Semantic Filtering for Video Stabilization

K. Karageorgos, A. Dimou, A. Axenopoulos, P. Daras F. Alvarez

Abstract

Moving objects pose a challenge to every video stabilization algorithm. We present a novel, efficient filtering technique that manages to remove outlier motion vectors caused from moving objects in a per-pixel smoothing setting. We leverage semantic information to change the calculation of optical flow, forcing the outliers to reside in the edges of our semantic mask. After a 'content-preserving warping' and a smoothing step we manage to produce stable and artifact-free videos.

1. Introduction

In recent years, video surveillance technology goes increasingly mobile following a wider trend. Body-worn cameras, in-car video systems and cameras installed on public transportation vehicles are only a few cases of mobile surveillance infrastructure. Moreover, Law Enforcement Agencies are increasingly including videos recorded by mobile devices in their investigations. While, this new source of videos opens up new opportunities for the authorities, it also introduces new challenges in terms of processing, manual or automatic. Besides the huge amount of recorded footage, the produced content is usually unstable and shaken, making their manual inspection an (even more) cumbersome procedure and its automated analysis problematic due to spatial inconsistency between frames.

Video stabilization is the process of generating a new compensated video sequence, where undesirable image motion is removed and has been steadily gaining in importance with the increasing use of mobile camera footage. Often, videos captured with a mobile device suffer from a significant amount of unexpected image motion caused by unintentional shake of their mounting, whether this is a hand, body or vehicle. Given an unstable video, the goal of video stabilization is to synthesize a new image sequence as seen from a new stabilized camera trajectory. A stabilized video is sometimes defined as a motionless video where the camera motion is completely removed. In this paper, we refer to stabilized video as a motion compensated video where only undesirable camera motion is removed. This distinction is critical since camera motion can contribute towards an aesthetically pleasing result and be instrumental for capturing the details of a scene [6].

The first step towards video stabilization involves the choice of a suitable model that will adequately represent camera motion. Optical flow is the most generic motion model and recent work has shown great potential in its use for video stabilization. However, the optical flow of a general video can be rather irregular, especially on moving objects at different depths of the scene, therefore, a motion model with strong spatio-temporal consistency and smoothness is required to stabilize the video. The approach of identifying discontinuous flows by spatio-temporal analysis and enforcing strong spatial smoothness to the optical flow neglects the semantic information of the scene contents, leading to severe artifacts when moving objects are very close to the camera or cover a large part of it. This is due to the fact that the distinction between background and foreground objects is obscured by their comparable size [16].

In this paper, we are proposing the use of semantic information extracted from the examined scene together with a dense 2D motion field to produce a model representing the camera motion. The derived model allows us to generate stabilized videos with good visual quality even in challenging cases such as scenes with large foreground objects which are common in footage from mobile cameras.

1.1. Related work

Video stabilization techniques can be roughly categorized regarding their underlying motion model as 2D and 3D methods. 2D stabilization methods use a cascade of geometric transformations (such as homography or affine models) to represent the camera motion, and smooth these

transformations to stabilize the video. The type of smoothing can have a dramatic effect on the qualitative evaluation of the result. One early method [17] used simple low-pass filtering, which requires very big temporal support to eliminate unwanted low frequency shaking (*e.g.* walking). Dealing with that, Chen *et al.* [4] applied polynomial curve fitting on top of Kalman-based filtering. Gleicher and F. Liu [6] broke camera trajectories into segments for individual smoothing, following principles of cinematography. Grundmann *et al.* [8] encapsulated this idea into an elegant L1-norm optimization, while S. Liu *et al.* [15] split the frame into multiple segments, each with its own path, and applied a joint stabilization method.

3D methods use the estimated camera position in space for stabilization and are, thus, heavily reliant on the effectiveness of structure from motion algorithms. Although they give superior results on complex scenes with parallax and depth changes, they are computationally heavier and less robust. An example of early work is from Beuhler *et al.* [3], who used a projective 3D reconstruction with an uncalibrated camera for video stabilization. F. Liu *et al.* [10] used 3D point correspondences to guide a novel and influential 'content-preserving' warping method, whose efficiency was later improved on planar regions by Zhou *et al.* [25]. S. Liu *et al.* [14] used a depth camera for robust stabilization.

In the middle ground between the two, 2.5D methods compensate for the lack of 3D information imposing additional constraints. F. Liu *et al.* [11] built on the observation that feature trajectories from a projective camera lie on a subspace and smoothed its basis eigenvectors. There is an extension of this method for stereoscopic videos as well [12]. Goldstein and Fattal [7] leverage the epipolar relations that exist among features of neighboring frames. Wang *et al.* [21] represented each trajectory as a Bezier curve and smoothed them with a spatio-temporal optimization. Though more robust than 3D methods, 2D ones demand reliable tracking to construct feature trajectories. We build on the work of S. Liu *et al.* [13, 16] which tries to alleviate the problem of acquiring long trajectories by smoothing the pixel profiles instead.

2. Methodology

In this paper, the assumption made in [16] that the motion vector of each pixel should approximate the trajectory of the corresponding point in the scene is adopted. Given this assumption, instead of smoothing feature trajectories, we can smooth the pixel profiles, where a pixel profile is defined as the accumulated optical flow vector at each pixel location. Thus, video stabilization can be achieved in a pixel-wise manner by using a pixel profile stabilization model. This assumption does not hold well, though, for scenes containing sharp depth changes and moving objects, as they can cause the optical flow field to be spatially uneven. In such cases, as it can be seen in figure 2, smoothing the pixel profiles leads to artifacts. Therefore, we must modify the initial optical flow and discard the motion vectors that cause these distortions. In [16] this is performed in two iterative filtering steps, a spatial and a temporal one, trying to enforce spatio-temporal consistency.

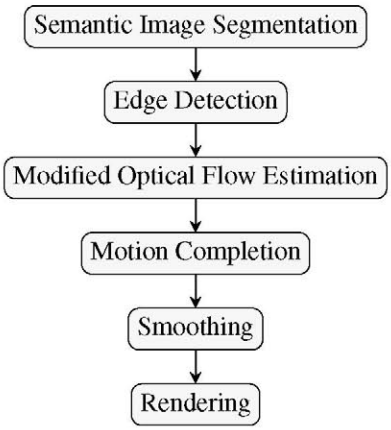

Figure 1. Methodology Outline

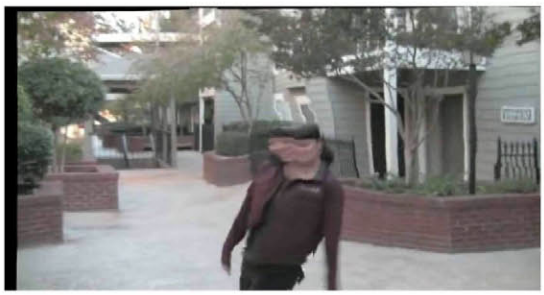

Figure 2. Unfiltered smoothing failure

Instead, we propose a novel method aiming to perform motion outliers rejection on the optical flow field exploiting semantic information in the context of video stabilization. For this purpose we leverage state of the art semantic segmentation [18, 24] of the scene examined to detect moving objects of interest in a surveillance scene, such as people or vehicles. Semantic segmentation masks provide the information necessary to reject irregular motion vectors, regardless of objects' size, in a single step, eliminating the need for an iterative approach and leading to a visually pleasing result.

2.1. Semantic optical flow refinement

Classical optical flow algorithms impose smoothness on the resulting flow field in order to solve the brightness constancy constraint equation [9]. This results in flow fields with smooth transitions between areas with different motions, producing motion irregularities within a single object.

(a) Semantic mask

(b) Optical flow calculated without our constraint and overlaid semantic boundary

(c) Completed flow field after naive filtering

Figure 3. Outlier filtering without optical flow refinement. Notice the "shade" under the right arm and the blue colored artifact that fall out of the semantic mask, resulting in insufficiently filtered flow.

