3D LAMP: a New Layered Panoramic Representation

Zhigang Zhu[+*] , Allen R. Hanson[+]
[+]Department of Computer Science , University of Massachusetts at Amherst, MA 01003
* Department of Computer Science and Technology, Tsinghua University, Beijing 100084
[+]Email: {zhu, hanson}@cs.umass.edu

Abstract

In this paper a compact representation, 3D Layered, Adaptive-resolution and Multi-perspective Panorama (LAMP), is proposed for representing large scale and 3D scenes with occlusion. Two kinds of 3D LAMP representations are constructed, i.e. the relief-like LAMP and the image-based LAMP, both of which concisely represent almost all the information from a long image sequence. The relief-like LAMP is basically a single extended multi-perspective panoramic view image with both texture and depth values, but each pixel has multiple values to represent results of occlusion recovery and resolution enhancement. The image-based LAMP, on the other hand, consists of a set of multi-perspective layers, each of which has both texture and depth maps, with adaptive time-sampling scales depending on depths of scene points. Several examples of 3D LAMP construction for real image sequences are given. The 3D LAMP is a concise and powerful representation for image-based rendering.

1. Introduction

The problem of modeling and rendering real scenes has received increasing attention in recent years in both the computer vision and computer graphics communities. Two effective ways have been proposed to represent video sequences of large-scale 3D scenes: panoramic (mosaic) representations and layered representations. A video mosaic is a composite of images created by registering overlapping frames. Many of the current successful image mosaic algorithms, however, only generate 2D mosaics from a camera rotating around its nodal point [1,2,3]. Creating multi-perspective stereo panoramas from one rotating camera off its nodal point was proposed by Ishiguro, et al [4], Peleg & Ben-Ezra [5], and Shum & Szeliski [6]. A system for creating a global view for visual navigation by pasting together columns from images taken by a smoothly translating camera (comprising only a vertical slit) was proposed by Zheng & Tsuji [7]. The moving slit paradigm was used as the basis of the 2D manifold projection approach for image mosaicing [8], the multiple-center-of-projection approach for image-based rendering [9], the epipolar plane analysis techniques for 3D reconstruction [15], and the parallel-perspective view interpolation methods for stereo mosaicing [16]. Multi-perspective panoramas (or mosaics) exhibit very attractive properties for visual representation and epipolar geometry; however, 3D recovery of stereo mosaics faces the same problems as traditional stereo methods - the correspondence problem and occlusion handling.

In a layered representation, a set of depth surfaces is first estimated from an image sequence from a single camera, and then combined to generate a new view. Wang and Adelson [10] addressed the problem as the computation of 2D affine motion models and a set of support layers from an image sequence. The *layered representation* that they proposed consists of three maps in each layer: a mosaicing intensity map, an alpha map, and a velocity map. Occlusion boundaries are represented as discontinuities in a layer's alpha map (opacity). This representation is a good choice for image compression of a video sequence, and for limited image synthesis of selected layers. Sawhney and Ayer [11] proposed a multiple motion estimation method based on a Minimum Description Length (MDL) principle. Their algorithms are computationally expensive and require combinatorial search to determine the correct number of layers and the "projective depth" of each point in a layer. Occlusion regions are not recovered in their layered model. Baker, Szeliski & Anandan [12] proposed a framework for extracting structure from stereo which represents a scene as a collection of approximately planar layers. Each layer consists of an explicit 3D plane equation, a texture map (a *sprite*), and a map with depth offsets relative to the plane. The initial estimates of the layers are recovered using techniques from parametric motion estimation, and then refined using a re-synthesis algorithm which takes into account both occlusion and mixed pixels. For more complex geometry, a *layered depth image* (LDI) has been proposed [13] which is a representation of a scene from a single input camera view, but with multiple pixels along each line of sight.

The goal of our work is to construct a layered and panoramic representation of a large-scale 3D scene with occlusion from translating video sequences. Our approach in this paper is based on a multi-perspective panoramic view image (PVI) [7] and a set of epipolar plane images (EPIs) [14] that are extracted from a long image sequence, and a panoramic depth map which is generated by analyzing the EPIs[15]. So we are actually solving four problems: 1) how to generate seamless PVIs and EPIs from video under a more general motion than a pure translation;

2) how to analyze the huge amount of data in EPIs robustly and efficiently to obtain dense depth information; 3) how to enhance resolution and recover occlusions in a PVI representation; and 4) how to represent a large scale 3D scene with occlusions efficiently and compactly. While the first two issues are very important in constructing a 3D model of a scene (which has been discussed in our previous work [15]), this paper will focus on the other two issues. We propose a new compact representation - 3D layered, adaptive-resolution and multi-perspective panorama (LAMP). The motivation for layering is to represent occluded regions and the different spatial resolutions of objects with different depth ranges; meanwhile the model is represented in the form of a seamless multi-perspective mosaic with viewpoints spanning a large distance. Two kinds of 3D LAMP representations are constructed, the relief-like LAMP and the image-based LAMP. The 3D LAMP representation is capable of synthesizing images of new views within a reasonably restricted but arbitrary moving space, as its intensity and depth maps contain almost all the information from the original image sequences.

