REAL-TIME 3D VOLUMETRIC HUMAN BODY RECONSTRUCTION FROM SINGLE VIEW RGB-D CAPTURE DEVICE

Rafael Diniz, Mylène Farias

University of Brasília

ABSTRACT

Recently, volumetric video based communications have gained a lot of attention, especially due to the emergence of devices that can capture scenes with 3D spatial information and display mixed reality environments. Nevertheless, capturing the world in 3D is not an easy task, with capture systems being usually composed by arrays of image sensors, sometimes paired with depth sensors. Unfortunately, these arrays are not easy assembly and calibrate to non-specialists for use, making their use in volumetric video applications a challenge. Additionally, the cost of these systems is still high, which limits their popularity in more mainstream communication applications. This work proposes a system that provides a way to reconstruct the head of a human speaker from single view frames captured using a single RGB-D camera (a Microsoft's Kinect 2 device). The proposed system generates volumetric video frames with a minimum number of occluded areas and missing parts. To achieve a good quality, the system prioritizes the data corresponding to the participants' face, therefore preserving important information from speaker's facial expressions. Our ultimate goal is to design an inexpensive system that can be used in volumetric video telepresence applications and even on volumetric video talk-show broadcasting application.

***Index Terms*—** 3d telepresence, 3d immersion, point cloud, point cloud registration, object reconstruction, volumetric video

1. INTRODUCTION

Currently, there a several volumetric video systems, like for example the 8i [1] and the Microsoft's systems [2]. Instead of using video-based processing techniques, some real-time 3D volumetric video systems use either a mesh or a point-cloud format to represent the 3D objects, which allows for a full 3D representation of the objects in the scene. Unfortunately, given the order and topology of the 3D-frames, the computational complexity of these systems is high [3]. In other words, the acquisition of a complete and accurate 3D representation of scene objects using the current 3D systems is not an easy task, given that these systems are usually composed by an array of calibrated sensors. It is worth pointing out that, unlike in regular 2D video applications, the 3D objects need to be reconstructed after the the acquisition.

Volumetric scene completion from single view RGB-D capture is also proposed by [4], which uses large categorized 3D models for object shape retrieval. Song [5] proposed a semantic scene completion, which uses a 3D convolutional neural network trained with large 3D scene datasets (E.g. 45,622 houses with 775,574 rooms). Yang [6] uses a generative adversarial network to reconstruct objects from a single view capture and uses large databases to train the network, and claims to be the state of art in its class. Instead of using a deep learning approach, our work relies only on fast geometric transformations, so it is fast and does not necessarily require a powerful GPU, like the deep learning implementations require. Yang proposes the highest resolution among the deep learning based volumetric reconstruction methods of 256^3 voxel space, while we propose to support at least a 1024^3 voxel space and more than one million of voxels, taking as reference the volumetric video test material sent ISO/IEC/MPEG Point-Cloud Compression group by 8i [1], which has frames of up to 1024^3 voxel resolution and typically more than one million occupied voxels. Voxelized point-cloud is a type of point-cloud used to represent solid objects, where each element is small cube, called voxel, analog to the pixel for 2D images dimension [7].

The proposed framework for volumetric reconstruction first creates a complete volumetric representation of each person's head which will join a volumetric video session. For this, we use a methodology based on the Truncated Signed Distance Function [8] and Kinect Fusion [9], which assembles the volumetric 3D object representation by moving the capture device around the object (in this case, a person). Then, the volumetric model of the person's head is stored, in a point-cloud format.

With the volumetric model of the person's head stored, and assuming that (1) the back of the head of the person is non-deformable; (2) the speaker is looking ahead during most of the time, allowing the RGB-D camera to capture the mouth and eyes of the speaker; and (3) self-occlusions do not occur often, we reconstruct the whole head of a speaker in order to be used for 3D telepresence. This way higher dynamics

Thanks to CAPES agency for funding.

of the object (mouth, nose, eyes) are fully present in the reconstructed 3D volumetric stream, and instead of methods specific for the human body (eg. [10]), our proposal can be extended to many types of objects where the dynamics are mostly present in just one face of the object.

