Creating Automatically Aligned Consensus Realities for AR Videoconferencing

Nicolas H. Lehment* Daniel Merget† Gerhard Rigoll‡

Institute for Human-Machine-Communication
Technische Universität München

Figure 1: Illustration of the basic idea of bilateral AR telepresence. Both participants (shown in purple) see their opposite conversation partner (shown in blue) overlaid as 3D presences onto their own immediate surroundings. The avatars are integrated as full-sized virtual objects and mirror the movement, gesturing etc. of the respective participant. As the remote user moves about, the avatar might appear to move through local obstacles.

Abstract

This paper presents an AR videoconferencing approach merging two remote rooms into a shared workspace. Such bilateral AR telepresence inherently suffers from breaks in immersion stemming from the different physical layouts of participating spaces. As a remedy, we develop an automatic alignment scheme which ensures that participants share a maximum of common features in their physical surroundings. The system optimizes alignment with regard to initial user position, free shared floor space, camera positioning and other factors. Thus we can reduce discrepancies between different room and furniture layouts without actually modifying the rooms themselves. A description and discussion of our alignment scheme is given along with an exemplary implementation on real-world datasets.

Index Terms: H.4.3 [Information Systems Applications]: Communications Applications—Computer Conferencing, Teleconferencing, and Videoconferencing;

1 Introduction

To date, videoconferencing and telepresence are mostly limited to flat displays or unwieldy, immersive setups. However, recent work by Maimone et al. [14] shows how our conversation partners can be augmented directly into our surroundings. This leads us to the question of how to define the common environment in which we hold such conversations. Current work in augmented reality (AR) videoconferencing can be split into two categories: Remote support scenarios [1, 6, 21, 17] and conversation scenarios [2, 14, 11]. The remote support scenarios usually immerse the remote advisor in a local scene while the AR component is intended to communicate annotations or markers to the local user. The surroundings of the remote advisor are disregarded and not communicated to the local user. Meanwhile, previous research in conversation scenarios tends to focus on the accurate display of users in empty rooms or resorts to "window" analogies [12, 11] in order to avoid conflicts between heterogeneous surroundings. Previous work in immersive telepresence avoided the problem entirely by replacing the real surroundings with a virtual consensus reality, as in [5, 10].

Recent advances in AR display technology [13] and affordable 3D scanning solutions [19, 16] suggest a new option: Bilateral telepresence between two participants where both rooms are merged into a common consensus reality. In this scenario, the remote user is rendered into the local user's office as an avatar. Conversely, the local user appears as an avatar in the remote office as well. This mutual telepresence is especially well suited for conversations including non-verbal communication, e.g., posture, gesturing etc.

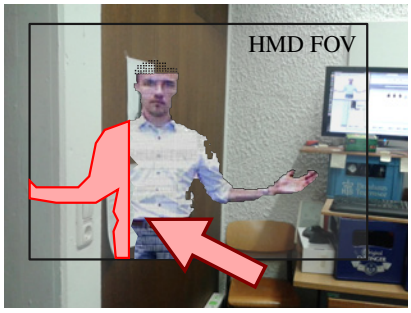

Figure 2: Bad alignment leading to conversation partner placed within wall. Affected part of pointcloud overlaid in red.

It is important to note that participating spaces usually do not have the same layout. A typical conflict might appear as follows: We stand in front of a large table in a conference room while our conversation partner's surroundings are far more cramped. As we

*e-mail: Lehment@tum.de
†e-mail: Daniel.Merget@tum.de
‡e-mail: Rigoll@tum.de

walk through our spacious room, our remote avatar appears to walk right into her cubicle wall, shattering the illusion of co-presence. Fig. 2 shows such a conflict.

In order to deal with such discrepancies in our environments, we aim to create a reality-based consensus space which uses 3D scans of both rooms to identify common layout features. In an initial optimization stage, the rooms are aligned for minimal discrepancies. The data generated during the alignment can then be used for visualizing remote obstacles to the local user in the course of the meeting.

2 System Overview

The entire bilateral AR telepresence system must fulfill four fundamental tasks: It must generate the consensus reality from room models, display the remote user and possible boundaries in the consensus reality, simultaneously exchange avatar data of the users and finally manage any interaction with additional AR content. The display of the remote users relates to an already very active field of ongoing research [14, 13]. Transmitting the avatars can be achieved either by puppeteering a pre-built avatar with pose tracking data [20, 24] or by streaming a live point cloud [8]. Interaction with AR content is an extensively studied topic [3, 18]. Since each task easily warrants its own paper, we focus this contribution on the generation of a consensus reality and start by formulating the requirements.

As the scenario is focussed on conversation, we want participants to start the meeting facing each other. The common free space should be as large as possible. There are usually a number of obstacles, such as tables, desks, walls, etc., so we must avoid initializing any user within such obstacles. Meetings might involve the display or manipulation of AR content, so there should be consensus surfaces on which these can be placed. A suitable surface might be a table present on both sides of the conversation. Finally, we can also consider the geometry of the rooms in order to achieve parallel fitting of walls.

