2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Dynamic Scoop-and-Flick Manipulation for Rapid Non-Prehensile High-Arc Object Transfer

Gijae Ahn*, Junwoo Lee*, Seunghwa Oh, Mujin Shin, Seung-Joon Yi, and Jungwon Seo

***Abstract*— This study presents *dynamic scoop-and-flick manipulation*, a robotic technique that achieves desired projectile motions of target objects through rapid, non-prehensile physical interactions. The method allows a robot to scoop objects resting on a surface and quickly launch them into projectile trajectories. We formulate a theoretical model of the technique and realize it through a hybrid approach that combines model-based reasoning and data-driven learning. The advantages—namely, rapid and accurate pick-and-place with reduced planning complexity—are validated in experiments conducted with a particularly challenging class of objects: low-profile items with small thickness.**

I. Introduction

Our interest lies in developing a fast and efficient method for robotic pick-and-place manipulation. Specifically, we aim to achieve desired projectile motions of target objects through rapid non-prehensile physical interactions. This paper presents our first step toward that goal, a strategy we refer to as *dynamic scoop-and-flick manipulation*. As illustrated in Fig. 1, the robot begins by scooping an object resting on the ground surface and then rapidly transitions to flicking it away. We focus on thin objects, which are particularly challenging to handle using conventional pick-and-place techniques. The proposed approach offers distinct advantages: it enables high-speed, non-prehensile operation and eliminates the need for sophisticated grasp planning, in contrast to prehensile approaches. Potential applications include high-volume pick-and-place scenarios where both efficiency and speed are paramount.

Our approach exemplifies dynamic manipulation, which explicitly incorporates object dynamics into the manipulation process. Early demonstrations include part reorientation through tray tilting [1] and robotic juggling [2]. Building on these foundations, [3] examined the planning and control of planar dynamic manipulation, later extended to three dimensions in [4], enabling a variety of dynamic tasks with a simple robot system. To address the inherent complexity of such problems, low-dimensional models such as the acrobot have proven effective; for example, see [5]. More recent studies have expanded dynamic manipulation to applications including in-hand repositioning [6] and heavy object transport [7]. The significance of dynamics is also evident in robotic locomotion research. A notable example is passive dynamic walking [8], which conceptualized walking as a gravity-driven process without active control.

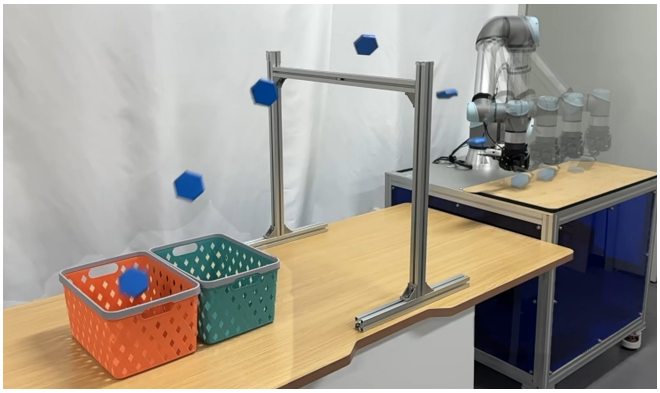


Fig. 1. Time-lapse of dynamic scoop-and-flick, showing the object is scooped with the flat fingertip, flicked upward, and subsequently crosses over the bar into the target bucket.

A particularly relevant form of dynamic manipulation for the presented study is robotic throwing. Robotic throwing has attracted considerable attention due to both its methodological complexity and practical importance. Design-oriented approaches include the development of a high-speed manipulator capable of throwing balls [9], a dexterous planar ball-throwing robot [10], and a hybrid gripper for grasp-and-throw tasks [11]. Throwing has also been demonstrated using a soft robotic arm [12] and a hydraulic arm [13], both exploiting centrifugal forces. Throwing is inherently sensitive, as its outcomes are strongly influenced by errors and uncertainties. To address this, [14] demonstrated that throwing motions can be designed to be robust to release uncertainty, while [15] showed that tracking error prediction and model predictive control can enhance motion accuracy. Recently, data-driven and learning-based control methods have shown promise in advancing robotic throwing capabilities. For example, [16] proposed a framework for collaborative grasping and throwing learning. [17] presented a throw control strategy that can learn effectively from limited data.

Our approach differs from much of the existing literature introduced in that it manipulates objects in a non-prehensile manner, without requiring firm grasps prior to throwing (Fig. 1). Despite its advantages such as high-speed operation and simplified grasp planning, it introduces challenges, particularly in controlling the non-prehensile manipulation

The authors are with the Department of Electric and Electronic Engineering, Pusan National University, Busan 46241, Republic of Korea. (*Corresponding author: Jungwon Seo* junseo.kr@pusan.ac.kr).

This work was supported by RS-2024-00422269 (Alchemist Project Program, Ministry of Trade, Industry, and Energy, Korea); RS-2024-00406796 (KIAT HRD Program for Industrial Innovation); RS-2025-00562200 (NRF grant funded by the Ministry of Science and ICT, Korea); and BK21FOUR, Creative Human Resource Education and Research Programs for ICT Convergence in the 4th Industrial Revolution.

