An Object-Based Approach to Image/Video-Based Synthesis and Processing for 3-D and Multiview Televisions

Shing-Chow Chan, *Member, IEEE*, Zhi-Feng Gan, *Student Member, IEEE*, King-To Ng, *Member, IEEE*,
Ka-Leung Ho, *Senior Member, IEEE*, and Heung-Yeung Shum, *Fellow, IEEE*

Abstract—**This paper proposes an object-based approach to a class of dynamic image-based representations called "plenoptic videos," where the plenoptic video sequences are segmented into image-based rendering (IBR) objects each with its image sequence, depth map, and other relevant information such as shape and alpha information. This allows desirable functionalities such as scalability of contents, error resilience, and interactivity with individual IBR objects to be supported. Moreover, the rendering quality in scenes with large depth variations can also be improved considerably. A portable capturing system consisting of two linear camera arrays was developed to verify the proposed approach. An important step in the object-based approach is to segment the objects in video streams into layers or IBR objects. To reduce the time for segmenting plenoptic videos under the semiautomatic technique, a new object tracking method based on the level-set method is proposed. Due to possible segmentation errors around object boundaries, natural matting with Bayesian approach is also incorporated into our system. Furthermore, extensions of conventional image processing algorithms to these IBR objects are studied and illustrated with examples. Experimental results are given to illustrate the efficiency of the tracking, matting, rendering, and processing algorithms under the proposed object-based framework.**

Index Terms—**Dynamic image-based representations, image-based rendering (IBR), object-based, plenoptic videos.**

I. Introduction

Multiview imaging (MVI) has attracted great attention recently due to its increasingly wide range of applications and the decreasing cost of digital cameras. This opens up many new and interesting research topics as well as applications, such as virtual view synthesis for three-dimensional (3-D) television (3-DTV) and entertainment, high-performance imaging, video processing and analysis for surveillance, distance learning, industry inspection, etc.

One of the most important applications in MVI probably is the development of advanced immersive viewing or visualization systems using, say, 3-D or multiview TVs. With the introduction of multiview TVs, it is expected that a new age of 3DTV systems will arrive in the nearest future. To realize these goals, however, there are still many new and challenging issues to be addressed. In particular, multiview systems normally require large amounts of storage and are considerably difficult to construct. Of more importance still, the various cameras in the camera array usually have very different characteristics and positions, which make the synthesis of virtual view (multiview synthesis) difficult.

Image-based rendering (IBR) refers to a collection of techniques and representations that allow 3-D scenes and objects to be visualized in a realistic way without the full 3-D model reconstruction. Since IBR uses images as the primary substrate, its potential for photorealistic visualization has tremendous appeal. It is not surprising that IBR has received increasing attention recently. In IBR [1]–[15], new views of scenes are reconstructed from a collection of densely sampled images or videos. The reconstruction problem (i.e., rendering) is treated as a multidimensional sampling problem, where new views are generated from densely sampled images and depth maps instead of building an accurate 3-D model of the scene. Since the data size associated with the image-based representations is usually very large, especially in the case of dynamic scenes, capturing, compression, and effective rendering are fundamental problems in IBR research [6], [7]. Different image-based representations have been proposed to simplify the capturing process and storage requirements. For a recent survey of IBR, readers are referred to [7] for more details. In previous works [13], [14], a class of dynamic image-based representations called the "plenoptic video" was proposed for dynamic scenes. The plenoptic video is a simplified light field for dynamic environments. It is obtained by capturing videos that are regularly placed along a series of line segments, instead of a 2-D plane, in the static light fields. The main motivation is to reduce the high dimensionality and excessive hardware cost in capturing dynamic representations. Despite the employed simplification, plenoptic videos can still provide a continuum of viewpoints, significant parallax, and lighting changes along line segments joining the camera arrays.

A difficult problem in rendering light fields and plenoptic videos is the excessive artifacts due to depth variations. If

Manuscript received January 6, 2007; revised June 30, 2007. First version published March 16, 2009; current version published June 19, 2009. This work was supported in part by Hong Kong Research Grant Council (RGC) and the Innovation and Technology fund (ITF). Parts of this work were presented at the Conference of 2005 IEEE International Symposium on Circuits and Systems and the conference of Visual Communications and Image Processing 2005. This paper was recommended by Associate Editor J. Ostermann.

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

H. Y. Shum is with the Microsoft Corporation, Redmond, WA 98052-7329 USA (e-mail: hshum@microsoft.com).

Digital Object Identifier 10.1109/TCSVT.2009.2017302

1051-8215/$25.00 © 2009 IEEE

the scene is free of occlusions, then the concept of plenoptic sampling [8] can be applied to determine the sampling rate in the camera plane. Unfortunately, because of depth discontinuities around object boundaries, the sampling rate is usually insufficient and significant, rendering artifacts due to occlusion are observed. Moreover, appropriate mean depths for objects have to be determined to avoid blurring within the objects and ghosting at the boundaries. Thus, depth segmentation or some kind of depth information is necessary in order to improve the rendering quality. Motivated by Gortler *et al.*'s work on lumigraph [4] and the layered depth images of Shade [5], we assume that each image pixel in a light field has a color as well as a depth value. Instead of using a global depth map, this representation, which can be viewed as local depth images between successive cameras, is less sensitive to errors in camera position and depth maps encountered in practical multicamera systems. Due to the limited amount of information that we can gather from images and videos, a very high resolution depth map is usually unavailable. Besides, the data rate of these detailed depth map sequences is very high. Fortunately, plenoptic sampling tells us that the dense sampling of image-based representation will tolerate this variation within the segments by interpolating the plenoptic function. In other words, it is highly desirable to focus on objects with large depth discontinuities. By properly segmenting the videos into objects at different depths, the rendering quality in a large environment can be considerably improved, as demonstrated by the pop-up light fields [9].

These observations motivate us to develop an object-based approach to plenoptic videos, where the plenoptic video sequences are segmented into IBR objects, each with its image sequence, depth map, and other relevant information such as shape information [15]. Therefore, desirable functionalities such as scalability of contents, error resilience, and interactivity with individual IBR objects (including random access at the object level) can be incorporated. For example, IBR objects can be processed, rendered, compressed, and transmitted separately. An important step in the object-based approach is to segment objects in plenoptic video sequences into layers or IBR objects with different depth values. To reduce the segmentation time and improve reliability, one possibility is to obtain an initial segmentation of the video at key views using semiautomatic tools [18] and rely on tracking techniques to segment the objects at different views and at subsequent time instants. Towards this end, an automatic object tracking approach using the level-set method is proposed in this paper [16], [17]. Our method, which utilizes local and global features of the image sequences and depth information, instead of global features exploited in [19], can achieve better tracking results for objects, especially with nonuniform energy distribution. Due to depth discontinuity and possible segmentation errors, matting [9], [20] is usually performed. By using the estimated alpha map and texture, it is also convenient to composite the IBR objects onto the background of the original or other plenoptic videos. The Bayesian approach in [20] is adopted in our system because of its good performance. After the objects in a plenoptic video have been extracted, the depth information for each IBR object can be further refined. An algorithm for rendering and postprocessing of plenoptic video with layered depth map is proposed, which has a low computational complexity. Another key problem associated with dynamic image-based representations is the compression of the tremendous amount of data. We have developed an object-based compression scheme for plenoptic videos to facilitate its rendering, transmission and storage. Due to page limitation, this object-based compression scheme is separately treated in a companion paper.

