IEEJ TRANSACTIONS ON ELECTRICAL AND ELECTRONIC ENGINEERING
IEEJ Trans 2018; **13**: 134–146
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI:10.1002/tee.22507

Paper

The Algorithm of Seamless Image Mosaic Based on A-KAZE Features Extraction and Reducing the Inclination of Image

Zhong Qu*[a], Non-member
Wei Bu*, Non-member
Ling Liu**, Non-member

The traditional feature point detection algorithm is based on the linear scale decomposition. In the SIFT (Scale Invariant Feature Transform) algorithm, features are obtained through building the image pyramid by the Gaussian filter. SIFT has good robustness but has some flaws as well. Gaussian filter neither preserve object boundaries nor smooth the same level details and noise at all scales, which impair the accuracy and distinctiveness of the feature point positioning. Nonlinear scale decomposition can solve these problems. In this paper, a new image mosaic algorithm based on A-KAZE feature is proposed to take advantages of the A-KAZE algorithm in terms of rotation invariance, illumination invariance, speed, and stability. The optimal stitching line is obtained and the multi-resolution fusion algorithm is used to fuse the image in order to achieve a satisfactory seamless image of high resolution. The whole straightening method is applied in the image mosaic to solve the problem of the tilt of multiple image mosaic. Experimental results show that stitching algorithm in this paper is faster and more robust compared to the traditional SIFT algorithm. © 2017 Institute of Electrical Engineers of Japan. Published by John Wiley & Sons, Inc.

Keywords: A-KAZE feature; nonlinear filter; image mosaic; seam line; multiresolution fusion

Received 12 June 2016; Revised 18 September 2016

1. Introduction

Image mosaic is the synthesis of multiple images with overlapping regions into a seamless high-resolution panoramic image. These images are taken at different perspectives, different times, or different degrees of exposure. Improving the real-time and stitching quality of image stitching is a challenging and hot topic in the field of computer vision [1].

Image mosaic is divided into two stages: image registration and image fusion. So far, in the field of image processing, experts at home and abroad have done a lot of research on image registration, and also have presented a lot of image registration methods. Common registration methods are: image registration based on gray level information, image registration based on transform domain, and image registration based on feature. Image fusion is the key technology of image mosaic. Direct stitching can produce obvious stitching seams and artifacts. The aim of fusion is to generate seamless images by solving the above problems. Scale Invariant Feature Transform (SIFT) is the most common traditional feature points detection algorithm used in image mosaics. In the process of constructing the image of the Pyramid, SIFT uses the linear Gauss expansion filter, which is easy to cause the obscure boundary and details loss [2,3]. Nonlinear scale decomposition can solve this kind of problem, therefore, this paper uses A-KAZE algorithm to extract image feature points [4,5].

Traditional stitching algorithm uses KD-tree and Best Bin First algorithm to complete the initial matching of two images, and use random sample consensus algorithm to estimate the affine transformation matrix. In this paper, the KNN matching of the target image to the reference image and the reference image to the target image are constructed and the common matching pair is used as the matching result to improve the accuracy of the initial matching.

The exposure difference between the images and the seam line cannot be solved by accurate image registration, so an effective image fusion method is needed to eliminate these problems. Most of the image mosaic use the method of weighted average fusion, weighted average processing for overlapping areas. However, if the image exposure difference is too large, the method is not effective and easily causes fuzziness in the image overlap region. In this paper, the method of seeking the optimal stitching line [6] and the improved Laplacian multiresolution fusion [7] are used to eliminate the gap between images. In the process of stitching, the cumulative errors will lead to the distortion of the panorama. In this paper, we straighten the whole panorama. Experimental results show that this algorithm performs better than the traditional algorithm.

Figure 1 is the total process of this paper. First, the A-KAZE feature is extracted from the reference image and the target image. Then, the initial matching pairs are obtained by the bidirectional KNN matching and the RANSAC algorithm is adopted to eliminate the false matching. After the registration of the two images, this paper finds optimal stitching line according to an energy function and uses multiresolution fusion to eliminate the stitching line. Finally, an automatic straightening of tilted and distorted panoramic images is carried out.

The rest of this paper is organized into seven sections. Section 2 describes the A-KAZE algorithm for extracting image features. Section 3 presents the details of the feature registration. Section 4 proposes the improved image fusion process, which includes finding the optimal stitching line and Laplacian multiresolution fusion. In Section 5, the straightening method is used to straighten the panoramic image. Section 6 shows experimental results for robustness evaluation. Section 7 is the conclusion of the paper.

[a] Correspondence to: Zhong Qu. E-mail: quzhong@cqupt.edu.cn

*College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

**College of Mobile Telecommunications, Chongqing University of Posts and Telecommunications, Chongqing, 401520, China

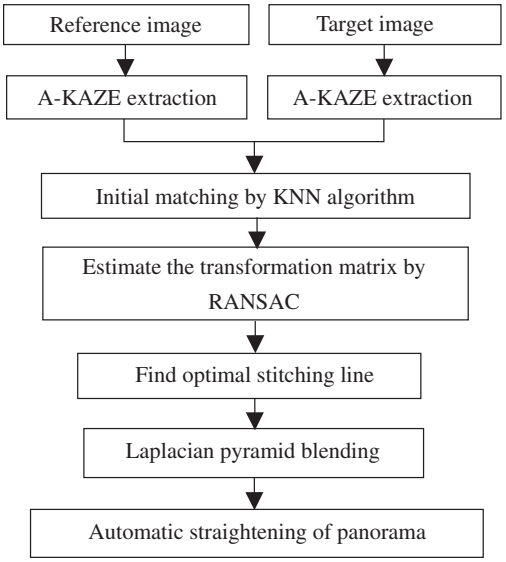


Fig. 1. The total process of this paper

2. Feature Point Extraction

The image mosaic algorithm based on the SIFT algorithm is a classic method, but the SIFT algorithm also has its own shortcomings. First, on the accuracy of the feature points detection, the SIFT algorithm is based on the linear filtering. At the expense of local precision, the SIFT algorithm for the feature point detection will lose image boundaries when building the image of the Gauss Pyramid, so it will reduce the accuracy of the feature points detection. Unlike its predecessor, the A-KAZE algorithm is based on nonlinear filtering. Nonlinear diffusion filtering constructs a stable nonlinear scale space with arbitrary step size and solves the problem of fuzzy boundaries and detail loss which is caused by the Gauss filter. Secondly, the time cost to extract feature points using SIFT is very high. So, it is very useful to find an accurate and efficient algorithm for feature point extraction. The A-KAZE feature point extraction algorithm meets the goal of the accuracy and high efficiency.

Similar to the SIFT feature extraction algorithm, the A-KAZE algorithm is divided into three steps to extract the feature points. First step is the establishment of nonlinear scale space, then comes the detection and localization of the spatial extreme points, the next step is the determination of the feature points direction, and final step is the generation of feature descriptor [4].

2.1. Building a nonlinear scale space Alcantarilla, Nuevo, Bartoli [4] proposed a novel and fast multiscale feature detection and description algorithm called A-KAZE in 2013. A-KAZE algorithm uses nonlinear diffusion filter, which is usually expressed as the divergence of a certain kind of flow function, to describe the brightness changes in different scale space. The nonlinear diffusion equation is:

$$\frac{\partial L}{\partial t} = div(c(x,y,t)\cdot\nabla L) \tag{1}$$

where L is an image matrix and time, t, is the scale parameter. div and ∇ represent the divergence and gradient operators, respectively. The conductivity function $c(x,y,t)$ is defined as:

$$c(x,y,t) = g(|\nabla L_\sigma(x,y,t)|) \tag{2}$$

∇L_σ represents the gradient of the image after Gauss smoothing.g is defined as:

$$g = \frac{1}{1+\frac{|\nabla L_\sigma|^2}{\lambda^2}} \tag{3}$$

The parameter λ is the contrast factor that controls the horizontal diffusion, and it decides the retaining of the number of edge information. The greater the value of λ is, the less the edge information is preserved.

