

## Research Article

# Robot Floor-Tiling Control Method Based on Finite-State Machine and Visual Measurement in Limited FOV

Shun Wang <sup>1</sup>, Huixing Zhou <sup>2,3</sup>, Zhongyue Zhang <sup>2</sup>, Xiaoyu Zheng <sup>2</sup>,  
and Yannan Lv <sup>2</sup>

<sup>1</sup>School of Civil and Transportation Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China

<sup>2</sup>School of Mechanical-Electronic and Vehicle Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China

<sup>3</sup>Beijing Engineering Research Center of Monitoring for Construction Safety, Beijing 100044, China

Correspondence should be addressed to Huixing Zhou; [zhouhuixing@bucea.edu.cn](mailto:zhouhuixing@bucea.edu.cn)

Received 15 May 2021; Revised 10 August 2021; Accepted 18 August 2021; Published 6 September 2021

Academic Editor: Yi-Zhang Jiang

Copyright © 2021 Shun Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, accelerative aging society is meeting the short supply of young and middle-aged labor. Particularly, most young people are reluctant to work in the construction industry, which has caused the labor cost of floor tiling to rise year by year. In addition, floor tiling requires workers to continuously bend over or lean over to work, which greatly jeopardizes the physical health of them. Therefore, advanced technology applied in floor tiling is highly demanded to replace the traditional manual method. On the context, the automatic method of floor tiling may promote the transformation and upgrading of the industry. Although a few robots for floor tiling have been developed, the automation of existing systems is still at a low level. This paper proposes a robot floor-tiling control strategy based on visual measurement feedback and finite-state machine, in which the calculation of tile position information in limited field of vision is obtained by an improved Canny edge detection and a Hough linear transformation. Moreover, an algorithm for complementing tile position information based on visual measurement is proposed, and the quality of tile laying is evaluated online. To evaluate the effect of the proposed control strategy, the experimental verifications are given. The experimental results indicate that the proposed method can complete the automatic floor tiling with high accuracy.

## 1. Introduction

*1.1. Background and Meaning.* With the rapid development of robot technology and sensor technology [1, 2], construction robot technology has made considerable progress for decades [3–5]. Many construction robots are applied to perform specific tasks on construction sites, such as construction fabrication robot [6], drilling robot [7], and marking robot [8]. The usage of robots to replace manual building maintenance and construction in harsh environment will improve the construction industry [9, 10], such as reducing the physical intensity labor, avoiding construction hazards, improving construction efficiency, and ensuring construction quality. Hence, researchers began to pay more attention to the research of construction robot technology [11, 12].

At present, in the decoration process, the floor tiling still relies on manual operation with some simple instruments (as shown in Figure 1), which is labor intensive and time-consuming. The workers must bend over for a long time to perform the floor tiling, which greatly harms their health [13, 14]. The floor-tiling construction requires the workers to have been qualified with professional skills to ensure construction quality. Due to differences in the proficiency of workers, some prominent problems arise in the floor-tiling construction, such as low efficiency, poor tile flatness, serious empty drum, and waste of construction materials caused by rework. In addition, floor tiling is a highly repetitive construction operation, while the size of the tiles is regular. Therefore, automatic floor tiling by the robot has the potential to replace manual operation, which is in accord with modern development direction of floor tiling.



FIGURE 1: Manual floor-tiling construction.

*1.2. Related Work.* Over the past decade, researchers have been devoting greatly into the research of floor-tiling robots. In 1996, Apostolopoulos et al. [15] proposed a concept of mobile robot to automatically lay tiles. In 2000, Navon [16] developed a floor-tiling robot that realized the location of tiles based on vision, and its efficiency is expected to increase by 2 to 5 times. In 2011, Khan et al. [17] presented a semiautomatic floor-tiling robot, which requires some manual operations to achieve floor tiling. In 2019, Li et al. [18] proposed a vision-based adaptive control method for tile paving, and Liu et al. [19, 20] developed a mobile robot for floor tiling that used laser sensors to measure size and position of tiles. However, in existing related studies, the floor-tiling methods adopt semiautomation or human-machine cooperation, and only some simple visual measurements are carried out. The robustness of the systems is poor. Currently, a simple fact is that the automation and intelligence degree of the robots are low.

*1.2.1. Ceramic Tile Position Measurement.* The ceramic tile position measurement is a prerequisite for accurate floor-tiling control. The existing research results mainly used laser sensors [17, 19, 20] and vision sensors [16–18] to measure the size and position of tiles. The measurement methods with laser sensors require multiple laser sensors or a single sensor for multipoint measurement. Multipoint measurement will make the measurement process more complex and time-consuming. Furthermore, the movement of the laser will induce new errors. The visual measurement method is simple and position characteristic information of ceramic tiles can be extracted from a single image. In addition, the application of robots in industrial environments provides a reference for the application on construction site. Tlach et al. [21] used a measuring assembly on the robot to achieve position measurement, which required the sensor probe to contact the measured object. As for robot online path optimization, Božek [22] proposed a robot path optimization which is applied to spot welding applications in automotive industry. This method realizes online real-time measurement to provide data support for path optimization.

Part of the research has studied ceramic tile measurement in a factory environment. Golkar and Prabuwo [23] proposed an image processing algorithm to measure the length of ceramic tiles. The systems used a canny edge detection to identify the boundary of ceramic tiles and

calculate the size of ceramic tiles, in which deviation was less than 2 mm. The function of this system is to distinguish the ceramic tiles of different specifications on the production line, so the dimensional accuracy is low. Hoceski et al. [24] proposed a defect detection method based on edge detection technology and adopted histogram subtraction to solve the problem of threshold definition of the canny edge detector. This method has good adaptability to ceramic tiles with different textures. Emam and Sayyedbarzani [25] used binocular vision combined with two cameras to measure the size deviation of ceramic tiles, in which measurement accuracy could reach 0.06 mm. Golkar et al. [26] used three cameras to measure the different dimensions of ceramic tiles (edge curvature, length and width, edge crack, and thickness) from multiple perspectives and obtained the target from the background by using a histogram subtraction. An attempt is made to find a suitable threshold to improve the system accuracy that made an effect to succeed. This system was used for automatic quality inspection of ceramic tiles in a production line. The ceramic tile measurement above mainly focuses on a single ceramic tile which is only one side or top view of one tile in the field of view, in which feature extraction is relatively simple.

On the construction site, Deng et al. [27] used Canny edge detection and Hough transform to detect the boundary of the ceramic tile to realize the monitoring of the construction progress of floor tiling. With the comparison of the edge detection results based on different gradient threshold settings, the suitable edge detector threshold was determined. However, the method adopted a manual method to set the threshold, and the measurement accuracy was low and could not meet the needs of robot floor tiling. Lin and Fang [28] proposed a vision-based quality evaluation system for floor tiling, which received the images of finished tiling and extracted geometric features from the images and then evaluated and determined the construction quality of floor tiling. Navon [16] paid attention to the position of the tiles and the damage of the tiles during the floor-tiling process, but the grayscale values of pixels between 31 and 224 were determined as a mistake through the experience in advance, and the robustness was poor.

*1.2.2. Robot Floor-Tiling Control Strategy.* At present, there is little research on robot floor-tiling control strategies. Apostolopoulos et al. [15] proposed a concept of the floor-tiling process control method based on position and force feedback, but the controller did not carry out actually. Navon [16] used a semiautomatic floor-tiling control method, which required manual assistance for positioning. Khan et al. and Li et al. [17, 18] mainly discussed the control method of floor tiling with a vision servo system. The experiments showed that the control method had better robustness, but the floor-tiling conditions were relatively simple. Some researchers have also studied the control methods of component installation in other task scenarios. Roveda et al. proposed a control framework [29] that can implement industrial interaction tasks in uncertain work scenarios. They also describe a human-robot cooperation

methodology to install heavy and bulky components based on marker-based visual servoing and force control [30]. Therefore, the stable control system for robotic floor tiling needs an in-depth study.

The control strategy based on the finite-state machine is to summarize it as a finite state when the system is running, and the state switching is realized by external signal excitation. Therefore, the whole system has good stability. The finite-state machine control method has been applied in many fields such as the centralized control for DC microgrids [31], the integrated navigation and control for robots [32], the fuel cell management strategy [33], and the road automatic driving control [34]. Floor tiling is a repetitive construction process. Each floor tiling can be divided into a finite number of motion states. Applying finite-state machine theory to control floor tiling can simplify the control complexity of the floor-tiling process and improve the robustness of the system.

*1.3. Innovation of This Article.* This paper aims at realizing a stable control method of floor tiling and proposes a floor-tiling control strategy based on the vision measurement in the limited FOV and finite-state machine control strategy. The main contributions of this paper are summarized as follows:

- (1) It proposes a fast and robust ceramic tile measurement method based on the vision with a limited FOV. Firstly, an image of the floor-tiling process is obtained in the limited FOV. Next, the edge lines of ceramic tiles are attained through an improved Canny edge detection algorithm and Hough transform algorithm. Finally, a clustering method is proposed to distinguish the edges of ceramic tiles, and the deviation between the desired position and actual position of the tile is calculated.
- (2) A floor-tiling control strategy based on finite-state machine theory is proposed. The floor-tiling process is split into finite states, and the important state parameters, especially the deviation from the actual position of the ceramic tile to the desired position of that, drive the finite-state machine to act floor-tiling state transition. The control strategy can systematically summarize the state of the floor-tiling process and improve the robustness of the control system.

