

Research Article

A New Robust Method for Mobile Robot Multifloor Navigation in Distributed Life Science Laboratories

Ali A. Abdulla,^{1,2} Hui Liu,¹ Norbert Stoll,³ and Kerstin Thurow¹

¹Center for Life Science Automation (CELISCA), University of Rostock, 18119 Rostock, Germany

²College of Engineering, University of Mosul, Mosul, Iraq

³Institute of Automation, University of Rostock, 18119 Rostock, Germany

Correspondence should be addressed to Ali A. Abdulla; ali.abdulla@celisca.de

Received 7 February 2016; Revised 16 May 2016; Accepted 29 May 2016

Academic Editor: Yongji Wang

Copyright © 2016 Ali A. Abdulla 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.

A new robust method is proposed for multifloor navigation in distributed Life Science Laboratories. This method proposes a solution for many technical issues including (a) mapping and localization with ceiling landmarks and a StarGazer module for achieving an accurate and low cost multifloor navigation system, (b) a new method for path planning to navigate across multiple floor environments called backbone method and embedded transportation management system, (c) elevator environment handler with the necessary procedures to interact with the elevator presenting a new approach for elevator entry button and internal buttons detection, and (d) communication system to get an expandable network; this method utilizes a TCP/IP network for the communication. Many experiments in real Life Science Laboratories proved the efficient performance of the developed multifloor navigation system in life science environment.

1. Introduction

In recent years, the growing interest towards automated system development caused a huge change in life science working processes [1, 2]. Laboratories robots are commonly used in automation processes [3, 4]. Different types of laboratories robots have been developed for transportation tasks including stationary robot [5, 6] and mobile robot [7, 8]. Many technical issues have to be handled for an effective mobile robot navigation system including localization [9, 10], communication [11, 12], path planning [13, 14], elevator aspects [15, 16], collision avoidance [17, 18], and multifloor navigation [19]. Liu et al. developed a new intelligent transportation system based on mobile robot H20 for lab ware transportation between different laboratories in single floor [20, 21]. This system uses a Floyd-Genetic algorithm for path planning and a blind grasping method for arm manipulation. Life Science Laboratories are often distributed on different floors; thus suitable methods for multifloor navigation are required. This will also lead to more and higher flexibility and will open the way to a complete automation of complex laboratories in life sciences. This paper will present an

innovative multifloor mobile robot navigation system for transportation tasks.

A survey of the actual literature will clarify the working method and demonstrate advantages/disadvantages of existing solutions. Yu et al. applied a computer vision for elevator button detection and recognition. The detection is divided into button panel (BP) detection and button recognition. A method called multiple partial model has been used to enhance the BP detection and combines structural inference, Hough transform, and multisymbol recognition to propose a method for button recognition. The experiments show a detection success rate of 80.6% for external button and 87.3% for internal buttons detection. It is impossible to detect the button when the detection of the BP has failed [22]. Baek and Lee used stereovision in mobile robot navigation to propose an elevator entrance door detection algorithm. This method includes three parts as follows: (a) image capturing and filtering, (b) implementation of an algorithm to extract the elevator door, and (c) detection of the size of the elevator. In this algorithm, the detection of the door depends on the camera angle, which has to be more than 50 degrees to ensure an easy detection of the elevator entrance [23].

Kang et al. proposed a new strategy for mobile robot navigation in multifloor environment. Digital image processing and template matching neural network have been used to find the outside elevator entry button, outside direction indicator, inside current floor, and destination floor button. This strategy works only for the elevator recognition without dealing with button pressing issues. Thus the robot cannot work autonomously [24]. Chae et al. utilized a group of sensors: laser range finder, two dead reckoning sensors, and passive artificial landmarks to build a mapping cart. The mapping cart is used to remove the human effort when building a large scale map. The method was applied in indoor T-city environments. The results show that this method is convenient and efficient in building a large scale map [25]. Ul-Haque and Prassler analyzed the performance of low cost sensors, called StarGazer module (SGM), in comparison with other sensors. The use of this technique leads to an increased detection of the position and angular accuracy, precision, reliability, repeatability, and robustness [26]. Troniak et al. proposed a multifloor mobile robot transportation system that integrates the stereovision, monocular vision, and laser range finder to sense the environment. The map is built based on an SLAM algorithm and is then used to find the plan to the destination. The template matching algorithm is used to find the elevator buttons. The accuracy of the elevator button detection from the outside reaches 85% and 63% from the inside. This system requires a long time to detect the elevator entrance button (4.6 s); also the robot needs a long time to enter the elevator which may cause it to miss the elevator. Thus human assistance is still required [27].

Although a few researches have focused on the multifloor navigation issue there is still the need to develop a new method with more features such as high speed, low cost, high precision, robustness, immunity to light effects, and easy expansion. In this study a smart passive landmark reader called SGM (Hagisonic Company, Korea) is adopted for localization purposes [28]. This module can meet all above-mentioned features but is used for one floor navigation only [26]. The main contribution of this work is the development of a new multifloor navigation system (mapping, localization, and path planning) taking into consideration the elevator environment handler system for laboratory transportation tasks. The organization of the paper is as follows: Section 2 describes the system architecture. The multifloor navigation system with relative map, localization, innovative path planning method, elevator aspects, and communication socket is described in Section 3. Analyzing the experimental results for the evaluation of the system is demonstrated in Section 4. Section 5 discusses the conclusions and describes the work related to a given problem.

2. System Structure

The architecture of the system is shown in Figure 1. It consists mainly of four blocks: Multifloor System (MFS), Robot Remote Center (RRC), Robot Motion Center (RMC), and mobile robot.

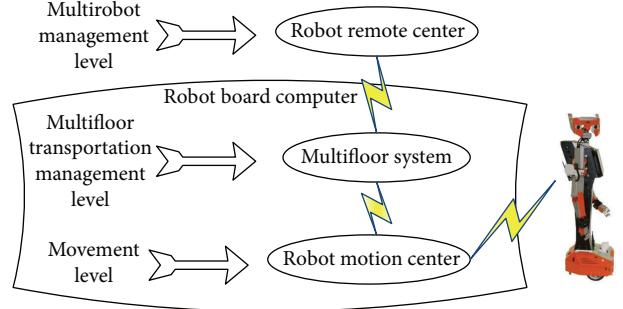


FIGURE 1: System structure.

The presented system can be explained as follows: (a) RRC is the highest level control. It manages transportation tasks by observing robots' status and then forwards incoming transportation task to the MFS of the appropriate robot. (b) The MFS is the transportation management level, which is installed on the H20 mobile robots. It communicates with both RRC and RMC. The MFS is responsible for mapping, localization, path planning, and automated door control. (c) The RMC is the movement execution level, which executes the incoming movement orders from the MFS by controlling the hardware modules of the H20 mobile robot. It also reads the mobile robot data including images stream, ultrasonic measurement distance sensor, infrared measurement distance sensors, and power status. Finally, the mobile robot represents the hardware level of the mobile robot system. The TCP/IP client/server architecture has been utilized to communicate between these controls levels. Figure 2 shows the data flow between RRC, MFS, and RMC in transportation task as follows: (a) the RRC sends the transportation order with requested start and end position. MFS replies to RRC with transportation status, which includes transportation start, transportation finished, and problem in transportation due to wrong points or paths. The MFS sends the robot position in multifloor environment as X, Y, angle, and ID and the power status of each battery periodically as well. (b) MFS sends orders to RMC including move straight, rotate, vision landmark reader data request, ultrasonic distance reading request, and power status request. When the RMC receives the MFS orders it reports with movement and rotation acknowledgment. Besides that, the RMC replies with the requested data information including current landmark reader data (X, Y, angle, and ID), the power data and status for both robot batteries, and ultrasonic distance measurement data.

3. Multifloor Navigation System

In mobile robot navigation systems, many issues have to be handled: mapping, localization, path planning, elevator, and communication systems. In this section, the mapping and localization methods will be proposed. The steps for path planning and elevator aspects will be described. Finally, the integration into the general automated mobile robot transportation system will be clarified.

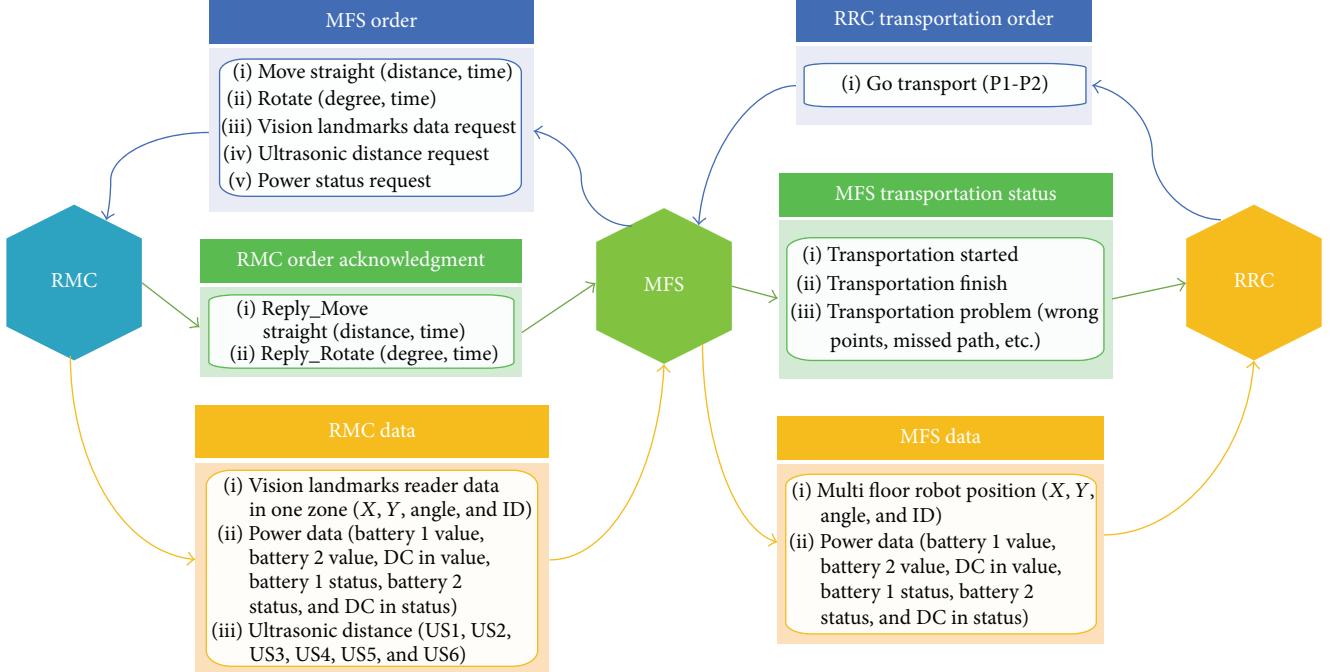


FIGURE 2: Data flow between MFS, RRC, and RMC.

