Optimization and Manipulation of Contextual Mutual Spaces for Multi-User Virtual and Augmented Reality Interaction

Mohammad Keshavarzi[* 1,2]   Allen Y. Yang[† 1]   Woojin Ko[‡ 1]   Luisa Caldas[§ 2]

[1] FHL Vive Center for Enhanced Reality, University of California, Berkeley
[2] XR Lab, Department of Architecture, University of California, Berkeley

ABSTRACT

Spatial computing experiences are physically constrained by the geometry and semantics of the local user environment. This limitation is elevated in remote multi-user interaction scenarios, where finding a common virtual ground physically accessible for all participants becomes challenging. Locating a common accessible virtual ground is difficult for the users themselves, particularly if they are not aware of the spatial properties of other participants. In this paper, we introduce a framework to generate an optimal mutual virtual space for a multi-user interaction setting where remote users' room spaces can have different layout and sizes. The framework further recommends movement of surrounding furniture objects that expand the size of the mutual space with minimal physical effort. Finally, we demonstrate the performance of our solution on real-world datasets and also a real HoloLens application. Results show the proposed algorithm can effectively discover optimal shareable space for multi-user virtual interaction and hence facilitate remote spatial computing communication in various collaborative workflows.

Index Terms: Computing methodologies—Computer graphics—Graphics systems and interfaces—Mixed / augmented reality; Human-centered computing—Human computer interaction —Interaction paradigms—Collaborative interaction; Applied computing—Decision analysis—Multi-criterion optimization and decision-making; Theory of computation—Mathematical optimization—Optimization with randomized search heuristics—Evolutionary algorithms

1 INTRODUCTION

The emerging fields of augmented reality (AR) and virtual reality (VR) have introduced a large number of exciting applications in telecommunication, immersive collaboration, and social media where multiple users can share a virtual environment. While much work has been done on 3D capturing methods, real-life avatar modeling, and virtual social platforms, one key challenge in AR/VR immersion is the scene understanding of the users' surrounding spaces and the question of how to optimally utilize them for immersive tasks.

More specifically, acquiring an accessible 3D workspace is a prerequisite for a virtual or augmented immersion experience. Furthermore, the augmentation of the virtual data in the physical space must be compatible with the contextual properties of the physical space, such as a floor that is standable, a chair that is sittable, and a wall that is also a physical barrier of virtual interactions. For many 6 degrees-of-freedom (DOF) VR applications, the user will often be asked to manually initiate a block of free space where the VR

---

[*]e-mail: mkeshavarzi@berkeley.edu
[†]e-mail: yang@eecs.berkeley.edu
[‡]e-mail: woojin_ko@berkeley.edu
[§]e-mail: lcaldas@berkeley.edu

immersion can be assumed to be safe. Inferencing the above contextual information for both AR and VR can be readily done using several well-established 3D modeling algorithms in computer vision. Current AR devices, such as the HoloLens or MagicLeap, integrate such algorithms to estimate the layout of the space, including floors, walls, and ceilings, and typical furniture objects such as tables and chairs. In this paper, we assume such contextual information of individual spaces to be available via either a manual or algorithmic process.

However, in scenarios where an immersive experience involves multiple users, understanding of spatial constraints is elevated to that of all involved users. Since different users may participate in the immersive experience from their own spaces, which can hold very contrasting contextual properties, a consensus must be established to identify a mutual space that respects the spatial constraints of all the participants. Yet, having users manually identify such a mutual space would be imprecise and labor intensive, especially when considering the fact that it would be difficult for a user to be aware of the contextual properties of the other users' spaces. Without more effective and efficient solutions, the establishment of a contextual mutual space will be a bottleneck for multi-user immersion experiences.

Motivated by this challenge, we present in this paper a novel method to optimize contextual mutual spaces in a multi-user immersion setting. Our method relies on existing semantic scene maps to identify shareable functional spaces, and is general enough to optimize contextual mutual spaces even when the users' spaces have very different layouts and sizes (see results in Figure 6). For illustration purposes, we will use standable and sittable as the two exemplary contextual functions to develop our method, and the proposed solution is compatible with other contextual functions that can be modeled by the same mathematical framework. The method formulates an optimization problem to seek the maximal mutual spaces. Furthermore, if one can assume the users have the freedom to rearrange furniture objects on the floor, we introduce a more delicate optimization process to further increase the mutual space's size while balancing the users' efforts to physically move the objects as another constraint. To effectively solve the above two problems, we propose to use a genetic algorithmic approach. Clearly, we believe other comparable algorithms that optimize these NP-Hard problems are equally effective. Nevertheless, our results validate a new approach capable of automatically recommending contextual mutual space to multiple participants of virtual immersion experiences in AR/VR applications.

We believe our proposed framework can play a role in facilitating remote workplace practices and virtual collaborations by decreasing the spatial requirements for tele-presence systems. Instead of setting up large open spaces required for such workflows, our system would allow users to join from their personal spaces, with minimum modifications to their surrounding environment. Physical and virtual re-arrangements would be optimized based on the number of participant and their local environments. In AR experiences, the topological relationship and line of sight between all participants would be maintained without any conflicts between remote users

Figure 1: Abstract illustration of our proposed framework a) initial settings with different spatial restrictions b) semantic segmentation defining standable (yellow boundaries) and sittable (orange boundaries) areas c) search for mutual sittable space (this step can be before, after or simultaneous with object repositioning) d) virtual arrangement of avatars with deterministic line of sight of all participants.

and local physical obstacles, while in VR, our system can recommend spatial modifications and provide the required interaction area between multiple users.

2 RELATED WORK

Immersive AR/VR systems have been widely explored for remote tele-presence applications, providing real-time capture, transmission and display between participants of the platform [5, 19, 34]. Using an array of cameras [33, 66, 68] or depth sensors monitoring the capture space [26, 45, 47, 57], holographic replicas or avatars [41, 71] of the virtual participants are projected in pre-defined local spaces. Such projections have been extensively developed using situated autostereo [47, 49], volumetric [25], lightfield [3], cylindrical [30], and holographic [8] displays. However, participants of such systems are mainly stationed in predefined spaces [7, 44, 72, 76] to avoid any geometrical conflicts with surrounding features in the projected space. Such approach limits free-form motion of the participants within each other's location, an important factor for achieving co-located presence.

The importance of free-form user movement and the ability to preserve mobility-based communication features such as walking, gestures, and head movement have been studied greatly in the context of co-located collaboration [4, 12, 29, 42]. Another vital aspect of sharing mutual space is described in Clark's work as grounding [17]. Grounding in communication (or common ground) is a concept that comprises the collection of "mutual knowledge, mutual beliefs, and mutual assumptions" that is essential for communication between two people. Successful grounding in communication requires parties

to coordinate both the content and process [36]. As content in spatial computing can also involve the surrounding space itself, providing a common virtual ground can be critical to allow all communication features to be reflected correctly.

More recent examples have explored how tele-presence can be conducted with less spatial constraints, allowing fluid user motion in both ends of the communication. Works of [20] and [6] are examples of such systems where users and their local interaction spaces are continuously captured using a cluster of registered depth and color cameras. However, these systems use stereoscopic projection which limits the ability for remote and local users to access each others space. Instead, spaces are virtually disconnected and interaction occurs through a window from one space into the other. Meanwhile, the Holoportation system introduced by Orts-Escolano et al. allows bilateral tele-presence between participants where participants share a common virtual ground [54]. Their system allows the remote user to be rendered into the local user's space as an avatar while the local user appears as an avatar in the remote user's space as well. Such an approach is also seen in [46], where the remote and local users do not share the same functional layout of rooms, but they are calibrated in order to provide the required mutual virtual ground between users.

