Augmented Reality Games for Neurological Rehabilitation

Final Report

Spring 2012

by

Jacob Poore

Baris Tevfik

Prepared to partially fulfill the requirements for

ECE402

Department of Electrical and Computer Engineering

Colorado State University

Fort Collins, Colorado 80523

Project advisor: Dr. Sudeep Pasricha

Website: www.argamesforrehab.tk

ABSTRACT

Many people are affected by neurological impairments due to traumatic brain injury (TBI) and stroke each year. It was estimated that 795,000 people suffered from stroke alone in 2010 in the United States.[1] It can happen to anyone regardless of age, race or sex at any time. Such impairments result in loss of ability to carry out day-to-day activities. Even reaching to a cup on the dinner table becomes a very difficult task for people with these impairments. Physical disability caused by stroke, TBI and other diseases such as cerebral palsy has often a very long recovery period. Current rehabilitation(rehab) programs offer variety of settings but extensively take place in special clinics or in hospitals. With many people who are looking to be treated, the system is overwhelmed with the demand and lacks necessary resources to help individuals. Moreover, long recovery time of such impairments, makes rehab programs expensive for people in need.

We realized that a relatively new technology, Augmented Reality(AR) can help the current system of rehabilitation for neurological impairments, by bringing the treatment to patients' homes. In the meantime, allowing doctors to track their patients' progress easily and make necessary adjustments to the rehab program. This technology also allows to keep the programs lost cost, which is valuable for those in need, but are not able to pay for other settings of rehab programs. We are developing a framework with AR games that aim to rehabilitate people and help them regain their ability to carry typical day-to-day activities. AR allows a patient to use real objects to interact with computer generated environments. Unique markers attached to real objects enable the system to track the position and orientation of each object. A daily use webcam captures the image of the markers, then the system augments the real environment with computer-generated 3D graphics.

In this paper we discuss the development of the AR games using available libraries and software, the technical difficulties we encountered, and possible future work. The framework is an early prototype containing four games. We currently don't have any data to show how effective the AR activities on hand-eye coordination, range-of-motion, grasp strength etc., as the games have not been tested on patients yet. Our plans for this semester included testing the framework on patients that can give useful feedback to us. However, Department of Occupational Therapy has not been able to carry out the necessary testing with patients, due to issues on their end.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter I - INTRODUCTION

Technological advancements are making our lives easier by helping with us to carry out daily activities. Computers control production lines in factories, banking transactions, planes and so many things in our world as today. Most people have personal computers at their homes to communicate with the rest of world, to get instantly news, access to information and to entertain. One major field, that computers are getting more important every day, is the medical sciences. Robots assist doctors to operate surgeries, computers perform DNA extractions, diagnose diseases, and so on. With the high number of patients who have neurological impairments caused by TBI, stroke and cerebral palsy, computers and technology become once again very important to rehabilitate such people. Current system of neurological rehabilitation has reached its capacity due to the high demand , and it is clear that a new approach is needed for neurological rehabilitation with the help of new technology.

As senior computer engineering students in Colorado State University, we have taken the task to develop a framework consisting of AR based computer games and activities that are focused to enhance hand-eye coordination, range-of-motion and grasp strength of neurologically impaired patients. With the success of the project, anyone in need will be able to perform their rehab sessions at the comfort of their home. Moreover doctors will be able to track their patients' progress and make necessary adjustments to the program. Along with this new approach to the neurological rehabilitation, we are hoping to ease the current rehabilitation system in the United States and elsewhere. One of our primary goal is also to make the framework very low cost to the end user. Because of this, we have developed our current framework with minimal hardware support, and mainly focused on the development of software based applications. All the user will need in terms of hardware to start the rehab will be a daily use laptop/desktop and a webcam. Our games are specifically intended to be used in upper-limb rehabilitation as it is a vital part of the human body to perform daily activities.

AR is relatively a very new technology for general public. One of the first works on augmented reality started in Boeing in 1990. It was developed to remove the needs for complex user manuals for cabling during construction.[2] Later, in 1992 further augmented reality work undertook in U.S Air Force. However, it wasn't until 1999 when Hirokazu Kato released the ARToolKit that began the open source developments of new AR based applications.

AR allows people to use real world objects to interact with computer generated environments such as video games. It eliminates the use of keyboard and mouse for game inputs, and provides an intuitive way to play the games. Markers attached to real objects enable the system (via a webcam) to track the position and orientation of each object as it is moved. The system then augments the captured image of the real environment with computer-generated graphics to present a variety of games to the patient.

A new platform called GoblinXNA[4] was released by Computer Graphics and User Interfaces Lab in Columbia University in 2009. It was developed by Ohan Oda and Steven Feiner. This new platform is based on the Microsoft's XNA Games Studios[5] and is a very resourceful tool to develop AR based video games. It uses ALVAR tracking library[6] to track the position of AR markers.