The rest of this paper is organized as follows. In Section 2, the framework of the floor tile robot II (FTR-II) system is briefly described, including software and hardware systems. It states the ceramic tile measurement method used to obtain the position of ceramic tiles in Section 3. After that, a floor-tiling control strategy based on a finite-state machine is proposed in Section 4. In Section 5, the experimental verifications with the floor-tiling robot system based on the proposed method are given. Finally, Section 6 summarizes the conclusions and future work.

## 2. Floor-Tiling Robot II (FTR-II) System

Floor-tiling robot II (FTR-II) system that was used to verify the robotic tile-paving control method mentioned in this

article consists of four function layers: user layer, system layer, drive layer, and execution layer, as shown in Figure 2.

The user layer is mainly for human-computer interaction. The execution layer implements the subactions of the tile-laying process, and its hardware includes UR10 robot, mobile platform, tile grabbing, and sensors. When the contact force exceeds 10 kg, the robot will immediately stop working and issue a warning, so the safety of FTR-II can be guaranteed. The driver layer is to solve the problem that the system layer cannot directly communicate with the execution layer hardware. The system layer realizes system control and real-time data analysis in the process of tile laying. The hardware and software components of the FTR-II system are shown in Table 1.

## 3. Vision-Based Ceramic Tile Position Measurement

*3.1. Floor Tiling Process Analysis.* Figure 3 shows the entire process of floor tiling on the site. Firstly, the worker manually lays the ceramic tile no.1 as a benchmark and the rest ceramic tiles are laid fully automatic by the robot in a sequence shown in Figure 3 from no. 2 to no.12. Excluding the floor-tiling process of the ceramic tile 1, the situations of the entire floor tiling are classified into 3 cases:

- (a) When the robot is laying the ceramic tiles no. 2 or no.3, the ceramic tile is on the right side of the placed tile No.1. Since there is no ceramic tile on the upper side, only the left ceramic tile no. 1 can be used as a benchmark, as shown in Figure 4(a).
- (b) When the robot is laying the ceramic tiles no. 4, no.7, or no.10, the ceramic tile to be laid is on the lower side of the laid tiles. Since there is no ceramic tile in the left of the view, only the top ceramic tile may be used as a benchmark, as shown in Figure 4(b).
- (c) When the robot is laying the other ceramic tiles, different from conditions (a) and (b), the ceramic tiles to be laid are all on the lower-right side of the ceramic tiles that are already laid. Thus, the ceramic tiles on the left and the top side could be used as a benchmark, as shown in Figure 4(c).

It should be noted that, as the three cases above, more tiles can be laid.

*3.2. Obtain Ceramic Tile Image in a Limited FOV.* The FTR-II can take an image to distinguish the position of a tile from the adjacent one. Then, the coordinate deviation between the actual position and the desired position is calculated. We hope to obtain an image, clear enough, and extract accurate edge feature information from it. We are supposed to take two main problems into consideration while determining the camera position:

- (1) The edge features of the ceramic tiles measured are detected accurately by the edge detection algorithm
- (2) It is required to reduce the viewing distance between the camera and the ceramic tile to ensure that the

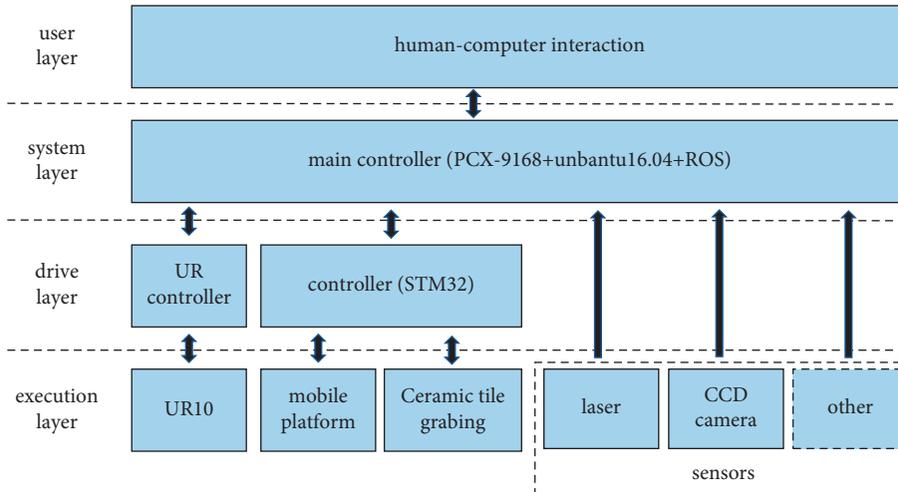


FIGURE 2: Frame of the FTR-II control system.

TABLE 1: System components of FTR-II.

Function module	Devices/tool	Description
Main controller	Embedded industrial computer PCX-9168 Ubuntu20.04 + ROS	(i) Control hardware platform (ii) Control software platform
Data processing and evaluation	Opencv3.4 Visual studio code 1.52.1for ubuntu20.04	(i) Convert image to grayscale one (ii) Image edge detection (iii) Image feature extraction (i) Floor position calculation (ii) Floor-tiling quality evaluation
Manipulator	UR10 ZK2A15K5HL-08	(i) Floor-tiling position movement
Ceramic tile grab	ZPT32CNJ10-B5-A10	(i) Vacuum generator (ii) Vacuum suction cups
Mobile platform	Mecanum wheel omnidirectional mobile platform	(i) Floor-tiling process moving
Vertical measurement module	Laser sensor keyence IL-150	(i) Vertical measurement feedback
Image capture	SONY IMX214 13M pixel CMOS sensor	(i) Digitize image

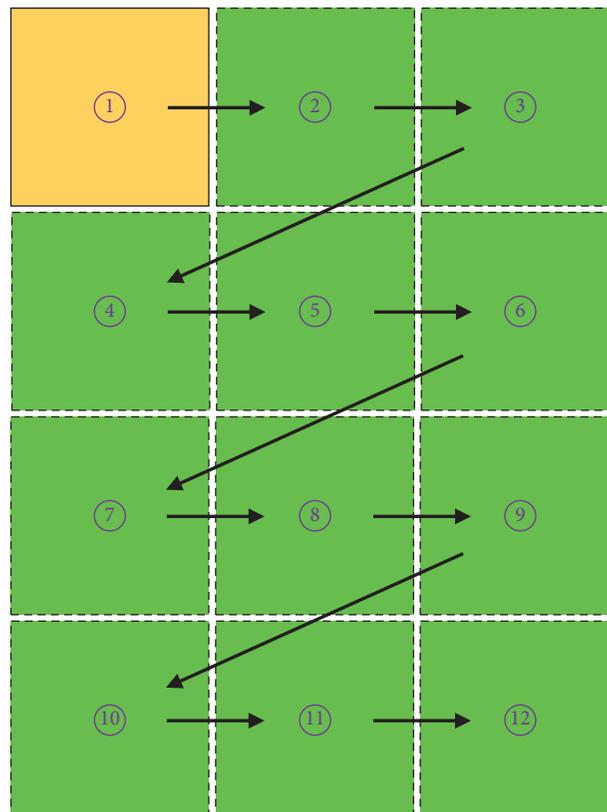


FIGURE 3: The process of ceramic tile laying.

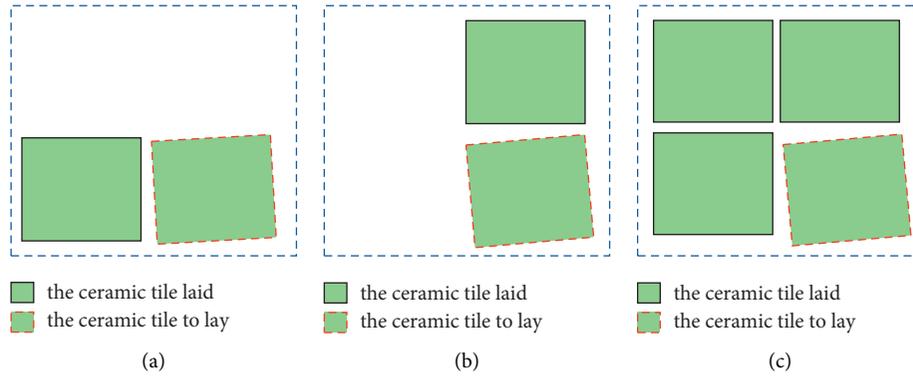


FIGURE 4: The classification of ceramic tile laying: (a) laying ceramic tile on the right side, (b) laying ceramic tile on the lower side, and (c) laying ceramic tile on the lower-right side.

actual physical size of each pixel point mapping can be as small as possible

The camera is mounted on the robotic arm to ensure that the distance between the camera and the tile is small enough to take a clearer image. However, the position of the robotic arm and the tile being laid is fixed. If the robot takes multiple images to cover the full workspace and merge them to create a single image, the camera should be mounted on another manipulator to realize that the position of the tiles remains unchanged during the process of taking images, which would make the system more complicated. According to the above analysis, this paper adopts a method of obtaining ceramic tile images with a limited field of view. Figure 5 shows the measurement method diagram of floor tiling based on the vision. The end effector with four suckers is grabbing a ceramic tile to lay, and a camera is mounted on the robotic arm with a connecting device in order to obtain images in the process of ceramic tile laying. It is important to note that the spatial position relationship between the camera and the laid tile is relatively fixed. Figure 6 shows the images of the ceramic tiles in the limited FOV under the three different conditions.

