Comparative Study and Implementation of Camera Control Techniques in 3D Games

Student: Yuhan Wang, 101165518
Supervisor: Oliver van Kaick

I. Introduction

In 3D video games, camera control plays a pivotal role in influencing a player's experience and immersion within the virtual environment. The camera acts as the player's eyes, offering a window into the gaming world and shaping their perception of the unfolding narrative. How the camera is controlled can impact gameplay dynamics, visual storytelling, and overall engagement. Consequently, the design and implementation of the camera control system in 3D games are crucial for creating a captivating gameplay experience.

During the era of 2D games, camera usage was straightforward, with static visuals and scrolling screens meeting most 2D gaming needs. However, 3D games, ranging from RPGs to action games to FPS and more, each require distinct camera behaviors, harmonizing with their gameplay mechanics, narrative intent, or artistic vision. The design of camera control systems that cater to these diverse requirements highlights the complexities of this domain.

This project aims to study the various methods of camera control in 3D video games, exploring, comparing, and implementing different camera control techniques. Through research and

analysis, this project endeavors to unearth the pros and cons of different camera control methodologies.

In the subsequent sections, this report will outline the project's objectives and describe the planned studies, implementation, tests, and comparative analyses of various camera control methods. The report will conclude by synthesizing the research findings, emphasizing the applicability of different camera control techniques across various game genres. Primarily, this report delves into the categorization, philosophies, and comparisons of 3D gaming perspectives, without diving deep into code or algorithmic analyses.

II. Objectives

2.1 Research and Analysis

The research phase of this project aims to comprehensively understand the myriad camera control techniques employed across different 3D games. This process will encompass a spectrum of mainstream game genres, ensuring a broad representation of camera behaviors.

My objective is to find out how lens control influences gameplay and the player’s experience. I will consult various resources, including academic papers and game content, in conjunction with each selected game under study.

2.2 Implementation

Based on the research phase, the emphasis during implementation will be on translating theoretical knowledge into practical application. Drawing from accumulated research findings, camera control techniques will be chosen and demonstrated within a sample Unity game scene.

2.3 Comparative Study

In this phase, tests will be conducted on implemented methods to gauge their viability, and comprehensive comparison will also be made. This comparison seeks to find the unique strengths and limitations of each method.

2.4 Conclusion

The concluding phase will encompass a synthesis of all preceding research and analyses.

III. Work Description

3.1 Identification

Broadly speaking, 3D game camera systems can be classified into two perspectives: third-person and first-person. The third-person perspective can further be categorized into fixed third-person and dynamic third-person views. It's worth noting that the perspective in a game is not static; some games employ different views to achieve varied effects. Nevertheless, this report will discuss these perspectives separately for contrast.

3.2 Analysis

3.2.1 First-Person Perspective

In the first-person perspective, the camera typically presents a view of the world as perceived from the eyes of the character controlled by the player. The player's movement is synchronized with the camera's movement, following the character's line of sight. Taking Counterstrike as an example, when the player moves the character to the right and adjusts the crosshair upward, the camera always points in the direction the player is facing (Refer to Figure 1, Figure 2). Hence, in general, the first-person perspective camera is easier to implement compared to the third-person perspective.

Figure 1

Figure 2

3.2.2 Fixed Third-Person Perspective

The third-person perspective is like a monitor camera's view. In this frame, players can see the character they control along with other elements in the scene. Typically, in third-person games, the camera position is separate from the player's character, placed behind or above the character. This perspective can be divided mainly into two types: fixed third-person and dynamic third-person perspectives.

In the fixed third-person perspective, the camera remains stationary, capturing a singular screen, area, or image. This perspective is more commonly used in 2D games, but that doesn't mean it's not used in 3D games. Early 3D games like "Resident Evil" have a fixed third-person perspective, switching camera positions based on the character's location. In a scene, as Chris enters a dining room and walks deeper (Figure 3), the viewpoint switches (Figure 4). If Chris walks near the

wall side, the view changes to an angle above the grandfather clock allowing players to inspect the clock and paintings (Figure 5). If Chris walks by the dining table, the viewpoint remains unchanged. Both routes eventually lead to a viewpoint facing the depth of the dining room (Figure 6). Due to the limited capabilities of the PS1, CAPCOM decided to overlay 3D character models on pre-rendered 2D static backgrounds. As a result, the game had to frequently change the fixed camera angles since the entire environment wasn't created in 3D (Turi, 2015). On the other hand, since players couldn't freely adjust the viewpoint, they would often encounter unknown enemies or jump scares, adding to the game's tension and horror.

Figure 3

Figure 4

Figure 5

Figure 6

In 3D games, the overhead fixed third-person perspective is more common. Dota is a typical example. Although players can technically move the camera's position on the single plane or zoom the viewpoint, in actual gameplay, players maintain a distant overhead view for most of the time. Thus, Dota can also be categorized under the fixed third-person perspective (Figure 7).

There are other applications of the fixed third-person perspective, such as during character dialogues with NPCs, where the camera needs to switch between close-ups of their faces.

Another example is the rear-view mirror perspective in racing games, allowing players to clearly see behind and make appropriate responses.

Figure 7, https://store.steampowered.com/app/570/Dota_2/

3.2.3 Dynamic Third-Person Perspective

In the dynamic third-person perspective, the camera can be dynamically adjusted relative to the player character's movement direction, in terms of distance or angular position. The dynamic third-person perspective is widely used in various adventure games, action games, and role-playing games. The dynamic perspective constantly focuses on the camera's orientation. Overall, factors determining the camera's orientation include:

Yaw: Rotation of the camera around the vertical axis.

Pitch: Rotation of the camera around the horizontal axis.

Roll: Rotation of the camera around its forward axis.

Look-at position: The camera's position in the world space.

View direction: The direction the camera "looks at," which is the camera's forward vector.

Taking "The Legend of Zelda: Tears of the Kingdom" as an example: The camera's initial position is behind Link (Figure 8). If player wants to move the camera to the upper left side of Link, you need to translate the camera to determine the Look-at position and rotate the camera clockwise around the vertical axis and counterclockwise around the horizontal axis to determine

the View direction, finally resulting in the perspective seen in Figure 9.

