# A Video-texture based Approach for Realistic Avatars of Co-located Users in Immersive Virtual Environments using Low-cost Hardware

Robin Horst[1,2,3], Sebastian Alberternst[1], Jan Sutter[1], Philipp Slusallek[1], Uwe Kloos[2] and Ralf Dörner[3]
[1]*German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany*
[2]*Reutlingen University of Applied Sciences, Reutlingen, Germany*
[3]*RheinMain University of Applied Sciences, Wiesbaden, Germany*

Keywords: Mixed Reality, Video Avatar, Multi-user Environments, Low-cost, Computer-supported Cooperative Work, Image Segmentation.

Abstract: Representing users within an immersive virtual environment is an essential functionality of a multi-person virtual reality system. Especially when communicative or collaborative tasks must be performed, there exist challenges about realistic embodying and integrating such avatar representations. A shared comprehension of local space and non-verbal communication (like gesture, posture or self-expressive cues) can support these tasks. In this paper, we introduce a novel approach to create realistic, video-texture based avatars of co-located users in real-time and integrate them in an immersive virtual environment. We show a straight forward and low-cost hard- and software solution to do so. We discuss technical design problems that arose during implementation and present a qualitative analysis on the usability of the concept from a user study, applying it to a training scenario in the automotive sector.

## 1 INTRODUCTION

Avatars are digital representations of users in virtual worlds. The high influence of realistic avatars on the perception, cognition and social interaction of users, compared to abstract user representations, is already accepted (e.g. (Latoschik et al., 2017; Bailenson et al., 2006; Waltemate et al., 2018; Roth et al., 2016)). Indicated by these studies, a higher degree of realism can improve the mentioned aspects concerning the application in virtual reality (VR) and immersive virtual environments (VEs), by transporting human non-verbal cues.

Especially when it comes to social interaction and communication, phenomenons like the Proteus Effect (Yee and Bailenson, 2007) can be observed.

These studies also indicate limitations of the current technical approaches to create highly realistic avatars for virtual environments: Complex and costly professional hardware setups are used (tracking, scanning), a priori operations are needed (attaching tracking points, 3D scanning), avatars are ”baked” after creation (e.g. deformation of clothes must be calculated at the expense of performance), avatars are rendered view-dependent (only one particular side of the user is captured) and on software level, there exist dependencies on (pre-trained) computer vision algorithms to segment textures. It is also indicated, that existing research focuses primarily on self-avatars and not on the representation of co-located co-users.

In this paper, we propose a novel approach to create realistic avatars of co-located users and integrate them into immersive virtual environments. We capture and process video-textures in real-time, relying on low-cost hardware only. We integrate the 2D video-texture into the 3D environment by projecting it onto a 2D plane and position it correctly into the virtual world. These transformations are derived by low-cost tracking hardware (e.g. HTC Vive). We propose a view-independent method to capture the textures from first person point of view (POV) using a simple color + depth camera (e.g. Kinect v2) and a rig. The textures of the co-located users are segmented without relying on existing computer vision methodology. We only require color and depth information and the room-scale tracking position. The hereby created avatars transport many non-visual cues and were received positively by participants of the evaluation, who collaboratively solved spatial tasks. Since a high degree of realism can also be achieved by



Horst, R., Alberternst, S., Sutter, J., Slusallek, P., Kloos, U. and Dörner, R.
A Video-texture based Approach for Realistic Avatars of Co-located Users in Immersive Virtual Environments using Low-cost Hardware.
DOI: 10.5220/0007311602090216
In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2019)*, pages 209-216
ISBN: 978-989-758-354-4

