Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2026, **5**, 2120

Vision-guided adaptive scooping for powder weighing in autonomous chemistry laboratories

Nikola Radulov,[a] Thomas Little,[a] Andrew I. Cooper [b] and Gabriella Pizzuto *[ab]

Autonomous, high-accuracy powder dispensing of heterogeneous solid materials remains an open challenge in automated chemistry laboratories. Existing systems often fail with diverse powder morphologies because they neglect the critical initial material acquisition step (*i.e.*, scooping). We present an end-to-end, vision-guided powder dispensing system that integrates an adaptive scooping mechanism with a deep reinforcement learning-based policy for dispensing. Our system utilises a parametrised scooping motion and assesses visually the acquired material volume after each scooping attempt. This visual feedback drives an iterative correction loop, allowing the robot to adjust its motion parameters to reject failures and obtain a suitable quantity for subsequent dispensing. We evaluated our system in real laboratory conditions using a set of 7 powders with varying physical properties. Our experiments demonstrate that the fully adaptive system outperformed fixed scooping baselines, achieving the lowest average absolute weighing error of 1.93 mg ± 2.04 mg across all materials.

Received 18th December 2025
Accepted 21st April 2026

DOI: 10.1039/d5dd00569h

rsc.li/digitaldiscovery

1 Introduction

Accelerating chemical and material discovery is critical to future societal and industrial impact.[1] Stemming from an increasing demand of global challenges, AI-driven robotics is essential to accelerate scientific discovery, improve data quality, and simultaneously safeguard human scientists' health in academic and industry R&D labs.[2] Autonomous robotic systems have shown success in executing different experiments;[3] however, these lack the generalisability and robustness necessary especially in the realm of early-stage materials discovery where sample properties are heterogeneous and unpredictable. This underscores the critical need for developing robust robotic solutions for foundational, low-level tasks that can be employed across multiple domains.

Solid dispensing is a core task across a variety of different chemical domains[3] including solid-state materials chemistry,[4] photocatalysis,[5] process chemistry,[6] and organic synthesis.[7] Several commercial platforms are available using mechanisms such as screw feeders, vibrators, or revolving plates to induce powder flow.[8,9] However, these systems often require time-consuming, material-specific calibration and inherently struggle with abrasive or sticky powders.[8,10] For these instruments to function reliably, large-grain solids frequently require pre-treatment, such as grinding, to prevent particles from blocking the dispensing heads. For instance, Jiang *et al.*[10] found dispensing failures to commonly occur due to either blockages in the dispensing pathways (*e.g.*, the cartridges in Chemspeed systems or the dosing heads in Quantos devices) or mechanical damage to internal moving parts caused by highly abrasive solids. Furthermore, these hardware limitations significantly impact operational costs:[8] while some platforms utilise relatively inexpensive consumables like glass capillaries, others rely on dosing heads with a pre-defined maximum number of usage cycles. As these consumable heads are often recommended to be restricted to a single specific solid, the running costs of the platform might increase.[8] To overcome these issues, recent works have focused on methodological advancements to handle complex powder dynamics. For example, Fermier *et al.*[11] developed a high-precision, open-loop powder dispensing station that relies on meticulous offline optimisation of material-specific physical parameters like vacuum flow rate. Jiang *et al.*[10] proposed a dual-arm robotic system that uses a spatula and an analytical balance to iteratively dispense and correct the weighed amount. Kadokawa *et al.*[12] trained a sim-to-real deep reinforcement learning (DRL) policy that modulates the parameters of a predefined shake and tilt motion for a robot arm, while Radulov *et al.*[13] extended this with a framework that uses material properties (*i.e.*, flowability) to create higher fidelity simulations for powder dispensing robotic policies. Despite these advances, autonomous, high-accuracy dispensing of solid materials remains an open challenge due to the wide range of material properties.

While the aforementioned works focus primarily on refining the final dispensing stage, they often overlook the initial acquisition of the material. Previous work that does address the material acquisition (or scooping) phase has largely been focused within the food industry.[14–16] For instance, Takahashi *et al.*[17] rely on visual feedback to select a grasp point for a target mass, while Grannen *et al.*[18] use a bimanual setup with visual feedback to prevent fragile item (food) breakage, rather than to control quantity. Tai *et al.*[19] use active perception to understand material properties, which aids in generalising to previously unseen data

[a]*School of Computer Science & Informatics, University of Liverpool, UK. E-mail: gabriella.pizzuto@liverpool.ac.uk*
[b]*Department of Chemistry, University of Liverpool, UK*

and consequently enhances the success rate of achieving a minimum scooping amount. Given that the food industry typically handles much larger quantities (in grams) than scientific labs (in milligrams), it mitigates the concern of small clump inclusion, which is a major source of error when weighing small quantities for materials science. For autonomous powder scooping in chemistry labs, Takahashi *et al.*[20] introduced a soft conical universal-hand, which relies on a physically adaptive end-effector for scooping powders, where its mechanical design allows it to conform to different container geometries without advanced sensing. Building on this hardware, Wang *et al.*[21] introduced a physics-based simulation framework that enables scooping in a wide range of container shapes. By exploiting this high-fidelity simulator and jointly optimising the tool's motion parameters and its rolling angle, their method produces strongly effective scooping trajectories for diverse geometries that transfer to the real world with minimal sim-to-real gap. Critically, however, none of these methods effectively address the coupled, sequential dependency between the scooping and dispensing stages. Since the same scooping action can yield variable outcomes depending on material properties, the amount scooped introduces significant perturbations into the dispensing system, directly affecting the final result. Previous works lack the integrated, multi-stage pipeline necessary to handle this variability.