(a) The frame from which we compute the semantic mask

(b) New flow that respects semantic segmentation boundaries

(c) Flow free from motion vectors that cause distortions

Figure 4. Outlier filtering with optical flow refinement. Notice the alignment between the motion vectors and the boundary in (b).

It is worth noting that this transition causes inaccurate motion vector estimation at both sides of the motion boundary, since the two motion fields influence each other. However, recent work on optical flow estimation has leveraged the use of additional information to improve flow precision, particularly at object boundaries [19, 20].

In [19] a variational energy minimization approach is employed on a grid of dense correspondences. This grid is a product of interpolation with respect to a geodesic distance, whose cost function penalizes boundary crossing. Normally, one would use an edge detection algorithm on the video frame to define these boundaries. Edge detectors that work on natural images, though, produce edges of varying strength, which do not adequately restrict the interpolation and result in flows that do not respect the boundaries of our semantic segmentation, as seen in Figure 3(b).

In this direction, we acquire a semantic segmentation mask for each frame in the examined video using [24] trained with the PASCAL VOC dataset that contains 21 labels including background. Given our application we are only interested in moving objects (*e.g.* persons, cars, motorbikes) and, thus, we discard all labels related to static objects or background (*e.g.* potted plant, sofa). A naive approach would be to discard every motion vector under the semantic mask as outlier. Not surprisingly, such a method fails because of the discrepancy between the object boundaries that are delineated from the motion vectors and the corresponding ones from the semantic masks (figure 3(c)).

Instead, we employ standard edge detection on the semantic masks, producing a set of crisp boundaries surrounding the, potentially moving, area of our frame. Leveraging the notion of geodesic distance that preserves object boundaries, we use these edges as input to the estimation of motion flow field to force the outlier vectors to reside within the boundaries of the moving object (figure 4(b)). Thus, the optical flow becomes consistent with our semantic segmentation simplifying the stabilization pipeline.

2.2. Motion completion

The next step is to complete the missing values of the optical flow field. We interpolate the outlier motion vectors from a grid formed in a content preserving way [10]. We use the motion vectors at the boundary of the semantic mask to form control points for the energy minimization problem:

$$E = E_d + \alpha E_s, \tag{1}$$

where E_d and E_s are the data and similarity terms, weighted by α. The data term is defined as a sum over all inlier points p:

$$E_d(V) = \sum_p \|V\pi_p - (p + u_p)\|, \tag{2}$$

with u_p being the initial optical flow at pixel p and V indicating the unknown vertices of the new grid that enclose

p. π_p is the vector of bilinear coordinates of point p at the initial grid. Thus, E_d weighs toward accurate reconstruction at each data point. However, this could force the rest of the pixels to be extremely warped or distorted which is counter-weighted by the similarity term:

$$E_s(V) = \sum_u \| u - u_1 - sR_{90}(u_0 - u_1)\|^2, \qquad (3)$$

$$R_{90} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

This term requires that each triangle, formed by u and two of its neighboring vertices u_0, u_1, follows a similarity transform. $s = \|u-u_1\|/\|u_0-u_1\|$ is a term computed from the original mesh. The new vertices are calculated minimizing a standard sparse linear system. The new motion values are then bilinearly interpolated using the resulting grid.

2.3. Stabilization

The stable video is produced by smoothing each pixel profile independently. We do not employ an adaptation scheme for the temporal window, since all these approaches require arbitrary thresholding and are heavily influenced from the frame rate. The smoothing is achieved minimizing the following objective function:

$$O(P_t) = \sum_t \Big(\|P_t - C_t\|^2 + \lambda \sum_{r \in \Omega_t} w_{t,r} \|P_t - P_r\|^2 \Big), \quad (4)$$

where C is the cumulative motion vector field of the input video at frame t and P the corresponding one of the output video. $w_{t,r}$ is the weight of past and future frames r in the temporal window Ω_t and is calculated by $w_{t,r} = exp(-\|r-t\|^2/(\Omega_t/3)^2)$. The first term of this sum is the similarity between the stabilized and the initial frames, a factor that minimizes cropping, while the second term expresses the similarity of the new frame to its neighboring ones, which maximizes stability. Finally, λ acts as a balancing term that allows us to favor the one over the other.

