CrossMark

Advanced digital image stabilization using similarity-constrained optimization

Dong Sung Pae[1] · Chi Gun An[2] · Tae Koo Kang[3] · Myo Taeg Lim[1] (ID)

Abstract

As many people have portable video devices such as cameras on cell phones and camcorders, image stabilization technique is a crucial and challenging task in computer vision applications, and many image stabilization techniques have been researched over many years. We propose a digital image stabilization method that only uses a software algorithm without additional hardware devices. Furthermore, a novel digital image stabilization method composed of three steps that use similarity-constrained nonlinear optimizer is introduced and applied to many unstabilized videos. First, a feature detection technique called moment-based speeded-up robust features (MSURF) is utilized to obtain the transformation matrix. Second, the k-means clustering algorithm is used to detect and remove some of the outliers that cause residual errors during feature matching. Third, the transformation matrix is optimized using nonlinear optimization algorithms to maintain the similarity of the transformation matrix. The experimental results prove that the proposed algorithm provides accurate image stabilization performance.

✉ Tae Koo Kang
  tkkang@smu.ac.kr

✉ Myo Taeg Lim
  mlim@korea.ac.kr

  Dong Sung Pae
  paeds915@korea.ac.kr

  Chi Gun An
  chigun.an@samsung.com

[1] School of Electrical Engineering, Korea University, Seoul, Republic of Korea

[2] Mechatronics R&D Center of Samsung electronics Co., Hwasung-si, Republic of Korea

[3] Department of Human Intelligence and Robot Engineering, Sangmyung University, Cheonan, Republic of Korea

⌓ Springer

1 Introduction

With the development of portable video devices such as cameras on mobile phones, camcorders, and action cams, many people have personal video imaging devices, and it has become increasingly popular to utilize the video imaging device to share images or personal broadcasts. Research is also carried out to recognize this video image [19]. However, such personal video imaging devices often have a problem in that the image shakes due to movement of the user and camera. To solve the problem, many researchers have developed image stabilization techniques divided into four types: 1) mechanical, 2) optical, 3) electronic, and 4) digital [5].

The mechanical, optical, and electronic image stabilization techniques can obtain good stabilization results, but image stabilization is only possible by additional mechanical compensation. In addition, mechanical compensation can have disadvantages such as an increase in the size of the imaging device and a cost increase due to the purchase of an additional device. Therefore, there has been a lot of research on the digital image stabilization technique using only software that consists of three stages: 1) motion estimation [7], 2) motion compensation [17], and 3) motion inpainting [15]. Since all three stages of digital image stabilization have value as a research field, we focus on the motion estimation stage to estimate camera motion.

Currently, there are two research methods for motion estimation. The first method is intensity-based estimation using changes in color information value (intensity) of the image frame [2, 14, 22, 26]. When the input image is cut out frame by frame, the time between each frame represents a very short time (30 fps standard, which is about 33 ms). Therefore, a pixel at one point does not change much in a short time, and the camera motion can be estimated as a result of estimating the color information of the pixel. This method has the advantage of using only a simple operation of the image frame pixels and thus has a short computational time. However, it has disadvantages such as illumination change, blurred image, and so on. The second method is feature point-based estimation whereby feature points such as edges and corners that can be seen in successive frames are extracted [3, 4, 8, 9, 12, 13, 16, 18, 21, 24, 25]. This method has the advantage of robust and accurate motion estimation for various environmental factors because it extracts feature points that are not affected by the surrounding video imaging environment. Because of these characteristics, it is used in many areas [23]. However, it has a disadvantage in that the computational time can be much longer compared with the color information-based method. In addition, a method of using the relationship between successive images was also proposed [20].

In general, the motion estimation stage assesses the motion of the camera in three steps: 1) local motion estimation, 2) accuracy improvement, and 3) global motion estimation. If the motion estimation is performed in three stages, accuracy improvement in separating the local motion and the global motion may take a long time or having to repeat the image stabilization may occur because of RANSAC (RANdom SAmple Consensus) in the accuracy improvement stage. Therefore, one-step motion estimation techniques with a simple framework have been studied [13].

In this paper, we propose a method to minimize the motion estimation to a one-step process and eliminate the problems that may occur in the accuracy improvement stage by using the k-means clustering algorithm. Finally, we propose an algorithm that minimizes the residual error of feature matching by optimizing the transformation matrix using the similarity constraint-based optimization algorithm.

The composition of the paper is as follows. In Section 2, the proposed algorithm is explained in detail, while in Section 3, experimental results are shown, and in Section 4, the contributions of this research are described.

2 Digital image stabilization using similarity-constrained optimization

2.1 System overview

In this chapter, we propose an advanced digital image stabilization method based on similarity-constrained optimization, as shown in Fig. 1. In order to improve image stabilization accuracy, feature points were extracted using the moment-based speeded-up robust features (MSURF) algorithm [10], which is a modified version of the conventional SURF algorithm using Gaussian-Hermite moments [11]. MSURF improves robustness of the image features and provides better matching accuracy.