While tele-presence systems via shared spaces present novel workflows for capturing and projecting virtual avatars, the issue of avoiding physical and virtual conflicts within the shared spaces is still an open challenge. Narang et al. [52] developed a system which generates non-colliding movements for human-like agents interacting with other agents or avatars in a virtual environment. Lang et al. [35] integrate scene semantics with a Markov chain Monte Carlo optimization method to find optimized locations for placing virtual agents close to a single user. Such approach addresses the spatial limitations of a single user, but not multiple constraints generated by multiple remote user. The work of Lehment et al. [37] may be the closest work to this paper, which proposes an automated method to align remote environments so that they minimize discrepancies in room obstacles and physical barriers. However, the method is limited to two spaces and uses a brute force search to calculate the consensus space between participants. Our method formulates rigorous optimization problems to search and manipulate a potentially unlimited number of spaces in order to find a mutual spatial boundary.

For virtual reality environments, techniques in redirected walking [60] also aim to resolve the possible conflicts of virtual and physical surroundings. While the focus is mainly on providing a natural locomotion of a local user, such techniques use subtle (redirected without the user's knowledge) [9, 11] or overt (detectable by the user) [23, 56, 73] strategies to manipulate the mapping between the user's real and virtual translation and rotation, resulting the user to avoid interference with edges of the usable space or physical obstacles. Architectural manipulation of virtual spaces has also been investigated by re-arranging virtual elements in blind-spots [63] or implementing self-overlapping [64] and flexible virtual spaces [69]. However, redirected walking techniques may introduce simulator sickness [53], interfere with spatial memory [73], and lead to higher cognitive load than real world locomotion [10]. Furthermore, while such strategies can be applied in VR environments, they cannot generally apply for AR experiences due to the see-through nature of AR. Even so, the ability to efficiently manipulate the real-world surroundings introduced by our system would provide more spatial freedom, especially in remote mulit-user scenarios, before applying multi-user redirecting walking techniques.

Part of our proposed system intends to determine an optimal arrangement of discrete spatial elements within a room. Such practice is often referred to as floorplanning [18]. Automated floorplanning methodologies have been widely investigated in architectural space layouts, construction [16, 55, 67], electronic design [14, 22, 50], and industrial operation research [1]. Floorplanning aims to achieve

a defined functional goal by efficiently generating and evaluating possible spatial combinations while addressing the geometrical and topological constraints of the spatial elements [24]. In electronic physical design floorplanning, proposed methodologies mostly aim at optimizing chip area and wirelengths to reduce interconnections and improve timing [27]. In construction site layout and planning, optimizing the interaction between facilities, such as total inter-facility transportation costs and frequency of inter-facility trips can also be implemented as objective functions [55]. In our proposed framework, we similarly integrate an objective function whose goal is to minimize the amount of effort required to move surrounding furniture while maximizing the area of the mutual virtual ground among all participants.

In floorplanning, various representation methods of spatial arrangements are coupled with optimization engines to efficiently search through all possible combinations of spatial elements. Floorplanning representations are generally divided into two main categories: slicing and non-slicing representations [70]. In slicing methodologies, the floor plan is recursively bisected until each part consists of a single module [74]. Non-slicing representation are utilized for more general use cases where no recursive bisection of a certain area takes place [21, 39, 43]. Multiple studies have integrated these representations with various optimization algorithms such as Simulated Annealing (SA) [31, 32, 74], Genetic Algorithms (GA) [22, 38, 51, 61, 75] and Particle Swarm Optimization (PSO) [15, 28, 48, 62, 65]. More recently, by applying learning based algorithms, hybrid neural networks[7] and annealed neural networks have been used to identify optimal site layout and solve construction site-level problems.[8]

3 METHODOLOGY

Our solution consists of the following four steps: (i) Semantic segmentation of surrounding environments; (ii) Topological scene graph generation; (iii) Mutual space identification; (iv) Optionally, manipulation of ground objects to further maximize the mutual space. In this section, we will elaborate on the details of the four steps. To start, we will define the terminologies and notations used in the paper.

Given a closed 3D room space in $\mathbb{R}^3$, one can project its enclosure, i.e., floors, ceilings, and walls, via an orthographic projection to form a 2D projection, which is commonly known as the floor plan of the space. If we assign the (x, y) coordinates on the floor-plan plane and the z coordinate perpendicular to the floor-plan plane, simplifying our optimization problems on the (x, y) plane significantly reduces the complexity of our algorithms. It also implies an assumption that there is no overlap between two objects on the (x, y) plane but with different z values. Nevertheless, we believe such simplification is reasonable for analyzing the majority of room structures and thus does not compromise the generality of our analysis provided herein.

Hence, we define for each user i their own room space expressed as a 2D floor plan as R_i. Each k-th object (e.g., furniture) in R_i is denoted as $O_{i,k}$. The collection of all n_i objects in R_i is denoted as $\mathcal{O}_i = \{O_{i,1}, O_{i,2}, ...O_{i,n_i}\}$. $\bar{O}_{i,k}$ represents the boundary of the object $O_{i,k}$. Similarly, $\bar{R}_i$ represents the boundary of the room R_i. Finally, we define the area function as $K(O)$.

3.1 Semantic Segmentation

Given the measurement of the surrounding physical environments as large sets of point cloud data, one can take advantage of the semantic segmentation methods widely investigated in computer vision literature [2, 40, 58] to segment their spatial boundaries and obtain their geometric properties, such as dimensions, position and orientation, object classification, functional shapes, and their weights. In doing so, we can convert the 3D point cloud data to labeled objects $O_{i,k}$ with a bounding box as $\bar{O}_{i,k}$.
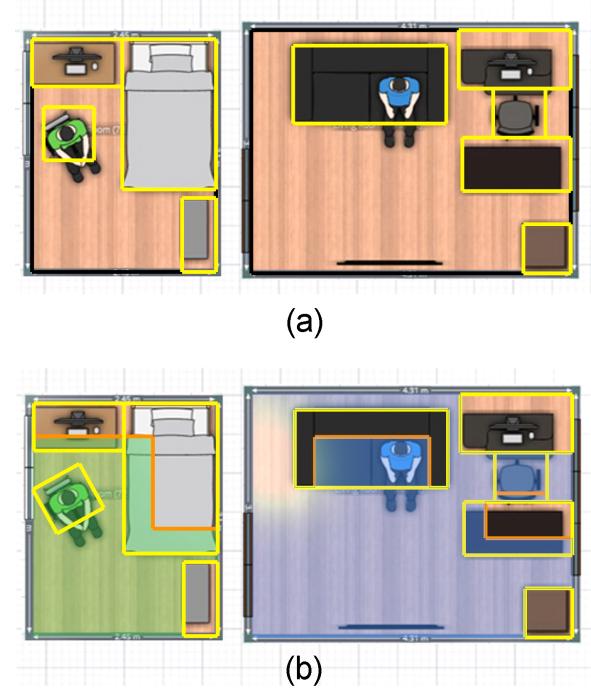

(a)

(b)

Figure 2: Comparison between available (a) standing only and (b) standing and sitting area in rooms.

Additionally, in this paper we exclude lightweight objects (such as pillows, alarm clocks, laptops, etc.) positioned on larger furniture. This is to simplify our calculations in the next steps as we assume these lightweight objects can be easily moved by the users and do not need to be considered in the optimization criteria. Such classification is dependent on the output labeled object categories above.

In the experiment section below, since the implementation of a computer vision algorithm for semantic segmentation is not the main focus of this paper, we will directly integrate a modified version of Matterport 3D [13] object classifier in our system. This module can be replaced with any other robust semantic segmentation algorithms, as long as they provide bounding box coordinates for each object category. In a companion Matterport 3D [13] dataset, out of 1,659 unique text labels, we classify 134 of the labels as lightweight objects and filter their corresponding bounding box from our workflow.

Figure 2(a) illustrates the result of semantic segmentation of two room spaces projected onto the (x, y)-plane.

