TEST Engineering & Management

March-April 2020
ISSN: 0193-4120 Page No. 1385 - 1390

Automated Car Parking

[1]Abrar Wafa, [2]Bara'ah Samkari, [3]Noor Zeba, [4]Sarah Al-Jaber, [5]Mohammed Shehata
[1,2,3,4,5]College of Engineering, EFFAT University, An Nazlah Al Yamaniyyah, Jeddah, 22332, Saudi Arabia
[1]awafa@effatuniversity.edu.sa, [2]effatuniversity.edu.sa, [3]nkailani@effatuniversity.edu.sa,
[4]saljaber@effatuniversity.edu.sa, [5]mshehata@effatuniversity.edu.sa

Article Info
Volume 83
Page Number: 1385 - 1390
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 14 March 2020*

Abstract
Mechanized auto parking framework could be utilized for private structures, inns, workplaces, strip mall and show rooms, colleges, government structures, airplane terminals, clinics, and stadium. The upsides of mechanized auto parking are effective use of spaces; diminishing the land space and expanding the quantity of stopped vehicles. In this work, a model of computerized auto parking framework to stop and recover autos consequently in simple and adequate way was fabricate. The philosophy that was utilized to accomplish the point of the undertaking was by making a model that uses NXT mechanical autonomy to assemble the computerized auto parking. NXT Robotics Kit was utilized to execute the forklifts to stop and recover the autos and used to fabricate the underground pile of beds at the carport and in addition the parking structure. The framework that were accomplished are the forward/in reverse and up/down developments to stop/recover autos from/to assigned spots, the up/down developments of the stack's beds. Additionally, extraordinary sorts of sensors were utilized to control the development. Besides, the framework's legitimacy was checked by testing part-by-part segments and additionally testing the entire framework.

Keywords: *Car park; automated; robotics; multilevel*

1. Introduction

The parking framework is at present finished in our standard life. There are various sorts of help attracted with parking structures The driver needs to scan for a parking space in parking zone [1]. Keeping in mind the end goal to diminish the exertion of the driver, the parking structures give direction to the driver to stop the vehicle. In light of this idea different parking framework are made. The data transmissions are made by adaptable and by web benefits in two or three parking framework. By then exceptional sensors and advancements are utilized for acknowledging parking framework [2].

The auto Parking Framework is before long an unbelievable structure by utilizing unmistakable advances and affected gets some information about. The Smart Parking Framework is finished in different conditions with different highlights, which manage their issues looked in their customary exercises, and these structures advantage the rich [3]. The Framework outlining was guided by the standard of layered handiness. There are three tired functionalities, the most diminished level including the segments of perceiving, information transmission is made in a center level, and upper level manages the breaking point of information, dealing with and customer interfaces [4].

There are couple of sorts of parking framework. In Centralized Assisted Parking Search (CAPS), the data dealing with will lay on the focal processor (server). The server will amass data from a sensor in the parking space and take choices freely. By the by, this unified framework always has some challenges [5, 6].

There is another method called Opportunistically Assisted Parking Search (OAPS), which utilizes a Mobile Storage Node (MSN) [7, 8]. The data stream is made with this MSN. This prompts increment help through its capacity. In any case, the data scattered by this middle isn't constantly helpful.

The Parking Guidance and Information System (PGIS) can be utilized for plans, for example, whole city zone or specific parking condition [9]. This System offers

March-April 2020
ISSN: 0193-4120 Page No. 1385 - 1390

data to drivers regarding the shot of spaces for parking auto in the parking zone. The data is sent by Static/interesting variable message signs (VMS) [10]. The heading required are given by this PGIS, which has four parts. The four bits are data spread instrument, data assembling part, control focus and media transmission structures. With the assistance of a helpful the contraption, the stream position of a driver is seen by utilizing Global Situating System (GPS), and this structure is made by utilizing a web [11].

