

作業失敗の予測学習に基づく 棚への収納作業における双腕手添え動作の選択的実行

Learning-based Task Failure Prediction and Selective Execution of Dual-arm Support Motion for Stowing Task

北川 晋吾 和田 健太郎 岡田 慧 稲葉 雅幸
Shingo Kitagawa Kentaro Wada Kei Okada Masayuki Inaba

東京大学工学部機械情報工学科
University of Tokyo, Faculty of Engineering, Department of Mechano-Informatics

近年、多様な物品を取り扱う倉庫におけるピッキング作業において、吸引ハンドの有効性が明らかになりつつあるが安定な把持が課題となっている。本研究では、画像を入力とした深層学習を用いて作業失敗を予測し、把持状態に応じて選択的に双腕手添え動作を実行する手法を提案する。検証として提案手法の棚への収納作業における作業の効率化に対する有効性を確認する。

1. はじめに

近年、倉庫におけるピッキング作業の自動化が求められており、Amazon Picking Challenge(APC)に代表されるロボット競技会が開催されている。この作業において、多様な物品を安定的に把持し動作を行うことが大きな課題であり、吸引ハンドを用いた手法の有効性が明らかになりつつある [Correll 16]。しかし、吸引ハンドは1点により物体を把持しているため把持が不安定になりやすく、環境との接触が原因で落下などの作業失敗を引き起こしてしまう場合がある。

本研究では、ピッキング作業の中でも棚への収納作業に着目し、把持物品の安定化のために双腕手添え動作を行うことを提案する。双腕手添え動作とは、既に把持している物品をもう片方の腕で下から支え上げる動作である。しかし、この動作は作業の安定性は向上するものの作業遂行時間は延長されてしまう。そこで、本研究では物品の把持状態をRGB画像によって観測し、必要な場合にのみ手添え動作を行うことを提案する。手添え動作を行うかどうかの選択は、各作業失敗の発生確率を予測し、その予測確率から手添え動作を行う場合と行わない場合の作業安定性について評価して行うことを提案する。各動作を行った場合の作業失敗の発生確率を予測し、各作業失敗に対して減点を設定することで計算できる作業得点の期待値に基づいて動作選択を行う。RGB画像からの作業失敗の発生予測には深層学習を用いた学習手法を用いる。検証実験としては、提案する動作選択システムが作業安定性と作業遂行時間の観点から作業の効率化に有効であることを示す。

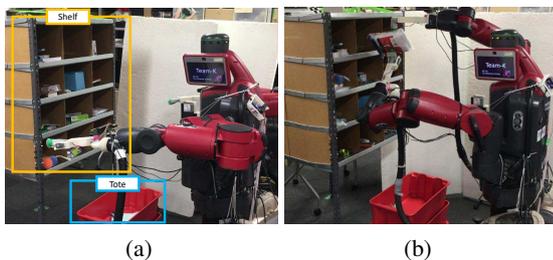


図1: (a) 棚への収納作業とは籠から指定された物品を棚に収納することである。(b) 提案する双腕手添え動作

連絡先: 北川晋吾, 東京大学工学部機械情報システム工学研究室,
113-8656 東京都文京区本郷7-3-1 工学部2号館7階73B2,
03-5841-7416, s-kitagawa@jsk.imi.i.u-tokyo.ac.jp

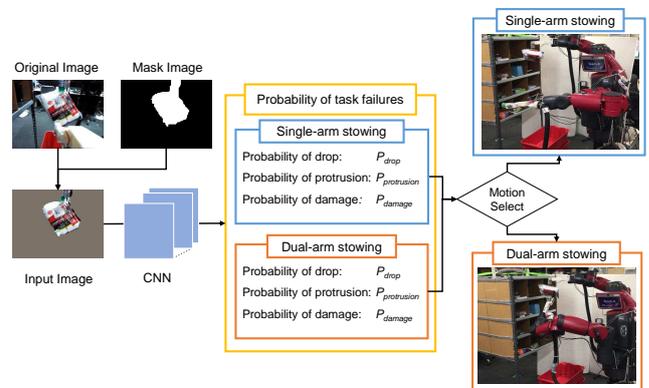


図2: 提案する動作選択システム構成図

2. 提案手法と関連研究

倉庫におけるピッキング作業において、吸引装置による不完全な三次元情報を用いた把持手法が実践されており、ロボット競技大会においても高い成績を収めている [Correll 16, Hernandez 16]。しかし、吸引ハンドによる把持は吸引力が1点のみに作用するため不安定になりやすい。そこで、本研究では単腕による不安定な把持が行われた場合に、もう片方の腕で把持物品を下から支える双腕手添え動作を行う。