Since images and videos are special cases of the plenoptic function, it is envisioned that many conventional image processing algorithms [21] such as coding, segmentation, etc. have a similar analogy in IBR. We shall refer to these generalizations as plenoptic video processing or plenoptic processing in general. Moreover, as we will be focusing on the object-based approach, we will refer to the associated processing operations as object-based plenoptic processing. Another objective of this paper is devoted to the extension of some commonly used image processing algorithms to IBR and their possible applications. In particular, this will allow us to greatly increase the viewing freedom such as synthesizing views away from the camera line segments and performing zooming, panning, looking upward and downward, etc. To verify the proposed approach, a portable plenoptic video system, which consists of two linear arrays each carrying six video cameras, for large environment and dynamic scenes was constructed. The rendering and processing results are very satisfactory, which demonstrate the usefulness of the proposed object-based approach.

The rest of this paper is organized as follows. After a brief review of the concept of plenoptic function and the plenoptic videos, the construction of the proposed capturing system is described in Section II. The proposed object tracking algorithm and matting algorithm are presented in Section III. Section IV is devoted to the rendering of the IBR objects. The proposed object-based processing algorithms are described in Section V. Experimental results using the plenoptic videos as an example are also given to illustrate the concept. Finally, conclusions are summarized in Section VI.

II. The Plenoptic Video System

A. Plenoptic Function

The seven-dimensional plenoptic function, $P_7 = (V_x, V_y, V_z, \theta, \phi, \lambda, \tau)$, was first coined by Adelson and Bergen [1] to describe all the radiant energy that is perceived at any 3-D viewing point (V_x, V_y, V_z), from every possible angle (θ, ϕ) for every wavelength λ, and at any time τ. Based on this function, theoretically, novel views at different positions and time instants can be reconstructed from its samples, provided that the sample rate is sufficiently high. Due to its high-dimensional nature, data reduction/compression, transmission, and efficient rendering of the plenoptic function are essential to IBR systems. One approach is to restrict the viewing freedom of users so that the dimension of representations can be reduced. Light fields [3] or lumigraph [4] are two important types of four-dimensional image-based representations where images on a camera plane are taken to render novel views of the scene.

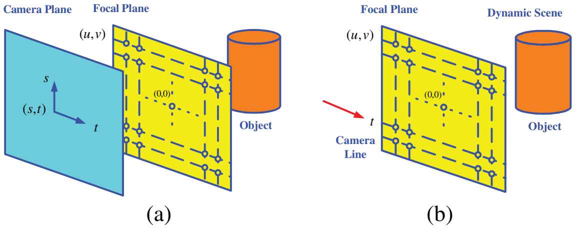

Fig. 1. (a) 4-D static light fields: viewpoints constrained on a 2-D plane. (b) 4-D SDLF viewpoints constrained along a line in a dynamic environment.

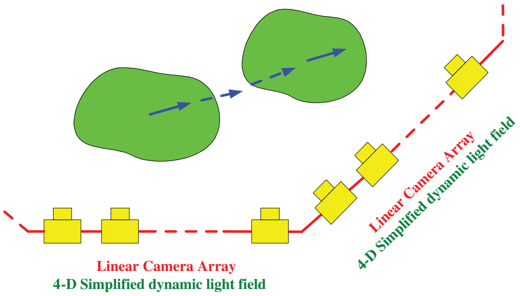
Linear Camera Array
4-D Simplified dynamic light field
Linear Camera Array
4-D Simplified dynamic light field

Fig. 2. Plenoptic videos: multiple linear camera array of 4-D SDLFs with viewpoints constrained along line segments.

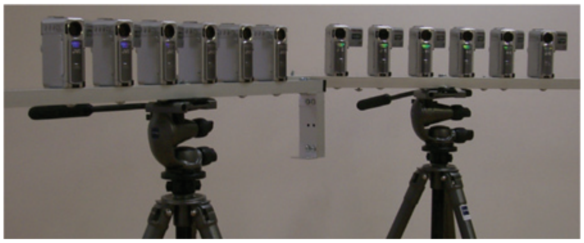

Fig. 3. Two linear camera arrays, each consists of six JVC video cameras.

Fig. 1(a) illustrates the principle of light field or lumigraph, where the (u, v) plane is the focal plane and (s, t) plane is the camera plane [4]. For dynamic light fields, the number of videos on a 2-D plane is usually very large. To avoid such a high dimensionality and the excessive hardware cost, a kind of simplified dynamic light field (SDLF) with viewpoints being constrained along line segments instead of a 2-D plane, as illustrated in Fig. 1(b), was proposed [11]–[14]. Because of the close relationship between the SDLF with traditional videos, we also referred it to as "plenoptic videos." Despite the simplicity of the overall system, significant parallax and lighting changes along the horizontal direction can also be observed.

B. The Plenoptic Video System

Previous attempts to generalize image-based representations to dynamic scenes are mostly based on 2-D panoramas. These include the QuickTime VR [2] and panoramic videos [10]. The panoramic video is a sequence of panoramas created at different locations along a path in space, which can be used to capture dynamic scenes at a stationary location or in general along a path with 360° of viewing freedom. The plenoptic video described in this paper is a simplified light field for dynamic environment as shown in Fig. 2, where viewpoints of the user are constrained along line segments to reduce the complexity of the dynamic IBR system. Unlike panoramic videos, users can still observe significant parallax and lighting changes. More recently, there have been attempts to construct light field video systems for different applications and characteristics. These include the Stanford multicamera array [22], the 3-D rendering system of Naemura *et al.* [23], and the (8 × 8) light field camera of Yang *et al.* [24]. The Stanford array consists of more than 100 cameras and is intended for large-environment applications. It uses a low-cost CMOS sensor and dedicated hardware for real-time compression. The systems in [24] and [25] consist of respectively 16 and 64 cameras and are intended for real-time rendering applications.

Fig. 3 shows the proposed plenoptic video system used to capture dynamic scenes in this paper. This system consists of two linear arrays of cameras, each hosting six JVC DR-DVP9ah video cameras. The spacing between successive cameras in the two linear arrays is 15 cm. The angle between the arrays is 165° connected together to form longer segments. Because the videos are recorded on tapes, the system is portable for capturing degree and can be flexibly adjusted. More arrays can be outdoor dynamic scenes. Along each linear camera array, a 4-D simplified dynamic light field is captured. The use of multiple linear arrays allows the user to have more viewing freedom in sport events and other live performance. The proposed system represents a design tradeoff between simplicity and viewing freedom. Other configurations can also be employed.

As those are off-the-shelf DV cameras, we do not have shutter synchronization. During capturing, an audio pulse is used to synchronize the DV cameras. The image frames that are closest to the time instance of the audio pulse in all the DV tapes data are measured. In our experiments, the rendering result is good and no temporal interpolation is needed. This is probably due to the stability of the clock generators in the DV cameras.

The cameras are calibrated using the method in [26]. In order to use this method to calibrate our camera array, a large reference grid is designed so that it can be seen simultaneously by all the cameras. By using the extracted intrinsic and extrinsic parameters of the cameras, the captured videos can be rectified for depth estimation and rendering. After capturing, the video data stored on the tapes can be transmitted to computers through FireWire interface. All these components are relatively inexpensive and they can readily be extended to include more cameras. Fig. 4 shows snapshots of plenoptic videos *Dance* and *Ping-Pong* captured by the proposed system. The resolution of these real-scene plenoptic videos is 720×576 pixels in 24-bit RGB format.