Our solution utilizes pre-recorded models of the participating spaces. The scans are assumed to consist of triangulated meshes which cover each user's entire local workspace. Our approach is constrained to rooms with uninterrupted, level floors and available knowledge of users' positions and headings at the moment of initialization. The process of automatic model building for indoor rooms is already covered extensively in previous literature [22, 19, 16], with a number of commercial and free systems available.

Each room is observed by three to four RGB-D cameras (Kinect and BumblebeeXB3). The controller of each camera acts as an independent server, providing a compressed RGB-D image stream of the local user as well as intrinsic and extrinsic calibration data. In order to exclude static background and furniture, the PBAS foreground segmentation provided by Hofmann et al. [7] is used. For compression, the *FFMPEG* library is used: RGB data is sent as a stream of JPG images, while depth data is transmitted using the lossless PNG format, yielding a typical frame size of around 15 kBytes per camera (640×480 resolution). This data is captured by the conversation partner's session managment engine, translated into 3D point clouds and then rendered into a composite representation of the remote user using the calibration data. Other than prebuilt avatars, the use of point clouds allows for the communication of lip movement and complex gestures [14]. The composite point cloud avatar is then augmented into the view of the scene, e.g, using a video-see-through HMD as shown in Fig. 5. The existing room models inform the occlusion handling for the point clouds.

3 Computing the Consensus Reality

Before we can begin to formulate an alignment scheme for the users' rooms, we have to define a metric for quantifying alignment. In a first step, the consensus reality is segmented into five types of spaces. Each space is treated as a separate map ($\mathbf{M}$). Most interactions take place in space which is unoccupied for both user "A" and user "B" ($\mathbf{M}_{\mathrm{F}}$). Other interactions are centered around common work surfaces ($\mathbf{M}_{\mathrm{S}}$), such as tables of similar height. There are also spaces which are blocked by physical objects for both users ($\mathbf{M}_{\mathrm{CO}}$). Finally, there are obstacles which are present in only one of the users' environments ($\mathbf{M}_{\mathrm{AO}}, \mathbf{M}_{\mathrm{BO}}$). All these spaces are computed from the pre-recorded triangular mesh models of each room and the relative positions of their floor plane origins $\omega = \{x, y, \theta_z\}$.

In order to reduce computational complexity, we project both 3D meshes orthographically into their own floor plane, resulting in two 2D maps of maximum vertex heights relative to the floor ($\mathbf{M}_A, \mathbf{M}_B$). Thus a $10m \times 10m$ sampling grid with 50 $^{\text{pixel}}/_{m}$ would correspond to two 500×500 arrays of floating point values. Common image processing techniques can be applied to these maps. Space behind walls is set to a fixed maximum value (e.g., $3m$), the floor planes have zero height. Two exemplary maps can be seen in Fig. 3, with the zero-height floor surfaces colored for easier recognition.

The advantage of reducing to a 2D map becomes apparent for complex room scans: Regardless of the number of vertices in the meshes, after these are mapped to the floor plane the subsequent optimization will run only on the fixed-size maps. Therefore optimization with complex meshes would require more operations in the initial mapping stage, but not in the repeated optimization steps afterwards.

Following the mapping, the floorplan $\mathbf{M}_B$ of room B is translated and rotated to $\mathbf{M}_B^{\omega}$ using ω and overlaid over $\mathbf{M}_A$ of room A. As illustrated in Fig. 3, we can now compute the free space $\mathbf{M}_{\mathrm{F}}(\mathbf{p}, \omega)$ for a given pose ω, processing each discrete map pixel $\mathbf{p} = (x, y)$ in parallel. We assume every object less than $c_{\text{floor}} = 0.1\ m$ tall to be part of the floor plane.

$$\mathbf{M}_{\mathrm{F}}(\mathbf{p}, \omega) = \begin{cases} 1 & \text{if} \quad \max(\mathbf{M}_A(\mathbf{p}), \mathbf{M}_B^{\omega}(\mathbf{p})) < c_{\text{floor}} \\ 0 & \text{else} \end{cases} \tag{1}$$

Two boolean operations yield maps of unilateral obstacles for the rooms A and B.

$$\mathbf{M}_{\mathrm{AO}}(\mathbf{p}, \omega) = (\neg \mathbf{M}_B^{\omega}(\mathbf{p})) \wedge \mathbf{M}_A(\mathbf{p}) \tag{2}$$

$$\mathbf{M}_{\mathrm{BO}}(\mathbf{p}, \omega) = (\neg \mathbf{M}_A(\mathbf{p})) \wedge \mathbf{M}_B^{\omega}(\mathbf{p}) \tag{3}$$

The consensus obstacles are found likewise by a simple AND operation on the two maps:

$$\mathbf{M}_{\mathrm{CO}}(\mathbf{p}, \omega) = \mathbf{M}_A(\mathbf{p}) \wedge \mathbf{M}_B^{\omega}(\mathbf{p}) \tag{4}$$

