A Framework for Natural Animation of Digitized Models

Edilson de Aguiar Rhaleb Zayer
Christian Theobalt
Marcus Magnor Hans-Peter Seidel

MPI–I–2006–4–003 July 2006

Authors' Addresses

Edilson de Aguiar, Rhaleb Zayer, Christian Theobalt, Hans-Peter Seidel
Max-Planck-Institut für Informatik
Stuhlsatzenhausweg 85
66123 Saarbrücken
{edeaguia,zayer,theobalt,hpseidel}@mpi-sb.mpg.de

Marcus A. Magnor
TU Braunschweig
Computer Graphics Lab
Mühlenpfordtstr. 23
38106 Braunschweig, Germany
m.magnor@tu-bs.de

Abstract

We present a novel versatile, fast and simple framework to generate high-quality animations of scanned human characters from input motion data. Our method is purely mesh-based and, in contrast to skeleton-based animation, requires only a minimum of manual interaction. The only manual step that is required to create moving virtual people is the placement of a sparse set of correspondences between triangles of an input mesh and triangles of the mesh to be animated. The proposed algorithm implicitly generates realistic body deformations, and can easily transfer motions between human subjects of completely different shape and proportions.

Our approach handles many different types of input data, e.g. other animated meshes and motion capture files, in just the same way. Finally, and most importantly, it creates animations at interactive frame rates. We feature two working prototype systems that demonstrate that our method can generate lifelike character animations from both marker-based and marker-less optical motion capture data.

Keywords

Animation, Mesh Deformation, Motion transfer, Motion Capturing

Contents

1 Introduction

In recent years, photo-realistic computer-generated animations of humans have become the most important visual effect in motion pictures and computer games. In order to obtain an authentic virtual actor, it is of great importance that she mimics as closely as possible the motion of her real-world counterpart. Even the slightest unnaturalness would be instantaneously unmasked by the unforgiving eye of the viewer and the illusion of seeing a real person would be compromised.

It is thus no wonder that the number of working hours that animators spend in order to live up to these high requirements in visual quality is considerable. To generate virtual people, they make use of a well-established but often inflexible set of tools (see also Sect. 2) that makes a high amount of manual interaction unavoidable. First, the geometry of the human body is hand-crafted in a modeling software or obtained from a laser scan of a real individual. In a second step, a kinematic skeleton model is implanted into the body by means of, at best, a semi-automatic procedure. In order to couple the skeleton with the surface mesh, an appropriate representation of pose-dependent skin deformation has to be found. Finally, a description of body motion in terms of joint parameters of the skeleton is required. It can either be designed in a computer or learned from a real person by means of motion capture. Although the interplay of all these steps delivers animations of stunning naturalness, the whole process is very labor-intensive and does not easily allow for the interchange of animation descriptions between different virtual persons.

In this paper, we present *Confluent Motion*, a novel versatile, fast and much simpler approach to animate virtual characters. Our method makes the following contributions:

Confluent Motion is a purely mesh-based animation paradigm that does not rely on kinematic skeletons. Nevertheless, it seamlessly integrates into an animator's traditional animation workflow. It can be used to realistically animate static meshes of arbitrary humans with minimal effort. To this end, only a sparse set of fixed correspondences between triangles of a moving input mesh and triangles of the output mesh has to be specified. Since only a coarse set of correspondences is required, we can use arbitrarily structured input meshes, i.e. even segmented or disconnected geometry models, that topologically differ from the output mesh. This enables Confluent Motion to process different types of inputs, such as animated meshes created by skilled artists, as well as raw motion data acquired with marker-based or marker-free

Figure 1: Confluent marker-based animation: Subsequent frames showing the female scan authentically performing a soccer kick. Motion data have been acquired by means of a marker-based motion capture system. Note the realistic protrusion of the chest when she blocks the ball, as well as the original head motion.

optical motion capture systems, in just the same way. The presented framework also produces realistic pose-dependent body deformations implicitly by means of a harmonic field interpolation. Furthermore, it solves the motion transfer problem, i.e. it enables the animator to interchange motions between persons of even widely different body proportions with no additional effort. Lastly, and most importantly, the method computes poses of the target mesh at interactive frame rates. This way, the animator is given the possibility to modify animations with instantaneous feedback. Although we regard Confluent Motion primarily as a tool for human animation, it can be applied in the same way to arbitrary moving subjects.

The paper proceeds with a review of closely related work in Sect. 2. An overview of the approach is given in Sect. 3, and the nuts and bolts of our shape deformation method are described in Sect. 4. We demonstrate that we can realistically animate people using raw motion data from both marker-based, Sect. 5, and marker-less, Sect. 6, optical motion capture systems as inputs. We also show that Confluent Motion allows for the creation of 3D videos of humans by animating highly-detailed laser scanned meshes. We show the high visual quality of our animations in Sect. 7 and conclude in Sect. 8.

2 Related Work

Confluent Motion provides a unified solution to many algorithmic subproblems in traditional human character animation by capitalizing on recent advances on mesh deformation techniques presented in the field of geometry processing.

The first step in human character animation is the acquisition of a human body model comprising a surface mesh and an underlying animation skeleton [5]. Surface geometry can either be hand-crafted or scanned from a real person [3]. The skeleton model is either manually designed or inferred from input motion data [15]. It is also feasible to jointly create surface and skeleton models by fitting a template to body scans [25, 4]. Body variations across different human individuals can also be encoded in the shape description [3, 4].

Mesh and skeleton have to be connected such that the surface deforms realistically with the body motion. A popular method serving this purpose is skinning [18]. It represents vertex displacements as weighted set of influences from adjacent joints. Weights can be hand-crafted or automatically inferred from examples [26, 32, 21]. Deformation models can also be created by interpolation between example scans [2]. Sand et al. [24] infer a skinning model by combining marker-based motion capture with a shape-from-silhouette method.

The virtual human is awakened by specifying motion parameters for the joints in the skeleton. Common methods to generate such motion descriptions are key-framing [9], physics-based animation [11] or optimization-based creation of physically plausible movements [12]. The most authentic motion data can be acquired through optical marker-based [6, 15] or marker-free motion capture [20]. Unfortunately, reusing motion capture data for subjects of different body proportions is not trivial, and requires computationally expensive motion editing [17, 13] and motion retargetting techniques [14, 31].

