Video-Based Character Animation

J. Starck, G. Miller and A. Hilton

Centre for Vision, Speech and Signal Processing
University of Surrey, Guildford, GU2 7XH, UK

Abstract

In this paper we introduce a video-based representation for free viewpoint visualization and motion control of 3D character models created from multiple view video sequences of real people. Previous approaches to video-based rendering provide no control of scene dynamics to manipulate, retarget, and create new 3D content from captured scenes. Here we contribute a new approach, combining image based reconstruction and video-based animation to allow controlled animation of people from captured multiple view video sequences. We represent a character as a motion graph of free viewpoint video motions for animation control. We introduce the use of geometry videos to represent reconstructed scenes of people for free viewpoint video rendering. We describe a novel spherical matching algorithm to derive global surface to surface correspondence in spherical geometry images for motion blending and the construction of seamless transitions between motion sequences. Finally, we demonstrate interactive video-based character animation with real-time rendering and free viewpoint visualization. This approach synthesizes highly realistic character animations with dynamic surface shape and appearance captured from multiple view video of people.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Animation

1. Introduction

In this paper we introduce a video-based representation for free viewpoint visualization and motion control of 3D character models created from multiple view video sequences of real people. Our work combines image based reconstruction [VBK02, MBR*00] and video-based animation techniques [SE02] to provide a new video-based approach to character animation. In the same way that motion capture technology captures complex human motions for convincing skeletal animation, video based character animation provides the appearance of complex surface dynamics such as cloth motion and wrinkles captured from multiple view video for synthesis of highly realistic human appearance in animation.

Image based reconstruction and rendering in computer vision and computer graphics concentrates on static scenes [LH96, GGSC96] and more recently dynamic scenes [ZKU*04] for free-viewpoint visualization. Free-viewpoint video rendering enables a change of view in a scene and special effects such as freeze frame or slow-motion fly around. However, it is limited to replaying the original captured motion. Providing control of scene dynamics opens up the potential to manipulate, retarget, and create new 3D content, ultimately at the quality of conventional video. Previous approaches to the motion control of people have used model fitting [CTMS03, SH03], where constraints on model shape do not accurately reproduce the geometry of loose clothing and hair, reducing the visual quality compared to model-free video-based rendering [ZKU*04, SH05]. Video-based animation techniques [SSSE00, SE02] have been proposed without the requirement for a prior geometric model, but have been restricted to a fixed viewpoint and non-rigid ob-

jects without self-occlusion. Animation control in video rendering of complex articulated objects such as people remains an open and challenging problem.

Here we present a new technique for controlled animation of people using free viewpoint video. We represent a 3D character as a motion graph [KGP02] of seamless character motions constructed from free-viewpoint video sequences of a real person. Our approach makes the following contributions:

- Reconstruction and representation of real people as sequences of 2D spherical *geometry images* [PH03], a *geometry video* [BSM*03] providing a temporally consistent structure for level of detail and frame rate control in free viewpoint video.
- A new multiple resolution coarse-to-fine spherical matching technique to derive dense surface-to-surface correspondence for geometry image sequences, enabling motion blending between geometry videos.
- Animation control using a motion graph of geometry video clips with seamless motion cycles and transitional motions constructed using spherical matching.

In Section 2 we review related work on image based rendering and video-based animation techniques. In Section 3, we outline the process of constructing a geometry video for a multiple view sequence of a person. In Section 4, we introduce a spherical matching algorithm for motion blending. In Section 5 we outline the construction of a motion graph for character animation and our video renderer for real-time rendering and free viewpoint visualization. Finally in Section 6 we present results for video-based character animation.

2. Background

2.1. Video-based rendering

Rendering novel views of dynamic scenes such as people from multiple view video capture has received considerable interest over the past decade. Kanade et al.'s [KRN97]*Virtualized Reality^{TM}* system used a 5m hemispherical dome of 51 cameras to capture sequences of a dynamic scene. Multiple view silhouettes [MTG97, MBR*00] and photo-consistency between views [VBK02] have been used to render novel views from smaller numbers of cameras. Temporal correspondence has also been introduced in recent work to improve surface reconstruction [VBK02, GM04]. Two problems arise with these approaches: quality of novel view rendering due to errors in geometric surface reconstruction; and reconstruction of an unstructured model at each time instant which does not allow the scene dynamics to be modified for reanimation.

High quality video-based rendering has been addressed using stereo reconstruction techniques for a 1D array of cameras [ZKU*04, SH05]. Zitnick et al. [ZKU*04] introduce a novel stereo approach that simultaneously mattes a general scene into two layers and reconstructs depth. Results demonstrate rendered views comparable to the captured video, but have limited viewpoint control and no control over scene dynamics. To allow reanimation of dynamic scenes of people, model-based reconstruction techniques which fit a generic humanoid model to observations from multiple views have been developed [CTMS03, SH03, PF01] . Constrained multiple view optimization of model pose and shape is performed to minimize the model re-projection error for silhouette [CTMS03] and stereo correspondence [SH03,PF01]. Model-based approaches reconstruct a structured representation with an articulated joint structure which can be manipulated for reanimation. Visual-quality of reconstructed models is limited due to both errors in reconstruction and constraints on model shape which do not represent geometry such as loose clothing or hair.