双腕によるマニピュレーション動作の利点は先行研究によって指摘されているが [Edsinger 07, Harada 12]、単腕による動作よりも工程が増えるため遂行に時間を要する。倉庫におけるピッキング作業は作業安定性の向上と作業遂行時間の短縮を目指す必要があるため、本研究では物品の把持状態に応じて必要な場合にのみ手添え動作を行う。

動作選択を画像から行う場合には、先行研究では選択可能な作業を行った各場合の作業成功率を Convolutional Neural Network (CNN) を用いて予測し、高い成功率を示す動作を実行する手法が提案されている [Levine 16, Pinto 16]。これらの研究では成功と失敗のみを終了状態として設定しているが、収納作業には落下やみ出しなど作業失敗は複数存在し同時発生しうる。そこで、本研究では各失敗の発生確率を画像を入力としたCNNを用いて予測し、その発生確率から計算できる作業得点の期待値を作業安定性の指標として動作選択に用いる。

3. 双腕手添え動作の設計

収納作業において、物品の落下、損傷、棚からはみ出しの3つの作業失敗を設定し、この作業失敗を回避する双腕手添え動作を設計する。棚からはみ出しとは Fig3(b) に示されるように、作業終了時に棚に物品が完全に収まっておらず、物品の一部が棚の外にはみ出している状態を指す。設計する手添え動作を Fig4 に示す。単腕で把持している物品の重心をもう一方の腕で下から支えあげること、物品の垂れ下がりや抑制し、棚の床と把持物品の接触を回避する。把持の際に物品の重心を吸引しているものとして、手添えを行う腕は把持点の重力方向下方から物品を支え上げる。

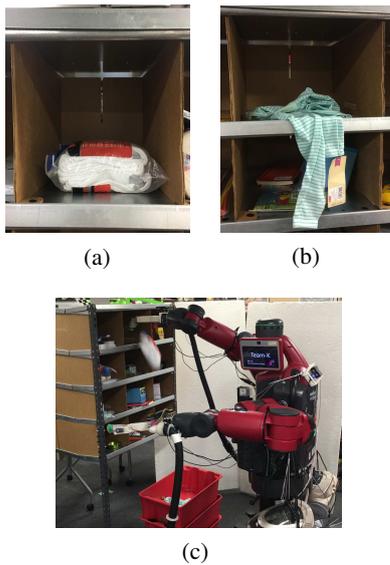


図 3: (a) 作業失敗なし (b) 棚からはみ出し (c) 落下

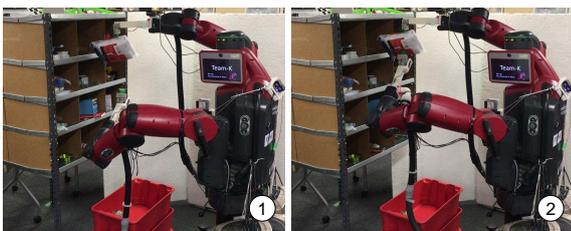


図 4: 双腕による手添え動作

4. CNN による作業失敗予測と動作選択

作業安定性を評価して手添え動作を行うべきかどうかを判断する。ここで区別のために、手添え動作を行う収納動作を双腕収納動作、行わない動作を単腕収納動作とする。作業安定性の指標として、本研究では各作業失敗に対して減点を設定することで計算できる作業得点を用いる。物品の把持状態を観測した画像から各作業失敗の発生確率を予測することで、各収納動作を行った場合に獲得できる作業得点の期待値を計算し、これを評価して動作選択を行う。作業失敗予測は画像を入力とする CNN を使い、実際に各収納動作を実行して収集した学習データセットで学習を行う。

4.1 作業失敗予測

学習データセット: 作業失敗予測を行う CNN の学習のためのデータセットを作成する。単腕・双腕の各収納動作を用いて収納作業を実際に行い、その際の把持物品の RGB 画像と作業結果をそれぞれ入力画像と出力ラベルとして収集する。把持物品の画像については、元画像 (Fig5(a)) から吸引ハンド手先の深度画像を基に把持物品とハンドの領域のみ自動的に抽出する (Fig5(b))。この処理の際にマスクされたピクセルの画素値は ImageNet [Krizhevsky 12] のデータセットの平均画素値に置き換える。出力ラベルについては、各収納動作を行った際に作業失敗が発生した場合には 1、発生しなければ 0 のラベルを与える。しかし、ある把持状況に対して単腕・双腕収納動作のどちらかしか実行することができないため、実行していない動作についてのラベルには -1 を与える。これによって出力ラベルとして長さ 6 のベクトルを収集する。

