PAPER

Fast Image Mosaicing Based on Histograms

Akihiro MORI† ***and*** **Seiichi UCHIDA**†a), ***Members***

SUMMARY This paper introduces a fast image mosaicing technique that does not require costly search on image domain (e.g., pixel-to-pixel correspondence search on the image domain) and the iterative optimization (e.g., gradient-based optimization, iterative optimization, and random optimization) of geometric transformation parameter. The proposed technique is organized in a two-step manner. At both steps, histograms are fully utilized for high computational efficiency. At the first step, a histogram of pixel feature values is utilized to detect pairs of pixels with the same rare feature values as candidates of corresponding pixel pairs. At the second step, a histogram of transformation parameter values is utilized to determine the most reliable transformation parameter value. Experimental results showed that the proposed technique can provide reasonable mosaicing results in most cases with very feasible computations.

key words: *mosaicing, histogram, correspondence problem*

1. Introduction

Mosaicing is a technique for stitching two (or more) images of the same scene. Generally, those images have overlapping areas and thus their registration by geometric transformation is a key task of mosaicing. Mosaicing has been applied to remote sensing images and aerial photographs for creating wide-area images. Recently, mosaicing has been applied to document images [1] for camera-based OCR. Mosaicing of video frames, called video mosaicing, is also an emerging application [2]–[4].

In this paper, we propose a new mosaicing technique which fully utilizes histograms. The proposed technique is comprised of two steps:

- The first step: Given two images having an overlap (Fig. 1), candidates of corresponding pixel pairs between those images are selected efficiently by using *pixel feature histograms* of both images (Fig. 2).
- The second step: The most reliable parameter value of geometric transformation is estimated by employing another histogram, called *parameter histogram*. The parameter histogram is useful for estimating the optimal transformation parameter while excluding the effect of erroneous pixel pairs caused by the first step.

Using the transformation parameter estimated by the second step, the mosaicing result (Fig. 3) is finally provided.

The main merit of the proposed technique is its efficient pixel-to-pixel correspondence search on the pixel feature histogram. Specifically, the proposed technique does not require two costly procedures, i.e., (i) correspondence search on image domain and (ii) iterative optimization (e.g., gradient-based optimization and random optimization) of transformation parameter values. Another merit is that the establishment of corresponding pixel pairs and the optimization of the parameter value are *consistently* performed on the same data structure, i.e., histogram. This consistency is quite useful when constructing an entire mosaicing system on a single hardware for fast histogram computing, such as [5], [6].

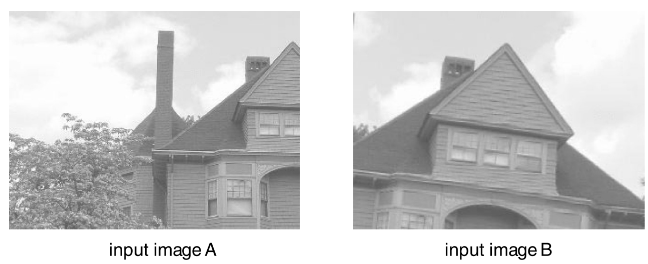


Fig. 1 A pair of images with an overlap.

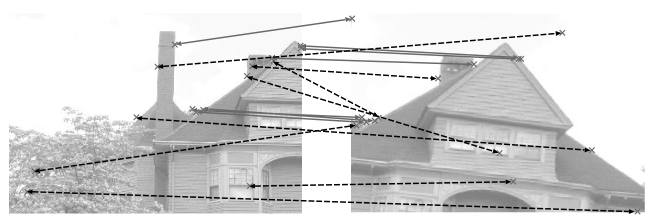


Fig. 2 Candidates of corresponding pixel pairs. (Solid line: correct corresponding pixel pair. Broken line: false corresponding pixel pair)

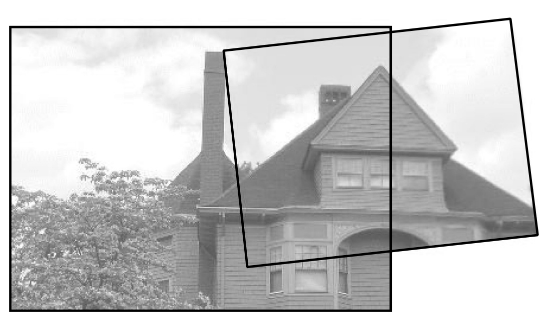


Fig. 3 A result of mosaicing.

Manuscript received August 14, 2006.
Manuscript revised December 25, 2007.

†The authors are with the Graduate School of Information Science and Electrical Engineering, Kyushu University, Fukuoka-shi, 812–8581 Japan.
a) E-mail: uchida@is.kyushu-u.ac.jp

DOI: 10.1093/ietisy/e91–d.11.2701

In order to show the above main merit clearly, we make several assumptions which simplify the mosaicing problem. The first assumption is the use of a simple similarity distortion model on registrating a pair of images. According to this assumption, we will tackle mosaicing problems of originally "flat" pictures, such as scanned photographs and document images. The second is the use of a simple RGB color feature as the pixel feature. The third is that exposure is almost fixed for all the images. All of those assumption can be removed rather easily in our future work because of the extensibility of the proposed technique.

