Efficient Video Stitching Based on Fast Structure Deformation

Jing Li, Wei Xu, Jianguo Zhang, Maojun Zhang, Zhengming Wang, and Xuelong Li, *Fellow, IEEE*

***Abstract*—In computer vision, video stitching is a very challenging problem. In this paper, we proposed an efficient and effective wide-view video stitching method based on fast structure deformation that is capable of simultaneously achieving quality stitching and computational efficiency. For a group of synchronized frames, firstly, an effective double-seam selection scheme is designed to search two distinct but structurally corresponding seams in the two original images. The seam location of the previous frame is further considered to preserve the interframe consistency. Secondly, along the double seams, 1-D feature detection and matching is performed to capture the structural relationship between the two adjacent views. Thirdly, after feature matching, we propose an efficient algorithm to linearly propagate the deformation vectors to eliminate structure misalignment. At last, image intensity misalignment is corrected by rapid gradient fusion based on the successive over relaxation iteration (SORI) solver. A principled solution to the initialization of the SORI significantly reduced the number of iterations required. We have compared favorably our method with seven state-of-the-art image and video stitching algorithms as well as traditional ones. Experimental results show that our method outperforms the existing ones compared in terms of overall stitching quality and computational efficiency.**

***Index Terms*—Computational efficiency, computer vision, intensity misalignment, machine learning, structure misalignment, video stitching.**

I. Introduction

IMAGE stitching is a very important problem for computer vision, and aims to combine multiple images with overlapped regions to produce a seamless and high-resolution image [1]–[3]. However, video stitching is far beyond, which means stitching a set of synchronized video frames sequentially across different views to generate a wide field-of-view and high-resolution video. Video stitching should address the important practical challenges of real-time processing and interframe consistency. Unfortunately, most of state-of-the-art image stitching algorithms cannot be directly applied to video stitching due to their main focus on stitching quality. Video stitching has potential applications in surveillance [4], investigation, monitoring [5], scene classification [6], action recognition [7], and etc.

Automatic image stitching has been an active research topic for decades. Registration is one of the key steps and numerous methods have been proposed in this area. Existing registration methods could be broadly classified into two categories: 1) local feature-based matching [8], [9] and 2) direct alignment based on pixels intensities or colors [10]. Though considerable progress has been achieved in this area to date and quite a few commercial softwares are available [11]–[13], tackling misalignment is still one of the major research challenges nowadays. Misalignment is often caused by the limitation of camera motion model, registration error, and multiviewpoint [1], [9], [14], which could lead to the structure and intensity inconsistency (often called visible artifacts) in the overlapped region. Fig. 1 shows an example of such misalignments in the stitching progress. For one pair of registered images [Fig. 1(a) and (b)] to be stitched, a simple pasting of left part of Fig. 1(a) and right part of Fig. 1(b) is shown in Fig. 1(d), where there are severe visible artifacts. The intensity misalignment of input images is caused by the difference of exposure parameters of cameras. Moreover, objects' displacement or difference in the apparent position caused by different view points of cameras (called parallax) mainly influences the structure consistency of input images, and then generates the discontinuity and mismatch between object boundaries, which are called structure misalignment, as shown in Fig. 1(c). Stitching algorithms should eliminate both intensity and structure misalignment properly to achieve natural transition between the stitched images.

Assuming all input images have been registered, stitching methods based on optimal seam and smooth transition firstly search for an optimal division line (denoted as seam) in the overlapped region of input images and then smoothly blend the pixels across the selected seam. This kind of strategy could eliminate intensity misalignment between input images, as shown in Fig. 1(e) and (f). But for structure misalignment in the overlapped region, these types of methods could only blur the discontinuous edges, thus resulting in an undesired blurred edge region, though less perceivable. Structure deformation [15], [16] deals with camera-model-independent spatial transformation of an image by considering structure constraints. Therefore, structure consistency of the image content around the seam could be

Manuscript received June 23, 2014; revised October 11, 2014; accepted December 3, 2014. This work was supported in part by the National Natural Science Foundation of China under Grant 61403403 and Grant 61125106, in part by the National University of Defense and Technology under Project JC13-05-02, and in part by the Shaanxi Key Innovation Team of Science and Technology under Grant 2012KCT-04. This paper was recommended by Associate Editor L. Shao.

J. Li, W. Xu, and M. Zhang are with the College of Information System and Management, National University of Defense Technology, Changsha 410073, China (e-mail: jingli@nudt.edu.cn; weixu@nudt.edu.cn).

J. Zhang is with the School of Computing, University of Dundee, Dundee DD1 4HN, U.K. (e-mail: jgzhang@computing.dundee.ac.uk).

Z. Wang is with the College of Science, National University of Defense Technology, Changsha 410073, China.

X. Li is with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, Shaanxi, P. R. China (e-mail: xuelong_li@opt.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2014.2381774

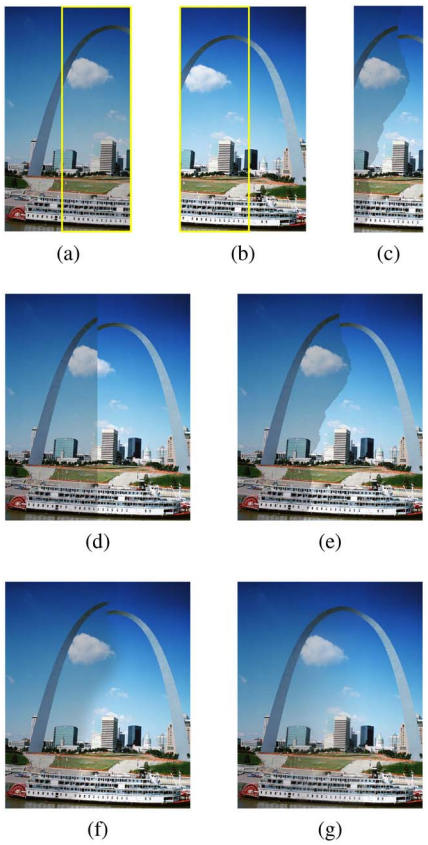


Fig. 1. Image stitching. (a) and (b) Two registered input images with distinct intensity and structure misalignment between them. (c) Simple pasting of the overlapped region of the input images. (d) Simple pasting result of input images. (e) Stitching result of pasting along optimal seam. (f) Stitching result generated by Hugin [12], which is a popular image stitching software. (g) Stitching result of our method.

well preserved. However, the state-of-the-art model of structure deformation proposed in [16] suffers from high computational cost due to its complexity, which makes it less applicable for real-time stitching.

In this paper, we concentrate on video stitching, and the main contributions are fourfolds.

1) We proposed an effective and efficient video stitching approach based on fast structure deformation, which is capable of stitching videos with high quality and low computational cost.
2) We presented a double-seam selection model by taking into account both spatial and temporal consistency, which has linear complexity.
3) We theoretically show that the deformation model could be formulated into an inverse linear interpolation problem, which is very efficient.
4) We proposed a principled solution to successive over relaxation iteration (SORI) solver for efficient gradient fusion. Fig. 1(g) shows an example of the stitching result using our method, where both intensity and structure misalignment are well eliminated.

Although our method is motivated by Jia and Tang [16], we would like to emphasize that this paper differs from theirs significantly in the following aspects.

1) This paper focuses on video stitching, and addresses the challenges of quality stitching and low computational cost simultaneously, while Jia and Tang [16] concentrated on image stitching, and failed to address the requirements of low-computational cost, which we believe is a key requirement for video stitching.
2) For double seam selection, our model takes into account both spatial and interframe consistency (temporal) constraints, which could effectively propagate seam candidates in video stitching. Combined with the well-known dynamic programming, the computational complexity of our model is linearly proportional to the area of the overlapped region. While the cost function of Jia and Tang's [16] method only considered gradient smoothness and similarity optimized by graph cut [17] with nonlinear complexity.
3) We formulate the problem of deformation quantization and propagation as an inverse linear interpolation of deformation vectors, while the method of Jia and Tang [16] relies on minimizing integrated energy functions. We show that our method could achieve perceptually the same stitching results.
4) We presented a fast gradient fusion method by introducing a principled solution to SORI initialization, while Jia and Tang [16] simply relied on Poisson equations for gradient fusion [18] and did not provide any further insights. Our method is verified in the experimental section.

The rest of this paper is organized as follows. Previous paper on image or video stitching and structure deformation is summarized in Section II. Section III describes in detail our methodology for video stitching based on fast structure deformation. Experimental results together with comparison with other stitching methods are presented in Section IV. Section V concludes this paper.

II. Related Work

In past few decades, considerable efforts have been devoted toward seamless image and video stitching. In this section, we review previous paper on image stitching, video stitching, and structure deformation that is most relevant to our method.

