Singleplayer table football

Description

Table football is a quick-paced game for 2 – 4 players. But what if you are alone and want to play? This project proposal would make that possible by creating a version where you play against a computer.

Figure 1: Table football table

The idea is that a program (probably executed on a laptop connected to the microcontroller) should be able to control one side of the game and that a (human) player controls the other side. That means the program should at least have the ability to rotate the handles and displace them, so two movement axis per pole must be considered.

Usually, a table football table is as big as a normal table and has 4 handles on each side (see *Figure 1*). But for the sake of this project's budget, the prototype should be based on a smaller version with only 2 handles on each side (seen in Figure 2). This will also reduce the complexity of the AI[1], which will potentially increase the reactiveness and calculation times (which is important for such a fast-paced game).

[1] I will use the term "AI" somewhat loosely, meaning the algorithm behind the computer controlled opponent.

Figure 2: Mini version which the prototype will be based on[2]

Some risks to consider when building the prototype are risks related to electronics and wiring (short circuit, unique ground level, etc.), as well as risks related to motors (back EMF, magnetic interference, blunt trauma) which are described in the course manual. Depending on the type of motors used, they may require LIPO batteries which come with their own set of risks (again described in the course manual). Additionally, the electronics should be protected from balls flying outside of the playing area, since they could destroy the whole circuitry.

Design Challenges

Besides the choice of materials and parts to use, there are a few choices that must be made in the planning phase.

An essential part of the prototype is the control of the rotation and displacement of two poles. This will most likely be done by using motors. The rotation can be achieved quite intuitively by placing motors behind the poles. But there are also alternatives that can be considered, for example a single motor and a belt could be used to rotate both poles at the same time. The displacement is less evident and requires more thinking and engineering.

[2] Src: https://images.thalia.media/00/-/de7d86e8f1a6450ab1dbabb0be00e400/prof-puzzle-tisch-fussball.jpeg

Maybe something can be done with a belt, or the displacement could also be kept the same for both poles. While designing this mechanism we have to account for friction between the poles and the table and carefully choose the motors such that they balance power and affordability.

The team also has to design a way to catch the balls behind the goals. This could be as simple as using a rectangular box behind, to catch the ball, but it can be refined by for example lowering the floor a bit so that the ball can't roll out again.

The software side of the project is also not trivial, since not only does it communicate with the microcontroller and the camera, but it also has to interpret the incoming image, predict where the ball will go and what actions must be taken and send those instructions to the microcontroller. The most critical factor in that process will be time, as the ball can move very fast, and the program should still be able to react. So, the main software challenges will be interpreting the image, deciding what action to take and doing so in a timely manner.

Another idea that is not essential but could optionally be realized is the addition of a score display using LEDs. The idea requires additionally two sensors for detecting when a ball enters a goal. A program can then update the LEDs. This idea is cool but might not be in the budget and it makes the main program more complex.

Parts & Cost Estimate

Here I try to get an upper bound to the cost. Of course this is a very rough estimate, but I try to rather account for a bit too much than not enough. I am heavily inspired in the choice of components by the Robopong mentioned below. Also, I calculate the cost for the basic version without the optional LED score display.

