Exploring Tangible Interfaces in the Design of a Hyrbid 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.

CCS Concepts

• **Do Not Use This Code → Generate the Correct Terms for Your Paper**; *Generate the Correct Terms for Your Paper*; Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

Keywords

Tangible Interface, Tower Defense, Interaction Design

ACM Reference Format:
Vivian Tsang, Jonathan Valgaeren, Quinten Vandenbergh, Sarah Geilenkirchen, and Robin Jorissen. 2018. Exploring Tangible Interfaces in the Design of a Hyrbid Tower Defense Game. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX).* ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

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 then, it has emerged in different fields to bridge the gap between the digital and the real world. With the rise of TUI's, also Tangible Interaction (TI), which describes the set of related research and design approaches[4], became prominent as a research area.

TUI's are no unknown concept within video games, with a video game controller as the prime example. Also other innovative types of controllers have been developed throughout the years, such as the Wii Controller, cardboard designs from the Nintendo Labo series, the Playstation Move and the Switch Ring-Con controller, to name a few.

Entertainment systems are ever-changing, with TUI's holding significant potential to provide novel gaming mechanics and experiences. This work explores this potential through the development of a hybrid tabletop tower defense game *Termite Terminator.* The player interacts with a digital game displayed on a screen by placing physical towers and using hand gestures to defend against virtual enemies, while encountering different kinds of interactions in-game as well as in real life.

The design and development process of this game is outlined in this paper, along with the different iterations and collected feedback from players.

2 Related Work

Tangible user interfaces allow for a different play style and dynamic between players, as shown by Xie [9]. 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 tangible game after the novelty wears of.

Speelpenning [8] created a tabletop game with both multi-touch and tangible artifacts and came to a similar conclusion as Campbell. They found that users preferred interacting 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

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

paper and project is to create a tower defense style game that users prefer both in terms of intuitiveness and fun.

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. [3] Furthermore, hand gestures can be explored as these offer natural interaction with an acceptable response time. [10] Lastly, the game itself should be enjoyable and will thus follow the attractive factors of tower defense games as explored by Zhang. [11]

3 Design and Development

3.1 Core Elements

The development of Termite Terminators 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

As we want our tower defense-style game with tangible towers be intuitive to play, we must bridge the gap between the game and the physical objects as much as possible. The illusion of placing the tangibles in the game is achieved by displaying the game world as a top-down view of it on a tabletop. Next, it is a must to be able to position these physical towers and translate their position in the game world, which will be further discussed in section 3.4 Camera-based Positioning.

Secondly, because we want to increase the interaction between the players and the game, compared to traditional digital tower defense games, the towers will be tangible and interactive. How we achieve different ways of interaction and the different tower designs are discussed in section 3.6 Hardware design. Also hand gesture detection is implemented as a way to make the interaction between the player and the game more interesting.

Besides interactions between the player and tangible towers, also interaction between the towers and game is required. Towers that have been placed must be noticed by the game environment, and interact with it accordingly. This is translated into attacking the enemies, as will be displayed on the screen, but also responding to attacks, by physically falling down when a tower "breaks".

As for the game itself, typical game design principles will 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 [11], the following factors are implemented:

- **Use Strategy To Win (Sense Of Control & Sense Of Enjoyment):** Strategize with freedom in tower placement
- **High degree of playability (Sense Of Challenge):** Variety of different tower interactions and behaviors
- **Suitable difficulty (Sense Of Urgency):** Waves get progressively more difficult, and dynamically adjusts its difficulty

Lastly, for a good user experience, visual and auditory feedback is required. An animation or sound playing when a tower is successfully placed on the screen communicates this action to the player. Also other auditory and visual cues are implemented to convey certain actions or states of the tower.

3.3 Game Design

3.3.1 Rules and Mechanics. This section outlines the different tangibles and mechanics in Termite Terminators. A more detailed description of every mechanic can be found in appendix A

Towers.

Rail Gun Tower : A strong tower that shoots through many enemies but can shoot only every 10 seconds.

Lighthouse Tower : A tower that constantly targets one enemy at the time with a beam that can be boosted when the player interacts with the tower.

Trigger Tower : A tower that shoots every time that player interacts with the tower.

Healing Hammer. As the towers can take damage and eventually die, the 'Healing Hammer' provides a way for the player to repair broken towers and then increase the health of towers. 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, will boost the amount of health being increased.

Spells. The freeze spell will stop all enemies within the affected area for a set amount of time (three seconds). To activate the spell the user holds their thumbs and index fingers in a triangle where they want to summon the effect and the effect has to be off cooldown.

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. Finally, there are type-specific enemies that can only be harmed by certain towers, adding an extra layer of strategy to the gameplay.

3.4 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 considerable options for the display: a projector can be top mounted and project 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 latter, 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. 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 can not be flawless due to deformation if the camera is not perfectly above the tower it is detecting. To conclude, we settle on using fiducial markers to detect the screen and the towers within it, more specifically we are using ArUco markers and detecting them using OpenCV [1] as implemented in KhairulIzwan's library available on GitHub [6].

Since we are using a camera to detect the towers on the play field, this allows for flexibility in adding extra features. The extra feature in question is 'spell casting'. By placing hands in a specific way, special effects will be caused in the game world and effecting the enemies. To detect hands and more specifically in what position they are, another open GitHub repository is used [7].

