Hybrid Skeleton Driven Surface Registration for Temporally Consistent Volumetric Video

João Regateiro Marco Volino Adrian Hilton
Centre for Vision, Speech and Signal Processing
University of Surrey
{j.regateiro, m.volino, a.hilton} @surrey.ac.uk

Abstract

This paper presents a hybrid skeleton-driven surface registration (HSDSR) approach to generate temporally consistent meshes from multiple view video of human subjects. 2D pose detections from multiple view video are used to estimate 3D skeletal pose on a per-frame basis. The 3D pose is embedded into a 3D surface reconstruction allowing any frame to be reposed into the shape from any other frame in the captured sequence. Skeletal motion transfer is performed by selecting a reference frame from the surface reconstruction data and reposing it to match the pose estimation of other frames in a sequence. This allows an initial coarse alignment to be performed prior to refinement by a patch-based non-rigid mesh deformation. The proposed approach overcomes limitations of previous work by reposing a reference mesh to match the pose of a target mesh reconstruction, providing a closer starting point for further non-rigid mesh deformation. It is shown that the proposed approach is able to achieve comparable results to existing model-based and model-free approaches. Finally, it is demonstrated that this framework provides an intuitive way for artists and animators to edit volumetric video.

1. Introduction

Motion capture (MoCap) using optical markers is the gold standard for human performance capture, and is widely used in industry, but only captures a low-dimensional representation of the human body, i.e. joint positions/angles. It requires a huge effort from artists to create believable character animations from MoCap data. To overcome this drawback, multiple view video capture techniques have evolved towards simultaneously capturing shape, motion and appearance.

Modern capture systems generally consist of a set of synchronised cameras that simultaneously capture the scene from different viewpoints [10, 11, 28]. Feature tracking and stereo matching techniques are applied to the data to create a 3D reconstruction of both static and dynamic scenes. 3D Performance Capture results in a sequence of 3D geometry objects of the actor or scene, integrating all visual features, such as dynamic shape, motion, and texture appearance. However, a problem arises where for each of the captured frames the geometry is temporally inconsistent, i.e. the shape topology and vertex connectivity are varying.

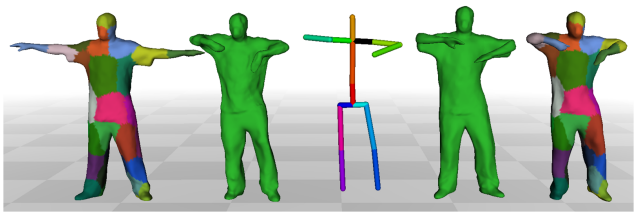

Figure 1. Processing stages of the proposed method: (left to right) Source mesh, target mesh reconstruction, 3D pose, reposing of source mesh using linear blend skinning and proposed aligned result.

Post-processing methods suffer from temporally inconsistent geometry, making it extremely difficult to propagate changes throughout a sequence. This content challenges most existing methods, as a consequence of the diversity in human motion and shape, and dynamic surfaces; it demonstrates difficulty on dense point-to-point correspondence schemes. Artists very often manually manipulate content to achieve the desired pose, or correct the geometry, spending significant work-time on such laborious task.

This work proposes an automated hybrid framework to generate temporally consistent reconstructions from dynamic mesh sequences. The core idea of the proposed approach is to embed a skeleton into a surface reconstruction and repose the shape prior to geometric alignment, see Figure 1. By performing a coarse alignment, as a pre-process, the proposed approach is able to handle large changes in pose between a reference and target shape. We also demonstrate that the resulting temporally consistent mesh representation can be used to facilitate keyframe-based editing for volumetric video.

The contributions of this paper can be summarised as follows:

1. A 4D surface tracking framework to temporally align mesh surfaces with inconsistent topology and difference in shape. The framework successfully maps a mesh surface onto a target using only geometry information.

2. A hybrid skeleton-driven surface registration (HS-DSR) to overcome the limitations of the 4D surface tracking framework, achieving more reliable and accurate results for complex mesh sequences. It provides results for longer sequences without the increase or propagation of errors generated in previous frames.