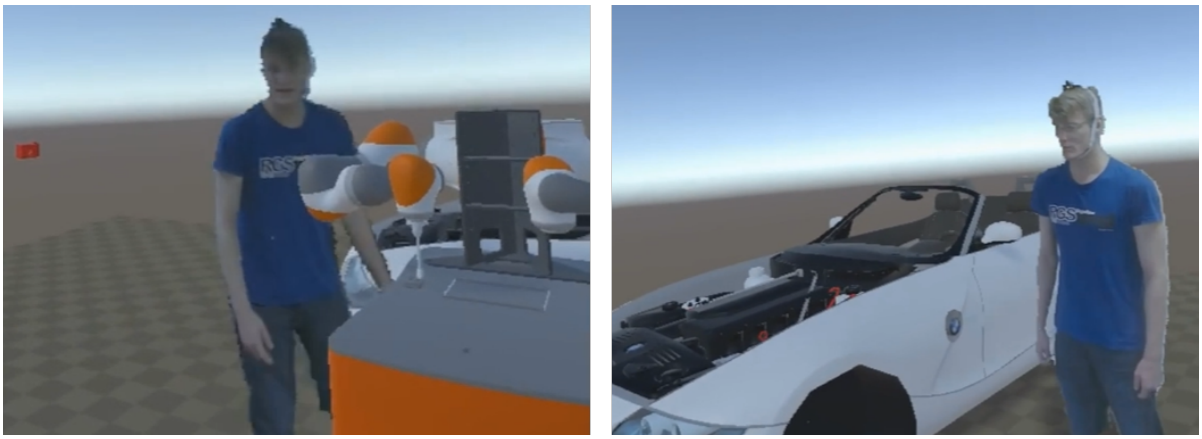

Figure 1: Fully integrated video-texture based avatar within the immersive virtual environment.

realistic avatar motions (Huang and Kallmann, 2009), high physicality (Lok et al., 2014) or haptic feedback (Kotranza et al., 2009), we explicitly focus on the realistic visual appearance of avatars in this work.

With this paper, we make the following contributions:

- We illustrate a low-cost technical system with a straight-forward algorithm to create real-time, video-texture based avatars with a high degree of visual realism by capturing and segmenting textures, integrating them into the immersive VE and handling occlusions.
- Our approach delineates from related work by combining the aspects: Low-cost hardware only, no dependence on pre-trained computer vision for segmentation, no a priori 3D scanning, no texture baking, view-independent rendering of the avatars, focus on co-located user avatars.
- We discuss design challenges and solutions by revealing issues during the hard- and software implementations and evaluating the approach during an application, within a training situation in the automotive sector, and a user study.

The paper is organized as follows: Next, we relate our approach to the existing work. Then, we show the design and an implementation of a prototype that incorporates our proposed approach. The paper is complemented by an evaluation section and completed with conclusions and future work.

## 2 RELATED WORK

Previous research in the creation and integration of realistic avatars offered partial solutions for the mentioned restrictions. Systems used in (Roth et al., 2016; Latoschik et al., 2017; Waltemate et al., 2018) are capable of creating highly realistic avatars and focus on highly professional and costly hardware setups. In (Latoschik et al., 2017) and (Waltemate et al., 2018), the impact of avatar realism and personalization in immersive VEs is studied, using 3D photogrammetric scans of users. Self-avatars are produced thereby. The authors of (Roth et al., 2016) however combine body and facial expression tracking to map it on the avatars at runtime. By using ”fishtank” VR, they propose real-time avatars of co-users and propose a socially immersive avatar mediated communication platform.

Work that focuses on the low-cost aspect (Regenbrecht et al., 2017; Nordby et al., 2016) proposes utilizing a Kinect camera to integrate video avatars of live-captured, co-located (Nordby et al., 2016) or pre-captured (Regenbrecht et al., 2017) co-users into the immersive VE. The static cameras provide visual information from one perspective, so that the avatars are rendered view-dependent. Texture information from other perspectives is not represented in the VE. Similar specifications are indicated in early work by Kato and Billinghurst (Kato and Billinghurst, 1999). Textures are captured and integrated dependent from the view within an augmented reality (AR) conference system. While the textures are highly realistic, they are integrated without further integration processing and displayed in a rectangular shape. Early work by Suma et al. (Suma et al., 2011) proposes to use low-cost depth sensors head worn and therefore addresses similar problems, as the difficulty to use built-in and black-box segmentation methods of the sensors, assuming a fixed position. An empty room with plenty of space between the segmented person and the walls or a fixed distance range is necessary in these solutions respectively. We use a straight-forward approach using a separate low-cost tracker, dedicated to the existing and consumer oriented VR-system, and a

thereon calculated bounding box (BB).

