VIDEO-BASED RENDERING

Richard Szeliski
Microsoft Research

Abstract

Image-based rendering is one of the hottest new areas in computer graphics. Instead of using 3D modeling and painting tools to construct graphics models by hand, IBR uses real-world imagery to rapidly create photorealistic shape and appearance models. However, IBR results to date have mostly been restricted to static objects and scenes. Video-based rendering brings the same kind of realism to computer animation, using video instead of still images as the source material. Examples of VBR include facial animation from sample videos, repetitive video textures that can be used to animate still scenes and photos, 3D environment walkthroughs built from panoramic video, and 3D video constructed from multiple synchronized cameras. In this paper, I survey a number of such systems developed by our group and by others, and suggest how this kind of approach has the potential to fundamentally transform the production (and consumption) of interactive visual media.

1 Introduction

Image-based rendering is a sub-field of computer graphics that uses multiple images (natural or synthetic) to achieve a high degree of realism and efficiency during rendering. Examples of image-based rendering systems include panoramic image mosaics [9, 50, 44], Concentric Mosaics [42], view interpolation and view morphing [8, 40], lightfields and Lumigraphs [27, 16], and Layered Depth Images and Sprites with Depth [41]. Central to each of these systems is the idea of taking two or more images and generating novel views using suitable combinations of image warping and blending. Some survey papers of this field include [22, 48, 43, 23, 49, 58].

More recently, researchers have started combining different kinds of photographs such as multiple exposures [15], flash and non-flash images [34], and photographs of the same people with different facial expressions [1]. Such applications of image-based rendering are now being called *computational photography* [26].

Video-based rendering is a newer area that focuses on using video sequences as inputs and generating novel videos as outputs. An early example of this work is Video Rewrite [6], in which various video clips of speaking person are manipulated and combined to make the person appear to say a novel utterance. Other early examples include the analysis of one or more videos of a person's head in order to create photorealistic facial animations [35, 5, 36]. The term *video-based rendering* itself was introduced in the Video Textures paper [39] and refers to techniques that extend ideas from image-based rendering to the temporal domain.

Of course, the idea of manipulating film and video to create novel effects (FX) is almost as old as the film industry itself. Techniques such as *blue screen matting* [45] have long been used to place actors in front of novel backgrounds, as well as to insert synthetic objects into live-action films. Other examples of interesting video processing operations include video stabilization [18], frame interpolation, image and video morphing [4], match move (camera tracking), wire removal, and of course, video compression.

More recently, the term Video-Based Rendering has been applied to the specific problem of generating novel viewpoint video from multiple video streams (http://www.video-based-rendering.org/) [29, 30]. In this survey paper, I take the more liberal view of the term and include topics such as video matting (§2), facial animation (§3), video textures (§4), 3D environment walkthroughs (§5), and interactive viewpoint video (§6).

2 Video Matting

Video matting refers to the process of extracting a foreground element from a film or video clip so that it can be inserted into another scene or composited in front of a novel object or character. Film-based *blue-screen matting* techniques have been used for decades in the film industry, but require the use of carefully controlled uniformly colored backgrounds [45]. More recently, Ruzon and Tomasi [38] introduced the idea of *natural image matting*, where a foreground matte (with fractional pixel opacities) can be extracted from a regular image once a rough outline of the object has been drawn. Chuang *et al.* [10] extended this technique using a Bayesian model of color variation and then later applied this to video sequences. In their Video Matting paper [11], they show how hand-drawn *trimaps* that delineate the foreground, background, and unknown regions can be propagated through time using optical flow techniques (Fig. 1). Background *clean plates* can also be constructed using global motion estimates and pixel stealing to improve the quality of the video mattes. In their subsequent Shadow Matting paper [12], Chuang *et al.* show how shadows from one video can be reconstructed and re-composited (properly draped over 3D objects) into another video clip.

More recently, a variety of increasingly sophisticated matting and rotoscoping techniques such as Poisson Matting [47] and

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

Figure 1: Video matting process [11]: (a) input video frame; (b) interpolated trimap (gray pixels are unknown); (c) computed alpha map (matte); (d) video composited over a novel background.