2. Basic 3D Panoramic Geometry

For completeness, we give a brief introduction to the constructions and representations of Panoramic View Images (PVIs) and Epipolar Plane Images (EPIs). Let us consider the situation in which a model of a large-scale scene will be constructed from a long and dense video sequence captured by a camera moving in a straight line whose line of sight is perpendicular to the motion. The resulting sequence obeys the following spatio-temporal (ST) perspective projection model

$$x(t) = f\frac{X+Vt}{Z}, y(t) = f\frac{Y}{Z} \qquad (1)$$

where (*X*,*Y*,*Z*) represents the 3D coordinate of a point at time *t*=0, *f* is the focal length, and *V* is the vehicle's speed . A feature point (*x*,*y*) forms a straight locus and its depth is

$$D = Z = f\frac{V}{v} = f\frac{Vdt}{dx} \qquad (2)$$

where $v = dx/dt$ is the slope of the straight locus. In other words, two kinds of useful 2D ST images can be extracted (1) Panoramic View Images (PVIs) – the *y*-*t* intersections in the *xyt* cube, which possesses most of the 2D information of the scene, and (2) Epipolar Plane Images (EPIs) – the *x*-*t* intersections in the *xyt* cube, whose ST locus orientations represent depths of scene points. Fig. 1 shows two PVIs that are extracted from *x*=0 and *x*=-56 of a 128*128*1024 *xyt* image cube of a building scene. Multi-perspective PVIs provide a compact representation for a large-scale scene, and stereo PVIs can be used to estimate the depth information of the scene [7, 6, 16]. In panoramic stereo (Eq. (2), Fig. 1), the "disparity" *dx* is fixed and the distance *D* is proportional to *dt*, the stereoscopic "displacement" in *t*, which means that depth resolutions are the same for different depths. However, we still face two problems in order to use stereo PVIs to recover 3D information – the correspondence problem and the occlusion problem.

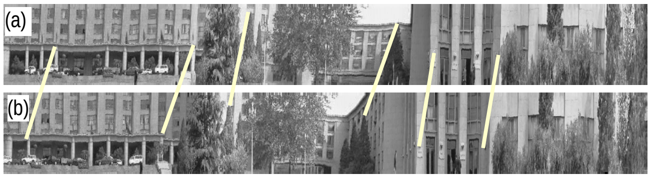

Fig. 1. Stereo PVIs:(a) x = 0; (b) x= -56. White lines indicate matches.

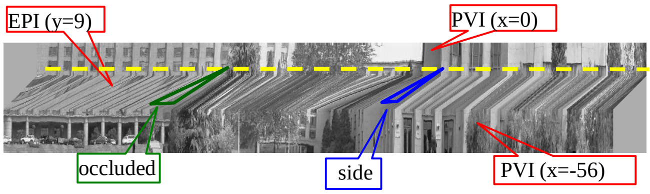

Fig. 2. Real 2D ST intersections

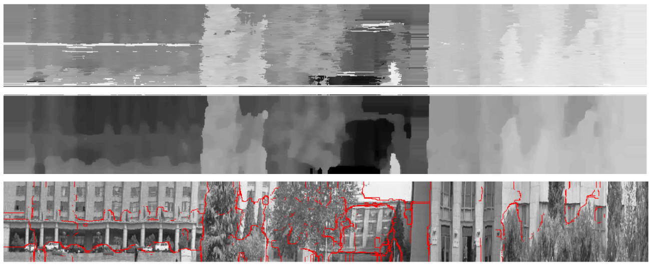

Fig. 3. 3D PVI construction : raw depth map, refined depth map and the texture maps with depth boundaries superimposed.

Our solution to these two problems is to effectively use the information that is continuous between two views, i.e. that of the epipolar plane images. Fig. 2 shows an EPI from which the occluded (and side) regions as well as depths can be recovered. In a real application, the motion of a camera can be a dominant translation plus unpredictable fluctuations. A stabilized image sequence with a 1D translation can be generated by image stabilization and image mosaicing [15]. While this 1D motion model will be used to develop our approach, LAMP representations can be constructed under a more general or different motion [5,6,16] and/or using other approaches [10-12].

3. Occlusion and Resolution Recovery

Our panoramic EPI approach [15] focuses on a "supporting" PVI (e.g. when x=0) and selectively uses the most essential information from all the EPIs. It consists of four important steps: locus orientation detection, motion boundary localization, depth-texture fusion and occlusion/resolution recovery. The results of the first three steps are a panoramic depth map with accurate depth boundaries as well as the corresponding texture map (Fig. 3). This section will present effective methods for occlusion and resolution recovery.

3.1. Occlusion recovery

Because the supporting PVI only contains information from a single viewing direction (for example, the direction perpendicular to motion direction), some parts of the scene that are visible in other parts of images from an original (or a stabilized) video sequence are missing due to occlusion. They will be recovered by analyzing depth occlusion relations in the EPIs. The basic algorithm is performed in each EPI after the panoramic depth map and its depth boundaries have been obtained. The algorithm consists of the following steps (Fig. 4, Fig. 5):

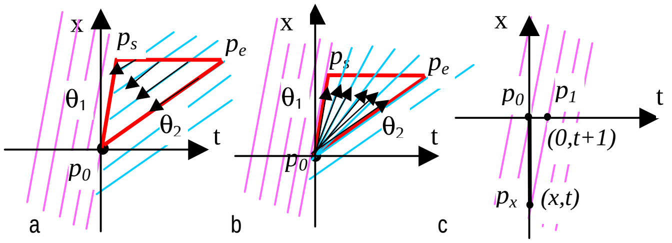

Fig. 4. Region classification and adaptive resolution. (a). locus patterns near an occlusion boundary (b). locus patterns of front- side surfaces (c). temporal resolution enhancement by using spatial resolution

Step 1. *Find the location of a depth boundary* – A point on a depth boundary, $p_0(x_0,t_0)$, and orientation angles (θ_2 and θ_1) of the occluded (far) and occluding (near) objects are encoded in the depth map. A point is considered as on the depth boundary with depth discontinuity, e.g. ($|\theta_1 - \theta_2| > 2^\circ$).

