A Dataset and Evaluation Framework for Deep Learning Based Video Stabilization Systems

Maria Silvia Ito and Ebroul Izquierdo
Queen Mary University of London, United Kingdom
Email: {m.s.ito, ebroul.izquierdo}@qmul.ac.uk

***Abstract*— Although traditional methods for video stabilization are time consuming and prone to failure, more promising Deep Learning based solutions have not been thoroughly studied yet. This is mostly due to the lack of suitable training and testing datasets. To address this problem, this paper introduces a comprehensive dataset for training and assessing techniques for video stabilization. It consists of many shaky video sequences, their stable videos and the respective motion parameters that map each frame of the stable video into the corresponding frame in the unstable video. An important aspect of the dataset is the availability of motion parameters. This critical feature enables better assessment of any video stabilization technique, since it allows for additional comparison of the estimated motion parameters with the provided Ground Truth motion parameters. Using this dataset, an evaluation framework for video stabilization technology is also introduced. To demonstrate the practical use of the introduced dataset and the evaluation framework, we compare the performance of two state of the art techniques for video stabilization. The results of this extensive evaluation are presented and both the database and the evaluation framework are also provided in this paper.**

I. INTRODUCTION

Video stabilization plays an important role nowadays, by either improving the viewing experience of amateur videos or the performance of video processing applications. This is caused by a large number of videos recorded with mobile devices by amateur users, who tend to induce jittery motion due to body tremors. Also, some videos are recorded with mounted cameras (e.g. surveillance and military applications) and moving platforms (Unmanned Aerial Vehicles and robots): in these cases, the jerkiness is mostly introduced by atmospheric disturbances [1].

Video Stabilization techniques aim at removing undesired camera motion from a given video to improve its quality, by generating a compensated video which preserves intentional global motion [1][2]. Although there are a number of devices (such as lenses, tripods, steady cams, gyroscopes, among others) to prevent camera shake during the video capture [3], these solutions are unfeasible for amateur utilization, since they are expensive or demand large equipment [1][2].

In this scenario, the most adequate technique for casual amateur recording is Digital Video Stabilization (DVS), which is convenient and economical [3]. Traditional DVS methods match features between neighbouring frames or track these features for a certain amount of frames. However, they are time consuming [2], and features tend to be sensitive to certain characteristics of the video. For instance, quick camera motion and textureless regions tend to produce low quality features, leading to a low number of matched features and a short length of tracked motion [2].

(a) Simple (b) Blurry (c) High Motion
(d) Dark (e) Textureless (f) Parallax
(g) Depth (h) Crowd (i) Object

Fig. 1: Videos utilized in our performance evaluation

In the past years, the literature has reported that Deep Learning (DL) approaches have demonstrated the ability to handle and process complex, large-scale datasets for addressing various computer vision challenges, such as super resolution [4], image deblurring [5], style transfer [6], among others. In this scenario, DL-based approaches for DVS have recently been proposed in the literature [7][8]. However, large datasets are essential for learning based algorithms: for DVS, such datasets would require at least pairs of synchronized steady-unsteady frames. The work in [7] points out that the lack of appropriate datasets is one of the root causes for the few DL-based DVS systems available to date, and provides a dataset for DVS. However, it consists of a short amount of pairs and does not provide motion parameters that map steady and unsteady frames.

Having in mind such shortage and the fact that current DL-based DVS systems estimate transformation matrices in their algorithms, this paper provides a dataset that contains such information. We leverage the knowledge provided by the dataset in [7] to produce a large dataset of artificially produced steady-unsteady pairs. Unlike [7], we provide motion parameters, which can decrease the complexity in the training process of future algorithms, since the loss function can be based on the motion parameters instead of the output frames.