By extending ideas on mesh-based surface deformation we have designed a new versatile and simple framework that overcomes several limitations of the classic animation pipeline. As highly detailed 3D triangle meshes become more and more accessible, there has been an increasing interest in devising techniques which can work directly on the these geometric representations without passing through intermediate pipelines such as the one mentioned above. In the mesh editing context, [1, 27, 33, 34, 16, 19] rely on the notion of differential coordinates to deform a mesh while preserving its geometric

detail. This notion was extended to the volumetric setting in [35]. The main difference between these schemes lies in the way they propagate the deformation across the mesh. On the animation side [29] propose an approach that is similar in spirit but aims at a different goal. Using a full body correspondence between different synthetic models, their method can transfer the motion of one to the other. [30] develops a mesh-based inverse kinematics framework with potential application to mesh animation.

Our system is most closely related to the SCAPE method [4]. The SCAPE model learns pose and shape variation across individuals from a database of body scans and can animate scanned human body geometry from motion capture by solving a nonlinear optimization problem. Confluent Motion addresses the animation problem by solving simple linear systems, hence delivering animations at interactive rates. By relying on the semantic similarities between characters it also solves the retargetting problem on-the-fly.

3 Overview

It is the guiding idea behind the development of Confluent Motion to provide animators with a simple and fast framework to directly apply captured motion to scanned human body models, Fig. 2. Thus, the input to our framework are motion data that have been measured from real individuals using marker-free or marker-less optical motion estimation methods. The first processing step transforms these sequences of key body poses into a sequence of postures of a simple triangle mesh model, henceforth termed *template mesh*. We take advantage of existing marker-free motion estimation methods that directly output such sequences of template mesh poses, Sect. 6. Marker-based motion capture data can be straightforwardly transformed into a moving template mesh representation using standard animation software, Sect. 5. At the heart of our approach is an algorithm to transfer motion from the moving template mesh onto the scanned body mesh, henceforth termed the *target mesh*. We formulate the motion transfer problem as a deformation transfer problem, Sect. 4. To this end, a sparse set of triangle correspondences between the template and the target mesh needs to be specified and an automatic deformation interpolation animates the target mesh. In Sect. 5 we show how Confluent Motion is applied to generate high-quality character animations from marker-based motion capture data. The employment of a marker-less motion estimation method as front-end to our pipeline is detailed in Sect. 6. We summarize our results in Sect. 7 and conclude in Sect. 8.

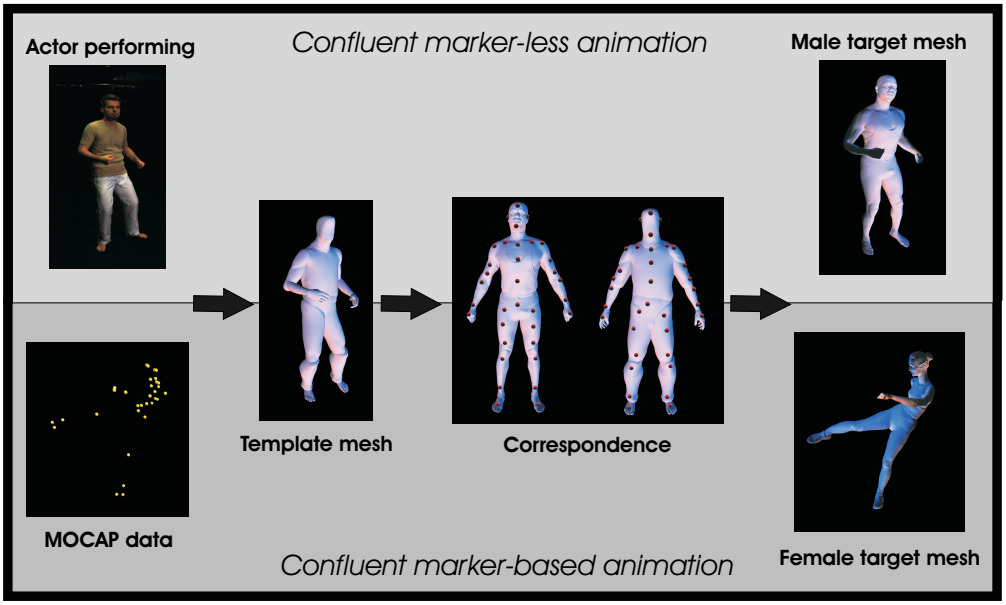


Figure 2: Illustration of the Confluent Motion pipeline.

4 Mesh Deformation

The algorithmic core of Confluent Motion is a mesh deformation method that transfers motion from the template mesh onto the target mesh. We regard motion transfer as a pure deformation interpolation problem, Sect. 4.1. This way, we put aside all difficulties relating to the dissimilarities between the template and the target, e.g. anatomical disparity (body proportions), and take advantage of their semantic similarities, e.g. the fact that both mesh representations have knees and elbows. For this purpose, the user is asked to specify a set of *correspondence triangles* between the two meshes, Sect. 4.2. In practice, this means that the user marks a set of triangles on the template and assigns to each of them a corresponding triangle on the target. We resort to this interactive step since there exists no viable automatic approach that can identify body segments on meshes standing in general poses. The first algorithmic challenge is to make the target mesh deform in the same way as the template by only considering the sparse set of representative triangle correspondences, Sect. 4.1. Furthermore, the generally large size of the high-resolution scans requires the use of fast numerical methods to allow for robust and efficient animation. Lastly, we have to take precautions to not make singularity problems of our deformation interpolation scheme deteriorate our animation results, Sect. 4.3.

4.1 Deformation Interpolation

We formulate the deformation transfer problem as a deformation interpolation problem. The motion of the template mesh from its reference pose (e.g. Fig. 3a) into another pose (e.g. Fig. 3c) can be captured by the deformation applied to a set of marked triangles. A correct interpolation of this deformation applied over the corresponding triangles of the target mesh would bring it from its own reference pose (e.g. Fig. 3b) into the template's pose (e.g. Fig. 3d,e). To this end, both reference poses are roughly aligned a priori. In the case of human animation, this deformation can be characterized as a simple rotation for each triangle and a translational degree of freedom.

