

RGLStab: A Robust Online Video Stabilization Framework Based on Grid Flow and LSTM

Yimin Zhi

Xi'an Polytechnic University

Junfeng Jing

Xi'an Polytechnic University

Meng Jiang

20200302@xpu.edu.cn

Xi'an Polytechnic University

Yongzhe Wang

Xi'an Polytechnic University

Xinyao Li

Xi'an Polytechnic University

Research Article

Keywords: Video stabilization, Motion estimation, Online process, Trajectory smoothing

Posted Date: October 14th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10924784/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

RGLStab: A Robust Online Video Stabilization Framework Based on Grid Flow and LSTM

Yimin Zhi, Junfeng Jing, Meng Jiang, Yongzhe Wang, Xinyao Li

Xi'an Polytechnic University, Xi'an, 710600, China.

Contributing authors: zhiyimin@foxmail.com; jingjunfeng0718@sina.com;
20200302@xpu.edu.cn; 18530293877@163.com; 240411060@stu.xpu.edu.cn;

Abstract

The video sequences captured by aircraft cameras are often affected by external environmental disturbances. The traditional feature point stabilization methods often encounter situations where feature tracking fails and matching becomes unstable; while the existing deep learning-based depth stabilization methods based on optical flow can handle complex jitter, they have high computational costs and are prone to introducing local distortions. To address these issues, this paper proposes an RGLStab online video stabilization framework based on grid sampling optical flow and temporal network, aiming to construct a video stabilizer that balances real-time performance and geometric fidelity. Firstly, in the motion estimation stage, a robust motion extraction mechanism combining structured optical flow sampling and a geometric RANSAC solution is designed. Secondly, in the trajectory smoothing stage, a full-causal temporal model is constructed using a unidirectional LSTM network to effectively separate and smooth reconstruct the high-frequency tremors in the temporal trajectory. Finally, in the frame synthesis stage, the video frame is reconstructed using global four-degree-of-freedom similarity transformation, and the non-rigid deformation artifacts are suppressed by the underlying spatial constraints. The experimental results in the NUS public dataset show that this framework outperforms the existing online stabilization algorithms in both qualitative and quantitative evaluation metrics, and the stabilization quality is comparable to some offline methods. This mechanism effectively improves feature failure and dynamic foreground interference problems under complex maneuvers such as quick rotation, achieving an online inference frame rate of 24.51 FPS while maintaining good robustness.

Keywords: Video stabilization, Motion estimation, Online process, Trajectory smoothing

1 Introduction

In recent years, the advancement of intelligent flight systems has led to the widespread deployment of airborne camera systems—serving as core visual sensors—in mission-critical applications such as aerial patrol, disaster reconnaissance, and real-time dynamic target tracking [1], all of which demand low-latency onboard processing.

However, factors such as complex airflow disturbances, mechanical vibrations of the rotor, and rapid drone maneuvers inevitably cause image jitter in video streams captured by airborne cameras. This nonlinear visual oscillation not only compromises operator observation experience, but also degrades performance of downstream computer vision tasks. Consequently, developing video stabilization algorithms that balance real-time performance with geometric fidelity has become a

critical requirement in the fields of aerial robotics and intelligent vision systems[2].

Most existing video stabilization technologies adopt an offline processing mode[3–7], leveraging either global motion estimation or long-range bidirectional sliding windows to model and smooth camera trajectories. However, these non-causal mechanisms that rely on future information will cause certain processing delays, making it difficult to meet the real-time application requirements. The online stabilization system needs to have low computing time consumption and follow an immediate processing pipeline. Although several existing real-time methods incorporate a limited number of future preview frames to enhance trajectory smoothness[8, 9], this strategy still involves a non-negligible buffer delay. This paper focuses on online real-time image stabilization. Based on the requirement of low latency, this study explores and uses the temporal movement patterns of historical frames. Without causal constraints relying on future frames, it relies on past and current observation data to complete posture prediction, aiming to achieve online image stabilization with no buffering delay.

During the motion estimation phase, traditional online image stabilization algorithms face significant challenges when dealing with complex maneuvers such as rapid rotation and translation commonly encountered in airborne platform[10]. These methods predominantly rely on sparse, handcrafted local features for inter-frame correspondence. Although there have been emerging feature point detection algorithms based on deep learning in recent years[11], which to some extent alleviate the challenges of feature extraction under complex lighting and weak texture conditions, both are essentially limited by the sparse point matching paradigm. This paradigm relies on the significant texture structures and corner points in the image. When the airborne camera encounters rapid rotation, translation, or dynamic blurring, it is prone to a decrease in the number of effective feature points and unstable matching correlations. This characteristic of discrete feature extraction makes it difficult for these algorithms to provide a continuous and robust global motion trajectory estimation in complex flight environments, which may lead to trajectory noise or compensation deviations.

In recent years, deep learning-based optical flow estimation and grid deformation stabilization methods have emerged[12–17], using dense pixel-level motion perception to enhance the resilience of systems against nonlinear jitter. However, existing deep stabilization solutions still face two major engineering challenges in airborne deployment. Firstly, high-precision dense optical flow networks incur substantial computational overhead, with their high-dimensional tensor iterations increasing inference latency and compromising real-time performance. Secondly, to eliminate depth parallax, current deep grid stabilization approaches tend to apply localized nonlinear deformations to video frames. Such non-rigid processing tends to alter the original rigid optical structures within frames, introducing local geometric distortions that compromise visual fidelity.

To address the aforementioned issues, this paper proposes an online video stabilization framework named RGLStab, which is based on grid sampling optical flow and temporal networks. In the motion estimation stage, this framework introduces a lightweight extraction mechanism that combines grid sampling of the flow field extracted by RAFT with the geometric RANSAC solution. This mechanism extracts a stable global four-degree-of-freedom similarity transformation matrix, leveraging the high-precision advantage of RAFT’s pixel-level motion field while improving the bottleneck of slow inference in traditional dense optical flow algorithms. In the trajectory smoothing stage, this framework builds a fully causal temporal model based on unidirectional LSTM. This model implicitly learns the kinematic laws of the physical stabilizer and can complete the smooth posture calculation based solely on historical memory and instantaneous observations, achieving online prediction without future frame dependence in a strict sense.

The main points of work in this article are threefold as follows.

1. This study proposes an online video stabilization framework RGLStab that integrates structured grid-based optical flow motion representation with long short term memory networks (LSTM). The framework achieves enhanced computational efficiency while maintaining robust video stabilization performance.

2. A motion extraction strategy based on the joint physical screening of the RAFT dense flow

field grid sampling and the RANSAC algorithm was designed. The high-dimensional and complex pixel-level relative motion was uniformly reduced and solved as a global four-degree-of-freedom similarity transformation matrix. This strategy strictly maintained the spatial geometric consistency of the global scene at the bottom-level constraints, effectively suppressing the local distortions and non-rigid deformations that are prone to occur under complex camera movements.

3. A complete causal time-series smoothing model based on unidirectional LSTM and a decoupling optimization strategy were constructed. This model can complete attitude inference solely based on the historical memory sequence and the current observation state, achieving low-latency trajectory prediction without future frame dependence. Moreover, this paper designs a decoupled weight-based mixed loss function. By imposing high-weight penalties on the rotation components, it effectively eliminates high-frequency mechanical tremors while maximizing the retention of the natural geometric proportion and spatio-temporal consistency of the stabilized image.

