Surface Capture for Performance-Based Animation

Jonathan Starck and Adrian Hilton
University of Surrey, Guildford, UK

Surface capture is a fully automated system for capturing a human's shape and appearance as well as motion from multiple video cameras to create highly realistic animated content from an actor's performance in full wardrobe.

Creating realistic animated models of people is a central task in digital content production. Traditionally, highly skilled artists and animators construct shape and appearance models for a digital character. They then define the character's motion at each time frame or specific key-frames in a motion sequence to create a digital performance. Increasingly, producers are using motion capture technology to record animations from an actor's performance. This technology reduces animation production time and captures natural movements to create a more believable production. However, motion capture requires the use of specialist suits and markers and only records skeletal motion. It lacks the detailed secondary surface dynamics of cloth and hair that provide the visual realism of a live performance.

Over the last decade, we've investigated studio capture technology with the objective of creating models of real people that accurately reflect the time-varying shape and appearance of the whole body with clothing.[1] We've developed a system for surface motion capture that unifies the acquisition of shape, appearance, and motion of the human body from multiple-view video (see also the "Related Work" sidebar on page 22). It captures a shape and appearance model for the performer in full wardrobe as well as the model animation and surface dynamics to create highly realistic digital content from the performance, unencumbered by a specialist suit.

Our system solves two key problems in performance capture: scene capture from a limited number of camera views and efficient scene representation for visualization. The multiple-view video and scene reconstruction in this work is available at http://www.ee.surrey.ac.uk/cvssp/vmrg/surfcap.

Surface capture

We record an actor's performance in a dedicated multiple-camera studio with controlled lighting and a chroma-key background. Because of cost, rather than use professional cameras such systems typically use machine vision cameras, which don't give the color quality required for production work in broadcast or film. Our work aims to demonstrate our surface capture technology's potential for high-quality entertainment content. We therefore developed a studio system using a limited number of professional film-quality high-definition (HD) cameras.

In our studio, we spaced eight HD cameras equally around a circle of 8 meters in diameter about 2 meters above the studio floor (see Figure 1). This gives a performance volume of $4 \times 4 \times 2$ meters with a wide-baseline 45-degree angle between adjacent camera views. We capture performances using Thomson Viper cameras in High-Definition Serial Digital Interface (HD-SDI) 20-bit 4:2:2 format with $1{,}920 \times 1{,}080$ resolution at 25 Hz progressive scan. Eight dedicated PC capture boxes using Digital Video Systems HD capture cards record synchronized video from all eight cameras uncompressed direct to disk. This studio setup enables high-quality performance capture from wide-baseline camera positions.

Studio calibration

Recording a performance from multiple video cameras shows the actor from a fixed set of viewpoints. Providing a complete 3D model for interactive visualization requires geometric surface reconstruction, as Figure 2 shows. To recover geometric shapes from camera images, we must calibrate the camera parameters. To calibrate a camera, we image a known object and derive the camera transformation that reproduces the object in a set of captured images. Techniques typically use a planar grid of known geometry that's visible from all cameras. Several public domain tools are available to achieve this (for example, http://www.vision.caltech.edu/bouguetj/calib_doc). This approach is suitable for camera systems in which the images have an overlapping field of view, but is impractical in studios where cameras surround the capture volume. For studio production where cameras might be reconfigured many times in one day with location and zoom changes, a simple and quick method of calibration is of key practical importance.

For rapid and flexible calibration of a multiple camera studio, we developed a wand-based calibration technique. Wand-based calibration uses two spherical

 0272-1716/07/$25.00 © 2007 IEEE

Related Work

Manually creating visually realistic digital models of people is a huge task made all the more difficult by our knowledge about how people appear and move in the world around us. Approaches to achieving realism in computer graphics range from physical simulation of surface geometry and light interaction in a scene to direct observation and replay from the real world. Over the last decade, computer graphics and vision techniques have converged to achieve realism by using observations from camera images. This synthesis by example process simplifies the complex task of physical simulation and can produce stunning results simply by reusing real-world content.

Kanade et al., who coined the term "virtualized reality," popularized reconstruction and rendering images of people from multiple camera views.[1] They used a 5-meter dome with 51 cameras to capture an actor's performance and replayed the event in 3D to create an immersive, virtualized view. Rendering virtual views of moving people from multiple cameras has since received considerable interest, and researchers have developed systems for multiple-view reconstruction and video-based rendering.[2-8] These techniques create a 3D video, also called *free-viewpoint video*, at a quality that now approaches the original video images.[9,10]

Whole-body images of people present several important challenges for conventional computer vision techniques in free-viewpoint video production, as Figure A on page 28 illustrates.

- *Uniform appearance.* Extended areas of uniform appearance for skin and clothing limit the image variation to accurately match between camera views to recover surface shape.
- *Self occlusions.* Articulation leads to self-occlusions that make matching ambiguous with multiple depths per pixel, depth discontinuities, and varying visibility across views.
- *Sparse features.* Shape reconstruction must match features such as clothing boundaries to recover appearance without discontinuities or blurring, but provide only sparse cues in reconstruction.
- *Specularities.* Non-Lambertian surfaces such as skin cause the surface appearance to change between camera views, making image matching ambiguous.
- *Wide baseline.* With a restricted number of cameras, you need a wide baseline configuration for 360-degree coverage, leading to large distortions in appearance between views.

Reconstruction algorithms are based either in the 2D domain, and search for image correspondence to triangulate 3D position; or in the 3D domain, and derive the volume that projects consistently into the camera views. Image-

Continued on page 28

1 Surface capture from multiple video images records an actor's complete appearance without requiring specialist suits and markers. We use eight camera views, providing 360-degree coverage from wide-baseline camera positions at 45-degree intervals.

markers at a known distance apart on a rigid rod. By acquiring video sequences of the moving wand, the technique builds up large sets of point correspondences between views in a short time. These point correspondences replace the planar grid and allow markers to be simultaneously visible from opposing camera views.

Our calibration algorithm uses wand markers to estimate both the intrinsic (focal-length, center-of-projection, and radial-distortion) and extrinsic (pose and orientation) camera parameters. This approach lets us calibrate studio camera systems in less than 10 minutes with an accuracy comparable to grid-based calibration. In addition, it doesn't require all cameras' fields of view to overlap, allowing flexible calibration of studio camera systems with extended capture volumes. A public domain implementation of the wand calibration and further technical details are available at http://www.ee.surrey.ac.uk/cvssp/vmrg/wandcalibration.

