Ensuring Area Coverage and Safety of a Reconfigurable Staircase Cleaning Robot

M. A. VIRAJ J. MUTHUGALA, (Member, IEEE), S. M. BHAGYA P. SAMARAKOON, (Student Member, IEEE), PRABAKARAN VEERAJAGADHESWAR, AND MOHAN RAJESH ELARA

Engineering Product Development Pillar, Singapore University of Technology and Design, Singapore 487372

Corresponding author: M. A. Viraj J. Muthugala (viraj_jagathpriya@sutd.edu.sg)

This research is supported by the National Robotics Programme under its Robotics Enabling Capabilities and Technologies (Funding Agency Project No. 192 25 00051), National Robotics Programme under its Robot Domain Specific (Funding Agency Project No. 192 22 00058), National Robotics Programme under its Robotics Domain Specific (Funding Agency Project No. 192 22 00108), and administered by the Agency for Science, Technology and Research.

ABSTRACT Multistorey buildings are rapidly constructing with the current world development. Cleaning of these buildings is a major concern due to several issues such as scarcity of labor. Cleaning robots are developed to cope with these issues. Among these cleaning robots, staircase cleaning robots play a vital role in a multistorey building. Therefore, this paper proposes a control strategy for a reconfigurable staircase cleaning robot to ensure area coverage and safety. A staircase cleaning robot should particularly maintain its orientation parallel to the riser while maintaining a proper clearance with the riser when performing the sideways movement on a thread to ensure safety and coverage. Two Fuzzy Logic Systems (FLSs) have been developed to correct the heading and the clearance distance during sideways movements. Precise measurements on heading error and clearance distance are essential for the decision-making process of the proposed control strategy. Thus, a vision-based perception mechanism has been developed to accurately perceive these essential measurements. The proposal of the control strategy to correct the heading and the clearance distance, and the development of the vision-based perception mechanism are the major contributions of the work. Experiments were conducted by deploying the developed robot in a typical staircase considering different conditions. The experiment results validate that the proposed method can ensure coverage and safety by properly maintaining the heading and clearance of the robot with respect to a riser.

INDEX TERMS Area coverage, cleaning robotics, reconfigurable robotics, robot safety, staircase cleaning.

I. INTRODUCTION

The world population is growing, and multistory building infrastructures are constructed to fulfill the living space requirements [1]. Cleaning is one of the frequent routine activities essential for maintaining the living standard of these building infrastructures. Usually, cleaning is a tedious, repetitive, and time-consuming task that requires intensive labor. The tedious and repetitive nature hampers the cleaning efficiency of humans. In addition to that, spending time on cleaning becomes difficult for most people due to socioeconomic behavior. On the other hand, human labor is expensive and scarce.

Robotics solutions have been introduced to solve the issues associated with human labor-based cleaning of building infrastructures [2]. These cleaning robotics solutions extend from domestic to industrial premises, including application domains such as floor cleaning [3], [4], facades cleaning [5], [6], wall cleaning [7], furniture cleaning [8], and duct cleaning [9]. Especially cleaning robots are immensely useful in combating the spread of infectious diseases such as COVID-19 since the robots can safely facilitate the cleaning and disinfection of vulnerable places such as quarantine facilities and hospitals [10], [11].

Staircases could be seen in most of the modern-day building infrastructure as a result of multistory nature. Thus, the attention of cleaning robotics research has been drifted toward the development of staircases cleaning robots [12]. However, the design and development of a staircase cleaning
The objective of developing a reconfigurable stair traversing robot is to perform effective area coverage on various staircase structures. MUlti purpose StairCase Accessing Reconfigurable Robot (MUSCARR) was developed under the principle of modular mobility concept with a linear actuation mechanism. The robot consists of three modules named as Front Leg Module (FLM), Center Body Module (CBM), and Back Leg Module (BLM) which all connected together through two linear actuators. All together the overall dimension will be $280 \text{ mm} \times 330 \text{ mm} \times 470 \text{ mm}$. A cleaning robot intended for staircase cleaning should have proper control for safe operation since the robot has to move on a narrow tread space. If proper controlling schemes are not utilized, a robot could fall off or collide with the stair risers. However, the majority of the staircase cleaning robots discussed above are mainly focused on developments of more effective climbing mechanisms and less attention has been paid in this niche. The work [24] proposed a sensory system and a control scheme to avoid the possible danger of falling off and collisions. Bump sensors are placed on the front of the robot to detect the collision. The rear is fixed with proximity sensors to detect cliffs. Furthermore, a front caster wheel is installed with a contact detecting mechanism. The robot could detect the possibility of falling off and the collision using the feedback of these sensors. Subsequently, the control scheme uses a rule-based decision-making approach to avoid danger; e.g., if rear proximity sensors detected a cliff during a backward movement robot stops the current action and moves forward. A staircase cleaning robot should move parallel to a stair riser while maintaining a proper distance gap with the riser to ensure the area coverage performance and the robot’s safety. However, the scheme proposed in the cited work lacks this ability.

