Tablo

Patrick Green, *CSE*, Bowen Yu, *CSE*, Joo Young Park, *CSE*, and Perveshwer Jaswal, *EE*

***Abstract*—Tablo is a gaming system that uses camera detection to allow users play board games on a two-dimensional (2D) display projected onto any household table. The motivation for such a system is so that users can play a number of their favorite board games on demand with the convenience of not having to worry about the anxieties of losing game pieces or having to organize stacks of containers for their board games. The concept of the whole system is that families and friends can play digitalized board games around a table in their kitchen or living room without hassle of managing many cardboard and plastic pieces that compose their board games. The design of Tablo is such that a projector is held overhead, about 5 to 6 feet above the user's table, and with it, the camera will be hung alongside the projector. The camera will use color detection, as the player will hold a specifically colored beacon that acts as the controller for the game, and will send data to the user's laptop so the computer can track the motion of the user's movement.**

***Index Terms*— Camera detection, Board Games, Projection**

I. Introduction

In this project, we present our gaming system, Tablo, that can help to improve the experience of playing interactive games such as board games. Board games have many small components and pieces, thus people face the issue of losing these pieces or have difficulties storing several of their board game sets. The digitization of these board games via Tablo will aid in removing the anxiety of losing pieces, as well as enabling the user to store numerous board games in one, compact set. Tablo can also offer players many more features to traditional board games, since all the games are digitized, we can add new features that can further customize the game settings and appearances. For example: we can change colors of the game pieces, add unique scoring systems, provide tutorials on how to play the board game, etc.

This problem has a very large market, according to [1], sales of board games have been rising every year for the past decade. Board games have been increasingly becoming more and more popular as friends and families are attracted to the notion of being able to bond and connect with each other face-to-face around a table. Studies shown in [2] support the notion that board games can enhance children's abilities to learn and grow to be adaptive. The visual graphics and hand-eye coordination allows for children to grow their critical thinking capabilities, and families have a more affinity to buy their children board games. Board games in general come with strategies and requires wits and logical reasoning to win against your opponents in any board games at any age.

The gaming industry has involved 1.2 billion people around the world, and there are tens of millions of people playing board games who would love for their gaming experience to be enhanced. Many digitalized board games exist for enhanced experience, such as Scrabble, however these digital versions of board games allow for only one game. This is to say that there is no database for multiple board games that can be saved in one system, in lieu that each board game has its own requirements and uniqueness to it such that one interface cannot accommodate for all types of board games. This is also to say that board games cannot have electronic features to them; board games are not only limited to just plastic pieces with no dynamics. According to [3], many implementations for digitalizing board games is explored. These techniques used are for the sole purpose to make board games exist in the digital world, not necessarily to make the experience of playing board games easier or more convenient. Figure 1 depicts our vision as to what our final product will be. Users would mount their projectors above the table and have a control box that is mountable on the side of the table for easier user interface.

A depiction of our final system that is intended to be demonstrated for the final presentation of Senior Design Project is shown in Figure 1. This merely shows the concept of our gaming system, which is that the board games will be projected onto a table, on which users can act upon to play their board games intuitively.

Fig. 1. Depiction of our gaming system. Projector and camera will be hung over user's table, about 5 feet to 6 feet above their table. The control box, where our PCB board will be used, is mounted on the edge of the table to allow users to perform necessary functions such as: pausing the game, resetting the game, or switching the amount of players in the game, just to name a few.

Our solution is to propose a more user-friendly environment in which anyone of any age can use the gaming system we propose to make playing board games more accessible and easier to play. The gaming system we call Tablo, a spin-off from table games, will store board games in one interface and will use camera detection to track user interactions to allow the user to play several hundreds of types of games in one sitting. We envision the users playing at a normal table in a regular household, with a small, light-weight camera overhead, and the users being able to see a projected view of the game they play on the table before them, with a control box that utilizes the printed circuit board (PCB) to switch form one game to another.

