Vision, Video, and Graphics (2005)
E. Trucco, M. Chantler (Editors)

3D Image Analysis and Synthesis at MPI Informatik

Christian Theobalt, Marcus A. Magnor, and Hans-Peter Seidel

Max-Planck-Institut für Informatik, 66123 Saarbrücken, Germany

Abstract

In the talk, we give a brief overview of the research done in the Computer Graphics Group and the Graphics-Optics-Vision Group of the Max-Planck-Institut für Informatik in the area of 3D Image Analysis and Synthesis. In this context, we address the whole pipeline ranging from the acquisition of computational scene models, over the algorithmic processing of these scene descriptions, to their photo-realistic rendition in the computer. This paper illustrates the questions that we are trying to answer by means of one of our research projects, video-based rendering. We have developed a model-based system to acquire, reconstruct and render free-viewpoint videos of human actors that nicely illustrates the concept of 3D Image Analysis and Synthesis.

1. Introduction

In computer graphics, it has always been the goal to generate virtual renditions of scenes in a computer that are indistinguishable from their real-world counterparts. To serve this purpose, computational models of virtual scenes have to be acquired, processed and passed on to a renderer which displays them. A general term which describes this pipeline is 3D Image Analysis and Synthesis (Fig. 1). In the Computer Graphics Group and the Graphics-Optics-Vision Group at the Max-Planck-Institut für Informatik, we research algorithmic solutions that enable us to solve individual subproblems within this pipeline.

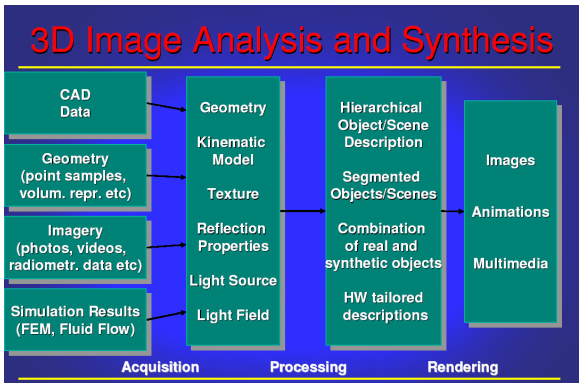


Figure 1: *The 3D Image Analysis and Synthesis pipeline.*

The first stage of the pipeline is to acquire a decent computational model representation of a scene. Among other things, this comprises the specification of the scene's geometry, the specification of texture and reflectance descriptions, as well as determining kinematic and dynamic properties. One option would be to design a model of a scene by hand "on the drawing table". However, it is often advantageous to look at the real thing and to extract the models by measuring them on objects in the real world. Geometry descriptions can be reconstructed by means of a laser-range scanner or by reconstructing them from multiple image or video data. Models of surface texture and light interaction can also be captured from image and video data if the incident illumination is controllable. A motion capture method can be employed to estimate kinematic models from multiple video streams of a moving scene.

Once a scene description has been acquired, it can be subject to further processing before it is rendered. For instance, this may involve the transformation of geometry descriptions into a form that is well-suited for rendition, the automatic extraction of features, or the construction of data structures that support efficient display.

Finally, once the scene description has been suitably preprocessed, it is available for display in a rendering system.

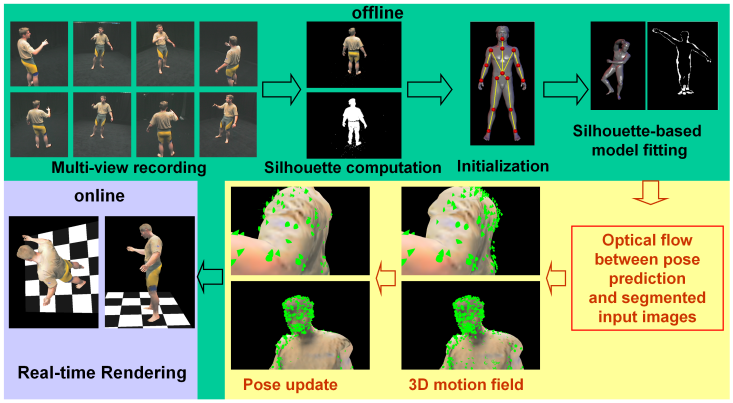


Figure 2: *Algorithmic workflow connecting the components of our model-based free-viewpoint video system. The optional steps of the texture-enhanced motion estimation scheme are shown with a yellow background.*

In the talk, many of our projects that implement the above pipeline are presented. In the remainder of this paper, we will present a video-based rendering system, which illustrates the algorithmic concepts that we employ to acquire, reconstruct and render real-world scenes from video data. It enables us to generate free-viewpoint videos of human actors from multiple synchronized video streams that can be rendered in real-time from arbitrary novel viewpoints.

