Puttybot - A sensorized robot for autonomous putty plastering

Zhao Liu[1], Dayuan Chen[1], Mahmoud A. Eldosoky[2], Zefeng Ye[3], Xin Jiang[1], Yun-Hui Liu[3], and Shuzhi Sam Ge[4]

[1]Harbin Institute of Technology Shenzhen
[2]University of Electronic Science and Technology of China
[3]The Chinese University of Hong Kong Department of Mechanical and Automation Engineering
[4]National University of Singapore Department of Electrical and Computer Engineering

September 14, 2023

Abstract

Plastering is dominated manually, exhibiting low levels of automation and inconsistent finished quality. A comprehensive review of literature indicates that extant plastering robots demonstrate a subpar performance when tasked with rectifying defects in transition area. The limitations encompass a lack of capacity to independently evaluate the quality of work or perform remedial plastering procedures. To address this issue, this research describes the system design of the Puttybot and a paradigm of plastering to solve the stated problems. The Puttybot consists of a mobile chassis, lift platform, and a macro/micro manipulator. The force-controlled scraper parameters have been calibrated to dynamically modify their rigidity in response to the applied putty. This strategy utilizes Convolutional Neural Networks to identify plastering defects and executes the plastering operation with force feedback. This paradigm's effectiveness was validated during an autonomous plastering trial wherein a large-scale wall was processed without human involvement.

Received <day> <Month>, <year>; | Revised <day> <Month>, <year>; | Accepted <day> <Month>, <year>

DOI: xxx/xxxx

ARTICLE TYPE

Puttybot - A sensorized robot for autonomous putty plastering

Zhao Liu[1] | Dayuan Chen[1] | Mahmoud A. Eldosoky[2] | Zefeng Ye[3] | Xin Jiang[1] | Yunhui Liu[3] | Shuzhi Sam Ge[4,5]

[1]Department of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, China
[2]School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China
[3]Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Shatin, New Territories, China
[4]Department of Electrical and Computer Engineering, National University of Singapore, Singapore, Singapore
[5]Institute for Future, Qingdao University, Qingdao, China

Correspondence
Xin Jiang, HIT Campus of University Town of Shenzhen, Taoyuan Street, Nanshan District, Shenzhen, China.
Email: x.jiang@ieee.org

Abstract

Plastering is dominated manually, exhibiting low levels of automation and inconsistent finished quality. A comprehensive review of literature indicates that extant plastering robots demonstrate a subpar performance when tasked with rectifying defects in transition area. The limitations encompass a lack of capacity to independently evaluate the quality of work or perform remedial plastering procedures. To address this issue, this research describes the system design of the Puttybot and a paradigm of plastering to solve the stated problems. The Puttybot consists of a mobile chassis, lift platform, and a macro/micro manipulator. The force-controlled scraper parameters have been calibrated to dynamically modify their rigidity in response to the applied putty. This strategy utilizes Convolutional Neural Networks to identify plastering defects and executes the plastering operation with force feedback. This paradigm's effectiveness was validated during an autonomous plastering trial wherein a large-scale wall was processed without human involvement.

KEYWORDS:
Interior finishing, Plastering robot, Impedance control, Convolutional Neural Network

1 | INTRODUCTION

The level of automation within the construction industry remains relatively limited[1]. This sector, on a global scale, confronts significant challenges stemming from labor costs, while also contending with safety concerns and issues related to construction efficiency[2]. Consequently, the relatively low degree of automation, along with the associated high costs and scarcity of labor, have spurred the exploration of construction robots as a potential solution.

Current construction robots commonly exhibit semi-automated features, as they lack the capability to independently evaluate construction quality or determine appropriate corrective measures. Instead, engineers must manually prescribe the robots' actions based on on-site observations and contextual understanding. Notably, plastering operations remain predominantly reliant on manual interventions. Existing plastering robots heavily rely on precise initial positioning and necessitate continuous worker intervention to achieve accuracy and adequately cover expansive wall surfaces[3]. This reliance on worker proficiency introduces substantial repeatability errors during position alignment, manifesting as visible imperfections in the transitional areas between adjacent workstations, which subsequently require manual rectification. To address these challenges, this paper introduces a paradigm for robot plastering, enhancing the quality of plastering across multiple workstations while ensuring stability and resilience. The achievement of full autonomy holds the potential to mitigate the prevalent labor shortages in this domain.

1.1 | Contributions of the paper

This study proposes a plastering robot characterized by the employment of a macro/micro manipulator and a paradigm of the robot plastering. The main contributions of this study are as follows.

(1) A novel paradigm for the robot plastering is proposed. The validity of the robot's workflow and plastering strategy has been verified through autonomous plastering conducted across multiple workstations.

(2) A motorized scraper is proposed for putty plastering. The system has the capability to not only control the inclination angle of the scraper blade in relation to the wall but also adjust its stiffness, enabling precise and intricate plastering operations.

(3) A Convolutional Neural Network-based plastering quality recognition method is proposed. This functionality empowers the robot to autonomously assess the quality of each plastering task. Consequently, the implementation of plastering quality control is accomplished through the formulation of corrective measures when substandard quality is detected.

(4) Autonomous plastering is achieved for a large-scale wall. The overall consistency of the plastering quality is guaranteed through multiple movements of the mobile chassis at each moving station. During the entire process, no human intervention is needed.

In Section 3, the robot system is comprehensively outlined, encompassing its essential hardware components and software architecture. Notably, the incorporation of the macro/micro manipulator optimizes the working space, elevates the quality of plastering, and enhances overall robustness. In Section 4, an exposition is provided on the control method of the Puttybot, which utilize the generalized kinematic inverse solution and impedance control. Moreover, the application of the AMCL algorithm for localization and navigation functions is elucidated. In Section 5, a plastering paradigm with proposed robot is introduced. To ensure compliance with the thickness and flatness requirements of the applied putty, we first discussed the key parameters that most significantly affect plastering quality and designed a robotic plastering workflow including automatic position alignment, target wall calibration. To achieve a smooth transition of the putty applied in two adjacent workstations, some techniques are proposed, which will be introduced in this section. Section 6 introduces two Convolutional Neural Networks that effectively monitor the quality of plastering. A strategy based on the network outputs is proposed to determine appropriate adjustments to the plastering parameters based on the detected quality. In Section 7, a comprehensive experimental demonstration is presented. Through the integration of the proposed technologies in this experiment, the automatic application of putty on a large-scale wall is successfully achieved. Finally, in Section 8, the conclusions drawn from the research findings are discussed, along with an outlook on potential avenues for future research.

2 | RELATED WORK

At present, interior finishing robots find primary application in various tasks such as floor tiling[4,5], ceiling installation[6], tiling[7], wall polishing[8], painting[9], plastering[3], and quality inspection[10]. Within the domain of plastering robots, a distinction can be made based on whether their operations entail direct contact with the working environment.

2.1 | Related painting and plaster spraying robots

Several researchers have conducted studies on plaster spraying robots, exploring the possibility of direct wall spraying without the need for scraping. The concept of a painting robot shares similarities with a putty spraying robot, as both operate in a non-contact manner with the wall. One notable example in this field is Pictobot[9], which specializes in painting high-rise warehouse walls. Before commencing the spraying process, Pictobot scans and reconstructs the scene by integrating ToF camera data with robotic arm motions. It then segments the resulting 3D point cloud, generates a painting sequence, and plans the spray posture to ensure consistent paint thickness. However, it is important to note that Pictobot is a semi-automatic system that requires human assistance for precise positioning the platform and navigating around doors and windows. Another notable development in the field includes the introduction of a digital design plaster[11] and an interactive robot for ornamental work[12]. In both cases, a manipulator equipped with a spraying gun is employed to carry out the plastering tasks. The trajectory of the robot's operation can be carefully designed and followed, enabling the realization of various coating layers and shapes. Selen et al.[13] proposed a mobile robot capable of plaster spraying on building elements. This system is designed to adapt the spraying distance and velocity after each spraying iteration. Additionally,[14] and[15] provide insights into system composition and control planning, offering a mathematical perspective on the subject matter.

Interior finishing tasks like painting and plaster spraying without force perception or contact have made significant advancements. However, in the case of plastering, the desired level of surface smoothness and base putty adhesion cannot be achieved without the additional step of scraping. Furthermore, tasks that involve force-controlled contact encounter issues of instability and unreliable performance due to imprecise measurement of interaction forces. To address these challenges, researchers have devised a specially designed end effector to offer effective solutions.

2.2 | Plastering machines/robots

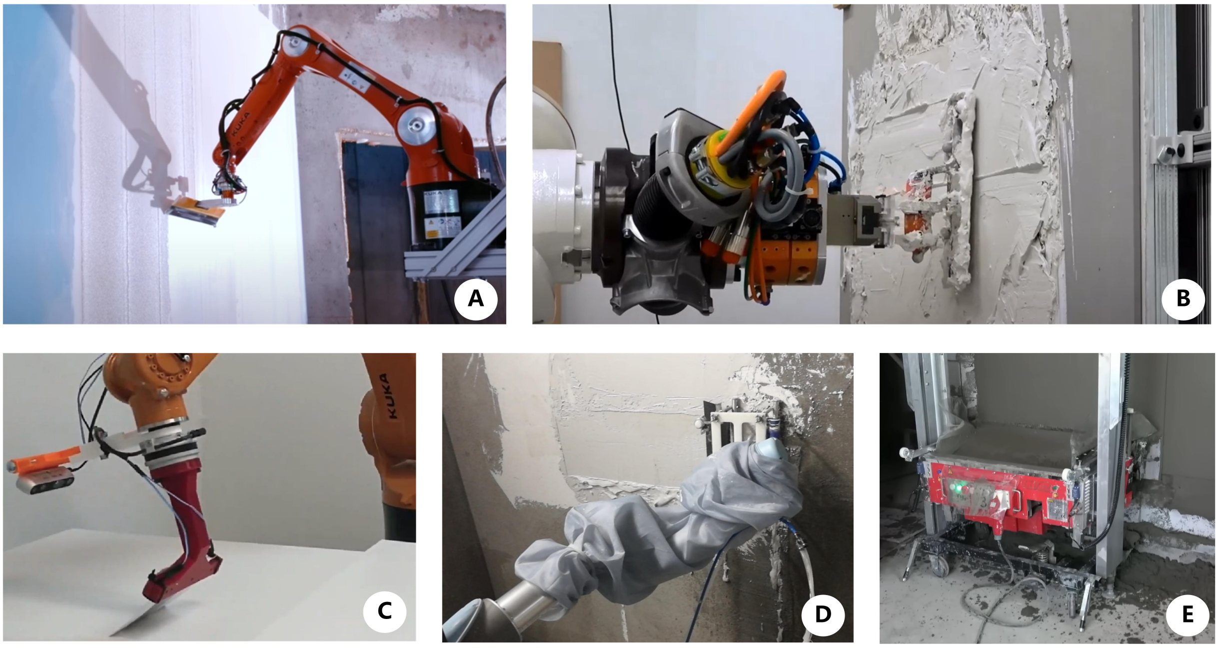


FIGURE 1 Plastering robots:(A)OKIBO plastering robot (B)CMU plastering robot (C)Plastering robot with a soft scraper (D)Plastering robot with a spring scraper (E)Semi-automatic wall-plastering machine.

