2024 IEEE/SICE International Symposium on System Integration (SII)
January 8-11, 2024. Ha Long, Vietnam

Real-Time Failure/Anomaly Prediction for Robot Motion Learning Based on Model Uncertainty Prediction

Hideyuki Ichiwara[1], Hiroshi Ito[1] and Kenjiro Yamamoto[1]

***Abstract*— End-to-end robot motion generation methods using deep learning have achieved various tasks. However, due to insufficient training or the occurrence of abnormal conditions, the model sometimes fails tasks unexpectedly. If failures/anomalies can be predicted before occurring, irreversible task failures can be prevented. In this paper, we propose a method of predicting model uncertainty to predict failures/anomalies in real-time. For a naive method, we used a model that predicts the robot's actions stochastically and also tried a method that predicts failure/anomaly on the basis of the variance. However, it was experimentally shown that the variance due to the variation of the training data and the uncertainty of the model cannot be distinguished. Therefore, by predicting the likelihood of the model, which corresponds to the degree of discrepancy between the model and observations, in real-time and treating it as the uncertainty of the model, we applied it to the prediction of failure/anomaly. The method's effectiveness was demonstrated by achieving a high judgment accuracy rate of 85% (17/20 cases) in an object-picking task.**

I. Introduction

End-to-end robot motion generation methods using deep learning have realized various manipulation tasks. For example, there are methods that use reinforcement learning (RL) [1][2][3] and methods that learn from demonstration data[4][5][6]. They use multiple modalities as input to the deep neural network (DNN), then output the action that the robot should take in real-time. However, the model failed tasks unexpectedly due to insufficient training of the model or the occurrence of abnormal conditions. Task failures can be reversible or irreversible, such as damage to the target object or robot. In particular, in the irreversible case, the target object or the jig needs to be replaced, which requires time and costs for recovery. Therefore, task failure needs to be predicted and prevented. Judging failure/anomaly in robot manipulation is not easy because there are many cases other than those that are clearly judged from torque/force sensor information, such as overload. Therefore, several methods have been proposed to judge task failure/anomaly [7][8][9][10]. However, these methods focus on judging only from observations, so failures are difficult to prevent.

Failures may be able to be predicted by removing the black boxness of DNNs. There are some methods for improving the interpretability of action selection norms through hierarchical action generation [11][12] and for improving visual interpretation by introducing visual attention [1][13]. These can improve the condition by allowing humans to interpret the reason for a failure after it occurs. However, to predict failures in real-time, it is necessary to judge whether each judgment rule is inappropriate, but the criteria are not self-evident.

In this paper, we focus on the intrinsic uncertainty of the model that generates the action as an interpretable quantity necessary for failure/anomaly prediction. We propose a method to stochastically predict actions that are often deterministically predicted to handle the uncertainty of the actions themselves. As a method of treating actions stochastically, we use a stochastic-continuous time recurrent neural network (S-CTRNN)[14] as a base model, which predicts the variance and mean of actions. For a naive method, we try to predict failures/anomalies on the basis of the variance predicted by S-CTRNN. However, we experimentally show that it is not possible to distinguish between the variance caused by the variability of the training data and the variance caused by the uncertainty of the model. Therefore, we propose a method for predicting the likelihood of the model during motion generation by obtaining the predicted value of the prediction error in the stochastic model and then judging the failure/anomaly on the basis of that value. Our main contributions are as follows: 1) Development of a real-time failure/anomaly prediction method on the basis of the likelihood prediction of motion generation models. 2) Experimental demonstration that failure/anomaly prediction is difficult when using conventional probabilistic models because prediction variance obtained from the models predominantly includes variation in training data. 3) Evaluation of the effectiveness of our method in a picking task using a real robot.

II. Related Work

A. Autonomous robot manipulation using deep learning

Related work consists of methods using RL [1][2][3] and methods learning from demonstration data [4][5][6]. RL requires a lot of trial and error, and many failures occur during not only execution but also learning. There are also methods that improve sample efficiency by applying model-based methods, such as world models [15][16], in combination with learning of the dynamics of the environment, even in environments where conventional model-free RL has been performed. However, even a simple pick&place still requires tens of thousands of trials and errors, which entails the risk of damage to the object or the robot itself. In contrast, deep predictive learning (DPL), a method that learns from demonstration data, does not require trial and error and is learned in a supervised learning framework. It has realized the manipulation of flexible objects [17][18] and complex long-term tasks consisting of multiple subtasks [19][20],