Since it is very important to accurately localize the depth boundary, we will give a brief summary on how it is achieved in the first three steps of our panoramic EPI approach [15], without explicitly tracking the loci in EPIs. First, a *Gaussian-Fourier Orientation Detector* (GFOD) is performed along a scanline (x=x_0) in the EPI, which is the intersection line of this EPI with the supporting PVI. The GFOD operator uses Gaussian-windowed Fourier transform to detect orientations of the image under the Gaussian window. A large window (e.g. 64x64) is used in order to detect accurate locus orientations. Multiple orientations will be detected for a certain temporal range when the GFOD operator moves across a depth boundary. Thus the Gaussian window is applied to reduce this range. However, the response of multiple (two in our current implementation) orientations does not only happen exactly at the point on the depth boundary. So in the second step, a *Motion Boundary Localizer* (MBL) is used to accurately localize the depth boundary. It measures intensity consistencies along the two detected orientations, taking occlusion/ reappearance relations into account. The best consistent measurement should be achieved right at the depth boundary since otherwise one of the measurements will cross both locus patterns (see Fig. 4 and Fig. 5). Then in the third step, a *depth-texture fusion* (DTF) algorithm is applied to reduce the errors produced in each PEI analysis (due to aperture problems, noises and etc.) and to refine depth boundary localizations. The refinement is based on the observation that a depth boundary almost always coincides with a intensity boundary in a visual scene. Fig. 3 compares the raw depth map (after the first two steps) and the refined depth map, and shows superimposed depth boundaries in the panoramic texture map.

Step 2. *Localize the missing part* – The missing (occluded) part is represented by a 1D (horizontal) spatio-temporal segment p_sp_e in the EPI, on which points have the same viewing directions but from moving viewpoints. It is calculated from the slopes of the two orientation patterns that have generated the depth boundary, and is denoted by an x coordinate and start/end frames (t_S/t_E). Basically the largest possible angle of viewing direction (the x position in the EPI) from that of the PVI (i.e. the x_0 coordinate) possesses the most missing information, but possible occlusions by nearby objects should be considered. For example, the second missing region from the right in Fig. 5b was determined by checking the occlusion of the locus patterns of the missing part against those of other nearby foreground objects (trees), resulting in an ST segment with smaller x coordinate, i.e. smaller viewing angle from that of the PVI. In this way a triangular region $p_0p_sp_e$ can be determined, and the 1D segment p_sp_e will be used as the texture of the missing part that is occluded by the foreground objects.

Step 3. *Verify the type of the missing part*. The triangular region also contains depth information of the missing part - the 1D segment p_sp_e. In principle, similar treatments can be made here as for the basic depth map. For simplicity, the missing parts are classified into two types in our current implementation: OCCLUDED and SIDE. If the loci within the triangular region form a parallel pattern of angle θ_2(Fig. 4a), then the missing part is classified as OCCLUDED, otherwise as SIDE if the angle of the oriented pattern

$$\theta_t = \theta_1 + \frac{\theta_2 - \theta_1}{t_e - t_s}(t - t_s) \qquad (3)$$

changes linearly inside the triangle (Fig. 4b). By assuming the missing part as each of the two types, an overall consistent measurement along the hypothesized locus orientations (indicated by arrows in Fig. 4a and b) can be calculated within the triangle region (similar measurements as in motion boundary localization [15]). The type of missing part is selected as the one with better consistent measurement of the two hypotheses. The loci's angle θ of the OCCLUDED region will be the same angle θ_2 as the occluded object, whereas loci's angle θ of the SIDE region gradually changes from θ_1 to θ_2 (or from θ_2 to θ_1), as expressed in Eq. (3). In this way the depths of the occluded or side region can be assigned. Fig. 4 only shows the situation of reappearance; readers can refer to Fig. 5 for other situations (occlusion and side).

3.2. Resolution recovery

The panoramic view image (PVI) in the *t* direction has been under-sampled if the image velocity *v* of a point in the PVI is great than 1 pixel/frame (see the right hand part of Fig. 3). To enhance temporal resolution, a *v*-pixel segment in the *x* direction is extracted from the EPI (as opposed to a single pixel in a PVI, for example, as was done for Fig. 1). Fig. 4c shows the principle. The nice feature is that the end point $p_x(x,t)$ of an x-segment p_0p_x at time *t* will exactly connect with the start point $p_1(0,t+1)$ of the x-segment at time *t+1* since they are on the same locus. This property will be used to generate seamless, adaptive-time panoramas. The thickness of the dark (i.e. red in color print) horizontal lines (including those of SIDE and OCCLUDED regions) in the two EPI tiles in Fig. 5 indicates the number of points (pixels) to be extracted in the *x* direction of this epipolar plane image. Fig 6 shows a seamless panoramic mosaic after resolution enhancement, where much higher temporal resolutions are achieved in the second part of the mosaic.

