VIDEO STITCHING WITH SMALL OVERLAP

1st Chaoyu Xie
University of Science and Technology of China
Anhui, China
xcy12345@mail.ustc.edu.cn

2nd Xuejin Chen
University of Science and Technology of China
Anhui, China
xjchen99@ustc.edu.cn

***Abstract*—Video stitching remains a challenging problem in computer vision. In this paper, we propose a novel method to stitch multiple videos which have a small overlapped region. The algorithm consists of three steps: (1) calibrating the camera and projecting the video frame into spherical coordinates. (2) detecting the edges of each spherically warped frame and calculating homegraphy matrix in each grid. (3) updating seam and stitching the origin videos to produce panoramic videos. The proposed method has proven to be more robust on small overlapped region. Experimental results show that our approach achieves better panoramic videos than state-of-the-art ones.**

***Index Terms*—video stitching, panorama, overlap, edge detection**

I. Introduction

Video stitching is the process to merge several videos including overlapped regions into a panoramic video. The holy goal of video stitching is to acquire a large view video that looks as natural as possible. As a result of the widespread use in security monitoring, virtual reality and medical image analysis, video stitching has become a hot topic in recent years.

However, in previous video stitching systems [1]–[4], cameras used to capture multiple videos usually have large overlap, which can be easily handled. Under this condition, there are lots of commercial softwares such as VideoStitch Studio [5], AutoPano [6], compute a stitching model, which is usually a 2D transformation, according to the relative position and angle between two cameras. When the overlap between cameras is very small, these methods and softwares can not work as good as we expected.

In this paper, we try to solve the problem of stitching videos which have small overlap. For example, a typical scenario we envision is: six cameras fixed on a tripod. Fig. 1 shows our device and the origin data. Although the field of view (FOV) of the cameras is very large, the overlap between the cameras is still small. Stitching such six videos is very challenging due to two major reasons: (1) the captured videos have small overlap that the feature points can not be matched correctly and (2) the structure of buildings in the panoramic video could have ghost. To deal with this problem, we proposed a new method which can handle the problem that we have listed above. Different from the period work [3], we mainly focus on video stitching which has a small overlapping area. Our algorithm first calculate the parameters of the cameras, and then project the videos into sphere. We extract the edge of each video, and according to the edge we mesh these video into grids frame by frame. And calculating the homegraphy matrix of each gird. Finally, we update seam and produce a panoramic video. Fig. 2 shows the video stitching process presented in this paper.

(a) (b)

Fig. 1. The device we use in this paper and the video example. (a) shows the six cameras fixed on a cube, (b) shows the example of videos captured by our devices.

II. Related work

In this section, we briefly review the most related works in image stitching and video stitching.

Image Stitching: Image stitching is a well-studied, yet still active research area in [7]–[17]. Early methods adopted a single homegraphy to align images. The single homegraphy is valid only when the camera rotates around its optical center or the scene captured is planar [9]. When there is parallax between the images, artifacts such as structure distortion and ghost occur. In general, there are two kinds of stitching strategies: warp-based [10]–[12] and seam-driven [14]–[16]. In the warp-based category, Gao *et al.* proposed to use two homographies for image stitching when the scene could be modeled roughly by two planes (ground plane and distant plane) [17]. Zaragoza *et al.* proposed an as-projective-as-possible (APAP) mesh deformation that warps images by following a global projective transformation and allows local non-projective deviations [12]. Chang *et al.* proposed a shape-preserving half-projective (SPHP) method to correct distortions in non-overlapping regions [11]. Lin *et al.* proposed Adaptive As-Natural-As-Possible (AANAP) which based on APAP but computes the warpping fully automated [13]. In the