[1]Hideyuki Ichiwara, Hiroshi Ito and Kenjiro Yamamoto are with Research & Development Group, Hitachi, Ltd., Ibaraki, 312-0034, Japan hideyuki.ichiwara.bn@hitachi.com

2024 IEEE/SICE International Symposium on System Integration (SII) | 979-8-3503-1207-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/SII58957.2024.10417333

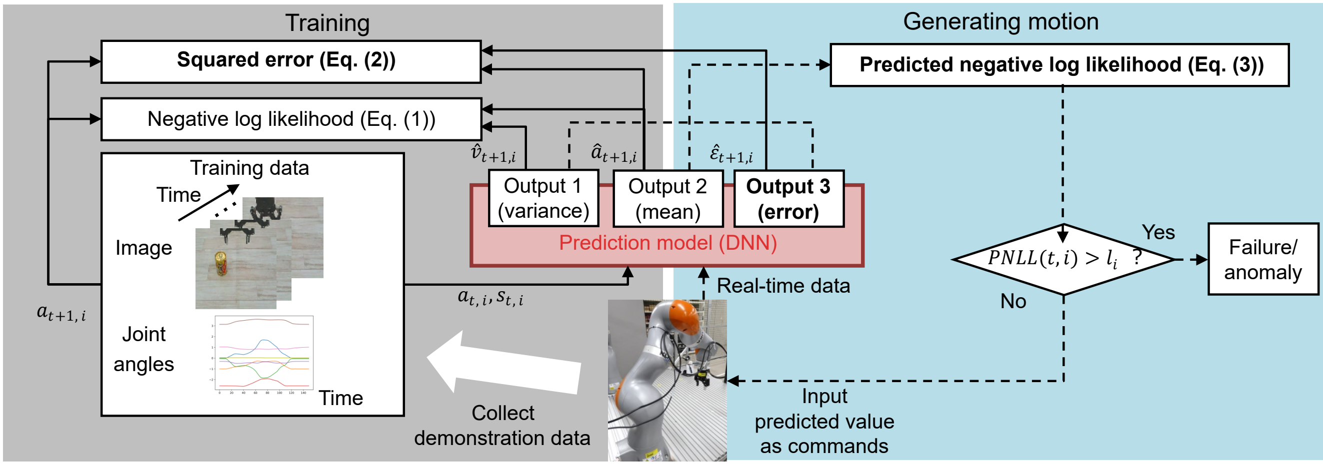

Fig. 1. Architecture of our method for predicting model uncertainty. (Left) Data flow when training. (Right) Data flow when generating motion and predicting model uncertainty. While stochastically predicting actions on the basis of the minimization of the negative log-likelihood, learning to predict the squared error of the action and predicting during inference makes it possible to predict the negative log-likelihood as well. We used the predicted negative log-likelihood to predict failures/anomalies.

which were difficult to achieve with conventional rule-based motion generation methods. Such methods use deterministic models, and unique actions can be obtained for input sensor information. Therefore, even in situations where prediction is uncertain, failures/anomalies are difficult to predict from actions because unique actions can be obtained in the same way as low-uncertainty situations. On the other hand, some stochastic models [10][14][18] for predicting the variance of actions have also been proposed, and it is expected that the uncertainty of actions can be judged from the variance.

B. Real-time failure/anomaly detection for manipulation

There are studies that generate motion and judge at the same time. Suzuki et al. [21] proposed generating recovery motions by predicting past motion trajectories in real-time to successfully determine whether situations not included in training data have occurred. For example, the scene is given where the robot is held by hand and the posture of the robot is changed by applying an external force. Although the method can judge external interference after the fact, it has difficulty predicting failures/anomalies due to a lack of training data and anomalies that occur in the initial time of a task, which are related to the uncertainty of the internal model. Hung et al.[10] proposed judging failure from the distribution of actions using a Bayesian neural network (BNN). Because this method determines whether a task has failed and focuses on recovery actions, failures/anomalies cannot be predicted in advance. Furthermore, the dispersion of actions is also caused by variations in training data, but they did not mention that. The purpose of this paper is to predict failures/anomalies on the basis of the uncertainty prediction of the model to prevent failures while considering the variation in training data.

