深層学習の多彩な画像応用

岡　谷　貴　之

〈東北大学情報科学研究科／理化学研究所 AIP センター　〒980–8579 仙台市青葉区荒巻字青葉 6〉
e-mail: okatani@vision.is.tohoku.ac.jp

AI（人工知能）の急速な発展をもたらしてきた深層学習は，近年，AIに限らない様々なサイエンス・工学の汎用的な問題解決方法としての地位を固めつつある．天文学やその周辺の問題解決に役立ちそうな（と著者が勝手に考える）深層学習のコンピュータビジョン（つまり，画像を扱うAI）の分野における適用事例をいくつか取り上げ，それぞれの技術的なハイライトを紹介する．

1. はじめに

著者の専門はコンピュータビジョン，つまり画像を扱うAIである．天文学については全くの素人であり，この分野でAIの技術に関するどのような需要があるのかよくわかっていない．にもかかわらず，本特集のために「機械学習のアルゴリズムや応用例についてのレビュー」を執筆してもらえまいかという依頼を引き受けた背景には，AIの技術――要するに深層学習のこと――が，AIの応用の枠を超えて，普遍的な問題解決の方法としての地位を確立しつつあり，そのことをなるべく多くの方々と共有したいという思いがある．

例には，タンパク質の3次元構造をアミノ酸塩基配列から予測するAlphafold [1]や，深層強化学習による核融合の制御[2]を挙げることができる．もちろん，AIの最近の飛躍的な進化――人との自然な対話や高度なQ&Aをこなし，プログラムの自動生成を可能にすると期待される「言語モデル」[3]など――も，深層学習がもたらしている．

これらの成功例に共通するのは，深層学習がもたらす技術の非連続的な跳躍である．跳躍の高さと応用の多様さは，普遍的な問題解決の方法としての深層学習の重要性を，強く感じさせる．理工系の大学の学部教育の共通科目として，機械学習やデータサイエンスに加えて，深層学習を取り入れることになるのではないか，と感じる．

以上のような思いを持ちつつ本稿では，主にコンピュータビジョン（平たく言えば「画像を扱うAI」のこと）の分野での深層学習の適用例を，いくつか紹介する．具体的には，画像変換，密な画像間対応付け，疎な画像間対応付け，新規視点画像合成，の4つの応用を取り上げる．いずれもコンピュータビジョンの特定の問題ではあるが，これらを通じて，深層学習がどんな風に様々な問題に適用されているかを紹介したい．

なお深層学習は，「ニューラルネットワークを用いた機械学習の方法」と一般的に捉えられるが，最近はおよそニューラルネットワークには見えないものへと進化している．核心にある技術的要素を挙げれば，i）「微分可能な（differentiable）演算」を，ii）解きたい問題に合わせて自在に組み合わせて「モデル（＝ネットワーク）」を設計し，iii）設計の手に負えない部分をデータを用いた学習に委ね，vi）学習は，誤差逆伝播法＋勾配降下法で行う，ということになる[4]．

2. 深層学習のあらまし

2.1 ネットワークの構造

上述のように，今では「ニューラルネットワーク」には見えない構造のネットワーク（以下「モデル」と呼ぶ）を用いることが珍しくなくなっている．古典的なニューラルネットワークは，層を積み重ねた均質な構造を持ち，どんな写像も表現しうる「万能」学習器として位置付けられ，データを用いた学習を通じ，望みの写像を表現することが期待されていたと言える．

その後，入力データの形式に応じた専用設計のモデルが開発された．画像のような「配列データ」を対象とする畳み込みニューラルネットワーク（CNN），系列データを対象とするリカレントネットワーク（RNN）は当然として，集合データ——入力となる集合内の要素の順序に結果が影響されない——に対するDeepSets [5]やPointNet [6]などのモデルや，Transformer [7, 8]，そしてグラフで表現されたデータを対象とするグラフニューラルネットワーク（GNN）[9]が作られた．

