As-Consistent-As-Possible Compositing of Virtual Objects and Video Sequences

Guofeng Zhang, Xueying Qin, Xiaobo An, Wei Chen and Hujun Bao

State Key Lab of CAD&CG, Zhejiang University, P.R.China

Correspondence to: Xueying Qin, e-mail: xyqin@cad.zju.edu.cn

Abstract

We present an efficient approach that merges the virtual objects into video sequences taken by a freely moving camera in a realistic manner. The composition is visually and geometrically consistent through three main steps. First, a robust camera tracking algorithm based on key frames is proposed, which precisely recovers the focal length with a novel multi-frame strategy. Next, the concerned 3D models of the real scenes are reconstructed by means of an extended multi-baseline algorithm. Finally, the virtual objects in the form of 3D models are integrated into the real scenes, with special cares on the interaction consistency including shadow casting, occlusions and object animation. A variety of experiments have been implemented, which demonstrate the robustness and efficiency of our approach.

Keywords: structure and motion recovery, 3D reconstruction, multi-baseline stereo, Augmented Video, interaction

Introduction

Over the past decade, Augmented Reality (AR), which aims to merge virtual objects into the real scenes, has become an invaluable technique for a wide variety of applications [2, 15]. Augmented Video is an off-line AR technique for highly demanding applications such as film-making, television and environmental assessments, in which seamless composition is of essential importance.

Most previous solutions of AR system concentrate on the geometry consistency of virtual and real scenes and thus require precise motion estimation of video camera and 3D models [4, 8]. The structure and motion recovery is a traditional problem in computer vision [17, 9, 7]. Some commercial software packages have been available, such as 2d3 Boujou [1] and REALVIZ MatchMover [19]. To the best of our knowledge, the detailed techniques used by these packages have not been published yet. Pollefeys *et al.* [17] proposed to begin with an initialization of projective structure and motion, followed by an upgrade to metric framework with self-calibration. They also employed a flexible multi-view stereo matching scheme [10] to obtain a dense estimation of the surface geometry. However, the self-calibration technique is not always stable, especially when the initially recovered projective matrices are not adequately accurate.

The goal of dense reconstruction is to extract 3D models after structure and motion recovery. Lu *et al.* presented a survey of 3D reconstruction algorithms [13]. Scharstein and Szeliski [21] introduced a two-frame stereo framework to categorize and evaluate two-frame stereo correspondences. One of its disadvantages is that using short baselines makes the matching easier at the cost of poor evaluation of depth, while using long baselines results in highly precise depth evaluation but more difficult matching. Then, multiple baseline approaches proposed in [14, 20] are popular techniques to reconstruct 3D models. However, most of them have proven to be time-consuming because the calculation is difficult to be processed on rectified scanlines.

Due to the difficulties of obtaining 3D models from the real scenes, few work has been focused on the interaction of virtual and real scenes, such as occlusions, shadows and inter-reflections. The virtual objects are usually pasted onto real scenes directly [4]. The occlusion effects can be obtained by means of several methods, such as blue screen techniques [22], and boundary registration method [11]. The 3D models of the real scenes are generally required to handle the shadows [18] and inter-reflection [5]. Alternatively, Chuang *et al.* [3] proposed to draw out the shadow displacement map from a fixed view and transfer the shadows from one image to another.

Based on our previous work [24], we propose a novel approach to merge the virtual objects into video sequences taken by a freely moving video camera. The contributions of this paper lie in several aspects. First, precise structure and motion are recovered based on precise estimation of the focal length by means of a new multi-frame strategy. Second,

we extend the two-frame scheme to multiple frames of a video sequence by a scanline-accelerated multiple baseline stereo method; the structures in multiple views are then used to construct the final 3D models. Third, built upon the reconstructed 3D models, realistic interaction effects are achieved, including occlusions, shadow casting and object animation.

The rest of this paper is organized as follows. In Section 2, our scheme on precise estimation of the camera motion is elaborated. Section 3 presents our extended multiple baseline method for the dense 3D reconstruction. The composition of the virtual objects and the real scenes is described in Section 4. Experimental results are given in Section 5. Finally, we conclude the whole paper.

Camera Motion Estimation of Video Sequences

Highly accurate estimation of camera motion is essential to match the geometry of the virtual objects and the real scenes. Here, we propose a robust camera tracking algorithm based on automatically extracted key frames, in which the focal length is firstly recovered by a robust multi-frame strategy.

Feature Matching

We adopt SIFT algorithm [12] to extract the features from each frame of the input video sequence and match the features frame by frame. Corresponding features are constrained according to the epipolar geometry theory [25]. We use RANSAC algorithm [6] to find a

set of inliers that have consistent epipolar geometry. Then the matched feature points in frames are chained. Each chain is called a matching set, which corresponds to a 3D point. Generally, the longer the length of a matching set is, the higher the reliability is, and more advantageous for structure and motion estimation.

