Autonomous Tracking For Volumetric Video Sequences

Matthew Moynihan
V-SENSE
Trinity College Dublin
mamoynih@tcd.ie

Susana Ruano
V-SENSE
Trinity College Dublin
ruanosas@tcd.ie

Rafael Pagés
Volograms
rafa@volograms.com

Aljosa Smolic
V-SENSE
Trinity College Dublin
smolica@tcd.ie

Abstract

As a rapidly growing medium, volumetric video is gaining attention beyond academia, reaching industry and creative communities alike. This brings new challenges to reduce the barrier to entry from a technical and economical point of view. We present a system for robustly and autonomously performing temporally coherent tracking for volumetric sequences, specifically targeting those from sparse setups or with noisy output. Our system will detect and recover missing pertinent geometry across highly incoherent sequences as well as provide users the option of propagating drastic topology edits. In this way, affordable multi-view setups can leverage temporal consistency to reduce processing and compression overheads while also generating more aesthetically pleasing volumetric sequences.

1. Introduction

Volumetric video creation through multi-view capture and processing of photo-realistic 3D human performances is an active research field that involves different disciplines such as computer vision, computer graphics and 3D geometry processing. The increasing interest in this field has been powered by new developments in immersive technologies (i.e., augmented, virtual and mixed reality), as these applications require more realistic and human content. To capture these realistic human performances, one typically needs a multi-camera system that records the performer from different viewpoints, such as the one proposed by Collet et al. [4] or Guo et al. [8], which uses more than one hundred high-end cameras (including infra-red projectors and cameras) to achieve the best reconstruction possible in a very controlled environment. In these systems, 3D reconstruction algorithms are run on a per-frame basis and the output is a sequence of 3D models (i.e., an independent mesh and texture image per frame). Some methods address this problem by enforcing temporal coherence in the 3D reconstruction process [19, 20, 21], however, to avoid storing large amounts of data per frame it becomes necessary to apply a mesh tracking algorithm that introduces temporal consistency in the sequence and enables the reuse of a significant amount of data. This compression can be facilitated by keeping the same topology for as long as possible throughout the sequence and updating only the mesh vertex positions. Furthermore, to enable heterogeneous sequences with variations in the mesh geometry and topology, it is necessary to split the sequence into regions controlled by keyframe meshes, similar to methods employed in video encoding. The current state of the art for mesh tracking in this manner works well when consecutive meshes are very similar to each other which is the case for high-end setups; however, they can fail when applied to capture methods which use sparser camera setups [12, 24] or even monocular systems [26, 27] where there is a significant amount of noise, or if geometry is lost (for example a hand or entire limb) due to the challenging capture conditions. Our proposed approach prioritises generality and scalability by applying temporal coherence to an unstructured series of meshes in a completely autonomous fashion, requiring no system priors, and supporting the challenging conditions presented above. Lastly, our system allows for the recovery of missing geom-

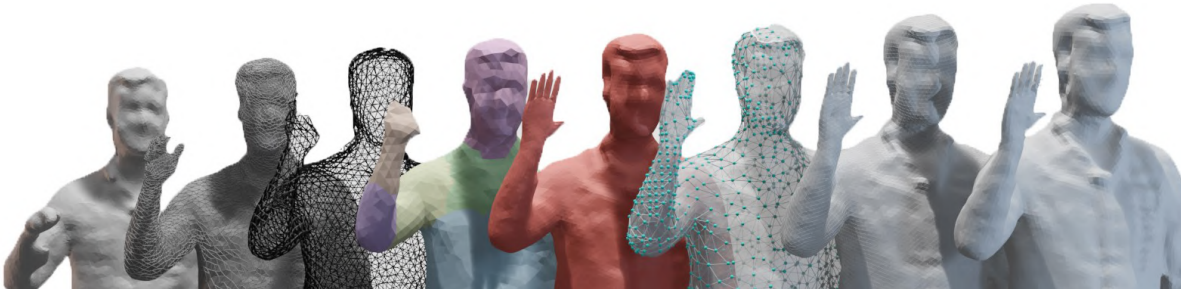

Figure 1. We present a robust, autonomous method for tracking volumetric sequences which can detect missing geometry and propagate user edits. Pictured left to right are step-by-step visualizations of the process. The input to our system is a temporally incoherent and noisy sequence of meshes. We perform pairwise registration using abstraction layers, volumetric segmentation and a keyframing system which allows for user edits, e.g. the hand recovered in red. We establish correspondences which maintain edits and propagate geometry throughout a graph-based deformation process.

etry and enables the user to introduce geometry edits that can be seamlessly propagated through the sequence. In particular, the proposed system presents the following contributions towards tracking noisy volumetric data from sparse multi-view capture:

- An automatic, similarity-driven keyframe selection process based on spherical harmonics that minimises keyframes and supports varying geometry and topology.
- A volume-based segmentation and registration method for robust tracking of volumetric sequences.
- A tracking system that enables missing geometry recovery and realistic propagation of user edits.

2. Related Work

Mesh Tracking. Mesh tracking and registration algorithms, especially when representing the shape and appearance of humans, are an essential part of volumetric video processing pipelines. Such systems use variably dense arrays of RGB and depth cameras to perform per-frame 3D reconstruction [4, 8], while other methods use monocular RGBD sensors [7, 35, 37, 38] and online character template generation [29, 30, 35, 36]. For each of these systems mesh tracking and registration is a fundamental process, ensuring temporal coherence for visual appeal and reduction of data overheads.

