Parametric Analysis of Autopilot Intelligent Robot (PIR) Using Multiple Surface Gradient Path MSGP

Olanrewaju E. Abikoye[1], Abdullateef O. Alabi[2*], O. Olaboye Yinusa[1]

[1]*Department of Mechanical Engineering, Federal Polytechnic Offa, P.M.B.420, Offa, Kwara State, Nigeria.*
[2]*Department of Computer Engineering Technology, Federal Polytechnic Offa, P.M.B.420, Offa, Kwara State, Nigeria.*

Authors' contributions

This work was carried out in collaboration among all authors. Author OEA designed the study, performed the statistical analysis, wrote the protocol and wrote the first draft of the manuscript. Author AOA managed the analyses of the study. Author OOY managed the literature searches. All authors read and approved the final manuscript.

Case Study

ABSTRACT

Robotic application is taking new dimensions around the globe, of which numerous problems are solved with embedded systems, this research introduces gradient vertices method from 3D geometric to perform data capturing using kinematic effect with aid of autopilot Intelligent Robotic (PIR). The research considered Multiple Surface Gradient Path MSGP using Toyota Camry 200x chases model using DC motor Pulse Wide Modulation (PMW). The discretion only Multiple Surface Gradients, distance values and angular pivots with respect to time. The PIR hardware "Raspberry Pi 3B" as the target board is interface with modular peripherals, using python programming language. Auto pilot is archived using different surface gradients and the digital images obtained during experience are stored for further analysis. The use of Tkinter GUI improved user experience in the extermination of the periodic oscillation, gradient values, proximate distance obtained by the PIR Final implementation. The deployment is completed by improvising a prototype model (PIR) suitable for Toyota Camry 200x. It is important to view it in the context of a

___

Corresponding author: Email: kitfad4u@gmail.com;

larger community policing framework. PIR can be classified as intermission robot that can be used for different activities with the available feature kinematic system which make it relevant for multi-purpose activities.

1. INTRODUCTION

The propagation of smooth Pilot Intelligent Robot (PIR) motion is important in the robotics world, such as to achieve accurate targets in a defined environment [1] and comfortable PIR car motion wheel with reflected action of the kinematic constraints. The wide usage of Raspberry Pi 3B single board is highly significant in the area of robotics, hardware interfacing, Robotic Computer Vision (RCV) and dynamical experimental work from researchers in the universe.

In recent years, a researcher has been indexing several asymptotically optional sampling-based motion planners which provide enough time frames, yielding a high quality path with respect to minimum-cost. According to Triebel R, et al. [2], a factor standard technique to generate smooth paths and attain specific target in less time that asymptotically optimal motion planners is a combination of a sampling-based or discrete motion for instance, which host a post-smoothing algorithm to removes the most significant path returned by the motion planner.

The use of car diagnostic devices in industries today is used to solve different problems. Taking instance of the auto mechanics environment, fully Integrating RCV on target devices as part of automotive technology for carrying out maintenance routine, in a constraining environment to human access is a priority in the research.

The use of a sample model to meet defined conditions in a motion planer is adopted to realize some experimental dataset and this is set as primary data sets for the deployment of a prototype model. At the end, enable easy ways on running full diagnosis of pre-defined target components. A Toyota car series is taken as a case study being a common used car model in Nigeria. The Chases of Toyota Camry is chosen as case study. In the literature review of the research, emphasis on Toyota chases layout using 2D and 3D graphics array and determinant of its coordinates. The use of gradient values,

topological factor and PIR kinematic modes is compared in section three of the research while the research analysis's follow suit on section four in for the research findings and implementation plans.

2. DETERMINATION OF X,Y,Z COORDINATE ON A 2D AND 3D GRAPHICS GRID VIEW

Dealing with 2D two dimensional objects and three dimensional objects 3D in an object geometric, The strategic for concatenations of smoothing operations with an algorithms of short-cutting according to S. Karaman and E. Frazzoli [3], removal of redundant vertices or solving of the existing problem with respect to optimization and with the feasible path define [4].

The evaluation of MSGP Multiple Surface Gradient Path MSGP provides a framework on the choice of PIR environment, starting from a field lines set, which determines a moving path on a grading scales, showing the positions of a moving PIR in the plan. The dimension measures its translation in 3D vector using as achieved in 3D Desk application 3D MAYA and COREL 11 Graphics respectively, the difference on motion planners as presented by [4], for a dynamic objects take a full form configuration space. Considering a robot with initial position (zero-sized) translating in a 2D or 3D in the workspace therefore a configuration can be represented using either two parameters (x, y) or three parameters (x,y,z).

According to Fig. 1 A and B in below, the 2D planner for MSGP is a framework for the research study. Taking length and breadth vertices line into account for PIR motion for a car model chases, a work space approximate length 187cm by 70 cm. An offset value of 9.55 mm for PIR routes on MSGP work space is taken as the relative position of PIR. Six concentric of radius 4.26 indicates the positions for front wheel and back wheel for Toyota chases wheel axles.

a. Vertices gradient   b. Vertices Gradient with Circular Point
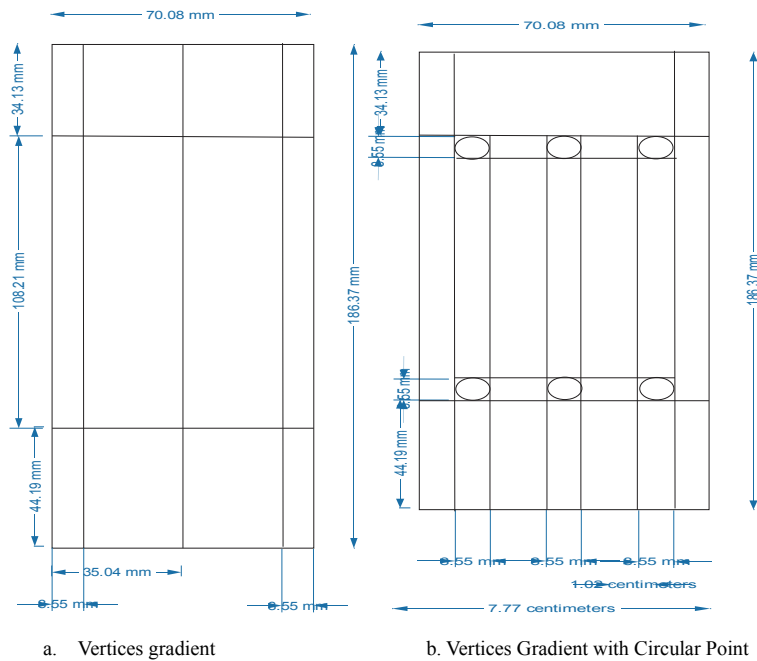
Fig. 1. 2D template design taking real time measurement of Toyota Camry 2004 model, value taken is used for vertices grid to represent major division, the circular point represent circular turning point for the PIR DC motor modeling presented on the research

2.1 Hardware Architecture Overview

Embedded system hardware is affordable components for experimental purposes, invention of new ideas and emerging solutions to industries today. This are becoming more popular today, selling outside target market according to [5,6], *"Raspberry Pi is a series of small single-board computers developed* in the United Kingdom by the Raspberry Pi Foundation in association with Broadcom" The overall architecture overview of Raspberry Pi for Pilot Intelligent Robot (PIR) breaks into six structural phases and each phase contain modular components which formed the core frame work of this research. The eco-system that is made of PIR includes the hardware architecture, software architecture, PIR secondary development environment, system firmware, Computer Visions illumination, and multipath gradient surface training.

