Todo innan första inskick:

Läs igenom allt, stavcheck etc, fixa we’re till we are. Läs igenom hans förra feedback och apply’a den här.

Att ta med:

Tekniska aspekter hur vi gjorde våra program (mobil och spel).

Hur vi kom fram till Version 1 av spelen, våra egna tester.

Technological results? Som tex. att vår ping var 60ms typ? Ska vi skriva det och isf var?

Ta en titt på releaseable app (.exe’ish / store?) för att kunna släppa över sweclockers etc för testning.

TESTA HUR INTUITIVT KONTROLLERNA ÄR? Dvs. berätta inte alls hur dom ska använda keyboard och telefon, utan se hur lång tid det tar för dom att förstå det själva.

Frågor för tester:

**HUR VIKTIGT ÄR BRA SCORE VS ATT HA KUL TYCKER DU I DETTA LÄGET?!?!**

**(för att kunna diskutera om resultaten, kanske varför dom som inte tycker telefonen är kul tycker det (för att dom tkr score är viktigt)).**

Detta är vår RQ / hypo:

Can the gaming experience be improved by having the user play with a phone as a controller instead of keyboard on a computer?

Can a modern phones inputs and features provide good enough accuracy and response time to be used in computer games? Which types of games are they suitable for and which type of games are they not?

Vi ska svara på dom 3, så vi behöver ställa frågor till testarna som kan ge oss svar.

Kanske stryka Q2 av dom ovan? Har inget direkt sett att testa det. För Q3 måste vi fråga specifikt om just input lag / accuracy.

Wireless, är det nåt vi vill snacka om? Fördelar etc med att vara wireless?

Isf kanske: <http://link.libris.kb.se/sfxbth?ctx_ver=Z39.88-2004&ctx_enc=info%3Aofi%2Fenc%3AUTF-8&rfr_id=info:sid/summon.serialssolutions.com&rft_val_fmt=info:ofi/fmt:kev:mtx:journal&rft.genre=article&rft.atitle=Game+controllers+go+wireless&rft.jtitle=EDN&rft.au=Karl+H+Torvmark&rft.au=Bill+Schweber&rft.date=2002-09-19&rft.pub=Reed+Business+Information%2C+a+division+of+Reed+Elsevier%2C+Inc&rft.issn=0012-7515&rft.eissn=2163-4084&rft.volume=47&rft.issue=20&rft.spage=30&rft.externalDocID=191719261>

Possible vibration referenses:

<http://bth.summon.serialssolutions.com/sv-SE/search?s.cmd=goToPage%285%29&s.fvf=ContentType,Patent,t&s.pn=4&s.q=game+controller+vibration>

<https://login.miman.bib.bth.se/login?qurl=http%3a%2f%2fieeexplore.ieee.org%2fxpls%2fabs_all.jsp%3farnumber%3d6329846>

<https://login.miman.bib.bth.se/login?qurl=http%3a%2f%2fportal.acm.org%2fcitation.cfm%3fid%3d1067391>

Denna verkar skitbra: The Phone that Touches you back

<http://books.google.se/books?id=apqd0YXTtkMC&pg=PA52&lpg=PA52&dq=%22Hitting+virtual+keys+on+the+screens+of+next-generation+cellphones+will+feel+just+like+pressing+real+buttons.%22&source=bl&ots=6AbrDO0jdQ&sig=0Ntcvi-DffNZyazVp8qSPoKEdbg&hl=en&sa=X&ei=zPd8Ucm-IuHK4ASRzoGoBQ&ved=0CC8Q6AEwAA#v=onepage&q=%22Hitting%20virtual%20keys%20on%20the%20screens%20of%20next-generation%20cellphones%20will%20feel%20just%20like%20pressing%20real%20buttons.%22&f=false>

Arbete precis som vårt typ:

<http://ieeexplore.ieee.org.miman.bib.bth.se/stamp/stamp.jsp?tp=&arnumber=6329846>

Source för vilka inputs en touch phone har:

<http://bth.summon.serialssolutions.com/sv-SE/search?s.q=gRmobile%3A>

första träffen (gRmobile: A Framework for Touch and Accelerometer Gesture Recognition for Mobile Games)

Oil drill using haptics arbete:

<http://link.springer.com.miman.bib.bth.se/content/pdf/10.1007%2F978-3-642-31404-9_13.pdf>

ArkivEx:

<http://www.bth.se/fou/cuppsats.nsf/all/69362f6bce4dcc5fc1257a350047d98a/$file/BTH2012Bistr%C3%B6m.pdf>

<http://www.bth.se/fou/cuppsats.nsf/all/86eabecdf5c10879c12578fc00628886/$file/BTH2011Lindstrom.pdf>

<http://www.bth.se/fou/cuppsats.nsf/all/616f6ed2c22e96cbc12577e4006ee658/$file/rapport_dynamisk_terranggenerering.pdf>

Eye tracking

<http://link.springer.com.miman.bib.bth.se/chapter/10.1007%2F978-3-642-25200-6_7>

Structure:

ABSTRACT / KEYWORDS?!

