DOCTORAL DISSERTATION

Agile Reconfigurable Robotic Assembly System

Takuya Kiyokawa

March 17, 2021

Graduate School of Science and Technology
Nara Institute of Science and Technology

A DOCTORAL DISSERTATION
submitted to Graduate School of Science and Technology,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Takuya Kiyokawa

Thesis Committee:

Professor Tsukasa Ogasawara	(Supervisor)
Professor Hirokazu Kato	(Co-supervisor)
Associate Professor Jun Takamatsu	(Co-supervisor)
Assistant Professor Gustavo Alfonso Garcia Ricardez	(Co-supervisor)
Assistant Professor Sung-Gwi Cho	(Co-supervisor)

Agile Reconfigurable Robotic Assembly System*

Takuya Kiyokawa

Abstract

There is an ever-increasing demand for a high-mix, low-volume production system that can quickly and flexibly respond to variations in both the type and quantity of products required by markets worldwide. To achieve automatic assembly that performs the high-mix, low-volume production for such market volatility, a general-purpose assembly robot is required to operate a wide variety of mechanical parts of various shapes.

Such a new-type production paradigm prefers, rather than a line production which enables mass production of a relatively small number of different products, a cell production which enables entire assembly process of customized products in one cell. Required specifications of the assembly system are frequently changed, and thus the use of an agile reconfigurable robot is expected. In the Assembly Challenge of the World Robot Summit, a global competition on automated robotic assembly, assuming changes in parts of a target product (a belt drive unit) to be assembled, participating teams tackle how to deal with the product changes with a robotic system.

The purpose of this dissertation is to clarify the system configuration methods that enable rapid deployment in response to the introduction of new products. First, I define problems of this dissertation considering the findings and remaining issues in the wide range of the field related to robotic assembly. Specifically, to address the problems, I propose three methods: (1) an automatic training dataset generation method for a quickly trainable vision system, (2) an assembly sequence generation method using only an assembled CAD model, and (3) an assembly method using a general-purpose flexible jig inspired by a jamming gripper.

The methods proposed in this dissertation have novelty, originality, and usefulness that can be differentiated from previous studies in terms of the following three aspects. (1) In order to utilize a vision system using deep learning that are recently getting attention due to its significant performance, I focused on the needs of rapid dataset generation to handle frequent product changes in the

*DOCTORAL DISSERTATION, Graduate School of Science and Technology,
Nara Institute of Science and Technology, March 17, 2021.

manufacturing industry. I developed an automatic dataset generation of training image datasets, that enables agile reconfiguration of the vision system. (2) Benefiting from the latest advance in three-dimensional modeling technologies due to the recent increase in processing speed of computers, I achieved automatic assembly sequence generation based on the difficulty of constraint state transition, which is related to the difficulty of assembly tasks. Using the three-dimensional model data, the proposed method can automatically extract interference relation, insertion conditions and degree of constraints used for evaluating the assembly sequence for a product including rigid and deformable parts. (3) I proposed a state-of-the-art flexible parts-fixing device named soft jig utilizing the grasping ability of a flexible gripper which draw attention in the field of current soft robotics. The soft jig can fix parts of various types and shapes during assembly operations, and thus can achieve a general-purpose assembly system that does not require the preparation of custom jigs according to the parts to be assembled.

Using a mechanical product including several rigid parts of various shapes and a deformable part, experiments were conducted to evaluate in terms of the versatility and accuracy of each method and the time required to reconfigure the system. The results prove that the training dataset collection time can be drastically reduced, feasible and easy-to-assemble sequences are generated, and fixing and assembling various parts with the proposed flexible jig are possible.

Keywords:

Agile Manufacturing, Assembly Robot, Assembly Planning, Soft Robotics, Automatic Annotation

迅速に再構成可能な組立ロボットシステム*

清川 拓哉

内容梗概

市場が求める製品の種類・量の変化に迅速かつ柔軟に対応できる変種変量生産システムが産業界で求められている．特に製造分野では，変種変量生産を行う自動組立システムを実現するために，製品の変更に迅速に対応できるシステムの実現が喫緊の課題であり，世界的に研究されている．

変種変量生産の実現には，比較的少品種の製品を大量生産可能なライン生産ではなく，製品の全組立工程を実行可能なセル生産方式が適している． そのため，仕様変更の多いカスタマイズ製品の全組立工程を，セル内で実現できる自律的なロボットシステムの活用が期待されている． 産業および研究分野の共通的な課題の解決を主目的とする，ロボットによる自動組立の世界大会においても，対象製品（ベルトドライブユニット）の部品変更を想定したルールが設けられており，ロボットシステムが部品変更に対してどのように迅速に対応するかが問われている．

本研究では，新製品の導入に対して，再構築を迅速に行うことが可能な組立ロボットシステムの構成法を明らかにすることを目的とする．まず，自動組立ロボットの研究分野における未解決の課題を整理し，対象とする課題を設定した．具体的には，認識システムの迅速な再構築手法の構築，CADモデルのみを用いる組立順序の生成手法の構築，汎用柔軟治具の開発を行い，多様な部品を含む組立製品を用いて実験を行った．

本論文で提案する手法は以下の3点から，これまでの研究とは差別化できる新規性，独自性と有用性を有する． (1)まず，近年注目される深層学習を用いた高精度な物体認識システムを製造現場において活用する際に，迅速な製品入れ替えに対応可能にするため，データセットの迅速な生成が必要である点が課題である．このデータセット生成の迅速性改善に着目して，認識システムの再構築においてボトルネックとなり得る学習画像データセットの生成を自動化する手法を構築した． (2)次に，近年の計算機の処理速度向上による3次元モデリング技術の発達の恩恵を受けて，3次元の製品設計モデルのみから，剛体だけでなく変形物体の部品情報を自動抽出して，さらに組立作業の難易度に関連する拘束状態遷移難度に基づき組立順序を自動生成することを可能にした． (3)最後に，ソフトロボティクス分野の再注目を背景に開

*奈良先端科学技術大学院大学 先端科学技術研究科 博士論文, 2021年3月17日.

発された柔軟グリッパの把持能力に活用される，ジャミング転移現象を応用して，柔軟治具という新しい部品固定装置を提案した．組立操作中における多様な形状の部品の固定能力をロボット実験により評価した．
　実験では，多様な形状の剛体と一つの変形物体を含む製品を用いて，各手法の汎用性と精度およびシステム再構成にかかる時間の観点から提案手法の評価を行った．結果として，自動画像データセット生成により，人手の生成と比較して生成時間を大幅に削減し，CADモデルのみから複数の制約条件を満たすような多目的最適化に基づく組立順序を自動で生成し，柔軟治具による部品の配置と固定および治具上での部品操作が可能であることを示した．

キーワード:

アジャイル製造，組立ロボット，組立計画，ソフトロボティクス，自動注釈

Contents

List of Figures

List of Tables

Acknowledgements

First of all, I would like to thank my advisors Professor Tsukasa Ogasawara and Associate Professor Jun Takamatsu. Thank you both for the amount of patience with my work and the time you have put into explaining difficulty in understanding my research presentations and papers.

Ogasawara-sensei, you gave me many opportunities and accepted spontaneous research activities. I am very grateful for giving me numerous challenging opportunities for five years from my Master course in NAIST. I believe that I could not have learned from all valuable experiences without your kind understanding.

Takamatsu-sensei, it has been a great help working with you for the five years in NAIST. I have learned more about doing research and its meaning from you than anyone else. You have guided me to more challenging but understandable research with a consistent story and taught me countless skills. Both of you have been very instrumental in considering my study about the robotics field I should challenge.

I would like to thank the rest of my committee members Professor Hirokazu Kato, Assistant Professor Gustavo Alfonso Garcia Ricardez, and Assistant Professor Sung-Gwi Cho for giving invaluable feedback throughout the research presentations and writing this dissertation. Your comments and criticisms really helped clear up the content and final dissertation.

Keita Tomochika, from the beginning of starting my graduate studies, I have felt a deep connection between us. Although I had a research background of biomechanics not robotics, you taught me your great deal of techniques and spent time with me on solving problems about robotics and its implementations. Thank you very much.

My research was partially supported by Exploratory IT Human Resources Project (MITOU Program) of Information-technology Promotion Agency, Japan (IPA) in the fiscal year 2018. I would like to thank Shudo-sensei for managing our project and collegues and staff in Mitou 2018 for your continuous support.

They are helpful to me to do the developments in Mitou 2018. It is because of your dedicated support that I was certified as a Super Creator from the Ministry of Economy, Trade and Industry in Japan.

My research is also based on results obtained from a project, JPNP14004, subsidized by the New Energy and Industrial Technology Development Organization (NEDO). I appreciate the continuous support.

In long-term internships in Microsoft during my doctoral course, I would like to give special thanks to Ikeuchi-sensei for giving me many valuable expericences and telling me about robotic assembly and manipulation studies conducted in the long-term history. Everything at Microsfot Research Asia in China and Microsoft Research in US was fresh and exciting for me. I also would like to thank for the supports from other developers and research staff in Microsoft Research in Beijing, Redmond, and Shinagawa.

Harada-sensei, and other members in Osaka University and AIST gave me a chance to discuss current issues in robotic assembly in several domains. The insightful discussion has motivated me continuously.

It has also incredibly motivated me that I have participated in 9 domestic and 3 international exhibitions or workshops to present my research and to discuss the technologies with industrial researchers. I would like to thank the staff of NAIST research promotion division, especially Nitta-sensei. In particular, I was very happy to have a demonstration exhibit related to my research at Consumer Electronics Show 2020 in US, and I was impressed by many large-scale demonstrations and interesting technical exhibits of other companies. This experience allowed me to reassess my inadequacy in pursuing comprehensibility in presentations and technologies in my research.

I would like to thank laboratory staff and my lab mates for the great work we have done together. The professional work of Owaki-san and Ikeda-san enabled the research activities without any inconvenience. I would like to express my respect and gratitude for the patient and hard working of the Master/Doctor course students who have worked with me, Sakuma-kun, Tariki-kun, Katayama-kun, Tatsuta-kun and so on.

Finally, I would like to thank my family for accepting me to continue such a long-term study. I would like to express my appreciation to all of you involved with me once again.

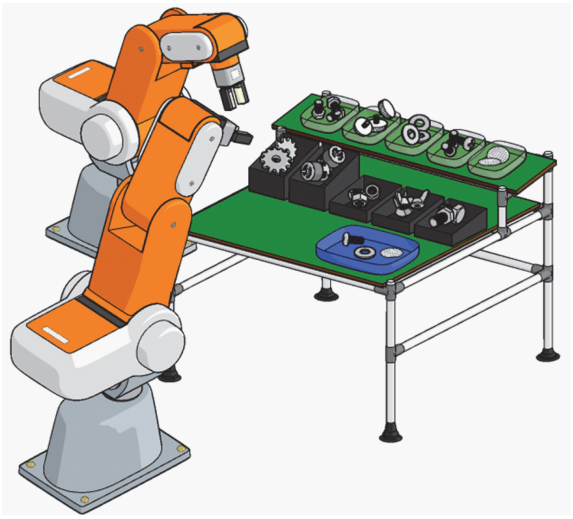
Chapter 1

Introduction

1.1. Need for Agile Reconfiguration

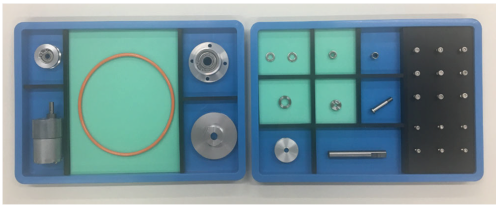
There is an ever-increasing demand for a production system that can quickly and flexibly respond to variations in both the type and quantity of products required by markets worldwide. Agile manufacturing - a recently popularised concept - has been advocated as the 21st-century manufacturing paradigm [Gunasekaran, 1999; Kim et al., 2019]. As represented by *Industry 4.0* [Rojko, 2017], *Industrial Internet Consortium* [IIC, 2017], and *Made-in-China 2025* [Li, 2018], countries around the world have proposed concepts and taken initiatives to achieve the new-type manufacturing. In Industry 4.0, the fourth revolution is characterized for promoting the mass-customized production system with the high degree of flexibility.

To achieve automatic assembly that performs the high-mix, low-volume production [Gödri et al., 2019; Karaulova and Shevtshenko, 2015; Onizawa et al., 2016] for such market volatility, a general-purpose assembly robot is required to operate a wide variety of mechanical parts of various shapes. Such a new-type production paradigm prefers, rather than a line production [Saif et al., 2014] which enables mass production of a relatively small number of different products, a cell production [Hara and Azuma, 1988; Maeda et al., 2007; Onori et al., 1997] which enables entire assembly process of customized products in one cell. Required specifications of the assembly system are frequently changed, and thus the use of an agile reconfigurable robot is expected [Gašpar et al., 2017; Karagiannis et al., 2019]. This dissertation aims to identify how to agilely reconfigure the system to handle such frequent introduction of new products.



(a) Layout example for kitting task

(b) Parts in bins



(c) Parts in trays

Figure 1.1. Kitting task of the Assembly Challenge in the WRS2018 [Yokokohji et al., 2019].

1.2. Problem Definition

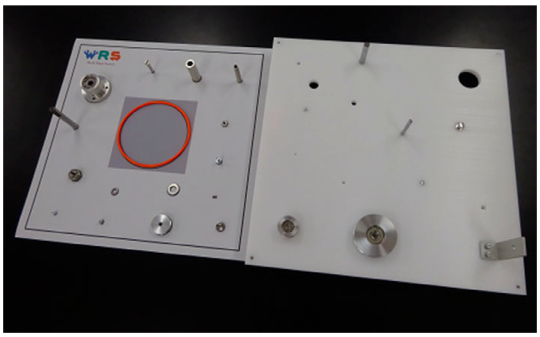
This section defines problems in this dissertation considering the findings and remaining issues in the wide range of the field related to robotic assembly.

1.2.1. New Challenge of Assembly in Worldwide Competition

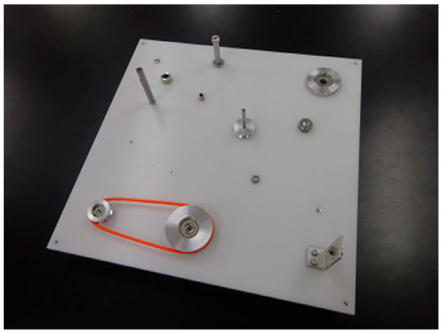
To promote breakthrough both on the technologies and research in automation and robotics, an assembly challenge in *World Robot Summit* (WRS) 2018 [WRS2018] was held. The competition has organized an assembly task setup based on open questions in the research field.

First of all, the competition assumed that the assembly task is positioned as the operation beginning from picking of parts in trays distinguished after kitting task of assembly parts from bins (Figure 1.1). After all operations including the kitting and assembly task, we obtain a designed product. The assembly task is distinguished from the task-board task consisted of assembly subtasks and disassembly subtasks executed using the task board (Figure 1.2). This dissertation focuses on this assembly task using the belt drive unit shown in Figure 1.3. Figure 1.4 depicts the relationships between the tasks.

Second, as an important challenge in the competition rules, how to deal with a robotic system to handle the changes in parts of the target product (a belt drive unit) was questioned. To simulate the new product introduced to the factory,

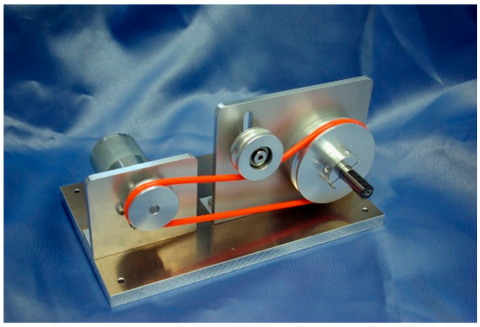


(a) Initial configuration (placement mat (left) and task board (right))

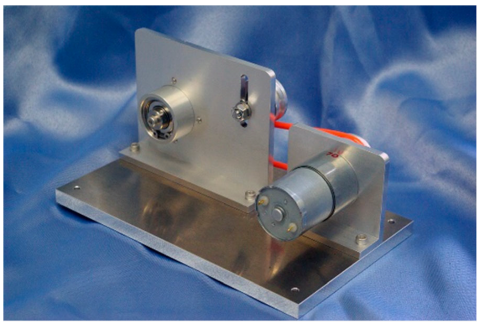


(b) Completed configuration

Figure 1.2. Task board used for the Assembly Challenge in the WRS2018 [Yokokohji et al., 2019].



(a) Front appearance



(b) Rear apperance

Figure 1.3. Belt drive unit used for the Assembly Challenge in the WRS2018 [Yokokohji et al., 2019].

several surprise parts are prepared. The surprise parts are designed differently from the parts used in the original belt drive unit, while keeping the nature of the model product. The details of the surprise parts were announced at the appropriate time just before the task starts on the competition days. In the competition, many teams ended up without touching the surprise parts at all.

Several teams have tackled the issues in aforementioned rules of the assembly challenge with novel robotic cell systems. Sloth *et al.* [Sloth et al., 2020], towards easy setup of required assembly tasks, proposed to program assembly tasks by demonstration merged with assembly primitives. Schelette *et al.* [Schlette et al., 2020] proposed a cell prodution system with collaborative robots. They mentioned multiple copies of the cell can be arranged in a highly reconfigurable, highly adaptable matrix structure in which several production flows can be handled concurrently.

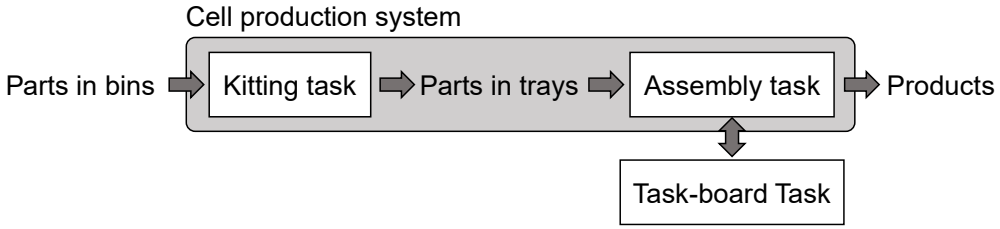


Figure 1.4. Relationship between tasks executed by cell production system.

In addition, robotic assembly systems for agile manufacturing were proposed such as a robust bin picking system using a robot hand equipped with tactile sensors [Tajima et al., 2020], a versatile robotic system which does not use jigs or commercial tool changers and no specialized end effectors [von Drigalski et al., 2020a], and a system comprising the use of only two hands specialized for assembly tasks and without the requirement of tool changers [Tennomi et al., 2020]. In this way, reconfigurable hardware structures were developed.

As another feature of the developed hardware structure, many teams implemented the assembly tasks utilizing rigid jigs and support parts to fix the parts to be manipulated by robot arms. The type and shape of surprise parts cannot be anticipated so we need a versatile jig proposed in Chapter 4.

Currently, soft robots based on flexible bodies are paid more attention than before, for its safety and versatility for environments. Applying technologies of the field of soft robotics [Lee et al., 2017b; Walker et al., 2020] are up-to-date approaches for the reconfigurability of robotic assembly. To manipulate assembly parts of various shapes in a lean manner, the robotic assembly operations with the flexible jig is more effective than automatically designing end effectors or jigs every time [Pham and Yeo, 1991].

Some approaches related to the software implementation are summarized in [von Drigalski et al., 2020b] as follows. The parts' specifications were given to the competitors one day before, and the actual parts were handed out two hours prior to their trial. This made it possible for many teams to teach some movements and perform at least some of the required assembly operations. Team Robotic Materials had primitives to grasp unknown objects and expected those to transfer directly to the surprise parts. Furthermore, they used simple assembly primitives written in Python, which they could quickly adapt. Team O2AS updated the models of the assembled parts, which updated their software with the new target positions. They aimed for mostly using the same code as in the assembly, just with updated target positions. Team SDU Robotics relied on a reusable, modular framework to reprogram or adapt workcells. The gripper-finger exchange system in combination with the 3D printers they brought to the competition allowed

them to create custom fingers for the new parts. Using play-back methods for quick programming of movements and agile pose estimation algorithms based on CAD models helped to implement assembly operations with surprise parts.

These approaches are limited to the robot motion generation, alignment of part poses, or localization of the objects [Gorjup et al., 2020], but some new parts require rethinking the assembly sequence as a high-level planning. CAD-based assembly planning proposed in this dissertation is effective for assembly planning assuming the agile reconfiguration. Automatic assembly planning including assembly sequencing [Jiménez, 2013] is a topic researched in a long-term [Homem de Mello and Sanderson, 1990; Makris et al., 2012; Rosell, 2004; Wang et al., 2009; Wilson and Latombe, 1994; Zha et al., 1998]. To achieve the assembly tasks with the robot, constrained manipulation motions [Jones and Wilson, 1996] are needed for precision operations.

Historically, assembly operation planning are based on state transitions of contact between the manipulated object and the environment around the object. Based on the contact state transitions, kinematical analysis of the manipulated object [Hirai, 1991; Hirukawa et al., 1991; Yokokohji et al., 1993; Yoshikawa et al., 1991], difinition of task primitives [Ikeuchi and Suehiro, 1994; McCarragher, 1996] and the task recognition has been tackled [Miura and Ikeuchi, 1998; Takamatsu, 2003]. Defining the contact state transitions is possible to enable the constrained manipulation motion during assembly operations. In addition, assembly parts are made by not only rigid metal materials but also soft materials such as a rubber band and a copper wire, thus handling the versatility of the assembly parts of various shapes and the materials are important in the assembly planning.

However, no research has tried sequence planning considering the contact state transitions as proposed in Chapter 3 for such various kinds of products. In addition, the parts should be recognized in each assembly task defined as the contact state transitions. However, to achieve the agile reconfiguration, automatic dataset collection method as proposed in Chapter 2 is required to replace time-consuming and man-powered dataset collection process.

1.2.2. Target Specifications of Robotic Assembly System

In this section, I discuss the relationship of the proposed method with the aspects of versatility and accuracy other than the agility of our major focus. In the first place, the ultimate goal of this research is to increase the versatility and agility while maintaining the precision of the conventional robotic assembly system. Each pair of these has its own trade-offs, thus I aim to construct a system that is a good compromise between the three axes, which is the location of the red point among the solutions that exist within the blue phase shown in the upper

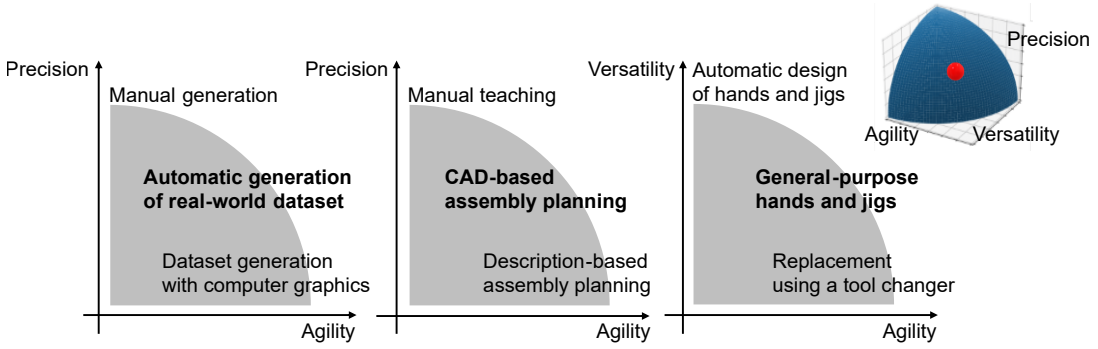


Figure 1.5. Target specifications in tradeoff relationships between agility, precision, and versatility of assemblies.

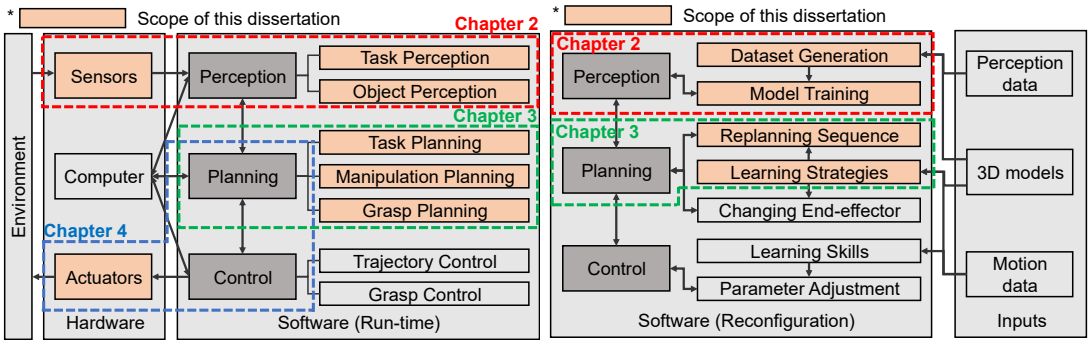
right corner of Figure 1.5. In fact, I placed each of the proposed methods for recognition, planning, and execution on the graphs with each two axes, taking into account the differences with other existing methods.

First, in terms of the proposed automatic generation of real-world dataset, I positioned it against other methods in terms of precision against the agility. The method using only computer graphics (CG) data is faster because it does not require moving objects or cameras in the real environment, but it is less accurate when no images of the real environment are used.

Second, the proposed automatic assembly sequence generation with CAD was also evaluated in terms of precision against the agility. In the case of the sequence generation from assembly instructions without using CAD, since we could obtain the assembly order from the instruction by applying natural language processing, the computation cost is relatively low because it is possible to avoid running optimization calculations based on the parts geometries. On the other hand, since the information contained in the instructions is secondary and tertiary data that possibly contains human errors, it may be possible to generate a more precise sequence from a 3D CAD model that has a uniform format and relatively less room for such errors.

Third, for the general-purpose hand and jig, I considered the versatility and the agility. First of all, in the case of automatic design of them, the time required for the manufacturing of the automatically designed hand and jig is problematic. On the other hand, in the case of using a tool changer, it is faster than automatic design as long as when multipurpose hands and jigs are readily available, but not as versatile as automatic design or the proposed soft jig. In this way, I positioned each proposed method devided in the recognition, planning, and execution.

Figure 1.6 shows the configuration of the robotic system for runtime execution and system reconfiguration assumed in this dissertation. Finally, they are set as



(a) Run-time execution (b) System reconfiguration

Figure 1.6. Overview of assumed robotic system configuration.

the problems that we make the basic configuration (Figure 1.6(a)) more general-purpose than the previous systems, and make the configuration for reconstruction (Figure 1.6(b)) more agile than the previous systems.

1.2.3. Scope of Dissertation

This dissertation tackles issues in the following three topics: (1) a part recognition system that can be quickly retrained, (2) a CAD-based assembly sequence generation method for products consisting of many parts including deformable objects, and (3) a general-purpose parts-fixing tool and a assembly strategy with the tool.

To reduce the time-consuming manual teaching process and programming effort in the automation of the assembly system, I give much attention to reusability of the data given to a robot system such as a CAD model, and the versatility and flexibility of the robot system such as a flexible jig and a faster trainable vision system.

The following topics lie outside the scope of this dissertation: the reconfigurable robotic systems by robot-robot [Argall et al., 2009; Marvel et al., 2018; Zhu and Hu, 2018] and human-robot [Raessa et al., 2020; Tsarouchi et al., 2017; Weckenborg et al., 2020] collaborative systems, the concrete strategies for assembly tasks such as high-precision grasping and assembly tasks with sensors [Li and Qiao, 2019], peg-in-hole [Park et al., 2017; Shibata et al., 2020; Watson et al., 2020], dual-arm manipulation motion planning [Harada et al., 2014].

1.3. Dissertation Outline

1.3.1. Relation Between Proposals

Figure 1.6 shows the configuration of the robotic system aformentioned. The hardware and software components related to chapters 2, 3, and 4 are shown as red, green and blue boxes, respectively. This dissertation concentrates on the versatility of the components and the agility of the system configuration that still need for the automated assembly system that can handle the surprise parts.

1.3.2. Chapter Description

Chapter 2 presents two frameworks to train the vision system faster than manual image dataset collection. To collect a human-annotated dataset for training deep convolutional neural networks is a very time-consuming and laborious process. To reduce the burden, I first proposed an automated annotation by placing one visual marker above the detection target object in the training phase. Since the target object poses can be calculated from detection results of the visual marker, once we link the relative object pose and object size to the visual marker, annotation data such as a label, a bounding box and the object pose can be obtained automatically. In the first approach, occasionally the marker hides the object surface. To avoid this issue, I propose placing a pedestal with multiple markers at the bottom of the object. If we use multiple markers, the object in the image can be annotated even when the object hides some of the markers. Besides that, the simple modification of placing the markers on the bottom allows the use of simple background masking to avoid the neural network learning the remaining markers in the training image as a feature of the object. Background masking can completely remove the markers during the training process. Experiments showed the proposed vision system using our automatic object annotation outperformed the vision system using manual annotation in terms of object detection, orientation estimation, and 2D position estimation while reducing the time required for dataset collection from 16.1 hours to 7.30 hours.

In the other framework, I developed an automatic image dataset collection with a robot arm to change poses of a camera and with a rotating stage to change orientations of a target object. The target issue in this framework is to remove differences in the appearance of target objects imaged in the two scenes, which are the dataset collection scene and real work scene such as a factory. If the differences remain, the performance of the trained detection system may decrease. Our proposed method focuses on filling the gaps in terms of the two differences in illumination and background. To do this, I propose to apply histogram matching and background synthesis to the source target object images. In the experiments

in a man-made workplace for a factory line, the detection results demonstrate that the proposed method enables us to obtain the performance higher than the other methods without considering the differences.

Chapter 3 explains the methods to plan the assembly tasks especially assembly sequences of parts, so the chapter is highly related to the task planning. I describe the formulation of how to search an optimal solution of feasible assembly sequences that satisfies the insertion relations between parts and the low difficulty of constraint state transitions. Aiming to generate easy-to-handle assembly sequences for robotic assembly, I tackle the assembly sequence generation by considering two tradeoff objectives: (1) insertion conditions and (2) degrees of constraints among assembled parts. I propose a multiobjective genetic algorithm to balance these two objectives for generating assembly sequences. Furthermore, the method of extracting part relation matrices including interference-free, insertion, and degree of constraint matrices is extended for application to 3D computer-aided design (CAD) models, including deformable parts. The interference of deformable parts with other parts can be easily investigated by scaling the deformable shapes. This automatic extraction of each part information can be considered related to semantic understanding since the high-level information about two-part relationships related to the interference, insertion, and constraint are extracted from the geometries obtained from the CAD model which is low-level data. I conducted a simulation experiment using the proposed method. The results show the possibility of obtaining Pareto-optimal solutions of assembly sequences for a 3D CAD model with 33 parts including a deformable part. This approach can potentially be extended to handle various types of deformable parts and to explore graspable sequences during assembly operations so this method can be used for grasp planning.

Chapter 4 deals with the versatility for object shapes during assembly operations on run-time. The robotic assembly system needs to manipulate an assembly part onto another fixed object (e.g. another part or the environment) such as peg-in-hole, screwing, and placement operations. To design a general-purpose assembly robot system that can handle objects of various shapes, I propose a soft jig capable of deforming according to the shape of the assembly parts. The soft jig is based on a jamming gripper (e.g. [Brown et al., 2010]) previously used for robot manipulation as a general-purpose robotic gripper developed in the field of soft robotics. The soft jig has a flexible membrane made of silicone, which has a high friction, elongation, and contraction rate for keeping parts fixed. The inside of the membrane is filled with glass beads to achieve a jamming transition. The usability of the soft jig was evaluated from the viewpoint of the versatility and fixing performance for various shapes and postures of parts in assembly operations. Since the soft jig allows for the placement and fixation of objects of

various shapes and sizes in various poses, the choice of assembly operations is widened. It is necessary to consider a specific manipulation planning for the use of the soft jig. This dissertation examines a method for determining the pose of object placement on the soft jig and its manipulation planning with the proposed flexible jig.

As a closure to this dissertation, Chapter 6 concludes the contributions of this dissertation and mentions the future direction of my research to achieve fully automated robotic assembly cell production.

1.4. Contributions

The major contributions of this dissertation are:

- Fully automated annotation methods with visual markers that were diminished later is effective for generating annotated images rapidly. To obtain the real-world image datasets effectively, unbiased dataset collection was proposed and evaluated. An automatic image dataset collection method with a small robotic arm and with a rotating stage enabled us to collect multi-view object images more quickly. A domain adaptation method with several image processing techniques were evaluated by training conventional deep learning-based object detection methods.

- CAD-based assembly planning methods for searching feasible assembly sequence that interferences between parts does not occur, satisfying insertion relationships, and with low difficulty of constraint state transitions. I designed the fitness functions for a genetic algorithm. To achieve calculating the fitness values based on the geometries extracted from the CAD model, I proposed automatic extraction methods of part information for not only rigid object but also the deformable objects. The proposed sequence generation method was evaluated using a product including many parts and a deformable rubber band.

- A state-of-the-art flexible part-fixing device named *soft jig* was proposed. The usability, flexibility, and fixing capability of the soft jig was evaluated in experiments using physical robotic arms. The fixing capability based on a jamming transition was clarified on the experiments of object placement and fixation against the application of external force.

1.5. Publication Note

Parts of the works done in this dissertation have appeared in previous publications. The proposal of faster trainable vision system was published in [Kiyokawa et al., 2019b], since then the method has evolved so far [Kiyokawa et al., 2019a]. The part of the 3D CAD-based assembly planning in Chapter 3 was covered by [Kiyokawa et al., 2020b]. The manipulation with flexibility in Chapter 4 was presented in [Kiyokawa et al., 2020a].

Chapter 2

Faster Trainable Vision System

2.1. Introduction

The best performing vision system in automated factories exploits the deep learning-based methods, such as *Deep Convolutional Neural Networks* (DCNNs) to simultaneously detect various products. Deep learning-based object detectors are able to infer the location and the label of objects in images even with such a variety of appearances [Girshick, 2015; Liu et al., 2016b; Redmon and Farhadi, 2018; Redmon et al., 2016; Ren et al., 2015; Tan et al., 2020; Zhao et al., 2019a]. Recent studies have focused on detection and pose estimation of multiple objects with the DCNN [Pathaka et al., 2018; Schwarz et al., 2018; Zeng et al., 2017; Zhao et al., 2019b]. Training of such models typically requires large amounts of annotated training data [Cai et al., 2017; Krasin et al., 2017; Real et al., 2017; Rennie et al., 2016] because the DCNN has a large number of parameters to be optimized in the training process. Manual annotation of the training dataset is a time-consuming and laborious process, and costly to obtain. As an example of the annotation effort, as shown in Figure 2.1, bounding boxes are drawn and labels are assigned for each object in every image.

An efficient way to make a large amount of training data is desirable for a deep learning-based vision system. Especially for high-mix low-volume production, the effort of this manual annotation cannot be ignored for the following two reasons. First, a large number of product types should be coped with. We need to prepare a moderate number of images for each product for training. Second, in this type of production, the lifecycle of the products is typically short. In each cycle, we need to train the network again.

This chapter first describes the proposed automatic annotation methods for real-world objects. In the methods, manual object arrangement and manual image

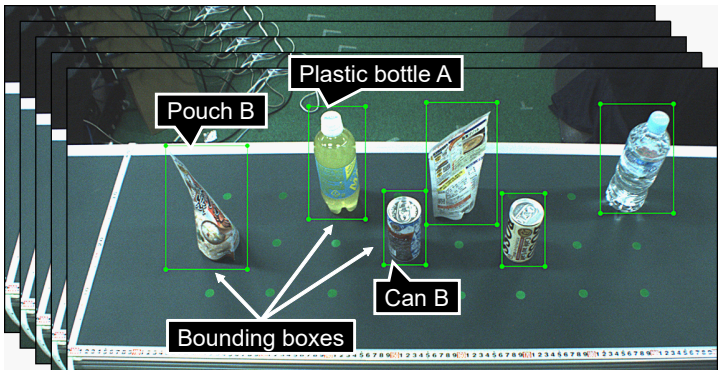


Figure 2.1. An example of the annotated image. The box around each object shows the bounding box annotated by humans. The label, the orientation, and the 2D position should be annotated for automated factory applications.

capturing are needed. Second, I propose an automatic image dataset collection with an automatic image capturing system and propose a domain adaptation method. The method enables the use of the collected images to train a vision system for a target domain.

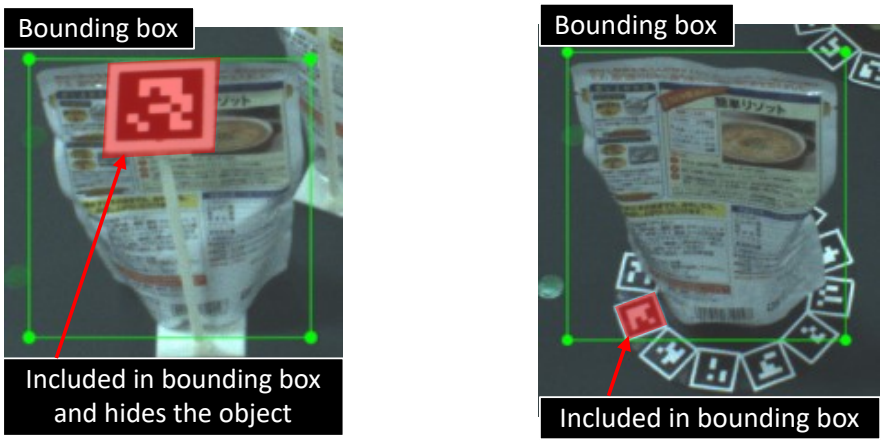
2.1.1. Real-world Collection with Automatic Annotation

I first propose a fully automatic object annotation to real-world images. The proposed method does not require any manual annotation. For the proposed method, in only the training phase, we use visual markers for object annotations. We associate the visual marker [Kato and Billinghurst, 1999] with each object and capture both of them in the same image to track the objetcs. The proposed algorithm then labels the object IDs using the marker IDs, calculates the 3D poses of objects by estimating the pose of the marker, and generates a bounding box around each object based on the object geometry and the pose of the corresponding marker.

To achieve the proposed method, we need to solve the following three issues:

1. Depending on the arrangement of the visual markers, the marker may hide the target object and reduce the visible area of the object (Figure 2.2 (a)). It is necessary to consider the arrangement of visual markers.

2. Visual markers are usually included in the images cropped with the bounding boxes (Figure 2.2 (a) and Figure 2.2 (b)). The neural–network erroneously learns that the markers are parts of the object appearance.



(a) Single marker (b) Multiple markers

Figure 2.2. Two problems that the marker hides the object and that the marker are included in the bounding box.

3. Considering the performance of the vision system, the diversity of the training dataset of real-world images should be kept while reducing the number of images to reduce the effort of data collection.

The first issue occurs where the physical visual marker hides the object surface. Since the proposed method uses visual markers for annotation, the markers always appear in the images without hiding. To avoid hiding the visual markers by the object, one idea is to put the marker near the camera, *i.e.*, the above of the object. Though we can place the marker above the object to make the marker visible, occasionally the marker hides the object surface. Such a hinder reduces the observed appearance feature of the object and thus may deteriorate the performance of the vision system. To avoid this issue, I propose to place a pedestal with multiple visual markers on the bottom of the object. The use of multiple markers makes the proposed automatic annotation robust; the proposed method works even when several markers are not detected or occluded. Our experimental results demonstrated that, comparing to the pedestal with a single marker placed at the top, the proposed pedestal design improves the performance of the vision system in terms of the object orientation estimation.

The second issue is related to the learning when we use the images including the visual markers. The visual markers in the bounding box tend to confuse the learning of object features. Since the product does not have markers in real use, this deteriorates the performance of the vision system. To reduce this deterioration, I propose using a background image to mask the region of the visual markers on the pedestal in the images. I experimentally prove that this simple

masking approach avoids erroneous learning.

The third issue concerns that not only must the training dataset be collected quickly, but also the usefulness of the collected data should be considered. To reduce the amount of the training data without sacrificing the performance of the detection, it is necessary to avoid capturing the object images with similar poses, while it is necessary to collect the object images with various poses. As a solution to the issue, I propose an unbiased dataset collection method by showing the dataset collection progress to dataset collection workers.

During collecting training data, the worker arranged the objects manually on an arbitrary place in the specific region on the conveyor belt where all the target objects can be captured by the camera. If the system shows the worker the histogram of the captured poses, the worker can choose the poses of which the object is not captured.

I experimentally evaluated the effectiveness of the proposed method by comparing the times needed for manual annotation and for the automatic annotation. I also measured the improvements in the performance of the robot vision system using the proposed approaches. I assume a vision system used for the factory robot capable of grasping various types of objects and picking them up according to the object orientation. The experiments evaluate the accuracies in the following three terms: (1) detection of multiple objects on the conveyor belt, (2) estimation of 2D object positions on the conveyor belt, and (3) estimation of object azimuth angle.

Our contributions in the framework for real-world collection with automatic annotation are threefold:

1. I propose to place a pedestal with visual markers at the bottom of the detection target object to avoid the issue of obstructing learning the object appearance.

2. I propose background masking of the marker area in the training image to avoid learning the visual markers as the parts of the object appearance.

3. I propose to show the dataset collection worker the collection progress in real-time by taking advantage of visual markers. Visual markers fixed to the object enable us to obtain the object pose data immediately after detecting the markers by the camera.

2.1.2. Fully Automated Collection with Domain Adaptation

As another framework, I develop an automatic system for image dataset collection. To use the system, a target object is placed on an automatic rotating

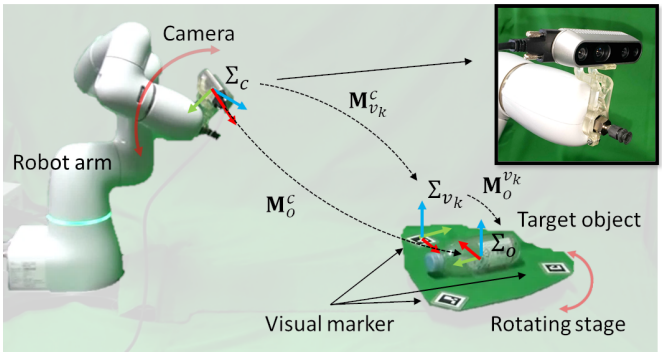


Figure 2.3. Robotic training dataset collection system that facilitates image capturing and automatically annotates labels and bounding boxes.

stage and imaged from multiple viewpoints using a hand–eye robot arm shown, as in Figure 2.3. The robot arm and rotating stage are automatically controlled while capturing images. The aforementioned proposed automatic annotation method utilizing the fiducial marker detection [Kato and Billinghurst, 1999] is applied to the images captured. To train the DL-based object detector, we place the collection-target object on the rotating table for image capture. However, we do not have to manually annotate the images.

Although object images in the real-world can be easily provided, they often appear differently from items found in the working environment. Thus, detection performance can decrease when collecting images without consideration for adaptation methods.

The industrial workplace exists in an indoor environment for this study. Thus, it can be fixed in terms of illumination and background. I propose methods to reduce the differences easily and effectively for such conditions.

This research focuses on two domain differences in terms of the illumination and the background between the image dataset collection environment and the real work environment in industry. First, we adjust the object size in the image to be as close as possible to the real one in the real work scene. Subsequently, we apply histogram matching (HM) to images using a RGB color space to reduce illumination differences. Based on our qualitative observations for RGB histograms of the object images captured in the real work environment, we apply histogram smoothing for the collected images to further make the RGB histogram resemble the destination images. Furthermore, to reduce the differences of background conditions, we use background-synthesized and histogram-matched images as the training images.

Our contributions in the framework for fully automated collection with domain adaptation are twofold:

1. The proposed robotic training dataset collection system uses a hand–eye robot arm, a rotating stage, and visual markers to facilitate object-image capturing from multiple viewpoints and with multiple annotations. The automatic annotation is calibration-free on the relative poses of the target object placed on the stage, because detection results of visual markers are used to annotate the object pose. The time required for the proposed automatic collection is 12.3 s: 93.4% faster than prior methods.

2. I propose a simple but effective object-image dataset adaptation method for collected images. As a benefit of applying object scaling, RGB-HM with histogram equalization (EQ), and background synthesis (BS) for the collected images, we achieve improved object detection accuracy. I further propose the addition of a small real-world dataset captured in the real work scene to the domain-adapted dataset. Training with this dataset achieves a detection accuracy of 79%, which is 39% higher than using the original one that lacks domain adaptation and real-world images.

2.2. Related Work

There is a vast literature in the area of object detection and pose estimation including effective dataset collection. As a brief review, I cover related research focusing on the following three aspects, which are previous approaches with the same scope as ours and other research methods tackling the same issues as ours.