Raspberry Pi 3B is a single board just released Raspberry Pi, the features include; core quad core 1.2 BROADCOM 64 bit CPU, 1GB memory, wireless LAN, Bluetooth connectivity on board. It has 40 GPIO pins for physical interfacing between Raspberry Pi and the outside world, it contains four USB ports for connecting USB devices, and a female stereo port for carrying audio and video signals. CS is a camera port used for connecting Raspberry Pi cameras, DTx is a display port used for connecting Raspberry Pi torch screens. The HDMI port onboard can be used to connect TV or Computer monitor, JR45 port for connecting to the internet, micro USB port for powering Raspberry Pi and finally micro SD Card port which is used for connecting Micro SD Card for loading operating system.

Pilot Intelligent Robot (PIR) is a subset Raspberry Pi robot technology that uses Raspberry Pi 3B as the core controller which has an independent development PWR.A53 power board for power distribution coupled with the motor servo driver board main for provide interface unit for PIR motion control according [7] findings.

2.2 The Ports and on Chips Identifications for PIR Target

Raspberry Pi is an experimental board that can be plugged into a computer monitor or television (TV) which uses standard keyboard and mouse (see Fig. 2 & 3). The General Purpose Input Output known as GPIO is universal since it is not defined for specific use. Its scalable GPIO allows

the user to define pin-out for computing by programming languages like scratches in Python. It's a normal computer unlike ideal computer systems which allow the user to do anything such as browsing the internet, playing high-definition video, making spreadsheets, word processing, playing games and many more. The Raspberry Pi has the ability to interact with the outside world and really application in the complement of a wide range of projects.
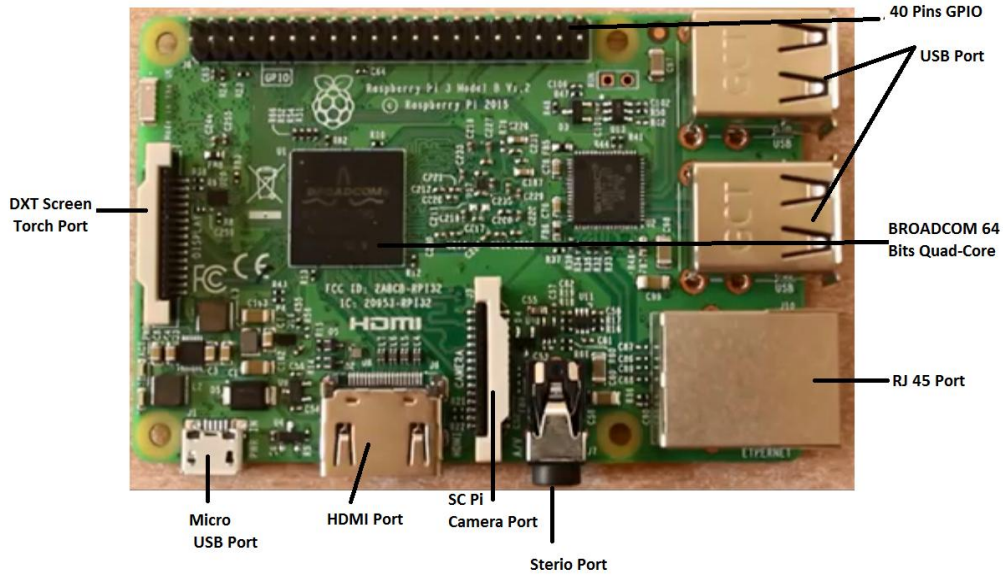

Fig. 2. Raspberry Pi single board front view features

Fig. 3. Raspberry Pi single board back view features

2.3 Major Interactive Feature of PIR on Raspberry Pi Features

1. Automatically onboard WiFi signal hot point, with control software connection.
2. PIR Capture USB camera video, and push it to the control software through the HTTP channel, the control software to decode video.
3. PIR Output logic level changes, acting on the driver module, used to control the PIR's movement.

2.4 The Function of PWR.A53 Power and Motor Servo Driver Board

Power supply voltage regulator will reduce the voltage of the battery to the voltage required by Raspberry Pi, and provide power for external devices, such as the motor servo, onboard chips, HS sensor, and other compatible interface board from GPIO (see Fig. 4). The Response to Raspberry Pi GIPO level changes translate on the motor rotation both clockwise and anticlockwise with respect to pulse wide modulation (PWM). There are many IO pin PWR.A53 boards unused which can be used to connect directly to Raspberry Pi for other users' experiments and constructions thus IO pin is arranged linearly and labelled in matrix order. The formation of linear lines "red, yellow and black" denote a standard communication protocol i.e Red for positive power, Yellow for GPIO

signal and black for negative power (See Fig. 4). The Raspberry Pi gives capabilities for researchers and learners to explore computing, and also learn how to program in languages like Scratch and Python [8].

2.5 The PIR Software Requirement and Operating System Selection

Raspbian OS Installation, the beginner always starts with New Out of the Box Software NOBS. As an easy operating system installer that contains Raspbian which provides the selection of alternative operating systems which each of them is downloaded from the internet and installed. The basic steps for pre-installation and post installations required a limited 16GB micro-SD card according to [7] described user guide.

2.6 Controlling a DC Motor with Raspberry Pi

There are different ways to control DC Motor, most common methods like transistor control with either PNP or NPN type, Diode control, and pulse wide Modulation to change the brightness is LED, duty cycle of the DC motor [9]. In case study of controlling DC motor on Raspberry Pi using transistor mode and this simple draw current from GPIO and the same setup is the same with LED control using transistor as shown in Fig. 5.
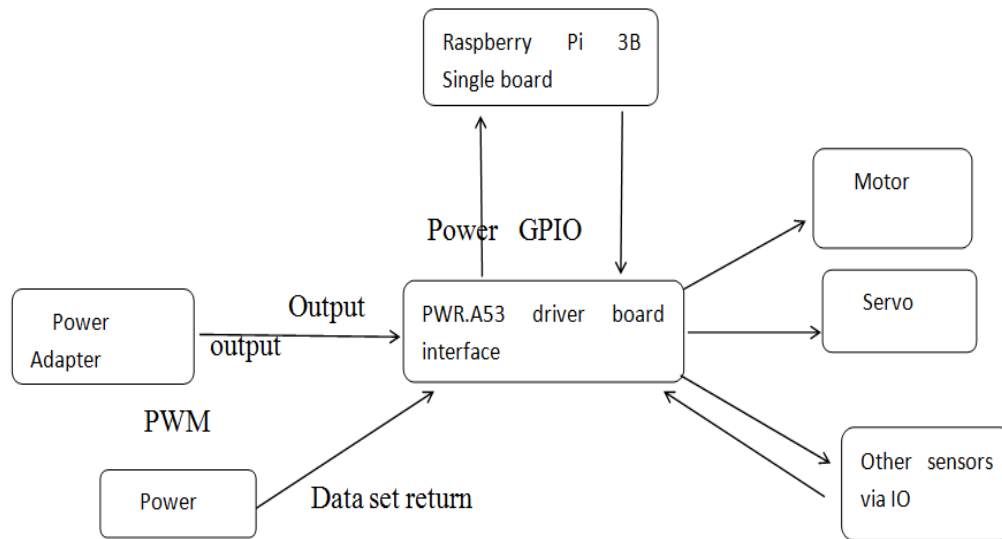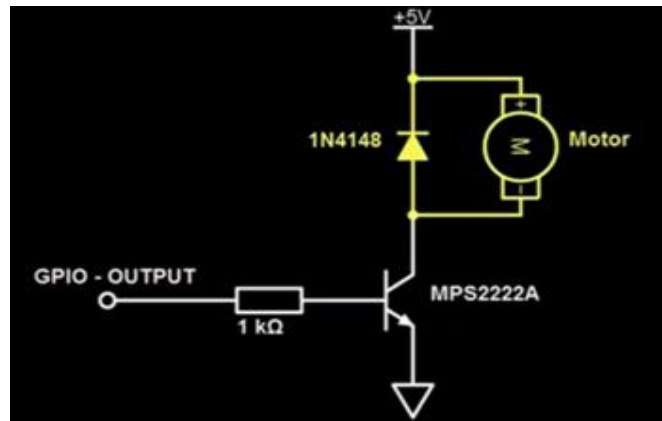