The use of a template-driven method helps constrain the problem focus toward reliable pose estimation. With recent developments in monocular 3D pose algorithms [3, 33], similarly, single-camera performance capture systems can produce reliable results [10, 34]. However, even if one was to take pose estimation for granted, the template deformation can still become a challenging task and quite often the approach will be some amalgamation of a customised avatar fitted to a pre-defined parametric model such as SMPL [18]. While the use of a template generally produces robust results, these systems cannot capture dynamic changes in topology without the use of some adaptive surface deformation. Habermann et. al [10] present a hybrid of pose-driven template deformation as well as graph-based surface alignment driven by 2D keypoints. While this system is more capable of modelling the dynamic motion of clothing, it is still unable to capture drastic changes in topology which would stray from the input template such as the introduction of new objects or changing clothes.

Some approaches acknowledge this problem and instead opt for the use of an evolving, canonical model which is constructed over the course of the capture [6, 23, 35]. These methods are well adapted to modelling temporally sensitive, high-frequency details and can faithfully produce temporally coherent models from noisy RGBD data. However, these systems are input-limited to the use of depth sensors which may not be as widely available or scalable as commodity RGB cameras. For the proposed work we seek to improve content created from scalable studio setups, some of which employ multiple arrays of RGB and infrared structured light sensors [4, 8] while others present extremely flexible and economical sparse arrays of commodity cameras only [12, 24]. Given a sequence of unstructured meshes generated from such setups, the general approach towards adding temporal coherence is to perform keyframe-based tracking of sequential mesh pairs. Like many of the previously addressed tracking algorithms, this work also leverages the deformation graph of [31]. The correspondences which guide the deformation in such graph-based approaches are often based on constrained ICP variants [16] or supported by photometric data [5]. Few systems address the scenario of missing geometry [11] and even so, they require strong priors and robust skeleton estimation. In contrast the proposed work requires no priors and doesn't impose any constraints on the mesh topology or number of independent components.

Keyframe Detection. Many sequential tracking sys-

tems for unstructured mesh sequences rely on some form of keyframing system in order to select the ideal candidate frames to begin tracking. Collet et. al [4] propose a number of heuristics metrics for keyframe selection based on the genus, surface area and number of connected components. These metrics are combined to formulate a *feasibility score* which is used to drive the keyframe selection.This approach is reasonably suited to consistent, high-quality input which would be expected from the system presented in [4]. However, when applied to the highly inconsistent data typical of sparse setups, any metric directly dependant on the input topology becomes uninformative (e.g. the mesh genus can be wrongly represented if the mesh presents numerous small holes). This same issue is present in the work by Guo et al. [8], which solves a discrete Markov Random Field inference problem to minimise the number of keyframes and reduce artifacts, but relies on the error of a mesh deformation method that takes very detailed and accurate mesh sequences. Huang et. al [11] present a keyframe selection system based on pose variance, however their approach relies on accurate skeleton fitting along with image and silhouette priors. While this approach works well for relatively high-quality data, when applied to the noisy data expected from sparse setups the skeleton-optimization approach becomes unreliable. Furthermore the joint-vertex skinning can suffer where the body shape is obscured by loose clothing. Our work instead opts for an autonomous keyframe system based on shape similarity via spherical harmonics descriptors. By using spherical harmonics as an abstract shape descriptor, a shape similarity map can be built that is robust to frequent and disruptive noise in the input sequence.

3. Method Overview

We propose a tracking system that applies spatio-temporal coherence whilst also remaining faithful to the underlying motion and structure of the captured volumetric sequence. This is a challenging task as the input to such a system typically involves a lot of temporal noise, can present high-speed motion and may require demanding shape deformation, especially if the sequences are captured with sparse camera setups. We propose a system which requires no priors other than the input mesh sequence and can be equally evaluated on any volumetric video platform which generates unstructured mesh sequences.

As abundant noise and irregularity can be expected, the proposed method seeks to generate simplistic representations of the input data for some steps of the system via abstraction layers, without the use of model fitting or templates in order to maintain generality. Abstraction layers are generated by detaching the vertex data from the mesh, filtering outliers and small unconnected components, and applying an adaptive isotropic remeshing [25] which results in a quasi-uniformly distributed set of sample points with sufficient density. This creates an abstraction of the input mesh which supports some key aspects of our system such as the preliminary step of automatic keyframe mesh selection driven by shape-similarity (Section 4). They are also used in the following step for establishing dense volumetric correspondences capable of detecting missing geometry and propagating user edits (Section 5). These correspondences drive a sequential registration by means of a deformation graph (Section 6). Finally, we apply a post-processing step in the form of a dynamic 3D Kalman filter applied to mesh vertices tracked across a region (Section 7).

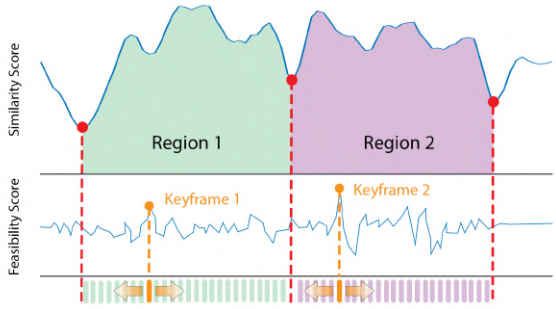

Figure 2. Shape similarity descriptors are used to generate a similarity score for each mesh which is used to define tracking regions. Keyframe meshes are selected by using a feasibility score within regions and tracking is then performed sequentially outwards from the keyframe mesh toward region boundaries.