A. Image Stitching

In early age, the research on image stitching is mainly focused on two aspects: 1) optimal seam selection and 2) smooth transition. Optimal seam is technically defined as the partition line with smallest difference in an overlapped region. Dynamic programming was deployed to search the optimal seam in rectangular overlapped regions of texture images in [19]. Optimal seam selection based on dynamic programming has also been applied in virtual microscopy [20], mobile panoramic mosaic [21], and airborne image stitching [22]. Another widely used optimal seam selection algorithm is graph cut [17]. An image and video composition scheme using graph cut was designed in [23]. Compared with dynamic programming, partitions with graph cut normally look better visually. However, it is computationally more expensive. A heuristic seam selection

algorithm was proposed in [24] to stitch images with moving objects in the overlapped region. Recently, an interactive seam selection model named Panorama weaving [25] was proposed to achieve fast and flexible seam processing. However, its interactive property makes it unsuitable for automatic video stitching.

Smooth transition algorithms are used to minimize visual difference of overlapped images. A smoothing function was defined in [26] to remove artificial edges between input images. Pyramid blending [27], [28] applies different masks to various frequency bands of images, where wide weighting range is used in low frequency bands and narrow range is for high frequency bands. Gradient domain fusion [18] converts seamless image stitching to an optimization problem in gradient domain. In [29], gradient domain fusion was achieved by solving Poisson equation. Different objective functions of gradient domain fusion were analyzed and compared in [30] and [31]. A quad-tree storage organization was used to accelerate gradient domain fusion in [32]. In order to cope with severe inconsistency of source images, Darabi *et al.* [33] proposed a patch-based synthesis method called image melding that produced a gradual transition between sources without sacrificing texture sharpness.

Images produced by purely optimal-seam based stitching method will be ideally seamless unless there is no difference between input images along the seam. However, such kind of ideal seams usually do not exist in practice. Hence the visual artificial edges are often inevitable, as shown in Fig. 2(d). Similarly, for stitching method merely based on smooth transition, ghost images may appear in the stitched image [34], as shown in Fig. 2(c). Therefore, the two kinds of approaches are complementary and usually combined in application, i.e., to smooth the transition around the optimal seam. The results of pyramid blending and gradient domain fusion based on optimal seam are shown in Fig. 2(e) and (f). These methods can eliminate intensity misalignment between input images, but do not perform well on correcting the structure misalignment.

B. Video Stitching

Compared with image stitching, video stitching needs to address the challenge of computationally efficiency, especially in applications such as real-time video surveillance, where the processing speed should be fast enough to enable real-time streaming. In early age, the hardware was not advanced enough to process video in real time, hence the research was mainly focused on post processing. The frames containing different parts of the scene were synthesized to generate a high resolution panoramic mosaic in [35], where the scanning of the scene was organized to be in a particular order. The establishment of the panoramic mosaic was then made independent with the scanner manner by motion estimation and feature matching [36], [37]. The mosaicking process could be accelerated by handling different kinds of frames respectively and compressing frames with matching relationships [38].

Nowadays, the computing power of modern processors has uplifted the bar of hardware constraints and makes real-time video processing possible. For camera group with fixed relative location [39], video stitching can be divided into two steps: 1) initialization and 2) stitching. In the initialization phase, the pose and distortion of cameras were estimated through feature detection and matching. In the stitching phase, each group of frames was projected into the same observation plane, and then blended with a nonlinear smoothing function. Better fusion effect could be achieved by using pyramid blending in stitching phase [40]. This two-step approach has already been used in video synthesis commercial softwares such as Autopano [41]. The neighboring frames of a video are correlated, therefore, sufficient usage of previous frame could accelerate the stitching of current frame [42], [43]. Though the methods mentioned above could achieve real-time video stitching, little consideration has been given on the quality of stitching, especially from the algorithm point of view. Therefore, intensity and structure misalignment cannot be effectively eliminated.

C. Structure Deformation

The idea of structure deformation is originated from medical image processing. Elastic registrations [44] align x-ray images of human brain through stretching and smoothing image content. Texture structure of images could be aligned based on matching relationship between object boundaries in two images [45]. Compared with the elastic registration, nonrigid viscous fluid registration [46] could handle input images with big content difference. A Bayesian-based mutual information technique combined with affine transformation was proposed for efficient medical image registration [47]. These methods align images using structure information of image content and deform images to correct structure misalignment. But it is difficult for these methods to handle natural images with detailed local textures.

Compared with the direct registration methods above, feature-based structure deformation could achieve better stitching results for natural images [8], [48], [49]. Because the matched feature point pairs reflect the corresponding relationships of image structures, the deformation could be done by stretching each pair of matched feature points together. Based on feature matching, Lin *et al.* [50] introduced a smoothly affine stitching field to handle parallax and provided a scene by avoiding abrupt protrusions. Zaragoza *et al*. [51] proposed a novel technique called moving direct linear transformation (moving DLT) that could significantly reduce the artifacts without compromising geometric realism of images. However, in most cases, accurate 2-D feature detection and matching is a time-consuming operation and mismatches may occur due to the complexity of image content. The method put forward in [15] enhanced the algorithm performance by detecting and matching feature points only in 1-D direction along the seam, whose stitching result is shown in Fig. 2(g). Jia and Tang [16] improved their approach by replacing the single optimal seam with double seams that across the same scene positions. The consistency of image content was better ensured, and the stitching result is shown in Fig. 2(h). When establishing the structure constraints, Jia and Tang [16] employed the triangulation of 2-D feature points, which made

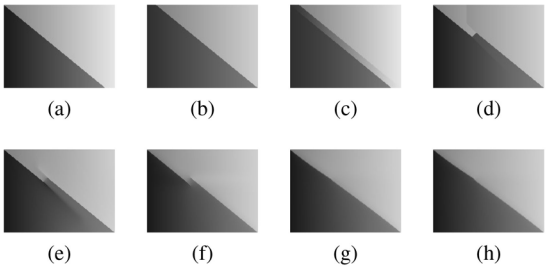


Fig. 2. Stitching results of different methods on toy examples with structure misalignment. (a) and (b) Overlapped region of two input images. (c) Result of alpha blending [26]. (d) Result of optimal seam [19]. (e) Result of pyramid blending [27] with optimal seam. (f) Result of gradient domain fusion [18] based on optimal seam. (g) Result of structure deformation with single seam [15]. (h) Result of structure deformation with double seams [16].

the structure constraints quite complex and difficult to optimize. Therefore, an iterative optimization technique has to be adopt while searching for double seams [16].

Overall, existing video stitching methods cannot ensure structure consistency in the overlapped region. Methods based on structure deformation can achieve better stitching quality but often with high computational cost, and therefore unsuitable for real-time stitching. To illustrate the concept of misalignment and the pitfalls of some popular methods for handling the misalignment, we designed a toy example with misalignments in Fig. 2. There is presence of both intensity and structure misalignment within them, which were shown as challenge for those popular methods. The results of different stitching methods are illustrated in Fig. 2(c)–(h).

III. Video Stitching Based on Fast Structure Deformation

In this section, we describe our methodology for video stitching based on fast structure deformation. In a video for a dynamic scene, the depth of pixels vary along time, the stitching parameters should also adjust accordingly. Therefore, for each group of synchronized video frames, our stitching method can be divided into four steps.

1) *Double-Seam Localization:* A model is developed to localize the double seams by considering both the spatial and temporal interframe constraints. Dynamic programming is employed to further select the two structurally corresponding seams.
2) *1-D Feature Detection and Matching:* The feature descriptors reflecting the abrupt changes across object boundaries are extracted and considered as the key feature points along the seams. The feature points from different sources are then matched using both their intensities and geometric locations.
3) *Deformation Quantization and Linear Propagation:* Each pair of matched feature points are moved to the expected coincident position. The offset of a feature point is defined as its deformation vector. In order to make the warped image perceivably smooth, the deformation vectors of the remaining image parts are then calculated through linear propagation.

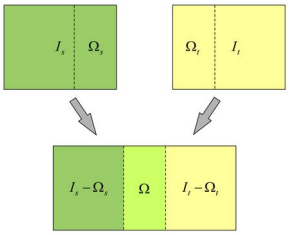


Fig. 3. Sketch of video stitching. I_s and I_t are two synchronized video frames. Ω_s and Ω_t are image parts of I_s and I_t in the overlapped region respectively. Ω is the overlapped region in the stitched frame. $I_s - \Omega_s$ and $I_t - \Omega_t$ denote other parts of I_s and I_t, respectively.

4) *Rapid Gradient Domain Fusion:* After obtaining the warped gradient images of the overlapped region, the final image is obtained by efficient SORI.

Implementation details of the four steps above will be discussed in the following parts of this section.

A. Efficient Double-Seam Selection

To illustrate the motivation of using double-seam, Fig. 3 shows an example where two video frames I_s and I_t are to be stitched. Let Ω to be the overlapped region in the stitched frame. Denote the image part of I_s in the overlapped region as Ω_s, and the image part of I_t in the overlapped region as Ω_t. Most existing seam selection schemes search for a single optimal seam in Ω_s, same as Ω_t. The single optimal seam usually performs well for input images with slight parallax. But for the situations where there are evident parallax between the overlapped image parts shown by Fig. 4(a) and (b), the result of single-seam selection could not preserve consistency of image structures along the seam, as shown in Fig. 4(c) and (d), which may cause visible artifacts in the stitched images. An effective solution for handling the difficulty above is to apply the double-seam selection, which means to select two distinct seams across the same scene location in Ω_s and Ω_t, respectively, as shown in Fig. 4(e) and (f). A seamless image Ω could be obtained by smoothly stretching the double seams together.