The Transit based data structure (TBIS) gives direction to stop and-ride working environments [12]. This structure wins concerning acknowledging, and it diminishes the inconvenience of the clients [12].

PC controlled section are utilized in robotized parking, where the security highlights are open [13]. This framework puts the auto in its apportioned space by PC controlled docks/lifts and once in a while client interest is required. A presentation of three-level programming outline which wires: Logical Layer (LL), Safety Layer (SL) and Hardware Abstraction Layer (HAL) to finish the benefit and great hoarding of vehicles protected [14,15].

The Smart Parking Framework is seen as favorable for the Car Park heads, Car Park supporters and furthermore in condition safeguarding. For the Car Park heads, the information collected by methods for the utilization of the Smart Parking Framework can be abused to envision future parking plans. Advocates are similarly prepared to benefit by Smart Parking Framework as Parking space can be totally utilized with a more powerful and upgraded framework.

Along these lines in this work, the primary point of computerized auto parking administration was to change more autos in a similar space, in this manner lessening the space expected to stop a similar number of autos. It is a technique for naturally parking and recovering autos. Thus, a model of staggered robotized auto parking administration was composed and actualized. This administration is completely computerized where the driver enters a numerical key and a determination, and after that the framework grabs the vehicle and stores it in an assigned parking spot. Upon recovery, the driver enters the key and the choice, and the framework grabs the auto and conveys it to the exit.

2. Prototype Description

2.1 Overall Description

At the point when an auto enters the passage region of the robotized stopping framework, an IR discovery subsystem identifies its quality. At that point the driver is elevated to enter a substantial key and to pick the choice of either stopping or recovering the auto. Each key is checked for precision and appointed an assigned parking space. After entering the right key, the auto is gotten alongside the bed from the stack framework and put in the assigned spot. At the point when the driver comes back to get the auto, he enters the substantial key for which the framework will check in its database (which contains the arrangements of every single stopped auto) and the auto is returned back to the carport. The stack framework will push down the beds to prepare for the approaching bed. The framework incorporates a mechanical lift with engines for picking the auto and putting it in the assigning spot.

2.2 Features

The NXT microcontroller is a 32-bit microprocessor with a large matrix display. It consists of 4 input ports and 3 output ports. It contains of a Bluetooth and USB communication link. It can control many stepper motors and many different types of sensors. It was easily programmed via the PC or MAC. The Stepper Motor has a worked in pivot sensor that estimates speed and separation, and reports back to the microcontroller. These consider exact advances and finish engine control inside one level of precision. A few engines were adjusted to drive at a similar speed. The circuits for the IR Transmitter and recipient were manufactured. The recipient is associated with the PC through the parallel port. The transmitter is given a yield all an opportunity to identify the nearness of an auto at the passage. Two arrangements of IR sensors situated at better places were utilized to identify the nearness of the auto.

3. Design Overview

3.1 Garage Block

The garage for the car parking is just a structure with metal scaffoldings to support the cars that are parked by the forklift. The forklift places the car along with the pallet in this garage. The dimensions of the garage are 17.95cm × 8.95cm × 8.95cm (L × W × H). The front view of the garage is shown in Figure 1.

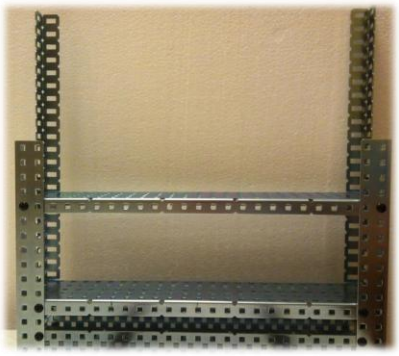

Figure 1: Garage Front View

3.2 IR Sensor Circuit

In the IR circuit, the transmitter circuit has an input voltage which is equal to 6 V, infra red rays, LED to know if there is a current is passing and a resistor value equal to 100 ohms (depends on the distance to be measured) to control the amount of current. The receiver

March-April 2020
ISSN: 0193-4120 Page No. 1385 - 1390

circuit has the output voltage from the transmitter which is very low, which is fed into a transistor to amplify it, the transistor sinks the current and increases the output to the parallel port into the computer. The IR circuit diagram is shown in Figure 2.

