Historical Perspectives on 4D Virtualized Reality

Takeo Kanade* and P. J. Narayanan†
*Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213. U. S. A.

†Centre for Visual Information Technology
International Institute of Information Technology
Gachibowli, Hyderabad 500032. India

Abstract

Recording dynamic events, such as a sports event, a ballet performance, or a lecture, digitally for experiencing in a spatiotemporally distant setting requires 4D capture: three dimensions for their geometry/appearance over the fourth dimension of time. Cameras are suitable for this task as they are non-intrusive, universal, and inexpensive. Computer Vision techniques have advanced sufficiently to make the 4D capture possible. In this paper, we present a historical perspective on the Virtualized Reality^{TM} system developed since early 90s to early 2000 at CMU for the 4D capture of dynamic events.

1 Introduction

Dynamic events in entertainment or education are of immense interest to human beings. In traditional media, each viewer absorbs the event from a fixed viewpoint determined by the position of seating. Can we record a dynamic event digitally and experience it in a more spatiotemporally free manner? A game could be viewed from any viewpoint of a particular player or even of the ball, or a ballet could be viewed from a virtual seat in the middle of the stage. We may also want to edit or modify the recorded event in creative ways, such as combining two events into a single one. The medium of digitized dynamic events can open up new vistas in immersive and participative entertainment, empowering the viewers to control their experience.. The use of such a medium need not be limited to entertainment; it can enhance the training experience in surgery so that a particularly difficult step can be revisited repeatedly, selecting a suitable vantage point each time.

Tele-presence is the ability to be present at an event taking place at a distance. If we can digitize a dynamic event and let a user immerse into it, we would achieve tele-experience (if live) or post-experience (if delayed), navigating through and interacting with the digitized event. The ability to experience a remote dynamic event in its richness, including the ability to navigate through the event unhindered, brings the world to us and can be argued to be the functional equivalent of tele-portation [26].

In early early 90s we began to develop multi-camera computer vision technologies to capture large, dynamic events in terms of their geometric and appearance aspects. Our system consists of a large number of cameras to capture the event inside a room from all directions. It produces the 4D event description, consisting of the 3D model of the scene together with its appearance across time. Tools similar to those used by virtual reality are used to experience the digitized event, either in real time or at a later time. We coined the term, *Virtualized Reality*^{TM} , to emphasize the aspect of converting real events to virtual ones. This paper presents the details of the Virtualized Reality System from a historical perspective.

The precursor to the Virtualized Reality project was the development of multi-camera multi-baseline video-rate stereo machine in the beginning of 90s. By 1993, we built a series of machines that could convert the input scene to a 256×256 8-bit depth map at the speed of 30 frames per second [21, 10, 11]. With such a machine, we could demonstrate the *z-key* technique by which a real scene and a virtual scene were merged by using the distance per pixel, instead of blue-key, for switching between the two [13]. Having realized that a dynamic scene could be digitized as a whole if it was observed by multiple cameras from multiple directions, we first built in mid-1994 the first Virtualized Reality system with 10 cameras from two directions, and then expanded to a 51-camera dome system in late

Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)
0-7695-2646-2/06 $20.00 © 2006 IEEE

1995 that could capture an event inside from a complete hemisphere. These earlier systems were analog and offline; videos were synced and recorded on video tapes with time codes and digitized later for processing. The system was upgraded into a 49-camera digital room in 1998 and into the current facility of a 48-camera large space in 2002, where all of the capturing is done in a complete digital and on-line manner.

Capturing geometric structures of an object or a small space and showing it as a textured model has been a standard practice in Computer Vision. The Virtualized Reality system, however, was one of the first to capture a large dynamic event with a large number of cameras and to turn them into a space-time representation with the intent of experiencing it later [12, 20]. Several efforts with similar goals appeared as well. Use of such technologies for virtual space teleconferencing was suggested by Fuchs et al [7]. The virtualized reality project of NRC, Canada broadly had the same goals and used a variety of modelling and rendering techniques to virtualize buildings, heritage sites, and mines [8]. The Multiple Perspective Interactive Video project [18] used a combination of static models, change detection, and shape from triangulation, to model and navigate through large spaces. Image-Based Rendering (IBR) techniques for capturing objects from multiple viewpoints and generating novel viewpoints [1, 6, 9, 15, 28, 34] became a topic of intensive study, initially for individual static objects, and later for dynamic scenes. The Digital Michelangelo project used high-quality range finders and aimed at archiving and preservation of cultural heritage [16].. Recently, it seems that there is a revival of interest in capturing dynamic events. The 3D Video recorder [33], the free viewpoint video [2], and video-based rendering [36] appeared in the last few years.

We analyze the requirements of an ideal dynamic event capture system in the next section. The design of our system and the reasons behind the choices made are presented in four subsequent sections. We also discuss the current status of the technology related to the system wherever relevant.

2 Event Digitization: Requirements

An ideal dynamic event digitization system should be usable for a variety of events. The essential features and requirements of a system to digitize such large dynamic events are summarized below.

Scalability: The event capture scheme should be scalable to large spaces. Scalability should also apply to parameters like visual quality, resolution, and model fidelity.

Non-intrusiveness: The digitizing should be non-intrusive and should merely record the goings-on without modifying it in any way. Thus, passive capture mechanisms should be used.