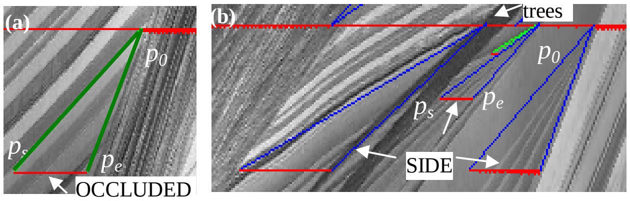

Fig. 5. Occlusion and resolution recovery results in real EPIs. (a) an OCCLUDED region; (b) three SIDE regions (two of them belong to a side facade separated by trees).

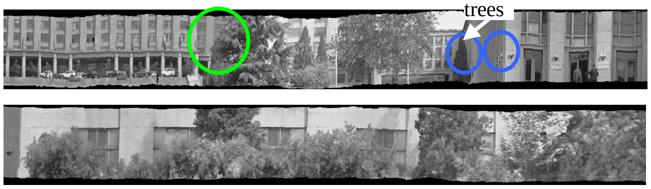

Fig. 6. Multi-viewpoint mosaic with adaptive time scales (The right edge of the upper part connects with the left edge of the bottom part). The width of each vertical slice from the corresponding original frame is determined by the dominant image velocity *v* of pixels along the *y*-axis in the corresponding PVI. Circles show the corresponding OCCLUDED and SIDE regions in Fig. 5.

4. Relief-like 3D LAMP Representation

Based on a 3D PVI and results of occlusion/resolution recovery, we propose a compact and comprehensive representation called **3D LAMP**- *Layered, Adaptive-resolution and Multi-perspective Panorama*. We will explain the LAMP model by an illustrative example in Fig. 7 where only a 1D scene is shown. Recall that a 3D ST image (a 2D EPI for a 1D scene in Fig. 7) includes everything from an image sequence. First, we will give a basic LAMP representation - *relief-like 3D LAMP* - that is directly carved from the 3D ST image (xyt image). The relief-like 3D LAMP basically cuts out some essential part of the 3D ST image and further assigns a depth value for each pixel. The following four properties of the LAMP representation make it very suitable for image-based modeling and rendering.

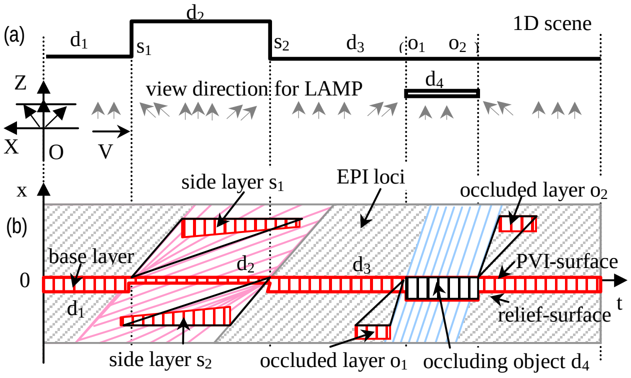

Fig. 7. A 1D illustrative scene and its LAMP representation. (a) 1D scene with four horizontal depths (d_1-d_4) and two sides (s_1,s_2), (b). relief-like LAMP and depth layering for image-based LAMP from an EPI.

1). It is a *panoramic* image-based representation with 3D information. A large-scale scene will be basically indexed by a seamless 2D panoramic view image (PVI) consisting of both texture and depth maps. This 2D "PVI surface" (see Fig. 3 for a real example) is the supporting (back) surface of the *base layer* in the relief-like LAMP, which makes the LAMP representation very efficient for archives (modeling) and retrieval (rendering).

2). It is a *multiple perspective* image. The PVI is a multi-viewpoint perspective image (y-t image). Each sub-image (one-column slit-image in concept) in a multi-perspective panorama is full perspective, but successive slit images have different viewpoints. The multi-perspective representation acts as a bridge between the modeling side from original perspective sequences and the rendering side for new perspective images, both with changing viewpoints over a large distance.

3). It has *adaptive* image resolution. In the base layer (in Fig. 7b, it includes depths d_1, d_2, d_3 and d_4), each point has an extension of *v*-pixel texture and depth information in the x direction to represent the resolution loss in the PVI when the slope of the locus is greater than 1 (*v*>1). In other words, the number of pixels ("temporal resolution") at each point in the PVI surface *adaptively* changes with the depth of that point. The adaptive temporal sampling horizontally and the inherent adaptive spatio-sampling of perspective projections vertically recover and preserve image resolutions of the original frames in a satisfying way.

The name "relief-like" comes from the fact that the appearance of the front surface (*relief surface*) of this representation is somewhat like a *relief sculpture*, in which forms and figures (with image velocities $v > 1$) are distinguished from a surrounding plane surface (PVI surface) (Fig. 7b; see Fig. 9 and Fig. 10 for real examples). Each pixel in a relief-like LAMP is associated with a location with (x,y,t) coordinates; for the PVI surface, we have x = 0. The end point in the "relief " surface in location (y,t) is exactly connected with the start point in the PVI surface in location (y, $t+1$) (see Fig. 4c, Fig. 10a). This nice feature allows us to generate seamless mosaics (see Fig. 6 and Fig. 13a).

4). It is a *layered* representation. Additional occluded layers, which are pieces of multi-perspective view images instead of a complete PVI-based layer, represent the occluded and side regions (region o_1, o_2, s_1 and s_2 in Fig. 7b). They are attached to the base layer with the same representation (texture, depth and adaptive resolution) as the base layer. Each of the occluded x-segments is attached to a depth boundary point (y,t_0), and has an x coordinate, a start time t_s and an end time t_e that mark its position in the *xyt* image (Fig. 7b; see also Fig. 4 and Fig. 5).

