IJCRT.ORG ISSN : 2320-2882
INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

# EMPLOYMENT OF LOW-COST AUTONOMOUS ROBOT IN SMART RESTAURANT

[1]Chandravadhana S., [2]Ohmsakthi vel R., [3]Nandhakumar R.

[1]Professor, [2]Assistant Professor, [3]Assistant Professor
[1]Department of Mechatronics Engineering,
[1]Agni College of Technology, Chennai, Tamil Nadu, India

***Abstract:*** Over the necessity and emerging development in mechatronics field, numbers of low cost automated products were raised to innovate and made human life more comfortable. This paper aims to design and employ an autonomous system to automate the various functionalities in restaurant. Because of shortage of labors, and the technology development, the restaurant owner shows their interest to make their restaurant as smart. For this, we are integrating some emerging technologies to achieve automation. Here, we used IoT, embedded devices and robots. Unlike other waiter robots, our robots are structured and programmed with ROS framework, AMLC and SLAM, which makes our system more intelligent and smart. Preliminary tests show that autonomous system work is acceptable, and the robot is able to reach the target table and deliver the food. The system can guide and help the customer in each level of occurrence.

***Index Terms* - Smart Restaurant, Low Cost Automation, IoT, Robotics and Embedded Networking**

## I. INTRODUCTION

As we know that emerging technologies would pay a great way to the most comfortable life as people's wish. The technological development raises the development of whole nation. The new emerging high valued intelligent systems shapes the cultures in various aspects. Various automated devices were developed for automation in industry and home. This expectation motivates us to create an automated product to serve in restaurant for better hospitality. Initially everyone raises a question, is it technically and economically feasible or not. We got motivation and started a work to design and implement a prototype for achieving automation in restaurant. Initially various research work has completed to automate completely in the restaurant using Graphical User Interface (GUI). In that work, they implement an automated product to deliver the things in time and those activities gets monitored by a technical person through graphical interface. In our work we are going to implement an automated system for the whole restaurant from greeting the customer, allocate the table, taking order from the customer, delivering the same to them in time and finally ends up with billing. By this product, the hotelier can satisfy the customer by reducing the idle time. Our automation process gets start from the customer's entry at the entrance itself. Entrance module (E1) can communicate with customer and allocate the table number and the security key. Then the customer occupied the allocated seat, meanwhile the entrance module sends the table number and number of customer to the cashier module (M1). In that table, the customer can order the food dishes through wireless technology, available in customer module (C1). C1 module transmit the digital data to the Chef module (D1) and display it with its table number, then only the chef can identified the order by whom. The trolley module (Wx) normally a mecanum wheeled robot carries the prepared food and docked the target table and delivers the food in the table. The trolley gets backs to its original position. Since the hotelier can accommodate two or three ROS based system, they can able to deliver the food items in short time. The customer can pay their bill and then leave restaurant with more comfort and satisfaction.

## II. TECHNOLOGIES INVOLVED

The synergetic combination of the three emerging technologies IoT, Embedded and Robotic Technology has lead and develop the low cost Autonomous systems rapidly with a high level adaption of structured networks. The autonomous systems are self-optimized, self-protected, self-healing and self-configuration.

### 2.1 Internet of Things (IoT)

A wide range of IoT technologies are currently employed in the autonomous systems. The IoT technologies and applications are bringing conceptual understanding of how the technology makes the business success and comfort and luxurious life to people. Citizen centric IoT open environments require tackling new technological trends and challenges. In this work, we are working to integrate IoT infrastructure and services with Robotics and embedded hardware to implement low cost autonomous systems. The system has implemented to achieve advanced functionality, along with novel applications, and new business models and investment opportunities. The autonomous sensor interfaces for the design of smart restaurant architecture consists of the input and output function modules, as mentioned in Figure 1. The entire system is controlled by the Arduino UNO board, which is programmed on the basis of our requirement. The IoT architecture comprises of the six modules with various units (Power supply, Sensors, Actuators, Wi-fi Module, Keypads, Wheels, Servo motors, LCD display, Arduino board with user interfaces and transceivers). The integration of those components makes the installation of the autonomous system easier to control, interface and provides optimal performance.

