UNIVERSITY OF CALIFORNIA SANTA BARBARA

PROJECT REPORT

Retargeting Virtual Worlds

Author:
Shwetha RAM

Committee:
Dr. Matthew Turk
Dr. Tobias Hollerer
Dr. Pradeep Sen

Submitted in partial fulfillment of the requirements for the degree of Master of Science

in

Electrical and Computer Engineering

March 23, 2017

Abstract

How do we present a VR experience to a user in the comfort of the living room?

The proliferation of several HMDs and the VR experiences they bring with them promise to take us a step forward from being surrounded by a virtual world to presence, the feeling of actually being there. But while the user is 'there', he or she is actually here in the physical environment, navigating a way through obstacles. Instead of asking the user to first find a large enough space devoid of obstacles, a scheme to 'retarget' a virtual world to a given physical environment is presented. Semantic relationships between objects are extracted from the virtual world. The retargeting aims to position the virtual world objects such that these semantic relationships are preserved and the match with the geometry of the physical environment is maximized. This retargeting scheme was successfully applied to retarget different virtual worlds to a real world physical environment.

Contents

List of Figures

Chapter 1

Introduction

It is well known that the sense of realism in a virtual world increases if the user is able to walk around in that world and explore it naturally. It was shown by Usoh et al., 1999 that compared to other forms of navigating a virtual world like push-button-fly or walking-in-place, real walking offers a greater sense of presence to the user. In their user study, the users were asked questions about performing a locomotion task in push-button-fly, walking-in-place and real walking scenarios. Given below is a result of their user study from which it becomes clear that real walking highly correlates with presence in a virtual world.

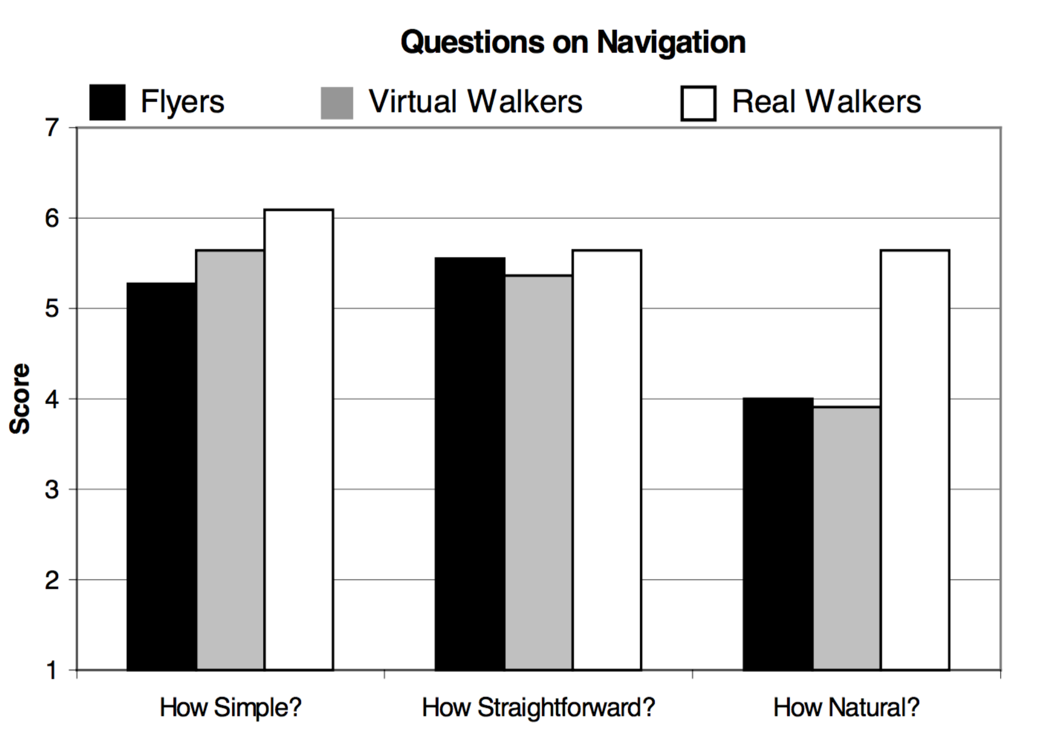

FIGURE 1.1: Real Walking increases Presence

Further, real walking also reduces discomfort experienced by a VR user as the number of conflicting sensory cues are reduced. Thus, in order to increase a sense of presence and reduce discomfort to the user, real walking is preferable to other methods of navigating a virtual world.

Today, we have the tracking capabilities and head-mounted display technologies to make real walking in virtual worlds possible. Ideally, we would like the user to forget about the physical environment/real world while experiencing the virtual world. But this is not possible due to the mismatch between the virtual world and the real world.

Users playing a game that involves shooting and dodging bullets would need to keep track of the gunmen in the virtual world and also be careful not to collide with the furniture in the real world. Users experiencing a calm beach scene would need to think before stretching their legs, lest they hit the opposite wall. These users are compelled to consider the layout of the physical environment while experiencing the virtual world, to ensure their safety. Both the physical environment and the virtual world demand the users' attention. This war of the worlds lowers the sense of presence in the virtual world, making the experience less compelling.

One way to solve these problems is to ask the user to find a physical environment of suitable layout and clear away all the obstacles. This would be cumbersome to the user who wants to experience VR in the comfort of the living room. Another way is to generate virtual worlds suited to the physical environment. While doing this, it is important to consider the semantics of the virtual world in order to keep it interesting and convey the designer's intent.

The fact that the user is immersed in the virtual world while navigating the physical environment poses many design and safety concerns. On one hand, we want to present the virtual world such that the intent of the designer is conveyed as completely as possible. On the other hand, we want to maximize the match between the virtual world and the physical environment which it overlays. The situation calls for an optimal approach that fulfils both the above mentioned goals.

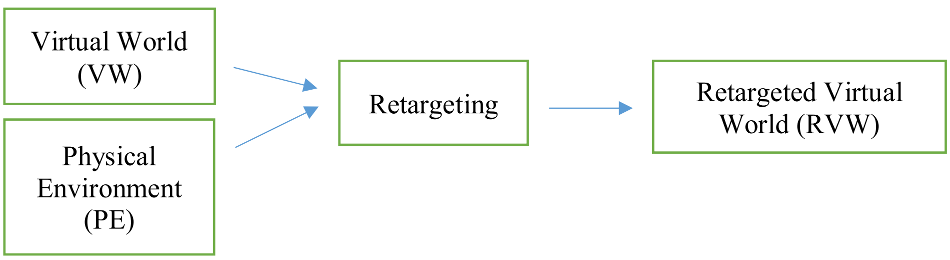

FIGURE 1.2: Retargeting Virtual Worlds - Block Diagram

Here, a scheme to 'retarget' a virtual world to a different physical environment is proposed. The retargeted virtual world aims to deliver an experience of the virtual world in the physical environment of the user. First, semantic relationships between the objects are extracted from the virtual world. While retargeting to a physical environment, the objects are placed such that the semantic relationships between objects are preserved and the match with the physical environment geometry is maximized.