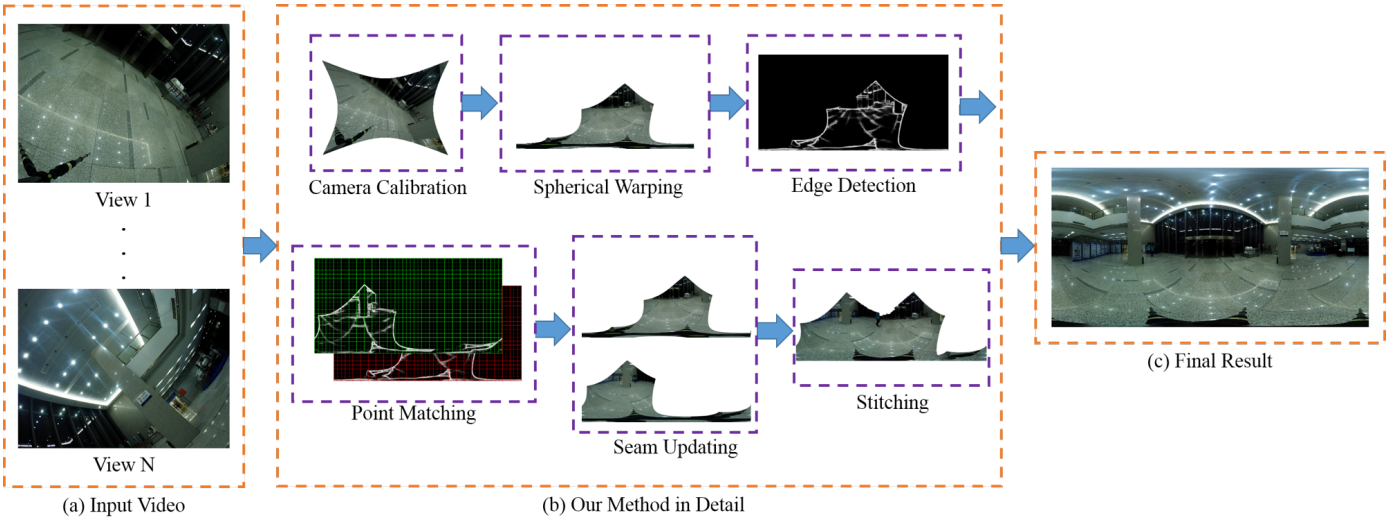

Fig. 2. The pipeline of our video stitching algorithm. First, we pre-process the video data, such as calibrating the camera and projecting the video frame into spherical coordinates. Second, we detect the edge of each spherically warped frame and re-calculate homography matrix in every grid. Third, updating seam and stitching the videos to produce a panoramic video.

seam-driven category, Agarwala *et al.* proposed photomontage that composites a photograph by cutting and stitching multiple photographes seamlessly [16], while Zhang and Liu looked for a homography that leads to a minimum energy seam to stitch large-parallax images [14]. In addition, several freewares and commercial softwares are available for performing image stitching. AutoStitch, Microsofts Image Compositing Editor, and Adobe's Photoshop CS6 mosaicing feature.

Video Stitching: Compared to image stitching, video stitching received much less attention. The main challenges are the moving objects in overlapped regions are difficult to handle and the complexity of the video stitching algorithm is too high. Different approaches have been proposed for different camera settings. In general, these can be divided into two categories: fixed cameras [1], [3], [18]–[20] and moving cameras [2], [4], [21]. In the fixed cameras category, Zheng *et al.* stitched the videos frame by frame just like image stitching [1]. Jiang *et al.* proposed spatial-temporal content-preserving warping (STCPW) to eliminate small deviations [3]. In the moving cameras category, Lin *et al.* proposed a method which is for independently moved mobile devices. They recovered the 3D camera paths and a sparse set of 3D scene points to stitching videos [21]. Guo *et al.* calculated both inter motions and intra motions, then calculated the camera path, and finally stitching them together [2]. Nie *et al.* which was similarly to [2] but could distinguish between right and false matches [4]. All these works have one thing in common, that is the overlap between the cameras is large. Under this condition, stitching become much more easier. However, in this paper, we will stitch videos which have a very small overlap between the cameras.

III. OUR METHOD

Fig. 2 shows the pipeline of our video stitching algorithm. The first step of our algorithm is calculating the parameters of the cameras, and projecting the origin video into a spherical coordinate system frame by frame, see details in Sec. III-A. The second step is extracting the edges on every frame and meshing the frame into grid and calculating homography matrix in each grid. Then matching the points in overlapped region which is described in Sec. III-B. The third step is stitching them into a panoramic video according to the spatial-temporal information, as described in Sec III-C.

