Distortion Score based Pose Selection for 3D Tele-Immersion

Suraj Raghuraman* Balakrishnan Prabhakaran†

The University of Texas at Dallas

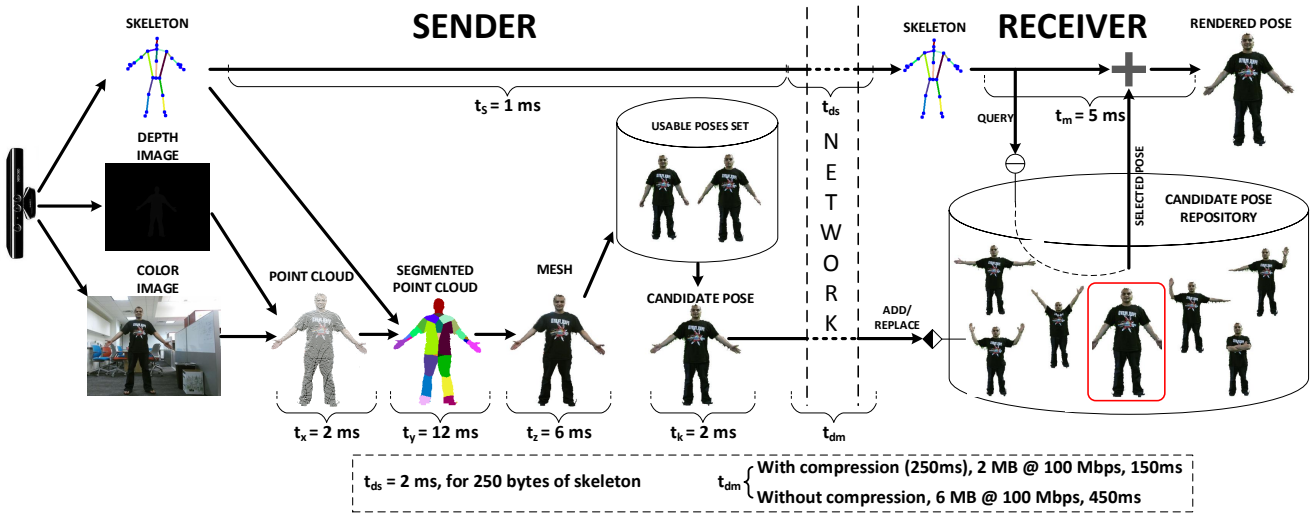

Figure 1: *Architecture of DISPOSE based 3DPTI, where t_{ds}, t_s are transmission and processing times for skeleton, respectively. t_ds, t_x, t_y, t_z, t_k are processing times for point cloud generation, segmentation, mesh generation, candidate pose selection and transmission respectively.*

Abstract

3D Tele-Immersion (3DTI) systems capture and transmit large volumes of data per frame to enable virtual world interaction between geographically distributed people. Large delays/latencies introduced during the transmission of these large volumes of data can lead to poor quality of experience of the 3DTI systems. Such poor experiences can possibly be overcome by animating the previously received mesh using the current skeletal data (that is very small in size and hence experiences much lower communication delays). However, using just the previously transmitted mesh for animation is not ideal and could render inconsistent results. In this paper, we present a DIstortion Score based Pose SElection (DISPOSE) approach to render the person by using an appropriate mesh for a given pose. Unlike pose space animation methods that require manual or offline time consuming pose set creation, our distortion score based scheme can choose the mesh to be transmitted and update the pose set accordingly. DISPOSE works with partial meshes and does not require dense registration enabling real time pose space creation. With DISPOSE incorporated into 3DTI, the latency for rendering the mesh on the receiving side is limited by only the transmission delay of the skeletal data (which is only around 250 bytes). Our evaluations show the effectiveness of DISPOSE for generating good quality online animation faster than real time.

*e-mail:suraj@utdallas.edu

†e-mail:bprabhakaran@utdallas.edu

VRST '15, November 13 – 15, 2015, Beijing, China.
 ISBN 978-1-4503-3990-2/15/11…$15.00
DOI: http://dx.doi.org/10.1145/2821592.2821600

CR Categories: H.4.3 [Information Systems Applications]: Communications Applications—[Computer conferencing, teleconferencing, and videoconferencing]; H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—[Artificial, augmented, and virtual realities];

Keywords: 3D Tele-immersion, Compression Algorithms, Animation, Pose Matching, Pose Selection, Distortion Score

1 Introduction

3D Tele-immersion (3DTI) systems are used in a wide range of applications like telepresence, tele-medicine, physical rehabilitation, gaming, teleart etc. [Kurillo and Bajcsy 2013]. 3DTI systems allow real time collaboration between people at different locations by rendering their live avatars in an interactive virtual world. To achieve this objective, multiple cameras are used to capture the scene at each location. Every frame of this captured information is then processed to generate 3D models, which are transmitted over Internet and rendered in the virtual world. The generated meshes are large in size (about 6 MB per camera) consisting of thousands of vertices and triangles. Even on high-speed networks, after lossy/lossless compression, transmission frame rate averages around 10 fps, resulting in a jittery rendering at the other side.

Skeleton based 3DTI transmission schemes [Lien et al. 2007; Raghuraman et al. 2013] overcome the issues of low frame rate present in the compression based systems by using skeletal data to animate the sender's 3D mesh. Smaller size skeleton data is transmitted every frame to allow the receiver to animate the current pose at the sender's side. The large sized 3D meshes are transmitted only intermittently, depending on the network bandwidth. The underlying assumption for doing so is that any recently transmitted mesh would produce good quality animation for a given skeleton.

However, situations where the most recently transmitted mesh and skeleton have a high amount of inconsistency would yield poor animation results. For example, if the most recently transmitted mesh has a person with folded arms and the most recent skeleton has a T-pose, then the resulting animation would definitely be poor due to the occlusion in the mesh itself. Therefore, choosing meshes to be transmitted based only on the availability of network bandwidth would compromise the quality of the animation.

Pose space or example based animation [Lewis et al. 2000; Xu et al. 2011] can successfully generate good quality animation using a few key poses of the mesh. So by retaining many more meshes instead of just using the last received mesh would enable the system to generate more realistic animation. 3DTI systems generate approximately 30 meshes a second. Online pose set generation in real time using traditional approaches is not feasible for such large volumes of data.

Proposed Approach: In this paper, we present a DIstortion Score based POse SElection (DISPOSE) approach to generate high quality animation by selecting the best possible outcome for a given pose. The images captured from the RGB-D cameras go through a series of processing steps to generate a deformable mesh. Artifacts are caused by various real world factors like self occlusion, clothing, lighting, etc. Moreover, the captured data that is noisy, along with inaccuracies in the processing pipeline, result in artifacts. DISPOSE models all these possible artifacts as distortion in the animated mesh. By studying these distortions, DISPOSE decides on the mesh that is most likely to generate the "optimal" possible estimation of the pose.