3.2 Topological Scene Graph

After identifying the bounding box, orientation, and category type of each object in the scene R_i, a topological graph is readily generated that describes the relationship and constraints of the objects between one each other within R_i. This step will allow us to identify usable spatial functions such as standing in virtual immersion, located between the objects. We categorize this type of functions as *standalone spatial functions*, and their spaces are called *standalone spaces*.

A topological scene graph will also allow us to identify other spatial functions on the objects themselves such as sitting on a chair and working on a table. But note that such functions as sitting or working are also constrained by the distances between the object that performs the function and its adjacent other objects. For example, a side of the table can not be utilized for working purposes if that side is adjacent to other furniture or building elements (such as walls,

doors, etc.). We categorize this type of functions as *auxiliary spatial functions*, and their spaces are called *auxiliary spaces*.

In this paper, we will use two spatial functions *standable* and *sittable* as an example to demonstrate how to integrate both standalone spatial functions and auxiliary spatial functions in the optimization of contextual mutual spaces for mutli-user interaction in AR/VR.

Finally, we emphasize that standalone spaces and auxiliary spaces are not mutually exclusive. For example, in this paper, we will classify that a standable space can be assumed to be sittable as well. However, the vice versa may not be true. For example, a portion of a sittable space involves a part of a bed object, which we will not assume to be standable. Such contextual constraints can be highly customizable based on the content of the AR/VR application. But the framework that we are introducing in this paper is general enough to accommodate other contextual interpretations of the standalone spatial functions and auxiliary spatial functions.

In our implementation, we use a doubly-linked data structure to construct the graph. For each side face of an object's bounding box we define the closest adjacent objects to the face and calculate the distance between the object and the specified face. This information would be stored at the object level, where topological distances and constraints are referenced using pointers.

Mathematically, for each object $O_{i,k}$, we define the function $\delta X_{\max}(O_{i,k})$ as the shortest distance between the points in $O_{i,k}$ that have the maximal x value and the other objects including $\bar{R}_i$. Similarly, we define the functions $\delta X_{\min}(\cdot)$, $\delta Y_{\max}(\cdot)$, and $\delta Y_{\min}(\cdot)$.

3.3 Mutual Space Identification

In this step, we will identify the geometrical boundaries of available spaces in each room and then align the calculated boundaries of all rooms to achieve maximum consensus on mutual spaces.

First, using the geometrical and topological properties extracted in previous steps, we are ready to calculate available spaces in each room based on two categories, namely, the standalone spaces and auxiliary spaces. Specifically, we will formulate the calculation of the two most typical spatial functions as examples again, namely, standable and sittable.

3.3.1 Standable Spaces

Standing spaces consist of the volume of the room in which no object located within a human user's height range is present. In such spaces, user movement can be performed freely without any risk of colliding with an object in the surrounding physical environment. Activities such as intense gaming or performative arts can be safely executed within these boundaries. Such spaces are also suitable for virtual reality experiences, where users may not be aware of the physical surroundings.

We calculate the available standing space (S) for room R_i simply as follows:

$$S_i = R_i - \bigcup_{k=1}^{n_i} O_{i,k}. \qquad (1)$$

3.3.2 Sittable Spaces

The calculation of maximal sittable spaces is more involved than that of the standable spaces above. As we mentioned before, sittable spaces normally extend the standable spaces by adding areas where humans are able to sit on. Furniture types such as sofas, chairs, and beds include sitting areas that can extend usable spaces of a room for social functions such as general meetings, design reviews, and conference calls.

To start, we define a sittable threshold $\varepsilon(O_{i,k})$ to calculate the sittable area within the bounding box of the object $O_{i,k}$. In other words, $\varepsilon(O_{i,k})$ is the maximum distance inward from an edge of the object's bounding box that can be comfortably sit on. We use
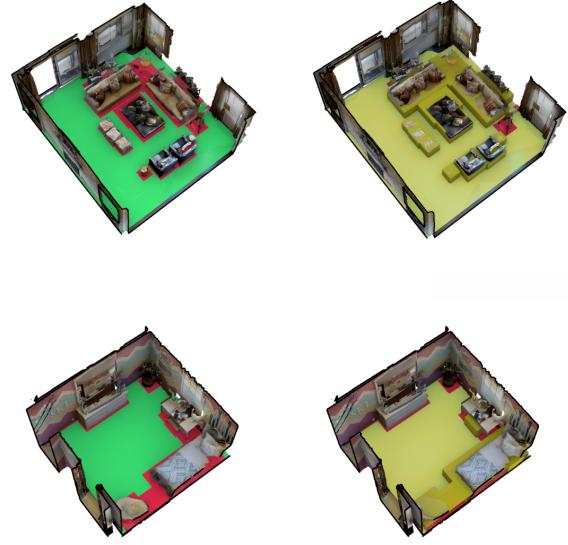

Figure 3: Standable (green), non-standable (red) and sittable spaces (yellow) for two example scenes from the Matterport 3D dataset.

measurements from [59] to define the ε of each furniture type. If object O is classified as non-sittable, then $\varepsilon(O) = 0$.

Therefore, we can first calculate the non-sittable area of an object O as

$$N(O) \doteq \{\forall p \in O : B(p, \varepsilon(O)) \cap O = B(p, \varepsilon(O))\}, \qquad (2)$$

where $B(p, \varepsilon(O))$ is a sphere in $\mathbb{R}^2$ centered at p and with radius $\varepsilon(O)$.

We note that sittable spaces do not necessarily comprise only objects to be sit on, but rather describe an area where a sittable object can be placed in. For example, while an individual may not be able to comfortably sit on the top of the table, but the foot space below the table can be considered as sittable space. Therefore, in such context the sittable area of the room is always larger than its standable area.

Moreover, sittable areas of each object in the room is constrained by the topological positioning of the object. If any of the object's boundaries is adjacent to a non-sittable object (such as a wall, bookshelf, etc) or does not contain enough standable area between itself and a non-sittable object, the sittable area of the side of the face should be excluded. For instance, if a table is positioned in the center of a room, with no other non-sittable object around it, the sittable area would be calculated by applying the sittable threshold to all four sides of the table's boundaries. However, if the table is positioned in the corner of the room, then there will be no sittable area accumulated for the sides that are adjacent to the wall.

To simplify our calculation, we define a surrounding boundary threshold $\rho(O)$ for object O, which measures the distance from any object's boundary point outward that allows that point to remain part of the sittable space of the object. In other words, if the boundary point is close to other objects or the room boundary within distance ρ, then that point can not be sit on. $C(O_{i,k})$ defined below collects all such points for exclusion from $O_{i,k}$ in room R_i:

$$C(O_{i,k}) = \begin{cases} \{\forall p \in O_{i,k} : B(p, \varepsilon(O_{i,k}) + \rho(O_{i,k})) \cap \bar{R}_i \neq \emptyset \\ \text{or } B(p, \varepsilon(O_{i,k}) + \rho(O_{i,k})) \cap O_{i,h} \neq \emptyset, h \neq k\} \end{cases} \qquad (3)$$

where $\emptyset$ denotes the empty set. Therefore, the sittable space of each

object O is simply defined as

$$A(O) = O - N(O) \cup C(O). \tag{4}$$

Finally, the total sittable space $A(R_i)$ for the room R_i is

$$A(R_i) = \bigcup_{k=1}^{n_i} A(O_{i,k}) + A(S_i). \tag{5}$$

Figure 3 illustrates two example rooms and compares their standing and sitting areas.

3.3.3 Maximizing Mutual Spaces

Now we consider an immersive experience where there are m subjects and therefore m room spaces $(R_1, R_2, \cdots, R_m)$, respectively. Then, in the (x,y)-coordinates, we define a rigid-body motion in $\mathbb{R}^2$ as $G(F, \theta)$, where θ describes a translation and a rotation.