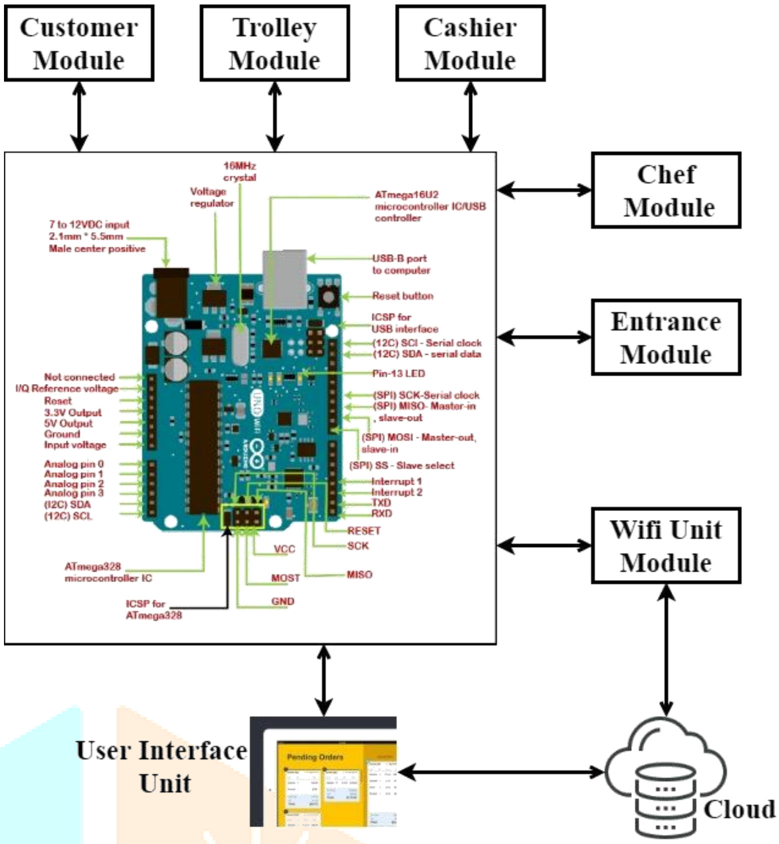


**Fig. 1 –** Autonomous Interfaces using IoT Technology

## 2.2 Embedded Networking

Industrial automation applications are used on factory floors contain most of the elements applicable to embedded systems and embedded networking. Looking at an embedded networking system from the industrial angle not only helps us to understand basic communication requirements, but this model can also easily be adapted to a variety of embedded systems.