2. Related Work

The main novelty of the proposed technique is the use of the pixel feature histogram to find corresponding pixel pairs with very low complexity. The simplest technique to find corresponding pixel pairs is two-dimensional search on image domain, which will require $O(N^4)$ computations where N is the height (or width) of a subjected image. For avoiding this intensive computations, many previous techniques based on matching of several pre-determined P feature points, or distinctive pixels, perform nearest neighbor search on feature domain (e.g., [7]–[9]). Those techniques will require less computations than the simple two-dimensional search; it, however, requires $O(N^2)$ computations for detecting P feature points and additional computations ($O(P^2)$ at most) for nearest neighbor search for establishing their correspondence. The important property which differentiates the proposed technique from the above previous techniques is that extraction of distinctive pixels and their correspondence search are done efficiently on the pixel feature histogram. In fact, the proposed technique requires only $O(N^2)$ computations for establishing pixel-to-pixel correspondence, while most of the following discussion will be done with a simpler version which requires $O(N^2) + O(M)$ computations, where M is the number of bins in the pixel feature histograms.

In contrast to the novelty of the pixel feature histogram, the parameter histogram is not a ground-breaking scheme and known as a special case of Hough transform [10]. Zhang [11] outlines histogram-based estimation framework and summarizes several improvement tips. Pose clustering by Stockman [12] is classic. Lowe [8] also has employed a histogram-based estimation technique to choose correct corresponding pixel pairs determined in his SIFT feature space. Hsieh et al. [13] have proposed a registration technique based on a histogram of the difference between edge angles of corresponding pixels. Generalized Hough transformation also a histogram-based parameter estimation technique, while it is not closely related to the proposed technique.

Note that our parameter histogram is slightly different from the histograms frequently used in the above attempts. As discussed below, we will use an adaptive histogram computed by a coarse-to-fine strategy instead of a conventional pre-specified histogram whose bin size is fixed.

In principle, we can use other techniques in the second step. RANSAC [14] is a promising alternative. In RANSAC, several landmark pixels are randomly paired between two images and then the validity (i.e., the degree of consensus) of the transformation parameter value derived from those pairs is examined using other landmark pixels. We, however, adhere to our histogram-based parameter estimation technique. This is because we cannot expect that RANSAC will provide correct transformation parameter by using pixel pairs selected by the first step. Lowe [8] has pointed-out that RANSAC will fail in the case that there are more erroneous pixel pairs than correct pairs. This is exactly our case, since most (70 ∼ 90%) of corresponding pixel pairs by the first step are erroneous. In addition, it is desirable to keep architectural consistency that the entire mosaicing procedure is based on histograms as noted in Sect. 1.

It is interesting to note that, Kim et al. [15] has proposed a histogram-based exposure adjustment technique. Thus, if we employ it as the preprocessing of the proposed technique, we can develop a complete image mosaicing system which totally relies on histograms.

3. The First Step: Selection of Candidates of Corresponding Pixel Pairs Using Pixel Feature Histogram

Let us consider a pair of images, $\boldsymbol{A}$ and $\boldsymbol{B}$, and assume that they have an overlap. Figure 1 shows an example. For the registration (i.e., mosaicing) of those images, pixel pairs having the same and infrequent feature values are selected as candidates of corresponding pixels. This selection can be done very efficiently utilizing pixel feature histograms as follows.

Figure 4 illustrates the main idea of the selection. For each of $\boldsymbol{A}$ and $\boldsymbol{B}$, a histogram of pixel feature values is created and then "infrequent bins" where the two histograms commonly have only one entry are searched for. Since we assume the fixed exposure, it is possible to select the two pixels voted into the infrequent bin as a candidate of corresponding pixels. In other words, if each of image $\boldsymbol{A}$ and $\boldsymbol{B}$ has only one pixel having a certain RGB value, those two pixels are paired as a candidate. If $\boldsymbol{C}$ denotes the set of the candidates and M denotes the number of bins, clearly $|\boldsymbol{C}| \leq M$.

Figure 2 shows the candidates of corresponding pixel pairs between the two images of Fig. 1 provided by the above selection procedure. Three-dimensional RGB color vector was used as pixel feature and therefore a three-dimensional histogram was created for each image. Since the color feature has large ambiguity on establishing pixel correspondence (that is, different objects may have the same color), the candidates provided include many false correspondences. Those erroneous candidates, however, will be automatically neglected in the following step where the most reliable transformation parameter is estimated using a histogram.