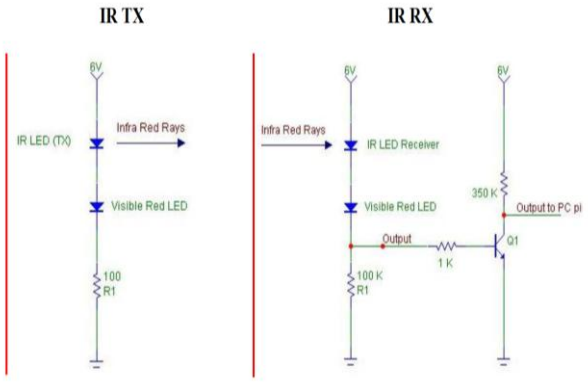

Figure 2: IR circuit

3.3 IR Sensor Circuit

The Push-Up Stack is the mechanism through which the pallets become available for the car to park. This stack consists of pallets on which the car is carried by the forklift and this system is connected to the NXT microcontroller. The stack's top is always aligned with the driveway (Figure 3. Every time a pallet is taken to a parking spot, the stack pushes up to realign the top of it with the drive way. However, when a car is retrieved, the stack pushes down the pallets. This is accomplished via the NXT, where a stepper motor, that is controlled by the NXT microcontroller; is used along with a gear and a string to transfer the rotational movement from the motor to a linear movement. The stepper motor moves a rotational step per instruction, and a continuous flow of move instructions are fed to the motor in order to make a smooth movement. The motor attached, controls the position of the pallets according to the instructions given by the microcontroller. When a key for specific space is entered in the keypad, the system checks the database for allotted space number. This information is passed on to the microcontroller to do the required action.

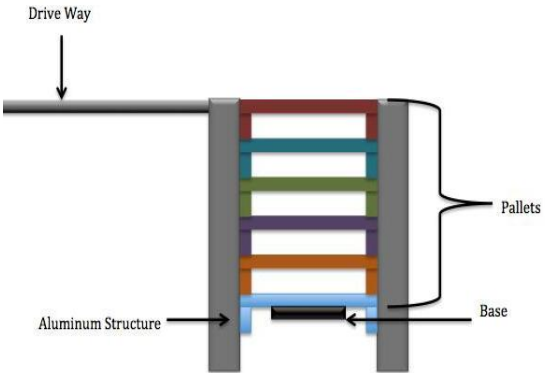

Figure 3: Pallets in the Push Up stack

3.4 Forklift Subsystem

The left and right movement of the forklift is achieved by using two-stepper motors. A stepper motor moves a rotational step per instruction. And in order to make a smooth continuous movement, a continuous flow of instructions are fed to the motor. When the flows of instructions stop, the stepper motor automatically stops. The all over movement of the forks was accomplished by utilizing gears with string connected, reaching out in vertical bearing. Moving the forklift here and there could be accomplished by, either utilizing engine and gears or direct actuators. In our task, we picked the apparatuses as it gives smoother consistent movement. If the auto was to be stopped in the second floor, guidance will be sent to the riggings to move the forks of the forklift to the upper level of the spot. Figure 4 shows the vertical gear and forklift system. The programming commands written in MATLAB were used in the mechanism to stop the forklift movements. The forklift's movements were stopped using the programming commands as it will be connected to the NXT microcontroller. In the second phase design, there are two parking spots available in two floors each. The stopping of the forklift was done by checking the "TachoCounter" in the program. The "TachoCounter" checks the "TachoLimit" after each step to make sure that the number of rotations per step of the stepper motor did not reach the Tacho Limit. The gears along with the string are used for the up and down movement of the forklift for placing the cars in the parking space. The two stepper motors are used for the right and left movement of forklift.