This paper proposes a control strategy for a reconfigurable staircase cleaning robot to improve its area coverage performance and safety. The robot’s reconfigurable design allows climbing up and down movements on a staircase and side movements required for the cleaning. The proposed control strategy is designed to maintain the robot’s side movements on treads parallel to the next riser while upholding a proper distance gap. A vision-based perception module has been introduced to estimate the distance to the next riser and the robot’s orientation with respect to the riser. Two Fuzzy Logic Systems (FLSs) have been deployed to the robot for controlling actions required for maintaining the orientation and the distance gap based on these estimations. The paper contributes to the state of the art by introducing a control strategy and a perception module to ensure area coverage and safety of a staircase cleaning robot. Section II discusses the robot hardware platform including mechanical, software, and system design. The motive for the controller, the perception module, and the methodology is explained in Section III. Results and discussion of the proposed system are given in IV. Concluding remarks are given in Section V.

II. ROBOT PLATFORM

The main objective of developing a reconfigurable stair traversing robot is to perform effective area coverage on various staircase structures. MUlti purpose StairCase Accessing Reconfigurable Robot (MUSCARR) was developed under the principle of modular mobility concept with a linear actuation mechanism. The robot consists of three modules named as Front Leg Module (FLM), Center Body Module (CBM), and Back Leg Module (BLM) which all connected together through two linear actuators. All together the overall dimension will be $280 \text{ mm} \times 330 \text{ mm} \times 470 \text{ mm}$. A cleaning robot intended for staircase cleaning should have proper control for safe operation since the robot has to move on a narrow tread space. If proper controlling schemes are not utilized, a robot could fall off or collide with the stair risers. However, the majority of the staircase cleaning robots discussed above are mainly focused on developments of more effective climbing mechanisms and less attention has been paid in this niche. The work [24] proposed a sensory system and a control scheme to avoid the possible danger of falling off and collisions. Bump sensors are placed on the front of the robot to detect the collision. The rear is fixed with proximity sensors to detect cliffs. Furthermore, a front caster wheel is installed with a contact detecting mechanism. The robot could detect the possibility of falling off and the collision using the feedback of these sensors. Subsequently, the control scheme uses a rule-based decision-making approach to avoid danger; e.g., if rear proximity sensors detected a cliff during a backward movement robot stops the current action and moves forward. A staircase cleaning robot should move parallel to a stair riser while maintaining a proper distance gap with the riser to ensure the area coverage performance and the robot’s safety. However, the scheme proposed in the cited work lacks this ability.

This paper proposes a control strategy for a reconfigurable staircase cleaning robot to improve its area coverage performance and safety. The robot’s reconfigurable design allows climbing up and down movements on a staircase and side movements required for the cleaning. The proposed control strategy is designed to maintain the robot’s side movements on treads parallel to the next riser while upholding a proper distance gap. A vision-based perception module has been introduced to estimate the distance to the next riser and the robot’s orientation with respect to the riser. Two Fuzzy Logic Systems (FLSs) have been deployed to the robot for controlling actions required for maintaining the orientation and the distance gap based on these estimations. The paper contributes to the state of the art by introducing a control strategy and a perception module to ensure area coverage and safety of a staircase cleaning robot. Section II discusses the robot hardware platform including mechanical, software, and system design. The motive for the controller, the perception module, and the methodology is explained in Section III. Results and discussion of the proposed system are given in IV. Concluding remarks are given in Section V.
(L × W × H). The dimensions of the robot were designed to match the typical staircases. The detailed breakdown of each component is depicted in Fig. 1.

A. MECHANICAL DESIGN

The developed robot can perform both vertical (ascending and descending) and horizontal motions (left and right) in the staircase. Similarly, the mechanical components were split into vertical components which support the vertical motion, and horizontal components that aid in sideways movements are shown in Fig. 2. As mentioned above, the robot was developed as several modules to achieve vertical motion. Thus, these three modules are connected serially through lead screw-based linear mechanical modules which are (LSM-1, and LSM-2). Both these modules are connected at the two edges of the CBM in the adverse direction. Each LSM module consists of a lead screw with the pitch size of 5mm, a moving nut connected with a mounting plate (MP), a stepper motor with a coupler (SM), and an electromechanical brake (EMB). The mounting plate MP1 and MP2 will be connected to the front leg module and back leg module respectively. When it is necessary, the stepper motor SM1 and SM2 will get activated to actuate the mounting plate in both directions which eventually shifts the modules FLM and BLM vertically. To fix or detain this vertical motion, the system used electromechanical break EMB1 and EMB2. In terms of operation, the above-mentioned components are mostly used in vertical operations. Since the developed robot operates in a batch-wise manner while traversing on a staircase, each translational action is defined to represent the module movements as shown in Fig. 3 (right). For instance, the FLM module up-down motion is defined as FL-Positive, and FL-Negative respectively. Similarly, other module’s actions are defined as CB-Positive, CB-Negative, and LB-Positive, and LB-Negative. For FL-Positive and FL-Negative motion the LSM1 is operated, BL-Positive and BL-Negative motion the LSM2 is operated, and CB-Positive and CB-Negative motion both LSM1 and LSM2 are operated.

B. SYSTEM DESIGN

Regarding the system design, the system is classified into components as local, perception, and global that executes the decision, sensing, and control actions for the robot. Fig. 4 shows the system architecture of the developed robot. Arduino-Mega based local control structure is developed that receives the decisional commands and passes the action commands to the motor drives. One Arduino Nano controller was used to control the stepper motor driver which requires uninterrupted pulse signals. The Arduino nano gets the signals from the Mega controller in a digital form. Before operating the stepper motor, the system requires to release the electromagnetic brake. In this regard, a relay circuit that was controlled by the Arduino Mega controller is deployed. Other local components that were connected to Arduino such as the DC motor driver pass the final motor primitives to the end effectors. Perceptional components are the critical components in executing our robot operation. Time of Flight (ToF) sensors are connected at the bottom of every single module. These sensors aid in the reconfiguration process of the robot. The local component information is collected by the Arduino Mega controller. On top of Arduino based perception sensor, a RealSense depth camera is mounted to identify the edge of each step to perform a flawless area coverage process. All this information from local and perceptional components transferred to the global component which is Nvidia Jetson Xavier industrial PC to process and generate high-level decisions. All these components are powered using a 24 V DC battery along with a 12 V step-down DC regulator.

