The Plenoptic Video

Shing-Chow Chan, *Member, IEEE*, King-To Ng, *Member, IEEE*, Zhi-Feng Gan, *Student Member*, Kin-Lok Chan, and Heung-Yeung Shum, *Senior Member, IEEE*

***Abstract*—This paper presents a system for capturing and rendering a dynamic image-based representation called the plenoptic video. It is a simplified light field for dynamic environments, where user viewpoints are constrained to the camera plane of a linear array of video cameras. Important issues such as multiple camera calibration, real-time compression, decompression and rendering are addressed. The system consists of a camera array of eight Sony CCX-Z11 CCD cameras and eight Pentium 4 1.8-GHz computers connected together through a 100 Base-T local area network. It is possible to perform software-assisted real-time MPEG-2 compression at a resolution of (720 × 480). Using selective transmission, we are able to stream continuously plenoptic video with (256 × 256) resolution at a rate of 15 f/s over the network. For rendering from raw data on the hard disk, real-time rendering can be achieved with a resolution of (720 × 480) and a rate of 15 f/s. A new compression algorithm using both temporal and spatial predictions is also proposed for the efficient compression of the plenoptic videos. Experimental results demonstrate the usefulness of the proposed parallel processing based system in capturing and rendering high-quality dynamic image-based representations using off-the-shelf equipment, and its potential applications in visualization and immersive television systems.**

***Index Terms*—Camera array, data compression, dynamic environment rendering, image-based rendering (IBR), parallel processing.**

I. Introduction

IMAGE-BASED RENDERING (IBR) has recently emerged as a promising alternative to three-dimensional (3-D) computer models for photo-realistic rendering of scenes and objects from a collection of densely sampled images. Central to IBR is the plenoptic function [1], which forms a new framework for developing sophisticated virtual reality and visualization systems. Another important advantage of IBR is the superior image quality that it offers over 3-D model building, especially for very complicated real world scenes. It also requires much less computational power for rendering, regardless of scene complexity. Unfortunately, image-based representations usually consist of hundreds or thousands of images, which involve large amounts of data. To simplify the capturing and storage of the plenoptic function, various image-based representations of lower dimensions have been advocated [2]–[11]. Most image-based representations reported so far deal with static scenes. This is largely attributed to the logistical difficulties in capturing and transmitting dynamic representations, which involve huge amounts of data. In fact, it has stimulated considerable research effort into efficient compression methods for various image-based representations such as the light field, lumigraph, and concentric mosaics [12]–[17]. A study of real-time capturing, compression, and rendering of image-based representations for dynamic environments is thus highly desirable. Such representations can also be viewed as versatile generalizations of traditional images and videos, which might be further developed into new interactive or immersive television systems.

Toward this goal, we constructed in this paper a system for real-time capturing, compression and rendering of a simplified light field for dynamic scenes. We coined these simplified dynamic light fields (SDLF) the plenoptic videos, because of their close relationship with traditional videos supporting multiple viewpoints. Through this system, it was also demonstrated how parallel processing and inexpensive equipment can be utilized to capture and process image-based representations of dynamic scenes efficiently and mostly in real-time, which is one of the major obstacles in dynamic IBR research. Unlike capturing static image-based representations, methods for calibrating multiple cameras and compressing the plenoptic videos have to be developed. In particular, an MPEG-2 like compression algorithm employing both spatial and temporal compensations is proposed for the efficient storage and transmission of plenoptic videos. It also addresses the important random access problem in light field rendering. Experimental results show that spatial prediction significantly improves the coding efficiency. Together with temporal prediction, most of the image pixels can be predicted satisfactorily. Immediate applications of the proposed system are "interactive 3-D electronic catalog or brochures" and "short plenoptic video advertisement clips," where the plenoptic videos are distributed either in form of DVDs or the Internet for viewing by potential customers on a computer (mouse-controlled). This was demonstrated in our demos, consisting of a glass music box and two lead crystals, which are usually very difficult to model with photo-realistic quality. Another possible application is a head and shoulder-type 3-D videophone, where the depth variation, like the music box sequence, is relatively small [18]. In these applications, not too many numbers of cameras, say eight or less, can be employed to convey to users a reasonable good sense of immersive viewing, without excessively increasing the transmission bandwidth. The rest of the paper is organized as follows: a brief overview of previous research related to the current work is given in Section II. The proposed plenoptic

Manuscript received Jul. 9, 2003; revised April 12, 2004 and January 13, 2005. This work was supported in part by the Research Grant Council of Hong Kong Special Administrative Region, China, under project HKU 7158/03E. Part of this work was presented at ICASSP'2003 and ISCAS'2004. This paper was recommended by Associate Editor J.-N. Hwang.

S.-C. Chan, K. T. Ng, Z. F. Gan, and K. L. Chan are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: scchan@eee.hku.hk; ktng@eee.hku.hk; zfgan@eee.hku.hk; kinlok@eee.hku.hk).

H. Y. Shum is with Microsoft Research Asia, Beijing 100080, China (e-mail: hshum@microsoft.com).

Digital Object Identifier 10.1109/TCSVT.2005.858616

1051-8215/$20.00 © 2005 IEEE

video system is described in Section III. Then, the design and implementation of the plenoptic video capturing system are explained in Section IV. Sections V and VI are devoted to the compression and rendering of the plenoptic video. Finally, conclusion and possible future work are given in Section VII.

II. Previous Work

The plenoptic video is a kind of simplified light field for dynamic environments. It belongs to the general class of image-based representations. There have been considerable advances in IBR research and many interesting representations were proposed. Interested readers are referred to a recent survey paper in [17] for more details. More recently, there were attempts to construct light field video systems [19]–[22] for different applications. These include the Stanford multicamera array [19], the 3-D rendering system of Naemura *et al.*. [20], and the (8 × 8) light field camera of Yang *et al.* [21]. The Stanford array consists of more than one hundred cameras and is intended for large environment applications. It uses low cost CMOS sensors and dedicated hardware for real-time compression. The systems in [20], [21] consist of, respectively, 16 and 64 cameras and are intended for real-time rendering applications. Unlike the Stanford array and our system, they do not support real-time compression of the captured videos. Our system has different design tradeoffs, which yield very good rendering quality. Our intention is not to compare directly with these systems, but rather to disseminate our experience in system construction and compression techniques for other researchers to build inexpensive arrays and fairly high-quality rendering systems using off-the-shelf equipment. The latter has been one of the major obstacles in the study of dynamic image-based representations. A more detailed comparison of these systems with the proposed system will be given in the Section IV-C.

III. Plenoptic Videos

A. Proposed Plenoptic Videos

Among the static image-based representations reported so far, light fields are relatively simpler to be generalized to dynamic scenes. Therefore, they are chosen in this study. From [23], the sampling rate of static light fields depends on the depth of the scene. In order to reduce the effect of aliasing, the number of cameras in a two-dimensional (2-D) arrangement can be very large, say 16 × 64. This might create hundreds of videos, which have to be compressed and stored in real-time. The calibration of such a large camera array is also problematic and very time consuming. To avoid this large dimensionality and the excessive hardware cost, we limit our study to light fields with viewpoints being constrained along a line. This simplified dynamic light field, which we call the plenoptic video, has a dimensionality of four. Apart from the simplicity of the overall system, there are two reasons for such a choice. For instance, the user can still observe significant parallax and lighting changes along the horizontal direction. Furthermore, the given number of cameras can be used to maximize the sampling rate along the horizontal direction and thus the risk of insufficient sampling in a 2-D configuration with the same number of cameras.