1. The first aspect is related to the methods to achieve a vision system for a picking robot in automated factories. I explain the latest approach for detecting objects and estimating the object pose in real-time, since a robot in the factory line can grasp and pick up the target object in an arbitrary pose.

2. The second aspect is related to effective methods to generate training dataset for the vision system. The time-consuming dataset collection is an essential problem to solve to quickly deploy a vision system for a factory line conducting high-mix low-volume production.

3. The third aspect is related to domain adaptation to utilize datasets collected in a domain for another target domain. To achieve automatic image dataset collection in a domain other than the target domain, I investigate the domain adaptation methods.

2.2.1. Vision System

One approach for object detection or pose estimation uses RGB-D sensors with 3D object models (CAD [He et al., 2017; Song et al., 2017; Wohlkinger et al., 2012] or 3D reconstruction [Zhu et al., 2014]). In the automated factories, a 3D object model is often available. These methods use disparity images obtained from an RGB-D sensor and match the images with the models. Due to the huge search space involved in image matching with the 3D object models, these methods are computationally expensive.

Various types of pipelines for RGB-based object detection and pose estimation have been proposed [Brachmann et al., 2016; Do et al., 2018; Kehl et al., 2017; Li et al., 2018; Rad and Lepetit, 2017; Sundermeyer et al., 2018]. Although these pipelines do not use the 3D object models unlike the RGB-D based template matching, the pipelines have been effectively used images rendered the known colored 3D object models in the training process and/or the online process. A single-shot approach was proposed [Tekin et al., 2018] for simultaneously detecting an object in an RGB image and predicting its 6D pose. As the performance, it is much faster,*e.g.,* 50 - 94 fps (depending on the image resolution) on Titan X (Pascal) GPU and suitable for the real-time vision system. An out-of-the-box CNN-based architecture using vision data only was proposed [Xiang et al., 2018], which performs three different tasks that lead to the 6D pose estimation, *i.e.,* semantic labeling, 3D translation estimation, and 3D rotation regression. By using a large scale video dataset for 6D object pose estimation, the convolutional neural–network can accurately estimate the 6D pose and handle occlusions and objects in cluttered scenes. They used the YCB dataset [Calli et al., 2015] including 133,827 frames for 21 objects in 92 videos.

2.2.2. Quickly Generating Training Datasets

Since while deep learning-based vision systems become faster and able to detect more categories of objects, the cost of humans manually annotating data remains very high, there are two major efforts to easily collect the large datasets. One approach is (1) data augmentation to enrich the image datasets for improving generalization ability of deep learning models and the other is to deal with (2) simplification of the labor-intensive annotation process with human-in-the-loop automatic annotations. This research is related to both of them.

Data augmentation

In the research of (1), a strategy utilizes a pre-trained detector. The most common approaches have trained a DCNN using a manually annotated small-scale dataset,

and then labeled the remaining data using the pre-trained DCNN [Adhikari et al., 2018; Sun et al., 2017]. Similar to the data augmentation approach, even using such a pre-trained model, rendered images can be used for training the actual objects [Mitash et al., 2017].

Another approach to augment the dataset includes changing the appearance and the background of objects in the image [Cubuk et al., 2019; Montserrat et al., 2017]. To increase datasets, random image cropping and patching have been done to improve the accuracy of the classification of images [Takahashi et al., 2018]. Random erasing has been tried to reduce the risk of over-fitting and makes the model robust to occlusion [Zhong et al., 2020]. They randomly assigned the pixels within the selected region of an arbitrary size with random values. An automatic search method for data augmentation policies directly from a dataset (*AutoAugment*) has been proposed in [Cubuk et al., 2019]. Each policy expresses several choices and orders of possible augmentation operations, where each operation is an image processing function (e.g., translation, rotation, or color normalization). *FastAutoAugment* in [Lim et al., 2019] is the improved policy extraction method and significantly faster than AutoAugment requiring thousands of GPU hours even for a small dataset.

Other approaches use object images such as rendered images using a 3D CAD model [Kehl et al., 2017; Peng et al., 2015] or cropped images from existing object recognition datasets [Georgakis et al., 2017]. A framework called *RenderGAN* was proposed as a novel extension to the GAN framework [Sixt et al., 2018]. The framework using GAN [Goodfellow et al., 2014] was succeeded in generating more realistic rendering samples from a basic 3D model. The reality of the generated images deeply depends on the quality of the rendering. I believe the policy of collecting a hand-crafted real-world images dataset such as ours can improve the performance of the vision system, since the dataset faithfully represents *real* situations.

Human-in-the-loop automatic annotation

In the research of (2), the strategy is annotations using an easy-to-use tool. The research [Su et al., 2012] have used a strategy to crowd-source bounding box annotations using a tool capable of drawing and verifying the bounding boxes with human workers. In a similar way, *LabelMe* [Russell et al., 2008] has been proposed a web-based tool to create boundaries around each object in an image and share the annotations on the web. To make human annotation easier, effective but easy-to-use annotation tools [Maninis et al., 2018; Papadopoulos et al., 2017] are proposed. The tool proposed in [Papadopoulos et al., 2017] asks the annotator to click on only four physical points on the object. The humans still spend their time on the annotation with the tools.

In a similar manner, bounding box and polygonal annotations for detection and instance segmentation are effectively conducted by interactive clicks with human annotators. The research [Russakovsky et al., 2015] has proposed a human-in-the-loop system that questions a human and makes the human answer. *Curve-GCN* is proposed to automatically predict the vertices of instances in the images [Ling et al., 2019]. Clicking the regions with wrong annotations in the bounding box generated for each instance can be done in [Benenson et al., 2019]. These human-in-the-loop polygonal annotations take only a few seconds for each image, but they also require corrective clicks for the vertices, owing to the need for annotation quality assurance.

Another interesting approach is the use of an RGBD sensor [Suchi et al., 2019] and visual markers [Akizuki and Hashimoto, 2019; De Gregorio et al., 2020] to automatically segment objects from the background. These approaches are like ours. However, in the previous approaches, the automatic collection of multi-view object images and their domain adaptations were out-of-scope. Our robotic training dataset collection system of multi-view images gives the dataset variety and quantity and is useful when training the garbage detector to handle various appearances. Image adaptation methods of reducing the differences of domains are necessary to enable faster image collection.

As another approach, closed-loop workflows were proposed [Adhikari et al., 2018; Papadopoulos et al., 2016], that include a human verification process to re-train the DCNNs. In this approach, workers verify whether a bounding box is correctly drawn and/or whether all object instances have bounding boxes. Since the proposed methods still rely on the effort of human annotators, it is not a fully automated annotation process such as ours. In the research method [Maiettini et al., 2017], a humanoid robot can be used to annotate through interaction with humans. In this approach, a human showed an object to the robot, and then the robot tracked the object as it was moved by the human. Although this method can automatically collect annotated images including objects in an enormous variety of backgrounds, the method requires effective and efficient human work. Unfortunately, compared to our approaches, these methods still require human intervention for the annotation process.

2.2.3. Domain Adaptation

Despite the many ideas explored, the predominant datasets were built by humans using bounding boxes or polygonal masks [Cordts et al., 2016; Deng et al., 2009; Everingham et al., 2015; Lin et al., 2014]. Our proposed method can automatically annotate object images without human intervention. Because there are differences in object appearance between the dataset collection environment

shown in Figure 2.3 and the real work environment, the collected dataset using the robotic collection system could not be directly used to train the object detector.

Domain adaptation is a specific scenario in transfer learning that can be used to effectively remove domain differences. For example, crawling from an image search is a fast image dataset collection method. Domain adaptation has been shown to be effective for the transfer learning of models in different computer vision tasks, including image classification [Tzeng et al., 2017], object recognition [Gopalan et al., 2011], object detection for indoor kitchen scenes [Georgakis et al., 2017], outdoor scenes [Hsu et al., 2019], water-colors [Inoue et al., 2018], and semantic segmentation [Luo et al., 2019].

The research in [Georgakis et al., 2017] tackled an issue like ours. To automatically generate image datasets that emulate real environments, they superimposed two-dimensional images of textured object models into images of real indoor environments reflecting a variety of locations and scales. They verified the efficacy of a seamless cloning (SC) method to mitigate the effects of changes in illumination and contrast. They also verified an object–scaling method that used the depth of the selected position of a real household environment.

In this study, I tackle the issue of domain adaptation for a collected object image dataset ourselves so that it can be adapted to a real waste-sorting problem. For this reason, I create a object image dataset using images of 33 aluminum cans, 33 glass bottles, and 33 plastic bottles.

I also strongly support the efficacy of domain adaptation for the real work environment. In particular, I evaluate more methods to mitigate the changes of object-size appearance, image illumination, contrast and background.

2.3. Real-world Collection with Automatic Annotation

2.3.1. Efficient Dataset Collection

At first, I describe what we must do before collecting training images. Second, I describe some ideas for improving the efficiency of image collection and describe a method for automatically annotating the collected images. Finally, I describe the object detection method used.

Preparation

Before all the processes, we obtain the camera calibration parameters $\mathbf{K}$ used to estimate several pixel positions in some processes, and then capture a background image to mask the markers. Also, we need to determine an approximate shape of the object in order to generate a bounding box around the object boundary.

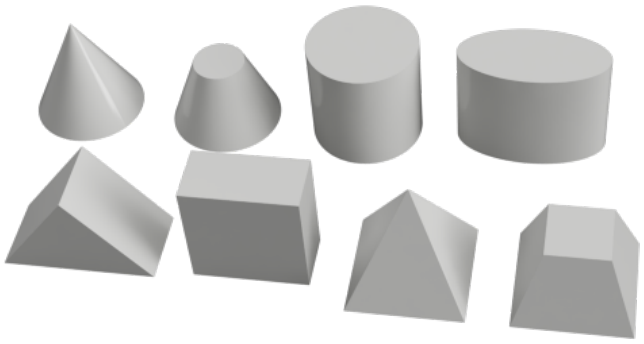


Figure 2.4. Primitive shapes used for approximation of the object shapes.

I define the approximate shape as the concatenation of two primitive shapes of the upper part and the lower part (I define the intermediate part as the third primitive shape if necessary). As shown in Figure 2.4, the primitive shapes include rectangular parallelepipeds, pyramids, cylinders, and cones. Figure 2.11 (a) shows the approximate shape of the plastic bottle. Such the plastic bottle can be expressed as a combination of a cone and a truncated cone.

Effective Image Collection

As the first step of the training process, we prepare the pedestal equipped with visual markers as shown in Figure 2.5. To eliminate hiding the object by the markers, we place the pedestal at the bottom of each object. Further, by attaching the multiple markers onto the pedestal so as to encircle the object, several markers are visible in any poses of the object. The examples of the pedestal-attached-objects are shown at the bottom of Figure 2.6. We prepare several different sized pedestals to handle the variety of the object size. Fortunately, the object size used in the experiments was just a little different.

After preparing the pedestal-attached-objects, we capture the objects in various poses in real environments. To efficiently collect the real-world object images unbiased by the object poses, the display shows the workers the collection progress. The proposed method, by detecting markers, enable us to obtain the histograms of object poses in real time. By showing the histograms of the collected object poses, the workers understand the necessary images of the object with the target poses.

The overview of the display is shown in Figure 2.7. Figure 2.7 shows the display example of the radar chart. One axis of the chart indicates one pattern of the object orientation. The axis of the right direction indicates 0° and the other axes are drawn in increments of 45° counterclockwise. The axes show the 8 object

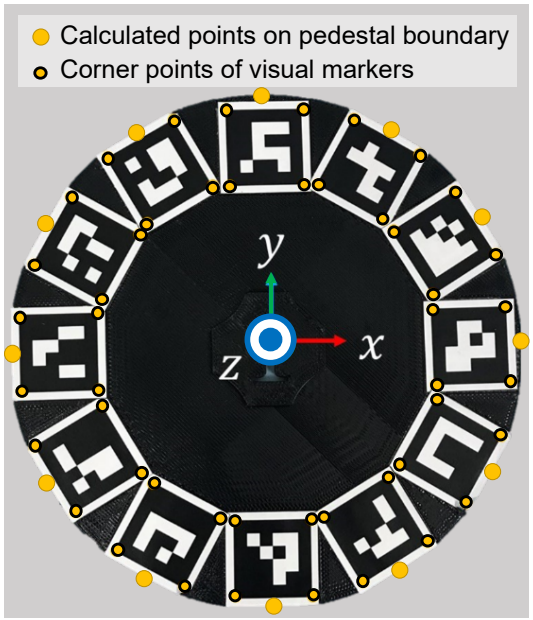


Figure 2.5. Pedestal to place a visual marker used in automatic annotation.

orientation patterns and each scale shows the number of collected object images in each orientation pattern. The positions of the radar charts correspond to the $3 \times 8 = 24$ positions on the conveyor belt where we placed objects.

2.3.2. Annotation with Visual Markers

Noise-Masked Single Marker

As shown in Figure 2.9, to generate training data without manual annotation, we place a visual marker on a 3D printed pedestal for each object. When attaching the marker and the object to the pedestal, the relative pose between the object and the marker is determined and the marker ID is mapped to the defined object label. Therefore, based on the ID and the pose of the detected marker in the image, we can obtain the corresponding object's label, position, and orientation as training data.

To hide the marker from the detector, I propose the method to overwrite all of the pixels of the marker area with a random noise image with uniformly distributed RGB values. Since the randomness provides the detector no information, we successfully lead the detector concentrating on learning the object appearance only.

To generate the bounding box shown in Figure 2.9, the target object is ap-

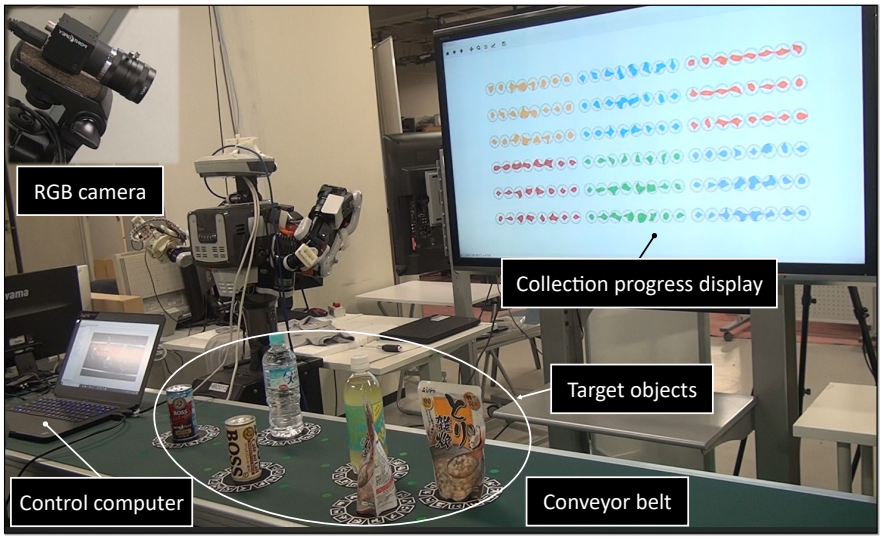


Figure 2.6. Environment on collecting the training dataset.

proximated with a rectangular parallelepiped. The height of the rectangular parallelepiped is defined as the distance from the bottom to the highest point of the object. The long side of the bottom surface of the rectangular parallelepiped is the distance between the furthest points in the outline of the bottom surface. The length of the short side of the bottom surface is determined by the length of the line segment connecting the furthest points such that the short side becomes a vertical line segment of the long side.

Then we calculate the bounding box size based on the object height h_o, the width w_o, and another side l_o. The width w_b and the height h_b of the bounding box are defined by

$$w_b = w_o|\cos\theta| + l_o|\sin\theta| + m_b, \tag{2.1}$$
$$h_b = w_o|\sin\theta| + l_o|\cos\theta| + h_o|\sin\phi| + m_b, \tag{2.2}$$

where m_b denotes the margin of the bounding box needed to reliably surround the objects, θ denotes the object orientation around the vertical axis, and ϕ denotes the angle of the camera with respect to the vertical axis.

We annotate the 2D positions of the objects on the conveyor belt in the image. The 2D position is defined as the center of the bottom surface of the object and is collected using the following procedure.

1. Measure the height of the pedestal and the position of the object on the pedestal.

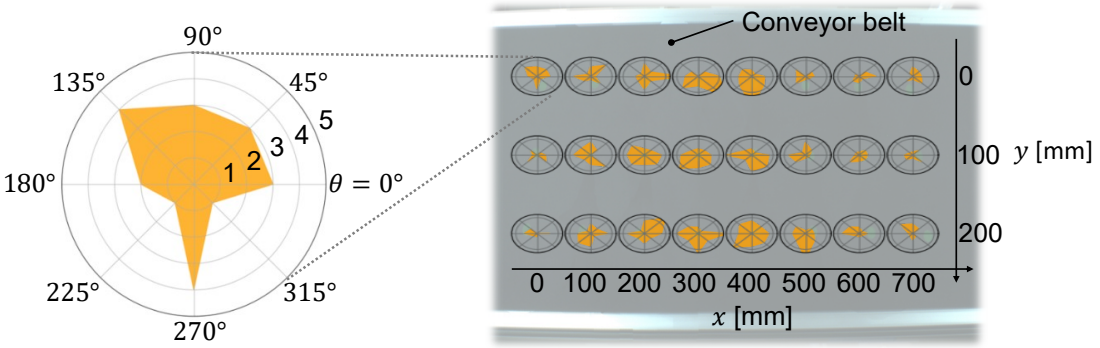


Figure 2.7. Example of the radar chart. The axes show the object orientation patterns and each scale shows the number of collected object images in each orientation pattern. The positions of the radar charts correspond to the positions on the conveyor belt.

2. Calculate the 3D position from the center of the marker to the center of the bottom surface of the object using these measured parameters.

3. Obtain the homogeneous transformation matrix between the camera coordinate system and the marker coordinate system.

4. Calculate the homogeneous transformation matrix between the object coordinate system and the camera coordinate system using the results of step 2 and 3.

5. Obtain the pixel position of the object by projective transformation using the result of step 4.

The center position of the bounding box in the image is the position of half the height from the bottom center of the object. The 2D position of the object for the training dataset is the center of the object bottom on the conveyor belt.

Background-masked Multiple Markers

According to the flow of Figure 2.8, the training dataset is extracted from each collected RGB image. The algorithm mainly consists of five processes for automatic object annotation. First, based on the detected marker IDs, the corresponding object labels are extracted from a database. Then, the next process determines the pixel positions of the four vertices of each bounding box enclosing all the pixel positions of the object boundary. Figure 2.11 (b) shows the generated bounding box. Simultaneously, the algorithm estimates the azimuth angle and the 2D position of the object. Finally, the background image is applied to all marker regions to generate the training image deleted with the pedestal region.

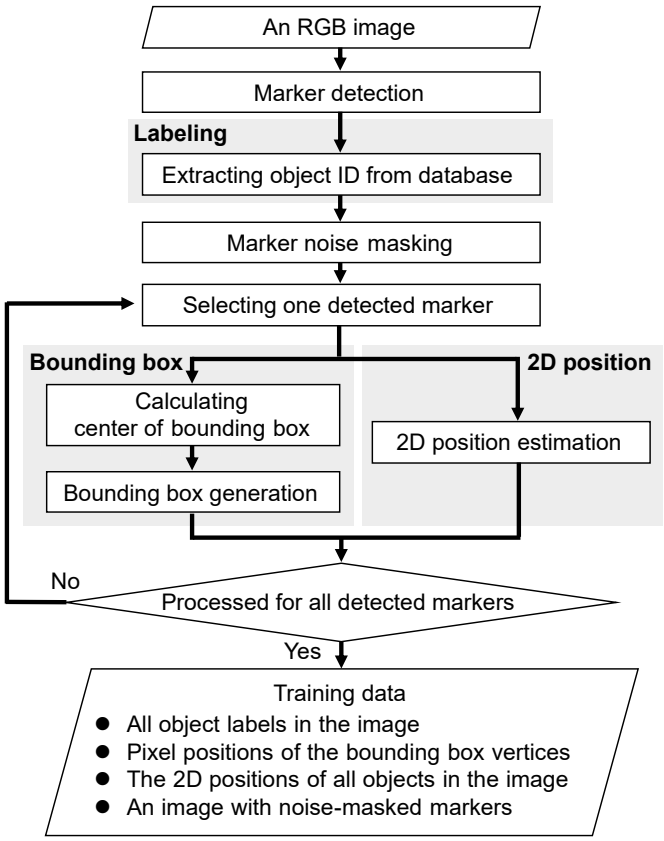


Figure 2.8. Flow to generate the training dataset. This flow mainly includes five processes to obtain object labels, bounding boxes, object orientations, 2D object positions, and images for training.

We estimate 3D object poses (the orientation and the 2D position) as annotations used for training the vision system. I define the coordinate systems and transformations in the formulation for calculating the pose of the object as shown in Figure 2.12. The possible configuration of the pedestal-attached-object is generally expressed as the homogeneous transformation matrix from the object coordinate system Σ_o to the conveyor belt coordinate system Σ_b. The parameter $\boldsymbol{r}$ represents the position vector. The parameter θ represents the scalar value of the azimuth angle. The possible configuration of each visual marker k is expressed as $\mathbf{M}^c_{v_k}(\boldsymbol{r}^c_{v_k}, \theta)$, which represents the pose of the visual marker coordinate system Σ_{v_k} $(k = 1, 2, ..., N$, where N is number of markers) with respect to the camera coordinate system Σ_c. In the same way the pose of Σ_{v_k} with regard to Σ_o will be denoted by $\mathbf{M}^{v_k}_o$. The matrix $\mathbf{M}^{v_k}_o$ is defined when attaching the marker onto the pedestal. The pose of Σ_c with regard to Σ_b denoted by $\mathbf{M}^c_b$ is defined after set

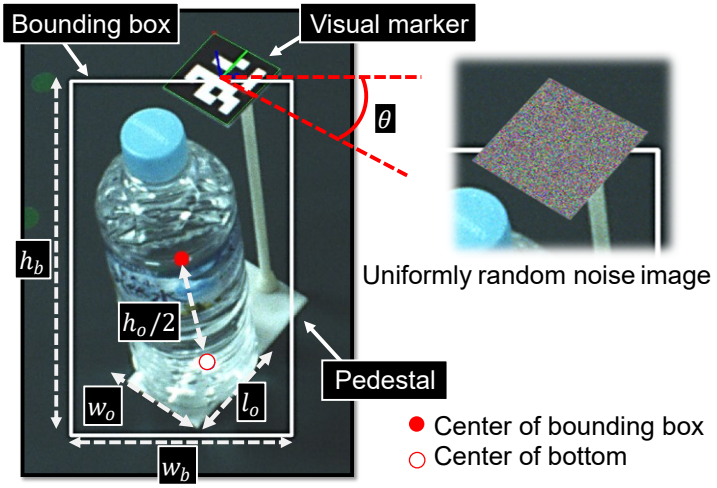


Figure 2.9. Setup of objects to annotate the training data. When fixing the marker and the object to the pedestal, the relative pose between the object and the marker are determined and the marker ID maps to the object label. From the marker ID and pose, we obtain the corresponding object's label and estimate the position and orientation θ of the object in the image.

up the conveyor belt and the camera. Based on the pose of Σ_c with regard to Σ_o denoted by $\mathbf{M}_o^c$, the pose of the object on the conveyor belt $\mathbf{M}_o^b$ can be expressed using the following equation:

$$\mathbf{M}_o^b = (\mathbf{M}_b^c)^{-1}\mathbf{M}_o^c, \quad \mathbf{M}_o^c = \mathbf{M}_{v_k}^c(\mathbf{r}_{v_k}^c, \theta)\mathbf{M}_o^{v_k}. \tag{2.3}$$

$\mathbf{M}_{v_k}^c$ can be obtained by detecting the visual marker based on [Garrido-Jurado et al., 2014], thus, we estimate the 3D pose of Σ_o. The pose can be calculated even when several markers are occluded because only one marker on the pedestal is enough to perform pose estimation of $\mathbf{M}_o^c$ at least. Further, since we can use the higher amount of point correspondences (more marker corners as shown in Figure 2.5) to solve the Perspective-n-Point problem [Fischler and Bolles, 1981; Wang et al., 2018], the obtained pose $\mathbf{M}_o^c$ is usually more accurate than using a single marker (with only four corners).

After estimating the object pose in the image, the bounding box is generated as the circumscribed rectangle surrounding the approximate shape of the object projected onto the image.

To remove the markers, we overwrite all of the pixels of the extracted pedestal region with a background image as shown in the procedure overview of Figure 2.13. The detailed procedure of the background-masking process is as follows.

1. Obtain the pixel position of the object of Σ_o in the image. Based on a pin-

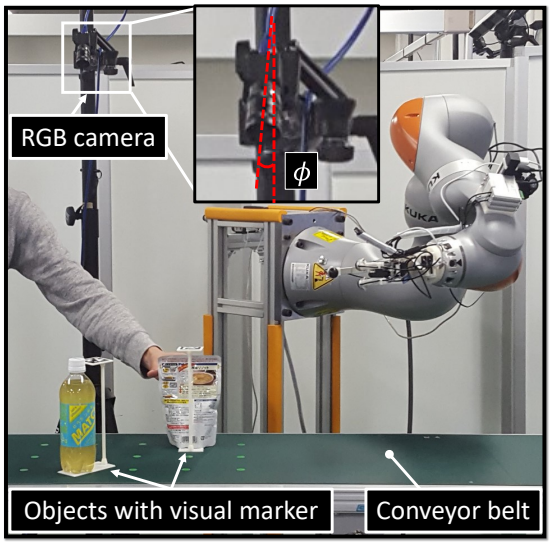


Figure 2.10. One image from collecting the training data. The ϕ shows the installation angle of the camera used to calculate the bounding box size.

hole camera model using the camera parameters $\mathbf{K}$, the 3D object position in Σ_o is projected into the pixel position of Σ_c in the image.

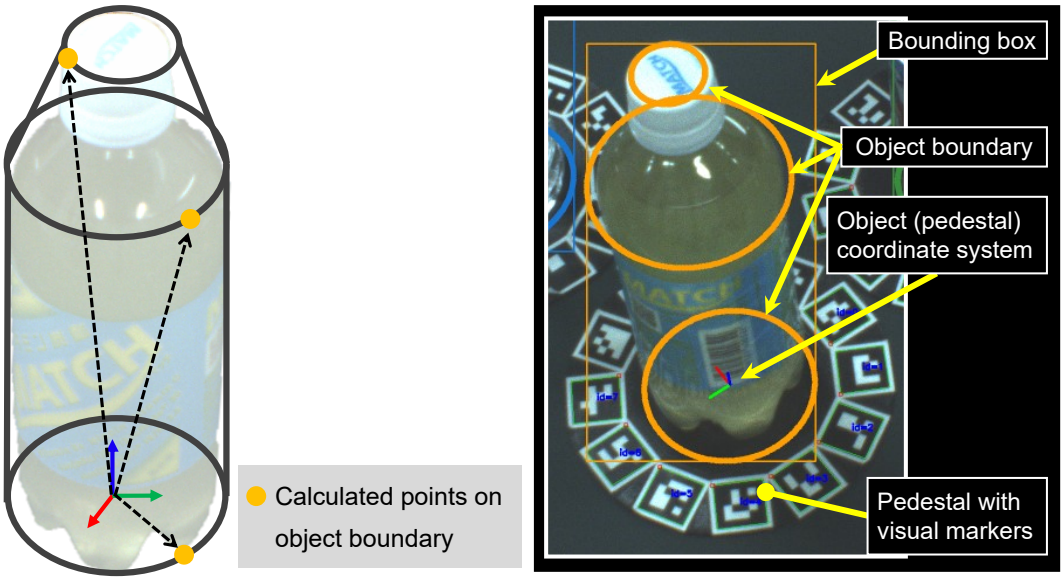
2. Calculate the pixel positions of the points on the pedestal boundary as shown in Figure 2.5. We obtain the 3D boundary positions using the known pedestal radius in Σ_o. Then we project the 3D position to the pixel position in the same process as step 1.

3. Estimate the more pixel positions of the pedestal boundary in the image by elliptic approximation [Fitzgibbon and Fisher, 1995] of these points.

4. Calculate the pixel positions of the points on the object boundary in the image, which is the shape boundary of the approximate shape prepared beforehand.

5. Create the pedestal mask image as shown in the fourth picture from the left of Figure 2.13, where is between the pedestal boundary (calculated in step 2) and the object boundary (calculated in step 3).

6. Fill the pedestal region with the background image using the pedestal mask image.



(a) Approximate shape (b) Generated bounding box

Figure 2.11. Generated bounding box using the approximate shape of a plastic bottle.

2.3.3. Deploying a Vision System

Figure 2.14 shows the processing flow of our vision system including the object detection, orientation estimation, and 2D position estimation. I used the *Single Shot MultiBox Detector* (SSD) [Liu et al., 2016b] for object detection. SSD is a fast single-shot object detector for multiple categories. For our experiment, I used SSD300 which uses 300 × 300 pixel images. SSD returns labels and bounding boxes of all the objects in the image. For each detected bounding box, the system applies the remaining two processes: orientation and 2D position estimation.

The proposed vision system estimates orientation and position separately. The orientation estimator first clips the input image with the bounding box of the detected object, shown in Figure 2.14 (b-1). Next, the clipped image is resized to 128 × 128 with the aspect ratio fixed. Then, given the resized image, the trained classifier estimates the object's orientation. The method estimates the azimuth angles every 45 degrees from 0 to 360 degrees. Thus, I built an 8-class classifier shown in Figure 2.14 (b-2). I used a convolutional neural–network inspired by the VGG network structure [Simonyan and Zisserman, 2015] for the estimation, which is shown in Figure 2.15. The network is constructed with the multiple convolution layers with small (5 × 5 and 3 × 3) convolution filters and fully connected layers are simply connected at the end.

Given the center position, width, and height of the bounding box as inputs, our

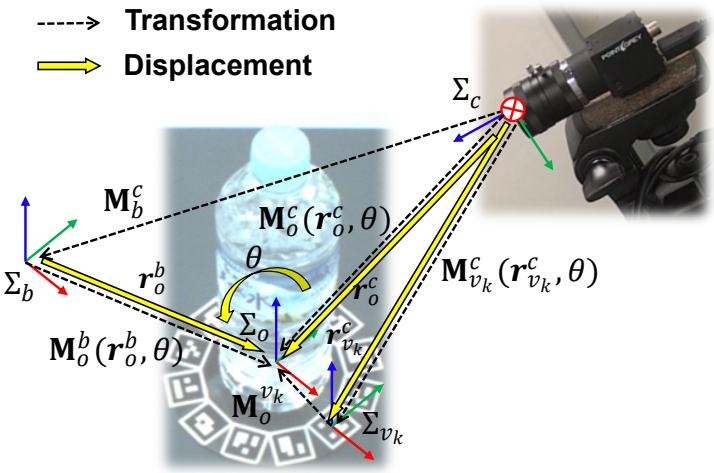


Figure 2.12. Coordinate systems and transformations in formulation for calculating object pose.

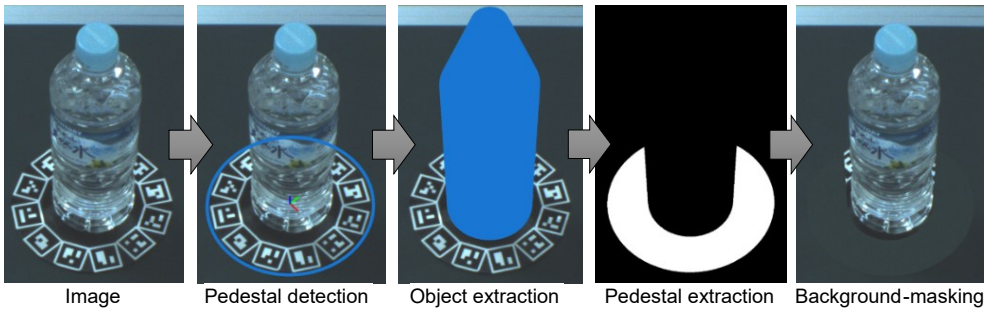


Figure 2.13. Background-masking process for deleting the pedestal from the captured image.

system estimates the 2D object position on the conveyor belt where the object is placed. I define the formula for the position estimation model by multivariate regression as below:

$$P_x = \sum_{i=1}^{4} a_i X_i, \quad P_y = \sum_{i=1}^{4} b_i X_i, \tag{2.4}$$

where P_x and P_y represent the positions in mm of the running direction and the width direction of the conveyor belt. a_i and b_i represent the regression coefficients of each position. Terms X_1 and X_2 represent the center of gravity position in x and y, while X_3 and X_4 represent the height and width of the estimated bounding box. All units of the inputs are in pixels.

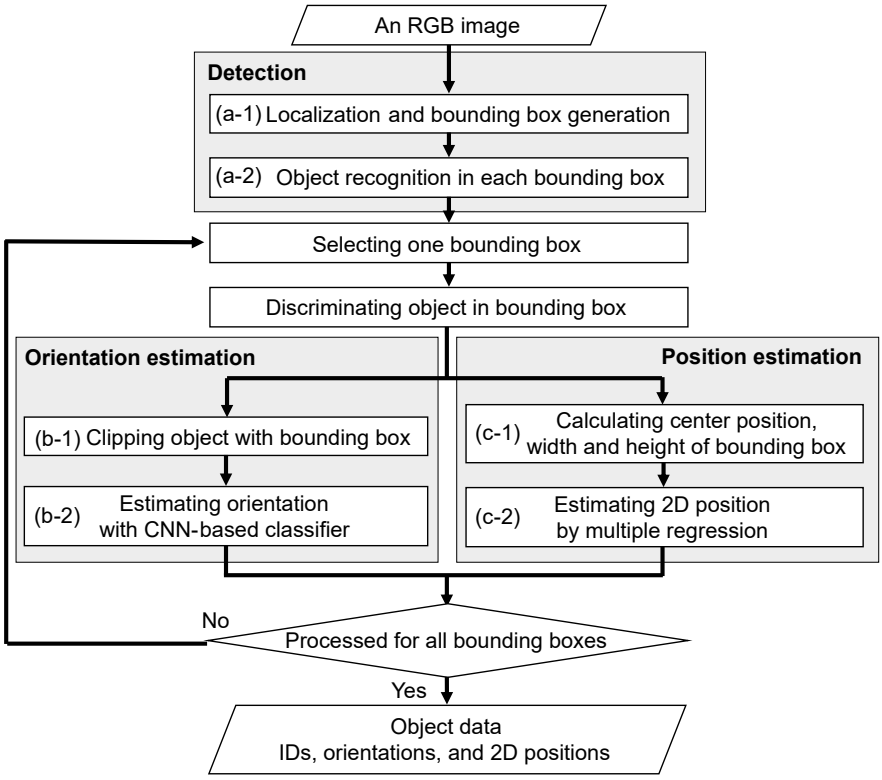


Figure 2.14. Processing flow of assumed robot vision system.

2.3.4. Experiments

Setup

In this experiment, two persons collected 1600 images including six objects shown in Figure 2.16. One person photographed the images while the other person manually changed the poses of the target objects. In the proposed method, the one person arranged objects in various poses on the conveyor belt while confirming the collection progress display in Section 2.3.1, but the single marker method does not perform the collection progress display. I used 500 images to train the model for each method, and 100 images to test the performance of the vision system. I used an RGB camera (Point Grey Research, Flea3 FL3-U3-88S2C) to capture the images. The camera was fixed above a conveyor belt (Okura Yusoki, BELCON MINI III) to reproduce the environment of a factory line. In manual annotation, one person created bounding boxes and assigned the object labels using a graphical annotation tool*.

*LabelImg (Available: `https://github.com/tzutalin/labelImg` [Accessed: 25- Nov- 2020])

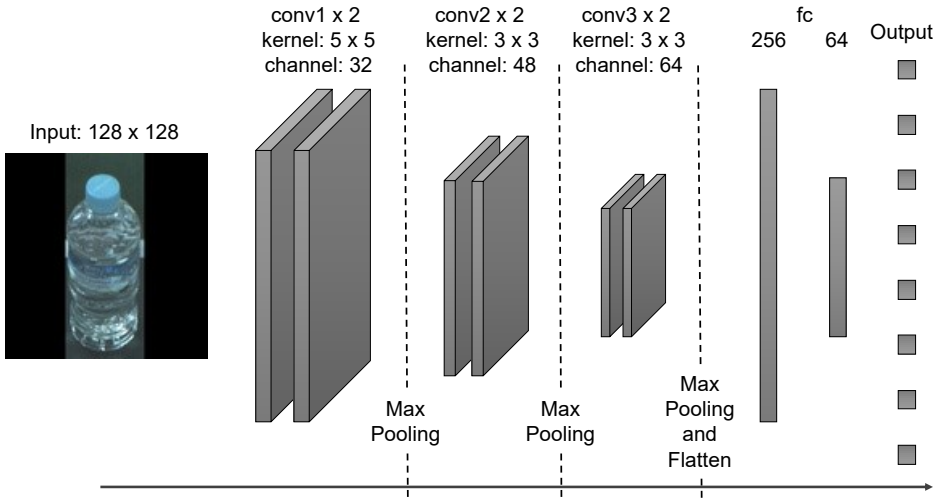


Figure 2.15. Proposed network for orientation estimation.

Figure 2.16. Appearance of six objects tested in our experiments.

To evaluate the proposed method in terms of the collection time and the performance of the vision system, I compare it with the method using manual annotation and the single marker method. The single marker method is an automatic annotation method that uses only one marker on the pedestal for one object based on the method described in Section 2.3.2. I used one square marker with 40 mm length on each side for the single marker and 12 square markers 31 mm length on each side for the multiple markers. For the multiple markers method, I used a 3d-printed circular pedestal of the radius 13.5 mm and 1 mm thick (Figure 2.5).

Time to generate training datasets

Table 2.2 shows the times needed to generate the training dataset of 500 images. The table shows the times for capturing, annotation, and the totals for the three methods. The results, with 436 minutes for automatic annotation and 966 minutes for manual annotation, prove that the proposed automatic annotation system

Table 2.1. Specifications of six objects tested in our experiments.

Name	Label	Feature
Plastic Bottle A	PB-A	Tall object
Plastic Bottle B	PB-B	Tall object
Pouch A	P-A	Deformable object
Pouch B	P-B	Deformable object
Can A	C-A	Small object
Can B	C-B	Small object

Table 2.2. Time to generate a training dataset [min].

	Single marker	Multiple markers	Manual
Capturing[a]	75.0	436	80.0
Annotation[b]	37.2		886
Total	112 (1.87 hours)	436 (7.30 hours)	966 (16.1 hours)

[a]The time for capturing 500 images.
[b]The time for annotating the images.

reduced the time to less than 50% of the time needed for manual annotation. In the case of the single marker method, the person does not confirm the collection progress display while arranging the objects on the conveyor. Therefore, the time by the single marker method is the shortest among the three methods.

Figure 2.17 shows the final state of the dataset collection progress. As one qualitative evaluation, the figure shows that the dataset of object poses are collected without bias. Although it is not the completely same histogram shape on each object pose, the result shows that the dataset was collected extensively in all the 24 positions and the 8 patterns of orientation. Figure 2.18, Figure 2.19, and Figure 2.20 show the examples of the collected images with bounding boxes by each method. The manual method does not edit the captured images, but the single marker method does noise-masking to marker regions for generating training image deleted with the markers as already described in Section 2.3.2.

Performance of Vision System

Figure 2.21 shows the results of object detection after training with each dataset. The values displayed next to the object labels are the confidence scores for the recognition. Table 2.3 shows the detection accuracy of each method. The detection accuracy with the proposed automatic annotation is as good or better than

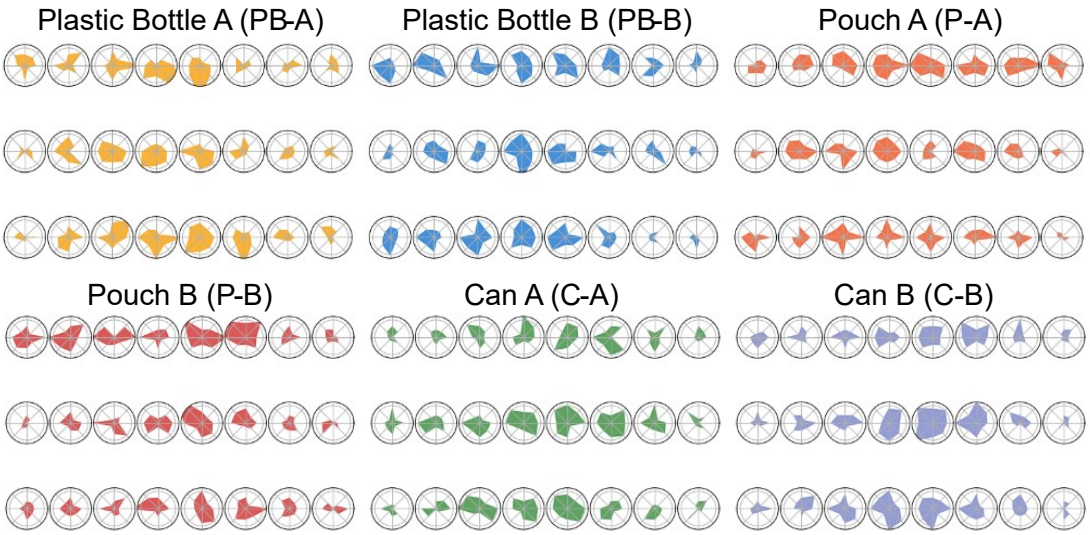
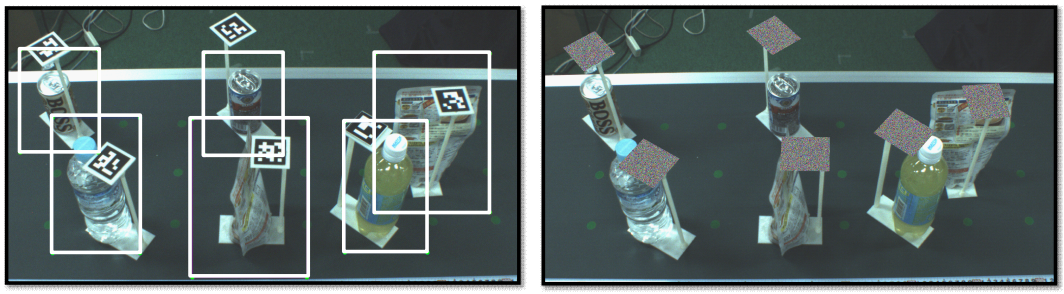


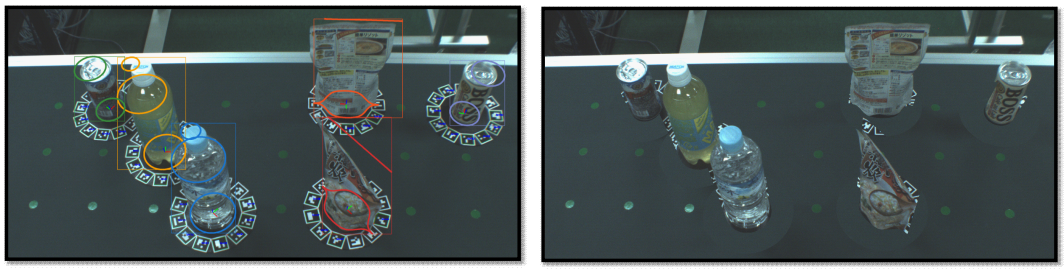
Figure 2.17. Final states of the collection progress.



(a) Automatically annotated image (b) Noise-masking image

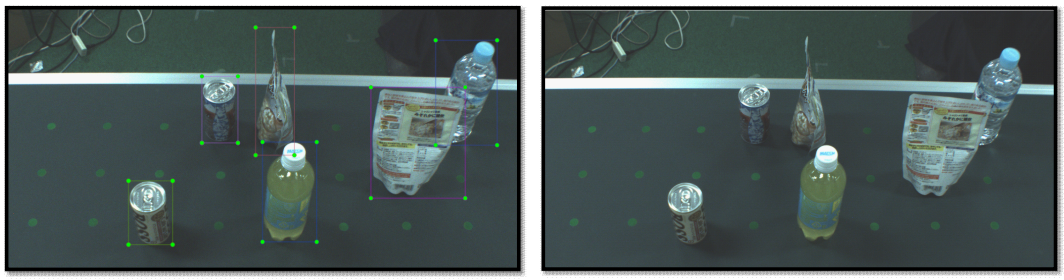
Figure 2.18. Image automatically annotated using the single visual marker.

the detection accuracy with the others. Specifically, the average accuracy of F-measure is 96% for the proposed automated annotation, while the accuracy using manual annotation is 85%, the accuracy using a single marker is 87%, and the accuracy using non-masking multiple markers is 92%. Figure 2.21 shows three sample images in each method. Figure 2.22 shows the result of the orientation estimation which are accuracy rates in the four methods. As described above, the orientation is estimated as the eight classes of the angles. If the estimator answers the collect angle label, I assume that estimation is successful. I calculate the accuracy rates using 100 test images. The figure includes the accuracy rates of each of the six objects. Although the result of the single marker method is less than 30%, the proposed background-masked multiple markers method has an accuracy rate exceeding 80% in all objects and its estimation accuracy is equivalent



(a) Automatically annotated image (b) Background-masking image

Figure 2.19. Image automatically annotated using the multiple visual markers.