*These authors contributed equally to this work.

process. In the remainder of the paper, our solution will be presented as follows. Sec. II introduces our manipulation model and evaluates its initial feasibility. Sec. III presents the implementation of our controller. Sec. IV reports experimental results.

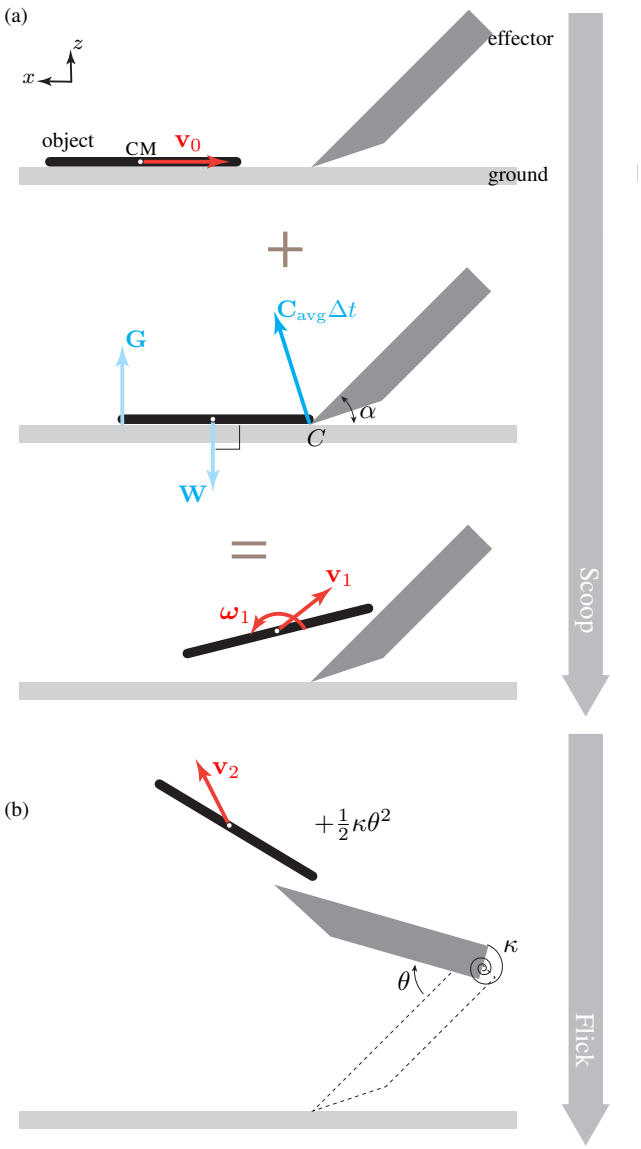


Fig. 2. Scoop-and-flick manipulation as a dynamic process. (a) *Scoop* phase. The object (with CM denoting its mass center) and the effector are about to collide (top). During the collision (middle), the corner C exerts a large impulsive force $\mathbf{C}_{\text{avg}}$, while other forces, including gravity $\mathbf{W}$ and reaction $\mathbf{G}$ at the other tip of the object, are considered non-impulsive. The bottom panel illustrates a feasible post-collision motion. (b) *Flick* phase. The elastic energy stored in a torsional spring with stiffness κ, collocated with the effector's joint, is released and contributes to launching the object with velocity $\mathbf{v}_2$.

II. Model and Feasibility

This section introduces our manipulation model and its initial feasibility analysis.

A. Dynamic Scoop-and-Flick

We investigate the possibility of scooping an object using a single effector of a robot, followed by launching it into projectile motion through a quick flicking action of the effector (Fig. 2). This approach could enable rapid pick-and-place by handling the object in a non-prehensile manner.

We begin by discussing how the initial scoop phase can be realized—that is, how the effector is able to enter the space beneath the object—using principles of rigid body dynamics. In Fig. 2a, an object (recall that we focus on low-profile objects with small thickness), modeled as a line segment on flat ground, is struck by an effector with a flat contact surface moving at velocity $\mathbf{v}_0$. In a frame moving with the effector, the object appears to approach with velocity $\mathbf{v}_0$. The collision occurs at corner C (formed by the effector and ground) over a very short time interval Δt, generating a large impulsive force $\mathbf{C}_{\text{avg}}$, with an impulse of $\mathbf{C}_{\text{avg}}\Delta t$. Non-impulsive forces such as gravity can be neglected due to their negligible impulse compared to the collision. To analyze the post-impact motion, we examine the object's angular momentum about C, denoted $\mathbf{H}_C$. Since $\mathbf{H}_C$ must be preserved and is initially zero before collision—because the initial velocity $\mathbf{v}_0$ is directed toward C for a zero-thickness object—the post-impact twist $[\boldsymbol{\omega}_1 \quad \mathbf{v}_1]^T$ must also satisfy

$$\mathbf{H}_C = I\boldsymbol{\omega}_1 + \mathbf{r}_C \times m\mathbf{v}_1 = \mathbf{0} \tag{1}$$