C. SOFTWARE ARCHITECTURE

The software architecture is developed on top of the Robot Operating System (ROS) Melodic distribution. The subscription of image streams and depth values for the RealSense camera is mainly handled by ROS nodes. Furthermore, for basic robot localization and publish transitional, and locomotion commands are conducted with aid of ROS. ROS
also involved creating a bridge between global, perception, and local devices. All software layers are executed inside the Jetson Xavier platform which runs with the ubuntu 18.04 environment.

III. ENSURING AREA COVERAGE AND SAFETY

A. DESIGN CONSIDERATIONS FOR THE CONTROLLER

The main requirements of a staircase cleaning robot are that it should be capable of rise-up and climb down and tread cleaning. To improve the cleaning coverage the robot, need to move from one end to another end in each tread. The robot needs to maintain a proper distance and parallel with the riser when moving in the tread. Otherwise, the robot might fall from the staircase or collide with the riser. Fig. 5(a) shows a situation where the robot is not aligned with the step. The clearance distance between the step and the robot is depicted as \( d_c \). This distance needs to be maintained during the navigation in the tread to cover the whole area. If \( d_c \) is high, the robot will fall from the tread, and if \( d_c \) is trivial, the robot will collapse with the riser. The distance between the robot and the rise can be changed due to the friction of the robot and the tread. To overcome these issues, the robot should maintain a proper distance as \( d_c \). This distance maintenance should apply to all the treads to improve the area coverage. Furthermore, during the robot ascending and descending from the riser, the robot’s initial orientation could be changed. Orientation correction is done by changing angular velocity (\( \omega \)) and X-axis linear velocity component (\( v_x \)). Y-axis linear velocity is maintained as a constant to move sideways.

B. PERCEPTION MODULE

The main objective of this work is to perform flawless area coverage in a single staircase by moving the robot sideways (\( L_{_MV} \) and \( R_{_MV} \)) directions while ensuring safety. In order to achieve such an ability, a few key information needs to be extracted that includes the robot’s orientation error (\( \phi \)) and the perpendicular distance between the robot’s footprint and the edge of a riser (\( d_r \)). By focusing on these two parameters, the system can avoid the robot from toppling down while performing the area coverage. To extract these measurements precisely, the system requires an accurate perception system and framework. The presented framework is activated once the robot reaches the staircase. Since the focus of this paper is on controlled step cleaning, this study assumed the robot is already in a staircase to start the cleaning process.

1) IMAGE ACQUISITION

In this work, the intention is to use a depth camera to capture both RGB images and depth information. This information is captured by using Intel RealSense camera D3435i. The camera operates at a frequency of 90 Hz for providing the depth image with a resolution of 1280 × 720. The camera provides the RGB image with a resolution of 1920 × 1080 and 30 frames per second. The depth information is acquired using infrared stereo technology with a maximum measuring range up to 3 m and a minimum range of 0.3 m. The camera can operate in a field of view of 69° in the horizontal direction and 42° in the vertical direction as shown in Fig. 6. The camera is fixed in the front of the Front Leg Module (FLM), and the view of the camera is turned in such a way that the robot’s edge and the step edge can be clearly seen. Further, the camera is mounted a bit lower in order to avoid observing other unwanted edges on the robot which affects the overall edge detection algorithm. For the demonstration purpose, Fig. 7 shows the viewable area of the depth camera in a typical staircase.
2) FEATURE EXTRACTION

The two features that need to be identified are the edge of the robot platform and the edge of the riser. The edge of the riser could be easily identified by using colorized depth image where the difference between the current step and the descending step can clearly be observed due to its distance difference. The Hough Transform is a popular technique to detect any shape that can be represented in a mathematical form [25–27]. The simplest case of Hough transform is detecting straight lines. An edge detected binary image should be fed to the Hough Line detection method. Hence, the Canny edge detection algorithm [28], which can produce a binarized edge detected image, is first applied to the depth image to extract the edge by utilizing the change in color represented by depth. Then, the Hough Line detection algorithm proposed in [27] is applied to the edge detected image to extract the line of the step edge as the first feature.

Similarly, for the next feature, which is the robot’s edge, the same procedure cannot be utilized due to the depth between the robot’s edge and the step floor (thread) having very little depth difference. The system uses the RGB image to overcome this issue and passes it to Canny edge detection and Hough line detection algorithm to acquire the robot’s line. With these two lines, the clearance distance between them is calculated to the intersection point, and the heading error (φ) are calculated. The outcome of the line extraction situation is shown in Fig. 8.