In this work, we address this critical initial scooping phase by proposing an adaptive scooping mechanism that leverages visual feedback for active quantity control and integrate it into an end-to-end powder dispensing system. We first define a parametrised scooping motion and systematically investigate its effect on the resulting powder quantity. To enhance robustness, we integrate visual feedback for detecting failed scooping attempts, and evaluate its performance in typical laboratory conditions (*e.g.*, with different lighting). This capability enables the intelligent coupling of our adaptive scooping module with a DRL-based powder dispensing strategy, based on our previous approach.[13] Finally, we evaluate the performance of the complete system for autonomously weighing heterogeneous materials, demonstrating how the adaptive scooping method outperforms a baseline method with predefined, fixed scooping motion.

In summary the contributions of this work are:

1. A parametrised scooping motion that allows for fine-grained control over the scooped powder quantity;

2. A vision-based feedback system for the robust detection of failed scooping instances and accurate estimation of the scooped powder quantities;

3. A demonstration and empirical evaluation of the adaptive end-to-end powder dispensing system across the tested range of powder materials.

2 Methodology

This work introduces an end-to-end method coupling the material scooping state with the dispensing stage to improve autonomous powder weighing. We first define scooping as a parametrised motion primitive (Section 2.1) and implement visual feedback (Section 2.2) for identifying scooping failures. Finally, we detail the complete system (Section 2.3) where we use an iterative correction loop to condition the quantity prior to the DRL-based dispensing method.[13]

2.1 Parametrised scooping motion

The parametrisation of the scooping motion is motivated by human heuristics observed in manual scooping tasks. Humans intuitively adjust their scooping motions based on the required volume and material consistency; for instance, they employ shallow, short movements when only a small quantity is needed. Similarly, the robotic scooping action is a parametrised parabolic trajectory originating from a user-defined point A, as depicted in Fig. 2. The trajectory is defined by its maximum depth D, horizontal length L, and a constant tool pitch angle θ. The tool pitch angle is critical for handling heterogeneous materials; for example, a steeper angle can help reject clumps during manipulation of cohesive samples. Concurrently, D and L directly control the scooped volume. To execute the trajectory, it is discretised into N waypoints for the tool centre point (TCP). Each point is defined in a 3D Cartesian space and can be sampled by using eqn (1), where $t \in [0, 1]$.

$$\begin{aligned} x(t) &= Ax + tL \\ y(t) &= Ay \\ z(t) &= Az - 4D \cdot t(1-t) \end{aligned} \tag{1}$$

The robot executes the trajectory *via* these waypoints using a Cartesian motion planner.[22] The impact of each parameter on the scooped quantity is detailed in Section 3.2.

2.2 Visual feedback for scooping failures

The variability in the physical properties of materials makes a fixed set of scooping parameters infeasible. A trajectory optimised for highly flowable materials, may scoop large clumps from cohesive materials, leading to quantity errors before dispensing begins. To address this, we implement a visual feedback system to evaluate the acquired material within the spatula. We formulate the scooping material detection task as a binary classification problem, classifying each attempt as successful or unsuccessful. The system estimates scooped volume by quantifying the material's pixel area, segmented using colour similarity to the source powder. The volume estimation process is given in Algorithm 1. Image data (F_{BGR}) is sampled every $\Delta t_{\mathrm{update}}$ seconds (using the RetrieveFrame() function) and transformed into the CIELAB colour space (F_{LAB}) to facilitate perceptually uniform colour comparisons (using the ConvertToLab() function). As the camera, tool, and source container have fixed relative positions after a scoop motion is executed, predefined region of interests (ROIs) are extracted (GetRoI()) for the tool (R_{S}) and the source container (R_{C}). The average colour of the material C_{ref} is computed from R_{C}. Subsequently, a binary mask (M_{binary}) is generated for the tool's ROI (R_{S}), which is achieved by thresholding the Euclidean distance (d) of each pixel's (p) colour and the reference colour (C_{ref}). The largest interconnected area within the mask (A_{powder}) is identified as the scooped material using the Block Based labelling with Decision Trees (BBDT) algorithm,[23] as

implemented in the OpenCV[24] library. A fill percentage (P_{fill}) is then computed as the ratio of the area of the powder (A_{powder}) to the total area of the tool's ROI (A_{spoon}). The latter is calculated by multiplying the width ($R_S \cdot w$) and the height ($R_S \cdot h$) of the tool's region of interest. To ensure a robust measurement, multiple samples are collected in a history buffer (H_{fill}) and the mean fill percentage is returned as the final volume estimate of the retrieved powder. Finally, the timestamp of the most recent sampling event (t_{update}) is set to the current time (t_{now}).

Algorithm 1 Vision-Guided Adaptive Scooping Method

Require: τ_{colour}: Colour distance threshold (in LAB space); Δt_{update}: Interval for updating scoop status; N: number of samples

$H_{fill} \leftarrow$ NewBuffer($maxlen = N$)
$t_{update} \leftarrow t_{now}$
while true **do**
  if $(t_{now} - t_{update}) \geq \Delta t_{update}$ **then**
    $F_{BGR} \leftarrow$ RetrieveFrame()
    $F_{LAB} \leftarrow$ ConvertToLab(F_{BGR})
    $R_C \leftarrow$ GetRoi($F_{LAB}, CONTAINER$)
    $R_S \leftarrow$ GetRoi($F_{LAB}, SPOON$)
    $C_{ref} \leftarrow \frac{\sum_{i=1}^{|R_C|} R_{Ci}}{|R_C|}$
    for p at (r,c) **in** R_S **do**
      $d \leftarrow \sqrt{(p - C_{ref})^2}$
      if $d < \tau_{colour}$ **then**
        $M_{binary}[r,c] \leftarrow 1$
      else
        $M_{binary}[r,c] \leftarrow 0$
      end if
    end for
    $A_{powder} \leftarrow$ BBDT(M_{binary})
    $A_{spoon} \leftarrow R_S.w \times R_S.h$
    $P_{fill} \leftarrow \frac{A_{powder}}{A_{spoon}} \times 100$
    $H_{fill} \leftarrow H_{fill} \cup P_{fill}$
    if $|H_{fill}| = N$ **then**
      return $\frac{\sum_{i=1}^{|H_{fill}|} H_{fill_i}}{|H_{fill}|}$
    end if
    $t_{update} \leftarrow t_{now}$
  end if
end while

2.3 End-to-end powder weighing system

To facilitate end-to-end automation, the system is designed to manage the complete experimental pipeline, including the loading and unloading of tools and source containers. A user first registers the coordinates of key locations for recipients and tools. Using these coordinates, the system calculates safe, collision-free paths for all transfer motions. These paths are constrained within a predefined "safezone", a designated area in the robot's workspace that needs to be kept clear of potential obstructions.