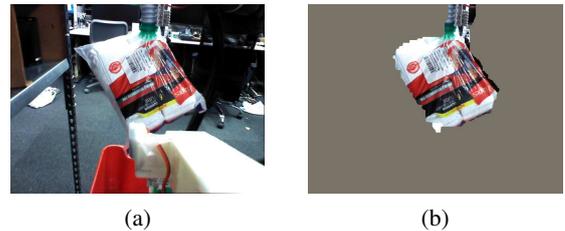


図 5: (a) 元画像 (b) 把持物品とハンドの領域のみ抽出した画像

ネットワーク構成: 作業失敗予測を行うネットワークは AlexNet [Krizhevsky 12] を基に設計し、詳しい構成については Fig6 に示す。出力層を除く畳み込み層と全結合層の活性化関数には Rectified Linear Unit 関数を用い、0 から 1 の連続した値を出力として得るために出力層には Sigmoid 関数を用いる。また、全てのプーリング層の手前でバッチ正規化 [Ioffe 15] を、出力層を除く全結合層でドロップアウト [Srivastava 14] を行う。

学習と損失関数: 事前学習として畳み込み層の重みは AlexNet [Krizhevsky 12] の学習済みモデルのものを用いる。バッチサイズ N_{batch} 、正解ラベル y 、ネットワーク出力 \hat{y} として、バッチ損失 L_{batch} は以下の損失関数から求める。

$$L_{batch} = -\frac{1}{N_{batch}} \sum_{i=1}^{N_{batch}} \sum_{i=1}^6 \delta(y_i) y_i \ln(\hat{y}_i) \quad (1)$$

このとき $\delta(y_i)$ は $y_i \in \{0, 1\}$ の場合は $\delta(y_i) = 1$ 、それ以外の場合は $\delta(y_i) = 0$ である。この損失関数によって長さ 6 のベクトルから、入力画像が収集された時に行われた収納動作に関する損失を計算することができる。 $N_{batch} = 1$ の場合には、入力画像が収集された時に行われた収納動作に関する損失のみが逆伝播されることとなり、1つのネットワークから両動作に関する作業失敗予測を同時に行うことができる。

4.2 動作選択アルゴリズム

各作業失敗に対して減点を設定することで各収納作業を行った場合の作業得点を計算し、これを作業安定性の指標として評価する動作選択アルゴリズムを設計する。この減点に重みをつけることで各作業失敗をどれだけ回避したいかを表現することができる。本研究では物品の落下を最も回避すべき作業失敗として他よりも重い減点を設定する。作業失敗の発生確率 $P_{failure}$ から計算できる作業得点の期待値 $E(score)$ は以下の

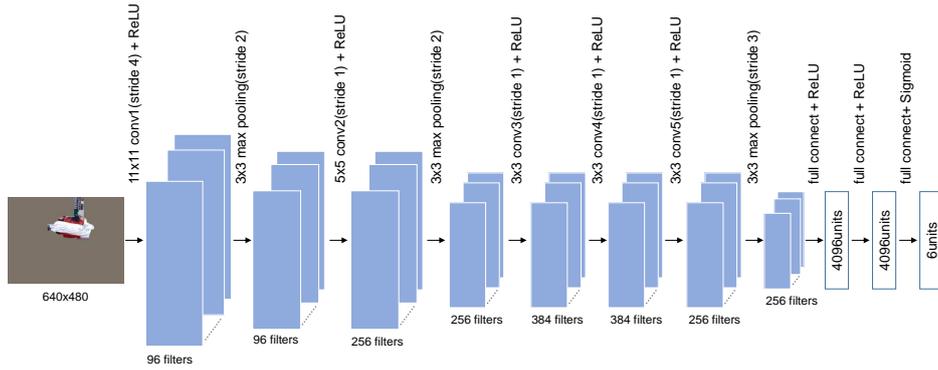


図 6: 作業失敗予測を行う CNN の構成図

ように計算する。

$$E(score) = \max(0, 10 - 10P_{damage} - 5P_{protrusion} - 5P_{damage}) \quad (2)$$

得られた作業得点の期待値を用いて Algorithm1 に示す動作選択アルゴリズムを設計する。原則として双腕収納動作を行った場合の作業得点の期待値 $E(score)_{dualarm}$ が単腕の場合 $E(score)_{singlearm}$ よりも十分に高い場合にのみ双腕収納動作を実行する。