Fig. 4. Power low in Pi 3 target board: Review of PIR communication using Raspberry Pi hardware architecture model

Fig. 5. MPS222A transistor schematic circuit of DC Motor Power, GPIO and GND terminal

The load is placed on the connector, the resistor 1k to limit the current passing into the base and the emitter connected to the ground (see Fig. 6). The main different comparing with Motor is that the DC motor produces magnetic field and the inclusion of the diode a across will prevent voltage sparks in case of when the DC motor is switch off as the magnetic field collapses [10]. In this scenario, DC 6V motors is used and with the 5Volts with respect to the ground connect. The connection required three nudes to the Raspberry Pi header, 5volts, GND and GPIO Output as shown is Fig. 6.

Thus in between speed can done by turn the GPIO very fast "NO and OFF" the motor will speed approximately half fast and done by setting the pulse wide modulation pin. This implied the chance in the duty cycle will change the speed of the motor as shown in Fig. 7.

The code written in Fig. 8 enables the DC motor to rotate clockwise direction with PWM 100 (high) and slow down to 0.00 continuously with sleep time 0.02 prompt. Now for the DC motor to change in the direction of rotation then L293D motor controller can be adopted. Study pin-Out of L293D as shown in Fig. 1, for Vss and Vs. The Vss passes the chip logic by the supply of 5volts from Raspberry Pi, the Vs is the voltage supply to for the DC motor and this can be satisfied to 36volts optimum. There are four (4) ground "GND" which are internally connected some time and can be connected to a heat sink with one more pin attached for additional loan. Each separate GND pin can control a single DC motor and for setting up a DC motor requires one side of the chip by enabling EN 1 pin and connecting it to 5Volt. The IN1 and IN2 will be connected to GPIO-OUT pin and set as Output, the out of the

L983D pins (OUT1 and OUT2) is connected to the DC Motor. The two input pins it's a two ways of controlling the direction of the motor so when both are set to LOW (0-0) then the Motor fails to move but when one is set HIGH and the other set LOW then the motor rotates either clockwise or anti-clockwise. The four control of the motor is achieved with L293D as shown in Fig. 9 and 10 respectively.

It is required to connect Vs and Ground to power supply with reference to GND of the Raspberry Pi for DC motor to move clockwise and anti-clockwise, PWM pin is set on pin 7 and 11 for motor to spin code in while loop condition and provided that PWM pin changes state of GPIO from in relative to speed rate. The first for loop indicates when the dc motor rotates in clockwise direction while the 2nd for loop makes the motor rotate in anti-clockwise direction as shown in Fig. 9.

2.7 The Design Approach for PIR on MSGP

The framework is present in the first section describing PIR motion over MSGP and the following terminology is derived for full kinematic. Though few basic steps to create a MSGP for design work space, this comprises of Pilot Intelligent Robotic (PIR) working space configuration,using district coordinate point gradient in 3D geometry (x,y ,z axis) [11,12].

The Autodesk 'Maya' application work space consists of three view ports "top, side and front respectively". This user experience is very friendly in the design of PIR work space. The design is visualized in the perspective view port. The coordinate line X is set to $x_{0}, x_{1}, x_{2}, x_{3}, x_{4}$

with Y coordinate y_0,y_1,y_2,y_3,y_4 and Z z_0,z_1,z_2,z_3,z_4 to form three dimensional arrays. The distance point is measured in centimeters as illustrated in Fig. 1.

2.8 PIR GUI Graphic User Interface Design

Python is a high level programming language is most suitable for embedded system and there are many GUI applications for apart from Tkinter. The Tkinter is used to create from end for application and its compliment python language.

As the standard Python for OOP Object Oriented Programming language, which support both windows and UNIX system [13]. The associate modules Tkinter toolkit is used to create de-facto standard GUI package thus Tkinter OOP available for both python 2 and Python 3 versions [14]. The OOP for PIR design view is splits into categories Pilot Drive, Kinematic cruse modes, Computer Vision commands and Servo positions. The features parental widget toolkits to create application orientations to target most popular device like phone and laptops as shown in Fig. 12.
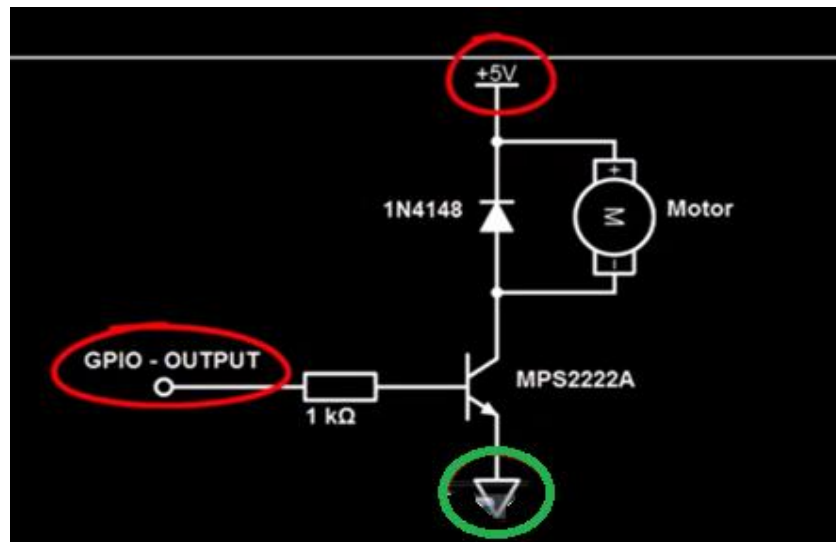
Fig. 6. Transistor logic connect of DC motor for raspberry Pi interface
If to turn the motor ON, the GPIO will be set HIGH and And for turn OFF the DC motor, GPIO will be set LOW
ON: GPIO = HIGH ……………………reference 1
OFF: GPIO=LOW………………………reference 2
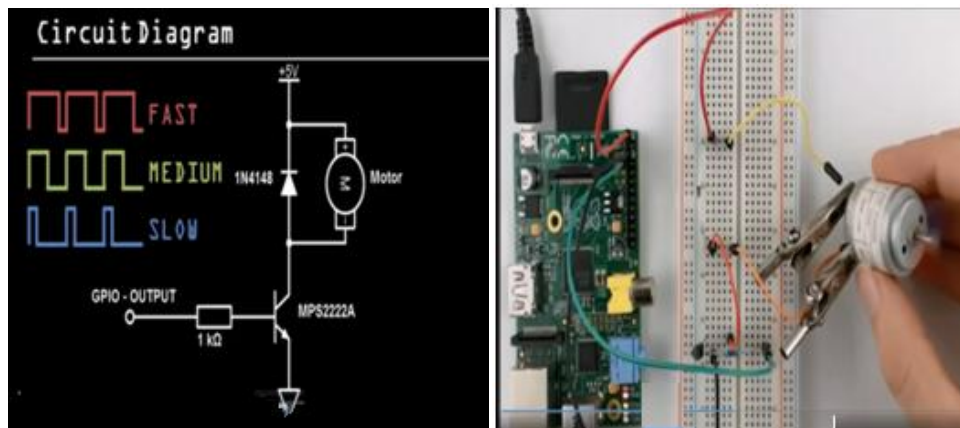
Fig. 7. Setup pulse wide modulation for DC motor in a showing DC motor duty cycle and b. the connection of DC motor to test run PWM efficiency with raspberry Pi
a. Transistor logic state for PWM of a DC Motor b. DC Motor Interface Test

