Infinite Generated Space in Virtual Reality with Impossible Spaces and Procedural Generation, Group 8

Bernardo Pinto, Carlos Marques, Hélio Martins, Miguel Malheiro, *Student, IST*

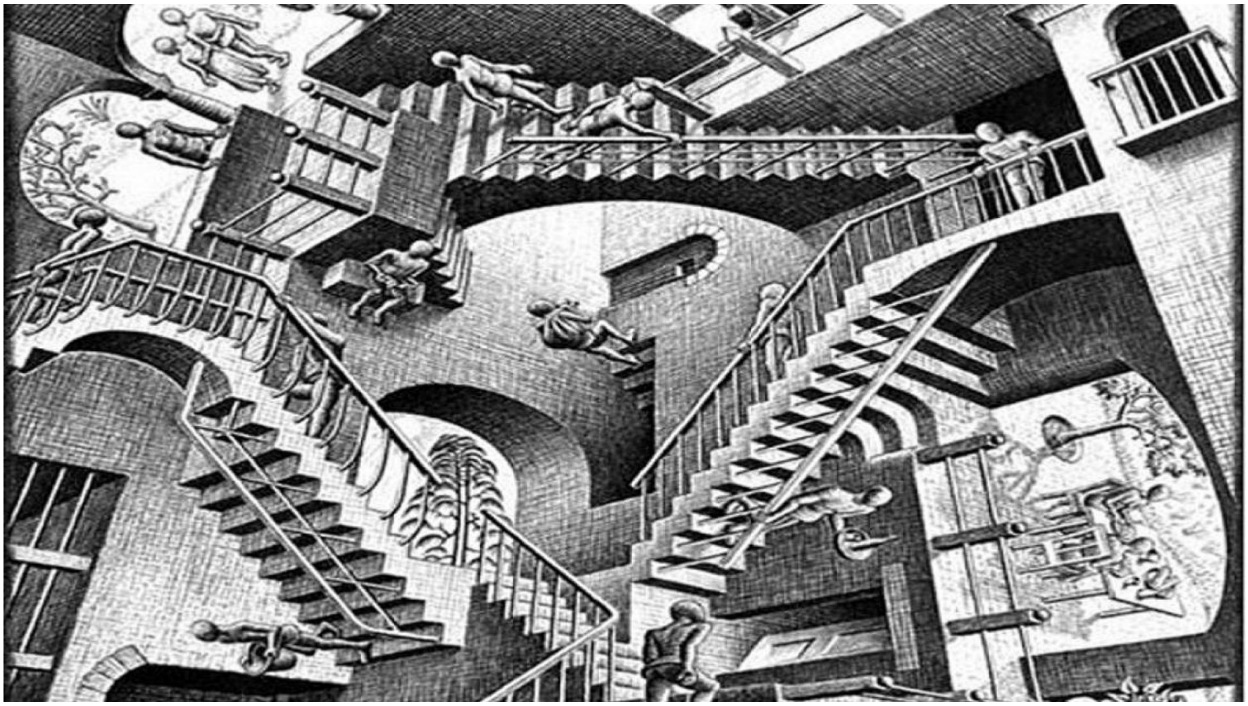

Fig. 1. Relativity by M. C. Escher.

Abstract—Most **Virtual Reality** applications use some type of physical controllers to control the movement of the person inside the virtual world, and for the few that do not use controllers they require a large room space, to prevent the user from walking into walls and losing immersion. Our main objective is to provide a way for people to walk around in expansive environments without needing a large play space of the same size. With our application, a person can use a 4x4 or more meters room as a play area, while being able to explore an endless array of rooms, without ever leaving that space.

Index Terms—Virtual Reality, Impossible Spaces, Immersion, Locomotion, Procedural Generation.

1 Introduction

Navigation on small spaces in Virtual Reality applications is mostly accomplished with the help of controllers, or/and redirection techniques, otherwise known as "redirected walking". Sadly, using controllers is not very immersive, and redirection techniques, while helpful, are tricky to use and have their limitations. We believe that, for a truly immersive experience, natural locomotion is required, and impossible spaces by themselves are not enough. Thus we introduce: Impossible spaces with redirected walking.

