Exploring Tangible Interfaces in the Design of a Hybrid Tower Defense Game

Vivian Tsang
Katholieke Universiteit Leuven
Leuven, Belgium

Jonathan Valgaeren
Katholieke Universiteit Leuven
Leuven, Belgium

Quinten Vandenbergh
Katholieke Universiteit Leuven
Leuven, Belgium

Sarah Geilenkirchen
Katholieke Universiteit Leuven
Leuven, Belgium

Robin Jorissen
Katholieke Universiteit Leuven
Leuven, Belgium

Abstract

This paper presents the design of a hybrid tabletop tower defense game called *Termite Terminator*. Following the popular strategy game format of the Tower Defense genre, players are challenged to protect a treasured picnic from waves of digital insect enemies with the use of physical tower models infused with electronics. Each tower type features distinct behaviors and interactions, along with additional tools to aid players in achieving their objective. During development, a special emphasis is placed on smooth interaction between the two interfaces through responsive feedback for the player. This work explores the use of tangible interfaces in games to enhance immersion while also providing a novel game experience.

Keywords

Tangible Interface, Tower Defense, Interaction Design

ACM Reference Format:
Vivian Tsang, Jonathan Valgaeren, Quinten Vandenbergh, Sarah Geilenkirchen, and Robin Jorissen. 2025. Exploring Tangible Interfaces in the Design of a Hybrid Tower Defense Game. In *Proceedings of (Master of Electronics and ICT Engineering Technology).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

Tangible User Interfaces (TUI) were introduced by Hiroshi Ishii and Brygg Ulmer with the aim to augment the real physical world by coupling digital information to everyday physical objects and environments [5]. Since their introduction, it has emerged in different fields to bridge the gap between the digital and the real world. With the rise of TUI's, also the research field of Tangible Interaction (TI) became prominent, which describes the set of related research and design approaches [4].

TUI's are no unknown concept within the video game world, with game controllers as the prime example. Over the years, other innovative controllers such as the Wii Remote, the cardboard-based Nintendo Labo series, the PlayStation Move, and the Switch Ring-Con have showcased the creative possibilities of tangible interfaces. Beyond mainstream controllers, also arcade games have employed different kinds of unique controllers that are tailored to a specific game, which have also shown to be successful. Entertainment systems are ever-changing, with TUI's holding significant potential to provide novel gaming mechanics and experiences.

While the novelty of TUI-based controls within games has been shown to be more fun to the player, these controls can also be less intuitive compared to their digital GUI-based counterparts [2]. As controllers act as the mediating layer to provide the link between the player and the game world, its properties also directly affect the player's immersion. If the controller proves to be less intuitive, it holds the player back from being able to fully engage in the gameplay [1]. An intuitive design for the tangibles should therefore be one of the main focuses to further optimize the gaming experience.

One effective approach to improving intuitiveness is through a strong natural mapping. Natural mapping refers to how closely actions within a game resemble the action in the real world. Controllers with a high perceived naturalness result in increased player immersion and enjoyment [9].

With the development of *Termite Terminator*, a hybrid tabletop tower defense game, we want to explore the potential of TUI-based games to increase fun and provide novel, immersive gaming experiences, while keeping intuitive design through naturalness in mind. In this game, the player attempts to defend a digital world against virtual enemies by placing physical towers on a screen or using hand gestures to cast spells. This hybrid setup will result in the user encountering different kinds of interactions in the game world and in reality through the tangibles.

The decision to develop a tower defense game, besides its popularity as a genre, naturally fits this TUI-based approach. With a top-down perspective and enemies advancing towards a goal, the act of placing a physical tower to defend feels natural and intuitive. The immediate feedback from this interaction, whether through visual cues or sounds, enhances the sense of interactivity and reinforces the connection between the player and the game world.

Master of Electronics and ICT Engineering Technology, KULeuven, GroepT

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Besides placing towers, also other mechanics are implemented to allow for more engaging gameplay. Examples include boosting towers' attacks through physical actions, watching a tower physically fall over as it dies, and attempting to repair it by wielding a *healing hammer*. The mechanics are again designed with intuitiveness in mind. Lastly, casting spells using hand gestures in such a way that the area framed by your hands instantly correspond to the spell-casting zone on the screen is another example of a natural interaction with the game environment.

The design and development process of this game is outlined in this paper and progresses in multiple stages, starting from an initial Wizard of Oz prototype and ending with a fully functional version. A detailed overview of the game is given for each iteration, alongside insights from player feedback and other observed behaviors during play-testing sessions.

2 Related Work

Tangible user interfaces allow for a different play style and dynamic between players, as shown by Xie [12]. Tabletop games with a tangible aspect allow children to play together if given enough space, where in a traditional computer game they would have to take turns. Children also found it easier to interact with the game by directly manipulating the tangible game elements than by using the computer mouse.

Most research has focused more on the intuitiveness rather than the fun of tangible game alternatives. According to Campbell [2], users found a tangible tower defense style game more fun, but less intuitive than a traditional GUI based game. The users also performed better in the traditional game. However, as the most common reason for preferring the tangible version was novelty, the researchers do wonder whether users will still enjoy the game after this wears off.