In this work we propose a temporally consistent structured representation based on spherical mapping and remeshing as a 2D geometry image. We introduce spherical matching that allows motion blending between geometry image sequences for motion control. This representation allows high-quality video-based rendering with motion control for reanimation without model-based constraints on surface shape.

2.2. Video-based animation

Video-based techniques such as video interpolation [BCS97, CG00] and video-sprites [SE02] provide an image-based representation of dynamic scenes allowing synthesis of novel image sequences with visual quality comparable to the captured video. Novel video sequences are synthesized by concatenating segments of the captured video based on a transition graph. This approach has been used to produce video-realistic synthesis of faces [BCS97, CG00, EGP02] or simple non-rigid subjects [SSSE00, SE02] with limited control over scene dynamics. Schodl et al. [SE02] use a similarity metric between video frames to identify transition points which minimize visual artifacts. Graphcut textures [KSE*03] optimized the transition over space and time between video sequences to achieve video quality composition of dynamic scenes.

These approaches are currently limited to a fixed viewpoint and non-rigid objects without self-occlusion. In this work we combine video-based rendering with video-based animation to provide a representation that allows free viewpoint visualization of complex non rigid motions of people with control of motion dynamics.

2.3. Motion graphs

Character animation from motion capture data often requires the modification of captured movement to satisfy constraints or transition seamlessly between sequences. Motion graphs for databases of human motion capture have been introduced

to allow reuse by recombining segments of the motion capture data [AFO03, KGP02, LCR*02, LAS02]. The problem of human motion synthesis is formulated as a graph search where the nodes correspond to frames in the motion database and edges are weighted by the similarity of frames. Novel animation sequences are synthesized by optimizing the path through the graph to satisfy external user constraints and minimize transition artifacts.

In this work we define a graph of human motions and transitional motions reconstructed from multiple view video. A similar approach is used in computer games for real-time character animation with a predefined set of transitions and corresponding sequences of motion capture data. Here the motions and transitions are defined by free viewpoint video sequences. A seamless blend between motions and transitions is achieved using spherical matching between two video sequences. The final motion graph enables controlled animation of the video-based character for seamless motion synthesis.

3. Video capture and representation

In this section we outline the process of reconstruction and representation of multiple view video as a geometry video. Figure 1 illustrates the process. Multiple view video sequences of people are first captured in a dedicated multiple camera studio. For each time frame, a triangulated surface mesh is then reconstructed. This surface is parameterized on the spherical domain [PH03] and resampled onto a spherical subdivision surface [PH03, ZBS04]. The subdivision mesh has a predefined mapping to the 2D domain and geometry plus color is mapped to a 2D geometry image. Finally, the geometry images created for a sequence are encoded as a single geometry video for rendering as free-viewpoint video.

3.1. Surface reconstruction

Multiple-view video is captured in a 10 camera studio with a blue-screen backdrop for foreground segmentation. Sony DXC-9100P 3-CCD color cameras are used, providing PAL-resolution progressive scan images at 25Hz. The cameras are positioned around a capture volume of 3m × 2m × 2m, sufficient for a single cycle of walk and jog motions. Intrinsic and extrinsic camera parameters are calibrated using a public domain calibration toolbox [Bou03].

Surface extraction is performed at each time frame using volumetric reconstruction. The visual-hull is first derived from segmented foreground image silhouettes, this is then refined to a color consistent photo-hull using the *Generalized Voxel Coloring* algorithm [CMS99]. The surface of the volumetric reconstruction is then extracted using the *Marching Cubes* algorithm [LC87] and smoothed to provide a continuous representation for the surface normals. Surface color is derived at each vertex by blending the sampled colors in each visible camera view [DYB98]. Blend weights are based
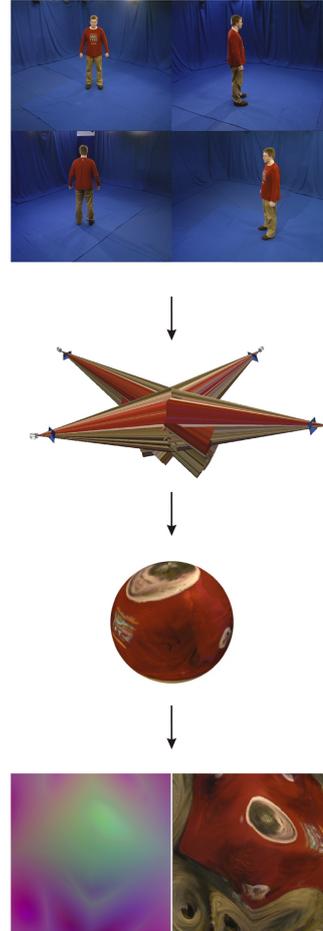

Figure 1: *Construction of a single frame of a sequence from multiple view video, showing volumetric reconstruction, spherical parameterization and mapping as an image with geometry plus color.*

on the sampling density in each view using the cosine of the angle between the camera to surface viewing direction and the surface normal.

3.2. Spherical mapping

Reconstructed surfaces are parameterized in the spherical domain using the coarse-to-fine technique introduced by Praun and Hoppe [PH03]. The starting point is a closed triangulated mesh M in $\Re^3$ of zero genus. It should be noted that this limits video capture to motions where the limbs do not connect with the body giving a non genus zero surface.