Summary

1.0 Introduction

Background, Purpose and Goals, Research Question, Method, (Limitations)

1.1 Purpose and Goal

1.2 Limitations

1.3 Research question and hypothesis

2.0 Background

Beskrivning av område, Related Work

2.1 Related Works

2.2 Different types of Natural Mappings

2.2.1 Direct Natural Mapping

2.2.2 Kinesic Natural Mapping

2.2.3 Incomplete Tangible Natural Mapping

2.2.4 Realistic Tangible Natural Mapping

2.3 Vibration as an output

3.0 Game Implementation

Experiment Design / Method, Implementation

3.1 Starchaser

3.2 Labyrinth

3.3 Sniper

3.4 Technology Game

4.0 Method

4.1Participants

4.2 Procedures

4.3 Measurements

5.0 Results

6.0 Discussion

6.1 Method Discussion?

6.2 Result Discussion?

7.0 Conclusion

8.0 Future work

References

Appendix (bilaga)

*Bachelor Thesis BTH LOGO*

*Computer Science*

*Thesis no: BCS-2013-xx*

*June 2013*

Using your phone as a game controller

to your PC

Marcus Löwegren

Rikard Johansson

School of Computing

Blekinge Institute of Technology

Box 520

SE – 372 25 Ronneby

Sweden

Summary

Many people in the world today own a smartphone. Smartphones of today usually have an advanced array of inputs in forms of tilting, touching and speaking, and outputs in forms of visual representation on the screen, vibration of the phone and speakers for sound. They also usually have a lot of connectivity in forms of WLAN, bluetooth and USB. Despite this we’re still not seeing a lot of interaction between computers and phones, especially within games. We believe that the high presence of phones amongst people combined with the advanced inputs and outputs of the phone and with the connectivity possibilities makes the phone a very viable option to be used as a game controller for the PC. We will experiment with this developing the underlying architecture for the phone to communicate with the PC as well as develop a few different games that we will let users test to see if the phone’s inputs are good enough to make it suitable for such a purpose. We will also attempt to find out if doing this increases the gaming experience, or in other words the enjoyment, of a PC game.

Keywords?:

Game Controller??, Android??, Natural Mapping??, PC Game??, Phone-Computer interaction??, Phone game?? Human-Computer Interaction?? HIC??

Index

1.0 Introduction

1.1 Purpose and Goal

1.2 Limitations

1.3 Research question and hypothesis

2.0 Background

2.1 Related Works

2.2 Different types of Natural Mappings

2.2.1 Direct Natural Mapping

2.2.2 Kinesic Natural Mapping

2.2.3 Incomplete Tangible Natural Mapping

2.2.4 Realistic Tangible Natural Mapping

2.3 Vibration as an output

3.0 Game Implementation

3.1 Starchaser

3.2 Labyrinth

3.3 Sniper

3.4 Game Technology

3.4.1 Starchaser Game Technology

3.4.2 Labyrinth Game Technology

3.4.3 Sniper Game Technology

4.0 Method

4.1 Participants

4.2 Procedures

4.3 Measurements

5.0 Results

6.0 Discussion

6.1 Method Discussion?

6.2 Result Discussion?

7.0 Conclusions

8.0 Future work

9.0 References

Appendix (bilaga)

1.0 Introduction

1.1 Purpose and Goal

Our goal with this project is to find out if it is possible to make a game more entertaining and fun by having the user control a PC-games with a phone instead of the traditional keyboard and mouse, and if the phone is suitable for such a task. We will experiment with different kinds of games in different genres and compare using keyboard and mouse to the phone as game controllers to find out if the phone’s inputs are good enough, and if it is fun to use.

1.2 Limitations

We will limit ourselves to single-player games using only one phone and we will also make limitations on what kinds of people we will use when testing our games because we do not think that we will be able to get a test group with adequate statistical distribution.

1.3 Research question and hypothesis

Can the gaming experience be improved by having the user play with a phone as a controller instead of keyboard and mouse on a computer?

Can a modern phones inputs and features provide good enough accuracy and response time to be used in computer games? Which types of games are they suitable for and which type of games are they not?

2.0 Background

For this project we will continue research within the areas of Human-Computer interaction, User Interface and Human Interface Device. The problems within this area lies around making controllers and input devices with good usability (that feels intuitive, is effective and is easy to use and learn) [4], and we will investigate this problem specifically for games. There are a lot of different controllers being used as inputs for games, ranging from the standard computer-input, keyboard and mouse, to new and innovative controllers such as the Wii-controller [3], Xbox Kinect [1] and Playstation Move [2], with the standard console-like controllers to Playstation [6] and Xbox [5] for example being somewhere in between. We will experiment with finding possibilities with using a phone as a controller, since a phone is something many people own today, making it a convenient game controller. Phones of today usually have several different inputs that measures elevation and acceleration of the phone as well as a touch-screen and microphone [16]. We will experiment using some of these inputs to find out if they are good enough to make a game-controller of the phone.

