

Dear Editor,

Re: IEEE RA-L 24-3608, One-Shot Demonstration for Slicing and Cutting Everyday Food Items

Thank you for giving us the opportunity to submit a revised draft of the manuscript "One-Shot Demonstration for Slicing and Cutting Everyday Food Items" for publication in the IEEE Robotics and Automation Letters. We appreciate the time and effort that you and the reviewers dedicated to providing feedback on our manuscript and are grateful for the insightful comments. With the help of your comments, we were able to improve our work. We addressed the issues raised by the reviewers. Please find attached a point-by-point explanation of all the changes made to the paper. All changes are highlighted in red in the paper.

Yours Sincerely,
Yi Liu

List of Changes

Location of Changes	Reason for Changes
Sec.I 2nd paragraph,	Comments addressed
Sec.II 4th paragraph,	Comments addressed
Sec.III-A,	Comments addressed
Sec.III-B.1,	Comments addressed
Sec.III-B.2,	Comments addressed
Sec.III-B.3,	Comments addressed
Sec.III-B.4,	Comments addressed
Sec.IV-C 5th paragraph,	Comments addressed
Sec.IV-D,	Comments addressed
Table I	New table
Sec.V,	Comments addressed

Reviewer 1

I appreciate the authors' detailed responses and the substantial revisions made to improve clarity, generalization analysis, and experimental validation. The added discussions on failure cases, trajectory update frequency, and planning time further strengthen the manuscript.

Comment 1

I still have some doubts regarding certain technical details about the force Compensation in Section III.B(3).

The authors correctly point out that the recorded force trajectory $f_d(t)$ from one object cannot be directly applied to another due to different physical properties. However, I am not entirely convinced about the reasoning behind Eq. (12) for the feedforward force \hat{f}_v , which is a weighted combination of $f_v(t)$, $f_v(t - 2)$, and $f'_y(t)$.

Since the weight $2T$ for $f'_y(t)$ is small, Eq. 12 is largely dominated by $f_v(t)$, which I understand as a low-pass-filtered position tracking error with exponential scaling. This raises the question of whether \hat{f}_v is essentially just a position correction term rather than a true force feedforward mechanism.

Thus, it is unclear how the learned force trajectory $f_y(t)$ shape contributes to tracking accuracy. To strengthen this claim, I suggest including a figure illustrating $f_y(t)$, $f_v(t)$, and \hat{f}_v over time t , along with further justification of how $f'_y(t)$ influences the reference force trajectory. (Fig. 6 does not include the plot of $f_y(t)$; the plot rendering quality should be improved).

Response 1

Thank you for this comment. We agree to add the mentioned figure, which can show the experimental results more clearly.

Firstly, we made a supplementary explanation of Equ. 9. In Equ. 9, the first term $(f_v(t_{n-2}) + 2Tf'_y(t_n))$ is obtained by taking the derivative of $f'_v(t_n) = (f_v(t_n) - f_v(t_{n-2}))/2T$. We replaced $f'_v(t)$ with $f'_y(t)$ to correct the direction of $f_v(t)$. Hence, we got the first term $f_v(t_n) = f_v(t_{n-2}) + 2Tf'_y(t_n)$. The second term of the equation represents the model's confidence in $f_v(t)$. We agree that $2T$ leads

to a smaller impact of $f'_y(t)$, but we believe this is a reasonable mathematical derivation result.

Furthermore, \hat{f}_v consists of two parts: f_v and f'_y . The former is generated based on the position error and can be regarded as a position correction term for the admittance controller, and the latter is a force feedforward term guided by the demonstration force shape. To avoid ambiguity, we explain it in Sec-III.B(3) and discuss and analyze the influence of the two terms in Sec-IV.C and Sec-IV.D.

Finally, we improved and added the mentioned figure and analyses in the experimental section. The legend of Figure 6b might be misleading to readers, where y_1 represented the demonstrated force trajectory, i.e., $f_y(t)$. We added the Fig. 6(c) for comparing the importance of $f_y(t)$, $f_v(t)$, and \hat{f}_v . We added the relevant discussion in IV-D(1). To avoid misunderstanding, we revised Force Compensation to Compensation in III.B(3) and added more details for explanation.

Comment 2

In section III.A, the stability analysis of the teleoperation manipulator control could be removed, as it does not seem directly relevant to the contributions of this paper and may distract readers.

Response 2

Thank you for this comment. We agree that the stability analysis of teleoperation is not the contribution of this paper. We used citations from other papers and removed the relevant stability analysis.

Comment 3

The notation p in R^6 is not accurate to fully represent postures in SE(3). Consider revising this to Euler angles with Cartesian position

Response 3

We appreciate this comment. We revised the contents to Euler angles with Cartesian position.

Comment 4

I don't think the actual controller is described anywhere. I believe the admittance controller in Section III.B(1) is designed for teleoperation in Section III.A, as it tracks the desired force trajectory $f_d(t)$. However, it is unclear what control law is used for learned manipulation. Clarifying this aspect would improve the completeness of the methodology.

Response 4

Thank you for this suggestion. The admittance controller we designed in III-B(1) can be used in both the teleoperation system and the robot independent control system (autonomous cutting). For the latter, $f_d(t)$ in the admittance controller is the force compensation value we need as a virtual force, and its control law is consistent with the position controller (PD controller). We have added relevant explanations in the context.

Comment 5

Section III.B(2) refers to the controller as impedance control, but since it does not explicitly regulate motor torques, it should be classified as an admittance controller.

Response 5

Thank you for this comment. We have corrected the misuse of this term in Sec-III.B(2).

Comment 6

In my opinion, renaming "Constraint 1" and "Constraint 2" to "Condition 1" and "Condition 2" would make the descriptions more natural and intuitive.

Response 6

Thank you for this suggestion. We changed the 'Constraint' to 'Condition' in the context.

Comment 7

The symbol t is used to represent both a discrete step index and continuous time, which may cause confusion. If t is intended as a discrete index, it should not appear in the continuous derivative of a function. Consider using a different symbol to clearly distinguish between the two.

Response 7

Thank you for this suggestion. We distinguish the use of t . The symbol t now represents continuous time. The symbol n now represents discrete step time.

Reviewer 2

This paper presents a robotic cutting framework that combines dynamic motion primitives (DMP), force compensation, and trajectory replanning to improve the execution of cutting skills for a variety of food products. This research effectively addresses the challenge of adapting robotic cutting strategies to the different physical properties of ingredients. Experimental results show that the proposed methods (cr-DMPs) have advantages over baseline methods (p-DMPs), especially in force trajectory tuning and handling of softer foods.

Comment 1

Clarity in the Problem Statement

- Chapter 1, Paragraph 2, Last Sentence

The earlier part of the paragraph describes the types of knife skills but does not establish a direct causal relationship with the goal stated in the last sentence. Providing a clearer connection would improve the logical flow.

- Chapter 2, Paragraph 4, Last Sentence

The preceding narrative mainly summarizes the correlation study itself but does not clearly illustrate the methodological limitations in adapting it to various objects. A more explicit discussion of these constraints would be beneficial.

Response 1

We appreciate this suggestion.

In Chapter 1, we improved the introduction to explain our purpose more clearly.

In Chapter 2, we added more discussion on the limitations of the mentioned methods.

Comment 2

Necessity and Clarity of the Obstacle Avoidance Term C_t

- Equation (7) and Related Discussion

The paper introduces the obstacle avoidance term C_t in Eq. (7). However, its necessity is not validated in subsequent discussions

or experiments. Would it be more appropriate to either provide a more detailed explanation or simplify the description in the section between Eq. (7) and Eq. (8)?

- Clarity of the Rotation Matrix R

The description of the rotation matrix R associated with C_t is somewhat ambiguous. The paper states that R is "the transformation between $(\mathbf{o} - \mathbf{y})$ and $\dot{\mathbf{y}}$." However, external references suggest that R is typically computed using:

$$\mathbf{r} = (\mathbf{o} - \mathbf{y}) \times \dot{\mathbf{y}}$$

which ensures that C_t outputs a force perpendicular to the velocity direction, thereby altering the velocity direction for obstacle avoidance. The current explanation does not explicitly establish this relationship, making the role of R unclear. It would be beneficial to clarify whether R is designed to guarantee that C_t acts perpendicular to the velocity.

