Harnessing Meta-Learning for Improving Full-Frame Video Stabilization

Muhammad Kashif Ali[1] Eun Woo Im[2] Dongjin Kim[1] Tae Hyun Kim[1†]
{kashifali, iameuandyou, dongjinkim, taehyunkim}@hanyang.ac.kr
[1]Dept. of Computer Science, [2]Dept. of Artificial Intelligence, Hanyang University

Abstract

Video stabilization is a longstanding computer vision problem, particularly pixel-level synthesis solutions for video stabilization which synthesize full frames add to the complexity of this task. These techniques aim to stabilize videos by synthesizing full frames while enhancing the stability of the considered video. This intensifies the complexity of the task due to the distinct mix of unique motion profiles and visual content present in each video sequence, making robust generalization with fixed parameters difficult. In our study, we introduce a novel approach to enhance the performance of pixel-level synthesis solutions for video stabilization by adapting these models to individual input video sequences. The proposed adaptation exploits low-level visual cues accessible during test-time to improve both the stability and quality of resulting videos. We highlight the efficacy of our methodology of "test-time adaptation" through simple fine-tuning of one of these models, followed by significant stability gain via the integration of meta-learning techniques. Notably, significant improvement is achieved with only a single adaptation step. The versatility of the proposed algorithm is demonstrated by consistently improving the performance of various pixel-level synthesis models for video stabilization in real-world scenarios.

1. Introduction

Today, the act of capturing and sharing visual content is deeply ingrained in our daily lives. Millions of users rely on social networking platforms like YouTube and Facebook to document and share their favorite experiences with others. However, the lack of specialized stabilization equipment, such as gimbals, often results in noticeably shaky and unstable videos. This jitter affects the overall user experience and hinders effective visual communication. Consequently, the field of video stabilization has attracted considerable attention from both videographers and researchers alike, offering the potential to enhance the visual experience and support various downstream vision tasks.

Traditionally, video stabilization methods have followed a straightforward pipeline of motion estimation, smoothing, and compensation techniques involving spatial transformations. Despite significant efforts to improve these transformation methods, the restoration process often comes at the expense of losing valuable visual content due to pixel projection, leading to irregular boundaries near the edges of stabilized videos. To mitigate this issue, cropping is commonly employed, resulting in loss of visual resolution. However, recent advances in deep learning methodologies have brought new possibilities for content preservation on the cropped region. Approaches such as inpainting the missing regions [9, 43] or defining an end-to-end pipeline that simultaneously stabilizes and synthesizes missing regions [1, 7, 31] offer promising solutions. However, achieving end-to-end feed-forward pixel-level stabilization remains challenging due to the inherent difficulty of this task and the diverse scenarios in real-world video.

Notably, the pioneering works of Choi *et al*. [7] and Ali *et al*. [1] have initiated the exploration of end-to-end full-frame video stabilization methods. Choi *et al*. [7] introduced an optical flow-based frame interpolation method (termed DIFRINT) that stabilizes videos through multiple temporal interpolations. On the other hand, Ali *et al*. [1] proposed Deep Motion-Blind Video Stabilization (DMBVS), a feed-forward method, which is trained on a dataset that consists of stable and unstable videos with similar perspectives. Despite their contributions, both methods face certain limitation, for instance, DIFRINT encounters challenges in preserving perceptual quality over multiple interpolation iterations and is prone to temporal artifacts near the motion boundaries where occlusion and dis-occlusion occur. Conversely, DMBVS generates visually appealing frames but lacks a mechanism to control the level of stability in the resulting videos.

To overcome these limitations, one potential approach is to make these models adaptive and leverage the spatiotemporal cues present in specific scenes, similar to the strategies

†: Corresponding author.

employed by classical approaches based on spatial transformations. However, a shortcoming of test-time adaptation in neural approaches is the considerable time and resources required to adapt to new data. This can be alleviated by employing techniques investigated in meta-learning literature, as similar techniques have been proven effective in various computer vision tasks such as video super-resolution [25], frame interpolation [11], and visual tracking methods [8]. We hypothesize that these techniques can also improve video stabilization approaches by quickly adapting to the input data at test time without using the ground truth stable data. Using these techniques, we can combine the strengths of deep learning methods, which provide superior quality, with classical methods that provide better stability, along with the added benefit of giving users more control over the stability and quality of the resulting videos.

In this work, we propose a scene-adaptive video stabilization method that can quickly adapt to unseen videos at test time. At test time, we improve both the picture quality and stability of full-frame video stabilization models. To the best of our knowledge, this is the first integration of meta-learning in the field of video stabilization. The proposed fast adaptation algorithm can be seamlessly integrated with any off-the-shelf end-to-end pixel synthesis stabilization models. Additionally, it allows the adapted models to achieve an $\sim 8\%$ absolute gain in stability and provides state-of-the-art results for pixel synthesis methods for video stabilization.