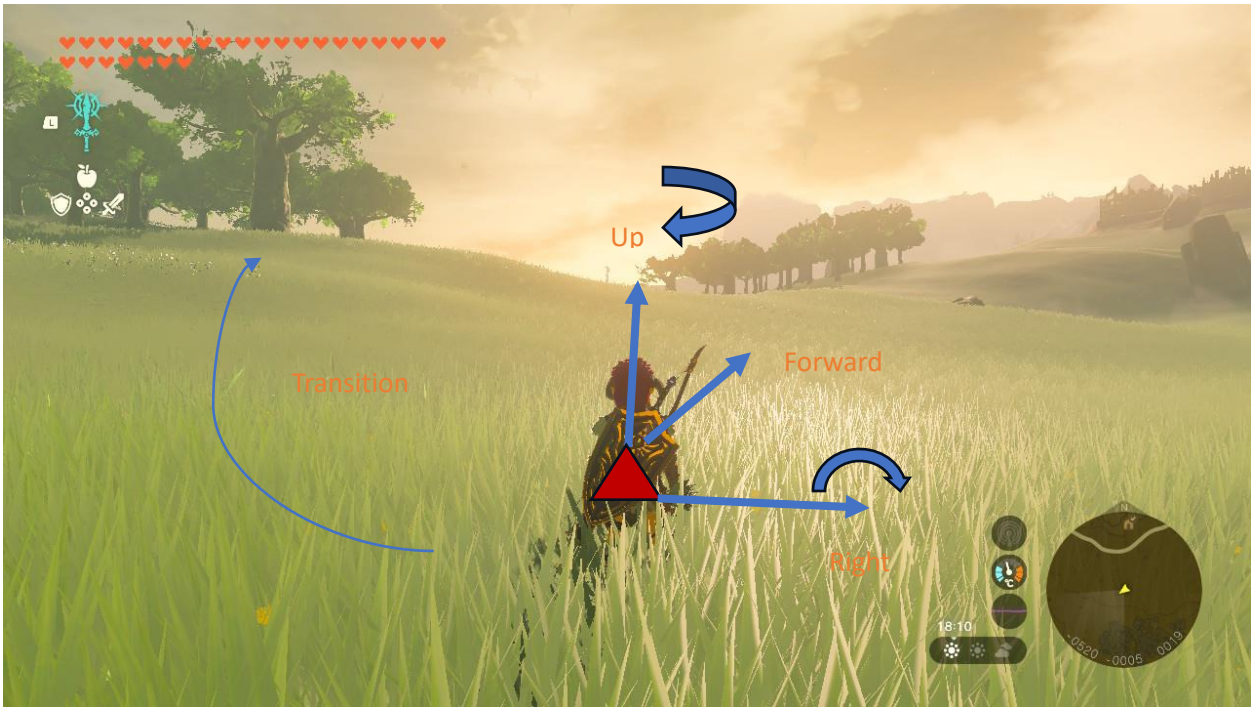

Figure 8

Figure 9

Roll is not common in the dynamic third-person perspective, mainly used in games with vehicles to enhance the player's sense of control (Figure 10).

Figure 10, https://store.steampowered.com/app/502500/ACE_COMBAT_7_SKIES_UNKNOWN/

3.2.4 Field of View

Another concept is the Field of View (FOV). In the context of computer graphics and video games, FOV refers to the extent of the scene that is visible through a camera or viewer's perspective. Using the game "Chivalry 2" as an example, compared to a smaller FOV (Figure 11, FOV = 80), a larger FOV (Figure 12, FOV = 120) allows players to see more of the game world, offering a more expansive view. This makes players feel more immersed in the vastness of the environment. However, a larger FOV might cause objects on the edge of the screen to stretch or

distort, affecting visual quality. A larger FOV might require more computational resources to render more of the scene, possibly impacting the game's performance. Some competitive-focused games might limit the advantage players can get from high FOVs.

Figure 11

Figure 12

3.3 Comparation

3.3.1 First-Person Perspective

One significant advantage of the first-person perspective is the enhanced sense of immersion. Through its unique realistic simulation and camera movements synchronized with the player's actions, players can swiftly establish a close connection with the game's protagonist. Players can naturally immerse themselves in the game, feeling as if they are the character, which greatly enhances the sense of immersion.

Another advantage of the first-person perspective is that compared to the other two perspectives, it can better emphasize the game's atmosphere in some games. It simulates the "human eye view," so players can't see the entire scene when observing the environment. For instance, some

horror games adopt a first-person view to render the game's atmosphere. In these games, as players explore unknown corners, they face uncertainties and surprises. With slow movements and subtle head turns, players might suddenly get "jump scared." This perspective effectively sets the mood, making the gaming experience more authentic.

However, the first-person perspective has some disadvantages. Firstly, its field of view is limited, potentially affecting a player's observation of their surroundings. "Mount & Blade II: Bannerlord" allows players to freely switch between first-person and dynamic third-person perspectives, serving as a good example. Clearly, in the first-person perspective (Figure 13), players can't see the NPC on the right, whereas the third-person view (Figure 14) easily allows this. Moreover, in the first-person view, the player's sight can be partially obscured by the horse's head, further limiting their field of view.

Figure 13

Figure 14

Additionally, the first-person perspective can distort players' distance judgment. As it simulates human visual experience, players might lack clear reference to measure distances between avatar and the environment. In the first-person view (Figure 15), players might struggle to determine if they can hit an NPC, whereas the third-person view (Figure 16) can easily determine the attack

range of the player's weapon and the distance between the player's character and the NPC.

Figure 15

Figure 16

Moreover, the first-person perspective can lead to 3D dizziness more easily. Studies have shown that first-person perspectives can cause dizziness more than third-person perspectives (Monteiro, et al., 2018). Thus, game developers need to consider this when deciding on the game's perspective.

3.3.2 Fixed Third-Person Perspective

As mentioned earlier, in games like Dota or some simulation management games, a distant fixed third-person perspective can offer players better control over the entire game scene and strategy.

Another use of the fixed third-person perspective in games is to guide the player's view via the camera. By showing arranged scenes and events, it is effective to catch the player's attention. Due to the fixed viewpoint, players can only focus on the direction pre-set by the game designers. From this point, designers can carefully organize in-game resources and events, propelling the player's progression and enriching their gaming experience. The previously mentioned "Resident Evil" is a prime example of this.