Here, the first term represents the rotational contribution from $\boldsymbol{\omega}_1$ (with I being the object's moment of inertia), while the second term represents the contribution from the moment of linear momentum about C, where $\mathbf{r}_C$ is the vector from C to $\mathbf{v}_1$, and m is the object's mass. Conservation requires these two terms to cancel, yielding a physically feasible outcome $[\boldsymbol{\omega}_1 \quad \mathbf{v}_1]^T$ depicted in Fig. 2a where the object rotates counterclockwise while its linear momentum balances in the opposite sense. For scooping to succeed, it is necessary to keep the effector's angle of attack α sufficiently shallow. If α is too steep, the horizontal component of $\mathbf{v}_1$ may reverse direction, causing the object to move away from the effector and leading to scooping failure. The analysis presented in this paragraph also appears in [18], where it is applied in the context of prehensile grasping.

Following a successful scoop, the object can be launched into projectile motion through a well-timed flicking action by the effector. One way to achieve this is by utilizing stored elastic energy, similar to a slingshot mechanism. Fig. 2b illustrates this approach using a torsional spring, storing and releasing elastic energy of $\frac{1}{2}\kappa\theta^2$. A portion of this energy is converted into the object's kinetic energy. In summary, the entire scoop-and-flick operation constitutes a fully dynamic process.

B. Initial Feasibility Assessment

The model of dynamic scoop-and-flick manipulation introduced in the previous subsection raises several technical concerns. It can be challenging to precisely predict the outcome based on first principles of the mechanics and dynamics involved in the effector-object interaction, especially when

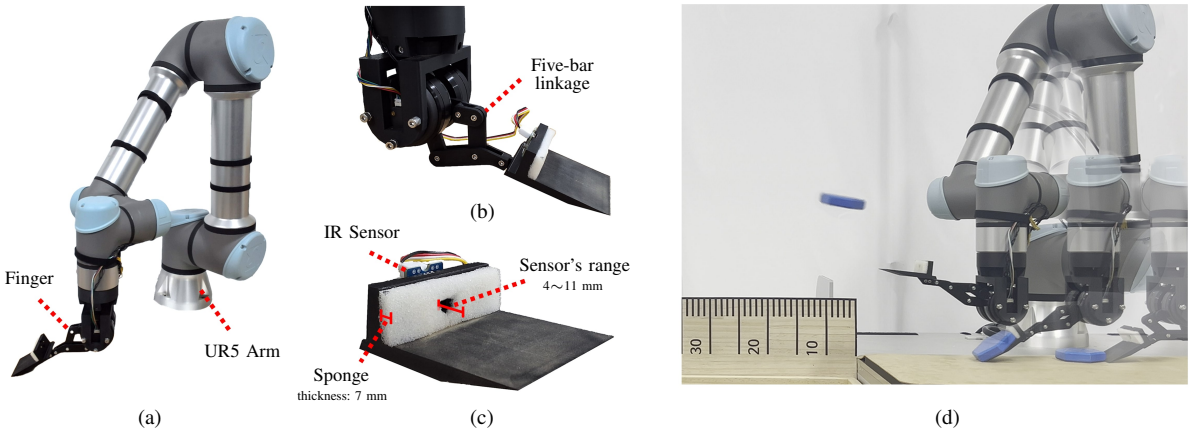


Fig. 3. (a) Our hardware system: a six-DOF arm with a two-DOF finger. (b) The two-DOF direct-drive finger implemented as a five-bar closed linkage. (c) L-shaped fingertip with an embedded IR proximity sensor. (d) The integrated arm-finger system performing a dynamic scoop-and-flick.

friction and compliance play significant roles. These factors introduce uncertainties, making it difficult to determine key outcomes of the motion. As a result, there may be significant variation in the object's projectile behavior.

As a first step to evaluate how significant these concerns are in practice, we configured a hardware system to conduct preliminary tests as follows. A robotic arm is equipped with our custom two degrees-of-freedom (DOF) finger (Fig. 3a), constructed as a five-bar parallel linkage (Fig. 3b). We adopted direct-drive electric motor actuation to enable high-speed motion, elastic energy storage and conversion, and adaptive physical interaction. Fig. 3c shows the L-shaped fingertip, which serves as the effector in Fig. 2. The inner wall of the fingertip houses an IR proximity sensor used to detect whether an object has been successfully scooped. If the object is close to the wall within the sensor's detection range (4-11 mm), the scoop is considered successful. The sensor is protected by a 7 mm thick sponge (note, the thickness was determined to match the sensor's detection range), which also functions as a damper to soften the impact on the object.

