

Article

Parallax-Robust Surveillance Video Stitching

Botao He ^{1,*} and Shaohua Yu ²

Received: 7 October 2015; Accepted: 17 December 2015; Published: 25 December 2015

Academic Editor: Gonzalo Pajares Martinsanz

¹ School of Optical and Electronic Information, Huazhong University of Science and Technology, Wuhan 430074, China

² Wuhan Research Institute of Posts and Telecommunications, Wuhan 430074, China; shyu@fhzz.com.cn

* Correspondence: hebotao@hust.edu.cn; Tel.: +86-27-87693908

Abstract: This paper presents a parallax-robust video stitching technique for timely synchronized surveillance video. An efficient two-stage video stitching procedure is proposed in this paper to build wide Field-of-View (FOV) videos for surveillance applications. In the stitching model calculation stage, we develop a layered warping algorithm to align the background scenes, which is location-dependent and turned out to be more robust to parallax than the traditional global projective warping methods. On the selective seam updating stage, we propose a change-detection based optimal seam selection approach to avert ghosting and artifacts caused by moving foregrounds. Experimental results demonstrate that our procedure can efficiently stitch multi-view videos into a wide FOV video output without ghosting and noticeable seams.

Keywords: video stitching; video surveillance; layered warping; parallax

1. Introduction

Image stitching, also called image mosaicing or panorama stitching, has received a great deal of attention in computer vision [1–8]. After decades of development, the fundamentals of image stitching are well studied and relatively mature now. There are many research works on image stitching [1–8], and it is typically solved by estimating a global 2D projective warp to align the input images. A 2D projective warp uses a homography parameterized by 3×3 matrices [1–3,9], which can preserve global image structures, but cannot handle parallax. It is correct only if the scene is planar or if the views differ purely by rotation. However, in practice, such conditions are usually hard to satisfy, thus ghosting and seams yield (see Figure 1). If there is parallax in input images, no global homography exists that can be used to align these images. When a global warp is used to stitch these images, ghosting like Figure 1a would appear. Some advanced image composition techniques such as seam cutting [10–12] can be used to relieve these artifacts. However, if there are moving objects across the seams, another kind of ghosting like Figure 1b would yield.

Previous research indicates that one of the most challenging problems to create seamless and drift-free panoramas is performing a correct image alignment rather than using a simple global projective model and then fix the alignment error [6,8,9]. Thus, some recent image stitching methods focus on using spatially-varying warping algorithms to align the images [6–8], these methods can handle parallax and allow for local deviation to some extent but require more computation.

With wide applications in robotics, industrial inspection, surveillance and navigation, video stitching faces all the problems as image stitching does and can be more challenging due to moving objects in videos. Some researchers tried to build panoramic images by aligning video sequences [13,14], which is panoramic image generation rather than expansion of the FOV of dynamic videos. Other works focus on freely moving devices [15–18], especially mobile devices, which include techniques such as efficient computation of temporal varying homography [15], optimal seam

selection for blending [16], and so on. However, due to complex computation and low resolution, they may not be suitable for surveillance application.

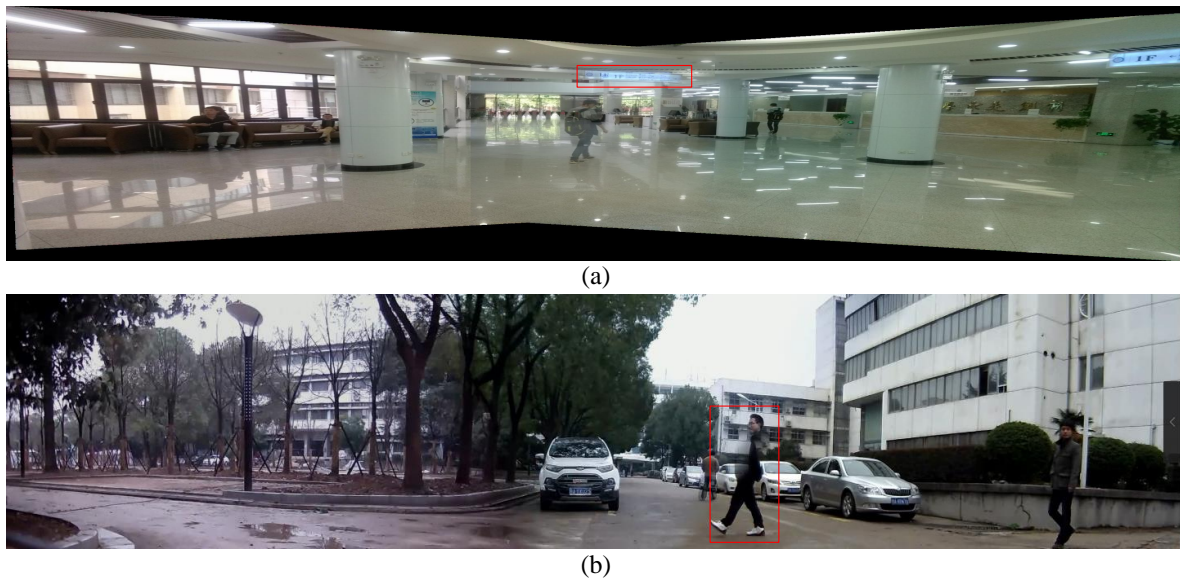


Figure 1. The main reason for ghosting comes from misalignment error. Here we show two types of ghosting in red boxes: (a) ghosting caused by using a global projective warping; (b) ghosting caused by persons moving across seams.

