Constructing 3D Natural Scene from Video Sequences with Vibrated Motions

Zhigang Zhu, Guangyou Xu, Xueyin Lin
Department of Computer Science and Technology, Tsinghua university, Beijing 100084, China
Email: zzg@vision.cs.tsinghua.edu.cn

Abstract

This paper presents a systematic approach to automatically construct the 3D natural scene from video sequences. The dense layered depth maps are derived from image sequences captured by a vibrated camera with only approximately known motion. The approach consists of (1) image stabilization by motion filtering and (2) depth estimation by spatio-temporal texture analysis. The two stage method not only generalized the so called panoramic image method and epipolar plane image method to handle the image sequence vibrations due to the un-controllable fluctuation of the camera, but also bypasses the feature extraction and matching problems encountered in stereo or visual motion. Our approach allows automatic modeling of the real environment for inclusion in the VR representation.

Keywords: *virtualized reality, layered representation, Image stabilization, panoramic view, epipolar plane image*

1. Introduction

The problem of virtual view generation of real scenes has received increasing attention in recent years. Existing work is this area can be divided into three classes: image-based method, model-based method and depth layering method. Image-based methods employ warping and morphing techniques to interpolate intermediate views from real images. In QuickTime VR[4], by capturing the 360-degree panoramic images of the scene from a fixed position, you can interactively adjust the view angles and zooming factors. Similarly in [7], image mosaics were constructed by registering and reducing the set of images into a single, larger resolution frame. Although this kind of representation needs fewer computation resources than that of model-based methods, the supported virtual views are limited only to a narrow range. Hirose[5] developed a camera system with GPS to produce a huge sequence of image data with viewpoint information. Then new images were synthesized from the finite data set using morphing technique. However this representation is data intensive and the selection and registration of reference points for morphing is a tedious work.

Model-based methods construct a full 3D model of objects by volumetric intersection method[8] or stereo method[6]. Then the model is reprojected to generate the desired views. The main difficulties of these approaches are the problems of camera setting, camera calibration and image registration to generate the full 3D models. Moezzi et al. [8] used 17 cameras while Kankade et al [6] used 51 cameras, all of which should be calibrated at first. In addition, these methods are not suitable for large-scale natural scene modeling.

In the third class, several depth layers are first estimated from image sequences and then are combined to generate a new view. Wang and Adelson[9] addressed the problem as the computation of 2D affine motion models and their support layers from an image sequence. The main problem of the approach is the usage of optical flow as the input of iterative motion clustering. It is also difficult to generate an arbitrary view with this approach. Chang and Zakhor[3] obtained depth information of pre-specified "reference frames" of an image sequence captured by an uncalibrated camera that scans a stationary scene, and transformed points on reference frames onto the image of the virtual views. However the reference frames were chosen rather arbitrarily and a desired view chosen far away from the reference frames lead to very erroneous results because the occluded or uncovered region could not be well represented. In addition, the matching of points in two frames proved to be problematic and there was noticeable decrease in performance as the baseline between two reference images increased.

In this paper, we address the problem of construct the 3D model of a static natural scene from an easy-obtained video sequence. Dense image sequences are captured by a vibrated camera with only approximately known translational motion. This corresponds to the interesting case of pointing a camera out the widow of a vehicle and driving down a street. Our approach falls into the third

class and overcomes most of the above-mentioned drawbacks. The two-stage method, image stabilization followed by panoramic epipolar plane image analysis, de-couples the (fluctuation) motion and the structure. In this way we generalize the panoramic view image approach[10] and epipolar plane image analysis[1] to more practical outdoor motion. The large Gaussian windowed Fourier spectrum method is proposed to detect the orientation of the motion texture, and the motion boundary is determined accurately by using global intensity similarity measurements along the detected orientations. Effective methods are presented for the occlusion recovery and depth interpolation. Our panoramic epipolar plane analysis algorithm is more effective since only the representative data are processed and the processing for each panoramic epipolar plane can work in parallel and without iterations. The scene is constructed as a set of panoramic depth layers, each of which consists of an intensity map and a depth map. Synthesized images of arbitrary views can be generated from the representation. Image segmentation, feature extraction and matching are avoided therefore the algorithm is fully adaptive and automatic.

