Project Number: MLC GP05

The Impact of Frame Rate and Resolution on Player Movement in First-Person Shooters

An Interactive Qualifying Project Report
submitted to the Faculty of
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
Degree of Bachelor of Science
by

____
Tim Connor

____
Adam Fiske

____
Ryan Kennedy

Date: April 30, 2006

Approved:

____
Professor Mark Claypool, Advisor

Abstract

The effects of frame rate and resolution on users' perception of digital media are a growing concern. This paper looks at the effects of these two factors on users' performance in movement related tasks in first person shooter games. In a user study, participants played several custom maps in Quake 3 Arena at different frame rates and resolutions and their performance was measured. It was found that lower frame rates and resolutions lowered users' overall performance in the video game.

TABLE OF CONTENTS

1. Introduction

Today's digital technologies are increasing at a rapid rate, and consumers are trying to keep up with these changes. With this growth comes a multitude of different media for consumers to use. These users want to experience media in the best form possible, but this is not always feasible due to the fact that not everyone can afford the best hardware. Producers of this media want to be able to supply their consumers with the highest possible quality regardless of the hardware levels in order to increase user satisfaction and hence profits. Broadly, digital media comes in two forms: "passive" media, which users simply watch such as video, and "active" media, which users interact with.

A popular active medium for consumers and developers is video games. Earlier studies have examined how factors such as frame rate and resolution affect users' perception in video games. A game genre which relies heavily on frame rate and resolution for playability is first-person shooters. Higher frame rates generate smooth transitions between rendered images on the screen, making the visuals more believable and provide smoother game play and better performance. Similarly, higher resolutions are desired for their increase in detail and their ability to show the user more of what is on the screen. Most users want to maximize both of these settings, but this is not always possible with limitations in hardware. Higher resolutions often diminish frame rate, while lower resolutions lead to greater frame rates. Developers and users struggle to find the best display settings for their video games.

Past research on the effects of resolution and frame rate in video games has primarily been restricted to the effects of display settings on shooting accuracy in first person shooters, with frame rate affecting accuracy significantly. The higher the frame rate, the higher the accuracy of the player to a finite point. Also, in general, the effects of different resolutions on the accuracy of the player were not so dramatic, if effective at all. In the realm of passive media, the effects of frame rate and resolution on users' perceptions of streaming video have also been studied in great detail. In this medium, users comprehend what they are seeing better at higher resolutions, while frame rate can be very low and users will still be able to understand all the information in the video.

Movement and shooting/accuracy are the two most important skills in first-person shooters, as is evident in the phrase “run and gun,” which often refers to the first-person shooter genre. Movement, being a very crucial game play element, is arguably the most important. Users walk, run, and jump constantly throughout their game play experience. Studies have not been done relating movement in first person shooters to frame rate and resolution, especially on the ability to navigate in 3-D space. This study looks at this element in particular in order to understand the impact of frame rate and resolution and provide information for users and developers alike to use when making trade-offs.

Four maps were created for Quake 3 Arena, a typical first-person shooter. These maps tested four different aspects of movement: walking, running, jumping, and recognition. The effects of frame-rate were studied on the walking, running, and jumping maps; and the effects of resolution were studied on the recognition map. Thirty-six users took part in the study, thirty-one of which were students or faculty from Worcester Polytechnic Institute. While testing the participants, data was gathered on how well the participants performed on each map with varying frame rates and resolution. After the study, this data was analyzed.

This analysis brought two conclusions. In the maps testing walking, running, and jumping, as the frame rate lowers from 15 fps to 7 fps to 3 fps, so does a user's performance. Also, an increase in frame rate does not necessarily result in better movement, but rather that a user can simply complete movement tasks quicker than at lower frame rates. In the maps testing recognition, lower resolutions lead to worse performance by the users, specifically at resolutions lower than 640x480.

The rest of this paper discusses previous work done in this area, and how this particular study was run to obtain the results summarized above. Section 2, Background, details the previous work done. Section 3, Methodology, describes how the study was devised, designed, and run and how all of the data was gathered. Section 4, Analysis, dissects the gathered data to make sense of it and details all valid information discovered in the study. Section 5, Conclusions, discusses all the information in a more general manner with concrete conclusions drawn and also discusses this study's contributions to the field of media research. Lastly, Section 5 also comments on future work that can be done to improve this study and studies implemented after it.

2. Background

In the area of the effects of frame rate and resolution on user perception of media, two main areas have been studied: passive and active media. Passive media are watched by the user and consist of movies and television. Active media are interacted with by users and consist of video games and computer software.

2.1 Passive Media

In a study by Sasse (2004), users watched a soccer game at different frame rates and resolutions. It was found that with frame rates as low as 6 frames per second the users still found the quality acceptable 80% of the time. In general, users tended to favor higher resolutions over greater frame rate to comprehend the information being presented to them. In another study (Tripathi, 2002) a system was created to decide whether frame rate or resolution was more important at any given time in a streaming video. This system detected movement in the each section of the video and scaled the quality of the stream to increase either frame rate, if there was a lot of motion, or resolution, if there was not. It was found that optimization under its guidelines improved perceived video quality by 50 percent. Of course, these two studies can only be applied to the passive portion of the tests described in this paper. If a user is performing poorly at a particular movement related task due to low frame rate or resolution, they will be able to perceive that this is the case, but how they can react in these low settings situations will be left up to frame rate's and resolution's effects in active media.

2.2 Active Media

A study by Bryson (1993) investigated the effects of lag and frame rate on tracking tasks.

One involved placing a cursor on a specified place on the screen as quickly as possible. The other task involved moving the cursor to keep it as near as possible to a target moving randomly about the screen. Both of these tasks were performed at frame rates of 2, 3, 4, 6, 10, 20, 30, and 60, with the result being that users performed better with higher frame rates. Tracking tasks are very similar to accuracy tasks in first person shooters.

A study by Reddy (1997) looks at users' ability to determine their direction in relation to a fixed object that they must track on the screen. The user had a simulated motion moving to the right or the left of the fixed position; their task was to decide the direction of the motion in the shortest amount of time. Frame rate (fps) and angle of motion were varied, setting the frame rates at small intervals between less than 1fps and 32fps; the resolution was fixed at 1280x1024. The angle of view had a fairly large impact on choosing the correct direction, but it does not relate to our study. Reddy's findings indicate that correctness and time degraded at low frame rates (between 2.3 and 11.5fps). Performance as a measure of time and correctness increased rapidly as frame rate reached 15fps. Past 15fps performance continued to increase, but at a much slower pace. This pertains to our experiment as movement at low frame rate is the key element we are studying; the ability to determine the direction one is moving is important. In this experiment the user does not control the movement, however we will be examining the effect of frame rate and resolution on user controlled movement.

Finally, a study (Claypool, 2006) used Quake III arena to test the effects of frame rate and resolution on shooting related tasks. Participants played some user-created maps and the number of kills each user achieved at different frame rates and resolutions were recorded. In particular, it used frame rates of 3, 5, 7, 15, 30, and 60 and resolutions of 320x240, 512x384, and 640x480. Results and found that as frame rate increased, so did number of kills, but resolution made little difference in user performance. Much of the methodology of our study was based on Claypool's study, such as the frame rates and resolutions used, and the game tested. We hope that in the future, these two papers can be looked at together, as they each study a particular half of the “run and gun” tactics in first-person shooters.

3. Methodology

This section will cover aspects of test development as well as our testing methods. Important topics covered are the development of movement related tasks, and our testing procedure. Subsections include: finding a game, test parameters, test maps, the game harness, and the final setup and procedure of the test.

3.1 Game Choice

The first step in beginning the study was finding a suitable game to test. There are a wide variety of first person shooters to choose from, and finding the most appropriate game for our study was a top priority. While choosing the latest and most popular would be an ideal solution, it also had to fit with the study rationale and testing capabilities. Some of the games considered were: Half-Life 2, Doom 3, Unreal 2003/2004, Quake III Arena, and Quake 4.

Whatever the final choice became, it needed to be able to change video setting parameters in some form of command line. This was needed in order to change video settings through a scripting mechanism; it would allow the game to be modified easier than in-game menus. A second consideration in finding a game was previous studies, and how the established conclusions of these studies could be built upon. Most games are too recent to have been the subject of studies, with the exception of Quake III and Unreal. The study using Quake III is discussed in Section 2.2; a study using Unreal 2003 (Beigbeder et al, 2002) examined lag and latency and their effect on movement. For these reasons, there were two possibilities: Unreal 2003/2004 (2003 and 2004 games are similar game-play wise, while 2004 adds vehicles), and Quake III Arena. After testing both Quake III and Unreal and searching online documentation, it was soon discovered that it was impossible to change the frame rate of Unreal 2003 through the command line. Quake III did have a command which allowed the maximum frame rate to be controlled. The rest of the games had no command to change frame rate. While Quake III is a fairly old game to be using in a study (released in 1999), it could perform the desired functions. Quake III is a typical first person shooter, with a focus on multiplayer not single player. At the time of its release, the graphics engine was brand new. Both games had related studies; however, the study using Quake III (Claypool, 2006) was more directly related to our intent, examining the effects of frame rate and resolution in first-person shooters. After reviewing these factors, the game chosen was Quake III based on the facts that frame rate could be easily manipulated, and a comparison to a previous study (Claypool, 2006) could be made.

3.2 Basis of Study

Once Quake 3 was decided upon as the game that would be tested, the exact basis of the study had to be decided. The previous study that had used Quake 3 already had studied the effects of frame-rate and resolution on shooting (Claypool 2006). It was decided to study something different than shooting so as not just repeat the same study again, but rather complete the study of interaction in first person shooters. The other aspect of first person shooters that was not covered previously is movement. Moving and shooting are integral parts of first person shooters. Three distinct types of movement were established as being important in first person shooters: walking, running, and jumping (specifically navigating while walking and running). Through initial pilot studies, resolution seemed not to affect these movement types substantially. A visual identification aspect was then introduced, because it is also important to first person shooters (targeting and various other visual cues) and it was related to resolution. Therefore, the study consisted of three movement tests, and a visual identification test. The following section details how maps were designed to fit these tests.

3.3 Map Development

After deciding on what game to use and what was to be studied, a testing system had to be developed within the game that would measure performance. The first part of this system was the map(s) to be used for the test; the other parts are described later in this chapter. Based on the movement types and recognition test that were determined previously, four maps were envisioned: a map that would test turning movements while walking; a map that would test these same movements but simply at a running speed twice that of the walking; a map that would test only jumping movements at a walking speed; and a map that would test path-finding skills. The first three maps were designed to test the effects lower frame-rates have on players' performance. The fourth map was developed as a way to test the effects lower resolutions have on players' performance.

For this first part, an open source Quake III map editor was employed: Q3Radiant. This particular editor was chosen because it was the official map editor id software used when developing Quake 3. Therefore Q3Radiant allowed for all possible map options to be implemented (if desired), customizable views, and even included a fully detailed editor manual available online. It should be noted that this manual was often needed as the interface for the editor was not immediately intuitive or user-friendly, as it was designed mainly to be used by its own developers.

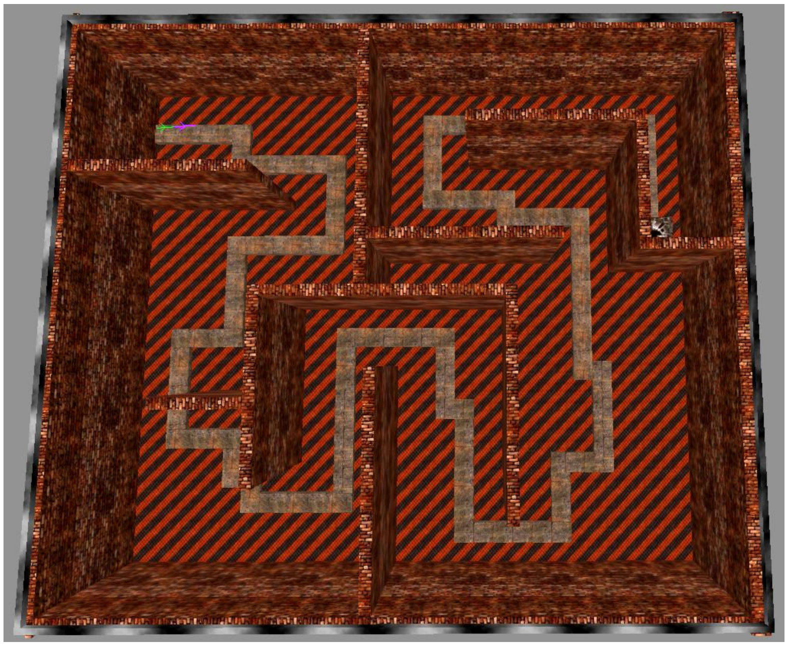


Figure 3.1

The first map designed was the simple walking map where the player moved at the game's default speed. Figure 3.1 shows a top-down view of the map; Figure 3.2 is a screen capture of the map. A pathway was built above a lava-like surface. It was built in such a way that it was impossible to skip sections by jumping over them. Walls separated sections of the path and sharp corners were minimized. The surface below the pathway was termed “invisible lava,” a substance that when stepped on would reduce the player's health by 5% every second. Originally, the plan was to use the actual lava available in the Quake 3 game, but it was found that this lava reduced the player's health by 30% every second. Such a large decrement in health would only allow the player to step on this lava a maximum of four times before dying, greatly increasing the variability of their performance.

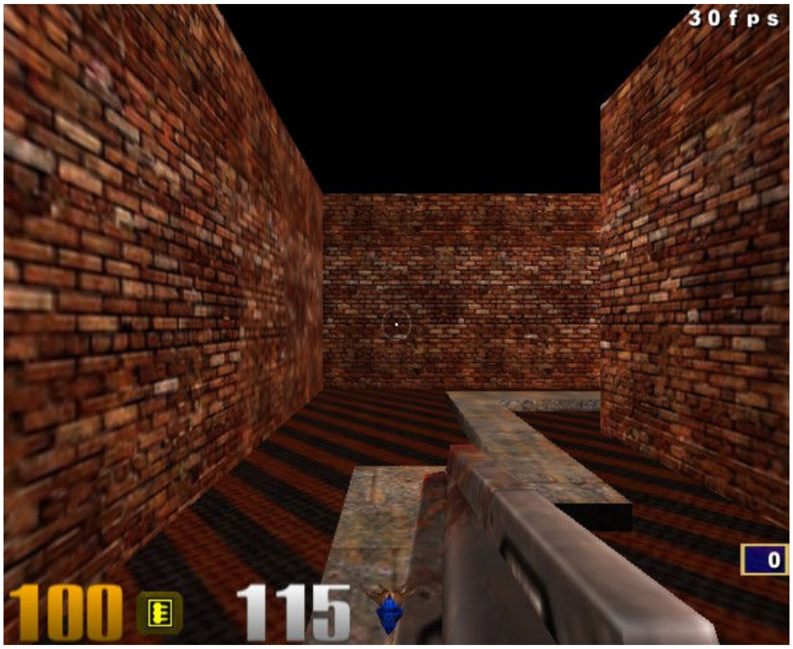


Figure 3.2

The second map, dubbed the “haste” map, was essentially the same at the walking map, except the player moved at twice the default running speed. This was accomplished by starting the player off with a “haste” artifact that doubled their movement speed.

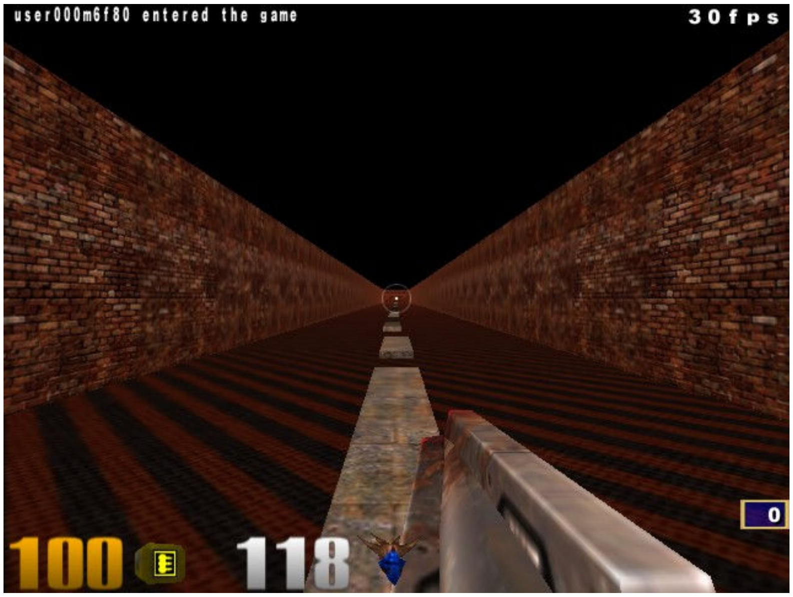


Figure 3.3

The third map produced was for testing jumping movements. Figure 3.3 shows a screen capture of the jumping map. This map was created by taking the walking/haste map and turning all of the corners of the path into gaps. The original snaking path of the walking/haste map became straight. Invisible lava still remained underneath the pathway and the player still received 5% health damage per second while standing on the caustic surface.