3. Keyframe-based editing for volumetric video to allow intuitive editing and propagation through a sequence, it generates novel sequences avoiding the need to recapture new datasets, maintaining surface detail and integrity.

2. Related Work

Human Motion Capture: Skeleton based techniques arise from the necessity of simulating human body motion. MoCap [22] estimates human movement based on tracking its skeleton using markers placed on key locations. Recent work from Wei *et al.* [32] introduces deep learning techniques to overcome a registration problem between complete or partial 3D models. They successfully demonstrate the benefits that deep learning offers to track human body poses with a real-time human body correspondence framework.

Convolutional Pose Machines (CPMs) [8] learn how to associate body parts to human-like shapes from a single image and estimate 2D human joints locations for multiple people. This allows more freedom on the type of data captured, because it does not rely on physical trackers, therefore the system is able to preserve all dynamics from the captured scene, such as clothing deformation and fine detail. A central challenge in computer vision over the past two decades has been markerless human motion capture from video [16, 21, 29, 33].

Surface reconstruction from multiple view video capture generates unstructured mesh sequences with both the vertex connectivity and geometry changing from frame-to-frame [11, 28]. To overcome this issue, methods for temporal alignment of mesh sequences have been introduced to obtain sequences with temporally consistent mesh structure [4, 6, 31]. Model-based techniques allow for a better mesh representation, including fine detail and body shape characteristics, however it requires prior information computations and it cannot reproduce accurately the scene captured, such as clothes and hair dynamics [31, 33]. Model-free approaches overcome limitations raised from model-based, obtaining a better representation of the captured scene, as result of not being tied to a parameterized template, however this raises problems such as, surface drifting, the accumulation of errors, and the inability of tracking two significantly different poses. Temporally inconsistent geometry does not ease the manipulation of mesh sequences, increase physical storage space, and requires a huge effort to create animation.

Geometry Alignment: Geometry alignment techniques require the establishment of correspondences between frames so it can correctly align meshes for every frame [2, 7, 9, 30]. Lipman and Sorkine *et al.* [19, 26, 34] propose a differential coordinate representation of a surface mesh, which preserves local detail information in the presence of large mesh deformations.

Huang *et al.* [15] presents a volumetric approach for 3D shape tracking that uses centroidal Voronoi tessellation representation to build a feature space where trained random forests return optimal correspondences. However, volumetric techniques tend to preserve the volume from frame to frame, therefore if the volume has a significant change the technique is not able to adapt. Coarse-to-fine matching demonstrates success for non-rigid registration [5, 24] across complex sequences, and is achieved by targeting subsets of a surface area and interactively decreasing its radius, preserving fine details such as dynamic clothing. These approaches focus on deforming a single reference frame through subsequent frames of the sequence, leading to the increase of artefacts on large changes in shape.

Non-sequential alignment approaches have also been investigated [3, 4, 17, 23], providing robust techniques that are able to handle larger non-rigid deformations in consecutive meshes of the reconstructed captured sequences. However, the shape comparison in the above techniques is computationally expensive and requires a full sequence as a prior, disallowing online applications.

3. Hybrid Skeletal-driven Surface Alignment

The objective of the proposed approach is to extract a temporally consistent mesh structure from multiple view video of a human actor. We estimate a 3D surface at each time instance t using a model-free reconstruction pipeline, described in Section 3.1, resulting in a temporally inconsistent mesh representation for a captured sequence. A 2D joint detector is utilised to first estimate 2D skeletal pose in each camera view at every time instance t. 2D joint locations are triangulated to estimate the 3D pose using a kinematic solver, described in Section 3.2. 3D skeletal pose is used to rig the reconstruction for a reference frame from the sequence to obtain an animatable representation of the specific person and their clothing. The animated reference