2. Motion filtering and image stabilization

2.1. The motion model

Suppose the camera is mounted on a vehicle moving on an approximate flat road surface. In order to construct the 3D model of the roadside scene, the optical axis of the camera is perpendicular to the motion direction and its horizontal axis is along that direction. Other setting is possible but an image rectification procedure[11] should be applied at first. Within a long time period [0,T], the motion of the vehicle (camera) consists of the approximately known smooth planar motion and the unknown vibration due to the bump of the vehicle. In most cases, the smooth motion can be approximated as translational motion with constant velocity V. The small vibrations between two successive frames are modeled as three rotation angles $\Omega_x, \Omega_y, \Omega_z$ and three translation components T_x, T_y, T_z (Fig. 1). The relationship between the coordinates in time t and t+1 of a point is

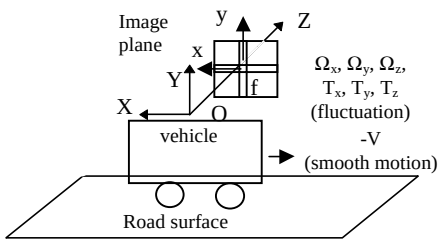


Fig. 1 Motion model

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} 1 & \Omega_y & -\Omega_z \\ -\Omega_y & 1 & \Omega_x \\ \Omega_z & -\Omega_x & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} T_x + V \\ T_y \\ T_z \end{pmatrix} \quad (1)$$

Under the pinhole camera model $(x, y) = \left(f\frac{X}{Z}, f\frac{Y}{Z} \right)$, the relation of image coordinates of the two successive frames is

$$s\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} -\frac{xy}{f} & \frac{x^2+f^2}{f} & -y \\ -\frac{y^2+f^2}{f} & \frac{xy}{f} & x \end{pmatrix} \begin{pmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{pmatrix} + \frac{1}{Z}\begin{pmatrix} f & 0 & -x \\ 0 & f & -y \end{pmatrix} \begin{pmatrix} T_x + V \\ T_y \\ T_z \end{pmatrix} \quad (2)$$

where s is the zooming factor. The relationship between image coordinate (x,y) and frame coordinates (u,v) is

$$(x, y) = (f s_x u,\ f s_y v) \quad (3)$$

where s_x and s_y are parameters of the camera's aspect ratio Given N pairs (u_i,v_i) and (u_i',v_i') at frame t and t+1, (i=1,...,N), we have 2N equations with 6+N unknown parameters:

$$\begin{cases} u_i' = u_i + a_i + b'u_i + cv_i + gu_i^2 + hu_i v_i \\ v_i' = v_i + d' + eu_i + b'v_i + gu_i v_i + hv_i^2 \end{cases} \quad (4)$$

where a′, b′ and c′ are approximated as constants for all the points in the case of very small (T_x,T_y,T_z) and

$$a_i = \frac{1}{s_x}\left(\Omega_y + \frac{T_x}{sZ_i}\right) + \frac{V}{s_x s Z_i} \overset{\Delta}{=} a' + \frac{V}{s_x s Z_i} \quad (5)$$

$$b' \overset{\Delta}{=} b_i = \frac{1}{s}\left(1 - \frac{T_z}{Z_i}\right) - 1, \quad d' \overset{\Delta}{=} d_i = \frac{1}{s_y}\left(-\Omega_x + \frac{T_y}{sZ_i}\right)$$

$$c = \frac{-s_y}{s_x s}\Omega_z, e = \frac{s_x}{s_y s}\Omega_z, g = \frac{s_x}{s}\Omega_y, h = -s_y\Omega_x$$

Equation (4) can be solved by giving more than 6 point pairs (N ≥ 6) in the successive frames using least square method [11].

2.2 Image stabilization by motion filtering

The image stabilization is to eliminate the vibration as if vehicle translated with constant velocity V within period [0,T]. The motion filtering is mainly to split a_i in equation (5) into two parts. By inserting t in equation (5), we can re-written it as

$$a_i(t) = a'(t) + \frac{V}{s_x s Z_i(t)} \quad (6)$$

Assume that $a_i(t)$, the equivalent translation component of each point along x axis in the image, is obtained for N(t) points (i=0,…,N(t)) at time t. The average of them is

$$\bar{a}(t) = a'(t) + a_s(t) \quad (7)$$

where

$$a_s(t) = \frac{V}{s_x s}\frac{1}{N(t)}\sum_{i=1}^{N(t)}\frac{1}{Z_i} \quad (8)$$

The accumulative average from time 0 to t is

$$A(t) = \int_0^t \bar{a}(\tau)d\tau = \int_0^t a'(\tau)d\tau + \int_0^t a_s(t)d\tau = A_F(t) + A_S(t) \quad (9)$$

where $A_S(t)$ is the accumulative smooth motion component and $A_F(t)$ is the accumulative fluctuation component.

In principle, $Z_i(t)$ remains unchanged under constant translation for any point i. It means that the locus of each point will become straight line after the effect of a′(t) and all other motion parameters is eliminated. Theoretically, if all the points are visible in the time period, then $a_s(t)$ remains unchanged in that time period. In this case $A_S(t)$ and $a_s(t)$ can be obtained by fitting a straight line to the set of points $\{(A(t), t), t=0,1,\ldots,T\}$. The slope of the straight line is $a_s(t)$. In this way the fluctuation component $A_F(t)$ can be obtained from equation (9).