III. Object Tracking and Matting

A. Object Tracking Using Level-Set Method

In the proposed approach, objects at large depth differences are segmented into layers and are compressed and rendered separately. This helps to avoid the artifacts at object boundaries due to depth discontinuities. In the proposed method,

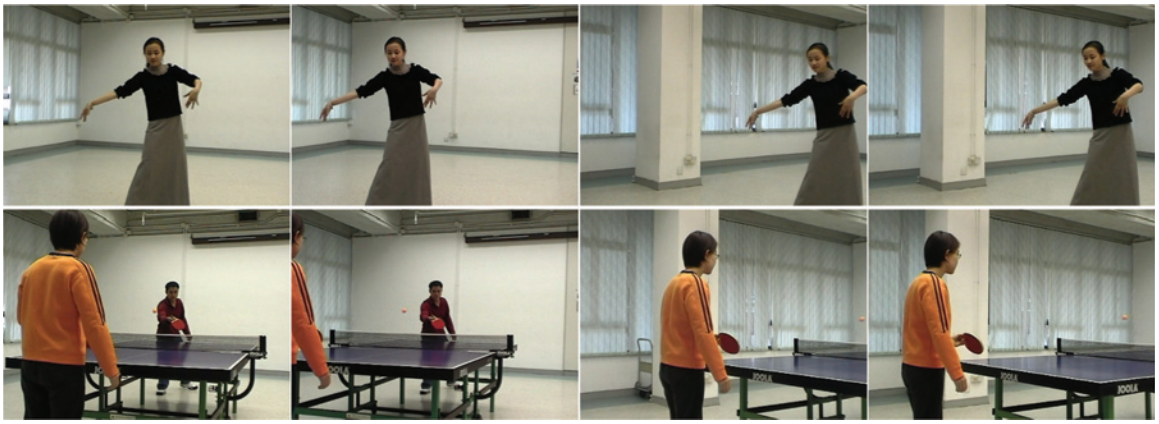

Fig. 4. Snapshots of the plenoptic videos: (upper) *Dance* and (lower) *Ping-Pong*.

an initial segmentation of the objects in, say, a key frame is first obtained using a semiautomatic approach [18]. Tracking techniques are then employed to segment the objects at other video streams and subsequent time instants. Our method is based on the level-set method or geometric partial differential equations (PDE). The initial segmentation provides prior information of the object to be segment and simplifies considerably the tracking of the objects at nearby camera views. The use of PDE and curvature-driven flows in tracking, segmentation, and image analysis has received great attention over the last few years [27]–[31]. The basic idea is to deform a given curve, surface, or image according to the PDE, and arrive at the desired result as the steady-state solution of this PDE. The problem can also be viewed as minimizing a certain energy function

$$U_I(C) = \int_I F(C, \boldsymbol{x}) dx \tag{1}$$

as a function of a curve or surface C. The subscript indicates that the energy is computed from the given images I. Usually, $F(C, \boldsymbol{x})$ is designed to measure the deviation of the desired curve from C at point $\boldsymbol{x}$. To minimize the functional in (1), the variational approach can be employed to convert it to a partial differential function. A necessary condition for C to be a local minimum of the functional is $U_I^{'}(C) = 0$. To solve it numerically, we usually start with an initial curve C_0 and let it evolve over a fictitious time variable t according to a PDE, which depends on the derivative $U_I^{'}(C)$ as follows:

$$\frac{\partial C(t)}{\partial t} = U_I^{'}(C(t)). \tag{2}$$

However, conventionally finite difference methods are unsuitable to solve (2), because the PDE might be singular at certain points. A major breakthrough in solving (2) is due to Sethian and Osher [32], and the method is commonly referred to as the *level-set method*. The basic idea behind the level-set method is to represent a curve or surface in an "implicit form" such as the zero level-sets or isophone of a higher dimensional function. More formally, the time evolution of curves $C(\boldsymbol{x}, t)$ is represented as the level-set of an embedding function $\phi(\boldsymbol{x}, t)$

$$L_c(\boldsymbol{x}, t) := \{(\boldsymbol{x}, t) \in R^3 \colon \phi(\boldsymbol{x}, t) = c\} \tag{3}$$

where c is a given real constant. Equation (2) can be rewritten as a PDE of $\phi(\boldsymbol{x}, t)$ as follows:

$$\frac{\partial \phi(t)}{\partial t} = \beta \|\nabla \phi\| \tag{4}$$

where β is the velocity of the flow in the normal direction and is derived from $U_I^{'}(C(t))$ above. The initial curve C_0 is associated with the level-set with $c = 0$, i.e. zero level-set, and its time evolution is computed numerically by solving the following equation for $\phi(t)$, after discretizing at a sufficiently small time interval or step Δt

$$\phi((n+1)\Delta t) = \phi(n\Delta t) + \Delta t \cdot G(\phi, \boldsymbol{x}) \tag{5}$$

where $G(\phi, \boldsymbol{x})$ is an appropriate approximation of the right-hand side of (4). The desired solution is obtained when the PDE converges at sufficiently large values of n.

For our object tracking and segmentation problem, we define the following energy function for curve C

$$U_I(C) = \gamma \int_I C_{\text{inside}} dxdy - \beta \int_I C_{\text{outside}} dxdy + \lambda Length(C) \tag{6}$$

where γ, β, and λ are positive parameters, $C_{\text{inside}}(x, y)$ and $C_{\text{outside}}(x, y)$ are two functions designed respectively, to control the expansion and contraction of the curve C at the location (x, y), and $Length(C)$ measures the length of the curve. In conventional level-set methods [19], the pixel values inside and outside the curve C are assumed to be independent and Gaussian-distributed with means c_{in} and c_{out}, respectively, inside and outside the curve, and then it can be shown that the PDE so obtained can be written as (details omitted due to page limitation)

$$\left.\frac{\partial \phi}{\partial t}\right|_{(x,y)} = \gamma \left(u_{(x,y)} - c_{\text{in}}\right)^2 - \beta (u_{(x,y)} - c_{out})^2 + \lambda \cdot \operatorname{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right) \tag{7}$$

where $u_{(x,y)}$ is the value of the pixel at location (x, y), and c_{in} and c_{out} denote the driving force inside and outside the curve C, respectively. The third term, which is derived from $Length(C)$, makes the curve smooth and continuous.

There are two different methods for determining c_{in} and c_{out}: global-based; and local-based. The global-based method

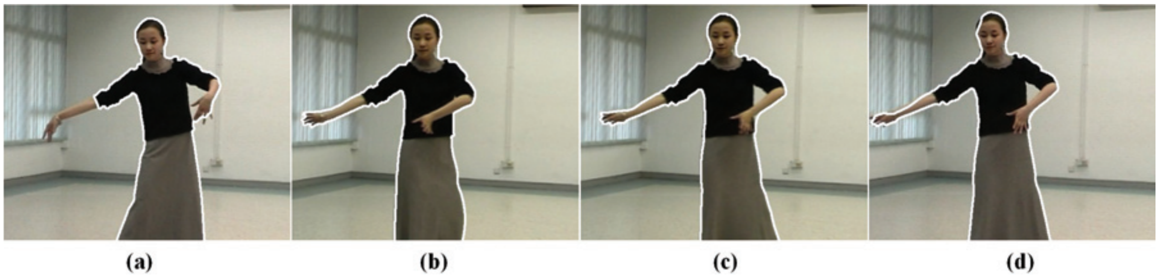

Fig. 5. (a) Tracking result of the global-based method. (b)-(d) Tracking results of the proposed method.