2. Free-Viewpoint Video - a Realization of 3D Image Analysis and Synthesis

Free-viewpoint video is a 3D video approach in which a real-world scene is recorded with multiple imaging sensors, and a dynamic model of the scene is reconstructed that can be rendered from any arbitrary novel viewpoint. We have developed a model-based algorithm that enables us to reconstruct free-viewpoint videos of human actors from video footage [1], Fig. 2.

The inputs to our method are multiple frame-synchronized video streams of a moving person which we have recorded in our multi-view video studio (Sect. 4). For estimating the time-varying appearance of the actor in a scene we use an analysis-by-synthesis scheme that employs an adaptable human body model (Sect. 5). The principle clue that we use to fit the model to the scene content is the overlap between the image silhouettes and the silhouettes of the projected model in each camera view (Sect. 6). We transform this criterion into a numerical error function which is efficiently evaluated in graphics hardware. Using multiple camera views of the actor standing in an initialization pose, the geometry of the body model as well as as its skeleton dimensions are automatically customized. The shape parameters of the model remain fixed throughout the whole 3D video sequence. The central component of our analysis-by-synthesis scheme is a silhouette-based marker-free motion capture approach (Sect. 6.2). For each time step of video it performs an optimization search for an optimal set of pose parameters of the model. The energy function guiding this search is the previously mentioned silhouette-overlap. The hierarchical structure of the human body makes the pose determination problem a compartmentalized one, i.e. individual sub-problems can be solved independently from each other. We profit from this fact and exploit this parallelism in both silhouette-match computation and pose parameter search [17]. We have also developed an augmented motion capture approach that takes into account texture information and silhouette data simultaneously [16] (Sect. 6.3).

The renderer displays a free-viewpoint video by showing the model in the sequence of captured body poses. Realistic time-varying surface textures are generated by projectively texturing the model with the appropriately blended input camera video frames (Sect. 7). This way, dynamic scenes can be realistically rendered in real-time from arbitrary viewpoints on standard graphics hardware (Sect. 8).

3. Related Work

A variety of different approaches have been proposed in the literature that aim at transforming 2D video and television into an immersive 3D medium. Free-viewpoint video is one category of 3D video in which the viewer shall be given the flexibility to interactively position himself at an arbitrary virtual location in a 3D. But the term 3D video also comprises

other techniques, such as depth-image-based [4] or panoramic video [7].

The trail for free-viewpoint video applications was paved by algorithms from image-based rendering that aim at reconstructing novel renderings of a scene from input images [14]. These techniques have motivated a new research direction that draws from experience in computer vision and computer graphics to explicitly create 3D video systems. In depth-image-based approaches, novel viewpoints of a scene are reconstructed from color video and depth maps [5]. In [2] and [20] dynamic 3D scene geometry is reconstructed via stereo algorithms from multiple video cameras, and during playback the viewer can attain novel viewpoints in between the recording imaging sensors. It is also possible to use a view morphing method to generate novel views from reference images [13]. An approach for combined visual hull reconstruction and stereo-based mesh fitting is presented in [15]. In [9, 19] a shape-from silhouette method is applied to reconstruct dynamic scenes from multiple video streams. Applying light-field based methods for free-viewpoint video has also been considered [3, 6].

While 3D video provides interactivity only on the viewer's side, in 3D TV the full pipeline from acquisition to display needs to run in real-time. A 3D TV system for a restricted set of novel viewpoints based on multiple video cameras for recording and multiple projectors for display has been presented in [10].

We propose a model-based system for free-viewpoint videos of human actors that employs a marker-less motion capture approach to estimate motion parameters. A comprehensive review of computer vision based motion capture algorithms can be found in the respective survey papers [11].

4. Acquisition

The video sequences used as inputs to our system are recorded in our multi-view video studio [18]. IEEE1394 cameras are placed in a convergent setup around the center of the scene. The video sequences used in our experiments are recorded from 8 static viewing positions arranged at approximately equal angles and distances around the center of the room. All cameras are synchronized and record at a resolution of 320x240 pixels and a frame rate of 15 fps (maximum frame rate with external trigger). The cameras are calibrated into a global coordinate system. In each video frame, the silhouette of the person in the foreground is computed via background subtraction.