Speelpenning [10] created a tabletop game with multi-touch and tangible artifacts and came to a conclusion similar to Campbell's. They found that users preferred to interact with the game through the tangibles rather than the multi-touch display, despite the tangibles' limitations in their implementation.

In conclusion, the above-mentioned papers found themselves in a trade-off between intuitiveness and fun. Hence, an aim of our paper and project is to create a tower defense-style game that users prefer both in terms of experience or fun, without sacrificing the intuitiveness.

To achieve this goal, previous research will be used as a foundation. Notably, it is essential to include visual feedback in the tabletop system and beyond it, on the tangibles itself. In a paper on subjective user experience with tangibles and a tabletop interface, a higher subjective user experience and preference score was found when visual feedback was present [3]. Visual feedback was additionally preferred over auditory and vibrotactile feedback.

Furthermore, the impact of integrating hand gesture mechanics within TUI-based games can be explored. This has previously been implemented in different games and has been shown to offer natural interaction with an acceptable response time [13]. But more importantly, these would add a new layer of interaction on top of the tangible towers, and depending on the implementation add to the fun-factor as it is 'more diverse and challenging.'

Lastly, based on most of the papers mentioned in this section, we predict that implementing tangible towers will have a positive effect on the interest of players. However, this is often due to curiosity and the feeling of novelty, and as such will wear off over time. So, it is critical that the core of the game is enjoyable, without relying on the novelty of the tangible aspect. For this we will draw inspiration from the 'attractive factors of tower defense games' as explored by Zhang [14].

3 Design and Development

3.1 Core Elements

The development of *Termite Terminator* aims to explore the potential of TUI's in video games to provide a novel, fun gaming experience. Different kinds of tangibles and interactions are experimented with to achieve this. The tangible component is translated into physical towers and other tools requiring different types of interactions from the player. Other interactions include hand gestures detected by a camera to achieve certain actions within the game.

3.2 Requirement Analysis

To ensure that the game is intuitive and fun to play, a list of requirements have been summarized.

In order to bridge the gap between the digital game environment and the tangible towers: (1) A top-down perspective of the gameplay should be provided on a tabletop, in order to achieve the illusion of placing physical towers in the game. (2) The position of these physical towers should be correctly mapped to their digital environment. (3) Immediate feedback should be given to the player through visual or auditory cues after an interaction occurs.

The tangibles should additionally be engaging and add to the gameplay experience: (4) The physical towers should each have their unique interactions to make the gameplay more interesting. (5) Interactions with other tools such as the healing hammer and hand gestures to cast spells should provide a natural and clear link between the player and the game. (6) Towers that have been placed down should be noticed by the game environment and invoke a reaction, such as activating the tower to attack or being attacked. (7) Interactions should not only be visualized in the game environment, but also in real life. For example, a tower should communicate its current health condition to the player through some physical movement.

As for the gameplay itself, (8) typical game design principles should be applied to make the game appealing and fun for the player. By

also taking into account the attractive factors of a Tower Defense game [14], the following factors should be implemented:

- **(8.1) Use Strategy To Win (Sense Of Control & Sense Of Enjoyment):** Players should be able to strategize through freedom in tower placement.
- **(8.2) High degree of playability (Sense Of Challenge):** Variety of different tower interactions, behaviors and levels should be implemented.

Additionally, (9) a leaderboard should be implemented to further promote a sense of challenge by including competition between players.

3.3 Game Design: Rules and Mechanics

This section outlines the different tangibles and mechanics in *Termite Terminator*.

The game is based on a tower defense game. A tower defense game is a strategy game where players place defensive structures (towers) to stop waves of enemies from reaching a target. Towers attack, and players manage resources to build, upgrade, or reposition them to counter different enemy types. The goal is to survive all waves without letting any enemies pass.

Towers.

Rail Gun Tower: A tower that shoots a powerful beam through all enemies on its path. The player has to manually trigger the tower to shoot and it features a ten second cooldown between shots.

Lighthouse Tower: A tower that constantly targets one enemy at a time with a beam that can be boosted when the player interacts with the tower. The interaction in this case consists of shining a light at the tower to energize it.

Trigger Tower: A tower that fires projectiles at an enemy at a steady rate, but allows the player to manually trigger additional shots, enabling an increased firing rate for greater impact.

Healing Hammer. As the towers can take damage and eventually die, the *Healing Hammer* provides a way for the player to repair broken towers to restore their functionality. Simply holding the hammer over one of the towers will slowly increase the health points. Shaking the hammer, as if you were hitting the towers to fix them, increases the rate at which the hammer heals the tower.

Spells. The freeze spell will stop all enemies within the affected area for a set amount of time. To activate the spell, the user holds their thumbs and index fingers in a triangle to frame the area where they want to summon the effect. The spell has a cool-down period before it can be used again.

Enemies. The game features a diverse range of enemies, each presenting unique challenges. Basic enemies are the easiest to defeat, serving as an introduction to the game. Tank enemies, on the other hand, have much higher health, requiring more effort and strategy to overcome. Some enemies are capable of damaging towers, posing a significant threat to the player's defenses. As an enemy gets damaged, it demotes to its weaker version. All of this combined will add an extra layer of strategy to the gameplay. The different types of enemies can be recognized by their distinct looks.