The above procedure requires $O(N^2) + O(M)$ computa-

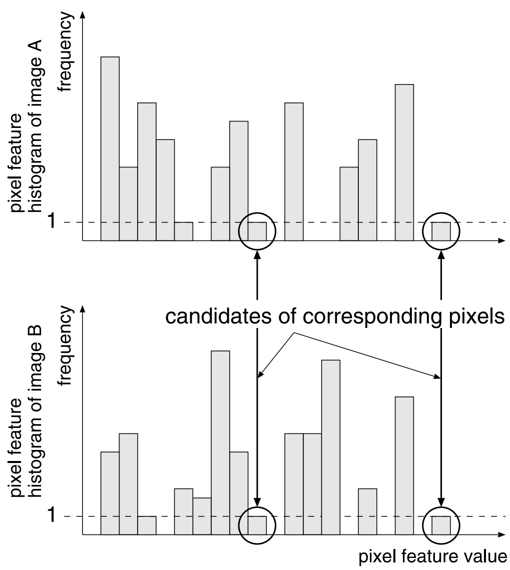


Fig. 4 Selection of candidates of corresponding pixel pairs using pixel feature histograms.

tions, where N is the height (or width) of the subjected image. Specifically, creation of two pixel feature histograms requires $O(N^2) + O(N^2) \sim O(N^2)$ computations and search of the infrequent bins from the histograms requires $O(M)$ computations. It is noteworthy that this computational complexity is independent of the number of the candidates, $|\boldsymbol{C}|$.

The performance of the selection procedure depends on the width of each bin of the histogram. A narrower bin will make the selection procedure sensitive to small change of pixel feature values and thus truly corresponding pixels may be voted into different bins. On the other hand, a wider bin will reduce pixels with infrequent features. Unfortunately, the optimal bin width for a given image pair is not obvious. In the following experiments, the bin width was fixed at 5. This means that a pixel feature histogram for 256-level RGB color feature was comprised of $(256/5)^3 \sim 1.3 \times 10^5$ bins.

While the experiment will be done with the above simple procedure, its computational complexity can be reduced to $O(N^2)$ with a slight modification. As detailed in Appendix, the reduction is realized by the fact that it is not necessary to prepare complete histograms for both images for our task. This reduction emphasizes the computational efficiency of the proposed technique over the previous techniques, which require not only $O(N^2)$ computations for detecting P distinctive pixels but also extra computations ($< O(P^2)$) for establishing their correspondence.

4. The Second Step: Estimation of Transformation Parameter Using Parameter Histogram

4.1 Transformation Parameter

As noted before, we assume similarity transformation, or Helmert transformation, for the registration of the subjected images, $\boldsymbol{A}$ and $\boldsymbol{B}$. Similarity transformation represents translation, rotation, and scaling by controlling four parameters $\{a, b, c, d\}$ as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c \\ d \end{bmatrix}. \tag{1}$$

The four parameters $\{a, b, c, d\}$ can be uniquely determined by using two pairs of corresponding pixels, $(x_1, y_1) \leftrightarrow (x_2, y_2)$ and $(x_3, y_3) \leftrightarrow (x_4, y_4)$, where (x_1, y_1) and (x_3, y_3) are pixels of $\boldsymbol{A}$ and (x_2, y_2) and (x_4, y_4) are pixels of $\boldsymbol{B}$.

The four parameters $\{a, b, c, d\}$ can be converted into more intuitive parameters, translation dx, dy, rotation dt, an scaling s, by

$$dx = c,\ dy = d,\ d\theta = \tan^{-1}(b/a),\ s = \sqrt{a^2 + b^2}. \tag{2}$$

4.2 Parameter Histogram

For the appropriate registration of $\boldsymbol{A}$ and $\boldsymbol{B}$, we should estimate the most reliable transformation parameter using the candidate set $\boldsymbol{C}$. Candidates from $\boldsymbol{C}$, however, are often erroneous due to the ambiguity of the pixel feature, as shown in Fig. 2. Those erroneous candidates should be excluded from transformation parameter estimation.

For robust estimation of the transformation parameter excluding the effect of the erroneous candidates, a parameter histogram is introduced. The parameter histogram is the histogram of the four-dimensional transformation parameter vectors, $(dx, dy, d\theta, s)$. As noted in Sect. 4.1, a parameter vector is provided by arbitrary combination of two corresponding pixel pairs in $\boldsymbol{C}$. Thus, the number of entries of the parameter histogram is $|\boldsymbol{C}|(|\boldsymbol{C}| - 1)/2$. Note that the parameter vector is discarded if it exceeds a pre-defined range. Thus the actual number of the entries is fewer than $|\boldsymbol{C}|(|\boldsymbol{C}| - 1)/2$.

The correct candidates in $\boldsymbol{C}$ will form a peak on the parameter histogram. The erroneous candidates will not form any peak because they will not have any mutual relation. Thus, using the peak (i.e., the mode) on the parameter histogram, the most reliable transformation parameter vector can be estimated while excluding the effect of erroneous candidates in $\boldsymbol{C}$. An efficient peak detection technique will be described in the next Sect. 4.3.

Figure 5 shows the distribution of the parameter vectors by all the pixel correspondence candidates in $\boldsymbol{C}$ for an image pair whose true transformation parameter is known. The distribution shows a peak around the true parameter value $((dx, dy) = (180.2, 20.5))$. The distribution also shows that transformation parameters derived from the erroneous candidates do not have any strong peak.

4.3 Peak Detection in Parameter Histogram

As noted in Sect. 3, the number of the candidates, $|\boldsymbol{C}|$, heavily depends on not only the subjected image pair but also the pixel feature (color, moment, etc.). This fact means that it is difficult to guess the number of entries of the parameter histogram and the appropriate bin width for detecting the peak.

