MR Video Fusion: Interactive 3D Modeling and Stitching on Wide-baseline Videos

Yi Zhou
State Key Laboratory of Virtual Reality Technology and Systems, Beihang University
Beijing, China
zhouyibuaa@buaa.edu.cn

Mingjun Cao
State Key Laboratory of Virtual Reality Technology and Systems, Beihang University
Beijing, China
cmj@163.com

Jingdi You
Beijing BigView Technology Co., Ltd
Beijing, China
yjd@bigviewcloud.com

Ming Meng
State Key Laboratory of Virtual Reality Technology and Systems, Beihang University
Beijing, China
mengming@buaa.edu.cn

Yuehua Wang
Department of Computer Science, Texas A and M University
Commerce, Texas
yuehua.wang@tamuc.edu

Zhong Zhou
State Key Laboratory of Virtual Reality Technology and Systems, Beihang University
Beijing, China
zz@buaa.edu.cn

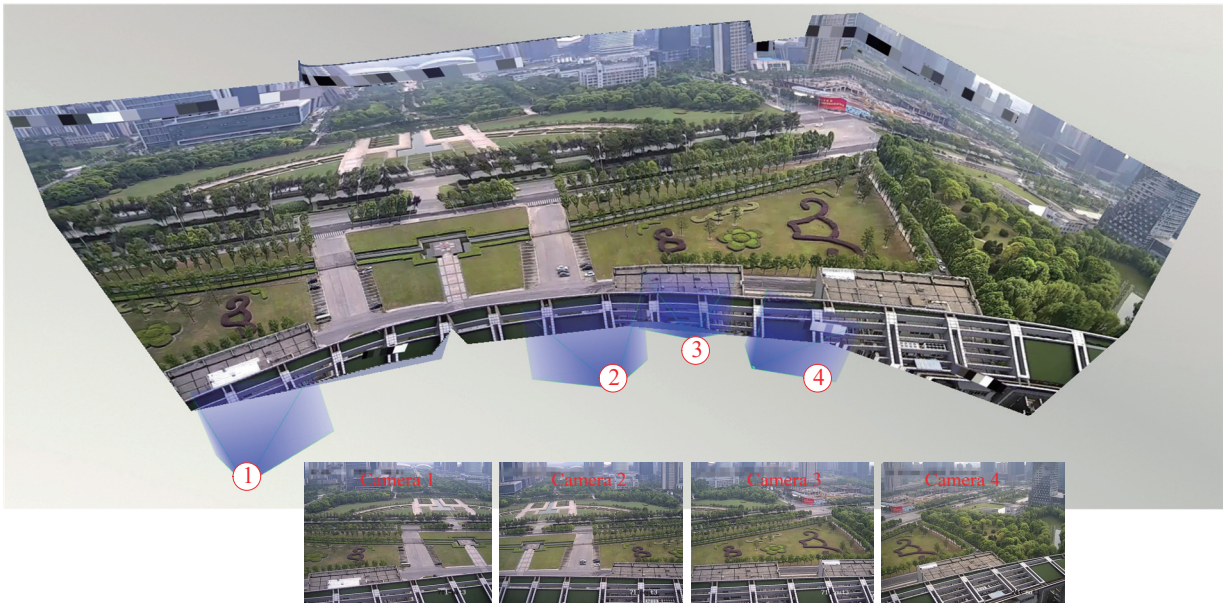


Figure 1: Mixed Reality Video fusion. Bottom: Four input videos with wide baselines, while the blue frustm indicates the cameras' locations and orientations; Top: Our 3D stitching result based on our modeling method.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

VRST '18, November 29-December 1, 2018, Tokyo, Japan

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-6086-9/18/11…$15.00
https://doi.org/10.1145/3281505.3281513

ABSTRACT

A major challenge facing camera networks today is how to effectively organizing and visualizing videos in the presence of complicated network connection and overwhelming and even increasing amount of data. Previous works focus on 2D stitching or dynamic projection to 3D models, such as panorama and Augmented Virtual Environment (AVE), and haven't given an ideal solution. We present a novel method of multiple video fusion in 3D environment, which produces a highly comprehensive imagery and yields a spatio-temporal consistent scene. User initially interact with a newly designed background model named video model to register and stitch videos' background frames offline.The method then

fuses the offline results to render videos in a real time manner. We demonstrate our system on 3 real scenes, each of which contains dozens of wide-baseline videos. The experimental results show that, our 3D modeling interface developed with the our presented model and method can efficiently assist the users to seamlessly integrate videos by comparing to commercial-off-the-shelf software with less operating complexity and more accurate 3D environment. The stitching method proposed by us is much more robust against the position, orientation, attribute differences among videos than the start-of-the-art methods. More importantly, this study sheds light on how to use the 3D techniques to solve 2D problems in realistic and we validate its feasibility.

CCS CONCEPTS

• **Computing methodologies → Mixed / augmented reality**; **Virtual reality**; *Reconstruction*; Camera calibration; Image-based rendering;

KEYWORDS

Video fusion, Video Tourism, Video Popup, Immersive Videos, Augmented Virtual Environment

ACM Reference Format:
Yi Zhou, Mingjun Cao, Jingdi You, Ming Meng, Yuehua Wang, and Zhong Zhou. 2018. MR Video Fusion: Interactive 3D Modeling and Stitching on Wide-baseline Videos. In *24th ACM Symposium on Virtual Reality Software and Technology (VRST '18), November 29-December 1, 2018, Tokyo, Japan.* 11 pages. https://doi.org/10.1145/3281505.3281513

1 INTRODUCTION

Internet of cameras and large-scale monitoring has been one of the most important and rapid growing revolutions in recent years, changing traditional ways of surveillance, traffic monitoring and controlling, home security, as well as crime investigation. In fact, the rapid increase of complicated camera networks and video data is posing unprecedent challenges to organize and visualize video imagery from fixed cameras effectively. Image-based rendering methods have provided a view synthesis technique to fuse videos collected from ordered cameras in the same location and generate a panorama mosaic, however, it still remains unclear how to stitch and navigate them in a single screen given theirs various positions, orientations, and field of view.

Indeed, virtual environment has offered a feasible way for us to integrate videos into a same 3D background. Such as [28, 38], they both conduct a wide range of virtual scenes for videos and project videos into a model surface as dynamic textures. However, their models are either created from LiDAR or from Google Earth, which brings origin errors if they directly use them as a background for 2D-3D registration. Their projection method will cause distorted textures due to un-accurate depth correspondence.

In this paper, we present a fusion system for a large number of videos, which offers highly comprehensive imagery and supports a spatio-temporal consistent scene. With our system, the relationships between cameras, such as relative position, is well explained, and the user do not have cognitive difficulties. The results demonstrate that our system can be used to effectively display complex scenes, such as square, junction, street, and provide a better user experience in both vision and interaction.

Our approach is based on an interactive image-based modeling, which allows users to rapidly draw the main parts of image scene, turn a 2D video into a 3D video model, and register them into a virtual scene. And based on modeling results, our method allows users to manually stitch overlapped planes, and generate a complete video imagery. After modeling and stitching, we provide the user capability to seamlessly browse videos from different virtual locations and smoothly transit from one to another, using our rendering method.

We show a typical video fusion scene in Figure 1 . This scene contains 4 videos whose real viewpoints belong to wide baselines catalog. These videos are well-stitched and rendered in real-times to a 3D scene. Such fusion offers users an immersive view, called "*Mixed Reality Video Fusion*" in this study.

The main contributions of our system are concluded as:

- A robust interactive modeling method for a single uncalibrated image, which fully uses the geometric information to build its 3D structure.
- A novel thought for video registration that converts 2D-3D registration to a 3D-3D registration and solves the unreasonable fusion of direct texture projective mapping.
- A novel 3D stitching method based on cameras' 3D pose and modeling result, and allows users to skim the result through any view.
- An opening platform for video visualization which integrates computer vision, graphics and user interface techniques, and quite easy to integrate video analysis method in the future.