13

```
import RPi.GPIO as GPIO # using RPi.GPIO module
from time import sleep # import function sleep for delay
GPIO.setmode(GPIO.BCM) # GPIO numbering
GPIO.setwarnings(False) # enable warning from GPIO
AN2 = 25 # set pwm2 pin on MDDS10 to GPIO 25
AN1 = 24 # set pwm1 pin on MDDS10 to GPIO 24
DIG2 = 23 # set dir2 pin on MDDS10 to GPIO 23
DIG1 = 18 # set dir1 pin on MDDS10 to GPIO 18
GPIO.setup(AN2, GPIO.OUT) # set pin as output
GPIO.setup(AN1, GPIO.OUT) # set pin as output
GPIO.setup(DIG2, GPIO.OUT) # set pin as output
GPIO.setup(DIG1, GPIO.OUT) # set pin as output
sleep(1) # delay for 1 seconds
p1 = GPIO.PWM(AN1, 100) # set pwm for M1
p2 = GPIO.PWM(AN2, 100) # set pwm for M2
try:
    while True:
print ("Forward"); # display &quot;Forward&quot; when program executed
GPIO.output(DIG1, GPIO.HIGH) # set DIG1 as high, dir2 = forward
GPIO.output(DIG2, GPIO.HIGH) # set DIG2 as high, dir1 = forward
p1.start(50) # set speed for M1, speed=0 - 100
p2.start(50) # set speed for M2, speed=0 - 100
sleep(1) # delay for 1 seconds
print ("Backward") # display &quot;Backward&quot; when program executed
GPIO.output(DIG1, GPIO.LOW) # set DIG1 as low, DIR = backward
GPIO.output(DIG2, GPIO.LOW) # set DIG2 as low, DIR = backward
p1.start(50) # set speed for M1, speed=0 - 100
p2.start(50) # set speed for M2, speed=0 - 100
sleep(1) # delay for 1 seconds
except: # exit program when keyboard interrupt
p1.start(0) # set speed M1=0
p2.start(0) # set speed M2=0
print ("Left") # display &quot;Left&quot; when program executed
GPIO.output(DIG1, GPIO.LOW) # set DIG1 as low, dir2 = backward
GPIO.output(DIG2, GPIO.HIGH) # set DIG2 as high, dir1 = forward
p1.start(25) # set speed for M1
p2.start(25) # set speed for M2
sleep(1) # delay for 1 seconds
print ("Right") # display &quot;Right&quot; when program executed
GPIO.output(DIG1, GPIO.HIGH) # set DIG1 as high, dir1 = forward
GPIO.output(DIG2, GPIO.LOW) # set DIG2 as low, dir2 = backward
p1.start(25) # set speed for M1
p2.start(25) # set speed for M2,
sleep(1) # delay for 1 seconds
```

Fig. 8. Source code running test on python language for DC motor control

The primary function defined for PIR in the research study in Computer Vision (CV), Profile of different surfaces know as Multi-Surface gradient (MSG), Vertices Path Cruise (VPC), Turning Point (TP) and Distance Point Drive (DPD)

2.9 Distance Points Value for PIR Position on MSGP

Three criteria are set as priority, distance translation along the surface area guided under MSGP layer, more so, the wheel of PIR taken to be 6.8cm with radius 3.4 cm. When set on motion is set PIR move a define direction as initialized in the programme segments. The data values is obtained for PIR Left-Pivot or Right-pivot at any given spot with respect to 0.45PWM on 180° smooth surface will enable the PIR DC Motor to target 90° turning point as shown in Table 1and 2 bellow.

The PIR efficiency and adaptability is refined by its performance over a resistance surfaces. The interval point as illustrated in MSGP on 3D on a point of inclination take average value of 21.8° with respect to 5.4 cm length. The resistance value is considered using gradient level taken from rough surfacing, which varies from 0.1 to 0.6 respectively.
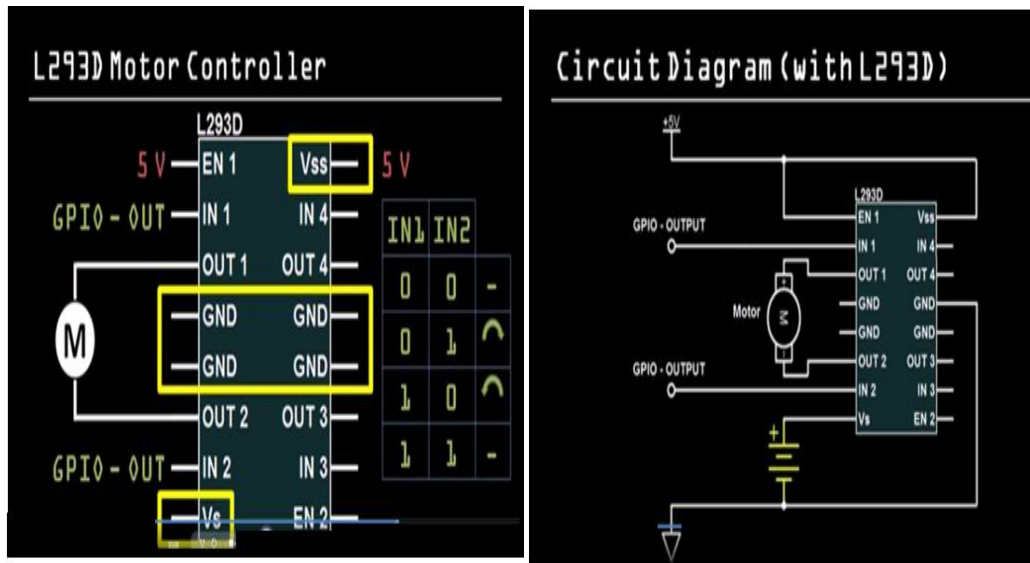
14

Fig. 9. To left is the Pin-Out Connections of L293D controller and to right is Overall socket diagram of L293D controller

Table 1. PIR DC motor data set taken for SMGP against PIR tyre turning point

SMGP	5.00	5.55	5.60	6.50
Turning Angle 90° in PWM	045	0.50	0.55	0.60

Table 2. Field experimental data obtained for PIR cruising on SMGP using inclination level meter and rough surface tester

Length Interval	Inclination Angel 1(Degree)	Roughness 1 (mm)	Inclination Angel 2 (degree)	Roughness 2 (mm)
20.6	3.6	0.3/5 mm	30.0	0.1/1 mm
41.2	7.3	0.2/3 mm	18.2	0.8/13 mm
64.8	3.1	0.6/8 mm	29.4	0.7/12 mm
82.4	6.6	0.1/1 mm	24.0	0.1/1 mm
103.0	2.0	0.4/6 mm	23.8	0.5/8 mm
123.6	5.6	0.5/8 mm	28.0	0.4/6 mm
144.2	10.6	0.4/6 mm	22.1	0.8/13 mm
164.8	5.4	1.0/17 mm	20.1	0.8/13 mm
186.0	5.8	1.5/26 mm	27.3	0.6/10 mm
Average	5.4		21.8	

3. RESEARCH METHODS

3.1 The Design Frame work Preview

The PIR design is made from raspberry pi hardware, it is construct as Plug-In Intermission robotic. A tool for feedback Intelligent for automobile technology built to capture data from car chases model, experiment is carried out using pi camera, DC motor control through manipulation for different surface areas known as MSGP configuration work space. Surface texture is tested, the surface topology that post

significant effect of distance cover my PIR kinematic and yield accurate results with respect of defined values. Time stamp for every translation in x and y plan is determined using position driving method in the PIR training segment. Train to retrain enable PIR to archive distance target as design in the MSGP. The source code is written in python programming language and to improve the user experience. GUI is adopted which feature Tkinter widget orientation and this is compatible to many device operating system for phone, laptop and desktop devices. Gathered data from conducted

15

experiment make the PIR application for automotive car diagnostic application and equally serve as intelligent system to data extraction is a constraints environment.