(a) Manually annotated image (b) Image used as training data

Figure 2.20. Image manually annotated.

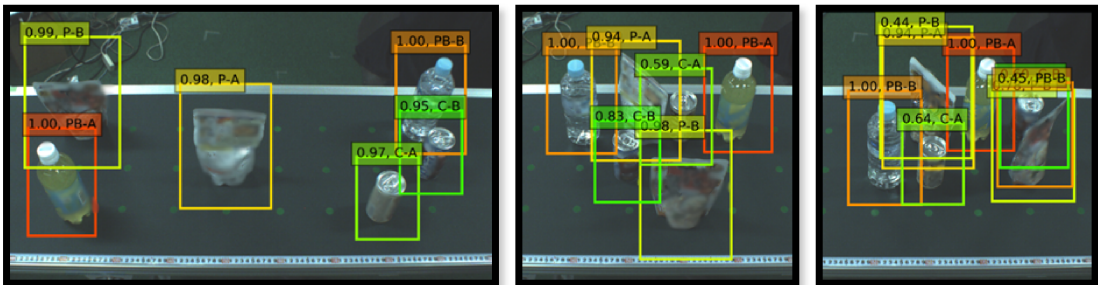
to the manual method.

Figure 2.23 shows the result of the 2D position estimation which are estimation errors in the three methods. With manual annotation, the error remains within about 30 mm, while the error with the proposed automatic annotation method remains within about 40 mm. The error ranges of the manual method and the proposed method are almost the same although the single marker method has an error of more than 50 mm. Since there is only a difference of about 10 mm at the most compared with the manual method, the position estimation accuracy is equal to the manual method.

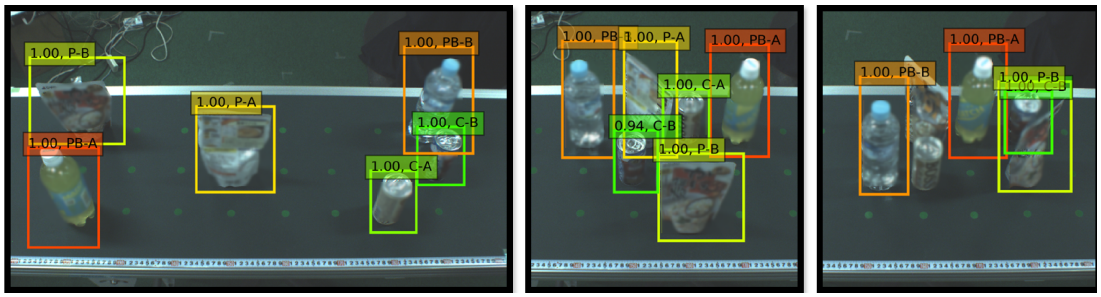
2.3.5. Discussion

Detection accuracy

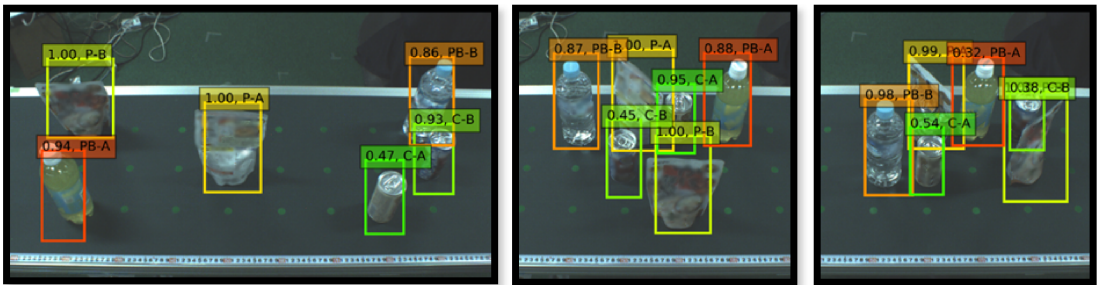
One reason why the detection accuracy of the proposed method is higher than the manual method is because the bounding box size created by the proposed method is tightly assigned with respect to the object pose even if the annotation target object is occluded by another object. Manual annotation can be done to create bounding boxes tracing the object outline but it is difficult to generate



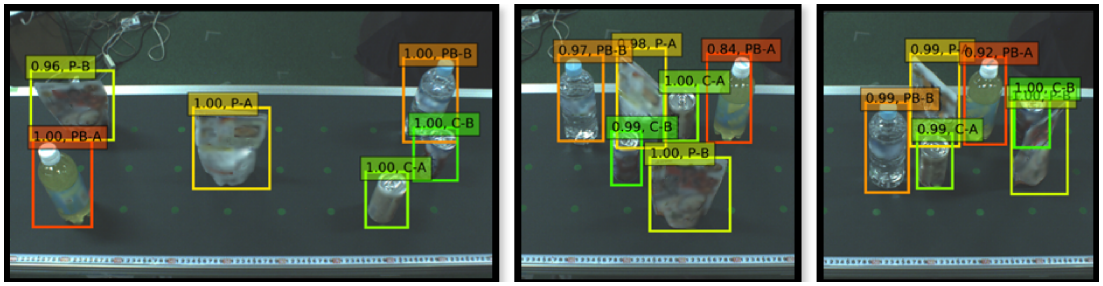
(a) Single marker



(b) Multiple markers (Non-masking)



(c) Multiple markers (Masked)



(d) Manual

Figure 2.21. Object detection results in the automatic annotation methods.

Table 2.3. Object detection accuracy [%].

Label	Single marker			Multiple markers								
				Non-masking[a]			Proposed[b]			Manual		
	F.[c]	P.[d]	R.[e]	F.	P.	R.	F.	P.	R.	F.	P.	R.
PB-A	100	100	100	99	97	100	98	99	97	99	98	100
PB-B	98	99	98	97	99	95	94	97	91	99	98	99
P-A	79	83	75	93	99	88	94	99	89	72	88	61
P-B	83	84	82	93	97	89	93	97	90	80	71	92
C-A	82	91	75	85	91	80	97	100	95	82	79	85
C-B	81	94	71	87	98	79	98	99	98	77	93	66
Mean	87	92	84	92	97	86	96	98	93	85	88	84

[a]The comparative method where the background-masking is disabled.
[b]The method with background-masking multiple markers.
[c]The initial letter of F-measure.
[d]The initial letter of Precision.
[e]The initial letter of Recall.

the bounding box surrounding the object if it is occluded. Such the possible misannotation is one cause of the failure of the detection.

In the single marker method described in Section 2.3.2, they did not propose a method to set the bounding box tightly around the object. As shown in Figure 2.18 (a), in the single marker method, another object may be included in the bounding box. As a result, especially in situations where the objects are close to each other and overlap, such as the rightmost image of Figure 2.21, the single marker method does not successfully detect the objects. Thus to successfully learn objects even in cluttered scenes, I believe that it is necessary to generate bounding boxes tightly along object boundaries such as our method produces.

Accuracy of annotation

In terms of object pose estimation, the single marker showed lower accuracy than the other two methods. At first, I thought that the error included in the dataset due to the single marker was larger than with the proposed multiple markers method. To test this, I evaluated the accuracy of the annotation in terms of the orientation and the 2D position compared with measurement values from the motion capture system shown in Figure 2.24 (a). The (b) and (c) of Figure 2.24 show the two objects used for each method. I measured 24 positions and 8 orientations of every 45 degrees from 0 to 360 degrees.

Table 2.4 shows the results of the detection rates, estimated orientation errors,

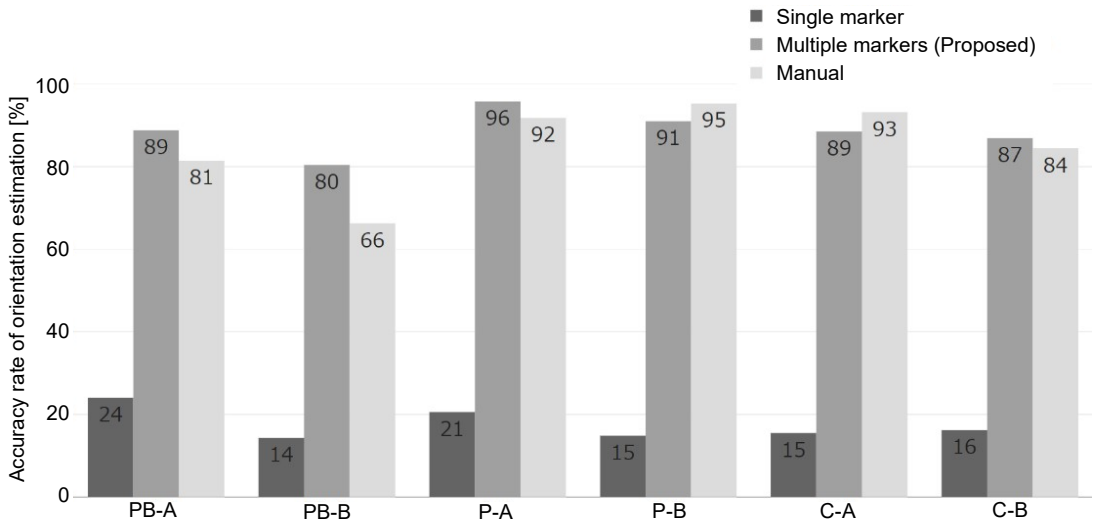


Figure 2.22. Result of object orientation estimation.

Table 2.4. Detection accuracy using the pedestal.

	Single marker		Multiple markers	
	Position	Orientation	Position	Orientation
Detection rate	80%	-	100%	-
Mean of error	32 mm	2.7°	55 mm	2.9°
Variance of error	32 mm	1.9°	56 mm	1.7°

and estimated distance errors for the two methods. Since the 2D position and orientation estimations of the two methods have almost the same accuracies, this indicates both datasets have very little difference in errors. The accuracy of the estimation in the marker positions is similar to that in the proposed method. Improving the position estimation is to use the markers that can estimate the position more accurately, such as [Tanaka et al., 2012]. It is also necessary to develop a robustly graspable robot hand by using force feedback and soft robotics technologies, such as [Homberg et al., 2019].

If we use the single marker, the object features cannot be effectively extracted in the training phase due to the object features hidden by the marker closer to the camera than the object. Since it is difficult to detect a single marker when it is placed on the object bottom, the bottom placement of multiple markers such as ours is more effective.

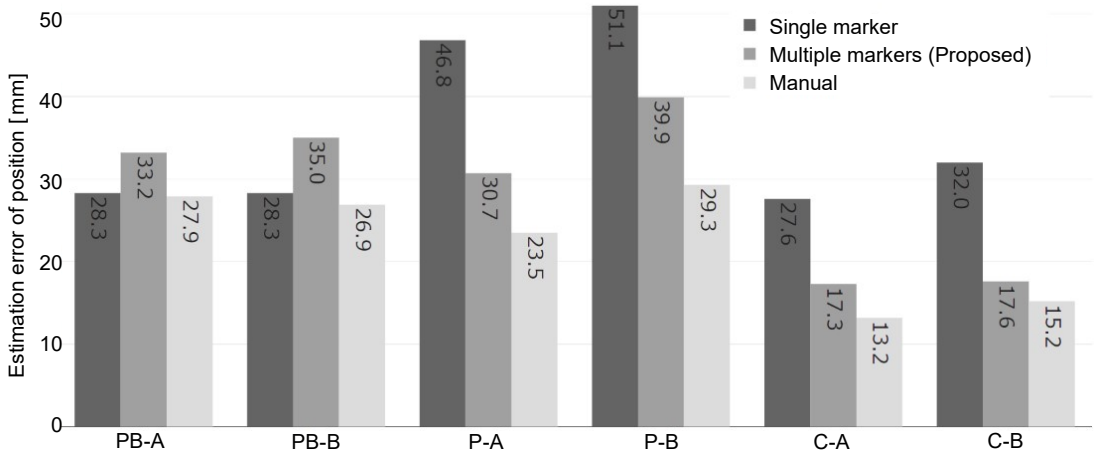


Figure 2.23. Result of 2D object position estimation.

Effect of Masking

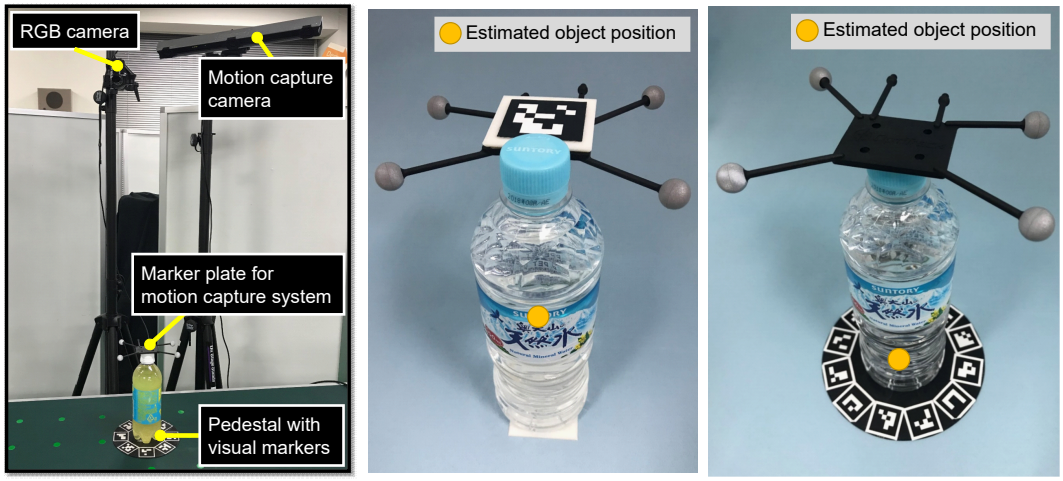
Figure 2.25 shows the attention maps[†] [Lu et al., 2012] that represent image regions where the DCNN pays attention in order to detect the object. Regions with high attention are drawn in red, regions with low attention are drawn in blue. In the case of not deleting the visual markers from the image, the DCNN pays attention to the visual marker as shown inside the red circle in the figures drawing the attention maps of the PB-A, P-A, P-B, and C-A. The corresponding attention maps of the proposed background-masking approach shows that the DCNN focuses on other parts than markers. I believe that deleting the visual marker by background-masking from the images is effective to avoid learning the feature of the visual marker or the deleted regions.

See the image region inside the red square in the images of PB-B and C-B. The figures show that the marker regions are still learned even if the region becomes small. The primitive shape-based object representation sometimes cannot extracts the region of the target object precisely due to its shape approximation errors. More precise region extraction is necessary to completely eliminate the markers, which is our future work.

Unbiased training dataset collection

As shown in Table 2.2, the total time of the proposed method was longer than the single marker method where objects are arranged randomly. However, the

[†]Displayed by keras-vis (Available: `https://github.com/raghakot/keras-vis` [Accessed: 25-Nov- 2020])



(a) Experimental setup (b) Single marker (c) Multiple markers

Figure 2.24. Experimental equipments for evaluating automatic object annotation.

effectiveness of the unbiased dataset for object pose estimation in robot picking is often pointed out. In fact, several famous datasets [Xiang et al., 2014, 2016] are created considering dense annotations of various poses and viewpoints. Also, Sahin *et al.* [Sahin and Kim, 2018] found that the unbiased dataset achieves more accurate estimation on textured-objects even at varying viewpoints. By the combination of all the other proposals, the proposed method gained accuracy compared to the single marker method.

In the current system, the worker decides the arrangement from the collection progress. One possibility to reduce the time is that the system suggests the arrangement considering the progress. I think there is room to reduce the time.

2.4. Fully Automated Collection with Domain Adaptation

This section first describes the proposed robotic training dataset collection system using a small hand–eye robot arm and an automatic rotating stage. Next, I explain the methods for reducing the differences of the illumination and the background. The object appearances differ between dataset-collection and real work environments.

For domain adaptation, I consider how to match the original domain of the generated training dataset to that of the target domain of the real work environment.

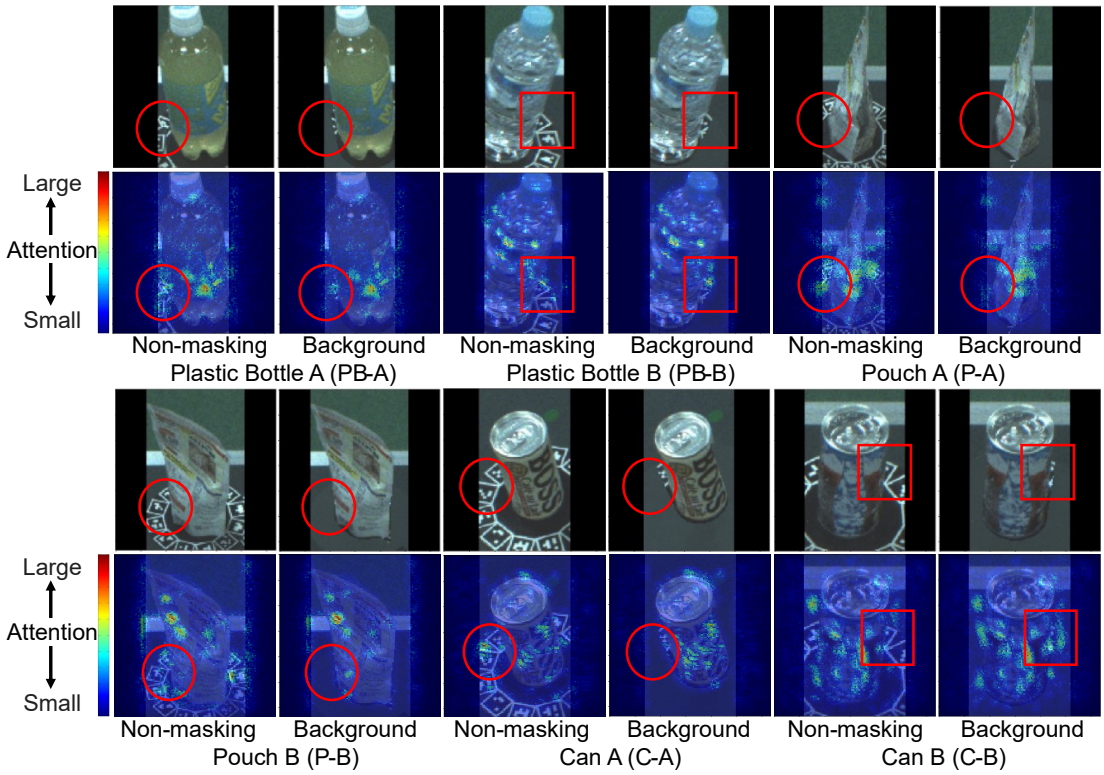


Figure 2.25. Attention maps which show how much attention each region of the image was paid to train the model.

2.4.1. Multi-viewpoint Object Image Acquisition

Figure 2.3 shows our robotic training dataset collection system that includes a small hand–eye robot arm and a controllable rotating stage. Using the small hand–eye robot arm equipped with an RGB camera, we collect images from multiple viewpoints by moving the robot arm to capture a target object placed on the automatic rotating stage.

An RGBD camera is used for both object-image dataset collection and the robot vision system for the industrial work, because I minimize the effects of the camera in the detection experiments. Depth information is not used to generate the training dataset, but the same camera as the real work environment is. The white balance and the exposure of the camera are fixed during image dataset collection and robot experiments. Figure 2.26 shows the proposed dataset collection procedure with its automatic annotation method. Figure 2.27 shows the process for the object region extraction shown in Figure 2.26. To extract the region in consideration of the outline blur caused by anti-aliasing, alpha matting is applied

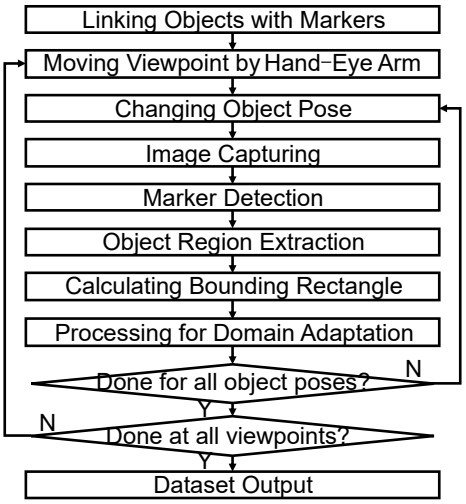


Figure 2.26. Flow of the image dataset collection by the proposed robotic training dataset collection system.

to the captured image. I used *large kernel matting*, a fast method for high quality matting [He et al., 2010]. I used a Python library *PyMatting* [Germer et al., 2020] for alpha matting. Trimap is used for alpha matting and is automatically generated by applying dilation processing to the image, which provides the logical product of the object's approximate shape area after marker detection and that of the foreground area after chroma-key processing.

The generated approximate object mask is according to the estimated object pose related to the camera. If coordinate systems for the hand–eye camera, k-th visual marker, and the object are Σ_c, Σ_{v_k}, and Σ_o, the transformation, $\mathbf{M}_o^c$, from Σ_c to Σ_o shown in Figure 2.3 is calculated as

$$\mathbf{M}_o^c = \mathbf{M}_{v_k}^c(\mathbf{r}_{v_k}^c, \boldsymbol{\theta}_{v_k}^c)\mathbf{M}_o^{v_k}. \tag{2.5}$$

where $\mathbf{M}_{v_k}^c$, $\mathbf{M}_o^c$, and $\mathbf{M}_o^{v_k}$ are transformations from Σ_c to Σ_{v_k}, from Σ_c to Σ_o, and from Σ_{v_k} toΣ_o, respectively. The translation vector, $\mathbf{r}_{v_k}^c$, and the rotation vector, $\theta_{v_k}^c$, are estimated from the detected visual markers.

2.4.2. Object Image Scaling for Consistency of Geometry

Object image scaling is applied to the collected images to reduce the differences in appearance caused by the varying distances between the camera and the object. To accomplish this, the size of the object placed on the automatic rotating stage is adjusted to be fitted to the size of the object placed on the conveyor in the real

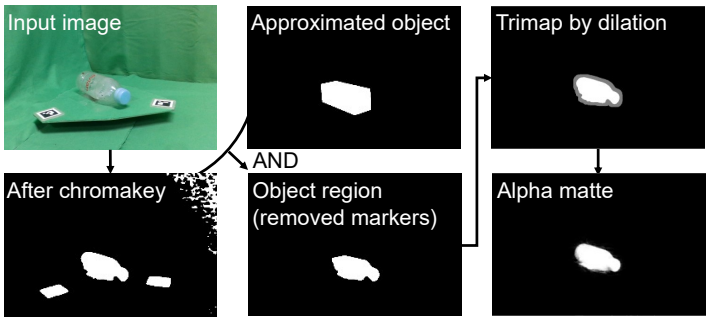


Figure 2.27. Extracted object region (bottom center) by applying AND operation with the image after chromakey (bottom left) and the image showing the approximated object (top center) in the estimated pose based on marker detection, automatically generated trimap (top right), and the generated alpha matte (bottom right) used for alpha matting.

work scene.

As shown in Figure 2.28, the visual markers on the marker board in both images are detected. For geometric consistency of the dataset images, the size of the object region in the image is adjusted according to the scaling parameter, k, estimated as

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} k & 0 \\ 0 & k \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \quad k = \frac{d_t}{d_s}, \tag{2.6}$$

where d_s and d_t are the distances from the camera coordinate system, Σ_c, to the marker board coordinate systems, Σ_s and Σ_t, of the source and target images.

2.4.3. Color Matching and Background Synthesis for Consistency of Illumination

For the color matching proposed in this study, histograms of pixel values in the RGB color space are calculated from an object-area image captured in the real work environment, and HM [Gonzalez and Woods, 2001] is performed. The generated image has a distribution similar to the illumination in the real work environment. Thus, the difference in the illumination is reduced.

The cumulative distribution, $cdf_s(i)$ $(i = 1, 2, .., l)$, of the input image's histogram, $\boldsymbol{h}_s$, is matched to the cumulative distribution, $cdf_t(i)$, of target image's

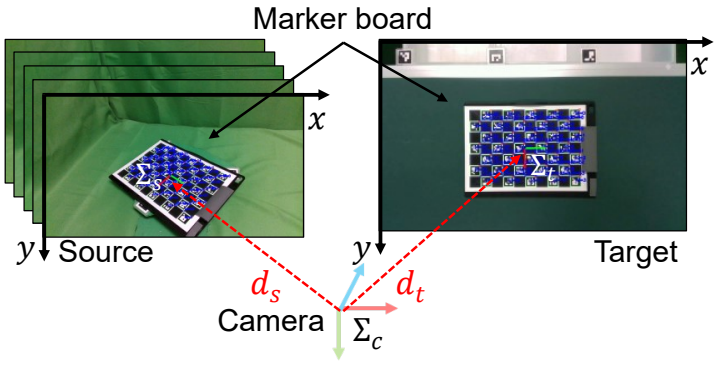


Figure 2.28. Illustration of calculating the scaling parameter, k, representing the distances from the camera to the center of the rotating stage used for dataset collection and one point of the conveyor in the real work scene.

histogram, $\boldsymbol{h}_t$. Each cumulative distribution function (CDF) is calculated as

$$cdf_s(i) = \sum_{j=1}^{i} \frac{h_s(j)}{N_s}, \quad cdf_t(i) = \sum_{j=1}^{i} \frac{h_t(j)}{N_t}, \tag{2.7}$$

where l is the number of bins in the histogram, and N_t and N_s are the number of pixels in each image.

To extract the boundary between the object and the background, using the automatically generated trimap, we apply alpha blending [Szeliski, 2011] to the image at the time of image collection to combine it with the background image captured in the real work environment. Then, we apply HM to the image of only the area within the bounding box of the object.

Images used for applying HM to the image of the plastic bottle are shown in Figure 2.29. The leftmost image shows the source image, the image to the right of the source image is a target image as the destination, the image to the right of the target image shows a result of the HM, and the rightmost image shows the image after EQ. We use *CLAHE* [Zuiderveld, 1994] to smooth jaggy histogram distributions by the EQ. Finally, background-synthesized and histogram-matched images are used to train the object detector.

2.4.4. Evaluating Object Detection Performance

Outline of experiments

First, to evaluate the quickness of the proposed robotic training dataset collection system, I compare the collection time by the proposed dataset generation with

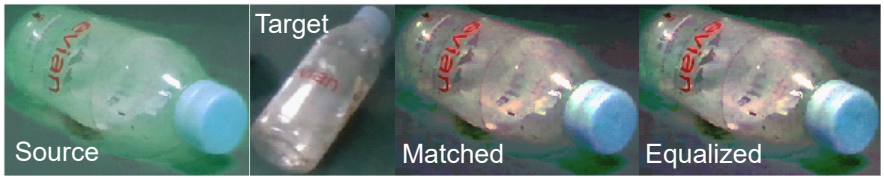


Figure 2.29. HM applied to a plastic bottle image. "Source" and "Target" indicate the input image and the image with the target histogram to match. "Matched" and "Equalized" are the images after application of HM and after application of the EQ of Matched, respectively.

Table 2.5. Average time to collect 100 image datasets for one object. Automatic or manual is shown next to the time measured.

	Type of automatic dataset collection					
	Single marker		Multiple markers		Proposed	
Process needed	Time [s]	A.[a]/M.[b]	Time [s]	A./M.	Time [s]	A./M.
Object placement	125	M.	727	M.	2.05	M.
Image acquisition		M.		M.	10.2	A.
Annotation	61.7	A.	-	A.	-	A.
Total	187	-	727	-	12.3	-

[a]The initial letter of Automatic.
[b]The initial letter of Manual.

the collection time by the manual dataset generation. Furthermore, I show the accuracy of the annotation results.

Second, I show the similarity of the images applied adaptation methods with those captured from the real scene. To evaluate the performance of the object detector trained with the image dataset that applied the proposed adaptation method having the highest similarity, I show the detection results of the target objects by the detector.

I used *ArUco*, an AR library [Garrido-Jurado et al., 2016; Romero-Ramirez et al., 2018] to detect AR markers for registering the object pose of each object image collected using the proposed robotic training dataset collection system. This object poses were used to generate an approximate object mask. The target objects contained 33 different aluminum cans, 33 glass bottles, and 33 plastic bottles, as shown in Figure 2.30. The target objects were sampled from the waste samples in a recycling factory for industrial waste items.

(a) Aluminum can

(b) Glass bottle

(c) Plastic bottle

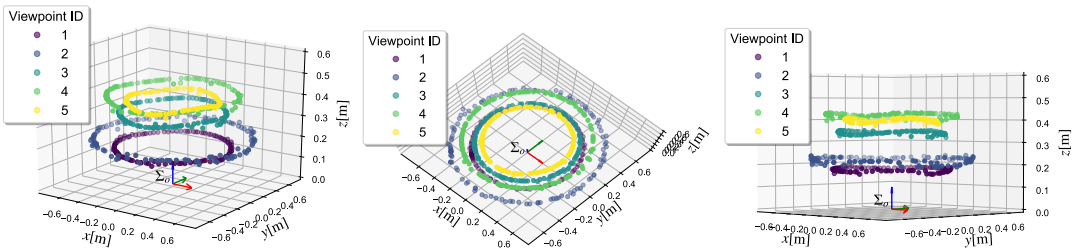
Figure 2.30. The waste samples of (a) aluminum cans; (b) glass bottles; and (c) plastic bottles used in the experiments.

Image dataset collection time

To demonstrate the effectivity of the automatic dataset collection in this framework comparing to the framework described in Section 2.3, this section describes the results of the comparison of times needed to collect image datasets.

Table 2.5 shows for each automatic dataset collection method the average time needed to collect 100 images for one target object and the method (automatic or manual) for three processes: object replacement, image acquisition, and annotation. The proposed dataset collection was completed in 12.3 s on average for 100 images of a single object. The results indicate that the time required for collecting the training set was incredibly shortened compared with the other methods. The viewpoints taken by the proposed robotic training dataset collection system are widely scattered as shown in Figure 2.31, suggesting that a dataset having large variations can be collected in a short period.

The total time required to collect the training set comprising 59,400 (120 object-



(a) Bird's-eye view (b) Top view (c) Side view

Figure 2.31. Variations of viewpoints taken by the proposed robotic training dataset collection system. Σ_o shows the object coordinate system shown in Figure 2.3. Viewpoint IDs from 1 to 5 represent the five viewpoint patterns adjusted by changing the joint pose of the small robot arm.

orientation patterns × five viewpoint patterns × 99 objects) images captured with a green screen was about 111 min. Such a short collection time enables us to easily increase the number of training sets when the target object increases or changes.

Quantitative evaluation of annotations

To evaluate the annotation results, the automatically object-extracted image is compared with the manually annotated image, as shown in Figure 2.32. Using a manual annotation tool[‡] and by clicking several points on the object contour in images, the images are annotated by humans for evaluation.

Based on true-positive (TP), false-positive (FP), and false-negative (FN) results, as shown in Figure 2.32, I calculated the intersection over union (IoU), precision, recall, and F-score [Wang et al., 2020b] as

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}, \tag{2.8}$$

$$\text{F-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{2.9}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{2.10}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{2.11}$$

Table 2.6 shows the results of the object region extraction in the training set.

In all trials and categories, the mean values of precision rated around 70%. The mean values of recall were rated higher than 95% and with smaller standard

[‡]labelme (Available: `https://github.com/wkentaro/labelme` [Accessed: 25- Nov- 2020])

Table 2.6. Results of our object region extraction in our automatic dataset generation. Each element shows mean±standard deviation of IoU [%], Precision [%], Recall [%], and F-score [%]. The mean values are calculated from randomly selected 33 images of each object category in the three categories.

	Metric [%]			
Object	IoU	Precision	Recall	F-score
Aluminum can	71±16	71±17	98±1.7	81±11
Glass bottle	67±17	69±18	96±4.4	79±13
Plastic bottle	77±14	78±15	97±2.2	86±9.8

deviations than those of precision. These results suggest that there were some false predictions. However, there were few missed pixels in the ground truth. As a result, the calculation provides a low IoU with a mean of F-score of 80%.

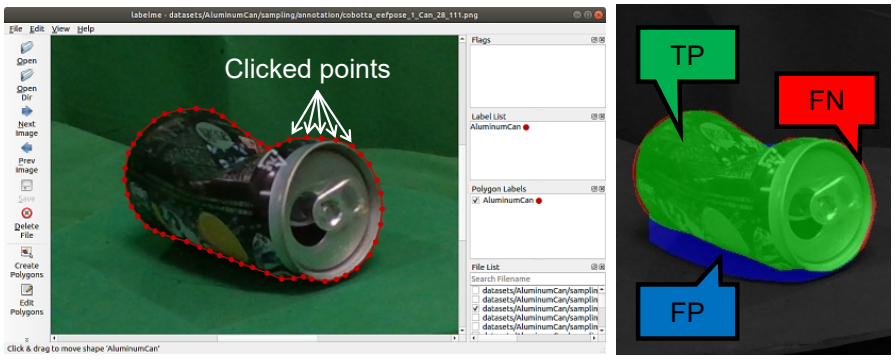
Effect of reducing differences from real work scene

In this section, I discuss the effect of the proposed method of reducing the difference from the real work environment. To evaluate the performance of the proposed color adjustment, I compare it with two other methods.

The first unifies color reproducibility by applying *color correction* (CC) using *ColorChecker Passport Photo* (X-Rite, Inc.), which has a panel of 24 industry-standard color-reference chips. The CC in this study is based on a color-transfer method that can adjust the colors in an image to match a target-image color profile [Berry et al., 2018]. The goal is to create a transform so that, when it is applied to the values of every pixel in a source image (the left of Figure 2.35), it returns values mapped to a target image (the right of Figure 2.35) profile [Gong et al., 2016].

The other is an easy-to-use image-rendering SC method [Pérez et al., 2003] used in the fields of computer graphics [Kakuta et al., 2007] and computer vision [Mukaigawa et al., 2001; Okura et al., 2015; Sato et al., 2005]. SC was used to create a photomontage by pasting an image region onto a new background using Poisson image editing [Pérez et al., 2003]. Figure 2.33 shows the results of CC, SC, and HM. The parameters needed in the methods described in this section are organized in the Table 2.7.

Figure 2.34 shows histograms in the RGB color space of the images in Figure 2.29. The histogram distributions in the RGB color space of the target image (Target) and the converted image (Matched) are visually similar after applying HM.



(a) Clicked points in annotation tool (b) Evaluation

Figure 2.32. Visualization of manual annotations needed to generate ground truth to evaluate the proposed automatic object region extraction. Left image shows the window of the annotation tool (labelme) and several annotated image points. Right image shows the parameterization of the evaluation results of the automatic object region extraction.

Table 2.7. Necessary images for adaptation methods.

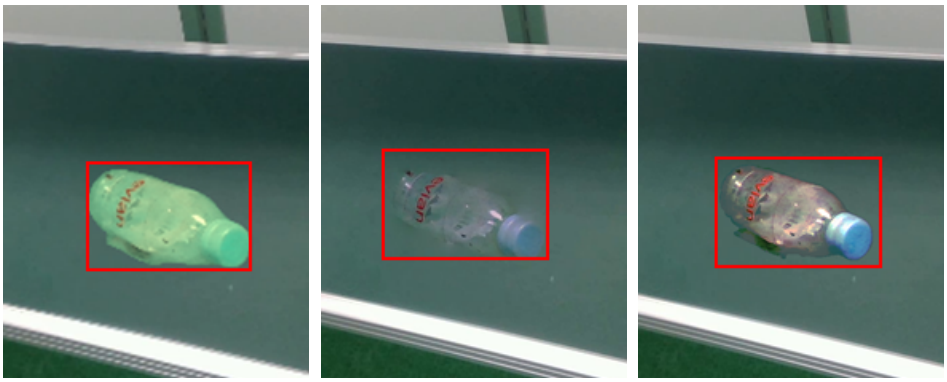
Method	Necessary images
Image scaling	One image pair including a calibration board
Background synthesis (BS)	One background image
Color correction (CC)	One image pair including a color checker
Seamless cloning (SC)	One background image
Histogram matching (HM)	One object image captured in a real scene

To conduct a quantitative evaluation, the distance between two histogram distributions were evaluated using *earth–mover's distance* (EMD) [Rubner et al., 2000] and *Bhattacharyya distance* (BD) [Bhattacharyya, 1943].

EMD is a distance measure based on the solution of the transport problem, which is a linear-programming problem. The minimum cost (L2 norm), d_{ij}, of the transportation from one distribution, $\boldsymbol{P}$, to the other distribution, $\boldsymbol{Q}$, and the minimum cost of the amount of cargo (flow), $F = f_{ij}$, transported from $\boldsymbol{P}$ to $\boldsymbol{Q}$ are calculated as

$$\mathrm{EMD}(\boldsymbol{P}, \boldsymbol{Q}, F) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}}. \tag{2.12}$$

If we consider two normalized histograms, $\boldsymbol{H}_s$ and $\boldsymbol{H}_t$, the BD between $\boldsymbol{H}_s$



(a) Color correction (b) Seamless cloning (c) Histogram matching

Figure 2.33. Comparison of appearances of synthesized images: (a) synthesized images with CC applied; (b) SC applied; and (c) HM applied.

and $\boldsymbol{H}_t$ is given by

$$\mathrm{BD}(\boldsymbol{H}_s, \boldsymbol{H}_t) = \sqrt{1 - \sum_{j=1}^{i} \sqrt{H_s(j) \times H_t(j)}}. \tag{2.13}$$

To evaluate the image similarity with the object image captured in the real scene, I calculated the histogram distributions of the four types, which include the original, BS, BS+CC, SC, and BS+HM.

The effects of the proposed method, BS+HM+EQ, were compared to those of BS+HM, HM, and BS, which are derivatives of the proposed method. I also compared the comparative methods BS+CC and SC as other color adjustment methods.

The calculated values of the EMD and BD in the RGB color space are shown in Table 2.8 and Table 2.9. To compare the images to the object images captured in the real scene, I used those cropped by the bounding boxes as shown in Figure 2.33 in red boxes.

The result of the CC shows that the EMD and BD are larger compared with the result of HM. In the case of the CC, the homography transformation matrix in the RGB color space must be calculated using source and target images, including the color checker shown in Figure 2.35. On the other hand, because the source shown in Figure 2.29 is converted to become similar to the target shown in Figure 2.29, for HM, a higher similarity was achieved.

The calculated values of the EMD and BD suggests that the similarity of the image was largely improved by applying HM, including the area translucent to the back of the object or the plastic bottle's cap. This is because the appearance was

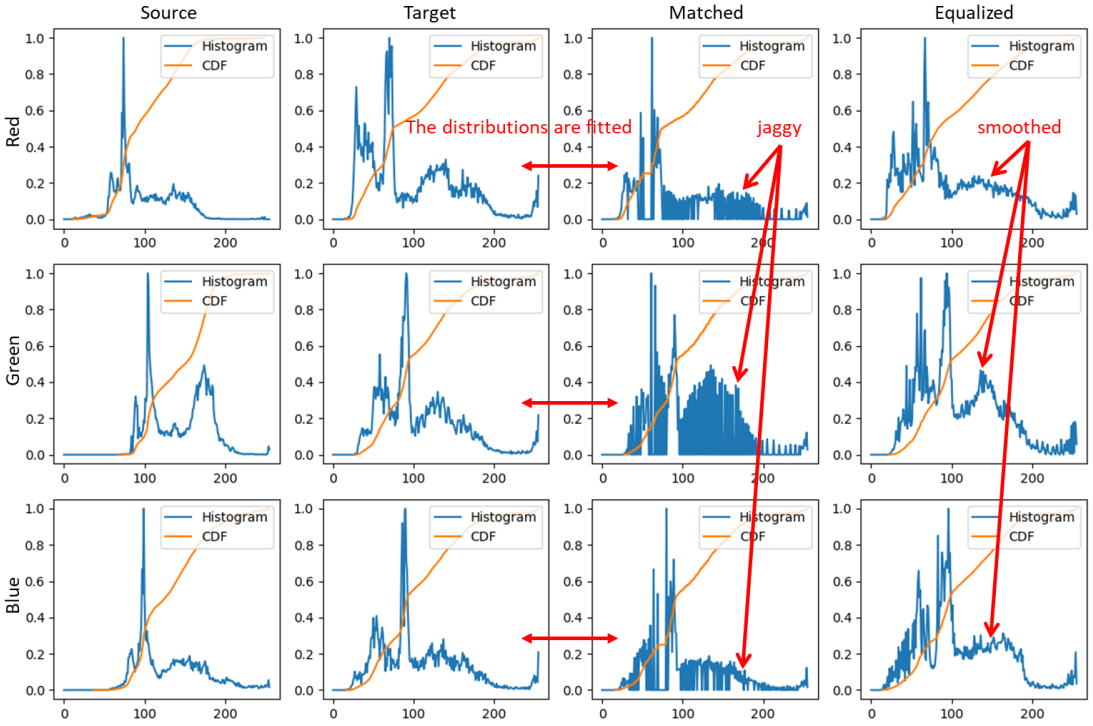


Figure 2.34. RGB histograms and CDFs of the image applied with HM. The graphs are histograms of the RGB color space of the four images on Figure 2.29. The title names atop each correspond to the names displayed in each image shown in Figure 2.29.

improved to approximate the target image. It also suggests that the BS+HM+EQ provided the highest similarity.

Detection accuracy

Table 2.10 shows mean values of detection accuracy for the three target-object categories. As an accuracy metric, I calculated the mean F-score when the IoU threshold was set to 0.5. I also calculated the F-score using detection results with a confidence value higher than 0.5. Using a training dataset automatically generated by the proposed method, detection was performed using a object detector with a trained model of the *single shot multibox detector* (SSD) [Liu et al., 2016b]. SSD is a general object detector with a convolutional neural–network architecture that learns different anchor boxes. Figure 2.36 shows the detection results.

The original shows the result of using 59,400 (120 object-orientation patterns × five viewpoint patterns × 99 objects) images captured with a green screen shown

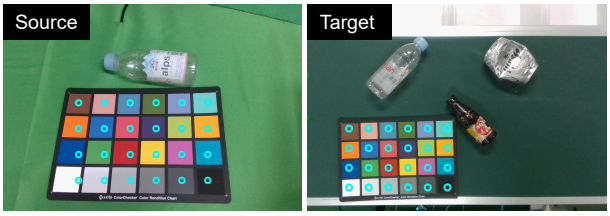
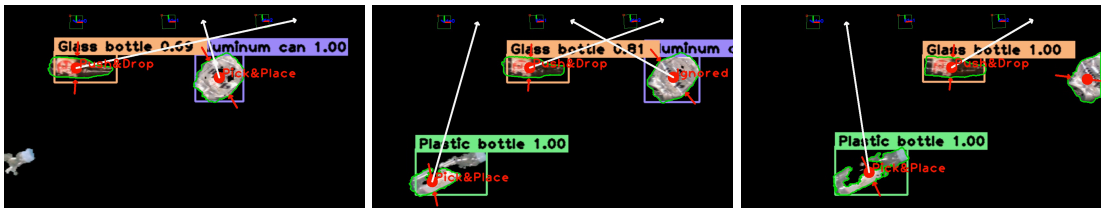


Figure 2.35. Images used for estimating the color homography transformation matrix for CC.

(a) Image sequence recorded in the real work environment



(b) Visualization of detection results of our object detector

Figure 2.36. (a) image captured in the real scene and (b) the image drawn from the detection results of the object detector.

in Figure 2.3. BS, BS+CC, SC, BS+HM, and BS+HM+EQ show image training sets subjected to BS only, BS and CC, SC, BS and HM; and BS+HM with EQ, respectively. Mixed show the training set that I randomly collected images from the three sets of Original, BS, and BS+HM+EQ. All the training sets include 59,400 images.