A unique 1-to-1 map is constructed for M onto the unit sphere $\hat{S}$, $(M \rightarrow \hat{S})$. The mesh M is simplified to a tetrahedron while creating a progressive mesh structure [Hop96]. Spherical mapping is achieved by first mapping the tetrahe-

dron to the unit sphere, then traversing the progressive mesh sequence and inserting vertices onto the sphere. A 1-to-1 map is achieved as progressive mesh construction maintains an embedding during mesh decimation and vertices are inserted into the kernel of their 1-neighborhood on the sphere, maintaining the embedding in the spherical domain [PH03].

For highly deformed genus-zero surfaces, producing a uniform sampling on the sphere during mapping is a challenging task. This represents a particular problem for the human body, where narrow limbs become highly distorted in the spherical domain. Praun and Hoppe [PH03] investigate several approaches to minimize distortion during mapping and perform mesh optimization to minimize the stretch in the representation with respect to the original mesh. In this work we do not seek to minimize the stretch, instead we seek a uniform sampling in the domain such that we can adequately sample highly deformed genus-zero surfaces. We propose a cost function derived from Zhou et al. [ZBS04] and optimize the mesh to minimize the vertex sampling density on the sphere during mapping.

The sampling density $D(\underline{s}_i)$ at a spherical vertex $\underline{s}_i$ is defined as follows, where i' spans the 1-neighborhood of a vertex i, N_i is the valence of the vertex and $\Delta_{ii'}$ is the spherical triangle with the directed edge ii' in the mesh.

$$D(\underline{s}_i) = \frac{N_i}{\sum_{i'} Area(\Delta_{ii'})} \quad (1)$$

We minimize the distortion in the vertex density on the sphere with respect to a target density D_0 using the following cost function, where $\nabla_{ii'}$ is the gradient on the edge ii', $\nabla_{ii'} = \underline{s}_{i'} - \underline{s}_i$.

$$E_{density}(S) = \sum_i \sum_{i'} \frac{D_0(\underline{s}_{i'})}{D(\underline{s}_{i'})} \|\nabla_{ii'} \underline{s}_i\|^2 \quad (2)$$

For density minimization on the unit sphere we assume a uniform low resolution target density of, $D_0 = 1$.

3.3. Image mapping

The spherical mesh is resampled onto a regular subdivision surface S by constructing a map $(S \to \hat{S})$ while maintaining the sampling density of the original mesh as proposed by Zhou et al. [ZBS04].

A subdivision surface S is constructed using regular 1-to-4 subdivision of a unit octahedron. The surface is updated to minimize the density cost function in Equation 2 with the target density derived from the surface $\hat{S}$. Gradient descent optimization corresponds to iterative application of the weighted laplacian operator $\Delta_{density}(\underline{s}_i)$ as proposed by Zhou et al. [ZBS04].

$$\Delta_{density}(\underline{s}_i) = \frac{1}{\sum_{i'} \frac{D_0(\underline{s}_{i'})}{D(\underline{s}_{i'})}} \sum_{i'} \frac{D_0(\underline{s}_{i'})}{D(\underline{s}_{i'})} (\underline{s}_i - \underline{s}_{i'}) \quad (3)$$

Given the mapping $M \to \hat{S}$ and $S \to \hat{S}$, we can resample the attributes of the original mesh M onto the uniform domain of the subdivision surface S. The surface is mapped to a geometry image by unfolding the subdivided octahedron onto a 2D grid with a 1-to-1 correspondence between the vertices of the subdivision surface and image pixels [PH03]. For a subdivided octahedron there will be $(2n+1) \times (2n+1)$ samples in the image domain. In this work we resample geometry onto a $(2n-1) \times (2n-1)$ image for efficient representation as a $(2n)^2$ image appropriate for rendering. We ensure that there is no loss of visual detail in resampling the geometry image by deriving color from the original camera images for the resampled geometry. We use multiple resolution blending [BA83] with spherical boundary continuity to fill missing colors in the final image. This process is performed for sequences of captured multiple view video to obtain a geometry video representation for each motion. The problem then is to blend different motions to achieve seamless character animation.

4. Motion blending

To construct seamless motion sequences for character animation it is necessary to blend between motions at the transition points, either at the start end of cyclic motions or when switching from one motion to another. In this section we introduce a multiple resolution coarse-to-fine algorithm for shape matching in the spherical domain. Spherical matching is employed to align the geometry images between overlapping motion sequences. Transitions between two sequences can then be created by blending between matched geometry images.

Surface matching for blending is formulated as a continuous bijective mapping between two surfaces $(S \to S_0)$ in the spherical domain. Our approach poses the mapping problem as an energy minimization task. We construct a cost function for joint estimation of correspondence and deformation in the spherical domain to minimize disparity in shape and appearance between two surfaces.

The matching function for minimization, $E_{cost}(S, \mathbf{W})$, is defined as a function of the set of vertices S for a surface in the spherical domain and a fuzzy multi-point correspondence $\mathbf{W} = \{w_{ij} \in (0,1)\}$ defining the match between two surfaces $S = \{\underline{s}_i\}$ and $S_0 = \{\underline{s}_j\}$. The function consists of three terms: $E_{disparity}$ measuring the distance in shape and color of the surfaces; E_{deform} measuring the distance between the surfaces in the spherical domain; and $E_{regularize}$ defining regularizing constraints on the deformation, subject to a control parameter λ.

$$E_{cost}(S, \mathbf{W}) = (1-\lambda)E_{disparity}(\mathbf{W}) \qquad (4)$$
$$+ (1-\lambda)E_{deform}(S, \mathbf{W})$$
$$+ \lambda E_{regularize}(S)$$

In the following sections we describe each of the components of the matching functional followed by the algorithm for optimization and spherical matching.