Analysing the difference in height on the consensus obstacles provides us with a map of consensus work surfaces. We assume an acceptable height difference of $c_{\text{diff}} = 10\ cm$ and maximum height of $c_{\max} = 1.5m$:

$$\mathbf{M}_{\mathrm{S}}(\mathbf{p}, \omega) = \begin{cases} \mathbf{M}_A(\mathbf{p}) & \text{if } \left\| \mathbf{M}_A(\mathbf{p}) - \mathbf{M}_B^{\omega}(\mathbf{p}) \right\| < c_{\text{diff}} \\ & \wedge \max\left(\mathbf{M}_A(\mathbf{p}), \mathbf{M}_B(\mathbf{p})\right) < c_{\max} \\ 0 & \text{else} \end{cases} \tag{5}$$

Besides the segmentation into these five types of space, the observability of the users is cruical for the AR videoconference. For an optimal alignment, both users should be observed by at least one camera throughout the common meeting space. Therefore a set of observability maps is computed for each user's room. As each camera broadcasts its calibration data, their view cones can be mapped to the floor planes for both rooms. Thus, the maps $\mathbf{M}_{\text{view}}^{\mathrm{A}}$ and $\mathbf{M}_{\text{view}}^{\mathrm{B}}$ show the number of cameras observing each spot in room A and B respectively. Observability score maps are then computed by adding the number of cameras for each floor element and dividing

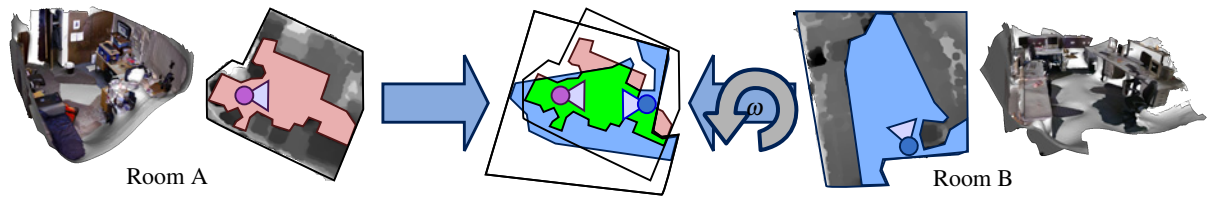

Figure 3: Basic approach for consensus space computation: Two rooms are mapped to the 2D floor plane, users' positions (shown in blue and purple) are remapped accordingly. The transformation ω is applied to the room B. Both maps are then overlaid and the energy terms are computed. For instance, the area mapped in green shows common free floorspace for a given pose used for computing $E_{\text{free}}(\omega)$.

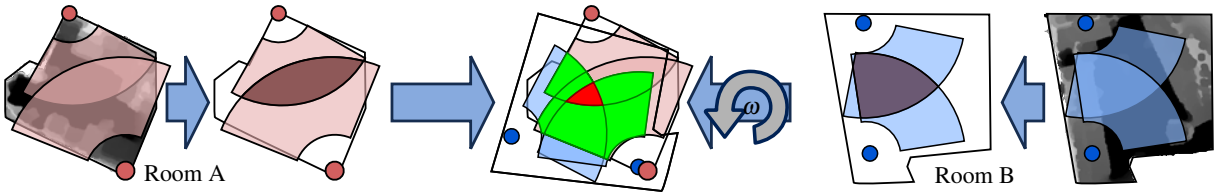

Figure 4: Exemplary calculation of mutual observability score $\mathbf{M}_{\text{OBS}}(\omega)$ for two scenes with two cameras each. Only the region marked in red permits user tracking and recording by two cameras in both rooms. Regions marked in green are visible by at least one camera in each scene. The red region is visible by both cameras in each scene, providing best observability

by the number $N_{\text{view}}^{\max}$ of desired views. For room A, this is expressed as follows:

$$\mathbf{M}_{\text{OBS}}^{\text{A}}(\mathbf{p}) = \begin{cases} \mathbf{M}_{\text{view}}^{\text{A}}(\mathbf{p})/N_{\text{view}}^{\max} & \text{if } \mathbf{M}_{\text{view}}^{\text{A}}(\mathbf{p}) < N_{\text{view}}^{\max} \\ 1 & \text{else} \end{cases} \tag{6}$$

For a floor section to be mutually observable, there must be at least one camera in room A and one camera in room B providing observations of a user standing in this section. The mutual observability score $\mathbf{M}_{\text{OBS}}$ is computed using the partial observability score maps and a given alignment pose ω. The process is visualized in Fig. 4. We can then combine the observability score map with the free space map to gain the observable free space map $\mathbf{M}_{\text{FOBS}}$:

$$\mathbf{M}_{\text{OBS}}(\mathbf{p},\omega) = \min\left(\mathbf{M}_{\text{OBS}}^{\text{A}}(\mathbf{p}), \mathbf{M}_{\text{OBS}}^{\text{B}}(\mathbf{p},\omega)\right) \tag{7}$$

$$\mathbf{M}_{\text{FOBS}}(\mathbf{p},\omega) = \mathbf{M}_{\text{OBS}}(\mathbf{p},\omega) \times \mathbf{M}_{\text{F}}(\mathbf{p},\omega) \tag{8}$$

So far we have described how to compute the different maps for a given pose ω. Based on these maps, an optimization stage computes the optimal alignment of both participating rooms. The alignment problem is formulated as an energy maximization problem, where the partial energy terms correspond to the requirements outlined in Sec. 2.