The construction method of A-KAZE feature scale space is similar to the SIFT algorithm, and the size of the scale is increasing according to logarithm [8]. The fast explicit diffusion (FED) algorithm is used to construct the image of Pyramid, and Pyramid is divided into O groups, each group contains S sublayer. All layers in the A-KAZE group are used with the same resolution as the original image. The O octave and S sublevel indexes are mapped to their corresponding scale as:

$$\sigma_i(o,s) = 2^{o+s/S} \tag{4}$$

In (4), $o\in[0\ldots O-1]$, $s\in[0\ldots S-1]$, $i\in[0\ldots M]$, where M is the total number of filtered images. The nonlinear diffusion filtering operates in time units [4], so it is needed to convert the scale parameters σ_i in pixels to time units as:(5),

$$t_i = \frac{1}{2}{\sigma_i}^2 \tag{5}$$

In formula (5), $i=\{0\ldots \mathrm{M}\}$. t_i is called evolution time. A-KAZE algorithm constructs nonlinear scale space through different evolution time.

Because there is no analytic solution to the nonlinear partial differential equation, A-KAZE algorithm uses FED algorithm [9] to get approximate solution and construct the image of Pyramid. The discretization of formula (1) using an explicit scheme can be expressed in formula (6),

$$\frac{L^{i+1}-L^i}{\tau} = A(L^i)L^i \tag{6}$$

$A(L^i)$ is a conductance matrix of image L^i and τ is a constant time step size for explicit diffusion. The solution L^{i+1} is computed from the solution at the previous evolution level L^i through formula (6), as shown in the formula (7),

$$L^{i+1} = (I+\tau A(L^i))L^i \tag{7}$$

where I is the identity matrix. Considering the *a priori* estimation $L^{i+1,0}=L^i$, A FED cycles can obtain the formula (8).

$$L^{i+1,j+1} = (I+\tau_j A(L^i))L^{i+1,j} \tag{8}$$

In formula (8), $j=0,\ \ldots\ ,n-1$. n is the number of the explicit diffusion steps. τ_j represents the corresponding variable step size, as shown in the formula (9),

$$\tau_j = \frac{\tau_{\max}}{2\cos^2\left(\frac{2j+1}{4n+2}\pi\right)} \tag{9}$$

In formula (9), $\tau_{\max}$ is the maximal step size that does not violate the stability condition of the explicit scheme [4].

Figure 2 shows the comparison between the Gaussian and nonlinear diffusion scale space for several evolution times. Top row shows Gaussian (linear diffusion) scale space, while bottom row shows nonlinear diffusion scale space. From the contrast in Fig. 2, Gauss filter leads to the obscure boundary and detail loss. On the contrary, nonlinear filtering overcomes the disadvantages of linear filtering.

2.2. Feature detection and location After nonlinear diffusion filtering, the Hessian matrixes for each of the filtered images L^i are computed in the different nonlinear scale space. The Eigenvalues of a matrix is used to classify the maxima of the

(a) (b) (c) (d)

Fig. 2. Comparison between the Gaussian and nonlinear diffusion scale space for several evolution times t_i. (a) $t_i = 5.12$ (b) $t_i = 7.24$ (c) $t_i = 14.48$ (d) $t_i = 20.48$

detector response in spatial location. The calculation of Hessian matrix is shown in the formula (10),

$$H(L^i) = \sigma_{i,norm}^2 (L_{xx}^i L_{yy}^i - L_{xy}^i L_{xy}^i) \tag{10}$$

In formula (10), $\sigma_{i,norm}^2 = \sigma_i / 2^{O^i} \cdot \sigma_{i,norm}^2$ is the normalized scale factor of the octave of each image in the nonlinear scale. Similar to the SIFT algorithm, extreme points are detected in A-KAZE by comparing a pixel to its 26 neighborhood in 3×3 regions at the current and adjacent scales.

Using the determined extreme point as the center, we find the main direction for the search radius of $6\sigma_i$ with a sampling step of size σ_i, to ensure that the feature points have rotation invariance. First-order differential value of all image pixels are weighted with a Gaussian centered at the interest point. Then the weighted value is used as the response value of image pixels. In a sliding window of a sector area of $\pi/3$ at a central angle, all of the response value summation is calculated. Traversing the entire circular area with the sector area, the direction of the Max-Sum is the main direction of the feature points.

2.3. Feature description After obtaining the position, scale and orientation information of the feature points, we need to establish a feature descriptor. A-KAZE uses a new kind of binary descriptor modified-local difference binary (M-LDB) to describe the feature points. M-LDB is a modification of local difference binary (LDB) descriptors. We subsample the grids in steps that are a function of the scale σ of the feature and rotate the grid of LDB based on the main orientation of the keypoint to enhance the robustness and rotation invariant [4].

We select a patch around the feature point, divide each image patch into $n \times n$ equal-sized grids and extract representative information from each grid cell [10]. Then, binary test operations on a pair of grid cells (i and j) is performed. Binary test operation ϖ is shown in the formula (11),

$$\varpi(\text{Func}(i),\ \text{Func}(j)) = \begin{cases} 1, \text{ if } (\text{Func}(i) - \text{Func}(j)) > 0,\ i \neq j \\ 0, \text{ otherwise} \end{cases} \tag{11}$$

where Func(·) represents the function for extracting information from a grid cell. Func(·) is defined as:

$$\text{Func}(\cdot) = \{\text{Func}_{\text{intensity}}(\cdot),\ \text{Func}_{dx}(\cdot),\ \text{Func}_{dy}(\cdot)\} \tag{12}$$

$$\text{Func}_{\text{intensity}}(i) = \frac{1}{m} \sum_{k=1\sim m} \text{Intensity}(k) \tag{13}$$

$$\text{Func}_{dx}(i) = \text{Gradient}_x(i) \tag{14}$$

$$\text{Func}_{dy}(i) = \text{Gradient}_y(i) \tag{15}$$

Figure 3 shows the performing binary tests on the pair of diagonal grids for three image patches.

Figure 3(a) represents three image patches with different pixel intensity values and distributions. In Fig. 3(b), ① is the average intensity value I, ② and ③ represent the gradient of the x direction and y direction, respectively. Red represents the upper left grid and yellow represents the lower right grid. Figure 3(c) shows the binary tests results of the three image patches calculated according to the formula (11).

The above operation may produce a long bit string, in which bits may have strong correlation and yield redundancy. A-KAZE employs a bit selection strategy, which chooses a subset of least relevant bits to form the final descriptor to improve matching and storage efficiency [10,11].

The original image is shown in Fig. 4. Image sequences in Fig. 4 are taken in a natural environment.

In Fig. 5, A-KAZE feature points extracted as shown in four group of original images under different shooting environment. The colorful points that are marked out in these images are the extracted feature points.