Also, performance evaluation in previous studies have mostly been non-reference (i.e., taking only the stabilized video into account). Since we provide a large amount of

Ground Truth (GT) and parameter (PR) data, we implement a full-reference evaluation framework, for performance evaluation based on GT, PR, and stabilized data. To demonstrate the practical use of the introduced dataset and the evaluation framework, we compare the performances of Estadeo [9], a 2D based DVS algorithm, and StabNet [7], a DL-based DVS algorithm, both described in Section IV. The contribution[1] of the paper is twofold: i) a dataset for training and testing DL-based DVS systems, which consists of a variety of video types, shown in Figure 1; ii) a full-reference evaluation framework, which considers GT frames and parameters.

The remainder of this paper is organized as follows. Section II presents the Related Work, Section III describes our dataset production, Section IV describes our evaluation framework, Section V discusses the experimental results, and Section VI presents our concluding remarks.

II. RELATED WORK

Deep Learning methods for DVS [7][8] leverage the capabilities of DL in computer vision tasks to estimate transformation parameters and create stable versions of unstable videos. However, large datasets for training purposes are required for developing these systems.

Most of the previous datasets [11][12] for DVS do not provide stable-unstable pairs of synchronised videos, which does not allow training DL systems. The work in [7] provides a dataset that contains such pairs. However, the synchronized video pairs were recorded with a handheld device, and the dataset provided 60 video pairs, which is not enough for training. In fact, for training StabNet (proposed in [7]), data augmentation techniques were utilized to the dataset. Also, the dataset lacks the corresponding motion parameters between the stable-unstable video pairs. This limits the development of DVS systems, since the only possibility during training is to compare GT and stabilized frames, not allowing a simpler comparison between motion parameters. An alternative to recording is to synthetically produce unsteady videos, which has been done in [13][14]. However, these works have not provided a dataset.

Another critical step for the development of DVS systems is the performance evaluation. In previous state of the art DVS systems proposals, [7][8][11], all the utilized evaluation methods were non-reference. Also, [10][15][16][17] propose evaluation frameworks for DVS systems. However, these frameworks consider non-reference evaluations, not taking advantage of the availability of GT videos. Finally, the work in [18] propose a full-reference evaluation framework, i.e., it considers GT videos in the process. However, such framework does not provide a video pair or motion parameter dataset, which makes such performance evaluation difficult.

To address the shortage of DL-suitable datasets, in this paper we provide a dataset of synchronised stable-unstable video pairs with the motion parameters that maps them. Since our video production is synthetic, our dataset can scale easily, and does not require time consuming recordings. With our dataset, we provide a full-reference performance evaluation framework. We believe these tools will contribute to the development of future DL-based DVS systems.

III. DATASET

This Section describes our dataset production. First, we collect the dataset provided by [7], then we compare each of the video pairs and, for each frame, we estimate a 2x3 transformation matrix. For some of the pairs, we are not capable of estimating such matrix, however the amount of transformation matrices obtained (approximately 23000 records) is considered a good sample for our purposes. We collect short videos from a free stock video repository[2], which contains several types of high quality and steady videos. We classify them into 9 categories: a)Simple, videos with the same depth and textured objects; b)Blurry; c)High motion; d)Dark; e)Textureless; f)Parallax; g)Discontinuous depth; h)Crowd, videos with large amounts of moving objects, with high motion and parallax; i)Close object, videos with at least one close object, leading to obstruction. The Simple category was selected because it contains features that have been previously addressed by DVS systems, and should not pose a problem to stabilization algorithms. The remaining video categories were selected because they contain one characteristic that poses a challenge to current DVS systems (as mentioned in Section I).

From this group of steady videos, we utilize the transformation matrices we obtained from [7] to produce our dataset of unsteady videos: for each frame in a given video, we randomly select one affine matrix and assign it to the given frame. To avoid a wobbly unnatural video, we utilize an Exponential Weighted Moving Average (EWMA) to smooth the transition between frames. We perform all these steps to make the unstable videos as close to real unstable videos as possible. We select one video from each category (shown in Figure 1) to evaluate the performance of state of the art DVS proposals.

IV. EVALUATION FRAMEWORK

This Section presents the metrics and formulae of our evaluation framework. Consider a stable-unstable video pair, which consist of n frames $S_{gt} = \{F_1, F_2, ..., F_n\}$ and $S_{un} = \{\hat{F}_1, \hat{F}_2, ...\hat{F}_n\}$, respectively. Also, consider a stabilized video, which is the output of a DVS system and consists of n frames $S_{st} = \{\bar{F}_1, \bar{F}_2, ..., \bar{F}_n\}$. The metrics we evaluate in our framework compare stabilized S_{st} and GT frames S_{gt}, and consist of:

i) Mean Square Error (MSE), Eq.1: the mean MSE between $\bar{F}_i$ and F_i frames of a video.

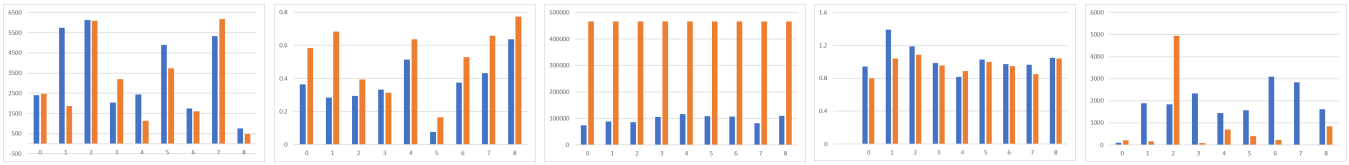
ii) Structural Similarity Index (SSIM), Eq.2: the mean SSIM between $\bar{F}_i$ and F_i frames of a video. Since these metrics compare $\bar{F}_i$ and F_i, they are computing all types of distortion (noise, blur, the distortion of straight lines, among others) in the resulting frame.

iii) Distance between features: shows how much a given feature has moved from GT to stabilized frame.

[1]Dataset and framework available at https://github.com/mariito/DVS_

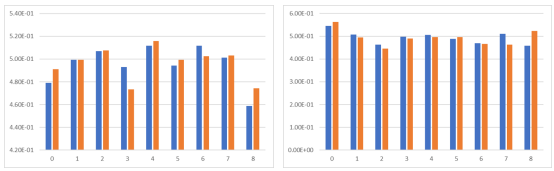
[2]https://www.pexels.com/

Fig. 2: Sample frames for video (c) (high motion)



(a) MSE (b) SSIM (c) Final Resolution (d) Resolution Loss (e) MSE-PR

Fig. 3: Experimental results. The blue bars represent StabNet, the orange bars represent Estadeo



(a) dx (b) dy

Fig. 4: Experimental results: 0-8 in the x-axis correspond to videos (a)-(i)

Consider a given set of features, which has coordinates $(x_{gt}, y_{gt})_1, (x_{gt}, y_{gt})_2, ..., (x_{gt}, y_{gt})_n$ in the GT frame and $(x_{st}, y_{st})_1, (x_{st}, y_{st})_2, ..., (x_{st}, y_{st})_n$ in the stabilized frame. The distance D_f between features can be computed as shown in Eq.3. The notion behind this is that, on average, the higher the mean distance between features in a frame pair, the more the stabilized frames are dislocated. The standard deviation would be another metric of interest, to understand how wobbly the stabilized frame is, and we will perform such study in future work.

$$MSE = \frac{1}{n} * \sum_{i=1}^{n} MSE(\bar{F}_i, F_i) \tag{1}$$

$$SSIM = \frac{1}{n} * \sum_{i=1}^{n} SSIM_i(\bar{F}_i, F_i) \tag{2}$$

$$D_f(x) = \frac{1}{n} * \sum_{i=1}^{n} |x_{st} - x_{gt}|_i, D_f(y) = \frac{1}{n} * \sum_{i=1}^{n} |y_{st} - y_{gt}|_i \tag{3}$$

iv) Resolution Loss, Eq.4: compares the average ratio of file size (F_{st} and F_{gt}) and number of pixels (P_{st} and P_{gt}) between S_{st} and S_{gt}, respectively. This metric aims at determining how much the video has been cropped, and if there has been any frame quality loss. We also compare the final resolution of the outputs to the DVS systems, given the same input size.

v) MSE_{PR}: aims at comparing estimated and GT motion parameters. Although the DVS systems we evaluate in this paper do not output motion parameters (only the stable frame), we leverage this metric and evaluate them with it. For that, we use the estimated motion parameters between frame pairs. Consider the transformation matrix M_{un}, utilized to produce the unsteady video and the one estimated between S_{gt} and S_{st}, M_{est}. Then, MSE_{PR} can be obtained according to Eq.5.

$$R_L = \frac{1}{n} * \sum_{i=1}^{n} \frac{(F_{st}/P_{st})_i}{(F_{gt}/P_{gt})_i} \tag{4}$$

$$MSE_{PR} = \frac{1}{n} * \sum_{i=1}^{n} MSE_i(M_{est}, M_{un}) \tag{5}$$

The previous state of the art papers propose slightly different metrics. The non-reference ones are not needed in our framework, since we can perform a full-reference evaluation. Some of the full-reference metrics will be added to our framework in future work. The next Section validates our framework by evaluating two state of the art studies: i) Estadeo [9] is a paper dedicated to implementing and exhaustively comparing classic DVS techniques and boundary conditions. In this paper, we will utilize the default options in the code provided by the author, which don't crop the border. ii) StabNet [7] is a low-latency, real-time, DL based method. It learns a set of transformations for each input frame, considering the previously stabilized frames in the video. StabNet showed that it can stabilize low quality videos, including: dark, blurry, watermarked, and noisy videos.

V. EXPERIMENTS

In this Section, we evaluate the selected state of the art proposals on DVS using the videos in Figure 1. Both proposals have been tested using the code provided by the authors. For [7], we utilized a pre-trained model provided. By following the authors' instructions, we tend to be fair to both systems, which is our main goal in our performance evaluation.

Figure 2 shows a randomly selected sample of (from left to right) the steady frame, unsteady frame, and the outputs to

StabNet and Estadeo, respectively. Figures 3 and 4 show the charts with the performance evaluation we executed with our framework. The blue bars represent the results we obtained for StabNet, whereas the orange bars represent the results for Estadeo. The x-axis represents the 9 different videos, shown in Figure 1: 0-8 correspond to videos (a) to (i), respectively. In our evaluation, we scaled the GT frames to have the same dimensions as the outputs to the systems, to be able to calculate metrics such as MSE.

By observing Figure 3a, we can note that both systems present high MSE, and it depends on the nature of the video. Most MSE values are similar, and when it differs significantly, it is mainly with StabNet presenting worse results. In Figure 3b, the output to Estadeo presents better results in most cases, and both systems have very low performance in video f (parallax). In Figure 3c, we can note that Estadeo's outputs present the same resolution regardless of the video characteristics, whereas StabNet outputs different resolutions, depending on the nature of the video. This could be an impairing factor, since all the resolutions of StabNet's outputs are significantly lower. One should note, however, that Estadeo, by default, does not crop the stabilized video and the resulting black borders could be annoying to the viewer. We can note, in Figure 3d, that the quality loss is similar for both systems in most cases, and when they differ significantly, Estadeo has the worst result. It means that, although Estadeo presents similar MSE and SSIM results, it presents worse performance when it comes to the video quality, which could be annoying to the viewer of the video, or could impair the performance of a given computer vision algorithm. This is an expected result, since it does not crop the black borders, presenting several black pixels.

Although the evaluated methods perform transformations differently, Figure 3e shows the comparison between ground truth motion parameters (M_{un}, from Eq 5) and the motion parameters obtained between ground truth and stabilized videos (M_{est}). Estadeo presents significantly better performance, apart from video c (high motion), in which case it has reasonably high MSE. This shows the influence of video type on the performance of a DVS algorithm.

Finally, Figures 4a and 4b, which show normalized distance between feature coordinates in the GT and stabilized frames, demonstrate that the features move at similar average distances at the x and y-axis for both Estadeo and StabNet. However, the movement in the x-axis is more sensitive to video content.