The last set (Real with 7) is a mixed training set that includes the Mixed and 80 images recorded in the real scene, as shown in Figure 2.37. The conveyor moves at a constant speed in one direction. Thus, if the image acquisition frequencies of the camera are aligned, the object positions in the images can be shifted at a constant interval. Therefore, if we apply manual annotation to only the images of the first frames appearing in the video, we can obtain the image sequence annotated by moving the bounding boxes. I collected the 80 images from two videos in the real work scene in this manner. To improve the quickness of video annotation, in a future work, we plan to use automatic video annotation methods [Kavasidis

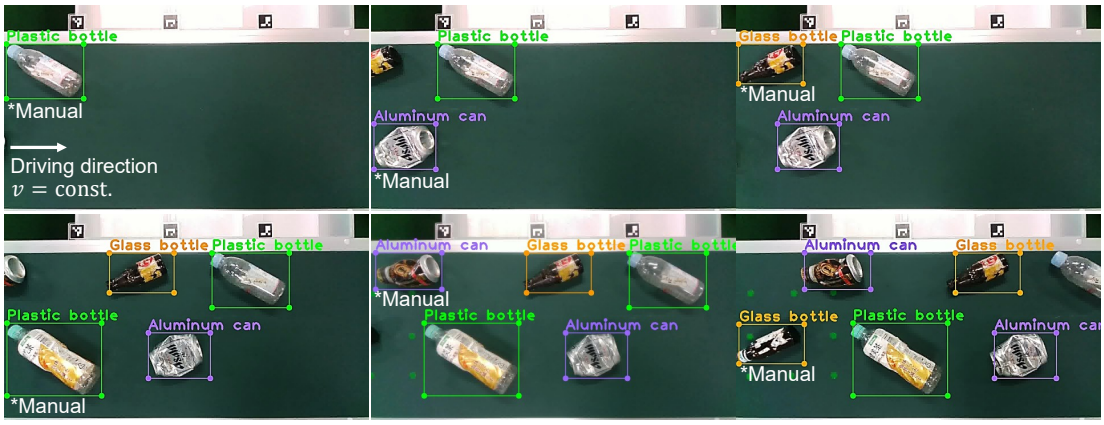


Figure 2.37. Real-world image sequences annotated by humans. *Manual indicates manually annotated bounding boxes. I conducted manual annotation to the video frame in which a new object first appeared. The other images were automatically annotated based on the constant speed of the conveyor and the camera framerate.

et al., 2014; Vondrick and Ramanan, 2011].

The detection results shown in Table 2.10 suggest that Mixed provided the highest accuracy of training without images recorded in the real work environment in the training sets except Real with 7. Therefore, our experimental results demonstrate that the accuracy of the object detector can be improved by applying the aforementioned object scaling, HM with EQ and BS to reduce the differences from the real work environment. Surprisingly, the detector with the BS-only dataset showed the almost same accuracy as did Mixed. The comparison for these detection accuracies should be done in the future using the backgrounds of various real work environments.

By adding the small real-world image dataset including the 80 images, I achieved the highest accuracies of detection, even when the number of items in the dataset was small. The small real-world image dataset not only significantly outperformed the other in terms of accuracy, but the images were also quickly collected. The time needed to capture a video was about 1 min, and the time needed to annotate only six objects in the six images was about 2 min. This was about 3 min total.

2.4.5. Discussion

Ensuring high consistencies of illumination and geometry

The purpose of this study, apart from reducing the time required for dataset collection, was to achieve a highly accurate detector. Within this context, for the

Table 2.8. Calculated values of EMD between the reference image (captured in the real scene) and processed images in the training sets. The histogram comparison was conducted in the RGB color space. The values that indicate the highest similarity are shown in bold.

	Object category		
Training set	Aluminum can	Glass bottle	Plastic bottle
Original	5.86e-1	8.45e-1	1.59e0
BS	7.55e-1	9.41e-1	1.88e0
BS+CC	8.65e-1	6.50e-1	1.77e0
SC	2.62e0	2.10e0	4.82e0
BS+HM	7.36e-3	5.04e-3	5.25e-3
BS+HM+EQ†	**6.27e-3**	**4.67e-3**	**3.96e-3**

† Proposed method in this study.

consistency of illumination, I proposed a method that matches only the luminance distribution information of the image without considering a camera-response function [Takamatsu et al., 2008] and the distribution of the light source [Hara et al., 2005; Imari et al., 2003] in the different environments. In reality, these optical models must be considered when obtaining more realistic images that are similar to real-world ones. However, estimation methods requiring less labor are needed.

In terms of geometric consistency, in this study, only the distance from the camera to the object was considered. However, a 3D model is needed to transform the geometry more precisely. One idea for generating realistic images via a 3D model requires free viewpoint image synthesis based on 3D shape reconstruction methods, such as *Space carving* [Kutulakos and Seitz, 2000], and a geometric registration and an alignment using an RGBD video [Choi et al., 2015].

Precise annotation

Figure 2.38 shows the four cases that had difficulty annotating collected images, especially for cases of difficult object-region extraction. The problematic images shown in Figure 2.38 include an object adhered to foreign substances, a semi-transparent object, a shadow under the object, and a green object.

The foreign substances shown in Figure 2.38(a) needs to be removed from the target object, because the object detector is not designed to recognize this part. Consequently, the robot cannot grasp and push the part. Figure 2.38(b) shows a misannotated semi-transparent object. For the automatic annotation, I could in the future use another method that does not rely exclusively on optical information. As shown in Figure 2.38(c), because it may be difficult to distinguish

Table 2.9. Calculated values of BD between the reference image (captured in the real scene) and processed images in the training sets. The histogram comparison was conducted in the RGB color space. The values that indicate the highest similarity are shown in bold.

	Object category		
Training set	Aluminum can	Glass bottle	Plastic bottle
Original	0.381	0.425	0.400
BS	0.436	0.476	0.445
BS+CC	0.403	0.419	0.428
SC	0.454	0.493	0.493
BS+HM	0.430	0.445	0.467
BS+HM+EQ†	**0.220**	**0.245**	**0.193**

† Proposed method in this study.

a boundary from a shadow, object region extraction may fail. In a future work, it will be necessary to improve the algorithm so that it is robust to shading by referring to illumination estimation methods [Finlayson et al., 2006; Panagopoulos et al., 2011] and DL [Nguyen et al., 2017; Qu et al., 2017]. To avoid difficulty of region extraction caused by similar colors, as shown in Figure 2.38(d), background coloring should be considered.

Application of this framework for robotic assembly

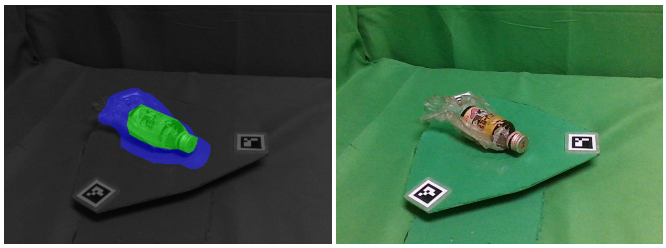
To achieve high-precision assembly operations, understanding 3D scenes is crucial. In this section, I showed the results of annotation with visual markers and the CAD models. We can use annotation methods for training datasets of vision systems conducting the semantic segmentation, object detection, and pose estimation according to required assembly tasks.

To detect more semantic information about the assembly parts from CAD models or images, we can use the part geometries extracted from CAD models to identify the features defined a certain manner but there are several definitions and extraction methods [Das and Swain, 2019; Hamidullah et al., 2006; Lupinetti et al., 2017; Rucco et al., 2019], although the recognition of assembly features has room to discuss, I believe the propsoed automatic annotation methods can be extended for such the assembly domain smoothly.

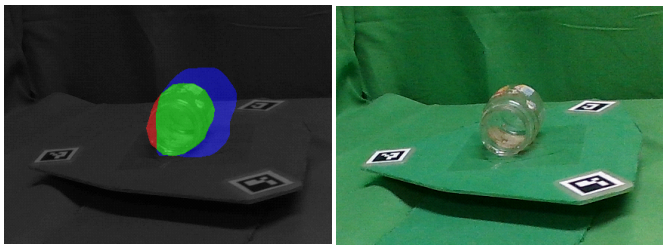
Table 2.10. F-scores of the object detection using DL-based object detector trained using each training set [%]. Mean indicates the mean values of F-score in the three object categories.

	Object category			
Training set	AC[*a]	GB[*b]	PB[*c]	Mean
1. Original	43	76	2.0	40
2. BS	57	45	34	45
3. BS+CC	19	51	23	31
4. SC	14	51	10	25
5. BS+HM	17	59	13	30
6. BS+HM+EQ	22	64	28	38
7. Mixed (1,2,6)	54	53	31	46
8. Real with 7[†]	72	89	75	79

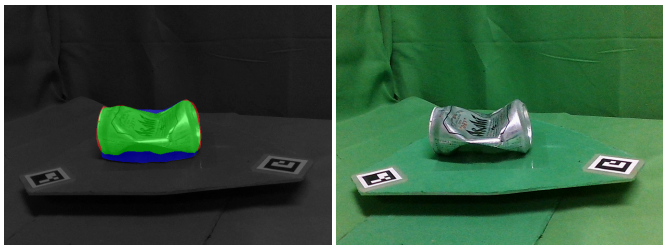
[*a] AC is the abbreviation of aluminum can.
[*b] GB is the abbreviation of glass bottle.
[*c] PB is the abbreviation of plastic bottle.
[†] Proposed method in this study.



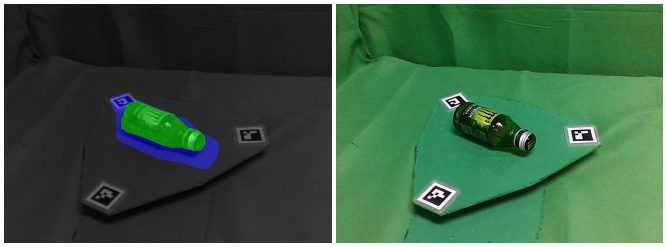
(a) Foreign substances adhered



(b) Semi-transparent object



(c) Shadow misannotated



(d) Green object

Figure 2.38. Problematic images difficult to annotate. The coloring in each left image is the same as that of Figure 2.32.

Chapter 3

3D CAD-Based Assembly Planning

3.1. Introduction

systems that can respond quickly to changes in market demands are needed [Gunasekaran et al., 2019]. For such agile manufacturing [Costa et al., 2017; Gunasekaran, 1999], assembly sequences must be generated rapidly. Several studies for assembly sequence generation (ASG) use 3D computer-aided design (CAD) models [Bahubalendruni and Biswal, 2015; Deepak et al., 2019; Lee et al., 2016].

The combinatorial optimization problem for ASG [Jiménez, 2013] is known to be NP-hard [Goldwasser and Motwani, 1999]. To obtain quasi-optimal solutions in realistic time, heuristic search methods have been used. Some researchers used genetic algorithms (GAs) [Chen and Liu, 2001; Smith and Smith, 2002; Smith et al., 2001] for the ASG in two dimensions. Pan *et al.* [Pan et al., 2006] generated multiple sequences from only a STEP file, a type of 3D CAD file; however, the final sequence had to be determined manually.

I generated preferable sequences for robots, initialized chromosomes of GA based on interferences between many parts (*e.g.,* 32) as described in Section 3.5. I used insertion relations (*e.g.,* plug-receptacle, peg-hole, and pin-slot) and defined *preferable insertion sequence condition* (hereinafter referred to as insertion condition) as the order in which the inserted object are assembled before another object to insert as described in Section 3.4.

However, as shown in Figure 3.1, an insertion sequence generated by the aforementioned method are simultaneously contacted to several parts. Such insertions are difficult to handle.

Assembly planning based on *constraints* defined by such the contact between parts have been discussed [Hirukawa and Iwata, 1991; Hirukawa et al., 1991; Yokokohji et al., 1993; Yoshikawa et al., 1991; Yu et al., 1996]. Robot task plan-

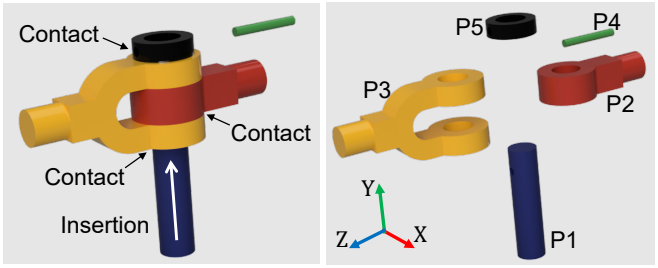


Figure 3.1. Insertion sequence (white arrow) of a part that creates three contact surfaces between parts for a model (#1).

ning based on contact state transitions defined by infinitesimal displacements of target objects have been extensively discussed [Hirai, 1991; Ikeuchi and Suehiro, 1994; Takamatsu, 2003; Takamatsu et al., 2007]. They chose an assembly sequence from several possible transitions of the contract states where the degree of constraint is increased slightly.

The insertion task (Figure 3.1) is difficult because of the difficulty in the contact state transitions. In this study, to reduce such difficulties in the transitions named *constraint state transition difficulty* (CSTD) proposed in [Yoshikawa et al., 1991], I redesigned the fitness function for the GA.

We use two fitness functions: a function to evaluate the insertion condition and a function to reduce the CSTD of the sequences. As the tradeoff between the two objectives, we need to solve a *multiobjective optimization* (MO) problem.

To minimize production time and cost, Choi *et al.* [Choi et al., 2009] tried multicriteria assembly sequencing using a given dataset of 19 parts. They did not discuss the criteria to reduce the difficulty of assembly operations and how to extract the necessary data from models. I performed the MO using a *multiobjective GA* (MOGA) [Coello et al., 2002] to investigate the possibility of finding a Pareto-optimal sequence.

The ASG for deformable parts is another issue that must be solved. All aforementioned methods can only handle rigid parts. I propose a 3D model-based method for obtaining interference-free, insertion, and degree of constraint matrices for deformable parts. Deformable objects with a large volume (*e.g.,* seat, cover, and cloth) are out-of-scope in this study, as each deformable object may require a shape-specific ASG.

Wolter *et al.* [Wolter and Kroll, 1996] proposed an operation method of string-like parts (*e.g.,* wires, cables, hoses and ropes) based on a state representation for part shapes. To recognize and plan a sequence of movement primitives for string-like deformable objects, Takamatsu *et al.* [Takamatsu et al., 2006] proposed a knot-state representation for knot-tying. Dual-armed assembly tasks based on an

elastic energy and a collision cost [Ramirez-Alpizar et al., 2018] and step-by-step assembly strategies demonstrated insertions of ring-shaped deformable objects such as a rubber band [Kim and Sloth, 2020] and a roller chain [Tatemura and Dobashi, 2020]. By deforming the part model, we determine interference-free directions and assembly order of string-like and ring-shaped deformable parts using our method [Tariki et al., 2020].

This study makes four contributions. (i) I designed a fitness function to generate sequences that the CSTD is minimized. (ii) I developed an MOGA that can find Pareto-optimal sequences. (iii) I extended the method for extracting two-part relations for deformable parts. (IV) To show robustness and reproducibility, I extensively evaluated our ASG using eight models having rigid and deformable parts.

3.2. Related Work

Over the past forty years, many studies [Bahubalendruni and Biswal, 2015] have been done on teachless assembly robot systems. These systems derive assembly sequences from CAD models, which are assumed to be available. Several research methods use additional materials, such as instructions about assembly planning [Agrawala et al., 2003], other 3D models with precedence constraints [Li et al., 2020], and digital product descriptions designed between designers and potential manufacturers [Sierla et al., 2018], but, ideally, it should be possible to estimate the assembly sequence using only a CAD model.

I summarize the research on such CAD-based assembly sequence generation considering three points: extracting feasible assembly sequences from CAD models, generating preferable assembly sequences using the relations between assembly parts based on geometry, topology, and attributes, and extracting constraints for planning easier assembly operations.

3.2.1. Exploring Feasible Assembly Sequences with CAD

The 3D geometrical information of assembly products is usually expressed in a 3D model, often using 3D CAD software. Many studies used Computer-Aided Three-dimensional Interactive Applications, which is a high-end software for CAD, CAM, CAE, etc. It has been used for creating *Liaison graphs* [Bahubalendruni and Biswal, 2016], obtaining primary parts joined by connectors [Bahubalendruni et al., 2017], detecting collisions between the assembled components [Pintzos et al., 2016], and detecting part-to-part interference [Agarwal et al., 2018]. These studies did not address selecting a preferable sequence from a set of many sequences. For another example, a previous system [Lupinetti et al., 2016] used

a commercial CAD software API to detect partial contacts between the entities (faces and edges) of the B-rep models of the parts. An API to determine feasible assembly sequences were used in [Michniewicz et al., 2016], although assembly was limited to one direction. Similarly, the approach presented in [Bahubalendruni et al., 2014] accesses the mating relation of a CAD model through commercial software. I used an open-source CAD library so that the barriers to introducing the proposed system are reduced, except for licensing.

To generate a feasible assembly sequence, most CAD-based approaches use a GA [Srinivas and Patnaik, 1994]. A GA was proposed in [Smith et al., 2001] to explore the feasible movements of the parts using a pre-computed part-interference relation named the interference-free matrix. Enhancements to the GA-based approach [Smith et al., 2001] are proposed in [Chen and Liu, 2001; Smith and Smith, 2002] to improve the performance. In these studies, the CAD-based geometric information is extracted properly, but the generated sequences were evaluated in a 2D, rather than 3D, space.

As an example of applying an exploring algorithm to assemblies in 3D space, the discrete artificial bee colony algorithm were used [Özkan Özmen et al., 2018]. The shape of the target products was limited to polyhedral shapes. Extracting the direction of disassembly using polygon meshes of the products were conducted in [Kardos and Váncza, 2018]. The method was only evaluated using a few assembly parts.

3.2.2. Preferable Sequence Based on Part Insertions

Feasible assembly sequences were found using matrices relating to the connections between parts in [Özkan Özmen et al., 2018; Smith and Smith, 2002; Smith et al., 2001]. Many previous studies derive the insertion relations between parts to identify part features [Perzylo et al., 2015], to detect exchanges in parts described in a CAD model [Eltaief et al., 2017], or to define additional geometric constraints on the CAD model as part of a robotic task description [Babic et al., 2008]. However, these papers did not consider how assembly sequences for products with many parts can be planned using such two-part relations. Like ours, there are many studies using artificial intelligence techniques [Deepak et al., 2019]. In these studies, although insertions were done during the assembly process, they did not describe the insertion relation-based assembly sequence generation. In addition, the necessary extraction of insertion relations from a CAD model has not yet been achieved. We can assemble a screw to be inserted by using a tool such as a screw-driver after we fixed a nut, instead of manually assembling the nut after the screw.

Thus, by considering the insertion relations, I consider how to determine the

preferable sequence from among several feasible sequences. To find the feasible and preferable sequences, the initial chromosome for GA is determined based on part interference collisions and a fitness function that considers insertion relations is designed.

3.2.3. Assembly Planning with Constraints

Constraints in automated assembly planning are broadly to include requirements, preferences, and suggestions to the planner [Jones and Wilson, 1996; Jones et al., 1998]. Such widespread constraints have been used in automated assembly planning to analyze and improve the stability of assembled parts [Mosemann et al., 1998], and also used for robotic assembly [Morris and Haynes, 1987]. Several studies are extracting assembly features including topological and geometrical constraints from information included in CAD models of products to be assembled [Neb, 2019]. Hasan *et al.* [Hasan and Wikander, 2017] described an extraction method of geometric constraints. Ou *et al.* [Ou and Xu, 2013] have proposed an automatic collection method of constraint data from CAD models for assembly planning and an algorithm for generating assembly sequences based on the assembly constraints such as Distance, Angle Offset, and Parallel, or mechanical constraints such as Rigid, Revolute, and Slider.

Generated assembly sequences are found based on interference between parts and the output of the system is a set of assembly sequences which are ranked based on the results of the stability analysis. However, whether the constraint functions are properly set or not is relaying on the model designer. Furthermore, most of the previous research articles are limited to the stability analysis using the constraints, which is related to the difficulty of assembly operations but not assumes the constraint transitions during assembly process like ours.

3.3. Overview of CAD-Based ASG

3.3.1. Assumptions

This study assumes the following two things. (1) I use a dual-arm robot, mechanical grippers, and assembly jigs for assembly operations as the current manufacturing industry. (2) The proposed algorithm outputs assembly sequences represented as orders of part IDs (*e.g.,* Part 3, Part 17,...) and the corresponding assembly directions (*e.g.,* −z, −z,...) in 3-axis.

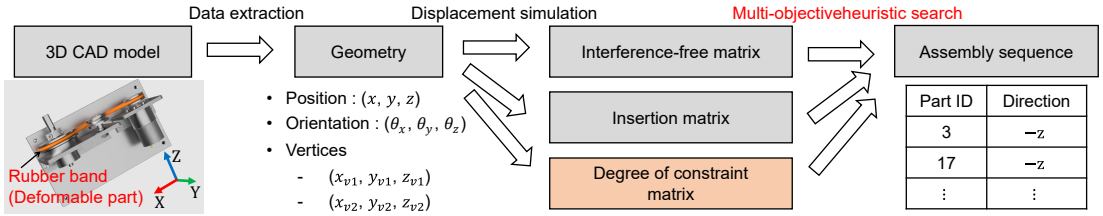


Figure 3.2. Overview of generating an assembly sequence from a 3D CAD model. The input is a 3D CAD model assembled and the output is a linear sequence of part IDs and the assembly direction in world coordinate system shown in the image at bottom left.

3.3.2. Generation Procedure

Figure 3.2 shows the proposed ASG. First, we extract the parts geometries from the CAD model assembled, then we calculate the interference-free, insertion, and proposed degree of constraint matrices. Second, the order and assembly direction of the parts are generated using the proposed MOGA.

3.4. Representing Assembly Parts Relations

In the proposed ASG, we need three matrices shown in Figure 3.2. In terms of the interference-free and insertion matrices of rigid parts, we extract geometric information from 3D models using a CAD software and calculate them by the method.

3.4.1. Interference-free Matrix

This section first describes what the interference-free matrix represents, then an example of the interference relation in the assembled parts, and finally how the interference-free matrix is generated.

The interference-free matrix represents whether part i can be assembled after assembling part j. To calculate the (i, j)-element of the matrix, first, part j is set to the assembled position. Next, part i is moved from the outside to the assembled position along the ± x, y, or z-direction. If part i collides with part j, the element is set to zero (meaning interference), otherwise, it is set to one.

Figure 3.3 shows an interference-free matrix in the six directions for a simple model. From the (3, 2)-element of the matrix in the +x direction, we find that if Part 2 is already assembled, Part 2 interferes with assembling Part 3 along the +x direction. From the (2, 3)-element of the matrix in the +x direction, we

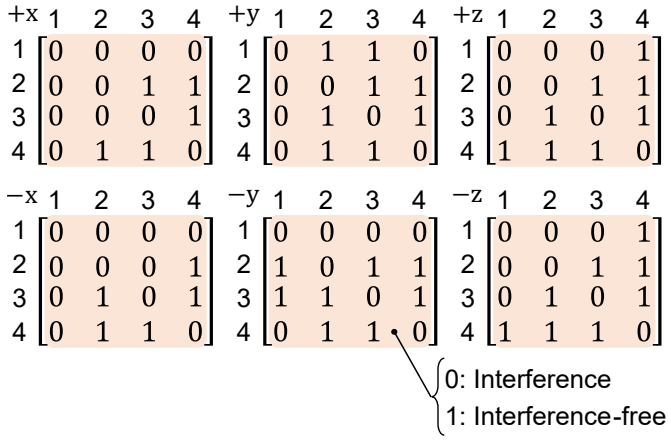


Figure 3.3. Interference-free matrix.

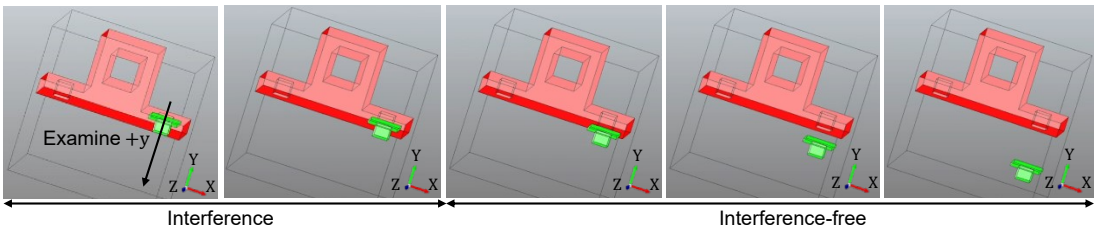
find that even when Part 3 is already assembled, Part 3 does not interfere with assembling Part 2 along the +x direction.

Figure 3.4 shows the process to calculate the interference-free matrix. The calculation requires the estimation of collisions between the two parts, corresponding to calculating the intersection of the two parts. Because PythonOCC includes standard CAD operations, such as *common* between two parts, the intersection is easily calculated, such as whether the volume of the intersection is zero or not. However, in assembly, one part often contacts another part without colliding. Thus, rounding errors in numerical calculation can erroneously indicate a collision.

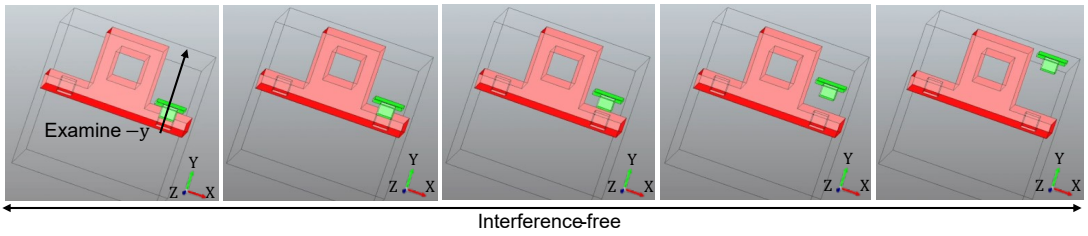
To make the calculation robust against such errors, I choose the threshold adaptively. First, we calculate the volume v_{in} of the intersection after assembling the two parts. Then, we move one part outward in small steps and calculate the volume of the intersection at each step. If the volume is greater than v_{in}, we conclude that the two parts collide. We calculate the intersection using the PythonOCC function *BRepAlgoAPI_Common*, which gives the intersection shape between the two parts, and calculate the intersection volume using the function *BRepGProp_VolumeProperties*, which displays the volume of the input shape.

3.4.2. Insertion Matrix

The insertion matrix represents whether part i is inserted into part j in the (i, j)-element of the matrix. Figure 3.5 shows an example of the insertion matrix for a simple model. From the (2,1)-element, we find that Part 2 is inserted into Part 1; Parts 1 and 2 are female and male, respectively. I define the female parts as



(a) Interference case



(b) Interference-free case

Figure 3.4. Process to confirm interference.

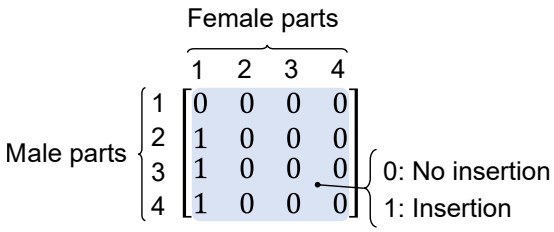


Figure 3.5. Insertion matrix.

the parts to be inserted into by other parts. The male parts are the parts to be inserted into the female parts.

PythonOCC functions can be used to calculate the insertion matrix. First, PythonOCC assigns a shape label (e.g., plane, cylinder, or cone) to each surface. The function *BRepAdaptor_Surface* allows us to obtain the surface types of the parts. The parts with hole-type surfaces are regarded as female parts. Second, we generate a small box at the center of the hole of the female part. The volume of the box is 1 mm^3. If there is a part that intersects with the box, the part is the male (inserting) part and a one is recorded as the corresponding element of the insertion matrix. The intersection between the box and the part is examined using the function *BRepAlgoAPI_Common*. Third, *BRepGProp_VolumeProperties* shows whether the volume of the intersection is greater than zero.

Figure 3.6 shows the process. The red part is the detected female part and the green box is the small box at the center coordinates of the hole of the female

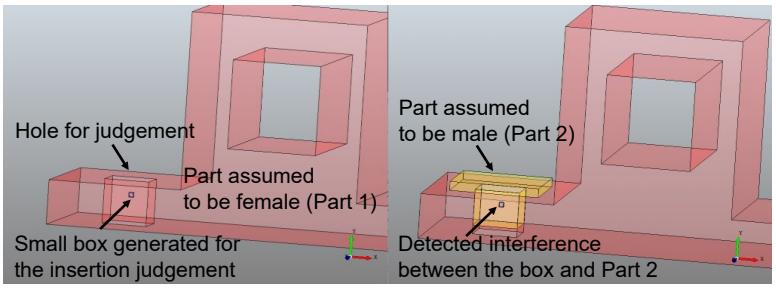


Figure 3.6. Making the insertion matrix.

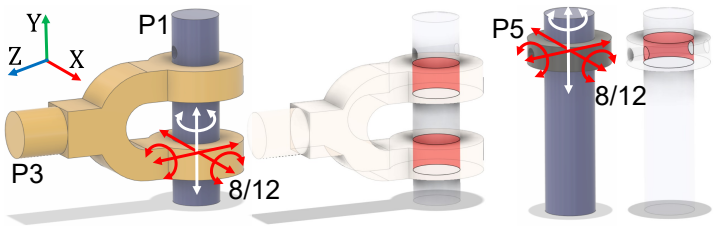


Figure 3.7. Degrees of constraint in two cases. Red-colored shapes are contact surfaces. Red-colored arrows show that P1's interfered directions on coordinate systems determined. The numbers at lower right of the arrows show the degree of constraint per degree of freedom.

part. The yellow part is assumed to be the male part.

3.4.3. Degree of Constraint Between Rigid Parts

We calculate degree of constraint $C(P_i, P_k)$ that indicates contact states between parts P_i and P_k. If there is no contact between the parts, this value is set to 0. According to Yoshikawa *et al.* [Yoshikawa et al., 1991], the degree of constraint is defined as:

$$C(P_i, P_k) = 12 - \sum_{j=1}^{12} F_j(P_i, P_k) \ \in \{0, 1, .., 11\}, \tag{3.1}$$

where $F_j(P_i, P_k)$ $(j = 1, 2, ..., 12)$ indicates constraint-free information for 12 directions of translational and rotational displacements $\pm$x, $\pm$y, $\pm$z and $\pm\Theta_x, \pm\Theta_y, \pm\Theta_z$ shown in Figure 3.7. This value is set as 1 if the parts do not interfere with each other after an infinitesimal displacement. Otherwise the value is 0.

To reduce the time to calculate the function $\boldsymbol{F}$, the interference-free information on the negative directions of all axes are calculated as the transpose of the matrix on the positive direction of each corresponding axis. In other words, for

example, moving P1 in the +x direction and moving P2 in the −x direction are the same in the relationship between P1 and P2, thus, $F_1(P_i, P_k) = F_2(P_k, P_i)$. Other directions have the same relationship. Finally, the matrix of degree of constraint $\boldsymbol{C}$ is then computed using Equation (3.1) as an element. Because $\boldsymbol{C}$ is symmetric, we calculate only the upper triangular component and calculate the other elements based on the relation $C(P_i, P_k) = C(P_k, P_i)$.

Given the assembly order $P_{O_1}, P_{O_2}, .., P_{O_k}$, the maximum CSTD H is calculated as:

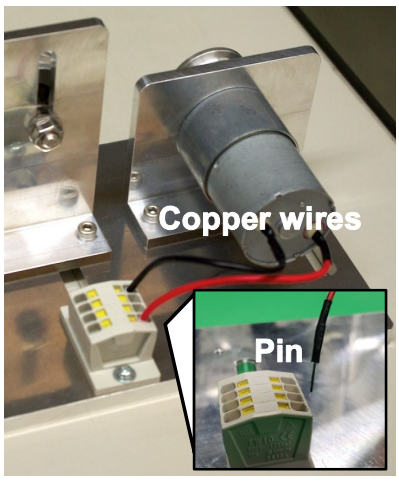
$$H \coloneqq \max_{k \in \{2,3,..,\eta\}} \sum_{i=1}^{k-1} C(P_{O_i}, P_{O_k}), \tag{3.2}$$

where $\sum_{i=1}^{k-1} C(P_{O_i}, P_{O_k})$ shows the CSTD in the assembly of the k-th part P_{O_k} and the other assembled parts $P_{O_1}, P_{O_2}, .., P_{O_{k-1}}$.

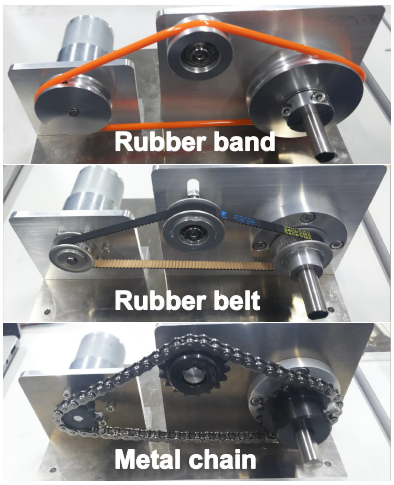
To calculate the CSTD, the constraint-free information of an arbitrary part is determined by investigating whether a part interferes with other parts, as illustrated in Figure 3.7. In the figure, the investigated target part is displaced in six positive and negative directions along the X, Y, and Z axes and rotated around the X, Y, and Z axes. The origin of the coordinate system is automatically determined as the center of gravity of the shape composed of a contact surface (constraint surface) between the two parts. The Z-axial positive direction of the coordinate system is determined as the direction vertically upward in a stable pose of the product with the widest bottom surface to place on a plane. If multiple contact surfaces are found, one of them is randomly selected. The positive directions of the X and Y axes are determined in the directions of the world coordinate system of the CAD model, and only the rotation center is set by the center of gravity. Figure 3.7 shows the determined axes on assembled parts in a model.

3.4.4. Extraction of Two-Part Relations for Deformable Parts

Figure 3.8 shows string-like deformable parts that will be used in the assembly challenge of WRS2020 [WRS2020] and the ring-shaped deformable parts used in the assembly challenge of WRS2018 [WRS2018]. This study concentrates on string-like deformable parts, such as the wire with a rigid pin shown in Figure 3.8(a) and ring-shaped deformable parts such as the rubber band, rubber belt, and metal chain shown in Figure 3.8(b).



(a) String-like [WRS2020]



(b) Ring-shaped [WRS2018]

Figure 3.8. Two types of deformable parts.

String-like parts

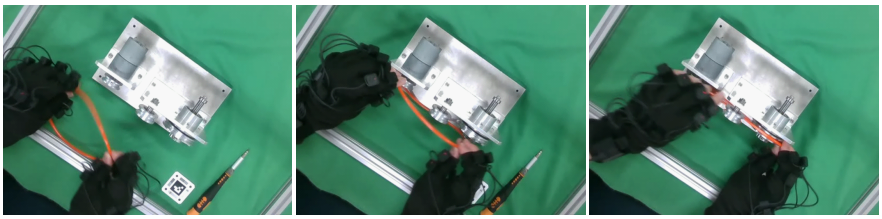
String-like deformable parts, such as a cable with a plug and a wire with pins, are often with a rigid body attached to the tip as shown in Figure 3.8(a). String-like deformable parts, such as connectors, cables and wires, appear frequently in assembly products. Both the plug and pin are attached for inserting into or connecting to others such as a socket and a hole. Thus, if the string-like deformable object has a rigid part connected to others, the two-part relations between the rigid part and others must be investigated.

For example, the vertices of string-like parts and the corresponding inserted part are recognized, then the system calculates the interference-free, insertion, degree of constraint matrices between them in the same way as the rigid parts. This implies that the deformable region in a string-like deformable object can be disregarded. The entanglement with other parts needs to be considered [Sánchez et al., 2020]; however, this is beyond our scope.

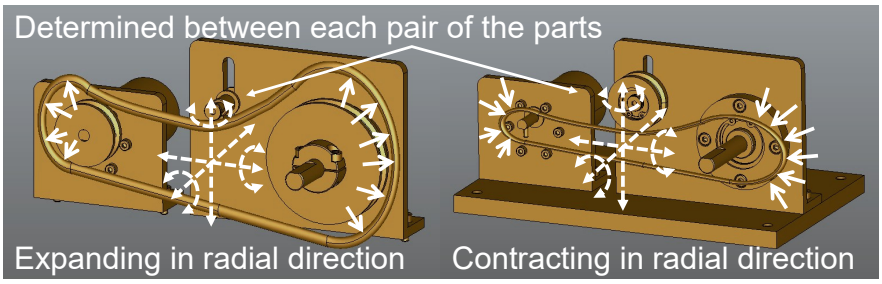
Ring-shaped parts

I describe an extraction method of the constraint-free information for a rubber band as an example of ring-shaped deformable parts. I assumed that the part deformability could be determined from the part name.

For example, the rubber band shown in Figure 3.9 transmits the rotation of the motor shaft to another pulley. The rubber band must be stretched and retracted in the radial direction when attached to a pulley groove in the assembly as humans



(a) Avoiding interference by deforming a rubber band



(b) Determination of interference

Figure 3.9. Interference determination for a deformable object. The model is deformed in radial direction (b) like a human do (a).

do.

By expanding or contracting the model in the radial direction, the constraint-free information of its deformed shape is extracted, as shown in Figure 3.9 (b). We change deformation scaling parameters. If one of the extracted constraint-free information with 12 directions becomes 1, the scaling parameters are adopted. The three matrices in the proposed ASG are obtained as with the rigid parts. The elements of insertion matrix for the ring-shaped parts are set as zero.

3.5. Generating Assembly Sequences

This section describes the method for generating the assembly sequence using the two relations described. Like the method in [Pan et al., 2006] I explore the feasible and preferable sequences through GA operator updates.

3.5.1. Initialization of Genetic Algorithm

Like other nonlinear optimizations, the initial chromosomes given to the GA affect the results of the optimization. Therefore, I propose a simple method to generate better initial chromosomes than random generation. The gene indicates an assembled part and the chromosome indicates an assembly sequence of parts.

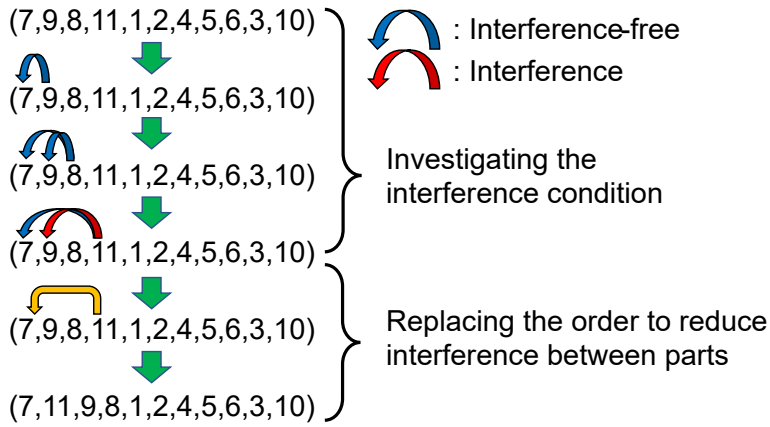


Figure 3.10. Genearation process of an initial chromosome.

To generate our initial chromosomes, first, we generate an assembly sequence randomly. Next, we check the feasibility of this sequence from the beginning. If we find that the i-th part collides with the j-th part ($j < i$), we change the sequence to assemble the i-th part just before the j-th part. The obtained chromosome is assigned as a part of the initial chromosome. This process is repeated a predefined number of times to generate the initial chromosomes.

Figure 3.10 shows an example of the proposed process for generating the initial chromosome from randomly generated chromosomes. In this example, because Part 11 collides with Part 9, we move Part 11 before Part 9.

3.5.2. Genetic Operation

We use the gene operation proposed in [Smith and Smith, 2002]. When the chromosomes of the i-th generation are given, n_t chromosomes with a high fitness value are selected for the next generation. The cross, mutation, cut-and-paste, and break-and-join operations are performed on the chromosome to generate the $i+1$-th generation chromosomes. Optimization is performed until the n_l-th generation chromosome.

Figure 3.11 shows the GA operations. Figure 3.11 (a) shows an example of the crossover operation. The crossover operator reorders the two parent assembly sequences following a randomly chosen crossover point. The sequence behind the crossover point is changed to the order in which it appears in the other sequence (the order of parts circled in red in Figure 3.11 (a)). The crossover operator produces two new offspring assembly sequences. Figure 3.11 (b) shows how the mutation operator swaps two randomly chosen parts within a single assembly sequence. Figure 3.11 (c) shows how the cut-and-paste is applied within a single

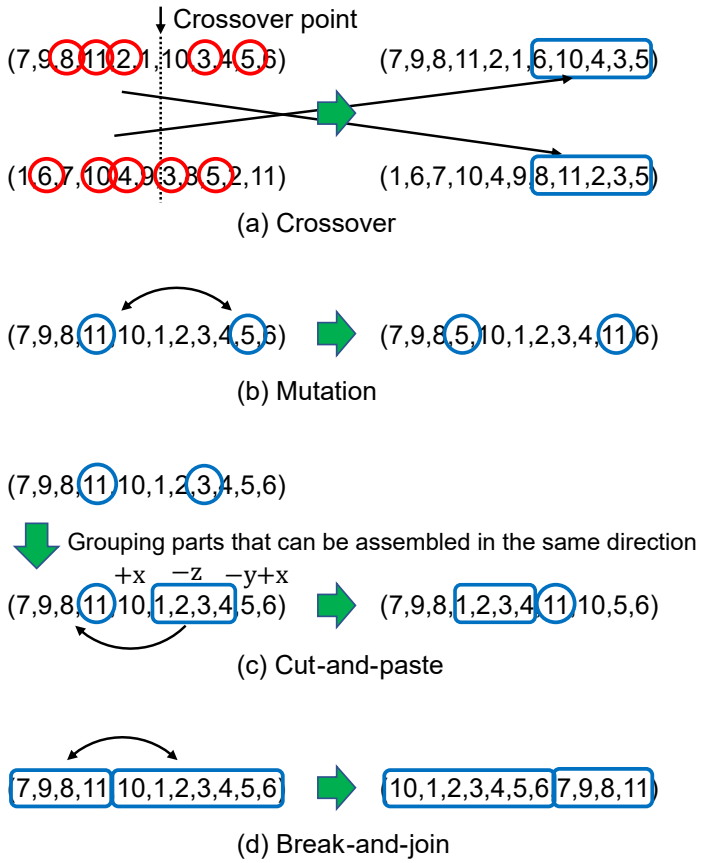


Figure 3.11. Genetic operators in the GA.

assembly sequence. The cut-and-paste operator moves a group of related parts from a randomly chosen cut position to a randomly chosen paste position. After selecting the cut position and the paste position, if a cut part and several adjacent parts use the same assembly direction, then the cut-and-paste operator moves the entire group of related parts to the chosen paste position. Figure 3.11 (d) shows how the break-and-join operator breaks a single assembly sequence into the two at a randomly chosen break point and then swaps the two parts. The probabilities of the cross, mutation, cut-and-paste, and break-and-join operations in our experiments were 20%, 10%, 35%, and 35%, respectively.

3.5.3. Designing Fitness Function

This section describes the designs of two fitness functions (A) and (B) used in the proposed method and the comparison method, respectively. The proposed

method defines a suitable sequence as one where a male part is assembled after the corresponding female part. The design of the fitness function follows the idea of manipulating a peg into a hole in a fixed part. The value goes up if the female part is assembled just before the male part and goes down if the female part is not assembled before the male part. Thus, our fitness function (hereinafter referred to as function (A)) is defined as:

$$f_i(s) = \begin{cases} 2\eta + \alpha(s) - \beta(s) - r(s) & \text{feasible} \\ \eta/2 & \text{infeasible} \end{cases}, \tag{3.3}$$

where η is the number of parts; s is the assembly sequence, and $r(s)$ is the number of changes in the direction of assembly. $\alpha(s)$ is the number of times that the female parts are assembled just before the male parts. The case where the parts collide with each other when assembling the parts is defined as infeasible, otherwise it is defined as feasible. $\beta(s)$ is the number of times that female parts are not assembled before male parts. The values of $\alpha(s)$ and $\beta(s)$ are easily calculated using the insertion matrix.

I used the fitness function defined in [Smith and Smith, 2002] for comparison in our experiment described in Section 3.6. This method considers a suitable sequence to be one with only a small number of changes in the assembly direction. Thus, they defined the fitness function (hereinafter referred to as function (B)) as in the following formula. They do not consider the insertion relations.

$$f_f(s) = \begin{cases} 2\eta - r(s) & \text{feasible} \\ \eta/2 & \text{infeasible} \end{cases}, \tag{3.4}$$

Finally, I describe the fitness function to decrease the CSTD. The fitness function is designed such that if the assembly is infeasible, the evaluation is the lowest; otherwise, it is designed such that the sequence with the lowest CSTD receives the highest evaluation. Minimizing the CSTD must be solved for each part assembly based on the fitness function (hereinafter referred to as fitness 2) calculated as:

$$f_c \coloneqq \begin{cases} 12(\eta - 1) - H & \text{feasible} \\ 0 & \text{infeasible} \end{cases}. \tag{3.5}$$

This value is 0 for infeasible assembly. The feasibility is determined using the method of Smith *et al.* [Smith et al., 2001]. In Equation (3.5), η is the number of parts and H is the maximum CSTD. According to the definition, the maximum constraint of two parts is 12; therefore, H in Equation (3.5) is less than $12(\eta - 1)$.