Surface geometry reconstruction

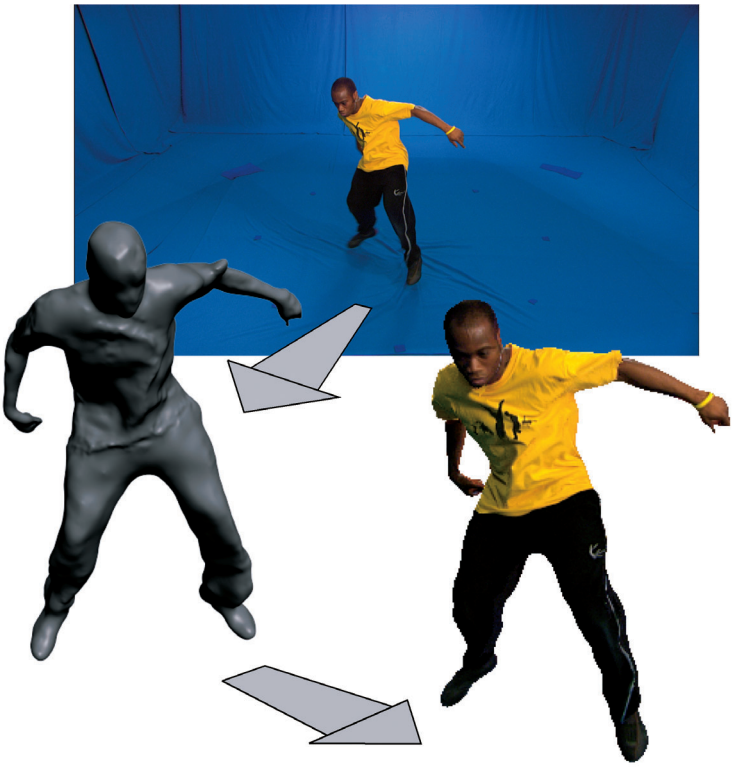
After calibrating the camera system, we can record a performance from multiple viewpoints for reconstruction. Scene reconstruction recovers a 3D model for a scene that places the appearance sampled in the camera images in correspondence in a process termed *image-based modeling*. Our reconstruction algorithm specifically provides robust shape reconstruction from wide-baseline camera views without losing visual detail such as creases in clothing. We achieve this by using multiple shape cues from camera images to constrain the geometry:

- With a fixed chroma-key backdrop, we can reliably extract foreground silhouettes to constrain the geometry's outline.
- Image features at appearance discontinuities, such as clothing boundaries, provide a dominant cue that can be reliably matched across wide-baseline images to constrain the surface position.
- Conventional appearance matching based on surface color or intensity provides only a weak cue with wide-baseline cameras that will define the dense surface geometry.

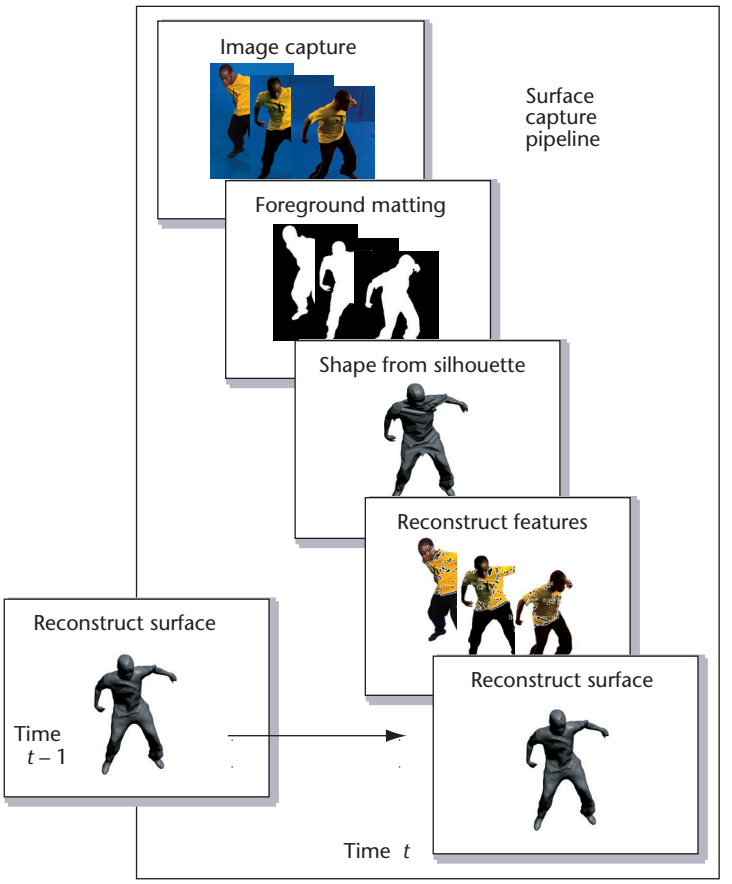
We combine all shape cues within a single framework that recovers the surface with a maximum appearance consistency between camera images while constrained to match extracted foreground silhouettes and feature correspondences.

We divide our surface reconstruction pipeline into seven distinct steps, outlined in Figure 3:

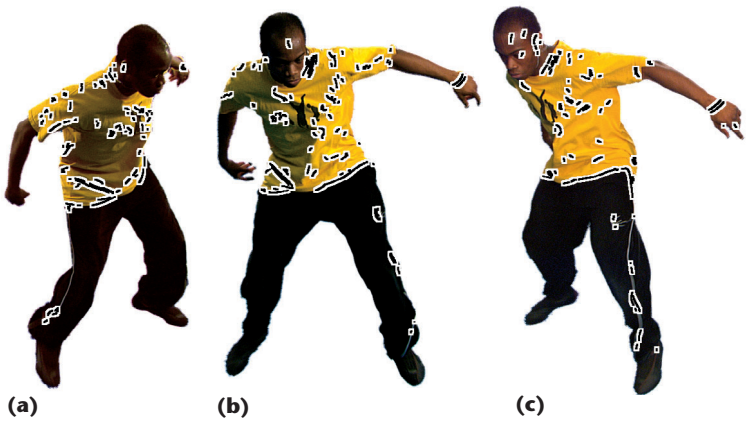
1. We record a multiple-view video sequence of an actor's performance.
2. We perform chroma-key matting to extract foreground silhouettes from the camera images, separating the image's foreground pixels from the known background color.
3. We extract an alpha matte for each image that defines the foreground opacity at each pixel and the foreground color where pixels in the original image are mixed between foreground and background.



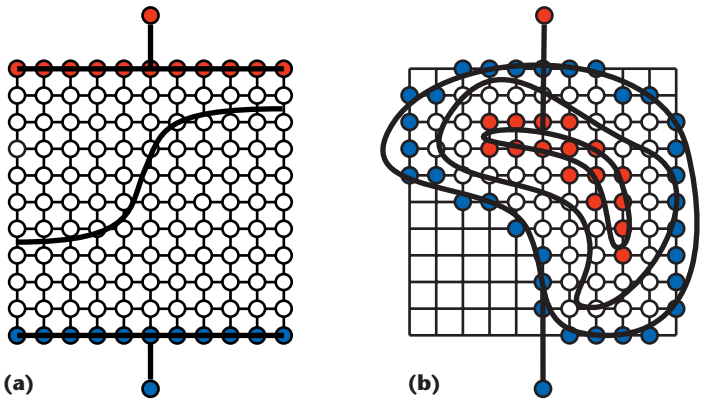
2 We use video capture to reconstruct a surface model of an actor for 3D visualization.



