scientific reports

Check for updates

OPEN

Control technology research of building wall spraying robot

Wang Xingyu[1]✉ & Xue Zining[1,2]

Construction robots, as a disruptive innovation in the construction industry, can effectively resolve industry pain points such as low construction efficiency, high safety risks, and high labor costs. For spraying robots of building exterior wall, the adoption of Human-Computer Interaction (HCI) control technology can effectively improve spraying quality, save raw materials, reduce construction costs, and enhance construction efficiency while ensuring personnel safety. The authors developed a prototype of a spraying robot based on HCI technology and proposed a real-time control method based on dynamic gesture recognition. This method focuses on the distance changes of 21 joints on the hand, combines the Support Vector Machine (SVM) classification algorithm, and constructs a real-time closed-loop control system including gesture acquisition, algorithm analysis, and robot execution. By capturing the movement trajectory of the right index finger and recognizing gestures of the left hand, the maximum error of turning head is only 0.89°, and the maximum error of the telescopic rod is only 0.21 mm. Through comparative experiments with automatic trajectory operation mode and HCI control mode at construction site, the HCI control mode can still maintain good spraying effects under different wind speed interference conditions, ensuring the continuity and accuracy of spraying quality.

Keywords Spraying robot, Construction, Gesture recognition, Support vector machine, Human-computer interaction

Applications of robots in the construction industry

With the rapid development of artificial intelligence technology, the construction industry is undergoing a disruptive transformation, robotics technology is entering the construction industry and gradually replacing traditional manual labor methods[1–3]. The innovative application of construction robots has addressed prominent issues in traditional manual construction, such as long project cycles, low construction efficiency, high labor costs, and significant safety hazards[4–7]. Applications of integrated robots and automation devices for mortar spraying and leveling have significantly improved construction efficiency. While maintaining high construction efficiency, the construction quality also meets standard requirements[8–10].

Through standardized construction processes and round-the-clock operation, robots can significantly improve construction speed while reducing raw material waste. A shopping mall construction project is an example, manual bricklaying needs 30 workers work continuously for 15 days, with labor costs exceeding $ 70,000. In contrast, robot construction can shorten the duration to 5 days and save more than 30% in costs.

Spraying robot prototype

As a type of construction robot, exterior wall spraying robots have become a research hotspot in related fields in recent years. LIU Yajun[11] studied intelligent spraying robot technology applied in various fields such as aircraft, automobile and building. He proposed a technical framework based on cloud-based intelligent spraying, and pointed out that control algorithms remain the main technical obstacle for spraying robots. Liangxi Xie[12] developed a spraying robot with capacity of autonomous movement and environment perception, the robot integrates various sensors and algorithms, enabling real-time perception of wall shape. Some Chinese enterprises, such as Bright Dream Robotics Company, have already started developing exterior wall spraying robots[13].

However, facing different architectural structures, the robot needs to be deployed to the construction site 2–3 months in advance, and can only be practically applied after multiple rounds of debugging. Moreover, under sudden weather changes such as strong wind, rain, and snow, it is difficult to guarantee the construction quality, especially in concave-convex areas like balconies and window frames, where it is hard to control the spraying quality. This requires the introduction of machine vision control technology. However, facing complex on-site environments and diverse architectural forms, the intelligent control technology relying solely on machine vision learning needs further exploration. In addition, since image processing algorithms are very sensitive to environmental noise, especially prone to misidentification in areas of light and shadow transitions, fully

---

[1]School of Mechanical and Vehicle Engineering, Jilin Engineering Normal University, Changchun, China. [2]Xue Zining author contributed equally to this work. ✉email: maklcewxy@163.com

intelligent control methods cannot handle emergencies. Therefore, HCI control is needed to ensure construction quality and improve construction efficiency through collaborative work between humans and robots.

Therefore, the authors have developed a prototype of a spraying robot for HCI control in complex environments, as shown in Fig. 1. The key is a multi-degree-of-freedom motion mechanism controlled by HCI technology. By controlling the movement of this mechanism, the spraying quality can be effectively improved. Currently, the invention patent "A Remote-Controlled Building Exterior Spraying Equipment" has been authorized in China, and this prototype has been applied to construction sites. This spraying robot can not only perform HCI control but also achieve traditional automatic control operation, with two control modes available for conducting comparative experiments.
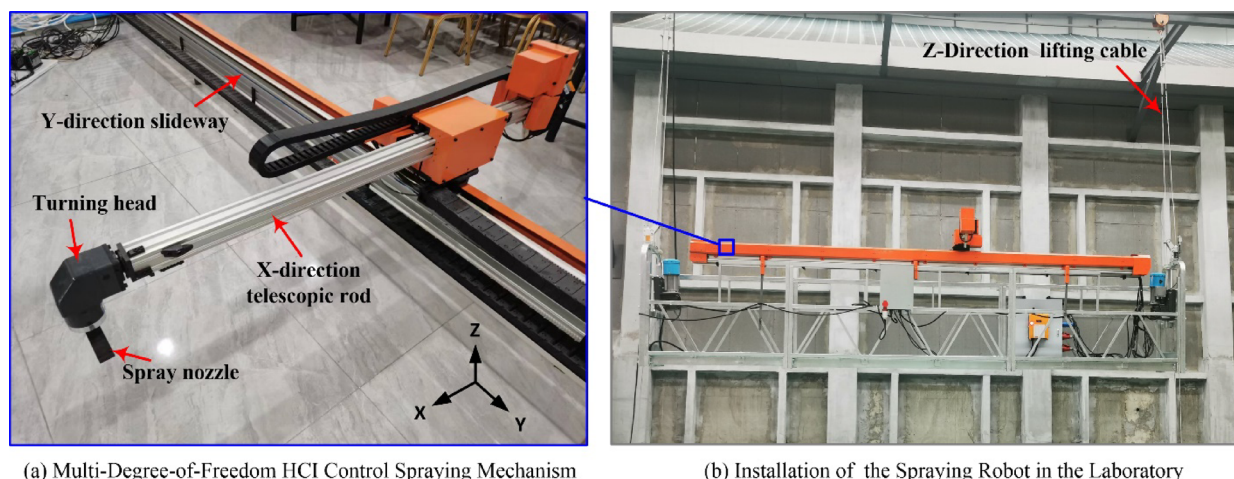
HCI control technology

With the continuous development of machine vision technology and deep learning technology, HCI technology has gradually become an important control method. In the past decade or so, gesture recognition based on computer vision has dominated the field of HCI. Generally, gesture images are captured by CCD (Charge-Coupled Device) and transmitted to the computer, and then gesture recognition is achieved through computer vision technology[14–17].

Common gesture recognition based HCI systems can be mainly divided into three categories:

(1) Sensor-based gesture interaction primarily relies on wearable devices. It features high sensitivity, good real-time performance, and strong immersion, but its high cost limits its widespread application. For example, G.Yuan[18] proposed a deep feature fusion network based on wearable sensors, aiming to improve the accuracy and real-time performance of gesture recognition by fusing multiple sensor data, and provided technical scheme for the development of intelligent interactive devices.

