Touchless Human-Mobile Robot Interaction using a Projectable Interactive Surface

R. Agarwal[1], P. Sharma[2], S.K. Saha[3], and T. Matsumaru[4]

***Abstract*— This paper showcases the development of a mobile robot integrated with Projectable Interactive Surface to facilitate its interaction with human users. The system was designed to interact with users of any physical attributes such as height, arm span etc. without re-calibrating it. The system was designed in such a way that there would be no need for the human to come in physical contact with the robot to give it instructions. This system uses a projector to render a virtual display on the ground allowing us to project large displays. Microsoft Kinect integrated in the systems performs a dual functionality of tracking the user movements along with mapping the surrounding environment. The gestures of the tracked user are interpreted and an audio visual signal is projected by the robot in response.**

I. INTRODUCTION

Robots in workplace and at homes have become a common norm these days. With increasing acceptance of robots in our societies there could be other challenges that come into play. Navigating through an unknown environment and remembering it can be difficult for a human. But with the help of smart interaction between humans and an autonomous robot that can map the same environment, we have come a long way in finding solutions to these problems.

In this paper, we are proposing an integration of two systems, IIT Delhi in-house developed industrial grade robot RoboMuse and the SAKSHAR system(projectable interactive surface)[1]. The technology of a projector: Kinect has been used to facilitate the interaction between the robot and a human who wishes to interact with it. The idea behind developing the RoboMuse 4.0 was to leave the robot in the common area in the university and help students and visitors find different locations in the campus by interacting with the robot. RoboMuse uses ROS (Robot Operating System) as its meta-operating software to integrate ultrasonic sensors and Kinect to implement collision avoidance. Using the Visual odometry from kinect, we implemented 3D environment mapping for navigation of the robot in unknown environments. With the Human Detection API's in Kinect the robot can identify nearby human and their joints and interact with the human from the touchless interface developed from Sakshar. The touchless interface allows a really easy way to interact with the robot with no need to be bend or stand in a particular order and at the same time avoids any kind of human-robot contact. The main motive behind the concept of interacting with the robot without toughing is to:

1) Reduce the chances of physical damage of the robot due to the form contact of the human with the robot
2) Improve easiness of communication of the human with the robot irrespective of his/her height because of communication by gestures and not by contact

The way the robot functions is it continuously navigates through the environment using its collision avoidance algorithm and performing a simultaneous mapping to identify covered areas. While navigating the area the Human Detection API continuously runs in the background of the ROS. Hence, as the robot identifies a human it shifts from the navigation phase and moves towards the human. As the robot reaches a certain already specified safe distance to human it starts its touchless interface in front of the user and waits for the input from the user. When the user is done interacting with the robot or after a certain period of no input from the user, the robot goes back into the autonomous navigation mode.

The concept of using a projected screen as a bidirectional interface started with research on step on interfaces. Bidirectional communication would mean the user would be able to give the robot instructions and the robot will be able to communicate back with the user through audio visual feedback. Based on the concept of a Step on Interface (SOI), the IDAT[2], [3], [4] device was designed for upper limb rehabilitation by training to improve eye-hand coordination[5], [6], [7], [8], [9]. Then the device named SAKSHAR was designed with the idea making primary education in developing countries more interactive and fun.

The system is different from the SAKSHAR system because the interaction with the projected screen has now been made touchless i.e. the user can stand at any height and use his hands from a height and watch the cursor move with the motion of his hand. The Kinect tracks the user in real time and then the projected position of the user's hand on the screen is where the cursor is formed.

A rough sketch of the paper will be as follows. Section II will give a brief introduction to the RoboMuse. Section III talks about the design and control of the system. Followed by this Section IV will discuss about the Sakshar in particular. Section V will discuss the integration of these systems and the specifications of the resultant robot and the tasks that it can perform. Section VI will outline the system performance

*Research supported by Waseda University (2016B-203), Kayamori Foundation of Informational Science Advancement (K26kenXIX-453) and BRNS, India.

R. Agarwal, P. Sharma and S. K. Saha are with the Dept. of Mechanical Engineering, IIT Delhi, Hauz Khas, New Delhi-110016, India; (e-mail: rishabhagarwal880@gmail.com, pratyusha1995@gmail.com, saha@mech.iitd.ac.in).