Response 2

Thank you for this comment.

For the Equation (7) and Related Discussion, we agree that C_t is not necessary in this paper because we did not set obstacles for the cutting trajectory. C_t is used to represent the status of obstacles in the original DMP. We clarified this in the context and removed the detailed explanation of C_t .

For the rotation matrix R , since we set $C_t = 0$ in the context, we would not explain C_t in the text. We would explain R by citing external literature here: $*C_t = \gamma R \dot{\mathbf{y}} \theta \exp(-\beta \theta)$, the angle θ is interpreted as the angle between the velocity vector $\dot{\mathbf{y}}$ and the difference vector $(\mathbf{o} - \mathbf{y})$ between the current position and the obstacle. The vector \mathbf{r} is the vector that is perpendicular to the plane spanned by $\dot{\mathbf{y}}$ and $(\mathbf{o} - \mathbf{y})$, and serves to define a rotation matrix R , which causes a rotation of 90 degrees about \mathbf{r} (Sciavicco, Siciliano, 2000). Intuitively, the coupling term adds a movement perpendicular to the current movement direction as a function of the distance vector to the obstacle (see Hoffmann et al., 2009, for more details).*

Comment 3

Justification for Dynamic Variables g and y_0

- Chapter 3, Section 4, First Paragraph

Although later sections discuss the need to adapt to changes in g and y_0 , introducing them as dynamic variables at this point feels somewhat abrupt. Providing a brief explanation here would help transition more smoothly into the discussion on their adaptability.

Response 3

Thank you for this suggestion. We added more explanation in the first paragraph of Sec-III.B(4) to make reading smoother.

Comment 4

Experimental Validation and Applicability to Complex Trajectories

- Chapter 4, "Experiments"

While the proposed approach has the potential to handle complex trajectories, the actual experiments only involve relatively simple cutting motions (even though three trajectories are tested, all of them remain within the plane of the knife).

Would it be possible to validate the approach on more complex cutting tasks, such as:

- Removing potato sprouts
- Trimming tomato tips
- Using the knife tip to carve curved patterns into food surfaces?

Response 4

Thank you for this comment.

We agree that interesting work such as trimming potato sprouts or carving curved patterns can enhance the highlights of this paper. However, in this paper, we are currently focusing on how to enable the robot to quickly learn multiple knife skills to cut unknown objects. The knife skills we designed include common cutting skills in daily life, which have been introduced in the introduction part, and we did not make more complex tasks. For example, trimming potato sprouts requires more sophisticated visual detection; when

performing complex engraving on an object, a fast one-shot imitation may result in poor engraving results or even failure. Using intelligent methods such as reinforcement learning/diffusion strategies with demonstration trajectories can achieve the results (our current work).

We add more complex tasks in future work but not in this paper. We have added this point in the future work section of this paper. In order to avoid ambiguity, the description of complex knife skills in the paper has been revised to common knife skills in daily life.

Comment 5

Experimental Data and Food Hardness Considerations

- Providing Hardness Parameters for Food Items

Instead of relying on subjective assessments of food hardness, providing quantitative hardness parameters would enhance the credibility of the experimental results.

For instance, apples and pears are generally considered to have similar hardness, yet the performance in id3 and id4 in p-DMPs shows noticeable differences. Clarifying whether this difference is due to actual hardness variation or other factors would improve result interpretation.

Response 5

Thank you for this comment. We agree to describe the hardness of food in a quantitative way. In this paper, a fixed pressing area and pressing distance are used to obtain the pressing force to represent the hardness of food. The methodology on how we quantify the hardness of the food are explained in the last paragraph of IV-C. The quantification of the hardness can be found in Table 1 of the revised paper.

As for pears and apples, the possible factor is that the pears we buy locally are softer than those in other areas/riper than the apples.

Comment 6

Chapter 3, First Paragraph, Middle Section

- *[where $p_s, p_m \in R^6$ represent the posture of the primary and secondary]*

The notation appears to be reversed. Should it be *[where $p_m, p_s \in R^6$ represent the posture of the primary and secondary]* instead?

- *[where $f_m, f_s \in R^6$ represent controlled force]*

It is recommended to provide a more detailed explanation of Controlled Force.

For readers unfamiliar with Bilateral Teleoperation Systems, this term may be misunderstood (e.g., it could be interpreted as a desired force rather than a force outputted by the robot).

Response 6

Thank you for this suggestion.

We revised the reversed notation to *[where $p_m, p_s \in R^6$ represent the posture of the primary and secondary]*.

We added more details to explain f_m and f_s .

One-Shot Demonstration for Slicing and Cutting Everyday Food Items

Yi Liu¹, Andreas Verleysen¹, Francis wyffels¹

Abstract— Cutting everyday food items presents a significant challenge in robotics due to the multiple types of knife skills and the unpredictable mechanical behaviour of materials during manipulation. To address this, we propose a one-shot demonstration-based framework that integrates the imitation of both position and force trajectories of knife skills using dynamic movement primitives (DMPs). Our approach incorporates a force compensation method to replicate the force trajectory and introduces relevant constraints on knife skills, allowing for trajectory re-planning with DMPs during the cutting process. Finally, we designed three knife skill demos for the robot and tested them on 14 unknown food items. The experiments are conducted to evaluate the effectiveness of the proposed force compensation and re-planning methods. The results demonstrate that our framework can successfully imitate various knife skills and cut previously unknown food items with high precision.

Index Terms—Learning from demonstration, manipulation planning

I. INTRODUCTION

CHEFS employ a combination of visual and tactile feedback to analyze cuts, enabling them to differentiate in complex materials, such as meat and fruit. Then, they make rapid, informed decisions to cut. Replicating chef-like knife skills in robots, particularly when manipulating everyday objects like animal tissue, remains a significant challenge.

Knife skills include different cutting methods. Chopping and dicing are used to completely cut through objects [1], and slicing can modify the shape of objects (e.g., fish, eggplant) without fully cutting them [2]. Carving is used to alter the surface of objects [3] like bread or chicken legs. **Different ingredients require different amounts of force [4].** It would be interesting to have a single model be able to use many knife skills similar to the above at the same time. We aim to create a robotic kitchen assistant that can handle various knife skills for various ingredients.

The robot can manipulate the knife in different ways. For specialized tasks like surgery, teleoperation allows a robot to control a scalpel, easing the surgeon's workload [5] for multiple organs. In everyday settings, robots often use automatic operations. Advanced methods simulate cutting in a virtual environment and then apply these strategies in real life [6]. However, this can struggle with simulating the objects' materials effectively. Training machine learning models with real-world data can improve performance [7] while data collection attempts exist [8], [9] gathering enough data remains difficult. Some approaches use demonstrations to avoid the need for large datasets [3], but these methods can be limited and may not work well with different types of objects. Therefore, an adaptable method that requires less data is considered.

¹Authors are with IDLab-AIRO, Ghent University – imec, Belgium (Corresponding author: yiyiliu.liu@ugent.be)

Videos are supplied in <https://0707yiliu.github.io/SoftBodySlicingVideo/>.

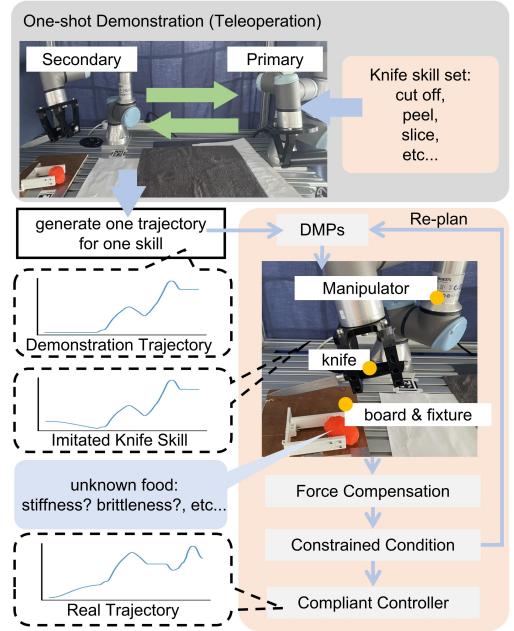


Fig. 1. The overall framework: a teleoperation system is utilized to generate the trajectories in the knife skill set, and the generated demonstration trajectories enable the robot to complete the cutting task through DMPs; the constrained condition is used to activate DMPs. Its environment includes a cutting board for fixing unknown objects and a manipulator for fixing the knife.