Conventional digital image stabilization techniques use the RANSAC algorithm in the motion estimation stage. Because it uses a random function, it has disadvantages in that different results can be obtained every time an experiment is carried out and the calculation time for the RANSAC algorithm is long. Therefore, we propose a k-means clustering algorithm and a similarity constraint-based optimization technique to solve this problem.
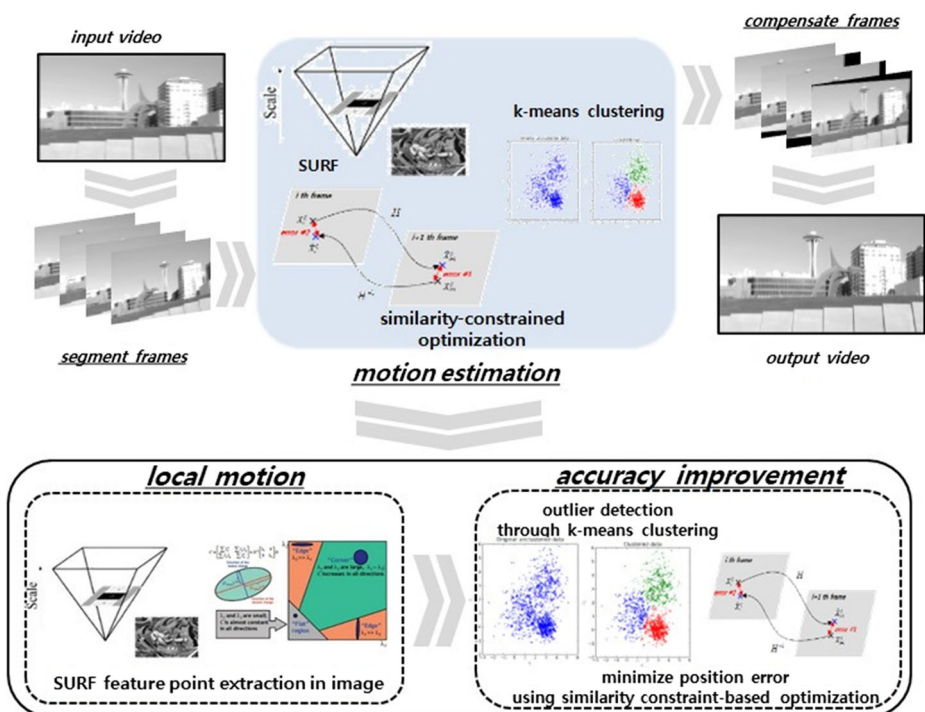

Fig. 1 Architecture of the proposed method

2.2 MSURF-based feature point detection

The first step in digital image stabilization is to divide the input image into several frames to obtain transformation matrix T representing the movement between the divided frames. In this paper, the MSURF algorithm [10] is applied for more accurate matching since it has better feature matching performance than SURF [1], which is similar to the scale-invariant feature transform (SIFT) algorithm [18] but with faster execution. Figure 2 shows an overview of the MSURF algorithm applied to an input image. While the conventional SURF algorithms used the Hessian matrix, MSURF uses the modified discrete Gaussian-Hermite moment (MDGHM) matrix for position $\mathbf{x} = (i, j)$ of each feature point as follows:

$$\mathbf{M}(\mathbf{x}, \sigma) = \begin{bmatrix} \hat{\eta}_{p,0}(\mathbf{x}, \sigma) & \hat{\eta}_{p,q}(\mathbf{x}, \sigma) \\ \hat{\eta}_{p,q}(\mathbf{x}, \sigma) & \hat{\eta}_{0,q}(\mathbf{x}, \sigma) \end{bmatrix}, \tag{1}$$

where $\hat{\eta}_{p,q}(\mathbf{x}, \sigma)$ is the MDGHM [11] at feature position $\mathbf{x}$ on input image I, and is defined as

$$\hat{\eta}_{p,q}(i, j, \sigma) = \frac{4}{(M-1)(N-1)} \sum_{u=1}^{M} \sum_{v=1}^{N} I\left(u+i-{}^{M}/_{2}, v+j-{}^{N}/_{2}\right) \hat{H}_p(x, \sigma) \hat{H}_q(y, \sigma), \tag{2}$$

$$\begin{cases} \hat{H}_p(x, \sigma) = \frac{2}{M-1} \frac{1}{\sqrt{2^p p! \sqrt{\pi} \sigma}} \exp\left(-x^2/2\sigma^2\right) H_p(x/\sigma), \\ \hat{H}_q(y, \sigma) = \frac{2}{N-1} \frac{1}{\sqrt{2^q q! \sqrt{\pi} \sigma}} \exp\left(-y^2/2\sigma^2\right) H_q(y/\sigma), \end{cases} \tag{3}$$

$$\begin{cases} H_p(x/\sigma) = (-1)^p \exp\left(\frac{x^2}{\sigma^2}\right) \frac{d^p}{dt^p} \exp\left(\frac{-x^2}{\sigma^2}\right), \\ H_q(y/\sigma) = (-1)^q \exp\left(\frac{y^2}{\sigma^2}\right) \frac{d^q}{dt^q} \exp\left(\frac{-y^2}{\sigma^2}\right). \end{cases} \tag{4}$$
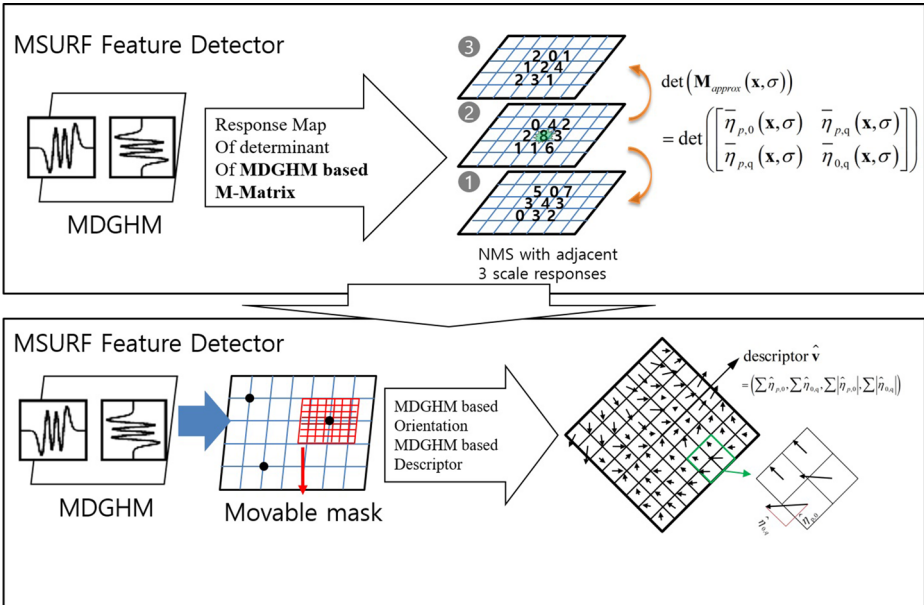

Fig. 2 Overview of MSURF algorithm

In (2), $M \times N$, (u, v), and (x, y) represent the size and position of the mask at feature position $\mathbf{x}$ on input image I. By using (1) to (4), the MSURF descriptor $\hat{\mathbf{v}}$ is generated as

$$descriptor\,\hat{\mathbf{v}} = \left(\sum \hat{\eta}_{p,0}, \sum \hat{\eta}_{0,q}, \sum \left| \hat{\eta}_{p,0} \right|, \sum \left| \hat{\eta}_{0,q} \right| \right). \tag{5}$$

As a result, robust feature points can be extracted using (5). Figure 3 illustrates the example input image to which to apply the MSURF algorithm. The center of the green circle in the figure represents the feature point of the input image frame, and the size of the circle represents the strength of the feature point.