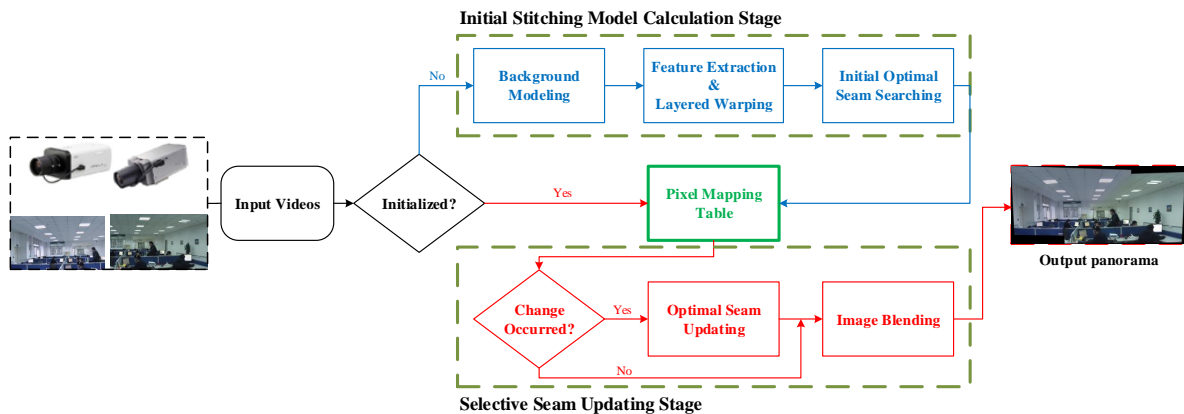


Figure 2. Outline of the proposed video stitching procedure. Our method consists of two stages: initial stitching model calculation stage and selective seam updating stage. In the initial stitching model calculation stage, we first use background modeling algorithm to generate still background of each input source video, then utilize layered warping algorithm to align background images, finally, we perform optimal seam searching and image blending to generate panorama backgrounds. The resulting stitching model is a mapping table in which each entry indicates the correspondence between the pixel index of source images and that of panorama image. To relieve the ghosting effect caused by moving objects, at selective seam updating stage, we perform seam updating according to whether there are objects moving across previous seams or not. Our method reaches a balance between suppressing ghosting artifacts and real-time requirement.

In this paper, we present an efficient parallax-robust surveillance video stitching procedure that combines layered warping and the change-detection based seam updating approach. As the alignment validity of video stitching is still crucial as it is in seamless image stitching, we use a layered warping method for video registration instead of a simple global projective warp.

Xiao *et al.* [19] provides a similar layer-based video registration algorithm, but it aims at aligning a single mission video sequence to the reference image via layer mosaics and region expansion, rather than building a dynamic panoramic video for motion monitoring. Through dividing matched feature pairs into multiple layers (or planes) and local alignment based on these layers, the layered warping method seems to be more robust to parallax. Moreover, the warping data for fixed surveillance videos are stored in an index table for “recycling-use” in subsequent frames to avoid repeated registration and interpolations. This index table is referred to as the initial stitching model in this paper. Aside from layered warping, a local change-detection based seam updating method for overlapping regions is performed to disambiguate the ghosting caused by moving foregrounds. Figure 2 shows the video stitching process presented in this paper.

2. Related Works

Recently, video stitching has drawn a lot of attentions [11,20,21] due to its wide usage in public security. Generally speaking, surveillance video stitching can be regarded as image stitching for every individual frame since the camera positions are always fixed. Different from image stitching technologies, video stitching requires more strict real-time processing ability, and large parallax and dynamic foregrounds must be carefully considered to obtain consistent wide field of view videos. Image stitching is relatively a well studied problem in computer vision [1,9,22]. Several freewares and commercial softwares are also available for performing image stitching, like AutoStitch [23], Microsoft’s Image Compositing Editor [24], and Adobe’s Photoshop CS5 [25] mosaicing feature. However, these approaches all work under the assumption that the input images contains little or no parallax, which implies that the scene is either sufficiently far away from the camera to be considered planar, or that the images have been taken from a camera carefully rotated about its center of projection. This assumption is too strict to be satisfied in real surveillance scenarios. Thus, misalignment artifacts like ghosting or broken image structures will make the final panorama visually unacceptable (see Figure 1).

Some advanced image composition techniques, such as seam cutting [10–12] and blending [26,27], have been employed to reduce the artifacts. However, these methods alone still cannot handle significant parallax. In this paper, we also use seam cutting and blending as the final steps to suppress artifacts. Recent studies on spatially-varying warping are another way out [6,7,28]. As-projective-as-possible (APAP) warps [7] employed local projective warps within the overlapping regions and performed moving direct linear transformation [29] to smoothly extrapolate local projective warps into the non-overlapping regions. Shape-Preserving Half-Projective (SPHP) warp [28] spatially combines a projective transformation and a similarity transformation and has the strengths of both. However, instead of improving alignment accuracy, its main concern is to decrease distortion of non-overlapping area caused by the projective transformation. So even if it introduces APAP [7] into their warp, they cannot solve the problem of structure distortion in the overlapping area in APAP [7]. Gao *et al.* [6] proposed to use a dual homography warp (DHW) algorithm for scenes containing two dominant planes (ground plane and distant plane). While it performs well if the required setting is true, it may fail when there are more than two planes in the source images. Inspired by DHW [6], we propose to use a layered warping algorithm to align the background scenes, which is location-dependent and turned out to be more robust to parallax than the traditional global projective warping methods and more flexible than DHW [6] which can only process images with two planes.