3 Overview of the automated surface capture pipeline.



4 We match feature lines between adjacent views and prune them to a left-right consistent set by enforcing reciprocal correspondence. We use the reconstructed surface contours to constrain surface reconstruction at the features.



5 Surface reconstruction as a cut on a discrete volumetric graph shown here in cross-section for (a) a planar scene and (b) a generalized scene. The maximum-flow on the graph between a source (blue) and sink (red) node saturates the set of edges where the edge weight is minimized and consistency between the camera images is maximized, corresponding to the scene surface.

4. We use the silhouettes to derive the *visual hull*—that is, the maximal volume in the scene that reproduces the silhouettes in the camera images.
5. The visual-hull defines an upper-bound on the scene's true volume and so constrains the feasible space for surface reconstruction.
6. We perform wide-baseline feature matching between the cameras to extract contours on the underlying surface inside the visual-hull that produce feature lines in the images.
7. Finally, we reconstruct the scene as the surface within the visual hull that passes through the surface features while reproducing the silhouette images and maximizing the consistency in appearance between views.

Visual-hull reconstruction. Shape reconstruction from silhouettes, or shape-from-silhouette (SFS), is a popular technique for scene reconstruction because of the reconstruction algorithm's simplicity and the robust nature of shape recovery in the studio setting, where the foreground can be reliably and consistently extracted from a known fixed background. However, SFS only provides an upper bound on the scene's volume; it doesn't reconstruct concavities that are occluded in silhouettes or match appearance across images. In addition, phantom false-positive volumes that are consistent with the image silhouettes can occur. Previous work has used multiple shape cues in reconstruction by iteratively refining the visual hull's shape. Local shape optimization is, however, subject to local minima and the surface retains the phantom structures in the visual hull. We extend recent work on global optimization techniques[2] to integrate multiple shape cues for robust wide-baseline reconstruction without restriction to a deformation with respect to the visual-hull surface.[3]

Feature matching. Once we've defined the extent of the scene by reconstructing the visual hull, we match surface features between views to derive constraints on the scene surface's location. Surface features correspond to local discontinuities in the surface appearance, and we extract candidate features in the camera images using a Canny-Deriche edge detector. We match each feature contour in an image with the appearance in an adjacent camera view. We first constrain correspondence to satisfy the camera epipolar geometry defining the relationship between observations in pairs of cameras. Each feature pixel corresponds to a ray in space connecting the scene surface and the camera's center of projection. This ray, in turn, projects to a line of pixels—an epipolar line—in an adjacent camera view, and the feature pixel can only match along this line. We further constrain correspondence by intersecting this ray with the feasible volume, defined by the visual hull giving a set of line segments in the adjacent camera view. We then derive the connected set of pixel correspondences in the adjacent view, which maximizes the image correlation for the feature contour. We verify correspondence by enforcing left-right consistency between views such that a feature pixel in one camera must match a feature pixel in an adjacent camera with a reciprocal correspondence. Figure 4 shows the set of left-right consistent features derived for a set of camera images.

Dense reconstruction. Feature reconstruction provides only a sparse set of 3D line segments that potentially lie on the scene surface. So, we perform dense surface reconstruction inside the volume defined by the visual hull. We adopt a global optimization approach by discretizing the volume and treating reconstruction as a maximum-flow/minimum-cut problem on a graph defined in the volume.

Figure 5 illustrates surface reconstruction as a network flow problem on a graph. Each discretized element of the volume, or *voxel*, forms a node in the graph, with graph edges connecting adjacent voxels. A cost defined by the consistency in appearance between camera images weights the edges. The maximum flow on the graph saturates the set of edges where the cost is minimized and the consistency is maximized. We can then extract the final surface as the set of saturated edges cutting the graph.

6 **Combining silhouette, feature, and stereo cues from eight wide-baseline camera views improves surface shape reconstruction over multiple-view stereo and shape from silhouette: (a) shape from silhouette, (b) merged multiple-view stereo, and (c) surface capture fusing silhouette, feature, and stereo cues.**

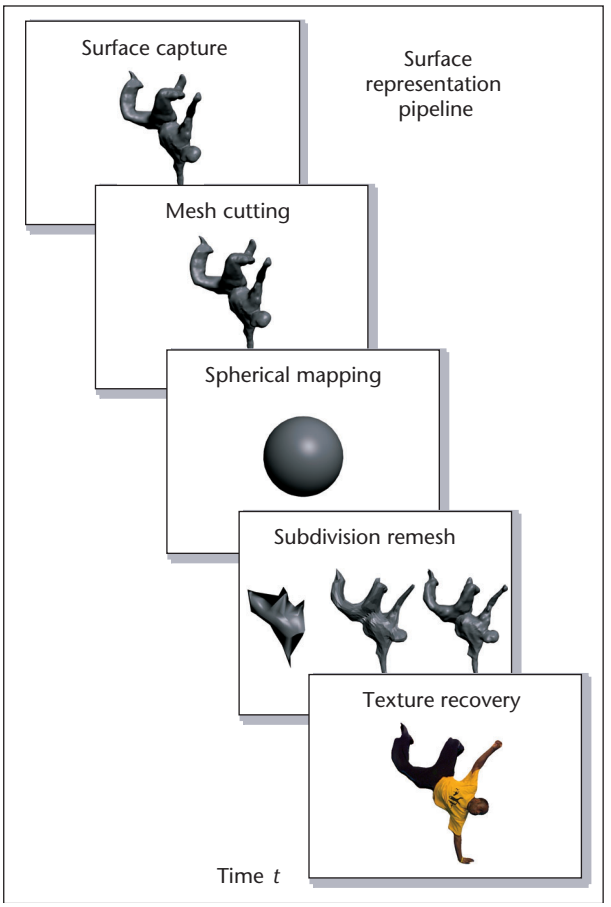
Some existing optimization methods use graph cuts, providing the global optimum that maximizes the correlation between views on the final surface.[4] Our approach adapts graph-cut optimization to derive a surface that passes through the reconstructed feature contours where possible and reproduces the initial set of silhouette images. We further constrain the reconstruction to be temporally consistent by minimizing the distance between surfaces at subsequent time frames.

Surface extraction. We extract the surface for the scene from the volume reconstruction as a triangulated mesh. We derive mesh vertices to subvoxel accuracy using a local search to maximize image consistency across all visible cameras. Figure 6 compares the result of surface reconstruction with results of conventional reconstruction techniques. The visual-hull alone (Figure 6a) provides only an approximate estimate of the scene geometry that incorporates phantom volumes. As Figure 6b shows, conventional multiview stereo leads to a noisy 3D surface estimate in matching adjacent views with wide-baseline cameras. Global surface optimization using multiple shape cues (see Figure 6c) combines robust shape reconstruction from silhouettes with appearance matching across camera views.

