The Eighth IEEE International Conference on Computer Vision, Vancouver, Canada, July 9-12, 2001

Parallel-Perspective Stereo Mosaics

Zhigang Zhu, Edward M. Riseman, Allen R. Hanson
Department of Computer Science, University of Massachusetts at Amherst, MA 01003
Email: {zhu, riseman, hanson}@cs.umass.edu

Abstract

In this paper we present a novel method for automatically and efficiently generating stereoscopic mosaics by seamless registration of optical data collected by a video camera mounted on an airborne platform that undergoes dominant translational motion. There are four critical points discussed in this paper: 1) Using a parallel-perspective representation, a pair of geometrically registered stereo mosaics can be constructed before we explicitly recover any 3D information under rather general motion. 2) A PRISM (parallel ray interpolation for stereo mosaicing) technique is proposed to make stereo mosaics seamless in the presence of motion parallax and for rather arbitrary scenes. A fast PRISM algorithm is presented and issues on stitching point selection and occlusion handling are discussed. 3) The epipolar geometry of parallel-perspective stereo mosaics generated under constrained 6 DOF motion is formulated, which shows optimal baselines, easy search for correspondence and constant depth resolution. 4) The proposed methods for the generation of stereo mosaics and then the reconstruction of a 3D map are efficient in both computation and storage. Experimental results on long video sequences are given.

1. Introduction

Recently, there have been attempts in a variety of applications to add 3D information into an image-based mosaic representation. Creating stereo mosaics from two rotating cameras was proposed by Huang & Hung [1]. More practical is the generation of stereo mosaics from a single off-center rotating camera by Peleg & Ben-Ezra [2] and Shum & Szeliski [3]. In fact, the idea of generating stereo panoramas for either an off-center rotating camera or a translating camera can be traced back to the earlier work in robot vision applications by Ishiguro, et al [4] and Zheng & Tsuji [5]. The attraction of the recent studies on off-center rotating cameras lies in how to make stereo mosaics with nice epipolar geometry and high image qualities and how to use them in image-based rendering. However, in stereo mosaics with a rotating camera, the viewpoints -- therefore the parallax -- are limited to images taken from a very small area, and the viewers are constrained to rotationally viewing the stereo representations. Translational motion, on the other hand, is the typical prevalent sensor motion during ground vehicle navigation [5] or aerial surveys[6]. In [2] the authors mentioned that the same techniques developed for a rotating camera could be applied to a translating camera, but it turns out that there has been little serious work on this topic. A rotating camera can be easily controlled to achieve the desired motion. On the contrary, the translation of a camera over a large distance is much hard to control. How to generate stereo mosaics under a rather general motion with dominant translation is still an unsolved problem and is the focus of this paper.

1.1. Related work

In this paper, we will address the problem of creating seamless and geometrically-registered 3D mosaics from a moving camera, undertaking a rather general motion and allowing viewpoints change over a large scale. Obviously use of standard 2D mosaicing techniques based on 2D image transformations such as a manifold projection [7] cannot generate a seamless mosaic in the presence of large motion parallax, particularly in the case of surfaces that are highly irregular or with large differences in heights. Many researches on seamless mosaics deal with video from a rotating camera. As a typical example, in generating seamless 2D mosaics from a hand-held camera, Shum & Szeliski [8] used a local alignment (de-ghosting) technique to compensate for the small amount of motion parallax introduced by small translations of the camera. Rousso, et al [9] suggested that a 2D orthogonal projection could be generated by taking a collection of strips, each with a width of one pixel, from interpolated camera views in between the original camera positions, but details were not provided. Kumar, et al [10] dealt with the geo-registration problem by utilizing an available geo-referenced aerial image with broader coverage, as well as an accompanying co-registered digital elevation map. In more general cases for generating image mosaics with parallax, several techniques have been proposed to explicitly estimate the camera motion and residual parallax by recovering a projective depth value for each pixel [11-13]. These approaches could produce geo-referenced mosaics; however, they are computationally intense, and since a final mosaic is represented in a reference perspective view, there could be serious occlusion problems due to large viewpoint differences between a single reference view and the rest of the views in the image sequence.

An important part of the work that follows is a new mosaic representation that can support seamless mosaicing under a rather general motion and also can capture inherent 3D information during the mosaic process. A *parallel-perspective* model is selected for representing mosaics in our approach since it is the closest form to the original perspective video sequence under large motion parallax, yet its geometry allows us to generate seamless stereo mosaics. To accomplish this, we propose a novel technique called PRISM (parallel ray interpolation for stereo mosaicing) to efficiently convert the sequence of *perspective* images with dramatically changing viewpoints into the parallel-perspective stereo mosaics.

2. Generalized Parallel-Perspective Stereo

The basic idea of parallel-perspective stereo mosaics under 1D translation has been proposed by Zhu et al [6], and Chai & Shum [14], showing the advantages of depth recovery from parallel projection in both epipolar geometry and depth resolution. Assume the motion of a camera is an ideal 1D translation, the optical axis is perpendicular to the motion, and the frames are dense enough. Then, we can generate two spatio-temporal images by extracting two columns[1] of pixels (perpendicular to the motion) at the front and rear edges of each frame in motion. The mosaic images thus generated are *parallel-perspective*, which have perspective projection in the direction perpendicular to the motion and parallel projection in the motion direction. In addition, these mosaics are obtained from two different oblique viewing angles of a single camera's field of view, so that a stereo pair of left and right mosaics captures the inherent 3D information.

