Initial Project Proposal

Year: 2022 Semester: Fall Project Name: R.A.C.H.E.L.

Creation Date: April 25, 2022 Last Modified: July 24, 2022

Team Members (#1 is Team Leader):

Member 1: Jack Myers Email: [myers395@purdue.edu](mailto:myers395@purdue.edu)

Member 2: Bartosz Stoppel Email: [bstoppel@purdue.edu](mailto:bstoppel@purdue.edu)

Member 3: Micah Morefield Email: [mmorefie@purdue.edu](mailto:mmorefie@purdue.edu)

Member 4: James Hubbard Email: [hubbar31@purdue.edu](mailto:hubbar31@purdue.edu)

1.0 Description of Problem:

Table tennis is one of the most popular sports in the world, with more people learning to play every year. There were an estimated 300 million players in 2017 [1]. Although this sport draws such a large audience, the experience has remained largely unchanged for decades. Table tennis is played with two paddles and a table; score is kept verbally. You win if your opponent cannot return the ball back to your side. What if there was another way to play table tennis? Through fun new concepts/variations and advanced equipment, we intend to enhance convenience of scorekeeping, develop training tools, and establish exciting modifications to the hundred-year old game of table tennis. Though our project may not directly solve pressing social issues, recreation is an integral part of a healthy and balanced lifestyle. The epidemics of obesity and clinical depression can be alleviated in part by stimulating physical activity and recreation [2] such as we intend to provide in our final product.

2.0 Proposed Solution:

In short, our project consists of a projector, a depth camera for motion tracking, and an array of contact microphones to detect when the table is struck by a table tennis ball. Of course, one also needs a table, a pair of paddles, and a ball. By projecting graphics onto the table and taking input from various sensors, we hope to create new and exciting variations on the standard game of table tennis. Our tool would not only streamline the process of scorekeeping, it allows for visualization of game data, such as potentially showing post-game heat maps of the ball’s path on the table or fun game modes utilizing projected graphics, which adds a twist on the game itself. We plan to implement gamemodes wherein one must serve to the lit up section of the table, or avoid spots the ball had previously hit. By using an Xbox Kinect as the motion tracker, we would be able to utilize well-supported gesture control libraries to allow the user to adjust the score or select different game modes. The extent and complexity of these alternative game modes depends primarily on our creativity and the amount of time we have for implementation.

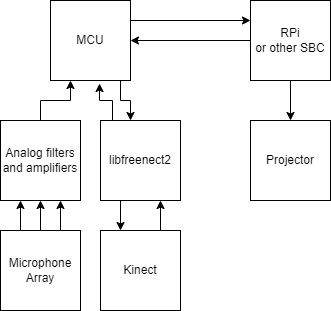
3.0 ECE477 Course Requirements Satisfaction

Every member of the team leads an active lifestyle, and we all enjoy playing table tennis. This personal interest in the project will help motivate our work and ensure that we develop a product which is fun for players of table tennis. Because our project is necessarily mounted on a table tennis table, it will be bulky and cumbersome in its final form. Nevertheless, we intend to package the processing components of our project in a neat and tidy housing, with minimal cables connecting to the projector and microphones. The physical construction inherent to our project is negligible, consisting of merely a mount for the camera and projector, as well as an enclosure. The final product will be attractive and presentable, despite its lack of portability.

3.1 Expected Microcontroller Responsibilities

In order to effectively accomplish the DSP tasks necessary for a working final product, the team needs to select a microcontroller which is suitable to act as both a system controller and a DSP. While the team has not yet settled on a specific chip, we know that the ideal microcontroller for our project has a large bit-width, a high rate of instruction execution, and floating-point capabilities. One example of such a microcontroller is the STM32F429NG. Alternatively, a chip from the STM32F427xx series (e.g. STM32F427VGT6) would be adequate and perhaps more readily available than chips in the F429xx series. Alternatively, the STM32F091RCT6 is widely available in the design lab and would prove to be an inexpensive and readily available microcontroller to use.