A. Pre-process For Video Stitching

Pre-process for video stitching composes of two parts, the one is calculating the parameters of the cameras, the other is projecting the video into a new coordinate system. The parameters of a camera consist intrinsics and extrinsics. According to [22], we calculate the intrinsics and extrinsics of the cameras. Then video can be projected into spherical coordinate system. Let $\mathbf{x} = [x\ y]^T$ be the point of spherically warped frame F and $\hat{\mathbf{x}}' = [\hat{x}'\ \hat{y}']^T$ be the point of corrected frame F_g. The relationship between F and F_g can be described as following equation:

$$\begin{gathered}\theta = (x - x_c)/f, \varphi = (y - y_c)/f \\ x' = sin\theta cos\varphi, y' = sin\varphi, z' = cos\theta sin\varphi \\ \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} = \mathbf{R} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \\ \hat{x}' = f\frac{\hat{x}}{\hat{z}} + x_c \\ \hat{y}' = f\frac{\hat{y}}{\hat{z}} + y_c \end{gathered} \tag{1}$$

where $\mathbf{R}$ stands for the extrinsics matrix, x_c and y_c stands for the center of F, f stands for the focal length of the camera. Fig 3 shows the origin frame, corrected frame and spherically warped frame.

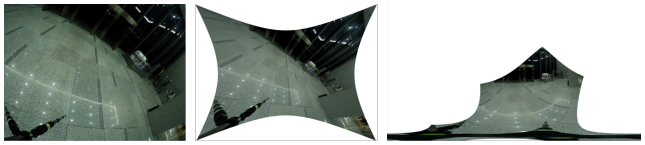

Fig. 3. (a) shows the origin frame, (b) depicts the corrected frame calculated from (a), (c) illustrates the spherically warped frame calculated from (b).

B. Edge Detection and Points Matching

Because the overlapped region is small, we can not match the feature points correctly. Although we got the spherically warped frame, the feature points can not be matched correctly because of the complicated scenes we captured. Under this condition, we propose to detect its edge and then match points according to the edge. We use a matured method which is called holistically-nested edge detection (HED) [23]. After we got the edge of each frame, we mesh every frame into $m \times n$ girds. In the overlapped region, it can be easily to match the points after detecting its edge. Let $\mathbf{a} = [i\ j]^T$ and $\mathbf{a}' = [i'\ j']^T$ be matching points across overlapping frames F_l and F_r. A projective warp or homegraphy aims to map $\mathbf{a}$ to $\mathbf{a}'$ following the relation:

$$\widetilde{\mathbf{a}}' = \mathbf{H}\widetilde{\mathbf{a}} \tag{2}$$

where $\widetilde{\mathbf{a}}$ is $\mathbf{a}$ in homogeneous coordinates, and $\mathbf{H} \in \mathbb{R}^{3\times3}$ defines the homegraphy. In the overlapped region, we can detect its nearest neighbor point as its matching point. Then calculate the homegraphy matrix in each grid. According to the matrix, the points can be projected to the correct region. However, there maybe some grids that have no points, we can calculate the homegraphy matrix according to its four neighbor grids. Let $\mathbf{H}_i$ be the homegraphy matrix in grid i. The $\mathbf{H}_i$ can be calculated as following if there is no points are detected:

$$\mathbf{H}_i = \frac{1}{4}\left(\mathbf{H}_{il} + \mathbf{H}_{ir} + \mathbf{H}_{iu} + \mathbf{H}_{id}\right) \tag{3}$$

where $\mathbf{H}_{il}$, $\mathbf{H}_{ir}$, $\mathbf{H}_{iu}$, $\mathbf{H}_{id}$ denote four neighbor homegraphy matrices of grid i. Fig. 4 illustrates the method we proposed.

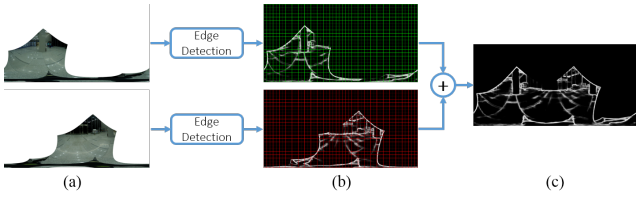

Fig. 4. First the spherically warped frame will be detected by HED [23], and the edge detected frame mesh into $m \times n$ grids, then the overlapped region calculated its homegraphy matrix in grid i. Finally output a new frame combined with two edge detected frames. (a) depicts the spherically warped frame, (b) shows the grids in each frames, (c) shows the new edge produced by (b).