Despite its strengths, the fixed third-person perspective is not without its limitations. A notable drawback is that it may diminish the sense of immersion and identification with the game character. Since players are presented with a distant and predetermined view, they might feel disconnected from the in-game world, lacking the direct and immediate engagement found in other perspectives. The result can be a reduced sense of personal involvement and immersion, which are critical aspects for many gamers in their gaming experience.

3.3.3 Dynamic Third-Person Perspective

The primary advantage of the dynamic third-person perspective is its flexibility and adaptability. The dynamic third-person view can adjust based on the movement of the player's character and the demands of the game scenario, offering greater adaptability and flexibility. This allows players to engage with the game world more naturally, whether they're exploring the environment or participating in combat, without compromising immersion.

In other words, the dynamic third-person perspective combines the strengths of both first-person and third-person views. Compared to the first-person perspective, the dynamic third-person view provides a better field of view and sense of distance (as mentioned earlier); compared to the fixed third-person perspective, the dynamic third-person perspective is closer to the player character, offering a heightened sense of immersion. Overall, the dynamic third-person perspective is a comprehensive viewpoint, which is why it's widely used in modern 3D games.

However, the adaptability of the dynamic third-person perspective also brings its downside: instability. The dynamic third-person perspective might lead to frequent camera adjustments and rotations when the game character moves, disorienting the player. The frequently changing views could unexpectedly alter the player's field of vision, affecting combat, exploration, and enemy localization.

Another drawback of the dynamic third-person perspective is the challenge of interacting with the environment. The dynamic view might cause environmental objects, like walls, to obstruct vision, making it hard for players to spot targets, lose their sense of direction, or face challenges during interactions.

Under the influence of these disadvantages, even games like "Zelda: BotW" and "TotK", which could be considered among the best of all time, sometimes suffer from poor perspectives that may leave players ponderously questioning, "Who am I?" and "Where am I?" in chaos (Figure 17).

Figure 17

From this, it can be seen that even top developers like Nintendo cannot completely avoid the shortcomings of the dynamic third-person perspective. The flexibility and adaptability of the dynamic third-person view also bring complexity. To develop an excellent dynamic third-person perspective camera control, high technology is necessary.

3.4 Implementation

In this section, I tried to create a scene in Unity with the aim of implementing a simple dynamic third-person perspective camera system.

Movement Controller.cs:

This script uses the CharacterController component to implement the basic movement of the character. Use WASD to move forward, backward, left, and right. Hold Shift to run and press the spacebar to jump.

ThirdPersonCameraController.cs, CameraCollisionTransparency:

These scripts manage the position, orientation, and zoom of the dynamic third-person camera. Use the mouse wheel to zoom the view (Figure 18, 19). Move the mouse to rotate the view. When the player tilts the view up or down, set an angle limit to prevent the camera from flipping excessively (Figure 20, 21). Use ray tracing to detect obstacles between the character and the camera. If there is, move the camera to the collision point and use transform.LookAt() to ensure the camera is facing the character (Figure 22, 23). When the camera is too close to the character, to avoid obstructing the view by getting stuck in the character model, make the character semi-transparent (Figure 24, 25). This scene implements some basic features of the dynamic third-person view and addresses the problem of obstacles obstructing the view in a simple environment.