By using impossible spaces with redirected walking, we can in theory have an infinite space, transversed with natural locomotion by the user, without ever needing to stop or readjust.

This of course brings some issues: To accomplish the impossible space, some tricks are required, which may break the immersive experience. With this paper, we intend on seeing if impossible spaces can be used in conjunction with redirected for an immersive experience, and how much the user can be tricked into believing the impossible space - or if the user notices the illusion, but does not break immersion.

In addition to this, we will also show how we can combine this with procedural generation, for a truly infinite space, that can be adapted for a multitude of uses.

• *Bernardo Pinto, Carlos Marques, Hélio Martins and Miguel Malheiro are with Instituto Superior Técnico. E-mail: bernardo.r.pinto — miguel.v.malheiro — helio.martins — carlos.a.marques @tecnico.ulisboa.pt.*

2 Related Work

Space has been a big limitation on bringing VR to more people; It's hard to justify buying a piece of equipment when you cannot properly utilize it, due to space restrictions. Thus, a lot of investigation has gone into maximizing the available space and different types of controls for locomotion.

More notably, redirected walking techniques such as translation and rotation gains, VE(Virtual Environment) rotation, in all its variation. Furthermore, those techniques have been done with different tricks. VE rotation for example, has been done with tricks such as : Rotating the VE while the user is rotating its head [3], while blinking and while the user is distracted by intentional distractions [3] for example.

But, as previously mentioned, those techniques only really shine on large virtual environments. This space problem asked for another solution, and impossible spaces were a good candidate. Impossible spaces work by creating spaces that represent an architectural illusion. Imagine that your room had portals to other rooms - If the transition is done properly, one would hardly notice the illusion. Oddly enough, even if the illusion is not perfect, the user's perception of space remains consistent [2].

From [1]: "Impossible virtual environments contain geometry that violates the rules of Euclidean space, and therefore cannot exist physically in reality. While there are many potential ways in which virtual content may transgress physical laws, for the purposes of maximizing effective walking space, we focus on one specific type of Euclidean violation: Overlapping Spaces". For our application, we decided to go with 100% overlapping expanding rooms, to maximize space.

Furthermore, to increase credibility of the impossible spaces and thus enhance the illusion, immersive narrative can be used. For example, if you have to go through a forest to get to a temple, deep vegetation can be used to lead the path that the user must take [2]. This could also be used to justify certain redirection techniques, such as translation gain on an icy surface.

Lastly, the idea of using procedural generation with impossible spaces has been investigated before, with our main inspiration coming from [4].

Our work was made with the one mentioned above as inspiration, but there are some differences and some points we tackled that are not on that paper. Here, we intend on expanding this concept of procedurally generated impossible spaces, by including elements of narrative, translation gains, and the option to have interactable objects in any of the rooms. Also, our rooms make use of 100% of the available space, by having a 1 meter wide corridor around the room, which takes the size of the remaining space. Therefore allowing for easy adaptation and usage for multiple purposes: Games, Museums, etc...

3 System Design

In our first iteration, a single room was built with corridors connecting it to 8 others. All the rooms were built and placed manually in the work space. The idea was to test the concept without adding the automated generation.

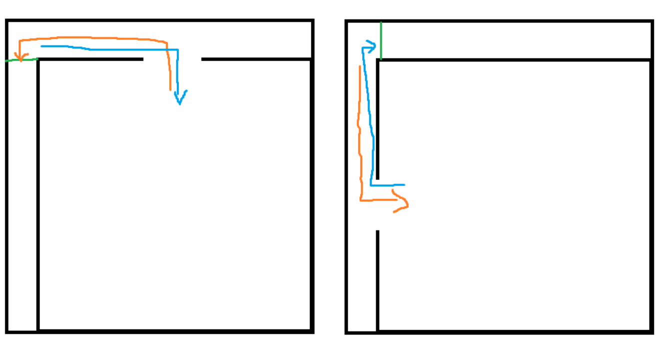

Fig. 2. Basic Portal placement to allow for simple transitions (the arrows simulate the path the user would travel in the virtual world while the green lines represent the portal positions)

Upon positive results, prefabricated rooms and corridors were created by hand and placed automatically, with the rooms having a maximum number of exits of 2. For the third iteration, room generation was automated, while still using prefabricated corridors; in this phase of development, we noticed that when moving from a corridor to a room some entrance ways would be replaced by solid walls and vice-versa.