During a long time period, some points will disappear while others will appear. In practice, we fit a smooth curve (or piecewise straight lines) to the point set and approximately separate the smooth motion and fluctuation component. The accumulations of all other motion components, $B_F(t)$, $C_F(t)$, $D_F(t)$,$E_F(t)$, $G_F(t)$ and $H_F(t)$, are calculated, for example,

$$B_F(t) = \int_0^t b'(\tau)d\tau, \quad C_F(t) = \int_0^t c(\tau)d\tau \quad (10)$$

and they are all assumed as fluctuation parameters. In this way any point (u_k, u_k) in any frame can be rectified to the stabilized location (u_k^p, v_k^p) by the pseudo-projective transformation

$$\begin{cases} u_k^p = u_k - \left(A_F + B_F u_k + C_F v_k + G_F u_k^2 + H_F u_k v_k\right) \\ v_k^p = u_k - \left(D_F + E_F u_k + F_F v_k + G_F u_k v_k + H_F v_k^2\right) \end{cases} \quad (11)$$

where (t) is omitted for the motion parameters.

2.3 Constructions of 2D ST images

The stabilized image sequence obeys the following ST perspective projection model

$$x(t) = f\frac{X + Vt}{Z}, y(t) = f\frac{Y}{Z} \quad (12)$$

where (X,Y,Z) represent the coordinates at time t = 0 and f is the equivalent focal length of the camera. Any feature point (x,y) forms a straight locus and the depth of the point (X,Y,Z) is

$$D = Z = f\frac{V}{v} \quad (13)$$

where $v = dx/dt$ is the slope of the straight locus. We can extract two kinds of interesting 2D ST images, one is the Panoramic View Image (PVI)[10], which grasps most of the 2D information of the roadside scene, the other is the Epipolar Plane Image (EPI)[1], whose oriented textures represent the depths of the points. The above processing is summarized as Algorithm 1.

[Algorithm 1] Image stabilization and 2D ST imaging

(1) Detecting the image velocities and their weights (beliefs) of representative points using a pyramid-based correlation method and solving equation (4) by a weighted least square method;

(2) Separating the fluctuation by motion filtering technique;

(3) Stabilizing the sequence by image rectification ;

(4) Constructing 2D PVI and EPIs.

The experiment results in Fig. 7 show that image stabilization play a vital role in the construction of good PVI and EPI when the fluctuation of the camera is severe otherwise the EPI approach is impossible. The results in Fig. 7 and Fig. 8 also show that the image stabilization method can reduce the fluctuation to a tolerant level for utility, though it is not absolutely eliminated.

3. Motion orientation and occlusion

3.1. Motion occlusion model

The 1st order motion texture model of the EPI can be expressed in the spatio-temporal domain as

$$g(x,t) = f(x - vt) \quad (14)$$

where $f(x)$ is the image of a single scan line at t=0. The model in the frequent domain is

$$G(\xi,\omega) = F(\xi)\delta(v\xi + \omega) \quad (15)$$

which states that the object points with the same depth and same constant translation only occupy a single straight line ($v\xi + \omega = 0$) passing through the origin in the frequency domain.

In the paper we model the 1st order motion occlusion in the xt image (EPI) as (Fig. 2)

$$g(x,t) = u(x - v_2 t) f_1(x - v_1 t) + (1 - u(x - v_2 t)) f_2(x - v_2 t) \quad (16)$$

where $v_1 < v_2$. The occluding mask $u(x - v_2 t)$ is the step function moving with velocity v_2. Its Fourier transform is

$$G(\xi,\omega) = \frac{1}{v_1 - v_2} F_1\left(\frac{v_2\xi + \omega}{v_2 - v_1}\right) U\left(\frac{v_1\xi + \omega}{v_1 - v_2}\right) + F_{u2}(\xi)\delta(v_2\xi + \omega) \quad (17)$$

where $F_{u2}(\xi) = F_2(\xi) - F_2(\xi) * U(\xi)$ is the Fourier transform of $f(x)(1-u(x))$, the visible parts of $f(x)$, and

$U(\xi)$ is the Fourier transform of *u(x)*. From equation (16) and (17) we can deduce that most of the energy spectrums lie in line $\xi = -\omega / v_1$ and line $\xi = -\omega / v_2$, which corresponding to the two layers respectively [11].