2 RELATED WORK

In this section, we introduce three main categories of work related to ours: multiple video visualization, interactive modeling from multiple images and image/video stitching.

2.1 Video Visualization in Virtual Environments

The requirement of multiple video visualization is raised mainly due to the cognitive burden for users when given a number of video thumbnails or cameras. The majority of methods focus on giving the videos context information to help the user understand. There are two main methods: one is to display synchronized multi-camera recordings alongside an interactive map of the recorded space in order to aid understanding [1, 2, 18, 21, 36], while another one is to project videos onto 3D models or a reference map which creates one single context for all the videos [10, 20, 28, 30, 31]. The authors [31] argued that, if the video is projected from the actual camera location with the correct camera parameters, the walls and floors in the video can seamlessly match the model. Their Video Flashlight system demonstrates texture projection's feasibility and gives an immersive walkthough experience for users. DeCamp et al. [30] apply video projection to fisheye cameras and build an indoor immersive system-HouseFly. Kim et al. [20] extend video projection to augment static aerial earth by designing four particular scenarios. Chen et al. [10] achieve dual-resolution for a video projection system.

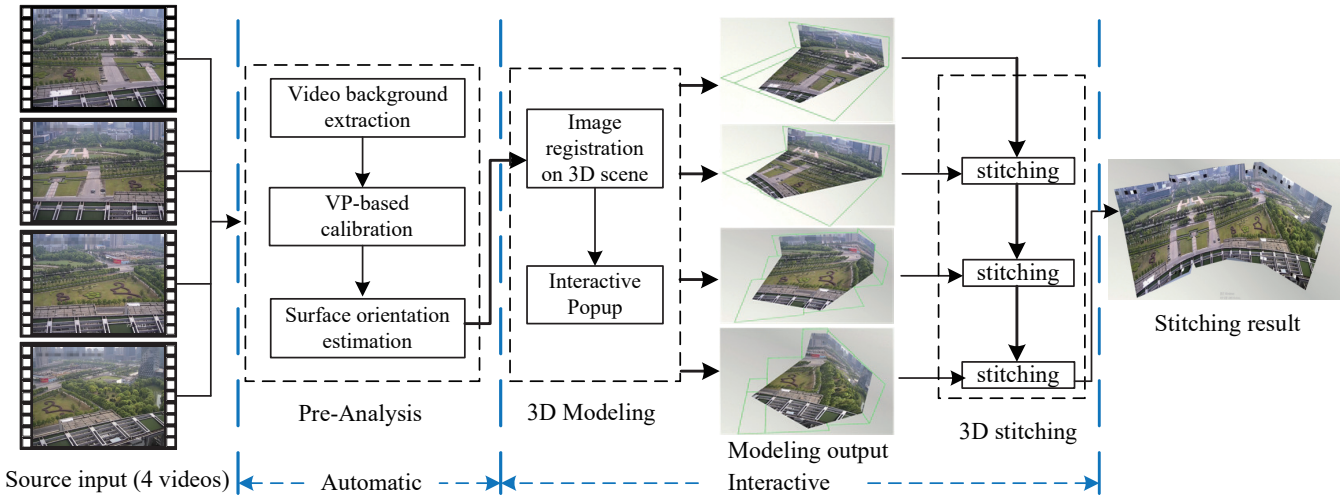


Figure 2: Overview of the proposed system. The entire process consists of two parts, offline and online. The whole process is divided into offline and online, the former includes automatic pre-processing, interactive modeling, and 3D stitching. The latter combines real-time rendering technology to achieve multi-video fusion (Limited by space, there is no illustration here).

However, there are problems for video projection. For example, if the viewpoint is far away from the captured location, severe distortion [26] and image fragmentary will arise due to the missing correspondence between image regions and the 3D model, leading to the result that video projection may not be selected by users in some tasks [41]. On the other hand, the method of video projection commonly needs accurate 3D models to register videos [28, 30]. But for these pre-established model, it is hard to guarantee its precision responding to 2D images and has a negative effect on visualization. We use a post-established method which gets a background model (called *video model*) from video frames. It not only keeps correspondences between models and images, but also reduces severe distortion and image fragmentary through manual modeling operations with an adjustable precision.

2.2 Interactive Modeling from Multiple Images

The problem of 3D modeling from images and videos has been studied for years. Many automatic methods [6, 8, 11, 40] have been proposed to reconstruct a complete scene from a single image. However, either of them has limitations, such as the view of images captured [11] (e.g., street side view), the type of modeling subject [6, 8, 40] (e.g., façade, symmetry object) or other limitations. What's more, these results need post-process to repair, and if we directly use the point cloud or the reconstructed mesh from point cloud as the model, the skew, tear extrusion and other distortions will appear in the models. The complexity is quite high and the way of data representation and visualization is not favorable by users. It turns out that for a single image captured from a common scene, interactive modeling is always the first choice.

For multiple images, the researchers use multi-view stereo (MVS) [12, 17, 32] or structure from motion (SFM) reconstruction [16, 33–35]. These methods extract and match feature points (or pixels) [12, 17, 32] , lines [16, 33, 35] and planes [34] from neighboring images under the narrow baseline condition. Although they can produce accurate, and even photo-realistic models, they also need serious overlapping. For sparsely distributed video capturing, only a low (and even no) overlapping appears, as shown in [20, 28]'s demo videos. It turns to be the main difficulty for multiple video visualization. Aimed at this difficulty, we propose an interactive method to rapidly model single video with its frame and then extend to multiple videos. What we rely on are the robust pre-geometric structure analysis and wide baseline line matching.

Particularly, we notice Sinha's interactive modeling in the field of architecture [35]. Without point clouds from SFM, this method cannot realize its following procedures. We borrow the speed-up idea of snapping from this method, and make our method more convenient for users. The commercial software, SketchUp[1], provides an interactive 3D reconstruction tool from multiple photos. This is similar to the Sinha's method, but it does not use the SFM to create point clouds and use any speed-up method, just manual vanishing point alignment for photo registration to 3D coordinates. This tool should be one reasonable comparison for our method.

2.3 Image/Video Stitching

Image stitching technique mainly focus on wide baseline images from cameras with great position difference, orientation difference or other attribute differences. Recent studies have achieved a good stitched result. These methods can be divided into two main categories. The former one, spatial-varying method, uses spatial-varying multi-model with local parameters instead of basic single-model with global parameters. For example, Lin et al. [24] employed a smooth varying affine model to align images, which works fairly well with moderate parallax. Zaragoza et al. [42] proposed an APAP warping method. This method divided images into

[1] SketchUp, http://www.sketchup.com/.

hundreds of grids, each of which is aligned by smooth varying homography, and combined with bundle adjustment method to eliminate cumulative error between multiple images. The latter local-warping method converts image stitching into energy minimizing by adding constraint terms, and each term keeps an original characteristic of input images. Chang et al. [7] proposed a SPHP warping, which smoothly transforms homography of overlapping region into similarity of non-overlapping regions. Lin et al. [23] proposed an AANAP warping, which combines linearized homography and global similarity to generate nature panorama. Chen et al. [9] proposed a GSP warping, which optimizes naturalness of panorama by combining global similarity and local similarity. Zhang et al. [43] studied street view, and raised a multi-view stitching method tolerating wide baseline.