This gaming system will strengthen the board game community and have a positive impact for players as this system shall remove the anxieties of losing game pieces and save players the hassle of storing numerous boxes containing their games, as opposed to one electronic gaming system. The system specifications listed in Table 1 shall ensure that the gaming experience with Tablo is an enjoyable one.

Table 1		
Requirement	**Specification**	**Value**
Portability	camera weight	< 5 ibs.
	PCB control	Mountable to table edge
	Projector weight	< 20 ibs.
Response	Delay	< 100 ms
Safety	Projector	Mountable up to 6 feet above table
User Compatibility	Number of players	1-4
	Display area	$2ft^2$ to $6ft^2$
	Control box	Adjustable gaming interface

Table 1. System specifications table for project Tablo. Qualitative and quantitative values listed. The power dissipation of the PCB should be no more than 50 mW.

The system specifications, shown in Table 1, reveal the needs for our project. We want a safe way to mount the projector over the table that users will be playing on, this will be above player's heads, so the weight of the projector must be minimal. We would like to have an adjustable display such that users can adjust the projection of their game onto their table, as different people have different table sizes. The control, where we shall implement our PCB board, will allow users to simply perform functions necessary to facilitate a convenient way to regulate the games being played. Functions that the control box will iterate are: resetting the game, pause, adjust camera sensitivity by the lighting in the room, and change the amount of players allowed in the game, just to name a few. In this work, we revise the outcomes of our mid-year design review (MDR).

II. Design

A. *Overview*

Tablo is intended to be played as a digitized board game, as such our prototype, see Figure 3, allows the users are able to play intuitively on a flat surface as if they were playing on their coffee table. This is accomplished by having a projector above the players' heads, mounted on the ceiling of their room, and have the game displayed onto their table. The emphasis here is, of course, users do not need cardboard to play their game, let alone a special piece of material, this design is meant to be played on any surface. This is possible due to our new and improved tracking method, using Open Source Computer Vision (openCV) library, "ArUco", which allows our camera to detect the movements of different players' controllers. See Fig. 2 for our final product.

Our system block diagram shows how our components work together, see Figure 2.

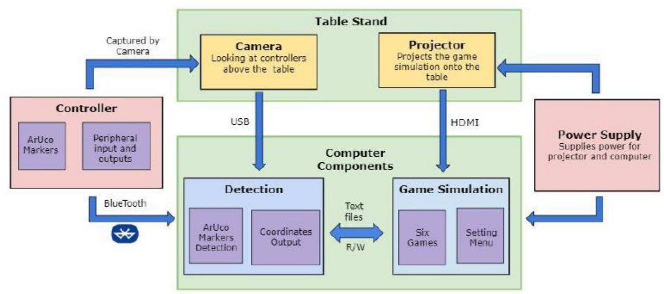


Fig. 2. System block diagram for Tablo. Only components using power supply are the projector and the user's laptop, the computer that runs the gaming simulations (the "Game Simulation" block). The Projector is connected to the computer for display purposes only, thus we use HDMI for the connection. The laptop has two algorithms running, one of course is for the games, and the other is for processing the images of the controllers that the camera sees, these binary bits from the markers are sent as text files for the game simulation to read. Please note that the diagram states we have operating Bluetooth connect from the controller to the computer, we explain the situation for our PCB in section III, project management, but for demonstrative purposes, wireless mouses were used.

As one can imagine, the users would be playing their favorite, classic board game, such as chess, and the controller has the ArUco marker exposed to allow the camera to see at all times from the ceiling. Alongside the camera, is the projector, both mounted on the ceiling, the camera will send information on position of the players' movements and identification (ID) tag to the computer, or the user's laptop. The computer processes the game and incorporates information in real time of what actions the player is taking to interact with the simulated objects, and displays all results and progression of the game in real time onto the table, via the projector. Figure 3 shows how the system is built to work together.

Fig. 3. Tablo, final product. Highlighted components are (from bottom to top): the surface area on which users can play games, the camera for tracking motion of users' controllers, our laptop for simulation of games, and the projector to display games. This apparatus was placed upon a table, such that the surface area on which users play the game can be played at a typical height of a table. The supportive, wooden beams are necessary to allow for players to experience playing board games wirelessly and digitally, as we could not mount the projector on the ceiling of the common room.

