Presented at the 4th DLSU Innovation and Technology Fair 2016
De La Salle University, Manila, Philippines
November 24 & 25, 2016

SPHEMO: A Teleoperated SPHErical MObile Robot with Video-streaming Capability

Camille Bianca C. Gomez[1], Patrick Joshua M. Gonzales[1], Angel Fatima M. Opulencia[1], Abigail C. Soriano[1], Noriel Mallari[2] and Alexander Abad[2,*]
Department of Electronics and Communications Engineering
[1]De La Salle University – Science and Technology Complex, Biñan Laguna, Philippines
[2] De La Salle University, 2401 Taft Ave. Manila, Philippines
**Corresponding Author: alexander.abad@dlsu.edu.ph@dlsu.edu.ph*

Abstract: This study focuses on the design and implementation of a spherical msobile robot called SPHEMO. It uses a 10.5-inch diameter hamster-ball as its casing. A DC motor is used to actuate its forward and backward movement and a servo motor to actuate its left and right movement by moving the weight of the pendulum inside the ball. SPHEMO uses a pendulum to stabilize its left and right turning motion. SPHEMO's movement and control are then tested in different types of controlled environment which includes an imitated air-condition ventilation duct and a classroom environment. SPHEMO is teleoperated using WiFi connection. It has a video-streaming capability. Control algorithms were programmed in an Aduino microcontroller with WiFi shield for teleoperation.

Key Words: Arduino WiFi shield, spherical mobile robot, stabilizer, SPHEMO

1. INTRODUCTION

The goal of this study is to design and implement a spherical mobile surveillance robot (SPHEMO) that can be controlled remotely using WiFi connection. The spherical mobile robot can move forward, backward, turn left and turn right. The spherical mobile robot can video-stream the view of its environment using a small camera mounted inside the ball. It can be used for surveillance and monitoring. SPHEMO can be used in remote surveillance and monitoring. It is a like a rolling closed-circuit television (CCTV) wherein it can be used to roam an area to gather information via WiFi and video-streaming. It can be deployed as a risk assessment or post-calamity assessment tool. It can roll on areas that are too small or too narrow for someone to pass through or difficult for human to crawl on such as air-condition ducts or low profile clearances. SPHEMO is portable, uses rechargeable battery, and can be used in one's home, office or warehouse as a rolling CCTV for security and surveillance.

2. RELATED LITERATURE

A comprehensive literature review on the design of spherical rolling robots was done by Vincent Crossley (2006) wherein he discussed the different advantages of spherical robots such as: a) very maneuverable, b) prevents getting stuck in corners, c) cannot overturned, d) have great capability to recover from collision with obstacles, e) can be useful in swarm applications and f) can be totally sealed to be used in hazardous environments. According to Crossley, spherical robots can be: a) Wheel based such as the first spherical robot by Halme et al. (1996) that has one wheel in contact with the bottom of the sphere, a two wheeled design by Husoy (2003) and a small car resting at the bottom of the sphere designed by Bicchi (1997); b) Two Independent Hemispheres such as the designed by Bhattacharya (2000) that is composed of two hemispheres; c) Pendulum based such as the GroundBot Rotundus; d) Relocation of Center of Gravity such as the robot SpheRobot designed by R. Mukherjee et al. (2002), robot August designed by Javadi and Mojabi (2002) and Electric Hamster (2011). The shifting weight inside the spherical robot changes the direction of motion. This is concept is the basis of the implementation of SPHEMO.

3. THEORETICAL FRAMEWORK & DESIGN CONSIDERATION

The goal of the researchers is to make a mobile surveillance camera that can be controlled using a computer application. The spherical mobile surveillance camera has been tested on a classroom environment and a prototype air condition ventilation duct. The design of SPHEMO is divided into two parts; hardware and the software part. Further discussions of these two parts are provided below.