We use a one-shot demonstration method, as shown in Fig. 1, to deploy multiple knife skills on the robot, thus overcoming the difficulty of requiring a large number of demonstrations. Then, the DMPs method based on force compensation is proposed to imitate the demonstrated trajectory. Finally, the imitated trajectory is utilized to cut the unknown food items under constraints, and the constraints can activate DMPs to replan the trajectory. The contributions of this paper are summarized as follows:

- 1) Generate various knife skills with position and force trajectories using a one-shot demonstration method based on teleoperation;
- 2) A force compensation method is proposed to generalize the DMPs model so that it can imitate the knife skills on unknown food items based on the demonstration trajectory;
- 3) Constraints designed based on the characteristics of the knife skills are used to activate DMPs during the cutting process, allowing the robot to re-plan the trajectory in real-time;
- 4) The proposed method is deployed on a real knife-wielding robot, enabling it to cut unknown food items

utilizing the learned knife skills.

II. RELATED WORK

Knife skills that robots learn cannot be limited to a single technique, as real-world tasks require a variety of skills. A basic knife skill is the ability to cut objects directly. Some researchers have proposed compliant manipulation approaches for vegetable cutting, using reinforcement learning to guide the process [10], while others use feedback from force/torque (F/T) sensors to control the cutting action [1], [4]. However, the knife skill in these tasks is limited to cutting, making these models unsuitable for tasks requiring multiple knife skills.

In some cases, we need to slice food rather than simply cut it off. Several studies focus on cutting different types of meat [2], illustrating slicing techniques without severing the object. When dealing with foods that have cores, researchers use F/T sensors to guide the knife and avoid the core while cutting [7]. In the medical field, knife skills are applied to tasks such as cutting the epidermis or organs [11]. The dynamic motion primitives (DMPs) [12] is used to reproduce demonstrated trajectories, enabling a robot to hold a scalpel and make incisions on tissue surfaces [3], and employ teleoperation to control robots with scalpels for internal organ surgeries [13]. The DMPs is also used to handle adapting the slicing skill [14]. These works demonstrate diverse knife skills, including different cutting paths. Deploying multiple cutting skills in robots is a key challenge we need to address.

For robotic knife manipulation, different cutting paths and scenarios require different control methods [15]. Teleoperation is a crucial technique for robot-assisted cutting [3]. To reduce reliance on human control, some researchers use shared control methods [16], allowing the operator to feel the robot's interaction with the environment while granting the robot a degree of autonomy. In common cutting scenarios, machine learning is used to control robots for object cutting. Long short-term memory models are employed to approximate the complex dynamics of food cutting, enabling the use of model predictive controllers for robot control [17]. Simulators, which mimic the state of food during cutting, are used to train reinforcement learning models for real-world cutting strategies [6]. However, these methods have difficulty in transferring to the real world, or these methods usually require extensive training data [8], [9], which is difficult to obtain from the real world.

To simplify the deployment of cutting strategies, some researchers have embedded sensors inside knives to provide feedback on contact with the object being cut, guiding the cutting path [18]. Other works use vision [19] or laser [20] to guide the cutting process, particularly for surface cuts, while demonstration-based methods generate cutting paths [3]. These methods allow robots to learn through repeated demonstrations [21]. Bilateral teleoperation methods are used to enhance the human-like quality of generated paths and minimize external forces from humans during demonstrations [22], [23]. While some of these methods are easy to deploy on robots, they have only been tried on a few types of objects and are not applicable to a wide range of objects with different physical properties. The challenge remains in developing adaptable cutting paths that can handle various objects.

III. METHODOLOGY

In this section, we present our proposed framework, which consists of a demonstration system and a trajectory imitation approach based on position error.

A. One-Shot Demonstration

Cutting food items requires defining cutting paths. We use a demonstration method to capture human-like cutting paths. To ensure that external forces from the operator do not affect the robot's interaction with the environment and obtain the force trajectory for our proposed imitation method, as shown in Fig. 1, we use a bilateral teleoperation approach to generate these paths. Because we focus on local demonstration trajectories, we do not consider the time delay of the teleoperation system. The basic primary-secondary model is as follows,

$$\mathbf{M}_m(\mathbf{p}_m)\ddot{\mathbf{p}}_m + \mathbf{C}_m(\mathbf{p}_m, \dot{\mathbf{p}}_m)\dot{\mathbf{p}}_m = \mathbf{f}_m - \mathbf{f}_h, \quad (1)$$

$$\mathbf{M}_s(\mathbf{p}_s)\ddot{\mathbf{p}}_s + \mathbf{C}_s(\mathbf{p}_s, \dot{\mathbf{p}}_s)\dot{\mathbf{p}}_s = \mathbf{f}_s - \mathbf{f}_e, \quad (2)$$

where $\mathbf{p}_m, \mathbf{p}_s \in \mathbb{R}^6$ represent the posture of the primary and secondary end-effectors in Euler angles with Cartesian position, $\mathbf{f}_m, \mathbf{f}_s \in \mathbb{R}^6$ represent controlled force outputted by the robot, $\mathbf{f}_h, \mathbf{f}_e \in \mathbb{R}^6$ are the external interaction forces from human and environment. $\mathbf{M}_m(\mathbf{p}_m), \mathbf{M}_s(\mathbf{p}_s), \mathbf{C}_m(\mathbf{p}_m, \dot{\mathbf{p}}_m), \mathbf{C}_s(\mathbf{p}_s, \dot{\mathbf{p}}_s)$ are the inertial matrix and the Coriolis matrices, respectively. The matrix dimensions are 6 by 6. The matrix \mathbf{Z} of this model is utilized to represent the system's stability by the Lyapunov method [24] with Equ 1 and Equ 2 as follows:

$$\begin{pmatrix} \mathbf{f}_m \\ -\mathbf{p}_s \end{pmatrix} = \mathbf{Z} \begin{pmatrix} \mathbf{p}_m \\ \mathbf{f}_s \end{pmatrix}, \quad \mathbf{Z} = \begin{pmatrix} \mathbf{Z}_m & k_c \\ j_c & \frac{1}{\mathbf{Z}_s} \end{pmatrix}, \quad (3)$$

where $\mathbf{Z}_m, \mathbf{Z}_s$ represent the physical property of the system, j_c and k_c , related to \mathbf{f}_h and \mathbf{f}_e , represent the scale of position and force respectively. Based on the above system, we can control the primary robot to let the secondary one cut food items to generate one parameterized demonstration trajectory $Y(p, f, t)$ for one knife skill, which includes the position $p(t)$ and the force $f(t)$ at time t .

B. Knife Skills Learning

Through III-A, we generate the demonstration trajectory that contains position $p_d(t)$ and force $f_d(t)$. However, the cut food item is unknown, including its position, physical properties, volume, etc. Therefore, we designed a compliance method for the robot based on DMPs so that the trajectory demonstrated in the one shot can be generalized to multiple food items.

1) *Admittance Control*: Since the robot with the obtained trajectories $p_d(t)$ and $f_d(t)$ needs to be tracked to enable the robot to control itself, the Eqn. 2 is rewritten as,

$$\mathbf{M}_d(\ddot{\mathbf{p}}_d - \ddot{\mathbf{p}}) + \mathbf{D}_d(\dot{\mathbf{p}}_d - \dot{\mathbf{p}}) + \mathbf{K}_d(\mathbf{p}_d - \mathbf{p}) = \mathbf{f}_d - \mathbf{f}_e, \quad (4)$$

where $\mathbf{p}_d, \dot{\mathbf{p}}_d, \ddot{\mathbf{p}}_d$ represent the end-effector's desired position, speed and acceleration. \mathbf{f}_d denotes the desired force. They can be obtained by sampling the trajectories $p_d(t)$ and $f_d(t)$ at time t . In particular, in the autonomous cutting part, we designed the same admittance controller, where \mathbf{f}_d is regarded as a virtual force compensation value (in III-B3). The \mathbf{M}_d , \mathbf{D}_d , and \mathbf{K}_d correspond to the inertia, damping, and stiffness characteristics. Then, we can control the position compliantly based on the F/T sensor's feedback to achieve the desired force.