これらは，学習にすべてを丸投げするのではなく，問題に関する知識を取り込んで学習を補おうとする，いわゆる「帰納バイアス」であると言える．さらに一歩進め，解きたい問題やデータの性質に合わせてモデルの詳細構造を設計する方法が，一般化している．この傾向は，画像への応用で特に顕著であり，後で紹介するのはそういった例である．

モデルの設計は，構造の単位となる「ブロック」を用意し，これを積み木のように組み合わせ，1つの大きな構造をデザインするように行う．個々のブロックは，その入力xから出力x'へのある写像$x \to x' = f(x; w)$を実行する．wはパラメータで，これを変えると写像が変化する．

ブロックを組み合わせて全体構造を作る基本的な（ただし唯一でない）方法は，ブロックをカスケードにつなぎ，（深）層構造を作ることである．ブロックを積み重ねてL層とした構造を考えると，$x^{(0)} = x$から始めて，$l = 1, \ldots, L$の順に，1つ下の層の出力$x^{(l-1)}$を入力に得て，これに演算を加えて，次のようにこの層の出力を得る．

$$x^{(l)} = f^{(l)}(x^{(l-1)}; w^{(l)}) \tag{1}$$

上の計算を繰り返した最後に$y = x^{(L)}$を得，全体で見れば$x \to y$の写像が実現される．

ブロックの計算$x' = f(x; w)$の中身は，古典的なニューラルネットワークでは，$x' = \sigma(Wx + b)$，つまりxの線形変換後に要素ごとに非線形写像（活性化関数）を適用する計算だったが，今や$x' = f(x; w)$はどんなものでも構わない．正確には，「微分可能」でありさえすればよい．

「微分可能」であるとは，i）出力x'をwで微分できることと，ii）x'をxで微分できることを指す．（i）の微分が計算できれば，学習（＝勾配降下法）でこのブロックのパラメータwを定められる．また，（ii）の微分が計算できれば，このブロックの上流（入力側）に別のブロックが接続されているとき，それのブロックに勾配を伝播でき，そこにあるパラメータも同様に学習で定められる．すべてのブロックがこのような微分可能性を備えていれば，モデル全体に散りばめられたパラメータをすべて，学習の対象とできることになる．

2.2 勾配に基づく学習

以上のような写像$x \to y$を与えるモデルのパラメータを，データを用いた学習で定める．学習の目的は，モデルが望みの写像を表現するよう，パラメータwを定めることである．そのために，写像の事例，つまり入出力ペア(x, y)を多数用意し，その入出力関係をなるべく忠実に再現するよう，モデルのパラメータwを定める（教師あり学習）．

具体的には，入力xに対する目標出力をtと書くと，これらのペアの集合

$$\mathcal{D} = \{(x_1, t_1), \ldots, (x_N, t_N)\} \tag{2}$$

を用意し，x_nに対するモデルの出力$y_n=y(x_n; w)$とその目標t_nの差をD全体にわたって求めた

$$L(w,\mathcal{D})=\frac{1}{N}\sum_{n=1}^{N} d(t_n,y_n) \qquad (3)$$

が小さくなるようにwを求める．$d(t, y)$はtとyの差を測る関数である．これは次の大規模非線形最小化問題

$$\min_{w} L(w,\mathcal{D}) \qquad (4)$$

に帰着される．最小化を行う方法には一般には様々な選択肢があるが，深層学習では勾配降下法一択である．これはひとえにwの成分数が大きいせいで，例えば画像分類では標準的に数千万程度以上[10]，自然言語で近年一般化した大規模言語モデルに至っては数千億に及ぶ[3]．

勾配降下法は，wの空間で，現在の解において目的関数Lが最も小さくなる方向，すなわち勾配$-\nabla_w L$の方向に解を更新する．モデルを構成するブロックがすべて，上述のように微分可能であれば，すべてのパラメータについて$-\nabla_w L$計算でき，勾配降下法が実行可能となる．