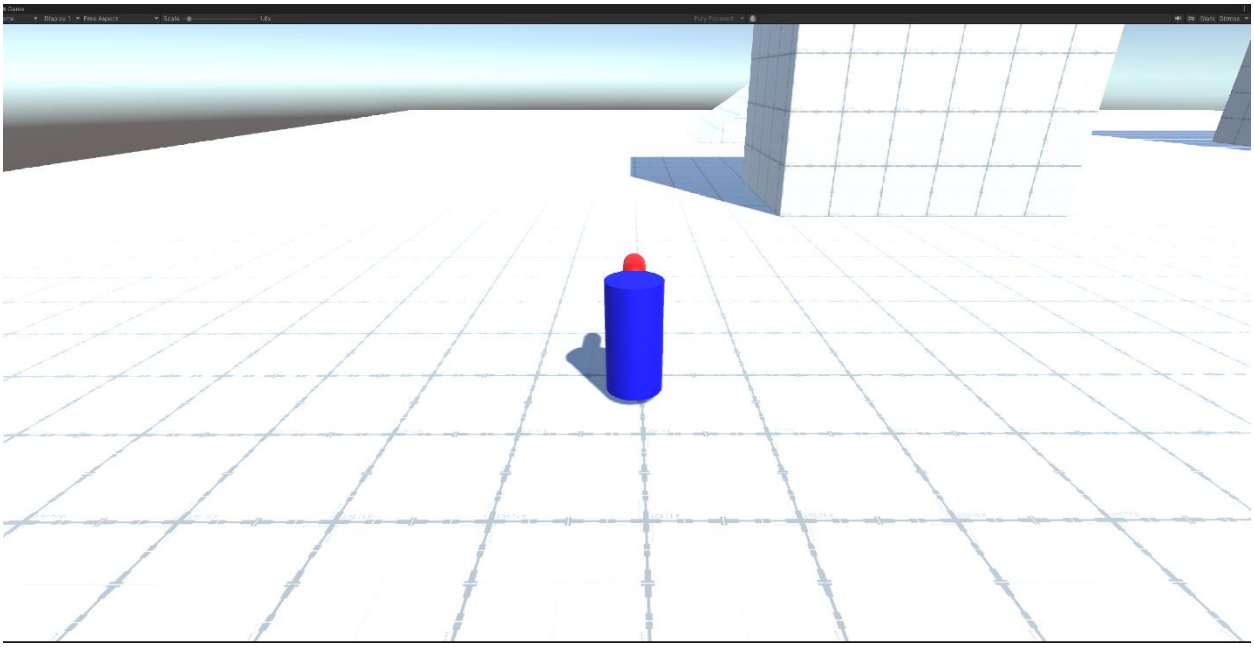

Figure 18

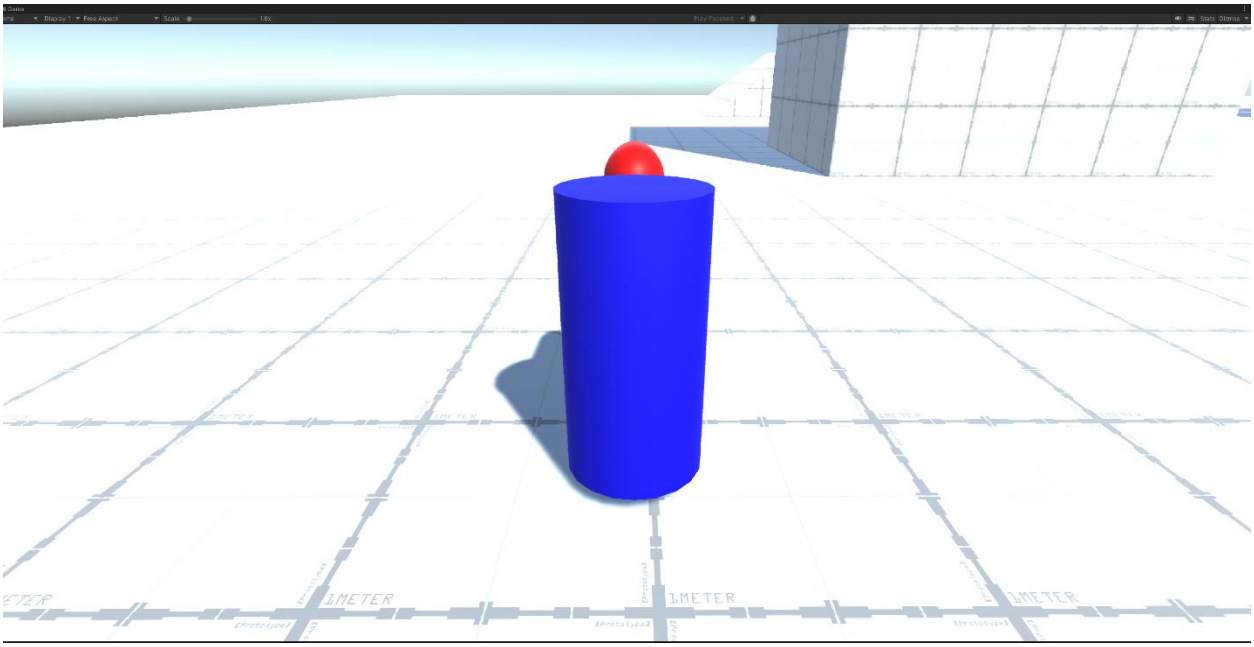

Figure 19

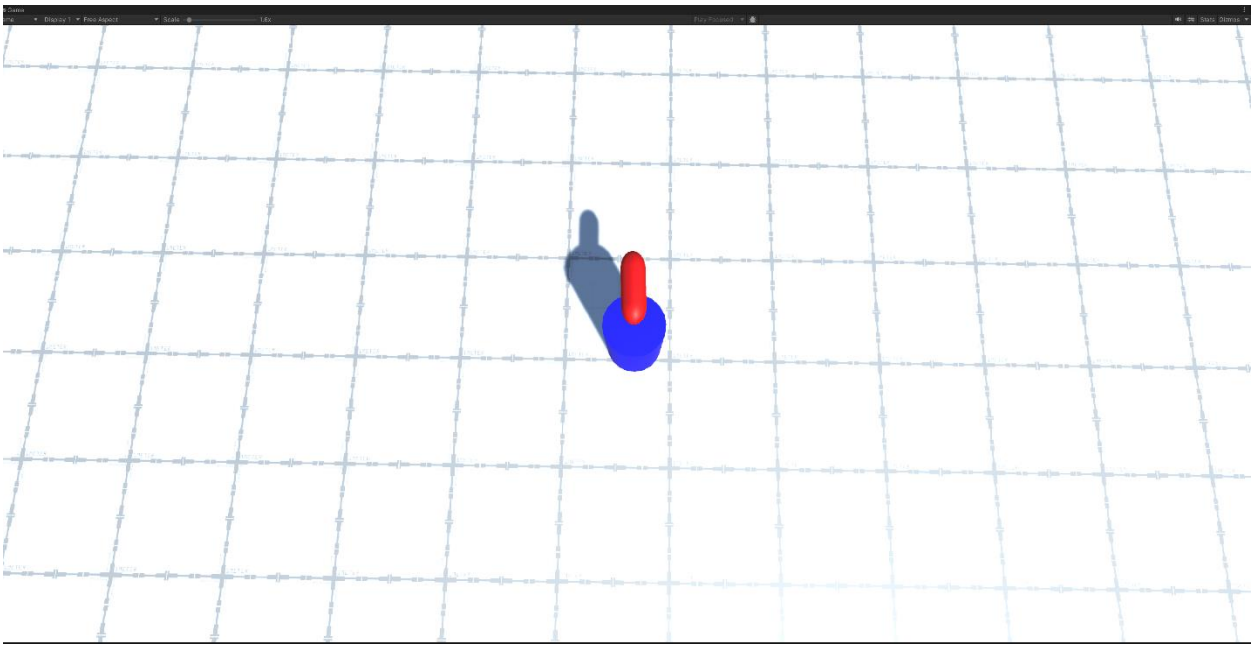

Figure 20