2) *Dynamic Movement Primitives*: A general DMPs model is expressed as

$$\tau^2 \ddot{y} = \alpha_y (\beta_y (g - y) - \tau \dot{y}) + u(s) + C_t, \quad (5)$$

where y is the demonstration trajectory, the $\alpha_y, \beta_y > 0$ are the coefficients, and $u(s)$ represents the nonlinear basic function superposition term, which enables the trajectory to converge from y to the goal g , and s represents the phase variable. The C_t is the constrained coupling term for obstacle avoidance [12]. We do not place obstacles on the cutting path ($C_t = 0$). To make Eqn. 5 converge to 0, the canonical system is expressed as $\tau \dot{s} = -\alpha_s s$, where $\alpha_s > 0$ is a coefficient. When $s = 0$, it means that the system converges to the goal g . Therefore, the basic function $u(s)$ is defined as

$$u(s) = \frac{\sum_{i=1}^N \varphi_i(s) \omega_i s}{\sum_{i=1}^N \varphi_i(s)} (g - y_0), \quad (6)$$

where y_0 represents the initial state, ω_i represents the weights of the radial basis functions $\varphi_i(s) = \exp(-h_i(s - c_i)^2)$, the N represents the number of $\varphi_i(s)$, h_i and c_i are the widths and centers of $\varphi_i(s)$ respectively. To ensure that the shape of the entire demonstration trajectory can be learned, the centre of $\varphi_i(s)$ is evenly distributed to the timestamps of the demonstration trajectory. To ensure the convergence speed in the later stage, we set $h_i = N/c_i$. Then, the DMPs can be generalized in a new task to generate a similar shape of trajectory by changing the y_0 and goal g .

3) *Compensation with Memory*: We can control the robot based on the F/T sensor's feedback in III-B1, and imitate the demonstrated trajectory to obtain position trajectory $p_y(t)$ and force trajectory $f_y(t)$ from III-B2. When facing the unknown food items to be cut, the robot can not determine the desired force of the food by $f_y(t)$ because the unknown food items have unknown physical properties, which are directly reflected in the force feedback. Therefore, we can only imitate the demonstrated trajectory $p_d(t)$ as $p_y(t)$ to cut. In this way, the F/T sensor's feedback $f(t)$ when cutting unknown food causes the shape of the original trajectory $p(t)$ to be changed. To ensure that the shape of $p(t)$ is close to the $p_y(t)$ one, we adjust the admittance controller by compensation.

Above all, we utilize the error $e_p(t) = p_y(t) - p(t)$ as the condition for compensation, where $e_p(t)$ represents the error between the imitated position $p_y(t)$ and the current position $p(t)$. Since we need to compensate for the shape of the entire trajectory, the $e_p(t)$ obtains the instantaneous error. We record K historical data of the error for calculation. Similar to the human forgetting curve [25], we set the remembered error $e_r(t)$ as follows,

$$e_r(t) = \sum_{t=0}^K \exp\left(-\frac{t}{S}\right) e_p(t), \quad (7)$$

where S indicates the relative memory strength. Based on previous research on manipulating food through force [26], the virtual force f_v obtained based on the error $e_r(t)$ is utilized as the compensation force, which can be expressed as a position-to-force mapping relationship by the momentum theorem and impulse theorem as follows:

$$f_v(t) = H(e_r) \frac{m e_r(t)}{T^2}, \quad (8)$$

where m represents the mass of the knife, T represents the sampling period. However, in extreme cases, T is too small

and $e_r(t)$ is too large, resulting in $f_v(t)$ becoming unreasonably large. We design the heuristic constraint $H(e_r) = \exp(-e_r/h)$ to scale $f_v(t)$. We can regard $f_v(t)$ as a position correction term for the admittance controller derived based on $e_r(t)$. The $f_v(t)$ can not be added directly to $f_y(t)$ as compensation because $f_y(t)$ can not be utilized as a reference for the unknown food. To track the force trajectory, the shape of $f_y(t)$ is considered as a reference for the compensation force. We take the force derivative $f'_y(t)$ as the shape of the force $f_y(t)$, and replace $f'_v(t)$ to $f'_y(t)$ to achieve the purpose of shape approximation. The force $\hat{f}_v(t)$ with shape compensation is as follows,

$$\hat{f}_v(t_n) = \alpha (f_v(t_{n-2}) + 2T f'_y(t_n)) + (1 - \alpha) f_v(t_n), \quad (9)$$

where $\alpha \in [0, 1]$ expresses the weight, $f'_y(t)$ represents the derivative of imitated force $f_y(t)$ at the current moment. n represents the discretization step. Lastly, we obtain the compensated force trajectory $\hat{f}_v(t)$, which can adapt the robot to the food items with unknown physical properties by changing the desired force \mathbf{f}_d in III-B1. The force and position are related in Eqn. 4, where the increase of $|\mathbf{f}_d - \mathbf{f}_e|$ speeds up the convergence of the position error $(\mathbf{p}_d - \mathbf{p})$, which has a similar meaning to $e_r(t)$. The reduction of the position error $e_r(t)$ reduces the influence of the virtual force trajectory $f_v(t)$ in Eqn. 8, which means that the shape of desired force converges to the shape of force trajectory $f_y(t)$.

4) *Constraint-based Re-planning*: Since the environment is not static, we need to make the model adaptive to the dynamic environment. The imitated trajectory $(p_y(t), f_y(t))$ in Eqn. 5 requires a given target g and the initial state y_0 , both of which are dynamic variables. Therefore, we need to use DMPs to re-plan a new trajectory during the execution of the robot. However, the time consumed in executing DMPs at each step seriously affects the working efficiency of the robot, so constraints on g and y_0 are considered to activate re-planning. The activation conditions are defined as **CONDITION 1** and **CONDITION 2** conditioned by the food's state.

CONDITION 1: dynamic position of the food items. The position of the food affects the change of $p_y(t)$. The position of the chopping board for fixing food is easy to obtain, we use it as the goal \mathbf{p}_g . Based on the fixed given initial state \mathbf{p}_{y_0} and the goal \mathbf{p}_g , we can use DMPs to re-plan the trajectory. The Eqn. 5 is rewritten in positional style as follows,

$$\tau^2 \ddot{y} = \alpha_y (\beta_y (1_{d < \delta} \mathbf{p}_g + 1_{d \geq \delta} \mathbf{p}_c - y) - \tau \dot{y}) + u(s), \quad (10)$$

where \mathbf{p}_g represents the target position previously recorded by DMPs, and \mathbf{p}_c represents the current position of the food. The indicator function $1_{d < \delta}$ and $1_{d \geq \delta}$ activates \mathbf{p}_g and \mathbf{p}_c through distance $d = \|\mathbf{p}_g - \mathbf{p}_c\|_2$ and threshold δ . In particular, $1_{d < \delta}$ does not repeatedly activate \mathbf{p}_g , and $1_{d \geq \delta}$ refreshes $\mathbf{p}_g = \mathbf{p}_c$.

CONDITION 2: thickness of the food items. Since the thickness of the food item is unknown, we divide the trajectory into two parts: reaching and cutting. To ensure the knife skills are fully implemented when cutting, the $f_y(t)$ is considered as a sign of phase division and the thickness can be found during the reaching phase. When $\mathbf{f}_y > \epsilon$ for the first time, the food items are touched, the time is considered as a segment node t_s , where the ϵ represents a threshold close to zero. The F/T sensor's feedback $\mathbf{f}_e > \epsilon$ before (early touch) or after (delayed touch) t_s , which means that trajectory y needs to be re-planned. By determining \mathbf{f}_e , we can obtain the tool position error $e_k = p(t) - p_y(t_s)$, which is added to the cutting phase as compensation,

$$\hat{p}_y(t) = e_k + p_y(t), \quad (11)$$

where $p_y(t)$ is regarded as the trajectory generated by DMPs in the original cutting phase, $\hat{p}_y(t)$ represents the compensated trajectory. When early touch, e_k is reasonable. When delayed touch, we fix the time t to make the tool touch the food by adding displacement Δp to p_d in Eqn. 4, which can refresh the value of $p(t)$ when restarting t after the touch.