4. Similarity-Driven Automatic Keyframe Mesh Selection

The goal of the keyframing system is to simultaneously minimize the cumulative error from sequential tracking and select the minimum number of meshes, N, which can encapsulate the shape and motion represented by an unstructured sequence of meshes, $M_{\{1..T\}}$. With this goal in mind we propose a system which partitions $M_{\{1..T\}}$ into sequential groups based on shape similarity. Thus, given a shape-similarity score for all meshes in the sequence which indicates a per-frame similarity to the other meshes, we infer that highly dissimilar frames will introduce errors when attempting to track back against other meshes in the sequence.

The central metric exercised in this process is the shape-similarity score. In order to establish shape similarity in a computationally effective manner, rotation-invariant descriptors, d_i, are generated for each mesh using the spherical harmonic representation system by Kazhdan et. al [14]. With this metric, we compute a similarity matrix among all meshes, $[d_i d_j^\top]_{1 \leq i,j \leq T}$, where the value at $[d_i, d_j]$ is the dot product of d_i and d_j. Mesh similarity score is then defined as the mean value of the matrix per row. To reduce high frequency variance, this one dimensional signal can then be filtered using a moving average filter.

Figure 2 illustrates the process further by plotting a typical similarity score overlaid by the determined tracking regions and keyframe meshes determined as above. From these keyframes the framewise registration will be performed outwardly toward region boundaries. By defining region boundaries on frames with low similarity score we effectively isolate the error that would be introduced by attempting to force dissimilar frames to register to adjacent frames. Despite filtering high-frequency variance in the similarity score, we still employ a fixed minimum separation value λ_{min} between selected minima i.e. region boundaries, which maintains a minimum keyframe to frame ratio.

Within each region, a keyframe must be selected which produces the smallest cumulative error when tracked sequentially towards the region boundaries. Collet et al. [4] propose a *feasibility score* based on hueristically determined characteristics of the mesh topology, specifically the surface area, genus and number of connected components. For noisy input this score is unreliable and incoherent. Instead we apply the score to abstracted representations of the input meshes which filters out high-frequency topology noise and provides coherent input. We further modify the equation to accommodate the larger impact of genus over surface area on keyframe selection and add a negative weight for region boundary proximity to discourage keyframe selection adjacent to tracking region boundaries.

5. Dense Volumetric Correspondences

Given a selection of keyframes and defined regions, the tracking process is performed outwardly from the keyframe up to the region boundaries as shown in Figure 2. Each pair-wise mesh registration is driven by robust, volumetric correspondences and a topologically coherent deformation graph. We use the abstraction-layer meshes on both the source and the target mesh, as a robust framework for matching reliably significant details. The use of abstraction means that the correspondence accuracy and cost is relatively constant regardless of the size of the input.

The abstraction layers are used as the basis to establish dense pairwise correspondences preserving robustness to missing geometry. This is done by volumetrically segmenting them, and performing a series of alignments from the source layer to the target layer via matching segments. To ensure a reasonable alignment there must be consistent segmentation between the source and target abstraction layers, so we need to segment the former and transfer that same segmentation to the latter.

Our approach follows the idea of a pseudo-semantic segmentation, i.e., creating segments at sharp changes in volume which generally resemble the boundaries of joints and limbs. In comparison with traditional animation rigs, this approach is motivated by the idea that articulated motion tends to be most non-rigid at joints and less so along bones.

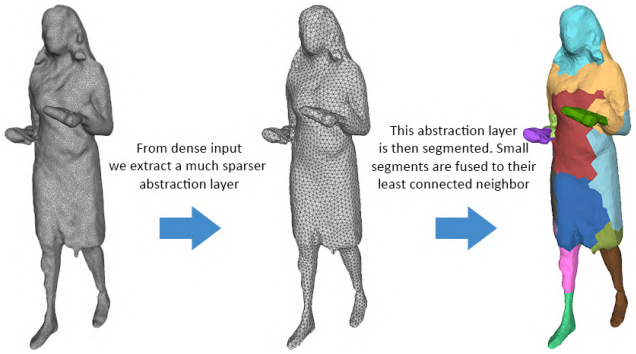

Figure 3. The abstraction and segmentation process as a precursor to segment-based alignment. A typical 25K vertex mesh is reduced to 4.5K and segmented.

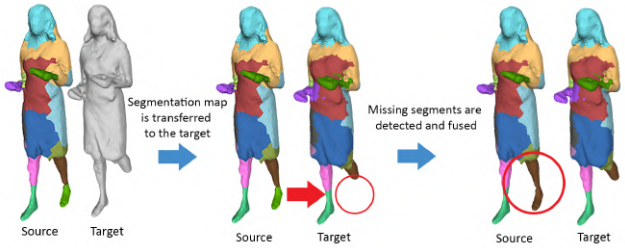

Figure 4. Segmentation map is transferred from the source abstraction layer to the target abstraction layer. Any missing segments are fused and flagged for rigid ICP.