The interior finishing robot company OKIBO from Israel [16] has developed a robot capable of both spraying and scraping tasks. The scraping operation is performed autonomously, eliminating the need for manual intervention. The end effector is equipped with a soft rubber scraper, and a thin layer of putty is applied. The target wall surface consists of KNAUF gypsum boards, known for their smoothness and superior flatness compared to concrete walls. However, even after the robot completes the process, some vertical scratches may remain on the wall, as illustrated in Fig. 1 -A. When dealing with cement-based materials and adhesive putty, the actuator and soft scraper may not meet the required putty thickness standards. Additionally, in the case of vertical plastering, putty can accumulate on the soft scraper, leading to inaccuracies in the scraper's processing angle. This issue becomes more pronounced when working with sticky putty. A rigid scraper is employed as the end effector for handling sticky putty, as depicted in Fig. 1 -B. By performing horizontal processing, the impact of accumulated putty caused by gravity can be minimized. However, the occurrence of scratches on the putty due to the scraper's edge remains unpredictable. To address this issue, Joshua et al. [17] proposed an image classification approach for defect detection and subsequent re-plastering. In a similar vein, Marsela et al. [18] utilized a camera to assess the deformation of the scraper, as shown in Fig. 1 -C, and regulated the inclination of the knife relative to the putty surface. Nonetheless, despite the precise control of the end effector, the elimination of scratches remains challenging. Additionally, the presence of putty on the scraper obstructs the camera's detection capabilities, limiting the effectiveness of this particular method. Li et al. [19] introduced a robotic system featuring a spring-loaded scraper as the end effector, as illustrated in Fig. 1 -D. The system leverages a neural network to classify defects and determine the optimal scraping direction. However, noticeable streaks persist in the putty between neighboring working areas. The Machine Manufacturing Co [20] developed a semi-automatic wall-plastering machine equipped with a lifting mechanism to address the limited reach of human workers. However, manual alignment and adjustment of the machine's position in relation to the wall are still necessary. One drawback of this machine is its limited autonomy, as workers need to manually align the station for each operation. Furthermore, this machine tends to consume more material than manual methods due to the increased amount of cement mortar that falls during the plastering process, as depicted in Fig. 1 -E.

Researchers commonly utilize scrapers made of materials with low stiffness, lacking the ability to adjust their stiffness according to the specific properties of the putty being used. Furthermore, the control of scraping angles in these machines is not dynamically regulated. Another drawback of current plastering robots lies in the strategy employed to ensure the desired flatness across the entire area during multi-station plastering.

Another crucial aspect of plastering automation pertains to the monitoring of quality. The inherent complexity of fluid putty's rheological properties leads to variations in surface shape, hardness, and viscosity over time[21]. Unlike rigid objects, putty cannot be handled in a fixed manner across different states. Consequently, it becomes imperative to assess the outcome after each plastering process and adaptively adjust the parameters accordingly. Quality monitoring assumes a pivotal role in plastering automation. Yan et al.[10] introduced a robotic system designed for autonomous quality inspection and assessment. This system employs cameras and laser scanners to scan entire rooms, detecting any structural defects. Liu et al.[22] implemented a concrete crack-detection method using U-net networks for vision data analysis. Elbehiery et al.[23] devised a control method that integrates image processing and morphological operation techniques to evaluate the surface quality of tiles. Furthermore, Convolutional Neural Networks[24,17,25] have been employed to detect and assess surface quality. These methodologies serve as valuable references for the monitoring of plastering quality.

3 | SYSTEM DESCRIPTION

The Puttybot is an interior finishing mobile robot platform that employs a macro/micro manipulator, as shown in Fig. 2 . The macro manipulator is a 6-DOF UR5 robot arm capable of large-scale motions. The micro manipulator is a 3-DOF parallel platform that helps to quickly respond to changes in the surface of the putty through precise motions. The combination of macro/micro manipulators can avoid manipulator singularity in a constrained environment and improve the dynamic response to variations in the wall surface[26].

3.1 | Robot platform

As shown in Fig. 2 , the Puttybot is mainly composed of a two-wheel drive mobile chassis, lift platform, UR5 robot arm, a parallel platform and a motorized scraper.

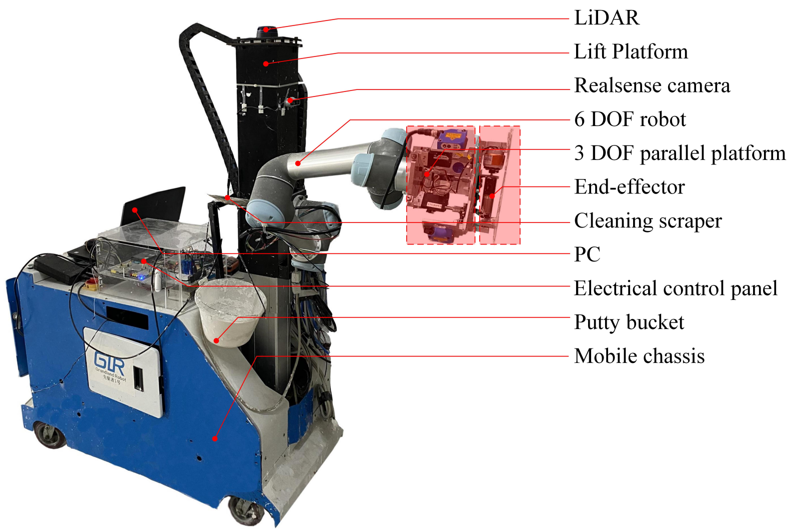


FIGURE 2 Robot hardware composition

To realize the communication between the various components, a Robot Operating System (ROS) was used as the software framework, and the communication frequency was 100 Hz, as shown in Fig. 3 . The main components are as followed.

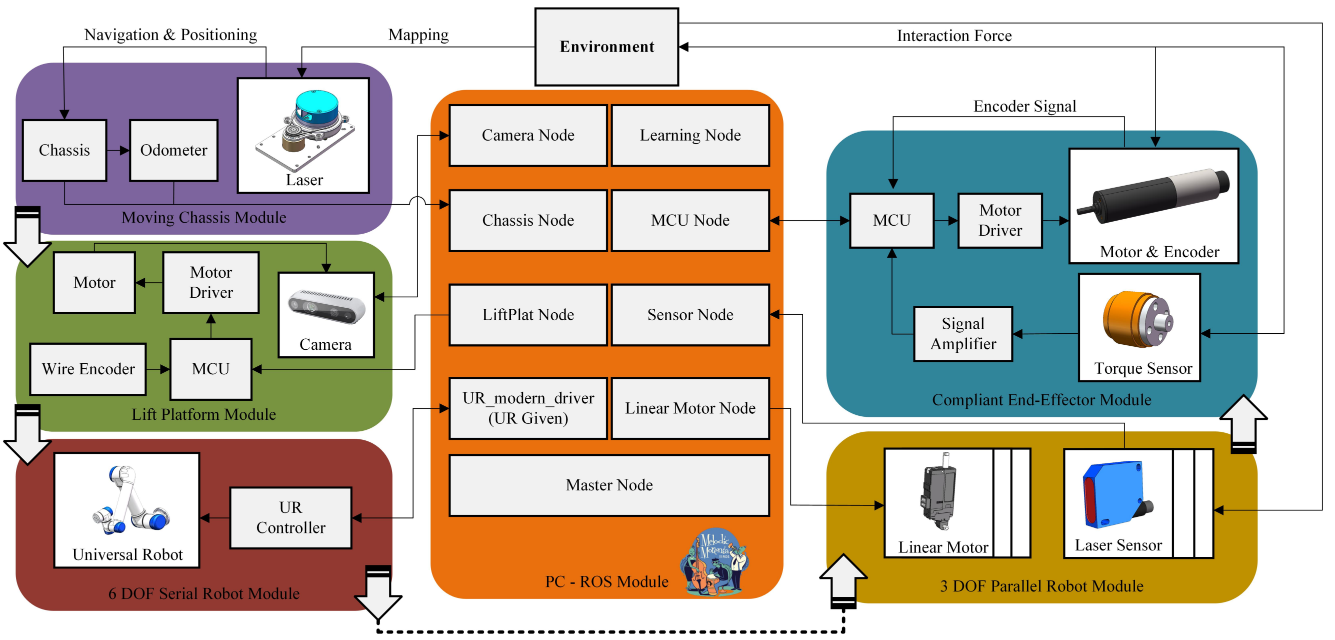


FIGURE 3 Robot module composition and system communication framework

- A control PC with an Intel Core i9-8950HK CPU @2.90GHz*12, and GeForce GTX1080 GPU running on Ubuntu 18.04 was used.
- The 6-DOF robot arm is a UR5 serial manipulator with a maximum payload of 5 kg and a control frequency of 125 Hz.
- The 3-DOF parallel platform is a 3-RPS parallel platform. The maximum payload and control frequency of each of the embedded linear motors are 4 kg and 200 Hz, respectively.
- The motorized scraper mounted at the end of the manipulator is the main component that contacts the wall, and its real-time response frequency can reach up to 1 kHz.
- The lift platform is equipped with a wire draw encoder, Realsense D435i, and 2D laser sensor.