Naturalness: We will have no control over the event or the ability to change aspects of it for the ease of digitizing. The lighting, the staging, and the sequencing should be natural and cannot be modified to suit the digitizing in the typical situation.

High fidelity: The experience of the digitized event should be comparable to the real one. This applies to all aspects of the event including shape, appearance, motion, etc.

Unhindered immersion: The digitized dynamic event should be viewable immersively from any viewpoint in the event space with few forbidden areas.

Tele- or post- experience: A tele-experience system allows exploration of the event live in real-time. Each viewer can experience a basketball match from the point of view of a specific player as it happens. A post-experience system allows exploration of the event at a later time and can afford expensive preprocessing of the captured content. A tele-experience system is a dream today, but a post-experience system also has many applications.

Active participation: A digitized event can be experienced actively. Full navigation of the event space and time should be available at view-time. It may also be possible to modify the event appearance or content after capture, making the viewer a participant.

The above points present a dream for a dynamic event digitizing system. To be practical, the system should also be economical. It is desirable to realize the system with standard components available in the market. This reduces the costs and gives us the ability to exploit the advances in technology. A trade-off exists between the cost and the capabilities. A tele-experience system with live digitizing and playback will require computing power that is several orders of magnitude higher than a post-experience system. It will also be convenient if the event digitization system computes virtual environment similar to those used in VR. These consist of geometry suitable for graphics rendering and appearance coded as surface properties and texture maps.

The Virtualized Reality system has been designed with many of the above requirements in mind, though some of the aspects have been simplified due to technical limitations or feasibility. We discuss four aspects of the dynamic event digitization and our approach to those in the following sections: Capture, Modelling and Representation, Experiencing, and Manipulation.

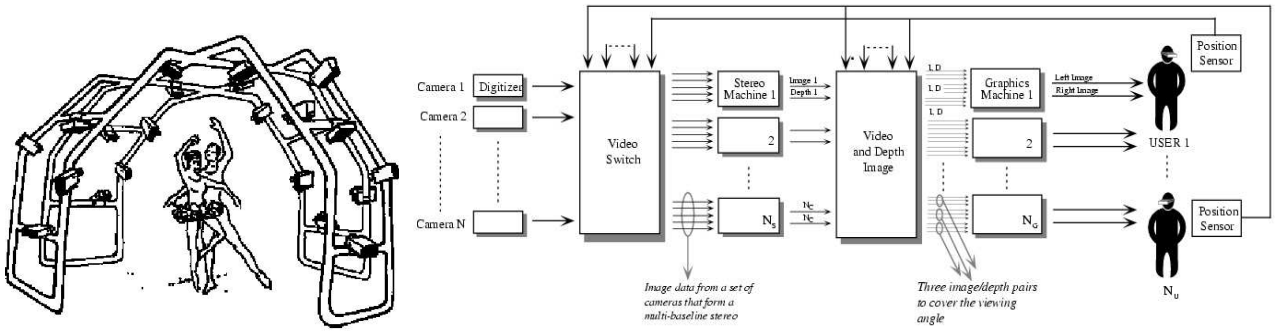

Figure 1: Concept and the block diagram of the Virtualized Reality system [12]

3 Non-Intrusive Event Capture

Figure 1 gives the conceptual block diagram of the Virtualized Reality system. The process starts with the capture of the dynamic event using multiple cameras placed at different points in the event space. We need to capture the geometric aspects of the events as well as its appearance. Range finders using lasers or light stripes can capture the geometric structure directly, by shining a laser beam or a structured pattern of light on the scene. This is intrusive as the event is changed visually for the actors and the viewers. They also do not extend to dynamic events. Lightstripe range finders and other active lighting based systems also suffer from the same problems. True non-intrusive capture of the shape and appearance of a dynamic event is possible with video cameras. Cameras capture the appearance directly in the form of images. Computer Vision techniques applied to these images can recover the geometric structure of the scene. Cameras are preferred capturing devices due to their non-intrusiveness, speed, economy, familiarity, and universality. Multiple cameras are required since no single camera can provide the view of a complete event. Multiple cameras are also required for structure recovery using stereo algorithms.

3.1 Event Capture System Setup

The first Virtualized Reality setup built in 1995, called the *3D Dome*, used 51 cameras mounted on a geodesic dome, 5 meters in diameter. It used industrial grade NTSC, monochrome, analog CCD video cameras for image capture. Lenses with 3.6mm focal length were used for a field of view close to 90 degrees. The cameras were arranged to provide all around views of the event and were sufficiently close for the computer vision algorithms to work well. The cameras looked at the center of the dome and had a volume of intersection close to 2m $\times$ 2m $\times$ 2m. The output of each camera was recorded on a separate consumer grade S-VHS VCR for later digitization and

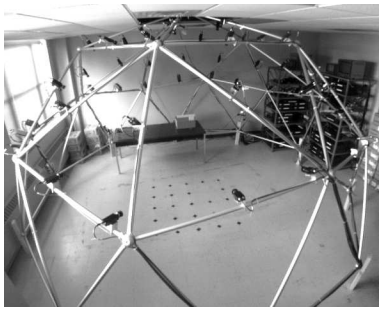

Figure 2: **3D Dome**: The first Virtualized Reality setup [12]. (a) The basic setup showing the dome, the cameras, and the VCRs. (b) The setup ready for recording.

processing. A separate editing VCR connected to a computer with a frame grabber was used for off-line, sequential digitization of the tapes. Figure 2 shows the setup with the rack of VCRs at the back. The cost of the setup was about $1000 per channel. Direct digital capture of multiple video channels was not technologically feasible until much later. Datacube had a product around that time that could capture a few seconds of video digitally. That system cost close to a $1 million and wasn't scalable in time or the number of channels. We could capture hours of the event using any number of cameras at the S-VHS quality level. Details of the capture system can be seen from the report we wrote on it [19].

