Technical Report TR #99-28, **Computer Science Department, University of Massachusetts at Amherst, April, 1999**

Automatic Geo-Correction of Video Mosaics for Environmental Monitoring

Zhigang Zhu, Edward M. Riseman,
Allen R. Hanson, Howard Schultz

Computer Vision Lab, Computer Science Department
University of Massachusetts at Amherst, MA 01003
{zhu, riseman, hanson, hschultz}@cs.umass.edu

http://www.cs.umass.edu/~zhu

Abstract

Environmental monitoring using automated analysis of high-resolution aerial video is an application of growing importance with its own set of technical challenges. A mosaic is a commonly used tool for representing the enormous amount of data generated from video sequences. In contrast to the usual application of mosaics as a user interface, the environmental monitoring domain requires accurate geo-corrected mosaics tied to real-world coordinates. The standard techniques of generating seamless mosaics using only image data in a frame-by-frame image registration process have serious problems when applied over extended periods of time due to error accumulation, even if the errors between two successive frames are very small. Our mosaics require both seamless registration of optical data, and the use of precise flight sensor data ("geo-data") to provide a globally correct track of the motion to keep errors from propagating.

Our instrumentation package involves 3D data from GPS, a laser profiler, and an INS system to provide an actual geographical track of the data in world coordinates, referred to as "geo-data". However, there is still error in the geo-data due to inherent noise in these sensors, and data must be interpolated because it is arriving at varying temporal rates.

Thus, we face the problem of utilizing the geo-data to constrain the processing of the video to generate a seamless geographically accurate mosaic, referred to as a "geo-corrected" video mosaic. By analyzing the motion model of the flight, a pseudo-parallel projection mosaic representation (P^3 mosaic) is proposed to represent the geo-corrected mosaic. Our automatic geo-mosaic method includes local registration, track generation, matching refinement, and a two-track-based mosaic composition.

The advantage of this approach is that sensor motion information is effectively employed in a simple and robust model to produce effective results. The system is automatic and reliable since self-initialization, failure detection and outlier handling are embedded in the overall system. The two-track mosaic correction results in seamless mosaic even if the geo-data is not precise, and the methodology allows re-correction of the geo-mosaic given additional geo-referenced information from other sources, such as match with geo-referenced aerial images. The implementation is very fast because no complicated nonlinear optimization procedure is involved. No calibration and

feature extraction are needed. Experiments with real airborne video images show that the proposed algorithms are practical in the important environmental applications.

Keywords: *image registration, video mosaic, motion analysis, geo-reference image, environmental modeling*

1. Introduction

1.1. Environmental Monitoring

A critical issue among nations in the coming decades will be how to manage the use of land and natural resources. The ability to analyze vegetation and habitat from remote sensing data is currently the central problem in natural resource management on a regional and global scale. Unfortunately, the use of satellite data has not enabled general and automatic ecosystem modeling. While Geographic Information Systems (GIS) have given us the capacity to store and retrieval significant amounts of relevant data, many of the dynamic changes of interest in ecosystems take place at a finer level of resolution than current techniques can obtain.

Our interdisciplinary NSF environmental monitoring project is being conducted jointly by researchers from the Computer Science Department and the Department of Forestry and Wildlife Management at the University of Massachusetts at Amherst. The goals of aerial image analysis include classification of ground cover, multi-image 3D terrain reconstruction, computation of forest biomass, and high performance computing over large image datasets. Here, we focus on a particular aspect of our work for determination of biomass in standing forests in collaboration with The Nature Conservancy (TNC) and the National Fish and Wildlife Foundation (NFWF). Several energy oriented companies are now managing large tracts of forest for their value as sequestered carbon rather than wood products, and this practice is expected to expand as carbon emission regulations become more strict. We are developing a methodology for estimating the standing biomass of forests from a combination of large-scale videography and data from a real-time instrumentation package of GPS, a profiling pulse laser, and a two degree-of-freedom INS system.

This paper presents results on mosaics produced from aerial analysis of forest tracts in Crookesville, Ohio managed by the American Electric Power (AEP) where ground truth was available, and we were able to demonstrate accurate estimation of biomass. With continued automation and development, this methodology could have a far reaching effect on developing a new economic value for forested land that remains intact in their environment. We will be applying and refining this capability in Bolivia and Brazil in the near future.

1.2. Goals of the Aerial Mosaic Process

Our specific goal here is to develop automated tools that can correlate video mosaics from high resolution low-altitude video sequences with lower resolution high-altitude aerial image data or satellite image data that are of lower spatial resolution as a tool for interpreting the lower resolution data. We only deal with wide-angle video mosaicing in this paper, but we will be using both zoom and wide-angle video in the future. The highest resolution data (zoom video) will be displayed to an environmental expert as a magnified view to allow accurate specification of training labels for species classification. Consequently, the video mosaics must be registered to high altitude photogrammetrically accurate ortho-rectified images (i.e. a geo-referenced image) to provide the environmental expert with a realistic visualization interface.

The previous manual approach used by our forestry experts [1] only utilized a fraction of the data that is available in the GPS-logged video through human judgement of feature matches on images at different resolutions displayed on multiple screens. Obviously, automatic mosaics of tree canopy that are geo-referenced are of critical importance when huge amounts of video data are to be processed. Our planned Bolivian project involves over 600 sites and probably more than 20 hours of video, and is impossible without automation.

1.3 Related Work

Creating panoramic images and high quality mosaic images from video sequences (or a collection of images) has attracted significant attention in the research community, industry, and government (through the DARPA Video and Surveillance program (VSAM)) [2-11]. Applications span a variety of fields, including panoramic photography, video compression, surveillance, and virtual environments. The existing mosaic methods can be divided into three classes: cylindrical mosaics, free mosaics and global registration-based mosaics.

Cylindrical Mosaics - In this approach, a camera pans around a scene to obtain a complete description of the surrounding environment, and a full view (360-degree) panorama is generated. Apple's QuickTime VR [2] captures a 360-degree panoramic image of a scene with a camera rotating horizontally from a fixed position. The overlap in images is registered first by the user and

then "stitched" together by the software at the best match. Recently Kang & Weiss [3] analyzed the error in constructing panoramic images and proposed a technique that has the advantage of not having to know the camera focal length *a priori*. In order to create a panorama, they first had to ensure that the camera is rotating about an axis passing through the nodal point. The correct focal length is determined by iterating the process of projecting original video images onto cylindrical surfaces given an estimate of the focal length. In other work, in order to generate panoramic mosaics from video on a hand-held camcorder, Sawhney et al [4] provided a method for automatic detection of a loop closure to warp the conic mosaic into a cylindrical mosaic. Zhu et al [5, 18] proposed a similar methodology independently to deal with more complex camera motion - 3 degree-of freedom (DOF) rotation, zoom and small translation. Due to scale change and accumulating error, this required warping from a deformed conic mosaic to a cylindrical panorama. Generally speaking, in this class the loop closure constraint is used to connect first and last matched frames.

Free Mosaics - In this class, only an implicit assumption is made about constrained camera motion (e.g. pan) or scene structure (e.g. a planar scene). Mann and Picard [6] discussed different transformation models – affine, bilinear, projective and pseudo-projective – to register and reduce the set of images into a single, larger composite frame. The final image mosaic is not a full 360-degree view, nor is 3D geometrical correctness guaranteed**.** Peleg and Herman [7] use manifold projection to enable the fast creation of low distortion panoramic mosaics under a more general motion than exact panning. The basic principle is the alignment of the strips that contribute to the mosaic, rather than the alignment of the entire overlap between frame. However, the issues of camera focal length changes and/or depth changes are not considered in this approach. The aim of the mosaic is for entertainment rather than applications requiring geometric precision. Morimoto and Chellappa [8] presented a fast 3D stabilization and mosaic construction system. A small set of automatically selected and tracked feature points is used and an extended Kalman filter framework is employed to estimate the 3D rotation between frames. Fast implementation on a Datacube MV200 platform is reported, and the system has been tested under a variety of situations that include dominant forward and lateral translations with small rotations, as well as panning.

Problems in Generating Precise Mosaics - Experience with the first two classes of mosaic generation techniques indicates that attempts at registering a large set of images with

photogrammetric precision result in significant difficulty. Since most of the mosaic methods operate on video images in a sequential, pairwise manner, small errors in registration accumulate from one pair of images to the next. The cumulative effect produces significant error in the position of the final image, even when the individual registration error between frames is very small. These errors are unavoidable if no other constraints are provided.

Global registration-based mosaics - Full view panoramic mosaic generation tries to solve this problem by matching the last frame with the first frame and forcing the original mosaic to warp to a cylinder. Some researchers use more general global constraints to ensure that the final mosaic (composed of all the images) is globally registered. Shum & Szeliski [9] proposed a global registration strategy for a full view panorama, which establishes point correspondences in a set of images. Minimizing the projected difference of these points results in global alignment; however, the search required to determine many point correspondences can be quite slow. Sawhney et al [10] developed a local-to-global algorithm which use constraints between non-consecutive but spatially neighboring frames. A global consistency estimation of alignment parameters is iteratively performed in order to match each frame to a consistent mosaic coordinate system. The large number of parameters makes computation prohibitive for more than a few frames. Practical application of this algorithm requires efficient optimization strategies. Davis[11] provided an efficient method for finding a globally consistent registration of all images by solving a sparse linear system of equations. However the sparse linear system is valid only if any image can register only with a few other images.

