Comp 369 - Assignment 3

Research on Game Backgrounds

Professor: Dr. Walter Ridgewell

Name: Daniel Tran

ID: 3339325

Date: September 14, 2017

The early development of games presented hardware limitations that restrict the creation of detailed backgrounds. This was seen on the Atari 2600 and some of it’s 2D games such as Pac-Man and Atari Breakout. The games were primitive, and the lack of background detailed allowed games to be developed easier, with more focus on the gameplay than looks. The progression of technology allowed for 2D games to significantly increase the detail in game backgrounds, using effects such as parallax scrolling to introduce a depth of field in the games. This technology was utilized in games such as Super Mario Bros 3 and Metroid Fusion. Later on, the development of 3D games became more prevalent. Many 3D games use skyboxes to display their backgrounds to provide a realistic, infinite span for the environment that the player plays in. Games like Skyrim use skyboxes to display backgrounds such as the sky, which can change depending on the time of day. Some 3D games utilize 3D models in their backgrounds, although the backgrounds are more 2D-like. Games such as Super Smash Brothers 4 exhibit this feature. Backgrounds developed in 3D games are made to feel more realistic to immerse the players in the game and game’s story.

Game backgrounds have evolved as technology has progressed through the years. Games are becoming more detailed and intricate with the development of better game engines, stronger computers. These advancements in technology have affected different gaming environments, such as 2D, 3D, and Virtual Reality games. Since these technologies allow for better rendering, it is imminent that game backgrounds are going to progress along with these technologies. This progression is important, as they add details to the game that add to it’s gameplay without directly interacting with the player. This is done by making the game more immersive and helps set the atmosphere of the game to match what the director intended. Thus, we will be observing the different backgrounds that were available at different era’s and will discuss the pros and cons to each type of background, the technologies required to implement such backgrounds, and a couple of games that have implemented such backgrounds.

The first discussion will be based off the examination of the backgrounds present in one of the earliest development of games. At the time of creation for these games, we are looking at systems such as the Atari 2600 (also known as the VCS). There are other systems similar to the Atari 2600, but the focus will be on this device in order to keep the discussion more concise.The Atari 2600 have relatively limited functionality and processing power, as Kohler states “unique hardware limitations forced game designers to jump through all sorts of hoops to squeeze more complex game designs out of the VCS” (Kohler, Chris, 2013). The games around this time were very primitive 2D games, where the backgrounds were generally one flat color or shade, such as black or blue, with little to no details in the backgrounds. This was due to the hardware limitations of the systems. The systems had difficulty displaying color properly as well, where things such as “white text on a black background appears almost rainbow-like in nature on color machines has to do with pixel placement” (Seppala, Timothy, 2015). This was one of the factors that made it difficult to display multiple colors on the screen next to each other, due discoloration. Another factor that affected the backgrounds and gameplay was that the VCS was designed with 128 bytes of RAM and did not have a frame buffer. Thus, “programmers had to generate graphics literally in real time, drawing on the screen as the television screen's electron gun was passing over the tube” (Kohler, Chris, 2013). In other words, this means that instead of drawing the sprites and images to RAM and displaying the images from RAM onto the screen, “2600 only has a very limited time to draw (which happens to be the time it takes the electron beam in your television to get from one side to the other)” (Nukey Shay, 2011). Another limitation was that the Atari 2600 didn’t have a separate graphics processor, so all of the graphics processing had to be done on the CPU (Seppala, Timothy, 2015). Thus, the technology required to create these simple backgrounds was a T.V. set with a Cathode ray tube that could transmit electrons through the screen and interact with the pixels on the screen as they were programmed. Some games that demonstrate this are Pac-man and Atari Breakout. Pac-man was originally developed for the Atari 2600 on March 16, 1982 (Wikipedia, n.d.), while Atari Breakout was developed on May 13, 1976 (Atari, n.d.). These games are both 2D games, but they share similarities in the backgrounds, which were backgrounds consisting of one color. An advantage to having backgrounds like this is that the simplicity allows the players to focus directly on the gameplay without anything in the background that could distract them. As well, this would decrease the difficulty of producing games because the programmers wouldn’t need to worry about trying to display aesthetically pleasing backgrounds in the time constraints presented by the system. They also wouldn’t have to deal with any discoloration of the backgrounds. A disadvantages to this is that the games are less immersive because the backgrounds don’t pertain directly to the game and provide nothing for the player in terms of aesthetics. The lack of change in scenery can cause the game to feel repetitive or boring.