The Camera Model

We parameterize each camera motion using seven parameters, i.e., the rotation expressed by three Euler angles $\Theta = (\theta_x, \theta_y, \theta_z)$, the translation expressed by a triple vector $t = (t_x, t_y, t_z)$, and the focal length f. The intrinsic matrix K can be written as:

$$K = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

where (c_x, c_y) denotes the image center. Each matching set represents a 3D point $\mathbf{X} = (X, Y, Z)^\top$, and its re-projection in each frame is:

$$\mathbf{x} = K(R\mathbf{X} + t) \tag{2}$$

where R denotes the rotation matrix represented by the rotational Euler angles and $\mathbf{x}$ is the the homogeneous image positions of $\mathbf{X}$ in a frame.

Typically, camera motion estimation aims to compute K, R, and t of each frame in a video sequence. If camera zooming in or out is excluded, the unknown camera intrinsic

parameter f remains constant, and can be calculated before the estimation of the extrinsic camera parameters.

According to the epipolar geometry theory, two corresponding points $\mathbf{x}'$ and $\mathbf{x}$ in two images take [25]:

$$\mathbf{x}'^{\top} F \mathbf{x} = 0 \tag{3}$$

where the fundamental matrix F has $F = K^{-\top}[t] \times RK^{-1}$. This indicates that a point in one image matches a line $F\mathbf{x}$ on another image. Thus, any corresponding point $\mathbf{x}'$ lies on the line $F\mathbf{x}$. The sum of symmetric epipolar distances of all corresponding points is used to measure the errors:

$$D = \sum_j (\mathbf{x}_j'^{\top} F \mathbf{x}_j)^2 \left(\frac{1}{(F\mathbf{x}_j)_1^2 + (F\mathbf{x}_j)_2^2} + \frac{1}{(F^{\top}\mathbf{x}_j')_1^2 + (F^{\top}\mathbf{x}_j')_2^2}\right) \tag{4}$$

Selecting Key Frames

As Fitzgibbon and Zisserman indicated in [7], the number and the length of the interest point tracks have a significant effect on the stableness and accuracy of structure and motion estimation. Now that we have obtained matching sets of feature points as described before, we choose only those matching sets whose track lengths are no less than N to participate in estimation. We call those matching sets *golden tracks*. In our experiments, the value of N is usually chosen in the range of 15-35.

Generally, the larger the interval of key frames, the longer the baseline, and the more beneficial to estimation. However, it may cause the shortness of matching sets if the interval

is too large. We set the interval to be $(N-1)/2$ to ensure that any golden track appears in at least two key frames, or say, it must be able to participate in estimation.

Nevertheless, if there is not adequate golden tracks between two successive key frames, we have to supplement some other matching sets appearing in these two key frames into the golden tracks. If this is insufficient, we insert a key frame between them, and select golden tracks in the same manner.

Practically, we select two key frames that are suitable for initializing the sequential structure and motion computation. It should be satisfied that the baseline between two frames is long enough with sufficient common golden tracks. Pollefeys *et al.* [17] proposed to use the image-based distance to size the length of a baseline:

$$b = median(d(H\mathbf{x}, \mathbf{x}')) \tag{5}$$

where H is the planar-homography and can be solved by minimizing b. By denoting n_{ij} as the number of common golden tracks between the i-th and j-th frames, we define the following formula to evaluate the suitability for two frames:

$$d_{ij} = n_{ij}^{\alpha} b_{ij} \tag{6}$$

Here we choose $\alpha = 0.5$ in our experience because highly common correspondences are unnecessary.

Estimation of The Focal Length Based on Multi-frames

The focal length plays a very important role in camera motion estimation. The robustness of its evaluation influences the accuracy of estimation of camera motion. However, it is very difficult to recover precise focal length from only two frames under the influences of matching noises. This has been examined by our experiments. As shown in Figure 1(a), even for any two key frames with a large camera motion, the curve of the re-projection errors is flat, which means that f is very unstable. On the contrast, the curve of re-projection errors has an obvious minimum if multiple frames are employed as shown in Figure 1(b). Therefore, the recovery of the focal length tends to be more precise with multiple frames.

Based on these investigation, we propose a multi-frame strategy to precisely recover the focal length f by the 1D searching process. In the first step, given a value of f, we compute R and t of the two key frames by minimizing the cost function D in Eq. 4 with Levenberg-Marquardt algorithm. Then the 3D position $\mathbf{X}_j$ of each common feature point can then be reconstructed from its two projections $\mathbf{x}_j$ and $\mathbf{x}'_j$ by computing the intersection of the two corresponding space rays. This process is called a *triangulation*. In the second step, we select several frames between these two key frames and solve their camera motion parameters. We define the cost function $e(f)$ for all re-projection errors in the selected frames as:

$$e(f) = \sum_{i \in \psi} \sum_{j \in \chi(i)} \|r_{ij}\|^2 \tag{7}$$

where ψ is the set of all selected frames, $\chi(i)$ is the superscript of a golden track on the i-th

frame, and r_{ij} is the residual of the j-th 3D point $\mathbf{X}_j$ in the i-th frame image:

$$\lambda_{ij}\hat{\mathbf{x}}_{ij} = K(R_i|t_i)\hat{\mathbf{X}}_j, r_{ij} = \hat{\mathbf{x}}_{ij} - \mathbf{x}_{ij} \tag{8}$$

Here, R_i and t_i are the rotation matrix and translation vector from the first key frame to the i-th frame, respectively.