T. Matsumaru is with Bio-Robotics and Human-Mechatronics Laboratory, Graduate School of IPS, Waseda University, Kitakyushu, Fukuoka, Japan; (email: matsumaru@waseda.jp).

during some of the tests. Lastly, the conclusion of the paper will discuss the advantages of this system over existing systems and the scope of it's future development.

II. RoboMuse 4.0 Brief Introduction

RoboMuse as IIT Delhis own ingenious mobile platform series to demonstrate 24/7 working scenario, started in 2009. Over the past 6 years, RoboMuse has had 3 iterations with each one adding something unique to the design. Over all these years the RoboMuse was built with the essential of aim of it, functioning without any human interaction.

Fig. 1. RoboMuse 4.0

RoboMuse 4.0 is a custom made robot which was constructed with the aim to navigate through an environment without encountering any collisions and without any human supervision, as shown in Fig 1. While mapping the environment the robot looks for any humans around it and activates Sakshar to interact with them. Thus to achieve this, it is essential that the robot performs SLAM (Simultaneous Localization and Mapping) along with human detection algorithm on Kinect V2 mounted on the robot. Thus the robot constantly maps the environment as it navigates through and in the map, it localizes itself using state of the art techniques, i.e it identifies its exact location in the map. Thus the above said feature is implemented on the RoboMuse 4.0 on a ROS [10]. This is because the ROS provides a universal platform which has minimal dependence on the robotic system, thereby allowing us to produce a universal solution which could be implemented on other similar custom made robots with minimal changes. The robot produces a 2D and a 3D map as it navigates through the environment. The vision of the robot is achieved through the Kinect sensor, whose depth image can be used to create a laser scan, which is the requirement to produce a map. Thus in the 2D map the robot clearly marks the accessible and the inaccessible areas, thereby safely navigating through the environment.

III. Mobile Base Design and control of RoboMuse 4.0

The methodology used to develop RoboMuse 4.0 can be broadly classified into 3 categories namely mechanical, electrical and programming aspects. The mechanical part includes the design, analysis and fabrication of the robot. The electrical aspects include the circuit connectivity for the whole robot, motor controlling, sensor wiring, etc. The programming aspect includes visual servoing through ROS, collision detection algorithms, etc.

A. Design and Analysis

Before the design phase, the concept of RoboMuse 4 was decided after brainstorming a number of ideas. In the design phase the chassis and the rack were designed in SolidWorks and then load analysis was performed through simulation.

1) Conceptualization: Initially there were a number of ideas on the concept of RoboMuse 4. Some of these include pick and place robot, waste picking robot, warehouse robot, personal robot, surveillance robot, etc. After a lot of brainstorming a consensus (based on the requirement, multitasking ability, etc) was reached to develop a surveillance robot with multitasking abilities. Planned improvements from previous RoboMuse robots are:

- Automatic movement without the help of a line using Kinect
- Collision detection using ultrasonic sensors
- Modular design to integrate multiple applications
- Implementing open source platforms
- Arduino ATmega processor

2) Chassis: The base of the robot has been designed based on the dimensional constraints of the actuators (motors) and the tasks to be performed by the robot. The chassis of the base was designed in Solidworks as shown in Fig 2. Simulation was carried out for deflection and Von Mises stress. The maximum deflection turned out to be in the order of 10^{-2} mm and the maximum stress was around $10^7 N/m^2$ which is much less compared to the Yield Stress of Aluminium (4x10^8).

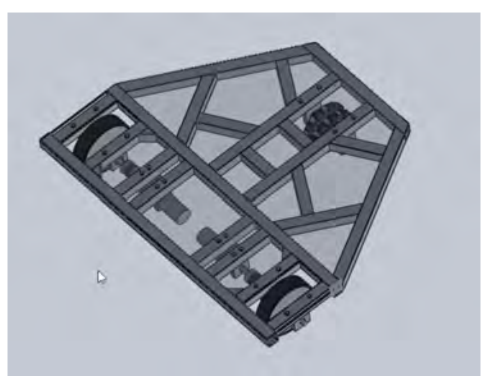

Fig. 2. CAD model of RoboMuse 4 chassis

3) Drive: The drive includes the motor, gear box, helical coupler, bearings, shaft, keys and the wheel as shown in Fig 3. The gearbox is attached to the motor at one end and a shaft at the other end. The shaft of the motor is linked to that of the wheel by means of a flexible helical coupler. The helical coupler provides flexibility deflections of the shafts and also transmits power efficiently. The wheel is supported by a shaft which in turn is supported by a set of bearings at both the ends. The shaft and the wheel are connected with the help of a key to transmit power. A pair of retaining rings is used to prevent the axial motion of the wheel over the shaft.