3.2 | Micro manipulator and the motorized scraper

The micro manipulator is a parallel platform designed as a 3-RPS mechanism, which stands for a 3-revolute-prismatic-spherical parallel mechanism. It consists of a base that is connected to the end of the UR5 manipulator, and a moving plate equipped with a motorized scraper. For the parallel platform, the linear motor used is the MightyZAP L12-20PT-3, which provides a peak force of 40 N. Additionally, a laser sensor with a resolution of 0.1 mm, specifically the SensoPart FT50-RLA-220-S1L8, is incorporated into the system. The motorized scraper includes various components such as an aluminum scraper (250 mm × 90 mm), a rotary torque sensor, a GP22C gear reducer with a reduction ratio of 109:1, a Maxon DCX22 motor with a peak torque of 0.9 Nm, and an encoder.

During interior finishing tasks, the motorized scraper comes into contact with the putty or wall, requiring it to be durable and resistant to corrosion. It is also important to monitor the state of the putty for optimal operation. The putty scraper is connected to the moving plate through a hinge, and its rotation is driven by a servo motor. A rotary torque sensor is positioned between the output shaft of the servo motor and the hinge shaft to measure the contact force exerted by the scraper. The assembly of the micro manipulator with the scraper can be observed in Fig. 4 .

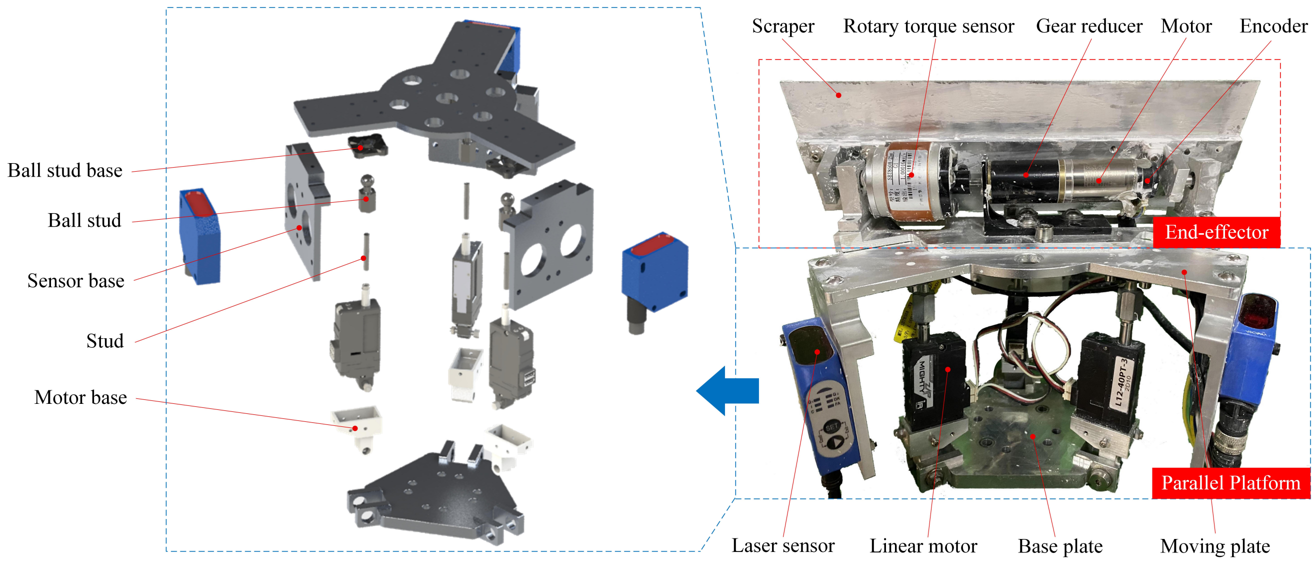


FIGURE 4 Micro manipulator with motorized scraper: the left side presents an exploded view of the parallel platform.

To prevent contamination of the finished surface and minimize wastage of putty, a specialized mechanism has been incorporated to effectively remove residual putty from the scraper, as depicted in Fig. 2 . This mechanism effectively removes any remaining putty from the scraper, which is then collected in a designated putty bucket for recycling purposes. The cleaning scraper is constructed using resilient manganese steel, ensuring its suitability for various scraper positions and postures.

4 | CONTROL METHOD

4.1 | Macro/micro manipulator control

4.1.1 | Position control of the macro/micro manipulator

The 6-DOF UR5 manipulator and 3-DOF parallel platform are connected serially forming a 9-DOF redundant manipulator. Denoting the Cartesian velocity of the end tip as $\dot{x}$, we have:

$$\dot{x} = J\dot{\theta} \tag{1}$$

where J is a 6×9 Jacobian matrix, and $\dot{\theta}$ is a 9×1 vector representing the velocities of the actuators. The Jacobian matrix can be formulated with two sub-matrixes as:

$$J = \begin{bmatrix} J_M & J_m \end{bmatrix} \tag{2}$$

where J_M represents the Jacobian matrix mapping the UR5 robot joint velocity to its end effector velocity, and J_m represents the Jacobian matrix mapping the linear actuator velocity to the velocity of the moving plate:

$$J_m = \begin{bmatrix} {}_5^W R & 0 \\ 0 & {}_5^W R \end{bmatrix} J_{RPS} \tag{3}$$

where ${}_5^W R$ represents the rotation matrix of the macro manipulator's end with respect to the world coordinate system, and J_{RPS} represents the Jacobian matrix of the parallel platform.

The relationship between the joint velocity and the velocity of the end effector is obtained by solving for J_{RPS}, ${}_5^W R$, and J_M with generalized kinematic inverse solution.

For operation in limited space, a position control method for redundant manipulator is utilized. The comprehensive kinetic model and its solution have been elaborated upon in the work by Chen D et al.[27].

4.1.2 | Control of the scraper force

A conventional impedance control method is applied when the scraper contacts the putty, as shown in Fig. 5 . The desired rotation angle of the scraper is specified according to the working conditions, whereas the desired impedance factors (stiffness and damping) are set in program. The contact force is measured using the rotary torque sensor mounted at the hinge of the scraper plate. To obtain the exact contact force, the measured value is subtracted by the weight of the mechanical part. The desired contact force following the impedance control is formulated:

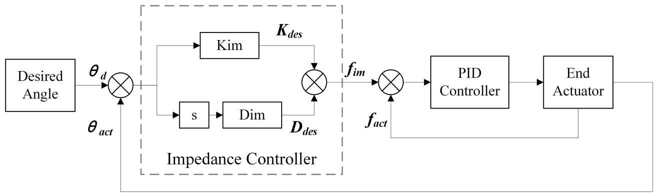


FIGURE 5 Force control framework

$$f_{im} = K_{im}(\theta_{des} - \theta_{act}) + D_{im}(\dot{\theta}_{des} - \dot{\theta}_{act}) \tag{4}$$

Where the f_{im} is the calculated torque by the impedance controller, the K_{im} and the D_{im} represents the stiffness and the damping parameters. θ_{des} represents the target degree of the scraper, and θ_{act} can be obtained by the encoder. The desired contact force f_{im} following the impedance model is realized by a PID control that regulates the force by comparing its actual value measured from the torque sensor.

4.2 | Mobile chassis navigation and positioning method

The Robot Operating System (ROS) incorporates the Adaptive Monte Carlo Localization (AMCL) algorithm to achieve accurate localization of the robot using a 2D floor map. AMCL utilizes a collection of particle filters within the 2D map to track and estimate the robot's position. This approach, based on Monte Carlo fusion algorithms, is particularly effective for large-scale localization tasks[28].

To facilitate accurate localization, the implementation of AMCL incorporates a laser sensor mounted on the top of lift platform. Considering the specific characteristics of the interior finishing environment and the motion capabilities of the mobile chassis, the AMCL is configured with a minimum of 1000 particles and a maximum of 6000 particles. This configuration allows for effective real-time position updates at a frequency of 10 Hz.

5 | PLASTERING PARADIGM WITH THE PROPOSED ROBOT

5.1 | The essential parameter for fine plastering

The coordinate system utilized by the Puttybot is illustrated in Fig. 6 . O_b denotes the origin of the UR5 base coordinate system, and O_t signifies the origin of the end effector coordinate system. Notably, the end effector coordinate system is positioned at the center of the moving plate within the parallel mechanism.

During the plastering process, the Puttybot performs adjustments to three key angles, namely ζ, ϕ, and θ_S. Considering that in plastering, the moving plate holding the scraper moves along the Y_b axis, then ζ represents the yaw angle. The yaw angular motion facilitates the control of putty thinness. On the other hand, ϕ denotes the roll angle. The roll angle describes the tilt angle of the scraper with respect the wall. By setting the roll angle, only the bottom edge of the scraper will leave a putty stripe after each lateral scraping. Lastly, θ_S corresponds to the scraper rotation angle, which governs the attainment of putty flatness.

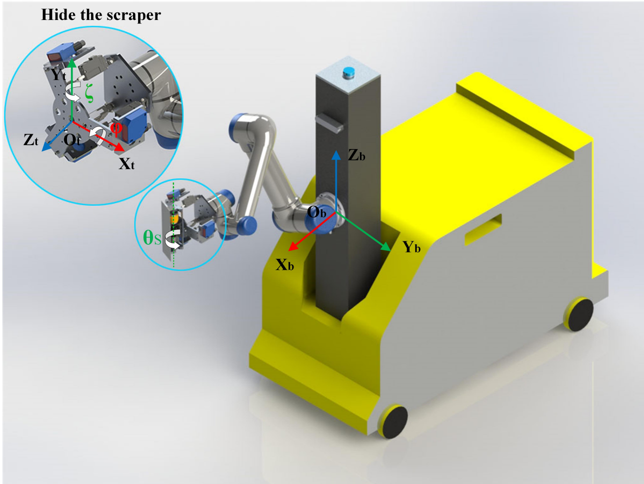


FIGURE 6 The three key parameters ζ, ϕ and θ_S used for describing the automatic plastering process. ζ represents the yaw angle of the moving plate (rotation around Z_b). ϕ denotes the roll angle of the moving plate (rotation around Y_b). θ_s corresponds to the rotation of the scraper plate. O_b denotes the origin of the UR5 base coordinate system, and O_t signifies the origin of the end effector coordinate system.