C. Seam Updating and Stitching

Using the method we proposed in Sec. III-A and Sec. III-B, the structure of buildings in the video can be preserved. However, when there is a moving object through the overlapped region, there may exist ghosts in the panoramic video. In the overlapping region, background is stitched using the method we proposed in Sec. III-B, but not surprisingly, foreground objects have ghost artifacts, especially when there is an object moving through the overlapped region. To stitch the two videos seamlessly, we adopt the overlaying strategy as in [24]. Firstly, we search a seam in the overlapping region according to our homegraphy matrices which was calculated in III-B. Secondly, use $\widetilde{g}_{i0}$ as the original gradient of pixel p_i in the present seam, and use $\widetilde{g}_{it}$ as the gradient of pixel p_i in time t. To calculate whether the gradient of pixels have a big change, we use the following rule:

$$C_t = \{p_i | \frac{\widetilde{g}_{it} - \widetilde{g}_{i0}}{\widetilde{g}_{i0}} > \sigma\} \tag{4}$$

where σ is a const, N_{cd} is depended on the total pixel number of the seam. If the total pixel number in C_t is bigger than N_{cd}, we think that there is an object moving through the overlapped region, and then, the seam will be updated. Finally, to stitch the video well, we use graphcut algorithm [25] to stitch the frames which have no object moving through the overlapped region. And the others used linear blending method. Fig 5 shows the method.

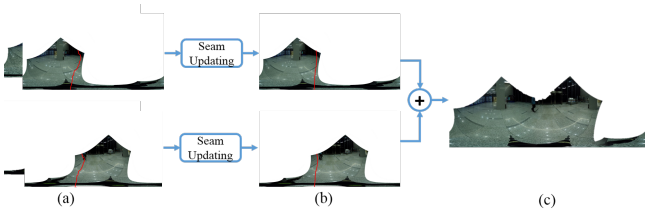

Fig. 5. Two continuous frames are updated seam according the moving the object. And then stitch them together. (a) shows the origin seam in red line, (b) shows the seam updates by algorithm [24] in red line, (c) shows the stitching result.

IV. RESULT

Since there is no publicly available video stitching benchmark data, we evaluate the proposed method on the video which we captured. The video datasets are captured by six fixed cameras (Gopro Hero4 Black) with different views which are synchronized. These videos are both captured by same type cameras and they are 1440p (1920×1440) at 24 fps. Because the overlapped region is very small, lots of method we listed in Sec. II failed to stitch them together. The compared methods include APAP and commercial software AutoPano. The video data used in APAP method in this paper was pre-processed. We compare the result as image stitching on each frame firstly. We randomly select one frame on image stitching. Fig. 6 shows the result of APAP, AutoPano and ours. From Fig. 6 we can see the result of APAP destroys the structure of building severely, such as the pillar and the light. There are severe ghost around pillar and light. As for commercial software, it also can not stitch the image well, such as the pillar have ghosts and the

structure of light breaks from the middle, but our method keeps the structure well, and has less ghost than the others.

For comparison of moving object through overlapped region, we compare the result of APAP, AutoPano and ours. Fig. 7 illustrates frames where an object are moving in the overlapped region and the background remains still. We select three consecutive frames for comparison. In Fig. 7, APAP and AutoPano have severe ghosts around the walking person in the panoramic video. The mans body is distorted in AutoPano method. However, using our proposed method, the man's body is kept well. Fig. 8, Fig. 9 show more comparison of our method and the others.

TABLE I
MOS OF THREE ALGORITHMS

Method	AutoPano	APAP	Ours
MOS	6.45	4.54	**7.43**

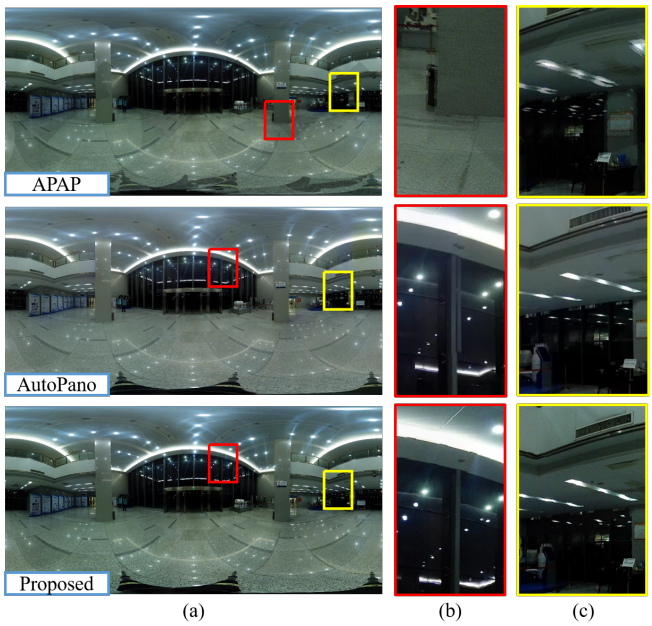

Fig. 6. Comparisons with APAP, AutoPano and our method. From left to right: (a) one of the stitched frames, (b) enlarged views of local detail in red box, (c) enlarged views of local detail in yellow box.

