SUGISAKA MASANORI

Journal of Advances in Artificial Life Robotics
Vol. 3(4); March (2023), pp. 230–237
ON LINE ISSN 2435-8061; ISSN-L 2435-8061
https://alife-robotics.org/jallr.html

Research Article

Error Recovery Patterns Focusing on the Revival Process from Failures in Manipulation Tasks

Akira Nakamura[1], Kensuke Harada[2]
[1]*Department of Information Systems, Faculty of Engineering, Saitama Institute of Technology, 1690 Fusaiji, Fukaya, Saitama 369-0293, Japan*
[2]*Robotic Manipulation Research Group, Systems Innovation Department, Graduate School of Engineering Science, Osaka University, 1-3 Machikaneyama, Toyonaka 560-8531, Japan*

ARTICLE INFO

Article History
Received 21 November 2022
Accepted 22 July 2023

Keywords
Planning
Modeling
Revival process of failure
Manipulation skill

ABSTRACT

In recent years, robots have played an important role in various places, including factories of the manufacturing industry as well as homes where people live. The number of robotic tasks with a high degree of difficulty is increasing because they are required to perform various types of tasks, and failures are likely to occur. Therefore, there is a growing demand for error recovery techniques. We propose an error recovery method that considers task stratification and error classification. This allows various error-recovery paths to be derived for a single error. In this study, recovery paths were classified into several patterns to facilitate selection of the optimal path.

1. Introduction

In recent years, the application of artificial intelligence (AI) in engineering has flourished significantly. Therefore, a large wave of AI has been adopted in the field of robotics. In the field of robot planning, many new methods that use AI techniques have emerged in recent years. Consequently, AI techniques are increasingly being applied to tasks that have not been currently targeted as robotic operations.

Although the adaptation of AI methods to new robotic tasks is becoming increasingly popular, many work failures occur. Regarding failures, in the case of repetitive tasks, such as industrial robots operating in plants, there is often an abundance of data, including multiple data on error handling, which can be recovered without problems using AI methods. However, in personal robots that operate at home and industrial robots that perform high-mix, low-volume production, execution tasks are essentially non-repetitive operations. Therefore, in most cases, there are insufficient satisfactory data on error handling, making it difficult to perform recovery using AI methods. The error recovery technique was studied even before recent AI methods became available [1-3]. We devised an error recovery method that considers task stratification and classification for such task errors [4-6]. The main parts of the robotic operation system consist of sensing, modeling, planning, and execution (Fig. 1). If a failure happens during work, the estimation of the cause of the error is performed, classification of errors is done, and system is corrected by modifying the parameters. The process was rerun with improved reliability using a modified system. The error recovery system that we considered runs in a loop, as shown in Fig. 1. While previous research targeted error classification and cause estimation, this study focused on recovery planning. In this study, we classified various paths that appeared while considering a recovery task into several patterns.

Corresponding author's E-mail: akira-nakamura@sit.ac.jp, harada@sys.es.osaka-u.ac.jp

The manipulation of robot tasks has been studied for many years. In particular, the work units of manipulation tasks were researched, and we called the work units "manipulation skills." Focusing on assembly and disassembly tasks, we showed that various tasks can be composed of multiple skill primitives that are units of operation [7-8]. For error recovery, we adopted an approach similar to planning based on the unit of operation [4-6]. In particular, we plan for error recovery by considering the path from the skill primitive, the unit of operation in which the error occurred, to the original sequence of operations.

There are two types of recovery planning patterns: backward and forward. Backward error recovery indicates that, when a failure happens, process returns to the step before the error occurred, and the work is restarted from that step. Conversely, forward error recovery is a process in which even if an error occurs, the process continues forward and returns to the original process with some degree of planning correction.

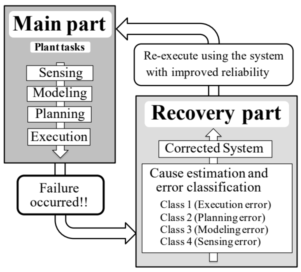


Fig. 1 Robotic system with main part and recovery part

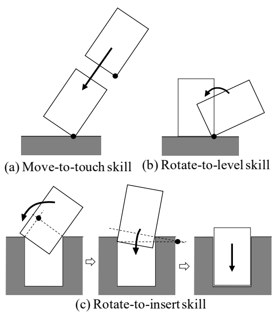


Fig. 2 Elementary skills

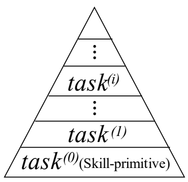


Fig. 3 Task Hierarchies

Currently, the error recovery considered is backward error recovery. In this case, the important points are which step to return and what return path to assume. In this study, the various paths for backward and forward error recovery were organized into several patterns.