DISPOSE is applied both on the sending and the receiving side of a regular 3DTI setup, as shown in Figure 1. At the sender's side, DISPOSE based 3DTI approach determines the candidate meshes that need to be transmitted over the network based on the effectiveness of the mesh to animate different poses. At the receiver's side, the DISPOSE approach also buffers distinct meshes to allow animation from the best candidate pose for the given skeleton. While creating an animation sequence, higher priority is given to the most recently selected pose to retain temporal coherence between the frames, therefore improving the perceived visual quality. The average mesh captured by a single RGB-D is about 6MB requiring 450ms to transmit on a 100 Mbps network. With compression [Yang et al. 2006a] requiring 250ms of processing time, mesh size is reduced to about 2MB, that would imply 150ms for transmission over 100 Mbps network. In comparison, every frame transmission DISPOSE requires 2ms to transmit the skeleton and 5ms to animate the mesh at the receiver side.

Our Contributions:

- Real-time quantification of distortions introduced during animation of a captured human 3D model using RGB-D cameras. The proposed distortion score shows linearity with respect to the visual quality of the 3D human meshes.
- Online creation of distinct example poses suitable for animation in real time.
- The proposed distortion score is shown to be effective in 3D tele-immersion both (i) at the sender side for selecting candidate meshes to be transmitted, and (ii) at the receiver side for selection from a repository of received meshes so that the animated sequence gives a better visual quality.
- Incorporation of the proposed DISPOSE method implies the latency in rendering the sender side activities are reduced to the transmission delay associated with skeletal data transmission. Since skeletal data is around 250 bytes, this latency is indeed very low.

2 Related Works

Many methods have been proposed to transmit 3DTI information. Silhoute based meshes are directly transmitted by [Petit et al. 2009]. Image and *zlib* compression are used by [Redert et al. 2002] and [Yang et al. 2006a] respectively. View dependent compression was proposed by [Shi et al. 2009] for mobile devices. A heuristic based loss less mesh compression was proposed by [Mekuria et al. 2014]. An block difference based stereo camera data compression was proposed by [Zhou et al. 2011]. Multi-stream adaptive compression based transmission was proposed by [Yang et al. 2006b; Yang et al. 2010]. A single merged mesh is transmitted by [Alexiadis et al. 2014; Mekuria et al. 2013; Alexiadis et al. 2013] for improved performance. Multiple views obtained using Kinect and bisection algorithm [Vasudevan et al. 2011] for meshing allowed [Kurillo and Bajcsy 2013] to develop a low latency compression 3DTI approach.

Skeleton based approaches [Lien et al. 2007; Raghuraman et al. 2013] extract the skeleton of the data and use interpolation to generate the next frame using the previously arrived information. While skeleton based approaches offer higher frame rates, the visual quality of the result depends on the actions being performed and the transmitted mesh.

Our approach uses multiple meshes to generate the animation at the receiver side. Animation using multiple pose meshes [Lewis et al. 2000; Weber et al. 2007] produces good quality result. These methods require set poses and highly accurate registration in case of captured meshes. Mesh to mesh registration takes significant amount of time [Weber et al. 2007]. The alternative is to use a weighted single mesh to generate animations with little artifact, reduced by some post processing [Vaillant et al. 2013]. Generating accurate weights for meshes is time consuming and so cannot be used in 3DTI systems. Combination methods using pose selection and interpolation have been shown effective for video generation [Xu et al. 2011]. However, these methods require offline mesh pose database creation and take seconds to synthesize a single frame.

Considering the fact that the overall process of generating weights or database is time consuming, our method instead focuses on determining the quality of the animation result to select the visually "optimal" result. Various mesh evaluation metrics have been proposed in literature. A surface approximation of Hausdroff distance was used by [Aspert et al. 2002] to perform more efficiently. Surface roughness based measure was proposed by [Corsini et al. 2007] for comparing watermarked meshes. Metrics based on simple distortion measures, such as Hausdroff distance and root mean square error do not correlate with the human visual perception. Distortion metric based on difference of structure (captured via curvature statistics) of meshes being compared was proposed by [Lavoué 2011]. Local mesh roughness derived from Gaussian curvature was proposed by [Wang et al. 2012]. Even though these methods correlate well with human visual perception their computation takes a few seconds for a single mesh in a frame.

3 3D Tele-Immersion Overview

A 3D Tele-Immersion systems allow geographically distributed users to be immersed in a unified virtual environment. To display the live avatar of the person multiple cameras are used to capture a single cuboid space. All the cameras are calibrated both intrinsically and extrinsically. For each session depending on the transmission scheme the following steps are performed each frame:

Acquisition The user is captured using an array of stereo or RGB-D cameras surrounding them. The user is then extracted from the image using background subtraction.

Figure 2: *The various artifacts (highlighted) generated by from left to right occlusion, rigging, segmentation, meshing and skinning.*

Meshing Captured range images are projected to 3D using the intrinsic calibration of the cameras. Local neighborhood information from the range images is used to triangulate points creating a single mesh for each camera [Pajarola et al. 2003]. These meshes are realigned based on the extrinsic calibration between the cameras. Then the aligned meshes are zippered together using [Turk and Levoy 1994] to create a single mesh for each user.

Compression and transmission: The generated mesh is compressed using zlib. The texture image is encoded as a jpeg image similar to [Redert et al. 2002] and [Yang et al. 2006a]. The data is then transmitted to the receiver for rendering.

Skeleton based 3DTI transmit compressed meshes when the network bandwidth is available. A reference skeleton is transmitted every frame allowing the receiver to animate the last received mesh to the senders pose. These are extra operations involved:

Skeleton Extraction: The skeleton is provided by the Kinect sensor using a vision based approach in real time. The extracted skeleton is inaccurate in cases of occlusion. Occlusion related inaccuracies can largely be avoided by combining multiple Kinects skeletons to a single more accurate skeleton [Yeung et al. 2013]. Even with multiple Kinects the skeleton can be inaccurate in situations with self occlusion.

Segmentation: If a image based transmission scheme is used then the depth images are directly transmitted to the receiver and the depth image is segmented into regions based on the skeleton. A region growing based approach as described in [Adams and Bischof 1994] is used to identify each part of the body. If a mesh is transmitted directly then a voronoi based approach to segment the mesh. The skeleton is projected on to 3D space. The distance between skeleton joint and the vertex is used to determine the segmentation.

Skinning: When a skeleton is received, the most recently segmented mesh is used to animate the mesh. Spherical blend skinning using a constant weight for each segment is used to animate the mesh. Since estimating accurate weights for each vertex is time consuming a rigid association is assumed for the entire segment.

4 Visual Quality Challenges

The visual quality of the result is affected by the network bandwidth restrictions, camera calibration, noise from the cameras, depth estimation, image distortion, etc. Typically the data captured from a Kinect is extremely noisy. While processing this noisy data, a lot of new errors are introduced depending on the type of processing. The noise generated at the source level by the RGB-D cameras as well as calibration errors between the cameras are not considered in this paper. We primarily focus on the errors generated by the following:

Occlusion: RGB-D cameras follow the pinhole capture model, resulting in a lot of missing elements while capturing a scene. For an item to be captured by the camera, it needs to have a direct line of sight to the camera. Since many of the poses (such as folded hands) cause occlusion, these poses would create an empty space in the mesh as shown in Figure 2.

