

Visual Fixation for 3D Video Stabilization

Christian Kurz, Thorsten Thormählen, and Hans-Peter Seidel

Max Planck Institute for Computer Science (MPII)

Campus E1_4

66123 Saarbrücken, Germany

phone: +49 681 9325 422

email: {ckurz, thormae}@mpi-inf.mpg.de

www: www.mpi-inf.mpg.de

February 24, 2011

Abstract

Visual fixation is employed by humans and some animals to keep a specific 3D location at the center of the visual gaze. Inspired by this phenomenon in nature, this paper explores the idea to transfer this mechanism to the context of video stabilization for a hand-held video camera. A novel approach is presented that stabilizes a video by fixating on automatically extracted 3D target points. This approach is different from existing automatic solutions that stabilize the video by smoothing. To determine the 3D target points, the recorded scene is analyzed with a state-of-the-art structure-from-motion algorithm, which estimates camera motion and reconstructs a 3D point cloud of the static scene objects. Special algorithms are presented that search either virtual or real 3D target points, which back-project close to the center of the image for as long a period of time as possible. The sta-

bilization algorithm then transforms the original images of the sequence so that these 3D target points are kept exactly in the center of the image, which, in case of real 3D target points, produces a perfectly stable result at the image center. Furthermore, different methods of additional user interaction are investigated. It is shown that the stabilization process can easily be controlled and that it can be combined with state-of-the-art tracking techniques in order to obtain a powerful image stabilization tool. The approach is evaluated on a variety of videos taken with a hand-held camera in natural scenes.

keywords : video stabilization, visual fixation, camera shake, camera motion estimation, structure-from-motion.

1 Introduction

When moving in an environment, the vision system of humans and several animals uses the process of ocular fixation that stabilizes the center of the visual gaze on a particular position in 3D space. Thereby, the movement of the eyes compensates the possible jitter introduced by the motion of the body [Car88]. Inspired by ocular fixation, in this paper we investigate, how the process of fixation can be used to stabilize the images of a video recorded with a hand-held video camera.

Current consumer cameras are usually equipped with video stabilization hardware to reduce camera shake; e.g., special lens systems or moveable

Digital Peer Publishing Licence

Any party may pass on this Work by electronic means and make it available for download under the terms and conditions of the current version of the Digital Peer Publishing Licence (DPPL). The text of the licence may be accessed and retrieved via Internet at <http://www.dipp.nrw.de/>.

First presented at the 6th European Conference on Visual Media Production (CVMP 2009) under the title: "Scene-aware Video Stabilization by Visual Fixation", extended and revised for JVRB

image sensors in combination with gyroscopic sensors [Fou91, KKM04]. However, these systems can usually compensate only small vibrations.

Software solutions offer greater flexibility and are able to remove undesired camera shakes of large amplitude. Most methods use block matching [LPLP09], track image features [CFR99], or estimate the optical flow [CLL06, CW08] between successive images. This information is then used to obtain the parameters of a 2D transformation between the images. The transformation parameters are then smoothed and the difference between the original and the smooth transformation is applied to compensate the undesired camera shake.

Different 2D transformations were explored, starting from a simple two-dimensional shift of the image [Ert02, LHY09] to affine transformations [CLL06]. Instead of using 2D transformations there are also approaches that employ 2.5D [JZX01] or 3D camera models [MC97, Krü99, BBM01, LGJA09], sometimes with substantial simplifications [WCCF09] to ease computation. Various smoothing approaches exist, e.g., Kalman filters [Ert02], particle filters [dBJSG08], the Viterbi method [Pil04], or other digital filters [LHY09].

Recently, an approach based on feature trajectory smoothing [LCCO09] has been described.

Fixation models loosely based on the human visual system have been used for improving optical flow by locally stabilizing very short images sequences in a sliding window [PLH07], and in determining and stabilizing the camera-object-distance [LWV98].

This paper has the following contributions:

- An image stabilization approach, which simulates ocular fixation used in human and animal vision by fixating the camera orientation to a specific 3D target point in the scene.
- A fully-automatic scene analyzation technique for the extraction of 3D target points to fixate on, either virtual or real.
- An extension of the automatic approach to incorporate different forms of user interaction to control the stabilization procedure.

The advantage of the described image stabilization technique, in contrast to smoothing, is that after stabilization the target point is kept perfectly stable in the image center. To extract the 3D target points from

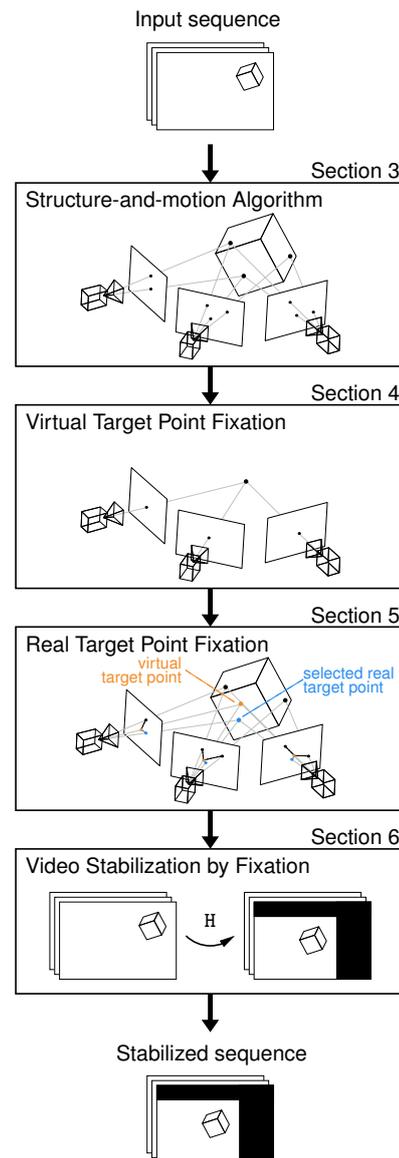


Figure 1: The processing pipeline of the stabilization by visual fixation algorithm.

the recorded scene, we first employ an off-the-shelf structure-from-motion tool. The 3D reconstruction of the scene is then analyzed, yielding the desired 3D target points. Thereby, the algorithm can either generate a virtual target point or a target point that is located on a real surface in the 3D scene. These target point extraction algorithms are simple to implement and require only a single user parameter, which controls directly how strongly the original image sequence is altered due to fixation.

As a novel contribution in this journal publication, we furthermore investigate different forms of user interaction to control the stabilization procedure, i.e., the manual selection of target points and the use of state-

of-the-art tracking techniques to generate a series of target points that allow the algorithm to fixate onto arbitrary static or moving objects. This enables the user to exert a great amount of control over the stabilization process, allowing certain artistic needs to be reached.

The paper is organized as follows. In the next section, we briefly summarize the notation used in this paper. In Section 3 we describe the information that is available after employing a state-of-the-art structure-from-motion approach. Sections 4 and 5 introduce the algorithms to extract the virtual and real target points from the recorded image sequence. Section 6 explains how the target points can be used for video stabilization. These sections correspond to individual steps of the algorithm, which is illustrated in Fig. 1. Additional user-control over the stabilization process is covered in Section 7. In Section 8, we report results of our experiments that show the performance of the suggested algorithms. The paper ends with concluding remarks in Section 9.

2 Notation

Throughout this paper, 2D points will be denoted by lower-case letters written in boldface (e.g., vector \mathbf{a}). In a similar manner, an upper-case boldface letter (\mathbf{A}) denotes a 3D point or 3-vector. Matrices are indicated by upper-case letters in typewriter font style (A). Scalar values are given by upper- and lower-case italic letters (A, a), unless specified otherwise.

3 Structure-from-Motion Algorithm

Reliable algorithms for camera motion estimation and 3D reconstruction of rigid objects from video have been developed over the last decades [GCH⁺02, PGV⁺04, THWS08]. Employing such a state-of-the-art structure-from-motion algorithm is the first step in our processing pipeline.

Consider an image sequence consisting of K images I_k , with $k = 1, \dots, K$. Let A_k be the 3×4 camera matrix corresponding to image I_k . First, corresponding 2D feature points $\mathbf{p}_{j,k}$ are determined in consecutive frames with the KLT-Tracker [ST94]. Using the corresponding feature points, the parameters of a camera model A_k are estimated for each frame. As shown in Fig. 2, for each feature track a corresponding 3D object point position is determined, resulting in set

of J 3D object points \mathbf{P}_j , with $j = 1, \dots, J$, where

$$\mathbf{p}_{j,k} \simeq A_k \mathbf{P}_j \quad . \quad (1)$$

Thereby, the 2D feature points $\mathbf{p}_{j,k} = (p_x, p_y, 1)^\top$ and 3D object points $\mathbf{P}_j = (P_x, P_y, P_z, 1)^\top$ are given in homogeneous coordinates.

The camera matrix A can be factorized into

$$A = KR[I | -C] \quad , \quad (2)$$

where the 3×3 calibration matrix K contains the intrinsic camera parameters (e.g., focal length or principal point offset), R is the 3×3 rotation matrix representing the camera orientation in the scene, and the camera center C describes the position of the camera in the scene.

4 Virtual Target Point Fixation

Once the camera motion parameters and 3D object points have been obtained, virtual 3D target points \mathbf{T}_i for fixation are estimated (due to the nature of the reconstruction process, these virtual 3D target points do not necessarily coincide with reconstructed 3D points \mathbf{P}_j). It is assumed that the camera operator tries to keep the respective object of interest centered in the image but introduces large jitter because of the hand-held camera. Given the principal point c_k of the camera view k , which is the intersection of the optical axis with the image plane, an estimate for the 3D target point \mathbf{T}_i can be found by a triangulation algorithm minimizing

$$\arg \min_{\mathbf{T}_i} \sum_{n \in \mathcal{N}_i} d(c_n, A_n \mathbf{T}_i) \quad , \quad (3)$$

where \mathcal{N}_i is a subset of the whole set of images $[1 \dots K]$ consisting of strictly consecutive images, and $d(\dots)$ denotes the Euclidean distance.

To determine a suitable subset of images \mathcal{N}_i for a target point, a multi-scale approach is employed, which evaluates the sequence at multiple time-scales.

The coarsest scale is assigned to scale index $S = 0$, while the index is incremented for the subsequent, refined scales. Given a specific scale with the corresponding scale index S , the total number N_S of consecutive images for all individual subsets \mathcal{N}_i for this scale is

$$N_S = K - S \quad . \quad (4)$$

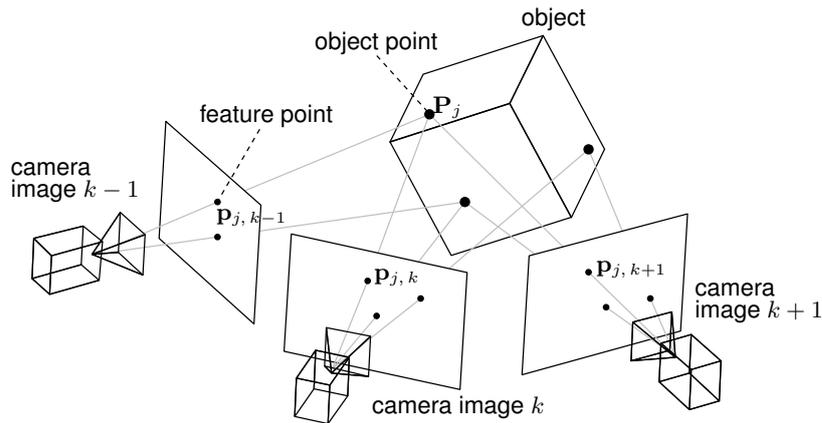


Figure 2: Result after structure-from-motion estimation. The projection of a 3D object point \mathbf{P}_j in the camera image at time k gives the tracked 2D feature point $\mathbf{p}_{j,k}$.

For any given scale with scale index S the maximum number M_S of possible subsets \mathcal{N}_i evaluates to

$$M_S = K - N_S + 1 = S + 1 \quad . \quad (5)$$

This is due to the fact that the subsets \mathcal{N}_i are required to consist only of strictly consecutive frames.

As an example, consider a sequence containing a total of $K = 90$ images for a scale with scale index $S = 30$. There are at most $M_{30} = 31$ different subsets \mathcal{N}_i with a length of $N_{30} = 60$ images each.

Starting at the coarsest scale, the algorithm evaluates all possible subsets of consecutive images, by checking if the residual error of Eq. (3) is below a certain user defined threshold τ . If this condition is satisfied, a target point candidate is created and stored in a candidate list, which is sorted ascendingly according to the residual error.

After processing all subsets, the target point candidate with the lowest residual error is selected and moved to the list of accepted target points. The corresponding image set is assigned to the accepted target point and excluded from further processing. All target point candidates, which share images with the accepted target point are removed from the candidate list. The process is repeated for the next target point candidate in the candidate list until the list is empty.

At the next finer time-scale all remaining possible subsets \mathcal{N}_i containing N_S consecutive images are considered. Once all subsets of a given scale have been processed, the scale index S is increased and the corresponding subsets of the next finer time-scale are considered, where it is made sure that only subsets not containing images assigned to subsets on coarser time-scales are selected. This reduces the number of possi-

ble subsets for all finer scales.

The algorithm terminates after all images have been assigned to an accepted target point or further refinement is no longer possible.

5 Real Target Point Fixation

In contrast to the virtual 3D target points obtained in the previous section, only a real 3D target point present on a surface in the scene permits a perfectly stable projection of the respective surface at the image center. Therefore, it is often desirable that the selected target point corresponds to a real 3D object point of the scene. When the user activates this real target point fixation, a suitable 3D object point is selected from the set of all J 3D object points \mathbf{P}_j for each virtual target point. Thereby, it is evaluated whether the back-projection of the 3D object points in the subset of images, which is assigned to the current virtual target point, is close to the principal point \mathbf{c}_n :

$$\epsilon_j = \sum_{n \in \mathcal{N}_i} d(\mathbf{c}_n, \mathbf{A}_n \mathbf{P}_j) \quad . \quad (6)$$

The 3D object point \mathbf{P}_j with the smallest error ϵ_j is selected.

Undesired results might be obtained for image sequences where 3D object points in the vicinity of the virtual target points were not generated during the structure-from-motion step due to a lack of interest points in the respective image regions. This problem can be solved by enforcing an additional threshold on the residual error ϵ_j and by reverting to the virtual target point if necessary.

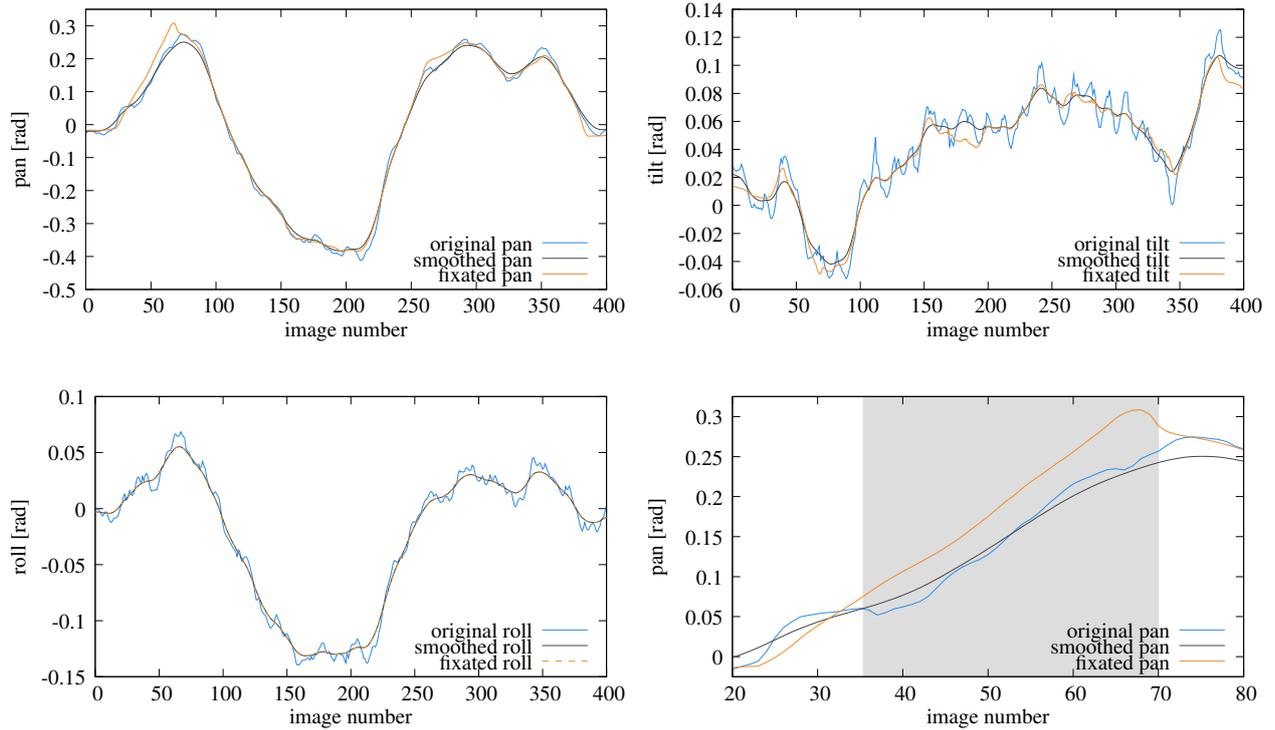


Figure 3: Example 1 - Comparison between the camera parameters estimated from the *original* image sequence, the *smoothed* parameters [CLL06], and the *fixated* parameters. Results for camera parameters pan, tilt, and roll are shown. The diagram in the lower right corner shows a detail magnification for the pan parameter. The gray region indicates the fixation to a target point.

6 Video Stabilization by Fixation

To stabilize the image sequence, a 2D transformation, given by the 3×3 matrix H_k , is applied to all images I_k of the sequence. If $(x', y')^\top$ and $(x, y)^\top$ are the pixel positions in the stabilized and unstabilized images, respectively, this operation can be written as

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \simeq H_k \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad (7)$$

with

$$H_k = (K_k R_k^{(s)})(K_k R_k)^{-1}. \quad (8)$$

The calibration matrices K_k and the rotation matrices R_k are known from the structure-from-motion algorithm. The rotation matrices $R_k^{(s)}$ are the smoothed versions.

A camera rotation matrix can be represented by three Euler angles, pan φ , tilt ϑ , and roll ρ with

$$R = R_y(\varphi) R_x(\vartheta) R_z(\rho), \quad (9)$$

where R_y , R_x , and R_z are rotations around the y , x , and z axis, respectively. Note that in Eq. 9 the index k is omitted for the sake of readability.

To find the smoothed rotation matrices $R_k^{(s)}$, a regularization framework, as presented in [CLL06], is employed. The regularization framework smoothes each of the three Euler angles independently and smoothed rotation matrices are generated from the smoothed Euler angles, as outlined in Eq. (9). Using this approach yields a smooth stabilization similar to the results presented in [CLL06].

In our case, however, the fixation on a target point constrains the pan and tilt angle, and only the roll angle can still be chosen arbitrarily. Therefore, the pan and tilt angle are not smoothed but are directly obtained from the fixation on the target point.

To achieve this, we exploit the fact that the rotation matrix R_k can be expressed as

$$R_k = \begin{pmatrix} \mathbf{U}_k^\top \\ \mathbf{V}_k^\top \\ \mathbf{W}_k^\top \end{pmatrix}, \quad (10)$$

where \mathbf{U}_k , \mathbf{V}_k , and \mathbf{W}_k are the three axes spanning the corresponding camera coordinate frame. The optical axis $\mathbf{W}_k^{(f)}$ of the fixated camera coordinate frame

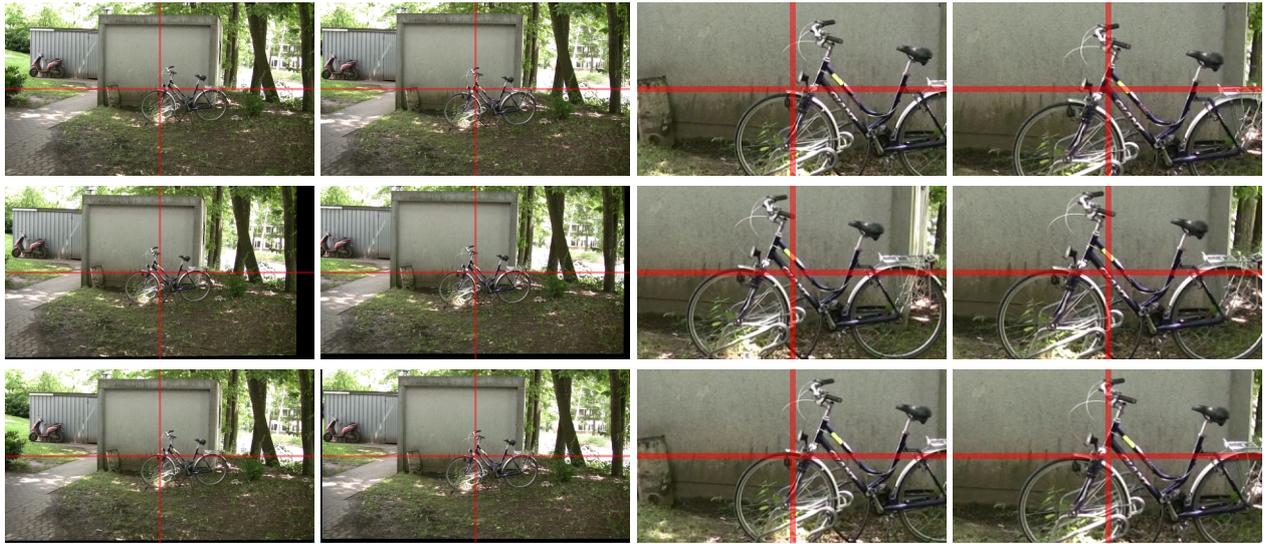


Figure 4: Example 1 - Original image sequence (top), result of stabilization by fixation (middle), result of smoothing with an affine model [CLL06] (bottom). The images on the right are magnifications. With the stabilization by fixation approach the center of the image is kept perfectly stable. The red marker lines were added to facilitate visual verification.

can be obtained using the relation

$$\mathbf{W}_k^{(f)} = \mathbf{T}_{i(k)} - \mathbf{C}_k \quad , \quad (11)$$

where $i(k)$ identifies the target point for camera image I_k . The other axes of the fixated coordinate system are calculated as

$$\mathbf{V}_k^{(f)} = \mathbf{W}_k^{(f)} \times \mathbf{U}_k \quad , \quad (12)$$

where \mathbf{U}_k is the horizontal axis of the original camera coordinate system and \times denotes the vector cross product, and

$$\mathbf{U}_k^{(f)} = \mathbf{V}_k^{(f)} \times \mathbf{W}_k^{(f)} \quad . \quad (13)$$

Once all three axes have been calculated, the corresponding Euler angels can easily be obtained.

As the fixation does not constrain the roll angle, in absence of other knowledge, the smoothed roll angle as given by the regularization framework is employed.

Since our approach perfectly stabilizes the given target point in the center of the corresponding images, it is clear that the transitions between adjacent target points can be very abrupt. In most cases this effect is not desired and a smooth transition between adjacent targets is preferred. This can be achieved by applying the regularization framework mentioned above on a short image sequence covering the transition. The user would thereby define the desired length of the transition as a number of images. We ensure that the transition images are taken from the last images corresponding to the current and the first images corresponding to

the next fixation point equally. Application of the regularization then yields the desired transition.

With the same technique, smoothed parameters can be calculated for longer parts of the image sequence that where not assigned to any target point.

7 Additional User Control

While the automatic methods introduced in Sections 4 and 5 do not require user interaction apart from selecting the threshold value τ , a certain amount of user interaction might be desirable at some point during the stabilization process. Since we are provided with a rich amount of information by the structure-from-motion reconstruction, the user is able to exert a high level of control on the stabilization.

7.1 User-specified Fixation Point

For example, the user is not restricted to using the 3D target points \mathbf{T}_i , neither virtual nor real, provided by the algorithm. Instead, the target points can be freely chosen from the full set of 3D object points \mathbf{P}_j , thereby allowing full control over the process and enabling specific stabilization requirements to be met.

7.2 Automatic Fixation on Arbitrary Targets

Bringing the concept of user-selected target points one step further, it is possible to ultimately specify 2D im-



Figure 5: Example 1 - Original image sequence (top), result of stabilization by fixation (middle), result of smoothing with an affine model [CLL06] (bottom).

age points on which the algorithm will then fixate during the stabilization process. Assuming a rotational stabilization model as specified in the previous section, the necessary corrections to the rotation matrix can easily be computed by treating the 2D image points as representative for all 3D points lying on the corresponding line of sight. Therefore, given a 2D image position \mathbf{x}_k , the rotation matrix $\mathbf{R}_k^{(f)}$ that fixates the camera in the 3D direction corresponding to this 2D image position can be obtained by using Eq. (10), where in contrast to Eq. (11) the optical axis of the fixated camera has to be expressed by

$$\mathbf{W}_k^{(f)} = \mathbf{S}_k, \quad (14)$$

where \mathbf{S}_k is the direction of the line of sight from the camera center \mathbf{C}_k through the 3D point lying on the image plane associated with \mathbf{x}_k .

The 2D image position \mathbf{x}_k can be specified independently for all images I_k , and therefore the selection of target points is no longer restricted to 3D points contained in the structure-from-motion reconstruction, i.e., elements of the static scene. Combined with state-of-the-art tracking techniques, our algorithm therefore yields a powerful tool to stabilize an image sequence while fixating arbitrary, stationary or moving objects. This is done by simply tracking the desired object through the image sequence and then providing the 2D image positions to our algorithm.

8 Results

In this section, we present four real-world examples of video stabilization by fixation. In addition, two examples featuring user interaction and control are given. Except examples 2 and 5, all examples are recorded with off-the-shelf consumer HDV cameras at a resolution of 1440×1080 pixels and a frame rate of 25 Hz. In examples 2 and 5 a SD camera with a resolution of 720×576 pixels was employed. The examples are also shown in the video provided with this submission.

Example 1 has a total length of 700 frames. With a threshold of $\tau = 5.0$ pixels eleven real target points were found. In Fig. 3 a comparison between the camera parameters estimated from the original image sequence, the smoothed parameters generated using the approach described in [CLL06], and the fixated parameters is shown. The deviation of the fixated parameters from the smoothed parameters is visible, especially in the shown detail magnification. Because the roll parameter is also smoothed during fixation the smoothed and fixated roll parameter curve are on top of each other.

For comparison, sample images of the stabilization by fixation approach are shown in Figures 4 and 5, along with the corresponding images obtained through stabilization with an affine model. To facilitate verification of the visual fixation, a red cross-hair at the center of the images is superimposed. It can be observed that the fixation approach, in contrast to the affine stabilization, keeps the same 3D location perfectly in the

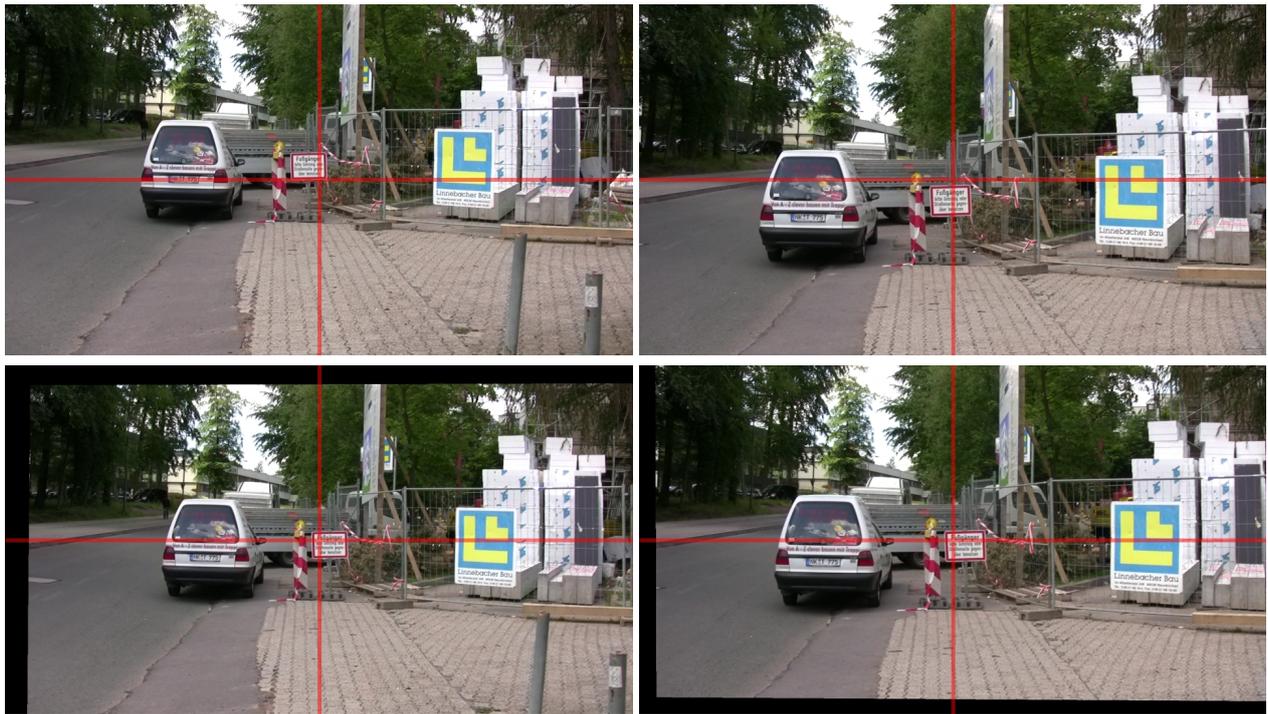


Figure 6: Example 3 - Original image sequence (top), result of stabilization by fixation (bottom).

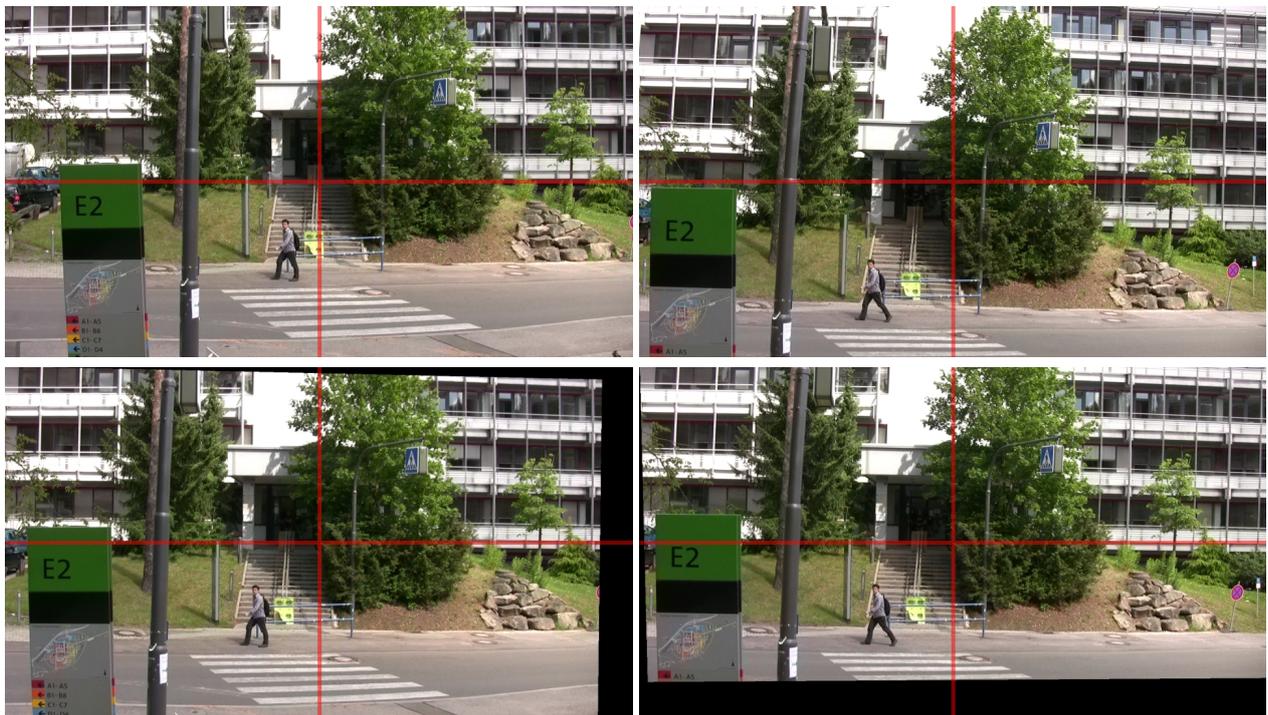


Figure 7: Example 4 - Original image sequence (top), result of stabilization by fixation (bottom).

image center.

Example 2 presents a sequence of 250 images with an approximate orbit motion around a dredger. A threshold of $\tau = 0.5$ pixels generated three real target points for stabilization by fixation. Sample im-

ages from the original and stabilized video are shown in Fig. 8.

In example 3 and 4 very strong camera shakes are compensated by our video stabilization approach. Therefore, a large threshold of $\tau = 50.0$ pixels was

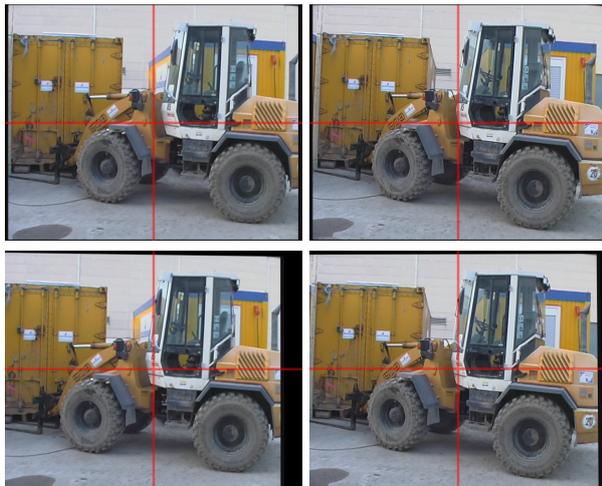


Figure 8: Example 2 - Original image sequence (top), result of stabilization by fixation (bottom).



Figure 9: Example 5 - Original image sequence (top), result of stabilization by fixation with user interaction (bottom). The arrow indicates the selected target point.

chosen. In example 3, shown in Fig. 6, two target points were established over a sequence of 212 images. In example 4, shown in Fig. 7, three target points were established over a sequence of 150 images.

In example 5, the video sequence already presented in example 2 is shown once again. In contrast to before, this time a specific point of the static scene that has been specified by the user is employed as target point for the stabilization algorithm. The point is located on the front tire of the dredger. Fig. 9 shows that the image is fixated perfectly onto the user-specified point throughout the sequence.

Example 6 features a video sequence of a dancing subject in a half pipe consisting of 400 images. A simple tracking algorithm based on mean shift track-

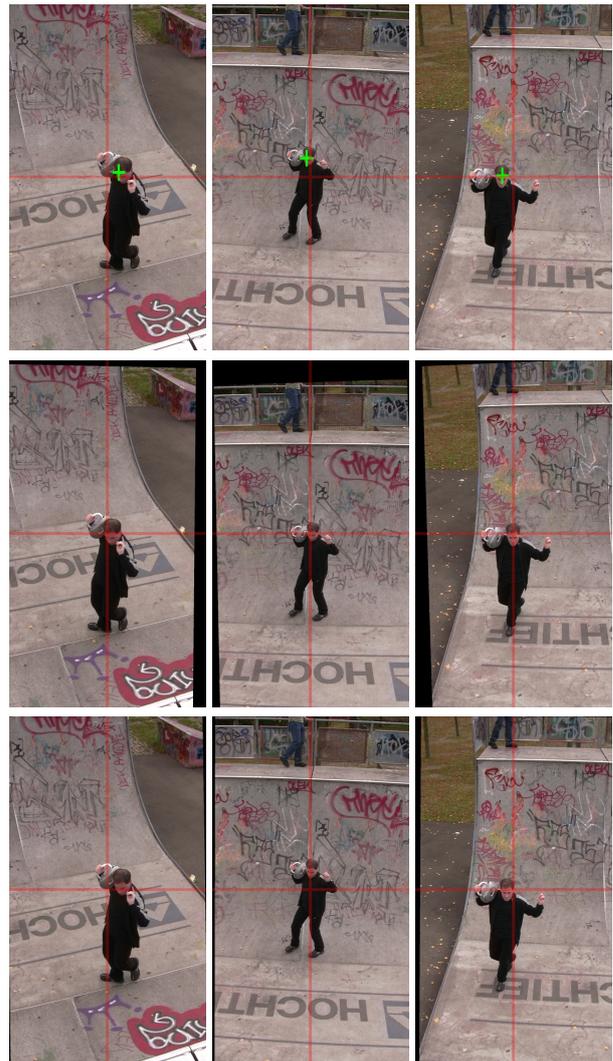


Figure 10: Example 6 - Original image sequence (top), result of stabilization by fixation (middle), result of smoothing with an affine model [CLL06] (bottom). The user-supplied tracking information is indicated by the green crosses in the top images. It is used to generate the result displayed in the middle row.

ing [CM97] is used to track the head of the subject. For each image in the input sequence an individual target point is created, guided by the tracking algorithm. As can be seen in Fig. 10, the video sequence is stabilized and fixated on the subject, albeit both the camera and the subject exhibit strong movement.

9 Conclusion

In this paper we presented a video stabilization approach that fixates the center of the image to a specific 3D target point. After analyzing the scene with

a structure-from-motion algorithm, these target points are automatically detected within the scene. The user can control how much the original sequence is altered by adjusting a single parameter τ . This user-supplied parameter specifies the maximum offset value of the projected target point to the image center in the original image sequence. In addition, various methods of additional user control were investigated. Apart from the automatic selection of virtual and real target points, the user has the possibility to choose a specific target to achieve a desired stabilization result. Furthermore, the algorithm can be combined with state-of-the-art tracking algorithms, yielding a powerful tool for image stabilization allowing the camera to fixate onto an arbitrary static or moving object.

In contrast to existing automatic approaches, our approach can achieve an absolutely stable result in the center of the images or the point the user has chosen as fixation target, respectively.

Using a single real 3D target point for stabilization may introduce a certain bias with respect to the actual position of the object of interest. The presented approach could possibly be further extended to take into account a group of 3D object points as a representation for the object of interest. The additional points could even be used as a measure to determine the camera roll angle, thereby enabling the approach to stabilize in-plane rotation beyond simple smoothing.

Another limitation of the approach is its dependency on the structure-from-motion algorithm. If this processing step provides wrong parameters, unpredictable results may occur. However, other automatic stabilization approaches are also dependent on reliable feature tracking. For scenes where the tracking of features is possible, state-of-the-art structure-from-motion also seldomly fails. If the camera performs a pure rotational motion, target points can not be found with the presented technique. However, similar techniques could be developed for this special case in the future.

Furthermore, our approach is offline by design. Even if the camera motion could be estimated in real-time, the process of target-point selection cannot be applied due to the lack of required input data.

A general problem, which occurs with all image stabilization techniques that apply a 2D transformation to the image, is that the translational motion of the camera and the resulting motion parallax can not be compensated. This can be perceived as residual jitter artifacts in some of the presented videos. These artifacts could only be removed if a high quality depth map

with occlusion information would be available for every pixel of all images (e.g., methods based on dense optical flow could deal with this issue in principle). This is left for future research.

Acknowledgements

This work has been partially funded by the Max Planck Center for Visual Computing and Communication (BMBF-FKZ01IMC01).

References

- [BBM01] Chris Buehler, Michael Bosse, and Leonard McMillan, *Non-Metric Image-Based Rendering for Video Stabilization*, IEEE Conference on Computer Vision and Pattern Recognition (Hawaii, USA), 2001, pp. 609–614, ISBN 0-7695-1272-0.
- [Car88] Roger H. S. Carpenter, *Movements of the Eyes*, 2nd ed., Pion, London, 1988, ISBN 0-85086-109-8.
- [CFR99] Alberto Censi, Andrea Fusiello, and Vito Roberto, *Image Stabilization by Features Tracking*, International Conference on Image Analysis and Processing (Venice, Italy), 1999, pp. 665–667, ISBN 0-7695-0040-4.
- [CLL06] Hung-Chang Chang, Shang-Hong Lai, and Kuang-Rong Lu, *A robust real-time video stabilization algorithm*, Journal on Visual Communications and Image Representation **17** (2006), no. 3, 659–673, ISSN 1047-3203.
- [CM97] Dorin Comaniciu and Peter Meer, *Robust Analysis of Feature Spaces: Color Image Segmentation*, IEEE Conference on Computer Vision and Pattern Recognition (San Juan, Puerto Rico), 1997, pp. 750–755, ISBN 0-8186-7822-4.
- [CW08] Jinhai Cai and Rodney A. Walker, *Robust Motion Estimation for Camcorders Mounted in Mobile Platforms*, Digital Image Computing: Techniques and Applications (DICTA), 2008, pp. 491–497, ISBN 978-0-7695-3456-5.

- [dBJSG08] Carlos R. del Blanco, Fernando Jau-reguizar, Luis Salgado, and Narciso Garcia, *Automatic Feature-Based Stabilization of Video with Intentional Motion through a Particle Filter*, International Conference on Advanced Concepts for Intelligent Vision Systems, vol. 5259, 2008, pp. 356–367, ISBN 978-3-540-88457-6.
- [Ert02] Sarp Ertürk, *Real-Time Digital Image Stabilization Using Kalman Filters*, Journal of Real Time Imaging **8** (2002), no. 4, 317–328, ISSN 1077-2014.
- [Fou91] Antoine Fournier, *Image stabilizing apparatus for a portable video camera*, US Patent 5012347, 1991.
- [GCH⁺02] Simon Gibson, Jon Cook, Toby Howard, Roger Hubbard, and Dan Oram, *Accurate Camera Calibration for Off-line, Video-Based Augmented Reality*, IEEE and ACM International Symposium on Mixed and Augmented Reality (Darmstadt, Germany), 2002, p. 37, ISBN 0-7695-1781-1.
- [JZX01] Jesse S. Jin, Zhigang Zhu, and Guangyou Xu, *Digital Video Sequence Stabilization Based on 2.5D Motion Estimation and Inertial Motion Filtering*, Journal of Real Time Imaging **7** (2001), no. 4, 357–365, ISSN 1077-2014.
- [KKM04] Toshimichi Kudo, Hideo Kawahara, and Junichi Murakami, *Vibration correction apparatus*, US Patent 6734901, 2004.
- [Krü99] Wolfgang Krüger, *Robust real-time ground plane motion compensation from a moving vehicle*, Machine Vision and Applications **11** (1999), no. 4, 203–212, ISSN 0932-8092.
- [LCCO09] Ken-Yi Lee, Yung-Yu Chuang, Bing-Yu Chen, and Ming Ouhyoung, *Video Stabilization using Robust Feature Trajectories*, IEEE International Conference on Computer Vision, 2009, ISSN 1550-5499, pp. 1397–1404.
- [LGJA09] Feng Liu, Michael Gleicher, Hailin Jin, and Aseem Agarwala, *Content-Preserving Warps for 3D Video Stabilization*, ACM Transactions on Graphics (Proceedings of SIGGRAPH 2009), 2009, pp. 44:1–44:9, ISBN 978-1-60558-726-4.
- [LHY09] Chin-Teng Lin, Chao-Ting Hong, and Chien-Ting Yang, *Real-Time Digital Image Stabilization System Using Modified Proportional Integrated Controller*, IEEE Transactions on Circuits and Systems for Video Technology **19** (2009), no. 3, 427–431, ISSN 1051-8215.
- [LPLP09] Jinhee Lee, Younguk Park, Sangkeun Lee, and Joonki Paik, *Statistical region selection for robust image stabilization using feature-histogram*, International Conference on Image Processing, 2009, pp. 1553–1556, ISBN 978-1-4244-5653-6.
- [LWV98] Chiou Peng Lam, Geoff A. W. West, and Svetha Venkatesh, *Stabilising the Camera-to-Fixation Point Distance in Active Vision*, Pattern Recognition **31** (1998), no. 10, 1431–1442, ISSN 0031-3203.
- [MC97] Carlos Morimoto and Rama Chellappa, *Fast 3D Stabilization and Mosaic Construction*, IEEE Conference on Computer Vision and Pattern Recognition (San Juan, Puerto Rico), 1997, pp. 660–665, ISBN 0-8186-7822-4.
- [PGV⁺04] Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, Frank Verbiest, Kurt Cornelis, Jan Tops, and Reinhard Koch, *Visual modeling with a hand-held camera*, International Journal of Computer Vision **59** (2004), no. 3, 207–232, ISSN 0920-5691.
- [Pil04] Maurizio Pilu, *Video stabilization as a variational problem and numerical solution with the viterbi method*, IEEE Conference on Computer Vision and Pattern Recognition (Washington DC, USA), vol. 1, 2004, pp. 625–630, ISBN 0-7695-2158-4.

- [PLH07] Karl Pauwels, Markus Lappe, and Marc M. Van Hulle, *Fixation as a Mechanism for Stabilization of Short Image Sequences*, International Journal of Computer Vision **72** (2007), no. 1, 67–78, ISSN 0920-5691.
- [ST94] Jianbo Shi and Carlo Tomasi, *Good Features to Track*, IEEE Conference on Computer Vision and Pattern Recognition (Seattle, USA), 1994, pp. 593–600, ISBN 0-8186-5825-8.
- [THWS08] Thorsten Thormählen, Nils Hasler, Michael Wand, and Hans-Peter Seidel, *Merging of Feature Tracks for Camera Motion Estimation from Video*, European Conference on Visual Media Production (London, UK), 2008, ISBN 978-0-86341-973-7.
- [WCCF09] J. M. Wang, H. P. Chou, S. W. Chen, and C. S. Fuh, *Video stabilization for a handheld camera based on 3D motion model*, International Conference on Image Processing, 2009, pp. 3477–3480, ISBN 978-1-4244-5653-6.

| Citation |
|--|
| Christian Kurz, Thorsten Thormählen, and Hans-Peter Seidel, <i>Visual Fixation for 3D Video Stabilization</i> , Journal of Virtual Reality and Broadcasting, 8(2011), no. 2, February 2011, urn:nbn:de:0009-6-28222, ISSN 1860-2037. |