Direct digital capture can provide higher quality images than using tapes. The quality of the image has impact on the appearance of the captured event through textures as well as on its structure computed from them. We built the *3D Room* facility, shown in Figure 3, with digital capture in 1998. The dome was replaced by a room to increase the working volume. The system captured the output of each camera directly onto the system memory and subsequently

Figure 3: **3D Room:** the Virtualized Reality setup with digital capture [14].

transferred it to secondary storage. The 3D Room system used 49 cameras; 10 were mounted on each wall of the room, and 9 on the ceiling. Seventeen PCs captured the event, each with 3 frame grabber cards and 512 MB of RAM. Video was captured in real-time into the main memory and later streamed to disk. The memory size limited capture to about 800 frames per PC, or just under 18 seconds of event time at 15 frames per second. Details on this facility can be obtained from a report on it [14].

Direct capture to a secondary storage device became feasible as the technology advanced. The *Virtualizing Studio* system, built in 2002, can capture the output of all cameras directly onto the hard disks. This facility uses 48 color cameras for event capture and increased the event space to a 6.1m×6.7m×4.3m room. Nine cameras are mounted on the ceiling and the rest along the walls at two different heights. High-end 3CCD cameras with automatic zoom are used. Figure 4 shows a panoramic view of 2 sides of the studio. Studio quality lighting in the room was used to alleviate some of the problems we faced in earlier setups. An array of 24 Linux PCs, each with 2 frame grabber cards, grab the action live. Each PC has 3 high-performance SCSI hard disks with bandwidth sufficient for live storage. The recording time is limited only by the available disk space. Hours of recording using 48 cameras in full colour is now possible with the Virtualizing Studio.

The technology of cameras, buses, motherboards, memory, and discs have advanced much further today. It is possible to digitally capture the outputs of a large number cameras and stream them directly to the disc today. The cameras used in the Virtualized Reality setup are standard video cameras with a resolution of 640×480. Firewire cameras of resolutions 1024×768 and beyond are available today. Firewire also gives sufficient bandwidth for 3-4 cameras to a single PC, which can store them onto the disk. The free-viewpoint video uses 7 Firewire cameras for capturing. The 3D video recorder from ETH uses three 3D bricks, each with 3 cameras, for capturing. The video-based rendering from MSR used 8 Firewire cameras arranged roughly along a line and capture the action live.

3.2 Frame Synchronization and Labelling

The cameras view the same dynamic event in discrete time intervals. These time instants have to be accurately synchronized and identified across cameras for later processing. There are two steps to synchronized multicamera capture. First, the camera frames must be synchronized to one another. Supplying a reference video signal as the genlock to all cameras will keep them in sync and will ensure they sample the world simultaneously.

The second step to line up the frames or time instants from different cameras. This requires giving a unique number to label each frame of all cameras. The frames with the same label represent a true snapshot of the scene from different cameras. We used the SMPTE standard Vertical Interval Time Code (VITC) mechanism for frame labelling. The VITC can be inserted into each video stream using an off-the-shelf time-code insertion unit. The common number identifying each time instant is supplied to the insertion units as a common, daisy-chained Longitudinal Time Code (LTC), another SMPTE standard time code mechanism. The LTC is generated from the reference video signal by another off-the-shelf time code generator device. VITC has the timing information embedded as visual markings into the vertical blanking portion of each field of the video frames to identify it. This information can be recovered from each field while digitization. Many equipments including professional video cameras and VCRs support LTC and VITC mechanisms. We set the frame grabber to capture the relevant portions of the vertical blanking region. The VITC time code can be recovered from the visual markings by a program.

We built a special unit to test the frame and field capture to ensure perfect frame synchronization and alignment. The unit showed frame and field numbers of the reference video signal accurately on a large LED display. The physical unit was imaged using all cameras and the alignment of the imaged frame number and the time recovered from the digitization was studied. The tests confirmed that perfect alignment to the field level. A detailed account of the synchronizing mechanism can be found in our technical report [19]. The same mechanism of frame alignment also worked for the later setups that captured the videos digitally.

The IEEE 1394 Firewire has emerged as a standard for connecting high-quality cameras. The Firewire bus has signals that help in synchronizing upto 4 cameras

Figure 4: **Virtualizing Studio**: Panoramic view

to the frame level. The bandwidth requirements for live capture of 4 cameras is high. Commercial devices are available that can synchronize multiple Firewire buses for scaling beyond 4 cameras. The video-based rendering system uses Firewire-based synchronization. The free-viewpoint video the 3D video recorder use externally triggered Firewire cameras.

