2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Learning Food Picking without Food: Fracture Anticipation by Breaking Reusable Fragile Objects

Rinto Yagawa[1], Reina Ishikawa[1], Masashi Hamaya[2], Kazutoshi Tanaka[2], Atsushi Hashimoto[2], Hideo Saito[1]

***Abstract*— Food picking is trivial for humans but not for robots, as foods are fragile. Presetting foods' physical properties does not help robots much due to the objects' inter- and intra-category diversity. A recent study proved that learning-based fracture anticipation with tactile sensors could overcome this problem; however, the method trains the model for each food to deal with intra-category differences, and tuning robots for each food leads to an undesirable amount of food consumption. This study proposes a novel framework for learning food-picking tasks without consuming foods. The key idea is to leverage the object-breaking experiences of several reusable fragile objects instead of consuming real foods while making the picking ability object-invariant with domain generalization (DG). In real-robot experiments, we trained a model with reusable objects (toy blocks, ping-pong balls, and jellies), selected based on the three common fracture types (crack, rupture, and crush). We then tested the model with four real food objects (tofu, bananas, potato chips, and tomatoes). The results showed that the proposed combination of reusable objects' breaking experiences and DG is effective for the food-picking task.**

I. Introduction

There is increasing demand for automation of food handling in food industries [1], agriculture [2], and cooking [3]. Robots are required to handle various food objects to adapt to rapid product changes in factories [1] and restaurants.

We are interested in robotic food picking, a non-trivial task, as food objects are often fragile. The physical properties of food objects should be considered for fracture-free picking. However, physical properties, such as stiffness, friction, and fracture force, differ significantly between objects (inter-food differences). In addition, the physical properties of the same type of ingredient may differ [4] due to natural fluctuations, plant maturation, localized variations, or different regions of origin (intra-food differences).

In a previous study, we demonstrated that learning food properties is essential for fracture-free picking [5]. We leveraged the breaking experiences of food objects to obtain food properties. During the training, a robot equipped with a universal two-finger gripper and tactile sensors broke the objects and measured the tactile signals, which describe when the fractures occur. We then trained a *fracture anticipation* network, which anticipates the timing of the object's future fracture. Fracture anticipation enables fracture-free picking by stopping the gripper motion immediately before the object breaks. According to the experimental results of online object

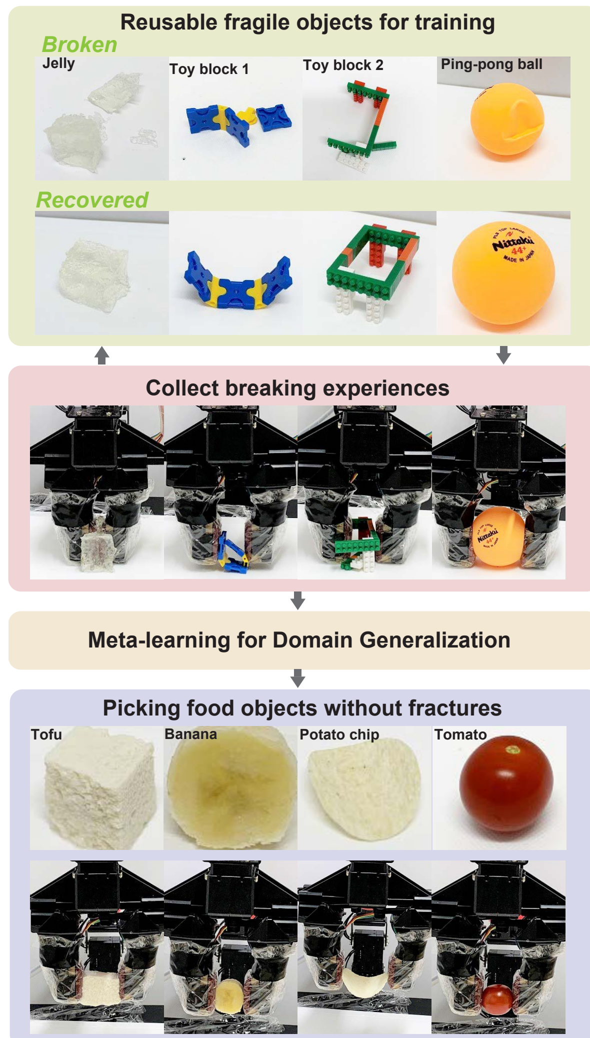

Fig. 1. The proposed framework. We allow the robot to explore reusable objects by loading them to the point of breakage. After training, the object can be grasped safely by anticipating the fracture point.

picking, the fracture anticipation method outperformed the fixed threshold method for all three test objects: tofu, potato chips, and bananas.

The previous method clarified that we could achieve food picking through object-breaking experiences; however, the method assumes a category-specific model and requires breaking many food objects in the category to train the model. For practical use, the network should be able to anticipate the fracture of unseen food objects. Furthermore, the amount of food consumption required for training is

*This work was supported by KAKENHI Grant Numbers JP19K14936 and JP21H04910.