Using the MSURF algorithm, the feature point matching between consecutive image frames can be confirmed. Figure 4 shows the distribution of matching points between consecutive image frames.

Since the location of feature points can be found by matching feature points between successive frames, the final transformation matrix T can be obtained using the similarity transformation matrix to maintain and compensate for the size and direction of structures in the image frames.

2.3 Outlier elimination by k-means clustering

When the feature point matching using the MSURF algorithm is performed, matching with outliers may occur. Since this can lead to ultimately calculating the wrong transformation matrix, outliers must be removed before the final transformation matrix can be obtained. Conventional image stabilization algorithms remove this problem using the RANSAC algorithm. However, since RANSAC can cause problems such as repeatability of the algorithm results and long computational time, other researches to replace it are now proceeding.

In this study, we applied the k-means clustering algorithm to solve this problem. After the feature points obtained through the MSURF algorithm are divided into k clusters, each cluster has one centroid. If the transformation matrix T is obtained by using these center points, the effect of the RANSAC algorithm can be achieved because the influence caused by the outliers is minimized. In addition, it is possible to solve the problem of biasing the transformation matrix where the feature points are concentrated in a specific region within the image frame. The local bias problem is an important issue that needs to be solved before calculating the final transformation matrix because if not carried out correctly, it can emphasize the transformation of a specific region and so calculate an incorrect transformation matrix. Figure 5 shows the matching using the center point of each cluster.

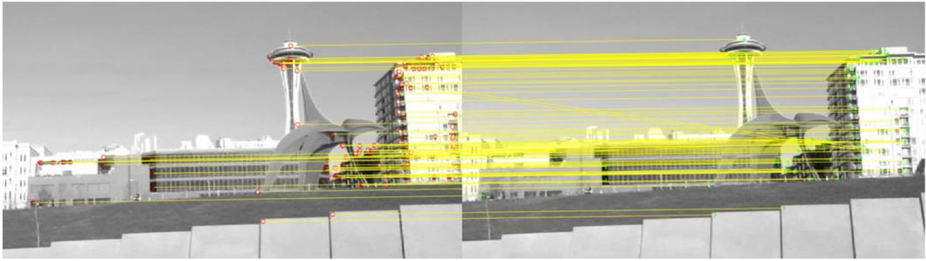**Fig. 3** Example image for applying MSURF algorithm

Fig. 4 Feature matching between two consecutive images

In the k-means clustering algorithm, to set the appropriate K is very important and proceeds by applying.

$$K \approx \sqrt{n/2}, \tag{6}$$

where n is the total number of data items. Since the number of feature-points in an image is usually several tens to several hundreds and the K value is determined using (6), it is impossible to solve the problem of local bias due to too much clustering.

In this study, we determine K value using elbow method. As K value increases, it calculates sum of squared errors (SSE) of k-mean cluster. SSE value is decreasing rapidly when K value is lower than real cluster, whereas decreasing slowly when K value is higher than real clusters. It determines K value in which the decreasing rate become changes. Figure 6 shows the experimental results of a comparison of inter-frame transformation fidelity (ITF) performance results according to the K values for the 100th and 101st frames.

The minimum value of K started at the reference value 5, which was set at the minimum value when the image frame was divided, and it can be seen that the ITF value increased as the K value increased, and then decreased again. This result was slightly different for each image frame but was fixed at 20, which was the K value showing the most stable performance.

2.4 Similarity-constrained optimization

Since the k-means clustering algorithm does not use the random function found in RANSAC, similar results were obtained in several experiments on the same image. Figure 7 shows a comparison of the results of outlier elimination for 100 trials between RANSAC and the k-means clustering-based method for the same frame. Although both methods were able
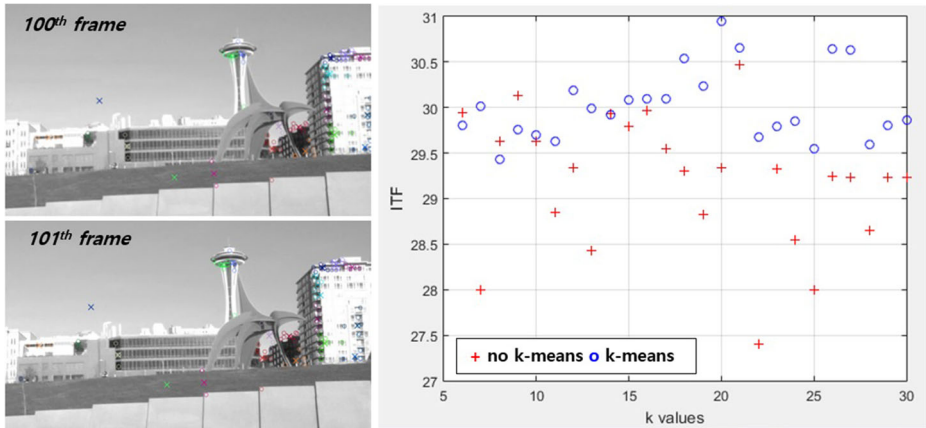

Fig. 5 Matching using each cluster center point

Fig. 6 ITF results according to K value

to produce similar mean results, the standard deviation (SD) of the k-means clustering-based method was about 57% lower than RANSAC as shown in Table 1.

The feature points between the previous frame and the next frame can be predicted by using the transformation matrix T representing the inter-frame conversion relation. We denote the ith feature point on the arbitrary frame t by $\mathbf{x}_t^i$. Let $\mathbf{x}_t^i$ moves to $\mathbf{x}_{t+1}^i$ through the transformation matrix T. In this feature matching process, the type of error which may occur is type 2, as shown in Fig. 8.