Rajesh K. M[19] designed a spray painting robot for indoor wall coating tasks. The robot uses ultrasonic sensors to detect the distance between the nozzle and the wall, and coordinates with a cascaded lifting mechanism to achieve effective control of the spraying results.

(2) Signal-based gesture interaction mainly depends on surface electromyography signals. It can accurately analyze gestures, but its signal acquisition and connection processes are complex. Guozhen Li[20] proposed a quadruple tactile sensor and applied it to a robotic hand. This sensor integrates the sensing capabilities of pressure, material thermal conductivity, object temperature, and ambient temperature, enabling robots to identify objects of different shapes, sizes, and materials. E. Rahimian[21] proposed a new architecture (named as FHGR which refers to "Few-shot Hand Gesture Recognition") that learns the mapping using a small number of data and quickly adapts to a new user/gesture by combing its prior experience. The proposed approach led to 83.99% classification accuracy on new repetitions with few-shot observations.

(3) Vision-based gesture interaction methods, such as those image sequence-based systems, are gaining increasing development and application due to simple hardware requirements and diverse algorithms. However, this method is limited by issues like lighting and occlusion. Typically, human hands can move at speeds exceeding 5 m per second in unconscious movements, and different gestures convey different meanings. Therefore, the accuracy and robustness of dynamic gesture recognition remain challenging research topics. Some brand-new interactive experiences have been realized with the innovative application of brain-computer interface, gesture, eye movement, electromyography, audio and other signals in artificial intelligence. P. S. Neethu[22] proposed a gesture detection and recognition method based on evolutionary neural network classification, and made improvements to enhance detection accuracy and recognition effect.

(a) Multi-Degree-of-Freedom HCI Control Spraying Mechanism   (b) Installation of the Spraying Robot in the Laboratory

Fig. 1. Spraying robot prototype based on HCI technology.

In addition to the aforementioned three control methods, some HCI control methods based on complex background information have also emerged successively. Onososen A.O[23] employed a four-stage review method to study 112 articles, aiming to assess the nature of research methods in construction robotics. By using bibliometric and systematic analysis methods, the study revealed the methodological diversity in construction robotics research and discussed the influencing factors between policy and stakeholders in construction robotics. Haitao, Wu[24] proposed a gesture recognition method based on thermal imaging, which can effectively recognize gestures under complex lighting conditions, thereby improving the robustness and accuracy of gesture recognition. Pan, M[25] proposed a framework for developing construction robots through an integrated multi-scenario approach, which aims to optimize the design and application of construction robots and improve their adaptability and efficiency in complex construction environments. Halder, S[26] provided a systematic review of using robots for building health inspection and monitoring, analyzed 269 relevant papers, discussed the current application status and technological advancements in this field. The review covers key technologies including autonomous navigation, knowledge extraction, motion control, and multi-robot collaboration. Minghui Wu[27] established both high-fidelity and low-fidelity models to simulate and evaluate the impact of human–robot interaction on construction productivity. Although the relationship between robot quantity and productivity was not linear, the analysis showed that human–robot interaction could significantly improve productivity, with a maximum increase of 22% in experiments. Dongmin Lee [28] has developed a digital twin-driven DRL machine learning method for construction robots by focusing on the dynamic changes of the construction environment. Through three dynamic tests, it has been proven that the construction time can be reduced by 36% using the DRL method, which is of great significance for construction automation. Lidia Atanasova[29] proposed a cooperative control strategy for "human–robot" collaboration mode in assembly processes. By integrating sensing and real-time motion planning, robots can perform autonomous contact-based complex assembly tasks in uncertain environments. This collaborative approach allows robots to guide motion through perception while enhancing perception capabilities through motion.

In recent years, building robots for special purposes have also been developed and applied. Yongding Tian[30] reviewed various mobile robot technologies applied in the construction industry for health monitoring of structures such as bridges and cables. Through robot vision inspection and modal recognition technology, these robots can detect defects and identify anomalies in bridges. The paper also analyzed the advantages and disadvantages of wall-climbing robots, cable-climbing robots, and flying drones in inspection tasks. Thanh Phuong Nguyen[31] developed a multi-wheeled robot for pipeline internal inspection and analyzed flexible motion computation. Through simulation verification, laboratory testing, and construction site validation, the feasibility of the multi-axis robot was proven, representing an advanced technological innovation in the construction industry.

Use hand joints to recognize gestures

At present, most static gesture recognition technologies are based on distance comparison methods, mainly with statistical analysis techniques, which requires deep learning of abundant gesture samples. For dynamic gesture recognition, it is necessary to track changes in hand contour information and map it to parameter space. Due to the huge computational requirements, its robustness and real-time performance often fall short of expectations. For example, different users performing the same gesture may yield significantly different recognition results, and even the same user performing the same gesture twice may produce vastly different results. In recent years, dynamic gesture recognition technology based on hand joints has gradually become a research hotspot. Since hand joints are a high-level abstract representation of gesture information, hand joints are simplified as 21 joint points, or 21 coordinate points. By characterizing gestures through changes in the spatial relationships of these coordinate points, recognition results can be made more accurate and the problem of low recognition rates caused by complex backgrounds can be overcome.

Considering the above reasons, the HCI control technology proposed in this paper is also based on the recognition of 21 hand joints, thereby enabling flexible and precise control of the spraying robot's movements.

This research was jointly completed by Wang Xingyu and Xue Zining. Wang Xingyu proposed the research topic and designed the research plan. He was responsible for hardware installation, algorithm development, programming, and led the data collection and organization. He also drafted the initial manuscript and made subsequent revisions. Xue Zining provided crucial technical support during the experiment, ensuring its smooth progress. He was responsible for data statistical analysis, chart production, and collaborated with Wang Xingyu on content revisions. Throughout the research process, the two authors worked closely together to advance the research and ensure the completion of the paper.

Overall technical solution

By detecting the position changes of 21 joints in the hand, the computer recognizes the meaning of gestures and sends control signals to drive movement of the spraying robot. The overall technical solution is shown in Fig. 2. The operator sends control commands to the spraying robot through a dynamic gesture capture camera. The robot performs various actions based on the gesture recognition results. The spraying quality is fed back to the operator through a monitoring camera, enabling real-time monitoring of the spraying process.

The meanings of the hand gestures are summarized as follows:

(1) When the left hand extends the index finger, and the right hand rotates the index finger, the spray nozzle around the W-axis rotates.
(2) When the left hand extends the index and middle fingers, and the right hand rotates the index finger, the turning head around the P-axis rotates.