The retargeting problem can be said to be composed of two important subproblems. The first is to formulate a fitness function to quantify how good a particular retargeting of the virtual world is for a given physical environment. This should take some stated retargeting goals into account. The second is the optimization which aims to achieve the best possible fitness and in turn, the optimal retargeting for the stated goals.

We define the following three goals that an ideal retargeting should accomplish. An ideal retargeting should,

1. Maximize geometric match with Physical Environment
2. Avoid Collision between Virtual World Objects
3. Preserve Semantics of the Virtual World after Retargeting

The primary goal in forcing a geometric match is to ensure that all regions in the physical environment that are occupied by obstacles should be represented by obstacles in the virtual world that overlays it. This is to ensure that the user does not collide with an obstacle in the physical environment, that appears as free space in the virtual world. In addition, we would like virtual world objects to overlay the obstacles in the physical environment that are most similar to them. It is important to note that only geometric factors (size, shape, etc.) are considered in the matching and semantic factors (like matching a chair to a chair) are not considered in this work. This is because, for the most compelling experiences, virtual world objects are very different from the obstacles in a typical living room.

The second goal stated is to avoid collision between virtual world objects. Although collisions are not desirable in any situation, they are more tolerable for some virtual world objects compared to others. For example, collision between rocks is acceptable but collision between furniture is not.

While striving for a geometric match and avoiding collisions is straightforward, the idea of semantics is ambiguous as there are multiple ways to define this. For this work, the semantics of a virtual world are defined to be comprised of the scene environment and the relative distance order and relative orientations of the virtual world objects.

The fitness function formulated captures how good a retargeting is, in terms of the goals for ideal retargeting stated above. A genetic algorithm was used to find the best possible fitness function value which corresponds to the optimal retargeting.

In general, it is hard to design an evaluation metric for the retargeting problem. However, the values attained by various terms of our fitness function give an indication of how good the retargeting is on those fronts.The more the geometry of a physical environment is like the virtual world, the more the virtual world retargeted to that environment resembles the original. But even for physical environments very different from the virtual world, the retargeted virtual world tries to convey an experience of the original.

Chapter 2

Related Work

In order for real walking in VR to be possible, the disconnect between the user's physical environment and the virtual world has been tackled in either of the two following ways. One way is to alter the physical environment to better suit the virtual world. Another way is to generate virtual worlds to suit a physical environment.

Most VR experiences that can be experienced through HMDs today [Eg.: "HTC VIVE"] require the user to clear away all the obstacles in the physical environment. The extreme in this direction would be location based VR, where the physical world is carefully designed to mirror the virtual world as in 'Real Virtuality' [Charbonnier and Trouche, 2015] and 'The Void' ["The Void, LLC US"]. While these experiences are compelling in their own right, they do not fit the use case of experiencing VR in one's living room with ease. The requirement to alter the physical environment limits the reach of VR and it is desirable that a user should be able to experience a virtual world in a physical environment that was not predesigned or even altered for this very purpose.

The idea of procedurally generating virtual worlds based on a set of rules has been around for a while now. These rules could be either specified by a designer or derived from exemplary worlds. In Yu et al., 2011, the rules for furniture placement are extracted from exemplary indoor scenes. In Merrell et al., 2011, a set of design guidelines for furniture layout are specified. These rules are used to automatically generate realistic furniture arrangements. As the application is the placement of furniture in an assumedly empty room, a match with the layout of the room is enforced while the issue of a match with real world obstacles does not arise.

In Gal et al., 2014, the rules for placement of AR application components include scene-consistency (relation to the environment) together with self-consistency (relation between the components). However, the scene-consistency rules are defined only with respect to planar surface features detected from the environment. Also, safety is not a concern here as the user is not likely to collide with real world obstacles that are visible to him through the AR interface.

In Sra et al., 2016, the idea of a semantic match between a real world obstacle and the virtual world object that overlays it is presented for one class of objects, namely, chairs. The real world obstacles that do not fall into this category are fenced out of the walkable area. Rules specified by the designer are enforced in the placement of virtual world objects outside these walkable areas.

In Shapira and Freedman, 2016, the virtual world objects are placed in order to maximize the geometric match with the real world objects upon which they are overlaid. Also, all the real world obstacles are included in the layout of the virtual world. However, no semantic relationships between the virtual world objects are considered and in the results presented, good geometric matches also give rise to semantically good looking scenes.

Most of the papers cited above require that the virtual world be described as a bag of objects and some optimality constraints between them. The designer is often tasked with formulating the constraints, which are solved to generate the optimal virtual world.

The goal of Sun, Wei, and Kaufman, 2016 is most similar to ours of all the cited works in that they attempt to map a virtual world (already designed by a designer) to the physical environment of the user. They do this by folding the virtual scene onto the physical environment and derive an altered rendering based on this mapping to guide user navigation. It is also notable that their mapping is only locally injective and the user can return to the same point in the physical environment multiple times and see a different view everytime. This solution is very different from ours.

Our retargeting scheme involves optimum placement of objects from the virtual world in order to maximize geometric matches with the physical environment while preserving semantic relationships in the virtual world.

The semantic relationships between the virtual world objects are derived from their positions and orientations in the virtual world before retargeting. After designing the environment of the virtual world which is preserved in the retargeted worlds, the designer is only required to specify the priority of the virtual world objects, the maximum number of times they can occur in a retargeted virtual world and restrictions on their scale and orientation. The geometric match between the real world obstacles and the virtual world objects is maximized, but semantics (like matching a chair to a chair) are not considered. This is because, for the most compelling experiences, virtual world objects are very different from the obstacles in a typical living room. By matching for the geometry, it is ensured that a real world obstacle corresponds to a virtual world object closest to it. And the user is protected from collisions with real world obstacles.

Chapter 3

Retargeting Scheme

This chapter gives the details of the retargeting algorithm.

3.1 Representation

A voxel representation of occupancy is used for the physical environment and the objects in the virtual world. In our implementation, a voxel is a cube of length 10cm and takes on binary values to indicate either occupancy (1) or unoccupancy (0). These voxel representations of the objects are compared with the voxel representation of the physical environment in order to determine how well a particular retargeting of the virtual world matches the physical environment. As stated previously, only a geometric match is aimed for. If semantic matching is desired in an application, more complex features will have to be employed at this stage.

One of the inputs to the retargeting system is a 3D reconstruction of the physical environment to which the virtual world is to be retargeted. The Matterport system ["Matterport,Inc"] was used for this purpose. The capture is done using the Matterport Pro 3D camera and an iPad App. The captured images are then processed in the Matterport cloud to get a mesh model of the physical environment. The input to this stage could be from any other 3D capture device like Kinect, Google Tango, etc.

This mesh is then processed using standard graphics techniques to get a voxel representation of the occupancy for the physical environment. Any holes in the reconstruction due to blind spots at the capture stage are also treated as obstacles. From this, the floor plan and room boundaries are derived to aid in the generation of the retargeted world. Only physical environments where the floor is a single plane (e.g.: no staircases) have been considered for this work.