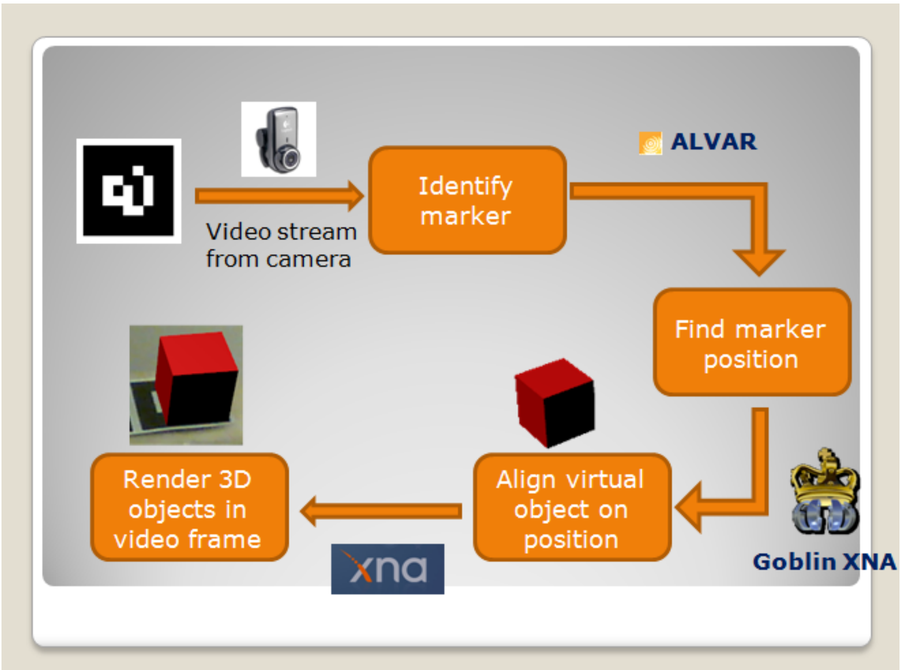


Figure 1. How It Works: A simple diagram of all the components of the framework.

The figure above displays a simple and brief sequence diagram of the actions taken by the system to show how AR works in our framework. First, the webcam stream containing the image of the marker, is decoded by ALVAR[6] and the unique marker is identified. Then. the position and the orientation of the marker relative to the camera is obtained from ALVAR[6] by Goblin XNA[4] and this information is used to align the virtual object. Finally this 3D object is rendered in the video frame. XNA Game Studio[5] provides the necessary tools to load contents, update and draw frames that are acquired by any computer game. The code is written in C#, developed on Microsoft Visual Studio 2008.

In the next chapter, we will be shortly discussing the previous work done by last year's team, Goblin XNA[4] and ALVAR[6]. In chapter III, the technical features of the individual games will be explained as well as the adjustable game area. In chapter IV, technical difficulties that we encountered will be discussed. In chapter V, we will discuss the design decisions and alternative approaches we may have taken. Then, chapters VI, VII, will briefly touch on product marketing and ethics. Finally, in chapter VIII, we will conclude the paper along with suggestions on possible future work.

Chapter II – PREVIOUS WORK

Our project is a continuation project of 2010-2011 senior design project, Wearable Computing, later changed the name to Upper-Limb Stroke Rehabilitation in second semester. Last years' team developed a framework that recognized different colors, and used this input to play games, draw etc. To do the color tracking they used light globes, since they provided better colors in different light settings. Although it was a good start for the project, color tracking did not give a comprehensive solution to AR games for upper-limb rehabilitation.

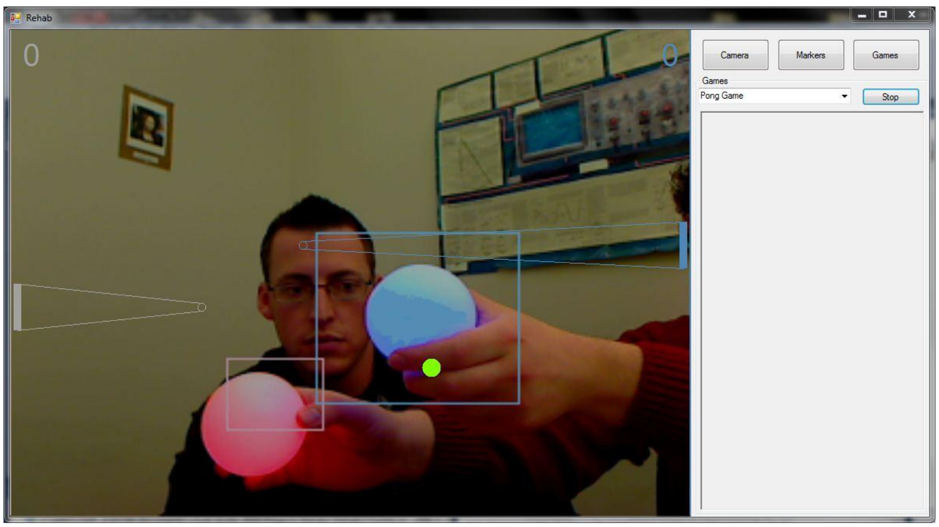


Figure 2. Last Year's Project: Pong game developed by last year's team using light globes for color tracking.

Goblin XNA[4] is an open-source platform for research on 3D user interfaces, with an emphasis on games. It's this unique characteristic of emphasizing on games made it a good candidate for basis of our project. It is written in C# and based on Microsoft XNA platform. The platform supports 6DOF(six degree of freedom) position and orientation tracking using marker-based camera tracking through ALVAR[6]. It uses a scene graph to support 3D scene manipulation and rendering, mixing real and virtual imagery. Physics is supported through the Newton Game Dynamics[7] library. Goblin XNA was developed by Computer Science Department at Columbia University.

ALVAR[6] is a software library for creating virtual and augmented reality applications. ALVAR has been developed by the VTT Technical Research Centre of Finland. The first versions of the library mainly support marker-based augmented reality applications. ALVAR is designed to be as flexible as possible. It offers high-level tools and methods for creating augmented reality applications with just a few lines of code. The library also includes interfaces for all of the low-level tools and methods, which makes it possible for the user to develop their own solutions using alternative approaches or completely new algorithms[6].