III. Method

A. Concept

Basically, robot actions and interactions with the environment result in task failures. Furthermore, when abnormal conditions such as external disturbances or environments different from those during learning occur, and observations differ from those assumed by the model, uncertainty arises in actions. Therefore, we thought that failures, including abnormal situations, could be predicted if we knew the uncertainty of the model that generated the actions. In this paper, to deal with the uncertainty of the actions themselves, we deal with the stochastically predicted actions. For a method of treating actions stochastically, we use S-CTRNN[14] as a base model, which predicts the variance and mean of actions by learning based on likelihood maximization. As will be described later, it is not possible to predict only from the variance. This is because the variance is due to the variability of the training data and the uncertainty of the model due to how out-of-distribution the observations are. Failure/anomaly prediction requires knowing the latter. Therefore, the likelihood of the model, which corresponds to the degree of discrepancy between the model and the observation, is newly predicted in real-time and treated as the uncertainty, which is applied to the prediction of failures/anomalies.

B. Architecture

Fig. 1 shows the overall structure of our method. The solid and dotted lines indicate the data flow during learning and motion generation, respectively. Training data is sensor data such as camera images and data related to motion commands such as joint angles and torques (actions). The time is $0 \sim T$, the action is an array with the dimension I_a, the other sensor data is an array with the dimension I_s, and the elements are denoted by $a_{t,i}$ and $s_{t,i}$, respectively. Using the training data, the model learns to predict the sensor information at the next time $t+1$ from those at time t. In a deterministic model, the value of the training data $a_{t,i}, s_{t,i}$ at time t is input to the model, the output $\hat{a}_{t+1,i}, \hat{s}_{t+1,i}$ is obtained, and the model is trained by using the squared error from the value of the training data $a_{t+1,i}, s_{t+1,i}$ at time $t+1$ as the loss function. In our method, we add the prediction variance $\hat{v}_{t+1,i}$ and prediction error of the action $\hat{\epsilon}_{t+1,i}$ as the output of the model by learning on the basis of likelihood maximization. Similar to S-CTRNN, negative log-likelihood (NLL) is used as a loss function to obtain prediction variance.

$$\mathrm{NLL} := \sum_{i \in I_a} \sum_{t \in T} \left\{ \frac{\ln(2\pi \hat{v}_{t+1,i})}{2} + \frac{(a_{t+1,i} - \hat{a}_{t+1,i})^2}{2\hat{v}_{t+1,i}} \right\}, \tag{1}$$

Note that a normal distribution is assumed, and the purpose of minimization is to use NLL instead of the likelihood. Also, by using Eq. (1) as the loss function, it is possible to learn while allowing variations in the training data. This is because, assuming that the learning data varies at a certain time, the squared error increases near that time, but NLL can be reduced by increasing the prediction variance $\hat{v}_{t+1,i}$ at that time. Furthermore, we consider that the larger the NLL, the greater the uncertainty of the model, and we predict the NLL at the time of inference and use it for failure/anomaly prediction. NLL consists of the prediction variance $\hat{v}_{t+1,i}$, the squared error between prediction mean $\hat{a}_{t+1,i}$ and the true value $a_{t+1,i}$. However, since the true value and the squared error cannot be obtained during inference, the NLL cannot be calculated. Therefore, we propose to predict the NLL (PNLL) shown in Eq. (3) by predicting the error $\hat{\epsilon}_{t+1,i} = y_{t+1,i} - \hat{y}_{t+1,i}$ using Eq. (2) as the loss function.

$$\mathrm{SE} := \sum_{i \in I_a} \sum_{t \in T} \left\{ (a_{t+1,i} - \hat{a}_{t+1,i})^2 - \hat{\epsilon}^2_{t+1,i} \right\}, \tag{2}$$

$$\mathrm{PNLL}(t,i) := \frac{\ln(2\pi \hat{v}_{t+1,i})}{2} + \frac{\hat{\epsilon}^2_{t+1,i}}{2\hat{v}_{t+1,i}}, \tag{3}$$

In DPL, other modalities are also predicted as subtasks to prevent overfitting to the action, but they are omitted in the figure. When generating motion, real-time sensor information is input to the model, and the predicted commands, such as joint angles and torque, are sent to the robot as the command. At the same time, PNLL is calculated and determined as a failure/anomaly when it exceeds the threshold l_i. The threshold l_i uses the maximum value of PNLL that is calculated for the training data.

