Stereoscopic Video Synthesis from a Monocular Video

Guofeng Zhang[1] Wei Hua[1*] Xueying Qin[1] Tien-Tsin Wong[2] Hujun Bao[1*]

[1]State Key Lab of CAD&CG, Zhejiang University
{zhangguofeng, huawei, xyqin, bao}@cad.zju.edu.cn

[2]The Chinese University of Hong Kong
ttwong@cse.cuhk.edu.hk

***Abstract*— This paper presents an automatic and robust approach to synthesize stereoscopic videos from ordinary monocular videos acquired by commodity video cameras. Instead of recovering the depth map, the proposed method synthesizes the binocular parallax in stereoscopic video directly from the motion parallax in monocular video. The synthesis is formulated as an optimization problem via introducing a cost function of the stereoscopic effects, the similarity, and the smoothness constraints. The optimization selects the most suitable frames in the input video for generating the stereoscopic video frames. With the optimized selection, convincing and smooth stereoscopic video can be synthesized even by simple constant-depth warping. No user interaction is required. We demonstrate the visually plausible results obtained given the input clips acquired by ordinary hand-held video camera.**

***Index Terms*— Stereoscopic video synthesis, parallax, optimization.**

I. Introduction

STEREO visualization provides users the important depth cue experienced in our daily life. Since the introduction of the parallax principle of stereo [1], various stereoscopic systems for displaying stereoscopic images and videos have been developed. Examples include the recently developed 3DTV system [2].

However, stereoscopic videos are normally inaccessible by general public due to the difficulty in generating stereoscopic videos. Acquiring stereoscopic videos from real world usually requires specialized devices. In addition, processing the captured videos requires specialized software or hardware and specialized skills. On the other hand, low-cost ordinary monocular video cameras are widely available. In this paper, we propose an automatic and efficient video-based rendering method to synthesize stereoscopic videos from the monocular videos. Although not all kinds of monocular videos can be used to synthesize stereoscopic videos, many are feasible, e.g. aerophotographic video.

A monocular video can be regarded as a set of plenoptic samples of the scene [3]. The synthesis of stereoscopic videos is basically a process of determining the proper samples and compositing them to give the left- and right- view sequences. Our method assumes the camera motion contains translational movement and the scene is fixed.

To synthesize stereoscopic videos, one may recover the depth values of samples, and reproject the samples to synthesize both views for each frame. This approach strongly relies on the accuracy of recovered depth values which in turn strongly depends on the availability of textures in the scene. Moreover, when the scene exhibits mirror reflection or highlight, the accuracy of depth recovery is even lowered. Our major contribution is to make use of the *motion parallax* in the monocular video and convert it to *binocular parallax* in a robust way, instead of explicitly recovering the dense depth maps. The whole process is done automatically. To synthesize realistic stereoscopic video, we formulate it as an optimization problem with an objective function that measures the loss of stereoscopic effects, similarity, and smoothness constraints. With the optimally selected frames, convincing stereoscopic video can be synthesized by simple view warping (Figure 1).

There are 3 major steps in our method. Firstly, we track the camera motion in the monocular video by a robust camera-tracking algorithm. Secondly, an iterative optimization algorithm is performed to determine the most suitable mono-frames for stereoscopic video synthesis. It selects two sequences of frames from the monocular video. The *i*-th frames in the two sequences are then warped into the binocular views corresponding to the *i*-th desired eyes (left and right) in the final step. Our major contribution is the optimization in the second step. It minimizes a cost function with the following objectives:

- The selected frames exhibit the most realistic stereoscopic effects after warping.
- The warped views are similar to the original ones.
- The synthesized stereo frames are smooth temporally.

II. Related Works

Early work in stereoscopic video generation employs 3D geometry [4]. However, 3D models are usually difficult to obtain for real-world scene. Generating stereo views from monocular video sequences can be achieved by first recovering the depth map [5], [6]. There have been many work in recovering depth in the area of computer vision. Stereo reconstruction [7], two-view or multi-view reconstruction [8], [9] have been proposed. However, fully automatic, context-independent, and accurate dense 3D reconstruction is still an open problem.

Image-based rendering [10], [11] aims at synthesizing novel views from images. Methods like light field [12], [13] and lumigraph [14] densely sample the scene in order to synthesize reasonable novel views even no geometry information is

*Corresponding Authors: Wei Hua and Hujun Bao

given. Other methods try to reduce the sampling rate by incorporating depth information or coarse 3D models. They include 3D warping [15], view interpolation [16], view morphing [17], image tours [18], and layered-depth images [19]. Sawhney et al. [20] synthesized high-resolution stereoscopic video given one high-resolution and one low-resolution views. Recent work in video-based rendering [21] utilizes multiple synchronized video cameras to generate the 3D video [22], or free viewpoint video [23]. Their goals are to synthesize arbitrary novel views. However, specialized hardware and/or reconstruction of 3D models are usually required. Techniques for stereo panoramic images [24], [25] have been proposed. They stitch images obtained from a single rotating camera mounted on a special rig or equipped with a specialized optical lens.

Homography can be used for rectifying a pair of still images to a stereo pair in stereo vision [26]. However, it may not suit for the video sequence since the change of orientations of rectified stereo pairs may not be smooth, which causes the resultant video looks shaky. Moreover, the baselines (the lines joining the stereo image pairs) of rectified stereo pairs may also not be the same throughout the video. This violates the property of the stereoscopic video. Rotem et al. [27] calculated a planar transformation between images in the sequence and aligned one input frame to another in order to synthesize the stereoscopic video sequence. This relies on the human capability to sense the residual parallax. Since it only uses a simple homograhy without the accurate camera motion recovery, the baseline of a stereo pair may not be calculated accurately, resulting in the length of baseline changes vigorously in the generated stereoscopic video sequence. In addition, there is no attempt to control the parallax errors along vertical direction. Hence there will be shaky motion in the generated stereoscopic video as evidenced by their results.

The proposed work synthesizes stereoscopic video from a monocular video sequence by *utilizing the motion parallax alone*. *No depth map recovery* is required. We make an in-depth analysis based on precise camera motion recovery, and formulate it as an optimization problem of the stereoscopic effects, the similarity, and the smoothness constraints.

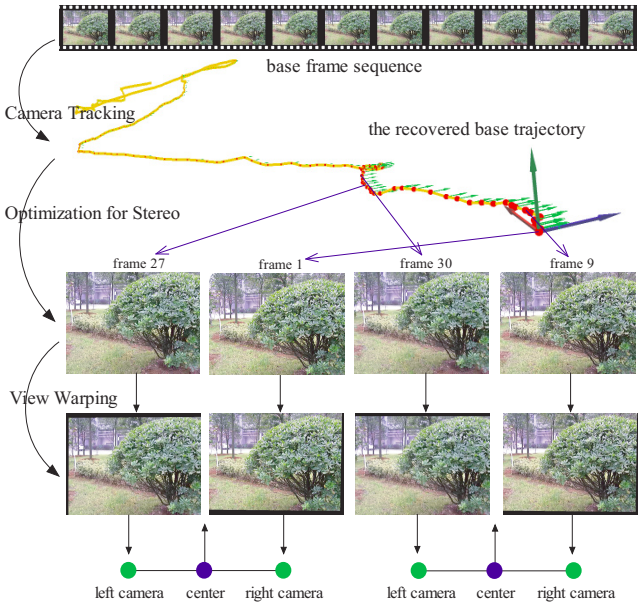


Fig. 1. Synthesizing stereoscopic video from monocular frames. At the bottom of this example, $\Pi^L = 27, 30$ and $\Pi^R = 1, 9$. The first frame in the stereoscopic video is warped from the base frame pair $(\mathbf{f}_{27}, \mathbf{f}_1)$, while the second one is warped from base frame pair $(\mathbf{f}_{30}, \mathbf{f}_9)$. The actual base frames for warping are selected by optimizing the cost function.