Thus, we prioritise the semi-rigid parts of the mesh to drive the correspondences. The pseudo-semantic segmentation map is created using the shape diameter function as proposed by Shapira et. al [28] and it is organised in a hierarchy from least-connected to most-connected components as a guide for resolving segmentation issues. For example, if the segmentation creates many small components, they are fused to the least-connected neighboring segment. Thus, fusion tends to occur from limb-ends towards the central component. Figure 3 shows the abstraction layer creation for a typical mesh and the segmentation result.

A global rigid ICP alignment is performed between the source and target abstraction layers prior to transferring the segmentation of the source layer to the target abstraction layer. Semi-sparse matches between target and segmented source are then calculated using ICP with strict normal alignment tolerance. Typically, in the case of missing geometry (e.g., a limb or other thin structure) there is a large mismatch in segment size. So we perform a coherence check to compare the size of a segment between the source and target abstraction layers and if a mismatch is detected, the segment is flagged to be fused with its nearest connected neighbor. The flagged segments are recorded and will aligned differently so as to preserve the structure. Figure 4 illustrates typical segmentation transfer from source to target.

Once the segment map has been successfully transferred,

a segment-wise alignment is performed using an augmented version of the Coherent Point Drift (CPD) algorithm [22], applied to the point cloud respresented by the vertices of the meshes. In some cases large segments can be encountered, for example, the central chest region or instances of multiple fused segments. Instead of applying the standard CPD algorithm and encountering performance bottlenecks due to size, we provide the following adaptation to the CPD algorithm which allows for upscaling the alignment that would register two smaller point clouds. This effectively approximates the alignment of a large dataset for the computational cost of a significantly smaller one. If the source and target segment are relatively large clouds S and T respectively, then given some uniformly downsampled clouds s and t, the alignment via standard CPD is given as:

$$s' = s + G_{st}W \tag{1}$$

where the aligned cloud s' is calculated as the input cloud plus the affinity matrix G_{st} times a weighted transformation matrix W, which is solved in the main part of the CPD algorithm. Following this calculation, if G_{st} is replaced by the affinity matrix between s and S i.e. GsS, the alignment can be upscaled to the original size of S by a second application of Equation 1:

$$S' = s' + G_{sS}W \tag{2}$$

Where W is the same transformation matrix solved for in Equation 1. This upscaling naturally simplifies the alignment calculated for W but requires much less computation time. Considering that at a segmentation level the alignment is approximately rigid, so any loss of accuracy due to scaling is negligible. This process is applied to all segments with the exception of those flagged with missing geometry. These segments instead undergo a purely rigid ICP alignment to prevent deforming a segment into a target which is significantly absent. This segment-based alignment of the source abstraction layer to the target abstraction layer can now be used to drive the deformation graph optimization.

6. Deformation Graph Construction and Application

After the first abstraction layer has been coarsely aligned with the target mesh via segment-based registration, a second layer of abstraction is created from the aligned first layer to assist in generating the structure for the deformation graph which will be used to smoothly reshape the source mesh towards the target. In brief, the deformation graph framework consists of a set of nodes evenly distributed about a mesh with edges connecting regions of influence. Each node n represents a rotation R_j and translation t_j for a set of nodes $n_j = n_1..n_J$. Thus, for any particular mesh M of vertices $v_m \epsilon M$, the transformed vertex $v_m^{'}$ is given by:

$$v_m^{'} = \sum_{n_j \epsilon N_{(v_m)}} w(v_j, n_j) \left[R_j(v_m - n_j) + n_j + t_j\right] \tag{3}$$

Where $N_{(v_m)}$ is the set of nodes which influence v_m and $w(v_j, n_j)$ is the skinning weight of a given node towards v_m, following the work of Li et al. [16]. The translations and rotations for each node are found by formulating them as a non-linear optimization problem. We model the optimization problem in this work on the cost function of Guo et al [9], driven by the aforementioned correspondences.

6.1. Detail Synthesis

Regardless of tracking accuracy, the nature of keyframing will introduce popping artifacts as the topology changes across a region boundary. To address this issue, one could attempt to directly re-align the output topology to the temporally coherent fine details in the input sequence as in [15]. This approach works best when the input noise is relatively small and fine surface details deform slowly. Given that the input to our system may exhibit extremely large perturbations due to noise, this approach will produce incoherent results. Instead we opt for a boundary-blending interpolation technique, analogous to deblocking filters used in decompression [17]. Given region sets of $0 < r \leq R$ containing tracked frames r_t, for timesteps $t \in [0..T]$, we perform a boundry-crossing alignment of the last frame in $(r-1)_{t=T}$ to the first frame in $r_{t=0}$ as if it were a normal pair-wise alignment. We then perform a highly non-rigid surface alignment by relaxing the rigidity parameters which creates a detail layer for synthesising surface level details. For each step between the final frame and the keyframe in $(r-1)$ we perform a LERP operation between the detail layer and coarse alignment in order to create a gradient between the deformations. Using cached transformations from the tracking process we can invert and accumulate them as needed to back-project the LERP states to each time step between the last frame in the region and the keyframe. This same process is repeated in the forward direction from $(r-1)$ to r. This approach has the advantage of being completely robust to surface noise as well as significantly reduced computational cost of reverse tracking and fusion due to the reuse of cached transformations.

