論文 / 著書情報
Article / Book Information

題目(和文)	家庭環境に向けた人間行動観察学習システムのためのトップダウン知識に基づく把持・操作スキルライブラリの設計
Title(English)	
著者(和文)	齊藤大智
Author(English)	Daichi Saito
出典(和文)	学位:博士(工学), 学位授与機関:東京科学大学, 報告番号:甲第327号, 授与年月日:2025年3月26日, 学位の種別:課程博士, 審査員:小池 英樹,篠田 浩一,三宅 美博,金﨑 朝子,井上 中順
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Institute of Science Tokyo, Report number:甲第327号, Conferred date:2025/3/26, Degree Type:Course doctor, Examiner:,,,,
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

博士論文

家庭環境に向けた
人間行動観察学習システムのための
トップダウン知識に基づく
把持・操作スキルライブラリの設計

齊藤 大智

東京科学大学
情報理工学院
情報工学コース

指導教員　小池 英樹

2025 年 1 月

概要

本論文では，手腕を巧みに用いた動作を含む家庭内作業が可能な家庭用ロボットに向けた人間行動観察学習 (Learning-from-Observation) のためのスキルライブラリの設計を目的としている．手腕を用いた動作の中でも，特に作業の遂行において必要不可欠な Grasp, In-hand manipulation，Compliant manipulation に焦点を当ててスキルを設計した．家庭内作業を行うためには，家庭環境において適した動作プリミティブの選択が可能であり，なおかつ非構造化環境において再利用可能なプリミティブスキルから構成されるようなスキルライブラリを設計することが重要である．しかしながら，これまでの研究ではこれらの要素を満たす設計ができていなかった．

そこで，本論文ではトップダウン知識に基づいてスキルライブラリを設計することを提案する．動作の分析や物体間の接触状態に対する数式的な解析によって，Grasp, In-hand manipulation, Compliant manipulation が必要な動作は，動作の背後にある目的や動作間に存在する共通部分といったトップダウン知識を基にいくつかの動作プリミティブに分類できることが知られている．本論文では，これまでの研究で得られたトップダウン知識を活用して家庭内作業の再分類を行い，動作プリミティブを導出する．そして，動作プリミティブごとに学習されたスキルからスキルライブラリを構成する．スキルは深層強化学習によって獲得させる．その際，プリミティブ内の動作で共通した要素に基づいた設計により，プリミティブが含む多様な動作に対しての再利用可能性を実現する．

提案したスキルライブラリとタスクプランナの一種である Learning-from-Observation の枠組みを組み合わせることにより，後続スキルを意識して適切なプリミティブを選択しなければ遂行できないような作業が実際できることを示した．また，シミュレーションと実機を用いた実験により，学習したスキルに再利用可能性があることを示した．

トップダウン知識に基づいてスキルライブラリを設計することによる利点は，プリミティブ内で共通した特徴に基づいてスキルが設計できるためにプリミティブ内の動作に汎用的なスキルが学習できる可能性がある点である．家庭内作業に含まれる動作は有限個のプリミティブに分類できる．そのため，トップダウン知識に基づいて各プリミティブに対してスキルの学習を行うことで，有限個のスキルで無数にある動作を実現できる可能性がある．また，上述した再利用可能性によって，スキルを一つのモジュールとしてシステムに組み込むことが可能になる．再利用可能なモジュールが複数ある場合，それらの組み合わせで記述できる様々な作業を実行できるようになる．そのため，対象とする作業が変わったとしてもスキルを再学習することなく，未知の作業でも実行ができるという利点がある．本論文は，そのような汎用的なスキルライブラリの実現への第一歩となる．

目次

図目次

表目次

第 1 章

序論

1.1 研究背景

世界中での高齢化に伴い，家庭内での介助が必要となる状況が増加している．しかし，共働き世帯の増加や核家族化の進行により，必ずしも家庭内介助が可能であるとは限らない．この問題を解決するために，家庭内作業を代行できるような家庭用ロボットの導入が望まれている．家庭内作業に含まれるような日常生活動作には物体の把持や操作といった手腕を用いた巧みな動作が多く出現することが示されている [29]．さらに，そのような動作を含む作業の代行を高齢者が家庭用ロボットに望んでいることが分かっている [30]．そのため，特に手腕を用いた巧みな動作を含む家庭内作業が可能な家庭用ロボットの導入が望まれる．家庭用ロボットでは，産業用ロボットのように一つの作業に特化させた一台を設置できるわけではなく，一台のロボットで多様な作業が遂行できることが求められる．そのため，一台のロボットに手腕を用いた巧みな動作を含む家庭内作業が可能なスキルを持たせることが重要である．

家庭内作業では様々な種類の複雑な動作を要する作業が頻繁に出現する．例えば，”冷蔵庫に入った箱の中身をコップに入れて，そのコップをカゴの中に置く”という，頻出でなおかつ類似の作業が多いものでさえ，冷蔵庫の取手を掴む，冷蔵庫を開く，冷蔵庫の中の箱を掴む，箱を持ち上げて取り出す，箱を指先で回転させる，箱を傾ける，箱を置く，コップを掴む，コップを持ち上げる，コップをカゴの中に置く，というように作業の達成までに多くの複雑な動作を行う必要がある (図 1.1)．そのため，手作業によるロボットプログラミングによってこのような作業をロボットに行わせることは困難である．

家庭内作業を行うスキルを自動で獲得させるために，強化学習 [1, 13, 22] や模倣学習 [25, 26] といった機械学習手法を用いて単一のニューラルネットワークにスキルを獲得させるという手法が注目されている．スキルとはロボットが対象動作を行うために必要な制御器のことを指す．このような手法によって，複雑な動作を伴う作業に対するスキルや，家庭環境のような非構造化環境に伴う不確かさに頑健なスキルが自動で獲得できるという利点がある．一方で，多様な作業に対する汎用的な報酬設計の困難さや学習分布外への汎化性能の低さ [31] によって，単一のニューラルネットワークへのスキル獲得では様々な動作の組み合わせによる多様な作業に対して汎用性が欠如してしまうという問題点があった．

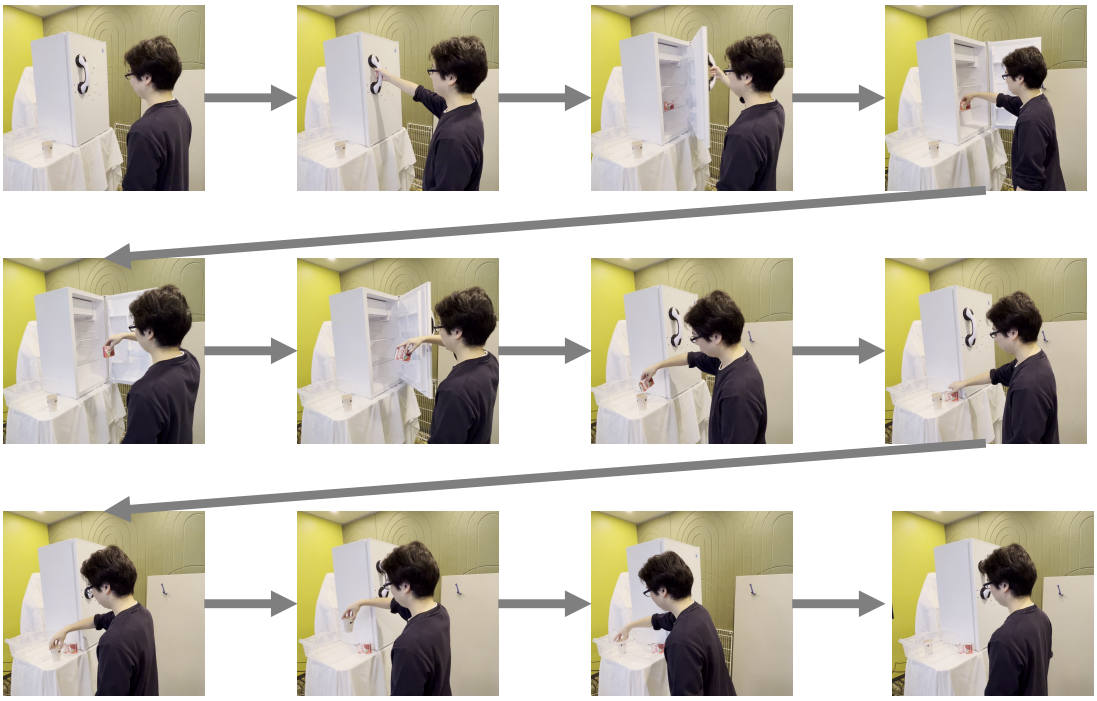


図 1.1　冷蔵庫に入った箱の中身をコップに入れて，そのコップをカゴの中に置くという家庭内作業の様子．何もしていない状態から始まり，冷蔵庫の取手を掴む，冷蔵庫を開く，冷蔵庫の中の箱を掴む，箱を持ち上げて取り出す，箱を指先で回転させる，箱を傾ける，箱を置く，コップを掴む，コップを持ち上げる，コップをカゴの中に置く，何もしていない状態に戻るまでの様子を示している．

このような問題を解決する方法として，作業全体を適当な動作単位に分割し，各動作単位に対して再利用可能なスキルを学習させ，学習したスキルを組み合わせることで多様な作業に対して汎用的な実行を可能にする手法がある．この手法では，複数の再利用可能なスキルを事前に定義しておいて，実行時にはタスクプランナを用いて適切なスキル実行順序を計画する．この複数個のスキルのまとまりはスキルライブラリと呼ばれ，スキル実行順序の計画はタスク計画と呼ばれる．タスクプランナとしては，古典的な PDDL [32] を用いる方法 [33, 34] や階層強化学習によって学習された方策 [35–37]，模倣学習によって学習された方策 [38]，大規模言語モデル [39–41] といった幅広いものが用いられる．例えば，Nasiriany らの研究 [37] では Metaworld [42]，Kitchen [43]，Robosuite [44] に含まれる作業を Grasp，Lift，Push，Twist という四つの動作単位に分割してスキルライブラリを定義し，階層強化学習によって学習された方策を用いてタスク計画を行なっている．このような手法には，スキルの組み合わせで表現可能な多様な作業に汎用的な実行を可能にするという利点がある．一方で，対象作業に応じた場当たり的なスキル列挙によって準備していないスキルが必要な場合に破綻してしまうという問題点がある．

こういった場当たり的なスキル列挙では把持や操作の種類を無視している．本来，把持や操作にはこれまでの動作解析 [8, 27] で示されている通り様々な種類がある．これらの種類は特定の作業におけるものではなく幅広い作業におけるものである．例えば，料理で瓶

を振る際の瓶に対する把持と掃除でモップ掛けする際のモップに対する把持は同一種類の指先と手のひらを物体に接触させるような把持である．また，料理でまな板を拭く時の操作と掃除でモップ掛けする時の操作は同一種類の単一面への接触を維持する操作となる．以上のように幅広い作業全体において共通した把持や操作の種類が存在する．このような動作の種類を無視した場合には，後続スキルに適したスキルが選択できない，各動作単位に対して別々のスキルを用意しなければならないといった問題が起きる．前者の問題に関しては，例えば，冷蔵庫を開けたい時に，ハンドルに引っ掛けるように把持しなければ扉を開けることが難しいが，ただ安定把持をすることだけを考慮して把持してしまった場合に指が滑って開けられないということが挙げられる．後者の問題に関しては，予測不能な数の動作が存在する家庭環境で実行させる際に重要な問題になる．そのような状況では，全ての動作に事前にスキルを用意することは不可能であるため，用意していないスキルが発生してしまう．

これらの問題を解決できる可能性がある分野として人間行動観察学習 (Learning-from-Observation, LfO) がある．LfO の枠組みでは，動作目的や物体への接触状態の変化の観察から網羅的に動作プリミティブを導出し，実行時には人間の実演から得られたプリミティブの組み合わせを参考に作業を遂行する．例えば，工場内での把持 [28, 45] や，棒の持ち替え [46, 47]，多面体の組み立て [3, 48] 等に対するプリミティブの導出が行われてきた．このように LfO ではこれまでは工場環境を想定して網羅的にプリミティブを導出してきた．このプリミティブの導出を家庭内作業に拡張できれば様々な複雑で多段階的な家庭内作業を獲得できるようになる可能性がある．

工場環境と家庭環境では当然ながら環境設定に差がある．そのため，これまで工場環境で発展してきた LfO の枠組みを家庭環境に拡張するためには，その差に由来する問題を解く必要がある．この問題の中で，特に重要な問題として以下の二点の問題が挙げられる．

1. 家庭内作業に適した動作プリミティブの選択可能性の欠如
2. 固定化されていない環境や作業のための多様な動作

前者は，家庭環境では工場環境と比較して動作の種類がより多様化していること [9]，道具を効率良く扱うために In-hand manipulation の重要性が増加すること [49] に由来する．後者は，固定化されていない環境における物体の位置姿勢や形状への不確かさに対処するために適応的な動作が求められること，抽象化されたプリミティブに含まれる様々な物体操作に汎用的な動作が求められることに由来する．これらの問題を解くためには，家庭内作業に向けたプリミティブの再考察をし，プリミティブ内であり得る動作へ再利用可能なスキル設計が必要となる．

1.2 研究目的

本論文では，以下の二点を満たすような家庭内作業向け LfO のスキルライブラリの設計を目的とする．

1. 家庭環境において適したプリミティブの選択が可能

2. 非構造化環境において再利用可能なプリミティブスキル

家庭内作業の中でも特に手腕を用いた巧みな動作を含む家庭内作業が可能なスキルライブラリの設計を対象とする．手腕を用いる動作の中には，片手のみでの動作や両手を用いた動作等があるが，本論文は片手のみでの作業に着目する．これは片手のみの作業には基礎的な動作が多く含まれており，日常生活における多くの作業が片手のみで完結できるからである．特に，作業の遂行において必要不可欠な Grasp，In-hand manipulation，Compliant manipulation に焦点を当ててスキルライブラリを設計する．Grasp とは物体を手で掴む動作のことである．これは多くの操作の前に必要となる基本的な動作であり，作業の実行において必要不可欠な能力である．In-hand manipulation とは指先のみを用いて手の中で物体の姿勢を変化させる操作である．物体操作の成功のためには，物体操作前に物体の姿勢を変化させて適切な把持を実現しておくことが必要であるため，In-hand manipulation は作業の実行に欠かせない．Compliant manipulation とは環境に拘束された物体の操作に必要な能力 [50] である．この操作は物理的に拘束された物体が多く存在する家庭内では特に多用される．例えば，ドアや引き出しのような拘束物体を安全に操作する場合に必要である．片手のみを用いる動作としては他にロボットの足と腕を協調させながら行う動作等も挙げられるが，本論文では手腕のみが動くことを前提に，作業の遂行において重要であり日常生活で頻出で基礎的な動作である Grasp，In-hand manipulation，Compliant manipulation に着目する．これらの三つの動作に対するスキルライブラリが設計できれば，既存のタスクプランナと組み合わせることで家庭内作業の多くが遂行できるようになる．

実は，動作の分析や物体間の接触状態に対する数式的な解析によって，Grasp, In-hand manipulation, Compliant manipulation はそれぞれいくつかの動作プリミティブに分類できることが示されている．Grasp であれば，把持後の物体操作において必要である把持力や操作性の有無から把持を分類したものが存在する [28, 51]．In-hand manipulation であれば，指先動作が接触状態の遷移によって分類できることが数学的な考察を基にして示されている [17, 46, 47]．Compliant manipulation であれば，数式的に導かれた物体と環境の間に存在する拘束によって物体操作が分類できることが示されている [27]．これらの分類は，動作の分析や数式的な解析によって得られた対象動作に関する知識であるトップダウン知識を基に導出されている．トップダウン知識としては具体的には動作の背後にある目的や動作間に存在する共通部分が挙げられる．動作の背後にある目的というのはその動作の後に実行されるスキルの達成のことである．これらのトップダウン知識に基づいて家庭内作業における動作を再分類することができる可能性がある．また，プリミティブが含む動作は共通の特徴を持つため，共通の特徴に着目した学習設計によって再利用可能なスキルの実現が期待される．

そこで，本論文ではトップダウン知識に基づいてスキルライブラリを設計することを提案する (図 1.2)．まず，家庭内作業における動作をトップダウン知識に基づいてプリミティブに分類し，それらのプリミティブからスキルライブラリを構成する．次に，動作プリミティブごとに共通の特徴に基づいてスキルを学習させる．学習時には，多様な動作に頑健なスキルを獲得させるために，深層強化学習によって学習を行う．

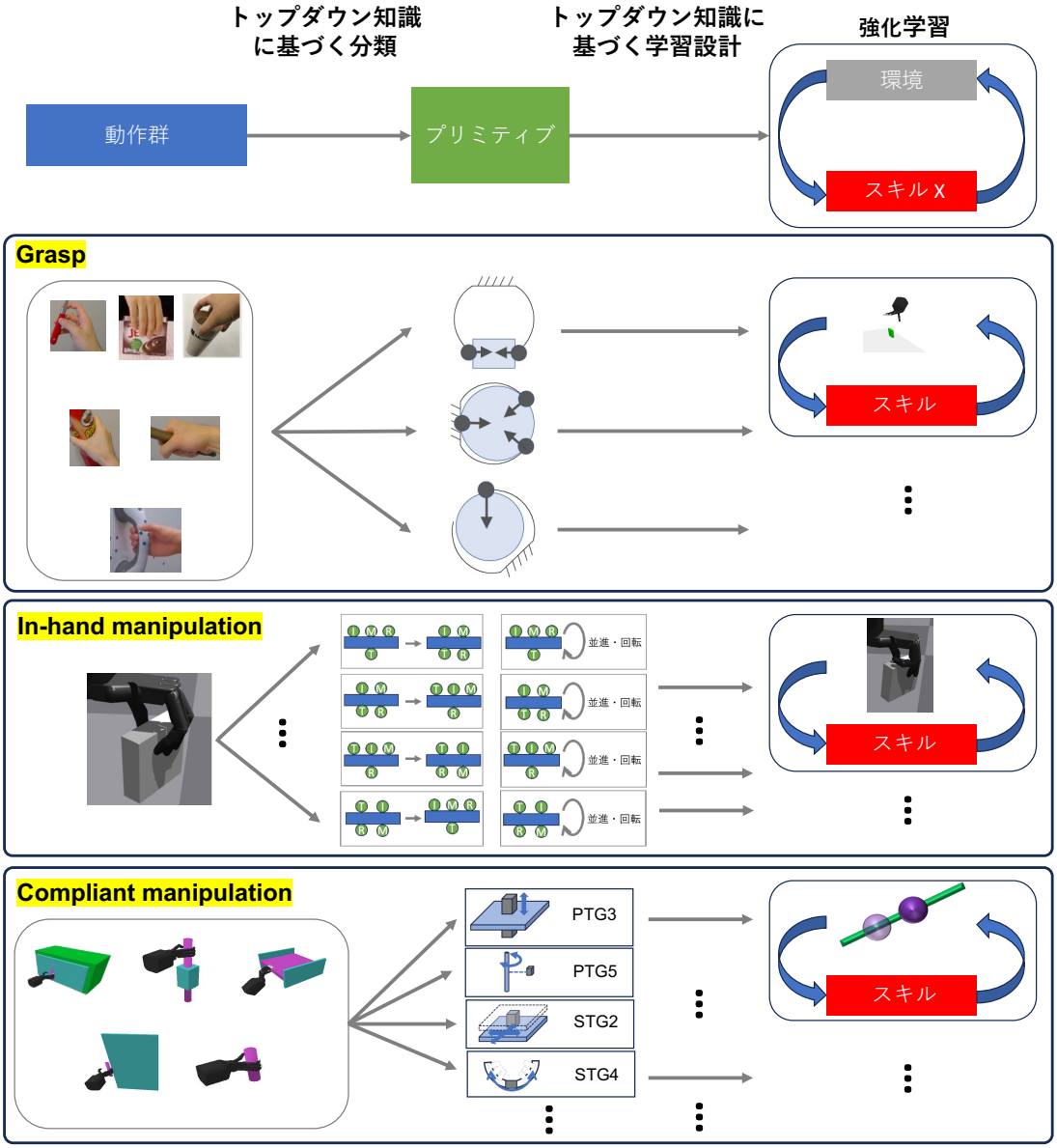


図 1.2　本論文の提案の概念図．家庭内作業群をトップダウン知識に基づいて複数の動作プリミティブに分割し，さらにトップダウン知識に基づいてスキルの学習設計を行う．

1.3　本論文の構成

本論文は全 8 章から構成される．以下に本論文の構成をまとめる．

本章，第 1 章では，家庭内作業の実行において重要なタスク計画におけるスキルライブラリの問題点について指摘し，スキルライブラリに求められる要素について議論を行った．

第 2 章では，これまでに行われてきたスキル設計や学習に関しての問題点を先行研究に触れながら明らかにする．特に，本論文では Grasp, In-hand manipulation, Compliant manipulation を対象にして，その問題点を整理する．

第 3 章では，明らかとなった問題点に対する解決策として，トップダウン知識に基づくスキルライブラリの設計を提案する．また，本論文が提案する手法の概要とその貢献を示す．

第 4 章では，Grasp に関するスキルライブラリの設計に関する提案と実装を行う．そして，シミュレーションと実機を用いたスキルの性能の検証や，獲得したライブラリによって可能になった作業を示す．

第 5 章では，In-hand manipulation に関するスキルライブラリの設計に関する提案と実装を行う．そして，シミュレーションを用いたスキルの性能の検証を行う．

第 6 章では，Compliant manipulation に関するスキルライブラリの設計に関する提案と実装を行う．そして，シミュレーションと実機を用いたスキルの性能の検証を行う．

第 7 章では，前章までの内容から，本論文が提案した手法の利点や現状の問題点，今後の方向性に関して述べる．

第 8 章では，各章で得られた結果や議論から本論文をまとめる．

1.4 研究成果の引用

本論文には国内研究会や国際会議，国際論文誌にて筆者が発表した以下の文献の一部が含まれる．

1. Saito, D., Sasabuchi, K., Wake, N., Takamatsu, J., Koike, H., Ikeuchi, K. (2022, November). Task-grasping from a demonstrated human strategy. In 2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids) (pp. 880-887). IEEE. (第 4 章)
2. Saito, D., Kanehira, A, Sasabuchi, K., Wake, N., Takamatsu, J., Koike, H., Ikeuchi, K. (2024, November). APriCoT: Action Primitives based on Contact-state Transition for In-Hand Tool Manipulation. In 2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids). IEEE. (第 5 章)
3. Saito, D., Sasabuchi, K., Wake, N., Kanehira, A., Takamatsu, J., Koike, H., Ikeuchi, K. Constraint-Aware Policy for Compliant Manipulation. Robotics 2024, 13, 8. (第 6 章)

第 2 章

関連研究

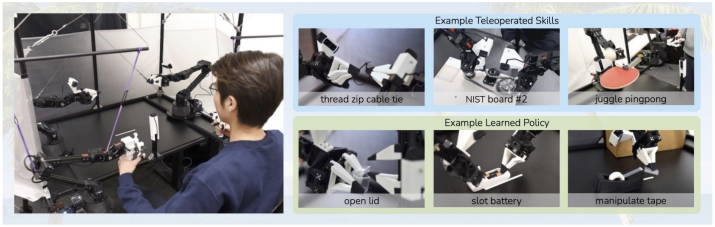
本章では，まず家庭環境のような非構造化環境におけるスキル自動獲得手法に関して概観する．次に, 本論文が対象とする Grasp, In-hand manipulation, Compliant manipulation のスキル設計や学習に関しての先行研究をまとめる．そして，家庭内作業に向けた LfO のスキルライブラリ設計という観点からそれぞれの問題点を明らかにする．

2.1 非構造化環境におけるスキル獲得の概観

2.1.1 強化学習や模倣学習による単一スキルの学習

家庭内作業では様々な種類の複雑な動作を要する作業が頻繁に出現する．そのため，手作業によるロボットプログラミングによってこのような作業をロボットに行わせることは困難である．家庭内作業を行うスキルを自動で獲得させるために，強化学習や模倣学習といった機械学習手法を用いて単一のニューラルネットワークにスキルを獲得させるという手法が注目されている．なぜなら，複雑な動作を伴う作業に対するスキルや，家庭環境のような非構造化環境に伴う不確かさに頑健なスキルが自動で獲得できるという利点があるからである．強化学習では巧みな報酬設計により，これまでに Grasp [6, 52–58]，In-hand manipulation [14, 16, 59–63]，Compliant manipulation [22, 23, 57, 64–66] の分野でのスキル獲得に成功しており，さらにいくつかの研究では実環境での動作も確認されている．例えば，Rajeswaran らは強化学習により把持や物体操作を含む作業に対するスキル獲得を行なった (図 2.1-(A))．一方で，単一作業に特化した学習が行われていた．多様な作業に対して汎用的に使用できる報酬を設計することが難しいため，複数作業に対する学習を行うことは困難である．複数作業を強化学習によって獲得する分野としてマルチタスク強化学習があるが，同様に報酬設計の困難さや対象とする作業数に対するスケーラビリティの欠如が問題となっている [67]．模倣学習では人間の実演データを真似ることを目的に実演データで教師あり学習を行うことで，同様の分野でスキル獲得に成功している [68–73]．特に，action chunking transformer [2] の登場により fine-grained manipulation と呼ばれる細かな動作が必要な作業が可能になり (図 2.1-(B))，模倣学習でこのような複雑な作業を獲得することが注目されている [74, 75]．このような研究では，単一の作業のみを対象として学習を行なっていた．大規模データ学習により，多様な作業を一つのニューラルネット

(A)



(B)

図 2.1　強化学習や模倣学習を用いた単一ニューラルネットワークでのスキル学習の例．(A) は Rajeswaran らによる把持や物体操作を含む作業への強化学習 [1]，(B) は Zhao らによる action chunking transformer を用いた fine-grained manipulation の学習 [2].

ワークで学習させるという研究も進歩している [25, 26, 76]．しかしながら，学習データ分布内の動作には汎化できるものの分布外の動作に対しては汎化性能が低いという問題点が指摘されている [31]．以上のように，多様な動作の組み合わせによって作業が膨大に増えるような家庭内において，多様な作業を強化学習や模倣学習を用いて単一ニューラルネットワークで学習することは困難である．

2.1.2　スキルの組み合わせによる学習

作業全体を適当な動作単位に分割し，各動作単位に対して再利用可能なスキルを学習させ，学習したスキルを組み合わせることで多様な作業に対して汎用的な実行を可能にする手法が提案されている．このような手法では，事前に定義した複数個の再利用可能なスキルが，タスクプランナにより計画された実行順序に基づいて実行される．タスクプランナとしては，古典的な PDDL [32] を用いる方法 [33, 34] や階層強化学習によって学習された方策 [35–37, 77]，Diffusion によるタスク計画の生成 [78]，大規模言語モデル [39–41] といった幅広いものが用いられる．例えば，Brohan らの研究 [39] では，作業を Pick, Place, Open and Close drawers, Navigate 等の動作単位に分割してスキルライブラリを定義し，大規模言語モデルの PaLM [79] を用いてタスク計画を行った (図 2.2-(A))．また，Nasiriany らの研究 [37] では Metaworld [42]，Kitchen [43]，Robosuite [44] に含まれる作業を Grasp，Lift，Push，Twist という四つの動作単位に分割してスキルライブラリを定義し，階層強化学習によって学習された方策を用いてタスク計画を行なった (図 2.2-(B))．このような手法には，スキルの組み合わせで表現可能な多様な作業に汎用的な実行を可能にするという利点がある．一方で，対象作業に応じた場当たり的なスキル列挙によって準備していないスキルが必要な場合に破綻してしまうという問題点がある．

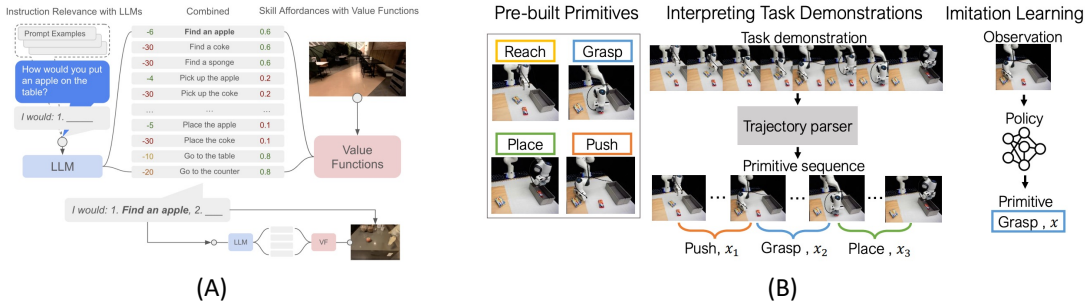


図 2.2　スキルの組み合わせによる学習の例．(A) は Brohan らによる LLM を用いたタスク計画 [1]，(B) は Nasiriany らによる階層強化学習を用いたタスク計画 [2].

2.1.3　人間行動観察学習 (Learning-from-Observation)

場当たり的なスキル列挙という問題を解決できる可能性がある分野として人間行動観察学習 (Learning-from-Observation, LfO) [3] がある．LfO の枠組みでは，動作目的や物体への接触状態の変化の観察から網羅的に動作プリミティブを導出し，実行時には人間の実演から得られたプリミティブの組み合わせを参考に作業を遂行する．例えば，これまでに，工場内での把持 [28, 45]，多面体の組み立て [3, 48]，機械部品の組み立て [80]，ひも結び [81]，ダンスにおける全身運動 [82]，持ち替え動作 [46, 47] 等に対するプリミティブの導出が行われてきた．これまでの LfO 分野における研究の多くは工場環境を想定して網羅的にプリミティブを導出してきた．例えば，Ikeuchi らの研究 [3] では，物体の接触状態の遷移に基づいて多面体の組み立てにおけるプリミティブの考察をした (図 2.3-(A))．家庭内作業を対象とした研究も存在し，人間による家庭内作業の実演からスキルパラメータを抽出する研究 [4, 83–87] や，Compliant manipulation のプリミティブを網羅的に求める研究 [27] が行われてきた．例えば，Wake らの研究 [4] では，人間の動作や言語指示を用いた LfO の枠組みを新たに提案した (図 2.3-(B))．一方で，家庭内作業における Grasp や In-hand manipulation のプリミティブを網羅的に求めることはなされておらず，さらに，再利用可能なプリミティブスキルの設計方法に関しても議論されていない．

本論文では，家庭内作業における Grasp，In-hand manipulation のプリミティブを導出し，さらに，これらの動作以外にも Compliant manipulation の動作も含めたプリミティブスキルの再利用可能な設計を提案する．これによって構築されるスキルライブラリは，LfO を実際に家庭内作業に適用することを可能にするだけではなく，これまでの非構造化環境におけるスキル獲得の分野全体において問題であった多様な家庭内作業の遂行を実現する．

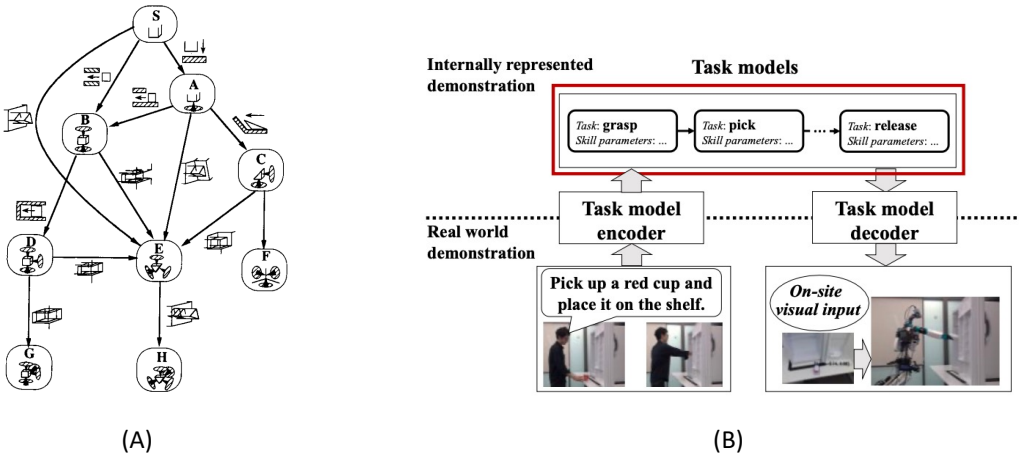
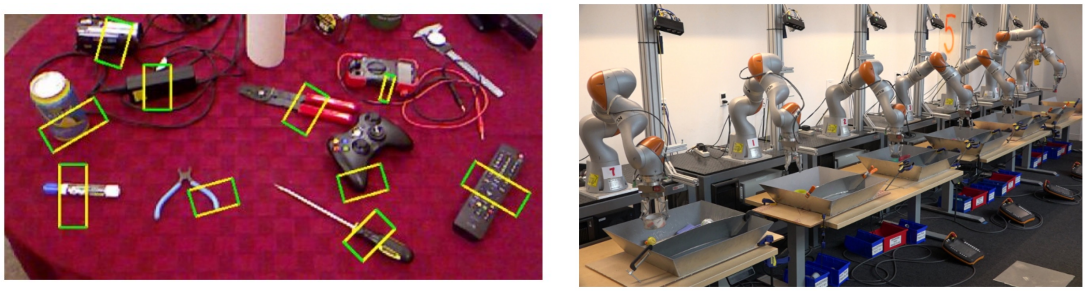


図 2.3　Learning-from-Observation 分野での研究例．(A) は Ikeuchi らによる多面体の組み立てにおけるプリミティブの考察 [3]，(B) は Wake らによる人間の動作や言語指示を用いた LfO の枠組みに関する研究 [4]．

2.2　Grasp

2.2.1　安定把持

Grasp に関する研究は数多く行われてきており [88]，初期の頃は主に物体を安定的に把持することに焦点が置かれていた．そのような研究では，安定把持を達成するための物体に対する拘束と，把持前の状態からその拘束に到達するまでの運動計画の二つを含む問題を解くようになった [89]．この問題を解くための手法の一つとして，物体のモデルを用いて把持計画を行う研究があった [90–93]．このような研究では，GraspIt! [94] のようなシミュレーションを用いて把持の安定性指標 [95] を評価することで指の配置を決定し，その配置を満たすための運動計画を行う．これらの手法では安定性指標の計算のために対象物体の詳細な三次元モデルを必要とする．そのため，家庭環境のような物体情報が未知の環境でこれらの手法を適用することは困難であった．この問題を解決するために，深層ニューラルネットワークを用いて画像から把持位置を推定する手法が提案された [5, 96–99]．例えば，Lenz らは，深層ニューラルネットワークを用いてロボット視点 RGBD 画像から把持位置を推定することで平行グリッパによる把持を可能にした (図 2.4-(A))．これらの手法ではフィードフォワード制御器を用いて把持動作を行っていたため，把持位置の推定誤差が大きくなると失敗するという問題があった．この問題を解決する手法として，触覚情報を参考にして把持位置を調整する方法 [100, 101] や，実際の把持動作データを用いた模倣学習 [102] や強化学習 [6, 52–58] によってフィードバック制御器を学習する手法がある．例えば，Kalashnikov らは実機で集めたデータを用いて強化学習を行うことで平行グリッパによる把持を学習した (図 2.4-(B))．以上の研究は，後続スキルのことは考慮しておらず，安定把持を行うことのみに焦点を置いて把持の計画や学習を行っていた．そのため，目的



(A) (B)

図 2.4　安定把持に焦点を当てた研究の例．(A) は Lenz らによる把持位置推定 [5]，(B) は Kalashnikov らによる安定把持の強化学習 [6]．

作業にとって不適切な把持を行うことによる作業の失敗を引き起こす可能性がある．

2.2.2　把持分類

安定把持の研究に並行して，与えられた作業に対して適切な把持を決定するために，把持を様々な観点で複数の型に分類する研究が行われてきた．人間の把持を物体の情報や動作目的に応じて分類したもの [7–9, 28, 103] や，ロボットの把持を安定性や操作性の観点から分類したものが存在する [51]．Cutkosky [7] や Kang ら [28] による分類では主に工場環境に焦点を当てて把持が分類された (図 2.5-(A))．その後，家庭環境に対しても把持分類がなされるようになった [8, 9, 103](図 2.5-(B,C))．そして，そのような分類から適切な把持を決定する手法も研究されてきた．例えば，与えられた作業や対象物体を用いて人間の把持分類の中から適した把持を決定する方法 [7] や，人間の実演から最適な把持を決定する方法 [28] が存在する．これらの研究では，把持分類は行われているものの，ロボットでどのように実行するかが不明であった．人間とロボットの手の構造の違いのために．人間の把持をそのままロボットで模倣することは難しい．そのため，人間の把持分類をそのままロボットに用いることはできない．

2.2.3　把持分類を活用したスキル実行

人間とロボットの手の構造の違いを解決するために，Kang らは人間の把持をロボットに対応付ける functional mapping を行った [45]．functional mapping では virtual finger [104] の理論に基づいて指の機能を特定し，その機能を担う指同士を対応付けることで人間とロボットの手の構造の違いを解決した．しかしながら，物体の位置姿勢や形状への不確かさに対して頑健ではないため，日常生活で用いることは困難である．人間の把持プリミティブごとに方策を強化学習で学習しておいて，実行時には与えられた作業に応じて方策を選択するという手法も存在する [10, 105](図 2.6)．この研究では物体の三次元モデルは不要ではあるが，主に指先のみを接触させて物体を把持する precision grasp や，指先と掌を接触させる power grasp の学習のみに焦点を当てられている (図 2.7-(A,B))．その

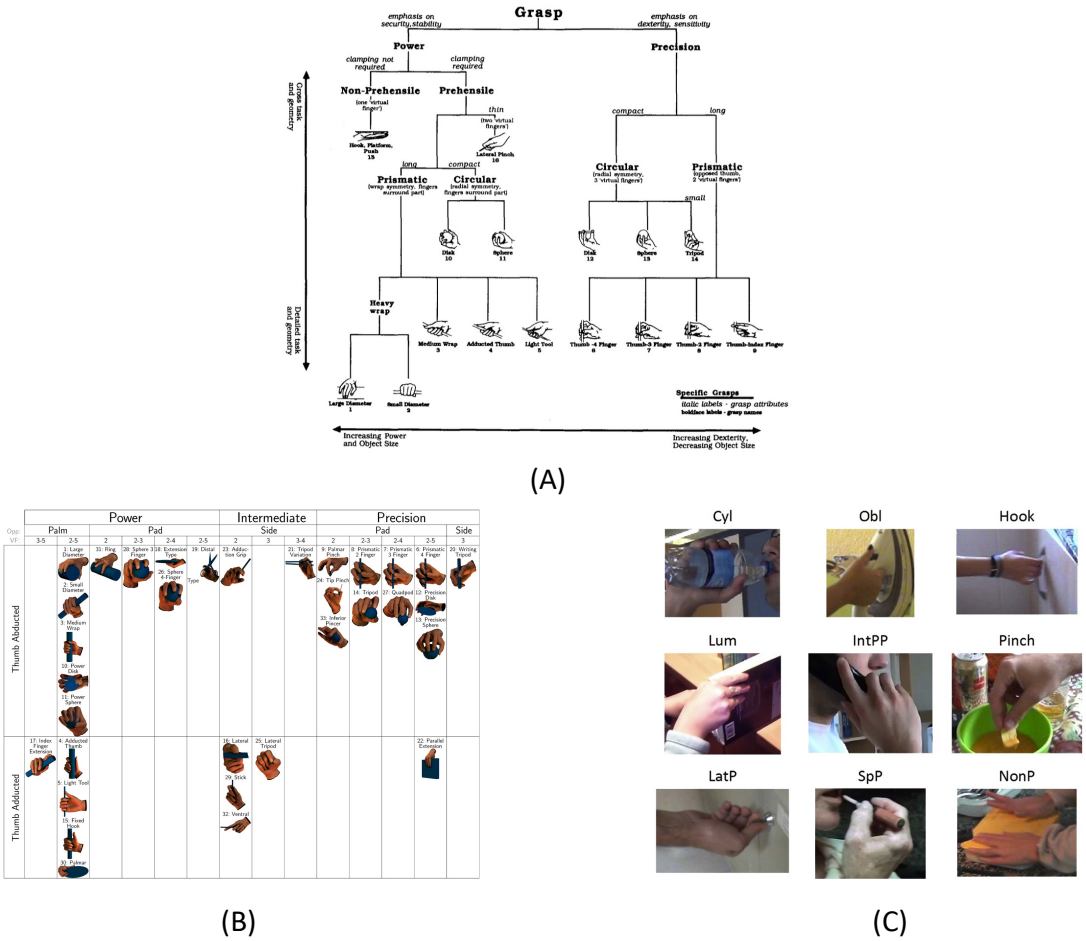


図 2.5　人間の把持分類に焦点を当てた研究の例．(A) が Cutkosky による分類 [7]，(B) が Feix らによる分類 [8]，(C) が Margarita らによる分類 [9].

ため，日常生活で多く出現する non-prehensile grasp (図 2.7-(C)) に関して議論されていない．これはドア開けや引き出し開けといった物体操作時に頻繁に使用される把持プリミティブである．また，人間の把持にはロボットにとって冗長なものが存在しており，安定性と操作性の観点 [51] から必要な方策はより削減できるはずである．例えば，4 本の指を用いて把持を行う Prismatic-3 finger や 5 本の指を用いて把持を行う Prismatic-4 finger は安定性や操作性的には同一とみなせる．一方で，[10, 105] ではそういった冗長性を無視して全ての把持に対して方策を設計しているために手間がかかる．さらに，把持を特徴づける接触点位置を無視した報酬設計のために目的の把持が学習される保証がない．

なお，把持プリミティブを意識することで対象作業に適した把持が可能になるため，これは一種のタスク指向把持と呼べる．しかしながら，タスク指向把持は一般的に作業に適した物体の把持位置を推定するような手法のことを指す [11, 12, 106–111]．これらの研究では，目的のタスクにおいて物体の機能を発揮できるような把持位置や接触分布を推定し，その推定結果を満たすように把持を行う．例えば，[11] では，タスクに適した把持位置を推定する (図 2.8-(A))．[12] では，物体の機能を発揮できるような接触分布を満たすように

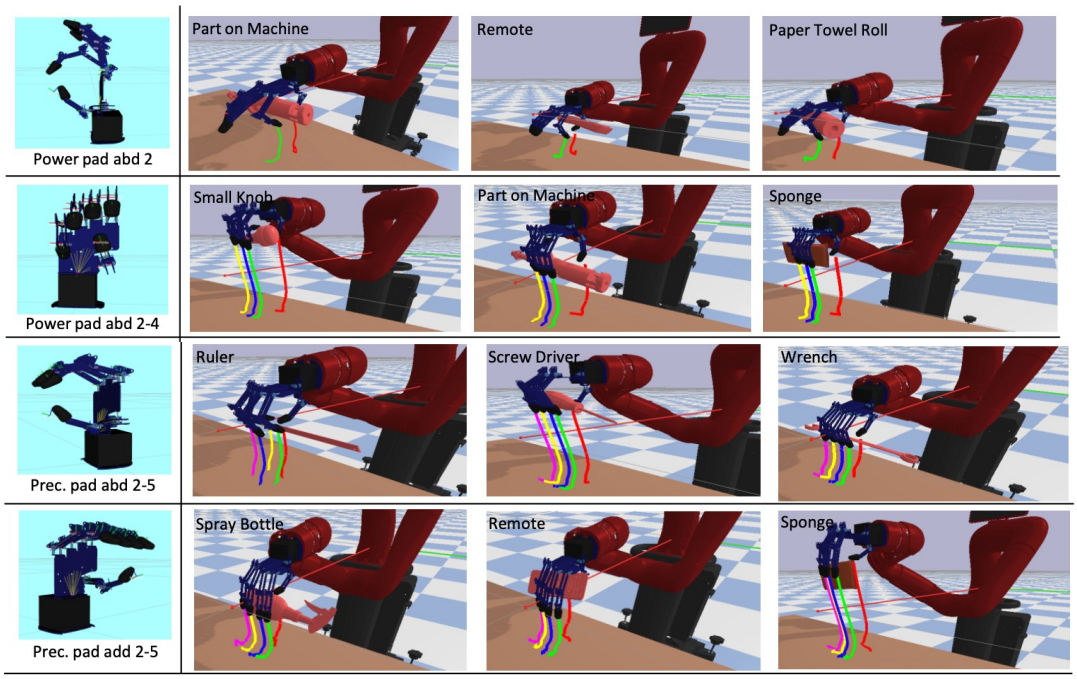


図 2.6　Li らによるタスクを考慮した把持 [10]．図は [10] 内の図を編集したものである．

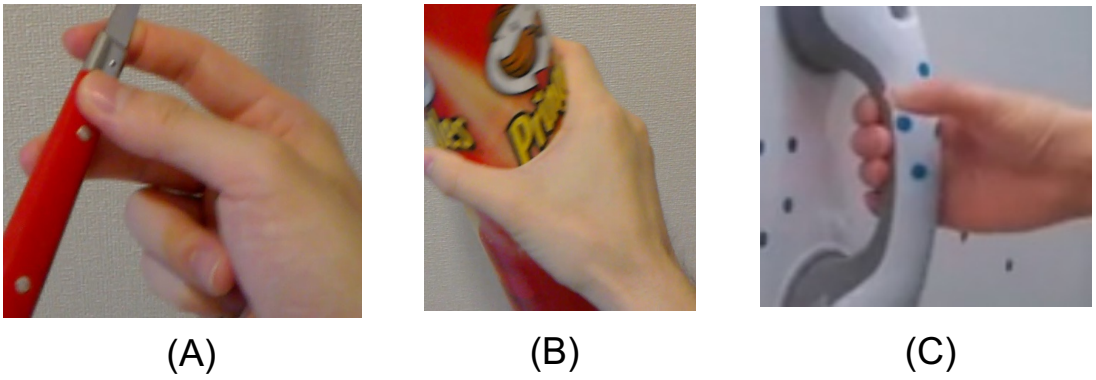


図 2.7　把持プリミティブの例．(A) が precision grasp，(B) が power grasp，(C) が non-prehensile grasp の一例．

把持を行うスキルを強化学習により獲得する (図 2.8-(B))．ここでは，物体のどこを把持するべきなのかに焦点を当てており，物体をどの把持プリミティブを選択するべきなのかに関しては無視されている．そのため，[8, 51] で議論されているような指先が持つ機能が満たされる保証はない．

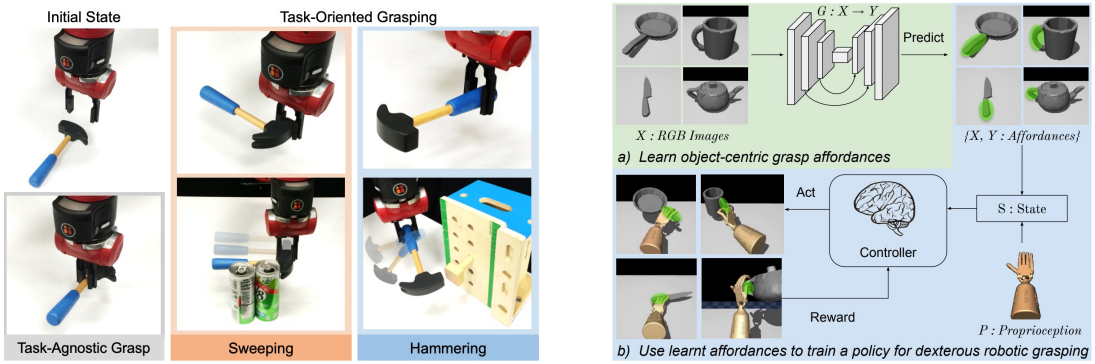


図 2.8　タスクに適した把持位置を推定する手法の例．(A) は Fang らによる把持位置推定後の実行例 [11]，(B) は Mandikal らによる接触分布を満たすような把持を行う枠組み [12] の図．

2.2.4　まとめ

以上の関連研究の特徴を表 2.1 に示す．この表は，家庭内作業を行うためのスキルライブラリに求められる要素である，家庭内作業において適したプリミティブの選択が可能，非構造化環境において再利用可能なスキルという観点から関連研究をまとめたものである．なお，非構造化環境において再利用可能というのは，物体の位置姿勢や形状への不確かさが発生する環境において適応的な動作が可能ということである．以上から，Grasp において上述したスキルライブラリに求められる要素の二点全てを満たす研究は未だに存在しない．

	家庭内作業において適したプリミティブの選択が可能	非構造化環境において再利用可能なスキル
[Lenz, et al., 2015] ほかの安定把持計画や学習 ([5, 6, 52–58, 90–93, 96–102])	×	×
[Cutkosky, et al., 1989] ほかの工場環境での人間の把持分類 ([7, 45])	×	×
[Margarita, et al., 2014] ほかの家庭環境での人間の把持分類 ([8, 9, 103])	○	×
[Li, et al., 2021] ほかの把持プリミティブスキルの学習 ([10, 105])	×	○

表 2.1　Grasp の関連研究のまとめ.

(A) (B)

図 2.9　In-hand manipulation の強化学習による実現の例．(A) は Andrychowicz らによるルービックキューの回転 [13]，(B) は Yang らによる任意の手の姿勢における物体の回転 [14] の図．

2.3　In-hand manipulation

2.3.1　物体姿勢変化のみの操作

In-hand manipulation は数十年間に渡って多くの研究がなされてきた [112]．初期の頃はロボットハンドや物体の精密なモデルを用いて動作計画をする研究があった [113–116]，しかしながら，実世界では不確かさの影響で，そのような精密なモデルに頼って制御することは難しい．そのため，実世界への適用は限られていた．

この問題を解決するために，高速に値を取得できるセンサを用いたフィードバック制御器を設計する手法 [117, 118] や，深層強化学習を用いて制御方策を学習する研究が増えている [13, 16, 59, 119–123]，例えば，手のひらの上に乗った様々な形状の物体の回転操作 [13, 119–127] や，指先だけで支えられた状態の物体に対しての回転操作 [14, 16, 59–63] がこれまでにできるようになった．例えば，Andrychowicz らは手のひら上でルービックキューブを回転させるスキルを強化学習により獲得させた (図 2.9-(A))．Yang らは指先のみで把持した状態の物体に対して任意の手の姿勢において物体を回転させるスキルを獲得させた (図 2.9-(B))．これらの研究では手の中での物体姿勢の変化のみに焦点が当てられているが，作業を行う上で重要な所望の把持プリミティブの実現が無視されていた．

In-hand manipulation を複数の動作に分割し，各動作に対して低レベル制御器を設計することを行った研究が存在する [128]．Li らは二次元空間上での in-hand manipulation を reposing，sliding，flipping という三つの動作プリミティブに分解した (図 2.11-(A))．一般に全体の動作を分割することで時空間的な探索量を削減して学習を容易できることが知られており [35, 36, 128–131]，この研究では分割されたそれぞれの動作に対して制御器の設計をすることで一連の動作の学習を容易にした．しかしながら，この研究では二次元空間の回転にのみ着目しており，さらに所望の把持プリミティブの実現に関しては議論されていない．

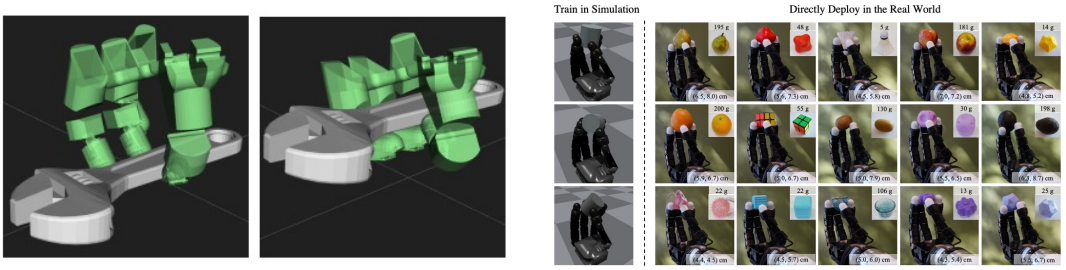


図 2.10　目的把持の実現を焦点に当てた強化学習の研究．(A) は Zarrin らによるレンチに対する把持の遷移 [15]，(B) は Qi らによるつまめる程度の物体に対する精密把持を維持したまま回転させる実行例 [16] の図．

2.3.2　目的把持の実現

物体姿勢の変化に加えて，所望の把持の達成まで含めた操作は in-hand tool manipulation と呼ばれる [15, 132]．このスキルを強化学習によって獲得させる研究も存在する [15, 132]．これらの手法ではレンチやドライバに対する把持の遷移を行なうスキルを強化学習により獲得させた (図 2.10-(A))．しかしながら，これらの手法ではロボットハンドや物体の精密なモデルが必要であった．そのため，実世界への応用が困難であった．Qi らの研究 [16, 59] では，指の関節角度に関して制約をかけることで，つまめる程度の小さな物体の回転後に精密把持が維持できることが確認されている (図 2.10-(B))．しかしながら，関節角度への制約によってこの手法で解けるのは時空間的な探索が狭くて済む操作のみであり，アスペクト比の高い直方体に対する操作のようなさらに広い空間の探索が必要な操作に関しては解くことができない．アスペクト比の高い形状であるペンの回転に関する研究もこれまでに多く行われてきた [63, 117, 118, 125–127] が，このような手法は物体姿勢の変化のみに着目している．

把持は接触状態によって表現できるため，in-hand manipulation で対象となる把持の変化は接触状態の遷移だとみなすのが自然である．この接触状態の遷移を detach, crossover, attach という三つの動作プリミティブとして定義し，これらのプリミティブを用いて物体操作の計画を行った研究が存在する [17, 46, 47]．工藤，Vinayaekhin らはタングルトポロジーを用いて棒状の物体に対する in-hand manipulation は三種類の動作で表現することを提案した (図 2.11-(B))．しかしながら，物体のモデルが既知であることが仮定されており，実世界への適用が限られる．

2.3.3　まとめ

以上の関連研究の特徴を表 2.2 に示す．この表は，家庭内作業を行うためのスキルライブラリに求められる要素である，家庭内作業において適したプリミティブの選択が可能，

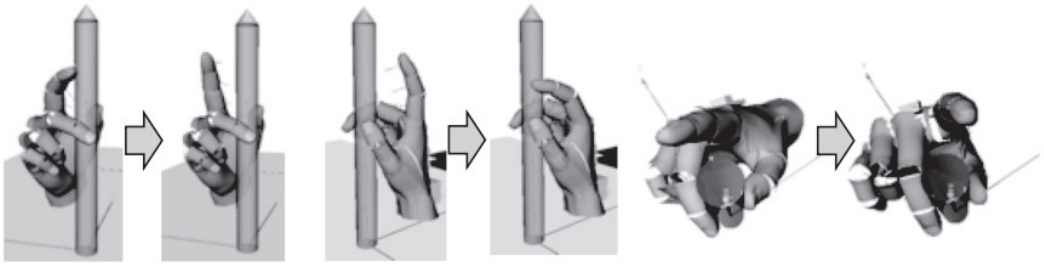


図 2.11　工藤，Vinayavekhin らによる棒状物体に対する in-hand manipulation のプリミティブへの分解 [17].

	家庭内作業において適したプリミティブの選択が可能	非構造化環境において再利用可能なスキル
[Li, et al., 1998] ほかの回転操作のみ意識した操作計画 ([113–116])	×	×
[Andrychowicz, et al., 2020] ほかの回転操作のみ意識したスキル学習 ([13, 63, 117–123, 125–128])	×	○
[Qi, et al., 2023] ほかのつまめる程度の物体に対するスキル学習 ([16, 59])	△	○
[Zarrin, et al., 2023] ほかの目的把持達成を意識したスキル学習 [15, 17, 46, 47, 132]	○	×

表 2.2　In-hand manipulation の関連研究のまとめ.

非構造化環境において再利用可能なスキルという観点から関連研究をまとめたものである．家庭内作業において適したプリミティブの選択が可能というのは，所望の把持を実現できるということである．非構造化環境において再利用可能なスキルというのは，物体の位置姿勢や形状への不確かさが発生する環境において適応的な動作が可能ということである．Qi らの研究 [16, 59] に関しては，比較的簡単な形状であるつまめる程度の大きさの物体に対して精密把持から精密把持への持ち替えにのみ成功できる．しかしながら，家庭内作業で登場するような道具の多くは横長の物体であり，このような物体に関しては扱えないことから△としている．以上から，in-hand manipulation において上述したスキルライブラリに求められる要素の二点全てを満たす研究は未だに存在しない.