The meshes of the objects in the virtual world are processed using the same techniques to get their voxel representations. They also have properties which are to be specified by the designer. These properties influence how the objects are represented in retargeted worlds. The properties include limits on how the object can be scaled and rotated. Each object is also associated with a priority. Objects with priority greater than zero are priority objects and those with priority less than zero are filler objects. Priority objects occur only once in the scene (if the same object occurs twice in the virtual world, they are counted as two different priority objects) but it is possible that some lower priority objects do not occur in some retargeted worlds. Also, it is attempted to preserve the semantic relationships between priority objects. Filler objects, on the other hand, can occur any number of times or be skipped altogether. They do not contribute to the semantic score.

For example, in the island scene, the trees and rocks are filler objects while the statues are priority objects.

3.2 Fitness Computation

The fitness function tries to quantify the success of a retargeting. Our goal in retargeting is to preserve the feel of the virtual world as much as possible. In other words, if two people experience a virtual world retargeted to two different physical environments, we would like their descriptions to match. At the same time, it is desired that every obstacle in the physical environment corresponds to an obstacle in the virtual world. This is to avoid the situation where the user collides with a real world obstacle because it appears as free space in the virtual world. Further, from the set of objects in the virtual world, we would like the real world obstacles to be replaced by objects that closely match them, whenever possible. Finally, we want to avoid collisions between the virtual world objects.

The three requirements stated above are quantified by the three scores, namely, Semantic Score, Mismatch Score and Collision Score. The retargeting is posed as an optimization problem where the value to be minimized is the weighted average of these three scores.

3.2.1 Mismatch Score

Mismatch Score quantifies the mismatch between the placement of the virtual world objects and the geometry of the physical environment.

$$MMS = m_1 \frac{\sum_{i \in PE}(PE(i)).(1 - RVW(i))}{\sum_{i \in PE} PE(i)} + m_2 \frac{\sum_{i \in PE}(1 - PE(i)).\,(RVW(i))}{\sum_{i \in PE}(1 - PE(i))} \tag{3.1}$$

Each voxel in the physical environment (PE) is compared with its overlapping voxel in the retargeted virtual world (RVW) by the inner product operator. PE(i) and RVW(i) refer to the ith voxels of the physical environment and the retargeted virtual world respectively. 1 stands for a unit vector of the same dimensionality as PE and RVW.

The first term quantifies the fraction of occupied voxels in the PE that are unoccupied in RVW. The second term quantifies the fraction of unoccupied voxels in the PE that are occupied in the RVW. The two terms quantify two different kinds of mismatches.

The values of the weights m_1 and m_2 are decided based on the desirability of each situation. If an obstacle in the PE appears as free space in the RVW, the user is at risk of colliding with the obstacle. If free space in PE is occupied by an obstacle in RVW, the situation, though not desirable, does not pose a danger to the user. Alternately, it is desirable for free space in RVW to coincide with free space in PE, but it is a lot more important for the obstacles in RVW to coincide with obstacles in PE. Thus we require m1>m2, accounting for the consequences of each kind of mismatch. It is important to note here that the second term, though not weighted very heavily, helps to preserve walkable spaces and prevent entire regions in the virtual world from becoming inaccessible.

The mismatch score should ideally be very low to indicate a very good match with the scene geometry.

In the virtual worlds we worked with, all the objects are constrained to be placed on the floor. The physical environments we worked with had high enough ceilings and all the obstacles the user needed to avoid were on the floor. For example, there are no chandeliers hanging so low that the user has to avoid them. As a result, it was sufficient to enforce geometric match for a height of 2 meters from the ground, taking even the tall users into account.

3.2.2 Collision Score

Collision Score quantifies the collision between the objects in the retargeted virtual world. It is calculated for each pair of objects. The collission score of a retargeted virtual world is defined as

$$CS = \frac{1}{N(N-1)} \sum_{m,n<N,m\neq n} \left(\frac{\sum_{i\in O_m,O_n} O_m(i).O_n(i)}{\sum_{i\in O_m} O_m(i)} + \frac{\sum_{i\in O_m,O_n} O_m(i).O_n(i)}{\sum_{i\in O_n} O_n(i)} \right) \tag{3.2}$$

It is the mean value of the fraction of pixels of each object that are colliding with the other object, normalized for all object pairs in the scene. The overlapping voxels are compared by the inner product operator. In the above expression, N is the total number of virtual world objects in the scene, m and n are the indices of individual objects and i is the voxel index.

Ideally, the collision score is zero to indicate no collision between objects.

3.2.3 Semantic Score

Semantic Score (SS) quantifies how well the semantics of the virtual world are preserved during retargeting. There can be a number of different ways of defining semantics of a virtual world. For this work, it was a design choice to define the semantics to be comprised of the virtual world environment and the relationships between the virtual world objects.

The environment of the virtual world comprises of the skybox, floor textures, wall textures, boundary elements, etc. These are preserved in every retargeting of the virtual world. Thus, the environment elements do not contribute to the semantic score and it is determined solely by the relationships between virtual world objects.

There could be different ways to define scores that capture the relationships between the virtual world objects. For this work, we have chosen to have one score of distances between the objects and another of their relative orientations.

An obvious choice for the distance score is the relative distances between virtual world objects. However, it is possible for a virtual world to be retargeted to a physical environment of very different layout and in such cases, it is not feasible to preserve the relative distances between objects. Attempting to do so would unnecessarily over-constrain the problem. Instead, we try to ensure that the objects which were farther from each other are not closer than other objects which were closer to each other after retargeting. In other words, we preserve the distance order.

For each object not in the right distance order, there is a penalty proportional to how far away it is from its right position. The distance rank score for the retargeted virtual world is the normalized sum of distance rank score of each object. Ideally, the distance rank should be exactly preserved and the score should be 0.

It was decided that preserving the distance order between virtual world objects is a sufficient score of distance to preserve the experience of the virtual world. By choosing to preserve relative distance order rather than relative distances, we are, in effect, ignoring some information about the virtual world. This becomes very obvious when the physical environment being retargeted to is not challenging. For example, consider the trivial case of retargeting a virtual world to a physical environment whose layout is same as that of the virtual world. Suppose further that there are no obstacles in it. In this case, it is easily possible to preserve both the relative distances and relative orientations exactly by replicating the arrangement in the virtual world. Our scheme, however, is not likely to converge to this solution as preserving relative distances was never a goal of the retargeting scheme. On the other hand, it works well in the typical case where we are faced with a physical environment of a different layout with some obstacles in it. In this typical case, if we forced a preservation of relative distances, we would have to compromise by settling for a poor match with the scene geometry or by allowing collisions.

The relative orientation score quantifies how well the relative orientation between two virtual world objects are preserved after retargeting. The relative orientation of one object with respect to another is the angle made by the x axis in its local coordinate system with the line joining the centers of the two objects. It is modelled as a Gaussian with its mean equal to the relative orientation in the virtual world and the standard deviation proportional to the distance between the two objects in the virtual world. This follows from the hypothesis that it is more important to preserve the relative orientations of objects which are closer to each other. The relative orientation score for the retargeted virtual world is the normalized sum of relative orientation score of each object. Ideally, the relative orientations should be exactly preserved and the score should equal one.

The semantic score is a sum of the two scores, namely, distance rank score and the relative orientation score. The semantic score is weighted by the priority of the objects. This ensures that higher priority objects are preferred over the lower priority ones and also that their semantic relationships are better preserved since their contribution to the semantic score is higher.