The binary term $\alpha_E(\omega) = \{0,1\}$ helps in avoiding undesired configurations and dominates the energy function. It is set to "0" for poses where one user is initialized within an obstacle, no line of sight is possible or other essential problems arise. Thus we avoid having a single strong partial term promoting an otherwise unfavorable configuration. The term serves as an abortion flag during optimization, further reducing computational overhead.

The term $E_{\text{free}}(\omega)$ ensures a maximum shared floor space. We discard inaccessible regions of the observable free space floorplan $\mathbf{M}_{\text{FOBS}}(\omega)$ by applying a morphological erosion operation with a circular kernel ($\varnothing\, 1\,m \approx$ twice avg. male shoulder width), resulting in the map $\mathbf{M}_{\text{FOBS}}^{\text{erode}}(\omega)$. In addition, the free space must be mutually observable, i.e., on both sides there must be at least one camera which provides RGB-D data for this area. The alignment aims to find common regions in which each user is observed by as many cameras as possible by using the observability score $\mathbf{M}_{\text{FOBS}}^{\text{erode}}(\mathbf{p},\omega)$.

$$E_{\text{free}}(\omega) = \frac{1}{\|\mathbf{M}_{\text{FOBS}}^{\text{erode}}(\omega)\|} \sum_{\mathbf{p}} \mathbf{M}_{\text{FOBS}}^{\text{erode}}(\mathbf{p},\omega) \tag{9}$$

The term $E_{\text{prox}}(\omega)$ penalizes close initial user positions, as to prevent initializing users in the same space. In preparation, users' positions $\mathbf{X}_a$ and $\mathbf{X}_b^{\omega}$ are mapped to the 2D plane of room A. The decay factor λ_{prox} ensures that a proximity of $d(\omega) = d_{\text{fade}}$ between users returns a score of $E_{\text{prox}}(\omega) = 0.95$, i.e., 95% of the maximum

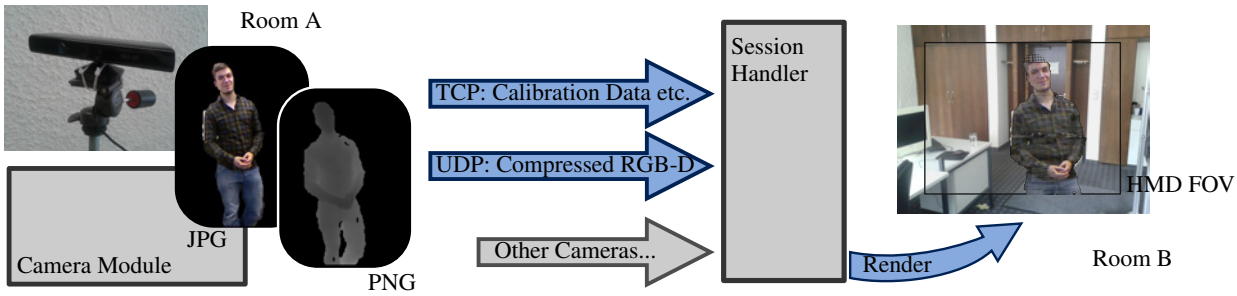

Figure 5: 3D point cloud transmission chain for a single camera module. The session handler receives data from several cameras and renders the remote user point cloud according to the perspective of the local user.

possible score for this term:

$$d(\omega) = \|\mathbf{X}_b^\omega - \mathbf{X}_a\|_{\text{euclid.}} \tag{10}$$

$$\lambda_{\text{prox}} = -\log(5\%)/(d_{\text{fade}} - d_{\text{min}})^2 \tag{11}$$

$$E_{\text{prox}}(\omega) = \begin{cases} 1 - exp\left(-\lambda_{\text{prox}} \cdot (d(\omega) - d_{\text{min}})^2\right) & \text{if } d(\omega) > d_{\text{min}} \\ 0 \quad \text{and} \quad \alpha_E(\omega) = 0 & \text{else} \end{cases} \tag{12}$$

Since users prefer to face each other at the moment of initialization, the term $E_{\text{head}}(\omega)$ increases for small angles between each user's heading and the connecting vector to the other conversation partner's position. The users' 2D headings relative to the floormap of room A are denoted as $\mathbf{D}_a$ and $\mathbf{D}_b^\omega$, the decay factor of the exponential is set to $\lambda_{\text{head}} = -\log(5\%)/(90^\circ)^2$. A relative angle of 90° therefore yields a partial score of $E^A_{\text{head}}(\omega) = 0.05$:

$$\theta_{A\to B}(\omega) = \cos^{-1}\left((\mathbf{X}_b^\omega - \mathbf{X}_a) \cdot \mathbf{D}_a\right) \tag{13}$$

$$E^A_{\text{head}}(\omega) = \exp\left(-\lambda_{\text{head}}\, \theta_{A\to B}(\omega)^2\right) \tag{14}$$

$$\theta_{B\to A}(\omega) = \cos^{-1}\left((\mathbf{X}_a - \mathbf{X}_b^\omega) \cdot \mathbf{D}_b^\omega\right) \tag{15}$$

$$E^B_{\text{head}}(\omega) = \exp\left(-\lambda_{\text{head}}\, \theta_{B\to A}(\omega)^2\right) \tag{16}$$

$$E_{\text{head}}(\omega) = 0.5 \cdot E^A_{\text{head}}(\omega) + 0.5 \cdot E^B_{\text{head}}(\omega) \tag{17}$$