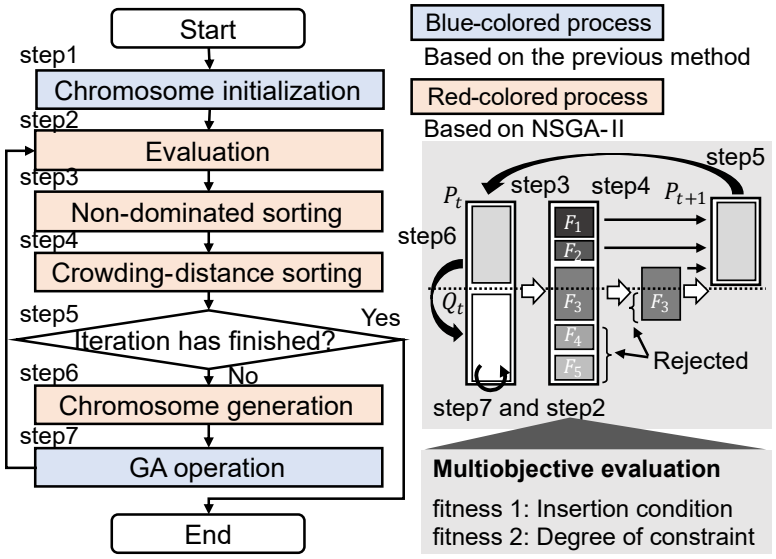


Figure 3.12. Assembly sequence optimization. The blue and red blocks are based on the previous method [Tariki et al., 2021] and NSGA-II [Deb et al., 2002], respectively.

3.5.4. Multiobjective Optimization

To solve the MO problem, I built an algorithm based on NSGA-II [Deb et al., 2002], an MOGA that provides high search performance for 2–3 objective MOPs. Figure 3.12 shows the proposed algorithm. We design the fitness function to evaluate the insertion condition and CSTD between parts. The blue part of Figure 3.12 is detailed in [Tariki et al., 2021] and includes chromosome coding, chromosome initialization, and genetic operation.

The chromosome initialization and the GA operations are based on our proposed method and the others processes are based on the NSGA-II. In our method evaluated in Section 3.7, the fitness 1 is set as the function (A) related to the insertion conditions between parts and the fitness 2 is set as the function (C) related to the degree of constraint between parts.

3.6. Evaluating Single-Objective Sequence Optimization

3.6.1. Setup

To evaluate the effect of randomness in the initialization, I performed 30 trials of the GA and terminated the optimization after 300 generations for each trial. The number of chromosomes used in each trial of GA was set as η, the number of parts. The proposed method is structured in two parts, from the proposed chromosome

Table 3.1. GA parameters used in our experiments.

Parameter	Value
Number of chromosomes	η
Crossover rate	0.2
Mutation rate	0.1
Cut-and-paste rate	0.35
Break-and-join rate	0.35
Number of generations	100
Number of iterations	10

initialization (referred to as *rearrange*) and function (A) used to evaluate the insertion relations (rearrange + function (A)). To evaluate the effectiveness of each part, they are compared with the random initialization method and function (B). The four possible combinations of the four methods, random + function (A), random + function (B), rearrange + function (A), and rearrange + function (B), are compared in the experiments.

Two case studies with two CAD models were prepared for the experiments. In Case Study 1, I used the 3D model from [Tao and Hu, 2017] to evaluate the effectiveness of the insertion matrix. In Case Study 2, I used the complex CAD model of the belt drive unit (Figure 3.15) consisting of 32 parts except the rubber band. In this case study, I verified whether the proposed method could automatically determine the assembly sequence for a product composed of many parts.

I verify the effectiveness of the proposed fitness function through evaluation of the insertion relations as represented by α and β. The first priority is to keep the insertion relations on male-female parts, and I determine whether it is difficult to satisfy the insertion relation without using function (B). Furthermore, comparing the proposed initialization with random initialization, I show if the proposed initialization is effective in finding feasible sequences for a model composed of many parts.

3.6.2. Case Study 1

Figure 3.1 shows the model used in this case study. The model has five parts. Part 1 is the bolt, which has a hole where Part 4 will be inserted. Parts 2 and 3 also have holes which line up, where Part 1 is inserted. Part 4 is a pin which is inserted in the hole in Part 1. Part 5 has two holes, one where Part 1 is inserted and the other where Part 4 is inserted through both Part 5 and Part 1.

Female parts

Male parts	1	2	3	4	5
1	0	1	1	0	1
2	0	0	0	0	0
3	0	0	0	0	0
4	1	0	0	0	1
5	0	0	0	0	0

Figure 3.13. Insertion matrix generated in Case Study 1.

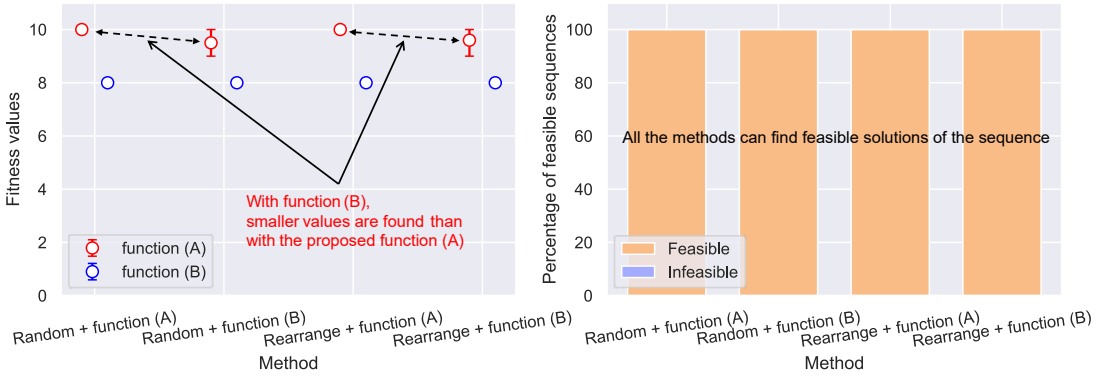


Figure 3.14. Statistics regarding sequence generation in Case Study 1.

Figure 3.13 shows the insertion matrix generated in Case Study 1. There are no extraction errors in the insertion relation matrix (0 or 1), this indicates 100.0% (25/25) accuracy achieved. Table 3.2 shows assembly sequences obtained by the proposed method (rearrange + function (A)) and the comparison methods. The sequence generated by the proposed method fulfills all the insertion conditions. On the other hand, if the other comparison methods are used, surprisingly the insertion condition of the sequence is achieved.

The left graph of Figure 3.14 shows the average fitness values (marked as circle) with minimum and maximum values (indicated by error bar) of fitness functions (A) and (B) of generated sequences over 30 trials of the optimization (absence of the error bars indicates that the calculated values were the same each trial). The right graph of Figure 3.14 shows a stacked bar graph showing the percentage of feasible sequences in the generated sequences. In such a case when the number of parts is small, there is not much difference in the fitness values of the two fitness functions of each method, and every final solution is a feasible sequence without any interference between parts. However, optimization under function (B) resulted in the output of solutions with a slightly bad evaluation of function

Table 3.2. Results of assembly sequence generation in Case Study 1.

Method[a]	Sequence	Direction	Fitness[b]
Random + function (A)	(2,3,5,1,4)	(+x,+x,+x,+y,+z)	10.0/8.0
Random + function (B)	(3,2,5,1,4)	(-x,-x,-x,+y,+z)	10.0/8.0
Rearrange + function (A)	(5,3,2,1,4)	(+z,+z,+z,+y,+z)	10.0/8.0
Rearrange + function (B)	(2,5,3,1,4)	(-x,-x,-x,-y,+z)	10.0/8.0

[a]Each method is distinguished by the chromosome initialization method and the fitness function used for GA optimization.
[b]Fitness values calculated by the fitness functions (A) and (B) shown as (A)/(B).

(A).

3.6.3. Case Study 2

Figure 3.15 shows the product used in this case study. There are 33 parts to this product. Because the proposed method cannot handle deformable parts, the rubber band was excluded, leaving 32 parts.

The ground truth of the insertion matrix, which is shown in Figure 3.16, was obtained manually. Figure 3.17 shows the insertion matrix generated in Case Study 2. Comparing Figure 3.17 and Figure 3.16, there are only three differences, which results in 99.71% (1021/1024) accuracy. Figure 3.18 shows the three pairs of the parts for which insertion relation extraction failed.

The (6,3)-element of Figure 3.18 (a) should be the insertion relation, but the system failed to extract the correct relation. Part 6 is inserted against Part 3, but Part 6 also has a hole in the center. Because the center of the hole in Part 3 is the hole in Part 6, it is not determined to be an insertion due to having no interference between the small box for Part 3 and Part 6 (described in Section 3.4.2).

The (9,6)-element (Figure 3.18 (b)) and (16,3)-element (Figure 3.18 (c)) should not be the insertion relation, but the system again failed to extract the correct relation. This seems to be because, if the female candidate (Part 6 or 3) has a shallow hole, and the small box generated at the center of the hole shifts even a little due to part placement errors, interference with the male candidate (Part 9 or 16) may be detected. Then as a result, the relation is determined to be an insertion. In other words, it is necessary to devise the placement and shape of small boxes.

In the first step of the GA, chromosome initialization, each method generated different chromosomes. For Case Study 2, three examples of the randomly generated initial chromosomes are:

Table 3.3. Results of assembly sequence generation in Case Study 2.

Method[a]	Sequence	Direction	Fitness[b]
Random + function (A)	(3,23,18,15,8, 14,17,21,7,19, 24,2,29,13,30, 31,25,5,1,6,16, 10,28,20,32,12, 27,4,22,9,11,26)	-	16.0/16.0
Random + function (B)	(6,16,11,22,31, 12,9,1,3,2,18, 17,26,13,20,24, 23,21,15,8,14, 25,29,30,27,32, 7,10,28,19,5,4)	(-z,-z,-z,-z,-z, -z,-z,-z,+z,+z, -y,-y,-z,-z,-y, -z,-z,-z,+z,+z, +z,+z,-z,-z,-z,-z, -z,-z,-z,-y,-y,+z)	41.0/55.0
Rearrange + function (A)	(15,1,3,6,23,11, 9,2,4,8,14,16, 17,19,21,22,24, 10,12,13,29,30, 7,26,27,31,5, 32,18,20,25,28)	(-z,-z,-z,-z,-z,-z, -z,-z,-z,+z,+z, +z,-y,-y,-y,-z, -z,-z,-z,-z,-z, -z,-z,-z,-z,-z,-z, -z,-y,-y,-y,+z,-z)	61.0/58.0
Rearrange + function (B)	(4,2,28,11,1, 12,13,29,30,6, 22,21,23,24,9, 10,31,26,5,27, 16,3,15,14,8,25, 32,20,17,18,19,7)	(-z,-z,-z,-z,-z,-z, -z,-z,-z,-z,-z,-z,-z, -z,-z,-z,-z,-z,-z,-z, -z,-z,-z,+z,+z, +z,+z,+z,+z,-y, -y,-y,-y.-y,-z)	53.0/61.0

[a]Each method is distinguished by the chromosome initialization method and the fitness function used for GA optimization.
[b]Fitness values calculated by the fitness functions (A) and (B) shown as (A)/(B).

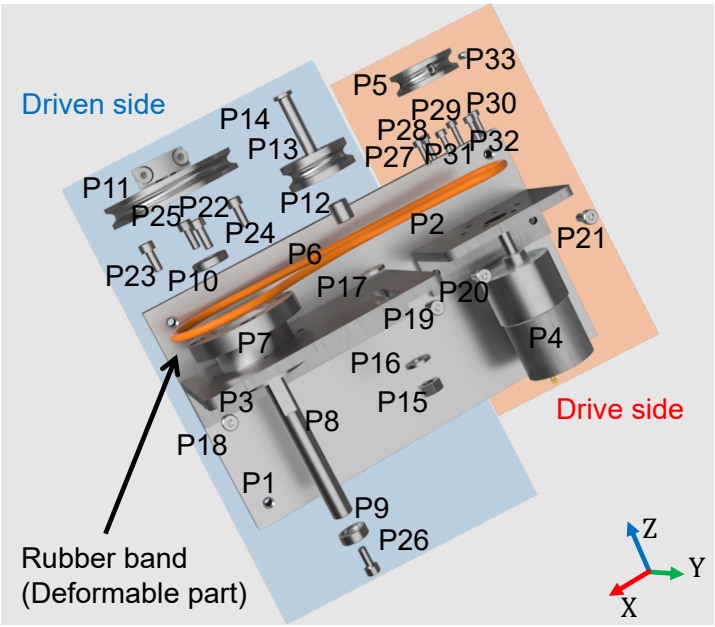


Figure 3.15. Rubber band-drive unit (#2) used in Case Study 2.

(19,8,9,6,2,27,31,13,1,7,5,25,12,24,23,20,4,
17,29,3,11,14,10,22,26,30,28,32,15,18,16,21),
(28,20,32,25,5,24,17,13,26,16,19,6,14,27,15,
9,21,29,7,30,23,11,3,22,31,10,8,18,12,1,2,4),
(24,14,31,20,17,8,21,18,2,12,4,1,23,28,9,10,30,
29,16,13,25,32,6,26,7,15,22,11,19,5,27,3).

All the fitness values of these randomly generated chromosomes are 16.0. This value indicates that at the initial point, there are still infeasible sequences or feasible but low evaluations. The proposed method initialized the chromosome as follows:

(24,28,12,13,23,3,7,27,9,10,25,21,15,5,17,29,
26,16,18,20,6,14,8,31,30,22,32,1,11,2,4,19),
(24,28,12,13,23,21,3,7,27,9,10,25,15,5,17,29,
26,16,18,20,6,14,8,31,30,22,32,1,11,2,4,19),
(24,28,11,12,16,13,23,21,6,22,3,7,27,9,8,10,
25,15,31,30,5,17,29,26,18,2,20,14,32,1,4,19).

The fitness values of the sequences are 16.0, 16.0, and 48.0, respectively. The value 48.0 indicates that this sequence is initially feasible.

The left graph of Figure 3.19 shows the average fitness values (marked as circle) with minimum and maximum values (indicated by error bar) of the fitness

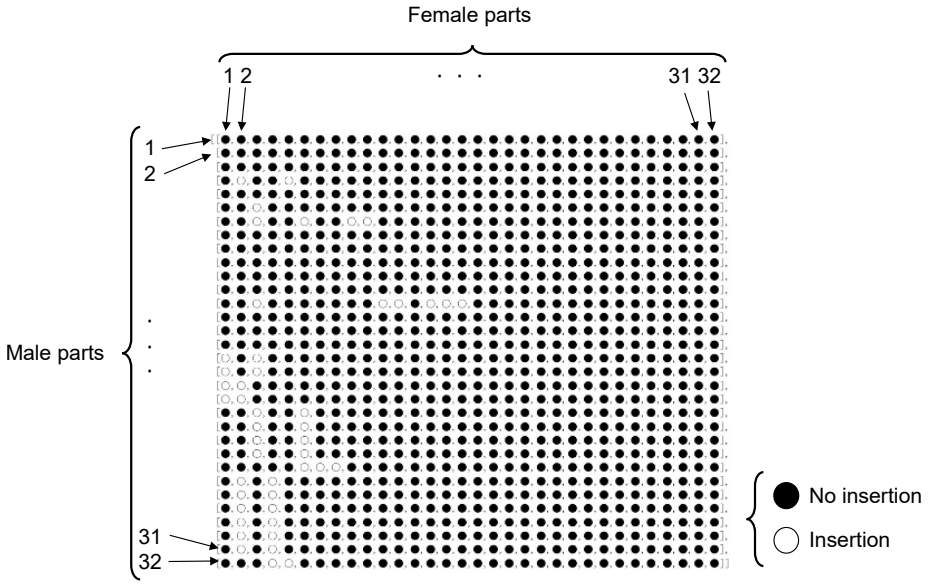


Figure 3.16. Ideal insertion matrix in Case Study 2.

functions (A) and (B) of generated sequences over 30 trials of the optimization (absence of error bars indicates that the calculated values were the same each trial). The right graph of Figure 3.19 shows a stacked bar graph showing the percentage of feasible sequences in the generated sequences.

First, with random initialization, function (A) did not necessarily give higher fitness values than the fitness values of the sequences generated by function (B). Secondly, when comparing the two initialization methods, when we use a method with rearrange initialization showed that the fitness values were higher whichever the two functions are used. This is influenced by the number of times that the exploring assembly sequence converged to an infeasible solution, as shown in the right graph of Figure 3.19. Furthermore, comparing the two methods with rearrange initialization, the fitness value of function (A) of the assembly sequences generated with function (A) was higher than with function (B), and the fitness value of the assembly sequences generated with function (B) was almost the same in each method. This makes the effectiveness of function (A) obvious. Therefore, I believe that both of the two proposed methods, rearrange initialization and function (A), are effective in generating an assembly sequence that is feasible and satisfies the insertion condition.

Figure 3.20 shows animated frames of the assembly sequences generated by both the rearrange + function (A) and rearrange + function (B) in Case Study 2. In Figure 3.20 (b), many male parts such as a motor and pulleys are assembled earlier. On the other hand, in Figure 3.20 (a), many male parts are assembled

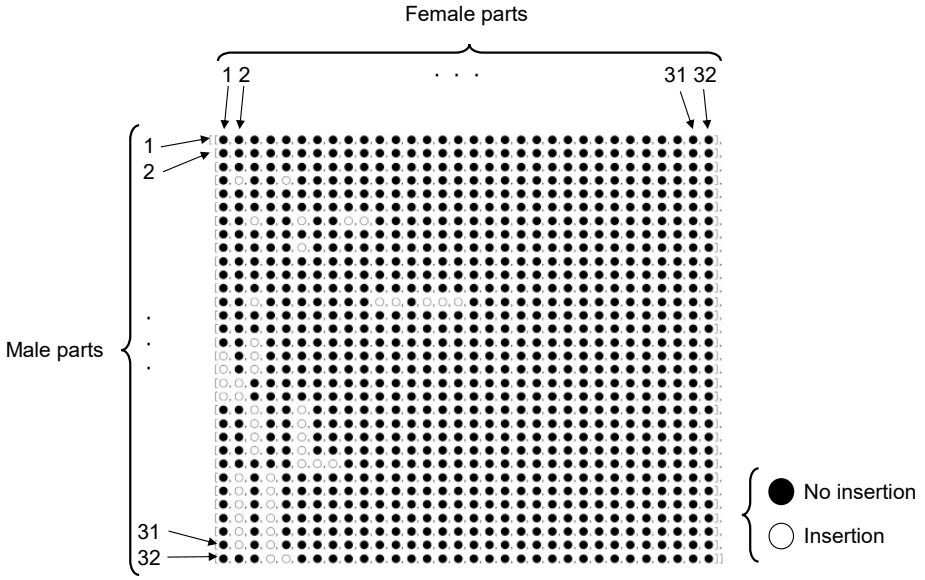
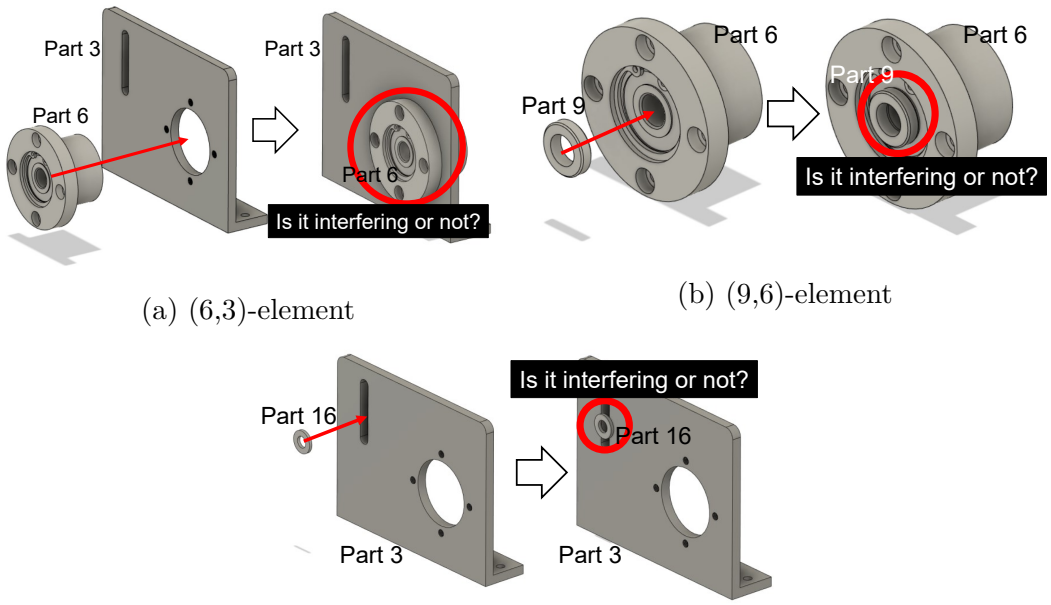


Figure 3.17. Insertion matrix generated in Case Study 2.

after assembling the female parts. For example, as shown from Step 1 to Step 2 and from Step 2 to Step 3 of Figure 3.20 (a), after Part 3 (Plate) is assembled, Part 6 (Pulley) is inserted into the hole in Part 3, after Part 2 (Plate) is assembled, Part 4 (Motor) is inserted into the hole in Part 2. These are the expected results of the proposed method.

Table 3.3 shows the assembly sequences generated using the proposed method (rearrange + function (A)) and the comparison methods. Comparing the final fitness values of functions (A) and (B) for the four methods shown in Table 3.3, the sequence generated by the proposed method had the highest fitness value with function (A). The fitness value of function (B) is also as high as that of rearrange + function (B). This means that the combination of rearrange initialization and the use of fitness function (A) allows us to generate the sequence expected in terms of both the direction changes and the insertion conditions. An assembly sequence with high fitness values was obtained even with random initialization in Case Study 1, but the insertion relation is hard to be satisfied for a model with numerous parts, as in Case Study 2. Optimization that limits the changes in assembly direction [Smith and Smith, 2002] does not produce an assembly sequence that takes the insertion relations into account, which would require the sacrifice of changing the orientation to some extent.



(a) (6,3)-element

(b) (9,6)-element

(c) (16,3)-element

Figure 3.18. Pairs of parts for which insertion relation extraction failed.

3.6.4. Discussion

The experimental results in Case Study 2 show the importance of considering the insertion relations. It is possible to consider the insertion relations using other methods, such as by understanding the functionalities of parts (screws, pins, nuts, etc.). This may require an additional effort of collecting the data for machine-learning-based recognition.

However, as in Case Study 1, this does not necessarily work well for generating the assembly sequence. It might be reasonable to insert Part 5 after assembling Parts 1, 2, and 3. Part 5 is a female part for Part 1. If Part 1 is assembled after Parts 2, 3, and 5, three parts are fixed by Part 1 at the same time. Yoshikawa *et al.* [Yoshikawa et al., 1991] suggests choosing a sequence where the number of constraints increases mildly. Though the insertion relations should be considered, as the result of Case Study 2 shows, the proposed fitness function has room for improvement.

I evaluated methods to support generating the assembly sequence of parts using only a CAD model. First, I evaluated a method for quickly generating better initial chromosomes for Genetic Algorithm (GA) to use to generate the assembly sequence of products with many parts (in our case, 32). Second, to generate an assembly sequence that satisfies the insertion conditions of assembled parts, I

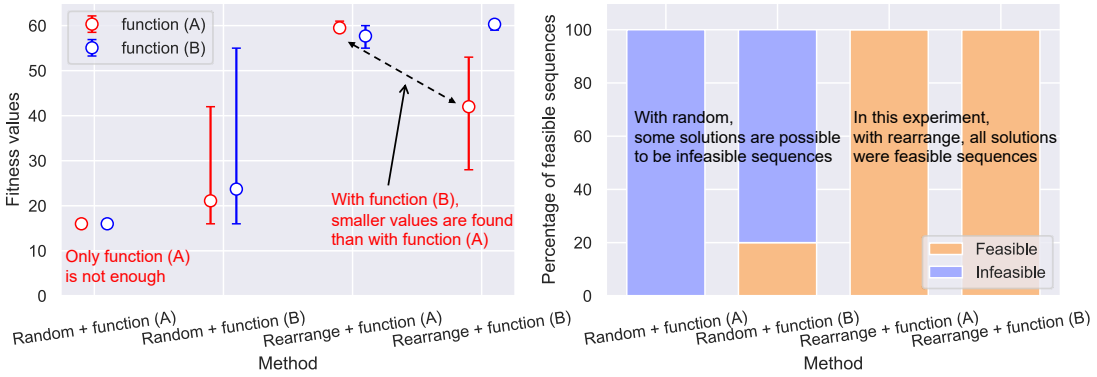
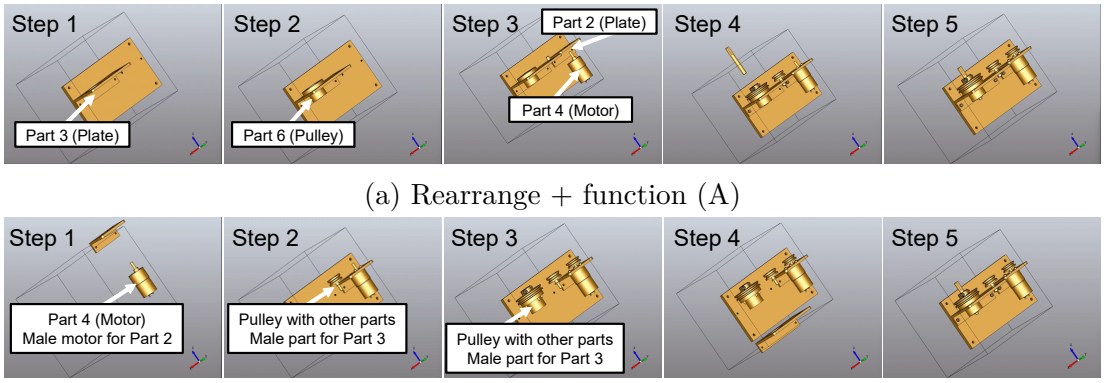


Figure 3.19. Statistics regarding sequence generation in Case Study 2.



(a) Rearrange + function (A)

(b) Rearrange + function (B)

Figure 3.20. Animation frames showing the results of Case Study 2.

proposed and evaluated a way to calculate insertion relations (e.g., male-female pairs) from a CAD model with PythonOCC. The fitness functions of GA used in this experiment was defined 1) to encourage assembling a male part just after the corresponding female part and 2) to avoid assembling a male part before the corresponding female part.

In the experiments of assembly sequence generation with and without the proposed methods, the proposed method was able to generate an assembly sequence for a product composed of 32 parts. The simulation results demonstrated the usefulness of the proposed chromosome initialization method and of insertion matrices; the generated assembly sequence had no interference between the parts and satisfied the insertion conditions.

3.7. Evaluating Multiobjective Sequence Optimization

3.7.1. Outline

I conducted three case studies of MOGA with parameters shown in Table 3.1. Case Study 1 used a model shown in Figure 3.1 to confirm whether the aforementioned problem can be solved.

Case Study 2 used a model of the rubber band-drive unit (Figure 3.15) comprising 33 parts used for an assembly challenge [Yokokohji et al., 2019]. I investigated the possibility of the ASG for many parts, including a deformable part.

Case Study 3 was conducted to verify whether the proposed method can generate sequences for several models where the types of some parts used are slightly different. I used eight models: two models used in Case Study 1 (#1) and 2 (#2), a model that includes copper wires with pins inserted to a terminal block extended from the rubber band-drive unit used in Case Study 2 (#3), two rubber band-drive units which are different from the model in Case Study 2 (#4 and #5), two rubber belt-drive units (#6 and #7), and a chain-drive unit (#8) used in the assembly challenge. Furthermore, using three models #1, #2, and #3, I evaluated the reproducibility of the ASG. Figure 3.21 shows models for Case Study 3. See the rulebooks [WRS2018; WRS2020] of assembly challenges for more details.

Furthermore, using relatively three different models #1, #2, and #3 out of the eight models, I evaluated the reproducibility of the ASG. The models for Case Study 3 are shown in Figure 3.21. See the rulebooks [WRS2018; WRS2020] of the assembly challenges for more specification details.

3.7.2. Case Study 1

Figure 3.22 shows the final assembly sequence with the highest sum of the fitness values of Equation (3.5) and Equation (3.3) among the assembly sequences of 10 trials. The sequence shown in the left-hand side in Figure 3.1, depicting the simultaneous occurrence of contacts was removed and the assembly sequence with a low CSTD was generated.

Figure 3.1 (left) shows that when P1 is inserted, constraints occur simultaneously on P2, P3, and P5, and the CSTD is 24 (= 8 + 8 + 8). In contrast, as shown in Figure 3.22, when P1 is inserted, constraints occur with both P3 and P5, and the CSTD is 16 (= 8 + 8). In the insertion of P5, the constraints with only P1 and P3 are 13 (= 8 + 5). In both these cases, the CSTD is less than 24. The assembly of the other parts also shows a CSTD of less than 16; thus, the maximum value of the CSTD could be reduced from 24 to 16.

3.7.3. Case Study 2

Figure 3.23(a) shows the convergence curve of the MOGA. The two curves shows the average fitness values of all chromosomes of each generation. They are calculated using fitness 1 (red curve) and fitness 2 (green curve). The number of interference-free sequences remained at 33, indicating that 100% of the generated sequences are feasible. This indicates that the values may have converged to quasi-optimal values during the first generation update. An unsteady variation is observed in the evaluated values until near the 20th generation update after which the fitness values of the generated sequences are stable and produce high fitness values.

In this study, the number of generation updates was 100; however, as shown in Table 3.4, even when the number was the small such as 1 or 5, the generated sequence was still feasible. There is room to adjust the number of generation updates to reduce the time required for MO.

Figure 3.24 shows the generated sequence with the highest sum of the fitness values depicted as the blue dot in Figure 3.23(b). Considering only the insertion condition, it would be reasonable to assemble P5 and P2 before P4. However, the CSTD of insertion sequence of P4 into P5 and P2 is high. In the generated sequence, P5 is assembled last, thus the CSTD in the assembly of P2, P4, and P5 was reduced.

Figure 3.23(b) shows the two fitness values of 33 generated sequences ($= \eta$). Fitness 1 is over $16.5 = \eta/2$ and fitness 2 is over 0 (these values means when infeasible) in all generated sequences, and an interference-free sequence was generated even in the assembly for deformable parts. The solution near the blue dot in Figure 3.23(b), where the sum of both fitness values is maximum, is possible to be a Pareto-optimal sequence.

To verify the Pareto optimality of the solution with the best fitness value, I investigated whether the other fitness value increases or when the order of one part is changed. In other words, because finding the optimal solution is NP-complete problem, in this experiment, I show that the solutions in the neighborhood is worse than our final solutions (the sequence with the highest sum of the fitness values).

Figure 3.23(c) shows fitness values of the sequences generated by reordering one part, thus the number of sequences simulated is 1024 ($= (\eta-1)^2$). The number of feasible sequences is 40.3% ($= 413/1024$). I confirmed that no sequence obtained by reordering increased both fitness values over the best solution shown as the blue dot in Figure 3.23(b). Therefore, the generated sequence may satisfy Pareto optimality.

Table 3.4. Computation times of the optimization on two CPUs in Case Study 2. Each item shows Ave.±Std. of the times in 10 trials. The rate of feasible sequences of the 10 sequences is 100% in all optimizations.

Generation update	Runtime of optimization [h] Core i7-3520M[a]	Core i9-9900KS[b]
1	0.901 ± 0.0500	0.336 ± 0.0180
5	1.53 ± 0.0617	0.572 ± 0.0246
10	2.31 ± 0.101	0.848 ± 0.0208
20	3.77 ± 0.0973	1.39 ± 0.0363

[a]Intel Core i7-3520M CPU@2.90GHz.
[b]Intel Core i9-9900KS CPU@4.00GHz.

Table 3.5. Reproducibility of the ASG evaluated in Case Study 3. Each element shows Ave.±Std. of the calculated fitness values. The percentage of feasible sequences for the evaluated models are 100%.

Model	fitness 1	fitness 2	Sum of 1 and 2
#1	9.50 ± 0.50	28.0 ± 4.00	37.5 ± 3.50
#2	51.7 ± 3.66	356 ± 2.69	408 ± 2.76
#3	58.7 ± 3.03	415 ± 1.20	473 ± 3.47

3.7.4. Case Study 3

The objective of this case study is to confirm the robustness and reproducibility of the proposed ASG. First, we calculate interference-free, insertion, and degree of constraint matrices for the eight models. In Figure 3.21, different assembly parts are written in letters inside each model image (#4∼#8). Because the models #4∼#8 have parts structure similar to the model #2, the two-part relations were extracted successfully. In the case of #4∼#8, using the extracted relations, the proposed ASG for all models was succeeded as with #2.

Subsequently, we apply the proposed ASG to three models #1, #2, and #3 that have largely different parts structures. Table 3.5 shows the average and variance of the maximum fitness values of the generated sequences for the three models. The percentages of feasible sequences for all models are 100%. Even multiple part changes exist in the product, the proposed method can achieve the ASG with a high reproducibility.

3.7.5. Discussion

Extensibility on Handling Deformable Parts

In the case of string-like deformable parts with snap-fit plugs, the assembly direction of the plug can be erroneously determined as interference. Recognizing the snap-fit connector as the assemblable object from the geometry in CAD are necessary [Shellshear et al., 2020].

In the case of a ring-shaped deformable parts, we require an assembled CAD model. The extent to which this deformation represented in the model depends on the product designer.

Ghandi *et al.* [Ghandi and Masehian, 2015] proposed an assembly sequence planning of deformable parts based on *Finite Element Method* (FEM) simulation. However, in the FEM analysis, a user must input parts geometries, the material, density, Young's modulus, coefficient of friction, and types of elements used for modeling its deformation behavior. The computational cost, manual input, laborious measurement, and accuracies of parameters for all parts influence the results of ASG. To replace such the method, I will develop an time-efficient ASG method based on the geometries and semantic information of parts.

3.7.6. Graspable Sequences Toward Grasp Planning

Once the assembly sequence is determined, the feasible grasping based on interferences between the robot end effector and parts must be determined. Figure 3.25 illustrates an easy approach for determining them. Figure 3.25 (a) and (b) show the process for determinating the grasping points and interferences in a sequence generated in Case Study 1. I used a parallel-jaw gripper (ROBOTIQ, Hand-E) developed for precise assembly operations. The following procedure was used.

1. Randomly sampling hand-crafted graspable points on the object surface
2. Generating concatenated models of the parts and the gripper by fixing a certain pose of the gripper
3. Determining the interference by moving the concatenated models in the simulation using CAD models

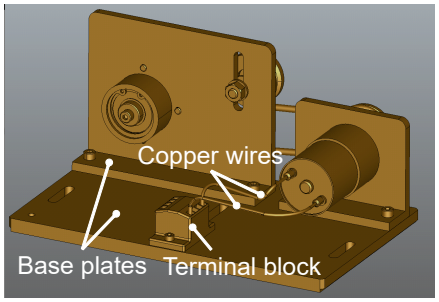
To achieve robotic grasping, such as by using the CAD-based method, we can determine the occurrence of interference.

3.7.7. Application Toward Robotic Assembly

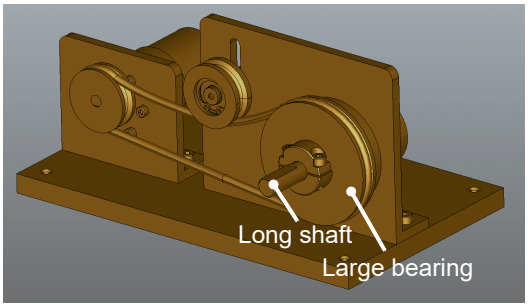
Once the graspable sequence is determined, robot manipulation motions should be considered. To confirm the feasibility and limitations of the assembly sequence generated in Case Study 2, I simulated the assembly motions shown in Figure 3.26.

I manually generated the robot hand trajectory and the grasping configuration. As a limitation of the generated sequence, after the insertion of rubber band, the robot needs to keep supporting the non-fixed parts. Based on the center of gravity of the parts (*e.g.,* [Gulivindala et al., 2020; Murali et al., 2019]), we must fix parts in stable poses on somewhere in each assembly order. Assembly jigs to fix various parts are useful. Since preparing custom-made jigs is time-consuming and labor-intensive, to achive high-mix low-volume production, I used *Soft jig* to fix all the parts. I could achieve an assembly operation using the general-purpose jigs.

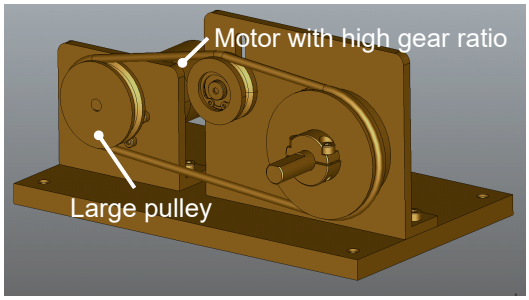
Compared to a linear assembly sequence discussed in this study, a parallel assembly sequence divided into sub-assemblies (*e.g.,* [Bahubalendruni et al., 2019; Yang et al., 2020]) are time-efficient. By generating divided sub-sequences, we can replace a linear sequence by a parallel sequence. For example, if multiple arms exist, the driven side (blue frame) and drive side (red frame) of the linear sequence shown in Figure 3.24 can be parallelized.



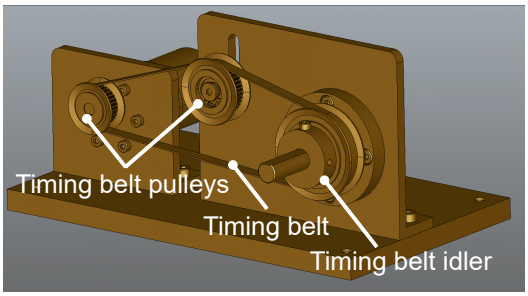
(a) Rubber band-drive unit (#3)



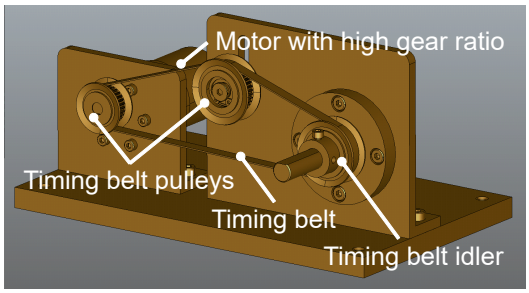
(b) Rubber band-drive unit (#4)



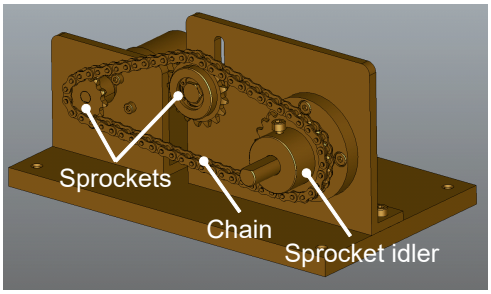
(c) Rubber band-drive unit (#5)



(d) Rubber belt-drive unit (#6)



(e) Rubber belt-drive unit (#7)



(f) Chain-drive unit (#8)

Figure 3.21. Models used in Case Study 3. The six models are revised from #2 shown in Figure 3.15. The replaced parts are shown in each figure.

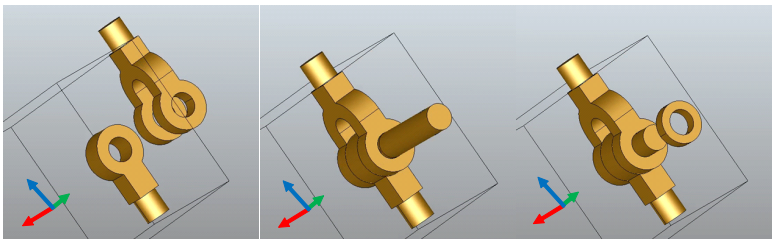
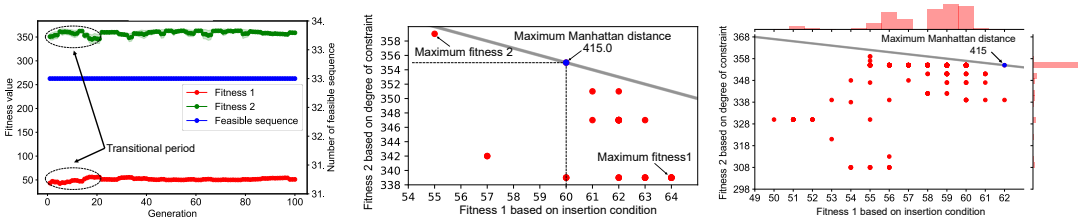


Figure 3.22. Generated sequence in Case Study 1.



(a) Convergence curve (b) Fitness values (c) Reordered sequence

Figure 3.23. Results of Case Study 2. (a) A progress of the fitness values and number of feasible sequences found during one optimization. (b) Fitness values in the 100-th generation. (c) Fitness values after reordering the generated sequence shown in (b) as a blue dot.

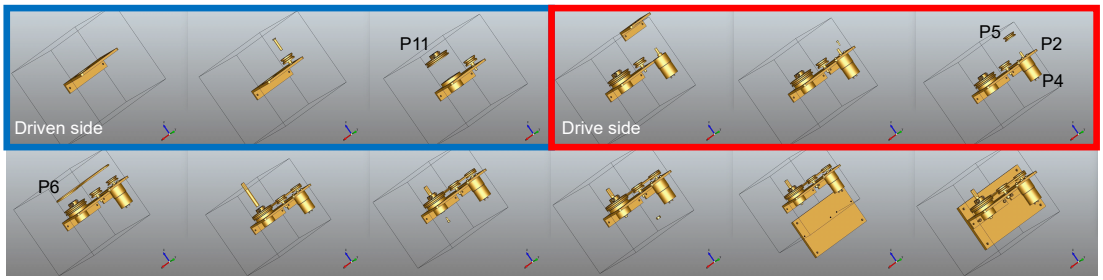
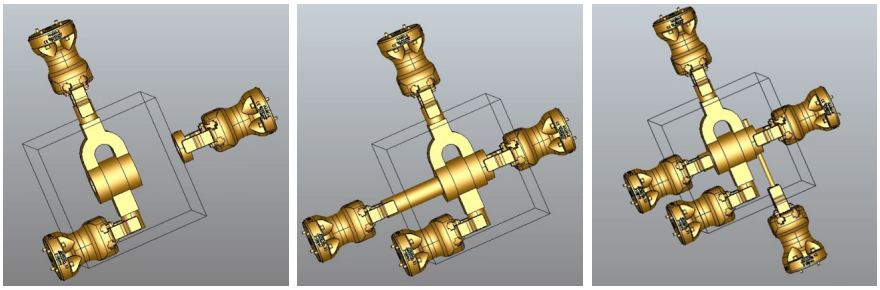
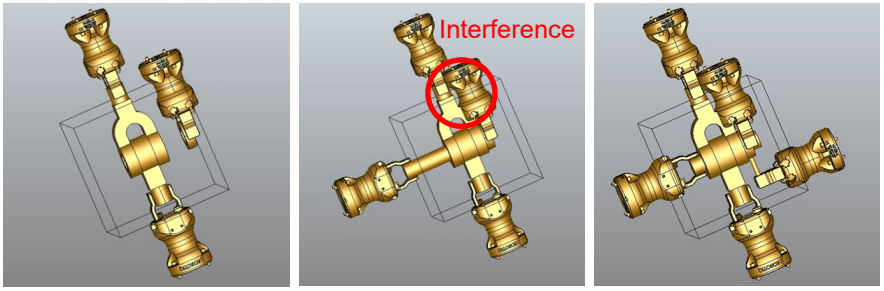


Figure 3.24. Generated sequence in Case Study 2. The order (assembly direction) is as follows 3 (-z), 17 (-z), 12 (-z), 13 (-z), 14 (-z), 7 (-z), 23 (-z), 25 (-z), 24 (-z), 22 (-z), 10 (-z), 11 (-z), 4 (-z), 2 (-z), 27 (-z), 29 (-z), 30 (-z), 31 (-z), 32 (-z), 28 (-z), 5 (-z), 33 (-y), 6 (-z), 8 (-z), 9 (+z), 26 (+z), 16 (+z), 15 (+z), 1 (+z), 19 (-y), 21 (-y), 20 (-y), 18 (-y).