Clothing: Skeletal animation is not used to deform clothes. Loose clothes are simulated independent of the over all body of the rigged model. The deformation of clothing relies on not just a single frame but a series of events before the current frame as well. In the skeleton based 3DTI systems, since the vertices of the clothes are treated the same as the body, large artifacts are created. After animation depending on the texture of the clothes inconsistencies between individual segments of the mesh are also clearly visible.

Rigging: In character animation, rigging should fit the hierarchical bone structure, also called the skeleton, to the mesh accurately. This is typically done by a person or by a semi-automatic script. The skeleton identification process identifies and fits the skeleton to the depth information with reasonably good accuracy. However, for many poses such as the one shown in Figure 2, this rigging process can yield bad results, causing a cascading set of errors from the various other parts of the system. Rigging error almost always leads to segmentation error, causing similar artifacts.

Segmentation: It is not always possible to determine the exact boundaries of each limb with a high degree of accuracy. If the segmentation is incorrect (refer Figure 2), the animation yielded would also be incorrect, resulting in cobweb artifacts.

Meshing: When the mesh is created from the depth image, it is not possible to determine and segment the exact boundaries of the various objects inside the scene. The segmentation information is not used while creating triangles for the mesh. The generated triangles therefore, can span across limbs, or connect the person to the object near them (Figure 2). Such a mesh, when used for animation, generates cobweb artifacts.

Skinning: The skinning, or the actual animation of the mesh by moving the vertices, is based on certain assumptions. The primary assumption is that all of the deformation occurring in the mesh should be a direct result of the changes in the skeleton. The actual influence is interpreted differently in every method resulting in different artifacts. Figure 2 shows the artifact generated using the most widely used Linear Blend Skinning (LBS) method. For LBS, the artifacts typically have a shrinking of the mesh around the joints. For the method used in [Raghuraman et al. 2013], it is a problem of the stretching of the bone joints.

In the later sections, we go on to show that all the artifacts that are generated can be reduced down to meshing artifacts. We then define the distortion score which measures the exact amount of artifact in the mesh. This score can be used to understand the visual quality of the mesh to be rendered.

5 DISPOSE

3D tele-immmersion system generates large number of meshes every second, only a few of which are useful in animating a mesh using the skeleton of the participant. In skeletal animation, a rigged mesh can generate few fixed animations, and the vertex skinning weights of the mesh are tuned to ensure that there are no artifacts during the animation. Despite tuning the associated weights, the animation produced by the rigged meshes is only accurate enough within the bounds of the motion currently being performed. Over time, with the introduction of previously unseen motions, there is a reasonably high probability that the mesh will deform in an undesirable manner to animate these motions. Therefore, the quality of animation is directly influenced by the vertex weights and mesh chosen. DIstortion Score based POse SElection (DISPOSE) provides a scheme of selecting the mesh which is most likely to produce a better quality animation for the given skeleton. Instead of using a single mesh of a person in a particular pose to animate all other poses, DISPOSE selects a mesh from a set of meshes that is

most likely to animate the current pose accurately without any artifacts. This real-time selection of optimal meshes for animating motions every frame is intended to overcome the disadvantages of using just a single mesh and enhance visualization accuracy.

Before proceeding further, we briefly discuss mesh optimality, with respect to estimating poses. We refer to a mesh using which all the other meshes can be animated accurately as an ideal rigged mesh. A rigged mesh with its weights estimated for all the possible poses will be an ideal rigged mesh. However, estimating the weights of the mesh is a computationally intensive procedure and thus not suitable for real-time applications. So we define a mesh to be in good pose if it can animate a few meshes accurately. Intuitively, the best way to select a good pose is to select a mesh with least occlusion. In accordance with this assumption, a mesh featuring a person's hands in their pockets or their arms folded would be considered very bad poses and never be selected for animating different poses. However, there is a practical limitation to this assumption. For instance, let us consider a scenario involving the person merely standing with their arms folded for the majority duration of the session. In this case, the best pose would be one with folded arms because it will require the least amount of deformation. Therefore, selecting a good pose is dependent not only on generic factors like occlusion, but also session or application specific factors like the activities being performed or the clothes being worn during the session.

Pose selection is not commutative. Although two poses might have similar skeletons, it might take significant effort to estimate one pose from the other. For example, it might be easy to use an up-arrow pose to estimate a pose with hands behind the back. However, using a pose with hands behind the back to animate an up-arrow pose would not yield good results primarily due to occlusion of the hands. For this reason, it is not a good idea to select a pose based only on skeletal similarity.

5.1 Formal Definition of the Pose Selection Problem

We now provide a formal definition for the problem of an optimal pose selection. We define a **Pose** to be a pair comprising of the mesh and its corresponding skeleton $p = \{M, S\}$. The mesh selection problem can be defined as follows:

Given a set of poses $P = \{p_1, p_2, ...p_n\}$ and a target pose skeleton S_T, select a pose $p = \{M, S\} \in P$ that animates the mesh M_T corresponding to skeleton S_T with the best visual quality.

Let us define a function $M_{out} = A(M_{in}, S_{in}, S_{out})$ that transforms a mesh M_{in} with skeleton S_{in} into a mesh M_{out}, given the corresponding skeleton S_{out}. We can use this function to redefine our problem as:

Given a set of poses $P = \{p_1, p_2, ...p_n\}$ and a target pose skeleton S_T, select a pose $p = \{M, S\} \in P$, such that the difference between a generated mesh $M_{AT} = A(M, S, S_T)$ and the actually captured mesh M_T is minimized.

The objective of this approach is to determine the optimal pose from a set of poses, so that the generated mesh M_{AT} is most similar to the actual mesh M_T. If it were possible to directly derive the difference $|M_T - M_{AT}|$, then the solution would be straightforward. However, due to the noisy camera capture, the meshes generated for the exact same scene can look very different at a detailed level. Even in the absence of noise, there can be variations in vertex density, triangulation etc. that might incorrectly manifest as errors due to animation. This kind of similarity is generally verified visually by people. The mesh quality metrics described in Section 2, predominantly concentrate on analyzing variations from the original to generate the score. In this case, it does not yield accurate results due to the noisy quality of both the original and target meshes.

Therefore, we redesign our problem from one of minimizing mesh difference to minimizing transformation artifacts. The objective is to measure the amount of artifacts generated as a result of transforming a mesh M into a mesh M_{AT} to correspond to the captured skeleton S_T. We refer to this as Distortion Score (σ). If the Distortion Score $\sigma(p)$ can be computed for each pose $p \in P$ with respect to the target skeleton S_T, then the optimal pose to be selected would be $argmin(\sigma(p))$.

5.2 Distortion Score

We define Distortion Score σ as the amount of deformation that mesh M in pose S has to undergo to look like mesh M_{AT} in pose S_T. Intuitively, lesser the deformation, lesser are the artifacts, and more accurate is the rendering. Using mesh frames in the local temporal neighborhood of the target pose has been shown to yield accurate animations [Raghuraman et al. 2013]. It has also been shown that large amount of artifacts are generated for some frames even in the same neighborhood. Therefore, merely selecting a pose based on the minimum effort distance does not necessarily ensure minimum artifact. As a result, skeleton similarity measurements cannot be used to estimate the σ.