[2] RY, RI, and HS are with Keio University, Yokohama 223-8522, Japan.
[1] MH, KT, and AH are with OMRON SINIC X Corporation, Tokyo 113-0033, Japan. masashi.hamaya@sinicx.com

economically and environmentally undesirable.

In this paper, we propose a method for making a robot grasp multiple unseen food categories without using a category-specific model trained by consuming a considerable amount of food in that category. For the proposed method, we train the model without food but with fragile non-food objects that can be reused many times to obtain training data. We assume that we can find fragile non-food objects with breaking properties similar to those of the target foods. In addition, to compensate for the gap between the reusable non-food objects and the target real food objects, we employ a domain generalization (DG) technique using meta-learning [6]. As DG maximizes the model's generalization ability with the help of diverse training data, the network can anticipate the fracture of unseen food objects without additional training.

We performed real-robot experiments to verify the proposed method. We selected two toy blocks, ping-pong balls, and jellies as the reusable fragile objects for training and tofu, bananas, potato chips, and tomatoes as the unseen food objects for testing. Note that the training objects are easily reused by rebuilding the blocks, heating the ping-pong balls, and melting and cooling the jellies.

Our contributions are as follows:

- We train the fracture anticipation network for food picking by breaking reusable fragile objects, which can avoid breaking the target food objects.
- We employ meta-learning for domain generalization (MLDG) that can handle unseen food objects. MLDG improves the picking performance for most of the target food objects.

The paper is organized as follows. In Section II, we introduce related works. In Section III, we describe the preliminaries. In Section IV, we explain the proposed method. In Sections V and VI, we present the experiments and the discussion, respectively. Finally, in Section VII, we conclude.

II. Related Works

In this section, we introduce related works. We focus on 1) food picking, 2) grasping with tactile sensors, and 3) meta-learning for robotic applications.

A. Food Picking

According to a survey [1], various food-picking approaches have been proposed. Many physically soft grippers have been developed for hardware approaches [7], [8]. Owing to their physical softness, such grippers fit food objects and handle them without breaking them. In contrast, we use a common robotic gripper and tactile sensors to pick foods. We can rapidly launch the picking system by attaching the tactile sensors to existing robotic systems.

For control approaches, many studies have developed mathematical models of food objects for picking by identifying Young's modulus [9] and viscoelasticity [10], [11]. Huang *et al.* [12] reproduced the deformation of food objects at grasping, such as tofu blocks, based on a simulation. The robots grasp foods based on their known physical properties without addressing the intra-category differences.

Recent learning approaches demonstrated grasping entangled [13] and cluttered [14] food objects using vision. However, it is difficult to estimate how strongly those food objects endure without being broken only from visual information. Unlike the studies, we propose picking food objects without breaking them using tactile sensors.

B. Grasping with Tactile Sensors

Many works have employed tactile sensors to enhance the grasping ability [15]. Tactile sensors are useful for grasping stability [16] and object localization [17].

Although most of the methods have focused on rigid objects, several studies have addressed grasping fragile objects using a heuristic controller [18], human demonstrations [19], [20], slip detection [21], [22], and deformation detection [23]. The proposed method is compatible with these methods. In addition, the proposed fracture-aware method would be beneficial for anticipating future fractures, leading to safer picking.

Nishimura *et al.* [24] detected the beginning of a fracture using a polynomial model and then stopped the gripper. However, this method requires real data by breaking the food objects for model fitting. In contrast, we train the fracture anticipation network with reusable objects. A DG technique facilitates dealing with unseen or unfamiliar object categories.

C. Meta-learning for Robotic Applications

Recently, meta-learning and meta-reinforcement learning techniques have been widely used for robotic applications [25] to quickly adapt to new environments, such as industrial insertion [26], [27], [28], drones [29], legged robots [30], and transfer to different robots [31]. The present study provides the first attempt to employ meta-learning for food-picking applications using MLDG [6] implemented with the model-agonistic meta-learning (MAML) algorithm [32] and long short-term memory (LSTM) [33].

III. Preliminaries

Our goal is to enable robots to pick up food objects without breaking them. In this study, we use the definitions of fractures and problem formulations presented in a previous study [5].

A. Definition of Fracture

We used a robot arm with a parallel gripper and two distributed tactile sensors, each of which has 16 taxels. Fig. 2 shows the signals of the tactile sensor when the robot closes its gripper at a constant speed and breaks food and reusable objects.

The force patterns of the objects differ, and the peak forces appear at different timings in each taxel. We also noticed visible fractures after the forces had peaked in all objects. Thus, in this study, a fracture was considered to occur around the peak force that first appeared among the 32 taxels of the two tactile sensors.