Pioneering work by Beck et al. (Beck et al., 2013) makes use of multiple Kinects to capture video representations of collaborators to display in the context of a shared virtual environment. Whereas Maimone et al. (Maimone and Fuchs, 2011) propose to use multiple Kinects to utilize 3D-reconstruction methods and then place users' representations within virtual worlds. There also exists other work in the broad field of real-time reconstruction of collocated or remote collaborators in virtual and mixed reality settings (e.g. (Lindlbauer and Wilson, 2018; Alexiadis et al., 2013; Fairchild et al., 2017), which makes use of similar technologies. We chose to use only a single low-cost depth camera instead and integrated our setup in a VE including and HMD for visualization.

In (Zhu et al., 2016), a Kinect and green screen based approach is proposed to separate users' textures from the background. Depth compositing then combines a live captured video of the immersive VE and the video-texture avatar into a single 2D video stream. This video stream is provided for a non-immersed audience to share experiences of the immersed user and the VE. This approach is close to fulfilling all criteria that our approach aims at, but consequently, the video texture approach does not integrate the avatar into the immersive VE. Still, by incorporating the depth data, (Zhu et al., 2016) already addresses occlusion handling.

Occlusion handling is of high importance for the integration of 2D video-texture avatars into an immersive VE, too, in order to preserve a highly immersive virtual world. Especially for the application in mixed reality (MR) (Milgram et al., 1995) systems, the correct synthesis of real and virtual information is essential. Detailed information about this problem can be found in current publications (e.g. (Walton and Steed, 2017; Regenbrecht et al., 2017; Hebborn et al., 2017; Fischer et al., 2004; Kiyokawa et al., 2003)). This work differentiates between the type of head-mounted displays that are used for creating the MR environment. The type is either optical see-through (OST) (e.g. Microsoft Hololens) or video see-through (VST) (e.g. Oculus Rift, HTC Vive). While OSTs utilize translucent display devices, VSTs augment the reality by processing a digital video stream (Azuma, 1997). While occlusion handling for OSTs is addressed in (Kiyokawa et al., 2003), Phantom Model Rendering is proposed in (Fischer et al., 2004) for calculating occlusions for VST systems. The latter uses a priori captured information about objects in the scene (e.g. 3D scans) and includes the volumetric depth information about the objects within the occlusion handling.

# 3 LOW-COST VIDEO-TEXTURE APPROACH

We address three key challenges with our approach: At first, we illustrate the low-cost technical setup that we rely on. Thereafter, we show how the video-texture is acquired and then segmented into the human user's texture and the background. Here, we propose an algorithm to elucidate the process. Finally, we state how the video-texture is integrated into an immersive VE.

## 3.1 Technical Setup

Our approach to create realistic avatars of co-located users in immersive VEs uses a low-cost hardware infrastructure. A room-scale VR system is used to create a tracked area, in which an immersed user and co-located persons can move freely (Fig. 2). Here we utilize an HTC Vive. To provide the video-texture and additional spatial information, the immersed user wears a head-mounted tracker-camera unit (tracam). The tracam unit consists of a Microsoft Kinect v2 color+depth camera, a Vive tracker and a helmet-rig, so that the tracam can be worn in addition to the HMD (Fig. 3). Users that are supposed to be captured by the tracam wear a Vive tracker only. On the software level, a game engine is utilized to control the tracking and rendering. Therefore, an instance of Unity runs the system. It is hosted on a single PC.

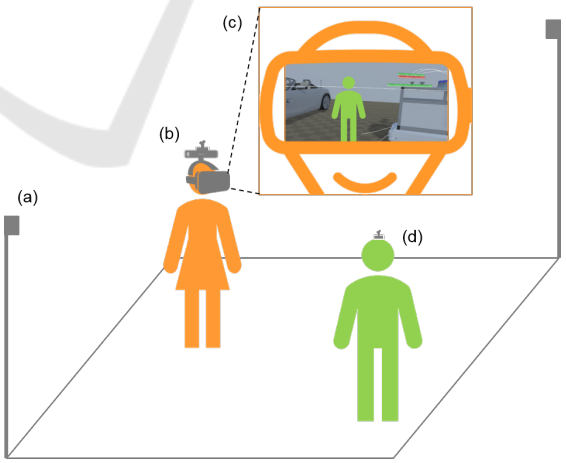


Figure 2: The spatial arrangement of the technical setup: (a) a room-scale VR tracking system; (b) an immersed user wearing a tracked HMD and a tracam unit; (c) the rendered virtual view of the immersed user, augmented by a realistic avatar representation of a co-located person with a tracker (d).