which is adopted in [19] utilizes all the pixels to drive curve C. The global-based method has the advantage of fast evolution speed and is less sensitive to noise. However, some fine features along the object's boundary to be tracked may be lost. Fig. 5(a) shows an example tracking result using the global-based method. It can be seen that the girl's right hand is outside the curve, because its mean is more similar to the background than to its body. On the contrary, the local-based method uses the local mean value inside a window instead of all the image pixels. In [27] and [33], a local-based method is exploited, where c_{in} and c_{out} are set as follows: $c_{\text{in}} = u_{(x+i,y+j)}$, where $(u_{(x,y)} - u_{(x+i,y+j)})^2$ is the minimum value over all integer pairs (i, j) such that $|i| \leq m$ and $|j| \leq m$ and pixel $(x+i, y+j)$ is inside the curve C; $c_{\text{out}} = u_{(x+i,y+j)}$, where $(u_{(x,y)} - u_{(x+i,y+j)})^2$ is the minimum value over all integer pairs (i, j) such that $|i| \leq m$ and $|j| \leq m$ and pixel $(x + i, y + j)$ is outside the curve C. Obviously, this method utilizes local features of the image to cope with objects having a nonuniform energy distribution. Unfortunately, this method is rather sensitive to image noise, because only one pixel is chosen for determining both c_{in} and c_{out}. Here we propose to combine the advantages of both the global-based and local-based methods by employing the following c_{in} and c_{out}

$$\begin{cases} c_{\text{in}} = \text{average } (u_{(x+i,y+j)}), \text{ where } |i| \leq m, |j| \leq m \text{ and} \\ \quad \text{pixel } (x+i, y+j) \text{ is inside the curve } C \\ c_{\text{out}} = \text{average } (u_{(x+i,y+j)}), \text{ where } |i| \leq m, |j| \leq m \text{ and} \\ \quad \text{pixel } (x+i, y+j) \text{ is outside the curve } C. \end{cases} \tag{8}$$

Although the initial contour obtained from other views help in providing prior information of the object shape, objects may overlap each other and may affect the performance of the level-set method based on intensity alone. Here we propose to segment the intensity information with disparity information computed from adjacent images. In computing the depth map, we use squared intensity differences as cost function, and aggregate the cost in a square window weighted by color similarity and geometric proximity as in Yoon's method [34]. The disparity map is first estimated by the pyramid Lucas–Kanade (LK) feature-tracking algorithm, which minimizes the cost/energy by the least-squares method. Instead of defining smoothness term in the energy function, the disparity map is anisotropic diffused after LK method. Finally, a symmetric stereo model [35] is introduced for occlusion detection and optimized with Belief-propagation. By adding an additional depth term to the level-set speed function (7), we get the new speed function as follows:

$$\begin{aligned} \left.\frac{\partial \phi}{\partial t}\right|_{(x,y)} &= \gamma_1(u_{(x,y)} - c_{\text{in}})^2 - \beta_1(u_{(x,y)} - c_{\text{out}})^2 \\ &\quad + \gamma_2(d_{(x,y)} - d_{\text{in}})^2 - \beta_2(d_{(x,y)} - d_{\text{out}})^2 \\ &\quad + \lambda \cdot div\left(\frac{\nabla \phi}{|\nabla \phi|}\right) \end{aligned} \tag{9}$$

where the $d_{(x,y)}$ is the depth value of pixel (x, y), and d_{in} and d_{out} denote the local mean depth inside and outside the curve C. Using this speed function, more satisfactory result can be obtained. To improve tracking results, a key frame is usually chosen for semiautomatic segmentation. Tracking is then performed in forward and backward time until the objects disappear.

B. Object Matting with Bayesian Approach

Due to possible segmentation errors around boundaries and finite sampling at depth discontinuities, it is preferable to calculate a soft, instead of a hard, membership function between the IBR objects and the background. In other words, the boundary pixels are assumed to be a linear combination of the corresponding pixels from the foreground and background

$$I = \alpha F + (1 - \alpha)B \tag{10}$$

where I, F, and B are the pixel's composite, foreground, and background colors, and $0 \leq \alpha \leq 1$ is the pixel's opacity component or the alpha map. Using this model, it is possible to matte a given object with the original at different views and other background. The digital analog of the matte (the α-map) was introduced by Porter and Duff [36] in 1984 to facilitate matting of objects. In natural matting, all variables α, F, and B need to be estimated and the problem is to find the most likely estimates for α, given the observation I. This can be formulated as the maximization of the posteriori probability $P(F, B, \alpha|I)$. Using the Bayesian rule, we have

$$\max_{F,B,\alpha} P(F, B, \alpha|I) = \max_{F,B,\alpha} P(I|F, B, \alpha)P(F, B, \alpha)/P(I). \tag{11}$$

Since the optimization parameters are independent of $P(I)$, the latter can be dropped. Further, if F, B, and α are assumed

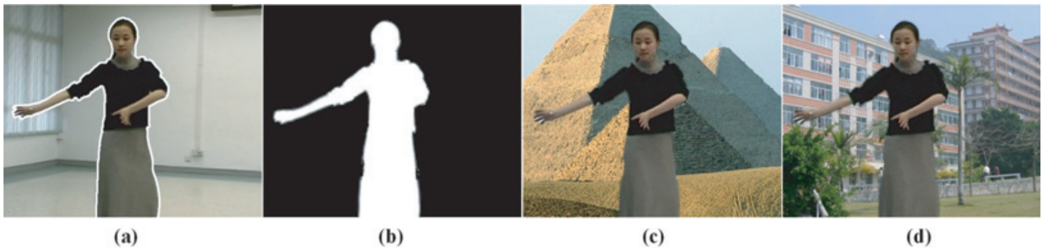

Fig. 6. (a) Input image. (b) Alpha map. (c) - (d) New images of compositing extracted foreground over other background scenes.

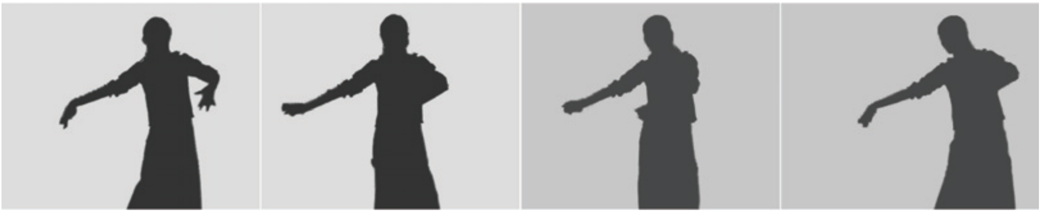

Fig. 7. Examples of layered depth maps in the *Dance* sequence.

to be independent, then (11) can be written as

$$\begin{aligned}\arg\max_{F,B,\alpha} & P(F,B,\alpha|I)\\ &= \arg\max_{F,B,\alpha} P(I|F,B,\alpha)P(F)P(B)P(\alpha)\\ &= \arg\max_{F,B,\alpha} \{\ln P(I|F,B,\alpha) + \ln P(F)\\ &\qquad + \ln P(B) + \ln P(\alpha)\}. \qquad (12)\end{aligned}$$

Taking the derivatives of (12), one gets a set of equations in the estimates of α, F, and B. Interested readers are referred to [20] for more information. Once the matting information is obtained, conventional stereo-matching algorithm can be employed to refine the depth values within each segment.

C. Experimental Results

The performance of the proposed tracking method is evaluated using the *Dance* sequence, which is a real-scene plenoptic video captured by our IBR capturing system. For each frame, the initial curve C_0 is the tracking result of the previous frame, and the object curve of the key frames is obtained manually using Lazy snapping [18]. The level-set contour evolution is implemented using the narrow band method, where (9) is used as the speed function. The window size m for the local energy calculation is fixed to 6. Fig. 5(a) shows the tracking result of the global-based method. The results of our method are shown in Fig. 5(b) and (c), where the boundary is well delineated. It can be seen from the results that the proposed method gives more reasonable result for objects with nonuniform energy distribution. Although the proposed intensity-based method is capable of tracking the objects satisfactorily for a number of frames, the performance will start to deteriorate due to accumulation of tracking errors. This is illustrated in Fig. 5(d), where parts of the girl's head and right hand are not well delineated. By using the proposed intensity and depth-based level-set method, considerably more satisfactory tracking results are obtained as shown later in Fig. 8.