Π^L, Π^R	the index subsequences in which the i-th elements $\Pi^L[i]$ and $\Pi^R[i]$ are the indices of the base frames to be warped to the i-th left-eye and right-eye frame in stereoscopic video sequence.
$\mathbf{s}$	a stereo-camera.
$\mathbf{S}$	the ordered set (sequence) of stereo-camera. $\mathbf{S}[i]$ is the i-th stereo-camera, equivalent to $\mathbf{s}_i$.
$\mathbf{b}$	a base camera.
$\mathbf{B}$	the ordered set (sequence) of base camera. $\mathbf{B}[i]$ is the i-th base camera, equivalent to $\mathbf{b}_i$.
$\mathbf{L}(\mathbf{s}), \mathbf{R}(\mathbf{s})$	the left and right cameras of stereo-camera $\mathbf{s}$.
$\mathbf{v}(\mathbf{b})$	the viewpoint of camera $\mathbf{b}$.
$\mathbf{v}(\mathbf{L}(\mathbf{s})), \mathbf{v}(\mathbf{R}(\mathbf{s}))$	the left and right viewpoint of stereo-camera $\mathbf{s}$.
$\mathbf{q}(\mathbf{b}), \mathbf{q}(\mathbf{s})$	the orientation of base camera $\mathbf{b}$ or stereo-camera $\mathbf{s}$ respectively, expressed by Euler angles.
$\mathbf{f}(\mathbf{m})$	the frame corresponding to camera $\mathbf{m}$.
$\mathbf{f}_k$	the k-th base frame, equivalent to $\mathbf{f}(\mathbf{b}_k)$.
$\mathbf{f}(\mathbf{m}_1) \rightarrow \mathbf{f}(\mathbf{m}_2)$	the warped view $\mathbf{f}$ from camera $\mathbf{m}_1$ to that of camera $\mathbf{m}_2$.

TABLE I
NOTATIONS.

III. Overview

Before presenting our algorithm, we first define the terminologies. We call the input monocular video sequence the *base frame sequence*, in which each frame is a *base frame*. The camera corresponding to a base frame is a *base camera*, and its viewpoint and viewing direction are called by *base viewpoint* and *base viewing direction* respectively. The ordered sequence of base viewpoints form a *base trajectory*. A *stereo-camera* consists of two monocular cameras, *left camera* and *right camera*. Both of them are in the same orientation and orthogonal to the line joining them. The center of projections of left and right cameras are called the left and right viewpoints respectively. The center of the *stereo-camera* lies at the midpoint of two cameras. These notations are listed in Table I.

We assume the interocular distance, the distance between the left and right viewpoints, is constant and denoted by d_{eye}. Thus, the extrinsic parameters of the stereo-camera can also be described by its center and the orientation of its viewing coordinate frame. We also assume that the intrinsic parameters of both left and right cameras are the same and unchanged throughout the whole sequence.

Our method exploits the temporal coherence of the monocular video sequence. The novel binocular views are synthesized by warping two properly selected base frames. The warping error between the warped and the true views is small when the difference (in terms of viewing parameters) between the original and target views is small. Thus, we need to carefully determine the center and the orientation of the desired stereo-

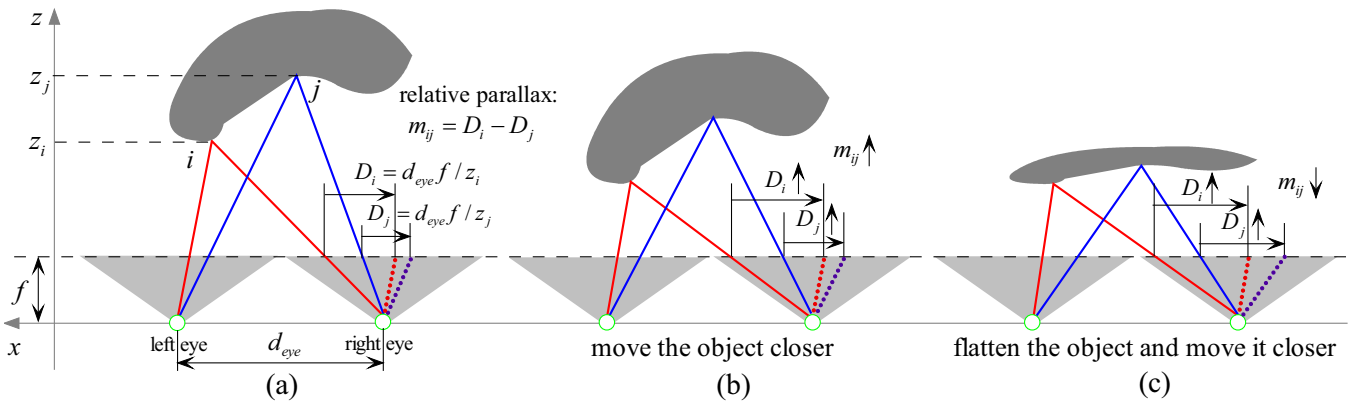


Fig. 2. Absolute parallax and relative parallax. (a) illustrates the absolute and relative parallax. (b) Moving the scene nearer, the absolute parallaxes become larger. (c) Extruding the scene toward the nearer distance, the relative parallaxes become smaller while the absolute parallaxes become larger.

camera, as well as the selection of two base frames, so that the following three objectives are achieved: (a) the binocular views obtained by warping exhibit the stereoscopic effects as realistic as possible; (b) the binocular views are close to the selected base frames; and (c) the change in viewpoint position and orientation of consecutive stereo-cameras are minimized. Otherwise, the generated stereoscopic video will be shaky. We begin the description with the input base frame sequence $\mathbf{F} = \{\mathbf{f}_k | k = 1, ..., K\}$. Here are the three major steps to perform:

- Step 1 For each base frame $\mathbf{f}_k$, we recover the extrinsic parameters of the corresponding base camera $\mathbf{b}_k$ in the set $\mathbf{B} = \{\mathbf{b}_k | k = 1, ..., K\}$.
- Step 2 Determine the stereo-camera sequence $\mathbf{S} = \{\mathbf{s}_i | i = 1, ..., N\}$ and the two index subsequences, Π^L and Π^R, satisfying criteria (a), (b) and (c) (explained in Section IV).
- Step 3 For $i = 1, \ldots, N$, performing view warping operations:

$$\begin{aligned} &\mathbf{f}(\mathbf{b}_l) \rightarrow \mathbf{f}(\mathbf{L}(\mathbf{s}_i)), l = \Pi^L[i] \\ &\mathbf{f}(\mathbf{b}_r) \rightarrow \mathbf{f}(\mathbf{R}(\mathbf{s}_i)), r = \Pi^R[i]. \end{aligned} \quad (1)$$

The output frames $\{\mathbf{f}(\mathbf{L}(\mathbf{s}_i)), \mathbf{f}(\mathbf{R}(\mathbf{s}_i)) | i = 1, ..., N\}$ form the resultant stereoscopic video sequence. This procedure is illustrated in Figure 1.

Step 1 involves the structure and motion recovery which is a classical problem in computer vision. Several methods [28], [8], [29], [30] have been proposed to recover the camera extrinsic parameters given a video sequence. In our implementation, we adopt the method proposed in [30] to automatically extract the camera motion parameters and the 3D positions of sparse feature points for each frame.

Step 2 is the most challenging and difficult part. We adopt an optimization process to determine $(\mathbf{S}, \Pi^L, \Pi^R)$ by minimizing the cost function $E(\mathbf{S}, \Pi^L, \Pi^R)$. This cost function consists of the stereo cost, the similarity cost, and the continuity cost, corresponding to the 3 objectives mentioned above. Section IV describes them in details.

Finally, in Step 3, we can warp the pair of chosen base frames (from Step 2) to obtain the left and right frames. There are several possible ways to achieve this view warping. A classical way for view warping is to produce 3D meshes by triangulating the sparse point cloud, and render each mesh with texture map to synthesize the desired view. However, the 3D points recovered in the first step are too sparse and unevenly distributed. Missing geometry and outlying points can sometimes cause distracting artifacts. Another approach is planar-homography that restricts the warping on a plane (planar impostor). It computes a planar transformation (or homography) by minimizing the average warping/disparity error of the recovered sparse 3D feature points. However, in our application to generate stereo frames, apparent visual artifact will be resulted if the warping plane is allowed to be arbitrarily oriented. Figure 9(a) shows one such example. The building and streetlamps are not parallel to each other in the synthetic left and right views, as the warping planes for generating the left and right views are not parallel. Note that human vision is more sensitive to such misalignment than the disparity errors. To avoid the artifact, we restrict the warping planes to be perpendicular to the viewing direction and aligned to the up vector of the stereo-camera. In other words, all pixels in the warped frame have the same depth z_c. Due to the uneven distribution of the recovered sparse 3D points, we use $z_c = 2(z_{min}^{-1} + z_{max}^{-1})^{-1}$ instead of a mean value. Here, $[z_{min}, z_{max}]$ is the depth range of the scene with respect to viewpoint of the associated base camera, which can be estimated automatically with the recovered sparse 3D points. This restriction is also adopted in the plentopic sampling analysis [13]. Even with such crude *constant-depth* assumption, convincing stereo frames can be synthesized (Figure 9(b)).