We designed a two-step double-seam selection scheme. The first step is to compute the single optimal seam as the approximate position of the double seams. The second step considers both the spatial and temporal constraints (interframe consistency), and searches for the double seams respectively, in Ω_s and Ω_t based on the location of single optimal seam. The scheme is described as follows.

The smoothness of the overlapped region could be measured by the combined intensity of its gradient images from the two source images

$$G_S = \|\nabla_x \Omega_s\| + \|\nabla_y \Omega_s\| + \|\nabla_x \Omega_t\| + \|\nabla_y \Omega_t\|. \quad (1)$$

The texture difference of the two images (Ω_s and Ω_t) could be measured by the norm of the difference in their corresponding gradient domain

$$G_D = \|\nabla_x \Omega_s - \nabla_x \Omega_t\| + \|\nabla_y \Omega_s - \nabla_y \Omega_t\| \quad (2)$$

where ∇_x and ∇_y are gradient operators in the image horizontal and vertical direction, respectively. $\|\cdot\|$ denotes the norm

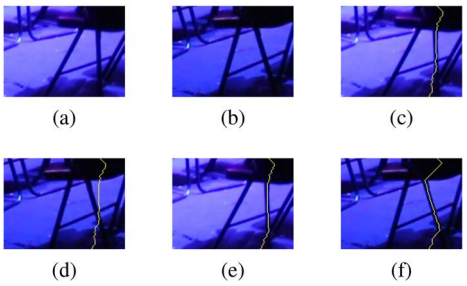


Fig. 4. Comparison between single seam and double seams. (a) and (b) Two overlapped image parts. (c) and (d) Single optimal seam locations in the two input images. (e) and (f) Two distinct but structural matched seams in the two input images, respectively (double seams).

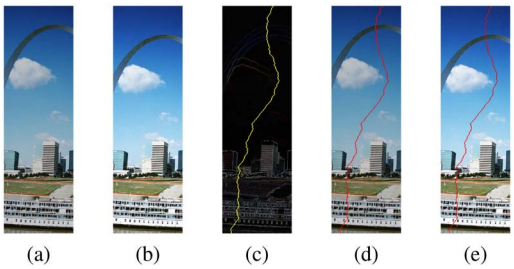


Fig. 5. Double-seam selection. (a) and (b) Overlapped image parts in left and right view. (c) Location of optimal seam S' on G_U when $\alpha = 0.5$. (d) and (e) Locations of double seams computed on the basis of S' and the gradient information.

function applied for each pixel. Here, we apply the standard National Television Standards Committee intensity conversion function $\|(r, g, b)\| = 0.2989\,|r| + 0.5780\,|g| + 0.1140\,|b|$ to merge the three channels of color images [52].

The smoothness and difference of the images could be combined in the following formula, based on which the optimal seam could be computed [16]:

$$G_U = \alpha G_S + (1 - \alpha) G_D. \tag{3}$$

The weight coefficient α is used to balance the influence of the smoothness and difference factors. It is set to 0.5 in our experiments which means equivalent attention is paid to them.

The optimal seam could be selected by optimizing a cost function built on (3). Different optimization methods have been developed and among them, the dynamic programming was employed in this paper, as it has relatively low computational cost and satisfactory result. Furthermore, its Markov property makes it easy to be implemented in hardware. Therefore, dynamic programming is performed on G_U to obtain the single optimal seam S', so that the horizontal coordinate of the seam at height y can be computed as $x = S'(y)$. For the overlapped image parts shown in Fig. 5(a) and (b), the obtained optimal S' is shown in Fig. 5(c).

Due to the registration error and the multiview parallax, the two seams in different views which should correspond to the same scene position usually do not perfectly coincide in practice. It is difficult to establish accurate structure matching relationships for natural images. The scheme proposed in [16] which is based on triangulation of feature points could handle images with sufficient features. But its complexity makes it unsuitable for video processing. It is worth to note that when the input images are registered, the location of double seams in different views should be similar. Based on this observation, we propose a double seam selection method by taking into consideration both the spatial and temporal constraints.

After defining the single optimal seam to be the reference location, the locations of the double seams S_s and S_t should not be far away from S'. The spatial constraint could be formulated into a penalty function R

$$R(x, y) = \left(x - S'(y)\right)^2 \tag{4}$$

where (x, y) denotes the image coordinate. Combined with the penalty function (4), we create the following cost function to compute the double seams from Ω_s and Ω_t, respectively:

$$\begin{cases} D_s = \|\nabla_x \Omega_s\| + \left\|\nabla_y \Omega_s\right\| + \gamma R \\ D_t = \|\nabla_x \Omega_t\| + \left\|\nabla_y \Omega_t\right\| + \gamma R \end{cases} \tag{5}$$

where parameter γ is used to control the strength of spatial constraints. It is set to 0.15 in our experiments and the value could be adjusted according to the seriousness of the parallax. For input images with evident parallax, larger space shift between seams should be allowed, which leads to a smaller γ to be employed and vice versa. Dynamic programming is running on D_s and D_t, respectively to get the location of S_s and S_t, as shown in Fig. 5(d) and (e).

The seam selection method presented above is for a single frame only without considering the interframe consistency. In a video for a scene, successive frames are temporally related, which implies that the locations of corresponding seams of two consecutive frames should not shift too much. Otherwise, due to the fact that regions the seams swept over are taken from different videos with perspective difference, the stitched video might not be smooth and jitter abnormally even if every single frame is visually seamless. To take this point into consideration, we formulate such a constraint into a penalty function to balance and associate the consecutive frames based on (3). Let S'_{p} be the optimal seam of previous frame. By adding a penalty function R_{p} based on S'_{p} to (3), we obtain the following cost function that is used to compute the optimal seam on the current frame:

$$G_{U,\mathrm{cur}} = G_U + \beta R_{\mathrm{p}} \tag{6}$$

where β is the control parameter of the temporal interframe constraint which is set to $f_p/300$ in our experiments. f_p denotes the frame rate of video. A high frame rate leads to small interframe difference (i.e., stronger interframe consistency), and therefore the corresponding temporal constraint should be stronger, which means a larger β should be used. Similar to the spatial constraint, we formulate the temporal constraint into a quadratic function

$$R_{\mathrm{p}}(x, y) = \left(x - S'_{\mathrm{p}}(y)\right)^2. \tag{7}$$

The formulation above ensures that the location of the double seam could shift smoothly across frames. Though the form of (4) and (7) are quite similar, (4) operates with S' and penalizes within current frame, while (4) is with S'_{p} and indicates a penalty for an interframe discrepancy.

During the computation of double seams, by setting the single optimal seam as reference location, the structure matching relationship of double seams and interframe consistency constraint are fully considered in this paper. The complexity of our approach for double seams searching is approximately three times as much as that of dynamic programming with relatively low computational cost, while Jia and Tang's approach [16] selected the double seams through a series of time-consuming operations, including graph cut, 2-D feature detection and matching, the triangulation of matched feature points, and finally, the iterative minimization of a complex cost function established upon prior procedures. Compared with Jia and Tang's [16] approach, our approach enables much faster computation of the double seams without sacrificing the stitching quality. The comparisons of the two approaches on both stitching quality and speed are shown in Section IV.

B. 1-D Feature Detection and Matching

Compared with gradual change of colors and intensities, human eyes are more sensitive to salient changes. When observing a natural image, people always pay attention on edges of objects and neglect the smooth color shading [53]. Therefore, it is natural to define the intersection of the seam and object boundaries as the feature point, which should be matched properly in the stitching process. By detecting and matching these 1-D feature points along the seams, the connecting relationship of object boundaries could be reflected in the matching results.

This definition of feature is equivalent to the local maximum in gradient domain [54]. Feature detection means searching for the points of the local maximum in gradient domain. Denote the pixel values of gradient images $|\nabla\Omega_s|$ and $|\nabla\Omega_t|$ at height y as $C_s(y)$ and $C_t(y)$, respectively. Because of the noise and detailed texture, the local peaks of C_s and C_t cannot indicate the location of object boundaries reliably, as shown by the black curves in Fig. 6(b) and (d). Therefore, it is necessary to smooth the curves before feature detection. The smoothing results C_s^{s} and C_t^{s} using 1-D Gaussian are shown as the blue curves in Fig. 6(b) and (d). The locations of feature points then could be robustly obtained by applying the nonmaximum suppression (NMS) [55]. We denoted $\{F_{s,1},\ldots,F_{s,m}\}$ as the set of m features on S_s, and $\{F_{t,1},\ldots,F_{t,n}\}$ as the set of n features on S_t, respectively.

Firstly, because of the exposure difference, even the intensity values of the matched feature points may differ largely, whilst they should be similar in the gradient domain as they reflect the same local texture structure. Secondly, relying on the fact that all images are registered, the geometric positions of a pair of matched feature points should be close to each other. By taking these two factors into consideration, we define the feature dissimilarity d_F that indicates the matching degree as follows:

$$d_F = \lambda d_G + (1-\lambda)\, d_E \tag{8}$$

where d_G is the gray-scale difference of two feature points in gradient domain and d_E is their geometric distance in the overlapped region. The coefficient λ is used to adjust the affection of the intensity and geometrical factors during feature matching. It is set to 0.8 in our experiments.

