Content-Based Dynamic 3D Mosaics

Zhigang Zhu, Hao Tang
Department of Computer Science, The City College of New York, New York, NY 10031
{zhu | tang}@cs.ccny.cuny.edu

Abstract

We propose a content-based 3D mosaic (CB3M) representation for long video sequences of 3D and dynamic scenes captured by a camera on a mobile platform. After a set of parallel-perspective (pushbroom) mosaics with varying viewing directions is generated, a multi-view, segmentation-based stereo matching algorithm is applied to extract parametric representations of the color, structure and motion of the dynamic and/or 3D objects in urban scenes. We use the fact that all the static objects obey the epipolar geometry of pushbroom stereo, whereas an independent moving object either violates the epipolar geometry if the motion is not in the direction of sensor motion or exhibits unusual 3D structures. The CB3M is a highly compressed visual representation for a very long video sequence of a dynamic 3D scene. More importantly, the CB3M representation has object contents of both 3D and motion. Experimental results are given for the CB3M construction for both simulated and real video sequences.

1. Introduction

In this paper we address the problems of visual representation for large amount of video stream data, of three-dimensional (3D) urban scenes in particular, captured by a camera mounted on an airborne or a ground mobile platform. Applications include airborne or ground video surveillance for moving target extraction, automated 3D urban scene construction, airborne traffic monitoring, and image-based modeling and rendering. For these applications, hours of video streams may be generated every time the mobile platform performs a data collection task. The data amount is in the order of 100 GB per hour for standard 640*480 raw color images. The huge amount of video data not only poses difficulties in data recording and archiving but also is prohibitive for users to retrieve and to review. In the past, video mosaic approaches [1,2,3,4] have been proposed for video representation and compression, but most of the work is for panning (rotating) cameras instead of moving (translating) cameras that are mostly used in the cases of airborne or ground mobile surveillance and scene modeling. In those applications, obvious motion parallax is the main characterization of the video sequences due to the self-motion of the sensors. Some work has been done in 3D reconstruction of panoramic mosaics [5,6], but usually the results are 3D depth maps of static scenes instead of high-level 3D representations for both static and dynamic target extraction and indexing. Layered representations [7,8,9] have been studied for motion sequence representations; however, the methods are usually computationally expensive, and the outputs are typically motion segmentation represented by affine planes instead of true 3D information. Efficient, high-level, content-based, and very low bit-rate representations of 3D scenes and moving targets are still in great demand.

We propose a content-based 3D mosaic representation (CB3M) for long video sequences of 3D and dynamic scenes captured by a camera mounted on a mobile platform. The motion of the camera has a dominant direction of motion (as on an airplane or ground vehicle), but 6 DOF motion is allowed. In the first step, a set of generalized parallel-perspective (pushbroom) mosaics with varying viewing directions is generated to capture both the 3D and dynamic aspects of the scene under the camera coverage. A ray interpolation approach [10] is used here to generate multiple seamless parallel-perspective mosaics under the obvious motion parallax of a translating camera. The set of the multi-view *dynamic* pushbroom mosaics, with a pair of stereo mosaics as the minimum sub-set, is a compact visual representation for a long video sequence of a *3D* scene with independent *moving* targets.

However, the 2D mosaic representation is still an image-based one without object content representation. Therefore, in the second step, a segmentation-based stereo matching algorithm [11] has been designed to extract parametric representation of the color, structure and motion of the dynamic and/or 3D objects in urban scenes, where a lot of planar surfaces exist. In the algorithm, we use the fact that all the static objects obey the epipolar geometry, i.e. along the epipolar lines of pushbroom stereo. An independent moving object, on the other hand, either violates the epipolar geometry if the motion is not in the direction of sensor motion or exhibits unusual 3D structure, e.g., obviously hanging above the road or hiding below the road. Further in this paper, multiple pairs of stereo mosaics are used for facilitating reliable stereo matching, occlusion handling, accurate 3D reconstruction and robust moving target detection.

Based on the above two steps, a content-based 3D mosaic (CB3M) representation is created for the long video sequence. This is a highly compressed visual representation for the video sequence of a dynamic 3D

scene. For example, a real image sequence of a campus scene has 1000 frames of 640*480 color images. With its CB3M representation, a compression ratio of more than 10,000 is achieved. More importantly, the CB3M representation has object contents. We will also show the accuracy of 3D reconstruction and moving target detection by using a simulated video sequence while ground truth data is available.

The rest of the paper is organized as the follows. In Section 2, the mathematical framework of the dynamic pushbroom stereo is given, and then its properties for moving target extraction are discussed. In Section 3, multi-view pushbroom mosaics are proposed for image-based rendering and for extracting 3D structure and moving targets. In Section 4, our multi-view stereo matching algorithm for 3D static and moving target extraction will be provided. Then in Section 5, the content-based 3D mosaic representation is described. Experimental results of CB3M representation construction will be given in Section 6 with both simulated and real video data. Section 7 gives concluding remarks.