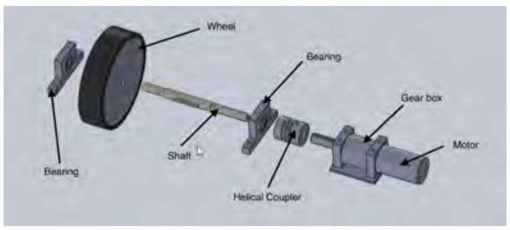

Fig. 3. Exploded view of drive

4) Rack: The main purpose of a rack is to equip the electrical equipment in two levels(bottom most), the laptop or Kinect in the uppermost layer and any objects to be transported or stored in the 3rd highest layer. The rack is designed to fit in the space above the drive. The rack is designed to be easily removable with respect to the chassis. The rack consists of four aluminium channels at the 4 corners and acrylic sheets as levels of the rack as shown in Fig 4. The maximum height of the rack is such that a laptop can be placed on it and it is easily accessible by a human being of height 180 cm. The rack was designed in Solidworks as shown below.

Fig. 4. CAD model of Rack

B. Electrical Circuits

The electrical circuit design determines the receptiveness of the robot. The simplicity and accessibility of the electrical components will ease the process of integrating new sensors and debugging if needed. To ensure the safety of the robot and their surroundings an emergency switch was mounted on the robot for completely stopping the robot.

1) Power Board: We have incorporated three 12V lead acid batteries to power all the components of the robot. The power board ensures the desired voltage output is generated as different devices need different voltages. Motors are running on 24 V whereas the microcontrollers and motor drivers need a 5V operating voltage.

2) Micro-Controller: The core of the circuit is an Arduino ATmega2560 microcontroller with 5V operating voltage. It is an open source hardware that can be programmed with the Arduino Software (IDE). Integrated with 54 Input-Output pins, it can easily incorporate multiple sensors and motors at the same time.

3) Motor Driver: To control the speed of motors it is essential to generate Pulse Width Modulated (PWM) signals which are generated by Sabertooth dual-motor driver with 25A continuous, 6-30V nominal capacity. Sabertooth allows us to control two motors with analog voltage, radio control, serial and packetized serial. Sabertooth has independent and speed+direction operating modes, making it the ideal driver for differential drive robots and more.

Sabertooth features screw terminal connectors - making it possible for you to build a robot without even soldering. Sabertooth also allows us to make very fast stops and reverses.

C. Operating Modes of RoboMuse

Running on ROS, the RoboMuse can easily transition from one mode to another based on the specified inputs. The RoboMuse uses the following specified modes to achieve the objective of smart autonomous human iteractive robot.

1) Collision Avoidance: In a step towards making robots intelligent, it is essential for them to sense their surrounding and make decisions accordingly. Integrating ultrasonic sensors and Kinect with RoboMuse equipped us with the ability to detect obstacles and navigate accordingly. Using the sensor odometry from both of the sensors and performing the sensor fusion, we implemented collision avoidance algorithm along with mapping the environment to remember the presence of the obstacles in the environment. The use of the ultrasonic sensor and Kinect are as specified bellow.

- Ultrasonic Sensors: These sensors based on measuring the properties of sound waves with a frequency above the human audible range. Ultrasonic sensors can be used to solve even the most complex tasks involving object detection or level measurement because their measuring method works reliably under almost all conditions[11]. The ultrasonic sensors we have used is the one with both transmitter and receiver integrated together and hence they work in reflection mode.
  In reflection mode (also known as echo ranging), an ultrasonic transmitter emits a short burst of sound in a particular direction[12]. The pulse bounces off a target and returns to the receiver after a time interval t. The receiver records the length of this time interval, and calculates the distance traveled r based on the speed of sound c:
  $$r = c * t^2 \qquad (1)$$

Three ultrasonic sensors are mounted on the front face of the robot with a relative angle of 60 degrees to cover the complete robot chassis.