Compared with image stitching, there are fewer prior works on video stitching. Different approaches have been proposed for different camera settings. For example, earlier researchers aimed at static cameras, He et al. [15] put forward panoramic video stitching in multi-camera surveillance system. While recent works pay more attention to fixed camera arrays, such as Surround 360 system raised by Facebook, and R2, R5 and R7 camera heads used in Google StreetView [3]. To stitch videos captured by these cameras, the pose relationships between cameras can be pre-calibrated to stitch frames globally, followed by some local warping procedures [19, 22, 29] to eliminate small deviations. However, for independently moved cameras, shakiness must be removed, since the relative position between cameras varies every moment. Guo et al. [14] and Su et al. [37] both take video stabilization into consideration to optimize stitching result. Wang al. [39] present a novel method to create bigger selfie video, called BiggerSelfie, combining a selfie video clip and an environment video without relying on specific hardwares. What is more, Nie et al. [27] optimized stitching and stabilization together to generate a unified optimization framework, which achieved state-of-the-art performance.

However, image/video stitching has its inherent limits. First of all, it cannot deal with image sets without a large overlapping region, which is really common in surveillance systems. Secondly, when the parallax is too large, the quality of stitching result is too poor, and cannot keep good visual experience in a 2D space. Last but not least, even though the panorama is stitched up well, the perspective relationships of stuffs in images are dilapidated due to image deformation, which means that we cannot create a well-structured model with this kind of panorama. We propose a novel 3D stitching method by using our modeling results. This method keeps the nature structure of video imagery, and has no limitation of baselines' length.

3 OVERVIEW

In this section, we provide an overview and motivation for the specific features of our system. For better comprehension of these features, we suggest the readers to browse the supplement videos first.

Our work is based on the idea of using camera pose (location, orientation, and field of view), compact video 3D models and 3D scene information to create new interfaces for browsing hundreds of videos concentrated in key regions. Given the camera pose and compact 3D models of videos, we can simply place the "video" into a common 3D environment which contains a dozen of complex 3D scene model. And it allows the user to virtually browse syncretic videos with a free viewport and smoothly transit from one video to another using our interface. The 3D models of videos are compact but effective bond to link the 2D images with 3D scene information, and the 3D scene enhances the comprehension of spatial geometric relationship between different cameras.

The core part of our system is an interactive modeling approach for rapidly constructing and stitching the 3D models of video's background. Using our modeling approach, the user can conveniently draw main parts of image scenes in both 2D and 3D views and observe a textured modeling result in real time. After modeling a video, the user can register the model into 3D scene by simply drawing a line in both views and continue model the parts whose depth cannot be estimated on 2D images. Meanwhile, we provide the ability to extend single-view modeling to multiple videos by a novel 3D stitching method, to gain a more competitive and determinate model of videos. Figure 2 illustrates the pipeline of our video fusion system.

4 PRE-ANALYSIS ON IMAGE GEOMETRIC STRUCTURE

Although our modeling method is interactive, it relies on accurate knowledge of the vanishing point and surface orientation estimated from observed scene. Our system starts by preprocessing the video frames using computer vision techniques for (a) extracting a background frame with little occlusion to features we used(such as lines), (b) estimating an accurate vanishing point from the background frame and (c) subsequently generating a surface orientation. These procedures are necessary steps before modeling, which decide the efficient and performance of modeling results.

Video background extraction. For a video of multiple frames, not all frames is suitable for modeling since moving objects exist in video. These objects occlude the static building and bring in other mistakes, so we need to select a background frame contains as little as dynamic objects for modeling. We use a classic background extraction method ViBe [4], which separates from foreground objects and offers a clean background with little noise. And then we fill the blank by simply performing mean filter across the N neighboring frames (in practice, N is 10). At last, the user may choose one of background frames to the following procedure.

Vanishing point-based calibration. Parallel lines are common in architectural scenes containing man-made structures. Under perspective projection, parallel lines appear to meet at a point in the image called the vanishing point (VP). Vanishing points have been extensively studied along with the geometry of image formation and have been found useful for camera calibration and 3D reconstruction from a single un-calibrated image. The details of the approach for vanishing point estimation and camera pose calculation can be found in Appendix.

Surface orientation estimation. "Orientation of a surface" is defined as the normal orientation of the surface in the world and "pixel orientation" as the orientation of the surface projected to the pixel. To obtain this per-pixel value, we employ the method from [5]. This method depends on accuracy grouping of line segments and we

guarantee this though our vanishing point estimation approach. We do not use this per-pixel orientation for direct modeling, but a guided modeling detailed in section 5.3.

5 INTERACTIVE MODELING FROM SINGLE IMAGE

In this section, we first introduce the scene graph representation for our models. And then we describe the image registration step. Lastly, we present several accelerating strategies for convenient constructing.

5.1 Primitives and Scene Graph Representation

The key difficulty of modeling is how to constrain the 3D positions of primitives (e.g. point, line, faces and circles) in different parts. For better organization, we adopt a scene graph to represent the geometric relation. This graph contains three kinds of primitives: point, line and face, and then subdivided into six classes of basic primitives according to the relationship of starting-ending point and the type of user operation, the generating relationship is shown in Figure 3. The point-to-face operation will create a new connect region if the user starts modeling from an isolated point, while the line-to-face and face-to-extraction will update its corresponding connect region.

To create a new face, the user interactively start to draw a point P_0 on the image. If P_0 is on the model, its 3D position can be calculated through the intersection of the view ray and model. Otherwise, it will give P_0 a random depth value z and create a new connected region respect to a local 3D orthogonal coordinate system. We treat P0 as a reference primitive in this connected regions and the primitives created from P_0 is decided by it. For modeling multiple parts, user can simply point out the contact points or shared lines. An example of connecting two separated-built cuboids is showed in Figure 4.

5.2 Image Registration with 3D Environment

Before modeling, we register the image to the 3D environment, which is one of the most important operation in our modeling tool. The 3D environment mainly refers pre-built models with CADs. And it can reduce to a single base map, for example, a ground picture for outdoor scenes, such as satellite imagery, a floor plan for indoor scenes. Features on these base maps are useful evidences for image registration. However, the captured image from fixed cameras commonly keeps a large angle with based map, and the base map may be texture-lacking map. It is hard to implement a wide-baseline feature matching between the base map and the image. So we use a simple assumption for images that objects in image is vertical-standing and the world's Z direction is parallel to vertical direction of camera coordinate. The 6-DOF registration problem degenerates into a XY-plane alignment question. By assigning two ground lines from each view, we describe the axis alignment method as below.

Considering a pair of 2D line l^i, 3D lines l^w from the image coordinate system of image plane and the world coordinate system in 3D environment, and two pairs of point correspondences (X_1^i, X_2^i) and (X_1^w, X_2^w). From l^i, we can build a local world coordinate system and a corresponding 3D line l^c,in this world[13], where l^c is along X axis and its endpoints is (X_1^c, X_2^c). They satisfy $X_i^c = sMX_i^w$ $(i=1,2)$,

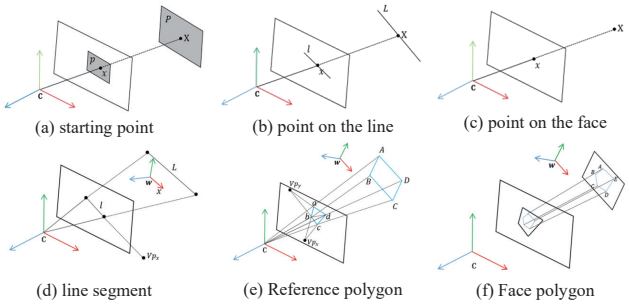


Figure 3: basic primitives and their classification.