5.2 | Workflow

5.2.1 | Localization of target wall

Once the mobile chassis has arrived near the target wall, a calibration procedure is employed to determine its precise location with respect to the robot's position. This calibration method is visually depicted in Fig. 7 . The end of the robot's manipulator is guided along a straight path, specifically along the Y_b axis. Meanwhile, the moving plate of the parallel mechanism is maintained parallel to the wall, aided by the laser displacement sensors that provide the necessary positional information. By measuring the generated values of $\Delta laser$ and $\Delta run\ distance$ between adjacent positions, the inclination angle of the wall when projected on the ground can be accurately calculated.

$$\sigma_M = \sigma_W = \arcsin(\frac{\Delta laser}{\Delta run\ distance}) \tag{5}$$

At the outset, the orientation of the moving plate with respect to the wall is indeterminate. However, utilizing distance measurement feedback from the three laser sensors, its orientation is promptly adjusted to align parallel to the wall. By ensuring this parallel alignment, any observed changes in the measured normal distance correspond proportionally to the displacement of the end effector. Ultimately, a straight-line fitting approach is employed to determine the calibration line, as depicted in Fig. 8 .

In Fig. 10 (b), the red and blue curves depict the calibrated and actual paths, respectively. Throughout the scraping process, the actual path is adjusted based on the varying measured torque caused by the unevenness of the wall surface, ensuring the continuity of the operation. The scraping operation is performed in the calibration direction.

TABLE 1 Stiffness and damping coefficients

Category	Stiffness	Damping
Scraper-Soft	5	0.2
Scraper-Standard	20	0.1
Scraper-Hard	50	0.1

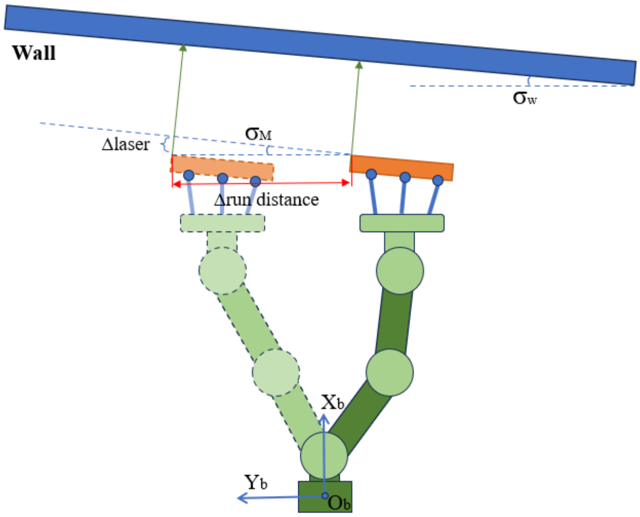


FIGURE 7 Calibration of the wall's position: O_b denotes the origin of the UR5 base coordinate system. σ_M is the calculated angle of the wall, and σ_W is the actual angle of the wall.

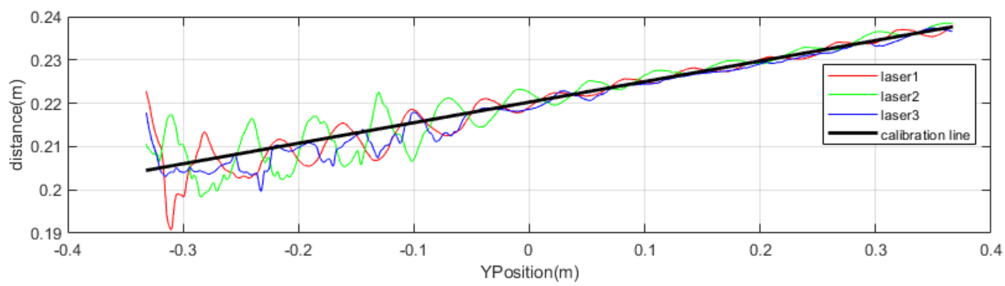


FIGURE 8 Recognition of the target wall surface: the moving plate armed with laser sensors is maintained parallel to the wall, stabilizing into a linear relationship gradually.

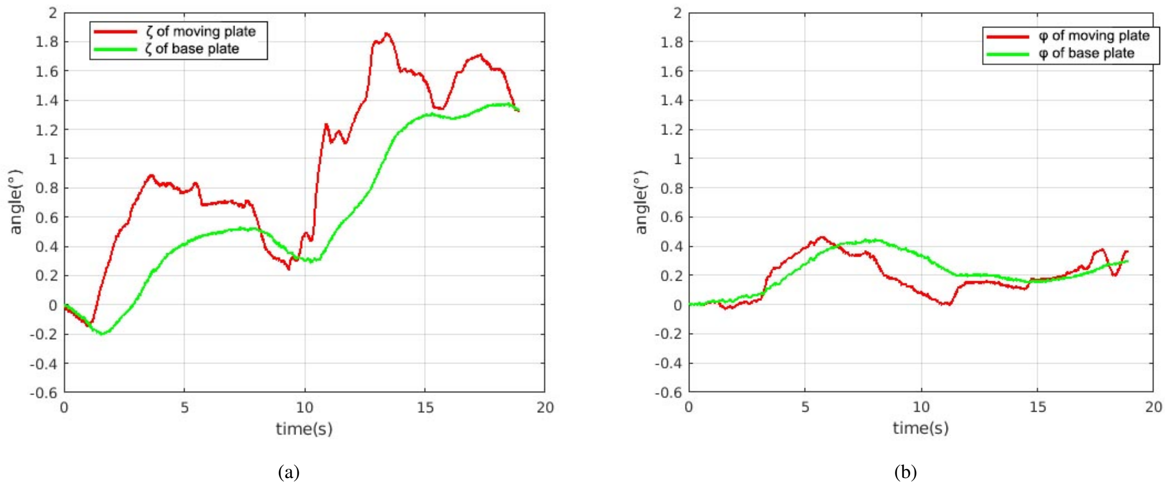
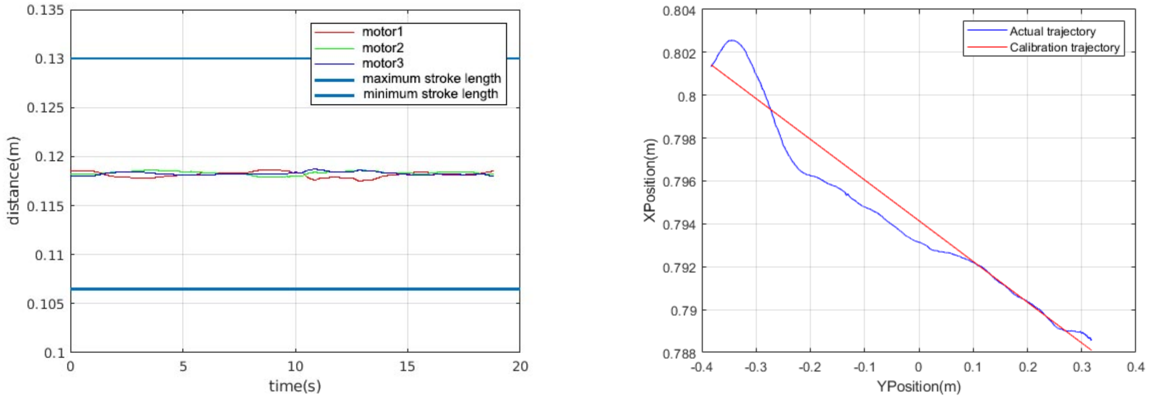


FIGURE 9 Strategy for motion and base plate compensation: (a) The ζ angular time response curves of the moving and base plates; (b) The ϕ angular time response curves of the moving and base plates. The pose change of the base plate lags behind that of the moving plate to enhance system stability.



(a) (b)

FIGURE 10 (a) Variation of stroke length: the stroke lengths of the linear motors maintain within the middle range of their movement capabilities to adapt wall surface changes; (b) Comparison of the calibrated path and actual path.

5.2.2 | Putty plastering with motorized scraper

The actual putty plastering process was realized using impedance control with variable settings of stiffness and damping coefficients. The objective of the control was to maintain the scraper inclination angle with respect to the wall to be $\theta_S = 30°$.

The three sets of stiffness and damping coefficients adopted in the experiments are listed in Table. 1 , which were tuned through tests under various working conditions to ensure good quality.

Due to the restricted motion range of the parallel mechanism, its ability to actively adapt to an uneven wall surface is greatly influenced by the current stroke length. When any of the linear motors reaches its limit of stroke length, no further adjustments in posture can be accomplished by the parallel mechanism. In such instances, the manipulator, which holds the parallel mechanism, must undertake the necessary adjustments to the end effector posture.

Considering the ability of both the parallel mechanism and the manipulator to modify the scraper's orientation with respect to the wall, it is crucial to devise a strategy that effectively harnesses their respective capabilities. The approach employed in this study is outlined as follows:

(1) The parallel mechanism, due to its excellent responsiveness, can promptly adapt to significant fluctuations in the elevation of the wall surface. Nonetheless, its capacity to accommodate variations in wall surface elevation is constrained by its limited range of motion. To ensure continuous readiness of the parallel mechanism for addressing wall surface changes, it is advisable to maintain the stroke lengths of all linear motors within the middle range of their movement capabilities.

(2) For the manipulator, its capability to adjust the scraper orientation should not be utilized when a drastic change in all surface elevations is encountered, because it implies a large dynamic load for the manipulator. Its tool-orientation adjustment capability should be utilized after the drastic change has been compensated for by the parallel mechanism. Its adjustment is to make the parallel mechanism move back to the neutral stroke position to make full use of its high response in subsequent operations. Therefore, the pose change of the manipulator lags behind that of the parallel mechanism.

To implement the compensation strategy described above, we have adjusted the PID parameters employed in the motion control of the manipulator. Specifically, a lower P gain has been configured to intentionally introduce a delay in the manipulator's posture tracking compared to the parallel mechanism. This deliberate adjustment ensures that the manipulator UR5 exhibits a lower response in terms of posture tracking. The figures illustrating typical tracking data, namely Fig. 9 (a) and Fig. 9 (b), demonstrate the resulting performance. Additionally, it is worth noting that the stroke positions of the linear motors have been maintained within the middle range, as evidenced by Fig. 10 (a).