With this technical setup, we are able to gather the following data for further capturing of non-immersed

peoples' texture from the first person POV of the immersed user:

- Color and depth data of the real environment, by the Kinect, including co-located users
- Tracked positions of the HMD, the tracam and co-located users, by the Vive tracking system
- Various data from the virtual Scene (e.g. viewing direction, color, 3D and depth), by the Unity instance

Figure 3: The low-cost tracam unit. Left: standalone; right: on top of an HTC Vive HMD.

## 3.2 Capturing People: Texture Acquisition and Segmentation

The process of capturing surrounding people from first person view of the immersed user is divided into two sub-tasks. We use the tracam-unit to capture color and correspondingly mapped depth information in real-time from first person view of the immersed user. Thereafter, we use a novel technique to segment tracked co-located people, based on the position and distance.

1. *Color and depth image acquisition* – We use the Kinect camera's sensors to capture a 1920x1080 pixel image in BGRA format and a 512x424 pixel depth image. The values of the depth image are mapped onto the color image correspondingly, so that we can assume for further processing, that both color and depth image information are available in 1920x1080 pixel resolution.
2. *Human segmentation* – Since we already make use of the Kinect API to acquire and map color and depth information in a naive implementation, it would be obvious to use the pre-trained computer vision back end of the Kinect API for segmentation, which is based on random decision trees (Shotton et al., 2011). To do so, the camera must not be moved and is constrained to remain in a static position. In our approach however, the

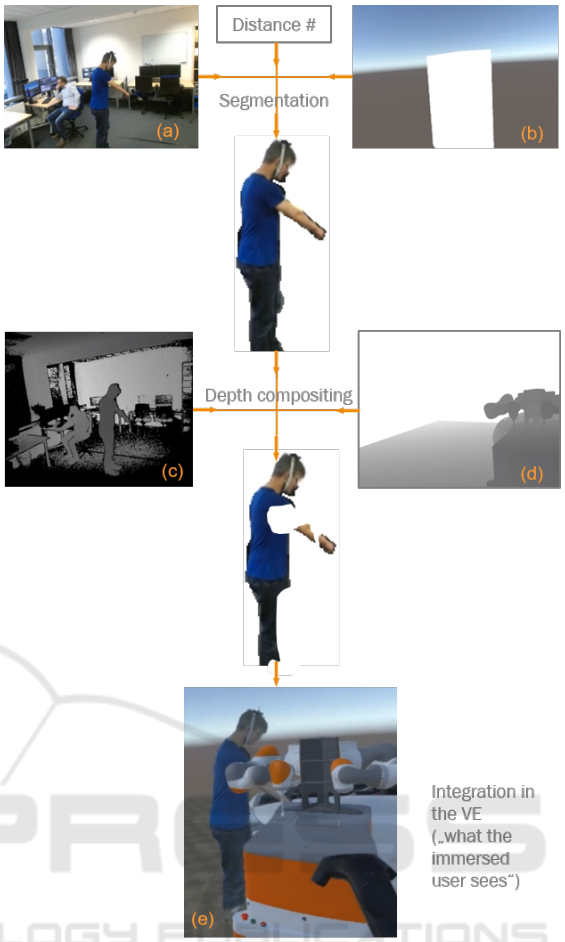


Figure 4: The schematic overview of the texture acquisition and segmentation process: (a) the color information of the real environment and (c) the corresponding depth image, provided by the Kinect (as part of the tracam); (b) the virtual BB, calculated by approximated human measures and the tracked position; (d) the depth information of the VE from first person perspective.

Kinect is part of the tracam and is mounted on the immersed user's head. Since the head is assumed to move freely and continuous during a VR experience, the standard Kinect human segmentation cannot be used in our approach.