If we want to maximize a mutual standable space, we can apply one $G(S_i, \theta_i)$ to each individual standable space S_i for the i-th user. The optimal rigid body motion then maximizes the area of the interaction space:

$$(\theta_1^*, \cdots, \theta_m^*) = \arg\max K(\bigcap_{i=1}^{m} G(S_i, \theta_i)). \tag{6}$$

Then the maximal mutual standable space can be calculated as

$$M_S(R_1, \cdots, R_m) = \bigcap_{i=1}^{m} G(S_i, \theta_i^*) \tag{7}$$

Similarly, one can calculate the maximal mutual sittable space $M_A(R_1, \cdots, R_m)$ by substituting the rigid body motions in (7) that maximizes their intersection area function in (6).

3.4 Furniture movement optimization

In the event where individual spaces R_i include movable furniture, additional optimization can be considered to potentially increase the maximal mutual spaces. Diverging from merely considering rigid-body motions to transform just the coordinate representation of the spaces, we consider moving furniture objects in space, which has an additional cost of human effort. Consequently, we will formulate this effort as part of our optimization objective.

More specifically, given a rigid-body motion G, we definite $\|G\|_t$ as the Euclidean distance of its translation vector. Then we define

$$E = w\|G\|_t, \tag{8}$$

where w is a given parameter that approximates the weight of each object. Note that such weight estimate can be looked up using architecture standards such as in [59]. Hence, if a room space R_i has n_i objects, then the total effort to re-arrange the space is

$$E(R_i, \Theta_i) = \sum_{k=1}^{n_i} w_k \|G(O_{i,k}, \theta_{i,k})\|_t, \tag{9}$$

where $\Theta_i = \{\theta_{i,1}, \cdots, \theta_{i,n_i}\}$ denotes the collection of n_i rigid-body motion parameters.

Since solving for the optimal object transformation is an NP-Hard problem, in this paper, we will demonstrate a heuristic-based but practical algorithm to optimize it in a step-by-step greedy fashion.

$$\min \sum_{i=1}^{m} E(R_i, \Theta_i^s) \quad \text{subj. to} \quad K^s(\bigcap_{i=1}^{m} G(S_i, \theta_i^s)) \text{ increases } 10\%, \tag{10}$$

where K^s indicates the area value at the s-th step with respect to transformation coefficients Θ_i^s and θ_i^s. The iteration would stop if the optimization cannot further increase the area of the mutual space.

Generation : 5

Generation : 12
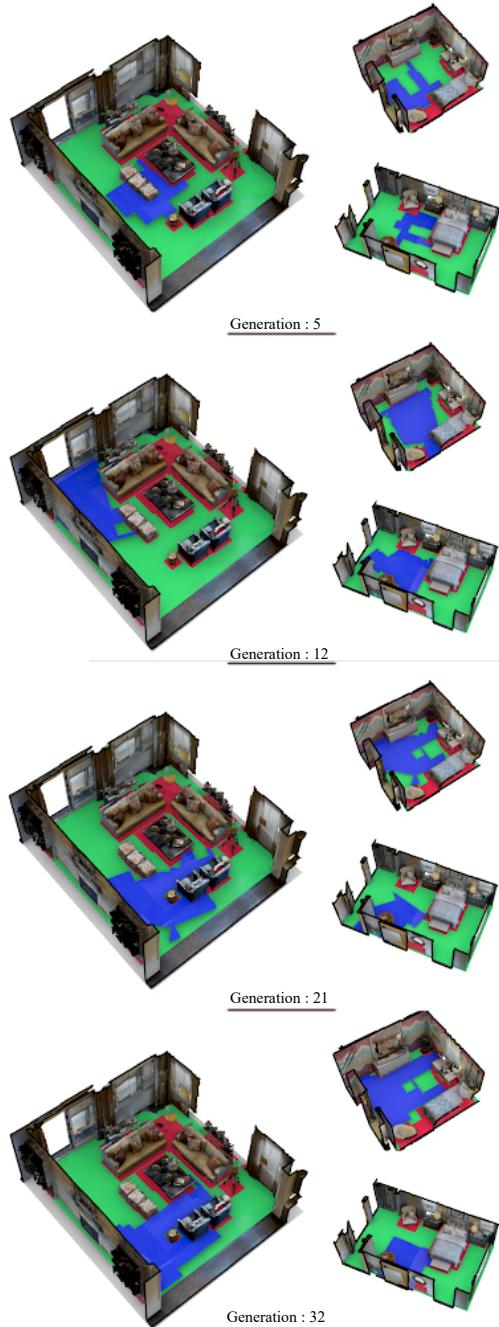
Generation : 21

Generation : 32

Figure 4: Mutual Spatial boundaries (blue) for different generations of the search mechanism. The green area indicates standable spaces and the red area indicates non-standable spaces. The result shows that the optimized mutual standable space increases over generations.

FamilyRoom
Area: 34.32 m^2
Objects: 10

Office
Area: 34.32 m^2
Objects: 8

Bedroom
Area: 50.31 m^2
Objects: 6

$M_s = 10.88$

${}^1G_s$ ${}^1K_s = 12.18$

$E_1 = 52.51$ $E_2 = 9.50$ $E_3 = 15.66$

${}^2G_s$ ${}^2K_s = 13.66$

$E_1 = 60.34$ $E_2 = 9.50$ $E_3 = 17.80$

${}^3G_s$ ${}^3K_s = 14.33$

$E_1 = 74.86$ $E_2 = 10.43$ $E_3 = 17.96$

${}^4G_s$ ${}^4K_s = 15.99$

$E_1 = 57.09$ $E_2 = 11.02$ $E_3 = 70.43$

${}^5G_s$ ${}^5K_s = 17.05$

$E_1 = 57.09$ $E_2 = 212.90$ $E_3 = 70.43$

M_s ${}^1K_s$ ${}^2K_s$ ${}^3K_s$ ${}^4K_s$ ${}^5K_s$

E

1500

1000

500

453.34

149.57
97.16
133.69
87.19
0

Pareto Front
Last GA Generation
Intermediate GA generations

K

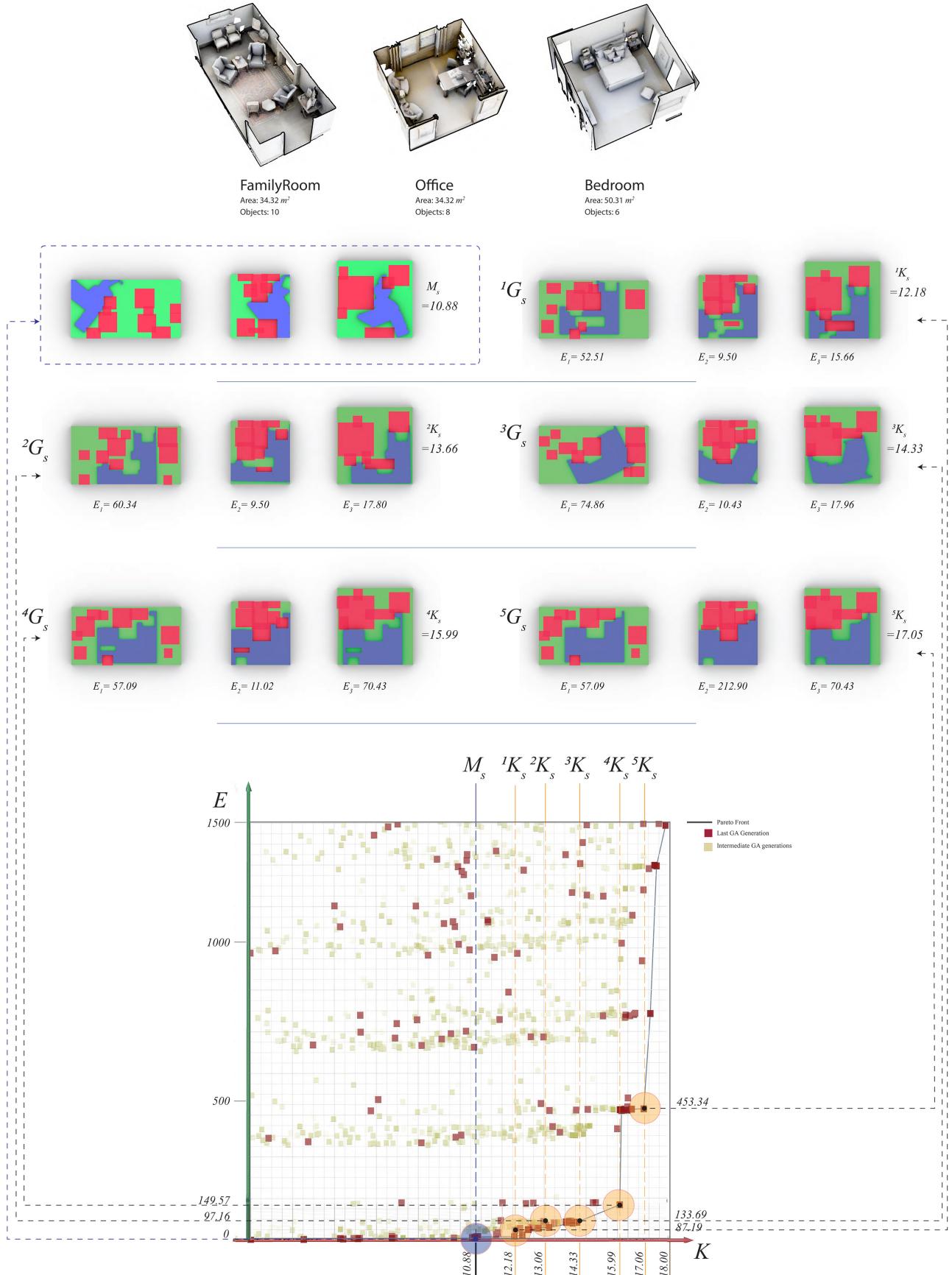10.88 12.18 13.06 14.33 15.99 17.06 18.00