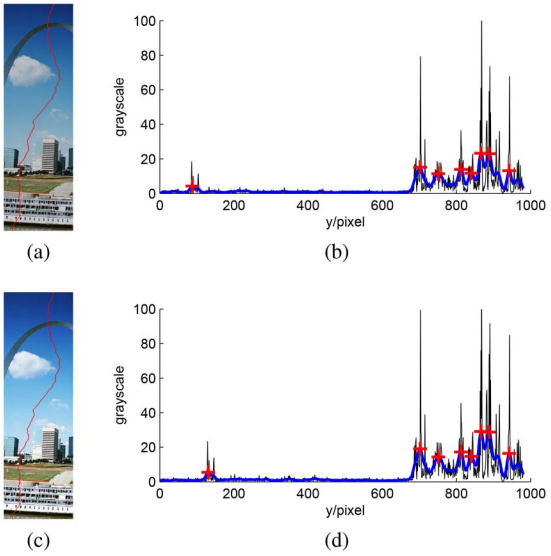


Fig. 6. 1-D feature detection. (a) Overlapped image part of left view with seam. (b) Feature detection of left view, where the black curve describes the variation trend of C_s, the blue curve describes the 1-D Gaussian smoothing result C_s^{s}, and the red marks indicate the feature locations. (c) Overlapped image part of right view with seam. (d) Feature detection of right view.

In order to reduce the effects of noise and detailed texture, the computation of d_G uses the Gaussian smoothed gray values. For a pair of feature points $F_{s,i}\left(x_{s,i}, y_{s,i}\right)$, $1 \le i \le m$ and $F_{t,j}\left(x_{t,j}, y_{t,j}\right)$, $1 \le j \le n$, d_G, and d_E are defined as follows:

$$d_G\left(F_{s,i}, F_{t,j}\right) = \frac{\left|C_s^{\mathrm{s}}\left(y_{s,i}\right) - C_t^{\mathrm{s}}\left(y_{t,j}\right)\right|}{\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\left|C_s^{\mathrm{s}}\left(y_{s,i}\right) - C_t^{\mathrm{s}}\left(y_{t,j}\right)\right|} \tag{9}$$

$$d_E\left(F_{s,i}, F_{t,j}\right) = \frac{\sqrt{\left(x_{s,i}-x_{t,j}\right)^2 + \left(y_{s,i}-y_{t,j}\right)^2}}{\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\sqrt{\left(x_{s,i}-x_{t,j}\right)^2 + \left(y_{s,i}-y_{t,j}\right)^2}}. \tag{10}$$

After computing the feature dissimilarity d_F of each feature pair from the sets $\{F_{s,1},\ldots,F_{s,m}\}$ and $\{F_{t,1},\ldots,F_{t,n}\}$, the feature matching could be achieved through greedy search or dynamic programming strategy.

The computational complexity of the feature detection algorithm is $O(H \times \ln W_{\mathrm{NMS}})$, where H is the number of pixels that the seam passed through. W_{NMS} is the moving window size in NMS [55]. The computational complexity of the feature matching algorithm is $O(m \times n)$, where m and n are the number of key feature points of the two seams, respectively.

C. Deformation Quantization and Linear Propagation

The double seams S_s and S_t are projections of the same objective curve in different views. They should be ideally coincident in the stitched image. As one option, the target seam location could be simply set to S_s or S_t, as done in [15] and [16]. A better option is to choose the single optimal seam S' as the target seam location because both S_s

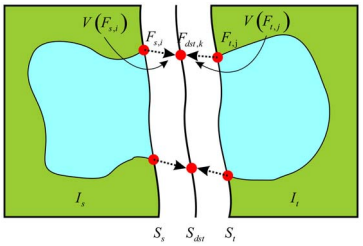


Fig. 7. Deformation quantization. S_s and S_t are seams of the two adjacent views, S_{dst} is the target seam, $F_{s,i}$ and $F_{t,j}$ are a pair of matched feature on double seams, $F_{\text{dst},k}$ is the coincident position of $F_{s,i}$ and $F_{t,j}$ after deformation, $V\left(F_{s,i}\right)$ and $V\left(F_{t,j}\right)$ deformation vectors which record the offset of $F_{s,i}$ and $F_{t,j}$, respectively.

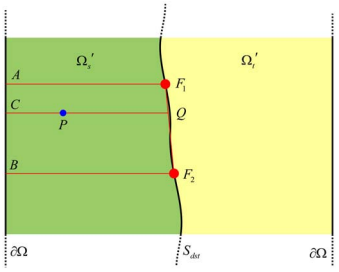


Fig. 8. Deformation propagation. $\partial\Omega$ is the boundary of overlapped region, S_{dst} is the coincident location of double seams after deformation, P is a point in Ω, F_1, and F_2 are two neighboring feature points in deformed image. P locates in trapezoidal area AF_1F_2B, and CQ is the horizontal line across P which meets AB at point Q. Deformation vector $V(Q)$ could be obtained by the linear interpolation of $V(A)$, $V(B)$, $V(F_1)$, and $V(F_2)$.

and S_t are derived from it. However, an ideal target seam location should also minimize the maximum deformation of input images, which means the best choice is $S_{\text{dst}} = (S_s + S_t/2)$. Therefore, in this paper, we move S_s and S_t to their medium location S_{dst} while stitching.

Notice that the location of the matched feature pairs reflects the connecting relationship of object boundaries across the seam. In order to ensure the continuity of these boundaries, each pair of matched feature points should be moved to one coincident location on S_{dst}, as shown in Fig. 7. Consider a pair of matched feature points $F_{s,i}(x_{s,i}, y_{s,i})$, $1 \le i \le m$ and $F_{t,j}(x_{t,j}, y_{t,j})$, $1 \le j \le n$, and assume that they meet at $F_{\text{dst},k}(x_{\text{dst},k}, y_{\text{dst},k})$, $1 \le k \le K$, where K is the number of matched feature pairs. Then, the image coordinate of $F_{\text{dst},k}$ is

$$\left(x_{\text{dst},k}, y_{\text{dst},k}\right) = \left(S_{\text{dst}}\left(\frac{y_{s,i}+y_{t,j}}{2}\right), \frac{y_{s,i}+y_{t,j}}{2}\right). \quad (11)$$

Define the deformation vector $V(F)$ to represent the geometric offset of a feature point $F \in \{F_{s,1}, F_{s,2}, \ldots, F_{s,m}\} \cup \{F_{t,1}, F_{t,2}, \ldots, F_{t,n}\}$

$$V(F) = \left(V_x(F), V_y(F)\right) \quad (12)$$

where $V_x(\cdot)$ and $V_y(\cdot)$ are horizontal and vertical component of $V(\cdot)$.

The definition of deformation vector is then extended to all pixels in the image. Deformation propagation is the process of propagating the deformation vector $V(\cdot)$ from the feature points to all other pixels smoothly. Denote the image range for propagation as Ω_P. Theoretically, Ω_P can be any arbitrary area that covers the seams. For computational convenience, we set $\Omega_P = \Omega$. Therefore, the other two image parts $I_s - \Omega_s$ and $I_t - \Omega_t$ would not be influenced by the deformation, which means

$$V(P) = 0 \ \forall P \in \partial\Omega \cup (I_s - \Omega_s) \cup (I_t - \Omega_t) \quad (13)$$

where $\partial\Omega$ is the boundary of the overlapped region. For a point $P_0(x_0, y_0)$ in the origin image Ω_s or Ω_t, a straight forward way to computing its corresponding point $P'(x', y')$ in the deformed image Ω, is defined as follows:

$$\begin{cases} x' = x_0 + V_x(P_0) \\ y' = y_0 + V_y(P_0). \end{cases} \quad (14)$$

Obviously this is a forward transformation. In most cases, the values of x', y' are uncontrolled super-pixel coordinates and not the desired location. It is not a trivial task to reconstruct a regular image from the output of (14). Instead, we consider the inverse mapping of $V(\cdot)$, denoted as $V^{-1}(\cdot)$. For a desired point $P(x, y)$ in the deformed image Ω, its corresponding location $P_0'\left(x_0', y_0'\right)$ in the origin image plane could be inferred from the following inverse mapping:

$$\begin{cases} x_0' = x - V_x\left(P_0'\right) = x + V_x^{-1}(P) \\ y_0' = y - V_y\left(P_0'\right) = y + V_y^{-1}(P). \end{cases} \quad (15)$$

The pixel value of $P_0'(x_0', y_0')$ could be easily obtained through bilinear interpolation in original image plane. Therefore, we use $V^{-1}(\cdot)$ instead of $V(\cdot)$ to quantify deformation. Computing $V^{-1}(\cdot)$ is a nontrivial task as it needs to optimize an energy function [16]. However, in the following, we theoretically show that the calculation of $V^{-1}(\cdot)$ could be formulated into a linear interpolation problem as shown in Fig. 8.