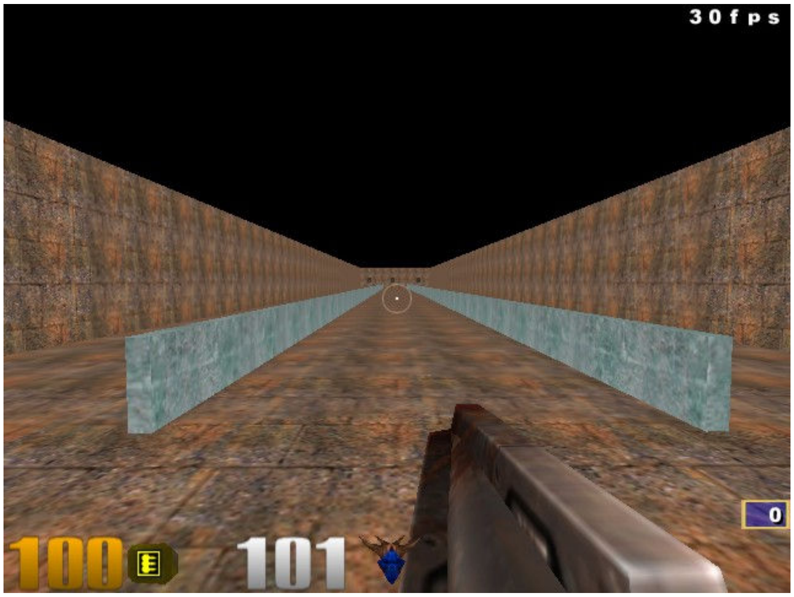


Figure 3.4

The fourth and final map was slightly different from the other three. This map only served to test the player's recognition skills. The map consisted of three hallways in front of which the player would start, as shown in Figure 3.4. The end of each hallway was clearly visible without moving from the initial position. At the end of each of these three hallways there was a button. One of the three buttons was different than the other two and it was the player's objective to move their player to the one that was different. When the correct one was reached, the map ended. The position of this different button was randomized at the end of a different hall on each loading of the map. Three versions of the map were used, each having the button at the end of a different hall. Refer to Appendix A for top-down views of the third and fourth maps.

3.4 Test Parameter Development

It was decided to use the same frame-rates and resolutions as the previous study, so results of both this new study and that study would be comparable and could possibly be looked on as a whole. The frame-rates were 15 fps (frames-per-second), 7 fps, and 3 fps. The resolutions were 640 x 480, 512 x 384, and 320 x 240. The frame-rates are all roughly half of each other, but the resolutions do not encompass the same linearity. A fourth resolution exists between 512 x 384 and 320 x 240: 400 x 300. This resolution was left out for two main reasons. The first is that it was simply not included in the previous study on shooting and the desire was to keep the two studies as similar as possible. The second was that with four sets of parameters, it would require the participants to sit through more maps. This resolution was between the bottom two tested resolutions and not the top two tested resolutions. It was found through test runs of Quake 3 the difference in visual detection happened between these two higher resolutions and therefore the missing 400 x 300 resolution would not detract from any conclusions about a user's performance through lower resolutions.

It was also determined that there would be some default, or “control” settings that would be used for getting the participants accustomed to the maps. For maps that would be testing frame-rate, the control resolution would be 800 x 600, as that was the game's default, the resolution Quake 3's developers predicted would be run-able on all users' computers. This would also be used for all variations of the frame-rate maps, as it was undesirable to have resolution affecting the participants' performance on maps that were only testing frame-rate performance. For the maps that would be testing resolution, the control resolution chosen was 640 x 480, as this was as high as the experimenters were testing. The control frame-rate used for all maps was to be 30 fps. This frame-rate was found through test runs of the game to not detract from any performance, and for this reason it would also be used on all variations of the resolution testing maps.

Finally, it was decided that a user's performance would be based on how much health they lost in a map and the amount of time it took them to complete said map. The exact performance rating was not able to be precisely developed at this early time in the testing, as it was unknown just how the users would actually perform on the tests. The time it took them to complete each map might never change, and that part of the performance rating would need to be dropped, or the alternate could happen where their loss in health never changed. Maybe one would need to be weighted higher than the other, because one might end up mattering far more than the other. At this point it could only be said that the final performance rating would only be derived from the user's loss in health and/or total time per map.

3.5 The Test Harness

A test harness was needed to automate the maps that were created. Both a manual harness and an automated harness were suggested and each idea had their own merits. A manual harness means that each map would be run individually, by the tester or subject. On one hand, the manual harness allowed for complete control over all aspects of the testing procedure; the experimenter could give extra time explaining what the participant needed to do and make sure everything was at the proper settings. However, a manual harness left room for more human error, would take more time, and would not be run the same every time through, creating more variability. An automated test harness would fully control the users' experience. The user would initiate the harness, and it would run the entire process. The advantage to automation is that the test would be run exactly the same each time. This leaves less room for error. The tester would not have to be involved at all. However, an automated harness could not account for changes to the test per user (user errors, or unexpected problems). In the end, it was decided that a mix of the two types of test harnesses would be best. The reasons leading to this decision are detailed below.

It was first decided that a completely automated harness would be used to run the testing process. Yet it was still intended to give out surveys in between each map to assess the players reactions to the map they just played. However this would not be possible to pause in between maps without an extensive amount of coding to be written for the harness. This was impractical given the goal of the project and it was decided that a simple demographics survey would be given before the test phase began to collect all the relevant data needed. No user feedback would be gathered between maps.

The next step in the user study was to record all data of the participant playing the game. The program FRAPS was chosen to do this (FRAPS, http://www.fraps.com, Beepa, v2.7.2). FRAPS is a Windows application compatible with OpenGL or Direct3D technology that can be used to take screenshots, record in-game video, and benchmark computer games performance.

Once the data recording procedure was decided upon, it was included in a harness. A program was written in Java that would randomly generate a .bat (batch) file with all the appropriate commands to run each map at its different settings (see Appendix B). At first writing, this program proved difficult. It was the original intent to have this program not only run all the maps, but also have FRAPS record a video of each map. However, it was found to be difficult to write such functionality into the Java program without the code being overly verbose and time-consuming to write. It was decided that this program would simply create the batch file of appropriate maps in random order and run it. First, it ran four non-random control maps (one of each basic map) at a maximum resolution and frame-rate. This was done in order to acclimate the participants with each of the maps so they would not be confused about what they were supposed to do and reduce learning curves in the data gathered. After these four initial maps, the remaining maps were loaded in random order with frame-rates and resolutions as described earlier.

To fix the FRAPS problem of having no easy way to integrate automatic recording into the Java program, it was decided that this portion of the test harness would be manually automated by the experimenters. The test administrator would begin FRAPS, recording a video when the map was loaded and would manually quit Quake 3 when the participant completed the map. FRAPS automatically ended recording of the video when the game quit. Also, for further recording purposes, it was decided that not only would FRAPS record a video of the participant's performance, but it would also record a benchmark log of their test. This log recorded the time it took the user in milliseconds; the minimum, maximum, and average frame-rate of the session; and the resolution at which the participant played.

A second smaller problem that was encountered during the debugging phase was that Quake would accept new frame-rates on an immediate load of the game, but it would not accept a new resolution until the second load of the game. Once this problem was discovered, the code was modified to load any new resolution on the previous load of the game so that when the new resolution was desired it would be loaded correctly.

In the end, the final game harness was as automated as it was possible to make it, while still requiring some manual control. This allowed for a minimal amount of human error, but still allowed the experimenters to have control over the tests, just in case the automated harness had any problems, the test PC crashed, or any other unforeseen problems occurred.

3.6 Testing Setup and Procedure

To carry out the tests, a suitable computer was needed for not only running Quake III, but also to capture the FRAPS recordings. Therefore, a computer was built to adequately handle both tasks simultaneously. The system specifications are as follows:

- CPU: AMD Athlon 64 3700+
- Memory: 2 GB
- Video Card: nVidia GeForce 6800 (512MB Video memory)
- Hard drive: 300 GB
- Monitors: 17” CRT
- Sound: Integrated audio chip, Creative 2.1 Stereo speakers
- OS: Windows XP SP2
- Quake III Arena: Version 1.3
- FRAPS: Version 2.7.2
- Accessories: Logitech optical wired mouse, keyboards

The computer was set up in a private room for the tests. In order to observe the user and to start/stop the FRAPS recording manually, two monitors and two keyboards were used. The user's space was arranged as a normal computer, however a second keyboard and monitor were placed behind a divider, away from the user. These allowed for supervision of the user and execution of the necessary actions to record the data, and exit the game. The user was unaware of the observation setup during the test.

For this study, users were solicited in a variety of ways. First, test subjects were found through friends and classmates. Faculty and staff in the Computer Science department also participated. Additional users were procured through CS courses, where students were encouraged by the faculty to take part in the study. All users were entered into a raffle for a $50 gift certificate.

To begin the test, the user was given an instruction sheet and a demographics survey (see Appendix C). The demographic survey was anonymous. The user was assigned a specific test number to attach the survey data with the computer data. The instruction sheet gave general directions on filing out the survey, and how to open the test program on the computer. The sheet also contained detailed information about the test maps, and the objectives in each one. Once the directions were read, the users were allowed to ask questions relating to the maps. The user then started the test program. During the test, if users became confused about the objectives of the map, the test conductor would remind the participant of the directions, but would give no explicit hints or suggestions for the map. Users were not informed beforehand that the tests involved changing frame rates and resolutions, only that different video settings would be used in each map.

The test program consisted of sixteen maps, four of each map type at the various settings: walking, jumping, haste, and resolution. The first four maps the user encountered in order were four control maps. These first four were walking, jumping, and haste at 30 frames per second and 800x600, and resolution at 30 frames per second and 640x480. The purpose of these maps was twofold: (1) to acquaint the user to the maps and objectives, and (2) to establish an unmodified game setting for control data. Following the first four, the program loaded at random (as discussed previously) the 12 other test maps which consisted of each map type at the various frame rates or resolutions.

At the start of each map, the test conductor started the FRAPS recording by pressing the pre-mapped key (F9). Upon map completion or character death, the test conductor exited Quake III by pressing the pre-mapped key (F5). This allows the next map to load (because of the test harness structure as discussed earlier) and also stops the FRAPS recording of that map. The FRAPS recording also generated a text file containing information about the map: time elapsed, frame rate, as well as the date. This information was used to determine the frame rates of the video files during data compiling and analysis. At the end of each user test, the video recordings and log file were placed in a folder corresponding to the user number assigned at the beginning of the test.

4. Analysis

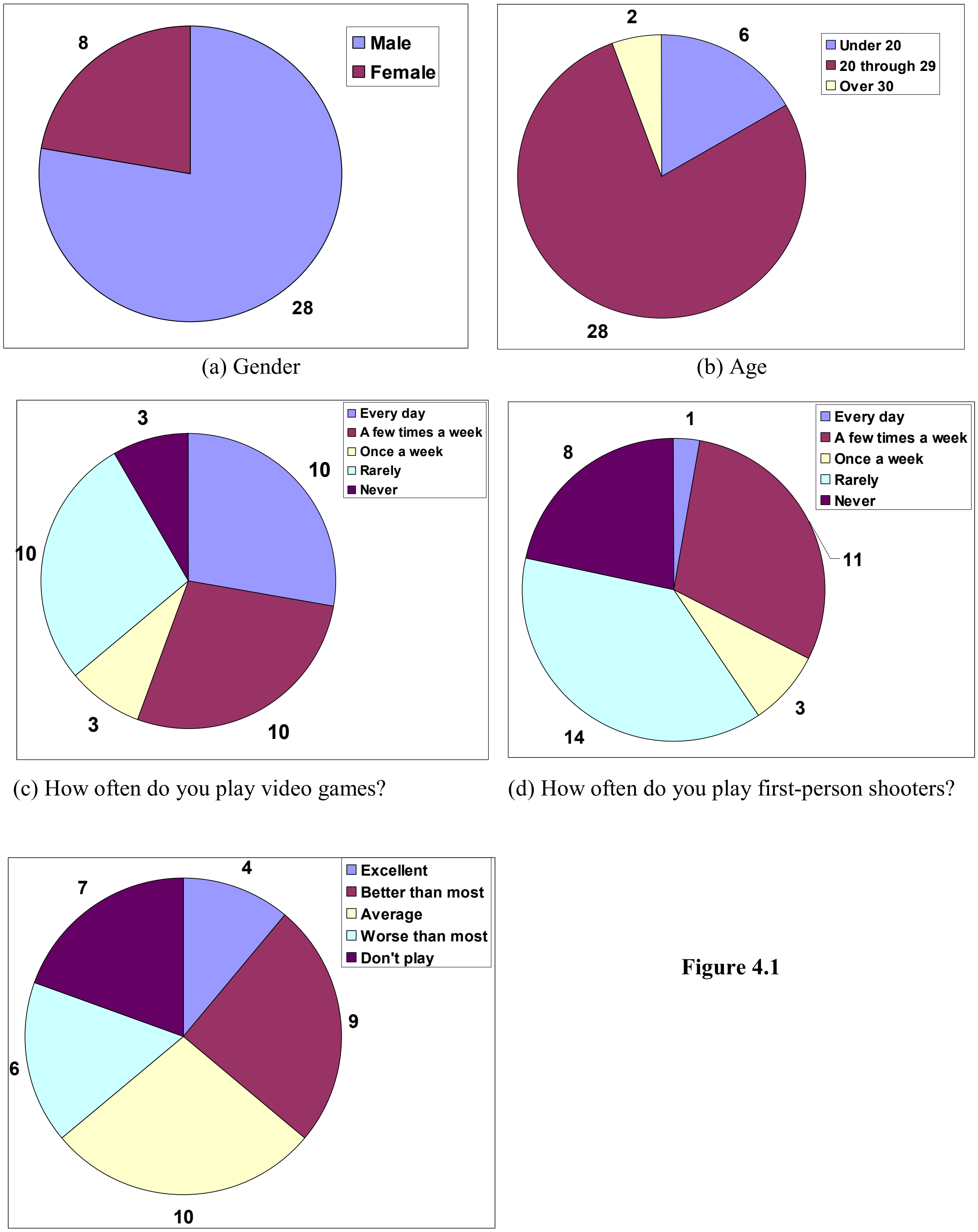
For analysis of our study, we will look at the data in several ways. First, we will examine the participants of our study. Then, the first and most important of these analyses is to evaluate the performance of all participants and see if any distinction in performance, time taken, or health lost can be made based on the changes in frame rate or resolution. We would also like to analyze how the participants' strategies change as our variables change. In a final analysis, we will look at results from all of our studies at once.

4.1 Demographics

Testing for the study lasted approximately two weeks, and 31 of our 36 participants were WPI students or faculty. Demographic information is summarized in Figure 4.1.

4.2 Performance Metrics

Before analysis began, it was decided that an overall performance rating for the frame rate trials was needed in order to combine the two numerical components of the trials, time taken and health lost. Resolution trials, however, needed only the time component of performance as it was not possible to lose any health in those maps. Since we only recorded health remaining at the end of each trial, in order to obtain the amount of health lost in each trial, we would have to subtract remaining health, the number we had recorded, from the maximum amount of health possible at the end of each study. This became an issue as, due to a game mechanic of Quake III Arena itself, players began each map with 120% health and lost one health point per second until 95% health was reached. To solve this issue, test runs of the frame rate maps were taken in order



(a) Gender

(b) Age

(c) How often do you play video games?

(d) How often do you play first-person shooters?