3.2 The Development Environment of Building

Given that the 186 cm Length of a Toyota Camry 2004 with 70cm breadth for 3D MSGP determines the configured working space. The split mesh with projection for both x and y coordinate vertices measures 8.6 cm, with the sub division four in the off-Set edge Loops is used to present total orientation for PIR positions. This description is a match size of Toyota Camry 2004 chase as presented in Fig. 10.

The study to move PIR DC in this defined path in the highlights give a symbols

$A_1,A_2,A_3,B_1,B_2,B_3,C_1,C_2,C_3$ with respect x,y coordinate as shown in Fig. 10 and 11. Distance for path way is considered to PIR kinematic system. Therefore, and each symbol is a defined position in the programme segment and is use as a points spot to data capture with PIR communication mode. This is used as reference in the whole framework for MSGP design.

3.3 Defined Functions for PIR using MSGP Sub-Divisions

The PIR trainer MSGP symbols $A_1,A_2,A_3,B_1,B_2,B_3,C_1,C_2,C_3$ is duely presented in 3D geometry indicate area "purple color" is the main training path for PIR, each vertices off-set grid is right angle (90°) with reference to Fig. 11. Take to be PIR Turning Points (SP).
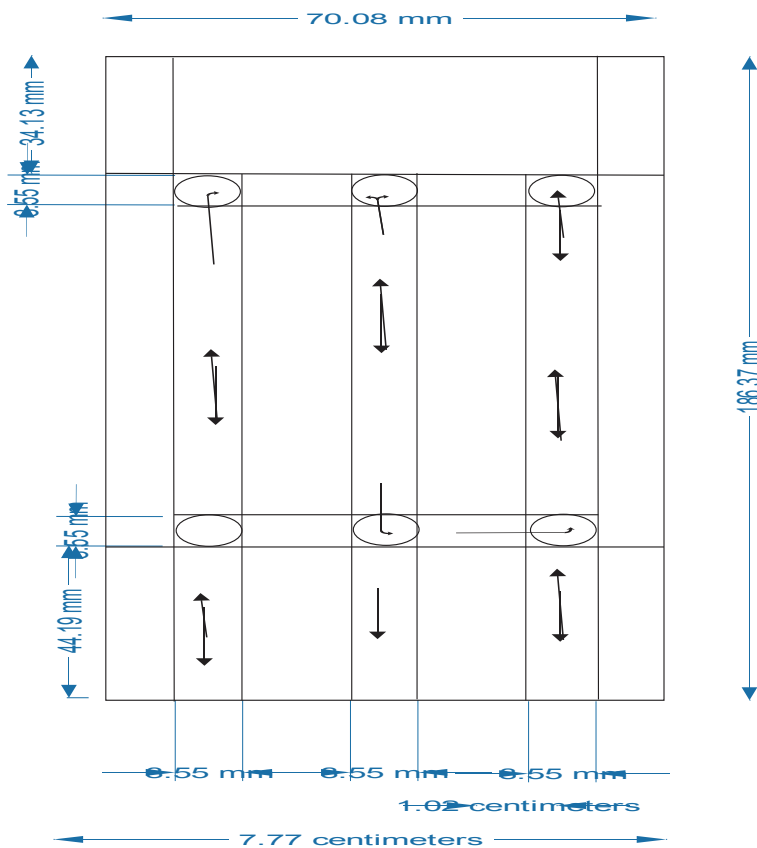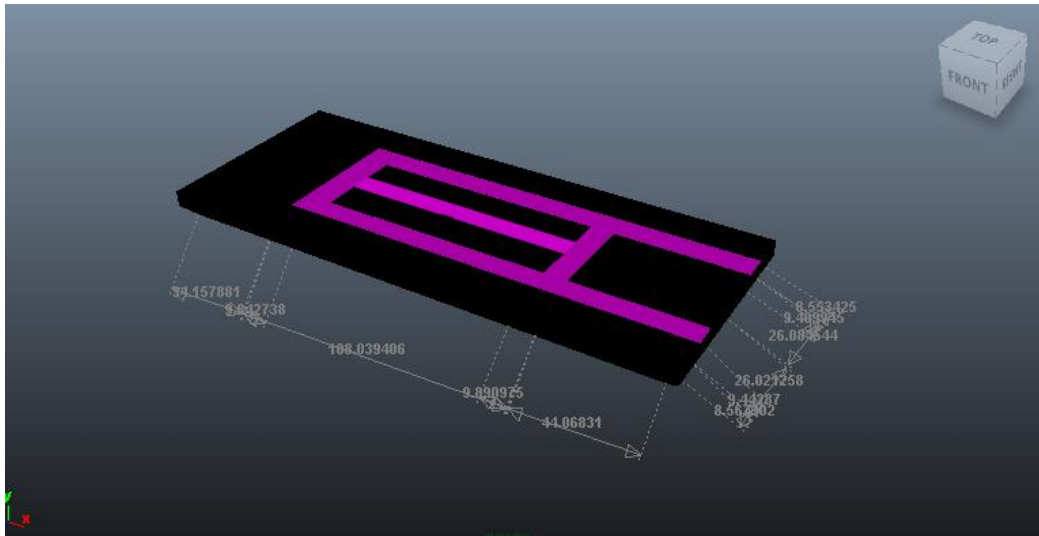

Fig. 10. The angular projection for the design of MSGP template. The multi-nudes and turning points is extracted which is represent circle 2D. The alignment of its distance values show vertices grid of Toyota Camry 2005 represents to outline measure 186 by 70 cm

Fig. 11. 3D Geometrical formation of MSGP trainers for PIR DC motor motion

The map out area in Red, Yellow and Light-Blue is a most significant area of PIR kinematic system. The Toyota Camry 200X front and back shafts position is consider as pivot point to be examined by PIR. This is represented as symbols $A_1,A_2,A_3,B_1,B_2,B_3,C_1,C_2,C_3$ to fit in to chases outline as presented Fig. 12,13 and 14 respectively. Red face represent chases bonnet position while yellow area present the boot section of the Toyota car chased model. The Light-blue face is the distance between front wheel and back wheel on the chases model.