We then conducted two sets of tests, with a 3D-printed low-profile prismatic object:

- End-to-end feasibility: First, the system successfully scooped an object resting on a tabletop and flicked it into projectile motion (Fig. 3d). The preliminary success can be attributed to selecting a sufficiently small angle of attack α and to chamfering the object's edges, which together facilitated a smoother scooping action.
- Repeatability of flicking outcomes: The system's performance was found to be repeatable across trials. Specifically, when an object with a diameter of approximately 7 cm was flicked into a sand pool using the same flicking action, the final impact footprints from five trials overlapped after flights of approximately 70 cm in range—the object's maximum horizontal displacement (Fig. 4a).

These findings indicate that if the dynamics of the scoop-and-flick process can be effectively modeled, the overall end-to-end process can be made controllable.

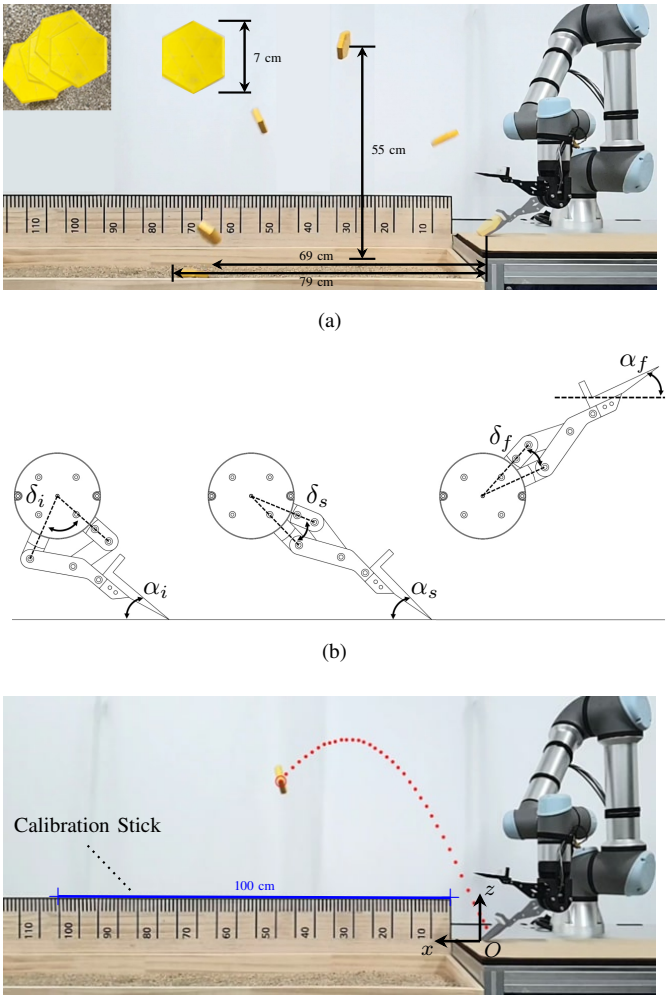


Fig. 4. (a) Final impact footprints of a low-profile test object from five trials, overlapped on a sand pool. (b) Parameters α and δ defining the configuration of the two-DOF finger. (c). Frame-by-frame tracking of the flicked object.

III. Control Implementation

Building on the preliminary feasibility results presented in the previous section, we now provide a detailed description of the implementation of the proposed technique.

A. Measurement of Flicking Dynamics

We collected data to characterize the dynamic manipulation process, with particular focus on the dynamics of the flicking motion. Consider that the configuration of the two-DOF finger is represented by two parameters (Fig. 4b)—α, the angle of attack of the fingertip with respect to the ground surface, and δ, the relative orientation between the two internal links. Initially, the finger is set to the configuration (α_i, δ_i). Using PD setpoint motion control, the finger is then commanded to move to the final configuration (α_f, δ_f), at which point the object will have been flicked away. This motion passes through an intermediate configuration (α_s, δ_s), extending the finger to enable more effective scooping.

Among the configuration parameters (Fig. 4b), we varied the initial and final angle of attack α_i and α_f, respectively. Tests were also conducted with objects of different masses—of the same type shown in Fig. 3 or 4—while holding other parameters fixed. The specific parameter values used are summarized as follows:

- Object mass m_{obj}: 14, 24, 33, 43 g
- δ_i: 83° (fixed)
- α_i: 20° (minimum) to 40° (maximum)
- δ_s: 25° (fixed)
- α_s: Determined such that the fingertip remains in contact with the ground at the value of δ_s. Accordingly, it dependently varies between 29.7° (minimum) and 43.3° (maximum).
- δ_f: 25° (fixed)
- α_f: −10° (minimum) to 5° (maximum). At negative values, the fingertip points upward.
- P-gain: Set to 12 initially and increased to 48 past the intermediate configuration (α_s, δ_s). The value refers to the ratio between motor torque and shaft position error.
- D-gain: Fixed at 0.6, representing the ratio between motor torque and shaft angular velocity error.

In each flicking attempt, we placed an object onto the finger at the intermediate configuration (α_s, δ_s), and then recorded the resulting trajectory $z = ax^2 + bx + c$ using Tracker[1], with respect to the reference frame shown in Fig. 4c. The collected data are summarized in Fig. 5, in the context of the range and maximum height of the flicked objects.