Thus, we employ a coarse-to-fine strategy where the bin width is adaptively and automatically determined according to the distribution of the parameter vectors. In this strategy, a bin having more entries than a threshold ϵ is equally partitioned into 2^D bins recursively, where D is the dimensionality of the parameter space and thus $D = 4$ for the similarity distortions. Figure 6 shows an example of the parameter histogram created by this strategy, where only a two-dimensional subspace of the four-dimensional parameter space were plotted like Fig. 5. The position where two dotted lines are intersecting orthogonally indicates the true parameter value.

If no bin has more entries than ϵ, peak bins are detected as the bins having the narrowest width. Note that there are at least 2^D peak bins. Among those peak bins, the $K(< 2^D)$ bins with the most entries are further selected. Then at each of the K bins, its entries (i.e., transformation parameter vectors) are averaged for K average transformation parameter vectors. Finally, the averaged transformation parameter vector which provides the minimum image difference on the overlapping area is determined as the most reliable parameter vector.

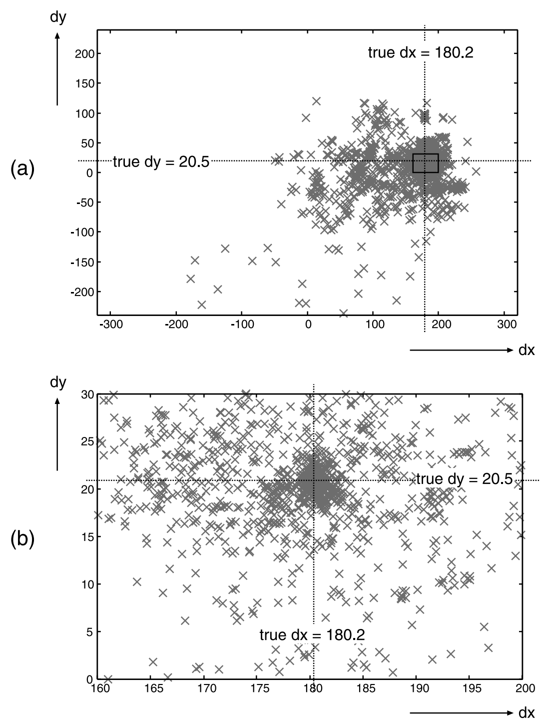


Fig. 5 (a) Distribution of transformation parameters by all the pixel correspondence candidates in $\boldsymbol{C}$. Only (dx, dy)-subspace is illustrated while the original parameter space is four-dimensional. (b) Close-up of the small rectangular area in (a).

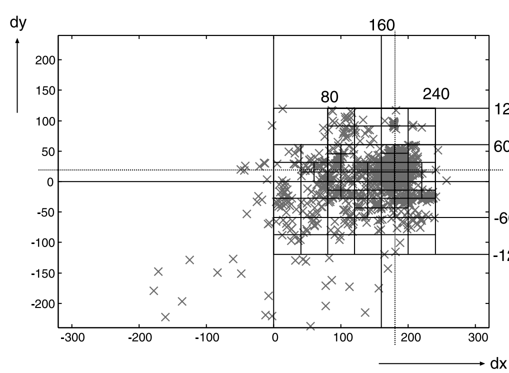


Fig. 6 Coarse-to-fine partition for creating parameter histogram.

5. Experimental Results

5.1 Mosaicing of Synthetic Image Pairs

5.1.1 Samples

An experiment was performed using 50 synthetic image pairs which are available for research purposes at [16] (the level3 set). Paired images were taken from the same photograph (of some real scene) with an overlap and one of the pair was transformed by a specific parameter vector $(dx, dt, d\theta, s)$. Thus, the true transformation parameter vector was known in this experiment. Note that even in these synthetic pairs, the color features of corresponding pixels were not the same because sub-pixel color interpolation was performed at the transformation for creating image pairs. Image sizes varied from 160×120 to 640×480.

The mosaicing results were evaluated in their registration accuracies and computation times. A computer with Pentium4 processor (2.80 GHz) and 1024 MB RAM was used. The posteriori evaluation of Sect. 4.3 was employed at $K = 5$.

5.1.2 Accuracy

Figure 7 shows the accuracies of 50 mosaicing results. The (in)accuracy was measured by the L^2-error of the estimated parameter value. If a result with a small error (< 1.0%) was

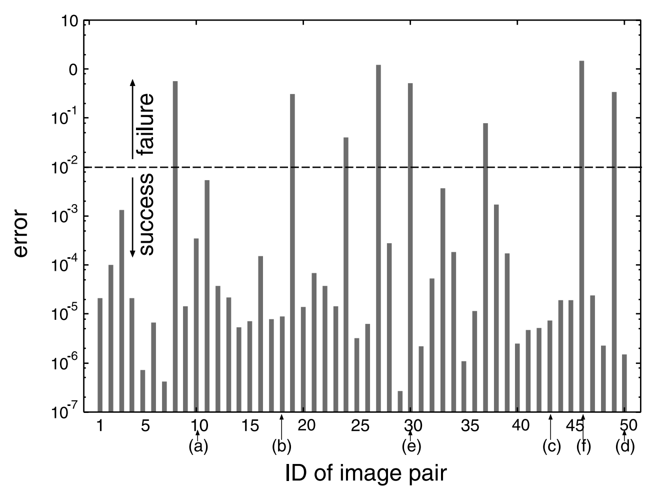


Fig. 7 Error of the estimated parameter vector. IDs of (a)–(f) correspond to the pairs of Fig. 8.