2. Dynamic Pushbroom Stereo Mosaics

For show the concept, let us first 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 of pixels (perpendicular to the motion) at the leading and trailing edges of each frame in motion (Fig. 1). 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. The geometry in this ideal case (i.e. 1D translation with constant speed) is the same as the linear pushbroom camera model [15]. Therefore we also call them *pushbroom stereo mosaics*.

In real applications, there are two challenging issues. The first problem is that the camera usually cannot be controlled with ideal 1D translation and camera poses are unknown; therefore, camera orientation estimation is needed. External orientation instruments, i.e., GPS and INS, can be used to ease the problem of camera orientation estimation [10, 16]. More general approaches using bundle adjustment techniques [14] are under investigation for efficiently estimating camera poses of long image sequences. In this paper, we assume that the extrinsic and intrinsic camera parameters are known at each camera location. The second problem is to generate dense parallel mosaics with a sparse, uneven, video sequence, under a more general motion, and for a complicated 3D scene. To solve this problem, we apply the parallel ray interpolation for stereo mosaics (PRISM) approach [10] for generating multiple pushbroom mosaics under constrained 6 DOF motion.

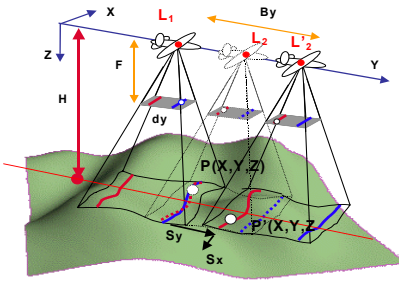

Fig. 1. Dynamic pushbroom stereo mosaics

Dynamic pushbroom stereo mosaics [11] can be generated in the same way. Fig. 1 illustrates the geometry. A 3D point P(X,Y,Z) on a target is first seen through the leading edge of an image frame when the camera is at location L_1. If the point P is static, we can expect to see it through the trailing edge of an image frame when the camera is at location L_2. The distance between leading and trailing edges is d_y (pixels), which denotes the constant "disparity" between this pair of images. However, if the point P moves during that time, the camera needs to be at a different location L'_2 to see this moving point through its trailing edge. For simplifying equations, we assume that the motion of the moving point between two observations (L_1 and L'_2) is a 2D motion (S_x, S_y), which indicates that the depth of the point does not change over that period of time. Therefore, the depth of the moving point can be calculated as

$$Z = F\frac{B_y - S_y}{d_y} \tag{1}$$

where F is the focal length of the camera and B_y is the distance of the two camera locations (in the y direction). Mapping this relation into stereo mosaics following the notation in [10], we have

$$Z = H(\frac{d_y + \Delta y - s_y}{d_y}) \tag{2}$$

and

$$(S_x, S_y) = (Z\frac{s_x}{F}, H\frac{s_y}{F}) = (Z\frac{\Delta x}{F}, H\frac{s_y}{F}) \tag{3}$$

where H is the depth of plane on which we want to align our stereo mosaics, $(\Delta x, \Delta y)$ is the visual motion of the moving 3D point P, which can be measured in the stereo mosaics. The vector (s_x, s_y) is the target motion represented in stereo mosaics. Obviously, we have $s_x = \Delta x$ in the case of a moving camera with 1D translational motion.

We have the following interesting observations about the dynamic pushbroom stereo geometry for 3D and moving target extraction.

(1) *Stereo fixation*. For a static point (i.e. $S_x = S_y = 0$), the visual motion of the point with a depth H is (0,0), indicating that the stereo mosaics thus generated fixate on the plane of depth H. This fixation facilitates the stereo matching and the detection of moving targets on that plane.

(2) *Motion accumulation*. For a moving point ($S_x \neq 0$ and/or $S_y \neq 0$), the motion between two observations accumulates over a period of time due to the large distance between the leading and trailing edges in creating the stereo mosaics. This will increase the discrimination capability for slowly moving objects viewed from a relatively fast moving aerial camera.

(3) *Epipolar constraints*. In the ideal case of 1D translation of the camera, the correspondences of static points are along horizontal epipolar lines, i.e., $\Delta x = 0$. Therefore, for a moving target P, the visual motion with nonzero Δx implies that the motion of the target in the x direction is not zero (i.e., $S_x \neq 0$). In other words, the correspondence pair of such a point will violate the epipolar line constraint for static points (i.e. $\Delta x = 0$).

(4) *3D constraints*. Even if the motion of the target happens to be in the direction of the camera's motion, we can still discriminate the moving target by examining 3D anomalies. Typically, a moving target (a vehicle or a human) moves on a flat ground surface (i.e., road) over the time period during which it is observed through the leading and trailing edges of video images with a limited field of view. We can usually assume that the moving target share the same depth as its surroundings, given that the distance of the camera from the ground is much larger than the height of the target. A moving target in the direction of camera movement, when treated as a static target, will show 3D anomaly - either hanging up above the road (when it moves to the opposite direction, i.e., $S_y < 0$), or hiding below the road (when it moves in the same direction, i.e., $S_y > 0$).