B. Learning-Based Flicking Control

From the collected data $\{(m_{\text{obj}}, \alpha_i, \alpha_f, a, b, c)\}$, we estimate a function $\mathbf{f}$

$$\mathbf{f} : (m_{\text{obj}}, a, b, c) \mapsto (\alpha_i, \alpha_f) \tag{2}$$

which maps the input four-tuple $(m_{\text{obj}}, a, b, c)$—representing the object to be handled and a target trajectory $z = ax^2 + bx + c$—to the output, the optimal finger control parameters, (α_i, α_f). To approximate this function, we employed a simple three-layer neural network architecture, following [12]. The network loss is defined while accounting for the density of the training samples. Specifically, the loss $\mathcal{L}$ is given by the weighted average

$$\mathcal{L} = \frac{\sum w_i e_i}{\sum w_i} \tag{3}$$

where e_i denotes the squared error between the predicted and ground-truth control parameters (α_i, α_f) for the i-th sample. The weight w_i corresponds to the inverse of the sample frequency, normalized using a Gaussian function. Accordingly, samples from less dense regions (see the darker area in Fig. 5) receive larger weights.

[1][online] https://opensourcephysics.github.io/tracker-website/

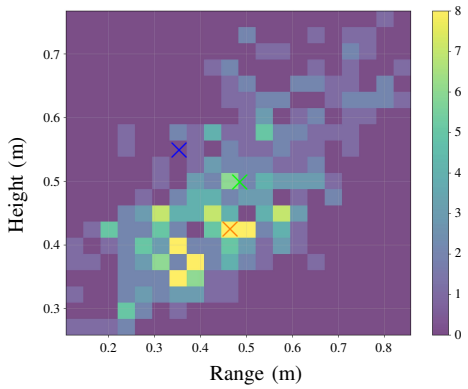


Fig. 5. Distribution of the 440 collected data points. Color indicates the number of occurrences. The × marks denote the query locations used to obtain the learning curves. The queries are selected from a low- (the one marked blue), moderate- (green), and high-density (orange) regions.

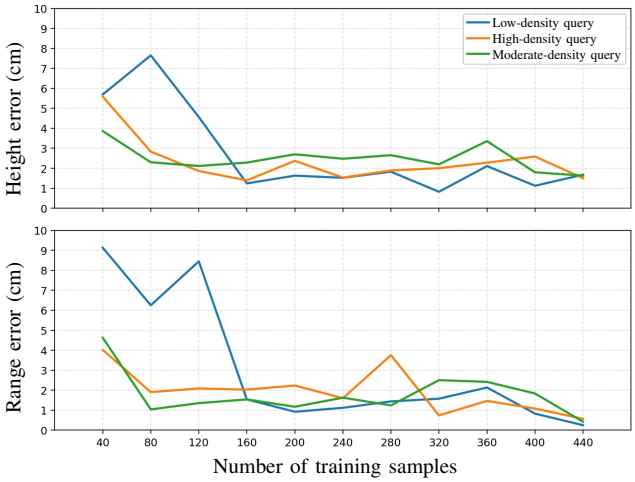


Fig. 6. Learning curves for height errors (top) and range errors (bottom). The colors used—blue, green, and orange—correspond to the low- (blue), moderate- (green), and high-density (orange) queries marked in Fig. 5.

Fig. 6 presents the learning curves, illustrating the prediction performance of the learned $\mathbf{f}$ in terms of the maximum height and range of the resulting projectile motion. Each plot includes three curves, corresponding to three queries selected from the low-, moderate-, and high-density regions of the histogram in Fig. 5. The height and range errors are defined

as the mean absolute error across five trials using an object whose mass was not included in the training samples. The curves demonstrate early saturation across all three queries, with approximately 440 samples required to achieve locally optimal accuracy.

IV. Experiments

This section presents an end-to-end implementation of dynamic scoop-and-flick and reports experimental results.

A. End-to-End Control Protocol

The flicking control discussed in Sec. III is integrated with the arm's motion control. In particular, the arm's tool center point (TCP) located at its wrist is commanded to move horizontally, thereby increasing the object's range, while leaving its maximum height essentially unchanged, in principle.

