Real-time 3D Reconstruction for Haptic Interaction

Timothy Mulumba, Mohamad Eid
New York University Abu Dhabi
Applied Interactive Multimedia (AIM) Laboratory,
Abu Dhabi, United Arab Emirates

***Abstract*—Accurate, robust and fast 3D reconstruction of objects in real-life scenes is a challenging task, studied by a variety of scientific communities. Such 3D models have a wide variety of applications ranging from entertainment, education and training, to cultural heritage, which is of special interest given the plethora of museums in the United Arab Emirates (UAE). This paper presents a novel classification of studies that perform 3D reconstruction of objects and a pilot study for enabling haptic interaction with said objects. Whereas most of the studies in the literature follow a familiar pattern; i.e. starting with a set of images from a scene, through the application of a mathematical algorithm, and finally to the rendering of a 3D model, fitness for haptic applications drives the proposed classification. The classification scheme includes: i) the mathematical algorithm, ii) Real-time vs non real-time, iii) Single object vs multi-object, and iv) Single vs multi sensors. An experimental study is conducted using real image sequences of a simple, single-object scene. Reconstruction in this experiment is achieved by the fusion of successive images, captured by an inexpensive depth sensor, specifically Microsoft's Kinect device. Furthermore, multimodal interaction is achieved by enabling haptic interaction with the reconstructed object.**

***Keywords*—3D reconstruction, Kinect sensor, haptic interaction, Omni**

I. Introduction

There is a concerted effort to bring to fruition a world of virtual museums, with lots of sculptures, paintings and other cultural heritage items. Imagine if the wonder of the Louvre, the majesty of the pyramids and the splendor of the Inca ruins was all at your fingertips. The magnitude of experiencing all these adventures first hand may not be supplanted by a 3D model in a virtual world but it is hoped that the virtual world where the object can be seen, magnified and examined from multiple view-points could at least rival this natural experience. In addition, archaeologists and students could carefully examine and document these objects, make experiments on them, place them in a new, or in the original ancient environment, as is usually the case in augmented reality, or complete them with missing parts: possibilities abound. Furthermore, advances in haptic technologies, a novel dimension of physical interaction (via the sense of touch) could soon be added to this virtual world.

Creating realistic 3D models of real world objects is a well researched problem in computer vision and computer graphics. These models can be used not only for virtual museums, but also in medicine for surgical simulations, in periodontal clinics for training inexperienced dentists, for architecture and entertainment (cartoons, movies, computer games). Such models require both precise geometry and detailed texture on the surface.

In this paper, we aim at providing a broad overview of the studies on 3D image reconstruction. Our goal is to provide a classification of existing studies that allow the dissection and comparison of the different components and design decisions with respect to speed, sensor selection, scene type and algorithm deployed. The ultimate goal is to adapt existing literature to support haptic interaction with real-time 3D reconstructed environments. Unlike other senses, haptics (tactile and kinesthetic) is not only a sensory channel to receive information, but it is also a channel for expressiveness through actions [1]. Incorporating haptics into multimedia systems has created a wide spectrum of interactive applications ranging from medical simulations [2] to gaming [3], tele-operation and interpersonal communication. We restrict our survey to studies in which the input to the method is a set of digital images (one, two or more) or video. This is referred to as image-based reconstruction and the output is a 3D model.

Our task, therefore, is to paint previous studies on 3D image reconstruction with a rather broad brush in so far as their characteristics pertain to our overarching goal, which is to obtain a 3D representation of the physical environment, enabling manipulation of the digitized multimodal contents via haptic modality.

The remainder of this paper is organized as follows: In Section II we describe related work, in Section III, we present our proposed classification scheme for the studies in the literature. In Section IV, we present a case study that extends representative work in the literature by enabling haptic interaction with the reconstructed environment. We discuss the results in Section V, and finally we conclude in Section VI with a discussion of future planned work.