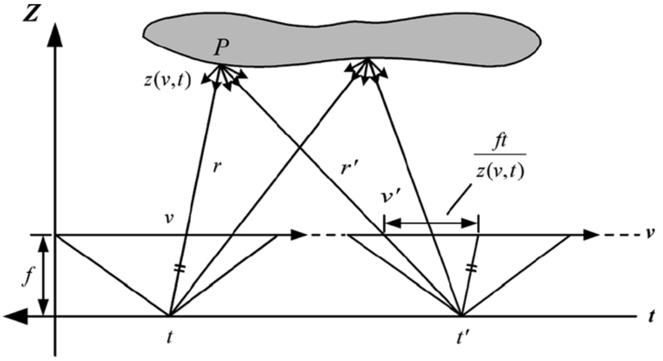

Fig. 1. Illustration of a 2-D light field section. A point is observed by two cameras t and t'. The object surface is assumed to be Lambertian.

B. An Approximate Sampling Analysis

The sampling analysis of static light fields was first studied in [23]. In the standard two-plane ray space parameterization, there is a camera plane, with parameters (s, t), and a focal plane, with parameters (u, v). Each ray in the parameterization is uniquely determined by the quadruple (u, v, s, t). For fixed values of s and t, we obtain an image taken at a location indexed by (u, v). Interested readers are referred to [6] for more details. Fig. 1 shows an example of a simple 2-D light field. Assuming a pinhole camera model, the pixel value observed is the convolution of the plenoptic function with the point spread function. In the camera position t', a ray r' of a point P on the object surface is observed, and it is recorded as a pixel at position v'. Whereas, for camera position t, the same point P at depth $z(v, t)$ is observed as ray r and is recorded as pixel v. The disparity, or the displacement of the image pixel, of P is $(ft/z(v, t))$, where f is the focal length. If the object surface is Lambertian,[1] pixels v and v' will be identical to each other. If $z(v, t)$ is known, then the rays captured at certain camera positions (say regularly along a straight line) can be used to reconstruct the images in between, if the sampling is sufficiently dense and there is no occlusion. Using the piecewise constant depth model, it was shown in [23] that the spectral support of the static light field is approximately bounded by its maximum and minimum depths. For the commonly used rectangular sampling lattice, the minimum sampling rate in the t direction is $f_t = K_{\Omega_v} f \cdot h_d$, where $K_{\Omega_v} = \min(B_v, 1/\delta_v), h_d = (z_{\min}^{-1} - z_{\max}^{-1})/2$. B_v and δ_v are, respectively, the bandwidth and output resolution of the light field in the v direction. $z_{\min}$ and $z_{\max}$ are, respectively, the minimum and maximum depth values of the scene, f is the focal length. For dynamic scenes, the minimum and maximum depth values $z_{\min}$ and $z_{\max}$ are functions of time τ (i.e., $z_{\min}(t)$ and $z_{\max}(t)$), which depend on the motion of the objects in the scene. In our rendering experiment, a mean depth, possibly time-varying, is assumed for the dynamic scene. A similar expression applies to the s direction.

[1] Since the pixel values v and v' will be linearly interpolated, if the angular variations of the bidirectional reflectance distribution function (BRDF) is bandlimited, then lighting charges for nonlambertian surface can also be captured with sufficient sampling.

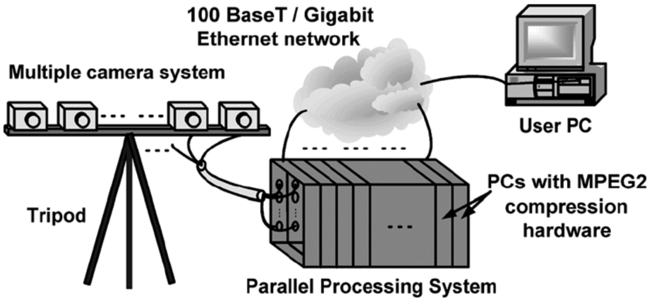

Fig. 2. Block diagram of the plenoptic video system.

IV. Plenoptic Video System

A. System Implementation and Results

Fig. 2 shows the block diagram of our plenoptic video capturing and processing system. A set of synchronized video cameras is used to capture the light field images at each time instant to form sequence of videos. The video signals are then fed to the real-time video compression boards in the parallel processing system. The compressed videos will be stored directly to the hard disk of the PCs. In our prototype system, eight Pentium 4 1.8-GHz computers are connected together through a 100 Base-T LAN as shown in Fig. 3(b). At 1.8 GHz, it is possible to perform software-assisted real-time MPEG-2 compression at a resolution of (720 × 480) using the Pinnacle PCTV capturing board. A camera array using eight Sony CCX-Z11 CCD cameras is constructed as shown in Fig. 3(a). The outputs are in NTSC format (525-line interlaced video at 25 f/s) and they are synchronized by modifying the electronics inside the cameras so that they operate on the same clock signal. The spacing between successive cameras is 2.5 cm and four tuning screws are used to control the tilting angles of each camera. Note that all these components are relative inexpensive and the system can readily be extended to include more cameras.

Our system uses closely spaced charged-coupled device (CCD) cameras to reduce problems due to insufficient sampling and to avoid the large imaging variations of CMOS cameras, which usually complicate camera calibration. It is also relatively easy to construct, as it requires only off-the-shelf components and readily available equipment. We have, however, compromised in the number of cameras being used. Another valuable feature of our system is its distributed nature, which allows us to capture, compress, process, and render the plenoptic video efficiently. We believe that parallel processing is essential to handle the demanding storage and computational requirements of plenoptic videos and other dynamic image-based representations. Although our prototype system has a linear configuration, other similar configurations such as (2 × 8) or (3 × 8) are possible and it will improve the viewing freedom of the users, and we believe that the resulting rendering quality will be similar to the one reported here. Due to the use of data compression and parallel processing, our system is reasonably scalable.

B. Camera Calibration

During construction, the cameras were carefully installed to the hardware stand and similar focal lengths and tilting angles were maintained. The cameras were then calibrated using the method in [24]. This method was originally proposed for calibrating a single camera and the relative position of the camera and the viewing angle with respect to a reference grid position can be estimated. More precisely, five images (the grid images) of a certain grid pattern, which consists of squares evenly spaced in a regular grid [Fig. 3(c)], are taken by the camera at five different positions. The corners of the squares in each grid image are then determined in order to recover the intrinsic and extrinsic parameters of the cameras. The videos captured by the cameras are then rectified for rendering. In addition, the relative positions of the cameras can be used in unstructured lumigraph rendering [35]. Before capturing using the Pinnacle PCTV MPEG-2 video capture boards at the eight PCs, the system clocks of the PCs are synchronized through the network.