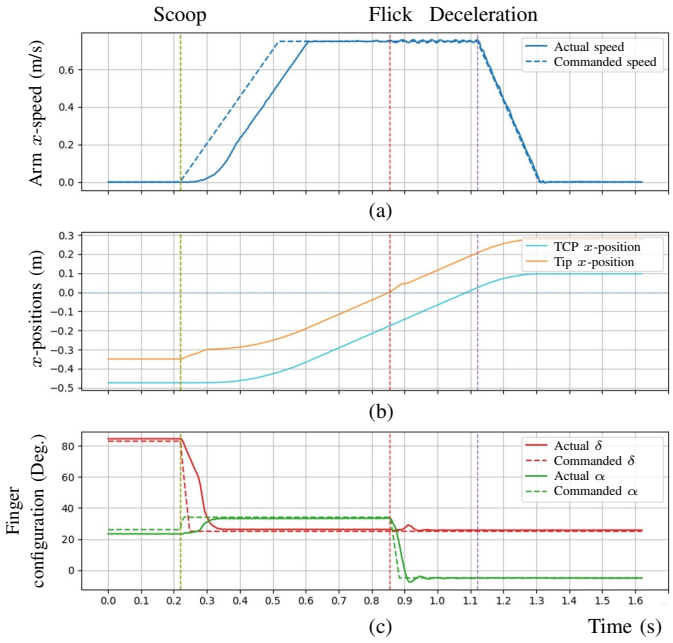


Fig. 7. (a) Horizontal speed profile of the arm, with scoop and flick occurring at approximately 0.22 s and 0.86 s, respectively. Deceleration begins around 1.12 s. (b) Position profiles of the arm's TCP and the fingertip. (c) Finger configuration profile represented by the two parameters α and δ.

However, because the arm is moving, it can be difficult to flick the object at the same horizontal position each time. To resolve this, the fingertip location is tracked at 500 Hz by combining the TCP position with the finger's forward kinematics. This high-frequency monitoring enables the flick command—triggered upon a successful scoop as detected by the proximity sensor—to be executed at nearly the same location each time.

Moreover, the arm is controlled to follow through after the flick command. Specifically, it maintains the commanded horizontal velocity for approximately 0.2 s following the flick before initiating deceleration. This follow-through ensures that the arm's horizontal velocity is fully transferred to the object during the flick. Fig. 7, describing how the arm's state—velocity and position—and the finger's configuration evolve over time, shows an example.

B. End-to-End Experiments

To verify the soundness of the integrated arm-finger system, we conducted two sets of experiments. First, we examined whether a nonzero arm horizontal speed contributes to the object's range as predicted. For example, a flick trial commanded for a 30 cm horizontal range and a 30 cm maximum height resulted in a flight duration of 0.495 s. With the arm speed set to 0.2 m/s, the predicted range increase was $0.495\ \text{s} \times 0.2\ \text{m/s} = 9.9$ cm. Across all such trials with varying flick commands (Table I, rows 1 to 3), the maximum range error—the difference between the predicted and measured range—was 3.7 cm, corresponding to less than 10% of the predicted range.

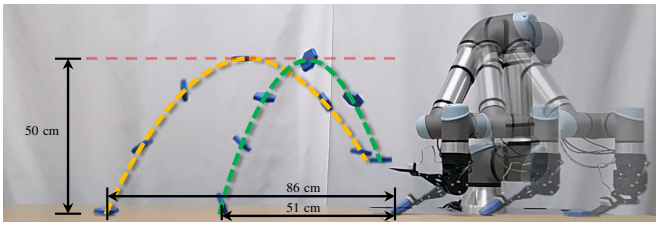


Fig. 8. Projectile trajectories for the same flick command with a stationary arm (green) and a moving arm (yellow), corresponding to rows 4 and 6 of Table I, respectively. The arm's horizontal speed increases the range while leaving the maximum height unchanged.

Second, we varied the arm's speed while keeping the flick command fixed (Fig. 8 and Table I, rows 4 to 7). Even at the maximum speed of 1.0 m/s, the range error was about 5 cm, corresponding to less than 5% of the predicted value.

C. Throw-in-Bucket Tasks

The successful end-to-end experiments motivated us to investigate a more challenging scenario: throwing objects into buckets over a horizontal bar (recall Fig. 1). This task can be viewed as analogous to a high jump combined with horizontal target reaching. The test objects comprise both 3D-printed and real-world low-profile objects. The 3D-printed objects differed from the training set in shape, mass, and the distribution of mass. The real-world objects include a small PCB panel and a carton box.

The finger control parameters (α_i, α_f) and the arm's horizontal velocity v_{arm} are determined in a two-step process. First, the finger parameters (α_i, α_f) are selected using $\mathbf{f}$ in Eq. 2 to generate a trajectory with a peak height sufficient to clear the horizontal bar. We set the target range equal to the target height, but it can also be set differently. Second, based on the resulting flight duration, the arm velocity v_{arm} is computed to ensure that the object's landing position coincides with the center of the bucket.

A trial is considered successful only if the object clears the bar—set at a height of 50 cm—and lands in one of the

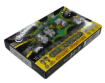
TABLE I

RESULTS OF END-TO-END EXPERIMENTS (ERRORS ARE THE MEAN ABSOLUTE ERROR OVER FIVE TRIALS)

Arm Speed (m/s)	Flight Duration (s)	Predicted Height (cm)	Predicted Range (cm)	Height error (cm)	Range error (cm)
0.2	0.495	30.0	39.9	2.2	3.7
0.2	0.571	40.0	51.4	1.2	1.3
0.2	0.639	50.0	62.8	0.6	1.9
0.0	0.639	50.0	50.0	1.4	0.7
0.2	0.639	50.0	62.8	0.6	1.9
0.5	0.639	50.0	81.9	0.3	5.7
1.0	0.639	50.0	113.9	0.9	5.2