3.3 Digitization

The 3D Dome setup used consumer-grade S-VHS VCRs and tapes. We used a computer-controlled professional VCR and a high-quality frame grabber connected to a Sun workstation for tape to digital conversion of videos. A special program reads the time code from the visual markings on each field stored on the tapes and captures all frames specified by the user. This requires playing the tape multiple times under computer control since the framegrabber to the computer bandwidth was slow. The 3D Room setup used direct capture into PC memory. A central control station coordinated with the PCs to start and stop the live capture between specified time code values. Each PC maintained a queued capture loop containing all requests for frames. The frame grabber generated an interrupt when a queued time code was encountered [14]. The Virtualizing Studio setup uses Linux PCs with high-speed SCSI disks capable of sending the captured frames to disk in real time. A similar queuing of capture requests ensured that a range of time codes is captured and streamed automatically to the hard disk as the event proceeds. The modern modelling efforts use digital capture of the video streams to a secondary storage device.

3.4 Camera Calibration

The cameras involved need to be calibrated to a common framework if their outputs have to be correlated with one another in anyway. Camera calibration fits an analytical model to the camera's projection, typically as an ideal pinhole model with a few parameters. Calibration is critically important as errors in calibration can distort 3D reconstructions systematically and amplify noise in subsequent processes. Calibrating a large number of cameras to a common reference frame is a challenging task, especially if the cameras are designed to cover a space from all sides. The scalability requirements of the system implies that the calibration procedure should be simple and extensible to a large number of cameras. Calibration can be weak or full depending on the model estimated; it can be performed with a special calibration procedure or be recovered from the world as the event happens. We opted for a full calibration involving as accurate a procedure as the event setup can afford as accurate calibration is fundamental to the entire downstream processes.

The calibration data is commonly represented using 5 intrinsic parameters – which determine the projection properties of the camera with respect to its internal coordinate frame – and 6 extrinsic parameters, which place the camera in a fixed world coordinate frame. The focal length, aspect ratio, skew, and the image center coordinates are the intrinsic parameters. The position and orientation of the camera in the global coordinate frame are the extrinsic parameters. Typically, one or two lens distortion parameters are also recovered when lenses of low focal lengths are used.

We used the strong calibration scheme by Tsai [30] to calibrate the Virtualized Reality system. Strong calibration algorithms are well researched and have with stable implementations. A strongly calibrated setup allows full 3D Euclidean reconstructions which enable handling the recovered models using standard tools. Also, strong calibration places each camera in a world frame independently and can be scaled to an arbitrary number of cameras.

We used a two-step approach to compute the intrinsic and extrinsic calibration parameters in the first setup. The intrinsic parameters are recovered as a first step in a calibration cell using a plane of points that was moved accurately in space. The point positions in

3D are known and the image positions can be recovered easily from the images. The extrinsic parameters were recovered after mounting the cameras in their places using known 3D structure visible to all cameras. The 3D Dome setup used a square pattern of dots on the floor for the second step. Subsequently, calibration was performed by moving a known linear arrangement of computer controlled LEDs in the working volume. A planar 2m×2m object with 64 LEDs arranged in an 8×8 grid was used for this purpose. The plane could be kept horizontal and positioned accurately in one of 5 levels using a carefully made base and a set of legs. The LEDs were turned on one at a time under computer control to ease its detection in the images. A cloud of calibration points can be obtained with minimal manual effort for each camera by sequencing through the LEDs for each plate position.

A similar calibration procedure is used by other event capture efforts also. Newer calibration procedures such as the one by Zhang [35] are known to require less exacting calibration objects and are more popular today.

4 Event Modelling and Representation

Experiencing the event, either live or later, requires the ability to give an immersive feeling to a viewer by placing the viewer anywhere in the event space. If very large number of views are available, we could provide a pseudo-immersive experience by switching to the view closest to the one demanded by the viewer. The EyeVision^{TM} system, an off-shoot of the Virtualized Reality project that was debuted at the 2001 Superbowl, provided such an illusion of immersion [29]. It involved coordinated automatic tracking and zooming of about 30 cameras arranged roughly in a circle, coupled with the ability to sequence through their views for each time instant. Many mosaicing techniques also provide this type of immersion by capturing a scene from inside looking outside [3]. Such techniques cannot provide unrestricted immersion as they cannot generate in-between views. A suitable model of the event is necessary to achieve unrestricted immersion. Implicit models are used by many IBR techniques. Geometric models enable the use of software and hardware tools from Computer Graphics and Virtual Reality.

We create a time-varying geometric model of the event in the form of triangulated surfaces using Computer Vision techniques. We used a multicamera stereo algorithm to recover the scene structure. A stereo program takes images from a combination of cameras and computes the depth for each visible point or pixel in the image. Such viewer-centered description of the world is called $2\frac{1}{2}$-D structure [17] in Computer Vision. Multiple $2\frac{1}{2}$-D estimates provide sufficient redundancy that can be used to correct inaccuracies in one or more of them. The view-dependent structures can be merged to get a global model of the event, if necessary. The appearance of the event is captured in the form of view-dependent or global texture models. The former is available from the cameras directly. The latter is computed by projecting the images from cameras to the global model. A textured polygon model thus obtained can be converted to any standard 3D model exchange formats for use with modelers or image generators.