The rest of the paper is organized as follows. Section 2 describes the concept of skill primitives in manipulation tasks. In Section 3, we describe our error recovery technique that considers hierarchy of tasks and classes of errors. Section 4 organizes the various error recovery paths into seven patterns, and Section 5 illustrates these examples. Finally, Section 6 presents our conclusions.

2. Concept of skill

This section briefly describes the skills defined as the units of motion for manipulation robots [7-8].

2.1. *Skill primitives*

Skill primitives, which are the action units of work, are derived by analyzing human behavior. A task, such as "assembly," is composed of the three fundamental skill primitives and various resemble skill primitives, where the three fundamental skill primitives are "move-to-touch," "rotate-to-level," and "rotate-to-insert," as shown in Fig. 2.

2.2. *Stratification of tasks*

The robot's operational tasks are handled hierarchically (Fig. 3). The lowest layer, excluding the servo layer, is the skill layer, which consists of motion units, such as the important skill primitives shown in Fig.2.

3. Error recovery

Failures happen during the operation of robotic equipment for various reasons. In manipulation operations in factories where mass production is conducted, it is possible to reduce the occurrence of errors by improving the accuracy and learning functions through repeated operations. However, in the case of manipulation robots that operate in factories for high-mix, low-volume production, or in general households, many errors can occur because non-repetitive tasks are fundamental. Therefore, techniques for recovering from error conditions are necessary for robots operating in such environments. We have conducted various studies on error recovery in manipulation robots [4-6]. This section provides an overview of the concepts of classification for error and our recovery procedures for errors.

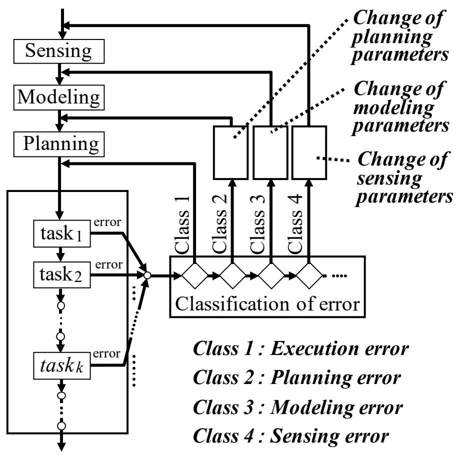


Fig. 4 Elementary flow of error recovery process

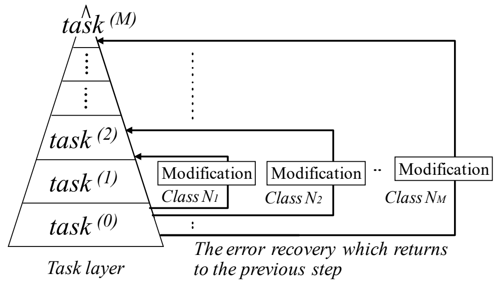


Fig. 5 The representation of task hierarchy and error recovery process flow

3.1. *Error classification*

Errors in robot manipulation operations originate from various sources. The main causes of errors can be categorized into four types: sensing, modeling, planning, and execution (Fig. 4) [4], [5], [6].

3.2. *Error recovery using classification*

When an error occurs, the cause of the error is estimated from the situation, and appropriate corrections are made based on the estimated error cause. There are two types of error recovery: forward error recovery, in which the execution process continues even after an error occurs, and backward error recovery, in which the running process backs to the previous step and the task restarts from that step. In our method, both types of error recovery are considered; however, we employ the backward error recovery-type method [4-6]. In backward error recovery, consideration is based on how far back to the step and what process should be used to return to the original process. Recovery routes are roughly divided into two types, depending on the extent to which the environment and objects have been destroyed. In the case of small errors, the process backs to the previous step of the motion primitives at the lowest level of the task hierarchy, whereas in the case of large errors, it returns to the previous step of the task units at a higher level (Figs. 4 and 5). In the case of sensing, modeling, and planning errors, the system parameters were corrected to reduce the probability of errors in subsequent runs of the same task. Our error recovery method is characterized by its two key performances: it derives a recovery path in the task where the error occurred and decreases the probability of the error occurring in the next task.