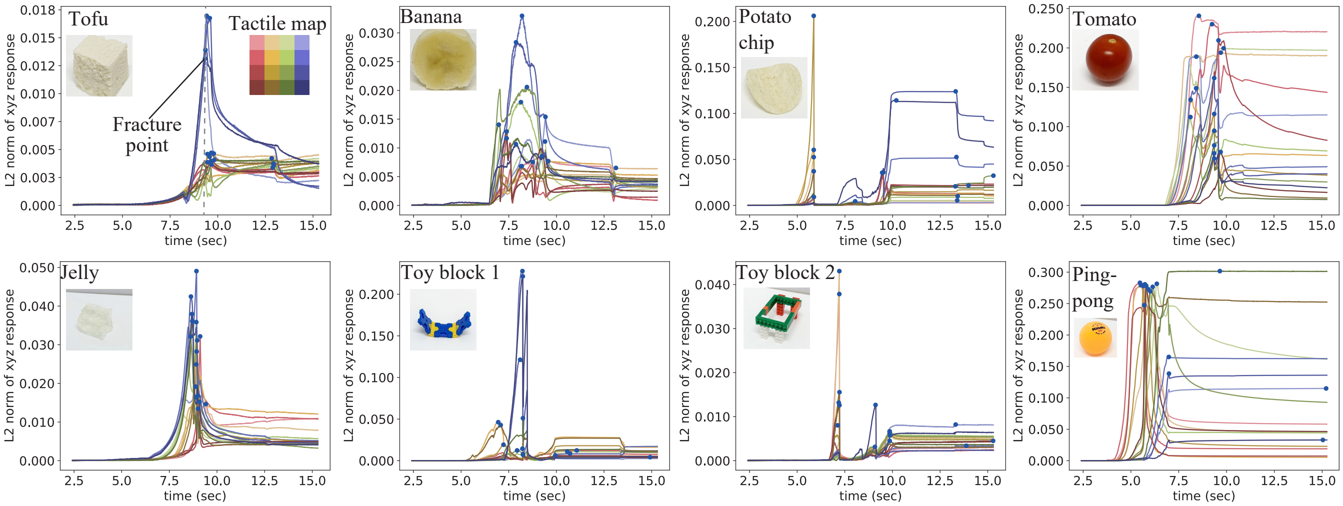

Fig. 2. Tactile signals when a robot breaks food objects (top) and reusable fragile objects (bottom). The horizontal axis indicates the time, and the vertical axis indicates the norm of the three-axis force signal. The line colors correspond to the locations of the taxels. We define a fracture as the time of the first force peak (dashed line).

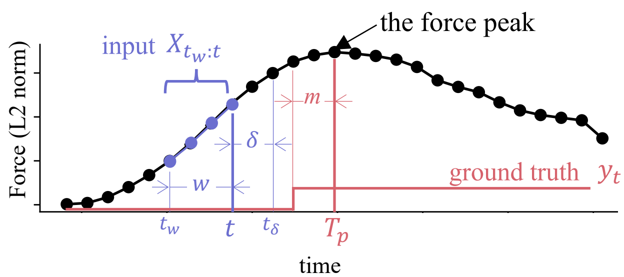

Fig. 3. Mathematical notation for fracture anticipation [5]. The input window $X_{t_w:t}$ slides along the time axis and predicts $\hat{y}_{t_\delta}$, where t_δ is the target time for anticipation. The ground truth y_t is True when $T_p - m \leq t_\delta$, where m represents the safety margin that should cover the actual fracture point before the peak.

B. Problem Formulation

Fig. 3 shows the mathematical notation for fracture anticipation for food picking. The robot must stop before it creates a fracture. To this end, the robot stops closing its gripper at the timestep $T_p - m$, where T_p is the peak time, and m is a safety margin that should cover the fracture timing (i.e., we assume the fracture occurs during $T_p - m$ to T_p). Let $t_w = t - w$ be the first timestep of the input observation and $t_\delta = t + \delta$ be the target timestep of the fracture anticipation. Then, the fracture anticipation problem is formulated as $y_{t_\delta} = f(X_{t_w:t})$, where $X_{t_w:t} = \{x_{t_w}, \ldots, x_t\}$ is a sequence of observations from the tactile sensors, and y_{t_δ} is a binary value that indicates whether the robot exceeds the fracture timing (True if $T_p - m \leq t_\delta$, False otherwise). When y_{t_δ} is True, the robot stops closing the gripper.

Training a fracture anticipation model requires a large number of breakage observations. To avoid food loss, we considered training the models using reusable fragile objects instead of real food objects. It is highly expected that the model trained with reusable fragile objects suffers in testing with food objects due to the shift in force patterns, as shown in Fig. 2, which is known as domain shift. Thus, we propose employing DG to overcome the domain shift problem.