4.1. Surface disparity

Data fit is based on the similarity of a set of attributes defined on the surfaces S and S_0. In energy minimization we seek the correspondence $\mathbf{W}$ that minimizes the disparity in shape and appearance using the position $\underline{x}$, surface normal $\underline{n}$ and RGB color $\underline{c}$ of the resampled surfaces. The disparity is measured using a squared error metric and a separate energy function is constructed for each attribute as follows

$$E_{disparity}(\mathbf{W}) = \frac{1}{\sigma_x^2}\sum_i\sum_j w_{ij}\|\underline{x}_i - \underline{x}_j\|^2 \qquad (5)$$
$$+ \frac{1}{\sigma_n^2}\sum_i\sum_j w_{ij}\|\underline{n}_i - \underline{n}_j\|^2$$
$$+ \frac{1}{\sigma_c^2}\sum_i\sum_j w_{ij}\|\underline{c}_i - \underline{c}_j\|^2$$

The disparity terms are normalized by the expected variance σ^2 for a correct match. The variance parameters provide control over the expected range of disparities between two surfaces for each of the attributes used in matching. In this work we set these as follows, $\sigma_x = 5cm$, $\sigma_n = 0.5(30^o)$, and $\sigma_c = 50 \in (0,255)$. These values were derived using hand selected vertex correspondences between motion sequences.

4.2. Spherical deformation

The matching problem is posed as the deformation of the surface $S = \{\underline{s}_i\}$ to minimize the disparity with respect to the surface $S_0 = \{\underline{s}_j\}$ in the spherical domain, $\underline{s} \in \Re^2$. Surface deformation is driven by minimizing the distance on the sphere based on the multi-point correspondence $\mathbf{W}$ between the surfaces. In practise we linearize the problem and compute distances in $\Re^3$ while constraining vertices to remain on the unit sphere. The deformation energy function is defined as

$$E_{deform}(S, \mathbf{W}) = \frac{1}{T^2}\sum_i\sum_j w_{ij}\|\underline{s}_i - \underline{s}_j\|^2 \qquad (6)$$
$$+ \sum_i\sum_j w_{ij}\left(\log(w_{ij}) - 1\right)$$

A deterministic annealing approach [CR03] is adopted for optimization using an entropy term and temperature control parameter T in the deformation functional. The term w_{ij} defines a multi-point fuzzy correspondence between the two surfaces and the temperature provides control of the correspondence during optimization. At a high temperature value the correspondence becomes more fuzzy and as the temperature is successively reduced the correspondence can be hardened, providing a coarse-to-fine refinement of the surface correspondence.

4.3. Regularization

Regularization is used to minimize the distortion in the spherical domain while developing the map $S \rightarrow S_0$. Vertices $\underline{s}_i$ are updated sequentially and constrained to remain within the kernel of the 1-neighborhood. This ensures that the surface S maintains an embedding on the sphere. Left unconstrained the surface S will quickly reach a local minima with vertices collapsed onto the border of the 1-neighborhood. Regularization minimizes the distortion in the distribution of vertices such that the surface S can continue to deform to minimize the disparity function. This is achieved by minimizing the vertex density in the spherical domain during optimization

$$E_{regularize}(S) = \frac{1}{T^2}E_{density}(S) \qquad (7)$$

A weighting parameter λ (Equation 4), is used to control the trade-off between regularization and data-fitting in optimization. In this work a fixed value, $\lambda = 0.75$, is used. Spherical matching was found to be relatively insensitive to the value of λ, provided that it was sufficient to prevent collapsed vertex neighborhoods in optimization. The regularization term simply maintains an even vertex distribution such that the surface is free to deform and minimize the data fitting cost function.

4.4. Multiple resolution coarse-to-fine optimization

Joint estimation of correspondence and deformation is solved using a strategy of alternately minimizing the matching cost with respect to the correspondence matrix W and then updating the surface S to satisfy the correspondence. For a fixed surface configuration S a unique solution exists for the correspondence as follows. The constraint $\sum_j w_{ij} = 1$ is imposed to enforce surface assignment $S \rightarrow S_0$.