In conclusion, a relief-like LAMP is composed of a "complete" base layer and a set of occluded layer pieces. It can be viewed as an essential part of the xyt image in which each pixel has two attributes – texture I and depth v – connected to its coordinates (x,y,t). The 3D coordinates of each pixel (x,y,t) in the 3D LAMP can be recovered by the following equation

$$Z = F\frac{V}{v},\ Y = y\frac{V}{v},\ X = x\frac{V}{v} - Vt \qquad (4)$$

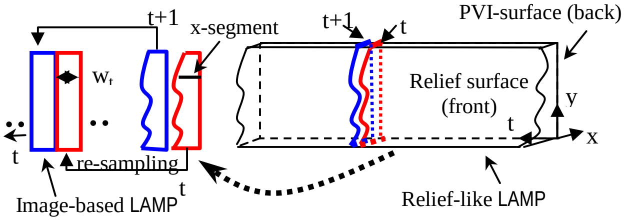

Fig. 8 . Temporal re-sampling and seamless mosaicing

5. Image-based 3D LAMP Representation

A relief-like LAMP can be viewed as having adaptive time sampling for every single pixel in the PVI surface and includes all the occluded regions. This representation is good for a complex scene that has many depth layers. However, in representation, the relief-like LAMP is an inhomogeneous representation rather than a set of 2D image arrays. In addition, the base layer usually includes objects at different depth levels, and the occluded layers are not merged into the regions they belong to (e.g. regions o_1 and o_2 in Fig. 7). A natural extension of the basic LAMP is to extract from it a more concise representation - an *image-based 3D LAMP*. In an image-based 3D LAMP, a scene is represented as multiple layers of 2D mosaiced images, in which each layer is represented by two 2D arrays - a texture map and a depth map. We wish to create a panoramic mosaic image for each layer that is both seamless and preserves adequate image resolutions. There are two steps necessary to fulfill this goal: depth layering and time re-sampling.

5.1. Depth layering

The motivation for layering is to represent occluded regions and different spatial resolutions of objects with different depth ranges in different layers. An image based LAMP is layered according to occluding relations rather than merely depths. The scene parts with varying (but not discontinuous) depths in a single layer will use *adaptive* temporal sampling rates to represent different resolutions (Section 5.2). Conceptually, the implementation of depth layering from a relief-like LAMP is straightforward since we know where the occluding regions in the *base layer*, and occluded or side regions in the *occluded layers,* should be put in the *image-based layers*. Generally speaking, layers either in the same depth range or with continuous depths will be merged into one single layer. Regions with occluding boundaries will be divided. For example, in Fig. 7b, the two occluded regions (o_1 and o_2) will be merged into the base layer with depth d_3, while the occluding region (d_1) originally included in the base layer will be separated out as a new layer. Ideally, side regions (s_1 and s_2 in Fig. 7b) should be inserted in the base layer since they can connect well (in both depth and texture) with the base layer. However, in our current implementation, we put them into two separate layers for simplicity in image-based representation. After depth layering, the basic LAMP is divided into several relief-like LAMPs ready for creating image-based layers. Each of them includes a base layer, with an x coordinate to indicate where it is taken from in a *xyt* cube. Different x coordinates in the relief-like LAMP imply different viewing directions of parallel projections in order to better capture surfaces with different orientations(see Fig. 7a).

5.2. Temporal re-sampling.

From each single-layer relief-like LAMP, we generate a seamless 2D panoramic image with both texture and depth attributes. In each column (t) of such a relief-like LAMP (Fig. 8), the number of pixels in the x direction of a point in the PVI surface is inversely proportional to the depth of that point. We want to warp each x-y slice with varying widths in the x direction into a t-y slice of equal-width w_t ($w_t \geq 1$) in the t direction, which will be stitched into a 2D seamless mosaic (Fig. 8). The width of the t-y slice image in time t can be selected adaptively as the dominant image velocity of all $v(y, t)$ in column t of the PVI-surface, e.g.

$$w_t = \bar{v}(t) = \underset{y}{median}\{v(y,t)\} \qquad (5)$$

Note that each x-segment of the x-y slice in a relief-like LAMP starts from x_s=0 and end at $x_e(y,t)$=-$v(y, t)$. A temporal re-sampling is performed for each x-segment by turning it into a t-segment of w_t-pixels (Fig. 8 and Eq. (5)).

In this way, super-time sampling is achieved in frame *t* such that each transformed w_t-pixel slice has a time unit of $1/w_t$. Hence the texture map and the depth map of a layer starting from time t_0 are represented as *I(y,k)* and *v(y,k)* where index *k* for time *t* is

$$k \in [\sum_{\tau=t_0}^{t-1} \bar{v}(\tau),\ \sum_{\tau=t_0}^{t-1} \bar{v}(\tau) + \bar{v}(t)],\ \bar{v}(t_0) = 0,\ t > t_0 \qquad (6)$$

For computing 3D coordinates, the super-sampled t_k corresponding to column *k* is stored in a 1D array as part of the image-based LAMP model:

$$t_k = t + \frac{k - \sum_{\tau=t_0}^{t-1} \bar{v}(\tau)}{\bar{v}(t)} \in [t, t+1),\ \bar{v}(t_0) = 0,\ t > t_0 \qquad (7)$$

In this manner, each column *k* is virtually a 1-column perspective image at the viewpoint (time) t_k. The densities of viewpoints adaptively change with depths of the scene. The parallel-perspective views between time *t* and *t+1* are approximated by transforming the perspective image in frame *t*. Note that this is a correspondenceless approach to implementing view interpolation different from the local match method for image mosaicing in [16]. Fig. 6 shows this idea in a seamless adaptive-resolution panoramic mosaic where the time scale in each instant *t* is determined by the dominant (median) depth of points along the corresponding vertical line (the *y* direction) in frame *t*. Results with occlusion will be given in the next section (see Fig. 13).