Algorithm 1 Stowing motion select

$E(score)_{singlearm}$: Expected score of single-arm stowing
 $E(score)_{dualarm}$: Expected score of dual-arm stowing
 $threshold$: Threshold for $E(score)$ comparison

```

if  $E(score)_{dualarm} - E(score)_{singlearm} > threshold$  then
    Execute Dual-arm Stowing Motion
else if  $E(score)_{singlearm} < 5$  then
    Execute Dual-arm Stowing Motion
else
    Execute Single-arm Stowing Motion
end if

```

5. 検証実験

検証実験では 2016 年の APC[Hernandez 16] で用いられた物品の中から Fig7 に示す作業失敗の発生しやすい 3 物品を取り扱う。いずれも柔軟物体であり垂れ下がりによる棚との接触が発生しうる。

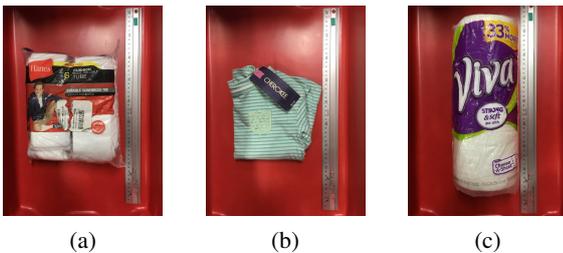


図 7: (a) 靴下 (b) 子供用 Tシャツ (c) ペーパータオル

5.1 データセット

CNN を用いて作業失敗予測を行うために単腕・双腕の両収納動作を計 304 回行いデータセットを作成した。データセットの内訳については表 1 に示す。

表 1: Stowing Task Dataset Statistics

Stowing Motion	Success	Drop	Protrusion	Damage	Total
Single-arm	74	48	28	1	151
Dual-arm	121	11	21	2	153

5.2 作業失敗予測

収集したデータセットの 8 割を学習用、2 割を検証用に分割して CNN の学習を行った。データ数が少ないため、10 回ランダムにデータセットを分割して学習を行った。予測精度については、ネットワークの出力について 0.5 の閾値を設定し、その値よりも高ければその出力が示す作業失敗は発生する、低ければ発生しないと予測しているとみなして計算を行った。30 回の平均精度は 72.46% となり、データセットの正解ラベル平均による平均精度の 64.02% を上回った。検証結果の例については Fig8 に示す。

5.3 双腕手添え動作の選択的実行

最終実験として学習したネットワークモデルを用いて選択的に手添え動作を行う収納作業システムを構成し、各物品に対して 10 回ずつ試行を行った。このとき、Algorithm1 の閾値 $threshold$ を 2.0 と設定して実験を行い、比較として単腕・双腕収納動作のみを行うシステムも同様の試行を行った。結果は表 2 に示す。検証実験では双腕収納動作は 30 回の試行において 9 回実行され、それ以外の試行では単腕収納動作が実行された。提案する動作選択システムによって双腕収納動作のみを行った場合と同等の高い作業安定性を実現しつつ、作業遂行時間の延長を抑制することが実現できた。