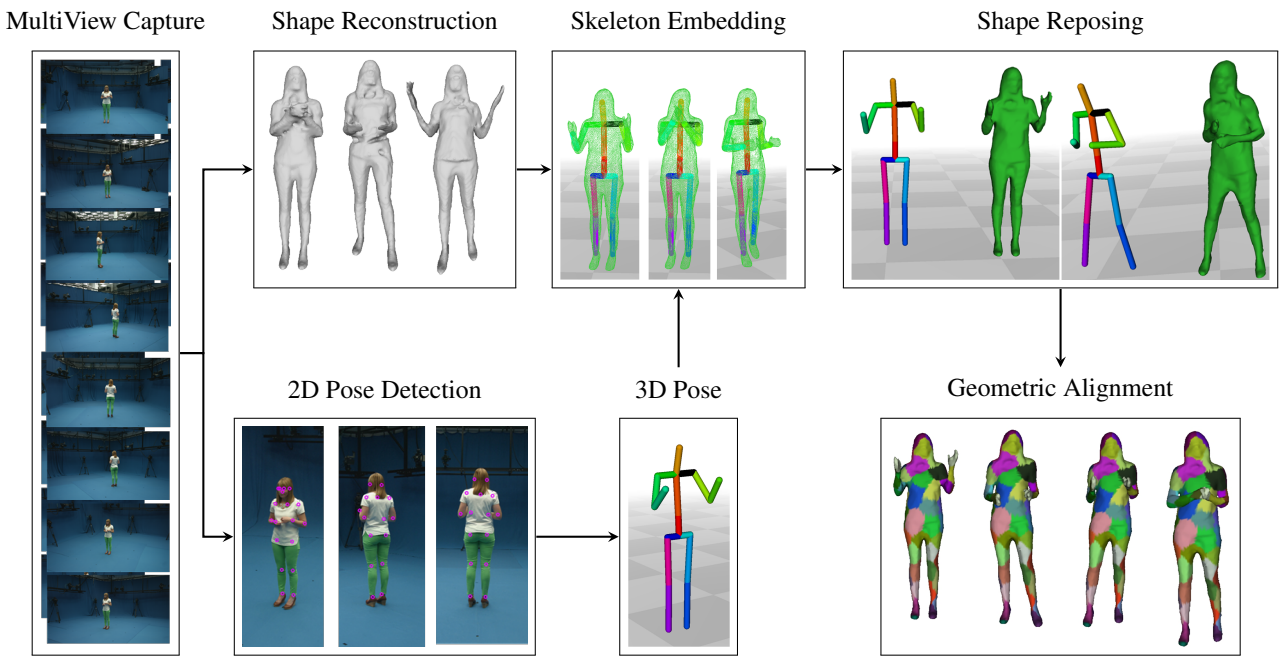

Figure 2. Proposed Hybrid Skeleton-driven surface registration pipeline, going from left to right, we have a multiple view capture system to retrieve shape and appearance. Followed by 2D pose detection and shape reconstruction, to retrieve 3D skeletal and surface vertex correspondence. Finalising with shape reposing to provide an initial pose estimation to the geometric alignment method.

frame is reposed using the 3D skeleton estimate for each frame to obtain a course alignment between the reference frame and target reconstruction, described in Section 3.3. Non-rigid patch-based Laplacian deformation is then used for detailed temporal alignment through non-rigid deformation of the reference surface, described in Section 3.4. This allows the correct alignment in sequences that include large changes in pose from frame-to-frame due to fast motion. An overview of the proposed hybrid skeleton-driven surface registration pipeline is shown in Figure 2.

3.1. Multiple View Surface Reconstruction

Input into our method is synchronised multiple view video $\{\{I_c^t\}_{c=1}^{N_C}\}_{t=1}^{N_T}$ recorded from N_C calibrated cameras $\{C_c\}_{c=1}^{N_C}$, where N_T is the number of frames, of a human actor in a controlled studio, see Figure 2. Shape reconstruction is performed by first extracting foreground silhouettes via chroma-keying allowing a visual hull [18] reconstruction at each time instance t. The visual hull is then refined using a volumetric graph-cut by matching stereo features across camera images $\{\{I_c^t\}_{c=1}^{N_C}\}$ to make the geometry photo-consistent [28]. Stereo refinement adds geometric detail to concave regions of the surface which cannot be recovered using visual hull reconstruction. This process results in a set of temporally incoherent meshes which have been independently reconstructed at each time instance.