1.4 Geo-Mosaic

Recently there have been a few reports on geographically-corrected ("geo-corrected" for short) mosaics. Kumar, et al [12] presented a geo-registration method that consists of (1) video-to-video frame alignment and local mosaic every second or so, (2) coarse indexing of the video mosaics in a high-attitude reference image using the geo-data, and (3) the fine geo-registration between the local video mosaics and the reference image. The time taken in the first step ranged from 30 seconds to 2 minutes for a triple of frames, each of size 320x240, on a Pentium 200MHz machine [12,13]. Step 2 and 3 rely on the matches between the video and the reference imagery that have a large time gap, and hence have quite different appearances. The fine geo-registration

requires knowledge of a reference image (geo-referenced aerial image with broader coverage) and accompanying co-registered DEM (digital elevation map). Twelve parameters are estimated by a nonlinear optimization performed in an iterative manner, requiring significant computational overhead. Bethel [14] reported the results on modeling of airborne pushbroom imagery – photography with a 1D scan system [15]. The orthorectified imagery is produced by exploiting control points and linear features (semi-automatically), and exploiting GPS/INS data wherever possible.

> *Our problem is that we will be acquiring immense amount of aerial video data that requires precise registration with broad coverage high-attitude data from a photogrammetrically accurate camera. Therefore given the geo-data from our instrumentation package defining the 3D global track of the camera, and range to the terrain at the center of each frame, what is a computationally efficient and fully automatic methodology for generating a seamless geo-corrected mosaic from a video sequence, in the absence of a high-attitude aerial image and an accompanying co-registered DEM ?*

Global registration might work if the global constraints involve registration across multiple images, but it is likely to be computationally expensive and there is no guarantee that it will be sufficiently accurate for our environmental monitoring application.

The solution to this quite challenging problem is enabled by a sophisticated aerial instrumentation package the augments the video data with 3D motion, location and range data. The geographical data ("geo-data") from our aerial instrumentation package - GPS, Laser, and INS - provides information that constrains (without accumulating error) the track of the global sensor motion and also gives distance to the often irregular terrain surface (e.g. tree canopy). However, there are still complex problems because each of the 3D aerial sensors has its own inherent noise characteristics, and each sensor collects data at varying temporal rates, which leads to temporal error as the data must be synchronized through interpolation. Thus, we must apply the 3D geo-data to fuse the 2D image sequences into a seamless and globally geo-corrected mosaic, which subsequently must be easily matched with high-altitude photogrammetric images.

In this paper we present a novel method for using this information to obtain seamless and geo-corrected mosaics. Besides the suitable problem modeling and the complete and full-automatic algorithms, the novelty of our work lies in a two-track geo-mosaic composition method that

achieves the required mosaic fidelity even if the geo-data is not accurate. The methodology allows re-correction of the geo-mosaic when given further geo-referenced information from other sources, such as a match with geo-referenced aerial images. Moreover no complex global optimization is used, and the algorithm is robust and fast. Our resulting mosaics are shown to be far more geographically accurate in the sense of geo-correction than a free mosaic. With no effort yet expended on code optimization and speedup, the video alignment step of our experimental prototype system takes only about 1 second for a pair of 320*240 color images on a Pentium 233 MHz PC, and the geo-correction and mosaic generation steps take far less time. The system has the potential to process data at video rate in the future.

The paper is organized as follows. In the next section, the airborne camera geometry and interframe motion models are given, and a pseudo-parallel projection mosaic representation is proposed. Section 3 describes a robust estimation algorithm for local image registration. Correction of mismatches guided by the geo-data and registration refinement for image mosaics are discussed in Section 4. This involves computing an estimated track from image registration and an expected track from geo-data**.** The central algorithm of creating a video mosaic that is both seamless and geo-corrected is presented in Section 5. Section 6 gives experimental results with real forestry video images over The Nature Conservancy (TNC) test site in Ohio. We compare results from a free mosaic (using commercial software of VideoBrush^{TM}), a geo-only mosaic (i.e. only using the 3D aerial instrumentation), and a geo-corrected mosaic. A discussion and conclusion is given in the last section.

2. Motion and Mosaic Models

2.1.Geographical data and geometry

The set of 3D geographical aerial data coming with the image sequences are captured with a "labtime", a common computer clock time in milliseconds that is used to synchronize the data. The set of sensor data and their recording rates are as follows:

Video Image Sequences are captured at a 30 Hz frame rate for both wide-angle video and zoom video**.**

A laser range profiler gives the distance D in meters of a point laser beam from aircraft to ground at 238Hz.

A Inertial Navigation System (INS) – the Watson box gyro provides rotation angles in degrees at 11 Hz. with

tip - the angle between gravity and the z axis of the aircraft in the direction of flight;

tilt - the angle between gravity and the z axis of the aircraft perpendicular to flight; and

heading - the clockwise direction-of-flight angle from north.

We use (A, B, Γ) to represent the heading, tip and tilt in radians.

GPS - this is standard GPS measuring the position of the camera at 1Hz (in future experiments differential GPS will be employed) with

altitude - the altitude of the aircraft from sea level in meters,

A/C northing and A/C easting - Universal Transverse Mercator (UTM) coordinates.

We use $\mathbf{T_w} = (T_e, T_n, T_a)^t$ to denote the 3D coordinates (east, north, altitude) of the camera center in a ground coordinate system.

It should be noted that the different devices, because they operating at varying rates, require us to employ linear interpolation to synchronize timing information from GPS running at 1 Hz to put all the temporal data in a common coordinate system. Of course this adds in additional error and must be accounted for in the geo-corrected mosaicing process.

The relationship between camera coordinates $\mathbf{X} = (X_t, Y_t, Z_t)^t$ at time t and ground coordinates $\mathbf{X_w} = (X_w, Y_w, Z_w)^t$ can be expressed as (Fig. 1)

$$\mathbf{X}_w = \mathbf{E}_w \mathbf{R}_w \mathbf{X}_t + \mathbf{T}_w \tag{1}$$

where

$$\mathbf{E}_w = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \quad \mathbf{R}_w = \begin{pmatrix} \cos A \cos \Gamma + \sin A \sin B \sin \Gamma & -\sin A \cos B & \cos A \sin \Gamma - \sin A \sin B \cos \Gamma \\ \sin A \cos \Gamma - \cos A \sin B \sin \Gamma & \cos A \cos B & \sin A \sin \Gamma + \cos A \sin B \cos \Gamma \\ -\cos B \sin \Gamma & -\sin B & \cos B \cos \Gamma \end{pmatrix} \tag{2}$$

The ground point coordinates (X_g, Y_g, Z_g) is the UTM location of a point on the ground that the laser beam has hit, and its altitude is relative to sea level. They are the assumed coordinates of the center of the video image, so we have $(X_t, Y_t, Z_t) = (0, 0, D_t)$ for this point. Therefore

$$\begin{pmatrix} X_g^{(t)} \\ Y_g^{(t)} \\ Z_g^{(t)} \end{pmatrix} = \begin{pmatrix} -D_t \cos A \sin \Gamma + D_t \sin A \sin B \cos \Gamma + T_e \\ D_t \sin A \sin \Gamma + D_t \cos A \sin B \cos \Gamma + T_n \\ -D_t \cos B \cos \Gamma + T_a \end{pmatrix} \tag{3}$$

where super index (t) and sub index t mean time t (or frame t).

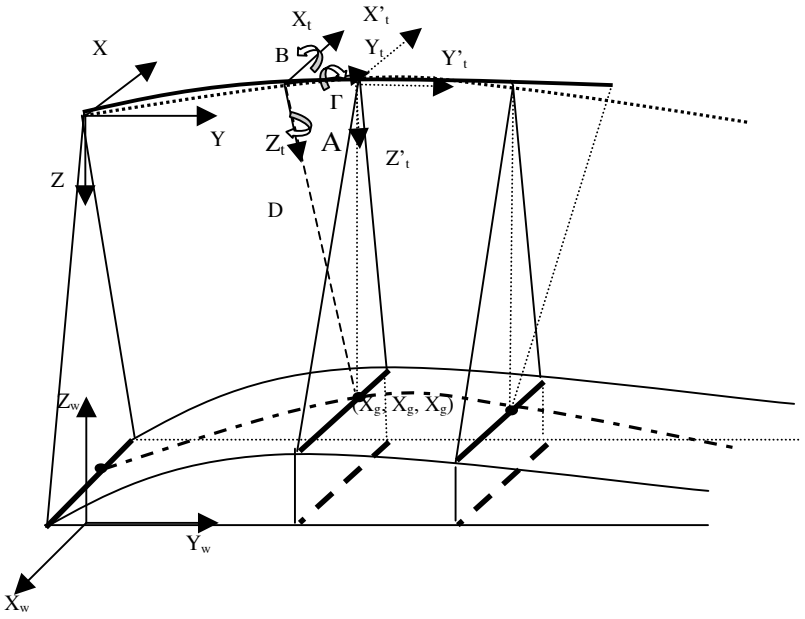


Fig. 1. Flight geometry

2.2. Interframe motion model

Equation (1) expresses the relation between the moving camera coordinate system and the fixed ground coordinate system. In order to determine the relationship between two successive video images, the interframe motion transformation must be derived. A 3D point $\mathbf{X_t} = (X_t, Y_t, Z_t)^t$ with image coordinates (u_t, v_t) at current time t will have moved from 3D point $\mathbf{X_{t-1}} = (X_{t-1}, Y_{t-1}, Z_{t-1})^t$ with the image point (u_{t-1}, v_{t-1}) at reference time t-1. The relation between the 3D coordinates is

$$\mathbf{X}_{t-1} = \mathbf{R}\mathbf{X}_t + \mathbf{T}$$

where

$$\mathbf{R} = \mathbf{R}_{w,t}^{-1}\mathbf{R}_{w,t-1}, \mathbf{T} = \mathbf{R}_{w,t}^{-1}\mathbf{E}_w^{-1}(\mathbf{T}_{\mathbf{w},t-1} - \mathbf{T}_{\mathbf{w},t}) \stackrel{\Delta}{=} (t_x, t_y, t_z)^t$$