2.1 Related Works

We have found a couple of previous studies in this area that are related to what we will do. “Implementing Mobile Phone As a Multi-Purpose Controller using 3D Sensor Technology” [7] is a study that is technologically close to what we will do but with a different focus. It focuses on the technology, whereas the technological parts are only a means to us. It does not at all investigate the fun and entertainment values which is what we will focus on. “Using Mobile Phones to Control Desktop Multiplayer Games” [8] is a study that also is close to what we will do technologically but it does not use the accelerometer of the phone. It says that the next step is to use the accelerometer because it allows the phone to facilitate a more natural way to control 3D games, which is what we intend to do. It also does not investigate fun and entertainment values of this kind of controller. “Mapping the road to fun: Natural video game controllers, presence, and game enjoyment” [9] is a study that compares the Wii-controller to a normal playstation controller and investigates how well the different controllers perform and how the participants enjoy the controllers. Methodically this study is similar to what we want to do, but we will be comparing a smartphone to a keyboard instead of a Wii-controller to a playstation-controller.

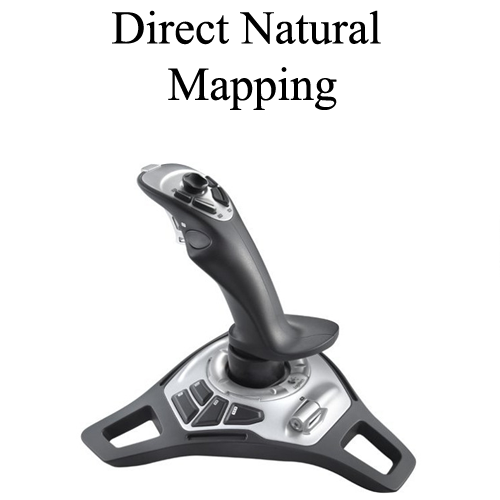
2.2 Different types of Natural Mappings

Natural mapping in the context of video games is typically thought of as how closely

actions represented in a game match the actions used to bring about that change in a real environment [11]. There are different kinds of Natural Mapping for game controllers. We are going to explain 4 of them: Direct Natural Mapping, Kinesic Natural Mapping, Incomplete Tangible Natural Mapping and Realistic Tangible Natural Mapping. Direct Natural Mapping is the keyboard and mouse, and Incomplete Tangible Natural Mapping is the phone.

2.2.1 Direct Natural Mapping

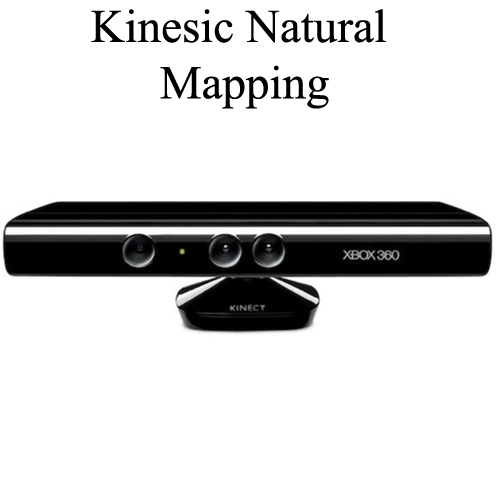
Direct Natural Mapping is the most basic way a controller can be mapped. An example from The Design of Everyday Things [10] of Direct Natural Mapping is a stove with four burners that are arranged in a two-by-two order. Most stoves would have the controllers for the burners lined up in a straight row while some stoves puts them in a two-by-two layout that correspond to the same two-by-two layout that the actual stove has. By using the second layout you make it more intuitive for the user to control the four burners separately. A more game related example could be a joystick (Fig. 1). If you implement so when you pull the joystick to the left your character turns right, the player would probably be troubled by how the controls worked and it would not be direct mapped, but if you would do it direct mapped so that when you pull the joystick to the left you turn to the left, the user would faster understand how the controls work. Tamborini and Skalski [13] argue that more naturally mapped gaming controllers should allow players to quickly access mental models of real-world behavior, if they exist for the player, thereby providing more accurate information about how to interact with the game, or even know how it works without testing it.



*Fig 1.*

2.2.2 Kinesic Natural Mapping

Kinesic Natural Mapping is when you use body movement that looks and feels like the movements that you would do in real life to control but you are not using any actual physical controller. An example of this mapping is Microsoft Kinect (Fig. 2). There are plenty of games that uses this device to capture the motions that the players does in front of it and uses them to control various game elements. By using this technique you let the player use his/her own physical experience and mental modes to control the games.



*Fig 2.*

2.2.3 Incomplete Tangible Natural Mapping

Incomplete Tangible Natural Mapping involves letting the user have a controller in the hand that represent the item that the user is going to control. The controller does not need to have the same shape or the weight but it has to be controlled the same way as the item on the screen or in the game environment. In a popular game called Wii Sports [14] there is a bowling mini game that uses a Nintendo Wii controller (Fig. 3) to simulate the control of the bowling ball, it is not the same shape or weight as a bowling ball but you are using the same motions as if you would throw a bowling ball in real life. The wrong shape and weight is what makes it incomplete mapping and not realistic mapping.



*Fig 3.*

2.2.4 Realistic Tangible Natural Mapping

Realistic Tangible Natural Mapping is the final and most realistic natural mapping and provides the highest amount of natural mapping of the four types. Many arcade games utilize this kind of mapping as you can only play one type of games on them. For example most arcade car racing games (Fig. 4) use realistic and tangible controllers such as steering wheel, a stick and pedals to control the game, this sort of games often have a special chair to make the driving experience even better. The controls let the player easily access the mental modes that he/she would use when he/she drives a normal car.