After the extraction of these two lines, the lines are converted to a cartesian form. These two lines are considered skew lines since the lines are not exactly parallel to each other. The feature of reference is the line extracted from the staircase edge is referred as S1S2. Line S1S2 can be represented as in (1), where \( m_S \) is the gradient and \( C_S \) is Y-axis intercept. The line extracted from the robot edge detection is denoted as \( R_1R_2 \). Line \( R_1R_2 \) can be represented as in (2), where \( m_R \) is the gradient and \( C_R \) is Y-axis intercept.

\[
\begin{align*}
    y &= m_Sx + C_S \quad (1) \\
    y &= m_Rx + C_R \quad (2)
\end{align*}
\]

The midpoint of the robot is considered as \( R \) where the shortest distance from \( R \) to \( S_1S_2 \) is taken as \( d_c \). This distance is represented by the line \( RS \). The gradient of the line \( RS \) becomes \(-1/m_R \) since the line is perpendicular to \( R_1R_2 \). Then the equation of the line \( RS \) can be obtained as (3). The Y-intercept \( C_{RS} \) can be found by substituting the known coordinates of point \( R \). The coordinates of the intersection point, \( S \) can be calculated using (1) and (3) since the point lies on both lines (i.e., \( RS \) and \( S_1S_2 \)). Finally, the distance \( d_c \) can be obtained by taking the Euclidean distance between the points, \( R \) and \( S \). The orientation error (φ), which is the angle between the lines \( S_1S_2 \) and \( R_1R_2 \) can be calculated from (4). The overall flow of the feature extraction process is given in Algorithm 1.

\[
\begin{align*}
    y &= \frac{1}{m_R}x + C_{RS} \quad (3) \\
    \phi &= \text{atan2}(\frac{m_S - m_R}{1 + m_Rm_S}) \quad (4)
\end{align*}
\]

**Algorithm 1: Feature Extraction**

**output**: \( \phi \) and \( d_c \);

**grab RGB_frame**;

**grab Depth_image**;

\((m_S, C_S) = \text{Step_detection}(Depth_image);\)

\((m_R, C_R) = \text{Robot_detection}(RGB_frame);\)

**calculate** \( \phi \) and \( d_c \) using (1 - 4);

**Return** \( \phi \) and \( d_c \);

**Function Step_detection (Depth_image):**

\( edge = \text{Canny edge detection}(Depth_image); \)

\( line = \text{Hough line detection}(edge); \)

\( m_S, C_S = \text{gradient and intercept of line}; \)

**Return** \( m_S, C_S \);

**end function**

**Function Robot_detection (RGB_frame):**

\( edge = \text{Canny edge detection}(RGB_frame); \)

\( line = \text{Hough line detection}(edge); \)

\( m_R, C_R = \text{gradient and intercept of line}; \)

**Return** \( m_R, C_R \);

**end function**
C. HEADING AND CLEARANCE CORRECTION

The control actions of the robot required to maintain the robot parallel to the riser with a proper clearance during the sideways movements are determined by two Fuzzy Logic Systems (FLSs). FLSs have been used in this regard due to their ability of effectively coping with inaccuracies of sensor information where the sensor information obtained from the perception module is uncertain [29], [30]. In addition to that, the underlying dynamics of the robot are difficult to model in this application due to the reconfigurable ability of the robot. In contrast, FLSs have the ability to model any complex process or system without the exact knowledge of the underlying dynamics of the process or the system [31], [32].

1) HEADING CORRECTION

The architecture of the FLS proposed for Heading Correction (FLS-HC) is depicted in Fig. 9. The inputs of the FLS-HC are the alignment error at the time step, $t$ (i.e., $\phi$), and the change of alignment error with respect to the previous time step (i.e., $\delta \phi$). Here, $\phi$ is retrieved from the perception module as explained in section, and $\delta \phi(t) = \phi(t) - \phi(t - 1)$. These two inputs are fuzzified in the fuzzification layer using the fuzzy membership functions given in Fig. 10(a). $\mu_{\phi}(\phi)$ and $\mu_{\delta \phi}(\delta \phi)$ are the fuzzified values corresponding to the inputs.

The rule base contains a set of linguistic rules that maps the input fuzzy sets with output fuzzy sets during the inferencing stage. The rule base of the FLS-HC has been defined based on expert knowledge such that the robot’s actions counter the error of the robot’s heading. For example, suppose the robot’s alignment has deviated toward the left from the goal. In that case, the angular velocity of the robot is set in the counter-clockwise direction to correct the heading error, where the robot is attempting to rotate right. Vice versa of this behavior is also applicable. The amount of angular velocity should be positively correlated with the magnitude of the heading error. In addition to that, the change of heading error is analyzed to minimize the oscillations and overshoots. For example, if a heading error change indicates a quick correction, the robot’s angular velocity is reduced to avoid possible oscillations due to overshooting. The rule base of the FLS-HC is given in Table 1. The fuzzy operators, t-norm and t-conorm have been considered as min and max. Thus, the firing strength of the $k$th fuzzy rule of the FLS-HC, $F_k$, is evaluated as in (5).

$$F_i = \min\{\mu_{\phi}(\phi), \mu_{\delta \phi}(\delta \phi)\}$$  \hspace{1cm} (5)

The output of the FLS-HC is the reference angular velocity of the robot (i.e., $\omega$). The corresponding output membership function is given in Fig. 10(b). The output fuzzy sets are clipped by the respective firing strength of the rule based on Mamdani implication method [33]. The resultant fuzzy consequents for the $k$th fuzzy rule of the FLS-HC, $\mu_{\omega_k}^i(\omega)$ is given in (6). These fuzzy consequents are aggregated to a single fuzzy set as given in (7). Here, the fuzzy max operator is used for the aggregation.