The autonomous powder dispensing process (Algorithm 2) begins with an iterative correction loop designed to ensure a successful initial scooping motion. A scooping motion is generated and executed (Scoop(θ_{current}, D_{current})) with an initial set of parameters, tool pitch θ_{init} and depth D_{init}, according to eqn (1). After the robot scoops the material, our vision system measures the volume within the spoon (MeasureVolume()) by implementing the method from Section 2.2 to estimate the acquired volume (P_{fill}) and comparing it against two predefined thresholds: a minimum threshold, t_{empty}, to detect insufficient acquisition, and a maximum threshold, t_{clumps}, to identify over-scooping, which is often caused by large clumps in cohesive materials. If the estimated volume is outside this acceptable range, the scooping parameters are adjusted, and the action is repeated. For an under-scoop ($P_{\text{fill}} < t_{\text{empty}}$), the trajectory is made more aggressive by first decreasing the pitch angle θ or, if the new pitch angle θ_{new} is below the lower limit, by increasing the depth D. For an over-scoop ($P_{\text{fill}} > t_{\text{clumps}}$), the trajectory is made less aggressive by first increasing θ, or, if the new pitch angle θ_{new} is above the upper limit, D is decreased. This iterative process is motivated by our experimental findings (Section 3.2) which demonstrated that θ and D provide the most direct control over the acquired material volume.

Algorithm 2 End-to-end Powder Weighing System

Require: t_{empty}: Threshold volume for detecting an empty scoop; t_{clumps}: Threshold volume used for detecting clumps; θ_{min}: Minimum allowed pitch; θ_{max}: Maximum allowed pitch;θ_{init}: Initial tool pitch D_{min}: Minimum allowed depth; D_{max}: Maximum allowed depth; D_{init}: Initial scoop depth; $\Delta\theta$: pitch change; ΔD: Depth change; π: Dispensing policy; w_{goal}: Amount to be dispensed; $\theta_{dispensing}$: The starting tool pitch for the dispensing phase; $E_{threshold}$: Maximum accepted error for a successful dispensing; $\mathscr{E}_{length}$: Dispensing episode length

$D_{current} \leftarrow D_{init}$
$\theta_{current} \leftarrow \theta_{init}$
while $D_{current} \in [D_{min}, D_{max}]$ **do**
  Scoop($\theta_{current}, D_{current}$)
  $P_{fill} \leftarrow$ MeasureVolume()
  if $P_{fill} < t_{empty}$ **then**
    $\theta_{new} \leftarrow \theta_{current} - \Delta\theta$
    if $\theta_{new} < \theta_{min}$ **then**
      $D_{current} \leftarrow D_{current} + \Delta D$
    else
      $\theta_{current} \leftarrow \theta_{new}$
    end if
  else if $P_{fill} > t_{clumps}$ **then**
    $\theta_{new} \leftarrow \theta_{current} + \Delta\theta$
    if $\theta_{new} > \theta_{max}$ **then**
      $D_{current} \leftarrow D_{current} - \Delta D$
    else
      $\theta_{current} \leftarrow \theta_{new}$
    end if
  else
    break
  end if
end while
$w_{current} \leftarrow 0$
$\theta \leftarrow \theta_{dispensing}$
$i \leftarrow 0$
while $|w_{current} - w_{goal}| > E_{threshold}$ **do**
  $a_{incline}, a_{shake} \leftarrow \pi(w_{current}, \theta, w_{goal})$
  $w_{current} \leftarrow$ Robot.execute($a_{incline}, a_{shake}$)
  $i \leftarrow i + 1$
  if $i > \mathscr{E}_{length}$ **then**
    break
  end if
end while

Once an acceptable quantity is contained in the spatula, the system transitions to the dispensing phase. We adapt the

flowability-informed powder weighing (FLIP) DRL-based policy,[13] specifically utilising the random version of FLIP, to ensure a controlled experiment. This approach avoids the complex, entangled interaction effects that would arise from combining our method with a separate curriculum learning policy, allowing for a clear evaluation of our scooping mechanism's specific impact. However, we modify the previous reward function to discourage overly aggressive actions and prevent the policy's tendency to overshoot the target mass. As shown in eqn (2), we introduce an exponential penalty term (ρ):

$$R_{\text{new}} = R\rho^{N} \tag{2}$$

where R is the original reward function from the previous work,[13] N is the current step, and the hyperparameter ρ is tuned and set to 1.1. All other hyperparameters remain as specified in the previous work.[13] The policy is transferred to the real system *via* zero-shot transfer, with output actions from the policy being directly fed to the robot (Robot.execute()). The speed and shake amplitude of the real system were calibrated to match the simulator. Before dispensing starts, the scale is reset ($w_{\text{current}} \leftarrow 0$) and the robot moves to the initial pouring position ($\theta_{\text{dispensing}}$). When dispensing, the system is allowed a total of $\varepsilon_{\text{length}}$ steps, each comprising an incline (a_{incline}) and a shake action (a_{shake}), similar to the approach by Kadokawa *et al.*[12] The dispensed weight (w_{current}) is updated at the end of each step, once both actions are executed. Additionally, we implement an early stopping mechanism, terminating the dispensing process when the absolute error $E = |w_{\text{dispensed}} - w_{\text{target}}|$ falls below $E_{\text{threshold}} = 1$ mg.