We calculate $e(f)$ iteratively by repeating the first and second steps and compute f that minimizes $e(f)$. The fibonacci searching method is suitable for this process. The initial values of R_i, t_i of each iteration can be set to the estimated values in the previous iteration. It is apparent that the first step is more time-consuming than the second step because 3D points $\hat{\mathbf{X}}_j$ are unknown and the cost function D is much more complex than e. Since 3D point $\hat{\mathbf{X}}_j$ have been estimated from the first step, the cost function e is easy to be obtained even when there are more frames involved. In our experiments, it costs less than half a minute to find an appropriate enough value for f.

Incremental Motion Estimation

Based on the recovered structure and motion of the first two key frames, other key frames are incrementally handled along with the refinement of the existing key frames.

For each successive new key frame, its camera motion represented by R and t is initialized as its previous one and then estimated by employing the recovered structure of the golden tracks to minimize the following re-projection errors:

$$\sum_j d(\mathbf{x}_j, K(R\mathbf{X}_j + t))^2 \tag{9}$$

Then, new golden tracks are reconstructed by the newly estimated motion. Therefore, both new motion and structure are obtained. Then we adopt the modified local bundle adjustment [26] to refine the existing structure and motion.

After all of the key frames are processed, camera motions of the whole sequence are recovered easily from the precise structure by Equation 9. Finally, if necessary, all of the structure and motion for the whole sequence are refined through a bundle adjustment [23, 9].

3D Reconstruction of The Real Scenes

The best way to achieve the most consistency between the virtual objects and real scenes is to recover the geometry of the real scenes, especially wherein the virtual objects and the real scenes interact. Although it is impractical to reconstruct all objects in scenes, all AR-based applications normally only concern only partial scenes, which can be indicated interactively.

Computations of The Dense Depth Maps

As shown in Figure 3(a), for two frames viewed from C_l and C_r, there are two homographies $\mathbf{H}^l$ and $\mathbf{H}^r$ (3×3 matrix) which map them to the left image I^l and the right image I^r on the common image plane Π. The common image plane is parallel to the baseline C_lC_r with the length B. As a result, the epipolar lines [25] in two frames become the same scanline. Thus, the dense matching is accelerated by handling it along scanlines. We employ the SAD (sum-of-absolute-differences) algorithm with a window W centered at (x_0, y_0) as the

matching function, to find the best disparity d at (x_0, y_0):

$$SAD(x_0, y_0, d) = \sum_{x,y \in W} |\mathbf{I}_l(x, y) - \mathbf{I}_r(x - d, y)| \tag{10}$$

After the dense matching is accomplished, the dense depth is calculated according to Equation 11:

$$Z = Bf/d \tag{11}$$

where Z is the depth, B is the length of the baseline, and d is the disparity of a point. In this way, a dense depth map is obtained.

Multi-Baseline Stereo of Video Sequences

The dense matching of image pairs usually does not produce accurate dense depth map due to noises. Our scheme is to exploit multi-baseline stereo of a video sequence to recover the dense depth map. If the camera moves along a line, multi-baseline stereo can be easily dealt with (see [14] for details). However, since the baselines are seldom parallel, the correspondences of a 3D point in different stereo image pairs are difficult to be mapped on the same scanline. A common measurement for all stereo image pairs in the video sequence is required to extend multi-baseline stereo to all frames in a video sequence.

Without loss of generality, we take any three frames F_1, F_2 and F_3 as an example (see Figure 3(b)). We use the subscripts to index the image pairs, and $(x, y, 1)$ to denote the uniform homogeneous coordinates in the image plane. We assume that the image pairs (F_1, F_2), and (F_1, F_3) are rectified as described in Section 3.1, and the projec-

tions of a 3D point P on the rectified images $I^l_{1,2}$ and $I^l_{1,3}$ are $\mathbf{x}_{12} = (x_{12}, y_{12}, 1)^\top$ and $\mathbf{x}_{13} = (x_{13}, y_{13}, 1)^\top$, respectively. According to Equation (11), the 3D position of P can be calculated by re-projecting $\mathbf{x}_{12}$ and $\mathbf{x}_{13}$ back to 3D space from $I^l_{1,2}$ and $I^l_{1,3}$, and expressed as $P_{12}(x_{12}B_{12}f/d_{12}, y_{12}B_{12}f/d_{12}, B_{12}f/d_{12})$, $P_{13}(x_{13}B_{13}f/d_{13}, y_{13}B_{13}f/d_{13}, B_{12}f/d_{13})$, respectively. P_{12} and P_{13} are different expressions of the same 3D point in different coordinate system. Therefore, we have $\|P_{12}\| = \|P_{13}\|$, yielding:

$$\frac{d_{12}}{B_{12}\sqrt{x_{12}^2 + y_{12}^2 + 1}} = \frac{d_{13}}{B_{13}\sqrt{x_{13}^2 + y_{13}^2 + 1}} \tag{12}$$