The distance rank score is to be minimized while the relative orientation score is to be maximized. The semantic score of a retargeted virtual world is defined as a weighted sum of the distance rank score and the relative orientation score. The weight for the relative orientation score is negative to ensure that it is maximized.

Finally, the fitness of a retargeted virtual world is a weighted average of the mismatch, collision and semantic scores. This is to be minimized for optimum retargeting.

$$Fitness = w_1.MMS + w_2.CS + w_3.SS \tag{3.3}$$

The values of w_1, w_2 and w_3 are determined based on the relative importance of each kind of score for the application.

3.3 Optimization

While the number of priority objects is known beforehand, the number of filler objects is a variable in the optimization. This problem is handled by specifying the maximum times each object can occur in the retargeted world and having binary selection terms that determine which of these occur in a particular retargeted virtual world. Thus, if we have 3 priority objects and 2 filler objects each of which can occur a maximum of 10 times, we have a total of 23 (3 + 2x10) possible objects in the retargeted virtual world. Each of these objects is associated with a selection term that determines whether or not it is present.

The goal of the optimization is to determine which of the objects are present, their positions, orientations and scales. In the virtual worlds considered, the objects were constrained to be on the floor and rotate only about the z axis. Under these conditions, we have two position terms, one orientation term, one scale term and one selection term, a total of five terms for each object.

This is a very high dimensional problem with a non convex solution space. Also, the voxel representation means that the two position terms are constrained to be integers. The high dimensionality, non convexity and mixed integer constraints make this a challenging optimization problem.

A genetic algorithm with elitism was employed to arrive at the optimal retargeted virtual world for a given physical environment. Each individual of the population is a retargeted virtual world. This choice made it easy to formulate the fitness function and object instantiation constraints. Also, the stochastic nature of the algorithm, which returns a different virtual world every time the algorithm is run in the same physical environment, adds to the experience.

The genetic algorithm used a population of 200 individuals with an elite count of 10. The fitness function is a weighted average of the mismatch, semantic and collision scores. For the initial population, the objects are randomly placed in some occupied areas of the physical environment with random values for orientation, scale and selection terms. The algorithm runs until the average relative change in the fitness function value over 50 stall generations is less than function tolerance. The best individual from the last generation gives us the configuration of objects in the optimally retargeted virtual world.

3.4 Retargeted Virtual World Generation

Floor textures, boundary elements and skybox comprise the environment of the virtual world and these are preserved in the retargeted world. The floor plan and the room boundaries detected aid in this process. The virtual world objects are placed as per the result of the optimization routine.

The generation of retargeted virtual worlds was implemented in Unity game engine. Various virtual world objects were downloaded from the unity asset store ["Unity"].

Chapter 4

Results and Discussion

In this chapter, some results of the retargeting scheme are presented.

The constraints in the physical environment and the constraints in the virtual world are two important parameters in this problem. So, a set of results are presented by varying these two parameters in order to give an intuition about the working of the algorithm. Finally, examples of retargeting virtual worlds to a real world physical environment are presented.

Further, the results also depend on how the three scores, namely, MS,CS and SS are weighted and the optimal weighting depends on the semantics of the virtual world.

4.1 Varying Constraints in the Physical Environment

The constraints in the physical environment include the layout of the physical environment and the number of real world obstacles. These influence the mismatch score. The more the obstacles, the more difficult it is to achieve an acceptable retargeting while optimizing for the three scores. Three representative cases with no obstacles, a few obstacles and many obstacles are presented for the same layout.

4.1.1 Only Layout Constraints - No Obstacles

When there are no real world obstacles, we only need to preserve the semantic relationships between virtual world objects and avoid collisions between them. The only change from the virtual world is in the different layout of the physical environment. The mismatch score becomes redundant in this case. This case is shown in Fig 4.1.

It is notable that even for this simple case, the relative distances between virtual world objects are not preserved. This is simply because it is not one of the goals of the optimization and not included in the fitness function. It is possible to choose to preserve relative distances too, in which case the problem would become even more constrained for the typical case of a different layout with obstacles. It was a design choice that preserving the relative distance order is sufficient to preserve the experience of the virtual world.

4.1.2 A few Obstacles

This case is shown in Fig 4.2. As the number of obstacles is reasonable, it is usually possible to achieve a retargeting that does well on all the three fronts.

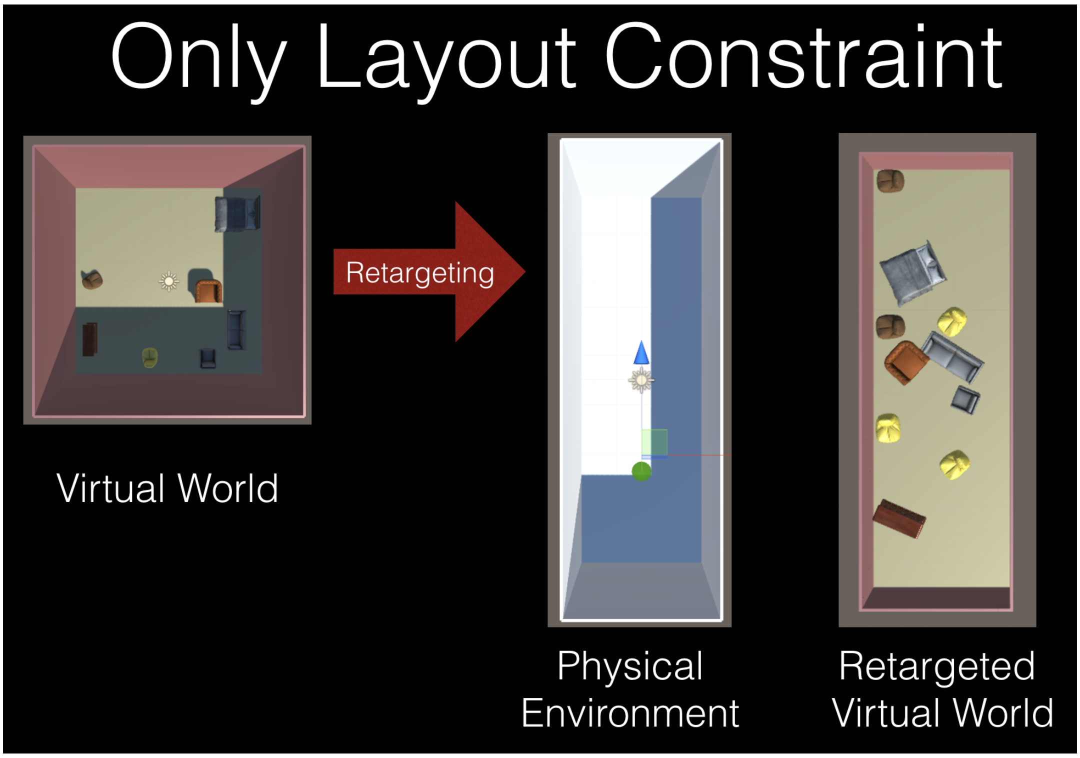

FIGURE 4.1: Only Layout Constraint

4.1.3 Many Obstacles

As the number of obstacles increases, optimizing for all the three criteria becomes hard. It is quite possible that such a solution does not exist. In this case, we are forced to make trade-offs.