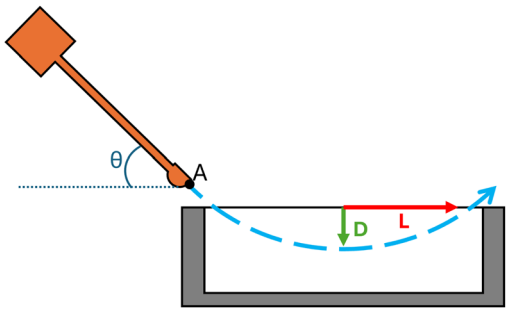

Fig. 2 The scooping action is defined as a parametrised parabolic motion where L is the length of the parabola, D is the parabola's depth and θ is the tool pitch angle.

3 Experiments

Our system was evaluated in real-world laboratory conditions. We focus on material control *via* parametrisation, robustness of visual feedback for detecting failed scoops, and comparing adaptive *versus* fixed scooping performance.

3.1 Experimental setup

Our setup (Fig. 1) utilises a Franka Research (FR) 3 (ref. 25) robot arm with a Robotiq 85F gripper mounted parallel to the working surface to facilitate tool manoeuvring, and an external Intel RealSense D405 RGB-D camera. A Sartorius Entris II precision scale weighs the samples, with the data fed directly to the dispensing policy. Both the spatula and the source container are initially set in custom holders for simplified pick-and-place operations. The robot's workflow for autonomously weighing samples involves the following sequential operations: retrieving the source container and positioning it onto the scale's designated holder (1), grasping the tool and traversing to the parametrised start point (2), iteratively executing scooping actions until the visual feedback system confirms successful material acquisition (3), transferring the tool above the 20 mL target vial (4), and running the DRL-based dispensing policy until the target weight is reached (5). We selected a set of 7 powders (silicon dioxide (SiO_2), sugar, sodium chloride (NaCl), semolina, sodium bicarbonate ($NaHCO_3$), pectin, and wheat flour) to rigorously test the handling capabilities of the automated system. These materials were chosen to represent distinct flow characteristics, ranging from free-flowing materials (*e.g.*, NaCl, sugar) to highly cohesive, fine powders prone to agglomeration and static bridging (*e.g.*, pectin, flour). We characterise the powder flow by determining the static flowability *via* the angle of repose (AoR), as presented in Table 1. By encompassing a broad spectrum of particle sizes and bulk densities, these powders serve as challenging, non-hazardous surrogates for the diverse materials typically encountered

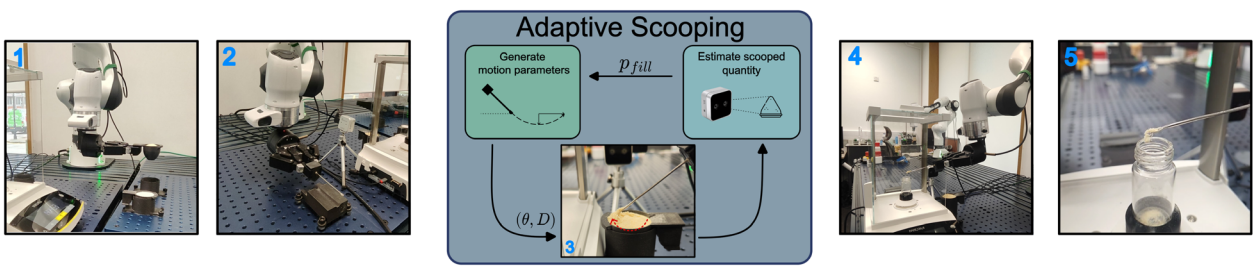

Fig. 1 An overview of the end-to-end system. Following user initialisation (Cartesian coordinates of containers and tool, materials, target weight), the system executes the following procedure: (1) the robot loads the source container for the scooping action; (2) it grasps the tool; (3) the system initiates an iterative scooping loop: a trajectory is executed, and the acquired volume is visually evaluated. If the volume is outside acceptable thresholds, the trajectory is reformulated and the process repeats until a suitable quantity is acquired. (4) The robot positions the tool loaded with powder above the glassware on the scale; (5) the material is dispensed using the DRL-based policy.

Table 1 Powder flowabilities expressed as the AoR, obtained using averaged manual measurements performed according to the ISO 8398:1989 standard

Material	AoR (°)
SiO_2	27.69 ± 2.54
Sugar	30.93 ± 0.87
NaCl	35.8 ± 1.71
Semolina	40.91 ± 1.85
$NaHCO_3$	43.14 ± 1.0
Pectin	46.35 ± 2.45
Flour	51.21 ± 1.25

across a wide range of materials chemistry workflows. This selection of materials allows for characterisation of the system's performance in a bench-safe environment, without the safety constraints associated with hazardous or reactive chemical handling.

3.2 On quantity control with parametrised scooping motion

To quantify the influence of each motion parameter (Section 2.1) on the acquired material quantity, an empirical grid search was conducted. The search space was defined for the tool pitch, $\theta \in [20°, 60°]$ (10° step); depth, $D \in [0.013, 0.017]$ m (0.001 m step); and length, $L \in [0.021, 0.029]$ m (0.002 m step). Each parameter was varied in isolation while the others were held constant at their central values ($\theta = 40°$, $D = 0.015$ m, $L = 0.025$ m). The results were averaged over 5 trials for each parameter combination and illustrate that D and θ were the primary influencing factors for the acquired quantity. The results for θ and D are shown in Fig. 3a and b respectively, where the highlighted area represents the target scooped quantity required for the ideal initial conditions for the DRL-based dispensing policy.[13] This target range is based on the conditions used to train the policy in simulation, which was exclusively exposed to initial states where the spoon was loaded with these specific quantities. Varying the scooping motion length L had a negligible effect (Fig. 3c), with the scooped quantities consistently remaining within a 10 mg range of each other. The only notable exception is sodium bicarbonate, which exhibits a descending trend. For this material, increasing the trajectory length resulted in a 13.8 mg decrease in the acquired quantity.