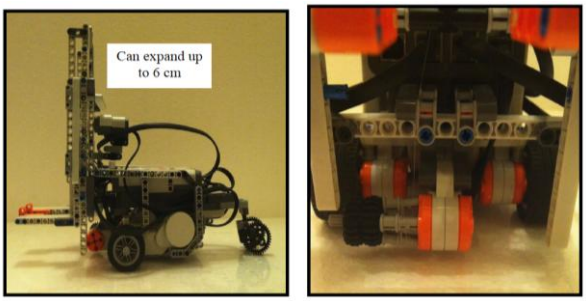

Figure 4: Vertical gear with string

3.5 Microcontroller

The microcontroller is the heart of the system. All the electrical system components were connected to the MCU. All the actions are executed on the basis of received signals or inputs. In this project, two microcontrollers were used to control different blocks. The stack system discussed in section 3.1 was controlled by the NXT 2.0 microcontroller from NXT robotics. The forklift subsystem was controlled by another NXT 2.0 microcontroller. Pair of Atmel microcontrollers controls the LEGO Mind Storm's NXT Brick. Figure 5 shows PC medium connections with microcontroller.

March-April 2020
ISSN: 0193-4120 Page No. 1385 - 1390

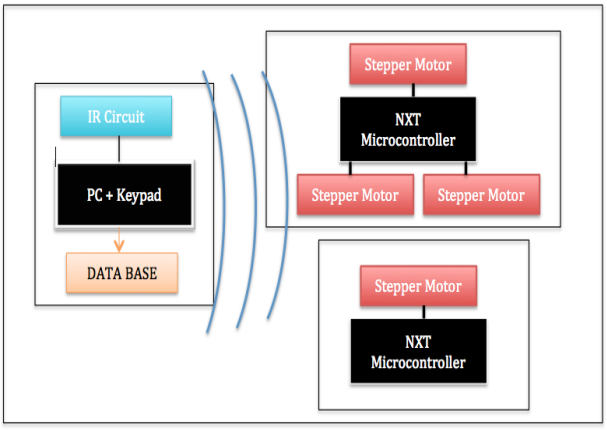

Figure 5: PC medium of connection with the MCU's

4. Verification and Validation

4.1 Component Test

The IR sensor was tested as a separate circuit before interfacing with the whole system. According to the Figure 5, the transmitter is sending signals to the receiver at all times without interruption and therefore, the current is present in the Base of the npn-transistor giving a "low" output to the PC of about 0.4V. As soon as an object was placed between them, the connection between the TX and RX was cut and the current connection to the Base was cut. This gave a "high" output of about 5.2 V to the PC.

The NXT is a shrewd PC controlled LEGO block and the cerebrum of the MINDSTORMS robot. It has four information ports for sensors, and three yield ports for engines. Each sensor and engine has its own testing and arrangement. To interface an engine to the NXT, we utilized one of the dark 6-wire links. One end of the link was connected to the engine and the other to the NXT's yield ports (A, B, or C).

In this undertaking, two distinct Computers were utilized. One with windows 64 bit to control the two NXT microcontrollers and another with windows 32 bit to get the readings from the infrared circuit. This has been done on the grounds that the two NXT microcontrollers can't be controlled on windows 32 bit working framework. In addition, windows 64 bit doesn't enable the client to peruse from the parallel port. Henceforth, a system was made between the two PCs and a mutual document (that has a connection in the MATLAB program of the NXT) was made where the parallel port readings were spared and refreshed.

Moreover, keeping in mind the end goal to control the NXT, a Bluetooth association was made. In the wake of ensuring that the NXT was turned on, it was watched that the Bluetooth is set to ON and the NXT is set to be noticeable. Moreover, it was ensured that Bluetooth was introduced and empowered on the PC.