4. Various patterns of recovery pathways

As described in Section 3, in the proposed mainstream approach to error recovery, the process is restarted by returning to the step before an error occurred. Specifically, the recovery process comes back to the previous step and attempts to return from that step. However, backward recovery is not the only possible approach. Other approaches, such as a forward recovery process that does not proceed backward are also possible. When a working task fails, it is possible to alter the surrounding environment by collapsing the spatial arrangement of objects or deforming their geometric shapes. Hence, when undoing an error condition, in many cases, the same work process used before the error occurs cannot be used in its original form without modification. Regardless of whether it is backward or forward, partial process reconstruction is required for the process from the occurrence of an error to its return to the original path. Partial sensing and modeling are necessary because the environment will be destroyed, and objects will be deformed.

This section explores various possible recovery procedures, considering the degree of destruction of the surrounding environment and damage to objects. Here, we describe the sequence of the recovery process by organizing it into seven patterns. Here, we refer to them as "Recovery Formulas (RF)" and sequentially abbreviate them as RF-I, RF-II, ..., RF-VII. In planning for error recovery, a two-step approach is adopted: select a feasible pattern among them, and then select a feasible sequence within the selected pattern.

4.1. *RF-I: Precise original repetition*

In this case, if an error causes the environment to change or the object to be deformed, the objects or environments are perfectly restored to its original states, and the operation resumes the same process from the original starting position (Fig. 6(a)). Even if an error occurs, and there is no change in the surrounding environment or objects, the same process is restarted directly to the original starting step.

Note that if objects are deformed owing to an error, it may be necessary to replace the original objects or parts of them with new objects. This object replacement is applicable for this (RF-I) case, as well as for other cases (RT-II to RT-VII). The requirement of a replacement is determined by sensing an object and checking its geometry. Furthermore, because replacement involves costs, such as parts and work expenses, it is necessary to consider the scope of replacement.

4.2. *RF-II: Incomplete original repetition from start step*

Let us consider the case in which an error causes the environment to be different or the objects to be deformed. In this method, instead of returning the environment and objects to their original states, they are returned to a level that does not interfere with the subsequent processes, and the operation resumes in the same process from the original starting step (Fig. 6(b)).

4.3. *RF-III: Original repetition from start step with concealment*

Let us consider the case in which the environment is different or the object is deformed owing to errors. In this method, restoration is performed on the surface by concealing the environment and objects that have been destroyed by an error, and the operation is resumed from the original starting position using the same sequence (Fig. 6(c)). The broken environment and objects will be hidden and will not cause any problems in subsequent work. However, if for some reason the concealed area is brought to the surface, the appearance of the object may be compromised, and the subsequent work progress may be affected. If concealment is discovered in a customer's product, problems, such as a decrease in product value, may occur.

4.4. *RF-IV: Incomplete original repetition from step along the way*

Let us consider the case in which an error causes the environment to be different or the objects to be deformed. In this method, instead of returning the environment and objects to their original states, they are returned to a level that does not interfere with the subsequent process, which is used to resume work from a step in the middle that is not the start (Fig. 6(d)).

4.5. *RF-V: Original repetition from step along the #way with concealment*

Let us consider the case in which the environment is different or the object is deformed owing to errors. In this method, restoration is performed on the surface by concealing the environment and objects that have been destroyed by an error, and the same process is used to resume work from an intermediate step that is not the start (Fig. 6 (e)). The broken environment and objects will be hidden and will not cause any problems in subsequent work. The problem that arises when a concealed area is brought to the surface is the same as that in RF-III.

4.6. *RF-VI: Recovery through different processes*

Let us consider the case in which an error causes the environment to be different or the objects to be deformed. In this method, instead of returning the environment or objects to their original state, they are returned to a level that does not interfere with the subsequent process, and a different procedure from the original is performed to resume work toward the end (Fig. 6(f)). Partial planning is indispensable to deliver a process to either reach its goal or return to a pre-goal step in the midway of the original sequence, while sensing and modeling are associated with partial planning. The final reaching step is the goal of the original sequence. Therefore, if the initially envisioned sequence can be recovered from the middle of the process, the original process can be used and it is unnecessary to create a new plan from the partway of the recovery sequence

4.7. *RF-VII: Continuation of work, including consideration of any recovery process that may be required later*

Let us consider the case in which the environment is different or the object is deformed owing to errors. With this method, the work continues, even if errors occur.