*Fig 4.*

2.3 Vibration as an output

When playing PC games with a mouse and keyboard we usually use 2 of our 5 senses to take in information from the games, sight and hearing. Game-controllers for consoles such as the Playstation 3 [6] often adds another sense, touch, also called the haptic sense, in form of vibration of the controller. The vibration can either be used to enhance virtual reality by for example emulating the recoil in a weapon when shooting, or by simply giving feedback to the player of something happening in the game. When playing a game the players vision is usually focused on the main task of the game [21]. Shifting the gaze to look at user interface objects such as texts telling you your score is distracting and cognitively demanding. Giving the player haptic feedback is advantageous in this kind of environment and lets the player focus their sight on the main tasks of the game. Humans receive visual and haptic information simultaneously and the nervous system is able to process them in parallel [20]. Humans also recognizes feedback in the form of touch faster than in the form of sight [19]. Vibration is a very common feature of phones that normally tells the person when there is an incoming call or when he/she receives a text message. Because of this implementing vibration when using the phone as a game controller is quite natural and simple way to tell the player that something important has happened. Even though the vibrators in phones and game controllers normally are too weak to convey actual forces [18], they can still be used to enhance gameplay in the form of giving the player simple feedback. The length of the vibration is important because the player must feel the vibration but the vibration should not distract or trouble the player, the optimal length of the vibration is between 50 ms and 200 ms [17].

3.0 Game Implementation

We chose to develop our own games and app that we were going to use in the testing, because we felt that doing so would give us complete freedom in changing parts of the program to tailor the system for its purpose. We ended up developing four games but we only used three games in the testing. The fourth game was a helicopter game that we ended up removing due to that it was going too complex to develop and we felt that it would be require too much time for the testers to understand it. The three games that we decided to test with were Starchaser, Labyrinth and Sniper. All three games will have a time limit of two minutes.

3.1 Starchaser (Fig. 5)

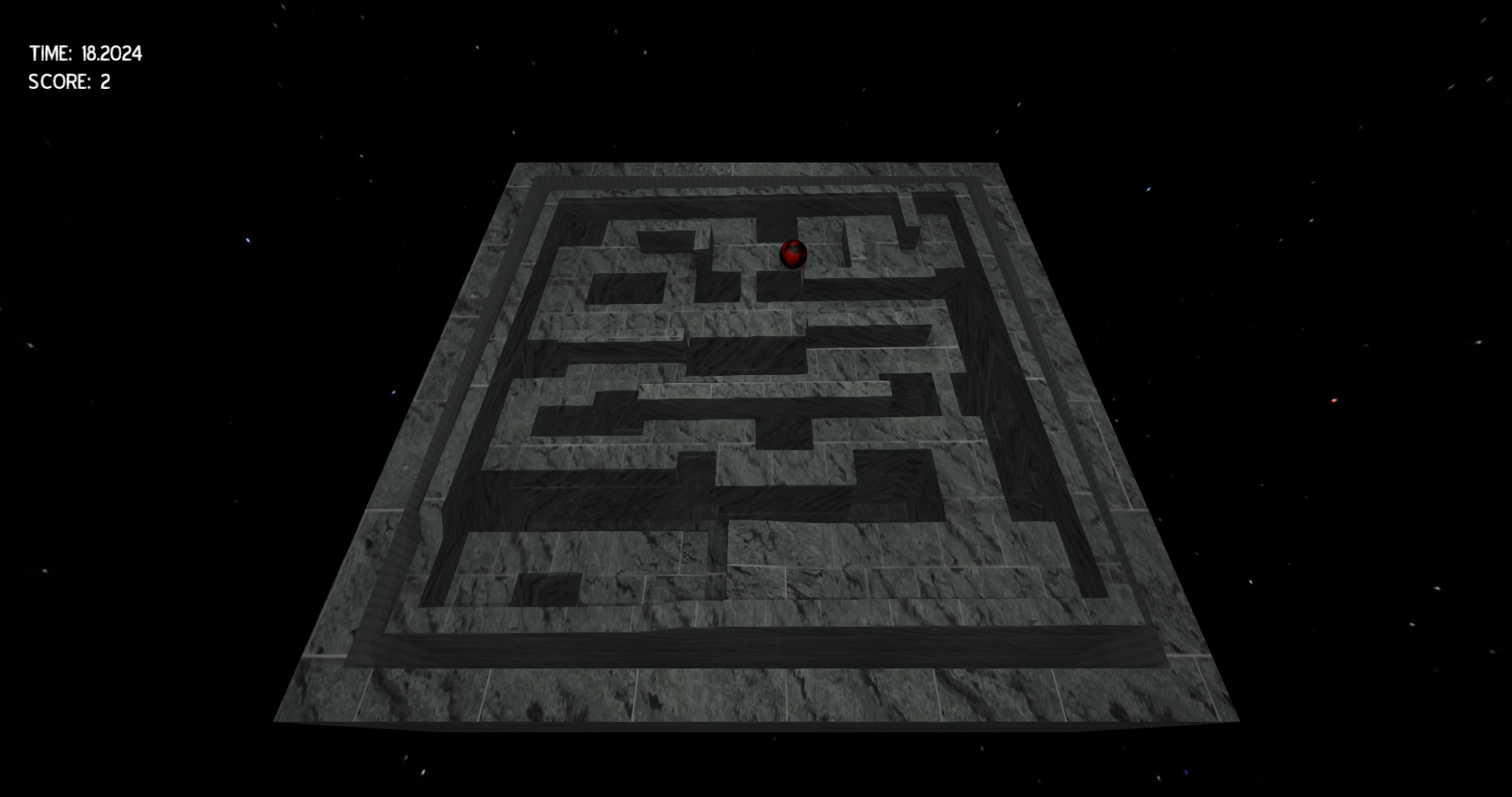
Starchaser is a racing game in space. The mission is to collect as many stars as possible before the time runs out. You gain one score for each star that you collect, and when you do collect a star the phone gives a short vibration as feedback, as well as a star being shown briefly on the computer screen. We wanted to test this game because the natural mappings does not really fit the mapping of a keyboard and mouse directly. When controlling with the keyboard you press “W” or “S” to pitch the spaceship up and down, you can press “A” and “D” to roll the spaceship. To change the speed of the spaceship you can press “+” and “-” on the numpad. To play with the phone you hold it in a steering wheel position and then you can pitch and roll the phone to do the same with the spaceship. If you want to increase or decrease the speed you touch the screen. If you touch at the top of the screen you accelerate and if you touch the bottom you decelerate.