TABLE II

RESULTS OF THROW-IN-BUCKET TASKS

Test Objects	3D-Printed (Used for training)	3D-Printed (Not used for training)	3D-Printed (Not used for training)	PCB 1	PCB 2	3D-Printed (With uneven mass distribution[1])	Carton Box
Dimensions (Diameter × Thickness)	 70 × 16 mm	 70 × 16 mm	 82 × 8 mm	 84 × 12 mm	 84 × 13 mm	 70 × 16 mm	 118 × 10 mm
Mass (g)	24	35	19	21	28	27	23
Success Rate (Bucket 1)	10/10	10/10	10/10	10/10	8/10	4/10	9/10
Success Rate (Bucket 2)	10/10	10/10	9/10	9/10	8/10	3/10	7/10

[1]Point mass (nut) attached off-center.

target buckets—placed at ranges of 115 and 135 cm, respectively. The aggregate results are summarized in Table II and illustrated in Fig. 9. The findings are as follows:

- Objects similar to those used for training show high success rates (Table II, columns 1 to 3). These successful outcomes generalize to the PCB panel objects (Table II, columns 4 and 5), due to their overall similarity to the trained set. The success supports the validity of the simplified planner $\mathbf{f}$, relying only on the object's mass and target location as input.
- The object with uneven mass distribution (Table II, column 6) was the most challenging. First, the scooping outcomes were highly sensitive to the orientation of the attached point mass. Second, in the resulting projectile motion, the object's maximum height was consistently lower than predicted, causing it to fall short of clearing the bar.
- The carton box (Table II, column 7) consistently fell short of its predicted range. While this did not cause noticeable failures for the nearer Bucket 1, it posed difficulties in reaching the farther Bucket 2. This discrepancy can be attributed to air resistance, as the carton box presents a larger face than the other objects.

V. CONCLUSION

We presented a rapid, non-prehensile object transfer method through the development of a dynamic scoop-and-flick strategy, implemented as a hybrid of model-based and data-driven approaches. Experiments demonstrate that a variety of low-profile objects can be accurately picked and placed via this high-speed dynamic process, defined solely by the target location and the object's mass. Future work will extend the method to handle a broader range of object properties such as mass distribution and aerodynamic effects.

REFERENCES

[1] M. A. Erdmann and M. T. Mason, "An exploration of sensorless manipulation," *IEEE J. Robot. Autom.*, vol. 4, no. 4, pp. 369–379, 1988.

[2] A. Rizzi and D. Koditschek, "Progress in spatial robot juggling," in *Proc. 1992 IEEE Int. Conf. Robot. Automat. (ICRA)*, pp. 775–780.

[3] K. M. Lynch and M. T. Mason, "Dynamic nonprehensile manipulation: Controllability, planning, and experiments," *Int. J. Robot. Res.*, vol. 18, no. 1, pp. 64–92, 1999.

[4] J. Z. Woodruff and K. M. Lynch, "Robotic contact juggling," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 1964–1981, 2023.

[5] P. Donner and M. Buss, "Cooperative swinging of complex pendulum-like objects: Experimental evaluation," *IEEE Trans. Robot.*, vol. 32, no. 3, pp. 744–753, 2016.

[6] J. Shi *et al.*, "Dynamic in-hand sliding manipulation," *IEEE Trans. Robot.*, vol. 33, no. 4, pp. 778–795, 2017.

[7] A. Nazir, P. Xu, and J. Seo, "Rock-and-walk manipulation: Object locomotion by passive rolling dynamics and periodic active control," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2354–2369, 2022.