なお，以上の「教師あり学習」が主流だが，そのほかにも，目標出力を明示的には与えない無教師学習，正解が自明なタスク（＝pretextタスク）を学習する自己教師あり学習[11]，教師あり学習と無教師学習との中間的な弱教師学習なども知られているが，ここでは省略する．

3. 事　　例

3.1 画像変換

画像変換は，画像xを別の画像yに変換する問題である（図1）．ノイズ除去，焦点ズレや物体の動きによるボケ（blur）の除去，霞や霧の除去，降雨の除去，影（shade）の除去，低解像度画像の解像度を向上させる超解像化，ハイダイナミックレンジ画像生成（HDRI），スタイル変換――例えば実写シーンを絵画風に変換――など幅広い応用が含まれる（図2）．

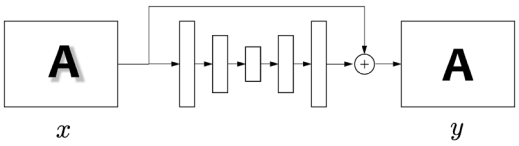

図1　画像変換の概念図と基本的なモデルの構造．入力をそのまま出力に伝達するスキップ接続を配置し，出力yと入力xの差分（画像の修正分）を予測する．

図1のように，画像変換で最も一般的なモデルは，入力から出力に至る間に複数の畳み込み層を配置し，この間ダウンサンプリングとアップサンプリングを伴いつつ，フィルタと畳み込みを何度も行う構造である．ノイズ除去のように入力と出力の差分が小さい場合は，その差分のみを予測の対象とすべく，図のように入力をそのまま出力に加算するスキップ接続（あるいは残差（residual）接続）を用いる．問題ごとに様々な知見を活かした，様々なモデルの構造が編み出されている．

学習は，変換前と変換後の画像ペア(x, y)の集合Dを用意し，これを用いて教師あり学習を行うのが最も一般的である．損失には，正解画像tとその予測yとの画素値の差の総和$\|t-y\|_p$（$p=1, 2$のp-ノルム）を用いるのが基本である．

対象が自然画像である場合，出力画像がより自然なものとなるように，物体認識を学習済みのCNNの中間層における表現，すなわちyとtをそれぞれこのCNNに入力した際の，中間層の特徴ベクトル間の差――認識（perceptual）損失と呼ぶ――を用いることがある．さらに画像の自然さを追求し，敵対的生成ネットワーク（GAN）の考え方を適用し，画像が合成されたものか，本物かを見極める識別器（discriminator）を導入し，これを画像変換を行うモデルと同時に，互いに競い合わせながら学習することもよく行われる[13]．ただし「自然さ」を追求し過ぎると，出力画像が「自然には見えるものの実際にはフェイク（＝作り物）」になってしまうので，用途に