The project proved to be successful in the manner that the system should be played as. The project invoked much joy from children playing games, and the system provided an easy, efficient means for users to play game wirelessly on a table. One downside is that our controllers did not come out as we would have liked due misfortunate events, to be explained in further detail in section III, project management. Our technical specifications are given in Table 2.

Specification	Goal	Actual
Response time	150 ms	286.25 ms
Camera detection	50 ms	6 ms
Tolerable speed of controller	0.3 m/s	0.198 m/s
Supported players	2+	2
Environment	Any lighting condition	300 lux to 500 lux

Table 2. System specification for Tablo for FPR. Response time of entire system, for user feedback is 286.25 ms. The tolerable speed that which the user can move the controller to allow the camera to detect the coordination of the ArUco markers is 0.198 m/s. The lighting condition is such that the camera can detect the markers in a normal environment for lighting, such as a library room, or classroom (lux is SI unit for flux per unit area, and 300 lux – 500 lux is the brightness of an average office room with fluorescent lights in the ceiling).

Thanks to our new detection mechanism that uses the ArUco markers which is from ArUco library in Opencv, we were able to achieve proper camera detection in normal lighting conditions. Our timing responses varied for different goals, what matters most is that the games we had in Demo Day showed us that players were able to play games effectively, since the action required to play the games intended do not need much fast hand movements, and the system response time was still appropriate to play any one of the games designed to be played enjoyably, though as the project ran for hours in demo day, the response in pong game became slower because the system was tired.

B. Detection Mechanism

Our detection mechanism has improved since MDR, and now allows our camera to detect players' controllers in any lighting condition in their rooms. Initially, we used color tracking, in which our "controller" was a red ball attached to the end of a stick that the user would move around. This mechanism proved unreliable, as the camera would lose tracking once the lighting condition changed in the environments. It needs to be readjusted every time the lighting condition changes. For example, if the room was dim-lighted, i.e. fluorescent lights from the ceiling are off and windows sealed, the camera would not be able to detect the red ball. The ArUco ID tags, or markers, allow for more reliable tracking for a variety of lighting environments. It also makes multi-player easier because all we need are distinct markers. See Figure 4, [5], for examples of ArUco markers.

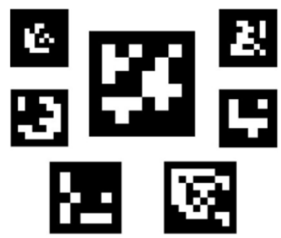


Fig. 4. Examples of ArUco markers. Each marker is unique by its size and pattern. Each pattern has an internal binary matrix that determines its identification. All markers have a black border surrounding the image of the white pixel patterns. Image courtesy of *docs.opencv.org/3.1.0*.

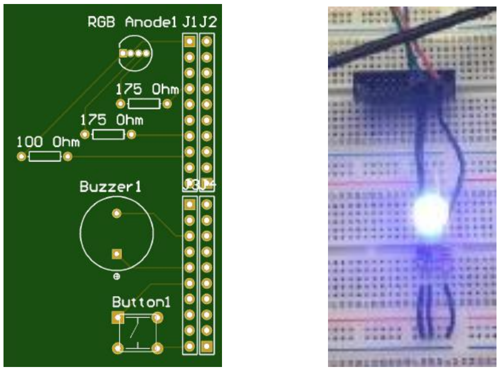
The advantage of using ArUco markers is that no matter how the image is rotated, our camera will be able to detect its movements and directionality, as opposed to a red ball that looks indefinitely the same no matter how it is rotated, see Figure 4 for example. All four corners of each marker has their own coordinate, the camera will see different orientations for each marker and be able to distinguish IDs unequivocally, which is why we are able to support multiple players for our games as well as accurately track their movements. Each marker's binary matrix is determined by its size, for example, we use the DICT 6x6 250 dictionary, thus all markers from this dictionary have 36 bits of information. See Fig 5.