4.1 Visible Surface Models

Stereo vision recovers the depth – the distance along the optical axis, perpendicular to the image plane – of points that can be identified uniquely in different cameras. This process involves two steps. Computing pixel correspondences or the image points that are the projections of the same world point between the views and triangulating using the baseline distance between cameras. The use of multiple cameras can improve the accuracy and reliability of the correspondences and the recovered structure. This is important for recovering dense structure for the scene. Multiple cameras reduce the effects of occlusion of parts of the scene from certain viewpoints. The multibaseline stereo (MBS) algorithm uses multiple cameras to obtain a more accurate and less ambiguous depth map of the scene [22]. One camera is designated as a reference camera and the depth values are computed with respect to it. Other cameras provide baselines for stereo matching. The match scores for all pairs are integrated into a single match score that has less uncertainty than the individual scores. Our algorithm can use an arbitrary number of cameras in general positions and orientations, given the calibration parameters for each. The disparity space of conventional stereo is replaced by a more general inverse depth space for computing matches.

The MBS algorithm computes depth estimate for each pixel with respect to the reference camera. This gives a $2\frac{1}{2}$-D description of the scene from the camera's point of view. Given the strong calibration parameters and the depth of a pixel, the (x, y, z) coordinates of the world point that projects to it can be computed. Figure 5 shows the reference camera image and the depth map corresponding to it for a number of cameras for the same time instant.

The cameras in our system are distributed nearly uniformly in space. We compute the $2\frac{1}{2}$-D structure with each of the 51 cameras as the reference, with cameras in the immediate neighborhood providing the

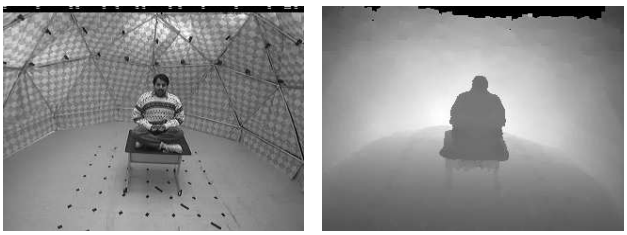

Figure 5: Reference images and the depth map computed using multibaseline stereo.

baselines. The number of cameras used for a single stereo computation ranges from 4 to 7 depending on the arrangement of cameras in the reference camera's neighborhood.

A 3D surface model can be created from a depth map by constructing 2 triangles for each 2×2 section using a diagonal. This procedure connects adjacent points of the point-cloud by a triangle edge. Occlusion boundaries using an appropriate threshold on the the difference of depth values along any side of a triangle. Such triangles form an occluding surface and need not be rendered. The surface model will have holes in their place This scheme converts the structure computed by stereo to a Visible Surface Model (VSM) which is a triangulated model of the surfaces visible from the reference camera's location. The VSM can be rendered as a triangle mesh with the camera image as texture to generate new views, particularly those close to the original camera.

The video-based rendering project uses a colour-segmentation based stereo algorithm to recover the depth maps. Disparity smoothing is performed to remove the effects of noise. They compute the depth map with respect to each of the 8 cameras used for acquisition. The 3D video project uses three 3D video bricks each of consists of three cameras and a projector for active lighting. Alternate frames have projected light to enable capture of accurate structure and unmodified texture.

4.2 Complete Scene Model

An individual surface model provides the partial structure of the scene. It is desirable to construct a global, coherent scene model that contains information from all surface models. We achieve this by merging the individual depth maps into a single Complete Scene Model (CSM) in a volumetric space. We modified the volumetric merging technique by Curless and Levoy [5] to suit the noisy data given by the MBS algorithm. This algorithm accumulates the evidence given by each depth map in a global voxel space. Each depth value contributes a zero value to the voxel that contains that point, positive values to the occluded voxels beyond the point, and negative values to empty voxels in the free-space upto the point. The isosurface of zero values after processing all depth values give the consensus surface of the scene and is recovered as a triangulated model using the marching cubes algorithm. This approach works better when gross errors could be present in each individual depth map as is the case with stereo on our scenes. More details on the volumetric merging process and its properties can be found in [25].

The CSM model so generated is a standard graphics model. A global texture is computed for this model by combining the best views for different portions of the model. New views can be generated quickly using such a model on a graphics workstation.

The 3D Video project computes a view independent representation of the scene as a cloud of points, which can be rendered using standard point-based rendering techniques. The free-viewpoint video effort also computes a global model as intersection of silhouettes. A photoconsistency enforcement step further reduces the effects of the outliers.

4.3 Shape from Silhouette

The silhouettes of the scene objects represent an irregular cone with the camera as its apex that contains them completely. If many views are available, the cones for each can be intersected to give a convex volume that will bound the model of the object. The bound gets tighter as more views are used in the process. The shape computed from silhouettes can be used in place of stereo or to enhance the stereo algorithms especially if models of individual objects or persons are being sought. We used such models to recover human models in specific situations [4].

5 Experiencing Digitized Events

Visual experiencing of a digitized dynamic event involves the generation of arbitrary views of it at view-time. An observer can be given the impression of walking or flying through the event independently if a succession of views along his or her path of motion can be shown.