In order to mitigate the occurrence of mechanical singularity configurations in the manipulator while executing the putty-plastering process, it is crucial to optimize the initial position of the end effector and devise appropriate protective measures. To this end, we employ the Levenberg-Marquardt (LM) method [29], wherein the resolved motion rate control formulation is enhanced by the inclusion of the λ weighting factor. This approach effectively prevents undesirably high joint velocities, ensuring a smooth and safe operation.

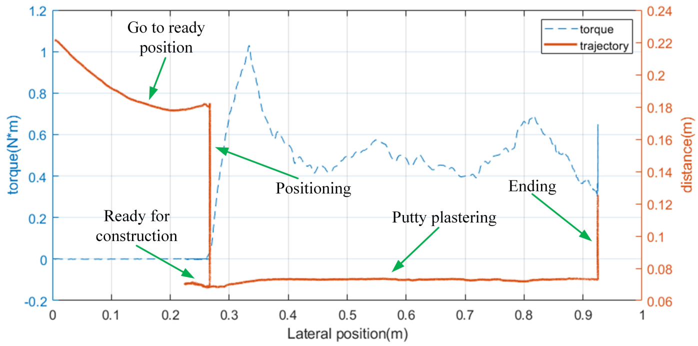


FIGURE 11 Measured torque variation and the trajectory of the moving plate

5.2.3 | Ending takeoff of the scraper

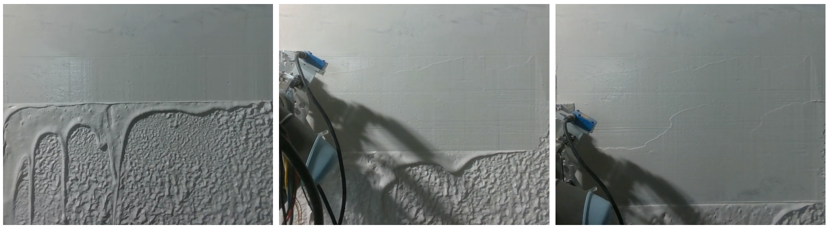
During the end of a plastering motion, the scraper is retracted from the wall, accompanied by an increase in the angle θ_S such as making the blade of the scraper taking off from the wall surface. This deliberate adjustment aims to minimize the adhesive force between the putty remnants and the scraper. In order to achieve this, it is necessary to design a suitable rotation speed $\dot{\theta_S}$ and manipulator tip escaping speed $\dot{Z}_t$. It is hard to eliminate the trailing putty due to the varying consistency of the putty, which produces different sticking forces. After comparing the results of several experiments, we empirically determined to set $\dot{\theta_S} = 8°/s$, which can effectively reduce the generation of pulled putty.

For the entire plastering process, Fig. 11 shows the measured torque variation and trajectory of the moving plate during a complete scraping.

5.3 | Plastering strategy in transition area

5.3.1 | Strategy between grids

While engaged in the plastering process, it has been observed that when the rectangular scraper blade aligns parallel to the putty surface, scratches occur along both edges of the scraper. To address this issue, the Puttybot proactively modifies the tilt angle ϕ of the blade, positioning the bottom edge of the blade closer to the wall than the top edge. Consequently, only the bottom edge of the blade contributes to the generation of scratches. To facilitate efficient plastering, a fundamental operation has been devised, involving lateral movement of the scraper from the leftmost to the rightmost point of the target grid.



(a) (b) (c)

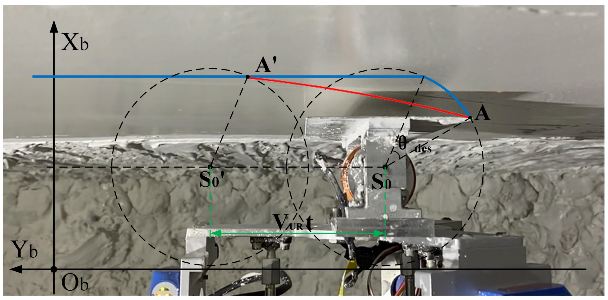
FIGURE 12 Effect of putty plastering with tilted blade: (a) First plastering; (b) Second plastering; (c) Third plastering.

Following each plastering operation, a deliberate displacement of the scraper by half of its width in the vertical direction was implemented. As a result, half of the previously plastered area is effectively covered in the subsequent step. This strategic approach serves a dual purpose: ensuring comprehensive coverage of the target wall area and mitigating the presence of scratches. Specifically, the scratches produced by the bottom edge of the blade in the previous step are rectified during the subsequent plastering process. The impact of this strategy is visually depicted in Fig. 12 .

5.3.2 | Strategy between workstations

When the mobile chassis transitions to the next workstation, the scraper is deliberately rotated to the desired angle and brought into contact with the putty. Once the measured torque stabilizes, the manipulator commences lateral movement for scraping. It is noteworthy that the initial contact between the scraper blade and the wall greatly influences the resulting scratch. Through our research, we have discovered that by increasing the tangential speed of the scraper blade relative to the contact surface, the severity of the scratch can be significantly mitigated. A comparative analysis between this innovative approach and the conventional strategy is presented in Fig. 13 (a).

In Fig. 13 (a), the blue and red lines represent two distinct types of approaching trajectories. The corresponding plastering effects of these two strategies are visually depicted in Fig. 13 . As observed from the figures, when the scraper blade follows a circular curve (represented by the red curve), the impact of the contact is minimized, resulting in almost imperceptible remaining scratches. We have successfully implemented this strategy for a continuous sequence consisting of three workstations, as introduced in [30].



(a)

(b) (c)

FIGURE 13 Different approach trajectories of scraper blade for the first contact: (a) Bird-eye view; (b) Actual plastering effect of blue trajectory; (c) Actual plastering effect of red trajectory.

6 | AUTOMATIC DETECTION OF PLASTERING QUALITY

Manual plastering relies on the coordination of the eyes, arms, and brain to determine the force and plastering angle. The same "eye-hand-brain" system was utilized for the Puttybot. The Puttybot was equipped with a Realsense D435i RGB-D camera, which was used to monitor the status of the process. The function of the "brain" is realized by a deep learning-based method. A Convolutional Neural Network was used to evaluate the plastering effect. If the effect of any plastering operation is detected as poor, the network should output a command for one more recovery plastering action, and it should also determine the appropriate parameter set for the new action.

6.1 | Plastering dataset collection

During the putty plastering process, in order to enhance efficiency, Puttybot initially carried out numerous trials using pre-set parameters. The scraping action was performed by manually selecting from these pre-set parameters, without employing any automatic detection methods. The results of these tests demonstrated that the diverse range of plastered putty conditions encountered in real-world interior finishing tasks could be effectively managed using a limited set of predetermined plastering parameters. Consequently, the development of an algorithm capable of automatically determining the appropriate parameter holds the potential to facilitate robotic plastering operations.

The wall putty state detection is divided into two steps.

(1) Judging the quality of plastering results is a binary classification problem; the network decision is either qualified or unqualified. If the result is qualified, the current processing grid is completed, and Puttybot then proceeds to the next processing grid.

(2) If the plastering result is recognized as unqualified, Puttybot needs to output the appropriate processing parameters according to the putty defect state, which is a multi-classification problem. Puttybot repeats the plastering operation using the newly determined parameters. This process continues until the processed wall is determined to be qualified.

As the binary classification label, "1" means that the plastered wall satisfies this requirement, and "0" means that the plastered effect does not.

The multi-classification network outputs three parameters related to the plastering effect: scraper position, scraper stiffness, and scraper tilt angle, which are listed in Table. 2 .

TABLE 2 Multi-classification network labels

Scraper position	Scraper stiffness	Scraper tilt angle
0.24	Scraper-Hard	PPR-Lower
0.12	Scraper-Standard	PPR-Standard
0.0	Scraper-Soft	

- Scraper position: For one grid, Puttybot is responsible for processing a rectangular area on the wall. Its height is 0.48m. According to the strategy mentioned in Subsection 5.3.1, this area is planned to be processed by three lateral plastering motions that start with the scraper at positions with the height 0.0m, 0.12m, and 0.24m respectively.
- Scraper stiffness: According to Table. 1 , the scraper stiffness was set to one of three levels.
- Scraper tilt angle: Through many experiments, two values of tilt angle were proven capable of meeting this requirement. PPR-Lower corresponds $\phi = -0.75°$ and PPR-Standard represents $\phi = 0.0°$.

TABLE 3 Pre-determined plastering parameters

Parameter set No.	Scraper position	Scraper stiffness	Scraper tilt angle
1	0.24	Scraper-Hard	PPR-Lower
2	0.12	Scraper-Standard	PPR-Lower
3	0.0	Scraper-Standard	PPR-Lower

The Convolutional Neural Network based method is designed for planning recovery re-plastering operations. Thus, the data collection starts after the first plastering to the target region has been completed and full coverage with the putty has been realized. The parameters used to realize the first coverage are listed in Table. 3 . From this moment, any further plastering action to remove the resultant defects is recorded as the data for training.

Dataset acquisition was performed by controlling Puttybot for plastering, using a Realsense D435i mounted on the lift platform. To ensure the validity of the dataset, the following principles were applied.

(1) Removal of the target stripes.

(2) No new stripes are created at the upper edge of the scraper.

(3) The presence of multiple defects is to be processed sequentially from top to bottom.

6.2 | Plastering parameter determination network

Through extensive testing, it has been discovered that the coefficients governing the plastering process can be determined based on visual features observed on the plastered wall surface. Consequently, the utilization of sensor feedback in an automated plastering system can be formulated as a vision-based classification problem. In this framework, the current state of the target wall surface, as captured by a camera, serves as the input to the network, while the output of the network corresponds to a selection from a predetermined set of plastering coefficient combinations.