IV. Experiments

A. Experimental setup

Fig. 2 shows the evaluation experiment environment for a real robot. The robot arm was KUKA LBR iiwa 7 R1400, and the gripper was Robotiq 2F-85 Adaptive Grippers. The camera used Intel's RealSense L515 and was installed in front of the arm so that the work environment could be captured. The experimental task was an object-picking task as a general task. Nine teaching positions were selected at intervals of 7 cm, motion data were collected three times at each point, and a total of 27 data were used for learning. The acquired training data was 10 seconds per case and was acquired in periods of 0.1 seconds (10 Hz). Learning data was collected by manipulating the robot arm by using a game controller to control the position of the end effector. The network configuration is based on [21], which is a combination of convolutional autoencoder (CAE) and long-short term memory (LSTM), as one of the general models. Image feature values are extracted by CAE and integrated with joint angles by LSTM. We trained two models, S-CTRNN and our method, and compared the judgment results.

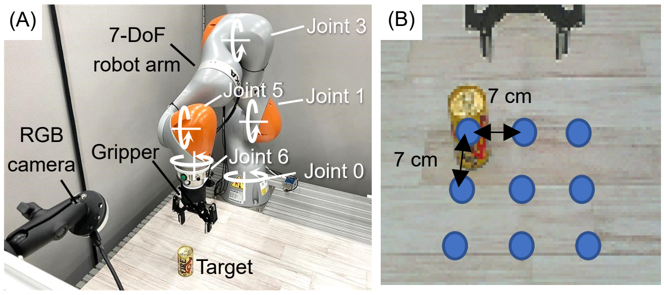

Fig. 2. Experimental setup. (A) Overview of an object-picking task. (B) Taught positions of the target object.

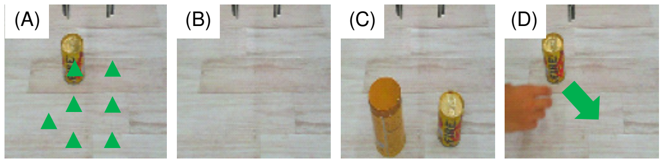

Fig. 3. Experimental conditions. (A) Untaught positions. △ are positions for the experiment. (B) Without target object. (C) With non-target object. (D) Dynamic obstruction.

We set a total of 20 cases, including the case where the target is at the taught position, where the task is expected to succeed, and four patterns (Fig.3) where failure is assumed. In addition, the robot generates motions using each model and examines whether variance and PNLL can be used to predict failures/anomalies. A) Untaught position: Assuming a case of failure due to insufficient learning. B) When there is no object: Assuming that the work environment is different from the assumption. C) When there are non-target items: Assuming cases where there are non-target items or obstacles. D) When the object is moved during work: Assuming that the work will be obstructed from the outside or that the work will collide with an obstacle. Furthremore, to evaluate the failure/anomaly prediction in the experiment, the action was generated even after the failure/anomaly was predicted.

B. Results

1) S-CTRNN: We found that the variances of task success and failure were the same, making failures/anomalies difficult to predict. As an example, Fig. 4(A) shows the predicted variance of the gripper closing degree when the task succeeds at the taught position and when the task fails at the untaught position. In both cases, the variance increased

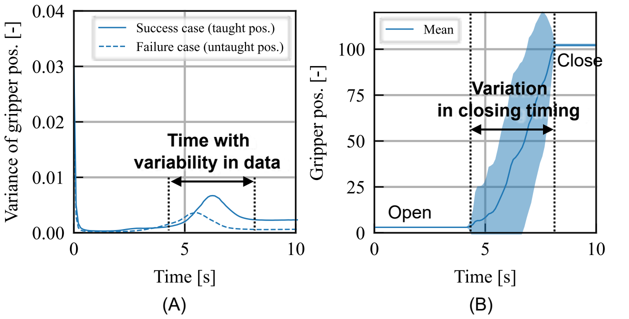

Fig. 4. Experimental results of S-CTRNN. (A) The predicted variance of gripper position. (B) Distribution of gripper position in the training data. Predicted variance mainly represents variation of training data.