3. A-KAZE Feature Registration

In this paper, A-KAZE feature points of the reference image and the target image are used as node to construct two KD-trees

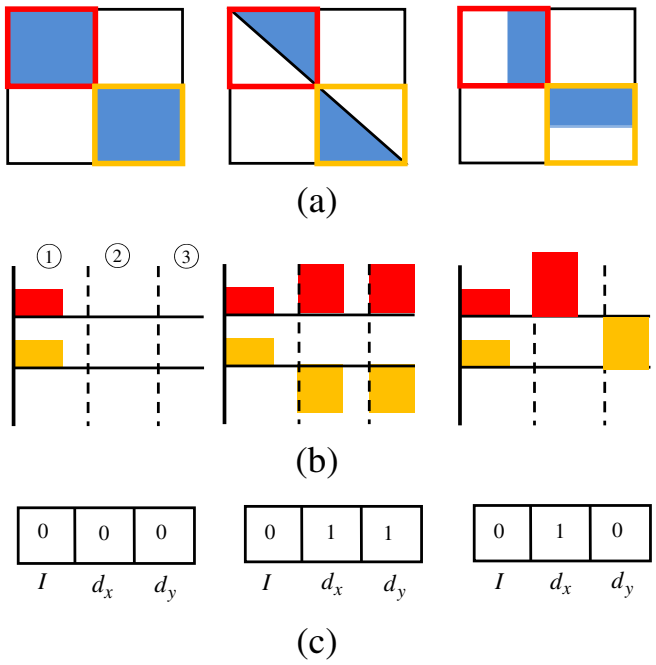


Fig. 3. The process in the establishment of feature descriptor

(KD-tree1 and KD-tree2), respectively. $K(k=2)$ nearest neighbor feature points (P_1,P_2) for each feature point of the target image are found from KD-tree1 by k-nearest neighbor algorithm. In same way, the k-nearest neighbor feature points $(P_1^{'},P_2^{'})$ for each feature point of the reference image can be found in KD-tree2. The Euclidean distance from the feature point of the target image to P_1 is d_1 and to P_2 is d_2. Similarly, the Euclidean distances from the feature point of the reference image to $P_1^{'}$ and $P_2^{'}$ are $d_1^{'}$ and $d_2^{'}$, respectively.

If $d_1/d_2 \le h$, the match is in line with the condition, and is put into the set A. Identically, if $d_1^{'}/d_2^{'} \le h$, this match is added to set B. After many experiments, the empirical value $h(h=0.6)$ is obtained.

We take the intersection of the sets A and B, which is the common match from A and B. Then, the results of common match is put into the set C. The total matching pairs of C are initial matchings.

In Fig. 6, KNN matching results from reference image to target image are shown in Fig. 6(a) and KNN matching results from target image to reference image are shown in Fig. 6(b).

Figure 7 shows the common matching results of the feature points. The common match is put after bidirectional KNN matching as the initial matching.

After getting the initial matching result, the RANSAC algorithm is adopted to remove the outer points and estimate the affine transformation matrix between images. The steps of the RANSAC algorithm are as follows:

1. $r(r=3)$ pairs of sets are selected randomly from the N matched pairs in the rough matching to estimate the parameters of the affine transformation matrix. The affine matrix structure is shown as formula (16),

$$H = \begin{bmatrix} \cos\theta & -\sin\theta & u \\ \sin\theta & \cos\theta & v \\ 0 & 0 & 1 \end{bmatrix} \tag{16}$$

where the variable θ represents the image rotation angle. u, v separately represent the translation distance in x axial direction and y axial direction.

2. The remainder matching pairs are used to test the estimated matrix. The matching feature point for the estimation matrix model is the interior point, and we put the matching feature point into the interior point set.
3. Steps 1 and 2 are repeated for n times and the numbers of internal points per time are calculated. When all matching points are performed, iteration is stopped. Select $r=3$ pairs of matching points at random (Ensure that the three pairs of feature points in the input image are composited into an invertible matrix). Given the probability that a feature match is correct between a pair of matching images (the

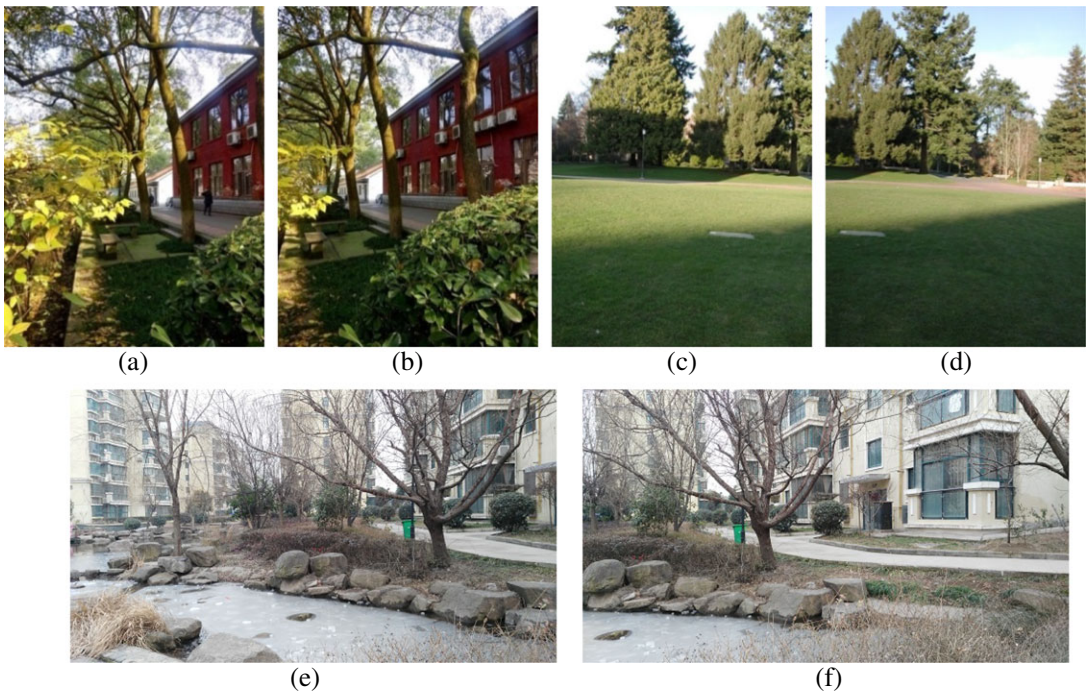


Fig. 4. Three groups of original image: (a) image 1, (b) image 2, (c) image 3, (d) image 4, (e) image 5, and (f) image 6

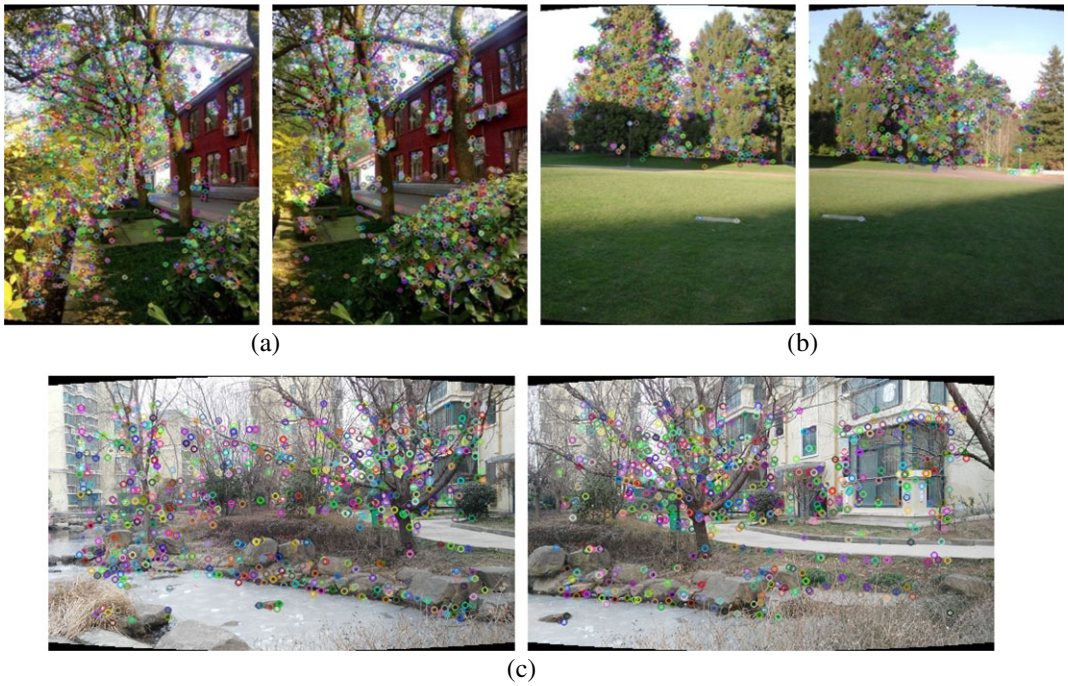


Fig. 5. The result of feature point extraction by A-KAZE algorithm: (a) A-KAZE features in Fig. 4(a) and (b), (b) A-KAZE features in Fig. 4(c) and (d), (c) A-KAZE features in Fig. 4(e) and (f)

