Information & Security

Published since 1998

D. Schreiber, A. Opitz & S. Veigl
vol. 55, no. 1 (2024): 95-109
https://doi.org/10.11610/isij.5542
ISSN 0861-5160 (print), ISSN 1314-2119 (online)

Research Article

Digital Stabilization of Thermal Videos for Border and Infrastructure Surveillance

David Schreiber, Andreas Opitz, and Stephan Veigl (✉)

AIT Austrian Institute of Technology GmbH, 1210 Vienna, Austria
https://www.ait.ac.at

ABSTRACT:

We present a video stabilization algorithm for a static thermal camera intended for a practical surveillance system, e.g., for border or infrastructure surveillance. In these scenarios, the camera is typically mounted on a pole and is therefore prone to considerable shaking due to high winds. These shakes correspond to erratic misalignment of the video frames and can degrade the performance of detection and tracking algorithms applied on the video frames, as well as annoy the security personnel monitoring the videos on a screen. Thermal images are difficult to process because these images are of low quality, have poor contrast and brightness, contain a smaller number of features, and are noisy. The proposed algorithm, based on a direct method, is simple yet fast and robust, overcoming these limitations and providing considerable frame alignment even in the case of severe camera shake, as demonstrated by experiments with a randomly added shake to the frames as well as with mechanical shake applied to the pole.

ARTICLE INFO:

RECEIVED:	18 SEP 2024
REVISED:	23 OCT 2024
ONLINE:	30 OCT 2024

KEYWORDS:

thermal camera, video stabilization, direct methods, image alignment, border surveillance, infrastructure surveillance

✉ E-mail: stephan.veigl@ait.ac.at

Introduction

Video stabilization is an important task of video-based outdoor surveillance, e.g., surveillance of borders and infrastructures (e.g., train stations). The proposed algorithm for video stabilization on a thermal camera was developed within two such projects: MOBILIZE (which is concerned with maintaining the operational safety of large railway systems against acts of sabotage, vandalism, or unauthorized entering/crossing of the tracks),[16] and EURMARS (improving border authority surveillance capabilities by enhancing critical quality characteristics of aerial and ground-based sensor platforms). [15] In such use cases, the sensors are typically mounted on a pole and are susceptible to harsh weather conditions and high winds, resulting in camera shake which leads to unwanted motion between successive video frames. On the one hand, this unwanted motion and jitter annoys the security personnel who observe the scenes on monitors. On the other hand, it deteriorates AI-based algorithms, such as object detection and tracking, which process these video frames.

There are three basic techniques available for video stabilization: optical, electronic, and digital image stabilization (OIS, EIS, and DIS, respectively). In the OIS method, an optical lens assembly compensates the disturbance in the opposite direction. This technique is suitable for very low amplitude vibrations such as atmospheric turbulence. In the EIS method, the camera is itself packaged in a mechanical housing equipped with a rate of disturbance sensor (gyro), motor assembly, and servo electronics. This method requires an expensive mechanical gimbal, sensors, actuators, and servo electronics, and depends on the camera's physical dimensions. The third type of stabilization method, i.e., DIS, is purely software-based and independent of the camera's physical dimensions. The advantages of using DIS techniques are that there are no moving and costly components and the ability to apply different algorithms to improve the stabilization.[12]

Digital stabilization methods can be classified into two-dimensional, three-dimensional, or hybrid methods. Two-dimensional methods are based on the 2D apparent motion between 2 video frames. However, 2D approaches cannot model the parallax generated by translational shifts, and thus only generally approximate 3D motion. Therefore, 3D methods were proposed to solve this problem. However, modeling parallax from a video sequence is a time-consuming and difficult process that may fail in many cases. Three-dimensional methods can present serious problems handling large objects in the foreground, in addition to having a higher computational cost. Thus, hybrid methods (2.5D) were proposed to transform 2D motion trajectories to achieve visually plausible views of 3D methods and robustness of 2D methods, but at the cost of distortion and physically inaccurate results.[6]

Two-dimensional stabilization methods are widely implemented in commercial packages due to their robustness and low cost. The main issues that are open challenges to the motion estimation between 2 frames, and thus to the video stabilization task include: low-textured background, large moving objects, large variations in object depths, depth and displacement ambiguity, large parallax, and the rolling shutter effect. Two-dimensional camera motion estimation can be classified as (i) global, where a single transformation matrix is computed for the entire frame, and (ii) local, with a matrix per image region. Global motion estimation is sufficient to compensate for camera motion in scenarios where the entire image is equally affected, i.e., when there is little depth variation or when the 3D compensation is only rotational. In these cases, global motion approach is attractive, as it can represent the motion accurately with only a few parameters.[6] The 2D methods can be classified as intensity-based, which align video frames through their raw intensities, and local features-based, which compute a sparse set of local features for the alignment. Approaches that use local features methods are more common in the literature. In this approach, a sparse set of distinct features are first extracted from each image separately, then their correspondences are analyzed. Finally, based on these correspondences, the motion is estimated.[6] In contrast to the global motion approaches outlined above, the local camera motion approach estimates the motion locally, where the motion can be represented at different levels of granularity, e.g., pixels (optical flow) or meshes/cells. Local motion approaches deal with the challenge of large parallax, and some also with the rolling shutter challenge; however, they are prone to the risk of not having sufficient features in every cell.[6]