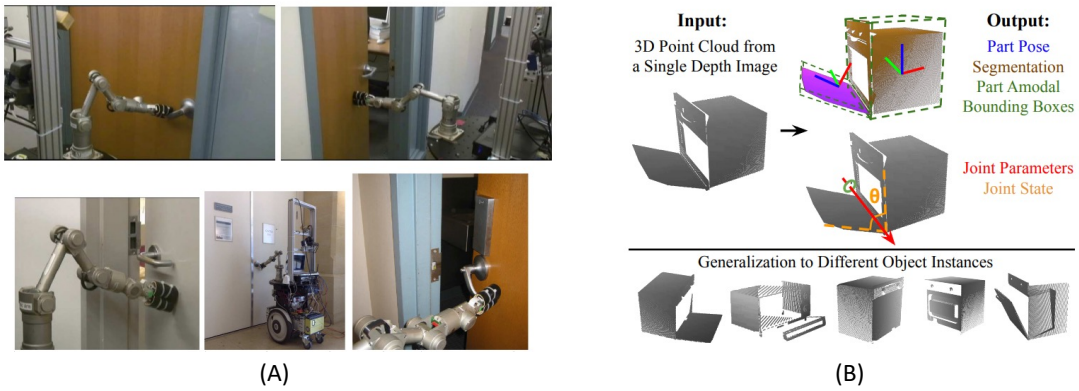


図 2.12　拘束物体のための物体の形状推定．(A) は Klingbeil らによるドアハンドルの位置姿勢推定を用いたドア開け [18]，(B) は Li らによる関節を持つ物体に対する形状推定 [19]．

2.4　Compliant manipulation

2.4.1　特定の操作に対するスキル設計

これまでの研究では，主に引き出しやドアを開けるための制御器の設計に焦点が当てられてきた．ドア開閉に関する先駆的な研究は [133] であり，そこでは既知のドアモデルに基づいてロボットの動作が計画されていた．非構造化環境のようなモデルが未知の環境を対象とした研究には，形状推定を用いた研究が存在する [18, 19, 134–140]．このような研究では，視覚入力からドアハンドルの位置や関節姿勢を推定し，この推定結果から運動軌道を計画していた．例えば，Klingbeil らは視覚入力からドアハンドルの三次元位置を推定し，この結果を用いてドア開けの運動計画を行なった (図 2.12-(A))．Li らは点群から関節を持つ物体の位置姿勢や関節のパラメータを推定した (図 2.12-(B))．この研究ではロボットでの実行は行なっていないが，この結果を用いて関節を持つ物体の操作を計画すること自体は可能である．しかしながら，推定精度は compliant manipulation を行うには不十分である．例えば，[138] のような手法では，実世界において回転軸の向きの推定誤差が約 20° 発生してしまう．このような推定精度の悪さが軌道計画に基づく手法の失敗の原因となっている．

上述した形状推定誤差や実世界におけるセンサ値の不確かさに対処するために，いくつかの研究では力・トルクセンサに基づくフィードバック制御器を提案している．[141] では，エンドエフェクタにかかる力が最小になるような軌道をたどるという単純な戦略に基づいて制御器が設計された．この戦略に基づいた制御器は [141] 以外にも多く用いられている [20, 21, 142, 143]．例えば，Schmid らは力・トルクに基づくフィードバック制御器を設計し，ドア開けに応用した (図 2.13-(A))．Jain らも同様に引き出しやドア開けにフィードバック制御器を応用した (図 2.13-(B))．．フィードバック制御器は人間による手

図 2.13　古典制御を用いた操作．(A) は Schmid らによる力・トルクを入力とした制御器による操作 [20]，(B) は Jain らによる引き出しや扉に対する操作 [21]．

作業の制御パラメータの調整が必要になり，この調整には専門的な知識が必要になる．これらの研究では力の大きさを用いて制御器を設計している．力の大きさは環境や対象操作によって変化するため，それらが変わるたびにパラメータの調整が必要となる．そのため，環境が固定化されていない実世界で用いるには手間がかかる．なお，このような手間を省くためにパラメータ調整を自動的に行う手法が提案されている [144–146]．この自動調整では，実世界でロボットが環境とインタラクションしてデータを集めることで調整を行う．Compliant manipulation が対象としている物体が環境に拘束されているような状況でデータ収集を行う場合，軌道誤差が大きい場合に大きな力がロボットと物体に直接的にかかる可能性がある．そのため，実世界のインタラクションを通して自動調整を行うことは危険が伴うため適していない．

強化学習では，適切に環境と報酬を設計すれば，自動的に制御方策を学習することが可能である．そのためフィードバック制御器に必要であった手作業による制御パラメータの調整を緩和できる．強化学習を応用して Compliant manipulation のための方策を学習した研究はいくつか存在する [22, 23, 57, 64–66]．例えば，Yahya らは実機を用いてドア開けデータを収集し，そのデータを用いて強化学習を行なった (図 2.14-(A))．これらの研究では，特定の操作のみに対応する環境と報酬を用意することでポリシーを設計している．例えば，これらの研究ではドアを開ける環境を用意し，ドアの角度を報酬として計算している．浦上らは様々なドア開けに汎化するための学習環境である DoorGym を提案している [23]．この学習済みポリシーは，様々な照明条件下でかつ様々なドアノブを持つドアに汎化可能 (図 2.14-(B)) であるが，ドアの開閉のみに着目している．そのため，学習された方策は学習対象の操作以外の操作には適用できない．

2.4.2　複数操作に対して汎用的なにスキル設計

Karayiannidis ら [24, 147] は，並進のみ可能な関節（Prismatic 関節）もしくは回転のみ可能な関節（Revolute 関節）を持つ物体の操作に対して汎用的な制御器を提案した (図 **??**)．上述したように，これらの研究では力の大きさを用いて制御器を設計しており，環境や対象操作が変わるたびにパラメータの調整が必要となる．

様々な操作に対して汎化した方策を学習することに焦点を当てた研究も存在す

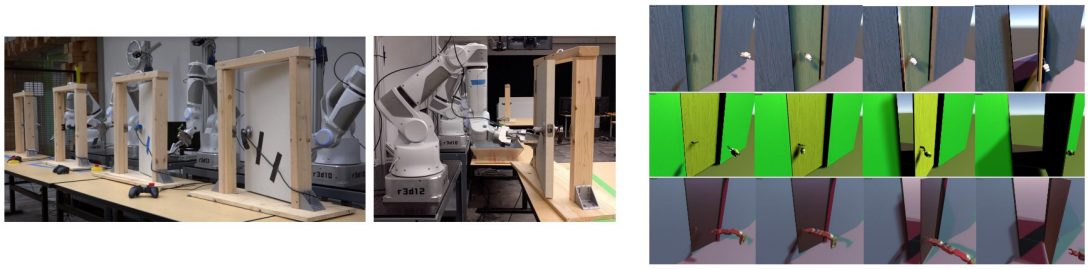


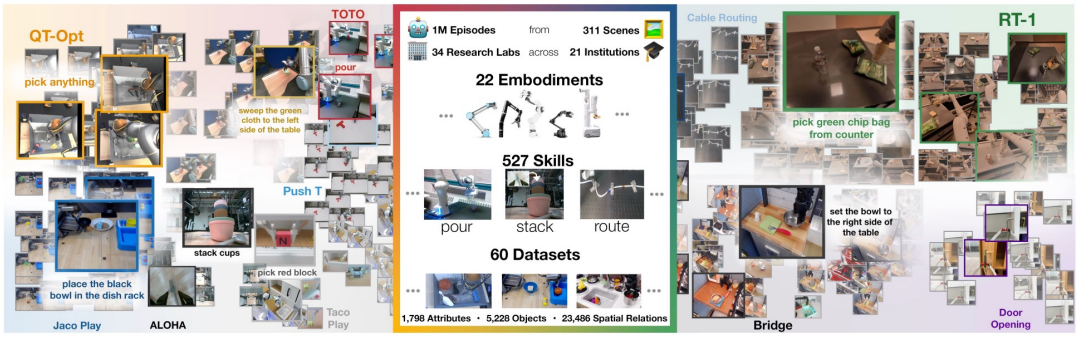
図 2.14　強化学習を用いた compliant manipulation の学習．(A) は Yahya らによる実機を用いたドア開けデータの収集と実行 [22]，(B) は Urakami らによる多様な環境でのドア開けの実行 [23].

図 2.15　Karayiannidis らによる Prismatic，Revolute 拘束を持つ物体に対する操作 [24].

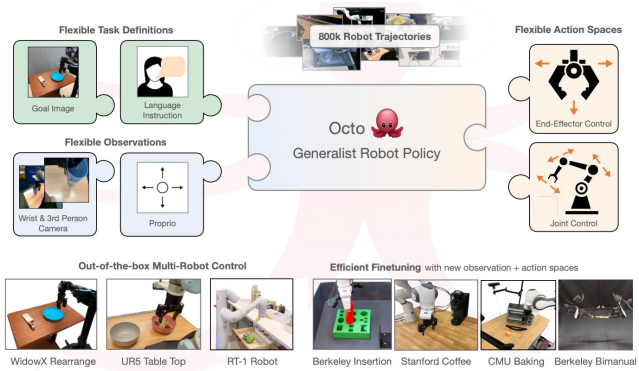
る [25, 26, 42, 76, 148–151]．このような研究には，様々な操作に対して強化学習を行うようなマルチタスク強化学習や，人間の専門家によって事前に収集された大規模データセットを用いて模倣学習を行うものがある．例えば，Open X-Embodiement [25] では，様々なロボットを用いて約 80 万エピソードのデータを収集した (図 2.16-(A))．Octo でも同様に大規模データを収集し，学習したデータやモデルをオープンソースとして公開した (図 2.16-(B))．これらの研究では，データセットに含まれた操作に対しては汎化できることが示されているが，含まれていない操作に対する汎化性能は未だに低い [31, 42]．構造化されていない環境において全ての操作を事前に想定しておくことはできないため，このような手法を未知の操作がよく出現する家庭内で動作させることは困難である．

2.4.3　Compliant manipulation の分類

全ての操作を具体的に列挙することはできないものの，物体に働く拘束に基づいて全ての操作を抽象的な表現によって分類できることは示されている [3, 27, 48]．例えば，池内らの研究 [27] では，物体間の接触状態の遷移を解析することで全ての操作を物理的及び意味的拘束に基づいて網羅的に分類した (図 2.17)．しかしながら，ある分類に属する操作に対する制御器の設計に関しては議論されていない．



(A)



(B)

図 2.16　大規模データを用いた模倣学習．(A) は Open X-Embodiement Collaboration による大規模データの概要 [25]，(B) は Octo Model Team による大規模データの概要と学習したモデルの実行 [26].

2.4.4　まとめ

以上の関連研究の特徴を表 2.3 に示す．この表は，家庭内作業を行うためのスキルライブラリに求められる要素である，家庭内作業において適したプリミティブの選択が可能，非構造化環境において再利用可能なスキルという観点から関連研究をまとめたものである．非構造化環境において再利用可能なスキルというのは，多様な操作に対して再利用可能であるということである．大規模データを用いて模倣学習を行う手法 [25,26,42,76,148–151] に関しては，プリミティブ内の動作かどうかに関わらず学習した様々な操作に対して汎用性がある一方で学習範囲外の操作に対しては汎用性が低いため，どちらも項目も△とした．以上から，compliant manipulation において上述したスキルライブラリに求められる要素の二点全てを満たす研究は未だに存在しない．

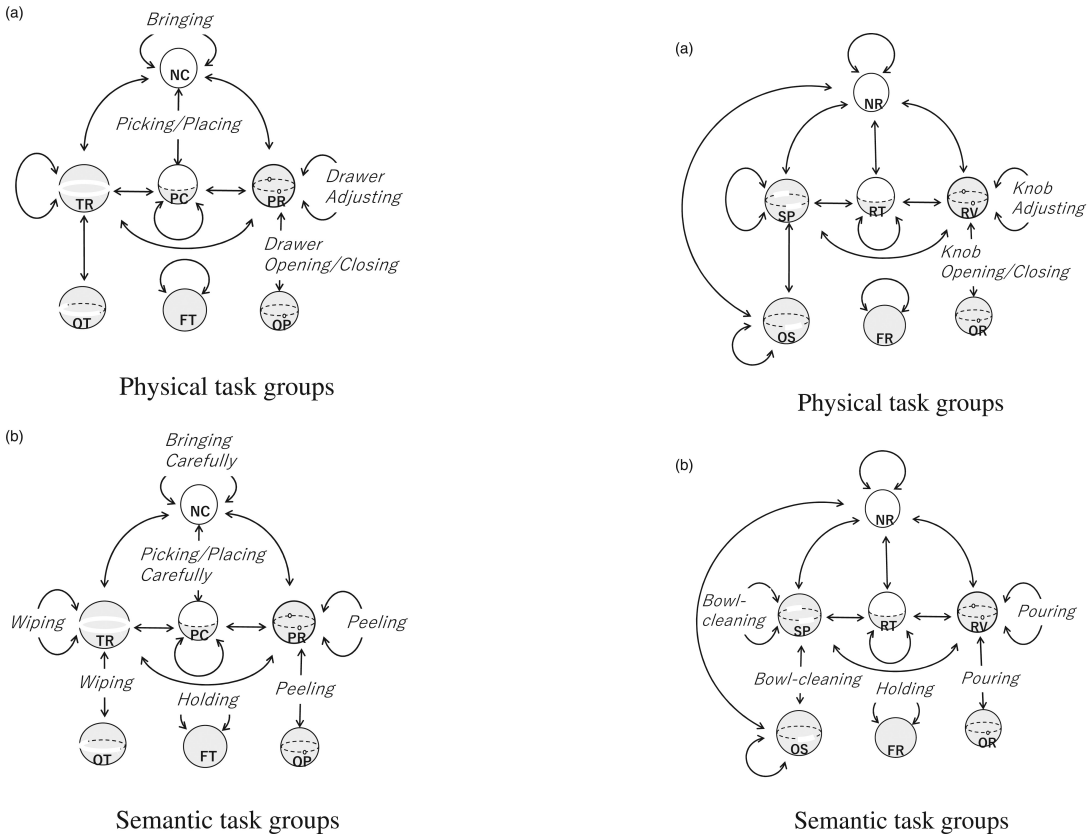


図 2.17　池内らによる接触状態から導出された物体拘束の遷移による操作の分類 [27].

	家庭内作業において適したプリミティブの選択が可能	非構造化環境において再利用可能なスキル
[Klingbeil, et al., 2010] ほかの特定操作に対するスキル設計 ([18, 19, 22, 23, 57, 64–66, 133–140])	×	×
[Karayiannidis, et al., 2016] ほかの共通拘束を持つ操作に汎用的なスキル設計 ([24, 147])	×	×
[Open X-Embodiement, 2023] ほかの複数操作に対するスキル学習 ([25, 26, 42, 76, 148–151])	△	△
[Ikeuchi, et al., 2024] ほかの compliant manipulation の分類 [3, 27, 48]	○	×

表 2.3　Compliant manipulation の関連研究のまとめ.

2.5 関連研究全体のまとめ

冒頭で述べたように，多様な家庭内作業を実行可能なロボットを実現するための一つの方法として，作業全体を適当な動作単位に分割し，各動作単位に対して再利用可能なスキルを学習させ，学習したスキルを組み合わせるということが挙げられた．これらのスキルは事前に複数個用意され，これらはスキルライブラリと呼ばれる．スキルライブラリとしてどのようなものが必要十分であるのかを考察する分野として人間行動観察学習があったが，これまでの多くの研究では工場環境が想定されており，家庭環境に関してはあまり取り扱われていなかったことが分かった．本論文で対象とする家庭内動作に関しては Grasp，In-hand manipulation は家庭内動作向けのプリミティブ分類はなされておらず，compliant manipulation に関しては分類は行われているものの，プリミティブスキルの設計はされていなかった．以上から，家庭内作業を行うための Grasp，In-hand manipulation，Compliant manipulation のスキルライブラリを新たに設計する必要があることが分かった．

家庭内作業を行うためのスキルライブラリに求められる要素としては，家庭内作業において適したプリミティブの選択が可能，非構造化環境において再利用可能なスキルという二つが求められる．これらの要素に着目して Grasp，In-hand manipulation，Compliant manipulation の関連研究を整理した．その結果，これら二つの要素を満たすスキルライブラリの設計は重要であるのにも関わらず，これまでに行われた研究は存在しないことが分かった．そこで，本論文では，Grasp，In-hand manipulation，Compliant manipulation の三つの分野に対して，そのようなスキルライブラリの設計を行うことを目指す．

第 3 章

トップダウン知識に基づくロボットスキルライブラリの設計の提案

3.1 目的

第 1 章で述べたように，本論文では，手腕を巧みに用いた動作を含む家庭内作業が可能な家庭用ロボットに向けたスキルライブラリの設計を目的とする．手腕を用いた動作の中でも，特にタスクの遂行において必要不可欠な Grasp，In-hand manipulation，Compliant manipulation に焦点を当ててスキルを設計する．スキルライブラリには以下の二点が求められる．

1. 家庭内作業において適したプリミティブの選択が可能
2. 非構造化環境において再利用可能なスキル

本論文では，以上の問題点を解決するために，トップダウン知識に基づいたロボットスキルライブラリの設計を提案する．以下では，各動作に対する問題点とそれに対するアプローチをさらに詳しく説明する．

なお，本論文で使用するロボットに関して，ロボットアームは逆運動学が十分に解ける自由度を持ち，物体操作に必要な力を十分に発揮可能であると仮定する．これによって，重い冷蔵庫の扉のような日用品等を操作可能であることが保証される．また，ロボットハンドとしては，人間が使用することを前提に作られた多様な家庭内用品を単一ハードウェアで扱うために多指ロボットハンドを使用する．なお，ここでいう多指ロボットハンドとは，対抗する指があり，かつ 4 本以上の指を持ったハンドのことである．4 本であるのは，In-hand manipulation で安定把持を維持するのに必要な最低本数を仮定するためである．プリミティブはこの仮定を満たしたハードウェアに依存しない観点で分類する．本論文で対象とするハンドは全てのプリミティブの実現が可能である．

3.2 アプローチ

本論文では，まず，家庭内作業における動作をトップダウン知識に基づいてプリミティブに分類する．そして，それらのプリミティブからスキルライブラリを構成する．次に，動

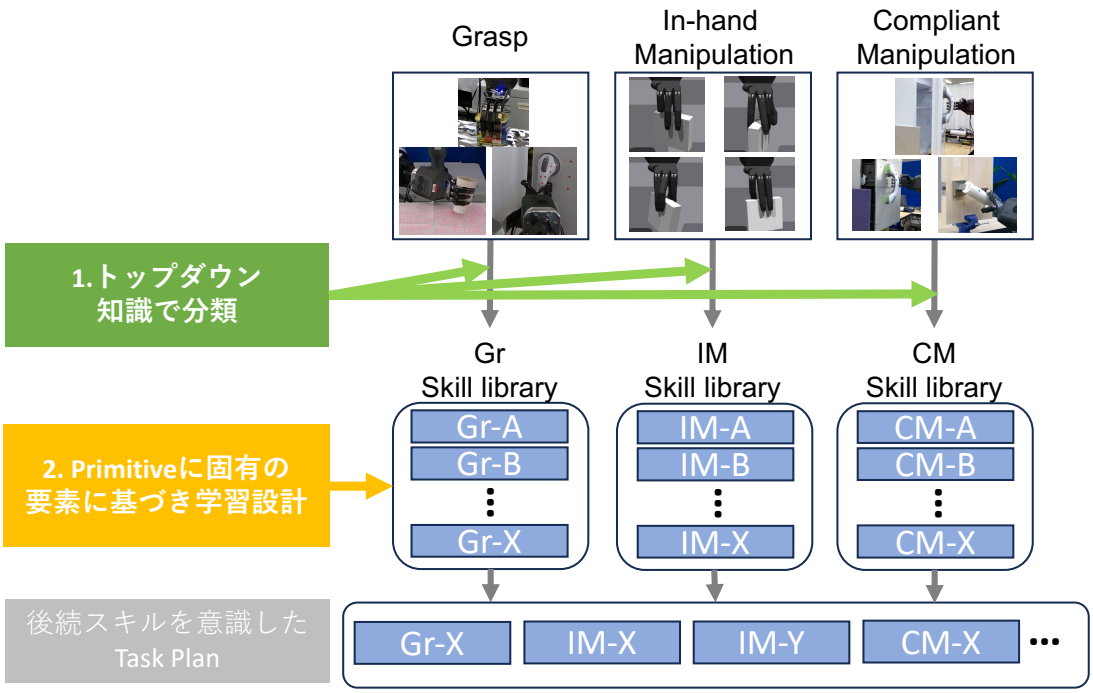


図 3.1　本論文の提案．トップダウン知識から分類されたプリミティブからライブラリを構成し，トップダウン知識に基づいて学習設計をする．Gr は Grasp，IM は In-hand manipulation，CM は Compliant manipulation の略である．

作プリミティブごとに共通の特徴に基づいてスキルを学習させることを提案する (図 3.1)．この学習設計によって，動作プリミティブの目的達成のための要素を実現するような学習が行える．さらに，共通の特徴に基づいた設計によりプリミティブが含む多様な動作に対しての再利用可能性を実現する．

本論文では，Learning-from-Observation(LfO) [3, 4] の枠組みに繋げることを前提としてスキルライブラリを設計する (図 3.2)．LfO では，まず人間の身振り手振りの実演と言語指示から動作の手順や動作の際に必要なスキルパラメータを獲得する．そして，このスキルパラメータから実行されるスキルやその実行順を決定する．そのため，LfO はタスクプランナの一種とみなすことができる．具体的には，実演全体の動作プリミティブへの分割やその順番，動作プリミティブの種類，操作の粗い軌道といったパラメータを取得できる．このようにパラメータを豊富に入手できるため，本論文ではタスクプランナとして LfO を使用することを前提とする．実行時には，実演時に得られたパラメータを用いてスキルライブラリからプリミティブスキルが選択される．なお，LfO では人間の実演をそのまま模倣するのではなく動作の本質的な部分のみを再現する．例えば，重い冷蔵庫の扉を開けるときに人間の実演では取っ手に指をひっかけることで扉が開けるが，このときに関節角度そのものを模倣するのではなく，ひっかけるように手を物体に接触させるという本質的な部分のみを再現する．以上のような動作の本質的な部分の再現によって，適切にスキル実行することが可能となる．このような動作の本質的な部分に着目してプリミティブ分類を行う必要がある．

LfO ではプリミティブの分類は物体に対する接触状態の遷移に基づいて行われる．本論文でもこの設計思想に準じてプリミティブの分類を行う．まず，作業をおおまかに物体と手が接触するまでとその後で分類する．この分類における前者を Grasp と定義する．後者は操作と呼ばれるが，その中でも，手と物体との接触状態を変化させる操作を In-hand manipulation，環境と物体との接触状態を変化させる操作を Compliant manipulation と定義する．そして，各動作単位に対して細かく観察すると，動作単位の中でも接触状態遷移の性質が異なり，この性質で動作単位をプリミティブに分類する．本論文では，この各動作単位に対するプリミティブ分類を目指す．分類時にプリミティブを大雑把に決めてプリミティブ間で動作に重複があるような状態になってしまうと，冗長な数のプリミティブスキルを用意しなければならない恐れがある．これを防ぐために，プリミティブを独立した観点から重複なく分けて，プリミティブの冗長性を減らす．

本論文では，プリミティブスキルの学習は強化学習によって行う．スキル学習において重要な点は，そのプリミティブの本質的な部分が再現されることである．強化学習ではプリミティブの本質的な部分を明示的に報酬で表現でき達成できたかどうかが確認できるため，強化学習によるスキル獲得を採用する．なお，ロボット学習の分野では逆強化学習や模倣学習のような実演の模倣も学習手法の選択肢として考えられるが，このような手法では意図したプリミティブの本質的な部分が再現される保証がないため，本論文では学習手法として採用しない．以下では，それぞれの動作に対するスキルライブラリの設計指針を述べる．

3.2.1 Grasp

後続スキルの成功のためには物体の把持時に適切に物体へ力をかけておくことが重要である．例えば，重い扉を引っ張ることを考える．この時，指先のみでハンドルを把持するのでは引っ張るのに十分な力が発揮できず失敗する．一方で，ハンドルに指を引っ掛ければハンドルを完全に拘束できるため，十分な力が発揮できる．以上のように，力のかけ方に着目してスキルを選択することが後続スキルの成功のためには重要である．この問題に関しては，人間の把持プリミティブを参考にすることで解決することができる．力のかけ方という観点で人間の把持をいくつかのプリミティブに分類したものとして Kang の分類がある [28]．Kang の分類は工場環境での動作の解析により導出されたものであるため，家庭内作業に対しては拡張の必要がある．さらに，LfO では人間の実演から得られた把持プリミティブに対応したスキルを選択することから，人間の把持とロボットの把持を対応づける必要がある．これは，人間とロボットでは手の構造が異なるため，単純に人間の把持をロボットで再現することは難しいことに由来する．

そこで本論文では，まず家庭環境での把持分類 [9] と Kang の把持分類を比較することで，Kang の分類を家庭環境における把持分類に拡張する．次に，人間の把持からロボットの把持への対応付けを行う．本論文では，力のかけ方という観点でのロボットの把持分類 [51] への対応付けを行う．その際，家庭環境で頻出の non-prehensile grasp に対応するプリミティブがロボットの把持分類に含まれていないため，これに対応するプリミ

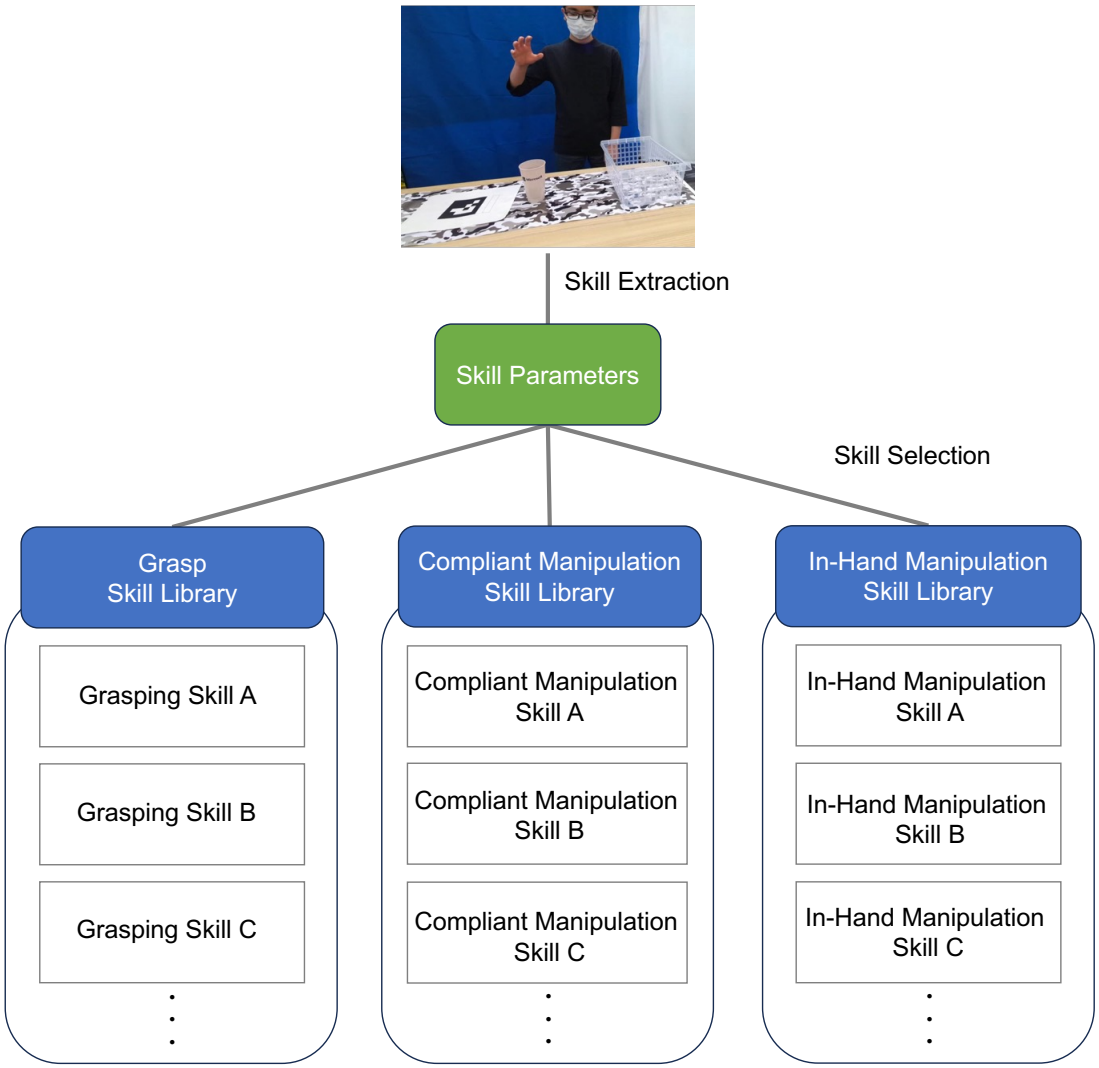


図 3.2　Learning-from-Observation へのスキルライブラリの組み込み．人間の実演からスキルパラメータが抽出され，そのパラメータから実行されるスキルやその実行順が決定される．

ティブを本論文では新たに Lazy-closure として導入する．そして，これらを含めた分類を Force-exertion type として提案する．

次に，Force-exertion type に含まれるプリミティブに対してスキル設計を行う．Kang らによる Contact web の議論 [28, 45] に基づけば，各プリミティブを特徴づけているのはその接触点の分布である．すなわち，どのように接触点が分布していて，その接触点からどの方向に力がかかっているかによってプリミティブが特徴づけられる．そこで，本論文では接触点と接触力方向に基づく報酬設計によって各プリミティブの強化学習を行うことを提案する (図 3.3).

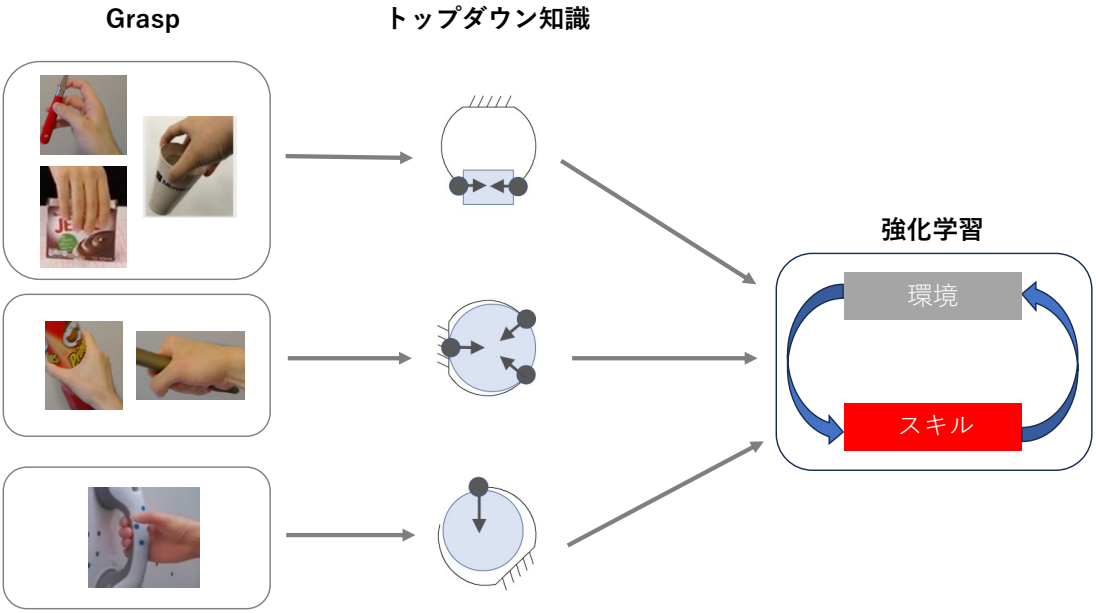


図 3.3　Grasp のスキル設計に関する提案の概念図.

3.2.2　In-hand manipulation

In-hand manipulation の中でも作業に適した把持プリミティブを実現することに焦点を当てたスキルの学習を目指す．把持プリミティブは接触点の分布といった接触状態によって特徴づけられるため，プリミティブの変化は接触状態の遷移とみなすことができる．この接触状態の遷移には，長期的な接触状態の変化と空間的に多様な動作が伴う．そのため，既存研究のように物体の回転のみに着目するような学習手法 [13,16,59,119–123] によって直接的にこのスキルを獲得することが困難である．

In-hand manipulation は全体としては長期的で複雑な動きをしているように見えるが，三種類の簡単な動作のまとまりから構成されており，この動作に着目することで短期的な動作に分割できる [17]．この動作とは detach, crossover, attach の三種類の動作である．detach とは指が物体から離れる動作，crossover とは指が物体をまたぐ動作，attach とは指が物体に接触する動作である．これらの動作によって操作全体を記述可能で，この記述に基づいて動作分割することで探索すべき領域を削減できる．

そこで本論文では，接触状態の遷移を三種類の動作を用いてより細かな単位に分割した APriCoT(Action Primitives based on Contact-state Transition) を導入し，これらの単位ごとに学習を行うことを提案する ((図 3.4))．この方法によって，元々の操作の持つ長期的な接触状態の遷移と空間的に多様な動作が緩和されるため，学習が容易になる．まず，接触状態としてあり得るものを列挙し，その状態間の遷移をプリミティブとする．次に，各プリミティブに対して，上述した三種類の動作が発生するように設計された報酬を用いて強化学習によるスキル獲得を行う．

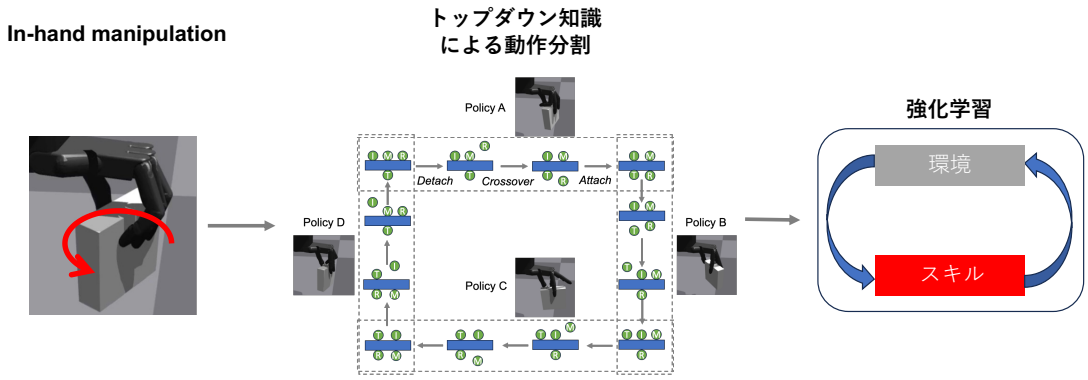


図 3.4　In-hand manipulation のスキル設計に関する提案の概念図.

3.2.3　Compliant manipulation

家庭環境ではどのような操作が出現するか事前に予期することは難しい．そのため，様々な操作に汎用的なスキルを学習することを目指す．多くの操作データを事前に収集しておき模倣学習することでデータに含まれる様々な操作に汎用的なスキルを学習することは可能である [25, 26, 42, 76, 148–151] が，未経験の操作に対しては汎化できない．そのため，家庭環境で動かすことは難しい．

実は，compliant manipulation が必要な操作は物理的拘束に基づいて有限個の操作プリミティブに分類することができる [27]．物理的拘束によって物体が動くことが可能もしくは不可能な方向が決定するが，この方向に関して共通の集合を持つ操作を一つの操作プリミティブとしている．例えば，引き出し開けや板を引く，棒を引くといった操作は物体の許容される運動方向がある直線上に拘束されているため，同一の操作プリミティブに属する．物体が許容されない方向に動こうとすると拘束によって物体に拘束力が発生する．したがって，物理的拘束に基づいて分類された操作は拘束力方向が共通しているという特徴を持っていることが分かる．

そこで本論文では，拘束力方向から物体の許容される運動方向を推定する Constraint-aware policy というスキルを提案する (図 3.5)．このスキルは拘束力方向を基にして動作するため，同一操作プリミティブ内に含まれる全ての操作に汎用的となる．このスキルは単一環境と報酬によって学習される．学習環境は手と操作物体を一体にした複合体と物理的拘束から構成される．この環境は拘束力に関する共通の特徴を抽出して実世界の操作を単純化したものとして設計されており，これが汎化において重要となる．報酬は拘束力の大きさを用いて設計される．

Compliant manipulation が必要な全ての操作は有限個のプリミティブに分類可能であるため，各プリミティブに対してスキルを設計していくことで全ての操作に対して適用可能なスキルライブラリが構築できる可能性がある．この方針の違いが上述した既存研究 [25, 26, 42, 76, 148–151] との大きな違いである．

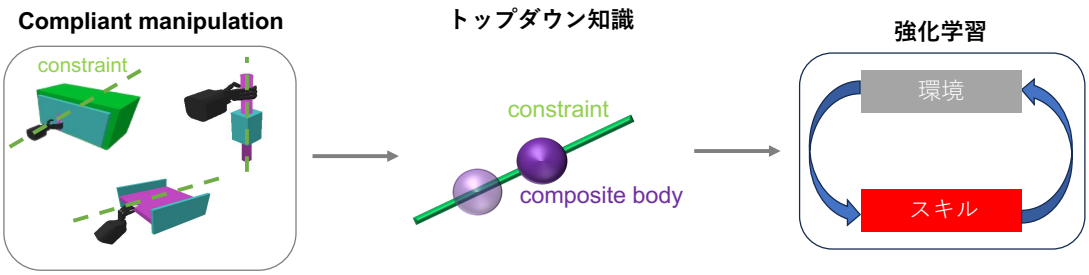


図 3.5　Compliant manipulation のスキル設計に関する提案の概念図.

3.3　本研究の貢献

本研究の貢献は以下の通りである.

1. トップダウン知識を用いたスキルライブラリの設計とその学習方法を提案する．提案手法により設計されたスキルライブラリは家庭内作業を遂行する上で重要な二つの要素を満たすように学習される．すなわち，家庭内作業において適したプリミティブの選択が可能であり，なおかつ非構造化環境において再利用可能なスキルによって構成される.
2. 提案するスキルライブラリとタスクプランナの一種である Learning-from-Observation の枠組みを組み合わせることにより，後続スキルを意識してスキル実行しなければ成功しないようなタスクが実際に成功することを示す.
3. シミュレーションと実機を用いて提案手法により学習されたスキルの性能を評価し，学習したスキルに再利用可能性があるのかどうかを確認する．また，学習されたスキルが実世界で実行可能であるかどうかを調査する.

3.4　将来的なシステム全体像

本論文が目指す将来的な LfO システムの全体像の概要を図 3.6 に示す．このシステムは teaching mode と execution mode の二つの段階に分かれている．Teaching mode では，まず人間の実演として一連の作業中の言語指示と動画データが入力される．この実演には，Grasp，In-hand manipulation，Compliant manipulation が含まれる．この実演データに対して task parser により動作分割を行った後に，recognition module を用いて各動作に対する skill parameter を取得する．Execution mode では，前段階で得られた skill parameter に応じたスキルをライブラリから選択し，このスキルと skill parameter から作業を実行する.

以下では，システム全体像の具体的な説明を行う．具体的な例を図 3.7 に示す．適したプリミティブを選択するために，一連の動作を動作単位に分割する必要があるが，これは言語指示から動詞に応じて分割する [85]．そして，各動作単位に対応するプリミティブを選

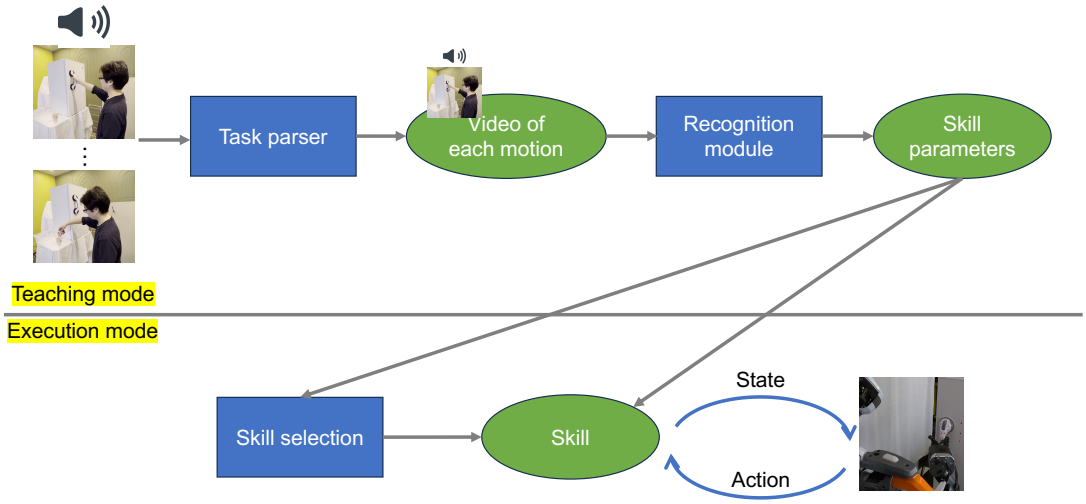


図 3.6　将来的な LfO システム全体像の概要.

択する．Grasp の場合は，物体把持時の画像を convolutional neural network に入力した際に得られるプリミティブの確率と，物体に対して事前に定義されたあり得るプリミティブの確率 (Affordance database) の二つを基にして，尤もらしいプリミティブが推定される [86]．さらに，把持までの手の軌道を手の位置姿勢推定を用いて求めることでアプローチ方向を取得する．Compliant manipulation の場合は，動詞と物体拘束の種類に関連があるという性質から言語指示入力から操作プリミティブが推定される [85]．この推定には BERT [152] が用いられる．さらに，操作中の手の軌道を手の位置姿勢推定を用いて取得する．In-hand manipulation の場合は，操作の最初と最後の時点の把持画像と物体名称から把持プリミティブが推定される．そして，物体位置姿勢推定を用いることで操作における物体位置姿勢の変化量を取得する．

Execution mode では，teaching mode で得られたスキルパラメータを用いてロボット動作を生成する．分割された各動作単位に対してスキルパラメータを基にした実行を順にしていくことでロボットが作業を遂行できる．Grasp の場合は，teaching mode で得られた把持プリミティブに対応したプリミティブスキルが選択され，そのスキルにアプローチ方向を含んだ状態を入力することで把持動作が実行される．Compliant manipulation の場合は，操作プリミティブに対応したプリミティブスキルが選択され，そのスキルによって手の軌道を修正しながら操作が実行される．In-hand manipulation の場合は，把持の遷移と物体位置姿勢の変化量に基づいて，適したプリミティブスキルが選択されて実行される．

本論文では，Grasp，Compliant manipulation，In-hand manipulation の三つの動作単位に対して，プリミティブスキルが含まれるスキルライブラリの設計を目指す．

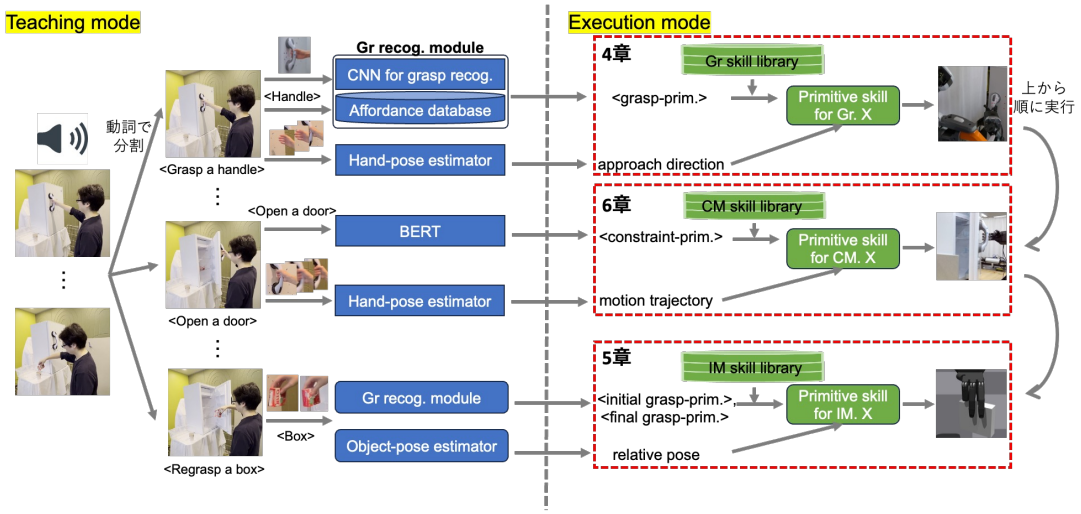


図 3.7　将来的な LfO システム全体像の具体例．Gr は Grasp，IM は In-hand manipulation，CM は Compliant manipulation の略である．

第 4 章

Grasp に関するスキルライブラリの設計

4.1 目的とアプローチ

本章では，家庭内作業に向けた LfO のための Grasp のスキルライブラリの設計を行うことを目的とする．家庭内作業の成功には，後続スキルに適した把持プリミティブを選択する必要がある．後続スキルの成功のためには，安定した把持を達成する以外にも，物体にどのように力をかけておくかが重要である．図 4.1 は後続スキルを意識せずに把持を行った場合と意識して把持を行った場合の比較を表している．図 4.1 は重い扉のハンドルを把持した後に扉を開くスキルの実行を示している．図 4.1 の上のようにハンドルを指先のみで把持した場合，扉を開くのに十分な力が発揮できず失敗する．一方で，ハンドルに指を引っ掛ければハンドルを完全に拘束できるため，十分な力が発揮できる．以上のように，力のかけ方に着目してスキルを選択することが後続スキルの成功のためには重要である．

この問題に関しては，人間の把持プリミティブを参考にすることで解決することができる．力のかけ方という観点で人間の把持をいくつかのプリミティブに分類したものとして Kang の分類がある [28]．Kang の分類は工場環境での動作の解析により導出されたものであるため，家庭内作業に対しては拡張の必要がある．さらに，LfO では人間の実演から得られた把持プリミティブに対応したスキルを選択することから，人間の把持とロボットの把持を対応づける必要がある．これは，人間とロボットでは手の構造が異なるため，単純に人間の把持をロボットで再現することは難しいことに由来する．例えば，Shadow Dexterous Hand Lite のような 4 本指のロボットハンドでは小指が存在しないため，Prismatic-3 grasp と Prismatic-4 grasp の差を再現することはできない．

そこで本章では，まず家庭環境での把持分類 [9] と Kang の把持分類を比較することで，Kang の分類を家庭環境における把持分類に拡張する．次に，人間の把持からロボットの把持への対応付けを行う．本論文では，力のかけ方という観点でのロボットの把持分類 [51] への対応付けを行う．ロボット把持分類は力のかけ方に依存して変化する安定性と操作性の観点から把持を分類したものである．したがって，Kang らによる人間の把持分類と吉川によるロボットの把持分類を対応付けすることを試みる．その際，non-prehensile contact

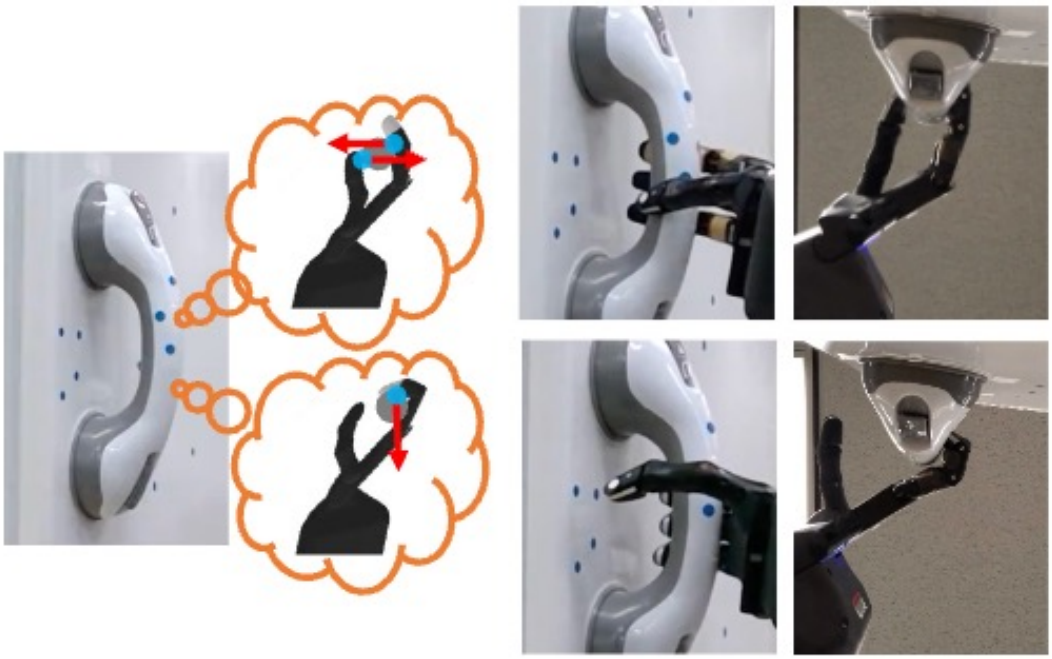


図 4.1　2 種類の把持候補を使ってドアを開ける例．上図が指先のみで把持した場合，下図が引っ掛けるように把持した場合．左図は 2 つの把持による物体への力のかけ方の違いを示している．青い点が接触点，赤い矢印が物体への力方向である．

grasp に対応するプリミティブがロボットの把持分類に含まれていないため，これに対応するプリミティブを本論文では新たに Lazy-closure として導入する．そして，これらを含めた分類を Force-exertion type として提案する．

次に，Force-exertion type に含まれるプリミティブに対してスキル設計を行う．Kang らによる Contact web の議論 [28] に基づけば，各プリミティブを特徴づけているのはその接触点の分布である．すなわち，どのように接触点が分布していて，その接触点からどの方向に力がかかっているかによってプリミティブが特徴づけられる．そこで，本論文では接触点と接触力方向に基づく報酬設計によって各プリミティブの強化学習を行うことを提案する．各プリミティブで定義された接触点と接触力方向を満たした把持を実現するように強化学習を行う．なお，このスキルは接触点群の座標系が推定できているという前提で設計される．この推定に関しては既存研究と同様に画像から推定する．そのために，接触点群認識器の学習も行う．したがって，スキル実行時はまず対象物体の写った深度画像から接触点群の座標系を取得し，この座標系を参考にして学習されたスキルを実行するという流れを取る (図 4.2)．

以降では，人間の把持とロボットの把持の対応付け，スキルの強化学習，接触点群認識に関して説明する．そして，LfO へのスキルの組み込みに関しても説明する．最後に，シミュレーション実験と実機実験を通して本手法の有効性を示す．

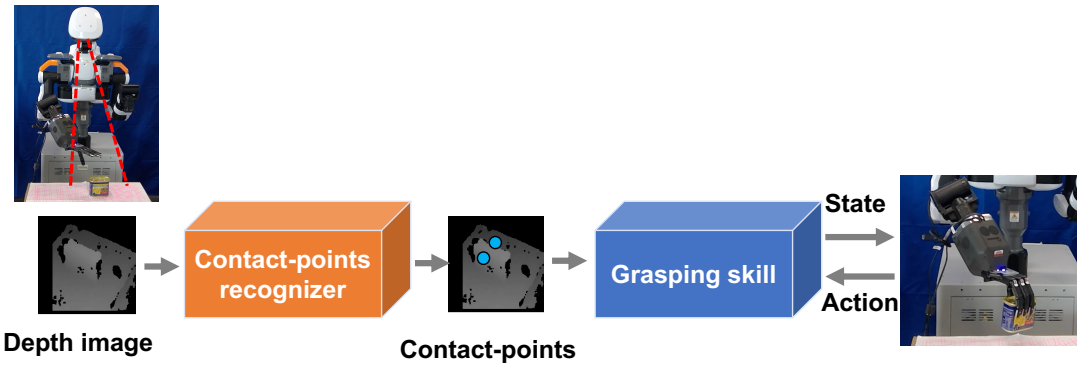


図 4.2　実行時のパイプライン．まず，ロボットに装着されたカメラから深度画像を取得する．次に取得された深度画像を接触点群認識器に入力し接触点群を出力する．得られた接触点群を参考に把持スキルを実行する．

4.2　人間の把持とロボットの把持の対応付け

まず Kang の把持分類と家庭環境での把持分類 [9] と比較することで，Kang の分類を家庭環境における分類に拡張する．Kang の分類では接触点の数や接触している指の種類や形状で把持を分類する．まず手のひらを接触点に含まないもの (Non-volar) と含むもの (Volar) で把持を大別する．Non-volar に関しては，指先のみで把持を行ったもの (fingertip) と指の側面も含めて把持を行ったもの (composite) でまた把持を分類する．次に，接触点の数によって把持をさらに細かく分類する．Volar に関しては指の形状によってさらに細かく分類する．Kang の分類は家庭環境における把持をほぼ網羅しているが，hook 把持だけが分類されていない．hook 把持は，手のひらと親指は把持に関与せず物体の重さは指のみにかかるというものであり，non-prehensile grasp の一種である．この hook 把持は Kang による分類では Non-volar の fingertip に分類することができる．そこで，Kang の分類にこの把持を non-prehensile contact grasp として加える (図 4.3)．以上の人間の把持分類をロボットの把持分類に対応付けさせる．

ロボットの把持分類としては吉川による力のかけ方によって分類されたものを採用する [51]．この分類はロボット把持を力のかけ方に依存して変化する安定性と操作性の観点から分類したものである．安定性の観点による分類とは，どんな外力に対しても拘束具から発生している力を変化させずに外力を打ち消すことができるかどうかで分類するということである．できる場合は form closure，できない場合は force closure として分類される．操作性の観点による分類とは，把持後に手の中で物体の姿勢を変化させることが可能かどうかで分類するということである．できる場合は active closure，できない場合は passive closure として分類される．吉川の分類では，これらの概念を用いて把持を active-force closure，passive-force closure，passive-form closure の三つに分類している．

これらの対応付けを表したものが図 4.4 である．Non-volar の fingertip に含まれる把持の多くは指先飲みによる拘束のため active-force closure に分類される．Non-volar の

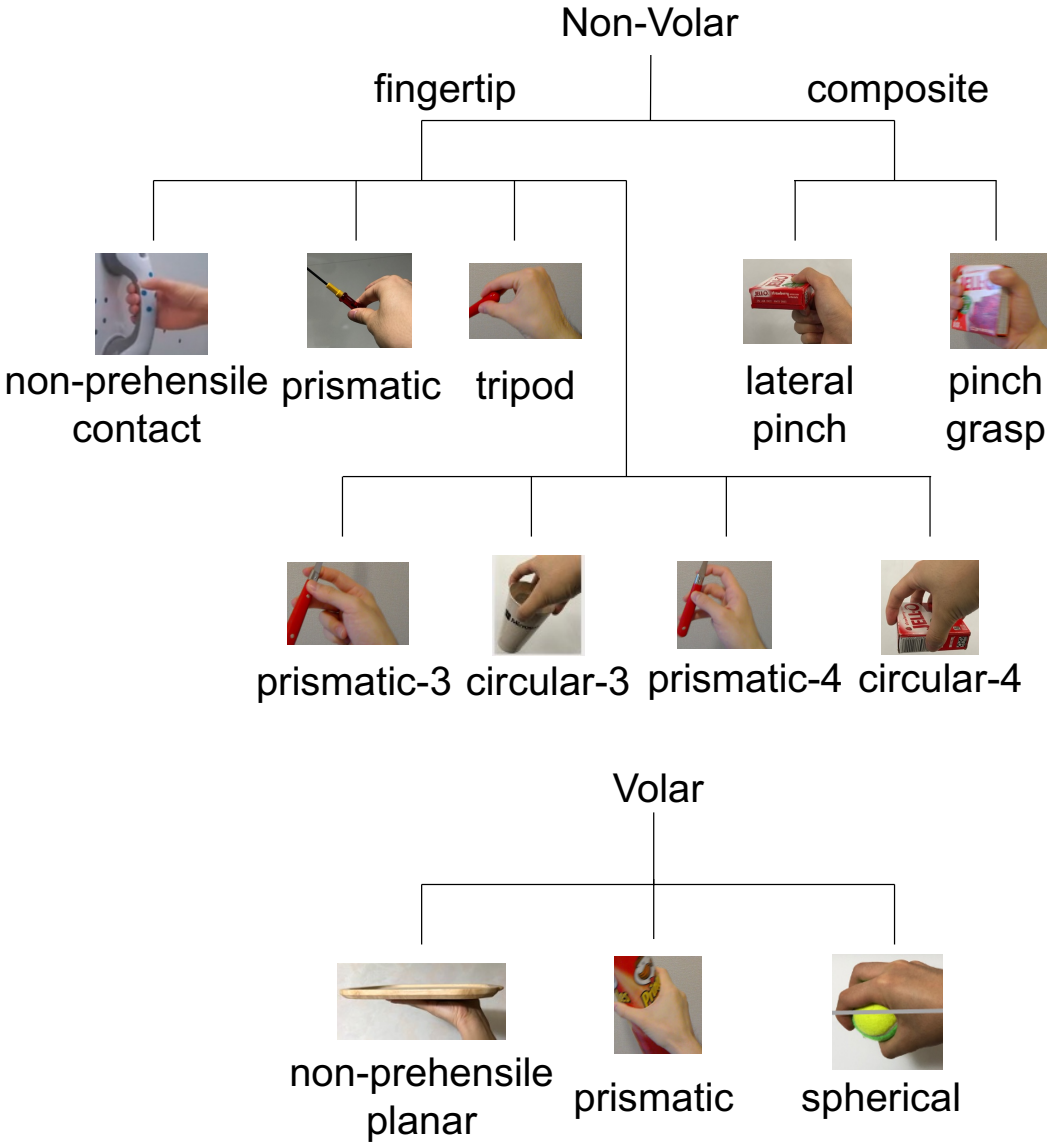


図 4.3　Kang による人間の把持の分類 [28]．図は [28] の図を編集したものである．

composite に含まれる把持や Volar に含まれる非把持以外のものは passive-force，passive-form closure として分類される．Volar の non-prehensile planar に関しては任意の方向からの力に抗うことは不可能であるので closure は成立されておらず non-closure として分類される．Non-volar の non-prehensile contact に関しては，簡単に closure に分類することができない．Non-prehensile contact は，扉に固定されたハンドルのような環境によって固定されている物体が把持対象物体である．そのため，把持対象物体が外力に対して自己補償されており，外力に抗うのに余分な力を必要としないため form closure とみなすことができる．一方で，把持の安定性を保ちながら手と対象物の位置関係を変えることができるという点では active closure とみなすことができる．これに対応する closure は吉川の分類には含まれていない．そこで本論文ではこの把持を lazy-closure と定義し，以上の把

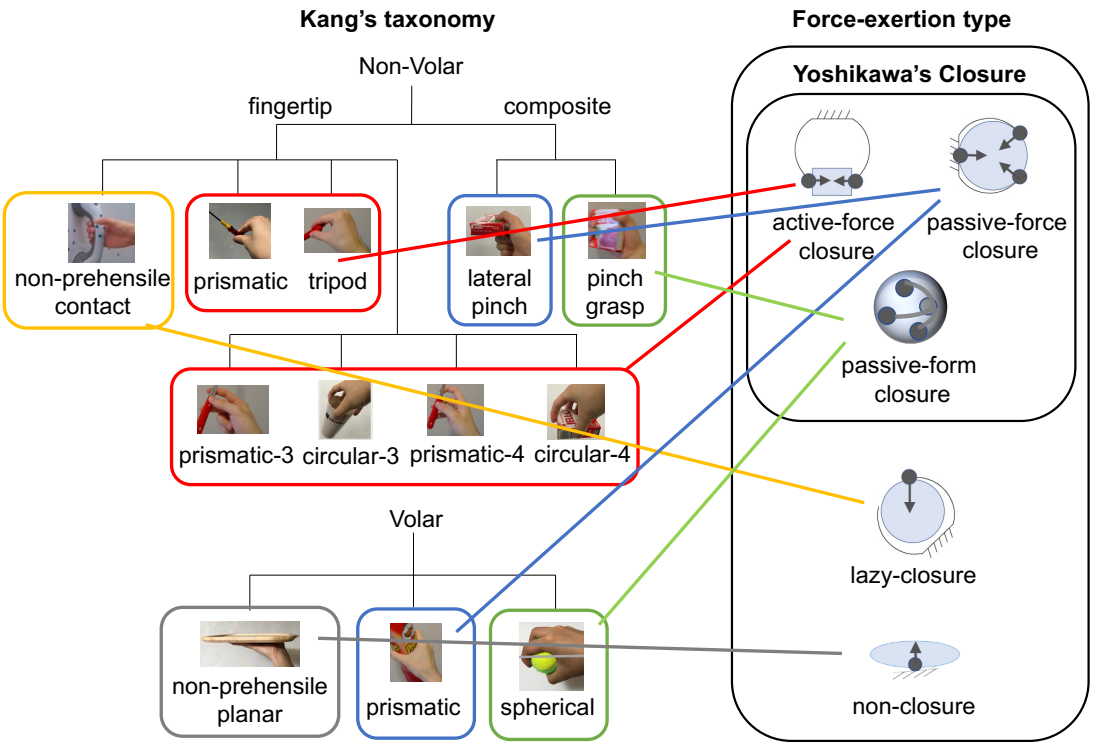


図 4.4　人間の把持分類 (左) と force-exertion type(右) との対応付け.