IV. The Cost Function

The cost function $E(\mathbf{S}, \Pi^L, \Pi^R)$ consists of three terms, the stereo cost E_S, the similarity cost E_Q, and the continuity costs of camera orientation E_{CQ} and location E_{CV}. Mathematically, $E(\mathbf{S}, \Pi^L, \Pi^R)$ is defined as:

$$E(\mathbf{S}, \Pi^L, \Pi^R) = w_S E_S + w_Q E_Q + (w_{CQ} E_{CQ} + w_{CV} E_{CV}), \quad (2)$$

where w_S, w_Q, w_{CQ} and w_{CV} are weights of the cost terms.

A. Stereo Cost

1) Relative Parallax: The sense of stereo is due to the fact that our left and right eyes see differently. The same

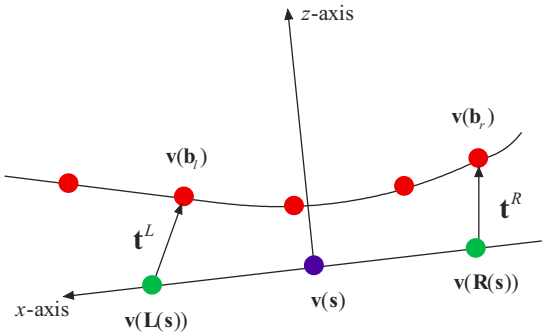


Fig. 3. The relationship between two base cameras and stereo-camera **s**. $\mathbf{v}(\mathbf{L}(\mathbf{s}))$ and $\mathbf{v}(\mathbf{R}(\mathbf{s}))$ are the left and right viewpoints of stereo-camera **s** respectively. $\mathbf{v}(\mathbf{s})$ is the center of stereo-camera **s**. $\mathbf{b}_l$ and $\mathbf{b}_r$ are the two base cameras, and their corresponding base frames will be warped to generate a pair of stereoscopic frames. $\mathbf{v}(\mathbf{b}_l)$ and $\mathbf{v}(\mathbf{b}_r)$ are their viewpoints respectively.

scene/object is spatially shifted in our left and right views. Such apparent position difference is called *binocular parallax*. In the 2D illustration of Figure 2(a), the viewing rays corresponding to the points i and j in the left view are overlaid onto the right view as indicated by the dotted red and blue lines. The displacements D_i and D_j are the parallaxes (binocular parallaxes). They are related to the interocular distance (d_{eye}), focal length (f), and depth (z_i). Obviously closer object results in larger parallax.

In this article, we argue that the sense of stereo relies *not* on the *absolute parallax*, but on the *relative parallax*. Relative parallax is the difference in parallax of two objects. The notion of relative parallax has long been used in the area of astronomy [31]. In this paper, the relative parallax is formally defined as follow. Consider Figure 2, a pixel $\mathbf{p}_i^L$ in the left view and its corresponding pixel $\mathbf{p}_i^R$ in the right view. The parallax of this pixel $\mathbf{p}_i$ is $\mathbf{D}_i = \mathbf{p}_i^L - \mathbf{p}_i^R$. The relative parallax with reference to another pixel $\mathbf{p}_j^L$ is defined as $\mathbf{m}_{ij} = \mathbf{D}_i - \mathbf{D}_j$. The parallax depends on their depths, focal length, and the interocular distance, $\mathbf{m}_{ij} = d_{eye} f(z_i^{-1} - z_j^{-1})$. Thus, for a pair of binocular images, we can define the *relative parallax matrix* $\mathbf{M}$ in which its element $\mathbf{m}_{ij}$ being the relative parallax of every pair of pixels $\mathbf{p}_i$ and $\mathbf{p}_j$.

Figure 2 explains why the relative parallax is more sensible than the absolute parallax in expressing the stereoscopic effect. The object in Figure 2(b) is moved closer to the viewer. The values of both the relative ($\mathbf{m}_{ij}$) and absolute (D_i, D_j) parallaxes are increased. In Figure 2(c), the object is not just moved closer but also flattened. Although the absolute parallax is increased, its relative parallax decreases.

To account for the relative parallax, we estimate the error in relative parallax between the synthetic (view-warped) and ideal stereo image pairs. Given the stereo-camera in the current iteration (it may change in the next iteration), the synthetic stereo frame is the one warped with the constant-depth assumption. It is the one that we can compute. The ideal stereo frame is the one that we can obtain if the true depth map is known. Obviously, the true depth map is *not* available. But we can still estimate the upper bound of this relative parallax error.

Each stereoscopic frame pair is synthesized by warping two chosen base frames. Let's denote the two base frames being considered for view warping in the current round of optimization as $\mathbf{f}_l$ (left candidate) and $\mathbf{f}_r$ (right candidate). If we have the true depth maps, we can correctly synthesize stereo pair $\mathbf{f}'_l$ and $\mathbf{f}'_r$ by a per-pixel warping. Let's denote the relative parallax matrix of this ideal stereo pair $(\mathbf{f}'_l, \mathbf{f}'_r)$ by $\mathbf{M}_G$. It is the ideal relative parallax matrix. Since the true depth map is not available, we can only warp the images with the constant-depth assumption. The relative parallax matrix of this synthetic stereo pair is denoted as $\mathbf{M}_W$. The matrix $\mathbf{M}_W - \mathbf{M}_G$ measures the error in relative parallax. Although we do not know $\mathbf{M}_G$, we can estimate a upper bound ε for the norm of the elements in $\mathbf{M}_W - \mathbf{M}_G$ (see Appendix for the derivation).

$$\varepsilon(\mathbf{s}, \mathbf{f}_l, \mathbf{f}_r) = f h_d \sqrt{(d_x + \frac{w}{2f} \cdot d_z)^2 + \mu (d_y + \frac{h}{2f} \cdot d_z)^2}, \quad (3)$$

where f is the focal length; $h_d = z_{min}^{-1} - z_{max}^{-1}$; w and h are the width and height of the base frames; μ is a constant greater than 1; and

$$d_x = |\, \mathbf{t}_x^L - \mathbf{t}_x^R \,|, \quad d_y = |\, \mathbf{t}_y^L - \mathbf{t}_y^R \,|, \quad d_z = |\, \mathbf{t}_z^L \,| + |\, \mathbf{t}_z^R \,|. \quad (4)$$

where $\mathbf{t}^L = \mathbf{v}(\mathbf{b}_l) - \mathbf{v}(\mathbf{L}(\mathbf{s}))$ and $\mathbf{t}^R = \mathbf{v}(\mathbf{b}_r) - \mathbf{v}(\mathbf{R}(\mathbf{s}))$ are the displacement vectors as illustrated in Figure 3. The intuition is that the deviation of the two displacement vectors, $\mathbf{t}^L$ (displacement between the candidate and ideal left viewpoints) and $\mathbf{t}^R$ (displacement between the candidate and ideal right viewpoints) should be close, especially in y axis.

Constant μ is the weight on y component. In our formulation, the x-axis is aligned with the line connecting the left and right viewpoints of the stereo-camera, the positive direction of z-axis is the viewing direction, and the positive direction of y-axis is the upward vector of the camera. The y component of relative parallax should be zero according to the stereovision theory, and any nonzero value will damage the stereoscopic effect. Therefore, we use $\mu(>1)$ to penalize any change in y direction caused by our view warping.

2) Warping Error: Besides the relative parallax error, the error due to warping should also be controlled to minimize visual artifact. We estimate the warping error as the maximum deviation between the pixel positions warped with constant-depth assumption and the ideal pixel positions if the true depths are known. If the deviation is too large, it will be easily aware by audiences. Note that minimizing the relative parallax error not necessarily minimizes the warping error. It is easy to demonstrate that the error due to warping the base frame pair $(\mathbf{f}_l, \mathbf{f}_r)$ is bounded by δ (see Appendix for the derivation),

$$\delta(\mathbf{s}, \mathbf{f}_l, \mathbf{f}_r) = \frac{\sqrt{2}}{2} f h_d \max(1, \frac{\sqrt{w^2 + h^2}}{2f}) \sqrt{\|\mathbf{t}^L\|^2 + \|\mathbf{t}^R\|^2}, \quad (5)$$

The goal of Equation 5 is to minimize the pixel position deviation via minimizing the displacement of viewpoints ($\mathbf{t}^L$ and $\mathbf{t}^R$). One assumption of Equation 5 is that the target and original views have the same viewing orientation. If the camera orientation of the target and original views are different, we can rectify the original views. The error due to the difference of camera orientation is accounted by the similarity cost (explained in Section IV-B).

