SIViP (2012) 6:647–667
DOI 10.1007/s11760-010-0194-4

ORIGINAL PAPER

Image mosaic with relaxed motion

Xianyong Fang · Jiejie Zhu · Bin Luo

Received: 27 December 2009 / Revised: 17 October 2010 / Accepted: 31 October 2010 / Published online: 21 November 2010

Abstract We propose a novel method to stitch images with relatively large roll or pitch called relaxed motion, which defies most existing mosaic algorithms. Our approach adopts a multi-resolution strategy, which combines the merits of both feature-based and intensity-based methods. The main contribution is a robust motion estimation procedure which integrates an adaptive multi-scale block matching algorithm called TV-BMA, a low contrast filter and a RANSAC motion rectification to jointly refine motion and feature matches. Based on $TV - L^1$ model, the proposed TV-BMA works on the coarsest layer to find a robust initial displacement field as the initial motion for source images. This motion estimation method can generate robust correspondences for further processing. In the subsequent camera calibration step, we also present two stable methods to estimate the camera matrix. To estimate the focal length, we combine the golden section search and the simplex method based on the angle invariance of feature vectors; to estimate the rotation matrix, we introduce a subspace trust region method, which matches features based on the rotation invariance. Extensive experiments show that our approach leads to improved accuracy and robustness for stitching images with relaxed motion.

X. Fang (✉) · B. Luo
Key Lab. of Intelligent Computing and Signal Processing of MOE, School of Computer Science and Technology, Anhui University, Anhui, China
e-mail: fangxianyong@ahu.edu.cn

B. Luo
e-mail: luobinahu@yahoo.com.cn

J. Zhu
Computer Vision Lab, University of Central Florida, Orlando, USA
e-mail: jjzhu@cs.ucf.edu

Keywords Image mosaic · Relaxed motion · Block matching algorithm · Low contrast filter · $TV - L^1$ model · Angle invariance · Rotation invariance

1 Introduction

Image mosaic or stitching refers to the problem of merging multiple images with overlapped views into a single composition. The traditional methods can only deal with camera rotating around a nearly fixed optical center. Using the terms in flight dynamics (Fig. 1a), this rotation direction is yaw (rotating around the vertical axis) with a nearly fixed optical center. However, (Fig. 1b), during the actual photographing process, rolling and pitching are also difficult to avoid. In this figure, line d bisecting the current image I_1 denotes the ideally horizontal position of the camera as it yaws around the nearly fixed optical center. The next image, I_2 or I_3, is taken with slight camera motion; However, I_4 is then taken with relatively large camera motion. While existing studies [1] can effectively stitch slightly rolled or pitched images (I_2 or I_3 with I_1 in Fig. 1b), to our best knowledge, there is no study on how to align relatively large rolled or pitched images (I_4 with I_1 in Fig. 1b). Referring to this type of camera motion as *relaxed motion*, we will study this motion problem in this paper and present a solution to image alignment and stitching.

Relaxed motion may come under the following two circumstances: (1) During the tedious capturing process, the user may carelessly rotate the camera in larger roll or pitch and (2) sometimes the user may deliberately rotate the camera in large roll or pitch to include some objects. Figure 2a and b show two examples from real scenes with unknown large roll or pitch. Their pixel correspondences are difficult

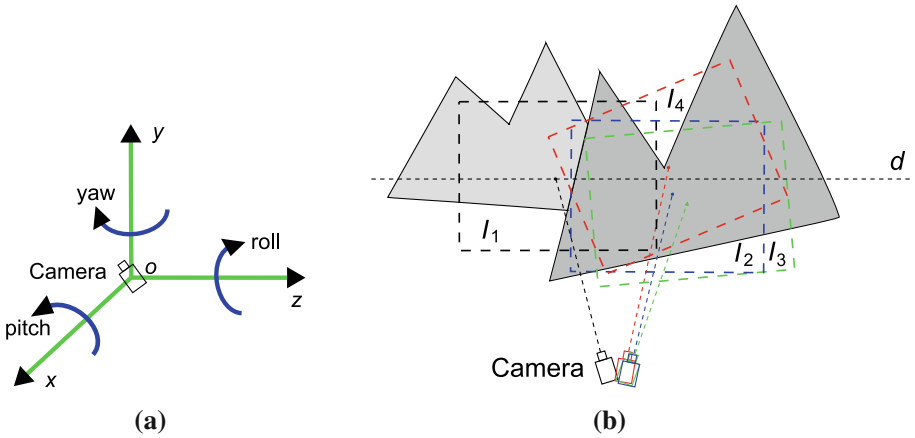

Fig. 1 Illustration of the relaxed motion. **a** Camera rotation directions in flight dynamics terms with o being the optical center. **b** The real camera motion during photographing. I_1 (with *black edges*) is the current image. I_2 (with *blue edges*), I_3 (with *green edges*) or I_4 (with *red edges*) is the subsequent image taken. Notice I_4 is obtained with larger roll or pitch than I_2 and I_3

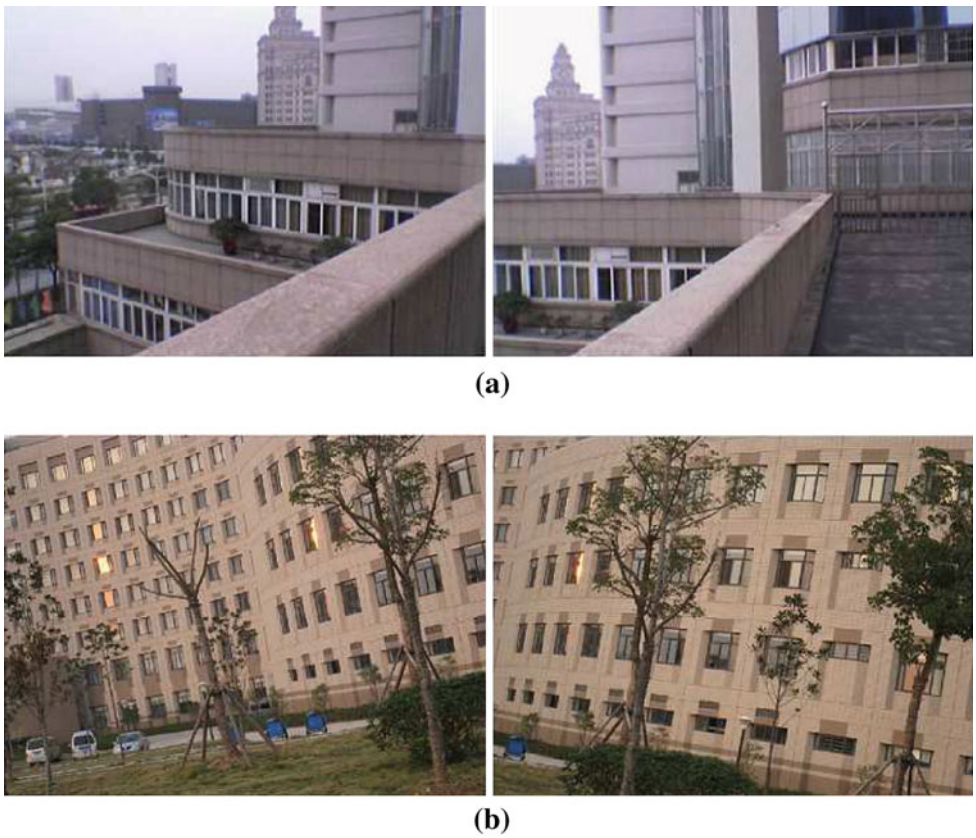

Fig. 2 The examples of relaxed motion. We can see that the positions of the corresponding structures in both image pairs have changed considerably due to large roll or pitch. These image pairs are difficult to be finely stitched with traditional methods. **a** Example of two 640×480 images. **b** Example of two $1,024 \times 768$ images

to compute and thus it is difficult to stitch them finely with traditional methods.

There are generally two types of image stitching methods: direct and feature-based method [1]. Direct method, such as [2–4], obtains the motion by directly minimizing the intensity difference. Some general methods, such as block matching algorithm (BMA) [5] and phase correlation [6], also fall into this category. Feature-based method, such as [7–10], refines motion with detected pixels (features), such as the features (e.g., Harris, Harris-Affine, SIFT) discussed in [11,12]. Both direct method and feature-based method have limitations. The direct method can easily end up to local optimum

in its intensity difference minimization, while feature-based method heavily relies on the distribution and salience of the features. This paper proposes an approach that effectively combines the merits of the two types of methods while avoiding the drawbacks.

Our approach is a multi-resolution stitching as it can update the camera parameters layer by layer and refine the calibration iteratively. For the coarsest layer, direct feature matching is unstable because of the limited features detected. A rough initialization obtained from direct method can be used to guide the feature matching. For the left layers, there may be many local optima when applying direct methods. But there are many different texture blocks which provides enough features for the feature-based method. Therefore, our multi-layer-based approach uses direct method in the coarsest layer to obtain the rough displacement field as the initial motion and uses feature-based method in the subsequent layers for refining the projective motion and camera parameters.

Feature matching is initialized by the estimated motion matrix and thus the estimation quality is very important for the further camera parameters computation. A robust motion estimation strategy is introduced to calculate motion and refine feature matching. In this strategy, a new adaptive BMA algorithm called TV-BMA is developed for the coarsest layer. Based on the regularized total variance (TV) by L^1-norm or $TV-L^1$ model, TV-BMA effectively computes the globally optimal displacement field with adaptively selected TV-scale images as the initial motion. In addition, two additional steps are taken in all layers to ensure an efficient projective motion: (1) A low contrast filter based on the edge response function is used to remove unstable matching pixels for the accurate localization of the matching features and (2) RANSAC is further used to remove the outliers and refine the motion.

The focal length and rotation matrix of each image can be estimated using the angle-invariant property of feature vectors and rotation-invariant property of feature matches. Formulated as a least squares problem, robust optimization method is the key to the convergence, where we also introduce a robust optimizer. The focal length is initialized by the golden section search and refined by the simplex method. The rotation matrix is obtained by the subspace trust region method.

This paper is organized as follows. After reviewing related work (Sect. 2), we will focus on our multi-resolution stitching method, i.e., the estimation of parameters in each pyramid layer. It consists of (1) the initial motion estimation algorithm TV-BMA for the coarsest layer (Sects. 2, 3) the remaining steps in the motion estimation strategy (the low contrast filter and RANSAC rectification) (Sects. 3, 4) the focal length estimation based on the angle invariance of feature vectors (Sect. 5) and the rotation calibration based on the rotation invariance of feature matches (Sect. 6). After the parameter estimation within each layer is discussed, Sect. 7 presents the proposed multi-resolution method and Sect. 8 presents experimental results. Discussions on this research are presented in Sect. 9 and the whole paper is concluded in Sect. 10.

2 Related work

Image stitching has a broad literature both in computer vision and computer graphics. In this section, we overview some algorithms that are closely related to our work. Interested readers can refer to [1] for more studies.

Perhaps the most important work is Szeliski et al. [13] which proposes a patch-based alignment to refine camera parameters with fine adjustment of the patch correspondences. However, there are two limitations in this study: (1) The patches are simply the square blocks evenly cut out from the image and thus this method produces huge numbers of patches or features and (2) the patch correspondences are built from the patch centers which might be of low contrast, flatly textured and illuminance-sensitive, and, therefore, the estimated motion can be unstable.

Zhou [14] proposed another approach, which in comparison with Szeliski et al. replaces the patches with limited number of features and applies a robust scheme based on the angle invariance of feature vectors and the rotation invariance of feature matches. A BMA based on illumination normalization (illumination-BMA) is also introduced to find the initial displacement field to refine feature matching. But the golden section search method to refine camera focal length is unstable because it lacks techniques to utilize the results from previous layer and can easily lead to local minimum. Their work does not state clearly which method is their best choice for rotation estimation among M-estimator, L-estimator, R-estimator and SVD. In this paper, we employ the similar idea from the illumination-BMA algorithm, the angle-invariant property and the rotation-invariant property. But we propose the simplex and the subspace trust region methods to refine the focal length and the rotation matrix, respectively.