IV. Proposed Method

In this section, we describe the proposed training framework for a fracture anticipation network. To train the network, we collected tactile signal data when the robot broke the reusable objects. We then applied meta-learning-based DG [6] to enhance the generalization ability of the trained network.

A. Fracture Anticipation Network

We aimed to predict a fracture at least δ timesteps in advance. To achieve this, we adopted a simple LSTM classifier.

Let $\mathcal{E} : X_{t_w:t} \rightarrow \{z_t, h_t^{\mathcal{E}}\}$ be an LSTM encoder, where z_t represents the output of the encoder. Then, z_t is fed to a dense layer $\mathcal{M} : z_t \rightarrow \hat{y}_{t_\delta}$, where $\hat{y}_{t_\delta}$ is an estimate of the binary fracture state y_{t_δ}. We trained the model with the loss function $L_{\text{ce}}(\hat{y}_{t_\delta}, y_{t_\delta})$, where L_{ce} represents the binary cross entropy.

B. Domain Generalization for Unseen Object Picking

Among many DG methods, we adopted a meta-learning-based method, MLDG [6], as it has a solid theoretical background and reliability, even with the fracture anticipation task. Because the original implementation in [6] uses an outdated meta-learning algorithm, we reimplemented it with MAML algorithm [32], a mature algorithm with a solid theoretical background and thus is effective for various network architectures, including LSTMs.

To avoid overfitting with meta-learning, we pre-trained the entire model once with all training samples and the standard loss function L_{ce}. Then, freezing the encoder $\mathcal{E}$, we updated only the parameters of $\mathcal{M}$ with MLDG.

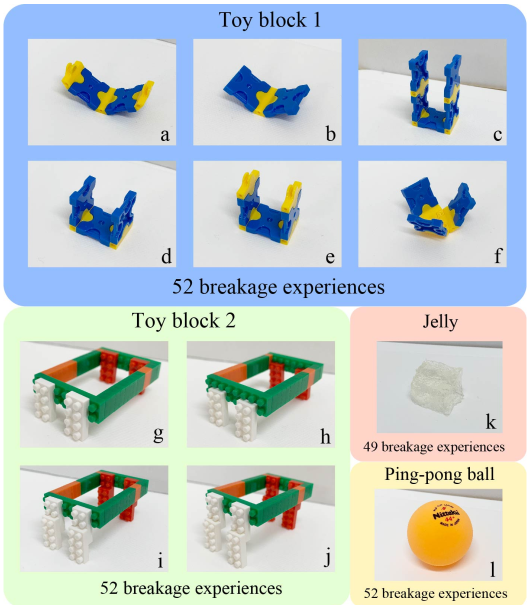

Fig. 4. Different forms of the reusable objects used to collect breakage experiences to create a dataset.

C. Reusable Fragile Objects

We used reusable fragile objects to train the fracture prediction model. The objects should be recovered effortlessly and express multiple fracture patterns similar to real food objects. Three types of fractures are assumed in this paper: *cracking*, *rupturing*, and *crushing*. To satisfy the requirements for reusing and covering the fracture types above, we carefully selected four objects: two toy blocks, ping-pong balls, and jellies.

Each toy block was composed of different kinds of small, rigid pieces. We assembled the pieces into multiple fragile shapes with rigid pieces to imitate the foods' intra-category differences (Fig. 4). A *cracking* fracture was simulated using a block of the pieces in such a way that the load of the grippers is applied to the connection of the blocks, and the connection is disconnected. We adopted two types of toy blocks because the proposed network tends to overfit with only one type.

The toy blocks can be reassembled as many times as needed. Ping-pong balls can also be reverted back to a sphere by warming and expanding the air inside, for example, by immersing them in boiling water. Jellies can be reshaped many times after they are broken by reheating them and then cooling them.

V. Experiments

To verify how effective the proposed method is in generalizing reusable objects to unknown non-reusable foods, we performed experiments with a real robot. This section describes the experimental setup, the test protocol for the experiments, and the results.

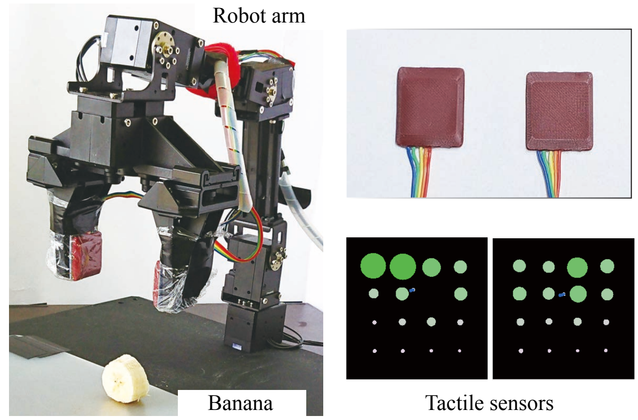

Fig. 5. The experimental setup. We used a parallel gripper and two distributed tactile sensors [5].