We addressed this issue by using a novel approach for object segmentation (Alg. 1). We utilize the positions of the immersed user and the co-located person to calculate a distance vector between them. Based on the tracking data of the co-located person, a 3D BB is rendered in the Unity instance, not visible to the users. Then an image is rendered with the resolution of the Kinect images, showing the BB from the immersed users perspective. The BB has the approximate measures of a human body (Fig. 4 (b)). We divide the person's

texture from the background pixelwise by checking whether a pixel's position falls onto a position of the corresponding virtual BB and furthermore has a distance to the tracam that is in range of an approximated human body around the position of the tracked co-located person. Thereafter, we segment the texture accordingly and pre-compute occlusions by virtual objects. For that, we perform a depth compositing, similar as described in (Zhu et al., 2016), based on the depth information of the VE view and the tracam.

Algorithm 1: Algorithm for VR bounding box segmentation of one image.

**INIT** arrays: *KinectDepthImage, KinectColorImage, BB˙DepthImage, BB˙ColorImage, OutputImage*
**INIT** ints: *UserToUserDistance, BB˙Indices*
1: fetch data from different external sources
2: wait until every source is available
3: **for all** pixels in *KinectColorImage* **do**
4: **if** (pixeldepth of *KinectDepthImage* = invalid) **or** (pixelcolor of *BB˙ColorImage* ! = pure white) **or** (pixeldepth of *KinectDepthImage* > pixeldepth of *BB˙DepthImage*) **then**
5: write alpha value 0 in *OutputImage* pixel
6: **else if** (*UserToUserDistance* >= pixeldepth in *KinectDepthImage* > 0 ) **then**
7: write pixelcolor of *KinectColorImage* in *OutputImage* pixel and update *BB˙Indices*
8: **else**
9: write alpha value 0 in *OutputImage* pixel
10: **end if**
11: **end for**
12: **if** (*BB˙Indices* describe a valid bounding box) **then**
13: send sub-image of *OutputImage* and *BB˙Indices* to VE instance for further texture integration
14: **end if**

## 3.3 Situating People: Integration Into the Immersive VE

Since the co-located person's texture is already segmented, it remains to integrate the texture into the 3D VE. We chose to project the texture onto a 2D plane. This plane always orthogonally faces the virtual camera that renders the VE within the HMD, since the texture is captured from first person POV. We set the size of the plane to the fixed resolution of 1920x1080 pixel, which is the maximum resolution of a texture we can expect from the Kinect's base image. The plane's distance to the virtual camera is adjusted accordingly to the tracked position of the co-located person. Calculated BB indices are used to project only relevant pixels on the plane. To preserve the correct size of the avatar, we also adjust the absolute size, so that longer distances to the camera result in upscaling the texture (Fig. 5 bottom left and right). Hence, we address occlusion handling and ensure that the texture covers a fixed size relative to the virtual view of the immersed user (Fig. 5 top left and right). Also included in this adjustment is the calibration of the tracam and the virtual camera. This can be performed automated and during runtime, since both the tracam and the HMD are tracked independently, Differing angles and scales of the head characteristics of individual users therefore are included.

Altogether, we refer to our approach as "virtual optical-video see-through", since we combine characteristics from the OST and the VST approach to create a MR rendering: A texture is captured by a camera (VST) and projected (OST), however not on a haptic glass/display, but on a virtual plane.

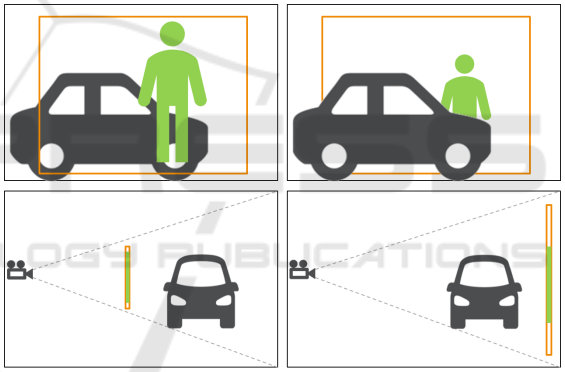

Figure 5: Schematic of the adaptive texture arrangement within the 3D VE. Green: Avatar-area of the rectangular 2D texture; orange: Boundary of the texture. Top left and right: The virtual scene from POV of the immersed user; bottom left and right: Corresponding side-view of the VE.

Regarding the occlusion handling of the texture and the virtual scene, common approaches for VST HMDs, as Phantom Model Rendering (Fischer et al., 2004), cannot be applied. This is, because we exclude a priori operations like scanning the co-located person. The deformation and movement of the human body complicate this approach, too. We already precalculated occlusions by comparing the depth data of the sources. Therefore, occlusions by objects, that are closer to the camera than the distance values of the texture, are handled appropriately.