3.2. 2D/3D Pose Estimation

2D Joint Detection: Given a set of captured images at a single time instance, $\{\{I_c^t\}_{c=1}^{N_C}\}$, we use a 2D pose detector to locate the skeletal joints of the subject. In the proposed pipeline, 2D joint detections are given by Convolutional Pose Machines (CPM) [8] which have been shown to be a robust 2D pose detector trained on a wide variety of subjects and poses. The detected 2D joint $j_i^c = \{p_i^c, l_i^c\}$ consists of a 2D pixel location p_i^c and confidence score l_i^c.

Joint Triangulation: Triangulation of each joint requires the detected 2D joint locations of the ith joint from each camera $\{j_i^c\}_{c=1}^{N_C}$, the camera intrinsic parameters K_c, rotation matrix R_c and translation vector t_c for all N_C cameras. We seek to find 3D position $\hat{J}_i$ for the ith joint that minimises the re-projection error in all camera against the detected 2D joints $\{j_i^c\}_{c=1}^{N_C}$. The 3D joint position $\hat{J}_i$ is optimised according to Equation 1. This incorporates the joint confidence l_i^c to make it robust against errors in joint detection. In the case that a joint is not detected, $l_i^c = 0$.

$$\arg\min_{\hat{J}_i} \sum_{c=1}^{N_C} l_i^c \| P(c; \hat{J}_i) - p_i^c \| \tag{1}$$

where p_i^c and l_i^c are the 2D detected joint location and joint confidence, respectively, for the ith joint in the cth camera. $P(c; \hat{J}_i)$ is the projection function which maps a 3D joint position into the camera image plane $P(c; \hat{J}_i) = K_c R_c \hat{J}_i +$

t_c. $\hat{j}_i$ is the optimised 3D position of the ith joint.

The collection of 3D joints positions $\hat{j}_i$ represent a skeletal structure at a time instance t. We define a skeletal structure as containing 16 3D joint positions $\hat{j}_i$ and 15 bones $\hat{b}_{ij}$. The bone $\hat{b}_{ij} = \{\hat{j}_i, \hat{j}_j\}$ is the relation between the ith and j^{th} joints. This leads to a hierarchy relationship between bones to create a kinematic structure that allows for skeleton animation.

Kinematic Structure: A kinematic structure can be represented as a bi-directional graph, where nodes represent 3D joint positions $\hat{j}_i$, and edges represent bones $\hat{b}_{ij} = \{\hat{j}_i, \hat{j}_j\}$. Each node has an associated a rotation R_i and translation t_i, defined as a transformation $T_i = R_i(\ )t_i(\ )$, where $R_i(\ )$ is an axis-angle rotation of degrees, and $t_i(\ )$ is a translation of the vector . Consequently, the joint transformation T_i is represented as a 4 4 transformation matrix at each time instance.

Let us consider the skeletal sequence denoted $\hat{S}$ as a collection of skeletal poses that represent all the 3D joint positions $\hat{j}_i$ and bones $\hat{b}_{ij} = \{\hat{j}_i, \hat{j}_j\}$ for the captured sequence. Hence, a skeletal structure at a time instance t is defined as $\hat{s}_t = \{\hat{j}_i, \hat{b}_{ij}\}$. The sequence $\hat{S}$ can only inform about the joints location and relationship, therefore to produce animation it is necessary to calculate the joint transformations that maps one frame $\hat{s}_t$ into $\hat{s}_{t+1}$. The kinematic structure transforms an initial pose onto $\hat{s}_t$, translating the motion from the captured sequence $\hat{S}$ to kinematic skeletal joint motion. The initial pose is a kinematic structure, composed of 16 nodes N_N and 15 edges N_B. Each edge has nodes associated, describing the dependency between nodes; nodes contain a transformation matrix T_i to calculate their spatial transformation from frame-to-frame relative to its parent.

