High-quality First-person Rendering Mixed Reality Gaming System for In Home Setting

Yu-Yen Chung*, Hung-Jui Guo†, Hiranya Garbha Kumar‡ and Balakrishnan Prabhakaran§
The University of Texas at Dallas, Richardson, Texas 75080-3021
{Yu-Yen.Chung*, Hung-Jui.Guo†, hiranya‡, bprabhakaran§}@utdallas.edu

***Abstract*—With the advent of low-cost RGB-D cameras, mixed reality serious games using 'live' 3D human avatars have become popular. Here, RGB-D cameras are used for capturing and transferring user's motion and texture onto the 3D human avatar in virtual environments. A system with a single camera is more suitable for such mixed reality games deployed in homes, considering the ease of setting up the system. In these mixed reality games, users can have either a third-person perspective or a first-person perspective of the virtual environments used in the games. Since first-person perspective provides a better Sense of Embodiment (SoE), in this paper, we explore the problem of providing a first-person perspective for mixed reality serious games played in homes. We propose a real time textured humanoid-avatar framework to provide a first-person perspective and address the challenges involved in setting up such a gaming system in homes. Our approach comprises: (a) SMPL humanoid model optimization for capturing user's movements continuously; (b) a real-time texture transferring and merging OpenGL pipeline to build a global texture atlas across multiple video frames. We target the proposed approach towards a serious game for amputees, called Mr.MAPP (Mixed Reality-based framework for Managing Phantom Pain), where amputee's intact limb is mirrored in real-time in the virtual environment. For this purpose, our framework also introduces a mirroring method to generate a textured phantom limb in the virtual environment. We carried out a series of visual and metrics-based studies to evaluate the effectiveness of the proposed approaches for skeletal pose fitting and texture transfer to SMPL humanoid models, as well as the mirroring and texturing missing limb (for future amputee based studies).**

***Index Terms*—Mixed Reality, Real-time Textured Humanoids, First-Person Perspective Rendering**

I. Introduction

With the advent of RGB-D cameras such as Microsoft Kinect, mixed reality systems and applications have incorporated "live" 3D human models [1]. These "live" 3D human models are generated in real-time by capturing the RGB (texture), human skeleton, and the depth data associated with a human in a 3D scene and using these data to create 3D mesh in real-time and applying texture over the mesh. Such "live" 3D human models give users a better sense of immersion as they can see details such as the dress the human is wearing, their facial emotions, etc. Sense of Embodiment (SoE) is defined as "the ensemble of sensations that arise in conjunction with being inside, having, and controlling a body" [2]. SoE is an essential part for enhancing user experience in an immersive virtual environment. Recent studies [3], [4] have indicated that users experiencing first-person perspective (1PP) in immersive environments had a better SoE toward their virtual body than those with a third-person perspective (3PP).

In-home Mixed Reality Serious Games: Once the "live" 3D human mesh models are generated, they can be manipulated just like other 3D graphical models. This fact was exploited by Mr. MAPP (Mixed reality-based framework for MAnaging Phantom Pain) [5], [6] to create an in-home serious game. Phantom pain is typically felt after amputation (or even paralysis) of limbs. Such phantom pain patients experience vivid sensations from the missing body part such as frozen movement or extreme pain. For intervention, mirror therapy [7] shows that perceptual exercise such as mirror box help the patient's brain to learn the fact that the limb is paralyzed or amputated. Mr.MAPP is a suite of serious games targeting upper and lower limb amputees to manage phantom pain. It 'replicates' a 3D graphical illusion of the intact limb, to create a similar illusion as the one in the mirror-box therapy.

A. Challenges for First-person Perspective In-home Serious Games

Mr.MAPP is successful in rendering 3PP but insufficient to reconstruct the portion invisible from the camera in 1PP. The person's mesh looks intact in 3PP (Fig. 1(b)) even though the right leg was folded as Fig. 1(a). In the 1PP as Fig. 1(c), however, the holes on lower limbs appears because the upper side of feet and lower part of leg were blocked by the sole. One possible solution is to use pre-defined humanoid models. Skinned Multi-Person Linear model (SMPL) [8] is parametric humanoid model which contains shape and pose parameters for customizing 3D model of a person without texture. With SMPL, users will be able to see the personalize humanoid avatar in the virtual environment. However, the model will not be as vivid as the "live" 3D human mesh generated in real-time using RGB-D cameras, since the SMPL model is an avatar without any texture.