Figure 5: Furniture optimization and manipulation. In each step, a 10% increase of mutual space area (K) is determined, while minimizing the overall effort needed (E) for the required transformation (G).

4 IMPLEMENTATION ON A 3D SCANNED DATASET

To comprehensively observe how the search and recommendation system performs given various rooms types with different spatial organizations, we take advantage of available 3D datasets to be able to experiment with large quantities of real-world case studies. We use the Matterport 3D [13] dataset and randomly sample subsets of varying sizes of 3D scanned scenes, and perform the search and recommendation practice on each subset to observe how the mutual spaces are identified and maximized with our algorithm. Matterport 3D is a large-scale RGB-D dataset containing 90 building-scale scenes. The dataset consists of various building types with diverse architecture styles, each including numerous spatial functionalities and furniture layouts. Annotations of building elements and furniture are provided with surface reconstructions as well as 2D and 3D semantic segmentation. For our experiments, we initially exclude spaces that are not generally used for multi-user interaction (bathroom, small corridors, stairs, closet, etc.). Furthermore, we randomly group the available rooms in groups of 2, 3, and 4. We utilize the object category labels provided in the dataset as the ground truth for our semantic labeling purposes.

We implement our framework using the Rhinoceros3D (R3D) software and its development libraries. For each room, we convert the labeling data structure provided by the dataset to our proposed topological scene graph. This provides the system with bounding boxes for each object and the topological constraints for their potential rearrangement. Using such a structure, we are able to extract the standable and sittable spaces for each room based on our proposed methodology. Figure 3 illustrates the available standable and sittable boundaries for two sample rooms processed by our system. We define a constant $\varepsilon_{O_{i,k}} = 70$ cm for all sittable objects.

Next, we integrate our algorithm with a robust Strength Pareto Evolutionary Algorithm 2 (SPEA 2) [77] available through the Octopus multi-objective optimization tool in R3D. The fitness function (6) is used to maximize the mutual space for calculated standable spaces. Our genotype is comprised of the transformation parameters $G(F,\theta)$ of each room, allowing free movement and orientation to achieve maximum spatial consensus. Therefore, a total of $3(n-1)$ genes are allocated for the search process. This process would result in the shape, position and orientation of the maximum mutual boundary of the assigned rooms. We use a population size of 100, mutation probability of 10%, mutation rate of 50% and crossover rate of 80% for our search. As our solution integrates a genetic search, we expect the result to gradually converge to the global optimum. Figure 5 shows how the mutual space boundary is progressively expanded with increase of the generations in our search.

Expanding further, we extend our search by manipulating the scene with alternative furniture arrangements. As the objective goal is to achieve an increased mutual spatial boundary area with minimum effort, we calculate the E based on the transformation parameters assigned to each object present in the room. However, in our current implementation, the genetic algorithm integrated in our solution is not capable of adapting dynamic genotype values, and therefore cannot update the topological values of each object (δX_{max}, δX_{min}, δY_{max}, δY_{min}) during the search process. Hence, to avoid transformations which result in physical conflicts of manipulated furniture, we penalize phenotypes that contain intersecting furniture within the scene. This penalty is added to the E value, lowering the probability of such phenotypes to be selected or survive throughout the genetic generations.

The optimization can either be (i) triggered in separate attempts for each step (s), where the mutual area value (K) is constrained based on the resulting step value, or (ii) executed in a single attempt where minimizing E and maximizing K are both set as objective functions. In the latter, M_S is defined as the solution which holds the largest K while $E = 0$. Executing the optimization in a one-time event is also likely to require additional computational cost due to the added complexity to the solution space.

5 RESULTS

Figure 5 illustrates our results for a furniture manipulation optimization task applied to three example rooms. A total of 34 objects are located in the rooms. To shorten our gene length we do not apply rotation transformations to objects. We use a population size of 250, mutation probability of 10%, mutation rate of 50% and crossover rate of 80% for the scene manipulation search. We visualize the standable, sittable and mutual boundaries for each spatial expansion step. Moreover we report the corresponding E for each room in the alternative furniture layout. Our results in this example indicate the solution can identify solutions which increase the maximum mutual boundary area up to 65% more than its initial state before furniture movement.

The optimization process was able to generate a well-defined Pareto front, as seen on the bottom of Figure 5, locating both the two extreme points and numerous intermediate trade-off points representing non-dominated solutions. The bottom region of the curve is flat, indicating that for a similar amount of effort, a significant increase in mutual standable area can be achieved. The trade-off frontier thus starts at point M_S, becoming very densely populated in its initial soft slope. This shows that for each modest increase in physical effort (that is, in moving furniture) there can be extensive gains in mutual shareable area, which is an interesting result. After $s = 4$, the Pareto front becomes increasingly steep, signaling that the user would now have to significantly increase physical effort levels for modest gains in shareable area. Point ${}^4G_s$ thus seems to indicate a breaking point of diminishing returns.

Similar to the M_S search, in smaller furniture optimization steps, the algorithm seeks solutions which are highly dependent on the transformation parameters $G(F,\theta)$ of the room itself, whereas in larger steps, we observe the algorithm correctly moving the objects to the more populated side of the room in order to increase the empty spaces in available. In rooms where objects are facing the center, and empty areas are initially located in the middle portion of the space, we see the objects being pushed towards the corners or outer perimeter of the room in order increase the initial unoccupied areas.

Due to the smaller gene size, calculating the optimal M_S (maximum mutual space without furniture manipulation) executes much faster compared to $E(R_i, \Theta_i^s)$ optimization, where the complexity of the search mechanism radically increases due to the additional object transformation parameters. The speed of the $E(R_i, \Theta_i^s)$ optimization is also highly dependent on the transformation range of each object, meaning that objects in larger rooms have more movement options to choose from than those in small, constrained rooms. We observe an example of this effect in the later augmented reality experiment (Section 6), where the smaller space (kitchen) dominates the search process, causing the final mutual outcome between the rooms to maintain a very similar shape to the open boundaries of the smaller space. While such an effect would still provide a well-constrained problem for medium-sized rooms with multiple objects (such as the conference room), there are many possible ways of fitting the smaller space in larger rooms with open spaces (such as the robotics laboratory), resulting in an under-constrained optimization problem.