7. Sequence Smoothing

Temporal noise may still be observed in the final result despite the smooth nature of the as-rigid-as-possible deformation framework. This noise usually takes the form of high frequency *flickering* of the vertex positions and can be visually unappealing. However, given a sequence of meshes which now share the same topology it becomes possible to filter the vertex positions over time against high frequency

noise. To achieve this we apply a standard 3D Kalman filter [13] to the new vertex positions within the calculated regions treating the keyframe as the initial position and each subsequent frame as a set of observations. The transition matrix used is a simple linear motion model for points in 3D Cartesian coordinates in order to maintain complete generality and avoid introducing constraints via any inherent assumptions of a more complex motion model. Regarding the model parameters, a small process noise Q and larger measurement noise R is used such that $R/Q \approx 1e2$, thus prioritizing smoother motion over observations. In practice this Kalman filter can inhibit motion over time and lead to noticeably larger popping effects between keyframes. To reduce this we would like the Kalman filter to be most effective when underlying motion is small and to ignore vertices with large per-frame displacement vectors. To address this we perform an offline motion dynamics analysis per vertex and use the displacement deltas to negatively impact the model correction. To this effect we reduce the lag of "genuine motion" and apply the filter in an adaptive manner.

8. Results

In the following section we validate the proposed method with quantitative, qualitative and ablation studies. We evaluate the keyframe selection metric in comparison to the feasibility score heuristic presented by Collet et. al [4]. We also assess the accuracy of the proposed correspondence and deformation framework against the state of the art using numerous challenging sequences, free from temporal noise as a baseline for ground-truth evaluation. Furthermore, we perform qualitative evaluation of several sequences with different levels of noise and artifacts, captured with sparse multi-view setups. Finally, we demonstrate the application of the geometry recovery, edit propagation and smoothing aspects through realistic examples.

8.1. Keyframing

To evaluate our proposed method for keyframe selection we illustrate the results of the similarity score compared to the feasibility metric proposed by [4] when applied to a challenging sequence with drastic topology changes, Figure 5. Furthermore, this sequence was captured in a budget studio using 12 RGB cameras and contains a lot of structured noise. We demonstrate the tracking results for this sequence using the proposed keyframe sequence against the greedy-selection algorithm proposed by Collet et. al [4]. The proposed approach produces smaller error while significantly reducing the number of keyframes needed, Table 1.

8.2. Tracking Evaluation

We evaluate the performance of our system against two state of the art approaches which best represent common

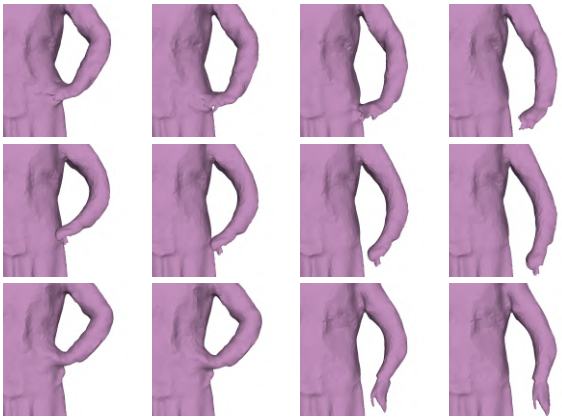

Figure 5. Autonomous keyframe selection: (top) input from a sequence featuring many similar topology changes. (mid) proposed algorithm which identifies a keyframe at t>t+3 and tracks from t>3 toward t. (bottom) the system of Collet et. al [4] which attempts to resolve the geometry change by stretching before eventually giving up and creating a new keyframe at t=t+2.

	Ours	Collet et. al [4]
Max Error	**0.0651**	0.0662
Median Error	**0.0205**	0.0208
# Keyframes	**13**	19

Table 1. Keyframe evaluation on twirl sequence containing 170 frames with large topological changes and fast motion. Errors correspond to Hausdorff distance in relative units.

Figure 6. Detail Synthesis: (A) and (B) show a topology change where tracking regions meet. (A) uses the temporal detail synthesis of Li et. al [15] while (B) is the proposed method.

techniques in surface-based non-rigid registration. The most general of which being Amberg et. al [1] which is applicable to any type of surface or motion and attempts to iteratively solve vertex positions globally with locally varying "stiffness". Lately, however more systems closely resemble that of [9], iteratively solving point-to-plane correspondence driven deformation graphs. To objectively evaluate the performance of our method we use the dataset from Vlasic et. al [32] which features mesh sequences generated by animating a pre-defined template. In this way the input can be considered free from reconstruction artefacts which establishes a reliable reference point for common error metrics like Hausdorff distance [2]. We also present qualita-

Sequence	Max Error			Median Error		
	Ours	[9]	[1]	Ours	[9]	[1]
Crane	0.0424	**0.3432**	0.3753	**0.0019**	0.0338	0.0278
Jumping	**0.2002**	1.4723	0.3549	**0.0019**	0.0382	0.0145
Bouncing	**0.0891**	0.9982	0.4234	**0.0027**	0.0565	0.0151
Handstand	**0.0054**	0.6450	0.1706	**0.0009**	0.0023	0.0032
Swing	0.2386	0.4298	**0.0813**	**0.0031**	0.0185	0.0074

Table 2. Ground-truth evaluation of tracking. Figures are relative to the scale of the input data. Results are given as Maximum Hausdorff Error (max) and Median Hausdorff Error (med).

tive results of each approach applied to a mix of the above dataset as well as volumetric data captured from multi-view capture setups. Furthermore, we demonstrate the ability of our system to propagate user edits and recover lost geometry by conducting experiments which would replicate some expected user edits or volumetric capture failure modes.