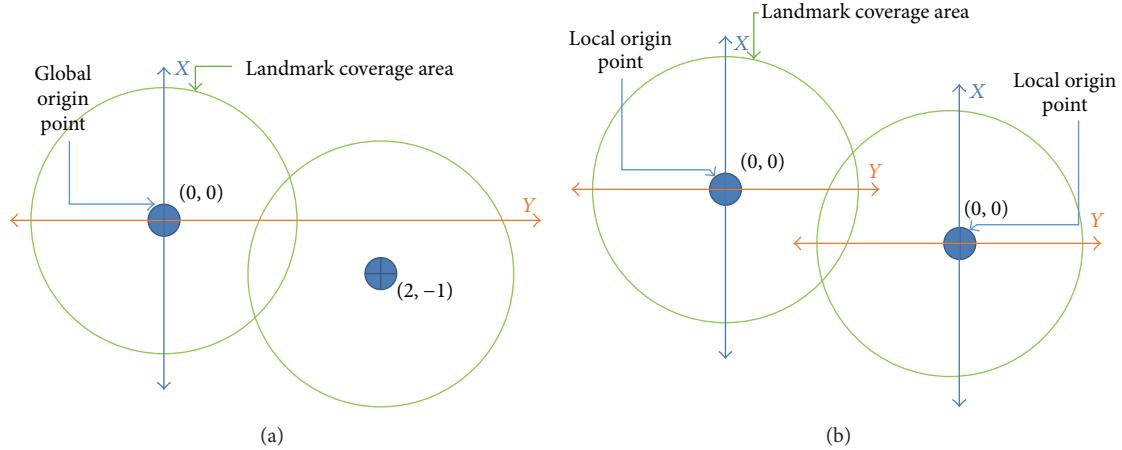


FIGURE 3: SGM modes. (a) Map mode. (b) Alone mode.

3.1. Mapping. In multifloor transportation systems certain features are required like high speed, easy coverage of a large area, low costs, and low sensitivity to light effects. The SGM is used to achieve the mentioned conditions, which consist of an infrared camera, a group of infrared LED projectors, an internal flash memory, and a digital image processing board. The SGM internal work procedures can be explained as projecting an infrared light, capturing the reflected IR images from passive landmarks, and analysis of the IR images to find the ID, angle, X , and Y . This module can work in alone mode or map mode. Figure 3(a) shows the map mode, which has only one global origin point while in alone mode every landmark has a local origin point as clarified in Figure 3(b).

Figure 4 shows the installed SGM at the top of the H20 mobile robot and the installed ceiling passive landmark in CELISCA.

Despite the advantages of the SGM, it cannot deal in mapping mode with multifloor environments for two reasons: (a) SGM can build only one map at the same time and (b) it cannot build a map inside the elevator due to height limitation (SGM cannot read a position correctly when the height of landmark to the SGM is less than 1.1 m).

The system was applied using SGM in mapping mode, which stores the complete map inside the SGM and this module gives the position in the building map to the robot and, according to the transportation task, the robot moves in a generated path [20]. This method works for single floor



FIGURE 4: SGM installed on H20 and the ceiling landmarks [20].

and requires a long time due to individual map building and path generation for each robot. A new mapping method is proposed and was applied in Life Science Laboratories to solve the multifloor mapping problem. In this method the landmarks ID and related information are stored in a relative map that is used with a navigation program to reach the required destination based on the planning path. Relative map structure can be presented as follows:

RelativeMap

$$\begin{aligned}
 & \left[\begin{array}{cccc} \text{IDF}_{1,1} & \text{IDF}_{1,2} & \cdots & \text{IDF}_{1,n-1} & \text{IDF}_{1,n} \\ \text{XF}_{1,1} & \text{XF}_{1,2} & \cdots & \text{XF}_{1,n-1} & \text{XF}_{1,n} \\ \text{YF}_{1,1} & \text{YF}_{1,2} & \cdots & \text{YF}_{1,n-1} & \text{YF}_{1,n} \\ \theta_{F_{1,1}} & \theta_{F_{1,2}} & \cdots & \theta_{F_{1,n-1}} & \theta_{F_{1,n}} \\ \text{IDF}_{2,1} & \text{IDF}_{2,2} & \cdots & \text{IDF}_{2,n-1} & \text{IDF}_{2,n} \\ \text{XF}_{2,1} & \text{XF}_{2,2} & \cdots & \text{XF}_{2,n-1} & \text{XF}_{2,n} \\ \text{YF}_{2,1} & \text{YF}_{2,2} & \cdots & \text{YF}_{2,n-1} & \text{YF}_{2,n} \\ \theta_{F_{2,1}} & \theta_{F_{2,2}} & \cdots & \theta_{F_{2,n-1}} & \theta_{F_{2,n}} \\ & & \vdots & & \\ & & \vdots & & \\ & & \vdots & & \end{array} \right] = \dots \\
 & \left[\begin{array}{cccc} \text{IDF}_{m-1,1} & \text{IDF}_{m-1,2} & \cdots & \text{IDF}_{m-1,n-1} & \text{IDF}_{m-1,n} \\ \text{XF}_{m-1,1} & \text{XF}_{m-1,2} & \cdots & \text{XF}_{m-1,n-1} & \text{XF}_{m-1,n} \\ \text{YF}_{m-1,1} & \text{YF}_{m-1,2} & \cdots & \text{YF}_{m-1,n-1} & \text{YF}_{m-1,n} \\ \theta_{F_{m-1,1}} & \theta_{F_{m-1,2}} & \cdots & \theta_{F_{m-1,n-1}} & \theta_{F_{m-1,n}} \\ \text{IDF}_{m,1} & \text{IDF}_{m,2} & \cdots & \text{IDF}_{m,n-1} & \text{IDF}_{m,n} \\ \text{XF}_{m,1} & \text{XF}_{m,2} & \cdots & \text{XF}_{m,n-1} & \text{XF}_{m,n} \\ \text{YF}_{m,1} & \text{YF}_{m,2} & \cdots & \text{YF}_{m,n-1} & \text{YF}_{m,n} \\ \theta_{F_{m,1}} & \theta_{F_{m,2}} & \cdots & \theta_{F_{m,n-1}} & \theta_{F_{m,n}} \end{array} \right]
 \end{aligned} \tag{1}$$

From (1), the map consists of multilayers each one representing one floor. Each layer contains the map elements which are landmark ID number (IDF), X position (XF) according to start point, Y position (YF) according to the

same reference point, and finally the angle (θF) which is also related to the reference point of the current floor. This map is created manually by special procedures: (a) collecting data: representation of the difference between two landmarks including the distance and angle relation. This step is done using a moving cart, which was positioned between every two landmarks and then the relations as ID, X_{ID} , Y_{ID} , and Angle_{ID} were recorded. (b) The second is entering the data into excel sheet with a required mathematic equation to find the relation between landmarks. Consider

$$\begin{aligned}
 X_{n+1} &= X_n + \Delta X_n, \\
 Y_{n+1} &= Y_n + \Delta Y_n,
 \end{aligned} \tag{2}$$

where ΔX , ΔY are the measured distance between the current two landmarks based on SGM reading. Consider

$$\theta_{n+1} = \theta_r - \theta_{LM}, \tag{3}$$

where θ_r is the reference point angle and θ_{LM} is the installed landmark orientation (0, 90, 180, and -90). The excel sheet was prepared for an easy finding of the relations after entering the whole data from the first stage. (c) Coding into the MF system: the relative positions for each landmark (ID, X_{relative} , Y_{relative} , and $\text{Angle}_{\text{relative}}$) were taken from the excel sheet and added to the floor layer. A unique reference point for all floors was defined to unify all the readings to the same reference. Consider

RelativeReference

$$= \begin{bmatrix} X_r F_1 & X_r F_2 & \cdots & X_r F_{n-1} & X_r F_n \\ Y_r F_1 & Y_r F_2 & \cdots & Y_r F_{n-1} & Y_r F_n \end{bmatrix}, \tag{4}$$

where $X_r F$, $Y_r F$ are a constant value for each floor to extract a unique reference in the whole map.

These procedures are done manually and require some time to be done accurately. Once realized it can be applied to N number of robots; thus it definitely minimizes the map/path generation time compared to the currently applied method.

3.2. Localization. Localization is considered a main key for robot's automation. It can be defined as estimating the absolute or relative position information. A simple and efficient localization method is required in laboratories of life science automation. The need to create a new localization method appeared after developing the relative map to localize the mobile robot in multifloor environments, while the earlier system depends completely on SGM in mapping and localization [20]. The new localization method is based on running the SGM in alone mode as HEX reader. In working mode, the localization method is used to translate the StarGazer reading into a useful information based on relative map using map matching method as clarified in Algorithm 1. This method starts by acquiring the raw data from the StarGazer reader and extracting the useful information about the position in the current landmark zone followed by finding the current floor based on ID numbers. Next step is the orientation translation to determine the correct robot coordinates and the

```

Input: SGM Raw data in alone mode ( $ID_{SGM}, X_{SGM}, Y_{SGM}, \theta_{SGM}$ )
Input: Relative Map [ $(ID, X_{Re}, Y_{Re}, \theta_{Re})_{1,1}, \dots, (ID, X_{Re}, Y_{Re}, \theta_{Re})_{m,n}$ ]
 $m$ : Number of Floor
 $n$ : Number of landmark for each floor
CF: Current Floor
CLP: Current Landmark Position
(1) For  $i = 1 : m$  //Floor Selection
(2)           For  $j = 1 : n$  //landmark Selection
(3)             If  $ID_{SGM} = \text{RelativeMap}[i, 1, j]$  //search about matched Landmark ID
(4)               CF =  $i$ 
(5)               CLP =  $j$ 
(6)             End IF
(7)           End For
(8)         End For
(9) Apply (5) to correct the landmark orientation ( $x_{SGO}, y_{SGO}, \theta_G$ )
(10) Apply (6) as following:  $X_G = XF_{CF,CLP} + X_{Re}F_{CF} + x_{SGO}$ 
(11) Apply (7) as following:  $Y_G = YF_{CF,CLP} + Y_{Re}F_{CF} + y_{SGO}$ 
(12) Return robot position in global map ( $X_G, Y_G, \theta_G, CF$ )

```

ALGORITHM 1: Localization method in multifloor environment.

global angle based on the predefined landmark orientation (θ_R) in relative map. Consider

$$\begin{aligned}
x_{SGO} &= \begin{cases} x_{SG}, & \theta_R = 0 \\ -x_{SG}, & \theta_R = 180 \\ y_{SG}, & \theta_R = 90 \\ -y_{SG}, & \theta_R = -90, \end{cases} \\
y_{SGO} &= \begin{cases} y_{SG}, & \theta_R = 0 \\ -y_{SG}, & \theta_R = 180 \\ -x_{SG}, & \theta_R = 90 \\ x_{SG}, & \theta_R = -90, \end{cases} \\
\theta_G &= \begin{cases} \theta_{SG}, & \theta_R = 0 \\ \theta_{SG} - 180, & \theta_R = 180 \\ \theta_{SG} - 90, & \theta_R = 90 \\ \theta_{SG} + 90, & \theta_R = -90. \end{cases}
\end{aligned} \quad (5)$$

Finally x , y StarGazer readings are translated into real coordinates in a global multifloor map:

$$x_G = x_{SGO} + x_R + x_{FR}, \quad (6)$$

$$y_G = y_{SGO} + y_R + y_{FR}, \quad (7)$$

where x_R , y_R are relative values from relative map and x_{FR} , y_{FR} are multifloor reference for the current floor, which is used as a multifloor unique reference.