The video stabilization problem can be tackled either by traditional methods, or by deep learning-based approaches. Deep neural networks have been demonstrated to be a powerful framework for solving various challenges in computer vision. However, deep neural networks require high computational overhead, special hardware such as expensive GPUs and extensive annotated dataset for training. Moreover, deep neural networks outperform conventional approaches when they are trained with a scene-specific datasets. These limitations render deep learning-based approaches impractical for border surveillance applications such as the target of the MOBILZE and EURMARS projects. Therefore, we focus on the state-of-the-art of conventional methods.

The state-of-the-art in video stabilization is well presented in the recent survey articles,[6,22] although they focus on RGB sensors. Much less work has been done regarding the stabilization of thermal videos. As border and infrastructure surveillance is needed 24 h per day, thermal cameras are used as they provide viewing conditions at pitch dark as well. However, thermal images are difficult to process as these images are of low quality, poor contrast, contain a smaller number of features, and are noisy.

In this paper, we present an online video stabilization approach for a static thermal camera mounted on a pole that is aimed at a practical surveillance system. Our approach is based on direct methods, which entails the estimation of motion parameters between two frames directly from measurable image quantities (pixel intensities), in contrast to “feature-based methods,” which first extract a sparse set of distinct features (key points) from each image separately, and then recovers and analyses their correspondences to determine the motion.[10] The proposed algorithm is simple yet real-time and robust, overcoming the limitations of thermal images and providing accurate frame alignment even in the case of severe camera shake, as demonstrated by experiments with a randomly added shake to the frames as well as with the application of mechanical shake to the pole. Our approach is appropriate for the stabilization of RGB videos as well; however, we focus here on thermal videos.

The rest of this paper is organized as follows. Section 0 reviews the state-of-the-art digital image stabilization, particularly for thermal sensors. Section 0 outlines our approach for thermal video stabilization. Section 0 presents experiments and an evaluation of our approach. Section 0 contains conclusions and future work.

Related Work

Most of the state-of-the-art approaches of 2D video stabilization methods were developed for RGB sensors.[6, 22] In contrast, thermal images are difficult to process as they are of lower resolution, lower quality, and contrast, contain a smaller number of features, and are noisy. State-of-the-art methods developed for RGB videos are bound to perform poorly on thermal images due to these issues. In fact, only a small number of previous approaches focused on the video stabilization problem for thermal sensors, and they commonly employ more straightforward methods. These methods were developed mainly for UAV-based applications such as monitoring of wildfires and aerial and border military surveillance.

Several stabilization approaches are based on block matching. Marcenaro used two methods to estimate motion between two consecutive frames: in the grid approach, a grid of points is placed on both the reference and the current frame, and a set of translations is applied to the grid and a correlation index is calculated.[13] The second approach is based on detection and tracking of feature points. Only translational motion is assumed. Estalayo et al. introduced a method for the stabilization of aerial FLIR videos using a multi-resolution pyramid to reduce computation cost.[7] First, an estimation of the local motions is carried out using block matching via the minimum absolute difference function. Next, the camera motion is estimated to determine if the global motion is pure translational or if it also includes rotational components. Shen and coworkers used a block matching technique with polynomial smoothing for stabilization, assuming a similarity transformation between two frames.[17]

Other approaches combine block matching with spectral methods. Fang and Xiaozhen proposed an electronic image stabilization algorithm based on efficient block matching on the plane.[8] This algorithm uses a hexagonal search algorithm and the bit-planes to estimate and compensate for the translational motion between video sequences at the same time. Next, the algorithm conducts the Laplace transform for the reference frame, selects several characteristics at the image edge to make block matching with the current frame, calculates and compensates for the rotational movement that may exist finally. The method of Kang and Park uses SAD block matching, assuming rotational and translational motion.[11] Frequency data is used to differentiate between the vibrations of the camera and global motion, namely by choosing the image blocks for the estimation of the global motion. Dios and Ollero employed the Fourier-Mellin transform performed on entire images, assuming translation and rotation motion.[5] However, performing the transformation on entire images is susceptible to outliers such as moving objects. A real-time video stabilization algorithm implemented on an FPGA is discussed by Araneda and Figueroa,[1] who used integral projections and the sum of absolute differences method for estimating motion.