The results of natural matting the IBR object are illustrated in Fig. 6. Fig. 6(a) and (b) shows an example snapshot of a segmented IBR object called "dancer" and its computed associated alpha map. Fig. 6(c) and (d) shows example renderings of the IBR object, after matting with two different backgrounds or scenes. Fig. 7 shows some of the layered depth maps in the *Dance* sequence. The tracking and matting results of the *Ping-Pong* sequence are shown in Figs. 8 and 9. It can be seen that the proposed intensity and depth information-based tracking method performs very well even in the *Ping-Pong* sequence, which consists of more than one overlapping objects.

IV. Rendering of IBR Objects

A difficult problem of rendering light fields and plenoptic videos is the excessive artifacts due to depth variations. Using the layered depth map it is possible to detect occlusion and interpolate the image pixels during rendering. In a previous paper [11], a depth-matching algorithm for rendering and postprocessing of plenoptic video with depth information was proposed. This algorithm brings satisfactory rendering results, but its arithmetic complexity is very high. Here, an improved rendering algorithm with a much lower computational complexity is proposed in this paper.

More precisely, instead of finding the depth value of the image pixel to be rendered from adjacent light field images, the two images are projected using the depth values of each pixel to the current viewing position. During the reconstruction of a pixel V in the viewing grid, two pixels are obtained from projecting the left and right images to the position of pixel V. If both of them have the same depth values, then there is no occlusion and the value of V can be interpolated from these pixels according to bilinear interpolation. On the other hand, if their depth values differ considerably (say larger than a threshold), then occlusion is detected. The projected pixel with a small depth value will then occlude the other. Therefore, the value of pixel V should be equal to the one with smaller depth value. Furthermore, if multiple pixels are projected to the location of pixel V, the intensity of pixel V is assigned to the one with the smallest depth value. If only one pixel from the left or right image is projected to the position of pixel V, the intensity of pixel V is set to the intensity of this

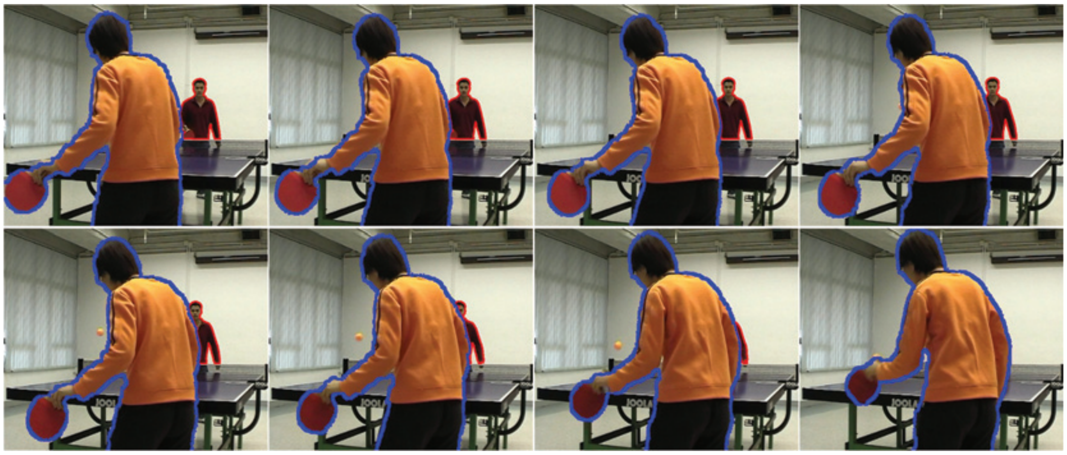

Fig. 8. Tracking results of the *Ping-Pong* sequence.

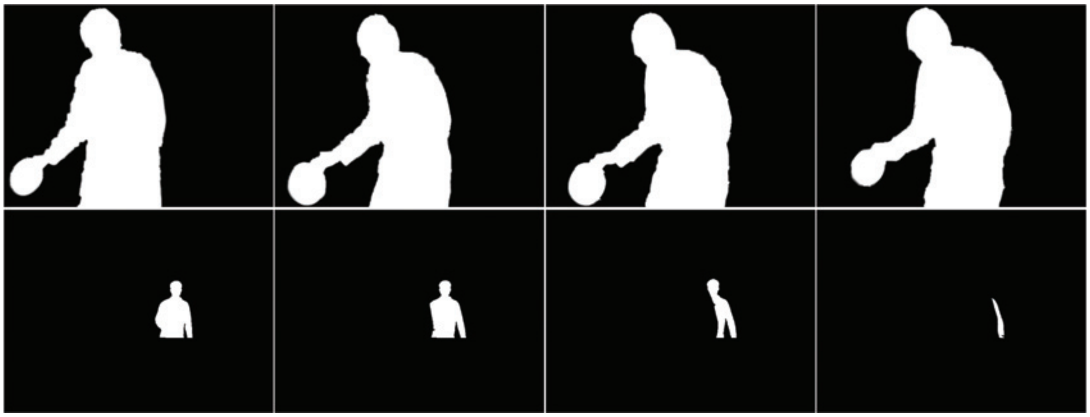

Fig. 9. Example alpha maps of the IBR objects "Player 1" and "Player 2" in the *Ping-Pong* sequence.

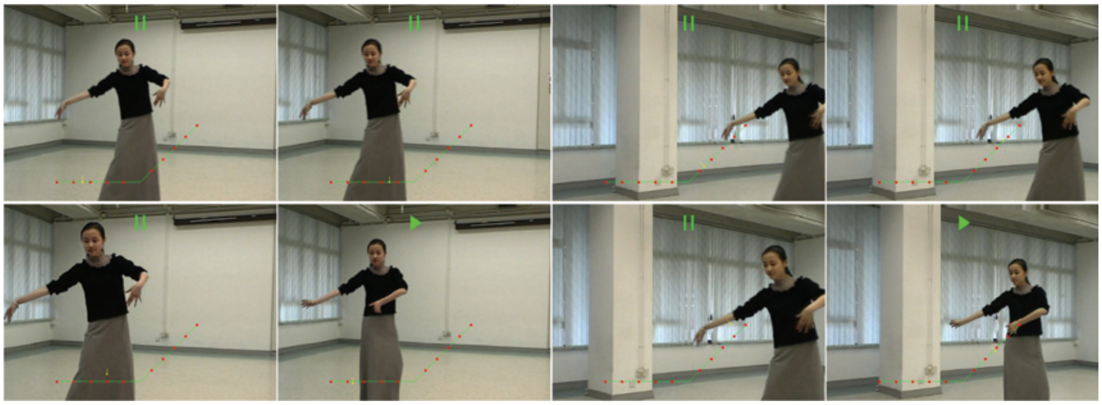

Fig. 10. Rendering results of the *Dance* sequence.

pixel. Finally, due to depth discontinuity, pixel V might not have any projected pixels from adjacent light field images. In this case, we employ the image consistency concept to "guess" the intensity of these pixels [11] from neighboring rendered pixels using interpolation or other inpainting techniques.

A. Experimental Results

The rendering results at different viewpoints of the *Dance* and *Ping-Pong* sequences are shown in Figs. 10 and 11. It can be seen that the object-based approach yields high-quality renderings and it is effective in suppressing the ghosting and blurring artifacts in the conventional approach with a single mean depth. Our rendering algorithm avoids a complicated full search of depth value used in [11], so the computational complexity of the proposed algorithm is reduced significantly. It takes about 1.3 s for the algorithm in [11] to render one frame (720 × 576), while that of the proposed algorithm is about 120 ms. Currently, all algorithms are run on the CPU. To speedup the overall performance, time consuming algorithms, such as pixel projecting (33.6%) and interpolation (51.1%), can be transferred to the sophisticated GPU.