This case is shown in Fig 4.3. In order to achieve a reasonable MMS, some collisions have been allowed.

Knowledge of the scene semantics can help us to make the right trade-offs. For example, collision between trees in a forest scene is acceptable but collision between furniture in an indoor scene is unacceptable. This knowledge of scene semantics can be used to weight the most important score heavily. However, it can be generalized that in any scene, mismatch score is very important to ensure the safety of the user by avoiding collision with real world objects.

In general, retargeting is hard when there are too many obstacles in the physical environment. It is also hard when the layout of the physical environment is very different from that of the virtual world.

4.2 Varying Constraints in the Virtual World

The constraints in the virtual world influence the importance of the semantic score in the optimization. The more the number of priority objects and lesser the number

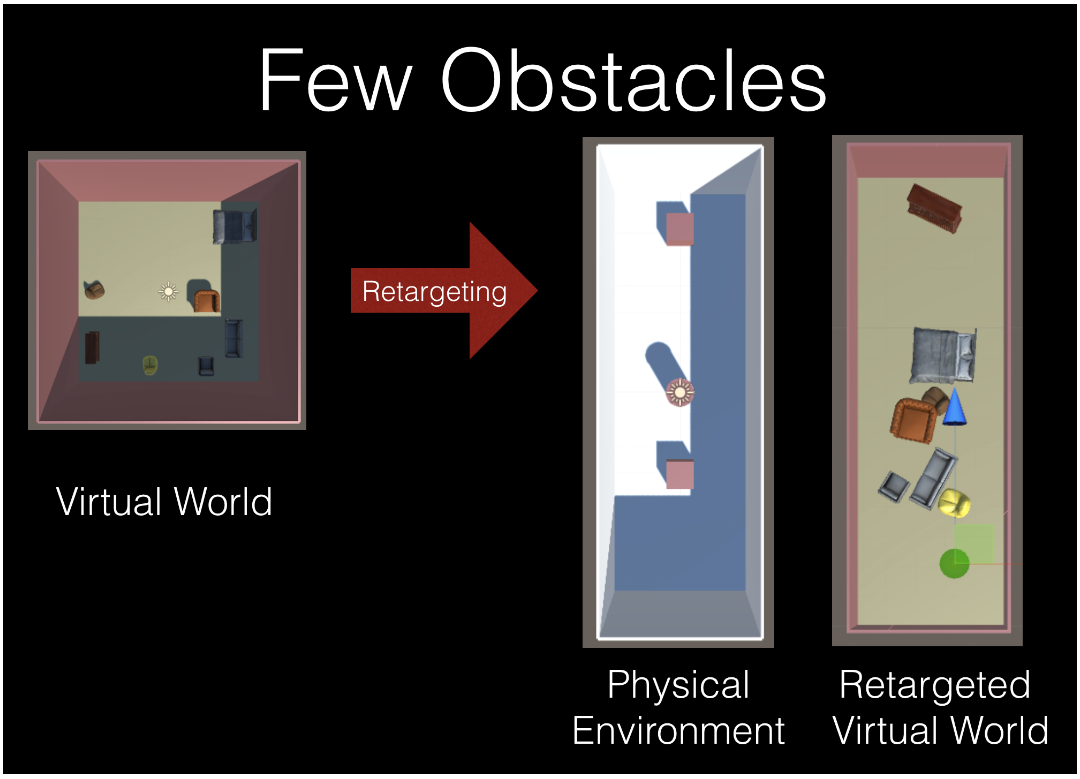

FIGURE 4.2: A few Obstacles

of filler objects, the more difficult it becomes to achieve a good retargeting. Representative cases are presented with varying number of priority and filler objects.

4.2.1 All Priority Objects

This case is the hardest as the number of objects is fixed and the semantic interrelationships between each pair has to be preserved. This has been shown in Fig. 4.4. It can be seen that in order to preserve the semantic relationships between all the virtual world objects, we have compromised a good match with the physical environment.

4.2.2 Some Priority and Some Filler Objects

In this case, the semantic score preserves the semantic relationships between the priority objects while the filler objects, that do not contribute to the semantic score, can be placed in such a way as to reduce the mismatch and collision scores.

This is the typical case for which the retargeting scheme was designed. Fig 4.1-4.3, Fig 4.6 and Fig 4.7 are examples of this case.

4.2.3 All Filler Objects

This case is the easiest in which the semantic score becomes redundant. As a consequence, it becomes easier to achieve lower levels of mismatch and collision scores. This case is shown in Fig 4.5.

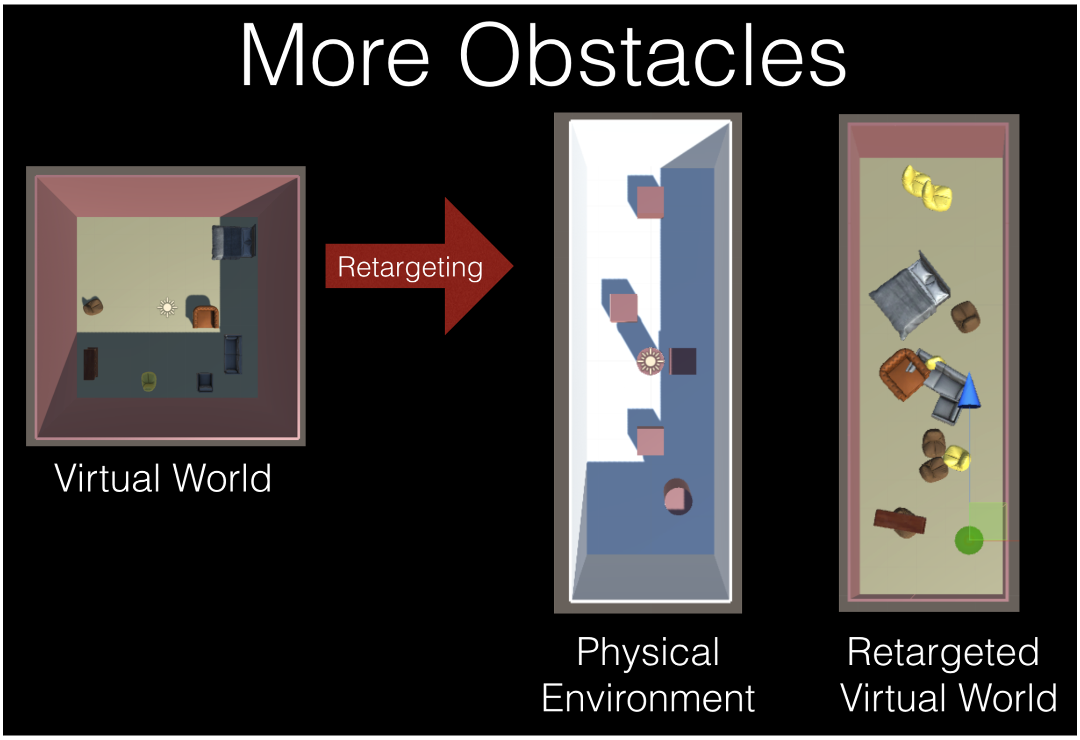

FIGURE 4.3: Many Obstacles

From the examples, we see that retargeting is hard when there are too many priority objects and too few filler objects in the virtual world. In general virtual worlds with not too many priority objects and a large number of filler objects are 'more retargetable'.