A major part of previous works employed key-point detection and matching. Yao, Hinz, and Stilla proposed stabilization for the analysis of traffic scenarios.[25] Feature points are detected via the Foerstner operator and a projective transformation between the 2 images is computed using RANSAC. The point correspondence is further used for estimating an affine transformation. Hong, Hong, and Yang provided a multiresolution video stabilization algorithm based on the Scale Invariant Feature Transform (SIFT) and assuming an affine transformation.[9] Haar Wavelet decomposition algorithm is used to filters out outliers. Wang, Hou, et al. use corner point detection and matching with a cubic spline for smoothing, with motion model consists of similarity transformation.[23] Zhou and Asari compared SIFT and SURF for motion estimation between frames and used Motion Vector Integration (MVI) with adaption damping for smoothing.[27] Walha, Wali, and Alimi used SIFT correspondences, assuming a similarity transformation.[21] The method of Xie et al. is based on speeded-up robust features (SURF), which are extracted and tracked in each frame, where the matching is refined via RANSAC, estimating the motion parameters of a 2D affine model.[24] Thillainayagi and Senthil used Scale Invariant Feature transform (SIFT) for key-point detection and matching between successive frames.[19] Then, the affine transformation model is used to estimate the global motion parameters between two successive frames. Deng et al. proposed a real-time image stabilization method based on optical flow and image matching with binary feature descriptors.[4] The global motion (similarity transformation) of consecutive frames is estimated by the pyramid Lucas-Kanade optical flow algorithm, and the interval frames image matching based on fast retina key-point (FREAK) algorithm is used to reduce the cumulative trajectory error. Khare, Singh, and Kaushik describe a fast stabilization approach for a thermal camera mounted on a vehicle, in the context of border military surveillance.[12] This algorithm employs correspondences between SURF features to estimate translation as well as rotational motion. Valero et al. compared and evaluated several approaches for thermal infrared video stabilization for aerial monitoring of active wildfires, including phase correlation and the Fourier-Mellin transform, and the SIFT, SURF, MSER, and KAZE features, and assuming 2D similarity transformation.[20] Best performance was achieved using the KAZE features where outliers during matching were discarded by a RANSAC algorithm.

Yousaf and coworkers reviewed real-time video stabilization methods for UAVs in the IR domain.[26] They note that stabilizing Infrared videos for real-time applications is still an open research area. This is because of the difficulty in detecting and tracking good features in IR videos due to lower resolution/ quality videos. Most of the available image stabilizing techniques are limited by several assumptions such as a high number of feature points throughout the video sequence. Available techniques fail in many challenging scenarios, such as large depth variation, quick camera motion, large moving foreground, motion blur, and rolling shutter effects. Most of the existing methods that achieve prominent performance have a high computational cost and are implemented as offline techniques.

Proposed Stabilization Algorithm

Stabilizing infrared videos for real-time applications is still an open research area. This is due to the difficulty in detecting and tracking good features in IR videos having low resolution and poor quality. Moreover, using key-points for local motion estimation is even less feasible, as not enough information would be available in each image region. For the MOBILIZE and EURMARS projects, a fast real-time online approach is needed that does not require offline training for each individual scenario, is easy to install and set up, and could run with modest computational resources by the end user. Hence, we seek a global 2D motion estimation.

Consequently, the 2D direct method approach has been chosen. *Direct Methods* are defined as methods for motion and/or shape estimation, which recover the unknown parameters directly from measurable image quantities at each pixel in the image. Feature-based methods minimize an error measure that is based on distances between a few corresponding features, while direct methods minimize an error measure that is based on direct image information collected from all pixels in the region of interest in the image. Direct methods were popular in the 1990s due to their high sub-pixel accuracy and their ability to lock on to a single dominant global motion even when multiple motions (outliers) are present.[10]

The extended Lucas-Kanade algorithm is employed to estimate the parameters of the global motion between two frames directly. Whereas the original Lucas-Kanade algorithm assumed pure translational motion and, therefore, used small patches in the image, the extended algorithm allows any motion model using arbitrarily large regions in the image. In particular, the inverse compositional algorithm proposed by Baker and Matthews[2] is a more efficient version of the algorithm, where the roles of the reference image and the new image are switched and as a result, the Hessian need not be updated at each iteration. We adopt the notation of Baker und Matthews.[2] Let I_1 and I_2 denote the reference and the second image, respectively, and let $\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})$ denote the parameterized set of allowed warps, where $\boldsymbol{p}$ is a vector of motion parameters. In the inverse compositional algorithm, the following quantity is minimized:

$$SSD = \sum_x \left[I_1\left(\boldsymbol{W}(\boldsymbol{x};\Delta\boldsymbol{p})\right) - I_2\left(\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})\right)\right]^2 \qquad (1)$$

with respect to Δp, and then the warp is updated:

$$\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p}) \leftarrow \boldsymbol{W}(\boldsymbol{x};\boldsymbol{p}) \circ \boldsymbol{W}(\boldsymbol{x};\boldsymbol{\Delta p})^{-1} \tag{2}$$