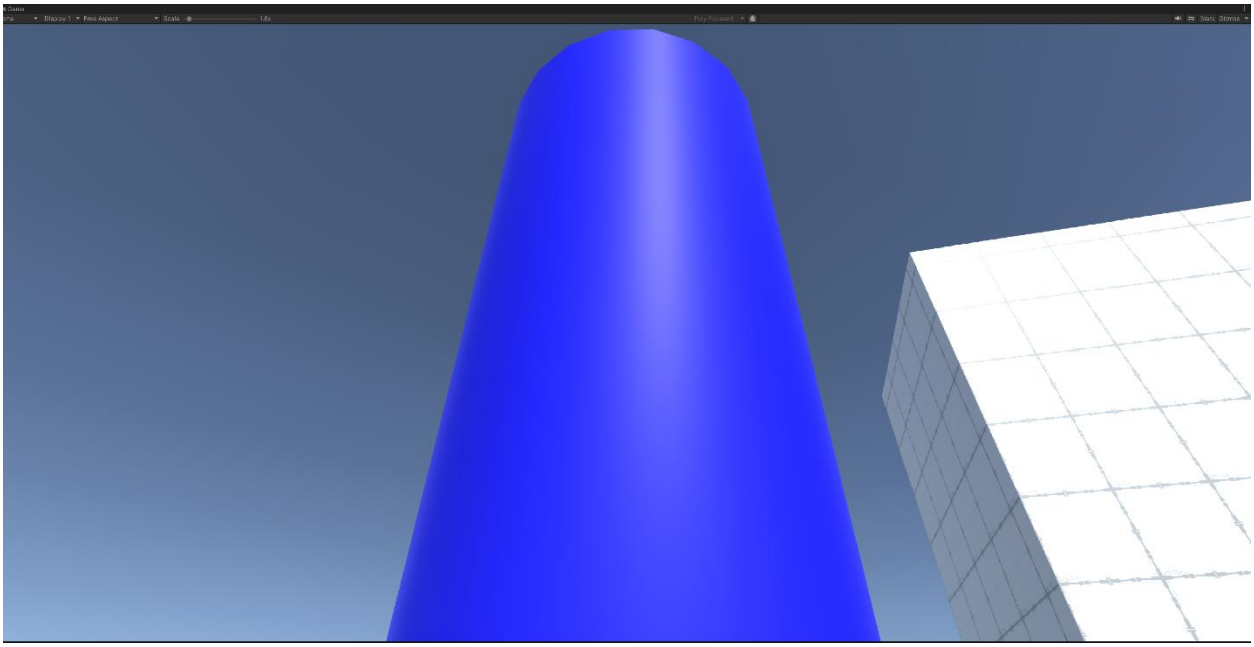

Figure 21

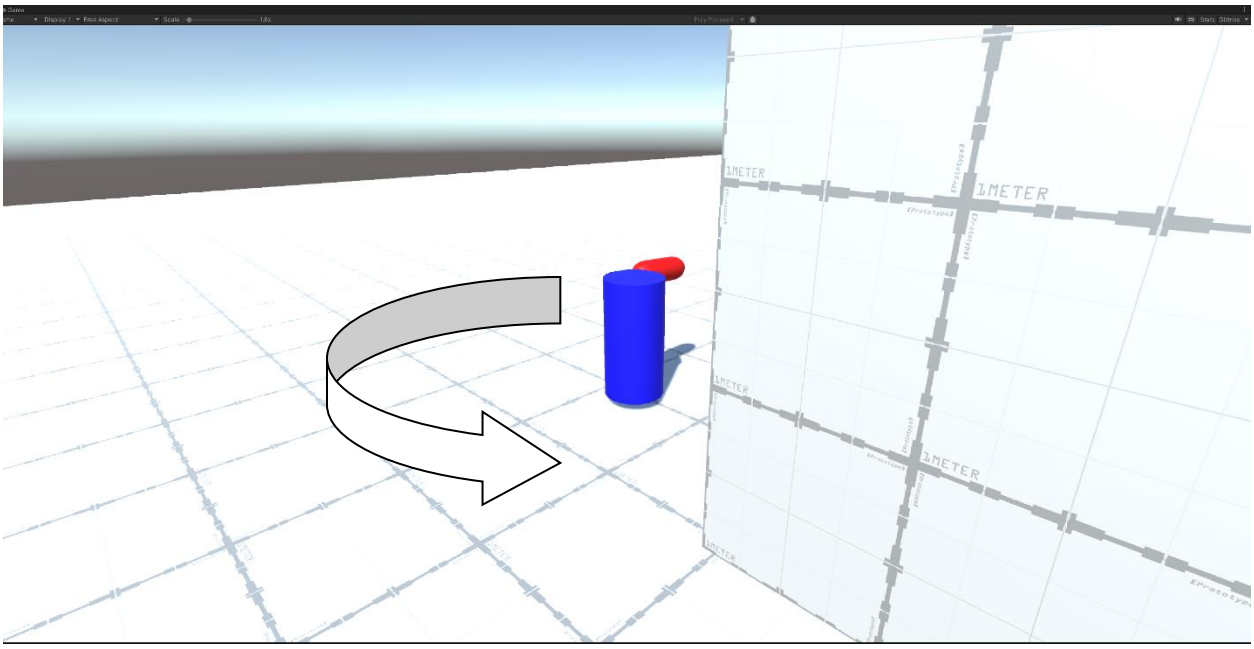

Figure 22

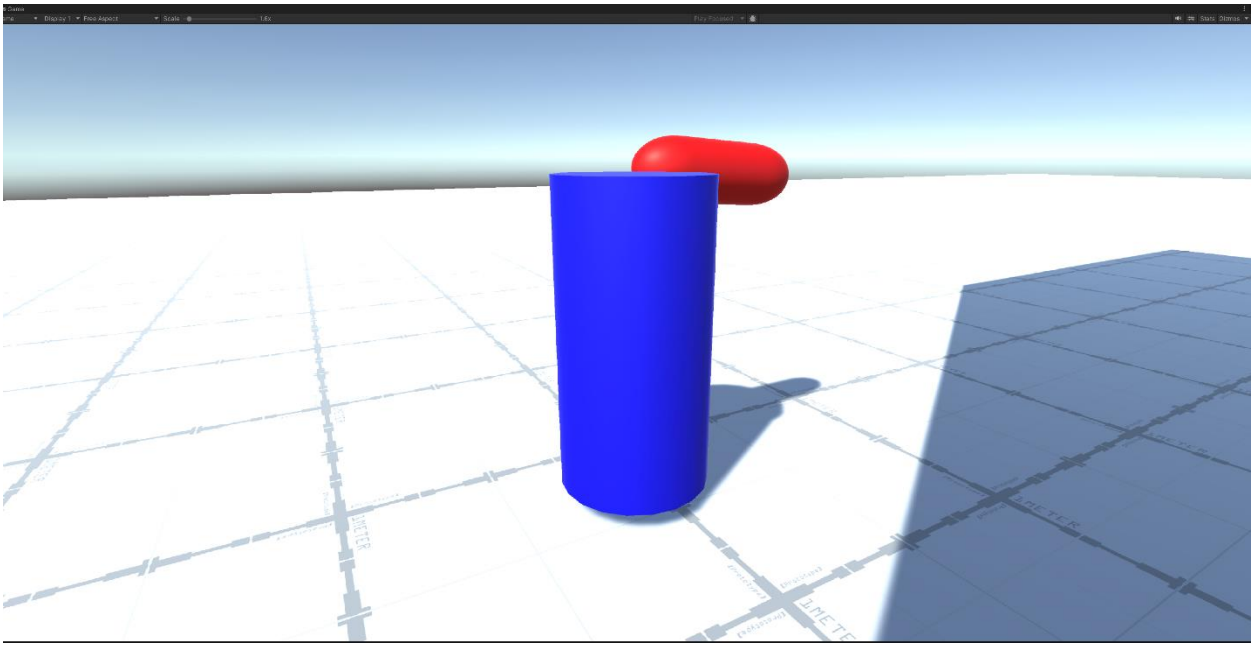

Figure 23

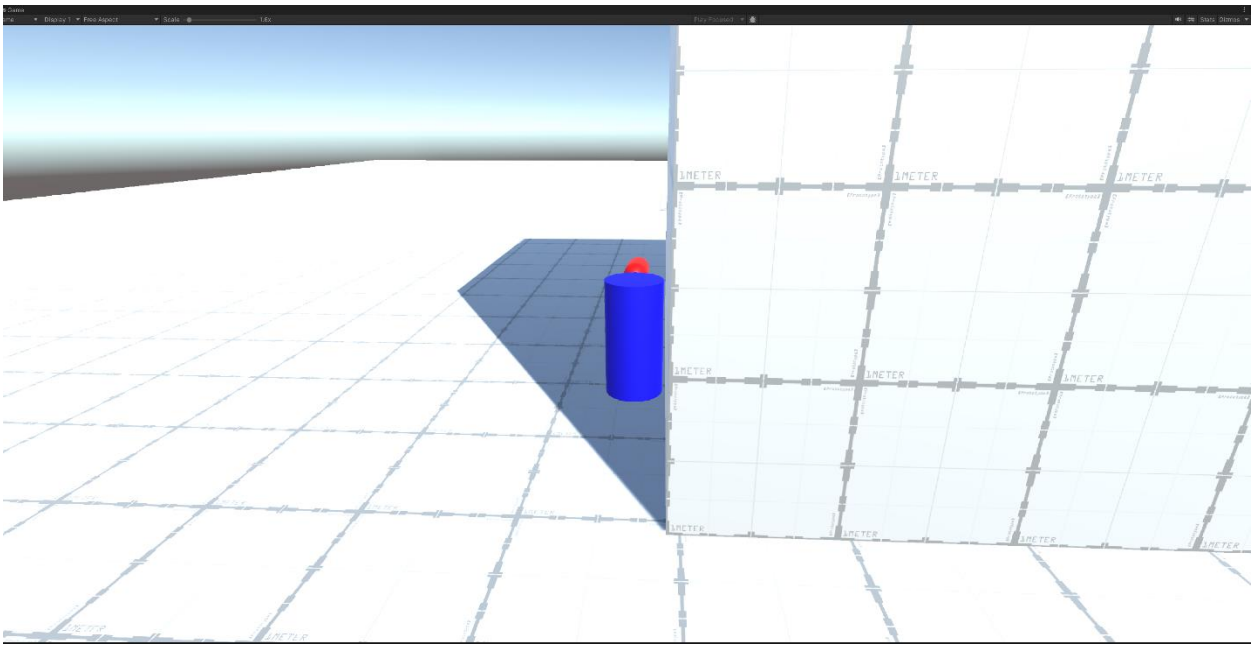