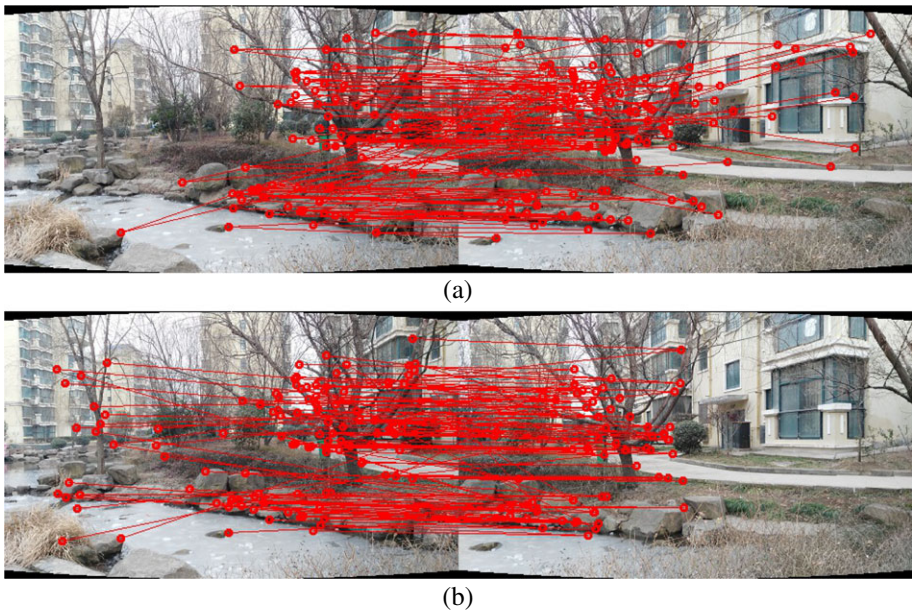


Fig. 6. The result of KNN matching: (a) KNN matching from reference image to target image (b) KNN matching from target image to reference image

Fig. 7. The result of common matching

inlier probability) is p_i [1]. The probability of finding the correct transformation after n trials is P.

$$P = 1 - (1 - (p_i)^r)^n \tag{17}$$

When the iteration number increases, the probability P will increase as well. For example, when $n = 300$ and $p_i = 0.5$, the probability that a correct affine transformation matrix is not found is approach to 1.0×10^{-18}.

4. We choose the matrix H as the affine transformation matrix, which corresponds to the maximum number of inliers. The formula for calculating the affine transformation matrix is shown in formula (18),

$$\begin{bmatrix} x_i^{'} \\ y_i^{'} \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & u \\ \sin\theta & \cos\theta & v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{18}$$

Because the transform matrix has six degrees of freedom, three pairs of matching feature points are randomly selected to estimate the transformation matrix.

Figure 8 shows the matching results after RANSAC excluding false matches.

In Fig. 9, (a) and (e) are the results of the initial matching based on SIFT features, (b) and (f) are the results of initial matching based on A-KAZE features. Figure 9(c) and (g) are the result of accurate matching based on SIFT features after eliminating the false matching by the RANSAC algorithm. Figure 9(d) and (h) are the result of accurate matching based on A-KAZE features after eliminating the false matching by the RANSAC algorithm. We define the match rate as shown in the formula (19),

$$\text{Match rate} = \frac{N_1}{N_2} \tag{19}$$

where N_1 is the number of matching pairs after eliminating the error matching and N_2 is the number of matching pairs in the initial matching.

Suppose that there are two images A and B with overlapping area, the coordinates of the target image are mapped to reference image coordinate system by means of the transformation matrix. The coordinates of the reference image A in the overlapping area are (X_i, Y_i). The corresponding coordinates $(X_i^{'}, Y_i^{'})$ of the target image B in the overlap region are transformed by the matrix H transform. The Euclidean distance is widely used in image processing to determine the similarity between images. In order to verify the accuracy of the transformed matrix, the mean value of the square sum of the corresponding point distance is used as the error index function to reflect the error of the matrix. Error index function is shown in formula (20).

$$E(H) = \frac{1}{2N} \sum_{i=1}^{N} [(Y_i^{'} - Y_i)^2 + (X_i^{'} - X_i)^2] \tag{20}$$

From Table III, we can see that the matrix obtained by algorithm in this paper has lower error value. So, it can be concluded that transform matrix H obtained by this algorithm is more accurate than the traditional algorithm.

It can be seen from the Tables I and II that the time cost of A-KAZE extraction is significantly less than the traditional SIFT algorithm. The number of A-KAZE features is slightly reduced compared to the number of the feature points extracted by SIFT, but we obtain a similar number of accurate matching pairs after RANSAC in both feature point extraction algorithm. We can clearly see that correct matching rate (match rate) obtained in this paper is significantly higher than the traditional SIFT feature matching method.

4. Image Fusion

4.1. Feature detection and location After image registration, direct synthesis will cause discontinuity of color transition and produce the image artifacts when there are moving objects [12]. So, it is needed to find an optimal stitching line to eliminate the artifacts and hide the image edges. The position of the stitching line is related to two factors (the color difference and the texture structure difference of the overlapping area). Human eyes are very sensitive to the color difference, so we need to focus on the color difference. Under the premise of little color difference, the reference factor of texture difference is to prevent the mosaic line through the image of the large difference in the texture of the object. A simple and efficient method is presented to find the suitable stitching line on an energy formula. A perfect stitching line requires the least amount of color difference and the most similar geometric structure on both sides of the stitching line [13]. In this paper, the idea of dynamic programming is used to obtain the optimal stitching line with minimum energy. The energy formula is defined as:

$$T(i,j) = \alpha \cdot E_C(i,j) + \beta \cdot E_G(i,j) \tag{21}$$

where E_C represents the color difference in 5×5 rectangular scope around stitching line pixels, E_G represents the change of texture, α and β are weight values, and $\alpha + \beta$ is equal to 1. The role of weight values is to adjust the proportion between the color difference and the texture difference. When α gets bigger, β will get smaller. In the process of searching for a mosaic line, it will lead to overly large proportion of color factors and will ignore the influence of image texture difference on both sides of the stitching seam. In contrast, when β gets bigger, α will get smaller, which will lead to overly small proportion of color factors. The irrationality of the weight values will produce a strong color difference on both sides of the stitching line. Based on many tests, α and β are equal to 0.83 and 0.17, respectively.

$$E_C = \left\| \frac{1}{5 \times 5} \sum_{m=-2}^{2} \sum_{n=-2}^{2} \text{img\_sub}(i+m, j+m) - \text{img\_sub}(i,j) \right\| \tag{22}$$

$$E_G = \text{sqrt}\left(\frac{1}{5 \times 5} \sum_{m=-2}^{2} \sum_{n=-2}^{2} \text{Gradient\_x}(i+m, j+n) \right)^2 + \left(\frac{1}{5 \times 5} \sum_{m=-2}^{2} \sum_{n=-2}^{2} \text{Gradient\_y}(i+m, j+n) \right)^2 \tag{23}$$

The accuracy and efficiency of the Scharr operator is stronger than the Sobel operator. In this paper, the Scharr operator is used to calculate the gradient.

$$Scharr\_X = \begin{bmatrix} -3 & 0 & 3 \\ -10 & 0 & 10 \\ -3 & 0 & 3 \end{bmatrix} \tag{24}$$

$$Scharr\_Y = \begin{bmatrix} -3 & -10 & -3 \\ 0 & 0 & 0 \\ 3 & 10 & 3 \end{bmatrix} \tag{25}$$

According to the energy formula, we use the idea of dynamic programming to find the optimal stitching line. The following steps are shown as follows:

1. Starting at the intersection P of overlapping regions, take the pixel value of the intersection as the first pixel value of the splicing line and derive the energy value of this point.