Performing a Taylor expansion on Eq. (1) gives:

$$SSD \approx \sum_{\boldsymbol{x}} \left[I_1\big(W(\boldsymbol{x};\boldsymbol{0})\big) + \nabla I_1 \frac{\partial \boldsymbol{W}}{\partial \boldsymbol{p}} \boldsymbol{\Delta p} - I_2\big(\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})\big) \right]^2 \tag{3}$$

where the least-squares solution is:

$$\boldsymbol{\Delta p} = H^{-1} \sum_{\boldsymbol{x}} \left[\nabla I_1 \frac{\partial \boldsymbol{W}}{\partial \boldsymbol{p}} \right]^T \left[I_2\big(\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})\big) - I_1(\boldsymbol{x}) \right] \tag{4}$$

where the Hessian H is given by:

$$H = \sum_{\boldsymbol{x}} \left[\nabla I_1 \frac{\partial \boldsymbol{W}}{\partial \boldsymbol{p}} \right]^T \left[\nabla I_1 \frac{\partial \boldsymbol{W}}{\partial \boldsymbol{p}} \right] \tag{5}$$

To overcome occlusions, errors in the boundary of the template, and violations of the brightness constancy assumption, the robust version of the Lucas-Kanade algorithm makes use of a weighted least square process, where such problematic pixels are considered as outliers and therefore suppressed in the computation, e.g., using an iterated reweighting approach.[3]

$$\boldsymbol{\Delta p} = H^{-1} \sum_{\boldsymbol{x}} \omega_{\boldsymbol{x}} \left[\nabla I_1 \frac{\partial \boldsymbol{W}}{\partial \boldsymbol{p}} \right]^T \left[I_2\big(\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p})\big) - I_1(\boldsymbol{x}) \right] \tag{6}$$

$$H = \sum_{\boldsymbol{x}} \omega_{\boldsymbol{x}} \left[\nabla I_1 \frac{\partial \boldsymbol{W}}{\partial \boldsymbol{p}} \right]^T \left[\nabla I_1 \frac{\partial \boldsymbol{W}}{\partial \boldsymbol{p}} \right] \tag{7}$$

The robust weights $\omega_{\boldsymbol{x}}$ are recomputed each iteration, and hence, in this case, the Hessian needs to be recomputed too.

The most general 2D motion model is affine transformation, comprised of 6 motion parameters. However, the problem is that due to the large number of parameters, it is hard to devise a robust method that rejects outliers. Reducing the number of parameters also eliminates the inherent ambiguity between rotations and translations for a moving plane.[14, 18] It is, therefore, beneficial to resort to a 3D motion model (the motion of the image plane in the 3D world) instead to gain an understanding of the dominant motion parameters and the possible motion ambiguities. Given a plane moving relative to the camera in the 3D world, where the motion is carried out during a short interval of time, the motion field of a pixel in the image plane is described by Longuet-Higgins:[14]

$$u = -\omega_1 \left(\frac{xy}{f}\right) - \omega_2 \left(f + \frac{x^2}{f}\right) + \omega_3 y - f \frac{T_1}{Z} - x \frac{T_3}{Z} \tag{8}$$

$$v = -\omega_1 \left(f + \frac{y^2}{f}\right) - \omega_2 \frac{xy}{f} - \omega_3 x + f \frac{T_2}{Z} - y \frac{T_3}{Z}\mathrm{x} \tag{9}$$

f and Z are the focal length and 3D depth, respectively; $(\frac{T_1}{Z}, \frac{T_2}{Z}, \frac{T_3}{Z})$ are the 3D-scaled translations, and $(\omega_1, \omega_2, \omega_3)$ are the 3D rotations. In the most general case, not only is the 3D motion of the plane unknown, but so is its orientation. In such a case, Eqs. (8) and (9) should be modified. However, when the scene is far away from the camera and its depth variations are relatively small, it suffices to assume the plane to be parallel to the image plane.

Given that the motion between subsequent frames is sufficiently small (high framerate and using a multi-resolution pyramid), and assuming small depth variations relative to the distance of the scene from the camera, the three dominant motion parameters are the x and y translations, as well as the rotation within the image plane. With this choice, the motion ambiguity between rotations and translations is removed. Our warp function reduces to:

$$\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p}) = \begin{bmatrix} cos(\theta) & sin(\theta) & a \\ -sin(\theta) & cos(\theta) & b \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{10}$$

where the update rule becomes:

$$\boldsymbol{W}(\boldsymbol{x};\boldsymbol{p}) \circ \boldsymbol{W}(\boldsymbol{x};\Delta\boldsymbol{p})^{-1} = \begin{bmatrix} cos(\theta-\Delta\theta) & sin(\theta-\Delta\theta) & a-\Delta a\cos(\theta-\Delta\theta)-\Delta b\sin(\theta-\Delta\theta) \\ -\sin(\theta-\Delta\theta) & cos(\theta-\Delta\theta) & b-\Delta b\cos(\theta-\Delta\theta)+\Delta a\sin(\theta-\Delta\theta) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{11}$$

Note, that in Eq. (11), an exact transformation expression is used rather than an approximated one, as is common in the literature.[2] When using the first image of the video as a reference template, the motion between reference and current images might be large, e.g., when the pole on which the sensor is mounted performs large oscillations.