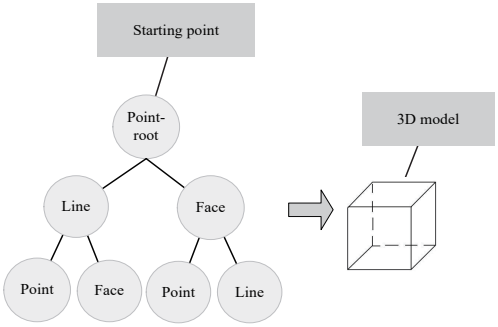


Figure 4: An example of scene graph.

where s is a scalar factor and $\boldsymbol{M}$ is a rigid transformation. We also compute the rotation angle θ from the X axis of 3D environment to l^w. We can simply solve s by setting $s=\|l^c\|/\|l^w\|$, and $\boldsymbol{M}$ by setting $\boldsymbol{M}= [R(l^c,\theta)\, X_1^w]$ where R(l^c,θ) is the rotation matrix along the l^c , and $\|\cdot\|$ denotes the mold of vector. So for all the primitives in video models, if s and $\boldsymbol{M}$ is known, we can transfer them into 3D environment.

Using registration, we can see a stitched imagery which combines the based map with the target image in the 2D image editor view. After registration, the user can start modeling in one window of the 2D image editor and the 3D model editor and simultaneously display the model results in another one. The base map is also a main different part from other traditional modeling tools.

5.3 Modeling Principle and User Interface

Based on the registration, we further to solve the problem of how to unproject 2D points to 3D points by the following modeling principle. Through vanish point analysis[13], we can get the unit axis vector of image space $\boldsymbol{D}_{imageplane}$: $[u\ v\ q]$ and the unit axis vector of camera space $\boldsymbol{D}_{camera}$: $[U\ V\ Q]$.

If we know the relationship $\boldsymbol{\lambda}_{AB}$ between any point B and its reference point A in scene graph, we can calculate B' position from A' position. Given the coordinate $\boldsymbol{X}_A$ of the reference point A in image space and the coordinate $\boldsymbol{X}_A^c$ in camera space, then the coordinate $\boldsymbol{X}_B$ of the point B in image space can be represented by

$\boldsymbol{X}_A$ and $\boldsymbol{D}_{imageplane}$:

$$\boldsymbol{X}_B = \boldsymbol{X}_A + \boldsymbol{\lambda}_{AB}\boldsymbol{D}_{imageplane} = \boldsymbol{X}_A + [\lambda_u\ \lambda_v\ \lambda_q]\begin{bmatrix} u \\ v \\ q \end{bmatrix}, \quad (1)$$

where $\boldsymbol{\lambda}_{AB} = [\lambda_u\ \lambda_v\ \lambda_q]$ is the linear coefficient for $\boldsymbol{D}_{imageplane}$ to represent the relationship AB. Then then the coordinate $\boldsymbol{X}_B^c$ of the point B in camera space can be represented by $\boldsymbol{X}_A^c$ and $\boldsymbol{D}_{camera}$:

$$\boldsymbol{X}_B^c = \boldsymbol{X}_A^c + \boldsymbol{\lambda}_{AB}\boldsymbol{D}_{camera} = \boldsymbol{X}_A^c + [\lambda_u\ \lambda_v\ \lambda_q]\begin{bmatrix} U \\ V \\ Q \end{bmatrix}, \quad (2)$$

Now the 3D world position of any 2D point B in virtual scene is

$$\boldsymbol{X}_B^w = s^{-1}\boldsymbol{M}^{-1}\boldsymbol{X}_B^c = s^{-1}\boldsymbol{M}^{-1}(\boldsymbol{X}_A^c + \boldsymbol{\lambda}_{AB}\boldsymbol{D}_{camera}). \quad (3)$$

We provide two basic modes of operation in the form of user interface, a 2D image editor with a sketch-based drawing interface and a 3D model editor with standard modeling operations for reconstruction.

The 2D image editor allows user to select an input image to sketch over. In this mode, we define several easy-to-understand interfaces for novice users, which has three basic operations including point-to-face, line-to-face and face-to-extraction. The former one is often used to create a new face without connecting to existing model, while the latter two are used to extend existing model rapidly. The user is able to model the main parts of 3D scenes on 2D images with these basic operations.

Our mesh generation is not done after the completion of the whole sketching, but realized after every operation. So we can use a convenient way to conduct a reference 3D view by using projection texture mapping. For 2D image parts with severe foreshortening, it is hard for users to recognize and locate the object from such a long distance. By using our tools, the user can see a relative good-quality patch which has a same unit of length with the axis in orthographic projection.

The 3D model view is not only used for visualization, but also for creating auxiliary primitives and modeling occluded parts, such as a distant building occluded by trees. We have bring a ground map into 3D view for assistance modeling. And the user can modeling, for example, a 3D building through a comparison between 2D image and the ground map. The ground map offers a constraint for the parts on the ground which share no contact cues. In the 3D model view, the capacity of modification is provided for users. So the auxiliary primitives can be created to offer an extra step for more complex modeling. After completing modeling, the auxiliary primitives may be deleted and will do not disturb the texture mapping result.

5.4 Accelerating Strategy

In order to make the drawing process easier and improve the speed of modeling, our user interface provides three forms of snapping: (a) the start point of a reference plane is asked to snap to preexisting plane, (b) the end point of a line is asked to snap to VP directions and (c) face normal is asked to snap to the preprocessed orientation map.

Attachment point snapping. To draw a reference plane always starts from a point on the image plane, and we need to know whether the projection of the start point is supported by a face and which face the point is located on. To get supporting face, we cast a view ray through the start point and intersect with all the preexisting face. The first intersected point is the right face supports the target point. With support analysis, pixels in image plane are simply constrained to the established structure. And this snapping reduce the number of the isolated parts.

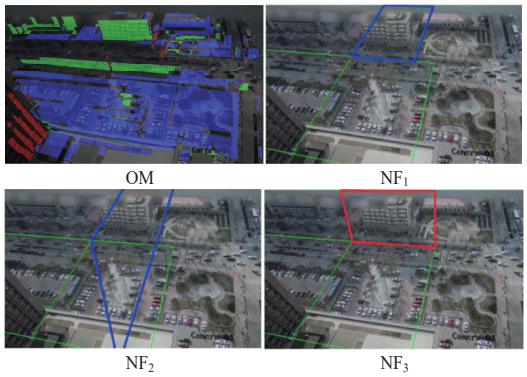


Figure 5: The proportion of positive pixels for the rectangle in the last image(NF_3) is larger than other two possible rectangles(NF_1, NF_2).

Line segments snapping. If any of the line segments drawn by the user almost passes through a VP(<15°), that line segment is snapped to exactly pass through it. Snapped line segments are constrained by the system to be parallel to one of the detected vanishing directions. The VP snapping feature is enabled by default, but can be easily disabled when necessary.

Plane snapping is created when the user draws the reference faces using point-to-face operation. We use the orientation map mentioned in Section 4 to decide the reference face's normal. As showed in Figure 5, three normal possibilities (NF_1, NF_2, NF_3) respectively corresponds to three different rectangles. And two sides of drawn rectangle pass through the other two VP directions except for the direction same as its normal. We statistically accounts for per-pixel orientation in the rectangle and choose the best rectangle i which has largest proportion of positive pixels (NF_p=NF_i), Commonly we have P=(N=NF_i) >30%.

$$P(N = NF_i) = \sum_{pixel \in R(i)\ \&\ N_{pixel}=NF_i} pixel \Big/ \sum_{pixel \in R(i)} pixel \quad (4)$$

The occluded pixels will not be into the statistics. The plane snapping is done during the drawing in real-time and also can be disabled when necessary. When using plane snapping, the user only need to draw the diagonal line. The system automatically decides the plane's orientation and snaps its two sides to VP directions.

6 MULTIPLE IMAGES REGISTRATION AND STITCHING

In addition to model a wider scope of scene, we extend our single image modeling method to multiple images.