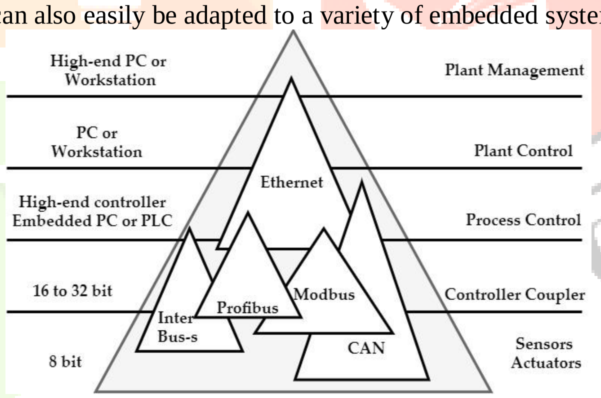


**Fig. 2 –** Communication in the automation pyramid

Figure 2 shows a variation of the industrial automation pyramid. It symbolizes the hierarchy in an industrial automation system. At the bottom is the sensor and actuator level with input and output elements that directly read switches and sensors like current speed of a conveyor belt, RPM valves from anything rotating or a current temperature. Typical actuators include hydraulic or pneumatic elements or electrical motors, which in industrial lingo are usually referred to as drives. Being at the bottom of the pyramid also symbolizes that in any installation these elements or modules are used in the highest quality, compared to modules in the layers above. Due to higher quantity, these modules are often price sensitive, as the price-per-module is multiplied by the large number of devices required. In the not-so-distant past of some decades ago, communication at this level was not computerized, meaning that every sensor or actuator was directly connected with its own set of wires to the next higher control level. Today, the trend is to equip more and more sensors and actuators with a networking interface. However, since single components on the lowest level are needed in large quantities, cost is still a major issue. Equipping simple sensors (which report a single or a few values) with high-performance processors and high-end network adaptors like Ethernet is simply not an option. Other technologies typically based on serial buses have been used for years because they can handle by some of the lowest performance (but not affordable) microprocessors and microcontrollers. CAN- the Controller Area Network – is just one of many contenders in the field of networking technologies that are suitable to reach into the lowest level of the automation pyramid. The next layer up is the Controller level. In this level controllers are used to collect all the inputs, perform some sort of control algorithm and transmit the appropriate commands back to the actuators, the output. The next layer of the automation pyramid is of only limited concern for embedded networking. With each level up in the pyramid the performance of systems needs to be higher, as it needs to handle multiple systems from the layer below. The communication requirements become more significant in the upper levels as more bandwidth is required to handle all the accumulated information coming from the multiple systems in the layers below. In these levels, interfaces to embedded networks are only used if a direct link to the lowest levels is required. Embedded systems using multiple microcontrollers and any sort of communication between them can often be directly compared to the lowest levels of the automation pyramid. There will be some sensors and actuators for the inputs and outputs and some sort of controller. Sometimes there might be truly distributed control (in which case the controller functionality is divided between modules) however the basic model and its consequences still apply. The closer a module is to the sensor and actuator level (or to the inputs and outputs), the more cost-sensitive it is and the more basic the communication requirements.

### 2.3 Robotic Technology

The use of robots as food servers in multicuisine restaurants is on an emerging trend in hospitality operations. Restaurant owners have shown an interest to use robots to aid in serving food to the customers, due to shortage of waiters. We have developed a prototype of a mobile waiter robot for trial runs in a dining restaurant. Unlike existing waiter robots, it is of low-cost autonomous with high payload and is able to travel and reach the target table without any collision. Though, every restaurant has their own interior designs and layouts, Robot Operating System (ROS) framework and it modular design helps our autonomous system can be more efficient and highly optimized. In ROS, trajectory path planning and navigation within a restaurant can effectively achieved by generating digital map of the environment (by SLAM algorithm) and position of the robot can be assessed by Adaptive Monte-Carlo (AMCL). Trials and tests show that autonomous navigation is achieved, and the robot is able to dock at a target table and roll out the serving tray. We have developed a mecanum wheel based prototype for use in casual dining restaurants. It is independent in navigation rather than following a designed path and this system is programmed and developed to reach at the table and delivers the trays to the dining table instead of the customer to receive the tray from it. This robot also has the capability to carry and deliver more trays than some of the present robots which is typically able to carry single tray. The waiter robot has designed with a base and an automatic three level dumbwaiter. The dumbwaiter is closed one, which helps to keep the food warm and clean. Once reached the target table, a door near dumbwaiter is lowered and the required tray is raised to the door and is rolled out onto the door. The base houses the motors driving four mecanum wheels, and other peripheral devices and components. The power source consists of two 24V lithium polymer (LiPo) batteries to supply power to the robot. The height of the robot is flexible to serve table with different heights by means of motorized lifting mechanism between base and the dumbwaiter. The robot and its motion can be simulated in the RVIZ space.

**Fig. 3** – Robot Structure (Prototype)

## III. System Deployment

### 3.1 Entrance Module

This module initiates the whole automation process at the entrance itself, hence it is called by so. Once the customer reaches the sensor reachable area, this module gets actuated and enquires the question "How many members are going to accommodate?" Once the customer replied by using available keypad, the controller checks for availability in and allocates the table with security code. If all the tables get accommodated, then the controller informed the customer to wait in common room. All the above collected information along with security code gets informed to cashier module by means of wireless transmission mode and also gets stored in the cloud for future purpose.