A. Setup

1) Robot Platform: We set up a robot arm (OpenMANIPULATOR-X, RM-X52-TNM, ROBOTIS Ltd.) [34] with four joints and a parallel gripper. We then collected the fracturing data and tested the robot's picking performance. Fig. 5 shows the robotic setup. The gripper position is controlled in Cartesian space, and we set the grippers to close at a constant speed. A Hall effect tactile sensor (uSkin XR 1944, XELA Robotics Co., Ltd.) [35] is affixed to the inward-facing side of each gripper. Each sensor is a composite of 4×4 taxels, where each taxel measures a three-axis force signal. In other words, we can observe a change of 96 forces for the left and right sides of the grippers. Tactile sensor data are sampled at approximately 40-Hz intervals.

2) Dataset Generation: We created a novel dataset that observes the change in tactile sensor values during a predetermined fracturing motion of the grippers. The target objects are placed at a fixed location under the gripper, where the grippers do not fail to fracture or lift the target object. In detail, the robot performs the following routine action on its gripper to fracture a target object:

1) The gripper is opened just above the target object's position.
2) The gripper is lowered until it barely touches the surface of the table.
3) The gripper is closed completely to fracture the target.
4) The gripper is opened and goes back to the initial position.

In addition, we generated a label that indicates whether the target object has been fractured at each timestep. For efficiency, we designed an algorithm to automatically detect the fracture point of the experiments by detecting the earliest local maximum among the 32 overall taxels. We utilized Python SciPy libraries for this peak detection. Before detecting the peak, the recorded data are first denoised in the frequency domain using the following procedure: First, each taxel datum is interpolated at a regular interval of 0.008

with linear interpolation. Second, we calculate the L2 norm from the 3D values of each tactile datum. Third, we convert the time domain data into the frequency domain using FFT and cut off the signals if their value is less than 0.0001. Finally, the denoised data in the time domain are recovered using IFFT. The interpolation, FFT, and IFFT procedures are also implemented with the Python SciPy package. Then, we define the first peak of the overall 32 tactile sensors as a fracture point T_p of the attempt. For each breakage experience, tactile sensor values for 1.5 s from 1 s before to 0.5 s after the fracture point T_p were used as the training. We set the window size w of input window $X_{t_w:t}$ to 10 in training and 11 at evaluation.

Based on the procedure above, we collected 52, 52, and 49 breakage experiences for two types of toy blocks, ping-pong balls, and jellies, respectively (i.e., this resulted in a total of 205 temporal sequences). The breakage experiences of the toy blocks were composed of six different forms of toy block 1 and four different forms of toy block 2 (Fig. 4). We changed the forms of the toy blocks, expecting to train the intra-category difference. This dataset is always used to train the fracture anticipation model in this paper. Thus, no food objects are involved in the training process.

3) Network Architecture: Following [5], we use a two-layered LSTM followed by one fully connected layer as encoder $\mathcal{E}$. Each layer of the LSTM has 32 hidden units. The $\{4 \times 4 \times 2 \times 3\}$-dimensional gradients of the tactile signals at each timestep are flattened into a 96-dimensional vector and fed to $\mathcal{E}$ as input $X_{t_w:t}$. Then, its 32-dimensional outputs are fed to $\mathcal{M}$, a fully connected layer. Finally, $\mathcal{M}$'s output is further processed via a sigmoid function, and we obtain an estimate $\hat{y}_\delta$. We use PyTorch 1.9.0 and CUDA 11.1 for the implementation.

4) Pre-training and Meta-learning: To train the model efficiently and effectively, we first pre-train the entire model ($\mathcal{E}$ and $\mathcal{M}$) with all training samples. Then, freezing the parameters of $\mathcal{E}$, we update only the parameters of $\mathcal{M}$ using the MAML algorithm. We carefully note that the pre-trained model (i.e., the model before meta-learning) is adopted as the baseline in the evaluation phase.

In the pre-training, we used the stochastic gradient descent (SGD) optimizer with its default parameters in PyTorch. We set the learning rate to 0.01 and its iteration number to 80.

We apply the meta-learning for 325 iterations. In each iteration, we repeat the meta-training and meta-validation steps four times. The (meta) train-validation split at each step was determined in a leave-one-object-out manner. Referring to the MAML setting [32], for each step, we took the 3×5 (three-way, five-shot classification) number of data as the batch size and split the (meta) train-validation as 2×5 and 1×5. We adopted learning rates of 0.0005 at meta-training and 0.005 at meta-validation. The SGD optimizer with the same parameters as the pre-training was used.

B. Evaluation Protocol

To validate the proposed learning strategy, we demonstrated food picking with a real robot, where we evaluated the success rate of the food picking with actual foods. For the picking target, we selected *tofu*, *potato chips*, *tomatoes*, and *bananas* to maintain inter-category diversity in their physical properties. We performed grasping trials 20 times for each of the four food objects. To evaluate the impact of DG, we compared the proposed method with the baseline, which is the pre-trained model before meta-learning.