2 Related Works

We briefly introduce traditional stabilization methods, deep learning stabilization methods, and online stabilization methods as follows:

2.1 Traditional Approaches

1) 2D methods simplify camera motion into geometric transformations on the image plane. Early 2D motion estimation approaches relied on the tracking of image features in multiple frames, achieving video stabilization through smoothing of the feature trajectory [6, 18]. However, large-scale camera movements often make it difficult to extract long feature trajectories. Subsequent research used motion models between adjacent frames to achieve feature matching. These methods depend on manually designed feature extractors for keypoint tracking[19, 20]. However, manual features often fail under sparse textures, severe lighting variations, or dynamic blurring caused by rapid movements. The loss of feature points and incorrect matching can lead to a decrease in the accuracy of global motion estimation, making

it difficult to cope with complex environmental disturbances.

2) In 3D modeling research[21–23], researchers utilize structure from motion (SfM) technology[24] to estimate camera paths and scene point clouds in three-dimensional space. Additional hardware such as depth cameras[25], gyros [8, 9], or light field cameras[26] can also be employed. Although excelling in handling parallax effects in complex scenes, 3D reconstruction remains computationally intensive and prone to system crashes during large-scale dynamic occlusions or aggressive maneuvers involving airborne equipment.

2.2 Deep Learning Methods

To overcome the limitations of traditional methods, deep learning techniques have been widely adopted in stabilization research in recent years. Some studies focus on building end-to-end generative networks that directly learn frame mapping relationships through large-scale stable/jittered video datasets, such as the DeepStab dataset[12]. Xu et al.[27] utilized adversarial frameworks to synthesize target images as a means of guiding deformation, while Yu et al.[15] focused on refining initial optical flow estimates into precise pixel-wise motion compensation fields. However, due to the DeepStab dataset that contains only 60 video segments, deep learning methods face challenges in generalization capability. Yu et al. adopted convolutional neural networks as optimizers instead of data learning to avoid overfitting on individual input samples[28].

Recent approaches have emphasized replacing traditional motion estimation modules with deep neural networks, such as deep homography transformations[29, 30] and deep grid flow methods[17], by introducing deep homography networks or grid optical flows for dense motion field extraction. However, existing mainstream deep grid stabilization schemes often apply local spatial nonlinear stretching to compensate for disparity. To address the local distortions caused by non-rigid deformation and the computational challenges posed by high-dimensional tensors, this paper proposes a stabilization strategy based on dense flow field sampling. We first use RAFT[31] to obtain pixel-level motion fields without texture constraints. Then, we introduce grid sampling and the RANSAC algorithm[32] to extract the global

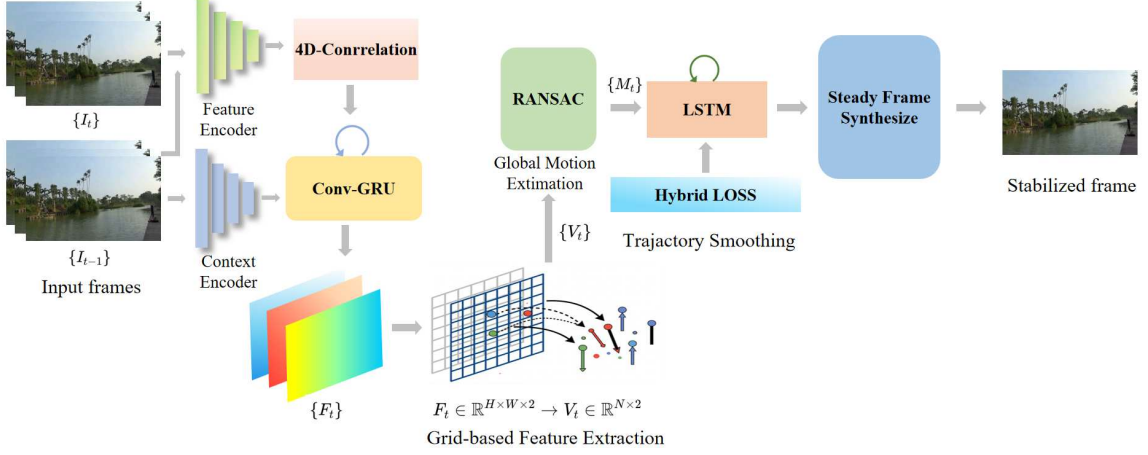


Fig. 1: The overall process of the proposed RGLStab framework consists of three core modules: the motion estimation module, the trajectory smoothing module, and the frame synthesis module. The motion estimation module employs the global RANSAC estimation strategy, which is implemented after sampling the optical flow grid extracted by RAFT. The trajectory smoothing module adopts the causal method of unidirectional LSTM and combines the decoupled supervision mechanism. The frame synthesis module utilizes the high-fidelity reconstruction strategy based on spatial unified transformation.

rigid transformation matrix. This design aims to balance the accuracy and efficiency of motion estimation, it utilizes the fine motion modeling capability of RAFT while overcoming the local texture distortion in grid deformation through rigid constraints and significantly improves the inference speed of the algorithm.

2.3 Online Approaches

Most existing video stabilization techniques adopt an offline processing mode, where global path optimization is carried out by obtaining a complete video sequence. However, in scenarios that require immediate visual feedback, the low latency performance and computational efficiency of the system are of great significance. In academic research, real-time methods typically refer to the processing frame rate of the algorithm being able to match the sampling speed of the hardware; while online methods emphasize the adherence to causality of the model, that is, performing stabilization without obtaining future frames. These online models[2, 10, 12] calculate the instantaneous posture based solely on the current observed data and historical trajectories, thereby effectively alleviating the buffering delay caused by relying on pre-read frames.

In the context of online stabilization, Liu et al.[11] introduced an important real-time framework that integrates the SuperPoint deep learning model for feature detection. This approach aims to enhance the robustness of feature extraction in complex environments, providing a flexible and parameter-controllable solution for low latency stabilization. However, the effectiveness of such methods may be limited by the inherent characteristics of sparse feature matching. In scenarios involving rapid rotation or significant zooming, sparse feature matching often becomes a bottleneck, potentially leading to tracking inconsistencies or system failures.

In this study, we used the proposed RGLStab online video stabilization framework, utilizing grid sampling for dense optical flow and LSTM to address these challenges.

3 Proposed Method

Our framework consists of three core modules: motion estimation, trajectory smoothing, and frame synthesis.

3.1 Motion Estimation

The motion estimation module is an important component of the framework. Its function

is to extract accurate and highly reliable global camera motion from consecutive video frames. Traditional optical flow estimation algorithms, such as the classic Lucas-Kanade algorithm[33] or Horn-Schunck algorithm[34], usually rely on the assumptions of constant brightness and spatial smoothness. However, when dealing with rapid camera movements, lighting variations, or dynamic blurring caused by airborne cameras, these methods often suffer from tracking failures and estimation errors due to invalid prior assumptions. To overcome the limitations of conventional assumptions, deep learning-based optical flow networks such as FlowNet[35] and PWC-Net[36], have significantly improved large-scale displacement perception through data-driven approaches. Nevertheless, early deep networks employing coarse-to-fine cascaded architectures still face challenges in extracting fine details and boundary motion when handling complex nonlinear high-frequency vibrations from drones. To enhance the robustness of motion estimation, this module introduces the dense optical flow architecture RAFT [31]. RAFT constructs high-dimensional all-pairs correlation volumes by calculating dot products between all pixel pairs across full images of consecutive frames, followed by multi-round optimization using gated recurrent units (GRUs) at fixed high resolutions. This dense iterative mechanism enables the system to output stable, high-quality 2D motion fields even under conditions of extensive local texture degradation or rapid rotation.