6. おわりに

本研究では、棚への収納作業において把持物品の状況に応じて選択的に双腕手添え動作を行うことを提案した。手添え

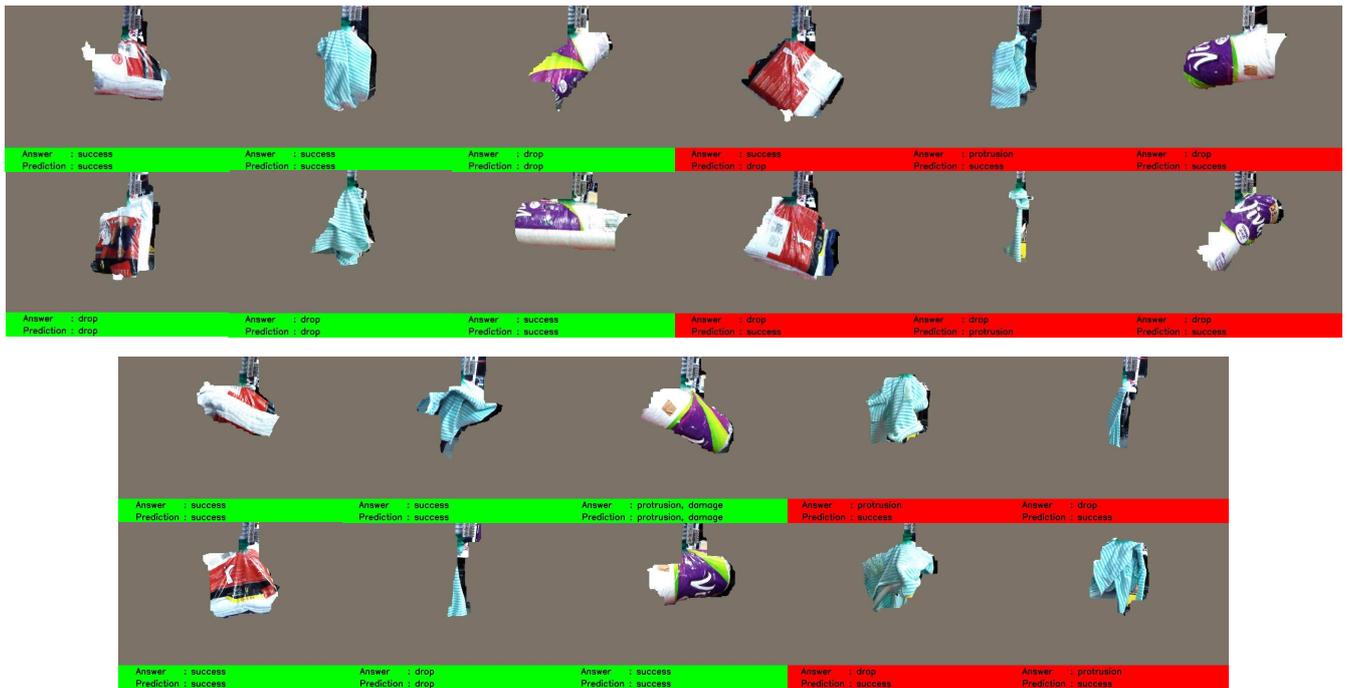


図 8: 単腕収納動作によって収集されたデータによる検証結果 (上) と 双腕収納動作によって収集されたデータによる検証結果 (下) 緑ラベルのものが正しく 作業失敗予測できており、赤ラベルのものは失敗している。

表 2: Evaluation of our method

Stowing System	Ave. Score	Ave. Time (s)
Single-arm Stowing System	6.0	52
Dual-arm Stowing System	8.8	67
Our Method	8.8	59

動作を行うかどうかの選択において、把持物品の RGB 画像を入力とした CNN を用いて作業失敗の発生確率を予測し、予測確率から作業安定性を評価することを提案した。検証として、提案した CNN が 72.46% の精度で作業失敗の予測ができることを示し、提案する動作選択を用いた収納作業システムが作業安定性と作業遂行時間の観点から優れていることを示した。

ロボット実機を用いた学習においてデータセット作成が時間かつ人的リソースが多く必要であり課題となっている。本研究の今後の展望としては自己監督的な学習を棚への収納作業システムに組み込むことを目指したい。

参考文献

[Correll 16] Correll, N., Bekris, K. E., Berenson, D., Brock, O., Causo, A., Hauser, K., Okada, K., Rodriguez, A., Romano, J. M., and Wurman, P. R.: Lessons from the Amazon Picking Challenge, *CoRR*, Vol. abs/1601.05484, (2016)

[Edsinger 07] Edsinger, A. and Kemp, C. C.: Two arms are better than one: A behavior based control system for assistive bimanual manipulation, in *Recent progress in robotics: Viable robotic service to human*, pp. 345–355, Springer (2007)

[Harada 12] Harada, K., Foissotte, T., Tsuji, T., Nagata, K., Yamanobe, N., Nakamura, A., and Kawai, Y.: Pick and place plan-

ning for dual-arm manipulators, in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2281–2286 IEEE (2012)

[Hernandez 16] Hernandez, C., Bharatheesha, M., Ko, W., Gaiser, H., Tan, J., Deurzen, van K., Vries, de M., Mil, B. V., Egmond, van J., Burger, R., Morariu, M., Ju, J., Germann, X., Ensing, R., Frankenhuyzen, van J., and Wisse, M.: Team Delft’s Robot Winner of the Amazon Picking Challenge 2016, *CoRR*, Vol. abs/1610.05514, (2016)

[Ioffe 15] Ioffe, S. and Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, *CoRR*, Vol. abs/1502.03167, (2015)

[Krizhevsky 12] Krizhevsky, A., Sutskever, I., and Hinton, G. E.: Imagenet classification with deep convolutional neural networks, in *Advances in neural information processing systems*, pp. 1097–1105 (2012)

[Levine 16] Levine, S., Pastor, P., Krizhevsky, A., and Quillen, D.: Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection, *CoRR*, Vol. abs/1603.02199, (2016)

[Pinto 16] Pinto, L. and Gupta, A.: Supersizing self-supervision: Learning to grasp from 50K tries and 700 robot hours, in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3406–3413 (2016)

[Srivastava 14] Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting., *Journal of Machine Learning Research*, Vol. 15, No. 1, pp. 1929–1958 (2014)