TABLE I

GRASPING PERFORMANCE (SUCCESS/ATTEMPT) FOR FOOD OBJECTS

Test Food	Baseline	Ours
Tofu (cuboid)	1/20	**11**/20
Tofu (cube)	13/20	**19**/20
Bananas	18/20	**20**/20
Tomatoes	**20**/20	16/20
Potato chips	17/20	**19**/20

In this experiment, we aimed to place the objects at the same initial location and pose as shown in the bottom row of Fig. 1, but this was done by hand; thus, there was a natural variance. We describe their shapes and poses in more detail. For tofu, we prepared two shape variations, *cuboid* and *cube*; they share the same height (20–30 mm) and depth (15–20 mm) but have different widths (40 mm and 20 mm, respectively). Note that the *width* $\times$ *depth* surface always faces the ground, while the *depth* $\times$ *height* surface contacts with the gripper. For potato chips, we placed the objects in a concave upward pose. We fixed the bananas' shapes to be round slices, with thicknesses ranging from approximately 7 to 15 mm. They were placed so that the round surface faced the front (thus, the gripper made contact with the side). Finally, we placed the tomatoes so that their stem sides faced behind (although we removed the stems in advance). They were sometimes tilted slightly as they were round.

C. Results

Fig. 6 shows photographs of successful picking to see how the robot picked the food objects. The robot stopped closing the gripper and picked up the objects without fracturing them.

Table I shows the picking performances of five food objects (tofu cuboids, tofu cubes, bananas, potato chips, and tomatoes) for the baseline and the proposed method. We counted an attempt as a *success* only if the object could be picked up without being crushed, cracked, or ruptured; otherwise, we counted the attempt as a *failure*. Note that throughout the experiments, most failure patterns are by fracture, and the robot did not stop its gripper too early.

The baseline model performed well with bananas, tomatoes, and potato chips. This shows that the diversity of reusable objects already covers those food objects well. DG further improved the success rates for bananas and potato chips. The significant performance gains for the tofu cuboids and tofu cubes demonstrated the extended generalization ability of the model with DG. On the other hand, it degraded the performance of tomatoes. Finding the cause of this negative effect remains for future work.