3. Multi-View Pushbroom Mosaics

A pair of stereo mosaics is a very efficient representation for both 3D structures and target movements. However, there are two remaining issues. First, stereo matching will be difficult due to the largely separated parallel views of the stereo pair. Second, for some unusual target movements, e.g. moving too fast, changing speed or direction, we may either have two rather different images in the two mosaics (if changing speed), or see the object only once (if changing direction), or never see the object (if it maintains the same speed as the camera and thus never shows up in the second edge window).

Therefore we propose to generate multi-view mosaics (more than 2), each of them with a set of parallel rays whose viewing direction is between the leading and the trailing edges (Fig. 2). The multiple mosaic representation is still efficient. Moreover, there are three benefits of using them. First, it eases the stereo correspondence problem in the same way as the multi-baseline stereo [17], particularly for more accurate 3D estimation and occlusion handling. Second, multiple pushbroom mosaics can be used for image-based rendering with stereo viewing in which the translation across the area is simply a shift of a pair of mosaics, and the change of viewing directions is simply a switch between two consecutive pairs of mosaics. Third, multiple mosaics can also facilitate 3D estimation of moving targets, and increase the possibility to detect moving targets with unusual movements and also to distinguish the movements of the specified targets (e.g., ground vehicles) from those of trees or flags in wind. In the next section, we will discuss a new method to extract both of 3D structures and moving targets from multiple dynamic pushbroom mosaics.

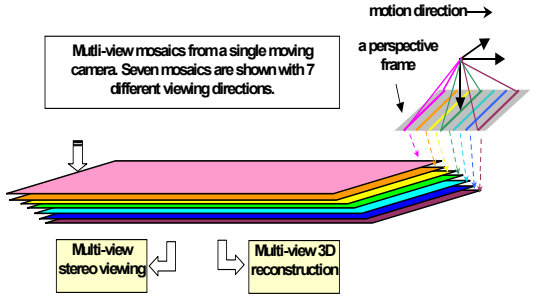

Fig. 2. Multi-view pushbroom mosaics

4. 3D and Motion Content Extraction

Based on the observations in Section 2, our previous work of segmentation-based stereo matching [11] integrates the estimation of 3D structure of an urban scene and the extraction of independent moving objects from a pair of dynamic pushbroom stereo mosaics. In this paper, we use the advantageous properties of multi-view mosaics, and propose a multi-view approach to perform both stereo matching and motion detection. In a set of pushbroom mosaics generated from a video sequence, the leftmost mosaic is used as the reference mosaic, therefore color segmentation is performed on this mosaic, and the so called *natural matching primitives* are extracted. Multiple natural matching primitives are defined with each homogeneous color image patch approximately corresponding to a planar patch in 3D. The patch and primitive representations are effective for both static and moving targets in man-made urban scenes with objects of largely textureless regions and sharp depth boundaries. Then matches of those patches with natural matching primitives are searched in the rest of the mosaics, one by one. After matching each stereo pair, a plane is fitted for each patch, and a set of planar parameters for the planar patch is estimated. Then multi-view matches are performed to obtain accurate and robust 3D results. In the following we will detail this multi-view approach.

4.1. Patch and interest point extraction

First, the reference mosaic of the stereo mosaic pair, i.e. the leftmost mosaic, is segmented, using the mean-shift-based approach proposed by Comanicu & Meer [12]. The segmented image consists of image regions (*patches*) with homogeneous color, and each of them is assumed to be a planar region in 3D space. All the neighboring patches of each patch are also recorded. For each patch, the boundary is defined as a closed curve. Then we use a line fitting approach to extract feature points for stereo matching. The boundary of each patch is first fitted with connected straight-line segments using an iterative curve splitting method. The connecting points between line segments are defined as *interest points*, around which the natural matching primitives are going to be defined.

4.2. Three-step stereo match

A three-step matching algorithm is designed and applied to the first stereo mosaic pair. Let the leftmost (reference) mosaic and the second (target) mosaic be denoted as I_1 and I_2, respectively. The matching process consists of the following three steps.

Step 1: Global match. For a frontal or near-frontal surface, all the pixels inside the patch (region) have similar visual displacements. Therefore, for each region in the mosaic I_1, the sum of absolute difference (SAD) is carried out for all pixels in this region between the two mosaics I_1 and I_2 with a 2D translation. Thus the initial visual displacement vector ($\Delta x, \Delta y$) of the region between I_1 and I_2 is obtained as the one minimizing the SAD.