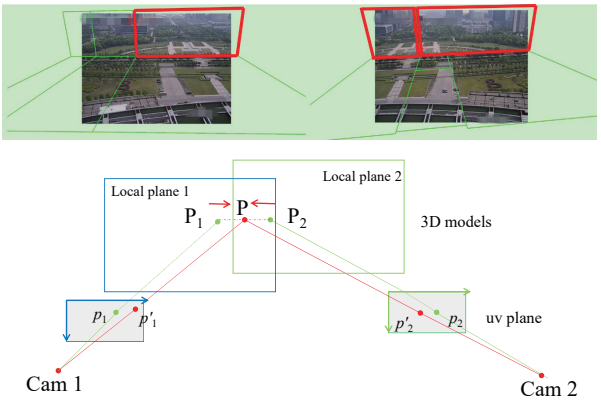


Figure 6: Local-plane-based feature matching.

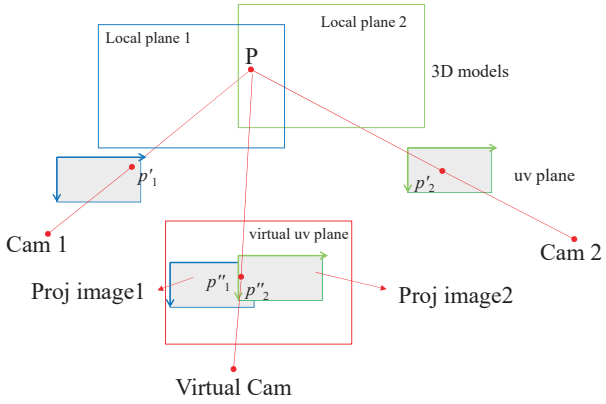


Figure 7: Virtual camera in seam pair generation.

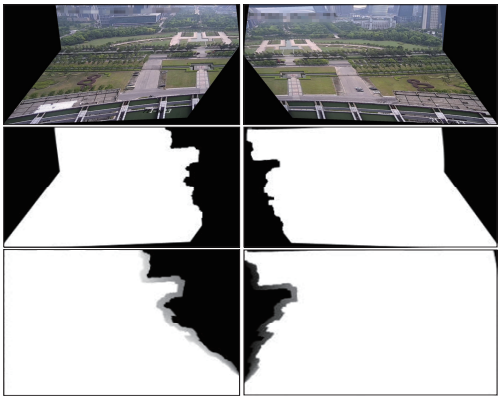


Figure 8: Seam pair generation. First line transformed image. Middle line seams in transformed view, last line seam pair in source image.

6.1 Multiple Images Registration

The user may create the model of first image I as discussed above, and then adds another image I' which has overlapped parts with it. We match the lines between two images [25], and get the line-correspondence in different views. A RANSAC-based method is used to select the optimal line to calculate the rotation and translation, which gives the minimal reprojection error for all the SIFT points located in built image regions. The rotation matrix is calculated by the angle between two optimal matching lines and the translation vector is decided by the matching feature points. After connecting I and I', the user can continue operate the original model on the later-coming images.

6.2 View-Based 3D Stitching

Since our interactive modeling method only builds the simplified structure of input frame, the modeling result still exists difference with real 3D scene, and isn't aligned well in overlapped region. Therefore, we propose a 3D model stitching method based on the 2D image stitching and above image modeling result. We use calibrated camera pose and 2D image matching information to align the neighboring 3D models.

Our method consists of three steps, including local plane feature matching (in 2D space), mesh-based warping (in 3D space) and seam pair generation (in 3D space).

Firstly, we choose several local modeled plane from above modeling result, such as the signed plane with red rectangle in Figure 6. Then we only use these local plane to extract and match features in source 2D image. Due to our non-global matching strategy, the matching error has a perceptible reduction. Finally, the local matches are used to register source images with APAP method [42]. A pre-matched mesh will be generated by this method, and its dense uniform distributed corners will be treated as constrained points for image warping.

Given one constrained point pair (p_1, p_2) in source image, we compute the projection locations (P_1,P_2) on 3D model, and interpolate one final point P with a defined weight w. This final point P is projected back to source images, and generate new point pair(p'_1,p'_2). The (p_1,p'_1) and (p_2,p'_2) compose a pair of control points, and can be used for common 2D mesh alignment. Finally, we use local planes' boundaries, normal and invariant points in 3D models to define warping constrain terms, and use a mesh optimization method to produce the optimized alignment result.

After warping, we need to generate seam for aligned image. A virtual view is chosen as a reference view for assisting seam pair generation, as shown in Figure 7. Then the whole content of two warped images can be projected into the image space of reference view with the camera pose parameters. We solve the optimal seam of reference image through minimizing the alignment error and color error. Finally, the seam of reference image is projected back to source images, which results a pair of source seam. The middle results of seam pair generation are shown in Figure 8.

Please notice that, our method need camera pose parameters and modeling result for constrained point pairs to project and back-project, this is obvious different with previous 2D image or video stitching. Our method is quite simple but work well in real applications.

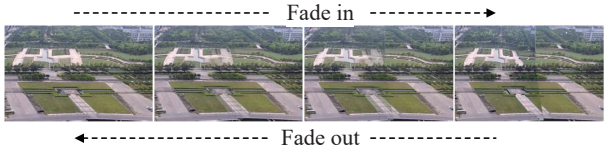


Figure 9: The fade in and fade out performance in Demo video. Left: fade in when stepping in the camera view, middle: stay in camera view, right: fade out when leaving the camera view.

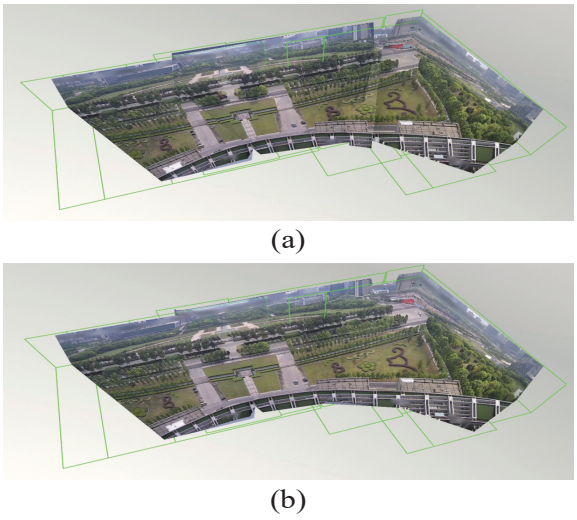


Figure 10: Fusion result comparison for the Square scene. (a) Before stitching (b) After stitching.

7 EXPERIMENTS

Our image modeling technique has been implemented in C++. And the rendering client is done with OSG and GLSL. All the tests are executed on a PC workstation with a NVIDIA GTX 1070 graphics card, 4-core Intel Core(R) I7 7700 at 3.70GHz, 16GB memory, and a 1000Mbps Ethernet connection to the campus network. Our experimental dataset comes from 3 real surveillance system, including two wide scene overlooked from a high building, and a long narrow streetside scene viewed from light poles. In total, 41 fixed HD cameras or virtual videos are used. The baseline of neighboring cameras is extreme wide, and images from those cameras are hard to be modeled by Multi-View Stereo(MVS). The scene used in this section is summarized in Table 1.

Real-time rendering technology. We use projective texture mapping to fusion images and videos into a 3D environment. This approach produces an accuracy and zero distorted texture at the captured view. By combining with shadow mapping rendering techniques, the system supports real-time visibility calculation. On the other hand, the user not only wants to observe stitched or fused videos but complete imagery without culling. So we use an alpha transform strategy as a supplement to render the video. When the viewpoint is close to the camera, the observer can see the whole imagery, and when the user left the view, the stitched result is shown, the standby plane and un-stitched parts will be faded. This kind of visualization performance is shown in Figure 9.