After obtaining the high-dimensional dense flow field $F_t \in \mathbb{R}^{H \times W \times 2}$ from RAFT output, directly applying pixel-level high-dimensional tensors to subsequent attitude computation would incur substantial computational and memory overhead. To address this, our module employs a low-time-complexity grid sampling dimensionality reduction strategy. With a spatial step size N for the sampling grid, the mathematical definition of the reduced sparse motion set can be expressed as follows by extracting feature motion vectors exclusively from grid vertices:

$$V_t = \{F_t(i, j) \mid i = k_h N, j = k_w N\} \quad (1)$$

Here, k_h and k_w denote the spatial indices of the grid in the height and width directions, respectively. This method extracts motion representations with excellent global consistency, and it does not require complex filtering processing. At the same time, it reduces the computational demands in the subsequent reasoning process.

To estimate the global motion from V_t , we apply RANSAC [32] to filter out dynamic outliers. This process yields a 2D similarity transformation matrix M_t for the current frame, constrained to four degrees of freedom:

$$M_t = \begin{bmatrix} s \cos \theta & -s \sin \theta & t_x \\ s \sin \theta & s \cos \theta & t_y \end{bmatrix} \quad (2)$$

Here, θ , (t_x, t_y) , and s represent the rotation angle, the translation vector, and the global scaling factor, respectively. By implementing this mapping, we can avoid independent local grid deformation and disproportionate stretching. Figure 2(a) shows the case of local distortion, and Figure 2(b) shows the case of high geometric fidelity. Therefore, our spatial transformation mechanism can suppress local distortion and deformation, and provide a stable reference for subsequent temporal smoothing.



Fig. 2: Comparison of images with and without local distortion. (a) Local distortion effect (b) Global geometric high-fidelity.

Unlike the use of complex high-degree deformation networks or full affine models, this framework tends to confine spatial transformations within four degrees of freedom (translation, rotation, and uniform scaling). The retention of scaling parameters aims to better adapt to the dynamic changes in the global field of view caused by the movement of the drone along the Z-axis.

At the same time, the introduction of similarity priors helps prevent the system from overfitting to dynamic scenes or high-frequency airflow vibrations, effectively avoiding the generation of shear deformation and unnatural stretching at the geometric level. Moreover, when dealing with scenes with significant depth, we avoid overcompensating for 3D disparity using 2D grids. The system mainly focuses on filtering out the global high-frequency camera jitter and has retained the scene depth relative motion that conforms to the real optical laws as much as possible. Ultimately, this strategy of moderately simplifying the transformation parameters provides a guaranty for achieving microsecond-level inference latency and high geometric fidelity, helping to enhance the system's robustness in complex flight environments.

3.2 Trajectory Smoothing

As the temporal prediction core, the trajectory smoothing module maps shaky raw motions into ideal paths under a strict zero-look-ahead constraint. Once the front-end outputs the 4-DOF similarity matrix M_t , we accumulate these matrices to determine the absolute pose of the camera in the global frame. This yields a vector of physical state that contains the parameters of translation (t_x, t_y) , rotation (θ) , and scaling (s) .

Unlike offline smoothing algorithms that rely on sliding windows or look-ahead frames, such approaches often introduce significant computational delays, making them inadequate for real-time requirements. To optimize response speed, this framework develops a fully causal temporal inference model based on unidirectional LSTM[37]. The model adheres to strict unidirectionality along the physical time axis, where at any given moment t , the network only receives observations of instantaneous physical states, along with hidden states and cell states that encapsulate historical motion patterns. Its internal inference process can be summarized as follows.

First, we construct a nonlinear time-series filter based on gate mechanisms. The network collaboratively evaluates the physical correlation between current observation vectors and historical trajectories through a forgetting gate f_t and an input gate i_t . Let the physical observation state of four-degrees of-freedom at the current moment be denoted as $x_t = [t_x, t_y, \theta, s]^T$, with its derivation

equation as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \quad (3)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \quad (4)$$

σ denotes the Sigmoid activation function, W and b represent the weight matrix and bias term of the corresponding gate, while h_{t-1} indicates the hidden state from the previous time step. This mechanism dynamically balances the influence of historical motion inertia with current instantaneous features, enabling effective extraction of stable image paths even during sudden high-frequency mechanical vibrations.

Subsequently, adaptive updates are performed to execute global trajectory memory. After extracting the candidate physical states $\tilde{C}_t$ at the current time, the system dynamically updates the cell states C_t under the regulation of the gating mechanism:

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, X_t] + b_c) \quad (5)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (6)$$

This process aims to mathematically extract high-frequency noise components from observation sequences, thereby achieving smooth processing of complex maneuver trajectories.

Finally, the output is implemented to achieve a smooth pose. Using the output gate mapping function o_t , the network computes the final hidden state h_t at the current time instant:

$$o_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o) \quad (7)$$

$$h_t = o_t \odot \tanh(C_t) \quad (8)$$

Throughout the forward propagation equation system, the model operates independently of any $t + 1$ or subsequent future observations, ensuring zero-latency performance at the logical infrastructure level. The updated hidden state h_t directly outputs the predicted smooth attitude values through fully connected layers. This four-degree-of-freedom physical state-based inference paradigm enables LSTM to simultaneously internalize complex dynamic patterns generated by camera plane movement, yaw rotation, and depth push-pull motions. By alleviating computational demands on onboard platforms, this architecture effectively meets real-time video stream processing requirements for low-latency performance.

3.3 Decoupled Hybrid Loss

To guide LSTM networks to spontaneously internalize the spatiotemporal consistency patterns of real-world physical stabilizers, this study proposes a decoupled hybrid loss function for geometric distortion perception. In traditional deep learning stabilization training, directly approximating predicted trajectories to real smooth trajectories (Ground Truth) often overlooks the differential impact of image translation, rotation, and scaling errors on physical geometry. In airborne camera imaging geometry, minor rotation prediction deviations can create pronounced leverage effects at image edges, which is a primary cause of significant black borders and loss of effective field of view (FOV) in final images.

Therefore, this system eliminates explicit smoothing constraints and instead implements weight decoupling for translation, rotation, and scaling components in the loss function under supervision with high-quality stabilization data. Let the ideal smooth reference pose be Y_{gt} and the predicted pose be Y_{pred} . For distance metric selection, this framework adopts the L1 norm[38] instead of the L2 norm[39]. Sudden mechanical impacts often cause gradient instability. To mitigate this, our loss function utilizes the L1 norm rather than L2 to provide steady linear gradients. Based on the decoupled 4-DOF motion state, the translation and rotation loss components are computed independently:

$$L_{trans} = \|[t_{x,pred}, t_{y,pred}] - [t_{x,gt}, t_{y,gt}]\|_1 \quad (9)$$

$$L_{rot} = \|\theta_{pred} - \theta_{gt}\|_1 \quad (10)$$

$$L_{scale} = \|s_{pred} - s_{gt}\|_1 \quad (11)$$

To supervise the trajectory smoothing, we formulate the total loss as a weighted sum of the decoupled components:

$$L_{total} = \lambda_{trans}L_{trans} + \lambda_{rot}L_{rot} + \lambda_{scale}L_{scale} \quad (12)$$

Rather than weighting all degrees of freedom equally, we assign specific penalty weights based on airborne camera dynamics. The translation weight is set to $\lambda_{trans} = 1.0$, which acts as an anchor for the global trajectory and prevents motion lag caused by over-smoothing. Because rotational jitter significantly reduces the valid field of view (FOV), we assign a higher penalty of

$\lambda_{rot} = 2.0$ to the rotational component. This forces the LSTM network to be highly sensitive to rotation errors. Finally, the scaling weight is set to $\lambda_{scale} = 1.0$ to prevent the network from over-compensating through unnatural zoom artifacts. This decoupling mechanism—combining translational anchoring, high-risk rotation penalties, and scaling constraints—allows the network to achieve optimal temporal smoothing while effectively suppressing artifacts like black edges and distortions at the algorithmic level, achieving a balanced compromise between stabilization smoothness and geometric fidelity.