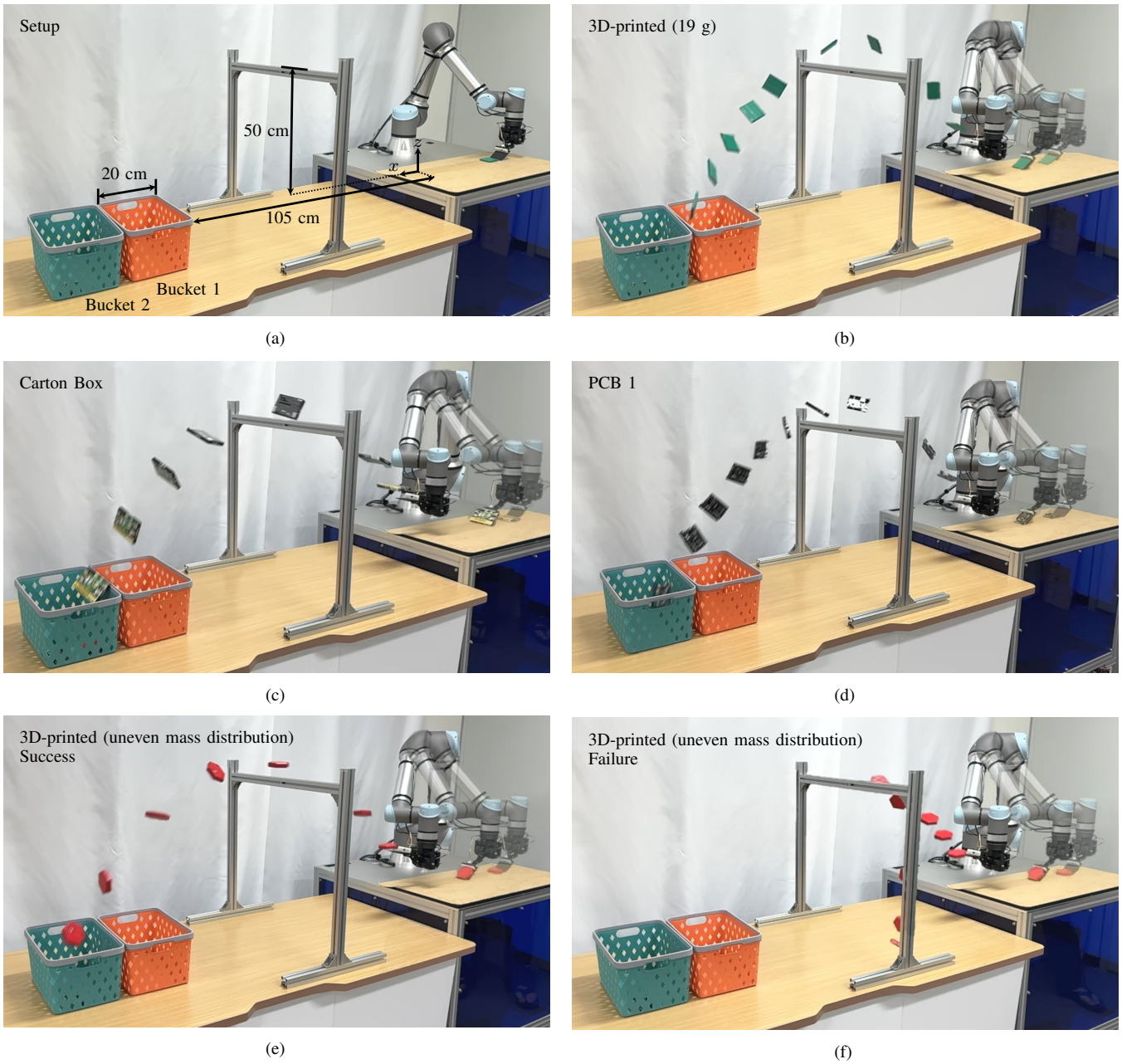


Fig. 9. Throw-in-bucket tasks. (a) Experimental setup. (b-f) Trials with different objects.

[8] T. McGeer, "Passive dynamic walking," *Int. J. Robot. Res.*, vol. 9, no. 2, pp. 62–82, 1990.

[9] T. Senoo, A. Namiki, and M. Ishikawa, "High-speed throwing motion based on kinetic chain approach," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 3206–3211.

[10] W. Mori, J. Ueda, and T. Ogasawara, "1-dof dynamic pitching robot that independently controls velocity, angular velocity, and direction of a ball: Contact models and motion planning," in *2009 IEEE Int. Conf. Robot. Automat. (ICRA)*, 2009, pp. 1655–1661.

[11] N. Govindan, B. Ramachandran, P. H. V. Sai, and K. M. Krishna, "A novel hybrid gripper capable of grasping and throwing manipulation," *IEEE/ASME Trans. Mechat.*, vol. 28, no. 6, pp. 3317–3328, 2023.

[12] D. Bianchi, G. Campinoti, C. Comitini, C. Laschi, A. Rizzo, A. M. Sabatini, and E. Falotico, "Softsling: A soft robotic arm control strategy to throw objects with circular run-ups," *IEEE Robotics and Automation Letters*, vol. 9, no. 10, pp. 8250–8257, 2024.

[13] L. Werner, F. Nan, P. Eyschen, F. A. Spinelli, H. Yang, and M. Hutter, "Dynamic throwing with robotic material handling machines," in *2024 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2024, pp. 98–104.

[14] Y. Liu and A. Billard, "Tube acceleration: Robust dexterous throwing against release uncertainty," *IEEE Transactions on Robotics*, vol. 40, pp. 2831–2849, 2024.

[15] A. Takahashi, M. Sato, and A. Namiki, "Dynamic compensation in throwing motion with high-speed robot hand-arm," in *2021 IEEE Int. Conf. Robot. Automat (ICRA)*, 2021, pp. 6287–6292.

[16] Y. Hou, Z. Fang, and J. Li, "Self-supervised multi-modal learning for collaborative robotic grasp-throw," *IEEE Robotics and Automation Letters*, vol. 9, no. 5, pp. 4250–4256, 2024.

[17] M. Monastirsky, O. Azulay, and A. Sintov, "Learning to throw with a handful of samples using decision transformers," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 576–583, 2023.

[18] H. Cha, I. Lee, and J. Seo, "High-speed scooping through dynamic manipulation: Model and practice," *IEEE Robotics and Automation Letters*, vol. 10, no. 2, pp. 1377–1384, 2025.