$$w'_{ij} = \exp\left(\frac{\|\underline{x}_i - \underline{x}_j\|^2}{\sigma_x^2}\right)\exp\left(\frac{\|\underline{n}_i - \underline{n}_j\|^2}{\sigma_n^2}\right) \qquad (8)$$
$$\times \exp\left(\frac{\|\underline{c}_i - \underline{c}_j\|^2}{\sigma_c^2}\right)\exp\left(\frac{\|\underline{s}_i - \underline{s}_j\|^2}{T^2}\right)$$
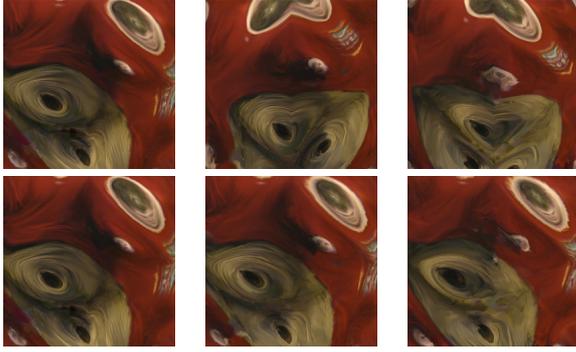
Figure 2: *Geometry images for frames 0, 5, 10 of a walk cycle showing (top) original images, (bottom) the result of spherical matching applied frame to frame in the sequence.*

$$w_{ij} = \frac{w'_{ij}}{\sum_j w'_{ij}} \qquad (9)$$

For a fixed correspondence **W** we use gradient descent to solve the continuous problem of surface deformation

$$\frac{dE_{cost}}{d\underline{s}_i} = (1-\lambda) w_{ij} \left(\underline{s}_i - \underline{s}_j \right) + \lambda \Delta_{density} \left(\underline{s}_i \right) \qquad (10)$$

The optimization algorithm proceeds by first finding the global rotation that minimizes the cost function. A multi-resolution strategy is then adopted for local optimization. Surface deformation is scheduled at successive resolutions of the subdivision surface S starting with the base octahedron. Coarse-to-fine matching is achieved by setting the temperature parameter T to restrict the range of matches w_{ij} to lie principally within the kernel of the 1-neighborhood for a vertex. This is achieved using a value of T equal to the edge length on the undeformed spherical mesh at each level of subdivision. Optimization at each resolution proceeds until the maximum update for a vertex reaches a fixed value $\varepsilon = 0.0001$ on the unit sphere.

The optimization algorithm terminates at the finest resolution of the subdivision surface S and provides a continuous 1-to-1 map between the surfaces S, S_0. The surface S can finally be resampled at the vertices of the target mesh S_0 and a new geometry image constructed such that S and S_0 are in correspondence in the 2D image domain for blending. Figure 2 shows the result of spherical matching applied to align geometry images.

5. Interactive character animation

In this section we outline the creation of motion graphs for video based character animation and the video renderer for interactive character control and free-viewpoint visualization.
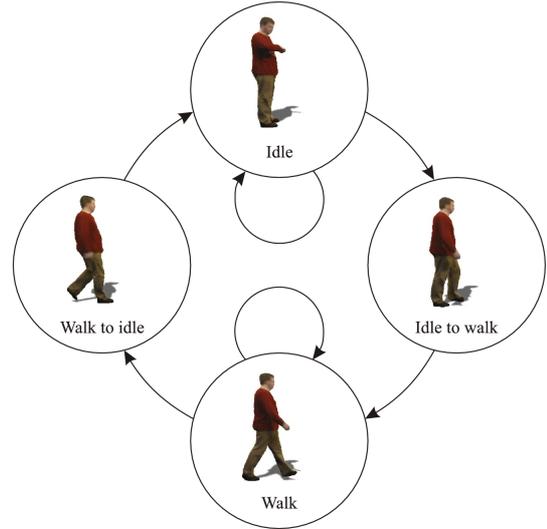

Figure 3: *Motion graph for character animation using cyclic (idle, walk) and transitional motions (walk to idle, idle to walk) represented by geometry videos.*

5.1. Motion graphs

We capture multiple view video sequences for a set of cyclic motions for a person such as *idle*, *walk* and *jog*. We also capture transitional motions such as *idle-walk*, *walk-idle*, *walk-jog* and *jog-walk*. The motion sequences and transitions are represented as a geometry video as outlined in Section 3. Seamless motions are created using the spherical matching algorithm introduced in Section 4. To create seamless cyclic motions the final frames of a sequence are blended with the frames preceding the start of the sequence. For transitional motions the start and end frames are blended with the preceding and succeeding cyclic motions. Motion sequences are temporally aligned using the heel strike position and a 5-frame linear blend is used between matched geometry images. The geometry videos are finally assembled into a motion graph. The graph contains a set of cyclic motion states for a character (*idle*, *walk* and *jog*) and a set feasible transitions (*idle-walk*, *walk-idle*, *walk-jog* and *jog-walk*) as illustrated in Figure 3. The motion graph allows interactive control of character state in the video renderer.

5.2. Video renderer

The motion graph for a character is defined by a set of motion states and transitional motions encoded as geometry videos in AVI format, together with the corresponding translational motion for the video sequences. We render geometry videos for free viewpoint visualization using Shader Model 3.0 compliant graphics hardware (nVidia 6600GT). This shader model supports texture lookup in the vertex shader, allowing the extraction of vertex position in a ge-

Figure 4: *255 × 255 geometry image with (left) geometry quantized at 1cm resolution with 24 bits/pixel, (middle) non-quantized geometry with 96 bits/pixel and (right) non-quantized geometry with view-dependent rendering.*

ometry image for real-time rendering. Our video renderer is implemented using DirectX 9 and the High Level Shader Language (HLSL). The renderer loads a motion graph for a character and provides interactive control of character state and the direction of character motion with free viewpoint control in visualization.

6. Results and Discussion

Character construction was tested using three subjects captured in multiple view video sequences performing idle, walking, jogging, and kicking motions together with the transitional movements such as walk-jog. Figure 1 illustrates a single frame of multiple view video recorded for an idle motion and the corresponding geometry image constructed for rendering.