In conclusion, in an image-based 3D LAMP, each layer is basically a 2D parallel-perspective panorama with an x coordinate (to indicate viewing direction), and has three components: 1) texture map I(y,k); 2) depth map v(y,k); and 3) a 1D super time-sampling array $t_k=t(k)$ (to indicate densities of viewpoints, or adaptive temporal resolutions). From this representation, we can find the corresponding 3D coordinates of a point *(y,k)* in the 3D LAMP by

$$Z = F\frac{V}{v},\ Y = y\frac{V}{v},\ X = x\frac{V}{v} - Vt_k \qquad (8)$$

6. 3D LAMP Construction and Rendering

6.1. 3D LAMP construction results

Fig. 9 and Fig. 10 show the results of constructing a relief-like LAMP for the main building (MB) sequence. The relief surface of the base layer of the LAMP model is shown in Fig. 9, which corresponds to the PVI-surface in Fig. 3. Recall that a relief-like LAMP representation is part of a 3D *xyt* cube. That is to say, for each pixel *(y,t)* in the PVI surface, there is a *v*-pixel *x*-segment in the x direction. The internal data of the base layer is displayed in Fig. 10a as the sequential head-tail arrangement of all the segments of the base layer in each row. It can be found that both the texture and the depth map show almost seamless connections between successive segments, and it is obvious that higher resolutions than in the corresponding PVI are recovered for nearer objects. The internal data of the relief-like LAMP, including all the occluding layers, is displayed in Fig. 10b as the similar head-tail arrangements of all the *x*-segments in all the layers, where the *x*-segments in the occluded layers are inserted into the locations of their corresponding depth boundaries (shown as red dots in Fig. 10). Comparing Fig. 10b to Fig. 10a, more data in the occluded regions are included. For example, the portion of the building's facade occluded by the trees and the side facades of the building are partially recovered .

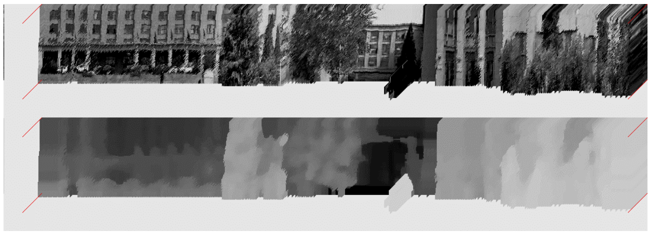

Fig. 9. Texture and depth of the relief surface in the base layer of the relief-like LAMP of the MB sequence

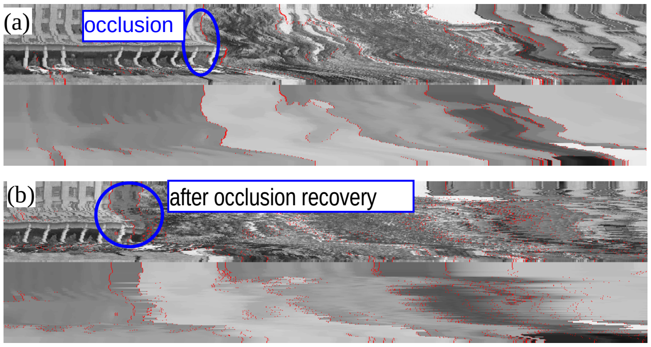

Fig. 10. Internal data of (a) the base layer, and (b) all the layers of (part of) the LAMP of the MB sequence.

We have shown in [15] how to make full use of the original image sequence by generating an extended panoramic image (XPI). Suppose that an image sequence has F frames of images of size W×H. An example is the frequently used flower garden (FG) sequence(W×H×F = 352×240×115). A PVI and an EPI is shown in Fig. 11a and Fig. 11b. It is unfortunate in this case that the panoramic view image turns out to be "narrow" due to the small number of frames and large interframe displacements. Therefore an extended panoramic image (XPI) is constructed, which is composed of the left half of frame m/2 (m is the GFOD window size), the PVI part formed by extracting center vertical lines from frame m/2 to frame F-m/2, and the right half of frame F-m/2 (Fig. 11c). Fig. 12 shows the results of 3D recovery of the XPI of the FG sequence. In the depth map, the tree trunk stands out distinctly from the background, and the gradual depth changes of the flower bed are detected. The occluded regions and resolutions are recovered in a way similar to that for a PVI image, except that the EPI analysis is performed along a zigzag line (Fig. 11b).