Fig. 8. The result after eliminate the error matching by RANSAC

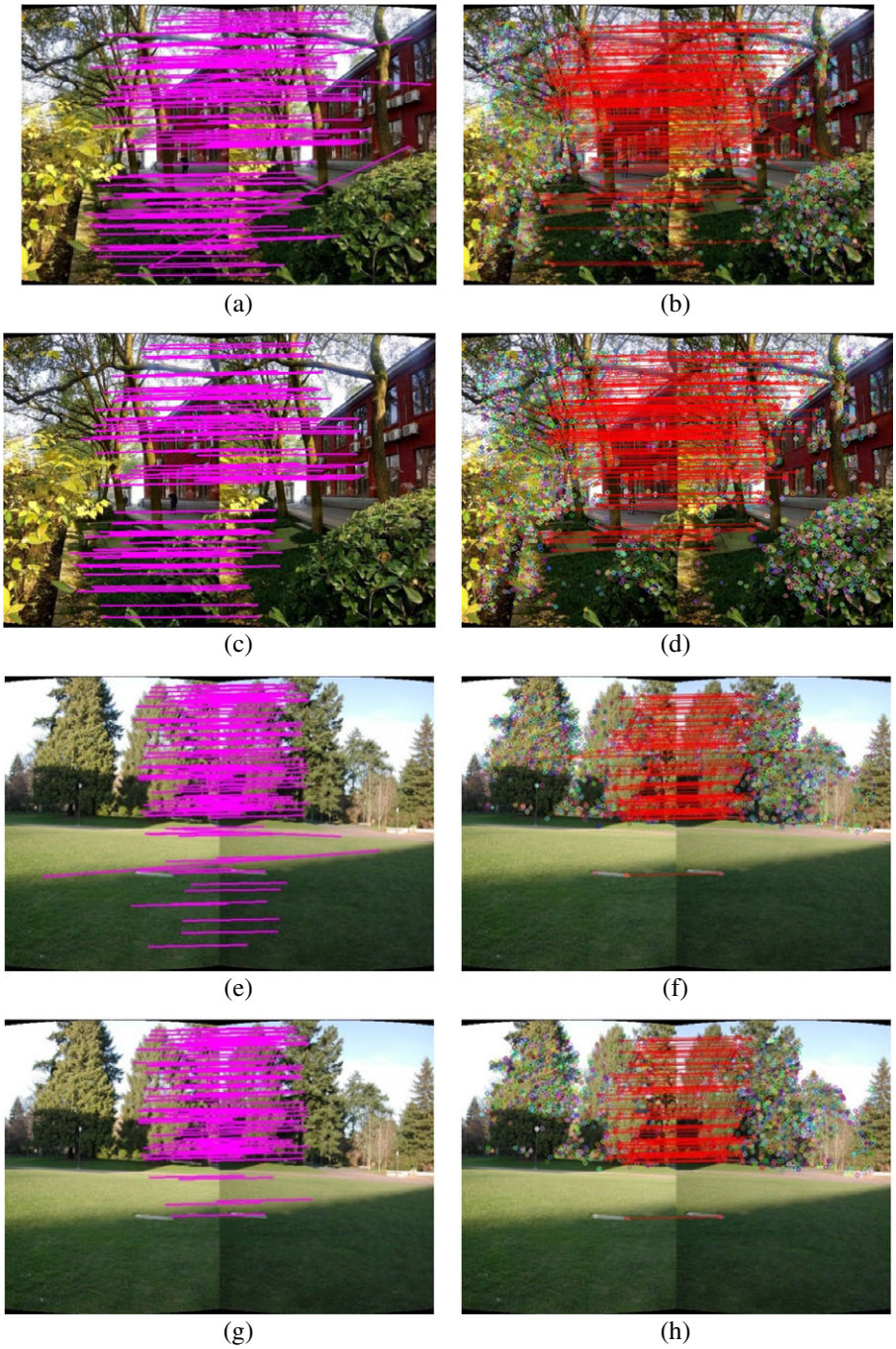


Fig. 9. The comparison of SIFT and A-KAZE feature points matching: (a) initial matching based on SIFT features, (b) initial matching based on A-KAZE features, (c) accurate matching based on SIFT features, (d) accurate matching based on A-KAZE features, (e) initial matching based on SIFT features, (f) initial matching based on A-KAZE features, (g) accurate matching based on SIFT features, and (h) accurate matching based on A-KAZE features

Table I. The comparison of experimental data for extracting feature points

Figure	Number of feature points		The time cost of feature extracting (ms)	
	SIFT	A-KAZE	SIFT	A-KAZE
Fig. 4(a) 375 × 500	1471	1845	1842	1220
Fig. 4(b) 375 × 500	1301	1555	1830	1170
Fig. 4(c) 384 × 512	1379	971	1642	1013
Fig. 4(d) 384 × 512	1255	1008	1466	1071
Fig. 4(e) 512 × 288	1001	895	1299	994
Fig. 4(f) 512 × 288	905	807	1226	969

Table II. The comparison of experimental data for SIFT and A-KAZE feature points matching

Figure	Number of match points before RANSAC in this paper	Number of match points before RANSAC in traditional SIFT feature matching	Number of match points after RANSAC in this paper	Number of match points after RANSAC in traditional SIFT feature matching	Match rate in this paper	Match rate in traditional SIFT feature matching
Fig. 4(a) and (b)	176	125	145	87	0.82	0.69
Fig. 4(c) and (d)	102	118	93	104	0.91	0.88
Fig. 4(e) and (f)	50	84	36	39	0.72	0.46

2. Then, we take three pixel points adjacent to P in the next line. The directions are toward to the bottom, bottom left 45° and bottom right 45°. We calculate the energy value of the pixel points in three directions and compare with the three energy values. The column of the minimum energy point is used as the expansion direction of the stitching line. The minimum energy point $P^{'}$'s current pixel point is on the stitching line. We set $P = P^{'}$. If the pixel point is not the last line in the picture, return step 2.
3. When extended to the last line, the obtained stitching line is the optimal with the minimum energy value.

The energy formula can predict whether the current position of the stitching line is close to the area with the large color and the structure difference. It provides a space of transformation for the next step of the stitching line search to avoid the large difference between the two sides of the stitching line. Otherwise it will have great difficulty in the process of eliminating the stitching line by fusion. The result of optimal stitching line is shown in Fig. 10.

4.2. Elimination of the stitching line Ideally, seamless image stitching can be realized by finding the optimal seam line. In fact, a different shooting angle will lead to different image exposure. So, in the actual operation, the image mosaic traces still exist.