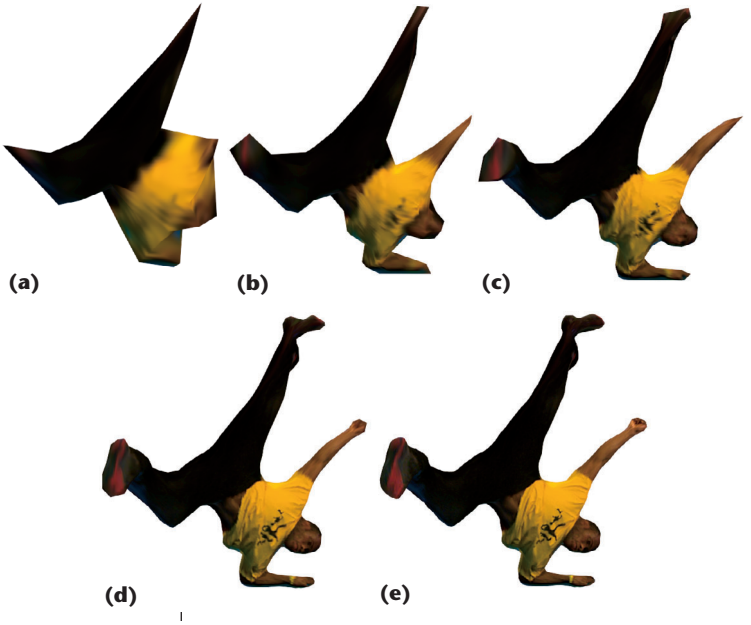
Scene representation

Studio capture records multiple video streams of a human performance from a specific set of viewpoints. Our studio calibration and scene reconstruction system enables high-quality recovery of the 3D time-varying surface during a performance. Even with a limited number of camera views, this capture represents a huge overhead in terms of the stored data. With eight high-definition images this equates to approximately 50 Mbytes per frame or 75 Gbytes per minute. Our system must efficiently represent this massive amount of video and geometry data to allow streaming for real-time rendering.

Figure 7 (on page 26 outlines our pipeline for constructing a structured surface representation. Surface capture initially provides a time-varying sequence of triangulated surface meshes in which the surface sampling, geometry, topology, and mesh connectivity changes at each time frame for a 3D object. We transform this unstructured representation to a single consistent mesh structure such that the mesh topology, connectivity, and texture domain is fixed, and only the geometry changes over time. We achieve this by mapping each mesh onto the spherical domain and remeshing as a fixed subdivision sphere.



7 Overview of the automated surface representation pipeline.



8 Remeshing as a constant topology subdivision surface reduces the geometry overhead and provides level-of-detail control. We compare the uncompressed overhead for the geometry and texture at different levels with the raw video capture at 50 Mbytes per frame: (a) 5 Kbytes per frame, (b) 19 Kbytes per frame, (c) 74 Kbytes per frame, (d) 291 Kbytes per frame, and (e) 1,158 Kbytes per frame.

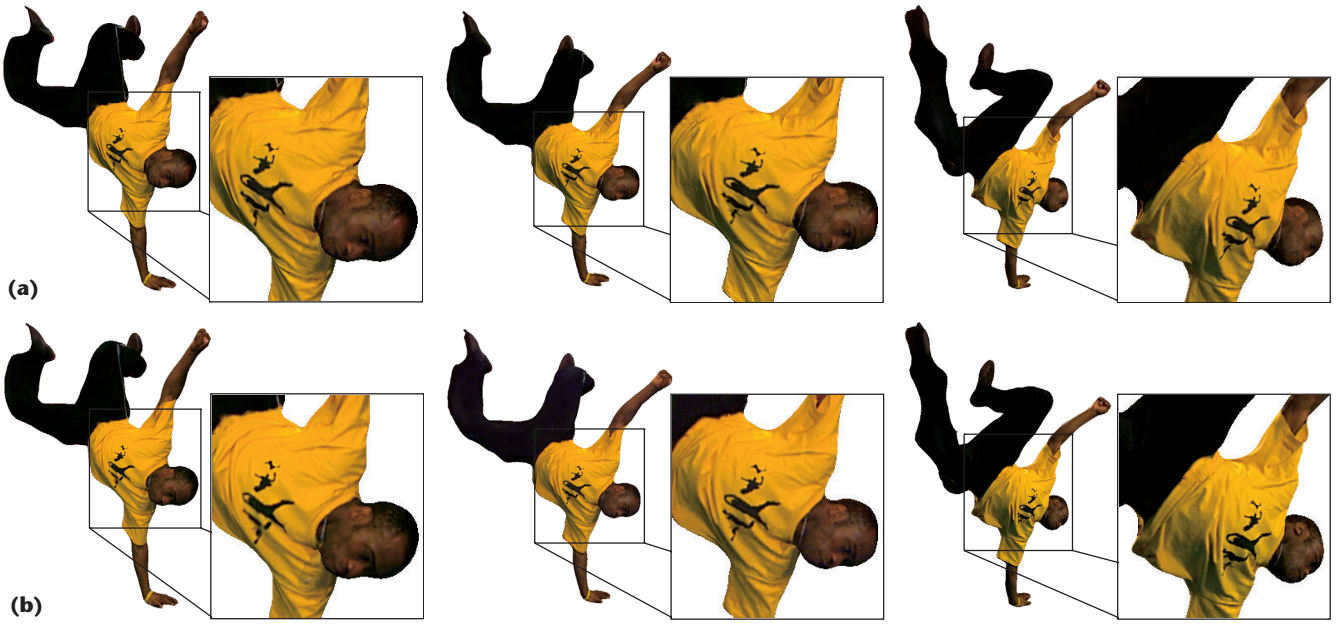
Praun and Hoppe[5] introduce a robust approach to parameterization of genus-zero surfaces in the spherical domain. *Surface genus* is the maximum number of cuts that can be made before a surface becomes disconnected, and genus-zero surfaces are topologically equivalent to a sphere. Our surface remeshing algorithm extends this technique to handle genus-N surfaces and incorporates adaptive resampling[6] to handle the mapping of highly deformed surfaces, such as the human body, onto the spherical domain. The final step is to combine the appearance from the original camera images as a single time-varying texture map for the subdivision surface.

Mesh cutting. Captured sequences of people often have non-genus-zero topology, and we first transform a genus-N surface using mesh cutting.[7] We construct a closed genus-zero mesh M by iteratively inserting a series of cuts on a reconstructed surface. The algorithm first creates a topological graph of isocurves on the mesh. Loops in the graph are then identified and the mesh is cut along the shortest isocurve in each loop. The mesh becomes topologically equivalent to a trimmed sphere and the holes are triangulated to create a closed genus-zero surface.