$$\mu_{\omega_k}^i(\omega) = F_k \land \mu_{\omega_k}^i(\omega)$$  \hspace{1cm} (6)

$$\mu_{\omega^i}(\omega) = \mu_{\omega_k}^i(\omega) \lor \mu_{\omega_k}^i(\omega) \lor \ldots \lor \mu_{\omega_k}^i(\omega)$$  \hspace{1cm} (7)

A deterministic crisp output is required for controlling the angular velocity of the robot. In this regard, the aggregated fuzzy set is defuzzified in the defuzzification layer. The center of area method is used for the defuzzification, and the crisp output is obtained as in (8). The decision surface of the FLC-HC is given in Fig. 11.

$$\omega^* = \frac{\int_{\omega} \omega \mu_{\omega^i}(\omega) d\omega}{\int_{\omega} \mu_{\omega}(\omega) d\omega}$$  \hspace{1cm} (8)

2) CLEARANCE CORRECTION

The architecture of the FLS designed for Clearance Correction (FLS-CC) is depicted in Fig. 12. The inputs of the FLC-CC are the error of the current clearance (i.e., $e_c$) and the
change of error with respect to the previous time step (i.e., $\delta e_c(t)$), where $\delta e_c(t) = e_c(t) - e_c(t-1)$. Current clearance error, $e_c$, is evaluated as in (9) where $R_c$ is the preferred clearance.

The fuzzification layer of the FLC-CC fuzzifies these inputs to $\mu_{e_c}$ and $\mu_{\delta e_c}$ by using the membership function given in Fig. 13(a).

$$e_c = R_c - d_c$$  \hspace{1cm} (9)

The output of the FLC-CC is the linear velocity of the robot in the direction of the robot’s X-axis (i.e., $v_x$). The output membership functions of the FLC-CC are given in Fig. 13(b). The input fuzzy sets are mapped to the output fuzzy sets using the rule base given in Table 2. The rule base of the FLS-CC has been defined such that the robot’s actions are to correct the clearance error. For example, if the robot’s closer to the riser, then the robot should move backward, where there should be a backward linear velocity or vice versa. The amount of this linear velocity should be dependent on the magnitude of the error, where the backward velocity should be higher if the robot is closer than the little closer state or vice versa. Furthermore, the change of clearance is to predict the clearance correction to minimize the oscillations caused by overshooting. The firing strength of the $l$th fuzzy rule of the FLC-CC, $R_l$, can be formulated as in (10) considering min and max as the fuzzy t-norm and t-conorm operators, respectively. The Mamdani implication method yields to the fuzzy consequents corresponding to this rule, $\mu_{v'_x}(v_x)$ as in (11) since the corresponding output fuzzy sets are clipped by the respective firing strength of the rule.

$$R_l = \min\{\mu_{v'_x}(v_x), \mu_{\delta v'_x}(\delta v_x)\}$$ \hspace{1cm} (10)

$$\mu_{v'_x}(v_x) = R_l \land \mu_{v_x}(v_x)$$ \hspace{1cm} (11)

The fuzzy consequents are aggregated into a single set as in (12) considering the fuzzy max operator for the aggregation. Finally, a deterministic crisp output is obtained by (13) using the center of area method for the defuzzification. The decision surface of the FLC-CC is depicted in Fig. 14.

$$\mu_{v_x}(v_x) = \mu_{v'_x}(v_x) \lor \mu_{v'_x}(v_x) \lor \ldots \mu_{v'_x}(v_x)$$ \hspace{1cm} (12)

$$v_x^* = \frac{\int v_x \mu_{v'_x}(v_x) dv_x}{\int \mu_{v'_x}(v_x) dv_x}$$ \hspace{1cm} (13)

3) COORDINATION BETWEEN FLS-HC AND FLS-CC

The overall coordinated flow of the heading and clearance correction algorithms is given in Algorithm 2. Both FLS-HC...
and FLS-CC are called in each run of the control loop. The time for inferencing the FLSs is negligible, where both the FLSs are executed without a considerable time difference. After determining the necessary control actions in each time step, \( t \) (i.e., \( v_x \) and \( \omega \)), the motion commands are passed to the locomotion unit to execute the actions. This process is continued until a sideway movement is completed.

Algorithm 2: Heading and Clearance Correction

```plaintext
Define \( R_c \);
Define \( v_Y \);
while sideway movement do
    \( d_c(t), \phi(t) = \text{Algorithm 1}() \);
    \( e_c(t) = R_c - d_c(t) \);
    \( \delta \phi(t) = \phi(t) - \phi(t - 1) \);
    \( \delta e_c(t) = e_c(t) - e_c(t - 1) \);
    \( v_x = \text{FLS-CC}(e_c(t), \delta e_c(t)) \);
    \( \omega = \text{FLS-HC}(\phi(t), \delta \phi(t)) \);
    Send motion commands \((v_x, v_Y, \omega)\);
    \( t++ \);
end while
```

IV. RESULTS AND DISCUSSION

A. EXPERIMENTAL SETUP

Experiments were conducted by deploying the robot to a typical staircase to validate the performance of the proposed control strategy. Four test cases with different characteristics were chosen in this regard. The preferred clearance distance, \( R_c \), was configured to 25 mm based on the dimensions of the staircase. The linear velocity of the robot for sideway movement, \( v_y \), was considered as 0.03 m/s.