In order to evaluate the performance of our algorithm, we made a user test interface. The interface shows the results of APAP, AutoPano and our method simultaneously. We use Mean opinion score (MOS) to evaluate the panoramic video. We use 1-10, 10 integers to represent the quality of the video, 1 represents the worst quality of the video, 10 represents the best quality of the video, and the method of calculating MOS is as follows:

$$MOS = \frac{\sum_{n=1}^{N} R_n}{N}$$

where N stands for the number of people scoring each method and R_n stands for a user's score for each type of video. We take $N = 250$ in this experiment. In the user study,

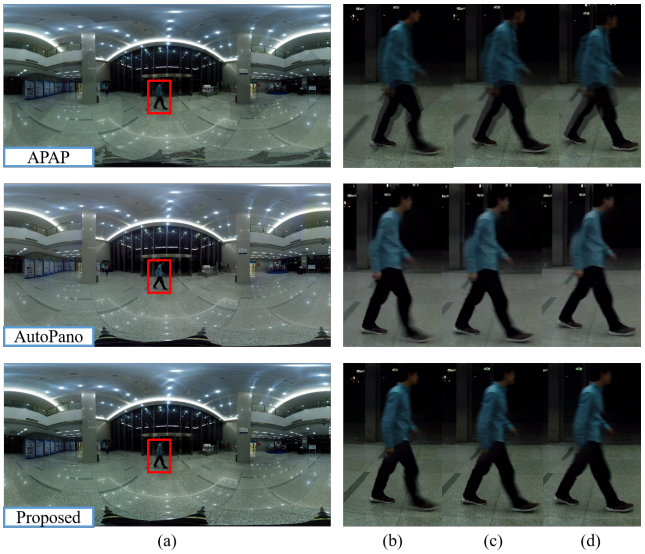

Fig. 7. Comparisons with APAP, AutoPano and our method when there is an object moving in the overlapped region. From left to right: (a) one of the stitched frames, (b) enlarged views of local detail in frame t_1, (c) enlarged views of local detail in frame t_2, (d) enlarged views of local detail in frame t_3.

50 users use interface to score the videos. The videos show on the interface according to the rules in which the video groups were random and the video types were random. And then we calculated MOS for each method. Table I shows the comparison. The MOS of our method is much higher than the others.

V. CONCLUSION

In this paper, we proposed a new method to stitch video which have a small overlapped region. We pre-process the video data, such as calibrating the camera and projecting the video frame into spherical spherical coordinates. We detect the edge of each spherically warped frame and re-calculate homography matrix in every grid. Using the time domain information to produce panoramic videos. Experimental results show that our approach achieves a better panoramic video than state-of-the-art ones. Our algorithm improves image alignment accuracy and reduces artifacts caused by moving objects. In the future, we would like to speed up our algorithm.

REFERENCES

[1] Mai Zheng, Xiaolin Chen, and Li Guo, "Stitching video from webcams," in *International Symposium on Visual Computing*, 2008.

[2] Heng Guo, Shuaicheng Liu, Tong He, Shuyuan Zhu, Bing Zeng, and Moncef Gabbouj, "Joint video stitching and stabilization from moving cameras," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5491–5503, 2016.

[3] Wei Jiang and Jinwei Gu, "Video stitching with spatial-temporal content-preserving warping," in *The IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015.

[4] Yongwei Nie, Tan Su, Zhensong Zhang, Hanqiu Sun, and Guiqing Li, "Dynamic video stitching via shakiness removing," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 164–178, 2018.

[5] http://www.video-stitch.com/studio/.

[6] http://www.kolor.com/autopano/.

[7] R. Szeliski, *Handbook of mathematical models in computer vision*, chapter Image Alignment and Sitching, pp. 273–292, Springer, 2004.

[8] Matthew Brown and David G Lowe, "Automatic panoramic image stitching using invariant features," *International journal of computer vision*, vol. 74, no. 1, pp. 59–73, 2007.

[9] Richard Hartley and Andrew Zisserman, *Multiple view geometry in computer vision*, Cambridge university press, 2003.