Apart from parallax, moving foregrounds are another reason for ghosting in video stitching. Although we propose to use layered warping to align images as accurate as possible and to utilize seam cutting to composite source images, some artifacts may still exist when objects move across the seam. Liu *et al.* [20] only used the stitching model calculated with first few frames to stitch following frames and didn’t consider the moving foregrounds. In contrast, Tennoe *et al.* [30] and Hu *et al.* [11] update the seam in every frame, which is very time-consuming. To balance between suppressing

the artifacts and the real-time requirement, we propose to first detect changes around the previous seams, and only perform seam update when there are moving objects across seams. Since the price of change detection around seams is much lower than that of updating seams, artifacts caused by moving foregrounds can be suppressed with acceptable time consumption by our method.

3. Initial Stitching Model Calculation

Since our focus is on improving image alignment accuracy and reducing artifacts caused by moving objects, we do not change the conventional pipeline [9] of image stitching with different number of input sources. For ease of illustration, in the following text, we only describe the layered warping algorithm and selective seam updating algorithm with two input videos. In the experiment section, we provide stitching results with both two and more than two input videos.

3.1. Background Image Generation and Feature Extraction

For fixed surveillance cameras, we only perform alignment at the stitching model calculation stage because the computation of temporal varying homography may result in palpable jitter of background scenes in the panoramic video. Background modeling is essential to avoid volatile foreground [31,32]. We take the first N_{gmm} frames of input videos to establish the background frame utilizing the Gaussian Mixture Model (GMM) [33]. SIFT [3,34] features of the background frame is extracted and matched into pairs through Best-Bin-First (BBF) algorithm [35].

3.2. Layered Warping

Inspired by DHW [6], we assume that different objects in a scene usually lie in different depth layers, the objects in the same layer (plane) shall be consistent with each other in spatial transformation. Compared to warping using a global homography or warping using dual homography, layered warping may be more adequate and robust for abundant scenes.

Layer registration. We denote the input images as I_1 and I_2 respectively, and the matching feature pairs as $\mathcal{F} = \{(\mathbf{p}_i^1, \mathbf{p}_i^2)\}_{i=1}^N$, where \mathbf{p}_i^k is the coordinate of the i -th matching point from I_k ($k = 1, 2$). Given the matching feature pairs, we first utilize Random Sample Consensus (RANSAC) algorithm [36] to robustly group the feature pairs into different layers, then estimate the homography for each layer. Denote the consistent matching feature pairs of layer k as \mathcal{L}_k , the number of matches in \mathcal{L}_k as $|\mathcal{L}_k|$ and its corresponding homography as \mathbf{H}_k . To guarantee the grouped layer to be representative, we introduce a threshold N_{min} which denotes the minimum number of matching pairs in $|\mathcal{L}_k|$. Layers whose number of matching pairs is smaller than N_{min} are simply dropped. The detailed layer registration process is presented in Algorithm 1.

Algorithm 1 Layer registration utilizing multiple-layer RANSAC

Input: Initial pair set $\mathcal{F}_1 = \mathcal{F}$, threshold N_{min} and iteration index $k = 0$;

Output: Each layer's matching pair set \mathcal{L}_k and its corresponding homography \mathbf{H}_k ;

repeat

$k \leftarrow k + 1$

 RANSAC in pair set \mathcal{F}_k for model $\mathbf{p}_{k_n}^1 \times \mathbf{H} \mathbf{p}_{k_n}^2 = 0$, where $(\mathbf{p}_{k_n}^1, \mathbf{p}_{k_n}^2) \in \mathcal{F}_k$;

 Divide outliers \mathbf{V}_{out} and inliers \mathbf{V}_{in} according to \mathbf{H} ;

if $|\mathbf{V}_{in}| \geq N_{min}$ **then**

 Set matching pair set of the k -th layer as $\mathcal{L}_k = \mathbf{V}_{in}$;

 Set homography of the k -th layer as $\mathbf{H}_k = \mathbf{H}$;

end if

 Set the pair set of next iteration as $\mathcal{F}_{k+1} = \mathbf{V}_{out}$;

until $|\mathcal{F}_{k+1}| < N_{min}$

Through the layer registration process, we divide the matched feature pairs to multiple layers, each of which contains a set of feature pairs that are consistent with a common homography. Figure 3 shows two examples of layer registration, in which feature points are illustrated as circles and points with the same color are from the same layer. From Figure 3 we can see that the grouped matching pairs of the same layer are almost from the same plane or with the same depth, which is in accordance with our expectation.