持プリミティブを含む分類を force-exertion type とする．なお，これらのタイプは指の本数や手の形状に基づくものではないため，図 4.5 に示すような単純な 1 自由度グリッパーであっても把持を区別できる．

Force-exertion type の中には一度の把持動作で実現できるものと，他の force-exertion type から遷移しなければ実現できないものが存在する．後者には passive-form closure と non-closure が該当する．これらは active-force closure からの遷移が必要になる．本論文では，一度の把持動作のみで実現可能な active-force closure，passive-force closure，lazy-closure のスキル設計に焦点を当てる．

4.3　スキルの強化学習

上述した Force-exertion type に含まれる active-force closure，passive-force closure，lazy-closure に対してスキル設計を行う．Kang らによる contact web の議論 [28] に基づけば，各プリミティブを特徴づけているのはその接触点の分布である．すなわち，どのように接触点が分布していて，その接触点からどの方向に力がかかっているかによってプリミティブが特徴づけられる．この接触点分布は図 4.5 の下のように表現できる．Active-force closure では物体を安定して把持できるように，物体を挟むように指先による二つの接触点が配置される．Passive-force closure では指先による二つの接触点に加えて，安定性を増すための手のひらによる接触点の合計三つが配置される．Lazy-closure では接触点が一つ

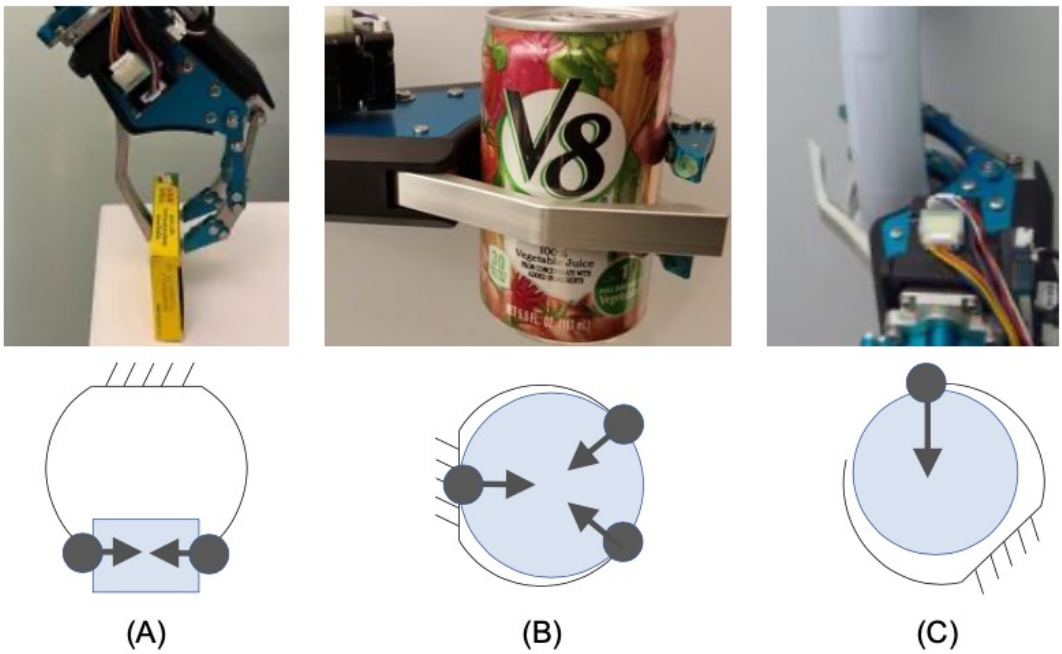


図 4.5　1 自由度グリッパーによる force-exertion type（上）と，それに対応する接触点の例（下）．黒丸は接触点，矢印は力の方向を示す．黒い円弧はグリッパーを表す．

となる．Lazy-closure の場合には，物体は把持を行わなくても安定しているため，動かしたい方向に力を発生させられるように接触点が配置される．なお，ここでは最小の数で接触点を表現しており，必要であればさらに多くの接触点を配置することもできる．最小数で表現することにより指の本数が異なるロボットハンドへも適用可能であるという利点があるため，最小接触点数で表現する．

各把持プリミティブに対するスキルは強化学習によって獲得させる．これは非構造化環境に伴う不確かさへの頑健性が高いスキルを学習するためである．強化学習ではロボットが環境とインタラクションを繰り返してスキルを学習する．この学習を実機で行うのはロボットの消耗が激しいために困難である．そこで，物理シミュレータを用いて学習を行う．本研究では，机上に把持対象物体のみが置いてある，もしくは扉に把持対象のハンドルのみが付いている環境を想定している．本論文では物体情報が未知な環境での多指ハンドによる把持を想定している．そのため，ロボットは指先位置と手の姿勢は取得できるが，物体の位置姿勢や大きさ，形状は取得できない．また，ロボットは指に物体が衝突しているかどうかの二値の力覚を検知できると想定している．さらに，ロボットは様々なアプローチ方向で物体を把持しに行くことを求められる．これは，後続スキルにとって適切な位置で物体を把持するため [153, 154] である．そのため，ロボットは物体の位置姿勢や形状に対する不確かさやアプローチ方向の変化に対して再利用可能なスキルを学習することを求められる．このような学習を行うために，常に把持プリミティブを特徴づける接触点分布を目指すことを促すような接触点分布に基づく報酬を設計する．

以降ではスキルの学習方法に関して説明する．

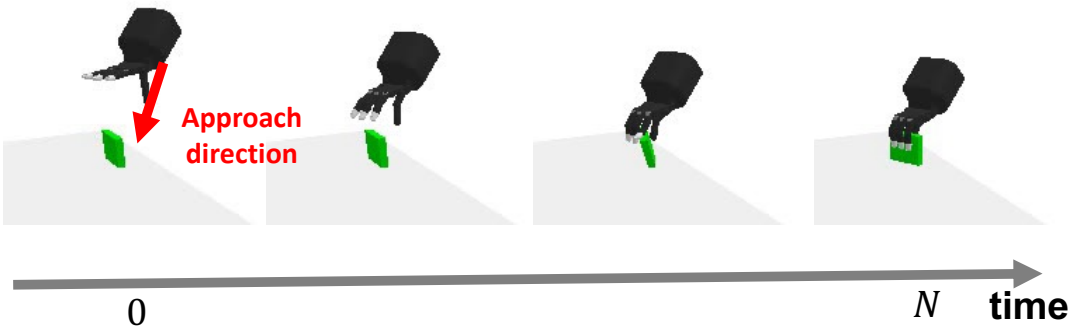


図 4.6　参照動作の例.

4.3.1　参照動作

多指ハンドは非常に多くの状態行動空間を持つため学習に非常に時間がかかると考えられる．そこで，事前に定義した参照動作を用いて状態行動空間を適切に限定する．参照動作を深層強化学習によって改善することで把持動作を学習する．以降では参照動作の生成に関して説明する．

参照動作は時刻 $t(0 \leq t \leq N)$ のロボットハンドの位置 $\boldsymbol{p}_t^{hand} \in \mathbb{R}^3$，姿勢 $\boldsymbol{q}_t^{hand} \in \mathbb{R}^4$，関節角度 $\boldsymbol{j}_t^{finger} \in \mathbb{R}^{16}$ から構成される．把持位置に到達した時刻が $t = N$ である．参照動作では手が把持物体に近づくようなアプローチ方向に進む (図 4.6)．

把持時の関節角度 $\boldsymbol{j}_N^{finger}$ を事前に収集した人間の実演から得られた関節角度 [155] とする．この時，関節角度は式 4.1 のように表される．

$$\boldsymbol{j}_t^{finger} = (1 - \frac{t}{N})\boldsymbol{j}_0^{finger} + \frac{t}{N}\boldsymbol{j}_N^{finger} \ (0 \leq t \leq N) \tag{4.1}$$

$\boldsymbol{p}_N^{hand}$ を把持時の手の位置であり，$\boldsymbol{p}_0^{hand}$ を $\boldsymbol{p}_N^{hand}$ からアプローチ方向 $\boldsymbol{d}_{app}$ と逆方向に n m 進んだ位置と定義する．把持動作を始める際の手の姿勢 $\boldsymbol{q}_0^{hand}$ はアプローチ方向によって定義される．この時，手の位置は式 4.2 で，手の姿勢は式 4.3 で表せる軌道を辿るようにする．

$$\boldsymbol{p}_t^{hand} = (1 - \frac{t}{N})\boldsymbol{p}_0^{hand} + \frac{t}{N}\boldsymbol{p}_N^{hand} \ (0 \leq t \leq N) \tag{4.2}$$

$$\boldsymbol{q}_t^{hand} = \boldsymbol{q}_0^{hand} \ (0 \leq t \leq N) \tag{4.3}$$

把持時の手の位置 $\boldsymbol{p}_N^{hand}$ は関節角度 $\boldsymbol{j}_N^{finger}$ と把持姿勢 $\boldsymbol{q}_N^{hand}$ から決定される．関節角度 $\boldsymbol{j}_N^{finger}$ と把持姿勢 $\boldsymbol{q}_N^{hand}$ の状態で，接触点群の平均位置と指定したロボットリンク群の平均位置が一致するように手の位置を決定する．active-force closure では中指と親指の指先の平均位置，passive-force closure では中指と親指の指先，手のひらの平均位置を一致させる．lazy-closure では中指の指先の位置を一致させる．

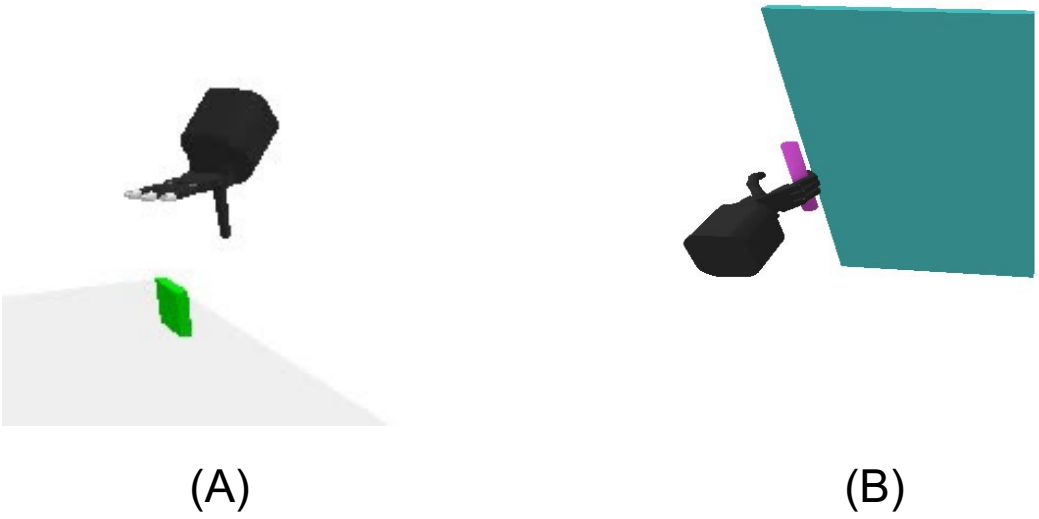


図 4.7　学習に用いる環境．(A) は active-force closure と passive-force closure の学習に用いる環境で机上に物体が置いてある．(B) は lazy-closure に用いる環境でドアにハンドルが付いた環境である．

4.3.2　環境，状態，行動，報酬

環境

図 4.7 は学習に用いる環境である．環境には日常生活で頻出する状況を再現したものを用いる．(A) は active-force closure と passive-force closure の学習に用いる環境で机上に物体が置いてある．(B) は lazy-closure に用いる環境でドアにハンドルが付いた環境である．

状態

時刻 t における状態 s_t は $s_t = \{\boldsymbol{p}_t^{hand}, \boldsymbol{q}_t^{hand}, \boldsymbol{j}_t^{finger}, \boldsymbol{p}_t^{finger}, \boldsymbol{h}_t^{finger}, \theta_{app}^{zenith}, \theta_{app}^{azimuth}, t\}$ とする．ここで，$\boldsymbol{p}_t^{finger}$ は，active-force closure では時刻 t における指先位置である．Passive-force closure では手のひらの位置も含まれる．Lazy-closure では，親指は関係ないため親指以外の指先位置を含む．$\boldsymbol{q}_t^{hand} \in \mathbb{R}^4$ は手の姿勢，h_t^{finger} は各指の力覚であり，0 か 1 の値で表される．$h_t^{finger} = 1$ の時が指に物体が衝突している時である．$\boldsymbol{p}_t^{finger}, \boldsymbol{q}_t^{hand}$ は初期ハンド座標系で表現される．これは物体とロボットの位置姿勢に依存しない座標系だからである．$\theta_{app}^{zenith}, \theta_{app}^{azimuth}$ はアプローチ方向を図 4.8 のように物体中心の球面座標系で表示したものである．

行動

行動 a_t は $a_t = \{\Delta \boldsymbol{j}_t^{finger}, \Delta u_t, \Delta v_t, \Delta w_t, \Delta q_t\}$ で定義される．本手法では事前に収集された人間の実演から生成された参照動作との差分を行動としている．Δj_t^{finger} は各指の

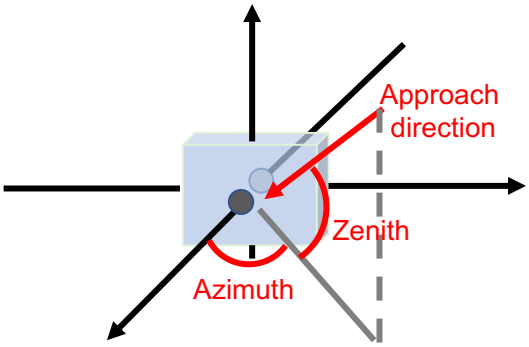


図 4.8　アプローチ方向の角度表示．図のように物体を中心とした球面座標系で，アプローチ方向を天頂角 (zenith) と方位角 (azimuth) で表現する．

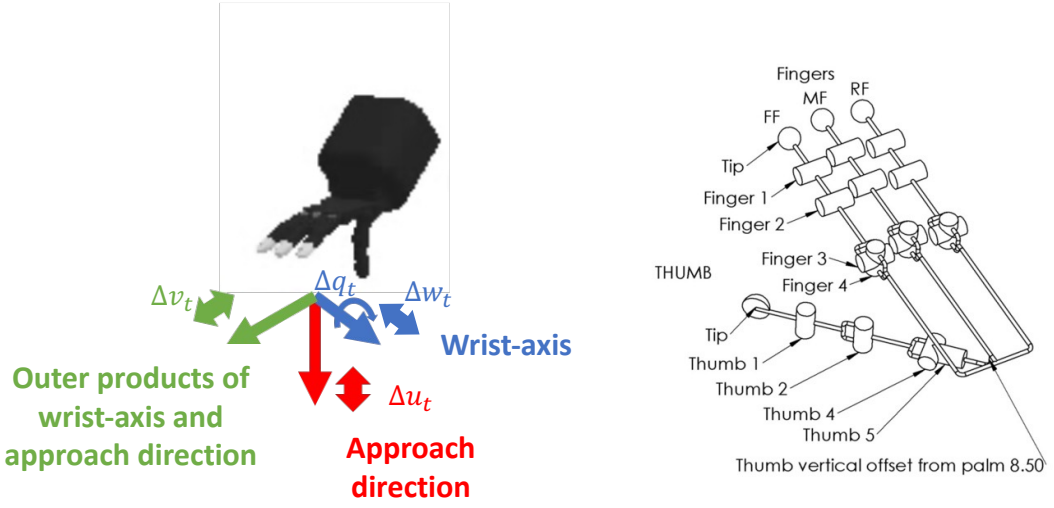


図 4.9　左は手の位置姿勢に関する行動を表している．赤矢印がアプローチ方向，青矢印が手首軸方向，緑矢印が外積方向である．右は本研究で用いるロボットハンドの構造である．

関節角度の差分，$\Delta u_t \in \mathbb{R}$ は手の位置のアプローチ方向成分の差分，$\Delta v_t \in \mathbb{R}$ は手の位置のアプローチ方向と手首軸との外積方向成分の差分，$\Delta w_t \in \mathbb{R}$ は手の位置の手首軸方向成分の差分，$\Delta q_t \in \mathbb{R}$ は手の姿勢の手首軸周りの回転の差分である ((図 4.9) 左)．

また，各指の探索空間を狭めるため，本研究で使用するロボットハンド (図 4.9 右) において，人差し指・中指・薬指の三本の指を連動させる．$\Delta \boldsymbol{j}_t^{index} \in \mathbb{R}^3$，$\Delta \boldsymbol{j}_t^{middle} \in \mathbb{R}^3, \Delta \boldsymbol{j}_t^{ring} \in \mathbb{R}^3$ とすると，$\Delta \boldsymbol{j}_t^{middle} = \Delta \boldsymbol{j}_t^{ring} = \Delta \boldsymbol{j}_t^{index}$ となる．これは，Virtual Finger [156] に基づいている．Virtual Finger とは，ある把持において同じ機能を持つ複数の指をまとめて同一の指 (Virtual Finger) として考えるという概念である．また，人差し指と薬指は中指に対してほぼ対称の位置に存在すると考え，人差し指の外転の角度が θ である時に，中指の関節角度は 0，薬指の関節角度は $-\theta$ とした．この関節角度は安定把持の形成に影響しないため学習は行わずに常に固定値とした．

報酬

報酬設計は異なる force-exertion type で共通の設計をしており，closure 理論 [95] を基に設計されている．報酬 r_t は以下のように，代表的な接触点ごとに与えられる報酬 r^{ctt} と，後続スキルの成否を促すための報酬 r^{post} の和，早期終了時に与えられる報酬 r^{term} で計算される．

$$r_t = r^{ctt} + r^{post} + r^{term} \tag{4.4}$$

r^{ctt} は以下のように計算される．

$$r^{ctt} = \sum_{i=1}^{N} \{\log(b^p - k_1 d^p_{i,t}) + \log(b^f - k_2 d^f_{i,t})\} \tag{4.5}$$

ここで，

$$d^p_{i,t} = \|\boldsymbol{p}_{i,t} - \boldsymbol{c}_i\|,$$
$$d^f_{i,t} = \arccos(\boldsymbol{f}_{i,t} \cdot \boldsymbol{n}_i).$$

である．N は代表的な接触点の数である．b^p, b^f は事前に定義された上限，k_1, k_2 は係数である．第一項は i 番目の指先の理想的な接触点位置 $\boldsymbol{c}_i$ と実際の指先位置 $\boldsymbol{p}_{i,t}$ の誤差に関する項であり，第二項は理想的な接触力方向 $\boldsymbol{n}_i$ と実際の方向 $\boldsymbol{f}_{i,t}$ の誤差に関する項である (図 4.10)．これらの項によって各 force-exertion type における理想的な接触点位置と力方向が満たされるような把持が学習される．

r^{post} は，active-force closure，passive-force closure では，最終時刻以外においては 0 であり，持ち上げた際に物体の位置姿勢がそれぞれ閾値 $\Theta^{pos}, \Theta^{rot}$ よりずれていなければ 1，失敗した場合に -1 が与えられる．Lazy-closure ではドアの角度 θ^{door} に応じた以下のような報酬 r^{post}_{lazy} が与えられる．これは，ハンドルに対して余分な力をかけないようにするためである．余分な力がかかっている場合，ドア開けを行う際にハンドルを破壊してしまう危険性がある．

$$r^{post}_{lazy} = \log(b^{door} - k_3 \theta^{door}) \tag{4.6}$$

早期終了は局所解に陥ることや実機実行時に危険性がある状態に到達することを防ぐために行う．前者に関しては，どの把持でも共通で，指定した時刻になっても物体と指が接触していない場合に早期終了させる．この場合には r^{term} に -100 が加算される．後者に関しては，二種類ある．一つ目はどの把持でも共通で，物体と指が接触した後に指が物体から離れた場合に早期終了させる．これにより物体が手から離れて把持ができない位置まで移動してしまうことを防ぐ．この場合には，r^{term} に -40 が加算される．二つ目は active-force closure や passive-force closure で使用するもので，手が物体を押し付けて環境を破壊してしまうことを防ぐために物体から机にかかる力の大きさが閾値 Θ^F を超えた場合に早期終了させる．この場合には，r^{term} に -10 が加算される．

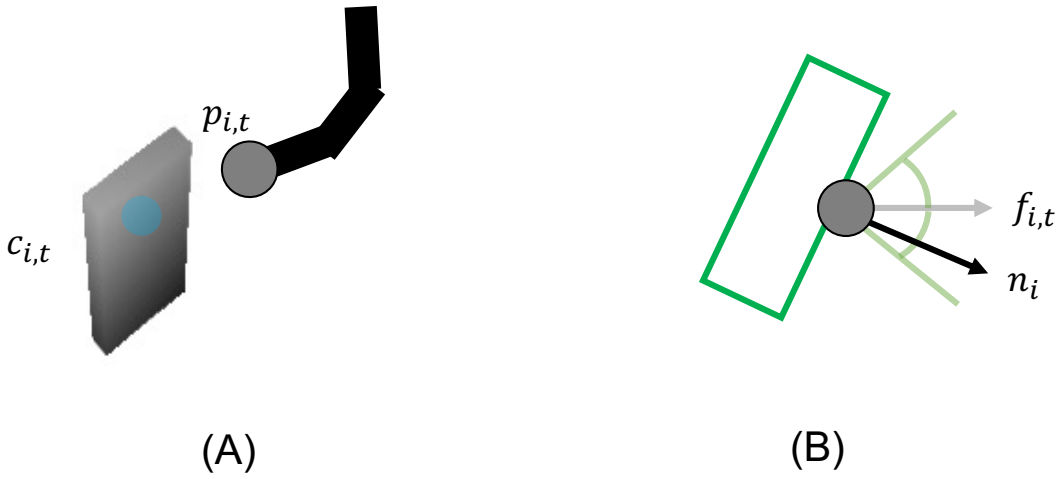


図 4.10　報酬 r^{ctt} の概要．(A) は第一項を表しており，i 番目の指先の理想的な接触点位置 $\boldsymbol{c}_i$ と実際の指先位置 $\boldsymbol{p}_{i,t}$ を近づける役割を持つ．(B) は第二項を表しており，理想的な接触力方向 $\boldsymbol{n}_i$ と実際の方向 $\boldsymbol{f}_{i,t}$ を近づける役割を持つ．

4.3.3　Domain randomization

対象物体は以下の式で表される Superquadrics [157] によってパラメータ化される．a_1，a_2，a_3 は物体の x 軸方向の大きさ，y 軸方向の大きさ，z 軸方向の大きさを表している．ϵ_1，ϵ_2 は xz 平面における形状，xy 平面における形状を表している．

$$\left(\left(\frac{x}{a_1}\right)^{\frac{2}{\epsilon_2}} + \left(\frac{y}{a_2}\right)^{\frac{2}{\epsilon_2}}\right)^{\frac{\epsilon_2}{\epsilon_1}} + \left(\frac{z}{a_3}\right)^{\frac{2}{\epsilon_1}} = 1 \tag{4.7}$$

様々な形状の物体に対する把持を学習するために，学習時には形状パラメータをランダムに変化させる．物体の大きさと物体上面の形状は YCB 物体データセット [158] の範囲内でランダムに変更される．また，接触点群の推定誤差として，シミュレータ上で並進と回転方向に対して一様分布からサンプリングされたノイズを付加する．さらに，実行時に関節角度が想定の値に収束しないことがあり得るため，関節角度指令値にも同様に一様分布からサンプリングされたノイズを付加する．アプローチ方向に関しても同様に事前に定義された範囲内でランダムに変化させる．

4.4　接触点群認識

物体の写った画像を $\boldsymbol{I}$，把持位置姿勢を $\boldsymbol{O}$ とする．物体の写った画像から接触点群とその点群が属する座標系の姿勢を出力する関数 $G : \boldsymbol{I} \rightarrow \boldsymbol{O}$ を設計する．この関数を手作業で設計することは困難であると考えられる．そこで，本研究では深層ニューラルネットワークを用いて関数近似をすることで所望の関数を得る．以下ではネットワークの詳細やデータセットの作成方法に関して説明する．また，強化学習における randomization の範囲を

決めるために，接触点群認識の学習結果を示す．

4.4.1 入出力のデータ形式

本節では深層ニューラルネットワークの入出力のデータ形式に関して説明する．本研究では入力として物体の写った画像を使用している．今回用いるロボットのカメラから得られる画像として RGB 画像と深層画像の二つが得られる．RGB 画像を用いた場合，正解ラベルの付け方によってはテクスチャに依存した推定になってしまう可能性がある．そこでテクスチャに依存しない推定をするために入力には深度画像を用いる．

出力は接触点群とその点群が属する座標系の姿勢である．接触点群と姿勢はカメラ座標系で表現されている．姿勢を表す方法は複数存在する．代表的な方法としてはオイラー角表現とクォータニオンが挙げられる．

オイラー角表現は三つの直交する回転軸とその回転軸に対する回転角で表現される．オイラー角は直感的に理解しやすいという利点がある．角度は一般的に $[0, 2\pi)$ の区間で表現されるが，0 と 2π は回転を表すものとしては等価な表現である．そのため，数値上は非連続点となってしまう．この非連続性は深層学習において非常に問題である．また，等価な回転を表現する組み合わせが複数個あり解の一意性を満たせないという点も深層学習において問題となる．

クォータニオンは複素数を拡張した数学系であり，一般に四つの実数 $x, y, z, w \in \mathbb{R}$ を用いて $xi + yj + zk + w$ として表現される．i, j, k はクォータニオンの単位を表す．クォータニオンは連続性を満たすが，クォータニオン $\boldsymbol{q}$ と $-\boldsymbol{q}$ は等価な回転を表現するため一意性を満たすことができない．しかしながら，等価な回転は高々二つであるためそのクォータニオンに近い方を選択するという操作をすることで解決することが可能である．以上の理由から出力である手の姿勢の表現にはクォータニオンを選択した．

したがって，関数の入力 $\boldsymbol{I}$ はロボット視点深度画像，出力 $\boldsymbol{O}$ はカメラ座標系での接触点群 $\boldsymbol{p} \in \mathbb{R}^{3\times n}$，姿勢 $\boldsymbol{q} \in \mathbb{R}^4$ となる．ここで n は force-exertion type ごとに事前に定義された接触点数である．深度画像は縦横 224 ピクセルに設定したため，$\boldsymbol{I} \in \mathbb{R}^{224\times 224\times 1}$ である．

4.4.2 ネットワーク構造

本研究では画像処理に特化した深層ニューラルネットワークである畳み込みニューラルネットワーク (CNN) を用いて把持位置推定を行う．ネットワーク構造は四層の CNN と三層の前結合層である (図 4.11)．カーネルサイズが $16 \times 3 \times 3$，$32 \times 3 \times 3$，$64 \times 3 \times 3$，$128 \times 3 \times 3$ の畳み込み層の後に，出力が $1024, 256, 3n + 4$ の全結合層となっている．畳み込み層の後には 2×2 の MaxPooling が適用される．活性化関数は最終層以外は ReLU を用いる．出力は $(-\infty, \infty)$ の区間を取り得るので最終層には線形関数を用いる．

学習の際に用いた損失関数 L は式 4.8 で表される．L は指先位置 $\boldsymbol{p}$ の誤差に関する $L2$ ノルムと手の姿勢 $\boldsymbol{q}$ の誤差に関する $L2$ ノルムの和である．この損失関数は [159] を参考に設計している．[159] の損失関数では $\boldsymbol{q}$ の誤差に関する $L2$ ノルムが $\|\hat{\boldsymbol{q}} - \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|}\|_2^2$ しか考慮

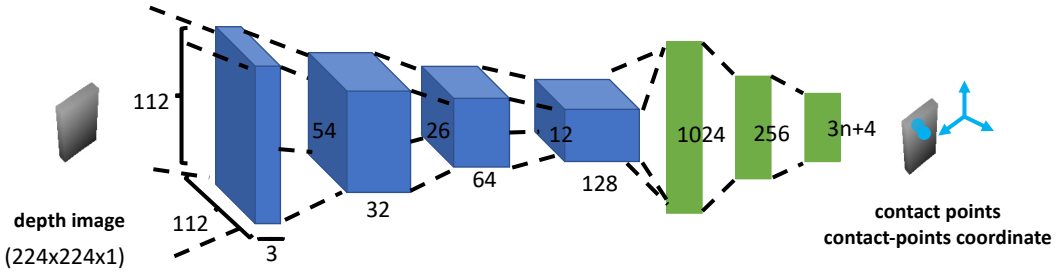


図 4.11　推定に用いるネットワークの構造.

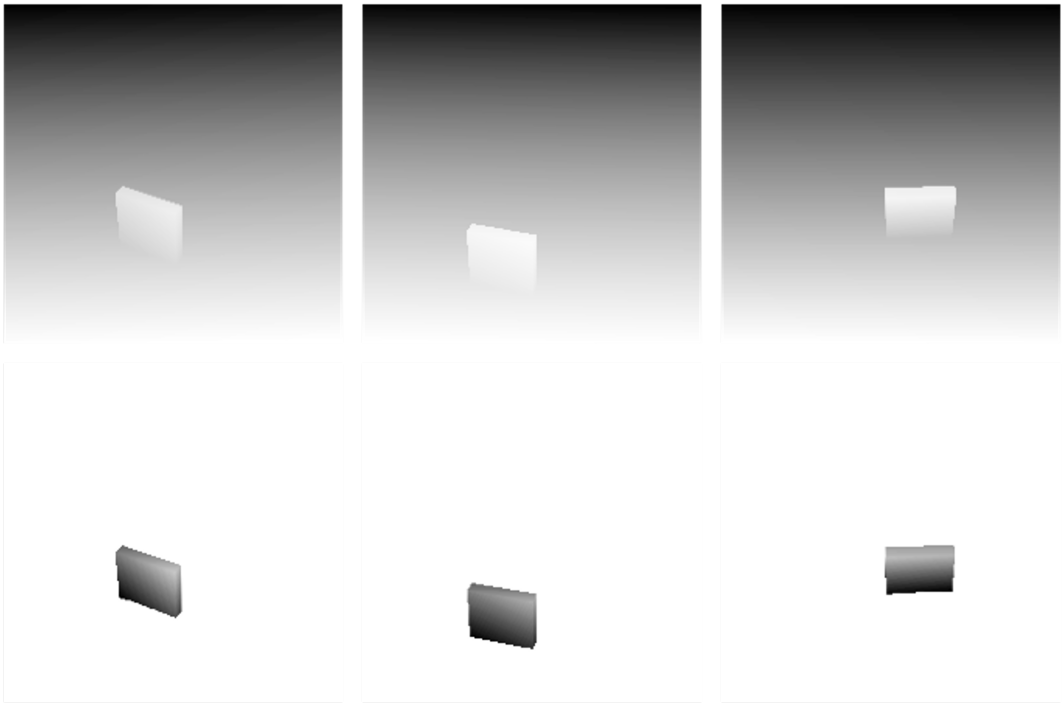


図 4.12　シミュレータ上で収集された深度画像．上段が平面除去前，下段が平面除去後の深度画像.

されていなかった．これはクォータニオン $\boldsymbol{q}$ と $-\boldsymbol{q}$ が同じ回転を表現するという解の一意性を満たさない性質を無視している．本手法では $\|\hat{q} - \frac{q}{\|q\|}\|_2^2$ と $\|\hat{\boldsymbol{q}} + \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|}\|_2^2$ の二つのうち小さい方を選択するようことでクォータニオンが解の一意性を満たさない性質に対処する．

$$L = \|\hat{\boldsymbol{p}} - \boldsymbol{p}\|_2^2 + min(\|\hat{\boldsymbol{q}} - \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|}\|_2^2, \|\hat{\boldsymbol{q}} + \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|}\|_2^2) \tag{4.8}$$

4.4.3　データセット

深度画像データセットを現実で収集することは非常に手間がかかる．そこで物理シミュレータである PyBullet [160] を用いて収集する．机上に対象物体のみが置かれている環境で深度画像が収集される．図 4.12 の上段が収集された深度画像の例である．カメラから物体までの距離やカメラの姿勢はランダム化されている．

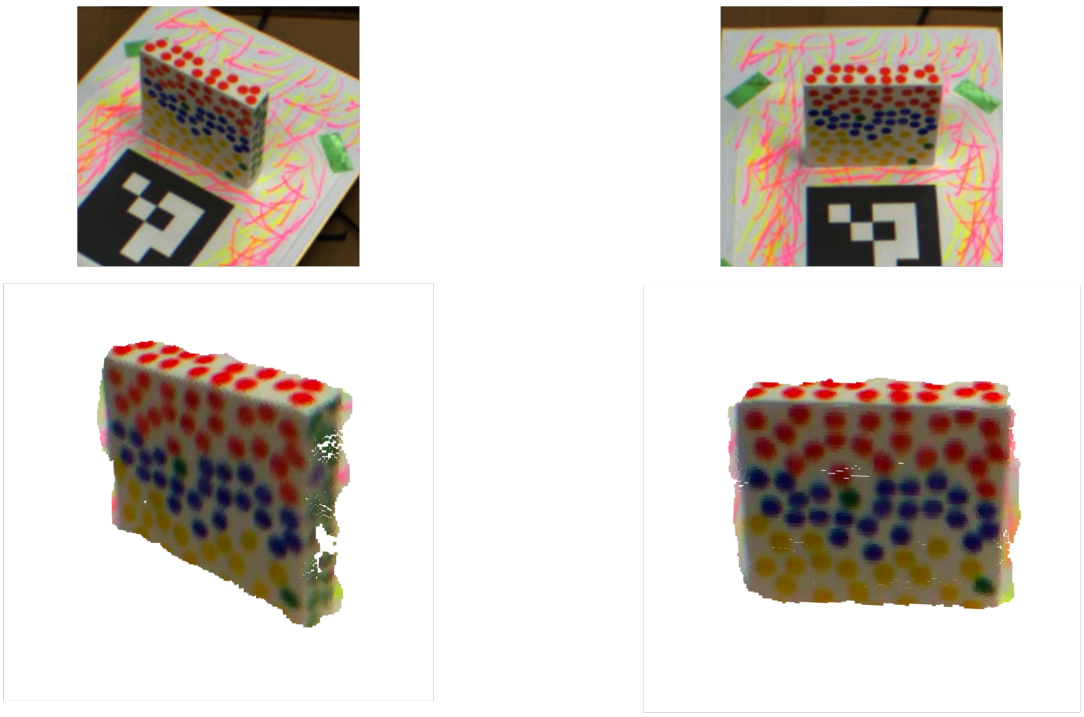


図 4.13　ステレオカメラで取得された深度画像から生成された点群．物体のエッジ部分が後景と繋がるまたは欠落している．

学習時はシミュレータで収集された深度画像を用いるが，実機での実行時は実画像を用いることになる．この場合に，シミュレーション画像と実画像との間のドメインギャップが推定に悪影響を与える可能性がある．そこで，ギャップを減らすために対象物体以外のピクセルを全て 0 で埋める前処理を行う．前処理では深度画像を点群に変換後，Random Sampling Consensus(RANSAC) [161] を用いて平面除去を行う．実機で深度画像を得る際にはステレオカメラを用いるが，ステレオカメラで得られる深度画像はその性質上，エッジ付近にノイズが多く発生する (図 4.13)．物体のエッジ付近が後景と繋がることで物体のサイズが少し大きくなることや物体のエッジ付近の欠落が起こる．そのため，エッジからマンハッタン距離が N_1 以内のピクセルを削る．その後，エッジからマンハッタン距離が N_2 以内のピクセルをランダムで 0 にする．

正解ラベルはカメラ座標系での接触点の位置と座標系の姿勢 (クォータニオン) である．

4.4.4　接触点群認識の評価

強化学習における認識誤差の randomization の範囲を決めるために，接触点群認識の結果を評価する．実行時における把持対象物体が常に学習時に用いた物体と厳密に同じ大きさ・形状の物体であるとは限らない．そこで本節では，事前に定義した大きさ・形状範囲から適当にサンプリングした物体のみで学習させた際に，学習時の物体と学習時とは大きさ・形状が異なる物体に対する推定精度が同等になるのかを調査する．すなわち，学習時の物体と学習時とは大きさ・形状が異なる物体での把持位置姿勢推定誤差を比較することで，把持位置推定のデータセットの内挿に対する汎用性を調査する．

実験条件

本評価では，active-force closure で把持する場合の接触点群認識に関して評価する．本実験では様々な把持プリミティブに対する学習において使用可能となるような妥当な精度を求めたいため，全ての指に対して推定精度の評価を行う．なお，実行時に用いられる指は一部の指である．$a_1 = 0.01, a_2 = 0.05, a_3 = 0.05, \epsilon_1 \simeq 0, \epsilon_2 \simeq 0$ 付近の $0.01 \leq a_1 \leq 0.05, 0.06 \leq a_2 \leq 0.20, 0.06 \leq a_3 \leq 0.16, \epsilon_1 \simeq 0, 0 < \epsilon_2 \leq 2$ の範囲の物体で視覚システムの学習・評価を行なった．ϵ_1 によって形状の変化した物体や $\epsilon_2 > 2$ の凸包ではない形状の物体は上からの精密把持の対象外としている．$a_1, a_2, a_3, \epsilon_2$ の範囲の端点と中点の組み合わせの物体 $81(= 3^4)$ 個を用いて学習を行なった．物体からカメラまでの距離やカメラの姿勢がランダム化されて収集された約 30 万枚の深度画像を使って学習された．バリデーションデータには 2.5 万枚の深度画像が含まれている．評価では a_1 を 0.01 ずつ動かした値，a_2, a_3 を 0.02 ずつ動かした値，ϵ_2 を 0.5 ずつ動かした値の組み合わせの物体と学習時に用いた物体の 1233 個の物体を用いた．各物体に対して深度画像が 63 枚取得された．図 4.14 は学習時の物体と評価時の物体のサンプリングを表現した図である．左図のように学習時は各軸 $(a_1, a_2, a_3, \epsilon_2)$ の端点と中点の組み合わせの物体 81 個 $(= 3^4)$ をサンプリングし，評価時は各軸に対してより細かく刻んだ値の組み合わせの物体をサンプリングした．評価時に関しては $a_1 \in \{0.01, 0.02, 0.03, 0.04, 0.05\}$，$a_2 \in \{0.06, 0.08, 0.1, 0.12, 0.14, 0.16\}$，$a_3 \in \{0.06, 0.08, 0.1, 0.12, 0.14, 0.16, 0.18, 0.2\}$，$\epsilon_2 \in \{\simeq 0, 0.5, 1.0, 1.5, 2.0\}$ として，この組み合わせの物体 1200 個をサンプリングした．

また，active-force closure においては，真値の xy 座標は物体座標系で常に同じであるが，z 座標に関しては物体上面から把持位置までの距離が常に等しくなるようにスケーリングされた．これは実行時に物体を押しつぶすことを防ぐためである．

深度画像の前処理として行う平面除去のパラメータ $d_{plane} = 1cm$ と設定した．また，エッジからマンハッタン距離が $N_1 = 5$ 以内のピクセルを削った．その後，エッジからマンハッタン距離が $N_2 = 2$ 以内のピクセルをランダムで 0 にする．

ネットワークの重みの初期化は ReLU と相性が良い He の初期化 [162] を用いた．また，最適化アルゴリズムには Adam [163] を用いた．さらに，過学習を防ぐために validation の精度が一定期間上がらなくなったら学習を打ち切る Early Stopping を行なった．15epoch の間にバリデーションデータの精度が改善されなければ学習を打ち切ることにした．学習率は 5×10^{-3} で Batch Size は 256 に設定した．また入力画像のピクセル $depth_{i,j}$ はデータセットに含まれる画像の全ピクセルの平均 m と標準偏差 σ を用いて標準化されている (式 4.9)．

$$depth'_{i,j} = \frac{depth_{i,j} - m}{\sigma} \tag{4.9}$$

結果

学習時に用いた物体群と学習時に用いていない物体群で評価した結果が表 4.1 である．評価指標として位置誤差には平均絶対誤差 $\|\hat{\boldsymbol{p}} - \boldsymbol{p}\|$ を用いた．姿勢誤差はクォータニオンの差 $\hat{\boldsymbol{q}}\boldsymbol{q}^{-1}$ に対して yaw-pitch-roll の絶対値の平均値を求めた．どちらの群の精度も同程

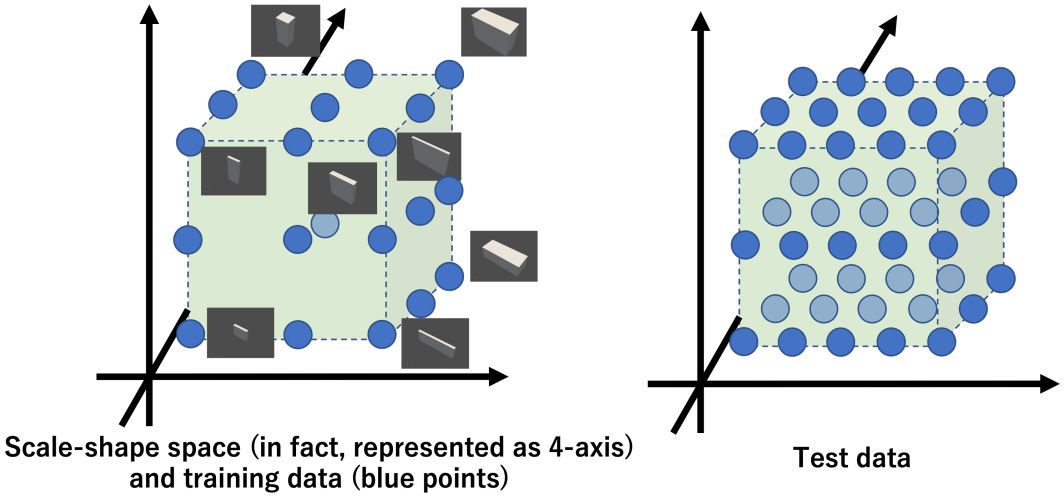


図 4.14　学習時と評価時の物体のサンプリングを表した図．緑の範囲が想定している大きさ・形状範囲で青丸がサンプリングした物体を表している．

度の精度であった．この結果からデータセットの内挿が推定できていることが分かる．

図 4.5 は学習に用いていない物体群での推定例である．青丸が真の把持位置で，赤丸が推定した把持位置である．青丸と赤丸がほぼ同じ位置にあり，未知の高さの物体であっても把持位置の z 座標が上手くスケーリングできている．

深層ニューラルネットワークはデータ分布を学習できるため，データ分布に含まれるデータに対しても推定することが可能であったのだと考えられる．すなわち，事前に定義した大きさ・形状範囲から適当にサンプリングした物体で学習させた場合に，その大きさ・形状範囲に含まれる全ての物体に対して同等の性能で推定を行うことが可能であったのだと考えられる．また，推定誤差に関しては指先位置が数 mm 程度，手の姿勢が 1° 程度であり，これは強化学習により十分に対処可能な範囲であると考えられる．

4.5　実行のための LfO への組み込み

本節では，上述した強化学習されたスキルと接触点群認識器がどのように LfO に組み込まれ，どのように後続スキルを考慮した把持が達成されるのかを説明する．このパイプラインは，把持実演と把持実行の 2 つのフェーズから構成される (図 4.16)．

人間による実演の段階では，人間が一度だけ実演を行い，その画像群と言語支持を取得する．これらの情報から認識パイプラインを用いて force-exertion type とアプローチ方向が認識される．認識パイプラインは既存研究 [83,86] を用いる．[83] では，把持の瞬間と対象物体の名前を認識することができ，またアプローチ方向も取得できる．[86] では，CNN ベースの分類器が恥の瞬間の手の画像と対象物体の名前に基づいて，用いられた人間の把持プリミティブが推定される．この把持プリミティブから前述したロボットの把持分類への対応付けを用いることで force-exertion type が取得できる．

表 4.1　学習時に用いた物体と用いていない物体での推定誤差の比較

(a) 指先位置の平均絶対誤差 (cm)

	学習時に用いた物体群	学習時に用いていない物体群
親指	0.242 ± 0.200	0.243 ± 0.192
人差し指	0.256 ± 0.206	0.260 ± 0.199
中指	0.248 ± 0.202	0.250 ± 0.196
薬指	0.252 ± 0.201	0.255 ± 0.198

(b) 座標姿勢の平均絶対誤差 (°)

	学習時に用いた物体群	学習時に用いていない物体群
yaw	0.938 ± 1.048	0.743 ± 0.905
pitch	0.450 ± 1.102	0.663 ± 0.704
roll	0.480 ± 0.466	0.249 ± 0.293

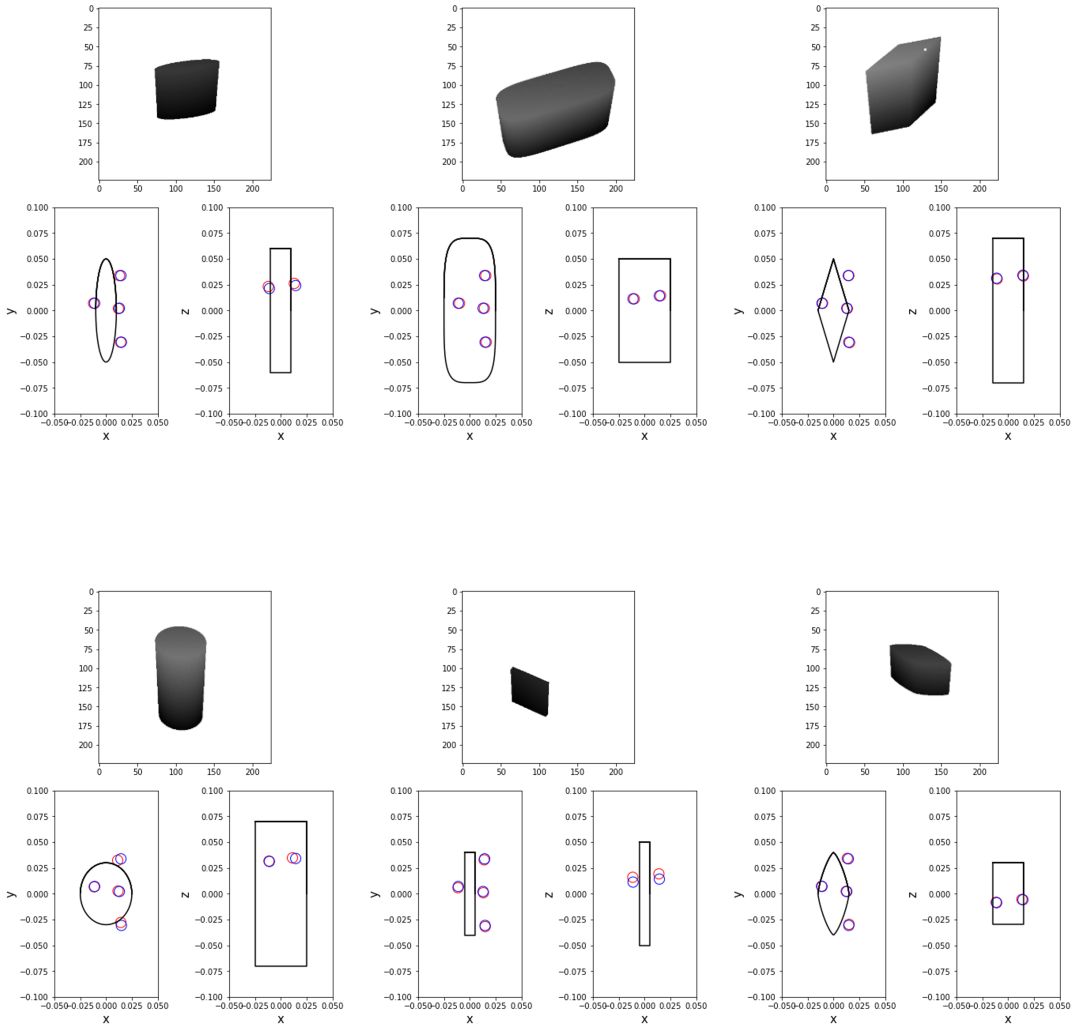


図 4.15　推定結果の例．青丸が真の把持位置，赤丸が推定された把持位置を表している．

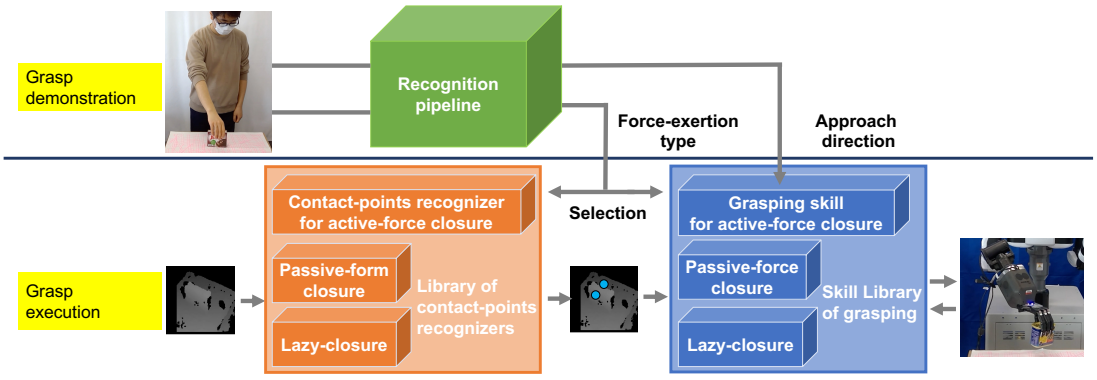


図 4.16　LfO と把持スキルを組み合わせた実行．人間の実演から得られた把持プリミティブに応じたスキルを選択する．そして実行時には人間の実演から得られたアプローチ方向を参考にして把持を実行する．

把持の実行の段階では，実演の段階で得られた force-exertion type に基づいて，接触点群認識器と強化学習スキルが選択される．なお，認識器とスキルは force-exertion type ごとに準備されている．認識器に入力される深度画像はロボットの頭部に装着されたステレオカメラから取得される．

ステレオカメラから得られた平行化後の画像を既存のステレオマッチング推定器 [164] を用いることで視差画像を取得し，その後視差画像をカメラパラメータを用いて深度画像へ変換する．深度画像は RANSAC を用いて平面除去が行われ，画像内に対象物体のみが写るようにされる．

実行中はスキルから出力された行動に従って手の位置姿勢や関節角度が決定される．手の位置姿勢は逆運動学を解くことによりロボットアームの関節角度に変換される．関節角度の可動域的に実現することができない位置姿勢であるような場合には逆運動学が解けない場合がある．逆運動学が解けない場合にはロボットの動作が停止する．

4.6　シミュレーション実験

4.6.1　準備

参照動作を時刻 $N = 6$ までとし，把持位置からアプローチ方向と逆向きに $n = 0.15$m 進んだ位置をロボットハンドの初期位置とした．接触点群認識の誤差に対処するために，スキル学習時にはロボットハンドの初期位置姿勢にノイズを乗せた．位置誤差では各軸に $[-0.5, 0.5]$cm のノイズを乗せた．姿勢誤差では各軸に $[-1.5, 1.5]^\circ$ のノイズを乗せた．また，実機においては関節角度の誤差があるため，出力された行動と等しい値の関節角度にならない場合もある．そこで，手の関節角度に $[-3, 3]^\circ$ の誤差を発生させた．物体の重さは 300g に固定された．物体の大きさ・形状とアプローチ方向の randomization は force-exertion type ごとに異なる．表 4.2 と 4.3 に物体の大きさ・形状とアプローチ方向の randomization の範囲を示す．物体の大きさ・形状に関しては，active-force closure で

表 4.2　物体の大きさ・形状の randomization の範囲.

	depth	width	height	ϵ_2
Active-force closure	$[0.01, 0.06]$m	$[0.06, 0.16]$m	$[0.06, 0.16]$m	$[0, 2]$
Passive-force closure	$[0.04, 0.06]$m	$[0.04, 0.06]$m	$[0.1, 0.16]$m	$[0, 2]$
Lazy-closure	$[0.02, 0.04]$m	$[0.02, 0.04]$m	0.1m	$[0, 2]$

表 4.3　アプローチ方向の randomization の範囲.

	azimuth	zenith
Active-force closure	$[-45, 45]^\circ$	$[30, 90]^\circ$
Passive-force closure	$[-30, 30]^\circ$	$[10, 30]^\circ$
Lazy-closure	$[0, 45]^\circ$	0°

は YCB object dataset [158] の Food items に含まれる物体の中でロボットハンドで操作できる大きさの物体を内包している．Passive-force closure，lazy-closure ではロボットの構造的に把持可能な範囲を用意した．アプローチ方向に関しては，各 force-exertion type にとってあり得る方向を用意した．

深層強化学習フレームワークとして Microsoft Project Bonsai[*1]を用いる．このフレームワークは Microsoft 社が提供している自律システム向け機械教示サービスである．いくつかの有名な強化学習アルゴリズムが実装されており，ユーザーは状態，行動，報酬の定義されたシミュレータを用意するだけで強化学習が可能である．強化学習アルゴリズムには Proximal Policy Optimization [165] を用いた．バッチサイズは 6000，学習率は 5×10^{-5} とした．スキルは，2 つの 256 次元隠れ層を持つ多層パーセプトロンによってパラメータ化されている．活性化関数には，[165] と同様に hyperbolic tangent($tanh$) を用いた．定期的に 30 エピソードを評価して，50 万イテレーションの間にエピソードの累積報酬和の平均値に改善がなければ学習を終了した．

本実験では，randomization の範囲が最も広く学習が難しい active-force closure に対するスキルの性能を評価する．active-force closure は複数の人間の把持プリミティブがマッピングされる先である．そのプリミティブには角柱によく用いられる prismatic-3 や prismatic-4 や，円柱によく用いられる circular-3 や circular-4 が含まれる．そのため，active-force closure のスキルには異なる形状の物体を把持できることが求められる．また，実世界の物体には様々な大きさの物体が存在するため，スキルは異なる大きさの物体を把持できることも求められる．そこで，本実験では一つのスキルで様々な大きさ・形状の物体を把持できるのかどうかを調査する．ϵ_2 を 0（直方体），1（楕円柱），2（上面が菱形の四角柱）に設定し，奥行きを $[2, 6]$cm，幅を $[6, 10]$cm の範囲で変化させた時の成功数を調査する．高さは 10cm に固定する．把持後に物体を持ち上げられた場合を成功と定義する．アプローチ方向の天頂角は 0°，方位角は 0° に固定する．

[*1] `https://www.microsoft.com/en-us/ai/autonomous-systems-project-bonsai`. 2024 年 10 月 24 日現在はサービスが終了している．