C. Experimental Results and Comparison

Fig. 4 shows snapshots of two plenoptic videos captured by the system (rectified): *Glass Music Box* and *Crystal Dragon*.[2] They are extracted from a plenoptic video of about half an hour long. In the *Glass Music Box* plenoptic video, a glass music box was placed at the center of the scene and it was rotating at a regular speed. A moving spotlight was used to change dynamically the lighting of the scene. It can be seen from the images that significant lighting changes, reflections, and parallax are captured. The *Crystal Dragon* sequence consists of a lead crystal in the shape of a dragon, which was placed on a wooden platform. Beside it is another crystal turtle, which was placed on a lighting platform that changes color periodically. A burning candle and a moving spotlight were also included to demonstrate the lighting changes and reflective properties of the scene. Since the capturing system is able to handle videos of more than an hour, the two plenoptic videos were taken in a single shot. The distances between the objects and the camera array were also varied to evaluate the effect of camera calibration on rendering quality. Each uncompressed video stream consumes about 30 Gbytes of storage. We have also generated a synthetic plenoptic video, called the “ball sequence,” using computer graphics techniques, which is shown in Fig. 5. It was rendered using the 3-D Studio Max software and the data sets consist of 16 × 1 24-bit RGB videos with 320 × 240 pixels and 24 f/s. Despite the relatively large depth variation, the use of a mean depth in rendering this plenoptic video does not introduce large rendering artifacts. Also, it was observed from the ball sequence that the plenoptic video is not very sensitive to occlusion if the depth variation is not too large. For large depth variations, artifacts in form of ghost images and image blurring will appear and more accurate geometrical information such as depth maps are required [23].

Here, we give a brief account of the systems reported in [19]–[22]. The Stanford Multi-Camera Array Project [19] was probably the first attempt to develop a large-scale camera array and capturing hardware toward the difficult problem of dynamic IBR modeling. It employs low-cost CMOS sensors and dedicated compression hardware. A preliminary six camera-array was reported in [19] and later extend to include more than one

[2] We have also captured a sequence called “Train.” Due to page limitation, the details are omitted and only its rendered result is presented in Section VI-A

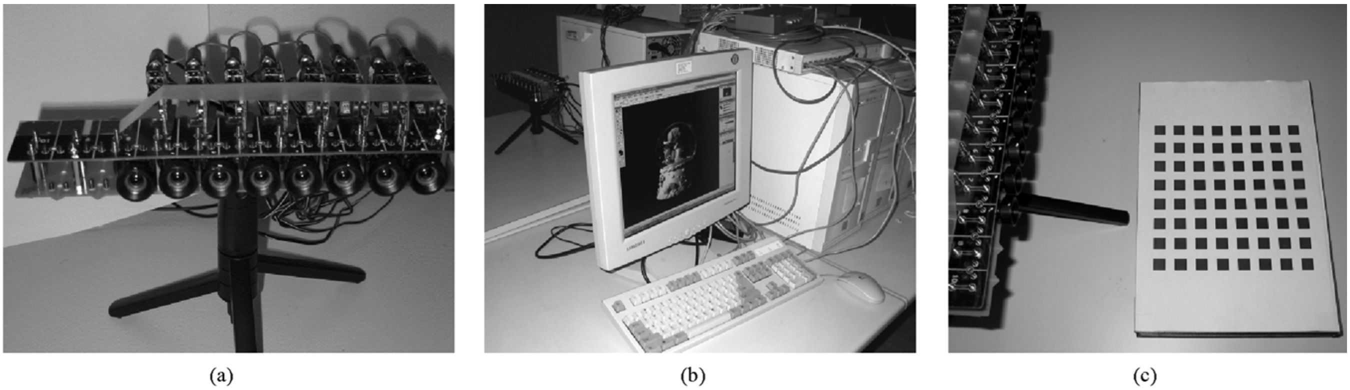

Fig. 3. Plenoptic video system. (a) Eight-camera array. (b) Capturing system with eight PCs connected by a 100 Base-T LAN. (c) Calibration pattern.

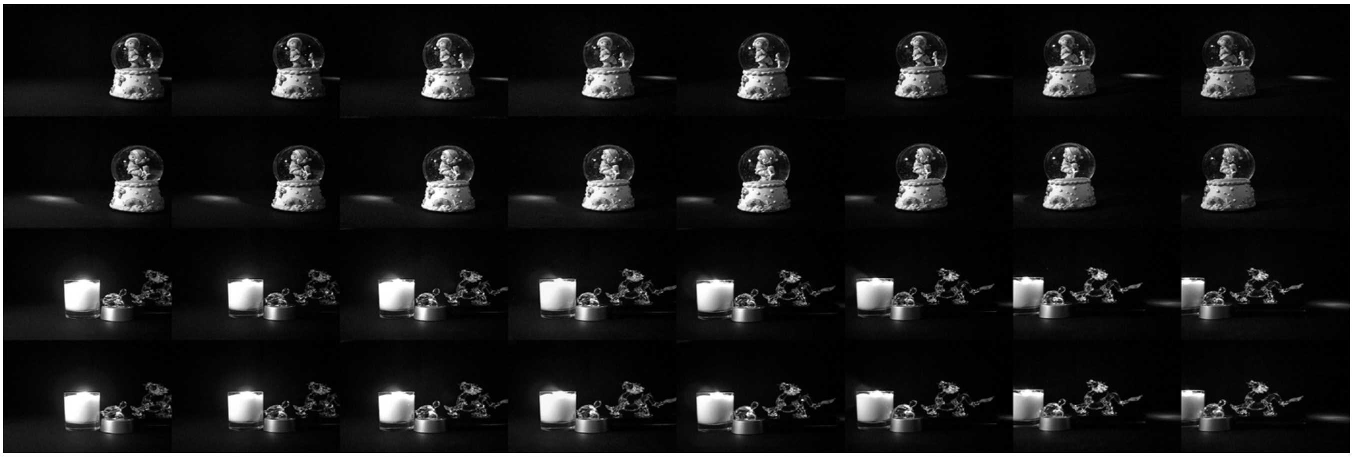

Fig. 4. Four snapshots of the plenoptic video *Glass Music Box* and *Crystal Dragon*. Each row consists of the eight images taken from the cameras (form left to right) at a given time instant.

hundred cameras. Subsequently, B. Goldlücke *et al.* [22] used the video sequences captured by the (3×2) array in [19] to investigate the rendering of light field videos. The images with a resolution of 320×240 from the six cameras are warped and blended, according to a pre-computed disparity map, to synthesize the novel views at (640×480) resolution with approximately 14 f/s $(\text{block size} = 4)$. They can handle larger depth variations but also potentially introduce artifacts due to inaccurate depth estimation. Since the objectives and design tradeoffs of our system and that in [19] are quite different, the results cannot be compared directly. One advantage of our system is its good rendering quality. This is largely attributed to the higher resolution and better quality of the CCD sensors,[3] smaller camera spacing, and camera calibration employed in our system. While the pioneer work in [19] is more concerned with large-scale modeling, our system is targeted toward modeling video objects and it has applications as electronic brochures and demonstration clips, as mentioned earlier.

In [25], the image pixels for rendering a given view are retrieved using hardware from an array of CMOS imaging sensors in order to avoid the high data rate for online rendering. The system of Naemura *et al.* [20] consists of 16 closely spaced CCD cameras in a (4×4) 2-D arrangement (can be reconfigured to a linear array similar to ours). It does not incorporate real-time data compression such as MPEG compression used in our system. Instead, dedicated processors (Sony YS-Q430) are used to combine the video sequences from four cameras to form a video sequence divided into four screens. Therefore, the resolution is significantly reduced because of the bandwidth constraint. The final rendered view, using an Onyx2 workstation, has a resolution of only 180×120. On the other hand, our captured plenoptic videos have a resolution of 720×480 pixels at 30 f/s. No camera calibration is performed in [20] and motion parallax is suppressed using linear translation operations. One distinct feature of this system is the use of a real-time depth map estimation board from Komatsu, FZ930 board $(280 \times 200$ pixels, 8-bits depth map) at 30 f/s, to divide the image into 3 layers for rendering (10 f/s at 180×120 resolution). Currently, our system does not address scenes with large depth variation because it is still a very challenging problem to estimate depth maps reliably. By limiting the depth variation and using light field rendering with a single mean depth (the approximate sampling analysis in Section III-B) alone, our system achieves fairly high-quality real-time rendering of raw video data at a resolution of 720×480 with 15 f/s. For rendering from compressed data, the resolution is reduced to 256×256, due to limitation of processing power and transmission bandwidth over the 100 Base-T network without transcoding.