Both approaches of Szeliski et al. and Zhou suffer from poor quality of the initial motion estimation which is very important as the initial guess for further refinement. In particular, when the neighboring images have a relatively large roll or pitch, the overlap area will have quite different pixels and thus it is difficult to reach global optimum. To counter this limitation, Chen et al. [15] propose the mutually exclusive scale component (MESC) to improve the initial motion estimation. MESC is built on the regularized total variance model $TV-L^1$ [16] and decomposes each image into several independent scale images (TV-scale images). In this manner, original one-pass matching of the whole image is turned into multi-pass matching with several TV-scale images. By

flattening the surface texture and thus retaining the object shape, it is much easier to find global optimal motion with TV-scale images than previous approaches for images with large roll and pitch.

Chen et al. work with satellite images, so the affine assumption of motion is valid. However, we are faced with a more challenging problem where images are captured with a hand-held camera of free projective motions. In addition, their registration method with three different MESC layers (scales) does not apply well in the general case where image resolution could vary considerably. Another disadvantage of Chen et al.'s approach is that the illumination normalization step is unstable because of the approximated reflectance. To overcome these limitations, we propose a new approach called TV-BMA, where TV-scale images with adaptive scale patterns are adopted to the illumination-BMA for estimating the horizontal and vertical displacements in the coarsest layer.

Unlike existing stitching studies, we propose two additional steps to refine the motions obtained from the previous layer. Since edge response function from Harris corner detector [17] has been proved to be an effective tool to remove low contrast features [12], we adopt it to remove low contrast pixels which might be flatly textured during the localization of matching features. To further improve the quality of feature matches, RANSAC [18], which has been proved to be very robust to remove outliers [8,19], is also incorporated.

Recently, there are two studies that also rely on the invariance properties of feature vectors and matches [20,21]. Our method avoids their complex parameterizations and instead iteratively refines the varying focal length and traditional 9-parameter rotation of the camera.

3 Motion initialization under $TV - L^1$ model

Assume two images I_i and I_j are the source images to be mosaicked. They have the same size and are decomposed into a $L-$layer image pyramid. All images $I_m^l (m \in \{i, j\}, 0 \leq l \leq L-1)$ on each pyramid are piled with increasing size from top to bottom and indexed from 0 to $L-1$. TV-scale images created from $TV - L^1$ model are utilized to find an optimal displacement field between the coarsest (top) layers of two source images. This displacement field is set to be the initial motion in the coarsest layer.

The MESC algorithm inspires TV-BMA and is built on the TV-scale image obtained from $TV-L^1$ model. Therefore, in the following, short introductions to the $TV-L^1$ model and its alternating solution will be given first. Then, MESC algorithm and traditional illumination-BMA algorithm will be reviewed briefly. We discuss the TV-BMA algorithm toward the end of the section.

3.1 The $TV - L^1$ model

Rudin, Osher and Fatemi (ROF) [22] first proposed the following constrained TV model for minimizing the total variation (TV) of the image I for its restoration or denoising. Defining the gradient of a gray image I as ∇I and its region as Ω yields the TV minimization problem

$$\begin{aligned} \min \quad & \int_{\Omega} \left\| \nabla \overline{I} \right\|_2 \\ \text{s. t.} \quad & \overline{I} + n = I \\ & \int_{\Omega} \|n\|_2^2 \leq \sigma^2 \end{aligned} \tag{1}$$

where $\overline{I}$ is the restored I without noise, $\int_{\Omega} \left\| \nabla \overline{I} \right\|_2$ is the total variance of image $\overline{I}$, n is the noise and σ^2 is an estimate of the noise variance in the image I.

In order to solve this problem, ROF and subsequent researchers considered the constrained minimization problem. Among them, Chan et al. [16] proposed the regularized L^1 functional, $TV - L^1$ model, which uses the L^1-norm as a measure of fidelity between the observed and denoised images

$$\min \quad \int_{\Omega} \left\| \nabla \overline{I} \right\|_2 + \lambda \left\| I - \overline{I} \right\|_1 \tag{2}$$

Two important properties in Eq. 2 make it especially attractive to us: (1) It can be used to extract different scale components according to different scales by setting different λ and (2) any particular pattern only exists in either $\overline{I}$ or n. For traditional Gaussian scale image or Laplacian pyramid, on the other hand, it is impossible to contain exclusive patterns, i.e., it is difficult to remove patterns with different scales. As demonstrated in [15], if there are different scales appearing in the same image, one may not end up with the desired solution. But an image can be decomposed by Eq. 2 progressively with each decomposition only representing patterns of one scale.

For images captured with relaxed motion, if the suitable comparison scale patterns exist in their respective TV-scale images, comparison between them then turns into comparing the representative shapes of flattened structures without having to look at quite different pixels. Therefore, such type of comparison can effectively avoid local optima. Since the illumination-BMA is robust in the traditional mosaic, we propose TV-BMA so that the TV-scale image with adaptive scale can be integrated with the illumination-BMA for motion initialization.

3.2 The efficient alternating solution to $TV - L^1$ model

The alternating algorithm proposed by Yang, Wang, Yin and Zhang [23–25] is used to solve the TV-regularization problem for recovering the images from blurred and noisy observations. This algorithm is fast and efficient. Its per-iteration mainly consists of several fast Fourier transforms, which is based on the half-quadratic technique proposed by Geman and Yang [26]. This algorithm can have different forms depending on 1-norm or 2-norm fidelity, and gray or color image. In the following, we will briefly review the techniques for recovering color image $\overline{I}$ with 1-norm fidelity, or with the $TV - L^1$ model. For more details of this algorithm, please refer to related work [23–25].

Let I_m be the identity matrix of order m, $\otimes$ be the Kronecker product and $(I_m \otimes D_i)\overline{I}$ be the first-order horizontal and vertical finite difference of $\overline{I}$ at pixel i. The discrete form of Eq. 2 is

$$\min_{\overline{I}} \sum_i \left\|(I_m \otimes D_i)\overline{I}\right\|_2 + \lambda \left\|K\overline{I} - I\right\|_1 \tag{3}$$

Equation 3 can be generalized as a local weighted $TV - L^1$-like model

$$\min_{\overline{I}} \sum_i \alpha_i \left\|G_i\overline{I}\right\|_2 + \lambda \left\|K\overline{I} - I\right\|_1 \tag{4}$$

where $\alpha_i > 0$ is a weighting parameter.

Let $z \in \Omega$ (Ω is the space of I) and $\mathbf{w}_i \in \mathbb{R}^q$ (q is the positive integer denoting the number of finite differences) be the auxiliary variables that approximate $K\overline{I} - I$ and $G_i\overline{I}$ in Eq. 4. According to the half-quadratic technique [26], Eq. 4 can be approximated by

$$\min_{w,z,\overline{I}} \sum_i \left(\alpha_i \|\mathbf{w}_i\|_2 + \frac{\beta}{2}\left\|\mathbf{w}_i - G_i\overline{I}\right\|_2^2\right) + \lambda\left(\|z\|_1 + \frac{\mu}{2}\left\|z - \left(K\overline{I} - I\right)\right\|_2^2\right) \tag{5}$$

where β and μ are the penalty parameters.

Equation 5 can be easily minimized by an iterative and alternating approach due to the fact that with any two of the three variables $\mathbf{w}$, z and $\overline{I}$ fixed, the minimizer of Eq. 5 with respect to the third one has a closed-form formula to compute. Especially to obtain $\overline{I}$, Yang et al. [25] reformulated this equation with special block circulant structure which can be obtained by a few two-dimensional discrete Fourier transforms and arithmetic operations. This approach is numerically stable for large values of β and μ. It also converges to a solution for any fixed $\beta, \mu > 0$. Therefore, in our current work, this alternating solution to the $TV - L^1$ model is adopted to obtain the TV-scale image.

Next, we will briefly introduce the MESC algorithm which is based on the $TV - L^1$ model and closely related to TV-BMA.

3.3 The MESC algorithm

MESC approach [15] registers images through decomposing an image into mutually exclusive scale components (MESC) based on the $TV - L^1$ model. A pattern in the original image only appears in one of these components because $TV - L^1$ model can generate different patterns at different scales as we discussed earlier. With those scale-exclusive patterns, the alignment of the original image pair turns into aligning corresponding layers independently and choosing the optimal transformation of all corresponding layers.

This algorithm works on three different scales to find the optimal transformation: (1) the image contains the large scale patterns obtained with a small λ_1, I_{s_1}; (2) the image contains the medium scale patterns obtained with a larger λ_2, I_{s_2}; and (3) the image contains the remaining small scale patterns obtained by $I_t - I_{s_2}$, I_{s_3}, where I_t is the illumination-normalized image generated by $\frac{I}{I_{s_1}}$. Using the correlation ratio as similarity metric, the algorithm iteratively works as follows:

1. Initialize the transformations of all corresponding scale images to be T_s;
2. Find the transformations T_1, T_2 and T_3 in the three corresponding scale images I_{s_1}, I_{s_2} and I_{s_3}, respectively, by Powell's local minimization method ;
3. Select the optimal transformation T_o among T_1, T_2, T_3 and T_s using the correlation ratio as the similarity metric [27];
4. Return to Step 1 and set $T_s = T_o$ if the algorithm does not converge.

The main problem of MESC is that only three scales are used without considering the vast scale variations in the real images. Therefore, we propose TV-BMA which adaptively selects a suitable scale for each image. Before TV-BMA is presented, the traditional illumination-BMA which is also the building block of TV-BMA is reviewed briefly as follows.

3.4 The Illumination-BMA

The illumination-normalized BMA [14] first computes the intensity differences between the observed region Ω_o and the overlapped matching region Ω_m during sliding Ω_m over Ω_o. Then it selects the displacement field associated with the minimum among all intensity differences.

Let the corresponding pixel in Ω_o and Ω_m be p_o and p_m, respectively, and the size of the overlapping area be $W \times H$. Then the intensity difference between Ω_o and Ω_m proposed by Zhou [14] is

$$e_{o,m} = \sum_{i,j}\sum_{k} \frac{\left|(p_{o_{i,j,k}} - p_{o_{\text{mean}}}) - (q_{m_{i,j,k}} - q_{m_{\text{mean}}})\right|}{W * H} \quad (6)$$

where $p_{o_{i,j,k}}$ and $q_{m_{i,j,k}}$ are the value of the kth color channel for p_o and p_m in the location (i, j), respectively. $p_{o_{\text{mean}}}$ and $q_{m_{\text{mean}}}$ are the average brightness of Ω_o and Ω_m, respectively, which are used to normalize the illumination.

3.5 The TV-BMA

Built on $TV - L^1$ model, MESC algorithm and illumination-BMA, the TV-BMA works in the following way. Illumination-BMA is applied first to the images in the coarsest layer to compute an optimal displacement field. If the intensity difference corresponding to the displacement field is less than the predefined threshold, this displacement field is used as the initial displacement field. Otherwise, there may be relatively large roll or pitch, and then an iterative estimation step is applied to find the optimal displacement field with suitable TV-scale images. In this step, progressively scale-decreased TV-scale images are fed into the illumination-BMA and the iteration process stops when an optimal displacement field is found or the maximum number of iterations has been reached. Denoting $illumBMA(I_1, I_2))$ as the illumination-BMA algorithm for images I_1 and I_2, M as the optimal displacement field obtained and $\bar{e}$ as the intensity difference corresponding to the optimal displacement field, we can illustrate TV-BMA in Algorithm 1.