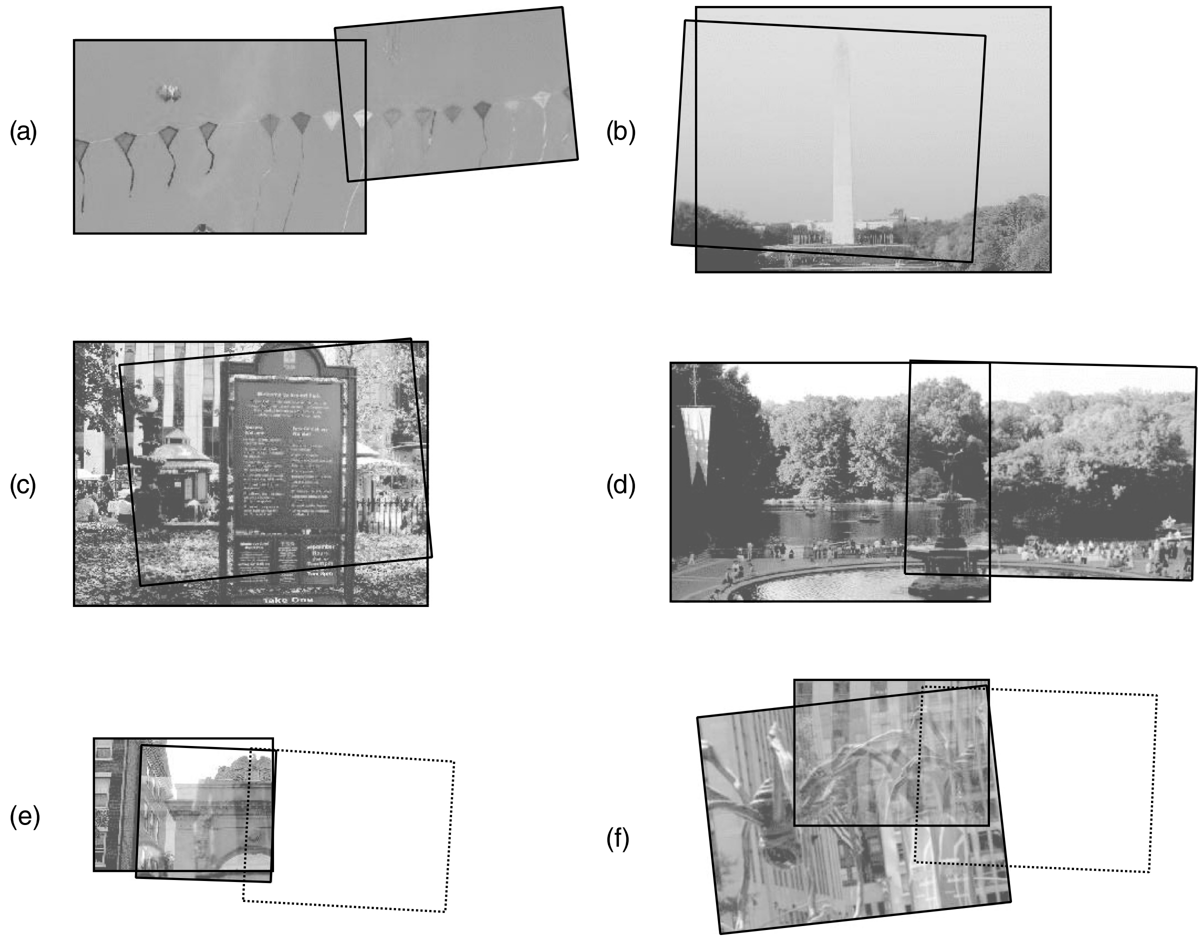


Fig. 8 Mosaicing results for several synthetic image pairs. (a)~(d): Successful results. (e), (f): Failure results. The true transformation is shown by a dotted rectangular.

treated as a successful result, the success rate has reached 84% (= 42/50). Note that this success/failure decision rule could show a good agreement with the decision by human's eyes.

Figure 8 shows several mosaicing results. Among those results, (a)~(d) are successful results (i.e., registration error < 1.0%) and (e) and (f) are failure results. In (b)–(d), the highest peak on the parameter histogram was successfully found at the true transformation parameter value. In (a), although the highest peak was not found at the true value due to a small overlap area, the posteriori evaluation helped to select the true value at the second peak.

Figure 9 shows the number of corresponding pixel pairs selected by the first step, i.e., $|\boldsymbol{C}|$. The average, minimum, and maximum numbers of pixel pairs were 192, 37, and 666, respectively. From those pairs, 4400, 51, and 57,608 transformation parameter vectors were calculated and voted into the parameter histogram.

Figure 9 also shows the ratio of correct corresponding pixel pairs to the total pairs. The ratio reveals that most corresponding pixel pairs were erroneous; only 15.2% were correct pairs on average. This fact emphasizes the robustness of the second step, i.e., the histogram-based parameter estimation. In addition, it also reveals that RANSAC will not be able to provide correct parameters as noted in Sect. 2.