Figure 4.1

(e) How would you evaluate your performance in first-person shooters?

to find this maximum health with "God mode" on, in order to make the avatar invincible and not subject to the same health loss due to lava as a normal participant.

Maximum health for each map was found to be 110, 102, and 112 for Walking, Jumping, and Haste, respectively. Resultantly, the following formulas were used to compute performance ratings:

Walking: Performance = (Time taken) x (110 – Health remaining)
Jumping: Performance = (Time taken) x (102 – Health remaining)
Haste: Performance = (Time taken) x (112 – Health remaining)
Resolution: Performance = Time taken

A lower number indicates a better performance since taking less time on any trial is preferable to having taken more time, and having lost less health is preferable to having lost more health.

We also came to find that, as expected, our data contained outliers. In all cases in our study, these outliers were participants who were simply not good enough at Quake III Arena to have their data included. These participants' trials either took too long or ended with a premature character death or severe injury. Outliers were determined by the 1.5 x IQR method. In this method, median (Q_2), first quartile (Q_1), and third quartile (Q_3) are determined in normal fashion. A lower bound was established at $Q_1 – 1.5$ x $(Q_3 - Q_1)$ and any values lower than that were removed. An upper bound was established at $Q_3 – 1.5$ x $(Q_3 - Q_1)$ and values exceeding that were removed. Due to our relatively small number of participants, we decided that when removing outliers, only individual data points would be removed as opposed to the participant's entire set of points. Additionally, any trial in which the participant's avatar died, due to prolonged exposure to the lava, was removed as it gives an exceedingly high value for health lost and a deceptively low value for time taken, as the participant did not play the trial to completion. These exclusions in our data gave us sufficient confidence that our data now included only differences due to our imposed variables, as opposed to vast differences in skill. The number of points removed varied from 1 (in Haste at 3 fps) to 9 (in Jumping at 15 fps). In most cases, only 3 to 4 points of data were removed. In all graphs in this analysis, these trimmed data sets are used.

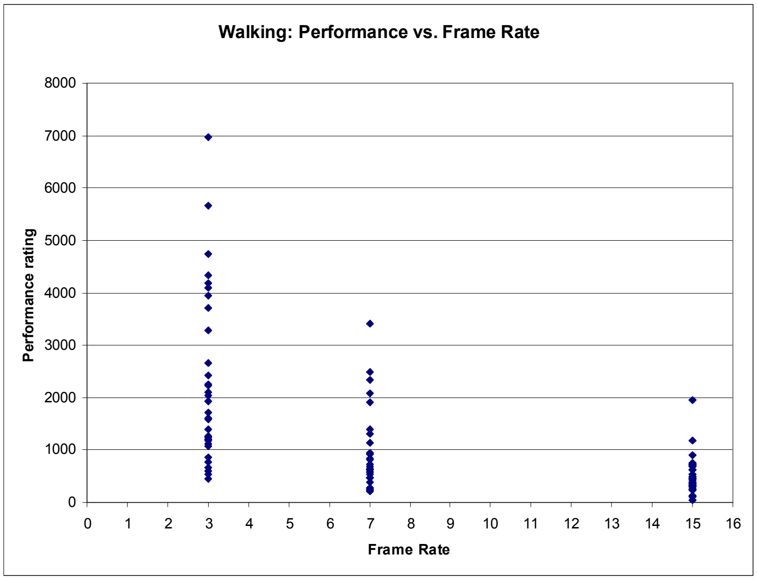


Figure 4.2

4.3 Analysis of Walking Maps

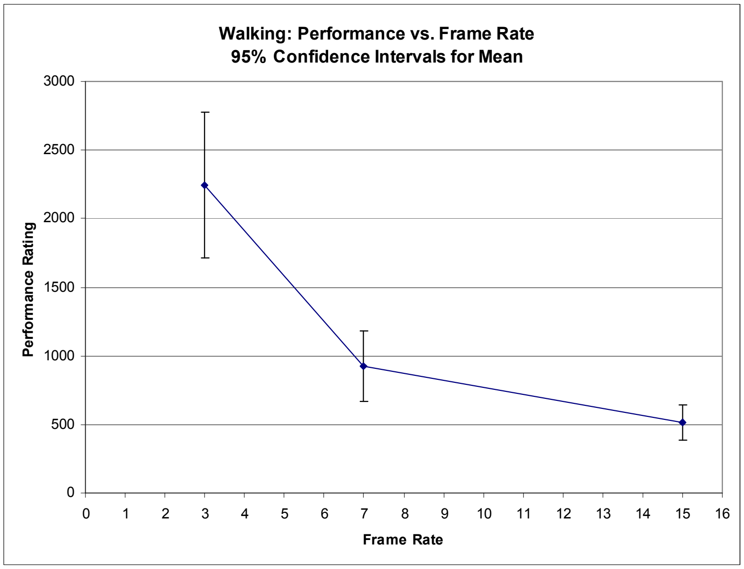


Figure 4.3

We begin our analysis with a look at results for our walking trials. Figure 4.2 shows the results for Performance vs. Frame Rate. One can see that as frame rate improves, the worst performance (indicated by the uppermost value in each column of points) improves greatly. Best performance (the lowest value in each column) improves only slightly, but that is because there was little room for improvement. As such, the spread of the data lessens by roughly 5000 in performance rating, indicating that higher frame rate provides participants the potential for better performance while lower frame rate allows those who are skilled to do well while others perform poorly

Figure 4.3 shows the mean values with confidence intervals. Confidence intervals allow us to say that, with a certain amount of confidence, the true value for the mean of the entire population lies within the interval around the sample mean. If these confidence intervals do not overlap, we can conclude a causal relationship between, in our case, frames per second and performance rating. In the graph, one can see that our confidence intervals, in fact, do not overlap. This allows us to conclude that, with 95% confidence, performance at 7 fps is strictly better than performance at 3 fps and that performance at 15 fps is strictly better than performance at 7fps.

In order to further investigate performance in our walking maps, we explore time taken and health lost separately. Figures 4.4 and 4.5 are graphs of confidence intervals for both time and health lost separately, respectively. Their coordinating scatter plots can be found in Appendix D.1. As one can see by the same confidence interval analysis as previously discussed, a significant difference comes at 3 fps on the Time vs. Frame Rate graph, indicating that time taken at 3 frames per second is strictly worse than performance at 7 fps and

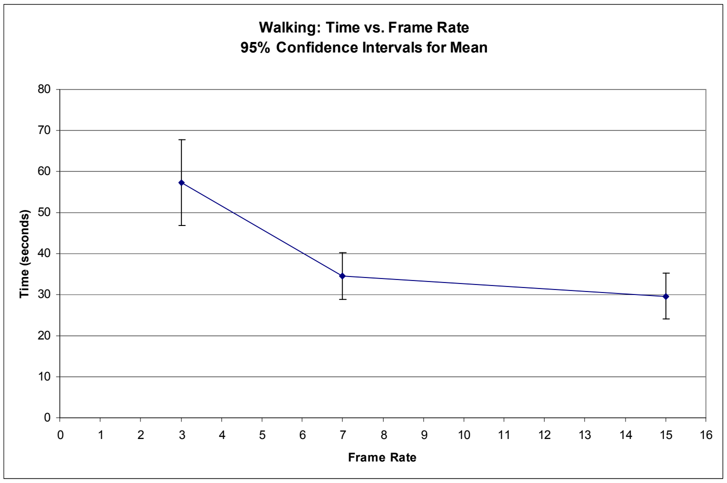


Figure 4.4

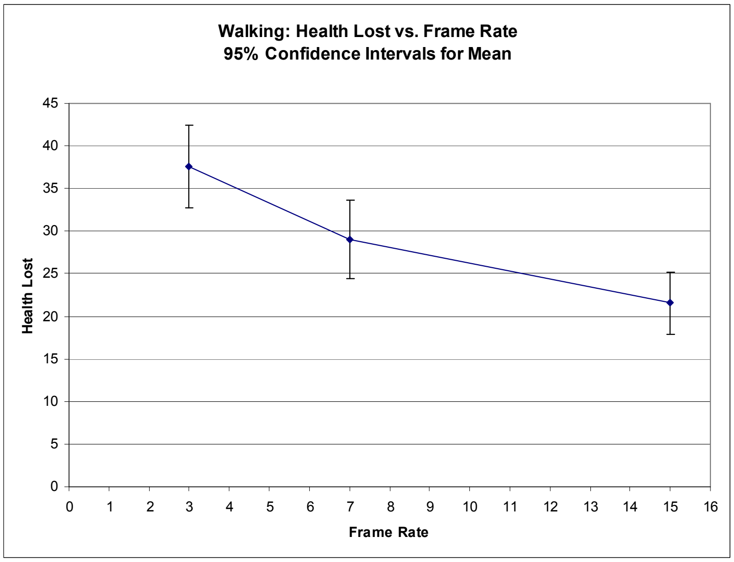
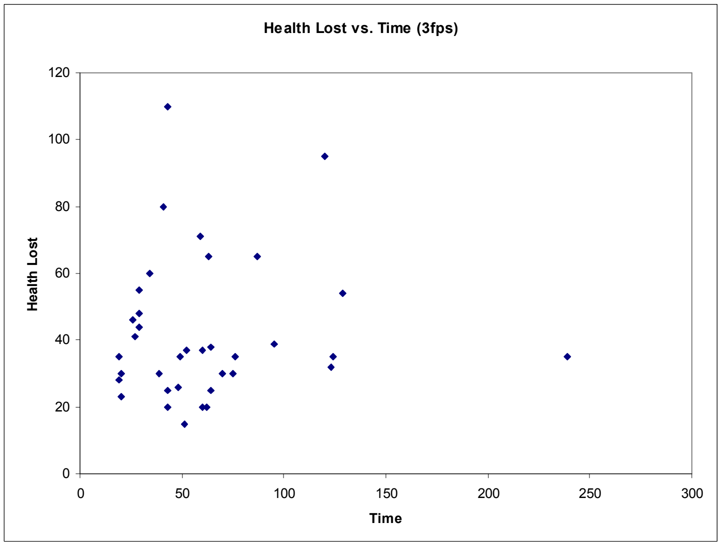
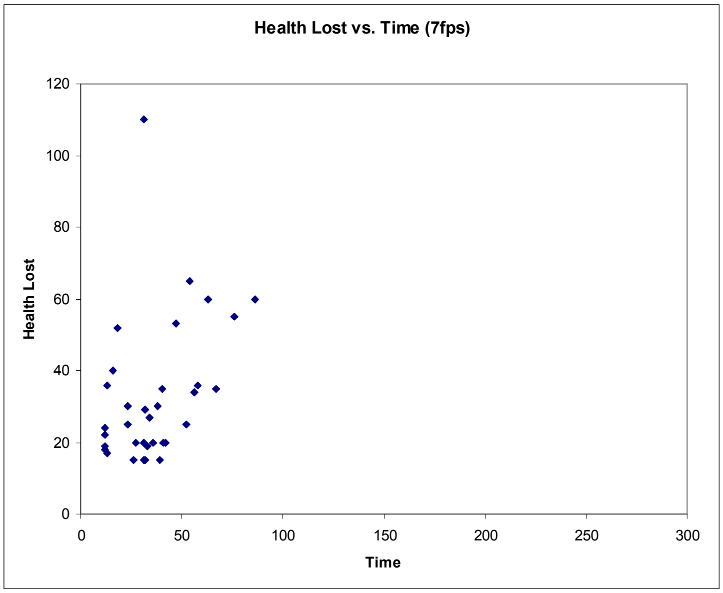


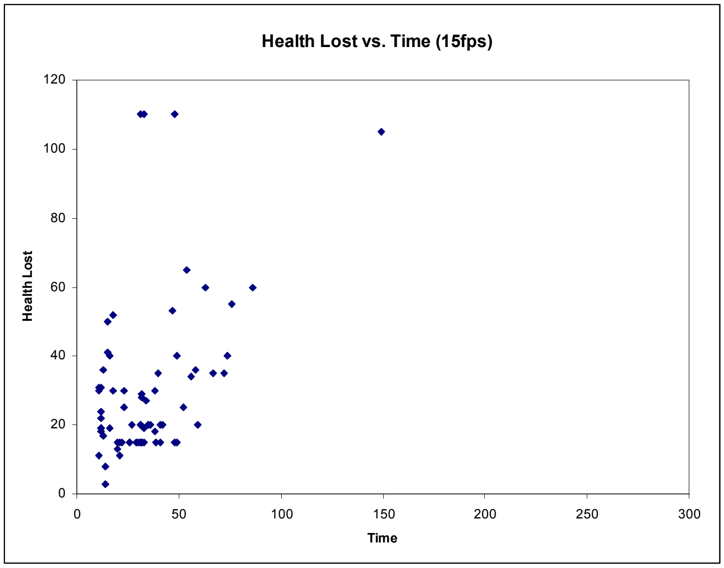
Figure 4.5

performance at 15 fps. Additionally, for health lost, 3 fps is significantly worse than at 15 fps.

Figures 4.6 through 4.8 provide another view of time taken and health lost. In these graphs, we see health lost graphed against time taken. As frame rate increases, the general cluster of points moves toward the left, indicating completion of trials in less time. Additionally, with the exception of some extreme values, the general trend of the cluster is in the downward direction, indicating the loss of less health as frame rate increases.







Top to bottom:
Figures 4.6 through 4.8

4.4 Analysis of Jumping Maps

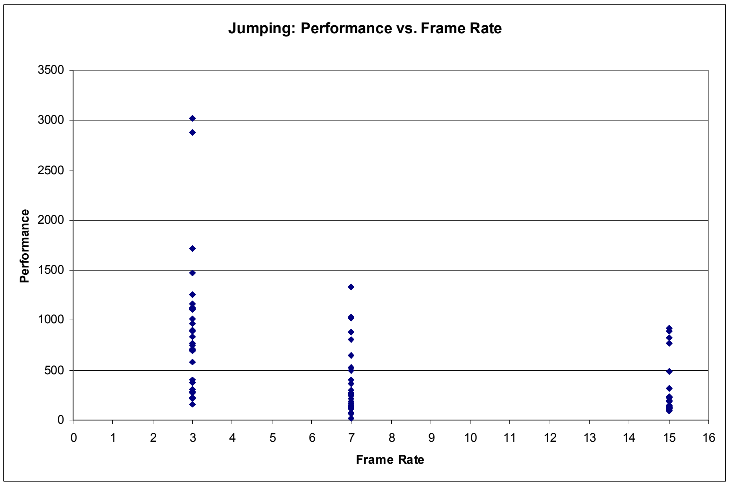


Figure 4.9

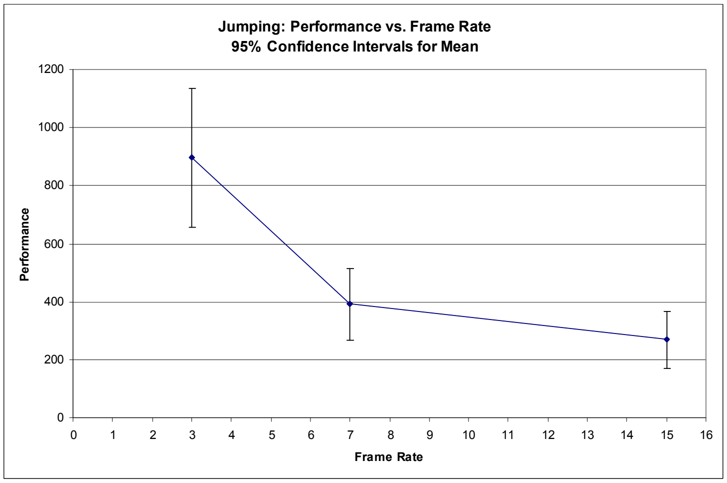


Figure 4.10

Analysis of jumping maps yields similar results. To begin, the scatter plot of Figure 4.9 follows the same general trend as did the scatter plot of the data for walking. Overall, the performance ratings are lower than those of walking. We believe this is due to the linearity of the jumping map; it is much harder for the player to wander off of a straight path into the lava than a deliberately curvy path. In the 3 fps column, with the exception of extreme values, the spread is significantly less than that of walking, and the decrease in spread as frame rate improves is less extreme. We believe this to be for the same reason mentioned above, the decreased level of difficulty. As this is apparent, it can be seen that it may be difficult to compare the results for walking and jumping maps.