Each engine has a worked in Rotational Sensor that controls the development unequivocally. The Rotation Sensor estimates engine pivots in degrees or full revolutions. It was tried by associating an engine to the NXT. The MATLAB code was composed by the calculation in the PC to which the NXT MCU was associated. The signs were continually being sent to the forklift through Bluetooth.

Also, some positioning errors were encountered while testing the motors. Loss of friction between the cardboard surface and the rubber tyres of the forklift was held accountable for these errors. Mainly because the comparison of the peak force on the rubber wheel and the frictional force between the wheel and the cardboard surface was made and the values calculated were very close to each other.

The peak force on the wheel was found by multiplying the torque constant (0.31739 Nm/A) and the peak current (0.69 A) fed through the stepper motor which was around 0.218 N per meter; and the frictional force was calculated to be 0.41 N (friction coefficient for cardboard and rubber = 0.5 multiplied with the weight of the forklift which is 0.81 kg).

To reduce this kind of errors, the steps ('tachoLimit') of the stepper motor were analyzed and revised through several trials and the number of steps for each path the forklift has to move was set.

This had to be done because of power limitations in NXT driver, and thermistor trip current in NXT motor. Another way to reduce the error could be by increasing friction between the wheel and the surface using a weak adhesive.

4.2 System Test

The system was tested to park and retrieve a car. When a car is to be parked, the space for the user to enter his key and the commands to park or retrieve the car, were programmed in MATLAB using GUI. The push buttons were generated by the MATLAB and the codes for the specific function had to be written for each one. The code is also used to generate the database for keeping the record of the parked cars. It also generates the pushbuttons at appropriate time and changes the color for the parking spaces to show if it is occupied or not for example, this code generates a red color to show that the space is taken and displays a green color to show that the space is open.

After the administrator began the framework, the driver was incited to enter a key for his assigned parking space and he was given the decision either to stop or recover his auto. Each key had been checked for exactness and had been doled out an assigned parking space.

In the event that there was an auto on the carport that has been distinguished by the IR circuit, which is associated with the PC by means of the pins of a parallel port; the driver was given the decision to stop his auto. Be that as it may, if the IR circuit doesn't recognize an auto,

there were three unique situations. In the event that the entered secret key was kept an eye on the database and the auto was at that point stopped, the driver was given the decision to recover his auto. In the event that the database was checked and the auto wasn't stopped or he entered a wrong secret word, he got a blunder.

At the point when a client enters his secret phrase on the keypad and parks; the microcontroller gets a request from the PC that an auto needs to come into the stopping carport. The auto was then set on the bed accessible. The bed is dependably in the leveled position i.e. when the forklift takes the auto on the bed; the stack pushes up another bed. Along these lines, each time a bed was taken to a parking space, the stack was pushed up to realign the highest point of the bed, with the level of the carport. From that point forward, the program sent the signs to the NXT microcontroller to move the forklift to the stack; this was proficient by sending persistent development guidelines to make smooth development.

At the point when a driver comes back to get his auto, he is required to pick a choice of "recovering" and after that enter a legitimate key. In the test, the framework checked in the database that rundowns all the stopped autos. The recorded auto was distinguished, the PC sent the request to the microcontroller to recover the auto; the second NXT microcontroller additionally sent a flag to the stack to push down the vertical heap of beds, to account for the approaching bed which had the auto on it. At that point, the forklift moved to the assigned parking space and conveyed the auto to the carport. At that point, it returned to the midpoint. In the event that the auto was not stopped, a blunder message was shown.

If there should be an occurrence of stopping, the Stack program pushed up the pile of beds after the forklifts gets the principal bed with the auto on it, to realign the beds with the drive way. While if there should be an occurrence of recovering, the Stack program pushes down the heap of beds to account for the approaching bed.