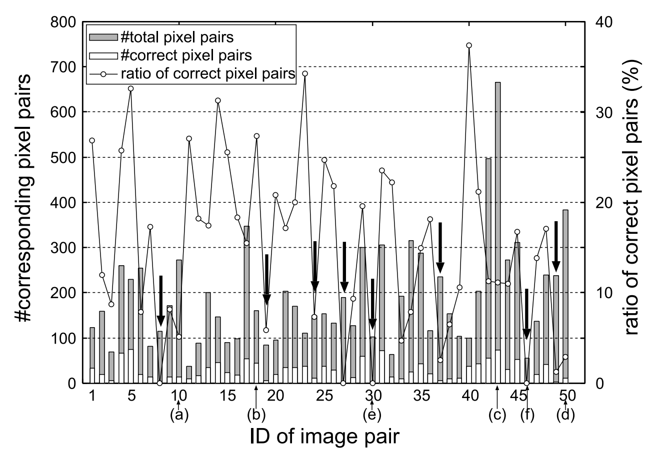


Fig. 9 The number of the candidates of corresponding pixel pairs (i.e., $|\boldsymbol{C}|$) and the ratio of correct pixel pairs. IDs of (a)–(f) correspond to the pairs of Fig. 8. Down-arrows indicate failure pairs.

Eight failure image pairs, which are indicated by down-arrows in Fig. 9, often had fewer correct corresponding pixel pairs. In fact, the correct pair ratios of those failure image pairs were below 7% (Conversely, all the image pairs with a correct pair ratio above 7% were *always* mosaiced successfully). A possible remedy to increase the correct pair ratio is the use of less-ambiguous (i.e., distinctive) pixel features. SIFT feature [8] is a promising choice, while its high dimensionality is a hurdle on creating its pixel feature histogram.

5.1.3 Computation Time

Figure 10 shows total computation time for mosaicing each image pair. Again, the experimental results in this section was obtained with the procedure in Sect. 3, i.e., the procedure which requires $O(N^2) + O(M)$ computations. The average, minimum, and maximum computation time were about 0.39, 0.05, and 2.0 s, respectively. The computational efficiency of the proposed technique was proved by comparing this result with the following result; a coarse-to-fine image mosaicing technique which based on an exhaustive search (where all possible quantized transformation parameter values are examined) required 380 s on average (This technique is a modified version of the test program published in [16]). In addition, the average computation time of 0.39 s is very comparable with other recent "fast" mosaicing technique, such as [17].

Figure 10 also shows the ratio of the computation time of the first step to the total computation time. The first step required only 3.9% of the total computation time on average. This result brings out the merit of the novel pixel feature histogram for establishing corresponding pixel pairs.

As shown in Fig. 10, one image pair (which corresponds to Fig. 8 (c)) required extremely longer time (2.0 s). Its large overlapping area with fine texture yields $|\boldsymbol{C}| = 666$ corresponding pixel pairs. From those pairs, $O(|\boldsymbol{C}|^2) \sim 57{,}608$ parameter vectors were voted to the parameter histogram.

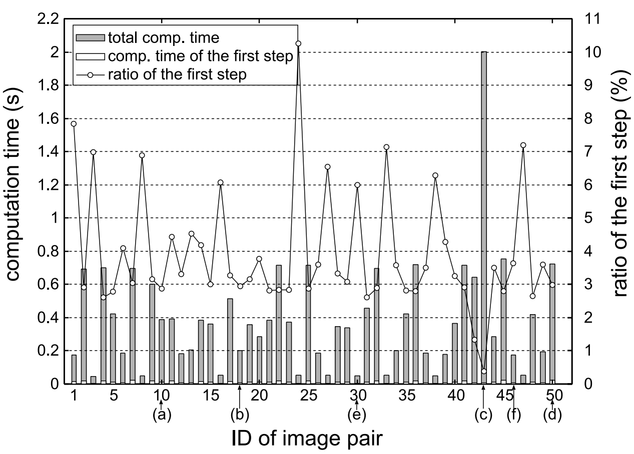


Fig. 10 Total computation time. The computation time of the first step (for selecting the candidates of corresponding pixel pairs) and its ratio to the total computation time are also plotted. IDs of (a)–(f) correspond to the pairs of Fig. 8.

5.2 Mosaicing of Real Image Pairs

Another experiment was performed using 120 real image pairs acquired by a digital camera. Each image pair was comprised of two overlapping front-parallel images capturing a color document on a uniform background. Different from the foregoing experiment, two paired images were acquired independently. Thus, paired images might have different exposures and correct corresponding pixels might have different colors.

While it is difficult to evaluate the accuracy of the mosaicing results quantitatively due to the lack of true parameter values, about half results were successful or showed slight errors which are observed like blur. Figure 11 shows two successful results. There was no large difference in computation time between this experiment and the foregoing experiment on the synthetic image pairs.