II. Related Work

Haptically interfacing with 3D models can be done either directly by providing haptic information of objects in the scene on the fly or indirectly by having pre-recorded haptic information in a database.

In [4], a method for real-time haptic interaction with RGB-D streams is presented. Forces are computed using depth information from Microsoft's Kinect device while the color stream is visually rendered. Additionally, fast collision detection ensures the proposed approach can be used in real-time. The interaction, however, is 2.5D since haptic feedback is only provided for visible surfaces and is unavailable for the reverse faces.

Contribution

Our contribution is two-fold; first we present a classification scheme for 3D reconstruction from a haptic interaction perspective since different studies emphasize different aspects of haptic interaction. We examine those that prioritize speed over quality and vice versa, and those that find a trade-off. We motivate that attention should be paid to single object 3D reconstruction due to the nature of haptic interaction and finally, we highlight multi-sensor studies mainly because of our requirement for unencumbered interaction.

Second, we identify an existing 3D reconstruction algorithm that meets all our haptic design requirements and present a preliminary implementation of haptic interaction on top of that 3D reconstruction.

III. CLASSIFICATION OF 3D RECONSTRUCTION LITERATURE

In order to support a comparison of studies on 3D object reconstruction, we propose a categorization scheme for such studies.

A. Overview of Classification Scheme

We present a set of design decisions that most studies have to consider. The proposed classification is based, therefore, on the observation that research on 3D object reconstruction generally considers the following components:

1) **Mathematical algorithm:** Algorithms for 3D reconstruction are often referred to as "Shape from X" depending on the particular cue used e.g. shape from shading, shape from stereo etc.
2) **Single object vs multi-object:** Single object scenes are usually easier to reconstruct than multi-object scenes because the interaction between objects in multi-object scenes necessitate more complex algorithms to account for occlusion.
   We assert that the single object classification could further be sub-divided into deformable and non-deformable objects.
3) **Real-time vs non real-time:** Different applications have different requirements with respect to speed of reconstruction. It is also incumbent upon us to define the demarcation between 'real time' and 'non real time' applications. Such a demarcation is subjective, as one might suspect, but for us with a haptic world view, real time means an execution time of less than a hundred milliseconds [5].
4) **Single vs multi sensors:** Single sensors are invariably used to scan around a scene, usually scenes that comprise a single object. The use of a single, static sensor is also possible but this only results in a 2.5D image since some part of the object would be out of range. Multi sensors on the hand are typically used to reconstruct 3D environments using multiple view-points.

It should be noted that most studies are multi-faceted and many cover all of the aspects listed above. However, it's our contention that a study that presents a novel algorithm for 3D object reconstruction is different from one that applies an existing algorithm to a different problem by modifying the setup/ scenario e.g. for haptic interaction.

B. Mathematical Algorithms

The main 3D reconstruction methods for inferring 3D models from 2D images are shape from stereo and shape from motion.

1) : Shape from stereo draws its inspiration from human binocular vision. Stereo algorithms take a pair of images and use the displacement of corresponding image features in order to estimate depth (distance from the camera). The output is often in the form of a depth map, which contains a depth value for each pixel in the input images [6].

2) : Shape from motion (more commonly referred to as structure from motion, or multi-view stereo) is a related field to shape from stereo. The movement of either the object or camera provides the information for reconstruction via the influence of perspective. Early approaches to structure from motion were based on analytical geometry. Two views of the same scene are related to a quantity known as the fundamental matrix, which describes the implicit geometry [7]. Use of the fundamental matrix became popular particularly for studying the uncalibrated case since it is dependent on 2D image observations alone. From this matrix, 3D shape can be recovered up to a projective transformation.

C. Single object vs multi-object

When dealing with single compact objects the visual hull provides a very useful constraint of the shape of an object. In [8], a stochastic algorithm is presented to improve surface provided by the visual hull and recover a representation of the objects colour/texture in the form of a texture map.