When the work becomes unsustainable, the broken environment and objects are modified, and the work continues (Fig. 6(g)). Local planning is required to implement modifications, sensing, and modeling runs for planning. Forward recovery processes, such as RF-VII, are symmetrical to the backward recovery processes outlined in RF-I through RF-V.

5. Illustrative various examples of patterns of recovery pathways

This section presents examples of the various patterns of recovery pathways described in Section 4.

5.1. *RF-I: Precise original repetition*

An example of RF-1 is considered in "Recovery Type I" in [6]. Errors that occurred in the assembly task, in which the hook was fixed to the plate with screws, were considered. An error is a failure in which a screw detaches and falls off in the middle of a task, but the task completely backs to the starting point, and the task is re-executed. This example does not describe the surrounding environment; however, if that environment changes as a result of work failure, it must be restored to its original state. There are multiple methods to restore the environment and objects to their original states. In particular, there is a cost difference depending on whether the original parts are used when restarting.

5.2. *RF-II: Incomplete original repetition from start step*

In this example, the surrounding environment is not described; however, if the environment changes owing to the failure of the work, it is acceptable if the environment is restored to a level where subsequent work is not affected. It is necessary to completely undo the difference between RF-I and RF-II. The main part of the recovery process was the same as that in the above case, except for matters related to the surrounding environment, as shown in "Recovery Type I" in [6]. In addition, [5] shows an illustration of a failure that affects the placement of surrounding objects and the recovery process.

5.3. *RF-III: Original repetition from start step with concealment*

An example of RF-III is illustrated together with an example of R-V, which is described later.

5.4. *RF-IV: Incomplete original repetition from step along the way*

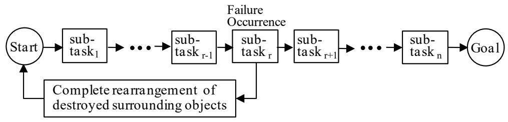
An example of RF-IV is shown in "Recovery Type II" in [6].

5.5. *RF-V: Original repetition from step along the way with concealment*

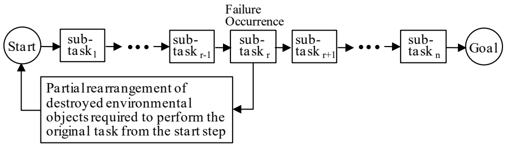
Consider a cake-making operation using the manipulation robot shown in Fig.7. Furthermore, we considered the manufacture of whole cakes suitable for obtaining cut cakes. To obtain evenly divided cakes from a whole cake, it is necessary to place decorations, such as strawberries, precisely with respect to the angle and position of the splitting machines. We will consider an error in which the decoration is misaligned horizontally, as illustrated in Fig. 8. In this case, the restoration work as shown below was performed. Initially, the side-way-shifted decorations were lifted. Next, the dent in the cream created by lifting was filled using a scraper, and the surface was cleaned. At the end of the recovery operation, the decoration returns to its correct position on the cake. Note that decorating the top of the cake with cream before placing ornaments, such as strawberries, as shown in Fig. 9, makes the restoration process more difficult.

One feature of the RF-V error recovery approach is concealing failures from sight. If the recovery process backs to the starting point, and the same sequence can be resumed, the procedure falls under the RF-III error recovery approach. In the cake decoration example, the RF-III error-recovery approach was easy to implement if the first strawberry placement failed. However, if the placement of the second or subsequent strawberry fails, it becomes difficult to return to the initial state.

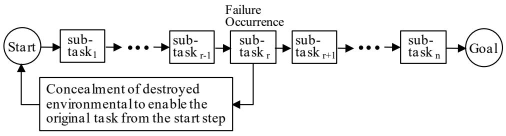
5.6. *Recovery through different processes*



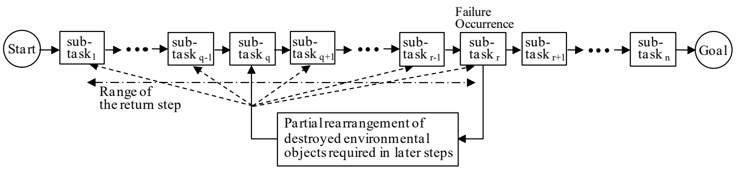
(a) (RF-I) Exact Original Repetition



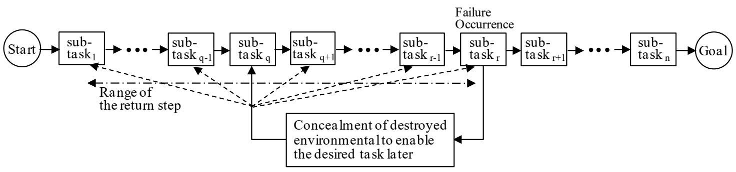
(b) (RF-II) Incomplete Original Reproduction from Start Step