The map matching method is coded in a way that easily allows an expansion to m number of floors and n number of landmarks. If any ceiling landmark cannot be recognized correctly during the mobile robot's movement

using the built multifloor map, the corresponding mobile robot finds and presents the missing landmark ID based on the last landmark and the next landmark. This landmark error-handling approach can guarantee a high-successful rate during robot movement.

3.3. Path Planning. Path planning is an important task to achieve an efficient navigation system. Path planning can be defined as a motion control from start point to the goal point with collision avoidance taking the shortest path to the destination. Many algorithms have been proposed and developed for this purpose [29–32]. Compared to the standard ones, the path planning issue has more challenges for the mobile robot transportation in Life Science Laboratories as follows: since the proposed path planning method will be applied in the real robot transportation system, the real-time planning results are desired. However, there are so many combinations of the transportation starting positions and destinations, so the expected path planning method should be able to combine the dynamic path combinations fast and flexibly. For this purpose, in this study a new path planning method (named as backbone method) is presented, which is comprised of an essential path for each floor and a group of flexible subpaths for all laboratory stations. The corresponding Internal Transportation Management System (ITMS) software has also been developed by adopting the proposed backbone method.

3.3.1. Internal Transportation Management System (ITMS). ITMS is a transportation management system inside the MFS. It has been coded based on multiple thread technique to increase the efficiency and the ability to access the MFS at any time. The management system shown in Figure 5 divides the incoming transporting task into small tasks and keeps monitoring the execution of these tasks continuously. It also informs the Robot Remote Center (RRC) about the current

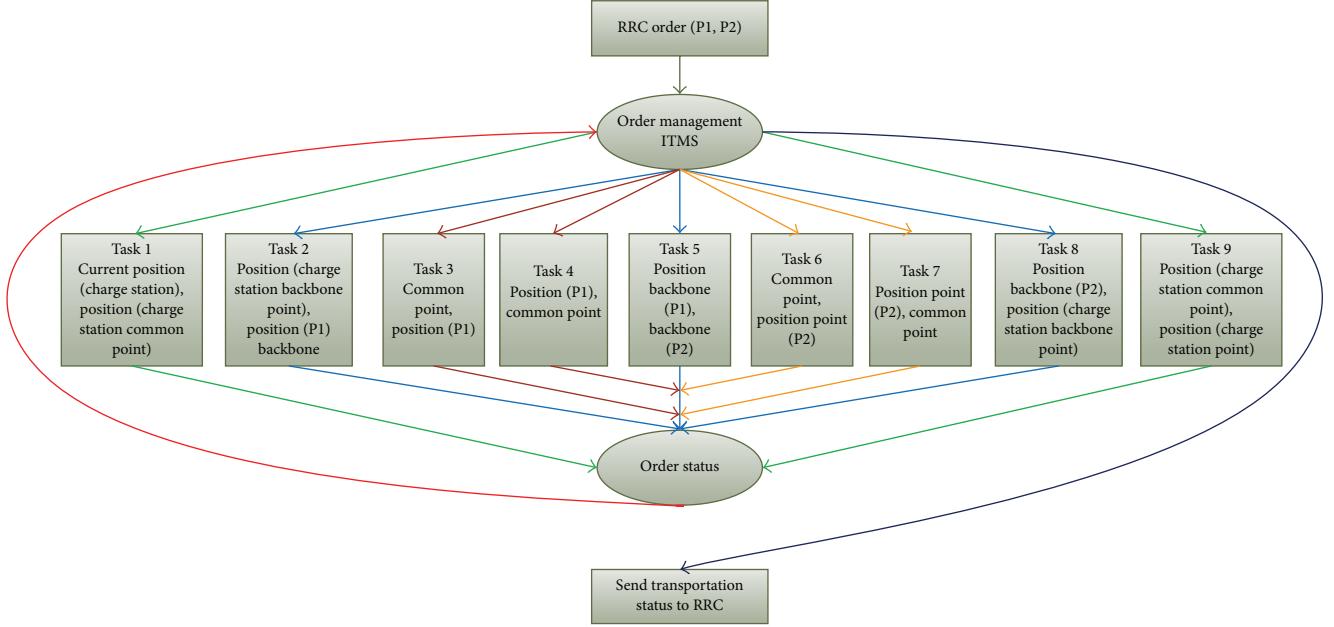


FIGURE 5: Internal transportation management system.

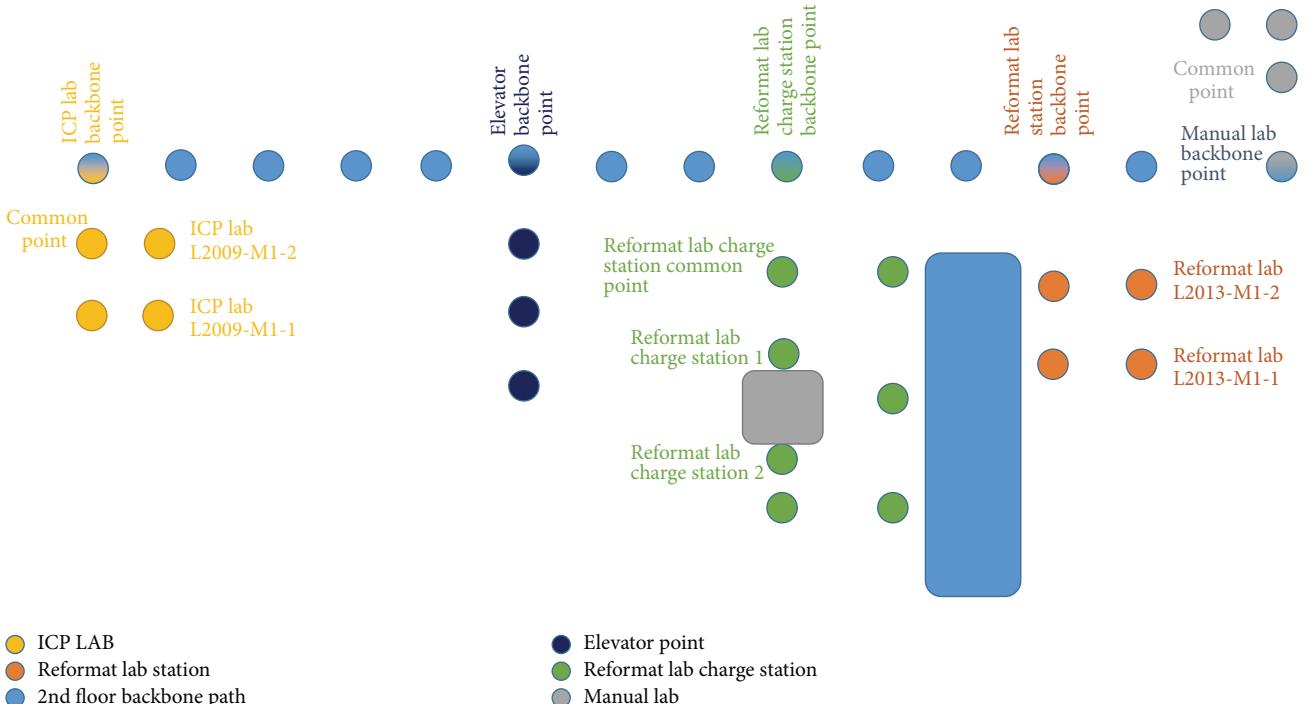


FIGURE 6: Backbone waypoint for 2nd floor in CELISCA.

transportation state (robot busy, transportation start, RAKM problem, wrong path/points parameters, and transportation finished). The ITMS chooses the right path and the points, sends it to the movement core, and waits till the end of this task. Then another path is loaded until the whole transportation task is done. When there is a problem on any task the ITMC stops other tasks immediately and informs the RRC.

3.3.2. The Backbone Method. This method is based on building a main path for each floor. This path has multiple key points leading to subpaths that complete the transportation path as shown in Figure 6. This method decreases the number of paths required to cover a large map with many stations (multiple source and multiple destination points). In addition to that, this method makes the path creation easier than normal static path. The number of paths required in the static

path method is $N * N - 1$ where N is the number of stations. The total number of path required in the backbone method is (backbone path + $N/2$). For a floor with six transportation station points with two charge station points, result was 56 paths in static method and 5 for the backbone path method. Table 1 shows the reduction of the required paths by the backbone method especially in multifloor environment. When the robot receives the transportation task from RRC, the ITMS divides the transportation task into small tasks and is starting the execution of these tasks as follows: (a) search inside the created paths to find the way from the charge station to the point near the backbone path. (b) Load the current floor essential path (backbone) and according to the source station position either move directly to the grasping station key point or move through multiple paths (current path towards elevator, elevator path, and grasping floor essential path to the grasping station key point) if the grasping station is on another floor. (c) Use the current grasping subpath to reach the grasping station and execute the required operation. (d) Repeat (b) and (c) taking into consideration that the movement is towards placing station. (e) Load the current floor backbone path and either move till reaching the charge key point on the path occurs or move through (current path towards elevator, elevator path, and charging floor essential path to the charging station key point) in case the charge station is on a different floor. (f) Use the charging path to reach the charge station and wait for the new transportation task.

3.4. Elevator Aspects. The elevator should be used to move across levels in multifloor environment. As a first step to handle the elevator, the robot moves towards the elevator door based on ceiling landmarks until reaching a calculated point which brings the robot in a right position to press the button and enter the elevator at the same time as clarified in Figure 7(d). After button pressing the mobile robot continuously checks the door to enter the elevator when the door is open. As for the next step, the robot localizes himself using the installed landmark inside the elevator. Next, the robot moves to a suitable point near the control panel to recognize the destination floor button and control the elevator door correctly. The developed methods for buttons detection were divided into outside and inside elevator as follows.

3.4.1. The Elevator Entry Button. Since the entry button and its panel are made from the same reflecting material, the detection of the button is difficult to realize. Thus, a landmark has been installed around the entry button with a specific shape and color to enable an easy and applicable recognition process for mobile robot. The elevator entry button detection process can be explained as follows: (a) initialize the Kinect sensor with the required frame rate, image resolution, and depth resolution. (b) Capture an image from Kinect RGB sensor. (c) Use an RGB channel filter to remove the selected band of colors. (d) Search for connected pixels (object) inside the image. (e) Select candidates based on specific width and height. (f) Check if the chosen candidates have square shape

TABLE 1: Static method and backbone method.