In a large number of 3D vision algorithms, the formulation of reconstruction is as an optimisation problem, where a shape is sought which has a maximal consistency with the input images. Optimisation by graph cuts, popular in stereo vision has also been applied to the problem of 3D surface reconstruction in a method presented in [9].

With respect to deformable reconstruction, [10] and [11] present novel non-rigid reconstruction pipelines using extended non-linear frameworks.

In [12], a key contribution towards multi-object modelling is that their method attempts to identify specular surfaces. Recently, Dai et al. [13] attempt to deal with multiple issues that plague 3D reconstruction including high quality surface remodeling, scalability, robust camera tracking, interactivity and real time rates.

One characteristic of multi-object scenes is the increased likelihood that there will be one or more non-lambertian surface somewhere in the scene. Another issue with highly detailed models is that the data is often too large to fit in the RAM of the machine [12].

D. Real-time vs non real-time

In [14], a viewpoint-based approach for quick fusion of multiple depth maps is presented. Depth maps are computed

in real-time from a set of images captured by moving cameras, using plane-sweeping stereo. The depth maps are fused within a visibility-based framework and the methodology is applied for view-point rendering of outdoor large-scale scenes.
In state-of-the-art work [15], [16], a high quality system for 3D applications is described. The authors include a method for the creation of highly accurate textured meshes from a stereo pair. Exploiting multiple stereo pairs, they generate multiple meshes in real-time, which are intelligently combined to synthesize high-quality intermediate views for given view-points. Recently, Dou et al. [17] present a new method for real-time high quality 4D (i.e. spatio-temporally coherent) performance capture, allowing for incremental nonrigid reconstruction from noisy input from multiple RGBD cameras. Their system demonstrates reconstructions of challenging nonrigid sequences, at real-time rates, including robust handling of large frame-to-frame motions and topology changes. In [11], Zollhofer et al. present a system that enables the real-time capture of general shapes undergoing non-rigid deformations using a single depth camera.
Methods that could be classified as non real-time require relatively high accuracy which translates into significant computation times. Examples of these include [12], [18] which have been described in Section III-C under the multi-object design consideration.

E. Single vs multi sensors

In [19], personalized avatars are created from a single RGB image and the corresponding depth map, captured by a Kinect sensor. In [20], the authors present an efficient system for scanning and off-line generating human body models using three Kinects and a rotating table. In [21], the problem of dynamic 3D indoor scenes reconstruction is addressed through the fast fusion of multiple depth scans, captured by a single hand-held Kinect sensor. In [22], an efficient system with six Kinects is presented. It deals with two important elements of a Tele-Presence system, 3D capturing and reconstruction and view-point-dependent rendering, demonstrating a good performance in terms of visual quality.
Recently, Du et al. [23] introduced a novel web-based interactive system to create, calibrate, and render dynamic video-based virtual reality scenes in head-mounted displays, as well as high-resolution wide-field-of-view tiled display walls. In [10], presented a novel approach for the reconstruction of dynamic geometric shapes using a single hand-held consumer-grade RGB-D sensor at real-time rates.

IV. Haptic Interaction

In this section, we experiment with our overarching goal of multimodal interaction in virtual environments. The experimentation process flow is summarized in Figure 1.

The process begins with a sensor that captures 2D images and/ or video; through a graphics loop that converts these 2D images to 3D models; to a haptic modeling that generates a haptic model and finally to haptic rendering that enables haptic interaction. These steps are explained in detail in the sections that follow.