Step 2: Local match. Since not all regions are frontal planes in 3D space, the pixels in each region do not have a fixed visual displacement. Thus, for each interest point of a patch, the best match is searched within a neighborhood area of the patch's initial visual displacement. Instead of using the conventional window-based match, we define the so-called *natural matching primitives* to conduct a sub-pixel stereo match. We define a region mask M of size mxm centered at that interest point such that

$$M(i,j) = \begin{cases} 1, \text{ if } (x+i, y+j) \in \mathbf{R} \\ 0, \text{ otherwise} \end{cases} \quad (4)$$

The size m of the natural window is adaptively changed depending on the size of the region R. In order that a few more pixels (1~2) around the region boundary are also included so that we have sufficient image features to match, a dilation operation is applied to the mask M. The weighted cross-correlation, based on the *natural window* centered at the point (x,y) in the reference mosaic, is defined as

$$C(\Delta x, \Delta y) = \frac{\sum_{i,j} M(i,j) I_1(x+i, y+j) I_2(x+i+\Delta x, y+j+\Delta y)}{\sum_{i,j} M(i,j)} \quad (5)$$

Note that we still carry out correlation between two color images but only on those interest points on each region boundary, and only with those pixels within the region and on the boundaries. A sub-pixel search is performed in order to improve the accuracy of 3D reconstruction. A match is marked as reliable if it passes the crosscheck proposed in Scharstein & Szeliski (2002) [18].

Step 3: Surface fitting. Since each homogeneous color region is assumed to be planar in 3D, a plane $aX+bY+cZ=d$, is fitted to each region after obtaining the 3D coordinates of the interest points of the region using the pushbroom stereo geometry (Eq. 2). We use a RANSAC method [19] to perform plane fitting. At each iteration, three interest points that are not co-linear are randomly selected to fit a plane, then the result is evaluated by warping all the interest points with reliable matches to the target mosaic. For each reliable matched interest point, if its warped location in the target mosaic is close to that determined by the local match, i.e. their distance is smaller than 1 pixel, then it is called a *reliable and accurate match point* (RAP). If the number of the RAPs is large enough (e.g., the ratio between the number of RAPs and that of all the reliable matched interest points is larger than or equal to 0.65), then the fitted plane is considered a good estimate.

4.3. Plane parameters from multiple mosaics

After the above three steps are applied to the first pair of stereo mosaics, initial estimations of the 3D structure of all the patches (regions) in the reference mosaic are obtained. Further, matches between the reference mosaic and each of the rest of the mosaics are then conducted. However the global match (Step 1 in the section 4.2) is not applied; instead, the initial visual displacement of each interest point on a patch is predicted from the result of this point estimated from the first stereo pair. From Eq. (2), we know the visual displacement Δy is proportional to the selected "disparity" (d_y) for a pair of stereo mosaics. Therefore the visual displacement of the interest point in consideration can be predicted except that the point is on a moving object that (1) does not move along the epipolar line of the pushbroom stereo; or (2) moves along the epipolar line, but with a varying speed. Those points will be reconsidered in the region merging and moving target detection stages. For refining the initial estimates of visual displacements based on the predictions from the results of the first pair of stereo mosaics, Steps 2 and 3 in Section 4.2 are performed to obtain new plane parameters for each pair of stereo mosaics starting from the second pair.

Suppose there are N pairs of stereo mosaics, constructed from $N+1$ pushbroom mosaics. Then N sets of plane parameters (a_k, b_k, c_k, d_k), $k=1,\ldots,N$, are obtained for each region (patch) in the reference mosaic. In order to obtain the most accurate plane parameters for each planar patch, the following steps are performed. First, for each pair of stereo mosaics, the patches in the reference mosaic are

warped to the target mosaic in order to compute a color sum of absolute differences (SAD) for each region, between warped and original target images. Then, among all the estimates for each patch, the set of plane parameters with the least SAD value is selected as the best plane estimate. Note that in the current implementation, only the patches visible in the leftmost (i.e., the reference) mosaics are considered.

4.4. Plane merging

After the plane parameters with the smallest SAD value have been obtained for each region, we will have a close look at the best SAD of each region. If the SAD value is less than a preset threshold, then the patch is marked as *reliable*. We have found that a large number of small regions around a large region corresponding to a surface (or part) of a 3D object are generated by color segmentation, and they are difficult to obtain accurate 3D estimates because of the lack of sufficient sizes and/or feature points. Therefore, we perform a modified version of the neighboring plane parameter hypothesis approach proposed by Tao et al [20] to infer better plane estimates. The main modification is that the parameters of a neighboring region are adopted *only if* it is marked reliable and the best neighboring plane parameters are accepted only when the match evaluation cost using the parameters is less than a threshold. The neighboring regions sharing the same plane parameter are then merged into one reliable region. This step is performed recursively till no more merges occur.

4.5. Moving object detection

After the plane merging stage, most of the small regions are merged together and marked as *reliable*. Moving object patches that move along epipolar lines should obtain reliable matches after the plane merging step, but they appear to be "floating" in air on below the surrounding ground, with depth discontinuities all around it. In other words, they can be identified by checking their 3D *anomalies*.