Equation 12 builds up the common measurement for all frames relating to F_1. It implies that if we plot the matching curves as a function of $d' = d/(B\sqrt{x^2 + y^2 + 1})$, i.e., $SAD(d')$, instead of $SAD(d)$, all curves should have the same minimum at d'. Therefore, rather than using matching curves $SAD_{12}(d')$, $SAD_{13}(d')$ separately, we employ $SAD(d') = \sum_i SAD_{1i}(d')$ as the matching curve for the first frame F_1 of multiple frame stereo as shown in Figure 3(c). In this way, the dense depth map of a frame is obtained by multi-baseline stereo. The outliers influence the accuracy of the depth matching due to noises, mismatching and/or occlusions of feature points. We adopt the method proposed by Koch *et al.* [10] to remove these outliers in this process, which decreases the noises efficiently.

With the known dense depth maps of key frames, the triangulation of the surfaces in scenes can be processed conveniently. We apply a Gaussian filter or bilateral filter on the depth map to remove noises further. Finally, several 3D models relating to the selected key frames are obtained by the process described above.

Merging 3D Models

To merge all 3D models relating to different key frames, a global coordinate system is desired. This coordinate system can be a plane, a cylinder, or a sphere. Due to the occlusions and invisibility, it is almost impossible to map all scenes onto this global coordinate system. In this sense, it remains a challenging problem in general.

Note that, the recovered surfaces may have a little difference with each other due to the existence of the reconstruction errors and noises. The surfaces which are parallel to the image plane have higher quality compared with those vertical to the image plane. We select a plane that is visible by all surfaces, and set its resolution as that of the image plane. We then perform image editing on all the depth maps to remove the regions with low quality.

For a sequence of 3D models, we project the 3D model of the first key frame onto the common plane. When the 3D model of the successive key frame is projected onto the plane, the projection is dropped if some pixel has been written. In this way, we get the connections of these two models. Usually, this connection region has different depth, and the errors are visible for shadow casting. We blend this connection region around the edge region to smooth the models. Accordingly, we merge the one of the successive key frame, till all selected key frames are integrated. Finally, a 3D model from all key frames are obtained.

In addition, due to the introduced noises and shortage of features, the reconstructed 3D model is typically coarse and not hole-free, especially when the object is relatively far away from the viewpoint. Since the model is merged on a plane, it can be treated as an image

whose elements record the depth information. Our scheme is to interactively specify the regions with holes and perform Poisson image editing [16] to repair them.

Integrating the Virtual Objects into The Real Scenes

With the obtained precise camera motions and 3D models, it is convenient to incorporate virtual objects into the video sequences. For seamless composition, the virtual and real scenes should be in the same illumination environments. Another main challenge to achieve this goal is the consistency of geometry and illumination interactions among the virtual and real objects, such as occlusions, action design of characters, shadow casting, inter-reflections, *etc*.

For inter-reflection of the virtual and real objects, the direction of dominated light has to be decided beforehand. In outdoor scenes, the direction of sunlight can be computed conveniently with the recovered 3D models. We can find the locations of an occluder and its shadow receiver extracted from certain key frame. We then set the vector between them as the lighting direction.

An amazing effect facilitated by the reconstructed 3D models is the shadows. Typically, there are two kinds of shadows. One is the shadows from the real scenes onto the virtual objects whose masks can be rendered with the 3D models of real scenes. The other one is casted from the virtual objects onto the real scenes. Difficulty happens with the later. If there have already partial shadows in the region of the real scenes, this region is desired to

be known. However, the recovered 3D models can hardly be used to generate the exactly identical shadow maps with those in the real scenes. In this case, the shadow regions with and without the virtual objects can be first obtained based on the reconstructed 3D models. The edges of the virtual shadows and real shadows are then detected and refined by common image processing techniques. Subsequently, the differences between both types of shadows can be removed by appropriate smoothing operators.

It remains a challenge for arbitrarily complex scenes because the 3D geometry, the illumination environment and the material of the real scenes, are difficult to be completely recovered.