The big advantage of the proposed reconstruction system is its simplicity in the capture setup. The proposed system uses for volumetric capture just one RGB-D capture device to reconstruct a 3D representation of a human figure (speaker), greatly simplifying the acquisition procedure in applications like mixed-reality volumetric video teleconferences.

The proposed solution can be used not only in live two-way communication, but also in volumetric video broadcasting and Internet volumetric video services where the a person is giving a speech, giving a class, presenting the news or playing an online game in volumetric video format. The rest of this article is composed by a section which describes the proposed system, followed by the results and finally by a section with the conclusions obtained.

2. PROPOSED SYSTEM

This section describes the proposed framework, and is composed of the following subsections: 3D model capture, 3D reconstruction and implementation and experimental setup.

2.1. 3D Model Capture

The proposed framework first creates a complete volumetric representation of the upper body of a person. For this, we use a methodology based on the Truncated Signed Distance Function [8] and Kinect Fusion [9], which assembles the complete 3D object by moving the capture device around the object. Figure 1 shows an example of a captured model.

Fig. 1. 3D captured model.

Not only the model is captured and stored in point-cloud format, but also two segmented parts of it are extracted and stored. One point-cloud extracted from the model is nose information and its neighboring region, including the eyes. The other stored point-cloud is the back of the head. The segmentation process uses maximum or minimum depth heuristics (depending on coordinate system) to identify each region. Figure 2 shows the segmented point-clouds extracted from a model.

Fig. 2. Nose and adjacency (bottom) and back of the head (top) point-clouds extracted from the model.

The 3D model is used in the reconstruction process where it is registered and merged to the point-cloud frame captured live from a Kinect device.

2.2. 3D reconstruction

After the model is captured, the proposed volumetric object reconstruction from a live capture system can start. A preprocessing is performed on each captured RGB and Depth frame pair:

- For each RGB and Depth frame pair captured, an alignment is necessary because the timestamps of the color and depth frames differ between 10ms to 20ms, which although less than a frame period ($\sim$33ms at 30fps);
- The RGB and Depth frames are converted to point-cloud, with camera coordinates converted to world coordinates, using Kinect's intrinsic parameters, as shown is Figure 3;

Fig. 3. Point-cloud from live single view capture.

Then the volumetric object reconstruction is performed as follows:

- The point-cloud obtained from the live feed has its nose and adjacent areas segmented;

- A transformation matrix is computed between the segmented "face" from the model and the segmented input point-cloud, using a fast global registration method that is computed;
- The segmented "back of the head" from the model is transformed using the computed transformation matrix;
- Finally, the transformed 3D model and live captured point-cloud "nose areas" are merged and the live reconstructed volumetric video frame is created.

The algorithm can be described in the following steps: (1) model is captured and stored in point-cloud format, as previously explained. Then the model is segmented (2), with the front face of the human head and the back of the head stored separately, by the application of a method which uses the depth information from the sensor as heuristic to identify the regions. This function is used to further optimize the computation efficiency of the registration between the model and the input live volumetric stream. After the initial two mentioned steps, for each captured pair of RGB and Depth frames, a step (3) is performed to align the captured RGB and Depth frames, because we realized that the timestamps of the color and depth frames differ in the range of 10ms to 20ms, which is less then a frame period which is approximate 33ms at 30fps, but still makes the registration of the RGB and Depth frames important, especially for high speed movements. The aligned RGB and Depth frames, together with the camera parameters, are converted (4) to point-cloud format. Then, in a similar approach of step 2, the point-cloud obtained from the live feed has its nose and adjacency segmented (6). Then, by the application of a fast global registration method [11] (7) between the segmented model and the segmented input point-cloud, a transformation matrix is obtained. Finally (8), the back of the head from the model is merged with the segmented live point-cloud input frame. Figure 4 shows the reconstructed point-cloud.