4. RESEARCH DATA AND ANALYSIS

4.1 Analysis of PIR Features in the MSGP Design Outlines

The model of PIR shown in Fig. 15 represent true prototype of our PIR hardware assembled for experiment trainer. This model in Fig. 16 is only achieved with 3D modeling basically to work out PIR motion through distance point measures is 4.4 cm in the geometric port view.
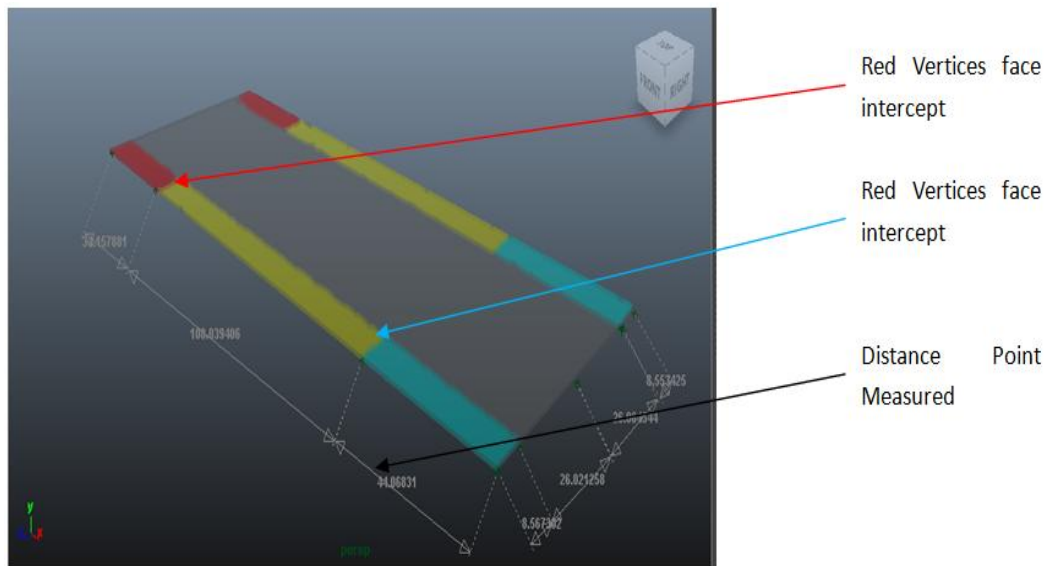
Fig. 12. Intercept faces vertices of the Toyota Camry 200X Series. The four wheels pivot points represent red and light-blue vertices on geometry
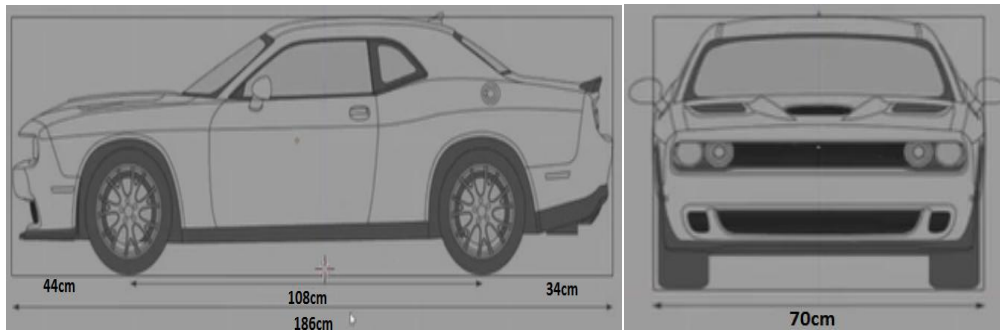
17

Fig. 13 and 14. showing the front view with gradient scale 44 cm, 108 cm and 34 cm by the side view with scale 70 cm
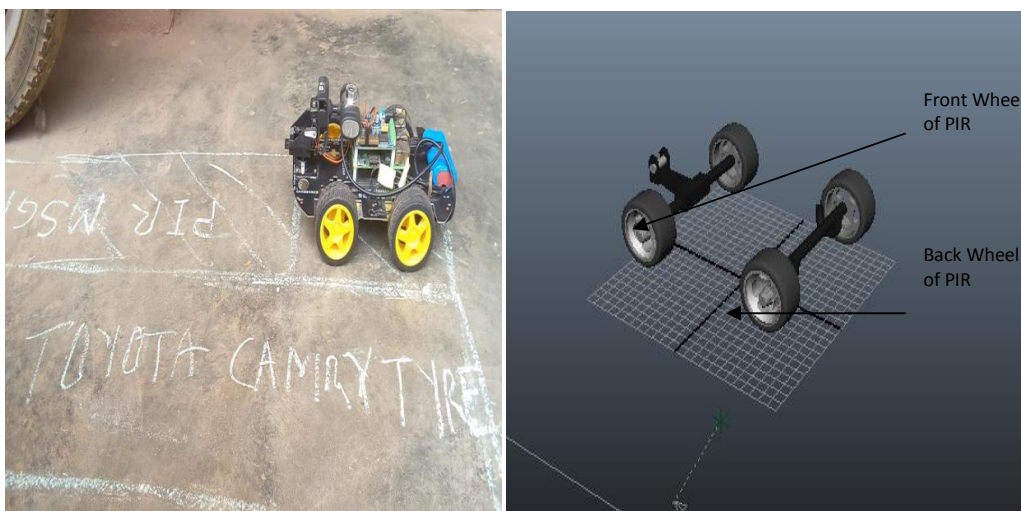

Fig. 15. A coned PIR hardware prototype model know as PIR to the right and 3D modeled prototype PIR for MSGP work configuration
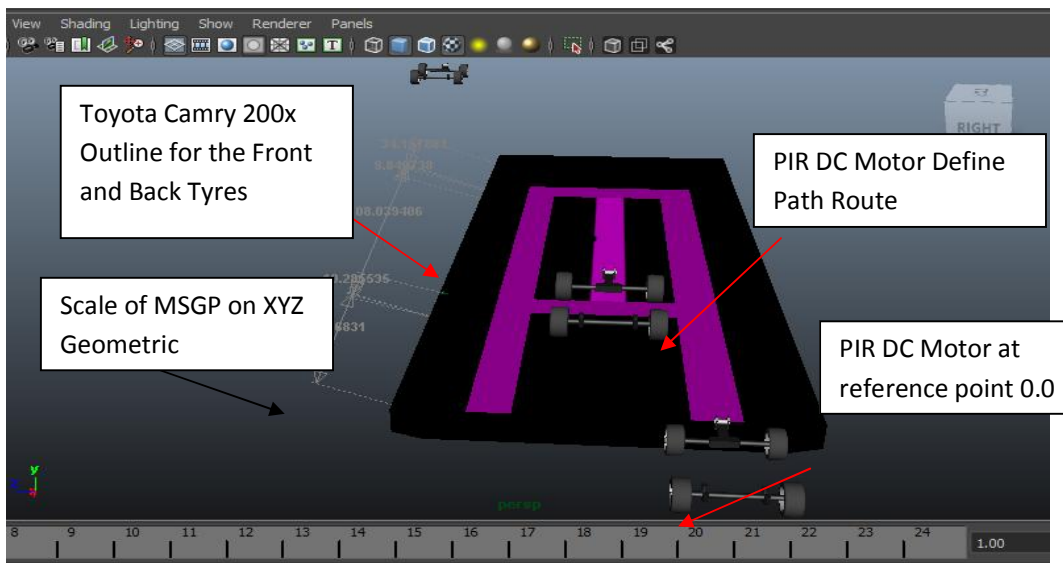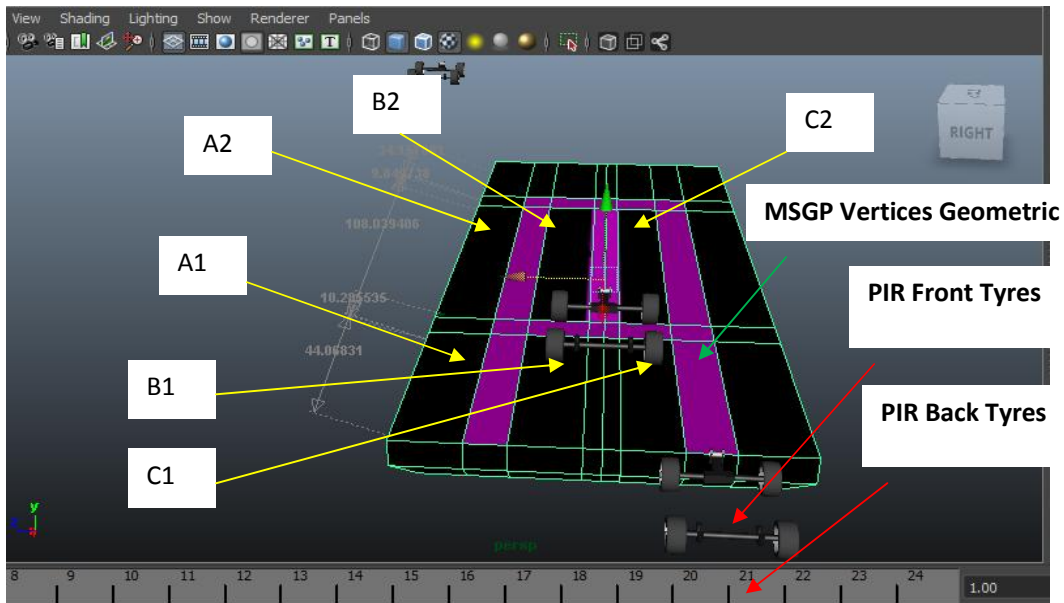