3.1 Hardware Design

The mechanical design consists of the diameter shaft with a gear connected to a geared DC motor. In order for the ball to rotate, the DC motor must be stabilized by a weight hanging on the diameter shaft of the sphere. The DC motor is the control mechanism for the forward and backward movement. With regards, to the turning to left or right direction, a pendulum is needed to shift the internal weight. The shifting of internal weight of the bottom part of the pendulum is made possible using a servo motor. The shifting weight concept for maneuvering the SPHEMO is based from a blog ("Building an Electic Hamster", 2011). This project do not only focused on maneuvering but it also includes surveillance. A small camera capable of video-streaming is mounted inside the SPHEMO. Figure 1 shows the whole mechanical structure of the SPHEMO. The size of the hamster ball used is 10.5-inch in diameter.

A DC geared motor is connected to the gear fixed on the shaft. The shaft is also fixed to the hamster ball so that as to when the motor rotates, so does the ball. The balancer is provided with washers to add more weight and to stabilize the ball.

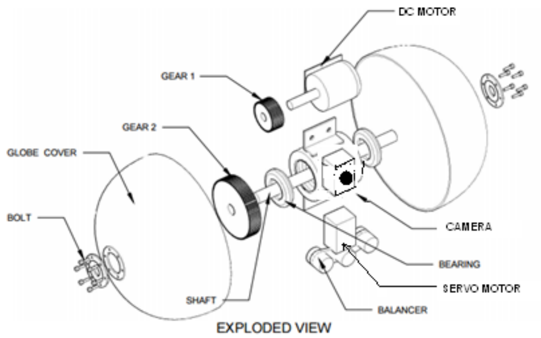

Fig. 1. Detailed Design of SPHEMO

A servo motor is connected to the balancer to make the pendulum lean to a certain direction – left or right. To close the ball, a nut is used at the end of the shaft to tighten the closure. The bearing in the middle is used so that the body will not move even if the ball does. Shown in Figure 2 is the actual mechanical structure of the SPHEMO.

A GoPro Hero3 (2014) camera is attached inside the SPHEMO. Two rechargeable batteries were used to power up SPHEMO. The first battery is Lithium Ion having 8.2V, 3000mAh specifications. This is used to power up the dc motor. The other battery is a Hi-TECH NI-MH 7.2 V, 1200mAh which is used to power up the Arduino and the servo motor. This project SPHEMO used an Arduino Mega 2560 (2016) microcontroller to control the motors, a Wi-Fi ESP 8266 (2016) module for WiFi wireless connection and MDD10A Dual Channel 10A DC Motor Driver (2016).

3.2 Software Design

There are different software designs for developed for the SPHEMO: a) Wi-Fi connection design for wireless control and video streaming, b) motor control algorithms and c) graphical user interface (GUI) for end-user controller. Two different languages were used to transmit data:

Presented at the 4th DLSU Innovation and Technology Fair 2016
De La Salle University, Manila, Philippines
November 24 & 25, 2016

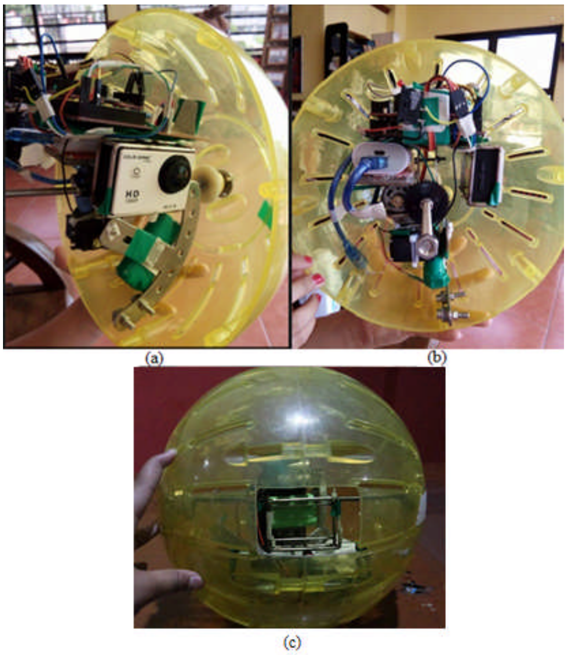

Fig. 2. (a) Front view, (b) Side view, (c) Actual Mechanical Structure of the SPHEMO

C/C++ for the Arduino and Java for Java Eclipse software development kit (SDK). These two languages were only applied and used for the control movement. The command will be sent through the Java Eclipse SDK. It will be translated into the Arduino in order for the robot to understand and perform the command. Video-streaming is done using GoPro camera via WiFi connection.