Figure 1: The different enemies

3.4 System Design

3.4.1 Camera-based Positioning. A first step in the design process of the setup is choosing the way the game will be displayed as briefly touched upon in section 3.2. The two essential components to discuss are the display and the positioning. There are three plausible options for the display: a projector can be top-mounted and projecting on a tabletop, a bottom-mounted projector displaying on a see-through material or lastly a large television can be laid horizontally on a table. We have chosen the last approach, so the game can be played without the need for a specialized structure holding the projector and allowing for a more modular setup.

As for the positioning of the towers, we initially considered using a grid detection system using NFC-technology to detect towers being placed within certain slots of said grid. However, this would create physical clutter, obscuring parts of the actual game and limit to possible locations of the tower. Hence, we landed upon a camera-based system as this would mean minimal clutter on the screen itself while allowing grid-less positioning of the towers on the map.

To detect the towers with a camera, there are again a few options. While a color-based detection system would preserve the look of the tangible objects, the underlying TV might interfere with the detection or the game will be limited in what colors can be used. If we were to use a shape-based detection system, the detection could be hindered due to deformation of the camera lens if the camera is not positioned perfectly above the tower it is detecting. We settled on using fiducial markers to detect the bounds of the screen and the towers within it. More specifically, we are using ArUco markers and detecting them using OpenCV [8] as implemented in KhairulIzwan's library available on GitHub [6].

Since we are using a camera to detect the towers on the play field, it allows for flexibility in adding extra features such as casting spells through hand gestures. By placing hands in a specific way, special effects will be caused in the game world and affect the enemies. To detect the hands, and more specifically in what position they are, another open GitHub repository is used [7].

Using three markers on the corners of the game screen, the absolute coordinates coming from the video feed of the camera are translated into a relative axis system. The relative system scales from zero to one with the origin of the system being the bottom left corner of the TV, with (1,0) and (0,1) being the top left and bottom right corners

respectively. This scalar enables the data to be easily translated to pixel values to correctly position the towers and spells within the Unity game.

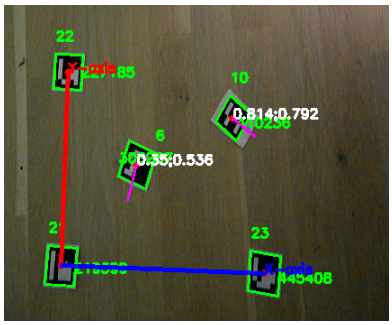

Figure 2: Marker Detection OpenCV

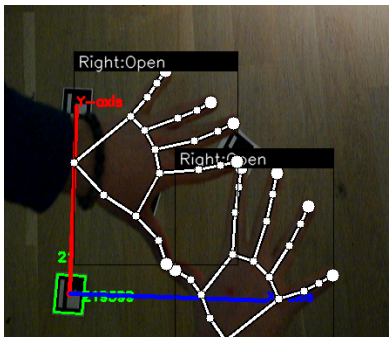

Figure 3: Hand Detection with OpenCV

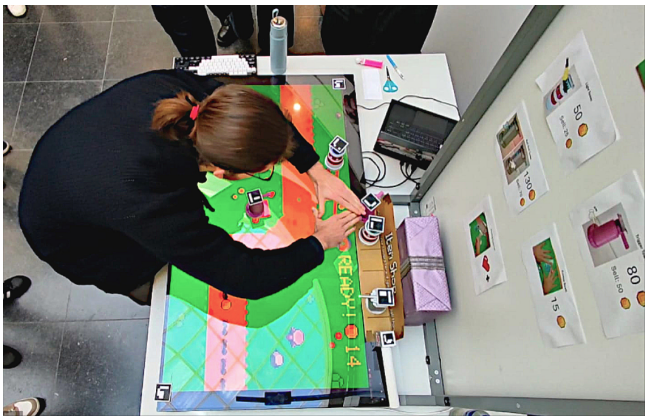

Figure 4: Camera vision is used to detect the fiducials and hand gestures

3.4.2 Hardware design. The game features three unique physical towers, each with distinct mechanics that enhance the interactive experience: the Rail Gun Tower, the Light Tower, and the Trigger Tower. These towers rely on tangible player interactions to operate and contribute to the dynamic gameplay.

The Rail Gun Tower requires players to charge and aim it at enemies before firing. It features an OLED screen that displays the charge level and an RGB LED that transitions from red to green to indicate when the tower is ready. Players press a button to fire once the tower is fully charged.

The Light Tower is boosted by shining a light onto it. It uses a light-dependent resistor (LDR) to detect changes in light intensity, allowing players to boost its performance with a flashlight or similar light source.

The Trigger Tower is the simplest of the three, firing at a steady rate and requiring players to rapidly press a button to increase its firing rate. Its straightforward design emphasizes quick reflexes and speed.