- Kinect: We opted to use Kinect V2, for our visual navigation purpose as it provides RGB and depth data which can be used in future to integrate autonomous navigation via SLAM (Simultaneous Localization And Mapping). The Kinect v2 face recognition, motion tracking, and resolution are much more precise than the Kinect one. Kinect V2 uses time of flight technology to determine the features and motion of certain objects[13].
  Using the developer kit available through Microsoft we integrated the video out in our setup. We also integrated Human Detection API along with direct feedback to stop the robot when it detects human and switches to mode 2 that is the Sakshar Mode[14].
  Using SLAM with Kinect on ROS the environment 3D mapping was achieved as shown in Fig 5.

Fig. 5. Image of a 3D Map Created using a Kinect

2) User Interface: To model an intuitive user interface we opted for Processing, an open source language development tool for writing programs in other computers. Useful when we want those other computers to ”talk” with an Arduino, for instance, to display or publish some data to the Arduino. Arduino comes with some basic examples for communicating with Processing[15]. These are useful for when we want to write both Arduino and Processing programs and have them talk to each other. This works best for communicating simple information.

IV. Touchless interactive interface for RoboMuse 4.0

The objective of this activity is to make the interactive surface self-movable to facilitate human-robot interaction in a moving robot such as RoboMuse. Mounting screens or tablets on RoboMuse will further add to the weight. Additionally, usage of a screen for interaction by the users will bring them physically close to the robot and increase chances of damage. A projectable interactive system will facilitate the same level of interaction without any physical contact. This will involve bringing a change in the Kinect calibration technique because now the entire setup will be in motion. The changes in the calibration have been discussed in the section that follows.

The most interesting part is how this system will improve how users, different in terms of height, arm span and palm structures will be able to interact with the robot with the same easy. The system relies on gestures of the hand(irrespective of the length). These gestures do not have to be made at a particular height or a particular way. The kinect detects the human hand irrespective of the person’s height. It is just recognized based on whether the torso of the human is visible or not. The orientation of the kinect and projection of the projector is set up in a way to enable people of different heights to comfortably use the system. The method of calibration further enables this and it shall be discussed in the subsequent sections.

Fig. 6. Projected interactive screen on the ground

V. System integration

A. Sakshar Mounting

To be able to mount the complete set-up on RoboMuse, a light, strong and detachable stand was designed for the projector and a mechanism to mount the Camera Stand of the Kinect was designed. The devices had to be locked into place because the complete setup would be in motion and there had to be no risk of the devices falling down when in motion. The Sakshar was to be mounted on the front side of the robot. This provokes the need for a mounting in order to accommodate the projector. For this purpose, a mounting is designed with aluminium sheets and channels of thickness 2 mm.

B. RoboMuse and Sakshar Integration

The device is programmed in such a way that the projector displays the interactive screen only when a human is detected within a particular radius of the robot. It is at this instant that the robot shifts from operating as a moving robot to displaying the interactive screen so the standing user can operate it as shown in Fig 6. The Kinect is used for a dual purpose on this robot and it would alternate between these modes of operation depending on whether or not a human is standing in close proximity to its region of projection. As of now the SAKSHAR mode uses the interface developed for making primary school education more interactive and fun but this can be replaced to suit the user's needs.

C. Calibration

The calibration of the setup earlier required us to place the Kinect at a distance such that the plane of projection and the person's body upto the torso was visible[1]. The programme would then instruct the user to touch the different end points of the projection plane to register the coordinates of the end points and construct the plane. This had to be done only once. An average of the data point recorded for each of the points was taken into account while constructing the plane. This minimised the error due to fluctuations in data recording.

