

Bumper Car Sumo

ECE 3992

Spring 2019

Jaden Simon - simonjaden223@gmail.com

Melvin Bosnjak - meco0597@gmail.com

Daniel Humeniuk - d.humeniuk@utah.edu

<http://bumpercarsumo.weebly.com/>

Bumper Car Sumo

Jaden Simon, Melvin Bosnjak, Daniel Humeniuk, *Students, University of Utah*

Abstract—Americans are not having fun anymore. The weight of capitalism is forcing families to work 60 hour weeks for low pay, leading to decreased enjoyment and productivity. An affordable and efficient form of entertainment would help solve this issue. In order to promote this, our group is designing a bumper car battle royale game. The project will have intuitive controls, simple yet fun mechanics, and a robust interface to allow for easy development. Continued play testing during development will allow us to monitor these objectives. Our ultimate goal is mutual enjoyment between us and our players.

I. INTRODUCTION

THE most important criteria of our game's success are cost and entertainment efficiency. Our game must yield high levels of fun per minute played, while keeping costs as low as possible without compromising the entertainment value. Competition in games is a key element for high entertainment value [1]. Because of this, our team decided to create a game where player-controlled robots attempt to push each other out of an arena. To facilitate gameplay, our robots need to be easily pushed around by other robots. Controls should be easy to enhance playability, though not too easy. We need a way to detect game conditions, such as when a robot is out of bounds or when a player wins. This could be done with a human referee, but that would take away immersion of the game. Because some players may not have friends to play with, it is desirable to have an AI to play against.

Since cost is a contributing factor to our design, we will want to use existing technologies as much as possible. Robots can be constructed as a two-wheeled platform with wireless modules, similar to a Segway. They will be controlled through a web interface accessible by a wide variety of devices. Computer vision libraries will allow us to track the location of each robot if painted differently. Exact positioning solves the problem of detecting game conditions, plus giving us plenty of information to use for a basic AI.

Our goal, above all else, is that our game is fun. Playtesting will be a major component of our design process, rethinking aspects of the game if it is not fun.

Costs will be mitigated, but entertainment value will never be sacrificed for savings. Player enjoyment is our primary measure of success.

II. BACKGROUND

The biggest component to our project is the robot. Two-wheeled wireless robots already exist and are mass produced, such as the Ollie (Fig 1). While the Ollie has many of the features we require for our project, it does not give the level of control necessary to make our game a reality. On top of that, it is expensive at over \$80.00 per unit. The specifications for the Ollie are freely available online and will help guide us through the design of our own robots.



Fig. 1: The Ollie from Sphero
Source: Adapted from [2]

Tracking of the robots would be a huge undertaking, but luckily there exists an open-source computer vision library called OpenCV. OpenCV provides algorithms to scan an image and identify desired patterns [3]. With this, tracking can be done with only a camera, a computer, and a small bit of code.

III. DESIGN

To complete the project, we will need controllers, robots, tracking, and a main control mechanism. The controllers will be implemented as a web application, allowing for easy cross compatibility. The robots will be created with two DC motors that are connected to WiFi or Bluetooth and powered by a battery. An overhead camera will be used to monitor the locations of our robots. Each controller application will communicate with a centralized hub which will handle game mechanics and controlling the robots. Separating these four components allows for simultaneous development by our team members.

A. Hub

The hub will be a Raspberry Pi device that will have four primary functions: To capture the game state from a camera peripheral, use computer vision to keep track of each robots location, host a web server that will have bi-directional communication with the controllers, and wireless communication to relay the players commands to their respective robots. Fig. 2 shows the basic illustration of the game. If time allows, then our AI implementation will be done on the hub. This component of the project requires no custom hardware and can be done entirely in software. Core hub software will take at most a month for full development and testing.