5.1 Using the CSM

The CSM model with its texture is equivalent to a graphics model of the scene. Standard graphics hardware and software can be used to navigate using such a model. Real-time navigation requires on high-end machines as the models constructed using the volumetric merging tend to have a large number of tiny triangles.

Figure 6 shows a few generated views of a virtualized scene involving two people and a basketball.

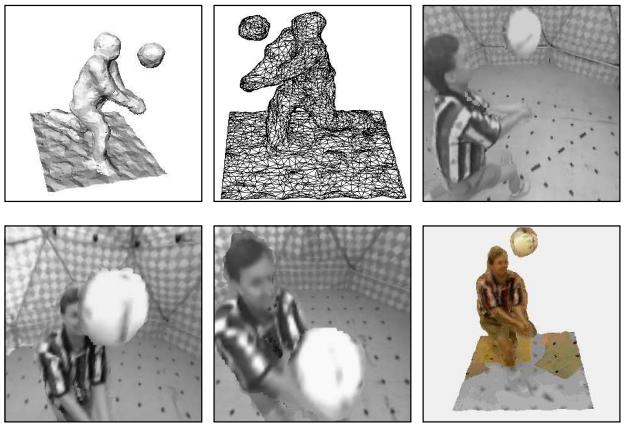

Figure 6: Views generated using the global model [20]. The non-textured view shows the surface model.

The untextured, surface view shows the global model. The textured views correspond to a viewer navigating through the space.

The 3D Video and free-viewpoint video efforts have global models of the scene and render them in ways very similar to the CSM rendering described above.

5.2 Using the VSMs

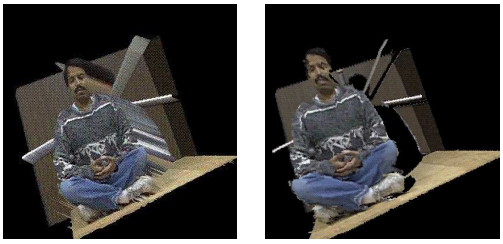

Figure 7: Rendering a scene using the stereo output of one cameras [12]. Left: Artificial membranes appear if the surface is assumed continuous. Right: Holes appear if depth discontinuities are detected and eliminated while rendering

It is also possible to generate arbitrary viewpoints using the VSMs without merging them. Each VSM represents a partial model of the scene surface visible from a camera. If a VSM is used to render new views, holes will appear corresponding to the regions of the scene which were not visible to that camera. These holes can be filled using nearby VSMs as these regions are visible to them. If the cameras are arranged so as to cover the space roughly uniformly, the occluded regions of one VSM will be visible in a neighboring VSM. Figure 8 demonstrates this situation. Given the location and orientation of a desired view, the VSM whose

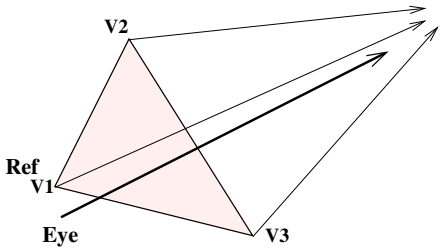

Figure 8: Closest VSM is chosen as reference VSM and two surrounding ones as supporting ones [24].

direction of view is closest in angle is selected as the *reference* VSM to generate most of the new. As the viewpoint moves away from the VSM, a pair of neighboring VSMs can be selected to provide hole-free views of the scene. We draw lines from the reference VSM to each of its neighboring VSMs. The sector which the extended view direction intersects identifies the two VSMs that, along with the reference VSM, cover the view from all directions. Figure 8 demonstrates the selection.

The new view is rendered as follows. First, the scene is rendered using the reference VSM. The hole regions in the rendered view are identified by rendering the occluding surfaces – consisting of the triangles of the mesh that exceeded the depth threshold – of the model. The scene is next rendered using the supporting VSMs in some order, limiting their impact to the hole regions. Since the supporting VSMs are selected to cover the scene from beyond the viewpoint, all the holes are usually filled. See [20] for more details on view generation.

The video-based rendering effort renders exactly two VSM models like our scheme but use matting near occlusion boundaries. By interpolating the colours near the boundary, they are able to provide continuous appearance variation even when the appearance differs between the VSMs.

5.3 Using Implicit Models

Explicit graphics models such as the VSM and the CSM are not necessary for new view generation. Image Based Rendering attempts to generate new views of a scene by manipulating existing views, given a sufficient number of input views of the scene. Several IBR methods have been reported in the past decade. They can be said to use implicit models of the visual structure of the scene. These could be the plenoptic function in the form of a lightfield [15] or a lumigraph [9], or another representation suitable for generating new views. The collection of images representing a snapshot of the region can be directly interpolated

to give new views, using the calibration information wherever necessary. The appearance-based view generation method can generate new views of a dynamic event for any viewpoint. This is performed by interpolating between two selected views using simple correspondence. The information from multiple cameras were analyzed together in a volumetric space to compute accurate and dense correspondence. Thus, this method was able to generate new views of even the occluded parts of the scene. More details can be found from the paper by Saito et al. [27] on this topic.