**3.3. Edge Detection of Ceramic Tile Image.** The Canny edge detection algorithm [35] was firstly proposed by Canny in 1986 and has been widely used in various computer vision systems [36, 37]. Canny edge detection is not susceptible to

noise interference and can detect real weak edges. One advantage is that two different thresholds are selected to obtain edges by excluding weak edges. However, the method of manually setting the high and low thresholds is not highly adaptable to varied environment.

This paper proposes an improved Canny edge detection algorithm to extract the edges of the ceramic tiles. Figure 7 demonstrates the implementation process of the algorithm, including taking an image, Grayscale image, Gaussian filtering, nonmaximum suppression, Gradient calculation, Otsu adaptive threshold, and edge detection image.

**3.3.1. Convert Image to Grayscale.** Since the image obtained by the camera is a color one, we use the formula  $0.299R + 0.587G + 0.114B$  to convert a color image to a gray-level one.

**3.3.2. Gaussian Filtering.** The discrete Gaussian convolution kernel matrix is  $(2k + 1) \times (2k + 1)$  dimension, which is calculated by formula (1). Therein,  $\sigma$  is the variance and  $k$  is the dimension of the kernel matrix. It is noticeable that the choice of  $H_{ij}$  will affect the performance of the Canny detector. With increasing size, the detector is less sensitive to noise, but the positioning error of edge detection will also increase slightly. This paper uses a  $3 \times 3$  window for detection, and the pixel to be filtered is  $e$ . After Gaussian filtering, the gray value of pixel  $e$  is shown in formula (2):

$$H_{ij} = \frac{1}{2\pi\sigma} \exp\left(-\frac{(i - (k + 1))^2 + (j - (k + 1))^2}{2\sigma^2}\right), \quad 1 \leq i, j \leq (2k + 1), \quad (1)$$

$$e = H * A \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} * \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \text{sum} \left\{ \begin{bmatrix} a \times h_{11} & b \times h_{12} & c \times h_{13} \\ d \times h_{21} & e \times h_{22} & f \times h_{23} \\ g \times h_{31} & h \times h_{32} & i \times h_{33} \end{bmatrix} \right\}, \quad (2)$$

where  $*$  is the convolution symbol and sum means the sum of all elements in the matrix.

**3.3.3. Gradient Calculation.**  $G_x(i, j)$  and  $G_y(i, j)$  are the first-order partial derivatives in the  $x$ -direction and

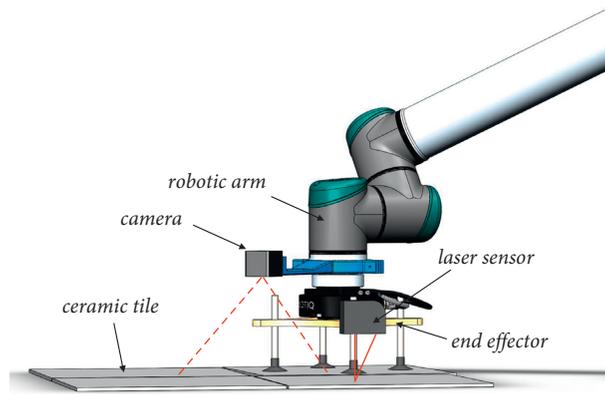


FIGURE 5: The diagram of floor tiling based on visual measurement.

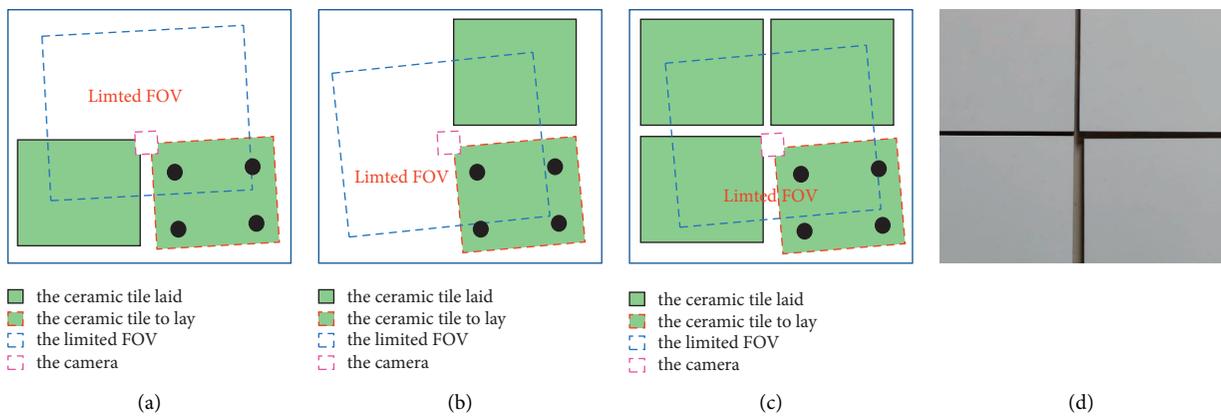


FIGURE 6: Acquire a tile image in a limited FOV: (a) working condition 1; (b) working condition 2; (c) working condition 3; (d) acquired ceramic tile image.

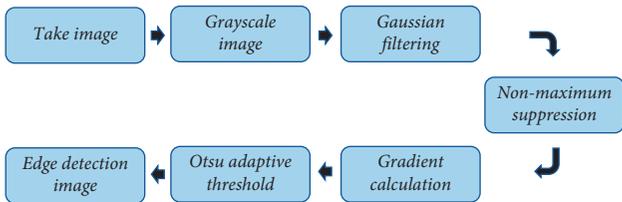


FIGURE 7: The improved canny edge detection algorithm.

$y$ -direction, respectively. The magnitude and direction of the gradient are shown in formulas (3) and (4):

$$G_x(x, y) = \sqrt{G_x^2(i, j) + G_y^2(i, j)}, \quad (3)$$

$$\theta(i, j) = \arctan\left(\frac{G_y(i, j)}{G_x(i, j)}\right). \quad (4)$$

3.3.4. *Nonmaxima Suppression (NMS)*. Nonmaximum suppression is used to obtain a more accurate image edge, which consists of many single pixels. As shown in Figure 8, the gradient direction of the observed pixel is perpendicular to the direction of the image edge. There are two adjacent pixel points not on the edge of the image, but the gradient direction may be

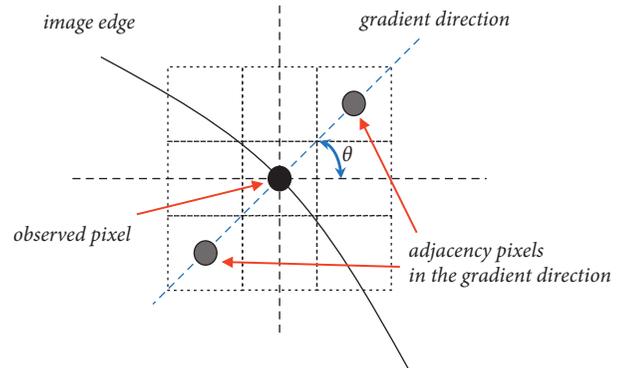


FIGURE 8: Nonmaximum suppression in the gradient direction.

calculated by formula (4). The gray value of the observed pixel is compared with the gray values of its adjacency pixels, and the gray value of the pixel nonmaximum is set to 0. Later on, the pixels on the image with nonzero gray values are traversed, and all nonmaximum pixels are eliminated.

3.3.5. *Self-Adaptive Threshold Selection Based on Otsu Algorithm*. The image processed by the NMS method still contains a lot of noise and false edges. Therefore, the Canny

algorithm uses double thresholds to eliminate noise and false edges. For the Canny algorithm, it is very important to choose a reasonable set of thresholds. The traditional Canny algorithm uses manually selected double-threshold values and cannot adapt to various environments where the overall brightness is varied. This paper uses a self-adaptive method based on the Otsu to select the threshold value [38].

It is supposed that  $Q = [0, T - 1]$  is the grayscale range of the image pixels. Taking  $t$  as the boundary, the set of pixels is divided into  $Q_0 = [0, t - 1]$  and  $Q_1 = [t, T - 1]$ , and the probabilities are  $\gamma_0 = \sum_{i=0}^{t-1} pi$  and  $\gamma_1 = \sum_{i=t}^{L-t} pi = 1 - \gamma_0$ , in the given order. The average gray values of  $Q_0$  and  $Q_1$  are  $m_0 = \sum_{i=0}^{t-1} (i(pi/\gamma_0))$  and  $m_1 = \sum_{i=t}^{L-t} (i(pi/\gamma_1))$ , respectively. Hence, the average gray value of  $Q$  is calculated as  $m = \gamma_0 m_0 + \gamma_1 m_1$ . The variance between the two classes is expressed in

$$\eta(t)^2 = \gamma_0(m_0 - m)^2 + \gamma_1(m - m_1)^2 = \gamma_0\gamma_1(m_0 - m_1)^2. \quad (5)$$

Calculating the maximum value of  $t$  through formula (5), we can obtain the upper threshold value that is marked as  $t^*$ .