All the artifacts are modeled as excessive deformation. For this, we define deformation to be the degree of change in the size of the triangles of the mesh. As mentioned in Section 4, meshing, rigging, segmentation, and skinning artifacts result in excessive stretching or shrinking of the triangles of the mesh. Occlusion causes holes in the mesh that cannot be directly mapped as deformation during animation. However, with a minor change to the manner in which meshing is performed, it is possible to cause triangle stretching even in case of occlusion artifacts. To achieve this, while meshing, we eliminate the criteria for checking the distance between the candidate points on the depth image to form the triangles. Occlusion is caused by the foreground object taking over the pixel that should have ideally been occupied by the person's body. The vertices in the foreground form triangles with the vertices in the background, resulting in the covering of the holes. Filtering the scene by only depth retains just the person, with the background completely removed. As a result, the person is never meshed with the background. When the person moves a limb, the triangles connecting the limb to the body get stretched, resulting in a higher σ score. Most of the clothing related artifacts are also captured using the relaxed distances while meshing. Minor wrinkles and texture mapping are not handled using this approach.

Distortion Score σ can be intuitively explained as the extent of stretching of the triangles in the mesh while animating the target mesh. The changes in the length of edges of the triangles provide a realistic estimate of how the change in the position of the vertices due to deformation affects the mesh. Holistic score calculated as the difference between all the edges of the original and deformed mesh would result in large variations in few of the edges going unnoticed. In order to give higher precedence to edge changes, only the edges that can be deformed are considered. In the 3DTI system, vertices can be associated with only one segment, so only the edges connecting two segments will ever get deformed. This reduces the number of calculations for the distortion score estimation significantly, but does not change the participation level of the edge.

The mesh is divided into different regions based on the segmentation information available from the rigging process. Edges can now be associated with two regions based on the location of the vertices. Distortion score is calculated using a regional scoring approach of weighted averaging regional edge changes. As seen in Figure 3 using a region based approach the edge changes in the head region are captured resulting in a high distortion score. Given

Figure 3: *The effect of using various scoring methods: (left to right) selected pose with holistic scoring, selected pose with regional scoring, animated result for holistic scoring(less than 40 triangles in the head region change), animated result for regional scoring.*

a mesh M transformed to M_{AT} having edges $e \in E$, regions $R = \{r_1, r_2...r_m\}, r_i \subset E$ then the regional distortion score σ_r is given by:

$$\sigma_r = \frac{\sum_{e \in r} \Delta len(e)}{\sum_{e \in r} len(e)} \quad (1)$$

Where $len(e)$ is the length of edge e and $\Delta len(e)$ is the change in its length due to transformation. Therefore, the overall Distortion Score σ is calculated as:

$$\sigma = \frac{1}{m} \sum_{r \in R} \omega_r \sigma_r \quad (2)$$

where ω_r is the weight factor associated with the corresponding region of the mesh. The region based approach provides greater level of detail on the influence of different regions on the overall distortion score for a mesh, facilitating fault isolation. For example, for the mesh shown in Figure 3, the regional scores indicate that the regions corresponding to the hand and the head are erroneous. This also allows us to assign higher priorities to certain regions of the body over others while computing σ for the entire mesh. For instance, for a session focusing on upper body rehabilitation, the upper body can be given higher ω than the lower part of the body. Some regions like the thumb, palm and foot are prone to capture errors. Reducing the ω values for these regions improves the result significantly.

5.3 Pose Selection

The optimal pose for a given target pose requires selecting the pose with the lowest σ with respect to the target pose. The search is in the order of $O(NE)$ where N is the number of candidate meshes and E is the number of edges in a mesh. A single camera mesh can contain hundreds of thousands of edges. So, every new addition to the pose set significantly increases the processing time for the linear search. We have considered two strategies for searching the pose repository and selecting the candidate mesh:

- Exhaustive Search: The search can be performed in parallel to calculate the σ for each mesh. However, even this strategy might result in slow selection for a large set of poses. Computing the σ for each and every pose in a large pose set and taking the minimum may not be feasible for real time performance.

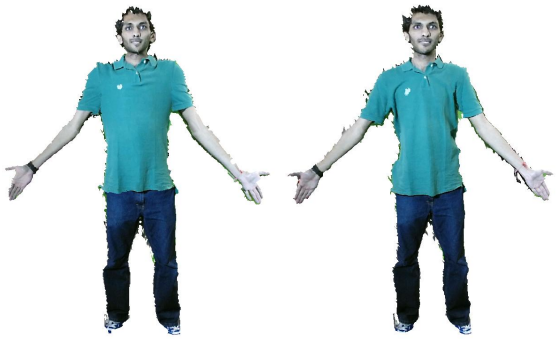

Figure 4: *Effect of selected pose on animation: T-pose (left) and arrow pose (right). Due to the stretching of shirt while in T-Pose the results look considerably different.*

- Threshold-based Search: The search uses an approximation to select the candidate pose with acceptable visual quality. If we are able to identify the σ associated with acceptable visual quality, then the search operation needs to last only until a pose that yields a σ less than the threshold is found and designated as adequately optimal. This significantly reduces the average search time for poses, even though the worst-case complexity remains the same.

  For this strategy, we use the session information to estimate the value for the threshold. A 3DTI session running for few minutes requires animating thousands of meshes. Each animation cycle results in an optimal pose to be selected from the pose set with a minimum σ value. The average of the σ values over multiple animation cycles can serve as a very good approximation of the desired visual quality. The starting threshold is set using prior knowledge about the score gained by running other sessions. However, during the initial stages, due to the limited availability of poses, the search is performed almost always on the entire pose set.

The performance of the search can be further improved using temporal information about the action in progress. Optimization can be performed based on the selection patterns of the poses in the pose set. For example, if certain poses are selected more often then higher priority can be given to calculating σ for those poses over others in the set. In Section 7.1.1, we have discussed the performance of the exhaustive and threshold-based search.

5.4 Animation Sequences

Selecting optimal pose for animating every frame without considering any temporal aspects might lead to temporal inconsistencies. For example, consider a scenario where a person is waving his/her arms, with the just the T-pose and arrow pose in the pose set, as shown in Figure 4. It is highly likely that the DISPOSE method will pick either of the poses in a repetitive alternate manner, resulting in a jittery animation sequence. In this particular situation, the shoulder region of the person changes significantly between the two poses, as seen in Figure 4, which causes the sequence to look even worse. Since the approach only considers the σ of each candidate mesh independently, and does not measure the quality or features of the candidate poses, these kinds of situations might occur quite frequently. Even though both of the poses are equally capable of animating the target skeleton, it is important from the animation sequence point of view to hold onto one pose for as long as possible. In the above example, if the same pose were used to animate the entire sequence then the result would be devoid of the twitching that may be caused by frequent switching of selected poses. Therefore,

the most recently selected pose is given higher priority over other poses in the pose set. This ensures that the most recently selected pose is held on to until its σ exceeds the cut-off threshold (discussed in Section 5.3). The resulting mesh animation sequences are more consistent and visually appealing.