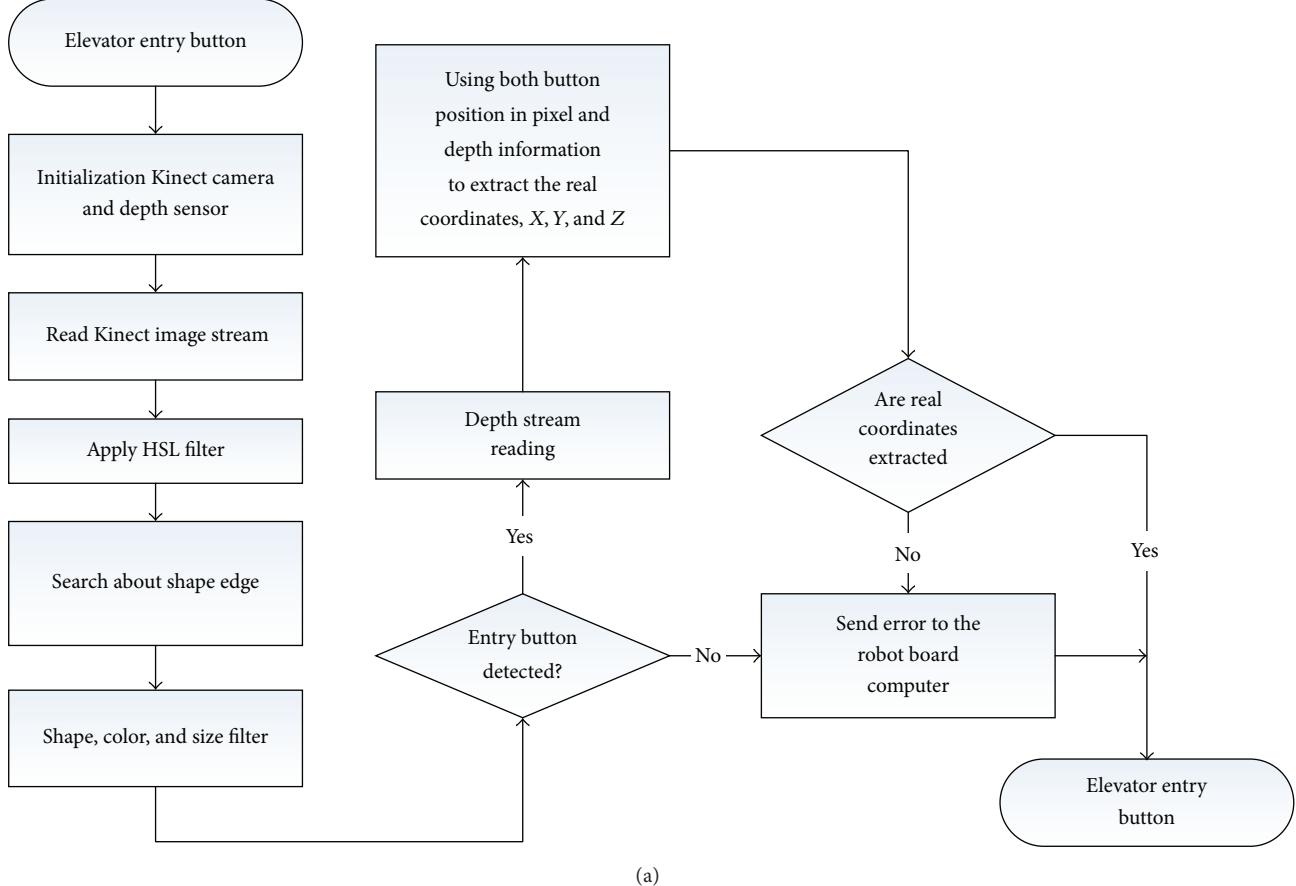
Number of transportation station points	Normal paths method	Backbone path method
8 (2nd) floors	56	5
12 (2nd + 3rd) floors	132	9

with the specified acceptance distortion and color. (g) Get the depth snapshot from the Kinect sensor; then do the mapping between position in pixel and depth to extract the real coordinates (X , Y , and Z). (h) Calibration between Kinect positions and the arm reference for arm pressing operation is performed.

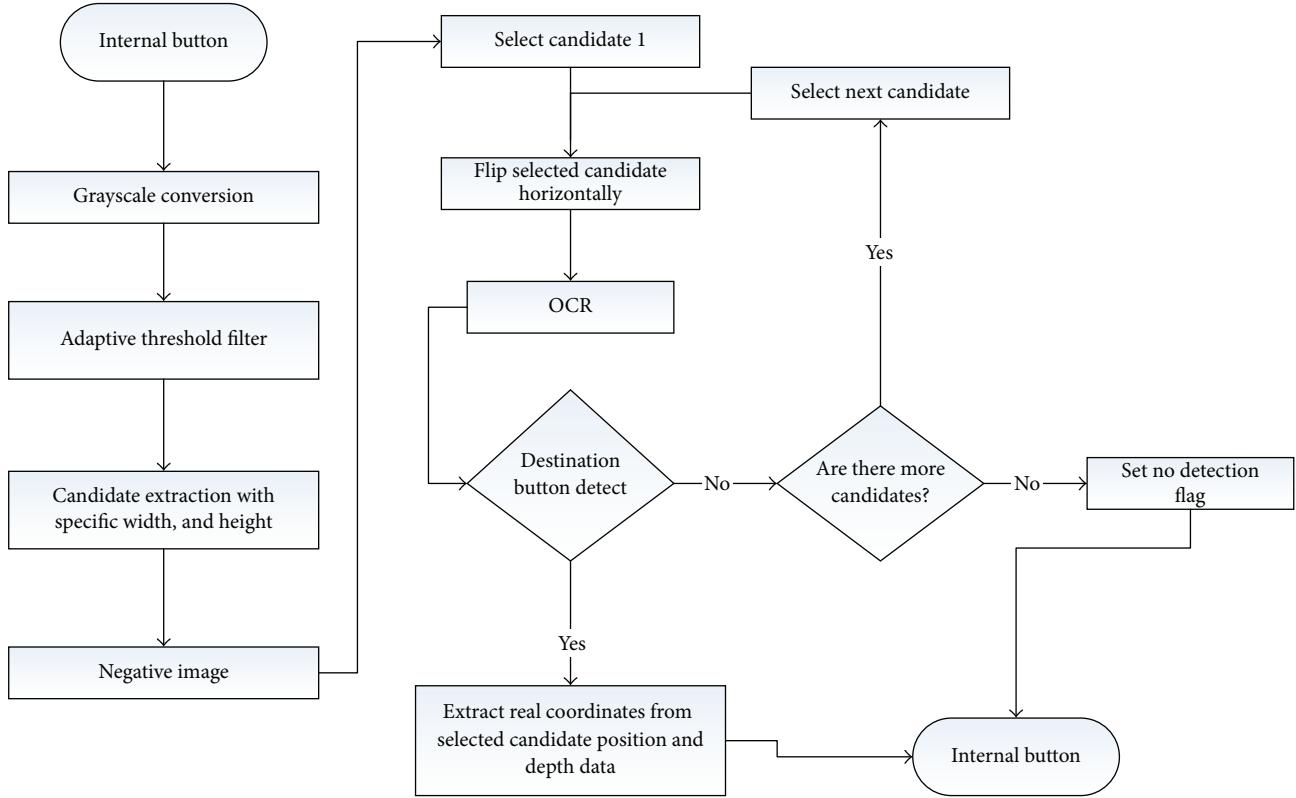
This method has 88.5% successful rate at a stable lighting condition. But the working environment has different lighting and sunlight conditions which easily affect the detection process with RGB color system. Thus, it reduces the successful rate. This problem has been solved using the HSL color representation which is more stable against the changing of lighting conditions as will be clarified in the experiment section. Figure 7(a) shows the HSL based button detection method.

3.4.2. The Elevator Internal Button. For successful buttons detection, the diameter of the button must be equal to or larger than 1.9 cm and the button label should be placed to the left of the related button based on American Disability Association (ADA) recommendations [33]. In this method, it is important to recognize each button label separately; thus the developed method using a combination of filters includes grayscale conversion that makes the captured image suitable for next stages, stretch contrast to improve the contrast of the image by stretching the domain of intensity, adaptive threshold to choose the best threshold under different light condition for binary image conversion that have been applied. Next, searching for each button's candidates using specific features (width and height), taking the inverse value of the pixel, and flipping the candidates horizontally (Kinect image stream has a mirror image) to make it suitable for the Optical Character Recognition (OCR) stage, each extracted candidate passes to OCR engine for comparison with the required destination, and finally based on the position of the matching candidate inside the image and depth information, the real coordinates are extracted and translated into robot arm reference. Figure 7(b) demonstrates the internal button detection method.

The proposed detailed robot-elevator interface controlling strategy is demonstrated in Figure 7(c). The elevator handler system has two methods for error handling: (1) in case of failure of the mobile robot to reach the required elevator button pushing area, the position and orientation correction function controls the robot to reach the defined position with high-accuracy orientations. This function checks the robot position after movement and tries to correct the robot position three times till it reaches the defined position