また，スキルは人間の実演から得られる様々なアプローチ方向に対処できる必要がある．そこで，アプローチ方向の変化に頑健であるかどうかも調査する．三つの物体に対して，天頂角を $[0, 60]°$，方位角を $[-45, 45]°$ の範囲で変化させた時の成功数を調査する．なお，アプローチ方向の変化への頑健性を調査したいため，認識誤差は発生させない．評価に用いる物体には，(A) 平均的な大きさ・形状の物体（奥行き 4cm，幅 10cm，$\epsilon_2 = 0$），(B) 把持が大きさ的に難しい物体（奥行き 4cm，幅 6cm，$\epsilon_2 = 0$），(C) 形状的に難しい物体（奥行き 4cm，幅 10cm，$\epsilon_2 = 2$）を用いた．

4.6.2 結果

物体の大きさ・形状への頑健性

図 4.17(A) は認識誤差のない場合の結果，(B) は認識誤差のある場合の結果である．認識誤差のない場合，物体の形状が変化しても把持することが可能であることが示された．認識誤差が加わると太い物体や曲率の高い物体の一部において失敗した．この結果から，様々な大きさ・形状の物体が把持できるスキルが学習できたものの，特に学習範囲の端に近い物体では認識誤差の影響で把持ができなくなってしまうことが分かった．そのため，より広範囲の物体に対して把持を行うためには，範囲をいくつかに分割して，各範囲に対してその範囲が得意なスキルを組み合わせる必要があることが分かる．

図 4.18 に把持の成功例を示す．(A) は倒れやすい薄い物体（奥行き 2cm，幅 6cm，$\epsilon_2 = 0$），(B) は把持しづらい厚い物体（奥行き 6cm，幅 10cm，$\epsilon_2 = 0$），(C) は楕円柱（奥行き 6cm，幅 10cm，$\epsilon_2 = 1$），(D) は上面が菱形の四角柱（奥行き 6cm，幅 10cm，$\epsilon_2 = 2$）での結果である．大きさや形状が異なっていても上手く把持できていることが確認できる．

アプローチ方向への頑健性

図 4.19 は異なる物体へのアプローチ方向の変化への頑健性の結果である．物体 (A) はすべてのアプローチ方向で把持に成功した．物体 (B) と (C) は，ほぼすべてのアプローチ方向で把持に成功したが，方位角の負の値が大きい場合は把持に失敗した．方位角に関する結果が非対称なのは，実験に使用したロボットハンドが右手であり，親指の付け根がロボットの人差し指の方に位置するという非対称な構造になっているためである．ロボットハンドの構造上，親指の接触位置への接触は負の方位角が大きい場合には困難であった．しかし，このロボットハンドを装着する右腕で把持を実行すれば，このようなアプローチ方向が起こることはほとんどないため，この失敗は無視できるものである．

図 4.20 に把持の成功例を示す．(A) は物体 (A) に対して天頂角と方位角が $30°, -30°$ のアプローチ方向で把持した際の結果である．(B) は物体 (B) に対して天頂角と方位角が $40°, 30°$ のアプローチ方向で把持した際の結果である．(C) は物体 (C) に対して天頂角と方位角が $60°, 30°$ のアプローチ方向で把持した際の結果である．アプローチ方向が変化しても手中で物体の姿勢を変化させて上手く把持できていることが分かる．

図 4.21 は失敗例である．(A) は物体 (A) に対して天頂角と方位角が $30°, -45°$ のアプローチ方向で把持した際の結果である．(B) は物体 (B) に対して天頂角と方位角が $60°, -45°$ のアプローチ方向で把持した際の結果である．どちらも物体の左右どちらかに

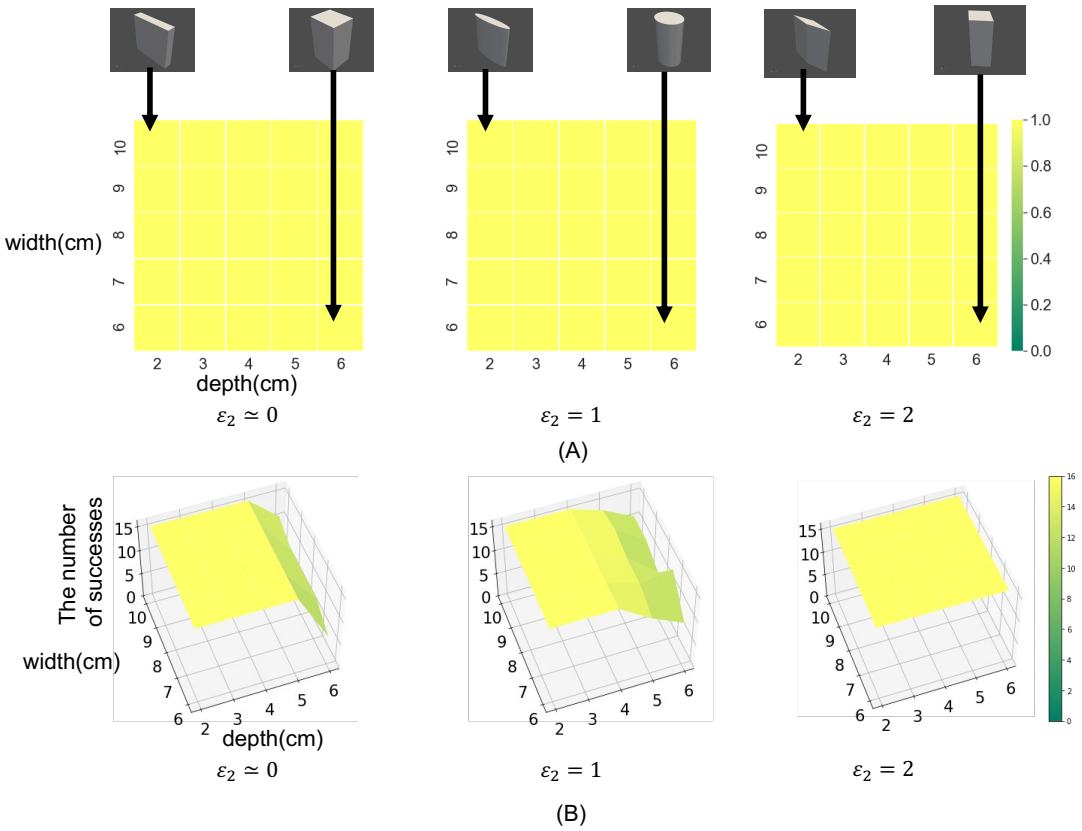


図 4.17　物体の大きさ・形状の変化への頑健性の評価．(A) は認識誤差のない場合の結果で黄色が成功，緑色が失敗を表す．(B) は認識誤差がある場合の結果である．

偏ってしまった結果，物体を横に押し出してしまって失敗してしまった．このような失敗は認識誤差が大きい場合にも同様に起こった．これに対処する方法の一つとしては，視覚的フィードバックを加えて手と物体の相対的な位置を基に適応的に行動するという方法が考えられる．

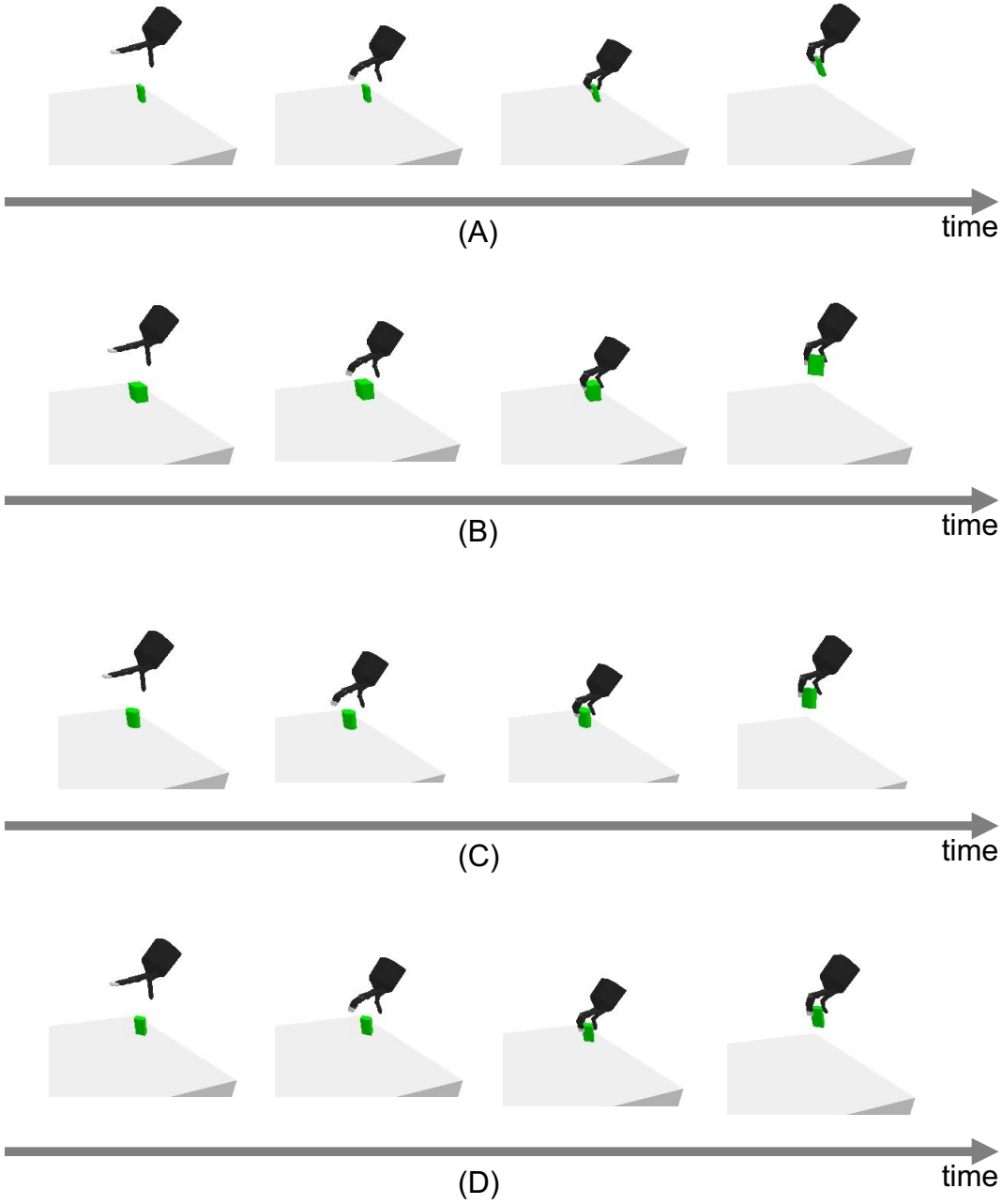


図 4.18　異なる物体での把持の結果例．(A) は倒れやすい薄い物体（奥行き 2cm，幅 6cm，$\epsilon_2 = 0$），(B) は把持しづらい厚い物体（奥行き 6cm，幅 10cm，$\epsilon_2 = 0$），(C) は楕円柱（奥行き 6cm，幅 10cm，$\epsilon_2 = 1$），(D) は上面が菱形の四角柱（奥行き 6cm，幅 10cm，$\epsilon_2 = 2$）での結果である．

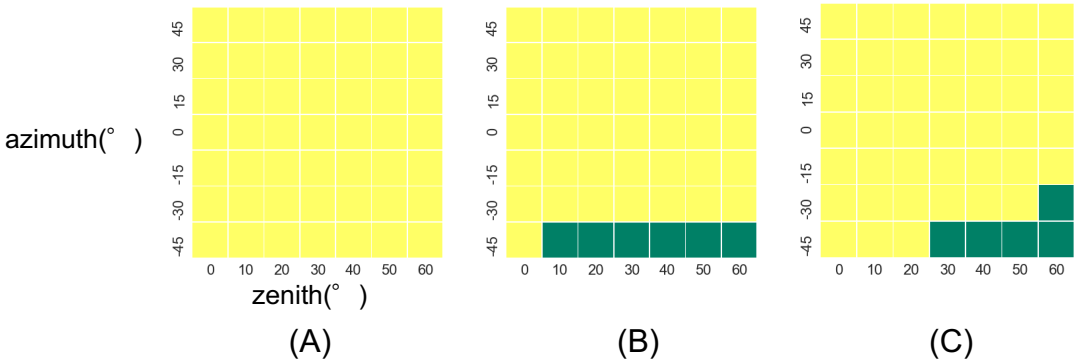


図 4.19　異なる物体へのアプローチ方向の変化への頑健性の評価．黄色が成功，緑色が失敗を表す．(A) は平均的な大きさ・形状の物体（奥行き 4cm，幅 10cm，$\epsilon_2 = 0$），(B) は把持が大きさ的に難しい物体（奥行き 4cm，幅 6cm，$\epsilon_2 = 0$），(C) は形状的に難しい物体（奥行き 4cm，幅 10cm，$\epsilon_2 = 2$）への結果である．

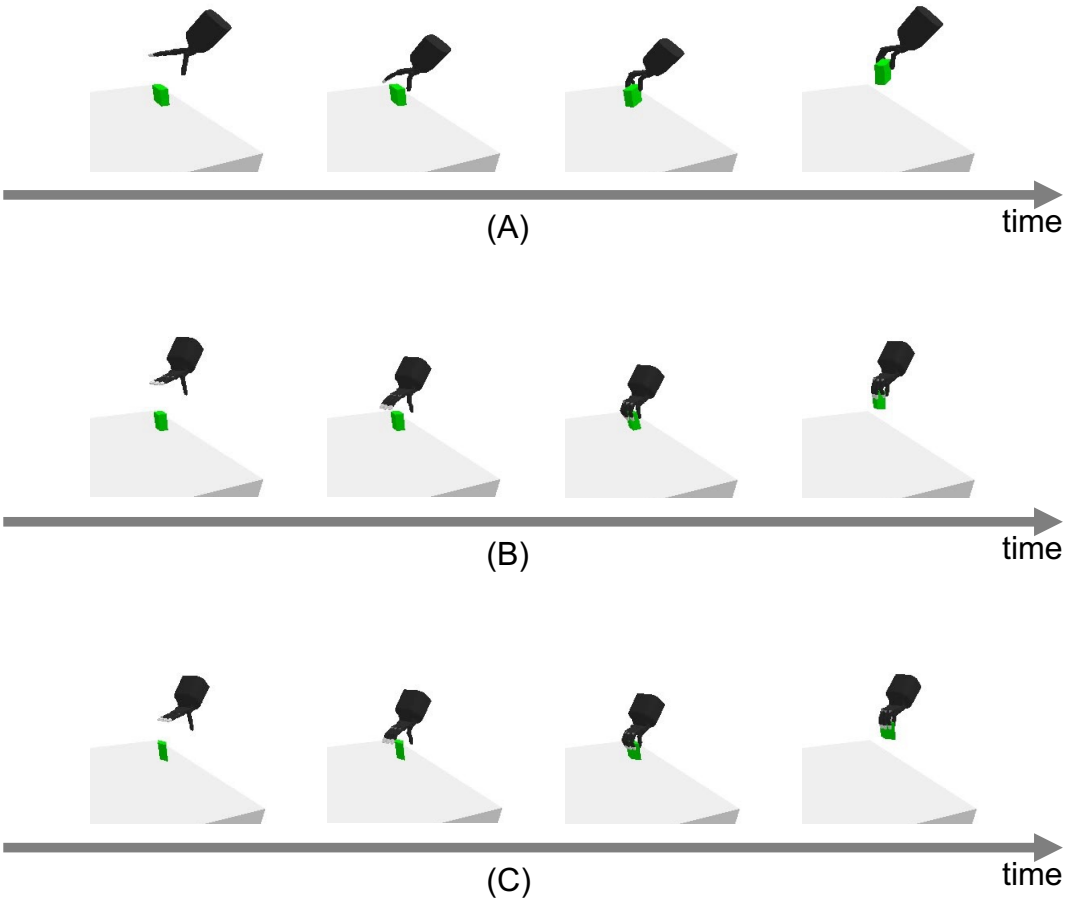


図 4.20　アプローチ方向を変化させた時の把持の結果例．(A) は物体 (A) に対して天頂角と方位角が $30°, -30°$ のアプローチ方向で把持した際の結果である．(B) は物体 (B) に対して天頂角と方位角が $40°, 30°$ のアプローチ方向で把持した際の結果である．(C) は物体 (C) に対して天頂角と方位角が $60°, 30°$ のアプローチ方向で把持した際の結果である．

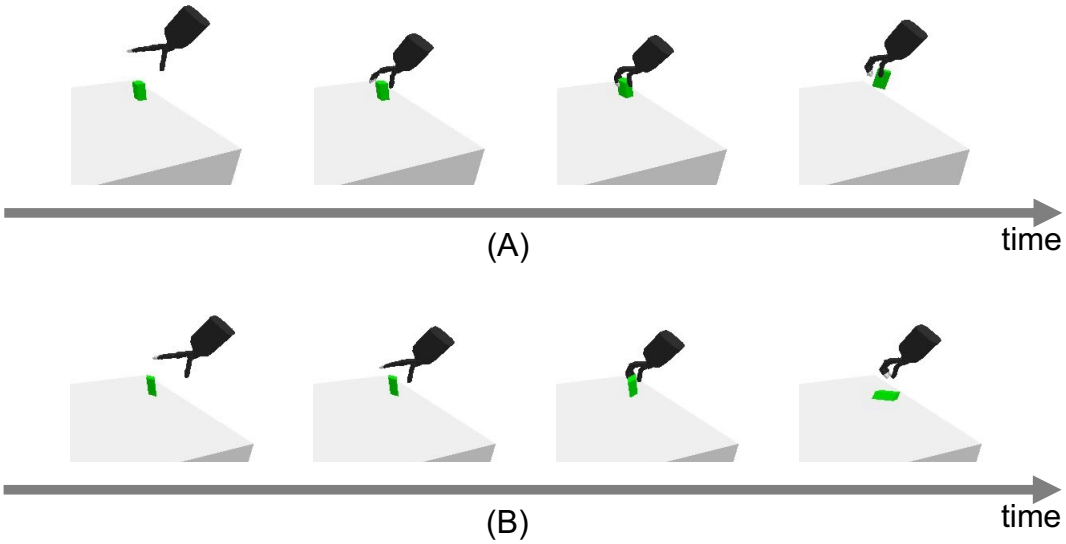


図 4.21　アプローチ方向を変化させた時の把持の結果例．(A) は物体 (A) に対して天頂角と方位角が $30°, -45°$ のアプローチ方向で把持した際の結果である．(B) は物体 (B) に対して天頂角と方位角が $60°, -45°$ のアプローチ方向で把持した際の結果である．

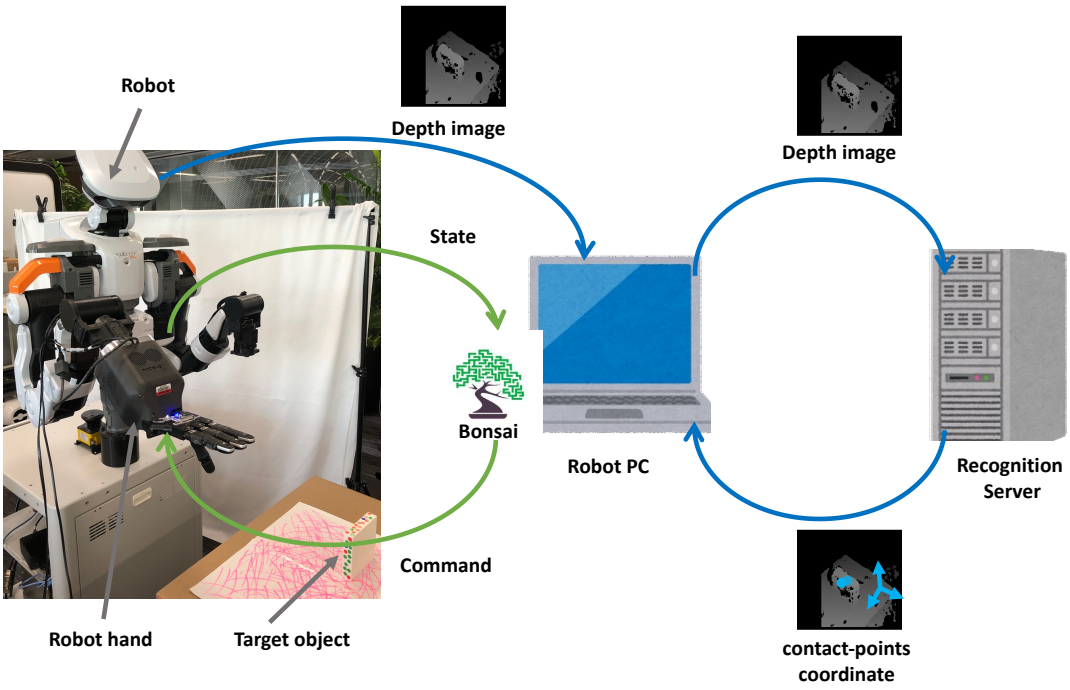


図 4.22　実機実験で用いたシステム全体像.

4.7　実機実験

4.7.1　準備

実機実験で用いたシステムの構成図を図 4.22 に示す．本システムは，ロボットハンドの位置姿勢や深度画像を取得するロボット，ロボットハンド，関節角度指令値をロボットへ送信するロボット PC，接触点群認識用のサーバーから構成されている．ロボットのカメラはステレオカメラである．ロボット PC では前節で学習したスキルを用いて状態から行動を決定する．状態から行動を決定する際には強化学習フレームワークの Microsoft Project Bonsai を用いている．そして，推定された行動から逆運動学を解くことでロボットアームの関節角度を計算する．逆運動学は Bio-IK [166] を用いて解く．その後，ロボットアームの関節角度とロボットハンドの関節角度をロボットへ送信する．ロボットにはカワダロボティクス株式会社の Nextage[*2]，ロボットハンドには Shadow Company の Shadow Dexterous Hand Lite[*3]を使用する．ロボットハンドの指先には力覚センサが装着されており，指先にかかっている圧力を測定できる．

実験では，まず active-force closure，passive-force closure，lazy-closure が追加の学習なしで実世界で動作するのかどうかを確認する．次に，作業の成功が評価しやすい active-force closure と passive-force closure のスキルを評価する．把持後に物体を持ち上げられ

[*2] https://nextage.kawadarobot.co.jp/
[*3] https://www.shadowrobot.com/dexterous-hand-series/

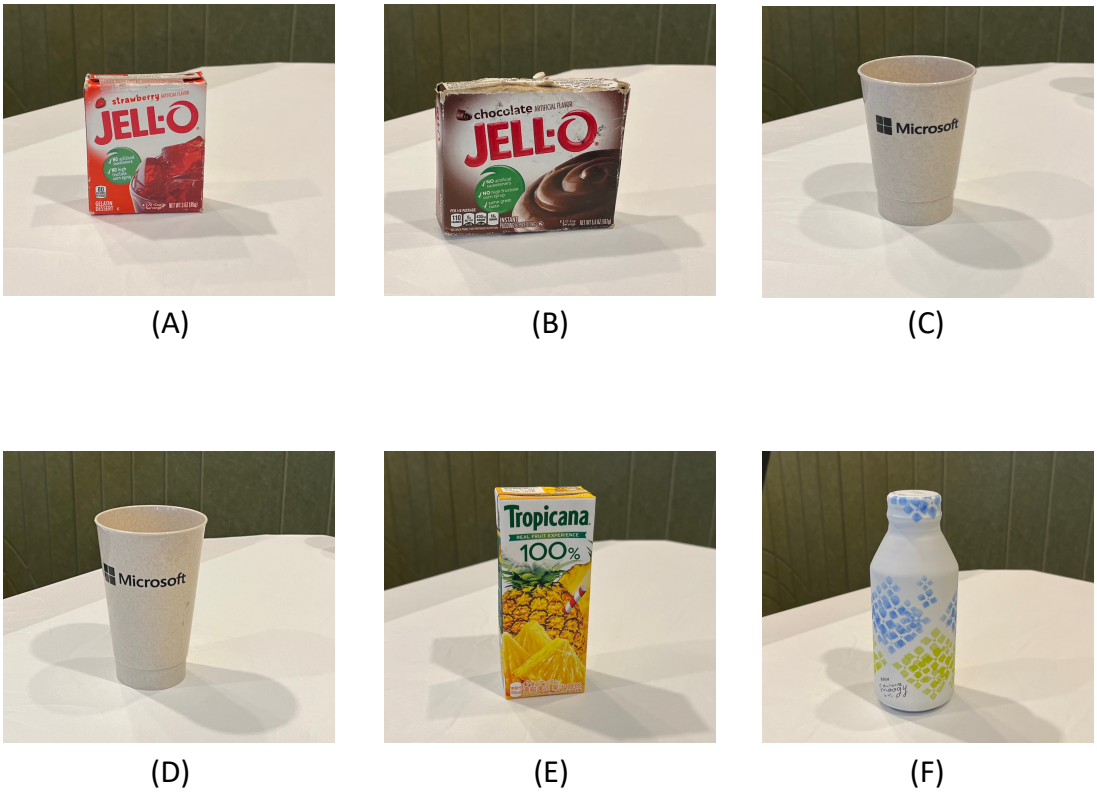


図 4.23　実機実験で使用する物体．物体 (A) は赤色の Jello の箱，物体 (B) は茶色の Jello の箱，物体 (C) は小さめのコップ，物体 (D) は大きめのコップ，物体 (E) はジュースのパック，物体 (F) はボトルである

た場合を成功と定義して，各物体に対して 5 回ずつ把持を行った際の成功数を数える．実験に使用する 6 種類の物体を図 4.23 に示す．物体の大きさ・形状は表 4.4 の通りである．物体 (A) は赤色の Jello の箱，物体 (B) は茶色の Jello の箱，物体 (C) は小さめのコップ，物体 (D) は大きめのコップ，物体 (E) はジュースのパック，物体 (F) はボトルである．物体 (A)，(B) は YCB object dataset に含まれる物体である．Active-force closure の実験では物体 (A)，(B)，(C) を用いる．Passive-force closure の実験では物体 (D)，(E)，(F) を用いる．評価時にはアプローチ方向を固定し，物体の位置姿勢をランダムに変化させた．

4.7.2　結果

図 4.24 に実機による把持の様子を示す．図 4.24-(A) がスパム缶に対する active-force closure, (B) がコップに対する passive-force closure, (C) がハンドルに対する lazy-closure の様子である．学習したスキルは追加の学習なしで実世界で動作することが確認できた．

表 4.5 と 4.6 に active-force closure，passive-force closure の実機での成功数を示す．どちらのスキルでも追加の学習なしで異なる大きさ・形状の物体を高い成功率で把持できていることが確認できた．一般的に，シミュレータと現実の間の誤差が生じるため，シミュレーションのみで学習させたスキルを実世界で動作させることは困難である．一方で，本

表 4.4　評価に用いた 6 個の物体の大きさ・形状．大きさの単位は cm である．(c)，(d) の大きさ・形状は上面のものである．(f) の大きさ・形状は底面のものである．

物体名	奥行き	幅	高さ	ϵ_2
(a)	2.5	7.0	8.5	0
(b)	3.2	11.0	8.7	0
(c)	7.5	7.5	10.0	1
(d)	8.5	8.5	13.0	1
(e)	3.7	5.3	13.0	0
(f)	6.5	6.5	16.5	1

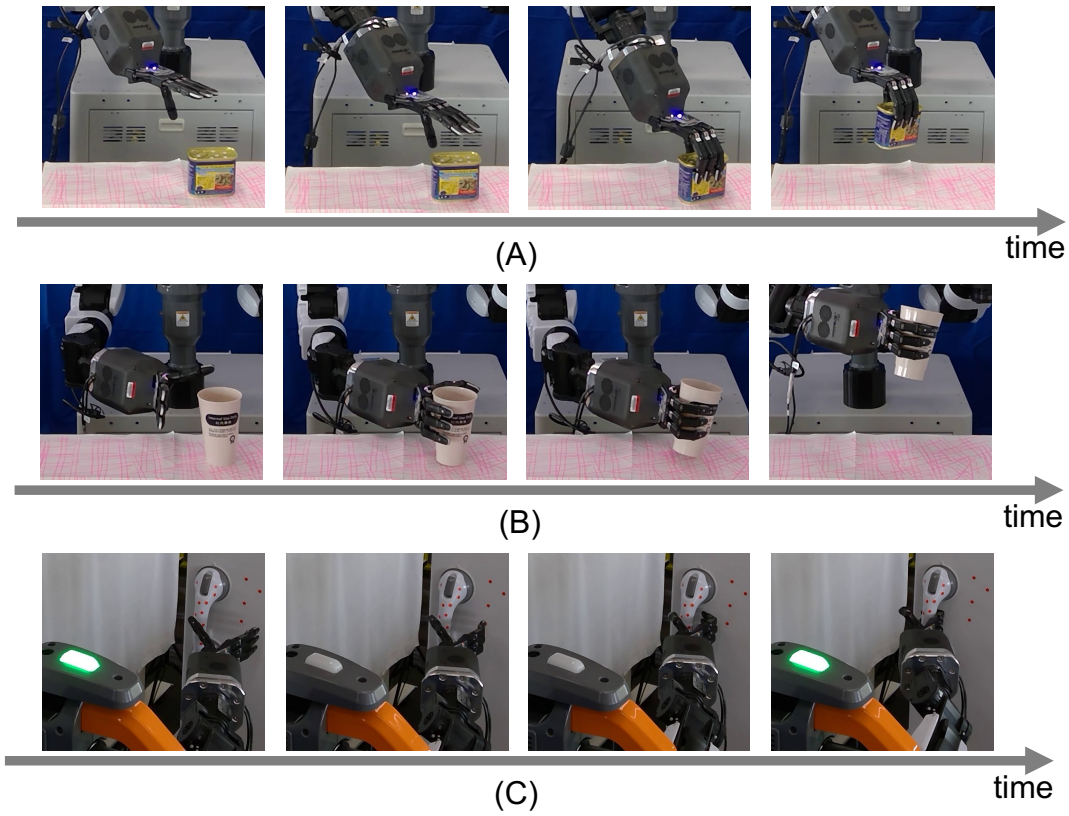


図 4.24　実機による把持の様子．(A) がスパム缶に対する active-force closure，(B) がコップに対する passive-force closure，(C) がハンドルに対する lazy-closure の様子である．

論文で提案したスキルは実世界でも動作することが分かった．これは，把持スキルの入力として指先位置，手の姿勢，物体と接触しているかどうかを表現する値を用いており，シミュレータと現実での誤差が小さく抑えられたことが理由だと考えられる．

図 4.25，4.26，4.27 に物体 (A)，(B)，(C) への active-force closure での把持の様子を示す．図 4.28，4.29，4.30 に物体 (A)，(B)，(C) への active-force closure での把持の様子を示す．どの物体でも位置姿勢が変化したとしても上手く把持できていることが分かる．

表 4.5　Active-force closure の実機での成功数.

物体 (A)	物体 (B)	物体 (C)
4/5	5/5	5/5

表 4.6　Passive-force closure の実機での成功数.

物体 (D)	物体 (E)	物体 (F)
5/5	5/5	5/5

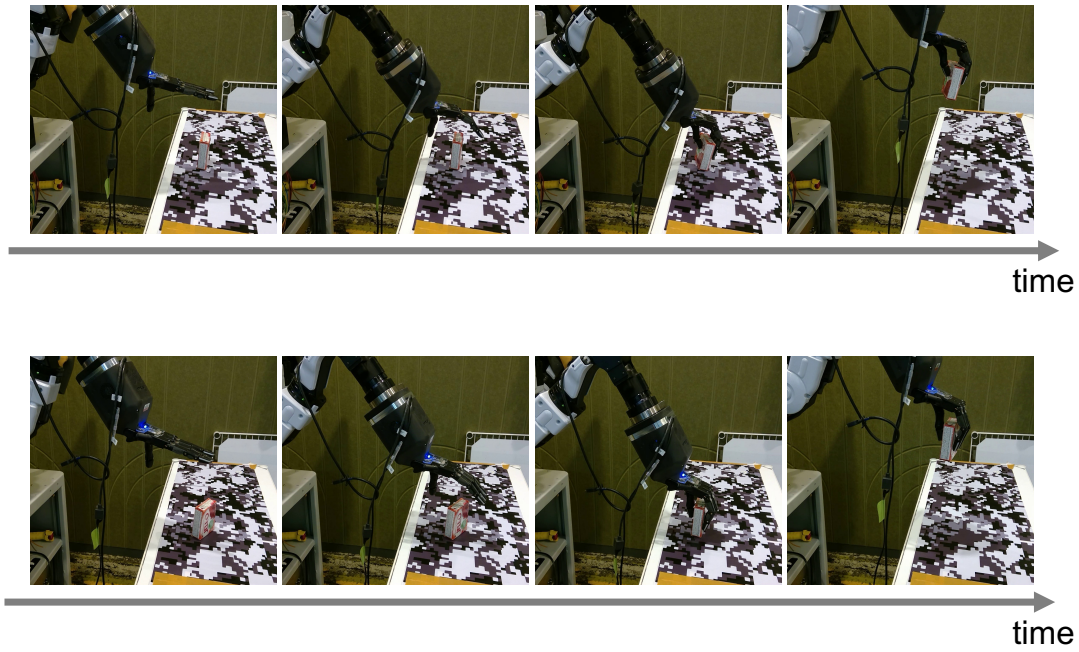


図 4.25　実機による物体 (A) に対する active-force closure での把持の様子.

図 4.31 に物体 (D),(E),(F) への passive-force closure での把持を上から見た様子を示す．学習したスキルは passive-force closure での特徴を満たした把持を形成していることが確認できる．物体 (D) の把持では，コップが変形してしまっていることが分かる．これは物体の大きさに対して関節が閉じすぎているためだと考えられる．これに対処する方法として，関節角度の指令値と実測値の差から物体の大きさを推定するような手法を追加するということが考えられる．また，物体の重さによって必要な把持力は変わるため，指先に摩擦力を測るセンサを装着すればより適切な把持力を物体にかけられるスキルが学習できると考えられる．

図 4.32 に接触点群認識の精度が悪い場合の把持の様子を示す．上図が物体 (A) の把持時の失敗例で，位置の推定精度が悪いために指で物体を倒してしまった．精度が悪かった理由として，画像の端に物体が写っていたことが理由として挙げられる．画像の端に物体が写るデータは接触点群認識の学習データに多くは含まれていないため，上手く学習されていなかった可能性がある．これに関しては，物体の位置に応じてロボットのカメラ

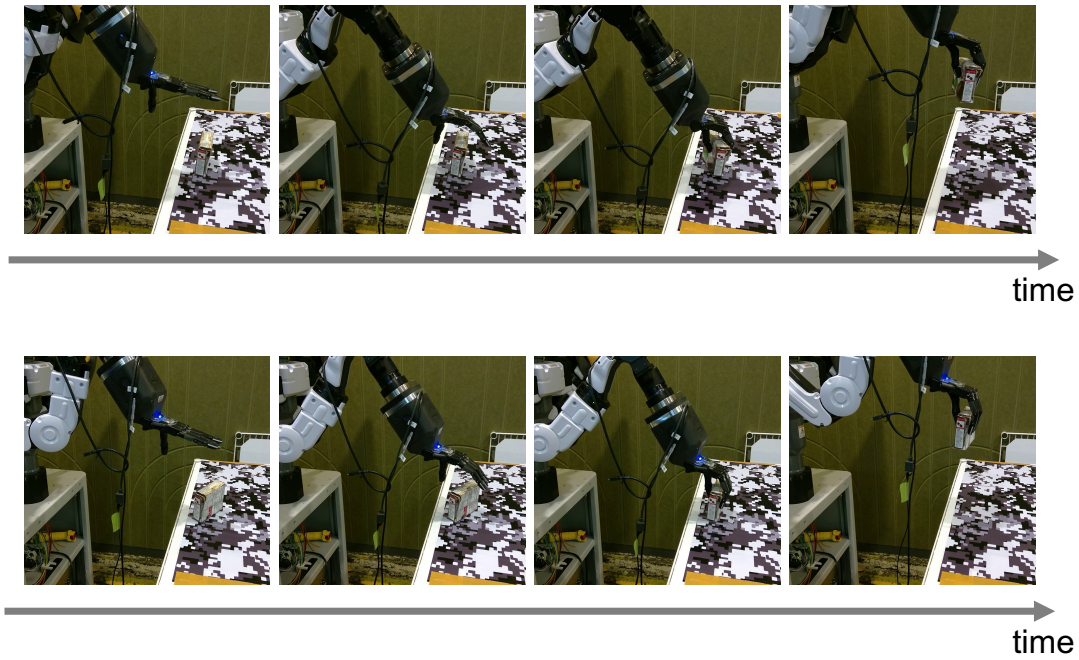


図 4.26　実機による物体 (B) に対する active-force closure での把持の様子.

を移動させることで対処可能である．下図が物体 (C) の把持時の例で，姿勢の推定精度が悪いために把持後の物体姿勢が崩れてしまった．物体 (C) の精度が悪いのは，コップが superquadrics では表現できない物体であるからである．接触点群認識を行う CNN は superquadrics で表現可能な物体のみが写った画像で学習されたため，コップのような形状の物体は学習分布範囲外となる．今後，認識の精度を上げるためにより広範囲の物体を用いて学習する必要がある．

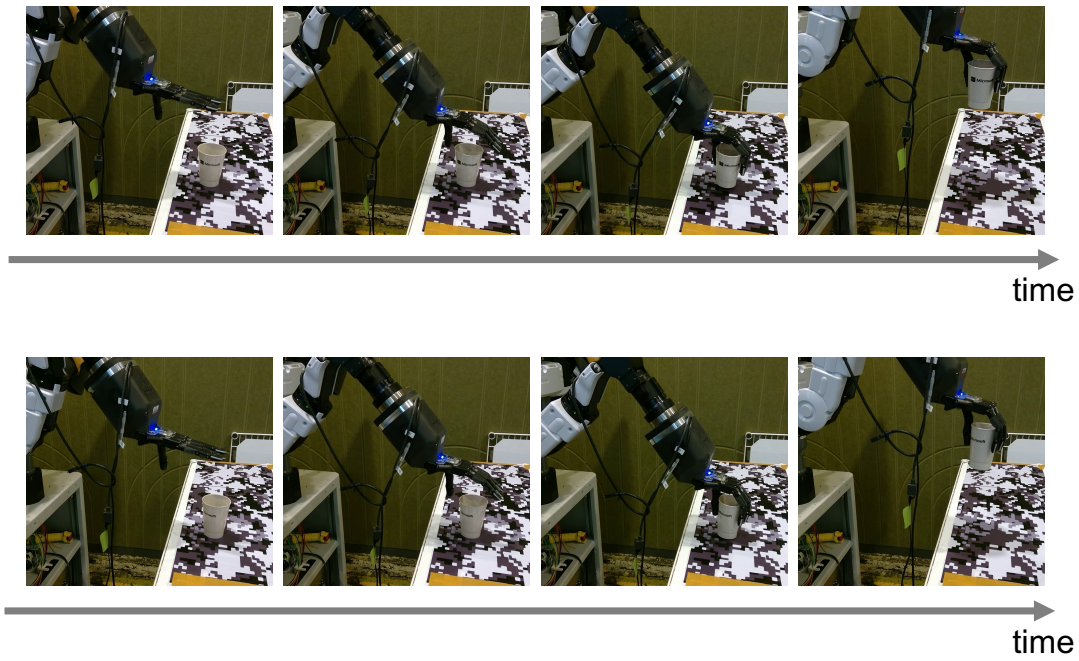


図 4.27　実機による物体 (C) に対する active-force closure での把持の様子.

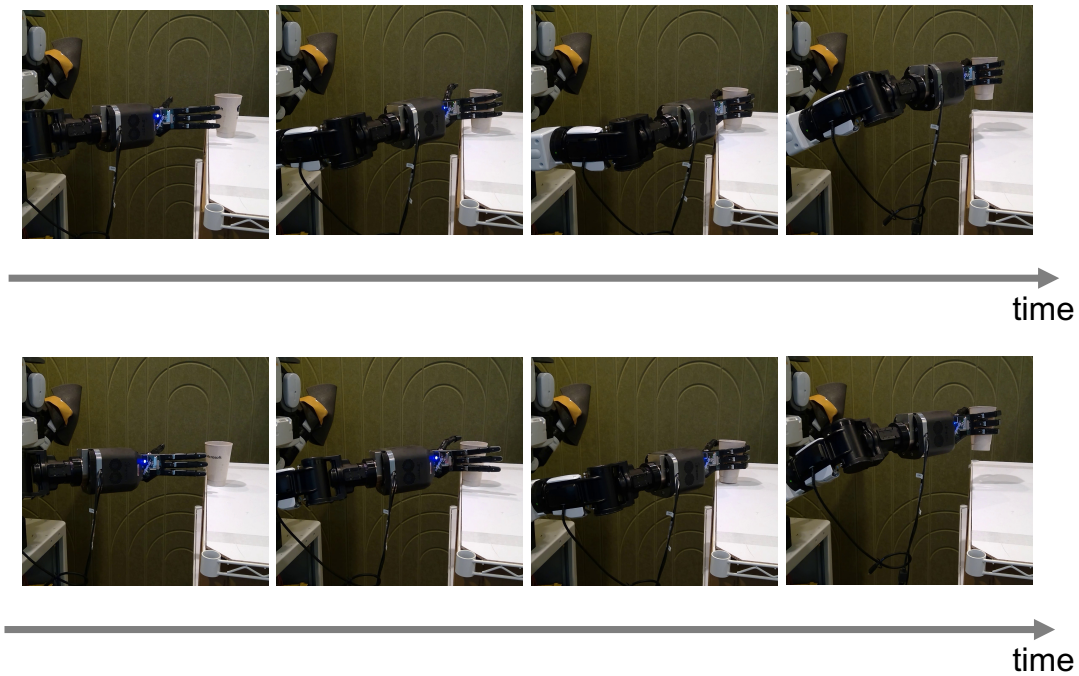


図 4.28　実機による物体 (D) に対する passive-force closure での把持の様子.

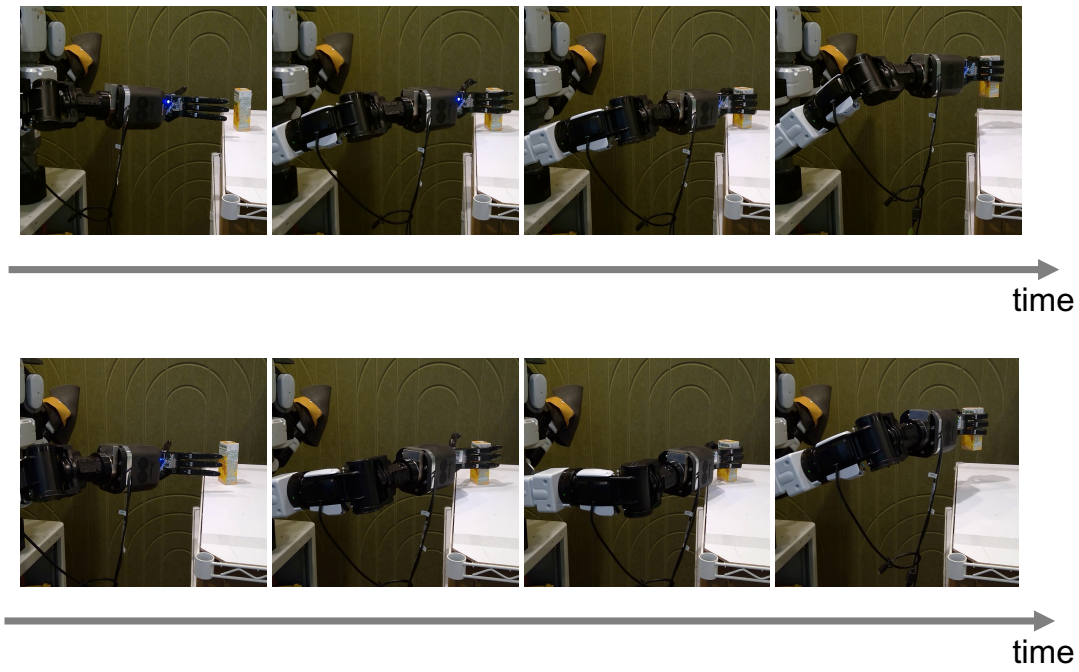


図 4.29　実機による物体 (E) に対する passive-force closure での把持の様子.

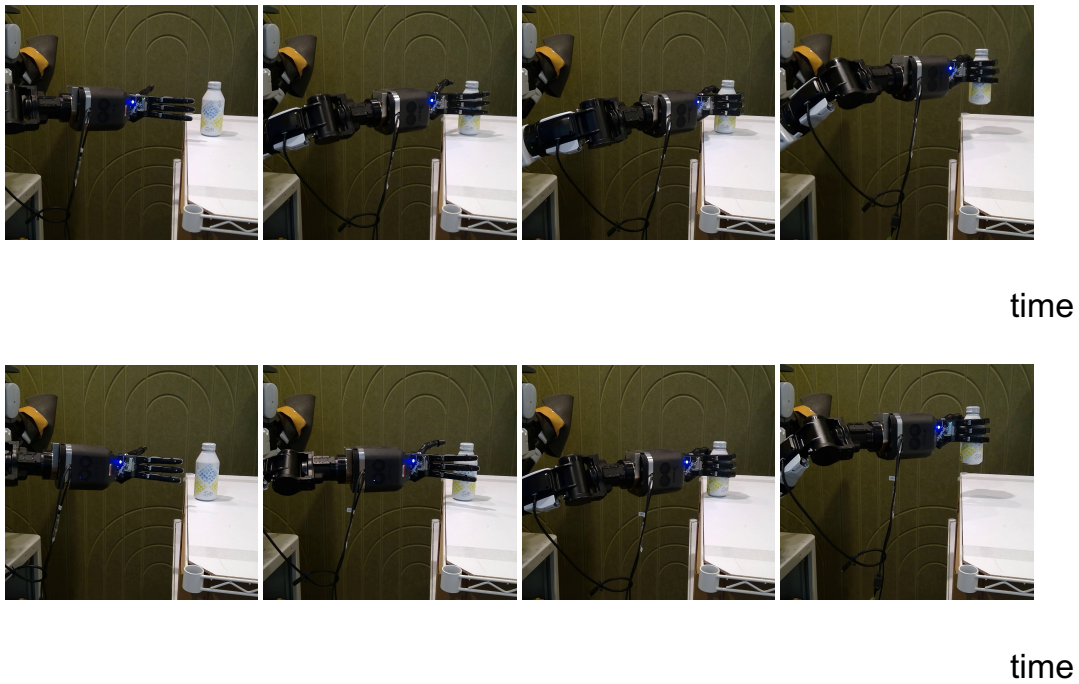


図 4.30　実機による物体 (F) に対する passive-force closure での把持の様子.

図 4.31　(D),(E),(F) は物体 (D),(E),(F) への passive-force closure での把持を上から見た様子.

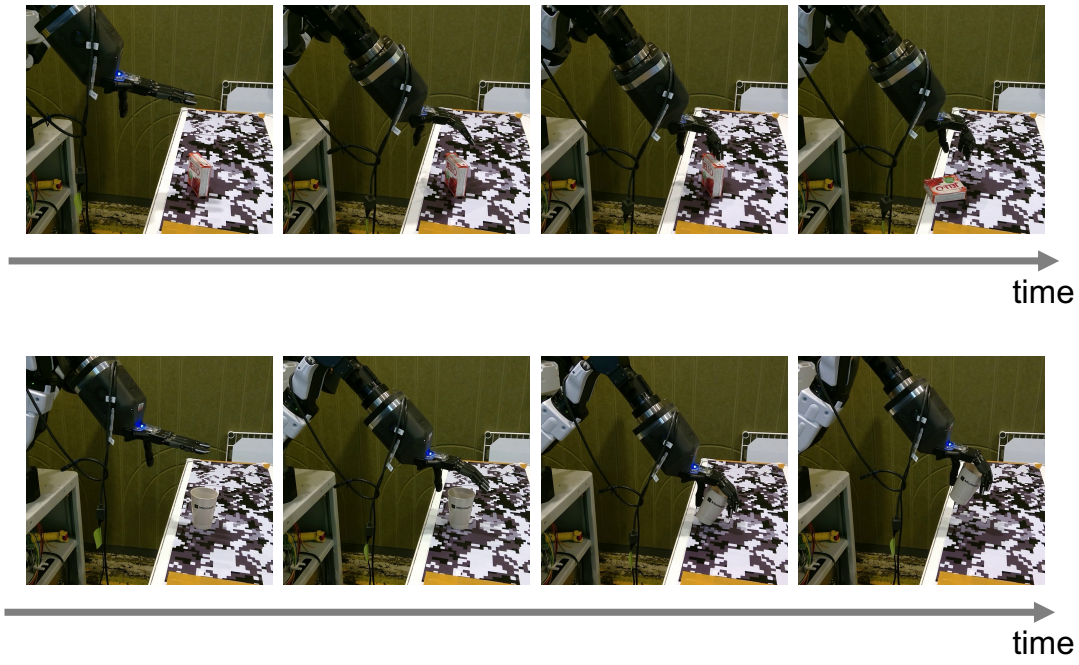


図 4.32　上図が物体 (A) の把持時の失敗例．下図が物体 (C) の把持時における接触点群認識の精度が悪い場合の例.