Traditional multiresolution fusion algorithm is used to fuse the extended image directly and is not suitable for the image stitching with high real-time requirement. The aim of image fusion is to eliminate exposure difference and artifact. Fusing around the stitching line can meet the requirement of image fusion. Based on lots of experiments, we propose improved Laplacian multiresolution fusion which limits the fusion range. Obviously, fused image in a limited range shortens the fusion time and guarantees the fusion efficiency.

In order to let the stitching line have a natural transition, the method of improved Laplacian multiresolution fusion is used to eliminate the stitching line by creating the mask image I_R of the stitching line. The area on the left side of the stitching line is filled with a pixel value of 0, and the right side is filled with a pixel value of 255. The minimum bounding rectangle R of the optimal splicing is the area framed by dashed frame, and the left and right boundary of R are derived as $x_{\min}$ and $x_{\max}$. According to the experimental results, an empirical threshold value $\xi(\xi = 30)$ is obtained. We take the rectangular area $R^{'}$ to make the left boundary as $x_{\min} - \xi$,

(a) (b)

Fig. 10. The comparison between stitching line and no stitching line: (a) the result of optimal stitching line and (b) results of direct stitching

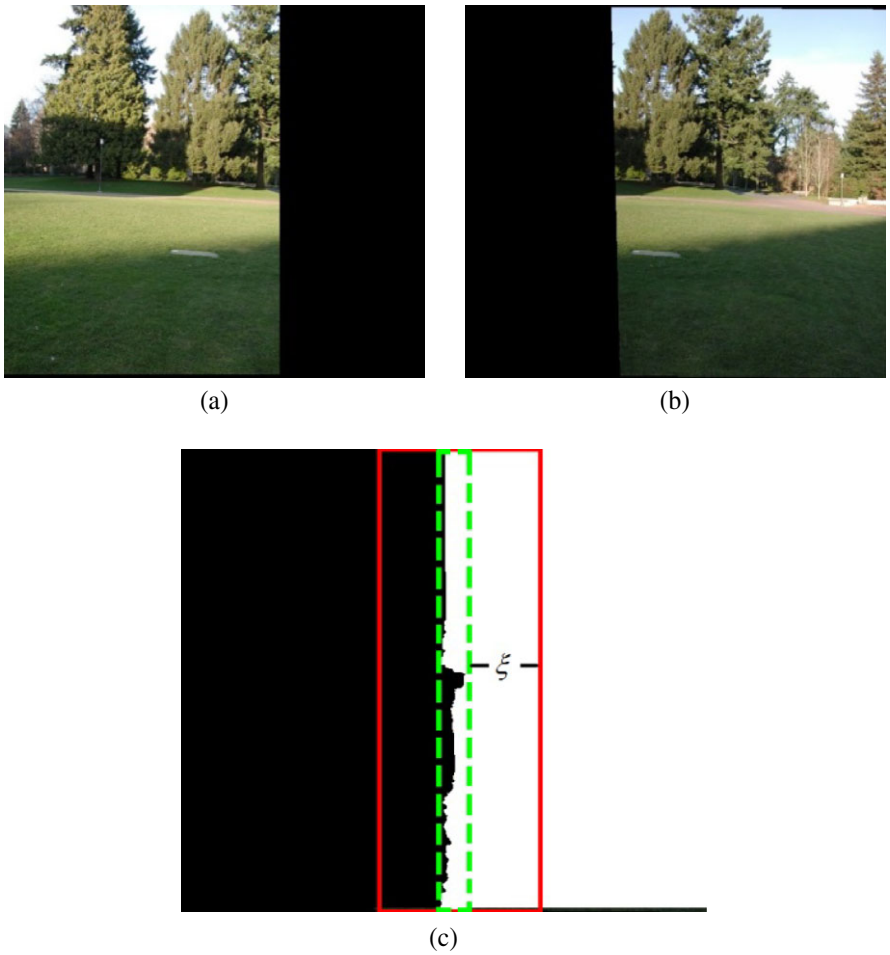


Fig. 11. Image before limited by rectangular $R^{`}$: (a) I_1 after filling (b) I_2 after filling (c) mask image I_R

the right boundary as $x_{\max}+\xi$. Rectangular area $R^{`}$ is framed by solid line frame (Fig. 11). The steps of improved Laplacian fusion algorithm are as follows:

1. The target image I_1 and reference image I_2 after registrations are expanded to the same size as the mask image, and the extended partial pixel values are assigned to 0.
2. Through the former step, we obtain three images of the same size as I_1, I_2, and I_R. In these three images, images with limited range of rectangular $R^{`}$ are obtained and represented as $I_1^{'}$, $I_2^{'}$, and $I_R^{'}$ respectively.
3. Laplacian Pyramidal decomposition method is used to analyze $I_1^{'}$ and $I_1^{'}$. L_1 and L_2 are recorded as two images of Laplacian Pyramid. First, the Gauss pyramid of $I_1^{'}$ and $I_2^{'}$ is built. The construction formula of Gauss Pyramid is shown in formula (26).

$$G_l(i,j)=\sum_{m=-2}^{2}\sum_{n=-2}^{2}\varpi(m,n)G_{l-1}(2i+m,2j+n) \tag{26}$$

where $1\le l\le N, 0\le i<R_l, 0\le j<C_l$. $\varpi(m,n)$ is a two-dimensional separable 5×5 window function. $\varpi(m,n)=h(m)^{*}h(n)$, $h(\cdot)$ is Gauss density distribution function. The expression of $\varpi(m,n)$ is as formula (27),

$$\varpi(m,n)=\frac{1}{256}\begin{bmatrix}1&4&6&4&1\\4&16&24&16&4\\6&24&36&24&6\\4&16&24&16&4\\1&4&6&4&1\end{bmatrix} \tag{27}$$

Then, we build Laplacian Pyramid,

$$\begin{cases}LP_l=G_l-G_{l+1}^{*}, 0\le l<N\\ LP_N=G_N, l=N\end{cases} \tag{28}$$

G_{i+1}^{*} is the same size as G_l and is obtained through the upsampling from G_{l+1} (Fig. 12).

4. The mask image is processed by Gauss expansion, which makes the area of the stitching line more smooth. Then, we create the Gauss Pyramid of I_R and it is recorded as G_R.
5. According to the specific fusion criteria, the two images $I_1^{'}$ and $I_2^{'}$ in each layer of Laplacian Pyramid are fused. The formula is as follows:

$$LS_l(i,j)=\frac{G_{R_l}(i,j)^{*}}{255}L_{1_l}(i,j)+\frac{(255-G_{R_l}(i,j))^{*}}{255}L_{2_l}(i,j) \tag{29}$$

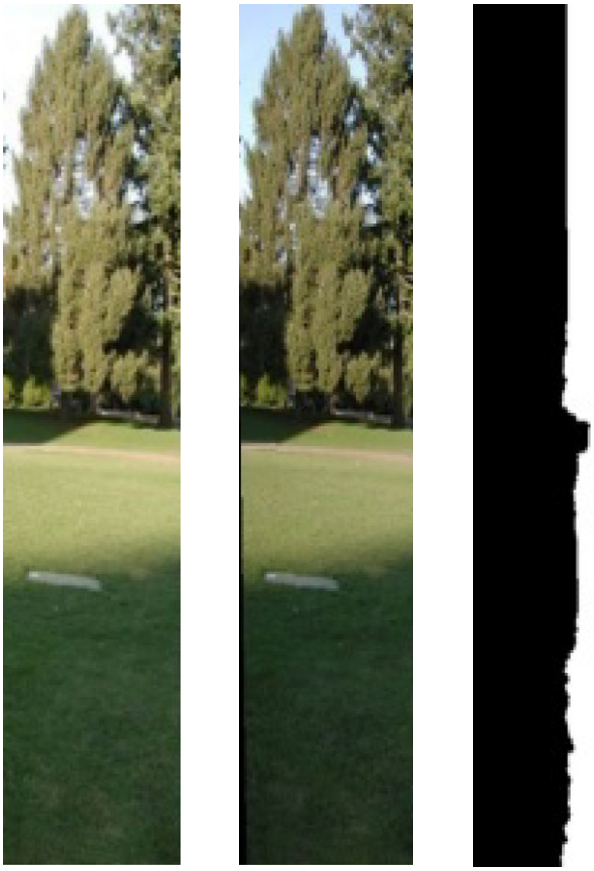
6. The Fusion Image of each Layer of Laplacian Pyramid is Reconstruct According to Formula (30)

$$\begin{cases}G_l=LP_l+G_{l+1}^{*}, 0\le l<N\\ G_N=LP_N, l=N\end{cases} \tag{30}$$

The experimental results after Laplacian multiresolution fusion are shown in Fig. 13.