(a) Direction of Movement   (b) Control Method Diagram   (c) Hand Gestures and Control Meanings
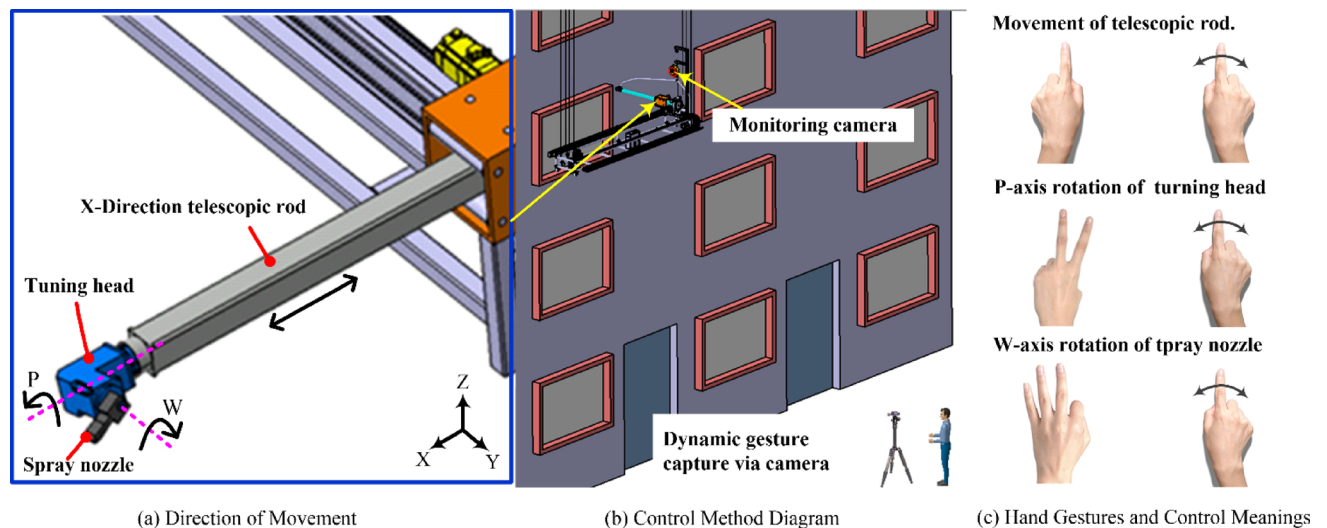
Fig. 2. Overall technical solution.

(3) When the left hand extends three fingers, and the right hand moves the index finger, the telescopic rod along the X-axis moves.

In the HCI control process, the left-hand gesture is first recognized to determine which part of the robot will be controlled (for example, spray nozzle, turning head, or telescopic rod). Then, moving the right index finger, the corresponding part can be driven. Flexibility and usability in spraying operations can be achieved by combination different method. Even when the operator is far from the spraying robot, the spraying task can still be well completed. For sliding in the Y direction and vertical movement in the Z direction, the robot's built-in automation program can be used without HCI. This is because these two movements are for adjusting the overall robot position and cannot precisely control the spray nozzle and turning head for spraying effect.

Hand gesture recognition

Simplifying the palm into 21 joints essentially transforms the palm into a point cloud model. The common practice is to distribute each joint into a spatial coordinate system, and use combinatorial mathematics theory and social network graph theory for recognition[32]. When the 21 coordinate points change frequently, the aggregation pattern of these points becomes extremely complex, significantly increasing the computational load for dynamic gesture recognition. To avoid traversing each coordinate point, this paper proposes a fast algorithm that reduces computational complexity while ensuring gesture recognition.

Hand joint recognition method

The hand key points recognition is to locate the positions and postures of hand joints. By accurately identifying these key points, gesture changes can be located and tracked. Common methods include graph model-based methods, template matching-based methods, and feature point detection-based methods. In recent years, with the optimization and upgrading of the OpenCV library, the accuracy and efficiency of deep learning-based hand joint recognition functions have significantly improved, with the average recognition time for hand joints less than 1 ms, providing technical support for HCI control. This paper uses the "MediaPipe.hands" gesture detection module in OpenCV, an open-source function model provided by Google Company, which has been well trained and can recognize finger posture changes in real-time with good robustness.

"MediaPipe.hands" gesture detection model numbers each joint, as shown in Fig. 3 below. Point 0 represents the center of the wrist. When each finger changes its movement, the gesture can be preliminarily classified through the set of joints. Therefore, we need to use a classifier algorithm to implement this function.

Classifier recognition algorithm

Currently, the main classifier recognition algorithms include logistic regression, naive bayes, K-nearest neighbors, decision trees, and support vector machines (SVM). The classification features of discrete points of hand joints on a two-dimensional coordinate plane are two-dimensional linear classification, and SVM is particularly suitable for binary linear classification of discrete targets.
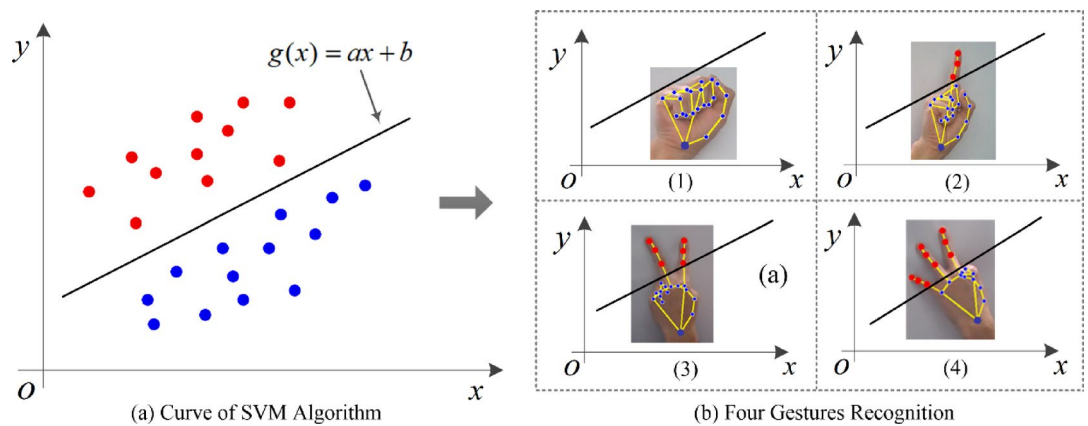
SVM algorithm is also applicable to machine learning problems in small sample scenarios. It can simplify common problems such as classification and regression. Specifically, when discrete points are linearly separable, the algorithm can find the optimal classification hyperplane for two types of samples in the original space. The classifier function $g(x)$ is expressed as follows.

$$g(x) = ax + b \qquad (1)$$

(a) Hand Joint Index   (b) Joint Names Corresponding to Each Number
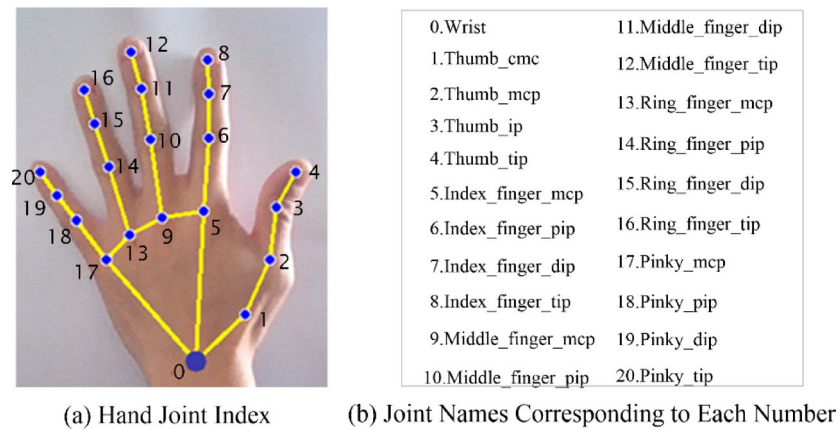
Fig. 3. Recognition of hand joints.