The remaining regions with *unreliable* marks fall into the following two categories: (1) moving objects with motion not obeying the pushbroom epipolar geometry; (2) occluded or partially occluded regions (usually those with dramatically different views across the multi-view mosaics), or regions with large illumination changes. For regions in the second category, their SADs in stereo matching evaluation are always very high. The regions in the first category correspond to those moving objects that do not move in the direction of camera motion, therefore they do not obey the pushbroom stereo epipolar geometry. Therefore, we perform a 2D-range search within its neighborhood area, and a global match step similar to the first step of normal stereo matching is carried out for each such patch. If a good match (i.e., with a small SAD) is found within the 2D search range, then the region is marked as a *moving* object.

5. CB3M: Content-Based 3D Mosaics

A content-based 3D mosaic (CB3M) representation is a set of video object (VO) primitives (patches) that are defined as

$$\mathbf{CB3M} = \{VO_i, i=1, \ldots, N\} = \{ (c_i, \mathbf{b}_i, \mathbf{n}_i, \mathbf{m}_i), i=1, \ldots, N\} \quad (6)$$

where (1) N is the number of VOs, i.e., "homogeneous" color patches (regions) before region merging; (2) c_i is the color (3 bytes) of the *ith* region; (3) $\mathbf{b_i}$ is the 2D boundary of the *ith* region in the left mosaic, chain-coded as $\mathbf{b_i} = \{(x_0, y_0), K_i, b1, b2, \ldots b_{Ki}\}$, where the starting point (x_0,y_0) has 4 bytes, and each chain code has 3 bits. K_i is the number of boundary points (which needs 4 bytes each) and $K = \sum K_i$ is the total for all regions; (4) $\mathbf{n_i} = (n_x, n_y, n_z, d)$ represents the plane parameters of the region in 3D, 4 bytes for each parameter; and (5) $\mathbf{m_i}$ represents the L motion parameters of the region if in motion (e.g. L =2 for 2D translation on the ground). Therefore the total data amount is

$$\begin{aligned} & N_{color} + N_{boundaryr} + N_{structure} + N_{motion} \\ & = 3N + (8N+3K/8) + 4*4N+4L*N_m \\ & = 27N+3K/8+4LN_m \text{ (bytes)} \end{aligned} \quad (7)$$

when each of the motion and structure parameters needs 4 bytes. In the above equation, N_m is the number of moving regions (which is much smaller than the total region number N). Note that the VO primitives are those patches before region merging in order to preserve the color information. However, the plane parameters are obtained after multiple regions with different colors but on a same plane *surface* are merged.

The proposed CB3M representations are highly compressed visual representations for very long video sequences of dynamic 3D scenes. The representations could fit into the MPEG-4 standard [13], in which a scene is described as a composition of several Video Objects (VOs), encoded separately.

6. Experimental Results

6.1. Results and analysis on a simulated scene

Nine parallel-perspective stereo mosaics were generated from a simulated video sequence of a simulated scene with ground truth data of both 3D and moving targets (Fig. 3). The sequence was generated using the following parameters. The virtual "aircraft" with a video camera flew at a 300-meter height above the scene along a 1D translational direction, and the motion direction is perpendicular to the optical axis of the camera. The focal length of the camera is 3000 pixels (as in Eqs. 1 and 3), and the camera moves with a constant speed. The 3D "buildings" are with heights from 5 to 120 meters above the ground, with different roof shapes (rectangular, round, frontal, ridged, slanting, and/or with small attachments). There are occlusions between buildings. Each of the eight moving objects has a height from 2 to 5 meters, and undertakes a 2D translational motion with

constant velocity during the period of the capture of the total 1640 frames of images. The velocity of the motion of each moving target is represented in centimeter (cm) per frame. Nine 1-column width slit windows are used to generate the nine mosaics (refer to Fig. 2), every pair of the two consecutive windows has a 40-pixel distance, and hence the total distance between the first and the last windows is 320 pixels. Fig. 3 shows three of the nine mosaics, (a) the leftmost, (b) the center, and (c) the rightmost views. Varying occlusions/visibilities can be seen in these mosaics.

From the nine mosaics, we use the leftmost mosaic as the reference image to match with the other eight mosaics. For each region in the reference mosaic, there are 8-plane estimation results, and the best estimate is selected for the 3D parametric representation of the region. The final "height" map (Fig. 3f) is rendered as a map of heights of objects from the ground, i.e. $-H\Delta y/d_y$, (normalized to a range from 0 to 255 for display). For comparison, we have also generated a height map (Fig. 3e) from the stereo matching results of only the first and the second mosaics (without region merging). It can be seen that by using the best parameter selections from multi-view mosaics and utilizing the plane merging step, finer 3D results are obtained for many building roofs, and more accurate results are obtained for sides of buildings. The numbers of regions before and after merging are 1,342 and 350, respectively. We also compare the final estimated height map with the ground truth data. The error histogram is generated for all the regions (excluding the moving targets); from the error distribution, we have found that the errors of 88.8% points in the reference mosaic are within ±4 meters. The absolute average value of the errors for those points is only 0.312 meters.