If the per-triangle rotations are specified as simple 3×3 cosine matrices, the interpolation has to cope with $9 \times m$ interpolation points, where m is the number of triangle correspondences. This gives rise to an intricate problem, especially when the correlation between the components of each rotation matrix is taken into account. Nevertheless, there have been some successful approaches to propagate the effect of changing local frames on the whole mesh [19, 27]. The quaternion representation, on the other hand, reduces the size of the range to $4 \times m$ interpolation points. Most of the existing methods for interpolating quaternion rotations rely on an intermediate function which guides the interpolation, as it has been shown in the case of the spherical or spline based interpolation. Such a function can be defined as distance function or heat kernels as in [33], or as a harmonic scalar field as proposed in [34]. Unfortunately, these methods do not apply to the current setting as they can only interpolate from a single set of starting points to a single set of ending points. Another breed of methods which allow for multiple point interpolation can be found in [7, 22]. While these techniques work well for locomotion, it is not clear how to extend them to our current general setup. In fact, it is difficult to find a function which can interpolate all the quaternion values and it is not clear how to assign specific rotations to all triangles in a mesh. Following an idea proposed in [34], we regard each component of a quaternion

$$Q = [w \; q_1 \; q_2 \; q_3] \tag{1}$$

as a scalar field defined over the entire mesh. Hence, given the values of these components at the marked triangles, we interpolate each scalar field independently. In order to guarantee a smooth interpolation we regard these scalar fields as harmonic fields defined over the mesh. The interpolation can then be performed efficiently by solving the Laplace equation

$$\nabla^2 Q \; = \; 0 \tag{2}$$

over the whole mesh with constraints at the correspondence triangles. Once the rotational components are computed, we average the quaternion rotations of the vertices to obtain a quaternion rotation for each triangle. This way we establish a geometric transformation for each triangle of the target mesh M. However, this last step destroys its original connectivity and yields a new fragmented mesh M'. In order to recover the original geometry of the mesh while satisfying the new rotations, we have to solve the problem in a least square sense. In the following we sketch a simple way to setup this optimization problem that eases the implementation effort. The problem can be rephrased as finding a new tight mesh having the same topology as the original target mesh, such that its differential coordinates encode the same geometric detail as the ones of the fragmented mesh M'. This can be achieved by satisfying the following equation in terms of the coordinates x of M and u of M':

$$\nabla_M^2 x \;=\; \nabla_{M'}^2 u \;. \tag{3}$$

In order to carry out this discretization correctly the topological difference between both meshes should be addressed. Technically, the differential coordinates of the fragmented mesh are computed by deriving the Laplacian operator for the fragmented mesh and then applying it to its coordinates. This, in fact, yields a vector of size $3 \times nT$, where nT is the number of triangles. We sum the components of this vector according to the connectivity of the original mesh M. This yields a new vector $U_{reduced}$ of size nV, where nV is the number of vertices in M, and the discrete form of equation (3) reads as simple as

$$LX = U_{reduced} \;. \tag{4}$$

In this linear system the matrix L is the discrete Laplace operator. In order to capture the local geometry of the mesh we use the geometric Laplacian, e.g. [10, 23], which is more sensitive to the irregularities of the triangulation in comparison to the uniform or graph Laplacian. During the processing of an animation sequence, the differential operator matrix does not change. Furthermore, since it is symmetric positive definite we can perform a sparse Cholesky decomposition as preprocessing step and perform only back substitution for each frame.

This enables us to compute novel poses of the target mesh at an interactive rate of 2 fps for meshes of the order of 30 to 50 thousand triangles. For highly detailed meshes, e.g $264K\Delta$ for the female Cyberware$^{\mathrm{TM}}$ model (Fig. 1), the algorithm performs at 10 s per frame.

(a) (b) (c) (d) (e)

Figure 3: A template model (a) and a high-resolution body scan (b) in their respective reference poses. - Influence of the number of markers on the quality of the deformation: the template in a pose obtained via motion capture (c). While 22 triangle correspondences already suffice to transfer this pose in good quality to the scan (d), 180 triangle correspondences reproduce even subtle details (e).

4.2 Correspondence Placement

As our method does not require any skeleton retargetting or full mesh correspondence, it is imperative that the choice of the sampling captures as much as possible of the geometric deformation. Fig. 3d shows the target mesh mimicking the pose of the template when only as few as 22 correspondences are used. Even though the pose is not reproduced in full detail, the overall pose is captured. This is why the knee is not bent in the same way as in the template. In Fig. 3e though, with 180 correspondences, the pose is faithfully reproduced, including the knee (taking into account differences in skeleton dimensions).

For cylindrically shaped body parts we require the user to specify a single correspondence triangle and we mark additional triangles automatically by taking additional directions in the cross sectional plane and intersecting them with the mesh. For geometrically more complex body parts, such as the lap or the shoulders, correspondences are fully specified by the user. Note that except from the positioning of the markers, the whole Confluent Motion approach is fully automatic. The number of triangle correspondences used in our animations ranges from 140 to 220, half of which is automatically generated.

Furthermore, the placement of the markers directly affects the pose-dependent

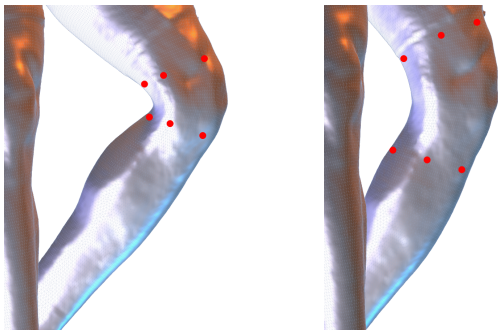


Figure 4: Influence of the markers' placement on the deformation quality: marking correspondence triangles (red dots) close to an (anatomical) joint creates a sharp bend in the skin (left) while increasing the distance to the joint enables smoother bending (right).

surface deformation of the target mesh. So the user does not have to tweak any weights as in the commonly used skinning methods. The principle here is simple: for having a sharp bend in the surface the correspondences should be placed close to either side of the joint, Fig. 4 (left). Increasing the distance of the markers from the joint allows for a softer bending, Fig. 4 (right).