6 DISPOSE in 3DTI

Skeletal schemes [Lien et al. 2007; Raghuraman et al. 2013] for 3D Tele-Immersion deform the last received mesh based on the current skeleton. These techniques discard the previously received meshes and do not have any selection criteria for the mesh being transmitted. To overcome these limitations, we propose the application of the DISPOSE in 3DTI.

DISPOSE is applied in two stages, as shown in Figure 1. The first stage involves directly streaming the skeleton and applying DISPOSE on the Candidate Pose Repository(CPR) at the receiver side to render the result. The second stage involves the processing of the captured data to a rigged mesh and maintaining the CPR. The meshes that need to be added in CPR are identified at the sender side and transmitted to the receiver side maintaining the actual CPR.

6.1 Sender Side - Identifying Candidate Poses

Among the poses captured by the camera, candidate poses for animation are identified based on their ability to animate other poses. While identifying the candidate poses, we follow the same criteria as identifying the ideal rigged pose. This candidate pose selection for streaming on the sender side can be stated as follows:

Let there be a universal set of all the possible poses that can be performed by the person, P_u. Let the poses performed by the person in a time duration be $P_w \subset P_u$. The candidate pose for streaming is $P_c \in P_w$, such that P_c can be used to animate a large number of poses in P_w, with a high degree of accuracy.

Candidate Pose Selection Strategy:

1. To identify P_c, we use the local neighborhood (in terms of time) of the poses generated around P_c. We use a simple algorithm with $O(N)$ (N being the number of poses in the neighborhood) complexity to identify the candidate pose in real time. This algorithm builds a Usable Pose Set (UPS) store the probable meshes for transmission. It also buffers all the skeletons that are transmitted in the recent past, at the sender side. This target set of skeletons is used to identify the candidate pose.
2. A *Rigging Quotient*, (χ), is calculated as N_p/N_s where N_p is the number of meshes animated accurately (σ, lower than selection threshold) by the pose and N_s is the total number of skeletons in the skeleton set. Any deformation
3. When the bandwidth is available to transmit the mesh, the pose with the highest rigging quotient (χ_s) is transmitted and all the other poses that occurred before the transmitted pose, except for poses with χ greater than Retain Threshold (RT), are removed. RT is calculated as a percentage of χ_s. The percentage used is configurable and is directly proportional to the network bandwidth.
4. All the skeletons that occurred before the transmitted pose are also removed from the skeleton set.
5. The χ values are recalculated and the retained poses are flagged for transmission. The next time there is enough bandwidth to transmit a pose, the meshes flagged for transmission that have a χ greater than RT are transmitted.

By transmitting the meshes that are "optimal" for animating in the local window, and are good at animating over a long period of time, the method ensures all the likely candidates are available at the receiver side.

While adding a pose, the σ of the pose is evaluated with respect to the existing poses in the usable pose set. If the pose has a low σ for an existing pose, then it replaces the pose in the usable pose set. If the mesh has high σ for all the poses in the usable pose set, then its σ is calculated with respect to all the other skeletons in the buffered skeleton set. If the pose has low σ for even one skeleton, then the pose is added to the useful poses set and the χ for all the poses are recalculated.

6.2 Candidate Pose Repository

On the receiver side, a Candidate Pose Repository (CPR) keeps track of all the poses identified as the candidate pose by the sender. For every skeleton received, DISPOSE is used to select the candidate pose from CPR and animate the target mesh for the corresponding skeleton. It is highly likely that in a high-speed network environment, the sender would send a large number of poses. As a result, the amount of time taken to select the candidate pose and animate the target pose would be significant. So, it is extremely important to limit the size of the CPR within the runtime bound of the system. If the animation using DISPOSE takes longer than the runtime bound, the CPR is pruned. We use an always-add-and-prune-when-required strategy for limiting the size of CPR.

A received pose is always added to CPR, since the new pose may have the latest facial expressions and/or other changes in the human mesh. This addition to CPR may require replacing an already existing candidate pose (to limit the number of poses in CPR). The pruning strategy is as follows:

- The σ of received candidate pose with respect to all the poses in the CPR is calculated, any pose in CPR having a σ less than the Equal Threshold(ET) is replaced by the new pose.(i.e. if the received candidate pose can animate one of the existing poses very accurately then it replaces that pose in CPR). ET is set to a value lower than 0.1 depending on the required pruning level.
- Each candidate mesh has an associated usage count and is arranged by the order of usage. New poses added to the CPR are considered as used when they are added. Also every time a candidate pose is selected, the usage counter for that particular candidate pose is incremented. While pruning, poses that have not been used or used very sparingly, are removed from the CPR.

Using the always-add-and-prune-when-required strategy DISPOSE gives good visual results in a timely manner. While experimenting, it was observed that the value of ET plays a key role in controlling the size of CPR.

7 Evaluation

Setup: Experiments were carried out on a system running Microsoft Windows 8.1, with an Intel i7 3.4ghz processor, 32GB RAM, and GTX 590 GPU. A single Microsoft KinectV2 sensor was used to capture the actions of the users. Only one sensor was used to ensure that the system was able to compensate for all kinds of occlusion. All the computations were performed on the CPU, i.e., a single machine was used to both store and process the information from the cameras. The entire processing pipeline to generate the rigged mesh as described in 3 was running on the same machine in parallel.

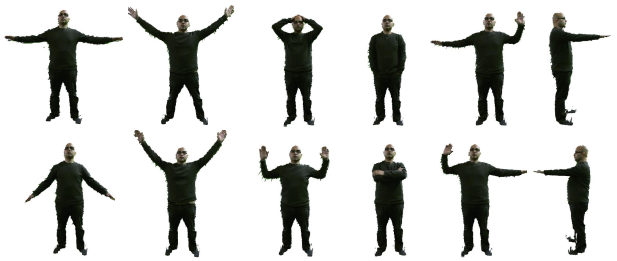

Figure 5: *Illustration of 12 fundamental poses captured in the dataset.*

Dataset: A dataset of 7 participants having varying physical attributes, captured using a single RGB-D camera, was used for evaluating the performance of the DISPOSE based 3DTI approach. A dataset consisting of performing actions largely facing the camera was captured. Due to the large size of data, this dataset would be made available on request. The actions involved two phases: (1) repeating 12 fundamental poses as shown in Figure 5 once and (2) performing activities that the participant chooses to perform for the rest of the duration of 2 minutes. The actions performed in the second phase varied across the participants, with several of these actions resulting in self-occlusion (due to the use of a single Kinect camera). A total of about 25,000 frames of information were captured using the 7 participants. All the data was used for the experiments and the user study.

7.1 Experiments

We carried out two categories of experiments to gauge the performance of the Distortion Score and the effectiveness of using the DISPOSE strategy to generate animation sequences. Distortion score was calculated with all the region weights set to 1. The threshold value was set to 0.15 for the threshold-based search.

7.1.1 Performance of Distortion Score

Linearity of Distortion Score: In the first experiment, the high level accuracy and progressive nature of the score is validated. The distortion score σ was calculated for the entire dataset using the 12 candidate poses. Meshes were clustered based on σ and visually inspected to verify the visual quality of each cluster. It was noticed that the quality of meshes is progressively worse when going from $\sigma = 0$ to $\sigma = 3$ (a small snapshot of the same is shown in Figure 6). It was possible to verify at a coarse level that increasing distortion scores capture increasing mesh artifacts. However, at a fine level, it was very hard to determine the difference in meshes very close to each other in terms of σ.