Finally, the proposed algorithm flow is shown in Algorithm 1. Due to the changes caused by node t_s , the target g and the initial state y_0 are defined in two phases respectively. In the reaching phase, \mathbf{p}_{y_0} remains unchanged, $\mathbf{p}_g = p_y(t_s)$. In the cutting phase, $\mathbf{p}_{y_0} = p(t)$, $\mathbf{p}_g = \hat{p}_y$.

Algorithm 1: Constrained Re-planning of DMPs

```

 $Y(p, f, t) \leftarrow p_d(t), f_d(t)$  from one-shot demonstration
 $y \leftarrow Y(p, f, t)$ , generate  $p_y(t), f_y(t)$  in Eqn. 5
 $t \leftarrow 0, g_f \leftarrow \text{false}$ 
//  $g_f$  is the gate used to switch in CONDITION 2
while true do
    update  $p_y(t)$  by CONDITION 1 based on Eqn. 10
    if  $t \leq t_s$  then
        update  $p_y(t)$  from Eqn. 10
    else
        update  $p_y(t)$  by CONDITION 2 based on Eqn. 11
    end if
    if  $f_e > \varepsilon$  and  $g_f = \text{false}$  then
         $g_f \leftarrow \text{true}, e_k \leftarrow p(t) - p_y(t_s), \mathbf{p}_{y_0} \leftarrow p(t), \mathbf{p}_g \leftarrow \hat{p}_y$ 
        update  $p_y(t), f_y(t)$  from Eqn. 10
    end if
     $f_d \leftarrow \hat{f}_v(t)$  from Eqn. 9
    move robot based on Eqn. 4
end while

```

IV. EXPERIMENTS AND RESULTS

To achieve the proposed methodology, we constructed the experiments to verify it, including experiments on teleoperation and knife skills. The former was used to verify the feasibility of the one-shot demonstration, and the generated trajectory was utilized in the knife skills experiment. The latter was used to verify the superiority of the proposed method.

A. Experimental Setup

All experiments were conducted utilizing the UR3e robot. As shown in Fig. 1, we utilized the robot's built-in F/T sensor to obtain force feedback. The secondary end was also used for autonomous control of the robot. We used the AprilTag markers [27] to obtain the food's posture. We provided some parameter-setting recommendations mentioned in the methodology, which researchers can adjust as needed. In III-A, the transferred matrix \mathbf{Z} was defined as a positive definite matrix where $\mathbf{Z}_m = \mathbf{Z}_s$ was identity matrix. To avoid excessive feedback from the primary end due to \mathbf{f}_e being too large at the secondary end, we set $k_c = j_c = 0.6$. For the admittance controller mentioned in III-B1, we set the inertia coefficient $\mathbf{M}_d = 1$ and the stiffness coefficient $\mathbf{K}_d = 5000$, and the damping was set to critical damping $\mathbf{D}_d = 2\sqrt{\mathbf{M}_d \mathbf{K}_d}$. In III-B2, we set $N = 50$ radial basis functions for Eqn. 6, and set the remaining parameters according to reference [12]. In III-B3, to enable the robot to have the ability to remember position and force, we set the strength $S = h = 0.01$, the sampling time $T = 0.01$ seconds. The period time of reactivating DMPs was $2T$. To make it follow the shape of the

demonstrated force trajectory, we set the weight $\alpha = 0.3$. The parameters mentioned above influenced the change in the force compensation value, specifically the cutting speed of the end-effector. Since these parameters were fixed, controlling the cutting speed was not taken into account. In III-B4, both thresholds were close to zero. The threshold value δ in meters was set to 0.01, and the threshold ε in Newton was set to 0.5.

B. Demonstration from Bilateral Teleoperation

We utilized the designed bilateral teleoperation system in III-A to control the secondary robot to cut food items and obtain demonstration trajectories. The generated demonstration trajectories were discussed in this section and applied in IV-C.

Each set of demonstration trajectories contained position and force. As shown in Fig. 2, we demonstrated the trajectory tracking performance of the bilateral teleoperation system, which includes the position \mathbf{p}_m and force \mathbf{f}_m of the primary and the position \mathbf{p}_s and force \mathbf{f}_s of the secondary of one knife skill. We could see that \mathbf{p}_s can track the \mathbf{p}_m . This was because the designed system was a local control system without time delay. The system's transparency could be guaranteed and generate the same effect as drag teaching. Similar to the drag teaching, the force \mathbf{f}_m of the primary included the force feedback of the secondary and the force applied by the human. Unlike drag teaching, the force \mathbf{f}_s of the secondary was purely the contact force between the manipulator and the environment, which can be utilized as a demonstration trajectory to reflect the change of force during the cutting process.

To show the superior performance of the proposed framework, we designed different knife skills. The knife skills we developed included: **cutting off** (y_1), which divided the food items into two parts; **cutting in** (y_2), which cut into the food items at an angle while keeping the food items intact, such as cutting into bread in order to add fillings; **shearing** (y_3), which cut along the food's surface without cutting deeper into the food items, such as cutting a cross in the surface of dough to enhance the appearance. The cut food items utilized to generate the demonstration trajectories was discussed. We considered the food with large surface tension to be cut for generating demonstration trajectories. This paper chose lemons as the cutting food for the experiment.

Finally, as shown in Fig. 3, the designed system generated one demonstration trajectory for each knife skill. We observed that the time taken by y_3 was the shortest among the trajectories, as it merely created a shallow cut on the food's surface. Similarly, from the Z-axis position trajectory, we could see that the y_3 has the shallowest time, while the y_1 was the deepest in the Z-axis. This was because y_1 was used to completely sever the food items, requiring the knife to make contact with the bottom plane. The X-axis and Y-axis represented the cutting direction of the knife. Fluctuations in the X-axis position, coupled with the stability of the Y-axis position, indicated that the knife was cutting in the X-axis direction. Meanwhile, fluctuations in the Y-axis rotation signified the knife moving downward as it penetrated the food items. In particular, since y_2 was cut obliquely into the food items, it had significant changes in its X-axis rotation and Y-axis position. All of the knife skills showed no significant change in Z-axis rotation because they had no churning action.

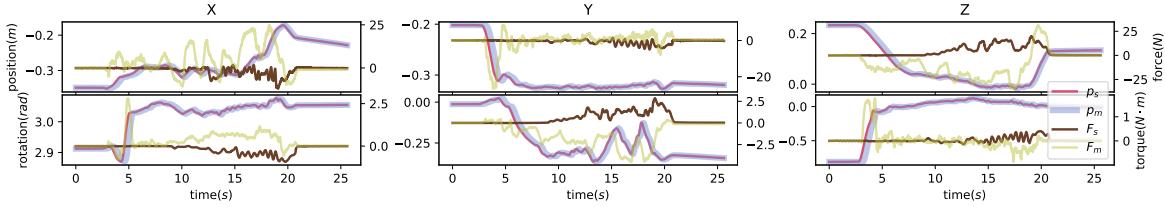


Fig. 2. The bilateral teleoperation system tracking results include the posture q_m of the master manipulator, the feedback f_m of the F/T sensor, the posture q_s of the slave and the feedback f_s of the F/T sensor.

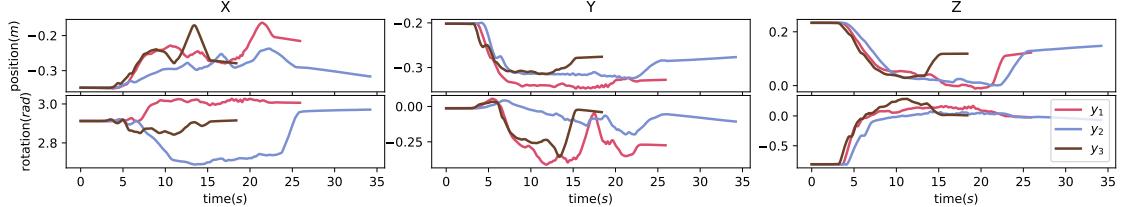


Fig. 3. The trajectory of the manipulator during the execution of knife skills, including three knife skills, each with varying execution times and motion trajectories.

C. Knife Skills Imitation

In this section, we presented the results of the framework proposed in III-B, including the use of various knife skills and cutting of unknown food items.