Figure 9 reveals the detection results of the traditional Canny edge detection algorithm and the improved Canny edge detection algorithm, where  $a$  and  $b$  are, respectively, the upper and lower limit thresholds. Comparing Figures 9(a) and 9(b), it is suggested that the detection results of the traditional Canny algorithm vary greatly by manually setting different thresholds. With self-adaptive selecting upper and lower thresholds, the edges detected based on the improved Canny edge detection algorithm illustrated in Figure 9(c) are better than that based on the traditional one.

The gray values of the image are greatly affected by the light, so the same upper and lower threshold cannot be used to detect edges of the tile image obtained in the construction environment, which has different lighting conditions. The improved Canny detection algorithm has better light intensity adaptability and is taken in this study.

**3.4. Extract Straight Lines and Recognize Tile Boundaries.** In a  $x - o - y$  cartesian coordinate system, a straight line is expressed as  $y = kx + b$ , and points on the same line have the same  $b$  and  $k$  values. If the straight line is converted to the  $b - o - k$  parameter coordinate system, it should be expressed as  $b = -xk + y$ . Points on the same straight line have the same  $b$  and  $k$  values, so in the parameter coordinate system, they intersect at one point. When the number of parameter straight lines that intersect at the same point reaches the preset value, we think that we have found enough points to form a straight line. This is the principle of Hough transform (HT). However, it is possible that the two edges of the image to detect have the same  $k$  and  $b$  value.

In this paper, we use progressive probabilistic Hough transform (PPHT) to extract tile edges in images [39]. In comparison with Hough transform, the main improvements of the PPHT algorithm are described as follows:

- (1) Use randomly selected pixels from the input image instead of traversing all pixels
- (2) Set the maximum pixel interval threshold  $\text{tre}(N_{\min})$  and the minimum straight line length threshold  $\text{tre}(L_{\min})$  allowed to form the connecting line segment
- (3) Find the longest line segment that is unbroken and does not exceed a given threshold

By setting the maximum pixel interval allowed to compose the connecting line segment, the PPHT algorithm can effectively distinguish two-tile edges detected as one edge by the HT algorithm. For example, Figure 10(b) shows that line 1 detected based on the HT algorithm is composed of two-tile boundaries, which coincide exactly on a straight line. When the PPHT algorithm is used to extract a straight line from the above grayscale image (Figure 10(a)), two straight lines can be acquired accurately (Figure 10(c)).

The tile acquired through the PPHT is expressed in the Cartesian coordinate system as  $a_k x + b_k y = c_k$ . The set of detected points is  $C_k = [x_i, y_i]$ ,  $0 \leq i \leq n_k$ , where  $n_k$  is the number of detected points on the line marked as  $k$ . Furthermore, the average values of  $\bar{x}_k$  and  $\bar{y}_k$  are calculated as

$$\begin{aligned} \bar{x}_k &= \frac{1}{n_k} \sum_{i=1}^i x_i, \\ \bar{y}_k &= \frac{1}{n_k} \sum_{i=1}^i y_i, \end{aligned} \quad (6)$$

$$0 < k \leq 8, 0 < i \leq n_k,$$

where  $n_k$  is the number of points included in  $C_k$ . The recognition strategy of tile boundaries is targeted to sort and compare the values of  $\bar{x}_k$  and  $\bar{y}_k$ , which are shown in Table 1 and finally to realize the recognition of 8 tile boundaries.

Table 2 describes a strategy for recognizing the tile boundaries. The summary of the strategy is as follows:

- (1) Sort  $\{\bar{x}_k\}$ .
- (2) Pick the minimum two points  $\bar{x}_i$  and  $\bar{x}_j$ , and then, compare them with the corresponding values of  $\bar{y}_i$  and  $\bar{y}_j$ . The smaller one is no. 1 boundary, and the larger one is no. 5 boundary.
- (3) Pick the maximum two points  $\bar{x}_i$  and  $\bar{x}_j$ , and then, compare the corresponding values of  $\bar{y}_i$  and  $\bar{y}_j$ . The smaller one is no. 4 boundary and the larger one is no. 8 boundary.
- (4) Sort  $\{\bar{y}_k\}$ . The same principle recognizes the tile boundaries 2, 3, 6, and 7.

The result is presented in Figure 11. We can get 8 straight lines numbered from 1 to 8.

**3.5. Position Calculation and Coordinate Transformation.** When we are laying ceramic tile, the deviation between the current position and the target position is calculated to drive the end effector of the robot to move. In the two-

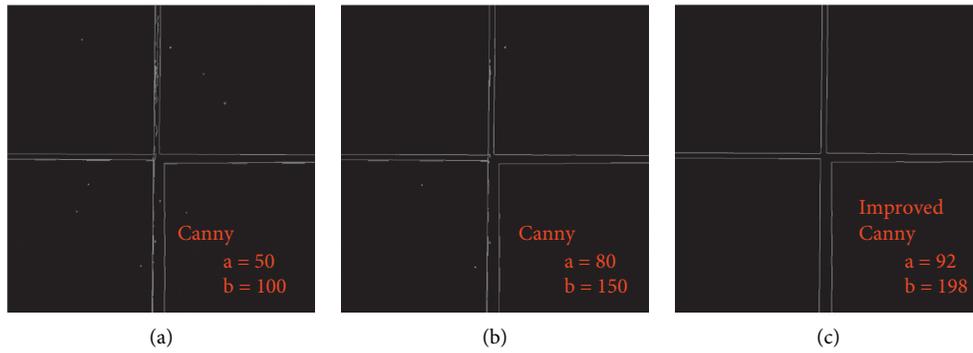


FIGURE 9: Canny edge detection results under different threshold settings: (a) canny edge detection with  $a = 50$  and  $b = 100$ ; (b) canny edge detection with  $a = 80$  and  $b = 150$ ; (c) improved canny edge detection.

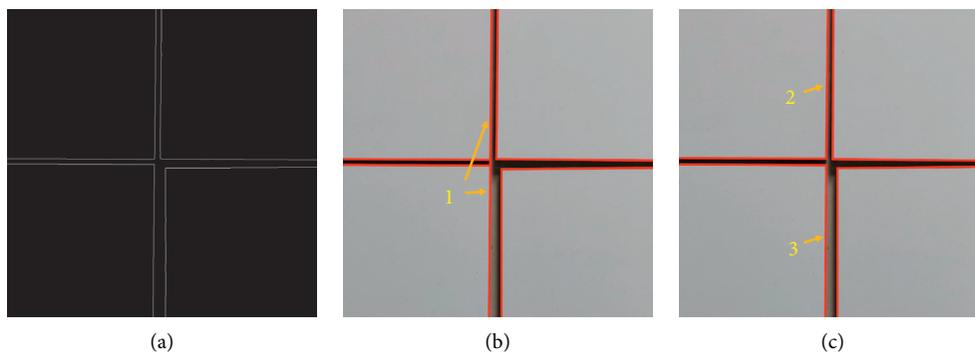


FIGURE 10: Hough transform for detecting ceramic tile boundaries: (a) improved canny edge detection; (b) HT algorithms; (c) PPHT algorithms.

TABLE 2: The recognition strategy of tile boundaries (condition 3 as an example).

Tile boundary number $k$	$\bar{x}_k$			$\bar{y}_k$		
	Two minimum	Two maximum	Compare values	Two minimum	Two maximum	Compare values
1	✓					
5	✓					$1 < 5$
2			$2 < 3$	✓		
3				✓		
6			$6 < 7$		✓	
7					✓	
4		✓				$4 < 8$
8		✓				

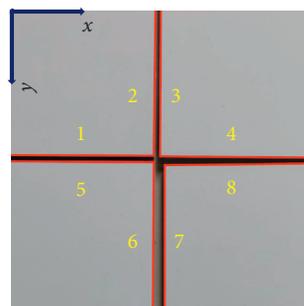


FIGURE 11: Ceramic tile edge detection and position recognition results.

dimensional rectangular coordinate  $x-o-y$  system, movement of the ceramic tile from current position to desired position can be divided into two processes: horizontal translate along vector  $\overrightarrow{D_1D_2} = (\Delta x, \Delta y)$  and rotate  $\theta$  degree about counterclockwise around  $D_2$ .  $D_2$  is the geometric center of the desired position and  $D_1$  is the geometric center of the current position, as shown in Figure 12.