Alongside a microcontroller, our team intends to use a variety of peripherals including microphones, cameras, and a consumer-grade SBC. These peripherals will work in conjunction with the MCU to process video frames and triangulate the sound of table strikes, facilitating precise tracking of the ball and, consequently, accurate scorekeeping. In order to interface with the projector, DisplayPort and HDMI are both adequate. One advantage of DisplayPort is that one only needs to purchase a license if they intend to use the logo, unlike HDMI. The Kinect depth camera interfaces over USB3.0, and the contact microphones are analog devices. The interface between our MCU and SBC is undecided; candidates include UART, USB, or even Ethernet.



3.2 Expected Printed Circuit Responsibilities

Included on our custom PCB will be a MCU, integrated circuits to simplify the use of DisplayPort/USB/Ethernet/UART, a power supply, analog filters and amplifiers to condition the signals from our microphones, and buttons to toggle the power. Additionally, invisible within the housing will be a hardware reset button, a programming header for our MCU, and indicator LEDs for the purposes of debugging.

4.0 Market Analysis

The customer demographic we aim to attract includes those who want one specific thing, a new twist to a game they love. However, our solution comes with a few specific barriers to entry one, being cost. To get an idea of how much our product would cost we will first gather a few components and their price. Of course, our design is subject to change, and if this product were to be manufactured at scale there would be some inherent savings, however, this is still a useful method for finding a rough estimate. Every component the team intends to incorporate into the final product is reasonably priced and readily available. The most expensive component by far is the projector, which is still within a tractable price range. We anticipate difficulties in procuring RaspberryPi SBCs, but numerous third-parties manufacture drop-in replacement SBCs which would be perfectly suitable for our project. All dollar amounts are in USD. Prices do not take into account confounding variables such as supply chain problems.

* Kinect or equivalent motion tracking camera: $50
* Projector: $50 - $100
* Raspberry Pi: $35 - $75
* Microcontroller: $20
* Contact microphones: $20
* Additional Hardware: $30

From this rough outline, our product could cost up to $295 in parts to manufacture. Additionally, the user would need a preexisting table tennis table, which would cost an average $200. For the projector’s image to reach the entire table, it would need roughly 10 feet of clearance above, depending on the projector and the lenses used. Thus, the consumer considering purchasing our product would need about $500 and an indoor space of about 12x12x10 feet. This is a relatively steep barrier to entry and would most likely filter out those not serious about table tennis.

The number of people playing table tennis is about 300 million and some conservative estimates say it is as low as 87 million. Additionally, one statistic found that there were 3,873 professional players worldwide [3]. From this we can postulate that our consumer population is not very large, with maybe 4,000 guaranteed customers. Not everyone who plays table tennis will want our product, but if our product were to be iterated and improved to the point that our analysis of the game was truly novel, and offered a noticeable edge for those training professionally, new customers such as training facilities and coaches might decide that this product is a necessity. With enterprise and professional customers, the door is opened for added complexity and thus, higher profit margins per unit.

5.0 Competitive Analysis:

5.1 Preliminary Patent Analysis:

5.1.1 Game table television and projector system:

This patent describes a table for a table game (e.g. billiards, table tennis, foosball, etc.) with a projector overhead which allows moving images to be displayed on the table as a game is being played [4]. This patent is not expired, but because it specifies that the images displayed on the table are “not associated with the … table game”, our project would not infringe if we intended to commercialize.

5.1.2 Method for acquiring ball-hitting gesture and ball-hitting speed of table tennis robot racket:

This patent describes formulas used for calculating ball speed and direction of a table tennis ball after being hit by a robotic paddle [5].