Algorithm 1 The TV-BMA algorithm

```
[M, ē] = illumBMA(I1, I2)
IF (ē < e1) THEN Output M
i = 1
REPEAT
  I1 = Tv(I1, λ)
  I2 = Tv(I2, λ)
  [M, ē] = illumBMA(I1, I2)
  λ = αλ
  i = i + 1
UNTIL ((ē < e2) ‖ (i > t))
IF i ≥ t THEN Output error_message
ELSE Output M
```

In the above algorithm, e_1 and e_2 correspond to the error thresholds of the illumination-BMA and TV-BMA, and t is the max iteration times. In practice, e_1 and e_2 are set to be 0.1 and 0.2, respectively, and t is set to be 10. α is the parameter to generate different TV-scale images and, in our experiments, it is a constant and set to be 1.5. λ is initialized to be $1.5/image\ scale$, while *image scale* is defined as in [15]. $Tv(I, \lambda)$ is the function to obtain the TV-scale image by Eq. 2.

In addition, for the illumination-BMA, in order to boost the intensity difference between Ω_o and Ω_m to better distinguish different blocks than with Eq. 6, the intensity difference $e_{o,m}$ is reformulated with squared differences as

$$e_{o,m} = \sum_{i,j}\sum_{k}\left(\frac{(p_{o_{i,j,k}} - p_{o_{\text{mean}}}) - (q_{m_{i,j,k}} - q_{m_{\text{mean}}})}{W * H * C}\right)^2 \quad (7)$$

where C is the number of the color channels.

4 Motion refinement

After computing the displacement field from TV-BMA as the initial motion of the coarsest layer, we perform an iterative process for layer-by-layer-based camera calibration refinement to stitch the source images. We introduce a motion refinement process to refine the motion for the feature detection in the next layer and simultaneously remove outliers for the subsequent camera calibration in the current layer.

4.1 The motion refinement and its embedded problem

Assume that the current layer is the cth layer. First, the features in I_i^c are detected by a corner detector. Then matching features in I_j^c are initialized by two different methods. If current layer is the coarsest layer, i.e., $c = 0$, each matching feature in I_j^c is located by the initial displacement field calculated with TV-BMA. If the cth layer is not the coarsest layer, i.e., $c \geq 0$, the features in I_j^c are located by the homography matrix computed in the previous layer.

The matching process may work pixel by pixel to locate each matching feature like in [14]. For each feature correspondence, first a source 8×8 block is defined for the feature in I_i^c with this feature as the block center, then a target 16×16 block is defined for the matching feature in I_j^c with that feature as the target block center. By translating the same sized block as the source block in the target block, illumination-BMA refines the position of the matching feature in sub-pixel accuracy.

The above process has two problems. One is that it is time-consuming since it checks every pixel in target block to localize the refined position, and some pixels' contrast is too low to check at all. The other problem is that outliers may exist because the inaccurate motion estimated from the previous layer and the exposure difference can misguide the pixel selection in the target block.

To tackle the first problem, a geometrical low contrast filter based on the corner response is applied to remove the low contrast pixels. There is no need to check the low contrast pixels because they are flatly textured surface points inside the target block. The removal of these unstable pixels before block matching improves the speed as well as the accuracy. The other problem can be solved by RANSAC method

which also obtains the correct projective motion for camera calibration.

4.2 The low contrast filter

The low contrast filter is built on the edge response function used in Harris corner filter. For each pixel $p(x, y)$, its Hessian matrix H is

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

The corner pixel has two large eigenvalues. In practice, we use the improved corner response function proposed by Noble [28]

$$h = \frac{\det(H)}{\mathrm{tr}(H)} = \frac{D_{xx} * D_{yy} - D_{xy}^2}{D_{xx} + D_{yy}} \tag{8}$$

where $\det(H)$ and $\mathrm{tr}(H)$ denote the determinant and trace of H, respectively. Large h implies the pixel is likely to be a corner, while low h indicates the pixel is of low contrast. By setting the threshold of h, the low contrast pixels in the target block will be excluded for block matching. In the Gaussian pyramid, the contrasts of pixels in the coarse image are lower than those in the fine image, so the threshold of h should be adjusted accordingly. Empirically, the threshold is set to be $30/\sigma^{L-l}$ for the lth layer in the L-layer Gaussian pyramid used in the paper.

4.3 Outlier removal

There are likely outliers from the inaccurate motion or the illumination difference after applying the low contrast filter. We apply RANSAC to check the globally geometrical consistency by computing an optimal projective homography as well as removing outliers. In our method, RANSAC implementation of Kovesi [29] is applied, which is based on the idea of Hartley and Zisserman [30]. In this implementation, two robust methods are adopted to obtain the homography as well as feature matches: (1) homography is computed by SVD and (2) a symmetric distance metric is used to select matches corresponding to a putative homography, where each matching feature is transformed to its matched feature space in distance computation.

After RANSAC, on one hand, the homography obtained is used to detect the initial feature correspondences in the next layer, which will be refined again by RANSAC to remove outliers and improve homography repeatedly. On the other hand, the focal length and the rotation matrix of each layer can be robustly estimated based on the angle invariance of feature vectors and the rotation invariance of feature matches, respectively. In the following, the methods for focal length and rotation matrix estimation are discussed.

5 Focal length estimation

We first explain the angle invariance of feature vectors, as demonstrated in Fig. 3. For the clarity of description, the camera coordinate is shown in Fig. 3a. According to this figure, the image coordinate of I is denoted as (O, X, Y) and the camera coordinate is denoted as (o, x, y) with the optical axis passing through the image center. Assuming the focal length is f, the pixel coordinates of 3D features A and B, A' and B', are $(x_a, y_a, -f)$ and $(x_b, y_b, -f)$, respectively and the angle between the feature vector $\overrightarrow{oA}$ and $\overrightarrow{oB}$ is θ. If A and B are captured in several images, θ will remain the same as shown in Fig. 3b. In this figure, A and B are imaged in I_i^c and I_j^c as A_1, B_1, and A_2 and B_2, respectively. The focal lengths for each image are f_i and f_j with O_i and O_j being the image centers. The coordinates of A_1, B_1, A_2 and B_2 can be written as $(x_{a1}, y_{a1}, -f_i)$, $(x_{b1}, y_{b1}, -f_i)$, $(x_{a2}, y_{a2}, -f_j)$ and $(x_{b2}, y_{b2}, -f_j)$. Denoting the vector angles between $\overrightarrow{oA_1}$ and $\overrightarrow{oB_1}$ as $\theta_{A_1B_1}$, and between $\overrightarrow{oA_2}$ and $\overrightarrow{oB_2}$ as $\theta_{A_2B_2}$, we have

$$\theta_{A_1B_1} = \theta_{A_2B_2} \tag{9}$$

Equation 9 formulates the angle invariance of feature vectors. When there are N pairs of feature matches, $\left(N^2 - N\right)/2$ of angle pairs satisfying Eq. 9 can be obtained. Denoting $\theta_{i,c,k}$ and $\theta_{j,c,k}$ as the kth vector angles in I_i^c and I_j^c, respectively, we have

$$\sum_{k=0}^{(N^2-N)/2} \theta_{i,c,k} = \sum_{k=0}^{(N^2-N)/2} \theta_{j,c,k} \tag{10}$$

Equation 10 can be solved by the weighted minimization method proposed in [14] to accelerate the convergence. Setting the weight w_k to be the total length of the kth corresponding feature vectors yields a minimization problem

$$e(f_i, f_j) = \sum_{k=0}^{(N^2-N)/2} w_k \left\| \theta_{i,c,k} - \theta_{j,c,k} \right\|_2 \tag{11}$$

The focal lengths for two images can be set to be equal if we use the associative property of matrix multiplication to transfer Eq. 12 shown in Sect. 6 with the same unknown focal length f. Therefore, f_i and f_j in Eq. 11 turn to be one focal length f to be estimated: in the coarsest layer, i.e., $c = 0$, f is estimated by golden section search with an initial search range; in the non-coarsest layer, i.e., $c > 0$, f is estimated by the simplex method with the initial value obtained from the previous layer. In practice, the initial search range is set to be $[0, 100 * max(image height, image width)]$.

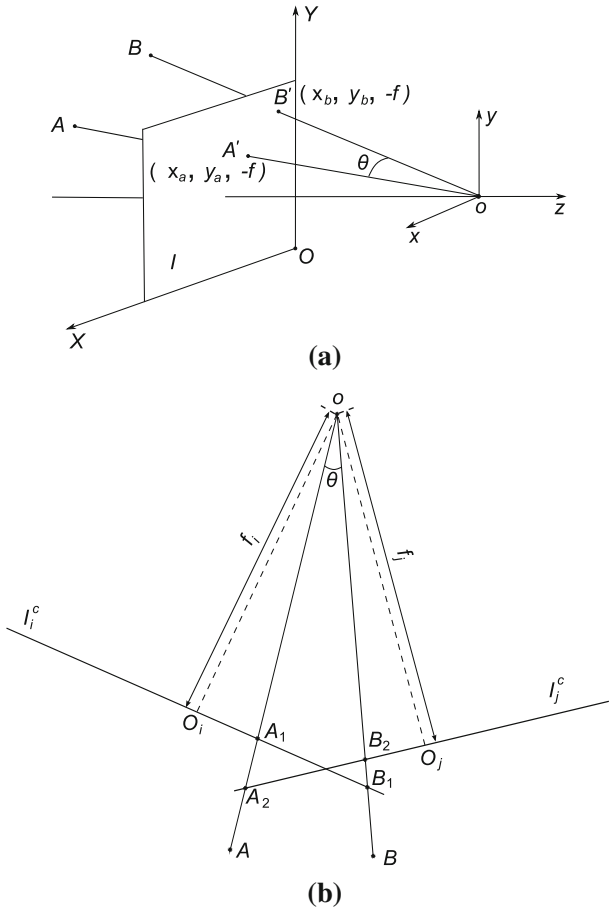

Fig. 3 Principle of the focal length and rotation matrix estimation. **a** Camera coordinate system. **b** The angle and rotation invariances between an image pair

6 Rotation estimation

After the focal length is obtained, the rotation matrix can be estimated by the rotation-invariant property of feature matches. First we explain this invariance property with Fig. 3b. If I_i^c rotates with a 9-parameter R to be I_j^c, for matching features A_1 and A_2, the relationship between A_1 and A_2 is

$$\begin{bmatrix} x_{a1} \\ y_{a1} \\ -f_i \end{bmatrix} = R \begin{bmatrix} x_{a2} \\ y_{a2} \\ -f_j \end{bmatrix}, \quad R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{12}$$

Equation 12 holds for B_1 and B_2 as well as other feature matches. This property is called rotation invariance of feature matches.

Similar to the focal length estimation method, we have

$$\sum_{k \in \mathcal{N}_{i,j}} p_{i,c,k} = \sum_{k \in \mathcal{N}_{i,j}} R p_{j,c,k} \tag{13}$$

where $\mathcal{N}_{i,j}$ is the total number of feature matches between I_i^c and I_j^c and $p_{i,c,k}$ and $p_{j,c,k}$ are the positions of the kth corresponding features.

The solution of the rotation matrix to Eq. 13 can also be written as a least squares problem with all feature matches considered as Eq. 11 for focal length estimation. But this time, the weight used in Eq. 11 is omitted since it will add to the sensitivity of the iterative solving process. Therefore, the following error function is to be minimized to find the rotation between I_i^c and I_j^c:

$$e(R) = \sum_{k \in \mathcal{N}_{i,j}} \left\| \text{norm}(p_{i,c,k}) - \text{norm}(R p_{j,c,k}) \right\|_2 \tag{14}$$

Function norm(v) is used to normalize the vector v. Equation 14 is solved by the subspace trust region method [31,32], which will be discussed in the following two sub-sections.