The next type of background under discussion is more recent but still relatively old. The context is 2D backgrounds that are present in older systems. Specifically, the type of backgrounds that will be discussed pertain to systems such as the SNES and Gameboy Advanced. These backgrounds are more detailed and have greater variety of colors in comparison to those discussed on the Atari 2600 platform. These details allow the game to be more aesthetically pleasing and immersive for the player. Many of the backgrounds in these games had parallax scrolling which “is a simple and effective way to create the illusion of depth in a 2D game” (Bone, Sonny, 2014). Parallax scrolling is defined as is “the apparent displacement of an observed object due to a change in the position of the observer” (Bone, Sonny, 2014). There are multiple methods that can be used to achieve this effect, such as the layer method, row/column scrolling, sprite method, animation method, or the raster method. One of the most common methods is the layer method, which moves the backgrounds and parts of the background at different than what the player is moving at. Parts of the background and also move at different rates than each other if these elements are on different layers. There can also be foreground layers that move to cover the user to create another depth of perception (Jagged85, 2015). These layers allow for easier control of speed and position of different elements in the scene. One of the other methods I want to discuss is the animation method, which is defined as “Scrolling displays built up of individual tiles can be made to 'float' over a repeating background layer by animating the individual tiles' bitmaps in order to portray the parallax effect” (Jagged85, 2015). The reason why these two techniques are discussed in more detail is because I believe they are the achievable in terms of implementation using the current knowledge discussed in the textbook. Since Mappy and MappyAL have inherently have different layer properties for each block, it is possible to use the layers to create a parallax scrolling effect (Neil Walker, 2009). Allegro also uses sprites, which allows the sprite method of parallax scrolling to effectively work as well. Some games that took advantage of this was Super Mario Bros 3 and Metroid Fusion. Super Mario Bros 3 was released on July 11, 2003 (DeF, 2016) for the Super Nintendo Entertainment System (SNES) and Metroid Fusion was released on November 19, 2002 (BeachThunder, 2015) on GameBoy Advanced (GBA). Both of these games utilize parallax scrolling to create a more immersive game by correlating different backgrounds to different levels to change the scenery throughout the game. An advantage to having backgrounds like the ones seen in these games are that they are more immersive for the player, presenting backgrounds that correlate with the gameplay to create an effect of a more polished game. A disadvantage is that it takes more time to develop these backgrounds, and more people are required to be dedicated to creating the artwork for them. As well, developers of games are now expected by the players to implement these kinds of backgrounds in each game that is released, otherwise the game would not look up to par in comparison to other games. This might also be a good thing, as it forces companies to put more effort into creating higher quality games. Both the SNES and GBA have far greater hardware capabilities than what was previously discussed on the Atari 2600. This advancement in hardware allows the detailed backgrounds seen in these games to be created and displayed. To get a better understanding of the advancements in hardware, we will take a look at the hardware specifications of each of these devices, in comparison to the Atari 2600. The Atari 2600 had an 8-bit CPU, 128 bytes of RAM, no VRAM, could display 4 colors at once and 5 sprites on the screen at any moment (problemkaputt, n.d.). The SNES had a 16-bit CPU, 128 kilobytes of RAM, 64 kilobytes of VRAM, could display 256 colors at once and 128 sprites on the screen at any moment (snescentral, n.d.). The GBA had a 32-bit CPU, 256 kilobytes of DRAM, 96 kilobytes of VRAM, could display 512-1024 colors depending on the mode, and have 256-1024 sprites on the screen at any moment (also depending on the mode) (nintendo.wiki, n.d.) (gamedev, n.d.). From looking at these hardware specifications of each of these devices, it is clear how hardware can affect the development of graphics and visuals for video games. We can see how devices like the SNES and GBA are more capable and provide developers with more to work with in terms of available processing power, colors, and sprite limitations, which make it possible for developers to allocate more resources into creating visually appealing backgrounds for their games.