Deep learning has recently demonstrated promising results, sometimes outperforming human performance in many computer vision tasks[31,32], such as classification[33,34] and object detection[35] and semantic segmentation[36,37]. Convolutional Neural Networks (CNNs) can efficiently and automatically separate out discriminative and valuable features directly from raw input images. In addition to maintaining the end-to-end training scheme, these capabilities of CNNs greatly reduce the expense and labor needed by conventional machine learning methods to extract handcrafted features manually.

Training CNN models for supervised machine learning tasks requires a large number of annotated images to avoid the overfitting effect. This poses a problem when the available annotated training dataset is limited and acquiring labelled images is labor-intensive, difficult, and time-consuming, as for our automatic wall putty scraping task. Shallow layers of CNNs learn features from input images that are general and task independent. By adapting the network architecture to the downstream task and fine-tuning the network parameters, a CNN trained on a large dataset for one specific vision task can be used in another related vision task.

The proposed networks are mainly based on a CNN (AlexNet) pre-trained on a sizable dataset (the ImageNet) for an object classification task. Then the knowledge gained from this task is transferred to solve our downstream tasks.

As shown in Fig. 14 , in addition to the original AlexNet classifier, two similar neural networks are added to the model. The binary network detects the plastering effect. The multi-classification network detects the plastering parameters: scraper position, scraper stiffness, and scraper tilt angle. The output layer of each classifier had a linear activation function. After applying the

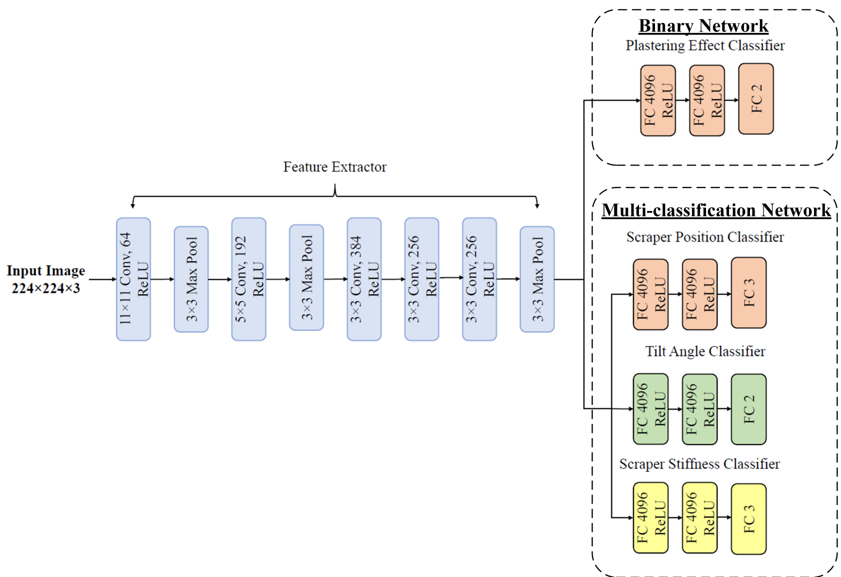


FIGURE 14 Network structure of the binary network and muti-classification network

average pooling operation, for an input image of size $(224 \times 224 \times 3)$, the size of the feature vector generated from the feature extractor part (convolutional layers) was $(6 \times 6 \times 256)$. This feature vector was then fed into the subsequent classifiers.

The structure of the model's feature extractor part is as: The first convolutional layer had a kernel filter of size (11×11) with 64 channels, stride = 4, and padding = 2. For an input image of size $(224 \times 224 \times 3)$, the feature map size after this convolution operation was $(55 \times 55 \times 64)$. Then, a ReLU nonlinearity and a max pooling operation with a kernel size of (3×3) and stride = 2 are applied; thus, the feature map size becomes $(27 \times 27 \times 64)$. This feature map was fed to the second layer.

The second convolutional layer has a kernel size of (5×5) with 192 channels and a stride of 1. The feature map size of this layer is $(13 \times 13 \times 192)$. The next three convolutional layers are similar, but have different numbers of channels. The three layers have a (3×3) kernel size with stride = 1 and padding = 1; therefore, the width and height of the propagating feature map remain fixed.

The third convolutional layer has a 384-channel kernel filter; thus, the feature map size is $(13 \times 13 \times 384)$. The fourth and fifth convolutional layers have a 256-channel kernel filter; therefore, the output feature map size from either layer is $(13 \times 13 \times 256)$. Finally, a max pooling operation with a (3×3) kernel size and stride = 2 is applied; thus, the size of the final feature map produced from the model feature extractor is $(6 \times 6 \times 256)$.

The feature extractor parameters were initialized with the values pre-trained on the ImageNet dataset and then fine-tuned during the training process. The parameters of the classifiers were randomly initialized and then fine-tuned during the training process.

The binary classification network is the same as the original AlexNet architecture, except that we modified the final output layer to have only two neurons with a linear activation function to match our task.

6.3 | Network training and testing

The training setup parameters of the classification networks were set as listed in Table. 4 . To further improve the computational speed, the high-resolution images were resized to 224 × 224 pixels. Because we have a small dataset, our model is prone to overfitting. We applied horizontal flipping to the training dataset to improve model generalization and mitigate the overfitting effect.

The training accuracy of the binary classification network was 100%, and the testing accuracy was 94.7%. Only one image was misclassified.

The original architecture of AlexNet was adapted to fulfill the requirements of our multi-classification network. Specifically, three additional neural network classifiers of the multi-classification network were incorporated to predict the parameters of scraper position, tilt angle, and scraper stiffness. Accuracy and error rates were individually computed for each parameter using the available image samples, encompassing both training and testing sets. Net accuracy was determined by assessing the simultaneous correct prediction of all three parameters within the sample set. Notably, the training accuracy for each classifier reached 100%, while the testing accuracies for scraper position, tilt angle, and stiffness classifiers were recorded at 92.8%, 92.8%, and 78.5%, respectively. The overall testing accuracy amounted to 57%. Notably, a positive correlation was observed between the scraper position and the remaining two parameters, resulting in the model attaining optimal testing accuracy for scraper position while achieving the best accuracy for the remaining parameters. Consequently, the model that yielded the highest testing accuracy for scraper position was saved and subsequently utilized for inference purposes.

For the network deployment, action communication in ROS is applied. An action does not block other tasks during server-side processing. In contrast, the client side can interrupt or send new target tasks when the server side processes tasks and receives the corresponding processing information.

7 | PUTTYBOT AT THE CONSTRUCTION SITE

We tested the Puttybot on an actual construction site consisting of a room with an area of 10 m × 7 m, three concrete walls, and one floor-to-ceiling window.

A rectangular area of 3.7 m × 1.2 m on the longest wall of the room (size of approximately 10 m × 3 m) was selected for testing the automatic plastering robot.

TABLE 4 Training parameters for classification network

Network	Dataset size(train+test)	Training epochs	Batch size	Optimizer	Learning rate
Binary	536+134	100	5	SGD	0.001
Multi-classification	451+113	300	5	SGD	0.001

For the purpose of this test, the interior wall putty was mixed with water in a mass ratio of 3:1. To ensure the uniformity of the putty, a sprayer machine was employed for coating. However, it should be acknowledged that during the spraying process, some putty may flow along the wall and eventually settle on the ground, resulting in certain wastage. The target area was partitioned into an 18-section grid, comprising three rows and six columns. Within each section, the proposed approach involving status detection and plastering with visual feedback was implemented. It is important to highlight that to ensure comprehensive coverage of the designated region, Puttybot necessitates completing six lateral movements of the mobile chassis and three vertical adjustments of the lift mechanism, as illustrated in Fig. 15 .

The effects observed on the grid following the plastering procedure are depicted in Fig. 16 . In certain sections, the surface quality satisfies the requirements after a single plastering operation, rendering re-plastering unnecessary (re-plastering times: 0). Conversely, several grids necessitated multiple re-plastering attempts. Notably, grid No.16 underwent five re-plastering cycles, and the evolution of its surface quality from the initial to the fifth recovery plastering is presented in Fig. 17 . The corresponding labels predicted by the multi-classification network are detailed in Table 5 . Despite the application of five plastering attempts, the defect could not be entirely eliminated, resulting in the same output for the sixth attempt. To prevent entering an infinite loop, the grid was bypassed, thereby maintaining the selection of the same parameters for the three consecutive processes. Subsequently, Puttybot proceeded to the subsequent grid. However, due to the overlapping nature of the adjacent regions, the horizontal streak left unresolved in grid No.16 was effectively addressed in grid No.17.

TABLE 5 Output parameters from the multi-classification in grid No.16

Re-plastering No.	Scraper position	Scraper stiffness	Scraper tilt angle
1	0.12	Scraper-Standard	PPR-Standard
2	0.24	Scraper-Standard	PPR-Standard
3	0.0	Scraper-Standard	PPR-Lower
4	0.12	Scraper-Standard	PPR-Lower
5	0.12	Scraper-Standard	PPR-Lower
6	0.12	Scraper-Standard	PPR-Lower

To gather the necessary data for evaluating the plastering effect, point cloud data of the room are collected. The collected point cloud is processed using SCENE software. Following autonomous plastering by the Puttybot, the processing area is recalculated and measures as 3.5830 m × 1.1009 m. The robot navigates through six workstations during the entire process, and the position of the Puttybot in the world coordinate system after plastering is recorded, as illustrated in Fig. 18 . Furthermore, no discernible defects are observed in the transition area between workstations, underscoring the robustness of the Puttybot and its ability to overcome challenges associated with positioning accuracy.

The point cloud representation of the untreated wall is illustrated in Fig. 19 (a). The wall's concrete base is discernible, with the presence of white spots denoting adhered putty particles. Subsequently, the wall after the Puttybot's automated processing is depicted in Fig. 19 (b). Leveraging the point cloud data from the plastered area, we employed the random sample consensus (RANSAC) algorithm to fit a plane to the area. Using this fitted plane as the reference, a heat map was generated wherein the point cloud was color-coded based on its distance from the reference plane. This heat map is presented in Fig. 19 (c). The maximum positive deviation and negative deviation from the reference plane were measured at 4.9 mm and -4.6 mm, respectively. Consequently, the overall flatness of the surface was less than 5 mm, exhibiting no noticeable pits, hollow drums, or other defects.