Experimental Results

We implemented our approach on a PC with an Intel P4 2.4G HZ CPU and 1024MB memory. For a video sequence of 126 frames, it costs about 6 minutes to track the feature points, and 2 minutes for the computation of the camera parameters. The time spent on the reconstruction of each 3D model in one key frame depends on the number of the multi-baselines. For typical case, e.g. 10, it costs about 4 minutes. Typically, key frames are selected with the interval of 15-30 frames. The images of the virtual objects, occlusion masks and shadow masks are generated with 3DSMAX by integrating the 3D models of the virtual objects and real scenes. The final composition takes about 5 minutes in our experiments. As a result, the total time for one video sequence of 126 frames is about 40 minutes without including

the time of the rendering and user interactions.

Four examples with 126,121,121,279 frames are demonstrated, of which several selected frames are illustrated in Figure 5 from top to bottom. The video camera of the first video sequence moves upwards to view the far temple, till the square in front of the temple appears. The 3D model shown in Figure 2(e) of the stairs are reconstructed from four key frames as shown in Figure 2(a-d).

One frame of each original video sequence is shown in Figure 4(a). The virtual objects with reconstructed 3D models are illustrated in Figure 4(b). Their corresponding composited frames are demonstrated in Figure 4(c). In order to view the shadows and occluding effects clearly, Figure 4(c) is magnified, yielding Figure 4(d). It is worthy mentioning that, in the first example, three characters and one balloon are animated and composed in the reconstructed model. The shadows of the animated objects are cast onto the reconstructed stairs. The red character which stands on the top of the stairs is partially occluded at the beginning and becomes totally visible with the moving of the video camera. For the second example, a virtual helicopter flies over the sculpture, and a character jumps into shadows. The ground is not recovered because there are not enough features. We simply define a plane to mimic the ground. In this way, correct shadow casted from the jumping character is achieved.

In the third example, its video camera focuses on the towers and moves in a nearly horizontal way. Note that, we insert a big virtual tower in the yard, whose foot part is occluded by the wall. For the forth one, the video sequence is taken by a hand-held camera

which makes the views are shaky. To our surprise, a virtual moving car on the road matches the background video sequence quite consistently, which demonstrates the robustness of our camera tracking system. The specular reflection effects of the car are processed by environment mapping. For more detailed information, please view our video submission.

Conclusions

We have proposed a novel method to integrate virtual objects into video sequences taken by a freely moving video camera. The virtual objects are coincided with the real scenes in the sense of geometrical, illumination and interaction consistencies by means of precise camera motion estimation and 3D reconstruction. Specifically, based on the recovered 3D models of the real scenes, interaction effects of the virtual objects and the real scenes are successfully achieved. The shadows cast from the virtual objects onto the real scenes and the real scenes onto the virtual objects are both obtained naturally. Moreover, the occlusion relationship of the real scenes and the virtual objects are also handled.

Acknowledgements

The second author would like to express her sincere appreciation to Prof. E. Nakamae for the fruitful discussion in the early stage of this study. This paper is supported by 973 program of China (No.2002CB312104), NSF of China (Grant No.60021201, Grant

No.60373035) and Specialized Research Fund for the Doctoral Program of Higher Education (No.20030335083).

References

[1] 2d3. http://www.2d3.com.

[2] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre. Recent advances in augmented reality. *IEEE Computer Graphics and Applications*, 21(6):34–47, 2001.

[3] Y.-Y. Chuang, D. B. Goldman, B. Curless, D. H.Salesin, and R. Szeliski. Shadow matting and compositing. In *Proceedings of ACM SIGGRAPH 2003*, pages 494–500, July 2003.

[4] K. Cornelis, M. Pollefeys, and L. J. V. Gool. Tracking based structure and motion recovery for augmented video productions. In *VRST*, pages 17–24, 2001.

[5] P. Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proceedings of ACM SIGGRAPH 1998*, pages 189 – 198, July.1998.

[6] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[7] A. Fitzgibbon and A. Zisserman. Automatic camera tracking. In M. Shah and R. Kumar, editors, *Video Registration*, chapter 2, pages 18–35. Kluwer, 2003.

[8] I. Gordon and D. G. Lowe. Scene modelling, recognition and tracking with invariant image features. In *ISMAR*, pages 110–119, 2004.

[9] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, Cambridge, 2000.

[10] R. Koch, M. Pollefeys, and L. V. Gool. Multi viewpoint stereo from uncalibrated video sequences. In *Proceedings of the 5th European Conference on Computer Vision-Volume I*, pages 55 – 71, 1998.

[11] V. Lepetit and M.-O. Berger. Handling occlusions in augmented reality systems: A semi-automatic method. In *Proc.Intl Symp. Augmented Reality 2000 (ISAR 00)*, pages 137–146, Los Alamitos, Calif., 2000.

[12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[13] Y. Lu, J. Z. Zhang, Q. M. J. Wu, and Z.-N. Li. A survey of motion-parallax-based 3D reconstruction algorithms. *IEEE Transaction on SMC-C*, 34(4):532–548, December 2004.

[14] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, April 1993.