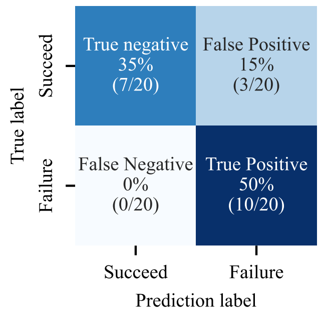

Fig. 5. Confusion matrix with our method. Accuracy rate is high overall (85%), and the rate of false negatives, which can cause task failure due to misjudgment, is of 0%.

markedly at the initial time and again around 6.0 s. The reason the variance is large at the initial time is that the context state of the LSTM is initialized to 0 at the initial time regardless of the position of the object, and the prediction variance is large. As time passes, the variance also decreases because the context state converges in accordance with the situation such as the position of the object. By changing the context state in accordance with the position of the object, the variance can be reduced even at the initial time. On the other hand, the increase in variance around 6.0 s was thought to be another cause. Fig. 4(B) shows the average gripper closing degree and the average ± standard deviation of the gripper closing degree of the training data. The horizontal and vertical axes indicate time and the degree of closure, respectively. The larger the value, the closer the gripper. Since the gripper closes when the object position is reached, the closing time differs depending on the distance from the initial position to the object. Also, even if the position of the object is the same, the timing to close the gripper differs depending on the trial because a person is teaching. In the training data of this experiment, as shown in Fig. 4(B), the gripper closing timing differs in the range of about 4.0 to 8.0 s, and the variance also increases in that range. In Fig. 4(A), the reason the prediction variance is large around 6.0 s is that the variance caused by the variation of the training data is dominantly learned. If there is no variation in the training data, even with S-CTRNN, the prediction variance is expected to increase due to the prediction uncertainty from the observation of failures/anomalies, and failures will be able to be distinguished from successes. However, this narrows the work range that can be handled. Also, data without variations in motion is difficult to obtain when a human gives instructions. Furthermore, although we omit the details, we found that similar results can be obtained with other stochastic models (Softmax transformation[18], BNN[10]) that can predict actions stochastically. Failures/anomalies are difficult to predict simply by stochastically predicting actions because it is not possible to distinguish between variation in training data and uncertainty in the model.

2) Our method: Fig. 5 shows the confusion matrix of failure/anomaly prediction results by our method. There were 10 success cases (negative) and 10 failure cases (positive). Our method achieved a true positive rate of 50% and a true negative rate of 35% for an overall accuracy rate of 85%. Furthermore, the false positive rate was 15%, while the false negative rate was 0%. This method can make conservative judgments while achieving a high accuracy rate. Therefore, the risk of damage due to false negatives can be reduced while suppressing the decrease in work throughput due to false positives. In the following, we show some examples and discuss judgment timing and judgment grounds.

Fig. 6(A1), (B1) show the failure case with an untaught object position. (A1) shows snapshots, and the images are the same as the input to the model. (B1) shows PNLL for each joint. The solid blue line indicates PNLL, and the dotted red line indicates the threshold, which is the criterion for failure/anomaly. The dimension of action is eight, including the seven degrees of freedom (7-DoF) joints and the gripper. PNLLs are shown about only joints with large movements in this task, and the positions and drive directions of each joint are shown in white in Fig. 2(A). In addition, in Fig. 6(B1), the direction in which the joint contributes to the motion in the image is described in parentheses. For example, joint0 (hereafter j0) is used to move the hand horizontally (left and right) in the image. In Fig. 6(B1), PNLL of j0 and j1 exceeded the threshold at time 4.0 s and 7.0 s, respectively. In this task, j0 is first moved to adjust the horizontal position. By adjusting the depth direction using j1, j3, and j5, the robot approaches and grasps the object. Therefore, it is thought that PNLL of j0 exceeded the threshold before the robot clearly failed the task because the prediction confidence was low when adjusting the horizontal position. After that, the prediction confidence was also low when adjusting the depth position, and PNLL of j1 exceeded the threshold. It was found that our model can judge that a task is likely to fail before clearly failing. Fig. 6(A2),(B2) shows the success case with an untaught object position. In judgment processing based on naive image recognition, the fact that an object is in an untaught position itself can be regarded as an abnormal state and can be misjudged. However, in our method, no PNLL of any joint exceeded the threshold or was misjudged. This is because the model generalizes to a few positional changes and can predict actions with a high certainty for untaught positions in this case. Therefore, it was