Several results with severe errors were also found

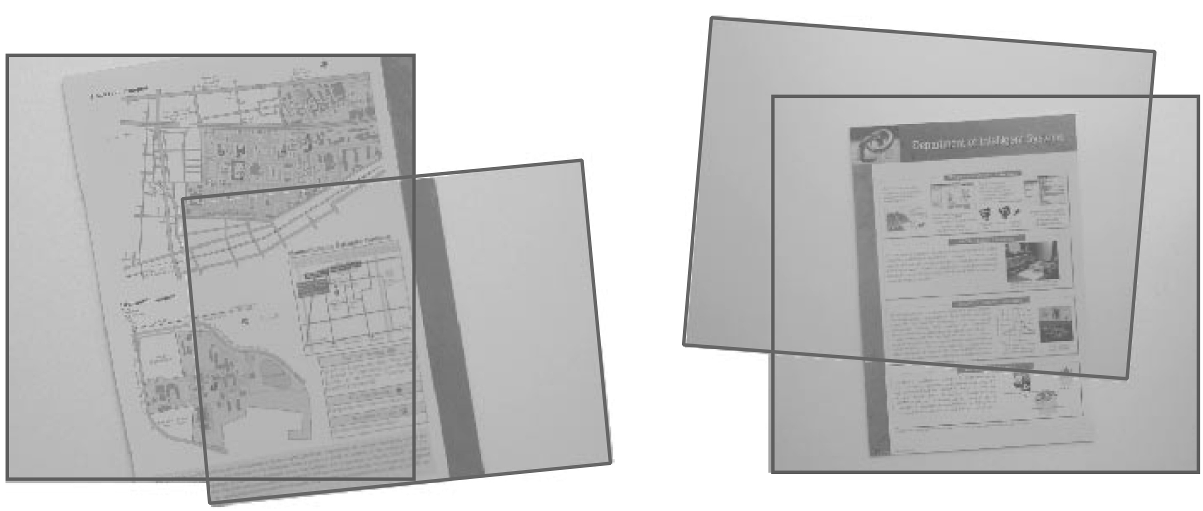


Fig. 11 Mosaicing of real image pairs.

mainly due to different exposures of paired images. That is, different exposures caused many erroneous pixel pairs and obscured correct pairs. Clearly, introduction of some exposure adjustment and/or pixel features other than naive RGB feature are necessary as our future work.

The similarity distortion model, which is assumed throughout this paper, is a hurdle for dealing with image pairs which undergo perspective distortion. In fact, the mosaicing results on real scenery images from a public database [18] were unsatisfactory. For dealing with perspective distortion, we must extend the parameter histogram to represent eight parameters of perspective transformation. While the idea of this extension is straightforward, it may be necessary to introduce several practical improvements in addition to the coarse-to-fine strategy.

6. Conclusion

A fast image mosaicing technique using two kinds of histograms was proposed. In the proposed technique, the candidates of corresponding pixel pairs are efficiently selected using a histogram of pixel feature values. Possible transformation parameter values derived from those candidates are then voted into another histogram, called parameter histogram, for determining the most reliable transformation parameter value. The proposed technique is fast because it does not require costly search on image domain and iterative parameter optimization. In addition, the proposed technique is robust to erroneous candidates of the corresponding pixel pairs because the effect of erroneous candidates is automatically excluded at the determination of the transformation parameter value.

Future work will be focused on the removal of the three assumptions introduced in Sect. 1, i.e., the assumption of the simple similarity distortion model, the assumption of the simple RGB pixel feature, and the assumption of fixed exposure. As experimentally revealed in Sect. 5.2, the perspective distortion model is necessary to deal with real scenery images. Pixel features other than RGB feature should be examined to reduce erroneous pixel pairs by the parameter histogram. Exposure adjustment, such as Kim et al. [15], is inevitable for registrating real image pairs as also revealed in Sect. 5.2. Experimental comparison with other mosaicing techniques is another important future work, although this paper has experimentally proved the superiority of the proposed technique over a simple coarse-to-fine mosaicing method in computational efficiency.

Acknowledgment

This research started for the 8th IEICE PRMU algorithm contest at image mosaicing. The authors thank Dr. N. Hagita (ATR), who was the organizing chair of the contest, Prof. S. Omachi (Tohoku Univ.), and other committee members of the contest for giving us the opportunity of the research. The fact that the best algorithm award of the contest went the proposed algorithm greatly encouraged the authors. The authors thank Prof. S. Omachi cordially again for permitting us to use sample image pairs prepared for the contest. This work was supported in part by the Research Grant (No.17700198) of The Ministry of Education, Culture, Sports, Science and Technology in Japan.