After the regions have been merged, we analyze all the reliable regions, and those with obvious 3D anomalies are marked as moving objects (along the epipolar lines). On the other hand, those unreliable regions (as possible candidates for moving objects not along the epipolar lines) further go through 2D-range searches for matches within their neighborhood areas (e.g., 30x30 2D range). The estimated motion parameters (s_x,s_y) (in pixels) of those detected moving targets from the first pair of stereo mosaics are marked on the CB3M representation in Fig. 3d. An error analysis of the 8 detected moving targets shows that the average error of the 2D motion estimation is (0.198, 0.008) in velocity (cm/frame), or (0.791, 0.033) in displacements (pixels) between the first pair of the stereo mosaics.

The compression of a video sequence comes from two steps: stereo mosaicing and then content extraction. For the simulated image sequence, we have 1640 frames of 640*480 color images, so the data amount is 1441 MB. The size of pair of the stereo mosaics is 1320*640*2, which has 4.83MB (without compression). The two mosaics in high-quality JPEG format only have 2*75 KB; therefore, a compression ratio of about 9,837 is achieved for the stereo mosaics (the first step). If all the nine mosaics are saved for mosaic-based rendering (Section 3), the data amount will be 9*75KB hence the compression ratio is about 2,186.

Then after color segmentation, 3D planar fitting and motion estimation, we obtained the CB3M representation (Fig. 3d) of the simulated video sequence, with the total number of the "homogeneous" color regions N = 1,342 and the total number of boundary points K = 119,477. The total amount of data in its CB3M representation is 80.8 KB (with a header). This real file size is consistent with the estimation of data amount using Eq. (7), which is about 79.2 KB. The CB3M data is further reduced to 19.4 KB with a simple lossless Winzip; therefore, the compression ratio is about **76,061**. Note that the compression ratio depends on how fine the color segmentation is. In the example shown in Fig. 3d, the main visual features of the scene are coded. More importantly, the CB3M representation has object contents which can be used for object indexing, retrieval and image-based rendering. The plane parameters (a,b,c,d) for the several representative regions are shown on the CB3M map in Fig. 3d (from left to right: one side of a ridged roof, a slanting roof, ground with depth Z= 300.0m, roof of a low building with Z = 289.0m, and side and roof of a tall building with Z=180.0 m).

6.2. Results on real video data

We also have performed experiments on a real video sequence when the airplane was about 300 meters above the ground. Nine mosaics were generated from the aerial video captured. Fig. 4a shows a pair of stereo mosaics from the nine mosaics, and a close-up window marked in the stereo mosaics, which includes both various 3D structures and moving objects (vehicles). Fig. 4b is a "height" map of that window generated using the proposed method. Note the sharp depth boundaries are obtained for the buildings with different heights and various roof shapes. The moving objects that have been detected across all the nine mosaics are shown by their boundaries (in red). The CB3M mosaic (of the window) is shown in Fig. 4c, with a color, a boundary, plane parameters and a motion vector (if in motion) for each patch (region).

Again we examine the compression of the real video sequence. For the real image sequence, we have 1000 frames of 640*480 color images, so the data amount is 879 MB. The size of pair of the stereo mosaics (Fig. 4a) is 4448*1616*2, which has 41MB. The two mosaics in high-quality JPEG format only have 2*560 KB; therefore, a compression ratio of about 800 is achieved for the stereo mosaics. If nine mosaics are all saved for mosaic-based rendering, then the data amount is 9*560KB so the compression ratio will be 179.

Then after color segmentation, 3D planar fitting and motion estimation, we obtained the CB3M representation of the video sequence, with the total number of the natural regions N = 6,112 and the total number of boundary points K = 420,445. The total amount of data in its CB3M representation is 316 KB. This real file size is consistent with the estimation using Eq. (7), about 315 KB. The CB3M data is further reduced to 90 KB with a simple lossless Winzip; therefore, the compression ratio is about **10,001**. Note that the CB3M representation in Fig. 4c consists of regions corresponding to rather large object surfaces in order to rapidly obtain robust 3D structures. For preserving fine details, we have tried to over-segment the reference mosaic into smaller patches; in that case the compression ratio will still be over 2000.

7. Concluding Remarks

In this paper we propose to construct a content-based 3D mosaic representation (CB3M) for long video sequences of 3D and dynamic scenes captured by a camera on a mobile platform. In real applications, the motion of the camera should have a dominant direction of motion (as on an airplane or ground vehicle), but 6 DOF motion is allowed. Multiple parallel-perspective (pushbroom) mosaics are generated to capture both the 3D and dynamic aspects of the scene under the camera coverage. The multi-view, segmentation-based stereo matching approach is able to extract parametric representations of the color, structure and motion of the dynamic and/or 3D objects, and to represent them as planar surface patches.

The content-based 3D mosaic (CB3M) representation is a highly compressed visual representation for very long video sequences of dynamic 3D scenes. The compression of a video sequence comes from both steps: stereo mosaicing and then content extraction. For both simulated and real image sequences of large-scale cultural scenes with many man-made buildings and vegetations, compression ratios of thousands to ten thousands are achieved. More importantly, the CB3M representation has object contents represented.