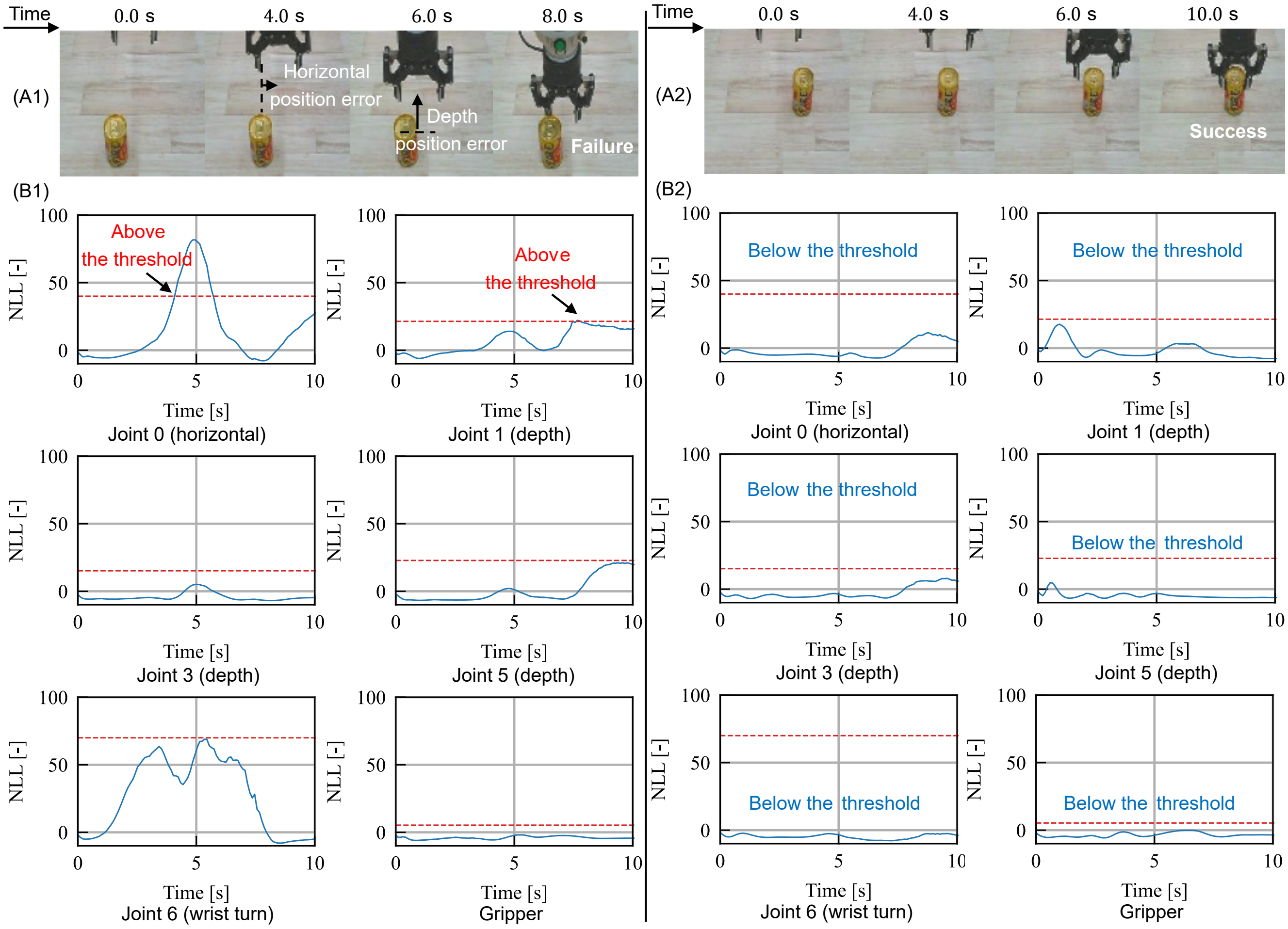

Fig. 6. Snapshots of the picking task with the untaught position. (A1), (B1): Input images and PNLL on failure. Failure-related joint angle PNLL increased and predicted failure. (A2), (B2): Input images and PNLL on success. All PNLLs were less than the threshold.

found that the model can correctly judge states that differ from those during training, such as changes in the position of the object, which can be generalized by the model. Fig. 7 shows the failure case with dynamic obstruction. There were disturbances at 5.0 and 7.0 s, but at both times, PNLL of j0 exceeded the threshold. It was found that the model can respond to dynamic changes in the external environment and predict in real-time. However, there were three cases when the robot succeeded in the task but was predicted to be a failure. This may be because PNLL corresponds to the uncertainty and not to the success or failure of the task. In such cases, it indicates that the model's prediction confidence is low and the need to take measures such as relearning or increasing the training data. This method can be applied to model quality control.