In Fig. 8, error #1 occurs because the corresponding feature point is actually located at the $\mathbf{x}_{t+1}^i$ on the frame $(t + 1)$. On the other hand, if the $\mathbf{x}_{t+1}^i$ is moved to the feature point on the frame t through the inverse transformation matrix T^{-1}, it moves to the $\mathbf{x}_t^i$ and the error #2 occurs. These errors are defined as a residual error e_r as shown in (7), and the components of the transformation matrix T are optimized in such a direction as to minimize the residual errors.

$$e_r = \sum_{i=1}^{n} \left\{ \left[T \times \mathbf{x}_t^i - \mathbf{x}_{t+1}^i \right]^2 + \left[T^{-1} \times \mathbf{x}_{t+1}^i - \mathbf{x}_t^i \right]^2 \right\}. \tag{7}$$

(a)   (b)
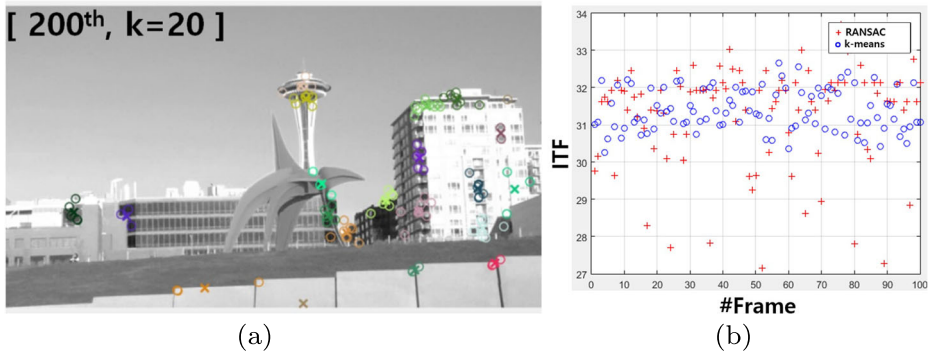
Fig. 7 Comparison for matching accuracy of features for RANSAC and K-mean clustering-based method: **a** an example image of test image sequences, **b** result data for ITF metric

Table 1 ITF Result data comparison with RANSAC and K-means

Method	ITF(mean)	ITF(SD)
RANSAC	31.29	1.3442
K-means	31.29	0.5733

The transformation matrix T consists of a 2×2 rotation matrix and a 2×1 translation component. Moreover, it is divided into similarity or affine transformation matrices according to the characteristics of the rotation matrix component.

The similarity transformation matrix has the property that the scale of the 2×2 rotation matrix component changes constantly, and so it has an advantage in that the size and direction of the structure in the image can be kept constant. On the other hand, if an affine transformation matrix is used, the degrees of freedom to compensate for the whole image region are higher than for the similarity transformation matrix because the 2×2 component has the property of varying independently. However, if the optimization algorithm is repeatedly applied to minimize the residual error, the size and direction of the structures in the image are distorted due to the high degrees of freedom. Figure 9 shows the result image using either the similarity or affine matrix as the transformation matrix.

If the similarity transformation matrix is used as in Fig. 9a, the size and direction of the structure in the image will be maintained even if the optimization algorithm is executed several times. However, if the affine transformation matrix is used as in Fig. 9b, there is a problem in that the size and direction of the structures in the image become distorted. Therefore, if the optimization algorithm is used in the digital image stabilization technique, it is advantageous to use the constraint that can maintain the similarity characteristic.

To use the similarity transformation property defined by nonlinear equations as a constraint condition, we use the Lagrange multiplier λ, which is a representative method of using constraint equations as a constraint condition for transformation matrix T and is defined as

$$\lambda \left(t_{11}{}^2 + t_{21}{}^2 - t_{12}{}^2 - t_{22}{}^2 \right), \text{ for } T = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ 0 & 0 & 1 \end{bmatrix}. \tag{8}$$
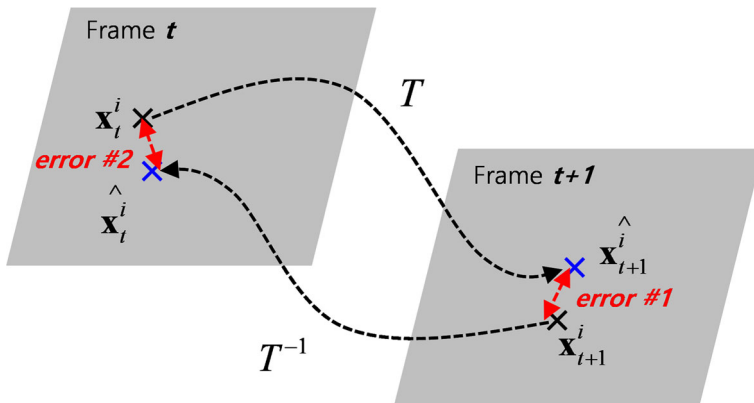

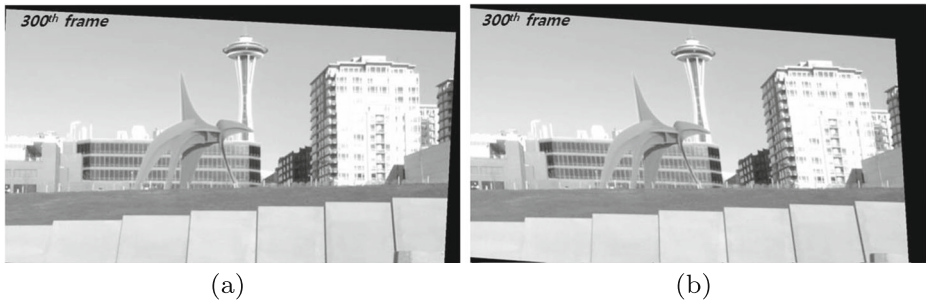**Fig. 8** Errors of the feature matching

Fig. 9 Comparison of stabilization results: **a** similarity matrix, and **b** affine matrix

The similarity transformation matrix T has the nonlinear characteristic of $t_{11}^2 + t_{21}^2 - t_{12}^2 - t_{22}^2 = 0$, thus if a Lagrange multiplier is used as the constraint condition expression, the result can be kept close to 0 at all times. Finally, from (7) and (8), we can define the Lagrangian to reduce the residual error, which can be expressed as

$$L\left(\mathbf{x}_t^i, \lambda\right) = e_r + \lambda \left(t_{11}^2 + t_{21}^2 - t_{12}^2 - t_{22}^2\right).$$

(9)

The objective function e_r in (9) minimizes the positional error for accurate feature point matching between the previous frame and the next frame, and the latter part of (9) is the constraint to maintain the characteristics of the similarity transformation matrix using the Lagrange multiplier. Therefore, we can solve (9) as the similarity-constrained optimization method. Since (9) has several nonlinear equations, we use sequential quadratic programming (SQP), which is a representative technique of nonlinear optimization.