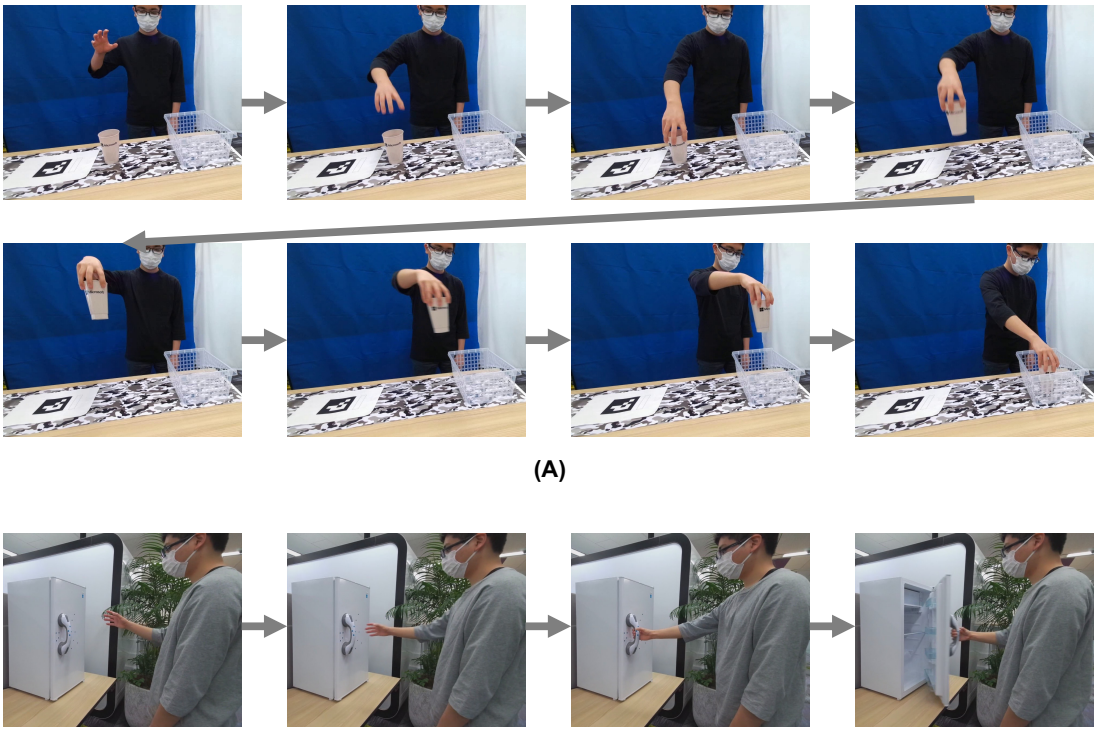


図 4.33　本実験で用いた人間の実演．(A) がコップを掴んでカゴに置く，(B) が冷蔵庫のハンドルを掴んでドアを開くという作業用の実演である．

4.8　LfO を用いた作業実行

本論文で提案した force-exertion type と 4.5 で説明したシステムの組み合わせによって，後続スキルの成功のために適切なアプローチ方向や把持プリミティブを考慮しなければならない作業が実行できることを示す．この作業の例として，本実験では (A) コップを掴んでカゴに置く，(B) 冷蔵庫のハンドルを掴んでドアを開く，という二つの作業を行う．(A) は適切な force-exertion type に加えて，アプローチ方向から適切な手の位置を実現しなければ達成できない作業，(B) は適切な force-exertion type を実現しなければ達成できない作業の例である．実験に用いた人間の実演を図 4.33 に示す．これらの実演から LfO システムを用いて実行に必要なパラメータを抽出した．

図 4.34 は (A) コップを掴んでカゴに置くという作業の結果である．上図は人間の実演を用いずに安定把持が実現できる passive-force closure で横から把持した場合の結果，下図は人間の実演を用いて active-force closure で上から把持した場合の結果である．(A) の場合，active-force closure と passive-force closure であれば安定把持を達成できる．人間の実演によって選択された active-force closure とアプローチ方向を参考に実行を行った場合，上から active-force closure による把持が行われた．その結果，カゴと腕が衝突することなくコップをカゴに置くことができた．一方で，passive-force closure で把持を行った

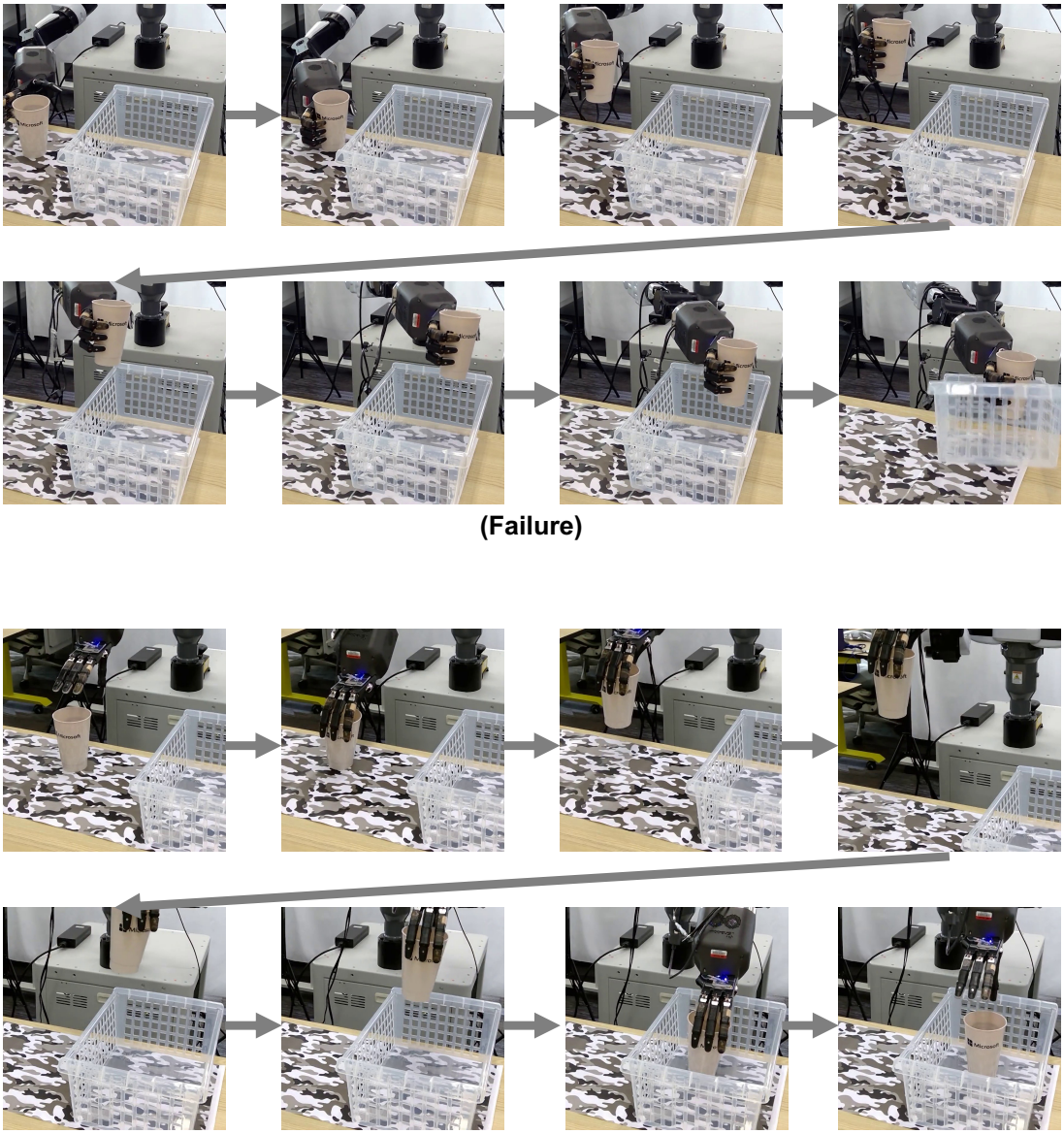
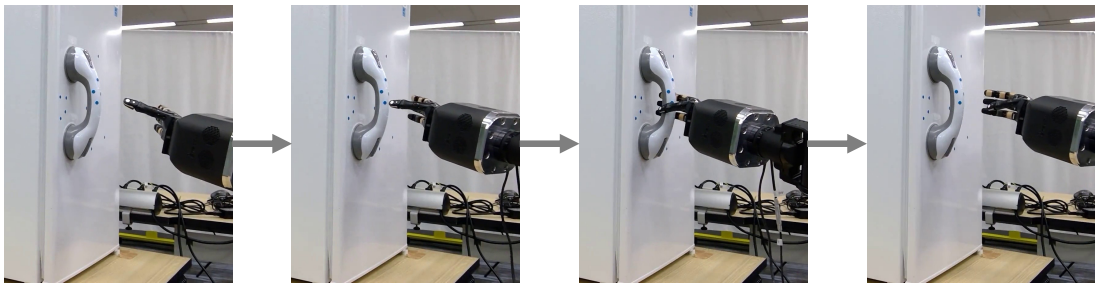


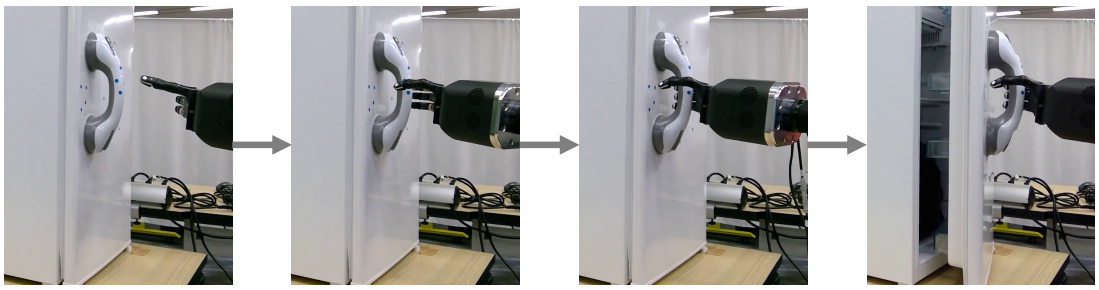
図 4.34　コップを掴んでカゴに置くという作業の結果である．(Failure) は passive-force closure でコップを把持した場合の結果で，腕がカゴと衝突して失敗した．(Success) は active-force closure でコップを把持した場合の結果で，カゴに置くことに成功した．

場合，コップを横からしか掴むことができない．その結果，コップをカゴに置く際にカゴと腕が衝突し，作業に失敗してしまった．

図 4.35 は (B) 冷蔵庫のハンドルを掴んでドアを開けるという作業の結果である．上図は人間の実演を用いずに安定把持が実現できる active-force closure で横から把持した場合の結果，下図は人間の実演を用いて lazy-closure で上から把持した場合の結果である．(B) の場合，安定把持という点ではどの force-exertion type でもハンドルを把持することができる．その中でも lazy-closure は，ハンドルの引く方向のみに直接力を発揮することがで



(Failure)



(Success)

図 4.35　冷蔵庫のハンドルを掴んでドアを開けるという作業の結果．(Failure) は active-force closure でハンドルを把持した場合の例で，ハンドルから指が滑って失敗した．(Success) は lazy-closure でハンドルを把持した場合の例で，扉を開けるのに成功した．

きるため，ドア開けの際に力を発揮するのに最も効率的な把持方法である．そのため，人間による実演で lazy-closure が選択された．lazy-closure を用いた場合には，ハンドルを引く方向に十分な力をかけることができたため，ドア開けに成功した．一方で，active-force closure でハンドルを把持した場合，把持力のみでは十分な力を発揮できずにハンドルから指が滑ってしまった．その結果，ドア開けに失敗した．以上から，提案した LfO システムによって適切な force-exertion type とアプローチ方向を選択することで，家庭内で頻出の作業が達成できることが示された．

LfO システムでは作業において必要なパラメータのみを抽出する．提案システムであれば特に force-exertion type の種類がパラメータの一部として抽出される．そして，その種類に応じたスキルが選択される．そのため，学習したスキルが物体の形状が異なっていても再利用可能である場合，実演時とは把持対象物体が異なっても適用することができる．なお，把持後に物体を動かす方向や距離に関しては実演時と同じパラメータを用いるため，実演時の環境と実行時の環境は似ている必要がある．図 4.36 はコップを掴んでカゴに置くという人間の実演を用いて，実演で用いた物体以外を操作した結果である．(A) は小さめのコップ，(B) はスパム缶を把持して操作した様子である．実演で用いた物体とは異なっていても作業が達成できることが確認できる．

人間と同等の自由度を持つロボットを用いる場合，LfO システムで得られた force-

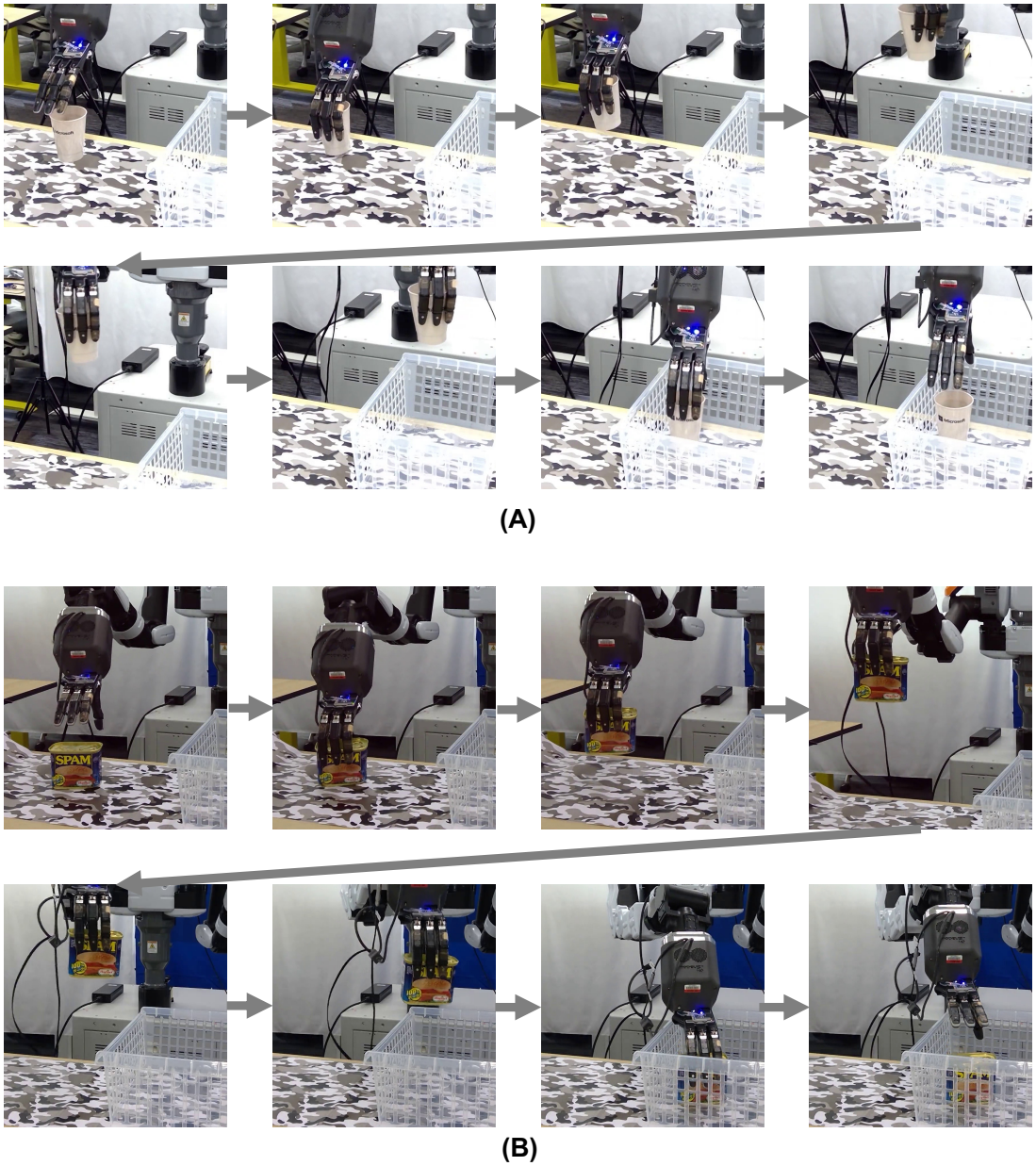


図 4.36　人間の実演とは異なる物体を用いて作業を行った結果．(A) は小さめのコップを用いた場合，(B) はスパム缶を用いた場合の結果である．

exertion type やアプローチ方向を用いることで達成不可能な姿勢の提示を緩和できるという利点もある．図 4.37 に棚の上に置かれたコップを棚の下に移動させるという作業の人間による実演と，それを用いたロボットによる実行結果を示す．この作業では棚の上にあるコップを横から passive-force closure で把持する．把持するだけであれば active-force closure で上から掴むことも可能であるが，この場合ロボットで実現できない姿勢が提示されてしまう．一方で，passive-force closure で横から掴むという姿勢は比較的実現しやすいため，作業が成功しやすい．

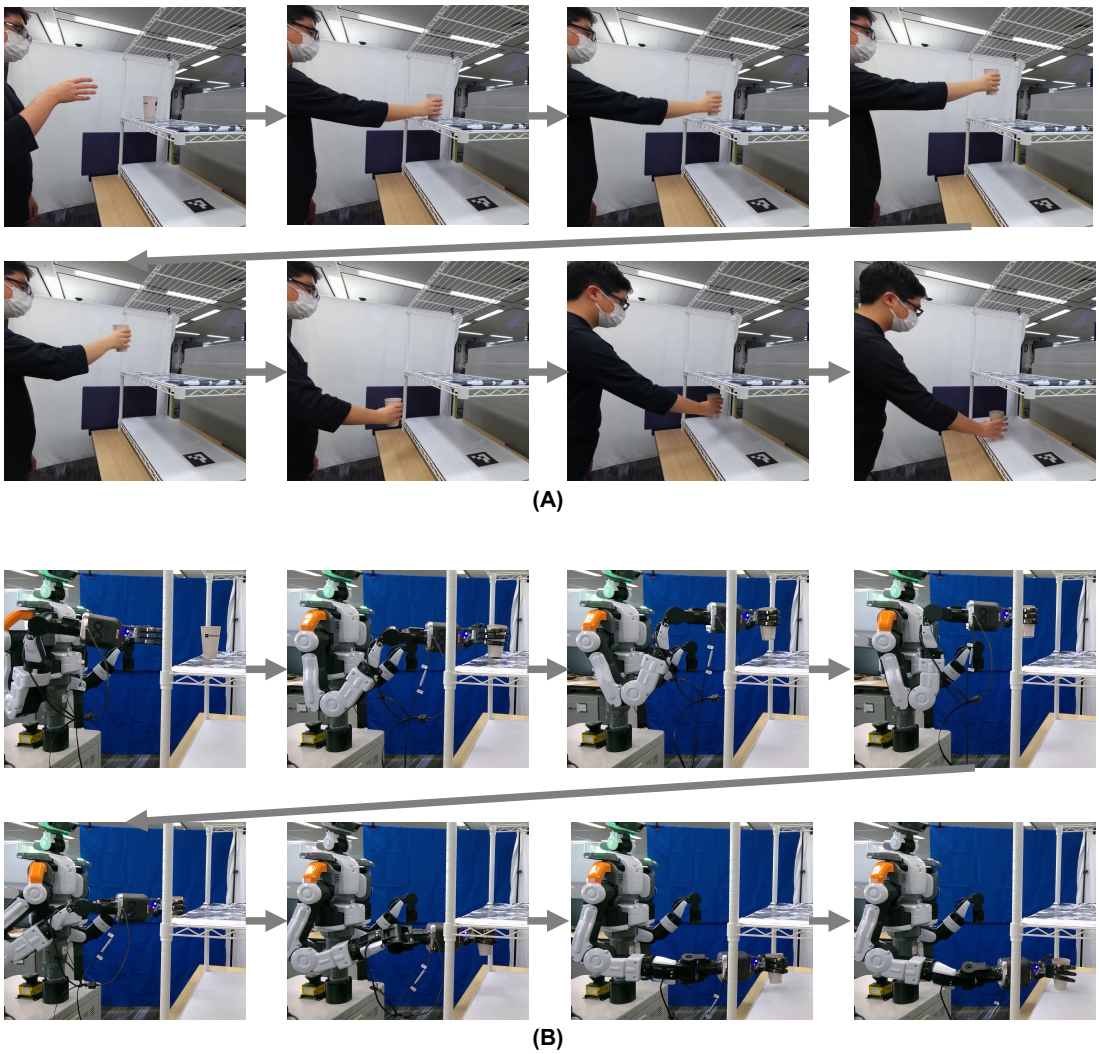


図 4.37　棚の上に置かれたコップを棚の下に移動させる作業の人間による実演 (A) と，ロボットによる実行結果 (B).

4.9 議論

4.9.1 実験結果に対する考察

本論文では，後続スキルが達成できるような把持を行うために，人間の把持やアプローチ方向を真似ることを提案した．これを実現するために，物体の大きさ・形状やアプローチ方向の変化に頑健なスキルを学習した．シミュレーション実験の結果，認識誤差のある場合に学習範囲の端に近い物体では認識誤差の影響で把持ができなくなってしまうことが分かったものの，それ以外の物体では頑健に把持が可能であることが分かった．そのため，得意な大きさ・形状範囲が異なる把持スキルを複数個組み合わせることで，将来的により広範囲の物体を把持できるようになる可能性がある．また，アプローチ方向の変化にも頑健であることが確認された．これによって，様々な人間の実演を実行できる．

実機実験の結果，追加の学習を行わなくても，実世界でも認識誤差や物体の大きさ・形状の変化に頑健に把持ができることが分かった．これは，把持スキルの入力として指先位置，手の姿勢，物体と接触しているかどうかを表現する値を用いており，シミュレータと現実での誤差が小さく抑えられたことが理由だと考えられる．

LfO と組み合わせた実行の結果，実世界で頻繁に行われるような作業が可能であることが示された．特に，適切な把持プリミティブを選択しなければ実行が困難である冷蔵庫のハンドルを掴んでドアを開けるという作業は，多くの安定把持のみを実現する手法や把持位置のみに着目した手法では困難であると考えられる．一方で，本論文のように force-exertion type を考慮して把持を行ったことで成功することができた．

提案した把持プリミティブは離散的に分類されたものであるため，スキル選択型の枠組みと相性が良い．そのため，LfO 以外のタスクプランナに組み込むことが可能であり，これまでのタスクプランナの手法と組み合わせることでより多様な作業ができるようになる可能性がある．

4.9.2 把持プリミティブの網羅性

本論文では，人間の把持を理解するために Kang による把持分類と吉川によるロボットの把持分類を用いて把持プリミティブを定義した．人間の把持分類を行なった研究としては，他にも Bullock らによる分析 [103] や Feix らによる分類 [8] がある．Bullock らは日常生活動作における把持の頻度を調査したが，その調査において頻出であった把持に関しては本論文で提案したプリミティブで網羅することができる．Feix らの分類では，いくつかの把持に関してより深く分類を行なっており，例えば箸やハサミに対する把持が追加されている．このようなプリミティブの変化を必要とするものも日常生活では多く存在する．このようなツール特有の把持は網羅することができない．このようなツールを使用したい場合には，これらのスキルもライブラリに加える必要がある．このような把持は一度の把持のみで実現することができず，passive-form closure や non-closure と同様に持ち替え動作が必要である．このような動作の設計は，次章で説明する in-hand manipulation

図 4.38　画像内に複数物体が写っている時の Grounded SAM 2 による segmentation の結果.

skill library を組み合わせることで実現できる可能性がある．なお，Feix らの分類では，non-prehensile grasp は除外している．non-prehensile grasp は，日常生活での作業中にしばしば現れるため無視することはできない．そのため，人間の把持分類として Kang の分類を用いることは妥当である．

4.9.3　机上が散らかっている場合の接触点群認識

本論文では画像内に対象物体のみが写っている場合を想定してシステムを実装した．実際にはこのような状況は稀であり，画像内に複数物体が写っている場合の方が多い．画像内に複数物体が写っている状況で本システムを動かすには，対象物体のみが写っている画像に変換する必要がある．これを実現する方法として，semantic segmentation を用いる方法が挙げられる．Grounded SAM 2 [167] のような言語で提示された対象物体を semantic segmentation する手法が存在する．カップ，茶色い箱，赤い箱が写っている画像と a cup. a brown box. a red box. という言語指示を与えた場合に，図 4.38 のように言語で与えられた物体を segmentation できる．この手法と LfO システムで得られた人間の言語指示を組み合わせることによって，対象物体のみが写っている画像に変換できる．本論文では，画像に対象物体が写っていることを前提としていたが，この手法との組み合わせにより対象物体を見つけ出すことも可能になると考えられる．以上のシステムの実装は今後の展望である．

4.9.4 パーツを考慮した把持

本論文では，物体が superquadrics で近似できると仮定して実験を行った．実際には，物体はいくつかのパーツから構成されており，複数の superquadrics を組み合わせた形になっていることが多い．複数のパーツから構成される物体の場合，その物体の機能を使用するためには，適切なパーツを把持する必要がある．本論文のシステムではこれを実現することは難しい．この問題に対処する方法の一つとして，タスク指向把持の分野 [107] で行われているように，画像上の物体のパーツを特定する part segmentation を組み合わせることが挙げられる．人間の言語指示から part segmentation することができれば，そのパーツに対する接触点群認識を行うことで適切な把持位置を実現できる可能性がある．

4.9.5 アフォーダンスの活用

環境は行動の選択肢を提供するというアフォーダンス [168] という考え方がある．把持動作の生成にアフォーダンスの概念を活用することできる．その一つとして把持プリミティブの選択がある．LfO システムでは把持プリミティブは人間の把持画像から選択するが，異なる把持でも人間の手の形状が似ている場合は画像から推定することが難しい．例えば，扉のハンドルに対して passive-force closure で把持する場合と，lazy-closure で把持する場合の手形状は酷似する可能性がある．このような場合に，ハンドルの把持では lazy-closure しか選択肢としてあり得ないといった事前のアフォーダンス知識があれば，あり得ない選択肢を排除することができる．実際に，Wake らの研究ではアフォーダンスを活用して把持推定を行なっている [86].

さらに，活用できる可能性があるものとして，把持力の調整が挙げられる．人間は物体の素材や質感からその物体の硬さを感じ取り把持力を適応的に調整することができるが，本論文で提案したプリミティブスキルでは把持力の調整はできない．ロボット側でもこのメカニズムを実装できれば，より幅広い物体を扱えるようになることが期待できる．

4.9.6 形状に応じたスキル選択

本論文で学習したスキルにより様々な大きさ・形状の物体を把持可能であることが示された．実世界には今回検証した物体の大きさ・形状範囲に含まれない物体も存在する．大きさ・形状が大きく異なる場合，必要な行動空間も大きく異なるため一つのスキルで対処できない可能性がある．より広範囲の物体にも対処する一つの方法として，複数個のスキルによって対処できる大きさ・形状範囲を広げることが考えられる．この場合には，複数個のスキルから対象物体に適したスキルを選択する機構を作る必要がある．この機構の一つとして，実行時の人間の実演時の指先位置を用いる方法が考えられる．指先位置は外部カメラからでは正確に撮ることが困難なため，人間は自分の指をトラッキングできるような頭部装着型カメラを装着して実演を行う必要がある．すなわち，指先によって形成される形状や対向指間の距離を活用することでスキルの選択が可能であると考えられる (図 4.39).

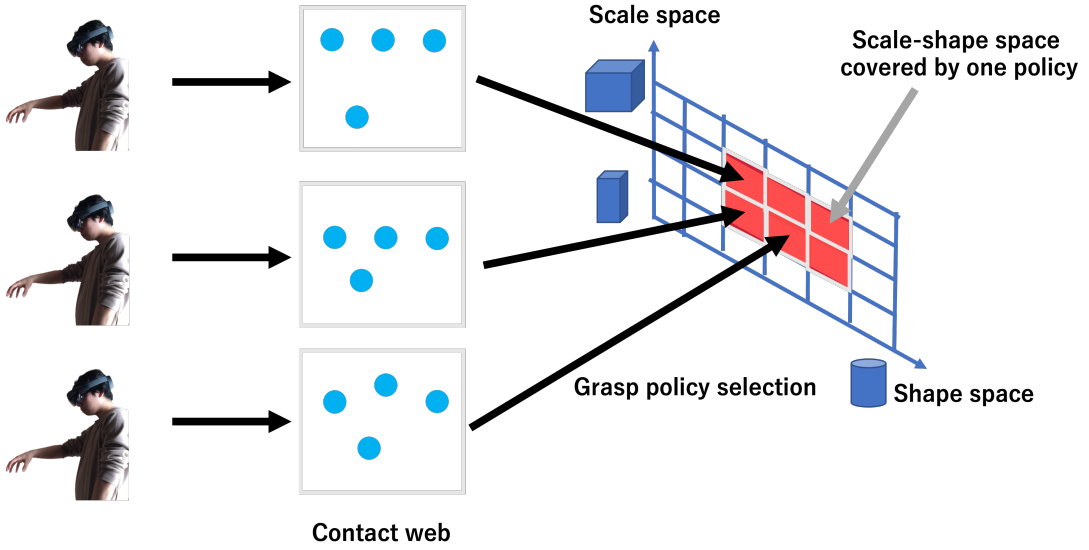


図 4.39　人間の実演の指先位置に応じた把持スキルの選択．人間は自分の指をトラッキングできるような頭部装着型カメラを装着して実演を行う．

また，対象物体に近い superquadrics のパラメータを推定 [169] し，そのパラメータからスキルを選択するという方法も考えられる．このような機構の検証は今後の展望である．

4.10　おわりに

後続スキルが達成できるような把持を行うために，人間の把持やアプローチ方向を真似ることを提案した．また，人間の把持をロボットにマッピングするために，物体への力のかけ方で分類された把持プリミティブである force-exertion type を提案した．force-exertion type には，吉川らによって提案されたロボットの把持分類以外に，人間の把持分類での non-prehensile grasp にあたる lazy-closure が新たに導入された．提案した force-exertion type と人間の実演から必要なパラメータを抽出できる Learning-from-Observation を組み合わせることで，(1) コップを掴んでカゴに入れる，(2) 冷蔵庫のハンドルを掴んでドアを開ける，という 2 つの作業が成功できることを確認した．また，学習した把持スキルにより異なる大きさ・形状の物体を把持できることを示した．これにより，物体形状ごとにスキルを設計する負担が軽減されるという利点がある．さらに，様々なアプローチ方向にも適用可能であることも示された．本研究の今後の拡張として，segmentation を組み合わせることによる更なる作業を考慮した把持の実現や，一度の把持動作のみでは実現不可能な把持プリミティブへの持ち替え動作の設計が挙げられる．

第 5 章

In-hand Manipulation に関するスキルライブラリの設計

5.1 目的とアプローチ

本章では，後続スキルの達成のための Grasp を実現することを念頭においた in-hand manipulation のスキルライブラリを設計する．In-hand manipulation に関して，多くの既存研究では手の中での物体操作のみを対象としている．しかし，後続スキルの達成を考慮すると，物体の使用を想定して手の中での物体操作だけではなく，操作後に目的の作業に適した把持プリミティブを実現するような操作を行う必要がある [15, 132]．把持が不適な場合，作業の達成のために期待された使い方で道具を使用できずに，作業に失敗する場合がある．そのため，操作後に作業に適した把持にしておくことは作業の達成において重要である．例えば，指で箱を振って箱の中身をカップの中に入れたい時に，箱の蓋がカップ側に来て，なおかつ箱を active-force closure で把持するようにしたい，という場合に使われる (図 5.1-(A))．適切に把持をしない場合，箱の蓋を指で塞いでしまう (図 5.1-(B))，物体の姿勢を変化させられない (図 5.1-(C)) 等が起こって作業が達成できなくなる．このような操作は日常生活で扱う道具の多くに対して行われる．以上のように in-hand manipulation により適切な把持プリミティブを達成するためには，操作性を持つ Active-force closure を始点として，任意の把持プリミティブを終点とする把持遷移を行うスキルが必要となる．様々な道具に対する操作を一つのハードウェアで行うことは汎用ロボットに必要な技能であり，in-hand manipulation をロボットに可能にさせることが期待される．

本論文の目的は In-hand manipulation スキルの深層強化学習による獲得である．深層強化学習を用いることで様々な不確かさに頑健なスキルの獲得が実現できるため，in-hand manipulation の分野に関しては深層強化学習による技能の獲得が盛んになっている [13, 16, 59, 120–122]．そのため，in-hand manipulation を行うための能力の強化学習による獲得も期待される．しかしながら，この操作をロボットに獲得させるのは未だに困難な問題である．この操作の獲得が困難であるのは，操作には (A) 長期的な接触状態の変化と (B) 空間的に多様な動作が必要という性質を持つことが原因である．In-hand manipulation では，目的の把持を達成するまでに何度も指の配置を変えなければならない

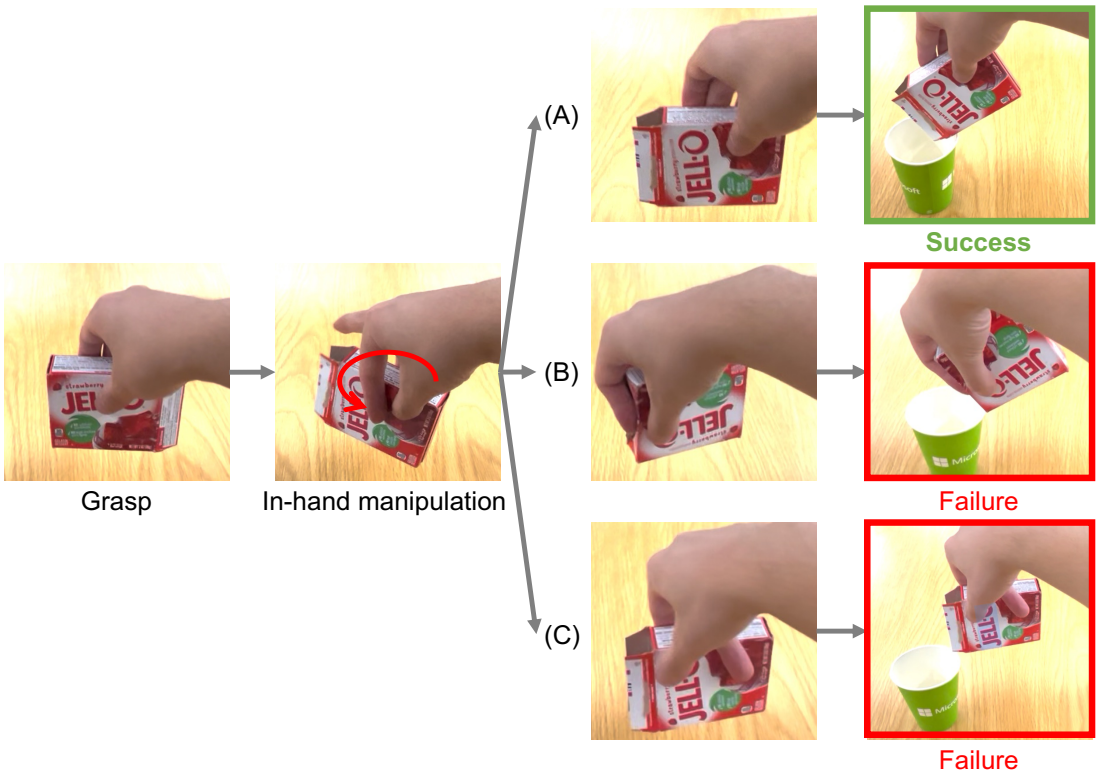


図 5.1　In-hand manipulation 後のスキル実行の例．図は箱の中身を指で振ってコップに注ぐ場合を示している．作業を達成するためには，指で箱を反時計回りに回転させて (左から 2 番目の画像の赤矢印)，かつ回転後に物体を (A) のように把持する必要がある．(B) や (C) のように把持してしまうと箱の蓋を指で塞いでしまう，物体の姿勢を適切に変化させられないといった問題が生じる．

ため，長期的に多くの接触状態の変化を伴う．さらに，現在の指の配置によって必要な動作が大きく変化する．例えば，図 5.1 の左の状況において物体に接触している親指を物体の逆側に移動させて再度接触させたい場合には，親指を動かしてそれ以外の指で物体の姿勢を変化させるように行動する必要がある．一方で，親指を移動させた後に他の指に同様の移動をさせたい場合，親指の移動の際に求められた動きとは全く異なる動きを移動させたい指にさせることを求められる．

深層強化学習でこのような性質を持つ操作を学習させる場合に，以下の二点が問題となる．一つ目は性質 (A) に由来する問題で，長期的な動作の末に把持を達成して得られる報酬が時間方向に疎であることである．このために，それまでの行動と報酬の対応付けが難しく学習が困難になる．二つ目は性質 (B) に由来する問題で，状態によって望ましい行動が大きく変わるために状態行動空間の中で探索すべき領域が広くなってしまうことである．この結果，サンプル効率が低下して学習が難しくなる．In-hand manipulation の学習のために単純に既存の物体操作のみを対象とした既存研究の報酬設計 [13, 120] に把持達成を促すための報酬を加える方法が考えられる．

しかしながら，把持達成という離散的な状態に対する報酬は性質 (A) によって時間的に

疎になってしまうため学習が困難になる．既存研究では関節角度の変位を制限する報酬を追加することで把持の達成に関する報酬を密にしていると見なすことができる．その結果，つまめる程度の大きさの物体の回転を対象として物体を回転させつつ active-force closure を維持するという in-hand manipulation を学習できる可能性がある [16, 59]．しかしながら，この報酬は学習中に探索される領域を制限してしまい，性質 (B) を持つ操作を学習することは難しい．

これらの問題点は，一般に長期的な動作を短期的な動作に分割することで解決することが可能である [35, 128, 130, 131]．実は，In-hand manipulation は全体としては長期的で複雑な動きをしているように見えるが，三種類の簡単な動作のまとまりから構成されており，この動作に着目することで短期的な動作に分割できる．把持は指先と物体の接触状態によって分類可能であり [170]，把持を変化させる操作である in-hand manipulation はこの接触状態の遷移とみなせる．この遷移は Detach, Crossover, Attach の三種類の行動表現によって記述できる [17]．detach とは指が物体から離れる動作，crossover とは指が物体をまたぐ動作，attach とは指が物体に接触する動作である (図 5.2)．そのため，これらの行動表現によって操作全体を記述可能で，この記述に基づいて動作分割することで探索すべき領域を削減できる．分割後に得られる各動作を学習することで，操作全体における探索空間の多様性から起こる学習の困難さを緩和できると考えられる．

そこで本論文では，接触状態の遷移を三種類の動作を用いてより細かな単位に分割した APriCoT(Action Primitives based on Contact-state Transition) を導入し，これらの単位ごとに学習を行うことを提案する ((図 5.2))．指先と物体の接触状態の遷移から導出されたプリミティブの組み合わせによって in-hand manipulation を実行する．すなわち，目的の把持を達成するための接触状態の遷移を考え，この遷移に対応する学習を新たに導入する．全ての指が物体に接触している形が安定であるため標準的な接触状態であるとし，これらの状態間を遷移するための detach-crossover-attach の一連の動作をプリミティブ動作とする．この動作を学習したものがプリミティブスキルである．接触状態の遷移は，把持の安定性と操作性を考慮して設計する．すなわち，前者は closure [95] の観点で把持が安定であること，後者は遷移後の操作性の観点で指同士が指の可動域を制限しないことの二点を考慮して設計される．そして，探索空間が広く最も学習が難しい crossover が一度のみ含まれる形で動作を分割する．プリミティブスキルは接触状態と指配置の遷移に基づいて設計される．この報酬は全てのプリミティブスキルで使用できる．この動作分割によって長期的な動作による学習の困難さを緩和することが可能となる [35, 128, 130, 131]．さらに，状態の変化に応じて探索空間が大きく変わる操作と比較して，各プリミティブ動作では状態が変化しても探索空間が類似しているため，探索がより容易な空間で学習することが可能になる．

本論文では，in-hand manipulation の一例として，図 5.1-(A) のような横長の物体を半回転させた後に active-force closure を達成するような操作を対象とする．この操作は日常生活でよく用いられる操作であり，横長の箱の回転以外にも，レンチ等の工具や蓋付きのカップの回転等に用いられる．本論文では四つの状態遷移に対するプリミティブ動作を対象とするが，再利用可能なプリミティブ動作を増やしていくことで本論文のアプローチ

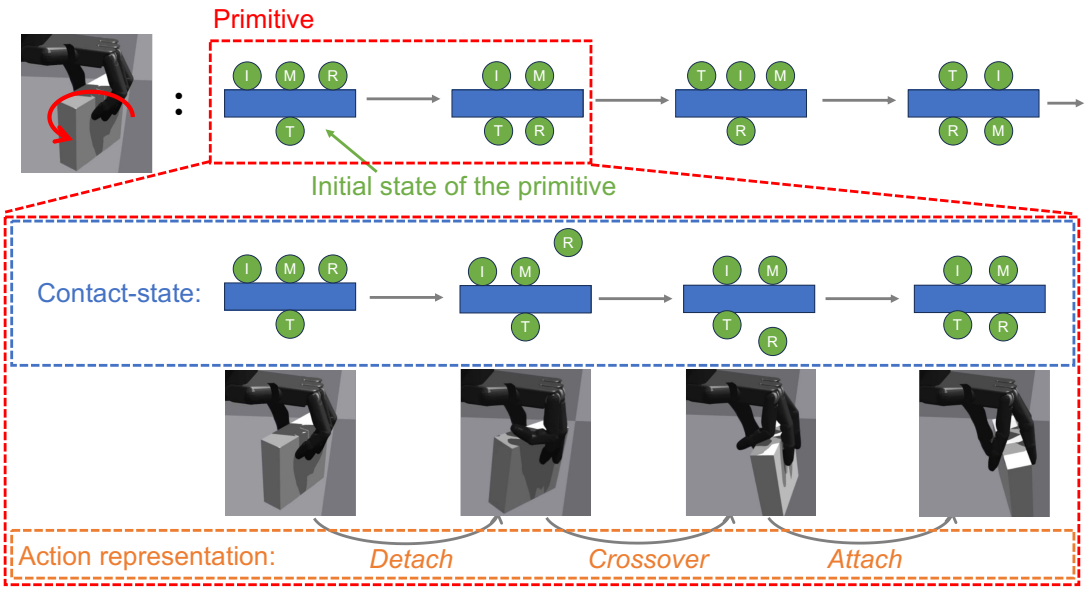


図 5.2　動作表現とプリミティブの説明．図中の I，M，R，T はそれぞれ人差し指，中指，薬指，親指を表す．この図は箱を反時計回りに回転させた場合の接触状態遷移の例である．Detach，Crossover，Attach は接触状態を遷移させる行動表現である．プリミティブの初期接触状態は，すべての指がオブジェクトに接触している最も安定した状態に設定される．

で様々な持ち替え動作に対しての in-hand manipulation ができる可能性がある．例えば，物体を半回転させて active-force closure から passive-form closure に持ち替える動作やその逆も可能となる．本論文はプリミティブスキルの組み合わせによって様々な in-hand manipulation を実現するための研究における第一歩となる．

以降では，接触状態遷移の考察，スキルの学習設計，シミュレーションと実世界との誤差 (Sim2Real Gap) への対処に関して説明する．最後に，シミュレーション実験を通して本手法の有効性を示す．

5.2　接触状態遷移の考察

接触状態の遷移グラフに着目して回転操作の手順を考察する．本論文では，簡単のために上面が棒形状の物体の操作を仮定する．これには食料品や工具等の家庭内で頻出のものが含まれる．接触状態の遷移は指先の detach, crossover, attach によって起こることが知られている [17] ため，これらの動作表現に基づいて遷移グラフを構築する．接触状態は把持の安定性と操作性を満たすもののみが採用される．すなわち，前者は closure [95] の観点で把持が安定であること，後者は遷移後の操作性の観点で指同士が指の可動域を制限しないことの二点を考慮して設計される．具体的には，把持が安定であるかどうかに関しては，prehensile grasp が実現される，一度に複数の指が crossover しないということを条件とする．操作性に関しては指同士がもつれないために親指，人差し指，中指，薬指の順に

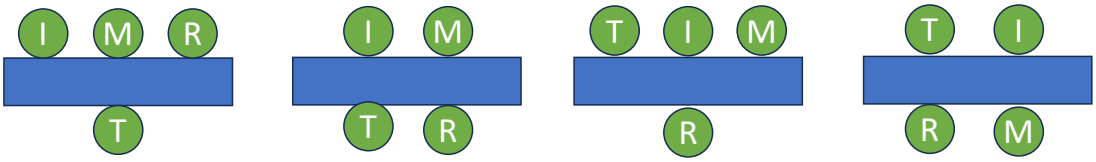


図 5.3　in-hand manipulation における接触状態．I，M，R，T はそれぞれ人差し指，中指，薬指，親指を表している．

時計回りに位置していることを条件とする．以上の条件を満たすものとして図 5.3 に示した四つの接触状態が列挙できる．

次に，In-hand manipulation は，detach, crossover, attach による接触状態の変化を行う操作と状態変化せずに物体の並進や回転による物体位置姿勢の変化をさせる操作に分類できるということに着目して，遷移可能な状態同士を結ぶことで接触状態遷移を列挙する (図 5.4)．本論文では，この遷移のそれぞれをプリミティブと定義すると，図 5.5 に示すようなスキルライブラリが設計される．例えば，このプリミティブを用いることで，本論文で対象とする横長の物体を半回転させた後に active-force closure を達成するような操作に関しては，図 5.6 のように遷移グラフを構築できる．この分割によって四つのプリミティブスキルが必要であることが分かる．まず，薬指を物体から detach して，crossover させ，attach することで接触状態を変化させる．次に親指，中指，人差し指も順に同様の動作をすることで，所望の把持を実現することができる．ここでは，これらのスキルを PolicyA, B, C, D と呼ぶ．これらの動作は detach-crossover-attach の一連の動作を持つ．この一連の動作が可能となるようにスキルの設計を行う．なお，図 5.6 とは逆向きに遷移する場合も存在するが，本論文では図の場合のみを学習の対象とする．逆向きの場合も同様に適用可能となるようなスキル設計を行う．図 5.6 は接触状態の遷移グラフの一例であり，遷移グラフは対象把持や物体に応じて変化する．これは人間の物体操作方法を観察して構築するという方法で解決できる可能性がある [3, 47, 155]．本論文では，遷移グラフの構築方法に関しては対象外とする．

5.3　スキルの学習設計

APriCoT を実現するための状態，行動の説明をした後に，提案された報酬と初期状態分布の設計方法に関して説明する．この報酬設計は接触状態を遷移させるようなプリミティブスキルの学習において使うことができる．その後，学習の実装として選択した teacher-student learning [16, 59, 171, 172] を用いた学習に関して説明する．

5.3.1　学習の定式化

まず，スキル学習をマルコフ決定過程として定式化する．本論文ではスキルは対象とする接触状態遷移の数 N だけ存在する．この遷移を i $(1 \leq i \leq N)$ とすると，次

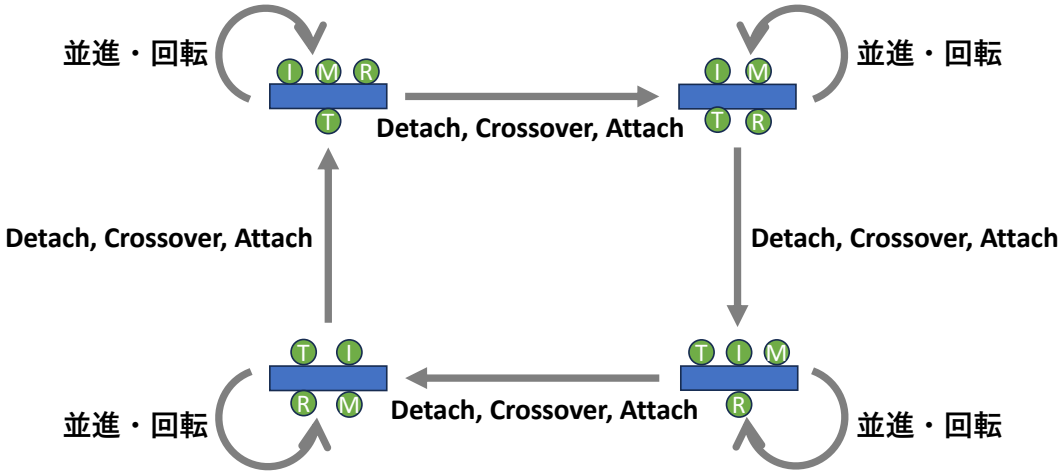


図 5.4　in-hand manipulation における接触状態遷移．I，M，R，T はそれぞれ人差し指，中指，薬指，親指を表している．

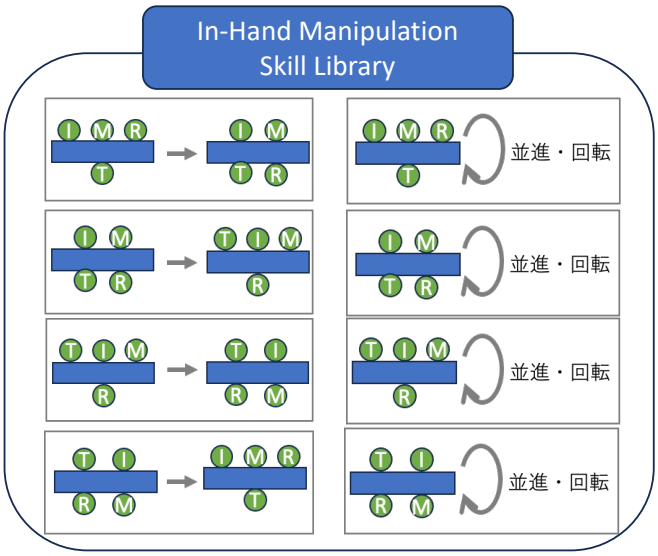


図 5.5　in-hand manipulation におけるスキルライブラリ．

のように定式化される．マルコフ決定過程は状態 $s^{(i)} \in \mathcal{S}^{(i)}$（$\mathcal{S}^{(i)}$ は状態空間），行動 $a^{(i)} \in \mathcal{A}^{(i)}$（$\mathcal{A}^{(i)}$ は行動空間），状態遷移 $\mathcal{T}^{(i)} : \mathcal{S}^{(i)} \times \mathcal{A}^{(i)} \to \mathcal{S}^{(i)}$，初期状態分布 $\rho_0^{(i)}$，報酬関数 $r^{(i)} : \mathcal{S}^{(i)} \times \mathcal{A}^{(i)} \to \mathbb{R}$ を持つ．目標は，割引率を $\gamma \in [0, 1)$ として累積報酬和 $J(\pi^{(i)}) = \mathbb{E}_{\pi^{(i)}}\left[\sum_{t=0}^{T-1} \gamma^t r^{(i)}\left(s_t^{(i)}, a_t^{(i)}\right)\right]$ を最大化するスキル $\pi^{(i)}(a^{(i)}|s^{(i)})$ を求めることである．ここで，T はエピソードの長さ，$s_0^{(i)} \sim \rho_0^{(i)}$，$a_t^{(i)} \sim \pi^{(i)}(s_t)$，$s_{t+1}^{(i)} = \mathcal{T}^{(i)}\left(s_t^{(i)}, a_t^{(i)}\right)$ である．学習されたスキル $\pi^{(i)}$ を逐次実行することで目標とする操作を達成する．この時，スキルを切り替えた際の初期状態は $s_0^{(i+1)} = \mathcal{T}^{(i)}\left(s_{T-1}^{(i)}, a_{T-1}^{(i)}\right)$ となる．

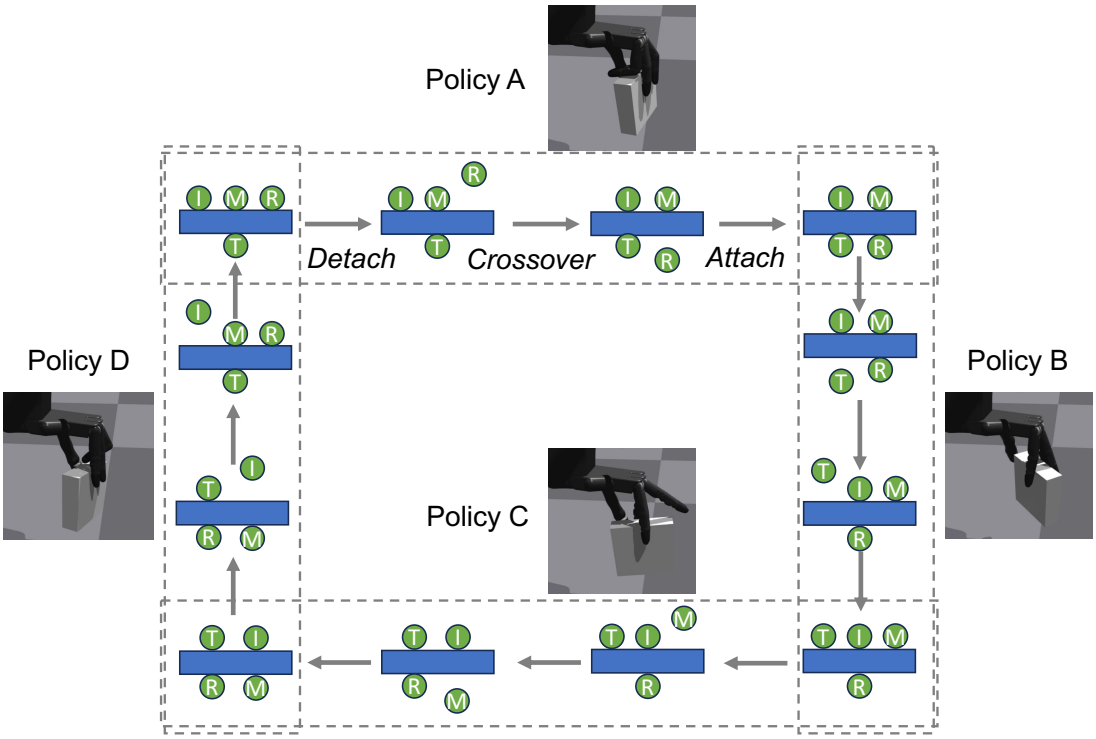


図 5.6　本論文の対象となる操作における接触状態の遷移．I，M，R，T はそれぞれ人差し指，中指，薬指，親指を表している．点線で囲まれた部分のそれぞれがプリミティブ動作である．

5.3.2　状態，行動，報酬の設計

状態

実世界へのスキルの適用を考えた時に，様々な形状の物体に対して適応的に行動できるスキルが学習されることが望ましい．そのようなスキルの学習にはセンサ情報から物体形状を推定することが必要である．センサ情報である関節角度の指令値と実行値の差分から大まかに物体形状推定をすることが可能であると考えられる．そこで，状態 s_t は関節角度の指令値 q_t と実測値 $\hat{q}_t$ を含む形で設計する．

行動

行動 a_t は関節角度の指令値の変化量 Δq_t とする．

5.3.3　報酬

報酬は以下の式 (5.1) で表されるように状態遷移に関する項 $r_{\text{transition}}$，物体位置姿勢の維持に関する項 r_{obj} に分割される．

$$r = r_{\text{transition}} + r_{\text{obj}} \tag{5.1}$$

$r_{\text{transition}}$ は式 (5.2) で表現される．これは detach, crossover, attach の行動 [17, 47] に基づいて設計される．detach, crossover, attach の順に行動することを学習するために，段階的に報酬が与えられるように報酬を設計する．

$$r_{\text{transition}} = w_{\text{det}} r_{\text{det}} + r_{\text{X}} + w_{\text{att}} r_{\text{att}} \tag{5.2}$$

$r_{\text{transition}}$ を構成する $r_{\text{det}}, r_{\text{X}}, r_{\text{att}}$ は配置が変化する指 f の状態に依存する．r_{det} は detach を促すための報酬で指が物体を跨ぐまで $r_{\text{det}} = -d_F^z$，跨いだら 0(d_F^z は物体座標系での物体上面から指までの距離)．r_{X} は crossover の成功へ誘導するための報酬である．指が物体を跨いでいなければ 0，跨いでかつ物体上面より上に指先が位置している場合に正の固定値の報酬 c_{X1}，跨いでかつ物体上面より下に指先が位置している場合に正の固定報酬 $c_{\text{X1}} + c_{\text{X2}}$．$r_{\text{att}}$ は attach を促すための報酬である．r_{att} は指が物体を跨いで物体上面より下に移動したら $r_{\text{att}} = -d_F^x$，その後所望の接触状態を達成したら $r_{\text{att}} = -d_F^x + c_{\text{att}}$，それ以外では 0(d_F^x は物体表面から指までの距離)．r_{det} や r_{att} を連続関数とすることで報酬を密にして学習の効率化を図る．

r_{obj} は式 (5.3) で表される．

$$r_{\text{obj}} = w_{\text{dir}} r_{\text{dir}} + w_{\text{rot}} r_{\text{rot}} + w_{\text{pos}} r_{\text{pos}} + r_{\text{term}} \tag{5.3}$$

r_{dir} は把持の安定性のために指と物体の面接触を促す報酬である．指が物体を跨ぐまでは $r_{\text{dir}} = -\sum_{f \neq F} \theta_f$ で，指を跨いだ後は $r_{dir} = -\sum \theta_f$(θ_f は指 f の先端が指す向きと物体上面に垂直な向きとのなす角度)．$r_{\text{rot}}, r_{\text{pos}}$ は物体の位置と重力軸 (yaw 軸) 周り以外の回転を初期状態から変化させないようにする報酬である．ここで $r_{\text{rot}} = -(\theta^{\text{roll}} + \theta^{\text{pitch}})$, $r_{\text{pos}} = -(p - p^{\text{init}})$($\theta^{\text{roll}}, \theta^{\text{pitch}}$ は物体を後ろから正面に突き抜ける直線と横に突き抜ける直線に対する回転角度，p は世界座標系での物体の位置，p^{init} は世界座標系での物体の初期位置)．r_{term} は早期終了を防ぐための報酬である．早期終了していない場合では $r_{\text{term}} = c_{\text{term}}$，早期終了した場合では $r_{\text{term}} = -c_{\text{term}}$．なお，早期終了は戻すことが難しい状態に陥った場合に起こる．これによって学習を効率化する．具体的には，物体が手から落ちた場合，指の面接触が維持できなくなった場合 ($\sum \theta_f > \Theta_1$)，物体の姿勢が大きく崩れた場合 ($\theta_{\text{roll}} > \Theta_2$) に起こる．

5.3.4 初期状態の設計

Initial-state design

学習時において，初期状態を多様なものにすることはスキルの頑健性にとって非常に重要である．さらに，skill chaining においてスキル全体の性能を上げるという観点で，スキルの入力分布を前段階のスキルの出力分布に近づけることも重要である [129, 130]．そこで本論文では，最初の遷移におけるスキルではシミュレーションで大量に生成された初期状態を用いて学習を行う．そして，それ以降のスキルでは前段階のスキルから出力された状態を初期状態として学習を行う (図 5.7-(A))．

シミュレーションでの初期状態の生成は以下の手順で行う．

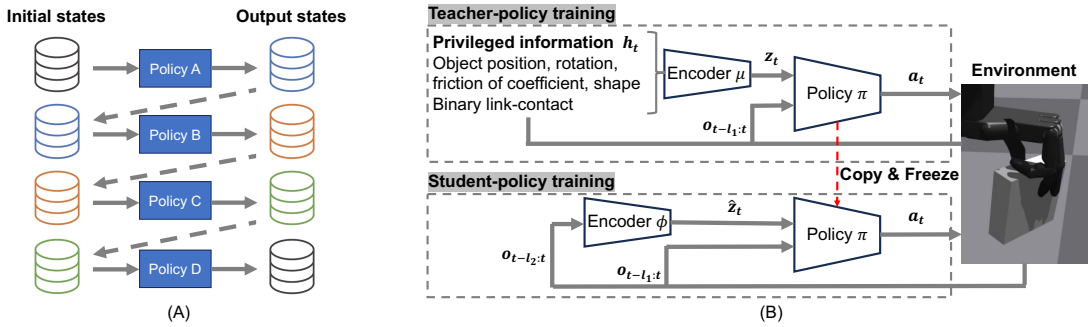


図 5.7　学習の概要．(A) はスキルの学習に用いる初期状態に関しての説明である．(B) は teacher-student learning の手順を示したものである．

1. 物体の位置姿勢と関節角度指令値の基準値に一様分布からサンプリングされたノイズを加える．
2. 50 イテレーション先までシミュレーションを進める．
3. 把持が維持されているものを初期状態として採用する．

このような手順で収集された初期状態を最初のスキルの学習に用いる．

5.3.5　Teacher-student learning

学習は既存研究と同様に teacher-student learning の枠組みを用いる [16, 59, 171, 172]．ゼロからセンサ情報のみでスキルを学習するのは物体に関する情報が少ないために時間がかかる可能性がある．そこで，学習の効率化を図るためにこの学習方法を用いる．teacher-student learning では，環境に関する特権情報とセンサ情報を用いて teacher policy を学習させ，その後 teacher policy を用いてセンサ情報のみを入力可能な student policy の学習を誘導する．これによって，学習の効率化が可能となる．さらに，student policy を環境の変化に頑健にさせることができる．teacher policy はスキル π とエンコーダ μ，student policy はスキル π とエンコーダ ϕ から構成される (図 5.7-(B))．teacher policy から student policy にスキル π のパラメータはコピーされる．エンコーダ μ は特権情報を潜在変数 z_t に符号化する．エンコーダ ϕ は複数時刻分のセンサ情報を潜在変数 $\hat{z}_t$ に符号化する．student policy の学習時には，これらの潜在変数を近づけるように学習を行う．すなわち，以下の損失関数 L を最適化することで，エンコーダ ϕ が学習される．

$$L = \|z_t - \hat{z}_t\|_2^2 \tag{5.4}$$

これによって特権情報とセンサ情報の対応を学習することができる．

特権情報 h_t としては物体に関する情報である物体の位置姿勢，形状，摩擦係数と指の各リンクと物体との二値の接触状態が含まれる．物体の形状に関しては superquadrics [157] のパラメータを含める．センサ情報 o_t は関節角度の指令値 q_t と実測値 $\hat{q}_t$，前の時刻における行動 a_{t-1} が含み，状態 s_t は o_t を過去の三時刻分含む．三時刻分を含むことで時系列情報を学習する．行動 a_t には関節角度の指令値の変化量 Δq_t が含まれる．したがって，

$s_t = \{h_t, o_{t-2:t}\}$, $a_t = \Delta q_t$ である．Teacher policy では，特権情報 h_t がまず全結合層からなるエンコーダ μ に入力され潜在変数 z_t が出力される．次にその潜在変数 z_t とセンサ情報 o_t が同時に全結合層のスキル π に入力され行動 a_t が出力される．Student policy では，センサ情報には関節角度の指令値 q_t と実測値 $\hat{q}_t$，前の時刻における行動 a_{t-1} が過去の三十時刻分が含まれる．$o_{t-29:t}$ が Temporal convolutional network で構成されたエンコーダ ϕ に入力され潜在変数 $\hat{z}_t$ が出力される．その後，$\hat{z}_t$ と o_t が同時に π に入力され a_t が出力される．

5.4 Domain Randomization

実世界でスキルを適用することを考慮すると，実世界に存在する不確かさに頑健なスキルを学習する必要がある．そこで本論文では，[13, 119] のと同様に環境や観測に関してランダマイズする．

環境に関しては，物体の形状と摩擦係数，重量，重心位置を変化させる．物体の形状は superquadrics のパラメータを変化させる．

観測に関しては，関節角度の実測値に対して一様分布からサンプリングされたノイズを加えることでランダマイズを行う．

5.5 シミュレーション実験

5.5.1 準備

大量のデータを高速に収集するために，学習には IsaacGym Simulator を用いた．学習時には 32768 個の環境を並列に動かすことでデータを収集した．各環境には四本指ハンドである Shadow dexterous hand lite と，superquadrics で表現される物体が含まれる．シミュレーション周波数は 120Hz で，制御周波数は 20Hz に設定した．各スキルの学習でエピソードは 100 イテレーションとした（$T = 100$）．

superquadrics や物体摩擦，観測される関節角度に対するランダマイズに用いたパラメータを表 5.1 に示す．本実験で用いた superquadrics の randomization の範囲は YCB object dataset [158] の Food items に含まれる物体の中でロボットハンドで操作できる大きさの物体を内包している．Teacher policy の学習アルゴリズムには Proximal Policy Optimization [165] を用いた．Student policy の最適化には Adam [163] を用いた．PolicyA,B,C,D に用いた報酬と早期終了に関するハイパーパラメータは表 5.2，5.3 のように設定された．

対象とする操作に対する本手法の有効性を示すために，実験では以下の二つの手法を Baseline として比較した．

1. Baseline A: 物体の手の中での回転のみを対象とする手法 [13] に変更を加えたもの．目的とする把持が達成された場合に正の報酬を返すように報酬設計が変更された．目的の操作は時間方向に多様な探索が必要であり，接触状態の遷移を考慮して動作

表 5.1　randomization の範囲.

Parameter	Range
Object depth	uniform($[0.02, 0.35]$)m
Object width	0.11m
Object height	0.1m
Object shape (ϵ_1)	10^{-6}
Object shape (ϵ_2)	uniform($[10^{-6}, 1]$)
Friction coefficients	uniform($[0.7, 1.3]$)
Joint noise	uniform($[-0.05, 0.05]$)rad

表 5.2　実験で用いた報酬と早期終了のハイパーパラメータ.

	c_{X1}	c_{X2}	c_{att}	c_{term}	Θ_1	Θ_2
A,B,C	0.1	0.2	1	0.1	120°	15°
D	0.2	0.5	0.5	0.1	120°	15°

表 5.3　実験で用いた報酬の係数のハイパーパラメータ.

	w_{det}	w_{att}	w_{dir}	w_{rot}	w_{pos}
A,B,C	0.1	0.1	0.1	0.1	0.01
D	2	1	0.1	0.1	0.01

分割をした方が目的としている把持を達成できるということを検証するために用意された.