The inter-frame rotation matrix $\mathbf{R}$ has the same form as $\mathbf{R_w}$ except that $(\mathrm{A}, \mathrm{B}, \Gamma)$ is substituted by inter-frame rotating angles (α, β, γ). If the rotation angles are small between the successive frames, e.g., less than 5 degrees , $\mathbf{R}$ can be approximated as

$$\mathbf{R} \approx \begin{pmatrix} \cos\alpha & -\sin\alpha & \gamma \\ \sin\alpha & \cos\alpha & \beta \\ -\gamma & -\beta & 1 \end{pmatrix} \approx \begin{pmatrix} 1 & -\alpha & \gamma \\ \alpha & 1 & \beta \\ -\gamma & -\beta & 1 \end{pmatrix} \tag{4}$$

The first approximation is made when α is not very small, and the second is made when all of the three angles are very small. Suppose the camera focal length f does not change during the motion. Using homogenous coordinates $\mathbf{u} = (u,v,1)^t$ for an image point, under a pinhole camera model

$$\begin{pmatrix} wu \\ wv \\ w \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \end{pmatrix} \tag{5}$$

we have

$$\begin{pmatrix} su_{t-1} \\ sv_{t-1} \\ s \end{pmatrix} \approx \begin{pmatrix} \cos\alpha & -\sin\alpha & f(Z_t\gamma + t_x)/Z_t \\ \sin\alpha & \cos\alpha & f(Z_t\beta + t_y)/Z_t \\ -\gamma/f & -\beta/f & (Z_t + t_z)/Z_t \end{pmatrix} \begin{pmatrix} u_t \\ v_t \\ 1 \end{pmatrix}$$

or

$$\mathbf{u}_{t-1} \approx \mathbf{M}_t \mathbf{u}_t \tag{6}$$

where

$$\mathrm{M}_t = \frac{1}{s}\begin{pmatrix} \cos\alpha & -\sin\alpha & t_{\mathrm{u}} \\ \sin\alpha & \cos\alpha & t_{\mathrm{v}} \\ 0 & 0 & 1 \end{pmatrix} \tag{7}$$

and

$$\begin{aligned} s &= (-u_t\gamma - v_t\beta + f + ft_z / Z_t)/f \\ t_u &= f(Z_t\gamma + t_x)/Z_t \\ t_v &= f(Z_t\beta + t_y)/Z_t \end{aligned} \tag{8}$$

For vertical tracking movement of the airborne camera (Fig. 1), involving tip, tilt, heading and range changes, we have very small β and γ. If the change in range (for the part of an image under consideration) is small relative to the range, then equation (6) can be treated as a 2D rigid inter-frame motion model, where $s \approx Z_{t-1}/Z_t$ is a scale factor associated with range changes, (t_u, t_v) is the translation vector representing (tilt/X-translation, tip/Y-translation), and α is the heading change.

When the inter-frame heading angle α is also very small, equation (6) can be further simplified as

$$\begin{cases} s \cdot u_{t-1} = u_t - v_t\alpha + t_u \\ s \cdot v_{t-1} = v_t + u_t\alpha + t_v \end{cases} \tag{9}$$

In Section 3, given more than 2 pairs of corresponding points between two frames, we can obtain the least squares solution for the motion parameters, s, t_u, t_v and α, in equation (9). The approximation errors are especially small for the narrow horizontal strip (the center scan lines) in the center of each image that will be used in our image mosaic algorithm (Fig. 3). For large heading angles, the results can be refined by an iterative warp-then-refinement method (Section 4). We argue here that even if general and optimal methods are always expected and progresses are frequently reported in motion analysis, a simplified and pragmatic model, given the specific task and the available data, is often more reliable and more efficient.

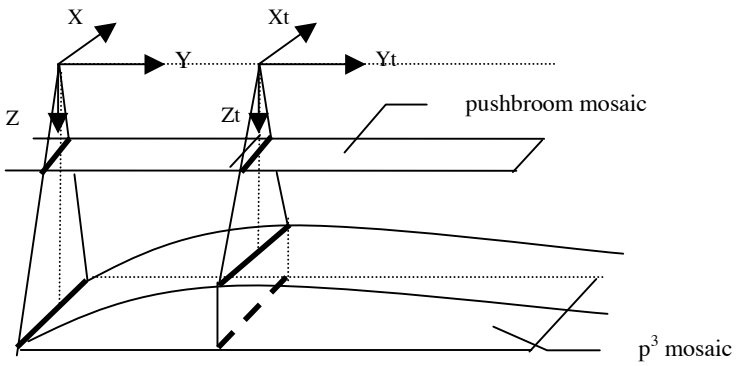


Fig.2. Pushbroom camera model and P^3 mosaic

2.3.Parallel-projection mosaic representation

A mosaic is an extended 2D image that is a composition of many of image frames. For a planar scene, a perspective mosaic image can be constructed. For a 3D rotational sequence, a cylindrical panorama can be created. In both cases, a planar projective transformation can be used. Here we propose a parallel-projection mosaic representation for a 3D scene when there are obvious range changes in the presence of translational camera motion.

(1). Semi-parallel-projection mosaic (Pushbroom mosaic)

Without loss of generality, we use the first frame coordinates (*X,Y,Z*) as the mosaic coordinate system (time $t = 0$) and assume that $(T_e, T_n, T_a)|_{t=0}$= (0,0,H), $(\mathrm{A}, \mathrm{B}, \Gamma)|_{t=0}$=(0,0,0), and the range from the camera nodal point to the ground is D_0. An ideal moving camera model is the linear pushbroom camera model [15], where at all times t, (T_e, T_n, T_a) = $(0, T_n, H)$ and $(\mathrm{A}, \mathrm{B}, \Gamma)$ =(0,0,0). The semi-parallel-projection mosaic coordinates (u_s, v_s) can be derived from

$$\begin{pmatrix} wu_s \\ v_s \\ w \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f/\mathrm{D}_0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \end{pmatrix} \tag{10}$$

Notice that the scale 1/ D_0 roughly maintains the aspect ratio of the mosaic image. The frame coordinates at time t can be expressed as

$$(u_t, v_t) = (f\frac{X}{Z}, f\frac{Y - T_y}{Z}) \tag{11}$$

where T_y is the offset of the camera center at time t from the reference frame. Semi-parallel means that in the direction of motion (Y axis) the image obeys parallel projection, while the image along the X axis is a perspective projection. The relation between the mosaic coordinates and frame coordinates is

$$(u_s, v_s) = (u_t, v_{st} + \frac{Z}{D_0} v_t) \tag{12}$$

where $(u_{st}, v_{st}) = (0, f\frac{T_y}{D})$ is the projection of the image center in the mosaic image at time t. If the image sequence is dense enough, and the velocity of the camera translation is constant or is known (i.e., T_y is known), we can use the scan lines near the center of the image (i.e. v_t = 0 so that we need not know Z in equation(12)) to construct the mosaic. Otherwise this 2D image mosaic cannot be generated from 2D registration of the image sequence if the ranges (Z) of the scene change, and are not known**.** Moreover this kind of mosaic image obeys different projections in the x and y directions, which is not expected in the geo-mosaics that we need. If the ego-motion of the camera is not an ideal linear pushbroom, the situation is even more complex.

(2). Full-parallel-projection mosaic (textured DTM)

A full parallel-projection mosaic can be expressed as

$$\begin{pmatrix} wu_f \\ wv_f \\ w \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{D}_0 \end{pmatrix} \tag{13}$$

It is actually a textured digital terrain map (DTM) – i.e. a texture-mapped range image on an (X,Y) grid). A full parallel-projection mosaic is achievable if a 3D DTM is available, and the match between 3D range map and 2D images is known [12]. In other words, we know the 3D

motion of the camera, intrinsic parameters of the camera, and the depth of every image point. The relation between the mosaic coordinates and frame coordinates is

$$(u_f, v_f) = (\frac{Z}{D_0} u_t, v_{ft} + \frac{Z}{D_0} v_t) \tag{14}$$

where (u_{ft}, v_{ft}) is the projection of image center at time t in the mosaic image.

(3). Pseudo-parallel-project mosaic (P^3 mosaic)

A full-parallel-projection mosaic is very different from a perspective projection in the sense that distant objects do not appear smaller than nearby objects, which is ideal for our geo-based mosaic. However we need the full 3D range map of the scene, and most of all, we need to match every image with the 3D model. Obviously the problem becomes one of 3D reconstruction from multiple calibrated images when the 3D map is not available *a priori*. Can we obtain the 2D geo-corrected mosaic in a much more efficient manner?

