

# ロボットの運動制御における深層学習の応用

## Application of Deep Learning in Robot Motion Control

東北大・工（非）\*沓澤 京

Kyo Kutsuzawa\*

\*Tohoku University

### 1. はじめに

深層学習はここ 10 年ほどにわたって幅広い分野で注目を集めている。画像処理や自然言語処理では飛躍的に優れた性能を挙げて実用化が進んでいる一方で、未だ応用への課題がある領域も多い。本稿では、深層学習の簡単な説明とともに、著者らが関わった応用例としてロボットの運動制御への適用事例を紹介する。

### 2. 深層学習

深層学習は階層的な処理によって複雑な概念を構築する、表現学習の一種として位置づけられる<sup>1)</sup>。深層学習の様々な手法すべてを漏れなくまとめる厳密な定義は難しいが、ほとんどの方法は以下の 2 つの特徴を有する。

1. 単純な処理を多層に重ねることで複雑な処理を実現する。
2. 各層のパラメータが勾配法で最適化される。

第一の特徴は深層学習を特徴づける重要な要素であり、これによって複雑かつ抽象的な概念が構築できるといわれる。例えば画像認識においては最初の層では単純な角や輪郭を検出し、続く層では輪郭の特徴量から被写体の構成要素（車輪、目、耳、など）を検出し……といったように低次の特徴量を組み合わせてより複雑で抽象度の高い特徴量を表現できる。この性質は誤差逆伝播法の再発見として知られる Rumelhart らの論文<sup>2)</sup>でも報告されている。階層構造の設計自由度は高く、中間層の次元数を絞込むことで高次元データの低次元な特徴表現を得たり、フィードバック結合を追加することで時系列データを処理したりと、様々なことができる。

第二の特徴は多量のパラメータを更新する方法として有効であり、ほぼすべてのモデルで採用される。各パラメータについて損失関数の勾配を求める方法には誤差逆伝播法が通常使われる。その基本原理は偏微分の連鎖律であり、損失関数と各層の処理とがすべて微分可能でさえあれば勾配が計算できるので、適用範囲が極めて広い。

近年ではスケーリング則<sup>3)</sup>や過剰パラメータ表現による汎化性能向上<sup>4)</sup>などの現象を背景に、多量のデータを使って大規模モデルを構築するという大きな流れがある。一方でそれが難しい分野も多い。その場合は、深層学習の設計自由度の高さを活かして強い帰納バイアス、すなわちタスクに関する専門知識を組み込む必要があると考えられる<sup>5)</sup>。

### 3. ロボット制御への適用事例：動力学的制約に合わせた動作生成

著者らが手掛けた応用事例として、「お好み焼きをひっくり返す動作の生成」について概要を述べる<sup>6,7)</sup>。このタスクは物体操作の一例であるが Fig. 1(a)のような制約が存在し、へらの角度に応じた範囲の加速度を加えないとお好み焼きが滑り落ちてしまう。この事例においては、一種類の軌道を生成すれば良いのではなく、様々な軌道を生成することを目標とした。実環境では目標位置（ひっくり返す位置）や速度の制限、動作範囲の制限などの状況が様々あると考えられるため、それらに合わせて軌道を生成する能力は必須である。

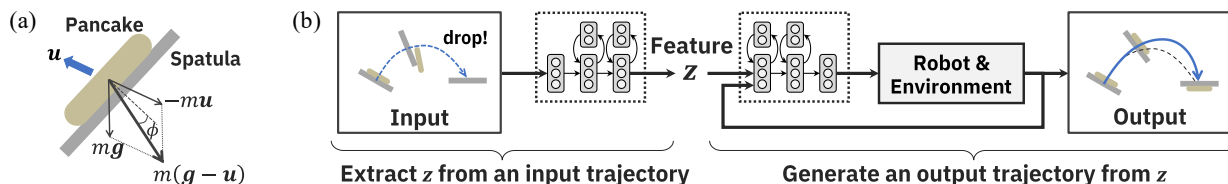


Fig. 1 Overview of the method. (a) Physics model of a pancake.  $u$ ,  $g$ , and  $\phi$  indicate acceleration, gravity, and angle of friction, respectively. (b) Architecture of the neural network.