3.3 On scooping robustness with visual feedback

To validate the performance of our vision-based estimation for detecting failed scoops, we conducted an experiment correlating the estimated fill percentage (P_{fill}) with the measured weight of the acquired sample. For each powder, we performed a series of scooping actions by systematically varying the tool pitch angle from 20° to 60° in 5° increments. Five trials were executed for each angle setting per material. In each trial, the P_{fill}-weight pair was recorded. Fig. 4 shows the results obtained for our materials. The optimal value for the empty-scoop threshold, $t_{empty} = 16$, is selected based on the empirical relationship between the estimated fill percentage (P_{fill}) and the measured weight. The goal is to identify the minimum P_{fill} that

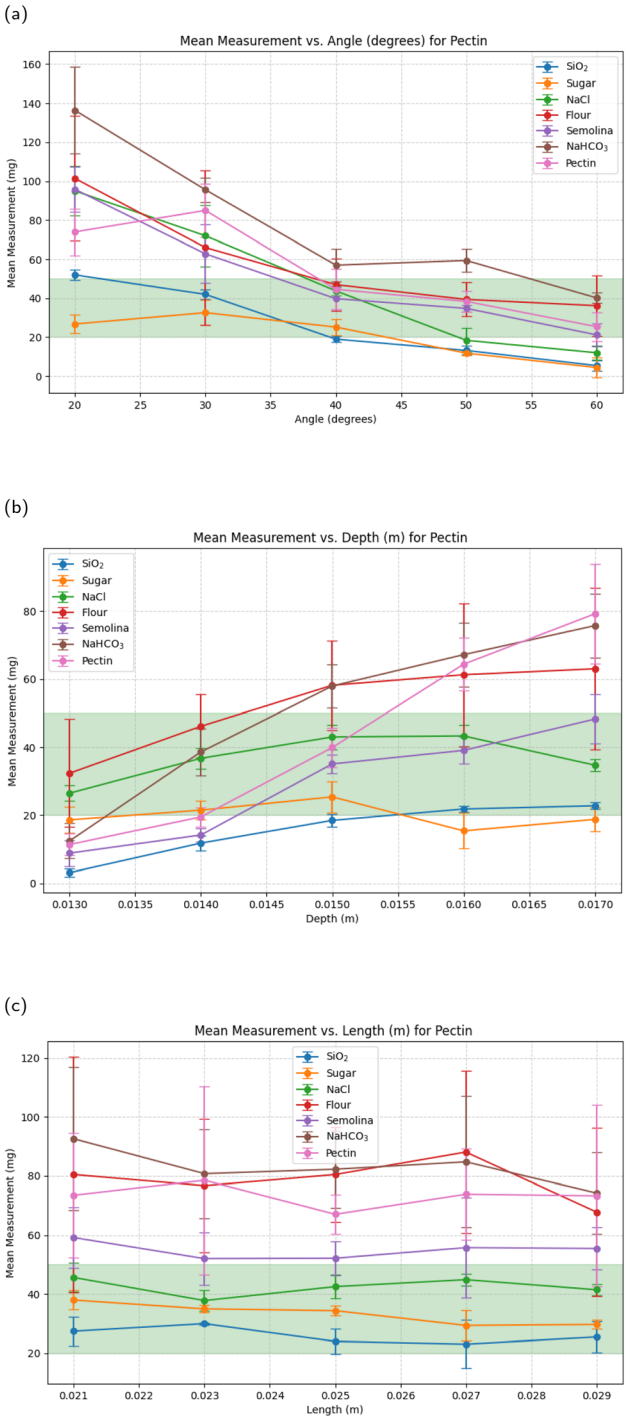

Fig. 3 Evolution of scooped quantity per material, (a) relative to tool pitch θ and (b) scooping trajectory depth D, averaged over 5 trials. Figure (c) shows scooped quantity per material relative to motion length L averaged over 5 trials.

ensures a scooped quantity of at least 20 mg for the largest possible subset of materials. This selection is further constrained such that the chosen threshold must be within the range of P_{fill} values observed for every material. This constraint is critical to prevent certain powders from being systematically misclassified as empty scoops.

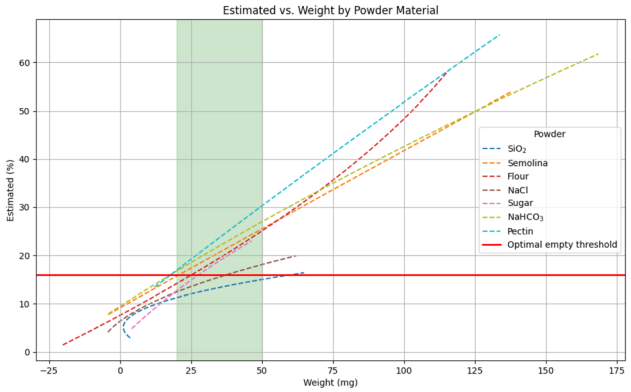

Fig. 4 The relational trends between the estimated fill percentage P_{fill} and the real sampled weight. The red line represents the chosen t_{empty} value. The highlighted areas represent the optimal weight conditions for the dispensing policy.