[15] G. Papagiannakis, S. Schertenleib, B. O'Kennedy, M. Arevalo-Poizat, N. Magnenat-Thalmann, A. Stoddart, and D. Thalmann. Mixing virtual and real scenes in the site of ancient pompeii. *Computer Animation and Virtual Worlds*, 16(1):11–24, February 2005.

[16] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Trans. Graph.*, 22(3):313–318, 2003.

[17] M. Pollefeys, L. J. V. Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera. *International Journal of Computer Vision*, 59(3):207–232, 2004.

[18] X. Qin, E. Nakamae, and K. Tadamura. Automatically compositing still images and landscape video sequences. *IEEE Computer Graphics and Appliactions*, 22(1):68–78, 2002.

[19] REALVIZ. http://www.realviz.com.

[20] T. Sato, M. Kanbara, N. Yokoya, and H. Takemura. Dense 3-d reconstruction of an outdoor scene by hundreds-baseline stereo using a hand-held video camera. *International Journal of Computer Vision*, 47(1-3):119–129, 2002.

[21] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

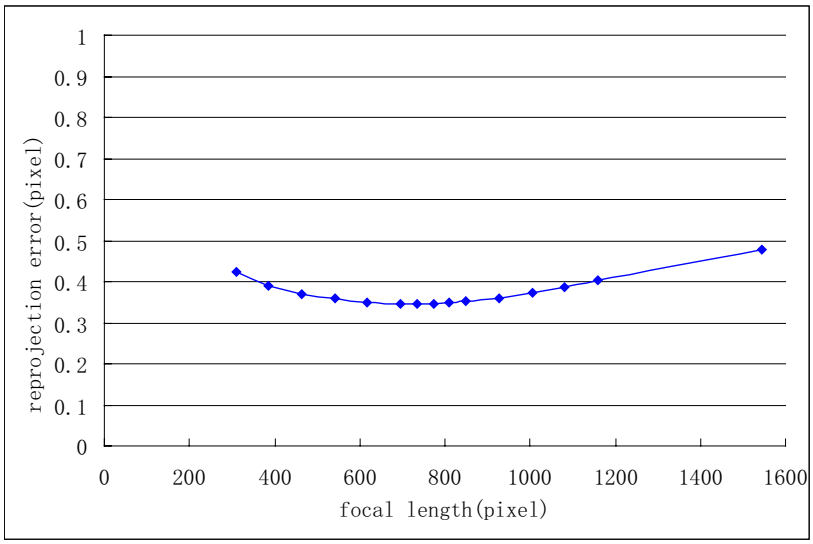
[22] A. R. Smith and J. F. Blinn. Blue screen matting. In *Proceedings of ACM SIGGRAPH 1996*, pages 259 – 268, July 1996.

[23] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Workshop on Vision Algorithms*, pages 298–372, 1999.

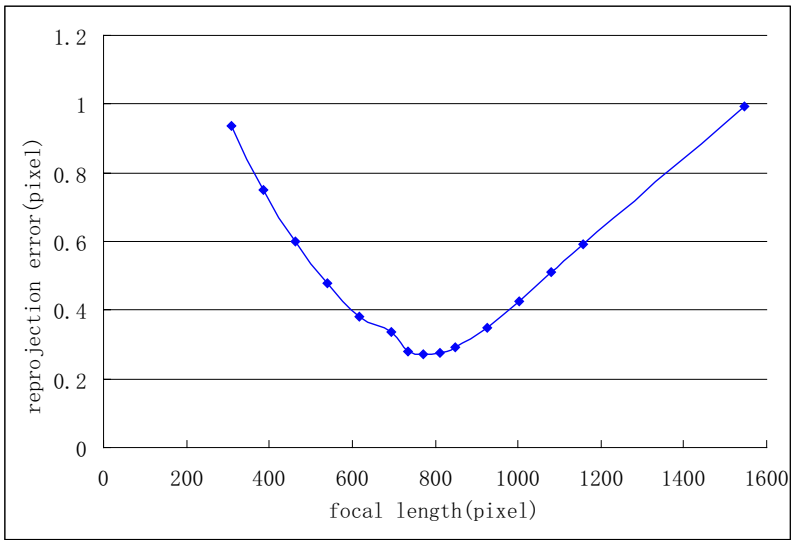
[24] G. Zhang, X. Qin, X. An, W. Chen, and H. Bao. Seamless compositing of virtual objects and video sequences. In *Proceedings of Pacific Graphics 2005*, pages 78–81, Macau, October 2005. Short paper.

[25] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27(2):161–195, 1998.

[26] Z. Zhang and Y. Shan. Incremental motion estimation through modified bundle adjustment. In *ICIP (2)*, pages 343–346, 2003.



(a)



(b)

Figure 1: Illustrations of the estimation of the focal length f. The true value of f is 772.5. (a) The cost function based on two frames. The energy curve approximates a flat line. (b) The cost function based on multiple frames. The energy curve takes the form of a parabola and has an obvious minimum.