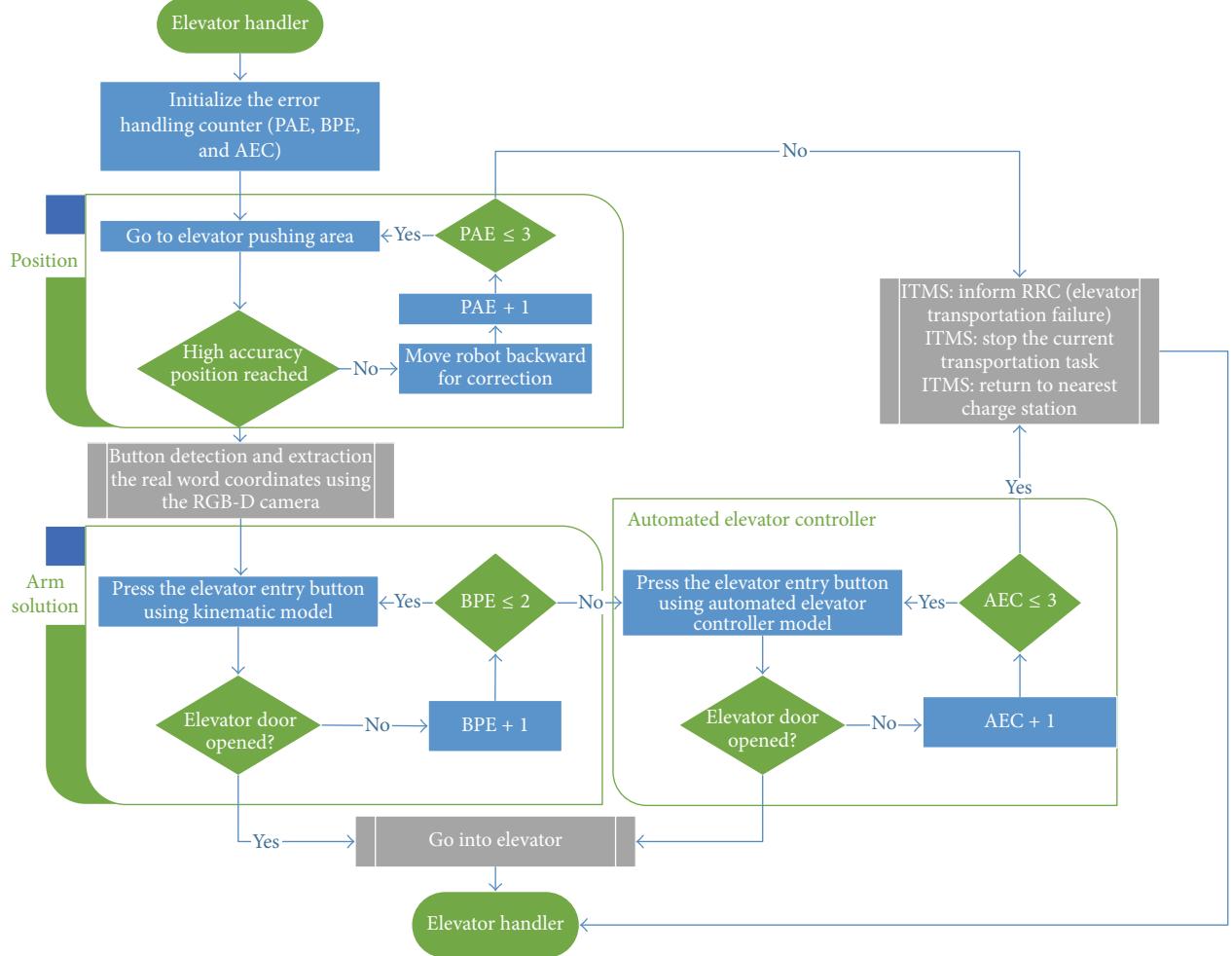


(a)



(b)

FIGURE 7: Continued.

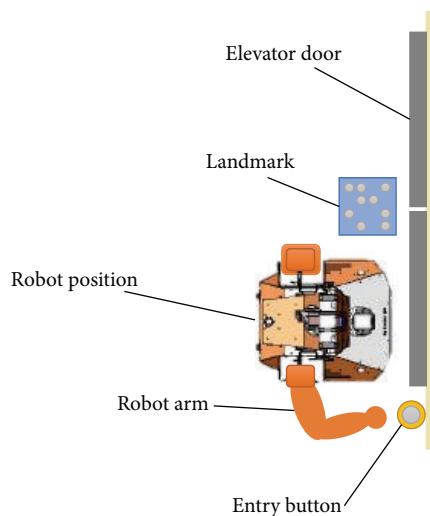


PAE: position accuracy error

PBE: button pressing error

AEC: automated elevator controller

(c)



(d)

FIGURE 7: Robot-elevator door controlling case. (a) HSL based entry button detection flow chart. (b) Internal buttons detection flow chart. (c) Button pressing flow chart. (d) Robot and elevator position.

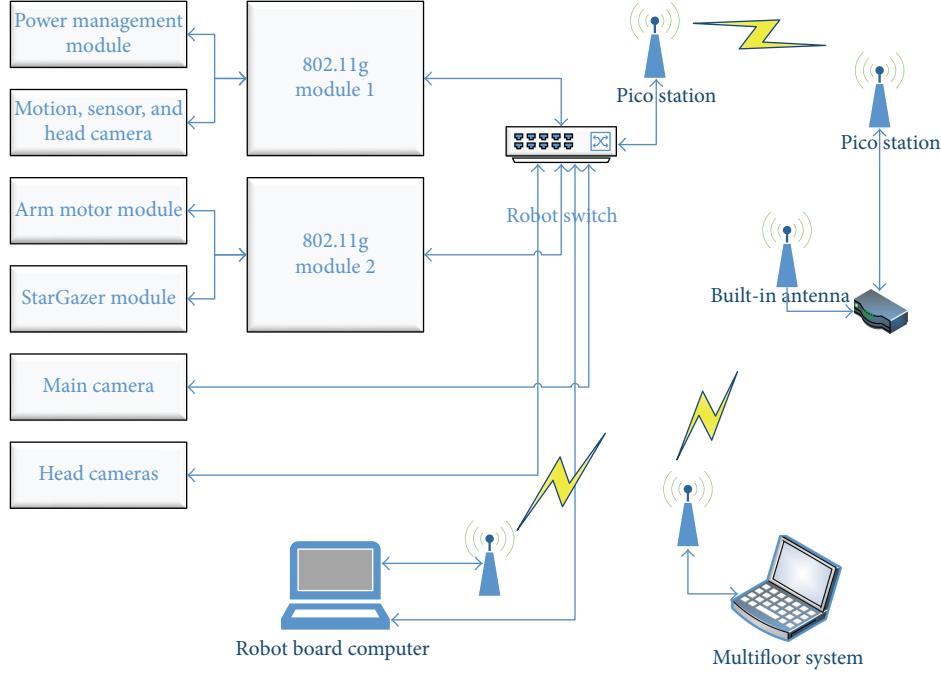


FIGURE 8: Communication system.

accurately. (2) A hybrid elevator controlling way starts after the robot identified the button position as (X , Y , and Z). The values are then passed to the solved kinematic arm models for button pressing. The robot checks the elevator door with ultrasonic sensor and controls the arm again to press the button when the door is still closed. If previous attempts have failed, the hybrid elevator controller selects the automated elevator controller over the socket method to open the door. If the functions listed above failed in handling the elevator, the ITMS informs the higher level controller RRC, stops the current transportation operation, and controls the robot till it reaches the charging station.

3.5. System Communication. In this study the wireless IEEE 802.11g network is utilized for laboratories transportation, which has a suitable width and fast data channel and can be easily expanded for large areas. The client/server architecture is adopted for data exchanging between the MFS and the mobile robot. The MFS represents the client socket and the onboard computer (inside mobile robot) is the server socket. Figure 8 shows the complete network architecture for the mobile robot. The SGM is connected to the internal switch through module 2, while the motion motors are connected through module 1. The MFS reads position information from SGM and sends movement orders through the Robot Motion Center.

A GUI of the MFS communication is displayed in Figure 9(a). From this figure it can be seen that this GUI is used to connect to the server socket, test position readings, and send movement and rotation orders. The server socket is illustrated in Figure 9(b) which shows the connection status and command history.

4. Experimental Results

Different experiments have been done to prove the performance of the presented MFS. In these experiments, three different kinds of verification are included: the validation of the navigation system, the recognition of the elevator buttons, and using the H20 arm to press the elevator button.

(A) The verification of the multifloor navigation includes three experiments. First the transportation task between laboratories in different floor was executed ten times. The performance of the relative map, localization based on relative map, backbone path planning method, communication system, and finally ITMS transportation management system are verified in this experiment. Figure 10 shows the multifloor GUI. Figure 11 clarifies the path created by the robot during its movement between 2nd and 3rd floor including the elevator in real Life Science Laboratories. Figure 12 shows the H20 mobile robot while executing the expected transportation task. Figure contents are explained as follows: in Figure 12(a) robot is initially in charge station; in Figure 12(b) robot leaves the charge station towards the current floor essential path and moves through it till it reaches the grasping station as shown in Figure 12(c); in Figure 12(d) robot is executing the grasping order and returns to the essential path to reach the elevator and in Figure 12(e) it is waiting till the elevator door is opening (see Figure 12(f)); robot enters the elevator and is leaving it after approaching the destination floor as shown in Figures 12(g)-12(i); it passes through destination essential path towards the placing station shown in Figures 12(j) and 12(k). After finishing the transportation task, the mobile robot returns to the charging station through the elevator; the charging station essential path is clarified in Figures 12(l) and 12(m); Figures 12(n) and 12(o) show the

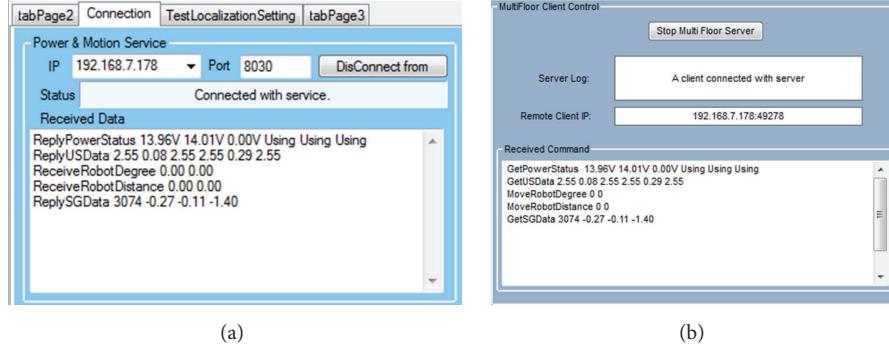


FIGURE 9: MFS and on-board computer client/server sockets.

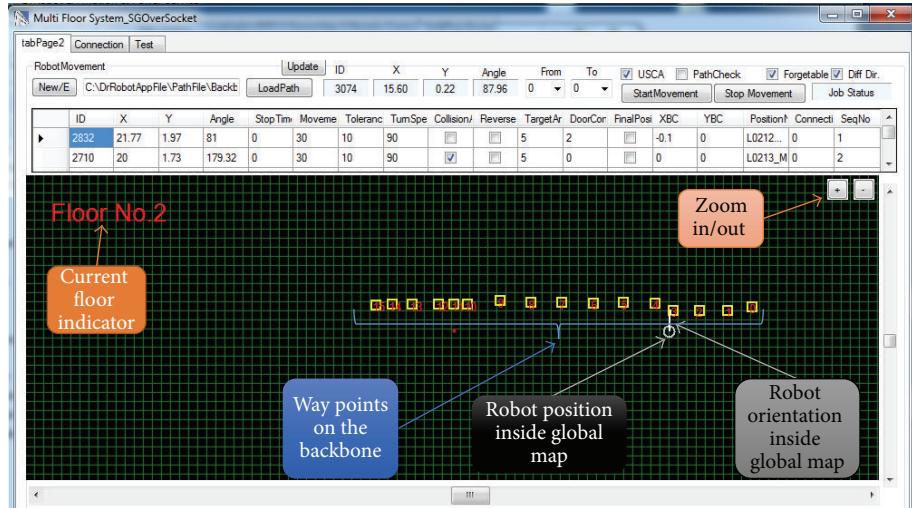


FIGURE 10: Multifloor GUI.