The selection of the initial starting point in the optimization algorithm is a very important issue and in the proposed method, it starts as near as possible to the global minimum because we initially use the transformation matrix with MSURF. In addition, the stopping criteria are also important because the optimization algorithm requires a lot of computational time or divergence. Figure 10 shows the conditions for finding the minimum residual error point in the initial start condition for SQP. The tolerance setting, which is the condition for stopping the algorithm, is set such that the change amount $\Delta\mathbf{x}$ of the transformation matrix T component is 10^{-10} and the change amount Δf of the residual error r is 0.1. Actually, during the operation of the algorithm, since the change amount of the transformation matrix T component and the change amount of the residual error r are both confirmed, the minimum point can be found by repeating the algorithm about several dozen times.

Figure 11 is an example of the optimization results for the residual error applied to the sequential quadratic programming method based on the setting value. In [16], it can be seen that the residual position error value of approximately 4.05 reduced through the initial k-means clustering converges to around 1.6 as the optimization algorithm proceeds. Finally, we can confirm that the variation of the position error converges to a setting threshold value of 0.1 or less, and the minimum point is judged only after being conducted 24 times in total.

Through the similarity-constrained optimization process, we can make the robust transformation matrix T and stabilize the image sequence using it.
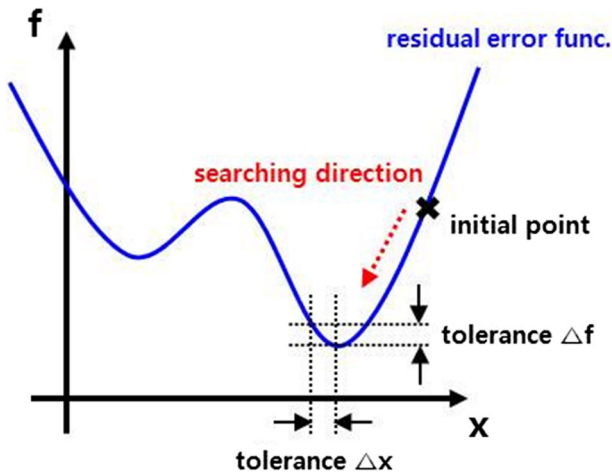
Fig. 10 Conditions for finding points with minimum residual error using SQP

3 Experiments

3.1 Experimental environment

3.1.1 Image dataset and comparison methods

We conducted an experiment to investigate the accuracy of the image stabilization technique. Figure 12 shows example images five different types of image sequences for evaluation. Figure 12a is a typical static scene sequence with no changes in the background, while Fig. 12b is an image sequence taken by the camera moving from left to right with the background; since the camera and the background move at the same time, the moving path of the camera can be canceled, and the images only reflect changes according to the photographing angle. Figure 12c includes a complex background having an excessive amount of feature information that can be a problem in image stabilization techniques using feature points or subspace detection techniques, while Fig. 12d is a drone hovering image sequence often occurring in real life. Finally, Fig. 12e is a dynamic image sequence taken under the condition that the camera and foreground objects in the image sequence move simultaneously.

In order to compare the performance of the image stabilization method, we selected four methods: original data without stabilization, methods from [13] and [6], and the proposed method; we evaluated their performance for the five image sequences shown in Fig. 12.

3.1.2 Evaluation

To compare image stabilization performance, we use ITF to evaluate the relative stabilization results between the image frames, which can be defined as

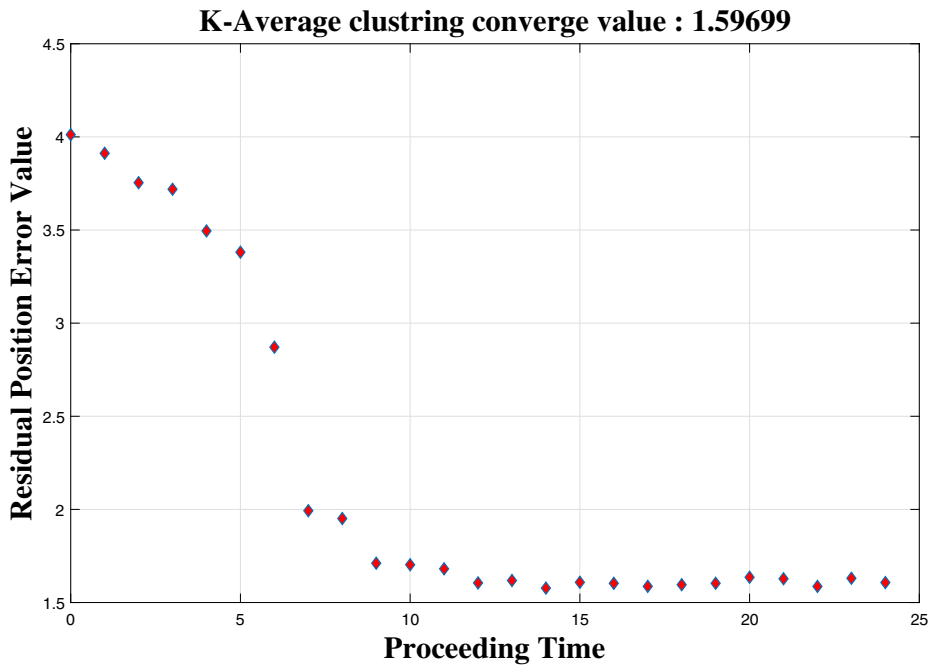$$ITF = \frac{1}{N_F - 1} \sum_{k=1}^{N_F - 1} PSNR(k), \tag{10}$$