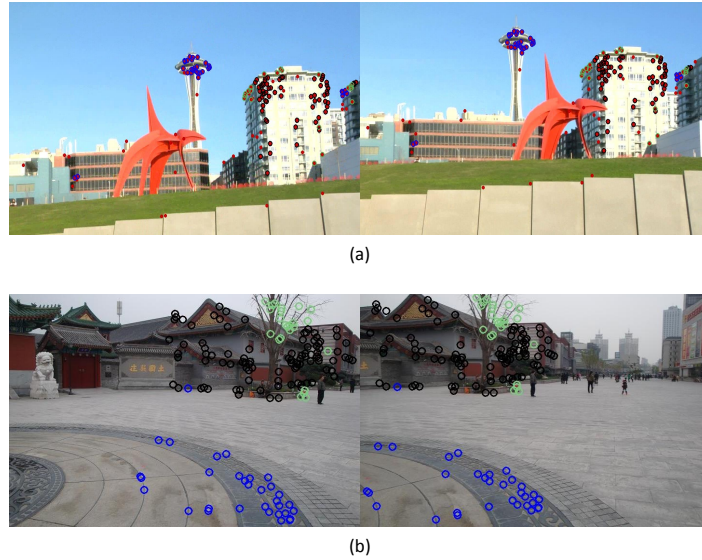


Figure 3. Examples of matched feature points in multiple layers, different color indicate different layers: (a) multiple-layer matched feature pairs of images with large parallax; (b) multiple-layer matched feature pairs of images with a distant plane and ground plane

Local alignment. To simplify calculations of local alignment, the source image is divided into $M \times N$ grids. Since a feature point is not usually coincident with any grid vertex, we use the distances between the grid center and its nearest neighbors in different layers to vote on the warp of the due grid. A grid g_j is represented by its center point \mathbf{c}_j . The homography of the grid g_j , denoted by \mathbf{H}_j^* , is computed by fusing \mathbf{H}_k of multiple layers using a weight w_k^j by Equation (1)

$$\mathbf{H}_j^* = \sum_k w_k^j \mathbf{H}_k \quad (1)$$

where $w_k^j = a_k^j / \sum_i a_i^j$ and a_k^j is a position dependent Gaussian weight:

$$a_k^j = \exp\left(-\frac{\|\mathbf{c}_j - \mathbf{p}_k^*\|^2}{\sigma^2}\right) \quad (2)$$

Here \mathbf{p}_k^* denotes the nearest neighboring feature point of \mathbf{c}_j in layer k and σ is a scale constant.

After deriving the local homography for each grid, the target pixel position \mathbf{p}' in the reference image for the source pixel at position \mathbf{p} in grid g_j can be easily obtained by Equation (3):

$$\mathbf{p}' = \mathbf{H}_j^* \mathbf{p} \quad (3)$$

This process is referred to as forward mapping [9]. To accelerate the computation, we only perform forward mapping once, and store the correspondence between source pixel positions and target pixel positions in the pixel mapping table. This pixel mapping table is exactly the stitching

model. The index tables are stored so that the warped image can be obtained by looking up each corresponding pixel in the source image instead of repeated transformation when new frames arrive. Figure 4a,b shows the panorama images with global projective warp and layered warp respectively. It is clear that the building in Figure 4b is better aligned than that in Figure 4a.

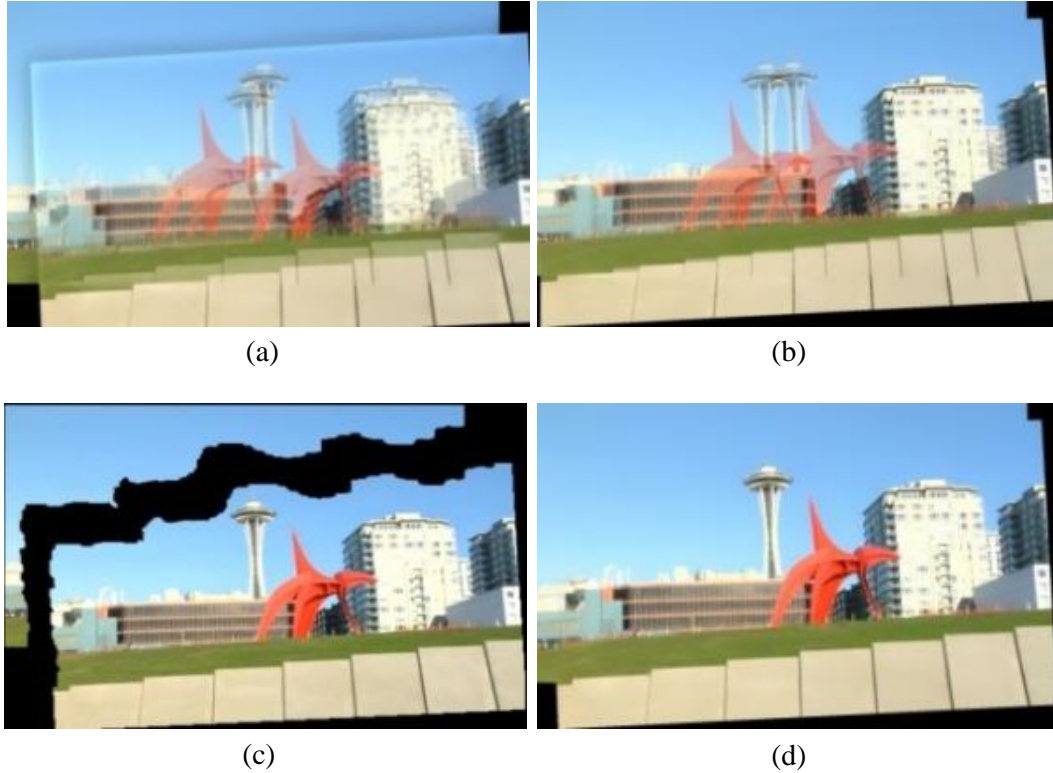


Figure 4. Examples of panoramas with (a) global projective warp and (b) layered warp, and seam selection results in (c). (d) is the final fused image according to selected seam in (c). The seam selection result (c) is based on stitched image (b) with layered warp.