5.1.3 Entertainment system providing dynamically augmented game surfaces for interactive fun and learning:

This patent nearly describes our proposed project - a projector system to heighten a tabletop game with object tracking and display [6]. It includes displaying fun graphics, such as flames trailing a bowling ball traveling down a lane, or an optimal shot in pool. This patent poses a problem in the case that we intended to commercialize our project, as we would certainly be infringing and the patent doesn’t expire until 2031.

5.2 Commercial Product Analysis:

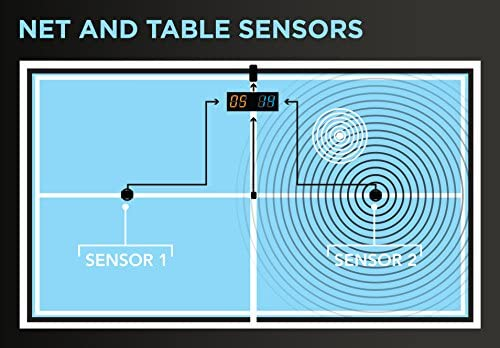
5.2.1 Eleven Table Tennis:

Eleven Table tennis is a virtual reality game available for the Meta Quest [7]. Eleven Table tennis aims at making the most realistic table tennis experience while also allowing for the flexibility of VR, such as not needing a physical table and playing with anyone across the world. And naturally, it automatically keeps score.

While this product is not similar whatsoever in implementation, it offers a fierce competitor in the end experience. Our product plans to enhance the experience, but our ability to do so is directly limited by the hardware we use to implement it. The game will still be played with paddles and the feedback/interaction is limited to what we can display on the table, and while this presents a vast amount of possibilities, it pales in comparison to what is accomplishable in a virtual environment.

While this product doesn’t particularly guide us with implementation details it does put into perspective the practicality of our product. Our product offers a unique experience, but consumers who want an increasingly in depth table tennis experience might turn to a virtual option due to things such as space requirement, cost and ease of use.

5.2.2 Stiga SensorScore:

The Stiga SensorScore is an automated scorekeeper for a ping pong table [8]. This tool comes with a net and two sensors that are attached underneath the table on either side. By using both sensors and input from the net, this device can accurately keep track of score. However there are edge cases where the SensorScore will falter, Instances such as the ball hitting the corner or edge of the table, will not be accurately accounted for by this devices, for this reason, a manual override is supplied in the form of a small panel attached to the side of the table, allowing the user to easily adjust the score if there is ever a discrepancy. This product accomplishes a core functionality that ours intends to, with a simplified design. In fact, incorporating sensors under the table, in conjunction with object tracking from a camera, could result in more accurate scorekeeping for our design.

If we choose to implement a similar technology as this product, calibration would be a key element to its success, which would rely on our ability to successfully capture and decipher the signal received from the contact sensors. Additionally, something this product allows for and a software feature we could easily implement, is a timeout. This is the amount of time after a bounce, that the device will wait before a score is tallied. Again this is a parameter that could be calibrated to the user’s preference.

5.2.3 Amicus Prime:

The amicus prime is a high end table tennis trainer complete with a tablet as a control interface. The purpose of the Amicus prime is to allow the user to create in depth practice routines and train individually on a standard table tennis table.

The tablet provided serves as the only interface between the automated ball server and the user. The tablet runs Android and connects using bluetooth. The server is capable of producing balls with a wide range of spin and direction. The Amicus prime has a clear target demographic: professional table tennis players who want to practice their skills and level up their game. Our proposed solution also targets professional players as well as casual ones.

When compared to our proposed solution, the two differ substantially. The Amicus prime is for a single user only, while our system is for two users at least. Additionally, the two differ on implementation as well. Firstly, our solution contains no moving parts, unlike the Amicus that implements three motors to achieve the desired serve. Our user interface also differs as we plan to present information using a projector and allow for input through hand detection or a physical panel.

Overall, these two products differ noticeably in the end experience, technical implementation, and target demographic, while both are still “enhancing table tennis”.