2. Baseline B: つまめる程度の大きさの物体のみを対象とした手法である [16]. 目的の操作では広い探索空間で学習する必要があリ，状態遷移に基づいて探索空間を狭めた方が操作の学習ができるということを検証するために用意された．この手法では指先位置の変化量を負の報酬として加えることで，指先位置を大きく変化させずに物体を回転させることを促している．その結果，つまめる程度の物体では目的とする把持を維持したまま物体を回転させることができる.

5.5.2　結果

ベースラインとの比較

Baseline の学習結果と本手法の学習結果を図 5.8 に示す．物体を半回転させて所望の把持を実現できていれば成功と定義した．Baseline では物体を手の中で半回転させる，目的とした把持を達成することを両方とも満たすようなスキルを学習することはできなかった．Baseline A では物体を手の中で半回転させることはできたものの，目的とした把持を達成することはできなかった．目的とする把持が達成されるまでに長いステップが必要で，把持の達成時にもらえる報酬が疎になってしまう．その結果探索が困難となるからである.

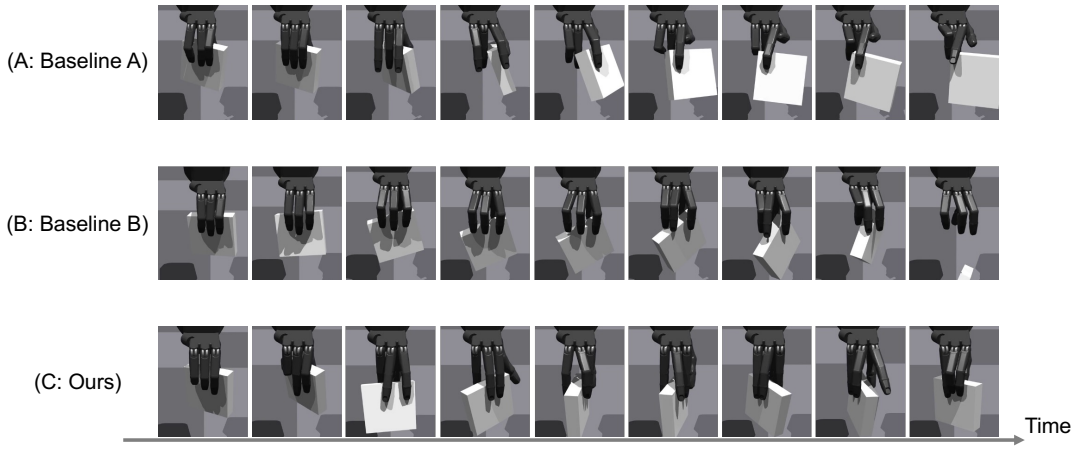


図 5.8　学習結果の例．(A) が Baseline A，(B) が Baseline B，(C) が APriCoT による結果．

Baseline B では crossover をするために，roll 方向周りに物体を回転させてしまった．そして，物体に接触可能な領域が減ってしまい，その結果把持が維持できずに物体を落下させてしまった．これは報酬によって指先位置の探索空間を制限している影響で，指先位置の変化が小さくても物体を回転させることができるように物体姿勢を変化させてしまっていることが原因である．一方で，我々の手法では接触状態の遷移や crossover を考慮して学習設計をしたことで，スキルが物体を回転させながら目的とした把持を達成することができた．

頑健性評価

本論文で学習されたスキルの頑健性を評価するために，物体の形状ごとに操作の成功率を評価した．物体には深さが $(2, 2.3, 2.6, 2.9, 3.2, 3.5)$cm，形状パラメータ ϵ_2 が $(10^{-6}, 0.25, 0.5, 0.75, 1)$ の組み合わせのものを用いた．ランダムにサンプリングされた 1000 個の初期状態に対しての成功率を求めた．各形状に対する成功率を図 5.9 に示す．ほぼ全ての形状で成功率 90% 以上を達成した．この結果から，本手法で学習されたスキルは物体形状の変化に頑健であることが確認できた．太い楕円中では成功率が 75% 程度であった．これは曲率が高いために，物体が横に滑り落ちやすくなっていたからである．

図 5.10 に把持の成功例を示す．(A),(B),(C),(D),(E),(F) はそれぞれ深さ，ϵ_2 が $(2, 10^{-6}), (2, 1), (2.75, 10^{-6}), (2.75, 1), (3.5, 10^{-6}), (3.5, 1)$ の物体への実行例である．異なる大きさ・形状の物体で初期状態が異なる場合でも，上手く実行できていることが確認できる．

報酬設計の比較

本論文で提案した報酬設計の妥当性を評価するために，報酬関数の比較を行なった．物体の位置姿勢の維持を促す項の $r_{\mathrm{rot}}, r_{\mathrm{pos}}$ のみが無い場合 (w/o pose) と，指の物体への面接触を促す項の r_{dir} のみが無い場合 (w/o direction) と提案した報酬設計 (ours) との比較を行った．具体的には，w/o pose では，$r_{\mathrm{obj}} = w_{\mathrm{dir}} r_{\mathrm{dir}} + r_{\mathrm{term}}$，w/o direction で

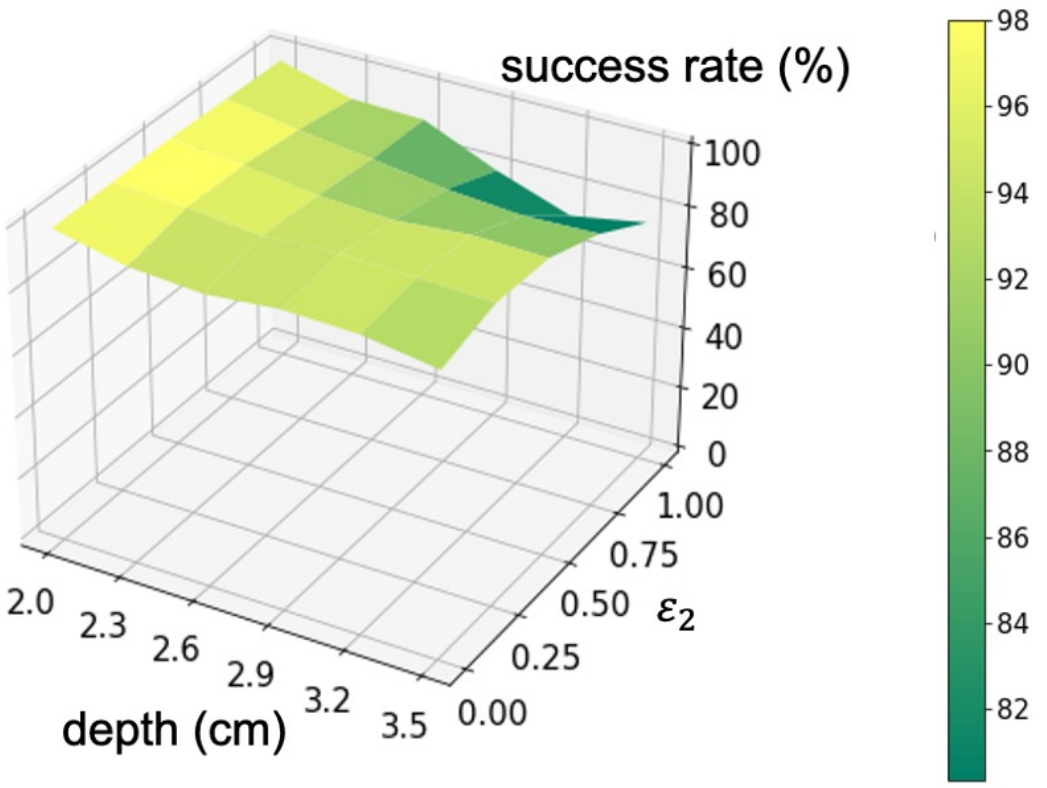


図 5.9　成功率を示した図．底面は物体の深さと形状を表している．縦軸は成功率を表している．

は，$r_{\mathrm{obj}} = w_{\mathrm{rot}} r_{\mathrm{rot}} + w_{\mathrm{pos}} r_{\mathrm{pos}} + r_{\mathrm{term}}$ となる．また，報酬設計によって平均成功率に有意な差があるのかを確認した．w/o pose と w/o direction の各物体形状への成功率の結果を図 5.11 に示す．また，各条件の平均成功率を図 5.12 に示す．成功率のデータに対して Shapiro-Wilk 検定を行った結果，正規性を示した ($p = 1.78 \times 10^{-7} < 0.05$)．次に，反復測定分散分析を行った結果，有意差があることを確認した ($p = 1.02 \times 10^{-8} < 0.05$)．そのため，対応ありの t 検定により各条件の比較を行った．その際，Bonferroni 法により多重比較検定補正を行なった．その結果，w/o pose と w/o direction，w/o pose と ours の間で有意差を確認した ($p = 2.23 \times 10^{-8}, 4.65 \times 10^{-5} < 0.05$)．

図 5.13 に w/o pose と w/o direction での実行例を示す．w/o pose の実行例では接触状態遷移に伴って roll 軸に対する物体姿勢が大きく変化して，最終的に物体への指の接触に失敗して物体を落としてしまった．このような失敗が多いため，提案した報酬設計と比較して有意に成功率が低かったのだと考えられる．物体姿勢の変化による失敗を防ぐために物体の位置姿勢の維持を促す項は必要である．w/o direction の実行例では上手く回転させることができており，提案した報酬設計と比較して有意に成功率が低くなるということは確認されたなかった．w/o direction では指の先端を物体に突き刺すように接触させることで把持を維持する．このような接触の仕方は点接触に近く，提案した報酬設計で学習される面接触と比較すると，把持の安定性が低下する．その結果，外力への頑健性が低下する可能性がある．以上の点から，面接触によって把持を維持する方が好ましい．したがって，

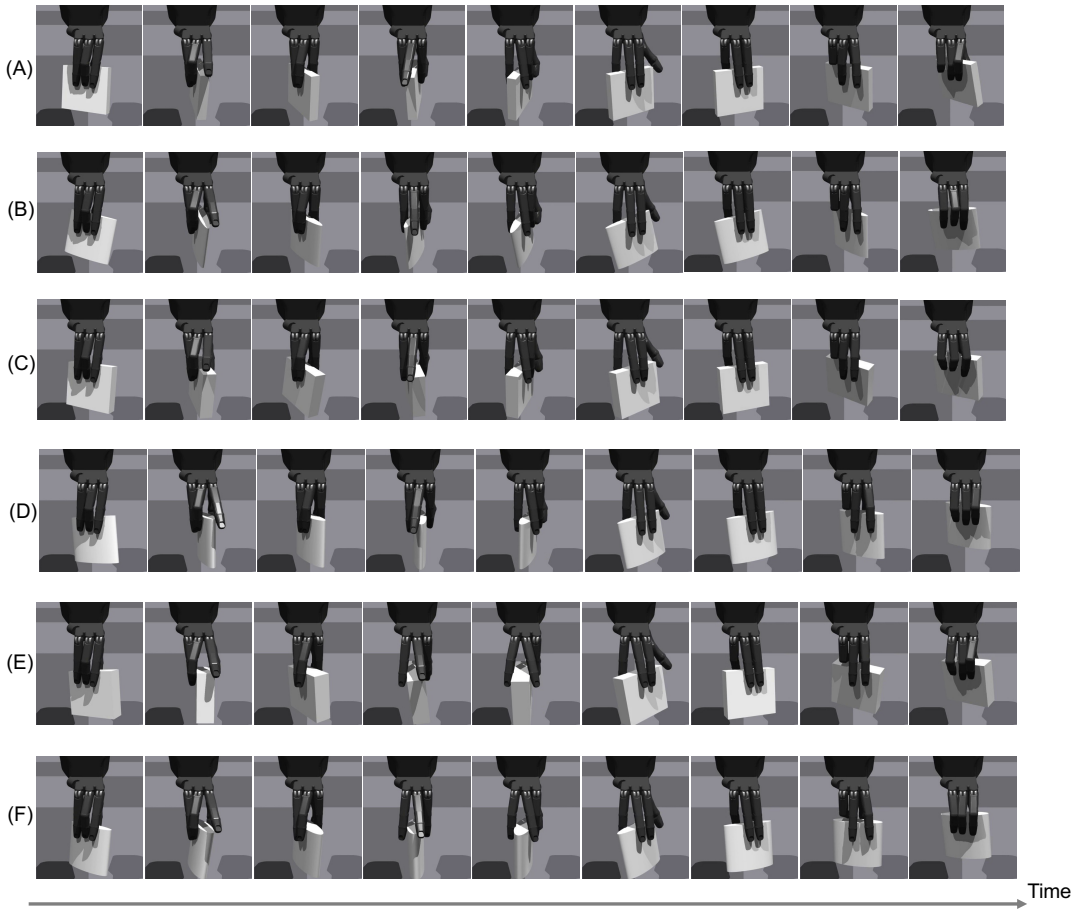


図 5.10　異なる大きさ・形状の物体に対する実行例．(A),(B),(C),(D),(E),(F) はそれぞれ深さ，ϵ_2 が $(2, 10^{-6}), (2, 1), (2.75, 10^{-6}), (2.75, 1), (3.5, 10^{-6}), (3.5, 1)$ の物体への実行例．

外力への頑健性の観点から，指の物体への面接触を促す項は必要である．

5.5.3　潜在変数の可視化

物体の形状ごとの特徴を学習できているかどうかを検証するために，潜在変数 $\hat{z}_t$ を t-SNE [173] を用いることで可視化して，その分布を確認した (図 5.14)．物体には深さが $(2, 2.75, 3.5)$cm，形状パラメータ ϵ_2 が $(0, 0.5, 1)$ の組み合わせのものを用いた．PolicyA,B,C,D を 100 イテレーション実行した際の潜在変数を図示した．物体形状が近い場合の点同士が近くに分布していることが確認できる．このことから物体形状の特徴を学習できていることが示唆される．この特徴の学習によって物体形状の変化に頑健なスキルが学習されたのだと考えられる．

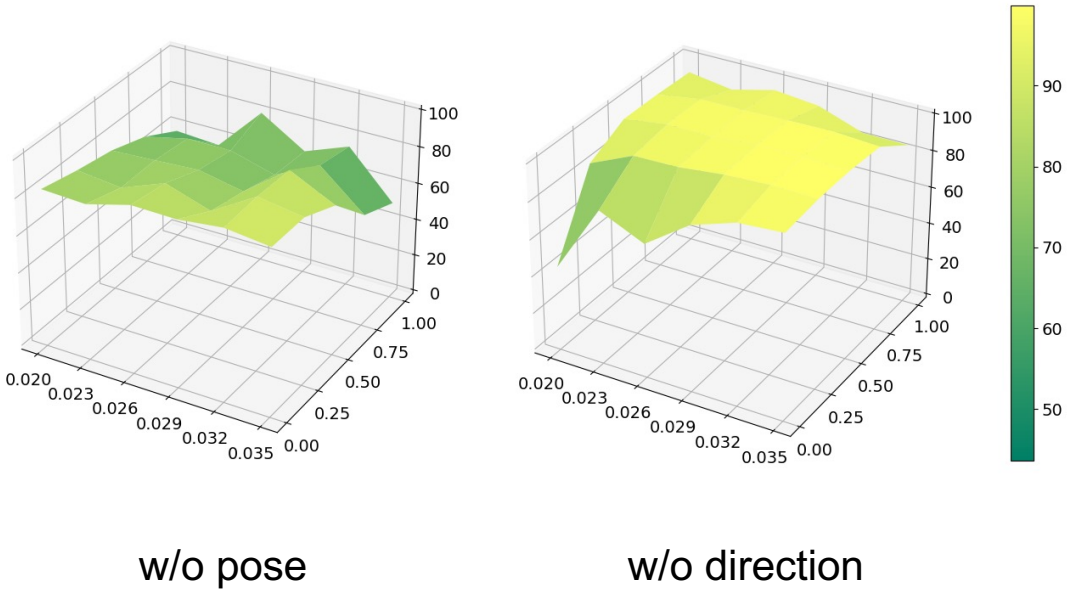


図 5.11　w/o pose と w/o direction における成功率を示した図．底面は物体の深さと形状を表している．縦軸は成功率を表している．

$*$: $p < 0.05$

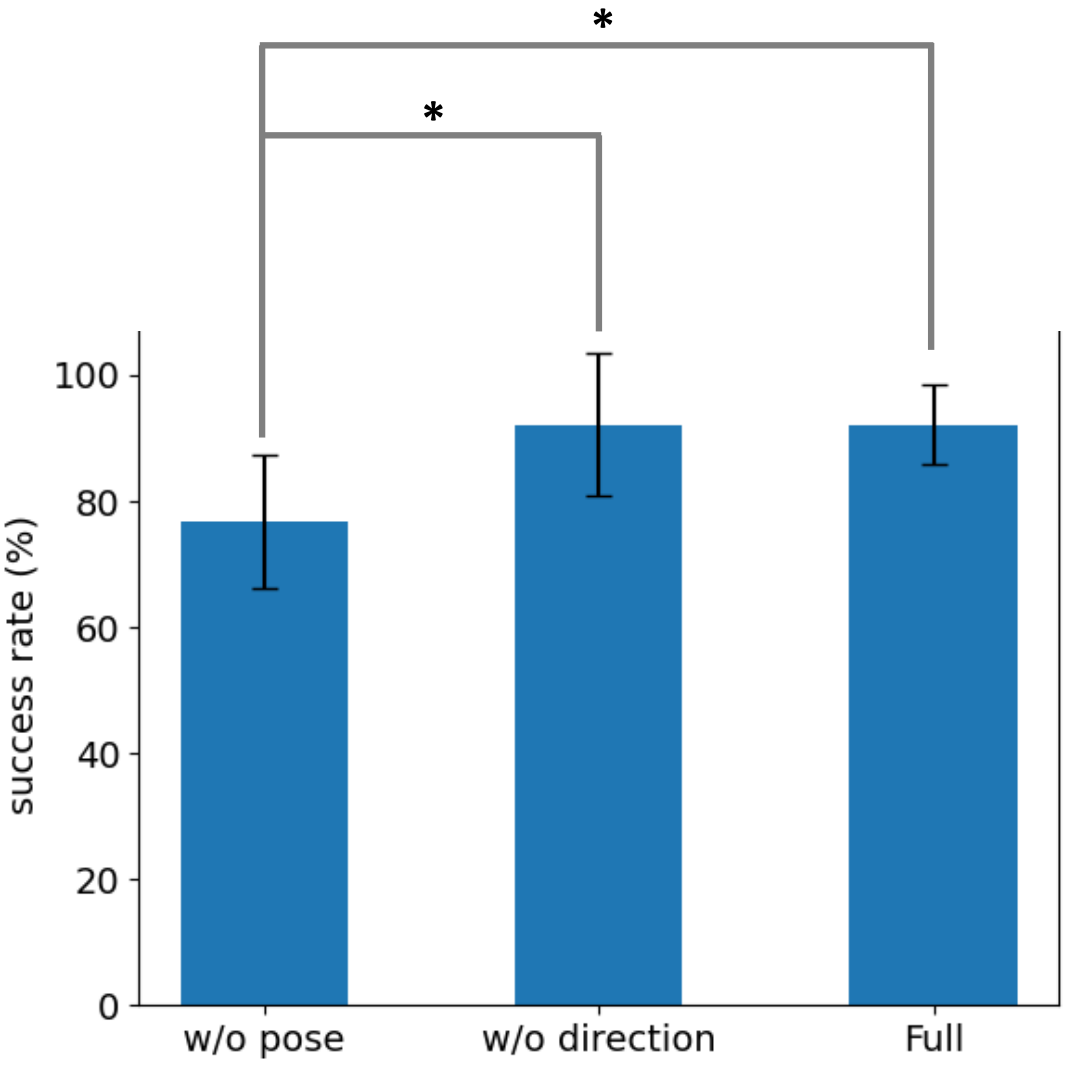


図 5.12　各条件における平均成功率を示した図.

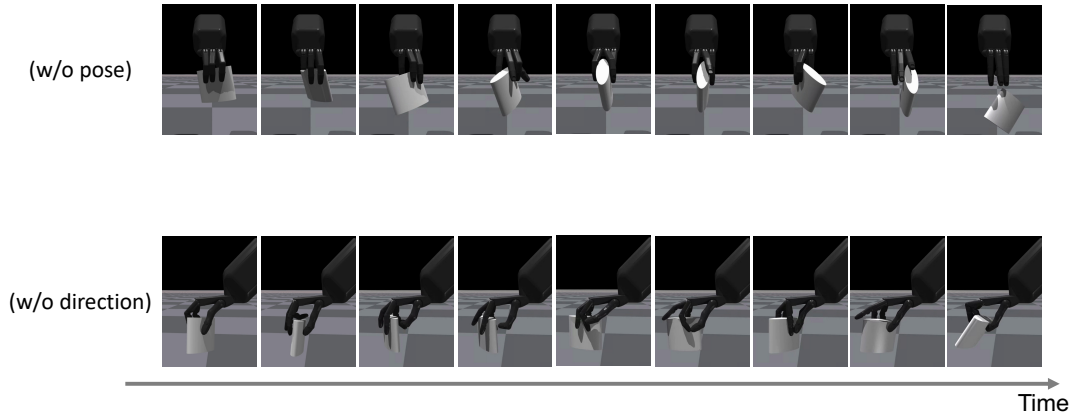


図 5.13　w/o pose と w/o direction での実行例.

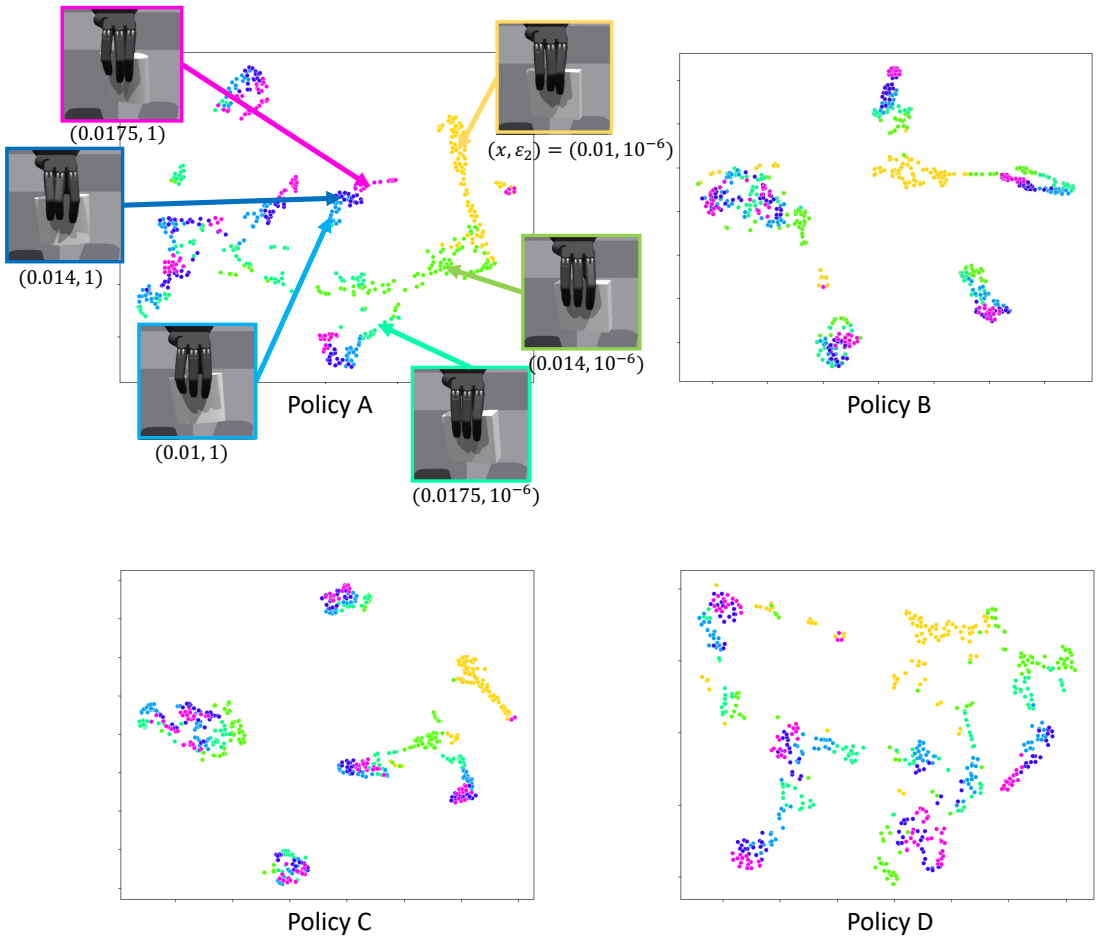


図 5.14　PolicyA,B,C,D 実行時の潜在変数の可視化の例．図中の点は 6 つの異なる形状の物体を用いたスキルの実行中に収集された潜在変数 $\hat{z}_t$ を表す．物体の深さと ϵ_2 は各画像の下に書かれている．異なる色は異なるオブジェクトに対応する．矢印の部分に見られるように同じ色の点は近くに集まっている．

5.6 議論

5.6.1 実験結果に対する考察

本論文では，把持プリミティブの達成までを行う in-hand manipulation を実行することを目的として，指先と物体の接触状態の遷移から導出されたプリミティブスキルの組み合わせによる手法を提案した．実験の結果，ベースラインでは手の中での回転と目的とする把持の達成の両方を満たすことができなかったが，本手法では両方を満たせることが確認された．操作後の指配置が既知ではない既存手法と異なり，本論文では操作後の指配置を既知と仮定した．指配置が既知であっても，既存手法で対象とした操作を学習することは長期動作の学習になるため非常に難しい．本論文では指配置の遷移に着目した動作分割を導入することで学習を簡単にすることができた．提案手法のような動作分割は学習時の探索空間を削減できるため，本実験で対象とした操作以外の状態行動空間の広い動作が含まれる操作に対しても有効である．例えば，本実験で対象とした操作後に物体が手のひらに近づくように操作して握力把持を実現するような，さらなる高度な操作にも適用可能であると考えられる．

潜在変数の可視化の結果から関節角度の情報から物体の形状を認識して適応的に行動を変えていることが示唆された．一方で，関節角度の情報からだけでは認識できない情報もある．その例として，物体の roll 方向の回転が挙げられる．この方向の回転を認識できない場合，この回転が大きくなっていき物体姿勢が崩れて手から落下する．さらに，物体の目標姿勢が決まっている操作の場合にはその姿勢を達成することが困難となる．本論文のように接触状態の遷移のみに着目した場合，物体の姿勢が所望の姿勢になっていることが保証されない．そのため，目標姿勢がある場合には物体姿勢の認識が可能な観測を入力として姿勢を調整する行動を出力するスキルを追加することが重要となる．物体姿勢の認識には既存手法 [13, 59, 120] のように視覚情報を用いることで対処することが可能である．

本実験では横方向以外にも縦方向にも長い物体を対象として実験をした．実際にはそれ以外の形状の物体も日常生活ではよく使用される．例えば，倒れた角柱や円柱のような物体が挙げられる．このような形状の物体に対しても本手法が適用できる可能性がある．どのような形状の物体まで本手法が適用できるかを検証することは今後の展望である．

本研究では実機実験は行わなかったが，学習したスキルは少しの工夫により実世界で適用可能になると考えられる．これは，実世界でも動作可能なこれまでの研究に従って，学習時に Domain randomization や teacher-student learning による Adaptation を行なったからである．これらの工夫によって sim2real gap をほぼ克服している可能性がある．更なる工夫として，実機の物理パラメータ同定や関節に過度な力をかけないための罰則の追加などが挙げられる．この検証は今後の展望である．

5.6.2 再利用可能な動作プリミティブの組み合わせによる多様な操作の実現

本論文では，active-force closure で把持された物体を半回転させて最初の active-force closure を達成するという基礎的な操作のみを対象として，動作プリミティブを設計した．把持プリミティブは active-force closure 以外にも存在するため，より多様な in-hand manipulation を行うには，少なくとも active-force closure から任意の把持プリミティブへ持ち替える操作分のスキルが必要である．一般に広い状態行動空間を探索する必要のある in-hand manipulation の学習には時間がかかるため，各持ち替え操作に対してスキルを用意していくのは時間的に困難である．

異なる操作において共通の指配置の遷移が出現することは多くある．そのため，ある指配置の遷移に対して設計されたスキルは一つの操作だけでなく，複数の操作で再利用可能であるはずである．こういった再利用性に着目してスキルを適切な順序で実行していくことで，様々な操作が実行可能となる．その結果，全ての操作を実行するのに必要なスキルの数を減らせる可能性がある．本論文は，そのような再利用性に着目して様々な操作を可能にするための研究の第一歩となる．

5.6.3 スキル実行順の決定

本論文では，まず接触状態の遷移グラフを考察して，その遷移に従ってスキルを実行した．全ての操作に対して遷移グラフを設計するのは手間がかかる．そのため，何らかの方法で自動で遷移グラフを構築できることが望ましい．その一つの方法として，LfO を応用して人間の実演の観察から自動的に遷移グラフを構築することが考えられる．具体的には，既存の画像認識手法 [110,174] を用いて人間の in-hand manipulation の動画から指と物体との接触や指配置を認識する方法や．仮想空間上での物体とのインタラクションから認識する方法 [155] が考えられる．

指と物体との接触や指配置を実演の画像から認識することが困難である場合も考えられる．この場合には，操作前後の把持プリミティブや物体位置姿勢の変化量を入力として，自動的に遷移グラフを構築できるような手法を開発する必要があると考えられる．例えば，階層強化学習 [175] を用いて low-level policy の実行順序を決定する high-level policy を学習するという方法や，現在の状態から適切なスキルを選択する selector [35] を学習するという方法が考えられる．物体の位置姿勢変化量は hand-object pose estimation の手法を用いることで取得することができる可能性がある [176]．これらの検証は今後の展望である．

5.6.4 他の把持プリミティブ実現への応用

前章では in-hand manipulation が必要である把持プリミティブの passive-form closure や non-closure のスキルは設計しなかったが，本章で提案したプリミティブを組み合わせ

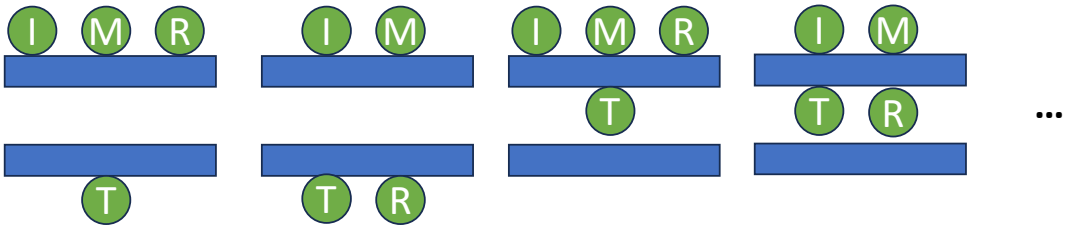


図 5.15　棒形状物体が 2 本ある場合の接触状態候補の例.

ることでこれらのプリミティブを実現できる可能性がある．例えば，物体の位置姿勢を変化させる in-hand manipulation のプリミティブを用いて，active-force closure の状態から手のひらに物体が接触するまで物体位置を変化させることで passive-form closure が実現できる．また，passive-form closure の状態から親指を detach することで non-closure へ遷移できる可能性もある．このように，in-hand manipulation のプリミティブを組み合わせることで実現できなかった把持プリミティブへの到達が期待できる．

はさみや箸に対する把持の実現に関しても in-hand manipulation が必要であるが，本論文で提案したプリミティブだけでは実現することができない．これを実現するためには，提案プリミティブを棒形状物体が 2 本ある場合のプリミティブへと拡張する必要がある．このためには，本論文の考察と同様に，棒形状物体が 2 本ある場合の接触状態候補 (図 5.15) を列挙してから，それらに対する遷移を導出する必要があると考えられる．このようなプリミティブの拡張は今後の展望である．

5.6.5　上面が棒形状以外の物体

本論文では，上面が棒形状のようなアスペクト比が高い物体を対象としてプリミティブを設計した．そのため，本論文で設計したプリミティブは，横長形状のものが多く日常生活でよく用いる工具は本手法で対応可能ではあるが，アスペクト比が低いボール，立方体のような物体に適用することはできない．そのような形状に関しては，つまめる程度の物体が多く，これは既存手法 [16, 59] で対応できる可能性がある．このようにアスペクト比に応じて用いる戦略を変えるということが一つの解決策として考えられる．

5.7　おわりに

手の中での物体の操作後に所望の把持を実現する in-hand manipulation の技能を獲得することを目的として，APriCoT (Action Primitives based on Contact-state Transition) を提案した．実験の結果，本手法では in-hand manipulation に必要な手の中での回転と目的とする把持の達成の両方を満たせることが確認された．これは接触状態遷移に着目して操作を短期的な動作に分割したことが有効であったのだと考えられる．また，潜在変数の可視化の結果から関節角度の情報から物体の形状を認識して適応的に行動を変えているこ

とが示唆された．本手法は異なる操作に対しても適用できる可能性がある．他の接触状態の遷移に対してもスキルを用意できれば多くの操作が実現できる．本研究はそのような研究の第一歩となる．

第 6 章

Compliant Manipulation に関するスキルライブラリの設計

6.1 目的とアプローチ

家庭内作業の多くでは．引き出しやドアを開けるなどの物理的に拘束された環境下で対象物を操作する必要がある．このような操作を行うロボットシステムは対象物や環境を傷つけないことを保証しなければならない．そのため，ロボットは環境から与えられる力，すなわち拘束力に基づいて実行中の手の軌道を調整する必要がある。このような操作を compliant manipulation と呼ぶ [50]．家庭環境では予測できない量の操作が存在するため，このような操作に対する汎用的なコントローラが家庭用ロボットの実現に期待される．そこで，本論文では強化学習を用いて様々な未知の操作に汎用的な compliant manipulation のスキル設計を行う．

実は，compliant manipulation が必要な操作は物理的拘束に基づいて有限個の操作プリミティブに分類することができる [27]．物理的拘束によって物体が動くことが可能もしくは不可能な方向が決定するが，この方向に関して共通の集合を持つ操作を一つの操作プリミティブとしている．例えば，引き出し開けや板を引く，棒を引くといった操作は物体の許容される運動方向がある直線上に拘束されているため，同一の操作プリミティブに属する．物体が許容されない方向に動こうとすると拘束によって物体に拘束力が発生する．したがって，物理的拘束に基づいて分類された操作は拘束力方向が共通しているという特徴を持っていることが分かる．

本論文では拘束力を用いて物体が動ける方向 (許容方向) を推定する Constraint-aware policy を提案する．このスキルを単一の環境と報酬を用いて制約群における未知の操作にも汎化できるように学習を行う (図 6.1 右)．この環境と報酬は，ロボットハンドと物体が一体となって動き，摩擦力などの内力が相殺された複合体とみなすことができる単一システム条件 (図 6.1 左) を仮定して設計されている．そのため，物体に作用する拘束力を得ることができる．この環境は，compliant manipulation に重要な物理的拘束力の共通特性を抽出することで実世界の操作を単純化したものとして設計されている．この仮定は，手をゆっくり動かすといった実行設計で容易に満たすことができるため，ロボットでの実行時

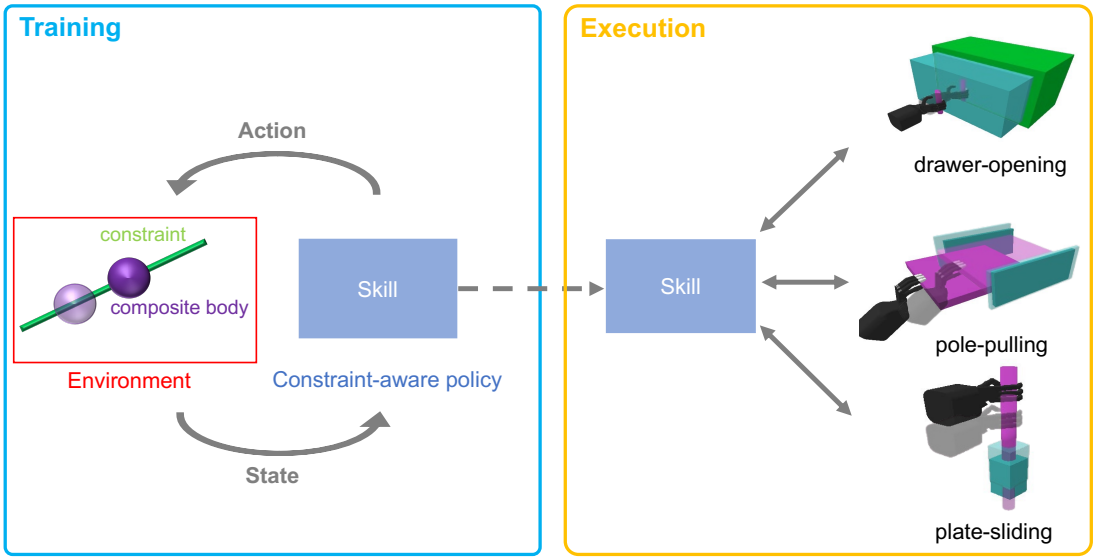


図 6.1　Constraint-aware policy に関する概念図.

でも近似的に満たすことが可能である．単一システム条件下では，拘束力の大きさが小さくなるに従って許容方向の推定誤差が小さくなるため，報酬は大きさのみを用いて計算される．

6.2　本研究の対象

本論文では，家庭環境における代表的な制約条件である prismatic 関節または revolute 関節を持つ操作プリミティブのスキルを設計する．ここでは，[27] に倣って，前者の操作プリミティブを PTG3，後者の操作プリミティブを PTG5 と呼ぶ．prismatic または revolute 関節の制約下では，物体はそれぞれ 1 自由度の並進と回転を持つ．図 6.2 のように，PTG3 に含まれる操作としては引き出し開け，棒引き，板引きが挙げられる．これらの操作では，物体の運動は一軸の並進に拘束される．PTG5 に含まれる操作としてはドア開け，ハンドル回しが挙げられる．これらの操作では，物体の運動は一軸の回転に拘束される．

同じ操作プリミティブに含まれる操作には共通の特徴がある．物体が非許容方向に移動しようとした時に，その方向とは逆に物体に大きな力が働く．この非許容方向の集合が操作プリミティブ内の操作間で共通しているため，物体にかかる力の方向も共通となる．この力を利用することで compliant manipulation が実現できるため，このような力の特徴に基づいてスキルを設計することで，同一プリミティブ内の操作に汎用的なスキルを実現する．

実は，PTG3 と PTG5 には共通する点がある．PTG5 における円運動は無限小の直線運動とみなすことができる．そのため，PTG3 も PTG5 も直線運動を行うことが共通している．したがって，PTG3 用のスキルを PTG5 にも適用できる可能性がある．そこ

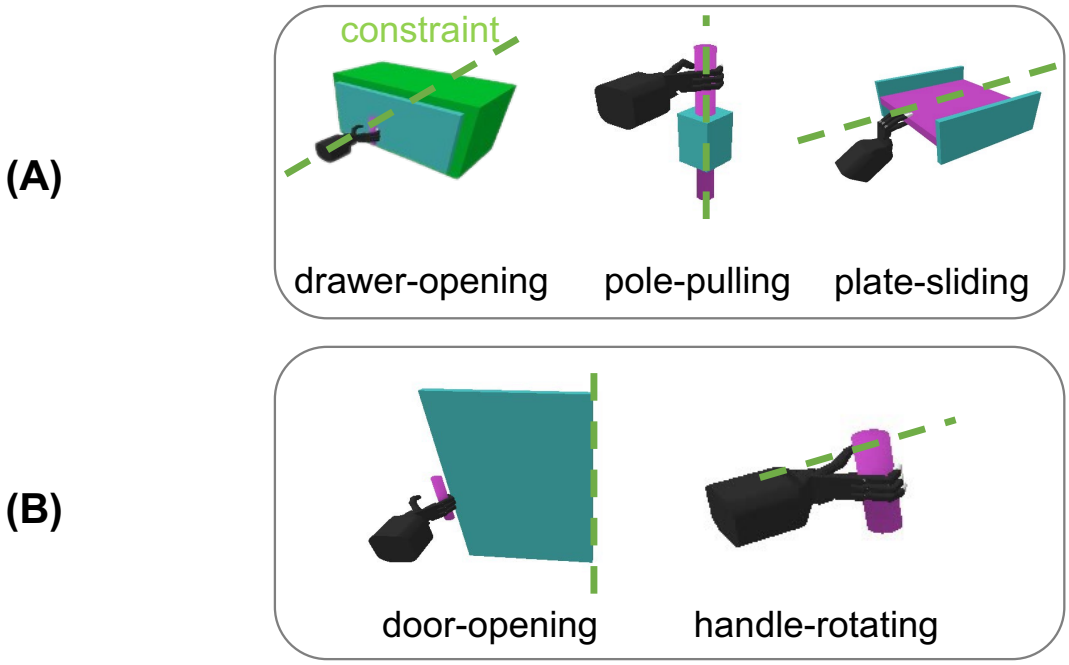


図 6.2　Compliant manipulation の例．(A) が prismatic 関節で拘束された操作の例，(B) が revolute 関節で拘束された操作の例である．

で，PTG3 用のスキルを PTG5 でも再利用できるようにスキルを設計することを目指す．PTG3 とは異なる点として，PTG5 に適用するためには運動中に単一システム条件を満たすように物体の回転に応じて手の姿勢を変化させる必要がある．この回転操作を PTG3 用のスキルと組み合わせることで PTG5 用のスキルを実現する．

6.3　仮定

本論文では，提案する constraint-aware policy の動作環境として以下の仮定を置く．

1. ロボットハンドと物体は一体となって動き，両者間の内力は相殺される (単一システム条件)
2. 操作される物体にかかる慣性力は無視できる
3. 関節機構の摩擦が十分に弱く，操作対象物が目的の軌道に沿って滑らかに移動できる
4. ロボットハンドの作業平面と回転軸の方向が既知であり，ロボットハンドと操作対象物は既知の平面上を移動できる

これらの仮定は現実的に満たすことが可能な仮定である．仮定 1,2 に関しては，操作の設計によって満たすことができる．仮定 1 は，ロボットハンドと物体にかかるトルクを減少させる追加のスキルによって満たすことができる．仮定 2 は，物体をゆっくり動かすことで満たすことができる．仮定 3 は，多くの家庭用物体によって満たされる．なぜなら，家

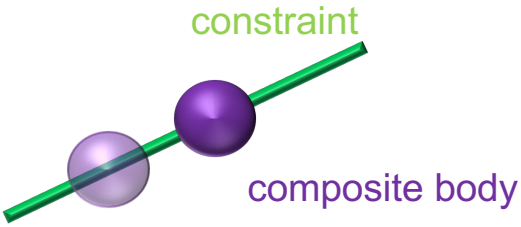


図 6.3　学習に用いる環境の概念図．単一の複合体（紫色の球体）と prismatic 関節（緑色の線）で構成される．

庭用物体は人間が簡単に扱えるように設計されているからである．最後に，本研究では，家庭環境を代表する 1 つの prismatic 関節または revolute 関節しか持たない物体に着目しているため，仮定 4 に関しては作業平面を容易に求めることができる．このような作業平面は，LfO の枠組みによって人間の手の軌跡から計算することが可能である [85]．もしくは，ロボットのカメラから得られた三次元モデルに人間が手動で注釈をつけることで可能である [177, 178]．

6.4　学習設計

本論文では，スキルの獲得に深層強化学習を用いる．深層強化学習では，古典制御器とは異なり，手動のパラメータ調整の手間を軽減され，認識誤差やセンサノイズなどの不確実性に対して頑健である．そのため，スキル獲得に深層強化学習を採用する．

6.4.1　環境

学習環境は単一システム条件に基づいて設計されている．この環境は単一の複合体と prismatic 関節から構成される (図 6.3)．この複合体は，単一システム条件下でのロボットハンドと操作対象物を表している．

この環境では，各時刻において複合体は与えられた進行方向 $\boldsymbol{d} \in \mathbb{R}^3$ に進もうとする．これによる複合体と拘束の相互作用の結果として，複合体に作用する力 $\boldsymbol{F} \in \mathbb{R}^3$ が得られる．拘束は拘束方程式として表現され，力は拘束力を含む運動方程式を解くことによって計算される [160]．単一システム条件によって，実行時のロボットハンドの手首で測定された力 $\boldsymbol{F}$ が複合体上の拘束力と同一であることは保証されている．

この環境設計は，物体同士の接触シミュレーションなど不安定な要素を考慮する必要がないため，シミュレーションコストが低いという利点がある．これによって，シミュレーション速度が向上し，学習の高速化につながる．さらに，この環境で訓練されたスキルはロボットハンド自体の特性に依存しないため，多様なロボットハンドに容易に適用できる可能性がある．

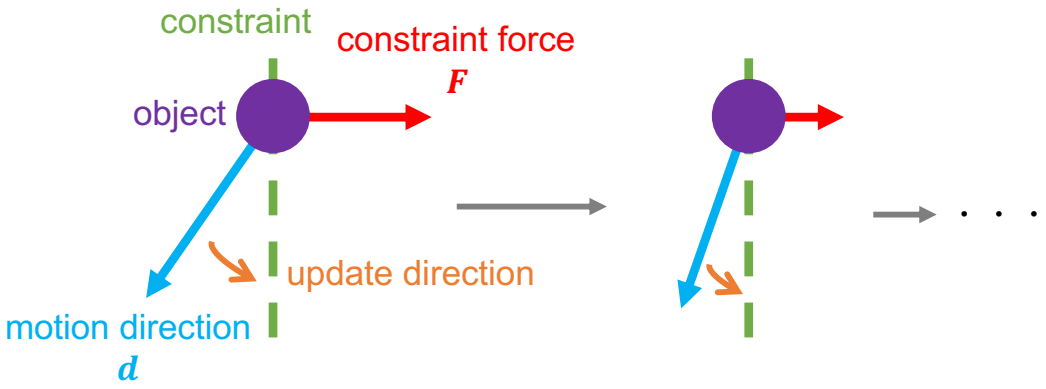


図 6.4　constraint-aware policy による運動方向の更新．紫色の円は物体，緑色の線は拘束を表す．物体が非許容方向に移動しようとすると，物体に拘束力が働く．この力が小さくなるように，運動方向が拘束力の方向に向かって修正される．

6.4.2　状態，行動，報酬の設計

状態

状態 $\boldsymbol{s}_t$ は，センサから得られた力を正規化したもの $\bar{\boldsymbol{F}}_t \in \mathbb{R}^3$ $(\bar{\boldsymbol{F}}_t = \frac{\boldsymbol{F}_t}{\|\boldsymbol{F}_t\|_2})$ と，ロボットハンドの進行方向 $\boldsymbol{d} \in \mathbb{R}^3$ から構成される．力を正規化することによって環境の変化によって生じる力の大きさの変化に対して頑健なスキルにすることができる．そのため，正規化された力ベクトルを利用することは重要である．しかし，拘束力が無視できるほど小さい場合，正規化することでセンサノイズや関節のしなりなどの様々なノイズが増幅されることに注意する必要がある．本研究では、これらの要因を無視できるほど拘束力が常に大きいと仮定する．これらの要因が無視できない場合には，あらかじめ設定した閾値よりも小さな力の大きさをゼロとして計算すればよい．

行動

行動 $\boldsymbol{a}_t \in \mathbb{R}^3$ は運動方向を修正する操作として定義される．$\boldsymbol{d}_t$ と $\boldsymbol{a}_t$ が与えられた時，次時刻 $t+1$ の進行方向は以下の式のように表される．

$$\boldsymbol{d}_{t+1} = \frac{\boldsymbol{d}_t + \boldsymbol{a}_t}{\|\boldsymbol{d}_t + \boldsymbol{a}_t\|_2} \tag{6.1}$$

物体が非許容方向に移動しようとすると，拘束力が物体に作用する．constraint-aware policy は拘束力が減少するように，この力の方向に向かって運動方向を修正する必要がある．図 6.4 のように、最適なスキルによる運動方向の更新では，物体と制約の相互作用によって生じる拘束力の大きさ $\|\boldsymbol{F}\|_2$ を最小化していくことが保証される．したがって、拘束力の方向を用いて運動方向を適切に修正することができる．なお，拘束力の方向は，関節機構の摩擦が拘束力より十分弱いという仮定 3 のもとで求めることができる．

報酬

最適なスキルを学習するためには，拘束条件に基づいて適切な報酬関数を設定する必要がある．そこで，運動方向が制約に沿わない場合を考える．PTG3 と PTG5 では、ロボットハンドが拘束に沿って移動しない場合，物体に物理的拘束による拘束力が働く．この力は，運動方向が拘束に沿っているときに最小化されます．従って，以下のように拘束力の大きさ $\|\boldsymbol{F}_t\|_2$ に基づいた報酬 r_t を提案する．

$$r_t = -\|\boldsymbol{F}_t\|_2 \tag{6.2}$$

6.4.3 単一システム条件のための工夫

Constraint-aware policy は，単一システム条件を仮定して学習され，さらに同様に実行もされる．単一システム条件を満たすためには，ロボットハンドと物体との間の相対的な位置と姿勢を維持しなければならない．この条件を満たすための 2 つの主な課題として，指先の滑りとロボットハンドと物体の接触不足が挙げられる．これらの課題に対しての工夫を以下で説明する．

指先の滑りへの対処

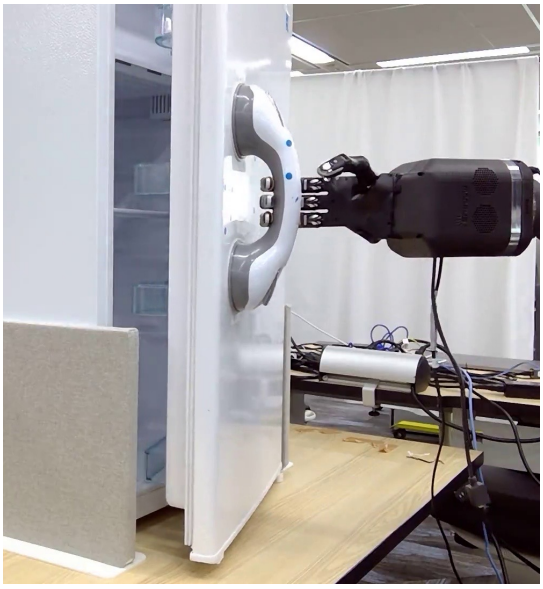
単一システム条件が崩れる要因の一つとして，大きな撃力によってロボットの指先が操作対象物で滑ってしまうことが挙げられる．この撃力は，主にロボットハンドが大きな並進量によって非許容方向に移動しようとした場合に発生する．そこで，この撃力を防ぐために，ロボットハンドがゆっくりと動くようにロボット制御系を実装する．

また，操作対象の向きが変化する PTG5 において，手の向きが一定であると指先の滑りが発生しやすい．そこで，以下のように運動方向の変化に応じて手の向きを変えることで指先の滑りを回避する．q_t を時刻 t におけるワールド座標系での手の向きを表すクォータニオンとして定義し，以下のように q_{t+1} を計算する．

$$\boldsymbol{q}_{t+1} = \Delta\boldsymbol{q}_t \otimes \boldsymbol{q}_t \tag{6.3}$$

ここで，$\Delta\boldsymbol{q}_t$ は，$\boldsymbol{d}_t$ と $\boldsymbol{d}_{t+1}$ の外積を中心に，$\boldsymbol{d}_t$ と $\boldsymbol{d}_{t+1}$ の角度を回転するクォータニオンを表す．この戦略は，必ずしも物体の向きに完全に連動した手の向きの変化を保証するものではない．手と操作される物体との間の相対的な向きが厳密に固定されていない場合にのみ適用することができる．その例として，lazy-closure を用いたドア開閉がある (図 6.5)．このように，手と操作対象との相対的な向きが厳密には固定されていなくても，Lazy-closure を用いることで接触領域が一定に保たれ，安定した操作でドアを開けることができる．

しかし，ハンドル回しのように，ハンドと操作対象物体との相対的な向きが厳密に固定されている場合には，ハンドの向きをより正確に変化させる方法が必要となる．このような場合には，ロボットハンドと操作対象物の相対姿勢が変化することでトルクが発生し，このトルクが拘束力として得らてしまうために推定移動方向の修正に失敗する．その結果，



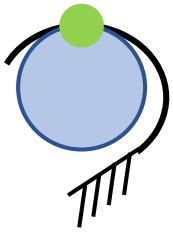


図 6.5　Lazy-closure によるドア開け．左側に実際の操作の図を示す．右側はロボットハンドが lazy-closure でハンドルを把持する様子を図式化したもので，青と緑の丸はそれぞれハンドルと接触点を示し，黒い円弧はハンドである．

ロボットハンドがハンドル上を滑ってしまうということが起きる (図 6.6)．これに対処するために，ロボットハンドが操作対象物に厳密に連動して回転するような追加のスキルが必要となる．このスキルは，操作対象物体への接触点群の中心を回転中心として手を適切に回転できるように設計される．回転軸は作業平面の法線に対応する．この追加のスキルは，回転軸周りのトルク τ を用いて，以下の処理により各時刻における適切な回転量 w を推定する．

1. ロボットハンドを回転軸周りに w だけ回転
2. 以下のように w の調整項の Δw を決定 ($\beta > 0$):

$$\begin{cases} \Delta w = 0 & (\|\tau\| \leq \alpha) \\ \Delta w = \beta & (\tau > \alpha) \\ \Delta w = -\beta & (\tau < \alpha) \end{cases}$$

3. w を $w + \Delta w$ に更新

w の初期値は $w = \frac{v}{r}$ で計算される．ここで，r は人間の実演から得られた回転半径，v は各時刻における並進量である．この方針は，w と一時刻の並進に適した回転量との過不足を計算することができる．(図 6.7) は constraint-aware policy と追加のスキルを組み合わせた際の実行の流れである．ロボットハンドは並進と回転を同時に行った後，tau が $alpha$ より大きい場合，tau が $alpha$ より小さくなるまで追加のスキルを実行して拘束力以外の力を最小化する．そうでない場合は，constraint-aware policy のみを実行する．このスキルではトルクを最小化するためのパラメータを手動で調整する必要があるが，これらのパラメータは古典制御器の制御パラメータより解釈性が高く調整が容易となる．

図 6.6　ロボットハンドとハンドルとの相対姿勢の変化によるハンドル回しの失敗.