In this paper, the stereo mosaicing mechanism is generalized to the case of 3D translation, assuming that the 3D curved motion track has a dominant translational motion (e.g. the *Y* direction in Fig. 1) so that a parallel projection can be generated in that direction. Under 3D translation, parallel stereo mosaics can be generated in the same way as in the case of 1D translation. The only difference is that viewpoints of the mosaics form a 3D curve instead of a 1D straight line. Without loss of generality, we assume that two vertical 1-column slit windows have $d_y/2$ offsets to the left and right of the center of the image respectively (Fig. 1). We define the "scaled" vector of a camera position **T**$= (T_x, T_y, T_z)$ (related to a common reference frame – the first frame in Fig. 1) as **t**$= (t_x, t_y, t_z) = F$ **T** $/ H$ in the mosaicing coordinates, where F is the focal length of the camera, and H is the height of a *fixation plane* (e.g., average height of the terrain) where one pixel in the y direction of the mosaics corresponds to H/F word distances in the plane. From the frame in the camera position (t_x, t_y), the front slit will be translated to $(t_x, t_y+d_y/2)$ in the "left eye" mosaic, while the rear slit will be paced $(t_x, t_y-d_y/2)$ in the "right eye" mosaic. This treatment reduces the distortion of the mosaics in the X direction. Here we assume that the translation in the Z direction is very small compared to the height H so that scale changes of the same regions in the stereo mosaics are small.

Suppose the corresponding pair of the 2D points (one from each mosaic), (x_l,y_l) and (x_r,y_r), of a 3D point (X,Y,Z), is generated from original frames in the camera positions (T_{xl},T_{yl},T_{zl}) and (T_{xr},T_{yr},T_{zr}) respectively. The *parallel-perspective projection model* of the stereo mosaics thus generated can be represented by the following equations

$$
\begin{aligned}
(x_l, y_l) &= \left(F\frac{X-T_{xl}}{Z-T_{zl}} + F\frac{T_{xl}}{H},\ F\frac{Y}{H} - (\frac{Z-T_{zl}}{H} - 1)\frac{d_y}{2} \right) \\
(x_r, y_r) &= \left(F\frac{X-T_{xr}}{Z-T_{zr}} + F\frac{T_{xr}}{H},\ F\frac{Y}{H} + (\frac{Z-T_{zr}}{H} - 1)\frac{d_y}{2} \right)
\end{aligned}
\qquad (1)
$$

Eq. (1) serves a function similar to the classical pin-hole perspective camera model. Note that T_{yl} and T_{yr} are not included in Eq. (1) thanks to the parallel projection in the Y direction. Stereo mosaics are generated only with the knowledge of the camera positions.

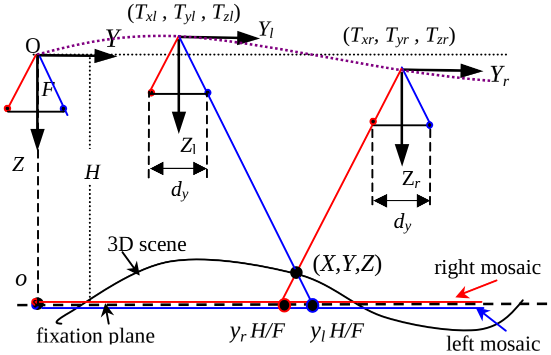

Fig. 1. Parallel-perspective stereo geometry. The X axis and the slit windows are perpendicular to the plane of the figure. Both mosaics are built on the fixation plane, but their unit is in pixel – each pixel represents H/F world distances.

Because of the way the stereo mosaics are generated, the viewpoints of both are on the same smooth 3D motion track. The camera position $\mathbf{t_r}$ of column y in the right mosaic is exactly the camera position $\mathbf{t_l}$ of column y+d_y in the left mosaic (see Fig. 1), i.e. $\mathbf{t_r}(y) = \mathbf{t_l}(y + d_y)$ are both only functions of the y coordinate. Let us denote the "mosaic displacement" vector in the stereo mosaics is $(\Delta x, \Delta y) = (x_r - x_l, y_r - y_l)$. In the general case of 3D translation, the depth of the point can be calculated from the stereo mosaics as

$$
Z = H(1 + \frac{\Delta y}{d_y}) + \frac{T_{zl} + T_{zr}}{2} \qquad (2)
$$

which implies that the depth (subtracting camera height changes) is proportional to the mosaic displacement. The corresponding point in the right mosaic of any point in the left mosaic will be on an *epipolar curve* determined by the left point and the 3D motion track, i.e.

$$
\Delta x = \frac{b_x \Delta y + b_z d_y (x_l - \frac{t_{xr} + t_{xl}}{2}) / F}{\Delta y + d_y + b_z d_y / (2F)} \qquad (3)
$$

where $b_x(y_l, \Delta y) = [t_{xl}(y_l + d_y + \Delta y) - t_{xl}(y_l)]$ and $b_z(y_l, \Delta y) = [t_{zl}(y_l + d_y + \Delta y) - t_{zl}(y_l)]$ are "baseline" functions in the x and z directions of variables y_l and Δy. Hence Δx is a nonlinear function of position (x_l, y_l) as well as displacement Δy, which is quite different from the epipolar geometry of two-view perspective stereo. The reason is that image columns with different y_l coordinates in parallel-perspective mosaics are projected from different viewpoints.

Since a fixed angle between the two viewing rays is selected for generating the stereo mosaics, the "disparities" (d_y) of all points are fixed; instead a geometry of optimal/adaptive

[1] We assume that the scanlines of the camera are in the motion direction.

baselines ($b_y = d_y + \Delta y$) for all the points is created. In other words, for any point in the left mosaic, searching for the match point in the right mosaic means finding an original frame in which this match pair has a pre-defined disparity (by the distance of the two slit windows) and hence has an adaptive baseline depending on the depth of the point.