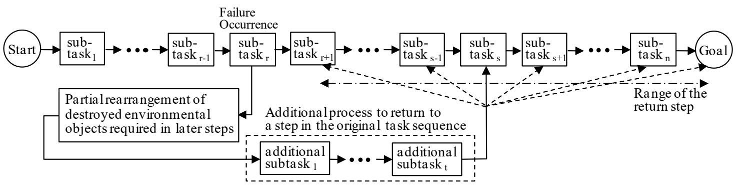
(c) (RF-III) Original Repetition from Start Step with concealment



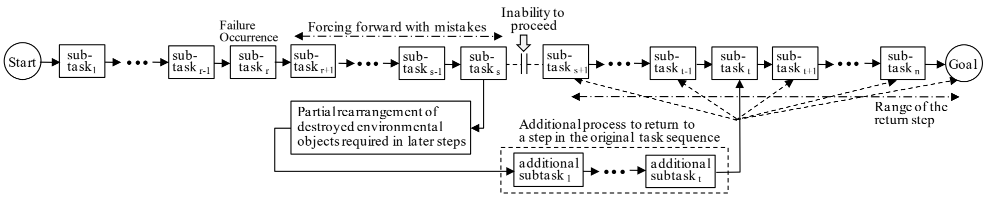
(d) (RF-IV) Incomplete Original Reproduction from Step along the Way



(e) (RF-V) Original Repetition from Step along the Way with concealment



(f) (RF-VI) Recovery through a different process



(g) (RF-VII) Continuation of work, including consideration of any recovery process that may be required later

Fig. 6 Seven patterns of error recovery

An example of RF-VI is the novel recovery process for the task presented in [6]. In this example, an assembly operation in which a hook was fixed to a plate with four screws was considered. As shown in Fig. 10, the recovery operation was done by toppled the object sideways to which the hook was attached such that the temporarily inserted screws were stable. In RF-VI, a work sequence that is separate from the initial sequence is executed to achieve the inital objective. Please note that this approach is not always feasible because it can cause problems, such as the inability to lay objects sideways.

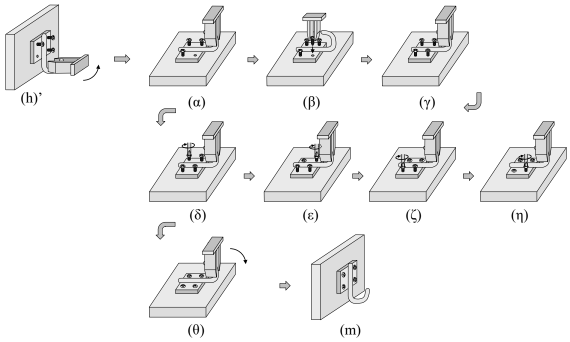


Fig. 10 The recovery task performed by laying the object on its side

5.7. *RF-VII: Continuation of work, including consideration of any recovery process that may be required later*

