群から入力を受ける。入力の局所的な受容野にユニットを接続するというアイデアは1960年代初めのパーセプトロンにさかのぼるが、それはHubelとWieselが猫の視覚系で局所的に感度の高い、方向選択性のニューロンを発見したのとほぼ同時だった[30]。局所結合は視覚学習の神経モデルで何度も用いられてきた[2], [18], [31]-[34]．局所受容野を持つニューロンは、エッジ、終点、コーナーなどの基本的な視覚的特徴（あるいは音声スペクトログラムなど他の信号における同様の特徴）を抽出することができる。これらの特徴は、高次の特徴を検出するために、後続の層によって結合される。前述したように、入力の歪みやずれにより、顕著な特徴の位置が変化することがある。また、画像の一部分で有効な素性検出器は、画像全体でも有効である可能性が高い。この知識は、画像上の異なる場所に受容野を持つユニットの集合に、同一のウエイトベクトルを持たせることで応用できる[15], [32], [34]．層内のユニットは、すべてのユニットが同じ重みのセットを共有する平面で編成されている。このような平面上のユニットの出力の集合を特徴マップと呼ぶ。特徴マップのユニットはすべて、画像の異なる部分に対して同じ操作を行うように制約されている。完全な畳み込み層は、複数の特徴マップ（異なるウェイトベクトル）から構成され、各位置で複数の特徴を抽出することができる。具体的な例として、図2に示すLeNet-5の第1層がある。LeNet-5の第1隠れ層のユニットは6つの平面で構成されており、それぞれが特徴マップとなっている。特徴マップのユニットは、25個の入力が、そのユニットの受容野と呼ばれる入力中の5×5個の領域に接続されている。各ユニットは25個の入力を持つため、25個の学習可能な係数と学習可能なバイアスを持つ。特徴マップの連続したユニットの受容野は、前の層の対応する連続したユニットを中心とする。そのため、隣接するユニットの受容野は重なり合う。例えば、LeNet-5の第1隠れ層では、水平方向に連続するユニットの受容野は4列5行に渡って重なっている。前述したように、特徴マップのすべてのユニットは25個の重みのセットとバイアスを共有しているので、入力上のすべての可能な位置で同じ特徴を検出する。層内の他の特徴マップは異なる重みとバイアスを使用し、異なるタイプの局所的特徴を抽出する。LeNet-5の場合、6つの特徴マップの同じ場所にある6つのユニットによって、各入力位置で6種類の特徴が抽出される。特徴マップを逐次的に実装する場合、局所受容野を持つ1つのユニットで入力画像を走査し、そのユニットの状態を特徴マップの対応する位置に格納することになる。この操作は畳み込みと等価であり、その後に加算バイアスとスカッシュ関数が続くため、畳み込みネットワークと呼ばれる。畳み込みのカーネルは、特徴マップのユニットが使用する接続重みの集合である。畳み込み層の興味深い特性は、入力画像がシフトした場合、特徴マップの出力も同じだけシフトするが、それ以外の場合は変更されないことである。この性質が、入力のずれや歪みに対する畳み込みネットワークの頑健性の基礎となっている。

一度検出された特徴は、その正確な位置はあまり重要ではなくなります。他の特徴との相対的なおおよその位置だけが重要である。例えば、入力画像の左上にほぼ水平なセグメントの終点があり、右上に角があり、画像の下部にほぼ垂直なセグメントの終点があることが分かれば、入力画像は7である

ことが分かります。これらの各特徴の正確な位置は、パターンの識別に無関係であるだけでなく、その位置が文字の異なるインスタンスに対して異なる可能性があるため、潜在的に有害である。特徴マップの中で特徴的な位置が符号化される精度を下げる簡単な方法は、特徴マップの空間解像度を下げることである。これはいわゆるサブサンプリング層で実現でき、局所平均とサブサンプリングを行うことで特徴マップの解像度を下げ、シフトや歪みに対する出力の感度を下げることができる。LeNet-5の第2隠れ層は、サブサンプリング層である。この層は6つの特徴マップからなり、前の層の各特徴マップに1つずつ対応する。各ユニットの受容野は、前の層の対応する特徴マップの2 2の領域である。各ユニットは4つの入力の平均を計算し、訓練可能な係数を掛け、訓練可能なバイアスを加え、その結果をシグモイド関数に通す。連続したユニットは重複しない連続した受容野を持つ。その結果、サブサンプリング層の特徴マップは前の層の特徴マップの半分の行と列を持つ。学習可能係数とバイアスは、シグモイド非線形性の効果を制御する。係数が小さい場合、ユニットは擬似線形モードで動作し、サブサンプリング層は単に入力をぼかすだけである。係数が大きい場合、サブサンプリングユニットはバイアスの値によって「ノイズの多いOR」または「ノイズの多いAND」関数を実行していると見ることができる。畳み込みとサブサンプリングの連続した層は、通常「バイ・ピラミッド」となって交互に繰り返され、各層で、空間解像度が低くなるにつれて、特徴マップの数が増加する。図2の第3隠れ層の各ユニットは、前の層のいくつかの特徴マップからの入力接続を持つことができる。この畳み込み／サブサンプリングの組み合わせは、HubelとWieselの「単純」「複雑」細胞の概念にヒントを得て、福島のNeocognitron [32] で実装されたが、当時は逆伝播法のようなグローバルな教師あり学習はなかった。このように空間分解能を漸減させながら、表現の豊かさ（特徴マップの数）を漸増させることにより、入力の幾何学的変換に対して大きな不変性を得ることができる。

すべての重みは逆伝播で学習されるため、畳み込みネットワークは自分自身の特徴抽出器を合成していると見ることができる。重みの共有は、自由なパラメータの数を減らすという興味深い副作用があり、それによって機械の「容量」を減らし、テストエラーと学習エラーの間のギャップを小さくすることができる[34]。図2のネットワークは345,308の接続を持つが、重み共有のため、訓練可能な自由パラメータは60,000に過ぎない。

固定サイズの畳み込みネットワークは、手書き文字認識[35]、[36]、機械印字文字認識[37]、オンライン手書き認識[38]、顔認識[39]など、多くのアプリケーションに適用されてきた。また、1つの時間次元に沿った重みを共有する固定サイズの畳み込みネットワークは、時間遅延NN（TDNN）として知られている。TDNNは音素認識（サブサンプリングなし）[40]、[41]、話し言葉認識（サブサンプリングあり）[42]、[43]、孤立手書き文字のオンライン認識[44]、署名検証[45]に利用されてきた。