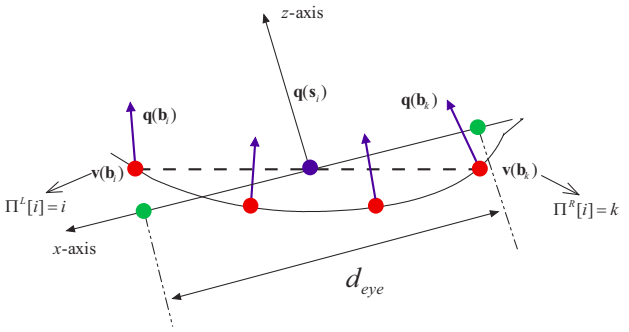


Fig. 4. Determination of the initial value of $(\mathbf{S}[i],\Pi^L[i],\Pi^R[i])$. $\mathbf{q}(\mathbf{s}_i)$, $\mathbf{q}(\mathbf{b}_l)$ and $\mathbf{q}(\mathbf{b}_k)$ are the orientations of camera $\mathbf{s}_i$, $\mathbf{b}_l$, and $\mathbf{b}_k$ respectively. The blue arrows indicate the viewing direction of these cameras.

Finally, we use both ε and δ to estimate the overall loss of stereoscopic effects due to the view warping. As the maximum relative parallax is $fh_d d_{eye}$, we use this maximum value to normalize ε and δ. Hence, the stereo cost of the entire stereoscopic sequence is defined as:

$$E_S(\mathbf{S},\Pi^L,\Pi^R) = \frac{1}{(fh_d d_{eye})^2}\sum_{i=1}^{N}(\varepsilon^2(\mathbf{S}[i],\Pi^L[i],\Pi^R[i]) + \delta^2(\mathbf{S}[i],\Pi^L[i],\Pi^R[i])). \quad (6)$$

B. Similarity Cost

The orientation of the two chosen base cameras $\mathbf{b}_l$ and $\mathbf{b}_r$ should be as close to that of the stereo-camera $\mathbf{s}$ as possible. This guarantees that the binocular views generated by viewing warping look similar to the original ones and they share the large common scene region. Therefore, we define the similarity cost for one stereo frame by:

$$\gamma(\mathbf{s},\mathbf{f}_l,\mathbf{f}_r) = \|\mathbf{q}(\mathbf{s})-\mathbf{q}(\mathbf{b}_l)\|^2 + \|\mathbf{q}(\mathbf{s})-\mathbf{q}(\mathbf{b}_r)\|^2, \quad (7)$$

where, $\mathbf{q}(\mathbf{s})$, $\mathbf{q}(\mathbf{b}_l)$ and $\mathbf{q}(\mathbf{b}_r)$ are the orientations of the stereo-camera $\mathbf{s}$, the left and right base cameras $\mathbf{b}_l$ and $\mathbf{b}_r$ respectively. Each is represented by a triplet of Euler angles. The similarity cost of the entire video sequence is defined by:

$$E_Q(\mathbf{S},\Pi^L,\Pi^R) = \sum_{i=1}^{N}\gamma(\mathbf{S}[i],\Pi^L[i],\Pi^R[i]). \quad (8)$$

C. Continuity Cost

The discontinuity of a video sequence is mainly caused by the unsteady rotational and translational speed of the camera. Therefore, to ensure the visual smoothness of the synthesized stereoscopic video, the rotational and translational acceleration should be minimized. Besides, since our stereoscopic video sequence is obtained by view warping, the change of the loss of stereoscopic effect should also be minimized to achieve visual smoothness. From Equations 5 and 3, the stereoscopic effect loss is dependent on the viewpoints of stereo-camera and the two candidate cameras. Thus, to ensure the stereo-camera moves steadily, the corresponding candidate cameras also have to move steadily. Hence, we define the continuity costs of the camera orientations, E_{CQ} and the location, E_{CV} as:

$$\begin{aligned}E_{CQ}(\mathbf{S},\Pi^L,\Pi^R) &= \sum_{i=2}^{N-1}\|2\mathbf{q}(\mathbf{s}_i)-\mathbf{q}(\mathbf{s}_{i+1})-\mathbf{q}(\mathbf{s}_{i-1})\|^2\\ E_{CV}(\mathbf{S},\Pi^L,\Pi^R) &= \frac{1}{d_{eye}^2}(\sum_{i=2}^{N-1}\|2\mathbf{v}(\mathbf{s}_i)-\mathbf{v}(\mathbf{s}_{i+1})-\mathbf{v}(\mathbf{s}_{i-1})\|^2\\ &+\sum_{i=2}^{N-1}\|2\mathbf{v}(\mathbf{b}_{\Pi^L[i]})-\mathbf{v}(\mathbf{b}_{\Pi^L[i+1]})-\mathbf{v}(\mathbf{b}_{\Pi^L[i-1]})\|^2\\ &+\sum_{i=2}^{N-1}\|2\mathbf{v}(\mathbf{b}_{\Pi^R[i]})-\mathbf{v}(\mathbf{b}_{\Pi^R[i+1]})-\mathbf{v}(\mathbf{b}_{\Pi^R[i-1]})\|^2).\end{aligned} \quad (9)$$

Here, we minimize the second derivative of the camera orientations and locations in order to reduce the discontinuity. It has been pointed out [32] that human are more sensitive to rotational vibrations, therefore E_{CQ} should be given larger weight. Generally, the weights of E_S and E_{CV} should be close to ensure the tradeoff between the warping errors and translational smoothness.

V. Optimization

Computing the optimal solution is challenging, as it involves both the combinatorial and continuous optimizations. We design an iterative algorithm to accomplish this task. Table II shows the pseudocode.

1) Find an initial solution of $\mathbf{S}$, Π^L, and Π^R.
2) Fix Π^L,Π^R, and find the optimal viewpoints of the stereo-cameras $\mathbf{V}=\{\mathbf{v}(\mathbf{s}_i)|\mathbf{s}_i=\mathbf{S}[i],i=1,...,N\}$, and viewing orientations $\mathbf{Q}=\{\mathbf{q}(\mathbf{s}_i)|\mathbf{s}_i=\mathbf{S}[i],i=1,...,N\}$ by minimizing E.
3) If E is small enough or doesn't improve from last iteration, terminate the iteration; otherwise, continue.
4) for $(i=1,...,N)$
   fix $\mathbf{v}(\mathbf{s}_i)$ and $\mathbf{q}(\mathbf{s}_i)$, and find the optimal $\Pi^L[i],\Pi^R[i]$ to minimize $w_S E_S + w_Q E_Q$.
5) Fix $\mathbf{Q}$, Π^L & Π^R, and refine $\mathbf{V}$ to minimize E.
6) Fix $\mathbf{Q}$ and $\mathbf{V}$, and refine Π^L and Π^R *locally* to minimize E.
7) Goto step 2.

TABLE II
ALGORITHM OF OPTIMIZATION.

Solving Π^L and Π^R involves a combinatorial optimization, which is too complicated to search globally for the best solution. However, if Π^L and Π^R are fixed, it becomes a nonlinear continuous optimization and can be optimized by Levenberg-Marquardt method (LM) efficiently. Therefore, we employ an optimization strategy which alternates between the continuous optimization and the discrete search. That is, instead of letting all parameters to change simultaneously, we temporarily fix discrete parameters to allow continuous optimization. Then we temporarily fix certain continuous parameters to allow discrete search. Such alternation continues in the next iteration.

We first initialize $\mathbf{S}$ (i.e. $\mathbf{V}$ and $\mathbf{Q}$), Π^L and Π^R (Section V-A). The initial $\mathbf{V},\Pi^L$ and Π^R are usually already close to optimal ones. Then in step 2, we fix Π^L and Π^R, and optimize the $\mathbf{V}$ and $\mathbf{Q}$ using standard continuous optimization method like Levenberg-Marquardt. If E is not sufficiently small, it means that Π^L and Π^R are not good enough and need to be adjusted in the following steps.

Steps 4-6 are mainly designed for adjusting the discrete parameters Π^L and Π^R. However, adjusting Π^L and Π^R is computationally expensive. In order to efficiently adjust Π^L and Π^R, we temporarily freeze E_{CQ} and E_{CV} (contain complex combinatorial optimization if Π^L and Π^R are not fixed) to their current values (step 4). Instead of optimizing the whole E, we only minimize for $w_S E_S + w_Q E_Q$. This is an implementation trick. Then in the following steps 5 and 6, we patch on this partial optimization. In step 5, we allow $\mathbf{V}$ to adjust in order to reflect the effect due to the previous change of Π^L and Π^R. This time we minimize for the whole E, (not just $w_S E_S + w_Q E_Q$). Once $\mathbf{V}$ adjusts, it affects Π^L and Π^R immediately. Finally in step 6, we *locally* adjust Π^L and Π^R to minimize for the whole E. With the partial optimization and the local adjustment, the adjustment on Π^L and Π^R becomes efficient.