The term $E_{\text{surf}}(\omega)$ increases for large common surfaces such as a table present in both rooms. The computation is nearly identical to $E_{\text{free}}(\omega)$, except that the map $\mathbf{M}_{\text{S}}$ is used instead of $\mathbf{M}^{\text{erode}}_{\text{FOBS}}$. For scenarios where a minimum surface size is desired, the term $E_{\text{mins}}(\omega)$ is set to "1" only if an sufficiently large, uninterrupted common surface A_{min} is available.

The energy terms discussed so far are utilitarian in nature. In order to promote visually pleasing alignments, the term $E_{\text{skew}}(\omega)$ returns high values for alignments in which walls and environment boundaries are aligned in parallel. In a first step, a probabilistic Hough transform [15] identifies major linear wall segments in room A. Their main angular directions are computed and collected in a histogram $\mathscr{H}^A_{\theta,\text{wall}}$. A function $f^i_{\text{peak}}(\theta)$ is assigned to each histogram peak θ_i, with wrap-around terms added to account for the $180^\circ \to 0^\circ$ discontinuity. We can then test parallel alignment by generating a second angular direction histogram $\mathscr{H}^B_{\theta,\text{wall}}$ for room B, applying the current rotation from ω and calculating the functions $f^i_{\text{peak}}(\theta)$ for each peak θ_B in $\mathscr{H}^B_{\theta,\text{wall}}$. $E_{\text{skew}}(\omega)$ is computed as the normalized sum of these peak terms using a standard skew deviation of $c_{\text{skew}} = 36^\circ$:

$$f^i_{\text{peak}}(\theta) = \exp\left(-0.5 \cdot \left(\frac{\theta_i - \theta}{c_{\text{skew}}}\right)^2\right) \tag{18}$$

$$E_{\text{skew}}(\omega) = \frac{1}{N^B_{\text{peaks}}} \sum_{\theta_B \in \mathscr{H}^B_{\theta,\text{wall}}} \max_i\left(f^i_{\text{peak}}(\theta_B)\right) \tag{19}$$

As the direction of alignment is the only parameter, the term only depends on the rotational component ω_θ of an alignment pose. Its influence on the total energy term is decidedly less pronounced for smaller and cluttered rooms, since the solution space is already heavily constrained. Our analysis of real-world rooms in Sec. 4 therefore focuses on the remaining terms.

We combine these partial energy terms into a single energy term by addition. In order to enforce limits on user placement, the previously introduced binary term α_E can abort further computation and forcibly return $E(\omega) = 0$. Thus we can avoid initializing users in remote walls or too close to each other. Additional weighting factors α_X allow reducing or disabling certain energy terms

$$\begin{aligned} E(\omega) = &\alpha_{\text{free}} E_{\text{free}}(\omega) + \alpha_{\text{prox}} E_{\text{prox}}(\omega) + \alpha_{\text{head}} E_{\text{head}}(\omega) \\ &+ \alpha_{\text{surf}} E_{\text{surf}}(\omega) + \alpha_{\text{mins}} E_{\text{mins}}(\omega) + \alpha_{\text{skew}} E_{\text{skew}}(\omega) \end{aligned} \tag{20}$$

The setting of weighting factors depends on the size of the participating spaces. In case of cramped spaces, there are few options for alignment without placing one user inside a wall or outside of observability. Large spaces on the other hand allow for more complex adjustments, as long as there is sufficient observability. Table 1 gives some typical settings for conversation scenarios in different types of space. In case of rooms of different size, the smaller room usually dictates the constraints on weighting factors.

Table 1: Typical weighting factors for conversation scenario

Room	α_{free}	α_{prox}	α_{head}	α_{surf}	α_{mins}	α_{skew}
$< 5m^2$	1.0	0.2	0	0.5	0	0
$< 10m^2$	1.0	0.5	0.5	0.5	0	0
$> 10m^2$	1.0	0.5	0.5	0.75	0.5	0.5

The resulting optimization problem is non-convex with a three-dimensional solution space for ω. For producing the illustrations in this paper a hierarchical brute-force solver is used, stepping over a fixed solution space range. Since any pose which sets α_E to zero will be discarded, our implementation progressively checks each knock-out condition as it computes the energy terms and discards an alignment as soon as $\alpha_E = 0$. As the computation of the consensus reality alignment is performed only once at the start of each teleconference, real-time performance is not relevant.

4 Application to Real Datasets

The suitability of our approach to automatically generated datasets was examined using a large office dataset provided by Steinbrücker et al. [22]. From this dataset, two rooms were isolated and remeshed using a screened Poisson approach [9] in order to close holes in the data. The floor plane origins and initial user positions were defined manually. As can be seen in Fig. 6, the resulting alignment provides a large common workspace and ensures that both users start facing each other.