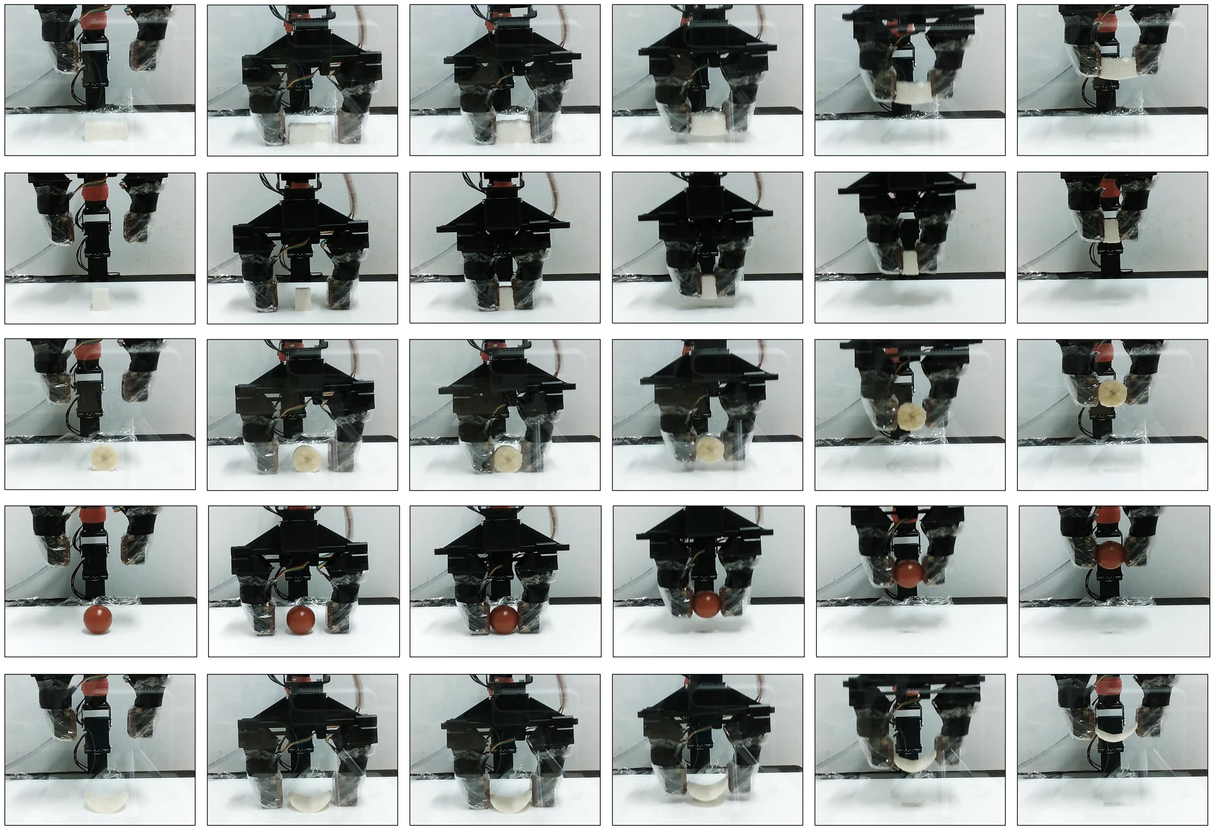

Fig. 6. Photographs of successful picking of five food objects.

VI. Discussion

In this section, we attempt to explain the experimental results with the signal similarity, based on Fig. 2, and the limitations of this research. In Fig. 2, the form of the observed signals of the potato chip is perceptually similar to that of the toy blocks. Although the model may consider more sophisticated signal dynamics, the similarity would result in the high performance of the baseline method for potato chips. Similarly, the signal of the jelly is similar to that of the banana's taxel, which first reached the peak.

On the other hand, the tofu signals have smaller intensity than any reusable objects, leading to failures with the baseline method. However, DG could obtain an intensity-invariant anticipation strategy due to the diversity of intensity among the training data, which improved the performance.

Unfortunately, DG had an adverse effect on tomatoes. When we focus on the signal before the peaks, that of the tomato looks similar to that of ping-pong balls in their forms and intensities. This significant similarity should contribute to the perfect performance of the baseline method, but DG muted the effect of the substantial similarity and degraded the performance.

As a limitation, we did not investigate which reusable objects were beneficial for food picking. Moreover, how to select the right kind of reusable object for a given category of food to be picked is debatable.

VII. Conclusion

We presented a strategy for picking fragile food objects without breaking them using a universal robot gripper and tactile sensors. As grasping objects is a fundamental task in food manipulation, the proposed method should be of use for robots in various food-related applications. This approach collects tactile signals by breaking the objects and subsequently trains a fracture prediction network. To verify the proposed method, we performed experiments using a real robot trained to pick up food objects. The results show that we can train a food-picking robot without food by combining the dataset generation by breaking reusable objects and a domain generalization technique.

References

[1] Z. Wang, S. Hirai, and S. Kawamura, "Challenges and opportunities in robotic food handling: A review," *Frontiers in Robotics and AI*, p. 433, 2022.

[2] J. J. Roldán, J. del Cerro, D. Garzón-Ramos, P. Garcia-Aunon, M. Garzón, J. De León, and A. Barrientos, "Robots in agriculture: State of art and practical experiences," *Service Robots*, pp. 67–90, 2018.

[3] J. Liu, Y. Chen, Z. Dong, S. Wang, S. Calinon, M. Li, and F. Chen, "Robot cooking with stir-fry: Bimanual non-prehensile manipulation of semi-fluid objects," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5159–5166, 2022.

[4] I. Lenz, R. A. Knepper, and A. Saxena, "DeepMPC: Learning deep latent features for model predictive control." in *Robotics: Science and Systems*, 2015.

[5] R. Ishikawa, M. Hamaya, F. von Drigalski, K. Tanaka, and A. Hashimoto, “Learning by breaking: Food fracture anticipation for robotic food manipulation,” *IEEE Access*, vol. 10, pp. 99 321–99 329, 2022.

[6] D. Li, Y. Yang, Y.-Z. Song, and T. Hospedales, “Learning to generalize: Meta-learning for domain generalization,” in *AAAI Conference on Artificial Intelligence*, 2018, pp. 3490–3497.

[7] Z. Wang, Y. Torigoe, and S. Hirai, “A prestressed soft gripper: Design, modeling, fabrication, and tests for food handling,” *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 1909–1916, 2017.

[8] J. Shintake, V. Cacucciolo, D. Floreano, and H. Shea, “Soft robotic grippers,” *Advanced Materials*, vol. 30, no. 29, p. 1707035, 2018.

[9] A. Sinha and A. Bhargav, “Young’s modulus estimation in food samples: Effect of experimental parameters,” *Mechanics & Industry*, vol. 21, no. 4, p. 404, 2020.

[10] N. Sakamoto, M. Higashimori, T. Tsuji, and M. Kaneko, “An optimum design of robotic hand for handling a visco-elastic object based on maxwell model,” in *IEEE International Conference on Robotics and Automation*, 2007, pp. 1219–1225.

[11] Z. Wang, S. Inoue, Y. Hashimoto, and S. Kawamura, “Measuring viscoelasticity and friction of tempuras for robotic handling,” *Journal of Food Engineering*, vol. 310, p. 110707, 2021.