5. The Model

In our system we apply a generic human body model consisting of 16 individual body segments. Each segment's surface is represented via a closed triangle mesh. The model's kinematics are defined via 17 joints that connect the body segments and form a hierarchical skeleton structure. 35 pose parameters are needed to completely define the pose of the body. In total, more than 21000 triangles make up the human body model (Fig. 3b).

The generic model does not, in general, have the same proportions as its human counterpart. To be able to adapt model size and proportions to the recorded person, each segment can be individually scaled, and its surface deformed. While the pose parameters vary over time in a reconstructed free-viewpoint video, the anthropomorphic shape parameters remain fixed once they have been initialized prior to motion capture.

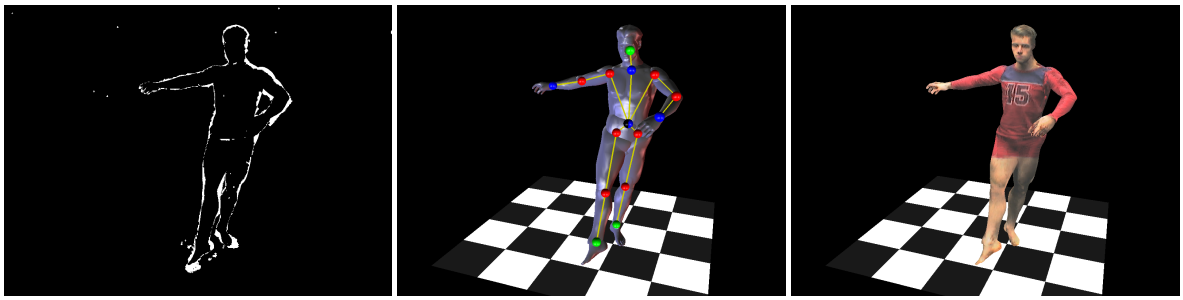
6. Silhouette Matching

The challenge in applying model-based analysis for free-viewpoint video reconstruction is to find a way how to automatically and robustly adapt the geometry model to the subject's appearance as it was recorded by the video cameras. Since the geometry model is suitably parameterized to alter, within anatomically plausible limits, its shape and pose, the problem consists of determining the parameter values that achieve the best match between the model and the video images. This task is regarded as an optimization problem. The subject's silhouettes, as seen from the different camera viewpoints, are used to match the model to the video images (an idea used in similar form in [8]): The model is rendered from all camera viewpoints, and the rendered images are thresholded to yield binary masks of the model's silhouettes. The rendered model silhouettes are then compared to the corresponding image silhouettes. As comparison measure, the number of silhouette pixels is determined that do not overlap. Conveniently, the exclusive-or (XOR) operation between the rendered model silhouette and the segmented video-image silhouette yields those pixels that are not overlapping. The sum of remaining pixels in all images is the mismatch score, with zero denoting an exact match.

This matching function can be evaluated very efficiently on graphics hardware (Fig. 3a). An Nvidia GeForce3^{TM} graphics card performs more than 100 of such matching function evaluations per second. Currently, the main limiting factor is the overhead generated by the read-back from the graphics board. To adapt model parameter values such that the mismatch score becomes minimal, a standard numerical optimization algorithm, such as Powell's method [12], runs on the CPU. The following subsections illustrate how we employ the silhouette-overlap criterion to initialize the body model and to determine its parameters of motion.

6.1. Initialization

To apply the silhouette-based model pose estimation algorithm to real-world multi-video footage, the generic geometry model must first be initialized, i.e. its proportions must be adapted to the subject in front of the cameras. To achieve this



(a) (b) (c)

Figure 3: *(a) Silhouette XOR; (b) body model; (c) textured body model from same camera view.*

we run a numerical minimization in the scaling parameter space of the model using the silhouette XOR energy function. The model provides one scaling and 16 deformation parameters per body segment that control the shape and proportions of the model. This way, all segment surfaces can be deformed until they closely match the actor's stature [1].

During model initialization, the actor stands still for a brief moment in a pre-defined pose to have his silhouettes recorded from all cameras. The generic model is rendered for this known initialization pose, and without user intervention, the model proportions are automatically adapted to the individual's silhouettes.