The final discussion for backgrounds will be based on backgrounds that are present in 3D games. 3D games have become more and more popular with the rise of better technologies and development of better game engines. The backgrounds that are present in these games are generally more defined and have a greater number of effects and moving objects to help create a more realistic environment. This amount of detail allows games to be more immersive, as backgrounds contribute more to developing the correct atmosphere for the games. Many 3D games create 3D backgrounds by using skyboxes. A skybox is “a cube made of six images that surrounds the game player” (Bell, Gavin, 1998); the skybox can also be called the cubic model if it uses six images to create the skybox. The skyboxes are drawn to create the effect of an infinite world, but will only create this effect if done correctly. Some difficulties bestowed upon the developers of these skybox backgrounds are that they have to be careful that “the borders between the skybox images match exactly, [and] you create images with the [correct] perspective” (Bell, Gavin, 1998). An alternative to using the box model for skyboxes is using a spherical model, which consists of a single image, rather than six images to create a cube. This is useful in creating a dome-like effect, which is useful in simulating backgrounds for the sky. A downside to using a spherical model is that the player’s view (which consists of straight lines in a cubic map), become curved in a spherical map. These curved lines create distortions in the backgrounds at different depths in the images. A reason that some developers may use a spherical model is that it is easier to work with, since they only have to deal with a single image, rather than six images in the cubic model. Although this may be an attracting reason to use a spherical model, it may not be a viable solution if hardware limitations are a great concern, since “interactive graphics hardware and software is designed to display flat surfaces, so cubic environment maps can be drawn quicker than spherical maps” (Bell, Gavin, 1998). A couple of 3D games that utilize impressive backgrounds or skybox backgrounds are Super Smash Bros 4 (SSB4), which was released on October 3, 2014 (Wikipedia, n.d.) for the Wii U and Nintendo 3DS, and Skyrim for the PC, XBox360, and PS3, which was released on November 11, 2011 (Wikipedia, n.d.). Although SSB4 is played as a 2D game, the models in the game are 3D, which allows the game to be considered a 3D game (gamefaqs, 2009). There are many maps that the players can play one, each yielding different backgrounds to match the battlefield. Some notable examples are Final Destination (ssbwiki, n.d.) and Battlefield (ssbwiki, n.d.). In the game, these backgrounds display extraordinary artwork, with immense amounts of details, effects, particles and 3D models that contribute to making the game more aesthetically pleasing. The backgrounds in the map also change color tones overtime, to simulate the passing of time. On the Battlefield map, this is apparent, as the background between day and night cycles. In Skyrim, the effects of skyboxes are much more apparent. In the game, you are a character that explores a 3D world. A great example of the use of skyboxes in the game is the sky in Skyrim, as well as objects off in the distance, such as mountains. Although the background in this game is not interactible for the user, it plays a big role in the game. The background allows the user to visually know what time of day it is, whether it is day or night time, which adds a lot to the gameplay in terms of realism. Although 3D backgrounds and skyboxes are important in creating an immersive environment in Skyrim, it also introduces the possibilities for bugs, which can ruin the whole experience. There are threads that discuss player’s problems with glitches in the skyboxes, where the player can see the edges between the different sides of the cubic model (King Jizzard & Future Wizard, 2014). As stated above, the effect of skyboxes to create the effect of an infinite world can only be done if implemented properly. This is one of the disadvantages of having 3D backgrounds that use skyboxes. It requires the developers to spend more time creating and debugging the backgrounds to avoid situations like the ones seen in the Skyrim thread. Also, these backgrounds are requiring specialized softwares, such as 3DS Max. Another disadvantage to having these kinds of backgrounds is that games become more CPU intensive, causing players to have to have more powerful computers or devices to play the games, making the games unaccessible to all users. The advantage of having these kinds of background is once again, aesthetics and creating an immersive environment for user. These kinds of backgrounds in the 3D environment make the games more realistic and appealing for the players.

Overall, backgrounds are an important aspect in games. Although they are not required, they have become a staple expectation in today’s gaming industry. It was seen that in the past, hardware for games was a bottleneck for developing game backgrounds. This, however, did not limit the game’s playability, but the aesthetics of the game. As technology in hardware and software progressed, it created more room for developers to create more intricate and attractive backgrounds that allow the player to become more engaged in the gameplay and game’s story. There are differences in the technologies used in 2D and 3D games, although they can run on the same hardware or use the same softwares to develop them. For instance, 2D games utilize parallax scrolling in order to simulate the illusion of depth, while 3D games utilize skyboxes to create detailed backgrounds for scenery in the player’s perspective.