The same downward trend is apparent in Figure 4.10, the 95% confidence intervals for mean graph. This time, however, the confidence intervals for 7 fps and 15 fps overlap, indicating there is no statistical difference between them. The confidence interval for 3 fps, though, overlaps with neither 7 fps nor 15 fps. This indicates a statistical difference between 3 fps and the other two tested. Thus, we can conclude with 95% confidence that performance at 3 fps is worse than at 7 fps or 15 fps.

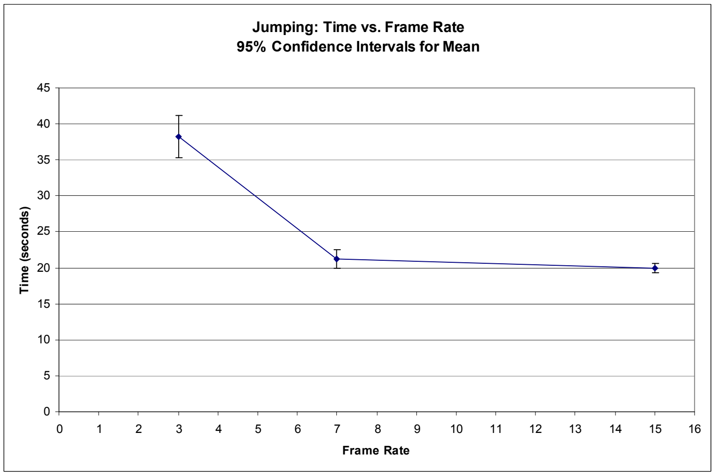


Figure 4.11

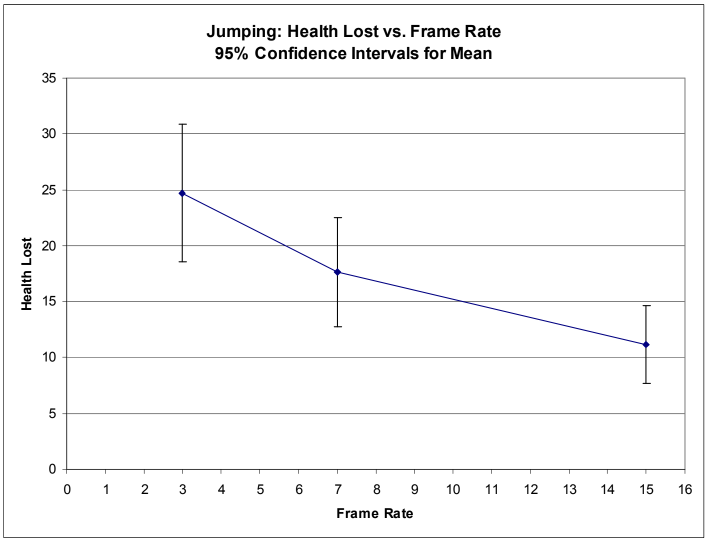


Figure 4.12

Further dissecting performance, we once again had examined time vs. frame rate on its own and health lost vs. frame rate on its own. Scatter plots of the data for this can be found in Appendix D.2. The graph in Figure 4.11, confidence intervals for time vs. frame rate, allows us to draw the conclusion that, in jumping maps for the frame rates tested, the time taken at 3 fps is significantly worse than at 7fps or 15 fps. As for health lost, Figure 4.12, we can conclude with 95% confidence that performance at 3 fps is significantly worse than performance at 15 fps, but nothing more.

Scatter plots of health lost vs. time taken can be found in the Appendix D.2. These graphs show a trend very similar to the one seen in the graphs for the walking maps. In general, the dot cluster moves to the left, to some threshold, and moves downward as frame rate increases. These movements indicate that as frame rate increases, time to completion of each map and health lost in each map decreases.

4.5 Analysis of Haste Maps

The final series of our frame rate tests, haste maps, show familiar trends. Figure 4.13, pictured to the right, shows the same general trend as that of walking and jumping. In general, as frame rate increases, the spread of the data lessens as worst performance improves and best performance does not. The graph of 95% confidence intervals for mean, Figure 4.14, shows that there is significant difference between performance at 3 fps and performance at 7fps and 15 fps, but there is not a significant difference between performance at 7 fps and 15 fps.

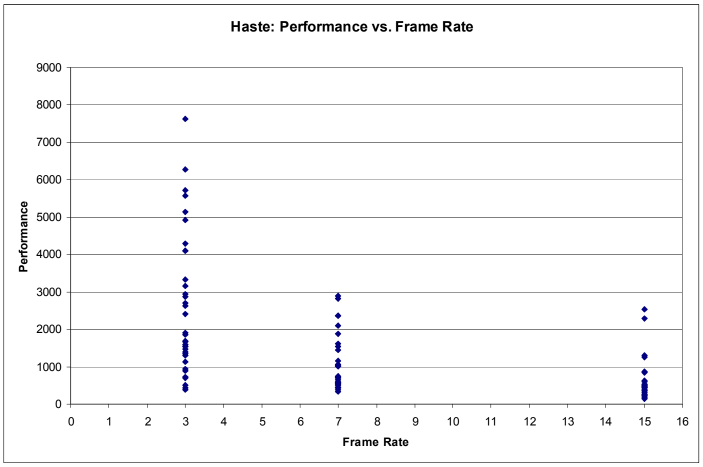


Figure 4.13

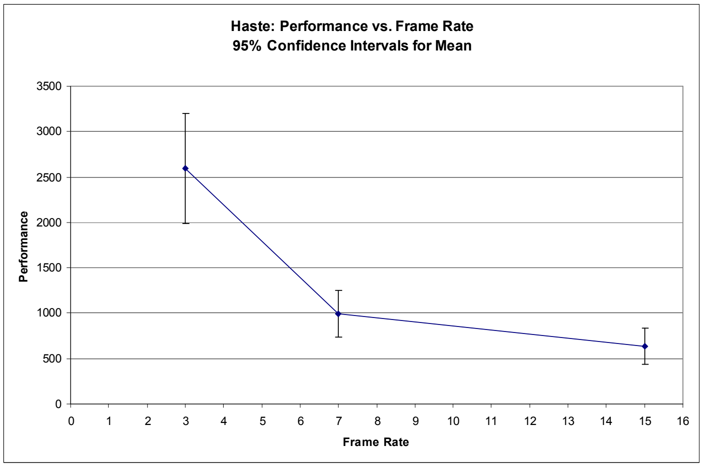
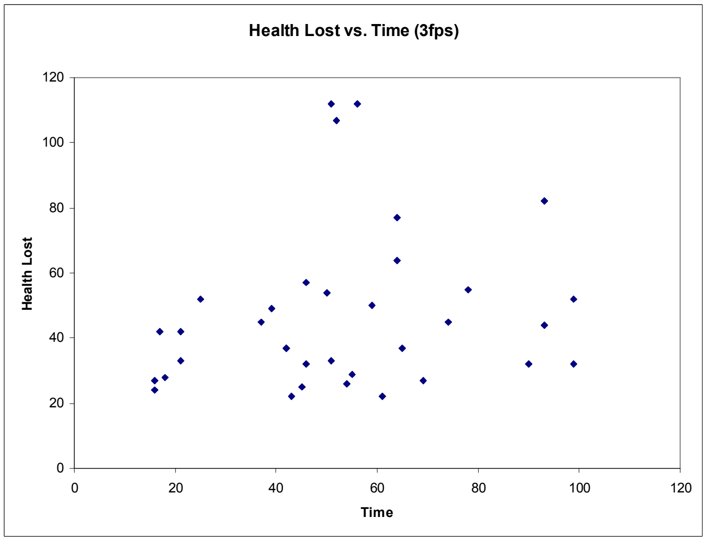
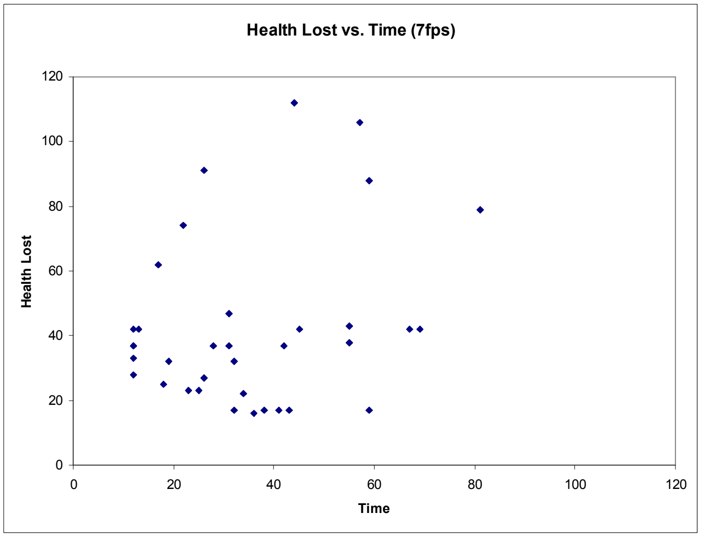


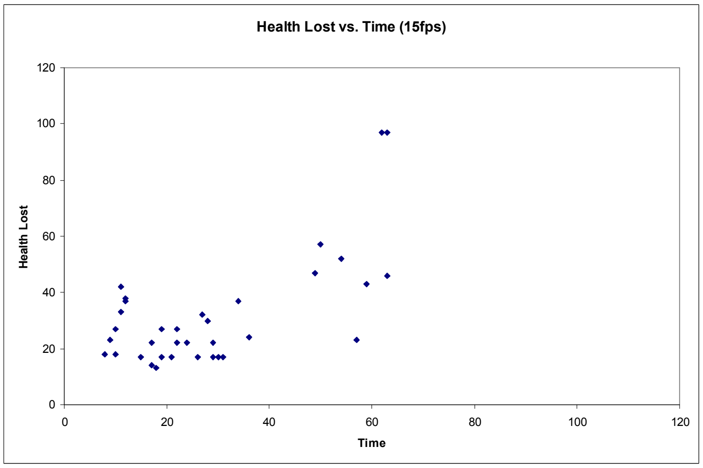
Figure 4.14

There is a drastic difference between performance at 3fps and performance at 15 fps. Comparing this gap to the one displayed in the analogous graph for walking tests, done on the same map with only a difference in speed applied, one can see that the change from 3 fps to 7 fps is significant. We believe this is due to the increased risk of making a mistake at low frame rates on the haste map, as movement speed is increased. Thus, players make more mistakes and exercise more caution, resulting in more health lost and more time taken respectively.

Taking a further look at time taken and health lost individually, we find results similar to those of the walking and jumping maps. Scatter plots and confidence intervals showing these results can be found in Appendix D.3. The scatter plots (Figures 4.15-4.17) of health lost vs. time taken are unique, though, in their unusually large spread. When compared to their analogous graphs for walking and jumping maps, found in the Appendices D.1 and D.2, one cannot help but notice the lack of a trend in the 3 fps map, except for the lack of participants who complete the map in a short amount of time but lost more than 60% health. While still not as convincing of a trend as in the other tests, the movement of the dot cluster still suggests that as frame rate increases, both time taken and health lost generally improve.







Top to bottom: Figures 4.15-4.17

4.6 Overall Analysis of Frame Rate Maps

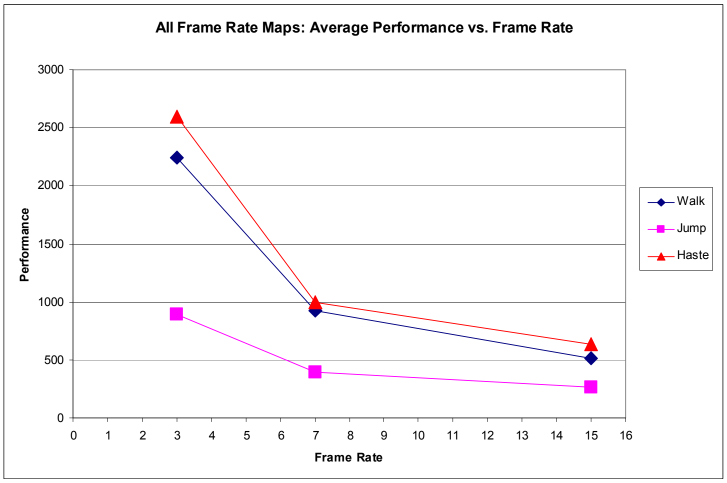


Figure 4.18

Examining the results of the frame rate maps together, qualitatively, certain drastic differences can be seen. Looking at the graph of performance vs. frame rate, Figure 4.18, it can be seen that, roughly, average performance for walking and haste maps are similar. They follow almost an identical trend, with average performance on haste maps slightly worse, again due to the increased chance to make a mistake (due to increased speed) in the haste maps. The average performance for jumping maps, while improving as frame rate increases, have a much better performance than walking and jumping overall. We believe this shows a disparity in difficulty of the jumping map when compared to either of the other frame rate maps.

Analysis of slope for Figure 4.18 provides us with additional information. Walking and Haste maps still have their almost identical, slightly separated trend. The graph of Jumping, however, has a less severe slope than either Walking or Haste. What this tells us is that as frame rate improves, Walking and Haste improve much more sharply than Jumping.

The graph showing average time vs. frame rate for all three frame rate maps (Figure 4.19) shows a similar result: walking and haste times are almost identical while jumping's time is much lower. This is interesting due to the fact that all three maps had identical path lengths (and thus, identical distances to be traversed). The reasoning for this, most likely, is that the turning that is necessary in walking and haste maps is not present in the jumping map. It is this turning that must account for the extra time. Analysis of slope, in this case, does not provide data as apparent as in the performance graph. It can be seen, though, that Walking does improve more sharply than Haste from 3 fps to 7fps and Jumping does not improve as quickly as Walking or Haste from 7 fps to 15 fps.

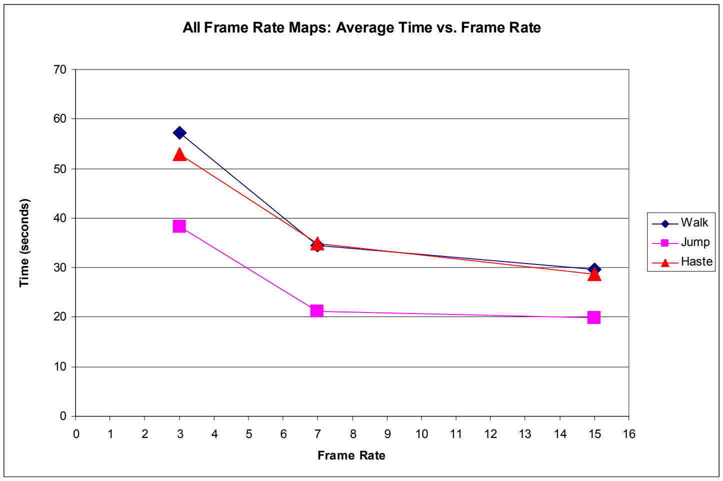


Figure 4.19

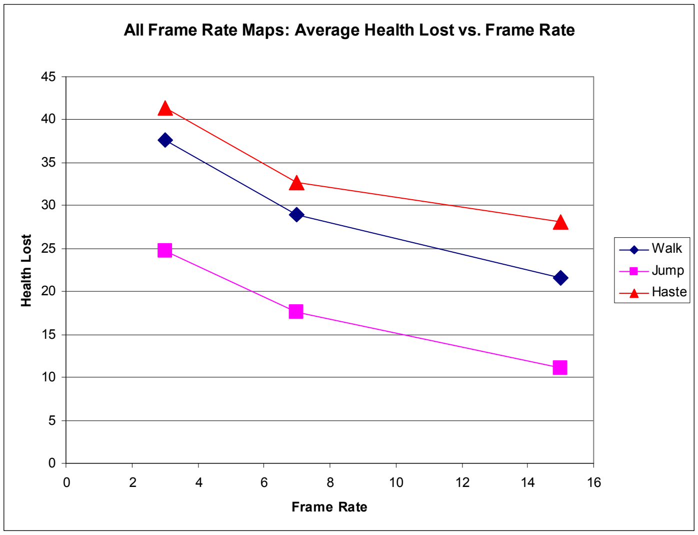


Figure 4.20

The same general trend is apparent in the graph of average health lost vs. frame rate, Figure 4.20. While separated only a few values in terms of health lost, walking and haste maps generally follow the same trend and are near each other. Again, though, the graph for the jumping results is significantly below the other two, indicating that, on average, our participants lost less health in jumping maps. This is mostly likely due to the fact that is it much harder to wander off of a straight path and into the lava than it is to wander off a path with turns. In this graph, difference in slope is even less significant than in the previous figures. However, it is seen that Walking improves slightly faster than Haste from 7 fps to 15 fps.