Consider a point $P(x, y)$ locating at the left side of S_{dst} and suppose $F_1(x_1, y_1)$ and $F_2(x_2, y_2)$ are two neighboring feature points, where $y_1 \le y \le y_2$. AB is a part of boundary $\partial\Omega$ and $V^{-1}(A) = V^{-1}(B) = V^{-1}(C) = 0$. $V^{-1}(F_1)$ and $V^{-1}(F_2)$ are already known. The deformation vector of $P(x, y)$ could then be obtained by linear interpolation as follows:

$$\begin{cases} V^{-1}(P) &= \dfrac{|PQ|}{|CQ|}V^{-1}(C) + \dfrac{|CP|}{|CQ|}V^{-1}(Q) \\ &= \dfrac{|CP|}{|CQ|}V^{-1}(Q) \\ |CQ| &= \mu\,|AF_1| + (1-\mu)\,|BF_2| \\ V^{-1}(Q) &= \mu V^{-1}(F_1) + (1-\mu)\,V^{-1}(F_2) \end{cases} \quad (16)$$

where in the overlapped region Ω, $\mu = (y_2 - y/y_2 - y_1)$, $|CP| = x - 1$, $|AF_1| = x_1 - 1$, $|AF_2| = x_2 - 1$. Similarly, we can compute the inverse deformation vector for any point P in the right side of S_{dst}. Then, $V^{-1}(\cdot)$ of all pixels in Ω could be obtained using (16). The pixel values of deformed image are then obtained using (15) with bilinear interpolation.

The definition of deformation vector in [15] and [16] also took the gradient image pixel value variation $V_{\|\nabla\|}(\cdot)$ into account. Then, the value of $V^{-1}(\cdot)$ of every pixel in the overlapped region could be obtained by solving the a minimization problem whose objective function is $\|\nabla V\|^2$. But the iterative optimization is computational expensive, which

(a) (b) (c) (d) (e) (f) (g)

Fig. 9. Comparison of iteration times. Using 0 as the initial value. (a) Result when iteration time $T_i = 10$. (b) Result when $T_i = 100$. (c) Result when $T_i = 1000$. (d) Result when $T_i = 10\,000$. (e) Result when $T_i = 20\,036$. Using approximate solution under the constraints in single direction. (f) Result when iteration time $T_i = 10$. (g) Whole stitched image combining all image parts.

is not applicable for video stitching. Our approach could be regarded as a linear approximation of the original minimization problem, with unnoticable difference of stitching quality with regard to the deformation smoothness when compared to the state-of-the-arts (as shown in Fig. 12 in the experimental section). We experimentally demonstrate that the proposed approach for the deformation propagation could achieve satisfactory stitching quality (see Section IV for details).

D. Efficient Gradient Domain Fusion

Image reconstruction from the gradient domain has been widely studied in different areas including image compression [56], image inpainting [57], image fusion [18], [30], [31], and etc. The advantages of operating in the gradient domain could achieve smooth color transition while keeping image structure information. Therefore, we apply gradient domain fusion to reconstruct the image in the overlapped region after structure deformation.

After the deformation propagation, we now have the gradient image (G_x, G_y) in place, where G_x and G_y represent the horizontal and vertical component, respectively. The goal of gradient domain fusion is to reconstruct the values of all pixels in Ω on the basis of (G_x, G_y) and the overlapped region boundary $\partial\Omega$. It could be achieved traditionally by solving Poisson eqnarrays [18], whose discrete version is equivalent to the following minimization problem:

$$\min_f \sum_{p\in\Omega-\partial\Omega, q\in N_p} \left(f(p) - f(q) - v_{pq}\right)^2 \tag{17}$$

where f is a 2-D function representing the image in the overlapped region to be optimized. p and q are two points in Ω, and N_p is the set of four-connected neighbors of p. v_{pq} is associated with the gradients of p and q, and its value could be computed as follows depending on their relative positions:

$$v_{pq} = \begin{cases} -\dfrac{G_x(p) + G_x(q)}{2} & x_p + 1 = x_q, y_p = y_q \\ \dfrac{G_x(p) + G_x(q)}{2} & x_p = x_q + 1, y_p = y_q \\ -\dfrac{G_y(p) + G_y(q)}{2} & x_p = x_q, y_p + 1 = y_q \\ \dfrac{G_y(p) + G_y(q)}{2} & x_p = x_q, y_p = y_q + 1. \end{cases} \tag{18}$$

The minimization problem described in (17) could be transformed into a system of linear equations, which is for all $p \in \Omega - \partial\Omega$

$$|N_p| f(p) - \sum_{q\in N_p\cap(\Omega-\partial\Omega)} f(q) = \sum_{q\in N_p\cap\partial\Omega} f(q) + \sum_{q\in N_p} v_{pq} \tag{19}$$

where $|N_p|$ is the number of neighbors of p. Here, $|N_p| = 4$. The coefficient matrix of (19) is a large-scale, sparse, symmetric, and positive definite matrix which is very difficult to be solved by conventional approaches such as Gaussian elimination. Normally, an iterative optimization technique has to be approached to solve (19). In this paper, we deploy a sparse matrix solver originally developed in [58] called SORI. We found that its convergence speed is still very sensitive to its initial value settings. Fig. 9 shows an example of the results of reconstructed images under different number of iterations. We can see that, to achieve a satisfactory reconstruction result as shown in Fig. 9(e), it requires 20 k iterations for an image of size 300×900. This computational speed is too slow to achieve high computational efficiency.

To efficiently solve (19), we develop a divide and conquer strategy. By considering the neighborhood relationship embedded in (19), we propose a principled solution to the initialization of the SORI solver by breaking down the neighborhood constraint into horizontal and vertical component separately. The purpose of this step is to construct a set of small-scale linear systems that could be easily solved by Gaussian elimination. For an image of size $H \times W$, the size of the coefficient matrix is $(H \times W) \times (H \times W)$. By considering the neighborhood relationship along the horizontal direction only, (19) would be simplified as follows:

$$2f(p) = f(q_l) + f(q_r) + \frac{G_x(q_l) - G_x(q_r)}{2} \tag{20}$$

where $x_p = x_{q_l} + 1 = x_{q_r} - 1$ and $y_p = y_{q_l} = y_{q_r}$ and this corresponds to the horizontal constraints set in (18). In (20), every row in the image corresponds to a system of linear equations whose coefficient matrix is of size $W \times W$. These H small-scale linear problems could be solved using Gauss elimination. The result of (20) can be reasonably considered as an approximate solution of (19), denoted as f_{dx}.

Similarly, if we consider the neighborhood relationship along the vertical direction, (19) would be simplified to a set

of W linear problems, each of which with a coefficient matrix size of $H \times H$ is formulated as follows:

$$2f(p) = f(q_u) + f(q_d) + \frac{G_y(q_u) - G_y(q_d)}{2} \tag{21}$$

where $x_p = x_{q_u} = x_{q_d}$ and $y_p = y_{q_u} + 1 = y_{q_d} - 1$ and this corresponds to the vertical constraints set in (18). By solving (21), we could get another approximate solution of (19), denoted as f_{dy}. We then define an image f_{init} as a weighted average of f_{dx} and f_{dy}

$$f_{\text{init}} = \eta f_{dx} + (1 - \eta) f_{dy}. \tag{22}$$

The image f_{init} contains structure information both in horizontal and vertical direction, and would be a reasonable initialization for (19). Using f_{init} as the initial value, the satisfactory reconstruction result could be reached when the number of iterations is 10 ($T_i = 10$), as shown in Fig. 9(f). Theoretically, the convergent solution of Poison equation is unrelated with the initial value. Therefore, the accuracy of the results would not be influenced by usage of f_{init}. Combined with other parts of input images, the whole stitched image is shown in Fig. 9(g). Obviously, the introduction of initial value f_{init} drastically improved the convergence speed, and the number of iterations required is reduced from 20 k to 10 only to reach the same reconstruction quality. It is worth noting that our scheme is very efficient for the rectangular overlapped regions, which is the case in the video-stitching application where the cameras can be easily aligned within the same plane. For the regions of arbitrary shapes, it is not directly applicable. However, this could be rectified by geometrical transformations such as rotating or rectangling after image registration.

Complexity and Efficiency Analysis: To summarize our methodology, in double-seam selection of Section III-A, structure information and difference of input images are considered to search two seams across the same scene position in the two adjacent views, which lays down the foundation for deformation and fusion. The selection strategy is based on the optimal single seam and takes both the spatial and temporal constraints into consideration. The computational complexity of search algorithm based on dynamic programming is $O(H \times W)$ for an overlapped region of size $H \times W$. In Section III-B, the connecting relationship of object boundaries is described by 1-D feature detection and matching along the seams. Its computational complexity is lower than $O(H \times W)$. The elimination of structure misalignment is achieved through deformation propagation in Section III-C. An efficient deformation propagation based linear interpolation is introduced. Its computational complexity is $O(H \times W)$. In Section III-D, the intensity misalignment between images is smoothed over the whole overlapped region using gradient domain fusion. To efficiently solve the corresponding Poisson equations, an approximate solution of two sets of linear equations for each single directions is obtained to construct the initial value. Compared with an initial value of zero, the required number of iterations is reduced from 20 k to 10. Therefore, the computational complexity of gradient domain fusion is also $O(H \times W)$. In a word, the video stitching method based on fast structure deformation proposed in this paper could eliminate intensity and structure misalignment between input frames effectively, and the total computational complexity of our method is $O(H \times W)$. Experimental results in Section IV confirm that our method is much faster than other structure deformation based stitching methods.