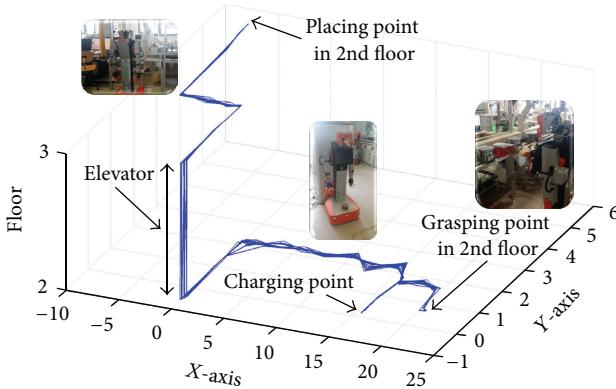


FIGURE 11: The real mobile robot transportation path on multifloor (repeating ten times).

charging procedure including the rotation and movement in backward direction till contacting the charge station. The robot executes the transportation task in multifloor environment efficiently and the internal systems have been verified.

The second experiment has been done to determine the repeatability of the different (grasping, placing, and charging) points in the transportation task. Repeatability is defined as the ability of mobile robot to reach the same position over a period of time while the accuracy is the difference between the reached point and optimal point. This test has been done in one floor for fifty times; each time the robot moves from the charging station towards position 1 for grasping operation, then moves till it reaches position 2 for placing, and returns back to the charging station, as clarified in Figure 13. This experiment has 100% success rate. Figure 14 clarifies the repeatability of the mobile robot for the important stations in the transportation path. In Figure 14(a) we notice that the maximum error reaching the grasping position is about 4 cm. The complicated and narrow area caused a lower mapping accuracy in this position than other tested positions. The repeatability of elevator entry button position had been investigated from the movement of mobile robot in multifloor experiment as clarified in Figure 14(d). The accuracy, mean, and the standard deviation for these points are listed in Table 2.

The last experiment was done for transportation tasks on a single floor to validate the integration of the MFS

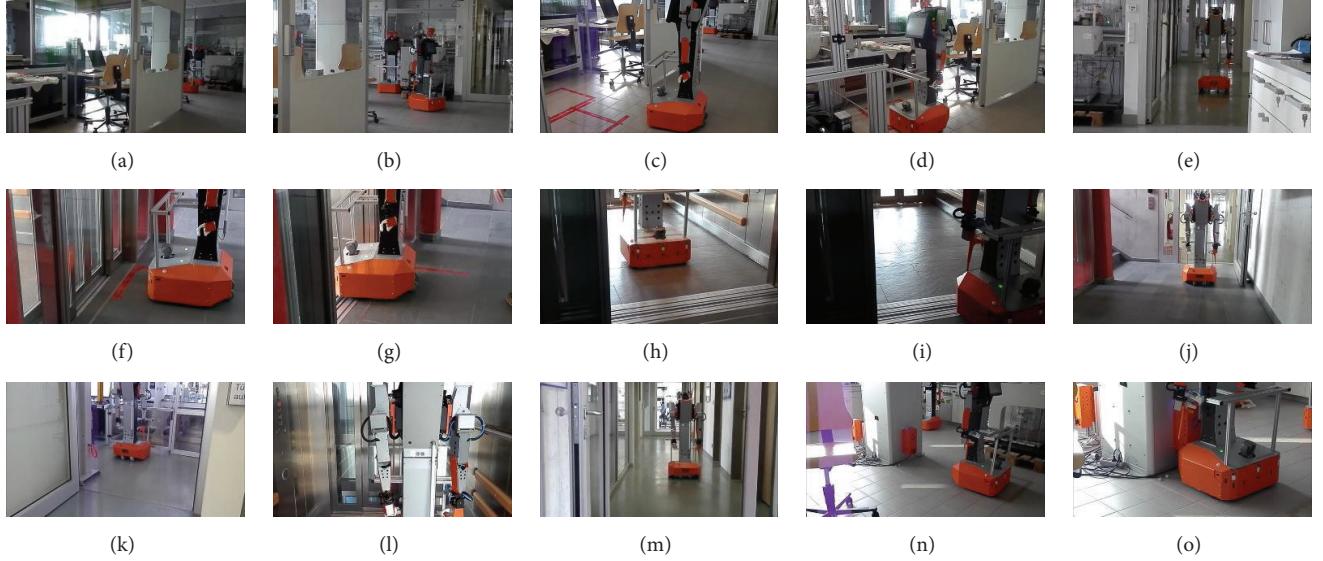


FIGURE 12: Transportation task in multifloor environment execution.

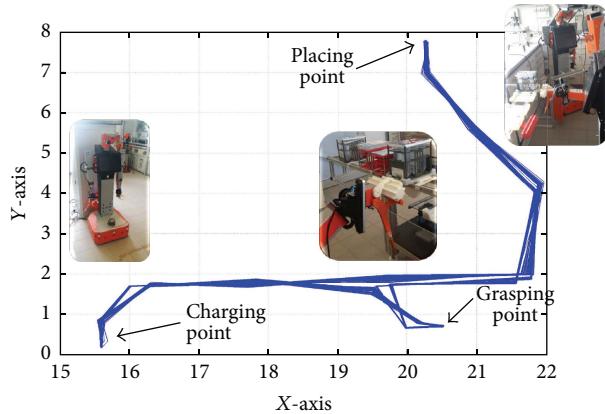


FIGURE 13: The real mobile robot transportation path on single floor (repeating fifty times).

TABLE 2: Accuracy, mean, and standard deviation of important robot position in meter.

	Axis	Accuracy	Mean	SD
Grasping point	X	0.04	20.506	0.008329931
	Y	0.01	0.702	0.011952286
Placing point	X	0.00	20.2682	0.014384
	Y	0.01	7.7676	0.012707
Charging point	X	0.01	15.596	0.006700594
	Y	0.02	0.1998	0.013013337
Elevator entry point	X	0.00	-0.14091	0.009439
	Y	0.01	0.938182	0.015374

with the multirobot management level (RRC). Once the RRC works, the connected robots will be listed as shown in Table 3. The RRC sends the transportation order to one of the available robots that has the highest charging rate to

TABLE 3: Connecting robot to RRC.

Robot ID	Robot IP	Connection status	Transportation status
1A	192.168.7.159	Connected	No task
2B	192.168.7.168	Disconnected	No task
3C	192.168.7.178	Disconnected	No task
4D	192.168.7.188	Connected	No task
5E	192.168.7.198	Disconnected	No task

execute the requested task. The MFS of the chosen robot receives this order as raw data (see Figure 15); then the ITMS parses it to receive the grasping/placing position as X, Y, and floor number. The execution of this task starts from the charge station, navigates towards point 1 (Grasping), then is directed to the placing position (point 2), and finally returns to the charge station. This experiment was repeated for ten times with 100% success rate in receiving the order from

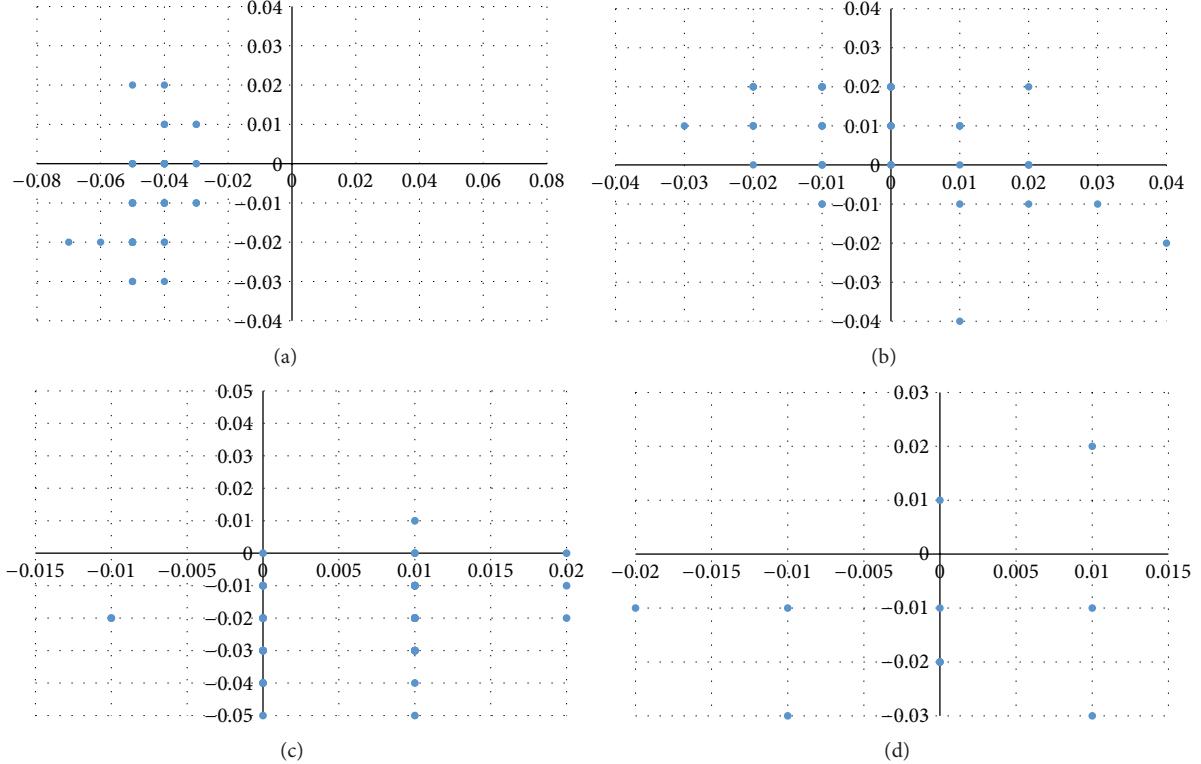


FIGURE 14: Robot movement position error. (a) Grasping position error. (b) Placing position error. (c) Charge station position error. (d) Entry button position error.



FIGURE 15: Received transportation order on MFS GUI.

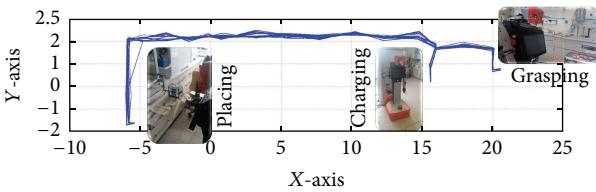


FIGURE 16: The real mobile robot transportation path on single floor.

higher level, controlling the automated doors in the robot path, and navigating to the grasping, placing, and charging position. The generated path of the mobile robot movements is shown in Figure 16. The accuracy and repeatability of the mobile robot in the important points (grasping, placing, and charging points) are demonstrated in Figure 17 while Table 4 lists the accuracy, mean, and standard deviation of these points.

TABLE 4: Accuracy, mean, and standard deviation of important robot position in meter.