Chapter III – TECHNICAL FEATURES

The rehabilitation system can be set-up in a very simple manner. An everyday use laptop and a webcam is sufficient in terms of electronic components. Markers can be printed on regular office paper and can be placed on wood, cardboard, LEGOs, your hand, or any type of material.

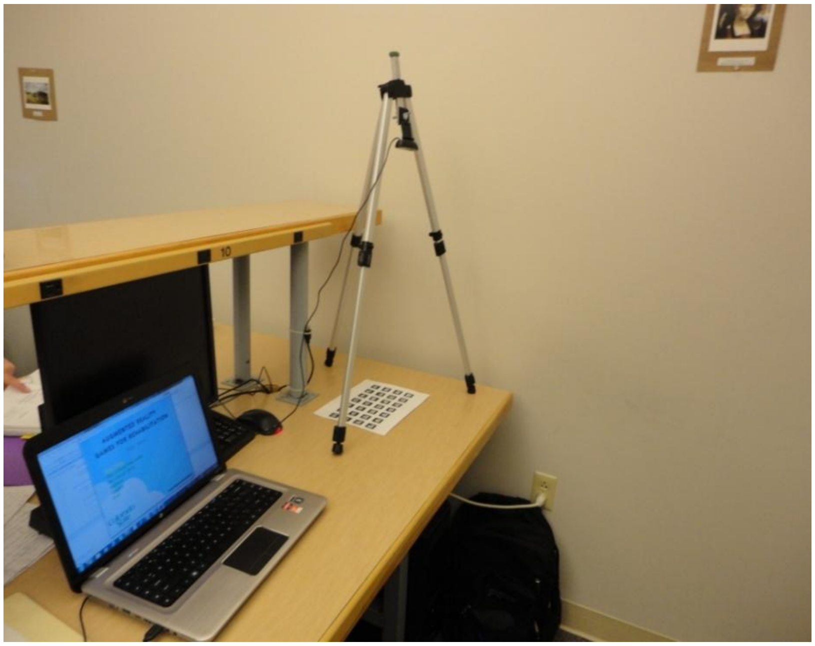


Figure 3. *AR Games Hardware System*: Tripod with a webcam, AR marker and a laptop.

The following is a brief explanation of the basics of the games.

After the transformation matrix and size of the game play area is acquired from setup ground(Section III.3), the game objects (edges, ground and ball) are place to corresponding locations.

```
edgeNode1 = new GeometryNode("Edge1");
edgeNode1.Model = new Box(5, sizeY + 5, 10);
```

Above, a GoblinXNA GeometryNode with dimensions X:5, Y:sizeY+5, Z:10 is created. "sizeY" parameter is the size of the game area in Y direction (the depth) relative to camera. This object becomes the red user's wall on the left (Figure 5b).

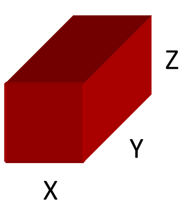


Figure 4. *Dimensions*

The default placement of objects on nodes is in the middle. Since we want to add the red edge on the left side of the game area, a TransformNode is created. Transform nodes allow a user to move rotate or scale whatever is a child of that node. Here a translation is made using a 3 dimensional vector.

```
TransformNode edgeNode1Transform = new TransformNode();
edgeNode1Transform.Translation = new Vector3((sizeX / 2) + 2.5f, 0, 0);

gameAreaTransform.AddChild(edgeNode1Transform);
edgeNode1Transform.AddChild(edgeNode1);
```

First the transformNode is added to the game area, then the object (red edge) is added to the transformNode. Other objects such as the ball and blue edge are created and added to game in the similar manner.

The physics in the game is handled by Newton Physics.[7] Each object's kinetic friction, static friction, elasticity is initialized. Also, each pair of objects has a `ContactProcessCallback` function that determines the steps to take when these two objects collide with another. For example; when the ball hits one of the balls, the score for the corresponding player is incremented, and a collision sound is played.

The initial speed of the ball in Pong is set by accessing its physics.

```
sphereNode.Physics.InitialLinearVelocity = ballSpeed;
```

ballSpeed is a three dimensional vector where Z:0, since the ball only moves in two dimensional space. X and Y values are initially set to 100 units, and can be changed during the game play.

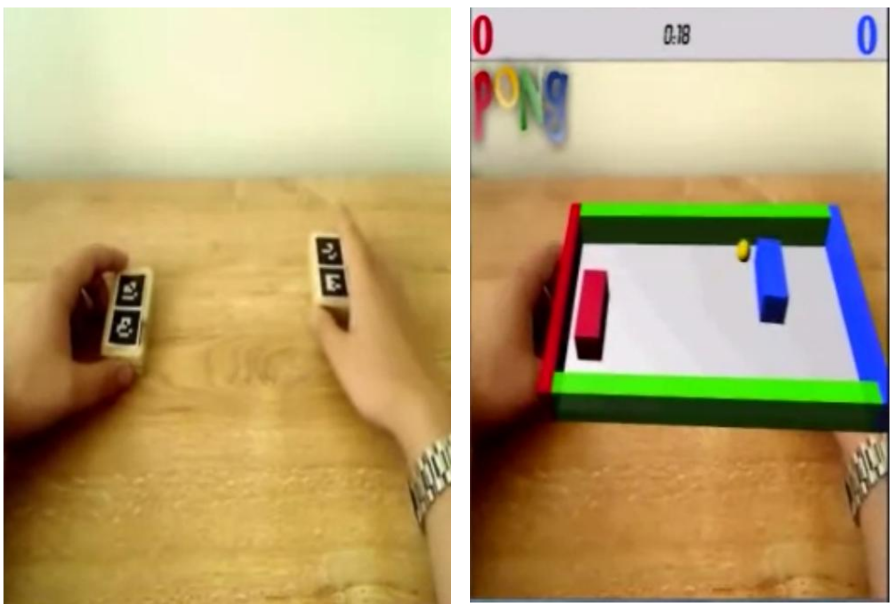
III.1 – Pong

Pong is one of the earliest computer games developed in 1970s. It is a tennis-like sports game that is played in two dimensional space. With its unique place in video game history, it is most widely recognizable by the general public. This characteristic of the game, makes it a good candidate for neurological rehabilitation. It provides a meaningful play that is physically challenging, to the patient.