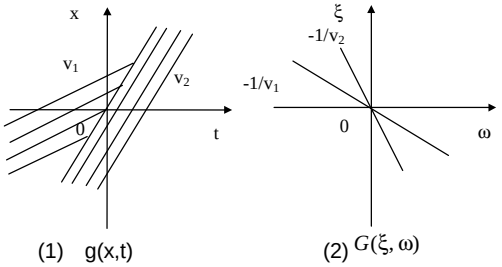


Fig.2. The motion occlusion model

3.2 Gaussian-Fourier orientation detector

We design a large window Gaussian-Fourier orientation detector (GFOD) in order to keep the precision for both the orientation of the texture and the localization of the motion boundary. If the spatio-temporal Gaussian window is defined as $w(x,t) = \exp(-\frac{x^2+t^2}{2\sigma^2})$, then the windowed 1st order motion texture can be represented as

$$g(x,t) = f(x - vt)w(x,t) \tag{18}$$

and its Fourier transform is

$$G(\xi,\omega) = c(v)G_w(\frac{\xi - v\omega}{v^2+1})W_t(v\xi + \omega) \tag{19}$$

where $c(v) = 2v/(v^2+1)$

$$F_w(\omega) \Leftrightarrow f_w(x) = f(x)e^{-\frac{x^2}{2(\sigma\sqrt{v^2+1})^2}},$$

$$W_t(\omega) = e^{-2(\sigma/\sqrt{v^2+1})^2\omega^2} \Leftrightarrow w_t(t) = e^{-\frac{t^2}{2(\sigma/\sqrt{v^2+1})^2}}$$

Again, most of the energy spectrums lie in the line $\xi = -\omega / v$. Similar result can be obtained in the case of multiple orientations and motion occlusions. The GFOD is listed as Algorithm 2 and an example is shown in Fig. 3.

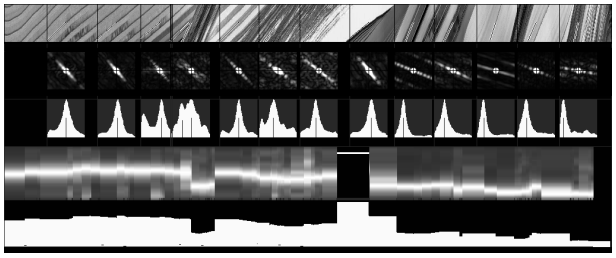


Fig. 3. Multiple orientation detection by GFOD (x-t image, energy spectrums of corresponding 64×64 blocks, orientation histogram, orientation energy distribution map and orientation angles)

[Algorithm 2]. Large window GFOD algorithm

(1) Selecting a suitable window size m×m (e.g. 64×64) given the resolution of the image. Let $\sigma^2 = \frac{m-1}{4}$.

(2) . Computing the Gaussian-Fourier transform $G_{t_i}(\xi,\omega)$ for the ST image centered at (x,t_i). The modified energy spectrum is calculated as

$$P(\xi,\omega) = \log\left(1 + G^2(\xi,\omega)\right) \tag{a2-1}$$

(3) Representing $G(\xi,\omega)$ in the polar coordinates (r,ϕ), i.e.

$$r = \sqrt{\xi^2+\omega^2}, \phi = \frac{\pi}{2} + \arctan\left(\frac{\xi}{\omega}\right) \tag{a2-2}$$

In this way we obtain the polar representation $P(r,\phi)$

(4) Calculating the orientation histogram

$$P_d(\phi) = \int_{r_1}^{r_2} P(r,\phi)dr, \quad \phi \in [0,\pi] \tag{a2-3}$$

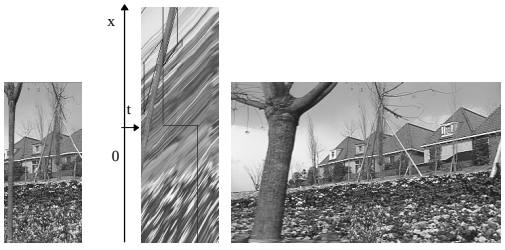
where ϕ corresponds to the orientation angle of the ST texture, and $[r_1,r_2]$ is the passband of the bandpass filter. The orientation energy distribution map $P_d(\phi,t)$ visually represents the depths of the points.

(5) Detecting multiple peaks $P_d(\theta_k)$ (k=0,1,…,K) in the orientation histogram. A motion boundary appears within the window if K > 1.

4. Panoramic epipolar plane analysis

Suppose that the image sequence has F frames of W×H images and the size of the Gaussian window is m×m. We construct a extended panoramic image(XPI) that composed of left halve of frame m/2, the PVI part formed by extracting the center vertical lines of from frame m/2 to frame F-m/2, and the right halve of the F-m/2 frame. XPI is more representative if the frame number is relatively small comparing to the size of each frame. Fig. 4 shows two frames of the typical FG sequence (W×H×F = 352×240×115). The panoramic image, epipolar plane image and extended panoramic image are shown in Fig. 5.

Fig. 4. Two frames of the Flower Garden (FG) sequence



(1) PVI (2) EPI (3) XPI

Fig. 5. Panorama and EPI of FG sequence

Panoramic epipolar plane method selectively processes the EPI parts around the PVI or XPI (e.g., the zigzag line in Fig. 5(2)). It consists of the following five steps.