	Axis	Accuracy	Mean	SD
Grasping point	X	0.001	20.55102	0.009717
	Y	0.019	0.729276	0.006221
Placing point	X	0.007	-5.39292	0.011949
	Y	0.009	-1.62057	0.007359
Charging point	X	0.002	15.59156	0.009182
	Y	-0.00768	0.212323	0.022121

(B) The mobile robot with Kinect sensor was moved towards the elevator as shown in Figure 18(a) to validate the elevator entry button detection algorithm. The detection program was started and tests were taken from many distances to detect the button and read the real position according to the mobile robot. The elevator entry button detection experiment has been repeated 100 times for each distance (45 cm, 50 cm, 55 cm, and 60 cm). The system can detect the entry button with high accuracy and speed. The average of the entry button detection success rate from different distances reaches 88.5% using the RGB detection method. Varying lightening condition against the Kinect vision sensor as shown in Figure 18(a) caused errors in detection. The developed method utilizes the HSL color representation instead of RGB representation to stabilize the color detection under different lightening condition. The success rate of the

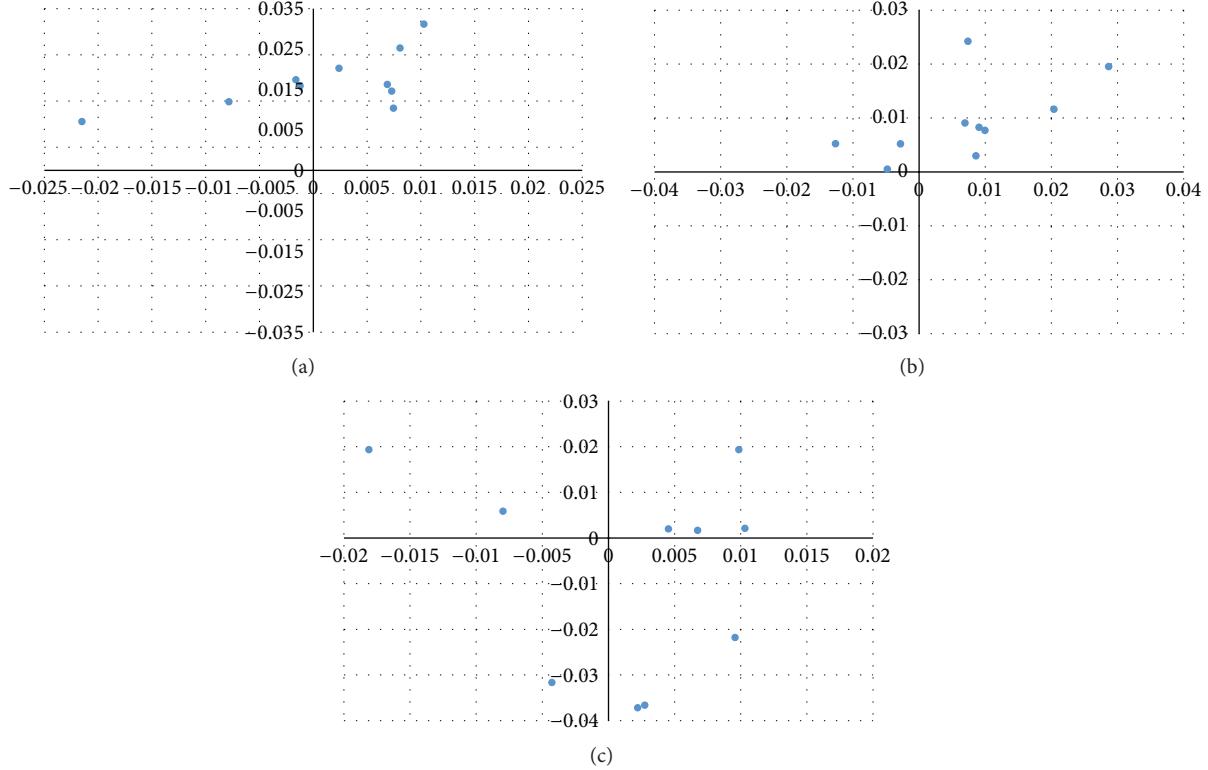


FIGURE 17: Robot movement position error. (a) Grasping position error. (b) Placing position error. (c) Charge station position error.

developed method reaches up to 99%. The contrast results between the HSL and RGB color representation detection methods are shown in Figure 18(b). The depth sensor is also analyzed for both methods; as clearly shown in Figure 18(c) the depth sensor starts to work in 50 cm distance.

The second experiment was executed to evaluate the performance of the internal button detection method. As a first step, the Kinect sensor has been placed in front of the internal buttons panel inside the elevator; then the internal button detection program starts and connects to the Kinect sensor. All buttons were selected one by one as a destination floor to be examined separately. Figure 19 shows the developed GUI for the elevator handler in operation; the left side shows the captured image with the red square drawn around the destination floor button. The detected button position in image coordinates is also shown in the same side, while the other side shows the processed image and the extracted real position. The collected candidates which work as inputs to the OCR engine are shown in Figure 20. This experiment was repeated 600 times to detect the specific button label with its real position. The internal button detection method showed a 98.66% success rate.

(C) The last experiment was performed twenty times to validate the robot arm pressing the elevator entry button as shown in Figure 21. A kinematic model has been built to control the robot arm. The robot position should be chosen correctly to put the entry button in the effective range of robot arm, in the same time allowing the robot to enter the elevator

when it is opened. The robot arm succeeds in pressing the button in the defined robot motion positions.

5. Conclusion

In this paper, a new method for multifloor transportation systems in distributed Life Science Laboratories is proposed. To meet all the requirements of life science automation, a ceiling landmark and a StarGazer module sensor are used for indoor localization. An innovative multifloor transportation system was developed using alone mode and relative map for multifloor mapping and localization, backbone path planning, and ITMS transportation management. The provided experiments show that the proposed approach is easily expandable to cover a large multifloor map, one navigation map for all the mobile robots which was impossible before [20]. Further the approach is applicable for any kind of mobile robots and has low costs, has high robustness, is ambient for light effects, and can work under dark conditions, which is not available in other localization systems. A robust high speed method is applied for elevator button detection (entry button and internal buttons). An IEEE 802.11g communication network with TCP/IP protocol is utilized, to establish an expandable communication network.

Competing Interests

The authors declare that they have no competing interests.

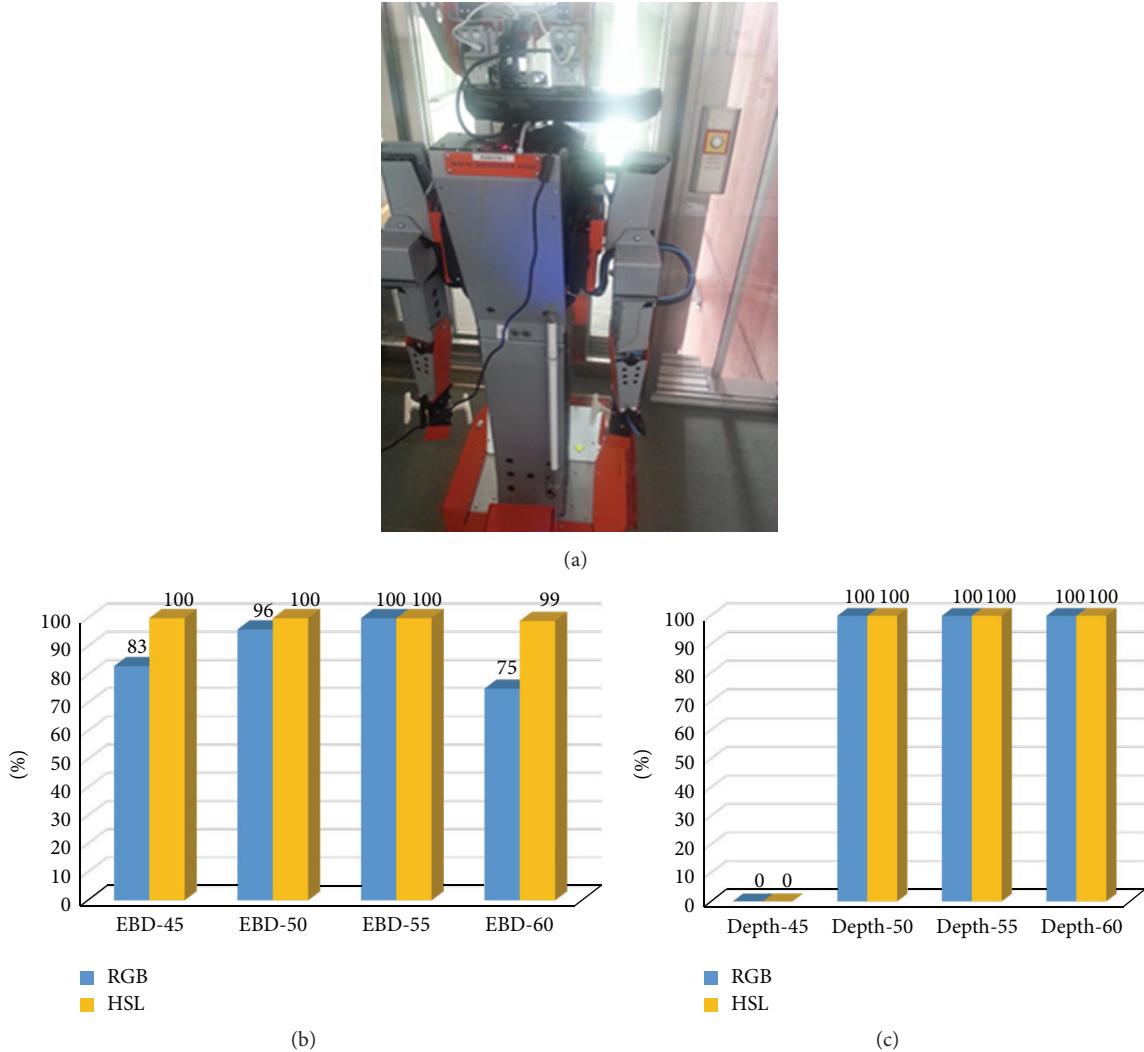


FIGURE 18: Experiment for elevator entry button detection. (a) Experiment environment for elevator entry button detection. (b) Detection success rate for RGB and HSL module. (c) Depth extraction success rate.

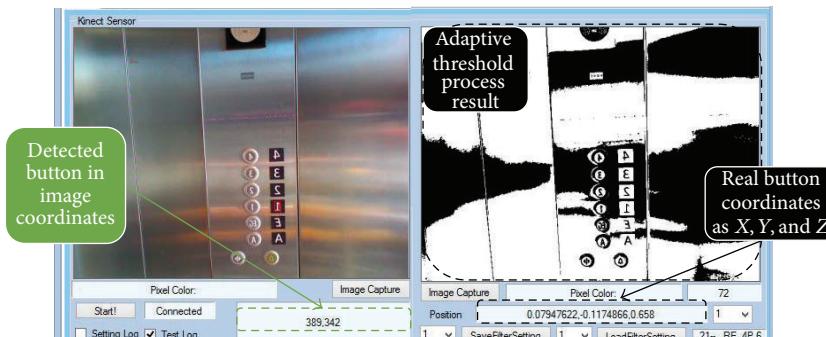


FIGURE 19: Elevator handler for internal buttons.