A challenge that must be considered at this point, is that the projection of the 3D data onto a 2D plane is accompanied by a loss of one dimension. As a consequence, the 2D plane could be placed behind a virtual

object in the scene, because the tracked position is behind the object. However, human extremities, like an outreaching arm, could occlude parts of the object that is rendered entirely in front of the texture. To solve this problem, we propose an offset of the video-textured plane towards the virtual camera. Since we track the torso or head, we assume that we track the middle position of the body and that extremities can only deviate from this position by one length of an arm. Practical tests with different distances indicated, that for the test users 70cm offset counteracted most of the occlusion problems, without visually effecting the quality. This indicates a challenge to be addressed in-depth in future work. The finally arranged and integrated texture is shown in Fig. 1.

# 4 EVALUATION

We already discussed several design issues of our video-texture based approach for realistic avatars of co-located users in immersive VEs using low-cost hardware. Additionally, we evaluated the approach in a preliminary qualitative user study on the usability of the system and the perception of the video-avatars. We applied the approach in a collaborative training scenario, where a trainee was guided through a construction environment in the automotive section. A collaboration with the trainer was necessary to solve several construction tasks.

## 4.1 User Study

The study involved 15 **participants** (10 male, aged 23 to 35, with Ø 27,37 and SD 3,85). They were recruited from the environment of the local university and engineering staff of the faculty of computer science. The system ran on the **technology** described in section 3 using a single windows machine (Intel Core i7 (3,5 GHz), 32 GB RAM, AMD Radeon R9 Fury). The HTC Vive system was utilized to calibrate a $\sim 12m^2$ tracked area for interaction.

The following sequenced **procedure** (after (MacKenzie, 2012)) was performed during the user study: Participants were welcomed and asked to fill out a demographic questionnaire. Thereafter they were instructed about the technical setup of the study. After about 12 minutes of participation and interaction with the experimenter (as co-located person) in a collaboratively spatial task environment, they were asked to fill out a post-study questionnaire. The experimenter could see himself within the virtual world on an external screen, using an approach as in (Zhu et al., 2016). Additionally, a semi-structured interview about the experience was conducted.

The **design** of the questionnaire items was related to subjective perception of the scene and the collaborator avatars. Open-ended questions were used to gather qualitative comments about satisfaction. The content of the VE and its task were related to car assembly, assuming that none of the participants had prior knowledge.

## 4.2 Discussion

We evaluated the qualitative results interview by identifying different themes and aspects by utilizing affinity diagrams (Beyer and Holtzblatt, 1997) as proposed by Preece, Rogers and Sharp (Preece et al., 2015). The qualitative observations and comments overall indicate, that the low-cost video-texture avatars were perceived as helpful and adequate to the purpose of visually collaborating and solving spatial tasks. In the open-ended questions the users were asked to describe what they liked/disliked in general, as well as specifically about the avatar representation and the communication and interaction with it. Participants positively connoted the visual representation, but mentioned some issues relating to technical concerns which must be regarded in further implementations: Depending on the angle of view of the immersed user towards the co-located person, parts of the avatar were cut off. Especially when the avatar tended to be placed in the outer peripheral regions of the view, our approach is limited by the Kinect's specifics. Small outer regions cannot be covered by the Kinect. Depth values are missing here due to different aspect ratios of the BGRA ($\sim$16:9) and the depth sensor ($\sim$64:53) as well as the Vive HMD ($\sim$9:10).

Furthermore, participants were asked about the quality of non-verbal cues during the interview and how they could interpret the co-located persons actions. Participants particularly suggested, that the visual cues of the video-texture avatars could be interpreted well. Gestures and facial expressions were understood especially at close distance, but decreasing quality of the avatar texture at higher distance was mentioned. This can be attributed to the fixed resolution of the Kinect, of which only one portion is used for the avatar texture. Allover, participants mentioned the usefulness of the avatars, but mentioned that the tracam-rig was quite uncomfortable for being mounted on the head. Unexpectedly, one participant mentioned an issue relating to the lighting of the avatar. In contrast to the virtually illuminated and lighted objects within the VE, our 2D video-texture

avatar is highly affected by the lighting that is prevailing in the real environment. These lighting conditions differed during the study, so that one participant found the avatar representations subjectively unnatural. The description as eerie and non-attractive/-human indicates a relation to an uncanny perception of the avatar (uncanny valley, (Mori, 1970)) and must be regarded in future approaches.