We believe that these trends show us that the jumping map used in our tests is generally easier to complete than the walking or haste maps. As such, as frame rate improves, the improvement in performance is less drastic than in the Walking and Haste maps. This is either due to the fact that the isolated task of accurate jumping is easier than walking or running or that bias in construction of our test maps created this result.

4.7 Analysis of Recognition Maps

Finally, we have our results for our resolution tests. Due to the fact that there are fewer parameters able to be studied in these tests, analysis of these results is limited to these two graphs, seen in Figures 4.21 and 4.22. In these graphs, the resolution studied is on the X-axis and the performance, measured in seconds of time, is on the Y-axis. In these graphs, not unlike previous graphs studied, a general improving trend can be seen as resolution increases. The scatter plot, ignoring extreme values, shows a slight improvement from 320x480 to 512x384 and a seemingly drastic improvement from 512x384 to 540x480. The spread of the data also greatly lessens as at the highest resolution. Figure 4.22, our graph of 95% confidence intervals for this information supports this claim. While our mean for time taken at 512x384 is less than that of 320x480, the difference is not statistically significant. What is significant, however, is time taken at 640x480. Our data for this highest resolution shows a significant difference from the two lesser resolutions. What this tells us is that time taken, and thus recognition of distant objects, is better at 640x480 than at 320x480 or even 512x384.

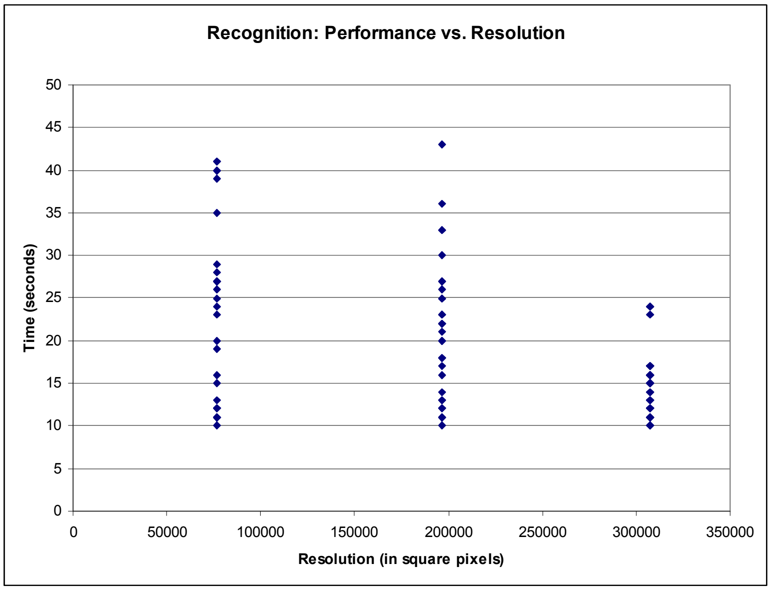


Figure 4.21

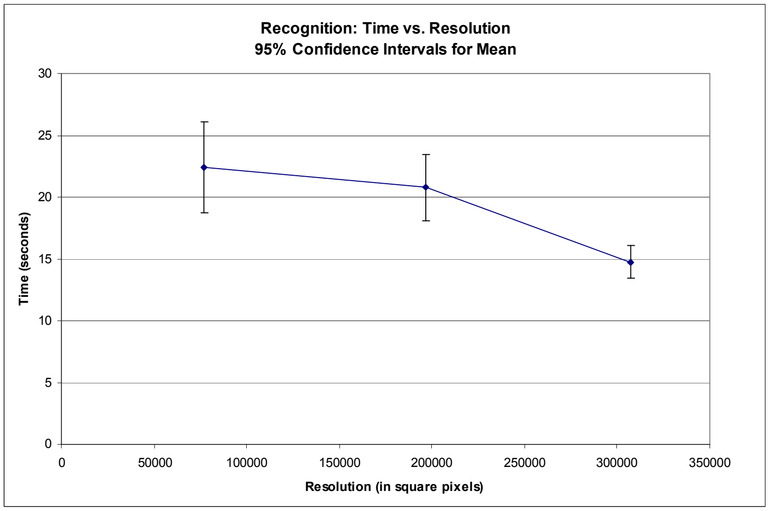


Figure 4.22

5. Conclusions

In the constantly increasing technological landscape it is necessary for users and developers alike to have the information and tools at their disposal to make decisions about their media. In the case of first person shooter video games, both groups need information to help make decisions about how to develop and run their media. This will allow users to get the best possible performance out of the game and themselves, and developers to know what settings to provide based on these following conclusions.

In our study we examined movement and recognition tasks in Quake III, a first person shooter game. Using custom maps, we performed a user study that tested walking, running, jumping, and visual identification at varying frame rates and resolutions. The following is what we can conclude from our data.

For the movement studies – walking, jumping, and running – it was found overall that that performance is significantly worse at 3 fps than at 7 or 15 fps. Additionally, in the case of the walking study, performance at 7 fps was also significantly worse than at 15 fps. Because confidence intervals shrink as the amount of data increases, it is believed that with a larger sample size, this significance would become standard across all of the tests, and the jumping and haste trials would be similar to the walking trials. Thus, with a larger sample size, our data would be similar to the study on shooting in Quake III Arena, where statistical significance was found between 3, 7, and 15 fps. However, our current data only suggests this trend. We believe that due to shooting being a more precision-oriented task, statistical significance was easier to determine in the previous study.

In these movement tasks, it was also found that as frame rate improves, both time taken to complete a task that improves and control over one’s in-game avatar improves. We know this due to the fact that for graphs of mean, the confidence intervals at 3 fps do not overlap with those at 15 fps. This trend is present in all frame rate and resolution maps, whether for performance, health alone, or time alone. Regarding time taken and health lost, we have also discovered a trend that a higher frame rate allows one to perform more reliably to their abilities, rather than be affected by the lower quality system settings.

Concerning the effects of resolution on recognition, it was found that resolution does, in fact, have an impact. The analysis showed that recognition at the highest resolution tested, 640x480 pixels, is significantly better than the lower two, 512x384 and 320x240. However, recognition between 512x384 and 320x480, were not significantly different. It is worthy of note that 512x384 and 640x480 are “adjacent” to each other in terms of order. Due to this, we are able to pinpoint 512x384 as the very highest resolution at which performance is still hindered by lack of quality system settings. Unfortunately, it is not possible to make any claims on the any resolutions higher than 640x480, so no assumptions can be made on whether or not this improvement in performance continues to grow. In future studies, research of these higher resolutions may be of significant interest.

These conclusions are similar to the results found in the previous study by Claypool (1999). As you may recall, it was found that frame-rate impacted user's performance in shooting related tasks, but resolution did not. Our data for the effects of frame rate on movement related tasks coincide with those found in the study, and can now be looked on as a whole. Our resolution task was not merely movement related, but was moreover a test on recognition, which we believe to be an integral part of the accuracy tasks that were studied by Claypool. In this way, we have continued the previous study in which they found no effects of resolution on accuracy, but we have found that resolution does in fact affect recognition which can be an integral part of some accuracy tasks. We have also discovered a difference in the studies' results that indicates that shooting is a more precision-oriented task than movement; the “knee” in the shooting study, the point at which the trend line bends the most, was at 15 fps. In our study, though frame rates above 15 fps were not studied, it is mathematically impossible that the “knee” could be anywhere but 7 fps unless the trend begins to change at higher frame rates.

The conclusions and trends that we have found allow players and developers to make decisions regarding minimum acceptability for first person shooter games. For players, this data can be used to make decisions about hardware purchases or graphic quality settings. For developers, it can be used to determine minimum default settings that their maps and game play should be able to run with no hindrance on performance.

From our study, future researchers now have new materials with which to continue studies into this field. To run the experiment again, or similar experiments, researchers can use the same Quake 3 maps for walking, running, jumping and recognition. Also, the test harness is available to use, or be modified to fit the needs of future studies. Our data is also available in its raw form to be analyzed in many different manners, to allow for new and different conclusions.

5.1 Future Work

Based on our data and conclusions, there are two areas in which future work could be done. The first area is demographics. Our results are formed as a general analysis of the overall data. However, it would be useful to see how performance was affected between males and females, and across age ranges. Studies could be done comparing performance between genders to determine if they fit out correlations. In the case of age, future studies could examine performance among age brackets to determine if indeed it affects performance. For this future work, the materials in this study can be reused.

The second area involves changing the game instead of the participants. One idea for a future study is to examine a newer first person shooter. Graphics have advanced greatly since Quake III, and a study of frame rate and resolution on a newer game would be more relevant for today. It would be interesting to observe whether an increase in graphic rendering would change the way video settings affect performance. Another important study would be to extend the research of the effects of frame rate and resolution to other genres of games, namely: real time strategies, fighting games, racing games, and sports games. Studies could be done to determine which genre is affected most by changes in video settings, and which is affected the least. In these cases, comparisons could be made across games of varying play styles.

BIBLIOGRAPHY

Beigbeder, Tom, Rory Coughlan, Corey Lusher, and John Plunkett. “The Effects of Packet Loss and Latency on Player Performance in Unreal Tournament 2003.” Worcester Polytechnic Institute, Worcester, MA, 2004.

Bryson, Steve. “Effects of Lag and Frame Rate on Various Tracking Tasks.” NASA Ames Research Center, Moffett Field, CA, 1993.

Claypool, Mark, Claypool, Kajal, and Damaa, Feissal. “The Effects of Frame Rate and Resolution on Users Playing First Person Shooter Games.” Multimedia Computing and Networking, San Jose, California, 2006.

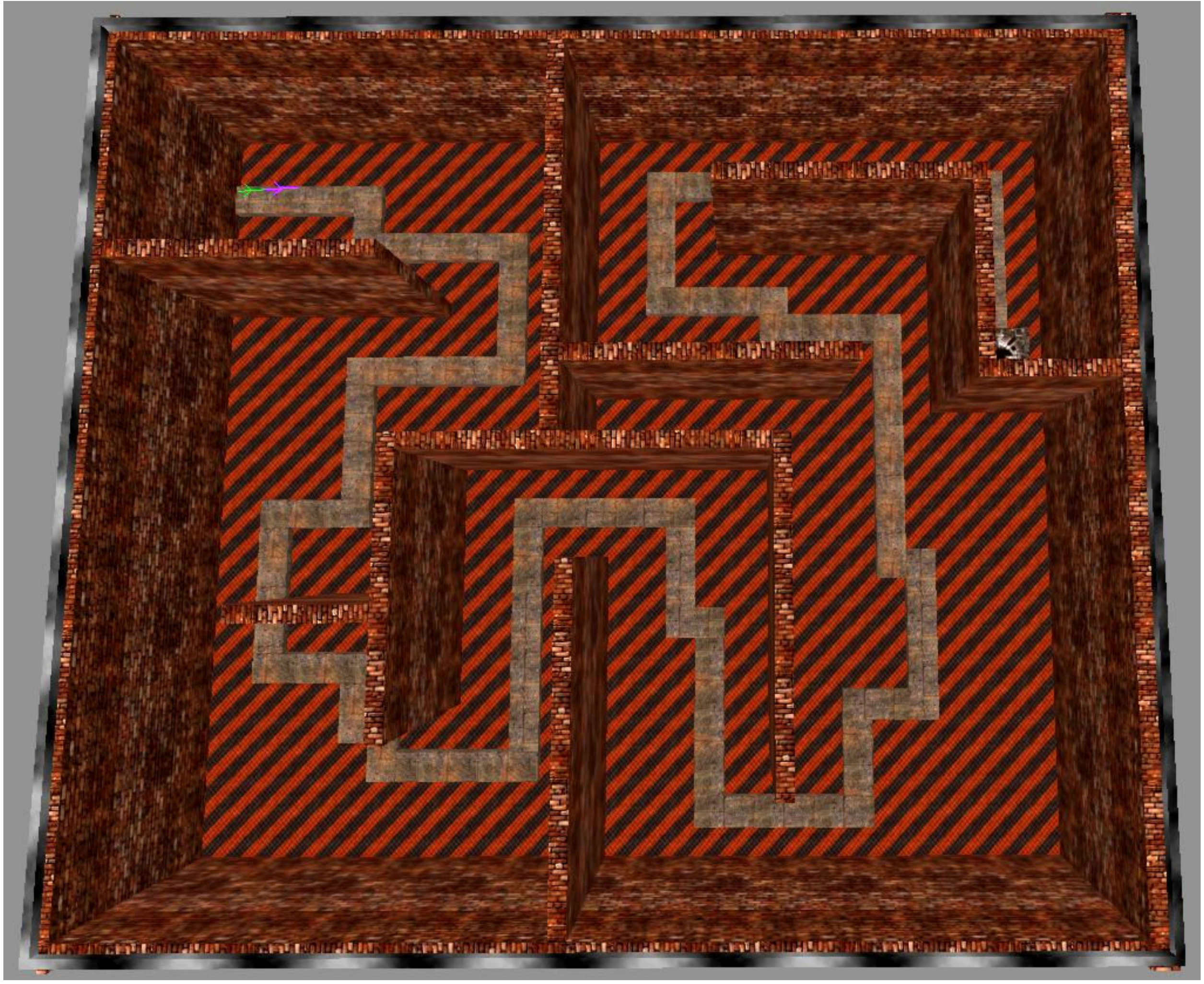
McCarthy, John D, Sasse, Angela, and Miras, Dimitrios. “Sharp or Smooth? Comparing the Effects of Video Quantization vs. Frame Rate for Streamed Video.” Conference on Human Factors in Computing Systems, Vienna, Austria, 2004.

Reddy, Martin. “The Effects of Low Frame Rate on a Measure for User Performance in Virtual Environments.” Technical Report ECS-CSG-36-97, University of Edinburg, Scotland, UK, 1997.

Tripathi, Avanish, and Claypool, Mark. “Improving Multimedia Streaming with Content Aware Video Scaling.” Intelligent Multimedia Computing and Networking, Durham, North Carolina, 2002.

APPENDIX A

Extraneous Map Materials



A.1 Overhead of Jumping/Haste Map

A.2 Overhead View of Jumping Map

A.3 Overhead View of Recognition Map

APPENDIX B
Java Harness

```
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.io.Writer;
import java.util.Random;

public class studyHarness {

        /**Base Directory for Quake executable
         */
        public static String baseDir = "C:\\Program Files\\Quake III Arena\\";

        /**bat file commands
         */
        public static String walk15 = "+r_mode 4 +com_maxfps 15 +map walking";
        public static String walk7 = "+r_mode 4 +com_maxfps 7 +map walking";
        public static String walk3 = "+r_mode 4 +com_maxfps 3 +map walking";
        public static String haste15 = "+r_mode 4 +com_maxfps 15 +map haste";
        public static String haste7 = "+r_mode 4 +com_maxfps 7 +map haste";
        public static String haste3 = "+r_mode 4 +com_maxfps 3 +map haste";
        public static String jump15 = "+r_mode 4 +com_maxfps 15 +map jumping";
        public static String jump7 = "+r_mode 4 +com_maxfps 7 +map jumping";
        public static String jump3 = "+r_mode 4 +com_maxfps 3 +map jumping";

        //res maps are not given here, randomly chosen later
        public static String res3 = "+r_mode 3 +com_maxfps 30 +map";
        public static String res2 = "+r_mode 2 +com_maxfps 30 +map";
        public static String res1 = "+r_mode 0 +com_maxfps 30 +map";

        public static void makeBat(String[] commands) throws IOException{
                File bat = new File(baseDir + "iqp.bat");
                Writer output = null;
            try {
              output = new BufferedWriter( new FileWriter(bat) );
              for(int i=0; i<commands.length;i++){
                 output.write( commands[i] + "\n");
              }
            }
            finally {
              if (output != null) output.close();

            }
        }

        public static String[] sortCommands(String[] commands){
                String temp;
                if((commands[0].substring(commands[0].length()-4,commands[0].length()-
1).compareTo("res")) == 0){
                        temp = commands[0];
                        commands[0] = commands[1];
                        commands[1] = temp;
                }
                for(int i=0; i<commands.length; i++){
                        if ((commands[i].substring(commands[i].length()-4,commands[i].length()-
1).compareTo("res")) == 0 ){
                                temp = commands[i].substring(0,10);
                                commands[i] = commands[i-1].substring(0,10) +
commands[i].substring(10,commands[i].length());
                                commands[i-1] = temp + commands[i-1].substring(10,commands[i-
1].length());
                        }
                }
                return commands;
        }
        public static String[] makeCommands(){

                String[] commands = new String[] {walk15, walk7, walk3,
                                                                             haste15, haste7,
haste3,
                                                                             jump15, jump7,
jump3};
                String[] resComs = new String[]{res3, res2, res1};
                String[] resMaps = new String[]{"res1", "res2", "res3"};

                randomize(resMaps);

                for(int i=0; i<resComs.length; i++){
                        resComs[i] = resComs[i] + " " + resMaps[i];
                }
                String[] commandsFull = new String[commands.length + resComs.length];
                for(int i=0; i<commands.length; i++){
                        commandsFull[i]=commands[i];
                }
                for(int j=0; j<resComs.length; j++){
                        commandsFull[j+commands.length]=resComs[j];
                }
                randomize(commandsFull);
                commandsFull = sortCommands(commandsFull);

                for (int i=0; i<commandsFull.length; i++){
                        commandsFull[i] = "\"" + baseDir + "quake3.exe\" " + commandsFull[i];
                        System.out.println(i + " " + commandsFull[i]);
                }
                return commandsFull;
        }

        public static void randomize(String[] s){
                int n = s.length;
                String temp;
                Random generator = new Random();
                while (n!=0){
                        int rand = generator.nextInt(n);
                        temp = s[n-1];
                        s[n-1] = s[rand];
                        s[rand] = temp;
                        n--;
                };

        }

        /**
         * @param args
         * @throws IOException
         */
        public static void main(String[] args) throws IOException {
                String[] commands =    makeCommands();
                makeBat(commands);
        }

}
```

APPENDIX C
Test Materials

INSTRUCTIONS (READ ALL BEFORE PROCEEDING)

1. Fill out survey.
2. Double-click icon on desktop labeled "IQPGO" (See explanation below)
3. At the end of each map you will explode. This indicates successful completion of the map.
4. Do this in all the maps.