Spherical parameterization. We construct an embedding for the mesh on the unit sphere,[5] and simplify the mesh M to a tetrahedron by iteratively removing vertices from it. We then map the tetrahedron to the unit sphere and reverse the vertex removal, reinserting the vertices onto the spherical domain. We achieve a 1-to-1 mapping because vertex removal maintains an embedding during mesh simplification, and the mapping inserts vertices into the kernel of their neighborhood on the sphere, maintaining the embedding in the spherical domain.

Adaptive remeshing. We resample the spherical mesh onto a regular quaternary subdivision of a unit octahedron S by constructing a map . Embedding a complex genus-zero surface such as a person on the unit sphere requires a high degree of mesh deformation, resulting in a highly distorted parameterization. To accurately represent complex geometry during mapping, we optimize the mesh to match the vertex sampling density.[6] Given the mapping, we can finally resample the attributes of the original mesh M onto the uniform domain of the subdivision surface S.

Texture recovery. We resample surface appearance in the texture domain from the camera view with the greatest surface sampling rate, retaining the highest resolution appearance in the texture. We group the assignment of mesh facets to camera images to create maximal contiguous regions, minimizing the boundary on the surface between images. We then use multiple resolution blending[8] with spherical surface continuity to construct a single seamless texture. This multiresolution approach ensures that the extent of texture blending corresponds to the spatial frequency of the image's features, preserving the higher frequency detail that can become blurred with simple linear texture-blending techniques. Blending at low frequencies can compen-



9 Accurate surface reconstruction lets us extract a single surface texture, resulting in a smaller overhead than conventional free-viewpoint video, where the original camera images serve as a set of view-dependent textures. (a) Single-texture rendering, 1 Mbyte per frame uncompressed. (b) View-dependent rendering, 50 Mbytes per frame uncompressed.

sate for discrepancies in the color balance between views, although in practice, color calibration should precede image capture.

Next, we transform the unstructured surface motion sequence into a single mesh structure as a subdivision sphere S and represent the time-varying surface geometry as a single time-varying vertex buffer with a predefined overhead. The subdivision connectivity in the mesh allows for level-of-detail control to manipulate this overhead in the geometric representation. Figure 8 illustrates the representation at different levels in the subdivision hierarchy for the surface.

Our texture-recovery process differs from current techniques for free-viewpoint rendering of human performance, which typically use the original video images as texture maps in a process termed *view-dependent texturing*. View-dependent texturing uses a subset of cameras that are closest to the virtual camera as textured images, with a weight defined according to the cameras' relative distance to the virtual viewpoint. By using the original camera images, this can retain the highest-resolution appearance in the representation and incorporate view-dependent lighting effects such as surface specularity.

View-dependent rendering is often used in vision research to overcome problems in surface reconstruction by reproducing the change in surface appearance that's sampled in the original camera images. There are however several limitations. Firstly, storing, streaming, and rendering all camera images at 50 Mbytes per frame results in a large overhead. Secondly, resolution and view-dependent reflectance effects are only retained where the geometry is exact, such that the camera images are well aligned for blending. Blending with incorrect alignment produces image blurring and double-exposure effects in a synthesized view. Our surface-capture technique optimizes the alignment of surface geometry between camera images such that we can recover a single texture from all cameras without visual artifacts, significantly reducing the overhead in representing appearance.

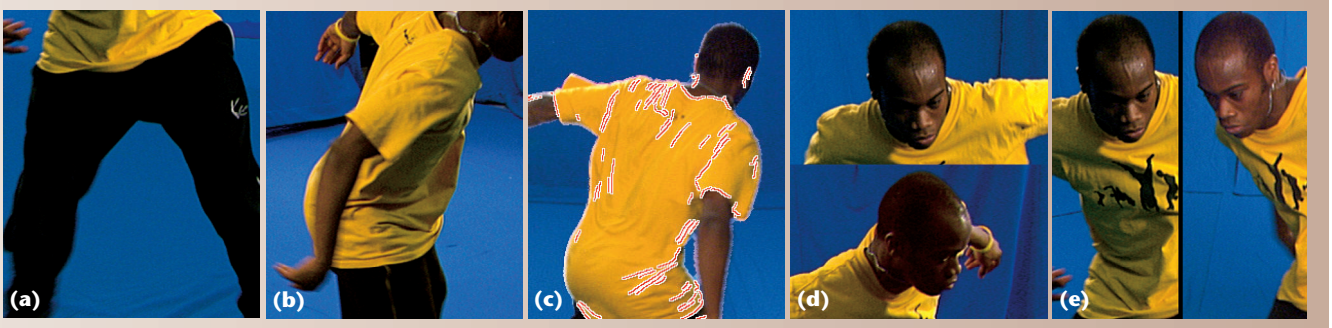
Figure 9 compares single-texture rendering with view-dependent rendering. The figure shows an equivalent visual quality for both techniques. You can see a small amount of blurring in the view-dependent rendering images, where we can't achieve exact subpixel alignment across a 45-degree camera baseline. We remove this blurring in the texture-resampling stage by recovering surface appearance from the camera images with the highest sampling rate and using a nonlinear multiple-resolution blend between the appearance in each camera.

Animation by example

Surface capture and representation provide the data necessary to replay a human performance in 3D while controlling the overhead to allow streaming and real-time rendering. This provides a 3D video representation, or free-viewpoint video, where the user controls the camera viewpoint. Our goal is to use free-viewpoint video for content production rather than to simply replay a fixed event as a virtualized reality.

Researchers have proposed example-based techniques for animation to reuse content captured using conventional motion-capture technology for skeletal motion synthesis.[9] These techniques use a library of captured performance and concatenate motion segments to produce new content. Animation by example has already found widespread use in computer games, which reuse motion capture to synthesize character actions. These techniques typically compile skeletal animation for a digital character into a library of motions, or *performance library*. They use animations from this library to construct a move tree graph structure defining

Related Work *Continued from page 22*



A Whole-body images present several important challenges with (a) uniform surface appearance, (b) self occlusions, (c) sparse features, (d) non-Lambertian surfaces, and (e) large distortions between views with wide baseline camera positions.

based correspondence forms the basis for conventional stereo vision, where pairs of camera images are matched to recover a surface.[1] However, image-based correspondence fails when matching is ambiguous. It requires fusing surfaces from stereo pairs, which is susceptible to errors in the individual surface reconstruction. A volumetric approach, on the other hand, allows inference of visibility and integration of appearance across all camera views without image correspondence. Shape-from-silhouette techniques[2] derive the visual hull, the maximal volume that reproduces a set of foreground silhouettes in the cameras. This is refined in space-carving techniques,[6] which provide the photo-hull—that is, the maximal volume that has a consistent foreground color across all visible camera images. Researchers have combined silhouette and color cues for robust shape reconstruction using iterative shape optimization techniques.[10-12]

Shape reconstruction for free-viewpoint video[1-10] simply allows the replay of a recorded event in 3D without the structure necessary to create or manipulate content for animation production. Model-based shape reconstruction techniques[3,13] fit a generic humanoid model to multiple-view images, providing a model structure to enable motion editing and retargeting. However, model-based techniques are limited by the predefined model structure, and you can't apply them to complex scenes with large changes in structure—for example, where loose clothing causes large changes in the surface geometry.