[10] Wen-Yan Lin, Siying Liu, Yasuyuki Matsushita, Tian-Tsong Ng, and Loong-Fah Cheong, "Smoothly varying affine stitching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[11] Che-Han Chang, Yoichi Sato, and Yung-Yu Chuang, "Shape-preserving half-projective warps for image stitching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[12] Julio Zaragoza, Tat-Jun Chin, Michael S Brown, and David Suter, "As-projective-as-possible image stitching with moving dlt," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[13] Chung-Ching Lin, Sharathchandra U Pankanti, Karthikeyan Natesan Ramamurthy, and Aleksandr Y Aravkin, "Adaptive as-natural-as-possible image stitching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognitio*, 2015.

[14] Fan Zhang and Feng Liu, "Parallax-tolerant image stitching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[15] Junhong Gao, Yu Li, Tat-Jun Chin, and Michael S Brown, "Seam-driven image stitching," in *Eurographics*, 2013.

[16] Aseem Agarwala, Mira Dontcheva, Maneesh Agrawala, Steven Drucker, Alex Colburn, Brian Curless, David Salesin, and Michael Cohen, "Interactive digital photomontage," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 294–302, 2004.

[17] Junhong Gao, Seon Joo Kim, and Michael S Brown, "Constructing image panoramas using dual-homography warping," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[18] Bin He, Gang Zhao, and Qifang Liu, "Panoramic video stitching in multi-camera surveillance system," in *Image and Vision Computing New Zealand*, 2010.

[19] Federico Perazzi, Alexander Sorkine-Hornung, Henning Zimmer, Peter Kaufmann, Oliver Wang, S Watson, and M Gross, "Panoramic video from unstructured camera arrays," in *Computer Graphics Forum*. Wiley Online Library, 2015, vol. 34, pp. 57–68.

[20] Jing Li, Wei Xu, Jianguo Zhang, Maojun Zhang, Zhengming Wang, and Xuelong Li, "Efficient video stitching based on fast structure deformation," *IEEE Transactions on Cybernetics*, vol. 45, no. 12, pp. 2707–2719, 2015.

[21] Kaimo Lin, Shuaicheng Liu, Loong-Fah Cheong, and Bing Zeng, "Seamless video stitching from hand-held camera inputs," in *Computer Graphics Forum*. Wiley Online Library, 2016, vol. 35, pp. 479–487.

[22] Zhengyou Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, 2000.

[23] Saining Xie and Zhuowen Tu, "Holistically-nested edge detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[24] Botao He and Shaohua Yu, "Parallax-robust surveillance video stitching," *Sensors*, vol. 16, no. 1, pp. 7, 2016.

[25] Yuri Boykov and Vladimir Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.

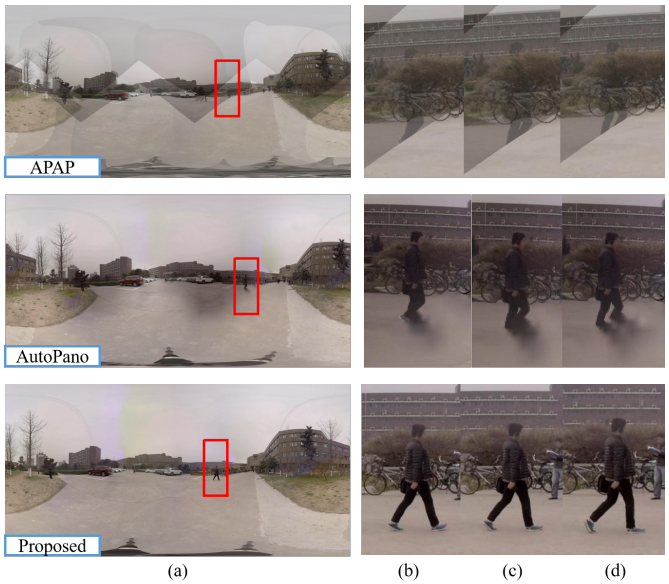

Fig. 8. Comparisons with APAP, AutoPano and our method when there is an object moving in the overlapped region. From left to right: (a) one of the stitched frames, (b) enlarged views of local detail in frame t_1, (c) enlarged views of local detail in frame t_2, (d) enlarged views of local detail in frame t_3.

Fig. 9. Comparisons with APAP, AutoPano and our method when there is an object moving in the overlapped region. From left to right: (a) one of the stitched frames, (b) enlarged views of local detail in frame t_1, (c) enlarged views of local detail in frame t_2, (d) enlarged views of local detail in frame t_3.