In the CB3M representation presented in this paper, however, many details and practical issues have not been considered. First, more experiments are needed with both simulated and real video sequences to evaluate the coding and compression capabilities of this representation. Second, in order to use the CB3M representations for real applications, further enhancements are also needed. For example, in the current implementation, only 3D parametric information of planar patches in a single reference mosaic is obtained. Since different visibilities are shown in mosaics with different viewing directions, we want to extend the approach presented in the paper to produce multiple depth maps with multiple reference mosaics and then integrate the results by performing occlusion analysis. Finally, developing higher-level representations that group the lower-level natural patches for physical objects may also be very useful for many applications. For example, the neighboring regions and merged surfaces, which have been extracted in the patch extraction and region merging stages, have not been represented in the current model.

8. Acknowledgments

This work is supported by AFRL under Grant Award No. FA8650-05-1-1853, by NSF under Grant No. CNS-0551598, by ARO under Grant No. W911NF-05-1-0011 and by New York Institute for Advanced Studies.

9. References

[1]. M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu, Mosaic representations of video sequences and their applications. Signal Processing: Image Communication, vol. 8, no. 4, May 1996.

[2]. R. Szeliski. Image mosaicing for tele-reality applications. In WACV'94: 44-53.

[3]. F. Odone, A. Fusiello and E. Trucco, Robust motion segmentation for content-based video coding, the 6th Conference on Content-based Multimedia Information Access, College de France, 2000: 594-601.

[4]. W. H. Leung and T. Chen, Compression with mosaic prediction for image-based rendering applications, IEEE Intl. Conf. Multimedia & Expo., New York, July 2000.

[5]. Y. Li, H.-Y. Shum, C.-K. Tang, R. Szeliski, Stereo reconstruction from multiperspective panoramas. IEEE Trans. on PAMI, 26(1), 2004: pp 45-62.

[6]. C. Sun and S. Peleg, Fast Panoramic Stereo matching using cylindrical maximum surfaces, IEEE Trans. SMC Part B, 34, Feb. 2004: 760-765.

[7]. J. Xiao and M. Shah, Motion layer extraction in the presence of occlusion using graph cut, In Proc. CVPR'04.

[8]. Y. Zhou and H. Tao, A background layer model for object tracking through occlusion," In Proc. ICCV'03: 1079-1085.

[9]. Q. Ke and T. Kanade, A subspace approach to layer extraction, In Proc. IEEE CVPR'01.

[10]. Z. Zhu, E.. Riseman, A. Hanson, Generalized parallel-perspective stereo mosaics from airborne videos, IEEE Trans. PAMI, 26(2), Feb 2004, 226-237.

[11]. [left blank for blind review].

[12]. D. Comanicu and P. Meer, Mean shift: a robust approach toward feature space analysis. PAMI, May 2002.

[13]. R. Koenen, F. Pereira and L. Chiariglione, MPEG-4: context and objectives. Signal Processing: Image Communications, 9(4), 1997:295-300.

[14]. B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon, Bundle adjustment - a modern synthesis, In Vision Algorithms: Theory and Practice, Lecture Notes in Computer Science, vol 1883, pp 298--372, 2000.

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

[16]. Z. Zhu, E. M. Riseman, A. R. Hanson and H. Schultz, An efficient method for geo-referenced video mosaicing for environmental monitoring. MVA, 16(4), 2005, 203-126

[17]. M. Okutomi and T. Kanade, 1993. A multiple-baseline stereo," IEEE Trans. PAMI, vol. 15, no. 4, pp. 353-363.

[18]. D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV 47(1/2/3):7-42, April-June 2002.

[19]. G. Medioni, S. Kang, Emerging Topics in Computer Vision. Prentice Hall, ISBN: 0131013661, 2004.

[20]. H. Tao, H. S. Sawhney and R. Kumar, 2001. A global matching framework for stereo computation, In Proc. ICCV'01.