B. Robot

Our robots will be cylindrically shaped with two wheels on each side of the body that can move in either direction (Fig. 3). In order to move forward, both of the wheels will spin at the same time at the same rate. To move backwards, the same principle applies in the opposite direction. Rotating the robot left or right requires spinning the motors in opposite directions. An ESP8266 microcontroller will provide WiFi and Bluetooth capability while also allowing for command interpretation from the hub. Our motors will require significant current so a custom driver will be required. Robots will require enough power to push each other around and so the torque of our motors is important. Additional components can be added to keep track of orientation or acceleration. Batteries of suitable capacity and voltage will be determined once the robot requirements are finalized. The robot will take up to 3 months to fully develop and test due to its requirement of custom hardware.

C. Controller

We want our game to be easily accessible by a wide range of players. A hardware controller would be easy to use, however, it significantly increases the cost of the project. Web applications can be run on numerous devices and are relatively easy to develop. Players would be able to access a web server on their own devices to control the robots. Our hub will act as the web server and process incoming commands, translating them into commands for the robots. The controller interface will be the arrow keys for a computer and buttons for a phone. Our web server may also send information about our robots back to the client, which may include orientation, speed, acceleration, or possibly a live camera feed. Basic user interface and web server software should only take two weeks.

D. Tracking

Tracking will be done by the hub using a camera and OpenCV. The casings for each robot will be designed in such a way to allow for easy identification by algorithms. Players who leave the camera's field of view will be considered eliminated by the hub, preventing control of the robot by the player. Since our camera will be static, the total play area is limited by our camera's height and field of view. Our play surface will be monochromatic which increases our tracking capability, though we should be able to track with an arbitrary surface. The time to perfect tracking is largely dependent on the hub and the robot, though 2 weeks seems like a reasonable amount of time.

IV. TIMELINE

A. Spring

Our primary goal for spring is to develop a simple prototype. We will utilize our embedded systems final project to develop and test our wireless communication with the ESP8266 to control motors. This should be enough to demonstrate basic wireless robot control.

B. Summer

The summer will consist of ordering and gathering the resources needed as well as detailed design for each component of the project. Basic documentation of our hardware and software interfaces will be completed by the end of the summer. The basic user controller interface will be completed. Our playing field may also be constructed at this time as it is not reliant on any specific component.