(a) Curve of SVM Algorithm   (b) Four Gestures Recognition

Fig. 4. Gesture recognition through classifier.

In Eq. (1), a is the slope, b is the intercept.

For typical binary discrete points, if function $g_k(x) = a_k x + b_k (k = 0, 1, 2, 3)$ is the diagonal line in Fig. 4, red and blue points will be strictly classified according to this diagonal line. Based on the same principle, when making a fist, points below the diagonal line are classified into one group, and there are no discrete points above the diagonal line, as shown in Fig. 4(1). When the index finger is extended, this function distributes three points of the index finger (red) above the diagonal line, and the remaining points (blue) below the diagonal line, as shown in Fig. 4(2). According to the same principle, we can distinguish between extending two fingers and extending three fingers, as shown in Fig. 4(3) and Fig. 4(4).

As the hand moves and gestures change, the coordinates of the 21 joints on each hand are in a dynamic process of change. Therefore, dynamically determining a and b is another key, so that function $g(x) = ax + b$ becomes available.

According to the principle of least squares, the smaller the deviation between samples and expectations is, the better, that is, to minimize the weighted sum of squares of the deviation between observed values and expected values. For linear fitting of equally precise observed values like fingertip points in this article, we can minimize the following equation.

$$\sum_{i=1}^{N} [y_i - (ax_i + b)]^2 \tag{2}$$

Taking partial derivatives of a and b in Eq. (2) above, we obtain the following equations.

$$\frac{\partial}{\partial a} \sum_{i=1}^{N} [y_i - (a + bx_i)]^2 = -2 \sum_{i=1}^{N} (y_i - a - bx_i) = 0 \tag{3}$$

$$\frac{\partial}{\partial b} \sum_{i=1}^{N} [y_i - (a + bx_i)]^2 = -2 \sum_{i=1}^{N} [y_i - (a + bx_i)]x_i = 0 \tag{4}$$

After calibration, the following system of equations can be obtained.

$$\begin{cases} aN + b \sum_{i=1}^{N} x_i = \sum_{i=1}^{N} y_i \\ a \sum_{i=1}^{N} x_i + b \sum_{i=1}^{N} x_i^2 = \sum_{i=1}^{N} x_i y_i \end{cases} \tag{5}$$

After solving the above system of equations, the optimal estimates of a and b can be obtained in $g(x)$, with the solution formula as follows.

$$\hat{a} = \frac{\left(\sum_{i=1}^{N} x_i^2 \right) \left(\sum_{i=1}^{N} y_i \right) - \left(\sum_{i=1}^{N} x_i \right) \left(\sum_{i=1}^{N} x_i y_i \right)}{m \left(\sum_{i=1}^{N} x_i^2 \right) - \left(\sum_{i=1}^{N} x_i \right)^2} \tag{6}$$

$$\hat{b} = \frac{m \left(\sum_{i=1}^{N} x_i y_i \right) - \left(\sum_{i=1}^{N} x_i \right) \left(\sum_{i=1}^{N} y_i \right)}{m \left(\sum_{i=1}^{N} x_i^2 \right) - \left(\sum_{i=1}^{N} x_i \right)^2} \tag{7}$$

According to the above equation, to determine whether $\hat{a}$ and $\hat{b}$ have a linear relationship with x_i and y_i, correlation analysis needs to be performed, and the correlation coefficient is set as r. The expression for correlation calculation is as follows:

$$r = \frac{\sum_{i=1}^{N} (x_i - \overline{x}) \sum_{i=1}^{N} (y_i - \overline{y})}{\sqrt{\sum_{i=1}^{N} (x_i - \overline{x})^2} \sqrt{\sum_{i=1}^{N} (y_i - \overline{y})^2}} \tag{8}$$

In Eq. (8), $\overline{x} = \sum_{i=1}^{N} x_i/m, \overline{y} = \sum_{i=1}^{N} y_i/m, r \in [-1, 1]$, when $|r| \to 1$, an optimal linear relationship is observed between a and b.

Left hand gesture recognition

The above classifier algorithm solves the problem of classifying finger joints and other palm joints. To further determine which finger is extended, the paper uses relative distance for judgment. Typically, measuring the similarity between two points is the main task of feature recognition. Common measurement methods include 'Euclidean distance' and 'Mahalanobis distance'. Among them, 'Euclidean distance' is the most direct distance expression, suitable for distance measurement between two points[33]. Since 'Euclidean distance' does not focus on the relationship between each vector and each dimension, and the importance of each dimension component is the same. This is quite consistent with the characteristics of hand joint distance, which makes 'Euclidean distance' transformation widely used in digital image processing, especially suitable for measuring skeletal joints in human body images.

If each m-dimensional eigenvector of i is X_i, and X_{ik} is the k-dimensional parameter of $X_i, |D_{i,j}|$ is the 'Euclidean distance' between individual i and individual j, then $|D_{i,j}|$ can be expressed as:

$$|D_{i,j}| = \sqrt{\sum_{k=1}^{m} (X_{ik} - X_{jk})^2} (i, j = 1, 2, 3...n : \ k = 1, 2, 3..m) \tag{9}$$

As the left hand is used to select and determine the controlled parts with the spraying robot, this paper adopts the distance from the left finger joints to the center point 0 of the wrist to make judgments. The 21 hand joints are projected onto a two-dimensional coordinate plane, and the position changes of a joint are expressed with x-axis and y-axis coordinate values. The left hand's posture can be determined through the following algorithm.

(1) Extend the index finger of the left hand

$$d_i = \sqrt{(p_{ix} - p_{0x})^2 + (p_{iy} - p_{0y})^2} (i = 1, 2 \cdots 20) \tag{10}$$

6

d_i is the distance from any joint on the finger to the wrist point 0.

p_{ix} and p_{iy} are the coordinates of the ith joint on the x-axis and y-axis respectively.

p_{0x} and p_{0y} are the coordinates of the wrist at point 0 on the x-axis and y-axis respectively.

When d_8 is greater than the distance from all other joints to point 0, the left hand is determined to be in the index finger pointing gesture, as shown in Fig. 4(2).

(2)   Extend the index and middle fingers of the left hand

$$d_{j,k} = \sqrt{(p_{jx} - p_{0x})^2 + (p_{jy} - p_{0y})^2} \qquad \begin{matrix}(j = 1, 2 \cdots 20)\\ + \sqrt{(p_{kx} - p_{0x})^2 + (p_{ky} - p_{0y})^2} \qquad (k = 1, 2 \cdots 20)\\ (j \neq k)\end{matrix} \tag{11}$$

p_{jx} and p_{jy} are the coordinates of the jth joint on the x-axis and y-axis respectively.

p_{kx} and p_{ky} are the coordinates of the kth joint on the x-axis and y-axis respectively.