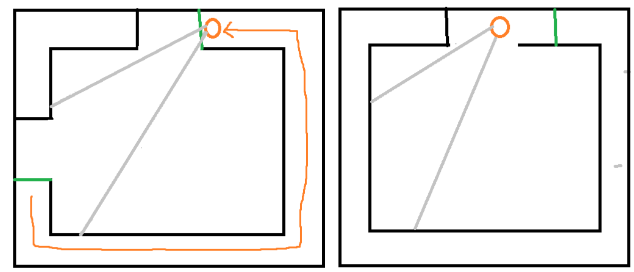

Fig. 3. Walls appear or disappear on teleportation between room

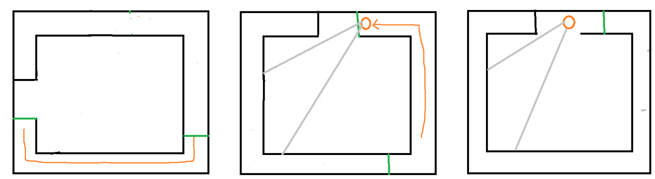

Fig. 4. Proposed solution to the wall problem

In the final iteration, the approach was changed so that both the rooms and the corridors were generated randomly (or taking into consideration an integer seed to pseudo-randomize the generation). The corridors in this last phase only constitute a corner and for larger corridors we chain multiple corners together.

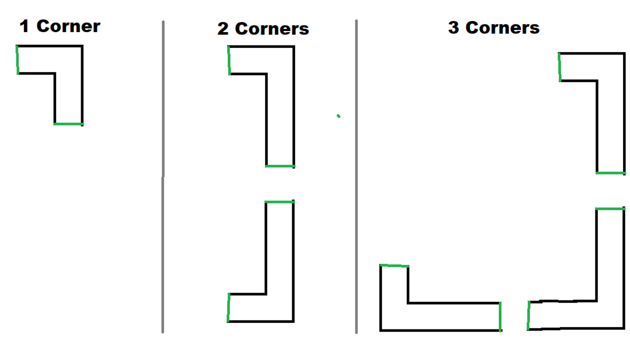

Fig. 5. Corridors composed of corners

Each corridor section partially copies the origin and destination room/corridor characteristics so as to prevent the bug mentioned in the third iteration.

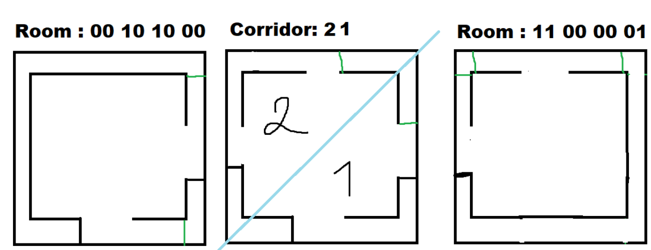

Fig. 6. Corridor copying the origin and destination rooms

For visibility purposes, we added textures to room floors where redirected walking was used to give the player visual clarification (ice slows the movement), and we started off by adding lights on the roof, but later changed to a lantern the player had on him at all times.

For future user testing, we implemented a simple task that requires the player to locate and carry a cube (spawned randomly) to a goal situated in a random room.

4 System Implementation

In terms of the actual logic behind the system, at first we used a cardinal based algorithm where each exit in a room was identified by its cardinal direction and its position relative to the door on a room, each room had 4 doors in it each of them could lead to up to two corridors left and right. An example Room name would be NR_SLR meaning that the room had a door in its North side and it had a corridor on its right it also had door in its South side with corridors both in its right and left, as seen in the image below.