Step 1. *Calculating the belief map of the orientation measurement.* The belief map corresponding to the panoramic view image $I_{PVI}(y,t)$ is

$$B(y,t) = G_t(y,t) - G_y(y,t) \quad (20)$$

where

$$G_t(y,t) = \frac{\partial I_{PVI}(y,t)}{\partial t}, G_y(y,t) = \frac{\partial I_{PVI}(y,t)}{\partial y} \quad (21)$$

The belief map means that the image motion in the textureless region can not be detected and there are aperture problems for the horizontal edges (along the motion direction). Depth estimations of the vertical edges are more robust.

Step 2. *Detection of multiple orientations.* For the epipolar plane image $I_{EPI}(x,t)$ corresponding to each y coordinate of the given PVI, orientations are only detected at the x coordinate (e.g. x =0) where the panorama has been taken from. The GFOD is applied to location (x,t_i) where $B(y,t_i)$ is great than a threshold (e.g., 2). Single or multiple orientation angles $\theta_k(k=1,\cdots,K)$ are determined (Algorithm 2). The image velocity can be calculated for each orientation as $v_k = \tan\theta_k$ .A motion boundary appears within the window if K>1 (typically K=2) and the processing goes to the next step.

Step 3. *Localization of the motion boundary*. In order that the algorithm is valid for most of the cases encountered in the natural scene, intensity similarities are measured within multiple-scale windows along the detected orientations $\theta_k(k=1,\cdots,K)$ and the one with greatest similarity is selected as the orientation at the detected point. In this way the depth/motion boundary can be accurately localized.

Suppose two orientations θ_1 and θ_2 ($\theta_1 > \theta_2$) are detected within the Gaussian window, where the former corresponds to nearer point and the latter corresponds to the farther one. The dissimilarity measurements along θ_k (k=1,2) for a given circular window of radius R are

$$\begin{cases} C_+(\theta_k, R) = \frac{1}{R}\sum_{r=1}^{R}\left[I(r,\theta_k)\right]^2 - \bar{I}_+^2(\theta_k, R) \\ C_-(\theta_k, R) = \frac{1}{R}\sum_{r=1}^{R} I^2(-r,\theta_k)^2 - \bar{I}_-^2(\theta_k, R) \end{cases} \quad (k=1,2) \quad (22)$$

where

$$r = \sqrt{(x-x_0)^2 + (t-t_0)^2}, \quad \bar{I}_\pm^2(\theta_k, R) = \frac{1}{R}\sum_{r=1}^{R} I(\pm r, \theta_k)$$

In the above equations, (x_0,t_0) is the center of the window , '+' and ' –' denote the measurements in positive and negative directions along the orientations respectively (refer to Fig. 2(1)). The cost functions for the near and far point are defined respectively as

$$E(\theta_1, R) = \frac{1}{2}(C_+(\theta_1, R) + C_-(\theta_1, R))/P_d(\theta_1) \quad (23)$$

and

$$E_\pm(\theta_2, R) = C_\pm(\theta_2, R)/P_d(\theta_2) \quad (24)$$

In equation (24) and the following equations, '-' is selected in case of occlusion (far to near) and '+' is selected in case of re-appearance (near to far).

In order to deal with cases of different kinds of object sizes, motion velocities and object occlusions, multiple scale cost functions, $E(\theta_1, R_i)$ and $E_\pm(\theta_2, R_i)$, i=1,2,3, are calculated within multiple scale window $R_1<R_2<R_3$, and multi-scale ratio is defined as

$$D_i = \frac{\max(E(\theta_1, R_i), E_\pm(\theta_2, R_i))}{\min(E(\theta_1, R_i), E_\pm(\theta_2, R_i))} \quad (25)$$

Scale p (p=1,2 or 3) with maximum D_P is selected for comparing the intensity similarities. The multi-scale method for motion boundary localization is summarized as Algorithm 3, and its performance can be clearly observed in Fig. 3 by comparing the orientation energy distribution map and orientation angles at the depth/motion boundaries.

Algorithm 3. Motion boundary localization

(1) Determining occlusion (-) or re-appearance (+) by simply judging if the new orientation angle is larger or smaller than the previous one when two orientations have been detected.
(2) Calculating multi-scale cost functions along θ_1 and θ_2 as $E(\theta_1, R_i)$ and $E_\pm(\theta_2, R_i)$, i=1,2,3.
(3) Selecting the most suitable window as R_p when $D_p = max\ D_i$;
(4) If $E(\theta_1, R_p) \le E_\pm(\theta_2, R_p)$ then select θ_1, else select θ_2

The selected orientation angle θ is refined by searching for the minimum dissimilarity measurement for a small angle range around θ. It is obvious that the accuracy of angles , especially that of the far object, can be improved by using more possible frames (Fig. 6).