[3] Our CCD sensor (USD$100) is about two times more expensive than the CMOS sensor in [19], with better color response and resolution.

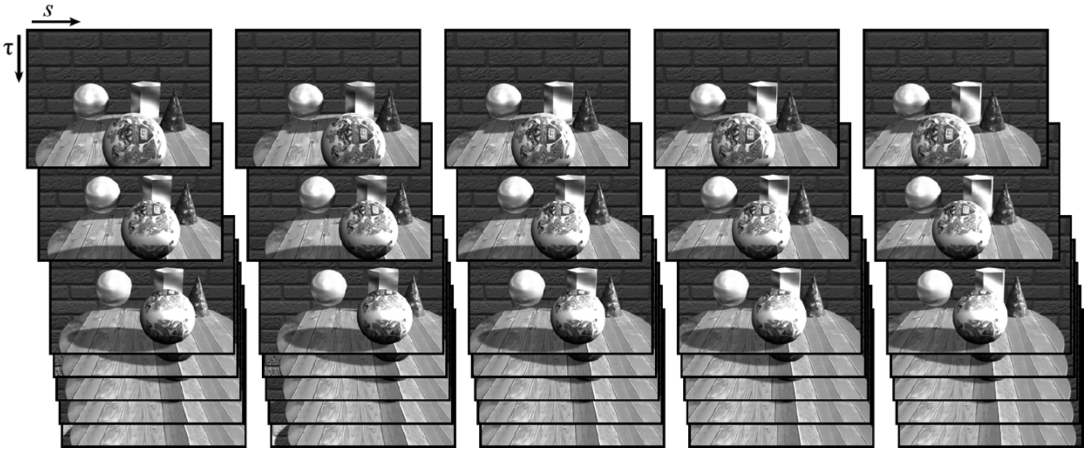

Fig. 5. Snapshots of the *Ball* sequence (only images from five virtual cameras are shown).

TABLE I
COMPARISON OF OUR SYSTEM WITH THE STATE-OF-THE-ART SYSTEMS IN [20], [21], AND [22]

	Plenoptic Video	**[20]**	**[21]**	**[22]**
Sensor	¼ inch color CCD (CCX-Z11)	¼ inch color CCD (XC-333)	¼ inch color CCD (iBOT)	CMOS (OV7620)
Data Resolution	720×480 30 f/s	720×480 30f/s (will be decimated due to no compression)	640×480 30 f/s	320×240 30 f/s
Rendering Resolution	From raw data 720×480 15f/s, streaming 256×256 15 f/s.	180×120 10 f/s (3 depth layers)	320×240 18f/s	640×480 13.5 f/s (block size 4)
Hardware Required	8×1 cameras, 2.5 cm apart. 9 Pentium 4 PCs, 8 Pinnacle PCTV boards, 100 Base-T LAN.	16 (4×4) cameras 3.1 cm apart. Dedicated processors (Sony YS-Q430), Onyx2 workstation (4 400MHz R12000) with a DIVO, real-time depth map estimation board, Komatsu, FZ930.	8×8 cameras, farther apart than [20]. 7 Pentium 4 PCs connected by firewire.	3×2 demo array. Approx. 40cm between cameras. Embedded microprocessor board, MPEG-2 video encoder, IEEE1394 interface to Ultra160 SCSI disk drives.
Rendering Method	Light field rendering with mean depth. Can be extended to include depth map.	A 3-layer depth map is used to blend the images for rendering.	Images divided into rectangular and rendered using different focal plane.	Images divided into meshes and are warped and blended using dense disparity map.
Camera Calibration	Zhang's algorithm [24] for all cameras using calibration patterns. White balancing for color correction.	Nil	Zhang's algorithm [24] for one camera and others with structure from motion algorithm. Might need manual color control adjustment.	Camera's color reproduction by calibration matrices.
Real Time Compression for Storage	MPEG-2 (1:240 compression ratio). Streamed to harddisk of individual PCs. Modified MPEG-2 algorithm for efficient storage and transmission.	For interactive image-based rendering. Very low resolution.	Nil. For interactive image-based rendering.	MPEG-2 compression (5 Mbytes /sec per video). Stream to host PC's SCSI hard drive.
Applications	Interactive 3D electronic catalog or brochure, short advertising clips, head and shoulder-type 3D videophone.	Interactive image-based rendering.	Interactive image-based rendering.	3D movie for large environment.

The (8 × 8) light field camera of Yang *et al.* [21] is mainly designed for interactive IBR. Unlike the Stanford light field camera and our system, all the videos from the video cameras are not recorded or stored due to difficulties in compressing the videos in real-time. Images from the cameras are divided into fragments and those fragments required to synthesize a given view are transmitted to a compositor for rendering. It is impossible to replay the videos as in our system, which resembles a traditional video system with continuous multiple viewpoints along a trajectory. The camera spacing is also very small to avoid aliasing. Camera calibration is done by first calibrating one of the cameras using Zhang's algorithm [24]. The rest of the cameras are calibrated using a structure from motion algorithm. Finally, a large nonlinear optimization is performed to cater for nonidentical intrinsic parameters of the cameras. Although our algorithm is also based on Zhang's algorithm, we use a larger calibration pattern for calibrating simultaneously all the cameras. Since no data is provided in [21], we are unable to compare