In Eq. (11),when $j = 8, k = 12$ and $d_{8,12}$ is greater than the sum of distances from all other joints to point 0, the left hand is determined to be stretching the index finger and middle finger, as shown in Fig. 4(3).

(3)   Extend three fingers with the left hand

$$d_{l,m,n} = \sqrt{(p_{qx} - p_{0x})^2 + (p_{qy} - p_{0y})^2} \qquad \begin{matrix}(q = 1, 2 \cdots 20)\\ + \sqrt{(p_{mx} - p_{0x})^2 + (p_{my} - p_{0y})^2} \qquad (m = 1, 2 \cdots 20)\\ + \sqrt{(p_{nx} - p_{0x})^2 + (p_{ny} - p_{0y})^2} \qquad (n = 1, 2 \cdots 20)\\ (q \neq m \neq n)\end{matrix} \tag{12}$$

p_{qx} and p_{qy} are the coordinates of the qth joint on the x-axis and y-axis respectively.

p_{mx} and p_{my} are the coordinates of the mth joint on the x-axis and y-axis respectively.

p_{nx} and p_{ny} are the coordinates of the nth joint on the x-axis and y-axis respectively.

In Eq. (12), when $q = 12, m = 16, n = 20$ and $d_{q,m,n}$ is greater than the sum of distances from all other joints to point 0, the left hand with three fingers is identified as shown in Fig. 4(4).

Although random or unconscious gestures may get the same recognition results, for accurate and expected outcomes, operators are required to perform correct gestures as much as possible to achieve optimal recognition results.
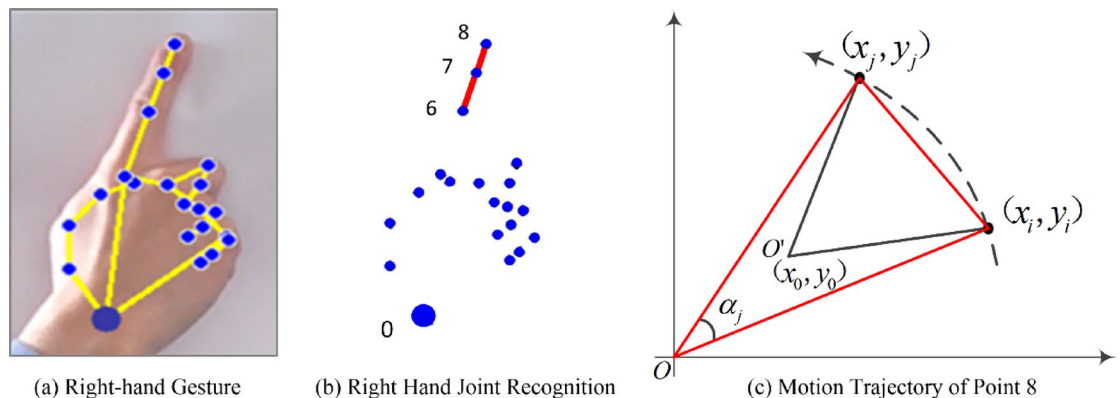
Right-hand movement algorithm

The extraction method for the right index finger joint is similar to that of the left hand. However, since the right hand is needed to drive the robot's movement, we only need to consider the right index finger's tip point 8 separately. By identifying the movement coordinates of point 8, we can obtain the control commands.

If the coordinate of the right index finger's tip point 8 is (x_i, y_i) at the initial moment, as the right index finger moves, the coordinate of point 8 changes to (x_j, y_j). Here, we classify the movement of the right index finger into two motion patterns:

(1)   Rotation of the right index finger.

Assume the oscillation trajectory of point 8 is approximately an arc, as shown by the dashed line in Fig. 5. However, as the finger rotates, the wrist also moves slightly, meaning point O' of the wrist also changes in the XY plane. To solve this problem, we use absolute coordinate point O to calculate the rotation angle of hand point 8, as shown in Eq. (13).

$$\alpha_j = |\arctan(y_j/x_j) - \arctan(y_i/x_i)| \tag{13}$$

(a) Right-hand Gesture   (b) Right Hand Joint Recognition   (c) Motion Trajectory of Point 8

Fig. 5. Schematic diagram of right index finger motion trajectory calculation.

(2) Horizontal movement of right index finger.

When the right index finger moves in translation, the coordinate value (x_j, y_j) continuously updates. We only need to calculate the distance between point (x_i, y_i) and point (x_j, y_j) to determine the distance the controlled object needs to move.

Experimental verification

Control accuracy verification

This paper developed a control program with Python software and conducted HCI control tests. The overall process is shown in Fig. 6 below. Step-1 is used to recognize 21 joints of the hand; Step-2 is used to determine gestures and select the robot parts to be driven; Step-3 is used to move the index finger to send control commands to the robot; Step-4 is the hardware system, mainly including video capture card and multi-degree-of-freedom motion controller. Through these four steps, interactive control of spraying robot can be achieved.

During the interactive control process, the recognition accuracy of the right index finger plays a crucial role in the control precision of the robot. Therefore, this section will detail the control effect of the right index finger movement. Standard curved and straight notched templates are used to drive the robot's turning head and telescopic rod, which helps in comparative analysis of control precision. As shown in Figs. 7 and 8, when the right index finger slides along the curved and straight notched templates, the camera captures the movement trajectory of fingertip point 8, and real-time feedback of the turning head and telescopic rod is obtained through the angle sensor and displacement sensor.

First, the index finger slowly rotates in the curved notched template, and the camera tracks the rotation process of point 8 in real-time, calculating its coordinate values and radius value on the XY plane. Then, it calculates the rotation angle α_j of point 8 at any moment, and outputs α_j to the control system to monitor the actual rotation angle of turning head. During the rotation of point 8 on the right hand from 0° to 180°, the feedback signal of the turning head is sampled at 40 equally spaced points, with each interval between sampling points being 4.5°. The actual angle of the turning head at each sampling point is recorded, and the radius value and rotation angle are plotted in polar coordinates. To verify the reliability of the algorithm, three repeated test processes were conducted, and the obtained actual trajectories of the turning head are shown as Test1, Test2, and Test3 in Fig. 7.

In three tests, the maximum error in the first test was 0.89°. Subsequently, with iterative training of the system on error correction, the accuracy was gradually improved, and the maximum error in the third test was only 0.55°.

Based on the same principle, the index finger slides within the straight notched template. As shown in Fig. 8, the camera recognizes the movement of point 8 and calculates its coordinate value (x_j, y_j) on the XY plane in real-time. The X-axis coordinate value x_j drives the movement of the telescopic rod. Throughout the entire range of index finger movement, the movement of the telescopic rod is recorded. 40 sampling points are selected at equal intervals, and the testing process is repeated three times. The actual movement trajectories of the telescopic rod are shown as Test1, Test2, and Test3 in the Fig. 8. In these three tests, the maximum error was 0.21 mm in the first test. Subsequently, with iterative error correction and training, the accuracy was gradually improved with a maximum error of only 0.14 mm in the third test.