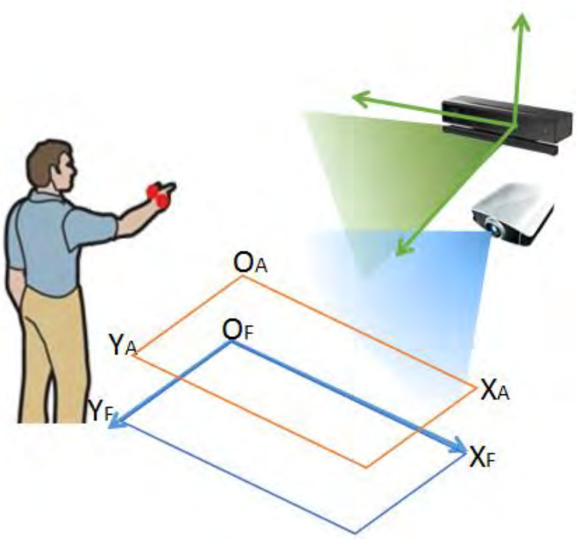

Fig. 7. Calibration of plane in the air parallel to floor plane

For the set up of the mobile robot , the plane of projection of the projector was the floor while the user would be standing upright and interacting with the device. The system had to be designed to work for people of all heights. A plane parallel to the floor plane was constructed in thin air at a height of 2 feet from the floor. This height was chosen for the system to work for people with heights greater than 2ft. The dimensions of the projected plane could be varied depending on the height of the projector and angle of inclination. This would give us flexibility to adjust the screen height. The dimensions of the screen are 30cm by 40cm as of now and can be varied. A rectangle with dimensions equal to that of the rectangle was drawn out. A similar procedure could now be followed for the rectangle traced out in thin air. The program would now believe that the user was interacting with a screen in thin air but the output displayed would be still displayed on the ground. Because this system continuously tracks the user position, the user can move freely to reach out to the buttons of his choice. In the earlier version of the projectable interactive surface[1], where the projected screen had to be made touch sensitive, the height of the hand from the plane would impose additional constraints for detection and touch. This constraint was now removed. Now the Kinect no longer had to be placed at an angle for it to view the projected plane. This way it could be mounted on top of the mobile robot facing the user and not having to look at the screen.

$$O_A = [o_1, o_2, o_3] \tag{2}$$

$$X_A = [x_1, x_2, x_3] \tag{3}$$

$$Y_A = [y_1, y_2, y_3] \tag{4}$$

$$OX_A = [o_1 - x_1, o_2 - x_2, o_3 - x_3 \tag{5}$$

$$OY_A = [o_1 - y_1, o_2 - y_2, o_3 - y_3 \tag{6}$$

There is a function of a continuous cursor which projects the location of hand over the plane on the screen. This has been discussed in the paper on SAKSHAR[1]. It is analogous to the function of a mouse. To decrease the cursor speed and improve user control the rectangle traced out was larger than the real size of the screen of projection. This way a larger rectangular area would be mapped to the small projection area of the projector. Now when a user appears in this region her/his hand begins to get detected. This was done so that the user would have to move a significant distance to be able to move the cursor otherwise even a small displacement of the hand would result in the cursor moving really fast. Hence the cursor speed was brought down and the user would be able to control the cursor better. Once the setup is calibrated this way and the Kinect and projector are not disturbed, their relative positions in space would stay the same and re-calibration will not be required.

$$Ax + By + Cz + D = 0 \tag{7}$$

$$A = (x_2y_3 - x_3y_2) \tag{8}$$

$$B = -(x_1y_3 - x_3y_1) \tag{9}$$

$$C = (x_1y_2 - x_2y_1) \tag{10}$$

$$D = -[o_1.(x_2y_3 - x_3y_2) - o_2.(x_1y_3 - x_3y_1) + o_3.(x_1y_2 - x_2y_1)] \tag{11}$$

VI. System Performance

To test the system functionality and performance we started testing the robot in the closed lab environment. An onboard PC running ROS was used to control the Robo-Muse and Sakshar. The robot was initialized to navigate the environment, avoiding obstacles and performing the 3D mapping simultaneously. While the robot was navigating it was continuously looking for a human in the environment. As soon as it detected the human it shifted from its navigation mode and moved close to the human and started its Sakshar

Fig. 8. User interacting with RoboMuse

mode. For the test purposes the interactive display was projecting a game.

The figure 8 shows a kid interacting with the robot. The robot was displayed in the open house of IIT Delhi and became particularly famous among the school kids. The idea of playing games with the robots exited children and parents alike. The idea of multiple users using the system with the same level of ease was validated during our trials in the Open house exhibition where multiple users of different heights and ages used our system without any resetting of the system or recalibration.

During the initial testing robot was able to avoid all the major obstacles but the environment map showed a lot of error due to the noise created by the robot motion. Also the current system using a onboard PC and projector has a very little runtime due to the high power consumption.

VII. CONCLUSIONS

A projectable interactive system was integrated with a mobile robot system. The Kinect was used to recognize gestures and commands given to the robot. The projector was used to display the responses of the robot to such actions. This system is better in comparison to it counterparts such as using a touch screen or buttons because this does not need the human to make any form of physical contact with the robot directly or indirectly and hence will minimize the risk of any damage to the system. The system is calibrated such that a user of any height can use the system effectively without any trouble. The calibration design helps us get over the ergonomic design constraint that is introduced because of the height, hand span or other physical attributes of the user. The calibration needs to be performed only once before use. The main motive behind building such a system is to make the interaction of humans and robots in the public space more convenient and hassle free.