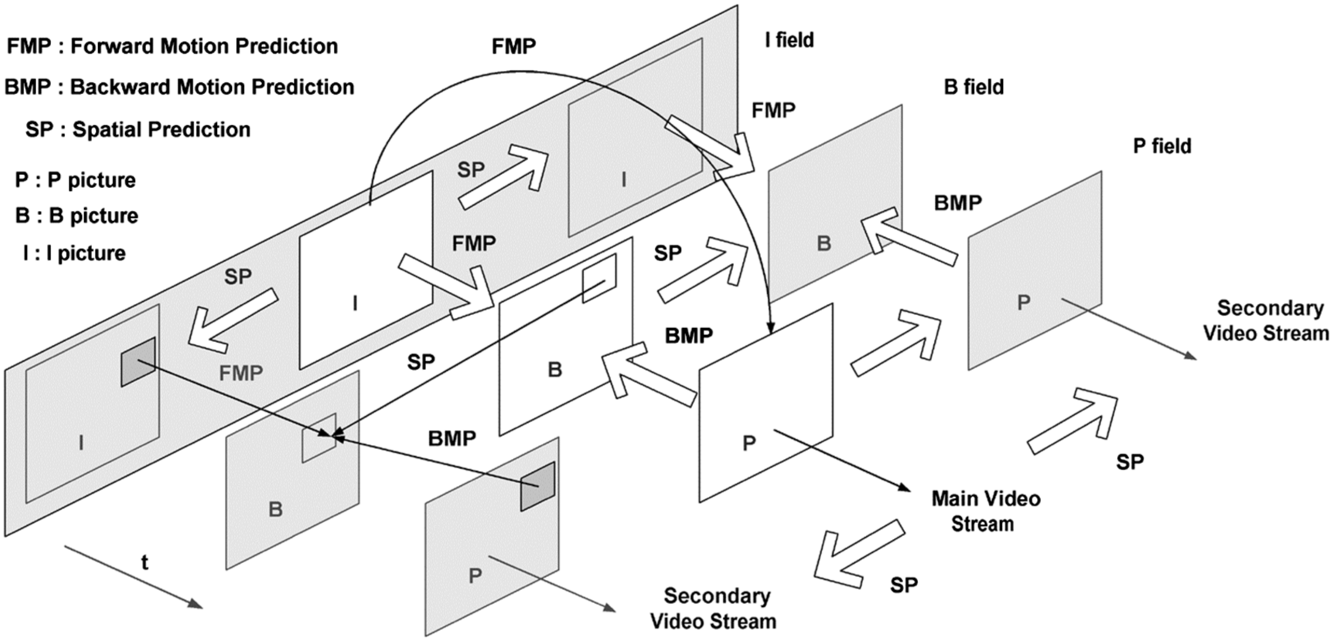

Fig. 6. Compression of the 4-D plenoptic video.

directly the accuracy of both methods. Manual color control adjustment in some of the sensors is necessary in [21]. The main features of the above systems are summarized in Table I.

V. Compression

A. Overview

In general, there are two approaches to reduce the data size of image-based representations [17]. The first one is to reduce their dimensionality, often by limiting viewpoints or sacrificing some realism. Light fields and Concentric Mosaics are such examples. The second approach is more classical, namely, to exploit the correlation (i.e., redundancy) within the representation using waveform coding or model-based techniques. The scene geometry might be utilized explicitly or implicitly. The second approach can further be classified into three broad categories: pixel-based methods [6], [10], disparity compensation/prediction methods [12], [15] and model-based/model-aided methods [26], [27]. Disparity compensation has been used in coding stereoscopic and multiview images [28]–[33]. Interested readers may refer to a survey paper in [17] for more detail. In this paper, an MPEG-2 like algorithm with temporal and disparity compensation is employed because of its good performance and relatively low complexity. Spatial prediction or disparity compensated prediction has been used in coding of static light fields [12]–[14] and stereo image coding [28], [29]. The coding algorithm considered here can be viewed as their generalization to the dynamic situation.

Providing *random access* to the compressed data for real-time rendering and efficient methods for exploring the redundancy are important problems in the compression of plenoptic videos [17]. It is because higher dimensional image-based representations such as the four-dimensional (4-D) light field, Lumigraph and the plenoptic video require random access at the pixel level. Since most existing compression algorithms employ entropy coding for better compression efficiency, the symbols after compression are of variable sizes. It is time-consuming to retrieve a single line or pixel directly from the compressed data. With efficient random access mechanisms, such as pointers to the compressed data stream, *selective decoding* [15], [16] or just-in-time (JIT) decoding [12] can be employed to decode on-line those pixels which are required for rendering. Similar problems exist in the transmission of plenoptic videos. This will be addressed in Section VI using the concept of selective transmission/reception and parallel processing techniques.

B. MPEG-2 Like Algorithm With Temporal and Disparity Compensation

As video streams in a plenoptic video are taken at nearby positions in a one-dimensional (1-D) array, they appear to be shifted relative to each other, because of the disparity of image pixels. In order to explore this correlation in the plenoptic video, we divide the video streams into groups and compress them together using both temporal and disparity compensations. The proposed compression method is shown in Fig. 6. Only three videos are shown for simplicity, and it is called a group of fields (GOF). To provide *random access to individual pictures*, we have adopted a modified MPEG-2 video compression algorithm [34] to encode the image frames. There are two types of video streams in the proposed dynamic light field: *main* and *secondary* video streams. Main video streams are encoded using the MPEG-2 algorithm, which can be decoded without reference to other video streams. The image frames in a main stream are divided into I-, P-, and B-pictures, where I-pictures are coded using intra-frame DCT-based transform coding, while P-pictures are coded by hybrid motion compensated/transform coding using previous I- or P-pictures as references. B-pictures are coded by a similar method except that forward and backward motion compensation is performed by using nearby I- or P-pictures as references which is indicated

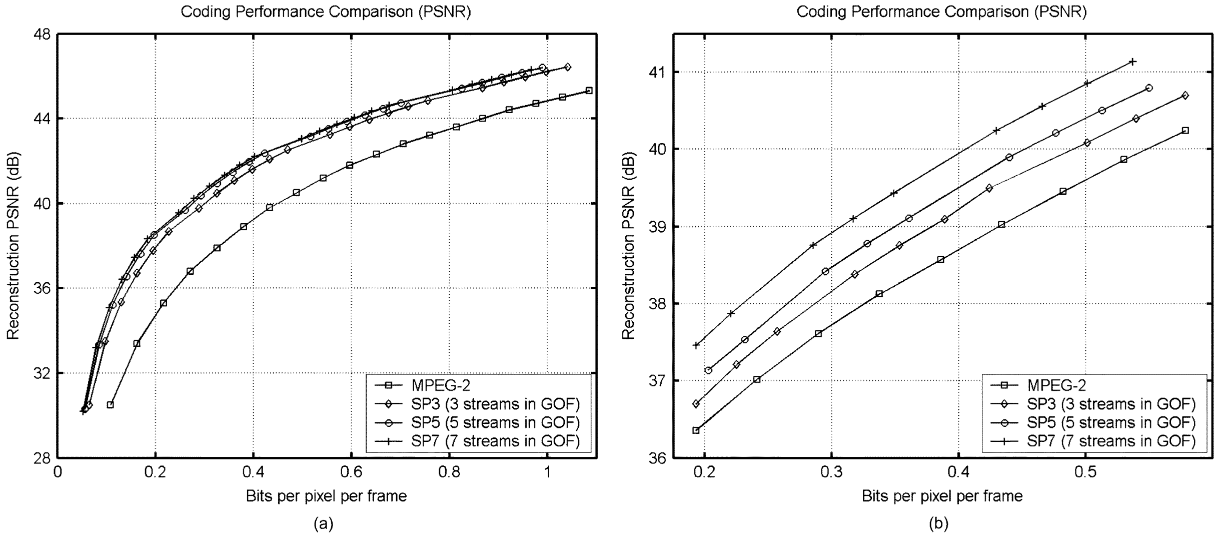

Fig. 7. Coding results of the plenoptic videos. (a) Synthetic *Ball* sequence. (b) *Glass Music Box* sequence.