4.3 Coping with Singularities

A limitation which is inherent to most systems based on rotation interpolation is the "candy-wrapper" collapse effect [21]. Our current approach is not immune to this problem and may suffer from its effects as well. In the following, we devise two simple techniques to tackle this problem.

In order to prevent the twisting collapse we need a simple way to predict when it happens. It occurs when some of the correspondence triangles undergo a rotation of 180 degrees. Conveniently, this can be easily detected by inspecting the first component of the quaternion representation. A null or a very small value indicates that such a situation is likely to occur. The first solution to this problem is a simple workaround which basically consists of changing the reference pose of the template and the scan model and using an intermediate pose as a reference. To this end, a previous frame for the template and the target can be used as the reference pose. Another alternative would be to create an intermediate pose for the template mesh and deform the target into a similar pose. This intermediate pose can be used as the new reference configuration.

The second solution would be to construct a shortest path between two points along which this problem occurs and interpolate the deformation along this line using the spherical quaternion interpolation commonly known as *SLERP*. This way, deformation can be interpolated to the triangles adjacent to the path. These new triangles will be used as constraints for the harmonic interpolation, Eq. (2).

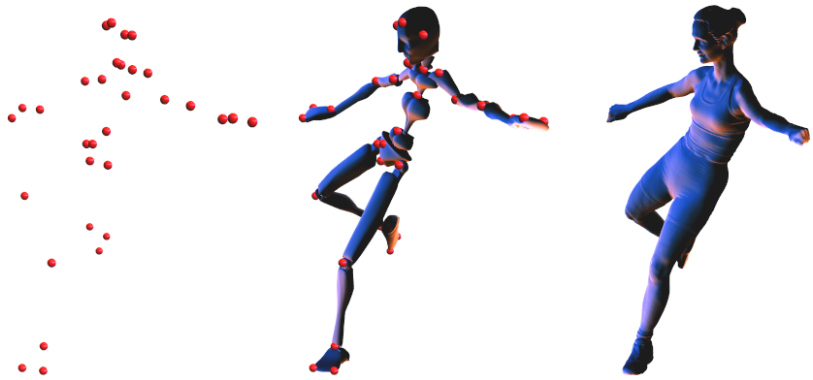
5 Confluent Marker-based Animation

By far the most authentic animation descriptions for moving virtual actors are obtained by measuring motion parameters of real people. The most established technique to achieve this purpose is marker-based optical motion capture [6]. Here, the body of a moving individual is equipped with optical markings at kinematically relevant body locations, e.g. around the joints. The person now performs in front of multiple cameras that reconstruct and track the 3D positions of the markings. From the 3D marker trajectories, a kinematic skeleton model of the person is generated. The motion is conveniently parameterized as rotational and translational parameters of the joints.

Marker-based tracking systems provide the animator with motion descriptions of unequaled accuracy and naturalness. Even subtle details in motion patterns are faithfully captured. Furthermore, the motion data comes in the correct skeleton-based parameterization straight away, enabling the animator to straightforwardly map them onto a virtual character.

The high quality of captured motion data, however, comes at the expense of many inflexibilities in their application. Firstly, motion parameters cannot easily be reused with virtual persons that differ in skeletal proportions from the captured individual. To make this possible, computationally expensive motion retargetting algorithms have to be applied [14]. Secondly, motion capture systems only deliver a description of human motion in terms of interconnected rigid bodies. The non-rigid deformations of the skin and the soft-tissue surrounding the bones have to be manually modeled, e.g. by means of vertex skinning [18].

We now demonstrate that the Confluent Motion paradigm can be straightforwardly applied to animate human body scans with motion capture data. Paradoxically, despite discarding the use of a kinematic animation skeleton it allows us to generate high-quality animations and to generate convincing



(a) (b) (c)

Figure 5: The set of input markers (a) are used to generate an intermediate biped model (b). By applying our deformation technique the acquired motion is realistically transfered to the final human body scan (c).

surface deformations with just the same simple processing steps. By the same token, the motion retargetting problem is implicitly solved.

The steps that have to be taken to animate a mesh confluently with the input data (Fig. 5a) are very simple and can be summarized in three sentences: First, using any standard animation software like 3D Studio Max^{TM}, the motion capture skeleton is transformed into a surface model in which the bones of the biped are represented as triangle meshes (Fig. 5b). Consequently, in a manual step static per-triangle correspondences between the triangulated biped and the scanned mesh are defined. Finally, our mesh deformation approach realistically moves and deforms the scanned mesh to accurately mimic the motion of the input model, and brings it to a correct global position (Fig. 5c).

We have applied our method to confluently animate a male and a female high-resolution mesh that have been generated with a Cyberware full-body scanner and that the company kindly provides for public use. Input motion capture data are taken from a database of motion files provided by Eyes, Japan Co. Ltd. Fig. 1 shows several frames of an animation in which we made the female model perform a soccer kick. The input is a motion capture file comprising of 90 key body poses. The actress realistically blocks the ball, kicks it and scores. Note that the animation nicely displays even subtle details like the protrusion of the chest during blocking. The skin deformation around the knees and the elbows is also authentically reproduced. Fig. 6

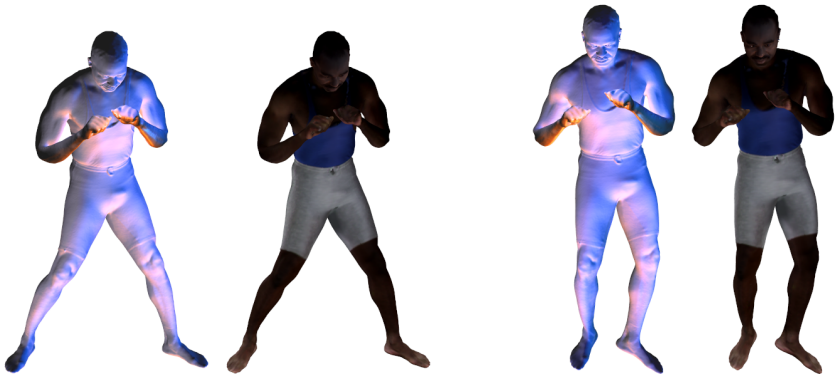


Figure 6: Male model boxing, rendered with and without static surface textures. Note the realistic skin deformation of the animated scanned mesh.

shows the male model performing boxing punches. Note that despite the fact that the input motions stem from persons with totally different anatomical dimensions, very natural animations free of retargetting artefacts (such as sliding feet) are generated. Our experiments confirm that Confluent Motion is a highly flexible and simple approach to create high-quality animations from arbitrary marker-based motion capture data.