Step 4. *Depth interpolation for the textureless region.* In order to obtain dense depth map, interpolations are applied to texture-less or weak-texture regions. The interpolation method is based on the fact that depth discontinuity almost always implies an occluding boundary or shading boundary. The angles between two instant t1 and t2 with estimated orientation angles θ_1 and θ_2 are linearly interpolated in case of smooth depth change ($|\theta_1 - \theta_2| < \mathrm{Tdis}$), and are assigned as $\theta(t) = \min(\theta_1, \theta_2)$ for depth discontinuity ($|\theta_1 - \theta_2| \ge \mathrm{Tdis}$), where T_{dis} is a predefined threshold.

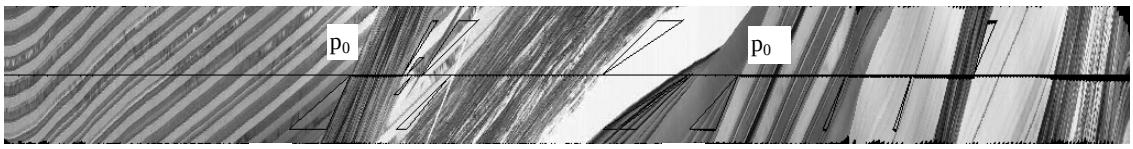


Fig. 6. Recovery of resolutions and occluded/ side regions

Step 5. *Recovery of resolutions and occluded/ side regions*. Because a PVI reserves the information only from a single viewing direction, some parts of the scene that are visible in some frames of the original sequence are lost. They can be recovered by analyzing the depth/motion occlusion. The algorithm includes three steps (Fig. 6).

Algorithm 4. Occlusion recovery

(1) Finding the location t_0 of the depth boundary, and the orientation angles (θ_1 and θ_2) of the occluded (far) and occluding (near) objects.

(2) Determining the spatio-temporal 1D segment denoted by x coordinate and the start/end frames (t_s/t_e). If the occlusion occurs then x <0; if the reappearance occurs then x>0. The start/end times and the magnitude of x depend on the two orientation angles and depth relations among nearby objects. In this way a triangular region $p_0p_sp_e$ has been determined.

(3) Verifying the types of the triangular regions. For simplicity, they are divided into two typical classes: the OCCLUDED one with same angle θ_1 and the SIDE with incremental angles from θ_1 to θ_2. These can be settled down by calculating and comparing the similarity measurements defined in (22) within the triangular region for the two cases.

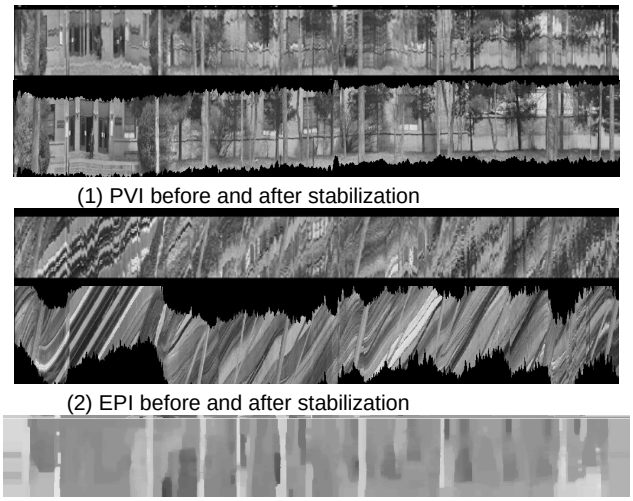
The original image resolution can be recovered according to the image velocity (v) of the point in the panorama. The thickness of the central black horizontal line in Fig. 6 indicates the number of points to be extracted in the x direction of the epipolar plane image.

5. The 3D scene construction

The 3D panoramic scene modeling is consists of four steps: (1) Image rectification and stabilization, (2) depth map acquisition, (3) fusion of depth and intensity map, (4) depth layer construction.

Step 1. *Image rectification and stabilization.* In order to compute the absolute depths of objects in the scene, calibration of the camera is needed. Fortunately the accurate intrinsic and extrinsic parameters of the camera are not a necessity for our purpose and our method. We can assume the optical axis passing through the center of the image and the approximate focal length *f* can be easily determined by a simple calibration procedure. We need not acquire the extrinsic parameters of the camera explicitly. What we need is to re-project the image as if the horizontal axis X of the camera is along the direction of the motion. This can be easily done by a pure image rotation transformation [11].

Now we have a "modified" camera whose motion is satisfied the motion model in Section 2.1. The image stabilization operation (Algorithm 1) is applied to the re-projected image sequence. Fig. 7 shows the panoramic view images and epipolar plane images before and after image stabilization for the 1024-frame TREE sequence. The sequence was captured when the camera was mounted on a hand-pushed vehicle and the actual velocity V was not measured. In real application V can be measured by other method in order to acquire absolute depth information. It can be seen from the stabilized PVI and EPI that the stabilization plays an important role in the construction of good PVI and EPIs when the fluctuations of the camera were severe, which included panning, tilting and rolling. Fig. 8 shows the stabilization of 1024-frame BUILDING sequence when the camera was mounted on a slowly moving car with tiny vibrations.