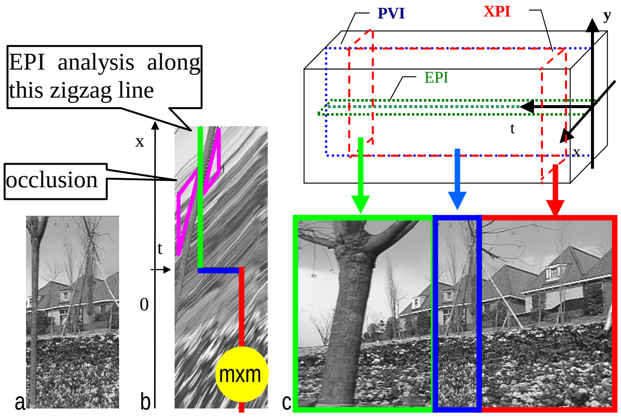

Fig. 11 PVI, EPI and XPI of the flower garden (FG) sequence

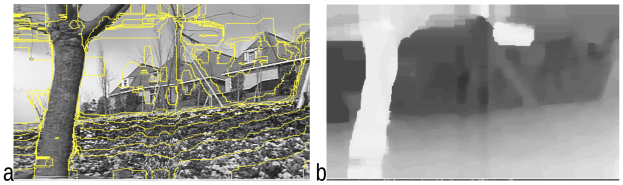

Fig. 12. Panoramic depth map for the FG sequence. (a) Isometric depth lines overlaid in the intensity map (b)panoramic depth map.

Fig. 13 shows the two extracted layers of the image-based LAMP representation for the FG sequence, each of which has both texture and depth maps. These two pairs of images, along with a 1D temporal sampling rate array (Eq. (7)) for each layer, are constructed from the 115-frame FG image sequence. In the *yt* part of the background layer of this extended image-based LAMP, the time-sampling rate is adaptively changed according to the dominant depth in each time instant. The time scales are less than 0.5 (frames); this means that higher resolution is achieved than that of the original PVI (shown in Fig. 12) in both texture and depth maps (the un-even super-timing in Fig. 13a is due to quantization). Furthermore, the background regions occluded by the tree trunk have been completely recovered, and have been merged into the background layer with both texture and depth values.

6.2. LAMP-based Rendering

The 3D LAMP model is capable of synthesizing images from new viewpoints differing from the viewpoints of the original image sequence thanks to its multi-perspective, adaptive resolution, occlusion representation and its depth information. The mapping from a pixel in a relief-like LAMP model to its 3D coordinate can be computed by using Eq. (4) and then it is straightforward to re-project it to the desired view (in practice an inverse mapping should be applied). Because of neighborhood relations of pixels in the LAMP representation (refer to Fig. 10), a rendering algorithm can easily perform interpolation between neighborhood pixels. In a relief-like LAMP, an attached layer is always occluded by the layer to which it is attached. So when rendering, the attached layers should be drawn first so that an occluded region could be seen correctly in a new view. Thanks to the adaptive resolution representation, the higher resolutions in the original images can be applied to a new view whose viewpoint is close to those of the original image sequence.

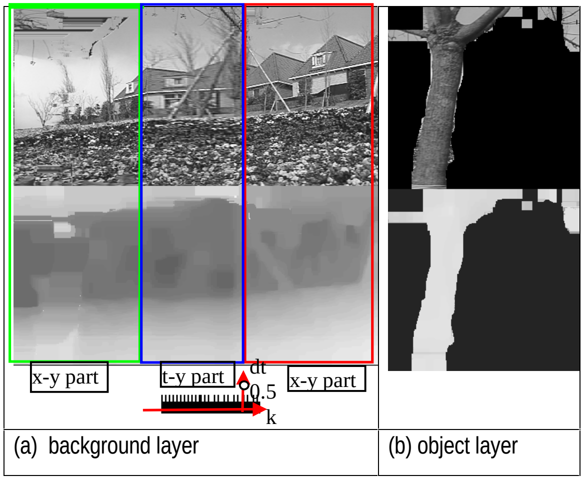

Fig. 13. Image-based LAMP model of the FG sequence

The rendering process is easier based on an image-based LAMP model. For a LAMP model based on an XPI (i.e., a combination of x-y and y-t images) in general, Eq. (4) and Eq.(8) should be used in the x-y parts and y-t part respectively. Fortunately, both equations are very simple and virtually the same (but reflect the different ways to obtain *x* and *t* indices) so that a fast implementation is feasible. In order to achieve a correct occlusion relation, a rendering from a new view should begin with the farthest layer and end with the nearest layer from the viewpoint of a synthetic image. Synthetic sequences with a virtual camera of 6 DOF motion generated from the LAMP model (with and without the object layer) of the flower garden sequence can be found on our web page [17].

7. Comparisons and Discussions

There are at least three existing image-based representations of large image sequences, which are based on parallel projection (i.e., a textured digital elevation map [DEM]), single-view mosaicing (e.g. Sprite) and parallel-perspective panorama (i.e. PVI). A parallel projection of the 3D PVI in Fig. 3 of the BUILDING sequence is shown in Fig. 14a that correctly shows the aspect ratio of depth and size (height and width) of the scene. However, the resolutions of the nearer objects are significantly decreased because of the evenly-sampled representation. A single view perspective representation, on the other hand, is not a good viewing method for surfaces of varying orientations and over a large distance (Fig. 14b). A very small depth estimation error in the view-based coordinates of a video frame could be greatly enlarged when re-projecting to a mosaicing image of a single referenced viewpoint.