B. RESULTS

The arrangements of the test cases and a set of snapshots taken during the robot movements are given in Fig. 15. The corresponding variations of the robot’s heading error (i.e., \( \phi \)) and clearance error (\( e_c \)) are plotted in Fig. 16. The view of the thread and riser through the perception module corresponding to the snapshots are given in Fig. 17.

In case ‘a’, the robot was initially placed with no noteworthy heading and clearance error (i.e., the robot was roughly parallel to the riser, and the clearance was \( R_c \)), as shown in Fig. 15(a). Then, the sideway movement of the
robot was commenced. The variations of the heading error and clearance error, in this case, are depicted in Fig. 16(a). The variations of the errors show that the ability of the control strategy in maintaining the heading and clearance properly during the sideways movement. Root Mean Square Errors (RMSEs) of heading and clearance were 0.19° and 4.5 mm, respectively. In addition to that, the robot movement (given in Fig. 15(a) and Fig. 17(a)) also confirms this behavior.

A scenario of the robot initiates the sideways movement with an initial heading error was considered case ‘b’. Initially, the heading error was 13. The error was decreased and converged to around zero (see Fig. 16(b)). Moreover, the proposed FLC-HC could rectify the initial heading error and subsequently manage an overall error (RMSE is 3.6°). In addition to that, the FLC-CC managed the clearance distance without a considerable error (RMSE of $d_C$ is 3.6 mm) while coping with the FLC-HC’s actions to correct...
the snapshots of the robot shown in Fig. 15(b) and Fig. 17(b).

In case ‘c’, the robot’s sideways movement was initiated with an initial error in the clearance distance with no initial error in heading, as shown in Fig. 15(c). The variations of $\varphi$ and $e_c$ during the robot’s moment of this case are plotted in Fig. 16(c). Initially, $e_c$ was -26 mm, and the actions of the FLC-CC were successful in reducing $e_c$, yielding to a lower RMSE (6.4 mm in here). Furthermore, the FLC-HC managed $\varphi$ around zero with trivial variations. The robot movement shown through snapshots in Fig. 15(c) and Fig. 17(c) are in compliance with the error variations. Thus, the proposed control strategy successfully corrected the initial clearance error while maintaining a lower heading error.

Case ‘d’ represents a situation where the robot is placed with initial errors in both heading and clearance distance, as shown in Fig. 15(d). The view through the perception module during this case is given in Fig. 17. According to the variation of $\varphi$ and $e_c$ (given in Fig. 16(d)), the proposed control strategy suppressed both errors through the actions. RMSEs were 3.14° and 7.13 mm for heading and clearance distance, respectively. The movement of the robot also indicates the same. Moreover, the proposed FLSs are performed well even errors are presented in both heading and clearance distance.

Overall, the proposed control strategy was capable of maintaining the clearance distance between the robot and a riser in a defined value while maintaining the robot parallel to the riser during sideways movements. The considered test cases cover most of the probable cases often encountered by the robot during the coverage. For example, initial errors on heading and clearance distance could often be encountered by the robot after climbing a step using its reconfiguration. Thus, the correction of heading and clearance distance is essential for the robot in such situations. Otherwise, the robot may collide with the riser or might topple up the step, which impedes safety. Furthermore, area coverage performance on a thread also depends on the proper management of these two factors. Therefore, managing these two factors could ensure the area coverage and safety of a staircase cleaning robot. As such, it can be concluded from the experimental observations that the proposed control strategy is beneficial in ensuring the area coverage and safety of a reconfigurable staircase cleaning robot.

V. CONCLUSION

Staircase cleaning is demanding for multistory buildings. Robots have been introduced for staircase cleaning to resolve the shortcomings of conventional methods based on human labor. However, less work has been conducted on developing robots targeted for staircase cleaning compared to the other application domains such as floor cleaning due to the complexity of climbing on steps. The robots and controllers designed for application domains such as floor and facades cleaning (e.g., [3]–[6]) could not apply to the context staircase cleaning due to their inability to cope with steps. On the other hand, the existing work on staircase cleaning robots (i.e., [15]–[19], [22]–[24]) has not focused on developing control strategies to correct alignment and clearance distance with a riser during the sideline movements. This feature is essential for ensuring the safety and area coverage performance of a staircase cleaning robot.

This paper proposed a control strategy for a reconfigurable staircase cleaning robot to ensure area coverage and safety. The robot’s ability to maintain the heading parallel to the next riser while upholding a proper clearance distance during the sideways movement is crucial for safety and coverage. Thus, two controllers have been designed to correct the robot heading and the clearance distance concurrently. Fuzzy logic has been utilized in this regard due to its proven ability to cope with uncertainties. Furthermore, a vision-based perception mechanism has been deployed to the robot to perceive the required inputs for the controllers, such as current heading error and the clearance distance.

Experiments have been conducted considering a set of test cases covering most of the possible encounters on typical staircase cleaning processes. According to the experimental results, the proposed control strategy is effective in keeping errors in heading and clearance distance at an acceptable level for safe and efficient staircase cleaning. Therefore, the work proposed in this paper would be beneficial in improving the productivity of staircase cleaning robots. Energy efficiency is a crucial requirement for a cleaning robot, and explorations on energy-efficient optimum coverage planning methods for staircase cleaning robots are proposed for future work. In addition to that, the method is to be extended for coping with different staircase configurations such as spiral staircases where step shape is not rectangular.