With regards to the firmware for motor control, this study utilized the Biscuit Platform (2015). It includes an operating system that implements a process scheduler that uses a cooperative multitasking, also known as non-preemptive multitasking, where the operating systems never initiates a context switch from a running process to another process. Each process yields control for a moment for other process to run. This type of scheduler saves clock cycles and memory space because it does not need to save register variables when switching from process to process. This operating system requires all process to be non-blocking. Test from this operating system suggest that processes are guaranteed to run in an accuracy of ±2mS. The operating system can suffer from lag when processes do not yield, like using delays or hold and wait, and may affect the responsiveness of all the other process.

The Biscuit Platform has its own library for using the UART function of the avr ATmega microcontrollers. This UART interface makes use of the interrupt vectors that are present in the avr Atmega memory address. Theses interrupt vectors are signal for microcontrollers stating that asynchronous processes are completed. The microcontroller sends a byte through the UART peripheral, but the operating system will not wait for all the bits to be sent by the UART peripheral, because that would mean wasting clock cycles which is inefficient. Instead, the bytes are enqueued for sending, and dequeued when the bytes preceding it are sent. When a byte is finished sending the UART peripheral will signal the microcontroller, using an interrupt, which it is finished and ready for the next byte to be sent. It's main difference between Arduino's Serial functions is that Biscuit's UART interface does not waste clock cycles waiting for an independent peripheral to finish, it makes use of a built in interrupt signal.

The Biscuit Platform contains a library for interfacing with a Wi-Fi module. This library includes functions for setting the Wi-Fi module to act as a client for a user created server. The processes and commands used for connecting to the server and the implementation of the UART interface for communicating to the Wi-Fi module is abstracted to the user. The Wi-Fi system also implements a stop-and-wait automatic repeat request error correction if there are errors that occurred in the processing of the command or in the connection itself. In case of sudden disconnection from the network or server, the Wi-Fi system will automatically reestablish its connection to the network and server, all sending request will be halted while connection is not established. The library is tested on ESP8266, a Wi-Fi module unit, and all commands in the system are based from it.

Shown in Fig. 3 is the Graphical User Interface (GUI) developed for the SPHEMO. Part of SPHEMO's feature is to record the commands of the user and imitate these actions once the user set the control to autopilot. The user would activate the "Remember" button from the GUI and the controller starts recording all the actions. The user will send command and SPHEMO records this command including the time and the speed. All these data will be stored in database. Once the user sets the control to autopilot mode, the recorded data that was stored in the database will be used as a set of commands to be performed by SPHEMO. This type of feature actually helps the user scan an area when the user is not around or is sleeping at night.

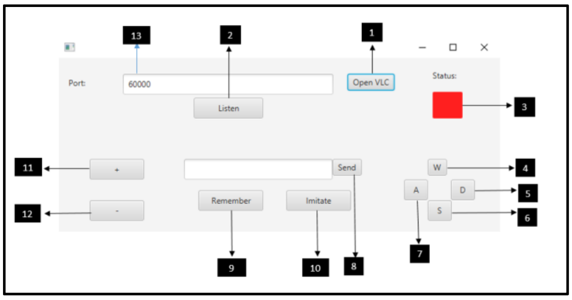

Fig. 3 Graphical User Interface (GUI) for SPHEMO

The description of each button of the GUI is as follows:

(1) Go Pro Camera Access Button
The GUI provides the two controller; one for the Go Pro and the other is for the controller of movement of SPHEMO. However, the one device controller will only be possible if the user has a Wi-Fi adapter which allows the laptop device to connect to two Wi-Fi connections (one for the Go Pro and the other for the hotspot connection). If the user is using a Wi-Fi adapter and was able to connect to the two connections, by clicking the Open VLC button, the user can access the Go Pro camera streaming.