8.2.1 Ground Truth Evaluation

For a fair evaluation of the tracking error introduced by each system, each dataset was given the same keyframes and tracking regions. In this way the error metric provides a direct indication of the correspondence robustness and deformation fidelity. The results of Table 2 shows that our system introduces fewer errors in multiple ground-truth sequences which exhibit highly dynamic and varying motions.

8.2.2 Qualitative Evaluation

It can be seen from Figure 7 that where fast motion is concerned, the proposed system shows robustness in both correspondence matching and large deformation. In contrast to [9] the use of volumetric correspondences over standard normal-constrained ICP methods allows for reliable matching along fast pose changes. The as-rigid-as-possible deformation constraint prevents any large pose changes in [9] despite the likely errors in correspondences resulting in either largely unchanged poses or extreme deformations where the solver struggled to converge. This is evident in (b) for all cases of Figure 7. In contrast, the naive global deformation of [1] exhibits very little robustness to bad correspondences and can compress thin structures due to fast motion. This is most clearly seen in the hands and feet in (c) where we see a larger range of motion has led to surface compression due to nearest-neighbour correspondences.

8.2.3 Persistent Geometry Evaluation

We demonstrate the ability of our system to recover and propagate pertinent features in some conventional and challenging sequences captured from multi-view volumetric systems. In particular Figure 8 illustrates a sequence which was highly occluded and contained a fast moving football being volleyed. Large sections of the mesh exhibit intermittent missing portions as well as difficulty reconstructing the ball, sometimes across many sequential frames. Our geometry aware system was able to retain important features including the ball, while still registering to the underlying motion. In comparison, template or skeleton-based approaches are simply unable to track foreign objects without manual intervention.

We further illustrate geometry propagation in Figure 9 as well as a sample case for user edits. In such a case the reconstruction failed to recover the finger detail in the hand of the actor (top right). The user may edit the nearest keyframe(s) and manually restore the data in any 3D modelling software. Afterwards, the system inherently detects the absent geometry through the tracking process and will propagate the edit throughout the frames influenced by the given keyframe. The system is also capable of much larger edits such as the addition of props. The added geometry becomes rigidly tracked along with the nearest connected component and thus it realistically follows the underlying motion while maintaining intact structure.

8.3. Detail Synthesis

We compare our detail synthesis approach to that of Li et. al [15] which was subsequently used by Guo et. al [9] and present the results in Figure 6 of a typical noisy sequence from a sparse camera studio setup. The benefits of the proposed boundary-aware detail synthesis can be seen as a smoother transition across frames while the approach of Li et. al [15] produces a sharp boundary transition with large topology changes, resulting in noticeable popping effects. In addition, the proposed method is robust to input noise as it only seeks to smooth tracking region boundaries while the synthesis of Li et. al [15] manifests input noise in the hands and hair.

8.4. Smoothing

Smoothing not only helps to reduce high-frequency, *flickering* motions, it also improves the quality of propagated user edits and recovered geometry without the need for expensive 3D flow. Referring back to the recovered fast-moving football in Figure 8 (Right, light blue), the motion of the ball becomes static in the recovered frames from having no connected reference segment to propagate to. The smoothing filter helps to interpolate the motion between the static frames and the next observation of the ball. Figure 8 (Right, dark blue) illustrates the ablation results where the smoothing process can help interpolate the missing motion. Thus, the smoothing and interpolating motion greatly improves the temporal coherence of the end result.

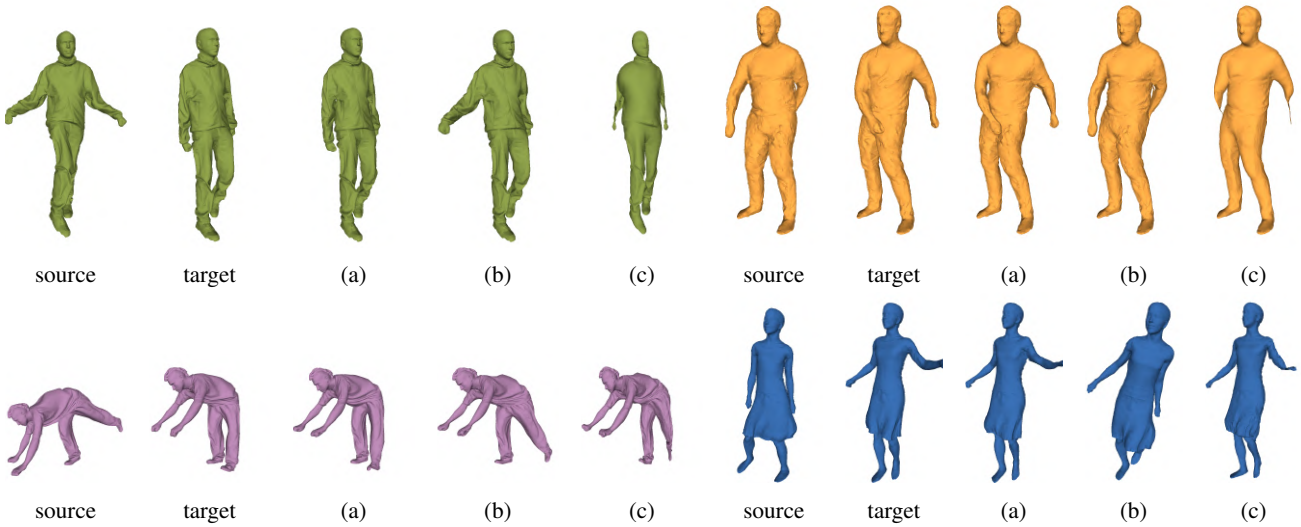

Figure 7. Qualitative results of some challenging sequences containing fast motion. Presented for each sequence are: the source, final target, (a): the proposed method, (b): Guo et al. [9], (c): Amberg et al.[1]. In each case the results are the output of successively tracking the frames between the source and target.