While this fixed threshold is highly effective for our current setup, it is important to note that t_{empty} requires manual retuning under two conditions. First, any hardware modifications that alter the camera's perspective or the dimensions of the scooping tool will change the designated RoI, requiring recalibration. Second, because P_{fill} acts as a visual proxy for volume, these thresholds are sensitive to extreme variations in material bulk density. Introducing a material with a significantly lower density than the optimised range could yield a high P_{fill} value while failing to meet the minimum required mass, necessitating a higher t_{empty} threshold to prevent false-positive detections.

We tested the system under varying illuminance intensities (40 lux, 130 lux, and 700 lux), defining a scoop as unsuccessful only if the extracted amount falls below the minimum threshold of the optimal dispensing conditions (<20 mg).

Fig. 5 presents the confusion matrices evaluated on the three different illuminance levels. For each illuminance, the evaluation data was comprised of:

1. Positive class: 105 points, consisting of 15 successful scoops for each powder;

2. Negative class: 60 points, comprised of 15 attempts with a completely empty source container and 15 attempts each on containers with the following materials: flour, sand or pectin. Since our volume approximation method relies on colour matching these specific powders were chosen as their colours are representative of the full range found within our dataset.

The different illuminance conditions were created by controlling the number of active lighting fixtures in the laboratory. Overall our system predicts successful scoops with 89.21% accuracy and unsuccessful scoops with 96% accuracy across all conditions.

Failure to detect empty scoops is a critical issue that leads to a reduction in efficiency. Furthermore, persistent acquisition failures under high-yield motion parameters can signal a depleted source container, triggering an alert for operator intervention.

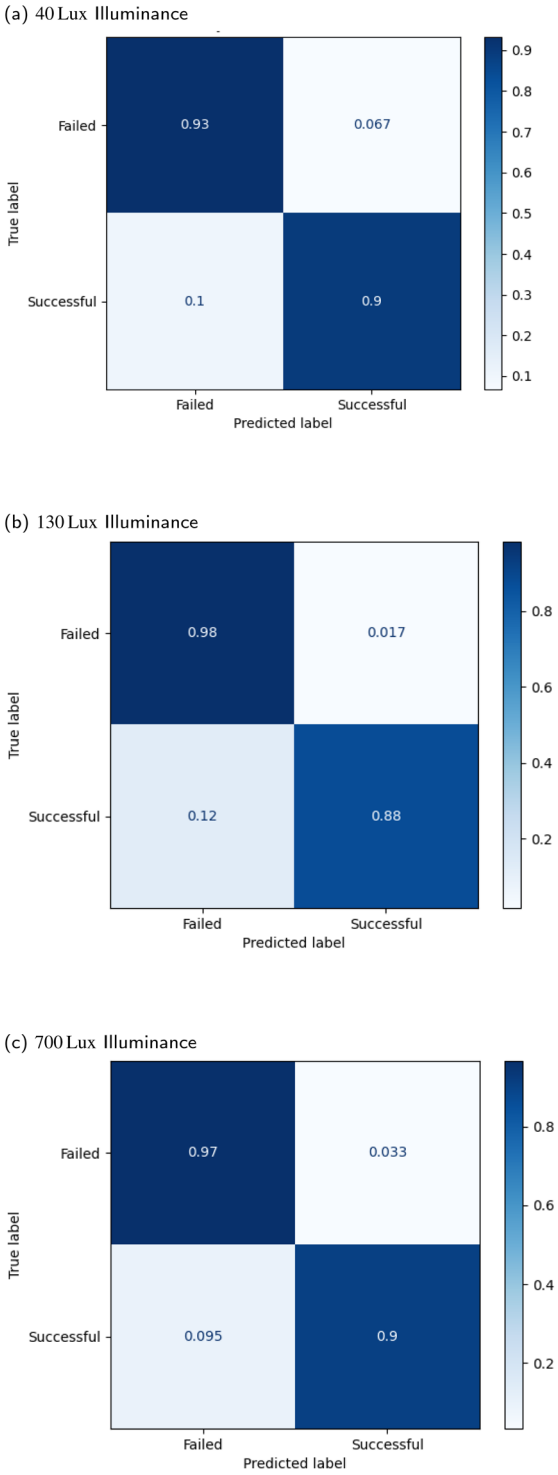

Fig. 5 Confusion matrices for detecting empty and successful scoops under varying lighting conditions.

3.4 Overall system evaluation

We conducted an end-to-end system evaluation of our adaptive scooping method against fixed-motion baselines. The baselines pair an empirically optimised scooping trajectory ($D = 0.015$ m, $\theta = 40°$) with either a low- or high-velocity dispensing profile,

Table 2 Real-world performance comparison for the powder weighing task with 15 mg target weight. Results represent the mean absolute error and standard deviation over 10 runs

Method	SiO_2	Sugar	NaCl	Semolina	$NaHCO_3$	Pectin	Flour
Baseline (high velocity)	**0.73 ± 0.76**	1.3 ± 1.36	3.85 ± 2.90	6.0 ± 5.28	7.4 ± 5.48	6.56 ± 7.83	18.06 ± 15.05
Baseline (low velocity)	0.97 ± 0.96	3.19 ± 2.05	7.30 ± 5.54	6.64 ± 4.57	23.86 ± 15.31	5.63 ± 6.42	11.68 ± 5.44
Adaptive scooping (high velocity)	9.71 ± 2.22	1.84 ± 1.23	11.80 ± 3.71	**1.52 ± 1.52**	2.21 ± 2.5	1.36 ± 0.85	3.85 ± 2.06
Adaptive scooping (dual velocity)	1.17 ± 1.08	**0.94 ± 0.82**	**2.65 ± 2.51**	2.50 ± 2.82	**1.98 ± 2.20**	**0.97 ± 0.96**	**3.3 ± 3.1**