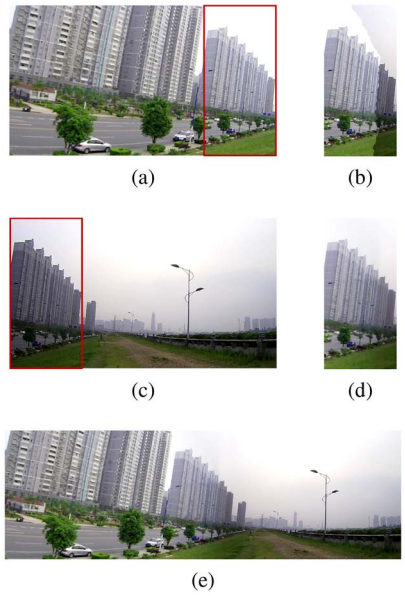


Fig. 10. Processing of intensity misalignment. (a) and (c) Two input images with distinct intensity misalignment. (b) Result of optimal seam algorithm, where there are salient intensity changes beside the seam. (d) Result of our method. The intensity misalignment is eliminated. (e) Entire stitched image.

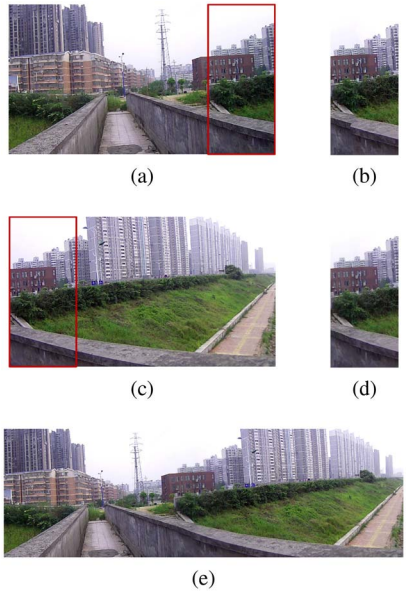


Fig. 11. Processing of structure misalignment. (a) and (c) Two input images with large depth variation. (b) Result of optimal seam algorithm. There is distinct structure misalignment in overlapped region. (d) Result of our method. The structure misalignment is eliminated. (e) Entire stitched image.

IV. Results

In this section, we carried out a group of experiments to evaluate the performance of our approach including: 1) testing

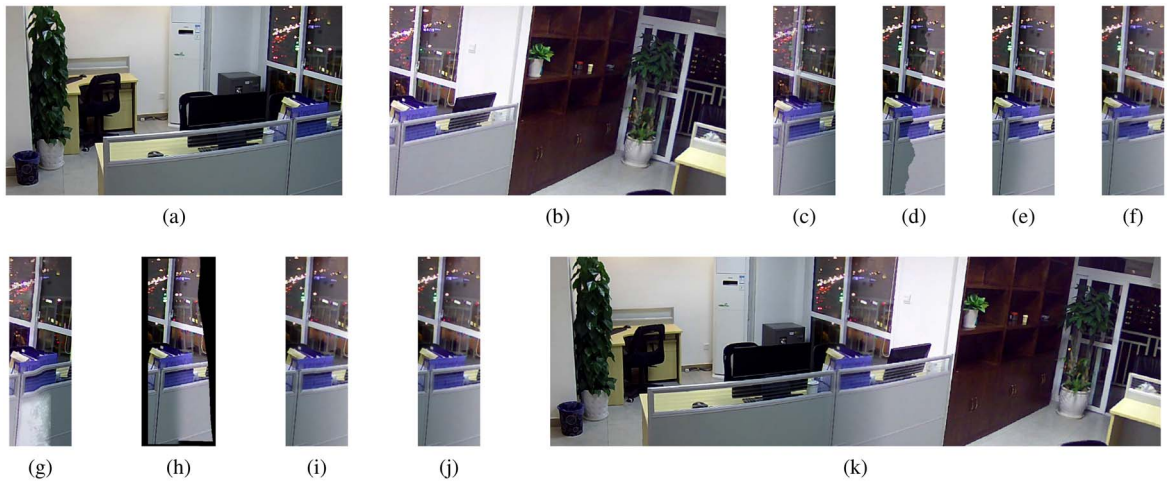


Fig. 12. Comparison with other methods. (a) and (b) Two input images with both intensity and structure misalignment. (c) Result of alpha blending [26]. (d) Result of optimal seam [19]. (e) Result of pyramid blending [27] with optimal seam. (f) Result of gradient domain fusion [18] based on optimal seam. (g) Result of image melding [33]. (h) Result of moving DLT [51]. (i) Result of Jia and Tang's method [16]. (j) Result of our method. (k) Entire stitched image.

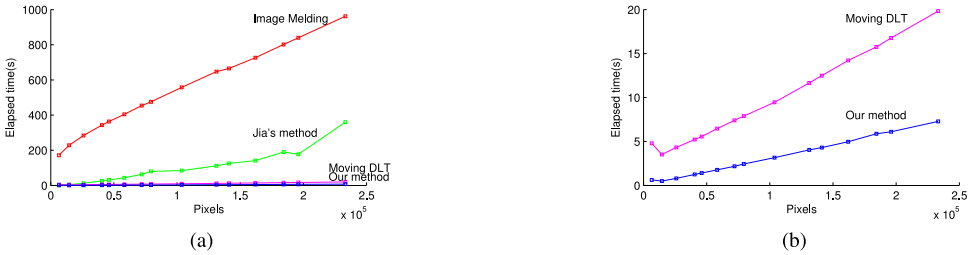


Fig. 13. Computational efficiency comparison (elapsed time). The horizontal axis indicates the number of pixels and the vertical axis indicates the elapsed time of the algorithms. (a) Comparison of computational cost of four advanced methods including ours. (b) Zoomed version obtained by magnifying (a) vertically. It shows that our method is faster than the other three methods, and its elapsed time is approximately linear to the number of pixels.

its capability for handling intensity and structure misalignment, respectively; 2) comparison with other seven methods in terms of stitching quality; 3) comparison of the computational efficiency with the state-of-the-art approaches including Jia and Tang's method [16], image melding [33], moving DLT [51]; and 4) video stitching results for indoor and outdoor dynamic scenes.

A. Intensity Misalignment

Because of the difference of exposure parameters in multiple views, intensity misalignment may exist between input images. Fig. 10(a) and (c) are two registered images with distinct illumination difference. Fig. 10(b) is the result of optimal seam algorithm [19]. There are salient intensity changes beside the seam. The result of our method is shown in Fig. 10(d). The intensity misalignment is well eliminated. Different from traditional weighted average schemes, in this paper, the smooth transition of colors is accomplished by gradient domain fusion which does not generate ghost images. Fig. 10(e) shows the whole stitched seamless image after combining the nonoverlapped regions.

B. Structural Misalignment

When the depth of field changes largely in the overlapped region, the parallax may cause structure misalignment of

Fig. 14. Multivideo capture devices. The relative locations of the three cameras are fixed. The synchronization of videos is achieved by the synchronized control of shutters.

images. Fig. 11(a) and (c) are two input images with strong structure elements. Fig. 11(b) shows the results using the optimal seam algorithm. It could be observed that, at the top of image where the objects are far away, the structure matched well, but at the bottom where the objects are nearer, there is distinct structure misalignment. The result of our method is shown in Fig. 11(d). The structure misalignment is eliminated. The object boundaries are well seamed by 1-D feature detection and matching along the double seams. The structure misalignment is removed through deformation propagation. The whole stitched seamless image is shown in Fig. 11(e).