(a) (b)

Fig. 5. (a) shows an example of one of the 6x6 ArUco markers we use with ID tag of 150. (b) shows our teammate, Joo Young, holding the ArUco marker in the case that will be put on our controller. As you see, the image is rotated,and the camera still detects it with the same ID tag, 150.

C. Impact Processing and Communication

Disappointingly, we were not able to establish communication with our laptop to the controller. See Figure 6 for our PCB layout and prototype. Our PCB was designed using the Altium application, and was shipped from JLCPCB. The cost, as per the invoice, is 9.30 USD, see Appendix A for prices of all components used in project, (we made our PCB of Ro-HS compliant).



(a) (b)

Fig. 6. (a) PCB layout. Layout is designed to fit on top of Raspberry PI3 GPIO pins, with an RGB LED, a buzzer for user feedback, and a button to allow users to reset the game at will. (b) Prototype shows LED circuit, connect to Raspberry PI3 (not shown) is able to light up from commands from Raspberry PI linux; this control is not from input via wireless connection from user's laptop.

Due to misfortunate events of delayed shipping and a breakdown in one of our team member's laptop, testing the Raspberry PI3 (RPI3) for wireless communication was hindered one week prior to FPR presentation. The LED was supposed to light up, should the player obtain a point, via Bluetooth communication, so the user is free to move around the controller without the inconvenience of wires disturbing their movements. This Bluetooth communication should have allowed the RPI3 to read text files from the user's laptop, letting the circuit know to light up the LED. If there were two players, then the ArUco ID marker would assign each player a specific color, i.e. player 1 has green color, and player 2 has color blue. We were in fact able to establish a functional Bluetooth detection for the RPI3, see Appendix C for details.

D. Data Analysis

In order to have a successful gaming experience, coordination is vital to allow the game to know where the user is in playing the game. The game must respond to the user's actions as quickly as possible and send feedback, in real-time, to the display for the best interactions. To summarize what our algorithm does, the program extracts coordinates using ArUco and parses them to another txt file and sends to simulator in real time. Therefore, the simulator can read the markers location as where it is in present. In addition, when detection of markers is lost due to sudden quick movements from the user, the program will use saved data from last location to keep the positioning stable in the game simulation. This appears as a red dot on the display, we call this the cursor. Then simulator displays the result of where the computer thinks where the user is, as a red dot, on the table, which usually follows the position of the ArUco marker itself. The format of our coordinate file is shown in Figure 7.

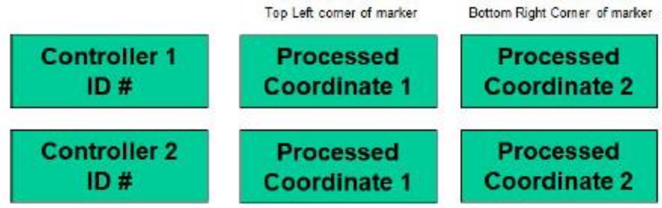


Fig. 7. Block diagram of logistics of coordinate data.

.

III. Project Management

III.I Project Deliverables

A list of our accomplishments for MDR is presented in Table 3. These tasks were no simple or easy jobs; teamwork was essential in getting these tasks done as best as we possibly could.

Table 2

FPR Deliverables	Status	Results
Functional camera detection	Completed	ArUco Markers
Two functional sensor inputs	Completed	2-player games
Detect interactions between user controller and simulated objects in real time	Completed	one button for game functions (varies for each type of game)
Display simulation of game on laptop	Completed	Stand is about 4 feet above table
PCB control commands outlined	Not Complete	Prototype on breadboard, wireless mouses used instead

Table 3. Table of MDR deliverables with status and next milestones listed next to each deliverable. The items listed in the "Next Steps" column are our final goals for the project.

We believe we have been able to meet most of our FPR deliverables, concerning the basic essentials of our project's functionality, which is the software side of our project, i.e. the ArUco detection and game production. The downside to this is we have failed to build a proper controller for users. Figure 7 shows our team's breakdown of responsibilities throughout the Spring 2019 semester. Coming out of CDR, it was apparent that our initial design for the PCB was inadequate and needed to be re-thought of, which was detrimental in the time constraint of ordering the PCBs and 3D printings.