In the three-dimensional motion space of the robot, the motion matrix of the robot end effector is expressed as

$$T_r = \begin{bmatrix} -\cos \theta & -\sin \theta & 0 & \Delta x \\ \sin \theta & \cos \theta & 0 & \Delta y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (7)$$

### 3.6. Complement Position Information of Ceramic Tiles.

To evaluate the quality of tile laying, we need to wait for all tiles to be laid, and the tiles initially solidify after 24 hours to ensure that the ground has sufficient strength. Afterwards, workers can step on the tiles to check the quality of the tiling. Therefore, this paper proposes a real-time quality evaluation method of laying tile based on the position information of tiles obtained during the tile laying process. However, the ceramic tile visual measurement servo system employs a limited FOV method to obtain images so that it cannot obtain the position information of the entire tiles through a single image. Therefore, only when all tiles are laid, we can obtain the position information of tiles.

This requires us to use the position information of partial tiles, which are classified by the three different conditions mentioned in front part, to complement the global information of the tiling. The specific method is as follows.

In Figure 13, tiles 2 and 3 are complemented by data collected under working condition 1, tiles 4, 7, and 10 are complemented by collecting data under working condition 2, and other tiles are complemented by collecting data under working condition 3.

The use of image complements' algorithms will introduce an assumption: the base tile is  $l \times l$  in side length and  $\Delta l$  in size deviation. The international standard BSE-NISO10545-2:1997 defines the characteristics of various shape and size errors of ceramic tiles, as well as the measurement and inspection methods of related characteristics [40]. The international standard ISO13006:2012 specifies the tolerance range of shape and size of various types of ceramic tiles [41].

We will analyze the impact of the reference tiles error on the measurement results. The actual size of tile 1 is  $(l + \Delta l_{11}) \times (l + \Delta l_{12})$ , where  $\Delta l_{11}$  is the  $x$ -coordinate deviation and  $\Delta l_{12}$  is the  $y$ -coordinate deviation. When we are laying tile 2, the size is calculated as  $(l + \Delta l_{21}) \times (l + \Delta l_{22})$ , where  $\Delta l_{21}$  and  $\Delta l_{22}$  are the vertical position deviation and horizontal position deviation of visual inspection, respectively. However, the actual size of tile 2 is  $(l + \Delta l_{21} + \Delta l_{11}) \times (l + \Delta l_{22} + \Delta l_{12})$ . In the same way, the tiles in the first column will introduce a  $x$ -coordinate deviation  $\Delta l_{11}$ , and the tiles in the first row will introduce a  $y$ -

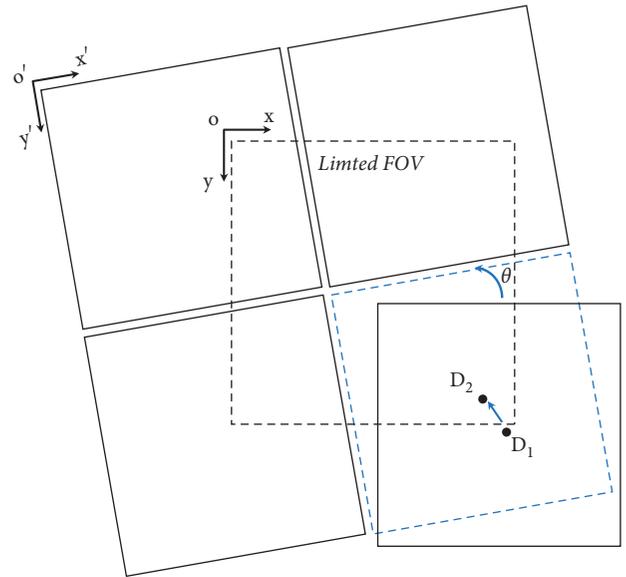
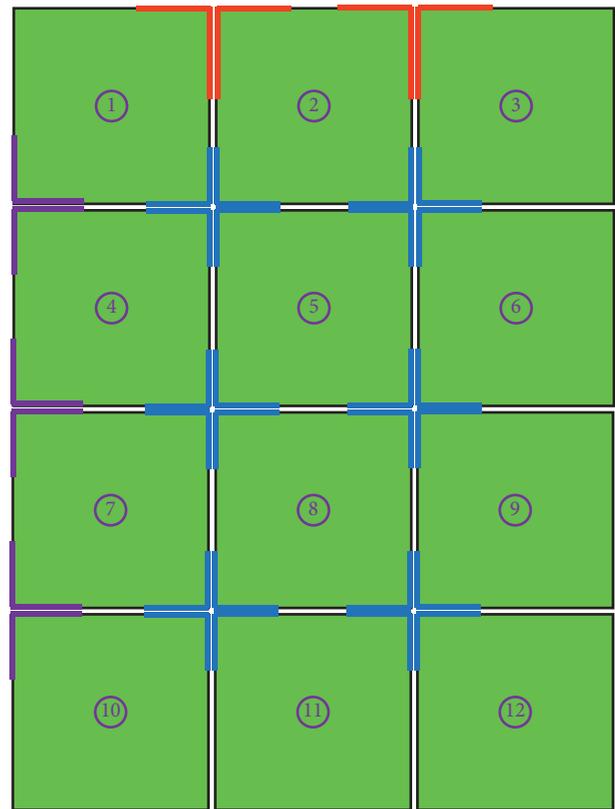


FIGURE 12: The actual position and expected position of the ceramic tile.



- T Position data in condition 1
- L Position data in condition 2
- + Position data in condition 3

FIGURE 13: Schematic diagram of the completion method of ceramic tile-paving effect.

coordinate deviation  $\Delta l_{12}$ . When the tiling effect is evaluated,  $\Delta l_{11}$  and  $\Delta l_{12}$  as intermediate variables do not affect the final evaluation of the straightness of the tiles. After the effect is

eliminated, we conduct a real-time quality evaluation of tiling, and the algorithm outline is as follows:

- (1) Obtain the pixel coordinates of the corner point from the edge of the tile measured based on vision
- (2) Use the least square method to fit a straight line
- (3) Calculate the distance of each point from points on the fitted straight line

## 4. Control Strategy Based on FSM

**4.1. Floor-Tiling Operation Control Variable and Status Parameter Information.** The floor-tiling robot system consists of a vision sensor, a mechanical arm, an end effector, a laser sensor, an edge detector, and a position calculator. In the floor-tiling process, the robot operation control variables correspond to different system execution units, which are shown in Table 3.

The operating status of the robot mainly reflects the current working status of the floor-tiling robot and is recorded in the form of status parameters, which mainly include the position information of tiles and operating conditions of tiling. Table 4 shows the system operation state parameters.

**4.2. Floor-Tiling Control Based on FSM.** A finite-state machine (FSM) is a mathematical model with discrete inputs and outputs. It can record the internal state of historical inputs, determine the control execution ability of the next state and behavior according to the current input, and can effectively describe the life cycle of a system.

This paper proposes a floor-tiling control strategy based on FSM for the first time. To facilitate the analysis, the FSM describes the action of laying tile as

$$\Sigma_{\text{fsm}} = \{S, A, P, C\}, \quad (8)$$

where  $S = \{S_1, S_2, \dots, S_n\}$  is a set of  $n$  robot states,  $A = \{A_1, A_2, \dots, A_n\}$  is a set of  $n$  robot program actions,  $P = \{P_1, P_2, \dots, P_n\}$  is a set of  $n$  robot state parameters, and  $C = \{C_1, C_2, \dots, C_n\}$  is a set of  $n$  state transition conditions.

The relationship between the 4 sets above is expressed as follows:

$$S_i = \{A_i, P_i, C_i\}, \quad i = 1, 2, \dots, n. \quad (9)$$

To facilitate the analysis and description of the state transition process, a directed graph  $G$  is defined as formula (10) based on the graph theory and formula (9):

$$G \triangleq \{S, E\}. \quad (10)$$

The floor-tiling states' set is abstracted as the vertices of the directed graph, and the state transition condition set  $C$  is abstracted as the edges of the directed graph; then, the edge set is defined as follows:

$$E \triangleq \{E_1, E_2, \dots, E_r\} = \{\{S_i, S_j\} \mid c_{i,j} \in C_{ij}\}, \quad (11)$$

where  $c_{i,j}$  is the condition for the transition from state  $S_i$  to state  $S_j$ , which is abstracted as the right of  $\{S_i, S_j\}$ . When the value of  $c_{i,j}$  is 1, the condition is satisfied. When the value is 0, the condition is not satisfied, and the value is  $-1$ , which means that the condition does not exist. The  $r$  adjacent vertices of  $S_i$  are expressed as follows:

$$S_i^a \triangleq \{S_{i,1}^a, S_{i,2}^a, \dots, S_{i,r}^a\} = \{S_j \mid (S_i, S_j) \in E\}, \quad (12)$$

where  $S_{i,i}^a = \{A_{i,r}^a, P_{i,r}^a, C_{i,r}^a\}$ . The FSM diagram of robot floor tiling is shown in Figure 14.