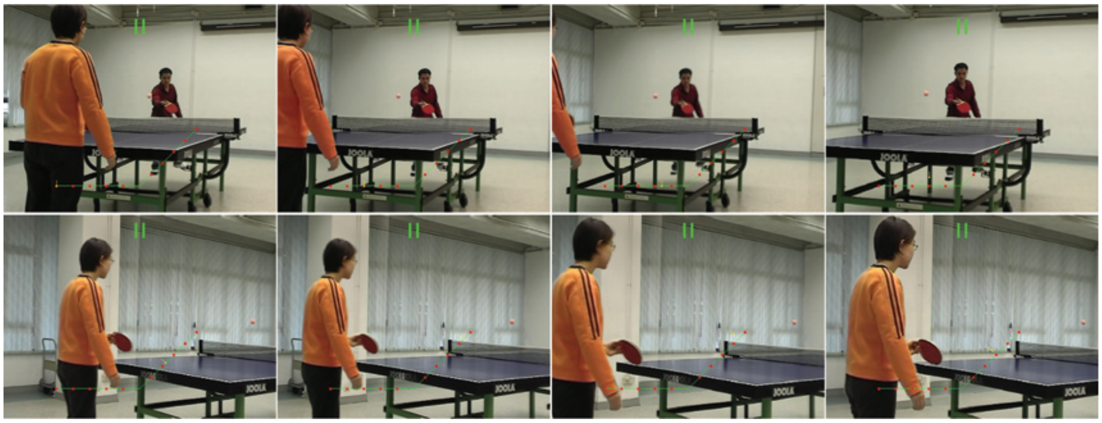

Fig. 11. Rendering results of the *Ping-Pong* sequence.

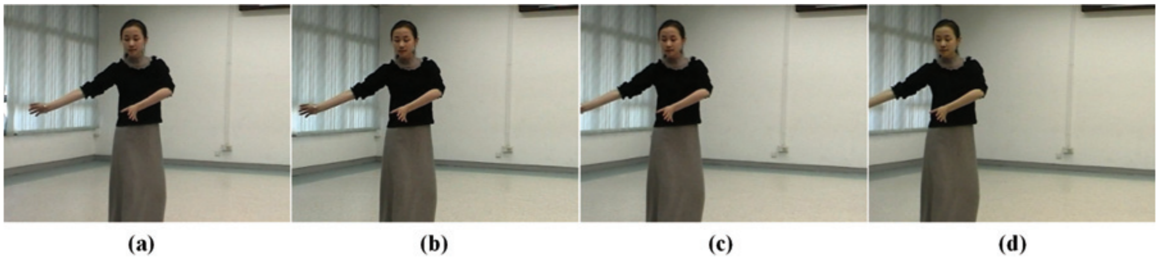

Fig. 12. (a)–(d) Snapshots of the plenoptic video *Dance* from camera one to camera four.

V. Object-Based Plenoptic Video Processing

The main difference between processing a single-image and plenoptic videos is in ensuring the image consistency constraints in multiple views of the same object. Ideally, when a group of pixels of an object in a given image is modified, then the "corresponding" pixels in other images should also be modified consistently. If scene geometry is also available, then the lighting and other physical constraints should also be observed. Due to the difficulty in acquiring accurately scene geometry and other physical parameters, it is unavoidable that the capability of automatic IBR processing is limited. Or, in other words, additional prior information must be provided by the users through appropriately designed user interface and other tools.

In what follows, we are interested in generalizing some commonly used image-processing algorithms to the IBR case under the object-based framework. In principle, one should determine the correspondence between image pixels and process them as a whole. Under the object-based framework, similar objects are segmented and grouped together. Therefore, it is easier to approximately satisfy the image consistency by processing the image pixels from the IBR object as a whole instead of from the entire images. We first start with image completion and then object enhancement and filtering.

A. Object Completion

When capturing the plenoptic videos, it may happen that some cameras cannot capture the whole object. Fig. 12 shows snapshots of the plenoptic video *Dance*. In the figure we can see that cameras 1 and 2 captured the whole object "girl," but part of girl's hand was not captured by cameras three and four. During rendering, when the viewpoint is moved to the region between cameras three and four, the reconstructed object is incomplete, as shown in Fig. 13(a). The goal of object completion is to inpaint those possible incomplete objects from the IBR object at other camera views.

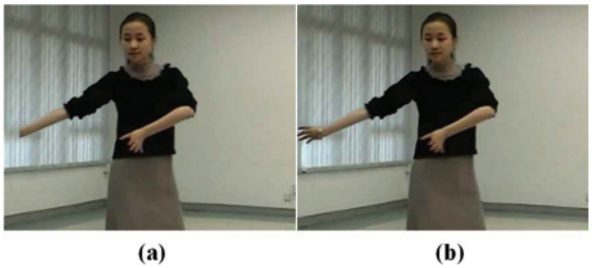

Fig. 13. (a) Rendered image at one viewpoint between camera three and four. (b) Rendering result after object completion processing.

Since the missing pixels are captured by camera one, it can be used to complete the IBR object. To this ends, we employ model-based motion estimation [37] to estimate the motion vectors of each pixel from the object in camera one to that in camera two. We then wrap the former to the position of the latter according to the motion vectors. After wrapping, the two objects would almost overlap each other except at image boundaries. Linear interpolation is then applied to reconstruct the missing portion. This process can also be viewed as a kind of inpainting process.

Because the distance between the object and cameras is large and all cameras are placed on a horizontal line, we approximate the motion or disparity of the object by an affine transformation

$$\mathbf{u}(\mathbf{x}) = \mathbf{X}(\mathbf{x})\mathbf{a} \tag{13}$$

where $\mathbf{u(x)}$ is the motion vector, $\mathbf{a}$ denotes the affine transformation vector $(a_1, a_2, a_3, a_4, a_5, a_6)^T$, and

$$\mathbf{X}(\mathbf{x}) = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix}. \tag{14}$$

The affine transformation vector $\mathbf{a}$ can be estimated by the least-squares method. Thus, the motion of each pixel in an object is completely specified by vector $\mathbf{a}$. After wrapping and linear interpolation, the completed object is shown in Fig. 13(b). By using object completion, we can also render views outside the line segments defined by the camera arrays since the missing pixels can be effectively inpainted. By synthesizing different views appropriately, we can also generate views of stereo and multiview display for 3-D and multiview TVs.

B. Object Enhancement

In many applications, an object captured in an image sequence may need to be pasted on a different background. During video editing, image enhancement algorithms such as readjustment of intensity, sharpening, or blurring may be needed to improve the quality of plenoptic videos. Since the objects are already segmented from the plenoptic video, we can therefore enhance selected objects individually instead of the whole image. This provides more flexibility in data processing. We now briefly outline a few different enhancement algorithms for the object-based approach.

1) Histogram Equalization for Object: Histogram equalization [21] can enhance the contrast of objects. A normalized histogram of object pixels is given by

$$p_r(r_k) = \frac{n_k}{n}, \quad k = 0, 1, 2, \ldots, L-1 \tag{15}$$

where n is the total number of pixels in the object, n_k is the number of pixels that have intensity level r_k, and L is the total number of possible intensity levels. The transformation function of histogram equalization has the form

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}, \quad k = 0, 1, 2, \ldots, L-1. \tag{16}$$

Thus, a processed object is obtained by mapping each pixel with intensity r_k in the input object into a corresponding pixel with level s_k via (16).