B. Real-time Textured Humanoid Framework

In this paper, we target the problem of generating high-quality first-person rendering in mixed reality serious games, especially for in-home applications. Such a rendering will use the texture data (in real-time) pertaining to the human in the 3D scene captured by a Kinect RGB-D camera. This real-time texture will be transferred from the human in the scene to a customized SMPL humanoid model and will provide an enhanced SoE in 1PP. The entire series of operations to achieve

978-1-7281-7463-1/20/$31.00 ©2020 IEEE
DOI 10.1109/AIVR50618.2020.00070

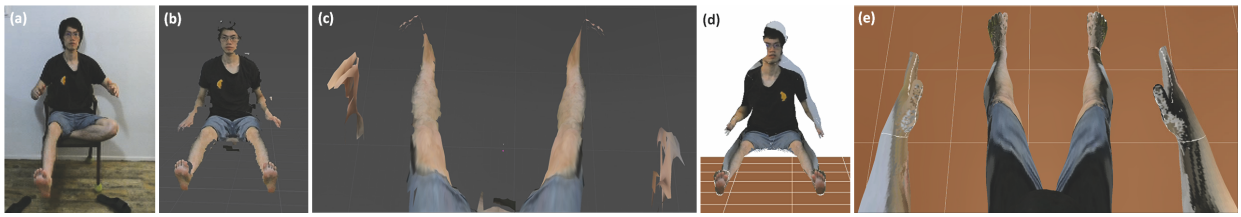


Fig. 1. Comparison the mirrored model from Mr.MAPP and proposed method with Kinect at 80 inches high. (a) RGB image from Kinect; (b) Mr.MAPP in third person perspective; (c) Mr.MAPP in first person perspective; (d) textured SMPL in third person perspective; (e) textured SMPL in first person perspective.

this goal include: two stage SMPL model optimization for matching the humanoid to the human in the scene, real-time texture registration and transfer from the human captured by a Kinect RGB-D camera. We target the proposed approach towards the serious game for amputees, Mr.MAPP. While the proposed approach will work for non-disabled people (with intact limbs) in a straight-forward manner, we outline an approach that can: (a) replicate the 3D graphical limb from the customized humanoid; (b) transfer the texture data from the intact human limb. Through the sample resulting model in 3PP from Fig. 1(d), we can see the position of body, hands and leg are mostly aligned with the SMPL model. Further, in 1PP as Fig. 1(e), basically most of the visible surface are textured. Hence, this approach can be incorporated in Mr.MAPP to provide better SoE for amputees using the serious games.

For validation, we carried out a series of visual and metrics-based studies to evaluate the effectiveness of the proposed approaches for: (i) real-time skeletal pose fitting and (ii) texture transfer to SMPL humanoid models, as well as the (iii) mirroring and texturing missing limb (for future amputee based studies).

Main contributions of this work includes:

- A two stage SMPL optimization strategy incorporating shape initialization and near real-time pose update. To enhance stability of initialization, a randomized repetition procedure was introduced and validated by Otte's Kinect V2 dataset [9].
- An approach for real-time mirroring of the missing limb (in case of amputees) so that the approach could be incorporated in Mr.MAPP for providing better SoE.
- A novel approach through simulating the visible area to evaluate texture retrieval quality with different positions of Kinect RGB-D camera setup.

II. Related Works

Reconstructing a live 3D human model is a challenging task due to the non-rigid deformation of body shape, potentially fast motion, and large range of possible poses. Typically, to achieve viewpoint free model, many approaches have been developed for fusing information from multiple cameras. On the contrary, for systems targeting regular in-home based users, reconstructing from a single camera would be a good alternative.