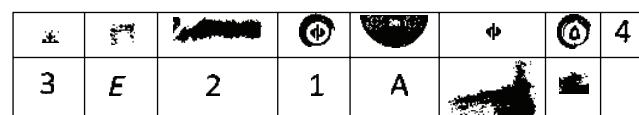


FIGURE 20: Extracted button candidates.

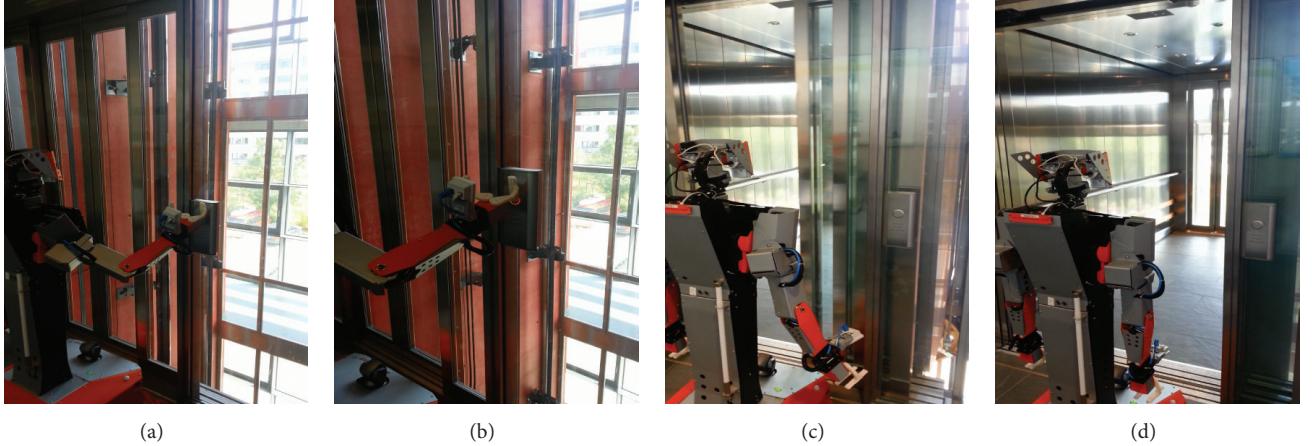


FIGURE 21: (a) Before the pressing. (b) Button pressing. (c) Robot arm returning to original position after pressing. (d) Ready to enter the elevator.

Acknowledgments

The authors would like to thank the Ministry of Higher Education and Scientific Research in Iraq for the scholarship provided by Mosul University, the Federal Ministry of Education and Research Germany for the financial support (FKZ: 03Z1KN11, 03Z1KII1), and the Dr Robot Company for the technical support of the H20 mobile robots.

References

- [1] D. S. Lütjohann, N. Jung, and S. Bräse, “Open source life science automation: design of experiments and data acquisition via ‘dial-a-device’,” *Chemometrics and Intelligent Laboratory Systems*, vol. 144, pp. 100–107, 2015.
- [2] B. Göde, S. Holzmüller-Laue, and K. Thurow, “New approach of standard-based end-to-end process automation in the life science automation,” *Chemie-Ingenieur-Technik*, vol. 87, no. 5, pp. 518–536, 2015.
- [3] J. G. Petersen, S. A. Bowyer, and F. R. Y. Baena, “Mass and friction optimization for natural motion in hands-on robotic surgery,” *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 201–213, 2016.
- [4] T. Chapman, “Lab automation and robotics: automation on the move,” *Nature*, vol. 421, no. 6923, pp. 661–666, 2003.
- [5] K. Sakaki, H. Esmaeilsabzali, S. Massah et al., “Localized, macromolecular transport for thin, adherent, single cells via an automated, single cell electroporation biomanipulator,” *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 11, pp. 3113–3123, 2013.
- [6] W. Gecks and S. T. Pedersen, “Robotics—an efficient tool for laboratory automation,” *IEEE Transactions on Industry Applications*, vol. 28, no. 4, pp. 938–944, 1992.
- [7] N. Miyata, J. Ota, T. Arai, and H. Asama, “Cooperative transport by multiple mobile robots in unknown static environments associated with real-time task assignment,” *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 769–780, 2002.
- [8] J. F. Montgomery, A. H. Fagg, and G. A. Bekey, “USC AFV-I: a behavior-based entry in the 1994 international aerial robotics competition,” *IEEE Expert*, vol. 10, no. 2, pp. 16–22, 1995.
- [9] S. W. Yoon, S.-B. Park, and J. S. Kim, “Kalman filter sensor fusion for Mecanum wheeled automated guided vehicle localization,” *Journal of Sensors*, vol. 2015, Article ID 347379, 7 pages, 2015.
- [10] J. López, C. Watkins, D. Pérez, and M. Díaz-Cacho, “Evaluating different landmark positioning systems within the RIDE architecture,” *Journal of Physical Agents*, vol. 7, no. 1, pp. 3–11, 2013.
- [11] Z. Mi, Y. Yang, and J. Y. Yang, “Restoring connectivity of mobile robotic sensor networks while avoiding obstacles,” *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4640–4650, 2015.
- [12] A. Birk, S. Schwertfeger, and K. Pathak, “A networking framework for teleoperation in safety, security, and rescue robotics,” *IEEE Wireless Communications*, vol. 16, no. 1, pp. 6–13, 2009.
- [13] R. M. K. Chetty, M. Singaperumal, and T. Nagarajan, “Distributed formation planning and navigation framework for wheeled mobile robots,” *Journal of Applied Sciences*, vol. 11, no. 9, pp. 1501–1509, 2011.
- [14] A. Ohya, A. Kosaka, and A. Kak, “Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing,” *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 969–978, 1998.
- [15] S.-W. Ma, X. Cui, H.-H. Lee, H.-R. Kim, J.-H. Lee, and H. Kim, “Robust elevator door recognition using LRF and camera,” *Journal of Institute of Control, Robotics and Systems*, vol. 18, no. 6, pp. 601–607, 2012.
- [16] J.-G. Kang, S.-Y. An, W.-S. Choi, and S.-Y. Oh, “Recognition and path planning strategy for autonomous navigation in the elevator environment,” *International Journal of Control, Automation and Systems*, vol. 8, no. 4, pp. 808–821, 2010.
- [17] P. K. Mohanty and D. R. Parhi, “A new hybrid optimization algorithm for multiple mobile robots navigation based on the CS-ANFIS approach,” *Memetic Computing*, vol. 7, no. 4, pp. 255–273, 2015.
- [18] M. H. Korayem, A. Zehfroosh, H. Tourajizadeh, and S. Manteghi, “Optimal motion planning of non-linear dynamic systems in the presence of obstacles and moving boundaries using SDRE: application on cable-suspended robot,” *Nonlinear Dynamics*, vol. 76, no. 2, pp. 1423–1441, 2014.
- [19] P. Bhargava, S. Krishnamoorthy, A. Shrivastava, A. K. Nakshathri, M. Mah, and A. Agrawala, “Locus: robust and calibration-free indoor localization, tracking and navigation

- for multi-story buildings,” *Journal of Location Based Services*, vol. 9, no. 3, pp. 187–208, 2015.
- [20] H. Liu, N. Stoll, S. Junginger, and K. Thurow, “Mobile robot for life science automation,” *International Journal of Advanced Robotic Systems*, vol. 10, pp. 1–14, 2013.
- [21] “Dr Robot Inc.: WiFi 802.11 robot, Network-based Robot, robotic, robot kit, humanoid robot, OEM solution,” http://www.drrobot.com/products_H20.asp.
- [22] X. Yu, L. Dong, Y. Li, and K. E. Hoe, “Lift-button detection and recognition for service robot in buildings,” in *Proceedings of the 16th IEEE International Conference on Image Processing (ICIP '09)*, pp. 313–316, IEEE, Cairo, Egypt, November 2009.
- [23] J. Baek and M. Lee, “A study on detecting elevator entrance door using stereo vision in multi floor environment,” in *Proceedings of the ICROS-SICE International Joint Conference (ICCAS-SICE '09)*, pp. 1370–1373, Fukuoka, Japan, August 2009.
- [24] J.-G. Kang, S.-Y. An, and S.-Y. Oh, “Navigation strategy for the service robot in the elevator environment,” in *Proceedings of the International Conference on Control, Automation and Systems (ICCAS '07)*, pp. 1092–1097, October 2007.
- [25] H. Chae, S.-H. Kim, Y.-C. Lee, and Christiand, “Applications of robot navigation based on artificial landmark in large scale public space,” in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO '11)*, pp. 721–726, December 2011.
- [26] I. Ul-Haque and E. Prassler, “Experimental evaluation of a low-cost mobile robot localization technique for large indoor public environments,” in *Proceedings of the Joint 41st International Symposium on Robotics (ISR '10) and 6th German Conference on Robotics (ROBOTIK '10)*, pp. 1–7, Munich, Germany, June 2010.
- [27] D. Troniak, J. Sattar, A. Gupta et al., “Charlie rides the elevator—integrating vision, navigation and manipulation towards multi-floor robot locomotion,” in *Proceedings of the 10th International Conference on Computer and Robot Vision (CRV '13)*, pp. 1–8, Regina, Canada, May 2013.
- [28] HAGISONIC: StarGazer Module, Localization Sensor, <http://www.hagisonic.com/>.
- [29] J. K. Goyal and K. S. Nagla, “A new approach of path planning for mobile robots,” in *Proceedings of the 3rd International Conference on Advances in Computing, Communications and Informatics (ICACCI '14)*, pp. 863–867, IEEE, New Delhi, India, September 2014.
- [30] B. Park, J. Choi, and W. K. Chung, “An efficient mobile robot path planning using hierarchical roadmap representation in indoor environment,” in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '12)*, pp. 180–186, Saint Paul, Minn, USA, May 2012.
- [31] J. Wang, Y. Sun, Z. Liu, P. Yang, and T. Lin, “Route planning based on floyd algorithm for intelligence transportation system,” in *Proceedings of the IEEE International Conference on Integration Technology (ICIT '07)*, pp. 544–546, Shenzhen, China, March 2007.
- [32] A. A. Abdulla, H. Liu, N. Stoll, and K. Thurow, “Multi-floor navigation method for mobile robot transportation based on StarGazer sensors in life science automation,” in *Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC '15)*, pp. 428–433, Pisa, Italy, May 2015.
- [33] “Department of Justice ADA Title III Regulation 28 CFR Part 36,” 1991, <http://www.ada.gov/reg3a.html#Anchor-11540>.