Recall that range information is available along the motion track of the camera center by our instrumentation. If we assume that the scene has constant depth D in the X direction in the camera's field of view, then a pseudo-parallel-projection mosaic (**P^3 mosaic**) can be constructed

$$\begin{pmatrix} wu_p \\ wv_p \\ w \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ D_0 \end{pmatrix} \tag{15}$$

The image transformation from a perspective image to a P^3 mosaic is

$$(u_p, v_p) = \frac{D(v_t)}{D_0}(u_t, v_t) + (0, v_{pt}) \tag{16}$$

where $(0, v_{pt})$ is the projection of center of the image at time t in the mosaic image, and $D(v_t)$ is the range in the track connecting the image centers of frame t-1, t and t+1. The properties of a P^3 mosaic are quite similar to a full-parallel-projection mosaic. Moreover it is much easier to construct. Notice that there is no difference between equations (15) and (13); however the generation of P^3 mosaic in equation (16) is different from textured DTM. In the following section we will present a fast and robust method to construct a geographically-corrected P^3 mosaic**.**

2.4. Generalization of P^3 mosaic geometry

In this subsection we generalize the linear pushbroom camera model to the real motion model of the airborne-mounted camera when the motion has 6 degrees of freedom (Fig. 1, equation (1)). During forward motion, we assume that the camera's tip and tilt do not change very much during a long flight, i.e. the plane does not "accumulate" large tip and tilt, B and Γ. However, the heading angle A can change significantly over a long flight. A 2D rigid motion model can be derived from equations (1) and (3) in a manner similar to equation (6). Let $\mathbf{u} = (u,v,1)^t$ be the coordinates in the mosaic coordinate system (i.e. frame t =0), and $\mathbf{u_t} = (u_t,v_t,1)^t$ in the current frame (i.e. frame t), we have

$$\mathbf{u} = \mathbf{P}_t \mathbf{u}_t \tag{17}$$

where

$$\mathbf{P}_t = \begin{pmatrix} S\cos A & -S\sin A & T_u \\ S\sin A & S\cos A & T_v \\ 0 & 0 & 1 \end{pmatrix} \tag{18}$$

and

$$S \approx D_t / D_0, T_u \approx fT_x/D_0, T_v \approx fT_y/D_0 \tag{19}$$

In equation (19), (T_x, T_y) is the X and Y coordinates in the reference camera coordinate system of the ground point (X_g, Y_g) in equation (3), and we have $(T_x, T_y) = (-X_g, Y_g)$ in our coordinate system definition. equations (17)-(19) implies that the mosaic image obeys parallel projection and the camera image approximately obeys a weak-perspective projection

$$(u_t,v_t) = (f X/D_t, f Y/D_t)$$

where D_t is the average depth of the portion of an image in time t that will be used in the mosaic. In other words, the original image is approximated by a weak-perspective projection of a "virtual camera" $X'_g Y'_g Z'_g$ with nodal point at $(X_g^{(t)}, Y_g^{(t)}, Z_g^{(t)} + D_t)$ (see Fig. 1).

It seems that with all the 3D geographical data and a robust 2D image registration process, our image mosaicing becomes a simple problem. However either of them alone will not result in both a seamless and geographically corrected mosaic. Using frame-by-frame image registration alone, we can achieve a seamless mosaic, but it will not exhibit geographical accuracy due to

from-frame-to-frame error accumulation, even if the errors between two successive frames are very small. These errors stem from model approximation, scene complexity, and image registration errors. On the other hand, the 3D geographical data (from GPS, laser ranges and INS) provides a globally correct track of the motion without error propagation. However the inherent (absolute) errors in the instrumentation are large, and how to match the 3D data with the 2D image is still a problem. The following sections describe an effective method to combine two different sources of data to achieve the seamless and geo-corrected mosaic, without camera calibration, 3D reconstruction or complex nonlinear global registration.

3. Initial Registration

The inter-frame image displacements are estimated by using a pyramid-based matching algorithm. The hierarchical algorithm consists of four steps: pyramid construction, hierarchical block matching, match evaluation and robust estimation of motion parameters.

Step 1: Generate the pyramids for the current and the reference (preceding) images. For computational efficiency, the final image displacements are only given for non-overlapping image blocks of a given size, say 16×16, in the finest layer (i.e. original image) of the reference frame. The matching process is carried out from coarse to fine resolution layers, starting from a layer with certain image size, e.g., 2 times as large as the matching block size. The list of the blocks is represented by their center coordinates $\{(u_i,v_i)$, i=0, ...,B-1$\}$ in the reference frame.

Step 2: Determine the image displacements. For each block in a layer of the reference frame, the absolute difference operation (a simple version of correlation) is carried out in an *adaptive* search window over the current frame pixel by pixel. Matches with largest correlation values are determined and the one with smallest displacement is selected as the best match. Notice that there may be several best matches due to similar patterns within the search window. The search window is "adaptive" in that the initial size of the search window is about half the image size in

the first layer, but it is reduced in the finer layers. The motion vectors for these blocks are presented by $\{(\Delta u_i, \Delta v_i), i=0, ...,B-1\}$.

Step 3: Evaluate each match by combining a texture measure with the correlation measurement. This step is important because the confidence values will serve as weights in the parameter estimation. The evaluation of the matching itself is calculated **from** the normalized absolute difference of each block as

$$d_i = 1.0 - \frac{1}{255 N_w} \sum_{(u,v) \in W(u_i, v_i)} | I(u,v) - I'(u + \Delta u_i, v + \Delta v_i) |$$

where $W(u_i, v_i)$ is the block centered at (u_i,v_i), N_w is the pixel number in the block, *I(.)* and *I'(.)* are the intensity values (0-255) in the reference and current frames, respectively. The texture is measured as the normalized average magnitude of the gradient image of the reference frame inside a given block i

$$g_i = \frac{1}{g_{\max} N_w} \sum_{(u,v) \in W(u_i, v_i)} | (\frac{\partial I(u,v)}{\partial u}, \frac{\partial I(u,v)}{\partial v}) |$$

where g_{max} is the maximum value of average magnitudes of all the blocks. The initial weight for the ith match is computed as

$$w_i^{(0)} = \frac{1 - e^{-\kappa d_i g_i}}{1 - e^{-\kappa}} \tag{20}$$

where $\kappa = 8.0$ in our experiments. Note that $w_i^{(0)} = 1$ iff $d_i=g_i=1$, and $w_i^{(0)} = 0$ if $d_i=0$ or $g_i=0$.

Step 4: Estimate inter-frame motion parameters. We use a weighted least mean square (WLMS) method to iteratively estimate the inter-frame motion parameters $\theta = (t_u, t_v, \alpha, s)$ in equation (9). The objective function is

$$J = \min \sum_i w_i^{(k)} (r_i^{(k)})^2, \ (r_i^{(k)})^2 = | \mathbf{u}_i - \theta^{(k)} (\mathbf{u}'_i) |^2 \tag{21}$$

where $\mathbf{u}_i = (u_i,v_i)^t$, $\mathbf{u'}_i = (u_i + \Delta u_i, v_i + \Delta v_i)^t$, i = 0,..., B-1, and the weight updating function is

$$w_i^{(k+1)} = \frac{w_i^{(0)}}{1 + (r_i^{(k)} / \rho)^2} \tag{22}$$

where the scale factor ρ is estimated as [16,17]

$$\rho = \underset{i}{median}(| r_i^{(k)} |) * 1.4826$$

assuming that the residuals can be modeled as a noisy Gaussian distribution (residuals for the non-dominant components are the outliers). It has been pointed out in [17] that a median-based estimate has excellent resistance to outliers. The iterative algorithm is given as follows.

---

(1): Initialize : k=0, $\theta^{(-1)} = (0,0,0,0)$.

(2): Find $\theta^{(k)}$ using WLMS method.

(3). Compute the distance $\Delta\theta^{(k)} = | \theta^{(k)} - \theta^{(k-1)} |$, and estimate the scale factor ρ based on the current residuals.

(4). If $| \frac{\Delta\theta^{(k)}}{\theta^{(k)}} | < \varepsilon$ (e.g. $1.0e^{-3}$), or $\rho < \varepsilon$, or iterating count k > MaxK (e.g., 20), then stop; else update the weights $w_i^{(k)}$, assign k = k+1, and then go to step (2).

The final result from this algorithm is the dominant motion of the points that satisfy the 2D rigid motion. Those points that do not satisfy the motion model – e.g. those having obvious different ranges from the average range of the entire image and the mismatched points - are treated as outliers by changing the weights in equation (22). Interested readers can find the difference between our approach and the approach in [17] in the objective function, which is based on residuals between the motion data and the parameter model, rather than an expensive

intensity difference between two images. Instead of directly applying the Geman-McLure function as in [17], the weight function in our approach combines the measures of block match reliability and the data-model residuals.