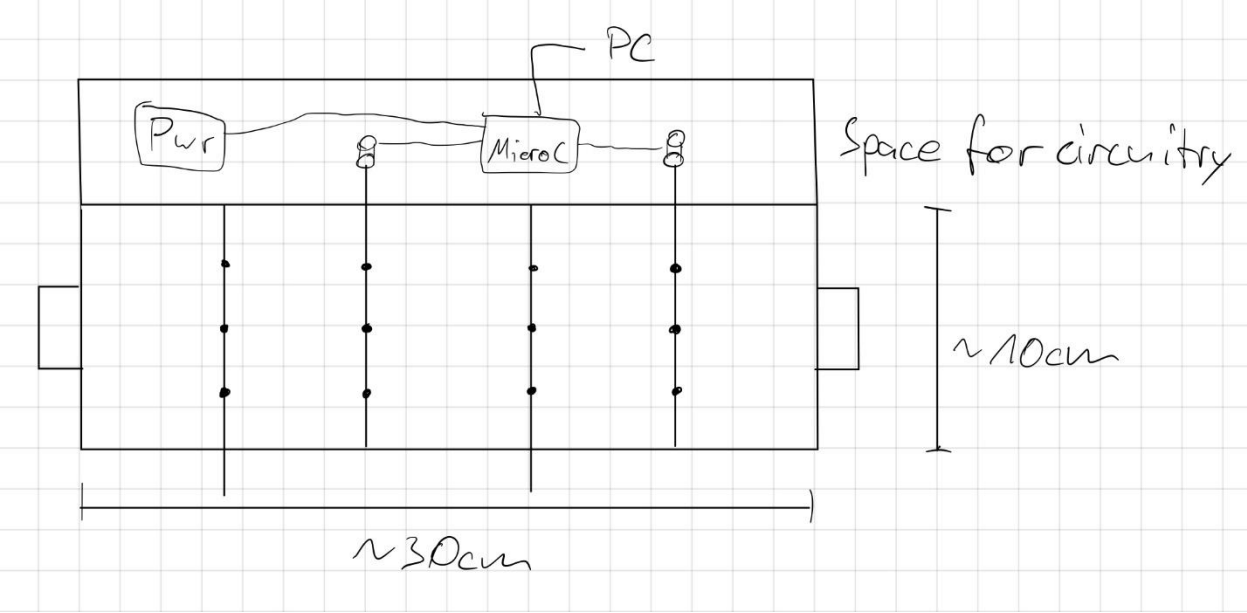

Figure 3: A very rough sketch of the layout and basic components

Based on the very rough sketch in Figure 3 we need not more than a square meter of material for the baseplate and the walls. Keep in mind that those measurements are probably a bit small (in the sketch) and might be changed later. Since that can be laser-cut, it is not a significant cost (less than 5 CHF if I remember correctly). For that and the poles I will calculate 50 CHF.

For the microcontroller I assume that an Arduino uno is enough, but I account for double the price just in case. That brings us to 10 CHF for microcontrollers.

Then we need an ESP32-CAM to give vision to our program, priced at 4 CHF.

We need up to 4 motors (each pole has to move in 2 ways), and we probably need similar specs to robopong. That totals to 4 * 15 = 60 CHF for four 17HS4401 motors and their drivers.

For the power supply I will assume the worst: That we need a LIPO battery which costs around 50 CHF tops online.

That totals the rough estimate at 174 CHF and considering that I might have forgotten some components, I round that up to 200 CHF for our upper bound of the cost.

Similar Projects / Useful Resources

Robopong

https://github.com/epfl-cs358/2024sp-robopong

Robopong is a CS358 project from last semester (Spring 2024) and shares certain aspects with this project. For example, it also has to deal with computer vision and has similar problems with choosing motors. Generally, it is a somewhat similar idea and has similar problems. Therefore, it can be useful to get inspiration from there.

Converting rotational motion to linear motion

https://www.youtube.com/watch?v=ve9M8d6KfdI

This video might be useful when figuring out how to do the two required movements. Since one movement is displacement (linear).

User Stories

The prototype should provide users with an engaging match of fast-paced table football. That means the bot[3] should not play predictably and the difficulty shouldn't be too easy. Naturally, the program or the prototype shouldn't break in any way and especially **be safe** (never endanger the user).

The idea is that the experience is as close to a normal match of table football as possible (e.g. same movement possible, poles not obstructed etc.).

If the LED score is implemented, the user should be able to see and reset the score.

Since table football is (relatively) well known is Switzerland, this prototype would be an attractive demo for all kinds of events (e.g. High Schools, etc.). It is engaging and connects a known game to the material learned at EPFL. I mean who doesn't like an interactive demo ?

[3] Computer controlled opponent

If done well, this project should also produce certain reusable artifacts, like a library for interpreting images or the design of the mechanism to move the poles (rotation and displacement). If a special protocol is used to communicate between the microcontroller and the program, that could also be reused.

Product Management

The prototype can be expanded in various ways. Although for future CS258 courses it would probably either require too many resources or not be complex enough. Possible follow up projects could be :

- Making two the algorithm play against itself and overhaul / optimize the algorithm
- Make it portable (kind of) by removing the dependency on a laptop or PC. That means make the algorithm run on a microcontroller or a chip on the board. This could possibly be achieved by a mixture of optimizing the algorithm and adding capable hardware.
- Make a version of the original size (*Figure 1*)

The first one is more a fun experiment but not commercially viable. The second and third one might have some viability and potentially a small market but since the idea of table football is to have fun with others, I think that there is not enough demand compared to the cost of production.

Annex

Rough Project Plan

The general steps that have to be realized for this project include (not final, not exhaustive, more a first idea of work to be done):

- Project Planning
    - Finalize project proposal
    - Order parts
    - Distribute work
    - ...
- Designing parts
    - Playing area, borders, protection of circuitry (wood or other base material, has to be sturdy enough)
    - Poles (wood, metal or similar)

    - Players (maybe 3d printed)
    - Ball
    - Apparatus which enables the microcontroller to move poles on his side **(significant portion of the work)**
- Laser cutting & 3D Printing
- Communication between camera and the laptop (via microcontroller ?)
- Control of poles via rotors
- Write main program and decision making (Maybe using ML)
- (Optional) Add an LED array that displays the points.
- Assembling everything
- Continuous testing and debugging

Sources

- https://www.youtube.com/watch?v=ve9M8d6KfdI
- https://github.com/epfl-cs358/2024sp-robopong
- https://images.thalia.media/00/-/de7d86e8f1a6450ab1dbabb0be00e400/prof-puzzle-tisch-fussball.jpeg
- https://www.orellfuessli.ch/shop/home/artikeldetails/A1069456633