6 Confluent Marker-less Animation

In the previous section we have demonstrated that our approach transforms motion data captured with state-of-the art intrusive measurement technology into high-quality animations. One major disadvantage of this type of input data is the fact that the acquisition situation has to be completely optimized for the sole purpose of motion estimation. In consequence, the person can hardly perform in a general environment and it is also impossible to make the video footage available to further processing, e.g. for the purpose of texture reconstruction. In contrast to marker-based systems, marker-less tracking methods allow for this [20]. Instead of inferring movement descriptions from artificially placed markings, they estimate motion parameters from image features in the raw video footage showing a moving person in an unmodified scene.

Using a marker-less human motion capture system as front-end to our Confluent Motion pipeline enables us create two intriguing applications, video-driven animation and confluent 3D video.

6.1 Video-driven Animation

For non-intrusively estimating animation parameters, we make use of the passive optical motion capture approach proposed in [8]. To this end, we record a moving person with eight static video cameras that are roughly placed in a circle around the center of the scene. From the frame-synchronized video streams, the shape and the motion parameters of the human are estimated. To achieve this purpose, a template model, Fig. 3a, comprising of a kinematic skeleton and sixteen separate closed surface segments is fitted to each time step of video by means of silhouette-matching. The output of the method conveniently represents the captured motion as a sequence in which the template model subsequently strikes the estimated body poses.

This output format can be directly used as input to our Confluent Motion pipeline. The animator specifies triangle correspondences between the template and the scanned mesh that shall be animated. Finally, our algorithm makes the output mesh mimic the motion that we have captured in video. Realistic surface deformations of the output mesh are implicitly generated. Please note that the fact that the moving input template comprises of individual triangle mesh segments does by no means limit the applicability of our method.

In order to demonstrate the performance of video-driven animation, we animate our female (264K triangle) and male (294K triangle) Cyberware scans with two very different captured motion sequences. The first sequence contains 156 frames and shows a female subject performing a capoeira move. The second sequence is 330 frames long and shows a dancing male subject. Fig. 7 shows a comparison between actual input video frames and two scans striking similar poses. It illustrates that body poses recorded on video can be faithfully transfered to 3D models of arbitrary human subjects. Differences in body shape and skeletal proportions can be completely neglected. Our results illustrate that the Confluent Motion pipeline provides animators with a tool to conveniently transfer motion extracted from normal video streams onto high-quality 3D models.

6.2 Confluent 3D Video

By means of video-driven animation, Sect. 6.1, we can also generate 3D videos of moving characters. In the traditional model-based approach to 3D video a simplified body model is used to carry out a passive optical motion estimation from multiple video streams [28, 8]. During rendering,

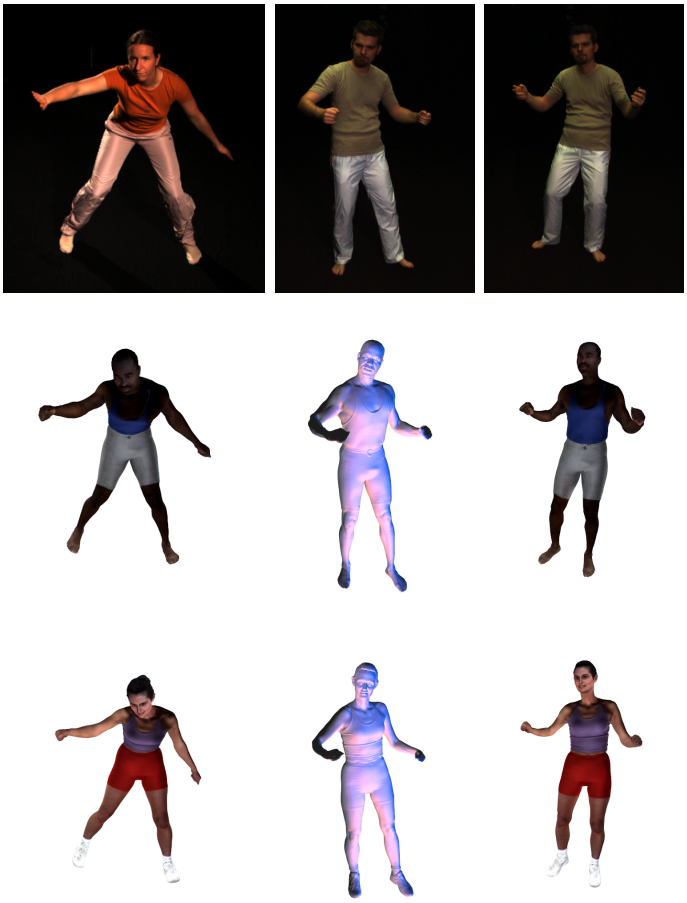


Figure 7: Video-driven animation: Motion parameters are extracted from raw video footage of human performances (top row). By this means, body poses of a video-taped individual can easily be mapped to body scans of other human subjects (second and third row). Note that skin deformations are naturally modeled (middle column). Scans are faithfully animated regardless of the differences in body shape and skeletal dimensions.

the same simplified shape template is displayed in the sequence of captured body poses and textured from the input videos. Although these methods deliver realistic free-viewpoint renditions of virtual actors, we expect that a more accurate underlying geometry increases realism even further. Confluent Motion enables us to decouple the model acquisition stage from the motion estimation and the rendering stages. This way, we can use an animated high-quality scan as the underlying geometry description.