IQPGO

- This will load different variations of four different Quake III maps.
- First, they will each be loaded once with unmodified settings. You are to get familiar with the maps and the control scheme here.

  Map 1: Walking
  You are on a path. You will get hurt if you step off the path. Get to the end of the map as quickly as possible while losing the least amount of health.

  Map 2: Jumping
  You are on a path. You will get hurt if you step off the path. You will need to jump in various places to do this. Get to the end of the map as quickly as possible while losing the least amount of health.

  Map 3: Running
  Now you are in Map 1 again, but you are moving twice as fast. The objective remains the same.

  Map 4: Recognition
  You are in a place with three straight paths before you. At the end of each path there is a visible button. One of them is different than the others. Pick the path with the different button at the end and press it as quickly as possible. If you choose the wrong path, you cannot jump over the divides.

- After these first four maps, they will be loaded with different video settings in random order.

Controls:

W: Forwards	**Up arrow**: Forwards
S: Backwards	**Down arrow**: Backwards
A: Strafe Left	**Left arrow**: Turn Left
D: Strafe Right	**Right arrow**: Turn Right

Spacebar: Jump
Mouse: Looks up, down / Turns left, right

DEMOGRAPHICS SURVEY

Please fill in or circle the appropriate responses. ___________

Gender:

Male______ Female______

Age: ______

How often do you play video games?

Everyday A few times a week Once a week Rarely Never

How often do you play first person shooters?

Everyday A few times a week Once a week Rarely Never

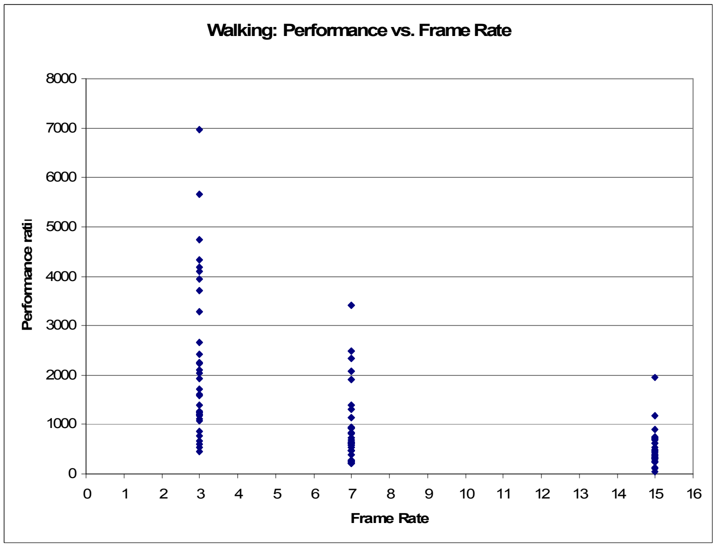
How would you evaluate your performance in first person shooters?

Excellent Better than most Average Worse than most Don’t play

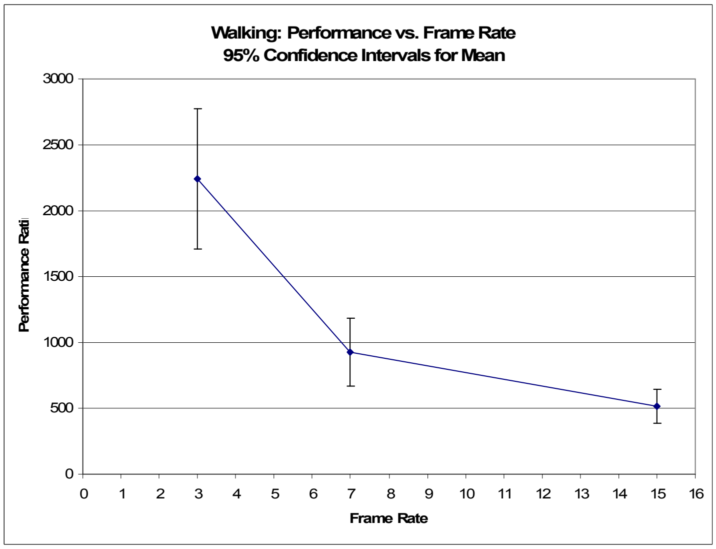
APPENDIX D
Graphs

D.1 Walking

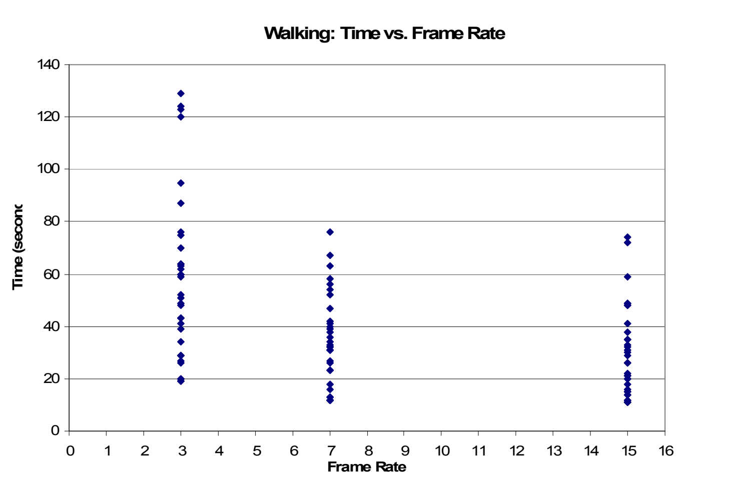
Performance vs. Frame Rate Scatter Plot



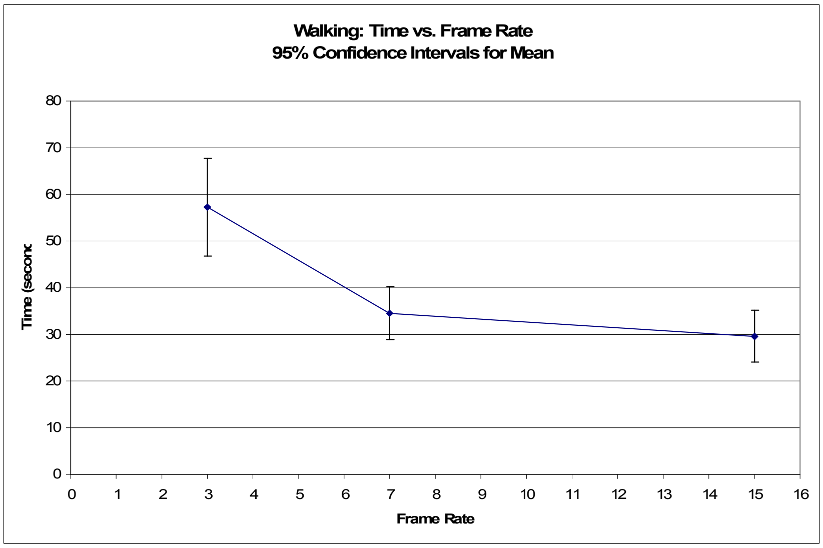
Performance vs. Frame Rate 95% Confidence intervals for mean



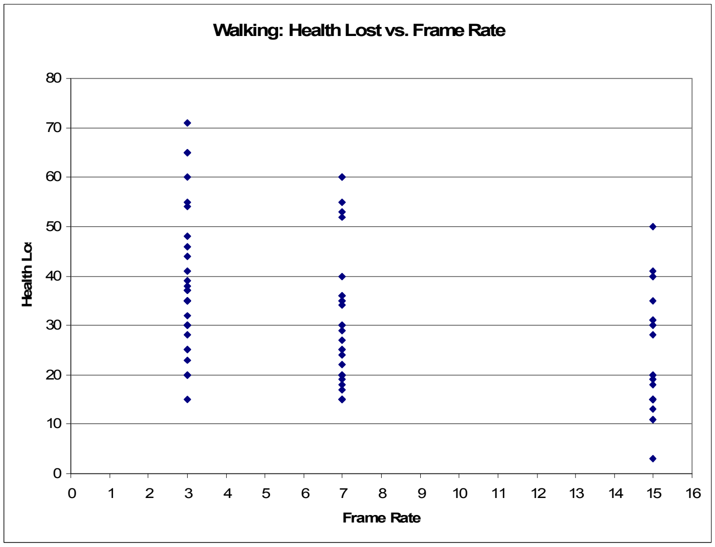
Time vs. Frame Rate Scatter Plot



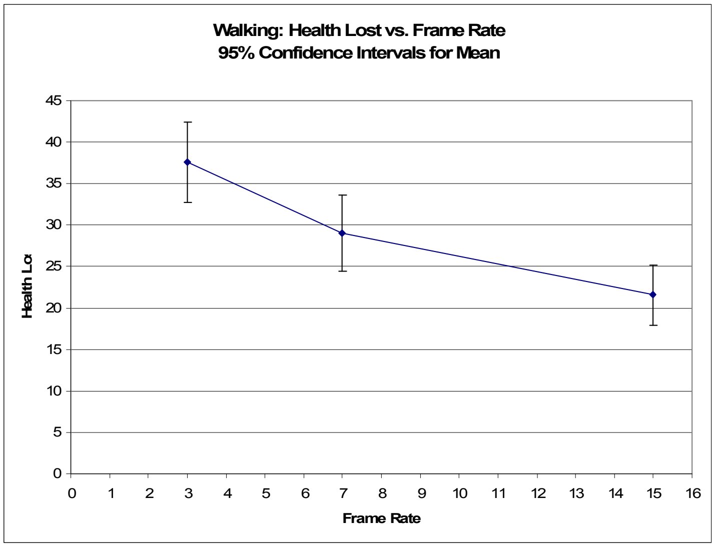
Time vs. Frame Rate
95% Confidence intervals for mean



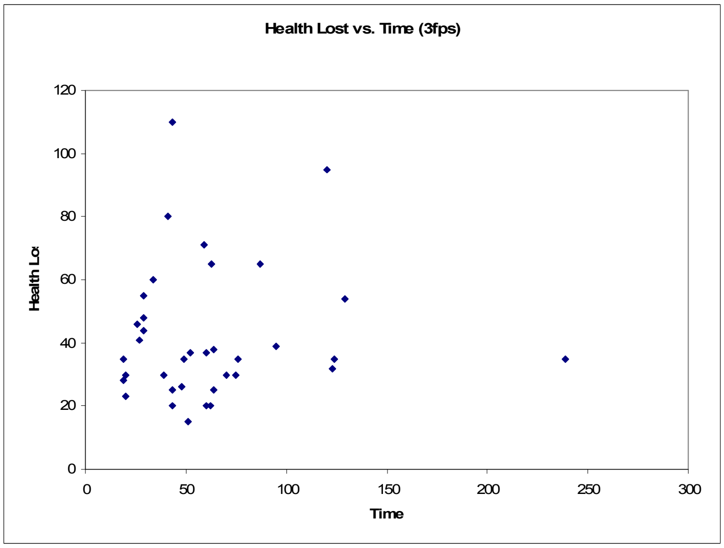
Health Lost vs. Frame Rate
Scatter Plot



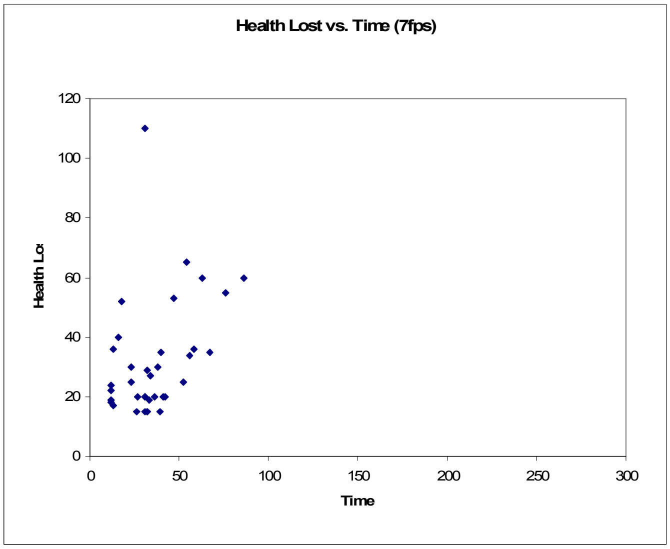
Health Lost vs. Frame Rate
95% Confidence intervals for mean



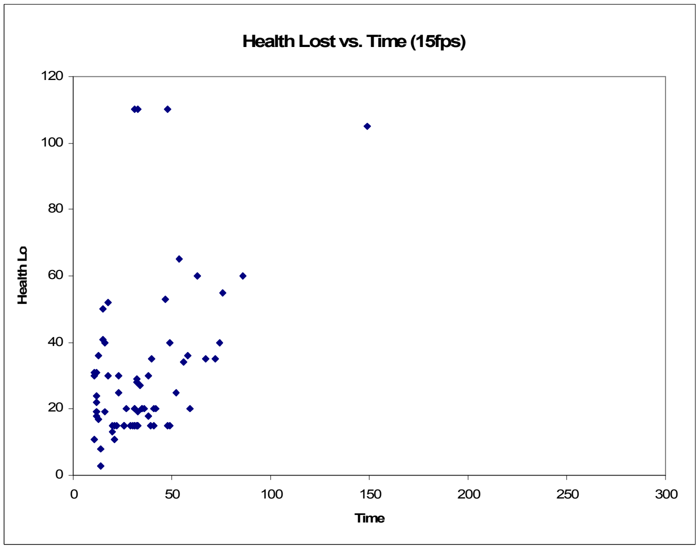
Health Lost vs. Time (3 fps)
Scatter Plot