Running Time of Distortion Score: Theoretically, the running time of DISPOSE is linear to the number of poses in CPR (i.e., O(NE), where N is the number of poses in CPR and E is the average number of edges in a pose). But parallel implementation of the exhaustive search approach on the test machine resulted in non linear growth as the number of meshes increased. As seen in Figure 7, after crossing a threshold of 19 candidate poses, the parallel exhaustive search starts to take longer than linear growth and becomes unfeasible to compute and use in real-time. Since the CPU was used the performance reduction was not due to the lack of memory as seen in the threshold search results, but due to the large number of threads used to compute the score.

In comparison, the threshold based selection described in Section 5.3 performs faster even with a large CPR. The advantage of stopping the search after finding a reasonably good match allows the CPR to grow to more than 100 poses, with the result computed in under 20ms. However, threshold-based search also takes the same time as exhaustive search when the quality threshold is set to very high quality and no good candidate pose is available. If the number of poses in CPR is pruned, the threshold based search time could be maintained below 20ms.

Efficacy of Threshold based Search: It is clear that the threshold search approach of pose selection is faster and more scaleable than the exhaustive search. The qualities of the result generated by both the exhaustive and the threshold-based approach were found to be highly similar. To study the difference in quality, frames with the highest variation between the threshold and the exhaustive search where selected. The typical threshold was found to be 0.15 on the entire dataset and used as the default starting value for the threshold. The highest variation found was 0.12 and the meshes always looked similar from the overall artifact point of view. As seen in Figure 8, it is very difficult to determine which of the two is of better quality after ignoring the edge noise from RGB-D capture.

7.1.2 Evaluating DISPOSE in 3DTI

Table 1: *Comparison of various 3DTI transmission strategies in literature.*

Method	Processing Latency (ms)	Payload Size(Mb)	FPS*
[Petit et al. 2009]	NA	29	15
[Zhou et al. 2011]	80	8	14
[Mekuria et al. 2014]	151	58	5
[Alexiadis et al. 2014]	500	7	20
[Yang et al. 2006b]	159.5	NA	5
[Yang et al. 2010]	106.9	NA	7
[Alexiadis et al. 2013]	136.7	NA	7.3
[Mekuria et al. 2013]	200	4.3	8
[Kurillo and Bajcsy 2013]	47	NA	20
[Raghuraman et al. 2013]	1	0.002	30
DISPOSE	6	0.002	30

NA: Data not available
* - as reported based on network conditions used

Latency and Frame-rates: Next, we compared the performance of DISPOSE-based 3DTI in terms of the payload size and the latency associated with the processing of the input, with the systems available in literature as shown in Table 1. Both skeleton based 3DTI and DISPOSE-based 3DTI have the lowest payload size and highest frame rates. Having a average payload size of just over 250 bytes and a bit rate of about 56kbps, the results generated by the animation approaches is very responsive and real time even over a low speed internet connection. With Internet bandwidth around 1Mbps upload speed, the animation approaches, after an initial lead-time for a single frame transfer, perform at the source frame rate of 30fps. Whereas all the compression based approaches take a longer time to update just one frame. The compression of [Lien et al. 2007] is comparable to our approach with the lowest quality settings. But due its large processing times per frame (in seconds), it was not included in the table.

User Study: We performed a user study where 37 participants compared the visual quality (appearance without artifacts) of skeleton based 3DTI [Raghuraman et al. 2013], DISPOSE-based 3DTI and the loss-less compressed stream [Yang et al. 2006a]. The participants were shown the recorded version of the compression based stream at 10Mbps, DISPOSE, skeleton based 3DTI and the original stream side by side. A total of 12 videos were generated from the dataset with videos 7 and 9 having high self occlusion. The participants where asked to rate the visual quality and the responsiveness

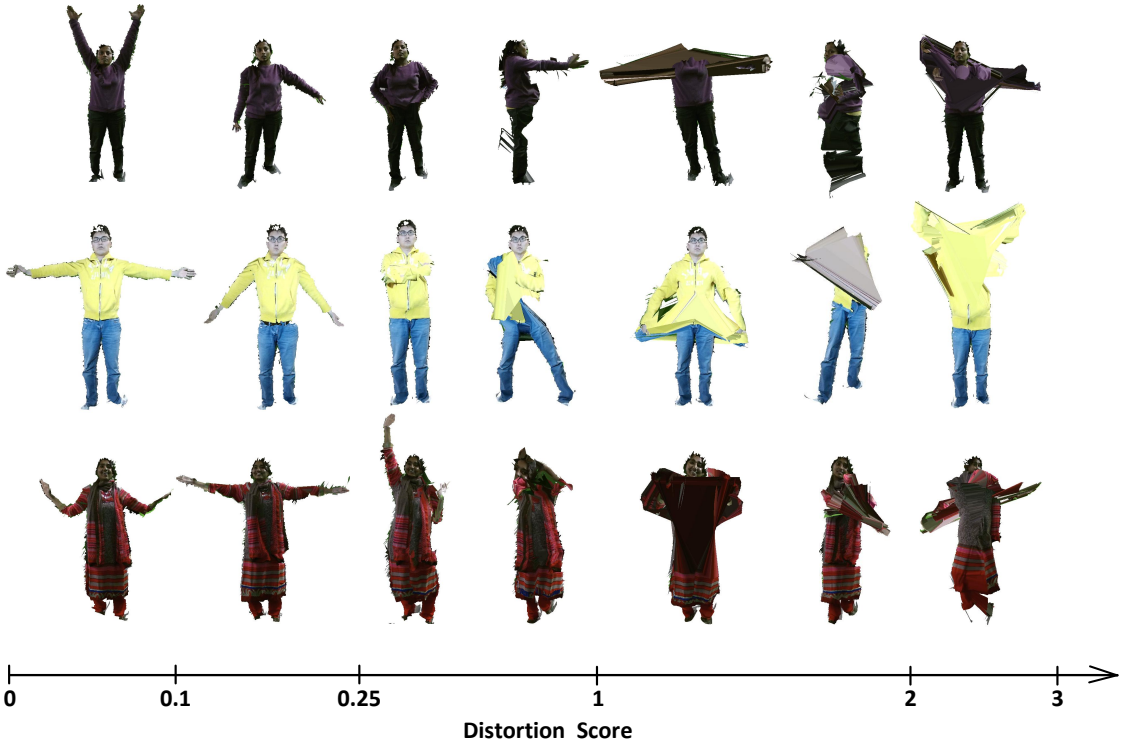

Figure 6: *Instances of Varying Distortion Score from multiple participants, in increasing order (left to right).*

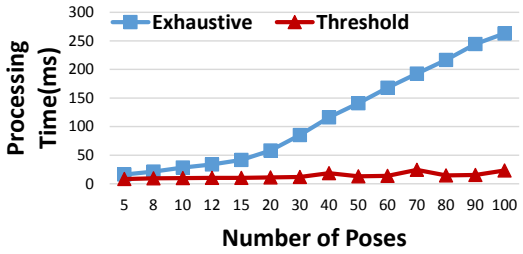

Figure 7: *Comparison of processing times for exhaustive search and threshold based search.*

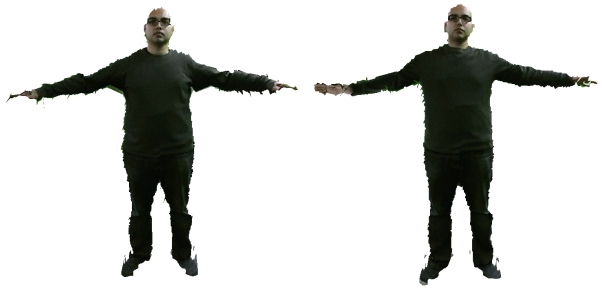

Figure 8: *The mesh selected by exhaustive search having $\sigma = 0.02$ (left) and threshold based selected pose (right) having $\sigma = 0.14$*

.