6.1 The subspace trust region method

The trust region method [31] is a class of optimization algorithms that replaces directly minimizing $f(x)$ with minimizing a simpler quadric function $q(x)$. $q(x)$ reasonably reflects the behavior of $f(x)$ in the neighborhood area Ω_n around the point x. This neighborhood area is called the trust region. The quadric $q(x)$ is defined by the first two terms of the Taylor series and Ω_n is usually spherical or ellipsoidal in shape. Let s be the trial step over Ω_n, this trust region subproblem that obtains $s = x_{k+1} - x_k$ can be written as

$$\min_s \left\{ \frac{1}{2} s^T H_k s + g_k^T s : \| Q_k s \|_2 \leq \Delta_k \right\} \tag{15}$$

where g_k is the gradient of f at the current point x_k, H_k is the Hessian matrix, Q_k is a diagonal scaling matrix, and Δ is a positive scalar. If $f(x_k + s) < f(x_k)$, the next point x_{k+1} is updated to be $x_k + s$; otherwise, it remains unchanged and Ω_n is shrunk for the next update.

Solving Eq. 15 in a reliable and efficient way is a non-trivial task because it can easily converge to a local minimum. One method is to replace the full dimension trust region with a lower dimension subspace, whereby local minimum problem can be alleviated and computing complexity is reduced.

In our experiments, we use the non-linear optimization function provided in MATLAB as the implementation of the subspace trust region method. It uses the two-dimensional (2D) subspace approach [33] where the 2D subspace is determined with the aid of a preconditioned conjugate gradient process, which forces global convergence via the steepest descent direction or negative curvature direction and achieves fast local convergence via the Newton step. The MATLAB optimization function is based on the interior-reflective Newton method described in [32]. The interior-reflective

Newton method does not require the solution of a general quadratic programming subproblem at each iteration and is very robust with respect to its convergence.

6.2 The initialization of the subspace trust region method

To apply the subspace trust region method, a good initial value of rotation is very important to ensure the convergence. We adopt the singular value decomposition (SVD) approach proposed by Umeyama [34]. This idea comes from the least-square method, which will be discussed briefly in the following. For more details, please refer to Umeyama [34].

According to Eq. 12,

$$p_{i,c,k} - Rp_{j,c,k} = 0 \tag{16}$$

For all feature pairs, we can obtain the sum, $e(x)$, of the squared residuals

$$e(R) = \sum_k \left(p_{i,c,k} - Rp_{j,c,k}\right)^T \left(p_{i,c,k} - Rp_{j,c,k}\right) \tag{17}$$

The least-square method finds the minimum of e, $\min e(R)$.

According to Umeyama [34], Eq. 17 can be further written as

$$e(R) = \sum_k p_{i,c,k}^T p_{i,c,k} - 2p_{i,c,k}^T Rp_{j,c,k} + p_{j,c,k}^T p_{j,c,k} \tag{18}$$

while

$$\begin{aligned} p_{i,c,k}^T Rp_{j,c,k} &= \sum_k \mathrm{tr}\left(R^T p_{i,c,k} p_{j,c,k}^T\right) \\ &= \mathrm{tr}\left(R^T \sum_k p_{j,c,k} p_{i,c,k}^T\right) \end{aligned} \tag{19}$$

Denoting

$$M = \sum_k p_{j,c,k} p_{i,c,k}^T \tag{20}$$

as the correlation matrix, we can write the singular value decomposition of matrix M as

$$M = uwv^T \tag{21}$$

where u and v are the orthogonal matrices and w is the diagonal matrix containing the singular values of M.

Umeyama [34] proves that the optimal rotation matrix R which minimizes $e(R)$ is uniquely determined when $rank(M) \geq m - 1$ (m denotes the number of row or column of the $m \times m$ square matrix R)

$$R = u \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det\left(uv^T\right) \end{bmatrix} v^T \tag{22}$$

Equation 22 shows the R is computed with the orthogonal matrices from the SVD of M. In practice, $rank(M) \geq m-1$, therefore, Eq. 22 is used as the initial rotation matrix for the subspace trust region method. This method of initialization is more accurate and convenient than manually specifying an initial value.

Given the subspace trust region method and its initialization, the method of rotation matrix estimation can be generalized as follows. For the coarsest layer, i.e., $c = 0$, the initial rotation can be obtained by Eq. 22. For the non-coarsest layers, i.e., $c > 0$, their rotation matrices are computed using the initial value obtained from the previous layer.

After the rotation in the current layer is obtained, the iteration of stitching refinement with images of next layer continues. When the computation of the rotation of the bottom layer is finished, the source images can be finally registered.

7 Summary of the proposed method

After above discussions, we now summarize the proposed image mosaic method for two images I_i and I_j as follows.

Step 1 Decompose each image into an $L-$layer multi-resolution pyramid with the coarsest layer as layer 0;
Step 2 Set current layer index $c = 0$;
Step 3 Obtain the initial motion M between I_i^c and I_j^c with TV-BMA (Algorithm 1);
Step 4 Repeat the following steps until $c > L - 1$,
  Step 4.1 Find the features in I_i^c and calculate the corresponding features in I_j^c under M with the low contrast filter;
  Step 4.2 Remove the outliers and calculate the new motion M by RANSAC;
  Step 4.3 If $c = 0$, calculate the focal length according to Eq. 11 with the golden section search method and then go to Step 4.5; otherwise go to Step 4.4;
  Step 4.4 Calculate the focal length according to Eq. 11 with the simplex method;
  Step 4.5 If $c = 0$, initialize the rotation R with SVD decomposition according to Eq. 22 and then go to Step 4.7; otherwise go to Step 4.6;
  Step 4.6 Refine R according to Eq. 14 with the subspace trust region method;
  Step 4.7 $c = c + 1$.

8 Experimental results

We now present our experimental results. Our approach is implemented in MATLAB. In all experiments, the Gaussian pyramid is created to represent the multi-resolution images. The pyramid creation process stops when the width or the height of a layer is less than 50 pixels, which will be the top layer.

Harris corner detector [29] is used to detect corners as features for each layer. The parameters of the detector are set as follows: the standard deviation of smoothing Gaussian starts as 1 in the coarsest layer and increases by a factor of 1.5 for each subsequent layer; the region radius for the non-maximal suppression is set to be 1 in the coarsest layer and increments by 1 for each subsequent layer; the threshold for the non-maximal suppression is set to be 5 in the coarsest layer and 300 for all other layers. The design goal of the settings is twofolded:: (1) It can obtain large number of features in the coarsest layer to compute a robust initial motion; (2) it also helps to remove unstable and low contrast pixels according to the increasing resolution of each layer. Moreover, if either matching feature is within 3 pixels of image border, the feature correspondence pair will be excluded from further motion refinement, in order to ensure sufficient neighboring area for comparing the similarity of a feature pair.

Two types of experiments are performed with all source images warped to the cylindrical surface to show the stitching result. Type 1 shows the process of mosaicking two neighboring images; type 2 shows the stitching of multiple images. While the first type is to show the basic idea presented in this paper, the second type shows the extension of our method to wide-angle mosaic of an image sequence. In our experiments, blending after the registration is not discussed because our main focus in this paper is to improve the registration accuracy, especially when the source images have relaxed motion. Therefore, in the stitched image, we sum the overlapping areas from warped source images directly with equal weights to demonstrate the registration performance. We believe with accurate registration, fine blending can be achieved by existing blending methods, such as those discussed in [1].

8.1 Two images

We first demonstrate our method with the image pair in Fig. 2a. Figure 4a and b are the 5th 40 × 30 layers of the Gaussian pyramids, respectively. The intensity differences of illumination-BMA according to different displacement fields are visualized as a mesh in Fig. 4c. As labeled *minimum* in the figure, the final output displacement is (36, 23). But this value is incorrect because there are many other similar local minimums. As Fig. 4a and b show, Gaussian scale image contains different scale objects mixed together and thus those two images cannot initialize a correct displacement field with the illumination-BMA.

But, according to TV-BMA, if we remove the textures inside the objects, such as the bell tower, the walls and the windows, through the TV-scale image, the general structure is much clearer and the illumination-BMA can be applied successfully. In the TV-BMA, the optimal displacement field is computed with only one iteration where λ is set to be 0.1750. The TV-scale images of Fig. 4a and b are shown in Fig. 5a and b, respectively. Clearly we can see large flattened patterns shown in Fig. 5a and b are similar in texture and shape. As Figs. 4c, 5c shows the mesh view of the intensity difference according to different displacement fields for these two TV-scale images. We can see the intensity difference of TV-BMA is very smooth with few noisy local optima. Therefore, the global minimal position (17,−6) is easier to obtain than with the illumination-BMA.

Table 1 shows the proportion of the low contrast pixels removed in each layer. *layer* means the pyramid layer. *pixels compared* means the average number of pixels compared inside the target block of the right image for one feature detected in the left image. *pixels removed* means the average number of pixels removed in the target block because of low contrast. *ratio* shows the ratio of *pixels removed* to *pixels compared* in percentage. Feature correspondences less than 2 pixels away from the border are removed from further low contrast filtering because those features do not have adequate overlapping areas to check their similarity and thus are unstable to be putative matches. Therefore, *pixels compared* is always less than 256 because there are less than 256 candidate positions in average during the refinement process. When the resolution is low, such as layer 0 and 1, few features are detected and those near-border features will have a large ratio in the total number of pixels to compare. In this case, we have a relatively small number of *pixels compared*, e.g., in layer 0, only 204.5 pixels are compared in average for one feature. However, as the resolution goes higher, such as layer 3 and 4, the proportion of feature correspondences along the border to all feature correspondences decreases. Then a higher *pixel compared* is obtained, e.g., almost 256 pixels compared (253.6727) for layer 4. As we can see, in each layer, about 25% of pixels are removed from block matching and thus the low contrast filter greatly improves the feature matching speed.

Table 2 shows the RANSAC rectification result in each layer. *putative matches* is the number of feature matches after the low contrast filter. *outliers* is the number of outliers detected by RANSAC. *ratio* shows the ratio of *outliers* to *putative matches* in percentage. From this table, we can see that the proportion of outliers is reduced as the layer index goes up since the motion parameters have been gradually improved by the low contrast filter and RANSAC. Figure 6 shows the 108 feature matches (inliers) obtained finally in the bottom layer (source images).

Focal lengths estimated and updated in each layer are shown in Table 3. The final stitching result obtained is shown in Fig. 7a with the green frame showing the overlapping area. Simply generated by averaging pixel values from both images, the overlapping area has no ghosting, which demonstrates the effectiveness of our method.