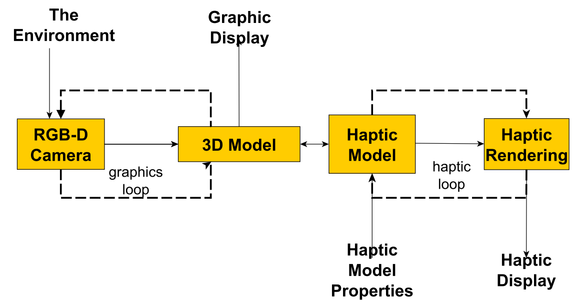

Fig. 1. Experiment process flow

A. Graphics Rendering

We utilized the work of Izadi et al. [21] whose solution is the basis for a software package known as 'Kinetic Fusion SDK (Software Development Kit)'. Whereas other methods using 3D scanners provided higher quality output but with slow reconstruction speeds, this study aimed at producing a quality 3D model with a fast reconstruction speed. The desirable computation speed is achieved by using a gaming card (GPU: Graphics Processor Unit) to offload computation. The idea was that even though GPU cores ($\approx$ 600 MHz) are slower than CPU cores ($\approx$ 2 GHz), there are many more GPU cores than CPU cores in a typical computer (500 vs 6). Additionally, GPU cores are specialised and are especially well suited to parallel, pixel-based calculations.
In this section we describe the hardware setup and give a brief overview of the adopted reconstruction approach.

1) The Kinect sensor (version 2): Kinect is a line of motion sensing input devices by Microsoft for Xbox 360 and Xbox One video game consoles and Windows PCs. Based around a webcam-style add-on peripheral, it enables users to control and interact with their console/computer without the need for a game controller, through a natural user interface using gestures and spoken commands. It is a composite sensor i.e. it provides depth, color (RGB), infrared and audio information. The Kinetic driver is based on a stereo matching algorithm and outputs an RGB image with a resolution of 1920 x 1080 at 30 frames per second and a corresponding depth image resolution of 512 x 424 [24].

2) Graphics loop hardware components: The setup is as depicted in Figure 2. The method runs in real time on a single host PC with an Intel Xeon processor (2.6 GHz, 6 cores) and 32 GB installed RAM, along with a graphics card NVidia Quadro K5000.
We also note here that the setup comprises a single Kinect sensor that we use variably as fixed sensor-moving object or moving sensor-fixed object.

3) Overview of the Kinect Fusion solution: The main stages of Kinect Fusion are depicted in Figure 3:

(a) **Depth Map Conversion:** The depth map from the Kinect is converted into a 3D point cloud.
(b) **Camera Tracking:** This is done by applying the Iterative

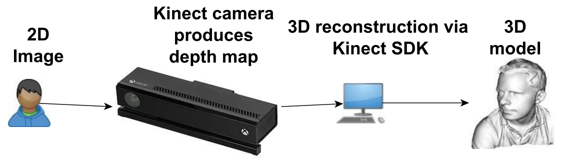

Fig. 2. The main hardware components

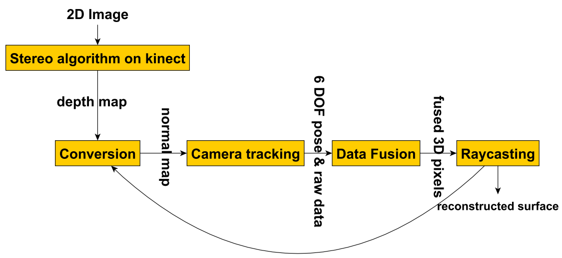

Fig. 3. Overview of kinect fusion tracking and reconstruction [21].

Closest Point (ICP) algorithm. This algorithm assumes that point clouds are roughly aligned. In the Kinect fusion case, point clouds are roughly aligned because of the camera's fast frame rate (30 fps). ICP works as follows:

- An association of points between successive frames is determined.
- Distance and angle compatibility is calculated given the association above. If too far away, the point is marked as an outlier.
- An energy function (sum of squared distances between points) is then minimized.
- An offset transformation between the point clouds is acquired and applied.
- Previous steps are iterated,

(c) **Volume Data Integration:** Once the 6 DOF orientation of the camera is known i.e. we know where the current frame lies in relation to previous, a global model is then integrated. Implicit surfaces are modeled with the Truncated Signed Distance Function (TSDF) of Curless & Levoy [25]. Instead of triangles and polygons, voxels, which are like 3D arrays, are used. Voxels within a certain distance to the probable surface store signed (+/-) distance values to the surface.