If histogram equalization is applied independently to the original color components (R, G, B), it will usually result in erroneous color. Therefore, image pixels are transformed from the RGB color space to the hue, intensity, saturation (HIS) color space. Histogram equalization is then applied only to the color intensity component, leaving the other color components (e.g., hue, saturation) unchanged. The rendering image before processing is depicted in Fig. 14(a). Fig. 14(b) and (c) shows the same image after object enhancement. Only the brightness of the object "girl" is modified and the background remains the same. For comparison, Fig. 14(d) shows the rendering result after histogram equalization for the whole image (not for object). We can see that the enhancement mainly happens at the background, and the foreground object, which is usually considered as more important part in an image, is enhanced slightly. This shows the flexibility of the object-based approach.

2) Object Deblurring: Deblurring [21] can effectively improve the quality of image. Like histogram equalization, deblurring can be applied to selected IBR objects. In general, deblurring approaches fall into two broad categories: spatial-domain methods; and frequency-domain methods. Here, a spatial domain filter using second-order Laplacian derivative is employed. Other frequency domain filters can also be used. The second-order Laplacian derivative is defined as

$$\begin{aligned} \nabla^2 f &= \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \\ &= f(x+1, y) + f(x-1, y) \\ &\quad + f(x, y+1) + f(x, y-1) - 4f(x, y). \end{aligned} \tag{17}$$

For RGB color space, the Laplacian of a pixel $c(x, y)$ is

$$\nabla^2 c(x, y) = \begin{bmatrix} \nabla^2 \mathrm{R}(x, y) \\ \nabla^2 \mathrm{G}(x, y) \\ \nabla^2 \mathrm{B}(x, y) \end{bmatrix}. \tag{18}$$

The modified pixel is given by

$$\hat{c}(x, y) = c(x, y) - \nabla^2 c(x, y). \tag{19}$$

By applying this derivative to each pixel of the IBR object, object deblurring can be achieved, as shown in Fig. 14(e).

3) Background Defocusing: The basic idea of background defocusing is to blur the background, so that objects in the foreground will be popped out. This object enhancement can also be viewed as a kind of special effect. For simplicity, the defocusing is approximated by an averaging or mean filter as follows:

$$\hat{c}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} c(s, t) \tag{20}$$

where S_{xy} is a rectangular window of size $m \times n$ centered at location (x, y). The window size adopted here is 5×5, and the result of background defocusing is shown in Fig. 14(f).

C. Background Transformation

With the help of the alpha map of an object, it is possible to matte a given object to a transformed background, such as a rotated background, in the plenoptic videos. This special effect is shown in Fig. 15. Fig. 15(a) is the original rendered image while Fig. 15(b) and (c) shows the result after the object "girl" is pasted to a new background. Further postprocessing is needed if shadow and other lighting effects are required. This is difficult to be carried out without the knowledge of the geometry of the scene. Therefore, for interactive rendering and relighting, capturing a rough geometry of the scene is of great importance and hence a fruitful area of research.

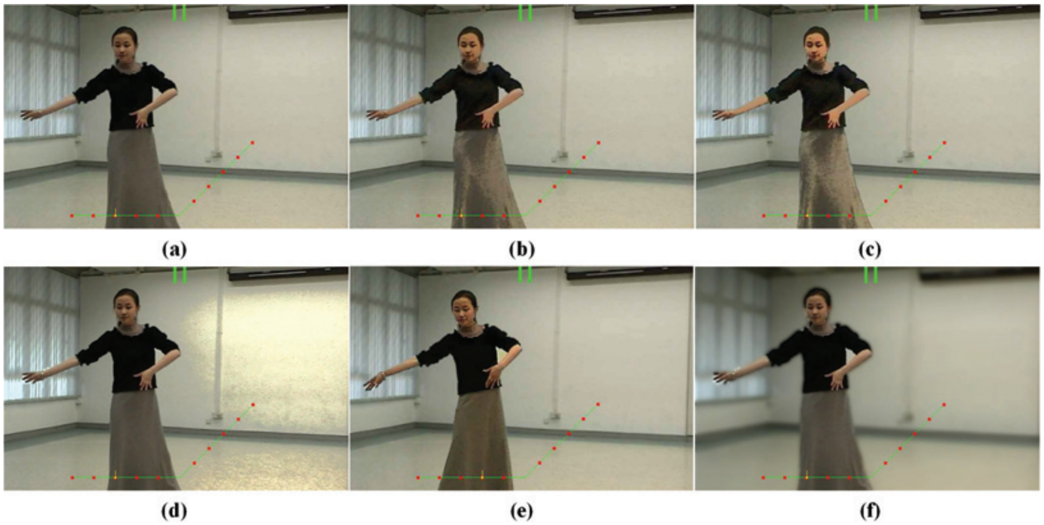

Fig. 14. Rendered images (a) before processing; (b)-(c) object enhancement processing; (d) after histogram equalization for the whole image; (e) after object deblurring processing; and (f) after background defocusing processing.

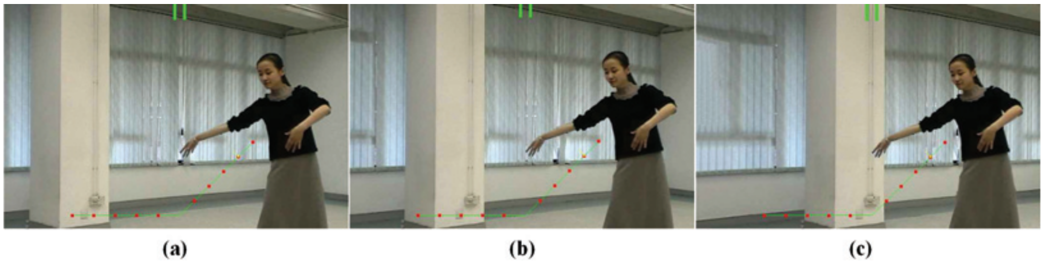

Fig. 15. (a) Original rendered image. (b)-(c) Rendered images after background transformation processing.

VI. Conclusion

We have presented an object-based approach to image-based synthesis and processing for 3-D and multiview TVs using the plenoptic videos as an example. The plenoptic video sequences are segmented into IBR objects each with its image sequence, depth map, and other relevant information such as shape information. A portable capturing system consisting of two linear camera arrays, each hosting six JVC video cameras, was developed to verify the proposed approach. We also proposed an object tracking approach based on the level-set method. The depth information for each IBR objects is estimated separately, and a rendering algorithm using layered depth map is also proposed. Natural matting with Bayesian approach is employed to improve the rendering quality under depth discontinuity and possible segmentation errors, and it allows us to composite the IBR objects onto different plenoptic videos. Furthermore, the concept of plenoptic processing is introduced and illustrated with several typical video processing operations and applications. Experimental results for tracking, matting, rendering, and processing using both the synthetic and real-world sequences demonstrate the usefulness, good quality, and flexibility of the proposed approaches.

References

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

[2] S. E. Chen, “QuickTime VR-An image-based approach to virtual environment navigation,” in *Proc. Comput. Graph. (SIGGRAPH’95)*, Aug. 1995, pp. 29–38.

[3] M. Levoy and P. Hanrahan, “Light field rendering,” in *Proc. Comput. Graph. (SIGGRAPH’96)*, Aug. 1996, pp. 31–42.

[4] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, “The lumigraph,” in *Proc. Comput. Graph. (SIGGRAPH’96)*, Aug. 1996, pp. 43–54.

[5] J. Shade, S. Gortler, L. W. He, and R. Szeliski, “Layered depth images,” in *Proc. Comput. Graph. SIGGRAPH’98*, pp. 231–242.

[6] 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.

[7] H. Y. Shum, S. C. Chan, and S. B. Kang, *Image-Based Rendering*, New York: Springer-Verlag, 2006.