*Starchaser. Fig 5.*

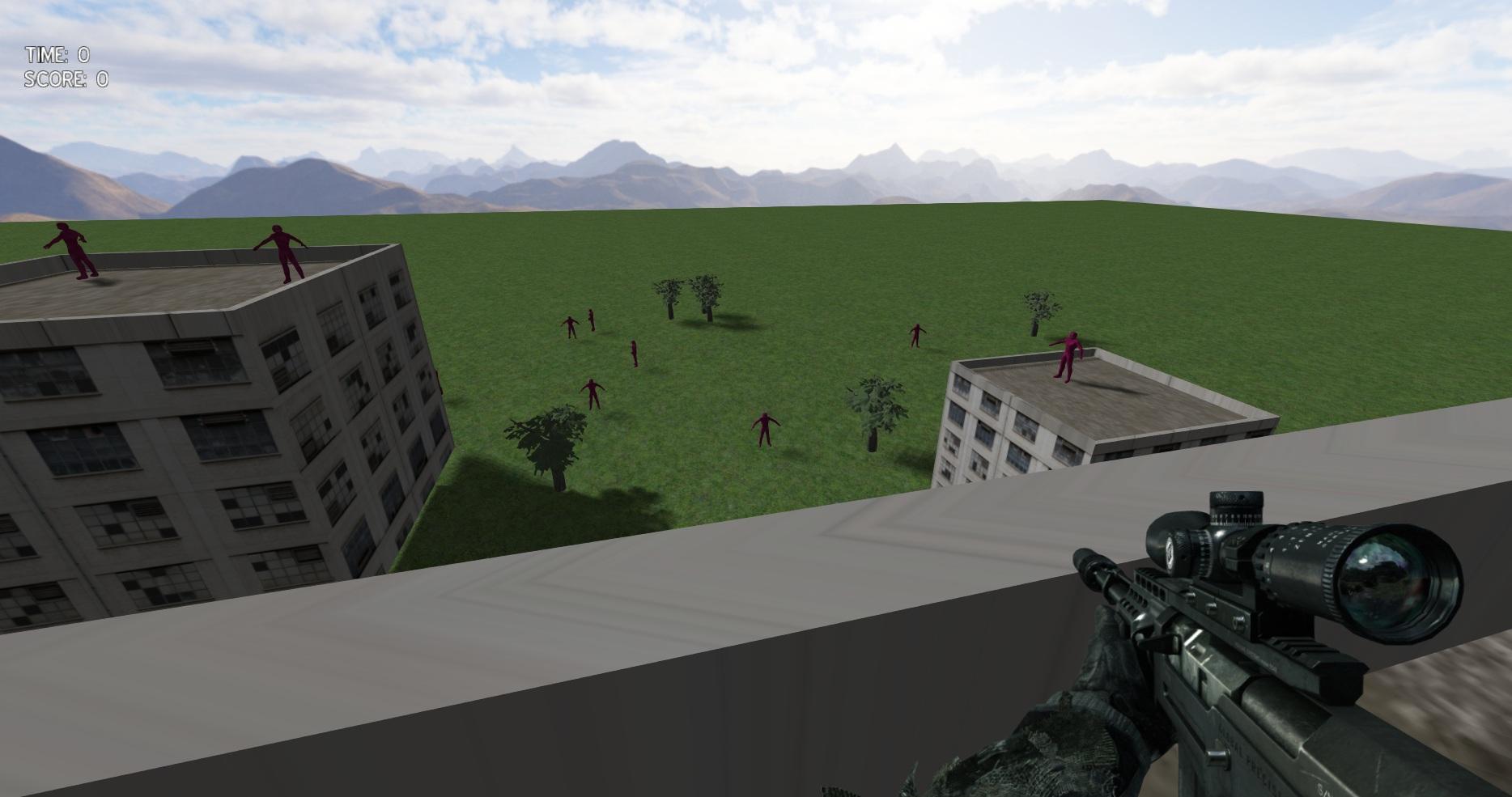
3.2 Labyrinth (Fig. 6).