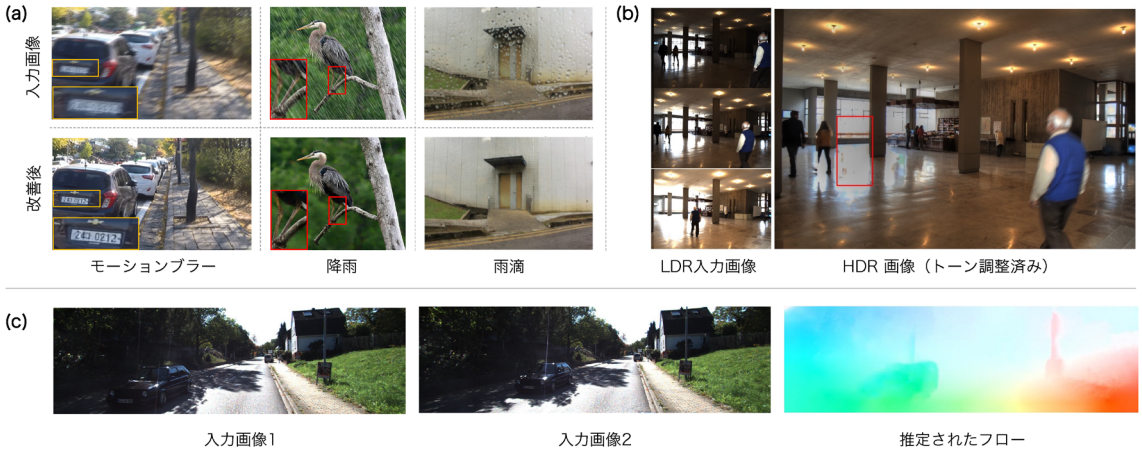

図2　（a）画像変換の画質改善への応用例．（b）ハイダイナミックレンジイメージング[12]．露出設定を変えて撮影した3枚の画像を入力に，高ダイナミックレンジの画像を生成する．動く物体の影響を受けていないことに注意．（c）連続撮影された画像からのオプティカルフローの推定．画像間でカメラは前方に移動しており，それに伴う画像上のフローベクトルが推定される．フローベクトルはその向きと強さが色で表されている（カラー図はweb版を参照）．

よって使い分けられる．

入力と正解のペア（x, t）が手に入らない場合には，その他の方策が取られる．例えばノイズ除去では，Noise2Noise [14] や Noise2Void [15] など，正解tを用いないで学習を行う多様なアプローチが提案されている．また，入力と正解の直接のペア（x, t）は手に入らないが，それぞれの集合（$\{x_n\}$と$\{t_m\}$）ならば得られるというときは，$x \to y \to x'$という循環の一致性を学習の手がかりとするCycleGANが利用できる[16]．

3.2　密な画像間対応付け

1つのシーンを異なる位置・姿勢にあるカメラで撮影した2枚の画像x_1, x_2上で，シーンの同じ点がどこに写っているかを知る「画像間対応付け」は，コンピュータビジョンの基本的な問題である．

典型的な問題にオプティカルフローの推定がある．これは動画像の撮影時，撮影対象がカメラと相対的に運動する場合，異なる時刻の画像x_1とx_2の間で，x_1に写るシーンの各点がx_2では少し移動することになるが，この移動量を推定する問題である（図1(c)）．移動量を（Δi, Δj）と書くと，シーンの同一点がx_1, x_2上で同じ濃淡値を持つと仮定して

$$x_1(i,j) = x_2(i+\Delta i, j+\Delta j) \tag{5}$$

という等式を得る．これに基づいて，各画素（i, j）の移動量（Δi, Δj）＝（p（i, j）, q（i, j））を求めたい．この問題は，画像処理的には，（i, j）周りの正方領域のパッチP_{ij}について，濃淡の差が最小になるような移動量（p, q）

$$(p,q) = \underset{(p,q)}{\operatorname{argmin}} \sum_{(k,l)\in P_{ij}} \| x_1(i+k, j+l) - x_2(i+p+k, j+q+l) \|_2^2, \tag{6}$$

を見つける問題に帰着される．ただし，テクスチャ（模様）が希薄な画像の部位では，このような（p, q）を定めようがないことがある．また，画像間の撮影条件が変わると（5）式が成り立たなくなることもあり，あるいは視点の移動に伴い他の物体に遮蔽されるなど，上の最小化だけではうまくいかない．深層学習以前は，様々な制約（たとえばフローの空間的な滑らかさなど）を導入し，画像全体での（p, q）に関する最適化に帰

着させることが一般的であった．ただし大規模な非線型最適化となり，計算量の増大や局所解の問題があった．

上述のように深層学習では，あるところまで問題の構造に合わせた計算を作り込み，そこから先を学習に委ねる．この問題の場合は，次のようにモデルの構造を設計する（図3）．まず入力画像x_1とx_2を別々に同じCNNに入力し，それぞれの特徴マップ$u_1(i, j)$, $u_2(i, j)$を得る．特徴マップはサイズ$W \times H \times C$の配列データであり，各座標(i, j)（$0 \le i \le W, 0 \le j \le H$）において$C$個の成分を持つベクトル$u_1, u_2$を格納したものである．