# References

Atari (n.d.) Date that Atari Breakout was released. Retrieved from

<https://www.atari.com/history/1972-1984-0>

BeachThunder (2015) Release date of Metroid Fusion. Retrieved from

<https://www.giantbomb.com/metroid-fusion/3030-10323/>

Bell, Gavin (1998) What is a skybox and the difference between the cubic model and the spherical model in skyboxes. Retrieved from

<https://www.gamasutra.com/view/feature/131701/creating_backgrounds_for_3d_games.php>

Bone, Sonny (2014) Parallax scrolling used to create illusion of depth in 2D games. Retrieved from

<https://gamedevelopment.tutsplus.com/tutorials/parallax-scrolling-a-simple-effective-way-to-add-depth-to-a-2d-game--cms-21510>

DeF (2016) Release date of Super Mario Bros 3. Retrieved from

<https://www.giantbomb.com/super-mario-bros-3-super-mario-advance-4/3030-20748/>

Gamedev (n.d.) GBA hardware specifications (colors and sprite limitations). Retrieved from

<https://www.gamedev.net/forums/topic/74146-gba-dev-and-tile-sizes/>

Jagged85 (2015) Layer method for parallax scrolling. Retrieved from

<https://www.giantbomb.com/parallax-scrolling/3015-2915/>

Jagged85 (2015) Animation method for parallax scrolling. Retrieved from

<https://www.giantbomb.com/parallax-scrolling/3015-2915/>

King Jizzard & Future Wizard (2014) Skyrim skybox glitch where the edges of the cubic model can be seen.

<https://steamcommunity.com/app/72850/discussions/0/617320628007926479/>

Kohler, Chris (2013) VCS’ unique hardware limitations forced game designers to jump through all sorts of hoops to squeeze more complex game designs out of the VCS. Retrieved from

<https://www.wired.com/2009/03/racing-the-beam/>

Kohler, Chris (2013) programmers had to generate graphics literally in real time, drawing on the screen as the television screen's electron gun was passing over the tube. Retrieved from

<https://www.wired.com/2009/03/racing-the-beam/>

Neil Walker (2009) Mappy and MappyAL for parallax scrolling. Retrieved from

<https://www.allegro.cc/forums/thread/599933>

Nintendo.wiki (n.d.) GBA Hardware specifications. Retrieved from

<http://nintendo.wikia.com/wiki/Game_Boy_Advance>

Problemkaputt (n.d.) Atari 2600 hardware specifications. Retrieved from

<http://problemkaputt.de/2k6specs.htm>

(Seppala, Timothy, 2015) Atari 2600 didn’t have a dedicated graphics processor and required the CPU to perform the processing. Retrieved from

<https://www.engadget.com/2015/08/31/apple-ii-atari-graphics-explainer/>

(Seppala, Timothy, 2015) Pixels displayed on the screen had discoloration due to pixel placement. Retrieved from

<https://www.engadget.com/2015/08/31/apple-ii-atari-graphics-explainer/>

Snescentral (n.d.) SNES hardware specifications. Retrieved from

<https://snescentral.com/article.php?id=0088>

Ssbwiki (n.d.) Final Destination map. Retrieved from

<https://www.ssbwiki.com/Final_Destination_(SSB4)>

Ssbwiki (n.d.) Battlefield map. Retrieved from

<https://www.ssbwiki.com/Battlefield_(SSB4)>

Nukey Shay (2016) Time for images to be displayed was very limited and depended on the time that it took for an electron beam to travel across the screen. Retrieved from

<http://atariage.com/forums/topic/178563-why-cant-the-atari-2600-display-better-graphics/>

Wikipedia (n.d.) Release date of Pac-man on Atari 2600. Retrieved from

<https://en.wikipedia.org/wiki/Pac-Man_(Atari_2600)>

Wikipedia (n.d.) Release date of SSB4. Retrieved from

<https://en.wikipedia.org/wiki/Super_Smash_Bros._for_Nintendo_3DS_and_Wii_U>

Wikipedia (n.d.) Release date of Skyrim. Retrieved from

<https://en.wikipedia.org/wiki/The_Elder_Scrolls_V:_Skyrim>