Labyrinth is a puzzle game based on an old game first introduced 1946 by BRIO [12]. The goal in this game is to get the ball to the other side of the gamefield without falling down in the holes in the gamefield. To do this you need to tilt the game in different directions. There are a variation of games that let you tilt the gamefield in different ways. We have fourteen checkpoint spread over the map to be able to calculate score. You increase your score by moving the ball to these checkpoints. When you reach a checkpoint the phone gives off a short vibration and a star is shown briefly on the computer screen to make you aware of this. When the game ends you will get the score equivalent to the last checkpoint that you reached (if you finish the game you get a score of fourteen and your time is instead recorded) . We chose this game because we wanted a game where the phone had a near perfect mapping to real life. When playing with the keyboard you use the WASD keys to tilt and to reset the ball to the second last checkpoint you press the “R” button. If you playing with the phone you hold it horizontally with the display facing up and the tilt it in different direction like you would in real life. If you want to reset the ball you press the “Reset” button on the touch screen.



*Labyrinth. Fig 6.*

3.3 Sniper (Fig. 7)

Sniper is a first person shooter where you play as a sniper on top of a rooftop. Your mission is to shoot down all the targets. You get one point for each target that you hit. Every time you shoot the phone gives off a short vibration to simulate recoil of the gun, and when you hit a target it gives off a longer vibration and briefly shows a star on the computer screen to make the player aware of this. Usually FPS games are played with the keyboard and mouse, so we chose this game to see if how the phone performs on grounds where the keyboard and mouse usually is the prefered option. You use the mouse to aim on the targets and you press the left mouse button to shot, and right mouse button to scope in. To move around on the roof you press “A” (left) and “D” (right) on your keyboard. When you are playing with the phone you use the virtual joystick on the phone which is located on the left side of the touchscreen to aim and you press the “Shot” button which is located on the right side of the touchscreen to fire a shot. If you want to move to the left or right you tilt the phone in the direction you want to move. To scope in you tilt the phone away from you briefly.

*Sniper. Fig 7.*

3.4 Game Technology

All our games were developed in C++ using Visual Studio 2012 and a DirectX 11 Graphics Engine. The app was developed for Android (Samsung Nexus Galaxy/X) using Eclipse. The PC program is fully controllable with the app. You can move backwards and forward in the menus and you can also close the program. When you start the app on the phone it scans the network that it is connected by via WLAN with subnet 255.255.255.0 and port 10 000 for a connection and connects to the first ip that runs a server that it finds. When connected to the PC it checks if there is any game started, if a game is already running it starts the corresponding activity on the app, if there is not any game running it shows the select-game menu and it is ready to use. We have also implemented vibration in the phone to be able to give feedback to the player that something noteable has happened. All vibrations are between 50 ms and 200 ms in length, this because for the player to notice the vibration has to be above 50 ms and more than 200 ms is registered as irritating.

3.4.1 Starchaser Game Technology

The Starchaser game has 10 predetermined positions where stars can spawn and it simply loops through them to find a new position for the next star. We decided to implement it like this so that the results couldn’t be affected by randomness of how the stars spawn. We think that because it being in space and there is no clear up and down learning the pattern of the way the stars spawns will require a lot more time with the game than what the testers gets. The rotation of the spaceship is limited to 1 radian per second around both axises and speed changes are limited to 10% per second. We also have a minimum speed of 10% to prevent people from standing still and turning freely. The phone input uses an algorithm to trim the values to better fit the game. Because the input being a normalized vector we divide both the axises used for rolling and pitching by the third axis. This ensures that when rolling and pitching at the same time you don’t have to turn the phone more in each direction. The values of each axises are also scaling exponentially between 0.0 and 0.33, and linearly between 0.33 and 1.0. This makes the sensitivity of the phone inputs lower when it’s almost aligned for moving straight forward and it’s needed to remove flickering back and forth when trying to hold the ship in steady course.

3.4.2 Labyrinth Game Technology

The Labyrinth game has 14 predetermined positions that are score-positions. When you are within 2 length units of distance from a score position you get a score worth the value of the place that position is. For example moving straight to score-position number 7 means that your score gets set to 7. Only score-positions higher than your current score are tested. For this game we implemented realistic physics using force and direction of the ball. Every update it checks the entire platform’s triangles for collision against itself and if collision is found it checks the normal that triangle and adds force to the ball in that direction. Gravity is constantly pulling the ball downwards. We added a limitation to how much the platform could rotate to prevent cheating with quick turning. This limitation is 0.39 radians per second of rotation when using the keyboard, and a maximum limitation of 0.5 radians per second of rotation when using the phone. The reason that the phone has a higher limitation is that the keyboard is digital in its inputs making it impossible to move the platform slowly. Having a 0.5 radians per second rotation limit on the keyboard resulted in a too high sensitivity we felt, and making small move got very hard then. The phone’s input values are divided by 4 as well to reduce the sensitivity of the phone, for the same reason as the keyboard, meaning that when rotating the phone 0.8 radians in real life, the platform only gets rotated 0.2 radians in game.