(2) Listen Button
The user has to click this button in order to establish the connection.

(3) Status Indicator
Once the Listen Button was clicked, the status indicator will turn yellow which means the controller is waiting for SPHEMO to connect. Once it turns green, the connection between the controller and SPHEMO has been established.

(4) "W" button
Commanding SPHEMO to move forward.

(5) "D" button
Commanding the pendulum of SPHEMO to move the angle to 150 degrees (turning right).

(6) "S" button
Commanding SPHEMO to move backward.

(7) "A" button
Commanding the pendulum of SPHEMO to move the angle to 30 degrees (turning left).

(8) Send button
This is used if the user wants to send series of commands.

(9) Remember button
The user has an option to switch to auto-pilot mode. Clicking the remember button once means the recording of the commands sent by the user will be recorded. Clicking it again means stops the recording.

(10) Imitate button
After clicking the remember button, clicking the imitate button allows SPHEMO to replicate the recorded manual control of the user.

(11) "+" Button
Increases the pulse width modulation of SPHEMO by 60. (Acceleration) This will take effect once the "W"or "S" button is pressed.

(12) "-" Button
Decreases the pulse width modulation of SPHEMO by 60 (Deceleration). This will take effect once the "W" button or "S" button is pressed.

(13) Port
The port is initialized on the code part. In this case, the researchers use 60000. No need to change the port number.

Presented at the 4th DLSU Innovation and Technology Fair 2016
De La Salle University, Manila, Philippines
November 24 & 25, 2016

4. RESULTS AND DISCUSSION

4.1 Implementing Multidirectional Control on SPHEMO

A. Forward and Backward Movement

The research assesses the straightness of the movement of SPHEMO by commanding the SPHEMO to only move forward and backward without controlling its turning in the middle of the run. The distance that was travelled by the SPHEMO is up to 2 meters. The distance was cut into 8 having a gap of 0.25 meters to determine the path where the SPHEMO landed on each of the distances.

Fig 4 and Fig 5 displays the graphical representation of the ball as it travels in a forward and backward direction.

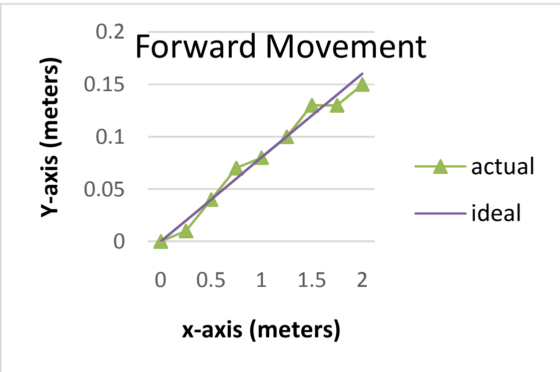

Fig. 4 Forward Movement

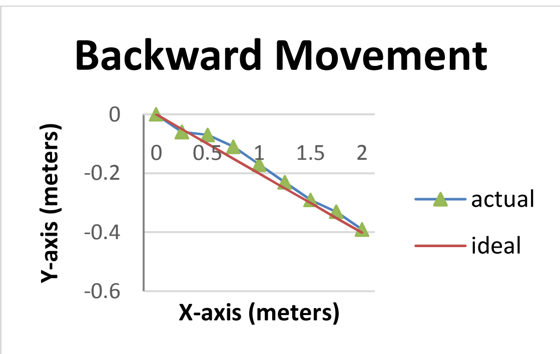

Fig. 5 Backward Movement

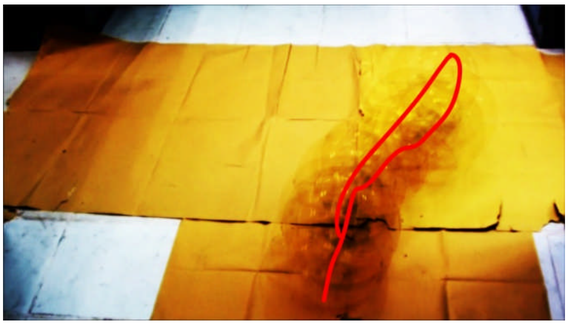

Fig. 6 Forward -Backward Movement Tracks

Shown in Fig. 6 is the actual forward-backward movement tracks performed by SPHEMO.