We create geometry images at a resolution of 255 × 255 with geometry quantized to 255 levels at 1cm resolution in a 24 bits/pixel RGB image. There is a trade-off in our representation between the visual quality that can be achieved

in rendering and the memory overhead required for the representation. At low image resolutions there will be undersampling of both geometry and appearance in the geometry videos. At high resolutions we can achieve greater geometric detail and visual fidelity to the original multiple view video, at the cost of greater storage requirements. In this work we target a compact representation for our motion graph to minimize the memory overhead for real-time rendering. We found that a minimum resolution of 255 × 255 was required to retain the geometry at the tips of the limbs when remeshing onto the subdivision domain *S*. Greater visual fidelity could be achieved either by using a larger number of bits per pixel to increase geometric accuracy, using a larger image size for color texture to increase the sampling for appearance or by using view dependent rendering from the original video for highly realistic rendering at the cost of significantly higher storage requirements for a complete motion graph. Figure 4 compares the effect of quantized and unquantized geometry as well as view-dependent rendering for an image resolution of 255 × 255.

Animation control is achieved using a motion graph of geometry video sequences. Here we capture cyclic motions such as walking and jogging, and the transitional motions between these states. A user can control character dynamics either through the direction of motion when the character is in a cyclic state or by controlling the state of the character in the motion graph and transitioning between different cyclic motions. Figure 5 shows rendered frames for the transition between two cyclic states in the motion graphs created for the three captured subjects. Seamless cyclic motions and blends between transitional and cyclic motions are achieved using spherical matching and motion blending between the geometry videos in the motion graph. We are therefore able to control and create new 3D dynamic scenes of people from short sections of multiple view video sequences represented as free viewpoint geometry videos.

One of the principal advantages of our video-based representation for animation is the realism that can be achieved by replaying the surface dynamics captured from the original multiple view video. Figures 6 and 7 show screen captures from the video renderer with interactive animation control and free viewpoint rendering. Video clips for the different animations are provided for download from [VMR]. These movies demonstrate the realism that can be achieved by replaying the captured cloth motion, giving a *video-realistic* looking result in character animation. Our video renderer was tested on an Intel Pentium IV 3.2GHz, 1GB RAM PC with a NVIDIA 6600GT graphics card running Windows XP. A maximum frame rate of 88fps was achieved with a complete motion graph for a character stored in texture memory.

The geometry video representation has several advantages as a representation for free viewpoint rendering from multiple view video sequences of people. Geometry images pro-

Figure 5: *Animation control using a motion graph of geometry videos showing the transitions between motion states for three subjects, (top) walk-idle-walk, (middle) walk-jog-walk, and (bottom) stance-kick-stance.*
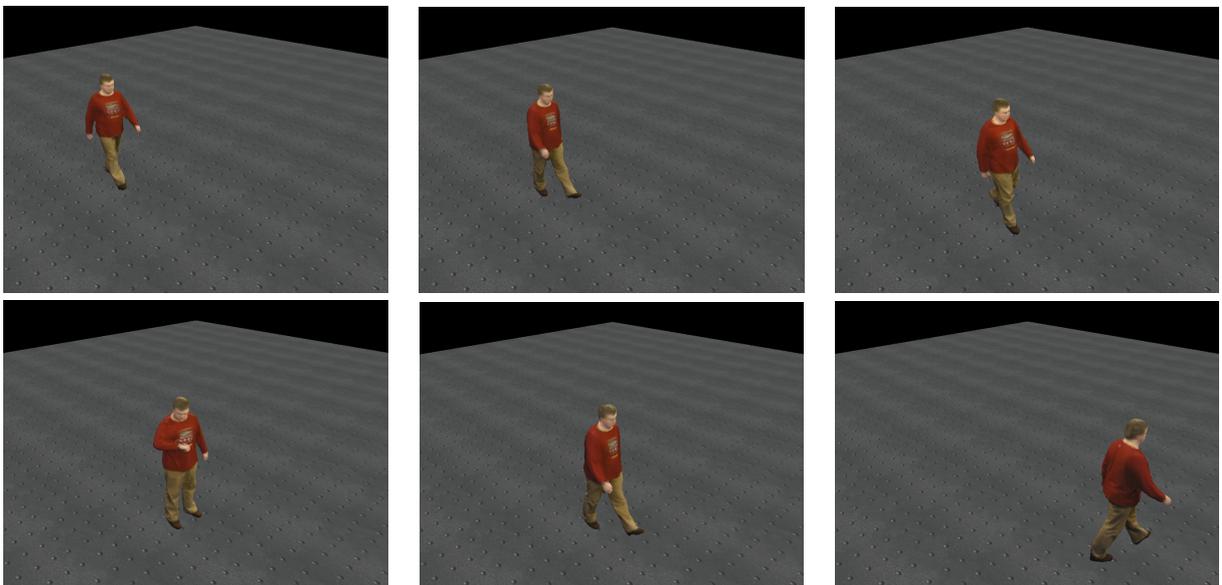

