CPSC 490 Senior Project Proposal
Brandon Liu, Eric Chien
Advisor: Lin Zhong

Gammix: Mixed-Reality Multiplayer Board Games

Introduction / Motivation

Many of us have played games like chess online. Especially through the pandemic, my family and I frequently played an online version of board game Settlers of Catan together through the website colonist.io. These online versions are fun, but lack the tactile experience of physically moving game pieces and seeing the board and gameplay live in 3-D space. But what if you could still play with others online, but using the real-life, physical board game in a straightforward way? We propose designing and building a system called Gammix for such mixed-reality multiplayer board game experiences.

We are inspired by the system built by engineer Shane Wighton that allowed him to play pool against other players online, but on an actual pool table.[1] It does so by projecting an interface onto the pool table, and by employing a custom built robotic pool cue that automatically lines up and makes the shot with the force and angle specified by the online player. While Wighton initially built the robotic pool cue and projector system as a way to guarantee sinking every shot via sophisticated software, the added benefit of enabling an online multiplayer experience on the physical table turned out to be the real highlight. The project left him wondering: “What about this game makes it so fun?” After seeing his system, I imagined a version of it but designed for and projected atop a Catan board, and I wondered the same thing: Would it be fun?

Project overview

To explore the question of what makes a mixed-reality game experience enjoyable, we propose a two part senior project. The first part consists of designing and building a minimally working prototype of Gammix, specifically to play checkers, on a physical board but against online players. We chose checkers as it is one of the simplest board games. This part will take place over the course of a month and a half, from mid-February to end March. Part one consists of refining the high level requirements detailed below into a full specification, designing each aspect of the overall system, then implementing them in software and hardware where needed. We plan to write our code in a public Github repository and release it open source. In the spirit of open source software, we will aim to architect our project in an extensible way, specifically for adding/modifying games, such that specifications for other popular games (like chess, or Settlers of Catan) may be added by ourselves or other contributors in the future in a relatively simple way. One example of such a consideration would be designing the software to be able to handle an arbitrary number of players (within reason), even though checkers is only a two person game.

[1] https://www.youtube.com/watch?v=vsTTXYxydOE

The second part of our senior project will consist of usability and user experience (UX) testing and interviews using our working prototype. This consists of formulating a full list of questions and a testing methodology, which we will create concurrently with our designs during part one and execute in part two. If needed, we may perform contextual inquiries and/or interviews with users playing board games in real life as well as their equivalents online to understand a general baseline experience. We would test our prototype in a location like the CEID on campus, separating the players in different rooms to simulate a distance. When designing our testing methodology, the overarching questions of “How easy is it to use and learn to use?” and “What can we learn from users’ experiences with this mixed reality game?” will guide us.

Deliverables

- Game board / Projected game interface
    - Add ArUco markers to detect specific features with computer vision code
        - This would include the corners of the board to enable accurate scaling and board state update
        - Also add to game specific pieces such as checkers and chess pieces (if needed) to be tracked by the camera relative to the board state
- Software / Phone app
    - Propose building an app using Flutter and the Dart programming language to allow for cross platform compatibility (iOS, Android), as well as built in access to the phone camera
    - Will connect over the internet to other players with the app so that the gameboards of both players will be synchronized to each other’s
    - Camera images will be processed using OpenCV ArUco detection code to state of the board, transmitting state updates to both players
    - Multiple games plan to be developed including checkers, chess, and potentially Catan
- Projector apparatus
    - A physical apparatus to mount the projector and phone, such that it can project the interface onto game board in real time as the game progresses
- Final written report
    - Detailing our design and implementation processes and outcomes with descriptions, diagrams, and/or photos
    - Sharing findings from our UX/usability testing

Part one estimated timeline / Subgoals

- Feb 21: Agree on a overall project specification/direction after meeting again with Prof. Zhong (0.5 week)
    - Also acquire/purchase any equipment such as projectors/cables if needed

- Feb 28: Finish an engineering design document for the Flutter application considering the relevant trade-offs, and start development (1 week)
    - App should be able to detect ArUco markers and connect to the projector
- March 7: Construct a working projector apparatus to hold the phone and projector over the game board using materials in the CEID or Architecture lab (1 week)
- March 14: Finish working detection of the board, and rending/scaling of the game board interface (1 week)
- Spring break
- March 28: Finish working code to detect game state and communicate it with a backend storing information about a game (1 week + spring break)

Additional Specifications Details

Game board / Projected game interface

The plan is to have a generic board of reasonable size with edges marked out with ArUco markers so that players with any sort of game board could theoretically integrate with our app without needing any specialized hardware (potentially, players could utilize any space of projectable size, without even a board). These ArUco markers delineate the playable space and would help initialize the play space. This initialization step will be programmed by us to analyze the area set by the ArUco markers by taking a picture with the camera and subsequently scaling the Gammix interface to the play board such that the projected interface is aligned with the actual board.

After initialization, our plan for implementing the rest of the board includes projecting a general grid to serve as the backbone of the analysis effort done by the camera component along with any other objects specified by the game. For the example of checkers or chess, after each user's move, the interface would project onto the board showing which their opponent had moved and to where, and then that player would update the board by picking up that piece and physically moving it to reflect that.

For relatively simple games, game pieces could be simply chips with ArUco markers attached to them with coding on the end of our detection algorithm to keep track of the location and state they are in relative to the master grid. For more complex games, more intricate methods of maintaining piece state info will likely need to be developed. In the case of checkers, with red and black interchangeable pieces, we anticipate simple color detection computer vision code will suffice.

Software / Phone app

The app serves as both the main software hub as well as interface for the user, actually rendering what will be projected onto the board. The idea for development of the app is to use Flutter, a cross platform development toolkit by Google, to build an app capable of connecting users across Wi-Fi into a Gammix game. The app would be responsible for synchronization of the game between two players in different locations. It will keep track of the local game state of the board of the player, periodically sending and receiving data from the other player's phone in

order to ensure that the board state is synchronized between actions from the two players. Since the games we currently intend to implement will be strictly turn based in nature, we will design a way to ensure that there is an order to the moves made, perhaps utilizing explicit movements/conditions to end or start a turn or to signal information to players.

Furthermore, the app will access the phones' camera to keep track of the player's board, taking photos in time intervals to be analyzed in OpenCV with ArUco markers. We wish to efficiently utilize the powerful processing capabilities of smart phones to achieve this goal. As the user interacts with the piece and makes their move, we can either detect when a move has been made automatically by calculating differences in captured images over a period of time, or simply have the player press a button when their turn is over for the MVP. By designing a general enough setup, ideally applicable to many different board games, we hope to create a system that will allow easy extensibility to even more varying and complicated games with unique rulesets.

Other issues that will have to be addressed include common issues that arise from utilizing camera based sensing such as the issue of hands or other limbs blocking ArUco markers or other parts of the board that need to be tracked.

Other design considerations

Some consideration will also be given to the nature of the board for our board games including size and length and the possibility of including other hardware pieces to extend the sensing capabilities of the project. Other considerations include the best way to position the camera relative to the projector and if the camera from a phone is sufficient or if another separate camera would be needed (including bridging a communication method between the phone and camera).