4.3 Retargeting to Real World Physical Environments

The previous examples give an intuition about the workings of the retargeting scheme. This section presents virtual worlds retargeted to a real world physical environment.

Fig 4.6 and Fig 4.7 show virtual worlds retargeted to a real world physical environment and the MMS, CS and SS obtained during the optimization. These scores give an idea of how successful the retargeting was in terms of the desired goals.

4.4 Additional Results - Second Filler Pass

A close observation of the presented results reveals that the optimized solution obtained from the genetic algorithm is not the optimal one. In most of the cases, it is possible to imagine an arrangement of the virtual world objects that would achieve better values of MMS, CS and SS. For example, in Fig 4.6, placing the two-seater sofa closer to the wall such that it faces the armchair and the one-seater would result in a better arrangement. More alarming is the fact that there are no obstacles in the virtual world in place of two one-seaters in the physical environment. These observations lead us to conclude that the genetic algorithm is not converging to the

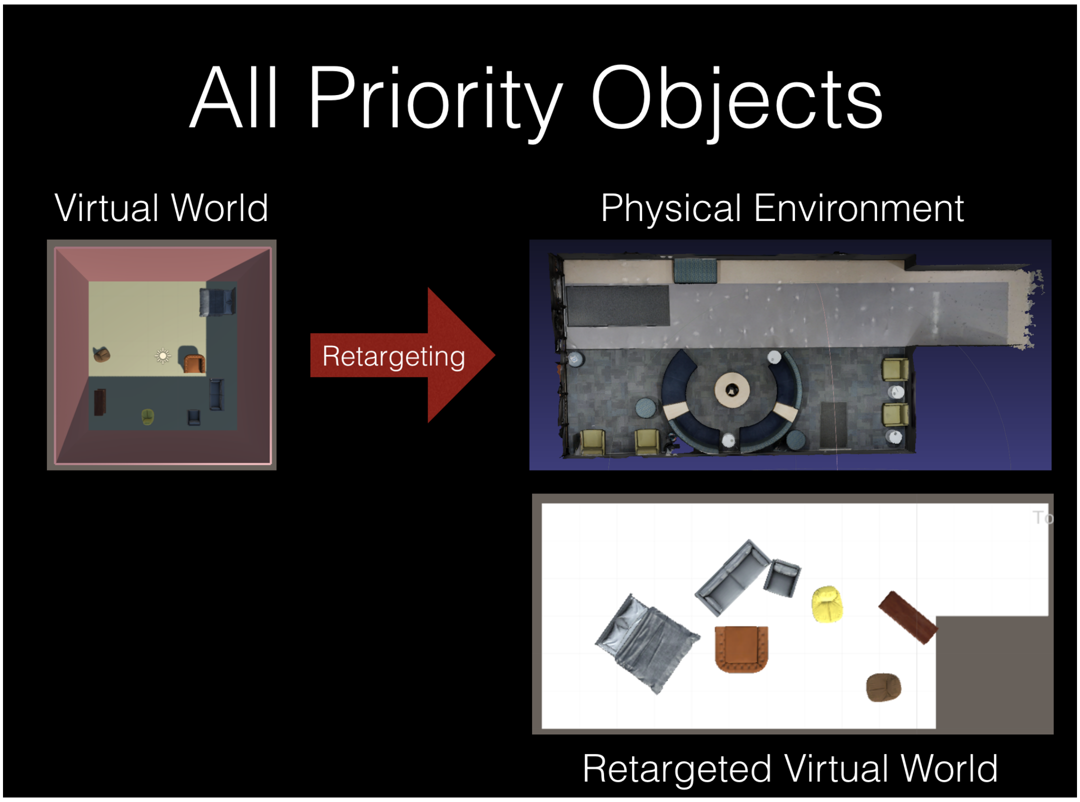

FIGURE 4.4: All Priority Objects

most optimal solution and is instead, getting caught in local minima. To fully solve this problem, we require a better optimization technique. However, it is worthwhile to try simpler techniques to alleviate the last of the above mentioned problems, as it is a safety concern.

In order to ensure that obstacles in the physical environment are always represented by obstacles in the retargeted virtual world, a second pass in which filler objects are suitably placed was carried out on the optimization result. This section presents the results obtained by this effort.

A greedy and simple sliding window algorithm was implemented to place the filler objects in the layout obtained from the optimization in order to improve the match with geometry of the physical environment while avoiding collisions. The filler object is placed at each location in the retargeted virtual world to determine the effect of such a placement. The filler object is finally placed at a particular location if the placement reduces MMS without increasing CS. Recall that SS is not affected by filler objects. We can place as many filler objects as required to get a desirable result.

It was observed that this second filler pass helped to improve the MMS without affecting the CS and the SS.

Fig 4.8 shows the result of Fig 4.6 after a second filler pass. It is notable that there are now brown bean bags (filler objects) in place of the one-seaters in the physical

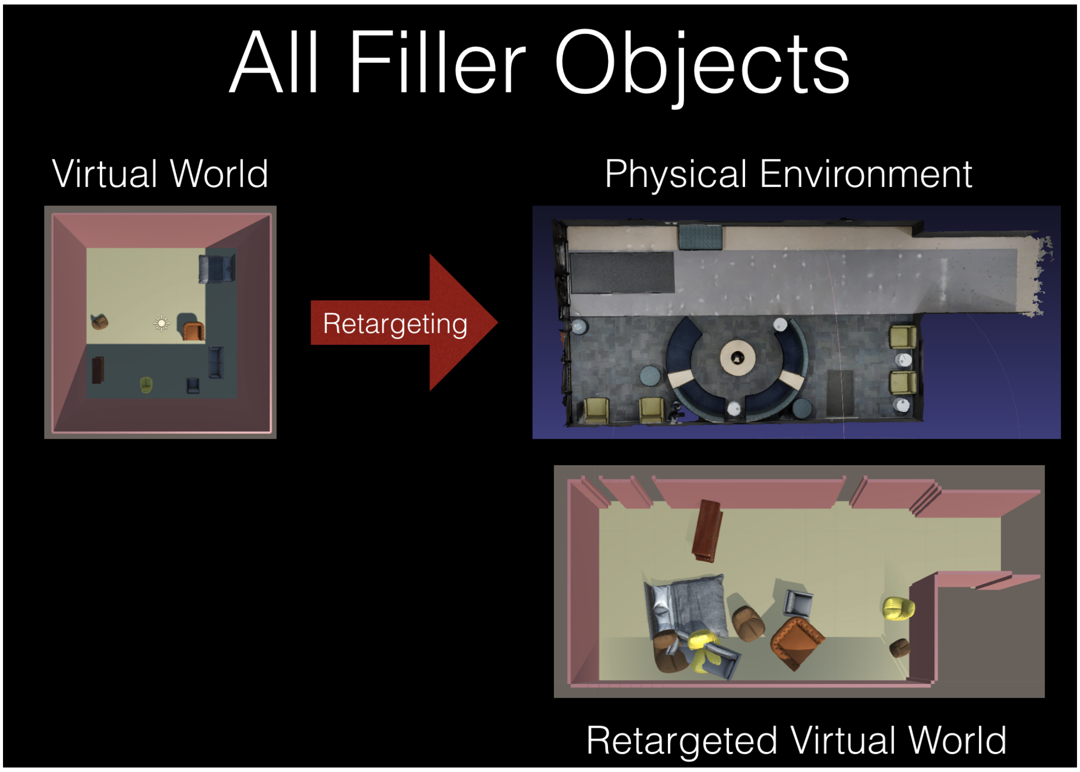

FIGURE 4.5: All Filler Objects

environment that were originally unrepresented in the retargeted virtual world. The new value of MMS is 0.4862.