Fig. 11 Residual error optimization using SQP

where N_F represents the total number of frames, and the peak signal-to-noise ratio (PSNR) represents the maximum signal-to-noise ratio; this is derived as

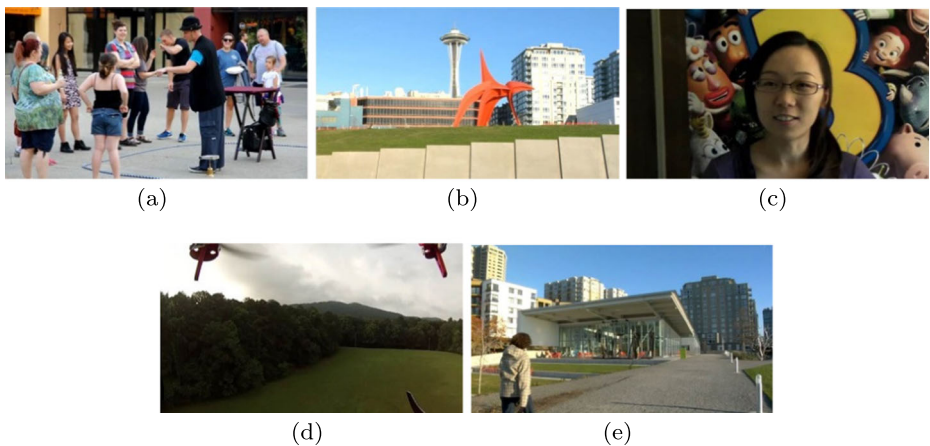$$PSNR(k) = 10 \log \frac{I_{MAX}}{MSE(k)}, \tag{11}$$

Fig. 12 Example images of five image sequence conditions for evaluation: **a** fixed camera, **b** horizontal moving camera with fixed objects, **c** fixed complex background, **d** forward moving camera with momentary vertical and horizontal motion, **e** forward moving camera with forwarding moving objects
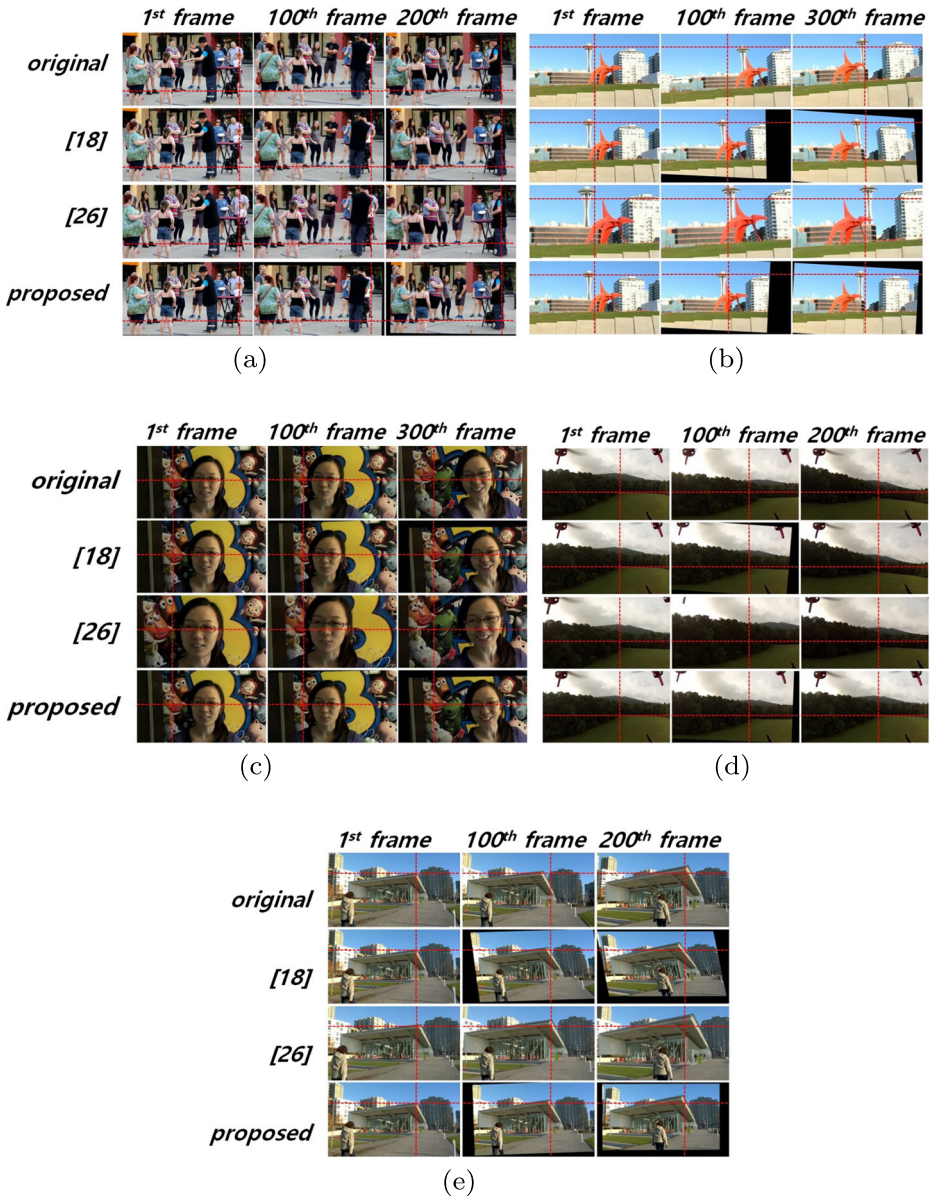
Fig. 13 Experimental results for five image sequences

in which

$$MSE(k) = \frac{1}{N} \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} (I_k(x, y) - I_{k+1}(x, y))^2, \text{ for } (x, y) \in S_{nb}, \quad (12)$$

where w, h, and I_{MAX} represent image width, height, and peak intensity values in input image I, respectively; and N and S_{nb} mean the non-blank region and the number of non-

blank pixel for input image I, respectively. The value of $I_{(}x$, $y)$ used in (12) represents the intensity value at position (x, y), and the difference in brightness value at the same position between consecutive frames is defined as an error.