A. Multi-views systems

With a set of RGB-D cameras, point cloud is a simple representation for volumetric video since it does not require much preprocessing for rendering. These point clouds from various cameras can be aligned based on the intrinsic and extrinsic parameters or Iterative Closest Point (ICP) algorithm [10]. Mekuria et al. [11] introduced a point cloud compression codec to transfer the thousands up to millions of points efficiently. Truncated signed distance function (TSDF) [12] is another data structure which fuse the depth data into a predefined volume. Holoportation [13] reconstructs both shape and color of whole scene based on the TSDF. In Fusion4D [1], they additionally maintained key volumes to deal with radically surface changing and smooth nonrigid motions was applied within the key volume sequence. Dou et al. [14] further improves the frame rate of Fusion4D by spectral embedding and more compact parameterization. For the texture reconstruction, Du et al. [15] apply majority voting to enhance the quality of resulting texture. As the task specific target on human performance capture, Xu et al. [16] uses SMPL [8] as a core shape and an outer layer to represent the structure of outfit.

B. Single view systems

In case of single view, the limited viewpoint and occlusion make the reconstruction even harder. Hence, to have a first-person perspective game with a single camera from different perspective, a prior model become more important in the setup. SMPL [8] is a parameterized human body model which has been applied in many researches to achieve better human shape or pose estimation. Further, techniques used by single view system vary depending on whether depth information is provided (using RGB-D cameras) or not (only RGB cameras).

1) RGB Camera-based Single View: With a proper model, some research could even build the reconstruction on top on RGB data without depth information. For a single photo, Bogo et al. [17] fit a SMPL model toward 2D skeleton joint with other shape constrains for reconstructing the pose and shape of the person. Pavlakos et al. [18] further extends SMPL with fully articulated hands and an expressive face to retrieve facial expression and hand gestures. For the case with a video, Alldieck et al. [19], [20] applied SMPL with an extra offset layer to reconstruct a textured model. For achieving finer texture, they further trained a CNN (Convolutional Neural Networks) for precise texture registration. However, the time

taken by the method is of the order of minutes, and not really in real-time. With a prebuilt textured mesh and parameterized motion, Xu et al. [21] and Habermann et al. [22] pre-trained a CNN to convert the 2D skeletal joint to 3D space then perform a non-rigid deformation based on the joints. Yet, the texture will stay the same unless the user reconstruct another model.

2) RGB-D Camera-based Single View: The depth sensor provides information about the 3D structure. Some approaches could still build on a model free framework. DynamicFusion [23] was simultaneously reconstructing a canonical scene and tracking a non-rigid deforming scene using TSDF in real-time without a prior model. Volumedeform [24] extended the work by using SIFT (Scale Invariant Feature Transform) features to align the images. Yu et al. [25] added the skeleton information in DynamicFusion to better fit a full body shape as targeting on human. Doublefusion [26] further leveraged a SMPL layer to improve the shape. Guo et al. [27] applied estimated albedo under Lambertian surface assumption to improve the tracking. These methods use TSDF with GPU to achieve real-time performance. Yet, the voxel grid may require large amount of GPU memory and thus bounded the shape to predefined volume [28]. To sum up, with single RGB camera only system, user will not be able to change their texture without a model re-building process. Moreover, it is harder to achieve real-time performance without the depth information. Since RGB-D cameras are easily available, our system uses a single RGB-D camera with SMPL model and provide real-time performance.

III. Proposed Method

Figure 2 summarizes the modular framework of our proposed system for high quality first-person rendering in mixed reality, along with the evaluation strategies (used in Section IV):

1) Fitting the SMPL model with the joints of each frame to represent the user's model in action. This operation includes model initialization along with pose updates.
2) For rendering, texture is transferred to the global atlas using RGB image data from Kinect with fitted SMPL model.
3) In the case of serious games for amputees, we mirror the intact limb of the amputee to create an illusion of the missing limb. This mirroring process includes creation of the missing skeletal joints and the associated texture information.

A. SMPL model optimization

The proposed approach fits the SMPL joints with the 3D joints detected by a Kinect RGB-D camera, in near real-time, for each frame. The optimization is carried out in two stages: (i) Shape initialization; (ii) Real-time pose update. L-BFGS (Limited-memory Broyden–Fletcher–Goldfarb–Shanno) optimization [29] is performed on each frame in both the stages.