Fig 4.9 shows the result of Fig 4.7 after a second filler pass. It is notable that there is now a rock (filler object) in place of the one-seaters in the physical environment that were originally unrepresented in the retargeted virtual world. The new value of MMS is 0.4427.

The CS and SS remain the same as the previous results. Thus, MMS has been lowered without allowing collisions or compromising semantics any further.

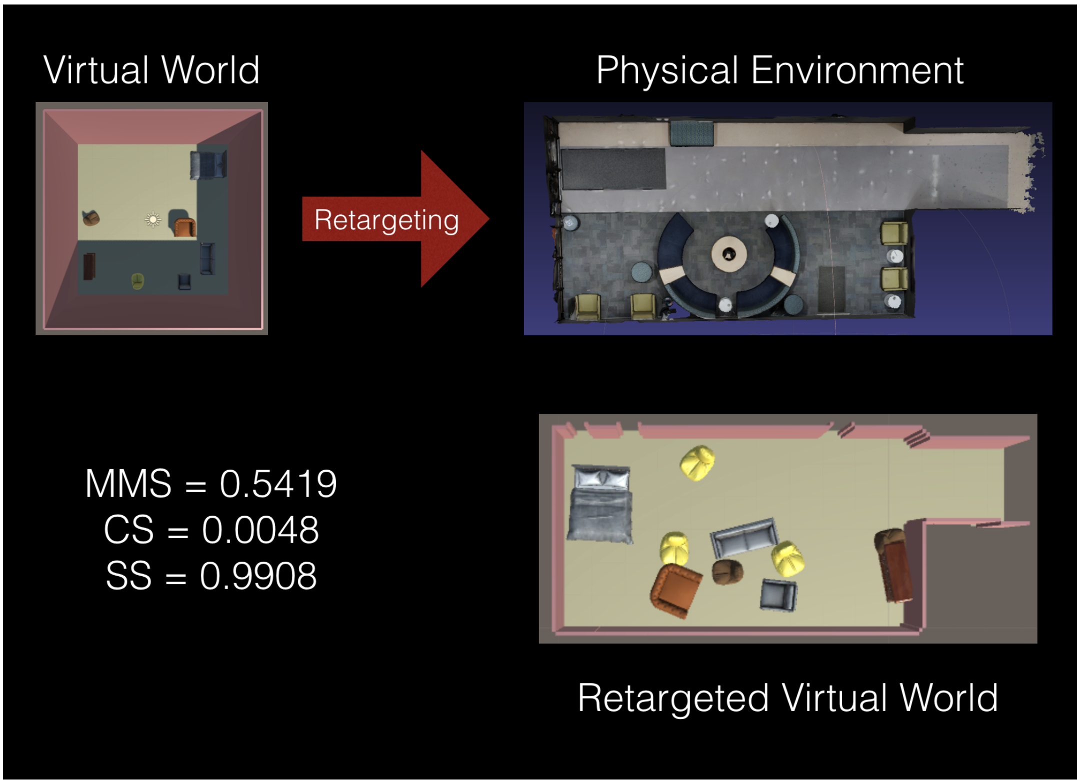

FIGURE 4.6: Living Room retargeted to CS Lounge

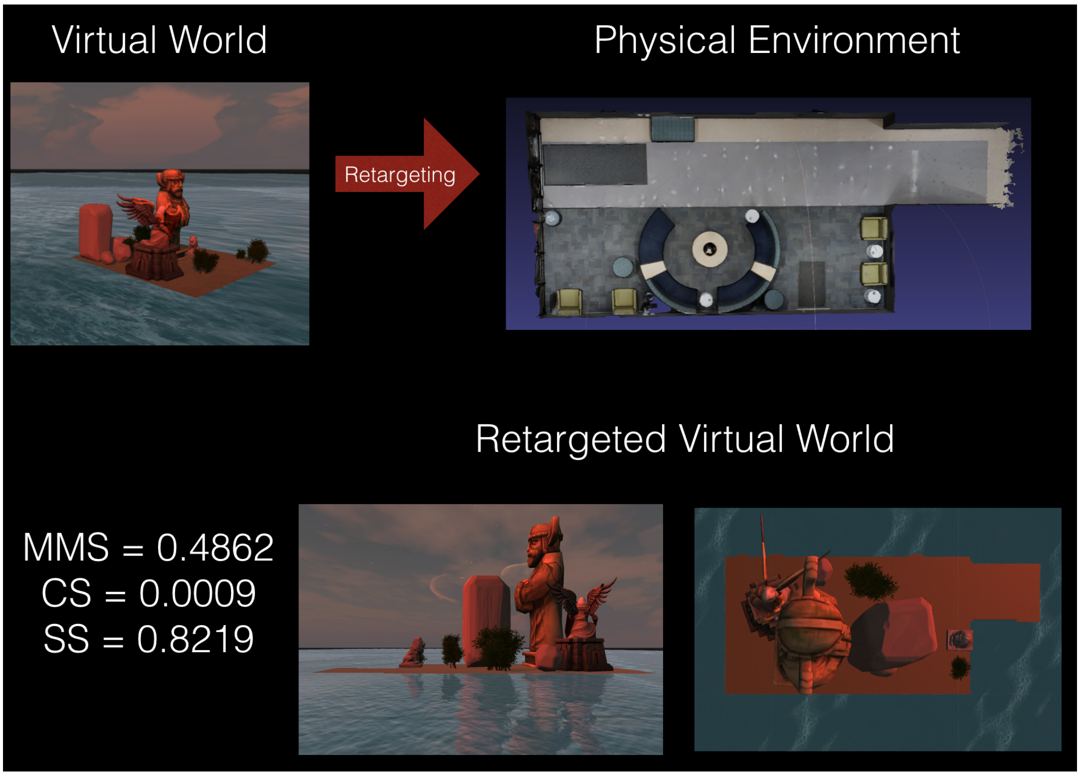

FIGURE 4.7: Island retargeted to CS Lounge

FIGURE 4.8: Living Room retargeted to CS Lounge after Second Filler Pass

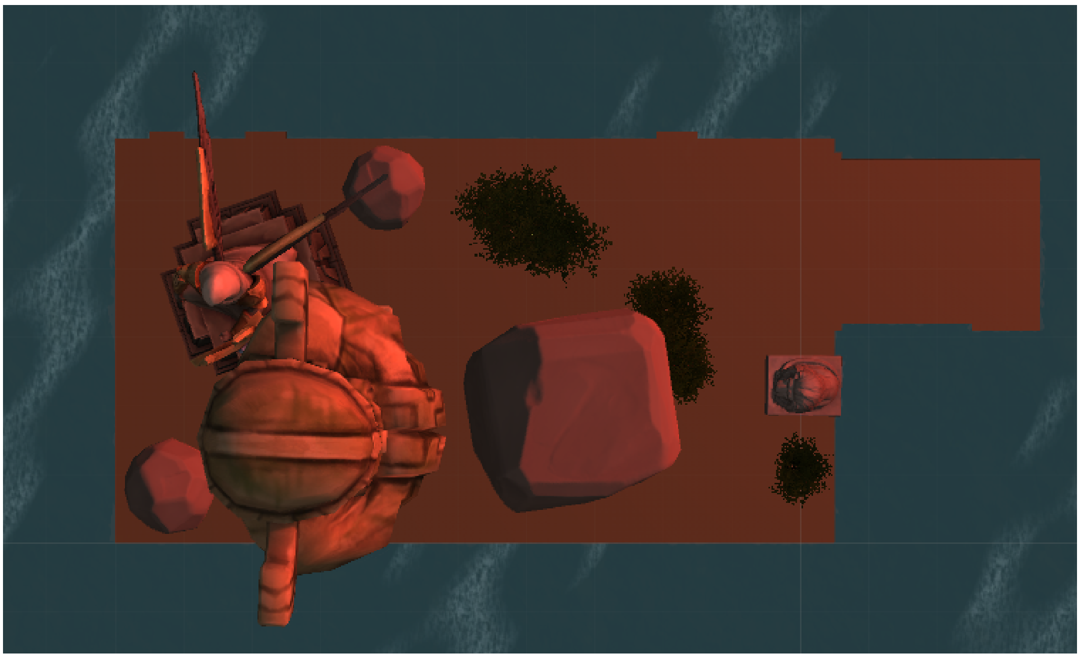

FIGURE 4.9: Island retargeted to CS Lounge after Second Filler Pass

Chapter 5

Conclusion and Future Work

A scheme to retarget a virtual world designed by a designer to the physical environment of the user was presented. The retargeting preserves the semantic relationships in the virtual world, while maximizing the geometric match with the physical environment of the user.

A primary assumption in our work is that an experience of the virtual world can be conveyed by its environment and certain relationships between virtual world objects. However, this information is not sufficient to convey the experience of some virtual worlds. For example, imagine an attempt to retarget a historic monument or a new apartment building. Retargeting would distort the layout of the virtual world and as layout is critical in these experiences, such virtual worlds are not very suitable for retargeting. On the other hand, the scheme works well for retargeting virtual worlds like outerspace, underwater, forests, etc. where the layout of the virtual world is not important and there are many filler objects. Thus, some virtual worlds are more 'retargetable' than others and our scheme is not suitable for every kind of virtual world. However, it is possible to retarget many virtual worlds with our scheme and create many compelling experiences.

A notable feature of our scheme is that it preserves the semantic relationships between the priority objects in the virtual world and places the filler objects in the virtual world to achieve a good geometric match with the physical environment. For this to work well, there should not be too many priority objects or too few filler objects in the virtual world. As shown in the results, placing additional filler objects as a second pass can increase the geometric match with the physical environment.

Another requirement for the scheme to work well is that the layout of the physical environment should not be very different from that of the virtual world. Also, the number of obstacles in the physical environment should not be very large. In such cases, the problem becomes very constrained and it is hard to find a retargeting that fulfills all the desirable conditions to an acceptable degree.

The symmetry of the virtual world objects might allow us to relax some semantic constraints. For example, relative orientation with respect to a round table can be ignored. However, this would require an additional level of annotation from the designer. This has not been considered in this work.

In this retargeting scheme, relative distance rank and relative orientation were used to capture the semantics of the virtual world. Some trade-offs between choosing relative distance order versus relative distance were presented earlier. There are many other factors that could be included in the semantics of the virtual world. We have not considered the relationships between the virtual world objects and the boundaries of the virtual world. For example, we would like to have a sofa facing away from the wall rather than one facing the wall. We may also want the sofa to be placed parallel to the wall. We have also not considered relationships between the virtual world objects and objects in the environment. For example, we might want a statue to be placed such that it faces the Sun.