References

[1] A. Zap palà, A. Gee, and M. Taylor, "Document mosaicing," Image Vis. Comput., vol.17, no.8, pp.589–595, June 1999.
[2] H. Li and D. Doermann, "Text enhancement in digital video using multiple frame integration," Proc. ACM Multimedia, pp.19–22, 1999.
[3] T. Sato, S. Ikeda, M. Kanbara, A. Iketani, N. Nakajima, N. Yokoya, and K. Yamada, "High-resolution video mosaicing for documents and photos by estimating camera motion," Computational Imaging II (Proc. SPIE, vol.5299), pp.246–253, 2004.
[4] H. Miyazaki, S. Uchida, and H. Sakoe, "Mosaicing-by-recognition: A technique for video-based text recognition," Proc. Int. Conf. Doc. Anal. Recog., vol.2 of 2, pp.904–908, 2005.
[5] H.-D. Cheng, X. Li, and L. Wang, "Parallel VLSI-oriented algorithm and architecture for computing histogram of images," Proc. Visual Communications and Image Processing '93 (Proc. SPIE, vol.2094), pp.1086–1093, 1993.
[6] J.-W. Jang, H. Park, and V.K. Prasanna, "A fast algorithm for computing a histogram on reconfigurable mesh," IEEE Trans. Pattern Anal. Mach. Intell., vol.17, no.2, pp.97–106, 1995.
[7] M. Brown and D.G. Lowe, "Recognising panoramas," Proc. Int. Conf. Comput. Vis., pp.1218–1225, 2003.
[8] D.G. Lowe, "Distinctive image features for scale-invariant keypoints," Int. J. Comput. Vis., vol.60, no.2, pp.91–110, 2004.
[9] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," IEEE Trans. Pattern Anal. Mach. Intell., vol.27, no.10, pp.1615–1630, 2005.
[10] J. Illingworth and J. Kittler, "A survey of the Hough transform," Comput. Vis. Graph. Image Process., vol.44, pp.87–116, 1988.
[11] Z. Zhang, "Parameter estimation techniques: A tutorial with application to conic fitting," Image Vis. Comput., vol.15, pp.59–76, 1997.
[12] G. Stockman, "Object recognition and localization via pose clustering," Comput. Vis. Graph. Image Process., vol.40, pp.361–387, 1987.
[13] J.-W. Hsieh, H.-Y.M. Lao, K.-C. Fan, M.-T. Ko, and Y.-P. Hung, "Image registration using a new edge-based approach," Image Vis. Comput. Anal., vol.67, no.2, pp.112–130, 1997.
[14] M.A. Fischler and R.C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," Commun. ACM, vol.24, no.6, pp.381–395, 1981.
[15] S.J. Kim and M. Pollefeys, "Radiometric alignment of image sequences," Proc. CVPR2004, pp.I-645–I-651, 2004.
[16] http://www.aso.ecei.tohoku.ac.jp/alcon2004/
[17] F. Isgró and M. Pilu, "A fast and robust image registration method based on an early consensus paradigm," Pattern Recognit. Lett., vol.25, pp.943–954, 2004.
[18] http://www.cs.ubc.ca/˜mbrown/panorama/testimages.zip

Appendix: Improvement of the First Step

The algorithm of the first step in Sect. 3 is comprised of three procedures; (i) creation of the pixel feature histogram for the image $\boldsymbol{A}$, (ii) creation of the pixel feature histogram for the image $\boldsymbol{B}$, and (iii) search of the infrequent bins from the histograms. The first two procedures require $O(N^2)$ computations and the last requires $O(M)$. Thus, $O(N^2) + O(M)$

computations are required for the first step in total.

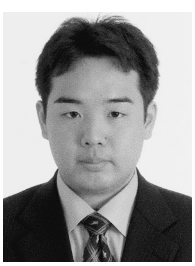
It can be shown that this computational complexity can be reduced to $O(N^2)$ by combining the last two procedures. The following is the improved algorithm:

1. Create the complete pixel feature histogram for the image $\boldsymbol{A}$.
2. Prepare a dynamic list L having no element.
3. For each pixel on the image $\boldsymbol{B}$, check the occurrence of its pixel feature value m in $\boldsymbol{A}$ using the pixel feature histogram. If the bin for m equals to 1 (i.e., the occurrence equals to 1), maintain L as follows:

    a. If the bin for m is not linked with any element in L, add a new element to L and make a link to the element from the bin.
    b. Otherwise, i.e., if the bin is already linked with an element of L, remove the element from L and mark the bin never to make a link from the bin again.

The resulting dynamic list L contains all the infrequent pixel features, which represent corresponding pixel pairs between $\boldsymbol{A}$ and $\boldsymbol{B}$. Steps 1, 2, and 3 of this algorithm require $O(N^2)$, $O(1)$, and $O(N^2)$ computations, respectively and therefore the algorithm requires $O(N^2)$ computations in total.

Through an experiment using the image pairs of Sect. 5.1, it was shown that the computation steps of the improved algorithm was about 1/20∼1/100 of the original algorithm. While the computations for the first step were far less than that for the second step as shown in Fig. 8 even in the original algorithm, its further reduction by the above improvement will emphasize the practical usefulness of the proposed technique.



Akihiro Mori received B.E. and M.E. degrees from Kyushu University in 2005 and 2007, respectively. From 2007, he has joined Toshiba Corporation. His research interests include pattern matching, gesture recognition, and image processing.

Seiichi Uchida received B.E., M.E., and Dr. Eng. degrees from Kyushu University in 1990, 1992 and 1999, respectively. From 1992 to 1996, he joined SECOM Co., Ltd., Tokyo, Japan where he worked on speech processing. Currently, he is a professor at Faculty of Information Science and Electrical Engineering, Kyushu University. His research interests include pattern recognition and image processing. Dr. Uchida is a member of IEEE and IPSJ.