In the above tests, the error of 40 sampling points per curve (including gesture recognition error and hardware error) represents the total system error. However, based on the feedback results, both for curve and straight notched interaction control, the spraying robot's turning head and telescopic rod can respond to gesture

① Inputs-"MediaPipe.hands" Gesture Detection Module in OpenCV
② Outputs-21 Joints in Each Palm
③ Inputs-Left Hand and Right Hand Gesture Recognition Algorithm
④ Outputs-Left Hand Gesture Recognition Result
⑤ Outputs-Right Hand Gesture Recognition Result
⑥ Inputs-Gesture Recognition Results Input to Hardware
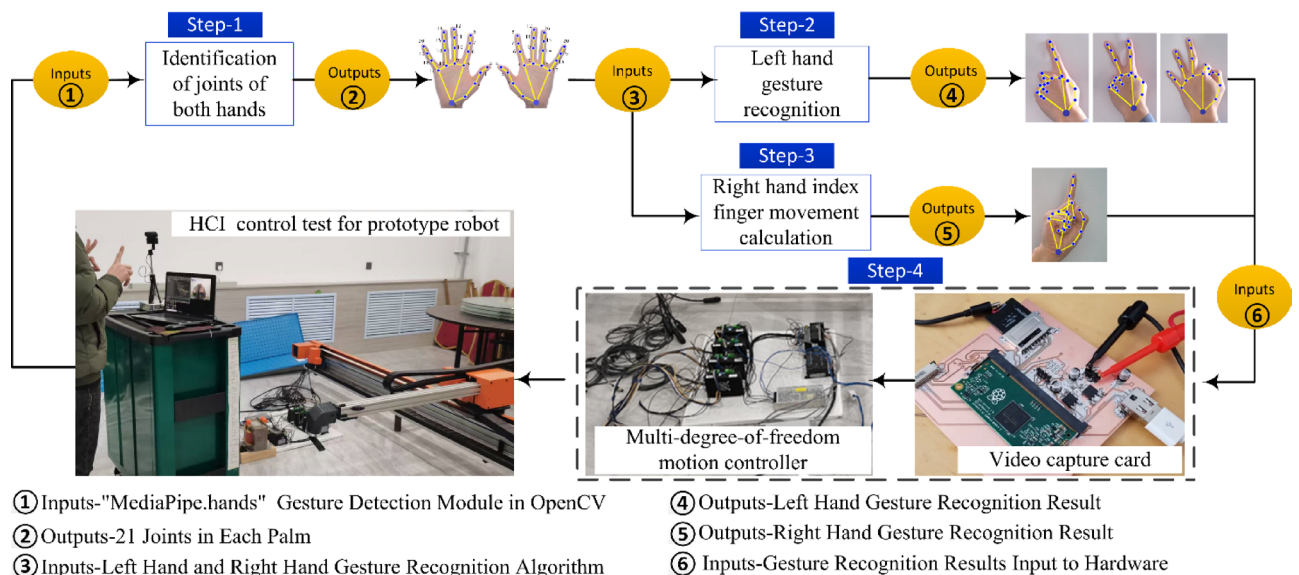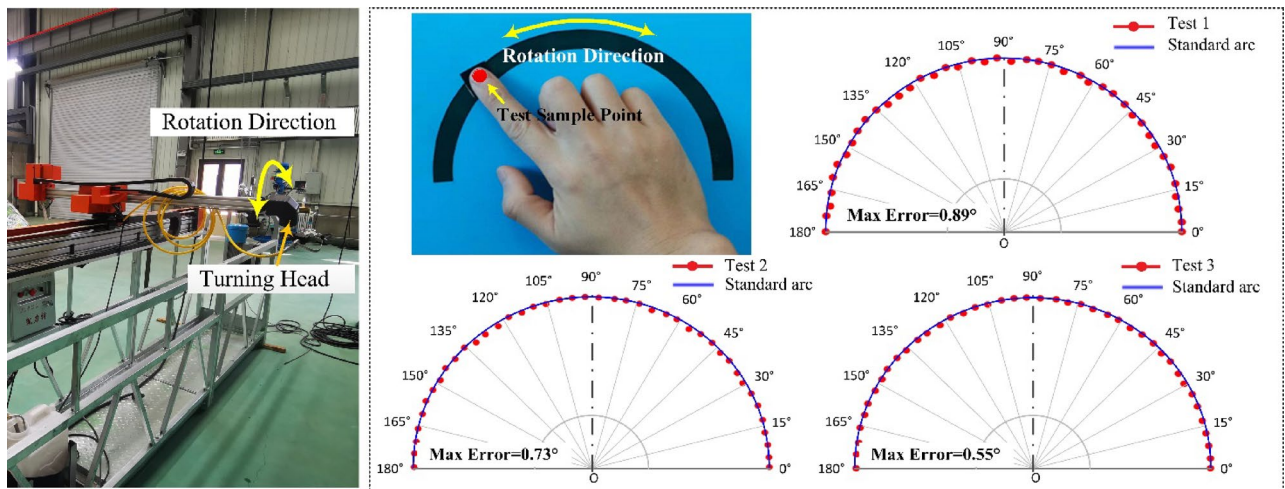
Fig. 6. HCI control process and methods.
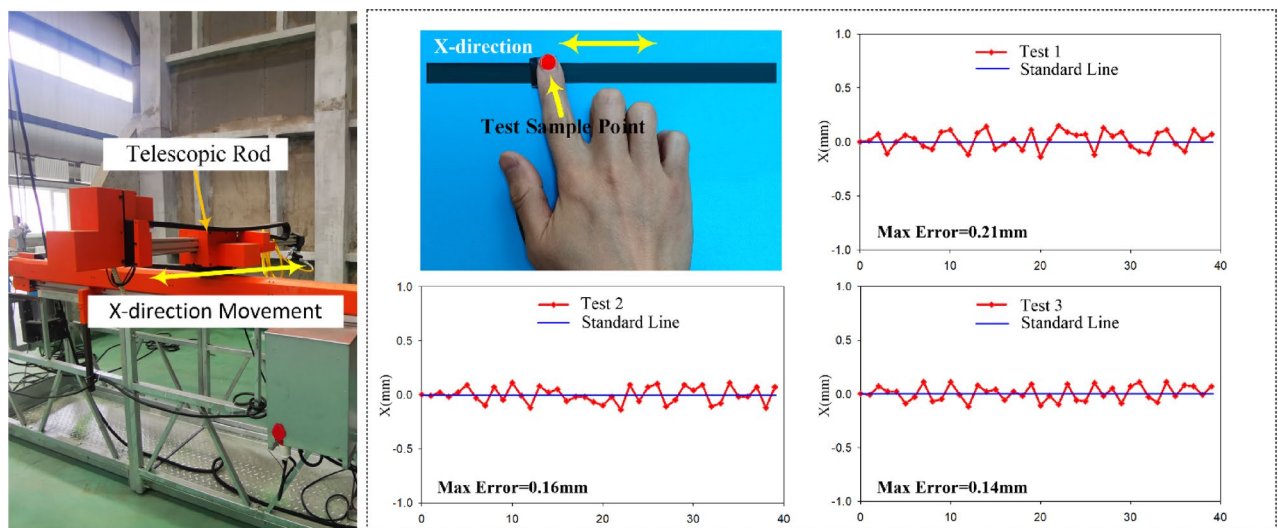
Fig. 7. Test verification of rotation.

**Fig. 8**. Test verification of X-direction movement.

commands, accurately understand gesture meanings, and demonstrate high motion accuracy. Moreover, the control accuracy gradually improves after multiple.

Spraying effect comparative test

To verify the effectiveness of HCI control method, the robot was applied to a construction site, to spray the newly-built residential buildings. The spraying robot can be set to two working modes: automatic operation and HCI control. Before the experiment, an air blower was fixedly installed on the spraying robot, with the air outlet 1 m away from spray nozzle. By changing the speed of the air blower, the impact of different wind speeds on spraying was simulated.

Under normal circumstances, when the wind is weak, the airflow disturbance to spraying is not strong. At this time, the automatic operation mode maintains good and stable construction quality. However, when the wind speed increases, it becomes difficult to accurately control the spraying area. The stronger the wind, the larger the area where the paint film deviates from the target area. After adopting HCI control, operators can monitor the spraying effect in real-time to adjust spray nozzle angle and spraying pressure flexibly, even with strong wind, so spraying quality can be guaranteed.

Four wind speeds are set 3 m/s, 5 m/s, 8 m/s, and 10 m/s. When the wind speed exceeds 10 m/s, it is strong wind, which does not meet construction conditions, so strong wind tests are not considered in this paper. From the comparison in the figures below, when the wind speed is 3 m/s, both spraying modes have high precision, with paint mostly within the target area, as shown in a1 and a2 in Fig. 9. As wind speed increases, the spraying precision of the automatic operation mode is significantly affected, with paint starting to spread to adjacent areas. When wind speed reaches 10 m/s, a large amount of paint splashes into adjacent areas, resulting in poor
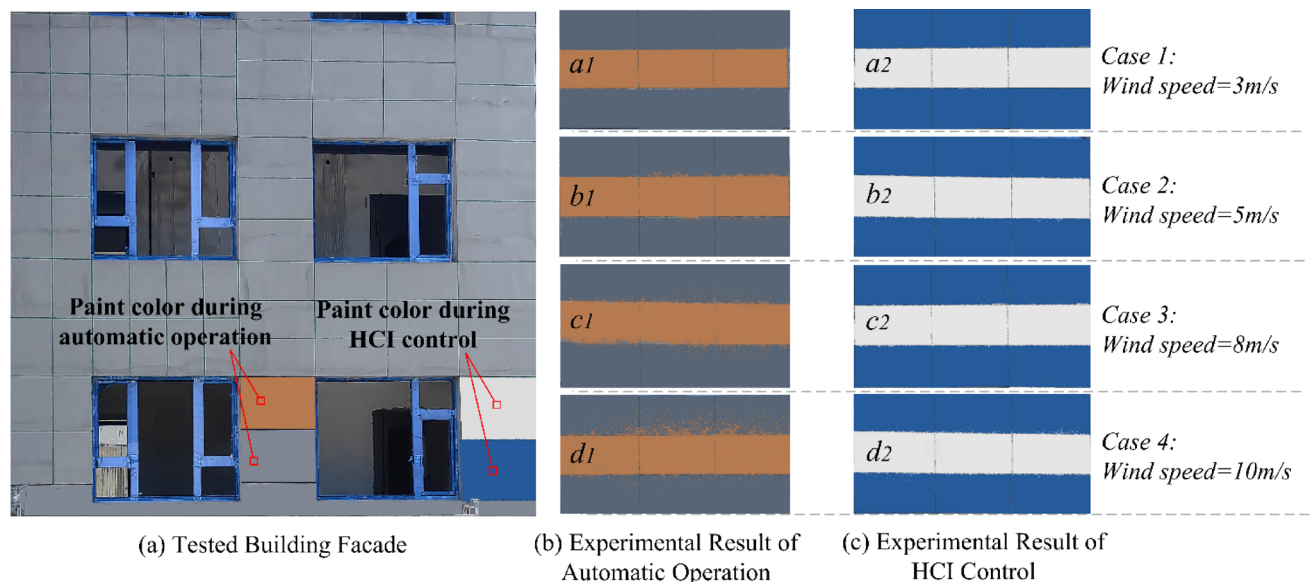
Fig. 9. Comparison of spraying effects between automatic operation and HCI control.

spraying quality, as shown in d1 in Fig. 9. However, under the same wind speed, due to human intervention, the HCI controlled spraying area remains highly accurate and little paint splashes into adjacent areas, as shown in d2 in Fig. 9.

Conclusion

This paper focuses on the brand-new field of construction robots, the authors developed a prototype of a HCI spraying robot and proposed a gesture recognition-based HCI control method. This method combines 'Euclidean distance' algorithm and SVM classification algorithm to realize the robot's recognition of hand movements. The authors conducted motion control accuracy tests and spraying effect comparison tests for the spraying robot.

In the motion accuracy tests, when the right index finger moved in the curved and straight notched template, both the turning head and telescopic rod of the spraying robot maintained synchronous movement. The tests were repeated three times, with the maximum error of the turning head interaction control being only 0.89°, and the maximum error of the telescopic rod interaction control being only 0.21 mm. The errors gradually decreased with increasing training times.

In the spraying effect comparison tests, we conducted practical application on a newly-built building with two control modes: robot automatic trajectory operation and HCI control. To verify the accuracy of the HCI control, an air blower was installed on the robot, and four wind speeds were set. Under airflow interference, the spraying quality of the HCI control method remained high. Even when the wind speed reached 10 m/s, the spraying effect was still very ideal through HCI control intervention. However, the spraying of automatic trajectory operation was poor, and lots of paint splashed around the target area. Therefore, raw material was wasted and construction efficiency was decreased.

It should be noted that although ideal test results were obtained with small computation requirements and high execution efficiency, these results were obtained through continuous debugging and repeated exploration. The author will continue to increase the number of test samples and repeatedly train the HCI control algorithm to further improve gesture recognition efficiency and spraying accuracy.

Data availability

All experimental data are presented in the paper. If needed, the source data and figures supporting the results of this study can be obtained from the corresponding author, maklcewxy@163.com.