Finally, we derive the roll, pitch and yaw, respectively, from the recovered rotation matrix: −3.9739°, −6.1362°

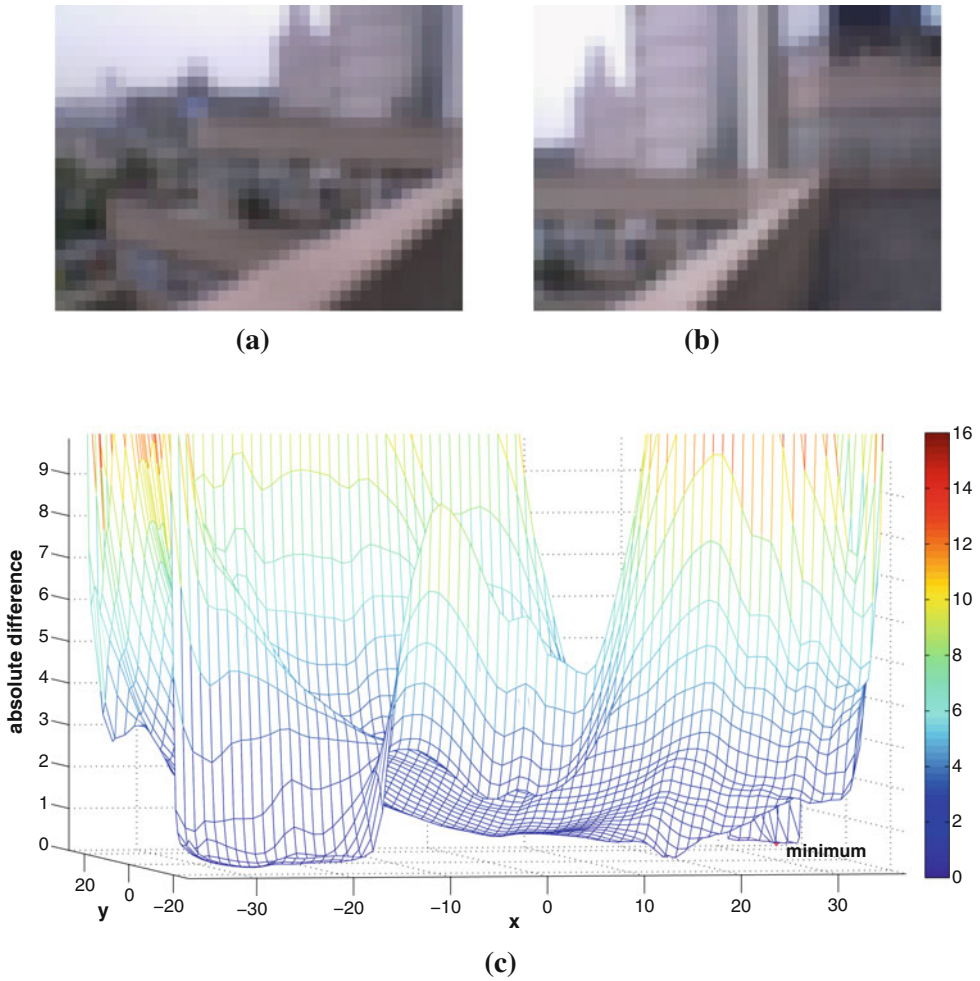

Fig. 4 The illumination-BMA approach for images in Fig. 2a. **a** The coarsest layer in the pyramid of the left image. **b** The coarsest layer in the pyramid of the right image. **c** The mesh view of the intensity difference of the overlapping area between Fig.4a and b under different displacement fields. The *x* and *y* axes represent the displacements of Fig. 4a and b in horizontal and vertical directions, respectively. *Minimum* corresponds to the optimal displacement field obtained finally

and 22.0206°. It is difficult for traditional BMA such as illumination-BMA to find the displacement field with such a large roll and pitch.

We also apply our method to images in Figs. 2b, and 7b shows the stitching result. In Fig. 7b, there is also no ghosting in the overlapping area. The roll, pitch and yaw rotation angles are 19.4849°, −3.8155° and 10.0586°, respectively.

8.1.1 Comparison with existing techniques

We compare our method to off-the-shelf techniques for the images shown in Figs. 2a and b. The Panorama Factory [35] is selected among many commercial products because it is highly rated and has an easy-to-use trial version. The latest trial version 5.3 is used for comparison. The popular open-source software Hugin [36] is also used as the state-of-the-art research for performance comparison. Developed through world-wide collaboration, Hugin incorporates a number of robust algorithms for image registration and panorama creation. The latest version 0.7.0 is used in our experiments.

The overlapping area in the final stitched image is used for comparison between The Panorama Factory, Hugin and ours. As described before, the overlapping area is obtained with equal weights from both warped source images so that the ghostings coming from inaccurate registration can be clearly seen. This method works fine for Hugin and our method since the warped source images of these two techniques are available. However, The Panorama Factory trial version does not proVIDE the warped source images, while the stitched result is already blended. Therefore, we have to make a detour to manually clip the overlapping area from the blended mosaic. Fortunately, as our experimental result shown below, it does not affect the visual judgment of the stitching quality because The Panorama Factory has the worst performance among them and its artifacts can be easily seen.

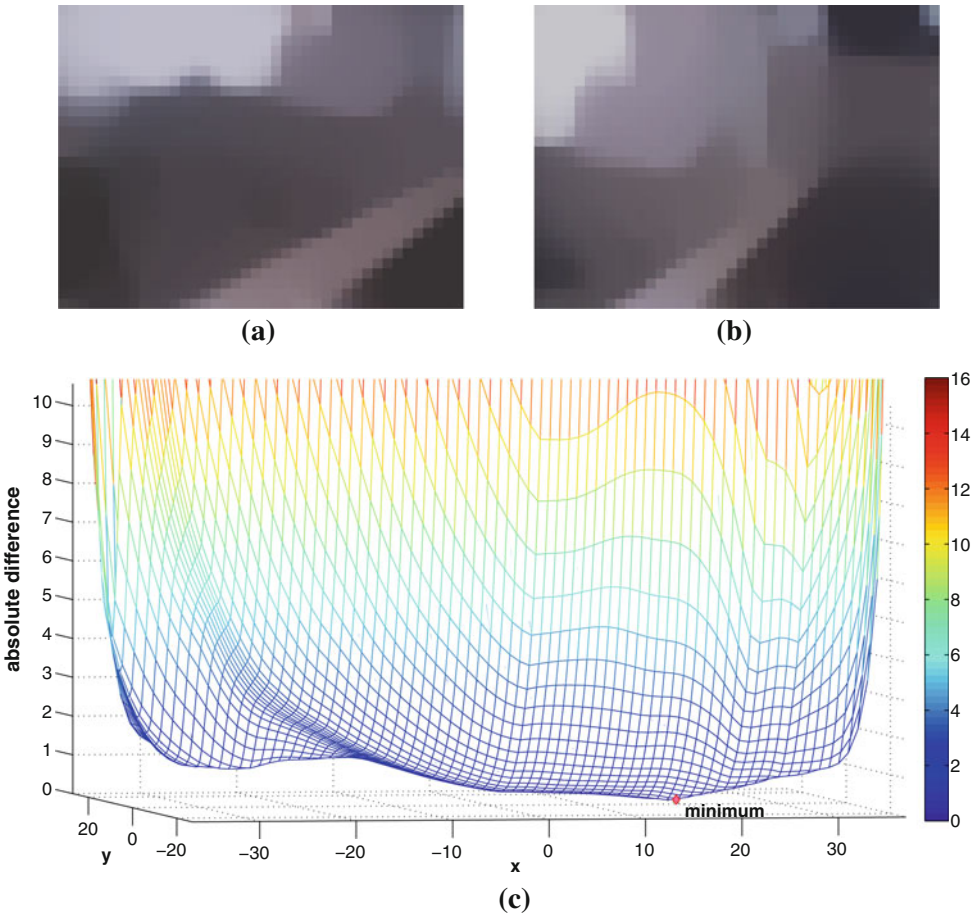

Fig. 5 The TV-BMA approach for images in Fig. 2a. **a** The TV-scale image of Fig. 4a with $\lambda = 0.1750$. **b** The TV-scale image of Fig. 4b with $\lambda = 0.1750$. **c** The mesh view of the intensity difference of the overlapping area between Fig. 5a and b under different displacement fields. The x and y axes represent the displacements of Fig. 5a and b in horizontal and vertical directions, respectively. *Minimum* corresponds to the optimal displacement field obtained finally

Table 1 The average pixels removed by the low contrast filter

Layer	0	1	2	3	4
Pixels compared	204.5000	227.6667	245.8667	247.6522	253.6727
Pixels removed	42.6250	81.1111	70.9000	73.6522	66.8727
Ratio	20.8435	35.6371	28.8368	29.7402	26.3618

Table 2 The outliers removed with RANSAC

Layer	0	1	2	3	4
Putative matches	8	18	30	69	110
Outliers	3	11	2	3	2
Ratio	37.5000	61.1111	6.6667	4.3478	1.8182

Figure 8 shows the overlapping area obtained for images in Fig. 2a and b after they are registered. Comparing these two figures, we can find that The Panorama Factory exhibits significant ghostings. Hugin generates slight ghostings at the top of the wall for Fig. 2a and serious ghostings for Fig. 2b, while there is no ghosting with our method for both figures. Therefore, our method achieves better performance than The Panorama Factory and Hugin.

Besides these two exemplar comparisons, more experiments with two images were made to test the performance of

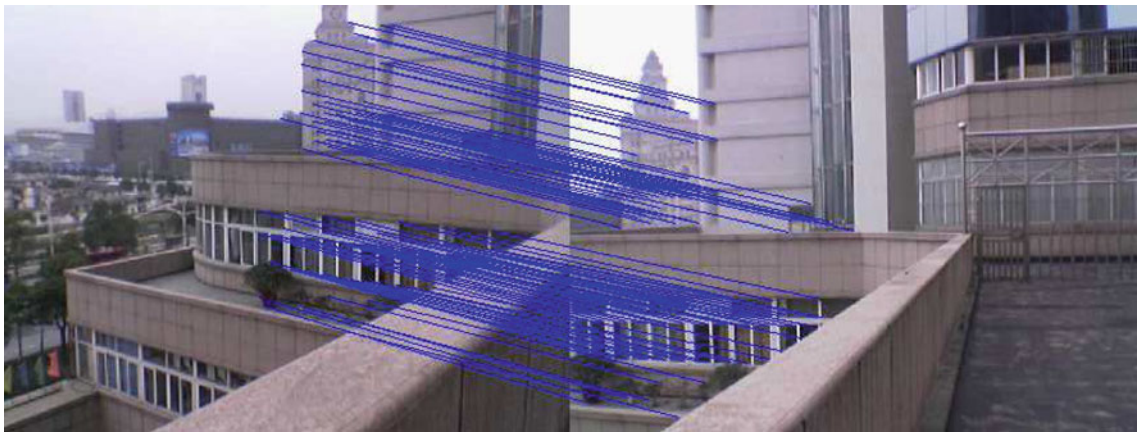

Fig. 6 The 108 feature matches obtained in the bottom layer for images of Fig. 2a

Table 3 The focal length estimated

Layer	0	1	2	3	4
Focal length	48.4642	84.9531	176.6588	350.3276	694.4687

our method and some of those examples will be discussed in the next sub-section to prove the advantages of our method.

8.1.2 Additional examples

Figure 9 gives five additional image pairs which are compared in the same way as in Fig. 8. Indexed from 0 to 9 for source images in Fig. 9a, the overlapping area of the stitching result from The Panorama Factory, Hugin and our method are shown, respectively, in Fig. 9b–d. Their relative rotations finally obtained from TV-BMA and λ used in TV-BMA are listed in Table 4.

Like the comparison presented in Fig. 8, we can find in Fig. 9 that the worst performance is again from The Panorama Factory where no image pair can be smoothly stitched. All the motions computed from this tool are less accurate than from other two methods, indicating that The Panorama Factory is not able to handle relaxed motion well.

We can also see that our method achieves better performance than Hugin for relaxed motion when pitch and roll are large, but not too large, as in image pairs (0, 1) and (2, 3). In the stitching results of these two image pairs, there is no ghosting with our method but there are still considerable ghostings with Hugin. The satisfactory result is attributed to our multi-resolution stitching pipeline, especially TV-BMA in one iteration (see λ shown in Table 4) which obtains an accurate initial motion for later multi-resolution stitching.