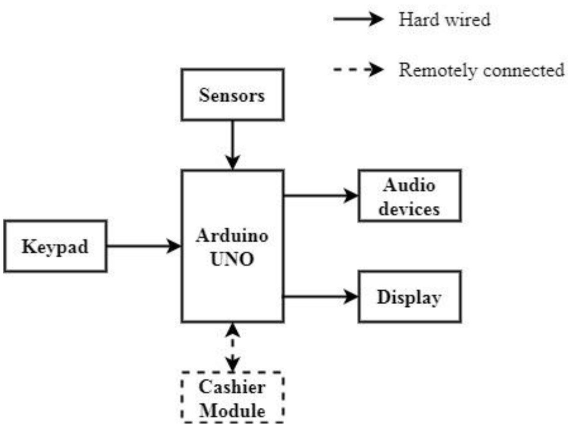


**Fig. 4** – Entrance Module

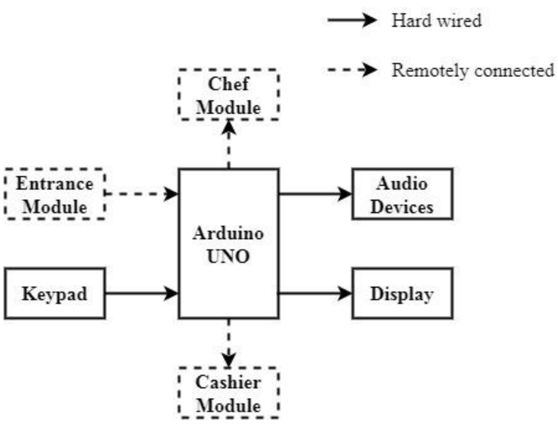


**Fig. 5** – Customer Module

### 3.2 Customer Module

Once the customer occupies the allocated table, they can order food items through the touch pad available in that table. Here, specially equipped devices are available for the choosing and confirming the food items and can get the details of the estimated wait time. Those digital data gets transmitted to both chef module as well as the cashier module. There is a inter communication between chef and customer module to avail the status of all food items. This module pays a way to create a communication with chef directly. By this process the idle time gets reduced as much for the customers.

### 3.3 Cashier Module

This is the heart of the system, which receives the different signals from various modules (Entrance Module, Chef Module and Customer Module). In this stage, the cashier manually checks the bill amount for the respective table number with some security code. Also, the cashier manually resets the table status (occupied / vacant) through Graphical User Interface (GUI). Here, all the collected data are stored in cloud using IoT technology for future purpose.

### 3.4 Chef Module

This module is nothing but the basic micro controller equipped with a RF receiver and the display unit. The receiver receives the signal from RF transmitter availed in Customer module. Then the microcontroller display the received data in the display with the identification number as table number. Once the display starts functions the chef gets alerts to prepare the ordered food.

### 3.5 Trolley Module

This trolley module is nothing but the prototype of mecanum wheeled robot with dumbwaiter. Once the chef prepared the food items, he can keep the dishes in the trolley. The trolley receives the input from the chef through keypad manually, then starts functioning to reach the respective table using SLAM algorithm The trolley delivers all the food items and starts to its home position by completing full circle.