References

1. Dörfer, K. et al. Additive manufacturing using mobile robots: Opportunities and challenges for building construction. *Cem. Concr. Res.* **158**, 106772. https://doi.org/10.1016/j.cemconres.2022.106772 (2022).
2. Zhao, S. et al. Application and development of autonomous robots in concrete construction: Challenges and opportunities. *Drones.* **6**(12), 424. https://doi.org/10.3390/drones6120424 (2022).
3. Bademosi, F. M. & Issa, R. R. A. Automation and robotics technologies deployment trends in construction. In *Automation and robotics in the architecture, engineering, and construction industry* (eds Jebelli, H. et al.) (Springer, Cham, 2022).
4. Melenbrink, N., Werfel, J. & Menges, A. On-site autonomous construction robots towards unsupervised building. *Autom. Constr.* **119**, 103312. https://doi.org/10.1016/j.autcon.2020.103312 (2020).

5. Pan, Y. & Zhang, L. Roles of artificial intelligence in construction engineering and management: A critical review and future trends. *Autom. Constr.* **122**, 103517. https://doi.org/10.1016/j.autcon.2020.103517 (2021).
6. Yuming, L. et al. Robotics in the construction sector: Trends, advances, and challenges. *J. Intell. Robot. Syst.* **110**, 72. https://doi.org/10.1007/s10846-024-02104-4 (2024).
7. Brosque, C. & Fischer, M. Safety, quality, schedule, and cost impacts of ten construction robots. *Constr. Robot.* **6**(2), 163–186. https://doi.org/10.1007/s41693-022-00072-5 (2022).
8. Wang, Y. et al. Mortar spraying and plastering integrated robot for wall construction. *Autom. Constr.* **139**, 105533. https://doi.org/10.1016/j.autcon.2024.105533 (2024).
9. Nyemba, W. R. et al. Conceptualization, development and design of a mortar spraying machine. *Sci. Direct.* **91**(2020), 396–401. https://doi.org/10.1016/j.procir.2020.03.105 (2020).
10. McAlorum, J. et al. Robotic spray coating of self-sensing metakaolin geopolymer for concrete monitoring. *Autom. Constr.* **121**, 103415. https://doi.org/10.1016/j.autcon.2020.103415 (2021).
11. Yajun, L. I. U. et al. Research progress and trend of key technology of intelligent spraying robot. *J. Mech. Eng.* **58**(7), 53–74. https://doi.org/10.3901/JME.2022.07.053 (2022).
12. Xie, L. et al. Intelligent spraying robot for building walls with mobility and perception. *Autom. Constr.* **139**, 104270. https://doi.org/10.1016/j.autcon.2022.104270 (2022).
13. Products of Bright Dream Robot. *Bright Dream Robotics Company.* https://www.bzlrobot.com/contents/3/77.html (2025).
14. Michalík, R. et al. Human-Robot motion control application with artificial intelligence for a cooperating yumi robot. *Electronics* **10**(16), 1976. https://doi.org/10.3390/electronics10161976 (2021).
15. Kloetzer, M. & Mahulea, C. Path planning for robotic teams based on LTL specifications and petri net models. *Discret. Event Dyn. Syst.* **30**(1), 55–79 (2020).
16. Hui, L. et al. Gesture control technology of CAD model based on situation model. *J. Mech. Eng.* **58**(10), 374–382. https://doi.org/10.3901/JME.2022.10.374 (2022).
17. Wang, R. P. et al. Pedestrian trajectory prediction method based on pedestrian pose. *J. Beijing Univ. Aeronaut. Astronaut.* **49**(7), 1743–1754. https://doi.org/10.13700/j.bh.1001-5965.2021.0557 (2023).
18. G.Yuan. et al., Hand gesture recognition using deep feature fusion network based on wearable sensors. *IEEE Sensors Journal.* **21**(1), 539–547. https://doi.org/10.1109/JSEN. 2020.3014276(2020).
19. Rajesh, K. M, Vineeth. P. D and Chaitanya S.K.N. Autonomous Wall Painting Robot. *IEEE International Conference for Emerging Technology.* https://doi.org/10.1109/INCET49848.2020.9154020 (2020).
20. Li, G. et al. Skin-inspired quadruple tactile sensors integrated on a robot hand enable object recognition. *Sci. Robot.* https://doi.org/10.1126/scirobotics.abc8134 (2020).
21. Rahimian, E. et al. FS-HGR: Few-shot learning for hand gesture recognition via electromyography. *IEEE Trans. Neural Syst. Rehabilit. Eng.* **29**, 1004–1015 (2021).
22. Neethu, P. S., Suguna, R. & Sathish, D. An efficient method for human hand gesture detection and recognition using deep learning convolutional neural networks. *Soft Comput.* **24**, 15239–15248 (2020).
23. Onososen, A. O., Musonda, I. & Ramabodu, M. Construction robotics and human-robot teams research methods. *Buildings* **12**(8), 1192 (2020).
24. Haitao, Wu. et al. Thermal image-based hand gesture recognition for worker-robot collaboration in the construction industry: A feasible study. *Adv. Eng. Inform.* **56**, 101939. https://doi.org/10.1016/j.aei.2023.101939 (2023).
25. Pan, M. et al. Structuring the context for construction robot development through integrated scenario approach. *Autom. Constr.* https://doi.org/10.1016/j.autcon.2020.103174 (2020).
26. Halder, S. & Afsari, K. Robots in inspection and monitoring of buildings and infrastructure: A systematic review. *Recent Adv. Mech. Robot. Syst.* **13**, 2304. https://doi.org/10.3390/app13042304 (2023).
27. Minghui, W., Lin, J. R. & Zhang, X. H. How human-robot collaboration impacts construction productivity An agent-based multi-fidelity modeling approach. *Adv. Eng. Inform.* **52**, 101589–101602. https://doi.org/10.1016/j.aei.2022.101589 (2022).
28. Lee, D. et al. Digital twin-driven deep reinforcement learning for adaptive task allocation in robotic construction. *Adv. Eng. Inform.* **53**, 101710. https://doi.org/10.1016/j.aei.2022.101710 (2022).
29. Lidia, A. et al. Building (with) human–robot teams: fabrication-aware design, planning, and coordination of cooperative assembly processes. *Constr. Robot.* https://doi.org/10.1007/s41693-025-00158-w (2025).
30. Tian, Y. et al. Intelligent robotic systems for structural health monitoring applications and future trends. *Autom. Constr.* **139**, 104273. https://doi.org/10.1016/j.autcon.2022.104273 (2022).
31. Nguyen, T. P. et al. Experimental verification of the field robotic system for pipeline maintenance. *IEEE Access.* **13**, 17382. https://doi.org/10.1109/ACCESS.2025.3529131 (2025).
32. Girvan, M. & Newman, M. E. J. Community structure in social and biological networks. *Nat. Acad. Sci.* **99**(12), 7821–7826. https://doi.org/10.2307/3058918 (2002).
33. Deza, M. M. & Deza, E. *Encyclopedia of distances* (Springer Berlin Heidelberg, 2013).

Acknowledgements

Author contributions

Author A(Wang Xingyu) developed the prototype of spraying robot, designed the algorithms and programs, and formulated the experimental program and clutched the paper. Author B(Xue Zining) did the experimental debugging, data collection, and the production of all graphs 1–9 in the paper.

Funding

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to W.X.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.