Kinematic Motion: To represent kinematic motion, we select an initial frame $\hat{s}_t$ and initialise pose as the reference pose. This work selects the first frame of the sequence as the initial pose . Once is solved for the first $\hat{s}_t$ frame, we calculate an initial list of nodes transformation T_i that relate to the first frame. Consequently, it is possible to map into any $\hat{S}$ given the initial estimation of the position and orientation. The pose is solved by calculation node transformations that satisfy the constraints given by $\hat{s}_t$.

The node transformation is solved using an inverse and forward kinematic chain approach, Equation 2, where $(\hat{b}_B)$ is a set of parent nodes of the bone $\{\hat{b}_b\}_{b=1}^{N_B} = \hat{b}_{ij}$, and $[R_{\hat{b}_b} | t_{\hat{b}_B}]$ is the transformation matrix of the bone $\hat{b}_B$, forward kinematics uses the opposite direction of transformation, i.e. the transformation is propagated from a node to all its descendents. Firstly, the global transformations of key joints are retrieved, using the hip joint as the root transformation relative to the initial pose . Secondly, we identify the rotation of the upper body, i.e. the rotation of the shoulders given by its parent joint - the neck in this instance. Finally, we retrieve the head orientation to be able to change the head rotation independent from the rest of the body. The remaining limbs are solved using inverse kinematics to give an initial estimate of their possible locations using the extremity joint as end-effectors. This approach allows for errors caused by incorrect or non-existent 2D detection.

The final step is a refinement of joint positions using forward kinematics to eliminate ambiguities caused by the previous step. The different skeleton poses are represented with the initial pose and a collection of node transformations T_i which allows the reposing of to all frames in $\hat{S}$. For every frame $\hat{s}_t \ 2 \ \hat{S}$, pose is mapped onto $\hat{s}_t$ using = $\%(\hat{s}_t; T_i)$, where $\%(\hat{s}_t; T_i)$ solves T_i for the new set of $\hat{s}_t$ constraints using an inverse and forward kinematic chain approach.

$$(\hat{b}_b) = \prod_{\hat{b}_b 2 (\hat{b}_b)}^{N_B} \left[\begin{array}{c|c} R_{\hat{b}_b} & t_{\hat{b}_b} \\ 0 & 1 \end{array}\right] \tag{2}$$

This gives a sequence of skeletal poses based on joint rotations from the estimated skeletal joint positions obtained by combining multiple view 2D pose using equation 1

3.3. Skeleton Embedding and Shape Reposing

Rigging and animating a reference frame mesh is achieved using an automated rigging framework [1]. Given a template reference frame and its skeleton reference pose, skinning weights for every vertex are estimated according to the set of nearest bones. Once the weights are calculated, we choose to use linear blend skinning (LBS) to repose the reference mesh. The animation framework receives as input an initial pose , a collection of joint transformations T_i and the vertex weights. This allows the reference mesh to be reposed according to the detected 3D pose for each frame in the sequence to obtain an initial coarse mesh approximation of the surface shape at that frame. This is then refined through geometric alignment to represent the detailed shape for each frame.

3.4. Geometric Alignment

This section describes a pipeline that converts an unstructured mesh sequence into a temporally consistent mesh sequence. The pipeline takes as input the reposed reference mesh obtained from the animation framework, Section 3.3. Temporally consistent meshes are produced by optimising for rigid-registration and non-rigid deformation using a Laplacian deformation.

Rigid Surface Registration: Geometric surface registration receives as input the initial animation mesh vertices N_P, and will iteratively find an optimal rigid transforma-

tion $T_p = \{R_p; t_p\}_{p=1}^{N_P}$, where R_p is a $3 \quad 3$ rotation matrix and t_p is a translation that aligns two sets of 3D vertices $\{P_p\}_{p=1}^{N_P}$ defined as patches. The following section describes the approach taken to refine transformation that overlaps the two data sets: Patch-based Iterative Closest Point (PbICP) is employed to estimate the rigid transformation from nearest point correspondence [4, 6].

Patch-based Iterative Closest Point Framework: PbICP consists of three processes: ICP, registration and correspondence search. The objective of this framework is to guarantee the best fitting between a patch and a target mesh. Patches are generated using a geodesic variant of the Lloyd's algorithm [20] which partitions the mesh into hierarchy of regions, and reshapes the data until it has uniformly-sized convex cells.