Figure 6: *Interactive character control and free viewpoint rendering for a subject performing walk and idle motions.*
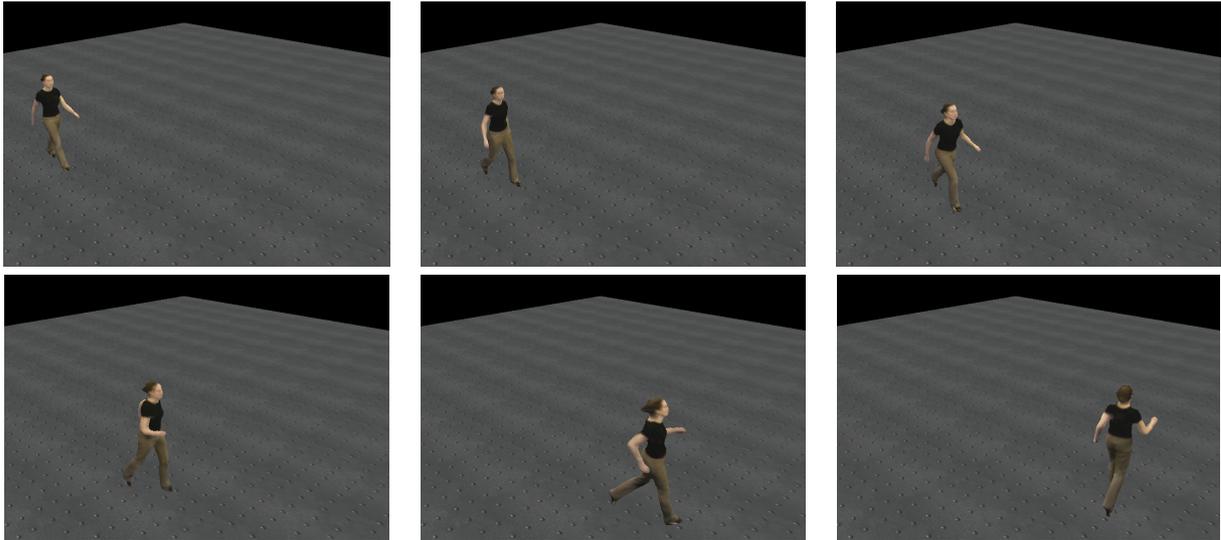
Figure 7: *Interactive character control and free viewpoint rendering for a subject performing walk and jog motions.*

vide a consistent temporal structure for level of detail and frame rate control [BSM*03] in free viewpoint video. Spherical matching enables the correspondence to be derived between spherical geometry images. This has been applied to blend motions in geometry videos. It also enables the recovery of temporal correspondence within a sequence. Figure 2 illustrates a temporally consistent geometry video created by matching successive frames of a walk cycle. This potentially provides a technique both for geometry and appearance editing in geometry videos, as well as for compression and efficient representation of free viewpoint video, subjects for future research.

7. Conclusions

A video-based representation has been introduced for free-viewpoint visualization and animation control of 3D character models of real people reconstructed from multiple view video. Character motions are represented as free viewpoint video reconstructed from multiple view video sequences captured in a 10 camera studio. Image based reconstruction is employed to extract a 3D surface for each frame of a motion sequence. Surfaces are parameterized on the spherical domain, then mapped to a 2D geometry image providing a geometry video representation for the video sequences. Spherical matching is used to blend motion sequences to provide seamless motion cycles and smooth transitions between different motions. A multiple resolution coarse-to-fine spherical matching algorithm has been developed that simultaneously optimizes across the entire spherical domain to provide the correspondence between spherical geometry images. The blended video sequences are constructed as an off-line process and represented as a motion graph for in-

teractive 3D character animation. A video renderer is used to provide real-time free viewpoint visualization and interactive character control, making use of Shader Model 3.0 compliant graphics hardware.

The representation for video-based character animation introduced in this paper has several significant advantages.

- Free viewpoint video captured from multiple view video sequences of people, creating highly realistic rendered animations with the dynamic shape and appearance captured in the original video.
- Compact representation of multiple view video for free viewpoint rendering using geometry videos, that provides a temporally consistent structure for level of detail and frame rate control in rendering [BSM*03].
- Spherical matching for motion blending between geometry videos to create seamless transitions between free viewpoint video sequences and provide temporal correspondence within a geometry video for compression.
- Character representation as a motion graph of free viewpoint video sequences for animation control and synthesis of new 3D content from captured motion clips with the realism of the captured video.
- Real-time free viewpoint visualization and interactive animation control of multiple view video using Shader Model 3.0 compliant graphics hardware.

References

[AFO03] ARIKAN O., FORSYTH D. A., O'BRIEN J. F.: Motion synthesis from annotations. *Proceedings ACM SIGGRAPH* (2003), 402–408. 3

[BA83] BURT P., ADELSON E.: A multiresolution spline

with application to image mosaics. *ACM Transactions on Graphics 2*, 4 (1983), 217–236. 4

[BCS97] BREGLER C., COVELL M., SLANEY M.: Video rewrite: Driving visual speech with audio. *Proceedings ACM SIGGRAPH* (1997), 1–8. 2

[Bou03] BOUGUET J.-Y.: *Camera Calibration Toolbox for Matlab: www.vision.caltech.edu/bouguetj/calib-doc*. Tech. rep., MRL-INTEL, 2003. 3