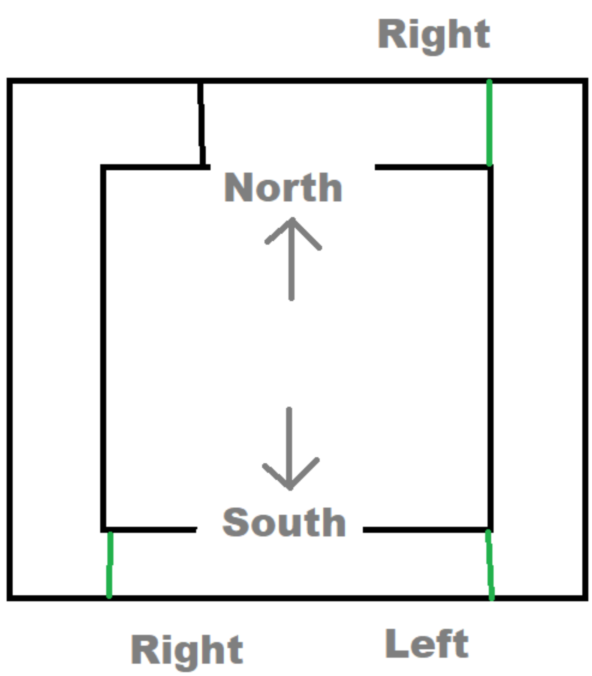

Fig. 7. Room NR_SLR as per the old nomenclature

In the updated logic, we used an algorithm based in an array of integers from 0-7 where each number represented a specific entrance way (0 was north left, 1 was north right, increasing clockwise around the room). This way, using the same example, the room name would be 01 00 11 00 (1s represent entrance ways, 0s are walled off).

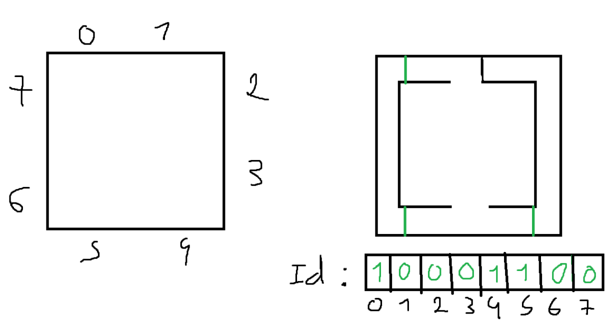

Fig. 8. New Room Nomenclature

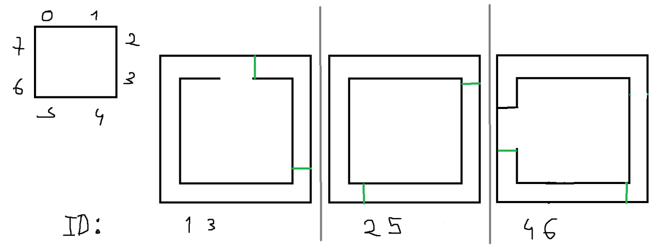

Fig. 9. New corridor nomenclature

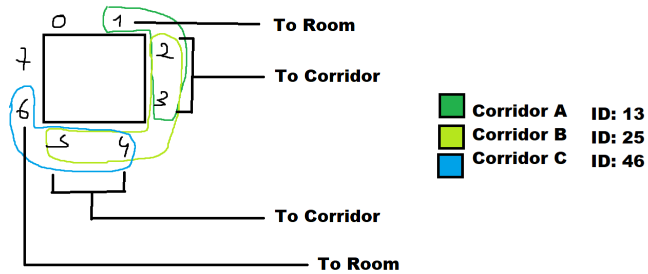

Fig. 10. Corridors composed by sections as by New Corridor Nomenclature

The final product is the Generator asset in unity:

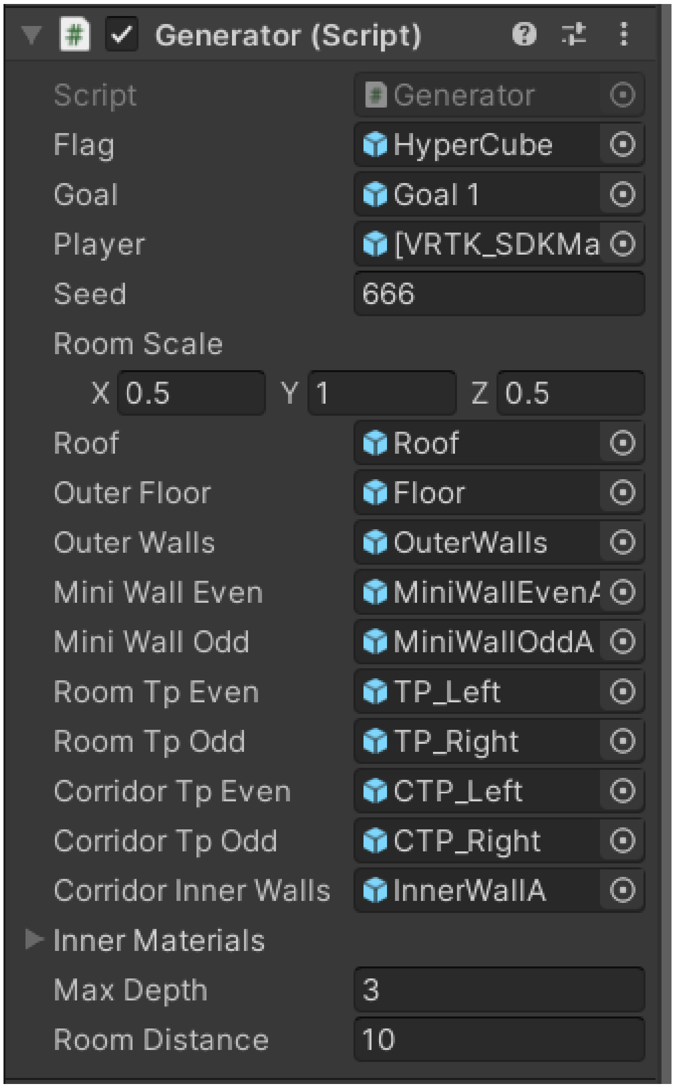

Fig. 11. Generator in unity inspector

Here you can customize the rooms generated, and change the generation seed. the size of the rooms, the distance between them, the user test object (flag) and the goal, the materials used for the corridors