We believe we have been able to meet our MDR deliverables, concerning the basic essentials of our project's functionality. The downside to this is we have yet to build the circuit that will be used as useful, user-friendly tool for players. The motion tracking mechanism that utilizes color detection is currently under revision. We aim to make this sensing mechanic work better to be error-free and more robust. However, alternative plans are being considered, such as shape tracking or hand detection. These alternative methods pose even greater risks however. The hand detection may be more difficult as users will constantly move their fingers in various ways, which can obscure how the camera will detect the user's hand. The shape tracking poses an inconvenience as there are inevitably many kinds of shapes that will appear on the board game, such as squares and circles for checkers, thus our controller cannot take the form of a simple shape that would be mistaken as an actual game piece.

The lighting issue poses troubles as well, if a room is too dark, the controller's red cloth appears as black and the camera will not detect anything, and too much light saturates the camera, thus rooms of only dim light are preferred. This makes having the projector on top of the filed where the camera will detect a potential problem. We intend to solve this problem with our PCB, by implementing photo-resistors to send feedback to the camera on the lighting of the room, so the camera can adjust its sensing at any time, for example if users want to play games at night or in the morning. This is why the ArUco markers were essential in accomplishing reliable tracking.

With our second functional sensing input we aim to add to our project, we should be able to allow users to play a greater variety of games that enables them to place objects arbitrarily. Our first sensing input, the ArUco detection, allows for users to move around within the game. If we are successful to implement a secondary input to be processed by the laptop, we can allow users to not only traverse the gaming field, but also have more interactions with the objects within the game. We envision some kind of switch that the user can press, which will correspond to turning on and off a light-emitting diode (LED) circuit that is safely and securely attached to the controller. The LED may send information in the form of infrared radiation (IR) which the camera can pick up and register the signal as the user holding an object. The user would have to continuously hold the switch to keep the IR LED circuit on, which corresponds to the user holding an object in the game, and the user would simply let go of the switch to let go of the object in the game. Specific duties for each team member is listed in Figure 8.

TASKS	RESPONSIBLE	START	END	DAYS	STATUS
Software Requirements	All	24-Jan	1-Mar	3	Completed
Hardware Requirements	All	24-Jan	1-Mar	36	Completed
Decide Products	All	24-Jan	1-Mar	36	Completed
Products ordering	All	20-Mar	27-Mar	7	Completed
Camera: ArUco ID	Bowen	21-Feb	28-Mar	35	Completed
Camera: Coordinate Output	Jooyoung	21-Feb	31-Mar	38	Completed
Simulation: Game Environment	Jooyoung	13-Feb	20-Mar	35	Completed
Simulation: Objects	Patrick	28-Feb	31-Mar	31	Completed
Simulation: Coordinate Reading & reaction	Patrick	20-Nov	26-Nov	6	Completed
PCB: Bluetooth communication	Pervesh	1-Apr	12-Apr	11	Completed
PCB: prototype, 3D printing	Pervesh	18-Mar	3-Apr	16	Incomplete
System Integrition	All	4-Apr	12-Apr	8	Completed
System Testing	All	13-Apr	21-Apr	8	Completed

Fig. 8. Gantt chart displayed with each group member's individual responsibilities. Chart shows how tasks have been split up during set time periods.

Our design for the PCB was delayed due to speculations from CDR reviews. We had a very ambitious goal of using Bluetooth components from the RPI3 in our PCB, but that proved too extraneous by our evaluators. A simpler design was proposed to us in bench-side meetings in the middle of March, in which we only needed sufficient components for user feedback, meaning our PCB was then used for simple LED circuits and the RPI3 can be another piece within our controller that communicates with the user's laptop wirelessly via Bluetooth. Having difficulties figuring out the design rules and logistics of how we wanted our PCB costed some time for shipping to receive our PCBs, however, we were able to have our finalized PCB delivered to us.