3.3. Optimal Seam Cutting

Though layered warping is parallax-robust, it is applied only in the stitching model calculation stage on the extracted background scene. We perform the seam selection method to disambiguate the ghosting caused by the moving foreground.

Optimal seam searching method, also called optimal seam selection [5] or seam cutting, is to search for an optimal seam path which is a pixel-formed continuous curve in the overlapping area to connect pairwise warped images.

The seam should neither introduce inconsistent scene elements nor intensity differences. Therefore, two criteria are applied in this paper to form the difference map of overlaps: the intensity energy E_{ij}^C and gradient energy E_{ij}^g which are defined as:

$$\begin{cases} E_{ij}^C &= \frac{\|I_A(i,j) - I_B(i,j)\|}{\max(I_A(i,j), I_B(i,j))} \\ E_{ij}^g &= \|\nabla I_A(i,j) - \nabla I_B(i,j)\|^2 \end{cases} \quad (4)$$

Here $I_A(i,j)$, $I_B(i,j)$, $\nabla I_A(i,j)$ and $\nabla I_B(i,j)$ are the intensity and gradient of pixel (i,j) in image A and B respectively. Finding the optimal stitching seam is an energy minimization problem (see Equation (5)) and can be converted to a binary Markov Random Field (MRF) labeling problem [37]:

$$\arg \min_{ij} \sum_{ij} (E_{ij}^C + \lambda E_{ij}^g) \quad (5)$$

To accelerate the computing process, the warped images and background images are down sampled before seam selection and restored to the original size after seam selection in our procedure. Figure 4c and d show an example of seam selection. From Figure 4c we can see, the selected seam mainly crosses flat areas with little gradients or intensity differences, thus the resulted panorama is visually consistent with no noticeable ghosting.

4. Selective Seam Updating

After initial stitching model calculation, the videos are overall pre-aligned. However, for moving foregrounds, the previous seam may lose its optimality or even miss information. Since the seam cutting algorithm requires complex computation even on the down-sampled frames, it is difficult to be used to update video frames in real-time. So we perform a change detection based seam updating method instead of real-time seam selection.

4.1. Change Detection around Previous Seams

First of all, we resize the new warped frames to a smaller scale as we did in the optimal seam selection section. Even if the images have been scaled down, direct calculation of the gradient of the two images and evaluation of the change may be expensive. However, we observed that compared with changes in non-overlapping area, those in the overlapping area are more likely to violate the optimal seam. Furthermore, only changes across the optimal seam may result in the failure of it (see Figure 5), which can be measured by gradient difference.

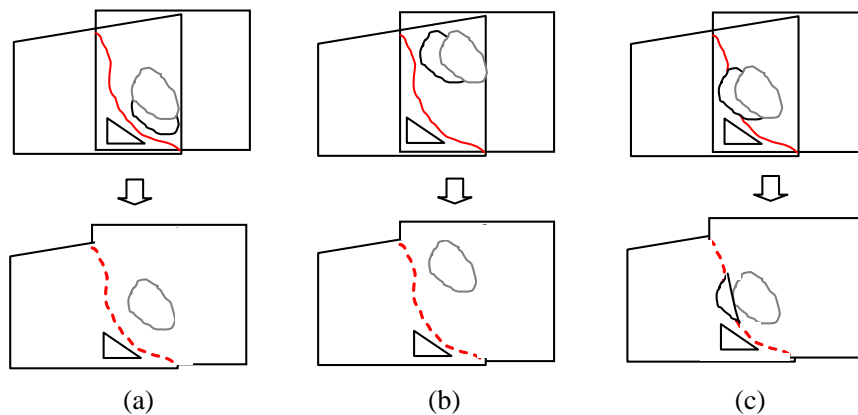


Figure 5. (a) The original aligned frame with a stitching line; (b) newly aligned frame with changes not cross the line; (c) frame with changes cross the line.

As an optimal seam is searched, the original gradient value set \tilde{G}_0 of the seam is stored. We set \tilde{g}_{i0} as the original gradient of pixel p_i in the present seam, and \tilde{g}_{it} as the gradient in time t . To calculate the number of pixels that have large gradient variations, we use the following rule to judge if changes occurred at pixel p_i or not:

$$C_t = \{p_i | \frac{\tilde{g}_{it} - \tilde{g}_{i0}}{\tilde{g}_{i0}} > \delta\} \quad (6)$$

If the total pixel number in C_t is bigger than N_{cd} , we consider that there are new moving objects in the overlapping area and the optimal seam shall be updated. In Equation (6), δ is a constant chosen empirically, whereas N_{cd} depends on \tilde{N} , which is the total pixel number of the seam. We set it as $0.3\tilde{N}$ in our experiments.

4.2. Seam Updating

For each new warped frame, change detection, as described in the previous section, is performed to see if an alteration of the seam is needed, if so, we select a new seam by the seam selection algorithm presented in Section 3.3, otherwise, we continue to use the previous seam and move on to the image blending process.