ロボットハンドと物体の接触

単一システム条件を維持するためには，操作対象物とロボットハンドは操作中に常に接触していなければならない．接触が保証されるのは，ロボットの手首にある力センサによって非零の拘束力が測定された場合である．そのため，接触条件は事前に与えられた非零の拘束力を維持することによって保証できる．したがって，constraint-aware policy に与える拘束力 $\boldsymbol{F}$ は，生のセンサ値 $\boldsymbol{F}_s$ から事前に定義された予備力 $\boldsymbol{F}_d$ を引いた値（すなわち，$\boldsymbol{F} = \boldsymbol{F}_s - \boldsymbol{F}_d$）として定義する．

ロボットハンドの位置の変位は，ロボットハンドと物体との間で許容方向と非許容方向との変位に分類される．ハンドが非許容方向に移動した場合，ハンドは物体に衝突する．この場合，単一システムの条件が維持される．推定された運動方向 $\boldsymbol{d}$ がハンドと物体の間の非許容方向から外れている場合，ハンドは物体から離れ単一システム条件は破られる．本研究では，運動方向は常にハンドと物体の間の非許容方向の範囲内にあると仮定する．

6.5　シミュレーション実験

提案した constraint-aware policy の性能を，運動方向の誤差が存在する場合に評価する．また，共通の拘束を持つ操作に対する提案スキルの汎化能力を検証する．

6.5.1　準備

学習環境は PyBullet シミュレータで実装し，constraint-aware policy は強化学習フレームワークである Microsoft Bonsai[*1]を用いて学習された．

環境のエピソードの長さは 5 タイムステップ ($T = 5$) とした。センサの不確実性を考慮するために，最初のタイムステップで観測された力と運動方向に一様分布からサンプリングされるノイズを加えた．スキルの学習には PPO [165] を用いた．バッチサイズは 6000，学習率は 5×10^{-5} とした．スキル pi_theta は，2 つの 256 次元隠れ層を持つ多層パーセプトロンによってパラメータ化されている．活性化関数には，hyperbolic tangent($tanh$) を用いた．

[*1] https://www.microsoft.com/en-us/ai/autonomous-systems-project-bonsai

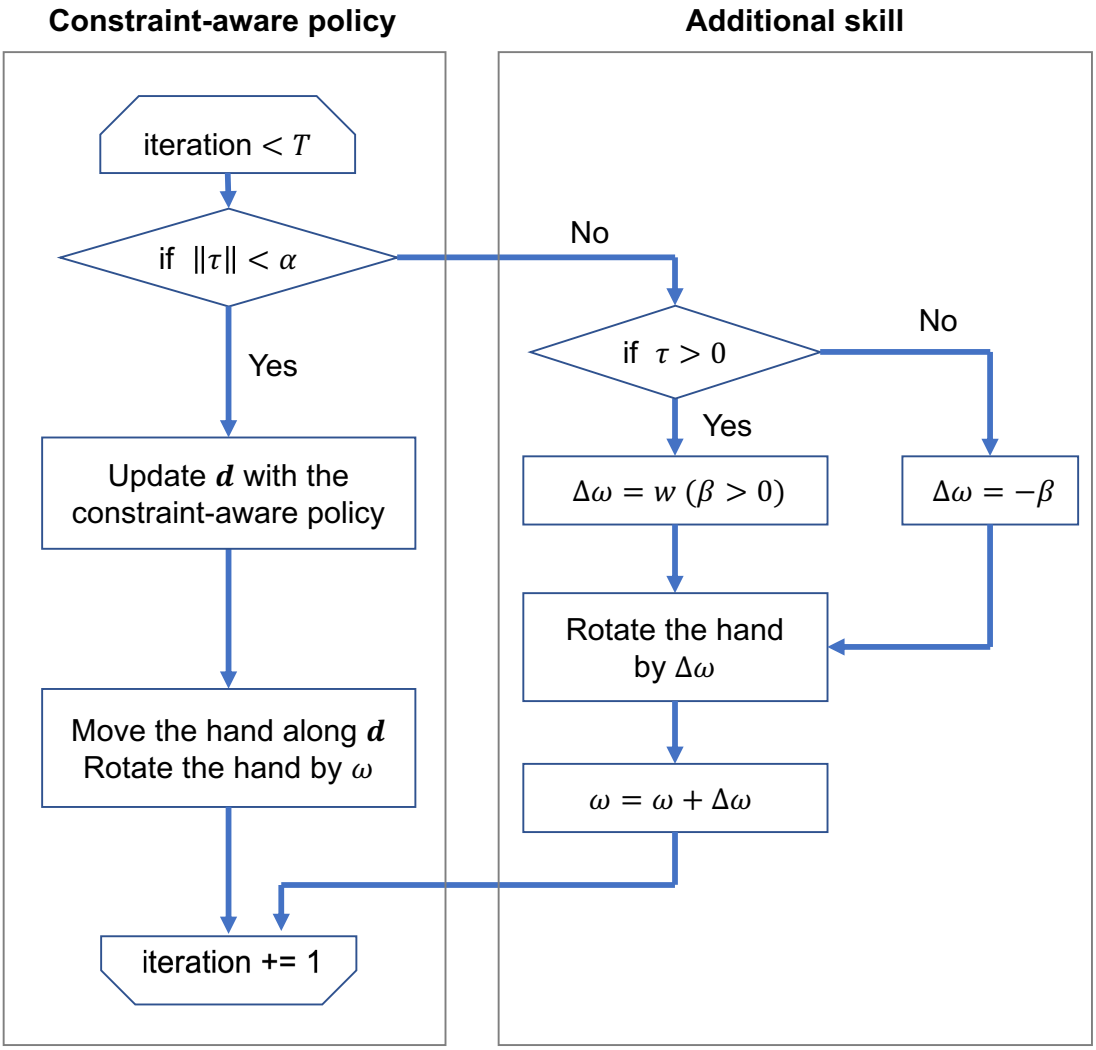


図 6.7　ハンドと操作対象物体との相対的な向きが厳密に固定されている場合の追加のスキル．回転軸周りのトルク $\|\tau\|$ が閾値 β より小さければ一般化された方針が実行され，そうでなければ追加の方針が実行される．T はエピソードの長さである．

学習された方針は PyBullet シミュレータを用いてテストされた．運動方向は制御ループ内で 100ms 毎に更新され，各タイムステップでロボットハンドは運動方向に沿って 1 cm 移動した．各評価において，ロボットハンドが物体を把持するところから始まる．把持は 4 章で学習されたスキルを用いて行われた．

6.5.2　運動方向誤差に対する性能

シミュレーションにおける引き出し開けを用いて，提案スキルの性能を評価した．引き出しは prismatic 関節で拘束され，引き出しが 25cm 移動した時点でエピソードが完了したとみなされた．引き出しのハンドルは lazy-closure を用いて把持した．

図 6.8 に結果を示す．初期運動方向は許容方向に 30°(図 8A) または −30°(図 8B) の誤

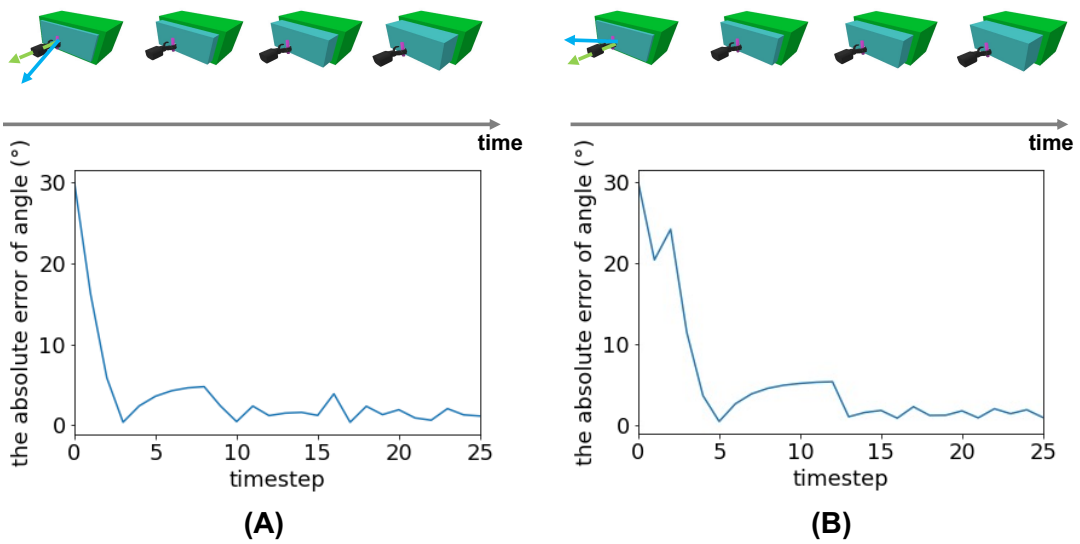


図 6.8　運動方向誤差がある場合のスキルの性能．(A) は許容方向から 30° ずらして初期運動方向を設定したもの，(B) は −30° ずらして初期運動方向を設定したものである．上段は引き出し開けのシミュレーション結果，下側は許容方向（緑矢印）と運動方向（青矢印）の間の相対角度の変化を示している．

差を乗せたものとした．どちらの場合も引き出しの開放は成功した．図 6.8 の折れ線は，許容拘束方向と現在の運動方向との間の相対角度の変化を表している．角度は 0° 付近に収束した．この結果は，提案スキルが，拘束力の方向から運動方向を推定できることを示している．

6.5.3　古典制御器との汎用性の比較

様々な操作に対する汎用性を評価するために，最新の古典制御器 [24] と比較した．実験には，(A) 引き出しを開ける，(B) 板を引く，(C) 棒を引くという三つの操作を用いた．これらの操作に用いる force-exertion type はそれぞれ lazy-closure，active-force closure，passive-force closure であり，再把持を必要としないタイプを網羅している．この実験では，初期運動方向は制約方向から 5° 刻みで −30° から 30° の範囲で誤差を乗せた．古典制御器の制御パラメータに関しては，引き出しを開ける操作を用いて手動で調整し，それ以外の操作では同じパラメータを使用した．

結果を表 6.1 に示す．提案した constraint-aware policy では 3 つの操作において全ての試行で成功した．古典制御器では，引き出し開けでは全ての試行で成功したが，パラメータチューニングに用いなかった板を引く操作と棒を引く操作では失敗した．この結果は、提案スキルが古典制御器よりも 3 つの操作に対してより汎用的なスキルとなっていることを示している．

古典制御器による操作の結果の例を図 6.9 に示す (Classical-A，Classical-B，Classical-C)．古典制御器は引き出しを開けることに成功したが，板引きや棒引きには失敗した．板

表 6.1　引き出しを開ける，板を引く，棒を引くという 3 つの操作について，提案スキル (Proposed) と古典制御器 (Classical) を用いて成功した試行回数の比較.

	Drawer-opening	Plate-sliding	Pole-pulling
Classical	13/13	0/13	0/13
Proposed	13/13	13/13	13/13

引きや棒引きでは推定方向がオーバーシュートすることによって指先に大きな力かかった．その結果，ロボットハンドは把持を維持することができず，物体を操作することができなかった．これは，把持によって変化しうる力の大きさに対してパラメータを調整したためである．例えば，物体と指の衝突による関節の屈曲度合いによって大きさが変化する．この度合いに影響を与える要因の一つが，把持によって異なる関節指令値である．実際には，把持だけでなく，手と物体の摩擦係数，物体の重さ，指関節の減衰係数，センサノイズなどによっても大きさは変化する．3 つの操作に対してパラメータを調整することも可能であるが，調整には専門的な知識が必要である．また，調整には事前に複数の環境を用意する必要があり，手間がかかる．これは家庭環境のような事前に環境を想定できないような場合には不適である．

提案スキルによる操作の結果の例を図 6.10 に示す (Proposed-A，Proposed-B，Proposed-C)．古典制御器とは異なり，提案スキルは 3 つの操作全てに成功した．これは，スキルの入力が，環境や把持の変化に敏感な力の大きさの代わりに，正規化された力を含んでいるからである．正規化された力を用いることで把持の変化に対して頑健になる．

6.5.4　PTG5 への性能

提案スキルを，回転関節を含む 2 つの異なる操作（ドア開けとハンドル回し）で実行した．初期運動方向は許容方向から 15° ずらした方向に設定された．ドア開けとハンドル回しでは，それぞれハンドルは lazy-closure と passive-force closure で把持された．なお，単一システム条件を満たすために提案スキルに追加で 6.4.3 で説明した工夫を行った．

結果は図 6.11 に示すように，提案スキルが適切に運動方向を変更できることが示された．このように，提案スキルは単一システム条件下で，prismatic 関節と revolute 関節の両方を持つ操作に対して実行可能である．また，回転半径が異なる場合でも，拘束力はハンドルから回転中心に向かうという性質は共通しているため，回転半径が異なる操作にも適用可能である．

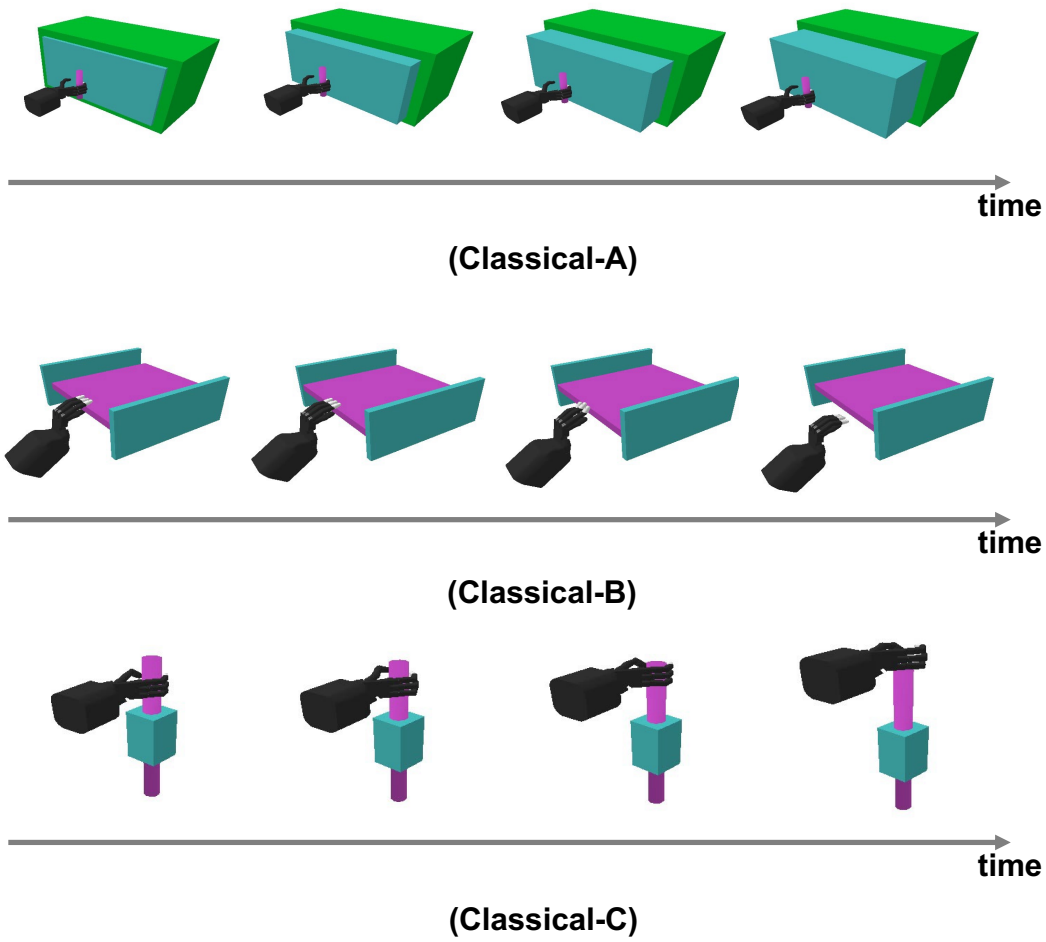


図 6.9　古典制御器による操作の結果.

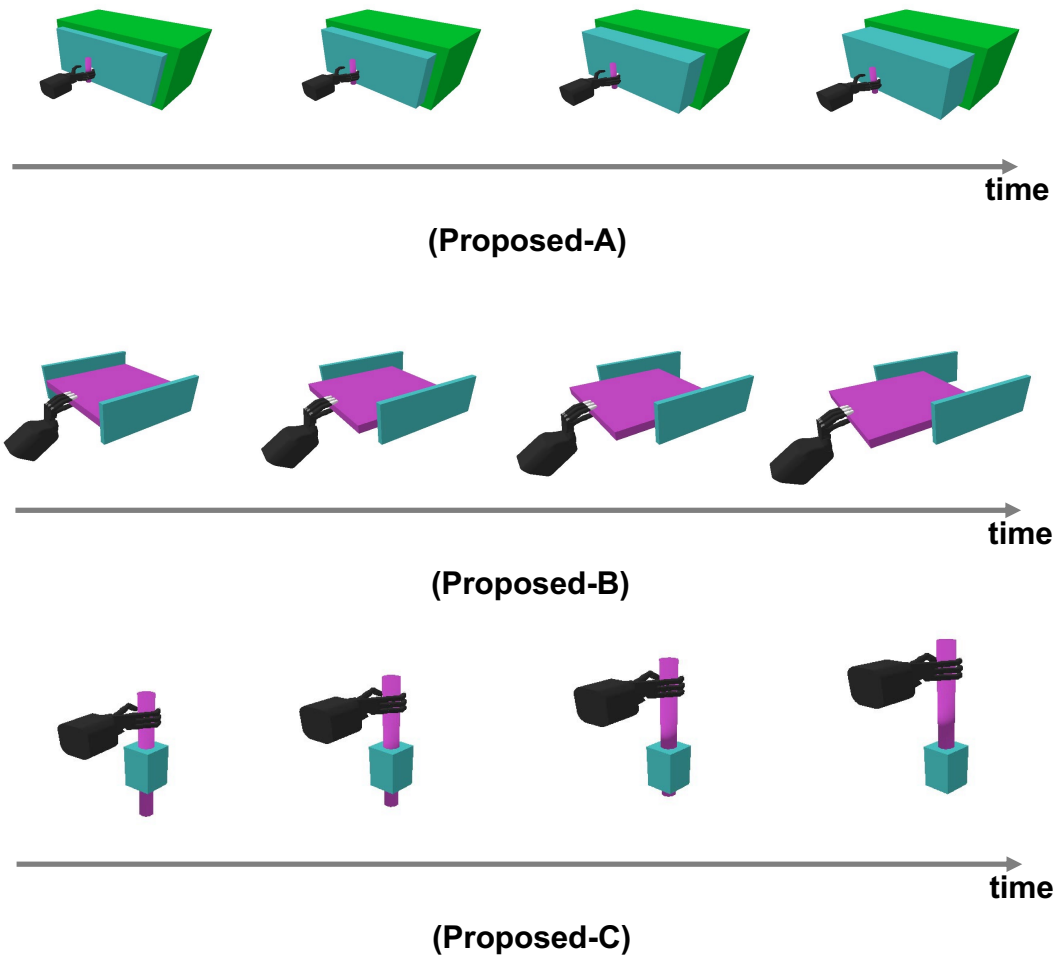


図 6.10　提案スキルによる操作の結果.

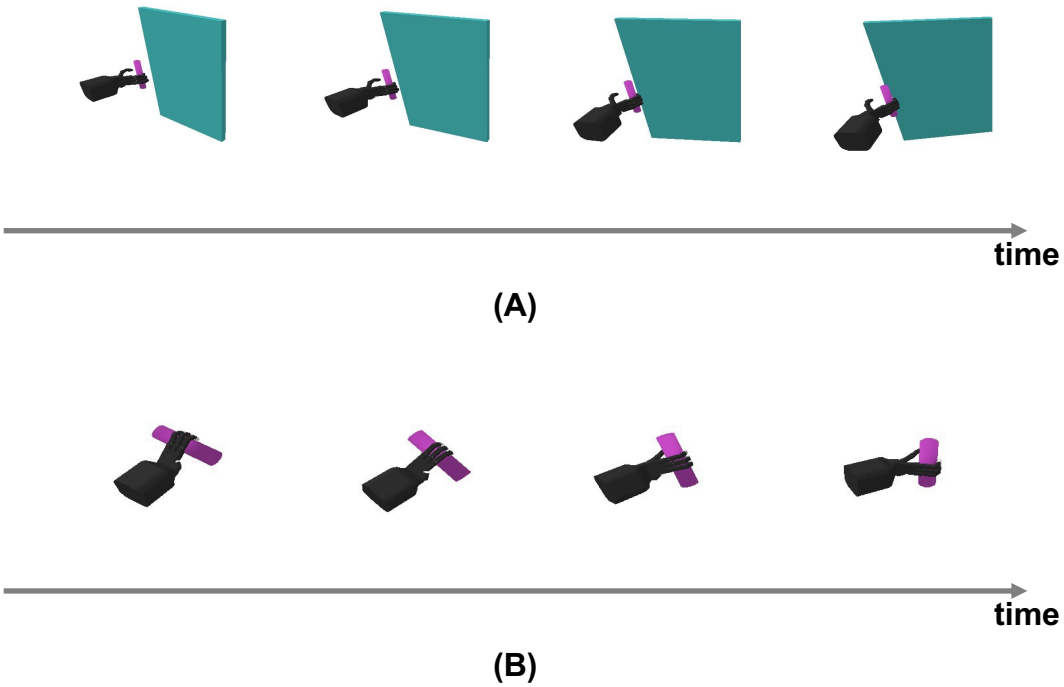


図 6.11　提案スキルによる操作の結果.

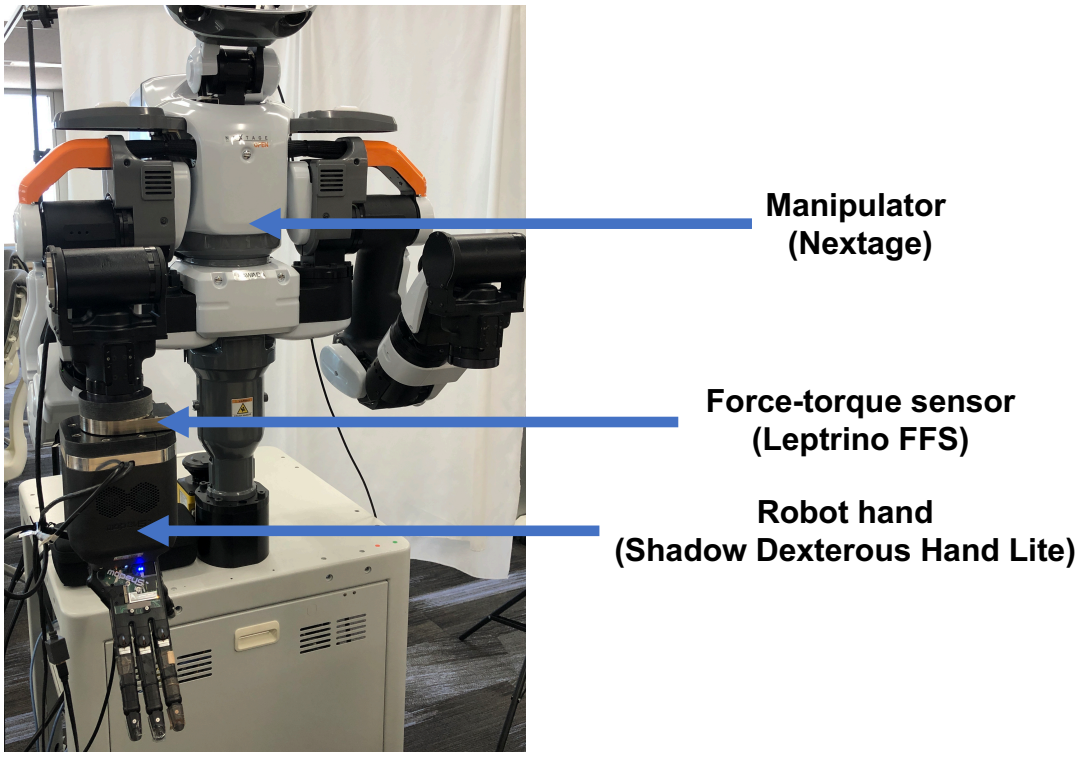


図 6.12　用いたロボットの図．腕とハンドの間に力センサが取り付けられている．

6.6　実機実験

本スキルの実世界のロボット上での実行可能性を評価する．

6.6.1　準備

実験には，腕に 6 自由度を持つ Nextage[*2]を利用した．このロボットには，4 本指のロボットハンドである Shadow dexterous hand lite[*3]を取り付けた．力センサには Leptrino FFS シリーズ[*4]を使用し，図に示すように Nextage と Shadow dexterous hand lite の間に取り付けた．

実験は二つ行われた．一つ目が引き出し開けによるスキルの性能調査である．この実験では引き出し開けを 10 回行い，その成功数を調査した．許容方向に約 $\pm 20^\circ$ の誤差を乗せた場合の試行を 5 回ずつ行った．二つ目が PTG5 への適用可能性の調査である．この実験では，ドア開けとハンドル回しが実機でも実行できるのかどうかを調査した．

[*2] https://nextage.kawadarobot.co.jp/
[*3] https://www.shadowrobot.com/dexterous-hand-series/
[*4] https://www.leptrino.co.jp/product/6axis-force-sensor

表 6.2　提案スキル (Proposed) で引き出し開けに成功した試行回数.

	Drawer-opening
Proposed	7/10

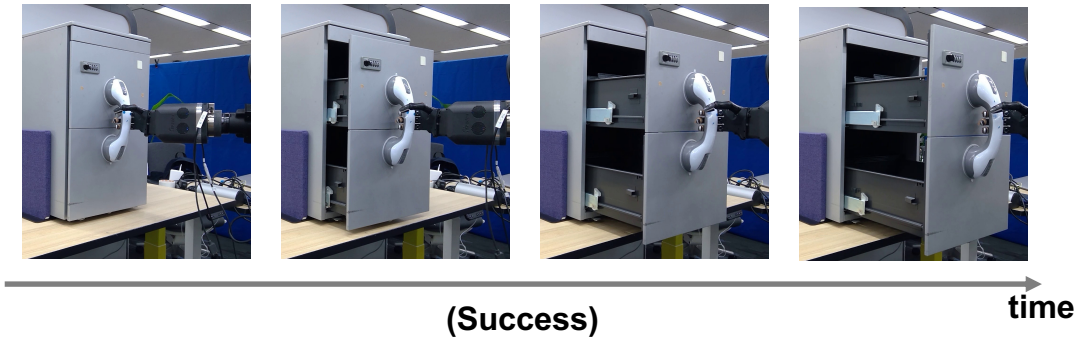
6.6.2　結果

表 6.2 に引き出し開けの成功数を示す．10 回中 7 回成功した．提案スキルは，力の大きさではなく正規化された力を用いることで，シミュレーションと現実の間のギャップを小さくしている．その結果，提案スキルでは追加の訓練なしに実世界に適用することができたのだと考えられる．

図 6.13 に提案スキルを用いた引き出し開けの成功例と失敗例を示す．また，図 6.14 に成功例の場合の許容方向の変化と力センサで得られた値の遷移を示す．左側は，操作時に使用した座標系を示しており，右上は許容方向 $(-1, 0, 0)$ と推定運動方向とのなす角度の変化を示している．また，右下のグラフは，手首の力センサで得られた力の大きさの変化を示している．成功例では，引き出しを開く動作の実行中に，角度誤差と拘束力の大きさが減少していることがわかる．

失敗例では，推定方向が大きくずれた際にハンドルから手が抜けて把持が維持できなくなってしまい，その結果，拘束力が取得できなくなり失敗した．このような失敗を防ぐ方法の一つとして，対象操作物体の把持が維持できなくなってしまった際に復帰する機構を追加することが挙げられる．

図 6.15 に (A) ドア開けと (B) ハンドル回しの実行結果の例を示す．実機でもシミュレーションと同様にドア開けやハンドル回しが実行可能であることが分かった．図 6.16 にドア開け中の手の位置と力方向の遷移を示す．右上はドア開け中の人差し指位置，運動方向，力方向の遷移を示しており，左上は操作開始時の人差し指位置を原点とした実行時の座標系を示す．下段は，運動方向と初期動作方向 $(-1, 0, 0)$ とのなす角度を示している．右上に示すように，観測された力方向に基づいて動作方向が変化し，その結果，ドア開けに成功したことがわかる．また，下段の図から，徐々に初期運動方向と推定運動方向のなす角度が大きくなっていることが実際に分かる．



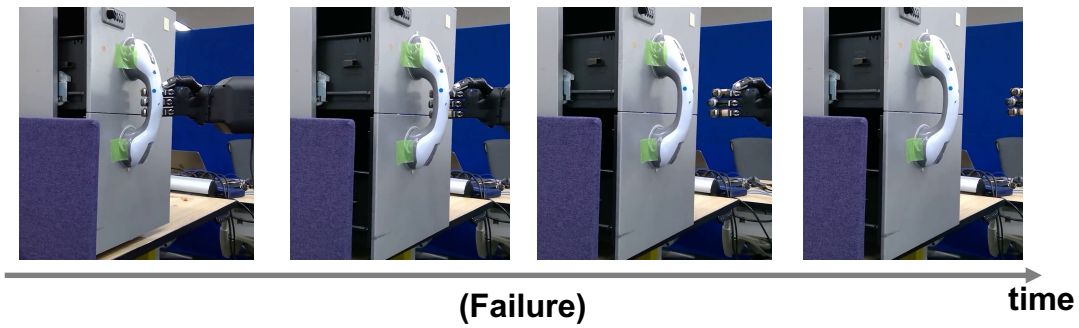


図 6.13　提案スキルを用いた引き出し開けの実行の成功例 (Success) と失敗例 (Failure).

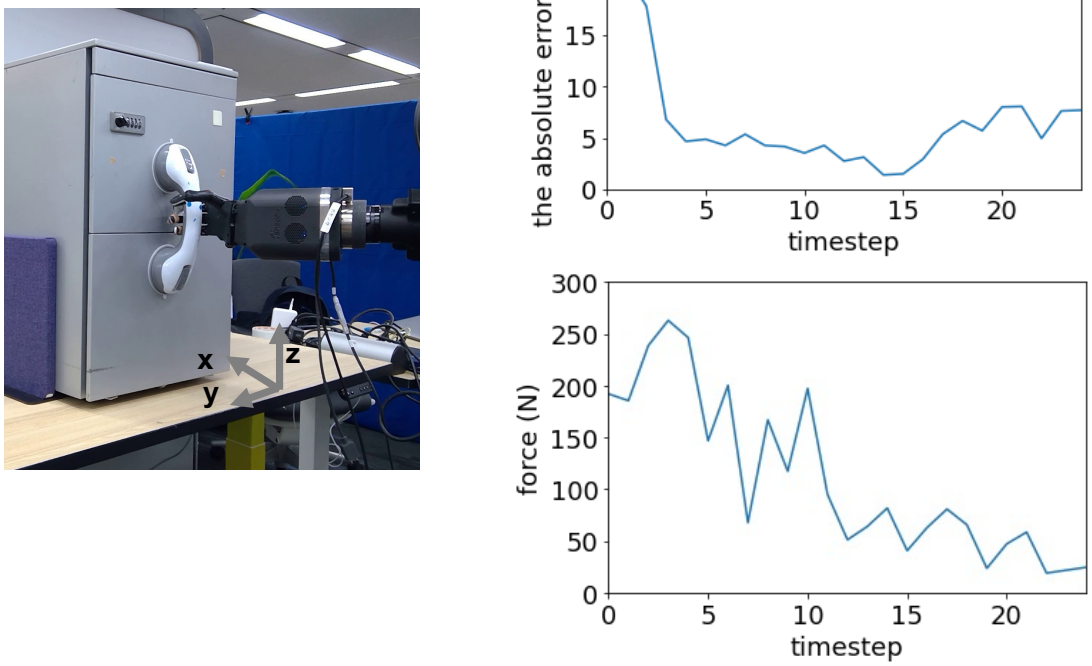


図 6.14　提案スキルを用いた引き出し開けの実行の際の推定方向と力の大きさの遷移．左上は座標系，右上は推定運動方向と許容方向 $(-1, 0, 0)$ のなす角度の変化，右下は力センサによる力の大きさの変化を示している．

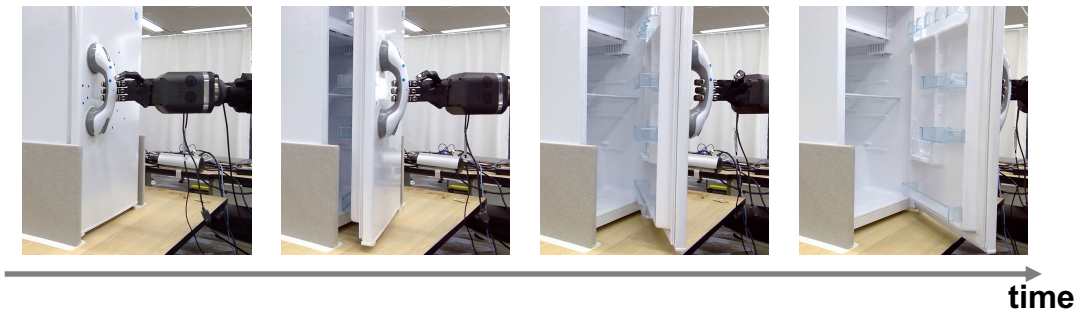


図 6.15　提案スキルの実機での実行の様子．(A) はドア開け，(B) はハンドル回しの結果である．

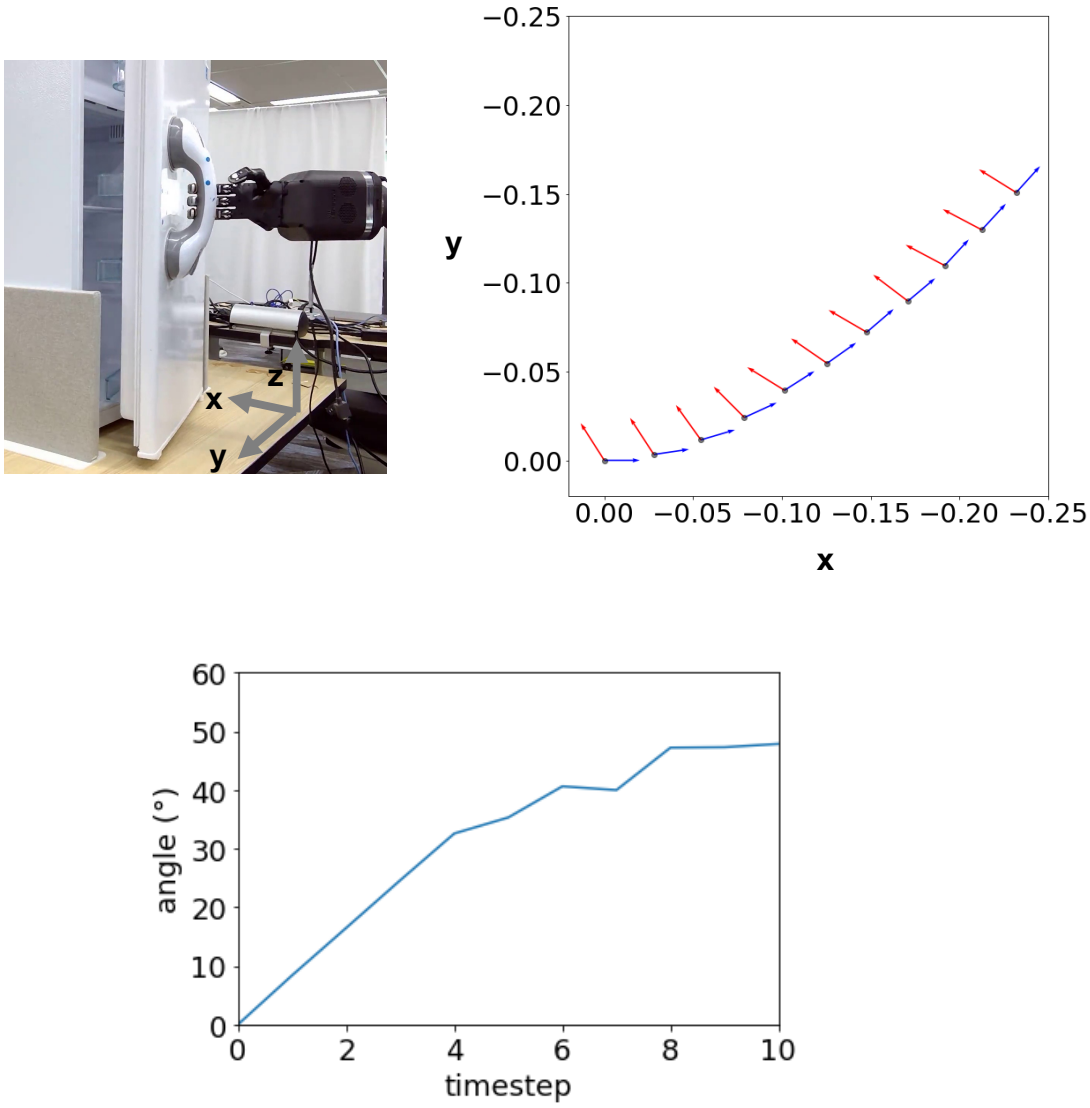


図 6.16　提案スキルを用いたドア開けの実行．左上は人差し指を原点とする座標系，右上は人差し指の位置（黒丸），運動方向（青矢印），力の方向（赤矢印）の遷移（メートル単位），下段は初期運動方向 $(-1, 0, 0)$ と運動方向のなす角度を示している．

6.7 議論

6.7.1 実験結果に対する考察

本論文では，物体に作用する拘束力の方向を用いて学習させて様々な未知の操作に対して汎化された constraint-aware policy を提案した．本実験では，PTG3 と PTG5 に含まれる操作に対する提案スキルの有効性を調べた．その結果、古典制御器では失敗した板を引く操作や棒を引く操作においても提案スキルを適用できた．環境と報酬は単純であるが，学習されたスキルは汎用的であることが分かった．さらに，ドア開けるとハンドル回しにも適用可能であることが分かった．最後に，提案スキルは追加の学習なしで実ロボット上で実行可能であった．以上の結果から，提案スキルは prismatic 関節や revolute 関節を用いた操作に汎用的であることが示唆され，これらの関節を持つより多くの操作に適用できる可能性があることが分かった．

パラメータ調整にかかる手間において，提案手法は古典制御器 [20, 24] と比較して優れている．古典制御器では制御パラメータを手動で調整する必要があるが，提案手法では学習によってパラメータを調整することができる．学習にかかる手間に関しては，強化学習や模倣学習 [23, 148, 149] によって学習する手法と比較して，本方法は 1 つの環境しか必要としない．これらの手法では，学習に必要な操作対象の環境（本研究では 5 つの環境）を全て用意するのに対し，提案手法では拘束と複合体のみの単純な 1 つの環境で学習が可能である．これは，操作プリミティブ内の拘束力の共通した特徴に基づく設計の利点である．

6.7.2 他の操作プリミティブへの適用可能性

家庭環境における多くの操作は，拘束に基づいて分類することができる [27]．この分類には prismatic 関節や revolute 関節を持つ操作プリミティブと，その他の拘束を持つ操作プリミティブが含まれる．家庭環境における様々な操作を実現するための一つ方法は，操作プリミティブごとにスキルを設計することである．家庭環境における様々な操作を実現するために，提案手法の概念を他の拘束に適用することができる可能性がある．本手法のように，同じ拘束を持つ様々な操作を 1 つの操作プリミティブとして考え，その中での共通の特徴を意識してスキル設計することが重要であると考えられる．共通の特徴を抽出することは今後の展望である．

本手法では力をフィードバックとして用いるスキルを設計した．この設計方法は，操作中に拘束の次元が変化しないようなものに関しては適用することができる．例えば，机を拭く操作等が含まれる操作プリミティブやお椀を拭く操作等が含まれる操作プリミティブに対しても同様のスキル設計が適用できる可能性がある．一方で，peg-in-hole のような操作では操作中に拘束の次元が変化してしまい，同様の設計は難しい．peg-in-hole 中に，ペグは空中で自由に動ける状態から穴に沿ってしか動けなくなるように拘束が変化する．このような場合には，力のみから許容方向を特定することが困難となる．これに対する解決策としては，視覚フィードバックを用いることが挙げられる．視覚フィードバックを用い

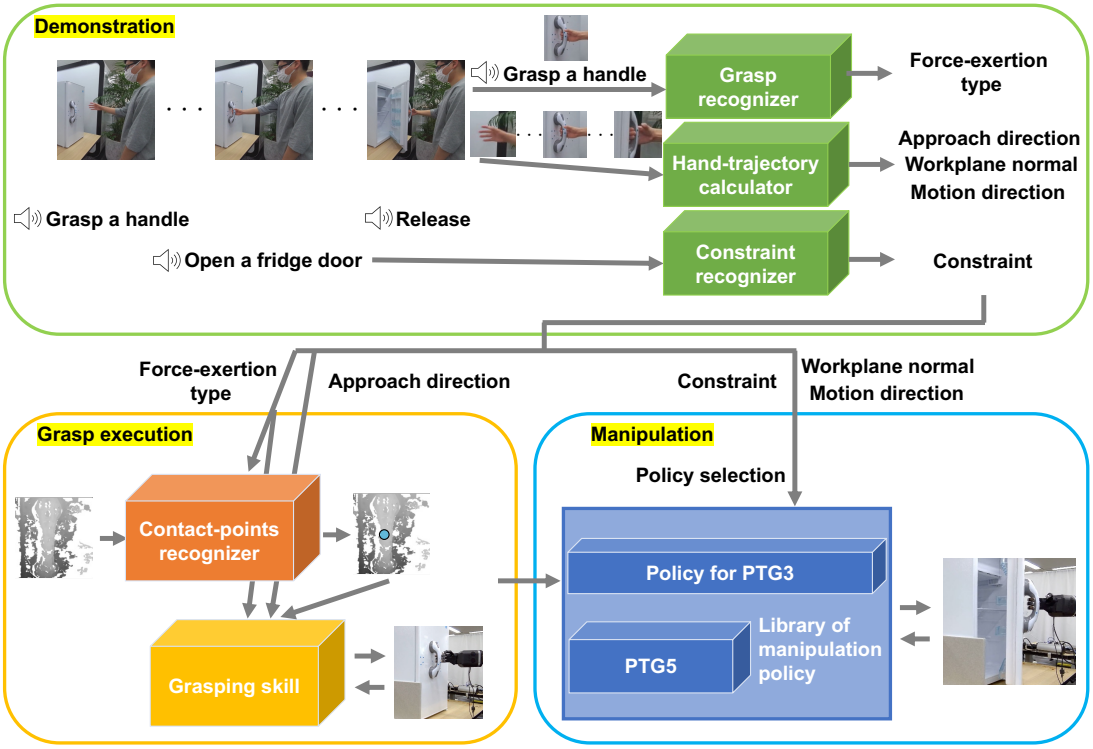


図 6.17　Constraint-aware policy の LfO への統合．図はドア開けの際の実行の流れを示している．

た汎用的なスキルの設計に関しては，今後の展望である．

6.7.3　LfO との組み合わせ

PTG5 では PTG3 とは異なり追加のスキルを必要とする．そのため，実行時には対象とする操作がどの操作プリミティブに含まれているのかに関する情報が必要となる．これは LfO の枠組みを用いることで，人間の言語指示から特定することが可能である [4, 85]．さらに，人間の実演動作から作業平面や移動方向の初期値，回転半径を推定することが可能であり，これは実行時に用いることができる (図 6.17)．

PTG5 では必要となる追加のスキルが対象物体を把持する際の force-exertion type によって異なることが分かった．Lazy-closure のような non-prehensile grasp で対象物体を把持した場合，手と物体の姿勢が大まかに連動していれば単一システム条件を満たすことができたが，passive-force closure のような prehensile grasp で対象物体を把持した場合には厳密に姿勢を連動させる必要がある．このような性質から必要となる追加のスキルが force-exertion type によって変化するのである．この追加のスキルの選択に関しても，LfO の枠組みを用いて人間の実演から force-exertion type を取得することで解決することができる．

6.7.4 異なるハードウェアへの再利用性

本論文では，単一システム条件を仮定し，ハードウェアの仕様を考慮することなくロボットハンドに適用可能な制約を考慮したポリシーを設計した．新しいハードウェアを使用する場合，ロボットプログラマは通常ソフトウェアを修正しなければならず，時間がかかることがある．この問題に対処するために，再利用性を可能にするいくつかのソフトウェアプログラムが既に開発されている．本研究の成果は，この分野へのもう一つの貢献である．提案スキルを使用することで異なるハードウェアへの再利用性を実現できる．再利用性を実証するために提案スキルの異なるロボットハンドへの再利用性を検証することは今後の展望である．

6.7.5 本手法の限界

単一システム条件の崩れ

提案スキルは単一システム条件下を想定して実装された．単一システム条件の違反例として，指先と操作対象との間の滑りがある．この滑りは，運動方向．回転軸，回転半径の推定誤差が大きいために発生する可能性がある．これらの問題は，提案スキルだけでは対処できない．解決策として，指先にかかる力を用いて指先を器用に動かすことで接触位置を維持するための追加のスキルを設計することが考えられる．これを実装するためには触覚センサが必要である．これは今後の展望である．

拘束力の正規化

関節の慣性力と摩擦は拘束力より弱く無視できると仮定した．本実験ではこの仮定が満たされ，この仮定を採用することができた．しかし，この仮定を満たさない場合もある．例えば、運動方向の推定誤差が 0° 付近の場合である．この場合，弱い慣性力や摩擦を正規化すると増幅されてしまう．この場合，システムが不安定になる．不安定性を回避するためには，あらかじめ定義した閾値より小さな力の大きさをゼロ値として計算する必要がある．閾値をどのように定義するかは今後の課題である．

in-hand manipulation による PTG5 の実行

本論文では，PTG5 の中でも回転中心が指先が作る凸包の中に含まれないような場合を対象とした．凸包の中に含まれる場合としては，つまみやコンロを回す操作が挙げられる．このような場合には，回転操作を無限小の並進とみなすことができないため，提案手法を適用することができない．一般にコンロを回す操作には指先の器用な動作である in-hand manipulation を使用することが多い．コンロを回す操作に関しては active-force closure を保ったまま物体を回転することになるので，前章で提案したプリミティブを応用することで操作できる可能性がある．この方法の設計に関しては今後の展望である．

PTG5 の中でも上述のように in-hand manipulation で行うか手の移動のみで操作を行うかで分類する必要があった．また，把持の種類によって必要となる追加のスキルが異な

るため，把持の種類でも分類する必要があった．このように汎用的なスキルを実現するためには，同一プリミティブ内でもさらに細かく分類が必要になる可能性がある．どの程度の細かさで分類する必要があるかについての議論も今後の展望である．

6.8 おわりに

本研究では，prismatic 関節や revolute 関節を持つ操作プリミティブ (PTG3, PTG5) に含まれる様々な操作に適用可能な constraint-aware policy を提案した．このスキルを学習するために，操作プリミティブが持つ共通の特徴に基づいて学習環境と報酬関数を設計した．実験の結果，シミュレーションにおいて運動方向の推定誤差が適用された場合であっても，PTG3 が含む操作 (引き出し開け，板引き，棒引き) において，単一のスキルで実行できることが示された．また，従来の古典制御器とは異なり，環境変化に対して頑健な実行が可能であった．また、PTG5 が含む操作 (ドア開け、ハンドル回し) においても提案スキルを実行することができた．さらに，引き出し開け，ドア開け，ハンドル回しの 3 つの操作について，実機で訓練なしに実行することに成功した．

提案スキルは単純な環境下で学習されたが，様々な操作に対して適用することができた．従来の強化学習手法では，対象となる操作ごとに環境と報酬を特別に設計していたのに対して，本手法は様々な操作に共通した拘束力に基づいて環境と報酬を設計した．これによって，PTG3 と PTG5 が含む操作に汎用的なスキルが設計できた．

様々な操作が可能なロボットシステムを実現するためには，各操作プリミティブに対して汎化されたスキルを設計することが重要である．家庭内操作は物理的制約によって有限個のプリミティブに分類することができる [27]．そのため，各プリミティブにスキルを用意できれば，全ての家庭内操作を実現できるようになる可能性がある．汎化にとって重要なことは，各プリミティブに共通する特徴に着目して環境と報酬を設計することである．本論文では，物理的制約を考慮する上で基本となる prismatic 関節と revolute 関節を持つ操作に対して constraint-aware policy の概念を検証した．これは，汎用的な家庭用ロボットの実現に向けた第一歩となる．

第 7 章

議論

本章では，各動作に対して本論文で提案した手法による実験結果や，それに対する考察を行う．特に，本論文の主題であった以下の二点を満たすスキルライブラリの設計に関して議論を行う．

1. 家庭環境において適したプリミティブの選択が可能
2. 非構造化環境において再利用可能なプリミティブスキル

さらに，本論文の限界やそれに対する今後の展望を述べる．

7.1 Grasp

本論文では，人間の把持分類やロボットの把持分類といったトップダウン知識に基づいて，作業の実行に重要である物体に対する力のかけ方によって分類された force-exertion type という新たな把持分類を提案した．force-exertion type と Learning-from-Observation(LfO) の枠組みを組み合わせることで，適切に把持を行わなければ上手くいかない作業を実現することができた．

多くのタスクプランナでは把持プリミティブが考慮されておらず安定把持を行うことのみに焦点が当てられているため，必ずしも目的作業にとって適切な把持が実現されるわけではない．そのため，場合によっては作業の実行に失敗してしまう．本論文では，物体への力のかけ方で把持プリミティブを設計し，タスクプランナの一種である LfO と組み合わせることで作業を実現した．このことから，要素 1 を満たしたスキル設計ができたと言える．

物体の大きさ・形状やアプローチ方向を randomization することで，これらが異なる場合であっても頑健に把持ができることが実験結果から示された．この結果は，物体の大きさ・形状やアプローチ方向が変化するたびにスキルを設計する必要がないということを示唆している．これらが変化すると把持に必要な動作も変化するはずだが，把持動作に対して再利用可能なスキルが学習できたということになる．このことから，要素 2 を満たしたスキル設計ができたと言える．

本論文の限界として，持ち替え動作が必要な把持プリミティブのスキル設計や，パーツを考慮した把持の実現が挙げられる．本論文では passive-form closure や no-closure に対

するスキルは設計しなかった．これらの把持プリミティブを実現するには，一度別の把持プリミティブで物体を掴んだ後に，目的とする把持に変化させる必要がある．例えば，passive-form closure であれば，物体を active-force closure で把持した後に指先で物体を巧みに操ることで実現できる．No-closure も同様の動作によって実現できる．これには In-hand manipulation のスキルが必要となり，第 5 章で設計したスキルを組み合わせることで実現できる可能性がある．これは今後の展望である．

本論文では，物体が superquadrics で近似できると仮定して実験を行ったが，実際には物体は複数のパーツから構成されることが多い．そのような場合，物体の機能を使用するには，適切なパーツを把持する必要がある．作業の達成のためには，パーツを考慮して把持することは無視できない要素である．これに関しては，画像上の物体のパーツを特定する part segmentation を組み合わせることで実現できる可能性がある．これも今後の展望である．

7.2 In-hand manipulation

本論文では，手の中での物体操作 (In-hand manipulation) だけではなく操作後に目的の作業に適した把持を実現するような操作である In-hand manipulation のスキルライブラリを設計した．In-hand manipulation は作業の実現に不可欠である．従来の in-hand manipulation に対する強化学習手法では，物体姿勢の変化のみが学習され，把持の実現に関して学習することが困難であった．本手法は，in-hand manipulation を接触状態の変化というトップダウン知識に基づいて複数のスキルに分割して学習し，それらを組み合わせることで in-hand manipulation を実現した．

実験の結果，本手法は active-force closure から active-force closure への把持の遷移を実現できることが示された．これは，接触状態の変化に基づいてスキルを分割し組み合わせたことで，目的とする接触状態の遷移の実現が保証されたからである．把持の遷移は接触状態の遷移とみなせるため，学習したスキルを組み合わせることで目的とする接触状態まで遷移することができた．このことから要素 1 を満たしたスキル設計ができたと考えられる．

物体の大きさ・形状の変化に頑健なスキル学習されていることも確認された．この結果から，物体の大きさ・形状の変化に対して再利用可能であるスキルが学習されたということが示唆される．関節角度の指令値と実測値との差から物体の大きさ・形状を認識することで，それに応じた動作を学習できたのだと考えられる．このことから要素 2 を満たしたスキル設計ができたと考えられる．

本論文の限界として，active-force closure から passive-form closure や no-closure へ遷移するためのスキル設計や，更なる接触状態の追加，視覚情報を組み合わせたスキルの設計が挙げられる．本論文では，active-force closure から active-force closure への遷移のみを対象にスキルを設計した．active-force closure から passive-form closure や no-closure への遷移も日常生活で多く見られるため，このスキルの設計もより広範囲の作業を実行するためには必要不可欠である．この遷移に関しては，本論文で設計した接触状態の遷移を

行うスキル以外に，接触状態を維持したまま物体の位置姿勢の変化を行うスキルを追加すれば実現できる可能性がある．この検証は今後の展望である．

本論文では，物体の向かい合う二面に指先がある接触状態のみに焦点を当ててスキルを設計したが，実際にはそれ以外の面に指先がある場合も存在する．例えば，箱の角を指先で押して箱を回転させる際に出現する．このような接触状態への遷移や，その接触状態の時の物体姿勢の変化を行うスキルを追加することでより多くの操作を行うことができる可能性がある．

本論文では常に物体の中心付近に指先があることを仮定してスキル設計を行った．物体の位置を変化させる場合には，この仮定は満たされない．そのような場合には，物体のどこに指先があるかを把握しておくことが操作にとって重要となる．これは物体と指先が写った画像や点群をスキルの入力に加えることで解決できる可能性がある．このスキルの学習が上手くできるかどうかの検証は今後の展望である．

7.3 Compliant manipulation

本論文では，拘束力に基づいて実行中の手の軌道を調整することで物理的拘束を持つ物体を操作する compliant manipulation 用のスキルを設計した．家庭環境には予測できない量の compliant manipulation が必要な操作が存在するため，要素 2 を満たすスキルを設計することが重要である．Compliant manipulation が必要な操作は物理的拘束により有限個のプリミティブに分類できる．そこで，本論文では各プリミティブに対して，そのプリミティブが含む操作に汎用的なスキルである constraint-aware policy を設計することを提案した．

実験の結果，prismatic 関節や revolute 関節を持つ物体に対する操作プリミティブである PTG3，PTG5 に対して汎用的なスキルが学習できていることが分かった．これはプリミティブ内に含まれる操作間に共通の特徴に基づいてスキルを設計したことによるものである．この結果から，要素 2 を満たすスキル設計ができたと考えられる．

本論文では，PTG3 と PTG5 に対してのみスキル設計を行ったが，同様に共通の特徴に基づいてスキル設計を行うことで他のプリミティブに対しても汎用的なスキルが実現できる可能性がある．Compliant manipulation が必要な操作は有限個のプリミティブに分類できるため，必要なスキルも有限個に抑えることができる．そのため，有限個のスキルで無限個の操作に対応できる可能性がある．様々な操作に汎用的なスキルを学習させる方法として，大量の操作データを模倣学習する手法 [25, 26, 42, 76, 148–151] も存在するが，分布外の操作に汎化することが未だに難しい [31]．そのため，現状では家庭環境に存在する予測できない量の操作に汎化することは困難である．一方で，本論文で提案した手法であれば，有限個のスキルを用意すればそれらに対処できる可能性がある．これはトップダウン知識を用いてスキル設計を行う利点である．

本論文の限界として，指先動作も含めたスキル設計，PTG5 用スキルが持つ仮定の緩和，PTG3，5 以外のプリミティブへのスキル設計が挙げられる．提案スキルは単一システム条件下を想定して実装された．単一システム条件の違反例として，指先と操作対象との間の

滑りがある．この滑りは，運動方向．回転軸，回転半径の推定誤差が大きいために発生する可能性がある．これらの問題は，手の動きだけを行う提案スキルだけでは対処できない．解決策として，指先にかかる力を用いて指先を器用に動かすことで接触位置を維持するための追加のスキルを設計することが考えられる．これを実装するためには触覚センサが必要である．これは今後の展望である．指先動作まで含めたスキルの学習に，本論文で学習したスキルが事前学習モデルとして機能する可能性もある．この検証も今後の展望である．

PTG5 用スキルは回転軸の方向が既知であるという仮定のもとで実行されている．この仮定はかなり強い仮定であり，実際には小さな誤差が発生してしまうことの方が多い．誤差が発生する場合，作業平面からの離脱による力が発生する．この離脱力に対処する動作まで含まれたスキル設計が必要になる．これは PTG3 用の環境ではなく PTG5 用の環境を用意して学習を行えば学習できる可能性がある．

本論文では，PTG3 と PTG5 に対してのみスキル設計を行ったが，実世界にはそれ以外のプリミティブも存在するため，より広範囲の操作も実現するためにはそれらに対するスキル設計も必要になる．本手法では力をフィードバックとして用いるスキルを設計した．この設計方法は，操作中に拘束の次元が変化しないようなものに関しては適用することができる．例えば，机を拭く操作等が含まれる操作プリミティブやお椀を拭く操作等が含まれる操作プリミティブに対しても同様のスキル設計が適用できる可能性がある．一方で，peg-in-hole のような操作では操作中に拘束の次元が変化してしまい，同様の設計は難しい．peg-in-hole 中に，ペグは空中で自由に動ける状態から穴に沿ってしか動けなくなるように拘束が変化する．このような場合には，力のみから許容方向を特定することが困難となる．これに対する解決策としては，視覚情報を用いることが挙げられる．視覚情報から拘束の変化を検知することで動作を修正すれば所望の操作ができる可能性がある．この検証は今後の展望である．

7.4 トップダウン知識による動作分類とスキル設計

本論文全体として，トップダウン知識に基づいて動作分類を行い，分類された各プリミティブに対してスキルを設計した．そして，このスキルをモジュールとして組み合わせることで目的の作業を再現した．トップダウン知識に基づいて動作分類を行うことの利点としては，プリミティブ内で共通した特徴に基づいてスキルが設計できるため，汎用的なスキルが学習できる可能性がある点である．これによって，学習したスキルは再利用可能性を持つ．学習したスキルが再利用可能である場合，そのスキルは一つのモジュールとしてシステムに組み込むことができる．再利用可能なモジュールが複数ある場合，それらの組み合わせで記述できる様々な作業を実行できるようになる．そのため，動作ごとに学習するスキルを学習する必要がなく，さらに，未知の作業でも再利用可能性によって実行ができる可能性があるという利点がある．

このような利点は，収集した動作データからスキルを学習するようなボトムアップ型の手法には無いものである．ボトムアップ型手法では，収集したデータ分布内であれば比較的高精度で困難な作業が実現できるものの，収集したデータ分布外の動作に対応できない

可能性がある．また，高精度で作業を実現したい場合には，その作業の動作データを収集し直す必要がある．そのため，未知の作業が多く存在する家庭環境でこのような手法を適用することは困難である．一方で，本手法では，作業ごとに学習し直す必要がなく，未知の作業にも適用できる可能性がある．今後，スキルを増やしていくことで，さらに多様な作業ができるようになると考えられる．

ただ，本論文で学習を行った compliant manipulation のスキルのように，再利用可能性のみに着目して設計してしまうと，指先動作のような再利用できない部分の学習が行えず成功率が下がってしまう可能性がある．この場合にはボトムアップ型手法と組み合わせていくことが重要であると考えらえる．その際に，in-hand manipulation の学習時のように適切に状態行動空間を分割して学習することで，学習分布が狭くなり汎化しやすくなる可能性がある．そのため，今後の方針として，トップダウン知識によって分類されたプリミティブ内に汎化するようなスキル学習が可能なボトムアップ型手法を検討するということが一つの方向として挙げられる．

7.5 今後の展望

7.5.1 学習時間の効率性の調査

作業を動作プリミティブに分割して学習を行うことで，複数の作業を行うのに必要なスキルの学習時間を短縮できる可能性がある．これは，プリミティブが異なる作業間において再利用可能性があるからである．図 7.1 にスキルの組み合わせの例を示す．例えば，物体を上から掴んでから他の場所に置く動作や板を掴んで引く動作を考えてみると，始めの把持の仕方は共通であるので，この把持動作を再利用することができる．同様に，板を掴んで引く動作，棒を掴んで引く動作，引き出しを引く動作では，把持の仕方は異なるものの，物体操作の仕方に関しては共通であり，この部分も再利用可能である．そのため，作業間で少なくとも把持や物体操作の部分に関しては汎用的に使えるため，作業に応じて何度も把持や物体操作を学習をさせ直すということは必要がなくなるはずである．目的作業が大量にある場合に，作業全体を学習するよりも，個別にプリミティブを学習して組み合わせる方が必要なスキルが少なくて済むはずである．さらに，学習時間も少なくて済むはずである．この効率性の検証は今後の展望である．

7.5.2 更なるスキルの設計

本論文で設計したスキルは一部のスキルのみであり，全ての操作に対して適用できるスキルライブラリを作成するには更なるスキルの設計が必要である．Grasp のスキルである passive-form closure や no-closure は in-hand manipulation のスキルが必要である．これに対しては本論文で学習したスキルが適用できる可能性がある．In-hand manipulation に関しては，更なるスキルとして接触状態の変化をさせずに物体の位置姿勢の変化を行うスキルが挙げられる．これに関しては，既存の in-hand manipulation の手法の報酬設計を基に学習できる可能性がある．Compliant manipulation に関しては，本論文では力フィー

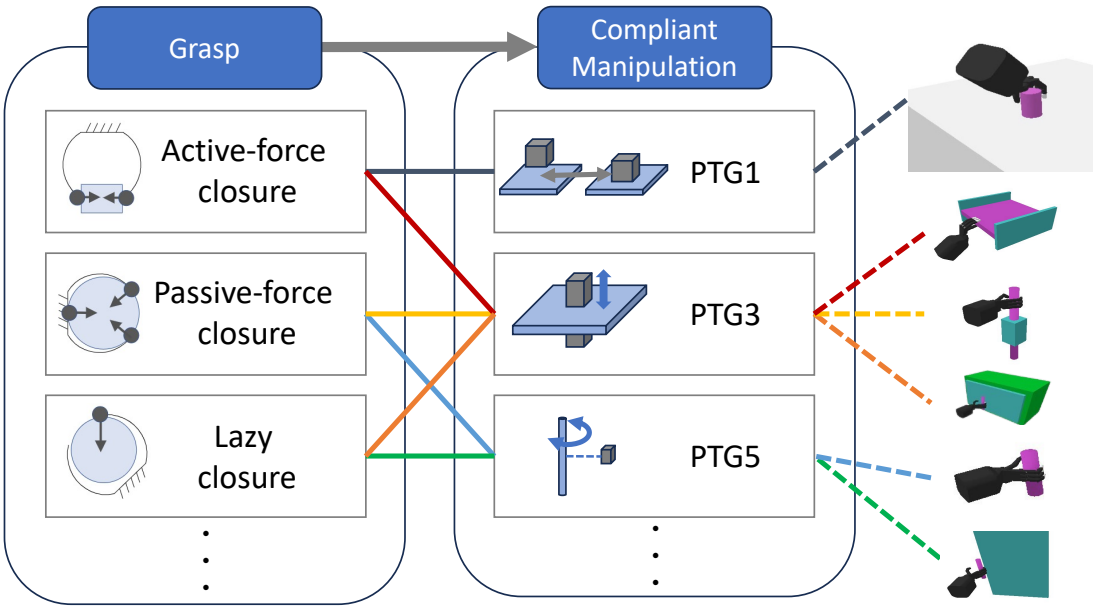


図 7.1　スキルの再利用可能性を活かしたスキルの組み合わせによる作業の実行例.