To demonstrate the feasibility of this approach in practice, we have acquired

Figure 8: Confluent Motion enables creation of 3D videos with high-quality geometry models. Due to the accurate geometry the rendered appearance of the actor (left subimages) nicely corresponds to his true appearance in the real world (right subimages).

full-body surface scans of several individuals in our studio. To this end, we merged several partial body scans performed with our MINOLTA VI-910 which is best suited for scanning small objects. Thus the quality of our scans is far below the quality of scans acquired using full-body scanners. For each scanned individual, we also recorded several motion sequences with multiple synchronized video cameras. We use the method from Sect. 6.1 to animate the scans from the captured motion data. During 3D video display, the animated scan is projectively textured with the captured video frames.

Fig. 8 shows two free-viewpoint renditions of a dynamically textured animated scan in comparison to input images of the test subject. The free-viewpoint renditions reflect the true appearance of the actor. Since we are given a better surface geometry, texture blending artefacts are hardly observed. Furthermore, we can even reproduce the true shape of the sweater which would not have been possible with a moving template. Remaining artefacts in the rendering can be clearly attributed to the non-optimal scanning apparatus we used. We do not see this as a principal limitation of our approach, as the results in Sect. 5 and Sect. 6.1 with Cyberware scans demonstrate.

7 Results and Discussion

To demonstrate the potential of Confluent Motion, we conducted several experiments with both marker-based and marker-less motion acquisition techniques.

Due to their high resolution, we used the Cyberware models provided with their original surface colors in most of our experiments. In Fig. 1 and Fig. 6 the models faithfully reproduce the acquired performances of professional athletes. Marker-less motion acquisition enables us to perform video-driven animation. Both of our models in Fig. 7 authentically mimic the human performances captured on video. This also allows for producing 3D video, Fig. 8.

Our method satisfies the deformation constraints in a least-square sense, while maintaining the smoothness and details of the original mesh which are encoded in the differential operators. Thus, in order to enforce any constraint, it is imperative to increase the number of correspondences associated with it. We can ensure stable feet placement simply by marking a sufficient number of constraints on the feet. Due to different body proportions between source and target mesh, achieving an exact match between source and target is impossible. However this is a general problem even for skeleton-based retargetting methods.

The results confirm that our method is capable of delivering visually convincing virtual character animation at a low interaction cost, but still at interactive rates. The method is able to process large data sets in the order of 200 to 300 KΔ in just seconds. For smaller sets of 30 to 50 KΔ the results are generated at 2 frames per second. All the experiments were conducted on a single 3.2GHz notebook.

We would also like to emphasize that our method can use arbitrary template meshes, which can be arbitrarily structured. For the vision-based results the fact that the template reflects the shape of the captured persons is a requirement of the motion capture method [8] and not of our approach.

As for any novel technique our method still has some limitations. For extreme deformation we note that there is generally some loss in volume due to the nature of our interpolation. We believe that using higher order differential operators or the volumetric approach proposed in [35] would reduce such artefacts although this might decrease the current numerical performance. We plan to investigate these ideas as a direction of future work.

We nonetheless devised a powerful framework for animating virtual human characters. Since our method relies only on setting up and solving linear systems, the implementation and the reproduction of our results are straightforward.

8 Conclusion

Confluent Motion is a new animation framework aiming at simplifying the traditional, not so straightforward acquisition-to-animation pipeline. The only manual interaction required is the selection of a small number of triangles to enforce the semantic correspondence between different models and guide the animation process. The proposed method is thus easy and intuitive to use and does not require any training. By means of the same efficient methodology Confluent Motion simultaneously solves the animation, the surface deformation and the motion retargetting problem.

As a direction for future work, we would like to combine our technique with an approach to learn per-time-step surface deformations from input video footage. We also expect that outfitting our virtual characters with simulated apparel would further increase the visual realism.

Bibliography

[1] M. Alexa. Differential coordinates for local mesh morphing and deformation. *The Visual Computer*, 19(2-3):105–114, 2003.

[2] B. Allen, B. Curless, and Z. Popovic. Articulated body deformation from range scan data. *ACM Trans. Graph.*, 21(3):612–619, 2002.

[3] B. Allen, B. Curless, and Z. Popovic. The space of human body shapes: reconstruction and parameterization from range scans. *ACM Trans. Graph.*, 22(3):587–594, 2003.

[4] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. *ACM Trans. Graph.*, 24(3):408–416, 2005.

[5] N. Badler, D. Metaxas, and N. Magnenat Thalmann. *Virtual Humans*. Morgan Kaufmann, 1999.

[6] B. Bodenheimer, C. Rose, S. Rosenthal, and J. Pella. The process of motion capture: Dealing with the data. In *Computer Animation and Simulation '97*, pages 3–18, Sept. 1997.

[7] S. R. Buss and J. P. Fillmore. Spherical averages and applications to spherical splines and interpolation. *ACM Trans. Graph.*, 20(2):95–126, 2001.

[8] J. Carranza, C. Theobalt, M. Magnor, and H.-P. Seidel. Free-viewpoint video of human actors. *ACM Transactions on Graphics (Proc. of SIGGRAPH'03)*, 22(3):569–577, 2003.

[9] J. Davis, M. Agrawala, E. Chuang, Z. Popovic, and D. Salesin. A sketching interface for articulated figure animation. In *Proc. of SCA '03*, pages 320–328. Eurographics Association, 2003.

[10] M. Desbrun, M. Meyer, and P. Alliez. Intrinsic parameterizations of surface meshes. *Computer Graphics Forum (Proc. Eurographics)*, 21(3):209–218, 2002.

[11] P. Faloutsos, M. van de Panne, and D. Terzopoulos. The virtual stuntman: dynamic characters with a repertoire of autonomous motor skills. *Computers and Graphics*, 25(6):933–953, 2001.

[12] A. C. Fang and N. S. Pollard. Efficient synthesis of physically valid human motion. *ACM Trans. Graph.*, 22(3):417–426, 2003.