B. Left and Right Movement

The researchers tracked the path of the SPHEMO as it turns left and right. After getting the points where it landed during the run, the researchers produce a parabola graph. This is to prove that the SPHEMO creates a circular path as it moves to these said directions. After the experiment, the researchers looked for the apex value. Apex is the highest part of something in a form of point, which in this case is the parabola. Fig 6 and Fig 7 presents the apex per trial, which is the highlighted value on the table above.

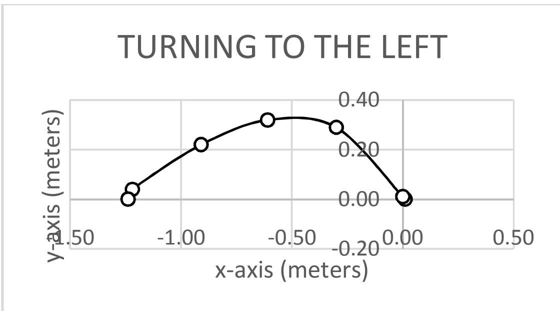

Fig. 6 Left Movement

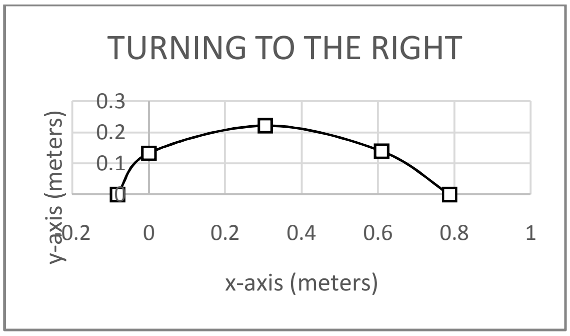

Fig. 7 Right Movement

Shown in Fig. 8 is the curve movement of SPHEMO.

Fig. 8 Curve Movement

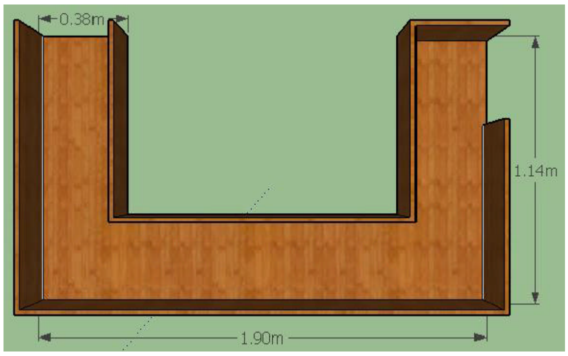

Fig. 8 Air-condition Duct

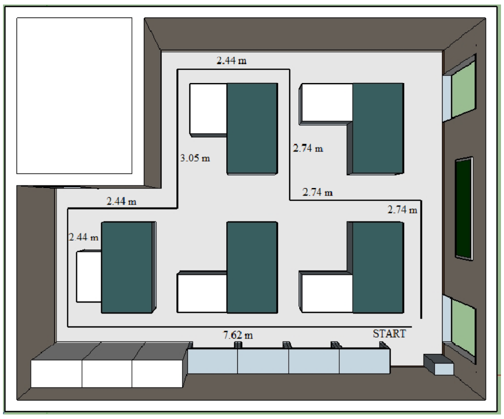

Fig. 7 Room Floor plan

4.2 Maneuver in Controlled Environment

A. Air-condition Duct

The researchers made a prototype for air-conditioned ventilation duct that is made of wood. It has a dimension of 0.38 by 0.38 meters and a total travel distance of 4.18 meters, the design is shown in Fig. 8.