The state information of the robot floor tiling is shown in Table 5. The process of the proposed control strategy for robotic laying a tile is as follows:

- (1) In the S0 state, the system checks the state of the robot and performs initialization operations.
- (2) When receiving the signal M1, the system turns to S1 and takes a floor-tiling image.
- (3) When the image has been acquired, the system runs the edge detector for edge detection. Under working condition 1, the system turns to S2; under working condition 2, the system turns to S3; under working condition 3, the system turns to S4.
- (4) When the edge detection is completed, the system turns to S5 and performs plane position calculation to generate a coordinate transformation matrix  $T_r$ , composed of the translation and rotation from the current position to the desired position.
- (5) When the  $T_r$  matrix has been calculated, the system turns to S8. The system measures the height of the tile to lay from the ground with a laser sensor.
- (6) When the position of the tile, including vertical height and horizontal posture, has been acquired, the system controls the robotic arm to move the tile from the current position to the desired position, and the system sequentially turns to S9, S6, and S7.
- (7) The system repeats steps (2)–(5) to calculate the position of the tile. If the position deviation of the tile exceeds the preset value, the system runs step (6).
- (8) When the position deviation of the tile is within the preset value range, the system turns to S10, the suction cup is released, and the laying of a tile is completed.

With the aforementioned method, the robotic floor tiling is realized.

## 5. Experimental Verification and Discussion

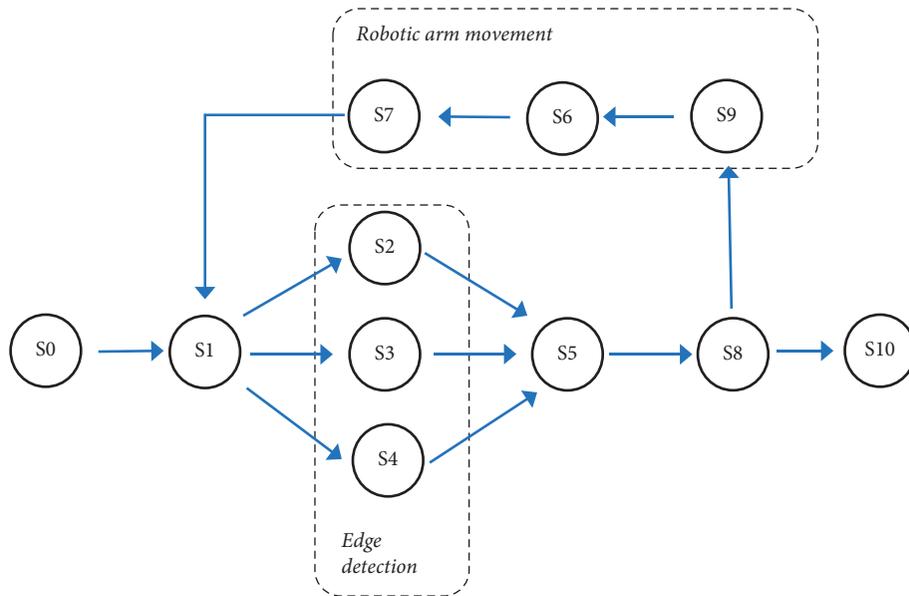
**5.1. Experiment Preparation.** To evaluate the performance of the proposed control strategy, this paper uses the floor-tiling robot II (FTR-II) to perform experimental verification at a laboratory site. Before the experiment, the sizes of 12 tiles are measured to ensure that they are suitable and avoid undermining the experimental results. Each tile is measured 3 times, and the average value is calculated. As shown in

TABLE 3: Robot operation control variables and function description.

Action execution	Control variable	Function
Camera	$Y_1$	Perform a camera action to get a picture
Robotic arm	$Y_2$	Perform a movement of the robotic arm
End effector	$Y_3$	Grab and place a ceramic tile with vacuum suction cups
Laser sensor	$Y_4$	Take vertical height measurement
Edge detector	$Y_5$	Detect the tile edge
Position calculation	$Y_6$	Calculate the deviation between the actual position of the tiles and the expected position

TABLE 4: Robot operating status parameter information.

Status parameter	Value	Function
Horizontal position deviation	$D_1$	$D_1 = (\Delta x, \Delta y)$
Angle position deviation	$D_2$	$D_2 = \{\theta\}$
Vertical height deviation	$D_3$	$D_3 = \{h\}$
Floor-tiling condition 1	con1	The floor-tiling condition shown in Figure 4(a)
Floor-tiling condition 2	con2	The floor-tiling condition shown in Figure 4(b)
Floor-tiling condition 3	con3	The floor-tiling condition shown in Figure 4(c)
Working status	$M_0$	Working
	$M_1$	Unworking



- S0: Initialize the system
- S1: Take an image of ceramic tiles
- S2: Detect the edges in condition 1
- S3: Detect the edges in condition 2
- S4: Detect the edges in condition 3
- S5: Calculate position of all ceramic tiles
- S6: Move the ceramic tile horizontally
- S7: Rotate the ceramic tile horizontally
- S8: Measure the height of the ceramic tile by laser sensor
- S9: Move the ceramic tile vertically
- S10: Finish paving a ceramic tile

FIGURE 14: The FSM diagram of robot floor tiling based on visual measurement.

Table 6, the average deviation is 0.05 mm, and the maximum deviation is 0.14 mm.

5.2. Experiment at a Laboratory Site. Figure 15 shows the results of the floor-tiling experiment at a laboratory site. FTR-II continuously lays 12 tiles with a size of 300 mm × 300 mm on

the ground, and the expected gap between adjacent tiles is 3 mm. The tiling sequence is shown in Figure 3, where tile 1 is used as a benchmark tile. With the limit of the robotic arm working range, the system lays tiles from no. 2 to no. 6 when the robot mobile platform is standstill. Then, the robot mobile platform moves 600 mm back to meet the working distance range of the robotic arm, and the system lays tiles from no. 7 to

TABLE 5: Status information of robot floor tiling.

Status	Function	Actions and parameters	Conversion condition
$S_0$	Initialize	Initialize $A_0, P_0$	$C_{0,1} = M1$
$S_1$	Take image	$A_1 = \{Y_1 = 1\}$	$C_{1,2} = \text{con}1 \cap \{Y_1 = 0\}$ $C_{1,3} = \text{con}1 \cap \{Y_1 = 0\}$
$S_2$	Edge detection	$A_2 = \{Y_5 = 1\}$	$C_{1,3} = \text{con}1 \cap \{Y_1 = 0\}$
$S_3$		$A_3 = \{Y_5 = 1\}$	$C_{2,5} = \text{con}1 \cap \{Y_5 = 0\}$
$S_4$		$A_4 = \{Y_5 = 1\}$	$C_{3,5} = \text{con}1 \cap \{Y_5 = 0\}$
$S_5$	Calculate position	$A_5 = \{Y_6 = 1\}$	$C_{4,5} = \text{con}1 \cap \{Y_5 = 0\}$
$S_6$		$A_6 = \{Y_4 = 0, Y_2 = 1\}, P_6 = D_1$	$C_{5,8} = \{Y_6 = 0\}$
$S_7$	Move robotic arm	$A_7 = \{Y_4 = 0, Y_2 = 1\}, P_7 = D_2$	$C_{6,7} = D_1 \cap \{Y_2 = 0\}$
$S_9$		$A_9 = \{Y_4 = 0, Y_2 = 1\}$	$C_{7,1} = D_2 \cap \{Y_6 = 0\}$
$S_8$	Measure height	$A_8 = \{Y_5 = 0, Y_4 = 1\}, P_8 = \{D_3\}$	$C_{9,6} = \{Y_4 = 0\}$
$S_{10}$	Releases tile	$A_{10} = \{Y_4 = 0, Y_3 = 0\}$	$C_{8,9} = D_3 \cap \{Y_6 = 0\}$ $C_{8,10} = \{\text{End}\} \cap M0$

TABLE 6: Dimensional measurements of ceramic tiles for experiment.

Tile size	No. 1 (mm)	No. 2 (mm)	No. 3 (mm)	No. 4 (mm)	No. 5 (mm)	No. 6 (mm)	No. 7 (mm)	No. 8 (mm)	No. 9 (mm)	No. 10 (mm)	No. 11 (mm)	No. 12 (mm)
X-coordinate	300.03	299.99	300.14	299.97	300.05	300.05	300.07	300.03	300.05	300.06	299.94	300.08
Y-coordinate	300.02	300.06	300.03	300.08	299.96	299.98	300.05	300.04	300.02	300.11	300.02	300.10



FIGURE 15: Robotic floor-tiling experimentation at a laboratory site.

no. 12. It should be noted that the movement of the mobile platform will cause a deviation of 2 mm. Since the system uses vision for real-time positioning, the deviations above will not affect the quality of laying tiles.

After the paving is completed, a line laser is used to detect the position of the tiles 2–12. The coordinate data of 48 points are obtained in one floor-tiling experiment, each tile in the 12 tiles with 4 fixed points. We have repeated the experiments 3 times, obtained a total of 144 points of coordinates and deviation values, and calculated the position deviation  $\Delta x$  and  $\Delta y$ , which is shown in Table 7.

Figure 16 illustrates the frequency distributions of  $\Delta x$  and  $\Delta y$  in the  $xI - oI - yI$  coordinate system. The average deviations in the  $x$ -direction and  $y$ -direction are 0.04 mm and 0.11 mm, respectively, while 98.5% of the measured values have deviations  $<0.5$  mm. Noticeably, there is no abnormal interruption in the whole floor-tiling process. The system takes 385 s to lay all 12 tiles, and the time required for place one ceramic tile is 32.1 s. Thus, we confirm that the control strategy proposed in this study can be used to perform accurate floor-tiling operations.