[13] M. Gleicher. Motion editing with space-time constraints. In *Proc. of 1997 Symposium on Interactive 3D Graphics*, page 139ff, 1997.

[14] M. Gleicher. Retargetting motion to new characters'. In *Proc. of ACM SIGGRAPH*, pages 33–42, 1998.

[15] L. Herda, P. Fua, R. Plänkers, R. Boulic, and D. Thalmann. Skeleton-based motion capture for robust reconstruction of human motion. In *CA '00: Proc. of the Computer Animation*, page 77ff. IEEE Computer Society, 2000.

[16] T. Igarashi, T. Moscovich, and J. F. Hughes. As-rigid-as-possible shape manipulation. *ACM Trans. Graph.*, 24(3):1134–1141, 2005.

[17] J. Lee and S. Y. Shin. A hierarchical approach to interactive motion editing for human-like figures. In *Proc. of ACM SIGGRAPH*, pages 39–48, 1999.

[18] J. P. Lewis, M. Cordner, and N. Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Proc. of ACM SIGGRAPH'00*, pages 165–172, New York, NY, USA, 2000.

[19] Y. Lipman, O. Sorkine, D. Levin, and D. Cohen-Or. Linear rotation-invariant coordinates for meshes. *ACM Trans. Graph.*, 24(3):479–487, 2005.

[20] T. B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *CVIU*, 81(3):231–268, 2001.

[21] A. Mohr and M. Gleicher. Building efficient, accurate character skins from examples. *ACM Trans. Graph.*, 22(3):562–568, 2003.

[22] S. I. Park, H. J. Shin, and S. Y. Shin. On-line locomotion generation based on motion blending. In *Proc. of SCA '02*, pages 105–111. ACM Press, 2002.

[23] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experiment. Math.*, 2(1):15–36, 1993.

[24] P. Sand, L. McMillan, and J. Popovic. Continuous capture of skin deformation. *ACM Trans. Graph.*, 22(3):578–586, 2003.

[25] H. Seo and N. Magnenat-Thalmann. An automatic modeling of human bodies from sizing parameters. In *Proc. of SI3D '03*, pages 19–26. ACM Press, 2003.

[26] P.-P. J. Sloan, I. Charles F. Rose, and M. F. Cohen. Shape by example. In *SI3D '01: Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 135–143. ACM Press, 2001.

[27] O. Sorkine, Y. Lipman, D. Cohen-Or, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian surface editing. In *Proc. of SGP'04*, pages 179–188, 2004.

[28] J. Starck and A. Hilton. Towards a 3D virtual studio for human appearance capture. In *Proc. of Vision, Video and Graphics*, pages 17–24, 2003.

[29] R. W. Sumner and J. Popovic. Deformation transfer for triangle meshes. *ACM Trans. Graph.*, 23(3):399–405, 2004.

[30] R. W. Sumner, M. Zwicker, C. Gotsman, and J. Popovic. Mesh-based inverse kinematics. *ACM Trans. Graph.*, 24(3):488–495, 2005.

[31] S. Tak and H.-S. Ko. A physically-based motion retargeting filter. *ACM Trans. Graph.*, 24(1):98–117, 2005.

[32] X. C. Wang and C. Phillips. Multi-weight enveloping: least-squares approximation techniques for skin animation. In *Proc. of SCA'02*, pages 129–138. ACM Press, 2002.

[33] Y. Yu, K. Zhou, D. Xu, X. Shi, H. Bao, B. Guo, and H.-Y. Shum. Mesh editing with Poisson-based gradient field manipulation. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 23(3):644–651, 2004.

[34] R. Zayer, C. Rössl, Z. Karni, and H.-P. Seidel. Harmonic guidance for surface deformation. In *Proc. of Eurographics 2005*, volume 24, pages 601–609, 2005.

[35] K. Zhou, J. Huang, J. Snyder, X. Liu, H. Bao, B. Guo, and H.-Y. Shum. Large mesh deformation using the volumetric graph laplacian. *ACM Trans. Graph.*, 24(3):496–503, 2005.

Below you find a list of the most recent technical reports of the Max-Planck-Institut für Informatik. They are available by anonymous ftp from `ftp.mpi-sb.mpg.de` under the directory `pub/papers/reports`. Most of the reports are also accessible via WWW using the URL `http://www.mpi-sb.mpg.de`. If you have any questions concerning ftp or WWW access, please contact `reports@mpi-sb.mpg.de`. Paper copies (which are not necessarily free of charge) can be ordered either by regular mail or by e-mail at the address below.

Max-Planck-Institut für Informatik
Library
attn. Anja Becker
Stuhlsatzenhausweg 85
66123 Saarbrücken
GERMANY
e-mail: `library@mpi-sb.mpg.de`

MPI-I-2006-5-004	F. Suchanek, G. Ifrim, G. Weikum	Combining Linguistic and Statistical Analysis to Extract Relations from Web Documents
MPI-I-2006-5-003	V. Scholz, M. Magnor	Garment Texture Editing in Monocular Video Sequences based on Color-Coded Printing Patterns
MPI-I-2006-5-002	H. Bast, D. Majumdar, R. Schenkel, M. Theobald, G. Weikum	IO-Top-k: Index-access Optimized Top-k Query Processing
MPI-I-2006-5-001	M. Bender, S. Michel, G. Weikum, P. Triantafilou	Overlap-Aware Global df Estimation in Distributed Information Retrieval Systems
MPI-I-2006-4-007	O. Schall, A. Belyaev, H. Seidel	Feature-preserving Non-local Denoising of Static and Time-varying Range Data
MPI-I-2006-4-005	S. Yoshizawa	?
MPI-I-2006-4-004	V. Havran, R. Herzog, H. Seidel	On Fast Construction of Spatial Hierarchies for Ray Tracing
MPI-I-2006-4-003	E. de Aguiar, R. Zayer, C. Theobalt, M. Magnor, H. Seidel	A Framework for Natural Animation of Digitized Models
MPI-I-2006-4-002	G. Ziegler, A. Tevs, C. Theobalt, H. Seidel	GPU Point List Generation through Histogram Pyramids
MPI-I-2006-4-001	R. Mantiuk	?
MPI-I-2006-2-001	T. Wies, V. Kuncak, K. Zee, A. Podelski, M. Rinard	On Verifying Complex Properties using Symbolic Shape Analysis
MPI-I-2006-1-007	I. Weber	?
MPI-I-2006-1-006	M. Kerber	Division-Free Computation of Subresultants Using Bezout Matrices
MPI-I-2006-1-005	I. Albrecht	?
MPI-I-2006-1-004	E. de Aguiar	?
MPI-I-2006-1-001	M. Dimitrios	?
MPI-I-2005-5-002	S. Siersdorfer, G. Weikum	Automated Retraining Methods for Document Classification and their Parameter Tuning
MPI-I-2005-4-006	C. Fuchs, M. Goesele, T. Chen, H. Seidel	An Emperical Model for Heterogeneous Translucent Objects