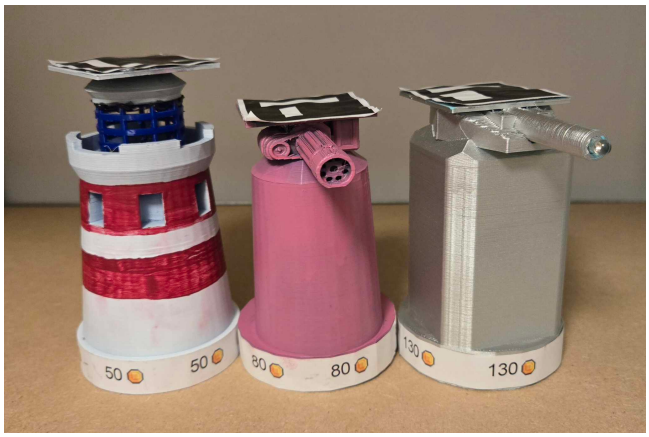

Figure 5: Final Tower Designs: Light Tower, Trigger Tower and Rail Gun Tower respectively

When any of the towers are destroyed, they physically fall over thanks to a servo motor in their base. Players can repair damaged towers using the *Healing Hammer*, a specialized hammer equipped with an accelerometer that detects shaking. Shaking the hammer near a destroyed tower restores its health faster and brings it back into play.

All towers are controlled using an ESP8266 micro-controller, which handles the sensors and actuators. A 9V battery powers the ESP8266, with a buck converter reducing the voltage to 5V. Communication between the towers and the main computer is facilitated using UDP (User Datagram Protocol). UDP is a lightweight, fast networking protocol that, while potentially less reliable than TCP, suits the game's design without issues. A table can be found in appendix 8 that shows the different IP's used and the messages that are sent and received.

Figure 6: The Healing Hammer

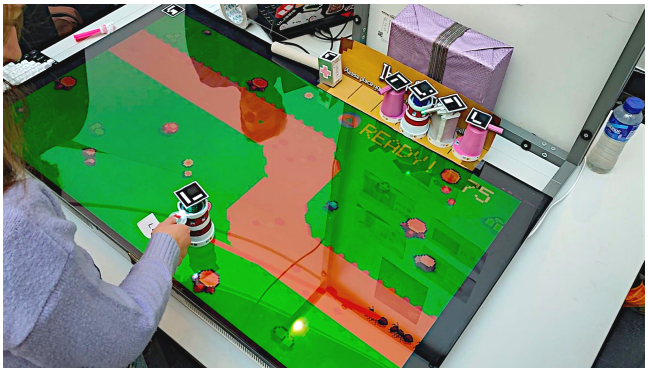

Figure 7: Light Tower is boosted by shining a light into it

3.4.3 *Unity Development.*

Programming Architecture. The Unity game engine is used for development of the game. As the game has multiple people working on different features in parallel, an emphasis needs to be placed on writing modular and extendable code to ensure consistency throughout the game. This is achieved by following object-oriented programming principles such as inheritance for the Tower and Enemy classes, as well as design patterns such as singletons for controller scripts.

Unity also facilitates this need for consistency with features such as the creation of prefab variants and animator override controllers, allowing to easily create new Enemy models with their own unique sprites. Also, the concept of ScriptableObjects, useful data containers, allows for easy creation of new enemy waves or shop items. Lastly, Unity Events allows for connections to be established between different scripts to achieve decoupling of code.

Feedback. To make the game more intuitive and responsive, various animations and sound effects have been incorporated. Each tower has its own unique sound effect when it fires, providing clear and engaging audible feedback. Background music plays continuously to ensure there are no moments of complete silence, helping to maintain the player's focus and immersion.

When an enemy is destroyed, a distinct sound effect is triggered, reinforcing the sense of accomplishment. Similarly, when the player loses, a specific sound plays to signal the end of the game.

Interactive animations enhance the player experience as well. For example, when a player picks up a tower, an animation shows that it cannot be placed on enemy paths, offering visual clarity. Additionally, animations accompany transactions, such as earning or spending money, making these actions more gratifying and noticeable.

Figure 8: A dirt cloud appears when a tower is placed (visual feedback)

Communication. To provide a connection between the towers, python-based camera scripts and the Unity project, the User Datagram Protocol (UDP) is used to communicate over a WiFi-network. The exact packages sent by the ESP8266's (towers) and python (fiducial and gesture tracking) are shown in appendix 8.

These UDP packages are received by a specified thread within the game and then analyzed depending on their size and sender. The 'Communication Controller' then directs the packages to their respective files to further process the information.

Figure 9: Image of the final game look

4 Iterative Development

4.1 Wizard of Oz Test

At this stage the game was a minimalist version. The TV displays a simple map over which very basic enemies attempt to make their way to the other side. The player is able to place down cardboard towers. Behind the scenes, a virtual tower is placed approximately where the player put it, in order to simulate the real interaction. This tower would automatically start shooting at the enemies passing by. An experimental shop system was also included in this test, where placing towers cost money and killing enemies would earn money.

While the reception of the concept itself was very positive, in general, more interaction was absolutely necessary. With the level of

interaction we had in our prototype, the end result would be a boring game. More interaction, but also more strategy is essential to keep the player entertained.

4.1.1 Strategy. Within the prototype created, there was no strategic element. The gameplay was limited to waiting for coins and buying towers. In reality, the player should not simply wait for coins, but while doing so should be thinking about what their next move should be, and how this would impact the game.

To achieve this, different towers should benefit from being placed in different positions, a specific orientation, scenarios or strategies in general. An example could be implementing a weather system that influences certain towers. Another one is involving different enemies that can focus on certain towers. Both of these examples force the player to think about what towers are being used at any given moment.