B. Room Environment

The researchers chose a room that has a smooth surface for the room environment. The room that was used is located in DLSU-STC at room E013. Fig 9 shown below is the top view floor plan of the room including the path and the distance where the SPHEMO will travel.

C. Manual Control and Camera View

In assessing the controls of the SPHEMO, 15 trials were conducted on a room environment for the manual control in order to test how long it would take for the SPHEMO to complete the whole course. The average time it took for the SPHEMO to finish the whole room environment using the manual control is 2 minutes and 43 second with a total distance of 26.21 meters. The average speed is 0.16 m/s. On the other hand, 5 trials were conducted in order to assess the control using the camera view. The average time it took for the SPHEMO to finish the whole run is 6 minutes. One of the problems that were encountered on the camera view control is that the camera has a delay which results to a longer time to finish the whole course. Figure 8 shows the camera view.

Fig. 8 Camera View

4.3 Autopilot System Assessment

The researchers conducted three different trials in order to assess the auto pilot system of the SPHEMO. The accuracy and precision were computed for the first two trials and precision on the last trial. For the first two trials, a target point was set on the starting and on the destination line. After manually controlling the SPHEMO, the SPHEMO has to repeat the same set of commands for 10 consecutive trials. As for the result of the first trial, the overall accuracy of the manual control with respect to the target point set is 82.90%. On the other hand, the accuracy of the path points to the manual control is 69.44% and the mean absolute deviation of the path points to each other is 0.02 meters on the starting line and 0.17 meters on the destination line. As for Trial 2, the overall accuracy of the manual control with respect to the target points set is 89.48% and the accuracy of the path points to the manually control is 56.57%. The absolute deviation of the path points on the starting line is 0.02 meters and 0.20 meters on the destination line. For the last trial, only the accuracy and absolute deviation of the path points to the manual control were computed. For Trial 3, the SPHEMO creates its own path with no target point set and this has to be repeated for 10 times having 8 sampling points. As for the result, the accuracy of the path points to the manual control is 34.54%. The mean absolute deviation for each sampling point is (0) 0.05 meters, (1) 0.07 meters, (2) 0.09 meters, (3), 0.25 meters, (4) 0.29 meters, (5) 0.26 meters, (6) 0.35 meters and (7) 0.37 meters. Figure 9 shows the sample graph of Trial 3 for all 10 repetition tests with 8 sampling points each.

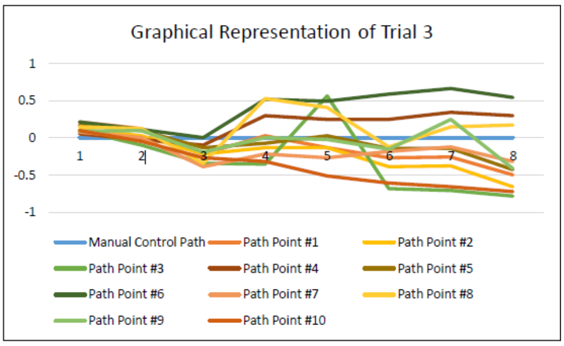

Fig 9. Trial 3 Representation Graph

4.4 Stability Assessment

The researchers designed the stability of SPHEMO by maintaining the weight on the center under the shaft of the SPHEMO's structure. Moreover, balancers were used at each side of the SPHEMO to achieve an equal weight on both sides. In order to test the stability of SPHEMO, 30 trials were conducted by tilting the SPHEMO on one side and the time it took for it to stabilize was recorded. The average time it took for the SPHEMO to stabilize is 11.54 seconds which means if the SPHEMO is titled on one side, it will take 11.54 seconds for the SPHEMO to go back to its stable positon.