3.1.3 Experimental results

We evaluated accuracy for the five cases. In the cases of (a) and (c), the position of the camera was fixed, thus they can be considered as static scenes whereas in the cases of (b), (d), and (e), the position of camera was continuously changed, and so they can be considered as dynamic scenes. Figure 13 illustrates the experimental results frame by frame for five image sequences and Table 2 shows the mean and SD for the ITF values. In Table 2, the higher the mean ITF value is, the higher the accuracy of the performed image stabilization, while the lower the ITF SD is, the higher precision of image stabilization has been performed.

As shown in Table 2, for most of the image dataset, the mean and SD for the ITF values obtained using the proposed method were better than those using the existing ones. In the case of (a), the proposed method showed a lower mean ITF value of 0.36 dB for [13] and 0.04 dB for [6], respectively. It stems from that images in (a) include relatively complex objects, and it reduce the stabilization accuracy, as influencing determination of the number of cluster in outlier removing step using the k-means algorithm. According to the mean ITF values, the proposed method exhibited enhancements over [13] and [6] of 0.24 dB and 0.27 dB in the case of (b), 0.29 dB and 2.06 dB in the case of (c), 11.47 dB and 2.86 dB in the case of (d), and 1.4 dB and 1.38 dB in the case of (e), respectively. The evaluation results demonstrate that the proposed method was superior to the existing methods under static and dynamic scene conditions in terms of stabilization accuracy.

Figure 13 illustrates the resulting images of the ground truth comparison with respect to the five different image type sequences. The left, middle, and right columns represent the 1st 100th, and 200th frames, respectively, in each image sequence. Moreover, the first to fourth rows demonstrate the stabilized images using the original image sequence, [6, 13], and the proposed method, respectively. To compare the methods easily, base lines are marked with red dashed lines in each image.

As shown in Fig. 13a, most methods except the original image sequence showed similar stabilization performances, which was stable for this case. In the case of Fig. 13b, we can see that the proposed method always showed the best performance because it always

Table 2 Mean and SD for the ITF values

Condition	ITF Value	Original	[13]	[6]	Proposed
(a)	Mean	20.59	28.91	28.59	28.55
	SD	8.12	5.00	4.73	4.77
(b)	Mean	18.84	31.91	31.88	32.15
	SD	3.52	3.63	11.77	2.70
(c)	Mean	21.61	28.36	26.59	28.65
	SD	13.51	17.84	17.23	16.77
(d)	Mean	25.16	20.76	29.37	32.23
	SD	8.66	212.15	18.62	24.97
(e)	Mean	19.48	28.10	28.12	29.50
	SD	3.62	5.55	6.49	3.75

Table 3 Computation time comparison

	Image Size	[13]	[6]	Proposed
FPS	720 × 1280	20.44	13.95	13.39
	360 × 640	29.12	23.31	22.54

pointed to a similar reference point. The results using [6] did not show good performance since the method is optimized for dynamic image sequence experiments with many camera movements. Moreover, in [13], as the optimization algorithm proceeded, the degrees of freedom of the transformation matrix became high resulting in the size and direction of the structures in the image becoming distorted. On the other hand, since the proposed method maintained similarity as the optimization algorithm proceeded, the size and direction of the structures in the image were kept constant.

In Fig. 13c, [6] showed the worst ITF performance compared to the other techniques, while [13] and the proposed method gave similar good mean and SD values. However, when comparing their values in Table 2, we can confirm that the proposed method has the best performance in terms of ITF value. Figure 13d shows the ground truth comparison of the drone image sequence. Compared with the other techniques, the proposed method showed the most stable results and obtained the best ITF mean value. However, it can be seen that the ITF SD greatly varied as the image sequence progressed because drones do not maintain consistent performance and many rolls in the x-axis direction occur when hovering. With [13], it is evident that the algorithm was no longer able to proceed due to image distortion from the 200th frame onward, while with [6], the overall ITF performance was consistent but the mean ITF value was lower than the proposed method (see Table 2).

The experimental results in Fig. 13e show a zooming effect in which the size of the background in the image changes because the camera moved forward. Since the proposed method compensates for camera shake by counteracting the shaking of the whole image, the size of the structure in the image became smaller due to the zoom effect, which illustrates a limitation of the proposed method that should be considered in future research. Furthermore, although [6, 13], and the proposed method showed similar ITF mean values, Table 2 shows that the proposed method obtained the best ITF mean and SD performance.

We also evaluated computation time for two cases; image size : 720 × 1280 and 360 × 640, because an algorithm based on a feature point depends on the size of the image and the computation time. The computation speeds (fps) of conventional approaches and the proposed method are compared in Table 3. The experiments were done with a AMD Ryzen 7 1700 8 Core 3.0GHz CPU with 16GB RAM and GTX1080 GPU. It can be shown that real-time implementation of image processing algorithms is possible only when fps results from 20 or higher are obtained. Although the proposed algorithm is slower than color information-based method, the result of the computation speed shows that proposed algorithm can be applied in real-time for small images.

4 Conclusions

In this paper, we propose an advanced image stabilization technique consisting of three steps. First, local motion features are detected using the MSURF algorithm. Second, outliers are removed from local motion using k-means clustering to remove outliers when detecting global motion, which up to now, has been achieved using the RANSAC algorithm. However,