(d) **Raycasting:** Finally, the volume is raycast to extract views of the implicit surface, for rendering to the user.

B. Haptic Rendering

The goal of this section is to enable haptic interaction with the 3D reconstructed object. The OpenHaptics 3.0 Haptic Library is utilised to render interaction with the Geomagic Touch Device (formerly Phantom Omni).

Touch is a motorized device that applies force feedback on the user's hand, allowing them to feel virtual objects and producing true-to-life touch sensations as user manipulates on-screen 3D objects.

The Touch model is portable and IEEE-1394a FireWire port interface ensures quick installation and ease of use.

1) The experimental setup: The setup is as depicted in Figure 4. The PC is the same as that described in Section IV-A under the graphics loop.

In addition, we note that the IDE used for development was the Microsoft Visual Studio 2010 Express version.

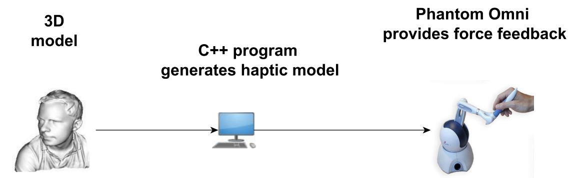

Fig. 4. The Haptics loop setup

2) Overview of the haptic solution: The main stages of the haptic solution are depicted in Figure 5:

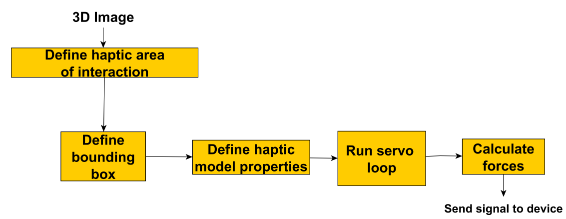

Fig. 5. Overview of the haptic solution

(a) **Define bounding box:** Collisions between complex objects are simplified by the use of bounding volumes. The object is embedded into the smallest possible bounding volume.

(b) **Define properties of haptic model:** Properties include stiffness (κ) of a virtual object and whether the object is grounded or not. Once this is determined, then surface texture can easily be abstracted away to the material of the object or by adjusting a friction coefficient (μ).

(c) **Collision detection:** This is one of the key tasks in haptic interaction with virtual environments. The pose (position, orientation) of the tool's Haptic Interaction Point (HIP) is computed and possible collisions with objects in the virtual environment are determined.

(d) **Run servo loop:** The servo loop is a tight control loop used to calculate forces to send to the haptic device. In order to render stable haptic feedback, this loop must be executed at a consistent 1 kHz or better. In order to maintain such a high update rate, the servo loop is executed in a separate, high priority thread.

The magnitude of the reaction force can be computed based on a simple assumption that the force is proportional to the penetration depth. We can further assume a frictionless surface so that the reaction force is determined with a vector normal to the surface of the 3D object at the point of contact. The most common way of modeling a stiff and grounded surface is based on a model consisting of a parallel connection of a spring with stiffness K and a damper with viscosity B [26]. Given $x = 0$ for an undeformed object and $x < 0$ inside the object boundaries, the modeled contact force equals:

$$F = \begin{cases} -(Kx + Bv), & x < 0, v < 0 \\ -Kx, & x < 0, v \geq 0 \\ 0, & x \geq 0 \end{cases} \quad (1)$$

In this case, the viscous damping behaves as a directed damper, that is active during the penetration into the object and passive during the withdrawal from the object. This enables a stable and damped contact with the object as well as a realistic contact rendering.