III.II Division of Labor and Means of Communication

The tasks are broken up by each team member's expertise. Bowen and Joo Young have been using their programming knowledge in C++ as well as expertise in image processing to create the video tracking program and the coordinate I/O. Patrick has been using his previous experiences in gaming to create the gaming simulations, as well as developing means to edit current games and produce new games. Pervesh, the electrical engineer major of the group, has been testing LEDs that are bright enough to be used as a secondary functional input, as well develop pin layout plans for the fabrication of the PCB.

The means of giving each other aid within the group stems from independent research. Each team member is assigned a task and then is free to execute that task as they see fit. Occasional issues arise within each group member's task, thus team meetings are set up over weekends to revise and resolve issues. Each team member is able to offer their own expertise in helping another team member's issue.

From MDR, for example, Pervesh has had troubles developing Arduino code to send data to the laptop, Patrick and Bowen were able to provide aid in developing code for the circuits' commands. Another case where Patrick and Bowen wanting to test which LEDs would suffice in motion tracking, Pervesh was able to create circuits to make sure the LEDs were biased properly to ensure current flow was in the correct direction with sufficient voltage supply.

We have designed preliminary commands and functions that the PCB will implement for our final product, we now must make greater efforts to have the PCB fabricated. We at least wish to have a prototype implemented on a bread board by CDR to demonstrate users can perform basic functions such as turning on and off the games.

The Gantt chart displayed in Figure 9 shows how each team member has contributed to the project.

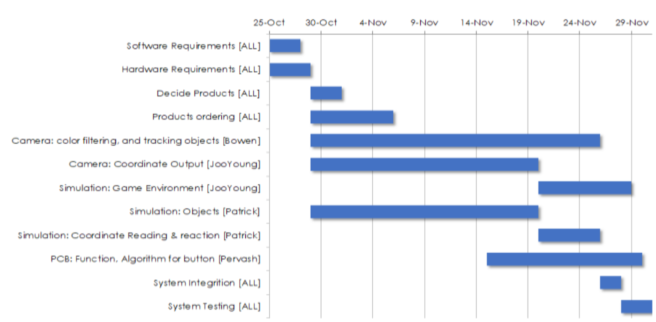


Fig. 9. Gantt chart displayed with each group member's individual responsibilities. Chart shows how tasks have been split up during set time periods.

We still carried out similar means of communication and aid for FPR, though may time-constraining issues had risen.

Concerning the PCB up to MDR, from our evaluators (Professor Lixin Goa and Professor Joseph Bardin) it was clear that we were not obtaining the correct idea for our controller due to misconceptions of the rules concerning raspberry pi and for user feedback. After our benchside meetings, we had then a clear idea as to what the PCB functions should be to make a usable controller for users, thus shows our team is able to adapt quickly to new ideas and input from the evaluators and ideas offered by the course director, Professor Hollot and Shira. However, at this point, there was little time to develop necessary components in time. Bowen and Joo Young were able to provide Pervesh with sample code to be able to get the RPI3 to operate with Bluetooth, and Patrick was able to provide feedback and help with the mechanics of the PCB layout designs. We were able to in fact get the RPI3 to operate on Bluetooth and create a functional circuit to light up an RGB LED, however by the time this was all done, the stand for the projector needed to be completed and the time necessary for a 3D-printed case for a controller would not been able to be completed in time for demo day.

Our two main methods of communications are a group text and multiple google docs. The group text allows us to always keep each other updated on how our tasks are going gives us an easy way to ask each other quick questions, and the google docs allow us to all access the information the other members have gathered and share things with each other that we think might be important information in the future.

With everyone being in different classes and working with complicated schedules, putting things in the google docs allows everyone to work on presentations and reports when the have the opportunity, instead of struggling to figure out when and where everyone can meet to work on things.

Aside from doing our own tasks individually on our own time, we meet regularly as a group to make sure all of the separate components still work and communicate correctly with each other. We also meet weekly with our advisor Professor Kelly to discuss the status of the project as a whole and to make sure timeline is still as close to our proposed schedule as possible.