1) Shape Initialization: Shape initialization is to determine the shape (β) and initial pose (θ) for the SMPL model from a set of frames of 3D human skeleton joints in the beginning. Due to small pose differences between consecutive frames,

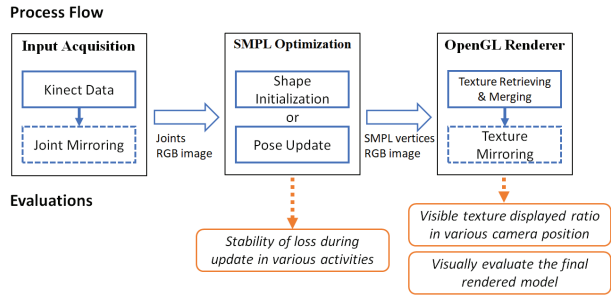


Fig. 2. Proposed Approach for High Quality First-Person Rendering with SMPL optimization and OpenGL

the set of optimized parameters could be applied as initial value for the next coming frame. Moreover, to avoid the objective function being trapped into local minimum, extra n_{rep} times repeated optimizing on randomized parameters were performed for each frame in the stage to determine the best result for the current frame. The objective function of the initializing stage, $E_{init}(\beta, \theta, \hat{J})$, considered the mismatch of joints position and regularization term for both θ and β.

$$E_{init}(\beta, \theta, \hat{J}) = \lambda_j E_j(\beta, \theta, \hat{J}) + \lambda_p E_p(\theta) + \lambda_t E_t(\theta) + \lambda_b E_b(\beta) \tag{1}$$

wherein λ_j, λ_t, λ_p and λ_b are the weighted parameters for the objective function. $E_j(\beta, \theta, \hat{J})$ is the loss corresponding to the joint mismatch between SMPL and estimated through Kinect V2. This term is the major source of the whole energy function.

$$E_j(\beta, \theta, \hat{J}) = \sum_{j \in J_m} \rho(R_\theta(J(\beta))_j - \hat{J}_j) \tag{2}$$

The set of joints that could be mapped between SMPL and Kinect camera extracted joints denote J_m. $\hat{J}$ refers to the joint set estimated by Kinect. R_θ is the global rigid transformation derived from θ. Hence, the 3D position of SMPL joints is $R_\theta(J(\beta))$. ρ is the robust differentiable Geman-McClure penalty function.

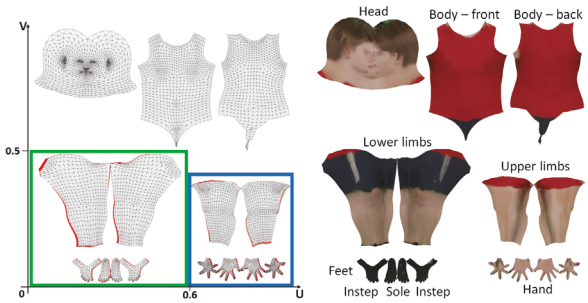
$$E_p(\theta) = min_j(-ln(g_j N(\theta; \mu_{\theta,j}, \Sigma_{\theta,j}))) \tag{3}$$

$$E_t(\theta) = exp(\sum_{i \in J_s} A(\theta)_i) + \sum_{j \in J_r} exp(A(\theta)_j) \tag{4}$$

$E_p(\theta)$ is the Gaussians mixture pose prior introduced from Bogo et al's work [17]. $E_t(\theta)$ is the term to regularize θ for handling/preventing the twisted or unnatural poses. $A(\theta)_i$ is the angle of i_{th} set of θ. For avoiding cumulative error over a few consecutive spine joints, angles corresponding to spine (J_s) and the rest (J_r) (except the global transformation) are calculated differently. Last, $E_b(\beta)$ is a ridge regularization term for beta.

$$E_b(\beta) = \|\beta\|^2 \tag{5}$$

2) Pose update: The pose update stage could be regarded as a simplified initialization, for achieving real-time performance. Once the shape has been initialized, β become a set of constants for the following pose update stage. Also, no extra



(a) Unwrap of neutral SMPL (b) Example atlas [19]

Fig. 3. SMPL texture atlas with limbs unwrapped symmetrically

repetition is required since θ has been properly initialized in the previous stage as well. $E_{pose}(\beta, \theta, \hat{J})$ shows the objective function for this stage.