A. Initialization

Firstly, we construct the initial selection. Let $\Pi^L[i] = i$ for $i = 1, ..., N$, *i.e.*, the base frame $\mathbf{f}_i$ will be the current candidate to be warped into the left view corresponding to the i-th left camera. Then, the remaining task is to search the proper base frame as the current candidate for the corresponding right view. Consider the i-th left camera, base camera $\mathbf{b}_k$ is the desired one if the distance between $\mathbf{b}_k$ and $\mathbf{b}_i$ is the closest one to the interocular distance d_{eye}. Its index is assigned to $\Pi^R[i]$, or $\Pi^R[i] = k$ such that $k > i$. It is natural to let the center and orientation of the i-th stereo-camera be the average of those of $\mathbf{b}_i$ and $\mathbf{b}_k$, i.e., $\mathbf{v}(\mathbf{s}_i) = (\mathbf{v}(\mathbf{b}_i) + \mathbf{v}(\mathbf{b}_k))/2$ and $\mathbf{q}(\mathbf{s}_i) = (\mathbf{q}(\mathbf{b}_i) + \mathbf{q}(\mathbf{b}_k))/2$. Next, according to the local coordinate system of the stereo-camera, if $\mathbf{b}_k$ is not on the right hand side of $\mathbf{b}_i$ when looking at the positive direction of the z-axis, the values in $\Pi^L[i]$ and $\Pi^R[i]$ are swapped. $\mathbf{v}(\mathbf{R}(\mathbf{s}_i))$ and $\mathbf{v}(\mathbf{L}(\mathbf{s}_i))$ are the left and right viewpoints of $\mathbf{s}_i$ and are equal to $\mathbf{v}(\mathbf{s}_i) \pm 0.5 d_{eye} \mathbf{e}_x$ respectively, where $\mathbf{e}_x$ is the x-axis direction vector. Figure 4 illustrates the initialization graphically.

B. Speed-up

During the adjustment of $\Pi^L[i]$ and $\Pi^R[i]$, the terms E_{CQ} and E_{CV} involve the complex combinatorial optimization in which its complexity grows exponentially with the number of frames. Therefore, we employ a practical trick. It firstly ignores the continuity cost in step 4. Then the continuity consideration is brought back in steps 5 and 6 for improving visual smoothness. In step 4, for each stereo frame i, its best candidate pair ($\Pi^L[i]$, $\Pi^R[i]$) is determined by fixing the stereo-camera $\mathbf{s}_i$ (both viewpoint and orientation) and minimizing the *part of objective function* $w_S E_S + w_Q E_Q$, i.e. $\sigma_i = w_S(\delta^2 + \varepsilon^2) + w_Q \gamma$. Energy terms E_{CQ} and E_{CV} are temporarily fixed and ignored. As ($\Pi^L[i]$, $\Pi^R[i]$) affects the center of stereo-camera $\mathbf{v}_i$, we then optimize $\mathbf{v}_i$ according to the selected pair using LM method in step 5.

The key is to efficiently select the best candidate pair in step 4. For stereo-camera $\mathbf{s}_i$, $\sigma_i = w_S(\delta^2 + \varepsilon^2) + w_Q \gamma$. From Equation 5, we know $\delta^2 = A(\|\mathbf{t}^L\|^2 + \|\mathbf{t}^R\|^2)$, where A is an invariant if w, h, h_d, and f are fixed. So, for either $\|\mathbf{t}^L\| > \sqrt{\sigma/(Aw_S)}$ or $\|\mathbf{t}^R\| > \sqrt{\sigma/(Aw_S)}$, $w_S \delta^2 > \sigma$ is true. Hence,

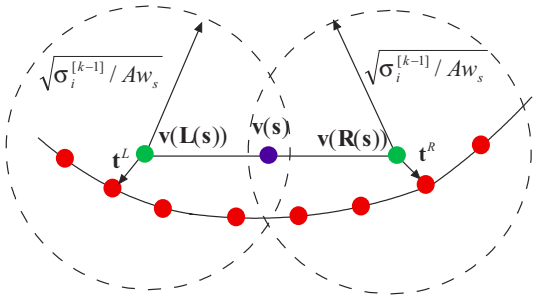


Fig. 5. The illustration of the determination of the appropriate base frames inside the spheres.

in the k-th iteration of the entire algorithm, we only select base camera pair candidates from those inside the spheres centered at left and right viewpoints with the radius equal to $\sqrt{\sigma_i^{[k-1]}/(Aw_S)}$, where $\sigma_i^{[k-1]}$ is the cost evaluated by using the values of $\Pi^L[i]$ and $\Pi^R[i]$ determined from the last iteration (or the $(k-1)$-th iteration), as shown in Figure 5. This scheme discards the inappropriate pairs, whose relevant cost $\sigma_i^{[k]}$ have:

$$\sigma_i^{[k]} \geq w_S(\delta^{[k]})^2 \geq \sigma_i^{[k-1]}, \tag{10}$$

where $\sigma_i^{[k]}$ is the current cost. Therefore, for each candidate pair inside the spheres, its cost $\sigma_i^{[k]}$ is calculated. The candidate pair, whose cost is the minimum and less than $\sigma_i^{[k-1]}$, is the desired one. Their indices are assigned to $\Pi^L[i]$ and $\Pi^R[i]$ accordingly. If there is no pair satisfying Equation 10, the current $\Pi^L[i]$ and $\Pi^R[i]$ are retained.

C. Optimization for Visual Smoothness

To maintain the visual smoothness, we control the accelerations of both left and right eyes. The accelerations are computed by the second-order difference of the eye positions. This smoothness is determined by E_{CQ} and E_{CV}. While E_{CQ} can be optimized easily by LM method in step 2, the optimization of E_S and E_{CV} are highly dependent on $\mathbf{V}$, Π^L and Π^R, and has a high combinatorial complexity. In step 6 of the pseudocode, when the viewpoints of stereo-cameras are fixed, E_{CV} merely relies on the sum of the norm of the acceleration of the left and right eyes. Due to the symmetry, we only explain the left eye in the following discussion.

The shaky candidate cameras are those whose accelerations exceed a tolerance a_{max}. Whenever shaky candidate exists, we should modify our choice of candidate frames. In general, such change of choice should involve the whole candidate index sequence. In practice, we only perform a local adjustment by modifying a candidate index *subsequence* centered at the detected shaky candidate. To simplify the discussion, we only describe the adjustment on the left-view frame index sequence Π^L. The right-view frame Π^R is adjusted similarly. Consider the n-element subsequence $\{\Pi^L[k_o], \Pi^L[k_o+1], ..., \Pi^L[k_o+n-1]\}$ where $k_o + \lceil n/2 \rceil$ is the detected shaky element, for every element $\Pi^L[k_o+i] = l_i$, its new value after adjustment can be any value in the range of $[l_i - m, l_i + m]$. In most of our experiments, m is 3 and n is 10. For each possible replacement, $w_S E_S + w_Q E_Q + w_{CV} E_{CV}$ is

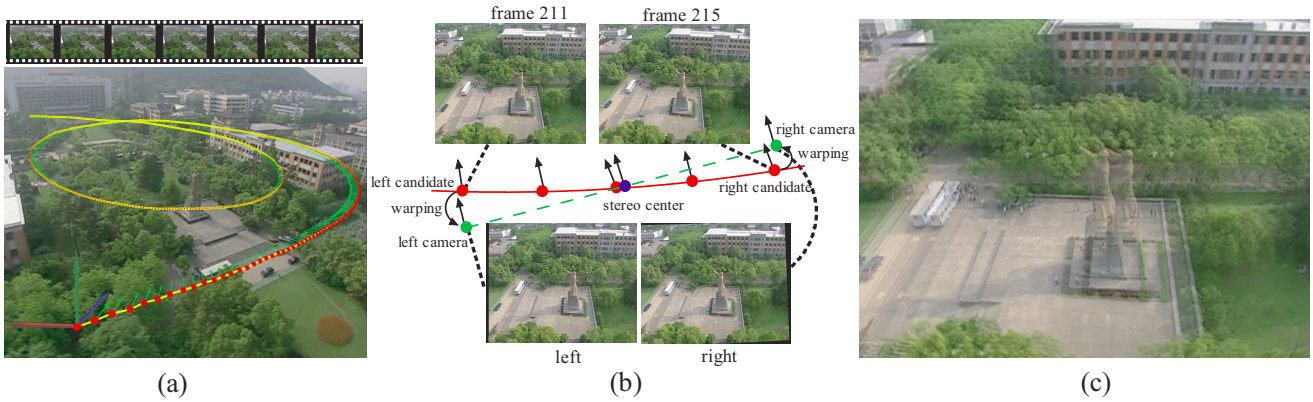


Fig. 6. An example of stereoscopic video generation. The input monocular video sequence is taken in the air. (a) shows the recovered base trajectory and a few frames from the base sequence. (b) illustrates the generation of a stereoscopic view pair. The blue dot coupled with 2 green dots indicate the virtual stereo-camera, where the green dots are the left and right cameras. (c) shows the composed stereo frame.