These steps are detailed in Algorithm 1 below:

Algorithm 1

Initialize haptic device and enable force output;
if Graphics loop is active; **then**
    Start scheduler, synchronize haptic and graphic loops;
    Begin haptic frame and get device position;
    for all Virtual objects O_i **do**
        Compare device position with the virtual object
        if Collision detected; **then**
            Calculate reaction force;
            Send force to haptic interface;
        end if
    end for
else
    Stop scheduler and disable haptic device;
end if

V. Results and Discussion

Using the setup in Figure 2, we proceeded to create a 3D model using the Microsoft Kinect. An example is depicted in Figure 6. The reconstruction was real-time (almost instantaneous), with the focus on a single object (human being).

Additionally, using the setup in Figure 4, we proceeded to setup haptic interaction with the virtual object. Of particular importance is the ability to setup multiple windows with different views which is especially convenient for teleoperation. One of these views is depicted in Figure 7. In this interaction a user is reconstructed while another interacts haptically with the reconstructed image, ostensibly in a remote location.

Robustness of the haptic interaction with the reconstructed object relies heavily on the fullness of reconstruction of the surface of the 3D object. Since the forces generated by the haptic device are proportional to the depth of penetration of the Haptic Interaction Point (HIP) into the surface of a reconstructed object, then if an object is defined as rigid but has discontinuities in its reconstructed surface, then the haptic tool would behave erratically if one of these discontinuities became the point of interaction. The situation is further exacerbated by the fact that the force rendered follows Hooke's law and therefore greater displacements lead to greater forces generated. We get around this by assigning a maximum force that the tool can haptically render.

Fig. 6. 3D model output

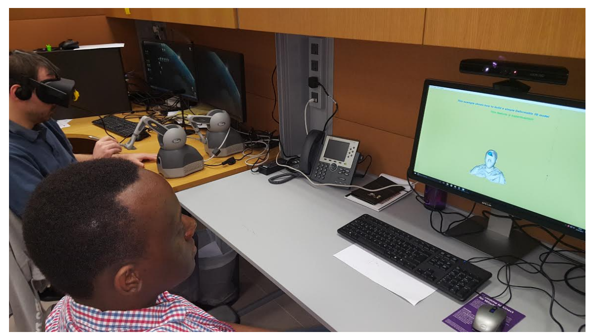

Fig. 7. Haptic interaction

Additionally, the force feedback applied to the haptic device should avoid sudden changes in magnitude and direction. This is achieved by gradually increasing or decreasing the force feedback except at the point of collision between the HIP and the virtual object.

The refresh rate for displaying forces on the haptic device is more than an order of magnitude higher than the refresh rate necessary for displaying images on the screen. As a result, the respective loops (graphics and haptics) are performed concurrently but in separate threads.

Finally, state synchronization of the two loops is important since both loops need access to the same info. The solution is to share the database of information and ease programming with the added burden of paying more attention to synchronization efforts (multi-threading).

In terms of performance, the 3D model is almost real-time with the delay only coming from rotating the object for full 3D coverage. The haptics loop shows a more noticeable delay at the first synchronization run, with a delay averaging 5s. All delays after this initial synchronization are under 1ms which is considered ideal for haptic interaction.

VI. Conclusion

This paper presents a novel classification for 3D image-based reconstruction. This classification has been based on design considerations and components that are a common thread within these studies. Given the design considerations for haptic rendering a study is presented with many of the properties necessary for haptic rendering. We provide an overview of this study and present a preliminary implementation of haptic interaction on top of an existing 3D reconstruction algorithm. Some future work includes adding texture to the haptic models to make them even more life-like. Another consideration could be allowing for deformable objects which are of particular interest for several applications.

References

[1] M. Eid, M. Orozco, and A. El Saddik, "A guided tour in haptic audio visual environments and applications," *International Journal of Advanced Media and Communication*, vol. 1, no. 3, pp. 265–297, 2007.

[2] A. M. Genecov, A. A. Stanley, and A. M. Okamura, "Perception of a haptic jamming display: Just noticeable differences in stiffness and geometry," in *2014 IEEE Haptics Symposium (HAPTICS)*. IEEE, 2014, pp. 333–338.