6 AUGMENTED REALITY VISUALIZATION

To explore the usability aspect of our solution in real-world scenarios, we deploy the resulting spatial segmentation in augmented reality using the Microsoft Hololens, a mixed reality HMD. In this experiment, three types of rooms were defined as potential telecommunication spaces: (i) a conventional meeting room, where a large conference table is placed in the middle of the room and unused spaces are located around the table (ii) a robotics laboratory, where working desks and equipment are mainly located around the
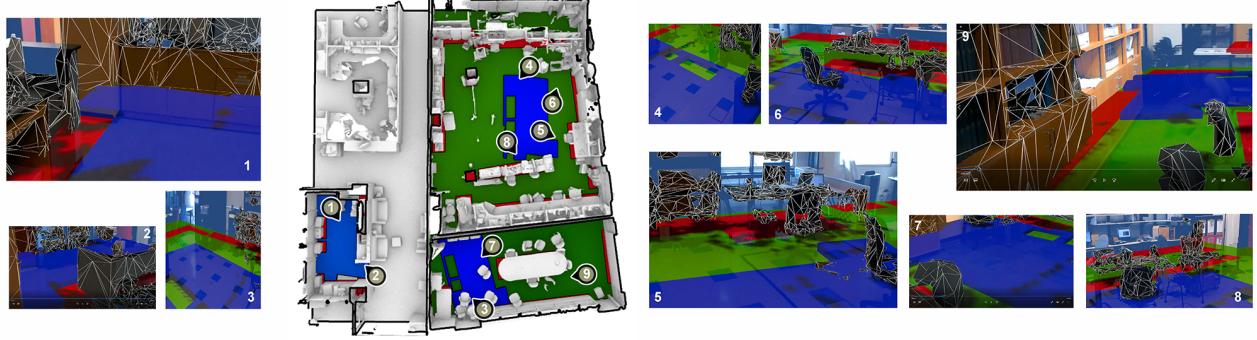
Figure 6: Screenshots from HoloLens illustrating the identified mutual boundaries as augmented overlays for three rooms: A) kitchen; B) conference room; C) robotic laboratory. Blue color indicates mutual boundaries, green color indicates standable spaces and red color indicates non-standable spaces.

perimeter of the room, while some larger equipment and a few tables are disorderly positioned around the central section of the lab (iii) a kitchen space, where surrounding appliances and cabinets are present in the scene.

After the initial scan of the surrounding environment by the user of each room, the geometrical mesh data is sent to a central server for processing. This process happens in an offline manner, as the current Hololens hardware is incapable of processing the computations that our solution would require. In addition, we scan the space using a Matterport camera, and perform the semantic segmentation step using Matterport classifications to locate the bounding boxes of all the furniture located in the room. We then feed the bounding box data to our algorithm for mutual boundary search. The implementation outputs spatial coordinates for standable and sittable areas which are automatically updated in the Unity Game Engine to be rendered in the Hololenses.

Figure 6 shows how the spatial boundary properties are visualized within the Hololens AR experience. The red spaces indicate non standable objects, the green spaces indicate standable boundaries, and the blue spaces indicate mutual boundaries that are accessible between all users. The visualized boundaries are positioned slightly above the floor level, allowing users to identify the mutual accessible ground between their local surrounding and the remote participant's spatial constraints.

Visualizing the mutual ground within the space itself using HoloLens allows us to understand how complex the problem can be when executed in a manual fashion. Some corner spaces that are not typically used as default social areas of an certain room, may become the only required common ground for interaction with other rooms. Overcoming this spatial bias is easily executed within the algorithm; meanwhile, this may not happen so easily and instantly when individuals are left to deal with it on their own.

However, due to the limited field of view of the HoloLens, detecting non-physical boundaries placed at a lower visual height becomes difficult to follow. This issue proved more challenging when walking closer to the non-orthogonal edges of mutual bounding area, where an individual could easily step outside the designated area. The shareable area also included a number of voids, which resulted on an inconsistent walking path inside the standable spaces. Moreover, the accuracy of the real-time mesh reconstruction in HoloLens played a critical role in calculating the required rendering occlusions for the visualized boundaries. This was mainly because the position of the the visualization was reflected close to the floor with many object placed over it, therefore failing to detect occluding objects, a fact that often misled the user in identifying whether the space was mutually accessible or not.

7 CONCLUSIONS

We introduce a novel optimization and manipulation framework to generate an optimal common virtual space for interactions that mostly involve standing and sitting. Our framework further recommends movement of surrounding furniture objects that can expand the size of the mutual space with minimal physical effort. We integrated our framework with a Strength Pareto Evolutionary Algorithm for an efficient search and optimization process. The multicriteria optimization process was able to generate a well-defined Pareto front of trade-offs between maximizing mutual space and minimizing physical effort. The Pareto front is more densely populated in some sections of the frontier than others, clearly identifying the best trade-offs region and the on-start of diminishing returns.

Furthermore, we experimented how the output solutions can be visualized using a HoloLens application. Results show that the proposed framework can effectively discover optimal shareable space for multi-user virtual interaction and thus provides better user experience compared to manually labeling shareable space, which would be a labor-intensive and imprecise workflow. In such context, if all participants stand within the calculated mutual spatial boundaries, the line of sight between all participants will be deterministic. In addition, no remote participant will be positioned in a conflicting location for any local user and would comply to the spatial constraints for all other participants.

There are, of course, limitations to the work. First, furniture with fixed positions are not automatically detected in our current implementation. We believe such feature can be integrated with further improvements in semantic segmentation methodologies, or can be optionally specified by the user whether an object is fixed or not. In addition, the furniture weight is calculated based on standard assumptions. We envision that with the growth of spatial computing procedures, such meta-data of the surrounding environment will be customizable by the user itself and can be loaded upon each mutual spatial search execution. Future work can comprise of integrating robust floorplanning representations with the current search mechanism to minimize computation cost and complexity. Lastly, usability studies can be conducted on how to improve the visualization strategies so participants can experience the required tele-communication functionalities while preserving the mutual spatial ground.

ACKNOWLEDGMENTS

REFERENCES

[1] S. Abdinnour-Helm and S. W. Hadley. Tabu search based heuristics for multi-floor facility layout. *International Journal of Production Research*, 38(2):365–383, 2000.

[2] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1534–1543, 2016. doi: 10.1109/CVPR.2016.170

[3] T. Balogh and P. T. Kovács. Real-time 3d light field transmission. In *Real-Time Image and Video Processing 2010*, vol. 7724, p. 772406. International Society for Optics and Photonics, 2010.

[4] E. Bardram. Activity-based computing: support for mobility and collaboration in ubiquitous computing. *Personal and Ubiquitous Computing*, 9(5):312–322, 2005.

[5] S. Beck, A. Kunert, A. Kulik, and B. Froehlich. Immersive group-to-group telepresence. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):616–625, 2013. doi: 10.1109/TVCG.2013.33

[6] S. Beck, A. Kunert, A. Kulik, and B. Froehlich. Immersive group-to-group telepresence. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):616–625, 2013.

[7] H. Benko, R. Jota, and A. Wilson. Miragetable: freehand interaction on a projected augmented reality tabletop. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 199–208. ACM, 2012.

[8] P.-A. Blanche, A. Bablumian, R. Voorakaranam, C. Christenson, W. Lin, T. Gu, D. Flores, P. Wang, W.-Y. Hsieh, M. Kathaperumal, et al. Holographic three-dimensional telepresence using large-area photorefractive polymer. *Nature*, 468(7320):80, 2010.

[9] B. Bolte and M. Lappe. Subliminal reorientation and repositioning in immersive virtual environments using saccadic suppression. *IEEE transactions on visualization and computer graphics*, 21(4):545–552, 2015.