4. Correction and Refinement

4.1. Global tracks from image registration and geo-data

Here we define a track as a sequence of 2D rigid motion parameters $\Theta = (\Theta^{(0)}, \Theta^{(1)},..., \Theta^{(F)})$, where $\Theta^{(t)} = (T_x^{(t)}, T_y^{(t)}, \mathrm{A}^{(t)}, S^{(t)})$ in equations (17)-(19), and F is the frame number. As in Section 2, select the first frame as the reference frame where the mosaic coordinate system is generated. We can find the geometric transformation between the current frame and the first frame recursively from equations (6) and (17), hence the "estimated track" from the image is

$$\Theta_I^{(t)} \Leftrightarrow \mathbf{P}_t = \prod_{j=0}^{t} \mathbf{M}_j = \mathbf{P}_{t-1}\mathbf{M}_t, t = 1,...,F;\ \mathbf{P}_0 = \mathbf{I} \tag{23}$$

and the image track length (measured in pixels) is calculated as

$$L_I = \sum_{t=1}^{F} |(T_u^{(t)}, T_v^{(t)}) - (T_u^{(t-1)}, T_v^{(t-1)})| \tag{24}$$

We can also find the track on the ground from the geo-data, as

$$\Theta_{GR}^{(t)} = (-X_g^{(t)}, Y_g^{(t)}, \mathrm{A}^{(t)}, D^{(t)} / D^{(0)}), t = 1,..., F \tag{25}$$

where $(X_g^{(t)}, Y_g^{(t)})$ is the corresponding point of the image center at frame t (equation(3)), $A^{(t)}$ is the heading angle, and $D^{(t)}$ is the average range of the ground points between $(X_g^{(t)}, Y_g^{(t)})$ and $(X_g^{(t-1)}, Y_g^{(t-1)})$ (see Fig.1 and Fig. 3). The track length (measured in meters) on the ground can be calculated as

$$L_G = \sum_{t=1}^{F} |(X_g^{(t)}, Y_g^{(t)}) - (X_g^{(t-1)}, Y_g^{(t-1)})| \tag{26}$$

From equation (19), the effective focal length for the camera image and also the mosaic image can be estimated as

$$f = D_0 \frac{L_I}{L_G} \tag{27}$$

so the "expected" geo-corrected mosaic track is

$$\Theta_D^{(t)} = (-f \frac{X_g^{(t)}}{D^{(0)}}, f \frac{Y_g^{(t)}}{D^{(0)}}, \mathrm{A}^{(t)}, \frac{D^{(t)}}{D^{(0)}}), t = 1,...,F \tag{28}$$

where $D^{(t)} = D_t$. (Both of these two notations are used in the paper for convenience).

Up to this point we have two initial tracks – one from the image, another from geo-data. Ideally, if there are no errors in models, data acquisition, and processing, they would be identical, but unfortunately, these conditions can never be satisfied. Notice the distinctly different ways that the tracks are derived. The estimated track from the image is the composition of interframe transformations with accumulating error, while the expected track is captured directly from absolute 3D geo-data, with its associated inherent absolute error, but is free from error propagation.

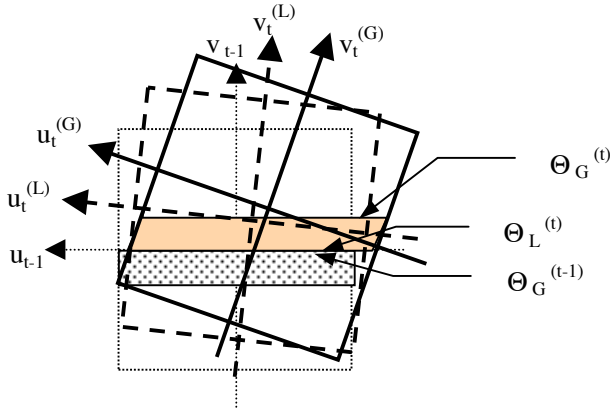


Fig.3. 2D image mosaic geometry

4.2. Match correction and refinement

In this subsection we discuss how to correct the mismatch between successive frames and refine the registration by referencing the geo-data track. If the initial estimation of interframe

motion parameters are significantly different from the results of the geo-data, and/or the *weighted* frame difference is very large, geo-data are used to estimate the initial values of the "expected" motion parameters, and then the corresponding frames are re-matched**.** Given that our goal for image registration is to create an image mosaic, the *weight* function employed for the image difference is a 1D Gaussian

$$h(u,v) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{v^2}{2\pi\sigma^2}} \tag{29}$$

which favors those points near the center scan-lines of the frames used in the mosaic images (refer to Fig.3). With the initial motion vectors of each block from the given initial interframe motion parameters, the match process will start from a suitable intermediate layer in which the initial displacements are detectable.

Even if no mismatch occurs, the refinement process is needed when the rotation angle α is large due to that α instead of $\sin\alpha$ is used in motion estimation. The refinement can be performed by iteratively warping the current image and re-matching the warped image with the reference image. We emphasize that $\boldsymbol{M}_t$ in equation (6) is used to warp the image, even if we still use linear equation (9) to estimate the motion parameters $\theta^{(m)} = (t_u, t_v, \alpha, s)|_m$, where (m) denotes the iteration count, so that errors will be reduced with decreasing residual rotating angles. The initial motion parameters $\theta^{(0)}$ for refinement are from the initial or previous match, while the initial weighting function is modified as

$$w_i^{(m,0)} = h(\mathbf{u}_i)\frac{1-e^{-\kappa d_i^{(m)} g_i}}{1-e^{-\kappa}} \tag{30}$$

The warping in the *m*th iteration can be expressed by

$$\mathbf{u}'^{(m)}_i = \mathbf{M}_t^{(m-1)}\mathbf{u}'^{(m-1)}_i, \mathbf{M}_t^{(m)} \Leftrightarrow \theta^{(m)}, m = 1,... \tag{31}$$

while the objective function is modified as

$$J_m = \min \sum_i w_i^{(m,k)} |\mathbf{u}_i - \theta^{(m,k)}(\mathbf{u}'^{(m)}_i)|^2, \begin{vmatrix} m = 1,... \\ k = 0,... \end{vmatrix} \tag{32}$$

The final transformation matrix for the current frame t is

$$\mathbf{M}_t = \prod_m \mathbf{M}_t^{(m)} \tag{33}$$

After registration correction and motion refinement, we re-compute the "refined" estimated track and the expected track, described in subsection 4.1. Notice that the geographical data is only used to correct the possible mis-registration between successive frames; it is possible to use motion smoothness constraints if the geo-data is not available, since the refinement of the motion parameters does not rely on the geo-data. From the estimated track, a seamless mosaic can be obtained. However the correction and refinement in this manner cannot correct the estimated track due to accumulating errors.

5. Geo-Mosaic Composition

We proposed a two-track method to build a geo-mosaic – a mosaic that is both seamless and geo-corrected. We already have two "tracks" of motion transformation parameters: the refined estimated image track $\Theta_{\mathbf{I}}$ and the updated expected track from geo-data, $\Theta_{\mathbf{D}}$. They are used to calculate two additional tracks: the primary track ("global-corrected track") $\Theta_{\mathbf{G}}$ that matches the global geographical track and the secondary track ("local-stitching track") $\Theta_{\mathbf{L}}$ that guarantees precise local stitching. The primary track $\Theta_{\mathbf{G}}$ is simply the expected track $\Theta_{\mathbf{D}}$, or a smooth version of it if $\Theta_{\mathbf{D}}$ is very noisy. The secondary track $\Theta_{\mathbf{L}}$ is calculated as follows:

$$\Theta_L^{(t)} \Leftrightarrow \mathbf{P}_L^{(t)} = \mathbf{P}_G^{(t-1)} \mathbf{M}_I^{(t)}, t = 1,... \tag{34}$$

where

$\Theta_L^{(t)} \Leftrightarrow \mathbf{P}_L^{(t)}$ is the (secondary) stitch transformation parameters and matrix of frame t

$\theta^{(t)} \Leftrightarrow \mathbf{M}_I^{(t)}$ is the interframe transformation from image registration of frame t and t-1

$\Theta_G^{(t-1)} \Leftrightarrow \mathbf{P}_G^{(t-1)}$ is the (primary) geo-transformation parameters of frame t-1

Now, how can we achieve precise local registration and correct the global track at the same time? From frame t, suppose we warp N+1 scanlines into the mosaic. These scanlines are expressed in the mosaic image, i.e., we use the inverse transform that maps from the mosaic to the original image frames. The transformation for the *ith* scanline is estimated as the linear interpolation of each parameters between $\Theta_L^{(t)}$ and $\Theta_G^{(t)}$

$$\Theta_i^{(t)} = \frac{(i-N)\Theta_L^{(t)} + i\Theta_G^{(t)}}{N}, i = 0,1,...,N \tag{35}$$

The mosaic process is to transform a line in frame t and paste it to the ith scanline in the mosaic (i=0,1,...,N). The first scanline from frame t will be precisely stitched to the last scanline from frame t-1, since the transformation between them is just the interframe image transformation $\mathbf{M}_I^{(t)}$; while the last scanline from frame t satisfies the geometrical transformation from the global track constraint, $\mathbf{P}_G^{(t)}$. Fig. 3 shows the geometry of line-by-line mosaic. More suitable interpolation methods can be used by considering the ranges along the line $v = 0$.