$$E_{pose}(\beta, \theta, \hat{J}) = \lambda_j E_j(\beta, \theta, \hat{J}) + \lambda_t E_t(\theta) \tag{6}$$

B. Texture Atlas and Missing limb generation

To render the fitted model with a vivid appearance, texturing is required. Rather than applying a synthetic and pre-defined texture, our approach seeks to transfer the texture from the 'live' human user that is captured by the Kinect RGB-D camera, to the fitted SMPL model. Specifically, we back project the triangle from world space to RGB image for retrieving the corresponding texture piece on the texture atlas of SMPL. To create a textured and interactable representation of the phantom limb for the amputee, both joints and texture will be mirrored from the intact counterpart. For joints mirroring, a method based on the joints of hip, shoulder or spine introduced from Mr.MAPP [5], [6] is applied to generate the joints on the amputee's missing limb. For texture mirroring, a SMPL was unwrapped symmetrically on both upper and lower limbs using Autodesk Maya[1]. Fig. 3 shows the unwrapped neutral SMPL. In Fig. 3a, the region labeled by green and blue box are region for limb mirroring. The red margin labeled the unsymmetrical area since the SMPL not exactly symmetrical on the both sides. Thus, with the limb mirroring texture atlas, the texture mirroring can be integrated into OpenGL efficiently.

IV. Experimental Results

Prototype of the proposed approach was developed on Python 3.6. The optimization is running on intel i7-8750H CPU with 32GB RAM using PyTorch 1.3.1. The rendering engine is Panda3D[2] running on dedicated graphic card NVIDIA GeForce RTX 2070, with OpenGL 4.5. The human body skeletal joints and RGB image from Kinect cameras are extracted using Kinect SDK 2.0. We conducted several experiments to evaluate the proposed approach with the following goals: (a) Skeletal pose to SMPL model fitting and pose updates; (b) Effectiveness of real-time texture transfer; (c) Missing limb generation and the associated texture transfer.

[1]https:/autodesk.com/maya
[2]https://www.panda3d.org/

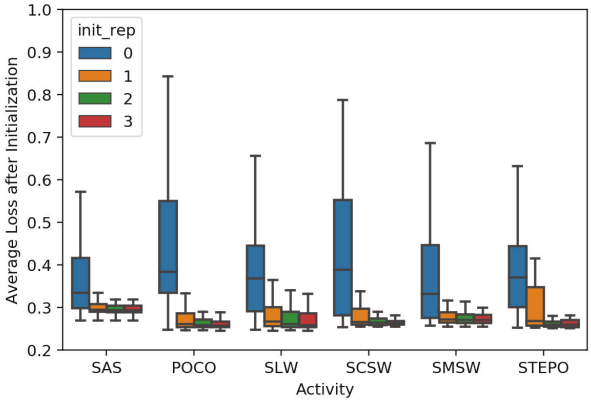


Fig. 4. Comparison of average loss after different repeat times of random initialization using Otte's activity dataset.

A. Shape Initialization

To enhance the stability, we introduced a random repetition strategy in the initialization process and validated the strategy using Otte's activity dataset [9]. In Otte et al's work, they created a set of Kinect RGB-D skeleton dataset with six different activities, which contains Stand up And Sit down (SAS), Short Comfortable Speed Walk (SCSW), Short Maximum Speed Walk (SMSW), Short Line walk (SLW), Stance with closed feet and open and closed eyes (POCO) and Walking on the spot (STEPO). Data were collected from a total of 19 user trials with 3 repetitions for STEPO and 5 repetitions for the rest activities. Fig. 4 shows the comparison of initialized result in different times of random repetitions. In this experiment, the beginning 10 frames of each trials were used for shape initialization then the average loss from the following 50 frames on pose update were collected as for evaluating the SMPL optimization. Each activity (with the above acronyms) is listed in X axis as a group and considered separately since the fitting result for different pose could be varied. The Y axis is distribution the average loss across trials. In each group, the blue boxes are results performed without repetitions in the initialization stage and the others are involving different number of repetitions. For all activities, the averaged loss and its variation in the following pose update stage are much lower when we have repetitions (as opposed to no repetitions). Thus, the repetitions result in better and more stable performance in the initialization stage.