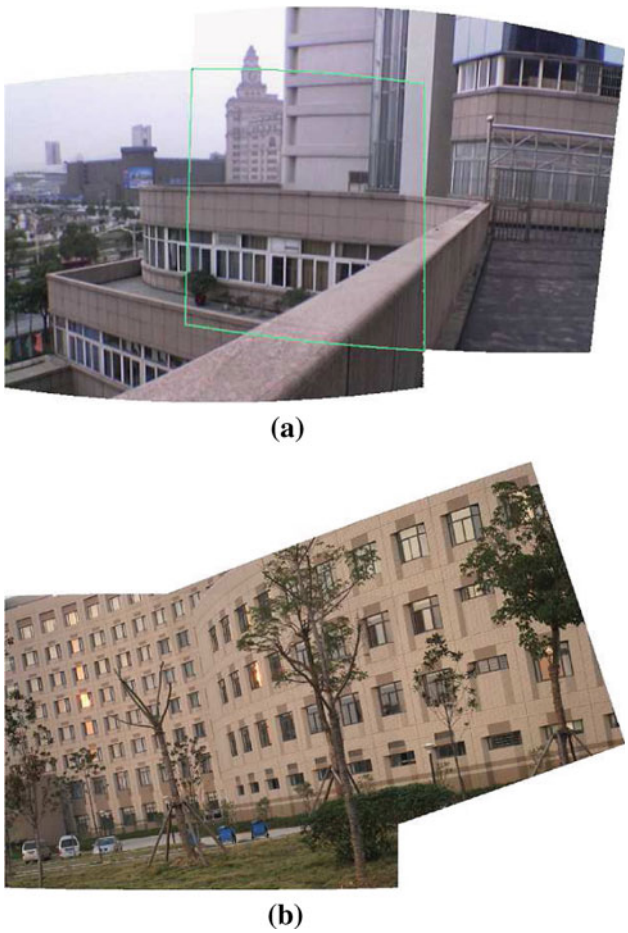

Fig. 7 The stitching result of Fig. 2a and b. **a** The stitching result of Fig. 2a. **b** The stitching result of Fig. 2b

When the roll angle or pitch angle become even larger, both Hugin and our method do not register images very well. Yet as image pair (4, 5) and (6, 7) shown in Fig. 9c and d, TV-BMA in one iteration (see λ shown in Table 4) can still

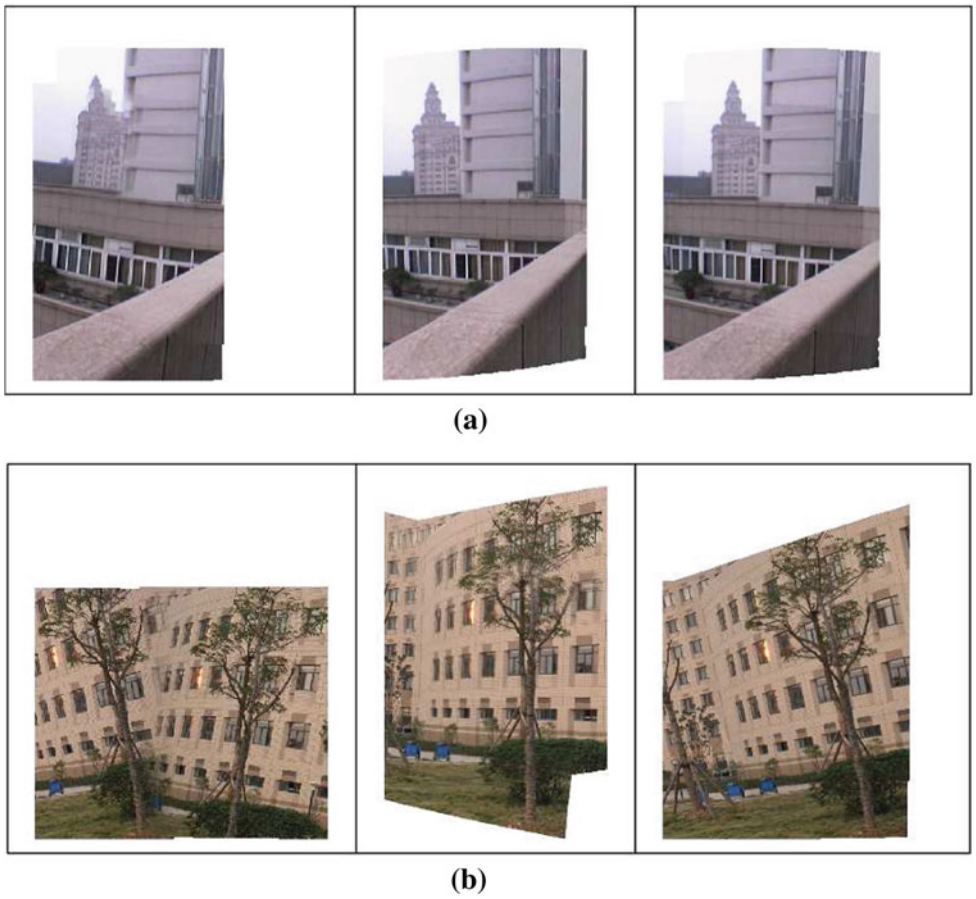

Fig. 8 Performance comparison between The Panorama Factory, Hugin and our method. The overlapping areas of stitched images from Fig. 2a and b are compared. The overlapping area from The Panorama Factory is cut from the blended mosaic while the overlapping area from Hugin and ours is an average of the source images. **a** The overlapping areas for images in Fig. 2a with The Panorama Factory, Hugin and our method, from left to right. **b** The overlapping areas for images in Fig. 2b with The Panorama Factory, Hugin and our method, from left to right

obtain a more robust initial displacement field and lead to more accurate registration than Hugin. There are fewer ghostings by our method than by Hugin, especially for image pair (6, 7).

While above comparison examples show the robustness of our method, we further show an additional example to demonstrate the advantages of our multi-resolution pipeline: it may achieve more robust stitching result than other two methods. Image pair (8, 9) is an example of the images impossible to stitch if the camera rotation leads to visible occlusion changes. Looking at the tallest tree in the image pair (8, 9), one can see significant occlusion changes, due to large camera motion. Normally it is difficult to compute a globally consistent motion between pixels in such type of overlapping area for image stitching. Our method stitches image pair (8, 9) under a roughly initialized displacement field although it does not completely avoid ghostings. However, there are still fewer ghostings for the trees with our method than with Hugin as shown in Fig. 9c and d. This demonstrates that our method obtains a high stitching quality even when there are occlusion changes, thanks to our multi-resolution strategy.

8.2 Multiple images

Our method also applies to stitching a sequence of images captured with relaxed motion. In this case, the registration method is applied to each neighboring image pair. First, all images are decomposed into pyramids and the motion of all top layers are initialized with TV-BMA. Then layer by layer refinement process begins. In each layer, the refinement process computes the motions and camera parameters of all images of this layer and then propagates the results to next layer. For the first image pair, i.e., first two images, the estimation method is the same as proposed in Sect. 7. For the remaining image pairs, while their motion and rotation refinement processes are the same as introduced in Sects. 4 and 6, their focal lengths are estimated in a slightly different way. Each image will estimate its own focal length with Eq. 11. This time f_i is set to be the focal lengths of the previous image and f_j is set to be the focal length of current image whose initial value is the same as its previous image in the same layer (when $c = 0$) or the double value of its previous layer in the same pyramid (when $c > 0$).

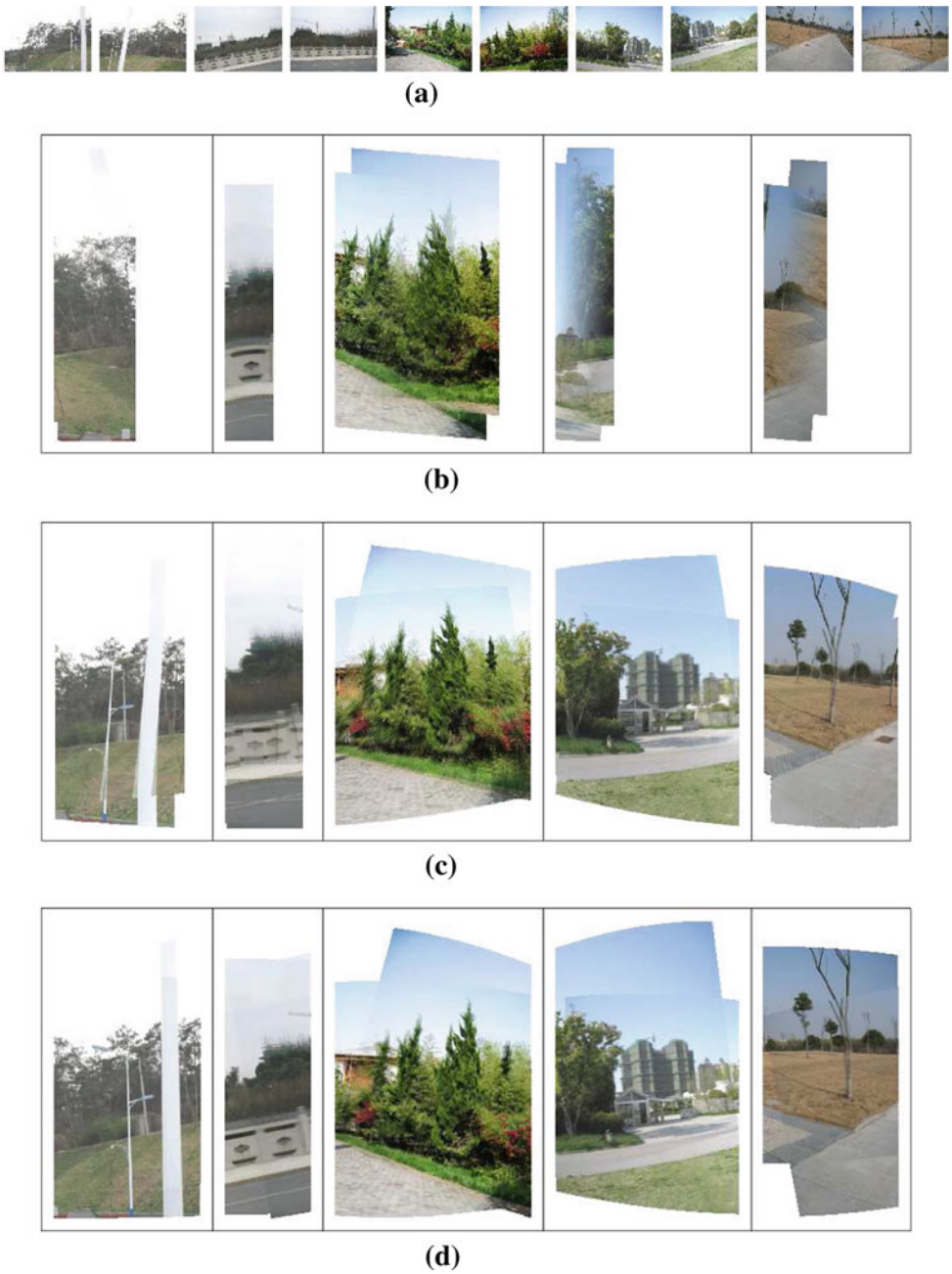

Fig. 9 More performance comparison between The Panorama Factory, Hugin and ours. The overlapping areas shown in this Figure are generated in the same way as in Fig. 8. **a** Five image pairs used in the performance comparison. These images are indexed as 0–9 from the left to the right. **b** The overlapping areas obtained by The Panorama Factory corresponding to the five image pairs shown in Fig. 9a. **c** The overlapping areas obtained by Hugin corresponding to the five image pairs shown in Fig. 9a. **d** The overlapping areas obtained by our method corresponding to the five image pairs shown in Fig. 9a

Table 4 The recovered angles and the λ computed with our method for the image pairs shown in Fig. 9. For the coarsest layer for image pair (2, 3), it is sized 47 × 36 and thus its start λ is 0.1472. For the left pairs, each has a coarsest 30 × 40 layer and thus its start λ is 0.1750. Only one iteration is needed for all pairs except the last one which cannot be registered correctly with TV-BMA, in which case, the initial displacement field is set manually *(manual)*

Image pair	(0, 1)	(2, 3)	(4, 5)	(6, 7)	(8, 9)
Roll	11.0646	10.2635	−18.1565	−13.5921	9.9290
Pitch	−2.9429	−1.4773	−2.9787	12.2426	5.9771
Yaw	13.0881	18.0531	23.2635	17.6844	25.7783
λ	0.1750	0.1472	0.1750	0.1750	*manual*

Fig. 10 A 5-image sequence of tall and long buildings. These images are indexed as 0–4 from left to right and the images marked by *red* rectangle have relaxed motions w.r.t their predecessors

Table 5 The finally detected feature matches in the bottom layer, recovered angles and λ adaptively selected in TV-BMA for neighboring image pairs in Fig. 10 by our method

Image pair	(0, 1)	(1, 2)	(2, 3)	(3, 4)
Feature matches	387	516	364	407
Roll	−6.7568	−4.7317	17.5688	−4.2039
Pitch	−0.5636	−0.6733	−0.1411	1.1539
Yaw	11.0102	8.6162	10.3370	5.9745
λ	0.1472	0	0.2207	0

Table 6 The final focal length estimates of images in Fig. 10

Image index	0	1	2	3	4
Focal length	1971.2340	1973.3381	1968.1410	1972.5369	1980.1795

First we show a 750 × 562 image sequence of tall and long buildings (Fig. 10). The images are indexed 0 through 4 from left to right for the convenience of discussion.