(1) PVI before and after stabilization

(2) EPI before and after stabilization

(3). Panoramic depth map of the TREE sequence

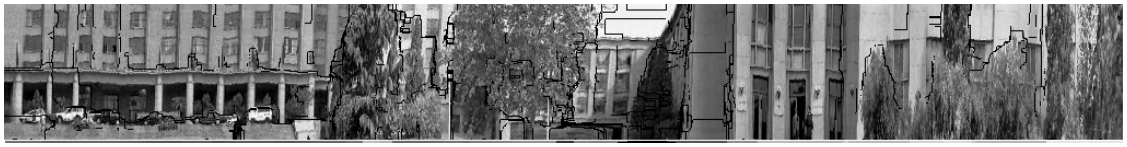
Fig.7. Results of the TREE sequence (128*128*1024)

Step 2. *Panoramic depth map acquisition.* The panoramic depth map corresponds to the PVI or the XPI. The depth map is acquired by the independent and parallel processing of H images of 2D panoramic epipolar planes. After the depth measurement belief map is calculated from the panorama, the depth information corresponding to each panoramic epipolar plane is obtained by executing step 2 to step 4 in Section 4. Fig. 9(2) shows the original panoramic depth map of the BUILDING sequence. The nearer depths are represented by brighter intensities.

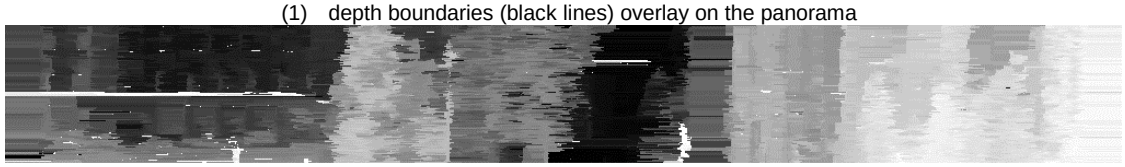
(1) PVI before stabilization

(2) PVI after stabilization

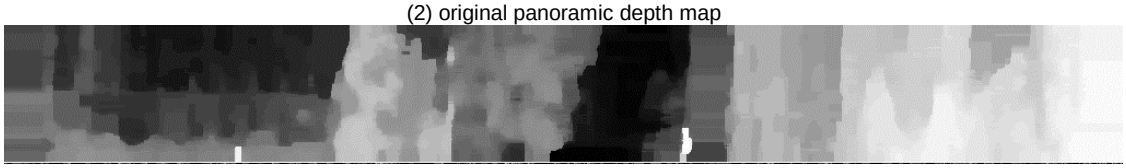
Fig. 8. Stabilization of the BUILDING sequence (128*128*1024)



(1) depth boundaries (black lines) overlay on the panorama



(2) original panoramic depth map



(3)panoramic depth map after depth-intensity fusion

Fig. 9. Panoramic depth map and the depth boundaries for the BUILDING sequence

Step 3. *Fusion of depth and intensity map.* It has been pointed out that depth information can not be completely recovered by using the motion cue only [2]. Notice that depth information is not available along the edge (white horizontal lines) of the platform in the left of the PVI in Fig. 9(2). The existing methods calculate the optical flows in the segmented images. The problem is that the accuracy of motion analysis depends on the performance of the image segmentation. In our method the fusion of motion and texture is carried out after the depth map is obtained. We use the fact that depth / motion boundary almost always takes place at the intensity / texture boundary.

Currently, a simple two-step algorithm was designed: (1) The median filter preserves the depth boundary while eliminates the errors due to the aperture problems and complex no-rigid motion of the trees, etc. (2).The intensity boundaries and the depth boundaries are detected along vertical directions in the PVI. If there exists no intensity boundary at a depth boundary, then we move the depth boundary to the place with a reasonable intensity boundary.

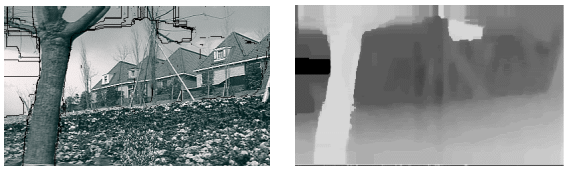
Fig. 9(3) shows the fusion result for the BUILDING sequence. The depth boundaries of the modified depth map were overlaid on the panoramic intensity image in Fig. 9(1). Most of the depth boundaries are accurately localized between trees and building. The depth changes of the walls, flags, and steps in the left of the panorama, and the moving objects before the building (two white spots at the bottom of the map) are detected.

Fig. 10 shows the results of the FG sequence. The tree distinctively stands out from the background and the depth of the grass land changes gradually. The panoramic depth map of the TREE sequence was shown in Fig. 7(3).