3.4 frame synthesis

The frame synthesis module serves as the final execution unit in the stabilization pipeline, with its core function being to translate the ideal temporal pose derived from the smoothing module into pixel-level spatial resampling operations, thereby generating visually stable video streams. Existing depth grid stabilization approaches generally treat frame synthesis as a local deformation process. These methods rely on spatially variable nonlinear warping techniques, which attempt to mitigate local parallax or jitter by applying independent displacement vectors to different grid vertices. However, this rendering mechanism, which models images as non-rigid deformable planes faces inherent physical limitations: weakly coupled nonlinear stretching between grids may disrupt the original optical geometry within frames. When drones encounter high-frequency mechanical maneuvers or complex airflow disturbances, such mechanisms are prone to causing localized distortions and the Jello Effect.

To suppress this phenomenon at the physical layer level, the frame synthesis module of this framework implements a global rigid constraint called Spatially-invariant Transformation. Under this mechanism, all pixels undergo uniform displacement through a unified set of linear algebraic mapping rules, thereby preserving the relative geometric relationships between every pixel in the entire image. Specifically, the system first calculates the required relative motion compensation matrix for the current frame by comparing the optimal smoothed pose prediction output of the LSTM with the original jitter pose. The original cumulative pose resolved at time t is denoted as

$P_{t,raw} = [t_{x,r}, t_{y,r}, \theta_r, s_r]^T$, The smoothed cumulative pose predicted by LSTM is $P_{t,smooth} = [t_{x,s}, t_{y,s}, \theta_s, s_s]^T$. To perform matrix operations, first construct it as the corresponding homogeneous transformation matrix:

$$M_{t,raw} = \begin{bmatrix} s_r \cos \theta_r & -s_r \sin \theta_r & t_{x,r} \\ s_r \sin \theta_r & s_r \cos \theta_r & t_{y,r} \\ 0 & 0 & 1 \end{bmatrix} \quad (13)$$

$$M_{t,smooth} = \begin{bmatrix} s_s \cos \theta_s & -s_s \sin \theta_s & t_{x,s} \\ s_s \sin \theta_s & s_s \cos \theta_s & t_{y,s} \\ 0 & 0 & 1 \end{bmatrix} \quad (14)$$

Subsequently, by performing inverse matrix calculations and matrix multiplication operations on homogeneous matrices, the relative motion compensation matrix $M_{t,comp}$ required to map the original jitter attitude to the ideal smooth attitude was precisely computed:

$$M_{t,comp} = M_{t,smooth} \cdot M_{t,raw}^{-1} \quad (15)$$

Finally, the module employs the four-degree-of-freedom compensation matrix that incorporates translation, rotation, and proportional scaling as the sole global driving force to perform inverse resampling on the entire frame. This reconstruction strategy treats the image as an incompressible rigid body, which not only demonstrates lower computational complexity and rendering latency compared to time-consuming grid-based nonlinear interpolation, but also provides the system with inherent immunity against local geometric distortions at the physical level.

To maintain an effective field of view (FOV) and avoid drastic dynamic cropping, this smoothing module is tightly coupled with our decoupled loss function. Existing algorithms often fail to adequately suppress rotation and scaling distortions, forcing them to significantly crop the image center to mask edge artifacts. Due to the high weight penalty we impose on rotation and scaling errors during training, the predicted poses by the network are physically constrained. In the inverse compensation stage, this design naturally prevents excessive image tilt and breathing effects. Therefore, we only need to perform a small amount of boundary truncation to remove invalid edge pixels, thereby generating a smooth and geometrically precise video with the maximum field of view.

4 Experimental Results

4.1 Experimental setup and implementation details

Many existing deep stabilization methods rely on end-to-end image-level training. This approach is not only computationally expensive, but also leaves trajectory optimization vulnerable to non-motion factors, such as lighting changes or texture interference. Rather than processing raw image sequences end-to-end, we decouple the kinematics from the spatial features through dimensionality reduction. By doing so, we force the network to focus exclusively on physical motion, drastically reducing computational overhead. For model training, we utilize the Motion-Stab dataset[40] due to its extensive collection of complex camera trajectories and high-frequency vibrations. During initial model training, the system processes stable-shake video frames from the dataset into front-end perception modules. Using RAFT dense optical flow sampling and RANSAC algorithms, the system transforms raw high-dimensional pixel streams into low-dimensional four-degree-of-freedom physical state sequences, including shake observation vectors X_t containing translation, rotation, and scaling, and ideal reference trajectories Y_t^{gt} . This preliminary feature extraction process effectively filters redundant image surface features, providing cleaner physical supervision signals for subsequent temporal networks.

After obtaining low-dimensional kinematic datasets, the system eliminates computational overhead for real-time image rendering and optical flow computation during training. Experiments were conducted on two NVIDIA RTX 2080Ti GPUs using the PyTorch framework, where lightweight temporal inference training was performed independently on the backend unidirectional LSTM network. To maintain network generalization capability during trajectory adjustment, the AdamW optimizer was used with an initial learning rate of 1×10^{-3} and weight decay set to 1×10^{-4} . This regularization measure can prevent the LSTM from over-adapting to specific high-frequency mechanical noise, thereby enhancing its robustness. In terms of model optimization, we employed the cosine annealing learning rate scheduler. This enabled the network to converge

rapidly in the early stages and smoothly converge to a stable global minimum as the training approached its end.

4.2 Metrics

We quantitatively evaluated our method using three standard metrics from Liu et al.[3]: Distortion, Stability, and Cropping ratio.

The Distortion score measures geometric fidelity by assessing the anisotropic scaling between the input and stabilized frames. It is calculated as the minimum-to-maximum eigenvalue ratio of the affine component of the inter-frame homography. We report the minimum ratio across the entire video sequence, where a value approaching 1 indicates well-preserved rigid structures and minimal artificial deformation.

To evaluate temporal smoothness, we compute the Stability score. Following[3], we extract the translation and rotation trajectories and analyze them in the frequency domain using FFT. The score is defined as the minimum ratio of low-frequency energy to total energy across all motion components. Higher values (closer to 1) confirm that high-frequency mechanical jitters have been effectively filtered out.

The cropping ratio is used to objectively quantify the proportion of effective physical field of view (FOV) that can be retained after stable image motion compensation. Since reverse camera compensation inevitably causes irregular blank areas at the image borders, this metric calculates the maximum area retention ratio to capture a rectangular image without black-edges by performing an inverse projection on the corner points of the borders.