IV. Conclusion

Since MDR, the ultimate success of Tablo was being able to track the controllers using the ArUco Markers. Tablo's robust mechanism makes it have great potential for many other applications, such as free-drawing for artwork, or be used to facilitate learning as a whiteboard for anyone draw digital images on a wall (simply a matter of mounting the camera and projector on their sides). The overall impression from Demo Day was that many families' children enjoyed playing the games we provided, which was exactly our intended audience and purpose for making this project. Though we were not able to create genuine controllers for our project, the wireless mouses inside plastic cases served users well for playing games effectively.

The tracking was essential for making the whole project work effectively, despite many variants of tracking such as: color tracking, hand motion, shape detection, we are glad to have finally make ArUco markers work. Due to many series of events, the PCB controller came up just short of being implemented for the final product. We had the essential circuitry and code corrected, the LED was able to light up on command from the RPI3 terminal, and we validated we could achieve Bluetooth communication with the RPI3, however, we were not able to integrate these features with the gaming program such that the RPI3 can

Appendix

A. Cost

Below is our comparison of the cost it took to develop our project, compared to the cost of each component's standard price. See table 4.

Part	Development Cost	Production Cost
PCB	$ 1.86	$ 1.40
Projector	$ 174.99	$ 104.99
Raspberry Pi3	Borrowed	$ 19.99
Wood Supplies	$ 94.99	$ 48.99
Total	$ 271.84	$ 175.37

Table 4. List of costs for each component of our project.

B. Design of Table Stand

Here you see our design for the table stand. As a product, the projector would be mounted on the ceiling of the user's home. However, for demonstrative purposes, we built a stand to support our projector, the display is dependent on how far from the table the projector is. We made the stand to be 3.67 feet, thus the display on the table will be 2.5 feet (30 inches) by 1.417 feet (17 inches). See Figure 10.

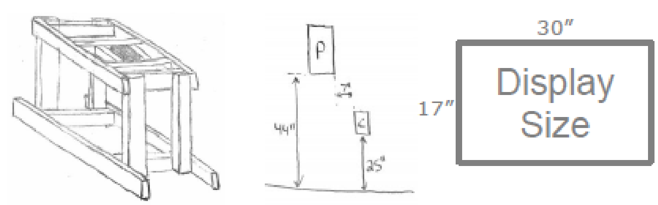


Fig. 10. The left most image shows the physical structure of our stand for the projector. The image in the middle shows a lateral view of the height above the table the projector, alongside the camera, will be. The image on the right shows the dimensions of the intended display size of the projector at a height of 44 inches.

C. Bluetooth Communication

We had to borrow a separate monitor from M5 to write commands to the RPI3 directly. We use Bluetooth for more reliable wireless connection, as opposed to WiFi, which is less secure and slower. See Figure 11 for validation of Bluetooth connection, and Figure 12 for terminal commands that also validate an established connection via Bluetooth.

Fig. 11. Image shows verification of Bluetooth communication from Raspberry PI3, we use one of our team member's smart phone to test if Bluetooth connection is possible. Image shows Bluetooth connection is established between the iPhone and Raspberry PI3, the device MDR-XB50BS is another monitor near our SDP lab bench.

Fig. 12. Linux shows the libraries imported aided in establishing Bluetooth connection. The address of our team member's device shows it is connected to the Raspberry PI3 in the last line.

REFERENCES

[1] Hutchcraft, Jack. *Vice*, "It's Official, Everyone: Board Games Are Cool Now," Gaming. 1^{st} March 2016. World Wide Web. 16^{th} December 2018.

[2] Belen, Darell. *SelfGrowth*, "How Board Games Affect Our Lives," Online Self Improvement Community. World Wide Web. 16^{th} December 2018.

[3] Rogerson, Melissa J., et al. "Digitising Boardgames: Issues and Tensions," Microsoft Research Centre for Social Natural User Interfaces University of Melbourne. 2015.

[4] "Electronic Components Datasheet Search," *alldatasheet*, ATMega328 Datasheet PDF, ATMEL Corporation. 24^{th} Novemeber 2018. World Wide Web.

[5] "OpenCY," *docs.opencv.org*, Detection of ArUco Markers.