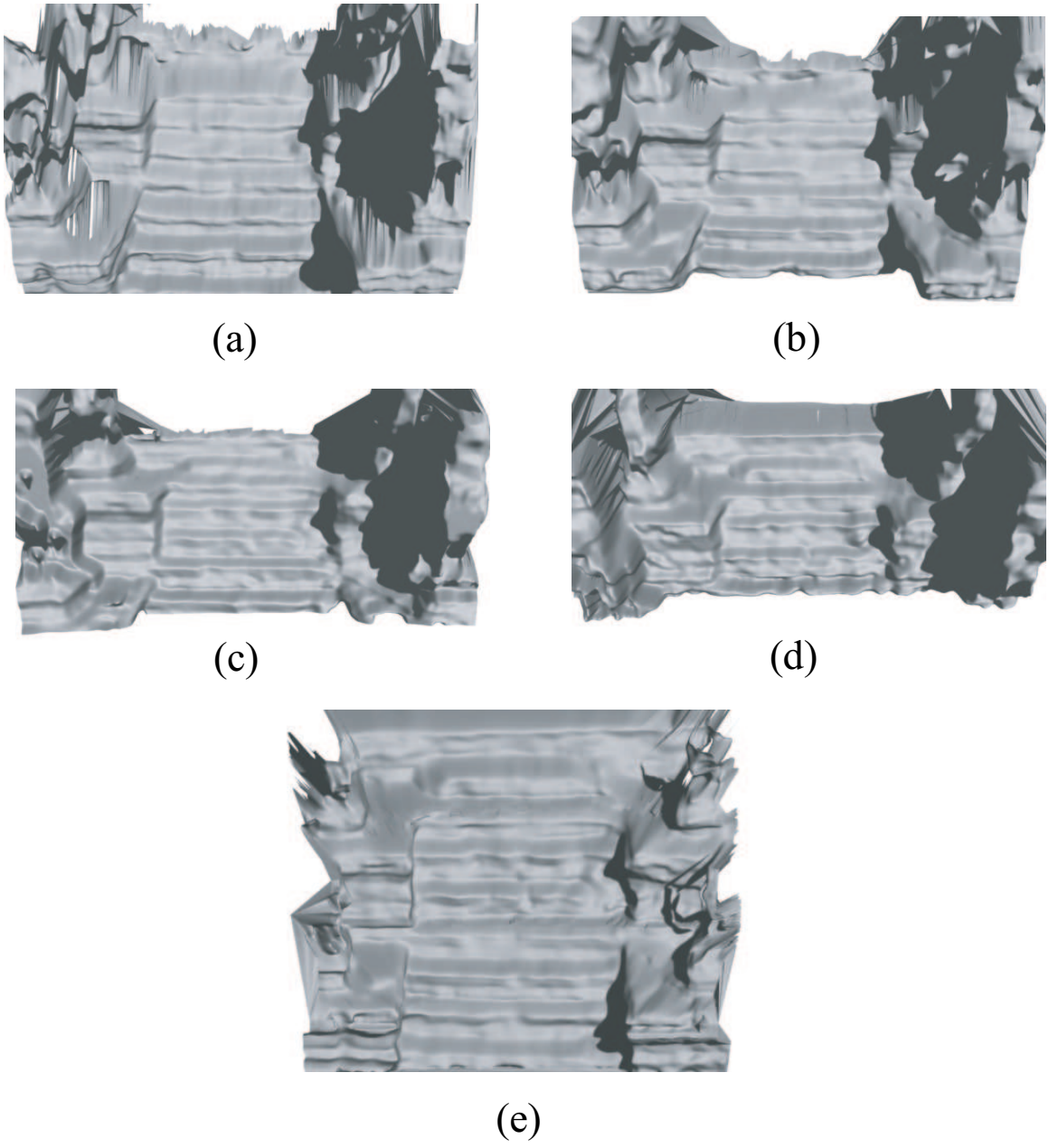
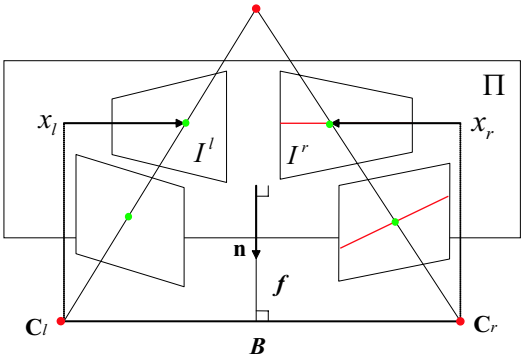
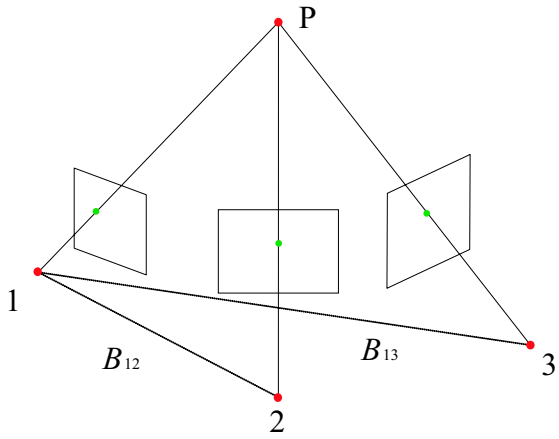


Figure 2: The reconstructed 3D models from the video sequence based on four key frames and the final model.