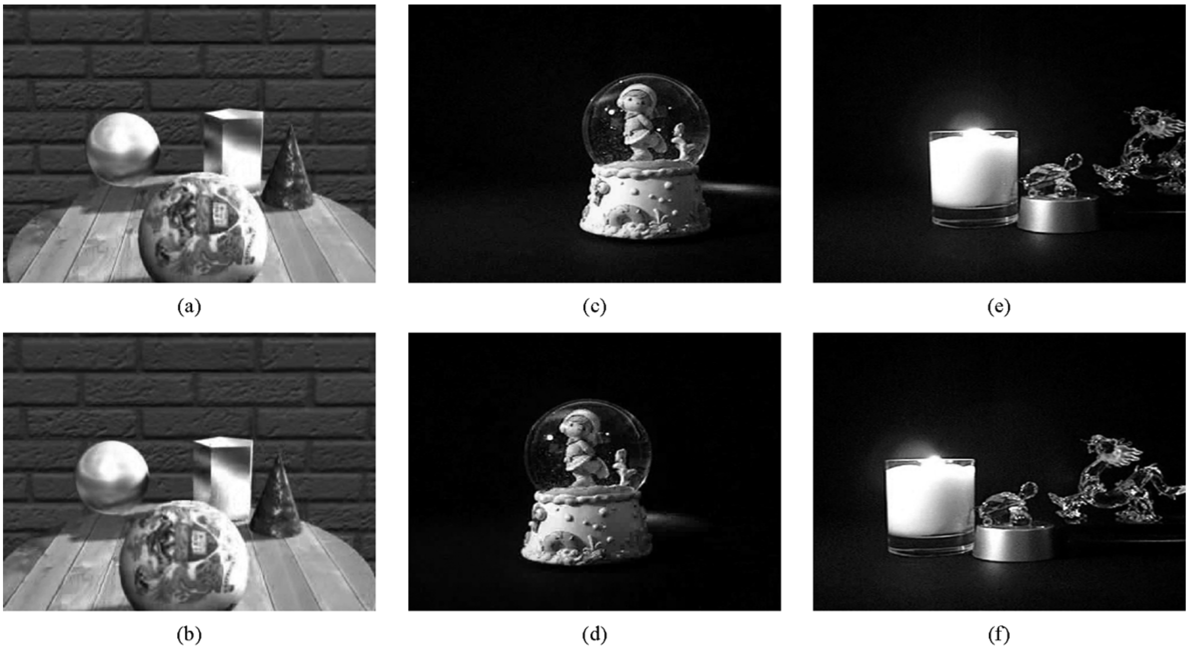

Fig. 8. Typical reconstructed images. The *Ball* sequence in the (a) main and (b) secondary video streams (194 kb/s per stream). The *Glass Music Box* sequence in the (c) main and (d) secondary video streams (583 kb/s per stream). The *Crystal Dragon* sequence in the (e) main and (f) secondary video streams (624 kb/s per stream).

by the block arrow in Fig. 6. The images captured at the same time instant as the I-pictures in a main stream constitute an I-field. Similarly, we define the P- and B-fields as the images containing, respectively, the P- and B-pictures of the main video stream. Pictures from the secondary stream in the I-field are encoded using *spatial prediction* from the reference I-picture in the I-field. It is because adjacent images appear to be shifted relative to each other, similar to the effect of linear motion in video coding. Pictures from the secondary stream in a P-field are predicted using spatial prediction from adjacent P-pictures in the main stream, and the forward motion compensation from the reference I- or P-fields in the same secondary stream. Pictures from the secondary stream in B-field are predicted using spatial prediction from adjacent B-picture in the main stream, and the forward/backward motion compensation from nearby reference I- and/or P-fields in the same secondary stream.

For simplicity, we have only included one main stream in each GOF. More sophisticated disparity compensation schemes such as bi-directional prediction with multiple main streams can be incorporated in a single GOF or successive group of blocks (GOBs). Our scheme can also be generalized to 2-D GOFs in the compression of 5D dynamic light fields, with main streams distributed on certain points in the 2-D array, instead of a 1-D array considered here. In order to maintain a more uniform reconstruction quality among the plenoptic videos, we allocate a higher bit rate to the main streams than the secondary streams because the I-pictures in the main streams usually require considerably more bits than P- and B-pictures. Furthermore, the rate

TABLE II
NUMBER OF MACROBLOCKS USED FOR DIFFERENT TYPES
(THE SYNTHETIC BALL SEQUENCE)

	Main Stream	Secondary Stream		
		d = 1	d = 2	d = 3
97 kbps				
Intra MB	9.3%	0.1%	0.2%	0.3%
Temporary MB	90.7%	44.5%	49.0%	50.9%
Spatial MB	0.0%	55.4%	50.8%	48.8%
743 kbps				
Intra MB	9.0%	0.1%	0.2%	0.2%
Temporary MB	91.0%	62.1%	66.8%	67.4%
Spatial MB	0.0%	37.8%	33.0%	32.4%
1.78 Mbps				
Intra MB	9.0%	0.1%	0.2%	0.2%
Temporary MB	91.0%	64.7%	69.4%	70.4%
Spatial MB	0.0%	35.2%	30.4%	29.4%

control algorithm of the MPEG-2 Test Model 5 is used to prevent buffer overflow and underflow problems, although other more sophisticated rate control algorithms can also be applied. To address the random access problem, *pointers* are embedded into the compressed data stream as in [15] and [16]. During rendering, the required macroblocks will be selectively decoded from the compressed data streams. This adds to the overhead in the compressed data streams.

C. Experimental Results

The proposed compression algorithm is evaluated using the *Glass Music Box*, the *Crystal Dragon*, and the synthetic sequence *Ball*. The *Glass Music Box* and *Crystal Dragon* plenoptic videos consist of 8×1 24-bit RGB videos with 720×480 pixels. Coding results for different numbers of video streams in a GOF are investigated, and they are plotted in Fig. 7. For SP3, we have three video streams in the GOF as illustrated previously in Fig. 6. For SP5 and SP7, we have five and seven video streams, respectively. As a comparison, we also compressed all the video streams of the synthetic and real plenoptic videos by the MPEG-2 algorithm independently. It can be seen that the proposed algorithm using both temporal and disparity compensation shows significant improvement over the independent coding scheme. This shows that there is a significant amount of spatial redundancy among the video sequences. When the number of video streams in the GOF, and hence the number of secondary streams, is increased, the peak signal-to-noise ratio (PSNR) improves because less I-pictures are coded and better disparity prediction is obtained in the plenoptic video. However, the difference between SP5 and SP7 is small because disparity compensation will be less effective when video streams are far apart. Fig. 8 shows several typical reconstructed images. They show good quality of reconstruction at: 194 kb/s per stream [0.105 bits per pixel per frame (bpp/f)] for the synthetic *Ball* sequence; 583 kb/s per stream for the *Glass Music Box* sequence (0.070 bpp/f) and 624 kb/s per stream for the *Crystal Dragon* sequence (0.075 bpp/f).

In order to study the performance of spatial prediction, we show in Table II the number of macroblocks using different prediction types. At a bit rate of 1.78M b/s per stream, secondary video streams which are closest to the main video streams ($d = 1$) have 35.2% of their macroblocks predicted by disparity compensation prediction. When the distance (d) from the main stream increases, the prediction is increasingly difficult and fewer macroblocks are predicted spatially. This situation might be improved by using bi-directional disparity compensation prediction. The rate drops to 29.4% when the distance is increased to 3. Furthermore, it is noted that this percentage depends on the target bit rate. For example, when we decrease the bit rate, more macroblocks (up to 50%) will employ spatial prediction.