### 5.3. Position Deviation Based on the Online Evaluation.

We get a total of 48 corner coordinates in 8 rows and 6 columns. The corner coordinates of the tiles in the coordinate system are distributed as shown in Figure 17. The black point is the origin coordinate, the blue points are the reference corner, the red points are the measurement corner, and the green points are the tile corner complemented. The coordinates' deviation of each point is calculated, and the results are shown in Table 8, where  $\Delta x$  is the  $x$ -coordinate deviation and  $\Delta y$  is the  $y$ -coordinate deviation.

Table 8 shows the deviation results calculated by the complement method. The average deviations in the  $x$ -direction and  $y$ -direction are 0.06 mm and 0.18 mm, respectively, while 91.5% of the measured values have deviations  $<0.5$  mm.

By comparing Table 6 with Table 7, the deviation of online evaluation is significantly higher than the deviation of the measurement deviation, which is shown in Table 9. It is supposed that the size of the tile is 300 mm  $\times$  300 mm, which allows the deviation of the actual size of the tile to be introduced into the online evaluation system.

TABLE 7: Results of the robot floor-tiling experiment.

Parameters	Average deviation (mm)	Standard deviation (mm)	Maximum deviation (mm)	Fraction of deviations < 0.5 mm (%)
$\Delta x$	0.04	0.04	0.54	98
$\Delta y$	0.11	0.03	0.51	99
$\sqrt{\Delta x^2 + \Delta y^2}$	0.23	0.02	0.74	92

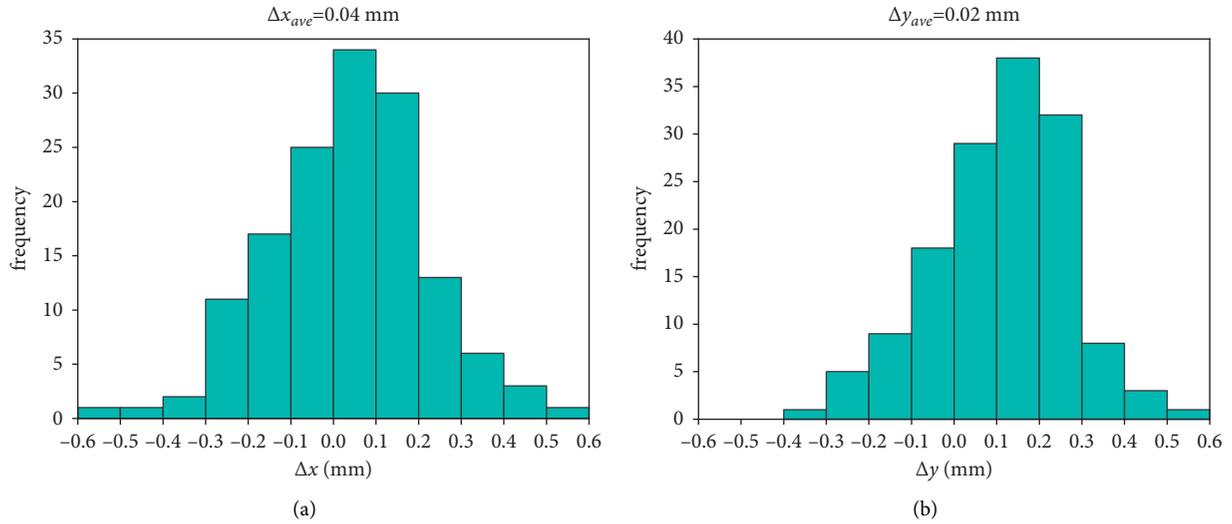


FIGURE 16: Frequency distributions of  $\Delta x$  and  $\Delta y$  at a laboratory site: (a) frequency distribution of  $\Delta x$ ; (b) frequency distribution of  $\Delta y$ .

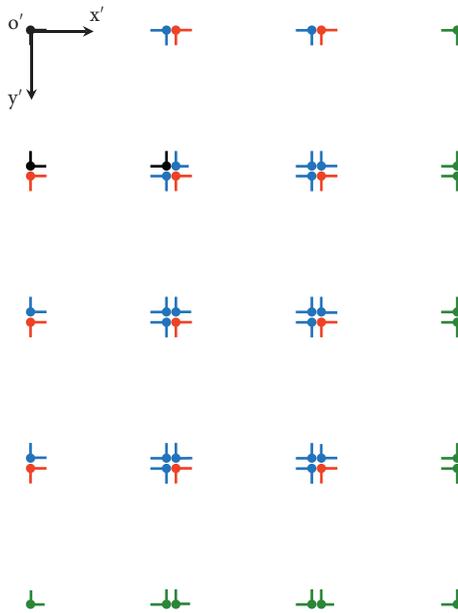


FIGURE 17: Calculation results' distribution of corner points.

TABLE 8: Results of the online detection evaluation.

Parameters	Average deviation (mm)	Standard deviation (mm)	Maximum deviation (mm)	Fraction of deviations < 0.5 mm (%)
$\Delta x$	0.06	0.16	0.64	89
$\Delta y$	0.18	0.14	0.54	94
$\sqrt{\Delta x^2 + \Delta y^2}$	0.33	0.19	0.84	84

TABLE 9: Comparison of online evaluation deviation and measurement deviation.

Parameters	Measurement deviation (mm)	Evaluation deviation (mm)	Error (%)
Average $\Delta x$	0.04	0.06	50
Average $\Delta y$	0.11	0.18	63
Maximum $\Delta x$	0.54	0.64	12
Maximum $\Delta y$	0.51	0.54	6

## 6. Conclusion and Future Research

This paper presents a robotic floor-tiling control method based on visual measurement feedback and finite-state machine. The self-developed FTR-II floor-tiling robot was used to conduct experiments at a laboratory site. Through the analysis of the experimental results, it can be concluded that

- (1) The control method of robot floor tiling proposed in this study is performed successfully
- (2) The deviations in the  $x$ - and  $y$ -coordinates are 0.04 mm and 0.11 mm, respectively, and 98.5% of all deviations are below 0.5 mm
- (3) The online evaluation system can realize the real-time online quality evaluation of floor tiling

However, the main limitation in this study is that the measurement method based on the limited FOV presupposes that the size of the tile is ideal. Dimensional deviation will be accumulated and will cause the tile position to exceed the permitted range. A possible effective solution is that the position information of multiple tiles is obtained at one time by adjusting the height of the camera. In addition, the detector in this paper can only identify the boundary of monochromatic tiles, and the edge detector will be brittle for patterned tiles.

The research of this paper may be not only applicable to control robotic floor-tiling scenes but also use for reference in the construction work under similar working conditions, such as curtain wall installation [42], brick masonry [43], and ceiling installation [44, 45]. In order to realize that the floor-tiling robot is able to perform actual tiling operations on real construction sites, more in-depth research direction may be developed on the research results of this paper, such as to solve the navigation and positioning problem of the mobile platform on construction site [46], to monitor the progress of the floor-tiling construction combined with BIM technology [47, 48], and to achieve precise positioning of tile laying with contact constraints through hybrid position/force control.

In future research, an edge detector should be developed to adapt to more general lighting conditions and tiles with different patterns. The authors will also develop a hybrid force/position control algorithm based on the visual measurement and force feedback. It may serve for the floor-tiling robot to achieve operation on real construction site and also develop the approaches to plan path and process monitoring on the construction site by the BIM and the mobile platform navigation and positioning technology.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This study was sponsored by the BUCEA Doctor Graduate Scientific Research Ability Improvement Project and Key Science and Technology Projects of China (KZ202110016024).

## References

- [1] U. Rembold, *Robot Technology and Applications*, CRC Press, Boca Raton, FL, USA, 2020.
- [2] B. Bayram and G. İnce, "Advances in robotics in the era of industry 4.0," *Industry 4.0: Managing the Digital Transformation*, Springer, Cham, Berlin, Germany, 2018, pp. 187–200, Springer Series in Advanced Manufacturing.
- [3] T. Bock, "The future of construction automation: technological disruption and the upcoming ubiquity of robotics," *Automation in Construction*, vol. 59, pp. 113–121, 2015.
- [4] M. Gharbia, A. Chang-Richards, Y. Lu, R. Y. Zhong, and H. Li, "Robotic technologies for on-site building construction: a systematic review," *Journal of Building Engineering*, vol. 32, Article ID 101584, 2020.
- [5] R. Bogue, "What are the prospects for robots in the construction industry," *Industrial Robot: International Journal*, vol. 45, 2018.
- [6] M. Gifftthaler, T. Sandy, and K. Dörfler, "Mobile robotic fabrication at 1:1 scale: the in situ fabricator," *Automation in Construction*, vol. 1, no. 3, 2017.
- [7] O. H. Rosenlund, *Mobile Drilling Robot-A Case Study of the Effects on the Construction Site*, Norges teknisk-naturvitenskaplige universitet, Trondheim, Norway, 2017.
- [8] T. Tsuruta, K. Miura, and M. Miyaguchi, "Mobile robot for marking free access floors at construction sites," *Automation in Construction*, vol. 107, Article ID 102912, 2019.
- [9] S. Samarakoon, M. Muthugala, and M. Elara, "Toward Obstacle-specific Morphology for a Reconfigurable Tiling Robot," *Journal of Ambient Intelligence and Humanized Computing*, 2021.
- [10] S. Samarakoon, M. Muthugala, A. Le, and M. R. Elara, "Toward complete area coverage of a reconfigurable tiling robot by following obstacle shape," *Complex & Intelligent Systems*, vol. 2, 2021.
- [11] Y.-C. Fang and R.-J. Dzung, "Accelerometer-based fall-potential detection algorithm for construction tiling operation," *Automation in Construction*, vol. 84, pp. 214–230, 2017.
- [12] N. King, M. Bechthold, A. Kane, and P. Michalatos, "Robotic tile placement: tools, techniques and feasibility," *Automation in Construction*, vol. 39, pp. 161–166, 2014.
- [13] G. Biswas, A. Bhattacharya, and R. Bhattacharya, "Occupational health status of construction workers: a review,"