5. Automatic Image Straightening

The pictures we received were taken from left to right. When the number of images to be stitched is large, we take the first sequence



(a) (b) (c)

Fig. 12. Image limited by rectangular $R^{'}$: (a) $I_1^{'}$ (b) $I_2^{'}$, (c) $I_R^{'}$

Fig. 13. The result after Laplacian fusion

image as the reference and stitch sequence images according to the world coordinate system. Due to the rotation of the camera when shooting, there will be a certain rotation relative to the world coordinates, and thus will produce the tilt distortion of the stitching results. At the same time, the cumulative error will be produced in the process of image mosaicking and leads to the tilt of the panoramic too. Therefore, in this paper, the whole image is straightened. The process of straightening the image is:

1. Before straightening, four vertex coordinates of the original image is recorded as in Fig. 14(a): Left upper $(a.x, a.y)$, Left bottom $(b.x, b.y)$, Right upper $(c.x, c.y)$, and Right bottom$(d.x, d.y)$. Then, as shown in Fig. 14(b), we set the coordinates of the four vertices after straightening as: Left upper $(a^{'}.x, a^{'}.y)$, Left bottom $(b^{'}.x, b^{'}.y)$, Right upper $(c^{'}.x, c^{'}.y)$, and Right bottom $(d^{'}.x, d^{'}.y)$.
   Due to the process of stitching from left to right in turn, left and bottom left vertex coordinates will not change after straightening. That is to say, $(a^{'}.x, a^{'}.y) = (a.x, a.y)$,$(b^{'}.x, b^{'}.y) = (b.x, b.y)$.
2. We estimate the length X of the image and the corresponding right and the bottom right vertex coordinates after straightening. As shown in Fig. 15, the image length is calculated by the formula (31),

$$X = MAX(c.x, d.x) - a.x \tag{31}$$

MAX$(c.x, d.x)$ is the maximum value of the horizontal coordinates for the top right and bottom right of the image.

The estimated value of the four vertices after straightening are: Left upper: $(a.x, a.y)$, Left bottom: $(b.x, b.y)$, Right upper: $(c^{'}.x, c^{'}.y) = (X, a.y)$, and Right bottom: $(d^{'}.x, d^{'}.y) = (X, b.y)$.

3. With matrix operation, a transformation matrix is calculated by the formula (32)

$$[u, v, w] = [x, y, 1] \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{bmatrix} \tag{32}$$

x and y are horizontal and vertical coordinates of the original image. $x^{'}$ and $y^{'}$ are horizontal and vertical coordinates of the target image. As shown in following formulas,

$$x^{'} = \frac{u}{w} = \frac{a_{00}x + a_{10}y + a_{20}}{a_{02}x + a_{12}y + a_{22}} \tag{33}$$

$$y^{'} = \frac{v}{w} = \frac{a_{01}x + a_{11}y + a_{21}}{a_{02}x + a_{12}y + a_{22}} \tag{34}$$

Eight unknown parameters in the transformation matrix can be solved by solving the linear equations, which are obtained by four pairs of vertex coordinates.

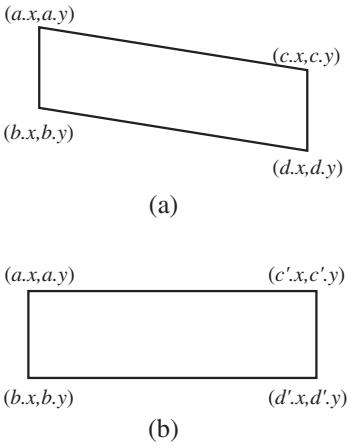


Fig. 14. Comparison of stitching result before straightening and after straightening: (a) stitching result before straightening, (b) stitching result after straightening

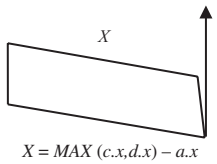


Fig. 15. Estimate the length of the image after straightening

(a) (b)

Fig. 16. Comparison of stitching result: (a) panorama before straightening (b) panorama after straightening

Fig. 17. Original image sequence (six images)

4. The transform matrix is used to correct the global image, and the bilinear interpolation method is used to carry out the interpolation for the pixels of the transformed image. The straightened images are shown in Figs 16(a) and 19(b).

In this paper, the automatic image straightening algorithm has restrictions on the acquisition of image. It does not allow a strong change of position when taking pictures. So, the image straightening algorithm proposed in this paper is not suitable for waveform distortion and rapid changes in the distortion. It is only suitable for tilt distortion due to slight rotation and cumulative error caused by the shot. Image straightening algorithm straightens the stitching result. It is equal to the correction of a distorted image and makes a nonrectangular image into a rectangular image.

6. The Experimental Results and Analysis

We select two different image sequences from natural scenes to demonstrate the performance of our algorithm. The experimental software and hardware environment are: Intel CPU: (R) Core (TM) 2.50 GHz i5−4200 M, Windows OS: 7, OpenCV 3.0.0 Library.

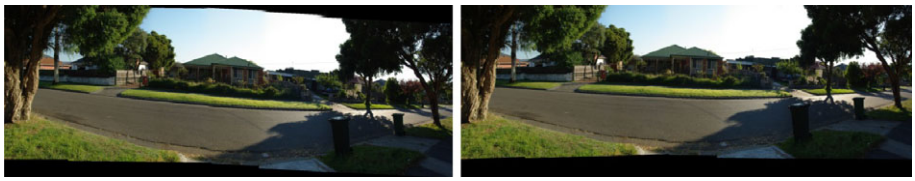
Figures 17 and 18 are six and five original image sequences, respectively. The resolution of Fig. 17 is 384×512. The resolution of Fig. 18 is 968×648. Figures 16(a) and 19(a) are obtained by image stitching from left to right in turn. Figures 16(b) and 19(b) are the results after automatic straightening.

The coordinate point P_1 is the midpoint of the left boundary for the panoramic image, and the point P_2 is the midpoint of the right edge for the panoramic image. As shown in Fig. 20, two coordinate points are connected into a straight line and get the included angle θ between this line and the horizontal line. This method is used to approximate the degree of inclination for the panorama to get the Table III.

Figures 16 and 19 show that the whole straightening of the image can effectively improve the tilt distortion of the panoramic image. Meanwhile, this method improves the quality of the panoramic image and provides less complexity. From Table IV, it can be seen that the degree of inclination for panorama after automatic straightening is close to zero. Compared to panorama before straightening, the results after automatic straightening are greatly improved.