Fig. 16. The Combined feature of PIR together with MSGP work configuration

Fig. 17. Geometric scaled vertices
The presented symbols, showing South-West (A1), North-West (A2),South-Centre(B1), North-Centre(B2), South-East(C1), North-East(C2). Considering each scale for geometric distance is taken as gradient floor for a given MSGP

The PIR is taken position B1 with reference to 44 cm vertices South-Centre position on MSGP sport as illustrated in Fig. 18. The fundamental bases is that the MSGP as the working environment, the PIR is set to reference position 0.0 on South-West coordinate, using experimental performance to determine distance cover by PIR as presented in Fig. 19. The User to set Motion command "Forward Distance" FD implied forward distance movement, this enable GPIO on PIR to move from 0.0cm South-West as the DC motor Pause Wide Modulation with 100Hz frequency on a distance 44 cm. The PIR DC I set to rest for 5 second in time T. The procedural functions is configure for other defined data set as presented in Table 3 for PIR to remain active as GPIO of the DC motor enable.

The approach for the data set is translated in Table 2 for PIR to present DC motor state for Distance Drive (DD). The position for symbols with respect to Distance is a data-set for the current position defined for PIR when GPIO is enabling for PIR DC motor.

4.2 Case Study Analysis of PIR Kinematic System Approach

PIR Kinematic Case 1: In the Fig. 18 in above indicate when PIR DC motion paths segments with intents to create stop point before PIR make turning referred to Fig. 1 as the PIR set to move on straight paths and make 90° turning point five consecutive times after put to rest. This value lengths given range is set to 44 cm + 108 cm +26.5 cm + 108 cm + 26.5 cm + 108 cm as motion path on MSGP geometry to that effect.

Table 3. The defined data set for MSGP plane which is denoted by x and y vertices intercept from the off-set edge loop

Geometric	Symbol	Distance (cm)	Pivot Left/Right (PWM)
South-West	A1	44	0.45
South-Center	B1	44	0.45
South-East	C1	44	0.45
North-East	A2	108	0.45
North-Center	B2	108	0.45
North-West	C2	108	0.45
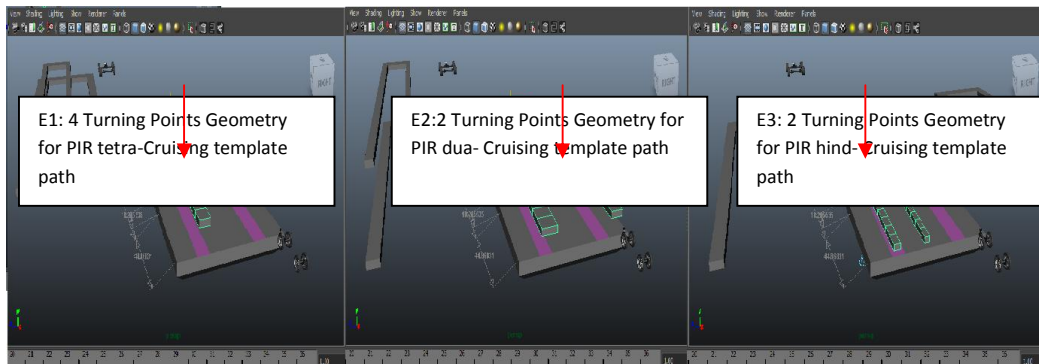
Fig. 18. The Extraction of cruising directions for PIR wheels path-way on MSGP
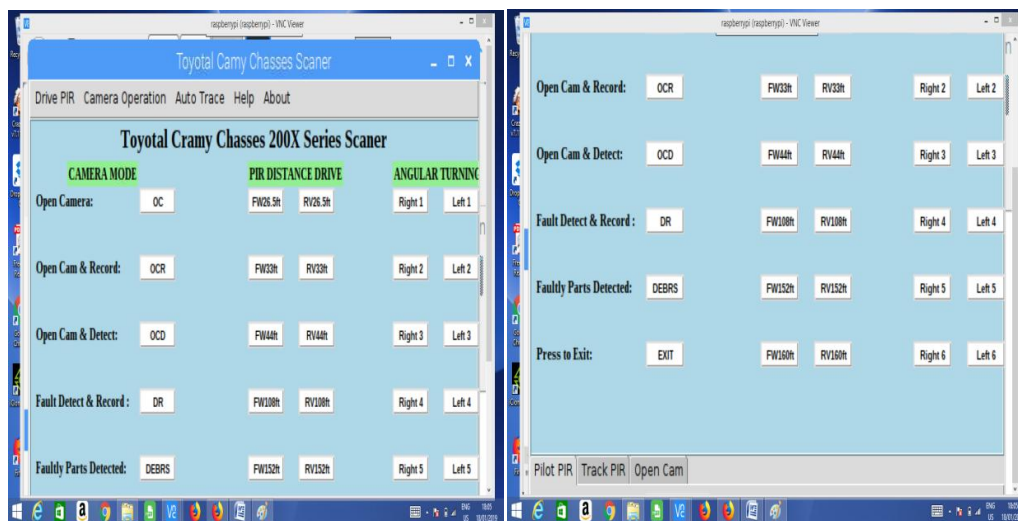
Fig. 19. To right is GUI application with caption toyota camry chasses 200X series scanner and to left is GUI application with caption toyota camry chasses 200X series scanner

PIR Kinematic Case 2: Study PIR turning points, this is more like rotating about same axis, a pivot left (right angle) is set achieve and this measure 0.45PWM from PIR DC motor, a turning of 90° is make to this effect and later set to rest. When a PIR DC status is set to rest, this set GPIO low in 5 second as defined for the experiment. Therefore, lengths given range is set to 152cm + 53 cm+152 cm is set to make angular turning points as shown in Fig. 20b express E2. The change on short-distance of 26.5 cm make different with Expression E2.