MPI-I-2005-4-005	G. Krawczyk, M. Goesele, H. Seidel	Photometric Calibration of High Dynamic Range Cameras
MPI-I-2005-4-004	C. Theobalt, N. Ahmed, E. De Aguiar, G. Ziegler, H. Lensch, M.A.,. Magnor, H. Seidel	Joint Motion and Reflectance Capture for Creating Relightable 3D Videos
MPI-I-2005-4-003	T. Langer, A.G. Belyaev, H. Seidel	Analysis and Design of Discrete Normals and Curvatures
MPI-I-2005-4-002	O. Schall, A. Belyaev, H. Seidel	Sparse Meshing of Uncertain and Noisy Surface Scattered Data
MPI-I-2005-4-001	M. Fuchs, V. Blanz, H. Lensch, H. Seidel	Reflectance from Images: A Model-Based Approach for Human Faces
MPI-I-2005-2-004	Y. Kazakov	A Framework of Refutational Theorem Proving for Saturation-Based Decision Procedures
MPI-I-2005-2-003	H.d. Nivelle	Using Resolution as a Decision Procedure
MPI-I-2005-2-002	P. Maier, W. Charatonik, L. Georgieva	Bounded Model Checking of Pointer Programs
MPI-I-2005-2-001	J. Hoffmann, C. Gomes, B. Selman	Bottleneck Behavior in CNF Formulas
MPI-I-2005-1-008	C. Gotsman, K. Kaligosi, K. Mehlhorn, D. Michail, E. Pyrga	Cycle Bases of Graphs and Sampled Manifolds
MPI-I-2005-1-008	D. Michail	?
MPI-I-2005-1-007	I. Katriel, M. Kutz	A Faster Algorithm for Computing a Longest Common Increasing Subsequence
MPI-I-2005-1-003	S. Baswana, K. Telikepalli	Improved Algorithms for All-Pairs Approximate Shortest Paths in Weighted Graphs
MPI-I-2005-1-002	I. Katriel, M. Kutz, M. Skutella	Reachability Substitutes for Planar Digraphs
MPI-I-2005-1-001	D. Michail	Rank-Maximal through Maximum Weight Matchings
MPI-I-2004-NWG3-001	M. Magnor	Axisymmetric Reconstruction and 3D Visualization of Bipolar Planetary Nebulae
MPI-I-2004-NWG1-001	B. Blanchet	Automatic Proof of Strong Secrecy for Security Protocols
MPI-I-2004-5-001	S. Siersdorfer, S. Sizov, G. Weikum	Goal-oriented Methods and Meta Methods for Document Classification and their Parameter Tuning
MPI-I-2004-4-006	K. Dmitriev, V. Havran, H. Seidel	Faster Ray Tracing with SIMD Shaft Culling
MPI-I-2004-4-005	I.P. Ivrissimtzis, W.-. Jeong, S. Lee, Y.a. Lee, H.-. Seidel	Neural Meshes: Surface Reconstruction with a Learning Algorithm
MPI-I-2004-4-004	R. Zayer, C. Rssl, H. Seidel	r-Adaptive Parameterization of Surfaces
MPI-I-2004-4-003	Y. Ohtake, A. Belyaev, H. Seidel	3D Scattered Data Interpolation and Approximation with Multilevel Compactly Supported RBFs
MPI-I-2004-4-002	Y. Ohtake, A. Belyaev, H. Seidel	Quadric-Based Mesh Reconstruction from Scattered Data
MPI-I-2004-4-001	J. Haber, C. Schmitt, M. Koster, H. Seidel	Modeling Hair using a Wisp Hair Model
MPI-I-2004-2-007	S. Wagner	Summaries for While Programs with Recursion

MPI-I-2004-2-002	P. Maier	Intuitionistic LTL and a New Characterization of Safety and Liveness
MPI-I-2004-2-001	H. de Nivelle, Y. Kazakov	Resolution Decision Procedures for the Guarded Fragment with Transitive Guards
MPI-I-2004-1-006	L.S. Chandran, N. Sivadasan	On the Hadwiger's Conjecture for Graph Products
MPI-I-2004-1-005	S. Schmitt, L. Fousse	A comparison of polynomial evaluation schemes
MPI-I-2004-1-004	N. Sivadasan, P. Sanders, M. Skutella	Online Scheduling with Bounded Migration
MPI-I-2004-1-003	I. Katriel	On Algorithms for Online Topological Ordering and Sorting
MPI-I-2004-1-002	P. Sanders, S. Pettie	A Simpler Linear Time 2/3 - epsilon Approximation for Maximum Weight Matching
MPI-I-2004-1-001	N. Beldiceanu, I. Katriel, S. Thiel	Filtering algorithms for the Same and UsedBy constraints
MPI-I-2003-NWG2-002	F. Eisenbrand	Fast integer programming in fixed dimension
MPI-I-2003-NWG2-001	L.S. Chandran, C.R. Subramanian	Girth and Treewidth
MPI-I-2003-4-009	N. Zakaria	FaceSketch: An Interface for Sketching and Coloring Cartoon Faces
MPI-I-2003-4-008	C. Roessl, I. Ivrissimtzis, H. Seidel	Tree-based triangle mesh connectivity encoding