Fig. 4. Reconstructed point-cloud using the proposed reconstruction from single view RGB-D input.

Using this framework, the higher dynamics of the object (changes in the mouth, nose, and eyes regions) can be fully represented in the reconstructed volumetric video stream. The proposed system aims to generate volumetric video frames with a minimum number of occluded areas and missing body parts.

At the heart of our approach is a fast global registration algorithm [11] which aligns a pre-processed 3D model to a pre-processed point-cloud obtained directly from the RGB-D frame from Kinect. Then the model aligned with the live single view 3D input are merged to re-create the head of the telepresence participant.

2.3. Implementation and Experimental Setup

The proposal is implemented in C and C++[1], and the code contains operations to perform the basic point cloud operations, like geometric transformations, point to point distance, crop and merge. The following libraries were used for the development of this work: OpenKinect's project libfreenect and libfreenect2 [12], for Kinects support, and Open3D [13], for the registration implementation among other useful volumetric data structures Open3D library provides. The proposals were tested in both a high-end computer and a notebook computer. The high-end computer is a dual eight-core (32 SMT[2]) Intel Xeon E5-2620, with 80GB of RAM memory and two video cards, a NVidia Quadro P6000 and a NVidia GeForce GTX 1080. The notebook computer is a Lenovo ThinkPad T430 with a dual-core (4 SMT) Intel Core i5-3320M with 8GB of RAM with an external NVidia GTX 1080 GPU (see fig. 5). The notebook setup is also used when capturing outside the laboratory.

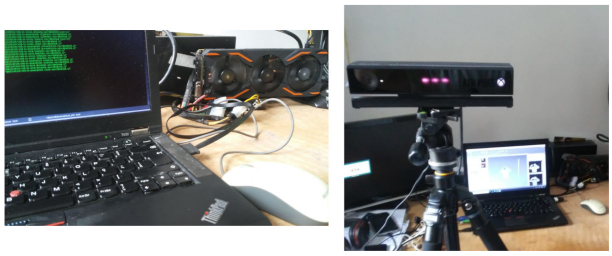

Fig. 5. Partial view of the notebook with an external GPU attached (left) and the Kinect 2 connected to the notebook computer used for capture outside the laboratory (right).

Early development was done using a Kinect 1 device, but the authors decided to proceed the development with the Kinect 2 device, which uses a time-of-flight ranging technology, as opposed to structured light based Kinect 1 [14]. Kinect 2 is also the most widely used RGB-D capture device for volumetric video production. Kinect 2 has a 1920x1080 RGB camera and its time-of-flight range sensor outputs a 512x424 depth frame. The depth sensor support distances of 0.5m to 4.5m, and provides a field-of-view of 70.6° by 60° (HxV), with millimeter accuracy, providing a better accuracy and less noisy output than the structured light based Kinect 1. Nevertheless, code was developed to interact with

[1] Implementation source code available on request.

[2] SMT: Simultaneous multithreading permits current CPUs to share CPU resources among 2 independent threads, improving the overall performance.

both versions of the Microsoft's RGB-D sensor, and works as expected with both devices. The provided RGB frame by Kinect 2 is scaled and chopped to match the 512x424 depth resolution.

3. RESULTS

Our tests show that each captured frame is processed, on average, under 33ms (the CPU used was a 16-core Intel Xeon E5-2620 at 2.1GHz). This acquisition and processing time guarantees that a 30fps input can be processed in realtime.

More specifically, our system is a computationally fast volumetric video system that reconstructs 3D representations of the speaker in real-time using a consumer CPU, with room left for optimization like GPU processing offload. The proposed method allows a better mixed reality experience when compared to incomplete and open volumetric representations produced by using just one RGB-D capture device. Our proposal aims to recreate a complete volumetric representation of each person in the teleconference session.

Similarly to methods that recreates a complete human body (eg. [10]), the proposed framework can be extended to capture different types of objects, for which the changes occur mostly in one side/face of the object.