5. CONCLUSIONS

In this study, the authors designed and implemented a mobile surveillance camera with stability control. The design is a spherical mobile robot called SPHEMO. It was successfully tested to roam a classroom and an air duct. Design of the SPHEMO was divided into two; hardware and software. For the hardware design of the SPHEMO, a shaft was installed inside together with a brushless dc motor to actuate its forward and backward movement and a servo motor to actuate its left and right movement. It also uses a pendulum to stabilize the left and right turning of the SPHEMO. As for the software design, the authors used two Wi-Fi connectivity in order to control the SPHEMO and the camera. SPHEMO also has successfully demonstrated its auto pilot system relative to the set target path by the authors and the allowable distance set by the authors.

The stability of the structure was also tested. The average time it took for the SPHEMO to stabilize when it is tilted on one side is 11.54 seconds. Additional tests were done in order to test the SPHEMO's limitations. The range limit of the Wi-Fi connection of the controller is only up to 8 to 10 meters while the Wi-Fi connection of the camera is 10 meters.

For future work, a higher motor capacity can be used in order to carry the whole weight of the SPHEMO. It is also recommended to use a higher battery capacity that can power up the whole system. Also, it is recommended to use a camera which does not have a delay so that it would be easier to control the SPHEMO. It also needs to improve SPHEMO's navigation by decreasing the percentage difference. Implement a longer range for Wi-Fi connection and to use other means of connectivity for the camera if possible, to eliminate the use of Wi-Fi adapter or the use of two devices. Lastly, use a clear ball for better video capture.

6. ACKNOWLEDGMENTS

The authors would like to express their utmost gratitude to the following who have done enormous contribution in making this study possible: to Dr. Melvin Cabatuan, Engr. Roy Francis Navea and Engr. Joseph Berlin Juanzon for their suggestions and recommendations in the design and implementation of the project; to their family and friends who consistently supported the researchers throughout the study; and most of all, to the Almighty God who made everything possible and their source of strength.

7. REFERENCES

Arduino MEGA 2560, (2016). Retrieved from https://www.arduino.cc/en/Main/ArduinoBoardMega2560

Bhattacharya, S. and Agrawal, S. (April 2000). "Experiments and motion planning of a spherical rolling robot," Proc. of the 2000 IEEE International Conference on Robotics and Automation

Bicchi, A.; Balluchi, A., Prattichizzo, D. and Gorelli, A.(1997). "Introducing the "SPHERICLE": an experimental testbed for research and teaching in nonholonomy," Proc. of the 1997 IEEE International Conference on Robotics and Automation,

Biscuit Platform, (2015, January 9) Retrieved from https://github.com/RedBearLab/Biscuit

"Building an Electic Hamster," Built to Spec blog. (2011, November 18). Retrieved from http://www.built-to-spec.com/blog/2011/11/18/building-an-electic-hamster/.

Crossley, V.A. (2006). A literature Review on the Design of Spherical Rolling Robots. Department of Mechanical Engineering Carnegie Mellon University, Pittsburgh.

Cytron, MDD10A Dual Channel 10A DC Motor Driver, (2016). Retrieved from http://www.robotshop.com/en/10a-5-30v-dual-channel-dc-motor-driver.html

ESP 8266 WiFi Shield, (2016). Retrieved from http://www.e-gizmo.com/KIT/esp11shield.html

GoPro Hero3. (2014, August 12). Retrieved from https://gopro.com/

GroundBot Rotundus, http://www.rotundus.se/

Halme, A.; Suomela J., Schönberg T. & Wang Y. (1996). A Spherical Mobile Micro-Robot for Scientific Applications, Proceedings of ASTRA '96, ESTEC, Noordwijk, The Netherlands, Nov. 1996

Husoy, K. (June 2003). "Instrumentation of a spherical mobile robot," NTNU, Department of Engineering Cybernetics, Trondheim

Javadi, A. H. and Mojabi, P. (May 2002). "Introducing August: a novel strategy for an omnidirectional spherical rolling robot," Proc. of the 2002 IEE International Conference on Robotics & Automation, Washington, DC.

Mukherjee,R. , Minor, M. A. and Pukrushpan, J. T. (December 1999). "Simple motion planning strategies for Spherobot: a spherical mobile robot," Proc. of The 38th Conference on Decision & Control, Phoenix, AZ.