B. Simulation of Texture Atlas Reconstruction with Different Camera Height

In this experiment, the goal is to estimate effect of camera height for texture atlas reconstruction in a first-person game. The example gaming scenario is kicking lower limb forward in a sitting pose as Bahirat et al.'s work [6]. In Alldieck et al's work [19], they released a dataset of total 24 the textured SMPL with shape parameter and additional personal offsets. We first recorded movements of the body joints in the example gaming scenario to fit the pose parameters for operating these

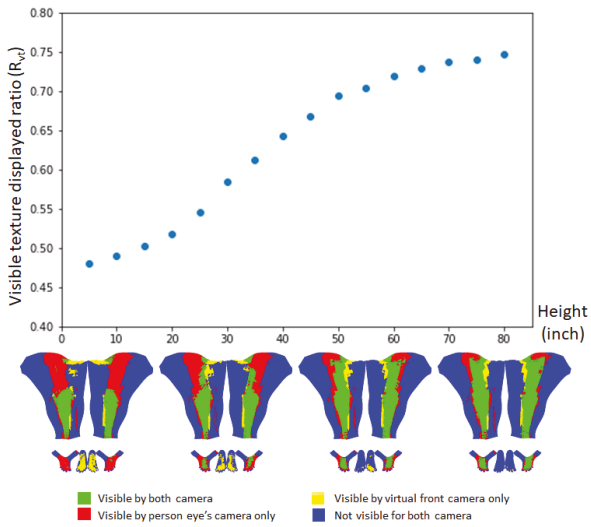


Fig. 5. Visible texture displayed ratio at different camera setup. The sub-figures below are the corresponding overlap of texture on lower limb portion with front camera at 5, 30, 55 and 80 inches high.

models. Then, to simulate the retrievable and visible texture areas, we use two virtual cameras: (i) a front camera was set at 2.5m in front of these personalized SMPL model; (ii) a virtual eye's camera with FoV 95^ox75^o located at 0.2m front of the head joint and looked toward to the fitted SMPL model's lower limb to mimic the eyes' vision. For evaluating the effectiveness of texture reconstruction with respect to the height of placement of the virtual front camera, we defined a visible texture displayed ratio ($R_{vt}(g, c)$):

$$R_{vt}(g, c) = \frac{area(T_{vis}(g) \cap T_{ret}(c))}{area(T_{vis}(g))} \tag{7}$$

$R_{vt} \in [0, 1]$, $R_{vt} = 1$ means all the portions of the 3D avatar seen by the user are textured; on the contrary, $R_{vt} = 0$ means none of object seen by the user is textured no matter how much texel has been retrieved. $T_{vis}(g)$ refers to the set of texel visible by the user in a gaming scenario (g), $T_{ret}(c)$ refers to the to the set of texel retrieved using camera configuration (c). The camera configuration may involve camera properties such as FoV, position and orientation in general but we focus on the effectiveness of camera height in our simulation. To calculate the R_{vt} here, the two parameters are estimated as below:

1) Visible to virtual front camera at testing height (T_{ret}): the texel correctly retrieved from over 80 percent of trials.
2) Visible to virtual eye's camera (T_{vis}): surface seen by virtual eye's camera in any trial.

The resulting R_{vt} in Figure 5 demonstrates this simple concept. As the camera height increases from 5 to 80 inches height, the corresponding R_{vt} increased from around 0.48 to 0.75. Moreover, the overlapped texture area on lower limb with cameras at 5, 30, 55 and 80 inches height are listed in the Figure 5. The meaning of each color is described as following:

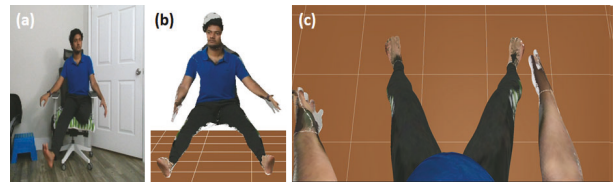


Fig. 6. Example of lower limb mirrored SMPL model. (a). RGB frame from Kinect; (b),(c). textured SMPL in 3PP and 1PP.