It is not required that there is an interaction between the player and the towers all the time. However, when the player is not physically active, they should be mentally active, thinking about strategy.

4.1.2 Influence of Actions. While the feedback given to the player to make the game more interactive was already part of the planned out development, further ideas were given to make the game more challenging for the player. For example, when implementing smaller potions or spells such as fire or ice spells, these shouldn't only influence the enemies. If these effects also apply to the player's towers, it will add a new dimension to where and when these can be used. This will encourage the player to think and strategize.

4.1.3 Coin System. It became evident within this short test session, that the buying and selling of towers and abilities needs some revision. When giving players no initial instructions, it was unclear to them what 'coins' were and what they were for. Furthermore, the system of selling towers has to be made fool proof. What happens when covering the fiducial, is it considered sold? Can a tower be lifted and moved? What if towers are stacked, covering each other's fiducials?

Figure 10: Cardboard towers are used in the Wizard of Oz prototype

4.2 Low-Fidelity Integrated Prototype

In this test, the connection between the camera-based tower tracking and Unity had been made. The software would now automatically place the towers where the user placed them. The rest of the setup was comparable to the Wizard of Oz prototype.

However, in this prototype, the fiducial trackers would be simple pieces of paper laid flat on the screen. During later testing we noticed that if the marker is placed on top of a tower, there is a noticeable shift due to the angle of the camera. This was partially fixed by integrating a calibration feature, carefully set before each use. But as this did not prove sufficient, it became required to mount the camera nearly perpendicular above the center of the screen, to minimize the parallax error.

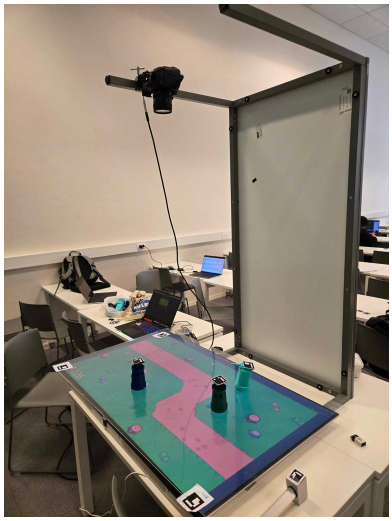

Figure 11: Perpendicular Camera Mounting

4.3 Interactive Prototype

4.3.1 Refinement of Core Aspects. During this stage of testing it was clear that the core technical aspects of the game are nearly complete. However, there is still a big challenge in refining the software to ensure the game remains enjoyable for the players. For the next few phases in the design process, we will have to use an iterative approach, emphasizing testing and adaptability.

Additionally, the game elements should be parameterized, so everything can be adjusted quickly when needed to. This creates flexibility in the game and will optimize the iterative design process.

4.3.2 Visual and Auditory Feedback. At this point the game does not have any sounds or special effects. The 'juiciness' of the game can be increased by adding these sounds and visual effects. This must be done in a way that the targeted sound effects reinforce players actions and visual effects so that actions of the game feel impactful and rewarding.

Visual effects can be a shaking effect of the screen or enemies having a dying animation, but also smoke or sparks under a tower if it is low on health. Small background animations can also bring more life to the game. Sound effects can be implemented when shooting guns or hitting an enemy. However, we must look out for not overdoing it. Focusing on stimulation but without treading into overstimulation is key.

4.4 Fully Functional Prototype

In the first interactive prototype we noticed some things we could change or implement to enhance the game. The first level should act as a tutorial, teaching basics like tower placement and enemy handling. Clear visual cues are a must, for example to indicate that towers cannot be placed on enemy paths.

Next, it became clear that the game lacked overall cohesion. A simple story, such as defending your lunch from invading ants, can tie gameplay to a relatable theme. Again, we noticed that feedback is key: sound effects should confirm tower purchases and signal when special abilities are ready.

New mechanics, such as tower-damaging enemies and collaborative tower effects, can add depth to the gameplay. These features can make the game feel more intuitive, engaging, and rewarding. On the other hand, this also acts upon the need for the player to be conscious about their moves, as discussed earlier in this chapter. More specifically, these features will add complexity to the user's decisions.

Furthermore, we noticed that users found the tower design unintuitive, especially when the button of the Trigger and Rail Gun Tower would be occluded by the tower itself. This left most players confused as to how to enable the towers. When they did find the button on the Railgun tower, they would be confused why it would not work back to back, as the display showing the charge level was also not visible from their viewpoint. A way to solve this is to provide a different visual feedback visible from all sides or at least the opposite side from the screen on the tower, with an LED as an example. A second issue was that the towers were not easily distinguishable. Testers found it difficult to identify each type of tower, as they weren't color-coded to their functionality. This was resolved by spray-painting each type of tower in a different color.

5 Play testing

5.1 First Wave of User Tests

During the final weeks of development, several features had been made but were not yet integrated with each other. So, for the first test with unfamiliar players, not all features were enabled. To be more precise, the playable game consisted of two fully interactive towers out of the three designed and a single wave of a handful of enemies.