If the motion of the camera is constrained to a 2D translation in the *XY* plane (i.e. $T_z=0$), the depth of the point can be simply derived as

$$Z = H(1+\frac{\Delta y}{d_y}) \tag{4}$$

The stereo mosaic displacement Δy is a function of the depth variation of the scene around the fixation plane H (which is almost true in the case of 3D translation). It is interesting to note that since the selection of the two mosaic coordinate systems brings a constant shift d_y to the scaled "baseline", it produces the fixation of the stereo mosaics to a horizontal "*fixation plane*" of an average height *H*. This is highly desirable for both stereo matching and stereoscopic viewing. The epipolar curve under 2D translation becomes

$$\Delta x = b_x(y_l, \Delta y)\frac{\Delta y}{\Delta y + d_y} \tag{5}$$

which is only a function of position y_l and Δy, but is independent of the coordinate *x*. We have three conclusions for the epipolar geometry of the parallel-perspective stereo:

1) *In the general case of 3D translation*, if we know the range of depth variation plus camera height changes, $\pm\Delta Z_m$, the search region for the corresponding point in the right mosaic is $\Delta y \in [-\frac{d_y}{H}\Delta Z_m, +\frac{d_y}{H}\Delta Z_m]$ (from Eq. (2)), and along an epipolar curve, which is different for every point (x_l, y_l) in general (Eq. (3)).

2) *In the case of 2D translation*, the epipolar curve for a given point (x_l,y_l) in the left mosaic passes through the location (x_l,y_l) in the right mosaic (Eq. (4)), which implies that the stereo mosaics are aligned for all the points whose depths are *H*. The same epipolar curve function (of y_l and Δy) is applied to all the points in the left mosaic with the same y_l coordinate.

3). *In the ideal case* where the viewpoints of stereo mosaics lie in a 1D straight line, the epipolar curves will turn out to be horizontal lines. Therefore we can apply most of the existing stereo match algorithms for rectified perspective stereo with little modification.

3. Mosaicing under 6 DOF Motion

This section discusses how to generate stereo mosaics under a more general motion (6 DOF). To generate meaningful and seamless stereo mosaics, we need to impose some constraints on the values and rates of changes of motion parameters of a camera (Fig. 2a). First, the motion must have a dominant direction. Second, the angular orientation of the camera is constrained to a range that precludes it turning more than 180°. Third, the rate of change of the angular orientation parameters must be slow enough to allow overlap of successive images. These constraints are all reasonable and are satisfied by a sensor mounted in a light aircraft with normal turbulence. Within these constraints, the camera can undergo six DOF motion. There are two steps necessary to generate a rectified image sequence that exhibits only 3D translation, and from which we can subsequently generate seamless mosaics:

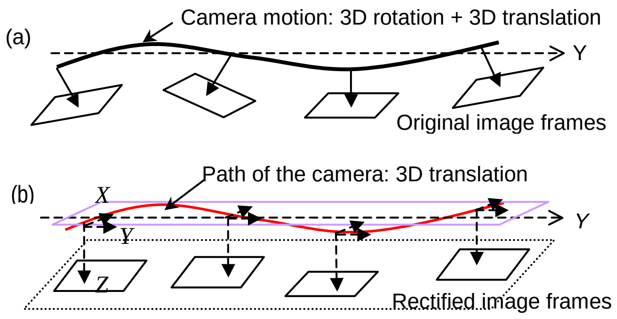
Camera motion: 3D rotation + 3D translation
(a)
Y
Original image frames
Path of the camera: 3D translation
(b)
X
Y
Y
Z
Rectified image frames

Fig. 2. Image rectification. (a) Original and (b) rectified image sequence.

Step 1. Camera orientation estimation. Using an internally pre-calibrated camera, the extrinsic camera parameters (camera orientations) can be determined from an aerial instrumentation system (GPS, INS and a laser profiler) [15] and bundle adjustment techniques [16]. The detail is out the scope of this paper, but the main point here is that we do not need to carry out dense match between two successive frames. Instead only sparse tie points widely distributed in the two images are needed to estimate the camera orientations. While the full calibration of parameters in all camera positions is a very difficult task in video, we have given a practical treatment [17] where nice stereo mosaics can be obtained without calibration. The results will be discussed in Section 7.

Step 2. Image rectification. An image rotation transformation is applied to each frame in order to eliminate the rotational components(Fig. 2b). In fact we only need to do this kind of transformation on two narrow slices in each frame that will contribute incrementally to each of the stereo mosaics. In our motion model, the 3D rotation is represented by a rotation matrix **R**, and the 3D translation is denoted by a vector $\boldsymbol{T} = (T_x,T_y,T_z)^t$. A 3D point $\mathbf{X_k} = (X_k,Y_k,Z_k)^T$ with image coordinates $\mathbf{u}_k = (u_k, v_k, 1)^t$ at current frame *k* can be related to its reference coordinates $\mathbf{X} = (X, Y, Z)^T$ by the following equation

$$\mathbf{X} = \mathbf{R}_k\mathbf{X}_k + \mathbf{T}_k \tag{6}$$

In the image rectification stage, a projective transformation $\mathbf{A}_k$ is applied to frame *k* of the video using the motion parameters obtained from the camera orientation estimation step:

$$\mathbf{u}_k^{\mathbf{p}} \cong \mathbf{A}_k\mathbf{u}_k \quad \mathbf{A}_k = \mathbf{F}\mathbf{R}_k\mathbf{F}^{-1}, \quad \mathbf{F} = \begin{pmatrix} F & 0 & 0 \\ 0 & F & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{7}$$

where $\mathbf{u}_k^{\mathbf{p}}$ is the reprojected image point of frame k, and F is the camera's focal length. The resulting video sequence will be a rectified image sequence as if it was captured by a "virtual" camera undergoing 3D translation (T_x,T_y,T_z). We assume that vehicle's motion is primarily along the Y axis after eliminating the rotation, so we will have T_x<<T_y , T_z<<T_y. This implies that the mosaic will be produced along the Y direction. If the translational component in the Z direction is much smaller than the distance itself, we use a scaling factor in the rectification for each frame to compensate for the Z translation so that the rectified sequence will only exhibit 2D translation [17]. Beside the nicer epipolar geometry, the corresponding image patches in stereo mosaics will have similar scales so that direct methods for stereo matching can be used.