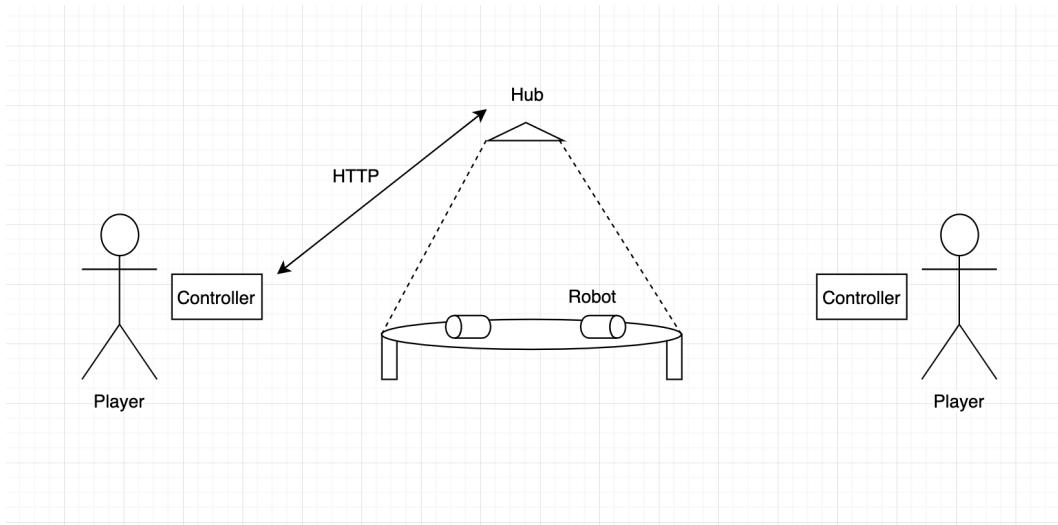


Fig. 2: Overall game design

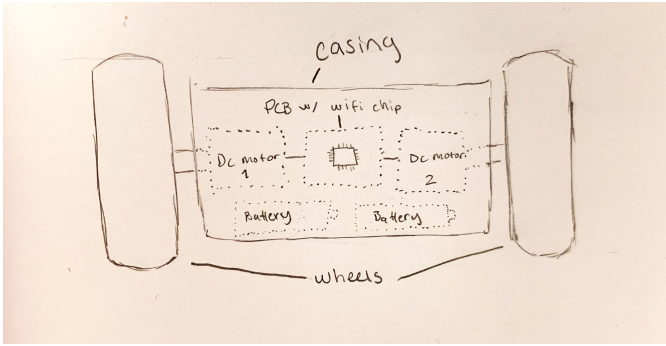


Fig. 3: Basic robot layout

C. Fall

Most of the legwork for the project will be done during the fall. With all components gathered and the design completed all that will be left to do is to assemble, test, and finalize documentation. Each project component will be worked on simultaneously. Integration between the four components of the project will not take long if our interfaces were designed correctly, perhaps only a week or two. The minimum requirements of the project is planned to be completed and tested a month before the demo day. A large amount of extra time allows us to experiment with additional features that will hopefully add to the entertainment value.

V. RESOURCES

Each robot will require 2 DC motors with wheels, several batteries, a PCB that includes an ESP8266 microcontroller and motor drivers for each DC motor, and a casing to enclose and protect the internals. The casing will be designed and 3d printed by our group.

Various ICs and small electrical components will be required based off our circuit designs. The design, testing, and implementation of the robot will take the bulk of the time required for this project.

A Raspberry Pi will act as the hub, handling all game mechanics and user-robot interactions. The Raspberry Pi 3 Model B+ has a quad core 1.4GHz processor which should be enough processing power for these tasks. Tracking is a fundamental component of the hub and will require a camera, ideally with high resolution and FPS. The options for the camera are the Raspberry Pi camera and the Logitech C922x Pro Stream Webcam. Software on the hub will depend on multiple open source libraries to function, though will still require significant custom software to operate.

The playing field will be constructed out of a simple table top with some 2 x 4's to hoist up the camera, with power and data cables running down to connect to the hub.

Because our controllers are entirely software based, users will need to bring their own device to interact with our demo. Most people carry a smart phone, so this should not be an issue. Basic controller software will not be a monumental task and so should take up very little time.

VI. TESTING

Much of our time spent testing will be focused on robot design and user experience. The most complicated part of our robot will be the integration of a battery, but once that is complete we will focus on debugging software communication protocols. In addition, play testing of the project is fundamental to our success. We must insure that our game is entertaining

through repeated testing of robot collisions and the user interface. Collisions with robots should feel powerful and the user interface should be responsive and free of frustration. Beyond that, we will need to experiment with the durability of our robots to make sure they do not break during play.

VII. PROJECT DEMO

On demo day, we hope to achieve a minimum requirement of 4 robots, basic user interface, and a single game mode. While 4 robots will provide enough competition to be entertaining, we ideally want 8 robots to allow more people to play. Basic user interface would only be options to control the robots in the 4 cardinal directions. More game modes will be simple to implement but add enormous entertainment value.

A. Setup

First players connect to our web server through a device of their choice. Connections made in previous rounds will be preserved. Players then choose an available robot to play as. Each selected robot must be placed in designated spots, which will have to be done by humans. Once the hub detects that all robots are in the play area, the game can begin.

B. Game Round

Each game round will depend on the game mode. In all of our game modes, players are eliminated if their robot leaves the play area. For a free-for-all, the game round ends when all but one player is eliminated. If a team mode is implemented, then a game round ends when only a single team remains. Game modes may have time limits, though this will depend on play testing during development.

C. Post Game

If we are able to meet our minimum requirements, then additional features will be added to aid user experience. This would include post game reports that contain information about the player such as eliminations, maximum speed, or the biggest hit taken. After any post game reports, our demo will start again at the setup.

REFERENCES

- [1] P. Vorderer, T. Hartmann, and C. Klimmt, "Explaining the enjoyment of playing video games: The role of competition," 01 2003.
- [2] Ollie. Accessed: March 6th, 2019. [Online]. Available: <https://www.sphero.com/ollie>
- [3] Opencv library. Accessed: March 6th, 2019. [Online]. Available: <https://opencv.org/>