A Hybrid Approach for Near-Range Video Stabilization

Shuaicheng Liu, *Member, IEEE*, Binhan Xu, Chuang Deng, *Member, IEEE*, Shuyuan Zhu, *Member, IEEE*, Bing Zeng, *Fellow, IEEE*, and Moncef Gabbouj, *Fellow, IEEE*

***Abstract*— Near-range videos contain objects that are close to the camera. These videos often contain discontinuous depth variation (DDV), which is the main challenge to the existing video stabilization methods. Traditionally, 2D methods are robust to various camera motions (e.g., quick rotation and zooming) under scenes with continuous depth variation (CDV). However, in the presence of DDV, they often generate wobbled results due to the limited ability of their 2D motion models. Alternatively, 3D methods are more robust in handling near-range videos. We show that, by compensating rotational motions and ignoring translational motions, near-range videos can be successfully stabilized by 3D methods without sacrificing the stability too much. However, it is time-consuming to reconstruct the 3D structures for the entire video and sometimes even impossible due to rapid camera motions. In this paper, we combine the advantages of 2D and 3D methods, yielding a hybrid approach that is robust to various camera motions and can handle the near-range scenarios well. To this end, we automatically partition the input video into CDV and DDV segments. Then, the 2D and 3D approaches are adopted for CDV and DDV clips, respectively. Finally, these segments are stitched seamlessly via a constrained optimization. We validate our method on a large variety of consumer videos.**

***Index Terms*— Continuous depth variation (CDV), discontinuous depth variation (DDV), near-range videos, spatial–temporal optimization, video stabilization.**

I. Introduction

VARIOUS video stabilization methods have been proposed in recent years to tackle challenging consumer-level videos [1], [2] that are captured during walking, running, riding, driving, and so on. These methods can handle scenes with large parallax quite well, as long as the depth variation is continuous. However, when the scene contains discontinuous depth variation (DDV), they often produce unpleasant results, e.g., suffering from wobbling artifacts or rolling back to the original footage. In fact, the DDV problem is the most challenging one that remains unresolved in the video stabilization.

According to the adopted motion model, video stabilization methods can be classified as 2D-based [3], 3D-based [4], and 2.5D-based [5]. The 2D methods rely on using 2D transformation (e.g., affine and homography) to model the camera motion. In principle, a homography can align two frames well only when the scene is flat or there is no camera translation at all. These conditions are difficult to meet in most real scenarios, so that the single-homography method would cause serious distortion after stabilization, especially for near-range footages. Studies in [6] and [7] adopt multiple homographies to align frames. They relax the problem to a certain extent, such that the scenes with continuous depth variation (CDV), instead of pure planes, can be well handled. However, they still cannot overcome the challenging DDV cases. The 3D-based methods reconstruct 3D camera trajectories and smooth them for stabilized videos. The 2.5D methods relax the full 3D reconstruction to partial 3D information, such as epipolar constraints [8] and subspace constraints [5].

The major drawback of 2D methods is their limited ability to handle DDV. On the other hand, their advantages are also obvious or even fascinating: they are fast, robust, and practical. In contrast, 3D methods have the potential to deal with DDV. However, they are slow, computationally expensive, and brittle.

In this paper, we aim to combine the strengths of both 2D and 3D video stabilization methods. Given an input video, we partition it into two types of segments: DDV and CDV. The 3D stabilization is applied to DDV segments, whereas CDV segments are stabilized by a 2D approach, namely, the bundled camera paths [6]. In particular, we apply the 3D reconstruction to DDV segments by smoothing only their camera rotational motions. Then, we put these segments back to the original video, leading to a partially stabilized video whereby some parts are stabilized and some parts not. Next, we apply a constrained bundled-path optimization to this recomposed footage, where the unstabilized CDV segments are stabilized and stitched seamlessly to the DDV segments. It is worth emphasizing that directly applying traditional 3D approaches (e.g., [4]) to process near-range videos would lead to noticeable wobbling effects. According to an evaluation on a host of real examples, we demonstrate that the compensation of translational motions would cause severe distortions while improving the video stability very slightly. Therefore, we argue

Manuscript received September 8, 2015; revised November 26, 2015, February 17, 2016, February 25, 2016, and March 24, 2016; accepted April 9, 2016. Date of publication April 20, 2016; date of current version September 5, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61502079, Grant 61370148, and Grant 61300091; in part by the Research Program for Scientific and Technological Innovation of Sichuan Province, China, under Grant 2014TD0006; and in part by State Grid Corporation of China through the Research and Development Project under Grant 521999150031. This paper was recommended by Associate Editor R. Hamzaoui. *(Corresponding author: Shuaicheng Liu and Bing Zeng.)*

S. Liu, B. Xu, S. Zhu, and B. Zeng are with the Institute of Image Processing, University of Electronic Science and Technology of China, Chengdu 610051, China (e-mail: liushuaicheng@uestc.edu.cn; eezeng@uestc.edu.cn).

C. Deng is with the School of Electrical and Information Engineering, Sichuan University, Chengdu 610065, China.

M. Gabbouj is with the Department of Signal Processing, Tampere University of Technology, Tampere 33720, Finland.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2016.2556587

that only rotational motion will be compensated for DDV segments. To the best of our knowledge, this is the first hybrid video stabilization system that merges the advantages of both the 2D and 3D methods (see [9] for demo videos).

The rest of this paper is organized as follows. Section II presents the related work. Section III analyzes the challenges of near-range video stabilization to motivate our work. Section IV discusses our hybrid method and presents the details on how 2D and 3D methods are, respectively, performed in the system. Section V presents some results with comparisons against the previous approaches. Section VI discusses a fast implementation method that does not involve structure-from-motion (SfM). Finally, Section VII concludes this paper.

II. Related Work

A. 2D Methods

The 2D methods estimate 2D transformations between consecutive video frames and concatenate these transformations to form a 2D camera path (also referred to as the camera trajectory). The input 2D camera path is shaky and undirected. Various types of low-pass filters have been designed to smooth the shaky path, resulting in a stabilized camera path that leads to a stabilized video. The existing 2D methods can be roughly classified into two types. The first one adopts linear 2D transformations, such as affine or homography models, and focuses on the design of smoothing algorithms. For instance, Matsushita *et al.* [3] adopted linear models and used simple Gaussian filtering to smooth the shaky camera path. Grundmann *et al.* [10] introduced cinematographical rules for the camera path design.

The other type adopts nonlinear 2D transformations (e.g., sets of homographies) and gives the priority to the motion representation between consecutive frames. For instance, to tackle the rolling-shutter effects [11], Grundmann *et al.* [7] estimated a nonlinear motion model (mixture of homographies) to represent the camera motion. Liu *et al.* [6] proposed to use bundled camera paths to model the camera motion, where the DDV scenes can be well represented. In particular, they adopted as-similar-as-possible warping to estimate a set of homographies according to a mesh structure and designed an algorithm to smooth bundled camera paths. Bai *et al.* [12] further introduced a user-assisted framework in which the user can adjust problematic frames to suppress artifacts.

B. 3D Methods

3D methods often rely on long feature tracks to reconstruct the 3D structures, including both 3D scene points and 3D camera motions, to stabilize the video. Buehler *et al.* [13] performed stabilization using the projective 3D reconstruction with an uncalibrated camera. Liu *et al.* [4] developed the first successful 3D video stabilization system. They reconstructed the 3D of the captured scene and adopted a mesh-based warping [14] for novel view rendering. More advanced image warping methods (e.g., [15]) could also be utilized to facilitate view synthesis. Zhou *et al.* [16] further introduced plane constraints to regularize the complanate structures in a video. For robustness, extra sensors are preferred to replace the vision-based 3D reconstruction. Liu *et al.* [17] used a depth camera and Smith *et al.* [18] adopted a light-field camera for both the camera motion estimation and the novel view rendering. More general sensors, such as gyroscopes in the mobile [1], [19], can be adopted to track the camera orientations, which shows significant improvement on many challenging sequences.

C. 2.5D Methods

The 2.5D methods directly smooth the trajectories of tracked features to stabilize the video. To this end, Liu *et al.* [5] smoothed some basis trajectories of the subspace formed by feature tracks. Goldstein and Fattal [8] utilized an epipolar transfer technique for stabilization. Wang *et al.* [20] represented each trajectory as a Bezier curve and performed smoothing with a spatial–temporal optimization. The 2.5D methods utilize partial 3D information embedded in the long feature tracks. It is different from our approach in that we directly combine the 2D and 3D methods to form a hybrid system.

In this paper, we adopt the bundled-path method [6] for 2D stabilization. Different from the methods in [10] and [21] that break the camera path into several segments, this paper directly divides the original input frames into different segments and smooths them by using either a 2D method or a 3D method.

III. Near-Range Video Analysis

Before getting into the details of our method, we would like to highlight the challenges of stabilizing near-range videos and analyze the underlaying reasons both physically and experimentally, which motivate the design of our method. In general, the 2D transformation cannot model the motion of near-range scenes. Although the 3D reconstruction can successfully recover the 3D camera motions, the render model in 3D approaches, such as as-rigid-as-possible warping [14], often fails during rendering. In the following, we first analyze the relationship between camera motions and scene depth, then perform some experiments to study the importance of smoothing rotational and translational motions, and finally describe what would be an idealized solution.

A. Exploring Physically

Fig. 1 shows several configurations regarding different camera placements and depth layers. Suppose that α and β denote two camera views, and the image is rendered from view α to view β. Dots A and B denote scene points. In Fig. 1(a), dots A and B stay approximately on the same plane, whereas they belong to different depth layers in Fig. 1(b)–(d) to mimic the depth variations in the near-range scenario.

In Fig. 1(a), two cameras only have translational motion. The captured pixels a and b in two views have the same spatial order (a is on top of b in both views), which is renderable by shrinking or stretching pixels between a and b. In Fig. 1(b), dots A and B belong to different depth layers. However, we can still obtain the same order of a and b in

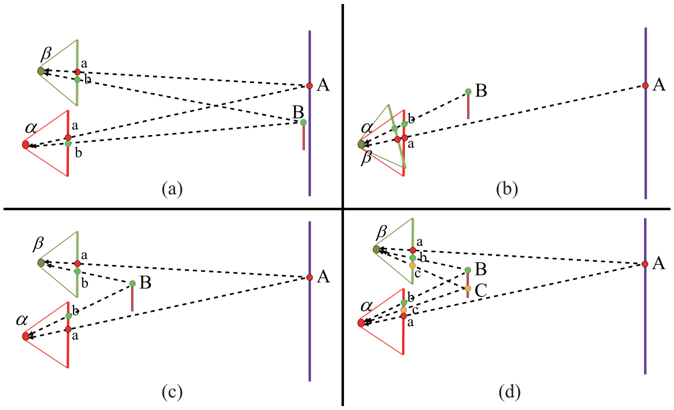


Fig. 1. α refers to the original shaky camera position and β denotes a stabilized camera position. A and B denote two scene points. Here, we need to render image from view α to view β. (a) Only translational camera motion where the scene dots A and B are approximately located on the same plane. (b) Rotational camera motion of near-range scenario. (c) Translational camera motion of near-range scenario. (d) Similar configuration as (c), with an additional point C be involved, showing a mixed render order bca to abc.

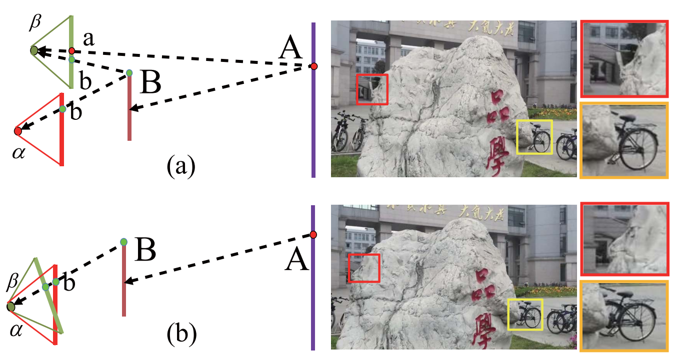


Fig. 2. (a) When a camera undergoes a translational motion in a near-range scenario, dot A cannot be observed in view α but appears in view β. If we want to render the view from α to view β, we need to find out the pixel value of a somewhere. (b) When cameras only have rotational motions, dot A cannot be viewed in both cameras. On the right-hand side, a real example shows the occlusion under the situation of (a).

two views, because cameras only undergo a rotational motion. In Fig. 1(c), the order is reversed (pixel a is originally below pixel b in view α; after warping, it is on top of pixel b in view β) when cameras have translational motion and dots A and B belong to different depth layers. The situation becomes even worse when an additional point C is involved in Fig. 1(d). It produces a mixed order, i.e., bca to abc. A simple shrinking or stretching cannot satisfy the purpose in this case. It is beyond the scope of the conventional image-based rendering, such as mesh warping. In this paper, we focus on the situations of Fig. 1(a) and (b).

Fig. 2(a) shows another scenario, in which the scene point A is occluded by the frontal depth layer and cannot be observed by camera α, but can be observed by camera β. To render view from α to β, we need to create the value of a. From β to α, we need to remove a. Both are challenging cases in image-based rendering, especially when pursuing a temporal consistent result. Meanwhile, when cameras only have a rotational motion, dot A either cannot be seen by both the cameras [Fig. 2(b)], or can be seen by both cameras [Fig. 1(b)], thus not introducing any problems.

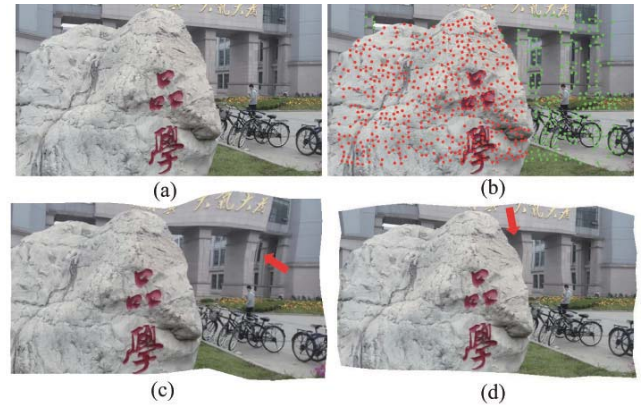


Fig. 3. Both 2D [6] and 3D [4] methods fail to stabilize near-range videos. (a) Original frame. (b) Detected feature points where background features are shown in green and foreground features are shown in red. (c) Result of 2D method. (d) Result of 3D method.

The right-hand side of Fig. 2 shows a real scenario corresponding to (a). The red and yellow rectangle regions show the occlusion of the stairs and bicycle wheel, respectively. Both are hard to synthesize or remove under the view changes. The aforementioned wobble artifacts are due to the failure of image-based rendering. The occluded content should appear or disappear when the camera position is changed. The situation becomes worse when contents occluded by objects are close to the camera. However, image warping used in video stabilization is not sufficient to handle this problem. On the other hand, Figs. 1(b) and 2(b) show that the compensation of only rotational motion may be indicative of a solution.

B. Wobble Effects

Having explored the near-range phenomenon physically, let us now discuss it experimentally. Fig. 3 shows an example. The frame resolution for this example is 640×360. Both the 2D [6] and 3D [4] methods fail to generate successful results. Here, we use the code of [6] for the 2D method, whereas we use our own implementation of [4] for the 3D method. Fig. 3(c) shows the result of the 2D method [6]. It can be seen that some image structures (e.g., the one indicated by the red arrow) are severely distorted. We compute the registration error between the current frame and its previous frame. The error is 5.17 pixels, which is extremely high compared with normal errors ($\sim$0.3–0.6, obtained from 20 successful examples), indicating the inaccuracy of motion estimation. Note that, although the bundled-path method utilizes multiple homographies to model the camera motion, it is confined to the parallax issue within CDV scenes, whereas the DDV scenes are beyond its scope.

Fig. 3(d) shows the result of the 3D method [4]. The camera motion has been successfully recovered by 3D reconstruction in this example. However, some image structures are still distorted (again, one example is indicated by the red arrow) due to the failure of the mesh-based render model. The original frame is transformed to its stabilized position by mesh warping guided by the motion vectors induced from each feature point.

The mesh-based render model, which is widely adopted in video stabilization, can only tolerate a small amount of motion difference within a frame during warping. Empirically, the difference is around one pixel, beyond which distortions would occur. Here, we collect features from foreground (red) and background (green), as shown in Fig. 3(b), and compute their average motion vectors. We found that the foreground points move 20.28 pixels toward their stabilized position, while the background points move only 15.58 pixels. This largely inconsistent movement causes the failure.

C. 3D Rotational and Translational Motion

In 3D stabilization, both the rotational and translational motions are compensated. Here, we design an experiment to illustrate the influence of smoothing each component. We examine ten videos collected from the project pages of the previous works [4], [6], [8], [17]. We smooth only camera rotations and compare the results with the results of smoothing both components. We evaluate the wobble effects and the stability of the results. Both scores are calculated in the same way as [6].

1) Wobble Score: We match features between input and output frames, and fit global homographies $B(t)$ between them. The anisotropic scaling of $B(t)$ measures the distortion, which can be computed by the ratio of the two largest eigenvalues of the affine part of $B(t)$. Each frame has a wobble score, among which we choose the worst one as the final wobble score. The best value (with no wobbling) is equal to 1, and a lower score denotes less wobbling.

2) Stability Score: We calculate the stability score to evaluate the smoothness of a video that is stabilized by different methods. We track the features of the stabilized video and then analyze these feature tracks in the frequency domain. We take a few of the lowest frequencies (second to sixth, dc is excluded) and calculate the energy percentage they occupy on the whole frequency domain. We take the average of all tracks (length should be >20 frames) as the final score. A larger value means the concentration of energies on lower frequencies, thus leading to a higher stability.

Fig. 4 shows the results of all the chosen videos. Two different smoothing strategies achieve similar performances in terms of both scores. Smoothing of translational motion does not exhibit significant improvement for stability, which can be visually confirmed in supplementary videos.

We then capture ten near-range videos and conduct the same experiment. Fig. 5 shows the results. Two smoothing strategies achieve similar stability scores. However, smoothing of translational motion yields larger wobble scores (corresponding to severe distortions).

D. Idealized Solution

Based on the above analysis and two experiments, we make two important observations.

1) *The Rotational Motion Is the Main Source of Instability:* According to stability scores in Fig. 5, the stability is barely improved when compensating translational motion.

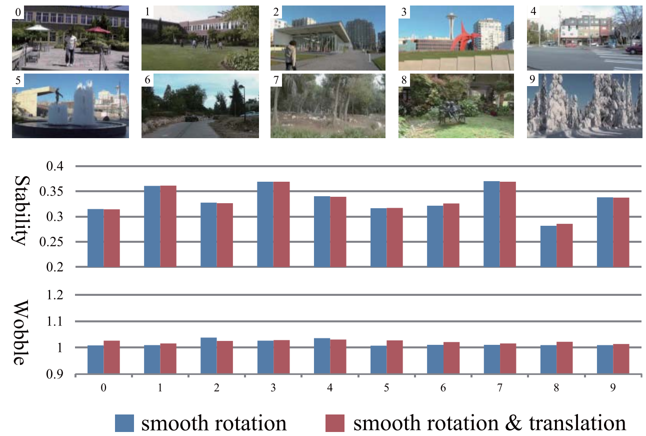


Fig. 4. Analysis of smoothing rotational and translational motion. The first row shows the thumbnails of the video. The second row and the third row show the stability and wobble scores of the results by two different smoothing strategies.

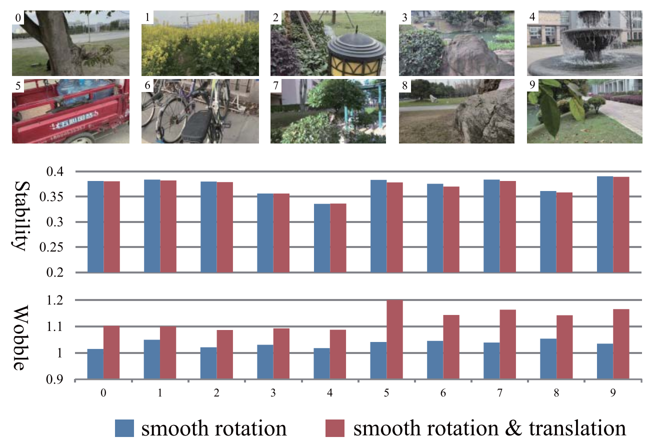


Fig. 5. Analysis of smoothing rotational and translational motion on near-range videos. The first row shows the thumbnails of the video. The second row and the third row show the stability and wobble scores of the results by two different smoothing strategies.

2) *The Smoothing of Translational Motion is the Main Source of Wobble Distortions:* According to wobble scores of Fig. 5, the distortion is deteriorated when compensating translational motion.

Fig. 6 shows an idealized solution. The curve represents objects that stay close to the camera. Red and green triangles denote the original shaky and stabilized camera positions, respectively. In Fig. 6(a), both rotational and translational motions are smoothed, whereas in Fig. 6(b), only rotational motions are smoothed in DDV segments. Although, the stabilized camera path in Fig. 6(a) is smoother than Fig. 6(b), it would produce a problematic result due to the render difficulties at DDV segments. On the other hand, both the motions in CDV segments can be smoothed freely. An idealized solution is to locate DDV and CDV clips correctly to apply different smoothing strategies accordingly.

E. Distinguishing DDV and CDV

As discussed in Figs. 1 and 2, the DDV segments will cause render issues during the view synthesize. We can warp frame t to frame $t+1$ by some parametric models

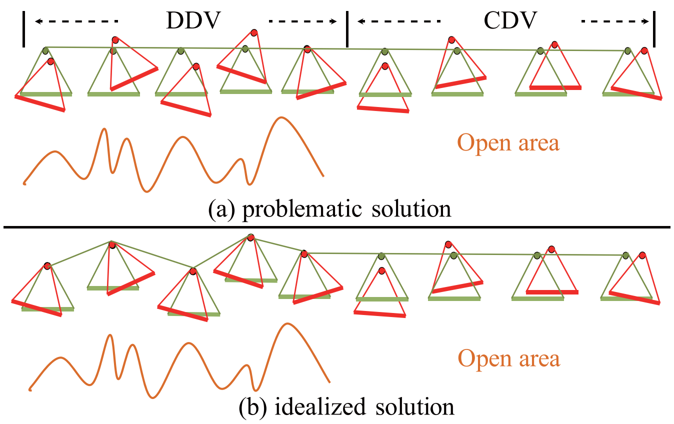


Fig. 6. Traditional solution versus our proposed solution. The curve denotes near-range objects and space without the curve denotes open area. Red and green cameras denote original shaky and stabilized positions, respectively. (a) Adjusting of camera center during near-range footage may introduce rendering problem, leading to wobble effects. (b) Our proposed solution only compensates rotational motion on near-range sequences.

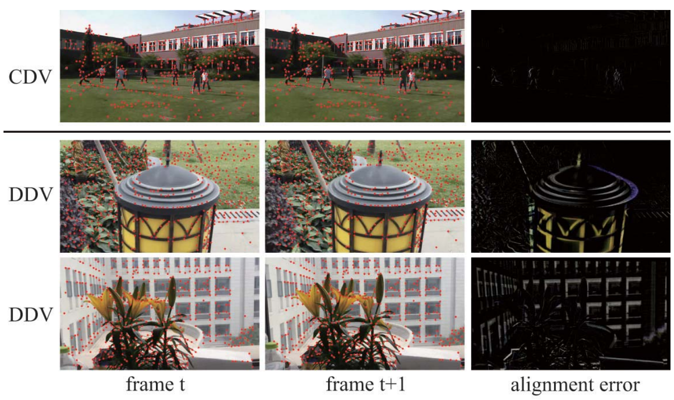


Fig. 7. Comparison of the single homography-based registration error between CDV and DDV. First two columns show two consecutive frames t and $t+1$, with matched feature points printed as red dots. A global homography is estimated to align two frames. The third column shows the error image after alignment. DDVs correspond to larger errors as compared with CDVs.

(e.g., homography). As parametric models cannot model the geometric transformation between two frames with near-range objects under camera translation, they often produce particularly high alignment errors, which indicate the presence of DDVs. Therefore, the alignment errors can be used as cues to distinguish DDV and CDV clips. Fig. 7 shows an example. We match the feature points between two consecutive frames, and fit a homography model between them. Then, we warp frame t to frame $t+1$ by the homography. The differences are calculated between warped frame t and frame $t+1$. A lower error infers a better alignment. A single homography is most suitable for scenes without large depth. Obviously, CDV is better positioned to a single homography than DDV, yielding a lower error. The example of CDVs is associated with the case shown in Fig. 1(a), whereas DDVs are with situations in Figs. 1(b) and 2(b). Equivalently and practically, we calculate fitting errors of feature points,[1] instead of image differences. For videos with a resolution of 640×360, the fitting errors of CDV fall into the range of 0.3–0.6 pixels, beyond which the clips should be considered as DDV segments. Note that different registration methods can be adopted, such as mesh warping, which may lead to a different threshold.

[1]Feature matches on dynamic objects can be rejected by RANSAC during homography estimation.

IV. Hybrid Stabilization

Our system pipeline is shown in Fig. 8. Given a shaky video [Fig. 8(a)], we first apply a traditional single-path video stabilization method. Then, we divide the original video into CDV segments [Fig. 8(b)] and DDV segments [Fig. 8(c)], based on the analysis of fitting error and the cropping ratio. Next, 3D stabilization is applied to the DDV segments [Fig. 8(d)], following which we obtain a partially stabilized video [Fig. 8(e)] by replacing the DDV segments with stabilized segments. The final result [Fig. 8(f)] is produced via a constrained optimization. In the following, we describe each component in detail.

A. Video Partition

There are two basic criteria when we choose stabilization methods.

1) When a frame contains fast camera motions (e.g., rapid rotation and fast zooming), we try to stick to the 2D approach for preserving the original motion.
2) When a frame (free from fast camera motions) is largely occupied by near-range objects, we are inclined to use the 3D approach to reduce wobbling.

In order to detect these two scenarios, we apply feature matches between neighboring frames and calculate homographies together with fitting errors between neighboring frames. Next, we use those homographies to conduct a single path stabilization [3] (a simple Gaussian smooth) to prestabilize the input video. We then measure the cropping ratio of each stabilized frame. The cropping ratio is the ratio of the remaining area (after stabilization) over the original frame area. Typically, when frames contain rapid motions, the stabilized results would have large empty regions, thus yielding a small cropping ratio; when consecutive frames contain near-range objects, the corresponding fitting error is large.

Therefore, we first segment the video into clips with fast-motion and fast-motion-free clips according to the cropping ratio. In our implementation, when the cropping ratio is lower than 0.75, we mark the frame with fast motion. We then only examine the fitting error of fast-motion-free clips. In particular, when the error exceeds 0.6, we mark the frames as DDV scenes. Notably, to preclude the noise during segmentation, we use erosion and expansion algorithms to merge the clips with unsatisfied length into their neighboring clips.

B. Rotational Motion for 3D Stabilization

After SfM, the camera projection matrix $P = K[R| - C]$ and 3D points are recovered, where K is the intrinsic, R is the rotation, and C is the camera center. The traditional 3D stabilization method [4] smooths both camera rotation and

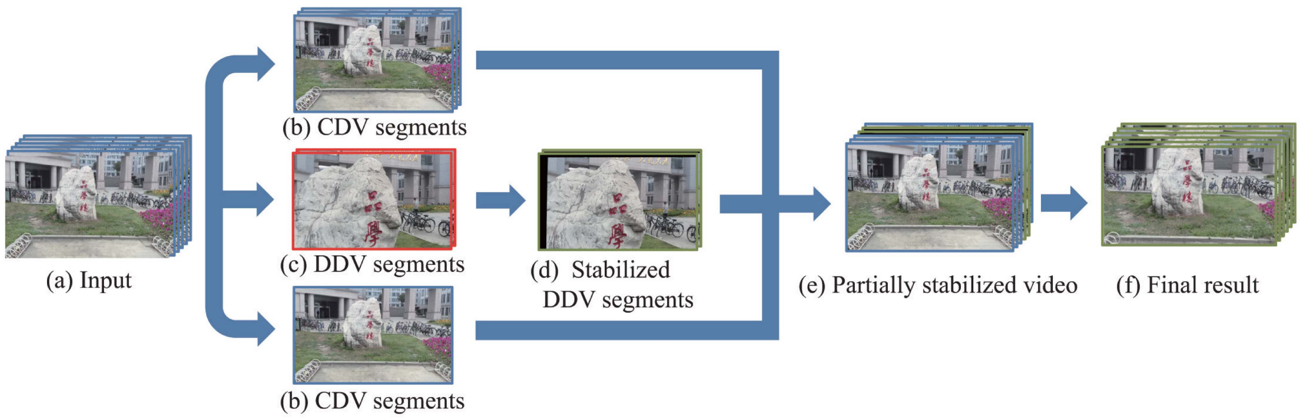


Fig. 8. Our system pipeline. (a) Input video is partitioned into (b) CDV segments and (c) DDV segments. DDV segments are stabilized by smoothing their camera rotational motion using the 3D stabilization method. (d) Stabilized DDV segments are inserted back to the original video, which generates (e) partially stabilized video. The bundled-path stabilization is applied to the CDV segments contained in (e), where the stabilized DDV segments are encouraged to keep their positions, while the CDV segments are adjusted to merge with DDV segments seamlessly to produce (f) final result.

translation to generate a new set of camera projection matrices. Projecting the 3D points through the original and smoothed projection matrices gives each frame a set of 2D point correspondences between its original and stabilized positions. Then, either a best-fitting homography or multiple mesh warping matrices are estimated from these correspondences, which brings the shaky frame to its stabilized novel view. As discussed in Section III-C, only rotations should be smoothed to avoid wobbling. Thus, the rotation R becomes R' after smoothing, while the camera center is invariably fixed. When a camera rotates only around its optical center, the induced transformation is an infinite homography [22], which is defined as

$$H = KR(R')^{-1}K^{-1}. \quad (1)$$

The previous best-fitting homography or the mesh warping matrix is replaced with this infinite homography, which is irrelevant of the 3D points. Thus, all the useful structures provided by SfM are the camera rotation R and the intrinsic K. Let us denote this infinite homography by $\hat{D}(t)$,[2] with t standing for the time index.

C. Revisit Bundled-Path Stabilization

We adopt the bundled-path method as our baseline for the 2D stabilization. For the completeness, we first describe the bundled-path method briefly. Later, we will modify it for a constrained optimization.

1) Smooth a Single Path: The homography F is estimated between neighboring frames in the original video. The camera path is defined as a concatenation of these homographies: $C(t) = F(t)F(t-1)\dots F(1)F(0), F(0) = I$. Given the original path $\mathbf{C} = \{C(t)\}$, the smoothed path $\mathbf{P} = \{P(t)\}$ is obtained by minimizing the energy

$$\mathcal{O}(\{P(t)\}) = \sum_t \|P(t) - C(t)\|^2 + \sum_t \left(\lambda_t \sum_{r\in\Omega_t} \omega_{t,r}(C) \cdot \|P(t) - P(r)\|^2 \right) \quad (2)$$

[2]We use the hat notation $\hat{(\cdot)}$ to denote infinite homography obtained from the 3D camera rotation.

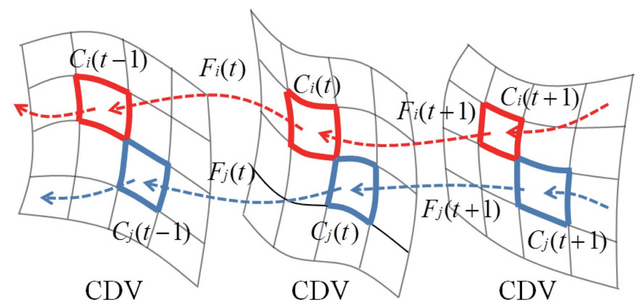


Fig. 9. Smoothing of bundled paths: it can process frames with CDVs.

where Ω_t denotes the neighborhood at frame t. The strength of smoothing is controlled by λ_t. The smoothing kernel $w_{t,r}$ is a bilateral smoothing weight. The output video is obtained by applying a transform B_t to the input video, which is defined as $B(t) = C^{-1}(t)P(t)$.

2) Smooth Bundled Paths: The algorithm divides each frame into 16×16 cells and estimates camera paths for each cell. The estimation is based on the mesh warping, which warps the frame t to frame $t-1$. Fig. 9 shows the initial state of bundled paths before optimization. All the paths are smoothed together by a space-time optimization

$$\sum_i \mathcal{O}(\{P_i(t)\}) + \sum_t \sum_{j\in N(i)} \|P_i(t) - P_j(t)\|^2 \quad (3)$$

where $N(i)$ includes eight neighbors of the cell i.

D. Joint Stabilization

In this section, we describe how the CDV segments are stabilized and stitched to the DDV segments. Similarly, we begin by presenting the method for a single path and then extend it to bundled paths. We define a control function $f(t)$, which takes the frame index t as input and returns a weight as output. The output of $f(t)$ is determined by the location of the input frame

$$f(t) = \begin{cases} 0, & t \in \text{CDV segment} \\ \tau, & t \in \text{buffer} \\ 1, & \text{otherwise.} \end{cases} \quad (4)$$

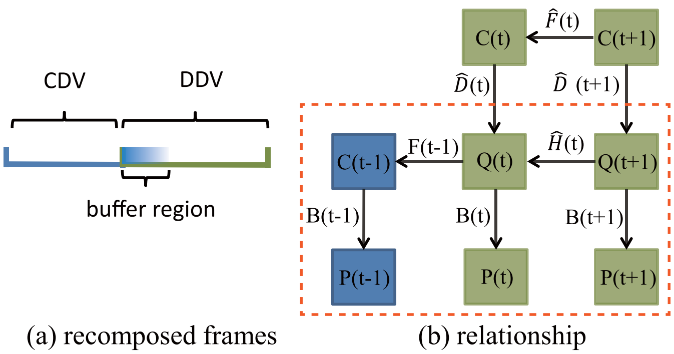


Fig. 10. (a) Buffer region in recomposed frames enabling a smooth transition between CDV and DDV clips. (b) Relationship between CDV and DDV segments.

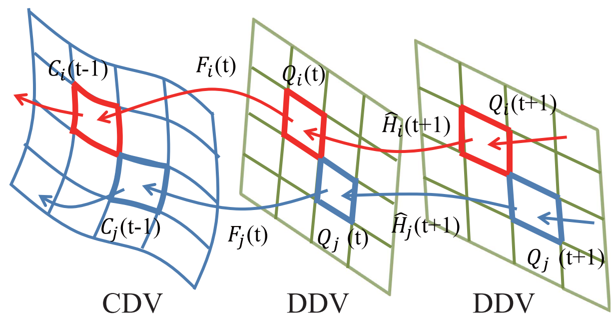


Fig. 11. Constrained optimization. The DDV segments are stabilized by infinite homographies previously, leading to planar shapes. The stabilized DDV paths are denoted by $Q_i(t)$. Together with the original paths $C_i(t)$, they generate a partially stabilized video, which is the input to the optimization.

Fig. 10(a) shows the three possible categories that a frame may belong to: the CDV segment, the DDV segment, and the buffer region. The buffer region is located at the boundary of the DDV segment. The frame in the buffer takes the value of τ, which is the distance between the frame and the boundary of the DDV segment. The distance is normalized to $(0, 1)$. The buffer region contains 20 frames, within which the smoothing method is hybrid.

Fig. 10(a) also shows the relationship between different paths. Similar to the notation of Section IV-C1, the original camera path and the optimally-smoothed camera path are still defined as $C(t)$ and $P(t)$. Now, we have an intermediate camera path $Q(t)$, which is transformed from the original path $C(t)$ by an infinite homography $\hat{D}(t)$. In particular, $C(t+1) = \hat{F}(t)C(t)$ and $Q(t+1) = \hat{H}(t)Q(t)$. $\hat{F}$ and $\hat{H}$ can be derived according to (1) from rotation matrices between neighboring frames. The optimization is performed within the dashed box, as shown in Fig. 10(b).

1) Optimize a Single Path: We would like to optimize the partially stabilized videos while maintaining the 3D stabilization results. The CDV segments are optimized in a similar way as (2), and the energy function is defined as

$$\begin{aligned}\mathcal{O}(\{P(t)\}) &= \sum_t (1-f(t))\|P(t)-C(t)\|^2 \\ &\quad + \sum_t f(t)\|P(t)-Q(t)\|^2 \\ &\quad + \sum_t \left(\lambda_t \sum_{r\in\Omega_t} \omega_{t,r}(C)\cdot\|P(t)-P(r)\|^2\right). \quad (5)\end{aligned}$$

We add an additional term, $\sum_t f(t)\|P(t)-Q(t)\|^2$, to constrain the optimized paths in DDV segments, such that these paths would stay close to the position $Q(t)$ to avoid wobble distortions. The energy is quadratic and can be solved by any linear system. Similar to [6], we use a Jacobi-based iterative solve [23]. The update rule is defined as

$$\begin{aligned}P^{(\xi+1)}(t) &= \frac{1}{\gamma}(1-f(t))C(t) + \frac{1}{\gamma}f(t)Q(t) \\ &\quad + \sum_{r\in\Omega_t, r\neq t} \frac{2\lambda_t\omega_{t,r}}{\gamma} P^{(\xi)}(r) \quad (6)\end{aligned}$$

where $\gamma = 1 + 2\lambda_t \sum_{r\in\omega, r\neq t} \omega_{t,r}$ and ξ is the iteration index. The final result is obtained by applying transformation $B(t)$ to CDV frames and chained transformation $\hat{D}(t)B(t)$ to DDV frames.

2) Optimize Bundled Paths: Each frame is divided into 16×16 cells. An additional constraint is provided to ensure the neighboring cells to have similar optimized paths. The energy function is defined as

$$\begin{aligned}\mathcal{O}(\{P(t)\}) &= \sum_t (1-f(t))\sum_i \|P_i(t)-C_i(t)\|^2 \\ &\quad + \sum_t f(t)\sum_i \|P_i(t)-Q_i(t)\|^2 \\ &\quad + \sum_t\sum_i \left(\lambda_t \sum_{r\in\Omega_t} \omega_{t,r}(C)\cdot\|P_i(t)-P_i(r)\|^2\right) \\ &\quad + \sum_t\sum_i\sum_{j\in N(i)} \|P_i(t)-P_j(t))\|^2. \quad (7)\end{aligned}$$

The update rule for the Jacobi-based iterative solver is

$$\begin{aligned}P_i^{(\xi+1)}(t) &= \frac{1}{\gamma'}\Bigg((1-f(t))C_i(t) + f(t)Q_i(t) \\ &\quad + \sum_{r\in\Omega_t, r\neq t} 2\lambda_t\omega_{t,r}P_i^{(\xi)}(r) + \sum_{j\in N(i), j\neq i} 2P_j^{\xi}(t)\Bigg) \quad (8)\end{aligned}$$

where

$$\gamma' = 1 + 2\lambda_t \sum_{r\in\omega, r\neq t} \omega_{t,r} + 2N(i) - 1.$$

Fig. 11 shows the current configuration of bundled paths before optimization. Note that, different from Fig. 9, the frames within DDV clips are stabilized by a transformation of an infinite homography obtained from the previous steps. Thus, the stabilized frames have a rigid shape, which means all the subpaths $Q(i)$ share the same value in the beginning. During the optimization, the subpaths may take different values, because the nonidentical bundled paths of CDV regions can be propagated to influence the DDV regions. This propagation within buffer regions enables a smooth transition between the motion of CDV and DDV segments.

(a) (b) (c)

Fig. 12. With and without the smoothness term (the third term) in (5). (a) Video volume. We illustrate the temporal image that corresponds to the red rectangle. (b) With the smoothness term. (c) Without the smoothness term.

(a) (b) (c)

Fig. 13. Failure cases reported in (a) epipolar [8] and (b) and (c) SteadyFlow [2].

Finally, we would like to emphasize the importance of the third term involved in (5) and (7). Fig. 12(a) shows a video volume. To illustrate the temporal stability, we focus on the volume portions marked by the red rectangle. If all the terms are considered [as in (5) and (7)], the result is fully stabilized, as shown in Fig. 12(b). However, if the third term is eliminated, the result is stabilized only partially, as shown in Fig. 12(c). Comparing these results, we can see that the third term influences the temporal smoothness significantly.

V. Experimental Results

We run our method on an Intel i7 2.3-GHz CPU and 4 GB RAM. We divide the video frame to 16 $\times$ 16 cells. When processing CDV clips, our system takes 400 ms to process a frame with a resolution of 720 pixels, during which the optimization only takes 20 ms. For DDV segments, we use Voodoo SfM [24] for 3D reconstruction. It normally takes 1–2 s to process a frame.

Our approach is targeted at stabilizing near-range videos that contain DDV clips. For videos without DDV scenes, our method is identical to the bundled-path stabilization. Therefore, any video that can be handled by the bundled-path method can also be processed by our approach. In other words, our result is, if not superior, at least comparable with the results obtained by using the bundled-path method. Thus, we focus our evaluation on videos that contain DDV clips. It is not surprising that almost all the failure cases reported in the previous methods are near-range videos.

A. Comparison With Previous Methods

Fig. 13 compares our method with the failure case reported in the epipolar method [8] and the SteadyFlow method [2]. Fig. 14 compares our method with the bundled-path method on near-range videos that are captured by ourselves. Note that the adaptive smoothing strategy in the bundled-path method will skip DDV segments, and thus keep the original frames unaltered in the result. Though the original frames contain no distortions, it is shaky. To facilitate the comparison, we turn OFF the adaptive smoothing and force the bundled-path

Fig. 14. Comparison with the bundled path [6] on near-range videos.

Fig. 15. Comparison with the spatially and temporally optimized method [20] on near-range videos. Our method consistently generates wobble-free videos.

method to stabilize the DDV segments. As can be seen from the accompanied videos, the result of the bundled-path approach suffers from severe wobble distortions during the DDV segments but becomes artifact-free as long as the video proceeds into CDV segments. Fig. 15 further compares our method with the spatial–temporal stabilization method [20], which enforces the spatial relationship of scene objects as a hard constraint to suppress distortion. However, we can still observe wobbling effects in their results.

B. Comparison With the State-of-the-Art Systems

We further compare our method with two well-known commercial systems on self-captured videos. The first one is the YouTube stabilizer, which is built upon the L_1-optimization [10] and the homography-mixture model [7]. We upload our videos onto the YouTube Web site and download the results that are automatically stabilized by the server. The other system is the Adobe After Effects CS6 Warp Stabilizer, which is based on the technology of subspace video stabilization [5]. As it is an interactive tool, we try our best to produce visually pleasant results. Fig. 16 shows the results. We examine 12 videos (mostly only with near-range objects) captured by ourselves. As illustrated, three approaches achieve comparable stabilization performances, with other two slightly better than ours on some cases. However, they meet the goal at the cost of the resultant in severe wobbling effects. For most DDV cases, their wobble scores exceed 1.1, which implies large perceptible distortions in the results. Notably, our method can handle dynamic near-range objects desirably with less distortion.

C. More Results

Fig. 17 shows some extra results. Each row shows an example. Arrows suggest the camera motion. The first example

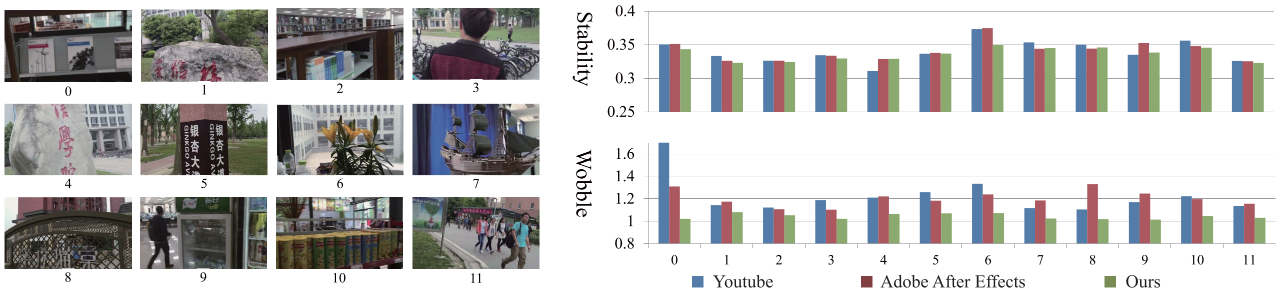


Fig. 16. Quantitative comparisons with the state-of-the-art systems (YouTube and Adobe After Effects) on near-range videos.

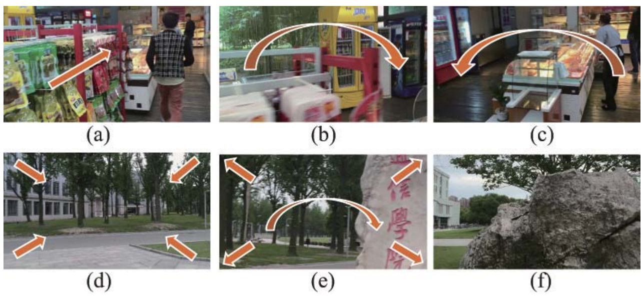


Fig. 17. Various camera motions with complex depth variations. Arrows: camera motion. (a) Camera moves forward. (b) Camera rotates from left to right. (c) Camera rotates from right to left. (d) Camera zooms in. (e) Camera rotates from left to right while zooming out. (f) Camera films a near-range object.

is an indoor case. The camera first moves forward, then it rotates from left to right quickly, and finally rotates back. In the second example, the camera contains a quick zooming initially, then it rotates from left to right during zooming out, and finally it films a near-range stone. In these examples, both the camera motion and the scene depth are complex. Our method can still handle these challenging cases successfully.

VI. Fast Implementation Without SfM

In this section, we present some details regarding the computation of rotation matrices to facilitate a practical implementation.

A. Rotation Recovery

The recovery of rotations from 3D reconstruction is not only computationally expensive but also not robust for many consumer-level videos. Equation (1) requires the rotation matrix R and the intrinsic K. There are various ways to estimate camera rotations, such as [25] and [26]. Here, we explore the classical method described in [22] and adapt it to our implementation. Given two frames, we first match features between them. Next, we estimate the fundamental matrix F with random sample consensus for outlier removal. Given the intrinsic matrix K, the essential matrix can be derived as: $E = K^T F K$ ((Eq. 9.12) of [22]). The camera can be extracted from the essential matrix up to a fourfold ambiguity, which gives four projection matrices that correspond to two rotations and two camera centers.

Let us ignore camera centers and focus on the two possible rotations. Taking the singular value decomposition of E gives UDV^T. The two candidates are ((Eq. 9.14) of [22])

$$R_1 = UWV^T \quad R_2 = UW^T V^T \tag{9}$$

where

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{10}$$

Notably, if one of the frames has an identity rotation, the other frame will have two possibilities. As described in [22], the triangulated 3D point should stay in front of the camera in order to select the correct project matrix among four candidates. Here, to select the correct rotation, we check the reprojection error induced by rotations between two frames. Two infinite homographies are obtained based on (1)

$$H_1 = K R_1 K^{-1} \quad H_2 = K R_2 K^{-1}. \tag{11}$$

The correct rotation is the one with the smallest reprojection error $\|x_i - H_{1,2}x_i'\|$. This approach requires the camera intrinsic K, which can be obtained by a precalibrated camera or by running a few frames of SfM. This way, we can accomplish each frame in about 25 ms, including FAST feature detection [27], Kanade-Lucas-Tomasi (KLT) tracker [28], and rotation estimation. However, it is not as accurate as the full SfM, in which the bundle adjustment is adopted for the improved accuracy of both camera positions and 3D points.

B. Adobe AE Plugin

Based on the fast rotation recovery, we implement our hybrid method into Adobe After Effects CC2015 as a plugin. We use the plugin Software Development Kit and OpenCV libraries. In particular, we use FAST features coupled with KLT optical flow tracker for feature computation, which gives a fast speed and good enough quality among other choices. We will release this plugin to the public after some refinement and stress testing.

Fig. 18 shows the interface. We can simply drag the video into the queue and drag the plugin onto the video for stabilization. We screen captured the process of stabilizing a near-range video and compared it with Adobe's built-in stabilization feature warp stabilizer (see the demo video [9]).

VII. Discussion

Our system is built upon the bundled-path stabilization method. By absorbing its merits, we can handle various types of videos, including videos with quick rotation, videos with rapid zooming, and videos captured during walking, running, riding, driving, and so on. More importantly, our system goes a step further, which is the handling of scenes with

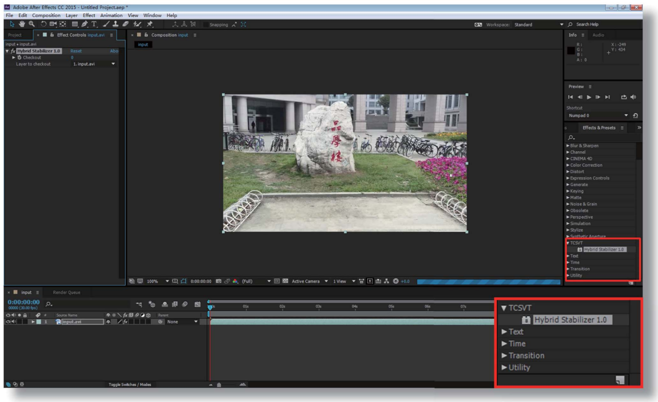


Fig. 18. Plugin, named hybrid stabilizer, has been implemented into Adobe After Effects CC2015.

near-range objects. The 3D reconstruction is applied only when necessary, which improves the robustness as well as the speed. However, there are several points we would like to clarify and discuss.

A. Dynamic Objects

When the scene contains dynamic objects of small size, 2D stabilization is performed in our approach. If it is a near-range object that consistently occupies over half the area of a frame, we should accordingly use the 3D approach. However, since the 3D reconstruction would fail under such tough situations, our system will mark these clips as CDV segments and apply the 2D stabilization method. The adaptive smoothing strategy adopted in the bundled-path stabilization will skip them and the result will be the original frames. This is so far the best way to avoid artifacts. Notably, in Fig. 16 (Case 3), we show a video in which the camera tracks a walking character who occupies a large area. Fortunately, it is still not large enough to cause the failure of 3D reconstruction, thus our method can process it successfully.

B. Gyroscope

Recall that all information we need to perform 3D stabilization is the camera rotation for each frame. To obtain rotations, of course, there are some alternative ways other than SfM, among which the gyroscope would be the best choice. It has been widely adopted in smartphones, and some works [1], [19] stabilize videos purely using the gyroscope information. Alternatively, the devices, such as Kinect [17] or light-field camera [18], would be some other candidates. Our system targets at stabilizing videos that have already been captured without any knowledge of sensor data. However, our method is compatible with any of these hardwares, as long as we are provided the camera orientation during filming.

C. Rolling Shutter

Most mobile cameras use a rolling-shutter sensor whereby each horizontal scanline of pixels is sequentially exposed and read out, resulting in a distorted image. The bundled-path method can correct the rolling-shutter effects for CDV clips. However, if DDV clips contain large distortions caused by rolling-shutter effects, the SfM would fail. This is one of our limitations. In these cases, we mark the DDV clips as CDV clips, which finally roll back to the original footage.

D. Translational Jitter

When objects are close to the camera, our system marks these clips as DDV clips and only compensates rotational motions. Thus, for DDV clips, the translational jitters remain in the stabilized video. Actually, we do want to remove the translational jitters, but not at the cost of bringing in wobble artifacts. The wobble is mainly caused by the occlusions, where the content appears and disappears when the camera position is changed. The compensation of translational motion for DDVs requires additional techniques, such as image inpainting techniques, to remove objects that should disappear in the new view, or synthesis technique to bring in objects that should appear in the new view. Both cases are challenging to video stabilization, which is considered another limitation. Compared with wobble artifacts that distort the image contents, the jitters caused by translations are of less importance. Moreover, compared with traditional bundled paths, which roll back to the original footage under such circumstance, our method can remove rotational jitters, rendering a more pleasant result.

E. Future Works

The translational jitter remains in the DDV segments after our stabilization. Although it is not as significant as rotational jitters, it is a source of shakiness that should be removed for an improved stability, especially for videos where translational jitters dominate. One possibility is to segment foregrounds and backgrounds, either manually [29] or automatically [30], and apply advanced spatial–temporal inpainting techniques [31], [32] to fill in holes caused by view point shift. We leave this as our future work. Besides, we would also like to integrate external sensors for acquiring the camera orientations.

VIII. Conclusion

We have presented a hybrid approach that combines the strength of 2D methods and 3D methods for near-range video stabilization. Our approach enjoys the merits of 2D methods to tackle various types of camera motions, while also benefiting from 3D methods to deal with challenging DDV scenarios. Input videos are partitioned into several clips, which are then stabilized by either the 2D bundle-path method or the 3D method developed on the basis of some previous traditional methods. The final result is generated by a constrained optimization that seamlessly stitches different clips. To improve the efficiency and robustness, we have also explored a fast implementation that does not require the full SfM for the rotation recovery.

References

[1] S. Bell, A. Troccoli, and K. Pulli, “A non-linear filter for gyroscope-based video stabilization,” in *Proc. ECCV*, 2014, pp. 294–308.

[2] S. Liu, L. Yuan, P. Tan, and J. Sun, “SteadyFlow: Spatially smooth optical flow for video stabilization,” in *Proc. CVPR*, Jun. 2014, pp. 4209–4216.

[3] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum, "Full-frame video stabilization with motion inpainting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1150–1163, Jul. 2006.

[4] F. Liu, M. Gleicher, H. Jin, and A. Agarwala, "Content-preserving warps for 3D video stabilization," *ACM Trans. Graph.*, vol. 28, no. 3, 2009, Art. no. 44.

[5] F. Liu, M. Gleicher, J. Wang, H. Jin, and A. Agarwala, "Subspace video stabilization," *ACM Trans. Graph.*, vol. 30, no. 1, 2011, Art. no. 4.

[6] S. Liu, L. Yuan, P. Tan, and J. Sun, "Bundled camera paths for video stabilization," *ACM Trans. Graph.*, vol. 32, no. 4, 2013, Art. no. 78.

[7] M. Grundmann, V. Kwatra, D. Castro, and I. Essa, "Calibration-free rolling shutter removal," in *Proc. ICCP*, 2012, pp. 1–8.

[8] A. Goldstein and R. Fattal, "Video stabilization using epipolar geometry," *ACM Trans. Graph.*, vol. 31, no. 5, 2012, Art. no. 126.

[9] S Liu *et al.*, "A hybrid approach for near-range video stabilization," *The Demo Video*. [Online]. Available: https://figshare.com/articles/A-Hybrid-Approach-for-Near-Range-Video-Stabilization/2742334

[10] M. Grundmann, V. Kwatra, and I. Essa, "Auto-directed video stabilization with robust L1 optimal camera paths," in *Proc. CVPR*, 2011, pp. 225–232.

[11] S. Baker, E. Bennett, S.-B. Kang, and R. Szeliski, "Removing rolling shutter wobble," in *Proc. CVPR*, 2010, pp. 2392–2399.

[12] J. Bai, A. Agarwala, M. Agrawala, and R. Ramamoorthi, "User-assisted video stabilization," *Comput. Graph. Forum*, vol. 33, no. 4, pp. 61–70, 2014.

[13] C. Buehler, M. Bosse, and L. McMillan, "Non-metric image-based rendering for video stabilization," in *Proc. CVPR*, 2001, pp. II-609–II-614.

[14] T. Igarashi, T. Moscovich, and J. F. Hughes, "As-rigid-as-possible shape manipulation," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 1134–1141, Jul. 2005.

[15] P. Kaufmann, O. Wang, A. Sorkine-Hornung, O. Sorkine-Hornung, A. Smolic, and M. Gross, "Finite element image warping," *Comput. Graph. Forum*, vol. 32, no. 2, pp. 31–39, May. 2013.

[16] Z. Zhou, H. Jin, and Y. Ma, "Plane-based content preserving warps for video stabilization," in *Proc. CVPR*, 2013, pp. 2299–2306.

[17] S. Liu, Y. Wang, L. Yuan, J. Bu, P. Tan, and J. Sun, "Video stabilization with a depth camera," in *Proc. CVPR*, 2012, pp. 89–95.

[18] B. M. Smith, L. Zhang, H. Jin, and A. Agarwala, "Light field video stabilization," in *Proc. ICCV*, 2009, pp. 341–348.

[19] A. Karpenko, D. E. Jacobs, J. Baek, and M. Levoy, "Digital video stabilization and rolling shutter correction using gyroscopes," Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA: Tech.Rep. CSTR 2011-03, 2011.

[20] Y.-S. Wang, F. Liu, P.-S. Hsu, and T.-Y. Lee, "Spatially and temporally optimized video stabilization," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 8, pp. 1354–1361, Aug. 2013.

[21] M. L. Gleicher and F. Liu, "Re-cinematography: Improving the camera dynamics of casual video," in *Proc. 15th ACM Int. Conf. Multimedia*, 2007, pp. 27–36.

[22] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[23] I. N. Bronshtein and K. A. Semendyayev, *Handbook of Mathematics*. New York, NY, USA: Springer-Verlag, 1997.

[24] *Voodoo Camera Tracker: A Tool for the Integration of Virtual and Real Scenes*. [Online]. Available: http://www.viscoda.com/index.php/en/products/non-commercial/voodoo-camera-tracker

[25] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon, "Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment," *Int. J. Robot. Res.*, vol. 31, no. 1, pp. 63–81, 2012.

[26] D. Martinec and T. Pajdla, "Robust rotation and translation estimation in multiview reconstruction," in *Proc. CVPR*, 2007, pp. 1–8.

[27] M. Trajković and M. Hedley, "Fast corner detection," *Image Vis. Comput.*, vol. 16, no. 2, pp. 75–87, 1998.

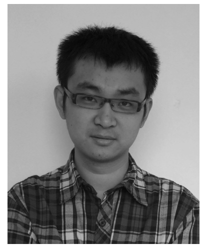
[28] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog. (CVPR)*, Jun. 1994, pp. 593–600.

[29] X. Bai and G. Sapiro, "A geodesic framework for fast interactive image and video segmentation and matting," in *Proc. ICCV*, 2007, pp. 1–8.

[30] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchical graph-based video segmentation," in *Proc. CVPR*, 2010, pp. 2141–2148.

[31] M. Granados, K. I. Kim, J. Tompkin, J. Kautz, and C. Theobalt, "Background inpainting for videos with dynamic objects and a free-moving camera," in *Proc. ECCV*, 2012, pp. 682–695.

[32] A. Newson, A. Almansa, M. Fradet, Y. Gousseau, and P. Pérez, "Video inpainting of complex scenes," *SIAM J. Imag. Sci.*, vol. 7, no. 4, pp. 1993–2019, 2014.



Shuaicheng Liu (M'15) received the B.E. degree from Sichuan University, Chengdu, China, in 2008, and the M.S. and Ph.D. degrees from the National University of Singapore, Singapore, in 2010 and 2014, respectively.

He joined University of Electronic Science and Technology of China, Chengdu, in 2014, where he is currently an Associate Professor with the Institute of Image Processing, School of Electronic Engineering. His research interests include computer vision and computer graphics.

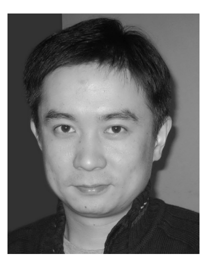
Binhan Xu received the B.Eng. degree in electronic and information engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2015. He is pursuing the master's degree in computer science with the University of California at Santa Barbara, Santa Barbara, CA, USA.

His research interests include computational photography, video processing, and augmented reality.

Chuang Deng (M'08) was born in Chengdu, China, in 1983. He received the B.E. degree in electronic and computer engineering from Tsinghua University, Beijing, China, in 2006, and the M.E. degree in electronic and computer engineering from Hong Kong University of Science and Technology, Hong Kong, in 2008. He is pursuing the Ph.D. degree with Sichuan University, Chengdu, China.

Since 2008, he has been with the Telecom and Automation Center, Sichuan Electric Power Corporation, Chengdu, where he has also been with the Emergency Management Center, focusing on information technology applications in power emergency management since 2012. He has authored 18 peer-reviewed research papers in national and international conferences and journals in power engineering. His research interests include design, construction, operation, and research on telecommunication infrastructures in power systems.

Mr. Deng is a member of the Chinese Society for Electric Engineering. He received the best paper award in the China's International Forum for Electric Power Safety Development and Emergency Management in 2009. He received the best paper award in the 2011 Annual Meeting on Information Technology in Power Industries.



Shuyuan Zhu (S'08–A'09–M'13) received the Ph.D. degree from Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2010.

From 2010 to 2012, he was with HKUST and Hong Kong Applied Science and Technology Research Institute Company Ltd., Hong Kong. In 2013, he joined the University of Electronic Science and Technology of China, Chengdu, China, where he is currently an Associate Professor with the School of Electronic Engineering. He has authored more than 40 research publications. His research interests include image/video compression, image processing, and compressive sensing.

Dr. Zhu is a member of the IEEE Circuits and Systems Society. He received the Top 10% Paper Award at the IEEE International Conference on Image Processing 2014. He was the Special Session Chair of Image Super-Resolution at the IEEE Digital Signal Processing 2015. He served as the Committee Member of the IEEE International Conference on Multimedia and Expo 2014, and serves as the Committee Member of the IEEE Visual Communications and Image Processing.

Bing Zeng (F'15) received the B.Eng. and M.Eng. degrees in electronics engineering from University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 1983 and 1986, respectively, and the Ph.D. degree in electrical engineering from Tampere University of Technology, Tampere, Finland, in 1991.

He was a Post-Doctoral Fellow with the University of Toronto, Toronto, ON, Canada, from 1991 to 1992, and a Researcher with Concordia University, Montréal, QC, Canada, from 1992 to 1993. He then joined Hong Kong University of Science and Technology (HKUST), Hong Kong. After 20 years at HKUST, he returned to UESTC in 2013 through China's 1000-Talent-Scheme, where, he leads the Institute of Image Processing to work on image and video processing, 3D and multiview video technology, and visual big data. During his tenure at HKUST and UESTC, he graduated more than 30 master's and Ph.D. students, filed eight international patents, and authored more than 200 papers.

Dr. Zeng received about 20 research grants. His three representing works are as follows: one paper on fast block motion estimation, published in the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 1994, has so far been SCI-cited more than 870 times (Google-cited more than 1960 times), currently stands at the seventh position among all papers published in the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY; one paper on smart padding for arbitrarily-shaped image blocks, published in the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2001, led to a patent that has been successfully licensed to companies; and one paper on directional discrete cosine transform, published in the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2008, received the 2011 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY Best Paper Award. He received the best paper award at China-Com three times (2009 Xi'an, 2010 Beijing, and 2012 Kunming). He received the Best Associate Editor Award in 2011. He also received a Second Class Natural Science Award (the first recipient) from the Chinese Ministry of Education in 2014. He served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY for eight years. He is on the Editorial Board of the *Journal of Visual Communication and Image Representation* and serves as the General Co-Chair of the 2016 IEEE Visual Communications and Image Processing to be held in Chengdu.

Moncef Gabbouj (F'11) received the B.S. degree in electrical engineering from Oklahoma State University–Stillwater, Stillwater, OK, USA, in 1985, and the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1986 and 1989, respectively.

He was an Academy of Finland Professor from 2011 to 2015. He held several visiting professorships at different universities. He is currently a Professor of Signal Processing with the Department of Signal Processing, Tampere University of Technology, Tampere, Finland. He is the TUT-Site Director of the National Science Foundation through its Industry/University Cooperative Research Center program for Visual and Decision Informatics. He has authored more than 650 publications and supervised 42 Ph.D. and 58 master's theses. His research interests include multimedia content-based analysis, indexing and retrieval, machine learning, nonlinear signal and image processing and analysis, voice conversion, and video processing and coding.

Dr. Gabbouj is a member of the Academia Europaea and the Finnish Academy of Science and Letters. He received the 2015 TUT Foundation Grand Award, the 2012 Nokia Foundation Visiting Professor Award, the 2005 Nokia Foundation Recognition Award, and several best paper awards. He was the Chairman of the IEEE Circuits and Systems (CAS) Technical Committee on Digital Signal Processing and a Committee Member of the IEEE Fourier Award for Signal Processing. He served as a Distinguished Lecturer of the IEEE CAS. He also served as an Associate Editor and a Guest Editor of many IEEE and international journals.