3.4.3 Sniper Game Technology

The Sniper game has 10 predetermined spawn locations for targets. Each target will spawn at its location and 5 of them will start moving back and forth between a waypoint. When a target is hit it will get removed, and it will respawn after 5 seconds at its spawn location. When shooting we have implemented a recoil that lasts for 1.1 seconds and a delay between shots for 1 second to increase realism and increase the difficulty of the game. Previous rotations of the phone gets stored and then compared to the current rotation to determine whether the phone has been rotated enough to trigger the effect of moving left and right and scoping in. This to prevent the phone from having a static default position, and the user can then hold the phone at whatever rotation he/she wants to. The emulated joystick on the phones touch screen uses an algorithm to trim its values. The values gets sent as a two-dimensional vector containing the distance from the center of the joystick that the user is touching the screen in X and Y coordinates. It uses a circular limit of how far you can move the joystick of 150 length units in any direction. The algorithm trims these values by first clamping them to 0 if they are too small to create a deadzone where the user can touch the phone without the aim moving. After that the value is exponentially increased to make the sensitivity non-linear to allow for small fine movements as well as large fast turns.

4.0 Method

A total of X participants played our 3 games to help us with our study. We let the participants play each game in a random order and starting with different input devices each time to prevent any learning to interfere with the results. They were going to test if the gaming experience was better or worse when steering with the phone instead of the keyboard. They were also test if the vibration in the phone was a good output.

4.1 Participants

The participants was mostly friends of ours that wanted to be in the experiment uncompelled we also got some people from the school that wanted to participate. The age range was from X to X years and X participants were female. We divided them into two groups, gamers and non-gamers, and we defined a gamer as a person who play videogames 10 hours a week. X persons was put in the gamer group and the rest was put in the non-gamer group. The reason we split into these groups are that we predict that people who are gamers will have a lot of experience using the keyboard and mouse as game-inputs which will give them an advantage when using them, while non-gamers will not have as much experience. We will also have a group of people that we will record with a camera to get more in depth result that we can analyze more.

4.2 Procedures

The test persons that came to the experiment room were asked to sit in a chair in front of the screen. We switched the first input device every person, so if person one started with keyboard then person two started with the phone. We told them to try the controllers out before we started the test but they were not allowed to play any of the games. When they knew how to steer and use the control they were asked to play each game for 2 minutes each regardless of the result in the game. When they had played with one device they were asked to play with the other after, and when they had played with both devices they switched to a new game. When they had played all the games we interviewed them and asked the same questions as we had on the online form (*Table 1*). By interviewing them we could assure that they understood all the questions and we could ask follow up questions if we wanted to. The other participants that we did not bring to the experiment room were just asked to play the games for 2 minutes and not to start with the same device all the time and then answer the online form that we made (*Table 1*).

4.3 Measurements

We measured the results from the experiment by reviewing the clips that we filmed when the participants played and looking at the answers each person submitted. We valued the results from the people in the experiment room more because we controlled that environment so nothing would interfere with the results. We put people in one of the two categories by looking at the question “How many hours do you play video game each week?” on the online form (*Table 1*).

*Table 1*.

General:

"Gender?" (Man or Woman)

"How old are you?" (Any number)

"How many hours do you play video game each week?" (0 - 5; 5 - 10; 10 - 15; 15+)

"On what device do you play games the most?" (Mobile Phone, Computer or Console)

Questions for each game (Space Ship, Labyrinth and Sniper):

"It was more fun to play the game with the mobile phone than the keyboard and mouse"(1 Strongly Disagree - 7 Strongly Agree)

"It was harder to play with the mobile phone than the keyboard and mouse"(1 Strongly Disagree - 7 Strongly Agree)

Overall:

"The vibration in the mobile phone increased the gaming experience."(1 Strongly Disagree - 7 Strongly Agree)

"What device was the most fun to play on?" (One of the two devices)

"What game was the most fun to play?" (One of the three games)

5.0 Results

(Kan inte skriva än)

6.0 Discussion

(Kan inte skriva än)

7.0 Conclusion

(Kan inte skriva än)

8.0 Future Work

Phone - Computer interaction is an area with much left to be done, especially within games. Based on our results we definitely think that the phone very well can be a viable option as a game controller for the PC. We see the potential of future work to be done where the possibilities of multiplayer is examined. Since phones are so accessible we can see the potential of multiplayer games using several phones as game controllers to a single computer being very popular. Since the technique we used for the communication between the phone and computer in theory supports up to somewhere around 250 simultaneous phones being connected to the computer, we could see the potential of people coming together to play computer games with each other in real life, without having to worry about buying additional controllers, or even bringing their controllers, since they would always have their phone with them anyway. Implementing classic board games such as trivial pursuit [15] could also work very well with the phone as each player's controller. Another potential area for future work could be integrating the phones speakers to be used as an output of game-sounds. The speakers in todays phones are getting better and better and could potentially replace external speakers or headset that people normally use today for playing sounds in games.

9.0 References

[1] Microsoft, Introducing kinect for Xbox 360, http://www.xbox.com/en-US/kinect, last visited, April 2013. [Online]. Available: http://www.xbox.com/en-US/kinect

[2] Sony, This Is How I Move™, http://us.playstation.com/ps3/playstation-move/, last visited, April 2013. [Online]. Available: http://us.playstation.com/ps3/playstation-move/

[3] Nintendo, Controls for Wii, http://www.nintendo.com/wii/what-is-wii/#/controls, last visited, April 2013. [Online]. Available: http://www.nintendo.com/wii/what-is-wii/#/controls

[4] Löwgren, Jonas; Stolterman, Erik, *Design av informationsteknik – materialet utan egenskaper*, 1998, Studentlitteratur, Lund.