この特徴ベクトルを用いて，(6)式と同様の考え方で，x_1の点(i, j)とx_2の$(i+p, j+q)$（のまわりの濃淡パタン）の類似度を，例えばベクトルの内積$u_1(i, j)^\top u_2(i+p, j+q)$で測ったとき，この類似度が最も高くなるような$(p, q)$を，各$(i, j)$で推定することを考える．

この推論をCNNで簡単に行えるように，事前にあらゆる(p, q)の組み合わせに対する類似度を機械的に計算し，中間的なデータとして表現しておく．具体的には，網羅的な（ただし必要な範囲での）組み合わせ(p, q)に，インデックス$m=1, \ldots$を振り，(p_m, q_m)（$m=1, \ldots$）と書く．そして次を計算しておく．

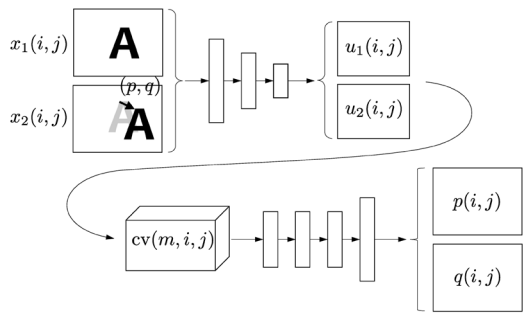

図3　オプティカルフローの推定．入力画像それぞれから特徴マップu_1, u_2を得て，そこからコストボリュームを機械的に作り，2段目のCNNに入力し，座標(i, j)でのフローベクトル(p, q)を出力．

$$\mathrm{cv}(m, i, j) = u_1(i, j)^\top u_2(i+p_m, j+q_m) \quad (7)$$

このcv(m, i, j)をコストボリュームと呼ぶが，これがあれば，各位置(i, j)において上の類似度が大きくなる(p, q)を得る問題は，m（つまり(p_m, q_m)）を選ぶだけで済むことになる．さらに，(p, q)が空間的に（隣り合う(i, j)と比べて）滑らかとなるようなもの――欲を言えば，さらに濃淡変化や遮蔽への配慮もなされたもの――を推定できるようにしたい．そこで，このコストボリュームを入力すると，$p(i, j)$と$q(i, j)$が出力されるようなCNNを考える．

そしてこのCNNを教師あり学習，つまり予測した(p, q)とその正解とのずれが小さくなるよう，最初の特徴ベクトルuを取り出すCNNと一緒に学習する．学習は，CGで合成した画像（正解となるフロー(p, q)が得られる）を数千から数万枚程度用いて行う．

以上の深層学習を用いたオプティカルフロー推定は，学習こそ時間を要するものの（最新のGPUサーバを用いて数時間～数日程度），推論時はモデル内の順方向の伝播1度だけで済むため，従来の非線型全体最適化（一般に反復計算を要する）に比べ，圧倒的に短い計算時間で，しかもより高精度にフローを算出できるようになった．

3.3　疎な画像間対応付け

上と類似した問題に，画像内で疎に選んだ点どうしを，画像間で対応づける問題がある（図4）．オプティカルフローとの違いは，同一シーンをかなり離れた視点から撮影した2枚の画像を対象とすることである．多視点画像を用いた3次元的な幾何学計算，いわゆるSfM（Structure from Motion）や視覚SLAM（Simultaneous Localization And Mapping）などの入口となる問題である．これらの技術は現在，例えばドローンで撮影した地上構造物の3次元測量や，自動車の自動運転や運転支援において，カメラ映像を基に車両の自己位置・姿勢を実時間計算する問題に応用されている．