One contribution of the proposed system is the design of a CPU efficient volumetric video system, which uses state-of-art registration techniques, put together to allow an efficient 3D capture setup for volumetric video communication systems, with real-time 3D object reconstruction from single RGB-D device. Obviously, one main concern of this work is achieved by providing a solution which can capture volumetric video using a very simple setup.

4. CONCLUSIONS

We show that it's possible to create volumetric video using a simple single RGB-D capture device connected to available hardware. Nevertheless, our approach still face challenges. Pre-processing steps made to the point-clouds before the registration step are very important for quality and real-time execution. Also the fast global registration technique incurs, sometimes, in an imperfect transformation matrix.

Concerning the quality of the reconstructed volumetric video frame, It's possible to notice that in the examples there is a small color temperature difference between the model (more sun was coming through the window at the moment of the capture) and the selected RGB-D frame. Also, there are blending artifacts, which can be clearly seem in Figure 6. A blending issue, for example, is found where the hair starts in the head, caused by the under sampling near the edges of an object.

In the case of our implementation of volumetric reconstruction using a single RGB-D capture device, we also realized that, apart of the registration step, an improvement related to lighting changes and the blending of the model with single view frame needs to be addressed. The first problem should be addressed by an algorithm to monitor and compensate light changes. The blending problems will be addressed with the development of a smart voxelization procedure which will correctly blend point clouds into one voxelized [7] point cloud. The blending will be able to prioritize selecting voxels from one point-cloud over another, which is important for our reconstruction method from single RGB-D input and also for any future work with multiple RGB-D capture devices working in parallel.

Fig. 6. Reconstructed volumetric video frame from single RGB-D sensor where it's possible to notice blending artifacts.

5. REFERENCES

[1] E d'Eon, B Harrison, T Myers, and PA Chou, "8i voxelized full bodies, version 2–a voxelized point cloud dataset," *document MPEG*, p. m74006, 2017.

[2] Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L Davidson, Sameh Khamis, Mingsong Dou, et al., "Holoportation: Virtual 3d teleportation in real-time," in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, 2016, pp. 741–754.

[3] Kazuo Sugimoto, Robert A Cohen, Dong Tian, and Anthony Vetro, "Trends in efficient representation of 3d point clouds," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2017*. IEEE, 2017, pp. 364–369.

[4] Jason Rock, Tanmay Gupta, Justin Thorsen, JunYoung Gwak, Daeyun Shin, and Derek Hoiem, "Completing 3d object shape from one depth image," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2484–2493.

[5] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser, "Semantic scene completion from a single depth image," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*. IEEE, 2017, pp. 190–198.

[6] Bo Yang, Stefano Rosa, Andrew Markham, Niki Trigoni, and Hongkai Wen, "3d object dense reconstruction from a single depth view," *arXiv preprint arXiv:1802.00411*, 2018.

[7] Tommy Hinks, Hamish Carr, Linh Truong-Hong, and Debra F Laefer, "Point cloud data conversion into solid models via point-based voxelization," *Journal of Surveying Engineering*, vol. 139, no. 2, pp. 72–83, 2012.

[8] Brian Curless and Marc 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.

[9] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 2011, pp. 127–136.

[10] Dimitrios S Alexiadis, Nikolaos Zioulis, Dimitrios Zarpalas, and Petros Daras, "Fast deformable model-based human performance capture and fvv using consumer-grade rgb-d sensors," *Pattern Recognition*, vol. 79, pp. 260–278, 2018.

[11] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun, "Fast global registration," in *European Conference on Computer Vision*. Springer, 2016, pp. 766–782.

[12] J Blake, F Echtler, and C Kerl, "Openkinect: Open source drivers for the kinect for windows v2 device," 2015.

[13] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun, "Open3D: A modern library for 3D data processing," *arXiv:1801.09847*, 2018.

[14] Diana Pagliari and Livio Pinto, "Calibration of kinect for xbox one and comparison between the two generations of microsoft sensors," *Sensors*, vol. 15, no. 11, pp. 27569–27589, 2015.