On the other hand, data-driven techniques with no prior model have demonstrated highly realistic synthesized animations in the 2D domain by replaying sequences from example video clips. Resampling video sequences of simple dynamic scenes[14] has achieved video-quality animation for a single fixed viewpoint. In related work,[15] we proposed animation by example using free-viewpoint video of human motions to provide a complete 3D digital representation. In this article, we present a complete system for surface capture and a free-viewpoint video representation suitable for use in animation production from a recorded human performance.

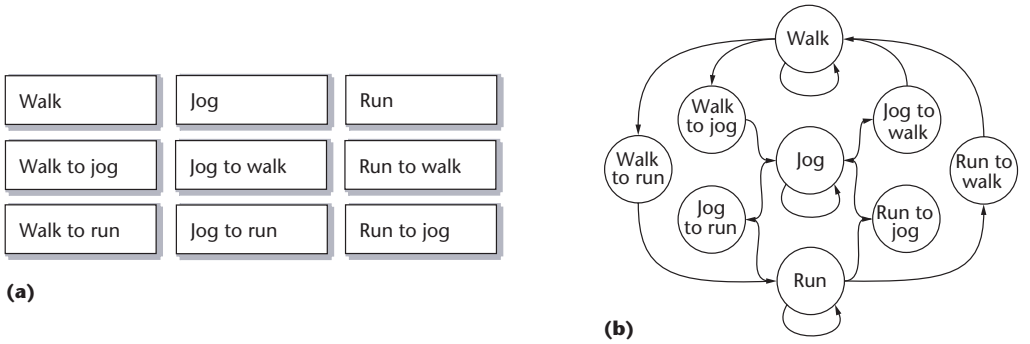
References

1. T. Kanade, P. Rander, and P. Narayanan, "Virtualized Reality: Constructing Virtual Worlds from Real Scenes," *IEEE MultiMedia*, vol. 4, no. 1, 1997, pp. 34-47.
2. W. Matusik, C. Buehler, and L. McMillan, "Polyhedral Visual Hulls for Real-time Rendering," *Proc. Eurographics Workshop Rendering*, Springer-Verlag, 2001, pp. 115-126.
3. J. Carranza et al., "Free-Viewpoint Video of Human Actors," *ACM Trans. Graphics*, vol. 22, no. 3, 2003, pp. 569-577.
4. O. Grau, "Studio Production System for Dynamic 3D Content," *Proc. Int'l Soc. Optical Eng. Visual Comm. and Image Processing*, vol. 5150, SPIE, 2003, pp. 80-89.
5. T. Matsuyama et al., "Real-time 3D Shape Reconstruction, Dynamic 3D Mesh Deformation, and High Fidelity Visualization for 3D Video," *Computer Vision and Image Understanding*, vol. 96, no. 3, 2004, pp. 393-434.
6. S. Vedula, S. Baker, and T. Kanade, "Image-Based Spatio-Temporal Modeling and View Interpolation of Dynamic Events," *ACM Trans. Graphics*, vol. 24, no. 2, 2005, pp. 240-261.
7. M. Waschbsch et al., "Scalable 3D Video of Dynamic Scenes," *The Visual Computer: Int'l J. Computer Graphics*, vol. 21, no. 8-10, 2005, pp. 629-638.
8. J. Allard et al., "The Grimage Platform: A Mixed Reality Environment for Interactions," *Proc. 4th IEEE Int'l Conf. Computer Vision Systems*, IEEE CS Press, 2006, p. 46.
9. C. Zitnick et al., "High-Quality Video View Interpolation Using a Layered Representation," *ACM Trans. Graphics* (Proc. Siggraph 2004), vol. 23, no. 3, 2004, pp. 600-608.
10. J. Starck and A. Hilton, "Virtual View Synthesis of People from Multiple View Video Sequences," *Graphical Models*, vol. 67, no. 6, 2005, pp. 600-620.
11. C. Hernandez Esteban and F. Schmitt, "Silhouette and Stereo Fusion for 3D Object Modeling," *Computer Vision and Image Understanding*, vol. 96, no. 3, 2004, pp. 367-392.
12. Y. Furukawa and J. Ponce, "Carved Visual Hulls for Image-Based Modeling," *Proc. European Conf. Computer Vision* (ECCV), Springer, 2006, pp. 564-577.
13. J. Starck and A. Hilton, "Model-Based Multiple View Reconstruction of People," *Proc. IEEE Int'l Conf. Computer Vision* (ICCV), IEEE CS Press, 2003, pp. 915-922.
14. A. Schodl and I. Essa, "Controlled Animation of Video Sprites," *Proc. ACM Symp. Computer Animation*, 2002, ACM Press, pp. 121-127.
15. J. Starck, G. Miller, and A. Hilton, "Video-Based Character Animation," *Proc. ACM Symp. Computer Animation*, 2005, ACM Press, pp. 49-58.