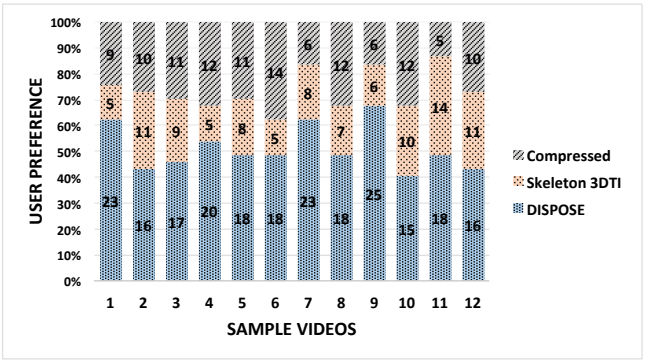

Figure 9: *Results of user study for preference of transmission strategies.*

on a scale of 1 to 10. Due to the absence of formal definitions and specifications for these qualitative measures, the responses reflect the participants' respective subjective interpretations. The preferred approach of the participants based on the choice of visual quality is shown in Figure 9. Both skeleton based 3DTI and DISPOSE approaches had high ratings for responsiveness, with the compression based stream considered the worst.

8 Discussion

DISPOSE approach to generate animation sequences is very effective at generating real time animation using an online stream of meshes (no correspondence) and skeletons. We discuss few observations on the approach below.

Ability to Handle Self Occlusion: This experiment focused on the ability of DISPOSE to improve the capture quality of the original meshes, in terms of self-occlusions. The raw images captured

Figure 10: *Capture with self occlusion: (left to right) captured mesh with occlusion, selected pose, animated mesh without occlusion.*

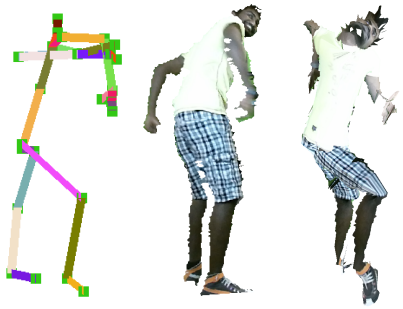

Figure 11: *Incorrect skeleton detection resulting in inaccurate animation: (left to right) detected skeleton, actual mesh, animated mesh.*

using Kinect and other pin hole based RGB-D cameras is highly susceptible to self occlusion as shown in Figure 10. Since the DISPOSE based approach chooses a mesh that would result in minimal distortion, it was noticed that the visual quality of the rendering was better at the receiver side than the sender side, due to portions of the mesh being missing due to self occlusion at the senders side. Since the pipeline to capture, mesh and render takes slightly longer than DISPOSE based rendering, using DISPOSE at the senders side can also give significant performance gains especially while using multiple camera capture.

Effect of Inaccurate Skeleton: As we mentioned earlier, we use the Microsoft SDK for extracting the skeletal data stream from the Kinect cameras. Skeletal data is quite accurate in normal exercise poses but is error prone in cases of occlusion. We observed that in about 780 of the 25,000 captured frames, the skeleton associated with the mesh was inaccurate. Since DISPOSE uses just the skeleton to animate the result, all the rendered sequences for these frames were inconsistent. In cases with heavy distortions as shown in Figure 11 the frames were not rendered to the user due to very high σ.

Unlike the animation part, pose set update process can tolerate large skeleton inaccuracies(greater than 5cm) effectively. Larger errors result in very high σ for the mesh resulting in low χ and hence is never added to the pose set. Certain poses with small skeletal joint errors can still have low σ and high χ forcing an addition into the pose set. Estimating a single skeleton using multiple kinects has been shown to be fairly resistant to occlusion [Yeung et al. 2013].

Skeleton similarity based selection: Pose selection based approaches [Xu et al. 2011; Lewis et al. 2000; Hilsmann et al. 2013] rely on the similarity between the pose and the mesh skeletons in the database to select the appropriate mesh to animate the pose. We found that skeleton similarity based indexing, distance metric based selection etc. work effectively in scenarios with proper meshing, rigging, segmentation and skinning. However, for online real time applications like 3DTI skeleton match based approaches generate many artifacts (Figure 3). Skeleton based strategies will be faster and may be required to achieve higher frame rates at lower processing costs. Currently using threshold-based search it is possible to achieve 30fps on the CPU for pose sets with about 200 poses.

Other Distortion/quality scores: Preliminary analysis by using faster quality metrics like the dihedral angle did not provide good quality results. Significant noise in the captured depth information affects the scores determined by the quality metrics tremendously. It was often observed that the animated meshes were given a higher score than even the original captured or pose meshes. Due to this ambiguity, no quantitative comparisons are performed using the quality scores.

Skinning artifacts: Our 3DTI system currently uses fixed weights for the entire segment. Spherical blend skinning produces joint artifacts even on expert weighted mesh. Selecting the best rendered mesh based on these artifacts is therefore beneficial even when multiple expert weighted meshes are available or in any of the pose space based approaches. DISPOSE simply selects the most suitable animate mesh based on σ and since it does not play any role in the actual vertex updates, the animation quality is ultimately decided by the skinning approach used to deform the meshes.

9 Conclusion

We proposed DIstortion Score based POse SElection (DISPOSE) approach to improve the responsiveness of 3D Tele-Immersion (3DTI). Though primarily designed for use in 3DTI, DISPOSE can be applied to generate better quality animation in pose space animation based situations. Modeling errors are mesh artifacts and quantifying them can enable generation of better approaches to animation.

In 3DTI, DISPOSE can be incorporated both: (i) at the sender side to select the candidate meshes to be streamed to the receiver; (ii) at the receiver side to select the mesh that can minimize the distortion associated with the rendering of the sender's actions. For this purpose, various artifacts introduced in the capture and processing stages of 3DTI are modeled as mesh distortions. These distortions are then quantified by the distortion score and the best result with minimal distortion is chosen. Temporal consistency is maintained by holding on to the selected mesh longer while skinning the animation sequences.

DISPOSE-based 3DTI has a very low latency (equivalent to the transmission delay of skeletal data which is around 250 bytes). Due to the low latency, DISPOSE-based 3DTI is able to maintain a higher rendering frame-rate than 3DTI systems employing compression based schemes. We have also shown that the visual quality of the rendering generated by DISPOSE-based 3DTI is better than skeleton based 3DTI. User studies also substantiate that the visual experience on rendering generated using DISPOSE-based 3DTI is better than compression or skeleton based 3DTI.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1012975. 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 National Science Foundation.