[BSM*03] BRICEÑO H., SANDER P., McMILLAN L., GORTLER S., HOPPE H.: Geometry videos: A new representation for 3d animations. *ACM Symposium on Computer Animation* (2003), 136–146. 2, 9

[CG00] COSATTO E., GRAF P.: Photo-realistic talking heads from image samples. *IEEE Transaction on Multimedia 2*, 3 (2000), 152–163. 2

[CMS99] CULBERTSON W. B., MALZBENDER T., SLABAUGH G.: Generalized voxel coloring. *Proceedings of the International Workshop on Vision Algorithms, Lecture Notes in Computer Science 1883* (1999), 100–115. 3

[CR03] CHUI H., RANGARAJAN A.: A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding 89* (2003), 114–141. 5

[CTMS03] CARRANZA J., THEOBALT C., MAGNOR M., SEIDEL H.-P.: Free-viewpoint video of human actors. *Proceedings ACM SIGGRAPH 22*, 3 (2003), 569–577. 1, 2

[DYB98] DEBEVEC P., YU Y., BORSHUKOV G.: Efficient view-dependent image-based rendering with projective texture-mapping. *9th Eurographics Rendering Workshop* (1998), 105–116. 3

[EGP02] EZZAT T., GEIGER G., POGGIO T.: Trainable videorealistic speech animation. *Proceedings ACM SIGGRAPH* (2002), 388–398. 2

[GGSC96] GORTLER S., GRZESZCZUK R., SZELISKI R., COHEN M.: The lumigraph. *Proceedings ACM SIGGRAPH 30* (1996), 43–54. 1

[GM04] GOLDLUECKE B., MARCUS M.: Space-time isosurface evolution for temporally coherent 3d reconstruction. *IEEE International Conference on Computer Vision and Pattern Recognition I* (2004), 350–355. 2

[Hop96] HOPPE H.: Progressive meshes. *Proceedings ACM SIGGRAPH* (1996), 99–108. 3

[KGP02] KOVAR L., GLEICHER M., PIGHIN F.: Motion graphs. *Proceedings ACM SIGGRAPH* (2002), 473–482. 2, 3

[KRN97] KANADE T., RANDER P., NARAYANAN P.: Virtualized reality: Constructing virtual worlds from real scenes. *IEEE Multimedia 4*, 1 (1997), 34–47. 2

[KSE*03] KWATRA V., SCHÖDL A., ESSA I., TURK G.,

BOBICK A.: Graphcut textures: Image and video synthesis using graph cuts. *Proceedings ACM SIGGRAPH* (2003), 277–286. 2

[LAS02] LI Y., ANG T., SHUM H.-Y.: Motion texture: A two-level statistical model for character motion synthesis. *Proceedings ACM SIGGRAPH* (2002), 465–471. 3

[LC87] LORENSON W., CLINE H.: Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics 21*, 4 (1987), 163–169. 3

[LCR*02] LEE J., CHAI J., REITSMA P. S., HODGINS J. K., POLLARD N. S.: Interactive control of avatars animated with human motion data. *Proceedings ACM SIGGRAPH* (2002), 491–500. 3

[LH96] LEVOY M., HANRAHAN P.: Light field rendering. *Proceedings ACM SIGGRAPH 30* (1996), 31–42. 1

[MBR*00] MATUSIK W., BUEHLER C., RASKAR R., GORTLER S., McMILLAN L.: Image-based visual hulls. *Proceedings of ACM SIGGRAPH* (2000), 369–374. 1, 2

[MTG97] MOEZZI S., TAI L., GERARD P.: Virtual view generation for 3d digital video. *IEEE Multimedia 4*, 1 (1997), 18–25. 2

[PF01] PLANKERS R., FUA P.: Articulated soft objects for video-based body modeling. *IEEE International Conference on Computer Vision* (2001), 394–401. 2

[PH03] PRAUN E., HOPPE H.: Spherical parameterization and remeshing. *Proceedings ACM SIGGRAPH* (2003), 340–349. 2, 3, 4

[SE02] SCHÖDL A., ESSA I.: Controlled animation of video sprites. *ACM Symposium on Computer Animation* (2002), 121–127. 1, 2

[SH03] STARCK J., HILTON A.: Model-based multiple view reconstruction of people. *IEEE International Conference on Computer Vision* (2003), 915–922. 1, 2

[SH05] STARCK J., HILTON A.: Virtual view synthesis of people from multiple view video sequences. *Graphical Models* (to appear 2005). 1, 2

[SSSE00] SCHÖDL A., SZELISKI R., SALESIN D., ESSA I.: Video textures. *ACM Symposium on Computer Animation* (2000), 489–498. 1, 2

[VBK02] VEDULA S., BAKER S., KANADE T.: Spatio-temporal view interpolation. *Eurographics Workshop on Rendering* (2002), 1–11. 1, 2

[VMR] www.ee.surrey.ac.uk/research/vssp/vmrg/vcphome.htm. 7

[ZBS04] ZHOU K., BAO H., SHI J.: 3d surface filtering using spherical harmonics. *Computer-Aided Design,Elsevier 36* (2004), 363–375. 3, 4

[ZKU*04] ZITNICK C., KANG S. B., UYTTENDAELE M., WINDER S. A. J., SZELISKI R.: High-quality video view interpolation using a layered representation. *Proceedings ACM SIGGRAPH 23* (2004), 600–608. 1, 2