The player's goal is to prevent the ball hitting to his side of the wall by rebounding the ball from the paddle. The paddle's are controlled by real objects(Figure 5a) that are, in this case, wooden rectangular boxes. Although, any shape of object can be used, as long as the markers are attached, the rectangular box shape gives the user an intuitive way to relate the computer generated image(Figure 5b) with the real object. If the player can't hit the ball, the opponent scores one point. The user wins by achieving 11 points against the opponent. A stopwatch starts when the game starts to record the elapsed time in the game. This information can be used to keep track of the patient's progress since the rehabilitation started.

The game play area is adjustable for different needs of different patients. This part of the system, is explained in section III.3. Moreover, the user can select the speed of the ball by pausing the game (esc button is used to pause the game). A patient who has just started the rehabilitation and who can't move their hands fast enough for the default speed, can start by setting the ball to go very slow. By incrementing the ball speed, a patient can put up goals that he wants to achieve over time.

In addition to adjustable game area and ball speed, the user can select the number of players before starting the game. 1 player option is a very basic mimicking of playing against computer. The blue paddle moves at a constant speed, and doesn't have any intelligence. 2 player option can be played by one person to enhance the coordination between two hands. Otherwise two different patients can play against each other, or one patient against a family member. For example; This would be a great way for parents to interact with their child who are going through neurological rehabilitation.



(a) (b)

Figure 5. *Pong*: (a) The real world scene, with the player holding the paddles. (b) Computer screen, displaying virtual scene of the game along with the augmented real world images.

III.2 - Follow the Leader

The second game we created is called follow the leader the basic premise of this game is that the player's marker must travel to a circle in the game area. The player is rewarded with more points the faster this is done. This game is very simple, two dimensional and because of this was another excellent candidate for rehabilitation.

The game initially started as simply going to the next circle, but as the game progressed we decided to begin adding more features. One of the first things we added was adjustable levels. All of the levels are saved in a file that can be changed to a user's specific need. This allows a doctor to make the circles pattern however they choose so they can help facilitate a specific patients rehabilitation needs. The next feature added was scores, the scores allow the doctors to track progress, through scores improving, and give users a goal to work towards. The point system works by adding a set amount of points per marker you reach then deducting points based on how long it takes you to get to the next one. Another feature is adjustable difficulties this allows the users or doctors to begin challenging themselves the difficulty will adjust the game play area, harder is larger and also increases the points for each marker. Also the points will decrease faster on harder difficulties. One of the last features put in was what we call line mode. This creates a dotted line between the marker you were at and the marker you are going to. The line changes color based on how far away from it you are. The closer you are the more points you get. (Figure 6b)

The last thing we began working on was making the game more visually attractive. To do this we add the ability to have custom themes. Currently a theme only includes the background and the player piece. We currently only have 2 themes plain and space and this is just due to time constraints on designing new models. We also added several things in the game to make it more appealing such as a running time and score counter. Finally we added a feature using the particle system of the physics engine to create an explosion of stars when the user reaches their mark. At the very end of the game the users are given a high score table and the ability to retry or continue onto the next levels.

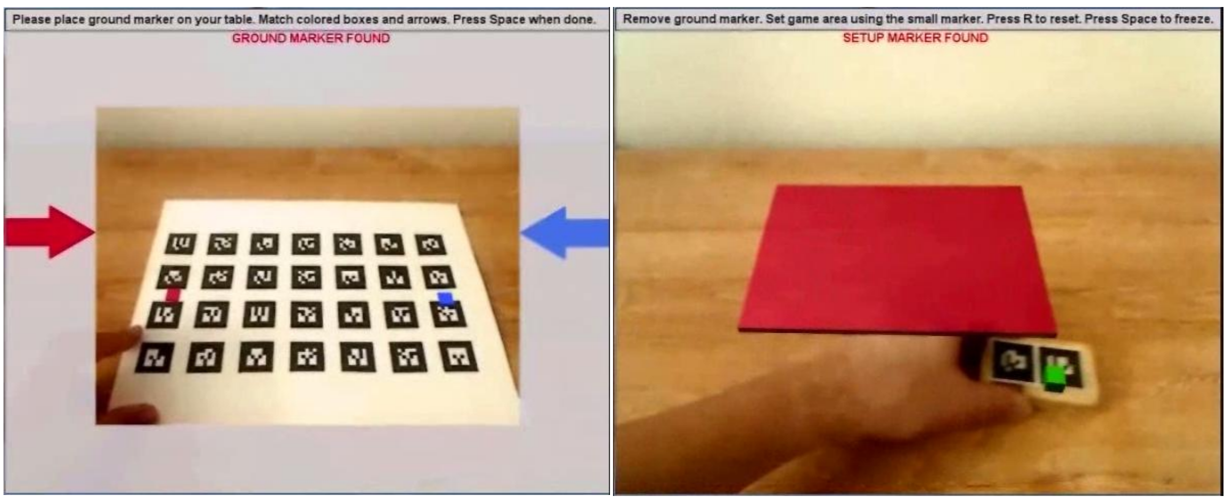
(a) (b)

Figure 6. *Follow The Leader*: (a) Showing what the user would see. (b) What would be on the computer screen.