4. PRISM: Mosaicing with Motion Parallax

Due to large and possibly varying displacements between each pair of successive frames in the image sequence, extracting one-column slices from each frame is not sufficient to form uniformly dense mosaics. There are two existing approaches for solving this problem. In a "manifold mosaic" [7], each image contributes a slice to the mosaic. For a translating camera, a manifold mosaic can be modeled as a *multi-perspective image*: each sub-image (with more than one column) is full perspective, but sub-images from different frames will have different viewpoints (Fig. 3a). This will cause geometric misalignments (seams) in the mosaic due to motion parallax under translation over surfaces with height variation. In the "3D mosaic + parallax" approach [11], a dense parallax map needs to be calculated for every pair of frames, and then additional pixels are added into the existing mosaic that is represented in the single reference perspective view.

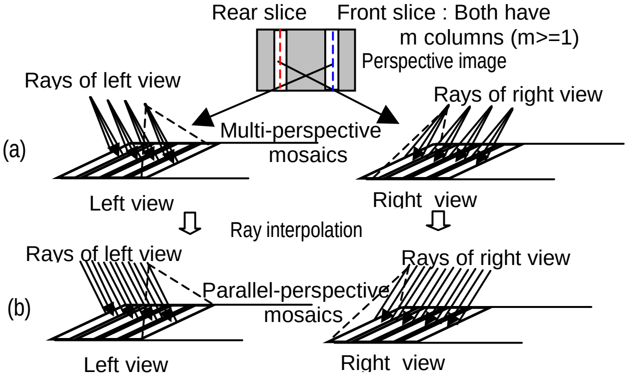

Fig. 3. Dense stereo mosaics with multi-perspective projection and parallel-perspective projection.

How can we generate seamless mosaics in a computationally effective way? The key to our approach lies in the parallel-perspective representation and a novel PRISM (*parallel ray interpolation for stereo mosaicing*) approach. For each of the left and right mosaics, we only need to take a front (or rear) slice of a certain width (determined by the interframe motion) from each frame, and perform local registration between the overlapping slices of successive frames. We then directly generate parallel interpolated *rays* between two known discrete perspective views for the left (or right) mosaic. Our approach is similar to image synthesis by view interpolation, which has been well studied in image-based rendering [18]. Fortunately in our case, we only need to perform small number of *parallel-perspective ray interpolation* instead of a complete view interpolation between a pair of successive images. In addition, the distance between two successive views are small, so the synthetic parallel-perspective rays between the two known views are not subject to serious occlusion problems.

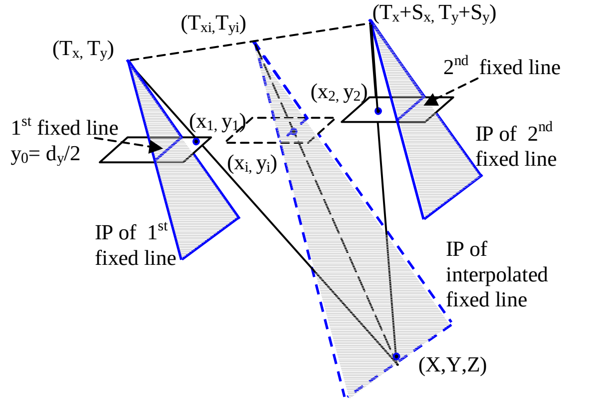

Fig. 4. PRISM: Ray interpolation by local match and ray re-projection

Let us examine the PRISM approach more rigorously in the case of 2D translation after image rectification. The extension to 3D translational case is quite straightforward; only the generalization of Eqs. (8) and (9) in the following are needed. First we define the central column of the front (or rear) mosaicing slice in each frame as a *fixed line*, which has been determined by the camera's location for that frame and the pre-selection of the front (or rear) slice window (Fig. 4, Fig. 5). An interpretation plane (IP) of the fixed line is a plane passing through the nodal point and the fixed line. By definition, the IPs of fixed lines for the left (or right) mosaic are parallel to each other. We take the left mosaic as an example. Suppose that (S_x, S_y) is the translational vector of the camera between the previous (1st) frame of viewpoint (T_x, T_y) and the current (2nd) frame of view point (T_x+S_x, T_y+S_y) (Fig. 4). We need to interpolate parallel-perspective rays between the *fixed lines* of the 1st and the 2nd frames for the mosaicing image. For each point (x_l, y_1)(to the right of the fixed line y_0=d_y/2) in frame(T_x, T_y), which will contribute to the left mosaic, we can find a corresponding point (x_2, y_2) (to the left of the fixed line) in frame (T_x+S_x, T_y+S_y). We assume that (x_1, y_1) and (x_2, y_2) are represented in their own frame coordinate systems, and intersect at a 3D point (X,Y,Z). Then the parallel reprojected viewpoint (T_{xi}, T_{yi}) of the correspondence pair can be computed as

$$T_{yi} = T_y + \frac{(y_1 - d_y/2)}{y_1 - y_2} S_y, \; T_{xi} = T_x + \frac{S_x}{S_y}(T_{yi} - T_y) \qquad (8)$$

where T_{yi} is calculated in a synthetic IP that passes through the point (X,Y,Z) and is *parallel* to the IPs of the fixed lines of the first and second frames. T_{xi} is calculated in a way such that all the viewpoints between (T_x,T_y) and (T_x+S_x, T_y+S_y) lie in a straight line. The mosaicing coordinates of the interpolated ray from this pair are

$$x_i = t_{xi} + x_1 - \frac{S_x}{S_y}(y_1 - \frac{d_y}{2}),\ y_i = t_{yi} + \frac{d_y}{2} \qquad (9)$$

where $(t_{xi}, t_{yi}) = (F\ T_{xi} / H, F\ T_{yi} / H)$ is the "scaled" viewpoint of the interpolated IP that includes the ray.