In order to examine the effect of the different energy terms more closely, we created a second dataset for which two rooms at our institute were scanned. The "ReconstructMe" software was used together with a Kinect camera to create $2 \times 2 \times 2\,m^3$ partial scans. These were then stitched together, resurfaced using a screened Poisson approach [9] and downsampled to the desired resolution by quadric edge collapsing [4] (see Fig. 1 and 3). The origins are set to be approximately in the middle of the free floor space.

As Fig. 6 shows, the alignment scheme successfully finds a solution which places both users in a position about $1.5\,m$ apart, facing each other. Additionally, there is a consensus floor surface of approx. $2\,m \times 2\,m$ and a consensus work surface on the tables (which happen to be about the same height). The alignment was performed with active hierarchical optimization on a $5\,m \times 5\,m$ sampling grid at $30\ ^{\text{pixel}}/_{m}$.

In order to verify the decoupling between model complexity and processing time, the alignment was repeated for decreasing resolutions of both models. As shown in Tab. 2, the time t_{align} spent on aligning the maps is indeed independent from model complexity. All data was computed on an i7-3770 CPU with 16 GB RAM, using 3DVIA Studio for rendering and scene management.

It is interesting to note the influence of the binary term α_E on the final result. As shown in Fig. 7, disabling this term can lead to unsatisfactory alignment of the users' positions. High values for $E_{\text{free}}(\omega)$ and $E_{\text{surf}}(\omega)$ tend to dominate the energy function and lead to poses where users are placed too close together, within walls

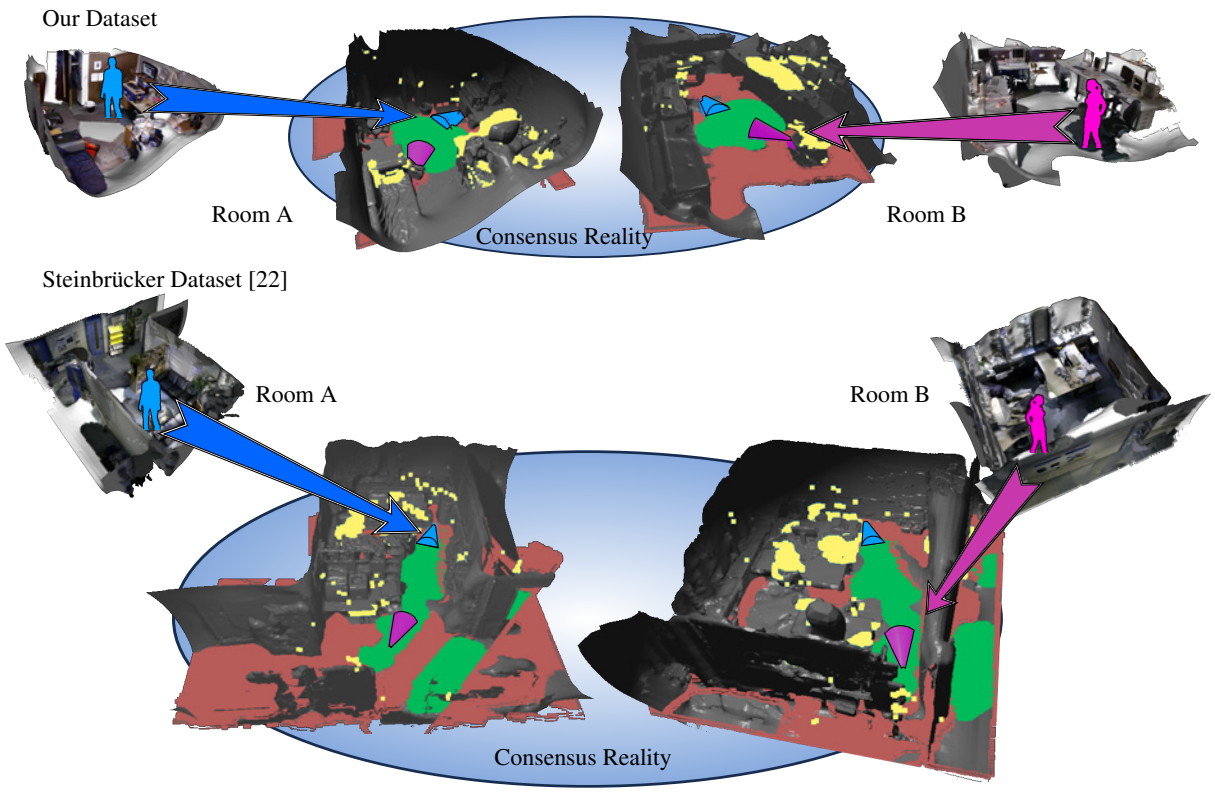

Figure 6: Alignment of two real rooms with users (blue and purple, represented as silhouettes and view-cones). Top row shows alignment with our own dataset, bottom row shows alignment using the dataset provided by Steinbrücker et al. [22]. Common free floor space shown in green, common work surfaces in yellow. Obstacles of Room B not present in Room A shown in light pink. Full coverage by 6 cameras per room.