The game was set up in a public area, allowing anyone interested to join and play. Around 10 people participated, most of them first-year engineering students. During this play test, we first and foremost wanted to analyze the intuitiveness. To investigate this, we purposely left out any instructions in the beginning, encouraging players to figure out how everything worked on their own. This immediately showed a drastic contrast to our own gameplay, with nearly every player unable to activate the towers before enemies reached the other side of the playing field and ended the game. However, briefly after explaining the functionality of the game and the towers, the reception of the game became rather positive. The players were pleasantly surprised with the two-way interaction between the towers and the game.

From this first wave of user testing, we were able to conclude that while the tangibility of the towers and the interaction between the towers and the game brought a new and exciting level of interaction and fun, the intuitiveness of the towers proved to be our number one obstacle preventing players from experiencing the fun. Furthermore, we also noticed that certain changes within the game went unnoticed by the player, for example a tower being boosted or the freeze spell costing money.

So, it is clear that our game requires so-called feed forward mechanisms as explored in the "Interaction Frogger" framework [11]. These are clear indicators of possible interaction points. Highlighting them can can guide users before they act, resulting in a more intuitive and frustration-free gameplay experience. Our specific situation calls for including visual or auditory cues to explain the features of all tangible components, as well as to notify the player of changes.

5.2 Second Wave of User Tests

5.2.1 Visual and Auditory Feedback. A first attempt was made for a feed forward system to aid in the lack of intuitiveness of our towers. Each time a new type of tower was placed in the game, a text would appear on the player's side of the tower explaining that specific tower. Some hints also included arrows that point to where the player had to interact with the tower by, for example, pressing a button. The pop-up would stay there for several seconds, giving the player time to read it.

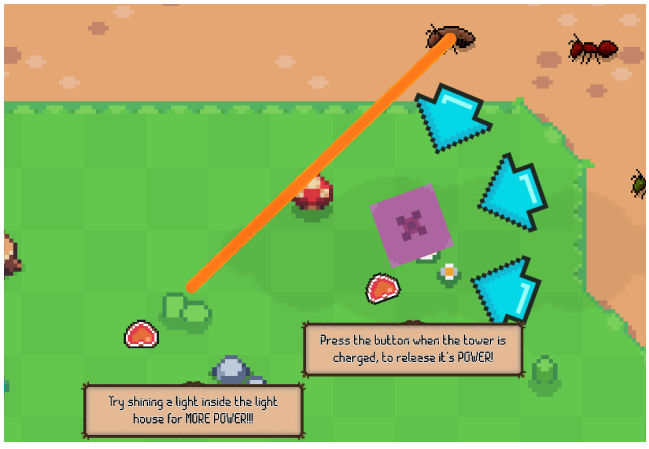

Figure 12: Implementation of Hints

The implementation of these hints was largely ineffective. Although some players completely missed the texts, most players now understood that they had to press a button to activate the Trigger Tower, but could not figure out what type of button they were looking for. Some tried pressing the fiducial marker on the top of the tower, others tried tapping the screen underneath the tower, all of them completely missing the small push button that the large arrows were pointing to.

Furthermore, stronger enemies now appeared vastly different from basic ones, appearing in different colors but also with hats. After damaging these stronger enemies, they would first lose the hat, and slowly demote into its weaker versions. This change drastically increased the clarity of the feedback on enemy strength and remaining health.

Lastly, since the previous test, other smaller sound effects were also included to notify the player of certain actions. These subtle changes helped show the player what actions were possible and which weren't.

5.2.2 Difficulty and Timing. As the last test only included a singular wave with a limited set of enemies, the players were quickly finished. To change this tech demo to a game, we implemented five waves with an increasing number of and increasingly stronger enemies. These were tuned throughout the testing, to give the player a rewarding feeling of progression without making the game too easy or too difficult too quickly.

5.3 Final Prototype and Public Demo

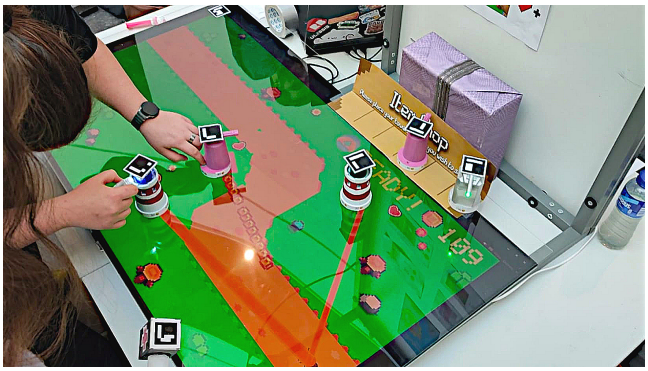

Figure 13: Termite Terminator at the public demo

In the final prototype, the game was fully functional, giving players the opportunity to try the three towers (Rail Gun, Lighthouse, and Trigger Tower), an interactive freezing spell, and the healing hammer. The unique mechanics of each tower allowed players to plot a strategy to overcome the different waves.

5.3.1 Intuitiveness. There is definitely a certain learning curve to the gameplay, before players had experience with all towers and could understand them enough to start this strategizing. However, a significant improvement in the user understanding of the game could be found after only a few play sessions. The addition and adjustment of the hints helped with the initial confusion, allowing the players to quickly understand the workings of each new tower, which gave them a benefit in the following attempts. More specifically, the in-game arrows were now reduced to one arrow in combination with the arrows drawn on the physical towers. Throughout the game, visual and auditory feedback pushes the player in the right direction and gives them a quick understanding of the enemy situation.