We have noticed that view interpolation has been suggested to generating seamless 2D mosaics under motion parallax [9]. The authors noted that in order to overcome the parallax problems, intermediate images could be synthetically generated between two original frames, and thus narrower strips used. Our work is different from theirs in two aspects. First, our approach is *direct* and much more efficient. We do not need to generate many new images between each pair of original frames. Instead we directly generate interpolated rays for the parallel-perspective mosaics from only two slices of a pair of successive frames. Second, we proposed to stitch two images in the middle of the two fixed lines and to consider the occluding problem so that views of points in the original images are as close as possible to the rays of the final mosaics. These issues will be further discussed in the next two sections.

5. A Fast PRISM Algorithm

We have designed a fast *3D mosaicing* algorithm based on the proposed PRISM method. It only requires matches between a set of point pairs in two successive images around their *stitching line*, which is defined as a virtual line in the middle of the two fixed lines (Fig. 5). Note that this stitching line is where a *2D mosaic* method is supposed to smoothly interface the two successive slices. The fast PRISM algorithm consists of the following four steps:

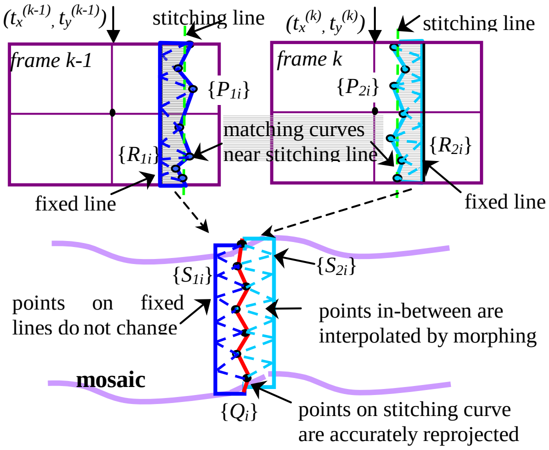
$(t_x^{(k-1)}, t_y^{(k-1)})$
stitching line
$(t_x^{(k)}, t_y^{(k)})$
stitching line
frame k-1
frame k
$\{P_{1i}\}$
$\{P_{2i}\}$
matching curves near stitching line
$\{R_{1i}\}$
$\{R_{2i}\}$
fixed line
fixed line
$\{S_{1i}\}$
$\{S_{2i}\}$
points on fixed lines do not change
points in-between are interpolated by morphing
mosaic
$\{Q_i\}$
points on stitching curve are accurately reprojected

Fig. 5. Image morphing and stitching

Step 1. *Slice determination* - Determine the fixed lines in the current frame k and the previous frame k-1 by their 2D scaled translational parameters $(t_x^{(k)}, t_y^{(k)})$ and $(t_x^{(k-1)}, t_y^{(k-1)})$. Then an "ideal" straight stitching line lies in the middle of the two fixed lines. Thus we have two overlapping slices, each of which starts from the fixed line and ends a small distance away from the stitching line (to ensure overlapping) (Fig. 5).

Step 2. *Match and ray interpolation* - Match a set of corresponding points as control point pairs in the two successive overlapping slices, $\{(P_{1i}, P_{2i}),\ i = 1,2,\ldots N\}$, in a given small region along epipolar lines, around the straight stitching line. We use a correlation-based method to find a pair of *matching curves* passing through the control points in the two frames. The control point pairs are determined by measuring both the gradient magnitudes and the correlation values of a small window centered at the control point. Then the destination locations Q_i $(i=1,\ldots,N)$ of the interpolated rays in the mosaic is computed for each corresponding pair (P_{1i}, P_{2i}) using Eq. (9). A curve that passes through the point list $\{Q_i\ (i=1,\ldots,N)\}$ is defined as the *stitching curve* where the two slices will be stitched after image warping (Fig. 5). Both the matching pairs and the destination points form *curves* instead of straight lines due to the depth variation of the control points.

Step 3. *Triangulation* - Select two sets of control points R_{mi} $(m=1,2;\ i=1,\ldots N$-$1)$ on the fixed lines in the two frames, whose x coordinates are determined by the fixed lines and whose y coordinates are the averages of P_{mi} and $P_{m,i+1}$ (m=1,2) for good triangulation. Map R_{1i} and R_{2i} into the mosaic coordinates as S_{1i} and S_{2i} $(i=1,\ldots N)$, by solely using interframe translations $(t_x^{(k)}, t_y^{(k)})$ and $(t_x^{(k-1)}, t_y^{(k-1)})$. For the *kth* frame, we generate two sets of corresponding triangles (Fig. 5): the source triangles by point sets $\{P_{2i}\}$ and $\{R_{2i}\}$, and the destination triangles by point sets $\{Q_i\}$ and $\{S_{2i}\}$. Do the same triangulation for the *(k-1)st* frame.

Step 4. *Warping* - For each of the two frames, warp each source triangle into the corresponding destination triangle, under the assumption that the region within each triangle is a planar surface given small interframe displacements. Since the two sets of destination triangles in the mosaic have the same control points on the stitching curve, the two slices will be naturally stitched in the mosaic.

6. More Discussions on Ray Interpolation

6.1. Determining stitching points

In principle, we need to match all the points between the two fixed lines of the successive frames to generate a complete parallel-perspective mosaic. In an effort to reduce the computational complexity, the fast PRISM algorithm only matches points on a "stitching curve" close to the center of the two fixed lines. The rest of the points are generated by image warping for one of the two frames, assuming that each triangle is small enough to be treated as a planar region. The locations of the stitching curves in the fast PRISM algorithm enable us to use the closest existing views to generate parallel-perspective rays. Using sparse control points and image warping, the fast PRISM algorithm only approximates the parallel-perspective geometry in stereo mosaics. However, the proposed PRISM approach can be implemented to use more feature points (thus smaller triangles) in the overlapping slices so that each triangle really covers a planar patch or a patch that is visually indistinguishable from a planar patch. Further experiments are underway.