III.3 - Setup Ground Area

The ground setup method was one of the last things we did in the first semester and the purpose of it was to calibrate the games to the users before they start.

The basic principle is first you place the marker on the ground and align it this allows the program to get the relative position of the play field, and more importantly the rotation of the ground (Figure 7a). Once the ground marker is placed and information obtained it can be removed. Then the user can use a marker that is used to play the game to draw the play area that they can reach this currently only works in the X and Y direction but future version will include a Z (Figure 7b). after the users reach as far as they can they exit to the main menu. All information obtained during the setup is returned to the parent method so all children can call the parent and obtain the size of the ground and the rotation matrix.



(a) (b)

Figure 7. *Setup Game Area*: (a) displays the ground marker to capture the table profile. (b) shows the user adjusting the ground area to their personal reach.

III.4 - Menu System

We have had some issues (constant crashes, freezes etc.) with the menu system and it was taking too long to debug than we initially thought. Therefore, we decided that it is best to have the games run individually without a menu, so we removed the menu system that was created initially. This allowed us to have the time to develop two more games.

III.5 - Whack a Mole

Whack a Mole is a classic game that has been around for decades. The mechanical versions of typical Whack a Mole machine consists of a large, waist-level cabinet with holes in its top and a large, soft, black mallet. Each hole contains a single plastic mole and the machinery necessary to move it up and down. Once the game starts, the moles will begin to pop up from their holes at random. The object of the game is to force the individual moles back into their holes by hitting them directly on the head with the mallet, thereby adding to the player's score.

When we were looking for different games that can be added to the framework of augment reality games for rehabilitation, we came across with this classic and fun game. The augmented reality version of Whack a Mole works the same way as its mechanical counterpart. The player plays with the small player piece that was used in our other games such as the paddle in Pong. This piece acts as the mallet in the game. The game area has an adjustable size as always. The game has 3 difficulty modes; easy, normal and hard. This gives the patients a great range of challenges. These modes let the mole to stay in its current position for 5 seconds, 3 seconds and 1 second respectively. Each hit and miss is distinctively shown on the screen with green or red colored text. Total hits and misses are also counted until the end of the game. The player has 2 minutes of time to play the game. The game has no lose or win situation at the end. However, number of hits and misses can reveal the progress of the patient easily over time.

One great characteristic that the other games don't have is that Whack a Mole allows patients to move their arms up and down. This different range of motion can be very helpful for daily tasks such as moving a coffee mug up-down. This game can be also very beneficial for hand-eye coordination since the patient will need to follow the mole as it changes its location quickly and move their hand accordingly to the next location.

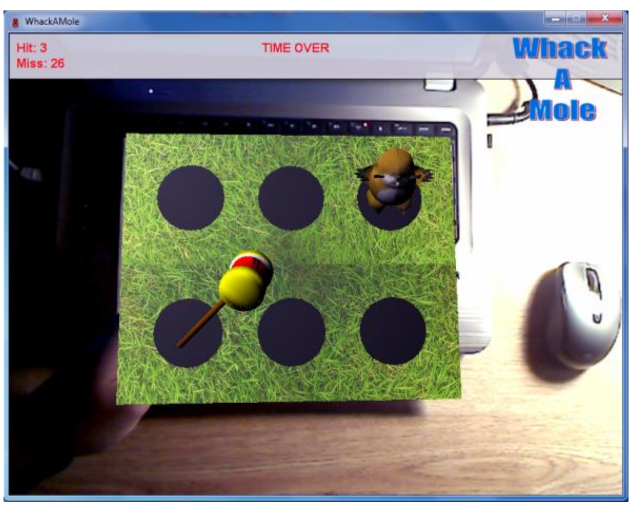


Figure 8. *Whack a Mole*: screen shot of WhackAMole game.

III.6 - Dodger

Another game we created is called Dodger. This game works by having the player's piece dodge other falling pieces in the game area. This game is very simple with many adjustable features. The adjustability of this game makes it excellent for rehabilitation exercises.

The premise of the game is simple; you are the red box and you must move out of the way of falling blue boxes. You have three lives. Each time a box collides, you lose a life. This game is excellent because the control we can have over each level. The amount of blue boxes is adjustable from one to as many as is wanted, the speed at which the boxes fall and their size is adjustable. The game has a scoring system that allows us to track a patient's progress, the longer a patient stays in the game the more points they acquire. This allows the doctors to track how the patient is doing and adjust the difficulty of the game through any of the options in the game. At the end of each game the user is presented with a high score screen this will show the users top 10 scores. This high score screen allows the patient to see their progress as well and gives them a metric by which to compare their own progress.

We wanted to begin to add more graphics to the game, so we began by adding a stylish background (Figure 9). But beyond simple graphics we just did not have enough time. The game also has a pause menu allowing the user to stop if they want a break. The game also detects whether or not you are actually in the play area and will pause until you re-enter in the game area, so that you cannot cheat by sitting on the side. This game is another perfect example of something simple that a doctor can easily adjust and can be made for varying difficulties of different patients.