5. Conclusion

The primary point of robotized auto stopping administration was to modify more autos in a similar space, along these lines diminishing the space expected to stop a similar number of autos. It is a strategy for consequently stopping and recovering cars. The point was to plan and execute a model of staggered computerized auto stopping administration. This administration is completely computerized where the driver enters a numerical key and a determination, and afterward the framework grabs the vehicle and stores it in an assigned parking spot. Upon recovery, the driver enters the key and the choice, and the framework gets the auto and conveys it to the exit. The undertaking was fulfilled. While actualizing this task, some specialized issues were confronted with respect to exactness, accuracy and memory space. These issues were later dealt with by changing the product dialect of the capacities written in the microcontrollers.

References

[1] Idris, M. Y. I., Leng, Y. Y., Tamil, E. M., Noor, N. M., & Razak, Z. (2009). Car park system: a review of smart parking system and its technology. *Information Technology Journal*, *8*(2), 101-113.

[2] Bonde, D. J., Shende, R. S., Kedari, A. S., Gaikwad, K. S., & Bhokre, A. U. (2014, January). Automated car parking system commanded by Android application. In *Computer Communication and Informatics (ICCCI), 2014 International Conference on* (pp. 1-4). IEEE.

[3] Rashid, M. M., Musa, A., Rahman, M. A., Farahana, N., & Farhana, A. (2012). Automatic parking management system and parking fee collection based on number plate recognition. *International Journal of Machine Learning and Computing*, *2*(2), 94.

[4] Huang, S. J., & Lin, G. Y. (2010). Parallel auto-parking of a model vehicle using a self-organizing fuzzy controller. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, *224*(8), 997-1012.

[5] Benhassine, S., Harizi, R., & Mraihi, R. (2014, May). Intelligent parking management system by multi-agent approach: The case of urban area of tunis. In *Advanced Logistics and Transport (ICALT), 2014 International Conference on* (pp. 65-71). IEEE.

[6] Polycarpou, E., Lambrinos, L., & Protopapadakis, E. (2013, June). Smart parking solutions for urban areas. In *2013 IEEE 14th International Symposium on* (pp. 1-6). IEEE.

[7] Kokolaki, E., Karaliopoulos, M., & Stavrakakis, I. (2012). Opportunistically assisted parking service discovery: Now it helps, now it does not. *Pervasive and Mobile Computing*, *8*(2), 210-227.

[8] Kokolaki, E., Karaliopoulos, M., & Stavrakakis, I. (2011, January). Value of information exposed: wireless networking solutions to the parking search problem. In *Wireless On-Demand Network Systems and Services (WONS), 2011 Eighth International Conference on* (pp. 187-194). IEEE.

[9] Chen, M., & Chang, T. (2011, June). A parking guidance and information system based on wireless sensor network. In *Information and Automation (ICIA), 2011 IEEE International Conference on* (pp. 601-605). IEEE.

[10] GUAN, H. Z., LIU, L. H., & LIAO, M. J. (2003). Approach for Planning of Parking Guidance and Information System [J]. *Journal*

of Highway and Transportation Research and Development, *1*(8), 34.

[11] Mei, Z., Tian, Y., & Li, D. (2012). Analysis of parking reliability guidance of urban parking variable message sign system. *Mathematical Problems in Engineering*, *2012*.

[12] Revathi, G., & Dhulipala, V. S. (2012, February). Smart parking systems and sensors: A survey. In *Computing, Communication and Applications (ICCCA), 2012 International Conference on* (pp. 1-5). IEEE.

[13] Serpen, G., & Dou, C. (2015). Automated robotic parking systems: real-time, concurrent and multi-robot path planning in dynamic environments. *Applied Intelligence*, *42*(2), 231-251.

[14] Mathijssen, A., & Pretorius, A. J. (2006, August). Verified design of an automated parking garage. In *International Workshop on Formal Methods for Industrial Critical Systems*(pp. 165-180). Springer, Berlin, Heidelberg.

[15] Hussain, A., Razak, H. A., & Mkpojiogu, E. O. C. (2017). The perceived usability of automated testing tools for mobile applications. *Journal of Engineering Science and Technology*, *12*(Special Is), 89–97