The key performance parameter of the system is it usability among users, which is extremely hard to quantify, but its acceptance rating could be used in future tests as one of the measuring parameters.

The future aspects would include implementing optimized motion control for reduction of noise in visual odometry. Also, using a single board PC and low power led projectors to enhance the runtime of the robot would be some important areas of focus.

VIII. ACKNOWLEDGMENT

The research was supported by a grant towards setting up of Programme for autonomous Robotics by BRNS, India. The collaborative works were supported by Graduate School of IPS, Waseda University through its Grant for Special Research Project (2015B-3461, 2016B-203). The authors acknowledge the help from J. P. Khatait for his perceptive criticism, Mr. Vishal Abhishek and members of the Robotics Club.

REFERENCES

[1] P. Sharma, R. P. Joshi, R. A. Boby, S. K. Saha and T. Matsumaru, ”Projectable interactive surface using microsoft kinect V2: Recovering information from coarse data to detect touch,” 2015 IEEE/SICE International Symposium on System Integration (SII), Nagoya, 2015, pp. 795-800.

[2] Liu, Y., Jiang, Y., and Matsumaru, T., 2012. Development of Image- projective Desktop Arm Trainer, IDAT, IEEE/SICE International Symposium on System Integration (SII), pp.355-360. doi: 10.1109/SII.2012.6426964

[3] Matsumaru,T.,Jian,Y.,andLiu,Y.,2013.Image-projectiveDesktop Arm Trainer IDAT for Therapy, 22nd IEEE International Symposium on Robot and Human Interactive Communication (IEEE RO-MAN 2013), pp.501-506. doi: 10.1109/ROMAN.2013.6628411

[4] Matsumaru,T.,Liu,Y.,Jiang,Y.,andDai,C.,2014.ImageProjecting Desktop Arm Trainer for Hand-Eye Coordination Training, Journal of Robotics and Mechatronic, vol.26, no.6, pp.704-717.

[5] Shiyang Dong, Takafumi Matsumaru: ”A Walking Training System with Customizable Trajectory Designing”, Paladyn. Journal of Behavioral Robotics, Volume 5, Issue 1, pp.35-52, (2014.06). DOI: 10.2478/pjbr-2014-0003

[6] Takafumi Matsumaru, Yasutada Horiuchi, Kosuke Akai, Yuichi Ito: ”Truly-Tender-Tailed Tag-Playing Robot Interface through Friendly Amusing Mobile Function”, Journal of Robotics and Mechatronics, Vol.22, No.3, pp.301-307, (2010.06). DOI: 10.20965/jrm.2010.p0301

[7] Takafumi Matsumaru, Kosuke Akai: ”Step-On Interface on Mobile Robot to Operate by Stepping on Projected Button”, The Open Automation and Control Systems Journal, Vol.2, pp.85-95, (2009.11). DOI: 10.2174/1874444300902010085

[8] Takafumi Matsumaru: ”A Characteristics Measurement of Two-dimensional Range Scanner and its Application”, The Open Automation and Control Systems Journal, Vol.2, pp.21-30, (2009.05). DOI: 10.2174/1874444300902010021

[9] Takafumi Matsumaru, Kosuke Akai: ”Functions of Mobile-Robot Step-On Interface”, Journal of Robotics and Mechatronics, Vol.21, No.2, pp.267-276, (2009.04). DOI: 10.20965/jrm.2009.p0267

[10] ”Documentation - ROS Wiki”. Wiki.ros.org. N.p., 2016. Web. 13 June 2016.

[11] Ultrasound. Retrieved from http://www.sensorwiki.org/doku.php/sensors/ultrasound

[12] *Ultrasonic Sensors. Retrieved from http://www.pepperl-fuchs.us/usa/en/classid182.htmlview=productgroupoverview*

[13] *How Does The Kinect 2 Compare To The Kinect 1? (2014). Retrieved from http://zugara.com/how-does-the-kinect-2-compare-to-the-kinect-1*

[14] *Developing with Kinect for Windows. Retrieved from https://developer.microsoft.com/en-us/windows/kinect/develop*

[15] Arduino Playground - Processing. Retrieved from http://playground.arduino.cc/Interfacing/Processing