The use of DC Motor for PIR data capturing base on defined functions, has creates give chance to have different test results on Pulse Wide Modulation (PMW). Four profiled structure is taken into account to the performance of PIR Difference floor gradients tested closely for PIR idea working environment. Each profiled environment is taken as MSGP and for PIR to fully operate to fulfilled user command from a click button. The GUI orientation is denotes with label buttons A-C. Each buttons is set functions for the PIR experiments test as shown in Figure. The angular turning point in the figure to the right indicate a functions name pivot left/right to trigger a Pulse Width Modulation (PWM) varying according to surface gradients for the values of 0.45PWM - 0.70PWM. as shown in Table 4.

Table 4. DC motor pulse width modulation for the pivot left and right

Right 1	Right 2	Right 3	Right 4	Right 5	Right 6
0.45	0.50	0.55	0.60	0.65	0.70
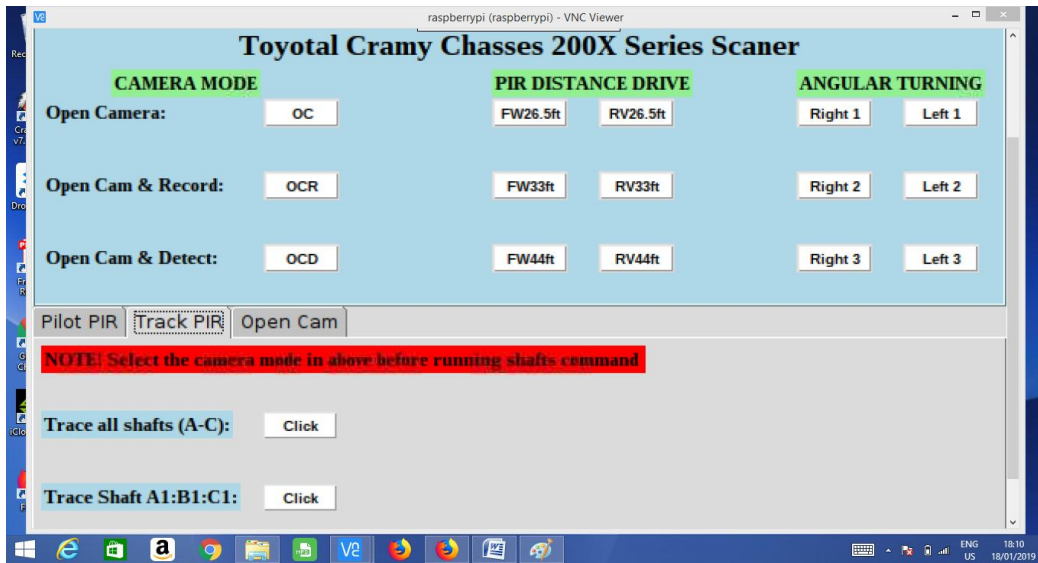
Fig. 20. The Tkinter lower frame of GUI application with caption toyota camry chasses 200X series scanner

The GUI app orientation features Toyota Camry 200x scanner, A PIR user control interface to executes user command functions with a click buttons. These are classified two. First is a check for state for all shafts and is presented on the orientation as "trance all shaft (a-c). This make PIR to maintains geometric positions (A_1, B_1, A_2, B_2) for every mouse clicks, a function executed to perform kinematic mode for PIR on MSGP configuration. In the second classification, Symbol A_1, B_1 allow PIR to move forward and backward on a straight route with constant frequencies. The third categories is angular turning commands, this allow PIR to make turning on MSGP configuration as shown in Fig. 20 and the PIR wheel match positions A_1, B_1, A_2, B_2 in this case, PIR is set to extract data when PGIO is set LOW. Therefore, for cruising PIR DC Motor motion set Forward, Reverse, Left Turn, Right-Turn using a click functions. The PIR DC motor GPIO is always set HIGH to activate functions and set LOW when function is completed executed using user command. With user button click functions set PIR to activate GPIO to execute kinematic for Left-Turn, Right-Turn, Reverse and Forward.

Motor, capture data using CV protocols and rendered MSGP as developed for the study. The external factor for working environment like surface tension, smoothness, roughness and topology of the PIR make prediction. The system behavioral of PIR is able adapt to new working environment considering its size, dimension, height and wheel diameter. The kinematic behavior of PIR is directly proportional to PMW effects. The PIR wheel radius is determinant fact for the accuracy in PIR movement in x,y coordinate, so therefore MSGP configuration environment is achieved with PIR kinematic principles.

The PIR is a reflection of a trained robot in an adaptive environment, therefore this application is widely use in area of artificial intelligent, data surfing, data capturing, and analysis. The GUI presentation improve user experience in reality in the development of automotive technology and further research is ongoing on the conceptual kinematic system to the application of Raspberry Pi technology for robotic applications and back testing of experimental procedures.

5. RECOMMENDATION AND CONCLUSION

The research on Raspberry Pi is an emerging trend in robotic, autopilot intelligent Robotic PIR used for case study is able to obey kinematic principle aid with experimental training of DC

ACKNOWLEDGEMENTS

21

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Dragan AD, Lee KC, Srinivasa SS. "Legibility and predictability of robot motion. in ACM/IEEE International Conference on Human-Robot Interaction (HRI), 2013;301–308.
2. Triebel R, Arras K, Alami R, Beyer L, Breuers S, Chatila R, Chetouani M, Cremers D, Evers V, Fiore M, et al. "Spencer: A socially aware service robot for passenger guidance and help in busy airports," in Field and Service Robotics (FSR). 2015;14:607–622.
3. Karaman S, Frazzoli E. Sampling-based algorithms for optimalmotion planning. International Journal of Robotics Research (IJRR). 2011;30(7):846–894.
4. Raspberry Pi. Installing operating system images - Raspberry Pi Documentation Date: 11-18-2018; 2018
Available:https://www.raspberrypi.org/documentation/installation/installing-images/
5. "Ten millionth Raspberry Pi, and a new kit – Raspberry Pi". We've beaten our wildest dreams by three orders of magnitude; 2016.
6. Janson L, Schmerling E, Clark A, Pavone M. Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions. International Journal of Robotics Research (IJRR). 2015;34(7):883–921.
7. Booting from NOOBS. Installing raspbian with NOOBS - Introduction | Raspberry Pi Projects; 2018.
Available:https://projects.raspberrypi.org/en/projects/noobs-install
8. Nash A, Daniel K, Koenig S, Felner A, Theta Any-angle path planning on grids. In AAAI Conference on Artificial Intelligence (AAAI). 2007;1177-1183.
9. Monk S. Make: Action: Movement, light and light sound with arduino and raspberry Pi. San Francisco, CA: Maker Media; 2016.
10. Chaitanya K, Karudaiyar G,. Deepak C, Reddy KB. 1st international Conference on data Engineering and Communication Technology, 469 Springer Verlag; 2017.
11. Hart PE, Nilsson NJ, Raphael B. A formal basis for the heuristic determination of minimum cost paths. IEEE Transactions on Systems Science and Cybernetics, 1968;4(2):100–107.
12. Hsu D, Latombe JC, Sorkin S. Placing a robot manipulator amid obstacles for optimized execution. In Proceedings of the International Symposium on Assembly and Task Planning (ISATP). 1999;280–285.
13. Geraerts R, Overmars MH. Creating high-quality paths for motion planning. International Journal of Robotics Research (IJRR). 2007;26(8):845–863.
14. Luna R, Ucan IAS, Moll M, Kavraki LE. Anytime solution optimization for sampling-based motion planning. In IEEE International Conference on Robotics and Automation (ICRA). 2013;5068–5074.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle4.com/review-history/63621