4.3. Blending

Seam cutting can eliminate ghosting, but it provides images without overlaps which may result in noticeable seams. We expand the seamed image with a spherical dilating kernel, and use a simple weighted linear blending method [9] to blend the images.

5. Experiments

To evaluate the performance of our video stitching procedure, we conduct some experiments on both still images with parallax and actual surveillance videos.

5.1. Experimental Settings

There are several empirical parameters which should be manually tuned for different cases. At the background modeling stage, we use the initial 20 frames to construct a GMM model with 5 components for each pixel. The frame number may be set larger if the scenes are more cluttered. In our experiment, N_{min} is set as 12, which is the minimum number of matching pairs for each layer, and the source 720P image is divided into 80×45 grids for layered warping. We provide stitching results with both two, three and four channels as input.

5.2. Stitching Still Images

To evaluate the proposed layered warp algorithm, we first conduct experiments on still images with parallax and compare our results with other image stitching methods in Figures 6 and 7.

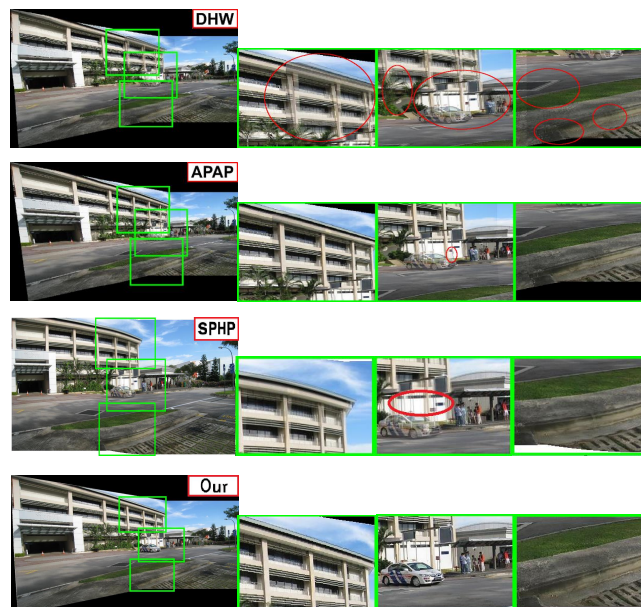


Figure 6. Comparisons among dual homography warp (DHW) [6], as-projective-as-possible (APAP) [18], Shape-Preserving Half-Projective (SPHP) [28] and Our algorithm, Red circles highlight errors.

DHW [6] only considers two layers (a distant plane and a ground plane), so it fails when the scene has multiple depth layers or contains complex objects. Our algorithm is more adequate and robust than DHW for abundant scenes (see Figure 6). Figure 7 shows the comparison among our algorithm and some state-of-art algorithms on a tough scene from [8]. As APAP tries to align two images over the whole overlapping region, it distorts the salient image structure, such as the pillar indicated by the red rectangle. SPHP also fails when the overlapping area covers most of the image and there is large parallax in it (see Figure 7c). From Figures 6 and 7 we can see, the stitching results of our algorithm contain no noticeable distortions and are more visually acceptable than other methods.

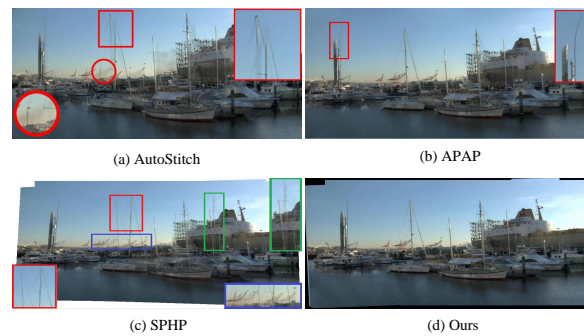


Figure 7. Comparisons among various stitching methods. colored rectangles and their zoom-in rectangles illustrate the false misalignment.

5.3. Stitching Fixed Surveillance Videos

To evaluate the effectiveness of selective seam updating strategy, we perform extensive experiments on fixed surveillance videos, and compare our results with that of without seam updating. The results are shown in Figure 8, from which we can see, when we do not perform seam updating in the video stitching stage, some obvious artifacts caused by moving objects would make the stitched video visually unacceptable. The comparison results indicate that the change detection based seam updating approach is rather helpful to avert ghosting and mis-alignments caused by moving targets, especially when there are moving objects across previous seams.

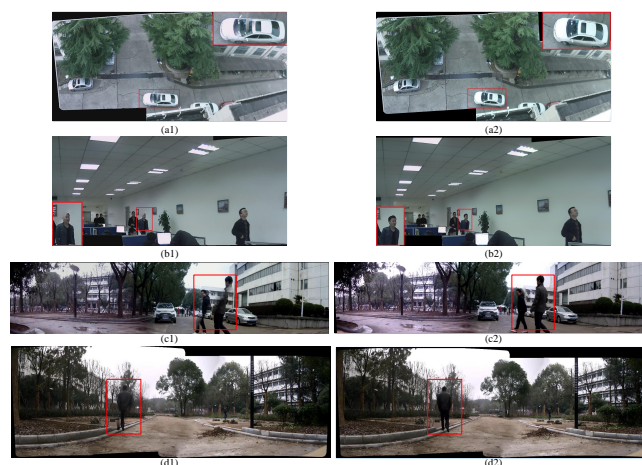


Figure 8. Frames taken from the output Field-of-View (FOV) videos: (a1–d1) are taken from the output videos using initialized seam without updating. (a2–d2) are taken from the output video using change-detection based seam updating approach. (a1), (a2), (b1) and (b2) are the stitching results of two channels of 720P videos. (c1) and (c2) are the stitching results of three channels of 720P videos. While for (d1) and (d2), they are the stitching results of four 720P videos.