[10] G. Bruder, P. Lubos, and F. Steinicke. Cognitive resource demands of redirected walking. *IEEE transactions on visualization and computer graphics*, 21(4):539–544, 2015.

[11] G. Bruder, F. Steinicke, P. Wieland, and M. Lappe. Tuning self-motion perception in virtual reality with visual illusions. *IEEE Transactions on Visualization and Computer Graphics*, 18(7):1068–1078, 2011.

[12] L. Caldas and M. Keshavarzi. Design immersion and virtual presence. *Technology| Architecture+ Design*, 3(2):249–251, 2019.

[13] A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niebner, M. Savva, S. Song, A. Zeng, and Y. Zhang. Matterport3D: Learning from RGB-D data in indoor environments. In *Proceedings - 2017 International Conference on 3D Vision, 3DV 2017*, pp. 667–676, 2018. doi: 10.1109/3DV.2017.00081

[14] Y.-C. Chang, Y.-W. Chang, G.-M. Wu, and S.-W. Wu. B*-trees: a new representation for non-slicing floorplans. In *Proceedings 37th Design Automation Conference*, p. 458–463, 2000. doi: 10.1109/DAC.2000.855354

[15] G. Chen, W. Guo, H. Cheng, X. Fen, and X. Fang. Vlsi floorplanning based on particle swarm optimization. In *2008 3rd International Conference on Intelligent System and Knowledge Engineering*, vol. 1, pp. 1020–1025. IEEE, 2008.

[16] M. Y. Cheng. Automated site layout of temporary construction facilities using-enhanced geographic information systems (gis). *Ph. D. Disst., Depart. of Civil Engineering, University of Texas at Austin, Texas, USA*, 1992.

[17] H. H. Clark, S. E. Brennan, et al. Grounding in communication. *Perspectives on socially shared cognition*, 13(1991):127–149, 1991.

[18] H. Eisenmann, and F. M. Johannes. Generic global placement and floorplanning. In *Proceedings of the 35th annual Design Automation Conference*, pp. 269–274. ACM, 1998.

[19] H. Fuchs, A. State, and J. C. Bazin. Immersive 3D telepresence. *Computer*, 47(7):46–52, 2014. doi: 10.1109/MC.2014.185

[20] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, et al. blue-c: a spatially immersive display and 3d video portal for telepresence. In *ACM Transactions on Graphics (TOG)*, vol. 22, pp. 819–827. ACM, 2003.

[21] P.-N. Guo, C.-K. Cheng, and T. Yoshimura. An o-tree representation of non-slicing floorplan and its applications. In *Proceedings 1999 Design Automation Conference (Cat. No. 99CH36361)*, p. 268–273. IEEE, 2003. doi: 10.1145/309847.309928

[22] B.-H. Gwee and M.-H. Lim. A ga with heuristic-based decoder for ic floorplanning. *INTEGRATION, the VLSI journal*, 28(2):157–172, 1999.

[23] V. Interrante, B. Ries, and L. Anderson. Seven league boots: A new metaphor for augmented locomotion through moderately large scale immersive virtual environments. In *2007 IEEE Symposium on 3D User interfaces*. IEEE, 2007.

[24] J. H. Jo and J. S. Gero. Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering*, 12(3):149–162, Jul 1998. doi: 10.1016/S0954-1810(97)00037-X

[25] A. Jones, M. Lang, G. Fyffe, X. Yu, J. Busch, I. McDowall, M. Bolas, and P. Debevec. Achieving eye contact in a one-to-many 3d video teleconferencing system. In *ACM Transactions on Graphics (TOG)*, vol. 28, p. 64. ACM, 2009.

[26] B. Jones, R. Sodhi, M. Murdock, R. Mehra, H. Benko, A. Wilson, E. Ofek, B. MacIntyre, N. Raghuvanshi, and L. Shapira. Roomalive: magical experiences enabled by scalable, adaptive projector-camera units. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pp. 637–644. ACM, 2014.

[27] A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu. *VLSI physical design: from graph partitioning to timing closure*. Springer Science & Business Media, 2011.

[28] P. Kaur. An enhanced algorithm for floorplan design using hybrid ant colony and particle swarm optimization. *Int. J. Res. Appl. Sci. Eng. Technol*, 2:473–477, 2014.

[29] M. Keshavarzi, M. Wu, M. N. Chin, R. N. Chin, and A. Y. Yang. Affordance analysis of virtual and augmented reality mediated communication. *arXiv preprint arXiv:1904.04723*, 2019.

[30] K. Kim, J. Bolton, A. Girouard, J. Cooperstock, and R. Vertegaal. Telehuman: effects of 3d perspective on gaze and pose estimation with a life-size cylindrical telepresence pod. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2531–2540. ACM, 2012.

[31] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

[32] K. Kiyota and K. Fujiyoshi. Simulated annealing search through general structure floor plans using sequence-pair. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, 88(6):28–38, 2005.

[33] G. Kurillo, R. Bajcsy, K. Nahrsted, and O. Kreylos. Immersive 3d environment for remote collaboration and training of physical activities. In *2008 IEEE Virtual Reality Conference*, pp. 269–270. IEEE, 2008.

[34] C. Kuster, N. Ranieri, Agustina, H. Zimmer, J. C. Bazin, C. Sun, T. Popa, and M. Gross. Towards next generation 3D teleconferencing systems. In *3DTV-Conference*, number Section 2, pp. 1–4. IEEE, oct 2012. doi: 10.1109/3DTV.2012.6365454

[35] V. Lang, W. Liang, and L.-F. Yu. Virtual agent positioning driven by scene semantics in mixed reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 767–775. IEEE, 2019.

[36] B. P. H. Lee. Mutual knowledge, background knowledge and shared beliefs: Their roles in establishing common ground. *Journal of Pragmatics*, 33(1):21–44, 2001. doi: 10.1016/S0378-2166(99)00128-9

[37] N. H. Lehment, D. Merget, and G. Rigoll. Creating automatically aligned consensus realities for AR videoconferencing. *ISMAR 2014 - IEEE International Symposium on Mixed and Augmented Reality - Science and Technology 2014, Proceedings*, (September):201–206, 2014.

[38] C.-T. Lin, D.-S. Chen, and Y.-W. Wang. An efficient genetic algorithm for slicing floorplan area optimization. In *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No. 02CH37353)*, vol. 2, pp. II–II. IEEE, 2002.

[39] J.-M. Lin and Y.-W. Chang. Tcg: A transitive closure graph-based representation for general floorplans. *IEEE transactions on very large scale integration (VLSI) systems*, 13(2):288–292, 2005.

[40] C. Liu, J. Wu, and Y. Furukawa. FloorNet: A Unified Framework for Floorplan Reconstruction from 3D Scans. pp. 1–18, 2018.

[41] S. Lombardi, J. Saragih, T. Simon, and Y. Sheikh. Deep appearance

models for face rendering. *ACM Trans. Graph.*, 37(4):68:1–68:13, July 2018. doi: 10.1145/3197517.3201401

[42] P. Luff and C. Heath. Mobility in collaboration. In *CSCW*, vol. 98, pp. 305–314, 1998.

[43] Y. Ma, S. Dong, X. Hong, Y. Cai, C.-K. Cheng, and J. Gu. Vlsi floorplanning with boundary constraints based on corner block list. In *Proceedings of the 2001 Asia and South Pacific Design Automation Conference*, pp. 509–514. ACM, 2001.

[44] A. Maimone and H. Fuchs. Encumbrance-free telepresence system with real-time 3d capture and display using commodity depth cameras. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pp. 137–146. IEEE, 2011.

[45] A. Maimone and H. Fuchs. Real-time volumetric 3d capture of room-sized scenes for telepresence. In *2012 3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, pp. 1–4. IEEE, 2012.

[46] A. Maimone, X. Yang, N. Dierk, A. State, M. Dou, and H. Fuchs. General-purpose telepresence with head-worn optical see-through displays and projector-based lighting. In *2013 IEEE Virtual Reality (VR)*, pp. 23–26. IEEE, 2013.