- International Journal of Medical Science and Public Health*, vol. 6, no. 4, pp. 669–675, 2017.
- [14] P. F. Tunji-Olayeni, A. O. Afolabi, and O. I. Okpalamoka, “Survey dataset on occupational hazards on construction sites,” *Data in brief*, vol. 18, pp. 1365–1371, 2018.
- [15] D. Apostolopoulos, H. Schempf, and J. West, “Mobile robot for automatic installation of floor tiles,” in *Proceedings of IEEE International Conference on Robotics and Automation*, Minneapolis, MN, USA, vol. 4, IEEE, April 1996.
- [16] R. Navon, “Process and quality control with a video camera, for a floor-tilling robot,” *Automation in Construction*, vol. 10, no. 1, pp. 113–125, 2000.
- [17] M. K. A. Khan, K. I. Saharuddin, I. Elamvazuthi, and P. Vasant, “A semi-automated floor tiling robotic system,” in *Proceedings of the 2011 IEEE Conference on Sustainable Utilization and Development in Engineering and Technology (STUDENT)*, October 2011.
- [18] X. Li, C. Sun, W. Cheng, X. Jiang, and Y. Liu, “Adaptive vision-based control for robotic tiling with uncalibrated cameras and limited FOV,” in *Proceedings of the 2019 IEEE 15th International Conference on Control and Automation (ICCA)*, pp. 168–173, IEEE, Edinburgh, UK, July 2019.
- [19] T. Liu, H. Zhou, H. Lan, Y. Li, Q. Wang, and X. Che, “Robotic floor-tiling platform system design,” in *Proceedings of the IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, pp. 5269–5274, IEEE, Lisbon, Portugal, October 2019.
- [20] T. Y. Liu, H. X. Zhou, and Y. N. Du, “Measuring and positioning system design of robotic floor tiling,” in *Proceedings of the International Symposium Automation and Robotics in Construction*, vol. 36, pp. 716–721, IAARC Publications, Banff Alberta, Canada, May 2019.
- [21] V. Tlach, M. Cisar, I. Kuric, and I. Zajačko, “Determination of the industrial robot positioning performance,” *MATEC Web of Conferences*, vol. 137, Article ID 01004, 2017.
- [22] P. Božek, “Robot path optimization for spot welding applications in automotive industry,” *Technical Gazette*, vol. 20, no. 5, pp. 913–917, 2013.
- [23] E. Golkar and A. S. Prabuwo, “Vision based length measuring system for ceramic tile borders,” *Procedia Technology*, vol. 11, pp. 771–777, 2013.
- [24] Z. Hocenski, S. Vasilic, and V. Hocenski, “Improved canny edge detector in ceramic tiles defect detection,” in *Proceedings of the IECON 2006-32nd Annual Conference on IEEE Industrial Electronics*, pp. 3328–3331, IEEE, Paris, France, November 2006.
- [25] S. M. Emam and S. A. Sayyedbarzani, “Dimensional deviation measurement of ceramic tiles according to ISO 10545-2 using the machine vision,” *International Journal of Advanced Manufacturing Technology*, vol. 100, no. 5, pp. 1405–1418, 2019.
- [26] E. Golkar, A. Patel, L. Yazdi, and A. S. Prabuwo, “Ceramic tile border defect detection algorithms in automated visual inspection system,” *Journal of American Science*, vol. 7, no. 6, pp. 542–550, 2011.
- [27] H. Deng, H. Hong, D. Luo, Y. Deng, and C. Su, “Automatic indoor construction process monitoring for tiles based on BIM and computer vision,” *Journal of Construction Engineering and Management*, vol. 146, no. 1, Article ID 04019095, 2020.
- [28] K.-L. Lin and J.-L. Fang, “Applications of computer vision on tile alignment inspection,” *Automation in Construction*, vol. 35, pp. 562–567, 2013.
- [29] L. Roveda, N. Castaman, P. Franceschi, S. Ghidoni, and N. Pedrocchi, “A control framework definition to overcome position/interaction dynamics uncertainties in force-controlled tasks,” in *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA)*, August 2020.
- [30] L. Roveda, N. Castaman, S. Ghidoni et al., “Human-robot cooperative interaction control for the installation of heavy and bulky components,” in *Proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, October 2018.
- [31] M. Saleh, Y. Esa, and A. Mohamed, “Centralized control for DC microgrid using finite state machine,” in *Proceedings of the 2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, IEEE, Washington, DC, USA, April 2017.
- [32] D. C. Conner and J. Willis, “Flexible navigation: finite state machine-based integrated navigation and control for ROS enabled robots,” in *Proceedings of the SoutheastCon 2017*, pp. 1–8, IEEE, Concord, NC, USA, April 2017.
- [33] Y. Wang, Z. Sun, and Z. Chen, “Energy management strategy for battery/supercapacitor/fuel cell hybrid source vehicles based on finite state machine,” *Applied Energy*, vol. 254, Article ID 113707, 2019.
- [34] M. Zhang, N. Li, A. Girard, and I. Kolmanovsky, “A finite state machine based automated driving controller and its stochastic optimization,” in *Proceedings of the Dynamic Systems and Control Conference*, October 2017, Article ID 58288: V002T07A002.
- [35] J. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
- [36] W. Rong, Z. Li, and W. Zhang, “An improved CANNY edge detection algorithm,” in *Proceedings of the 2014 IEEE international conference on mechatronics and automation*, pp. 577–582, IEEE, Beijing, China, November 2014.
- [37] R. Biswas and J. Sil, “An improved canny edge detection algorithm based on type-2 fuzzy sets,” *Procedia Technology*, vol. 4, pp. 820–824, 2012.
- [38] M. Fang, G. X. Yue, and Q. Yu, “The study on an application of otsu method in canny operator,” in *Proceedings of the 2009 International Symposium on Information Processing (ISIP 2009)*, vol. 109, Academy Publisher, Huangshan, China, August 2009.
- [39] J. Matas, C. Galambos, and J. Kittler, “Robust detection of lines using the progressive probabilistic hough transform,” *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 119–137, 2000.
- [40] Tiles-Part, “Ceramic. 2: “dermination of dimensions and surface quality” national standart corporation,” ISO, Geneva, Switzerland, SNI ISO, vol. 105452, 2010.
- [41] Iso 13006/2012, “Cramic tiles–definitions, classification, characteristics and marking, annex K and L,” pp. 38–43, ISO, Geneva, Switzerland, 2012, ISO 13006/2012.
- [42] A. K. Ali, O. J. Lee, and H. Song, “Robot-based facade spatial assembly optimization,” *Journal of Building Engineering*, vol. 33, Article ID 101556, 2021.
- [43] S. Parascho, I. X. Han, A. Beghini et al., “LightVault: a design and robotic fabrication method for complex masonry structures,” *Advances in Architectural Geometry*, vol. 25, 2021.
- [44] S. Lee, J. Lee, C. Han, K. Lee, and S. Lee, “Human Robot Cooperative Control and Task Planning for a Glass Ceiling Installation Robot,” in *The 25th International Symposium on Automation and Robotics in Construction*, Vilnius, Lithuania, June 2008.

- [45] M. Gharbia, A. Chang-Richards, Y. Lu, R. Y. Zhing, and H. Li, "Robotic technologies for on-site building construction: a systematic review," *Journal of Building Engineering*, vol. 32, Article ID 101584, 2020.
- [46] R. Pirník, M. Hruboš, D. Nemeč, T. Mravec, and P. Božek, "Integration of inertial sensor data into control of the mobile platform," in *Proceedings of the Federated Conference on Software Development and Object Technologies*, pp. 271–282, Springer, Cham, Žilina, Slovakia, November 2015.
- [47] X. Li, J. Xu, and Q. Zhang, "Research on construction schedule management based on BIM technology," *Procedia engineering*, vol. 174, pp. 657–667, 2017.
- [48] P. Martinez, B. Barkokebas, F. Hamzeh, M. Al-Hussein, and R. Ahmad, "A vision-based approach for automatic progress tracking of floor paneling in offsite construction facilities," *Automation in Construction*, vol. 125, Article ID 103620, 2021.