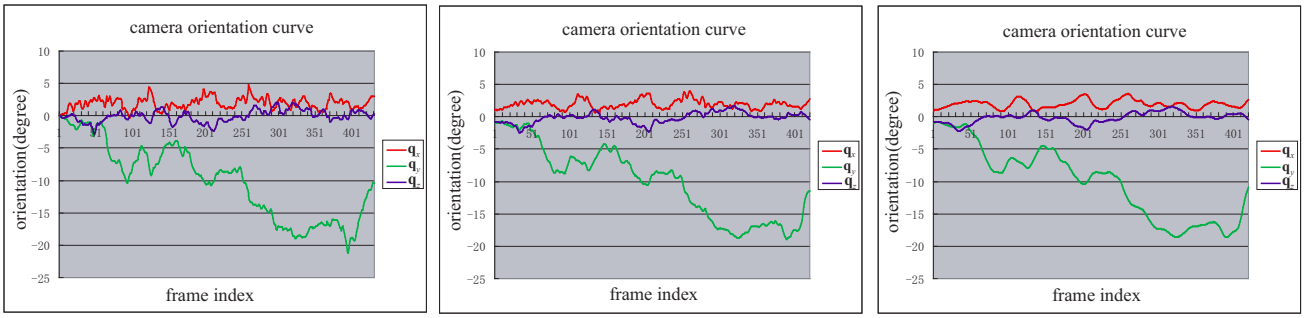


Fig. 7. Plots of Euler angles of the computed orientation. (a) is the recovered camera orientation of the base sequence (monocular video). (b) and (c) show the camera orientations of the corresponding stereo-camera computed with different w_{CQ} settings. In (b) $w_S = 1, w_Q = 100, w_{CQ} = 100, w_{CV} = 1$ while in (c) $w_S = 1, w_Q = 100, w_{CQ} = 10000, w_{CV} = 1$.

recomputed and the one with the minimal $w_S E_S + w_Q E_Q + w_{CV} E_{CV}$ is selected for replacement in order to improve smoothness. Then we return to step 2, the viewpoints and orientations of stereo-camera are further optimized according to the updated Π^L and Π^R by means of the LM method.

Since in each iteration the overall cost E is guaranteed to be decreased, the iteration converges at a minimal point. Although it may not be a global optimal solution, convincing solutions are obtained in all of our experiments.

VI. Results and Discussions

We have tested our method with several monocular video sequences from either movies or home-made video clips acquired via a hand-held video camera. All experiments are carried out on a PC with Intel Pentium IV 2.4 GHz CPU and 1 GB memory. Appealing results are obtained in our experiments. Figures 6 and 8 show two synthesized stereoscopic video sequences. The input monocular video sequence in Figure 6 is taken in the air. Video in Figure 8 is taken indoor. In Figure 8, we show the disparity of 5 sample pixels. Note that how our method correctly reflects the relative depth of scene objects.

The statistics of the four video sequences are listed in Table III. In the table, d_{eye} is interocular distance, and μ is the penalty factor for parallax in y direction (see Equation 3).

Sequence in	Fig. 1	Fig. 6	Fig. 8	Fig. 10
Number of frame	431	861	441	370
Time for camera tracking	26 min.	80 min.	30 min.	22 min.
Iteration number of optimization	2	1	2	3
Time for optimization	27 sec.	20 sec.	35 sec.	21 sec.
Time for view warping and video output	10 min.	20 min.	10 min.	8 min.
d_{eye}	10	4	12	10
μ penalty factor for y dir.	4	4	4	4
w_S	1	1	1	1
w_Q	100	100	100	100
w_{CQ}	10,000	100	100	10,000
w_{CV}	1	1	1	1

TABLE III
The Performance Statistics.

From the table, the optimization time is small. Camera tracking consumes most of the time. From our experiments, the number of iterations for the optimization is around 1 to 3. Such small number of iterations means that the initial solution is close to the optimal ones.

The weights in the cost function are user-specified. Table III lists their values. In our experiments, we set $w_S = w_{CV} = 1$, and $w_Q = 100$. The choice of w_{CQ} is highly dependent on the smoothness of the input video sequence. For the sequences

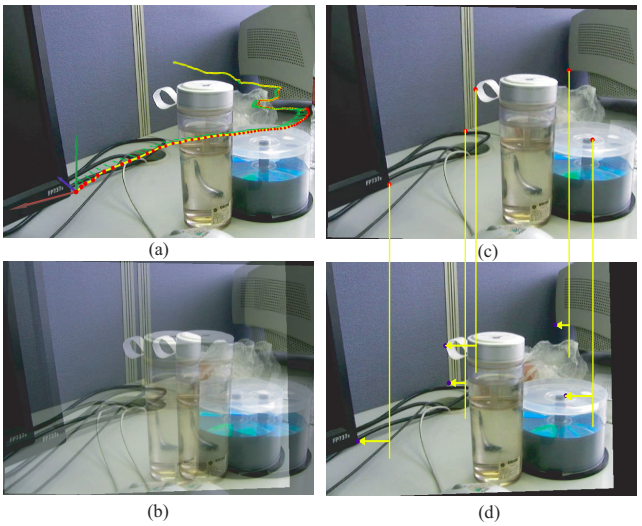


Fig. 8. Another example of stereoscopic video generation. (a) shows the recovered base trajectory. The two images in the (c) and (d) are the warping result of the base frames, and form a stereoscopic view pair in the result. The arrows in the images show the degree of binocular parallax of five points in the scene. It can be found that the remote points have the small parallax, whereas the near points have the large ones. (b) is the composition image of (c) and (d).

(a) view warping with planar-homography (b) view warping with constant-depth

Fig. 9. Comparison of planar-homography and constant-depth view warping. In (a), the building and the streetlamps in the left and right views are not parallel. Moreover, some farther objects even have much larger disparities than those nearer ones. Not similar objectionable artifact are found in the result from constant-depth warping (b).

(a) (b)

Fig. 10. A poor example of stereoscopic video generation. The input monocular sequence is taken by a hand-held camera moving in the direction of the camera viewing direction. (a) shows the recovered base trajectory. Since the angle between the moving direction and the viewing direction is very small, the binocular parallax is hard to be converted from the motion parallax. As the result, all binocular parallax of the sample points in the scene are almost identical, and the generated stereoscopic video does not properly show the depth cue.

extracted from professional movies (normally with smooth motion), w_{CQ} can be set to about 100. For the video captured by hand-held camera (like Figure 1), w_{CQ} should be greater than 100. Figure 7 shows the camera orientations (in Euler angles) of the base sequence and those of stereo sequence with different w_{CQ} settings. It shows that larger value of w_{CQ} leads to a smoother change of computed orientation, hence the result is less shaky. As the search window m of local adjustment for shaky camera (Section V-C) increases, the smoothness of result also increases, but with the trade-off of higher computational cost. In our experiments, we found $m = 3$ is a good choice to balance the trade-off between the performance and quality. In general, adjusting the weights trades among the smoothness, stereoscopic effect and/or visual similarity.

Recall that in Section III, we have justified why the simple but restrictive constant-depth view warping, instead of the more general planar-homography, is adopted. Figure 9(a) shows a stereo-frame from view warping with the planar-homography. Note that the building and streetlamps in the left and right synthetic views are not parallel. This artifact can be easily recognized by human vision. Even worse, some farther objects have much larger disparities than those closer objects. In contrast, the result from the constant-depth view warping (Figure 9(b)) does not cause similar objectionable artifacts.

Since no depth map is used, our approach has some limitations.