the latter method has a problem in that the result is different each time the algorithm is executed because it uses the random function several times to determine the outliers. However, k-means clustering applied to remove outliers can guarantee the repeatability of the image stabilization results because it does not use the random function in the calculation process. Third, the calculated transformation matrix is optimized using the similarity-constrained optimization method. Digital image stabilization techniques use a similarity matrix, an affine matrix, or a projective matrix as a transformation matrix. Since the affine matrix or the projective matrix has high degrees of freedom of the transformation matrix components, this can result in a problem whereby the size and direction of the structures in the image may become distorted as the optimization algorithm proceeds. Therefore, constraints that maintain the similarity characteristics of the transformation matrix are needed for the optimization algorithm to proceed, and the similarity-constrained optimization method is able to do this during optimization. From the experimental results, we confirmed the superiority of the proposed algorithm over previously reported ones by applying it to various image sequences.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Bay H, Ess A, Tuytelaars T, Gool LV (2008) Speeded-up robust features (surf). Comput Vis Image Underst 110(3):346
2. Bosco A, Bruna A, Battiato S, Bella G, Puglisi G (2008) Digital video stabilization through curve warping techniques. IEEE Trans Consumer Electron 54(2):220
3. Erturk S (2002) Real-time digital image stabilization using kalman filters. Real-Time Imag 8(4):317
4. Favorskaya MN, Jain LC, Buryachenko V (2014) Computer vision in control systems. Springer, Berlin
5. Forsyth DA, Ponce J (2002) Computer vision: a modern approach. Prentice Hall, Upper Saddle River
6. Grundmann M, Kwatra V, Essa I (2011) Auto-directed video stabilization with robust l1 optimal camera paths. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 225–232
7. Hadawale N, Nair S, Ukirde N, Andre SB (2018) Real time implementation of video stabilization. Int J Current Tredns Sci Technol 8(1):182
8. He J, Zhang D, Balzano L, Tao T (2014) Iterative grassmannian optimization for robust image alignment. Image Vis Comput 32(10):800
9. Jeon S, Yoon I, Yang S, Kim B, Kim J (2016) Robust feature detection using particle keypoints and its application to video stabilization in a consumer handheld camera. In: IEEE international conference on consumer electronics (ICCE), pp 217–218
10. Kang T, Choi I, Lim MT (2015) Mdghm-surf: a robust local image descriptor based on modified discrete gaussian-hermite moment. Pattern Recogn 48(3):670
11. Kang T, Zhang H, Kim D, Park G (2012) Enhanced sift descriptor based on modified discrete gaussian–hermite moment. ETRI J 34(4):572
12. Kim SK, Kang SJ, Wang TS, Ko SJ (2013) Feature point classification based global motion estimation for video stabilization. IEEE Trans Consumer Electron 59(1):267
13. Kim SW, Yin S, Yun K, Choi JY (2014) Spatio-temporal weighting in local patches for direct estimation of camera motion in video stabilization. Comput Vis Image Underst 118:71
14. Kir B, Kurt M, Urhan O (2015) Local binary pattern based fast digital image stabilization. IEEE Signal Process Lett 22(3):341

15. Kumar V, Mukherjee J, Mandal SKD (2016) Image inpainting through metric labeling via guided patch mixing. IEEE Trans Image Process 25(11):5212
16. Lee TH, Lee YG, Song BC (2014) Fast 3d video stabilizatoin using roi-based warping. J Vis Commun Image Represent 25(5):943
17. Li D, Wang Z (2017) Video superresolution via motion compensation and deep residual learning. IEEE Trans Comput Imag 3(4):749
18. Lowe D (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91
19. Luo M, Chang X, Li Z, Nie L, Mauptmann AG, Zheng Q (2017) Simple to complex cross-model learning to rank. Comput Vis Image Underst 163:67
20. Ma Z, Chang X, Xu Z, Sebe N, Hauptmann AG (2018) Joint attributes and event analysis for multimedia event detection. IEEE Trans Neural Netw Learn Syst 29(7):2921
21. Okade M, Biswas PK (2014) Improving video stabilization using multi-resolution mser features. IETE J Res 60(5):373
22. Puglisi G, Battiato S (2011) A robust image alignment algorithm for video stabilization purposes. IEEE Trans Circuit Syst Video Technol 21(10):1390
23. Wang S, Chang X, Li X, Long G, Yao L, Sheng QZ (2016) Diagnosis code assignment using sparsity-based disease correlation embedding. IEEE Trans Knowl Data Eng 28(12):3191
24. Xu J, Chang H, Yang S, Wang M (2012) Fast feature-based video stabilization without accumulative global motion estimation. IEEE Trans Consumer Electron 58(3):993
25. Yang J, Schonfeld D, Mohamed M (2009) Robust video stabilization based on particle filter tracking of projected camera motion. IEEE Trans Circuit Syst Video Technol 19(7):945
26. Yeni AA, Erturk S (2005) Fast digital image stabilization using one bit transform based sub-image motion estimation. IEEE Trans Consumer Electron 51(3):917

Dong Sung Pae received the B.S. degree in electrical engineering from Korea University, Seoul, South Korea, in 2013, where he has been working toward the Ph.D. degree with the School of Electrical Engineering since 2013. His current research interests include computer vision, feature extractor, video stabilization, artificial intelligence, and their application to intelligence vehicle systems.

Chi Gun An received the B.S. and M.S. degrees in Department of Mechatronics from Korea University, in 2007 and 2015, repectively. He have been researching about industrial robot's motion control since 2007 in Mechatronics R&D Center of Samsung electronics Co., Hwasung-si, Korea. His current research interests include robotics, computer vision, video stabilization, artificial intelligence, and their application to robot systems.

Tae Koo Kang received his B.S. in Applied Electrical Engineering, M.S. in Visual Image Processing, Ph.D. in Electrical Engineering from the Korea University, Seoul, Republic of Korea, in 2001, 2004, and 2012 respectively. He is now a Research Professor in the School of Electrical Engineering at Korea University. His research interests include computer vision, robotics, artificial intelligence, and machine learning.

Myo Taeg Lim received his B.S. and M.S. degrees in Electrical Engineering from Korea University, Korea in 1985 and 1987, respectively. He also received M.S. and Ph.D. degrees in Electrical Engineering from Rutgers University, NJ, USA in 1990 and 1994, respectively.

He was a Senior Research Engineer, Samsung Advanced Institute of Technology and a Professor in the Department of Control and Instrumentation, National Changwon University, Korea. Since 1996, he has been a Professor in the School of Electrical Engineering at Korea University. His research interests include optimal and robust control, vision based motion control, and autonomous mobile robots. He is the author or coauthor of more than 80 journal papers and two books (Optimal Control of Singularly Perturbed Linear Systems and Application: High-Accuracy Techniques, Control Engineering Series, Marcel Dekker, New York, 2001; Optimal Control: Weakly Coupled Systems and Applications, Automation and Control Engineering Series, CRC Press, New York, 2009).

Dr. Lim currently serves as an Editor for International Journal of Control, Automation, and Systems. He is a Fellow of the Institute of Control, Robotics and Systems, and a member of the IEEE and Korea Institute of Electrical Engineers.