さて，この問題では，正確さや効率性を重視し，キーポイントと呼ばれる画像上の少数（～数百・数千）の「特徴的な点」を選び，それらを画像間で対応付ける．キーポイントには，どの方向から見ても位置が正確に決められ，また画像間で紛れることなく対応付けができるよう，点周りの局所的な濃淡構造が独特であるような点を選ぶ．このキーポイントの選択・決定にも深層学習が使われるが，ここではキーポイントを画像間で対応付ける問題を考える．

キーポイントが1, 2枚目の画像でそれぞれK_1, K_2個あるとする．図5のように，最初の画像のキーポイント$i(=1,\ldots,K_1)$について，上と同様，特徴ベクトル$u_{1i}(\in\mathbb{R}^C)$を取り出し，2枚の画像のキーポイント$j(=1,\ldots,K_2)$についても同様に，特徴ベクトル$u_{2j}(\in\mathbb{R}^C)$を取り出す．オプティカルフローと同様，これらの類似度$u_{1i}^{\top}u_{2j}$がなるべく高いペア(i,j)を選びたい．

ただしそのとき，画像全体での整合性も考慮する必要がある．具体的には，i）2枚の画像間でのキーポイント対応には満たすべき制約があり（詳細は省く），これを満たさない点どうしを対応づけてしまうと誤りとなる．また，ii）画像間でキーポイントは必ず1対1で対応しなければならない．つまり1対多やその逆になることはあり得ない．

以上の条件（i）と（ii）を満たしつつ，個々の対応の類似度が全体として高くなるようなものを見出したい．SuperGlue [17]は，この問題を解く1つの方法であり，（i）の解決にグラフニューラルネットワーク（GNN）を用い，（ii）の解決にSinkhornアルゴリズムを用いている．

GNNは，キーポイント間で特徴ベクトルと画像内の位置を比較し，（i）を満たすような，対応すべきキーポイント同士が，より近い特徴ベクトルuを持つよう，特徴ベクトルを修正する．「修正する」といってもそのやり方が明らかであるわけではなく，正確には，「画像間の正しい対応が予測できる」ような学習を通じて，そんな修正能力が獲得されることを期待する．

一方，Sinkhornアルゴリズムは，「最適輸送問題」を近似的に解く方法であり，ここでは1対1の割り当てを解くために導入されている．このアルゴリズム自体には学習の対象となるパラメータは存在せず，ただ決まった計算を行うだけである．

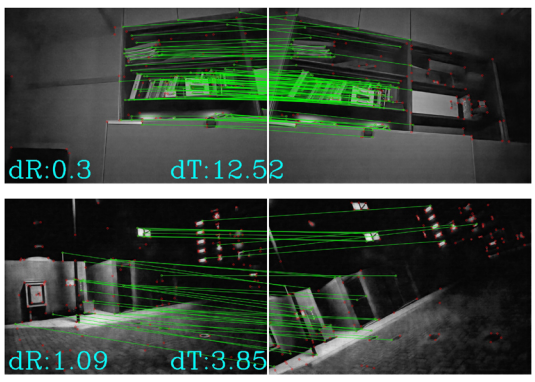

図4　異なる視点の画像間でシーンの同一点を対応づける問題の実行例（カラー図はweb版を参照）．

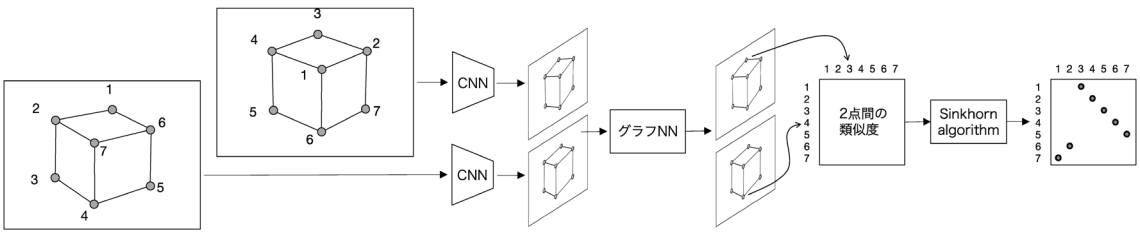

図5　同一シーンの異なる視点からの画像間で，疎な点の対応を求める問題とそれを深層学習で解くSuperGlue[17]の概要．1対1の割り当てにSinkhornアルゴリズムを用いることで，勾配降下法でモデル全体の学習が可能となる．