8 | CONCLUSION AND OUTLOOK

In this research, we present Puttybot, a robotic system equipped with a macro/micro manipulator designed for putty plastering tasks. We propose a new paradigm for robotics plastering, validating the effectiveness of its workflow and plastering strategy through autonomous plastering conducted at multiple workstations. The integrated technology enable the robot to autonomously recognize the processing quality and plan re-plastering operations. These features are not achieved in previous construction robots. The comparative analysis of manual plastering, OKIBO, and Puttybot, focusing on construction capacity and efficiency,

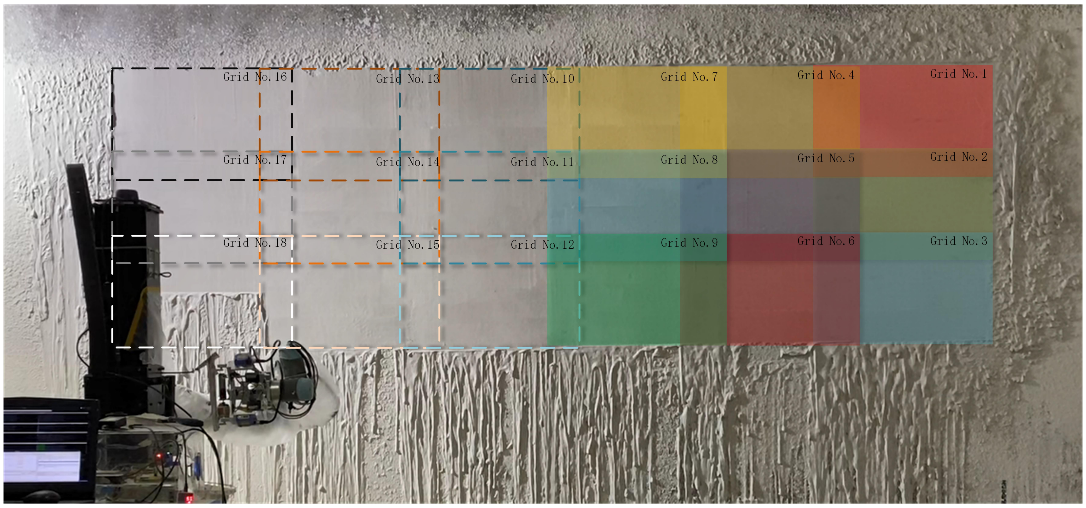


FIGURE 15 Puttybot processing grid No.18

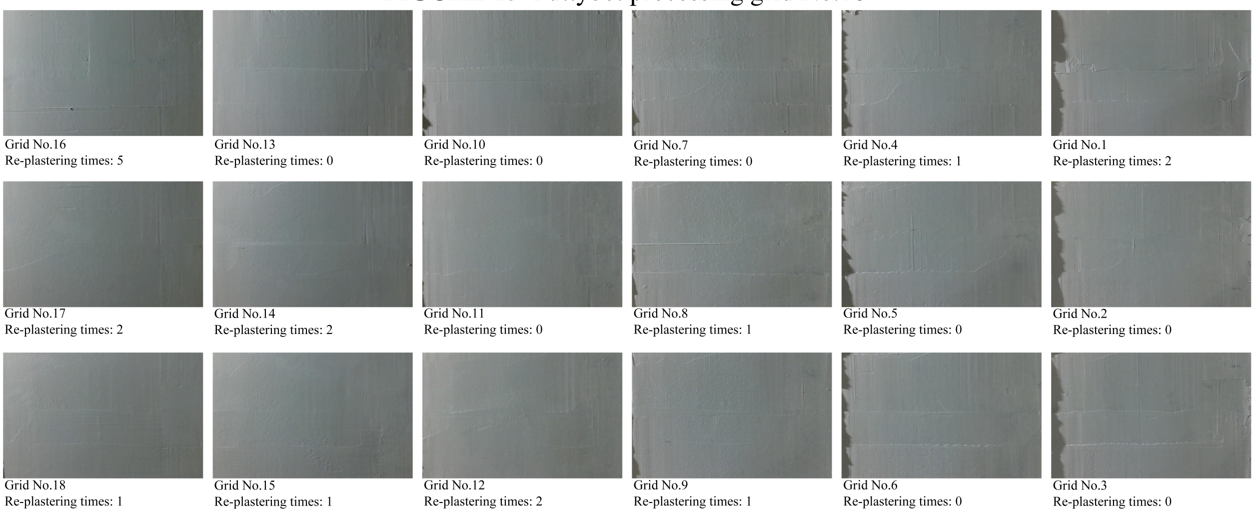


FIGURE 16 Plastering results of each grid (Re-plastering times indicate the number of re-plastering operations conducted)

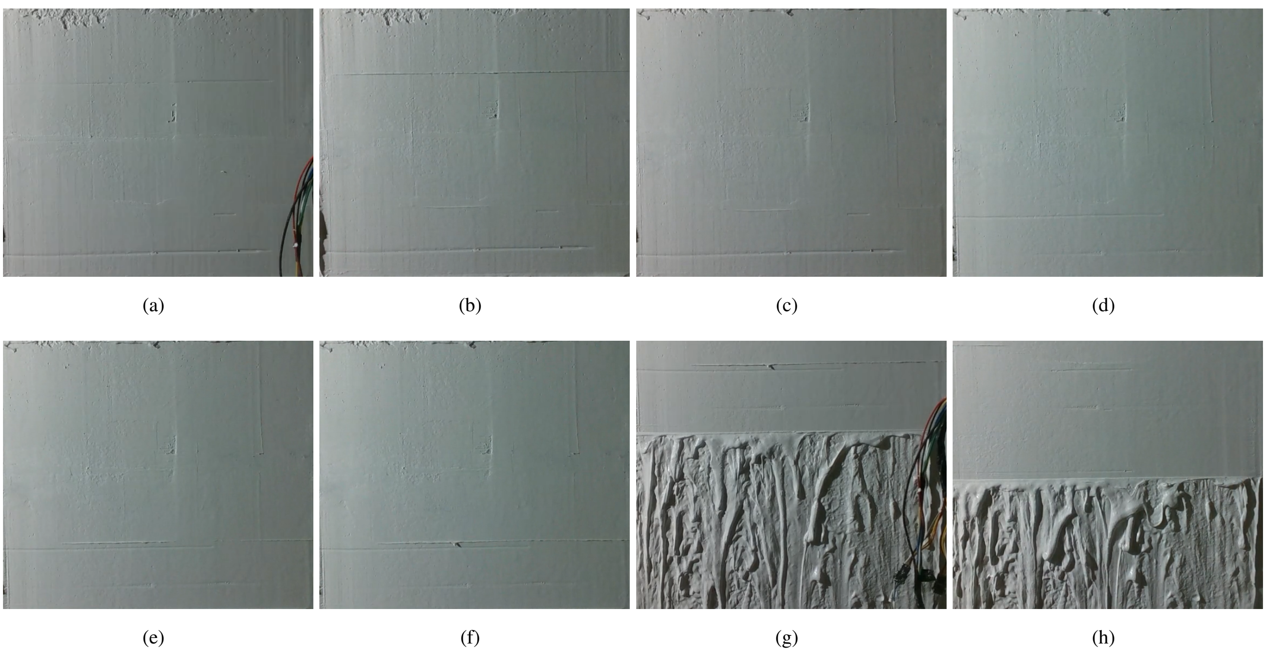


FIGURE 17 Variation of defect in grid No. 16: (a) Defect in grid No. 16 after pre-programmed lateral plastering; (b)-(f) Re-plastering result in sequence; (g) Grid No.17 before plastering; (h) Grid No.17 after first plastering.

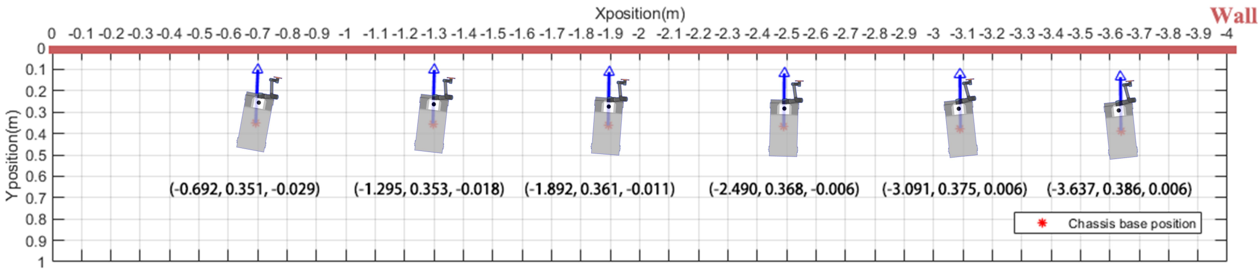


FIGURE 18 Puttybot's posture during plastering: the Puttybot moved from right to left with 6 workstations.

is presented in Table. 6 . However, it is important to note that these statistics alone may not accurately depict the actual business value due to variations in cost and plastering standards. A study[38] highlights the daily total production in Turkey. Based on interviews with workers and online data sources[39], three skilled workers require a day to complete an area of about 120 m^2, with each skilled worker costing about 300 RMB per day. OKIBO claims to be three times more efficient than manual work[16]. Regarding Puttybot, it should be noted that the operational speed is limited due to safety requirements during experimental trials, resulting in reduced construction efficiency. In addition with large-sized scraper, the construction speed can be easily improved, which needs to be optimized. The cost of using Puttybot includes expenses for a corner-processing plasterer, amounting to about 150 RMB per day. In conclusion, Puttybot offers cost advantages, fewer limitations, and meets the coating requirements effectively.

To further improve plastering quality, more experiments using the plastering parameters illustrated in this paper are required. In more complex environments, such as internal and external corners, piping, and socket holes, Puttybot also needs to perceive the geometric structure. In addition, wall painting can also be realized by changing the end effector. As the research in this study is further developed, Puttybot will carry out more functions and accomplish more renovation tasks.

TABLE 6 Comparison of plastering capabilities and efficiency of manual, OKIBO, and Puttybot

Builder	Area/day	Cost/day	Limitation	Effect
Manual	120 m^2	900 RMB	Harmless environment	Limited stability
OKIBO	360 m^2	Unknown	Thin putty, flat base, corners	Stripes between neighbor zone
Puttybot	60 m^2	150 RMB	Corners	Flatness of 5 mm

ACKNOWLEDGMENT

This work was supported by the following project: National Natural Science Foundation of China under Grant No. U1813202.