# 5 CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel approach for creating video-texture based realistic avatars of co-located users and integrating them in immersive virtual environments. Our approach is characterized by the following aspects:

- *Low-cost hardware* – We only rely on low-cost hardware, like an HTC Vive, a Kinect and a do-it-yourself tracam rig.
- *No dependence on pre-trained computer vision* – Our approach does not harness any pre-trained models for recognition or segmentation of the video textures.
- *No prior 3D scanning* – Some approaches that create and integrate realistic avatars, make highly use of a priori actions, like 3D scanning. We have shown how realistic avatars can be created and integrated entirely at runtime.
- *No texture baking* – We create personalized avatars dynamically, so that the texture can be changed during the use (e.g. changing appearance of clothes during movement).
- *View in-dependent rendering* – Due to the tracam unit, we are able to capture textures and render the avatar representations with respect to the POV of the immersed user. As a consequence, it can be seen by the user intuitively from all perspectives.
- *Focus on co-located people* – Our approach makes use of the physical closeness of co-located people. They are captured by a camera and can move freely within the room-scale area.

We have shown the technical details and discussed design challenges that arose during application of the concept. The integration of video-textures as avatars is covered, with emphasis on capturing and segmenting textures and arranging them within the 3D VE, considering occlusion handling. The use of a BB, whose location is defined by the head tracking information rather than just arbitrary depth thresholding, is an addition that has the potential to help keep the implementation robust when used in a crowded space, but must be evaluated in future implementations. Furthermore, we evaluated the concept qualitatively in a user-study. This indicated issues and phenomenons, which sincerely must be considered during future work within the area of creating realistic avatars.

Since our work lays the technical foundation, we will focus on issues we pointed out, next. One aspect that we intentionally left out of regard in this work is the synchronization. All producing sources currently work on the same frames per second count (30; except for the final VR rendering) We could already indicate within the user-study, that the delay between the avatar-motions and the remaining VE was of neglectable matter for our users. Still, this aspect provides space for further technical work. We consider real-time streaming technology (e.g. real-time streaming protocol (Schulzrinne, 1998)) as an appropriate methodology for MR media synchronization.

We believe that our approach constitutes an important stride towards low-cost avatar creation for immersive VEs. We pursue the vision, that this will become increasingly important, as VR technology already reached the consumer market, so that a future application in social media and games is predictably necessary.

# ACKNOWLEDGEMENTS

The work is supported by the Federal Ministry of Education and Research of Germany in the project HybriT (Funding number:01IS16026A). The work furthermore is minor supported by the Federal Ministry of Education and Research of Germany in the project Innovative Hochschule (Funding number: 03IHS071).

# REFERENCES

Alexiadis, D. S., Zarpalas, D., and Daras, P. (2013). Real-time, realistic full-body 3d reconstruction and texture mapping from multiple kinects. In *IVMSP Workshop, 2013 IEEE 11th*, pages 1–4. IEEE.

Azuma, R. T. (1997). A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*, 6(4):355–385.

Bailenson, J. N., Yee, N., Merget, D., and Schroeder, R. (2006). The effect of behavioral realism and form realism of real-time avatar faces on verbal disclosure, nonverbal disclosure, emotion recognition, and copresence in dyadic interaction. *Presence: Teleoperators and Virtual Environments*, 15(4):359–372.

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

Beyer, H. and Holtzblatt, K. (1997). *Contextual design: defining customer-centered systems*. Elsevier.

Fairchild, A. J., Campion, S. P., García, A. S., Wolff, R., Fernando, T., and Roberts, D. J. (2017). A mixed reality telepresence system for collaborative space operation. *IEEE Trans. Circuits Syst. Video Techn.*, 27(4):814–827.

Fischer, J., Bartz, D., and Straßer, W. (2004). Occlusion handling for medical augmented reality using a volumetric phantom model. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 174–177. ACM.