6.2. Dealing with occlusion

If the distance between two successive views is small, thus the synthetic rays of parallel-perspective projection between the two known views are not subject to serious occlusion problems, since the two frames have almost the same occlusion relations. However, if the interframe motion is not small, we need to consider the problem of where to select the stitching points for best view synthesis under obvious occlusions. Fig. 6 shows an example on how the fast PRISM algorithm can be improved to work in the presence of occlusion. First, a pair of corresponding points at the occluding boundaries that can be seen in both images is selected as the stitching point. Then those points that can only be seen from the first image are warped from the first image to the mosaic. Note that different treatments should be made be for the cases of re-appearances and occlusions, and for left mosaics and right mosaics with different viewing directions.

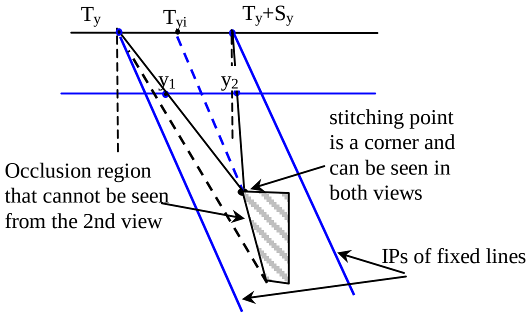
T_y
T_{yi}
T_y+S_y
y_1
y_2
stitching point is a corner and can be seen in both views
Occlusion region that cannot be seen from the 2nd view
IPs of fixed lines

Fig. 6. Dealing with occlusion: an example

6.3. From 3D to 1D track of viewpoints

Recall that in the fast PRISM algorithm, we do not change the viewpoints of the fixed lines of the existing frames. As a result, the viewpoints of the stereo mosaics are on the original (3D) camera path. If we want to generate a pair of parallel-perspective stereo mosaics with horizontal epipolar lines in the case of 3D translation, we need to completely generate stereo mosaics with all the viewpoints along a straight line. That is to say we need to synthesize every pixel in the stereo mosaics instead of directly copying the original pixels on the "fixed lines". The solution here is to fit a straight track using the points along the real 3D camera path, and then generate synthetic parallel-perspective rays along the fitted straight track by using the known camera views. In principle, the similar ray interpolation technique as the PRISM approach in Section 4 can be used. The ray synthesis will be better if the 3D camera path does not vary too much from the fitted straight track. It is one of the interesting issues we will investigate further.

7. Experimental Analysis

7.1. Parallax analysis and ray interpolation

Fig. 7 shows a real example of the local match and ray interpolation, where the interframe motion is $(s_x, s_y) = (3, 36)$ pixels, and points on the top of a long narrow building have 1-2 pixels of additional motion parallax. As we will see next, the 1-2 pixel geometric misalignments, especially of linear structures, are clearly visible to human eyes. Moreover, the perspective distortion causing the seams will introduce errors in 3D reconstruction using the parallel-perspective geometry of stereo mosaics. In this example, the distance between the front and the rear slice windows is d_y = 192 pixels, the average height of the aerial camera from the ground is H = 300 m. The relative y displacement of the building roof (to the ground) in the stereo mosaics is about Δy = -12 pixels. Using Eq. (4) we can computed that the "absolute" depth of the roof from the camera is Z = 281.25 m, and the "relative" height of the roof to the ground is ΔZ = 18.75 m. A 1-pixel misalignment will introduce a depth (height) error of δZ = 1.56 m, even if the stereo mosaics have extremely large "disparity" (d_y=192). While the relative error of the "absolute" depth of the roof ($\delta Z/Z$) is only about 0.55%, the relative error of its "relative" height ($\delta Z/\Delta Z$) is as high as 8.3%. So geometric-seamless mosaicing is very important for accurate 3D estimation.

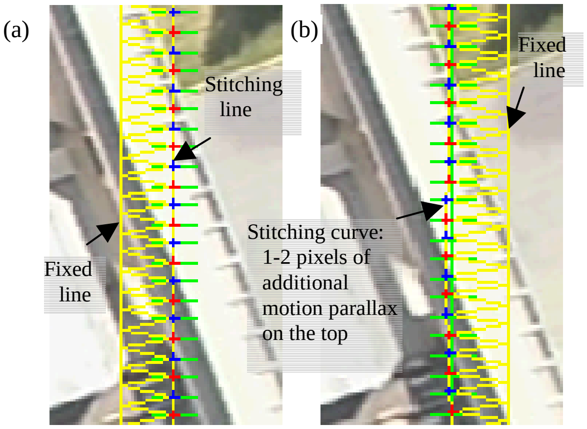
(a)
(b)
Stitching line
Fixed line
Fixed line
Stitching curve: 1-2 pixels of additional motion parallax on the top

Fig. 7. Local match and triangulation. Portions of (a) the first and (b) the second frames with (3, 36) pixels interframe motion. The light (green) crosses show the initially selected points in the previous frame and its initial matches in the current frame by using the global transformation. The dark (blue and red) crosses show the correct final match pairs by feature selection and correlation. The fixed lines, stitching lines and the triangulation results are shown as yellow. The local match results show that control points on the roof of the narrow building have larger motion parallax than ground points.