References

ADAMS, R., AND BISCHOF, L. 1994. Seeded region growing. *IEEE Trans. Pattern Anal. Mach. Intell. 16*, 6 (June), 641–647.

ALEXIADIS, D., ZARPALAS, D., AND DARAS, P. 2013. Real-time, full 3-d reconstruction of moving foreground objects from multiple consumer depth cameras. *Multimedia, IEEE Transactions on 15*, 2 (Feb), 339–358.

ALEXIADIS, D., ZARPALAS, D., AND DARAS, P. 2014. Fast and smooth 3d reconstruction using multiple rgb-depth sensors. In *Visual Communications and Image Processing Conference, 2014 IEEE*, 173–176.

ASPERT, N., SANTA-CRUZ, D., AND EBRAHIMI, T. 2002. Mesh: measuring errors between surfaces using the hausdorff distance. In *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, vol. 1, 705–708.

CORSINI, M., GELASCA, E., EBRAHIMI, T., AND BARNI, M. 2007. Watermarked 3-d mesh quality assessment. *Multimedia, IEEE Transactions on 9*, 2 (Feb), 247–256.

HILSMANN, A., FECHTELER, P., AND EISERT, P. 2013. Pose space image based rendering. *Computer Graphics Forum 32*, 2pt3, 265–274.

KURILLO, G., AND BAJCSY, R. 2013. 3d teleimmersion for collaboration and interaction of geographically distributed users. *Virtual Reality 17*, 1, 29–43.

LAVOUÉ, G. 2011. A multiscale metric for 3d mesh visual quality assessment. *Computer Graphics Forum 30*, 5, 1427–1437.

LEWIS, J. P., CORDNER, M., AND FONG, N. 2000. Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '00, 165–172.

LIEN, J.-M., KURILLO, G., AND BAJCSY, R. 2007. Skeleton-based data compression for multi-camera tele-immersion system. In *Advances in Visual Computing*, vol. 4841 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 714–723.

MEKURIA, R., SANNA, M., ASIOLI, S., IZQUIERDO, E., BULTERMAN, D. C. A., AND CESAR, P. 2013. A 3d tele-immersion system based on live captured mesh geometry. In *Proceedings of the 4th ACM Multimedia Systems Conference*, ACM, New York, NY, USA, MMSys '13, 24–35.

MEKURIA, R., SANNA, M., IZQUIERDO, E., BULTERMAN, D., AND CESAR, P. 2014. Enabling geometry-based 3-d tele-immersion with fast mesh compression and linear rateless coding. *Multimedia, IEEE Transactions on 16*, 7 (Nov), 1809–1820.

PAJAROLA, R., SAINZ, M., AND MENG, Y. 2003. Depth-mesh objects: Fast depth-image meshing and warping. Tech. rep.

PETIT, B., LESAGE, J.-D., BOYER, E., FRANCO, J.-S., AND RAFFIN, B. 2009. Remote and collaborative 3d interactions. In *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video, 2009*, 1–4.

RAGHURAMAN, S., VENKATRAMAN, K., WANG, Z., PRABHAKARAN, B., AND GUO, X. 2013. A 3d tele-immersion streaming approach using skeleton-based prediction. In *ACM Multimedia*, 721–724.

REDERT, A., DE BEECK, M., FEHN, C., IJSSELSTEIJN, W., POLLEFEYS, M., VAN GOOL, L., OFEK, E., SEXTON, I., AND SURMAN, P. 2002. Advanced three-dimensional television system technologies. In *3D Data Processing Visualization and Transmission*, 313 – 319.

SHI, S., JEON, W. J., NAHRSTEDT, K., AND CAMPBELL, R. H. 2009. M-teeve: Real-time 3d video interaction and broadcasting framework for mobile devices. In *ICST International Conference on Immersive Telecommunications (IMMERSCOM)*.

TURK, G., AND LEVOY, M. 1994. Zippered polygon meshes from range images. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, ACM, New York, NY, USA, SIGGRAPH '94, 311–318.

VAILLANT, R., BARTHE, L., GUENNEBAUD, G., CANI, M.-P., ROHMER, D., WYVILL, B., GOURMEL, O., AND PAULIN, M. 2013. Implicit skinning: Real-time skin deformation with contact modeling. *ACM Trans. Graph. 32*, 4 (July), 125:1–125:12.

VASUDEVAN, R., KURILLO, G., LOBATON, E., BERNARDIN, T., KREYLOS, O., BAJCSY, R., AND NAHRSTEDT, K. 2011. High-quality visualization for geographically distributed 3-d teleimmersive applications. *Multimedia, IEEE Transactions on 13*, 3 (June), 573–584.

WANG, K., TORKHANI, F., AND MONTANVERT, A. 2012. A fast roughness-based approach to the assessment of 3d mesh visual quality. *Computers And Graphics 36*, 7, 808 – 818. Augmented Reality Computer Graphics in China.

WEBER, O., SORKINE, O., LIPMAN, Y., AND GOTSMAN, C. 2007. Context-aware skeletal shape deformation. *Computer Graphics Forum 26*, 3, 265–274.

XU, F., LIU, Y., STOLL, C., TOMPKIN, J., BHARAJ, G., DAI, Q., SEIDEL, H.-P., KAUTZ, J., AND THEOBALT, C. 2011. Video-based characters: Creating new human performances from a multi-view video database. In *ACM SIGGRAPH 2011 Papers*, ACM, New York, NY, USA, SIGGRAPH '11, 32:1–32:10.

YANG, Z., CUI, Y., ANWAR, Z., BOCCHINO, R., KIYANCLAR, N., NAHRSTEDT, K., CAMPBELL, R. H., AND YURCIK, W. 2006. Real-time 3d video compression for tele-immersive environments. In *Proceedings of Multimedia Computing and Networking 2006*.

YANG, Z., YU, B., NAHRSTEDT, K., AND BAJSCY, R. 2006. A multi-stream adaptation framework for bandwidth management in 3d tele-immersion. In *Proceedings of the 2006 International Workshop on Network and Operating Systems Support for Digital Audio and Video*, ACM, New York, NY, USA, NOSSDAV '06, 14:1–14:6.

YANG, Z., WU, W., NAHRSTEDT, K., KURILLO, G., AND BAJCSY, R. 2010. Enabling multi-party 3d tele-immersive environments with viewcast. *ACM Trans. Multimedia Comput. Commun. Appl. 6*, 2 (Mar.), 7:1–7:30.

YEUNG, K.-Y., KWOK, T.-H., AND WANG, C. C. 2013. Improved skeleton tracking by duplex kinects: A practical approach for real-time applications. *Journal of Computing and Information Science in Engineering 13*, 4, 041007.

ZHOU, Z., CHEN, X., ZHANG, L., AND CHANG, X. 2011. Internet-wide multi-party tele-immersion framework for remote 3d collaboration. In *VR Innovation (ISVRI), 2011 IEEE International Symposium on*, 183–188.