Figure 24

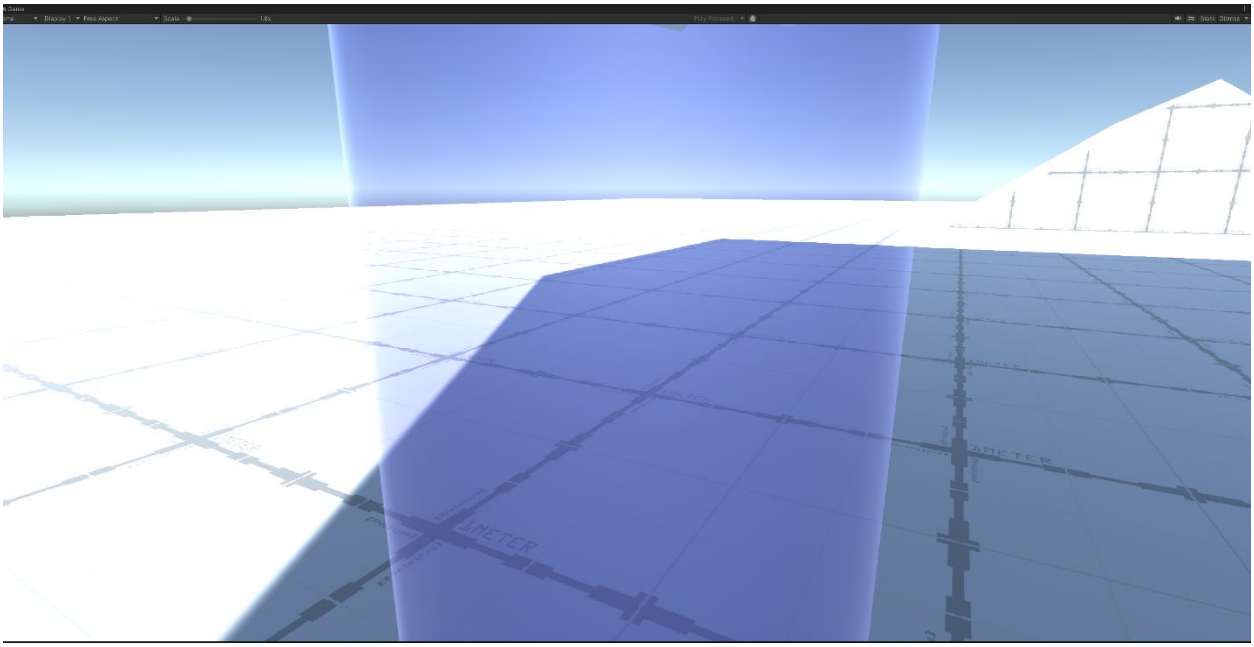

Figure 25

3.5 Findings

Although the scene in 3.4 is feasible in a simple environment, it is not suitable for all scenes.