We can choose to include several such scores to capture the semantic information in a virtual world. Choosing the right set of scores to capture the semantics of the virtual world to a required degree is an AI problem.

It was mentioned earlier that the relative importance of MMS, CS and SS are dependent on the virtual world and the physical environment to which it is being retargeted. MMS is not important in a physical environment that does not not have obstacles. CS is not very important if the virtual world objects are such that collision is not objectionable (Eg: rocks and trees). SS is not very important if the virtual world objects are all fillers. The optimum weighting of MMS, CS and SS for a given virtual world and physical environment for a given application is another interesting problem that has not been addressed.

Finally, it is observed that the optimized solution is often some local minimum and not the global minimum. As a result, we are not performing the best possible retargeting. The system would greatly benefit the employment of a better optimization technique that can converge to the global minimum in this high dimensional non-convex solution space.

The disconnect between a user's physical environment and the virtual world can seriously hamper the sense of presence in the virtual world. However, both altering the physical environment to suit the virtual world and generating virtual worlds entirely based on the physical environment come with disadvantages. Altering the physical environment is not attractive to the user and generating virtual worlds only based on physical environment ignores semantics. The retargeting scheme described tries to establish a middle ground. It is a step towards keeping the war of the virtual and physical worlds from making VR experiences less compelling.

Bibliography

Charbonnier, Caecilia and Vincent Trouche (2015). *Real Virtuality*. Tech. rep. http://www.artanim.ch/en/projects-detail.php?id=12. Artanim Interactive.

Gal, Ran et al. (2014). "FLARE: Fast layout for augmented reality applications". In: *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*. IEEE, pp. 207–212.

"HTC VIVE". In: https://www.vive.com/us/.

"Matterport,Inc". In: https://matterport.com/.

Merrell, Paul et al. (2011). "Interactive furniture layout using interior design guidelines". In: *ACM Transactions on Graphics (TOG)*. Vol. 30. 4. ACM, p. 87.

Shapira, Lior and Daniel Freedman (2016). "Reality Skins: Creating Immersive and Tactile Virtual Environments". In: *Mixed and Augmented Reality (ISMAR), 2016 IEEE International Symposium on*. IEEE, pp. 115–124.

Sra, Misha et al. (2016). "Procedurally generated virtual reality from 3D reconstructed physical space". In: *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*. ACM, pp. 191–200.

Sun, Qi, Li-Yi Wei, and Arie Kaufman (2016). "Mapping Virtual and Physical Reality". In: *ACM Trans. Graph.* 35.4, 64:1–64:12. ISSN: 0730-0301. DOI: 10.1145/2897824.2925883. URL: http://doi.acm.org/10.1145/2897824.2925883.

"The Void, LLC US". In: https://thevoid.com/.

"Unity". In: https://www.assetstore.unity3d.com/.

Usoh, Martin et al. (1999). "Walking> walking-in-place> flying, in virtual environments". In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., pp. 359–364.

Yu, Lap Fai et al. (2011). "Make it home: automatic optimization of furniture arrangement". In: