ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 165

International Conference on Modeling, Simulation, Optimization and Numerical Techniques (SMONT 2019)

A Novel Joint Optimization Method for Adaptive Hand-held Video Stabilization Based on Spatial-temporal Consistency

Xiao Li[1], Shuai Li[1,2,*], Hong Qin[3] and Aimin Hao[1]

[1]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China

[2]Beihang University Qingdao Research Institute, Qingdao 266000, China

[3]Department of Computer Science, Stony Brook University, New York 11794, USA

*Corresponding author

***Abstract*—Video stabilization for hand-held camera is vital in many high-level video enhancement applications. Although quite a few proposed approaches have achieved remarkable success in recent years, many technical challenges still prevail for video (from hand-held camera) containing wide-range scenes and highly-variable objects. In particular, we are lacking effective and versatile strategies to adaptively handle saliency preservation, parallax diminution, self-adaptive smoothing, cropping area decrement and video completeness in a consistent spatial-temporal fashion. To ameliorate, this paper develops a novel, joint optimization method to successively respect spatial structure consistency and temporal feature constraints. Our aim is to devise a new adaptive video stabilization technique by resorting to new modeling strategies. This paper's key originality is hinged upon joint utility of both self-adaptive intrinsic mode functions (IMFs) based on empirical mode decomposition (EMD) in temporal domain for video signal and mesh-structure constraint enforcement in the spatial domain. As a result, our new approach can optimize camera trajectory of wobbly video, and synchronously fine-tune the camera path based on key features to make the shaking video much closer to the original trend. To validate our joint optimization approach for adaptive video stabilization, we conduct comprehensive experiments on public benchmarks, and make extensive and quantitative evaluations with available state-of-the-art methods as well as popular commercial software. All of our experiments demonstrate the advantages of the joint optimization method in terms of versatility, accuracy, and efficiency.**

Keywords—hand-held Camera; adaptive video stabilization; spatial structure consistency; feature-centric EMD; self-adaptive IMF selection; video extrapolation and interpolation

I. INTRODUCTION AND MOTIVATION

The hand-held devices used by amateurs, such as mobile phones or portable camcorders, tablet PCs and frequently-used cameras, have become popular, however, these videos captured by the hand-held devices tend to be shaky and make viewers uncomfortable, because the devices have very simple stabilization equipments. Video stabilization aims to remove such visible frame-to-frame jitters and shakes in the wobbly video. It is one of the most active research subjects in computer vision, and can benefit many high-level video enhancement applications, such as manual observation, video discrimination, video detection, video tracking and video compression, etc.

Given multiple videos captured with a hand-held device (e.g., a cell-phone or a portable camcorder), most state-of-the-art video stabilization methods either learn a 2D linear motion model by estimating and smoothing a linear transformation (affine or homography) between consecutive frames [6][14][16], or resort to 3D curved camera motion by dealing with the parallax in principle to generate strongly stabilized results [3]. After a long period of evolution, both 2D and 3D approaches have achieved great success. However, to better combine the traditional problems of feature detection, feature registration and camera trajectory analysis into a unified stabilizing framework, some challenges are still not fully resolved, which are summarized as follows.

First, from the perspective of producing a stable parallax-free camera trajectory based on saliency preservation, the parallax caused by non-trivial depth variation in the scene makes the estimating become an ill-posed problem, because different regions may require different trajectories and spatially-variant homographies. Although spatial multidimensional reconstruction can deal with the parallax in principle and generate more stable results, however, multidimensional motion model estimation is far less robust due to various degenerations and frequently ignores the conservation of saliency features, such as rapid rotation, feature tracking failure, camera zooming and motion blur. Therefore, how to simultaneously exploit the spatially-variant homographies to produce a uniform parallax-free homography based on saliency preservation is urgently needed in video stabilization.

Second, from the perspective of smoothing camera motion adaptively, current methods more or less suffer from the following problems. It is difficult to handle more challenging cases (e.g., rapid motion, fast scene transition, large occlusion) by straightforwardly smoothing original camera motion without sufficient self-adaptation, because the camera motion tends to be smoothed excessively/insufficiently under some circumstances. For example, the unstable video tends to be excessively cut or the wobbly video is inclined to be shaking. Thus, considering the optimizing quality of original video, how to design adaptive smoothing model to analyze and smooth the shaking video, is extremely essential for the robust and high-quality result.

Third, from the perspective of cutting down the cropping area of optimized videos, current methods more or less suffer

from the following problems. Lots of techniques pay too much attention to smoothing effect and ignore dropping the cropping area of smoothed video. Thus, considering the preservation of original image contents, it needs an accurate but simple strategy to flexibly preserve the contents of original video, while successfully suppressing both its high frequency jitters and low-frequency bounces.

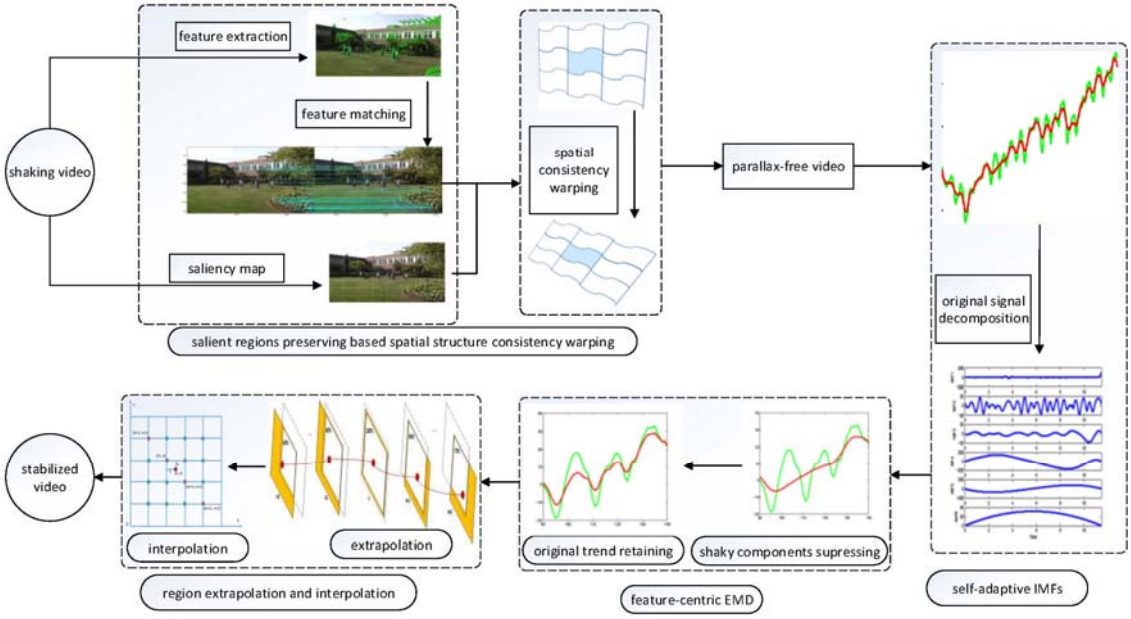


FIGURE I. THE PIPELINE OF OUR FRAMEWORK.

Fourth, from the perspective of the completeness of the stabilized video, although various high-level image interpolation and extrapolation methods have been proved effective, however, without substantially considering the influence of adjacent pixels, some unexpected results may occur. For example, the discontinuity of resampling values may produce significant mosaics and jaggy. Thus, considering the uncertainty of image interpolation and extrapolation, it needs an accurate but simple strategy to flexibly complete the missing part of the original video.

To tackle the aforementioned challenges, we shall concentrate on the self-adaptive video stabilization by taking full advantages of both 2D and 3D methods. Specifically, our salient contributions can be summarized as follows:

- We propose a homography estimation method of spatial structure consistency with preserving salient image regions, which gives rise to the efficient and effective revealing of the intrinsic motion consistence among different regions in wobbly video, while still being able to flexibly build a unified camera motion path.
- We propose a self-adaptive IMF ratios technique based on Empirical Mode Decomposition (EMD), which can make the shaking video be more stable, and also facilitate the analysis and optimization of unstable videos.
- We propose a feature-centric strategy based on Gaussian distribution, which can observably cut down the cropping area of stable videos, while maintaining the stability rate of optimized video.
- We propose an efficient video matching and completing method to extrapolate the lost region and interpolate the overlapping part among the consecutive frames of the original video.

II. RELATED WORKS

A. 2D Video Stabilization Methods

2D video stabilization methods estimate the 2D transformations between consecutive video frames to represent camera motion, and smooth the video camera trajectory over time to generate a stable video. For example, Zhang et al. [29] present a novel formulation of video stabilization in the space of geometric transformations. Dragon et al. [4] and Narayana et al. [22] use two frames based motion segmentation techniques. Gleicher et al. [5] break camera trajectories into segments for individual smoothing. However, these methods tend to ignore the different motions distributed in the different parts of the frame. To solve the problem, we propose a spatial structure consistency homography estimation method.

Besides, Hu et al. [10] presents a fast and effective method based on features to obtain real-time video stabilization for vehicle video recorder system. Aguilar et al. [1] use a combination of geometric transformations and outliers rejection to obtain a robust inter-frame motion estimation, and a Kalman filter based on an ANN learned model of the MAV. Grundmann et al. [8] apply an elegant L1-norm optimization for camera path smoothing. Grundmann et al. [7] further adopt a homography-array-based motion model to deal with the rolling shutter effects. Liu et al. [19] use a spatially-variant model to represent the motion between video frames, and design an appropriate

smoothing technique for this model. Liu et al. [20] propose a novel motion model, which is a specific kind of optical flow by enforcing strong spatial coherence, to represent the motion between neighboring video frames for stabilization. Such methods frequently warp all the objects of the frames in the video to distort the important objects. Based on the above reasons, our method puts forwards the salient image regions protection based re-targeting.

Although conventional 2D video stabilization can significantly reduce the camera shakes, it cannot simulate the ideal camera path. Since 2D method has no knowledge of the real 3D input trajectory, and it is unable to infer the 3D camera trajectory and the physical distribution in 3D scene.

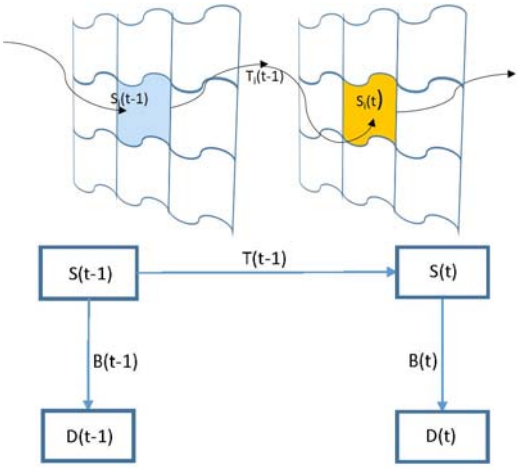


FIGURE II. (A) IN THE I MESH, RELATIONSHIPS AMONG CAMERA POSE SI(T-1) AT FRAME T-1, CAMERA POSE SI(T) AT FRAME T, AND HOMOGRAPHY OF SPATIAL STRUCTURE CONSISTENCY TI(T-1). (B) RELATIONSHIPS AMONG ORIGINAL PATH {S(T)}, SMOOTHED PATH {D(T)}, AND TRANSFORMATIONS {B(T)}.

B. 3D Video Stabilization Methods

3D video stabilization methods can recreate dynamic scene structure from a single source video, and estimate 3D camera motion for stabilization. For example, Liu et al. [18] automatically partition the input video into CDV and DDV segments. Buehler et al. [3] design stabilization algorithms from a projective 3D reconstruction with an un-calibrated camera. Liu et al. [15] develop a 3D stabilization system with content-preserving warping. Zhou et al. [30] further introduce plane constraints for video stabilization. Liu et al. [17] use a depth camera for robust stabilization. Generally speaking, these 3D methods produce good results, especially on scenes with non-trivial depth changes. However, 3D reconstruction (or a depth sensor) is demanded. Some recent methods relax this requirement by exploiting partial 3D information embedded in long feature tracks. Moreover, the methods demand too much manual intervention to affect the executive efficiency. We propose the self-adaptive analysis and optimization strategy to ameliorate the faultiness of the existing methods.

Besides, Liu et al. [16] smooth a set of input 2D motion trajectories and resemble visually-plausible views of the imaged scene in some subspaces for stabilization. Wang et al. [28] represent each trajectory as a Bezier curve, maintain the spatial relations between trajectories formulate stabilization as a spatial-temporal optimization problem, which avoids visual distortion. Goldstein and Fattal [6] use projective scene reconstruction for jitter video sequences to alleviate the strain on long feature tracks. Smith et al. [26] employ a space-time optimization method to stabilize a light-field camera. Liu et al. [19] extend the subspace method to deal with stereoscopic videos. However, the above methods usually cut off too much effective information of the original frames. We propose an efficient video matching and completing method to extrapolate the lost region and interpolate the overlapping part among the consecutive frames of the original video. In addition, the 3D methods can produce better results compared with other methods. However, 3D reconstruction is fragile, especially for consumer-level videos.

III. METHOD OVERVIEW

Our method is more flexible and adaptive, since we regard the camera bundled-trajectories motion as the noisy signal, and adaptively suppress the noise without manual intervention. In addition, our method can achieve stabilization effects similar to the 3D methods while retaining the efficiency and robustness of 2D methods over challenging videos. As shown in Figure I, our method could be divided into four components, including salient regions preserving based homography estimation of spatial structure consistency, self-adaptive IMFs, feature-centric EMD, region extrapolation based on the adaptive temporal range and interpolation based on the cubic spline interpolation.

Salient regions preserving based homography estimation of spatial structure consistency is content- aware, which can preserve salient and visually prominent regions. It warps frames in the wobbly video based on a saliency map. Each frame is uniformly divided into multiple grids. The local homography in each grid is estimated to build spatially mesh-wise inconsistent camera paths. The method can eliminate the parallax between different grids in the same video frames while preserving salient regions.

Self-adaptive IMFs can read all the frames in the wobbly video and find their SIFT features. And the geometric transformation can be estimated from the matching point pairs. The original signal is decomposed into a finite and some small number of components. Finally, we can calculate the optimizing ratio of the IMF using CVX.

Feature-centric EMD aims to keep the centric of the feature of EMD. Our method brings in the weighted Gaussian distribution to make the new path keep the trend of the original path while suppressing the jitters. Region extrapolation and interpolation is carried out with the adaptive temporal range and the cubic spline interpolation, which adaptively chooses the temporal range and solves the gray value of the unknown pixel by the weighted interpolation of its nearby sixteen gray scale values to drastically lower the cropping area of the stabilized video.

IV. SALIENCY PRESERVING BASED SPATIAL STRUCTURE CONSISTENCY WARPING

For the shaky handhold video, spatial inconsistency in different regions of each frame may bring out lots of distortions and artifacts when stabilizing the wobbly video. It is meaningful

to reduce the influences of spatially mesh-wise inconsistency in different regions. To this end, video re-targeting based on spatial structure consistency is utilized while preserving salient and visually prominent regions of each frame in the videos. Inspired by content-aware image and video retargeting techniques [25], to respect salient regions, we conduct homography estimation of spatial structure consistency.

FIGURE III. (A) THE LEFT PICTURE SHOWS THE RESULT WITH TRADITIONAL WARPING METHOD. MARGINAL AREA OF LEFT PICTURE IS JAGGED AND ITS CONTENT IS TORTILE. (B) THE RIGHT PICTURE SHOWS THE RESULT WITH OUR SALIENCY WARPING METHOD. THE ENTIRE PICTURE IS REGULAR AND ITS CONTENT IS NOT TWISTED.

A. SIFT Based Spatial Structure Homography Construction

Using the method proposed by Lowe et al. [21], we frame-wisely extract the SIFT features from the wobbly video. From the SIFT features, we can get the descriptor component and the location component. One SIFT match is accepted only if its Euclidean distance is less than $distRatio$ times the distance to the second closest match. Generally, we set $distRatio$ as 0.1. The Euclidean distance from the point s to t is the length of the line segment connecting them. Through the above process, the $M \times 2$ matrices of [x,y] coordinates can be obtained. Outliers in matched points of two frames are excluded by using M-estimator SAmple Consensus (MSAC) algorithm. As shown in the second image of the first rectangle of Figure I, the geometric transformation maps the inliers in matched points of the left frame to the inliers in those of the right frame. Using geometric transformation algorithm in [24], we could compute and denote the geometric transformation with the 3×3 matrix $T_t = \begin{bmatrix} R_t & O_t \\ 0 & 1 \end{bmatrix}$. Here R_t and O_t are the 2×2 rotation matrix and 2×1 translation vectors, representing the camera motion orientation and position in the global coordinate system respectively. As shown in Figure II, the relative camera motion at time t can be represented by a 2D Euclidean transformation T_t, satisfying $S_t = S_{t-1}T_{t-1}$. We denote $S_t = \begin{bmatrix} \tilde{R}_t & \tilde{O}_t \\ 0 & 1 \end{bmatrix}$.

B. Spatial Structure Consistency Optimization

A uniform grid is overlaid over the image with $\hat{N}$ columns and $\hat{M}$ rows. The target is to compute a deformed grid for the resized image. Consistent with common image retargeting methods, the saliency map $\Psi(\hat{x}, \hat{y})$ is used to assign an importance value between 0 and 1 to every pixel of the image. A deformation that preserves the image in the salient zones as much as possible could be computed, and the unavoidable distortion could be concentrated in less important areas. We average the saliency values inside every cell of the grid on the original image, while the saliency vector Ψ_i could be obtained.

Based on the saliency vector, our optimization makes further efforts to bring down the influences of spatially mesh-wise inconsistency, which greatly decrease the parallax. Based on the previous camera path $S_i(t-1)$ and the local homographies $T_i(t-1)$, we can define spatially mesh-wise inconsistent camera paths for the whole video. Let $S_i(t)$ be the camera pose of the grid cell i at frame t. It can be formulated as $S_i(t) = S_i(t-1)T_i(t-1)$. Here taking $S_i(1)$ as the identity matrix, we can derive the next equation as $S_i(t) = \prod_{\hat{t}=1}^{t-1} T_i(\hat{t})$. We uniformly divide the frame into multiple grids. As shown in Figure II (b), $D(t)$ denotes the smoothed path, and $B(t)$ denotes the transformation from the original path $S(t)$ to the smoothed path $D(t)$.

As shown in Figure II(a), each grid has one trajectory, which is denoted by $S_i(t)$. $T_i(t-1)$ denotes the estimated local homographies at the same grid cell i from $S_i(t-1)$ to $S_i(t)$. These camera trajectories of spatial structure consistency could be smoothed by

$$\mathcal{O}(D(t)) = argmin(\Sigma_i(\|D_i(t) - S_i(t)\|^2 + \lambda_t \Psi_i \Sigma_{j \in \Omega(i)} \|D_i(t) - D_j(t)\|^2)). \quad (1)$$

As shown in Eq. (1), $S = \{S(t)\}$ is the original path and $D = \{D(t)\}$ is the optimized path. $\Omega(i)$ represents the eight neighbors of the grid cell i. Data term $\|D_i(t) - S_i(t)\|$ guarantees the new camera path to be close to the original one to reduce cropping and distortion, while $\|D_i(t) - D_j(t)\|$ can keep the current grid cell be consistent with the nearby neighbors. Parameter λ_t is used to balance the above two terms. For the marginal grid cell, we set its value is the same as those of its inexistent neighbors. Namely it can be formulated as $D_j(t) = D_i(t)$ when j is non-existent. This optimization is quadratic and its optimum result can be obtained by solving a large sparse linear system. The above solution is updated by a Jacobi-based iteration [12].

$$D_i^{(\delta+1)} = \frac{S(t)}{1+2\lambda_t} + \Sigma \frac{2\lambda_t \Psi_i}{1+2\lambda_t} D_j^{(\delta)}. \quad (2)$$

In Eq. (2), δ is the iteration index. At initialization, $D^{(0)}(t) = S(t)$. Then we can get the optimized paths $D_i(t)$.

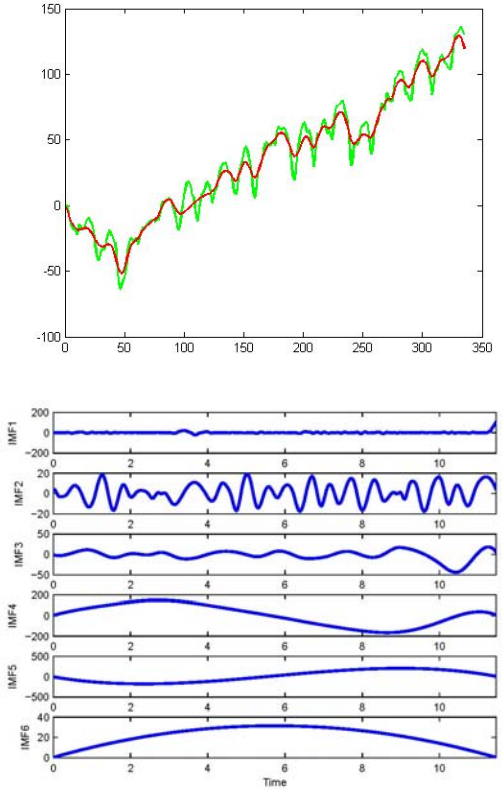


FIGURE IV. (A) THE GREEN LINE DENOTES THE ORIGINAL SIGNAL BEFORE SMOOTHING, AND THE RED LINE DENOTES THE OPTIMIZED MOTION SIGNAL AFTER SMOOTHING. (B) THE IMF SIGNALS DECOMPOSED FROM THE ORIGINAL SIGNAL.

Using $B(t) = S^{-1}(t)D(t)$, the original video frames could be transformed into the ones with spatial structure consistency while preserving salient regions. Figure III demonstrates the effectiveness of our saliency warping, which is better than traditional warping.

With the help of this technique, we eliminate the parallax between the spatially-variant grid cells within each frame. However, it cannot eliminate the jitters between the different video frames. We will describe the video stabilization between the different frames in the next section. In other words, we perform spatial smoothing based on spatial structure consistency and temporal smoothing based on temporal, feature-centric EMD, respectively.

V. SELF-ADAPTIVE IMFS BASED TEMPORAL, FEATURE-CENTRIC EMD OPTIMIZATION

SIFT Based Motion Signal Construction. We set S_1 as the arbitrary value at the first frame. Hence, the camera poses can be computed by chaining the relative motions between consecutive frames via $S_t = S_1T_1 \dots T_{t-1}$. We can convert a 3×3 transform T_t to a scale-rotation-translation transform, which returns the scale, rotation, and translation parameters, and the reconstituted transform T_t. We only focus on the scale, rotation, and translation parameters, and ignore other factors in our paper. The previous method only focuses on the optimization of the x-coordinate and the y-coordinate.

In practice, we find that it may bring about some potential problems. In order to solve these problems, our method focuses on the more comprehensive parameters, such as the scale, the angle, the x-coordinate, and the y-coordinate. The rotation parameter contains the angle. The translation parameter contains the x-coordinate and the y-coordinate. We then concatenate the scale, rotation, and translation parameters to a 4D vector $\hat{S}_t$ to represent the camera pose at time t. We regard the component in the vector $\hat{S}_t$ as a motion signal, denoted as the green line in Figure IV (a).

A. Self-Adaptive IMFs

EMD can decompose any complicated signal to generate IMFs via a sifting process, which needs to iteratively perform the following steps. The first step is to find the local extrema points (maxima and minima). The second step is to employ cubic spline interpolation to generate upper and lower envelopes. The third step is to judge whether the remainder is the IMF or not. The final step is to judge whether the residue is monotonic. Specifically, it can decompose the original signal $\hat{S}$ via

$$\hat{S} = \sum_{k=1}^{N} f_k + r_N. \tag{3}$$

Here $f_k(k = 1, \dots, N)$ are IMFs, and r_N is the corresponding residue. Figure IV (b) demonstrate the $IMF_k(k = 1, \dots, 5)$, and IMF_6 denotes the residue. To be easy to express and calculate, from now on we set $f_{N+1} = r_N$. In another word, we regard the residual as the last IMF.

The original signal shown in Figure IV (a) is decomposed into the IMFs and residual (Figure IV (b)). As shown in Figure IV (b), six IMFs represent the signals decomposed from the original signal. In order to stabilize the video, the high frequency signals should be smoothed. The optimal camera trajectory, denoted as the red line in Figure IV (a), is obtained by minimizing the following objective function.

$$\mathcal{O}(\alpha) = argmin(\|\nabla(\sum_{k=1}^{N+1} \alpha_k f_k)\|_1 + \|\nabla^2(\sum_{k=1}^{N+1} \alpha_k f_k)\|_1 + \|\nabla^3(\sum_{k=1}^{N+1} \alpha_k f_k)\|_1 + W\|\sum_{k=1}^{N+1} \alpha_k f_k - \hat{S}\|_1). \tag{4}$$

Here α denotes the ratio of the IMF. We use X to denote the variable. When $X = \sum_{k=1} \alpha_k f_k$, $\|\nabla(X)\|_1$, $\|\nabla^2(X)\|_1$ and $\|\nabla^3(X)\|_1$ are the $L1$ norms of the first order, second order and third order derivatives of X respectively. The minimum of the sum of $\|\nabla(X)\|_1$, $\|\nabla^2(X)\|_1$ and $\|\nabla^3(X)\|_1$ smooths the IMFs (shown in Figure IV (b)) to remove the jitters in the unstable video. $\hat{S}$ denotes the original signal, shown in Figure IV(a). The minimum of the difference of $\sum_{k=1} \alpha_k f_k$ and $\hat{S}$ keeps the original signal be close to the optimized signal to avoid excessive cropping. W is the adaptive equilibrium factor, which is used to balance the above four terms. This paper empirically sets W to 0.1. In summary, our optimization method comprehensively considers multiple competing factors, such as

eliminating vibration, excluding excessive cropping, and minimizing the distortional deformation. The optimization depicted in Eq. (4) is the convex optimization problem, which can be solved by Disciplined Convex Programming (CVX). The smoothing algorithm is documented in Eq. (4).

Following the Eq. (4), the new ratios of the IMFs are figured out. Then the optimized motion signal (shown in Figure IV (a)) could be calculated via $\hat{T} = \sum_{k=1}^{N+1} \hat{\alpha_k} f_k$. $\hat{T}$ is the optimized motion signal, denoted by the red line in Figure IV (a). $\hat{\alpha_k}$ is the new ratio of the IMF computed by the Eq. (4). Figure IV(a) shows the camera trajectories before and after smoothing in green and red line respectively.

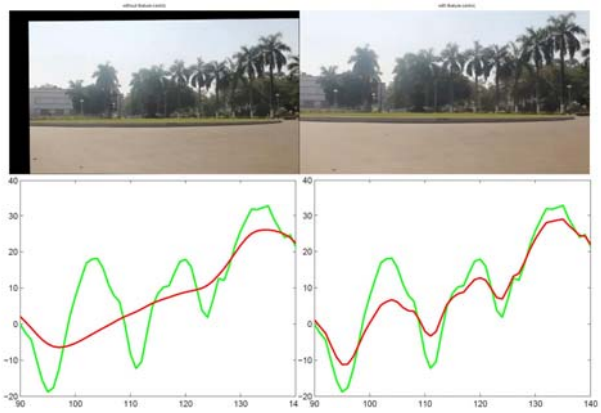


FIGURE V. (A) THE LEFT PICTURE SHOWS THE RESULT WITHOUT FEATURE-CENTRIC EMD. THE RIGHT PICTURE SHOWS THE RESULT WITH FEATURE-CENTRIC EMD. THE LEFT PICTURE IS EXCESSIVELY CROPPED. (B) THE GREEN LINE DENOTES THE MOTION SIGNAL OF ORIGINAL PATH. THE RED LINE DENOTES THE MOTION SIGNAL OF SMOOTHING PATH WITHOUT FEATURE-CENTRIC EMD. (C) THE GREEN LINE DENOTES MOTION SIGNAL OF THE ORIGINAL PATH. THE RED LINE DENOTES THE MOTION SIGNAL OF THE SMOOTHING PATH WITH FEATURE-CENTRIC EMD.

Traditional adaptive path smoothing method mainly executes path smoothing by simple mathematical algorithm, which lacks self-learning ability and repeated iteration. Our self-adaptive IMFs method converts path smoothing problem to signal processing problem, which is self-learning and iterative. Our method could compute the ratio of each IMF by autonomic learning, then improves smoothness by repeated iteration.

B. Feature-Centric EMD

As shown in Figure V (b), the green line denotes the motion signal of the original path, and the red line denotes the motion signal of the smoothing path without feature-centric EMD. The original is over-smoothed and loses the original trend of the motion, which leads to excessive cropping, as shown in left image of the Figure V (a). The left picture shows the result without feature-centric EMD. The right picture shows the result with the feature-centric EMD. The left picture is excessively cropped. To keep the tendency of the original EMD motion signal, we define the extreme point of the original motion signal as the feature. In other words, centric feature means the motion trend of EMD signals. To keep the centric feature of EMD signals while smoothing the signals, our feature-centric EMD is formulated in Eq. [equ.feature1].

$$\tilde{T}_t = (1-\tilde{W})\frac{\Sigma_{\tilde{t}\in\omega_t}(G_t(\|S_{\tilde{t}}-S_t\|)\hat{T}_{\tilde{t}})}{\Sigma_{\tilde{t}\in\omega_t}G_t(\|S_{\tilde{t}}-S_t\|)} + \tilde{W}\frac{\Sigma_{\tilde{t}\in\omega_t}(G_t(\|\hat{T}_{\tilde{t}}-\hat{T}_t\|)S_{\tilde{t}})}{\Sigma_{\tilde{t}\in\omega_t}G_t(\|\hat{T}_{\tilde{t}}-\hat{T}_t\|)}. \quad (5)$$

All the parameters used in this method are the optimal values obtained in the experiments. Here we empirically set ω_t denotes the 60 neighboring frames. We bring in the Gaussian functions $G_t()$, and empirically set the standard deviation of $G_t()$ to 10. S_t denotes the original value at the frame t without the feature-centric EMD. $S_{\tilde{t}}$ denotes the original value at the frame $\tilde{t}$ without the feature-centric EMD. $\hat{T}_t$ denotes the optimized value at the frame t. $\hat{T}_{\tilde{t}}$ denotes the optimized value at the frame $\tilde{t}$. $\hat{T}_t$ denotes the value at the frame t with the feature-centric EMD. Eq. [equ.feature1] makes the new path keep the trend of the original path, while successfully suppressing both high frequency jitters and low-frequency bounces of the original path.

In Eq. [equ.feature1], $\frac{\Sigma_{\tilde{t}\in\omega_t}(G_t(\|S_{\tilde{t}}-S_t\|)\hat{T}_{\tilde{t}})}{\Sigma_{\tilde{t}\in\omega_t}G_t(\|S_{\tilde{t}}-S_t\|)}$ mainly suppresses the shaky components of the original path, simultaneously keeping its initial trend. Meanwhile $\frac{\Sigma_{\tilde{t}\in\omega_t}(G_t(\|\hat{T}_{\tilde{t}}-\hat{T}_t\|)S_{\tilde{t}})}{\Sigma_{\tilde{t}\in\omega_t}G_t(\|\hat{T}_{\tilde{t}}-\hat{T}_t\|)}$ mainly makes the new path keep the trend of the original path, moreover suppressing its trembling signal. $\tilde{W}$ is the adaptive equilibrium factor ranging from 0 to 1, which is used to balance the above two terms. We set $\tilde{W}$ as 0.4 for all our examples. When smoothing the signal, the fast panning or scene transition may cause rapid signal motion. In this case, some excessive cropping may be yielded by inappropriate smoothing. The motion signal may significantly deviate from its original path, as indicated by the green lines in Figure V (b).

Our adaptive feature-centric EMD method can accommodate sudden camera motions to a certain degree. The result from our adaptive smoothing produces much less cropping, as shown in Figure V (c). The green line denotes the motion signal of the original path. The red line denotes the motion signal of the smoothing path with feature-centric EMD, which gives rise to the perfect result (shown in the right side of Figure V (a)).

C. Region Extrapolation and Interpolation

To complete the blank caused by the frame translation, rotation and stretching (Figure V(a)), we carry out the lost region extrapolation and the overlapping part interpolation using video extrapolation based on the adaptive temporal range (Figure VI) and interpolation based on cubic spline interpolation (Figure VII). Due to faint change of adjacent frames, this paper ignores foreground/background detections.

We select library images with our adaptive temporal range method (Figure VI) and warp them by the As-Similar-As-Possible algorithm proposed in [13]. Using the warped images based on our adaptive temporal range method, we extrapolate all the video frames following the method by [27]. Specifically, we first detect and match the features of scale-invariant feature transform (SIFT) between the frame t and its neighbors.Then we could compute the matching ratio by the division of the number

of the matching features and the number of the total features. As shown in Figure VI, the percentage is the value of the matching ratio. With the decline of the matching ratio, the transformed library images will be distorted. So we empirically set the threshold as 65% and select the frames with the matching ratio greater than 65% as library images. So our adaptive temporal range is $[t-E+1, t+E]$ in Figure VI.

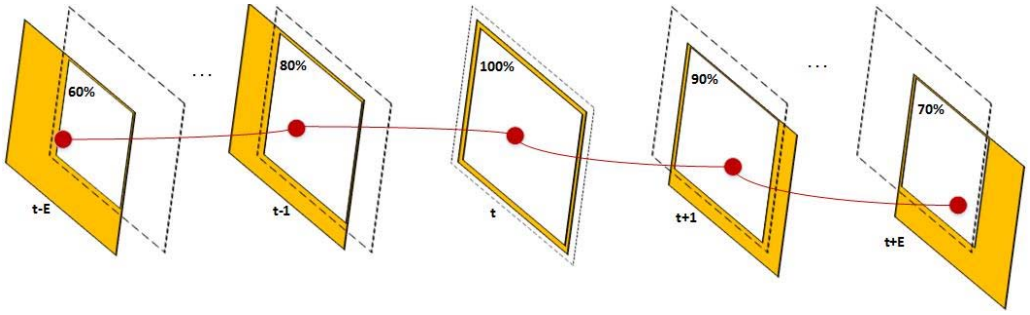


FIGURE VI. ESTIMATION OF ADAPTIVE TEMPORAL RANGE. THE RED LINE DENOTES THE SIFT FEATURE TRAJECTORY OF ALL THE ADJACENT FRAMES. THE WHITE DOTTED BOX DENOTES THE FRAME T. THE YELLOW BOX DENOTES THE NEIGHBOR OF THE FRAME T. THE WHITE SOLID BOX DENOTES THE MATCHING RATE BETWEEN THE FRAME T AND ITS NEIGHBORS. THE ADAPTIVE TEMPORAL RANGE IS SELECTED SUCH THAT THE MATCHING RATE (SHOWN IN THE UPPER LEFT CORNER OF THE SOLID BOX) IS GREATER THAN OUR THRESHOLD.

As there may be some uneven transition in the overlapping part after extrapolation. We improve it with cubic spline interpolation. Figure VII shows the interpolating gray value of the unknown pixel (x, y), which could be solved by the weighted interpolation of its nearby sixteen gray scale values. The gray scale formula of the unknown pixel is defined as:

$$g(x,y) = g(m+u, n+v) = A\,\tilde{B}\,C. \tag{6}$$

The values of A, $\tilde{B}$ and C are computed as:

$$\mathbf{A} = [Z(1+v) \quad Z(v) \quad Z(1-v) \quad Z(2-v)], \tag{7}$$

$$\tilde{B} = \begin{bmatrix} g(m-1,n-1) & g(m-1,n) & g(m-1,n+1) & g(m-1,n+2) \\ g(m,n-1) & g(m,n) & g(m,n+1) & g(m,n+2) \\ g(m+1,n-1) & g(m+1,n) & g(m+1,n+1) & g(m+1,n+2) \\ g(m+2,n-1) & g(m+2,n) & g(m+2,n+1) & g(m+2,n+2) \end{bmatrix}, \tag{8}$$

$$\mathbf{C} = [Z(1+u) \quad Z(u) \quad Z(1-u) \quad Z(2-u)]^T. \tag{9}$$

The value of $Z(\tilde{x})$ is computed as:

$$Z(\tilde{x}) = \begin{cases} 1 - 2|\tilde{x}|^2 + |\tilde{x}|^3 & 0 \le |\tilde{x}| < 1 \\ 4 - 8|\tilde{x}| + 5|\tilde{x}|^2 - |\tilde{x}|^3 & 1 \le |\tilde{x}| < 2 \\ 0 & |\tilde{x}| \ge 2 \end{cases} \tag{10}$$

VI. EXPERIMENTS AND EVALUATIONS

We run our method on an Intel i7 3.4GHZ Eight-core computer with 16G RAM. We extract 500-800 SIFT features per frame. For motion estimation, we always divide the video frame to 8×8 cells. For a video of 1280×720 resolution, our stabilizing system takes 800 milliseconds to process a frame. Specifically, we spend 250ms, 500ms, 15ms and 35ms to extract features, estimate motion, optimize camera paths and render the final result. All original and result videos will be provided later. In order to facilitate comparisons, we evaluate our method on some challenging examples from publicly available videos in prior publications. These example videos include large parallax, dynamic objects, large depth variations, and rolling shutter effects. All comparisons and results accompanied by our paper will be provided later. We collect a comprehensive dataset of more than 200 short videos (less than 70 seconds) from the previous publications and the internet. To compare the advantages and disadvantages of a method in different situations, our data is divided into multiple categories based on camera motion and scene type. They are classified as regular, repeatedly back-and-forth panning, quick rotation, zooming, parallax, fast, crowd, running, and Rolling Shutter Effects categories, which will be provided later.

A. Comparisons Over Different Complex Videos

Videos with Repeatedly Back-And-Forth Panning. We evaluate our method on four videos with repeatedly back-and-forth panning, which come from the paper [23]. Our 2D motion model achieves results with similar visual quality of 3D method. The video thumbnails are shown in Figure VIII. We compare our results with the method proposed by Okade et al. [23]. As can be seen from the accompany videos, the results from [23] contain massive black edges and jitters at some image regions. Please refer to our accompany videos for more vivid comparisons.

Videos with Quick Rotation. We evaluate our method on four videos with quick rotation, which come from the paper by Liu et al. [19]. Our 2D motion model achieves results with similar visual quality of 3D method. The video thumbnails are shown in Figure IX. We compare our results with the method proposed by Liu et al. [19]. As can be seen from the accompany videos, the results from [19] contain massive excessive cropping at some image regions. Please refer to our accompany videos for more vivid comparisons.

Videos with Zooming. We evaluate our method on four videos with zooming, which come from the paper [19]. Our 2D motion model achieves results with similar visual quality of 3D method. The video thumbnails are shown in Figure X. We compare our results with the method proposed by Liu et al. [19]. As can be seen from the accompany videos, the results from [19]

also contain massive excessive cropping at some image regions. Please refer to accompany videos for more vivid comparisons.

Videos with Parallax. We evaluate our method on four videos with parallax, which come from the methods proposed by

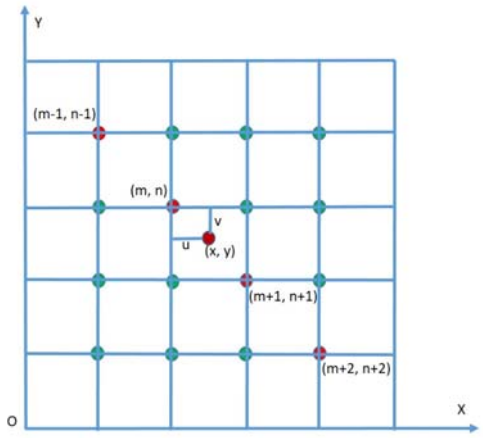


FIGURE VII. THE GRAY VALUE OF THE UNKNOWN PIXEL (X, Y)

Liu et al. [19] and Liu et al. [20]. Our 2D motion model achieves results with similar visual quality of 3D method. The video thumbnails are shown in Figure XI. We compare our results with the method by Liu et al. [19] and Liu et al. [20]. As can be seen from the accompany videos, the results from [19] and Liu et al. [20] contain massive excessive cropping at some image regions. Please refer to the accompany videos for more vivid comparisons.

Videos with Rolling Shutter Effects. Rolling shutter is a method for image capture, in which a still picture (in a still camera) or each frame of a video (in a video camera) is captured not by taking a snapshot of the entire scene at single instant in time but rather by scanning across the scene rapidly, either vertically or horizontally. In other words, not all parts of the image of the scene are recorded at exactly the same time. (Though, during playback, the entire image of the scene is displayed at once, as if it represents a single instant in time.) Thus, it produces predictable distortions of fast-moving objects or rapid flashes of light. It is in contrast with 'global shutter', in which the entire frame is captured at the same instant. Rolling shutter effects may cause spatially-variant motions in different regions of the frames in the shaking videos. It could be modeled as spatially-variant high frequency thrashing. Our optimization model is based on spatial structure consistency warping, so it can simultaneously rectify rolling shutter effects when smoothing camera bundled-trajectories. Figure XII shows four videos with rolling shutter effects, coming from Guo et al. [9], Liu et al. [19] and Liu et al. [20]. Our method can produce similar quality as other state-of-art techniques ([2][7][11][20]).

B. Comparisons With State-Of-The-Art Methods

As shown in Figure XIII, the first row shows the frames from 177 to 180 smoothed by Okade et al. [23]. The second row shows the frames from 177 to 180 smoothed by our method. We mark the sequence number with the red numbers on the upper left corner of each frame. The regions on the edge of the picture in the first row demonstrate that, the frame is seriously distorted. The regions on the edge of the picture in the second row demonstrate that, the frame is smoothed better. The supplemented video could verify that, the video smoothed by our method is more stable than that of Okade et al. [23]. As shown in Figure XIV, the first row shows the frames from 240 to 243 smoothed by Liu et al. [20]. The second row shows the frames from 240 to 243 smoothed by our method. We mark the sequence number on the upper left corner and the arrow pointing to the white makers on the lower left corner of each frame. The regions marked by the arrow in the first row demonstrate that, the white marker is close to the edge of the image. The regions marked by the arrow in the second row demonstrate that, the white marker is far from the edge of the image. The accompanied video could verify that, the video smoothed by Liu et al. [20] is cropped more background than that by our method.

C. Comparisons With Popular Commercial Softwares

We further compare our method with two popular commercial systems. We conduct the comparisons on the videos with publicly-available results. The first system is the YouTube stabilizer, which is based on the combination of the L1-optimization method proposed by Grundmann et al. [8] and the homography mixture method proposed by Grundmann et al. [7]. The YouTube stabilizer is an on-line parameter-free tool, which could automatically stabilize the uploaded videos. We upload our videos to YouTube and download the smoothed results. Another system is the warp stabilizer in the Adobe Premiere Pro CC, which is based on the subspace stabilization method proposed by Liu et al. [16]. Since it is an interactive system, which can be tuned via a few parameters, such as smoothness, position, boundary, scaling rate and cropping rate. We tune the above parameters time and again to generate the best results.

As shown in Figure XV, the first row shows the frames from 184 to 187 smoothed by the YouTube stabilizer. The second row shows the frames from 184 to 187 smoothed by our method. We mark the sequence number on the upper left corner, and the arrow on the distortion place of each frame. The regions marked by the arrow in the first row demonstrate that, the ground is distorted and blurred. The regions marked by the arrow in the second row demonstrate that, the ground is smoothed better. The supplemented video could verify that, the video smoothed by our method is more stable than that of the YouTube stabilizer.

As shown in Figure XVI, the first row shows the frames from 430 to 433 smoothed by the warp stabilizer in the Adobe Premiere Pro CC. The second row shows the frames from 430 to 433 smoothed by our method. We mark the sequence number on the upper left corner, and the arrow on the distortion place of each frame. The regions marked by the arrow in the first row demonstrate that, the stage is distorted and blurred. The regions marked by the arrow in the second row demonstrate that, the stage is smoothed better. The objects in the second row are far more than the first row, which indicates that our pruning rate is lower than the warp stabilizer. The accompanied video could further verify that, the video stabilized by our method is more stable than that by the warp stabilizer.

D. Comparisons With and Without Proposed Components

Videos with and without saliency preserving. As shown in Figure XVII, the first row shows the frames from 1 to 4 smoothed without saliency preserving. The second row shows

the optical flow of frames from 1 to 4 smoothed without saliency preserving. The third row shows the frames from 1 to 4 smoothed with saliency preserving. The fourth row shows the optical flow of frames from 1 to 4 smoothed with saliency preserving. The regions on the edge of the picture in the first row and the third one demonstrate that, the frames with saliency preserving are smoothed better. Comparing Figure XVII (b) with Figure XVII (d), the optical flow smoothed with saliency preserving is clearer than that without saliency preserving.

FIGURE VIII. THE COMPARISON WITH THE METHOD BY OKADE ET AL. [23] OVER FOUR VIDEOS WITH REPEATED BACK-AND-FORTH PANNING.

FIGURE IX. COMPARISON WITH THE METHOD BY LIU ET AL. [19] OVER FOUR VIDEOS WITH QUICK ROTATIONS.

FIGURE X. THE COMPARISONS WITH THE METHOD PROPOSED BY LIU ET AL. [19] OVER FOUR VIDEOS WITH ZOOMING.

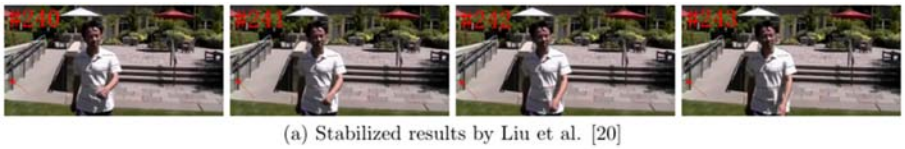
FIGURE XI. COMPARISONS WITH THE METHODS PROPOSED BY LIU ET AL. [19] AND LIU ET AL. [20] OVER FOUR VIDEOS WITH PARALLAX.

FIGURE XII. COMPARISONS WITH THE METHODS PROPOSED BY GUO ET AL. [9], LIU ET AL. [19] AND LIU ET AL. [20] OVER FOUR VIDEOS WITH ROLLING SHUTTER EFFECTS.

(a) Stabilized result by Okade et al. [23]

(b) Stabilized result by our method

FIGURE XIII. (A) THE FIRST ROW SHOWS THE FRAMES FROM 177 TO 180 SMOOTHED BY THE METHOD OF OKADE ET AL. [23]. (B) THE SECOND ROW SHOWS THE FRAMES FROM 177 TO 180 SMOOTHED BY OUR METHOD. WE MARK THE SEQUENCE NUMBER ON THE UPPER LEFT CORNER OF EACH FRAME.



(a) Stabilized results by Liu et al. [20]

(b) Stabilized results by our method

FIGURE XIV. (A) THE FIRST ROW SHOWS THE FRAMES FROM 240 TO 243 SMOOTHED BY LIU ET AL. [20]. (B) THE SECOND ROW SHOWS THE FRAMES FROM 240 TO 243 SMOOTHED BY OUR METHOD. WE MARK THE SEQUENCE NUMBER ON THE UPPER LEFT CORNER, AND THE ARROW POINTING TO THE WHITE MAKERS ON THE LOWER LEFT CORNER OF EACH FRAME.

(a) Stabilized results by the YouTube stabilizer

(b) Stabilized results by our method

FIGURE XV. (A) THE FIRST ROW SHOWS THE FRAMES FROM 184 TO 187 SMOOTHED BY THE YOUTUBE STABILIZER. (B) THE SECOND ROW SHOWS THE FRAMES FROM 184 TO 187 SMOOTHED BY OUR METHOD. WE MARK THE SEQUENCE NUMBER ON THE UPPER LEFT CORNER, AND THE ARROW ON THE DISTORTION PLACE OF EACH FRAME.

(a) Stabilized results by the warp stabilizer

(b) Stabilized results by our method

FIGURE XVI. (A) THE FIRST ROW SHOWS THE FRAMES FROM 430 TO 433 SMOOTHED BY THE WARP STABILIZER. (B) THE SECOND ROW SHOWS THE FRAMES FROM 430 TO 433 SMOOTHED BY OUR METHOD. WE MARK THE SEQUENCE NUMBER ON THE UPPER LEFT CORNER, AND THE ARROW ON THE DISTORTION PLACE OF EACH FRAME.

Videos with and without self-adaptive IMFs. As shown in Figure XVIII, the first row shows the frames from 40 to 43 smoothed without self-adaptive IMFs. The second row shows the optical flow of frames from 40 to 43 smoothed without self-adaptive IMFs. The third row shows the frames from 40 to 43 smoothed with self-adaptive IMFs. The fourth row shows the optical flow of frames from 40 to 43 smoothed with self-adaptive IMFs. The zebra crossing on the center of the picture in the first row and the third one demonstrate that, the frames with self-adaptive IMFs are smoothed better. Comparing Figure XVIII (b) with Figure XVIII (d), the optical flow smoothed with self-adaptive IMFs is clearer than that without self-adaptive IMFs.

E. Quantitative Evaluation

To quantitatively evaluate and measure the results from different aspects, we define two objective metrics: Cropping Rate and Stability Rate. In order to protect the salient area, there is a process of retargeting in the method of spatial structure consistency optimization. The change of frame size and aspect ratio makes it impossible to calculate remaining area quantitatively, so this paper has to design a new measure method. For Cropping Rate, we use the feature of the scale-invariant feature transform (SIFT) to compare the cropping rate of the result. We first detect the features in each frame between input video and output video.

FIGURE XVII. (A) THE FIRST ROW SHOWS THE FRAMES FROM 1 TO 4 SMOOTHED WITHOUT SALIENCY PRESERVING. (B) THE SECOND ROW SHOWS THE OPTICAL FLOW OF FRAMES FROM 1 TO 4 SMOOTHED WITHOUT SALIENCY PRESERVING. (C) THE THIRD ROW SHOWS THE FRAMES FROM 1 TO 4 SMOOTHED WITH SALIENCY PRESERVING. (D) THE FOURTH ROW SHOWS THE OPTICAL FLOW OF FRAMES FROM 1 TO 4 SMOOTHED WITH SALIENCY PRESERVING.

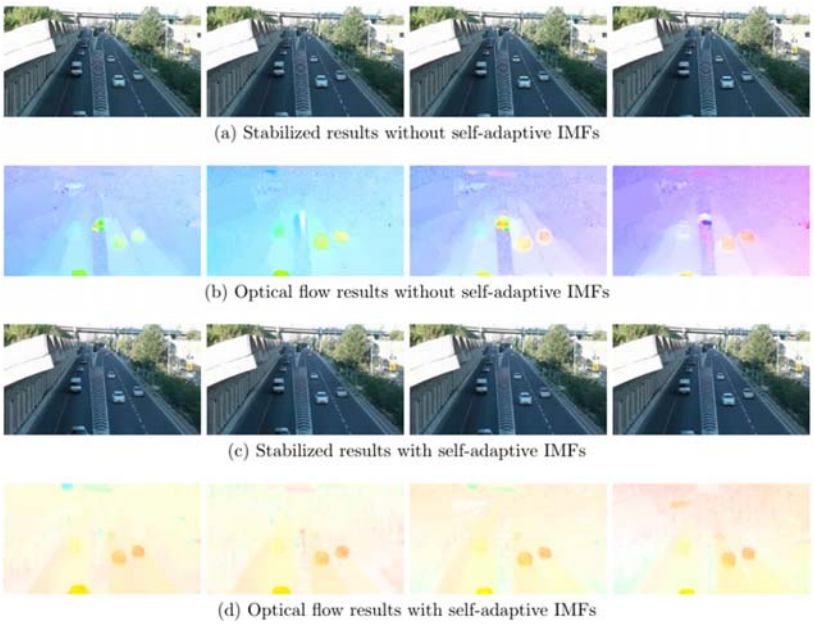


FIGURE XVIII. (A) THE FIRST ROW SHOWS THE FRAMES FROM 40 TO 43 SMOOTHED WITHOUT SELF-ADAPTIVE IMFS. (B) THE SECOND ROW SHOWS THE OPTICAL FLOW OF FRAMES FROM 40 TO 43 SMOOTHED WITHOUT SELF-ADAPTIVE IMFS. (C) THE THIRD ROW SHOWS THE FRAMES FROM 40 TO 43 SMOOTHED WITH SELF-ADAPTIVE IMFS. (D) THE FOURTH ROW SHOWS THE OPTICAL FLOW OF FRAMES FROM 40 TO 43 SMOOTHED WITH SELF-ADAPTIVE IMFS.

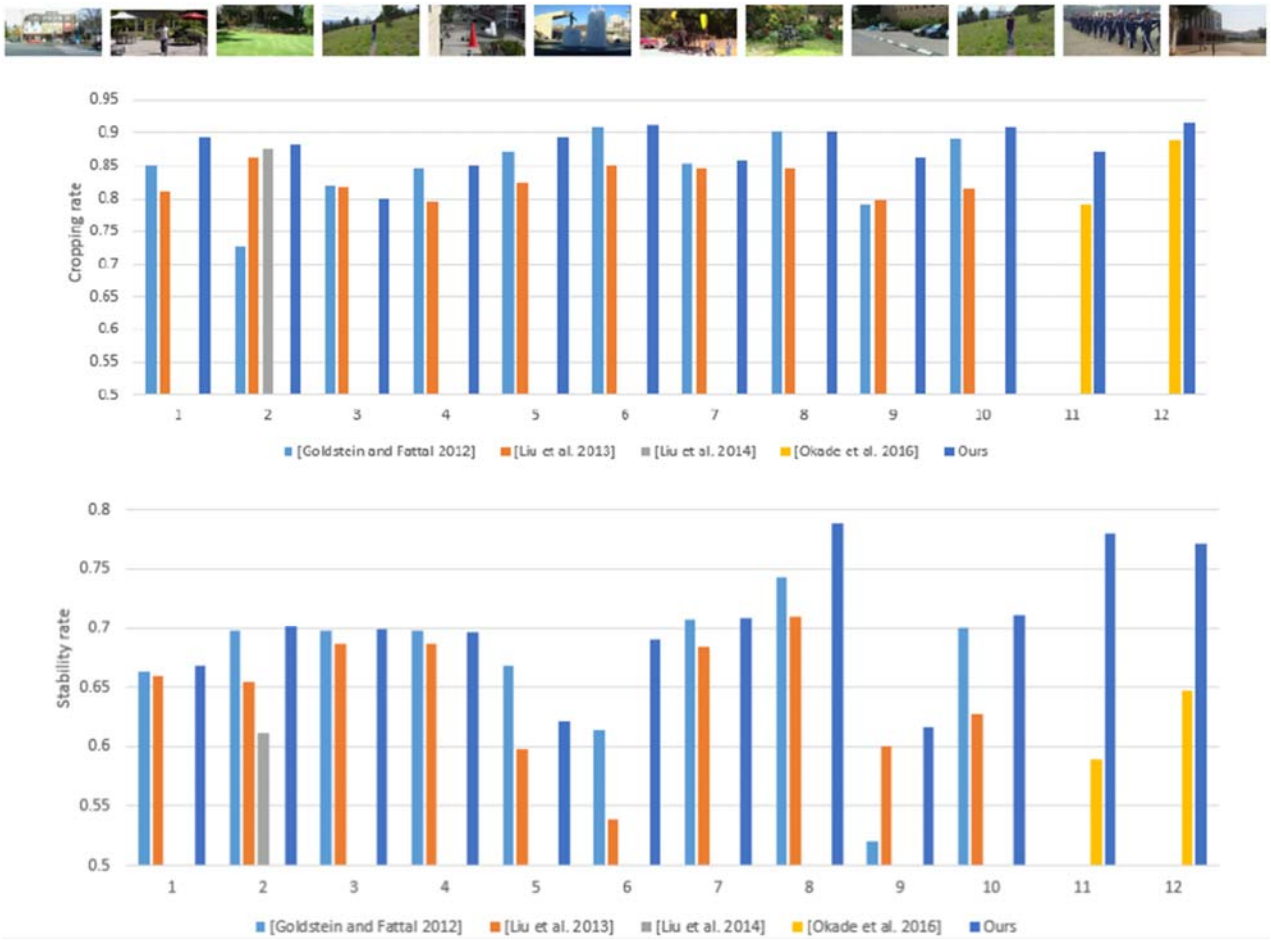


FIGURE XIX. QUANTITATIVE COMPARISONS WITH STATE-OF-THE-ART METHODS ON PUBLICLY-AVAILABLE DATA.

Then, we could compute the cropping ratio by the division of the number of the matching features and the number of the total features. We average these computations of all frames as the final metric.

As for Stability Rate, our basic assumption is that, the more energy is contained in the low frequency part of the motion, the more stable a video is. We first fit a global homography at each frame between input video and output video. Computationally, we estimate our self-adaptive IMFs based camera paths to approximate the true motion in a video. We do not smooth out anything after the estimation. Then, we extract translation and rotation components from each path. Each component is a 1D temporal signal. Finally, we evaluate the energy percentage of the low frequency components in these 1D signals to measure the stability. Specifically, we use the EMD method to decompose a signal into multiple Intrinsic Mode Functions (IMFs) with a trend, and apply the Hilbert spectral analysis (HSA) method to the IMFs to obtain instantaneous frequency data. Then, we take half of the lowest frequencies and calculate the energy percentage over full frequencies. We take the smallest measurement among the translation and rotation as the final metric.

We compare our results with some previous results shown in [**Goldstein and Fattal 2012; Liu et al. 2013; Liu et al. 2014; Okade et al. 2016**]. We collect twelve test videos from these papers (thumbnails in Figure XIX), and compare our results with their published results (all from the project web pages of the authors). Overall, all methods generate similar stability both subjectively and quantitatively (Figure XIX) on these examples, while our results are slightly better on some videos in terms of cropping ratio.

VII. CONCLUSIONS AND DISCUSSIONS

In this paper, we have systematically presented a novel automatic method to address a suite of research challenges encountered in video stabilization. It is the first time that, the Empirical Mode Decomposition improved via convex optimization is applied to the field of the video stabilization. With the help of Empirical Mode Decomposition, our method could autonomously learn the optimization strategy for the video stabilization. The extensive experiments demonstrate the effectiveness of our method in handling wobbly videos with high-variability and broad-coverage of things and/or entities. All of these technical innovations contribute to automatic video stabilization with state-of-the-art performance in accuracy, versatility, flexibility, and efficiency.

Nevertheless, our method may still have tremendous rooms to improve. For example, although our EMD-based motion model could adaptively proceed the frame registration between

the adjacent frames to stabilize the video, however, it may bring about excessive cropping or sacrifice motion accuracy and eventual stability of the result in some cases, such as the repeatedly quick back-and-forth movement, severe occlusions and running. Besides, to minimize the geometrical distortion, our model tries our best efforts to enforce strong coherence between grid cells. In this way, it may further sacrifice motion accuracy. The quality of the frames may be degraded by the blurry while stabilizing the unstable videos. In the future, we would utilize a more intelligent approach to extend this kind of representation to the camera path optimization and deblur the low-quality video based on Artificial Neural Network.

VIII. ACKNOWLEDGEMENTS

This research is supported in part by National Natural Science Foundation of China (NO. 61672077 and 61532002), Applied Basic Research Program of Qingdao (NO. 161013xx), National Science Foundation of USA (NO. IIS-0949467, IIS-1047715, IIS-1715985, and IIS-1049448), and capital health research and development of special 2016-1-4011.

REFERENCES

[1] Aguilar W G, Angulo C. Real-Time Model-Based Video Stabilization for Microaerial Vehicles[J]. Neural Processing Letters, 2016, 43(2): 459-477.

[2] Baker S, Bennett E P, Kang S B, et al. Removing rolling shutter wobble[C]. computer vision and pattern recognition, 2010: 2392-2399.

[3] Buehler C, Bosse M, Mcmillan L, et al. Non-metric image-based rendering for video stabilization[C]. computer vision and pattern recognition, 2001: 609-614.

[4] Dragon R, Rosenhahn B, Ostermann J, et al. Multi-scale clustering of frame-to-frame correspondences for motion segmentation[C]. european conference on computer vision, 2012: 445-458.

[5] Gleicher M, Liu F. Re-cinematography: improving the camera dynamics of casual video[C]. acm multimedia, 2007: 27-36.

[6] Goldstein A, Fattal R. Video stabilization using epipolar geometry[J]. ACM Transactions on Graphics, 2012, 31(5).

[7] Grundmann M, Kwatra V, Castro D, et al. Calibration-free rolling shutter removal[C]. international conference on computational photography, 2012: 1-8.

[8] Grundmann M, Kwatra V, Essa I A, et al. Auto-directed video stabilization with robust L1 optimal camera paths[C]. computer vision and pattern recognition, 2011: 225-232.

[9] Guo H, Liu S, Zhu S, et al. Joint bundled camera paths for stereoscopic video stabilization[C]. international conference on image processing, 2016: 1071-1075.

[10] Hu W, Chen C, Su Y, et al. Feature-based real-time video stabilization for vehicle video recorder system[J]. Multimedia Tools and Applications, 2018, 77(5): 5107-5127.

[11] Alexandre Karpenko, David Jacobs, Jongmin Baek, and Marc Levoy. Digital video stabilization and rolling shutter correction using gyroscopes. CSTR, 1:2, 2011.

[12] Kuipers L, Timman R, Cohen J, et al. Handbook of mathematics[J]. The Mathematical Gazette, 1970, 54(389).

[13] Levi Z, Gotsman C. D-Snake: Image Registration by As-Similar-As-Possible Template Deformation[J]. IEEE Transactions on Visualization and Computer Graphics, 2013, 19(2): 331-343.

[14] Xiao Li, Shuai Li, Hong Qin, and Aimin Hao. Spatiotemporal consistency-based adaptive hand-held video stabilization. SCIENCE CHINA Information Sciences, doi: 10.1007/s11432-018-9764-0, 2019.

[15] Liu F, Gleicher M, Jin H, et al. Content-preserving warps for 3D video stabilization[J]. international conference on computer graphics and interactive techniques, 2009, 28(3).

[16] Liu F, Gleicher M, Wang J, et al. Subspace video stabilization[J]. ACM Transactions on Graphics, 2011, 30(1).

[17] Liu S, Wang Y, Yuan L, et al. Video stabilization with a depth camera[C]. computer vision and pattern recognition, 2012: 89-95.

[18] Liu S, Xu B, Deng C, et al. A Hybrid Approach for Near-Range Video Stabilization[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2017, 27(9): 1922-1933.

[19] Liu S, Yuan L, Tan P, et al. Bundled camera paths for video stabilization[J]. international conference on computer graphics and interactive techniques, 2013, 32(4).

[20] Liu S, Yuan L, Tan P, et al. SteadyFlow: Spatially Smooth Optical Flow for Video Stabilization[C]. computer vision and pattern recognition, 2014: 4209-4216.

[21] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110.

[22] Narayana M, Hanson A R, Learnedmiller E G, et al. Coherent Motion Segmentation in Moving Camera Videos Using Optical Flow Orientations[J]. international conference on computer vision, 2013: 1577-1584.

[23] Okade M, Patel G, Biswas P K, et al. Robust Learning-Based Camera Motion Characterization Scheme With Applications to Video Stabilization[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2016, 26(3): 453-466.

[24] Page G F. MULTIPLE VIEW GEOMETRY IN COMPUTER VISION, by Richard Hartley and Andrew Zisserman, CUP, Cambridge, UK, 2003, vi+560 pp., ISBN 0-521-54051-8. (Paperback £44.95)[J]. Robotica, 2005, 23(2): 271-271.

[25] Rubinstein M, Shamir A, Avidan S, et al. Improved seam carving for video retargeting[J]. international conference on computer graphics and interactive techniques, 2008, 27(3).

[26] Smith B M, Zhang L, Jin H, et al. Light field video stabilization[C]. international conference on computer vision, 2009: 341-348.

[27] Wang M, Lai Y, Liang Y, et al. BiggerPicture: data-driven image extrapolation using graph matching[J]. international conference on computer graphics and interactive techniques, 2014, 33(6).

[28] Wang Y, Liu F, Hsu P S, et al. Spatially and Temporally Optimized Video Stabilization[J]. IEEE Transactions on Visualization and Computer Graphics, 2013, 19(8): 1354-1361.

[29] Zhang L, Chen X, Kong X, et al. Geodesic Video Stabilization in Transformation Space[J]. IEEE Transactions on Image Processing, 2017, 26(5): 2219-2229.

[30] Zhou Z, Jin H, Ma Y, et al. Plane-Based Content Preserving Warps for Video Stabilization[C]. computer vision and pattern recognition, 2013: 2299-2306.