For example, if the perspective of 3.4 is used in a game with intense combat, players can easily lose their sense of location due to changes in perspective. We can study how MONSTER HUNTER RISE handles perspective issues. In a mounted state (Figure 26), hunters typically don't fight. Currently, the game handles the camera in a way similar to 3.4. When an obstacle blocks the view, move the camera to the collision point (Figure 27) and ensure the camera faces the character. When the lens is too close to the hunter, the hunter is directly hidden (Figure 28). When the hunter is standing (Figure 29), the game assumes the hunter may be in combat, so the camera moves above and behind the hunter's head for a better view (Figure 30). When the camera and the hunter are blocked by a monster, the view does not change, but the hunter's

outline is directly displayed, ensuring that the player does not lose the hunter's position in the battle (Figure 31).

Figure 26

Figure 27

Figure 28

Figure 29

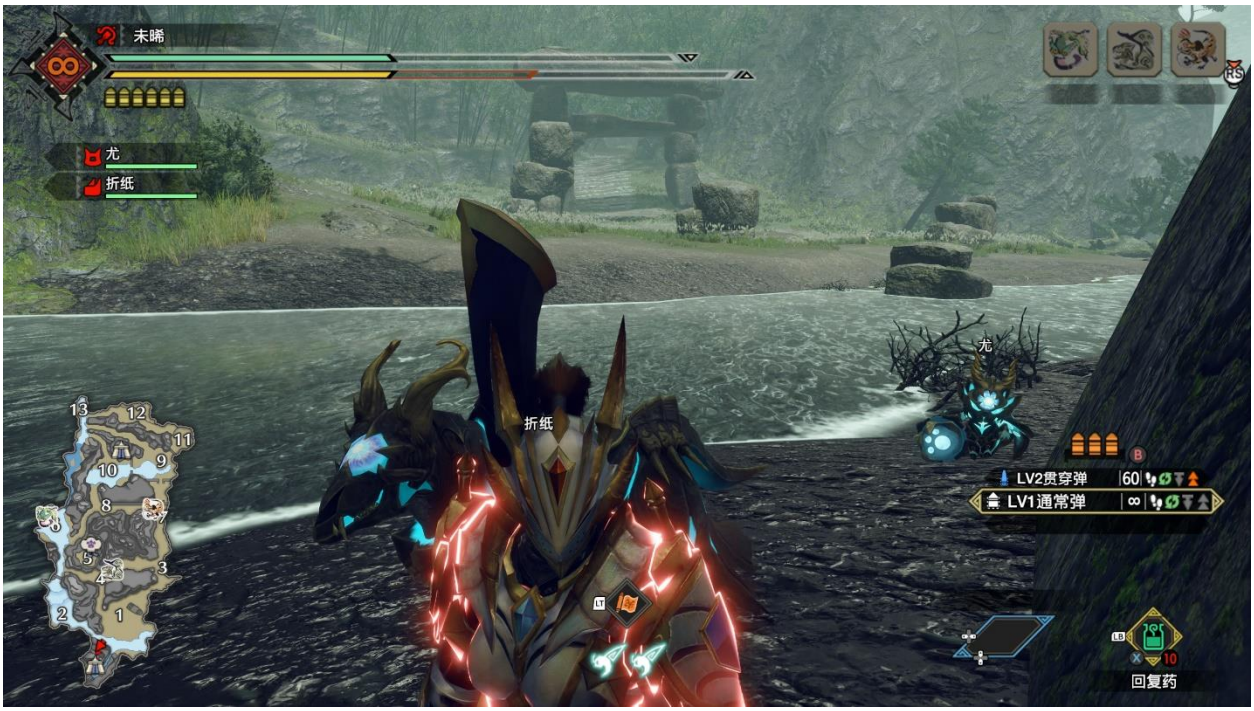

Figure 30

Figure 31

Another idea is to not change the camera position but to hide or make the walls transparent. This method is recorded in the patents of KONAMI (Figure 32, 33, (Japan Patent No. JP,09-299613,A, 1997)) and SEGA (Figure 34, (Japan Patent No. JP,09-050541,A, 1997)).