An example of RF-VII is shown in "Recovery Type

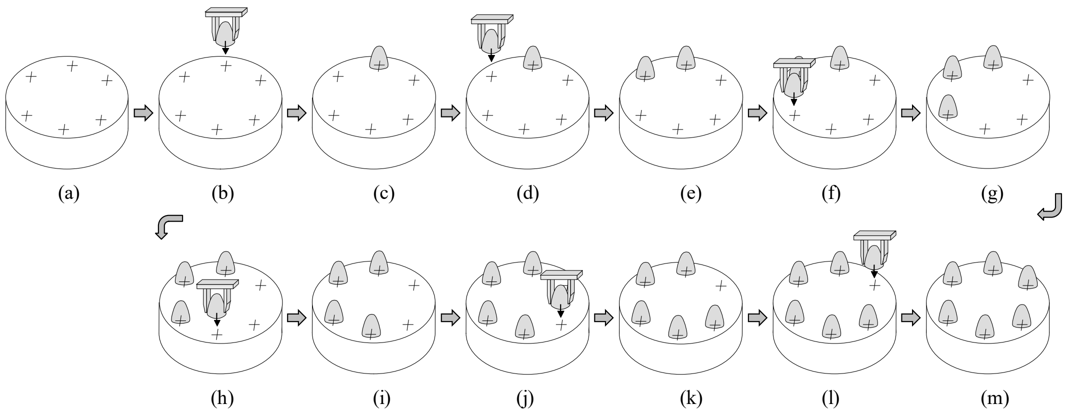


Fig. 7 Cake-making task

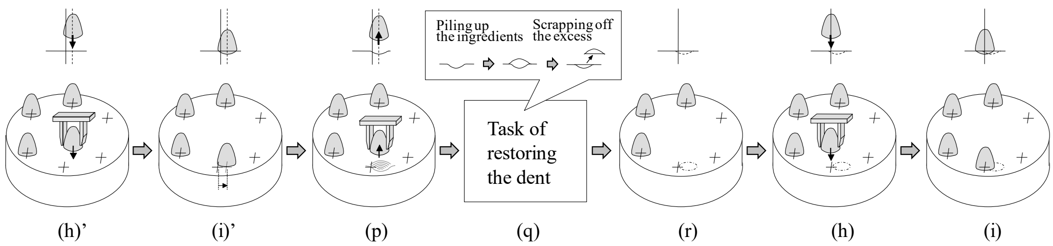


Fig. 8 Error and recovery process in cake-making task

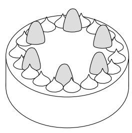


Fig. 9 Whole Cake with Cream Decoration

III" in [6], where a forward recovery process is considered.

6. Conclusion

Non-repetitive tasks with manipulation robots are prone to errors. This is because data on improvements in accuracy and the effects of learning through repetition were unavailable. We developed a procedure of error recovery that considers stratification of task and classification of error. This causes numerous error recovery processes. In this study, the various paths that appeared when considering a recovery task were classified into seven patterns and organized.

There are various paths for the recovery process when an error occurs, and there are different ways to choose the path. Future work will involve understanding the circumstances under which an error occurs to guide the determination of the optimal recovery path. Although the method for selecting recovery paths varies depending on the robot system in operation, it is important to derive a systematic method for selection in a unified system.

References

1. B. R. Donald, Planning multi-step error detection and recovery strategies, Int. J. Robot. Res., 9(1) (1990) 3-60.
2. E. D. Lello, M. Klotzbucher, T. D. Laet and H. Bruyninckx, Bayesian Time-Series Models for Continuous Fault Detection and Recognition in Industrial Robotics Tasks, in Proc. IEEE/RSJ Int. Conf. on Intell. Robots Syst., (Tokyo, Japan, 2013), 5827-5833.
3. A. S. Wang and O. Kroemer, Learning Robust Manipulation Strategies with Multimodal State Transition Models and Recovery Heuristics, in Proc. IEEE Int. Conf. Robot. Autom., (Montreal, Canada, 2019), 1309-1315.
4. A. Nakamura, K. Nagata, K. Harada, Y. Domae, Using Importance Ranks to Derive Suitable Timing of Visual Sensing in Manipulation Task Containing Error Recovery, J. Robotics, Networking and Artificial Life, 5(4), 2019, pp. 283-288.
5. A. Nakamura, K. Nagata, K. Harada, N. Yamanobe, T. Tsuji, T. Foissotte and Y. Kawai, Error recovery using task stratification and error classification for manipula-tion robots in various fields, in Proc IEEE/RSJ Int. Conf. on Intell. Robots Syst., (Tokyo, Japan, 2013), 3535-3542.
6. A. Nakamura, N. Yamanobe, I. R. Alpizar, K. Harada and Y. Domae, Selection of Optimal Error Recovery Process using Evaluation Standards in Automated Plants, J. Robotics, Networking and Artificial Life, 8(3), 2021, pp. 211-217.
7. T. Hasegawa, T. Suehiro and K, Takase, A robot system for unstructured environments based on an environment model and manipulation skills, in Proc. IEEE Int. Conf. Robot. Autom., (Sacramento, USA, 1991), 916-923.
8. A. Nakamura, K. Kitagaki and T. Suehiro, Using simplified geometric models in skill-based manipulation, Advanced Robotics, 18(8), 2004, pp. 835-858.

Authors Introduction

Prof. Akira Nakamura

9. He received the Ph.D. degree in Electrical Engineering from Keio University in 1991. From 2021, he has been working as a Professor at Faculty of Engineering of Saitama Institute of Technology.

Prof. Kensuke Harada

0. He received his Doctoral degrees in Mechanical Engineering from Kyoto University in 1997. From 2016, he has been working as a Professor at Graduate School of Engineering Science, Osaka University.