このタスクを遂行するため、著者らは「与えられた軌道をお好み焼きを滑り落とさない軌道に変換する」というアプローチで、Fig. 1(b)に示す構造のニューラルネットワークを実装した<sup>8)</sup>。このモデルは入力軌道を変えることで様々な軌道が得られ、入力軌道をつくる際には動力学的制約を考えなくても良い。このモデルは 2 つのリカレントニューラルネットワークから構成され、その間では一連の運動の時系列をひとつの特徴量  $z$  に圧縮表現している。

このモデルを学習させる際の損失関数として、以下のものを用いた。

$$L = \frac{1}{T} \sum_{t=0}^{T-1} (\|x_t - y_t\|^2 + kL_c), \quad L_c = \min(0, \phi - \arctan \mu_{\max}) \quad (1)$$

ここで、 $T$ は時系列長、 $x_t$ と $y_t$ は入出力軌道中のへらの位置、 $k$ は損失の重み係数、 $\mu_{\max}$ は静止摩擦係数である。こ

の損失関数の第1項は軌道の再現誤差であり、入出力軌道が一致するように学習が進む。第2項の $L_c$ は動力的制約のペナルティであり、静止摩擦の範囲を超えたときにペナルティが生じるような関数になっている。この構造の損失関数を使うことで、ランダムな軌道を入力すれば出力軌道は入力軌道を再現しつつも動力的制約を満たすものになっていくと期待される。損失関数によって動力的制約を満たすように方向づけられるので、事前に滑り落とさない軌道のデータセットを用意する必要はなく、代わりに計算機内で無制限に学習データを生成することができる。

上述のモデルをさらに活用し、生成される軌道の微調整をおこなう技術も開発した<sup>7)</sup>。具体的には、Fig. 1(b)のモデルの特徴量 $\mathbf{z}$ を変えれば出力軌道も変化することを利用し、出力軌道に対する適当な評価関数 $J$ を使って以下の更新式に従って特徴量 $\mathbf{z}$ を更新する。

$$\mathbf{z} \leftarrow \mathbf{z} - \eta \frac{\partial J}{\partial \mathbf{z}} \quad (2)$$

ここで $\eta$ は学習係数であり、導関数 $\partial J / \partial \mathbf{z}$ は環境の状態方程式のモデル化できていれば誤差逆伝播法で計算できる。