5.1. Explanation of the geo-mosaic geometry

The line-by-line 2D rigid transformation sequences capture some of the range changes and perspective distortion between the image centers of the two temporally adjacent frames. Let $\Theta_L^{(t)} = (T_{uL}, T_{vL}, A_L, S_L)|_t$, $\Theta_G^{(t)} = (T_{uG}, T_{vG}, A_G, S_G)|_t$. In our flight setting, the flight moves along the Y axis, so the number of scanlines from frame t is $N = |T_{vG}^{(t)} - T_{vG}^{(t-1)}|$. From the mosaic geometry we know that the relation between scanline i in the current frame t and the mosaic coordinate v_i is

$$i = N - (v_i - T_{vG}^{(t)}) \tag{36}$$

so equation (34) can be re-written as

$$\Theta_i = \Theta + v_i \Delta\Theta, \quad i = 0,1,...,N \tag{37}$$

where the super index (t) is dropped off for convenience, and

$$\Theta_i = (T_{ui}, T_{vi}, A_i, S_i)$$

$$\Theta = (T_u, T_v, A, S) = \Theta_G + \frac{T_{vG}}{N}(\Theta_G - \Theta_L)$$

$$\Delta\Theta = (\Delta T_u, \Delta T_v, \Delta A, \Delta S) = -\frac{T_{vG}}{N}(\Theta_G - \Theta_L).$$

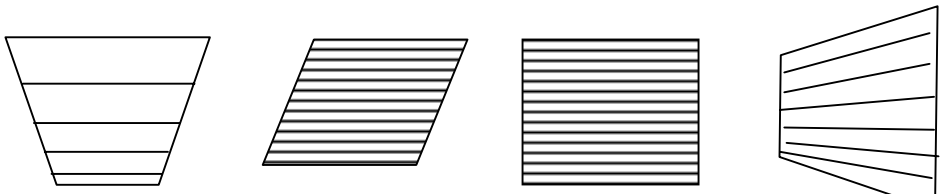
Thus the line-by-line 2D rigid transformation sequence can be represented by a unified equation

$$\begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix} = \begin{pmatrix} \mathrm{S}_i \cos \mathrm{A}_i & -\mathrm{S}_i \sin \mathrm{A}_i & T_{\mathrm{ui}} \\ \mathrm{S}_i \sin \mathrm{A}_i & \mathrm{S}_i \cos \mathrm{A}_i & T_{\mathrm{vi}} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_t \\ v_t \\ 1 \end{pmatrix} \tag{38}$$

where $\Theta_i = (T_{ui}, T_{vi}, A_i, S_i)$ is a function of $\mathrm{v_i}$. Equation (38) is a nonlinear transformation in general. If we assume $\Delta A = 0$, the following homogeneous transformation can be derived from equations (37) and (38)

$$\begin{pmatrix} wu_i \\ wv_i \\ w \end{pmatrix} = \begin{pmatrix} S\cos \mathrm{A} + C_1 & -S\sin \mathrm{A} + C_2 & \mathrm{T_u} + C3 \\ \mathrm{S}\sin \mathrm{A} & \mathrm{S}\cos\mathrm{A} & T_{\mathrm{v}} \\ -\Delta\mathrm{S}\sin \mathrm{A} & -\Delta\mathrm{S}c\cos\mathrm{A} & 1 - \Delta T_{\mathrm{v}} \end{pmatrix} \begin{pmatrix} u_t \\ v_t \\ 1 \end{pmatrix} \tag{39}$$

where $C_1 = -S\Delta\mathbf{T}^t\mathbf{A}_1 + \Delta S\mathbf{T}^t\mathbf{A}_1$, $C_2 = S\Delta\mathbf{T}^t\mathbf{A}_2 - \Delta S\mathbf{T}^t\mathbf{A}_2$, $C_3 = -\mathbf{T}^t(\Delta T_v, \Delta T_u)$ (t denotes transpose), and $\Delta\mathbf{T} = (\Delta T_{\mathrm{u}}, \Delta T_{\mathrm{v}})$, $\mathbf{T} = (T_{\mathrm{u}}, T_{\mathrm{v}})$, $\mathbf{A}_1 = (-\sin \mathrm{A}, \cos \mathrm{A})$, $\mathbf{A}_2 = (\cos \mathrm{A}, \sin \mathrm{A})$.



(a) $\Delta\mathrm{S} \neq 0$ (b) $\Delta\mathrm{T_u} \neq 0$ (c) $\Delta\mathrm{T_v} \neq 0$ (d) $\Delta\mathrm{A} \neq 0$

Fig. 4. Examples of line-by-line transformations

Fig. 4 shows some simple examples of the transformations, corresponding to the following four special cases.

(1) If $\Delta T_u = 0$, $\Delta A = 0$, $\Delta S = 0$, but $\Delta T_v \neq 0$, the two-track transformations bring scaling $(1 - \Delta T_v)$ in the v direction

$$\begin{pmatrix} u_i \\ wv_i \\ w \end{pmatrix} = \begin{pmatrix} S\cos A & -S\sin A & T_u \\ S\sin A & S\cos A & T_v \\ 0 & 0 & 1-\Delta T_v \end{pmatrix} \begin{pmatrix} u_t \\ v_t \\ 1 \end{pmatrix}$$

(2) If $\Delta T_v = 0$, $\Delta A = 0$, $\Delta S = 0$, but $\Delta T_u \neq 0$, the two-track transformations bring shearing in the u direction

$$\begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix} = \begin{pmatrix} S(\cos A + \Delta T_u \sin A) & -S(\sin A - \Delta T_u \cos A) & T_u + \Delta T_u T_v \\ S\sin A & S\cos A & T_v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_t \\ v_t \\ 1 \end{pmatrix}$$

(3) If $\Delta T_u = 0$, $\Delta T_v = 0$, $\Delta A = 0$, but $\Delta S \neq 0$, the two-track transformations bring perspective effect to the image

$$\begin{pmatrix} wu_i \\ wv_i \\ w \end{pmatrix} = \begin{pmatrix} S\cos A + \Delta S\mathbf{T}^t\mathbf{A}_1 & -S\sin A - \Delta S\mathbf{T}^t\mathbf{A}_2 & T_u \\ S\sin A & S\cos A & T_v \\ -\Delta S\sin A & -\Delta S\cos A & 1 \end{pmatrix} \begin{pmatrix} u_t \\ v_t \\ 1 \end{pmatrix}$$

(4) If $\Delta T_u = 0$, $\Delta T_v = 0$, $\Delta S = 0$, but $\Delta A \neq 0$, the two-track transformations cannot be expressed as a linear transformation in equation (38). However they are very close to a projective transformation:

$$\begin{pmatrix} wu_i \\ wv_i \\ w \end{pmatrix} = \begin{pmatrix} S(\cos A - T_u \Delta A) & -S(\sin A + T_v \Delta A) & T_u \\ S\sin A & S\cos A & T_v \\ -S\Delta A & 0 & 1 \end{pmatrix} \begin{pmatrix} u_t \\ v_t \\ 1 \end{pmatrix} \qquad (40)$$

For the real geographical image mosaic, the difference between the interframe transformations from image registration and the geo-data is very small, so the line-by-line transformations compensate the original distortion due to 3D geometry and 2D perspective

projection, and bias due to error propagation, rather than bring in additional distortion to the geo-mosaic. In fact, the two-track methods can correct the track of the free mosaic to any expected track if the track is smooth. It is clear that the two-track algorithm is computationally fast.

6. Experimental Results

Fig. 5. shows four frames of a 53-frame sampled video sequence for which a full set of geo-data are available. The original image sequence is sub-sampled to 1 frame per second with image size 320*240 in our experiment (see online MPEG file **TR99-28-1.mpg**). Recalling the data rates for GPS/INS/Laser data, every digitized frame is linearly interpolated to correspond to a GPS location and INS rotation angles for the camera. Between the center of consecutive frames there are about 238 range samples.

It should be noted that there are obvious illumination changes due to auto iris effects (see Fig. 5 (c) and (d)). The interframe translation along the Y axis is about 60-70 pixels, which is more than 1/4 of the image height**.** The algorithms for local registration and refinement are effective in the construction of a seamless free mosaic (shown at 20% scale in Fig. 6; the full-resolution mosaic can be found online in JPEG file **TR99-28-2.jpg**). Although it appears to be seamless, there are obvious differences between the estimated track from image registration and the expected track from the geo-data.

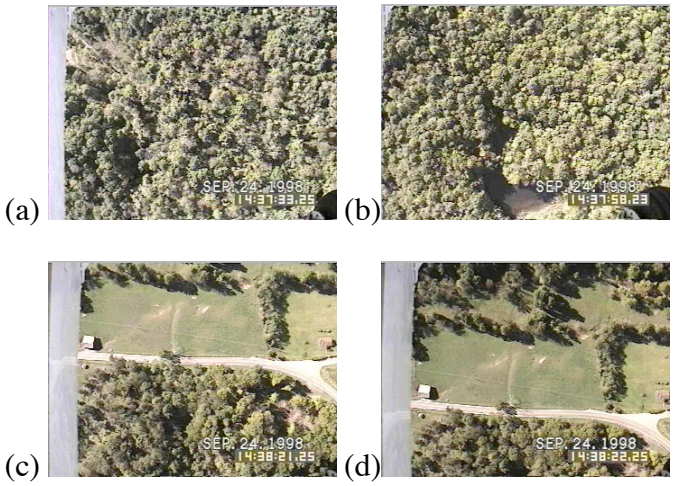
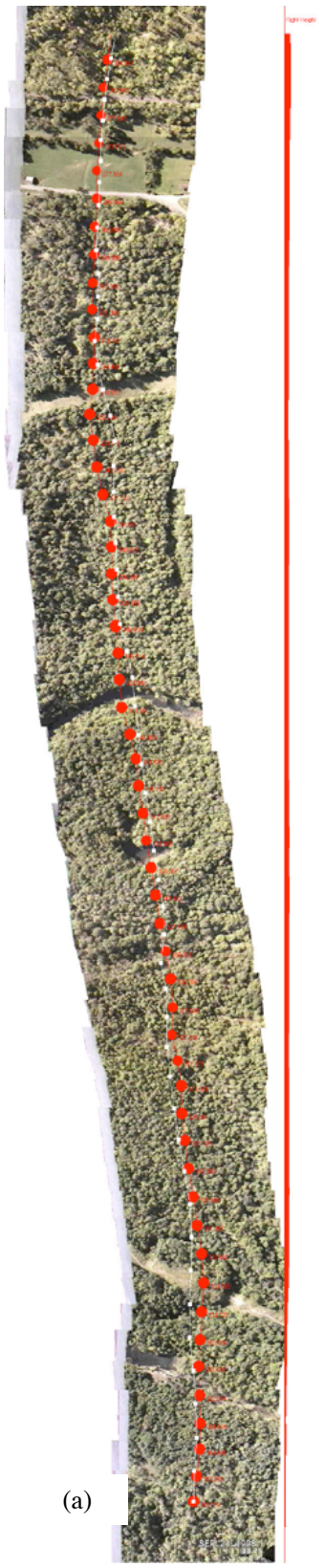