ただし，アルゴリズムの出力が入力で微分できるという意味で「微分可能」であるため，学習を通じて，上流のGNNや，入力直後の特徴ベクトルを取り出すCNNまでも，一緒に学習可能である．

SuperGlueは，以上の構造を持つモデルを用い，キーポイントとその画像特徴（＋位置情報）を入力し，それらキーポイントの対応（対応するものがないものもある）を正しく予測できるよう，その正解を与えて学習を実行する．問題に合わせて，Sinkhornアルゴリズムという微分可能な「1対1割り当て」の演算要素を取り入れつつ，全体が巧みに設計されているところが興味深い．

3.4 新規視点画像合成と陰的表現

あるシーンをいくつかの方向から撮影した複数枚の画像を使って，新しい（撮影していない）視点からの画像を合成する問題を，新規視点画像合成と言う．様々な異なるアプローチがあるが，ここでは，空間における輝度の分布（radiance field）をニューラルネットワークで表現するNeRF（Neural Radiance Field）と呼ぶ方法を説明する[18]．なおここでは，各画像を撮影したカメラの位置・姿勢を含む投影パラメータは既知であるとする（つまり画像の点を指定するとその点の空間の光線が分かる条件）．

さて，空間の点(x, y, z)とその点を見る方向(θ, ϕ)の計5つの値を入力に受け取り，その点の密度（〜物体の有無）ρと色を表すRGB値の，計4つの値を出力するような写像を考える．今，対象とするシーンについて，そんな写像$(x, y, z, \theta, \phi) \to (\rho, r, g, b)$がわかっていれば，同シーンの任意の視点の画像を生成できる．具体的にはその画像の各画素について，空間の光線を手前から奥まで走査し，光線上の各点で(ρ, r, g, b)を得て，ρが大きな値をとる点での(r, g, b)値を，その画素の画素値とすればよい（ボリュームレンダリング）．

NeRFでは，この写像をニューラルネットワークで表現し，与えられた多視点画像のそれぞれについて，このようなレンダリングによって画像が再現されるように，ネットワークを学習する．つまり，再現画像yと実際の画像tの差$\| y-t \|^2$を最小化し，ネットワークのパラメータを定める．このとき，上のボリュームレンダリングの計算が「微分可能」であるようにして，上の差の最小化は，誤差逆伝播で計算した勾配をもちいた勾配降下法で実行可能となる．

ネットワークには，10層程度の全結合層からなる古典的な構造のものを用いる．ただし，上述の(x, y, z, θ, ϕ)を直接ネットワークの入力としてしまうと，ネットワークの表現力が足りず，シャープな画像の再現は叶わないことが知られている．そこで，これら5つのパラメータをより多くの値で冗長に表現する方法（位置エンコーディングと呼ぶ）が考案された．これにより，学習に用いた視点の画像のみならず，未知の視点での画像をも，高い品質で合成することができるようになった（図6）．

以上のNeRFは，新規視点画像生成方法として画期的であった．ニューラルネットワークが果たす役割は比較的単純で，写像$(x, y, z, \theta, \phi) \to (\rho, r, g, b)$を，入力の標本点（入力画像の各画素が与える光線）について学習させ，与えていない入力の点に対する出力を，内挿によって計算しているだけとみなせる．しかしながら，そんな簡単な内挿によって高品質な自由視点画像を作れることは知られておらず，深層学習の知られざる可能性を新たに知らしめるものとなった．また，画像のレンダリングが微分可能であることに目をつけ，