First, we chose 14 everyday food items utilized for testing in this experiment shown in Fig. 4. In particular, since our framework focused on the learning of knife skills and did not consider the influence of unknown internal obstacles, none of the food had a hard pit. All the physical properties of the food items were unknown. The position of the food was random, and the range of the position was the reachable range of the manipulator. We selected ± 10 cm within the reachable range and ± 20 degrees in the rotation direction for testing. The initial posture was that the food was placed naturally on the chopping board. As shown in Fig. 1, they were fixed by a fixture to prevent them from falling off during cutting.

Then, we designed the experiment to verify the effect of the proposed framework. Especially, we selected one knife skill (y_1) outlined in IV-B for exhibition. The choice of knife skill was beyond the scope of the proposed framework, as the manipulator did not automatically determine the appropriate knife skill. Instead, the requirements were specified manually. The methods compared in the experiment include imitating the demonstration trajectory with pure DMPs (p-DMPs), DMPs with force compensation method (f-DMPs), and the entire proposed framework (cr-DMPs). Since the position of the food items was random, the initial imitation trajectory in p-DMPs and f-DMPs underwent a re-planning to obtain the reasonable imitation trajectory.

As shown in Fig. 5, we presented three representative food items (apple, kiwi, fish) and three selected knife skills (y_1 - y_3) applied in cr-DMPs. The time in each knife skill represented the state of all food in each time period. To show the performance difference between the methods, as shown in Fig. 6, we selected one food item (apple) and a common knife skill (y_1) to compare all three methods. In Fig. 6(a), p-DMPs and f-DMPs planned their trajectories based on the initial position of the food, while the trajectory of cr-DMPs changes in real-time because of the re-planning method. In Fig. 6(b), we showed the force trajectory for the demonstration and

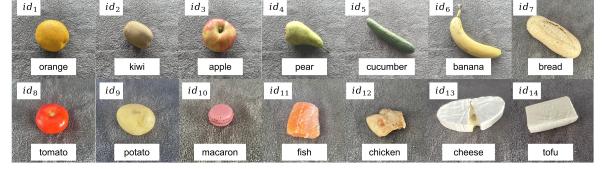


Fig. 4. The food items being cut. They were not familiar with the proposed framework, which included everyday items such as fruits, meats, and snacks.

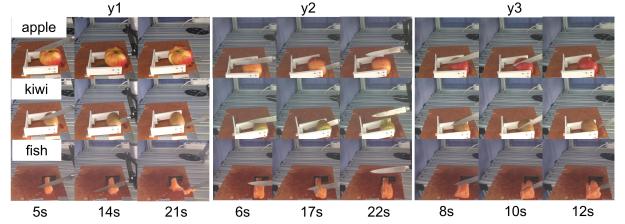


Fig. 5. The performance of the knife skills. We selected three types of food items with varying degrees of softness to demonstrate the process of cutting. The representation of all mentioned food items can be seen in the [video](#).

the one for cutting the apple, where \hat{f}_v represented the force compensation values in f-DMPs and cr-DMPs. in Fig. 6(c), f_{y_1} represented the demonstration force trajectory. We compared the performance of each factor in Equ. 9 in p-DMPs to compare their importance.

Finally, we utilized p-DMPs, f-DMPs, and cr-DMPs to test the food items in Fig. 4 independently five times. If the cutting requirements were met, the test was considered a success, otherwise, it failed. As shown in Fig. 7, we displayed the test success rates of all methods, with each set of bars representing a trajectory. Table I showed the mean values of cosine distance e_f of the force on the Z-axis in five tests. The closer the feedback force f_e was to the demonstration force f_y , the closer the e_f was to 1. **It was difficult to quantify the various properties of all food items (hardness, surface tension, density, brittleness, etc.), as they depended on factors such as where the food was grown, the season, and the size of the food.**

For the cutting task, we quantified the hardness of the food w as a reference, where w is defined as the force required to press the food vertically 5 mm using the same tool (a circular flat surface with a diameter of 1 cm).

D. Discussion

In IV-B, we generated demonstration trajectories using the bilateral teleoperation system and discussed the stability and transparency of this approach. The tracking performance of the proposed method met the framework's requirements, allowing it to generate both position and pure force trajectories.

1) *Force Compensation:* In Fig. 6(b), p-DMPs had no \hat{f}_v because it had no force compensation method deployed. In p-DMPs vs. f-DMPs, the shapes of the forces on the X-axis were similar, and on the Z-axis, the force shape of f-DMPs during cutting was close to the demonstrated force one, while the force of p-DMPs during cutting was too large and could not be maintained. For cr-DMPs, the effect of force compensation was similar to that of f-DMPs. In Fig. 6(c), the f'_y was used to guide the shape of the force. It varied greatly when the knife was in contact with food items, and its influence was reduced in the force compensation due to the scaling effect of $2T$ in Equ. 9, for example, in the cutting direction, it could generate a mean force between -1 N and 1 N . For f_v in Equ 8, it represented the correction term for the position error, eliminating the effect of f'_y , which could produce a force compensation value of -4 N to 4 N . Therefore, the compensation method we proposed for the admittance controller was to use f_v as the dominant factor. Moreover, the f'_y had a relatively small influence in \hat{f}_v and had a guiding role in the shape of the trajectory. As shown in Fig. 7, for the all mentioned knife skills, the cr-DMPs had a better test result than other methods, which adapted well to food items with different physical properties. Since the meat was soft and had some fiber links inside, all methods failed several times in manipulating meat (id_{11} , id_{12}). For fruits (id_1 to id_6), the cr-DMPs and f-DMPs performed better than p-DMPs in all knife skills because they deployed the force compensation policy that adapted to different stiffness food. Similarly, the f-DMPs performed better than p-DMPs in the food with the hard surface (id_{10}) and the soft surface (id_{14}).

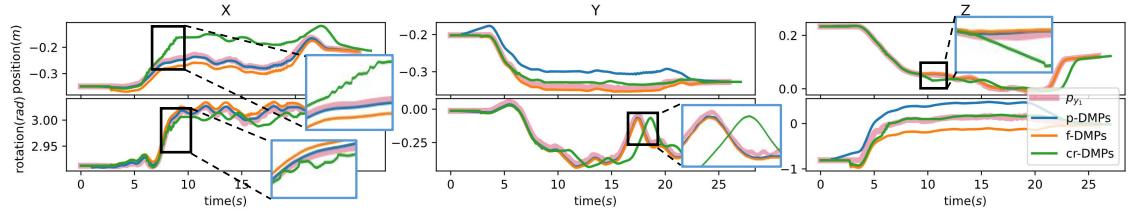
Additionally, we showed the reference hardness and related force compensation performance in Table I. Combined with the reference hardness w , we can see that when w was different, the cutting performance was similar (id_4 vs. id_{14}). In particular, when cutting on the surface of food (see p-DMPs with y_3), the foods (id_1 , id_3 vs. id_2 , id_4) with large w were not easy to cut, this was because y_3 only pressed and slid on the food surface once, and it was not easy to succeed on food with a large w . The reference quantitative factor w we proposed was used as a one-sided discussion, the performance was affected by other physical factors, for example, for foods with similarly large w (id_3 vs. id_{10}), the more fragile food (id_{10}) was not easy to cut; for foods with similarly small w (id_{13} vs. id_{14}), the food (id_{13}) with greater viscosity was not easy to cut. For the e_f , the temporal offset of the cr-DMPs could be obtained from the position on the Z-axis, thus aligning it with other trajectories. We focused on the force in the cutting direction (Z-axis) where the variation was greatest. Overall, the e_f of p-DMPs was low and the e_f of cr-DMPs was high for all food items and all skills, which was because p-DMPs had no force compensation and re-planning. For hard food items, such as apple (id_3), the e_f of p-DMPs was close to 0.9, which means that the force

feedback of food with high hardness was obvious and did not require much compensation. For soft food items, such as cheese (id_{13}), the e_f of p-DMPs was significantly smaller than that of other models, which reflects the importance of force compensation. In particular, id_{11} and id_{12} were meat, and all models had a low success rate for all knife skills applied on them. For id_{11} and id_{12} , it can be seen that the e_f was lower when the success rate was lower, which means that not all knife skills can be successfully applied to all food items (such as food with complex fiber structures inside). However, based on p-DMPs vs. cr-DMPs, we got that the imitation of force trajectory shape can improve the success rate of knife skills applied to the food items.