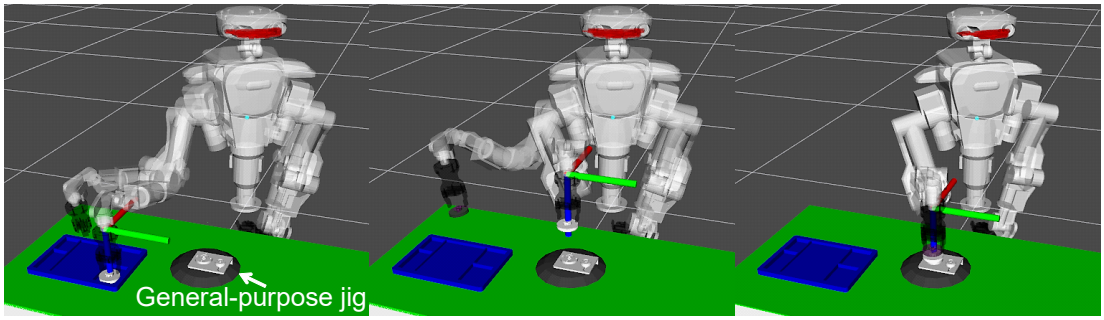


(a) Interference-free sequence

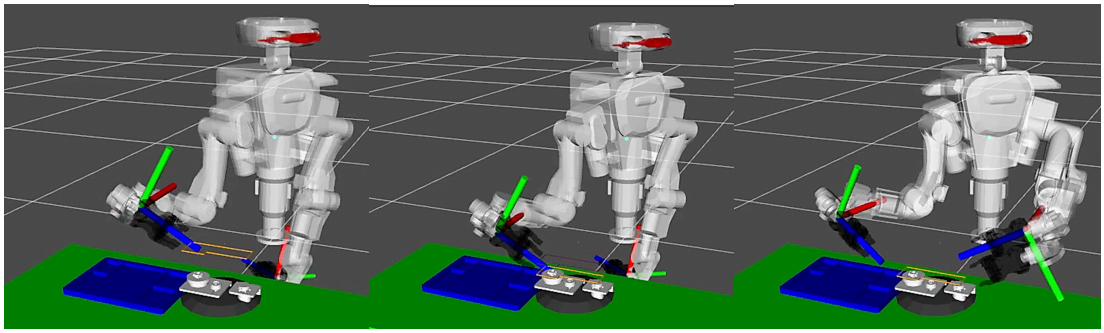


(b) Sequence that occurs interferences

Figure 3.25. Succeeded (a) and failed (b) simulation examples with a robotic gripper's model for graspable sequences.



(a) Pick-and-place of a idller (P11 shown in Figure 3.15)



(b) Pick-and-place of a rubber band (P6 shown in Figure 3.15)

Figure 3.26. A succeeded simulation example of robot motions with the graspable sequence.

Chapter 4

Manipulation with Soft Jig

4.1. Introduction

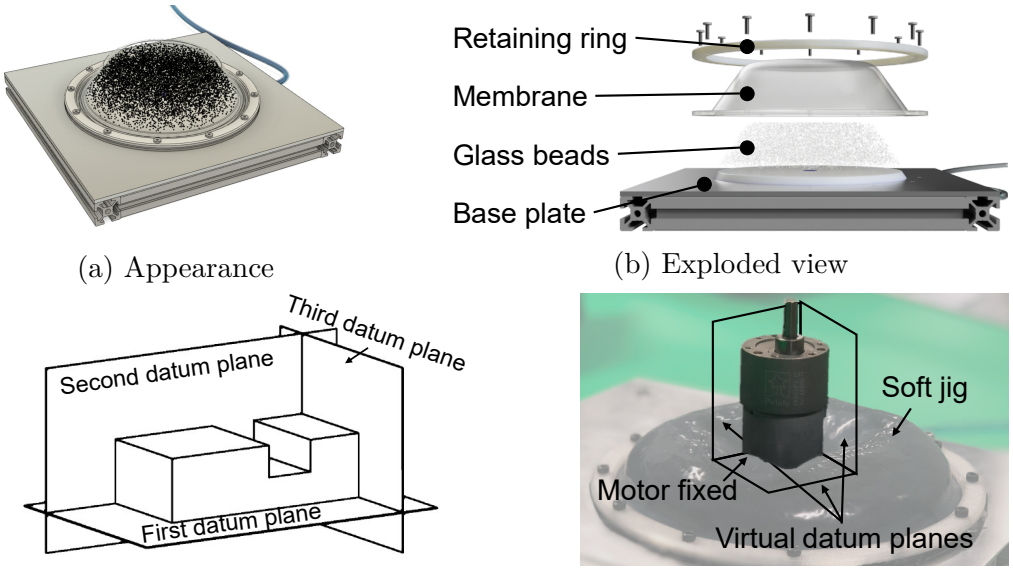
Generally, jigs are used to efficiently assemble different types of products [Rajan et al., 1999; Zhang et al., 2019] for mass production. However, in the high-mix low-volume production, it is impractical to develop custom-made jigs every time when a product is replaced.

I propose a deformable fixing device named *soft jig*, as illustrated in (a) and (b) of Figure 4.1. A part placement and positioning method on the soft jig is proposed. I evaluate the performance of the soft jig by executing assembly operations with a physical robot.

A soft jig is highly versatile as a fixture for different parts with various shapes as the jig surface deforms according to the shapes. The fixing ability of the soft jig based on a jamming transition can be used to hold assembly parts by creating datum planes (Figure 4.1(c)) on a membrane of the soft jig (Figure 4.1(d)).

The primary contribution of this study is to provide a new parts-fixing device for robotic assembly: in particular, the proposed soft jig provides a new concept of the general-purpose assembly jig that can be used for a flexible assembly robot system. I propose a design of the soft jig capable of functioning the jamming transition as the fixing ability.

In parts-assembly with the soft jig, we determine fixed parts and their postures based on the three requirements: (1) to contact between parts in one assembly operation (*e.g.,* placement and insertion), ignoring the parts without any contacts (2) to determine reachable directions of a fixed part to parts assembled on the fixed part, selecting the fixed parts and the postures by extracting interference-free parts-displacement directions using CAD, and (3) to make a part posture a low center of gravity (CoG) to achieve mechanically stable, selecting the fixed



(a) Appearance

(b) Exploded view

(c) Datum planes [Trappey and Liu, 1990]

(d) Creation of datum planes

Figure 4.1. Design of soft jig. The appearance (a) and exploded view (b) of soft jig are illustrated. Datum planes (c) (this figure is made with reference to Fig. 2 in [Trappey and Liu, 1990]) is needed to fix objects in a certain pose and it is created on the malleable membrane (d).

posture based on CAD-based calculations of CoGs for all candidates of the fixed-parts.

Our experiments demonstrated the fixing performance and versatility of the soft jig for different fixing configurations. Moreover, I examined the feasibility of assembly operations by an actual robot. I further discussed parts pose estimation.

4.2. Related Work

Several jig-less operation methods [Kim et al., 2013; Naing et al., 2000] and designing methods of general-purpose jigs [Bi and Zhang, 2001] have been proposed to reduce the human effort for designing custom-made jigs. Several researchers [Fathianathan et al., 2007; Grippo et al., 1987; Whybrew and Ngoi, 1992] developed systems for automatically designing rigid modular fixtures from a combination of elements such as locators, blocks and clamps. The designs are determined based on the work piece geometry to be fixed. Shi *et al.* [Shi et al., 2020] developed a fixing device with many metal pins that adapt to the shape of fixed parts.

Table 4.1. Comparison of general-purpose assembly jigs.

Method	Easiness of fixing	Versatility	Parts-positioning
Jig with supports	✗*a	✓*b	✓*c
Soft jig (proposed)	✓*d	✓*e	✗*f

*a Supports need to be placed in postures
*b Surrounding supports fit the object shape
*c Rigid supports fix the object in a certain pose
*d On-off control of air pressure
*e Deformable body fits the object shape
*f Stiffness of the malleable membrane is lower that the metal jig surface

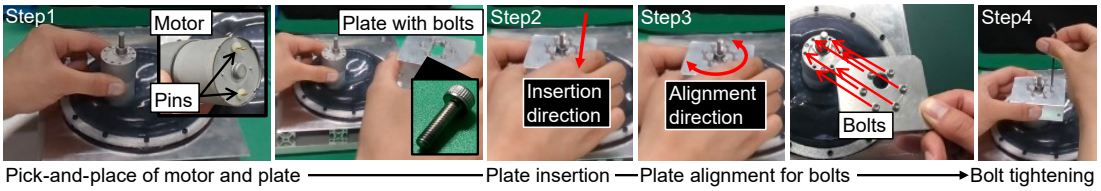


Figure 4.2. Manual assembly sequence with the soft jig. In the pick-and-place of the motor, we consider if the plate can be approached onto the soft jig. Furthermore, we can achieve fixing a motor even if the bottom surface of the motor has pins because of the malleable membrane.

Although high-precision positioning is possible with rigid jigs, they needs to be replaced according to the fixed parts with different shapes. Furthermore, previous approach to substitute the rigid jigs needs control of the actuators [Kim et al., 2013; Naing et al., 2000; Shi et al., 2020] or calculation for the shape optimization [Bi and Zhang, 2001; Grippo et al., 1987; Whybrew and Ngoi, 1992]. Hence, the easiness of fixing and versatility are relatively low. Table 4.1 summarizes the comparison between the previous and proposed general-purpose assembly jigs. The proposed fixture requires a pose estimation whereas increasing the easiness of fixing and the versatility.

Several studies have used flexible robotic end-effectors that fit the object shapes to be manipulated [Lee et al., 2017a; Watanabe et al., 2017]. Brown *et al.* [Brown et al., 2010] proposed a jamming gripper that can grasp rigid objects of various shapes. The gripper surface is covered with a silicon membrane filled with powder particles. The extensibility of the jamming gripper has been discussed in terms of recognition of part shapes [Alspach et al., 2019; Sakuma et al., 2018] and sensing during robotic manipulation [Lu et al., 2020; Sakuma et al., 2019].

The applicability of jamming gripper has been researched for different purposes, such as feet of robots to walk on natural terrain [Chopra et al., 2020; Lathrop

et al., 2020], and to climb walls [Fujita et al., 2018]. Such soft robotics technologies are expected to be applied to the field of robotic assembly [Hamaya et al., 2020a,b] for the high-mix low-volume production.

4.3. Assumptions and Problem Setting

The fixing planning and shape of the custom-made jigs for mass production are designed according to an assembly sequence. Contrary, the fixing planning of the soft jig for high-mix low-volume production needs to be considered independently from the assembly sequence of the short life-cycle products. Thus, a certain assembly sequence is given.

As an example, let us consider the assembly task shown in Figure 4.2, three types of parts shown in Figure 4.3(a) are handled. The assembly task includes fundamental operations frequently conducted by humans: picking, placing, inserting, and screwing [Fukuda et al., 2019; Yamazaki et al., 2018]. We first grasp the motor, then the motor shaft is inserted into the plate's hole. Subsequently, we aligns the bolts with the motor's holes, and finally the bolts are tightened.

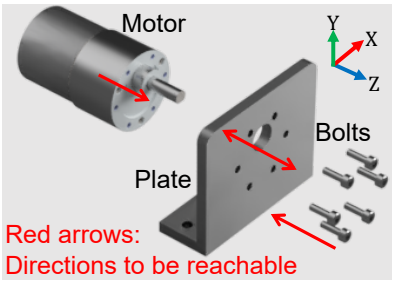
One assembly step consists of assembling two (partially assembled) parts, thus the parts that should be in contact with other parts are target parts for the assembly with the soft jig. One way to achieve the one step is that one part is fixed and the other part is manipulated. Since the manipulated part needs to be reachable from outside onto the fixed part, the fixed pose must be planned as interference-free. At least one manipulator should manipulate a part on the part stably fixed. Thus, we make the CoG of the fixed part low.

4.4. Design of Soft Jig

The designed structure of the soft jig is depicted in (a) and (b) of Figure 4.1. Figure 4.4 shows the specifications of the proposed soft jig. Silicone rubber with a Shore A hardness of 2 (Smooth-On, Dragon Skin FX-Pro) was used for the elastomer membrane (1 mm in thickness and 160 mm in diameter) to form the malleable surface of a bag.

The bag with 296 cm^3 capacity was filled with glass beads of 450 g with a diameter of approximately 1 mm (Fuji Manufacturing Co., Ltd., Fuji Glass Beads FGB-20). I selected the glass beads because they do not corrode. The curvature radius of the bag was about 60 mm.

By vacumming out air inside the jig from an air port under the jig base, the rigidity of the soft jig can be altered, and a target part can be fixed. We use an off-board vacuum pump and the confining pressure inside the bag was about



(a) Structure of model

	Motor	Plate	Bolts
Motor	0	1	0
Plate	0	0	0
Bolts	1	1	0

(b) Contact matrix

+x	Motor	Plate	Bolts
Motor	0	0	0
Plate	0	0	0
Bolts	0	0	0

+y	Motor	Plate	Bolts
Motor	0	0	0
Plate	0	0	0
Bolts	0	0	0

+z	Motor	Plate	Bolts
Motor	0	1	1
Plate	0	0	1
Bolts	0	0	0

−x	Motor	Plate	Bolts
Motor	0	0	0
Plate	0	0	0
Bolts	0	0	0

−y	Motor	Plate	Bolts
Motor	0	0	0
Plate	0	0	0
Bolts	0	0	0

−z	Motor	Plate	Bolts
Motor	0	0	0
Plate	1	0	0
Bolts	1	1	0

(c) Interference-free matrix

Figure 4.3. Assembly parts, contact matrix, and interference-free matrix used in our experiments. The two matrices are calculated based on a CAD model assembled and are used to configure the parts-fixing.

90 kPa. The parts-fixing performance benefits from the high friction, elongation, and contraction ratio of the elastomer material.

The parts-fixing process is as follows: before placing the target part, the jig surface is initialized by pumping air into the jig because the fixing performance depends on the initial shape of the membrane [Amend et al., 2012]. Subsequently, the target part is grasped, transported, and placed on the jig. In a state of pushing the part onto the jig, the part are fixed by taking advantage of the jamming transition by evacuating air from inside the jig.

4.5. Configuring Parts-Fixing

The proposed parts-fixing algorithm (Algorithm 1) is based on the three requirements (Section 4.1) and assumptions (Section 4.3). Given an assembly sequence, we decide which assembly part to place in which pose. Specifically, the proposed

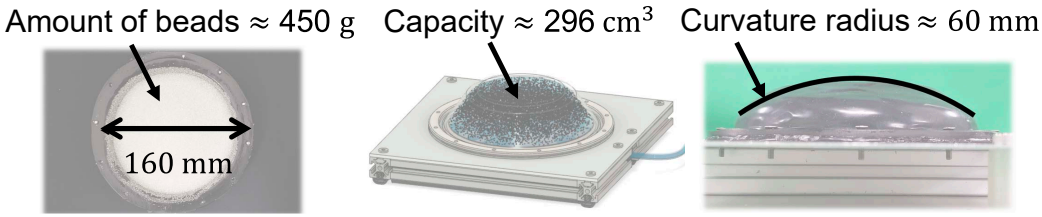


Figure 4.4. Specifications of soft jig.

algorithm selects a fixed part and a fixing posture that allows other objects to reach the fixed part. In addition, the CoG position of the fixed part will be low.

Given an assembly order $\{P_1, P_2, .., P_\eta\}$, we can obtain a list $\hat{\boldsymbol{P}}^*$ of the fixed parts and a list $\hat{\boldsymbol{A}}^*$ of the fixed postures. The process starts from initializing a target part P_t as P_1 and a fixing parts list $\boldsymbol{A}_{det}$, $\hat{\boldsymbol{P}}^*$, and $\hat{\boldsymbol{A}}^*$ as empty lists.

In the main routine, we first calculate a reachable direction list $\boldsymbol{A}(P_i, P_k)$. To achieve interference-free operations, we calculate the reachable direction matrix $\boldsymbol{W}_j$ ($j \in \{+x, -x, +y, -y, +z, -z\}$) with contact matrix $\boldsymbol{C}$ [Bedeoui et al., 2019] and interference-free matrix $\boldsymbol{M}_j$ shown in Figure 4.3(b) and (c). The reachable direction matrix, in which the element in the reachable from j direction is 1, is written as:

$$\boldsymbol{W}_j = (\boldsymbol{C} + \boldsymbol{C}^{\mathrm{T}}) \odot \boldsymbol{M}_j, \tag{4.1}$$

where $\odot$ is Hadamard product of matrices.

Here, the parts are expressed as $P_1, P_2, .., P_\eta$ (where η is the number of parts). The reachable direction list $\boldsymbol{A}(P_i, P_k)$, with regard to the translational displacement is calculated as:

$$(W_{+x}(P_i, P_k), W_{-x}(P_i, P_k), \ldots, W_{-z}(P_i, P_k)). \tag{4.2}$$

If $\boldsymbol{A}(P_i, P_k)$ was a list filled with 0, the process is ended. Otherwise, our method generates a model P_c combined of P_t and P_i. Following process conducts updating P_t by P_c and substituting the determined postures $\boldsymbol{A}(P_i, P_k)$ for $\boldsymbol{A}_{det}$.

If $\boldsymbol{A}_{det}$ includes two or more 1, then we calculate CoG $\boldsymbol{p}_G = [x_G, y_G, z_G]^{\mathrm{T}}$ of the model P_t using CAD. We determine a posture $\hat{A}$ based on the CoG is calculated as:

$$\boldsymbol{p}_G = \frac{\Sigma_{i=0}^{\eta} m_i \boldsymbol{p}_{G,i}}{\Sigma_{i=0}^{\eta} m_i}, \tag{4.3}$$

where m_i and $\boldsymbol{p}_{G,i}$ are the mass and CoG of i-th part.

Algorithm 1 Parts-Fixing Configuration Algorithm

Input: An assembly order $\{P_1, P_2, .., P_\eta\}$
Output: Parts $\hat{\boldsymbol{P}}^*$ to be fixed in certain postures $\hat{\boldsymbol{A}}^*$

```
1: procedure CONFIGURE-FIXING-PARTS
2:     P_t ← P_1
3:     Set A_det, P̂*, and Â* to empty lists
4:     for i = 2, ..., η do
5:         Calculate A(P_t, P_i) using Equation (4.2)
6:         if Elements of A(P_t, P_i) are all 0 then
7:             break
8:         Generate a model P_c combined of P_t and P_i
9:         P_t ← the combined part P_c
10:        A_det ← the determined postures A(P_t, P_i)
11:        if A_det includes two or more 1 then
12:            Calculate CoG of the model P_t using CAD
13:            Determine a posture Â based on the CoG
14:        else
15:            Â ← the first element of A_det
16:        Set P̂ to the bottom part in the posture Â
17:        Add P̂ and Â to P̂* and Â*, respectively
```

If $\boldsymbol{A}_{det}$ does not include two or more 1, we substitute the first element of $\boldsymbol{A}_{det}$ for $\hat{A}$. Then, we set $\hat{P}$ to the bottom part in the posture $\hat{A}$ and add $\hat{P}$ and $\hat{A}$ to $\hat{\boldsymbol{P}}^*$ and $\hat{\boldsymbol{A}}^*$. This one routine is repeated $\eta - 1$ times where η is number of parts.

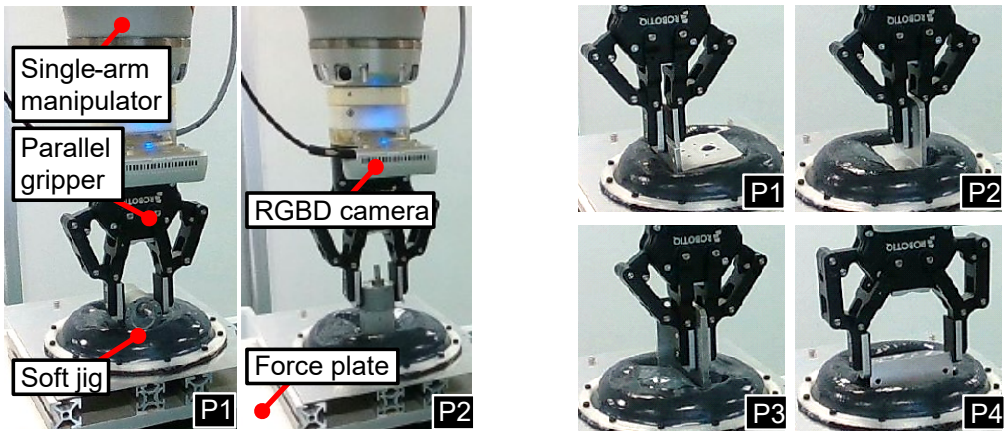
4.6. Experiments

4.6.1. Outline

I used parts shown in Figure 4.3(a), where a motor and a plate are fixed with bolts. The parts were prepared for a belt drive unit used in an assembly challenge [Yokokohji et al., 2019]. Using the parts, I performed the following three experiments.

The first experiment was to experiment the determination of the proposed parts-fixing configuration algorithm. Two cases of assembly sequences were tested in Section 4.6.2.

The second experiment evaluated two fixing abilities: an ability of maintaining the fixed pose and a holding ability against external forces. I conducted parts-placement experiments with (Section 4.6.3) and without (Section 4.6.4) external



(a) Motor (b) Plate

Figure 4.5. Two different postures of the motor and four different postures of the plate evaluated in the expriments of Section 4.6.3.

Table 4.2. Fixing configurations determined from the calculated values of Equation (4.2). The iterations 1 and 2 correspond to the iterations of the loop of the Algorithm 1.

	Iteration 1 ($i = 2$)		Iteration 2 ($i = 3$)	
Case	Equation (4.2)	$\hat{A}$ / $\hat{P}$	Equation (4.2)	$\hat{A}$ / $\hat{P}$
1	$\boldsymbol{A}$(motor, plate)	+z / motor	$\boldsymbol{A}(P_{c1}, \text{bolts})^{*a}$	+z / motor
2	$\boldsymbol{A}$(plate, bolts)	+z / plate	$\boldsymbol{A}(P_{c2}, \text{motor})^{*b}$	-z / motor

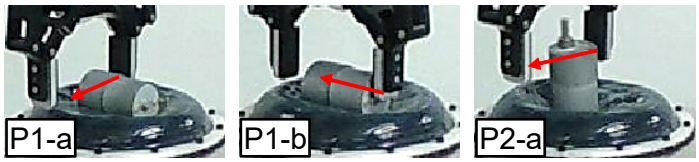
*a P_{c1} represents the combined part of motor and plate
*b P_{c2} represents the combined part of plate and bolts

force application using a manipulator equipped with a parallel-jaw gripper. I evaluated the fixing ability based on holding forces, moving distances, and success rates.

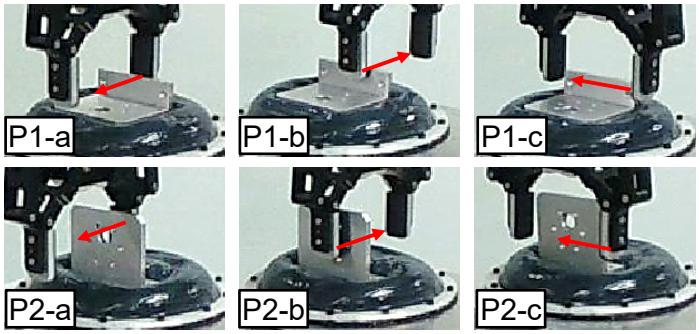
The third experiment (Section 4.6.5) involved verifying feasibility of assembly operations (Figure 4.2) with a robot arm equipped with a parallel-jaw gripper.

4.6.2. Determining Fixed Parts and Their Pose

Table 4.2 shows the results of the parts-fixing configuration for two cases of assembly sequences. Case 1 is {motor, plate, bolts}. Case 2 is {plate, bolts, motor}. The assembly of small bolts was not selected in the algorithm because if the bolts are positioned under the plate, the CoG position of the model combined the plate



(a) Motor



(b) Plate

Figure 4.6. Different placement postures (P1 and P2) and pushing directions (a, b, and c) for the motor and plate evaluated in the expriments of Section 4.6.4.

and bolts are higher than the upside-down posture.

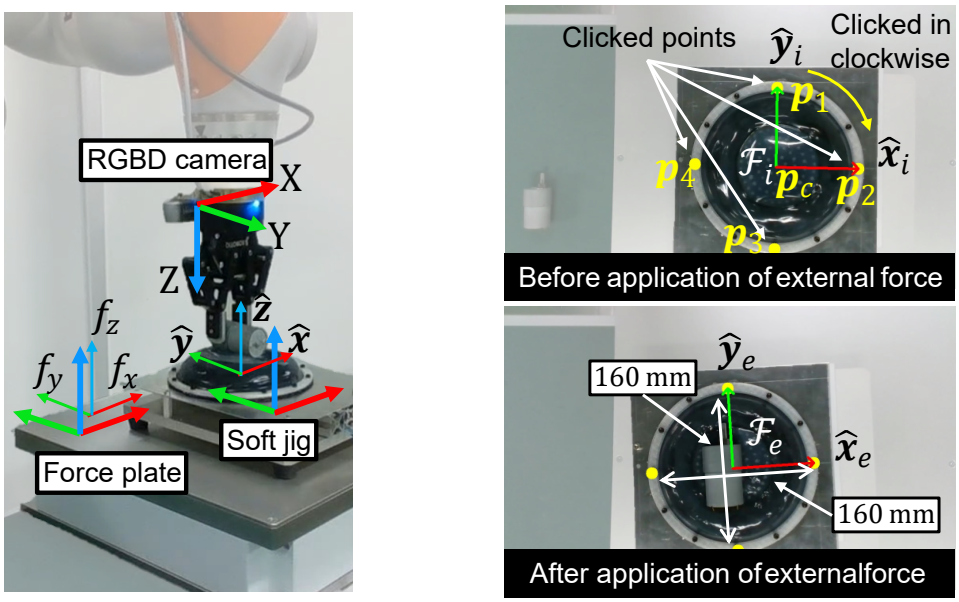
In Case 1, the parts-fixing configuration determined by Algorithm 1 is the motor placed in the posture as shown in Step 1 of Figure 4.2. This posture is difficult to achieve in the base plate of a metal jig because pins are attached at the bottom of the motor as shown on the top right picture in Step 1 of Figure 4.2. With the soft jig, this posture was achievable. The plate is inserted onto the fixed motor, then the blots are screwed onto the plate fixed with the motor.

In Case 2, firstly, the bolts are placed onto the fixed plate in the posture P4 shown in Figure 4.5(b). Secondaly, the plate with bolts are inserted onto the fixed motor.

4.6.3. Evaluating Versatility to Fixing-Posture

To evaluate the versatility of the soft jig for the shape and posture of the parts, I investigated whether the placement postures were maintained after released from the gripper.

In addition to the determined posture in the previous section, postures for comparison were shown in Figure 4.5. In the case of motor, the shape is axisymmetric, so there are two ways to place it in an axis-aligned manner. The two postures are with the side (P1) or bottom surface (P2) of the cylinder shape being in contact with the jig.



(a) Condition

(b) Displacement

Figure 4.7. Experiments to evaluate parts-fixing performance. I used a force plate to measure normal and shearing forces. I calculated the displacement of the jig using the manually clicked points on both images of before and after application of the external forces.

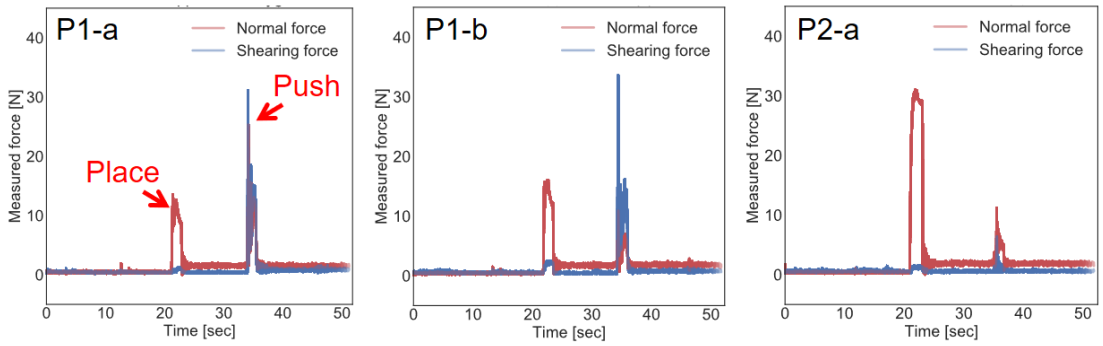
Four different postures of the plate were prepared. These include the postures of the back or front side of the insertion hole facing straight up (Figure 4.5(b) P1 or P4). I further prepared the postures that the bottom (P2) or side surface (P3) is contacted to the jig surface.

The gripper's trajectory and grasping configurations were manually generated. The trial was regarded as successful if the parts were upright, even if the gripper released the part, *i.e.,* if the resting state was possible. I tried ten trials and checked whether each trial was success or failure by eyes. The success rates were 100% (= 10/10) for all postures of the two parts, thus the versatility of soft jig against the placed shapes is high.

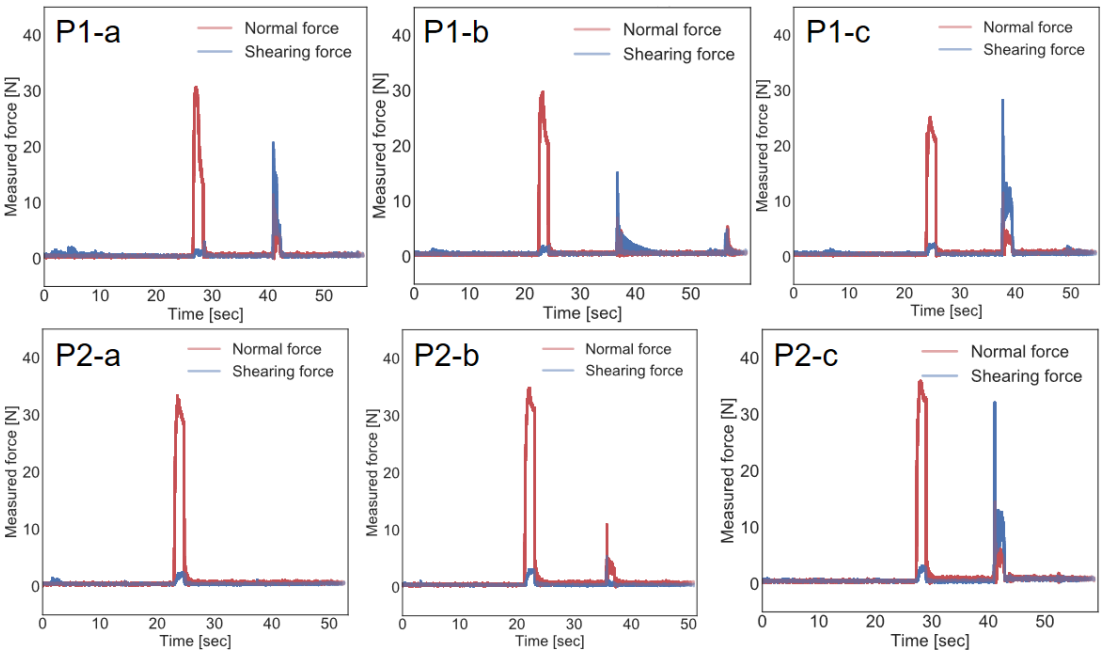
4.6.4. Evaluating Parts-Fixing Against External Force

The fixing performance is evaluated based on holding forces, moving distances, and success rates. The first is an evaluation of the fixing performance based on the holding force when an external force is applied; the higher the performance, the higher the holding force should be.

The second is an evaluation based on moving distances when an external force

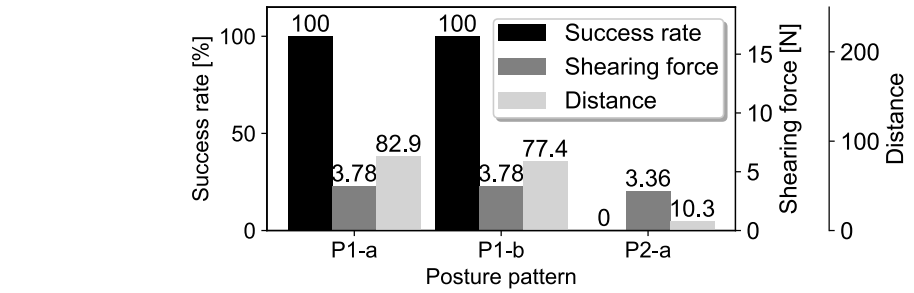


(a) Motor

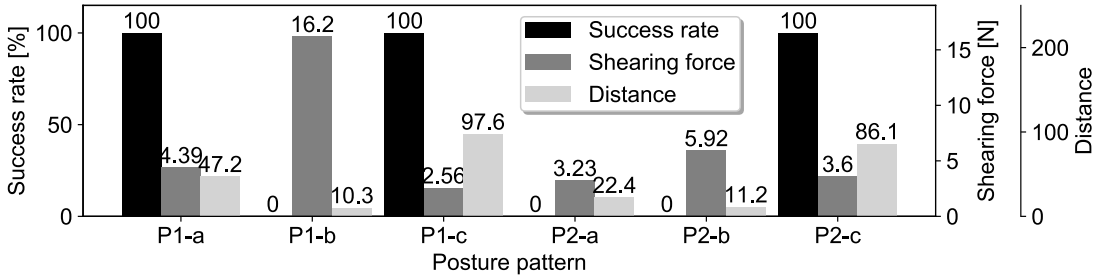


(b) Plate

Figure 4.8. Normal and shearing forces applied under the soft jig during placing and pushing operations performed around the two peaks.



(a) Motor



(b) Plate

Figure 4.9. Performance of fixing the motor and plate. Each figure shows the success rate (number of successes in five trials), the average of the maximum value of shearing force, and the average distance of the soft jig before and after the external force application in five trials.

is applied. I defined the fixing success based on the distance of the jig base itself, which is not fixed anywhere. The jig was moved in response to the pushing motion of 70 mm straight-line trajectory to apply the external force. If the part posture is not changed againt the pushing motion, all the force should be converted into the jig movement, so the amount of movement is larger. If the posture is changed, that amount of the jig movement must be low. Therefore, if the part is firmly fixed on the jig, it should move as much as the distance pushed by a robot. But considering the elasticity of the silicon membrane, I defined successful fixtuation as the distance more than 63 mm (90% of the 70 mm pushing trajectory).

Holding force

Figure 4.7(a) shows the experimental setup including the robot arm with a gripper to apply the external force. I set a force plate and an RGBD camera to measure the holding force and the jig movement. I measured the forces applied to the lower part of the jig when the fixed parts were pushed by the straight-line trajectory of the gripper. I also measured how much the soft jig moved before and after

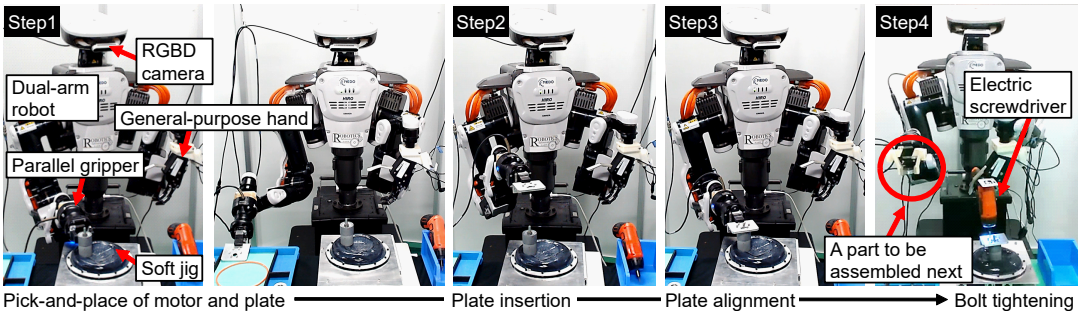


Figure 4.10. Assembly sequence with the soft jig and a robot. The assembly order was the same as the manual assembly shown in Figure 4.2.

the external force application. Figure 4.6 shows the postures of the parts in the experiments to evaluate fixing performance.

Here, the resulting force on the contact surface was calculated as the magnitude of normal force F_n and shearing force F_s by the following equations:

$$F_n = |f_z|, \quad F_s = \sqrt{f_x^2 + f_y^2}, \tag{4.4}$$

where f_x, f_y, and f_z are the measured forces in the coordinate system of the force plate shown in Figure 4.7(a). Figure 4.8 shows the calculated values of the normal and shearing forces during the operations including placing and pushing of the parts on the jig. The graph IDs on the top left on each graph correspond to the IDs in Figure 4.6. The two peaks on all graphs except P2-a of the plate show the force values at the timing of placement and pushing, respectively. In P2-a of the plate, the plate fell on the jig at the timing when the gripper made contact with the plate in the pushing operation. Thus, only one peak exists because the forces by pushing could not be measured.

Moving distance

The displacements of the jig after the application of external force were also measured. The hand-eye camera (Intel Corporation, RealSense D435) recorded RGBD images before and after applying the external force to the part fixed on the jig in contact with the force plate.

I used two images shown in Figure 4.7(b) before and after applying the external force. I calculated the distance $d(\mathcal{F}_i, \mathcal{F}_e)$ between configuration frames of the soft jig before $\mathcal{F}_i$ and after $\mathcal{F}_e$ applying the external force as:

$$d(\mathcal{F}_i, \mathcal{F}_e) := \sqrt{d(\hat{x}_i, \hat{x}_e)^2 + d(\hat{y}_i, \hat{y}_e)^2}. \tag{4.5}$$

Equation (4.5) is a metric used to calculate the distance between two configuration frames proposed in [Ahuactzin and Gupta, 1999]. I calculated the poses $(\hat{x}_i, \hat{y}_i)$ of $\mathcal{F}_i$ and the poses $(\hat{x}_e, \hat{y}_e)$ of $\mathcal{F}_e$ in [pixel] using four points $\boldsymbol{p}_k (k = 1, 2, 3, 4) \in \mathbb{R}^2$, as shown in Figure 4.7(b). I clicked on the four screws of the jigs in the images as $\boldsymbol{p}_k$. Each screw has a central angle of 90° and fixes the membrane onto the jig base. I converted the unit of the distance from pixels to mm based on the known width 160 mm of the jig. $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ are calculated as:

$$(\hat{\boldsymbol{x}},\ \hat{\boldsymbol{y}}) = (\boldsymbol{p}_2 - \boldsymbol{p}_c,\ \boldsymbol{p}_1 - \boldsymbol{p}_c), \tag{4.6}$$

$$\boldsymbol{p}_c = \frac{\boldsymbol{p}_1 + \boldsymbol{p}_2 + \boldsymbol{p}_3 + \boldsymbol{p}_4}{4}. \tag{4.7}$$

$\hat{\boldsymbol{z}}$ is set to $\mathbf{0}$ because the images shown in Figure 4.7(b) was captured from directly above, then the fixed surface of the jig remained horizontal even after the external force application.

Factors underlying successful fixing

Figure 4.9 shows average maximum values of shearing force in five trials. Figure 4.9 shows the success rates in the five trials. The success rate was 0% for P2-a of Motor, P1-b of Plate, P2-a and P2-b. In these cases, the first peak of the normal force shown in Figure 4.6 indicates that the pushing force is applied to the same extent as in other cases. However, since the maximum force at the second peak of the shearing force is lower than other cases, suggesting the fixing has failed.

Figure 4.9 shows the calculated values of $d(\mathcal{F}_i, \mathcal{F}_e)$. The failure cases resulted in a low amount of displacement compared to the successful ones. The difference between the mean displacements of the failure and success cases is 24.0 mm ($= 78.2 - 54.2$), and the value of the success cases is significantly larger than that of the failure cases. 78.2 mm was larger than the original pushing distance 70.0 mm because of over-displacement due to an acceleration of the pushing motion. The soft jig is hardened by the jamming transition, thus the deformation itself does not cause of the small resultant displacement.

In the case of low-height postures such as P1-a and P1-b of the motor and P1-a and P1-c of the plate, as far as I confirmed with our eyes, the displacement did not occur despite the direct external force. Thus, the success rates of the four cases were 100%. The posture P2-c of the plate was firmly fixed, although it was a high-height posture. This is because the pushing action of the external force pushes the fixed object into the inside of the jig as it tries to rotate on the axis perpendicular to the pushed direction.

Against the external forces, to fix high-height postures P2-a of both parts, high

datum planes surrounding the side surfaces of the parts are required as shown in Fig. 1 (d). Since such the large datum plane was not generated, the trials were unsuccessful. The results suggest that selecting the placement posture and forming the datum plane are important.

4.6.5. Feasibility of Assembly Operations for Fixed Parts

In this section, I confirm the force generated during the actual assembly operations expanding the experiments in the previous section. The assembly operation by the dual-arm robot was executed on the procedure of Figure 4.2. Dual arms equipped two types of grippers: a parallel-jaw gripper used for grasping assembly parts and a general-purpose gripper used for grasping a tool such as the electric driver. As shown in Figure 4.10, the operation was divided into four steps. I used two arms to avoid regrasping a tool. All operations were performed with hand-crafted trajectories of one hand.

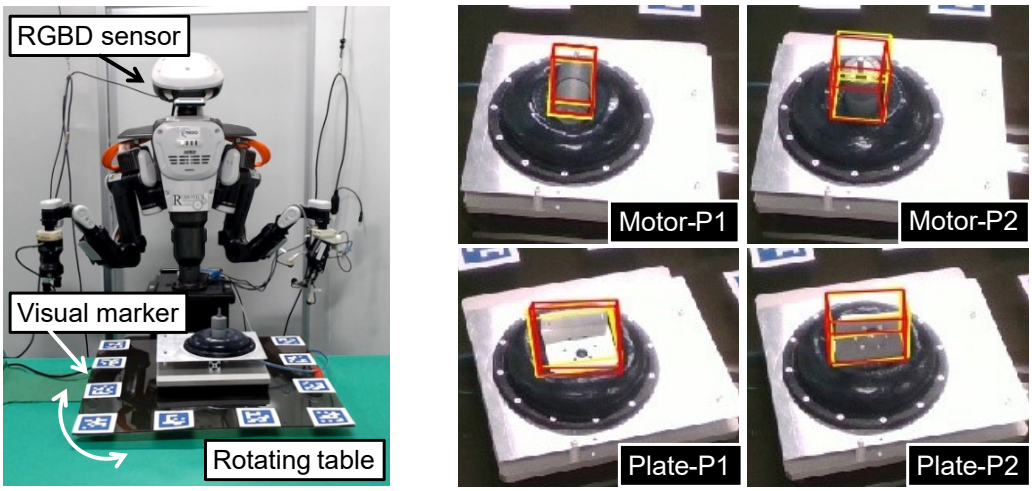
Thorough the experiments, given the assembly sequence and the gripper's trajectories, the robot could execute pick-and-place, insertion, and tightening of parts using the soft jig. The gripper did not apply external force directly to fixed parts; instead the external force was applied via the grasped plate during insertion of the plate into the jig-fixed motor or by the grasped electric driver during screwing of the bolts. Even under such the external forces, the displacement of the motor on the jig was not significant, suggesting that it may be useful for fixing a part during assembly operations.

4.7. Discussion

To show a durability of the soft jig toward an industrial application, I discuss the parts pose estimation. To design the concrete method is out-of-scope, but I discuss the possibility and future issues. Since the precise parts-positioning with the soft jig is difficult than using a metal jig, we need to remove the uncertainty of the part pose by a pose estimation method.

To confirm the 6D pose estimation task for fixed parts, I apply *PVNet* [Peng et al., 2019], one of deep learning-based algorithms [Tekin et al., 2018; Tremblay et al., 2018]. To train the network, I leverage a quick dataset collection method using visual markers proposed in [Hinterstoisser et al., 2012; Kiyokawa et al., 2019a,b], as shown in Figure 4.11(a). These methods reduce human effort for training and enable to use PVNet for the high-mix low-volume production. Both in training and testing, I use a head-mounted camera of the robot.

Figure 4.11 (b) shows the results of the pose estimation applied to test images showing different poses of the motor and the plate. The test images are not



(a) System (b) Visualization of estimation results

Figure 4.11. Dataset collection system and results of 6D pose estimation of the fixed parts.

used for training. The yellow 3D bounding boxes in the images show the box calculated based on the visual markers, which is the grund truth. The red boxes with similar shapes to the yellow boxes show the 6D poses estimated by the trained model. The results suggest that the pose in a wide viewing angle can be accurately tracked and the appearance of the soft jig does not deteriorate the pose estimation. To insert a part to the motor shaft, the system needs to determine the insertion motions based on the shaft pose.