V. Conclusion

In this paper, we proposed a method to predict the uncertainty of the model to predict failures/anomalies in real-time in the autonomous motion generation of the robot. As a result of verifying the effectiveness of this method in item picking work using a real robot, it was found that failures/abnormalities can be predicted in real time before they actually occur. Furthermore, not only the taught range but also the untaught range generalized from the teaching can be considered and predicted. Next, we aim to generate exploratory behavior on the basis of model uncertainty and to generate recovery behavior on the basis of judgment.

Acknowledgement

Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.

References

[1] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel, "Deep spatial autoencoders for visuomotor learning," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 512–519.

[2] Y. Tsurumine, Y. Cui, E. Uchibe, and T. Matsubara, "Deep reinforcement learning with smooth policy update: Application to robotic cloth manipulation," *Robotics and Autonomous Systems*, vol. 112, pp. 72–83, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889018303245

[3] H. van Hoof, N. Chen, M. Karl, P. van der Smagt, and J. Peters, "Stable reinforcement learning with autoencoders for tactile and visual data," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 3928–3934.

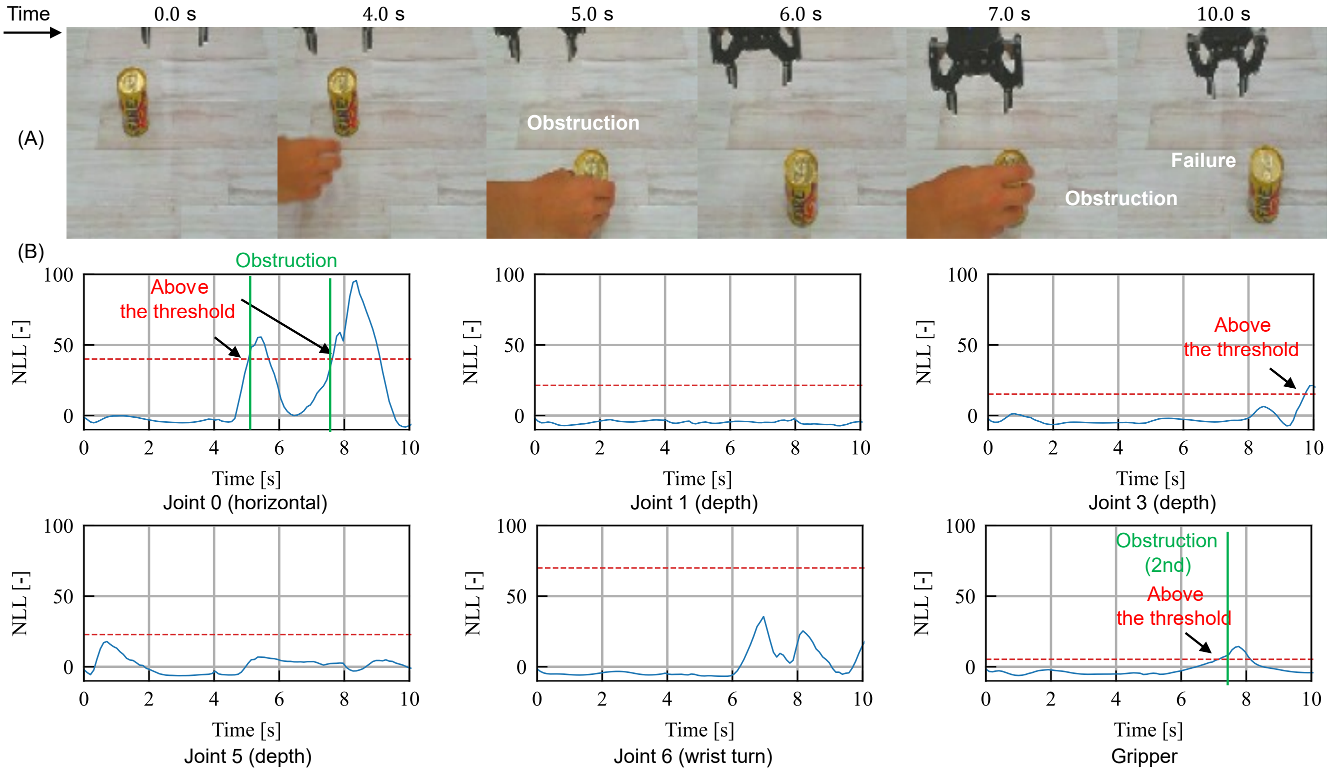

Fig. 7. Snapshots of the task with dynamic obstruction. (A) Input images. (B) PNLL. Obstructed 5.0 s and 7.5 s. Each time, PNLL increased in j0 associated with misalignment, predicting failure.