6 Manipulating a Digitized Event

Can the virtualized dynamic event be edited and modified similar to scenes created by hand? If the basic representation of the model is compatible with standard graphics models, it can be manipulated using standard tools. The manipulation could involve adding synthetic or virtualized objects into it, removing objects from it, changing the appearance of one or more objects, etc. A good graphics environment model has spatially compact objects arranged in hierarchies to facilitate efficient handling. The models created from images do not automatically come with a hierarchical structure and this could be a drawback in their use. Spatial partitioning techniques can be used to generate acceptable hierarchies. However, the identification of distinct objects will have to be performed with considerable human assistance.

Figure 9: Compositing two virtualized events [31]. Top row shows the models of the 1-person event and the 2-person event that has been composited to a 3-person event. Bottom row shows the model event progression over time and textured views of it.

We can introduce other objects into a virtualized scene easily it has a coordinate frame and geometric structure. The new object can be transformed appropriately before placing it. Such objects could be captured models too. Deleting an object from a virtualized event needs more care as the models do not have easy part labels. However, tools to slice existing triangle meshes are available and can be used to interactively define the object of interest and remove it from the model. Figure 9 shows the results of compositing two independently produced virtualized events. A more interesting operation is to change the appearance of an object in the scene. For example, synthetic costumes could be added to a character in a virtualized event. Appearance is captured in virtualized models using the texture map. Thus, altering the appearance involves editing the texture map. The texture map could be edited using a standard image editing tool. The portion of the geometry whose appearance has to be altered needs to be identified in the model first. Since each geometric triangle has an associated texture triangle, identification of the texture triangle for modification can follow as a second step. The scene flow parameters can help propagate the changes in appearance made in the model of one time instant to be propagated to the subsequent ones. Altering the shape of one or more objects in an event is a more difficult task. A standard editing tool can manipulate these triangulated models. However, the lack of hierarchical geometric structure in the models make this more challenging.

Compositing different virtualized scenes was also demonstrated by the video-based rendering project. The free-viewpoint video effort is primarily aimed at capturing models of human actors that can subsequently by placed in other virtual environments.

7 Conclusions and Challenges

In this paper, we presented Virtualized Reality – the first project that successfully modelled large dynamic events – from a historical perspective. Virtualized Reality successfully combined synchronous capture, multicamera stereo, shape from silhouettes, and novel view generation to digitize large, multiperson, dynamic events and let viewers post-experience it from viewpoints of their choice. Limited participative experiencing was also demonstrated.

Multicamera modelling and digitization of large dynamic models remains an important and challenging problem even today. We would like to present some analysis of the present and speculations for the future. Several aspects of the process still remain challenges.

- Cameras with higher resolution and fidelity will continue to appear. Stereo algorithms are steadily improving in performance. Special sensors that recover appearance and structure simul-

taneously are available in the laboratories and will play a big role in the process. Algorithms that combine motion, silhouettes, shading, and other information can be combined in a joint framework to recover global structure from a set of images.

- Image-based rendering with implicit structure may also play a dominant role in immersive experience of digitized events. Hybrid representations that use approximate geometry and good texture are promising [1]. Dynamic lightfields and Depth Image based rendering are extending IBR to dynamic scenes [34, 23].
- Representations of dynamic scenes are many. Depth image based representations are natural for captured images, easy to render, provide excellent quality due to the locality properties, and compress well [34, 23, 32]. Global models will be used less as they are difficult and non-robust to compute. Point-based representations have become more prominent today and can capture the essence of current process of modelling from images. Ordered points may become important to represent digitized dynamic events.
- The computation and rendering requirements for such a system is high and will continue to grow with resolution and fidelity of the expected models. The Graphics Processor Units (GPUs) present immense opportunities to provide much of this. GPUs can be used for real time rendering of captured data directly [32]. The current GPUs can in addition store large models in compressed format and perform decompression, selection, and rendering. Future generation GPUs may also directly capture images from cameras and recover the structure much quicker than the CPUs.
- Matching the appearance from multiple cameras is still a challenge. Matting and blending technique can avoid sharp changes in appearance in a single view. The appearance can, however, change considerably as different views provide the primary image. Accurate photometric calibration of cameras are required for this purpose.

Many of these challenges will be overcome in the near future by the community. Immersive and participative visual medium will open up new vistas in entertainment.

References

[1] C. Buehler, M. Bosse, L. McMillan, S. J. Gortler, and M. F. Cohen. Unstructured Lumigraph Rendering. In *SIGGRAPH*, 2001.

[2] J. Carranza, C. Theobalt, M. A. Magnor, and H.-P. Seidel. Free-viewpoint video of human actors. *ACM Trans. Graph.*, 22(3):569–577, 2003.

[3] S. E. Chen. QuickTime VR: an image-based approach to virtual environment navigation. In *SIGGRAPH*, pages 29–38, 1995.

[4] G. K. M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *CVPR (1)*, pages 77–84, 2003.

[5] B. Curless and M. Levoy. A Volumetric Method for Building Complex Models from Range Images. In *SIGGRAPH*, 1996.

[6] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and Rendering Architecture from Photographs: A Hybrid Geometry and Image-Based Approach. In *SIGGRAPH*, 1996.