Obviously, an exact match between model outline and image silhouettes is not attainable since the parameterized model has far too few degrees of freedom. Thanks to advanced rendering techniques (Section 7) an exact match is neither needed, nor is it actually desired: Because the recorded person may wear relatively loose, flexible clothes, the silhouette outlines can be expected to be inaccurate, anyway. By not being dependent on exact image silhouette information, model-based motion analysis is capable of robustly handling also non-rigid object surfaces.

The initialization procedure takes only a few seconds, after which the segments' scaling parameter values and surface deformation values are known. These are kept fixed from now on. During motion capture, only the 35 joint parameters are optimized to follow the motion of the dancer.

6.2. Motion Capture

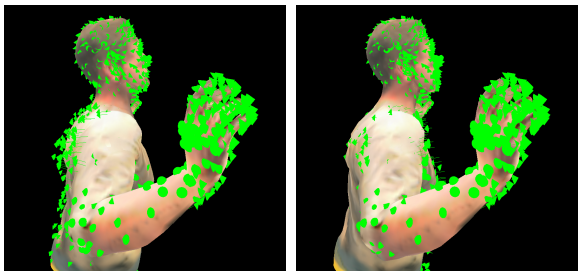
Since any form of visual markers in the scene would necessarily change its natural appearance, we developed a markerless human motion capture method to acquire free-viewpoint videos based on our a-priori model. In our method, the individualized geometry model automatically tracks the motion of a person by optimizing the 35 joint parameters for each time step. This is achieved by matching the projected body model to the segmented silhouette images of the person in each of the input camera views so that the model performs the same movements as the human in front of the cameras.

For numerical optimization of the pose parameters we employ a standard non-linear optimization method, such as Powell's method. The energy function employed is the previously described silhouette overlap criterion. To efficiently avoid local minima and to obtain reliable model pose parameter values, the parameters are not all optimized simultaneously. Instead, the model's hierarchical structure is exploited. Model parameter estimation is performed in descending order with respect to the individual segments' impact on silhouette appearance and their position along the model's kinematic chain. First, the position and orientation of the torso are varied to find its 3D location. Next the arms and legs are fitted using a joint parameterization for their lower and upper parts. Finally, the hands and the feet are regarded.

To avoid local, sub-optimal error minima for the arms and legs a limited regular grid search precedes the optimization search. This procedure accelerates convergence and effectively avoids local minima. Inter-penetrations between limbs are prevented by incorporating a collision check based on bounding boxes into the parameter estimation.

The motion parameters as well as the body deformation parameters are saved in our proprietary free-viewpoint video file format that serves as input for the real-time renderer.

The compartmentalized nature of the pose determination problem can be exploited to accelerate the motion capture process [17]. Firstly, the energy function evaluation itself can be sped up by appropriately constraining rendering window sizes, and thereby reducing the amount of data traveling between the graphics board and the system memory. Secondly, during XOR computation, unchanging body parts can be excluded from rendering which leads to even further increased evaluation speeds.



(a) (b)

Figure 4: *Body model with corrective motion field (green arrows) before (a) and after (b) pose update.*

Finally, we have implemented the silhouette-based tracking system as a distributed client-server application using 5 CPUs and GPUs which enables us to perform motion capture at near interactive frame rates.

6.3. Texture-enhanced Motion Capture

The accuracy at which body poses are captured directly influences the visual quality of the rendered free-viewpoint videos. If the model's geometry is not correctly aligned with the person in the real world, our texture generation algorithm (Sect. 7) projects input video frames onto incorrect geometry. This, in turn, leads to ghosting artifacts in the final renderings.

The silhouette-based motion capture approach faithfully captures even fast and complex body poses. However, slight inaccuracies in the measured poses may exist, which are mainly due to the limited image resolution and the lack of salient shape features on some body parts. The texture information which is available at no additional processing cost helps to correct these pose inaccuracies. We have thus designed a two-step predictor-corrector scheme that employs texture and silhouette data to infer the body pose at a single time step [16], Fig. 4: First, a set of pose parameters is computed by means of the original silhouette-based fitting method. In a second step, a corrective 3D motion field is computed by comparing the predicted model appearance to the real video footage. To this end, the model striking the silhouette-fitted pose is textured with the input video frames of the previous times step and rendered back into all camera views. Misalignments in the image planes of all cameras between the predicted appearance of the model and the measured appearance are identified via optical flow. From the multi-view optical flow field a 3D corrective motion field is reconstructed. From this, corrective pose update parameters are computed that bring the model into optimal multi-view silhouette- and color-consistency.