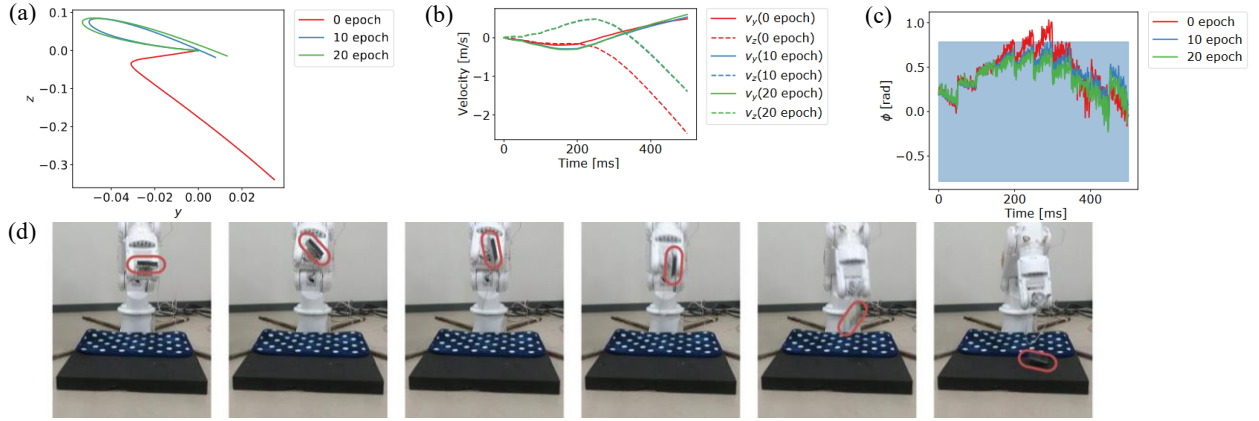


Fig. 2 Movements generated by adjusting the feature<sup>7)</sup>. (a) Trajectories of the spatula. It starts from the origin and rotates  $-3\pi/4$  rad. Epoch indicates the number of updates. (b) Time series of the velocity. (c) Time series of the angle of friction. Pancakes do not slip within the blue area. (d) Snapshots of the robot movement after 20 epochs.

提案手法を用いて軌道の変形および特徴量の調整の効果を検証した。特徴量の次元を 64 とし、ランダムに生成した 1638400 本の軌道を使って訓練した。学習済モデルに対して 0.5 s かけてヘラをその場で  $-3\pi/4$  rad 回転させる軌道を入力し、出力軌道の速度および動作範囲が小さくなるように評価関数を記述して式(2)に従って特徴量を調整した。更新幅は $\eta = 100$ とした。得られた結果を Fig. 2 に示す。Figure 2(a)–(c)の軌道を見ると、単に学習済モデルに軌道を入力するだけでもほぼ滑り落とさないような軌道が出力されたが、運動中に摩擦角が制限をわずかに超えており、さらに移動範囲も速度も大きいためロボットが実行できないものであった。しかし、その軌道をもとに特徴量 $\mathbf{z}$ を更新することで、20 回の更新後には移動範囲と速度が小さくなり、摩擦角も指定範囲内に収まる軌道になった。この軌道は実際のロボットの性能でも再現可能であり、Fig. 2(d)のようにお好み焼きを模した対象物をひっくり返すことに成功した。

#### 4. おわりに

本稿では、深層学習の簡単な概要とともに、ロボット制御への適用事例を紹介した。紹介した事例においては、モデルの構造と損失関数を工夫することで無数の学習データを自動生成できるようにしつつ、モデル内部の特徴量を更新することで出力動作の微調整を実現した。著者らはこの他にも、神経科学の知見をモデル構築に活用して未学習の動作も生成可能な高い汎化能力を実現する<sup>8)</sup>など、様々な方法で深層学習のロボット制御への応用をおこなっている。

#### 文献

- 1) I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, MIT press, 2016.
- 2) D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.
- 3) J. Kaplan, S. McCandlish, T. Henighan, *et al.*, “Scaling laws for neural language models,” *ArXiv preprint*, 2001.08361, 2020.
- 4) Sebastian Bubeck, Mark Sellke, “A universal law of robustness via isoperimetry,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 28811–28822, 2021.
- 5) N. Roy, I. Posner, T. Barfoot, *et al.*, “From machine learning to robotics: challenges and opportunities for embodied intelligence,” *arXiv preprint*, 2110.15245, 2021.
- 6) K. Kutsuzawa, S. Sakaino, and T. Tsuji, “Sequence-to-sequence models for trajectory deformation of dynamic manipulation,” *the 43rd Annual Conference of the IEEE Industrial Electronics Society (IECON 2017)*, 2017, pp. 5227–5232.
- 7) K. Kutsuzawa, S. Sakaino, and T. Tsuji, “Trajectory adjustment for nonprehensile manipulation using latent space of trained sequence-to-sequence model,” *Advanced Robotics*, vol. 33, no. 21, pp. 1144–1154, 2019.
- 8) K. Kutsuzawa and M. Hayashibe, “Imitation learning with time-varying synergy for compact representation of spatiotemporal structures,” *IEEE Access*, vol. 11, pp. 34150–34162, 2023.