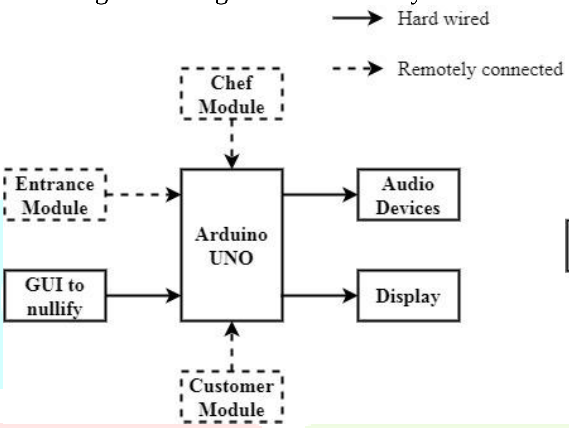


**Fig. 6** – Cashier Module

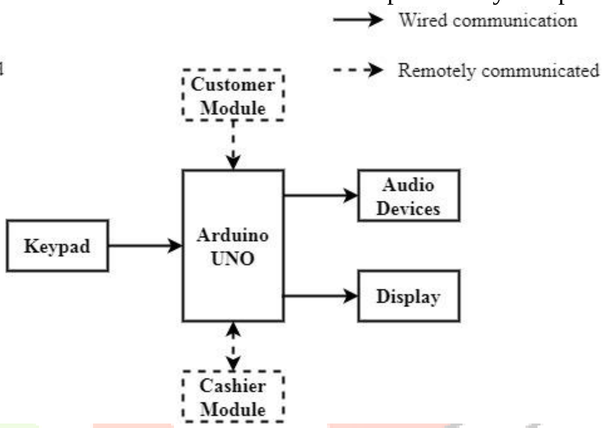


**Fig. 7** – Chef Module

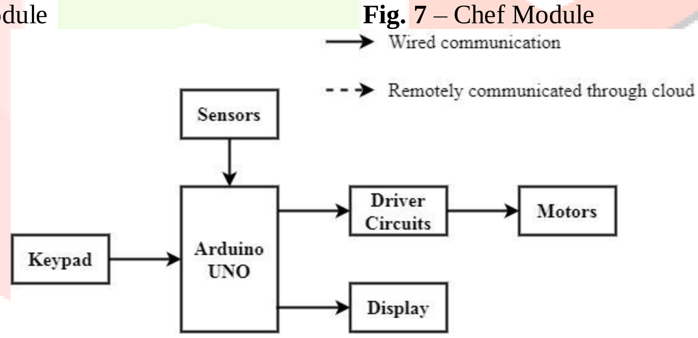


**Fig. 8** – Trolley Module

### 3.6 Localization and Navigation

In our work, the navigation has been achieved by ROS, which facilitates the availability of odometry, SLAM, localization, and path planning. For robot operation, various motion sensors are employed to estimate the position of the robot is referred as odometry. The observed data from the wheel are being displayed as an accepted ROS message using Arduino UNO. However, odometry gets accumulated with errors over time because of the wheel's slip, wear and tear. The information about orientation and heading is collected by the Attitude Heading Reporting System (AHRS) equipped with triple axis gyroscope, accelerometer and magnetometer. A Low cost laser scanner (RP-Lidar) is used to generate a map for the robot. In a typical mobile robot application, the robot is driven in the restaurant to generate a computer map. As the robot moves, range data from a LiDAR is collected and used by a SLAM algorithm to produce a map. The ability for a robot to locate itself in an environment is a common problem in mobile robots (Fox et al., 1998). Adaptive-Monte Carlo Localization (AMCL) uses a probabilistic localization technique and particle filters to track the pose of the robot against a map loaded from the map server. AMCL (also known as particle filter localization) is an algorithm for robots to localize using a particle filter. Given a map of the environment, the algorithm estimates the position and orientation of a robot as it moves and senses the environment. The algorithm uses a particle filter to represent the distribution of likely states, with each particle representing a possible state, i.e., a hypothesis of where the robot is. The algorithm typically starts with a uniform random distribution of particles over the configuration space, meaning the robot has no information about where it is and assumes it is equally likely to be at any point in space. Whenever the robot moves, it shifts the particles to predict its new state after the movement. Whenever the robot senses something, the particles are resampled based on recursive Bayesian estimation, i.e., how well the actual sensed data correlate with the predicted state. Ultimately, the particles should converge towards the actual position of the robot.