Health Lost vs. Time (7 fps)
Scatter Plot

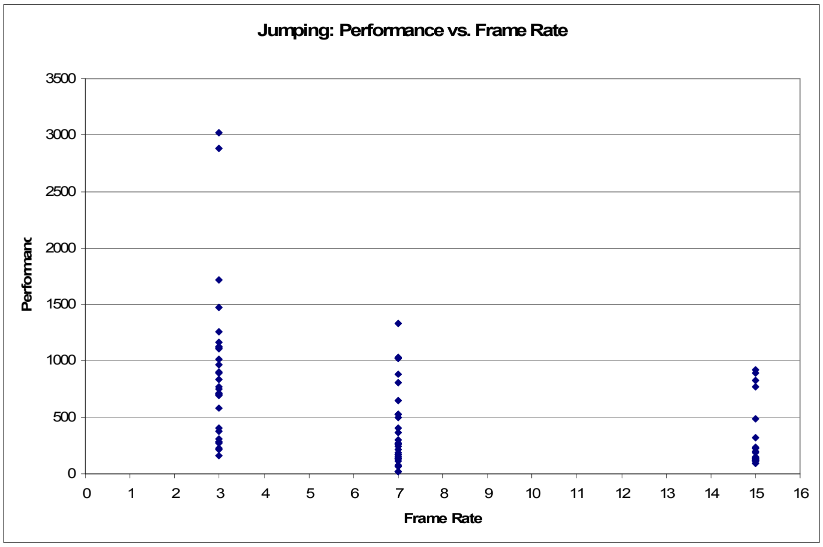


Health Lost vs. Time (15 fps)
Scatter Plot

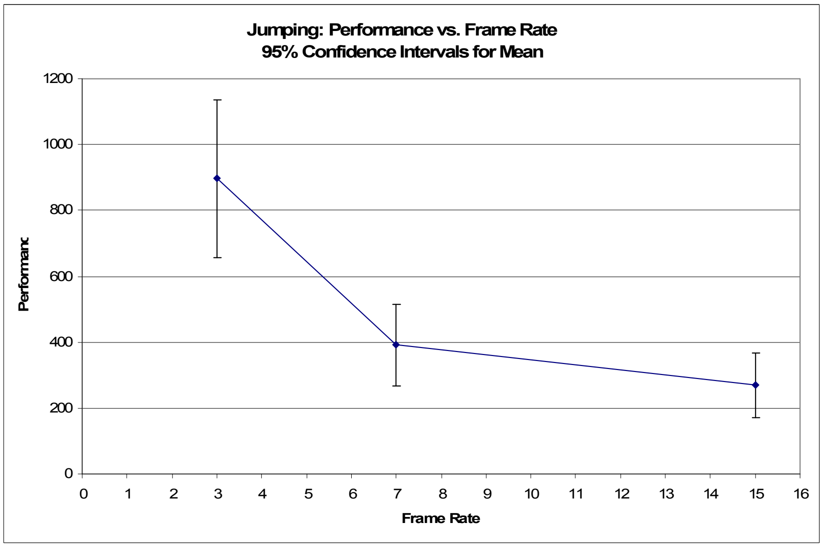


D.2 Jumping

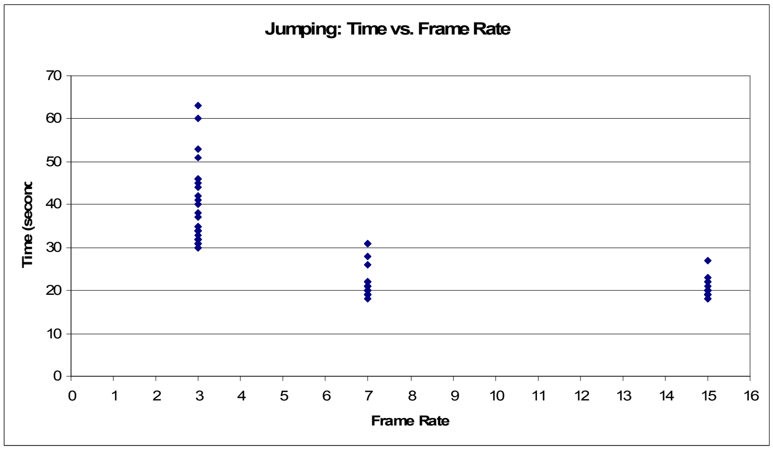
Performance vs. Frame Rate
Scatter Plot



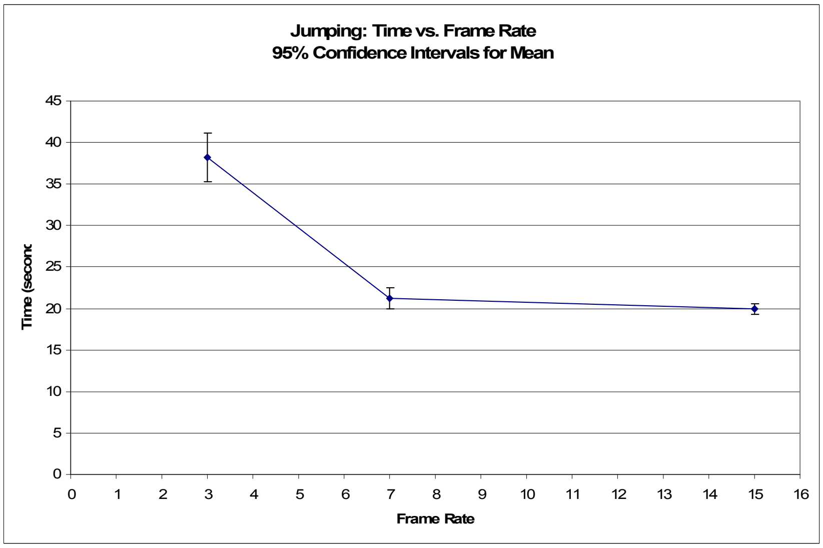
Performance vs. Frame Rate
95% Confidence intervals for mean



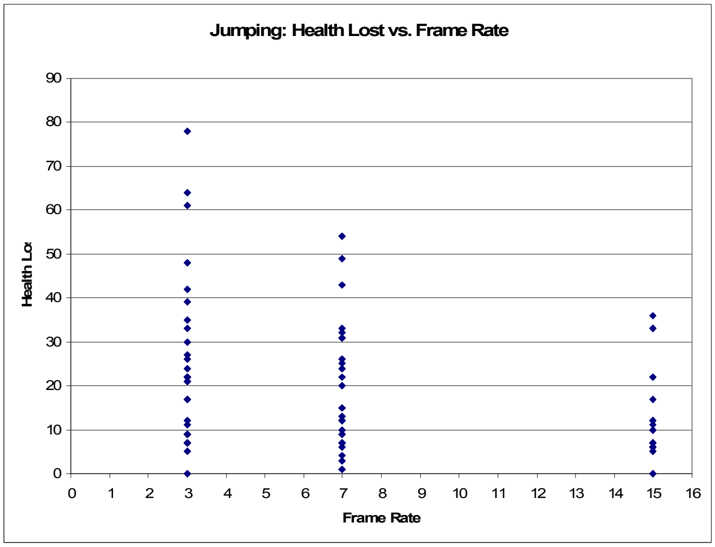
Time vs. Frame Rate
Scatter Plot



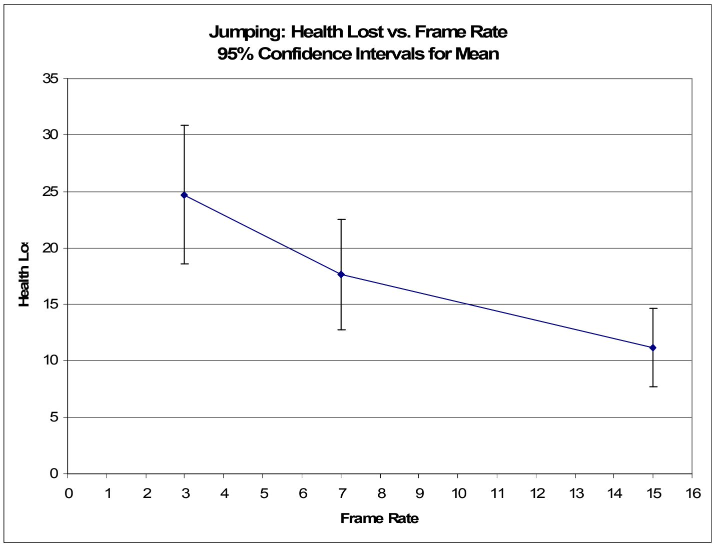
Time vs. Frame Rate
95% Confidence intervals for mean



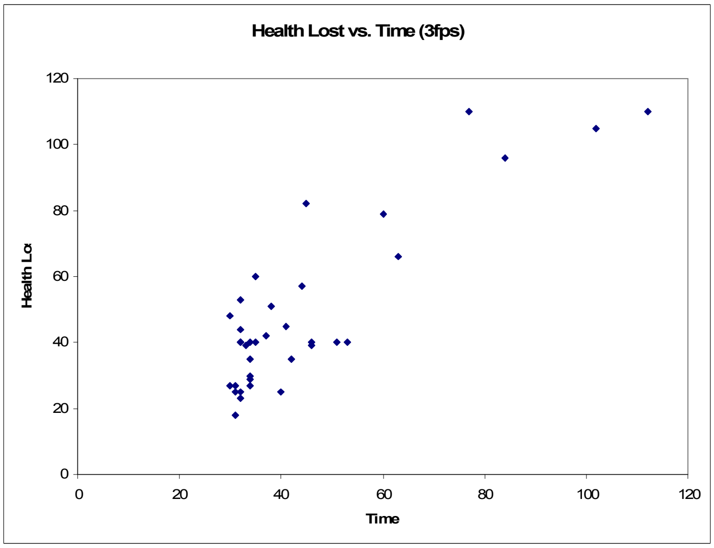
Hsealth Lost vs. Frame Rate
Scatter Plot



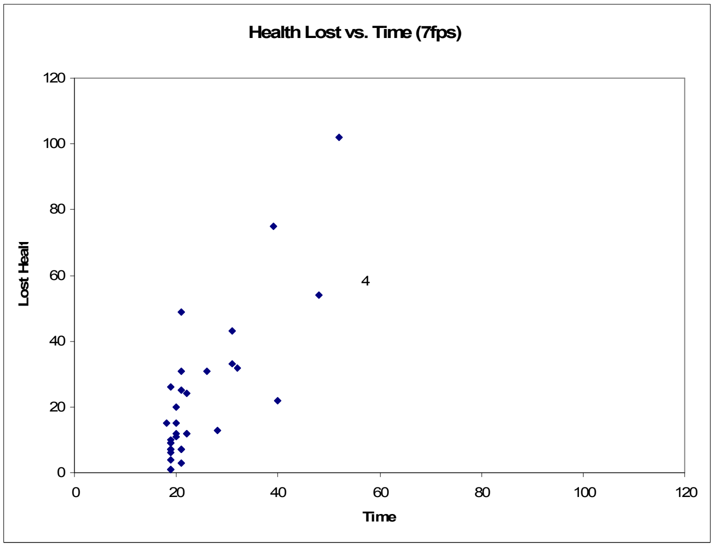
Health Lost vs. Frame Rate
95% Confidence intervals for mean



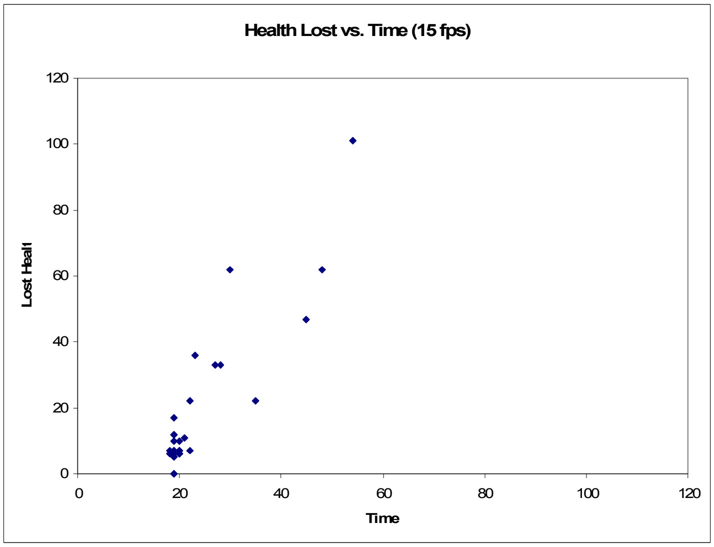
Health Lost vs. Time (3 fps)
Scatter Plot



Health Lost vs. Time (7 fps)
Scatter Plot

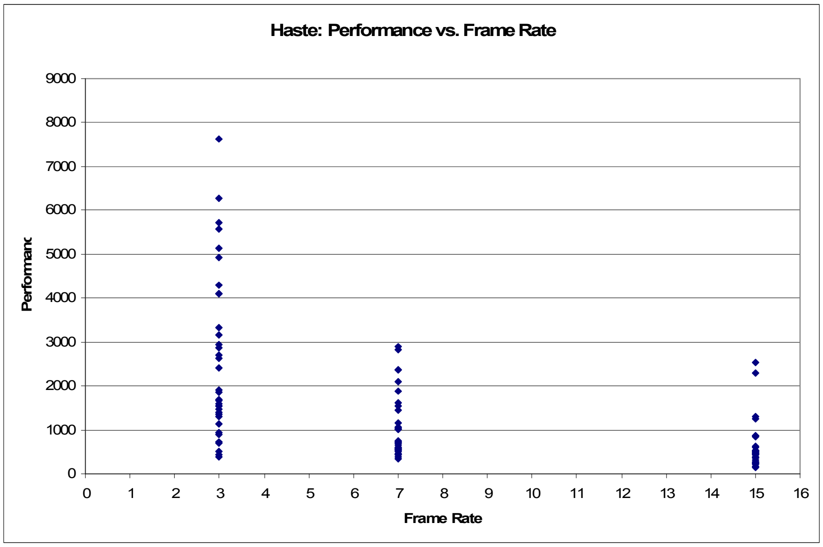


Health Lost vs. Time (15 fps)
Scatter Plot

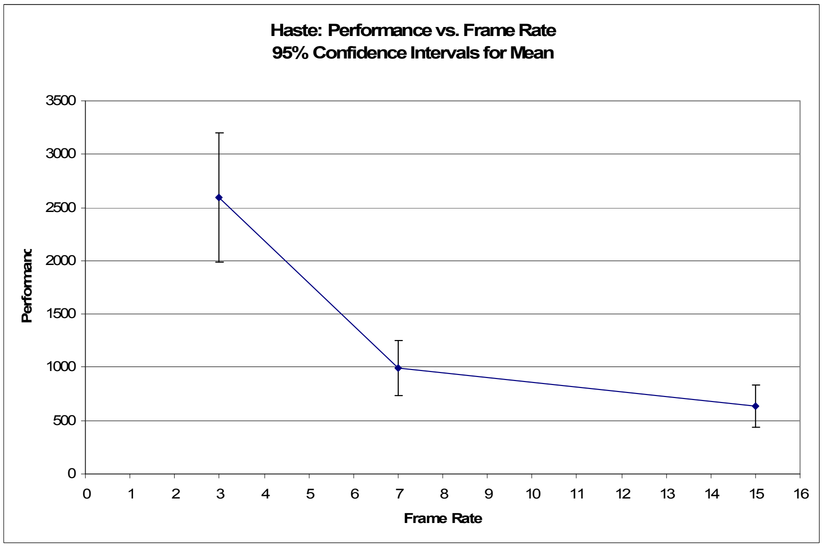


D.3 Haste

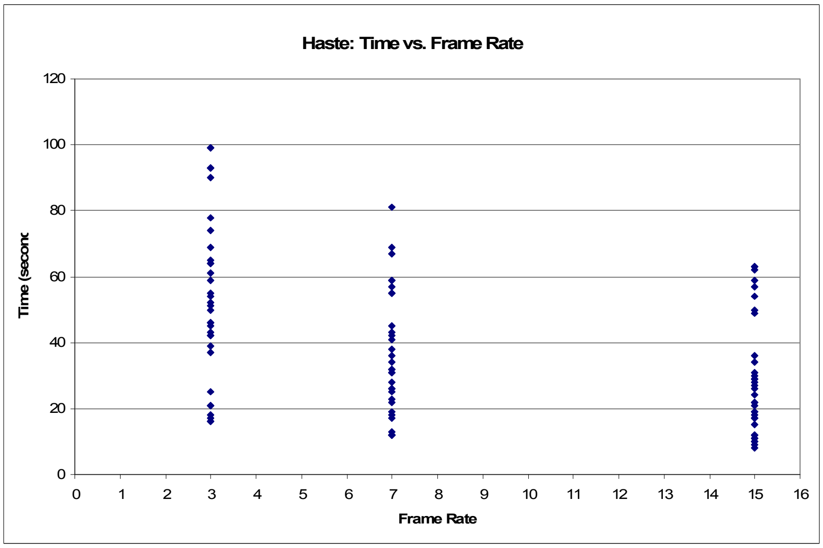
Performance vs. Frame Rate Scatter Plot



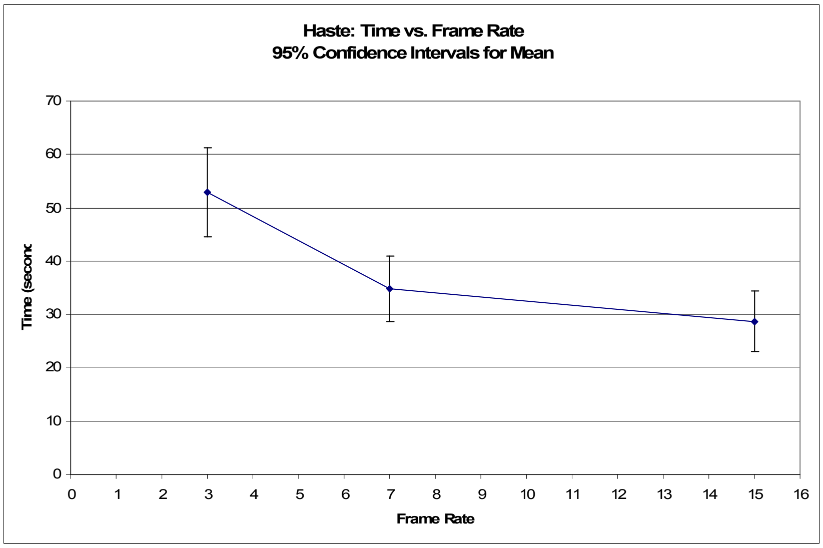
Performance vs. Frame Rate 95% Confidence intervals for mean