Feedback from testers did highlight that even though the initial learning phase requires a lot of effort, the game's mechanics became more intuitive over time. This created a satisfying progression experience, as the players would be able to reach waves beyond the one they failed at previously. This emphasized that more introductory levels in future iterations could prove useful in introducing new players into the game. However, in addition to this, we conclude that we were able to increase the much-needed intuitiveness of our game with new players with the strategies mentioned above.

5.3.2 Experience. Once players understood the game's mechanics and how to use the towers effectively, their immersion and enjoyment became evident visually and auditory as the game and players became more hectic with each new wave. The physical controls of the towers allowed for satisfying and precise shots, as well as hilarious actions whenever players would panic due to an enemy attempting to destroy the tower. The chaotic nature of the final game was one of its greatest strengths, significantly enhancing the overall fun and excitement

In particular, elements such as the freezing spell and the railgun tower were well received by many. They added an additional layer of strategy to the gameplay, allowing the player to line up enemies and get rid of a large group of enemies in one go, or use them to save the game barely on time. In general, the game felt exciting and rewarding and was enjoyed by the majority of the players.

Another notable observation was that, despite being designed as a single-player game, its increasing difficulty in later levels naturally encouraged players to collaborate and strategize. Players often divided roles, with one focusing on attacking enemies by interacting with the towers, while the other took charge of healing. During the final stages of a level, both players worked together intensively, boosting all towers in a coordinated effort to overcome the challenge and achieve victory.

6 Discussion

During this paper, the central question was as follows: How can tangible interfaces be used to enhance the player experience in a hybrid tabletop tower defense game, without sacrificing intuitiveness? To address this question, we examine the challenges we faced in designing the project and explore opportunities and limitations for future improvements.

6.1 Challenges

During the development and testing phases, several major obstacles were identified. One significant issue was the confusion of the user

when interacting with the different tangible towers, in particular the Rail Gun and Trigger towers. Hidden points of interaction and unclear feedback clearly disrupted the user experience. Drawing on the "Interaction Frogger" framework [11], this confusion can be linked to a lack of effective coupling between user actions and the function of the tower. To address this, inherent feedback (such as tactile or visual clues directly tied to the action) and augmented feedback (such as LED's or animations to indicate readiness) should be integrated. Additionally, feed forward mechanisms, such as clear indicators of interaction points can guide users before they act. By enhancing these aspects, we can create a more intuitive and frustration-free gameplay experience.

Another obstacle was the lack of strategic depth in the gameplay. Although the core mechanics were functional, players quickly expressed a desire for more dynamic and strategic elements. This issue can be partly contributed to the interaction design, more notably the insufficient augmented and functional feed forward and feedback played a vital role in the player's difficulty in understanding the coin system and overall game goals. More distinct or even obvious feedback aids in signaling the impact of earning or spending coins as an example. On the other hand, adding feed forward helps users in anticipating the consequences and even possibilities of their strategic decisions. In reference to the earlier obstacle, an improvement in these interaction design elements could enhance both gameplay intuitiveness and strategic depth.

Visual and auditory feedback clearly plays a crucial role in enhancing immersive gameplay. However, some feedback elements, while clear to use as designers, appeared unclear to the user and led to immersion-breaking confusion. This was especially noticeable with the button of the Trigger Tower. The button was small, and even though it was sufficiently highlighted on the screen with multiple virtual arrows to it, users still found it difficult to notice and interact with effectively. The same issue presented itself when the path was highlighted to indicate no towers could be placed on the path. Users would still attempt to place the towers, leading to more user confusion.

These issues clarify that there is a need for improved augmented feedback. This can be done through bigger buttons or more prominent markers or even by adding more sounds that guide users' actions or make the constraints of the game more clear. If these specific shortcomings are addressed, this will ensure that the feedback can more effectively communicate the essential information. As a result, this will reduce user frustration and enhance the overall experience. This also underlines the importance of refining the feedback mechanisms to balance the intuitiveness and engagement throughout the development process.

6.2 Future Work and Limitations

Any future work or similar implementation should focus on addressing the above-mentioned challenges. While implementing similar changes as we did with *Termite Terminator* might result in a similar uplift of intuitiveness, this is far from a guide. Each application must go through iterative design and user testing and adjust the mechanisms to that specific situation. However, this would still rely heavily on enhancing the visual and audio feedback, as seen in this paper, to improve the player experience by adding more intuitive feedback mechanisms. A good example of the complexity, is including separate animations or sounds for each tower. Careful fine-tuning was required to make certain actions noticeable, while others were more towards the background of the gameplay.

Another topic that needs to be focused on is the refinement of the strategy. This can go from including deeper strategic elements such as varied tower upgrades or more enemy types or even new map layouts. In addition, clearer instructions can be included as tutorials to ensure that players understand the core game mechanics and the goals of the game, without forcing them to lose a few games until they have explored all features.

By applying these adjustments to *Termite Terminator* or an alternative hybrid tabletop game, with a focus on clarity, strategic depth, and balanced feedback, the tangible interface will enhance the overall user experience.