Fig. 5. Sampled frames from the forestry video sequence



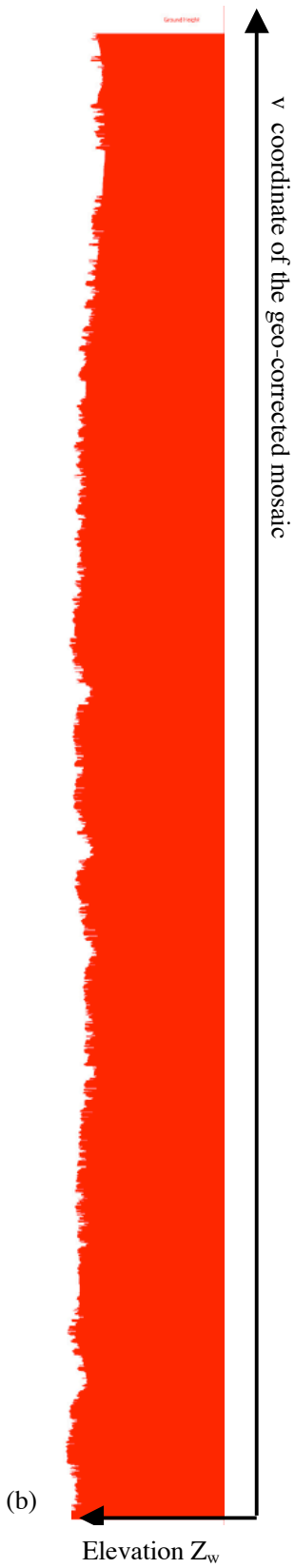
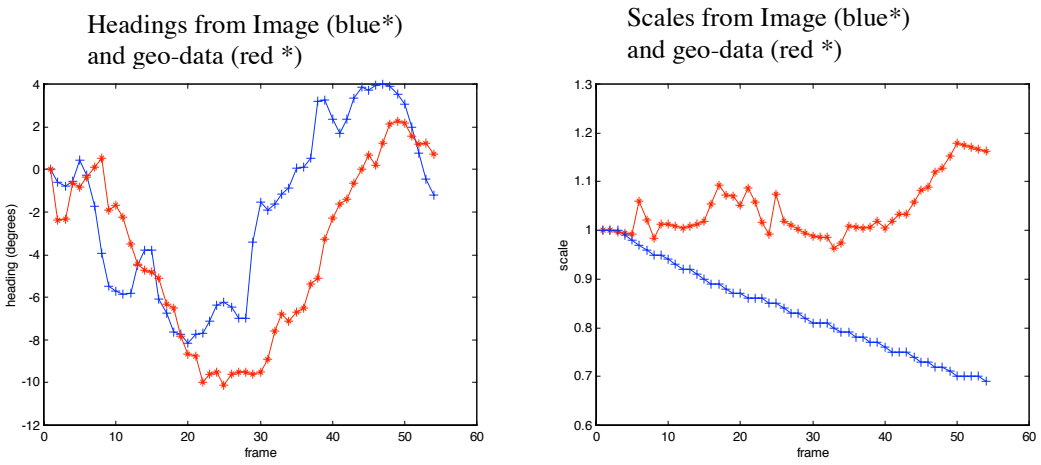


Fig. 6. Our free mosaic Fig. 7. VideoBrush mosaic Fig. 8. Geo-corrected mosaic



(a) (b)

Fig. 9 Headings and Scales in the two tracks

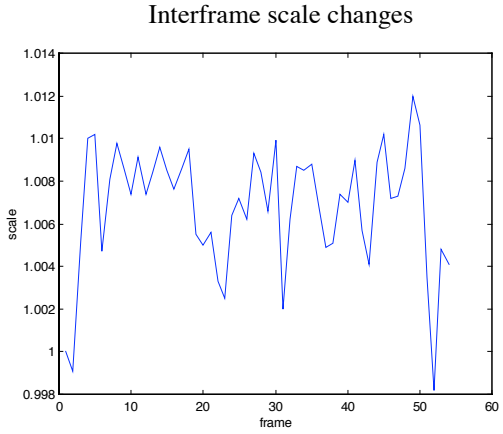


Fig. 10. Interframe scale changes

By using the two-track method described in Section 5, a seamless, geo-corrected mosaic is created (Fig. 8; **TR99-28-3.jpg**). The (translation components of) the expected track and the estimated track are superimposed in the geo-corrected mosaic in red and white respectively. The translation components of the two tracks are found to be very close to each other except for certain locations. Fig. 9 (a) and (b) show the comparisons of the headings and scales of the two tracks, respectively. The global trends of the headings are similar, but the scales are quite different, which is obvious by comparing the mosaics in Fig. 6 and Fig. 8. The expected scales

are calculated from the absolute geo-data, but the estimated scales are accumulated from interframe motion parameters. Although the estimated interframe scales are within 0.998 –1.012 (Fig. 10), which is quite close to the real situation, the accumulating errors are as large as 30% by the end of the 53 frame sequence. Readers should note that we will be doing sequences of many minutes to hours in the future.

The two-track method corrects this accumulating error frame by frame, while maintaining precise stitching of successive frames. To show the role of the secondary (stitch) track, we compare our geo-corrected mosaic to a geo-only mosaic where only the normalized expected transformation is applied. It is obvious that the geo-only mosaic is not seamless even though the global track is faithful to the geographical data, which is not accurate enough for a seamless mosaic. The geo-corrected mosaic satisfies both of the requirements. Fig. 11 is a comparison of a sub-image (33% scale in the figure) of the same portion of the geo-corrected mosaics and geo-only mosaic. The entire geo-only mosaic can be found in the online JPEG file **TR99-28--4.jpg**.

(a) geo-corrected (b) geo-only

Fig. 11. Zoom-in comparison of geo-corrected and geo-only mosaic

(a) geo-corrected (b) VideoBrush

Fig. 12. Zoom-in comparison of geo-corrected and VideoBrush mosaic

For comparison, we also generated a free mosaic using the evaluation version of a commercial software VideoBrush 2.0 (Fig. 7; **TR99-28-5.jpg**). From their published papers [4, 7] related to this system and the mosaic results, we find that no scaling is applied to the mosaic and the track is smoothed. While the mosaic is seamless, and color/ luminance blending is handled, it is not geo-corrected; for example the scale is not changed as a function of ranges of ground points. To evaluate our geo-mosaic result, we superimpose both our geo-mosaic and VideoBrush's mosaic on the high attitude image which is taken at 5k ft above the ground at about the same time as the video sequence. This high-attitude image has then been warped into a geo-reference frame. Therefore the warped high-attitude image is an approximation of a parallel projection. To register the mosaics with the high-attitude image, we manually select two points in the high-attitude image and find corresponding points in both the geo-mosaic and VideoBrush mosaic respectively; and use 2D rigid transformation of translation, rotation and scaling. Note that no deformation is brought in; only the scaling and orientation of each mosaic are changed.

Fig. 12 is a comparison of a sub-image of the same portion of the geo-corrected mosaics and VideoBrush mosaic superimposed on the high attitude image. The entire mosaic can be found in the online JPEG files **TR99-28-6.jpg** (geo-mosaic) and **TR99-28-7.jpg** (VideoBrush mosaic). Image points along the center track in our geo-mosaic register precisely with the high-attitude image, and at the border of the mosaic there are only small errors (due to the assumption of the constant range on u direction- a discussion and extension is given in the next subsection). As expected, the VideoBrush mosaic cannot register with the high-attitude image (Fig. 12b). It should be noted here that two feature points are selected in the head and tail of both of the mosaics; one of them is under the white circle (O) in each of the image in Fig. 12. Notice the obvious different location errors of a white building (pointed by an arrow) below the road in the right of each image. The reason for large offset in VideoBrush's mosaic is that it does not change the scales with the change of the ranges, which is obvious in this part of the scene. Remember that the video used in this example is only about 1 minute, but we are carrying out experiments on much longer image sequences in actual environmental applications.