Table 2: Processing time over model complexity

Faces per Room	$t_{\text{map A}}$	$t_{\text{map B}}$	t_{align}
100	1.28 ms	1.04 ms	9967.11 ms
1k	2.92 ms	2.52 ms	9989.55 ms
10k	17.27 ms	16.88 ms	10022.6 ms
100k	87.57 ms	73.20 ms	10192 ms
1M	690.16 ms	689.95 ms	9967.06 ms

or not facing each other. This is best illustrated by comparing the total energy terms $E(\omega)$ computed with and without the α_E term. As Fig. 8 illustrates, without the constraints the energy function develops a twisting tube shape due to the proximity and heading terms, overlaid by the free floor space and consensus surface terms. So even in cases where the proximity $E_{\text{prox}}(\omega)$ and heading terms $E_{\text{head}}(\omega)$ become minimal, there are high contributions by $E_{\text{free}}(\omega)$ and $E_{\text{surf}}(\omega)$. Meanwhile, the missing constraints allow for poses placing the users within obstacles. Once the constraints are enacted by setting α_E to zero for undesired poses, the total energy $E(\omega)$ is drastically simplified, as illegal poses lead to $E(\omega_{\text{illegal}}) = 0$.

Even when using the binary term to ensure compliant poses, there are initial conditions where our system cannot find an optimal solution. For example, if one of the users starts the initialization while facing a wall, the heading term $E_{\text{head}}(\omega)$ will come into conflict with the free floor space term $E_{\text{free}}(\omega)$. Future research should be dedicated to detecting and resolving such scenarios. A possible solution might be a gradual relaxation of alignment constraints or even non-rigid mapping of remote spaces, e.g., by adaption of *redirected walking* [23] for AR conferencing. For best results with our approach, both users should be facing a large free space during initialization.

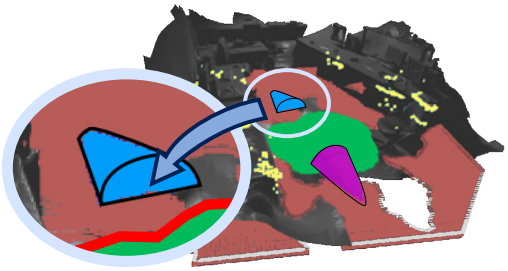

Figure 7: Example of illegal user placement seen from Room B after disabling hierachical optimization. Note the blue user position within the remote obstacle, marked red / light pink.

The adaption of the room alignment to different camera placements is shown in Fig. 9. In case of larger rooms, the inclusion of the mutual observability score allows the algorithm to center the alignment on floor space with best coverage by multiple cameras for both participants. In case of smaller rooms, there is usually less room for adaption. It should be noted that currently only the number of cameras pointed at a position is considered, without regard for the actual angle of observation or obstruction by furniture. Future work might thus aim to provide feedback to the users as they distribute cameras for optimal coverage of arbitrary rooms.

5 Conclusions & Outlook

In this paper the problem of heterogeneous user surroundings in bilateral AR videoconferencing was examined. We began by identifying the requirements on a bilateral telepresence scenario, ranking

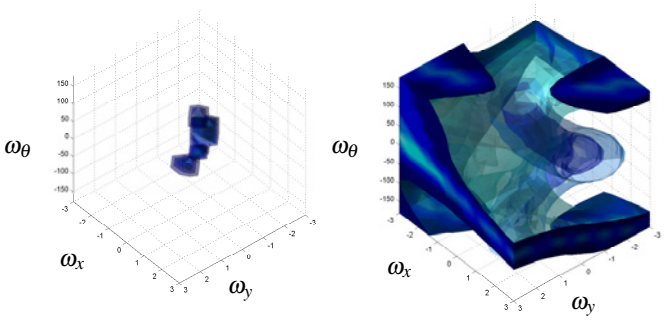

Figure 8: Comparison between total energy with hierarchical optimization enabled (left) and disabled (right). Lighter tones indicate higher values.

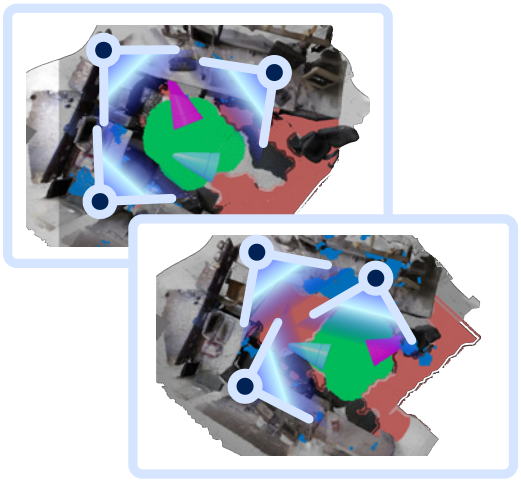

Figure 9: Adaption of room alignment to different camera placements. Consensus free space shown in green. Initial user heading restrictions were loosend for this example.

them by importance. These requirements were translated into an optimization scheme running on 2D maps of the two rooms. We went on to demonstrate an implementation of this optimized alignment using real-world scans.