Table 5 gives the λ, which is adaptively selected for creating the TV-scale image in TV-BMA for the motion initialization, the final feature matches and rotation angles obtained. In this table, *feature matches* represents the final number of feature matches obtained for bottom layer image pairs. Since the coarsest layer is of size 47 × 36 and thus the starting λ in the iteration step of TV-BMA is 0.1472. A zero λ means the displacement field is recovered without the iterative estimation in TV-BMA. For the two marked with red rectangles of Fig. 10, they have large relative roll (−6.7568° and 17.5688°) than others. Therefore, as shown in Table 5, there are additional iteration steps used in TV-BMA when registering them to their respective previous images.

Table 6 shows the final focal lengths obtained. Figure 11a shows the final stitching result. The mask image in Fig. 11b shows the corresponding placement of source images. Since the focal lengths and rotation matrices are accurately recovered, these images can be finely stitched.

Figure 12 gives another example sequence where all five 1,024 × 768 images are captured with deliberate relaxed motions. Again, images are indexed 0 through 4 from left to right. The two images in the red rectangle need more iterations in TV-BMA, which will be discussed next.

Table 7 is obtained for this sequence like Table 5. The size of the top layer is 64 × 48 and the initial λ is 0.1094. All image pairs have to be estimated through the iteration step in TV-BMA since there are about 20 degree roll for all image pairs.

We find that only pair (2, 3) (images in red rectangle in Fig. 12) requires five additional iterations in TV-BMA. This is because large scale patterns have quite different illumination and orientations and thus only additional smaller scale patterns can help find the best displacement field, as explained in Fig. 13. As the green rectangles demonstrate, small λ tends to smear out patterns and textures for comparison (Fig. 13a) and thus it is difficult to find the displacement field between the two images. But when the λ becomes large enough (Fig. 13b), sufficient number of patterns are available to align the images and to obtain the initial displacement field.

The final computed focal length for each source image is shown in Table 8. The final stitching result is shown in Fig. 14a which has only one apparent ghosting road light. Considering the error accumulation through multiple images, the distortions in the camera lens as well as the occlusion changes because of large roll between image 3 and 4 (recall the image pair (8, 9) of Fig. 9), such ghosting is acceptable. The mask image shown in Fig. 14b displays the position and shape of each warped image.

8.2.1 Additional examples

We also show two long sequences with relaxed motion between neighboring images. The stitching cannot be done by simply applying our local registration method discussed so far because of the significant error accumulation among

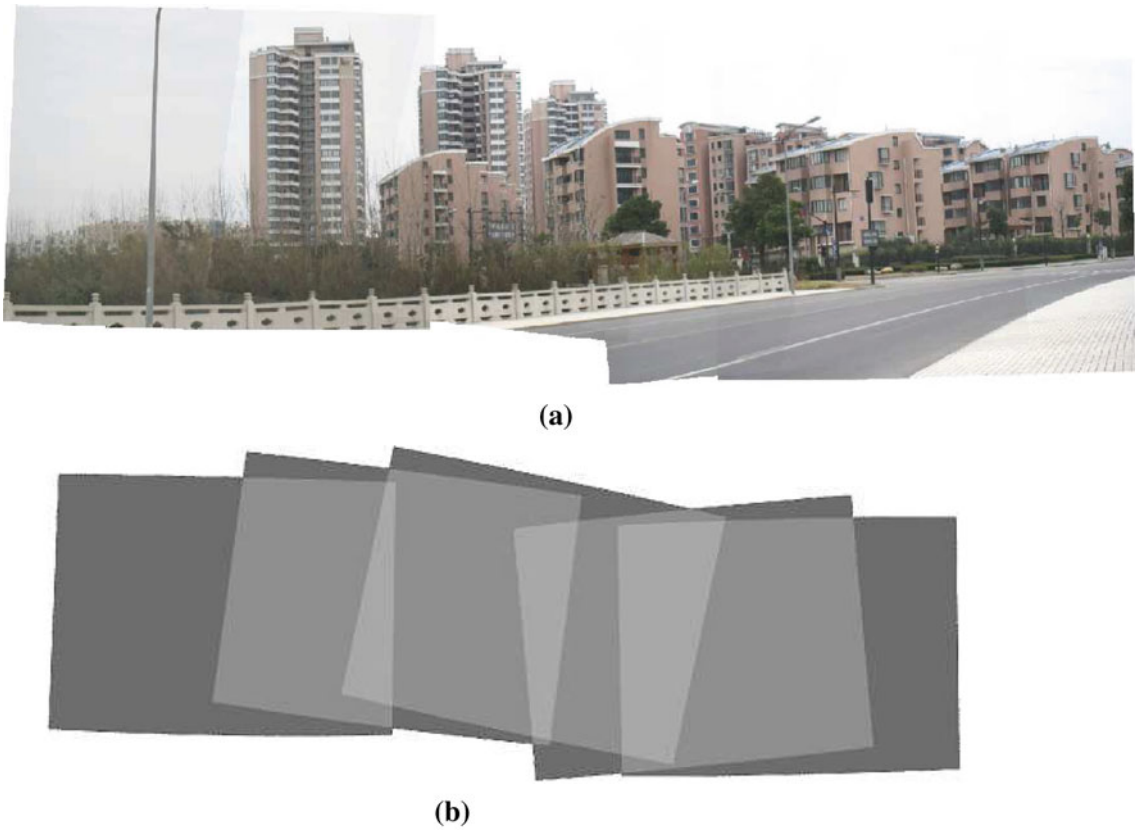

Fig. 11 The stitching result of images in Fig. 10. **a** The stitching result. **b** The mask image showing the position and shape of each stitched image in Fig. 11a

Fig. 12 A 5-image sequence captured with deliberately relaxed motions. These images are indexed as 0–4 from left to right. The images inside the red rectangle need more iterations than others because of different illumination and orientations in the patterns of their overlapping area

image pairs for a long sequence. As such, our proposed method is first taken to be the local registration method and applied in the same way as stitching the two 5-image sequences discussed before. Then our recently proposed new bundle adjustment method [37] is applied as the global registration method to stitch these images. The bundle adjustment method is used to remove the error accumulation of the motion between neighboring images through adjusting the parameters of all images together. For more details on the bundle adjustment method, please refer to [37].

Figure 15 shows the nine images registered together. There are significant pitch or roll motion between neighboring images as Fig. 15a shows, yet they are successfully stitched by our method (Fig. 15b).

Figure 16 gives another sequence of 10 noisy images. Our proposed local and global alignment method can still successfully stitch them without any ghosting. In this case, we simply omit the low contrast filter for obtaining more features. Figure 16b shows the final stitching result. This result is also difficult to obtain without the proposed registration method in this paper.

9 Discussion

In this section, we discuss how much roll and pitch are supported by previous methods and our algorithm, respectively. It is difficult to quantify the supported roll and pitch angles

Table 7 The finally detected feature matches in the top layer, recovered rotation angles and adaptively selected λ with TV-BMA for images in Fig. 12

Image pair	(0, 1)	(1, 2)	(2, 3)	(3, 4)
Feature matches	575	342	224	439
Roll	19.3783	−15.6459	19.6835	−21.0502
Pitch	0.2425	2.7956	−1.4949	−0.2480
Yaw	6.7424	9.6152	9.0181	10.2574
λ	0.1094	0.1094	0.8306	0.1094

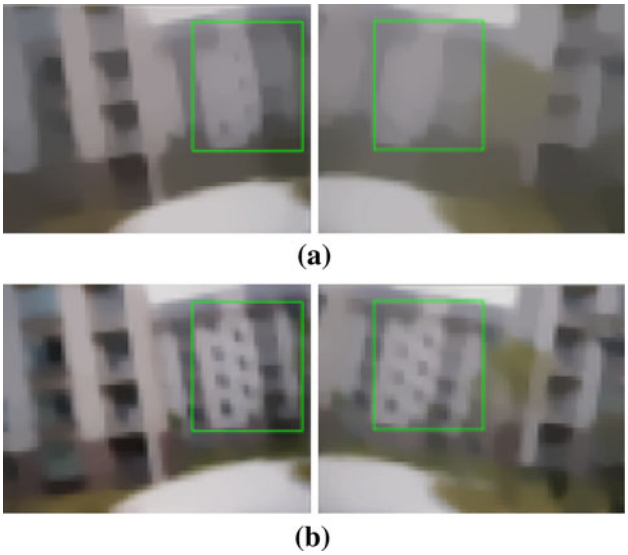

Fig. 13 The comparisons of image pair (2, 3) in Fig. 12. When λ is small (0.1094), the TV-scale images are too small for the illumination-BMA. Therefore, TV-BMA adaptively selects $\lambda = 0.8306$ to generate the TV-scale image with enough details to obtain a global optimum. **a** The TV-scale images when $\lambda = 0.1094$. **b** The TV-scale images when $\lambda = 0.8306$

in former research since there is no such reports to our best knowledge. However, we argue that: existing work cannot robustly cope with an image pair having relatively large roll and pitch, especially when the absolute roll and pitch is about 20° and 6°, respectively. Our proposed method specially targets large motions as this with TV-BMA.

However, there are at least two questions left to answer: (1) Can we handle even larger roll or pitch? and (2) can we have large roll, pitch and even raw simultaneously? For the first question, we find it is rather difficult for our method to deal with roll and pitch larger than 20° and 6°, respectively. The main reason is that there will be significant occlusion changes for the image pair with disappearance of existing surfaces and newly appearing surfaces on the right image, and thus the two images are unlikely to be stitched well. This is also the reason that the lamp in Fig. 14a has ghostings, note that the absolute roll angle between the last image pair is more than 20° (−21.0502°). The effect of pitch and roll on occlusion changes also relates to the focal length. This is due to the fact that the object appears large when the focal length turns small. Therefore, a minor occlusion change in the image with a large focal length can appear to be a large occlusion change in the image with a small focal length. For the second question, we can also conclude that it is difficult to have roll, pitch and yaw to be around 20°, 6° and 20° simultaneously because it will also have drastic occlusion changes with considerable existing surfaces disappearing and new surfaces appearing. This is one of the reasons why there is no finely stitched image pair having relatively large roll, pitch and yaw at the same time in all experiments discussed in this paper.