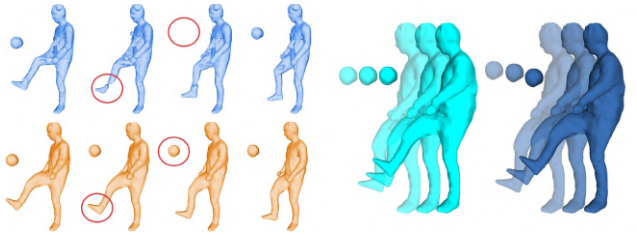

Figure 8. Fast moving objects can be lost or cause occlusions (left, top row). The proposed system can track multiple moving objects and provide geometry recovery (left, bottom row). Pictured right (light blue) are 3 successive frames tracked without motion smoothing. Pictured right (dark blue), the same 3 frames where interpolation has occurred as a result of motion smoothing.

9. Conclusions and Future Work

We present a robust autonomous tracking algorithm which can detect discrepancies in input data and can propagate pertinent geometry. The system outperforms the state of the art for available datasets and requires no priors of the input sequence. Dense volumetric correspondences through shape abstraction provide an indiscriminate shape registration framework which is robust to large or fast motions. Furthermore, our system allows for drastic alterations of the input mesh which can be reliably integrated with the underlying motion, enabling a new domain for creative freedom and post-production. While the presented approach is robust and achieves large data reductions, it still requires keyframes which is a larger workload than solving for a global template. It would be desirable to extend this work to create a global template without resorting to the constraints of parametric templates or pre-defined animation rigs. Also,

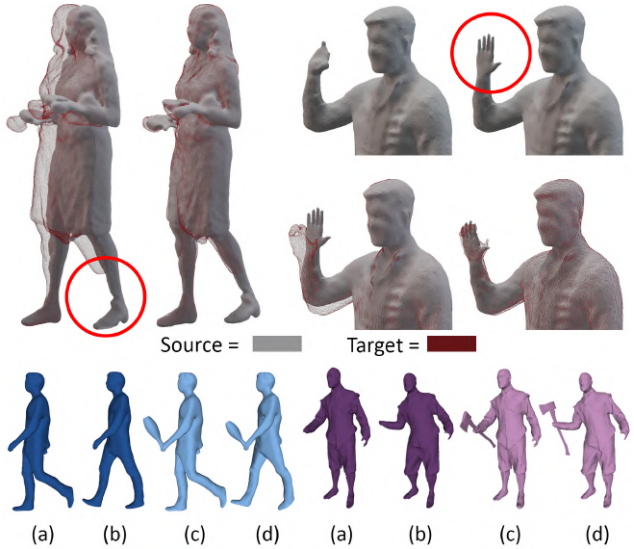

Figure 9. Geometry recovery & propagation. Top left: A missing leg is recovered from a walking sequence. Top right: A user manually restores the hand to a keyframe which is then propagated. Bottom: User edits may also be extreme additions such as props.(a) source, (b) target, (c) edited source (d) propagated to target over multiple frames

while 3D Kalman smoothing produces visually appealing results, it could likely be improved upon with further exploration of its many evolutions. It is hoped that this approach may inspire further work towards low-end, cost-effect volumetric video such that the popularity of the medium may continue to flourish beyond niche groups within academia and industry toward the creative communities.

10. Acknowledgements

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under the Grant No. 15/RP/2776 and from equipment provided by NVIDIA. We would also like to thank Dr. Mairéad Grogan for her assistance with CPD-ICP and Vologrames for providing volumetric sequences.

References

[1] Brian Amberg, Sami Romdhani, and Thomas Vetter. Optimal step nonrigid icp algorithms for surface registration. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[2] Nicolas Aspert, Diego Santa-Cruz, and Touradj Ebrahimi. Mesh: Measuring errors between surfaces using the hausdorff distance. In *Proceedings. IEEE international conference on multimedia and expo*, volume 1, pages 705–708. IEEE, 2002.

[3] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[4] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (ToG)*, 34(4):1–13, 2015.

[5] Mingsong Dou, Henry Fuchs, and Jan-Michael Frahm. Scanning and tracking dynamic objects with commodity depth cameras. In *2013 IEEE international symposium on mixed and augmented Reality (ISMAR)*, pages 99–106. IEEE, 2013.

[6] Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, et al. Fusion4d: Real-time performance capture of challenging scenes. *ACM Transactions on Graphics (TOG)*, 35(4):1–13, 2016.

[7] Mingsong Dou, Jonathan Taylor, Henry Fuchs, Andrew Fitzgibbon, and Shahram Izadi. 3d scanning deformable objects with a single rgbd sensor. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 493–501, 2015.

[8] Kaiwen Guo, Peter Lincoln, Philip Davidson, Jay Busch, Xueming Yu, Matt Whalen, Geoff Harvey, Sergio Orts-Escolano, Rohit Pandey, Jason Dourgarian, et al. The relightables: volumetric performance capture of humans with realistic relighting. *ACM Transactions on Graphics (TOG)*, 38(6):1–19, 2019.