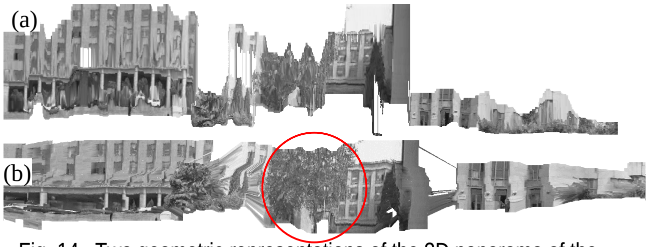

Fig. 14. Two geometric representations of the 3D panorama of the BULIDING sequence. (a). Parallel projection; (b) Perspective projection from a viewpoint at the center of the path where the video was captured

A parallel-perspective panoramic view image (PVI) is a nice image-based representation that naturally represents the image data captured by a camera translating over a long distance. Based on multi-perspective panorama, we have proposed a still compact and more powerful representation - 3D layered, adaptive-resolution and multi-perspective panorama (LAMP). To our knowledge, this seems to be the first piece of work that integrates multi-perspective panoramas and layered representations with adaptive image resolutions in a unified model. The relief-like LAMP is basically a single extended multi-perspective panoramic view image (PVI) with both texture and depth values, but each pixel has multiple values to represent results of occlusion recovery and adaptive resolution enhancement. The image-based LAMP, on the other hand, consists of a set of multi-perspective layers, each of which has both texture and depth maps, with adaptive densities of viewpoints depending on depths of scene points. No assumption is made on the structures of a scene in constructing its 3D LAMP representations.

The LAMP representation is related to such representations as PVIs, sprites and layered depth images (LDIs). However, it is more than a multi-perspective PVI in that depth, adaptive-resolution and occlusion are added. It is different from a "sprite" (or LDI) since the sprite or LDI is a view of a scene from a single input camera view, and is without adaptive image resolution. Interesting readers can compare our results [17] to those of other researchers [10, 11, 13]. A comparison in number of input images, results of layered representation, and algorithm performance for the flower garden sequence is summarized in Table 1.

The 3D LAMP representations are concise, practical and powerful representations for image-based modeling and rendering. In future work we are going to extend the 3D LAMP model to represented a large-scale scene when the camera moves along more general paths.

Acknowledgements

This work is partially supported by China National Science Foundation under contact No. 69805003.

References

[1]. Chen, S. E. 1995. QuickTime VR - an image based approach to virtual environment navigation. In *SIGGRAPH 95*: 29-38.

[2]. Shum, H.-Y. and Szeliski, R. 1997. Panoramic Image Mosaics. *Microsoft Research, MSR-TR-97-23*

[3]. Sawhney, H. S. R. Kumar, G. Gendel, J. Bergen, D.Dixon, V. Paragano, 1998. VideoBrush^{TM}: Experiences with consumer video mosaicing. In *WACV'98*: 56-62.

[4]. Ishiguro, H., Yamamoto, M. and Tsuji S. 1990, Omni-directional stereo for making global map. In *CVPR'90*: 540-547.

[5]. Peleg, S. and Ben-Ezra, M. 1999. Stereo panorama with a single camera. In *CVPR'99*: 395-401.

[6]. Shum, H.-Y. and Szeliski, R. 1999. Stereo reconstruction from multiperspective panoramas. In *ICCV'99*: 14-21.

[7]. Zheng, J. Y. and Tsuji, S. 1992. Panoramic representation for route recognition by a mobile robot. *IJCV*, 9(1): 55-76.

[8]. Peleg, S. and Herman, J. 1997. Panoramic mosaics by manifold projection. In *CVPR'97*: 338-343.

[9]. Rademacher, P. and Bishop, G. 1998. Multiple-center-of-projection images. In *SIGGRAPH'98*: 199-206.

[10]. Wang, J. and Adelson, E. H. 1994. Representation moving images with layers. *IEEE Trans. on IP*, 3(5): 625-638.

[11]. Sawhney, H. S. and Ayer, S. 1996. Compact representation of videos through dominant and multiple motion estimation. *IEEE Trans. PAMI*, 18(8): 814-830.

[12]. Baker, S., Szeliski, R. and Anandan, P. 1998. A layered approach to stereo reconstruction. In *CVPR'98:* 434-441.

[13]. Shade, J., Gortler, S., He. L. and Szeliski, R. 1998. Layered depth image. In *SIGGRAPH'98:* 231-242.

[14]. Bolles, R. C., Baker, H. H. and Marimont, D. H. 1987. Epipolar-plane image analysis: an approach to determining structure from motion. *IJCV*, 1(1): 7-55.

[15]. Z. Zhu, G. Xu, X. Lin, Panoramic EPI generation and analysis of video from a moving platform with vibration, In *CVPR'99*: 531-537.

[16]. Z. Zhu, E. M. Riseman, A. R. Hanson, Parallel-perspective stereo mosaics, In *ICCV'01*, Vancouver, Canada, July 2001.

[17]. Z. Zhu, 3D LAMP for image-based rendering, http://vis-www.cs.umass.edu/~zhu/panorama3D.html.

Table 1. Comparison of 4 layered representation results for the flower garden sequence

	Wang-Adelson [10]	**Shawney-Ayer [11]**	**Baker-Szeliski-Anandan[12]**	**Image-based LAMP**
Input	first 30 frames (720x480)	a few frames	first 9 even frames	all 115 frames (350x240)
Rep.	3 planar layers (tree, flower bed, house)	4 layers (tree, flower bed, house, sky)	6 layers of Sprites (3 tree, 2 flower bed, 1 house)	2 layers of LAMP (tree, background)
Depth	no	no	yes	yes, each value from 64 frames
Occlusion	recovered	not recovered	recovered	recovered
Multi-view	Affine mosaic rep.	single view mosaic	single view mosaic	Multiple-view mosaic
Adaptive res.	no	no	a perspective image	adaptive resolution
Performance	40 mins / 30 frames in a HP 9000 series 700 workstation	not available	not available	14 mins / 115 frames in a 400 MHz PC (general C code)