10 Conclusions

This paper presents a new multi-resolution method for mosaicking images captured with relatively large roll or pitch movement called relaxed motion. It integrates direct method to find the initial motion and feature-based method to calibrate the camera layer by layer. The main contribution lies in our motion estimation method. First, an adaptive BMA called TV-BMA is proposed whereby $TV - L^1$ model is applied to generate TV-scale images of the coarsest layer with appropriate details for illumination-BMA. TV-BMA greatly improves the accuracy of motion estimation of images with a relatively large roll or pitch. Second, the low contrast filter and RANSAC remove noisy low contrast pixels and ensure global geometrical consistency. Our results show we can obtain a much stable projective homography for feature detection and reliable inliers for camera calibration. On the basis of the angle-invariant property of feature vectors and the rotation variance property of feature matches, we also pro-

Table 8 The focal lengths estimated finally for images in Fig. 12

Image index	0	1	2	3	4
Focal length	2996.7052	2996.7054	2993.4695	2998.9570	2997.8543

Fig. 14 The stitching result of images in Fig. 12. **a** The stitching result. **b** The mask image showing the position and shape of each stitched image in Fig. 14a

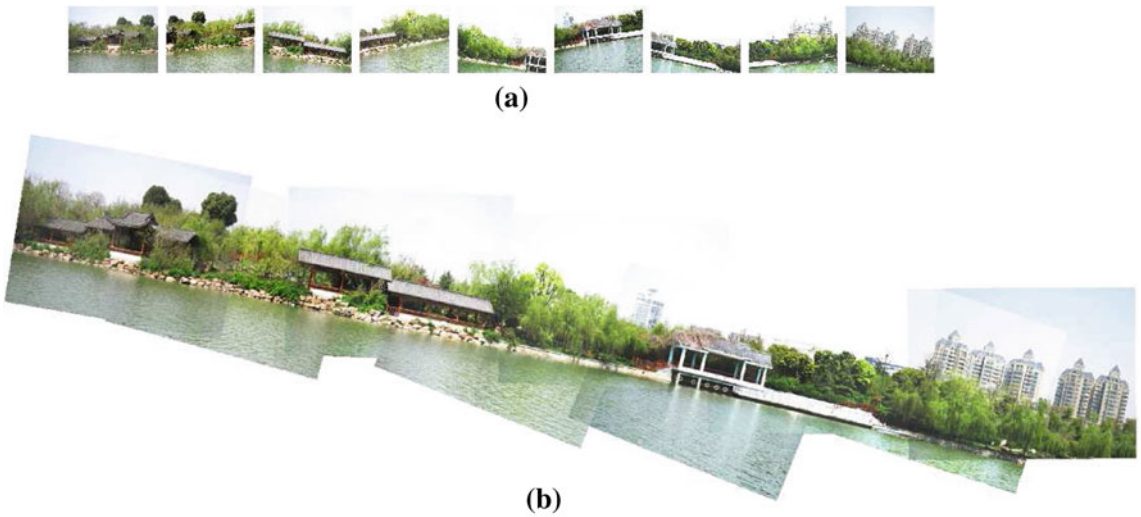

Fig. 15 A clear nine-image sequence registered by our method and bundle adjustment. **a** The source images. **b** The registered result

pose a combination of non-linear optimization methods to improve the estimation accuracy of focal length and rotation matrix, which are critical to final stitching. These methods include golden section search, simplex method and subspace trust region method. Extensive experiments demonstrate the efficiency of our method in mosaicking images with relaxed motion.

However, there are also some problems with the proposed method. One problem is that the illumination normalization method in the illumination-BMA cannot cope with

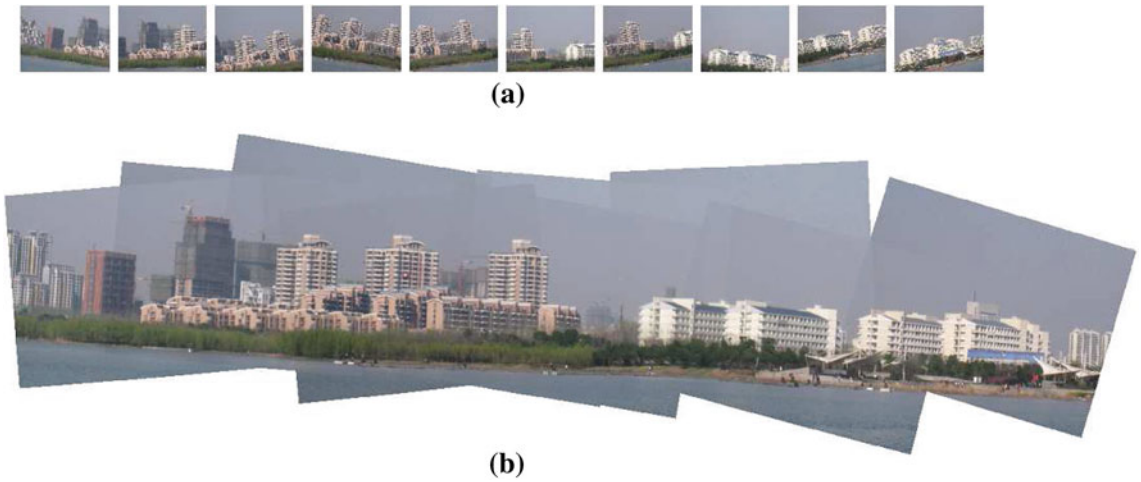

Fig. 16 A noisy ten-image sequence registered by our method and bundle adjustment. **a** The source images. **b** The registration result

large illumination variation between neighboring images. We tried gamma correction to work around the difficulty but it does not help much. Some images cannot be automatically stitched when this illumination normalization problem appears because TV-BMA does not converge. The pair (8, 9) shown in Fig. 9 is an example of this problem, where we manually set the displacement field for comparing the performance. Another problem is how to handle images with few detected features. Currently, we simply lower the threshold and the region radius of Harris corner detector to increase the number of features or skip the low contrast filter when there is very few features (Fig. 16 is such an example.). Jin [21] proposes a minimum solution for aligning two images where only three feature matches are required. It is a possible solution we will study it in the future.

In the future, we will also study image mosaicking of other types of hard-to-stitch images, such as images with very small overlap, large exposure difference and apparent lens distortions. We envision a more flexible mosaic system if these goals are reached.

Acknowledgments Some test images were provided courtesy of ArcSoft China (Hangzhou). We would like to thank anonymous reviewers for their constructive comments on this paper. Their valuable feedbacks are truly appreciated. Special thanks to Pengcheng Wu for his kind help of proof-reading. The work is under the co-support of National Nature Science Foundation of China (No. 61003131 and 61003038), the Key Science Fund for Higher Education of Anhui Province, China (KJ2010A010) and the Key Science Fund for Youth Researchers of Anhui University (2009QN009A).

References

1. Szeliski, R.: Image alignment and stitching: a tutorial. Found. Trends Comput. Graph. Comput. Vis. **2**(1), 1–104 (2006)
2. Szeliski, R.: Video mosaics for virtual environments. IEEE Comput. Graph. Appl. **16**(2), 22–30 (1996)
3. Peleg, S., Herman, J.: Panoramic mosaics by manifold projection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR '97), pp. 338. IEEE Computer Society, Washington (1997)
4. Bartoli, A., Zisserman, A.: Direct estimation of non-rigid registration. In: Proceedings of British Machine Vision Conference 2004 (BMVC 2004), pp. 899–908. British Machine Vision Association, Kingston University, London (2004)
5. Huang, Y.-W., Chen, C.-Y., Tsai, C.-H., Shen, C.-F., Chen, L.-G.: Survey on block matching motion estimation algorithms and architectures with new results. J. VLSI Signal Process. **42**, 297–320 (2006)
6. Zitova, B., Flusser, J.: Image registration methods: a survey. Image Vis. Comput. **21**, 977–1000 (2003)
7. Cho, S.-H., Chung, Y.-K., Lee, J.Y.: Automatic image mosaic system using image feature detection and taylor series. In: Proceedings of the Seventh International Conference on Digital Image Computing: Techniques and Applications (DICTA 2003), pp. 549–560. CSIRO Publishing, Macquarie University, Sydney (2003, Dec)
8. Brown, M., Lowe, D.G.: Recognising panoramas. In: Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV '03), pp. 1218–1225. IEEE Computer Society, Washington (2003)
9. Fang, X., Zhang, M., Pan, Z., Wang, P.: A new method of manifold mosaic for large displacement images. J. Comput. Sci. Technol. **21**(2), 218–223 (2006)
10. Brown, M., Lowe, D.G.: Automatic panoramic image stitching using invariant features. Int. J. Comput. Vis. **74**(1), 59–73 (2007)
11. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**, 1615–1630 (2005)
12. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
13. Szeliski, R., Shum, H.-Y.: Creating full view panoramic image mosaics and environment maps. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '97), pp. 251–258. ACM Press/Addison-Wesley Publishing Co, New York (1997)
14. Zhou, L.: Image matching using resolution pyramids with geometric constraints. Patent number US 6785427 (2004, Aug)
15. Chen, T., Huang, T.S.: Optimizing image registration by mutually exclusive scale components. In: Proceedings of IEEE 11th International Conference on Computer Vision (ICCV 2007), pp. 1–8. IEEE, Rio de Janeiro (2007, Oct)

16. Chan, T.F., Esedoglu, S.: Aspects of total variation regularized l^1 function approximation. SIAM J. Appl. Math. **75**(5), 1817–1837 (2004)
17. Harris, C., Stephens, M.: A combined corner and edge detection. In: Proceedings of The Fourth Alvey Vision Conference, pp. 147–151 (1988)
18. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**, 381–395 (1981)
19. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. 2nd edn. Cambridge University Press, Cambridge (2004)
20. Brown, M., Hartley, R.I., Nister, D.: Minimal solutions for panoramic stitching. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07), vol. 0, pp. 1–8. IEEE Computer Society, Los Alamitos (2007)
21. Jin, H.: A three-point minimal solution for panoramic stitching with lens distortion. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), vol. 0, pp. 1–8. IEEE Computer Society, Anchorage (2008)
22. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys. D **60**(1–4), 259–268 (1992)
23. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM J. Imaging Sci. **1**(3), 248–272 (2008)
24. Yang, J., Yin, W., Zhang, Y., Wang, Y.: A Fast Algorithm for Edge-Preserving Variational Multichannel Image Restoration. Tech. Rep. TR08-09. Department of Computational and Applied Mathematics, Rice University (2008, July)
25. Yang, J., Zhang, Y., Yin, W.: An efficient tvl1 algorithm for deblurring multichannel images corrupted by impulsive noise. SIAM J. Sci. Comput. **31**(4), 2842–2865 (2009)
26. Geman, D., Yang, C.: Nonlinear image recovery with half-quadratic regularization. IEEE Trans. Image Process. **5**(7), 932–946 (1995)
27. Roche, A., Malandain, G., Pennec, X., Ayache, N.: The correlation ratio as a new similarity measure for multimodal image registration. In: MICCAI '98: Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 1115–1124. Springer-Verlag, London (1998)
28. Noble, A.: Descriptions of Image Surfaces. PhD thesis. Department of Engineering Science, Oxford University (1989)
29. Kovesi, P.D.: MATLAB and Octave Functions for Computer Vision and Image Processing. School of Computer Science & Software Engineering, The University of Western Australia. Available from: http://www.csse.uwa.edu.au/~pk/research/matlabfns/
30. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, New York (2003)
31. Moré, J.J., Sorensen, D.: Computing a trust region step. SIAM J. Sci. Stat. Comput. **4**(3), 553–572 (1983)
32. Coleman, T.F., Li, Y.: An interior trust region approach for nonlinear minimization subject to bounds. SIAM J. Optim. **6**, 418–445 (1996)
33. Byrd, R., Schnabel, R., Shultz, G.: Approximate solution of the trust region problem by minimization over two-dimensional subspaces. Math. Progr. **40**, 247–263 (1988)
34. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Trans. Pattern Anal. Mach. Intell. **13**, 376–380 (1991)
35. Strait, J., Smoky City Design, L.: The panorama factory (2009, May)
36. d'Angelo, P., Behrmann, K.-U., Wilkins, D., Halley, E., Ukai, I., Postle, B., Jin, J., Mesec, Z., Jenny, A., Yaniv, Z., Januszewski, M., Patterson, G., Sharpless, T., Levy, Y.: Hugin—panorama photo stitcher (2010, Mar)
37. Fang, X., Luo, B., Zhao, H., Tang, J., Zhai, S.: A new multi-resolution image stitching with local and global alignment. IET Comput. Vision (2010, to appear)