[5] Microsoft, Xbox 360 Controller, http://www.xbox.com/en-US/xbox360/accessories/controllers/Xbox360Controller, last visited, April 2013. [Online]. Available: http://www.xbox.com/en-US/xbox360/accessories/controllers/Xbox360Controller

[6] Sony, DUALSHOCK®3 Wireless Controller, http://us.playstation.com/ps3/accessories/dualshock-3-wireless-controller-ps3.html, last visited, April 2013. [Online]. Available: http://us.playstation.com/ps3/accessories/dualshock-3-wireless-controller-ps3.html

[7] Kiran, Bhagiya; Patel, Sankita, Implementing mobile phone as a multi-purpose controller using 3D sensor technology, In 2010 3rd IEEE International Conference on Computer Science and Information Technology, ISBN 9781424455379, Volume 2, pp. 584 - 588.

[8] Malfatti, Silvano Maneck; Ferreira dos Santos, Fernando; Rodrigues dos Santos, Selan, Using Mobile Phones to Control Desktop Multiplayer Games In 2010 Brazilian Symposium on Games and Digital Entertainment, ISSN 2159-6654, ISBN 9781612843919, pp. 230 - 238.

[9] Skalski, Paul; Tamborini, Ron; Shelton, Ashleigh; Buncher, Michael; Lindmark, Pete; Mapping the road to fun: Natural video game controllers, presence, and game enjoyment, In New Media Society 2011 13: 224, originally published online 12 October 2010, DOI: 10.1177/1461444810370949, The online version of this article can be found at: http://nms.sagepub.com/content/13/2/224

[10] Norman, Donald A. The Design of Everyday Things, 1988 Basic Books, New York.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*[11] Tamborini, Ron; Bowman, Nicholas David; Presence in video games. In: Bracken, Cheryl Campanella; Skalski, Paul; (eds) (2010) Immersed in Media: Telepresence in Everyday Life, 87–109, Routledge, New York. (Författare stämmer, kapitel finns, kan dock inte läsa texten så det måste kollas.)

[12] BRIO, Labyrinth Game, http://www.brio.net/ToPlay/3\_years/Games/34000\_Labyrinth.aspx, last visited, April 2013, [Online], Available: http://www.brio.net/ToPlay/3\_years/Games/34000\_Labyrinth.aspx

[13] Tamborini Ron, Skalski Paul; The role of presence in the experience of electronic games. In: Vorderer, Peter; Bryant, Jennings; (eds) (2006), Playing Video Games: Motives, Responses, and Consequences, 225–240, Mahwah, NJ: Lawrence Erlbaum.

[14] Nintendo, Wii Sports http://www.nintendo.com/games/detail/1OTtO06SP7M52gi5m8pD6CnahbW8CzxE, April 2013, [Online], Available: http://www.nintendo.com/games/detail/1OTtO06SP7M52gi5m8pD6CnahbW8CzxE

[15] Mary Bellis, The History of Trivial Pursuit http://inventors.about.com/library/inventors/bl\_trivia\_pursuit.htm[, April 2013, [Online], Available:](http://www.nintendo.com/games/detail/1OTtO06SP7M52gi5m8pD6CnahbW8CzxE) http://inventors.about.com/library/inventors/bl\_trivia\_pursuit.htm

[16] Joselli, Mark; Clua, Esteban; grmobile: A framework for touch and accelerometer gesture recognition for mobile games in Proceedings of the 2009 VIII Brazilian Symposium on Games and Digital Entertainment, ser. SBGAMES ’09, Washington, DC, USA: IEEE Computer Society, 2009, pp.141–150, last visited, May 2013, [Online]. Available: http://dx.doi.org/10.1109/SBGAMES.2009.24

[17] Kaaresoja, Topi; Linjama, Jukka, Perception of Short Tactile Pulses Generated by a Vibration Motor in a Mobile Phone. In: (2005) WHC 2005. First Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, pp. 471–472.

[18] Badshah, Akash; Gupta, Sidhant; Morris, Daniel; N Patel, Shwetak; Tan, Desney; GyroTab: A Handheld Device that Provides Reactive Torque Feedback. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM New York, NY, USA ©2012, pp. 3153-3156.

[19] Burdea, Grigore, Haptics issues in virtual environments, In Computer Graphics International, 2000. Proceedings, 295-302.

Länk nedan till den ovan: (ta bort innan inlämning)

<http://www.ti.rutgers.edu/publications/papers/2000_ieee_cgi.pdf>

[20] Goldstein, Bruce, Sensation & Perception, 8th ed. Wadsworth, Belmont, 2009

Har inte läst boken, bara snott referensen från ett arbete:

<http://link.springer.com.miman.bib.bth.se/content/pdf/10.1007%2F978-3-642-31404-9_13.pdf>

Han referenserar 7th edition, men på BTH finns bara 8th, borde vara samma typ...

[21] El-Nasr, Magy Seif; Yan, Su, Visual Attention in 3D Video Games in Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology, ACE '06 New York, NY, USA ©2006, Article No.: 22.

Appendix