C. Comparison With Others

The comparison of our method with other methods are shown in Fig. 12. Fig. 12(a) and (b) are two input images

TABLE I
STITCHING QUALITY COMPARISON OF DIFFERENT METHODS

Method	Blurring	Ghosting	Intensity misalignment	Structural misalignment	Regional tweak
Alpha Blending [26]	No	No	Yes	No	Yes
Optimal Seam [19]	Yes	Yes	No	No	Yes
Pyramid Blending [27]	Yes	No	Yes	No	Yes
Gradient Fusion [18]	Yes	No	Yes	No	Yes
Image Melding [33]	No	Yes	No	Yes	Yes
Moving DLT [51]	Yes	Yes	Yes	Yes	No
Jia's Method [16]	Yes	Yes	Yes	Yes	Yes
Our Method	Yes	Yes	Yes	Yes	Yes

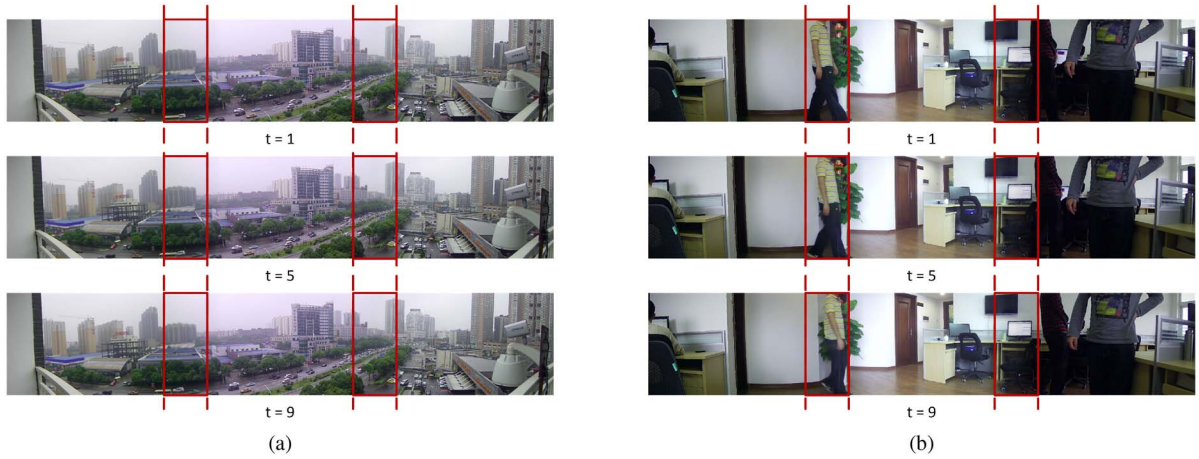


Fig. 15. Stitching results of successive video frames. (a) Outdoor dynamic scene. (b) Indoor dynamic scene. The sampling interval is four frames. The boxed areas indicate the overlapped regions.

with both intensity and structure misalignment. The result of alpha blending [26] is shown in Fig. 12(c). It could be seen that there are serious ghost images in overlapped region. The stitching result of optimal seam [19] is shown in Fig. 12(d), where the intensity and structure inconsistency are all distinct. The result by applying pyramid blending in [27] and optimal seam is shown in Fig. 12(e). The ghost image phenomenon is well avoided and the intensity difference is smoothed. But the structure misalignment still exists. Fig. 12(f) is the result of gradient domain fusion [18]. Similar to pyramid blending, this method could smooth the intensity difference of images but could not handle structure misalignment. The result by applying image melding [33] is shown in Fig. 12(g). There are unexpected artifacts in monochromatic areas where this patch-based method does not work well. Fig. 12(h) shows the result generated by moving DLT [51], in which the structure misalignment is eliminated. Meanwhile, the shape of overlapped region is also geometrically tweaked and the nonoverlapped regions have to be tweaked accordingly. This leads to additional computational cost. Result generated by Jia and Tang's [16] method is presented in Fig. 12(i). Similar stitching result could be achieved using our method, as shown in Fig. 12(j), where the structure misalignment of images is eliminated effectively (it is much faster than Jia and Tang's [16] method. see the efficiency comparison section for details). Apparently, the stitching quality is better than traditional methods and the shape of overlapped region remains unchanged. The whole stitched image produced using our method is shown in Fig. 12(k).

Beyond visual observations, five types of typical visual artifacts were selected as the criteria to compare the capability of above stitching methods on our dataset. Table I summarizes the qualitative comparisons of these methods against each criterion, i.e., whether each type of artifacts could be handled by each of the methods considered. From the table, we can see that both our method and Jia's method are capable of handling all of the five artifacts. We further compare their efficiency in the following section.

D. Efficiency Comparison

To verify the computational efficiency of our method, several groups of images with different size are chosen to be stitched. Our method is compared with the state-of-the-art video and image stitching approaches including image melding, Jia's method, moving DLT in same experimental settings with a 1.8 GHz Intel Core i5 CPU and a 4 GB memory. The computational costs in terms of processing speed of different methods are depicted by the curves in Fig. 13, where Fig. 13(b) is a zoomed version and obtained by magnifying Fig. 13(a) vertically to distinguish the two relatively fast methods. It is obvious that our method has significant advantages in computational speed compared with the existing methods. Moreover, as shown in Fig. 13(b), the computational cost of the proposed approach is linearly related with the number of

pixels in overlapped region. This confirms the complexity analysis in Section III that the computational complexity of our method is $O(H \times W)$.

E. Video Stitching for Dynamic Scenes

As shown in Fig. 14, we designed a multivideo capture device for the video stitching experiment with three-view synchronized videos. The relative locations of the three cameras are fixed so that their relative location and pose are stable. The frame rate is set as 30 frames/s, and the size of each frame is 1280×720 p. For every group of synchronized frames, the three source images are projected into the same observation plane and then stitched using our method. Fig. 15 shows the video stitching results of both outdoor and indoor dynamic scene. For the convenience of observing the dynamics of image contents, stitched frames are sampled using an interval of four frames. It can be seen that the local structure consistency of moving objects are well kept, and no abnormal flashes would occur while playing the stitched video.

V. Conclusion

In this paper, we propose an efficient video stitching method based on fast structure deformation. It is capable of eliminating the intensity and structure misalignment of input images, while achieving high computational efficiency. Specifically, we introduce a double-seam selection model by taking the spatial and temporal constraints into consideration. We theoretically show that the deformation model could be formulated into an inverse interpolation problem with low computational complexity. A principled solution is proposed to SORI for efficient gradient fusion. We performed an overall computational complexity analysis for our approach and showed that it is linear to the areas of the overlapped region. Our results of efficiency comparison verified this. We compared our approach with other seven approaches, and results demonstrate that our method could achieve better stitching quality than traditional methods (or comparable to the methods of Jia and Tang [16]) with lowest computational cost.

The application of our video stitching method requires that the distance of optical centers of cameras should be much less than the depth of field. Otherwise, there would be large parallax between input images which may lead to difficulties of accurate structure matching. If there are plenty of complex and similar structures in the scene, the mismatch of feature points may exist. Those challenges will be addressed in our future work.

References

[1] R. Szeliski, "Image alignment and stitching: A tutorial," *Found. Trends Comput. Graph. Vis.*, vol. 2, no. 1, pp. 1–104, 2006.

[2] C.-S. Chen, W.-T. Hsieh, and J.-H. Chen, "Panoramic appearance-based recognition of video contents using matching graphs," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 179–199, Feb. 2004.

[3] S. Chew *et al.*, "In the pursuit of effective affective computing: The relationship between features and registration," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 4, pp. 1006–1016, Aug. 2012.

[4] G. Rogez, J. Rihan, J. Guerrero, and C. Orrite, "Monocular 3-D gait tracking in surveillance scenes," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 894–909, Jun. 2014.

[5] N. Liu, H. Wu, and L. Lin, "Hierarchical ensemble of background models for PTZ-based video surveillance," *IEEE Trans. Cybern.*, vol. 45, no. 1, pp. 89–102, May 2014.

[6] X. Z. L. Zhang and L. Shao, "Learning object-to-class kernels for scene classification," *IEEE Trans. Image Process.*, vol. 23, no. 8, pp. 3241–3253, Aug. 2014.

[7] F. Zhu and L. Shao, "Weakly-supervised cross-domain dictionary learning for visual recognition," *Int. J. Comput. Vis.*, vol. 109, nos. 1–2, pp. 42–59, 2014.

[8] Q. Wu and Y. Yu, "Feature matching and deformation for texture synthesis," *ACM Trans. Graph. (TOG)*, vol. 23, no. 3, pp. 364–367, 2004.

[9] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *Int. J. Comput. Vis.*, vol. 74, no. 1, pp. 59–73, 2007.

[10] F. M. Candocia, "Jointly registering images in domain and range by piecewise linear comparametric analysis," *IEEE Trans. Image Process.*, vol. 12, no. 4, pp. 409–419, Apr. 2003.

[11] P. Tools. (2013). *Panorama Tools*. [Online]. Available: http://www.panotools.org/dersch/, accessed May 3, 2013.

[12] Hugin. (2013). *Hugin—Panorama Photo Stitcher*. [Online]. Available: http://hugin.sourceforge.net/, accessed May 5, 2013.

[13] ArcSoft. (2013). *Panorama Maker*. [Online]. Available: http://www.arcsoft.com/panorama-maker/, accessed May 12, 2013.

[14] Q. Zhi and J. R. Cooperstock, "Toward dynamic image mosaic generation with robustness to parallax," *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 366–378, Jan. 2012.

[15] J. Jia and C.-K. Tang, "Eliminating structure and intensity misalignment in image stitching," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2. Beijing, China, 2005, pp. 1651–1658.

[16] J. Jia and C.-K. Tang, "Image stitching using structure deformation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 4, pp. 617–631, Apr. 2008.

[17] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.

[18] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graph. (TOG)*, vol. 22, no. 3, pp. 313–318, 2003.

[19] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proc. 28th Annu. Conf. Comput. Graph. Interact. Tech.*, Los Angeles, CA, USA, 2001, pp. 341–346.

[20] B. Appleton, A. P. Bradley, and M. Wildermoth, "Towards optimal image stitching for virtual microscopy," in *Proc. IEEE Digit. Image Comput. Tech. Appl. (DICTA)*, 2005, p. 44.