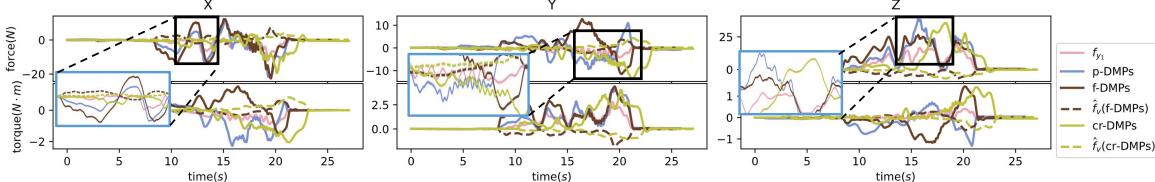
Finally, the amplitude of force compensation affected the cutting speed of the manipulator. We discussed this using knife skill y_1 as the representative and gave the average cutting speed of some food items on the Z-axis. For the hard food item (id_3), the cutting speed of p-DMPs was 0.941 cm/s , the cutting speed of f-DMPs was 1.357 cm/s , and that of cr-DMPs was 1.418 cm/s . The cutting speed of f-DMPs and cr-DMPs were faster than that of p-DMPs because the former deployed the force compensation mechanism and obtained a larger virtual force. Due to the deployment of re-planning, cr-DMPs cut slightly faster than f-DMPs, depending on the magnitude of the change in food location. For the soft and thin food item (id_{11}), the cutting speed of p-DMPs was 0.807 cm/s , the cutting speed of f-DMPs was 1.106 cm/s , and that of cr-DMPs was 1.291 cm/s . We could see that their cutting speed was less than that of hard food items. This was because the force used to cut soft food was lower than that used to cut hard food, so the force required to compensate was lower, resulting in a lower cutting speed. Similar to the performance when cutting hard food, the deployment of force compensation and re-planning method accelerated the cutting speed.

2) *Re-planning with DMPs:* As shown in Fig. 5, it can be seen that the proposed framework enabled the robot to reproduce knife skills on unknown food items using similar paths. Regarding stiffness, the apple was hard, the kiwi was soft, and the fish was deformable. The three knife skills had different cutting paths. As can be seen from the corresponding time regions, combined with Fig. 3, the proposed framework allowed the robot to reproduce knife skills on unknown and different food items utilizing similar paths. Since the food's information was unknown, the re-planning method proposed in III-B4 was used to activate DMPs, which occupied the time of the original planned trajectory. Therefore, the knife posture was different for each food item at the same time.

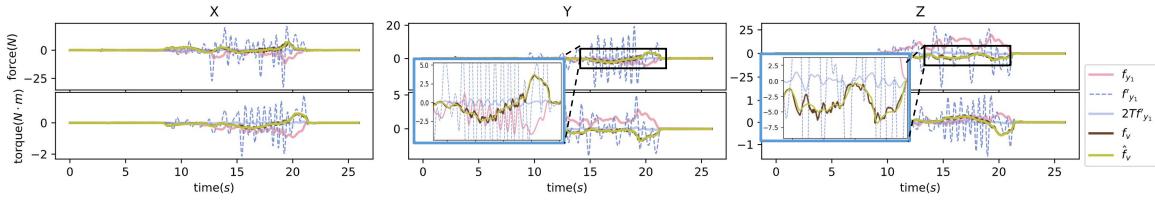
As shown in Fig. 6(a), We could see that the X-axis position and rotation of cr-DMPs fluctuated significantly compared to p-DMPs and f-DMPs between 5 and 8 seconds. This was because the posture of the cutting board that receives the food changed during this period. Other methods had no re-planning constraints and still maintained the shape of the original trajectory. It was noticed that in the Z-axis position, cr-DMPs dropped at a certain speed between 8 and 10 seconds, which means that it did not detect the food items at the moment and explored the thickness of the food items by pressing down. In the same period of time, p-DMPs and f-DMPs followed the original trajectory to complete the cutting action. Due to the thickness of the food items, the exploration of cr-DMPs occupied part of the original trajectory, causing the subsequent trajectory to lag, which can be seen from the peak between 15 and 20 seconds of rotation on the Y-axis.



(a) Each of the three mentioned methods was used to apply the same knife skill (y_1) to the same food item (apple). The results showed different shapes of the knife skill on the posture. Zoomed-in details highlight the differences between the various methods.



(b) The F/T sensor's feedback of using the same knife skill on different models. The zoomed-in details highlight the differences between the various methods. The method p-DMPs did not include the force compensation \hat{f}_v .



(c) The performance of p-DMPs using the compensation method was used to compare various factors (f_y vs. \hat{f}_v vs. $2Tf'_y$) in Eqn 9.

Fig. 6. The (a) showed the position information, the (b) showed the force information, and the (c) showed the comparison of various factors.

TABLE I

MEAN COSINE DISTANCE e_f BETWEEN DEMONSTRATION FORCE f_y AND FEEDBACK FORCE f_e ON Z-AXIS, AND THE REFERENCE HARDNESS w .

	id_1	id_2	id_3	id_4	id_5	id_6	id_7	id_8	id_9	id_{10}	id_{11}	id_{12}	id_{13}	id_{14}	
w (unit:N)	18.70	13.45	20.32	17.72	17.03	17.56	18.30	14.53	14.94	19.21	10.16	11.17	6.76	5.23	
p-DMPs	y_1	0.815	0.756	0.860	0.813	0.841	0.813	0.832	0.831	0.834	0.735	0.791	0.801	0.759	0.817
	y_2	0.804	0.839	0.853	0.868	0.814	0.800	0.828	0.743	0.797	0.755	0.764	0.735	0.840	0.811
	y_3	0.787	0.823	0.785	0.837	0.851	0.798	0.774	0.763	0.791	0.721	0.753	0.807	0.788	0.819
f-DMPs	y_1	0.932	0.907	0.927	0.916	0.906	0.891	0.889	0.936	0.929	0.921	0.907	0.910	0.899	0.909
	y_2	0.903	0.934	0.913	0.909	0.918	0.910	0.915	0.889	0.901	0.928	0.907	0.896	0.913	0.901
	y_3	0.908	0.910	0.893	0.914	0.891	0.922	0.931	0.901	0.904	0.859	0.867	0.864	0.903	0.916
cr-DMPs	y_1	0.912	0.924	0.933	0.936	0.923	0.904	0.914	0.906	0.909	0.910	0.901	0.924	0.936	0.918
	y_2	0.925	0.911	0.906	0.941	0.915	0.929	0.902	0.932	0.922	0.920	0.930	0.917	0.933	0.912
	y_3	0.942	0.915	0.895	0.899	0.929	0.921	0.908	0.930	0.916	0.893	0.916	0.927	0.931	0.929

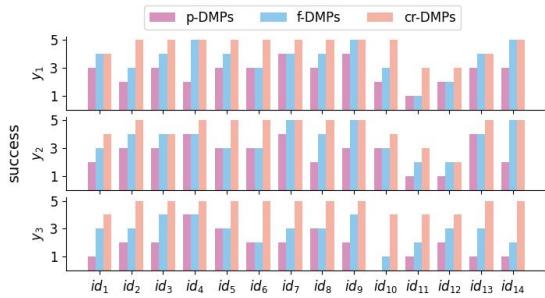


Fig. 7. The test results for all the food items mentioned in Fig. 4. The horizontal axis shows the identification of different food, the vertical axis shows the number of successful tests, and the different lines show the test results of different knife skills. We tested each food item five times using each method.

Then, the knife skills y_1 and y_2 had similar trajectory, which results in the y_2 appearing when testing y_1 or vice versa.