Table 1: Experiment setup

Scene	Video size	Video resolution	Average baseline length	Overlap rate (mean/max/min)
Square	4	1080P	46.2meters	23.7/33/17%
Junction	4	1080P	48.9meters	25.5/48/7%
Street	33	960P	21.6meters	23.0/44/16%

Modeling and stitching performance. Our modeling tool provides a 2D image editor with a sketch-based drawing interface and a 3D model editor with standard modeling operations for reconstruction. The user can choose operations using a button or key shortcut. The system also provides conventional menu selection, view control, texture deformation and other operations. Our stitching operation is also integrated in our modeling tool, and during this operation, the user is only asked to appoint several constraint matching points in both images and invariable points in each image. After warping and seam generation, a seamless 3D model is stitched by above video models. The technique is tested and evaluated on real surveillance videos as we demonstrate in this section. As shown in the accompanying video, most of the examples were modeled and stitched in a few minutes or less.

Photographs themselves often have some distortions from an ideal perspective projection, especially if an object is close to the camera or taken with a wide angle lens. In this case, fisheye correction should be applied before modeling, we currently provide it by integrating a method [25].

We describe the performance figures in detail. Figure 11 shows a fusion result of Junction with 4 videos. In the left column, we show the input frame for modeling and draw our modeling lines on the input frame. In the second column, these frames are separately transformed to a 3D model with a new view. The third column gives a culled model with alpha blending. The rightmost column shows the comparison between our fusion result with 3D stitching and without 3D stitching.

Figure 10 shows the fusion result comparison for Square scene, while its modeling result is shown in overview figure. Note that although the left two image are crudely modeled with just one folding, and the standing planes has depth difference, but our stitching method can still work under this condition and results in a desirable seam. In Figure 12(a), we show a complete rendered fusion view of Street scene, where 33 videos are used to model and stitch. The cameras used for capturing these videos are randomly mounted on one side of the street, which only make sure that they can cover the pedestrian street without any dead corner. Our method models and stitches all the case successfully, and the close-up view of one scene piece is shown is Figure 12(b), while three further close-up views of Figure 12(b) are shown in the rest of sub figures.

Comparison with 2D stitching methods. We compare our 3D stitching method with three 2D stitching method, including APAP [42], GSP [9] , WB [43]. For a fair comparison, we give those compared methods same manual correspondences as our method. One typical result of successful stitched cases for all the three compared methods is shown Figure 13. The GSP method results "ghost" errors,

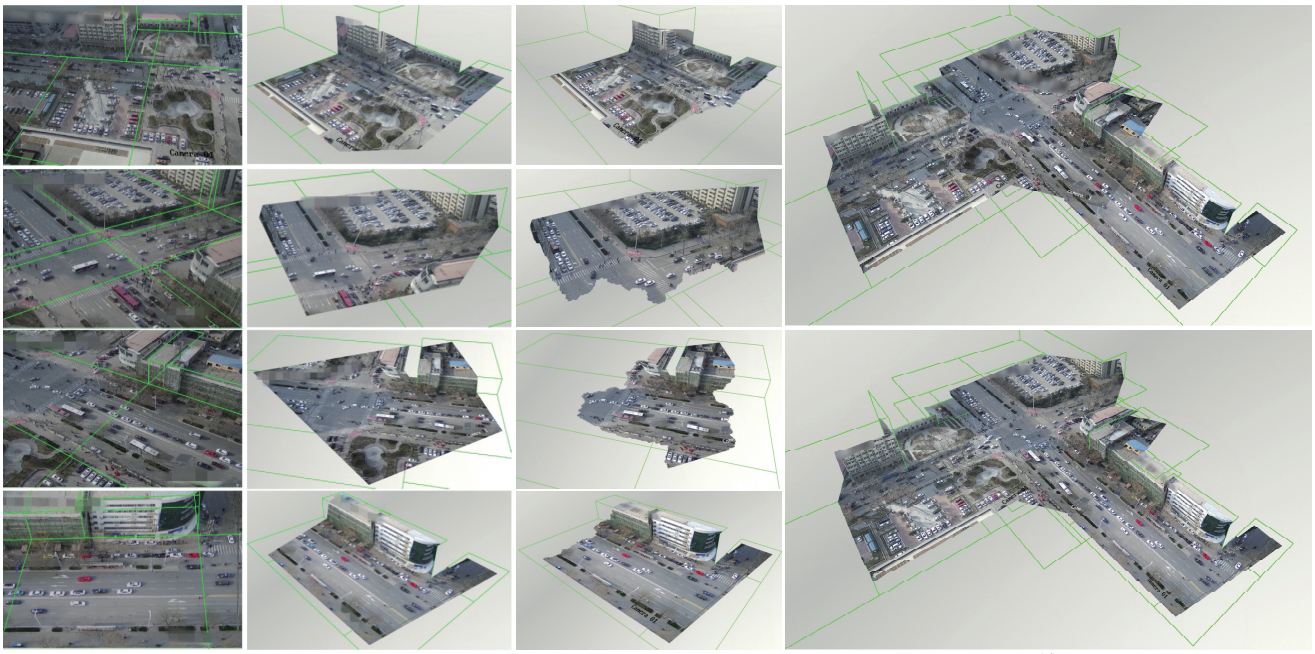


Figure 11: Modeling and stitching videos performance of Junction. (a) Input frames with line drawings (b) Interactive modeling result. (c) Culling result after stitching of close frames (d) Top, the whole modeling result before 3D stitching, and bottom, the entire modeling result after 3D stitching.

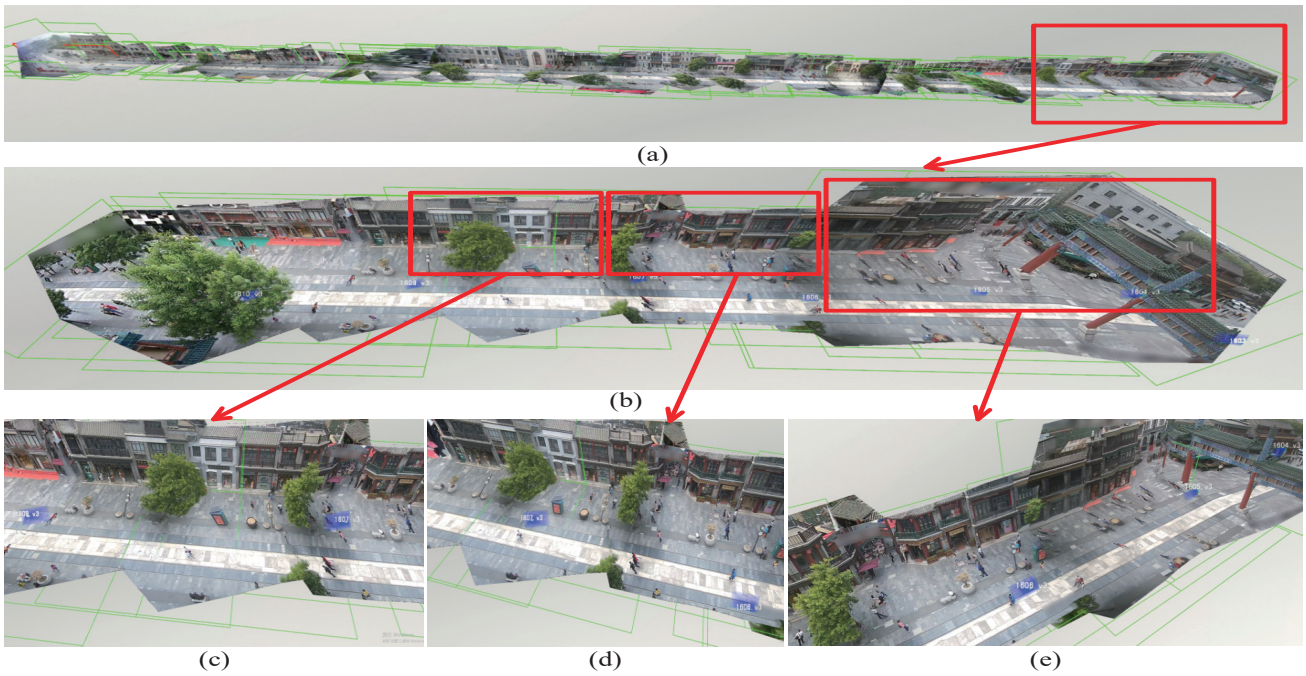
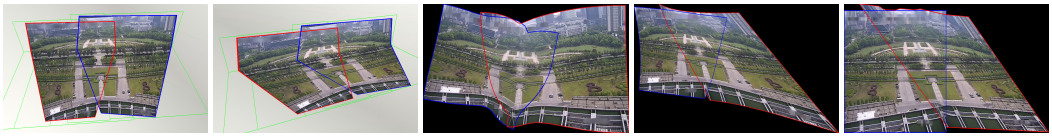
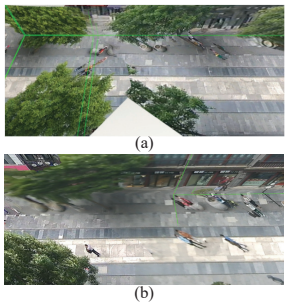