[4] K. Noda, H. Arie, Y. Suga, and T. Ogata, "Multimodal integration learning of robot behavior using deep neural networks," *Robotics and Autonomous Systems*, vol. 62, no. 6, pp. 721–736, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889014000396

[5] T. Adachi, K. Fujimoto, S. Sakaino, and T. Tsuji, "Imitation learning for object manipulation based on position/force information using bilateral control," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 3648–3653.

[6] H. Kim, Y. Ohmura, and Y. Kuniyoshi, "Transformer-based deep imitation learning for dual-arm robot manipulation," *CoRR*, vol. abs/2108.00385, 2021. [Online]. Available: https://arxiv.org/abs/2108.00385

[7] A. Inceoglu, E. E. Aksoy, A. C. Ak, and S. Sariel, "Fino-net: A deep multimodal sensor fusion framework for manipulation failure detection," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 6841–6847.

[8] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1544–1551, 2018.

[9] A. Alvanpour, S. K. Das, C. K. Robinson, O. Nasraoui, and D. Popa, "Robot failure mode prediction with explainable machine learning," in *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*, 2020, pp. 61–66.

[10] C.-M. Hung, L. Sun, Y. Wu, I. Havoutis, and I. Posner, "Introspective visuomotor control: exploiting uncertainty in deep visuomotor control for failure recovery," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6293–6299.

[11] B. Beyret, A. Shafti, and A. A. Faisal, "Dot-to-dot: Explainable hierarchical reinforcement learning for robotic manipulation," in *2019 IEEE/RSJ International Conference on intelligent robots and systems (IROS)*. IEEE, 2019, pp. 5014–5019.

[12] D. Zhang, Q. Li, Y. Zheng, L. Wei, D. Zhang, and Z. Zhang, "Explainable hierarchical imitation learning for robotic drink pouring," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3871–3887, 2022.

[13] A. Y. Yasutomi, H. Ichiwara, H. Ito, H. Mori, and T. Ogata, "Visual spatial attention and proprioceptive data-driven reinforcement learning for robust peg-in-hole task under variable conditions," *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1834–1841, 2023.

[14] S. Murata, J. Namikawa, H. Arie, S. Sugano, and J. Tani, "Learning to reproduce fluctuating time series by inferring their time-dependent stochastic properties: Application in robot learning via tutoring," *IEEE Transactions on Autonomous Mental Development*, vol. 5, no. 4, pp. 298–310, 2013.

[15] P. Wu, A. Escontrela, D. Hafner, K. Goldberg, and P. Abbeel, "Daydreamer: World models for physical robot learning," *arXiv preprint arXiv:2206.14176*, 2022.

[16] T. Taniguchi, S. Murata, M. Suzuki, D. Ognibene, P. Lanillos, E. Ugur, L. Jamone, T. Nakamura, A. Ciria, B. Lara *et al.*, "World models and predictive coding for cognitive and developmental robotics: frontiers and challenges," *Advanced Robotics*, pp. 1–27, 2023.

[17] K. Suzuki, M. Kanamura, Y. Suga, H. Mori, and T. Ogata, "In-air knotting of rope using dual-arm robot based on deep learning," *CoRR*, vol. abs/2103.09402, 2021. [Online]. Available: https://arxiv.org/abs/2103.09402

[18] H. Ichiwara, H. Ito, K. Yamamoto, H. Mori, and T. Ogata, "Contact-rich manipulation of a flexible object based on deep predictive learning using vision and tactility," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 5375–5381.

[19] H. Ito, K. Yamamoto, H. Mori, and T. Ogata, "Efficient multitask learning with an embodied predictive model for door opening and entry with whole-body control," *Science Robotics*, vol. 7, no. 65, p. eaax8177, 2022. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.aax8177

[20] H. Ichiwara, H. Ito, K. Yamamoto, H. Mori, and T. Ogata, "Multimodal time series learning of robots based on distributed and integrated modalities: Verification with a simulator and actual robots," in *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2023.

[21] K. Suzuki, H. Mori, and T. Ogata, "Compensation for undefined behaviors during robot task execution by switching controllers depending on embedded dynamics in rnn," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3475–3482, 2021.