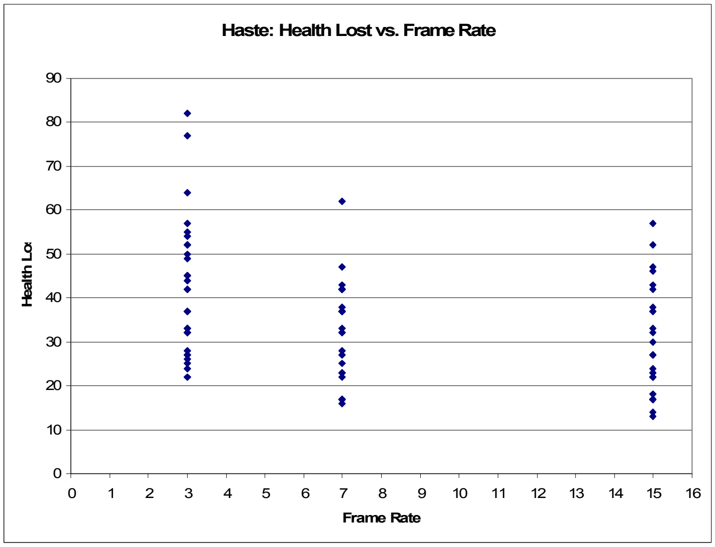
Time vs. Frame Rate Scatter Plot



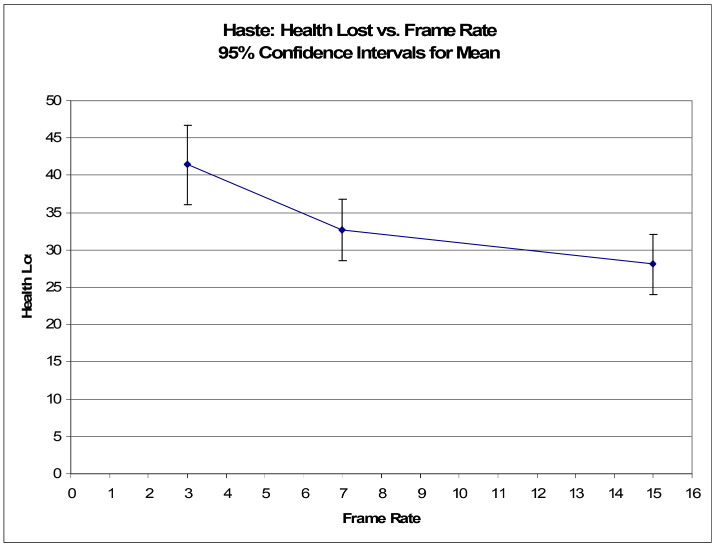
Time vs. Frame Rate
95% Confidence intervals for mean



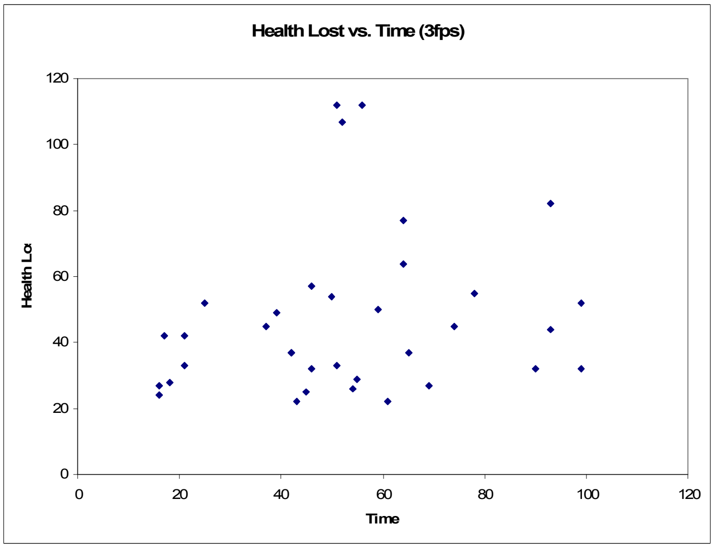
Health Lost vs. Frame Rate
Scatter Plot



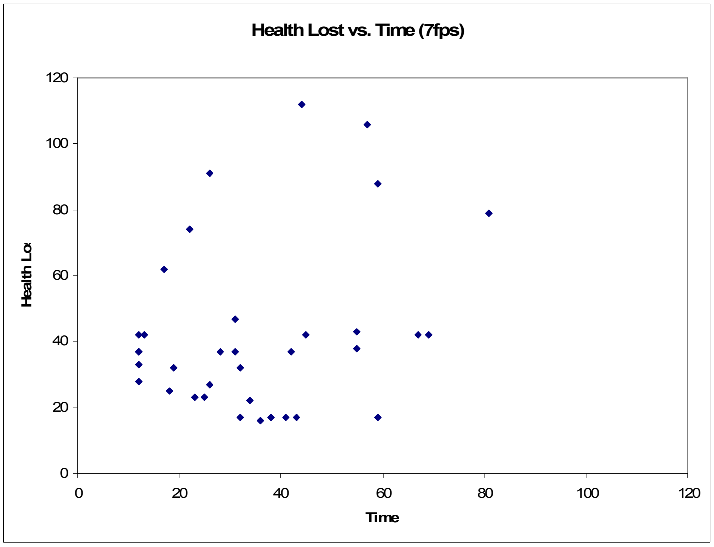
Health Lost vs. Frame Rate
95% Confidence intervals for mean



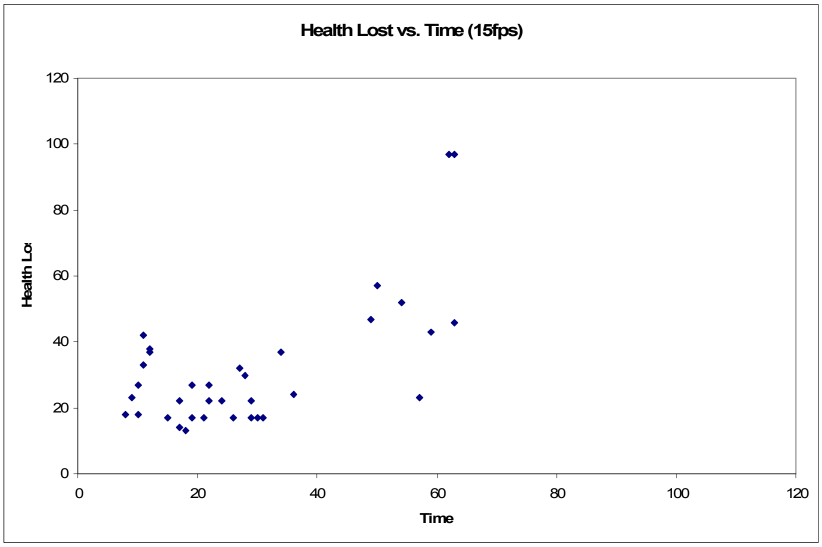
Health Lost vs. Time (3 fps)
Scatter Plot



Health Lost vs. Time (7 fps)
Scatter Plot

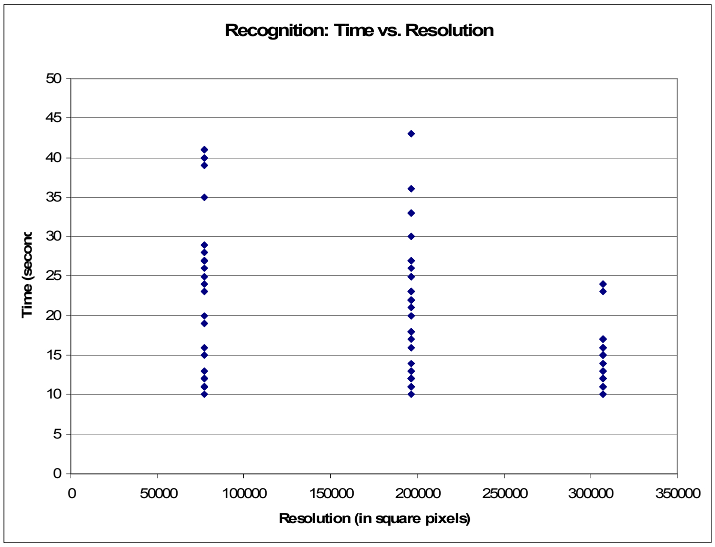


Health Lost vs. Time (15 fps)
Scatter Plot

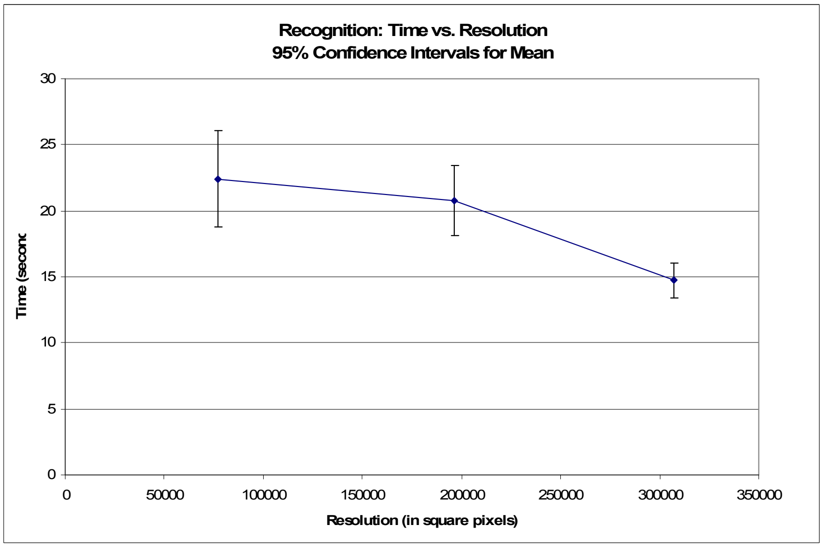


D.4 Recognition

Time vs. Resolution
Scatter Plot

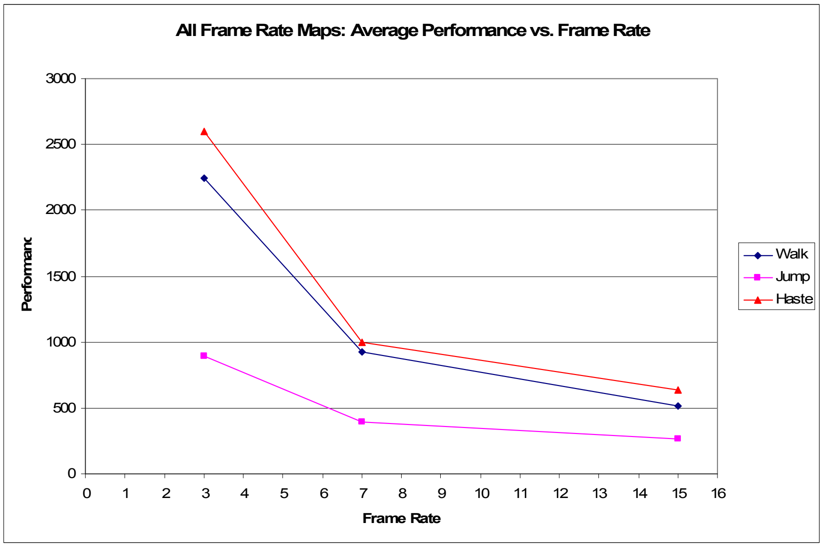


Time vs. Resolution
95% Confidence intervals for mean

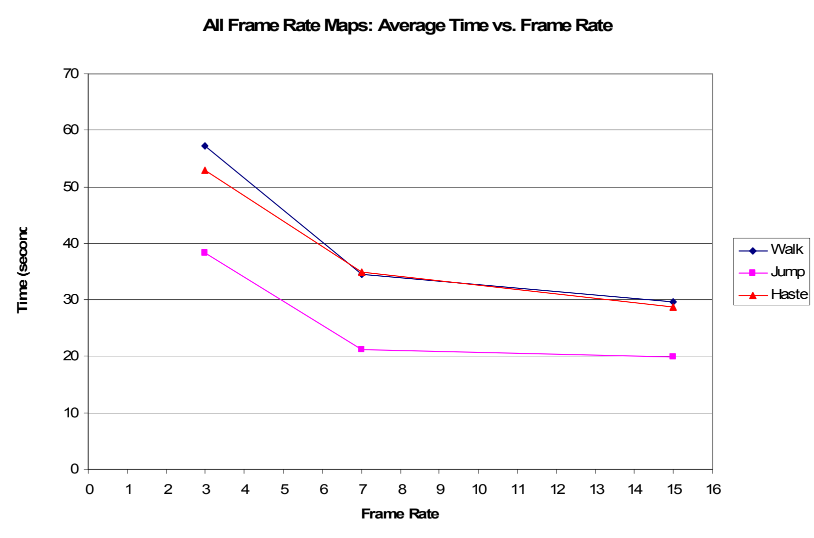


D.5 Combined Frame-Rates

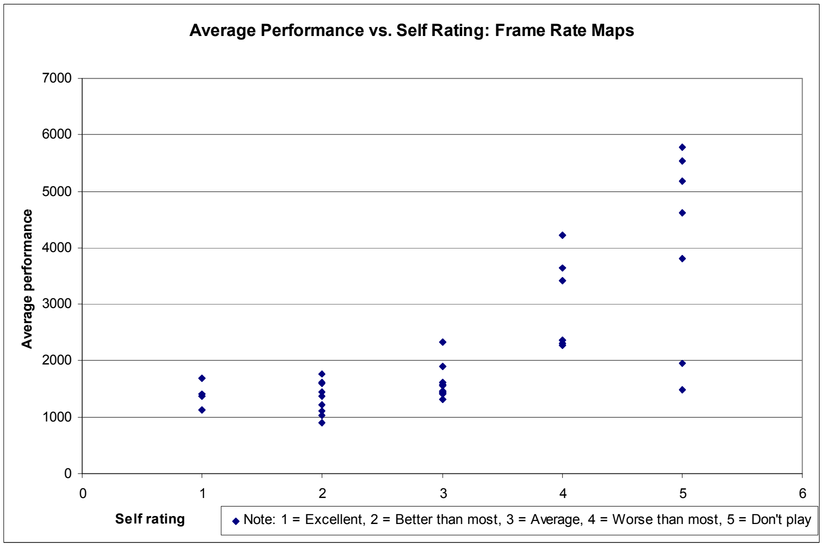
Average Performance vs. Frame Rate
Line graph of means



Average Time vs. Frame Rate
Line graph of means



D.6 Other

Average Performance vs. Self Rating: Frame Rate Maps	

Average Performance vs. Self Rating: Frame Rate Maps
Average performance
Self rating
Note: 1 = Excellent, 2 = Better than most, 3 = Average, 4 = Worse than most, 5 = Don't play