5.4. Time Analysis

Apart from suppressing ghosting caused by parallax and moving targets, the real-time requirement of video stitching task should also be seriously considered when developing algorithms. To evaluate the speed of the proposed method, we take two videos as input and test the time consumption for different video resolutions. To make a fair comparison, the speed of stitching with temporal varying homography, layered warping without selective seam updating and layered warping with selective seam updating are all listed in Table 1. These experiments are all conducted on a PC with Intel Core i3 2.33 GHz CPU with two cores and 2GB RAM. All tested algorithms are implemented with C++.

From Table 1 we can see, as the index table has already been established according to the registration result in the stitching model calculation stage, frames can be projected to the panoramic frame by directly indexing instead of alignment and mapping for every new frame, thus the stitching efficiency of the proposed method is greatly improved. Table 1 also shows that the selective seam updating process slows down the frame stitching process to some extent.

Table 1. Comparisons of the speed among different video stitching methods on input-videos with different resolutions.

Resolution	Stitching with Temporal Varying Homography	Proposed Algorithm without Seam Updating	Proposed Algorithm with Seam Updating
720p: 1280 × 720	4.764 s	0.051 s	0.083 s
480P: 720 × 480	2.326 s	0.035 s	0.045 s
CIF: 352 × 288	1.025 s	0.021 s	0.032 s

6. Conclusions

This paper presents an efficient parallax-robust stitching algorithm for fixed surveillance videos. The proposed method consists of two parts: alignment work done at the stitching model calculation stage and change detection based frame updating at the video stitching stage. The algorithm uses layered warping method to pre-align the background scene which is robust to scene parallax. As for each new frame, an efficient change detection based seam updating method is adopted to avert ghosting and artifacts caused by moving foregrounds. Thus, the algorithm can provide good stitching performance with no ghosting and artifacts for dynamic scenes efficiently.

Author Contributions: The work was realized with the collaboration of all of the authors. Botao He and Shaohua Yu contributed to the algorithm design and code implementation. Botao He contributed to the data collection and baseline algorithm implementation and drafted the early version of the manuscript. Shaohua Yu organized the work and provided the funding, supervised the research and critically reviewed the draft of the paper. All authors discussed the results and implications, commented on the manuscript at all stages and approved the final version.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Brown, M.; Lowe, D. Recognising Panoramas. In Proceedings of the Ninth IEEE International Conference on Computer Vision, Nice, France, 13–16 October 2003; pp. 1218–1225.
2. Szeliski, R. Video mosaics for virtual environments. *IEEE Comput. Graph. Appl.* **1996**, *16*, 22–30.
3. Eden, A.; Uyttendaele, M.; Szeliski, R. Seamless image stitching of scenes with large motions and exposure differences. *Comput. Vis. Pattern Recognit.* **2006**, doi:10.1109/CVPR.2006.268.
4. Levin, A.; Zomet, A.; Peleg, S.; Weiss, Y. *Seamless Image Stitching in the Gradient Domain*; Springer Berlin Heidelberg: Berlin, Germany, 2004; pp. 377–389.
5. Mills, A.; Dudek, G. Image stitching with dynamic elements. *Image Vis. Comput.* **2009**, *27*, 1593–1602.