REFERENCES

[1] C. Thuesen and N. Koch-Ørvod, “Construction transformation,” in Proc. Sustain Conf. Creating Technol. Sustain. Soc., vol. 2018, p. A1.
[2] M. Hazas, J. Albrecht, A. J. Brush, and M. Hazas, “Roombas and landroids: Do domestic service robots save energy?” IEEE Pervasive Comput., vol. 20, no. 2, pp. 54–57, Apr. 2021.
[3] P. Veerajagadheswar, M. R. Elara, T. Pathmakumar, and V. Ayyalusami, “A tilting-theoretic approach to efficient area coverage in a tetris-inspired floor cleaning robot,” IEEE Access, vol. 6, pp. 35260–35271, 2018.
[4] S. M. B. P. Samarakoona, M. A. V. J. Muthugala, A. V. Le, and M. R. Elara, “Toward complete area coverage of a reconfigurable tiling robot by following obstacle shape,” Complex Intell. Syst., vol. 7, no. 2, pp. 741–751, Apr. 2021.
[5] H. Chae, G. Park, J. Lee, K. Kim, T. Kim, H. S. Kim, and T. Seo, “Facade cleaning robot with manipulating and sensing devices equipped on a gondola,” IEEE/ASME Trans. Mechatronics, vol. 26, no. 4, pp. 1719–1727, Aug. 2021.
[6] M. A. V. J. Muthugala, M. Vega-Heredia, A. Vengades, G. Srinavasa, and M. R. Elara, “Design of an adhesion-aware facade cleaning robot,” in Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS), Dec. 2019, pp. 1441–1447.
[7] M. A. V. J. Muthugala, S. M. B. P. Samarakoona, M. M. Rayguru, B. Ramalingam, and M. R. Elara, “Wall-following behavior for a disinfection robot using type 1 and type 2 fuzzy logic systems,” Sensors, vol. 20, no. 16, p. 4445, Aug. 2020.
[8] S. Elliott and M. Cakmak, “Robotic cleaning through dirt rearrangement planning with learned transition models,” in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2018, pp. 1623–1630.
[9] F. Ito, T. Kawaguchi, M. Kamata, Y. Yamada, and T. Nakamura, “Proposal of a peristaltic motion type duct cleaning robot for traveling in a flexible pipe,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Nov. 2019, pp. 6611–6621.

[10] Y. Shen, D. Guo, F. Long, L. A. Mateos, H. Ding, Z. Xiu, R. B. Hellman, A. King, S. Chen, C. Zhang, and H. Tan, “Robots under COVID-19 pandemic: A comprehensive survey,” IEEE Access, vol. 9, pp. 1590–1615, 2021.

[11] S. Sumartojo and D. Lugli, “Lively robots: Robotic technologies in COVID-19,” Social Cultural Geography, early access, May 2021, doi: 10.1080/14649365.2021.1921245.

[12] R. K. Megalingam, A. Prem, A. H. Nair, A. J. Pillai, and B. S. Nair, “Stair case cleaning robot: Design considerations and a case study,” in Proc. Int. Conf. Commun. Signal Process. (ICCSSP), 2016, pp. 760–764.

[13] S. H. Kakade, A. I. Kohare, S. P. Rakhunde, P. G. Bhojar, A. O. Pandey, and J. Y. Hande, “A review on automatic staircase climbing platform,” Int. J. Eng. Technol., vol. 5, no. 3, Mar. 2018.

[14] S. A. Sundaram, H. Wang, D. Ding, and R. A. Cooper, “Step-climbing power wheelchairs: A literature review,” Topics Spinal Cord Injury Rehabil., vol. 23, no. 2, pp. 98–109, 2017.

[15] H. Xin, C. Li, and P. Peng, “Design of low power triangle type stair cleaning machine,” in Proc. 2nd Int. Conf. Mater. Sci. Mach. Eng. Energy, 2017, pp. 1573–1578.

[16] T. Kakudou, K. Watanabe, and I. Nagai, “Study on mobile mechanism for a stair cleaning robot-design of translational locomotion mechanism,” in Proc. 11st Int. Conf. Control. Autom. Syst., 2011, pp. 1213–1216.

[17] T. Kakudou, K. Watanabe, and I. Nagai, “Mobile mechanism of a robot for cleaning and locomotion on stairs,” in Proc. Annu. Conf. (SICE), 2012, pp. 145–148.

[18] C.-C. Liu, Y.-P. Kang, and S.-N. Yu, “Hardware and software integration for domestic stairs cleaning robot,” in Proc. Annu. Conf., Jun. 2011, pp. 663–670.

[19] L. Zhang, Y. Yang, Y. Gu, X. Sun, X. Yao, and L. Shuai, “A new compact stair-cleaning robot,” Mech. Robot., vol. 8, no. 4, pp. 1–10, Aug. 2016.

[20] M. A. V. J. Muthugala, S. M. B. P. Samarakoon, and M. R. Elara, “Tradeoff between area coverage and energy usage of a self-reconfigurable floor cleaning robot based on user preference,” IEEE Access, vol. 8, pp. 76267–76275, 2020.

[21] S. M. B. P. Samarakoon, M. A. V. J. Muthugala, A. V. Le, and M. R. Elara, “HTetro-infi: A reconfigurable floor cleaning robot with infinite morphologies,” IEEE Access, vol. 8, pp. 69816–69828, 2020.

[22] M. Iyas, S. Yayao, R. E. Mohan, M. Devarassu, and M. Kalimuthu, “Design of sTetro: A modular, reconfigurable, and autonomous staircase cleaning robot,” J. Sensors, vol. 2018, Jul. 2018, Art. no. 8190802.