[47] W. Matusik and H. Pfister. 3d tv: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. In *ACM Transactions on Graphics (TOG)*, vol. 23, pp. 814–824. ACM, 2004.

[48] D. J. Moni and S. Arumugam. Vlsi floorplanning based on hybrid particle swarm optimization. *Karunya Journal of Research*, 1(1):111–121, 2009.

[49] K. Nagano, A. Jones, J. Liu, J. Busch, X. Yu, M. Bolas, and P. Debevec. An autostereoscopic projector array optimized for 3d facial display. In *ACM SIGGRAPH 2013 Emerging Technologies*, p. 3. ACM, 2013.

[50] S. Nakatake, K. Fujiyoshi, H. Murata, and Y. Kajitani. Module placement on bsg-structure and ic layout applications. In *Proceedings of the 1996 IEEE/ACM international conference on Computer-aided design*, pp. 484–491. IEEE Computer Society, 1997.

[51] S. Nakaya, T. Koide, and S. Wakabayashi. An adaptive genetic algorithm for vlsi floorplanning based on sequence-pair. In *2000 IEEE International Symposium on Circuits and Systems. Emerging Technologies for the 21st Century. Proceedings (IEEE Cat No. 00CH36353)*, vol. 3, pp. 65–68. IEEE, 2000.

[52] S. Narang, A. Best, and D. Manocha. Simulating movement interactions between avatars & agents in virtual worlds using human motion constraints. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 9–16. IEEE, 2018.

[53] N. C. Nilsson, T. Peck, G. Bruder, E. Hodgson, S. Serafin, M. Whitton, F. Steinicke, and E. S. Rosenberg. 15 Years of Research on Redirected Walking in Immersive Virtual Environments. *IEEE Computer Graphics and Applications*, 38(2):44–56, 2018. doi: 10.1109/MCG.2018.111125628

[54] S. Orts-Escolano, M. Dou, V. Tankovich, C. Loop, Q. Cai, P. A. Chou, S. Mennicken, J. Valentin, V. Pradeep, S. Wang, S. B. Kang, C. Rhemann, P. Kohli, Y. Lutchyn, C. Keskin, S. Izadi, S. Fanello, W. Chang, A. Kowdle, Y. Degtyarev, D. Kim, P. L. Davidson, and S. Khamis. Holoportation. pp. 741–754, 2017. doi: 10.1145/2984511.2984517

[55] H. M. Osman, M. E. Georgy, and M. E. Ibrahim. A hybrid cad-based construction site layout planning system using genetic algorithms. *Automation in Construction*, 12(6):749–764, 2003. doi: 10.1016/S0926-5805(03)00058-X

[56] T. C. Peck, H. Fuchs, and M. C. Whitton. An evaluation of navigational ability comparing redirected free exploration with distractors to walking-in-place and joystick locomotio interfaces. In *2011 IEEE Virtual Reality Conference*, pp. 55–62. IEEE, 2011.

[57] T. Pejsa, J. Kantor, H. Benko, E. Ofek, and A. Wilson. Room2room: Enabling life-size telepresence in a projected augmented reality environment. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, CSCW '16, pp. 1716–1725. ACM, New York, NY, USA, 2016. doi: 10.1145/2818048.2819965

[58] C. R. Qi, H. Su, M. Niessner, A. Dai, M. Yan, and L. J. Guibas. Volumetric and Multi-View CNNs for Object Classification on 3D Data. 2016. doi: 10.1109/CVPR.2016.609

[59] C. G. Ramsey. *Architectural graphic standards*. John Wiley & Sons, 2007.

[60] S. Razzaque, Z. Kohn, and M. C. Whitton. Redirected Walking. *Proceedings of EUROGRAPHICS*, pp. 289–294, 2001.

[61] M. Rebaudengo and M. S. Reorda. Gallo: A genetic algorithm for floorplan area optimization. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 15(8):943–951, 1996.

[62] B. Sowmya and M. Sunil. Minimization of floorplanning area and wire length interconnection using particle swarm optimization. *International Journal of Emerging Technology and Advanced Engineering*, 3(8), 2013.

[63] E. A. Suma, S. Clark, D. Krum, S. Finkelstein, M. Bolas, and Z. Warte. Leveraging change blindness for redirection in virtual environments. In *Proceedings - IEEE Virtual Reality*, pp. 159–166, 2011. doi: 10.1109/VR.2011.5759455

[64] E. A. Suma, Z. Lipps, S. Finkelstein, D. M. Krum, and M. Bolas. Impossible spaces: Maximizing natural walking in virtual environments with self-overlapping architecture. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):555–564, 2012. doi: 10.1109/TVCG.2012.47

[65] T.-Y. Sun, S.-T. Hsieh, H.-M. Wang, and C.-W. Lin. Floorplanning based on particle swarm optimization. In *IEEE Computer Society Annual Symposium on Emerging VLSI Technologies and Architectures (ISVLSI'06)*, pp. 5–pp. IEEE, 2006.

[66] T. Tanikawa, Y. Suzuki, K. Hirota, and M. Hirose. Real world video avatar: real-time and real-size transmission and presentation of human figure. In *Proceedings of the 2005 international conference on Augmented tele-existence*, pp. 112–118. ACM, 2005.

[67] I. D. Tommelein, R. E. Levitt, B. Hayes-Roth, and T. Confrey. Sightplan experiments: alternate strategies for site layout design. *Computing in Civil Engineering*, 5(1):42–63, 1991. doi: 10.1007/BF01927759

[68] H. Towles, W.-C. Chen, R. Yang, S.-U. Kum, , H. Fuchs, N. Kelshikar, J. Mulligan, K. Daniilidis, L. Holden, B. Zeleznik, A. Sadagic, and J. Lanier. 3d tele-collaboration over internet2. In *In: International Workshop on Immersive Telepresence, Juan Les Pins*. Citeseer, 2002.

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

[70] L.-T. Wang, Y.-W. Chang, and K.-T. T. Cheng, eds. *Electronic Design Automation: Synthesis, Verification, and Test*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2009.

[71] S.-E. Wei, J. Saragih, T. Simon, A. W. Harley, S. Lombardi, M. Perdoch, A. Hypes, D. Wang, H. Badino, and Y. Sheikh. Vr facial animation via multiview image translation. *ACM Transactions on Graphics (TOG)*, 38(4):67, 2019.

[72] W.-C. Wen, H. Towles, L. Nyland, G. Welch, and H. Fuchs. Toward a compelling sensation of telepresence: Demonstrating a portal to a distant (static) office. In *Proceedings Visualization 2000. VIS 2000 (Cat. No. 00CH37145)*, pp. 327–333. IEEE, 2000.

[73] B. Williams, G. Narasimham, B. Rump, T. P. McNamara, T. H. Carr, J. Rieser, and B. Bodenheimer. Exploring large virtual environments with an hmd when physical space is limited. In *Proceedings of the 4th symposium on Applied perception in graphics and visualization*, pp. 41–48, 2007.

[74] D. F. Wong and C. L. Liu. A New Algorithm for Floorplan Design. In *Proceedings of the 23rd ACM/IEEE Design Automation Conference*, DAC '86, pp. 101–107. IEEE Press, Piscataway, NJ, USA, 1986.

[75] W. Xiaogang et al. Vlsi floorplanning method based on genetic algorithms [j]. *Microprocessors*, (1):1, 2002.

[76] C. Zhang, Q. Cai, P. A. Chou, Z. Zhang, and R. Martin-Brualla. Viewport: A distributed, immersive teleconferencing system with infrared dot pattern. *IEEE MultiMedia*, 20(1):17–27, 2013.

[77] E. Zitzler, M. Laumanns, and L. Thiele. Spea2: Improving the strength pareto evolutionary algorithm. *TIK-report*, 103, 2001.