[3] J. Cha, M. Eid, L. Rahal, and A. El Saddik, "Hugme: An interpersonal haptic communication system," in *Haptic Audio visual Environments and Games, 2008. HAVE 2008. IEEE International Workshop on*. IEEE, 2008, pp. 99–102.

[4] S. Rasool and A. Sourin, "Real-time haptic interaction with rgbd video streams," *The Visual Computer*, pp. 1–11, 2016.

[5] M. Eid, J. Cha, and A. El Saddik, "Admux: An adaptive multiplexer for haptic–audio–visual data communication," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 1, pp. 21–31, 2011.

[6] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International journal of computer vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[7] Q.-T. Luong and O. D. Faugeras, "The fundamental matrix: Theory, algorithms, and stability analysis," *International journal of computer vision*, vol. 17, no. 1, pp. 43–75, 1996.

[8] J. Isidro and S. Sclaroff, "Stochastic refinement of the visual hull to satisfy photometric and silhouette consistency constraints," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 1335–1342.

[9] G. Vogiatzis, P. H. Torr, and R. Cipolla, "Multi-view stereo via volumetric graph-cuts," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2. IEEE, 2005, pp. 391–398.

[10] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "Volumedeform: Real-time volumetric non-rigid reconstruction," *arXiv preprint arXiv:1603.08161*, 2016.

[11] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt *et al.*, "Real-time non-rigid reconstruction using an rgb-d camera," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, p. 156, 2014.

[12] C. J. Poelman and T. Kanade, "A paraperspective factorization method for shape and motion recovery," *IEEE transactions on pattern analysis and machine intelligence*, vol. 19, no. 3, pp. 206–218, 1997.

[13] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt, "Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface re-integration," *arXiv preprint arXiv:1604.01093*, 2016.

[14] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nistér, and M. Pollefeys, "Real-time visibility-based fusion of depth maps," in *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

[15] R. Vasudevan, G. Kurillo, E. Lobaton, T. Bernardin, O. Kreylos, R. Bajcsy, and K. Nahrstedt, "High-quality visualization for geographically distributed 3-d teleimmersive applications," *IEEE Transactions on Multimedia*, vol. 13, no. 3, pp. 573–584, 2011.

[16] R. Vasudevan, Z. Zhou, G. Kurillo, E. Lobaton, R. Bajcsy, and K. Nahrstedt, "Real-time stereo-vision system for 3d teleimmersive collaboration," in *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1208–1213.

[17] 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, p. 114, 2016.

[18] T. Yu, N. Xu, and N. Ahuja, "Shape and view independent reflectance map from multiple views," *International journal of computer vision*, vol. 73, no. 2, pp. 123–138, 2007.

[19] M. Zollhöfer, M. Martinek, G. Greiner, M. Stamminger, and J. Süßmuth, "Automatic reconstruction of personalized avatars from 3d face scans," *Computer Animation and Virtual Worlds*, vol. 22, no. 2-3, pp. 195–202, 2011.

[20] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan, "Scanning 3d full human bodies using kinects," *IEEE transactions on visualization and computer graphics*, vol. 18, no. 4, pp. 643–650, 2012.

[21] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison *et al.*, "Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera," in *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 2011, pp. 559–568.

[22] A. Maimone and H. Fuchs, "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*. IEEE, 2011, pp. 137–146.

[23] R. Du, S. Bista, and A. Varshney, "Video fields: fusing multiple surveillance videos into a dynamic virtual environment," in *Proceedings of the 21st International Conference on Web3D Technology*. ACM, 2016, pp. 165–172.

[24] E. Lachat, H. Macher, M. Mittet, T. Landes, and P. Grussenmeyer, "First experiences with kinect v2 sensor for close range 3d modelling," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 40, no. 5, p. 93, 2015.

[25] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 1996, pp. 303–312.

[26] J. Podobnik *et al.*, *Haptics for virtual reality and teleoperation*. Springer Science & Business Media, 2012, vol. 67.