図6　NeRFによる新規視点画像合成の様子．入力画像枚数6, 24, 48枚の場合の結果．

与えられた画像が忠実に再現されるように学習するだけで，目的が果たされる点も新しかった．

NeRFのように，1つの信号$s(x)$をニューラルネットワーク$s(x)\sim y=f(x; w)$のように表現する方法は，様々な目的で使われている．NeRFでは，離散的な座標$\{x_n\}_{n=1, 2, \ldots}$での標本値$\{s_i\}_{i=1, 2, \ldots}$を，$y=f(x; w)$で再現・記憶すべく学習することで，$x\neq x_n$の$s(x)$を予測（＝内挿）している．このとき，標本値$s_n=s(x_n)$そのものの代わりに，各x_nで$s(x)$が満たす何らかの条件

$$F_m(s, \nabla_x s, \nabla_x^2 s, \ldots)=0, m=1, \ldots, M \qquad (8)$$

を与え，これを満たすようにネットワークを学習し，結果的に$s(x)$を表現させる方法もある．上の条件は$s(x)$を「陰に（implicit）」に指定しているとみなせるため，「（ニューラルネットワークによる）陰的な表現（implicit neural representation）」と呼ばれる[19]．なお，(8)の制約が$s(x)$の微分を含む場合は，以上の方法は微分方程式を解いていることにも相当する．

この方法は，例えば物体の3次元形状を表現するのに使われている[20, 21]．1つの物体の形状を，任意の空間の点xから物体表面までの符号付き距離（signed distance function, 以下SDF）$s(x)$——物体の外と中で距離の符号を変化させるようにしたもの——で表す．例えば，物体表面を計測して得た離散的な点群（point cloud）が与えられたとき，SDF$s(x)$をニューラルネットワークで表現するよう学習する．物体の形状を効率よく，かつ高精度に表現する方法として知られている．

参考文献

[1] Senior, A. W., et al., 2020, Nature, 577, 706
[2] Degrave, J., et al., 2022, Nature, 602, 414
[3] Brown, T. B., et al., 2020, arXiv preprint arXiv: 2005.14165
[4] 岡谷貴之, 2022, 深層学習（MLPシリーズ）第2版（講談社）
[5] Zaheer, M., et al., 2017, arXiv preprint arXiv: 1703.06114
[6] Qi, C. R., et al., 2017, Proc. CVPR, 652
[7] Vaswani, A., et al., 2017, arXiv preprint arXiv: 1706.03762
[8] Lee, J., et al., 2019, Proc. ICML, 3744
[9] Zhou, J., et al., 2020, AI Open, 1, 57
[10] He, K., et al., 2016, Proc. CVPR, 770
[11] He, K., et al., 2020, Proc. CVPR, 9729
[12] Ye, Q., et al., 2021, Proc. ACMMM, 5290
[13] Isola, P., et al., 2017, Proc. CVPR, 1125
[14] Lehtinen, J., et al., 2018, arXiv preprint arXiv: 1803.04189
[15] Krull, A., et al., 2019, Proc. CVPR, 2129
[16] Zhu, J.-Y., et al., 2017, Proc. ICCV, 2223
[17] Sarlin, P.-E., et al., 2020, Proc. CVPR, 4938
[18] Mildenhall, B., et al., 2020, Proc. ECCV, 405
[19] Sitzmann, V., et al., 2020, NeurIPS, 33
[20] Mescheder, L., et al., 2019, Proc. CVPR, 4460
[21] Park, J. J., et al., 2019, Proc. CVPR, 165

Colorful Deep Learning Applications to Computer Vision

Takayuki Okatani

Graduate School of Information Science, Tohoku University/RIKEN Center for AIP, 6 Aramaki Aza Aoba, 980–8579 Sendai, Japan

Abstract: Deep learning, which has led to the rapid development of artificial intelligence (AI), has recently consolidated its position as a general-purpose problem-solving method in various sciences and engineering, not limited to AI. In this paper, the author discusses several applications of deep learning in computer vision (i.e., AI for image processing) that may be useful for solving problems in astronomy and related fields and then introduces the technical highlights of each application.