

物体把持位置検出における逐次学習手法の検討

鈴木 彼方† 横田 泰斗† 金澤 裕治† 竹林 知善†
† 富士通研究所

1 はじめに

近年、工場における製造ラインの自動化が進められており、特に物体のピッキングタスクへのロボットのタスク学習への深層学習の適用が盛んに行われている [1][2]。中でも深層強化学習 [3] や、試行結果から学習データを逐次的に増やしていく逐次学習 [4] は、大量のデータセットを事前に用意する必要がないことから、注目を集めている。その際、対象物の 3D モデルを事前に準備することで学習を限られた時間内で収束させることや、試行結果を適切にフィードバックすることは、実タスクに適用するにあたり重要である。

本研究では上記に対し、シミュレータで作成した対象物を抽象化したプリミティブな物体の事前学習と、把持成功率に基づく更新量の修正を加えた物体検出ベースの逐次学習手法を提案する。評価実験では提案手法をバラ置きピッキングタスクに適用し、把持位置検出の逐次学習を行った。

2 学習方式

2.1 把持位置検出学習

本研究では物体の把持位置を検出する学習モデルに Single Shot Multibox Detector (SSD)[5] を用いる。SSD は物体検出に用いられる深層学習モデルの一種で、デフォルトで用意される大量のアスペクト比の異なる矩形領域から正解の矩形領域を出力するように学習される。本研究では SSD の出力は物体の把持位置 (x, y, w, θ) を矩形表現 $(bx1, by1, bx2, by2)$ に変更し学習を行う。また、通常の SSD から角度 θ と把持成功率 s を予測し、把持位置か否かの分類を行うよう修正する。これらは正解との二乗誤差 MSE により最適化される。把持位置ラベル (positive) と背景ラベル (negative) を合わせた誤差 $Loss$ は以下で表される。

$$Loss = \sum_b \left(S_b Loss_{pos,b} + \sum_k \alpha_{k,b} Loss_{neg,k,b} \right) \quad (1)$$

$$\alpha_k = \begin{cases} \frac{1}{2} (1.0 - conf_k) & \text{if } k \leq K, \\ 1.0 & \text{otherwise} \end{cases} \quad (2)$$

ここで $conf$ はクラス確率を、 α_k はクラス確率上位 k 番目の矩形に対する係数を表す。係数 α_k はクラス確率上位となるほど小さくなるように更新量の調整を行

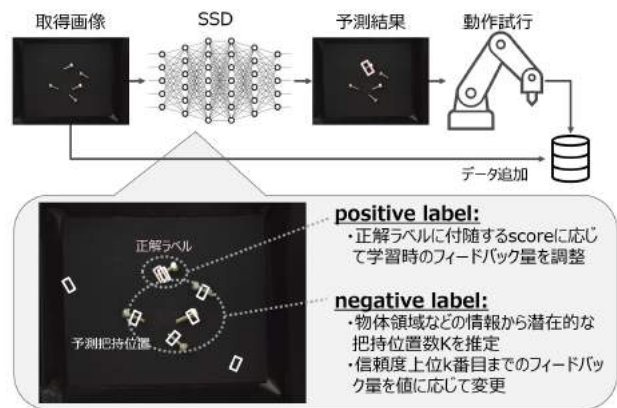


図 1: 提案手法の概要

う。これは後述する逐次学習では逐次的に学習データを追加することから、1 画像に対して 1 つの正解ラベルのみが与えられ、潜在的に正解である領域も不正解 (背景ラベル) と学習してしまう恐れがあるためである。

また K は推定される潜在的な把持位置数を表し、更新量を調整する範囲を決定する。本研究では背景画像との差分法により、物体数 N と物体領域を算出した後、各領域に属する把持領域から求められる。ただし、本研究では、単純化のために一つの物体領域につき一つの把持箇所があるものとしている。

b はミニバッチ内の正解データ番号を表し、 S_b は観測により得られる真の把持成功率を表す。提案手法では各正解矩形に対応する S_b をかけあわせることにより、適切な把持位置が優先的に学習される。

2.2 動作試行による逐次学習

提案手法はモデルにより予測した物体把持位置をロボットにより試行し、学習データを逐次的に追加しながら学習を行う。処理手順の概要を Algorithm. 1 に示す。

はじめに人間により数枚の正解データを作成する。これは実験初期において学習が始まらないことを回避し、逐次学習の正解データ取得を効率化するためである。上記で生成した訓練データセットを用いた学習と並列に、動作試行により逐次学習を行う。学習モデルにより予測された把持位置を、物体領域内に含まれる範囲で把持成功率 s の順に試行し、成否判定を行う。成功であった場合、試行データを訓練データに加える。一方、学習モデルにより予測結果を試行し、一定回数失敗を繰り返した場合、把持可能位置の探索を行う。ここでは物体領域内に含まれる範囲でランダムに決定・試行し、成否判定を行う。上記を繰り返し、試行が一定時間経過したら処理を終了とする。

Self-Supervised Learning for Object Picking Task
Kanata Suzuki, Yasuto Yokota, Yuuji Kanazawa, Tomoyoshi Takebayashi
(Fujitsu Laboratories LTD.)