Table 3 Real-world performance comparison for the powder weighing task with 20 mg target weight. Results represent the mean absolute error and standard deviation over 10 runs

Method	SiO_2	Sugar	NaCl	Semolina	$NaHCO_3$	Pectin	Flour
Baseline (high velocity)	2.4 ± 2.0	1.23 ± 0.65	5.28 ± 2.93	2.71 ± 4.14	6.29 ± 8.34	4.94 ± 2.71	27.48 ± 19.40
Baseline (low velocity)	**1.5 ± 1.57**	1.25 ± 0.96	8.54 ± 5.30	5.00 ± 5.35	23.83 ± 13.35	7.59 ± 6.08	14.43 ± 5.59
Adaptive scooping (high velocity)	4.31 ± 1.46	**0.63 ± 0.35**	8.80 ± 3.94	2.48 ± 2.17	3.97 ± 4.14	3.03 ± 3.15	3.55 ± 3.00
Adaptive scooping (dual velocity)	1.88 ± 0.97	1.4 ± 0.85	**2.55 ± 1.57**	**2.21 ± 1.60**	**1.64 ± 1.11**	**2.33 ± 1.58**	**3.2 ± 2.42**

tuned for flowable and cohesive powders, respectively. We tested a fixed high velocity adaptive scooping configuration and a fully adaptive version that selects the dispensing velocity based on the final scooping pitch angle (40° threshold). Vision system thresholds were set to $t_{\text{empty}} = 16°$ and $t_{\text{clumps}} = 28°$. As shown in Tables 2 and 3, the adaptive systems outperform the baselines on cohesive powders (*e.g.*, flour, pectin). This performance improvement over fixed-motion baselines stems from the system's ability to regulate the acquired mass prior to dispensing. When handling cohesive materials, the standard fixed scooping trajectory frequently overloads the spoon or acquires large clumps. These excessive initial quantities fall significantly outside the optimal operational bounds of the DRL-based dispensing policy, inevitably leading to target overshoots. In contrast, the vision-guided adaptive scooping strategy successfully filters out these clumps and prevents overloading during the acquisition phase. The dual velocity adaptive scooping system achieves the lowest average powder weighing error across all trials (1.93 mg), followed by high velocity adaptive scooping (4.61 mg), the fixed scooping high velocity system (6.11 mg) and finally, the fixed scooping low velocity system (8.19 mg). However, the adaptive system with a fixed high velocity performed poorly on highly flowable materials (sand and salt). Excess material resulting from the low pitch angle, combined with high-velocity dispensing, caused the policy to overshoot the target. The dual velocity system successfully mitigated this by selecting a lower speed, demonstrating the benefit of a fully adaptive pipeline.

4 Conclusion

In this work, we presented an end-to-end, vision-guided powder dispensing system that successfully integrates an adaptive scooping mechanism with a DRL-based dispensing policy. We introduced a parametrised scooping motion and used visual feedback to robustly identify failed attempts. Our evaluation showed that this adaptive system outperforms fixed-motion baselines and demonstrated the critical benefit of dynamically coupling material acquisition and dispensing.

One limitation of the current scooping-based approach is its reliance on a sufficiently preloaded volume of powder in the source container to ensure consistent retrieval. In our standard experimental setup, an initial material quantity of between 13.81 g and 34.4 g was used. While adjusting the container size and scooping trajectory length can reduce this operational minimum, it remains a constraint within practical self-driving labs when handling scarce, expensive, or custom-synthesised materials. Consequently, while the current system is highly robust for bulk reagent handling, future work will focus on adapting the robotic trajectory and end-effector design to accommodate sub-gram quantities, such as precision scooping directly from small vials. Furthermore, while our current visual feedback approach excels at detecting empty scoops, the system's current reliance on manually defined thresholds (*e.g.*, t_{empty}, t_{clumps}) based on a 2D area proxy (P_{fill}) limits its immediate portability. To resolve this we plan to explore how machine learning-based vision systems and direct mass feedback can be integrated to further improve the accuracy and robustness of material acquisition.

Author contributions

Authors are listed alphabetically by contribution. Conceptualisation: Gabriella Pizzuto, Nikola Radulov. Data curation: Thomas Little, Nikola Radulov. Formal analysis: Thomas Little, Nikola Radulov. Funding acquisition: Gabriella Pizzuto. Investigation: Thomas Little, Nikola Radulov. Methodology: Gabriella Pizzuto, Nikola Radulov. Project administration: Gabriella Pizzuto. Resources: Gabriella Pizzuto. Software: Thomas Little, Nikola Radulov. Supervision: Andrew I. Cooper, Gabriella Pizzuto. Validation: Thomas Little, Nikola Radulov. Visualisation: Nikola Radulov. Writing – original draft: Gabriella Pizzuto, Nikola Radulov. Writing – review & editing: Andrew I. Cooper, Gabriella Pizzuto, Nikola Radulov.

Conflicts of interest

There are no conflicts to declare.

Data availability

The code, hardware designs, and models required to replicate the experiments will be open-sourced and archived on Zenodo at: https://doi.org/10.5281/zenodo.19357593. The repository includes python scripts for controlling an FR3 robot, the attached Robotiq 85F Gripper as well as python interfaces for Sartorius Entris II and Fisherbrand FPRS22 precision scales. 3D models for custom made assets are also provided together with the trained models used in the presented experiments. We make available an unedited video demonstration of the end-to-end system: https://youtu.be/1GqivYt166c.

Supplementary information (SI) is available. See DOI: https://doi.org/10.1039/d5dd00569h.

Acknowledgements

This work was supported by the Leverhulme Trust through the Leverhulme Research Centre for Functional Materials Design, the Royal Academy of Engineering under the Research Fellowship Scheme, EPSRC through the AI for chemistry: AIchemy hub (EP/Y028775/1) and Henry Royce Institute through the Undergraduate Internship Scheme. A. I. C. thanks the Royal Society for a Research Professorship (RSRP\S2\232003).