[21] S. J. Ha, H. Koo, S. H. Lee, N. I. Cho, and S. K. Kim, "Panorama mosaic optimization for mobile camera systems," *IEEE Trans. Consum. Electron.*, vol. 53, no. 4, pp. 1217–1225, Nov. 2007.

[22] H. Gu, Y. Yu, and W. Sun, "A new optimal seam selection method for airborne image stitching," in *Proc. IEEE Int. Workshop Imag. Syst. Tech.*, Shenzhen, China, 2009, pp. 159–163.

[23] V. Kwatra, A. Schödl, I. Essa, G. Turk, and A. Bobick, "Graphcut textures: Image and video synthesis using graph cuts," *ACM Trans. Graph. (TOG)*, vol. 22, no. 3, pp. 277–286, 2003.

[24] A. Mills and G. Dudek, "Image stitching with dynamic elements," *Image Vis. Comput.*, vol. 27, no. 10, pp. 1593–1602, 2009.

[25] B. Summa, J. Tierny, and V. Pascucci, "Panorama weaving: Fast and flexible seam processing," *ACM Trans. Graph. (TOG)*, vol. 31, no. 4, p. 83, 2012.

[26] S. Peleg, "Elimination of seams from photomosaics," *Comput. Graph. Image Process.*, vol. 16, no. 1, pp. 90–94, 1981.

[27] P. J. Burt and E. H. Adelson, "A multiresolution spline with application to image mosaics," *ACM Trans. Graph. (TOG)*, vol. 2, no. 4, pp. 217–236, 1983.

[28] E. H. Adelson, C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden, "Pyramid methods in image processing," *RCA Eng.*, vol. 29, no. 6, pp. 33–41, 1984.

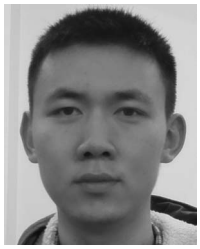
[29] A. Agarwala *et al.*, "Interactive digital photomontage," *ACM Trans. Graph. (TOG)*, vol. 23, no. 3, pp. 294–302, 2004.

[30] A. Levin, A. Zomet, S. Peleg, and Y. Weiss, "Seamless image stitching in the gradient domain," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Prague, Czech Republic, 2004, pp. 377–389.

[31] A. Zomet, A. Levin, S. Peleg, and Y. Weiss, "Seamless image stitching by minimizing false edges," *IEEE Trans. Image Process.*, vol. 15, no. 4, pp. 969–977, Apr. 2006.

[32] A. Agarwala, "Efficient gradient-domain compositing using quadtrees," *ACM Trans. Graph. (TOG)*, vol. 26, no. 3, p. 94, 2007.

[33] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, "Image melding: Combining inconsistent images using patch-based synthesis," *ACM Trans. Graph. (TOG)*, vol. 31, no. 4, p. 82, 2012.
[34] M. Uyttendaele, A. Eden, and R. Skeliski, "Eliminating ghosting and exposure artifacts in image mosaics," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Kauai, HI, USA, 2001, pp. II-509–II-516.
[35] H. S. Sawhney, S. Hsu, and R. Kumar, "Robust video mosaicing through topology inference and local to global alignment," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Freiburg, Germany, 1998, pp. 103–119.
[36] C.-T. Hsu, T.-H. Cheng, R. A. Beuker, and J.-K. Horng, "Feature-based video mosaic," in *Proc. IEEE Int. Conf. Image Process.*, vol. 2. Vancouver, BC, Canada, 2000, pp. 887–890.
[37] A. Smolic and T. Wiegand, "High-resolution video mosaicing," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3. Thessaloniki, Greece, 2001, pp. 872–875.
[38] D. Steedly, C. Pal, and R. Szeliski, "Efficiently registering video into panoramic mosaics," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2. Beijing, China, 2005, pp. 1300–1307.
[39] M. Zheng, X. Chen, and L. Guo, "Stitching video from webcams," in *Advances in Visual Computing*. Berlin, Germany: Springer, 2008, pp. 420–429.
[40] B. He, G. Zhao, and Q. Liu, "Panoramic video stitching in multi-camera surveillance system," in *Proc. 25th Int. Conf. Image Vis. Comput. (IVCNZ)*, Queenstown, New Zealand, 2010, pp. 1–6.
[41] Kolor. (2014). *Kolor Autopano Video*. [Online]. Available: http://www.kolor.com/, accessed Mar. 10, 2014.
[42] M. El-Saban, M. Izz, and A. Kaheel, "Fast stitching of videos captured from freely moving devices by exploiting temporal redundancy," in *Proc. IEEE Int. Conf. Image Process.*, Hong Kong, 2010, pp. 1193–1196.
[43] M. El-Saban, M. Izz, A. Kaheel, and M. Refaat, "Improved optimal seam selection blending for fast video stitching of videos captured from freely moving devices," in *Proc. IEEE Int. Conf. Image Process.*, Brussels, Belgium, 2011, pp. 1481–1484.
[44] R. Bajcsy and S. Kovačič, "Multiresolution elastic matching," *Comput. Vis. Graph. Image Process.*, vol. 46, no. 1, pp. 1–21, 1989.
[45] C. Davatzikos, J. L. Prince, and R. N. Bryan, "Image registration based on boundary mapping," *IEEE Trans. Med. Imag.*, vol. 15, no. 1, pp. 112–115, Feb. 1996.
[46] M. Bro-Nielsen and C. Gramkow, "Fast fluid registration of medical images," in *Proc. Vis. Biomed. Comput.*, Hamburg, Germany, 1996, pp. 265–276.
[47] H. Zhou *et al.*, "Towards efficient registration of medical images," *Comput. Med. Imag. Graph.*, vol. 31, no. 6, pp. 374–382, 2007.
[48] H. Fang and J. C. Hart, "Textureshop: Texture synthesis as a photograph editing tool," *ACM Trans. Graph. (TOG)*, vol. 23, no. 3, pp. 354–359, 2004.
[49] W. Xu, W. Chen, J. Zhang, and M. Zhang, "Angle consistency for registration between catadioptric omni-images and orthorectified aerial images," *IET Image Process.*, vol. 7, no. 4, pp. 343–354, 2013.
[50] W.-Y. Lin, S. Liu, Y. Matsushita, T.-T. Ng, and L.-F. Cheong, "Smoothly varying affine stitching," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, 2011, pp. 345–352.
[51] J. Zaragoza, T.-J. Chin, M. S. Brown, and D. Suter, "As-projective-as-possible image stitching with moving DLT," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Portland, OR, USA, 2013, pp. 2339–2346.
[52] C. A. Poynton, *Digital Video and HDTV: Algorithms and Interfaces*. Boston, MA, USA: Morgan Kaufmann, 2003.
[53] D. Marr and A. Vision, *A Computational Investigation into the Human Representation and Processing of Visual Information*. San Francisco, CA, USA: W. H. Freeman, 1982.
[54] P. Perona and J. Malik, "Detecting and localizing edges composed of steps, peaks and roofs," in *Proc. IEEE Int. Conf. Comput. Vis.*, Osaka, Japan, 1990, pp. 52–57.
[55] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. IAPR Int. Conf. Pattern Recognit.*, vol. 3. Hong Kong, 2006, pp. 850–855.
[56] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," *ACM Trans. Graph. (TOG)*, vol. 21, no. 3, pp. 249–256, 2002.
[57] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels," *IEEE Trans. Image Process.*, vol. 10, no. 8, pp. 1200–1211, Aug. 2001.
[58] J. Bolz, I. Farmer, E. Grinspun, and P. Schröoder, "Sparse matrix solvers on the GPU: Conjugate gradients and multigrid," *ACM Trans. Graph. (TOG)*, vol. 22, no. 3, pp. 917–924, 2003.



Jing Li received the bachelor's degree from the National University of Defense Technology, Changsha, China, in 2011, where he is currently pursuing the Ph.D. degree from the College of Information System and Management.

His current research interests include computer vision, image processing, and image analysis.

Wei Xu received the B.S. and M.Sc. degrees in information engineering and the Ph.D. degree in system engineering, all from the National University of Defense Technology, Changsha, China, in 1996, 1999, and 2007, respectively.

He is currently an Associate Professor with the Department of System Engineering, National University of Defense Technology. His current research interests include multimedia technology, virtual reality, and image processing.

Jianguo Zhang received the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2002.

He is currently a Senior Lecturer (Associate Professor) with the School of Computing, University of Dundee, Dundee U.K. His current research interests include visual surveillance, object recognition, image processing, medical image analysis, and machine learning.

Maojun Zhang received the Ph.D. degree in systems engineering from the National University of Defense Technology, Changsha, China, in 1997.

He is currently a Professor with the Department of System Engineering, National University of Defense Technology, Changsha, China. His current research interests include computer vision, information system engineering, system simulation, and virtual reality technology.

Zhengming Wang received the Ph.D. degree from the National University of Defense Technology, Changsha, China, in 1998.

He is a Distinguished Professor with the College of Science, National University of Defense Technology, Changsha, China. He has authored three academic monographs and over 50 papers in the field of data mining, system analysis, and time series analysis.

Xuelong Li (M'02–SM'07–F'12) is a Full Professor with the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China.