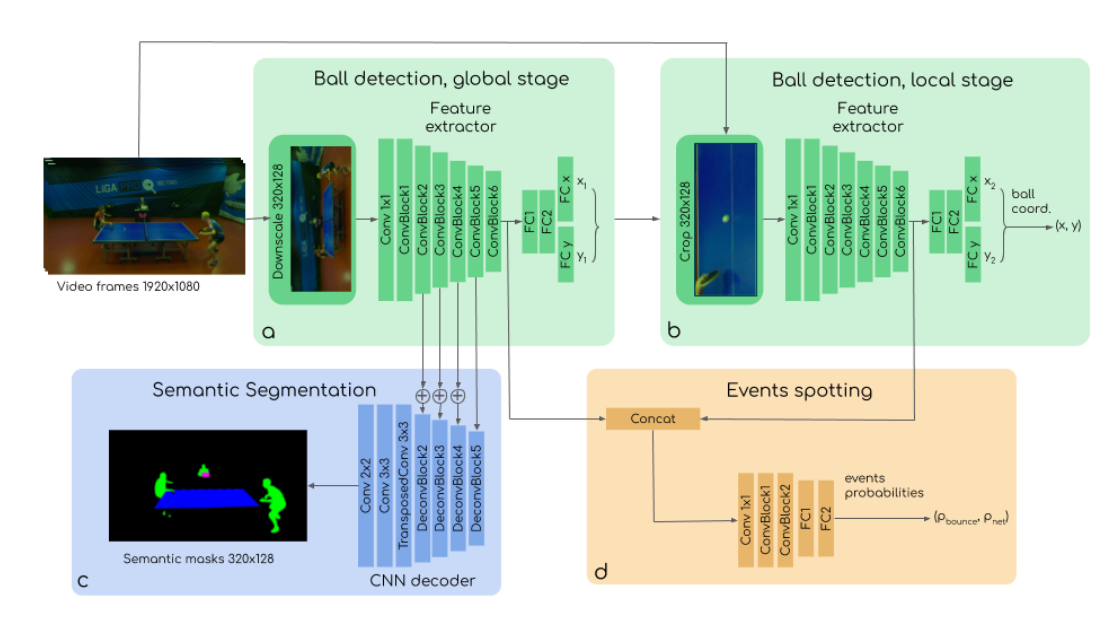
5.3 Open Source Project Analysis:

5.3.1 ODAS:

**O**pen embedde**D** **A**udition **S**ystem is a portable library designed to perform real-time sound source localization[10][11][12]. We plan to use contact microphones affixed to the bottom of the playing surface as one of the data streams for ball tracking. With an array of such microphones, not only can our project detect that the table was struck by a ball (as opposed to limbs, paddles, etc.), it will be able to triangulate the location of the strike and cross-check with the depth camera for improved accuracy. ODAS implements many of the techniques necessary for that task in a way which is portable to our MCU.

5.3.2 TTNet-PyTorch:

TTNet is a neural network designed to process table tennis games in real-time, using a single 120Hz video feed [13]. The purpose of the project is to help tournament organizers record game statistics quickly and accurately without the intervention of a human referee. A wide variety of data must be collected to accurately ascertain the state of the game, including the position of the table, the position of the ball, and the location of bounces. The primary benefit of this solution is its ability to be implemented with a small amount of inexpensive equipment. Whereas previous solutions required the use of several high-speed cameras to accurately track the trajectory of the ball, and computing power beyond what an average consumer can afford, TTNet requires only one camera at a frame rate and resolution achievable by many smartphones, and can process games in real time using only on consumer-grade GPU. Compared to our proposed solution, TTNet serves a slightly different purpose. Our goal is to extend the game of table tennis by projecting graphics onto the table, TTNet aims only to document the state of a standard table tennis game. TTNet also makes use of a different set of sensor inputs; instead of a depth camera and contact microphones, TTNet takes a standard color video feed as its only input. TTNet-PyTorch is an implementation of TTNet using the well-known machine learning library, PyTorch [14].