The patches maintain local mesh detail and allow for re-positioning onto the target surface, thus creating a new pose with minimal loss in surface detail. The registration is performed using a variant of ICP [14, 25, 35] represented by Equation 3, and guarantees to return the best transformation between two meshes. The correspondence search computes the closest point from all source N_A to target N_F vertices within the patches, and vice-versa, taking into account the vertex normals and distances. Thus, the search area is constrained to be the best approximation to the patch. To stop the rigid transformation from drifting, a constraint on the correspondence search is introduced, rejecting vertex matches where the difference in normal angle is greater than $50 \quad$.

$$E = \arg\min_{R_p; t_p} \sum_{a=1}^{N_A} \min_{f \in N_F} w_a \| R_p p_{pa} + t_p \quad q_f \|^2 \qquad (3)$$

where w_a is the weight corresponding to the residual distance between source $\{p_{pa}\}_{a=1}^{N_A}$ and target points $\{q_f\}_{f=1}^{N_F}$.

Equation 3 estimates an optimal transformation that minimises the error for a patch has been found. This is done by re-calculating the closest points on every iteration, allowing the patch to find different correspondences throughout the fitting process.

Laplacian Deformation Framework: The Laplacian consists of differential coordinates to represent a mesh surface [19, 34]. Differential coordinates [12, 13] represent local geometric detail. The PbICP results in an approximation of the target mesh, although rigid displacement tends to cause artefacts and remove the integrity. The Laplacian deformation regularises the mesh to its original shape condition and fit to the target mesh. To solve the Laplacian deformation problem, we first constrain a group of vertices so we can find a least squares solution for the unconstrained vertices, allowing the mesh M to deform.

$$\hat{v}_u = \arg\min_{v_u} \| L v_u \quad (v_k) \|^2 + \| W_c (v_u \quad v_k) \|^2 \qquad (4)$$

where v_u is the set of unknown deformed vertex locations to be computed and v_k is the known vertex locations. The vertices v_k can also represent hard constraints, depending on if the vertex is predefined to be exactly transformed to its target position. Consequently each element of v_k is defined with the following constraints.

$$v_k = \begin{cases} v_k \text{ if the location of the } k^{th} \text{ vertex is known} \\ 0 \text{ otherwise} \end{cases} \qquad (5)$$

The rigidity of the mesh is defined as $\| L v_u \quad \|^2$, allowing the mesh surface to be manipulated onto the target shape. The preservation of surface detail is described by a constraint energy as $\| W_c (v_u \quad v_k) \|^2$, providing uniform weighting along the mesh surface, preserving detail and shape. The variable W_c represents the diagonal weight matrix, where each position on its diagonal is 1 if the corresponding vertex is constrained, and 0 otherwise. To solve the location for all vertices based on soft constraints, Equation 4 is minimised with respect to v_u. The Laplacian operator $L = G^T D G$, where G is the discrete gradient operator, and D is the diagonal matrix of triangle areas and (v_k) is the differential coordinates of v_k [27].

4. Results and Evaluation

This section presents results and evaluation for the proposed hybrid skeletal-driven surface registration, presented in section 3 compared with previous model-based and model-free alignment approaches.

Error Metric: Evaluation is performed using one-sided Hausdorff distance defined as $H_B(A) = \sup_{a \in A} d(a; B)$, where $d(a; B)$ is the distance from a point a to a set B, which has shown good measurements between two 3D meshes. The comparison was performed between the final temporally consistent result and the target mesh reconstruction at each time instance. We have evaluated our results against both model-based and model-free approaches.

Model-based: We have measured the results against two publicly available sequences from Vlasic *et al.* [31], Samba and Crane datasets (Figure 3 row 1 and row 2) to the target mesh reconstruction and compared with the proposed results, see Figure 4 and Figure 5. From Figure 7, it is visible that the mean error is smaller then 1 cm, demonstrating the ability to maintain local fine details on loose clothing sequences. Comparing with Vlasic datasets we were able to maintain the same human pose and maintain the clothing dynamics tracked from the proposed pipeline.