[8] J. X. Chai, X. Tong, S. C. Chan, and H. Y. Shum, “Plenoptic sampling,” in *Proc. Comput. Graph. (SIGGRAPH’00)*, Jul. 2000, pp. 307–318.

[9] 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.

[10] K. T. Ng, S. C. Chan, and H. Y. Shum, “The data compression and transmission aspects of panoramic videos,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 82–95, Jan. 2005.

[11] Z. F. Gan, S. C. Chan, K. T. Ng, K. L. Chan, and H. Y. Shum, “On the rendering and post-processing of simplified dynamic light fields with depth information,” in *Proc. IEEE ICASSP*, vol. 3, May 2004, pp. 321–324.

[12] 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.

[13] S. C. Chan, K. T. Ng, Z. F. Gan, K. L. Chan, and H. Y. Shum, “The plenoptic videos: Capturing, rendering and compression,” in *Proc. IEEE ISCAS*, vol. 3, May 2004, pp. 905–908.

[14] S. C. Chan, K. T. Ng, Z. F. Gan, K. L. Chan, and H. Y. Shum, “The plenoptic videos,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 12, pp. 1650–1659, Dec. 2005.

[15] 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 Syst.*, vol. 4, May 2005, pp. 3435–3438.

[16] Z. F. Gan, S. C. Chan, K. T. Ng, and H. Y. Shum, "Object tracking for a class of dynamic image-based representations," in *Proc. SPIE Visual Commun. Image Process.*, Jul. 2005, pp. 1267–1274.

[17] Z. F. Gan, S. C. Chan, and H. Y. Shum, "Object tracking and matting for a class of dynamic image-based representations," in *Proc. IEEE Advanced Video Signal-Based Surveillance*, Sep. 2005, pp. 81–86.

[18] Y. Li, J. Sun, C. K. Tang, and H. Y. Shum, "Lazy snapping," in *Proc. Comput. Graph. (SIGGRAPH'04)*, pp. 303–308.

[19] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.

[20] Y. Y. Chuang, B. Curless, D. Salesin, and R. Szeliski, "A bayesian approach to digital matting," in *Proc. IEEE Conf. CVPR*, vol. 2, Dec. 2001, pp. 264–271.

[21] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Englewood Cliffs, NJ: Prentice Hall, 2002.

[22] B. Wilburn, M. Smulski, H. H. Lee, and M. Horowitz, "The light field video camera," in *Proc. SPIE Electron. Imaging: Media Process. '2002*, vol. 4674, Jan. 2002, pp. 29–36.

[23] T. Naemura, J. Tago, and H. Harashima, "Real-time video-based modeling and rendering of 3-D scenes," *IEEE Trans. Comput. Graph. Applicat.*, vol. 22, no. 2, pp. 66–73, Mar.–Apr. 2002.

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

[25] B. Goldlücke, M. Magnor, and B. Wilburn, "Hardware-accelerated dynamic light field rendering," in *Proc. VMV'2002*, pp. 455–462.

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

[27] A. R. Mansouri, "Region tracking via level set PDEs without motion computation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 947–961, Jul. 2002.

[28] N. Paragios and R. Deriche, "Geodesic active contours and levels sets for the detection and tracking of moving objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 3, pp. 266–280, Mar. 2000.

[29] G. Sapiro, *Geometric Partial Differential Equations and Image Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2001.

[30] J. A. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*. Cambridge, U.K.: Cambridge Univ. Press, 1996.

[31] S. J. Osher and R. P. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*. New York: Springer-Verlag, 2002.

[32] S. J. Osher and J. A. Sethian, "Fronts propagation with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.

[33] A. Yilmaz, X. Li, and M. Shah, "Object contour tracking using level sets," in *Proc. ACCV 2004*, Korea.

[34] K. J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 650–656, Apr. 2006.

[35] S. S. Intille and A. F. Bobick, "Disparity-space images and large occlusion stereo," in *Proc. ECCV*, 1994, pp. 179–186.

[36] T. Porter and T. Duff, "Compositing digital image," in *Proc. Comput. Graph. (SIGGRAPH'84)*, Jul. 1984, pp. 253–259.

[37] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, "Hierarchical model-based motion estimation," in *Proc. Eur. Conf. Comput. Vision*, Santa Margharita Ligure, Italy, 1992, pp. 237–252.

Shing-Chow Chan (S'87–M'92) received the B.Sc. (Eng) and Ph.D. degrees from The University of Hong Kong, Hong Kong, China, in 1986 and 1992, respectively.

He joined The University of Hong Kong in 1994 and is now an Associate Professor in the Department of Electrical and Electronic Engineering. He was a Visiting Researcher with the Microsoft Corporation, Redmond, WA, and Microsoft, Beijing, China, in 1998 and 1999, respectively. His research interests include fast transform algorithms, filter design and realization, multirate signal processing, and image-based rendering.

Dr. Chen is member of the Digital Signal Processing Technical Committee of the IEEE Circuits and Systems Society. He was Chairman of the IEEE Hong Kong Chapter of Signal Processing from 2000 to 2002. He is currently an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I—REGULAR PAPERS and the *Journal of VLSI Signal Processing and Video Technology*.

Zhi-Feng Gan (S') received the B.E. and M.E. degrees in electrical and information engineering from Huazhong University of Science and Technology, Wuhan, China, in 1999 and 2001, respectively. He received the Ph.D. degree in electrical and electronic engineering from The University of Hong Kong, Hong Kong, China in 2006.

He is currently a Research Associate in the Department of Electrical and Electronic Engineering, The University of Hong Kong. His research interests include image-based rendering, computer vision, and image/video processing.

King-To Ng (S'96–M'03) received the B.Eng. degree in computer engineering from the City University of Hong Kong, Hong Kong, China, in 1994, and the M.Phil. and Ph.D. degrees in electrical and electronic engineering from The University of Hong Kong, Hong Kong, China, in 1998 and 2003, respectively.

In 2004, he worked as a Visiting Associate Researcher at Microsoft Research Asia, Beijing, China. Currently, he is a Postdoctoral Fellow in the Department of Electrical and Electronic Engineering, The University of Hong Kong. His research interests include visual communication, image-based rendering, and video broadcast and transmission.

Ka-Leung Ho (M'80) received the B.Sc (Eng.) and the M.Phil. degrees in electrical engineering from The University of Hong Kong, Hong Kong, China, in 1971 and 1973, respectively, and the Ph.D. degree from the University of London, London, in 1977.

In 1984, he joined the Department of Electrical and Electronic Engineering, The University of Hong Kong. His current research interests include signal processing and communications systems.

Dr. Ho is a Chartered Engineer of the Engineering Council, U.K., a Fellow of the IEE, and a Member of the HKIE.

Heung-Yeung Shum (SM'01–F'06) received the Ph.D. degree in robotics from the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, in 1996.

He joined Microsoft Research, Redmond, WA, in 1996, as a Researcher. He moved to Beijing as one of the founding members of Microsoft Research China, Beijing, China, which was later renamed Microsoft Research Asia. There, he began a nine-year tenure as a Research Manager, subsequently moving on to become Assistant Managing Director and then Managing Director of Microsoft Research Asia, Distinguished Engineer, and Corporate Vice President. He is currently the Corporate Vice President responsible for search product development at the Microsoft Corporation, Redmond, WA. Previously, he oversaw the research activities at Microsoft Research Asia and the lab's collaborations with universities in the Asia Pacific region, and was responsible for the Internet Services Research Center, an applied research organization dedicated to long-term and short-term technology investments in search and advertising at Microsoft. He has published more than 100 papers on computer vision, computer graphics, pattern recognition, statistical learning, and robotics. He holds more than 50 U.S. patents.

Dr. Shum is a Fellow of the ACM.