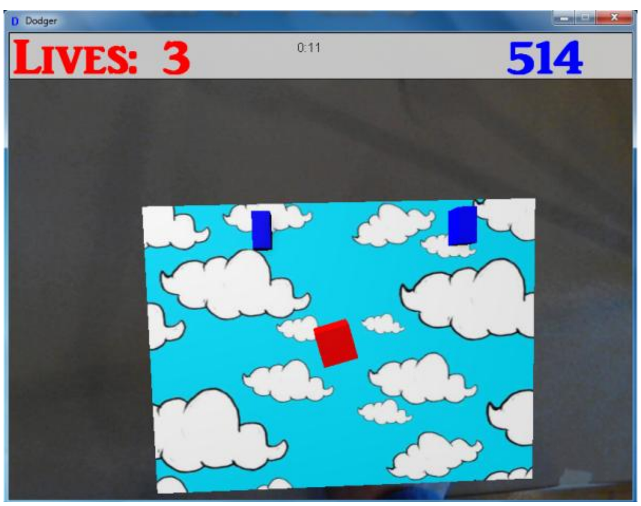


Figure 9. *Dodger*: screen shot of Dodger game. The Player is the red box trying to dodge the blue boxes.

Chapter IV – TECHNICAL PROBLEMS

We have encountered some technical problems throughout the developments of the AR Games for Rehabilitation framework. One problem was with alignment of 3D boxes with real objects. They appeared to move down in Z dimension when they got further away from camera, even though the object stayed on a flat surface during the game play. We solved this problem by setting the Z value of transformation matrix to zero.

One other problem arose when we started developing the set-up game area functionality. Getting the profile of the table and keeping the adjustable game area on this profile seemed to be a harder job than we initially thought. It appears that it is related with the first problem. When the user moves the small marker to set-up the game area, the area increases as it is expected in X dimension. However, moving the marker in Y direction further away moves the game area down, and moving the marker closer to camera in Y direction moves the game area up. This seems to be due to the set-up of the camera. If the camera does not stand right above the desk, and faces the desk from the user point of view, moving the object in Y direction actually moves the object further away from the camera, thus changing its Z value too. This issue has not been solved completely yet, but is being worked on.

Another problem we have encountered is to do with the game state management, the menu system. The games kept crashing seemingly randomly when trying to load a second time. Not freeing the resources of the previous game seems to be contributing to this problem, but the exact cause has not been found. Another crash happens when trying to load a music file. This issue has been reported many times on the XNA forums, and it is believed that a bug in XNA is causing this crash type. The crashes associated with the menu system (game state management) became very constant after adding other screens (pause menu, options menu in each game etc.). After dealing with it for almost a month and not getting anywhere, we decided it is best to remove the menu system out of the framework.

The placement of camera is an important key to provide a meaningful play. It either needs to positioned with the same point of view of the user, or placed right above the game area as in Figure 3. The first option causes some problems with the positioning of 3D objects on the screen as mentioned before. Once these issues are solved, this position of the camera will be the best to use. Placing the camera right above the desk on the other hand does not give a good perspective of depth. Placing the camera opposite side of the user view, results in mirror image of the game, and as a result makes the game impossible to play. Moreover placing the camera in any other position does not give any intuition to the game play, making the game play very difficult. Perhaps, these issues can be solved by AR goggles, in contrast making the system much more costly.

The final difficulty we ran into was with the design of 3D objects. As of now, we are either using simple 3D objects like boxes and spheres or previously created free-of-charge 3D objects. The

3D design is another field of research and we decided that we can't devote our time to creation of our own 3D objects. We experimented with Blender, and found that even creating simple objects takes excessive amount of time, consequently we decided to use already available 3D models.

Chapter V – DESIGN DECISIONS AND ALTERNATIVE APPROACHES

Our project was a continuation project from a previous year. We looked at their code and what they had accomplished after a year and decided to try a different approach. Previous team had spent most of their time attempting to master object recognition and tracking, their final product was good but not perfect. So at the beginning of the year we decided to begin exploring other alternatives. Our main concern with previous years work was that in using large colored globes it prevented any user who was not able to hold them from using the system. This is when we struck upon the idea of AR tags. These tags are generated using the ALVAR tracking library[6] and are just printed out on any paper and thus can be attached to anything. This gave us the freedom that if a user is working on strength building it can be taped to a dumbbell. But if a user cannot control their fingers or hand we can apply it to the back of their hand. Both approaches have no affect on the game but with the tags our audience that can use our software grows.

When looking for software to use we found Goblin XNA[4] which uses Microsoft XNA Game Studio[5] and ALVAR tracking library[6] to create a good framework for AR games. This was chosen of other alternatives mainly because of the inclusion of XNA Game Studio. XNA Game Studio has the majority of the methods we would need for creating any game. These include game timers and all the methods for creating 3D objects and manipulating them.

ALVAR tracking library was chosen primarily because Goblin XNA uses it without modification. But also because after trying out several other libraries we found it worked best with the XNA framework. Minimizing the work we had to do to get started. We also looked at using something like Kinect or trying to track just a hand or foot. If we used the Kinect it created a price barrier for our system. And we were unable to find a system that could accurately track hands or fingers without needing a lot of CPU power behind it. This would create another barrier needing a very powerful system to run anything we created. We also found that while these systems were able to track hands or fingers they were not able to accurately render a 3D object nearly as well as something like using AR tags.

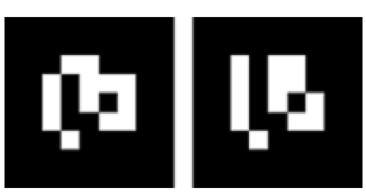


Figure 10. AR Tag: Used in our games. It is was generated using ALVAR.