[23] A. V. Le, P. T. Kyaw, R. E. Mohan, S. H. M. Sre, A. Rajendran, K. Boopathi, and N. H. K. H. Nhan, “Autonomous floor and staircase cleaning framework by reconfigurable sTetro robot with perception sensors,” J. Intell. Robot. Syst., vol. 101, no. 1, pp. 1–19, Jan. 2021.

[24] L. Zhang, Y. Yang, L. Li, Y. Gu, H. Chen, and L. Shuai, “Avoidance of falling and collision for a stair-cleaning robot,” in Proc. 35th Chin. Control Conf. (CCC), Jul. 2016, pp. 6250–6255.

[25] L. Shapiro and G. Stockman, Computer Vision. Upper Saddle River, NJ, USA: Prentice-Hall, 2001.

[26] F. Göngör and O. Tutsöy, “Design of a chair recognition algorithm and implementation to a humanoid robot,” in Proc. Int. Conf. Sci. Academic Congress, 2018, pp. 1–15.

[27] J. Matas, C. Galambos, and J. Kittler, “Robust detection of lines using the progressive probabilistic Hough transform,” Comput. Vis. Image Understand., vol. 78, no. 1, pp. 119–137, Apr. 2002.

[28] J. Canny, “A computational approach to edge detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[29] M. A. V. J. Muthugala, M. Vega-Heredia, R. E. Mohan, and S. R. Vishal, “Design and control of a wall cleaning robot with adhesion-awareness,” Symmetry, vol. 12, no. 1, p. 122, Jan. 2020.

[30] K. B. Phan, H. T. Ha, and S. V. Hoang, “Eliminating the effect of uncertainties of cutting forces by fuzzy controller for robots in milling process,” Appl. Sci., vol. 10, no. 5, p. 1685, Mar. 2020.

[31] S. M. B. P. Samarakoon, M. A. V. J. Muthugala, and M. R. Elara, “Toward obstacle-specific morphology for a reconfigurable tiling robot,” J. Ambient Intell. Hum. Comput., vol. 4, pp. 1–13, Jun. 2021.

[32] L. Ibarra and C. Webb, “Advantages of fuzzy control while dealing with complex/unknown model dynamics: A quadcopter example,” New Appl. Artif. Intell., vol. 1, p. 93, Aug. 2016.

[33] T. J. Ross, Fuzzy Logic With Engineering Applications. Hoboken, NJ, USA: Wiley, 2005.

M. A. VIRAJ J. MUTHUGALA (Member, IEEE) received the B.Sc.Eng. and M.Sc. degrees in electrical engineering from the University of Moratuwa, Sri Lanka, in 2014 and 2018, respectively. He is currently a Research Fellow with the Engineering Product Development Pillar, Singapore University of Technology and Design (SUTD), Singapore. His current research interests include maintenance and inspection robotics, reconfigurable robotics, service robotics, human–robot interaction, and intelligent systems.

Dr. Muthugala was a recipient of the Nanjing City Prize in IEEE Ro-Man 2018, and a Runner-up in Mini Drone Competition at IEEE/RSJ IROS 2018. He was also the Assistant Secretary of the IEEE Robotics and Automation Society, Sri Lanka Section Chapter, from 2017 to 2019.

S. M. BHAGYA P. SAMARAKOON (Student Member, IEEE) received the B.Sc.Eng. and M.Sc. degrees in electrical engineering from the University of Moratuwa, Sri Lanka, in 2017 and 2018, respectively. She is currently pursuing the Ph.D. degree with the Engineering Product Development Pillar, Singapore University of Technology and Design, Singapore.

Her research interests include reconfigurable robotics, cleaning robotics, human–robot interaction, and service robotics.

Ms. Samarakoon was a recipient of the Nanjing City Prize in IEEE Ro-Man 2018, and a Runner-up in Mini Drone Competition at IEEE/RSJ IROS in 2018. She is also the Secretary of the IEEE Women in Engineering (WIE), Singapore Section.

PRABAKARAN VEERAJAGADHESWAR received the bachelor’s degree in electronics and instrumentation engineering from Sathyabarna University, India, in 2013, and the Ph.D. degree in advanced engineering from Tokyo Denki University, in 2019. He is currently working as a Research Fellow with the ROAR Laboratory, Singapore University of Technology and Design. He is also a Visiting Instructor for a design course with the International Design Institute, Zhejiang University, China. His research interests include the development of complete coverage path planning, SLAM framework, and embedded control for reconfigurable and climbing robots. He received the SG Mark Design Award in 2017 for the designing of h-Tetro, a self-reconfigurable cleaning robot.

MOHAN RAJESH ELARA received the B.E. degree from Bharathiar University, India, in 2003, and the M.Sc. and Ph.D. degrees from Nanyang Technological University, in 2005 and 2012, respectively. He is currently an Assistant Professor with the Engineering Product Development Pillar, Singapore University of Technology and Design (SUTD). He is also a Visiting Faculty Member of the International Design Institute, Zhejiang University, China. He has published more than 80 papers in leading journals, books, and conferences. His research interests include robotics with an emphasis on self-reconfigurable platforms as well as research problems related to robot ergonomics and autonomous systems. He was a recipient of the SG Mark Design Award in 2016 and 2017, the ASEE Best of Design in Engineering Award in 2012, and the Tan Kah Kee Young Inventors’ Award in 2010.