[7] H. Fuchs, G. Bishop, K. Arthur, L. McMillan, R. Bajcsy, S. Lee, H. Farid, and T. Kanade. Virtual Space Teleconferencing using a Sea of Cameras. In *Proceedings of the First International Symposium on Medical Robotics and Computer Assisted Surgery*, pages 161–167, 1994.

[8] G. Godin, J. A. Beraldin, J. Taylor, L. Cournoyer, M. Rioux, S. El-Hakim, R. Baribeau, F. Blais, P. Boulanger, J. Domey, and M. Picard. Active optical 3d imaging for heritage applications. *IEEE Comput. Graph. Appl.*, 22(5):24–36, 2002.

[9] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The Lumigraph. In *SIGGRAPH*, 1996.

[10] T. Kanade. Very Fast 3-D Sensing Hardware. In *In Sixth International Symposium of Robotics Research*, pages 185–198, 1993.

[11] T. Kanade, H. Kano, S. Kimura, E. Kawamura, A. Yoshida, and K. Oda. CMU Video-Rate Stereo Machine. In *Proc. of Mobile Mapping Symposium (MMS'95)*, 1995.

[12] T. Kanade, P. J. Narayanan, and P. W. Rander. Virtualized Reality: Concept and Early Results. In *In IEEE Workshop on the Representation of Visual Scenes*, 1995.

[13] T. Kanade, K. Oda, A. Yoshida, M. Tanaka, , and H. Kano. Video-Rate Z Keying: A New Method for Merging Images. Technical Report CMU-RI-TR-95-38, Carnegie Mellon University Robotics Institute, 1995.

[14] T. Kanade, H. Saito, and S. Vedula. The 3D Room: Digitizing Time-Varying 3D Events by Synchronized Multiple Video Streams. Technical Report CMU-RI-TR-98-34, Robotics Institute, Carnegie Mellon University, 1998.

[15] M. Levoy and P. Hanrahan. Light Field Rendering. In *SIGGRAPH*, 1996.

[16] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. E. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk. The digital michelangelo project: 3d scanning of large statues. In *SIGGRAPH*, pages 131–144, 2000.

[17] D. Marr. *Vision*. W. H. Freeman, 1982.

[18] S. Moezzi, L.-C. Tai, and P. Gerard. Virtual View Generation for 3D Digital Video. *IEEE Multimedia*, 4(1):18–26, 1997.

[19] P. J. Narayanan, P. W. Rander, and T. Kanade. Synchronizing and Capturing Every Frame from Multiple Cameras. Technical Report CMU-RI-TR-95-25, Robotics Institute, Carnegie Mellon University, 1995.

[20] P. J. Narayanan, P. W. Rander, and T. Kanade. Constructing Virtual Worlds Using Dense Stereo. In *IEEE International Conference on Computer Vision (ICCV)*, Jan 1998.

[21] M. Okutomi and T. Kanade. A Multiple Baseline Stereo. In *Proceedings of Computer Vision and Pattern Recognition*, 1991.

[22] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353 – 363, 1993.

[23] S. K. Penta and P. J. Narayanan. Compression of multiple depth maps for ibr. *The Visual Computer*, 21(8-10):611–618, 2005.

[24] P. Rander, P. J. Narayanan, and T. Kanade. Virtualized Reality: Constructing Time-Varying Virtual Worlds from Real World Events. In *Visualization*, 1997.

[25] P. W. Rander, P. J. Narayanan, and T. Kanade. Recovery of Dynamic Scene Structure rom Multiple Image Sequences. In *Proc of Multisensor Fusion and Integration for Intelligent Systems (MFI 96)*, 1996.

[26] R. Reddy. Towards Teleportation, Time Travel and Immortality. In *ACM 50th Anniversary Conference*, March 1997.

[27] H. Saito, S. Baba, M. Kimura, S. Vedula, and T. Kanade. Appearance-Based Virtual View Generation of Temporally-Varying Events from Multi-Camera Images in the 3D Room. In *Proceedings of the International Conference on 3-D Digital Imaging and Modelling*, October 1999.

[28] H.-Y. Shum, S. B. Kang, and S.-C. Chan. Survey of image-based representations and compression techniques. *IEEE Trans. Circuits Syst. Video Techn.*, 13(11):1020–1037, 2003.

[29] The New York Times. Turning the Super Bowl Into a Game of Pixels. January 25, 2001.

[30] R. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323 – 344, 1987.

[31] S. Vedula, P. Rander, H. Saito, and T. Kanade. Modeling, combining, and rendering dynamic real-world events from image sequences. In *4th Internl Conference on Virtual Systems and Multimedia*, 1998.

[32] P. Verlani, A. Goswami, P. J. Narayanan, S. Dwivedi, and S. K. Penta. Depth Images: Representations and Real-time Rendering. In *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, 2006.

[33] M. Waschbüsch, S. Würmlin, D. Cotting, F. Sadlo, and M. H. Gross. Scalable 3d video of dynamic scenes. *The Visual Computer*, 21(8-10):629–638, 2005.

[34] Q. Wu, K.-T. Ng, S.-C. Chan, and H. Shum. An object-based compression system for a class of dynamic image-based representations. In *Visual Communications and Image Processing*, 2005.

[35] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[36] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *ACM Trans. Graph.*, 23(3):600–608, 2004.