ドバックのみを入力とするスキル設計を行ったが，今後は視覚フィードバックも加えたマルチモーダルな学習が必要になると考えられる．この学習方法に関しては今後検証する必要がある．

7.5.3　スキルを組み合わせた実行

本論文で設計したスキルは LfO のようなタスクプランナと接続することで，逐次的に実行することが可能である．このような実行の際に，前のスキルの結果得られる状態が実行しようとしているスキルが想定する入力分布外になってしまうような分布シフトが起きてしまった場合，現在のスキルが失敗してしまう可能性がある (図 7.2)．例えば，active-force closure で把持した位置が in-hand manipulation のスキルの想定した位置とは離れてしまっている場合に，上手く実行できないということが起こる．これに対処する方法として，より入力分布を広げて学習するということが挙げられる．また，実行時に上手くいかない場合に再学習を行うということも一つの解決策である．本論文が提案したシステムのように動作を分割しておくことで，上手くいかないスキルのみを取り出して再学習ができる．動作を分割していない場合には動作全体を再学習する必要があるため，このようなことはできない．これはトップダウン知識を基に動作を分割することの利点である．

7.5.4　ボトムアップ型手法との組み合わせ

上述したように，トップダウン知識に基づいて再利用可能性のみに着目してスキルを設計してしまうと，再利用できない部分の学習が行えず作業の成功率が下がってしまう可能性がある．より成功率を上げる方法として，再利用できない部分も含めてボトムアップ型

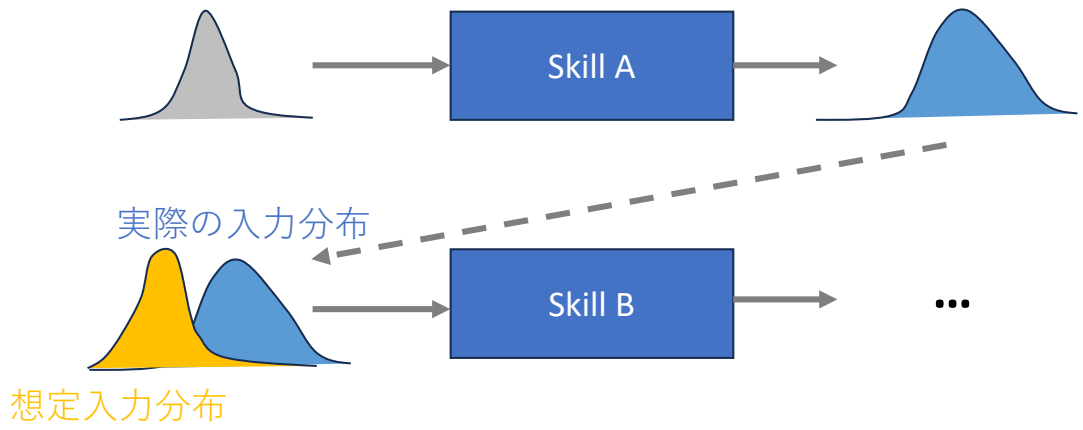


図 7.2　分布シフトによる失敗．青色で示した分布が Skill A から入力される分布で，黄色で示した分布が Skill B が想定した入力分布である．

手法を適用することが挙げられる．

ボトムアップ型手法の適用時には従来の研究のように全ての動作を学習しようとするのではなく，トップダウン知識に基づいて適切に状態行動空間を分割して学習することで，学習分布が狭くなり汎化しやすくなる可能性がある．そこで，今後の方針として，トップダウン知識によって分類されたプリミティブ内に汎化するようなスキル学習が可能なボトムアップ型手法を検討するということが一つの方向として挙げられる．具体的には，あるプリミティブ内に含まれる複数の動作のデータを収集し，模倣学習することが挙げられる．データ間に共通部分があるため，全く関係ない動作が混ざったデータを学習するよりも，汎化性能が上がる可能性がある．さらに，非共通部分の分布は全く関係ない動作が混ざったデータよりも狭いため，汎化できる可能性は比較的高いと考えられる．

7.5.5　動作の本質的な部分の抽出

本論文では，動作の本質的な部分を抽出することでプリミティブを分類した．この抽出は動作の観察によって行われている．そのため，観察から動作の本質が得ることが難しいような場合には本手法のようにプリミティブ分類を行うことが困難である．例えば，柔軟物体のような接触状態が容易に定義できないような場合が挙げられる．このような場合への対処方法としては二つ挙げられる．一つ目は，接触状態ではなく物体の状態そのものに着目する方法である．例えば，ひも結びをひもの状態によって表現する [81] というように，作業中の物体自体の状態遷移をプリミティブとして定義する方法が考えられる．二つ目は，大量の実演データから重要な部分を抽出する方法である．人間が模倣を行う際のミラーニューロンシステムにおいて，単なる行動の模倣ではなく行動価値を推定している可能性があることが報告されている [179]．これは行動の本質的な部分を抽出していることだと捉えることができる．このような抽出を計算機で行う方法として逆強化学習が挙げられる．逆強化学習を用いてある動作における重要な部分を抽出するという方法が，接触状態

が容易に定義できないような場合のプリミティブ設計に役立つ可能性がある．

7.5.6　家庭用ロボットのデザイン

今後，本論文で提案したスキルを実行できるロボットを家庭用ロボットとして導入する場合に，どのようなデザインのものが適しているのかどうかを考える必要がある．家庭内作業は無数にあるため，産業用ロボットのように作業ごとにロボットを導入するのは非現実的であり，一台で多様な作業を行えることが好ましい．そのためには，多様な物体を扱うための多指ハンド，多様なハンド姿勢を達成するための多関節アーム，多様な作業場へ移動するための足をロボットのデザインとして採用するべきだと考えられる．以上の議論はロボットの機能的な観点での必要条件であるが，家庭へ導入する際には人間に受け入れられるかどうかも重要である．実は，人型ロボットの方がそうでないロボットよりも好まれるということが報告されている [180]．また，人間にとって予測可能な動きをする方が心理的負荷が低いということも報告されている [181]．関節の数が多いロボットの場合，逆運動学の解が多く存在するため人間の予測した動きから外れやすくなると考えられる．おそらく，人間と同程度の関節数でないと予測が難しいと考えられる．以上から，これらの報告を考慮すると，ロボットのデザイン面では人体形状を模倣したロボットを開発するべきであると考えられる．さらに，今後は人間が予測可能な動きを再現するような関節角度を求める逆運動学の手法を検討することが重要になってくると考えられる．

7.5.7　求められる家庭内作業の網羅

家庭用ロボットにどんな作業を代替して欲しいかを調べた研究 [30] において，人間よりもロボットの方に代行して欲しく，かつ肉体的な作業で理論上可能なものを分析した．その結果，Locomotion や物体認識は既存手法 [167, 182] によってできるという前提で現状のプリミティブスキルで 60% 程度が可能で，過半数の作業は理論的には実行可能になったと考えられる．一方で，不可能な作業は，柔軟物体操作，拘束が途中で変化する操作，双腕操作が関わる作業であった．今後は，柔軟物体操作や双腕操作の設計や，拘束が途中で変化する操作のために視覚フィードバックを含めたスキル設計を行なっていくことが実行可能な作業を増やしていくために重要である．

7.6　本論文の限界

以上を踏まえて本論文の限界を述べる．

7.6.1　スキルライブラリの LfO システムへの統合

本論文では Grasp, In-hand manipulation, Compliant manipulation の三つの分野に対して，LfO に繋ぐことができるスキルライブラリの構築を行なった．しかしながら，これら三つを統合した実行に関しては行うことができておらず，システム全体の評価は行っていない．これは今後の展望である．

7.6.2 学習を行わなかったプリミティブスキルの学習設計

プリミティブの考察を行なったものの，プリミティブの学習を行なったのは基礎的で頻出のプリミティブのみである．そのため，全てのプリミティブに対してスキルが学習できたわけではない．学習を行わなかったプリミティブスキルの学習を行い，スキルライブラリを完成させることが次の段階として挙げられる．その際に，本論文で提案したようなプリミティブに固有の共通特徴に基づいた学習設計思想が活用できる可能性がある．この検証は今後の展望である．

7.6.3 異なるハードウェアでのプリミティブスキル学習

本論文で提案した Grasp，In-hand manipulation のプリミティブにはハードウェアに前提がある．Grasp では少なくとも対向する二つの指があることが前提になっている．In-hand manipulation では少なくとも 4 本の指があることが前提となっている．以上のようなハードウェアの制約さえ満たせば，提案したプリミティブが変化することはない．一方で，プリミティブスキルは再学習が必要である．これは強化学習時の環境がハードウェア依存の環境になっているからである．状態や行動，報酬関数といった学習設計自体はハードウェアに非依存であるように設計したため，本論文で提案した学習設計が再利用できる可能性がある．このような異なるハードウェアでのプリミティブ学習は今後の展望である．

第 8 章

結論

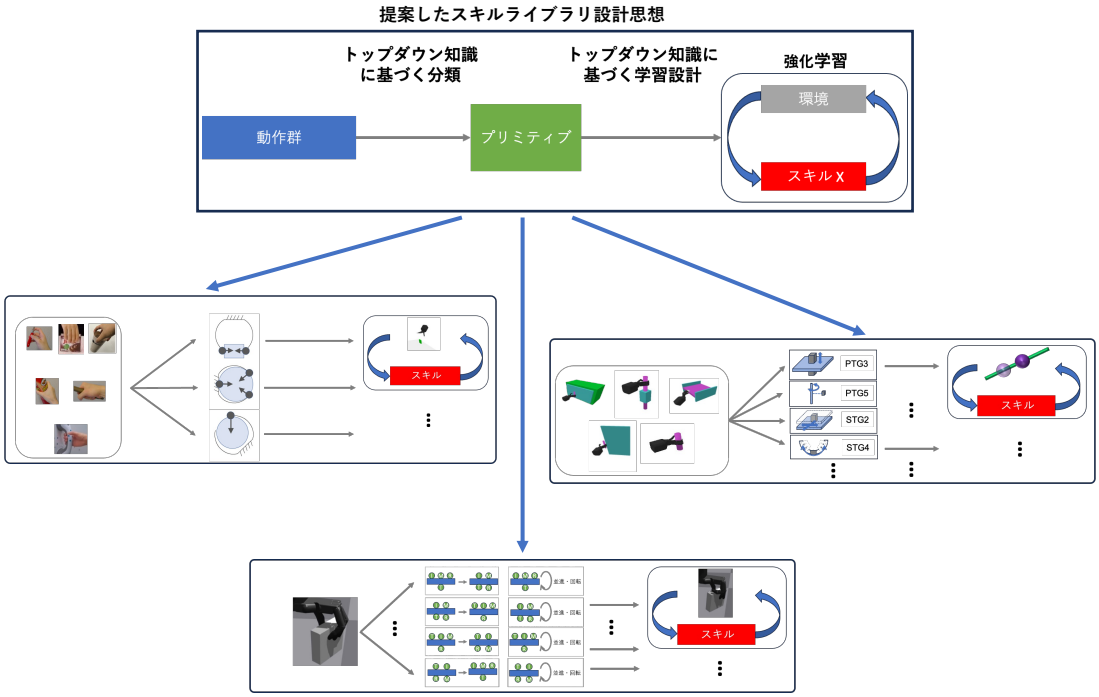


図 8.1　本論文で提案したスキル設計思想とその応用例.

本論文では，手腕を巧みに用いた動作を含む家庭内作業が可能な家庭用ロボットに向けたスキルライブラリの設計を目的として，手腕を用いた動作の中でも，特に作業の遂行において必要不可欠な Grasp，In-hand manipulation，Compliant manipulation に焦点を当ててスキルを設計した．家庭内作業を行うためには，以下の二点を満たすスキルライブラリを設計する必要があるが，これまでの研究ではこれらの要素を満たす設計ができていなかった．

1. 家庭環境において適したプリミティブの選択が可能
2. 非構造化環境において再利用可能なプリミティブスキル

本論文では，これまでの動作分類の研究で得られたトップダウン知識に基づいて，スキル

を設計することで上述した要素満たしたスキルライブラリを設計した (図 8.1).

第 4 章では，Grasp に関するスキルライブラリを設計した．物体への力のかけ方というトップダウン知識に基づいて新たに force-exertion type という把持分類を提案した．提案した force-exertion type と人間の実演から必要なパラメータを抽出できる Learning-from-Observation と提案したスキルライブラリを組み合わせることで，(1) コップを掴んでカゴに入れる，(2) 冷蔵庫のハンドルを掴んでドアを開ける，という，後続スキルの成功のために適切に把持を選択する必要がある 2 つの作業が実機で成功できることを確認した．また，学習したスキルは物体の大きさ・形状やアプローチ方向の変化に頑健であることを示した．これによりスキルが様々な動作に対して再利用可能性があることが分かった．

第 5 章では，In-hand manipulation に関するスキルライブラリを提案した．接触状態の遷移を起こす三種類の動作というトップダウン知識を用いて，操作全体をより細かな単位に分割した APriCoT(Action Primitives based on Contact-state Transition) を導入し，これらの単位ごとに学習を行うことを提案した．この単位を用いることで接触状態が適切に遷移されることが保証される．その結果，目的の把持を実現することができる．実験の結果，本手法を用いて物体を回転させて active-force closure から active-force closure に遷移させる操作ができることを確認した．また，学習したスキルは物体の大きさ・形状の変化に頑健であることを示した．これによりスキルが様々な動作に対して再利用可能性があることが確認できた．

第 6 章では，Compliant manipulation に関するスキルライブラリを設計した．操作プリミティブが含む操作に汎用的なスキルである constraint-aware policy を提案した．Constraint-aware policy は操作プリミティブが持つ物理的拘束というトップダウン知識に基づいて設計された環境と報酬を用いて学習される．実験の結果，prismatic 関節や revolute 関節によって拘束された物体の操作のまとまりである PTG3 や PTG5 が含む操作に汎用的であることが確認された．これによりスキルの様々な動作に対する再利用可能性が示唆された．

以上のように，本論文ではトップダウン知識に基づいて上述した二つの要素を満たすスキルライブラリを設計した．その中でも家庭内で頻出でかつ基礎的な Grasp，In-hand manipulation，Compliant manipulation を対象として，スキルライブラリを設計した．そして，シミュレーションや実世界において設計したスキルの評価を行い，非構造化環境でのスキルの再利用可能性を確認した．

トップダウン知識に基づいてスキルライブラリを設計することによって，プリミティブ内で共通した特徴に基づいてスキルが設計できるため，プリミティブ内の動作に汎用的なスキルが学習できる可能性がある点である．家庭内作業に含まれる動作は有限個のプリミティブに分類できる．そのため，トップダウン知識に基づいて各プリミティブに対してスキルの学習を行うことで，有限個のスキルで無数にある動作を実現できる可能性がある．また，上述した再利用可能性によって，スキルを一つのモジュールとしてシステムに組み込むことが可能になる．再利用可能なモジュールが複数ある場合，それらの組み合わせで記述できる様々な作業を実行できるようになる．そのため，対象とする作業が変わったとしてもスキルを再学習することなく，未知の作業でも実行ができる可能性があるという利

点がある．本論文は，そのような汎用的なスキルライブラリの実現への第一歩となる．

また，本論文では LfO 向けとしてスキルライブラリを設計したが，設計したライブラリは LfO 以外にも他のタスクプランナに組み込むことができる可能性がある．例えば，階層強化学習や階層模倣学習，大規模言語モデルといったタスクプランナにスキルライブラリを組み込むことで，人間の実演から得ていた様々なスキルパラメータをタスクプランナで推論でき，作業の指示のみで多様な家庭内作業が可能になる．このように，本論文で提案した家庭内作業向けスキルライブラリは，LfO 分野を含めた広範囲なタスクプランナに組み込み可能であり，家庭内作業のスキル自動獲得といった分野全体に対して貢献できるものである．

謝辞

本論文は筆者が東京科学大学情報理工学院情報工学系情報工学コース博士課程在籍中の研究成果をまとめたものである．

本研究を遂行するにあたり，本研究の機会を与えて頂き，さらに熱心に御指導を頂きました小池英樹先生に深く感謝致します．ロボットを用いた研究を行いたい筆者に実際にロボットに触れられる機会を与えて頂いたことで非常に有意義な経験ができ，さらに本論文の執筆に繋がりました．

素晴らしい研究環境を与えて頂き，さらに業務や研究等で御多忙にも関わらず毎週のようにミーティングで丁寧に御指導を頂きました Microsoft Applied Robotics Research の池内克史先生に深く感謝致します．池内先生との議論では，機械に全てを任せないで人間と協調するべきという思想や要素還元主義的な設計を全面に押し出す姿を多く見ました．この姿から，問題を解く際に自分なりの思想を持つことが研究を進めていく上で重要であるのだということを学ばせて頂きました．筆者が博士課程在籍中に行った研究はこれらの思想に大きく触発されています．研究者としての目指すべき姿を示していただいたことにも加えて感謝致します．

御多忙にも関わらず研究の進め方や実装，論文の書き方で何度も貴重な御指導や御助言を頂きました Microsoft Applied Robotics Research の高松淳先生，笹渕一宏さん，和家尚希さん，兼平篤志さんに深く感謝致します．ハードウェアやロボットプログラミングの実装に関する知識がほぼ無かった筆者に対して皆様が丁寧に教えてくださったことで本研究が実現できました．また，ロボットの修理のためにロボットを分解しなければならなかった時に，御多忙な中でも一緒に分解して頂いたことに感謝致します．ロボットの分解を通して多くのことを学ばせて頂きました．さらに，論文の推敲の際に，筆者の書いた未熟な文章を何度も読んで繰り返し丁寧にコメントを残してくださったことに深く感謝致します．論理の流れを意識して書くことの難しさやその修正の仕方を学べたことは，文章を書くことが苦手な筆者にとって非常に実りある経験になりました．

共に研究生活を過ごしてきた小池研究室の皆様にも感謝致します．

最後に長期間に渡って学生生活を支援してくださいました家族に心より感謝致します．

参考文献

[1] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087*, 2017.

[2] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*.

[3] Katsushi Ikeuchi and Takashi Suehiro. Toward an assembly plan from observation. i. task recognition with polyhedral objects. *IEEE transactions on robotics and automation*, Vol. 10, No. 3, pp. 368–385, 1994.

[4] Naoki Wake, Riku Arakawa, Iori Yanokura, Takuya Kiyokawa, Kazuhiro Sasabuchi, Jun Takamatsu, and Katsushi Ikeuchi. A learning-from-observation framework: One-shot robot teaching for grasp-manipulation-release household operations. In *2021 IEEE/SICE International Symposium on System Integration (SII)*, pp. 461–466. IEEE, 2021.

[5] Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, Vol. 34, No. 4-5, pp. 705–724, 2015.

[6] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning*, pp. 651–673. PMLR, 2018.

[7] Mark R Cutkosky, et al. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Transactions on robotics and automation*, Vol. 5, No. 3, pp. 269–279, 1989.

[8] Thomas Feix, Javier Romero, Heinz-Bodo Schmiedmayer, Aaron M Dollar, and Danica Kragic. The grasp taxonomy of human grasp types. *IEEE Transactions on human-machine systems*, Vol. 46, No. 1, pp. 66–77, 2015.

[9] Margarita Vergara, J.L. Sancho-Bru, V. Gracia-Ibáñez, and A. Pérez-González. An introductory study of common grasps used by adults during performance of activities of daily living. *Journal of Hand Therapy*, Vol. 27, No. 3, pp. 225–234,

2014.

[10] Hui Li, Yinlong Zhang, Yanan Li, and Hongsheng He. Learning task-oriented dexterous grasping from human knowledge. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6192–6198. IEEE, 2021.

[11] Kuan Fang, Yuke Zhu, Animesh Garg, Andrey Kurenkov, Viraj Mehta, Li Fei-Fei, and Silvio Savarese. Learning task-oriented grasping for tool manipulation from simulated self-supervision. *The International Journal of Robotics Research*, Vol. 39, No. 2-3, pp. 202–216, 2020.

[12] Priyanka Mandikal and Kristen Grauman. Learning dexterous grasping with object-centric visual affordances. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6169–6176. IEEE, 2021.

[13] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *IJRR*, Vol. 39, No. 1, pp. 3–20, 2020.

[14] Max Yang, Chenghua Lu, Alex Church, Yijiong Lin, Chris Ford, Haoran Li, Efi Psomopoulou, David AW Barton, and Nathan F Lepora. Anyrotate: Gravity-invariant in-hand object rotation with sim-to-real touch. *arXiv preprint arXiv:2405.07391*, 2024.

[15] Rana Soltani Zarrin, Rianna Jitosho, and Katsu Yamane. Hybrid learning-and model-based planning and control of in-hand manipulation. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8720–8726. IEEE, 2023.

[16] Haozhi Qi, Ashish Kumar, Roberto Calandra, Yi Ma, and Jitendra Malik. In-hand object rotation via rapid motor adaptation. In *Conference on Robot Learning*, pp. 1722–1732. PMLR, 2023.

[17] 工藤俊亮, 佐藤啓宏, 池内克史ほか. タングルトポロジーを用いたロボットハンドによる人間の持ち替え動作の模倣. 日本ロボット学会誌, Vol. 33, No. 7, pp. 514–523, 2015.

[18] Ellen Klingbeil, Ashutosh Saxena, and Andrew Y. Ng. Learning to open new doors. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2751–2757, 2010.

[19] Xiaolong Li, He Wang, Li Yi, Leonidas J. Guibas, A. Lynn Abbott, and Shuran Song. Category-level articulated object pose estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3703–3712, 2020.

[20] Andreas J Schmid, Nicolas Gorges, Dirk Goger, and Heinz Worn. Opening a door with a humanoid robot using multi-sensory tactile feedback. In *2008 IEEE International Conference on Robotics and Automation*, pp. 285–291. IEEE, 2008.

[21] Advait Jain and Charles C Kemp. Pulling open doors and drawers: Coordinating an omni-directional base and a compliant arm with equilibrium point control. In *2010 IEEE International Conference on Robotics and Automation*, pp. 1807–1814. IEEE, 2010.

[22] Ali Yahya, Adrian Li, Mrinal Kalakrishnan, Yevgen Chebotar, and Sergey Levine. Collective robot reinforcement learning with distributed asynchronous guided policy search. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 79–86. IEEE, 2017.

[23] Yusuke Urakami, Alec Hodgkinson, Casey Carlin, Randall Leu, Luca Rigazio, and Pieter Abbeel. Doorgym: A scalable door opening environment and baseline agent. *arXiv preprint arXiv:1908.01887*, 2019.

[24] Yiannis Karayiannidis, Christian Smith, Francisco Eli Vina Barrientos, Petter Ögren, and Danica Kragic. An adaptive control approach for opening doors and drawers under uncertainties. *IEEE Transactions on Robotics*, Vol. 32, No. 1, pp. 161–175, 2016.

[25] Abby O' Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024.

[26] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Charles Xu, Jianlan Luo, Tobias Kreiman, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.

[27] Katsushi Ikeuchi, Naoki Wake, Kazuhiro Sasabuchi, and Jun Takamatsu. Semantic constraints to represent common sense required in household actions for multimodal learning-from-observation robot. *The International Journal of Robotics Research*, Vol. 43, No. 2, pp. 134–170, 2024.

[28] Sing Bing Kang and Katsushi Ikeuchi. Grasp recognition using the contact web. In *IROS*, pp. 194–201, 1992.

[29] Yoshio Matsumoto, Yoshifumi Nishida, Yoichi Motomura, and Yayoi Okawa. A concept of needs-oriented design and evaluation of assistive robots based on icf. In *2011 IEEE International Conference on Rehabilitation Robotics*, pp. 1–6. IEEE, 2011.

[30] Cory-Ann Smarr, Tracy L Mitzner, Jenay M Beer, Akanksha Prakash, Tiffany L Chen, Charles C Kemp, and Wendy A Rogers. Domestic robots for older adults: attitudes, preferences, and potential. *International journal of social robotics*,

Vol. 6, pp. 229–247, 2014.

[31] Zhijie Wang, Zhehua Zhou, Jiayang Song, Yuheng Huang, Zhan Shu, and Lei Ma. Towards testing and evaluating vision-language-action models for robotic manipulation: An empirical study. *arXiv preprint arXiv:2409.12894*, 2024.

[32] Constructions Aeronautiques, Adele Howe, Craig Knoblock, ISI Drew McDermott, Ashwin Ram, Manuela Veloso, Daniel Weld, David Wilkins Sri, Anthony Barrett, Dave Christianson, et al. Pddl— the planning domain definition language. *Technical Report, Tech. Rep.*, 1998.

[33] B Bonet. Planning as heuristic search. *Artificial Intelligence*, 2001.

[34] Malte Helmert. The fast downward planning system. *Journal of Artificial Intelligence Research*, Vol. 26, pp. 191–246, 2006.

[35] Aditya Gudimella, Ross Story, Matineh Shaker, Ruofan Kong, Matthew Brown, Victor Shnayder, and Marcos Campos. Deep reinforcement learning for dexterous manipulation with concept networks. *arXiv preprint arXiv:1709.06977*, 2017.

[36] Murtaza Dalal, Deepak Pathak, and Russ R Salakhutdinov. Accelerating robotic reinforcement learning via parameterized action primitives. *Advances in Neural Information Processing Systems*, Vol. 34, pp. 21847–21859, 2021.

[37] Soroush Nasiriany, Huihan Liu, and Yuke Zhu. Augmenting reinforcement learning with behavior primitives for diverse manipulation tasks. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 7477–7484. IEEE, 2022.

[38] Jianlan Luo, Charles Xu, Xinyang Geng, Gilbert Feng, Kuan Fang, Liam Tan, Stefan Schaal, and Sergey Levine. Multi-stage cable routing through hierarchical imitation learning. *IEEE Transactions on Robotics*, 2024.

[39] Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, et al. Do as i can, not as i say: Grounding language in robotic affordances. In *Conference on robot learning*, pp. 287–318. PMLR, 2023.

[40] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. Progprompt: program generation for situated robot task planning using large language models. *Autonomous Robots*, Vol. 47, No. 8, pp. 999–1012, 2023.

[41] Keisuke Shirai, Cristian C Beltran-Hernandez, Masashi Hamaya, Atsushi Hashimoto, Shohei Tanaka, Kento Kawaharazuka, Kazutoshi Tanaka, Yoshitaka Ushiku, and Shinsuke Mori. Vision-language interpreter for robot task planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2051–2058. IEEE, 2024.

[42] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pp. 1094–1100.

PMLR, 2020.

[43] Abhishek Gupta, Vikash Kumar, Corey Lynch, Sergey Levine, and Karol Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. In *Conference on Robot Learning*, pp. 1025–1037. PMLR, 2020.

[44] Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Kevin Lin, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. In *arXiv preprint arXiv:2009.12293*, 2020.

[45] Sing Bing Kang and Katsushi Ikeuchi. Toward automatic robot instruction from perception-mapping human grasps to manipulator grasps. *IEEE transactions on robotics and automation*, Vol. 13, No. 1, pp. 81–95, 1997.

[46] Phongtharin Vinayavekhin, Shunsuke Kudoh, and Katsushi Ikeuchi. Towards an automatic robot regrasping movement based on human demonstration using tangle topology. In *2011 IEEE International Conference on Robotics and Automation*, pp. 3332–3339. IEEE, 2011.

[47] Phongtharin Vinayavekhin, Shunsuke Kudoh, Jun Takamatsu, Yoshihiro Sato, and Katsushi Ikeuchi. Representation and mapping of dexterous manipulation through task primitives. In *2013 IEEE International Conference on Robotics and Automation*, pp. 3722–3729. IEEE, 2013.

[48] Jun Takamatsu, Koichi Ogawara, Hiroshi Kimura, and Katsushi Ikeuchi. Recognizing assembly tasks through human demonstration. *The International Journal of Robotics Research*, Vol. 26, No. 7, pp. 641–659, 2007.

[49] Nathan Elangovan, Ricardo V Godoy, Felipe Sanches, Ke Wang, Tom White, Patrick Jarvis, and Minas Liarokapis. On human grasping and manipulation in kitchens: Automated annotation, insights, and metrics for effective data collection. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11329–11335. IEEE, 2023.

[50] Matthew T Mason. Compliance and force control for computer controlled manipulators. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 11, No. 6, pp. 418–432, 1981.

[51] T Yoshikawa. Passive and active closures by constraining mechanisms. In *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 1477–1484. IEEE, 1996.

[52] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, Vol. 17, No. 1, pp. 1334–1373, 2016.

[53] Kanishka Rao, Chris Harris, Alex Irpan, Sergey Levine, Julian Ibarz, and Mohi Khansari. Rl-cyclegan: Reinforcement learning aware simulation-to-real. In *Pro-*

ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11157–11166, 2020.

[54] Yevgen Chebotar, Karol Hausman, Zhe Su, Gaurav S Sukhatme, and Stefan Schaal. Self-supervised regrasping using spatio-temporal tactile features and reinforcement learning. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1960–1966. IEEE, 2016.

[55] Bohan Wu, Iretiayo Akinola, Jacob Varley, and Peter K. Allen. Mat: Multi-fingered adaptive tactile grasping via deep reinforcement learning. In Leslie Pack Kaelbling, Danica Kragic, and Komei Sugiura, editors, *Proceedings of the Conference on Robot Learning*, Vol. 100 of *Proceedings of Machine Learning Research*, pp. 142–161. PMLR, 30 Oct–01 Nov 2020.

[56] Hamza Merzić, Miroslav Bogdanović, Daniel Kappler, Ludovic Righetti, and Jeannette Bohg. Leveraging contact forces for learning to grasp. In *2019 international conference on robotics and automation (ICRA)*, pp. 3615–3621. IEEE, 2019.

[57] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning Complex Dexterous Manipulation with Deep Reinforcement Learning and Demonstrations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2018.

[58] Edwin Valarezo Añazco, Patricio Rivera Lopez, Nahyeon Park, Jiheon Oh, Gahyeon Ryu, Mugahed A Al-antari, and Tae-Seong Kim. Natural object manipulation using anthropomorphic robotic hand through deep reinforcement learning and deep grasping probability network. *Applied Intelligence*, Vol. 51, No. 2, pp. 1041–1055, 2021.

[59] Haozhi Qi, Brent Yi, Sudharshan Suresh, Mike Lambeta, Yi Ma, Roberto Calandra, and Jitendra Malik. General in-hand object rotation with vision and touch. In *Conference on Robot Learning*, pp. 2549–2564. PMLR, 2023.

[60] Johannes Pitz, Lennart Röstel, Leon Sievers, and Berthold Bäuml. Dextrous tactile in-hand manipulation using a modular reinforcement learning architecture. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1852–1858. IEEE, 2023.

[61] Gagan Khandate, Maximilian Haas-Heger, and Matei Ciocarlie. On the feasibility of learning finger-gaiting in-hand manipulation with intrinsic sensing. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2752–2758. IEEE, 2022.

[62] Henry J Charlesworth and Giovanni Montana. Solving challenging dexterous manipulation tasks with trajectory optimisation and reinforcement learning. In *International Conference on Machine Learning*, pp. 1496–1506. PMLR, 2021.

[63] Jun Wang, Ying Yuan, Haichuan Che, Haozhi Qi, Yi Ma, Jitendra Malik,

and Xiaolong Wang. Lessons from learning to spin" pens". *arXiv preprint arXiv:2407.18902*, 2024.

[64] Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 3389–3396. IEEE, 2017.

[65] Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. *Advances in neural information processing systems*, Vol. 31, , 2018.

[66] Yufeng Sun, Lin Zhang, and Ou Ma. Force-vision sensor fusion improves learning-based approach for self-closing door pulling. *IEEE Access*, Vol. 9, pp. 137188–137197, 2021.

[67] Nelson Vithayathil Varghese and Qusay H Mahmoud. A survey of multi-task deep reinforcement learning. *Electronics*, Vol. 9, No. 9, p. 1363, 2020.

[68] Yan Duan, Marcin Andrychowicz, Bradly Stadie, OpenAI Jonathan Ho, Jonas Schneider, Ilya Sutskever, Pieter Abbeel, and Wojciech Zaremba. One-shot imitation learning. *Advances in neural information processing systems*, Vol. 30, , 2017.

[69] Tianhao Zhang, Zoe McCarthy, Owen Jow, Dennis Lee, Xi Chen, Ken Goldberg, and Pieter Abbeel. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 5628–5635. IEEE, 2018.

[70] Edward Johns. Coarse-to-fine imitation learning: Robot manipulation from a single demonstration. In *2021 IEEE international conference on robotics and automation (ICRA)*, pp. 4613–4619. IEEE, 2021.

[71] Nam Jun Cho, Sang Hyoung Lee, Jong Bok Kim, and Il Hong Suh. Learning, improving, and generalizing motor skills for the peg-in-hole tasks based on imitation learning and self-learning. *Applied Sciences*, Vol. 10, No. 8, p. 2719, 2020.

[72] Jonatan S Dyrstad, Elling Ruud Øye, Annette Stahl, and John Reidar Mathiassen. Teaching a robot to grasp real fish by imitation learning from a human supervisor in virtual reality. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7185–7192. IEEE, 2018.

[73] Shaunak A Mehta and Rana Soltani Zarrin. On the feasibility of a mixed-method approach for solving long horizon task-oriented dexterous manipulation. In *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pp. 949–956. IEEE, 2024.

[74] Thanpimon Buamanee, Masato Kobayashi, Yuki Uranishi, and Haruo Takemura. Bi-act: Bilateral control-based imitation learning via action chunking with transformer. In *2024 IEEE International Conference on Advanced Intelligent Mecha-*

tronics (AIM), pp. 410–415, 2024.

[75] Andrew Choong-Won Lee, Ian Chuang, Ling-Yuan Chen, and Iman Soltani. Interact: Inter-dependency aware action chunking with hierarchical attention transformers for bimanual manipulation. In *8th Annual Conference on Robot Learning*.

[76] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

[77] Ayano Hiranaka, Minjune Hwang, Sharon Lee, Chen Wang, Li Fei-Fei, Jiajun Wu, and Ruohan Zhang. Primitive skill-based robot learning from human evaluative feedback. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7817–7824. IEEE, 2023.

[78] Utkarsh Aashu Mishra, Shangjie Xue, Yongxin Chen, and Danfei Xu. Generative skill chaining: Long-horizon skill planning with diffusion models. In *Conference on Robot Learning*, pp. 2905–2925. PMLR, 2023.

[79] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, Vol. 24, No. 240, pp. 1–113, 2023.

[80] Yoshihiro Sato, Keni Bernardin, Hiroshi Kimura, and Katsushi Ikeuchi. Task analysis based on observing hands and objects by vision. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2, pp. 1208–1213. IEEE, 2002.

[81] J. Takamatsu, T. Morita, K. Ogawara, H. Kimura, and K. Ikeuchi. Representation for knot-tying tasks. *IEEE Transactions on Robotics*, Vol. 22, No. 1, pp. 65–78, 2006.

[82] Shin'ichiro Nakaoka, Atsushi Nakazawa, Fumio Kanehiro, Kenji Kaneko, Mitsuharu Morisawa, Hirohisa Hirukawa, and Katsushi Ikeuchi. Learning from observation paradigm: Leg task models for enabling a biped humanoid robot to imitate human dances. *The International Journal of Robotics Research*, Vol. 26, No. 8, pp. 829–844, 2007.

[83] Naoki Wake, Iori Yanokura, Kazuhiro Sasabuchi, and Katsushi Ikeuchi. Verbal focus-of-attention system for learning-from-observation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10377–10384. IEEE, 2021.

[84] Iori Yanaokura, Naoki Wake, Kazuhiro Sasabuchi, Riku Arakawa, Kei Okada, Jun Takamatsu, Masayuki Inaba, and Katsushi Ikeuchi. A multimodal learning-from-observation towards all-at-once robot teaching using task cohesion. In *2022 IEEE/SICE International Symposium on System Integration (SII)*, pp. 367–374.

IEEE, 2022.

[85] Naoki Wake, Atsushi Kanehira, Kazuhiro Sasabuchi, Jun Takamatsu, and Katsushi Ikeuchi. Interactive task encoding system for learning-from-observation. In *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1061–1066. IEEE, 2023.

[86] Naoki Wake, Daichi Saito, Kazuhiro Sasabuchi, Hideki Koike, and Katsushi Ikeuchi. Text-driven object affordance for guiding grasp-type recognition in multimodal robot teaching. *Machine Vision and Applications*, Vol. 34, No. 4, p. 58, 2023.

[87] Naoki Wake, Atsushi Kanehira, Kazuhiro Sasabuchi, Jun Takamatsu, and Katsushi Ikeuchi. Gpt-4v (ision) for robotics: Multimodal task planning from human demonstration. *IEEE Robotics and Automation Letters*, 2024.

[88] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, Vol. 1, pp. 348–353. IEEE, 2000.

[89] Randy C Brost. Automatic grasp planning in the presence of uncertainty. *The International Journal of Robotics Research*, Vol. 7, No. 1, pp. 3–17, 1988.

[90] Andrew T Miller, Steffen Knoop, Henrik I Christensen, and Peter K Allen. Automatic grasp planning using shape primitives. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, Vol. 2, pp. 1824–1829. IEEE, 2003.

[91] Corey Goldfeder, Peter K Allen, Claire Lackner, and Raphael Pelossof. Grasp planning via decomposition trees. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 4679–4684. IEEE, 2007.

[92] Kai Huebner, Steffen Ruthotto, and Danica Kragic. Minimum volume bounding box decomposition for shape approximation in robot grasping. In *2008 IEEE International Conference on Robotics and Automation*, pp. 1628–1633. IEEE, 2008.

[93] Mehmet Remzi Dogar, Kaijen Hsiao, Matei T Ciocarlie, and Siddhartha S Srinivasa. Physics-based grasp planning through clutter. In *Robotics: Science and systems*, Vol. 8, pp. 57–64, 2012.

[94] Andrew T Miller and Peter K Allen. Graspit! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, Vol. 11, No. 4, pp. 110–122, 2004.

[95] Carlo Ferrari, John F Canny, et al. Planning optimal grasps. In *ICRA*, Vol. 3, p. 6, 1992.

[96] Joseph Redmon and Anelia Angelova. Real-time grasp detection using convolutional neural networks. In *2015 IEEE international conference on robotics and*

automation (ICRA), pp. 1316–1322. IEEE, 2015.

[97] Zhichao Wang, Zhiqi Li, Bin Wang, and Hong Liu. Robot grasp detection using multimodal deep convolutional neural networks. *Advances in Mechanical Engineering*, Vol. 8, No. 9, p. 1687814016668077, 2016.

[98] Di Guo, Fuchun Sun, Tao Kong, and Huaping Liu. Deep vision networks for real-time robotic grasp detection. *International Journal of Advanced Robotic Systems*, Vol. 14, No. 1, p. 1729881416682706, 2016.

[99] Sulabh Kumra and Christopher Kanan. Robotic grasp detection using deep convolutional neural networks. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 769–776. IEEE, 2017.

[100] Roberto Calandra, Andrew Owens, Manu Upadhyaya, Wenzhen Yuan, Justin Lin, Edward H Adelson, and Sergey Levine. The feeling of success: Does touch sensing help predict grasp outcomes? In *Conference on Robot Learning*, pp. 314–323. PMLR, 2017.

[101] Francois R Hogan, Maria Bauza, Oleguer Canal, Elliott Donlon, and Alberto Rodriguez. Tactile regrasp: Grasp adjustments via simulated tactile transformations. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2963–2970. IEEE, 2018.

[102] Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International journal of robotics research*, Vol. 37, No. 4-5, pp. 421–436, 2018.

[103] Ian M Bullock, Raymond R Ma, and Aaron M Dollar. A hand-centric classification of human and robot dexterous manipulation. *IEEE transactions on Haptics*, Vol. 6, No. 2, pp. 129–144, 2012.

[104] Michael Anthony Arbib. Coordinated control programs for movements of the hand. *Experimantal Brain Research*, Vol. 10, pp. 111–129, 1985.

[105] Hui Li, Dang Tran, Xinyu Zhang, and Hongsheng He. Knowledge augmentation and task planning in large language models for dexterous grasping. In *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, pp. 1–8. IEEE, 2023.

[106] Zengyi Qin, Kuan Fang, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Keto: Learning keypoint representations for tool manipulation, 2019.

[107] Mia Kokic, Danica Kragic, and Jeannette Bohg. Learning task-oriented grasping from human activity datasets. *IEEE Robotics and Automation Letters*, Vol. 5, No. 2, pp. 3352–3359, 2020.

[108] Samarth Brahmbhatt, Cusuh Ham, Charles C Kemp, and James Hays. Contactdb: Analyzing and predicting grasp contact via thermal imaging. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*,

pp. 8709–8719, 2019.

[109] Samarth Brahmbhatt, Ankur Handa, James Hays, and Dieter Fox. Contactgrasp: Functional multi-finger grasp synthesis from contact. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2386–2393. IEEE, 2019.

[110] Samarth Brahmbhatt, Chengcheng Tang, Christopher D Twigg, Charles C Kemp, and James Hays. Contactpose: A dataset of grasps with object contact and hand pose. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pp. 361–378. Springer, 2020.

[111] Priyanka Mandikal and Kristen Grauman. Dexvip: Learning dexterous grasping with human hand pose priors from video. In *Conference on Robot Learning*, pp. 651–661. PMLR, 2022.

[112] Allison M Okamura, Niels Smaby, and Mark R Cutkosky. An overview of dexterous manipulation. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, Vol. 1, pp. 255–262. IEEE, 2000.

[113] Li Han and Jeffrey C Trinkle. Dextrous manipulation by rolling and finger gaiting. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*, Vol. 1, pp. 730–735. IEEE, 1998.

[114] Daniela Rus. In-hand dexterous manipulation of piecewise-smooth 3-d objects. *The International Journal of Robotics Research*, Vol. 18, No. 4, pp. 355–381, 1999.

[115] Igor Mordatch, Zoran Popović, and Emanuel Todorov. Contact-invariant optimization for hand manipulation. In *Proceedings of the ACM SIGGRAPH/Eurographics symposium on computer animation*, pp. 137–144, 2012.

[116] Yunfei Bai and C Karen Liu. Dexterous manipulation using both palm and fingers. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1560–1565. IEEE, 2014.

[117] Tatsuya Ishihara, Akio Namiki, Masatoshi Ishikawa, and Makoto Shimojo. Dynamic pen spinning using a high-speed multifingered hand with high-speed tactile sensor. In *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pp. 258–263. IEEE, 2006.

[118] Shoma Nakatani and Yuji Yamakawa. Dynamic manipulation like normal-type pen spinning by a high-speed robot hand and a high-speed vision system. In *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 636–642. IEEE, 2023.

[119] Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael

Ribas, et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.

[120] Ankur Handa, Arthur Allshire, Viktor Makoviychuk, Aleksei Petrenko, Ritvik Singh, Jingzhou Liu, Denys Makoviichuk, Karl Van Wyk, Alexander Zhurkevich, Balakumar Sundaralingam, et al. Dextreme: Transfer of agile in-hand manipulation from simulation to reality. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5977–5984. IEEE, 2023.

[121] Zhao-Heng Yin, Binghao Huang, Yuzhe Qin, Qifeng Chen, and Xiaolong Wang. Rotating without seeing: Towards in-hand dexterity through touch. *arXiv preprint arXiv:2303.10880*, 2023.

[122] Tao Chen, Jie Xu, and Pulkit Agrawal. A system for general in-hand object re-orientation. In *Conference on Robot Learning*, pp. 297–307. PMLR, 2022.

[123] Sridhar Pandian Arunachalam, Sneha Silwal, Ben Evans, and Lerrel Pinto. Dexterous imitation made easy: A learning-based framework for efficient dexterous manipulation. In *2023 ieee international conference on robotics and automation (icra)*, pp. 5954–5961. IEEE, 2023.

[124] Andrew Melnik, Luca Lach, Matthias Plappert, Timo Korthals, Robert Haschke, and Helge Ritter. Using tactile sensing to improve the sample efficiency and performance of deep deterministic policy gradients for simulated in-hand manipulation tasks. *Frontiers in Robotics and AI*, Vol. 8, p. 538773, 2021.

[125] Vikash Kumar, Abhishek Gupta, Emanuel Todorov, and Sergey Levine. Learning dexterous manipulation policies from experience and imitation. *arXiv preprint arXiv:1611.05095*, 2016.

[126] Vikash Kumar, Emanuel Todorov, and Sergey Levine. Optimal control with learned local models: Application to dexterous manipulation. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 378–383. IEEE, 2016.

[127] Yecheng Jason Ma, William Liang, Guanzhi Wang, De-An Huang, Osbert Bastani, Dinesh Jayaraman, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Eureka: Human-level reward design via coding large language models. *arXiv preprint arXiv:2310.12931*, 2023.

[128] Tingguang Li, Krishnan Srinivasan, Max Qing-Hu Meng, Wenzhen Yuan, and Jeannette Bohg. Learning hierarchical control for robust in-hand manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8855–8862. IEEE, 2020.

[129] George Konidaris and Andrew Barto. Skill discovery in continuous reinforcement learning domains using skill chaining. *Advances in neural information processing systems*, Vol. 22, , 2009.

[130] Alexander Clegg, Wenhao Yu, Jie Tan, C. Karen Liu, and Greg Turk. Learning

to dress: synthesizing human dressing motion via deep reinforcement learning. *ACM Trans. Graph.*, Vol. 37, No. 6, dec 2018.

[131] Yuanpei Chen, Chen Wang, Li Fei-Fei, and Karen Liu. Sequential dexterity: Chaining dexterous policies for long-horizon manipulation. In *Conference on Robot Learning*, pp. 3809–3829. PMLR, 2023.

[132] Ethan K Gordon and Rana Soltani Zarrin. Online augmentation of learned grasp sequence policies for more adaptable and data-efficient in-hand manipulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5970–5976. IEEE, 2023.

[133] K. Nagatani and S.I. Yuta. An experiment on opening-door-behavior by an autonomous mobile robot with a manipulator. In *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, Vol. 2, pp. 45–50 vol.2, 1995.

[134] Ben Abbatematteo, Stefanie Tellex, and George Konidaris. Learning to generalize kinematic models to novel objects. In Leslie Pack Kaelbling, Danica Kragic, and Komei Sugiura, editors, *Proceedings of the Conference on Robot Learning*, Vol. 100 of *Proceedings of Machine Learning Research*, pp. 1289–1299. PMLR, 30 Oct–01 Nov 2020.

[135] Thomas Rühr, Jürgen Sturm, Dejan Pangercic, Michael Beetz, and Daniel Cremers. A generalized framework for opening doors and drawers in kitchen environments. In *2012 IEEE International Conference on Robotics and Automation*, pp. 3852–3858, 2012.

[136] Miguel Arduengo, Carme Torras, and Luis Sentis. Robust and adaptive door operation with a mobile robot. *Intelligent Service Robotics*, Vol. 14, No. 3, pp. 409–425, 2021.

[137] Liu Liu, Han Xue, Wenqiang Xu, Haoyuan Fu, and Cewu Lu. Toward real-world category-level articulation pose estimation. *IEEE Transactions on Image Processing*, Vol. 31, pp. 1072–1083, 2022.

[138] Ajinkya Jain, Rudolf Lioutikov, Caleb Chuck, and Scott Niekum. Screwnet: Category-independent articulation model estimation from depth images using screw theory. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13670–13677, 2021.

[139] Ben Eisner, Harry Zhang, and David Held. Flowbot3d: Learning 3d articulation flow to manipulate articulated objects. *arXiv preprint arXiv:2205.04382*, 2022.

[140] Fangyin Wei, Rohan Chabra, Lingni Ma, Christoph Lassner, Michael Zollhöfer, Szymon Rusinkiewicz, Chris Sweeney, Richard Newcombe, and Mira Slavcheva. Self-supervised neural articulated shape and appearance models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15816–15826, 2022.

[141] G. Niemeyer and J.-J.E. Slotine. A simple strategy for opening an unknown door. In *Proceedings of International Conference on Robotics and Automation*, Vol. 2, pp. 1448–1453 vol.2, 1997.

[142] Woojin Chung, Changju Rhee, Youngbo Shim, Hyungjin Lee, and Shinsuk Park. Door-opening control of a service robot using the multifingered robot hand. *IEEE Transactions on Industrial Electronics*, Vol. 56, No. 10, pp. 3975–3984, 2009.

[143] Dedi Ma, Hesheng Wang, and Weidong Chen. Unknown constrained mechanisms operation based on dynamic hybrid compliance control. In *2011 IEEE International Conference on Robotics and Biomimetics*, pp. 2366–2371. IEEE, 2011.

[144] Davide Pilastro, Roberto Oboe, and Tomoyuki Shimono. A nonlinear adaptive compliance controller for rehabilitation. *IEEJ Journal of Industry Applications*, Vol. 5, No. 2, pp. 123–131, 2016.

[145] Luca Corrá, Roberto Oboe, and Tomoyuki Shimono. Adaptive optimal control for rehabilitation systems. In *IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society*, pp. 5197–5202. IEEE, 2017.

[146] Shrey Pareek, Harris NIsar, et al. Ar3n: A reinforcement learning-based assist-as-needed controller for robotic rehabilitation. *arXiv preprint arXiv:2303.00085*, 2023.

[147] Yiannis Karayiannidis, Christian Smith, Petter Ögren, and Danica Kragic. Adaptive force/velocity control for opening unknown doors1. *IFAC Proceedings Volumes*, Vol. 45, No. 22, pp. 753–758, 2012.

[148] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[149] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022.

[150] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pp. 785–799. PMLR, 2023.

[151] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[152] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, Vol. 1, p. 2. Minneapolis, Minnesota, 2019.

[153] Kazuhiro Sasabuchi, Naoki Wake, and Katsushi Ikeuchi. Task-oriented motion mapping on robots of various configuration using body role division. *IEEE Robotics and Automation Letters*, Vol. 6, No. 2, pp. 413–420, 2020.

[154] Daichi Saito, Kazuhiro Sasabuchi, Naoki Wake, Jun Takamatsu, Hideki Koike, and Katsushi Ikeuchi. Task-grasping from a demonstrated human strategy. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pp. 880–887. IEEE, 2022.

[155] Daichi Saito, Naoki Wake, Kazuhiro Sasabuchi, Hideki Koike, and Katsushi Ikeuchi. Contact web status presentation for freehand grasping in mr-based robot-teaching. In *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 167–171, 2021.

[156] Thea Iberall. Grasp planning from human prehension. In *IJCAI*, Vol. 87, pp. 1153–1157. Citeseer, 1987.

[157] Alan H Barr. Superquadrics and angle-preserving transformations. *IEEE Computer graphics and Applications*, Vol. 1, No. 01, pp. 11–23, 1981.

[158] Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 international conference on advanced robotics (ICAR)*, pp. 510–517. IEEE, 2015.

[159] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 2938–2946, 2015.

[160] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. `http://pybullet.org`, 2016–2021.

[161] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, Vol. 24, No. 6, pp. 381–395, 1981.

[162] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

[163] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[164] Jiankun Li, Peisen Wang, Pengfei Xiong, Tao Cai, Ziwei Yan, Lei Yang, Jiangyu Liu, Haoqiang Fan, and Shuaicheng Liu. Practical stereo matching via cascaded recurrent network with adaptive correlation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16263–16272, 2022.

[165] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[166] Sebastian Starke, Norman Hendrich, and Jianwei Zhang. A memetic evolutionary algorithm for real-time articulated kinematic motion. In *2017 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2473–2479. IEEE, 2017.

[167] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

[168] James J Gibson. *The ecological approach to visual perception: classic edition.* Psychology press, 2014.

[169] Tim Oblak, Jaka Šircelj, Vitomir Štruc, Peter Peer, Franc Solina, and Aleš Jaklič. Learning to predict superquadric parameters from depth images with explicit and implicit supervision. *IEEE Access*, Vol. 9, pp. 1087–1102, 2020.

[170] Sing Bing Kang and Katsushi Ikeuchi. Toward automatic robot instruction from perception-recognizing a grasp from observation. *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 4, pp. 432–443, 1993.

[171] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *Conference on Robot Learning*, pp. 66–75. PMLR, 2020.

[172] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, Vol. 5, No. 47, p. eabc5986, 2020.

[173] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, Vol. 9, No. 11, 2008.

[174] Taein Kwon, Bugra Tekin, Jan Stühmer, Federica Bogo, and Marc Pollefeys. H2o: Two hands manipulating objects for first person interaction recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10138–10148, 2021.

[175] Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. Hierarchical reinforcement learning: A comprehensive survey. *ACM Computing Surveys (CSUR)*, Vol. 54, No. 5, pp. 1–35, 2021.

[176] Dinh-Cuong Hoang, Phan Xuan Tan, Anh-Nhat Nguyen, Duy-Quang Vu, Van-Duc Vu, Thu-Uyen Nguyen, Ngoc-Anh Hoang, Khanh-Toan Phan, Duc-Thanh Tran, Van-Thiep Nguyen, Quang-Tri Duong, Ngoc-Trung Ho, Cong-Trinh Tran, Van-Hiep Duong, and Phuc-Quan Ngo. Multi-modal hand-object pose estimation with adaptive fusion and interaction learning. *IEEE Access*, Vol. 12, pp. 54339–54351, 2024.

[177] Sina Masnadi, Joseph J LaViola Jr, Jana Pavlasek, Xiaofan Zhu, Karthik Desingh, and Odest Chadwicke Jenkins. Sketching affordances for human-in-the-loop robotic manipulation tasks. *2nd Robot Teammates Operating in Dynamic, Unstructured Environments (RT-DUNE)*, 2019.

[178] Daichi Saito, Masashi Shibata, Yunzhuo Wang, Takeo Igarashi, and Keita Higuchi. Ghoja: Human-in-the-loop joint pose optimization based on geometric constraint and human common-sense. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pp. 1–6, 2024.

[179] Sven Collette, Wolfgang M Pauli, Peter Bossaerts, and John O'Doherty. Neural computations underlying inverse reinforcement learning in the human brain. *Elife*, Vol. 6, p. e29718, 2017.

[180] Maartje MA de Graaf and Somaya Ben Allouch. Users' preferences of robots for domestic use. In *Proceedings of the 2014 ACM/IEEE international conference on Human-Robot interaction*, pp. 146–147, 2014.

[181] Robin Jeanne Kirschner, Henning Mayer, Lisa Burr, Nico Mansfeld, Saeed Abdolshah, and Sami Haddadin. Expectable motion unit: Avoiding hazards from human involuntary motions in human-robot interaction. *IEEE Robotics and Automation Letters*, Vol. 7, No. 2, pp. 2993–3000, 2022.

[182] Ilija Radosavovic, Tete Xiao, Bike Zhang, Trevor Darrell, Jitendra Malik, and Koushil Sreenath. Real-world humanoid locomotion with reinforcement learning. *Science Robotics*, Vol. 9, No. 89, p. eadi9579, 2024.