7. Rendering

A high-quality 3D geometry model is now available that closely matches the dynamic object in the scene over the entire length of the sequence. Our renderer displays the free-viewpoint video photo-realistically by rendering the model in the sequence of captured body poses and by projectively texturing the model with the segmented video frames. Time-varying cloth folds and creases, shadows and facial expressions are faithfully reproduced, lending a very natural, dynamic appearance to the rendered object (Fig. 5). To attain optimal rendering quality, the video textures need to be processed off-line prior to rendering: Since the final surface texture at each time step consists of multiple images taken from different viewpoints, the images need to be appropriately blended in order to appear as one consistent object surface texture. Also, local visibility must be taken into account, and any adverse effects due to inevitable small differences between model geometry and the true 3D object surface must be countered efficiently. For appropriate blending of the input camera views, per-vertex blending weights need to be computed and the visibility of each vertex in every input camera view needs to be determined. If surface reflectance can be assumed to be approximately Lambertian, view-dependent reflection effects play no significant role. Thus, the weights are computed independent of the output view in such a way that the camera seeing a vertex best gets the highest blending weight. This is achieved by assigning the reciprocal of the angle between the vertex normal and a camera's viewing direction as blending weight to each camera's texture fragment. An additional rescaling function is applied to these weights that allows for the flexible adjustment of the influence of the best camera on the final texture.

The 0/1-visibility of each vertex in each input camera view is precomputed and saved as part of the free-viewpoint video file. Since the silhouette outlines do not always exactly correspond to the projected model outlines in each camera view, we apply an extended visibility computation from a set of displaced camera views to avoid projection artifacts.

Finally, while too generously segmented video frames do not affect rendering quality, too small outlines can cause annoying untextured regions. To counter such artifacts, all image silhouettes are expanded by a couple of pixels prior to rendering.

During rendering, the color from each texture image is multiplied by its vertex-associated normalized blending weight and its 0/1-visibility in the programmable fragment stage of the graphics board. The final pixel color is the sum of the scaled texture colors.

Optionally, our renderer can also reproduce view-dependent appearance effects by means of view-dependent rescaling of the view-independent blending weights.

Figure 5: *Free-viewpoint video of a ballet dancer rendered into a virtual model of our acquisition room.*

8. Results and Conclusions

Our free-viewpoint video system can robustly reconstruct even as complex motion as expressive jazz dance (Fig. 6). Even slight details in time-varying surface appearance, such as wrinkles in clothing and facial expressions are faithfully captured in the dynamic surface textures. The precise motion data and the high-quality textures lend the rendered 3D videos a very natural appearance even from viewpoints that are very different from any input camera view (Fig. 5).

If pure silhouette-fitting is applied to determine pose parameters, average fitting times below 1 s for a single time step are feasible on a PC with a 1.8 GHZ CPU [1, 17]. If the motion-field enhanced pose estimation is applied, it takes between 10 and 45 s to fit the model, depending on what parameters have been chosen for the 2D optical flow algorithm [16]. The reconstructed free-viewpoint videos can be rendered at video rate even on a 1.8 GHz PC featuring a graphics board with a rather old Nvidia GeForce 3^{TM} GPU.

In the future, we plan to further extend our model-based free-viewpoint video approach. Firstly, we want to capitalize on the compact dynamic scene representation and investigate ways for efficient model-based 3D video encoding. Secondly, we intend to estimate time-varying surface reflectance in addition to the geometry in order to be able to realistically implant free-viewpoint videos into arbitrary novel environments.

In this paper, after elaborating on 3D Image Analysis and Synthesis in general, we have illustrated the concept by means of a specific example, namely a method to acquire, reconstruct and render free-viewpoint videos. We have shown that it is possible to generate realistic renditions of real-world scenes by looking at the real thing.

References

1. J. Carranza, C. Theobalt, M. Magnor, and H.-P. Seidel. Free-viewpoint video of human actors. In *Proc. of ACM SIGGRAPH*, pages 569–577, 2003. 2, 4, 6

2. K.M. Cheung, T. Kanade, J.-Y. Bouguet, and M. Holler. A real time system for robust 3D voxel reconstruction of human motions. In *Proc. of CVPR*, volume 2, pages 714–720, 2000. 3