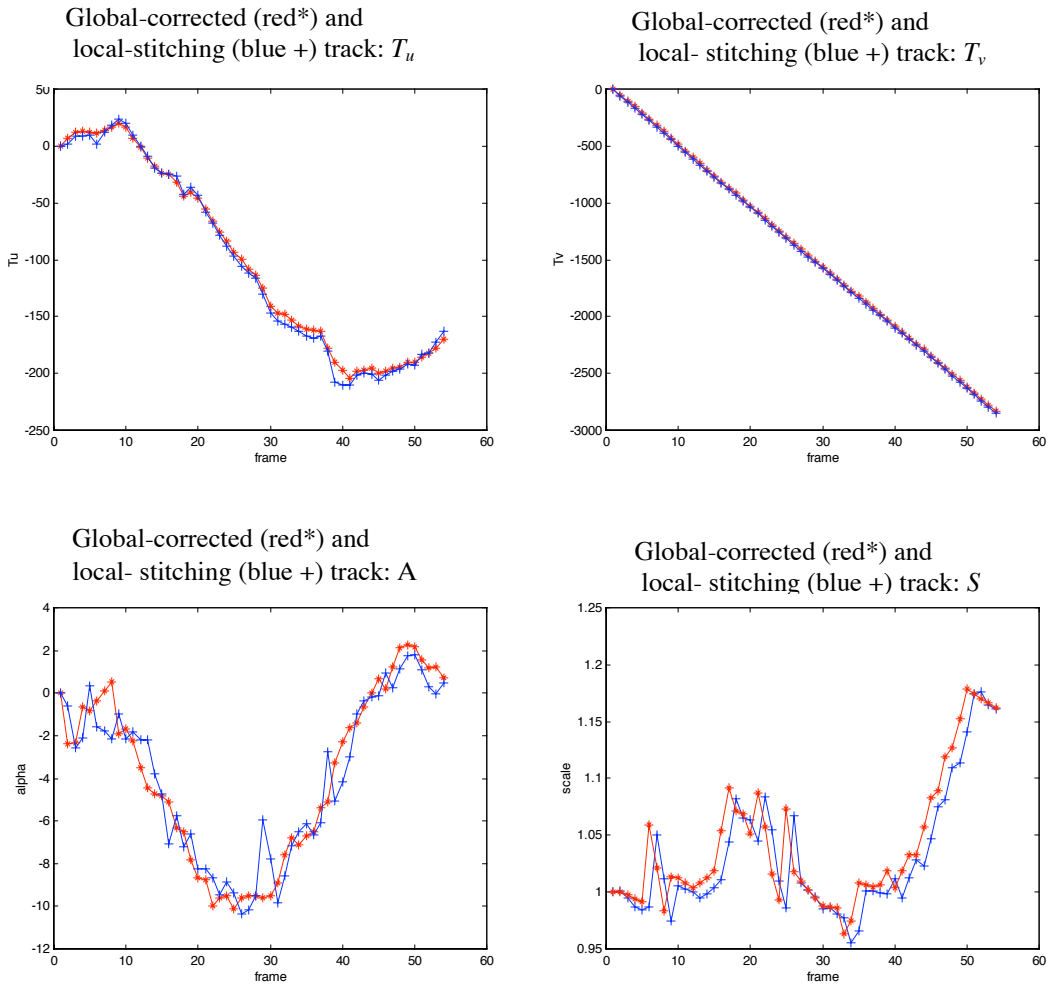


Fig. 13. Two tracks: global corrected and local precise

In the geo-corrected mosaic of Fig. 8, the matching of the 2D image with the 3D geo-data are also shown. Each red circle and the attached number indicate the center of each frame and the ground altitude in meters of that point. The radius of the red circle represents the error in pixels corresponding to a ±5 m location error on the ground. The recovered altitudes of the flight and the altitudes of the ground along the track are shown beside the geo-mosaic as histograms. The geo-corrected mosaic image matches quite well with the geo-data, for example, the roads and the grassland in the mosaic image. Fig. 13 shows the 4 components of the two tracks: the global-corrected tacks and local- stitching tracks that has the image mosaicing process incorporated. They lay close to each other, but differences exist.

7. Conclusion and Discussion

A new method of creating a seamless and geo-corrected video mosaic has been presented. By analyzing the motion model of the flight, a pseudo-parallel projection mosaic representation (P^3 mosaic) is developed to represent the geo-corrected mosaic given the available geographical data. A complete geo-mosaic method, including local registration, track generation, matching refinement and two-track-based mosaic composition are provided. The advantages of this approach are that sensor motion information is effectively employed in a simple model to produce effective results, and a fast, robust and practical implementation is achieved.

Comparing with Kumar et al's geo-registration method [12], our approach has three distinctive features. (1) Large geo-referenced video mosaic from a long image sequence can be generated before the match of video and reference imagery. Thus the computational burden for the registration of overlapped video frames (or a video mosaic in every 1 second) with the reference image can be greatly reduce, if we have to do so. (2) Only geo-data from GPS/INS/Laser are used to generate a geo-mosaic, without the need of a geo-referenced image and the accompanying DEM. Notice that besides the computational burden and difficulties in matching two different kinds of images in their method, the error in 3D DEM may distort the video mosaic such that seamless-ness may not be guaranteed. (3) Our algorithms are fast and reliable. The time for frame alignment is about 1 s for a pair of images comparing to their 30s – 2 mins for a triple of images on the similar machine. This is due to the different modeling and methodologies.

However we should point out that there are limitations in our current implementation. One potential weakness of our current work is the assumption of constant range along the x-axis. The simplified model we use is due to the availability of range along the optical axis and hence along the center line of the flight path. As a result, image points along the center track in our geo-mosaic register precisely with the high-attitude image, but at the border of the mosaic there are small errors. This can be improved by the following extensions.

- *Generalization of the geo-corrected mosaic method*. If a DEM is available or motion parallax can be reliably applied, more complicated model, e.g. projective transformation model can be utilized between two video frames. Note that the way a geo-mosaic is created – one strip from each frame, and one transform per scanline, so projective transformation would model the depth change along x axis well without dramatically increasing the computational burden if pointwise 3D data is a applied. Geo-referenced DEM can be used to generate the corresponding "projective track", and the same two-track method can be applied except that the transformations between scanlines are changed to projective transformation. In the absence of a available DEM, motion parallax can be explored – though with some difficulties.

- *Registration of aerial image and video mosaics*. With a geo-reference aerial imagery available, the registration can be carried out to reduce the error from the model simplification, without the need of a accompanying DEM. After a few reliable matches along the boundary of the video mosaic are established, the same technique of line-by-line transformations can be used before or after the generation of the video mosaic. The geo-registration between the aerial image and a few video images provide more accurate global track, while seamless and high quality mosaic can be guaranteed by our approach.

We will be carrying out a project in the Noel Kempff Mercado National Park expansion zone in Bolivia as part of a potential purchase of land area half the size of Massachusetts. We hope to demonstrate in that biomass survey that we can provide more accurate estimates of biomass at a considerably lower cost than the present method of extensive ground plot monitoring and statistical extrapolation. This geo-corrected mosaics are a key part of this process.

Acknowledgements

This work is partially supported by National Science Foundation (NSF) under grant number EIA-9726401, and the National Fish and Wildlife Foundation (NFWF) under Agreement #98089. Additional funding for this work is supported by the Nature Conservancy in conjunction with the Winrock Corporation. The authors are grateful to Mr. Kris Denio and Mr. Chris Holmes for helping collect the video and process the geographical data.

References

[1]. D. M. Slaymaker, K. M. L. Jones, C. R. Griffin and J. T. Finn, Mapping deciduous forests in Southern New England using aerial videography and hyerclustered multi-temporal Landsat TM imagery. Gap Analysis, A Landscape Approach to Biodiversity Planning, American Society for Photogrammetry and Remote Sensing, Bethesda, Maryland, 1996.

[2]. S. E. Chen, QuickTime VR - an image based approach to virtual environment navigation, *Proc. SIGGRAPH 95*, pp. 29-38, New York, 1995. ACM.

[3]. S. B. Kang, R. Weiss, Characteristics of errors in compositing panoramic images, *Proc. CVPR 97*, 103-109.

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

[5]. Z. Zhu, G. Xu, E.M. Riseman and A. R. Hanson, Fast Generation of Dynamic and Multi-Resolution 360° Panorama from Video Sequences, *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, Florence, Italy, June 7-11, 1999, to appear.

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

[7]. S. Peleg, J. Herman, Panoramic Mosaics by Manifold Projection. *Proc. CVPR 97,* 338-343.

[8]. C. Morimoto, R. Chellappa, Fast 3D stabilization and mosaic construction. *Proc. CVPR 97,* 660-665.

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

[10]. H.S. Sawhney, S. Hsu and R. Kumar, Robust video mosaicing through topology inference and local to global alignment, *Proc. ECCV* 98, vol. 2. 103-119.

[11]. J. Davis, Mosaics of scenes with moving objects, *Proc. CVPR* 98, 354-360.

[12]. R. Kumar, H. Sawhney, J. Asmuth, J. Pope and S. Hsu, Registration of Video to Geo-referenced Imagery, *ICPR98*, vol. 2: 1393-1400

[13]. H. Sawhney, R. Kumar, True multi-image alignment and its application to mosaicing and lens distortion correction, *IEEE Trans PAMI*, vol. 21, no 3, 1999: 235-243

[14]. J. Bethel, Photogrammetric research for image registration and surface extraction. *MURI Mid-term Review Meeting*, March 24, 1999

[15]. R. Gupta , R. Hartley, Linear pushbroom cameras, *IEEE Trans PAMI*, 19(9), Sep. 1997: 963-975

[16]. P.J. Rousseeuw, A.M. Leroy, *Robust Regression and Outlier Detection*, J. Wiley & Sons, New York, 1987.

[17]. Sawhney H S, Ayer S, Compact representation of videos through dominant and multiple motion estimation", *IEEE Trans. PAMI,* Vol. 18, No. 8, Aug 1996, pp814-830

[18]. Zhigang Zhu, Guangyou Xu, Allen R. Hanson and Edward M. Riseman, Fast construction of dynamic and multi-resolution 360-degree panorama from video sequences. *Technical Report TR #99-10*, Computer Science Department, University of Massachusetts at Amherst, February, 1999.