We briefly looked at integrating hardware into our system. Using hardware would potentially allow increased tracking and the ability to track patients vitals. While this would be good, one of the decisions we came up with early on is, we wanted a cheap and simple system that anyone could use. Most hardware systems cost quite a bit of money and setup can quickly become complicated. These barriers can dissuade users from using our system and as such we decided to stay away if at all possible. This is not to say it is not a possibility but if we add any hardware it will not be necessary but rather improve some facet of the games. But keep the games standalone only enhance with extra hardware.

Another choice we had to make was about how to orient the camera. We tested several different ways from the side on top and lots of different distances. What seemed to work the best, meaning 3D rendered objects aligned with their real world tags, was having the camera directly above the play field. The height we found has more to do with the user than with the actually height but most of our runs were at most 2 feet above (Figure 3). The issues come in when you are too far the camera cannot focus on the markers. This would be very apparent as the game would not recognize any markers. This is easily solved by moving the camera closer or obtaining a higher resolution camera.

Most decisions for this project were based upon a simple idea of making the design and setup easy. We wanted to use technology that already existed and integrate it all together creating a suite of games that are easily adjustable and track progress. All our decisions were based on this and should be apparent from the path we took. We believe we achieved what we set out to do though there is a more to do.

Chapter VI – PRODUCT MARKETING

Our product was initially meant for any users. We decided near the end that we should focus more on children. This idea was presented us by Dr. Malcolm in Department of Occupational Therapy. To that end we began making our games flashier so that we can appeal to children. All our games are also very simple so younger children will not have any trouble. All our games are intuitive you do not need instructions for how to play them, thus making it even better for children. By using games the hope is that people will enjoy their rehabilitation rather then it just simply being tedious and boring.

Our primary market is the doctors seeking to help their patients. The software would allow them to help patient's rehabilitation and allow them to monitor their use and progress. But this is not the only market if you are in the process of upper limb rehabilitation this software is of great use to you as well. What it comes down to is anyone who needs any kind of upper limb rehabilitation will benefit from this software.

Chapter VII – ETHICS

Our primary ethical dilemma in designing our games is we are using several different software libraries that others have created. Some of these are under a general public license others are free for non-commercial use. Now under the ASME code of ethics "Engineers shall respect the proprietary information and intellectual property rights of others, including charitable organizations and professional societies in the engineering field."[9] Clearly this is of great concern to us, which is why we make sure that we properly citing all of our sources.

Another potential issue we have to be aware of is that this data can be considered medical data. To that effect we are looking at encrypting the data outputted from the games. Patients medical information is and should be highly confidential. The National Society of Professional Engineers says "Engineers shall not disclose, without consent, confidential information concerning the business affairs or technical processes of any present or former client or employer, or public body on which they serve."[8] So all medical data gathered by the games needs to be protected as to protect the patients.

Toward the end of our project we were able to meet with Dr. Malcolm from CSU's Occupational Therapy department and were able to come up with some compromises to allow this to be used in a medical environment without serious altering. We decided that all the high scores will have nothing showing who got what score. That way they are all anonymous. We began working on getting patient testing going as well. We altered our games according to what the Occupational Therapy Department thought would be better and began getting things set for use in real medical trials. Unfortunately due to scheduling conflict between research subjects and the Occupational Therapy Department, we were not able to get and patient testing done, but our project is at the phase where that is the next step.

Chapter VIII – CONCLUSIONS AND FUTURE WORK

Our primary goal was to develop a cheap and user friendly suite of games for upper limb rehabilitation. To this affect we looked at previous years work as well as lots of work that other people had done in this field. From them we based on where our system would start.

VIII.1 – Conclusions

We based our work on what previous work had already done. This allowed us to skip a lot of the development stages of some of the more difficult parts, such as the AR tracking library which we utilized an existing library ALVAR[6]. This allowed us to begin on almost day one creating games. Our main challenges when creating these games came from our lack of knowledge in 3d graphic design. We overcame this slowly through building these games. When creating pong we started wanting to make a simple game in two dimensions. Our main challenge was overcoming using multiple tags interacting together. This hinged on using the physics engine to tell the program if any of the 3d objects have collided. The follow the leader game was our attempt at beginning to create a game that is almost completely adjustable. Follow the leader allowed us to have a completely adjustable game area that began scaling up as the users' progress through more difficult levels. It also allowed us to create a loadable level that doctors could set the pattern the player must follow. In the second semester we worked on two more games: Whack a Mole and Dodger. Toward the end we tried to work with the Occupational Therapy Department to get patient to test our games. However, we were not be able to carry out any testing.

Our goal in our games was to create games simple enough for someone with severe injury but with enough scaling that even fully capable people would have a challenge. All our games have adjustable game play areas as well as scaling difficulties. The last thing that we felt all our games needed was the ability for doctors to track their patients progress. Our games have scores and high score files that will save patients scores. The scores scale with difficulty so harder game more points through this we can see progress in patients by simply looking at the high scores.

Overall this project has given a glimpse in to the difficulty that people going through rehabilitation have. We had to design our games in such a way that we found overly simple but still allowing for increased difficulty. We learned a lot about 3d graphic and video game design, as well as the capabilities of people with upper limb disabilities. We set out to build several games that could be used by anyone, and we think we achieved that goal.

VIII.2 – Future Work

The next major step for this project is to begin patient trials. The knowledge that would be gained from patient testing is irreplaceable.

We attempted to make a menu system to link all the games together and was unsuccessful due to time constraints that we allowed for debugging. Adding this feature back into the game is very important. Having a complex interface might intimidate some patients and might not be very easy for patients to get used it.