- Green: is the area visible to both eye's and the virtual front camera. Hence, these textures would contribute to the user's vision in the scene.
- Red: is the area visible only to the eye's camera. More red area on upper face of feet and limb in the sub-figure on the lower virtual front camera setup. In other word, user will not see texture on these areas eventually it might lower the fidelity of the game.
- Yellow: is visible only to the virtual camera at the front position, which means user will not notice this area in the given gaming scenario.
- Blue: the area which will not be captured neither by eye's nor virtual front camera.

Through this series of sub-figure, as height of the virtual front camera increases, the camera view gradually aligned with the person's perspective since the green area increase. As a result, raising the camera's height is improving the texture retrieval in this gaming scenario. The result consistent with intuition since user will see more upper face of their lower limb in 1PP in our example gaming scenario.

C. Missing limb generation

One main motivation of our system is to have a textured humanoid avatar with a generated phantom limb. The lower limb mirrored examples in 3PP and 1PP are provided in Fig. 1d, e. and Fig. 6b, c. Basically all the visible surface in 1PP are hole-free and textured. While, there are still some areas without proper texture on feet, hands, and pants, since the SMPL may not exactly match a body in the image. In summary, our system provides a vivid limb-mirrored textured model for the lower limb actions in first-person perspective with the Kincet RGB-D camera at the front.

V. Concluding Remarks

In this work, we introduced a real-time textured humanoid avatar framework for mixed reality games in 1PP using a single RGB-D camera, especially for in-home deployment. The framework reduced the camera position's effect through a skeleton-based virtual floor calibration. We employed a two-stage optimization strategy to update SMPL model pose with respect to the pose of the user. Texture transfer to the SMPL model was carried out by capturing a set of frames and incorporating the accumulated texture with a global atlas, in real-time. In the case of serious games for managing phantom pain, the system could generate a vivid phantom limb by mirroring the joints, limb, and the associated texture, from the amputee's in-tact limb. To increase the visible texture display

ratio, we can ask users to perform some more initializing activity to let the Kinect RGB-D camera capture more texture or adjust the front camera to a higher position. However, for texture transferring, we found some portion of face, hand and leg may not be well aligned if we perform a back projection directly. Thus, a more sophisticated registration approach may improve the texture retrieval of the system. Our prototype runs in near real-time, with SMPL optimization being carried out on CPU. Therefore, optimizing the system to fully leverage GPU capabilities will be the future steps to reduce the execution time. Due to the COVID-19 pandemic, we were limited by the ability to integrate our system for carrying out human participants-based study for evaluating SoE (Sense of Embodiment) with the proposed approach for first-person perspective. We will take that (human participant study) also as a future work.

VI. Acknowledgements

This material is based upon work supported by the US Army Research Office (ARO) Grant W911NF-17-1-0299. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the ARO.

References

[1] M. Dou, S. Khamis, Y. Degtyarev, P. Davidson, S. R. Fanello, A. Kowdle, S. O. Escolano, C. Rhemann, D. Kim, J. Taylor *et al.*, "Fusion4d: Real-time performance capture of challenging scenes," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–13, 2016.

[2] K. Kilteni, R. Groten, and M. Slater, "The sense of embodiment in virtual reality," *Presence: Teleoperators and Virtual Environments*, vol. 21, no. 4, pp. 373–387, 2012.

[3] G. Gorisse, O. Christmann, E. A. Amato, and S. Richir, "First-and third-person perspectives in immersive virtual environments: Presence and performance analysis of embodied users," *Frontiers in Robotics and AI*, vol. 4, p. 33, 2017.

[4] H. G. Debarba, S. Bovet, R. Salomon, O. Blanke, B. Herbelin, and R. Boulic, "Characterizing first and third person viewpoints and their alternation for embodied interaction in virtual reality," *PloS one*, vol. 12, no. 12, 2017.

[5] K. Bahirat, T. Annaswamy, and B. Prabhakaran, "Mr. mapp: Mixed reality for managing phantom pain," in *Proceedings of the 2017 ACM on Multimedia Conference*. ACM, 2017, pp. 1558–1566.

[6] K. Bahirat, Y.-Y. Chung, T. Annaswamy, G. Raval, K. Desai, B. Prabhakaran, and M. Riegler, "Using mr. mapp for lower limb phantom pain management," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 1071–1075.