Chapter 5

Discussion

5.1. Integration Potential

This section discusses several methods that combines each of the methods proposed for the basic functions of the robotic assembly system in terms of recognition, planning, and execution.

For example, the assembly using a soft jig cannot be performed accurately unless the parts on the jig are correctly recognized. In this case, the proposed method using the visual markers and the rotating table can perform automatic data collection to train the vision system that can recognize the parts on the soft jig.

Regarding the combination of recognition and planning, it is necessary to correctly recognize success or failure of the operatons in the planned assembly sequence. Therefore, we may build a visual system that recognizes whether the constraint relationships to be satisfied between parts are met as per the planned assembly order. This vision system can be reconstructed quickly by automatically generating the assembly sequence using CAD and automatically generating the image dataset using our system.

5.2. Agility in Reconfiguration

This section discusses the comparative evaluation of the agility. The agility of reconfiguration of robotic assembly system was evaluated previously in simulation [Downs et al., 2016] but there are no metrics defined for a real robotic system. Therefore, I evaluated the real robotic system in terms of the versatility of the manipulation system and the time to train the vision system. The ultimate goal of this research is to focus on improving precision, versatility, and agility. Therefore,

Table 5.1. Comparison of the methods in terms of agility, versatility, and precision.

Method		Agility	Versatility	Precision	Time [h]
Recognition	Real-world	+		+	7.3
	CG	++		−	0.45
Planning	CAD	+		+	6.3
	Instruction	++		−	< 6.3
Execution	Soft jig	++	+	−	< 1.0
	Tool changer	++	−	+	1.0

the comparisons of agility, versatility, and precision are summarized in Table 5.1, and those with known actual time are shown together.

First, I compare the automatic real-world dataset generation with the CG-based dataset generation. CG-based method is faster, but in the object recognition results evaluated in this thesis, the accuracy is a little lower when no real-world data is used. On the other hand, the proposed automatic generation method significantly improves the collection time compared to the conventional manual annotation. This is the result of 500 images collection and annotation data generation.

Second, I compare the CAD-based sequence generation with the manual sequence generation. Description-based sequence generation is faster when not running optimization calculations based on the parts geometries. CAD-based sequence generation is less prone to human errors and variations in the format of the design document.

Finally, I compare the soft jig with the tool changer. In the case of the tool changer, it takes time and preparation to determine multiple hands and jigs used for each operation, and it also takes time to reposition them before execution. On the other hand, in the case of the soft jig, in addition to changing the jig to another jig with a different scale according to the size of the target part, it may only be necessary to reposition the jig considering the range of motion of the robot's fingers, and this can be done in less than an hour.

5.3. Remaining Issues

The achievements and remaining issues are summarized in Table 5.2 and Table 5.3. HC in the table stands for hard-coded, which is the part programmed by human. These are the parts that should be constructed in future works or based on other studies.

Table 5.2. Functions implemented for the real-time execution. The table indicates whether each function is a general-purpose type or a specialized type.

Basic configuration	General-purpose	Specialized
Object perception	✓	
Task perception		✓ (HC)
Task planning	✓	
Manipulation planning	✓	
Grasp planning		✓ (HC)
Trajectory control	✓	
Grasp control		✓ (HC)

Table 5.3. Functions implemented for the system reconfiguration. The table indicates whether each function is automatic or manual.

Configuration for reconfiguration	Automatic	Manual
Dataset generation	✓	
Model training	✓	
Replanning sequence	✓	
Learning strategies		✓ (HC)
Reconfiguring hardwares		✓ (HC)
Learning skills		✓ (HC)
Parameter adjustment		✓ (HC)

Chapter 6

Conclusion

This dissertation aims to identify how to agilely reconfigure the system to handle the frequent introduction of new products.

6.1. Contributions

Here, I summarize the major contributions of this dissertation

- Fully automated annotation methods with visual markers that were diminished later is effective for generating annotated images rapidly. To obtain the real-world image datasets, unbiased dataset collection was proposed and evaluated. Automatic image dataset collection method with a small robotic arm and a rotating stage enabled us to collect multi-view object images more quickly. A domain adaptation method with several image processing techniques were evaluated by training conventional deep learning-based object detection methods.

- CAD-based assembly planning methods for searching feasible assembly sequence without interferences of parts, satisfying insertion relationships, and with low difficulty of constraint state transition. I designed the fitness functions for genetic algorithms for the heuristic search. To achieve calculating the fitness based on the geometries extracted from the CAD model, automatic extraction methods of part information for not only rigid object but also the deformable objects are proposed and evaluated using a product including many parts and a deformable rubber band.

- A state-of-the-art flexible part fixing device named soft jig was proposed. The usability and flexibility of the soft jig was evaluated, in terms of the

capablility of fixing assembly parts with robotic arms and assembly parts. The fixing capability based on jamming transition was clarifiend on the experiments of object placement and the application of external force.

6.2. Future Directions

This dissertation mainly discuss agile reconfiguration methods related to task planning, manipulation planning and grasp planning. The more challenging future issues are related to following two aspects.

How could the assembly system be adapted to more drastic change of products? In order to quickly respond to the changes of the target product not only parts, we need to discuss several remaining issues. Unlike in the case of the part changes, semantics related to assembly operations and assembly parts [Savarimuthu et al., 2018; Shiraki et al., 2014] can be obtained from the assembly operator or videos with *Learning from Demonstration* (LfD) or a similar approach. Although there are many studies on human imitation learning for assembly tasks [Zhu and Hu, 2018], there are no successful examples using actual mechanical parts. As our future work, I will develop a system to clarify the possibility of agile reconfiguration for product changes.

Over the past four decades, there are numerous research articles on autonomous robotic assembly systems [Kyrarini et al., 2019] based on robot-robot collaboration [Argall et al., 2009; Maeda et al., 2007; Marvel et al., 2018; Zhu and Hu, 2018] and human-robot collaboration [Krüger et al., 2009; Raessa et al., 2020; Tsarouchi et al., 2017; Weckenborg et al., 2020]. To achieve the agile reconfiguration, several modular reconfigurable systems [Heilala and Voho, 2001; Tsukune et al., 1993] were proposed. Furthermore, planning methods for the reconfiguration of each system have been proposed, such as cell layout planning [Laemmle and Gust, 2019; Zhang and Fang, 2017], scheduling of the multi-robot assembly cell [Glibert et al., 1990] and relocating multiple robots [Arai et al., 2000; Maeda et al., 2003; Makris et al., 2012].

In the research [Krüger et al., 2009], they mention the flexibility and changeability of assembly processes require a close cooperation between the worker and the assembly robot. They also conclude that the interaction between humans and robots improves the efficiency of individual complex assembly processes, particularly when a robot serves as an intelligent assistant.

There are many studies on the framework in which robots learn from humans through two-way or one-way communication. Many research conducted Learning from Observation/Demonstration [Atkeson and Schaal, 1997; Ikeuchi et al., 2018; Nakaoka et al., 2007; Pastor et al., 2009; Savarimuthu et al., 2018; Schaal, 1996]

(Programming by Demonstration [Billard et al., 2008] and Learning by Watching [Kuniyoshi et al., 1994]). The learning from demonstration methods have been applied to assembly tasks to obtain the skillful motions for the robots from humans [Hamaya et al., 2014; Wang et al., 2020a]. Generating robotic assembly tasks from human demonstrations [Ding et al., 2020] and generating collaborative assembly tasks [Malik and Bilberg, 2019] have been much attentioned.

In our future work, to achieve high-level planning quickly if product has changed, I will explore a combined system such as the human-robot collaboration system that can learn assembly tasks from natural demonstrations by a human not only the instructions and object data.

What technologies are needed to achieve a more robust assembly system in uncertain environments? The key issues to keep the robustness even for such frequent product changes are realtime assembly error recorvery based on semantic understanding of the changes, and improving the reconfigurability in the hardware structure for robotic assembly systems.

To recover from errors of the robot system during assembly operations, we need a system that re-executes the operations based on the results of the operations. An approach is to construct a robot system that can deal with failures by extracting the constraint direction between parts and executing a predefined recovery action for the constraint direction that cannot be satisfied. Error recovery methods in assembly operations are an important unsolved problem [Fujita et al., 2018] in achieving a general-purpose assembly robotic system, where it was found that no methods have been established in the assembly challenge of WRS'18.

Aforementioned error recovery methods online are required but what we need more is a framework for an error-less assembly system. Creating a way to deal with each error is a naive approach, so it unnecessarily increases the complexity of the system. Rather, the direction of improving the performance of each module that can be easily controlled and reconfigured, such as the soft jig proposed in this dissertation, reduces the complexity of the system and makes it difficult for human errors and artificial bugs to occur. A general parts feeder [Domae et al., 2020] and a mechanical tool to extend the general grippers [Hu et al., 2019] are also interesting approach in this direction.

In terms of the hardware reconfiguration, a challenging but siginificantly effective approach is to achieve not modular but self-reconfigurable (self-assembly or self-repair) robots [Murooka et al., 2019] for autonomous assembly system. Many conventional self-reconfigurable robots are easily reconstructed by making the hardware itself modular [Liu et al., 2016a; Yim et al., 2007; Yoshida et al., 2002]. The modular systems are tend to be complex structures. However, exploring the reconfigurability of general-purpose robot hands and gripper-mounted

robot arms are expected in industry because it may not require major changes to the currently deployed system.

References

Smart factory applications in discrete manufacturing. white paper, industrial internet consortium, 2017.

B. Adhikari, J. Peltomaki, J. Puura, and H. Huttunen. Faster bounding box annotation for object detection in indoor scenes. In *European Workshop on Visual Information Processing (EUVIP)*, pages 1–6, 2018.

D. Agarwal, T. T. Robinson, and C. G. Armstrong. A CAD based framework for optimizing performance while ensuring assembly fit. *Recent Advances in Intelligent Manufacturing*, 923:73–83, 2018.

M. Agrawala, D. Phan, J. Heiser, J. Haymaker, J. Klingner, P. Hanrahan, and B. Tversky. Designing effective step-by-step assembly instructions. *ACM Transactions on Graphics*, 22:828–837, 2003.

J.-M. Ahuactzin and K. Gupta. The kinematic roadmap: A motion planning based global approach for inverse kinematics of redundant robots. *IEEE Transactions on Robotics and Automation*, 15(4):653–669, 1999.

S. Akizuki and M. Hashimoto. Semi-automatic training data generation for semantic segmentation using 6DoF pose estimation. In *VISAPP*, pages 607–613, 2019.

A. Alspach, K. Hashimoto, N. Kuppuswamy, and R. Tedrake. Soft-bubble: A highly compliant dense geometry tactile sensor for robot manipulation. In *IEEE International Conference on Soft Robotics (Robosoft)*, pages 597–604, 2019.

J. R. Amend, E. Brown, N. Rodenberg, H. M. Jaeger, and H. Lipson. A positive pressure universal gripper based on the jamming of granular material. *IEEE Transactions on Robotics*, 28(2):341–350, 2012.

T. Arai, Y. Aiyama, Y. Maeda, M. Sugi, and J. Ota. Agile assembly system by "Plug and Produce". *CIRP Annals*, 49(1):1–4, 2000.

B. D. Argall, S. Chernova, M. Veloso, and B. Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483, 2009.

C. G. Atkeson and S. Schaal. Learning tasks from a single demonstration. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1706–1712, 1997.

B. Babic, N. Nesic, and Z. Miljkovic. A review of automated feature recognition with rule-based pattern recognition. *Computers in Industry*, 59(4):321–337, 2008.

M. V. A. R. Bahubalendruni and B. B. Biswal. A review on assembly sequence generation and its automation. *Proc. of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 230(5):824–838, 2015.

M. V. A. R. Bahubalendruni and B. B. Biswal. Liaison concatenation - a method to obtain feasible assembly sequences from 3D-CAD product. *Sadhana*, 41(1): 67–74, 2016.

M. V. A. R. Bahubalendruni, B. B. Biswal, and V. Upadhyay. Assembly sequence generation and automation. In *International Conference on Design, Manufacturing and Mechatronics*, pages 185–192, 2014.

M. V. A. R. Bahubalendruni, B. B. Biswal, and B. B. V. L. Deepak. Computer aided assembly attributes retrieval methods for automated assembly sequence generation. *World Academy of Science, Engineering and Technology International Journal of Mechanical and Mechatronics Engineering*, 11(4):759–767, 2017.

M. V. A. R. Bahubalendruni, A. Gulivindala, M. Kumar, B. B. Biswal, and L. N. Annepu. A hybrid conjugated method for assembly sequence generation and explode view generation. *Assembly Automation*, 39(1):211–225, 2019.

A. Bedeoui, R. Ben Hadj, M. Hammadi, M. Trigui, and N. Aifaoui. Assembly sequence plan generation of heavy machines based on the stability criterion. *The International Journal of Advanced Manufacturing Technology*, 102:2745–2755, 2019.

R. Benenson, S. Popov, and V. Ferrari. Large-scale interactive object segmentation with human annotators. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11700–11709, 2019.

J. C. Berry, N. Fahlgren, A. P. Pokorny, R. Bart, and K. M. Veley. An automated, high-throughput method for standardizing image color profiles to improve image-based plant phenotyping. *PeerJ*, 6:e5727, 2018.

A. Bhattacharyya. On a measure of divergence between two statistical populations defined by probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35:99–109, 1943.

Z. M. Bi and W. J. Zhang. Flexible fixture design and automation: Review, issues and future directions. *International Journal of Production Research*, 39 (13):2867–2894, 2001.

A. Billard, S. Calinon, R. Dillmann, and S. Schaal. *Handbook of robotics, Survey: Robot programming by demonstration*, chapter 59, pages 201–213. MIT Press, 2008.

E. Brachmann, F. Michel, A. Krull, M. Y. Yang, S. Gumhold, and C. Rother. Uncertainty-driven 6D pose estimation of objects and scenes from a single RGB image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3364–3372, 2016.

E. Brown, N. Rodenberg, J. Amend, A. Mozeika, E. Steltz, M. R. Zakin, H. Lipson, and H. M. Jaeger. Universal robotic gripper based on the jamming of granular material. *PNAS*, 107(44):18809–18814, 2010.

Z. Cai, J. Han, L. Liu, and L. Shao. RGB-D datasets using microsoft kinect or similar sensors: a survey. *Multimedia Tools and Applications*, 76(3):4313–4355, 2017.

B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar. The YCB object and model set: Towards common benchmarks for manipulation research. In *ICAR*, pages 510–517, 2015.

S.-F. Chen and Y.-J. Liu. An adaptive genetic assembly-sequence planner. *International Journal of Computer Integrated Manufacturing*, 14(5):489–500, 2001.

S. Choi, Q.-Y. Zhou, and V. Koltun. Robust reconstruction of indoor scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5556–5565, 2015.

Y.-K. Choi, D. M. Lee, and Y. B. Cho. An approach to multi-criteria assembly sequence planning using genetic algorithms. *International Journal of Advanced Manufacturing Technology*, 42:180–188, 2009.

S. Chopra, M. T. Tolley, and N. Gravish. Granular jamming feet enable improved foot-ground interactions for robot mobility on deformable ground. *IEEE Robotics and Automation Letters (RA-L)*, 5(3):3975–3981, 2020.

C. A. C. Coello, D. A. V. Veldhuizen, and G. B. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, New York, 2002.

M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016.

R. J. Costa, F. Silva, and R. D. Campilho. A novel concept of agile assembly machine for sets applied in the automotive industry. *The International Journal of Advanced Manufacturing Technology*, 91:4043–4054, 2017.

E. D. Cubuk, D. M. B. Zoph, V. Vasudevan, and Q. V. Le. AutoAugment: Learning augmentation strategies from data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 113–123, 2019.

S. K. Das and A. K. Swain. Classification, representation and automatic extraction of adhesively bonded assembly features. *Assembly Automation*, 39(4): 607–623, 2019.

D. De Gregorio, A. Tonioni, G. Palli, and L. Di Stefano. Semiautomatic labeling for deep learning in robotics. *IEEE Transactions on Automation Science and Engineering*, 17(2):611–620, 2020.

K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

B. B. V. L. Deepak, G. B. Murali, M. V. A. R. Bahubalendruni, and B. B. Biswal. Assembly sequence planning using soft computing methods: A review. *Proc. of the Institution of Mechanical Engineers, Part E: Journal of Process Mechanical Engineering*, 233(3):653–683, 2019.

J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

G. Ding, Y. Liu, X. Zang, X. Zhang, G. Liu, and J. Zhao. A task-learning strategy for robotic assembly tasks from human demonstrations. *Sensors*, 20(19), 2020.

T.-T. Do, M. Cai, T. Pham, and I. Reid. Deep-6DPose: Recovering 6D object pose from a single RGB image. *CoRR*, abs/1802.10367, 2018.

Y. Domae, A. Noda, T. Nagatani, and W. Wan. Robotic general parts feeder: Bin-picking, regrasping, and kitting. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5004–5010, 2020.

A. Downs, W. Harrison, and C. Schlenoff. Test methods for robot agility in manufacturing. *Industrial robot*, 43(5):563–572, 2016.

A. Eltaief, B. Louhichi, S. Remy, and B. Eynard. A CAD assembly management model:mates reconciliation and change propagation. In *International Conference on Design and Modeling of Mechanical Systems*, pages 459–471, 2017.

M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, and J. Winn. The PASCAL visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, 2015.

M. Fathianathan, A. S. Kumar, and A. Y. C. Nee. An adaptive machining fixture design system for automatically dealing with design changes. *ASME Journal of Computing and Information Science in Engineering*, 7(3):259–268, 2007.

G. D. Finlayson, S. D. Hordley, Cheng Lu, and M. S. Drew. On the removal of shadows from images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):59–68, 2006.

M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

A. W. Fitzgibbon and R. B. Fisher. A buyer's guide to conic fitting. In *British Machine Vision Conference (BMVC)*, pages 513–522, 1995.

M. Fujita, S. Ikeda, T. Fujimoto, T. Shimizu, S. Ikemoto, and T. Miyamoto. Development of universal vacuum gripper for wall-climbing robot. *Advanced Robotics*, 32(6):283–296, 2018.

K. Fukuda, I. G. Ramirez-Alpizar, N. Yamanobe, D. Petit, K. Nagata, and K. Harada. Recognition of assembly tasks based on the actions associated to the manipulated objects. In *IEEE/SICE International Symposium on System Integration (SII)*, pages 193–198, 2019.

S. Garrido-Jurado, R. M. noz Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and R. Medina-Carnicer. Generation of fiducial marker dictionaries using mixed integer linear programming. *Pattern Recognition*, 51:481–491, 2016.

T. Gašpar, B. Ridge, R. Bevec, M. Bem, I. Kovač, and A. Ude. Rapid hardware and software reconfiguration in a robotic workcell. In *International Conference on Advanced Robotics (ICAR)*, pages 229–236, 2017.

G. Georgakis, A. Mousavian, A. C. Berg, and J. Košecká. Synthesizing training data for object detection in indoor scenes. In *Robotics: Science and Systems (RSS)*, 2017.

T. Germer, T. Uelwer, S. Conrad, and S. Harmeling. PyMatting: A python library for alpha matting. *Journal of Open Source Software*, 5(54):2481, 2020.

S. Ghandi and E. Masehian. Assembly sequence planning of rigid and flexible parts. *Journal of Manufacturing Systems*, 36:128–146, 2015.

R. Girshick. Fast R-CNN. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

P. R. Glibert, D. Coupez, Y. M. Peng, and A. Delchambre. Scheduling of a multi-robot assembly cell. *Computer Integrated Manufacturing Systems*, 3(4): 236–245, 1990.

M. Goldwasser and R. Motwani. Complexity measures for assembly sequences. *International Journal of Computational Geometry & Applications*, 9(4-5):371–417, 1999.

H. Gong, G. D. Finlayson, and R. B. Fisher. Recoding color transfer as a color homography. In *British Machine Vision Conference (BMVC)*, pages 17.1–17.11, 2016.

R. C. Gonzalez and R. E. Woods. *Digital Image Processing 2nd Edition*, chapter 3, pages 94–102. Prentice Hall, 2001.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 2672–2680, 2014.

R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 999–1006, 2011.

G. Gorjup, G. Gao, A. Dwivedi, and M. Liarokapis. Combining compliance control, cad based localization, and a multi-modal gripper for rapid and robust programming of assembly tasks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9064–9071, 2020.

P. M. Grippo, M. V. Gandhi, and B. S. Thompson. The computer-aided design of modular fixturing systems. *International Journal of Advanced Manufacturing Technology*, 2(2):75–88, 1987.

A. K. Gulivindala, M. V. A. R. Bahubalendruni, S. S. V. P. Varupala, and K. Sankaranarayanasamy. A heuristic method with a novel stability concept to perform parallel assembly sequence planning by subassembly detection. *Assembly Automation*, 40(5):779–787, 2020.

A. Gunasekaran. Agile manufacturing: A framework for research and development. *International Journal of Production Economics*, 62(1):87–105, 1999.

A. Gunasekaran, Y. Y. Yusuf, E. O. Adeleye, T. Papadopoulos, D. Kovvuri, and D. G. Geyi. Agile manufacturing: an evolutionary review of practices. *International Journal of Production Research*, 57(15-16):5154–5174, 2019.

I. Gödri, C. Kardos, A. Pfeiffer, and J. Váncza. Data analytics-based decision support workflow for high-mix low-volume production systems. *CIRP Annals*, 68(1):471–474, 2019.

M. Hamaya, F. von Drigalski, T. Matsubara, K. Tanaka, R. Lee, C. Nakashima, Y. Shibata, and Y. Ijiri. Learning soft robotic assembly strategies from successful and failed demonstrations. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 928–934, 2014.

M. Hamaya, R. Lee, K. Tanaka, F. von Drigalski, C. Nakashima, Y. Shibata, and Y. Ijiri. Learning robotic assembly tasks with lower dimensional systems by leveraging physical softness and environmental constraints. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 7747–7753, 2020a.

M. Hamaya, F. von Drigalski, T. Matsubara, K. Tanaka, R. Lee, C. Nakashima, Y. Shibata, and Y. Ijiri. Learning soft robotic assembly strategies from successful and failed demonstrations. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8309–8315, 2020b.

Hamidullah, E. Bohez, and M. A. Irfan. Assembly features: Definition, classification, and instantiation. In *International Conference on Emerging Technologies*, pages 617–623, 2006.

K. Hara, K. Nishino, and K. Ikeuchi. Multiple light sources and reflectance property estimation based on a mixture of spherical distributions. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1627–1634, 2005.

S. Hara and K. Azuma. Cell production system for assembly. *Robotics and Computer-Integrated Manufacturing*, 4(3-4):379–385, 1988.

K. Harada, T. Tsuji, and J. Laumond. A manipulation motion planner for dual-arm industrial manipulators. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 928–934, 2014.

B. Hasan and J. Wikander. Features extraction from cad as a basis for assembly process planning. In *Technological Innovation for Smart Systems*, pages 144–153, 2017.

K. He, J. Sun, and X. Tang. Fast matting using large kernel matting laplacian matrices. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2165–2172, 2010.

R. He, J. Rojas, and Y. Guan. A 3D object detection and pose estimation pipeline using RGB-D images. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1527–1532, 2017.

J. Heilala and P. Voho. Modular reconfigurable flexible final assembly systems. *Assembly Automation*, 21(1):20–30, 2001.

S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes. In *Asian Conference on Computer Vision*, pages 548–562, 2012.

S. Hirai. *Analysis and Planning of Manipulation Using the Theory of Polyhedral Convex Cones*. PhD thesis, Kyoto University, 1991.

H. Hirukawa and K. Iwata. Recognition of contact state based on geometric model. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1507–1512, 1991.

H. Hirukawa, T. Matsui, and K. Takase. A general algorithm for derivation and analysis of constraint for motion of polyhedra in contact. In *Workshop on IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 38–43, 1991.

B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus. Robust proprioceptive grasping with a soft robot hand. *Autonomous Robots*, 43(3):681–696, 2019.

L. S. Homem de Mello and A. C. Sanderson. AND/OR graph representation of assembly plans. *IEEE Transactions on Robotics and Automation*, 6(2):188–199, 1990.

H.-K. Hsu, W.-C. Hung, H.-Y. Tseng, C.-H. Yao, Y.-H. Tsai, M. Singh, and M.-H. Yang. Progressive domain adaptation for object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–5, 2019.

Z. Hu, W. Wan, and K. Harada. Designing a mechanical tool for robots with two-finger parallel grippers. *IEEE Robotics and Automation Letters (RA-L)*, 4 (3):2981–2988, 2019.

K. Ikeuchi and T. Suehiro. Toward an assembly plan from observation Part I: Task recognition with polyhedral objects. *IEEE Transactions on Robotics and Automation*, 10(3):368–385, 1994.

K. Ikeuchi, Z. Ma, Z. Yan, S. Kudoh, and M. Nakamura. Describing upper-body motions based on Labanotation for Learning-from-Observation robots. *International Journal of Computer Vision*, 126:1415–1429, 2018.

S. Imari, S. Yoichi, and I. Katsushi. Illumination from shadows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(3):290–300, 2003.

N. Inoue, R. Furuta, T. Yamasaki, and K. Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5001–5009, 2018.

P. Jiménez. Survey on assembly sequencing: a combinatorial and geometrical perspective. *Journal of Intellingent Manufacturing*, 24:235–250, 2013.

R. E. Jones and R. H. Wilson. A survey of constraints in automated assembly planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1525–1532, 1996.

R. E. Jones, R. H. Wilson, and T. L. Calton. On constraints in assembly planning. *IEEE Transactions on Robotics and Automation*, 14(6):849–863, 1998.

T. Kakuta, T. Oishi, and K. Ikeuchi. Real-time soft shadows in mixed reality using shadowing planes. In *IAPR Conference on Machine Vision Applications (IAPR MVA)*, pages 195–198, 2007.

P. Karagiannis, S. A. Matthaiakis, D. Andronas, K. Filis, C. Giannoulis, G. Michalos, and S. Makris. Reconfigurable assembly station: A consumer goods industry paradigm. *Procedia CIRP*, 81:1406–1411, 2019.

T. Karaulova and E. Shevtshenko. Work-cells concept development for high mix low volume market conditions. *Procedia Engineering*, 100:90–99, 2015.

C. Kardos and J. Váncza. Application of generic CAD models for supporting feature based assembly process planning. *Procedia CIRP*, 67:446–451, 2018.

H. Kato and M. Billinghurst. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *IWAR*, pages 85–94, 1999.

I. Kavasidis, S. Palazzo, D. R. Salvo, D. Giordano, and C. Spampinato. An innovative web-based collaborative platform for video annotation. *Multimedia Tools and Applications*, 70:413–432, 2014.

W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab. SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1521–1529, 2017.

W. Kim, M. Lorenzini, P. Balatti, P. D. H. Nguyen, U. Pattacini, V. Tikhanoff, L. Peternel, C. Fantacci, L. Natale, G. Metta, and A. Ajoudani. Adaptable workstations for human-robot collaboration: A reconfigurable framework for improving worker ergonomics and productivity. *IEEE Robotics & Automation Magazine*, 26(3):14–26, 2019.

Y. Kim and C. Sloth. Assembly strategy for deformable ring-shaped objects. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 357–358, 2020.

Y.-L. Kim, H.-C. Song, and J.-B. Song. Force control based jigless assembly strategy of a unit box using dual-arm and friction. In *IEEE International Symposium on Robotics (ISR)*, pages 1–3, 2013.

T. Kiyokawa, K. Tomochika, J. Takamatsu, and T. Ogasawara. Efficient collection and automatic annotation of real-world object images by taking advantage of post-diminished multiple visual markers. *Advanced Robotics*, 33(24):1264–1280, 2019a.

T. Kiyokawa, K. Tomochika, J. Takamatsu, and T. Ogasawara. Fully automated annotation with noise-masked visual markers for deep-learning-based object detection. *IEEE Robotics and Automation Letters (RA-L)*, 4(2):1972–1977, 2019b.

T. Kiyokawa, T. Sakuma, J. Takamatsu, and T. Ogasawara. Soft jig-driven assembly operations. *IEEE Robotics and Automation Letters (RA-L)*, 2020a.

T. Kiyokawa, J. Takamatsu, and T. Ogasawara. Assembly sequences based on multiple criteria against products with deformable parts. *IEEE Robotics and Automation Letters (RA-L)*, 2020b.

I. Krasin, T. Duerig, N. Alldrin, A. Veit, S. Abu-El-Haija, S. Belongie, D. Cai, Z. Feng, V. Ferrari, V. Gomes, A. Gupta, D. Narayanan, C. Sun, G. Chechik, and K. Murphy. OpenImages: A public dataset for large-scale multi-label and multi-class image classification, 2017. URL `https://github.com/openimages`.

J. Krüger, T. K. Lien, and A. Verl. Cooperation of human and machines in assembly lines. *CIRP Annals*, 58(2):628–646, 2009.

Y. Kuniyoshi, M. Inaba, and H. Inoue. Learning by watching: extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10(6):799–822, 1994.

K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.

M. Kyrarini, M. A. Haseeb, D. Ristić-Durrant, and A. Gräser. Robot learning of industrial assembly task via human demonstrations. *Autonomous Robots*, 43: 239–257, 2019.

A. Laemmle and S. Gust. Automatic layout generation of robotic production cells in a 3d manufacturing simulation environment. *Procedia CIRP*, 84:316–321, 2019.

E. Lathrop, I. Adibnazari, N. Gravish, and M. T. Tolley. Shear strengthened granular jamming feet for improved performance over natural terrain. In *IEEE International Conference on Soft Robotics (Robosoft)*, pages 388–393, 2020.

C. Lee, M. Kim, Y. J. Kim, N. Hong, S. Ryu, H. J. Kim, and S. Kim. Soft robot review. *International Journal of Control, Automation and Systems*, 15:3–15, 2017a.

C. Lee, M. Kim, Y. J. Kim, N. Hong, S. Ryu, H. J. Kim, and S. Kim. Soft robot review. *International Journal of Control, Automation and Systems*, 15:3–15, 2017b.

K. Lee, S. Joo, and H. I. Christensen. An assembly sequence generation of a product family for robot programming. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1268–1274, 2016.

L. Li. China's manufacturing locus in 2025: With a comparison of "Made-in-China 2025" and "Industry 4.0". *Technological Forecasting and Social Change*, 135:66–74, 2018.

R. Li and H. Qiao. A survey of methods and strategies for high-precision robotic grasping and assembly tasks—some new trends. *IEEE/ASME Transactions on Mechatronics*, 24(6):2718–2732, 2019.

Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox. DeepIM: Deep iterative matching for 6D pose estimation. In *European Conference on Computer Vision (ECCV)*, pages 683–698, 2018.

Z. Li, J. Wanga, M. S. Anwar, and Z. Zheng. An efficient method for generating assembly precedence constraints on 3D models based on a block sequence structure. *Computer-Aided Design*, 118:102773, 2020.

S. Lim, I. Kim, T. Kim, C. Kim, and S. Kim. Fast AutoAugment. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, page 6665–6675, 2019.

T.-Y. Lin, M. Maire, B. Serge, H. James, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, 2014.

H. Ling, J. Gao, A. Kar, W. Chen, and S. Fidler. Fast interactive object annotation with Curve-GCN. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5257–5266, 2019.

J. Liu, X. Zhang, and G. Hao. Survey on research and development of reconfigurable modular robots. *Advances in Mechanical Engineering*, 8(8):1–21, 2016a.

W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*, pages 21–37, 2016b.

Q. Lu, L. He, T. Nanayakkara, and N. Rojas. Precise in-hand manipulation of soft objects using soft fingertips with tactile sensing and active deformation. In *IEEE International Conference on Soft Robotics (Robosoft)*, pages 52–57, 2020.

Y. Lu, W. Zhang, C. Jin, and X. Xue. Learning attention map from images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1067–1074, 2012.

Y. Luo, L. Zheng, T. Guan, J. Yu, and Y. Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2507–2516, 2019.

K. Lupinetti, F. Giannini, M. Monti, and J.-P. Pernot. Automatic extraction of assembly component relationships for assembly model retrieval. *Procedia CIRP*, 50:472–477, 2016.

K. Lupinetti, F. Giannini, M. Monti, M. Rucco, and J.-P. Pernot. Identification of functional sets in mechanical assembly models. *International Conference on Innovative Design and Manufacturing*, pages 1–6, 2017.

Y. Maeda, H. Kikuchi, H. Izawa, H. Ogawa, M. Sugi, and T. Arai. An easily reconfigurable robotic assembly system. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2586–2591, 2003.

Y. Maeda, H. Kikuchi, H. Izawa, H. Ogawa, M. Sugi, and T. Arai. "Plug & Produce" functions for an easily reconfigurable robotic assembly cell. *Assembly Automation*, 27(3):253–260, 2007.

E. Maiettini, G. Pasquale, L. Rosasco, and L. Natale. Interactive data collection for deep learning object detectors on humanoid robots. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 862–868, 2017.

S. Makris, G. Michalos, A. A. Eytan, and G. Chryssolouris. Cooperating robots for reconfigurable assembly operations: Review and challenges. *Procedia CIRP*, 3:346–351, 2012.

A. A. Malik and A. Bilberg. Collaborative robots in assembly: A practical approach for tasks distribution. *Procedia CIRP*, 81:665–670, 2019.

K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool. Deep Extreme Cut: From extreme points to object segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 616–625, 2018.

J. A. Marvel, R. Bostelman, and J. Falco. Multi-robot assembly strategies and metrics. *ACM Computing Surveys*, 51(1), 2018.

B. J. McCarragher. Task primitives for the discrete event modeling and control of 6-DOF assembly tasks. *IEEE Transactions on Robotics and Automation*, 12 (2):280–289, 1996.

J. Michniewicz, G. Reinhart, and S. Boschert. CAD-based automated assembly planning for variable products in modular production systems. *Procedia CIRP*, 44:44–49, 2016.

C. Mitash, K. E. Bekris, and A. Boularias. A self-supervised learning system for object detection using physics simulation and multi-view pose estimation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 545–551, 2017.

J. Miura and K. Ikeuchi. Task-oriented generation of visual sensing strategies in assembly tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):126–138, 1998.

D. M. Montserrat, Q. Lin, J. Allebach, and E. J. Delp. Training object detection and recognition CNN models using data augmentation. In *IS&T International Symposium on Electronic Imaging*, pages 27–36, 2017.

G. Morris and L. Haynes. Robotic assembly by constraints. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 4, pages 1507–1515, 1987.

H. Mosemann, F. Rohrdanz, and F. Wahl. Assembly stability as a constraint for assembly sequence planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 233–238, 1998.

Y. Mukaigawa, H. Miyaki, S. Mihashi, and T. Shakunaga. Photometric image-based rendering for image generation in arbitrary illumination. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 652–659, 2001.

G. B. Murali, B. B. V. L. Deepak, M. V. A. R. Bahubalendruni, and B. B. Biswal. Optimal robotic assembly sequence planning using stability graph through stable assembly subset identification. *Proc. of the Institution of Mechanical Engineers, Part C: J. of Mechanical Engineering Science*, 233(15):5410–5430, 2019.

T. Murooka, K. Okada, and M. Inaba. Self-repair and self-extension by tightening screws based on precise calculation of screw pose of self-body with CAD data and graph search with regrasping a driver. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 79–84, 2019.

S. Naing, G. Burley, R. Odi, A. Williamson, and J. Corbett. Design for tooling to enable jigless assembly – an integrated methodology for jigless assembly. *SAE Transactions*, 109:299–311, 2000.

S. Nakaoka, A. Nakazawa, F. Kanehiro, K. Kaneko, M. Morisawa, H. Hirukawa, and K. Ikeuchi. Learning from observation paradigm: Leg task models for enabling a biped humanoid robot to imitate human dances. *International Journal of Robotics Research*, 26(8):829–844, 2007.

A. Neb. Review on approaches to generate assembly sequences by extraction of assembly features from 3d models. *Procedia CIRP*, 81:856–861, 2019.

V. Nguyen, T. F. Y. Vicente, M. Zhao, M. Hoai, and D. Samaras. Shadow detection with conditional generative adversarial networks. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4520–4528, 2017.

F. Okura, M. Kanbara, and N. Yokoya. Mixed-reality world exploration using image-based rendering. *ACM Journal on Computing and Cultural Heritage*, 8 (2):9:1–9:26, 2015.

R. Onizawa, T. Takamura, M. Tanaka, and S. Motohashi. Next-generation iot-based production system for high-mix low-volume products in an era of globalization - activities at hitachi's omika works. *Hitachi Review*, 65:58–63, 2016.

M. Onori, B. Langbeck, and P. Gröndahl. The MARK III flexible automatic assembly cell. *Robotics and Computer-Integrated Manufacturing*, 13(3):193–202, 1997.

L.-M. Ou and X. Xu. Relationship matrix based automatic assembly sequence generation from a cad model. *Computer-Aided Design*, 45(7):1053–1067, 2013.

Özkan Özmen, T. Batbat, T. Özen, C. Sinanoğlu, and A. Güven. Optimum assembly sequence planning system using discrete artificial bee colony algorithm. *Mathematical Problems in Engineering*, pages 1–14, 2018.

C. Pan, S. S.-F. Smith, and G. C. Smith. Automatic assembly sequence planning from STEP CAD files. *International Journal of Computer Integrated Manufacturing*, 19(8):775–783, 2006.

A. Panagopoulos, C. Wang, D. Samaras, and N. Paragios. Illumination estimation and cast shadow detection through a higher-order graphical model. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 673–680, 2011.

D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, and V. Ferrari. We don't need no bounding-boxes: Training object class detectors using only human verification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 854–863, 2016.

D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, and V. Ferrari. Extreme clicking for efficient object annotation. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4930–4939, 2017.

H. Park, J. Park, D. Lee, J. Park, M. Baeg, and J. Bae. Compliance-based robotic peg-in-hole assembly strategy without force feedback. *IEEE Transactions on Industrial Electronics*, 64(8):6299–6309, 2017.

P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal. Learning and generalization of motor skills by learning from demonstration. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 763–768, 2009.

A. R. Pathaka, M. Pandeya, and S. Rautaray. Application of deep learning for object detection. In *International Conference on Computational Intelligence and Data Science (ICCIDS)*, pages 1706–1717, 2018.

S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao. PVNet: Pixel-wise voting network for 6DoF pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4561–4570, 2019.

X. Peng, B. Sun, K. Ali, and K. Saenko. Learning deep object detectors from 3D models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1278–286, 2015.

P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Transactions on Graphics*, 22(3):313–318, 2003.

A. Perzylo, N. Somani, M. Rickert, and A. Knoll. An ontology for CAD data and geometric constraints as a link between product models and semantic robot task descriptions. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4197–4203, 2015.

D. T. Pham and S. H. Yeo. Strategies for gripper design and selection in robotic assembly. *International Journal of Production Research*, 29(2):303–316, 1991.

G. Pintzos, C. Triantafyllou, N. Papakostas, D. Mourtzis, and G. Chryssolouris. Assembly precedence diagram generation through assembly tiers determination. *International Journal of Computer Integrated Manufacturing*, 29(10): 1045–1057, 2016.

L. Qu, J. Tian, S. He, Y. Tan, and R. W. H. Lau. DeshadowNet: A multi-context embedding deep network for shadow removal. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2308–2316, 2017.

M. Rad and V. Lepetit. BB8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3848–3856, 2017.

M. Raessa, J. C. Y. Chen, W. Wan, and K. Harada. Human-in-the-loop robotic manipulation planning for collaborative assembly. *IEEE Transactions on Automation Science and Engineering*, 17(4):1800–1813, 2020.

V. N. Rajan, K. Sivasubramanian, and J. E. Fernandez. Accessibility and ergonomic analysis of assembly product and jig designs. *International Journal of Industrial Ergonomics*, 23(5):473–487, 1999.

I. G. Ramirez-Alpizar, K. Harada, and E. Yoshida. A simple assembly planner for the insertion of ring-shaped deformable objects. *Assembly Automation*, 38 (2):182–194, 2018.

E. Real, J. Shlens, S. Mazzocchi, X. Pan, and V. Vanhoucke. YouTube-BoundingBoxes: A large high-precision human-annotated data set for object detection in video. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7464–7473, 2017.

J. Redmon and A. Farhadi. YOLOv3: An incremental improvement. *CoRR*, abs/1804.02767, 2018.

J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once: Unified, real-time object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.

S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 91–99, 2015.

C. Rennie, R. Shome, K. E. Bekris, and A. F. D. Souza. A dataset for improved RGBD-based object detection and pose estimation for warehouse pick-and-place. *IEEE Robotics and Automation Letters (RA-L)*, 1(2), 2016.

A. Rojko. Industry 4.0 concept: Background and overview. *International Journal of Interactive Mobile Technologies*, 11(5):77, 2017.

F. J. Romero-Ramirez, R. Muñoz-Salinas, and R. Medina-Carnicer. Speeded up detection of squared fiducial markers. *Image and Vision Computing*, 76:38–47, 2018.

J. Rosell. Assembly and task planning using petri nets: A survey. *Proc. of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 218(8):987–994, 2004.

Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

M. Rucco, F. Giannini, K. Lupinetti, and M. Monti. A methodology for part classification with supervised machine learning. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 33(1):100–113, 2019.

O. Russakovsky, L.-J. Li, and L. Fei-Fei. Best of both worlds: Human-machine collaboration for object annotation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2121–2131, 2015.

B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008.

C. Sahin and T.-K. Kim. Recovering 6D object pose: A review and multi-modal analysis. In *European Conference on Computer Vision (ECCV)*, pages 15–31, 2018.

U. Saif, Z. Guan, B. Wang, J. Mirza, and S. Huang. A survey on assembly lines and its types. *Frontiers of Mechanical Engineering*, 9:95–105, 2014.

T. Sakuma, F. von Drigalski, M. Ding, J. Takamatsu, and T. Ogasawara. A universal gripper using optical sensing to acquire tactile information and membrane deformation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6431–6436, 2018.

T. Sakuma, E. Phillips, G. A. G. Ricardez, M. Ding, J. Takamatsu, and T. Ogasawara. A parallel gripper with a universal fingertip device using optical sensing and jamming transition for maintaining stable grasps. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5814–5819, 2019.

I. Sato, M. Hayashida, F. Kai, Y. Sato, and K. Ikeuchi. Fast image synthesis of virtual objects in a real scene with natural shadings. *Systems and Computers in Japan*, 36(14):102–111, 2005.

T. R. Savarimuthu, A. G. Buch, C. Schlette, N. Wantia, J. Roßmann, D. Martïnez, G. Alenyà, C. Torras, A. Ude, B. Nemec, A. Kramberger, F. Wörgötter, E. E. Aksoy, J. Papon, S. Haller, J. Piater, and N. Krüger. Teaching a robot the semantics of assembly tasks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(5):670–692, 2018.

S. Schaal. Learning from demonstration. In *Annual Conference on Neural Information Processing Systems (NIPS)*, page 1040–1046, 1996.

C. Schlette, A. G. Buch, F. Hagelskjær, I. Iturrate, D. Kraft, A. Kramberger, A. P. Lindvig, S. Mathiesen, H. G. Petersen, M. H. Rasmussen, T. R. Savarimuthu, C. Sloth, L. C. Sørensen, and T. N. Thulesen. Towards robot cell matrices for agile production – SDU Robotics' assembly cell at the WRC 2018. *Advanced Robotics*, 34(7-8):422–438, 2020.

M. Schwarz, A. Milan, A. S. Periyasamy, and S. Behnke. RGB-D object detection and semantic segmentation for autonomous manipulation in clutter. *International Journal of Robotics Research*, 37(4-5):437–451, 2018.

E. Shellshear, Y. Li, R. Bohlin, and J. S. Carlson. Contact-based bounding volume hierarchy for assembly tasks. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 9185–9190, 2020.

P. Shi, Z. Hu, K. Nagata, W. Wan, Y. Domae, and K. Harada. An adaptive pin array fixture to fix multiple parts. In *JSME Robotics and Mechatronics Conference (ROBOMECH)*, pages 2A2–K10, 2020.

M. Shibata, H. Dobashi, W. Uemura, S. Kotosaka, Y. Aiyama, T. Sakaguchi, Y. Kawai, A. Noda, K. Yokoi, and Y. Yokokohji. Task-board task for assembling a belt drive unit. *Advanced Robotics*, 34(7-8):454–476, 2020.

Y. Shiraki, K. Nagata, N. Yamanobe, A. Nakamura, K. Harada, D. Sato, and D. N. Nenchev. Modeling of everyday objects for semantic grasp. In *IEEE International Symposium on Robot and Human Interactive Communication*, pages 750–755, 2014.

S. Sierla, V. Kyrki, P. Aarnio, and V. Vyatkin. Automatic assembly planning based on digital product descriptions. *Computers in Industry*, 97:34–46, 2018.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, pages 1–14, 2015.