1) The scene should be static, otherwise the moving objects will be warped incorrectly. Because the left-eye and right-eye views are the warping results of the input frames at different time instances, warping them results in inconsistent object motion. Nevertheless, human vision may accept small inconsistent movements.
2) As our method relies on the motion parallax to synthesize the stereo parallax, it fails when there is no horizontal parallax in the input video. Examples include the case when the video is captured from a fixed viewpoint, the case when the viewing and motion directions coincide (Figure 10), and when the input video contains only vertical motion.
3) Our method tries to minimize the relative parallax error and warping error, and keep them consistent. However, since it is based on a crude *constant-depth* assumption, a large relative parallax/warping error may still occur and not be quite consistent in some cases. For example, when a originally panning camera suddenly changes its trajectory and moves forward, it is very difficult to keep all the parallaxes consistent. In this case, the objects whose depths are close to the optimal depth value (i.e. $z_c = 2(z_{min}^{-1} + z_{max}^{-1})^{-1}$) have more consistent parallaxes. On the other hand, the parallaxes of the objects whose depths are far away from the optimal depth value may be jittered. In practice, the regions with inconsistent parallaxes are usually not the visual focus and human vision have a higher tolerance.
4) If the focal length of input video varies, the output video may contain error. The simplest way to work around the problem is to preprocess the input video. A more sophisticated approach is to incorporate focal-length variation in the cost function. This is one of our

future directions.

VII. Conclusions

In this paper, we present a novel automatic synthesis of stereoscopic video sequence from the monocular one. Instead of recovering the depth map, we exploit the motion parallax. This allows us to avoid the objectionable visual artifact due to the inaccurately recovered 3D information. We formulate the video synthesis problem as an optimization problem. The introduced cost function considers the stereoscopic effects, the similarity, and the smoothness objectives. Users can adjust the weights to trade among these three objectives. Convincing results evidence the robustness and the efficiency of our approach. Despite of limitations, the proposed method is useful in many scenarios in which the video contains the panning motion.

Acknowledgments

We would like to thank all reviewers for their valuable suggestions to improve the paper. Thanks to Xiaohuang Huang and Yuanlong Shao for their help during video production. This work is supported by 973 Program of China (No. 2002CB312104) and NSF of China (No. 60633070), and partially affiliated with the CUHK Virtual Reality, Visualization and Imaging Research Centre as well as the Microsoft-CUHK Joint Laboratory for Human-Centric Computing and Interface Technologies.

Appendix

Assume the coordinate system is set to align with the left camera (right camera), and the camera views along z-axis. Refer to Table IV for the meaning of the notations used in this appendix. Then the position of the candidate camera is $\mathbf{t}=(\mathbf{t}_x,\mathbf{t}_y,\mathbf{t}_z)$. For simplicity, we assume the candidate camera and left (right) camera have the same orientation. If their orientations are different, we can rectify them beforehand. Without loss of generality, we choose a pixel p which 3D homogeneous coordinate is $(x,y,1,1/z)$ in the coordinate system of the candidate camera. From the candidate camera to the left (right) camera, its coordinate becomes $(\frac{xz+\mathbf{t}_x}{z+\mathbf{t}_z},\frac{yz+\mathbf{t}_y}{z+\mathbf{t}_z},1,1/(z+\mathbf{t}_z))$. Then the offset in the image is $d=\left(f\frac{\mathbf{t}_x-x\mathbf{t}_z}{z+\mathbf{t}_z},f\frac{\mathbf{t}_y-y\mathbf{t}_z}{z+\mathbf{t}_z}\right)^\top$. For convenience, we replace $z+\mathbf{t}_z$ with z by simply offset the coordinate, hence $d=\left(f\frac{\mathbf{t}_x-x\mathbf{t}_z}{z},f\frac{\mathbf{t}_y-y\mathbf{t}_z}{z}\right)^\top$. We assume the depths of scene are in the range of $[z_{min},z_{max}]$. During the view warping, we assume the depth is constant and equal to $z_c=2(z_{min}^{-1}+z_{max}^{-1})^{-1}$ over the whole image. This results in $d^W=\left(f\frac{\mathbf{t}_x-x\mathbf{t}_z}{z_c},f\frac{\mathbf{t}_y-y\mathbf{t}_z}{z_c}\right)^\top$. Here, we define Δd as the offset error due to the uncertainty of depth.

$$\begin{aligned}\Delta d &= \left(f(\mathbf{t}_x-x\mathbf{t}_z)(\tfrac{1}{z_c}-\tfrac{1}{z}),f(\mathbf{t}_y-y\mathbf{t}_z)(\tfrac{1}{z_c}-\tfrac{1}{z})\right)^\top\\ &\le \tfrac{fh_d}{2}(|\mathbf{t}_x-x\mathbf{t}_z|,|\mathbf{t}_y-y\mathbf{t}_z|)^\top\end{aligned}$$

$p_i(x_i,y_i,1,1/z_i)$	the homogeneous 3D coordinate of pixel i.
z_i	the depth of pixel i.
w, h	the width and height of the image.
f	the focal length.
$\mathbf{t}^L(\mathbf{t}^R)$	the relative translation between left (right) and the candidate cameras.
D_i	the parallax of pixel i.
D_i^W	the parallax of pixel i warping with constant depth z_c.
$d_i^L(d_i^R)$	the image offset of pixel i between left (right) and candidate cameras.
$d_i^{LW}(d_i^{RW})$	the image offset of pixel i between left (right) and candidate cameras warping with constant depth z_c.
e_{ij}	the relative parallax error between pixels i and j.

TABLE IV
Notations used in the Appendix.

Since $|x_i|\le\frac{w}{2f},|y_i|\le\frac{h}{2f}$, we have

$$\begin{aligned}||\Delta d|| &\le \tfrac{1}{2}fh_d\sqrt{(\mathbf{t}_x-x\mathbf{t}_z)^2+(\mathbf{t}_y-y\mathbf{t}_z)^2}\\ &\le \tfrac{1}{2}fh_d\sqrt{(|\mathbf{t}_x|+\tfrac{w}{2f}|\mathbf{t}_z|)^2+(|\mathbf{t}_y|+\tfrac{h}{2f}|\mathbf{t}_z|)^2}\\ &\le \tfrac{\sqrt{2}}{2}fh_d\max\left(1,\tfrac{\sqrt{w^2+h^2}}{2f}\right)||\mathbf{t}||\end{aligned}$$

Therefore, considering the parallax errors of both left and right cameras, we obtain Equation 5.

Next, we derive Equation 3. For any pixel $p_i(x_i,y_i,1,1/z_i)$ in the coordinate system of the candidate camera, having the following:

$$\begin{aligned}d_i^L &= \begin{pmatrix}\frac{\mathbf{t}_x^L-x_i^L\mathbf{t}_z^L}{z_i}f\\ \frac{\mathbf{t}_y^L-y_i^L\mathbf{t}_z^L}{z_i}f\end{pmatrix}, d_i^{LW}=\begin{pmatrix}\frac{\mathbf{t}_x^L-x_i^L\mathbf{t}_z^L}{z_c}f\\ \frac{\mathbf{t}_y^L-y_i^L\mathbf{t}_z^L}{z_c}f\end{pmatrix}\\ d_i^R &= \begin{pmatrix}\frac{\mathbf{t}_x^R-x_i^R\mathbf{t}_z^R}{z_i}f\\ \frac{\mathbf{t}_y^R-y_i^R\mathbf{t}_z^R}{z_i}f\end{pmatrix}, d_i^{RW}=\begin{pmatrix}\frac{\mathbf{t}_x^R-x_i^R\mathbf{t}_z^R}{z_c}f\\ \frac{\mathbf{t}_y^R-y_i^R\mathbf{t}_z^R}{z_c}f\end{pmatrix}\\ D_i^W &= D_i+(d_i^{LW}-d_i^{RW})-(d_i^L-d_i^R)\end{aligned}$$

Then the relative parallax error between pixels i and j:

$$\begin{aligned}e_{ij} &= (D_i^W-D_j^W)-(D_i-D_j)\\ &= (d_i^{LW}-d_i^{RW})-(d_i^L-d_i^R)-((d_j^{LW}-d_j^{RW})-(d_j^L-d_j^R))\\ &= f\begin{pmatrix}-(\mathbf{t}_x^L-\mathbf{t}_x^R)(\frac{1}{z_i}-\frac{1}{z_j})+P_x^L\mathbf{t}_z^L-P_x^R\mathbf{t}_z^R\\ -(\mathbf{t}_y^L-\mathbf{t}_y^R)(\frac{1}{z_i}-\frac{1}{z_j})+P_y^L\mathbf{t}_z^L-P_y^R\mathbf{t}_z^R\end{pmatrix}\\ &\le fh_d\begin{pmatrix}|\mathbf{t}_x^L-\mathbf{t}_x^R|+(|\mathbf{t}_z^L|+|\mathbf{t}_z^R|)\frac{w}{2f}\\ |\mathbf{t}_y^L-\mathbf{t}_y^R|+(|\mathbf{t}_z^L|+|\mathbf{t}_z^R|)\frac{h}{2f}\end{pmatrix}\end{aligned}$$