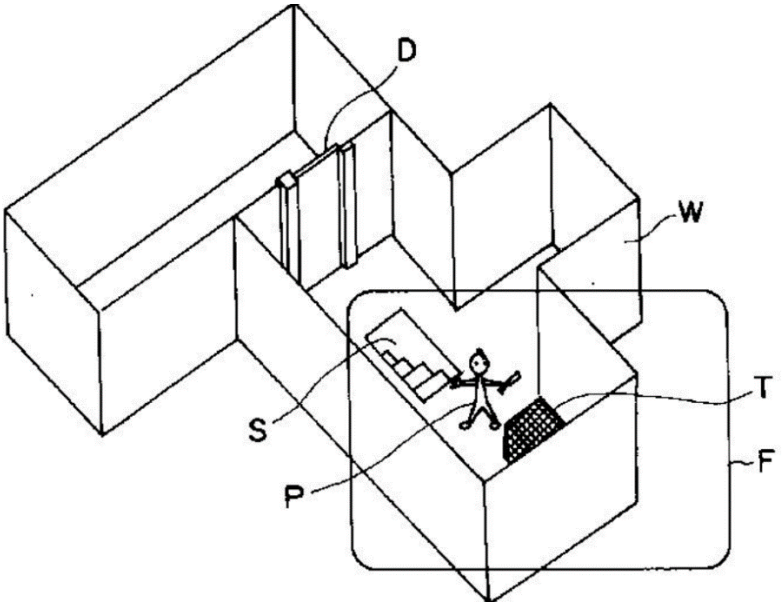

Figure 32

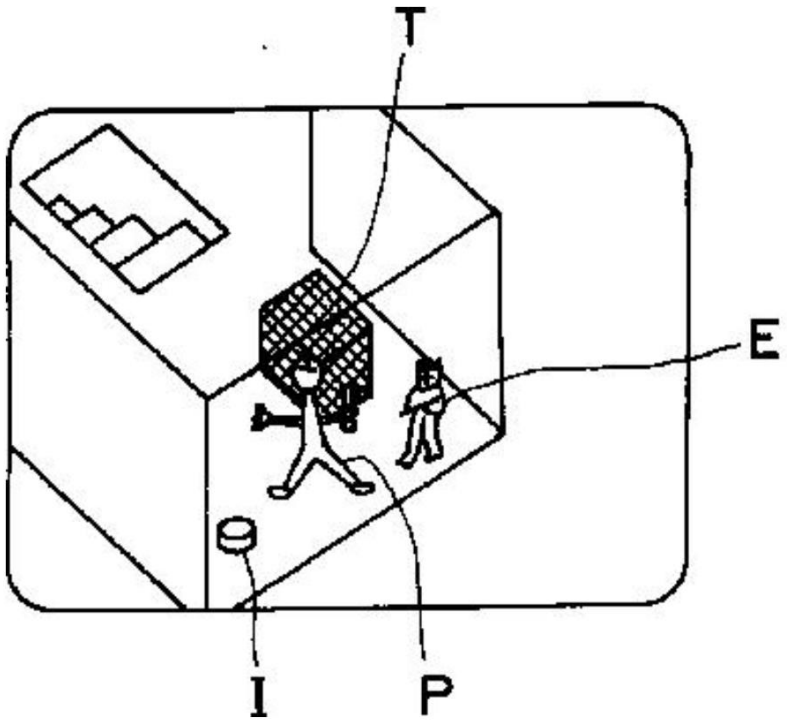

Figure 33

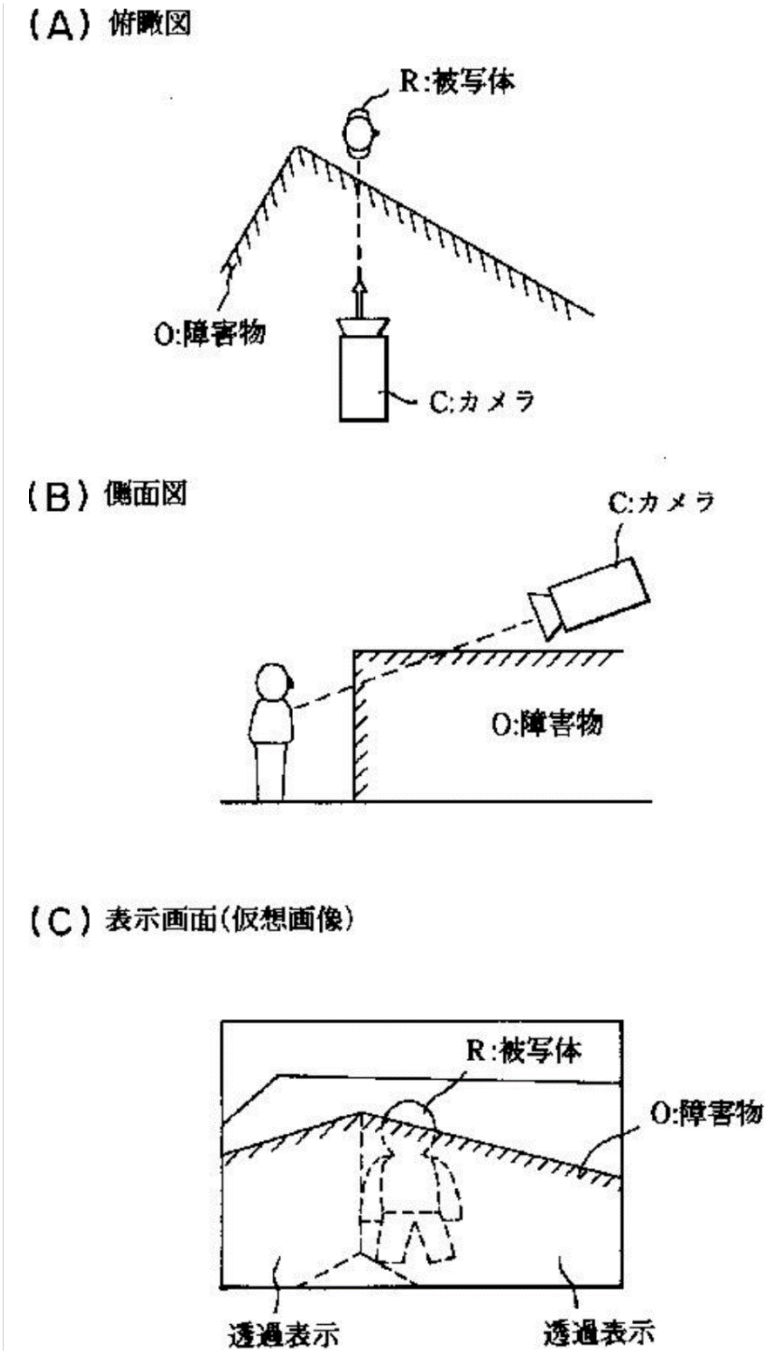

Figure 34

There are other considerations to take: For instance, in a scene filled with obstacles, should we really allow the camera to constantly move to avoid these obstructions? In situations where there are numerous enemies, should the camera automatically zoom out to give players a clearer view of each adversary, or even shift from a dynamic third-person perspective to a long-range third-person view? These questions are worthy of in-depth discussion.

Consequently, there isn't a one-size-fits-all solution for camera perspectives. What's more crucial is to flexibly adapt and apply various methods according to the specific needs of different scenes. The challenge and complexity in designing 3D game perspectives lie precisely here.

IIII. Conclusion

The realm of video game design is expansive, this report only deep into one of aspects: camera controls.

From the immersive first-person perspective that place players directly into the action, to the strategic advantages offered by third-person perspective, each offers a unique way to gameplay. We have seen how the perspective changes player’s engagement, strategy, and enjoyment.

From examples of industry giants like Nintendo, achieving a balance is not easy. Even with the latest technological advancements, developers continuously break obstacles.

But perhaps the most significant thing is the realization that there isn't a one-size-fits-all solution. The best camera perspective is that serves the game's purpose. The core principle remains: to create meaningful, and immersive experiences for players around the world.

References:

KENJI, Y. (1997). *Japan Patent No. JP,09-050541,A.*

Monteiro, D., Liang, H., Xu, W., Brucker, M., Nanjappan, V., & & Yue, Y. (2018). Evaluating enjoyment, presence, and emulator sickness in VR games based on first- and third- person viewing perspectives. *Computer Animation and Virtual Worlds*, pp. 29(3–4).

RIIDAA, U. A. (1997). *Japan Patent No. JP,09-299613,A.*

Turi, T. (2015, May 10). *The Essentials – Resident Evil*. Retrieved from Game Informer: https://www.gameinformer.com/b/features/archive/2015/05/10/the-essentials-resident-evil.aspx