7.2. Practical treatment and 3D mosaicing results

The 3D camera orientation estimation techniques using bundle adjustments to generate georeferenced stereo mosaics were still being developed as this paper was written. Consequently, Fig. 8 shows the "free" mosaicing results where camera orientations were estimated by registering the planar ground surface of the scene via dominant motion analysis (for the detailed algorithm please see [17]) without camera calibration and bundle adjustments. However the effect of seamless mosaicing is clearly shown in this example, and such practical treatment can be applied to many image-based rendering and VR applications not requiring geo-referenced mosaics. For evaluating our fast PRISM algorithm, we compare three cases: Fig. 8a shows a tile of the multi-perspective mosaic generated using 2D mosaic method from a temporally sub-sampled image sequence (every 10 frames, i.e. the interframe motion is about 40 pixels). Geometric misalignments (seams) at the interfaces of successive slices are obvious, especially along building boundaries with depth discontinuity[2]. Fig. 8b shows a tile of the parallel-perspective mosaic of the same temporal sub-sampled image sequence as in Fig. 8a but this time the proposed 3D mosaicing algorithm PRISM is used. Most of the geometric seams visible in Fig. 8a are eliminated in Fig. 8b. Fig. 8c shows a tile of a multi-perspective mosaic when all the frames are used (i.e. the interframe motion is less than 4 pixels). In this case the multi-perspective mosaic is very close to a parallel-perspective mosaic; however, there are still "seams" in some places, e.g. areas indicated by a rectangle. It can be seen that the sparse-sampled parallel-perspective mosaic is better than the dense-sampled multi-perspective mosaic since local matches along stitching lines eliminate misalignments between two successive slices. These misalignments may be introduced by 3D structure of the scene, errors in motion modeling and errors in camera motion estimation. The fast PRISM algorithm also results in a great saving of space and time since the algorithm will work on a highly sub-sampled sequence.

7.3. 3D reconstruction from stereo mosaics

There are two benefits of generating a seamless stereo mosaic pair. First, a human can perceive the 3D scene with a stereo mosaic pair (e.g. using polarized glasses) without any 3D recovery [17]. Second, for 3D recovery, matches are only performed on the stereo mosaics, not on individual video frames. Stereo mosaic methods also solve the baseline versus field-of-view (FOV) dilemma efficiently by extending the FOV in the direction of the dominant motion. More important, the parallel-perspective stereo mosaics have fixed “disparities” and optimal/adaptive baselines for all the points. As an example, Fig. 9 shows the derived "depth" map (i.e., displacement map) from the pair of parallel-perspective stereo mosaics of a forest scene with d_y=224 (pixels). The depth map is obtained by using the Umass Terrest system based on a hierarchical sub-pixel dense correlation method [19]. In the depth map, mosaic displacement (Δy in Eq. (4)) is encoded as brightness (brightness is from 0 when $\Delta y = 18.3$ pixels, to 255 when $\Delta y = -16.2$ pixels), so higher elevations (i.e. points closer to the camera) are brighter. It should be noted that the parallel-perspective stereo mosaics were created by the fast 3D mosaicing algorithm PRISM, with the camera orientation parameters estimated by the same dominant motion analysis as in Fig. 8. Here, the fixation plane is a "virtual" plane with an average distance (*H=390 m*) from the scene to the camera. However, promising depth information has been obtained. Work on 3D recovery from parallel-perspective mosaics with accurate camera orientation and sub-pixel geometric-seamless mosaicing is underway.

8. Concluding Remarks and Future Work

We have studied the representation geometry and generation of a parallel-perspective stereoscopic mosaic pair from an image sequence captured by a camera with constrained 3D rotation and 3D translation. The inherent 3D feature of the stereo mosaics includes two aspects: (1) A 3D mosaicing process consists of a global image rectification that eliminates rotation effects, followed by a fine local transformation and ray interpolation that accounts for the interframe motion parallax due to the 3D structure of a scene. (2) The final mosaics are a stereo pair that embodies 3D information of the scene with optimal baseline. In the PRISM approach for large-scale 3D scene modeling, the computation of "match" is efficiently scattered in three steps: camera pose estimation, image mosaicing and 3D reconstruction. In estimating camera poses (for image rectification), only sparse tie points widely distributed in the two images are needed. In generating stereo mosaics, matches are only performed for ray interpolation between small overlapping regions of successive frames. In using stereo mosaics for 3D recovery, matches are only carried out between the two final mosaics, which is equivalent to finding a matching frame for every point in one of the mosaics with a fixed disparity. Thus stereo mosaics using parallel-perspective projection are a compact and efficient way to represent 3D information of a scene over a large spatial scale under a rather general motion. Future work includes calibration estimation for geo-mosaics, seamless mosaics under large motion and occlusion, and 3D reconstruction of urban scenes from stereo mosaics.

Acknowledgements

This work is partially supported by NSF EIA-9726401, and NSF CNPq EIA9970046. The authors would like to thank Howard Schultz for generating the depth map using his Terrest system, and Harpal Bassali for implementing the fast PRISM algorithm.

References

[1]. H.-C. Huang and Y.-P. Hung, Panoramic stereo imaging system with automatic disparity warping and seaming, *Graphical Models and Image Process.*, 60(3): 196-208, 1998.

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

[3]. H. -Y. Shum and R, Szeliski, Stereo reconstruction from multiperspective panoramas, *ICCV99*, 14-21, 1999.

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

[2] Please look along many building boundaries associating with depth changes in the entire 4160x1536 mosaics at [20].

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

[6]. Z. Zhu, A. R. Hanson, H. Schultz, F. Stolle, E. M. Riseman, Stereo mosaics from a moving video camera for environmental monitoring, *Int. Workshop on Digital and Computational Video*, 1999, Tampa, Florida, pp 45-54.

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

[8]. H. -Y. Shum and R, Szeliski, Construction and refinement of panoramic mosaics with global and local alignment, *ICCV'98*: 953-958.

[9]. B. Rousso, S. Peleg, I. Finci, A. Rav-Acha, Universal mosaicing using pipe projection, *ICCV'98*, pp 945-952.

[10]. R. Kumar, H. Sawhney, J. Asmuth, J. Pope and S. Hsu, Registration of video to geo-referenced imagery, *ICPR98*, vol. 2: 1393-1400

[11]. R. Kumar, P. Anandan, M. Irani, J. Bergen and K. Hanna, Representation of scenes from collections of images, In *IEEE Workshop on Presentation of Visual Scenes*, 1995: 10-17.