6. Gao, J.; Kim, S.J.; Brown, M.S. Constructing Image Panoramas Using Dual-Homography Warping. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 20–25 June 2011; pp. 49–56.
7. Zaragoza, J.; Chin, T.J.; Tran, Q.H.; Brown, M.S.; Suter, D. As-projective-as-possible image stitching with moving DLT. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 1285–1298.
8. Zhang, F.; Liu, F. Parallax-Tolerant Image Stitching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 3262–3269.
9. Szeliski, R. Image alignment and stitching: A tutorial. *Found. Trends Comput. Graph. Vis.* **2006**, *2*, 1–104.
10. Agarwala, A.; Dontcheva, M.; Agrawala, M.; Drucker, S.; Colburn, A.; Curless, B.; Salesin, D.; Cohen, M. Interactive digital photomontage. *ACM Trans. Graph.* **2004**, *23*, 294–302.
11. Hu, J.; Zhang, D.Q.; Yu, H.; Chen, C.W. Discontinuous Seam Cutting for Enhanced Video Stitching. In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), 29 June–3 July 2015; pp. 1–6.
12. Kwatra, V.; Schödl, A.; Essa, I.; Turk, G.; Bobick, A. Graphcut textures: Image and video synthesis using graph cuts. *ACM Trans. Graph.* **2003**, *22*, 277–286.
13. Steedly, D.; Pal, C.; Szeliski, R. Efficiently Registering Video into Panoramic Mosaics. In Proceedings of the Tenth IEEE International Conference on Computer Vision, Beijing, China, 17–21 October 2005; pp. 1300–1307.
14. Hsu, C.T.; Tsan, Y.C. Mosaics of video sequences with moving objects. *Signal Process. Image Comm.* **2004**, *19*, 81–98.
15. El-Saban, M.; Izz, M.; Kaheel, A. Fast Stitching of Videos Captured from Freely Moving Devices by Exploiting Temporal Redundancy. In Proceedings of the 17th IEEE International Conference on Image Processing (ICIP), Hong Kong, China, 26–29 September 2010; pp. 1193–1196.
16. El-Saban, M.; Izz, M.; Kaheel, A.; Refaat, M. Improved Optimal Seam Selection Blending for Fast Video Stitching of Videos Captured from Freely Moving Devices. In Proceedings of 18th IEEE International Conference on the Image Processing (ICIP), Brussels, Belgium, 11–14 September 2011; pp. 1481–1484.
17. Okumura, K.I.; Raut, S.; Gu, Q.; Aoyama, T.; Takaki, T.; Ishii, I. Real-Time Feature-Based Video Mosaicing at 500 fps. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 3–7 November 2013; pp. 2665–2670.
18. Au, A.; Liang, J. Ztitch: A Mobile Phone Application for Immersive Panorama Creation, Navigation, and Social Sharing. In Proceedings of the IEEE 14th International Workshop on Multimedia Signal (MMSP), Banff, AB, USA, 17–19 September 2012; pp. 13–18.
19. Xiao, J.; Shah, M. Layer-based video registration. *Mach. Vis. Appl.* **2005**, *16*, 75–84.
20. Liu, H.; Tang, C.; Wu, S.; Wang, H. Real-Time Video Surveillance for Large Scenes. In Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 9–11 November 2011; pp. 1–4.
21. Zeng, W.; Zhang, H. Depth Adaptive Video Stitching. In Proceedings of the Eighth IEEE/ACIS International Conference on Computer and Information Science, Shanghai, China, 1–3 June 2009; pp. 1100–1105.
22. Brown, M.; Lowe, D.G. Automatic panoramic image stitching using invariant features. *Int. J. Comput. Vis.* **2007**, *74*, 59–73.
23. AutoStitch. Available online: <http://cvlab.epfl.ch/~brown/autostitch/autostitch.html> (accessed on 18 December 2015).
24. Microsoft Image Composite Editor. Available online: <http://research.microsoft.com/en-us/um/redmond/groups/ivm/ICE/> (accessed on 18 December 2015).
25. Adobe Photoshop CS5. Available online: <http://www.adobe.com/products/photoshop> (accessed on 18 December 2015).
26. Burt, P.J.; Adelson, E.H. A multiresolution spline with application to image mosaics. *ACM Trans. Graph.* **1983**, *2*, 217–236.
27. Pérez, P.; Gangnet, M.; Blake, A. Poisson image editing. *ACM Trans. Graph.* **2003**, *22*, 313–318.
28. Chang, C.H.; Sato, Y.; Chuang, Y.Y. Shape-Preserving Half-Projective Warps for Image Stitching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 3254–3261.

29. Alexa, M.; Behr, J.; Cohen-Or, D.; Fleishman, S.; Levin, D.; Silva, C.T. Computing and rendering point set surfaces. *IEEE Trans. Vis. Comput. Graph.* **2003**, *9*, 3–15.
30. Tennoe, M.; Helgedagsrud, E.; Næss, M.; Alstad, H.K.; Stensland, H.K.; Gaddam, V.R.; Johansen, D.; Griwodz, C.; Halvorsen, P. Efficient Implementation and Processing of a Real-Time Panorama Video Pipeline. In Proceedings of the IEEE International Symposium on Multimedia (ISM), Anaheim, CA, USA, 9–11 December 2013; pp. 76–83.
31. Kumar, P.; Dick, A.; Brooks, M.J. Integrated Bayesian Multi-Cue Tracker for Objects Observed from Moving Cameras. In Proceedings of the 23rd International Conference on Image and Vision Computing New Zealand, Christchurch, New Zealand, 26–28 November 2008; pp. 1–6.
32. Kumar, P.; Dick, A.; Brooks, M.J. Multiple Target Tracking with an Efficient Compact Colour Correlogram. In Proceedings of the 10th International Conference on Control, Automation, Robotics and Vision, Hanoi, Vietnam, 17–20 December 2008; pp. 699–704.
33. Zivkovic, Z. Improved adaptive Gaussian mixture model for background subtraction. *Proc. Int. Conf. ICPR Pattern Recognit.* **2004**, *2*, 28–31.
34. Kumar, P.; Ranganath, S.; Huang, W. Queue Based Fast Background Modelling and Fast Hysteresis Thresholding for Better Foreground Segmentation. In Proceedings of the Fourth Pacific Rim Conference on Multimedia Information, Communications and Signal, Singapore, 15–18 December 2003; pp. 743–747.
35. Beis, J.S.; Lowe, D.G. Shape Indexing Using Approximate Nearest-Neighbour Search in High-Dimensional Spaces. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, Territory, 17–19 June 1997; pp. 1000–1006.
36. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. ACM* **1981**, *24*, 381–395.
37. Szeliski, R.; Zabih, R.; Scharstein, D.; Veksler, O.; Kolmogorov, V.; Agarwala, A.; Tappen, M.; Rother, C. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 1068–1080.



© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).