Therefore, as shown in Fig. 7, for y_1 and y_2 , all methods had the similar test result. In y_3 , compared to y_1 and y_2 , the p-DMPs and f-DMPs had significantly lower success rates because y_3 only sliced the surface of the food and needed to explore the surface first. The re-planning policy was not employed on the p-DMPs and f-DMPs, therefore, they would follow the original generated trajectory. In contrast, y_1 and y_2 had low cutting depths on the Z-axis. Therefore, for p-DMPs and f-DMPs, there were more successful times than the test on y_3 . Thanks to the re-planning policy, the cr-DMPs had the best result on y_3 , and similar to f-DMPs, because of the force compensation policy deployment, it could distinguish y_1 and y_2 by force feedback, so it has a better success rate in these two knife skills than p-DMPs.

Finally, as shown in Table I, the f-DMPs and cr-DMPs performed similarly because they both deployed force compensation mechanisms. The difference between them lies in the knife skill y_3 . Since y_3 only made a shallow cut on the surface of the food, the thickness of the food items affected

the model's performance on y_3 . When cutting thinner food items (id_{11} and id_{12}), it can be seen from ef and success rate that f-DMPs could not complete the task well, but the deployment of the re-planning method compensated for the thickness differences to reduce phase errors in the shape of the force trajectory.

In conclusion, through the bilateral teleoperation system, we were able to generate demonstration trajectories that enabled the robot to imitate both position and force trajectory shapes. Force compensation allowed the robot to apply forces closer to human levels, enhancing performance beyond simple tracking. The proposed re-planning constraints enabled the robot to dynamically adjust its trajectory, allowing for flexible adaptation to unknown environments.

V. CONCLUSION AND FUTURE WORK

This work proposes a new framework for daily robotic manipulation of unknown food items, mimicking human cutting techniques and applying learned knife skills to handle unfamiliar food items. A bilateral teleoperation system generates demonstration trajectories, capturing the manipulator's position and force data from the slave. These one-shot demonstrations are then imitated by cr-DMPs, which incorporate force compensation to replicate the force trajectory and include constraints for re-planning the generated paths. The cr-DMPs enable the creation of smooth, high-frequency action sequences without significantly increasing inference time, making them suitable for real-world robotic applications. In summary, the proposed framework offers a gentle, demonstration-based method for manipulating unknown food items daily. Its ability to learn from limited data while maintaining high-quality motion characteristics makes it a promising approach for future developments in autonomous systems.

However, there are several limitations to the proposed approach, which are the focus of ongoing and future work. First, knife techniques vary widely, and the strategies for each technique differ, making it beneficial to classify these techniques to better imitate the corresponding trajectories. Second, while a marker is used on the board to determine the food's posture, it cannot assess its thickness. Integrating visual recognition methods could enhance and expand the information obtained in posture. **Moreover, more complex knife skills can be attempted, such as removing potato sprouts, which require the assistance of precise visual detection methods.** Lastly, the current fixture requires manual operation to hold the food items in place. In the future, a dual-arm framework could be developed, replacing the fixture with additional manipulators to hold the food items autonomously.

REFERENCES

- [1] I. Mitsioni, Y. Karayannidis, J. A. Stork, and D. Kragic, "Data-driven model predictive control for the contact-rich task of food cutting," in *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pp. 244–250, 2019.
- [2] W. Xu, Y. He, J. Li, J. Zhou, E. Xu, W. Wang, and D. Liu, "Robotization and intelligent digital systems in the meat cutting industry: From the perspectives of robotic cutting, perception, and digital development," *Trends in Food Science and Technology*, vol. 135, pp. 234–251, 2023.
- [3] A. Straizys, M. Burke, and S. Ramamoorthy, "Learning robotic cutting from demonstration: Non-holonomic dmps using the udwadia-kalaba method," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5034–5040, 2023.
- [4] X. Mu, Y. Xue, and Y.-B. Jia, "Robotic cutting: Mechanics and control of knife motion," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 3066–3072, 2019.
- [5] G. Gonzalez, M. Agarwal, M. V. Balakuntala, M. Masudur Rahman, U. Kaur, R. M. Voyles, V. Aggarwal, Y. Xue, and J. Wachs, "Deserts: Delay-tolerant semi-autonomous robot teleoperation for surgery," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 12693–12700, 2021.
- [6] C. C. Beltran-Hernandez, N. Erbetti, and M. Hamaya, "Sliceit!: Simulation-based reinforcement learning for compliant robotic food slicing," in *ICRA2024 Workshop on Cooking Robotics: Perception and Motion Planning*, 2024.
- [7] Z. Xu, Z. Xian, X. Lin, C. Chi, Z. Huang, C. Gan, and S. Song, "Roboninja: Learning an adaptive cutting policy for multi-material objects," in *Proceedings of Robotics: Science and Systems (RSS)*, 2023.
- [8] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," in *Conference on Robot Learning (CoRL)*, 2024.
- [9] A. O'Neill et al, "Open x-embodiment: Robotic learning datasets and rt-x models : Open x-embodiment collaboration0," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903, 2024.
- [10] A. Padalkar, M. Nieuwenhuisen, S. Schneider, and D. Schulz., "Learning to close the gap: Combining task frame formalism and reinforcement learning for compliant vegetable cutting," in *Proceedings of the 17th International Conference on Informatics in Control, Automation and Robotics - ICINCO*, pp. 221–231, INSTICC, SciTePress, 2020.
- [11] O. M. Omisore, S. Han, J. Xiong, H. Li, Z. Li, and L. Wang, "A review on flexible robotic systems for minimally invasive surgery," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 1, pp. 631–644, 2022.
- [12] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors," *Neural Computation*, vol. 25, no. 2, pp. 328–373, 2013.
- [13] Y.-H. Hwang, S.-R. Kang, S.-W. Cha, and S.-B. Choi, "A robot-assisted cutting surgery of human-like tissues using a haptic master operated by magnetorheological clutches and brakes," *Smart Materials and Structures*, vol. 28, no. 6, p. 065016, 2019.
- [14] K. Zhang, M. Sharma, M. Veloso, and O. Kroemer, "Leveraging multi-modal haptic sensory data for robust cutting," in *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pp. 409–416, 2019.
- [15] J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, and Y. Mezouar, "Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.
- [16] R. Rahal, F. Abi-Farraj, P. R. Giordano, and C. Pacchierotti, "Haptic shared-control methods for robotic cutting under nonholonomic constraints," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8151–8157, 2019.
- [17] I. Mitsioni, Y. Karayannidis, and D. Kragic, "Modelling and learning dynamics for robotic food-cutting," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, pp. 1194–1200, 2021.
- [18] A. Mason, D. Romanov, L. E. Cordova-Lopez, and O. Korostynska, "Smart knife: Integrated intelligence for robotic meat cutting," *IEEE Sensors Journal*, vol. 22, no. 21, pp. 20475–20483, 2022.
- [19] L. Han, H. Wang, Z. Liu, W. Chen, and X. Zhang, "Vision-based cutting control of deformable objects with surface tracking," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 4, pp. 2016–2026, 2021.
- [20] Y. Liu, C. Guo, and M. J. Er, "Robotic 3-d laser-guided approach for efficient cutting of porcine belly," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 5, pp. 2963–2972, 2022.
- [21] R. Wu and A. Billard, "Learning from demonstration and interactive control of variable-impedance to cut soft tissues," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 5, pp. 2740–2751, 2022.
- [22] J.-K. Lee and J.-H. Ryu, "Learning robotic rotational manipulation skill from bilateral teleoperation," in *2022 19th International Conference on Ubiquitous Robots (UR)*, pp. 318–325, 2022.
- [23] K. Liang, Y. Wang, L. Pan, Y. Tang, J. Li, Y. Lin, and M. Pan, "A robot learning from demonstration method based on neural network and teleoperation," *Arabian Journal for Science and Engineering*, vol. 49, no. 2, pp. 1659–1672, 2024.
- [24] A. M. LYAPUNOV, "The general problem of the stability of motion," *International Journal of Control*, vol. 55, no. 3, pp. 531–534, 1992.
- [25] D. E. Trahan and G. J. Larabee, "Effect of normal aging on rate of forgetting," *Neuropsychology*, vol. 6, no. 2, p. 115, 1992.
- [26] X. Mu, Y. Xue, and Y.-B. Jia, "Dexterous robotic cutting based on fracture mechanics and force control," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 4, pp. 5198–5215, 2024.
- [27] J. Wang and E. Olson, "Apriltag 2: Efficient and robust fiducial detection," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4193–4198, 2016.