[9] Kaiwen Guo, Feng Xu, Yangang Wang, Yebin Liu, and Qionghai Dai. Robust non-rigid motion tracking and surface reconstruction using l0 regularization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3083–3091, 2015.

[10] Marc Habermann, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. Deepcap: Monocular human performance capture using weak supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, jun 2020.

[11] Chun-Hao Huang, Edmond Boyer, Nassir Navab, and Slobodan Ilic. Human shape and pose tracking using keyframes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3446–3453, 2014.

[12] Zeng Huang, Tianye Li, Weikai Chen, Yajie Zhao, Jun Xing, Chloe LeGendre, Linjie Luo, Chongyang Ma, and Hao Li. Deep volumetric video from very sparse multi-view performance capture. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 336–354, 2018.

[13] R Kalman. A new approach to linear filtering and prediction problems. *J. Basic Eng., Trans. ASME, D*, 82:35–45, 1960.

[14] Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. Rotation invariant spherical harmonic representation of 3 d shape descriptors. In *Symposium on geometry processing*, volume 6, pages 156–164, 2003.

[15] Hao Li, Bart Adams, Leonidas J Guibas, and Mark Pauly. Robust single-view geometry and motion reconstruction. *ACM Transactions on Graphics (ToG)*, 28(5):1–10, 2009.

[16] Hao Li, Robert W Sumner, and Mark Pauly. Global correspondence optimization for non-rigid registration of depth scans. *Computer graphics forum*, 27(5):1421–1430, 2008.

[17] Peter List, Anthony Joch, Jani Lainema, Gisle Bjontegaard, and Marta Karczewicz. Adaptive deblocking filter. *IEEE transactions on circuits and systems for video technology*, 13(7):614–619, 2003.

[18] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015.

[19] Matthew Moynihan, Rafael Pagés, and Aljosa Smolic. A self-regulating spatio-temporal filter for volumetric video point clouds. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics*, pages 391–408. Springer, 2019.

[20] Matthew Moynihan, Rafael Pagés, and Aljosa Smolic. Spatio-temporal upsampling for free viewpoint video point clouds. In *VISIGRAPP*, pages 684–692, 2019.

[21] Armin Mustafa, Hansung Kim, Jean-Yves Guillemaut, and Adrian Hilton. Temporally coherent 4d reconstruction of complex dynamic scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4660–4669, 2016.

[22] Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE transactions on pattern analysis and machine intelligence*, 32(12):2262–2275, 2010.

[23] Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 343–352, 2015.

[24] Rafael Pagés, Konstantinos Amplianitis, D Monaghan, J Ondřej, and A Smolić. Affordable content creation for free-viewpoint video and vr/ar applications. *Journal of Visual Communication and Image Representation*, 53:192–201, 2018.

[25] Nico Pietroni, Marco Tarini, and Paolo Cignoni. Almost isometric mesh parameterization through abstract domains. *IEEE Transactions on Visualization and Computer Graphics*, 16(4):621–635, 2009.

[26] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2304–2314, 2019.

[27] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 84–93, 2020.

[28] Lior Shapira, Ariel Shamir, and Daniel Cohen-Or. Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer*, 24(4):249, 2008.

[29] Miroslava Slavcheva, Maximilian Baust, Daniel Cremers, and Slobodan Ilic. Killingfusion: Non-rigid 3d reconstruction without correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1395, 2017.

[30] Miroslava Slavcheva, Maximilian Baust, and Slobodan Ilic. Sobolevfusion: 3d reconstruction of scenes undergoing free non-rigid motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2646–2655, 2018.

[31] Robert W Sumner, Johannes Schmid, and Mark Pauly. Embedded deformation for shape manipulation. In *ACM SIGGRAPH 2007 papers*, pages 80–es. ACM New York, NY, USA, 2007.

[32] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. Articulated mesh animation from multi-view silhouettes. In *ACM SIGGRAPH 2008 papers*, pages 1–9, 2008. Accessed June 2020 `http://people.csail.mit.edu/drdaniel/mesh_animation/`.

[33] Donglai Xiang, Hanbyul Joo, and Yaser Sheikh. Monocular total capture: Posing face, body, and hands in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10965–10974, 2019.

[34] Weipeng Xu, Avishek Chatterjee, Michael Zollhöfer, Helge Rhodin, Dushyant Mehta, Hans-Peter Seidel, and Christian Theobalt. Monoperfcap: Human performance capture from monocular video. *ACM Transactions on Graphics (ToG)*, 37(2):1–15, 2018.

[35] Tao Yu, Zerong Zheng, Kaiwen Guo, Jianhui Zhao, Qionghai Dai, Hao Li, Gerard Pons-Moll, and Yebin Liu. Doublefusion: Real-time capture of human performances with inner body shapes from a single depth sensor. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7287–7296, 2018.

[36] Qing Zhang, Bo Fu, Mao Ye, and Ruigang Yang. Quality dynamic human body modeling using a single low-cost depth camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 676–683, 2014.

[37] Zerong Zheng, Tao Yu, Hao Li, Kaiwen Guo, Qionghai Dai, Lu Fang, and Yebin Liu. Hybridfusion: Real-time performance capture using a single depth sensor and sparse imus. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 384–400, 2018.

[38] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (ToG)*, 33(4):1–12, 2014.