Hebborn, A. K., Höhner, N., and Müller, S. (2017). Occlusion matting: Realistic occlusion handling for augmented reality applications. In *Mixed and Augmented Reality (ISMAR), 2017 IEEE International Symposium on*, pages 62–71. IEEE.

Huang, Y. and Kallmann, M. (2009). Interactive demonstration of pointing gestures for virtual trainers. In *International Conference on Human-Computer Interaction*, pages 178–187. Springer.

Kato, H. and Billinghurst, M. (1999). Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 85–94. IEEE.

Kiyokawa, K., Billinghurst, M., Campbell, B., and Woods, E. (2003). An occlusion-capable optical see-through head mount display for supporting co-located collaboration. In *Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, page 133. IEEE Computer Society.

Kotranza, A., Lok, B., Pugh, C. M., and Lind, D. S. (2009). Virtual humans that touch back: enhancing nonverbal communication with virtual humans through bidirectional touch. In *Virtual Reality Conference, 2009. VR 2009. IEEE*, pages 175–178. IEEE.

Latoschik, M. E., Roth, D., Gall, D., Achenbach, J., Waltemate, T., and Botsch, M. (2017). The effect of avatar realism in immersive social virtual realities. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*, page 39. ACM.

Lindlbauer, D. and Wilson, A. D. (2018). Remixed reality: Manipulating space and time in augmented reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 129. ACM.

Lok, B., Chuah, J. H., Robb, A., Cordar, A., Lampotang, S., Wendling, A., and White, C. (2014). Mixed-reality humans for team training. *IEEE Computer Graphics and Applications*, 34(3):72–75.

MacKenzie, I. S. (2012). *Human-computer interaction: An empirical research perspective*. Newnes.

Maimone, A. and Fuchs, H. (2011). Encumbrance-free telepresence system with real-time 3d capture and display using commodity depth cameras. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 137–146. IEEE.

Milgram, P., Takemura, H., Utsumi, A., and Kishino, F. (1995). Augmented reality: A class of displays on the reality-virtuality continuum. In *Telemanipulator and telepresence technologies*, volume 2351, pages 282–293. International Society for Optics and Photonics.

Mori, M. (1970). The uncanny valley. *Energy*, 7(4):33–35.

Nordby, K., Gernez, E., and Børresen, S. (2016). Efficient use of virtual and mixed reality in conceptual design of maritime work places. In *15th International Conference on Computer and IT Applications in the Maritime Industries-COMPIT'16*.

Preece, J., Rogers, Y., and Sharp, H. (2015). *Interaction design: beyond human-computer interaction*. John Wiley & Sons.

Regenbrecht, H., Meng, K., Reepen, A., Beck, S., and Langlotz, T. (2017). Mixed voxel reality: Presence and embodiment in low fidelity, visually coherent, mixed reality environments. In *Mixed and Augmented Reality (ISMAR), 2017 IEEE International Symposium on*, pages 90–99. IEEE.

Roth, D., Waldow, K., Stetter, F., Bente, G., Latoschik, M. E., and Fuhrmann, A. (2016). Siamc: a socially immersive avatar mediated communication platform. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*, pages 357–358. ACM.

Schulzrinne, H. (1998). Real time streaming protocol (rtsp).

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1297–1304. Ieee.

Suma, E. A., Krum, D. M., and Bolas, M. (2011). Sharing space in mixed and virtual reality environments using a low-cost depth sensor. In *VR Innovation (ISVRI), 2011 IEEE International Symposium on*, pages 349–350. IEEE.

Waltemate, T., Gall, D., Roth, D., Botsch, M., and Latoschik, M. E. (2018). The impact of avatar personalization and immersion on virtual body ownership, presence, and emotional response. *IEEE transactions on visualization and computer graphics*, 24(4):1643–1652.

Walton, D. R. and Steed, A. (2017). Accurate real-time occlusion for mixed reality. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*, page 11. ACM.

Yee, N. and Bailenson, J. (2007). The proteus effect: The effect of transformed self-representation on behavior. *Human communication research*, 33(3):271–290.

Zhu, Y., Zhu, K., Fu, Q., Chen, X., Gong, H., and Yu, J. (2016). Save: shared augmented virtual environment for real-time mixed reality applications. In *Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry-Volume 1*, pages 13–21. ACM.