[7] V. S. Ramachandran and D. Rogers-Ramachandran, "Synaesthesia in phantom limbs induced with mirrors," *Proceedings of the Royal Society of London. Series B: Biological Sciences*, vol. 263, no. 1369, pp. 377–386, apr 1996. [Online]. Available: https://doi.org/10.1098/rspb.1996.0058

[8] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM transactions on graphics (TOG)*, vol. 34, no. 6, pp. 1–16, 2015.

[9] K. Otte, B. Kayser, S. Mansow-Model, J. Verrel, F. Paul, A. U. Brandt, and T. Schmitz-Hübsch, "Accuracy and reliability of the kinect version 2 for clinical measurement of motor function," *PloS one*, vol. 11, no. 11, 2016.

[10] S. Rusinkiewicz and M. Levoy, "Efficient variants of the icp algorithm," in *Proceedings third international conference on 3-D digital imaging and modeling*. IEEE, 2001, pp. 145–152.

[11] R. Mekuria, K. Blom, and P. Cesar, "Design, implementation, and evaluation of a point cloud codec for tele-immersive video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 4, pp. 828–842, 2016.

[12] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 2011, pp. 127–136.

[13] S. Orts-Escolano, C. Rhemann, S. Fanello, W. Chang, A. Kowdle, Y. Degtyarev, D. Kim, P. L. Davidson, S. Khamis, M. Dou *et al.*, "Holoportation: Virtual 3d teleportation in real-time," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, 2016, pp. 741–754.

[14] M. Dou, P. Davidson, S. R. Fanello, S. Khamis, A. Kowdle, C. Rhemann, V. Tankovich, and S. Izadi, "Motion2fusion: real-time volumetric performance capture," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, p. 246, 2017.

[15] R. Du, M. Chuang, W. Chang, H. Hoppe, and A. Varshney, "Montage4d: Interactive seamless fusion of multiview video textures," 2018.

[16] L. Xu, Z. Su, L. Han, T. Yu, Y. Liu, and F. Lu, "Unstructuredfusion: Realtime 4d geometry and texture reconstruction using commercialrgbd cameras," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[17] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image," in *European Conference on Computer Vision*. Springer, 2016, pp. 561–578.

[18] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 975–10 985.

[19] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons-Moll, "Video based reconstruction of 3d people models," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[20] T. Alldieck, M. Magnor, W. Xu, C. Theobalt, and G. Pons -Moll, "Detailed human avatars from monocular video," in *2018 International Conference on 3D Vision (3DV)*. IEEE, 2018, pp. 98–109.

[21] W. Xu, A. Chatterjee, M. Zollhöfer, H. Rhodin, D. Mehta, H.-P. Seidel, and C. Theobalt, "Monoperfcap: Human performance capture from monocular video," *ACM Transactions on Graphics (ToG)*, vol. 37, no. 2, pp. 1–15, 2018.

[22] M. Habermann, W. Xu, M. Zollhoefer, G. Pons-Moll, and C. Theobalt, "Livecap: Real-time human performance capture from monocular video," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 2, pp. 1–17, 2019.

[23] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 343–352.

[24] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "Volumedeform: Real-time volumetric non-rigid reconstruction," in *European Conference on Computer Vision*. Springer, 2016, pp. 362–379.

[25] T. Yu, K. Guo, F. Xu, Y. Dong, Z. Su, J. Zhao, J. Li, Q. Dai, and Y. Liu, "Bodyfusion: Real-time capture of human motion and surface geometry using a single depth camera," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 910–919.

[26] T. Yu, Z. Zheng, K. Guo, J. Zhao, Q. Dai, H. Li, G. Pons-Moll, and Y. Liu, "Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7287–7296.

[27] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu, "Real-time geometry, albedo, and motion reconstruction using a single rgb-d camera," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, p. 1, 2017.

[28] M. Zollhöfer, P. Stotko, A. Görlitz, C. Theobalt, M. Nießner, R. Klein, and A. Kolb, "State of the art on 3d reconstruction with rgb-d cameras," in *Computer graphics forum*, vol. 37, no. 2. Wiley Online Library, 2018, pp. 625–652.

[29] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.