VI. RENDERING

There are several major consideratons and challenges in the real-time rendering of plenoptic videos. Due to the difficulties in controlling the positions of the image sensors inside the cameras, the optical centers of the cameras do not usually lie on a straight line or even on the same plane. This problem is less serious in capturing static light fields where the relative positions of the camera can be accurately controlled. Fortunately, the relative positions of the cameras can still be recovered from the camera calibration described in Section IV-B. Since the coordinates calculated do not lie on a straight line, unstructured lumigrah rendering as proposed in [35] has to be used. In our experiments, we found that the geometric distortion and the rotation of the cameras could be satisfactorily compensated, partially because of the manual adjustment of the cameras prior to the capturing. The second problem concerns the artifacts encountered due to the incorrect depth estimation. For the *Glass Music Box* and *Crystal Dragon* sequences, the depth variation is relatively small and according to the plenotpic sampling analysis [23], the rendering artifacts will be small as long as the focus plane is chosen as the mean depth of the scene. For more complicated scenes, more geometry information would be required. The final problem is the real-time rendering of the plenoptic video. If the plenoptic video is decoded into raw images and stored on a hard disk, real-time rendering can readily be achieved. However, the memory requirement is very large and the playback time is limited. If the plenoptic video is rendered from the compressed bit stream, then even with the use of selective decoding the computational requirement for the decoding and rendering is very large. The basic idea of selective transmission/rendering is to decode in parallel the multiple streams of the videos in a network of computers, and transmit those pixels required to the rendering machine over the network, possibly with simple compression. This offloads the rendering machine at the expense of longer user response time. However, we believe that selective transmission is essential to the distribution of plenoptic video in future applications.

A. Experimental Results

Using selective transmission, we are able to stream continuously plenoptic video with (256×256) resolution at a rate of 15 f/s over the network. Due to network delay, there is a slight

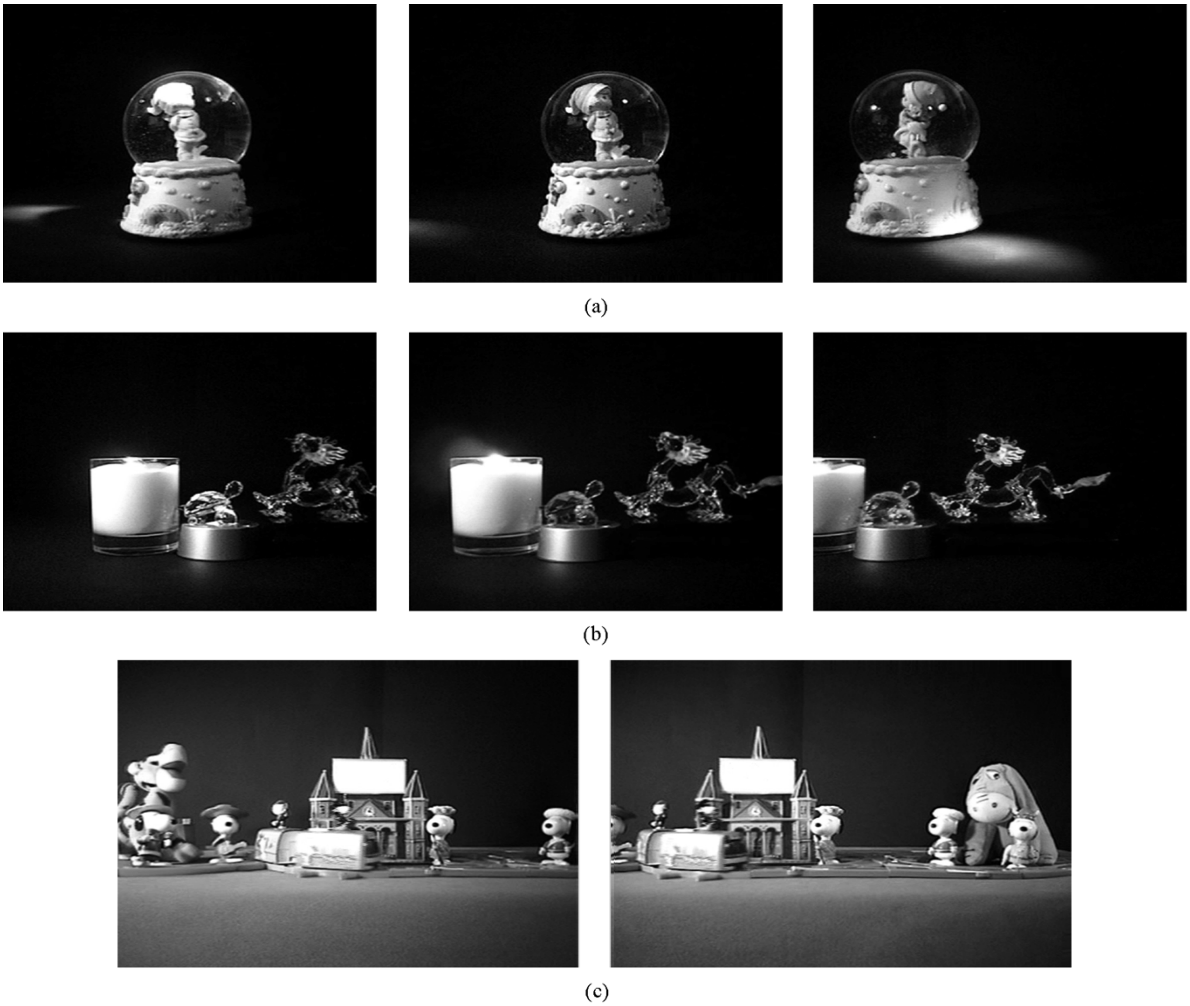

Fig. 9. Renderings from the real-time plenoptic video render. (a)-(b) Three virtual views at two different time instants for the *Glass Music Box* and the *Crystal Dragon* sequences, respectively. (c) Two virtual views at two different time instants for the *Train* sequence (pictures at the same row are views rendered at the same time instant).

delay in the user response. The frame rate and the resolution can be increased if the raw data stream is compressed by simple coding method such as vector quantization. For rendering from raw data on the hard disk, real-time rendering can be achieved with a resolution of (720 × 480) and a rate of 15 f/s. Fig. 9 shows several virtual views rendered from the *Glass Music Box*, the *Crystal Dragon* and the *Train* plenoptic videos.[4] It can be seen that the lighting changes and reflective properties of the glass and lead crystal are well captured. The *Train* sequence demonstrates that scenes with more complicated details, occlusion, and moving objects (the toy train in the middle) can be rendered with reasonably good quality. It was found that slight artifacts, in the form of ghosting and blurring, are still present in some of the rendered images, because of the difficulty in determining exactly the camera positions and inaccurate depth values. It was also found that the artifacts are less noticable if the objects are farther away from the camera planes because of the reduced resolution of the images as well as the reduced sensitivity of the image pixels to the errors due to camera calibration and depth values.

VII. Conclusion and Future Work

We have presented a novel system for capturing, compression and rendering of SDLF, called the plenoptic videos. By appropriate system design, we have demonstrated that dynamic image-based representations of high dimensionality can be captured and processed using off-the-shelf components and readily available equipment. Methods for calibrating multiple cameras and compressing video data in the plenoptic video system were also developed. This provides much insight and experience into the development of and experimentation with other dynamic IBR capturing systems. To handle more complicated scenes and achieve a better quality, depth information/correction in form of dense depth map [36] or image layers with constant depth as in the pop-up light field [37] can be incorporated. Finally, we hope the experience and findings in this work will facilitate further development and widespread use of dynamic image-based representations as an efficient means for visualization, especially for 3-D immersive TV systems.

[4] Note the good color quality of the sequences because of the use of CCD sensors.