References

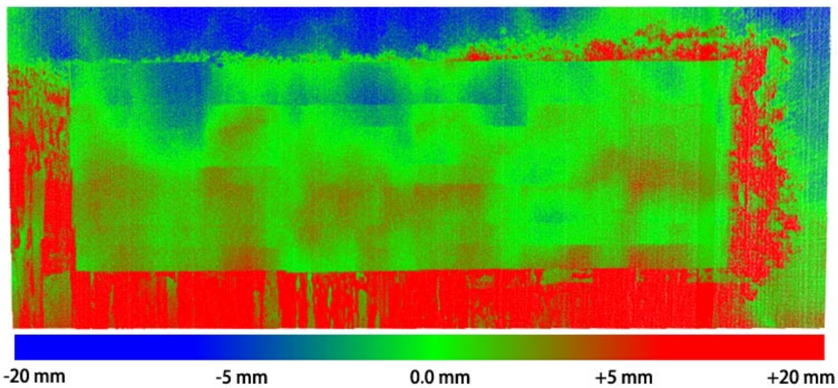
1. Bademosi FM, Issa RRA. *Automation and Robotics Technologies Deployment Trends in Construction*: 1–30; Cham: Springer International Publishing . 2022. https://doi.org/10.1007/978-3-030-77163-8_1

2. Bock T. The future of construction automation: Technological disruption and the upcoming ubiquity of robotics. *Automation in Construction* 2015; 59: 113–121. doi: https://doi.org/10.1016/j.autcon.2015.07.022.

3. Bulgakov A, Bock T, Otto J. Robot Manipulators for Plastering Work. In: 2019 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon). IEEE; 2019; Vladivostok, Russia: 1–5

(a)

(b)



(c)

FIGURE 19 Autonomous plastered result: (a) Point cloud of the untreated wall; (b) Plastered effect; (c) Heat map of plastered area.

4. Wang S, Zhou H, Zhang Z, Zheng X, Lv Y. Robot Floor-Tiling Control Method Based on Finite-State Machine and Visual Measurement in Limited FOV. *Advances in Civil Engineering* 2021; 2021: 1–16. https://www.hindawi.com/journals/ace/2021/8372815/doi: 10.1155/2021/8372815

5. Liu T, Zhou H, Lan H, Li Y, Wang Q, Che X. Robotic floor-tiling platform system design. In: IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society. IEEE; 2019; Lisbon, Portugal: 5269–5274

6. Fukui R, Yamada Y, Mitsudome K, Sano K, Warisawa S. HanGrawler: Large-Payload and High-Speed Ceiling Mobile Robot Using Crawler. *IEEE Transactions on Robotics* 2020; 36(4): 1053-1066. doi: 10.1109/TRO.2020.2973100

7. Budig M. Robotic High Rises No. 03: Research Projects. *FCL Magazine Special Issue* 2014.

8. Zhou Y, Li X, Yue L, et al. Global Vision-Based Impedance Control for Robotic Wall Polishing. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE; 2019; Macau, China: 6022–6027

9. Asadi E, Li B, Chen IM. Pictobot: A Cooperative Painting Robot for Interior Finishing of Industrial Developments. *IEEE Robot. Automat. Mag.* 2018; 25(2): 82–94. https://ieeexplore.ieee.org/document/8360499/doi: 10.1109/MRA.2018.2816972

10. Yan RJ, Kayacan E, Chen IM, Tiong LK, Wu J. QuicaBot: Quality Inspection and Assessment Robot. *IEEE Transactions on Automation Science and Engineering* 2019; 16(2): 506-517. doi: 10.1109/TASE.2018.2829927

11. Ercan Jenny S, Lloret-Fritschi E, Jenny D, et al. Robotic Plaster Spraying: Crafting Surfaces with Adaptive Thin-Layer Printing. *3D Printing and Additive Manufacturing* 2022; 9(3): 177-188. doi: 10.1089/3dp.2020.0355

12. Mitterberger D, Ercan Jenny S, Vasey L, et al. Interactive Robotic Plastering: Augmented Interactive Design and Fabrication for On-Site Robotic Plastering. In: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems. Association for Computing Machinery; 2022; New York, NY, USA. https://doi.org/10.1145/3491102.3501842

13. Ercan Jenny S, Lloret-Fritschi E, Gramazio F, Kohler M. Crafting plaster through continuous mobile robotic fabrication on-site. *Construction Robotics* 2020; 4: 261–271.

14. Bulgakov A, Emelianov S. Robotic system for plaster and finishing works on the construction site. In: 2016 Future Technologies Conference (FTC). IEEE. ; 2016: 885–889.

15. Bock T, Buzalo N, Bulgakov A. Mathematical Description and Optimization of Robot Control for Plastering Works. In: 2018 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon). ; 2018: 1-5

16. OKIBO . SMART ROBOTICS FOR CONSTRUCTION SITES. 2021. https://okibo.com/.

17. Bard J, Bidgoli A, Chi WW. Image Classification for Robotic Plastering with Convolutional Neural Network. In: Willmann J, Block P, Hutter M, Byrne K, Schork T., eds. *Robotic Fabrication in Architecture, Art and Design 2018*Cham: Springer International Publishing. 2019 (pp. 3–15)

18. Polic M, Maric B, Orsag M. Soft robotics approach to autonomous plastering. In: 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE). IEEE. ; 2021: 482–487.

19. Li X, Jiang X. Development of a Robot System for Applying Putty on Plastered Walls. In: 2018 IEEE International Conference on Mechatronics and Automation (ICMA). IEEE; 2018; Changchun: 1417–1422

20. FoshanBreakThrough MMC. Breakthrough plastering machine. 2021. https://cn.tupomachinery.com/.

21. Ruiz-Agudo E, Rodriguez-Navarro C. Microstructure and Rheology of Lime Putty. *Langmuir* 2010; 26(6): 3868-3877. PMID: 19916534doi: 10.1021/la903430z

22. Liu Z, Cao Y, Wang Y, Wang W. Computer vision-based concrete crack detection using U-net fully convolutional networks. *Automation in Construction* 2019; 104: 129–139. doi: 10.1016/j.autcon.2019.04.005

23. Elbehiery H, Hefnawy A, Elewa M. Surface Defects Detection for Ceramic Tiles UsingImage Processing and Morphological Techniques. *International Journal of Computer and Information Engineering* 2007; 1(5): 1488 - 1492.

24. Zhang Z, Wen G, Chen S. Weld image deep learning-based on-line defects detection using convolutional neural networks for Al alloy in robotic arc welding. *Journal of Manufacturing Processes* 2019; 45: 208–216. doi: 10.1016/j.jmapro.2019.06.023

25. Jiang X, Li X. Robotized interior finishing operations with visual feedback. *Industrial Robot: the international journal of robotics research and application*; 49(1): 141–149. Publisher: Emerald Publishing Limiteddoi: 10.1108/IR-02-2021-0034

26. Kakinuma Y, Ogawa S, Koto K. Robot polishing control with an active end effector based on macro-micro mechanism and the extended Preston's law. *CIRP Annals*; 71(1): 341–344. doi: https://doi.org/10.1016/j.cirp.2022.04.074

27. Chen D, Li K, Jiang X, Liu Y. Development of Macro-Micro Robot with a Compliant End Effector for Putty Applying. In: 2020 IEEE International Conference on Real-time Computing and Robotics (RCAR). ; 2020: 193-198

28. Thrun S, Beetz M, Bennewitz M, et al. Probabilistic Algorithms and the Interactive Museum Tour-Guide Robot Minerva. *The International Journal of Robotics Research* 2000; 19(11): 972-999. doi: 10.1177/02783640022067922

29. Gavin HP. The Levenberg-Marquardt algorithm for nonlinear least squares curve-fitting problems. *Department of civil and environmental engineering, Duke University* 2019; 19.

30. Liu Z, Chen D, Jiang X, Liu Y. Putty Plastering Realized by a Force Controlled Robotic Scraper. In: 2021 IEEE International Conference on Robotics and Biomimetics (ROBIO). IEEE. ; 2021: 1034-1039. doi:10.1109/ROBIO54168.2021.9739274

31. Alsakka F, Assaf S, El-Chami I, Al-Hussein M. Computer vision applications in offsite construction. *Automation in Construction* 2023; 154: 104980.

32. Xiang L, Gai J, Bao Y, Yu J, Schnable PS, Tang L. Field-based robotic leaf angle detection and characterization of maize plants using stereo vision and deep convolutional neural networks. *Journal of Field Robotics* 2023; 40(5): 1034-1053. doi: https://doi.org/10.1002/rob.22166

33. Ganesan G, Chinnappan J. Hybridization of ResNet with YOLO classifier for automated paddy leaf disease recognition: An optimized model. *Journal of Field Robotics* 2022; 39(7): 1085-1109. doi: https://doi.org/10.1002/rob.22089

34. Idjaton K, Janvier R, Balawi M, Desquesnes X, Brunetaud X, Treuillet S. Detection of limestone spalling in 3D survey images using deep learning. *Automation in Construction* 2023; 152: 104919.

35. Speth S, Gonçalves A, Rigault B, et al. Deep learning with RGB and thermal images onboard a drone for monitoring operations. *Journal of Field Robotics* 2022; 39(6): 840-868. doi: https://doi.org/10.1002/rob.22082

36. Minaee S, Boykov Y, Porikli F, Plaza A, Kehtarnavaz N, Terzopoulos D. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence* 2021; 44(7): 3523–3542.

37. Li L, Zhou T, Wang W, Li J, Yang Y. Deep Hierarchical Semantic Segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). ; 2022: 1246-1257.

38. Gerek IH, Erdis E, Mistikoglu G, Usmen MA. Evaluation of plastering crew performance in building projects using data envelopment analysis. *Technological and Economic Development of Economy* 2016; 22(6): 926–940.

39. salary.com . Lather and Plasterer Salary in China. 2021. https://www.salary.com/research/cn-salary/alternate/lather-and-plasterer-salary/cn.