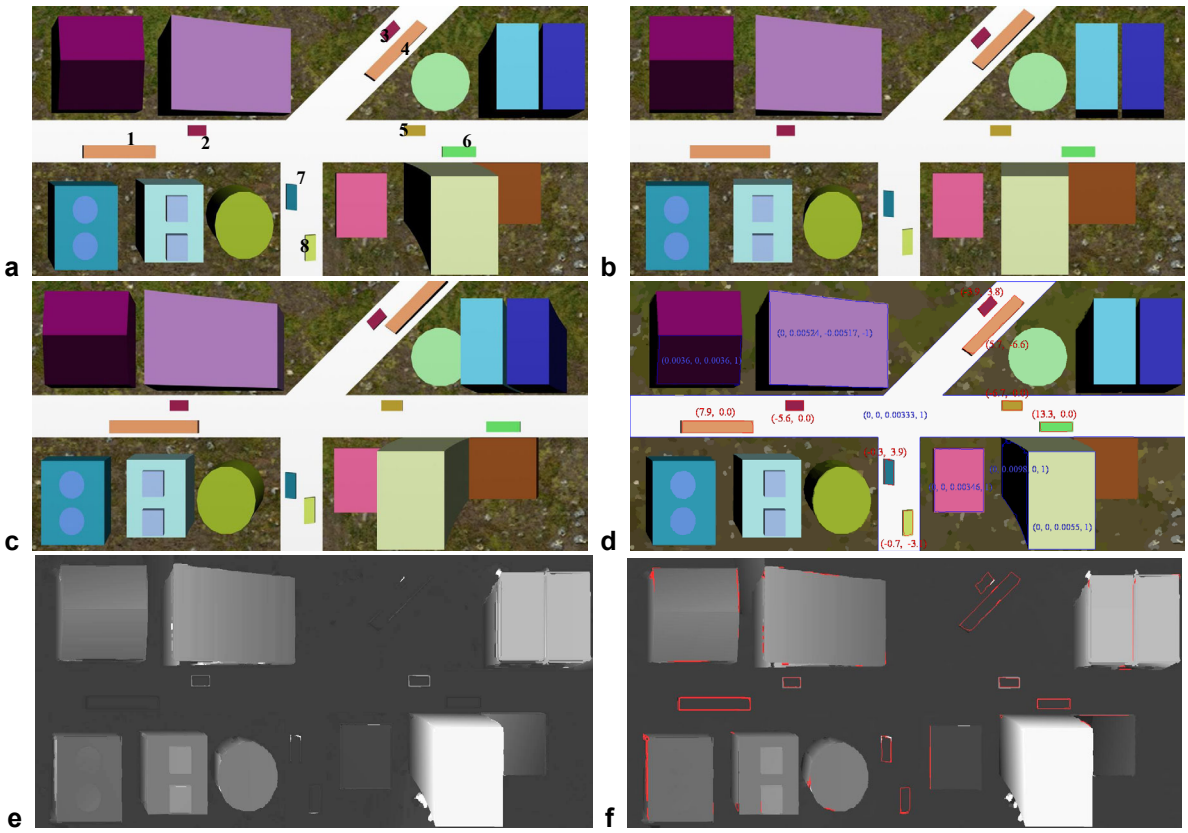

Fig 3. (a) The leftmost, (b) center and (c) rightmost views of the nine mosaics of a simulated scene. The CB3M representation is shown in (d), rendered by average region colors. Some plane parameters *(a,b,c,d)* (in blue) and motion displacements (s_x, s_y) (in red) are labeled. For comparison, (e) and (f) show the rendered "height" maps of the scene from the 1st stereo pair and from all the mosaics, respectively.

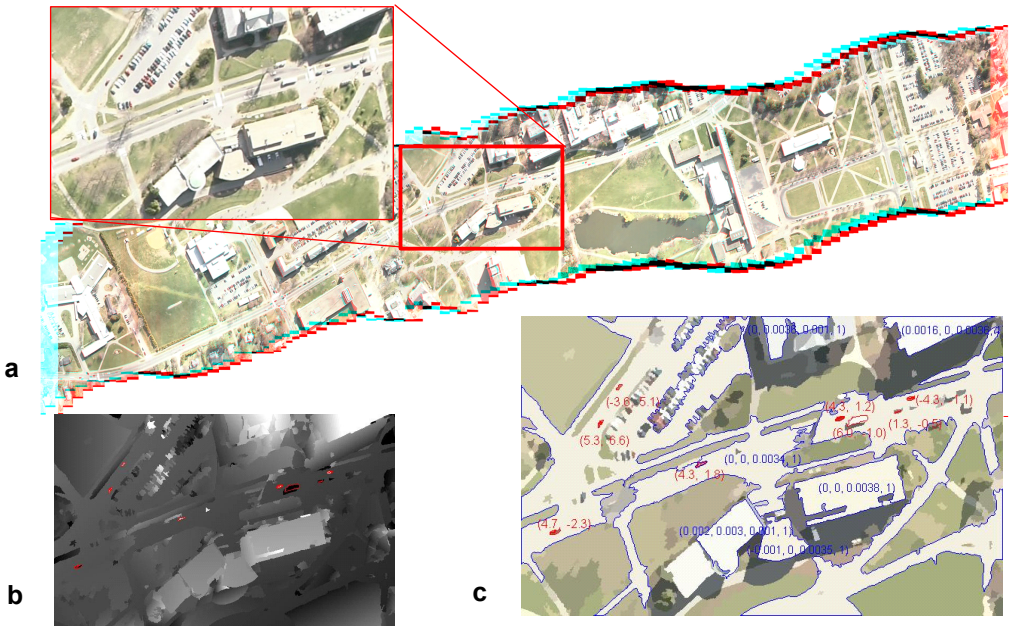

Fig 4. Content-based 3D mosaic representation of an aerial video sequence. (a) A pair of stereo mosaics from the total nine mosaics and a close-up window; (b) the height map of the objects inside that window; (c) the CB3M representation with some of the regions labeled by their boundaries and plane parameters (in blue), and the detected moving targets marked by their boundaries and motion vectors (in red).