VI. CONCLUSION

This paper provided fundamental tools for the development of Deep Learning based Video Stabilization Systems. The first is a comprehensive synthetic dataset, which consists of Ground Truth and unsteady frames, and the motion parameters that map each frame of the stable video into the corresponding frame in the unstable video. We also presented a full-reference performance evaluation of DVS algorithms, which takes into account the provided features in the dataset. Then, we compared the performances of two state of the art DVS systems: a 2D based method, Estadeo, and a DL-based approach, StabNet. Both systems showed similar performance, although Estadeo performed better in some scenarios. This is proof that there are still opportunities for further enhancements of DL-based systems for DVS and showcases the importance of our work, which will assist the development of these systems. As future work, we intend to devise a statistical model based on [7] for the motion parameters, between steady and unsteady videos, in an attempt to simplify the learning process of DVS systems. We also intend to add metrics and existing DVS algorithms code to our framework, to facilitate the comparison of DVS systems.

ACKNOWLEDGEMENTS

The research activity leading to the publication has been partially funded by the European Union Horizon 2020 research and innovation program under grant agreement No. 787123 (PERSONA RIA project).

REFERENCES

[1] D Shukla et al., A new composite multiconstrained differential radon warping approach for digital video affine motion stabilization, Comput. Vis. Image Underst., Feb. 2017.
[2] S. Liu et al., Codingflow: Enable video coding for video stabilization, IEEE Trans. on Image Processing, July 2017.
[3] J. Dong and H. Liu, Video stabilization for strict real-time applications, IEEE Trans. on Circuits and Systems for Video Technology, April 2017.
[4] C. Dong et al., Image super-resolution using deep convolutional networks, IEEE Transactions on Pattern Analysis and Machine Intelligence, Feb 2016.
[5] S. Su et al., Deep video deblurring for hand-held cameras, IEEE Conference on Computer Vision and Pattern Recognition, Jul 2017
[6] H. Huang et al., Real-time neural style transfer for videos, IEEE Conference on Computer Vision and Pattern Recognition, July 2017
[7] M. Wang et al., "Deep Online Video Stabilization With Multi-Grid Warping Transformation Learning," in IEEE Transactions on Image Processing, May 2019.
[8] SenZhe Xu et al. "Deep Video Stabilization Using Adversarial Networks", in Computer Graphics Forum, 2018.
[9] J Snchez, Comparison of Motion Smoothing Strategies for Video Stabilization using Parametric Models, Image Processing On Line, 7 (2017)
[10] W. Guilluy et al, "A performance evaluation framework for video stabilization methods," 2018 7th European Workshop on Visual Information Processing, 2018
[11] S Liu et al. 2013. Bundled camera paths for video stabilization. ACM Trans. Graph., July 2013)
[12] Y. J. Koh et al., "Video Stabilization Based on Feature Trajectory Augmentation and Selection and Robust Mesh Grid Warping," in IEEE Transactions on Image Processing, Dec. 2015.
[13] H. Qu et al., "Shaking video synthesis for video stabilization performance assessment", IEEE International Conference on Visual Communications and Image Processing, 2013.
[14] S. Lu et al., "Synthesis of Shaking Video Using Motion Capture Data and Dynamic 3D Scene Modeling," IEEE International Conference on Image Processing, 2018.
[15] L. Zhang et al., "Intrinsic Motion Stability Assessment for Video Stabilization," in IEEE Transactions on Visualization and Computer Graphics, 2019.
[16] C. Zhang et al., "Qualitative assessment of video stabilization and mosaicking systems" IEEE Workshop on Applications of Computer Vision, 2008.
[17] B. Zhai et al., "A Multi-scale Evaluation Method for Motion Filtering in Digital Image Stabilization," IEEE International Conference on Tools with Artificial Intelligence, 2015.
[18] M. Niskanen et al., "Video Stabilization Performance Assessment," IEEE International Conference on Multimedia and Expo, 2006.