Figure 12: (a) The entire fusion result for the Street scene with 33 videos. (b) A close-up view of Street scene using 7 videos (c)(d)(e) Details of Fig. 12(b) are visible in a further close-up of the new view. The blue frustum in the scene indicates the camera pose of source videos, while the videos' name are shown with white words.



(a) (b) (c) (d) (e)

Figure 13: Comparison on stitching results. (a) Our 3D stitching result(view A). (b) Our 3D stitching result(view B). (c) GSP's result [9]. (d) WB's result[43]. (e) APAP's result [42].



(a) (b)

Figure 14: Failures. (a) Since the left camera's orientation nearly opposite to the right camera's orientation, the local plane-based matching fails. (b) Due to perspective projection, the resolution of image pieces projected on the ground is inconsistent and the local blurring occurs.

while the WB and APAP methods both make a highly distorted warp. Only our method aligns the two images with few artifacts. The root cause of this phenomenon is that a proper virtual view is used in our method, which is generated with projecting the 3D modeling result, and makes that the warp operation is done under the constraint of 3D geometric structure. So the 3D stitching method is not only more effective than 2D stitching methods, but keeps the nature of 3D scene, which allows users to skim the result through any view.

We conducted a user study to evaluate the usability and efficiency of our tool. Seven novice users of our tool and two expert users of commercial software were asked to participate in a modeling and stitching task involving 11 different models from images, which is divided into 2 groups. The modeling time taken was recorded, and the models generated were evaluated by five different evaluators. The statistics is gathered and reported. Thanks to the automatic pre-analysis, the user of our tool skips the calibration step, and their modeling speed is about twice faster than the commercial tool, Google SketchUp, while achieving a comparable modeling quality. A more benefit of our tool is that our tool provides stitching operation and generates a model with seamless texture, which would cost the artist's several hours to map. More details of the user study can be found in the supplementary material.

8 CONCLUSION

We have presented a novel method and proof-of-concept system of video fusion with a common 3D environment by modeling and stitching video background frames. Specifically, before manual operations, automatic geometric pre-analysis is conducted for modeling acceleration. The model is then derived by digesting and accommodating a large range of video frames with complex scenes in the real surveillance system. Its modeling speed is much faster than all of commodity off-the-shelf software. The proposed 3D stitch method can robustly stitch the videos in the presence of the position, orientation, attribute differences. With our method, videos captured from different locations can be integrated into the same scene and generated a spatio-temporal imagery.

Our work has several limitations. Firstly, the camera calibration may fail in certain conditions where scenes have few parallel lines. In such conditions, we allow user manually to correct the parallel lines in images like Google SketchUp. Secondly, complex shapes (e.g., sofa, chairs) cannot be modeled completely using our method. We plan to create a 3D model database to rich our model primitives, allowing the system to find the best matched one by parameterized matching. However, this would increase the modeling time and degrade the quality of 3D models. Thirdly, natural plants (such as trees) appear really hard to be modeled given the unpredetermined growth structure and appearance. For hedging camera pair, our method cannot always find correct matching and produces an unsatisfactory view for stitching(see Figure 14(a)). This requires that the users keep adjusting the result carefully, making sure that the imagery is seamless.

Additionally, our method appears less effective in dealing with situations that image pieces projected to a certain plane has strong resolution difference, resulting a blurring effect, as shown in Figure 14(b). This is due to that our method enforce the modeling result to fit the 3D scene without limiting serious warping.

Our method can be enhanced in several ways. We can extend our methods to build panorama or PTZ cameras, which are frequently adopted for surveillance systems in recent years. This would need us to detect lines in distortion condition, and build a complete model for one panorama, similar to [3]. We can also study the illumination consistency between real videos and virtual background, which is necessary for AR applications. Since we have multiple real cameras in only one background, it will be a challenge to make the background keep the same exposure as all the cameras. We will build a unified database and web interfaces for users to upload and model their own videos. Our ultimate goal is to integrate all the cameras from one scene to others even though they are in different cities or countries. If the network is practicable, the user can explore a "live Google earth" composed of real 3D videos.

ACKNOWLEDGMENTS

The work is supported by the Natural Science Foundation of China under Grant No.: 61572061, 61502020.

REFERENCES

[1] Ivanov Yuri A, Wren Christopher R, Sorokin Alexander, and Kaur Ishwinder. 2007. Visualizing the History of Living Spaces. *TVCG* 13, 6 (2007), 1153–1160.

[2] Girgensohn Andreas, Kimber Don, Vaughan Jim, Yang Tao, Shipman Frank M, Turner Thea, Rieffel Eleanor G, Wilcox Lynn, Chen Francine, and Dunnigan Tony. 2007. DOTS: support for effective video surveillance. (2007), 423–432.

[3] D Anguelov, C Dulong, D Filip, C Frueh, S Lafon, R Lyon, A Ogale, L Vincent, and J Weaver. 2010. Google Street View: Capturing the World at Street Level. *Computer* 43, 6 (2010), 32–38.

[4] O Barnich and Droogenbroeck M Van. 2011. ViBe: a universal background subtraction algorithm for video sequences. *IEEE Transactions on Image Processing* 20, 6 (2011), 1709.

[5] Lee D. C., Hebert M., and Kanade T. 2009. Geometric reasoning for single image structure recovery. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2136–2143.

[6] Yan Pei Cao, Tao Ju, Zhao Fu, and Shi Min Hu. 2014. Interactive Image-Guided Modeling of Extruded Shapes. *Computer Graphics Forum* 33, 7 (2014), 101–110.

[7] Che Han Chang, Yoichi Sato, and Yung Yu Chuang. 2014. Shape-Preserving Half-Projective Warps for Image Stitching. In *IEEE Conference on Computer Vision and Pattern Recognition*. 3254–3261.

[8] Tao Chen, Zhe Zhu, Ariel Shamir, Shi Min Hu, and Daniel Cohen-Or. 2013. 3-Sweep:extracting editable objects from a single photo. *Acm Transactions on Graphics* 32, 6 (2013), 1–10.

[9] Yu Sheng Chen and Yung Yu Chuang. 2016. *Natural Image Stitching with the Global Similarity Prior*. Springer International Publishing. 186–201 pages.