7 Conclusion

In this study, the exploration of the integration of tangible interfaces in a hybrid tabletop tower defense game, *Termite Terminator*, was done. With iterative design and development, it was demonstrated how tangible interactions could enhance player immersion and engagement in gaming. By incorporating physical towers alongside interactive spells and tools, a unique gameplay experience was provided that bridged the gap between digital and physical realms.

During development, several challenges were revealed. These were particularly present in the interaction design, strategic depth, and most notably, feedback mechanisms. These challenges were addressed with improvements to feedback systems, better feed forward design, and enhanced gameplay strategies. This highlighted the potential for tangible interfaces to create a more intuitive and immersive gaming experience, when correct and sufficient information is given to the user. With user play-testing, these findings were validated and showed that players adapted quickly to the game mechanics and even found satisfaction in the progression they made in the game, despite the initial learning curve.

For future iterations, the focus should lie on refining strategic depth, improving the user feedback clarity, and making the game less cluttered for new players by giving information to the user at a well-paced tempo. If these areas were to be addressed, the potential of tangible interfaces in gaming can be further realized and pave the way for new innovative hybrid games that effectively combine the benefits of physical and digital interactions.

Acknowledgments

First and foremost, we would like to specially thank our coaches Jeroen Wauters and Nianmei Zhou, providing continuous feedback and ideas during this project and bringing our game to the next level. Furthermore, we would like to give a special thanks to Robbe Decapmaker for aiding in the design of the PCB and Pieter Nagels

for providing his 3D-printer to make some of the towers, and give helpful advice during game design. Lastly, a special thanks to all the play-testers and their feedback.

References

[1] Johan Blomberg. 2018. The semiotics of the game controller. *Game Studies* 18, 2 (2018), 311–323.

[2] John Campbell and Xharmagne Carandang. 2012. Comparing Graphical and Tangible User Interfaces for a Tower Defense Game. *AMCIS 2012 Proceedings. 11.* (2012).

[3] Jan Erp, Alexander Toet, Koos Meijer, Joris Janssen, and Arnoud Jong. 2015. Subjective User Experience and Performance with Active Tangibles on a Tabletop Interface, Vol. 9189. https://doi.org/10.1007/978-3-319-20804-6_20

[4] Eva Hornecker. [n. d.]. Tangible interaction. https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/tangible-interaction

[5] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '97)*. Association for Computing Machinery, New York, NY, USA, 234–241. https://doi.org/10.1145/258549.258715

[6] KhairulIzwan. 2020. Aruco-markers-with-opencv-and-python. https://github.com/KhairulIzwan/ArUco-markers-with-OpenCV-and-Python/tree/main

[7] Kinivi, Kazuhito00, and nkise nlab. 2021. hand-gesture-recognition-mediapipe. https://github.com/kinivi/hand-gesture-recognition-mediapipe

[8] OpenCV. 2024. https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html

[9] Paul Skalski, Ron Tamborini, Ashleigh Shelton, Michael Buncher, and Pete Lindmark. 2011. Mapping the road to fun: Natural video game controllers, presence, and game enjoyment. *New Media & Society* 13, 2 (2011), 224–242. https://doi.org/10.1177/1461444810370949 arXiv:https://doi.org/10.1177/1461444810370949

[10] Tess Speelpenning, Alissa N Antle, Tanja Doering, and Elise Van Den Hoven. 2011. Exploring how tangible tools enable collaboration in a multi-touch tabletop game. In *Human-Computer Interaction–INTERACT 2011: 13th IFIP TC 13 International Conference, Lisbon, Portugal, September 5-9, 2011, Proceedings, Part II 13*. Springer, 605–621.

[11] S. A. G. Wensveen, J. P. Djajadiningrat, and C. J. Overbeeke. 2004. Interaction frogger: a design framework to couple action and function through feedback and feedforward. In *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques* (Cambridge, MA, USA) *(DIS '04)*. Association for Computing Machinery, New York, NY, USA, 177–184. https://doi.org/10.1145/1013115.1013140

[12] Lesley Xie, Alissa N Antle, and Nima Motamedi. 2008. Are tangibles more fun? Comparing children's enjoyment and engagement using physical, graphical and tangible user interfaces. In *Proceedings of the 2nd international conference on Tangible and embedded interaction*. 191–198.

[13] Chi-Ho Yeung, Man-Wa Lam, Hong-Ching Chan, and Oscar C. Au. 2008. Vision-Based Hand Gesture Interactions for Large LCD-TV Display Tabletop Systems. In *Advances in Multimedia Information Processing - PCM 2008*, Yueh-Min Ray Huang, Changsheng Xu, Kuo-Sheng Cheng, Jar-Ferr Kevin Yang, M. N. S. Swamy, Shipeng Li, and Jen-Wen Ding (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 89–98.

[14] Baoyi Zhang. 2018. Exploring the Attractive Factors of Mobile Tower Defense Games. https://doi.org/10.2991/icassee-18.2018.107

8 Communication Information

Tower Type	IP Addresses	Send	Receive
Light	192.168.24.1, .2	"b" to boost	"k" to kill, "r" to repair
Trigger	192.168.24.3, .4	"b" to boost	"k" to kill, "r" to repair
Railgun	192.168.24.5, .6	"b" to boost	"k" to kill, "r" to repair
Hammer	192.168.24.7	"hr" when shaking	/

Table 1: Communication Mapping for Towers and Hammer