Here,

$$\begin{aligned}P_x^L &= x_i^L(\tfrac{1}{z_i}-\tfrac{1}{z_c})-x_j^L(\tfrac{1}{z_j}-\tfrac{1}{z_c}), P_x^R=x_i^R(\tfrac{1}{z_i}-\tfrac{1}{z_c})-x_j^R(\tfrac{1}{z_j}-\tfrac{1}{z_c})\\ P_y^L &= y_i^L(\tfrac{1}{z_i}-\tfrac{1}{z_c})-y_j^L(\tfrac{1}{z_j}-\tfrac{1}{z_c}), P_y^R=y_i^R(\tfrac{1}{z_i}-\tfrac{1}{z_c})-y_j^R(\tfrac{1}{z_j}-\tfrac{1}{z_c})\end{aligned}$$

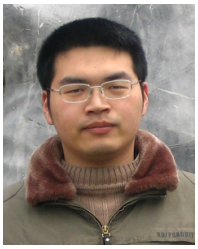
Hence, we obtain Equation 3.

References

[1] T. Okoshi, *Three-Dimensional Imaging Techniques*. Academic Press., 1976.

[2] W. Matusik and H. Pfister, "3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 814–824, 2004.

[3] E. H. Adelson and J. R. Bergen, "The plenoptic function and the elements of early vision," in *Computational Models of Visual Processing*, M. S. Landy and J. A. Movshon, Eds. MIT Press, 1991, ch. 1, pp. 3–20.
[4] D. V. Morland, "Computer-generated stereograms: a new dimension for the graphic arts," in *SIGGRAPH '76: Proceedings of the 3rd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1976, pp. 19–24.
[5] Y. Matsumoto, H. Terasaki, K. Sugimoto, and T. Arakawa, "Conversion system of monocular image sequence to stereo using motion parallax," in *Proc. of SPIE Stereo. Disp. and VR Sys*, vol. 3012, May 1997, pp. 108–115.
[6] P. Harman, "Home based 3D entertainment - an overview." in *IEEE Intl. Conf. on Image Processing*, 2000, pp. 1–4.
[7] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dence two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1/2/3, pp. 7–42, April – June 2002.
[8] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
[9] Y. Lu, J. Z. Zhang, Q. M. J. Wu, and Z.-N. Li, "A survey of motion-parallax-based 3D reconstruction algorithms," *IEEE Transaction on SMC-C*, vol. 34, no. 4, pp. 532–548, December 2004.
[10] H.-Y. Shum and S. B. Kang, "A review of image-based rendering techniques," in *Proc. of IEEE/SPIE Visual Communications and Image Processing (VCIP)*, 2000, pp. 2–13.
[11] C. Bregler, M. F. Cohen, P. Debevec, L. McMillan, F. X. Sillion, and R. Szeliski, "Image-based modeling, rendering, and lighting," in *SIGGRAPH 1999 Course #39*, 1999.
[12] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1996, pp. 31–42.
[13] J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum, "Plenoptic sampling," in *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000, pp. 307–318.
[14] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1996, pp. 43–54.
[15] W. R. Mark, L. McMillan, and G. Bishop, "Post-rendering 3D warping," in *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*. New York, NY, USA: ACM Press, 1997, pp. 7–ff.
[16] S. E. Chen and L. Williams, "View interpolation for image synthesis," in *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1993, pp. 279–288.
[17] S. M. Seitz and C. R. Dyer, "View morphing," in *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1996, pp. 21–30.
[18] Y. Horry, K. Anjyo, and K. Arai, "Tour into the picture: Using a spidery mesh interface to make animation from a single image," in *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, Los Angeles, 1997, pp. 225–232.
[19] J. Shade, S. Gortler, L. wei He, and R. Szeliski, "Layered depth images," in *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1998, pp. 231–242.
[20] H. S. Sawhney, Y. Guo, K. Hanna, R. Kumar, S. Adkins, and S. Zhou, "Hybrid stereo camera: an IBR approach for synthesis of very high resolution stereoscopic image sequences," in *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 2001, pp. 451–460.
[21] M. Magnor, M. Pollefeys, G. Cheung, W. Matusik, and C. Theobalt, "Video-based rendering," in *SIGGRAPH 2005 Course #16*, 2005.
[22] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.
[23] J. Carranza, C. Theobalt, M. A. Magnor, and H.-P. Seidel, "Free-viewpoint video of human actors," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 569–577, 2003.
[24] H. Huang and Y. Hung, "Panoramic stereo imaging system with automatic disparity warping and seaming," in *Proceedings of International Conference on Image Processing and Character Recognition (ICS'96)*, December 1996, pp. 48–55.
[25] S. Peleg, M. Ben-Ezra, and Y. Pritch, "Omnistereo: Panoramic stereo imaging," *IEEE Transaction on Pattern Ananlysis and Machine Intellegence*, vol. 23, no. 3, pp. 279–290, 2001.
[26] C. T. Loop and Z. Zhang, "Computing rectifying homographies for stereo vision." in *CVPR*, 1999, pp. 1125–1131.
[27] E. Rotem, K. Wolowelsky, and D. Pelz, "Automatic video to stereoscopic video conversion," A. J. Woods, M. T. Bolas, J. O. Merritt, and I. E. McDowall, Eds., vol. 5664, no. 1. SPIE, 2005, pp. 198–206. [Online]. Available: http://link.aip.org/link/?PSI/5664/198/1
[28] A. Fitzgibbon and A. Zisserman, "Automatic camera tracking," in *Video Registration*, M. Shah and R. Kumar, Eds. Kluwer, 2003, ch. 2, pp. 18–35.
[29] M. Pollefeys, L. J. V. Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch, "Visual modeling with a hand-held camera." *International Journal of Computer Vision*, vol. 59, no. 3, pp. 207–232, 2004.
[30] G. Zhang, X. Qin, X. An, W. Chen, and H. Bao, "As-consistent-as-possible compositing of virtual objects and video sequences." *Computer Animation and Virtual Worlds*, vol. 17, no. 3-4, pp. 305–314, 2006.
[31] T. W. Backhouse, "Absolute and relative parallax," *The Observatory*, vol. 11, pp. 343–343, Sept. 1888.
[32] Z. Duric and A. Rosenfeld, "Stabilization of image sequences," University of Maryland, Tech. Rep. CAR-TR-778, July 1995.



Guofeng Zhang received his BS degree in Computer Science from Zhejiang University, P.R.China, in 2003. Currently, he is a PhD candidate in computer science at State Key Laboratory of CAD&CG, Zhejiang University. His main research interests include camera tracking, 3D reconstruction, augmented reality and video enhancement.

Wei Hua received the BS degree in biomedical engineering from Zhejiang University in 1996, and the PhD degree in applied mathematics from Zhejiang University in 2002. Currently, he is an associate professor of the State Key Laboratory of CAD&CG of Zhejiang University. His research interests include real-time simulation and rendering, virtual reality and software engineering.

Xueying Qin received her PhD from Hiroshima University of Japan in 2001, and MS and BS from Zhejiang University and Peking University in 1991 and 1988, respectively. Currently, she is an associate professor of Zhejiang University. Her main research interests are augmented reality, video-based rendering, and photo-realistic rendering.

Tien-Tsin Wong received the B.Sci., M.Phil., and PhD. degrees in computer science from the Chinese University of Hong Kong in 1992, 1994, and 1998, respectively. Currently, he is a Professor in the Department of Computer Science & Engineering, Chinese University of Hong Kong. His main research interest is computer graphics, including image-based rendering, natural phenomena modeling, and multimedia data compression. He received *IEEE Transactions on Multimedia Prize Paper Award 2005* and *Young Researcher Award 2004*.

Hujun Bao received his Bachelor and PhD in applied mathematics from Zhejiang University in 1987 and 1993. He is currently the director of State Key Laboratory of CAD&CG of Zhejiang University. He is also the principal investigator of the virtual reality project sponsored by Ministry of Science and Technology of China. His research interests include realistic image synthesis, realtime rendering technique, digital geometry processing, field-based surface modeling, virtual reality and video processing.