Step 4. *Depth layer construction.* Starting from the panoramic depth map, the resolutions of the near objects are enhanced and the occluded and side regions that are not visible in the panorama are recovered (Algorithm 4). Then we developed a compact representation called panoramic layered depth setting, which resembles scenery used in theatrical performance. In our representation, each layer consists of the panoramic intensity map and the panoramic depth map.

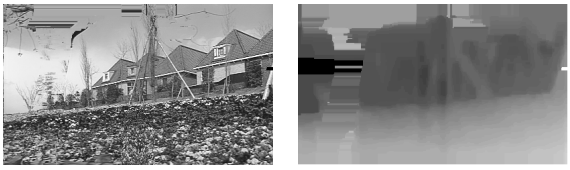
The panoramic layered depth setting is layered according to the occluding relations of the scene. The motivation of layering is to represent not only occluded

regions but also different spatial resolutions of objects with different depth ranges. In the spatio-temporal part (y-t image) of the extended panorama, different time scales are used for different depth layers of the scene. The panoramic layered depth setting is a compact representation for long image sequence, and is capable of synthesizing images of arbitrary views because it has both the intensity and the depth maps deriving almost all the information from the original image sequence. Solid models could be created by farther analysis of the layered depth maps.

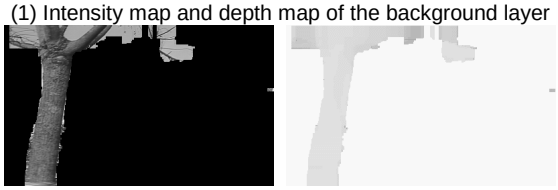


(1) depth boundaries (black lines) (2)panoramic depth map

Fig. 10. Panoramic depth map for the FG sequence



(1) Intensity map and depth map of the background layer



(2) Intensity map and depth map of the object layer

Fig. 11. Layered depth setting of the FG sequence

The background layer and the object layer are shown in Fig. 11. It should be noticed that the occluded regions in the extended panorama of Fig. 5(3) by the tree were recovered. Fig. 12 shows the preliminary results of the synthesized images of the entire scene from a new vantage point with and without the tree. A demonstration of virtual view generation is provided at Web site[12].

6. Conclusions

Structure from motion is still an open problem. In order to construct 3D natural scenes from video sequences, we make reasonable constraints to the motion of the camera. However the motion model is not an ideal one but a practical type which describes the motion of a ordinary mobile platform moving on the general roads. Systematic approach is proposed that gives a full solution from image sequence to 3D model. Image segmentation, feature extraction and matching are avoided therefore the algorithm is fully automatic. Synthesized images of arbitrary views can be generated from the model. The fusion of depth map and spatial structures (texture, edges) to obtain more accurate 3D models needs further study.

(1) with the tree

(2) without the tree

Fig. 12. Synthesized images of the entire scene of FG sequence

Acknowledgment

This work is funded by the China High Technology Program under contract No. 863-306-ZD-10-22. The authors thank the anonymous reviewers for their valuable comments and suggestions.

References

[1] Baker H H, Bolles R C, Generalization epipolar-plane image analysis on the spatio-temporal surface, Int. J. CV. 3. 1989:33-49

[2] Black M J, Jepson A D, Estimating optical flow in segmented images using variable-order parametric models with local deformations, IEEE Trans PAMI, 18(10), 1996: 972-986.

[3] Chang N L, Zakhor A, View generation for three-dimensional scene from video sequence, IEEE Trans Image Processing, 6(4), 1997: 584-598

[4] Chen S E, QuickTime VR - an image based approach to virtual environment navigation, Proc Siggraph 95, ACM Press, New York, 1995:29-38

[5] Hirose M, Image-based virtual world generation, IEEE Multimedia, January-March 1997:27-33

[6] Kanade T, Rander P and Narayanan P J, Virtualized reality: Constructing virtual worlds from real scenes, IEEE Multimedia, January-March 1997:34-47

[7] Mann S, Picard R W, Video orbit of the projective group: a new perspective on image mosaicing, Technical Report No.338, M.I.T. Media Lab Perceptual Computing Section,1995

[8] Moezzi S, Tai L-C, Gerard P, Virtual view generation for 3D digital video, IEEE Multimedia, January-March 1997: 27-33

[9] Wang J, Adelson E H, Representation moving images with layers, IEEE Trans. on Image Processing, 3(5),1994: 625-638.

[10] Zheng J Y, Tsuji S, Panoramic representation for route recognition by a mobile robot. Int J. CV, vol 9, no 1 1992: 55-76

[11] Zhu Z G, Environment modeling for .visual navigation, Ph.D. Dissertation , Tsinghua University, May 1997.

[12] Zhu Z G, Song L, The layered and multi-resolution panorama viewer, http:// vision.cs.tsinghua.edu.cn/~zzg/lamp.html.