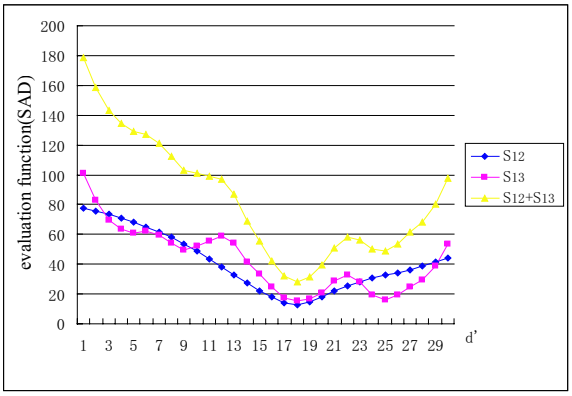


$$d = x_l - x_r$$

(a)



(b)



(c)

Figure 3: Illustrations of the multi-baseline technique. (a) Two-view geometry; (b) Three-view geometry; (c) The accuracy comparison for different baselines with two stereo pairs.

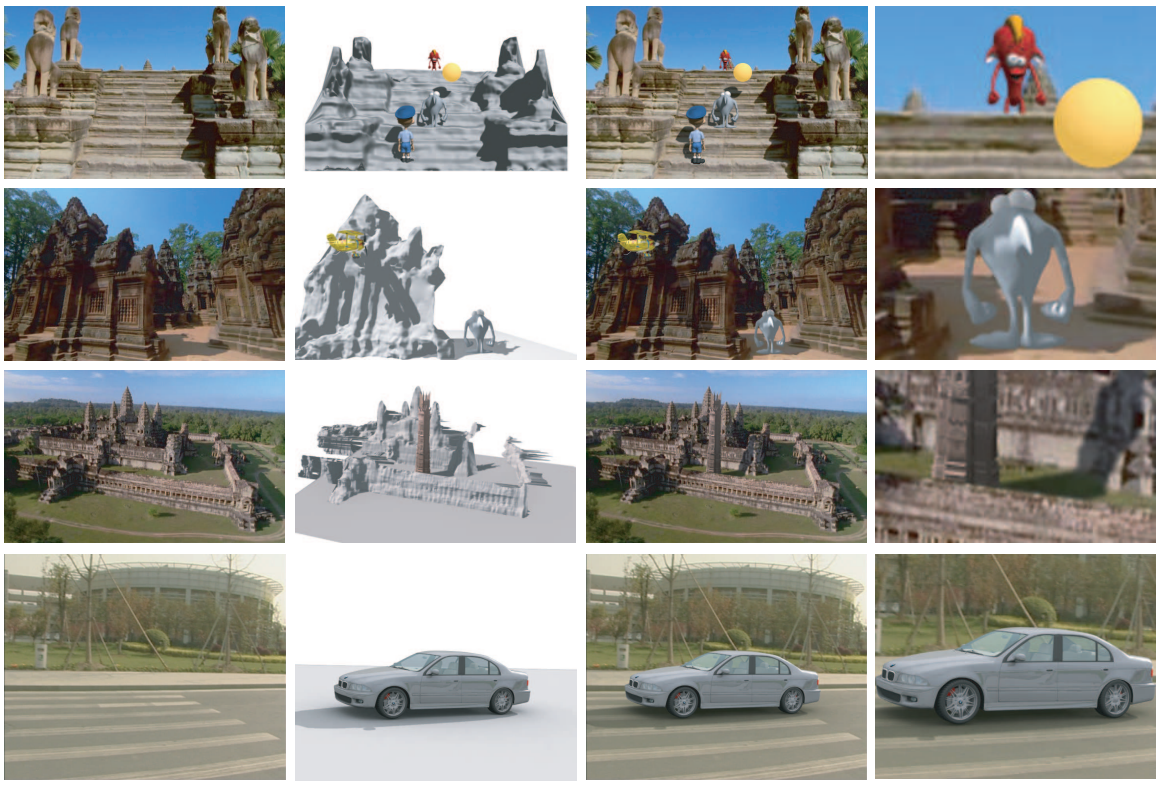


Figure 4: (a) One selected frame image; (b) Animated objects are composed with the 3D models scenes taking account of shadows and occlusions; (c) Animated objects are composed in the video sequence taking account of shadows and occlusions; (d) The magnified snapshots of (c).

Figure 5: Selected key frames from four composite video sequences.