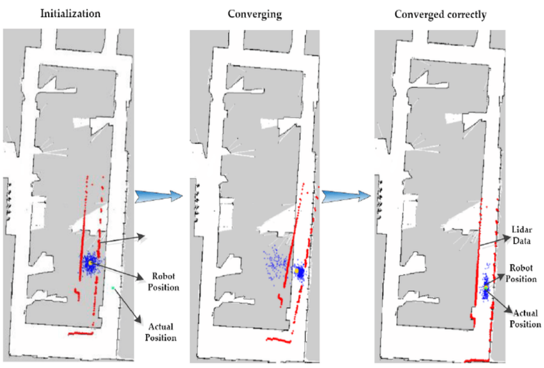


**Fig. 9** – ACML process

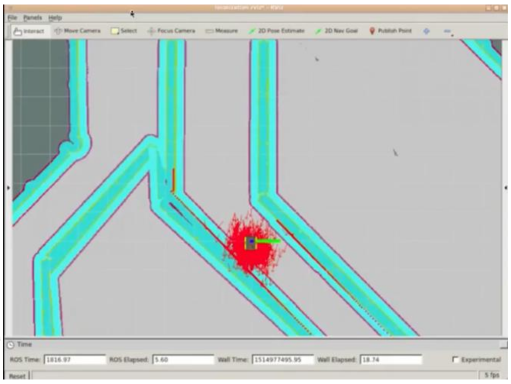

**Fig. 10** – Road Map and Navigation

Fig. 10 shows the robot on the cost-map. Dark green and yellow colour indicates the boundary of the environment for the global and local Costmaps respectively, while grey area is the space that the robot can move in. The light green line indicates the path planned which the robot will navigate through.

## IV. Experiments and Results

There are two considerations, related to locate and navigate the robot to deliver food to the right table. The first is the global navigation issue (navigating and locating the waiter robot to the target table) and the second is local navigation (recognizing and docking at the correct table). To solve the first one, the location of the target table and robot must be known. Two approaches were available with: using Indoor Positioning System (IPS), and using the ROS navigation and localization modules. The objective is to get the robot close to the table to effect docking. With mapping, odometry and motor actuation errors, it is unlikely that the robot will be sufficiently close to the target table. It is also unlikely to be parallel to the table to dock and to deliver the food items. Two approaches were experimented with it. One uses IR range finders for docking. Two infra-red sensors located on the robot will provide distances to the edge of the table. A program is written to adjust the robot until the two sets of readings are about 0.5 m to the edge of the table. Once the robot is almost parallel, the robot will begin its serving routine. The docking sequence and program is implemented in ROS and the flow chart of this implementation is show in Fig. 11. In the second approach the robot control uses a pixy camera with pan and tilt angles. The camera tracks a colour tag and outputs the pan and tilt angles to the robot controller closer to the table. Results are encouraging but improvements to use pixy to track the flashing LED of the pager is being evaluated.