Figure 2: 3D face morphing example [35]: two samples faces ("neutral" and "joy") are interpolated both in shape and appearance to create a photorealistic looking animation.

Video Cutout [54] have been developed. In their most recent work (inspired by interactive segmentation approaches such as Grab Cut [37] and Lazy Snapping [28]), Wang and Cohen [53] show how the tedious process of drawing trimaps can be replaced by the quicker process of drawing strokes in the foreground and background regions. While these advances hold the promise of increasingly accurate and rapid creation of foreground mattes, the "holy grail" of fully automated parsing of scenes into foreground and background elements remains elusive [24].

3 Video-based facial animation

One of the earliest examples of video-based rendering in which new sequences were generated by automatically re-targetting captured video is Video Rewrite [6]. In this system, a library of video clips was analyzed using a speech recognizer to associate video frames with spoken phonemens. A new video could then be synthesized to match a novel spoken phrase (e.g., by a voice actor) by extracting the appropriate video frames in groups of three phonemes (*tri-phones*) and then compositing and blending the lip portions of the video into some reference video clip. The results were quite convincing and hard to distinguish from real video footage.

A more commonly used approach to computer generated facial animation is to use a 3D head model [33]. Unfortunately, purely synthetic models have a difficult time mimicking the lifelike appearance of real human skin with all of its subtle wrinkles and motions. For this reason, Pighin *et al.* [35] developed a system for modeling 3D head shapes and appearances from multiple photographs. The resulting texture-mapped 3D models could then be *morphed* in 3D (by interpolating and blending the shapes and appearances of different expressions) to create photorealsitic looking animations (Fig. 2).

To create the facial animations in [35], Pighin *et al.* manually specified keyframe expressions for different facial regions based on reference video footage. In subsequent work [36], they developed an off-line video-based tracking system that used analysis-by-synthesis to estimate the facial expression blend parameters for each input video frame. These parameters could then be used to animate a different character, perform a virtual camera move, modify the lighting, or apply virtual makeup to a character [36].

A more sophisticated system for modeling faces from photographs was developed by Blanz and Vetter [5]. The input to their system was a set of 3D laser-scanned models of different people, which were registered and then analyzed using principal component analysis (PCA) to create a "morphable model" that could be used to modify gender, age, appearance, and expression. The resulting system could then be used to create a photorealistic 3D model from just a single reference image.

Most recently, Zhang *et al.* [59] constructed a real-time facial

(a) Candle flame (b) Balloons (c) Watering Can (d) Fish Tank

Figure 3: Selected stills from generated video textures.

shape and appearance acquisition system using a collection of stereo cameras and active lighting. (See their paper for a nice review of 3D facial image-based modeling work.) Their system allows them to to construct frame-rate 3D range scans, fit these data to a 3D head model, and create performance-driven 3D facial animations with the most photorealistic appearance yet to date.

4 Video textures

Perhaps the purest form of video-based rendering is Video Textures [39], in which input video frames are contatenated together to create novel stochastic animations without the use of *any* 3D or geometric proxies. In its simplest form, a video texture is a random shuffling of the input video frames that appears to create an infinitely varying dynamic movie. Consider the problem of animating a continuous candle flame from just a short video clip (Fig. 3a). In order to sequence the frames in an order that appears natural, we first find short subsequences of video frames that have similar appearance. The midpoints of corresponding subsequences become *jump points* (show as red arcs in Fig. 3a) where the video texture player can jump forward or backward in time. In a sense, the video analysis part corresponds to discovering the hidden Markov chain underlying the video generation process.

When the video is more complicated, as in the Balloons sequence, it may not be possible to find two frames that match. In this case, the video can be decomposed into different regions separated by still pixels (shown using different colors in Fig. 3b), and each region can be analyzed and synthesized separately. Any remaining glitches that occur during playback can be reduced using image morphing techniques.

Video textures can also be used to shorten or lengthen a segment of video without changing the apparent speed of the content. Consider the watering can video shown in Fig. 3c. By turning the middle portion where the water is flowing continuously into a video texture, a shorter or longer portion of video can be generated by jumping forward or backward through the video clip.

Video textures can also be used as replacements for traditional computer generated animations. To demonstrate this, we analyzed a video of a fish swimming in a fish tank to extract an alpha-matted *sprite* whose position and velocity were recorded. To create a new animation using this fish (Fig. 3d), the next video frame is chosen so as to simultaneously match the desired keyframed velocity and the appearance of the previous frame while placing the sprite along the desired motion path. The resulting video-based animation can be used either to interact with a user as in a computer game or to follow a keyframed path. These kind of video-based animations may one day replace or augment traditional computer generated 3D animations because of their ease of acquisition and the resulting photorealism of the generated video.

An alternative approach to video texture generation is to analyze the video as a 3D shaped noise volume [51, 3, 46]. Soatto *et al.* call the resulting synthesized noise images *dynamic textures*, although in their original work, the response of their linear dynamic system (LDS) would decay over time. Wang and Zhu [55] extend the range of textured motions using ideas from computer graphics such as particle and wave motion and also show how video can be replaced with hand-drawn cartoon frames (a technique originally developed for facial animation [57, 7]).

4.1 Animating Pictures

Creating video-based animations from short video clips provides a compelling way to bring what might otherwise be static scenes (such as panoramas) to life [2]. An even more challenging problem is to animate a single picture or painting. Chuang *et al.* [13] attack this problem by first manually segmenting and matting the still image into a number of layers and inpainting [14] the background holes (Fig. 4). Each layer is then animated using a synthetic wind model, which ripples the water, bobs the boats, sways the trees, and moves the clouds. The resulting animated layers are then composited together to add a compelling sense of liveness to an otherwise static scene. While the current system requires a user to matte the layers and adjust their animation properties, a challenging topic for future research would be to design

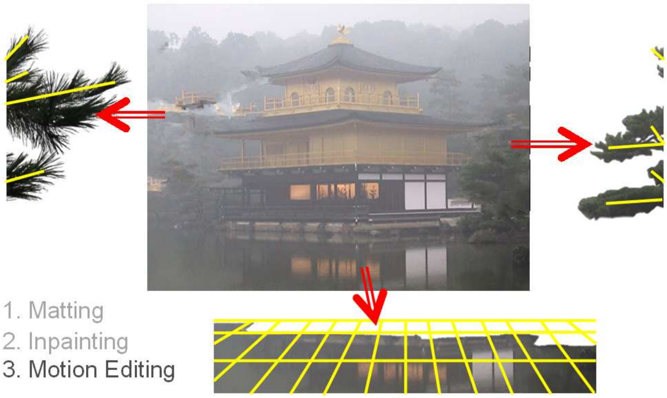
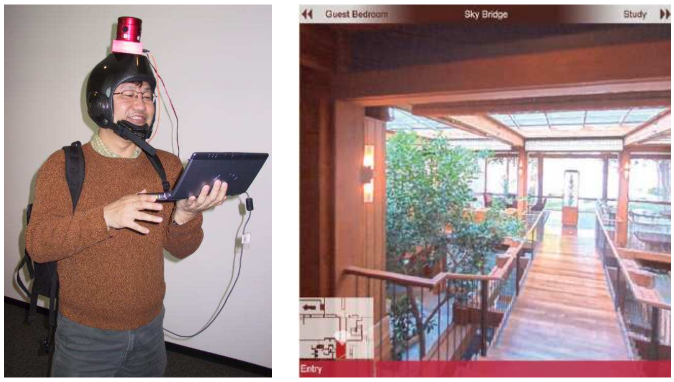


Figure 4: Animating a still photograph of the Kinkakuji Temple using layered motion models. The trees are animated to sway in the wind while the water is animated with a dynamic temporal ripple texture.



(a) (b)

Figure 5: Video-based walkthroughs: (a) helmet-mounted acquisition system, (b) screen shot of interactive tour showing a map in the lower left hand corner.

a recognition and learning-based approach to perform the animation automatically.

5 Video-based walkthroughs

The techniques we have described so far can be used to add a variety of special effects to video as well as to create new video-based animations. A more ambitious goal, however, would be to create a novel interactive visual medium based on video-based rendering ideas.

This desire to somehow bridge the gap between traditional interactive animation, as experienced in computer games, and traditional narrative storytelling, as seen in film and video, is what inspired the Image Based Reality project at Microsoft Research [52]. Creating a fully interactive and reactive 3D photorealistic environment as exemplified by the Star Trek Holodeck is, unfortunately, beyond the reach of today's technology.

Instead, as a first step, we decided to investigate whether we could provide a compelling sense of exploration in a fully photorealistic 3D environment based purely on acquired real-world imagery. In order to capture such an environment within a reasonable amount of time, we developed a novel $360°$ video camera that could easily be mounted on a helmet for rapid indoor and outdoor environment acquisition (Fig. 5a) [52].

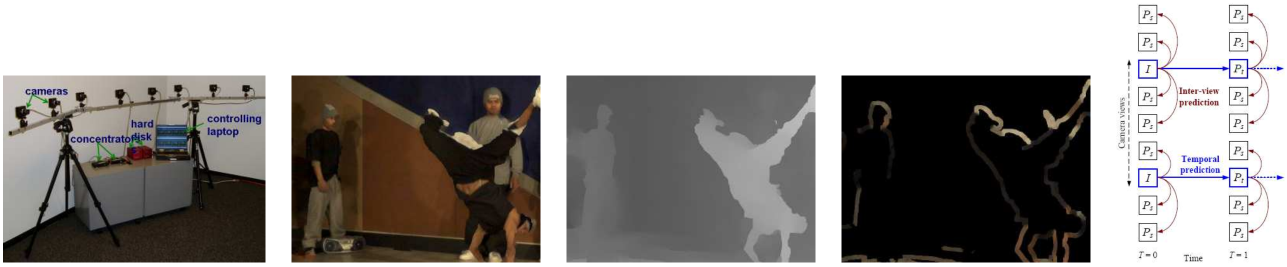
Once the video has been captured, it is then processed to stabilize and stitch the videos from the six cameras together. The locations of individual video frames are then correlated to a hand-drawn map of the environment using a process called *map correlation* [25]. Additional effects such as spatially localized sound or pop-up images and descriptions (hotspots) can then be added to the experience using an XML authoring tool. The entire experience can then be played back at interactive rates on a PC or laptop using a mouse or joystick to navigate in 3D (Fig. 5b) [52]. While the user is constrained to move on a "rail" that follows the original camera motion (with potential branching at path crossings), most users do not mind this restriction since it makes the navigation paradigm simple and intuitive to use. Staying along the path of the original video allows us to avoid doing any *view interpolation*, which can degrade the quality of the visual experience when a sufficiently accurate 3D proxy geometry is not available for the scene.

Our experience with the system to date has been that it provides a very compelling impression of 3D navigation in a real-world environment. However, since we do not at present compute any 3D geometry, we cannot as yet mix this video-based experience with additional 3D graphics such as might be required for home remodeling or computer gaming. Building such a system would be allow us to extend our purely video-based walkthrough experiences to more interactive 3D environments.

6 Interactive 3D Video

While our video-based walkthroughs provide a compelling sense of 3D space, they do not, unfortunately, retain any of the dynamic aspects of live video (with the possible exception of video textures that can be mapped into the environment). This is because we trade off space vs. time along the moving camera path, i.e., we create a compelling illusion of continuous motion by mapping the panoramic video frames to different locations. Capturing a large 3D environment with continuous live video would require too many cameras to be practical using today's technology.

However, there is another compelling application of video-based rendering that *does* support the acquisition and playback of live video. This concept was first pioneered by Takeo Kanade's Virtualized Reality project at Carnegie-Mellon University [20, 21] and has since been emulated and extended



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

Figure 6: Video View Interpolation: (a) capture rig consisting of 8 cameras; (b) recovered background color image; (c) recovered depth map; (d) recovered boundary image (color channel); (e) multi-viewpoint video compression scheme.

by a large number of groups around the world [32, 29, 30]. (See http://www.video-based-rendering.org/ for links to current work as well as other paper in these proceedings [17].)

In this restricted setup, cameras are focused on a compact *working volume* of space within which the live action can occur. Both dome-like [21, 32, 29] and grid-like [31, 56] arrangements of cameras have been used. In our own work [60], we used a linear array of cameras arranged in an arc to capture dance performances involving several dancers (Fig. 6a).

While many systems for 3D TV and video view interpolation have been built, our particular emphasis is on generating the highest possible interpolation quality at real-time rendering rates, as well as the efficient compression of the multi-view video streams (Fig. 6e) [60]. To achieve these goal, we designed a novel segmentation-based stereo matching algorithm specially tailored to accurately locate depth discontinuities while also preserving depth coherence in untextured background areas (Fig. 6c). Because assigning a single per-pixel depth can lead to aliasing problems during view interpolation, we also devised a two-layer (boundary) model for the video stream, where a thin boundary strip with per-pixel alpha (opacity) values is computed along the boundary of each depth discontinuity (Fig. 6d). The computed depth maps are used to spatially predict in-between camera streams from reference streams (Fig. 6e). Finally, a novel rendering algorithm using custom GPU pixel shading code was developed to interpolate and blend the two layers from each of the two neighboring views (streams) to synthesize a continuously varying viewpoint video in real time under interactive user control.

We believe that the building blocks developed in our research will become essential components of next-generation 3D TV authoring, compression, and rendering systems, as the systematic exploitation of sub-pixel accurate scene geometry is essential for a wide variety of photorealistic 3D TV applications such as z-keying [19], virtual (free) viewpoint interpolation, and efficient prediction for compression.

7 Conclusions

Video-based rendering has the potential to revolutionize the process of visual media production. Capturing large amounts of video, whether through the use of multiple video cameras, video-based exploration of 3D spaces, or video libraries suitable for video-based animation will be the first step. Combining these videos using advanced computer vision and rendering algorithms will then enable the creation of compelling new interactive user experiences.

The basic building blocks sketched in this survey, namely video matting, 3D video-based object and environment modeling, video textures, and high-quality video view interpolation will all play essential roles in this next generation of authoring and playback systems. The resulting hybrid experiences will allow users (and directors) to produce and explore novel combinations of real-world and computer generated imagery in manners that have not yet been envisioned.

References

[1] A. Agarwala et al. Interactive digital photomontage. *ACM Transactions on Graphics*, 23(3):292–300, August 2004.

[2] A. Agarwala et al. Panoramic video textures. *ACM Transactions on Graphics*, 24(3):821–827, August 2005.

[3] Z. Bar-Joseph et al. Texture mixing and texture movie synthesis using statistical learning. *IEEE Transactions on Visualization and Computer Graphics*, 7(2):120–135, April-June 2001.

[4] T. Beier and S. Neely. Feature-based image metamorphosis. *Computer Graphics (SIGGRAPH'92)*, 26(2):35–42, July 1992.

[5] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. *Proceedings of SIGGRAPH 99*, pages 187–194, August 1999. ISBN 0-20148-560-5. Held in Los Angeles, California.

[6] C. Bregler, M. Covell, and M. Slaney. Video rewrite: Driving visual speech with audio. *Computer Graphics (SIGGRAPH'97)*, pages 353–360, August 1997.

[7] I. Buck et al. Performance-driven hand-drawn animation. In *Symposium on Non Photorealistic Animation and Rendering*, pages 101–108, Annecy, June 2000. ACM SIGGRAPH.

[8] S. Chen and L. Williams. View interpolation for image synthesis. *Computer Graphics (SIGGRAPH'93)*, pages 279–288, August 1993.

[9] S. E. Chen. QuickTime VR – an image-based approach to virtual environment navigation. *Computer Graphics (SIGGRAPH'95)*, pages 29–38, August 1995.

[10] Y.-Y. Chuang et al. A Bayesian approach to digital matting. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2001)*, volume II, pages 264–271, Kauai, Hawaii, December 2001.

[11] Y.-Y. Chuang et al. Video matting of complex scenes. *ACM Transactions on Graphics*, 21(3):243–248, July 2002.

[12] Y.-Y. Chuang et al. Shadow matting. *ACM Transactions on Graphics*, 22(3):494–500, July 2003.

[13] Y.-Y. Chuang et al. Animating pictures with stochastic motion textures. *ACM Transactions on Graphics*, 24(3):853–860, August 2005.

[14] A. Criminisi, P. Pérez, and K. Toyama. Object removal by exemplar-based inpainting. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2003)*, volume II, pages 721–728, Madison, WI, June 2003.

[15] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. *Proceedings of SIGGRAPH 97*, pages 369–378, August 1997. ISBN 0-89791-896-7. Held in Los Angeles, California.

[16] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The Lumigraph. In *Computer Graphics Proceedings, Annual Conference Series*, pages 43–54, Proc. SIGGRAPH'96 (New Orleans), August 1996. ACM SIGGRAPH.

[17] O. Grau et al., editors. *Second European Conference on Visual Media Production*, The IEE Savoy Place, London, UK, November 2005.

[18] M. Hansen, P. Anandan, K. Dana, G. van der Wal, and P. Burt. Real-time scene stabilization and mosaic construction. In *IEEE Workshop on Applications of Computer Vision (WACV'94)*, pages 54–62, Sarasota, December 1994. IEEE Computer Society.

[19] T. Kanade et al. A stereo machine for video-rate dense depth mapping and its new applications. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'96)*, pages 196–202, San Francisco, June 1996.

[20] T. Kanade, P. J. Narayanan, and P. W. Rander. Virtualized reality: Concepts and early results. In *IEEE Workshop on Representations of Visual Scenes*, pages 69–76, Cambridge, Massachusetts, June 1995.

[21] T. Kanade, P. W. Rander, and P. J. Narayanan. Virtualized reality: constructing virtual worlds from real scenes. *IEEE MultiMedia Magazine*, 1(1):34–47, Jan-March 1997.

[22] S. B Kang. A survey of image-based rendering techniques. Technical Report 97/4, Digital Equipment Corporation, Cambridge Research Lab, August 1997.

[23] S. B. Kang, R. Szeliski, and P. Anandan. The geometry-image representation tradeoff for rendering. In *International Conference on Image Processing (ICIP-2000)*, volume II, pages 13–16, Vancouver, September 2000.

[24] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Learning layered motion segmentations of video. In *Tenth International Conference on Computer Vision (ICCV 2005)*, Beijing, China, October 2005.

[25] A. Levin and R. Szeliski. Visual odometry and map correlation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2004)*, volume I, pages 611–618, Washington, DC, June 2004.

[26] M. Levoy, F. Durand, and R. Szeliski, editors. *Symposium on Computation Photography and Video*, May 2005. http://photo.csail.mit.edu/.

[27] M. Levoy and P. Hanrahan. Light field rendering. In *Computer Graphics Proceedings, Annual Conference Series*, pages 31–42, Proc. SIGGRAPH'96 (New Orleans), August 1996. ACM SIGGRAPH.

[28] Y. Li et al. Lazy snapping. *ACM Transactions on Graphics*, 23(3):303–308, August 2004.

[29] M. Magnor. *Video-Based Rendering*. A. K. Peters, 2005.

[30] M. Magnor et al., editors. *Video-based Rendering*, SIGGRAPH 2005 Course 16, August 2005.

[31] W. Matusik and H. Pfister. 3d tv: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. *ACM Transactions on Graphics*, 23(3):814–824, August 2004.

[32] S. Moezzi et al. Reality modeling and visualization from multiple video sequences. *IEEE Computer Graphics and Applications*, 16(6):58–63, November 1996.

[33] F. I. Parke and K. Waters. *Computer Facial Animation*. A K Peters, Wellesley, Massachusetts, 1996.

[34] G. Petschnigg et al. Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics*, 23(3):664–672, August 2004.

[35] F. Pighin, J. Hecker, D. Lischinski, D. H. Salesin, and R. Szeliski. Synthesizing realistic facial expressions from photographs. In *Computer Graphics (SIGGRAPH'98 Proceedings)*, pages 75–84, Orlando, July 1998. ACM SIGGRAPH.

[36] F. Pighin, D. H. Salesin, and R. Szeliski. Resynthesizing facial animation through 3D model-based tracking. In *Seventh International Conference on Computer Vision (ICCV'99)*, pages 143–150, Kerkyra, Greece, September 1999.

[37] C. Rother, V. Kolmogorov, and A. Blake. "GrabCut" - interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, August 2004.

[38] M. A. Ruzon and C. Tomasi. Alpha estimation in natural images. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'2000)*, volume 1, pages 18–25, Hilton Head Island, June 2000.

[39] A. Schödl, R. Szeliski, D. H. Salesin, and I. Essa. Video textures. In *Computer Graphics (SIGGRAPH'2000 Proceedings)*, pages 489–498, New Orleans, July 2000. ACM SIGGRAPH.

[40] S. M. Seitz and C. M. Dyer. View morphing. In *Computer Graphics Proceedings, Annual Conference Series*, pages 21–30, Proc. SIGGRAPH'96 (New Orleans), August 1996. ACM SIGGRAPH.

[41] J. Shade, S. Gortler, L.-W. He, and R. Szeliski. Layered depth images. In *Computer Graphics (SIGGRAPH'98 Proceedings)*, pages 231–242, Orlando, July 1998. ACM SIGGRAPH.

[42] H.-Y. Shum and L.-W. He. Rendering with concentric mosaics. In *SIGGRAPH'99*, pages 299–306, Los Angeles, August 1999. ACM SIGGRAPH.

[43] H.-Y.. Shum and S. B. Kang. A review of image-based rendering techniques. In *SPIE Visual Communication and Image Processing (VCIP 2000)*, pages 2–13, Perth, Australia, June 2003.

[44] H.-Y. Shum and R. Szeliski. Construction of panoramic mosaics with global and local alignment. *International Journal of Computer Vision*, 36(2):101–130, February 2000. Erratum published July 2002, 48(2):151-152.

[45] A. R. Smith and J. F. Blinn. Blue screen matting. In *Computer Graphics Proceedings, Annual Conference Series*, pages 259–268, Proc. SIGGRAPH'96 (New Orleans), August 1996. ACM SIGGRAPH.

[46] S. Soatto, G. Doretto, and Y. N. Wu. Dynamic textures. In *Eighth International Conference on Computer Vision (ICCV 2001)*, volume II, pages 439–446, Vancouver, Canada, July 2001.

[47] J. Sun et al. Poisson matting. *ACM Transactions on Graphics*, 23(3):315–321, August 2004.

[48] R. Szeliski. Stereo algorithms and representations for image-based rendering. In *British Machine Vision Conference (BMVC'99)*, volume 2, pages 314–328, Nottingham, England, September 1999.

[49] R. Szeliski. Image-based modeling and rendering. In *Fourth International Workshop on Cooperative and Distributed Vision*, pages 413–429, Kyoto, Japan, March 2001.

[50] R. Szeliski and H.-Y. Shum. Creating full view panoramic image mosaics and texture-mapped models. *Computer Graphics (SIGGRAPH'97 Proceedings)*, pages 251–258, August 1997.

[51] M. Szummer and R. W. Picard. Temporal texture modeling. In *IEEE International Conference on Image Processing (ICIP-96)*, volume 3, pages 823–826, Lausanne, 1996.

[52] M. Uyttendaele et al. Image-based interactive exploration of real-world environments. *IEEE Computer Graphics and Applications*, 24(3), May/June 2004.

[53] J. Wang and M. F. Cohen. An iterative optimization approach for unified image segmentation and matting. In *Tenth International Conference on Computer Vision (ICCV 2005)*, Beijing, China, October 2005.

[54] J. Wang et al. Video cutout. *ACM Transactions on Graphics*, 24(3):585–594, August 2005.

[55] Y. Wang and S.-C. Zhu. Modeling textured motion: Particle, wave and sketch. In *Ninth International Conference on Computer Vision (ICCV 2003)*, pages 213–220, Nice, France, October 2003.

[56] B. Wilburn et al. High performance imaging using large camera arrays. *ACM Transactions on Graphics*, 24(3):765–776, August 2005.

[57] L. Williams. Performace driven facial animation. *Computer Graphics*, 24(4):235–242, 1990.

[58] C. Zhang and T. Chen. A survey on image-based rendering - representation, sampling and compression. *EURASIP Signal Processing: Image Communication*, 19(1):1–28, January 2004.

[59] Li Zhang, Noah Snavely, Brian Curless, and Steven M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *ACM Transactions on Graphics*, 23(3):548–558, August 2004.

[60] C. L. Zitnick et al. High-quality video view interpolation using a layered representation. *ACM Transactions on Graphics*, 23(3):600–608, August 2004.