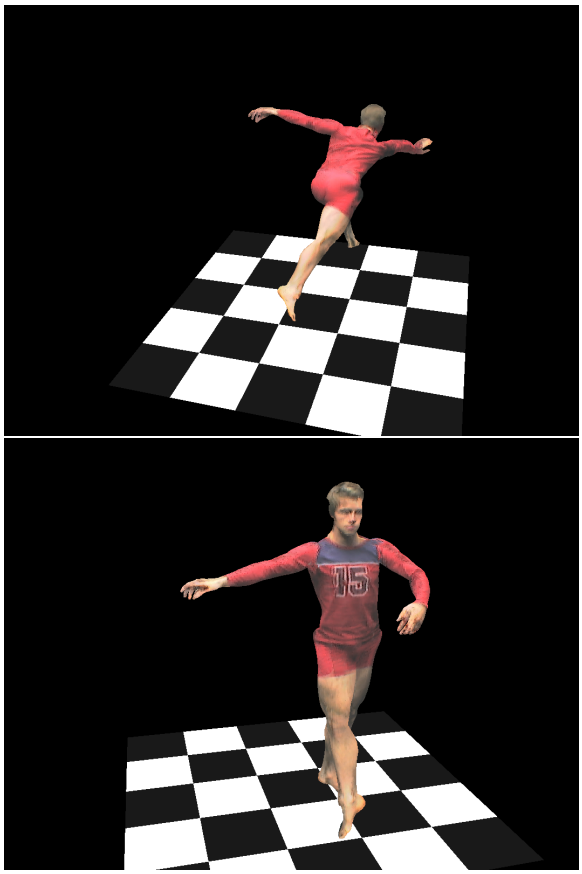


Figure 6: *Novel virtual views of a dancing performance.*

3. M. Droese, T. Fujii, and M. Tanimoto. Ray-space interpolation based on filtering in disparity domain. In *Proc. 3D Image Conference*, 2004. 3

4. C. Fehn. Depth-Image-Based Rendering (DIBR), Compression and Transmission for a New Approach on 3D-TV. In *Proceedings of SPIE Stereoscopic Displays and Virtual Reality Systems XI*, pages 93–104, San Jose, CA, USA, January 2004. 3

5. C. Fehn. Depth-image-based rendering (dibr), compression and transmission for a new approach on 3d-tv. In *Proc. Stereoscopic Displays and Applications*, page nn. to appear, 2004. 3

6. B. Goldlücke, M. Magnor, and B. Wilburn. Hardware-accelerated dynamic light field rendering. In *Proceedings Vision, Modeling and Visualization (VMV)*, pages 455–462, Erlangen, Germany, 2002. aka. 3

7. D. Kimber, J. Foote, and S. Lertsithichai. Flyabout: spatially indexed panoramic video. In *ACM Multimedia*, pages 339–347, 2001. 3

8. H. Lensch, W. Heidrich, and H. P. Seidel. A silhouette-based algorithm for texture registration and stitching. *Graphical Models*, 64(3):245–262, 2001. 3

9. T. Matsuyama and T. Takai. Generation, visualization, and editing of 3D video. In *Proc. of 3DPVT'02*, page 234ff, 2002. 3

10. W. Matusik and H.-P. Pfister. 3D TV: A scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. In *Proc. of ACM SIGGRAPH 2004*, pages 814–824, 2004. 3

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

12. W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes*. Cambridge University Press, 1992. 3

13. S.M. Seitz and C.R. Dyer. View morphing. In *Proc. of Siggraph96*, pages 21–30. ACM. 3

14. H.-Y. Shum and S.B. Kang. A review of image-based rendering techniques. In *Proc. of IEEE/SPIE VCIP*, pages 2–13, 2000. 3

15. 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. 3

16. C. Theobalt, J. Carranza, M. Magnor, and H.-P. Seidel. Enhancing silhouette-based human motion capture with 3D motion fields. In *Proc. Pacific Graphics*, pages 185–193. IEEE, 2003. 2, 5, 6

17. C. Theobalt, J. Carranza, M.A. Magnor, and H.-P. Seidel. A parallel framework for silhouette-based human motion capture. In *Proc. of VMV*, pages 207–214, 2003. 2, 4, 6

18. C. Theobalt, M. Li, M. Magnor, and Seidel. H.-P. A flexible and versatile studio for synchronized multi-view video recording. In *Proc. of Vision, Video and Graphics*, pages 9–16, 2003. 3

19. S. Wuermlin, E. Lamboray, O.G. Staadt, and M.H. Gross. 3D video recorder. In *Proc. of Pacific Graphics*, pages 325–334. IEEE, 2002. 3

20. C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High quality video view interpolation using a layered representation. In *Proc. of ACM SIGGRAPH*, pages 600–608, 2004. 3