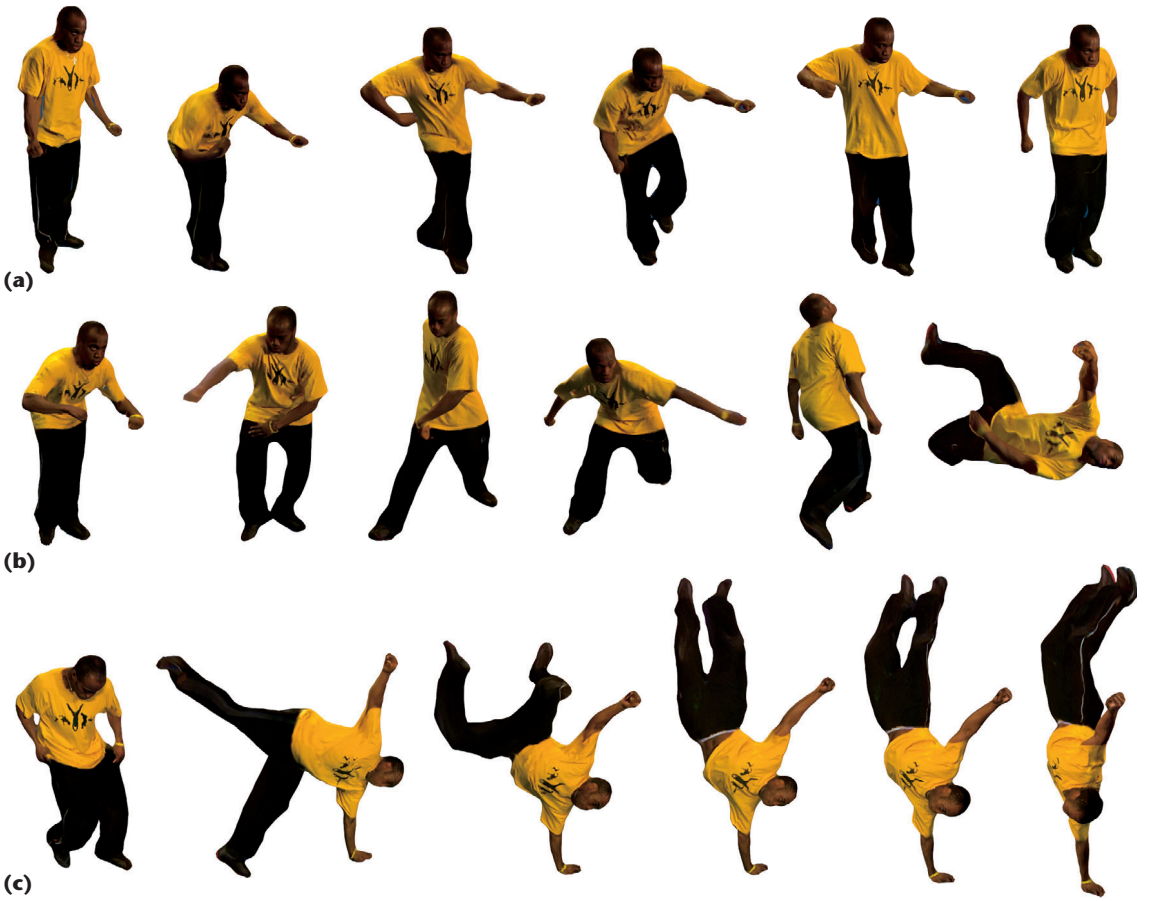
a character's flow of motion. Users control the character's state in this move tree at runtime. Figure 10 illustrates the character control concept through a graph of animation states in a move tree.

Surface capture records human performance as a temporal sequence of surface shape and appearance. We use our surface-capture system to record a performance library from an actor and construct a move tree for interactive character control. We ask the actor to perform a series of predefined motions (such as walking, jogging, and running) that form the building blocks for animation synthesis. As is typical in computer games, we manually construct the performance library by defining motion clips' start and end points. We then define transition points between motions. Animation representation using surface capture reproduces an actor's complete appearance rather than simply the skeletal motion, and recreates the detailed surface motion dynamics recorded in the original video images to produce a highly realistic digital performance.

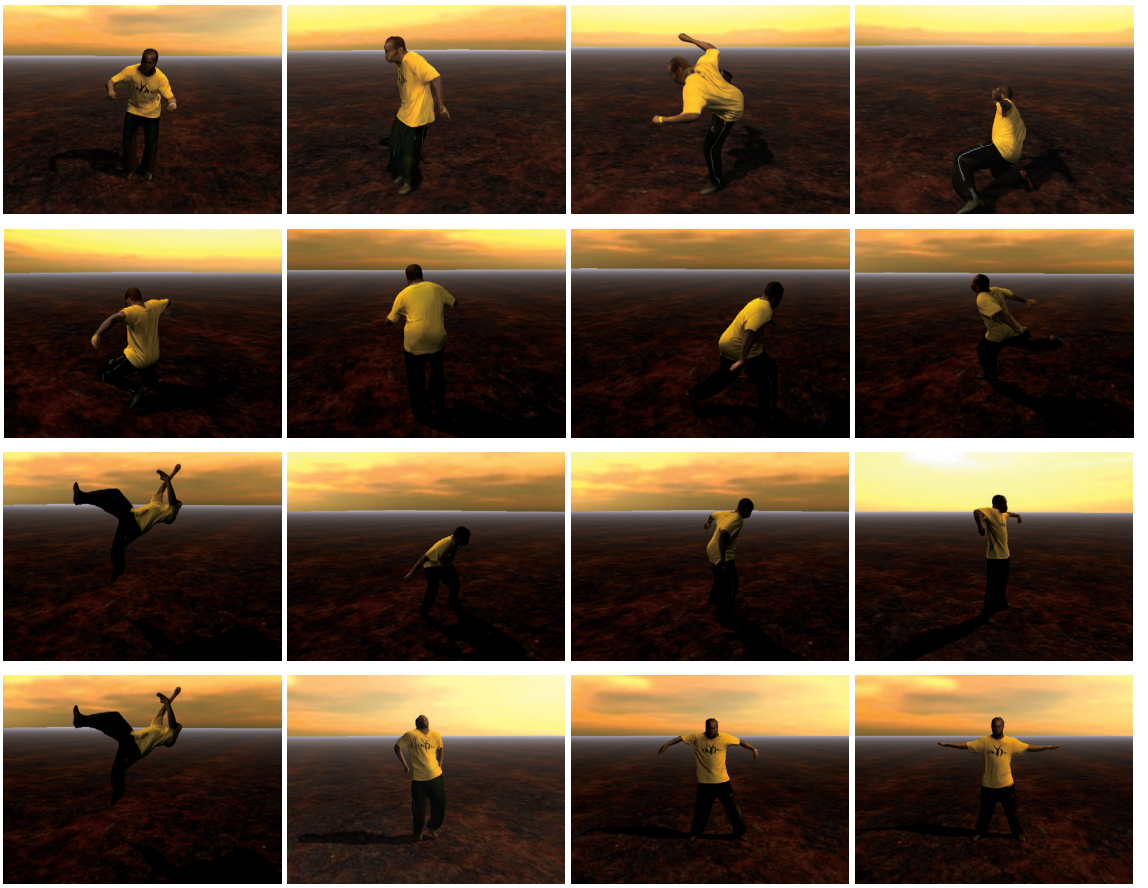
We show results for a street dancer performing a variety of movements while wearing loose-fitting clothing. The performance included body popping, kicks, jumps, handstands, and freeform moves. Figure 11 shows the captured surface motion sequences from views not aligned with the original camera views. These results