4.3 Quantitative Results

To objectively evaluate the comprehensive stabilization performance of this framework, we conducted quantitative comparisons with multiple representative stabilization algorithms on the NUS Stabilization Benchmark Dataset[3]. The comparison table includes traditional classical methods such as Bundled[3], Meshflow[10], as well as deep learning-based solutions such as Deep3D[21], DUT[41], Yu et al.[15], FuSta[42], StabNet[12], PWStableNet[13], SuperPoint[11]

and NNDVS[40]. In defining the temporal processing architecture, Meshflow[10], StabNet[12], SuperPoint[11], NNDVS[40], and our proposed method strictly adhere to the online processing logic without future frame dependencies, while other baseline methods belong to the offline approach category. All experiments utilized publicly available source code and pre-trained models of benchmark methods to ensure fairness in comparisons. Quantitative evaluation was divided into five typical complex scenarios: Regular, Fast Rotation, Intense Zooming, Crowd, and Parallax. The overall quantitative performance of key metrics is presented in Table 1.

Table 1 summarizes the quantitative evaluation results of various methods in five typical categories in the NUS dataset and at the global average level. Evaluation data demonstrates that offline methods generally exhibit superior quantification metrics in complex scenarios, thanks to auxiliary global video sequences or future frame information. Since the online method cannot obtain the information of subsequent frames, there is naturally a difficulty in matching the smooth effect of the offline algorithm. However, our framework has narrowed this performance gap. For distortion metrics, our method outperforms the classic Meshflow and the recent deep learning-based methods. NNDVS has a decrease in geometric fidelity, although it uses a deep network, its mesh deformation model has a phenomenon that leads to local non-rigid artifacts. We avoid this problem by enforcing a global 4-degree-of-freedom similarity constraint (via RANSAC) in the motion estimation stage. This constraint locks the ratio of the eigenvalues of the transformation within a safe and high-fidelity range. By strictly limiting the spatial mapping, we directly prevent unnatural pixel stretching, making robust geometric integrity a fundamental physical characteristic of our pipeline.

A primary goal of our method is to optimally balance the Stability Score (SS) with the Field of View (FOV) retention ratio. FFT analysis of the trajectory sequences shows that our SS rivals that of offline algorithms, proving that the unidirectional LSTM effectively filters high-frequency mechanical vibrations without needing future-frame buffers. However, temporal smoothness often comes at a cost. Traditional approaches tend to overcompensate for rotational motion,

Table 1: Quantitative performance comparison of various stabilization methods. Values represent averages across each category for cropping ratio, distortion value, and stability score. Higher scores denote superior performance across all metrics. Top-tier results are highlighted with bold text.

Metrics	Type	Methods	Regular	Rotation	Zooming	Crowd	Parallax	Average
Cropping	Offline	Bundled	0.6658	0.6477	0.5773	0.6685	0.7158	0.6550
		Yu et al.	0.9486	0.7661	0.8971	0.9081	0.9209	0.8882
		Deep3D	0.8321	0.7085	0.7634	0.7700	0.7553	0.7659
		PWStableNet	0.9319	0.8801	0.9303	0.9181	0.9244	0.9170
		DUT	0.8485	0.7609	0.7986	0.7986	0.9254	0.8264
		Fusta	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	Online	Meshflow	0.8081	0.7578	0.7773	0.7888	0.8121	0.7888
		StabNet	0.7491	0.7417	0.7376	0.7247	0.6967	0.7300
		SuperPoint	0.9227	0.7681	0.8901	0.8932	0.9063	0.8761
		NNDVS	0.9433	0.9021	0.9185	0.9247	0.9326	0.9242
		Ours	0.9645	0.9419	0.9551	0.9255	0.9247	0.9423
Distortion	Offline	Bundled	0.9409	0.8692	0.9073	0.8910	0.8965	0.9010
		Yu et al.	0.9752	0.7616	0.8630	0.9191	0.9219	0.8882
		Deep3D	0.9663	0.9688	0.9762	0.9594	0.9793	0.9700
		PWStableNet	0.9852	0.9727	0.9792	0.9790	0.9852	0.9803
		DUT	0.9600	0.8940	0.8606	0.8617	0.9274	0.9007
		Fusta	0.9791	0.9194	0.9829	0.9791	0.9803	0.9682
	Online	Meshflow	0.9168	0.7679	0.8715	0.8246	0.8429	0.8447
		StabNet	0.8393	0.7205	0.7807	0.7725	0.7660	0.7758
		SuperPoint	0.9827	0.9617	0.9682	0.9586	0.9690	0.9680
		NNDVS	0.9417	0.9347	0.9536	0.9716	0.9727	0.9549
		Ours	0.9781	0.9855	0.9863	0.9732	0.9707	0.9788
Stability	Offline	Bundled	0.9048	0.9194	0.9236	0.8744	0.8964	0.9037
		Yu et al.	0.8427	0.9017	0.8907	0.8660	0.8896	0.8781
		Deep3D	0.8491	0.9123	0.9026	0.8704	0.8805	0.8830
		PWStableNet	0.8162	0.9251	0.8906	0.8162	0.8549	0.8606
		DUT	0.9217	0.8703	0.9221	0.9042	0.8827	0.9002
		Fusta	0.8973	0.8774	0.9027	0.8670	0.8527	0.8794
	Online	Meshflow	0.8386	0.9092	0.8839	0.8280	0.8488	0.8617
		StabNet	0.8406	0.8384	0.8697	0.7912	0.8289	0.8338
		SuperPoint	0.8962	0.8421	0.9051	0.9129	0.8445	0.8802
		NNDVS	0.8030	0.9341	0.8982	0.8408	0.8762	0.8705
		Ours	0.8415	0.9198	0.8807	0.8924	0.8498	0.8768

forcing them to aggressively crop the image center and severely degrade the FOV. We avoid this trap through our decoupled hybrid loss. By explicitly applying a high penalty to rotation errors ($\lambda_{rot} = 2.0$), we physically constrain the network from making wild inverse rotational compensations. As a result, compared to baselines like Meshflow, our framework retains a significantly larger usable

FOV while maintaining excellent high-frequency jitter suppression.

In practical applications, our framework achieves the optimal balance between improving accuracy and reducing computational costs. Although current high-precision methods (such as DUT[41]) can achieve high stability, they come at the cost of complex network architectures and

slow multi-round iterative compensation. Similarly, methods like Deep3D[21] can handle complex spatial operations, but they rely on offline pre-computed depth prior information. By avoiding these burdensome dependencies, our method offers a lightweight and reliable stabilization solution. Compared to these computationally intensive or heavily offline-dependent solutions, our system maintains highly competitive core stabilization quantification metrics without introducing physical 3D sensors and under strict single-pass forward propagation constraints. Crucially, through architectural synergy between front-end RAFT dense optical flow processing and back-end unidirectional LSTM inference, the system achieves significant optimization in single-frame inference latency and computational resource consumption compared to complex rendering networks. This strategy of achieving ultra-low latency and lightweight deployment through moderate model parameter simplification fully validates the framework’s logical consistency between theoretical algorithm design and engineering physical constraints, providing a highly practical solution paradigm for real-time stabilization in computing-limited drones and portable mobile platforms.

To evaluate the computational performance of various methods in practical engineering deployment scenarios, we conducted a systematic comparative analysis of inference frame rates (FPS) in multiple stabilization algorithms. It should be objectively noted that the time-series data in this system were measured on NVIDIA RTX 2080Ti platforms. Table 2 demonstrates that in offline method arrays, algorithms typically allow access to global video sequences or incorporate future frame data to achieve smoother trajectory optimization, though this often comes at the cost of reduced processing speed and increased buffer latency. The test results indicate that FuSta, constrained by its complex global spatiotemporal optimization mechanism, achieves only 0.047 FPS runtime efficiency, while Deep3D demonstrates only 0.32 FPS during complex depth calculations and 3D reconstruction tasks. Additionally, the bundle method and the DUT method achieve runtime efficiencies of approximately 3.5 FPS and 3.87 FPS, respectively. Although PWStableNet exhibits relatively favorable inference speed (17.76 FPS) in offline arrays, the inherent causal latency

inherent to offline architectures remains a significant engineering limitation when addressing real-time stream processing and obstacle avoidance challenges in airborne platforms such as drones.

Table 2: Comparison of inference frame rate among different methods.

Type	Methods	FPS
Offline	Bundled	3.5
	Deep3D	0.32
	DUT	3.87
	PWStableNet	17.76
	Fusta	0.047
Online	Meshflow	0.30
	StabNet	1.08
	SuperPoint	30.67
	NNDVS	14.63
	Ours	24.51

In online method arrays dedicated to low-latency processing, the operational efficiency of various solutions demonstrates significant hierarchical differences. The traditional Meshflow method achieves only 0.30 FPS during mesh fragmentation partitioning and local deformation processing, while the early deep learning approach StabNet tests at approximately 1.08 FPS. In contrast, the NNDVS method achieves notable efficiency optimization, reaching 14.63 FPS. Notably, although the SuperPoint method achieves around 30.67 FPS processing speed through ultra-lightweight sparse feature point matching, our system introduces a computationally intensive RAFT dense optical flow architecture at the front-end to ensure perceptual robustness under extreme conditions. Through backend grid sampling dimensionality reduction and unidirectional LSTM inference, it still achieves an online processing frame rate of 24.51 FPS. By balancing four-degree-of-freedom spatial constraints with higher geometric fidelity, this framework effectively mitigates complex image-level rendering overheads, establishing a more practically valuable online engineering trade-off between computational speed and stabilization quality.

4.4 Qualitative Analysis

Figure 3 provides a qualitative visual comparison of our framework with the main online methods (StabNet, SuperPoint, and NNDVS). From the second and third rows showing complex camera movements, it can be observed that StabNet and SuperPoint overcompensate for rotations when smoothing high-frequency trajectories. Therefore, these networks produce obvious dynamic black borders to force spatial displacements, as shown in the red box.

To generate clean video streams, these methods require aggressive center cropping during post-processing, leading to significant loss of effective field of view (FOV). While NNDVS employing depth grid deformation models improves global FOV preservation, its spatially-variant rendering mechanism tends to disrupt inherent rigid optical structures when processing complex depth or dynamic foreground elements. As indicated by the red arrow in the third row, NNDVS reconstructed images exhibit unnatural distortions and geometric aberrations at the edges of the architectural column and the textures of the ground.

In contrast, the stabilized image frames output by this system demonstrate superior fidelity in both spatial structure and field of view breadth. Benefiting from global rigid constraints imposed by the front-end four-degree-of-freedom similarity transformation and high-weight penalties applied to rotation components during model optimization, our framework effectively suppresses local nonlinear deformations and excessive global rotation during inverse motion compensation. Analysis of output results reveals that horizontal reference objects and vertical architectural structures (such as metal pillars on the right side of the third row) maintain accurate orthogonal perspective relationships, with the algorithm being inherently resistant to local artifacts jello effect. Simultaneously, the system effectively avoids exaggerated black edge distortion at image edges, achieving smooth visual experience while preserving the original video’s effective physical field of view. This validates the framework’s optimal engineering trade-off between geometric fidelity and temporal stability.

4.5 Research on the Importance of LSTM

After validating the spatial architecture’s effectiveness, we further evaluated the nonlinear filtering capabilities of the temporal inference core. By replacing unidirectional LSTM modules with traditional Kalman filters[43], we compared the inference performance of purely linear temporal models in airborne environments. Figure 4 illustrates the comparison of smoothing results between the two approaches. Constrained by the linear Gaussian assumption, Kalman filters struggle to balance smoothing smoothness and response latency when handling high-frequency shocks and large-scale nonlinear maneuvers: increasing smoothing confidence leads to motion lag, while prioritizing rapid tracking results in limited high-frequency jitter filtering, causing stability scores to decline. In contrast, the LSTM modules in our framework leverage nonlinear gating mechanisms to function as adaptive soft temporal filters. They effectively approximate target trajectories and isolate transient disturbances without future frame delays, demonstrating the advantages of deep nonlinear networks in complex physical simulations.

4.6 Research on decoupled hybrid loss

To validate the effectiveness of decoupled hybrid loss functions in practical engineering deployment, we conducted visual qualitative ablation analysis of the core rotation penalty weight. Figure 5 intuitively demonstrates the stable image rendering results of the system when handling complex drone maneuvers under different weight configurations.

In the first control group, we employed a traditional equal-weight constraint strategy, setting $\lambda_{trans} = 1.0, \lambda_{rot} = 1.0, \lambda_{scale} = 1.0$. As shown in the upper row of Figure 5, when spatial degrees of freedom were penalized proportionally, the LSTM network tended to produce significant reverse rotation compensation during smooth trajectory adjustment. This extensive geometric extrapolation triggered pronounced leverage effects at image boundaries, resulting in substantial black invalid regions after secondary rendering truncation and causing a certain degree of FOV reduction.

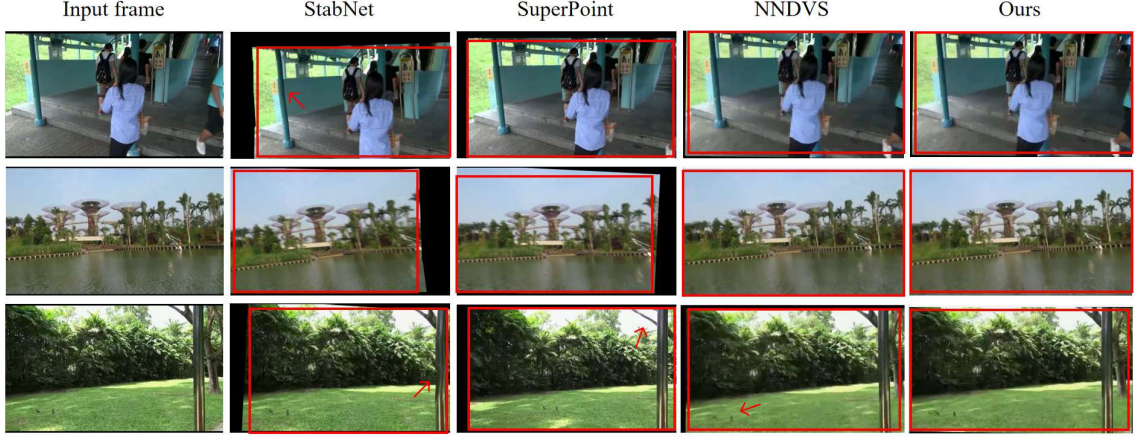


Fig. 3: Qualitative evaluation against competitive online baselines, including StabNet, SuperPoint, and NNDVS. Red boxes delineate the preserved image regions post-cropping, while red arrows denote instances of shearing and geometric distortion.

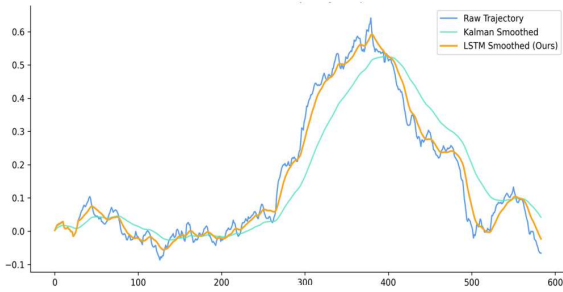


Fig. 4: Comparison chart of smoothing results obtained after processing with different trajectory smoothing strategies. Original trajectory (blue line), Kalman smoothing results (green line), and LSTM prediction results (orange line).



Fig. 5: The system's stabilized image rendering results under complex drone maneuvers with different weight configurations. The upper row shows the equal-weight strategy, while the lower row demonstrates the decoupled-weight strategy.

In the second control group, we implemented the decoupling weight strategy proposed by our system, appropriately increasing penalties for rotation components, specifically setting $\lambda_{trans} = 1.0$, $\lambda_{rot} = 2.0$, and $\lambda_{scale} = 1.0$. As shown in the lower row of Figure 5, the network demonstrated improved spatial convergence capabilities during temporal inference after introducing the higher weight constraint of $\lambda_{rot} = 2.0$. In contrast, our system explicitly restricts the magnitude of rotation divergence during single-frame compensation, substantially reducing the black borders. As these visual comparisons illustrate, the decoupled regularization strategy successfully prevents severe field-of-view (FOV) loss. Rather than merely smoothing the trajectory at the expense of the image edges, our approach preserves the natural geometric proportions and maximizes the usable FOV.

5 Limitation

By constraining the spatial transformation to a 4-DOF similarity model, we prioritize geometric fidelity and tracking robustness during high-speed rotations. Although this rigid prior effectively suppresses non-linear distortions, it inherently limits the system's ability to handle certain non-rigid artifacts. Specifically, similarity mapping is mathematically insufficient to correct for rolling shutter distortion (RSD) resulting from CMOS readout

delays. Furthermore, during aggressive 3D maneuvers—such as large-angle pitching—a global 2D transformation cannot resolve the complex perspective warping caused by significant depth disparities. Recognizing these constraints is essential to define the operational boundaries of our current framework. Under current airborne computational resource constraints, this strategy represents an engineering trade-off between real-time inference efficiency and the overall geometric consistency of the visual output. Future research could explore lightweight dynamic estimation mechanisms for camera intrinsic matrices and distortion coefficients while maintaining global motion dominance, aiming to achieve optimal balance between two-dimensional high-fidelity tracking and three-dimensional perspective correction.

6 Conclusion

To address the engineering challenge of maintaining stable video images while balancing inference efficiency and spatial geometric fidelity, this paper proposes a robust stabilization framework RGLStab, based on grid sampling optical flow and sequential networks. This framework innovatively combines the dimensionality reduction mechanism of dense flow field grids with the strictly causal unidirectional LSTM sequential reasoning, achieving low-latency pose prediction without future frame dependencies. Through the front-end joint physical screening strategy, the system fundamentally overcomes the tracking failure problem of traditional sparse features in extreme conditions such as rapid rotation; at the same time, by leveraging global four-degree-of-freedom spatial constraints and non-equal-weight decoupled loss functions, the system successfully suppresses non-rigid local distortions, while maximizing the retention of the effective physical field of view (FOV) while filtering out high-frequency mechanical tremors. Experiments show that this method not only meets real-time requirements with an online inference efficiency of 24.51 FPS, but also achieves the stabilization quality comparable to offline global algorithms, providing a highly practical and engineering-oriented solution for real-time-sensitive tasks on airborne platforms.

Declarations

Funding

The author thanks the Scientific Research Program Funded by Shaanxi Provincial Education Department (Program No.23JP057), Young Talent Fund of Association for Science and Technology in Shaanxi, China (Program No.20240135), National Natural Science Foundation of China (Grant No.62573337) and Local Community Service Research Program Funded by Shaanxi Provincial Education Department (Program No.24JC036).

References

- [1] Wang, Y., Wang, H., Wu, J., Liu, Y., Lun, Y.: UAV Standoff Tracking for Narrow-Area Target in Complex Environment. *IEEE Syst. J.* **16**(3), 4583–4594 (2021) <https://doi.org/10.1109/JSYST.2021.3093631>
- [2] Liu, S., Zhang, Z., Liu, Z., Tan, P., Zeng, B.: Minimum Latency Deep Online Video Stabilization and Its Extensions. *IEEE Trans. Pattern Anal. Mach. Intell.* **47**(2), 1238–1249 (2024) <https://doi.org/10.1109/TPAMI.2024.3493175>
- [3] Liu, S., Yuan, L., Tan, P., Sun, J.: Bundled Camera Paths for Video Stabilization. *ACM Trans. Graph.* **32**(4), 1–10 (2013) <https://doi.org/10.1145/2461912.2461995>
- [4] Grundmann, M., Kwatra, V., Essa, I.: Auto-Directed Video Stabilization with Robust L1 Optimal Camera Paths. In: *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 225–232 (2011). <https://doi.org/10.1109/CVPR.2011.5995525>
- [5] Liu, S., Yuan, L., Tan, P., Sun, J.: Steadyflow: Spatially Smooth Optical Flow for Video Stabilization. In: *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 4209–4216 (2014). <https://doi.org/10.1109/CVPR.2014.534>
- [6] Wang, Y.S., Liu, F., Hsu, P.S., Lee, T.Y.: Spatially and Temporally Optimized Video Stabilization. *IEEE Trans. Vis. Comput.*

- Graph. **19**(8), 1354–1361 (2013) <https://doi.org/10.1109/TVCG.2013.29>
- [7] Bradley, A., Klivington, J., Triscari, J., Merwe, R.: Cinematic L1 Video Stabilization with a Log-Homography Model. In: Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), pp. 1040–1048 (2021). <https://doi.org/10.1109/WACV48630.2021.00109>
 - [8] Xu, Z., Liu, S., Guo, M., Zeng, B.: Deep Online Fused Video Stabilization. IEEE Trans. Circuits Syst. Video Technol. **32**(12), 8632–8646 (2022) <https://doi.org/10.1109/TCSVT.2022.3190479>
 - [9] Zhang, Y., Liu, S., Zeng, B.: Deep Online Video Stabilization Using IMU Sensors. IEEE Robotics Autom. Lett. **7**(2), 2970–2977 (2022) <https://doi.org/10.1109/LRA.2022.3144517>
 - [10] Liu, S., Guo, M., Zhang, Z., Yuan, L., Sun, J., Zeng, B.: Meshflow: Minimum Latency Online Video Stabilization. In: Proc. Eur. Conf. Comput. Vis. (ECCV), pp. 800–815 (2016). https://doi.org/10.1007/978-3-319-46466-4_48
 - [11] Liu, T., Wan, G., Bai, H., Kong, X., Tang, B., Wang, F.: Real-Time Video Stabilization Algorithm Based on SuperPoint. IEEE Trans. Instrum. Meas. **73**, 1–13 (2024) <https://doi.org/10.1109/TIM.2023.3342849> . Art no. 5001513
 - [12] Wang, M., Yang, G.Y., Lin, J.K., Zhang, S.H., Shamir, A., Lu, S.P., Hu, S.M.: Deep Online Video Stabilization With Multi-Grid Warping Transformation Learning. IEEE Trans. Image Process. **28**(5), 2283–2292 (2018) <https://doi.org/10.1109/TIP.2018.2885233>
 - [13] Zhao, M., Ling, Q.: PWStableNet: Learning Pixel-Wise Warping Maps for Video Stabilization. IEEE Trans. Image Process. **29**, 3582–3595 (2020) <https://doi.org/10.1109/TIP.2020.2964434>
 - [14] Liu, S., Li, M., Zhu, S., Zeng, B.: Codingflow: Enable Video Coding for Video Stabilization. IEEE Trans. Image Process. **26**(7), 3291–3302 (2017) <https://doi.org/10.1109/TIP.2017.2695884>
 - [15] Yu, J., Ramamoorthi, R.: Learning Video Stabilization Using Optical Flow. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 8159–8167 (2020). <https://doi.org/10.1109/CVPR42600.2020.00818>
 - [16] He, J., Liu, S., Chen, J., Zhu, S., Zeng, B.: PixStabNet: Fast Multi-Scale Deep Online Video Stabilization With Pixel-Based Warping. IEEE Trans. Circuits Syst. Video Technol. **34**(8), 6982–6996 (2024) <https://doi.org/10.1109/TCSVT.2024.3353456>
 - [17] Liu, S., Zhang, Z., Tan, P., Zeng, B.: DeepMeshFlow: Content Adaptive Mesh Deformation for Robust Image Registration. IEEE Trans. Pattern Anal. Mach. Intell. **45**(12), 14545–14561 (2023) <https://doi.org/10.1109/TPAMI.2023.3297121>
 - [18] Lee, K.Y., Chuang, Y.Y., Chen, B.Y., Ouhyoung, M.: Video Stabilization Using Robust Feature Trajectories. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 1397–1404 (2009). <https://doi.org/10.1109/CVPR.2009.5206771>
 - [19] Zhang, Y., Yao, H., Xu, P., Ji, R., Sun, X., Liu, X.: Video Stabilization Based on Saliency Driven SIFT Matching and Discriminative RANSAC. In: Proc. 3rd Int. Conf. Internet Multimedia Comput. Service (ICIMCS), pp. 65–69 (2011). <https://doi.org/10.1145/2043674.2043693>
 - [20] Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An Efficient Alternative to SIFT or SURF. In: Proc. IEEE Int. Conf. Comput. Vis. (ICCV), pp. 2564–2571 (2011). <https://doi.org/10.1109/ICCV.2011.6126544>
 - [21] Lee, Y.C., Tseng, K.W., Chen, Y.T., Chen,