Eventually there will be a need to add in a feature to allow remote data uploading from patient's computer to his/her doctor's. This will allow a doctor to setup the games at the office for a patient, then send them home and receive daily or weekly details on each patient.

REFERENCES

[1] National Stroke Association. Stroke 101 [Online], Available: http://www.stroke.org/site/DocServer/STROKE101_2009.pdf?docID=4541, 2009

[2] When was the augmented reality invented? [Online], Available: http://www.vrs.org.uk/augmented-reality/invention.html, 2009

[3] Game State Management Sample. Available: http://create.msdn.com/en-US/education/catalog/sample/game_state_management, May 2011.

[4] Goblin XNA, Available: http://goblinxna.codeplex.com, November 2011

[5] Microsoft XNA Game Studio, Available: http://msdn.microsoft.com/en-us/aa937791 , November 2011

[6] ALVAR, Available: http://virtual.vtt.fi/virtual/proj2/multimedia/alvar.html, November 2011

[7] Newton Physics Engine, Available: http://newtondynamics.com/forum/newton.php

[8] National Society of Professional Engineers. NSPE Code of Ethics for Engineers [Online], Available: http://www.nspe.org/Ethics/CodeofEthics/index.html

[9] American Society of Mechanical Engineers. CODE OF ETHICS OF ENGINEERS [Online], Available: http://files.asme.org/ASMEORG/Governance/3675.pdf

[10] Vuzix Wrap 920AR. Available: http://www.vuzix.com/consumer/products_wrap920ar.html

BIBLIOGRAPHY

Akira Sano. An Application for Creating Full-scale Augmented Reality Content without 3D Modeling Skills, 2011

J. W. Burke, M. D J. McNeill, D. K. Charles, P. J. Morrow. Serious Games for Upper Limb Rehabilitation Following Stroke, 2009

J. W. Burke, M. D J. McNeill, D. K. Charles, P. J. Morrow. Augmented Reality Games for Upper-Limb Stroke Rehabilitation, 2010

Ohan Oda, Colin MacAllister, Steven K. Feiner. Goblin XNA User Manual. Version 3.6.0, July 2011

Appendix A - GLOSSARY

AR – **Augmented Reality**. Is a live, direct or indirect, view of a physical, real-world environment whose elements are augmented by computer-generated sensory input such as sound, video, graphics or GPS data.

AR Goggles – A head-set containing a display and a camera to enhance the AR experience.

ARToolKit – A computer tracking library for creation of strong augmented reality applications that overlay virtual imagery on the real world.

Blender – A free and open-source 3D computer graphics software product used for creating animated films, visual effects, interactive 3D applications or video games.

Cerebral Palsy – A group of disorders that can involve brain and nervous system functions such as movement, learning, hearing, seeing, and thinking.

C# – A general purpose object-oriented programming language made by Microsoft. C# is part of the .NET framework and has syntax very similar to Java.

FTL – **Follow The Leader**. The second game developed for our project.

GUI – **Graphical User Interface**. Type of user interface that allows users to interact with electronic devices with images rather than text commands.

Kinect – A motion sensing input device by Microsoft for the Xbox 360 video game console.

Pong – One of the earliest arcade video games, and is a tennis sports game featuring simple two-dimensional graphics.

Rehab – **Rehabilitation**. Methods for retraining neural pathways or training new neural pathways to regain or improve neurocognitive functioning that has been diminished by disease or traumatic injury.

Stroke – Rapid loss of brain function(s) due to disturbance in the blood supply to the brain.

TBI – **Traumatic Brain Injury**. Occurs when an external force traumatically injures the brain.

Upper-Limb – The region in a human extending from the deltoid region to the hand, including the arm, axilla and shoulder.

Visual Studio – An integrated development environment (IDE) from Microsoft that is used to develop console and graphical user interface applications.

Appendix B – BUDGET

As of the end of second semester, no significant amount of money has spent on the project. We used everyday computers and webcams along with free software libraries and platforms. No electronic equipment was necessary to purchase. Additionally, we had the equipment used in last year's project. Ideally, we wanted to make the system available to everyone including who have limited budgets. Anyone who has a pc and a webcam can benefit the augment reality games for rehabilitation. Only place where we spent money was to buy a really cheap tripod to place our camera on to it. A tripod can be purchased for less than 10 dollars at most of electronic shops or large retail stores.

The games consume rather large amount of CPU power. Although our laptops have been sufficient at this point, if we had additional funding, we might have purchased a high performance laptop for demos of the games. Even though, a simple webcam is good enough for our framework, AR goggles such as Vuzix Wrap 920AR+ [10] would make the games easier to play. They cost around $1500, and they are well out of our budget provided by ECE department. They are on the list of possible items that can be bought with additional funding.

ACKNOWLEDGMENTS

Dr. Sudeep Pasricha – Our advisor who met with us every week and pushed us to do more than the bare minimum for our project. He also gave useful advice for new ideas in developing the AR games.

Olivera Notaros – Our advisor who helped facilitate communication with outside parties and gave a fresh perspective on our design.

Dr. Matthew P. Malcolm – Our advisor who helped point us in the proper direction when making games for medical needs.

Goblin XNA – For developing a framework that help making AR games relatively simple.

XNA Game Studio – For creating a framework that makes 3D graphics and games much simpler.

ALVAR – For creating a robust tracking library offered free of charge.

Newton Physics – For creating a physics engine free of charge.

Google 3D Warehouse - For providing free 3D models.

Freesound.org - For providing free sound effects.

2012 - Engineering Days Senior Design Showcase

3rd Place Winner