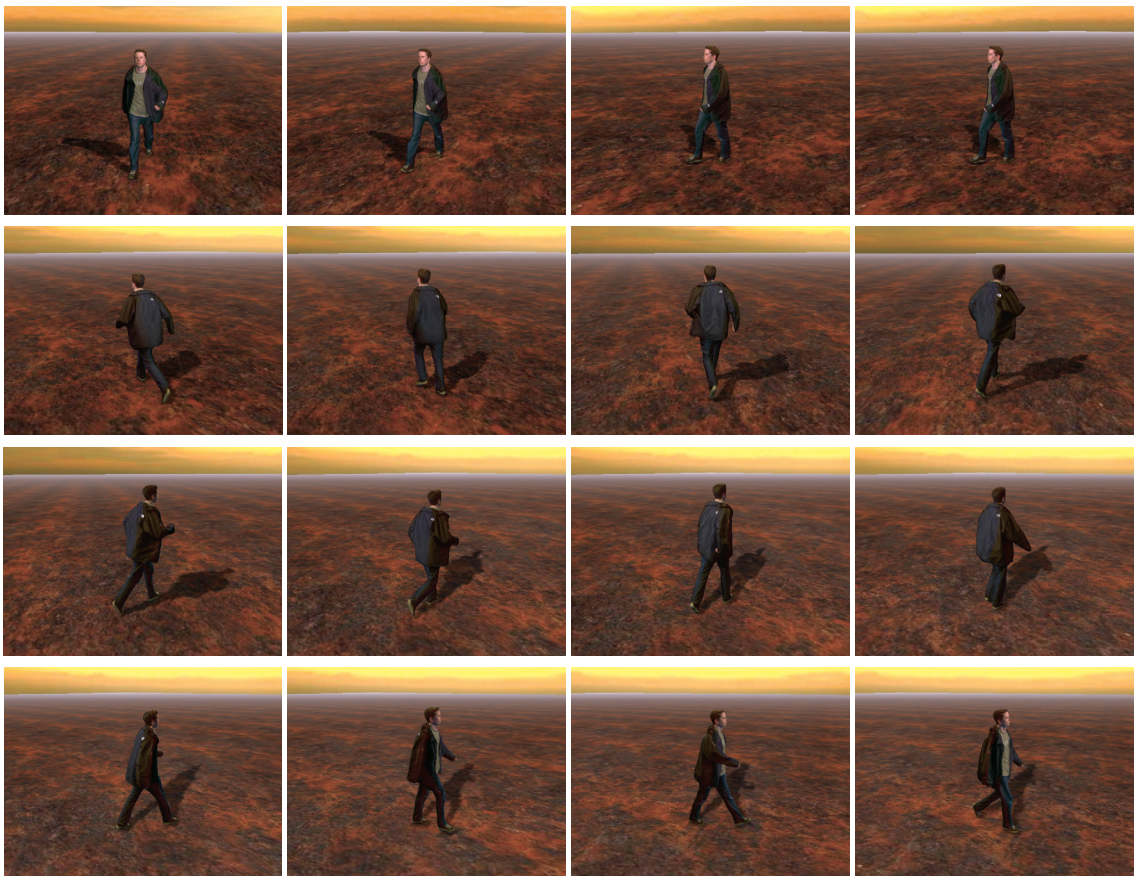
10 **Animation-by-example techniques transform (a) a performance library of captured motions into (b) a move tree that defines a digital character's feasible transitions. A user controls the character at runtime by controlling the character's state in the move tree.**



11 **Captured surface animation sequences: (a) lock, (b) kick, and (c) hand sequences.**



12 Transition between lock and pop using a kick sequence, rendered interactively from 360-degree views in the public domain Axiom games engine.



13 Interactive animation control with a character changing between walk and run motions. Synthesized views demonstrate the reproduction of the cloth motion recorded in the actor's coat.

demonstrate both that we can obtain a high visual quality and that the rendered sequences for novel views reproduce a life-like appearance for the digital character. Figure 12 shows results of rendering with interactive animation and viewpoint control in a public domain games engine (http://axiomengine.sourceforge.net).

Figure 13 shows animation control for a second character wearing a long coat. In these views, a user directs the transition between walking and running motions. This demonstrates the capture and reproduction of loose clothing for an actor in full wardrobe. The rendering performance for the representation achieved 300 frames per second on an Nvidia 6600GT graphics card with an uncompressed overhead of 1 Mbyte per frame. (Video sequences captured from the game engine are available at http://www.ee.surrey.ac.uk/cvssp/vmrg/surfcap). They demonstrate that surface motion capture reproduces the natural dynamics of loose clothing and achieves video-quality rendering in animation that is comparable in resolution and detail to conventional video.

Conclusion

Animation production is currently limited by the manual construction of a character motion tree and the transitions between animation states within this tree. Ideally, an actor would perform an arbitrary set of motions within a multiple camera studio. The system would automatically transform these captured surface sequences into a representation that lets users recreate any human motion from the performance library. This represents a challenging task for future work.

First, surfaces between arbitrary body poses must correspond so that the correspondence between motion segments can be determined to connect and then seamlessly blend segments. To date, only limited work has addressed whole-body correspondence and this work has only been applied to blending similar surface shapes.

Second, artistic control is required to direct and edit the content produced. Currently only high-level control can be achieved through the animation state of a character. An artist must be able to interactively manipulate the character animation at a finer level to synthesize new content that wasn't recorded in the original surface capture. ■

References

1. A. Hilton et al., "Whole-Body Modeling of People from Multi-view Images to Populate Virtual Worlds," *The Visual Computer: Int'l J. Computer Graphics*, vol. 16, no. 7, 2000, pp. 411-436.
2. G. Vogiatzis, P. Torr, and R. Cipolla, "Multi-view Stereo via Volumetric Graph-Cuts," *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (CVPR), vol. 2, IEEE CS Press, 2005, pp. 391-398.
3. J. Starck, G. Miller, and A. Hilton, "Volumetric Stereo with Silhouette and Feature Constraints," *Proc. British Machine Vision Conf.* (BMVC), vol. 3, British Machine Vision Assoc., 2006, pp. 1189-1198, 2006.
4. Y. Boykov and V. Kolmogorov, "Computing Geodesics and Minimal Surfaces via Graph Cuts," *Proc. IEEE Int'l Conf. Computer Vision* (ICCV), IEEE CS Press, 2003, pp. 26-33.
5. E. Praun and H. Hoppe, "Spherical Parameterization and Remeshing," *ACM Trans. Graphics*, 2003, pp. 340-349.
6. K. Zhou, H. Bao, and J. Shi, "3D Surface Filtering Using Spherical Harmonics," *Computer-Aided Design*, vol. 36, no. 4, 2004, pp. 363-375.
7. D. Steiner and A. Fischer, "Cutting 3D Freeform Objects with Genus-n into Single Boundary Surfaces Using Topological Graphs," *Proc. ACM Symp. Solid Modeling and Applications*, ACM Press, 2002, pp. 336-343.
8. P. Burt and E. Adelson, "A Multiresolution Spline with Application to Image Mosaics," *ACM Trans. Graphics*, vol. 2, no. 4, 1983, pp. 217-236.
9. L. Kovar, M. Gleicher, and F. Pighin, "Motion Graphs," *ACM Trans. Graphics*, 2002, pp. 473-482.

Jonathan Starck *is a senior research fellow in multiple view scene analysis and studio-based 3D content production at the University of Surrey. His research interests include image-based modeling for computer graphics and animation. He has a PhD in computer vision from Surrey University. Contact him at j.starck@surrey.ac.uk.*

Adrian Hilton *is a professor of computer vision and graphics at the University of Surrey. His research interest is robust computer vision to model and understand real world scenes for entertainment and communication. He is an area editor of Computer Vision and Image Understanding and a member of IEE, IEEE, and ACM. Contact him at a.hilton@surrey.ac.uk.*