- C.C., Chen, C.S., Hung, Y.P.: 3D Video Stabilization With Depth Estimation by CNN-Based Optimization. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 10616–10625 (2021). <https://doi.org/10.1109/CVPR46437.2021.01048>
- [22] Chuan, Y., Xu, J., Huang, Y., Shi, S., Lu, S.P.: 3D Multi-Frame Fusion for Video Stabilization. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 10672–10681 (2022). <https://doi.org/10.1109/CVPR52688.2022.01042>
- [23] Lin, Z., Chen, Z., Li, Z., Xu, L., Yu, J.: GaVS: 3D-Grounded Video Stabilization via Temporally-Consistent Local Reconstruction and Rendering. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 19742–19752 (2024). <https://doi.org/10.1109/CVPR52733.2024.01865>
- [24] Liu, F., Gleicher, M., Jin, H., Agarwala, A.: Content-Preserving Warps for 3D Video Stabilization. *ACM Trans. Graph.* **28**(3), 1–9 (2009) <https://doi.org/10.1145/1531326.1531350>
- [25] Liu, S., Wang, Y., Yuan, L., Bu, J., Tan, P., Sun, J.: Video Stabilization With a Depth Camera. In: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 89–95 (2012). <https://doi.org/10.1109/CVPR.2012.6247662>
- [26] Smith, B.M., Zhang, L., Jin, H., Agarwala, A.: Light Field Video Stabilization. In: Proc. IEEE Int. Conf. Comput. Vis. (ICCV), pp. 341–348 (2009). <https://doi.org/10.1109/ICCV.2009.5459416>
- [27] Xu, S.Z., Hu, J., Wang, M., Mu, T.J., Hu, S.M.: Deep Video Stabilization Using Adversarial Networks. *Comput. Graph. Forum* **37**(7), 267–276 (2018) <https://doi.org/10.1111/cgf.13530>
- [28] Yu, J., Ramamoorthi, R.: Robust Video Stabilization by Optimization in CNN Weight Space. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 3800–3808 (2019). <https://doi.org/10.1109/CVPR.2019.00392>
- [29] Hong, M., Lu, Y., Ye, N., Lin, C., Zhao, Q., Liu, S.: Unsupervised Homography Estimation with Coplanarity-Aware GAN. In: Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 17663–17672 (2022). <https://doi.org/10.1109/CVPR52688.2022.01715>
- [30] Ye, N., Wang, C., Fan, H., Liu, S.: Motion Basis Learning for Unsupervised Deep Homography Estimation with Subspace Projection. In: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 13117–13125 (2021). <https://doi.org/10.1109/ICCV48922.2021.01288>
- [31] Teed, Z., Deng, J.: RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. In: Proc. Eur. Conf. Comput. Vis. (ECCV), pp. 402–419 (2020). https://doi.org/10.1007/978-3-030-58536-5_24
- [32] Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **24**(6), 381–395 (1981) <https://doi.org/10.1145/358669.358692>
- [33] Lucas, B.D., Kanade, T.: An Iterative Image Registration Technique with an Application to Stereo Vision. In: Proc. 7th Int. Joint Conf. Artif. Intell. (IJCAI), pp. 674–679 (1981)
- [34] Horn, B.K.P., Schunck, B.G.: Determining Optical Flow. *Artif. Intell.* **17**(1-3), 185–203 (1981) [https://doi.org/10.1016/0004-3702\(81\)90024-2](https://doi.org/10.1016/0004-3702(81)90024-2)
- [35] Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Van Der Smagt, P., Cremers, D., Brox, T.: FlowNet: Learning Optical Flow with Convolutional Networks. In: Proc. IEEE Int. Conf. Comput. Vis. (ICCV), pp. 2758–2766 (2015). <https://doi.org/10.1109/ICCV.2015.316>
- [36] Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. In:

- Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 8934–8943 (2018). <https://doi.org/10.1109/CVPR.2018.00931>
- [37] Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. *Neural Comput.* **9**(8), 1735–1780 (1997) <https://doi.org/10.1162/neco.1997.9.8.1735>
- [38] Tibshirani, R.: Regression Shrinkage and Selection via the Lasso. *J. Roy. Stat. Soc. Ser. B (Methodol.)* **58**(1), 267–288 (1996) <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>
- [39] Hoerl, A.E., Kennard, R.W.: Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* **12**(1), 55–67 (1970) <https://doi.org/10.1080/00401706.1970.10488634>
- [40] Zhang, Z., Liu, Z., Tan, P., Zeng, B., Liu, S.: Minimum Latency Deep Online Video Stabilization. In: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 23030–23039 (2023). <https://doi.org/10.1109/ICCV51070.2023.02106>
- [41] Xu, Y., Zhang, J., Maybank, S.J., Tao, D.: DUT: Learning Video Stabilization by Simply Watching Unstable Videos. *IEEE Trans. Image Process.* **31**, 4306–4320 (2022) <https://doi.org/10.1109/TIP.2022.3177341>
- [42] Liu, Y.-L., Lai, W.-S., Yang, M.-H., Chuang, Y.-Y., Huang, J.-B.: Hybrid Neural Fusion for Full-frame Video Stabilization. In: Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), pp. 2263–2273 (2021). <https://doi.org/10.1109/ICCV48922.2021.00227>
- [43] Kalman, R.E.: A New Approach to Linear Filtering and Prediction Problems. *J. Basic Eng.* **82**(1), 35–45 (1960) <https://doi.org/10.1115/1.3662552>