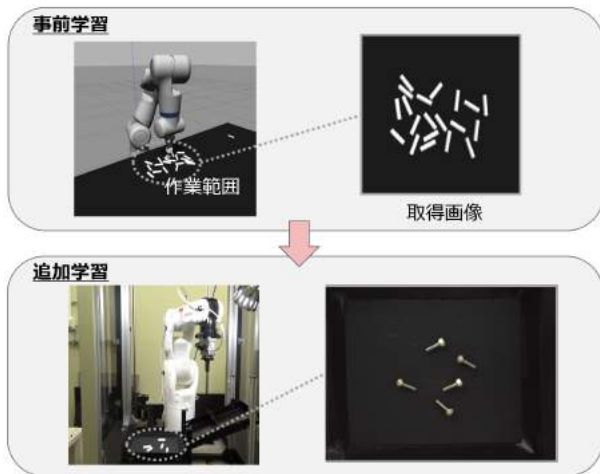


図 2: バラ置きピッキングタスク

Algorithm 1 逐次学習手順

```

1: make training data  $D$ , and start training
2: while ( $CurrentTime \leq MaxTime$ ) do
3:    $i \leftarrow ObjectNum$ 
4:   while ( $i > 0$ ) do
5:     if  $j < MaxTrialNum$  then
6:        $d_{pos}(x, y, w, \theta) \leftarrow d_{pred,s_j}(x, y, w, \theta)$ 
7:        $j \leftarrow j + 1$ 
8:     else
9:        $d_{pos}(x, y, w, \theta) \leftarrow d_{rand}(x, y, w, \theta)$ 
10:       $q_{pos}(x, y, w, \theta) \leftarrow d_{pos}(x, y, w, \theta)$ 
11:      if success to pick up then
12:        add  $d_{pos}(x, y, w, \theta)$  to  $D$ 
13:       $i, j \leftarrow i - 1, 0$ 

```

前記の逐次学習の処理は、実機での検証では非常に時間がかかる。学習時間短縮のために、シミュレータでの事前学習は従来手法 [1][2] で有効性が確認されているが、対象物それぞれにモデルを用意することはコストがかかる。本研究では実物の把持位置学習を行う際には、事前に用意した単純形状の 3D モデルによる学習を事前学習として行った (図 2)。

3 評価実験

3.1 実験設定

実験は円柱 20 個、及び、ボルト 5 個のピッキングタスクを行った。実験環境を図 2 に示す。物体はカメラ画角内のロボットの可動域内の範囲で重なりなく 20 個設置されており、取り切ったら再設置される。本研究では把持成功率は把持位置の中心からの距離とした。SSD への入力は $300 \times 300 \times 3$ [pix] 画像を用い、最適化には Adam を用いた。それぞれのパラメータは $\alpha = 0.001$, $\beta = 0.9$ である。また評価時には、ランダムな位置に設置し、各個数 100 回の試行を行い成功率を算出する。

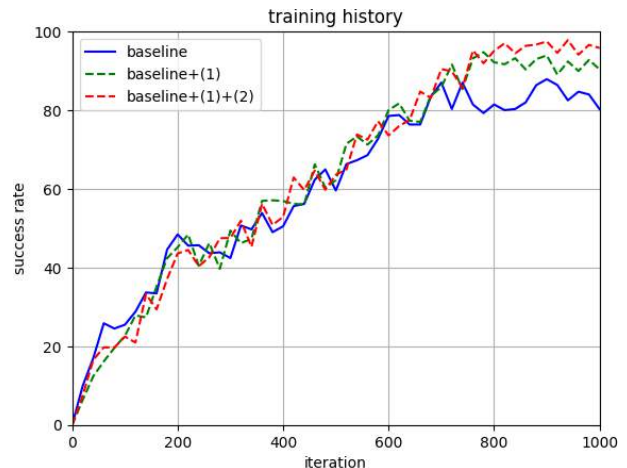


図 3: シミュレータ上での把持成功率

3.2 結果と考察

評価実験として、把持成功率に基づく更新量修正 (1)、潜在的な成功領域の考慮 (2) の影響を確認するため、シミュレータ上で円柱に対して把持学習を行った。学習経過に伴う成功率を図 3 に示す。この時、把持成功条件は予測把持位置が中心から 1.0[cm]、把持角度が 5[deg] 以内、かつ、降下時に物体に触れないこととした。通常の SSD での成功率が 88.4% に対し、(1)(2) を加えた提案手法は成功率 96.8% に達しており、提案手法の有効性が確認できた。

また、実機での実験の際には事前にシミュレータ上で学習した円柱モデルからの追加学習を行った。本実験ではボルトを単純化したプリミティブモデルとして、円柱を扱った。学習 1 時間終了後の成功率は学習初期で 50.0%、経過 60 分で 92.0% となっており、事前学習が学習の効率化に大きく影響することが確認できた。

4 まとめと今後の展望

本研究ではピッキングタスクにおいて、動作成否に基づいて教示ラベルを追加していく逐次学習の効率化を検証した。シミュレータ、及び、実機での評価実験を行い、提案手法の効果が確認された。今後の展望として、汎用的な正解尺度の設計を行う。

参考文献

- [1] J.Mahler, et al., "Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics," in Proc. Robotics: Science and Systems (RSS), 2017.
- [2] K.Bousmalis, et al., "Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping," in Proc. IEEE Int. Conf. Robots and Automation (ICRA), 2018.
- [3] S.Levine, et al., "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," Int. J. Robot. Res., vol.37, no.4-5, pp.421-436, 2018.
- [4] L.Pinto, et al., "Supersizing Self-supervision: Learning to Grasp from 50K Tries and 700 Robot Hours," in Proc. IEEE Int. Conf. Robots and Automation (ICRA), 2016.
- [5] W.Liu, et al., "Ssd: Single shot multibox detector," in European Conf. Computer Vision (ECCV), 2016.