5.3.3 OpenKinect (libfreenect2):

libfreenect2 is a userspace driver for the second-generation of Kinect depth cameras which is compatible with Windows, Linux, and MacOS [15][16]. While there is an adequate proprietary API for the Kinect (known as “Kinect for Windows”), it is only available for Windows devices and is only distributed as a binary. The advantage of libfreenect2 over the proprietary driver is the ease with which it can be modified to suit our needs, as well as the elimination of legal hangups which would prevent our team from being able to distribute the driver alongside or even within our product without a paid license. libfreenect2 is distributed under the GNU General Public License, which allows for largely unfettered modification and redistribution.

6.0 Sources Cited:

[3] Statista Research Department. “Ranked Professional Tennis Players Worldwide by Gender 2017.” Statista. <https://www.statista.com/statistics/1125205/ranked-professional-tennis-players-gender/> (retrieved 07/24/22)

[4] “US10124240B2 - Game Table Television and Projector System, and Method for Same.” Google Patents. <https://patents.google.com/patent/US10124240B2/en> (retrieved 07/24/22)

[5] “CN102200760A - Method for Acquiring Ball-Hitting Gesture and Ball-Hitting Speed of Ping-Pong Robot Racket.” Google Patents. <https://patents.google.com/patent/CN102200760A/en> (retrieved 07/24/22)

[6] “US8292733B2 - Entertainment System Providing Dynamically Augmented Game Surfaces for Interactive Fun and Learning.” Google Patents. <https://patents.google.com/patent/US8292733B2/en> (retrieved 07/24/22)

[7] For Fun Labs. “Eleven Table Tennis”. Oculus. <https://www.oculus.com/experiences/quest/1995434190525828/> (retrieved 07/24/22)

[8] STIGA. “STIGA SensorScore Fully Automated Table Tennis Scorekeeper”. Amazon. <https://www.amazon.com/STIGA-SensorScore-Fully-Automated-Innovative-Technology/dp/B011UNAG40> (retrieved 07/24/22)

[9] Amicus. “AMICUS PRIME”. Butterfly. <https://en.butterfly.tt/amicus-prime.html> (retrieved 07/24/22)

[10] François Grondin, François Michaud. “Lightweight and Optimized Sound Source Localization and Tracking Methods for

Open and Closed Microphone Array Configurations.” arXiv. <https://arxiv.org/pdf/1812.00115.pdf> (retrieved 07/24/22)

[11] Grondin, François, et al. “ODAs: Open Embedded Audition System.” Frontiers. <https://doi.org/10.3389/frobt.2022.854444> (retrieved 07/24/22)

[12] Introlab. “Introlab/Odas: Odas: Open Embedded Audition System.” GitHub. <https://github.com/introlab/odas> (retrieved 07/24/22)

[13] Roman Voeikov, Nikolay Falaleev, Ruslan Baikulov. “TTNet: Real-time temporal and spatial video analysis of table tennis.” arXiv. <https://arxiv.org/pdf/2004.09927.pdf> (retrieved 07/24/22)

[14] Maudzung. “TTNet Real-time Analysis System for Table Tennis PyTorch.” GitHub. <https://github.com/maudzung/TTNet-Real-time-Analysis-System-for-Table-Tennis-Pytorch> (retrieved 07/24/22)

[15] OpenKinect. “OpenKinect/Libfreenect2: Open Source Drivers for the Kinect for Windows V2 Device.” GitHub. <https://github.com/OpenKinect/libfreenect2> (retrieved 07/24/22)

[16] Xiang, Lingzhu, et al. “Libfreenect2: Release 0.2.” Zenodo. <https://doi.org/10.5281/zenodo.50641> (retrieved 07/24/22)

Appendix 1: Concept Sketch