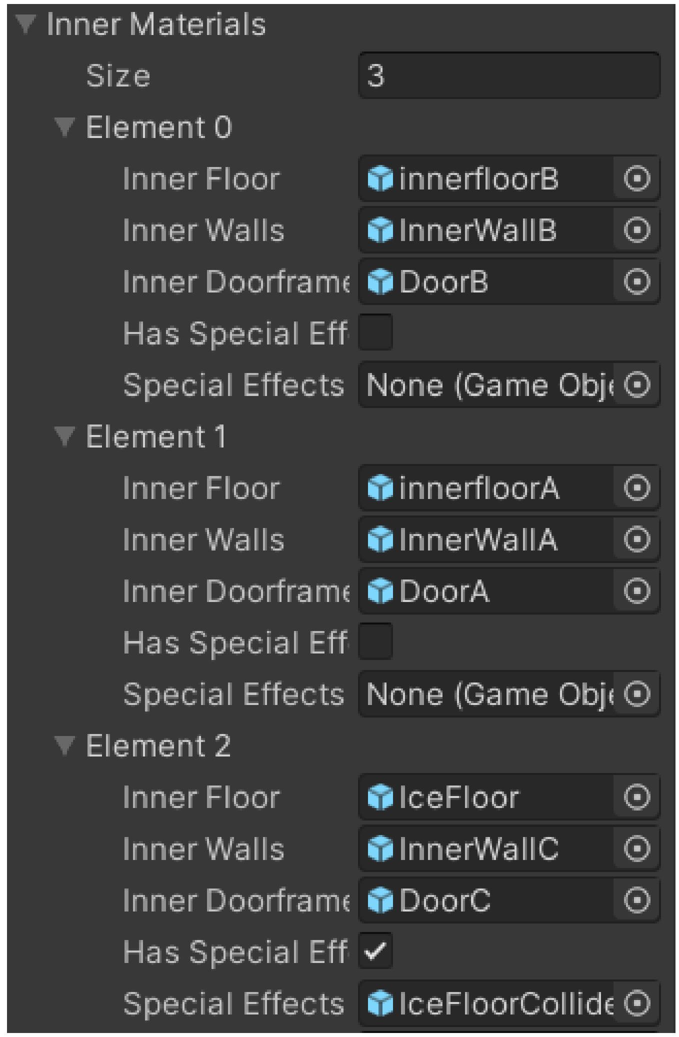

Fig. 12. Materials used to create the rooms (different aesthetics/ biomes and effects)

5 Limitations

Our current work, albeit taking full advantage of the available space, only supports rooms of one shape: Square rooms. This is because we based the generated rooms on a structure with an inner and outer part, that uses portals. For this, using squares made sense, because it allowed for easy transitions. Furthermore, the walls need to have 90º degrees between them, and the doors need to have doors in the middle of the walls. Regarding the procedural generation, using a depth higher than 3 takes quite a lot of time, and is therefore not advised.

6 Discussion and Future Work

We learned that even in a cramped environment, we can still have a large expansive map. Due to the scalable nature of our rooms, it was easy to adapt to various play areas. In addition, despite the overlapping nature of the rooms, the teleportation between them was achieved in a seamless manner. Even being aware of the impossible nature of the map, we found that our immersion was not broken and that we felt comfortable navigating through the map. One of the things that could be improved upon though, is navigation. Because every room has the same basic shape, even featuring doors in different parts of the room, it was hard for us to navigate, especially when they lead to branching paths. While they did make the world technically bigger, our enjoyment went down as figuring out where to go became a taxing task. Additionally, as mentioned above, every room is the same, as such the layouts can get repetitive. Adding more interactivity to the map as a whole and even individual rooms, not only would help with navigation, but would also keep the experience more engaging. Finally, one other thing that could be done is object-mapping. Given that the goal of this project is to bring huge worlds to small rooms, a lot of people may not be able to remove obstacles from the play area. Mapping these objects and adding them as obstacles to the virtual environment will allow for an even more effective use of the space. This was especially noticeable under remote working conditions, as it was hard for us to clear enough space to test from home.

7 Conclusion

In this paper, we saw some of the current solutions for navigation in virtual environments and how they affected user immersion and experience, discussed some of the methods used and their applications, and finally proposed our own solution, which integrates many of the discussed solutions, is extensible and aims to keep the user immersed, by allowing him to navigate seamlessly through the environment. Through this work, we show that user immersion can be kept through the combination of multiple techniques, and that by doing this, we can create a powerful tool that takes advantage of the limited space, and hopefully encourage developers to use solutions like ours, so that those who don't have large play areas for VR, do not get excluded.

Acknowledgments

The authors wish to thank Augusto Esteves, Joaquim Jorge, Chris Clarke and Rafael Kuffner. This work was made possible by Joaquim Jorge, by lending the necessary gear (Oculus Quest 2) and giving us classes alongside with Augusto Esteves. We would also like to thank Chris Clarke and Rafael Kuffner for their feedback in the early phase of our project.

References

[1] Evan, Zachary, Samantha, David, and Mark. Impossible spaces: Maximizing natural walking in virtual environments with self-overlapping architecture. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):555–564, Apr. 2012.

[2] J. A. Fisher, A. Garg, K. P. Singh, and W. Wang. Designing intentional impossible spaces in virtual reality narratives: A case study. In *2017 IEEE Virtual Reality (VR)*, pp. 379–380, 2017. doi: 10.1109/VR.2017.7892335

[3] T. C. Peck, H. Fuchs, and M. C. Whitton. Evaluation of reorientation techniques and distractors for walking in large virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 15(3):383–394, 2009. doi: 10.1109/TVCG.2008.191

[4] K. Vasylevska, H. Kaufmann, M. Bolas, and E. A. Suma. Flexible spaces: Dynamic layout generation for infinite walking in virtual environments. In *2013 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 39–42, 2013. doi: 10.1109/3DUI.2013.6550194