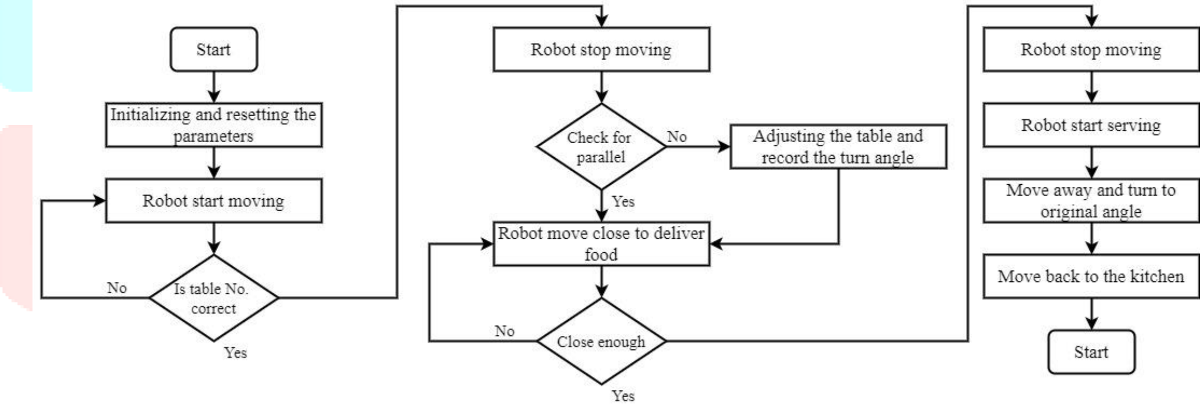


**Fig. 11 -** Flowchart

The development and implementation of the ROS based autonomous system as waiter, improves productivity and helps in employing waiter robots in various restaurants, where interior designs and table arrangements vary so much. The layout plan can be converted to a ROS map efficiently by a single person using an in-house designed movable stable platform carrying the LiDAR and SLAM. The robot is able to localize itself with respect to this map using ROS AMCL process. However, as observed in our tests, robot navigation is affected by real tables and chairs, and it became necessary to use mechanisms to bring the robot closer to the table to serve the food. This will make the use of autonomous robots navigation and localization more robust and suitable for use in different types of food outlets.

## V. Conclusion

By the end of our work, the individual functions – modelling, mapping, navigation, programs developments and the docking, are developed and evaluated in the ROS based robot embedded controller. The prototype pf the robot was tested in terms of localization, navigation, finding target table and docking. The results are much appreciable and gave us a new hope to proceed further. The work presented here, creates an innovation among budding engineers to develop an autonomous system to deliver food. The mobile robotics part of the system is best developed in ROS as there is a community of researchers constantly improving the implementation of planning, navigation and localization algorithms of robots. Deploying the waiter robot is much faster in different food outlet floor plans and table arrangements.

## REFERENCES


[1] Ang Benson (2016), Robot Lucy at your service at newly opened Rong Heng Seafood, The Strait Times Lifestye, http://www.straitstimes.com/lifestyle/food/robot-lucy-at-your-service[Accessed 16 Mar 2016]

[2] Fox, D., Burgard, W. and Thrun, S. (1998). Active Markov localization for mobile robots.Robotics and Autonomous Systems, 25(3-4), pp.195-207.

[3] Kohlbrecher, S., Meyer, J., Graber, T., Petersen, K., von Stryk, O. and Klingauf, U (2013). Hector Open Source Modules for Autonomous Mapping and Navigation with Rescue Robots Proceedings of 17th Robo Cup international symposium

[4] Lin LC and Shih HY (2013), Modeling and Adaptive Control of an Omni-Mecanum-Wheeled Robot, Intelligent Control and Automation, 4, 166-179.

[5] P. Velsquez, L. Vsquez, C. Correa, and D. Rivera. (2017) “A low-cost IoT based environmental monitoring system. A citizen approach to pollution awareness.” Chilean Conference on Electrical and Electronic Eng. Inf. Commun. Tech. (CHILECON) 2017: 16

[6] F. Wu, C. Rdiger, and M. R. Yuce. (2017) “Real-time performance of a self-powered environmental IoT sensor network system.” Sensors 17(2): 282.

[7] Rajiv. 2018) “Specifications and Applications of Narrow Band IoT (NB-IoT).” [Online]. Available: https://www.rfpage.com/specifications-and-applications-of-narrow-band-iot/. [Accessed: 01-Dec-2018].

[8] K.Yelamarthi, M.S.Aman, A. Abdelgawad. (2017) “An Application-Driven Modular IoT Architecture.” *Wireless Communications and Mobile Computing* **17:**116.

[9] Andrew Howard, Maja J Mataric,´ and Gaurav S. Sukhatme. Relaxation on a mesh: a formalism for generalized localization. In Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS), pages 1055–1060, Wailea, Hawaii, October 2001

[10] Barry Brian Werger. Ayllu: Distributed port-arbitrated behavior-based control. In Lynne E. Parker, George Bekey, and Jacob Barhen, editors, Distributed Autonomous Robotic Systems 4, pages 25–34. Springer-Verlag, Knoxville, Tennessee, October 2000.