Evaluation

Quantitative Evaluation

We have used the AXIS Q1952-E thermal camera [28] with a resolution of 640x480. The Lucas-Kanade algorithm was implemented on a multi-scale pyramid to cope with large inter-frame motion. The number of the pyramid's levels is limited by the sensor resolution. A 4-level pyramid was employed, where the lowest resolution is 80x60 pixels. For the evaluation, a thermal video was recorded from a high floor of a building, mimicking a border surveillance scenario with a surveillance tower, although the scenario contains considerable depth variations, more than expected to be present in a border surveillance scenario (see Fig. 1, top). Such a scenario would be susceptible to a large parallax error. Still, for evaluation purposes, it does not matter, as the camera shake is simulated by adding randomly generated image transformations. To estimate the accuracy of the stabilization algorithm, a random artificial shake was added to each frame, according to Eq. (10). As during the recording, a slight wind was blowing, the stabilization algorithm was first running on the originally recorded frames, and the shake due to the wind was estimated. Next, an artificial shake was generated by random Gaussian noise, with a standard deviation of 5 pixels for the translations and 3 degrees for the rotation. During the evaluation, the ground truth (randomly applied transformations) was compared to the motion estimation of the stabilization algorithm for each frame after the estimated motion due to the wind was factored out.

The stabilization of all video frames was performed relative to the first image of the video. The region of interest in the image, which serves as the reference template for the registration algorithm, was chosen manually (see Fig. 1, top image).

The region of interest could be found automatically by searching for image regions that contain sufficient gradients in both the horizontal and vertical axes. The size of the template is 200x117 pixels. The robust iterated algorithm (Eqs. 6-7) was employed, where robust weights were calculated only for the two highest resolutions of the image pyramid, using a binary robust estimator. Even when employing the robust iterated Lucas-Kanade algorithm, where the Hessian needs to be recomputed at each iteration, a straightforward MATLAB implementation achieved a runtime of 51 FPS on a desktop PC with Intel(R) Xeon(R) CPU E5-1620 v3 @ 3.50GHz, 64-Bit processor and 32,0 GB RAM. About 70 % of the runtime was consumed by the image warping – the *interp2* MATLAB function. A speedup by a factor of up to 10 is expected when using optimized implementation.

Next, the estimated accuracy of the stabilization algorithm is presented on a video clip of 1000 frames. Table 1 (2nd to 4th rows) shows the discrepancy between the estimated motion parameters and the ground truth. As can be seen, the error of estimation is less than 0.1 pixels for the x and y translations, and less than 0.1 degrees for the rotation. As the algorithm was applied twice in a row, first to estimate the motion due to the wind and secondly to estimate the combined motion of the wind and the applied random transformation, the combined error was divided by a factor of $\sqrt{2}$.

Qualitative Evaluation

During the recordings, the thermal sensor has been affected only by slight wind. To test the stabilization algorithm under significant and more realistic sensor shake, additional mechanical shakes were applied manually to the pole on which the sensor was mounted. The shakes were applied in various magnitudes, involving rotations and translations of the pole as much as possible. Figure 2 shows some exemplary images of the un-stabilized (left) vs. stabilized frames (right). The stabilized frames are cropped due to the existence of blank borders. For the three frames shown in Figure 2, the motion estimated provided by the algorithm relative to the reference frame was as follows:

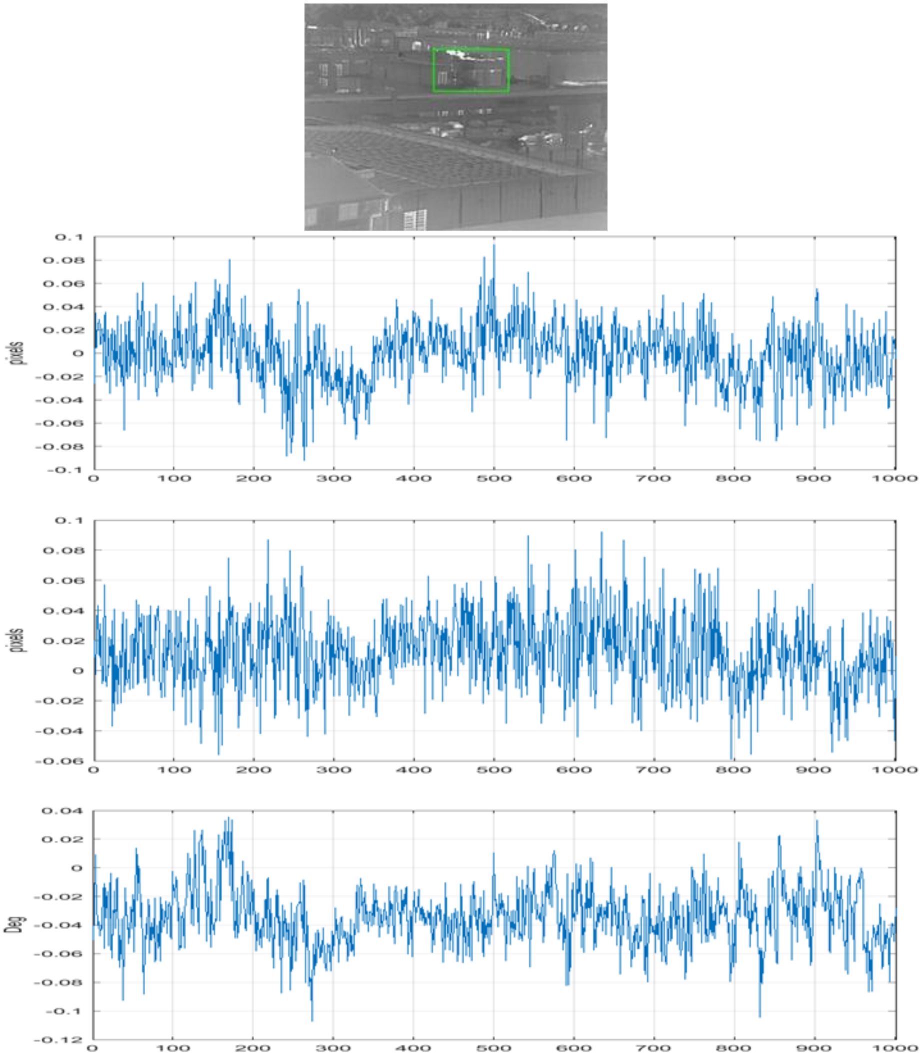

Figure 1. Recorded scenario with (green) reference template (top), error of the estimated motion parameters x- and y-translations (second and third row), and rotation (bottom)

Qualitatively, the algorithm was able to converge on most of the frames of the recorded video, and in those cases, the frames seemed well stabilized. On several frames where the sensor was treated by an especially violent shake, the algorithm did not converge. As our approach is based on 2D global motion estimation, and as the recorded scene has large depth variations and the reference template used corresponds to distant objects, the parallax distortions were apparent on objects closer to the camera.

Table 1. Motion estimation for the exemplary frames shown in Figure 2.

	x-translation (pixels)	y-translation (pixels)	rotation (degrees)
frame 1 (top)	2.0	23.7	-0.1
frame 2 (middle)	39.8	-11.7	-0.5
frame 3 (bottom)	-14.2	-15.4	0.9

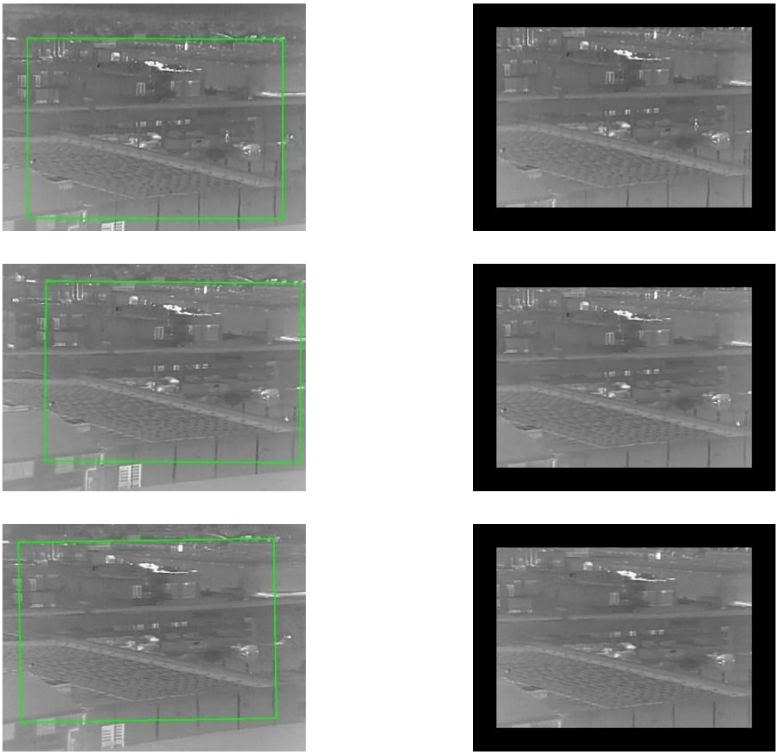

Figure 2: Mechanical shaking of the pole: Exemplary images of un-stabilized frames on the left and stabilized frames on the right).

A qualitative comparison of the proposed stabilization algorithm against the provided built-in stabilization mechanism of the sensor itself clearly showed that, in contrast to the proposed method, the built-in stabilization of the camera was not able to cope with the large shakes which were applied to the pole.

Conclusions and Future Work

Stabilizing infrared videos for real-time applications is still an open research area. This is due to the difficulty in detecting and tracking good features in IR videos having low resolution and poor quality. Moreover, using key-points for local motion estimation is even less feasible, as not enough information would be available in each image region. For surveillance of borders and infrastructures, a fast real-time approach is needed that does not require offline training, is easy to install and set up, and could run with modest computational resources by the end user. We have implemented such a video stabilization algorithm, which estimates the global 2D motion of video frames relative to a reference frame based on a robust extended Lucas-Kanade algorithm. The algorithm is evaluated both quantitatively (by adding random motion to the images) and qualitatively (by manually shaking the pole on which the camera is mounted) and is shown to handle severe shakes. The magnitude of shake, which can be coped with by the algorithm, is limited by the sensor's resolution. For example, doubling the resolution would enable adding one level to the image pyramid on which the algorithm runs and thus increase the magnitude of the shakes by a factor of 2.

Currently, the sub-region of the reference frame (reference template) is chosen manually. It is conceivable to automatically select a reference template by searching for a sub-region in the image which has gradients of sufficient magnitude, and presumably lies at the upper region of the frame, thus corresponding to a more distant scene that is less likely to be impeded by moving objects such as persons or vehicles). Alternatively, several sub-regions could be selected and used either simultaneously for a global or separately for a local motion estimation.

Future work will also include the integration of global navigation satellite system (GNSS) information. One of the advantages of using 3D motion parameters, as in Eq.(8) and (9), is that the 3D rotation parameters can be obtained from a GNSS device adjacent to and synchronized with the sensor. When the 3D rotations are provided by the GNSS device, the stabilization algorithm could be modified to estimate the remaining three 3D scaled translations. Employing the full 6 degrees of freedom would increase the precision and robustness of the algorithm.

Acknowledgements

This research was jointly carried out in the scope of the following two projects:

FFG Forschung wirkt.

Bundesministerium Klimaschutz, Umwelt, Energie, Mobilität, Innovation und Technologie

The project MOBILIZE is funded by the Austrian "Mobility of the Future" program – an initiative of the Federal Ministry for Climate Protection, Environment, Energy, Mobility, Innovation and Technology (BMK)

Co-funded by the European Union

EURMARS has received funding from the Europe-an Union's Horizon 2020 research and innovation programme under grant agreement No 101073985

References

1. Luis Araneda and Miguel Figueroa, "Real-Time Digital Video Stabilization on an FPGA," *17th Euromicro Conference on Digital System Design*, 2014, pp. 90-97, https://doi.org/10.1109/DSD.2014.26.

2. Simon Baker and Iain Matthews, "20 Years On: A Unifying Framework," *International Journal of Computer Vision* 56 (2004): 221-255, https://doi.org/10.1023/B:VISI.0000011205.11775.fdV .

3. Simon Baker, Ralph Gross, Iain Matthews, and Takahiro Ishikawa, "Lucas-Kanad 20 Years on: A Unifying Framework: Part 2," Tech. Report, CMU-RI-TR-03-01, 2003, Robotics Institute, Carnegie Mellon University.

4. Zilong Deng, Dongxiao Yang, Xiaohu Zhang, Yuguang Dong, Chengbo Liu, and Qiang Shen, "Real-Time Image Stabilization Method Based on Optical Flow and Binary Point Feature Matching," *Electronics* 9, no. 1 (2020), https://doi.org/10.3390/electronics9010198.

5. J. Ramiro Martinez-de Dios and Anibal Ollero, "A Technique for Stabilization of Sequences of Infrared Images Taken with Hovering UAVs," World Automation Congress, WAC '06, 2006, pp. 1-6, https://doi.org/10.1109/WAC.2006.375999.

6. Marcos Roberto e Souza, Helena de Almeida Maia, and Helio Pedrini, "Survey on Digital Video Stabilization: Concepts, Methods, and Challenges," *ACM Computing Surveys* 55, no. 3 (2023), https://doi.org/10.1145/3494525.

7. Enrique Estalayo, Luis Salgado, Fernando Jaureguizar, and Narciso García, "Efficient image stabilization and automatic target detection in aerial FLIR sequences," SPIE - The International Society for Optical Engineering, June 2006, https://doi.org/10.1117/12.665817.

8. Luo Fang and Qin Xiaozhen, "An Electronic Image Stabilization Algorithm Based on Efficient Block Matching on the Bitplane," *Open Journal of Applied Sciences* 3 (2013): 1-5, https://doi.org/10.4236/ojapps.2013.31B001.

9. Steven Hong, Tracey Hong, and Wu Yang, "Multi-resolution unmanned aerial vehicle video stabilization," *IEEE National Aerospace & Electronics Conference*, 2010, pp. 126-131, https://doi.org/10.1109/NAECON.2010.5712935.