L. Sixt, B. Wild, and T. Landgraf. RenderGAN: Generating realistic labeled data. *Frontiers in Robotics and AI*, 5(66):1–9, 2018.

C. Sloth, A. Kramberger, and I. Iturrate. Towards easy setup of robotic assembly tasks. *Advanced Robotics*, 34(7-8):499–513, 2020.

G. C. Smith and S. S.-F. Smith. An enhanced genetic algorithm for automated assembly planning. *Robotics and Computer Integrated Manufacturing*, 18(5-6): 355–364, 2002.

S. S.-F. Smith, G. C. Smith, and X. Liao. Automatic stable assembly sequence generation and evaluation. *Journal of Manufacturing Systems*, 20(4):225–235, 2001.

K.-T. Song, C.-H. Wu, and S.-Y. Jiang. CAD-based pose estimation design for random bin picking using a RGB-D camera. *Journal of Intelligent & Robotic Systems*, 87(3-4):455–470, 2017.

M. Srinivas and L. M. Patnaik. Genetic algorithms: a survey. *Computer*, 27(6): 17–26, 1994.

H. Su, J. Deng, and L. Fei-Fei. Crowdsourcing annotations for visual object detection. In *AAAI Human Computation Workshop*, 2012.

M. Suchi, T. Patten, D. Fischinger, and M. Vincze. EasyLabel: A semi-automatic pixel-wise object annotation tool for creating robotic RGB-D datasets. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6678–6684, 2019.

L. Sun, C. Zhao, and R. Stolkin. Weakly-supervised DCNN for RGB-D object recognition in real-world applications which lack large-scale annotated training data. *CoRR*, abs/1703.06370, 2017.

M. Sundermeyer, Z.-C. Marton, M. Durner, M. Brucker, and R. Triebel. Implicit 3D orientation learning for 6D object detection from RGB images. In *European Conference on Computer Vision (ECCV)*, pages 699–715, 2018.

R. Szeliski. *Computer Vision: Algorithms and Applications*, chapter 2. Springer-Verlag London, 2011.

D. Sánchez, W. Wan, and K. Harada. Tethered tool manipulation planning with cable maneuvering. *IEEE Robotics and Automation Letters*, 5(2):2777–2784, 2020.

S. Tajima, S. Wakamatsu, T. Abe, M. Tennomi, K. Morita, H. Ubata, A. Okamura, Y. Hirai, K. Morino, Y. Suzuki, T. Tsuji, K. Yamazaki, and T. Watanabe. Robust bin-picking system using tactile sensor. *Advanced Robotics*, 34 (7-8):439–453, 2020.

R. Takahashi, T. Matsubara, and K. Uehara. RICAP: Random image cropping and patching data augmentation for deep CNNs. In *ACML*, pages 786–798, 2018.

J. Takamatsu. *Abstraction of Manipulation Tasks to Automatically Generate Robot Motion from Observation.* PhD thesis, University of Tokyo, 2003.

J. Takamatsu, T. Morita, K. Ogawara, H. Kimura, and K. Ikeuchi. Representation for knot-tying tasks. *IEEE Transactions on Robotics*, 22(1):65–78, 2006.

J. Takamatsu, K. Ogawara, H. Kimura, and K. Ikeuchi. Recognizing assembly tasks through human demonstration. *International Journal of Robotics Research*, 26(7):641–659, 2007.

J. Takamatsu, Y. Matsushita, and K. Ikeuchi. Estimating camera response functions using probabilistic intensity similarity. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

M. Tan, R. Pang, and Q. V. Le. EfficientDet: Scalable and efficient object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10781–10790, 2020.

H. Tanaka, Y. Sumi, and Y. Matsumoto. A visual marker for precise pose estimation based on lenticular lenses. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5222–5227, 2012.

S. Tao and M. Hu. A contact relation analysis approach to assembly sequence planning for assembly models. *Computer-Aided Design and Applications*, 14 (6):720–733, 2017.

K. Tariki, T. Kiyokawa, G. A. G. Ricardez, J. Takamatsu, and T. Ogasawara. 3D model-based assembly sequence optimization using insertionable properties of parts. In *IEEE/SICE International Symposium on System Integration (SII)*, pages 1400–1405, 2020.

K. Tariki, T. Kiyokawa, T. Nagatani, and J. Takamatsu. Generating complex assembly sequences from 3D CAD models considering insertion relations. *Advanced Robotics*, 2021.

K. Tatemura and H. Dobashi. Strategy for roller chain assembly with parallel jaw gripper. *IEEE Robotics and Automation Letters (RA-L)*, 5(2):2435–2442, 2020.

B. Tekin, S. N. Sinha, and P. Fua. Real-time seamless single shot 6D object pose prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 292–301, 2018.

M. Tennomi, A. Okamura, Y. Nakamura, T. Abe, S. Wakamatsu, S. Tajima, T. Nishimura, Y. Hirai, T. Sawada, N. Ichikawa, T. Tsuji, K. Yamazaki, Y. Suzuki, and T. Watanabe. Development of assembly system with quick and low-cost installation. *Advanced Robotics*, 34(7-8):531–545, 2020.

J. C. Trappey and C. R. Liu. A literature survey of fixturedesign automation. *International Journal of Advanced Manufacturing Technology*, 5:240–255, 1990.

J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. In *Conference on Robot Learning*, volume 87, pages 306–316, 2018.

P. Tsarouchi, A.-S. Matthaiakis, S. Makris, and G. Chryssolouris. On a human-robot collaboration in an assembly cell. *International Journal of Computer Integrated Manufacturing*, 30(6):580–589, 2017.

H. Tsukune, M. Tsukamoto, T. Matsushita, F. Tomita, K. Okada, T. Ogasawara, K. Takase, and T. Yuba. Modular manufacturing. *Journal of Intelligent Manufacturing*, 4:163–181, 1993.

E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7167–7176, 2017.

F. von Drigalski, C. Nakashima, Y. Shibata, Y. Konishi, J. C. Triyonoputro, K. Nie, D. Petit, T. Ueshiba, R. Takase, Y. Domae, T. Yoshioka, Y. Ijiri, I. G. Ramirez-Alpizar, W. Wan, and K. Harada. Team O2AS at the world robot summit 2018: an approach to robotic kitting and assembly tasks using general purpose grippers and tools. *Advanced Robotics*, 34(7-8):514–530, 2020a.

F. von Drigalski, C. Schlette, M. Rudorfer, N. Correll, J. C. Triyonoputro, W. Wan, T. Tsuji, and T. Watanabe. Robots assembling machines: learning from the World Robot Summit 2018 Assembly Challenge. *Advanced Robotics*, 34(7-8):408–421, 2020b.

C. Vondrick and D. Ramanan. Video annotation and tracking with active learning. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 28–36, 2011.

J. Walker, T. Zidek, C. Harbel, S. Yoon, F. Strickland, S. Kumar, and M. Shin. Soft robotics: A review of recent developments of pneumatic soft actuators. *Actuators*, 9(3), 2020.

L. Wang, S. Keshavarzmanesh, H.-Y. Feng, and R. O. Buchal. Assembly process planning and its future in collaborative manufacturing: a review. *International Journal of Advanced Manufacturing Technology*, 41(1-2):132–144, 2009.

P. Wang, G. Xu, Y. Cheng, and Q. Yu. A simple, robust and fast method for the perspective-n-point problem. *Pattern Recognition Letters*, 108:31–37, 2018.

Y. Wang, K. Harada, and W. Wan. Motion planning of skillful motions in assembly process through human demonstration. *Advanced Robotics*, 34(16): 1079–1093, 2020a.

Z. Wang, E. Wang, and Y. Zhu. Image segmentation evaluation: a survey of methods. *Artificial Intelligence Review*, 53:5637–5674, 2020b.

T. Watanabe, K. Yamazaki, and Y. Yokokohji. Survey of robotic manipulation studies intending practical applications in real environments -object recognition, soft robot hand, and challenge program and benchmarking-. *Advanced Robotics*, 31(19-20):1114–1132, 2017.

J. Watson, A. Miller, and N. Correll. Autonomous industrial assembly using force, torque, and rgb-d sensing. *Advanced Robotics*, 34(7-8):546–559, 2020.

C. Weckenborg, K. Kieckhäfer, C. Müller, M. Grunewald, and T. S. Spengler. Balancing of assembly lines with collaborative robots. *Business Research*, 13: 93–132, 2020.

K. Whybrew and B. K. A. Ngoi. Computer aided design of modular fixture assembly. *International Journal of Advanced Manufacturing Technology*, 7: 267–276, 1992.

R. H. Wilson and J.-C. Latombe. Geometric reasoning about mechanical assembly. *Artificial Intelligence*, 71:371–396, 1994.

W. Wohlkinger, A. Aldoma, R. B. Rusu, and M. Vincze. 3DNet: Large-scale object class recognition from CAD models. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5384–5391, 2012.

J. Wolter and E. Kroll. Toward assembly sequence planning with flexible parts. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1517–1524, 1996.

WRS2018. Industrial robotics category assembly challenge rules and regulations. `https://worldrobotsummit.org/download/rulebook-en/rulebook-Assembly_Challenge.pdf`, October 2018.

WRS2020. Industrial robotics category assembly challenge rules and regulations. `https://worldrobotsummit.org/wrs2020/challenge/download/Rules/DetailedRules_Assembly_EN.pdf`, January 2020.

Y. Xiang, R. Mottaghi, and S. Savarese. Beyond PASCAL: A benchmark for 3D object detection in the wild. In *Workshop on Applications of Computer Vision (WACV)*, pages 75–82, 2014.

Y. Xiang, W. Kim, W. Chen, J. Ji, C. Choy, H. Su, R. Mottaghi, L. Guibas, and S. Savarese. ObjectNet3D: A large scale database for 3D object recognition. In *European Conference on Computer Vision (ECCV)*, pages 160–176, 2016.

Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018.

K. Yamazaki, T. Higashide, D. Tanaka, and K. Nagahama. Assembly manipulation understanding based on 3D object pose estimation and human motion estimation. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 802–807, 2018.

H. Yang, J. Chen, C. Wang, J. Cui, and W. Wei. Intelligent planning of product assembly sequences based on spatio-temporal semantic knowledge. *Assembly Automation*, 40(5), 2020.

M. Yim, W. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, and G. S. Chirikjian. Modular self-reconfigurable robot systems [grand challenges of robotics]. *IEEE Robotics & Automation Magazine*, 14(1):43–52, 2007.

Y. Yokokohji, Y. Yu, N. Nakasu, and T. Yoshikawa. Quasi-dynamic manipulation of constrained object by robot fingers in assembly tasks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 144–151, 1993.

Y. Yokokohji, Y. Kawai, M. Shibata, Y. Aiyama, S. Kotosaka, W. Uemura, A. Noda, H. Dobashi, T. Sakaguchi, and K. Yokoi. Assembly challenge: a robot competition of the industrial robotics category, World Robot Summit – summary of the pre-competition in 2018. *Advanced Robotics*, 33(17):876–899, 2019.

E. Yoshida, S. Murata, A. Kamimura, K. Tomita, H. Kurokawa, and S. Kokaji. A self-reconfigurable modular robot: Reconfiguration planning and experiments. *International Journal of Robotics Research*, 21(10-11):903–915, 2002.

T. Yoshikawa, Y. Yokokohji, and Y. Yu. Assembly planning operation strategies based on the degree of constraint. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 682–687, 1991.

Y. Yu, Y. Yokokohji, and T. Yoshikawa. Two kinds of degree of freedom in constraint state and their application to assembly planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1993–1999, 1996.

A. Zeng, K.-T. Yu, S. Song, D. Suo, E. W. Jr., A. Rodriguez, and J. Xiao. Multi-view self-supervised deep learning for 6D pose estimation in the Amazon Picking Challenge. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1386–1393, 2017.

X. F. Zha, S. Y. E. Lim, and S. C. Fok. Integrated intelligent design and assembly planning: A survey. *International Journal of Advanced Manufacturing Technology*, 14(9):664–685, 1998.

H. Zhang, L. Zheng, P. Wang, and W. Fan. Intelligent configuring for agile joint jig based on smart composite jig model. *The International Journal of Advanced Manufacturing Technology*, pages 1–23, 2019.

J. Zhang and X. Fang. Challenges and key technologies in robotic cell layout design and optimization. *Journal of Mechanical Engineering Science*, 231(15): 2912–2924, 2017.

Q. Zhao, T. Sheng, Y. Wang, Z. Tang, Y. Chen, L. Cai, and H. Ling. M2Det: A single-shot object detector based on multi-level feature pyramid network. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 9259–9266, 2019a.

Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu. Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 30 (11):3212–3232, 2019b.

Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. Random erasing data augmentation. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 13001–13008, 2020.

M. Zhu, K. G. Derpanis, Y. Yang, S. Brahmbhatt, M. Zhang, C. Phillips, M. Lecce, and K. Daniilidis. Single image 3D object detection and pose estimation for grasping. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3936–3943, 2014.

Z. Zhu and H. Hu. Robot learning from demonstration in robotic assembly: A survey. *Robotics*, 7(2):17, 2018.

K. Zuiderveld. Contrast limited adaptive histograph equalization. *Graphic Gems IV. San Diego: Academic Press Professional*, pages 474–485, 1994.

Publication List

Journal (refereed)

1. Takuya Kiyokawa, Keita Tomochika, Jun Takamatsu and Tsukasa Ogasawara: "Efficient Collection and Automatic Annotation of Real-World Object Images by Taking Advantage of Post-Diminished Multiple Visual Markers," Advanced Robotics, vol. 33, no. 24, pp. 1264-1280, 2019. (corresponds to Chapter 2)

2. Takuya Kiyokawa, Keita Tomochika, Jun Takamatsu and Tsukasa Ogasawara: "Fully Automated Annotation with Noise-Masked Visual Markers for Deep-Learning-Based Object Detection," IEEE Robotics and Automation Letters (RA-L), vol. 4, no. 2, pp. 1972-1977, 2019. (presented at ICRA2019) (corresponds to Chapter 2)

International Conference (refereed)

1. Takuya Kiyokawa, Jun Takamatsu and Tsukasa Ogasawara: "Assembly Sequences Based on Multiple Criteria Against Products with Deformable Parts," in IEEE International Conference on Robotics and Automation (ICRA), Online, June, 2021. (accepted) (corresponds to Chapter 3)

Journal (in Japanese, refereed)

1. 清川 拓哉, 佐久間 達也, 高松 淳, 小笠原 司: "組立部品の形状と挙動に応じて変形する柔軟治具", 日本ロボット学会誌, vol. 39, no. 3, 2021. (RSJ2020にて発表) (corresponds to Chapter 4)

2. 清川 拓哉, 田力 健人, 高松 淳, 小笠原 司: "変形物体を含む製品の組立作業計画を考慮した3次元CADモデルからの組立順序生成", 日本ロボット学会誌, vol. 39, no. 2, 2021. (RSJ2020にて発表) (corresponds to Chapter 3)

Domestic Conference (in Japanese)

1. 清川 拓哉, 高松 淳, 小笠原 司: “容器包装廃棄物を分別するロボットビジョンシステムのための自動学習データセット生成と実環境との差異を考慮した学習”, ロボティクス・メカトロニクス講演会2020（ROBOMECH2020）, 2P2-J13, オンライン, 5月, 2020. (corresponds to Chapter 2)

2. 清川 拓哉, 友近 圭汰, 高松 淳, 小笠原 司: “ハンドアイカメラ搭載のマニピュレータと回転ステージを用いた陳列商品検出用の学習データセットの自動収集システム”, 第20回 計測自動制御学会システムインテグレーション部門講演会（SI2019）, 1C5-03, 高松, 12月, 2019. (corresponds to Chapter 2)

3. 田力 健人, 清川 拓哉, 永谷 智貴, 高松 淳, 小笠原 司: “部品の3次元モデルのみを用いた部品数の多い製品に対する組立順序の最適化”, 第37回 日本ロボット学会学術講演会（RSJ2019）, 3I1-04, 東京, 9月, 2019. (corresponds to Chapter 3)

4. 清川 拓哉, 友近 圭汰, 高松 淳, 小笠原 司: “Web アプリを用いた多視点画像データセットの効率的な収集手法”, 第19回 計測自動制御学会システムインテグレーション部門講演会（SI2018）, 3D4-12, 大阪, 12月, 2018. (corresponds to Chapter 2)

5. 友近 圭汰, 清川 拓哉, 高松 淳, 小笠原 司: “視覚マーカを用いた物体検出のための偏りのない学習データ自動収集システム”, ロボティクス・メカトロニクス講演会2018（ROBOMECH2018）, 2A2-J18, 北九州, 6月, 2018. (corresponds to Chapter 2)

Patent

1. Keita Tomochika, Takuya Kiyokawa, Tsukasa Ogasawara, Jun Takamatsu and Ming Ding: “Creation Method of Training Data Set and Apparatus,” Japanese Patent Application No. 2018-066282, 2018 (PCT pending, 2020). (corresponds to Chapter 2)

2. Keita Tomochika, Takuya Kiyokawa, Tsukasa Ogasawara, Jun Takamatsu and Ming Ding: “Creation Method of Training Data Set, Object Recognition and Pose Estimation Method,” Japanese Patent No. 6474179, 2017. (corresponds to Chapter 2)

Award

1. Special Award, Technical Academy Research Presentation in IIFES, November, 2019. (corresponds to Chapter 2)
2. Certified as Super Creator by METI and IPA of Japan, MITOU program 2018, May, 2019. (corresponds to Chapter 2)
3. Excellent Research Project Award, NAIST CICP 2018, February, 2019. (corresponds to Chapter 3)

Related Publication

Journal (refereed)

1. Kento Tariki, Takuya Kiyokawa, Tomoki Nagatani, Jun Takamatsu and Tsukasa Ogasawara: "Generating Complex Assembly Sequences from 3D CAD Models Considering Insertion Relations," Advanced Robotics, vol. 35, no. 6, pp. 337-348, 2021.

2. Yuto Tsuchiya, Takuya Kiyokawa, Gustavo Alfonso Garcia Ricardez, Jun Takamatsu and Tsukasa Ogasawara: "Pouring from Deformable Containers Using Dual-Arm Manipulation and Tactile Sensing," International Journal of Robotic Computing (IJRC), vol. 1, no. 2, pp. 123-143, 2019.

International Conference (refereed)

1. Takuya Kiyokawa, Tatsuya Sakuma, Jun Takamatsu and Tsukasa Ogasawara: "Soft-Jig-Driven Assembly Operations," in IEEE International Conference on Robotics and Automation (ICRA), Online, June, 2021. (accepted)

2. 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 IEEE/SICE International Symposium on System Integration (SII), TuC2.2, Online, January, 2021.

3. Hiroki Katayama, Takuya Kiyokawa, Jun Takamatsu and Tsukasa Ogasawara: "Azimuth Angle Estimation Based on Sound Wave Reflection for Mirrors and Transparent Objects," in IEEE/SICE International Symposium on System Integration (SII), WeC3.1, Online, January, 2021.

4. Naoki Shirakura, Takuya Kiyokawa, Hikaru Kumamoto, Jun Takamatsu and

Tsukasa Ogasawara: “Semi-automatic Collection of Marine Debris by Collaborating UAV and UUV,” in IEEE International Conference on Robotic Computing (IRC), 118, Online, November, 2020.

5. Kento Tariki, Takuya Kiyokawa, Gustavo Alfonso Garcia Ricardez, Jun Takamatsu and Tsukasa Ogasawara: “3D Model-Based Assembly Sequence Optimization Using Insertionable Properties of Parts,” in IEEE/SICE International Symposium on System Integration (SII), We2E.6, Hawaii, USA, January, 2020.

6. Kento Tariki, Takuya Kiyokawa, Tomoki Nagatani, Jun Takamatsu and Tsukasa Ogasawara: “3D Model-Based Non-interference Assembly Sequence Generation for Products with a Large Number of Parts,” in IEEE International Conference on Cybernetics and Intelligent Systems and IEEE International Conference on Robotics, Automation and Mechatoronics (CIS-RAM), MoB3.63, Bangkok, Thailand, November, 2019.

7. Takuya Kiyokawa, Ming Ding, Gustavo Alfonso Garcia Ricardez, Jun Takamatsu and Tsukasa Ogasawara: “Generation of a Tactile-Based Pouring Motion Using Fingertip Force Sensors,” in IEEE/SICE International Symposium on System Integration (SII), Tu2D.3, Paris, France, January, 2019.

Domestic Conference (in Japanese)

1. 吉本 幸太郎, 清川 拓哉, 高松 淳, 小笠原 司: “リサイクルロボットのための熱画像を用いた容器包装廃棄物の領域抽出と材料分類”, 第26回ロボティクスシンポジア, 3A1, オンライン, 3月, 2021. (査読あり)

2. 清川 拓哉, 龍田 侑弥, 片山 寛基, 高松 淳, 小笠原 司: “ロボットアームを用いたアルミニウム廃材の密集率調整システム”, 第21回 計測自動制御学会システムインテグレーション部門講演会（SI2020）, 1G1-18, オンライン, 12月, 2020.

3. 龍田 侑弥, 佐久間 達也, 清川 拓哉, 高松 淳, 小笠原 司: “平板廃棄物把持のためのニードル付きジャミング吸着グリッパの開発”, 第21回 計測自動制御学会システムインテグレーション部門講演会（SI2020）, 2E2-07, オンライン, 12月, 2020.

4. 龍田 侑弥, 清川 拓哉, 高松 淳, 小笠原 司: “球状ジャミンググリッパを用いた物体操作 —力制御を必要としない転がし操作—”, 第38回 日本ロボット学会学術講演会（RSJ2020）, 3B2-04, オンライン, 10月, 2020.

5. 清川 拓哉, 片山 寛基, 高松 淳, 小笠原 司: “Active Vaporによる鏡面と透明物

体表面の法線推定”, 第23回 画像の認識・理解シンポジウム（MIRU2020）, OS3-2A-3, オンライン, 8月, 2020. (査読あり, ショート発表)

6. 清川 拓哉, 高松 淳, 小笠原 司: “包装商品のランダムピッキングシステムのためのパッケージ表面の法線方向と不可視率の自動アノテーション”, 2020年度 人工知能学会全国大会（第34回）（JSAI2020）, 4Rin1-93, オンライン, 6月, 2020. (査読あり)

7. 白倉 尚貴, 熊本 光, 清川 拓哉, 清水 拓海, 高松 淳, 小笠原 司: “水中ドローンによるティーチングプレイバックを用いた浮遊物の半自動回収システム”, 第20回 計測自動制御学会システムインテグレーション部門講演会（SI2019）, 3C5-10, 高松, 12月, 2019.

8. 清川 拓哉, 高松 淳, 笹渕 一宏, 小笠原 司, 池内克史: “視覚マーカを貼付した物体の観察による回転関節のパラメータ推定”, 第37回 日本ロボット学会学術講演会（RSJ2019）, 3E2-03, 東京, 9月, 2019.

9. 片山 寛基, 清川 拓哉, 高松 淳, 小笠原 司: “音波の周波数による反射特性の変化を用いた物体面の法線方向推定の可能性の検証”, 第37回 日本ロボット学会学術講演会（RSJ2019）, 2I2-03, 東京, 9月, 2019.

10. 土屋 裕杜, 清川 拓哉, Gustavo Alfonso Garcia Ricardez, 高松 淳, 小笠原 司: “双腕ロボットを用いた力覚情報に基づく柔軟容器からの注ぎ動作”, 第19回 計測自動制御学会システムインテグレーション部門講演会（SI2018）, 3A2-07, 大阪, 12月, 2018.

11. 北村 勇希, 清川 拓哉, Ming Ding, 高松 淳, 小笠原 司: “タッチケアロボットのための温度制御可能な受動機構を有する柔軟指の開発”, 第36回 日本ロボット学会学術講演会（RSJ2018）, 2D1-01, 春日井, 9月, 2018.

12. 趙 崇貴, 清川 拓哉, 友近 圭汰, 吉川雅博, 小笠原 司: “前腕形状計測に基づく手の動作認識によるロボットアームの操作”, ヒューマンインタフェース学会ヒューマンインタフェースシンポジウム2017, 7D2-2, 大阪, 9月, 2017.

Appendix

A. Automatic Dataset Collection in Other Use Cases

A.1. Pose Adjustments of Camera and Target Object

Figure 6.1 shows a large-scale dataset collection system compared to the automatic training dataset collection system using the small robot arm described in Chapter 2. This system can automatically collect a large amount of object images on the conveyor. Thus this system can be used to collect the dataset for training a detection system used for a line production rather than the cell production.

The system generates a training dataset unbiased in terms of the data quantity of the position and orientation of target objects. The collection time is in a shorter time than manual method. In the experiments, I verified the effectiveness of the automatic system by comparing with the manual method in both the time to collect training data and the accuracy of the trained vision system.

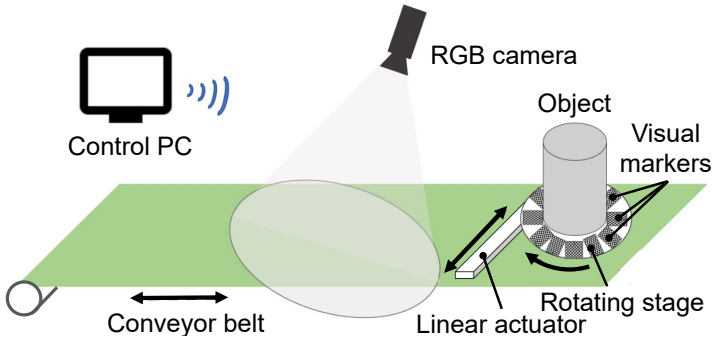
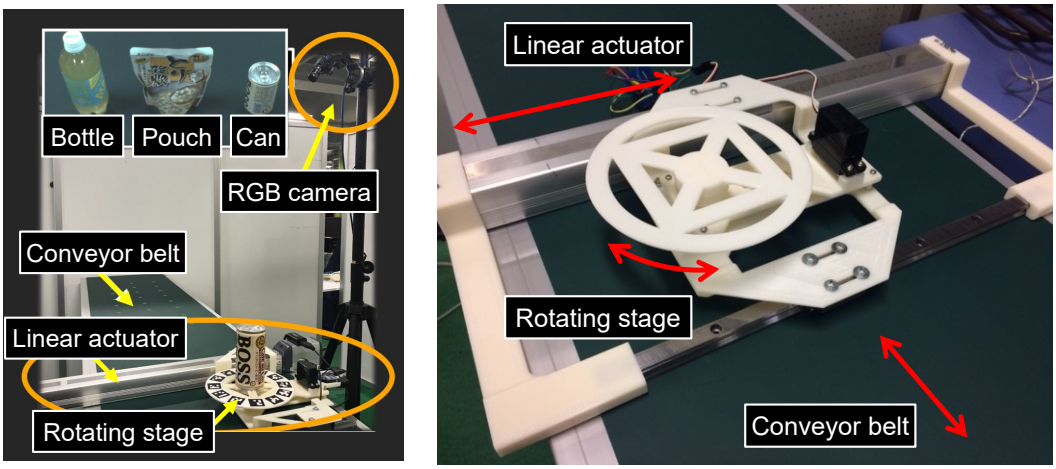


Figure 6.1. Overview of automatic dataset collection system with a conveyor belt.



(a) Experiment environment

(b) Pose adjustment system

Figure 6.2. Automatic dataset collection system with a pose adjustment system.

System overview

Figure 6.2 shows the system consisting of a conveyor belt (Okura Yusoki, BELCON MINI III), a linear actuator (IAI, CP4-SA5C-I-42P-12-400-P3-M), a rotating stage (OptoSigma, OSMS-60YAW) to move and rotate the object (hereinafter, pose adjustment device), and a monocular RGB camera (FLIR Systems, Flea3 FL30U3-88S2C). The RGB camera fixed on a tripod captures a video of the object during the movements of the conveyor, linear actuator and rotating stage.

Figure 6.3 shows the processing flow of collecting the training image dataset. First, we attach visual markers to the turntable on the rotating stage, where the target object is placed. Second, multiple images of the target object are captured, while changing their position and orientation on the conveyor belt. The pose adjustment system are controled to avoid overlap in position and orientation of the object captured on the conveyor to collect an unbiased image dataset.

Third, using the detection results of the visual markers and object size information, we assign the bounding box and category label as the object annotations to the images. Forth, we masked the images with a background image to delete unnecessaly object areas. Finally, we train a deep-learning-based detection system using the generated training dataset.

Evaluation

Experimental setup In the experiment, the detection system is trained using the training datasets collected by the manual method and the automatic method. I compare the time required to generate the training datasets and the accuracies

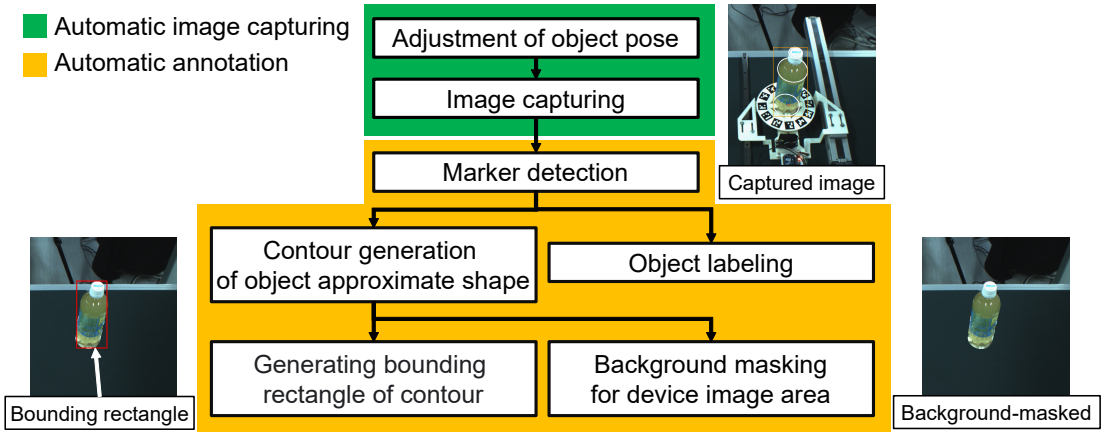


Figure 6.3. Process flow of generating images for training dataset.

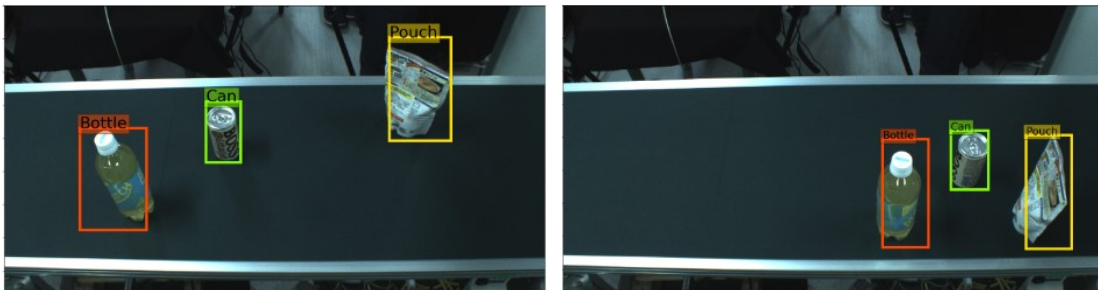


Figure 6.4. Detection results of the vision system trained with the datasets collected using the pose adjsutment system.

of the detection systems trained using the two datasets. Figure 6.2 (a) shows the experimental environment. I selected three types of objects with different shapes shown in Figure 6.2 (a) as detection-target object. I collected 500 images for one object.

In the manual collection, two people took pictures and annotated the images manually. When collecting the image dataset, one person randomly arranged the position and orientation of the object, and the other person took the image. The orientations were arranged in 45-degree increments. In the manual collection, 500 images are taken so that three types of objects can be captured in an image. We use labelImg* for the manual annotations.

In the automatic collection, the conveyor transport the object at low speed of the belt movements. The linear actuator changes the object position from 0 to 250 mm in 50-mm increments. The rotating stage changes the orientation in

*LabelImg (Available: `https://github.com/tzutalin/labelImg` [Accessed: 25- Nov- 2020])

Table 6.1. Time to collect training data [min].

	Manual	Automatic
Capturing time	80	139
Annotating time	217	
Total time	297 (5.0 h)	139 (2.3 h)

Table 6.2. Detection performance [%].

	Manual			Automatic		
Object	F.	Prec.	Rec.	F.	Prec.	Rec.
Bottle	99	98	100	73	98	58
Pouch	72	88	61	91	99	84
Can	82	79	85	94	92	97
Mean	84	88	82	84	96	77

45-degree increments.

By controlling the object position moved by the drive of the conveyor with a time command of 1 second, it was possible to take pictures at 9 points in the drive direction of the conveyor.

Collection time Table 6.1 shows the time to generate the training dataset for each method. The automatic method reduced 53.2% of the generation time of the manual method. The main reason for the reduction in time is that the automatic system allows the dataset collection process including the image capturing and the annotation to be performed in parallel. The current system takes longer to capture images compared to manual system, but the annotation time is included in the time to take the image, resulting in 0 minutes.

Accuracy of detection system I used 100 images to evaluate the trained detection system. I compared the results of object detection using the two detection systems with the true values (manually annotated data). Among the detected objects, those with reliability of 60 percent or more are determined to be correctly detected. Figure 6.4 shows the detection results of the automatic collection method.

Table 6.2 shows the detection accuracies. The results in the table refer to the F-measure, Precision, and Recall, respectively. The results of the manual method show that the accuracy of the bottle detection is very high. For the detection of

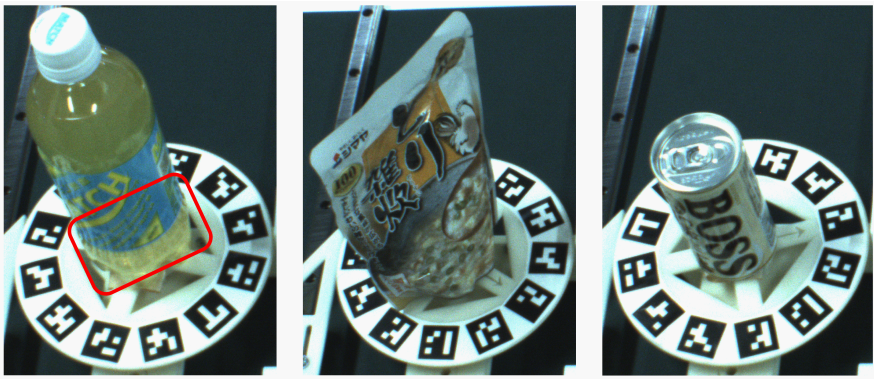
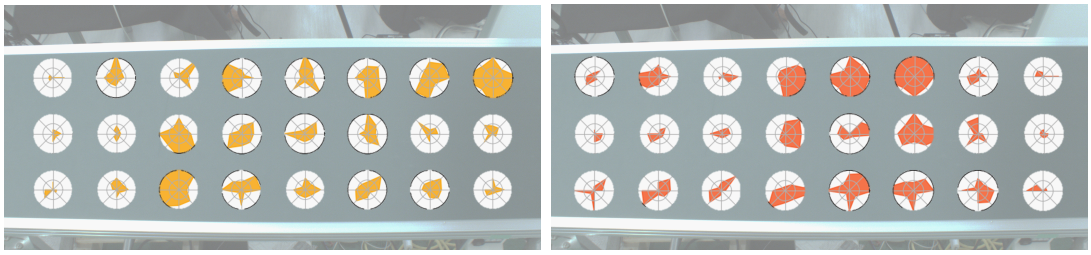


Figure 6.5. Reflections appeared on the object surface.



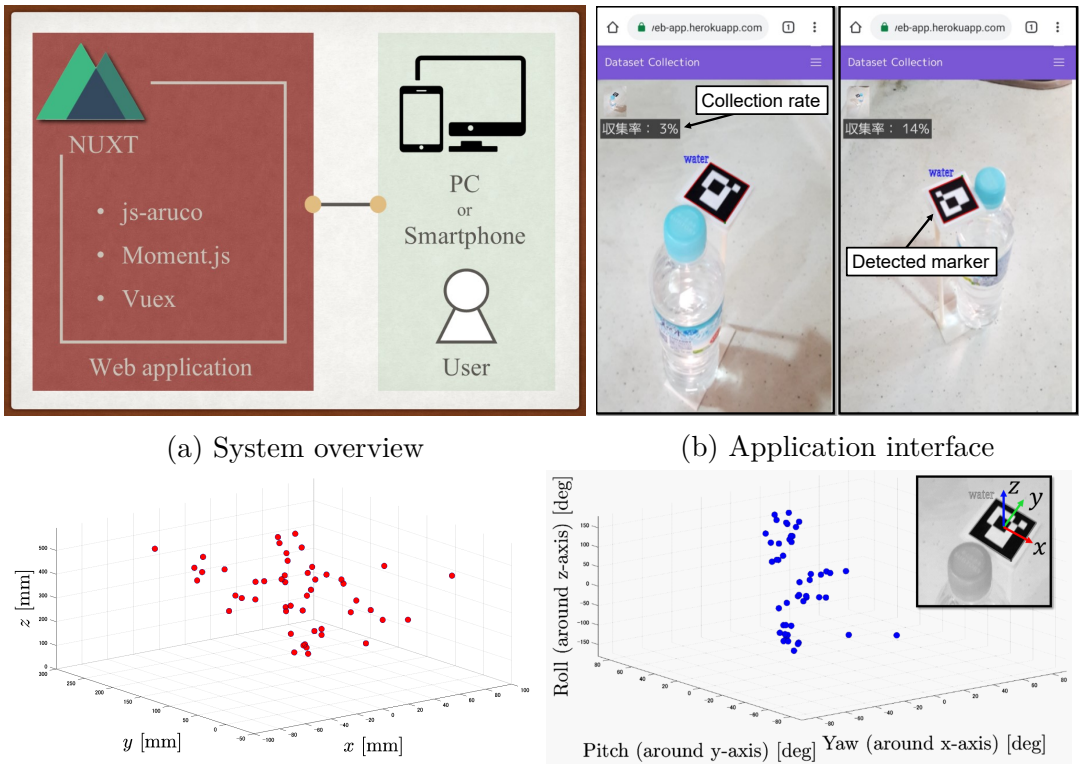
(a) Bottle (b) Pouch

Figure 6.6. Visualization of the collected dataset in terms of the position and orientation of the object.

pouches and cans, the results of the automatic method are much more accurate than the results by manual method. In the precision of the automatic method, the all values are very high, but, the recall of the bottles is significantly worse.

Discussion There are two possible reasons of the extremely low recall of the bottle in the automatic method. One of the drawbacks in the use of visual markers is that the marker-detection error in the Z axial direction is large, and when the error is large, an error of 10 degrees or more may occur. Consequently, since the bottle height is high, unexpected masking to the area of the target object occurs. The other reason is that, as shown in Figure 6.5, the marker apperance is reflected on the bottle surface. The reflection of the visual marker was not be appered on the surfaces of the pouch and can, so it is highly possible that the reflection was the reason.

As shown on the visualization of Figure 6.6, the object position and orientation of the pouches and cans included in the training images have variations. Also, for the pouch and can, the results by automatic method shows high accuracies in



(a) System overview

(b) Application interface

(c) Scattering collected position data

(d) Scattering collected orientation data

Figure 6.7. Overview of Web application for automatic dataset collection.

terms of F-measure, precision, and recall values. In the case of a deep-learning-based detection system, in order to achieve highly accurate detection, there is a possibility that a training dataset unbiased in terms of the position and orientation of the target object is better.

A.2. Human-in-the-Loop Collection using Web Application

I propose an image dataset collection system using a Web application. With the proposed Web application displaying the collection rate, users can efficiently collect a multi-view image dataset by moving the smartphone's or tablet's camera. Figure 6.7 shows the overview of the application configuration and interface to collect and annotate images.

Collection of multi-viewpoint image datasets

The purpose of this system is to efficiently collect image datasets of objects from multiple viewpoints even in any places. The Web application allows users to easily record multi-viewpoint images from free viewpoints. By implementing the function of automatic annotation by visual markers that I have proposed as a Web application, the user's operation is only to start the application and move the device itself. In order to collect an unbiased multi-viewpoint image dataset, the system saves images and annotations as a dataset so that the viewpoint information (position and orientation of the object seen from the camera) does not overlap with the existing data. Furthermore, the function of displaying the data collection rate implemented in the Web application allows the user to quickly collect data by relying on the collection rate.

I developed the Web application using NUXT[†]. I used a JavaScript library to satisfy various functions of the application. Heroku[‡] allows us to easily deploy, manage, and scale Web applications, temporarily publish the Web application to the Internet and then publish it on the Internet. We just run the application on the smartphone device.

I developed a Web application that does not depend on the smartphone device instead of a native application (Android application, iOS application, etc.). The implemented application can be operated on smartphones, tablets, PCs, etc.

Collection rate display function

The collection rate is displayed in real time on the screen of the Web application for the purpose of collecting multi-viewpoint images unbiased in terms of the object pose. The collection rate is displayed in the upper left of the screen. The collection rate is calculated based on the data of the 3D position and orientation of the visual marker in the camera coordinate system. If there is a difference of 25 mm or more in any of the axial directions of x, y, and z in the 3D position data acquired so far, the collection rate will increase and the image and annotation data will be saved. Furthermore, even if the position data does not differ, if there is a difference of 30 deg or more in the rotation angle around any axis of x, y, and z in the 3D posture data, the image and annotation are saved. This can reduce duplication of data related to the viewpoint. The thresholds of 25 mm and 30 deg were empirically set so that they could be collected over a wide range.

The display of the collection rate leads to the gamification of the image dataset collection. Users can enjoy collecting while thinking about ways to increase the

[†]Available: `https://nuxtjs.org/` [Accessed: 25- Nov- 2020]
[‡]Available: `https://jp.heroku.com/` [Accessed: 25- Nov- 2020]

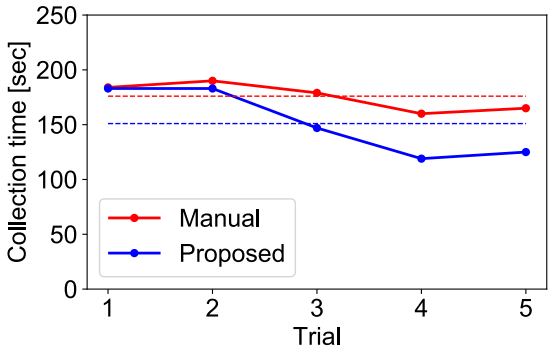


Figure 6.8. Time to collect the image dataset [sec].

collection rate efficiently. If such a game can be provided, rapid image dataset collection can be achieved in the end.

Evaluation

Experimental setup The collection procedure of the dataset in the experiment is as follows:

1. Attach a visual marker at a position where the object can be easily detected.
2. Start the Web application on your device.
3. Move the camera while checking the collection rate on the Web application interface to take a multi-viewpoint image (image capturing and data saving are automatically executed on the back end of the Web application).

A visual marker is fixed onto a 3D-printed jig, and an object is attached to the jig to fix the relative position and orientation between the marker and the object. I conducted five trials to collect 50 images using an Android smartphone, and if 50 images are collected, the collection rate is set to 100 percent (100%).

Collection time The time to collect the 50 images in five trials is shown in Figure 6.8. The collection times in the five trials by manual method are 184, 190, 179, 160 and 165 seconds. The average time of the five trials is 176 seconds which is shown in Figure 6.8 as the red dotted line. The collection times in five trials by manual method are 183, 183, 147, 119 and 125 seconds. The average time of the five trials is 151 seconds which is shown in Figure 6.8 as the blue dotted line. In the manual method, the annotator took images one by one using the native camera application of the smartphone without using the proposed Web application. When using the proposed Web application, it took less than 3 minutes for the

collection in all the five trials. With the current prototype system, it is possible to collect 1000 image data sets in one hour.

Compared with the method without using the Web application, the average collection time by the proposed method is shortened by 25 seconds. In the case of manual method, the camera is kept stationary so that the visual markers are not blurred. In the proposed method, the resting time can be shortened by changing the viewpoint when the collection rate improves.

In the proposed method, the collection time of the five trials tends to decrease. I believe that one of the reasons for the large decrease in the five trials is due to the function of displaying the collection rate. The display of the collection rate shows the progress of the collection status, thus the user can learn how to improve the collection rate efficiently by seeing it as a kind of reward. Displaying the collection rate may accelerate the user's familiarity with the application, and is useful as a method for efficiently collecting the image dataset.

Variability of collected data The plot of the 3D position and orientation data in the 50 images collected in the first trial is shown in Figure 6.7. The plotted points are widely spread, indicating that a high degree of variability in 3D position and orientation has been collected. Therefore, the proposed system enables us to collect unbiased multi-viewpoint image datasets efficiently.