[12] I. Huang, Y. Narang, C. Eppner, B. Sundaralingam, M. Macklin, R. Bajcsy, T. Hermans, and D. Fox, “Defgraspsim: Physics-based simulation of grasp outcomes for 3d deformable objects,” *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6274–6281, 2022.

[13] K. Takahashi, N. Fukaya, and A. Ummadisingu, “Target-mass grasping of entangled food using pre-grasping & post-grasping,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1222–1229, 2021.

[14] A. Ummadisingu, K. Takahashi, and N. Fukaya, “Cluttered food grasping with adaptive fingers and synthetic-data trained object detection,” *arXiv preprint arXiv:2203.05187*, 2022.

[15] A. Yamaguchi and C. G. Atkeson, “Recent progress in tactile sensing and sensors for robotic manipulation: Can we turn tactile sensing into vision?” *Advanced Robotics*, vol. 33, no. 14, pp. 661–673, 2019.

[16] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine, “More than a feeling: Learning to grasp and regrasp using vision and touch,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3300–3307, 2018.

[17] M. Bauza, O. Canal, and A. Rodriguez, “Tactile mapping and localization from high-resolution tactile imprints,” in *IEEE International Conference on Robotics and Automation*, 2019, pp. 3811–3817.

[18] J. M. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K. J. Kuchenbecker, “Human-inspired robotic grasp control with tactile sensing,” *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1067–1079, 2011.

[19] E. Misimi, A. Olofsson, A. Eilertsen, E. R. Øye, and J. R. Mathiassen, “Robotic handling of compliant food objects by robust learning from demonstration,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018, pp. 6972–6979.

[20] A. Lillienskiold, R. Rahal, P. R. Giordano, C. Pacchierotti, and E. Misimi, “Human-inspired haptic-enabled learning from prehensile move demonstrations,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021.

[21] A. Yamaguchi and C. G. Atkeson, “Tactile behaviors with the vision-based tactile sensor fingervision,” *International Journal of Humanoid Robotics*, vol. 16, no. 3, p. 1940002, 2019.

[22] T. Narita, S. Nagakari, W. Conus, T. Tsuboi, and K. Nagasaka, “Theoretical derivation and realization of adaptive grasping based on rotational incipient slip detection,” in *IEEE International Conference on Robotics and Automation*, 2020, pp. 531–537.

[23] S. Funabashi, T. Isobe, S. Ogasa, T. Ogata, A. Schmitz, T. P. Tomo, and S. Sugano, “Stable in-grasp manipulation with a low-cost robot hand by using 3-axis tactile sensors with a CNN,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 9166–9173.

[24] T. Nishimura, Y. Suzuki, T. Tsuji, and T. Watanabe, “Fluid pressure monitoring-based strategy for delicate grasping of fragile objects by a robotic hand with fluid fingertips,” *Sensors*, vol. 19, no. 4, p. 782, 2019.

[25] M. Huisman, J. N. Van Rijn, and A. Plaat, “A survey of deep meta-learning,” *Artificial Intelligence Review*, vol. 54, no. 6, pp. 4483–4541, 2021.

[26] T. Z. Zhao, J. Luo, O. Sushkov, R. Pevceviciute, N. Heess, J. Scholz, S. Schaal, and S. Levine, “Offline meta-reinforcement learning for industrial insertion,” in *IEEE International Conference on Robotics and Automation*, 2022, pp. 6386–6393.

[27] Z. Zhao, A. Nagabandi, K. Rakelly, C. Finn, and S. Levine, “Meld: Meta-reinforcement learning from images via latent state models,” in *Conference on Robot Learning*, 2021, pp. 1246–1261.

[28] G. Schoettler, A. Nair, J. A. Ojea, S. Levine, and E. Solowjow, “Meta-reinforcement learning for robotic industrial insertion tasks,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 9728–9735.

[29] S. Belkhale, R. Li, G. Kahn, R. McAllister, R. Calandra, and S. Levine, “Model-based meta-reinforcement learning for flight with suspended payloads,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1471–1478, 2021.

[30] R. Kaushik, T. Anne, and J.-B. Mouret, “Fast online adaptation in robotics through meta-learning embeddings of simulated priors,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 5269–5276.

[31] A. Ghadirzadeh, X. Chen, P. Poklukar, C. Finn, M. Björkman, and D. Kragic, “Bayesian meta-learning for few-shot policy adaptation across robotic platforms,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2021, pp. 1274–1280.

[32] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in *International Conference on Machine Learning*, 2017, pp. 1126–1135.

[33] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[34] Robotis, “OpenMANIPULATOR-X e-manual,” https://emanual.robotis.com/docs/en/platform/openmanipulator_x/overview/.

[35] XELA Robotics, “Model XR1944,” https://xelarobotics.com/xr1944.