10. M. Irani and P. Anandan, "About Direct Methods," In *Vision Algorithms: Theory and Practice*, IWVA 1999, Lecture Notes in Computer Science, vol. 1883 (Berlin, Heidelberg: Springer, 2000), https://doi.org/10.1007/3-540-44480-7_18.

[11] Seokhoon Kang and Chanhyuk Park, "Motion-estimation-based Stabilization of Infrared Video," Multimedia Tools and Applications 76 (2017): 1-13, https://doi.org/10.1007/s11042-017-4647-4.

[12] Sudhir Khare, Manvendra Singh, and Brajesh Kumar Kaushik, "Fast and Robust Video Stabilisation with Preserved Intentional Camera Motion and Smear Removal for Infrared Video," *IET Image Processing* 14 (2020): 376-383, https://doi.org/10.1049/iet-ipr.2019.0764.

[13] Lucio Marcenaro, Gianni Vernazza, and Carlo S. Regazzoni, "Image Stabilization Algorithms for Video-Surveillance Applications," *International Conference on Image Processing*, 2001, pp. 349-352, https://doi.org/10.1109/ICIP.2001.959025.

[14] Hugh Christopher Longuet-Higgins, "The Visual Ambiguity of a Moving Plane," Proceedings of the Royal Society of London, 1984, pp. 165-175.

[15] Project EURMARS, 2024, https://eurmars-project.eu.

[16] Project MOBILIZE, 2023, https://projekte.ffg.at/projekt/4105746.

[17] He Shen, Quan Pan, Yongmei Cheng, and Ying Yu, "Fast video Stabilization Algorithm for UAV," *IEEE International Conference on Intelligent Computing and Intelligent Systems*, 2009, https://doi.org/10.1109/ICICISYS.2009.5357609.

[18] Gideon P. Stein, Ofer Mano, and Amnon Shashua, "A Robust Method for Computing Vehicle Ego-motion," *IEEE Intelligent Vehicles Symposium*, 2000, pp. 362-368, https://doi.org/10.1109/IVS.2000.898370.

[19] R. Thillainayagi and Kumar K. Senthil, "Video Stabilization Technique for Thermal Infrared Aerial Surveillance," *Online International Conference on Green Engineering and Technologies*, 2016, pp. 1-6, https://doi.org/10.1109/GET.2016.7.

[20] Mario Miguel Valero, et al., "Thermal Infrared Video Stabilization for Aerial Monitoring of Active Wildfires," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14 (2021): 2817-2832, https://doi.org/10.1109/JSTARS.2021.3059054.

[21] Ahlem Walha, Ali Wali, and Adel M. Alimi, "Video Stabilization for Aerial Video Surveillance," *AASRI Procedia* 4 (2013): 72-77, https://doi.org/10.1016/j.aasri.2013.10.012.

[22] Yiming Wang, Qian Huang, Chuanxu Jiang, Jiwen Liu, Mingzhou Shang, and Zhuang Miao, "Video Stabilization: A Comprehensive Survey," *Neurocomputing* 516 (2023): 205-230, https://doi.org/10.1016/j.neucom.2022.10.008.

[23] Yue Wang, ZuJun Hou, Karianto Leman, and Richard Chang, "Real-Time Video Stabilization for Unmanned Aerial Vehicles," IAPR International Workshop on Machine Vision Applications, 2011.

[24] Xie Zheng, Shaohui Cui, Gang Wang, and Jinlun Li, "Video Stabilization System Based on Speeded-up Robust Features," *Proceedings of the 2015 International Industrial Informatics and Computer Engineering Conference*, March 2015.

[25] Wei Yao, S. Hinz, and Uwe Stilla, "Automatic Analysis of Traffic Scenario from Airborne Thermal Infrared Video," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 37, part B3a (2008): pp. 223-228.

[26] Adeel Yousaf, Khurram Khurshid, Jaleed Khan, and Muhammad Shehzad Hanif, "Real time video Stabilization Methods in IR Domain for UAVs — A Review," *5th IEEE International Conference on Aerospace Science and Engineering (ICASE*), Institute of Space Technology (IST), Islamabad, Pakistan, 2017, pp. 1-9.

[27] Minqi Zhou and Vijayan K. Asari, "A Fast Video Stabilization System Based on Speeded-up Robust Features," In *Advances in Visual Computing, ISVC 2011*, Lecture Notes in Computer Science, vol. 6939 (Berlin, Heidelberg: Springer, 2011), https://doi.org/10.1007/978-3-642-2403.

About the Authors

David **Schreiber** a former computer vision scientist in the Center for Digital Safety & Security of the Austrian Institute of Technology (AIT).
https://orcid.org/0000-0001-9080-1255

Andreas **Opitz** is a researcher at the Austrian Institute of Technology GmbH (AIT).

Stephan **Veigl** is a Research Engineer in the Center for Digital Safety & Security of the Austrian Institute of Technology (AIT) since 2007.
https://orcid.org/0009-0009-0909-3702