References

1 A. I. Cooper, P. Courtney, K. Darvish, M. Eckhoff, H. Fakhruldeen, A. Gabrielli, A. Garg, S. Haddadin, K. Harada, J. Hein, M. Hübner, D. Knobbe, G. Pizzuto, F. Shkurti, R. Shrestha, K. Thurow, R. Vescovi, B. Vogel-Heuser, Á. Wolf, N. Yoshikawa, Y. Zeng, Z. Zhou and H. Zwirnmann, *Sci. Robot.*, 2025, **10**, eadv7932.

2 S. X. Leong, C. E. Griesbach, R. Zhang, K. Darvish, Y. Zhao, A. Mandal, Y. Zou, H. Hao, V. Bernales and A. Aspuru-Guzik, *Nat. Rev. Chem.*, 2025, **9**, 707–722.

3 G. Tom, S. P. Schmid, S. G. Baird, Y. Cao, K. Darvish, H. Hao, S. Lo, S. Pablo-García, E. M. Rajaonson, M. Skreta, N. Yoshikawa, S. Corapi, G. D. Akkoc, F. Strieth-Kalthoff, M. Seifrid and A. Aspuru-Guzik, *Chem. Rev.*, 2024, **124**, 9633–9732.

4 A. M. Lunt, H. Fakhruldeen, G. Pizzuto, L. Longley, A. White, N. Rankin, R. Clowes, B. Alston, L. Gigli, G. M. Day, A. I. Cooper and S. Y. Chong, *Chem. Sci.*, 2024, **15**, 2456–2463.

5 B. Burger, P. M. Maffettone, V. V. Gusev, C. M. Aitchison, Y. Bai, X. yan Wang, X. Li, B. M. Alston, B. Li, R. Clowes, N. Rankin, B. Harris, R. S. Sprick and A. I. Cooper, *Nature*, 2020, **583**, 237–241.

6 E. J. Brass, S. Veeramani, Z. Zhou, H. Fakhruldeen, J. S. Manzano, R. Clowes, I. Akpinar, M. R. Ward, J. W. Ward and A. I. Cooper, *Digital Discovery*, 2026, **5**, 1363–1371.

7 T. Dai, S. Vijayakrishnan, F. T. Szczypiński, J.-F. Ayme, E. Simaei, T. Fellowes, R. Clowes, L. Kotopanov, C. E. Shields, Z. Zhou, J. W. Ward and A. I. Cooper, *Nature*, 2024, **635**, 890–897.

8 M. N. Bahr, M. A. Morris, N. P. Tu and A. Nandkeolyar, *Org. Process Res. Dev.*, 2020, **24**, 2752–2761.

9 M. N. Bahr, D. B. Damon, S. D. Yates, A. S. Chin, J. D. Christopher, S. Cromer, N. Perrotto, J. Quiroz and V. Rosso, *Org. Process Res. Dev.*, 2018, **22**, 1500–1508.

10 Y. Jiang, H. Fakhruldeen, G. Pizzuto, L. Longley, A. He, T. Dai, R. Clowes, N. Rankin and A. I. Cooper, *Digital Discovery*, 2023, **2**, 1733–1744.

11 A. M. Fermier, J. Troisi, E. C. Heritage, M. A. Drexel, P. Gallea and K. A. Swinney, *Analyst*, 2003, **128**, 790–795.

12 Y. Kadokawa, M. Hamaya and K. Tanaka, *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 2932–2939.

13 N. Radulov, A. Wright, T. Little, A. I. Cooper and G. Pizzuto, *IEEE International Conference on Robotics and Automation*, 2026.

14 Z. Wang, H. Furuta, S. Hirai and S. Kawamura, *Front. Rob. AI*, 2021, **8**, DOI: 10.3389/frobt.2021.640805.

15 Y. Ohshima, Y. Kobayashi, T. Kaneko, A. Yamashita and H. Asama, *2013 IEEE Workshop on Robot Vision (WORV)*, 2013, pp. 82–87.

16 Y. Kuriyama, Y. Okino, Z. Wang and S. Hirai, *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, 2019, pp. 114–119.

17 K. Takahashi, W. Ko, A. Ummadisingu and S. Maeda, *2021 IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 2620–2626.

18 J. Grannen, Y. Wu, S. Belkhale and D. Sadigh, *Proc. Mach. Learn. Res.*, 2022, **205**, 1510–1519.

19 Y.-L. Tai, Y. C. Chiu, Y.-W. Chao and Y.-T. Chen, *Proceedings of The 7th Conference on Robot Learning*, 2023, pp. 849–865.

20 T. Takahashi, C. C. Beltran-Hernandez, Y. Kuroda, K. Tanaka, M. Hamaya and Y. Ushiku, *2025 IEEE Int. Conf. Robot. Autom. (ICRA)*, 2025.

21 Y. Wang, C. C. Beltran-Hernandez, T. Takahashi and M. Hamaya, Simulation-Driven Evolutionary Motion Parameterization for Contact-Rich Granular Scooping with a Soft Conical Robotic Hand, *arXiv*, 2026, preprint, arXiv:2604.05531, 10.48550/arXiv.2604.05531, https://arxiv.org/abs/2604.05531.

22 T. Schneider, franky: High-Level Control Library for Franka Robots, https://github.com/TimSchneider42/franky.

23 C. Grana, D. Borghesani and R. Cucchiara, *IEEE Trans. Image Process.*, 2010, **19**, 1596–1609.

24 G. Bradski, *Dr. Dobb's Journal of Software Tools*, 2000.

25 S. Haddadin, *IEEE Rob. Autom. Mag.*, 2024, **31**, 136–148.