The optimization is solved by a Jacobi-based iteration [2] for each pixel by:

$$P_t^{(\xi+1)} = \frac{1}{\gamma}\Big(C_t + \lambda \sum_{r \in \Omega_t, r \neq t} w_{t,r} P_r^{(\xi)} \Big), \qquad (5)$$

with the scalar $\gamma = 1 + \lambda \sum_r w_{t,r}$ and ξ being the iteration index (by default, $\xi = 10$). Note that unlike Liu *et al*. our algorithm runs only once. We render the final result by warping each frame with a dense displacement field $B_t = P_t - C_t$.

(a) Scene with many faces (b) Inaccurate semantic mask

Figure 7. Semantic segmentation failure

3. Experimental results

We conducted a wide range of experiments on publicly available baseline videos with moving objects, occlusions and parallax. Additionally, we experimented on videos from the surveillance domain, especially police body-cam videos, which contain highly irregular motion (*e.g*. walking, running) and occlusions, especially from persons, bystanders etc.

Our method manages to successfully filter out moving objects in the majority of cases. Figure 5 shows a typical failure case for most trajectory based methods, where an object covering a significant portion of the screen crosses the field of view. Naturally, such an object has a big effect on the flow field and if we stabilize the video without some way of filtering we see visible artifacts (*e.g*. the elongated head of the lady in the foreground, together with the warped body of the lady in the background in row 2). Our output is stable and without artifacts. Similarly, in the surveillance domain video of figure 6, which again contains a significantly big moving object and heavy shake, one can clearly see the distortion on the face of the officer, especially on the last frame of row 2, which does not exist in our output. The presented results are qualitative, since result quantification is not a trivial matter in video stabilization, due to the fact that there are no benchmarks or widely accepted metrics available.

3.1. Implementation details

We implemented our method in Python and run it on commodity pc hardware consisting of an i7-6700K CPU, GTX 1070 GPU with 32 GBs of RAM on Ubuntu Linux 14.04. For the initial semantic segmentation masks and optical flow we used the, publicly available, CRF-RNN [1, 24] as well as the GPU implementation of DeepMatching [22] in conjunction with EpicFlow [19]. For the videos in our domain we empirically choose $\alpha = 1$, $\lambda = 1$ as they give the most pleasing results.

4. Conclusions

We presented a novel video stabilization pipeline that leverages the latest advances in semantic image segmen-

Figure 5. Typical failure case for trajectory based methods. Our system manages to stabilize this heavily occluded scene. The rows from top to bottom correspond to the original, stabilized without filtering and successfully stabilized cases. Notice the heavy distortions in the second row.

Figure 6. Four frames of a video in the surveillance domain. Again, the first row depicts the original, unstable, video, the second one is a stabilized without semantic filtering and the third a stabilized version with our method. Notice the distortions around the officer's head at the last row, while our results remain crisp.

tation and fuses this information to refine the calculation of optical flow. This way we manage to produce stable, artifact-free videos in scenes with moving objects, occlusions and parallax.

4.1. Limitations and future work

Our method does not fall in the realm of 3D methods and, as a result, cannot provide 3D camera motion planning. The degree of stabilization, though, can be controlled by selecting the appropriate temporal support. Our method relies on the quality of optical flow calculation and image segmentation, which, as seen in figure 7, can identify persons unexpectedly (*e.g.* toys, posters). Temporally consistent semantic segmentation is a possible solution for the removal of such artifacts, something that we are keen to explore.

Since we have shown that it is possible to integrate deep learning methods in the filtering stage of a stabilization pipeline, we would be interested in examining the smoothing and result synthesis steps also. There are promising results in the field of novel view synthesis [5] and image inpainting [23] which we are keen to explore and could lead to a fully neural, full frame architecture.

References

[1] A. Arnab, S. Jayasumana, S. Zheng, and P. H. S. Torr. Higher order conditional random fields in deep neural networks. In *European Conference on Computer Vision (ECCV)*, 2016.

[2] I. N. Bronshtein and K. A. Semendyayev. *Handbook of mathematics*. Springer Science & Business Media, 2013.

[3] C. Buehler, M. Bosse, and L. McMillan. Non-metric image-based rendering for video stabilization. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–II. IEEE, 2001.

[4] B.-Y. Chen, K.-Y. Lee, W.-T. Huang, and J.-S. Lin. Capturing intention-based full-frame video stabilization. In *Computer Graphics Forum*, volume 27, pages 1805–1814. Wiley Online Library, 2008.

[5] J. Flynn, I. Neulander, J. Philbin, and N. Snavely. Deepstereo: Learning to predict new views from the world's imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5515–5524, 2016.

[6] M. L. Gleicher and F. Liu. Re-cinematography: improving the camera dynamics of casual video. In *Proceedings of the 15th ACM international conference on Multimedia*, pages 27–36. ACM, 2007.

[7] A. Goldstein and R. Fattal. Video stabilization using epipolar geometry. *ACM Transactions on Graphics (TOG)*, 31(5):126, 2012.

[8] M. Grundmann, V. Kwatra, and I. Essa. Auto-directed video stabilization with robust l1 optimal camera paths. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 225–232. IEEE, 2011.

[9] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[10] F. Liu, M. Gleicher, H. Jin, and A. Agarwala. Content-preserving warps for 3d video stabilization. *ACM Transactions on Graphics (TOG)*, 28(3):44, 2009.

[11] F. Liu, M. Gleicher, J. Wang, H. Jin, and A. Agarwala. Subspace video stabilization. *ACM Transactions on Graphics (TOG)*, 30(1):4, 2011.

[12] F. Liu, Y. Niu, and H. Jin. Joint subspace stabilization for stereoscopic video. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 73–80, 2013.

[13] S. Liu, P. Tan, L. Yuan, J. Sun, and B. Zeng. Meshflow: Minimum latency online video stabilization. In *European Conference on Computer Vision*, pages 800–815. Springer, 2016.

[14] S. Liu, Y. Wang, L. Yuan, J. Bu, P. Tan, and J. Sun. Video stabilization with a depth camera. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 89–95. IEEE, 2012.

[15] S. Liu, L. Yuan, P. Tan, and J. Sun. Bundled camera paths for video stabilization. *ACM Transactions on Graphics (TOG)*, 32(4):78, 2013.

[16] S. Liu, L. Yuan, P. Tan, and J. Sun. Steadyflow: Spatially smooth optical flow for video stabilization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4209–4216, 2014.

[17] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum. Full-frame video stabilization with motion inpainting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1150–1163, 2006.

[18] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016.

[19] Revaud, Jerome, Weinzaepfel, Philippe, Harchaoui, Zaid, and Schmid, Cordelia. EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow. In *Computer Vision and Pattern Recognition*, 2015.

[20] L. Sevilla-Lara, D. Sun, V. Jampani, and M. J. Black. Optical flow with semantic segmentation and localized layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3889–3898, 2016.

[21] Y.-S. Wang, F. Liu, P.-S. Hsu, and T.-Y. Lee. Spatially and temporally optimized video stabilization. *IEEE transactions on visualization and computer graphics*, 19(8):1354–1361, 2013.

[22] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *IEEE Intenational Conference on Computer Vision (ICCV)*, Sydney, Australia, Dec. 2013.

[23] R. Yeh, C. Chen, T. Y. Lim, M. Hasegawa-Johnson, and M. N. Do. Semantic image inpainting with perceptual and contextual losses. *arXiv preprint arXiv:1607.07539*, 2016.

[24] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. Conditional random fields as recurrent neural networks. In *International Conference on Computer Vision (ICCV)*, 2015.

[25] Z. Zhou, H. Jin, and Y. Ma. Plane-based content preserving warps for video stabilization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2299–2306, 2013.