[12]. H.S. Sawhney, Simplifying motion and structure analysis using planar parallax and image warping. ICPR'94: 403- 408

[13]. R. Szeliski and S. B. Kang, Direct methods for visual scene reconstruction, In *IEEE Workshop on Presentation of Visual Scenes*, 1995: 26-33

[14]. J. Chai and H. -Y. Shum, Parallel projections for stereo reconstruction, *CVPR'00*: II 493-500.

[15]. Schultz, H., Hanson, A., Riseman, E., Stolle, F., Zhu. Z., A system for real-time generation of geo-referenced terrain models, *SPIE Symposium on Enabling Technologies for Law Enforcement*, Boston MA, Nov 5-8, 2000

[16]. C. C. Slama (Ed.), *Manual of Photogrammetry*, Fourth Edition, American Society of Photogrammetry, 1980.

[17]. Z. Zhu, E. M. Riseman, A. R. Hanson, Theory and practice in making seamless stereo mosaics from airborne video, CS *TR #01-01*, Umass-Amherst, Jan. 2001

[18]. S. M. Seitz and C. R. Dyer, Physically-valid view synthesis by image interpolation, In *IEEE Workshop on Presentation of Visual Scenes*, 1995.

[19]. H. Schultz. Terrain reconstruction from widely separated images, In *SPIE*. Orlando, FL, 1995.

[20]. Z. Zhu, PRISM: Parallel ray interpolation for stereo mosaics, http://www.cs.umass.edu/~zhu/StereoMosaic.html.

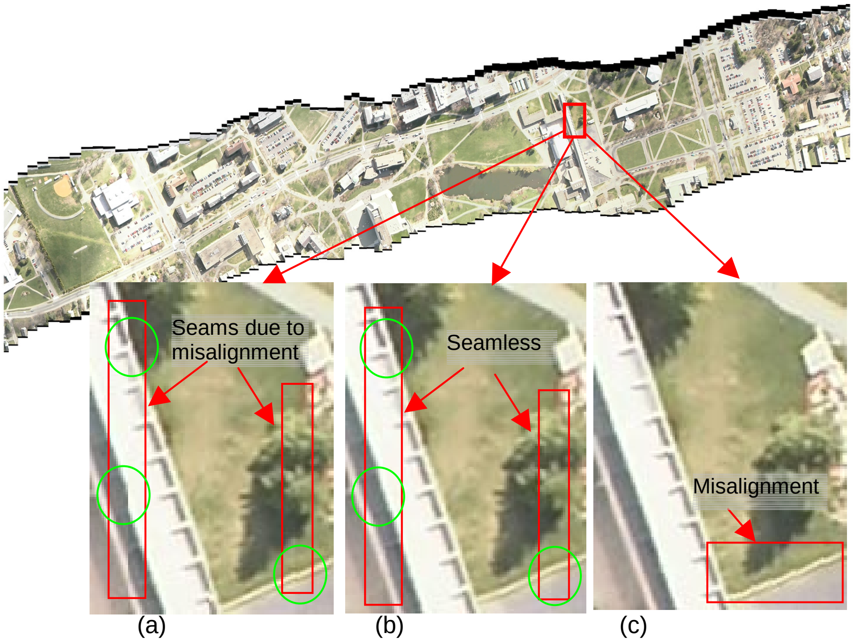

Fig. 8. Mosaics of a campus scene from an airborne camera. The parallel-perspective mosaic shown is the left mosaic generated from a sub-sampled "sparse" image sequence (every 10 frames of a 1000-frame sequence) using our 3D mosaicing algorithm PRISM. The three zoom images compare (a)multi-perspective mosaic of sparse image sequence (with geometric seams in the indicated areas, especially within the circles where depth changes); (b) parallel-perspective mosaic of sparse image sequence (no seams) and (c) multi-perspective mosaic of dense image sequence (using all the 1000 frames).

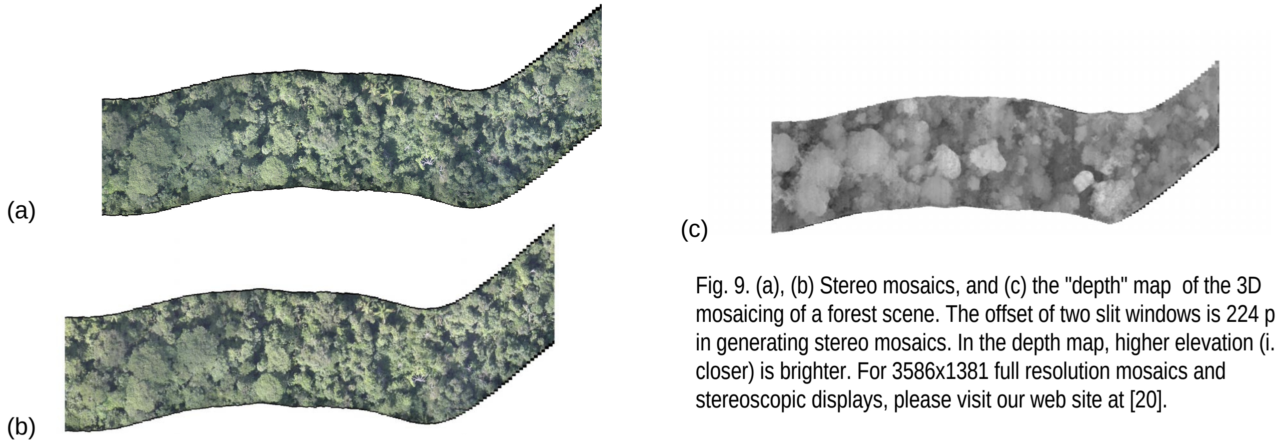

Fig. 9. (a), (b) Stereo mosaics, and (c) the "depth" map of the 3D mosaicing of a forest scene. The offset of two slit windows is 224 pixels in generating stereo mosaics. In the depth map, higher elevation (i.e. closer) is brighter. For 3586x1381 full resolution mosaics and stereoscopic displays, please visit our web site at [20].