Using three markers on the corners of said TV, the absolute coordinates coming from the video feed of the camera is 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.

3.5 Visual 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 satisfying and noticeable.

3.6 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 powered 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 lamp or similar light source.

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

When any of the towers are destroyed, they physically collapse, thanks to a servo motor at their base. Players can repair damaged towers using a specialized hammer equipped with an accelerometer that detects shaking. Shaking the hammer near a destroyed tower restores its health 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 less reliable than TCP, suits the game's design. To ensure critical messages are received, they are sent multiple times, mitigating UDP's lack of delivery guarantees. In the appendix can a table be found that shows the different IP's used and the messages that are send and received (Communication Mapping for Towers and Hammer).

3.7 Unity Development

3.7.1 The blueprint: 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 classes, as well as design patterns such as singletons for the controller scripts.

Unity also facilitates this need for consistency with features such as the creation of prefab variants and animator override controllers. Also the concept of ScriptableObjects, useful data containers, allow 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.

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 B Communication Information.

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.

4 Results

4.1 Wizard of Oz Testing

At this stage the game was a minimalistic 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 cardboard towers. Behind the scenes, a virtual tower is placed approximately where the player put it. This tower would automatically start shooting at the enemies passing by. An experimental shop system was also including 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 put the player to thought.

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?

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 test.

However, in this test 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 is important to mount the camera nearly perpendicular above the center of the screen, to minimize the parallax error.

4.3 Interaction Prototype

4.3.1 Refinement of Core Aspects. During the 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 consistently 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.

Firstly, we should focus on refining the gameplay. One way we will do this is creating a set of different game layouts, using simple tools such as plastic cups to simulate the game mechanics and gather feedback from playtesting. This will help us immediately identify what works and what does not.

Secondly, 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 the shaking of the screen or enemies having animations when for example dying, but also smoke or sparks under a tower if it is low on health. 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 preventing 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, like lasers, 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 the tower designs come over as unintuitive to most testers, especially when the button of the Trigger and Railgun 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.

4.5 Play Testing

During these final few weeks of development, several features had been made but not yet integrated with each other. So, for the first test not every feature was enabled. To be more precise, the playable game consisted of two fully interactive towers and a handful of enemies.

The reception of this reduced version was still positive after the testers realized how the towers function. So, it is clear that the integration of the quality of life improvements and including visual or auditory cues to explain the features of all tangible components are both essential to focus on for the finished product.

4.6 Final Prototype

To be tested.

5 Discussion

During this paper, the central question was: How can tangible interfaces be used to enhance the player experience in a hybrid tabletop tower defense game? To address this question, the focus will lie on the current limitations of the design and look ahead to future work.

5.1 Limitations

During the development and testing phases, several limitations were identified. One significant issue was user confusion during interactions with different towers. This confusion was especially prominent with the Rail Gun and Trigger Towers due to hidden interaction points and unclear feedback. Users experienced frustration, which hindered smooth gameplay. This suggests that clearer visual cues on the towers are needed to create more intuitive interactions.

Another limitation was the lack of strategic depth in the gameplay. Although the core mechanics were functional, players expressed a desire for more dynamic and strategic elements. The unclear instructions regarding the coin system and overall game goals likely contributed to this gap in strategic depth, affecting players' understanding of the game mechanics.

In terms of visual and auditory feedback, it plays a crucial role in enhancing immersive gameplay. However, the feedback was sometimes either excessive or unclear, leading to confusion. This detracted from the immersive experience, highlighting the challenge of achieving the right balance between providing adequate feedback and avoiding overstimulation.

5.2 Future Work

The future work of this paper should focus on addressing these limitations through iterative design and user testing. This could be done by enhancing the visual and audio feedback. This can improve the player experience by adding more intuitive feedback mechanisms. This could include separate animations or sounds for each tower.

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 so that it is ensured that players understand the core game mechanics and the goals of the game.

By adding these adjustments the tangible interface will enhance the overall user experience. With the focus laying on clarity, strategic depth and balanced feedback, the future of Termite Terminator will enhance the player experience better. To conclude, whilst it is possible to argue that Termite Terminator is an interesting and engaging game, some improvements with interaction design, feedback systems, and the complexity of gameplay need to be made in order to enhance the prospects of tangible interfaces in hybrid games.

6 Conclusion

A conclusion can only be made at the end of the project.

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

[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] 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.

[9] 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.

[10] 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.

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

A Termite Terminator: detailed overview

Here we will dive in depth of all the game rules, enemy and tower design, with all the choices such as cost and effects further explained. These final settings will be determined with playtesting, to adjust the strength accordingly.

B Communication Information

Tower Type	IP Addresses	Send	
Light Tower	192.168.24.1, 192.168.24.2	"b" to boost the tower	"k" ki
Trigger Tower	192.168.24.3, 192.168.24.4	"b" to boost the tower	"k" ki
Railgun Tower	192.168.24.5, 192.168.24.6	"b" to boost the tower	"k" kill
Hammer	192.168.24.7	"hr" movement detected	

Table 1: Communication Mapping for Towers and Hammer

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009