ACKNOWLEDGMENT

The authors would like to thank Mr. L. C. Koo for building the camera system. The assistance of Dr. X. Tong and Dr. X. Liu from Microsoft Research Asia in rendering the plenoptic video is highly appreciated. The help of Ms. P. K. Chan in preparing the demonstration is gratefully acknowledged. The authors would also like to thank Dr. Y. S. Hung for proofreading an initial draft of this paper and useful discussions.

REFERENCES

[1] E. H. Adelson and J. Bergen, "The plenoptic function and the elements of early vision," in *Computational Models of Visual Processing*. Cambridge, MA: MIT Press, 1991, pp. 3–20.

[2] D. G. Aliaga and I. Carlbom, "Plenoptic stitching: A scalable method for reconstructing 3-D interactive walkthroughs," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'01)*, Aug. 2001, pp. 443–450.

[3] S. E. Chen and L. Williams, "View interpolation for image synthesis," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'93)*, Aug. 1993, pp. 279–288.

[4] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry-and image-based approach," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'96)*, Aug. 1996, pp. 11–20.

[5] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'96)*, Aug. 1996, pp. 43–54.

[6] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'96)*, Aug. 1996, pp. 31–42.

[7] L. McMillan and G. Bishop, "Plenoptic modeling: An image-based rendering system," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'95)*, Aug. 1995, pp. 39–46.

[8] S. M. Seitz and C. M. Dyer, "View morphing," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'96)*, Aug. 1996, pp. 21–30.

[9] J. Shade, S. Gortler, L. W. He, and R. Szeliski, "Layered depth images," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'98)*, Orlando, FL, Jul. 1998, pp. 231–242.

[10] H. Y. Shum and L. W. He, "Rendering with concentric mosaics," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'99)*, Los Angeles, CA, Aug. 1999, pp. 299–306.

[11] R. Szeliski and H. Y. Shum, "Creating full view panoramic image mosaics and environment maps," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'97)*, Aug. 1997, pp. 251–258.

[12] J. Li, H. Y. Shum, and Y. Q. Zhang, "On the compression of image based rendering scene: a comparison among block, reference and wavelet coders," *Int. J. Image Graph.*, vol. 1, no. 1, pp. 45–61, 2001.

[13] M. Magnor and B. Girod, "Data compression for light-field rendering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 3, pp. 338–343, Apr. 2000.

[14] X. Tong and R. M. Gray, "Coding of multi-view images for immersive viewing," in *Proc. IEEE ICASSP 2000*, vol. 4, Jun. 2000, pp. 1879–1882.

[15] H. Y. Shum, K. T. Ng, and S. C. Chan, "Virtual reality using the concentric mosaic: Construction, rendering and data compression," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, Sep. 2000, pp. 644–647.

[16] K. T. Ng, S. C. Chan, H. Y. Shum, and S. B. Kong, "On the data compression and transmission aspects of panoramic video," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Oct. 2001, pp. 105–108.

[17] H. Y. Shum, S. B. Kang, and S. C. Chan, "Survey of image-based representations and compression techniques," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 11, pp. 1020–1037, Nov. 2003.

[18] Z. F. Gan, S. C. Chan, K. T. Ng, and H. Y. Shum, "An object-based approach to plenoptic videos," in *Proc. IEEE Int. Symp. Circuits and Systems*, Kobe, Japan, 2005, pp. 3435–3438.

[19] B. Wilburn, M. Smulski, K. Lee, and M. Horowitz, "The light field video camera," in *Proc. SPIE Electronic Imaging: Media Processors*, vol. 4674, Jan. 2002, pp. 29–36.

[20] T. Naemura, J. Tago, and H. Harashima, "Real-time video-based modeling and rendering of 3-D scenes," *IEEE Computer Graph. Applicat.*, pp. 66–73, Mar.–Apr. 2002.

[21] J. C. Yang, M. Everett, C. Buehler, and L. McMillan, "A real-time distributed light field camera," in *Proc. Eurographics Workshop on Rendering*, 2002, pp. 77–86.

[22] B. Goldlücke, M. Magnor, and B. Wilburn, "Hardware-accelerated dynamic light field rendering," in *Proc. Vision, Modeling, and Visualization (VMV-2002)*, Erlangen, Germany, Nov. 2002, pp. 455–462.

[23] J. X. Chai, X. Tong, S. C. Chan, and H. Y. Shum, "Plenoptic sampling," in *Proc. Annu Conf. Computer Graphics (SIGGRAPH'00)*, Jul. 2000, pp. 307–318.

[24] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[25] R. Ooi, T. Hamamoto, T. Naemura, and K. Aizawa, "Pixel independent random access image sensor for real time image-based rendering system," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Oct. 2001, pp. 193–196.

[26] M. Magnor and B. Girod, "Model-aided coding of multi-viewpoint image data," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, Sep. 2000, pp. 919–922.

[27] ——, "Model-based coding of multi-viewpoint imagery," *SPIE Visual Commun. Image Process.*, vol. 4067, no. 2, pp. 14–22, Jun. 2000.

[28] M. E. Lukacs, "Predictive coding of multi-viewpoint image sets," in *Proc. IEEE ICASSP*, 1986, pp. 521–524.

[29] J. R. Ohm, "Stereo/multiview encoding using the MPEG family of standards," in *Proc. Electronic Imaging*, San Diego, CA, Jan. 1999, pp. 242–253.

[30] A. Puri, R. V. Kollarits, and B. G. Haskell, "Basics of stereoscopic video, new compression results with MPEG-2 and a proposal for MPEG-4," *J. Signal Process.: Image Commun.*, vol. 10, pp. 201–234, 1997.

[31] T. Naemua, M. Kaneko, and H. Harashima, "Compression and representation of 3-D images," *IEICE Trans. Inf. Syst.*, vol. E82-D, no. 3, pp. 558–567, 1999.

[32] J. R. Ohm, "Encoding and reconstruction of multiview video objects: Looking at data compression in the context of the MPEG-4 multimedia standard," *IEEE Signal Process. Mag.*, vol. 16, no. 3, pp. 47–54, May 1999.

[33] M. G. Strintzis and S. Malasiotis, "Object-based coding of stereoscopic and 3-D image sequences: A review," *IEEE Signal Process. Mag.*, vol. 16, no. 3, pp. 14–29, May 1999.

[34] *Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Video*, Nov. 1994. ITU-T Recommendation H.262/ISO/IEC 13 818-2.

[35] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen, "Unstructured lumigraph rendering," in *Proc. Annu. Conf. Computer Graphics (SIGGRAPH'01)*, Aug. 2001, pp. 425–432.

[36] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vis.*, vol. 47, no. 1, pp. 7–42, May 2002.

[37] H. Y. Shum, J. Sun, S. Yamazaki, Y. Li, and C. K. Tang, "Pop-up light field: An interactive image-based modeling and rendering system," *ACM Trans. Graph.*, vol. 23, no. 2, pp. 143–162, Apr. 2004.

[38] S. C. Chan, K. T. Ng, Z. F. Gan, K. L. Chan, and H. Y. Shum, "The compression of simplified dynamic light fields," in *Proc. IEEE ICASSP* , vol. 3, Apr. 2003, pp. 653–656.

[39] ——, "The plenoptic videos: Capturing, rendering and compression," in *Proc. IEEE ISCAS*, vol. 3, May 2004, pp. 905–908.