[10] Chen Shen Chi, Lee Chung Yi, Lin Chih Wei, and Chan Iok Long. 2012. 2D and 3D visualization with dual-resolution for surveillance. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 23–30.

[11] Hoiem Derek, Efros Alexei A., and Hebert Martial. 2007. Recovering Surface Layout from an Image. In *Ijcv*. 151–172.

[12] Y Furukawa and J Ponce. 2010. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 8 (2010), 1362–1376.

[13] E. Guillou, D. Meneveaux, E. Maisel, and K. Bouatouch. 2000. Using vanishing points for camera calibration and coarse 3D reconstruction from a single image. *Visual Computer* 16, 7 (2000), 396–410.

[14] Heng Guo, Shuaicheng Liu, Tong He, Shuyuan Zhu, Bing Zeng, and Moncef Gabbouj. 2016. Joint Video Stitching and Stabilization From Moving Cameras. *IEEE Trans Image Process* 25, 11 (2016), 5491–5503.

[15] Bin He, Gang Zhao, and Qifang Liu. 2012. Panoramic video stitching in multi-camera surveillance system. In *Image and Vision Computing New Zealand*. 1–6.

[16] Anton Van Den Hengel, Anthony Dick, Ben Ward, and Philip H. S. Torr. 2007. VideoTrace:rapid interactive scene modelling from video. 86.

[17] V. H. Hiep, R. Keriven, P. Labatut, and J. P. Pons. 2009. Towards high-resolution large-scale multi-view stereo. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 1430–1437.

[18] Hua Huang, Hong Liu, and Lei Zhang. 2014. VideoWeb: Space-Time Aware Presentation of a Videoclip Collection. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 4, 1 (2014), 142–152.

[19] Wei Jiang and Jinwei Gu. 2015. Video stitching with spatial-temporal content-preserving warping. In *Computer Vision and Pattern Recognition Workshops*. 42–48.

[20] Kim Kihwan, Oh Sangmin, Lee Jeonggyu, and Essa Irfan. 2011. Augmenting aerial earth maps with dynamic information from videos. *Virtual Reality* 15, 2-3 (2011), 185–200.

[21] Kwang In Kim, Kwang In Kim, Christian Theobalt, and Christian Theobalt. 2012. Videoscapes: exploring sparse, unstructured video collections. *Acm Transactions on Graphics* 31, 4 (2012), 68.

[22] J. Li, W. Xu, J. Zhang, M. Zhang, Z. Wang, and X. Li. 2015. Efficient Video Stitching Based on Fast Structure Deformation. *IEEE Transactions on Cybernetics* 45, 12 (2015), 2707–2719.

[23] Chung Ching Lin, Sharathchandra U. Pankanti, Karthikeyan Natesan Ramamurthy, and Aleksandr Y. Aravkin. 2015. Adaptive as-natural-as-possible image stitching. In *Computer Vision and Pattern Recognition*. 1155–1163.

[24] Wen Yan Lin, Siying Liu, Y Matsushita, and Tian Tsong Ng. 2011. Smoothly varying affine stitching. In *Computer Vision and Pattern Recognition*. 345–352.

[25] Alvarez Luis, Gomez Luis, and Sendra Rafael. 2010. Algebraic Lens Distortion Model Estimation. *Image Processing on Line* 1 (2010).

[26] Ming Meng, Yi Zhou, Chong Tan, and Zhong Zhou. 2018. Viewpoint Quality Evaluation for Augmented Virtual Environment. In *Advances in Multimedia Information Processing - PCM 2018 - 19th Pacific-Rim Conference on Multimedia, Hefei, China, September 21-22, 2018, Proceedings, Part III*. 223–234. https://doi.org/10.1007/978-3-030-00764-5_21

[27] Y. Nie, T. Su, Z. Zhang, H. Sun, and G. Li. 2017. Dynamic Video Stitching via Shakiness Removing. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society* PP, 99 (2017), 1–1.

[28] Sebe Ismail Oner, Hu Jinhui, You Suya, and Neumann Ulrich. 2003. 3D video surveillance with Augmented Virtual Environments. In *Proc. ACM Int'l. Workshop on Video Surveillance, Usa, Nov.* 107–112. https://doi.org/10.1145/982452.982466

[29] F. Perazzi, A. Sorkine-Hornung, H. Zimmer, P. Kaufmann, O. Wang, S. Watson, and M. Gross. 2015. Panoramic Video from Unstructured Camera Arrays. In *Computer Graphics Forum*. 57–68.

[30] Decamp Philip, Shaw George, Kubat Rony, and Roy Deb. 2010. An immersive system for browsing and visualizing surveillance video. In *ACM International Conference on Multimedia*. 371–380.

[31] Sawhney Harpreet S, Arpa Aydin, Kumar Rakesh, Samarasekera Supun, Aggarwal Manoj, Hsu Steven C, Nister David, and Hanna Keith J. 2002. Video flashlights: real time rendering of multiple videos for immersive model visualization. (2002), 157–168.

[32] Nader Salman and Mariette Yvinec. 2009. Surface Reconstruction from Multi-View Stereo of Large-Scale Outdoor Scenes. *International Journal of Virtual Reality* Volume 9, Number 1 (2009), 19–26.

[33] Grant Schindler, Panchapagesan Krishnamurthy, and Frank Dellaert. 2007. Line-Based Structure from Motion for Urban Environments.. In *International Symposium on 3d Data Processing, Visualization, and Transmission*. 846–853.

[34] S. N Sinha, D Steedly, and R Szeliski. 2009. Piecewise planar stereo for image-based rendering. In *IEEE International Conference on Computer Vision*. 1881–1888.

[35] Sudipta N. Sinha, Drew Steedly, Richard Szeliski, Maneesh Agrawala, and Marc Pollefeys. 2008. Interactive 3D architectural modeling from unordered photo collections. In *Acm Siggraph Asia*. 1–10.

[36] Noah Snavely, Steven M. Seitz, and Richard Szeliski. 2006. Photo Tourism: Exploring Photo Collections In 3D. *Acm Transactions on Graphics* 25, 3 (2006), págs. 835–846.

[37] Tan Su, Yongwei Nie, Zhensong Zhang, Hanqiu Sun, and Guiqing Li. 2016. Video stitching for handheld inputs via combined video stabilization. In *SIGGRAPH ASIA 2016 Technical Briefs*. 25.

[38] Neumann U., You Suya, Hu Jinhui, and Jiang Bolan. 2003. Augmented virtual environments (AVE): dynamic fusion of imagery and 3D models. In *IEEE Virtual Reality*. 61.

[39] Miao Wang, Ariel Shamir, Guo Ye Yang, Jin Kun Lin, Guo Wei Yang, Shao Ping Lu, and Shi Min Hu. 2018. BiggerSelfie: Selfie Video Expansion with Hand-held Camera. *IEEE Transactions on Image Processing* (2018), 1–1.

[40] Jianxiong Xiao, Tian Fang, Peng Zhao, Maxime Lhuillier, and Long Quan. 2009. Image-based street-side city modeling. *Acm Transactions on Graphics* 28, 5 (2009), 1–12.

[41] Wang Yi, Krum David M., Coelho Enylton M., and Bowman Doug A. 2007. Contextualized Videos: Combining Videos with Environment Models to Support Situational Understanding. *IEEE Trans Vis Comput Graph* 13, 6 (2007), 1568–1575.

[42] Julio Zaragoza, Tat Jun Chin, Michael S. Brown, and David Suter. 2013. As-Projective-As-Possible Image Stitching with Moving DLT. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2339–2346.

[43] G. Zhang, Y. He, W. Chen, J. Jia, and H. Bao. 2016. Multi-Viewpoint Panorama Construction With Wide-Baseline Images. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society* 25, 7 (2016), 3099.