Based on experimental results of the above, it is proven that A-KAZE outperforms SIFT in regard of feature point extraction speed. Figure 21 indicates that the effective feature points detected in this paper and the number of correct matching pairs is higher compared with traditional feature matching algorithm. From

Fig. 18. Original image sequence (five images)



(a) (b)

Fig. 19. Comparison of stitching result: (a) panorama before straightening (b) panorama after straightening

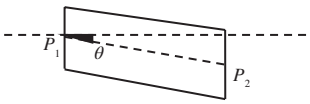


Fig. 20. The degree of inclination

Table III. The comparison data for error of matrix H

Figure	Traditional algorithm	The algorithm in this paper
Figs 1 and 2	0.0300	0.0147
Figs 3 and 4	0.0015	0.0007
Figs 5 and 6	0.0079	0.0040

Table IV. The comparison data of panorama inclination

Figure	Inclination of panorama from left to right in turn stitching	Inclination of panorama obtained by the method in this paper
Fig. 16(a)	1.10°	0.04°
Fig. 19(a)	4.68°	0.05°

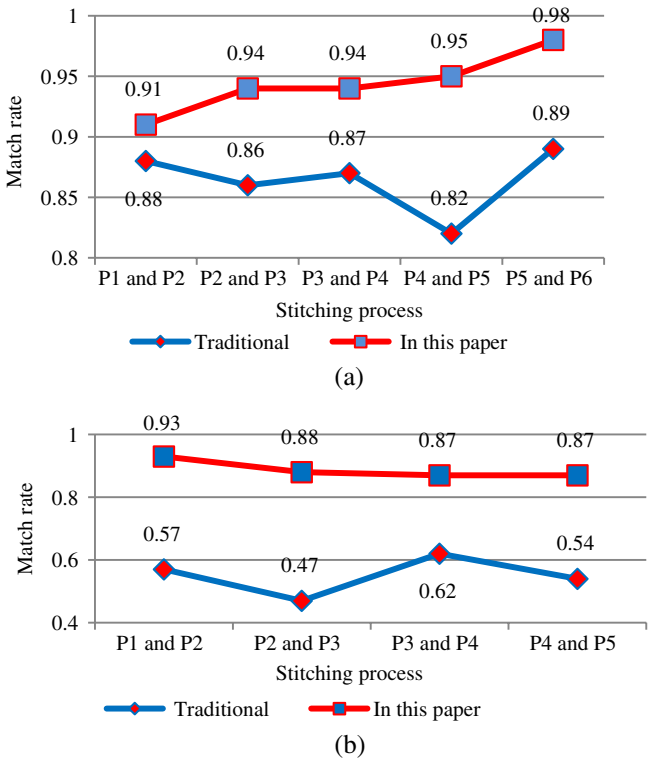


Fig. 21. The comparison data of match rate between the traditional SIFT feature matching and the method in this paper: (a) The match rate of Fig. 17 (b) The match rate of Fig. 18

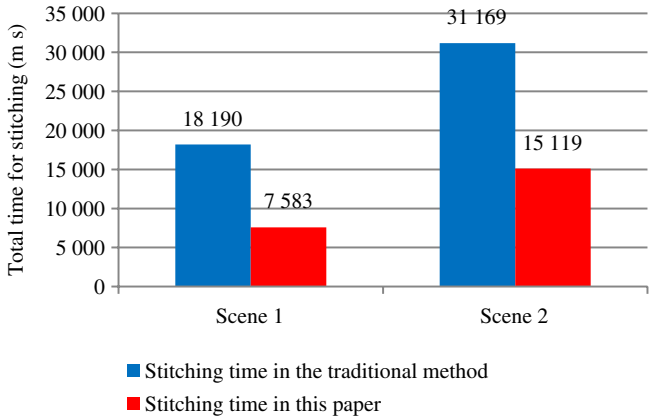


Fig. 22. The comparison data of stitching time between the traditional method and the method in this paper

Fig. 22, it can be clearly seen that the time cost of stitching image in this paper is lower and the efficiency is higher. All of the experimental results above show that this algorithm can ensure the accuracy and improve the efficiency of image stitching.

7. Conclusion

Different from the traditional stitching method, the image stitching algorithm proposed in this paper takes advantages of A-KAZE and solves the problems of obscure boundary and detail lost. At the same time, this algorithm reduces the complexity of feature extraction, shortens the time for feature extraction, and improves efficiency of image stitching. An improved Laplacian multiresolution fusion algorithm is used to eliminate the stitching line. Finally, the panoramic image is automatically straightened in order to effectively solve the problem of tilt. The experimental results show that this algorithm not only improves the efficiency of the image mosaic, but also realizes the seamless image stitching.

Acknowledgments

This work is supported by Chongqing Basic and Frontier Research Project (grant number cstc2015jcyjBX0090, cstc2014jcyjA40033 and cstc2015jcyjA40034), and Outstanding Achievements Transformation Projects of University in Chongqing (grant number KJZH14219). The authors wish to thank the associate editors and anonymous reviewers for their valuable comments and suggestions on this paper.

References

(1) Brown M, Lowe DG. Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision* 2007; **74**(1):59–73.
(2) Lowe DG. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 2004; **60**(2):91–110.
(3) Bay H, Ess A, Tuytelaars T, Gool LV. Speeded-up robust features (SURF). *Computer Vision and Image Understanding* 2008; **110**(3):346–359.

(4) Alcantarilla P, Nuevo J, Bartoli A. Fast explicit diffusion for accelerated features in nonlinear scale spaces. *British Machine Vision Conference*, Bristol, UK, 2013; 13.1–13.11.
(5) Alcantarilla PF, Bartoli A, Davison AJ. KAZE features. *Lecture Notes in Computer Science* 2012; **7577**(1):214–227.
(6) Huang CM, Lin SW, Chen JH. Efficient image stitching of continuous image sequence with image and seam selections. *IEEE Sensors Journal* 2015; **15**(10):5910–5918.
(7) Liu Z, Blasch E, Xue Z, Zhao JY, Laganiere R, Wu W. Objective assessment of multiresolution image fusion algorithms for context enhancement in night vision: a comparative study. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2012; **34**(1):94–109.
(8) Wang Z, Chen Y, Zhu Z, Zhao W. An automatic panoramic image mosaic method based on graph model. *Multimedia Tools and Applications* 2016; **75**(5):2725–2740.
(9) Chertock A, Kurganov A, Petrova G. Fast explicit operator splitting method for convection–diffusion equations. *International Journal for Numerical Methods in Fluids* 2009; **59**(3):309–332.
(10) Yang X, Cheng KT. LDB: an ultra-fast feature for scalable Augmented Reality on mobile devices. *IEEE International Symposium on Mixed and Augmented Reality*, Atlanta, GA, 2012; 49–57.
(11) Chen C, Xu Z. A novel binary feature descriptor for accelerated robust matching. *Journal of Convergence Information Technology* 2013; **8**(3):72–79.
(12) Tang Y, Shin J, Liao HC. De-ghosting method for image stitching. *International Journal of Digital Content Technology and Its Applications* 2012; **6**(18):17–24.
(13) Kumar M, Conger DD, Miller RL, Luo J, Radha H. A distortion-sensitive seam carving algorithm for content-aware image resizing. *Journal of Signal Processing Systems* 2011; **65**(2):159–169.

Zhong Qu (Non-member) received the Ph.D. degree in Computer Application Technology from Chongqing University in 2009 and the M.S. degree in Computer Architecture from Chongqing University in 2003. He is currently a professor in Chongqing University of Posts and Telecommunications. His research interests are in the areas of digital image processing, digital media technology, and cloud computing.

Wei Bu (Non-member) is a M.S. degree candidate of Chongqing University of Posts and Telecommunications in Computer Science and Technology. Her main research interest is digital image processing.

Ling Liu (Non-member) received the M.S. degree in Computer Technology from Chongqing University of Posts and Telecommunications in 2014. Her main research interest is digital image processing.