This initial alignment scheme is applicable to any AR videoconferencing scenario where a mutual inclusion of both users' environments is desired. Theoretically, our algorithm should be extensible to more participating rooms once the brute-force solver used in this paper is substituted for a more efficient optimization algorithm. The aligned floor maps produced by our approach can also be rendered in order to communicate established boundaries to the conversation partners (e.g., walls). Rendering the floorplan into the AR view helps to avoid obstacles and in finding consensus work surfaces. However, the actual implementation of this rendering depends on the chosen display and is therefore outside the scope of this paper.

As first affordable consumer head mounted displays (HMDs) are appearing on the horizon, we can soon expect to see our avatars walk in far-away offices and visit relatives in distant cities. The approach presented here brings this vision closer to realization by finding the proverbial common ground between participants, while simultaneously preserving the proper etiquette of not walking through other people's walls.

References

[1] M. Adcock, S. Anderson, and B. Thomas. Remotefusion: Real time depth camera fusion for remote collaboration on physical tasks. In *SIGGRAPH VRCAI*, pages 235–242, 2013.

[2] M. Billinghurst, A. Cheok, S. Prince, and H. Kato. Real world teleconferencing. *Computer Graphics and Applications, IEEE*, 22:11 – 13, 2002.

[3] R. Budhiraja, G. Lee, and M. Billinghurst. Interaction techniques for HMD-HHD hybrid ar systems. In *ISMAR*, pages 243–244, 2013.

[4] M. Garland and P. S. Heckbert. Surface simplification using quadric error metrics. In *Conf. on Computer Graphics and Interactive Techniques*, pages 209–216, 1997.

[5] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, K. Strehlke, A. V. Moere, and O. Staadt. Blue-c: A spatially immersive display and 3d video portal for telepresence. In *SIGGRAPH*, pages 819–827, 2003.

[6] P. Gurevich, J. Lanir, B. Cohen, and R. Stone. Teleadvisor: A versatile augmented reality tool for remote assistance. In *SIGCHI*, pages 619–622, 2012.

[7] M. Hofmann, P. Tiefenbacher, and G. Rigoll. Background segmentation with feedback: The pixel-based adaptive segmenter. In *CVPR*, pages 38 –43, 2012.

[8] J. Kammerl, N. Blodow, R. Rusu, S. Gedikli, M. Beetz, and E. Steinbach. Real-time compression of point cloud streams. In *ICRA*, pages 778–785, 2012.

[9] M. Kazhdan and H. Hoppe. Screened poisson surface reconstruction. *ACM Trans. Graph.*, 32:29:1–29:13, 2013.

[10] G. Kurillo and R. Bajcsy. 3d teleimmersion for collaboration and interaction of geographically distributed users. *Virtual Reality*, 17(1):29–43, 2013.

[11] N. H. Lehment, K. Erhardt, and G. Rigoll. Interface design for an inexpensive hands-free collaborative videoconferencing system. In *ISMAR*, pages 295 –296, 2012.

[12] A. Maimone and H. Fuchs. Encumbrance-free telepresence system with real-time 3D capture and display using commodity depth cameras. In *ISMAR*, pages 137 –146, 2011.

[13] A. Maimone and H. Fuchs. Computational augmented reality eyeglasses. In *ISMAR*, pages 29–38, 2013.

[14] 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 *IEEE VR*, pages 23–26, 2013.

[15] J. Matas, C. Galambos, and J. Kittler. Robust detection of lines using the progressive probabilistic hough transform. *Computer Vision and Image Understanding*, 78:119–137, 2000.

[16] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, pages 127 –136, 2011.

[17] O. Oyekoya, R. Stone, W. Steptoe, L. Alkurdi, S. Klare, A. Peer, T. Weyrich, B. Cohen, F. Tecchia, and A. Steed. Supporting interoperability and presence awareness in collaborative mixed reality environments. In *VRST*, pages 165–174, 2013.

[18] T. Piumsomboon, A. Clark, A. Umakatsu, and M. Billinghurst. Physically-based natural hand and tangible ar interaction for face-to-face collaboration on a tabletop. In *3DUI*, pages 155 –156, 2012.

[19] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. Kelly, and A. J. Davison. Slam++: Simultaneous localisation and mapping at the level of objects. In *CVPR*, pages 1352–1359, 2013.

[20] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from a single depth image. In *CVPR*, pages 116–124, 2011.

[21] R. S. Sodhi, B. R. Jones, D. Forsyth, B. P. Bailey, and G. Maciocci. Bethere: 3d mobile collaboration with spatial input. In *SIGCHI*, pages 179–188, 2013.

[22] F. Steinbruecker, C. Kerl, J. Sturm, and D. Cremers. Large-scale multi-resolution surface reconstruction from rgb-d sequences. In *ICCV*, pages 3264 – 3271, 2013.

[23] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe. Analyses of human sensitivity to redirected walking. In *VRST*, pages 149–156, 2008.

[24] L. Vera, J. Gimeno, I. Coma, and M. Fernndez. Augmented mirror: Interactive augmented reality system based on kinect. In *INTERACT*, pages 483–486. Springer Berlin Heidelberg, 2011.