We summarize our contributions as follows:

• We integrate the meta-learning algorithm, which improves the performance of full-frame video stabilization models by adapting model parameters to various scenes with distinct motion profiles and content.

• Our method equips these fixed-performance models with a moderate control mechanism for various aspects of video stabilization and consistently improves the performance in these aspects by increasing the number of adaptation steps.

• We achieve SOTA video stabilization results on the evaluation datasets and our method outperforms the long-standing SOTA methods for this task.

2. Related works

This section summarizes the related literature on video stabilization and meta-learning for computer vision tasks.

2.1. Video stabilization

Conventionally, video stabilization approaches can be classified into three distinct categories, 3D, 2.5D and 2D approaches. The 3D approaches for video stabilization model the camera trajectories in the 3D space. Various techniques such as depth information [29], gyroscopic data [22] structure from motion [27], light fields [37], and 3D plane constraints [50] have been used to stabilize videos in 3D space. Despite their ingenious formulations, these approaches face difficulties in handling dynamic scenes containing multiple moving objects; therefore, 2D approaches which limit their scope to spatial transformations like homography and affine transformations became the tool of choice for researchers. Generally, these approaches track and stabilize the trajectories of prominent features. Doing so introduces loss of visual content near the frame boundaries which is often concealed by cropping and up-scaling the resultant video.

For 2D stabilization, Buehler *et al*. [4] estimated camera poses in shaky videos and used non-metric image-based rendering to stabilize videos. Matsushita *et al*. [34] estimated simplistic 2D global transformations to warp the unstable frames to produce stable video, and Liu *et al*. [30] extended this phenomenon to grid-based warping for smoothing feature trajectories. Grundmann *et al*. [18] presented an L_1-based objective function for estimating stable camera trajectories, whereas Liu *et al*. [28] utilized eigen-trajectory smoothing for this task. Goldstein *et al*. [17], Lee *et al*. [24], and Wang *et al*. [42] employed epipolar geometry-based optimization models for stabilizing videos.

Inspired by these approaches and looking at their shortcomings in handling the independent motion of multiple objects, Liu *et al*. [30] highlighted the importance of "*relatively*" denser inter-frame motion through optical flow for video stabilization. Their findings inspired most of the modern video stabilization methodologies that are currently being used professionally to this day in apps like Blink, Adobe Premiere Pro, and Deshaker. Many recent works [7, 31, 43, 44, 46, 47] rely on optical flow as an irreplaceable backbone for the definition of their approaches. Geo et al.[16] further improved on these methods and fine-tuned a conventional flow estimation network to estimate only the camera motion component of optical flow (termed global optical flow) and used it to define warping fields for video stabilization. Please note that, unlike the conventional deep stabilization methods, Ali *et al*. [1] highlighted the importance of perspective in training data and the power of traditional deep convolutional neural networks (CNNs) by learning to synthesize stable frames entirely through learned implicit motion compensation from neighboring frames, and Choi *et al*. [7] proposed an iterative interpolation strategy for stabilizing videos. Please note that these two methods are the only proposed methods for pixel synthesis end-to-end full-frame video stabilization.

2.2. Meta learning and test-time optimization

For deep video stabilization methods, some literature has been investigated on test-time adaptation inspired by the conventional optimization approaches. Yu *et al*. [46] proposed to stabilize videos by optimizing the motion vector warp field in CNN weight-space. Liu *et al*. [31] propose to learn radiance fields for distinct scenes, and Xu *et*

al. [44] defined a pipeline inspired by [18, 30] with the help of a modular pipeline catering to estimating and iteratively smoothing the motion trajectories and reprojecting the unstable frames to follow a smooth global motion profile. Despite the ingenuity of these approaches, these methods significantly hamper the time required for stabilizing videos.

Contrary to the conventional optimization-based video stabilization approaches, we aim to investigate faster test-time adaptability for full-frame video stabilization approaches inspired by its recent success in various computer vision tasks such as video super-resolution [19, 25], visual tracking [8], video segmentation [3], object detection [13], human pose estimation [6], image enhancement [33], and video frame interpolation [10]. Typically, meta-learning algorithms can be categorized into three main groups: metric-based, network-based, and optimization (or gradient)-based algorithms. From the optimization-based category of meta-learning, model agnostic meta-learning (MAML) [14] has become the tool of choice for researchers investigating computer vision tasks [5, 15, 20, 23, 26, 32, 36, 38, 40, 45, 49, 51] due to its effectiveness, generalizability, and simplicity.

In light of recent literature, and its success in low-level computer vision tasks, we investigate the applicability of this technique for pixel-level synthesis solutions for video stabilization and propose a new algorithm that combines the strengths of conventional spatial transformation-guided video stabilization approaches and regressive properties of pixel-level synthesis video stabilization approaches. The proposed algorithm allows the parameters of the feed-forward video stabilization models to be updated quickly with respect to the unique motion profiles and diverse image content present in each scene and allows the adapted model to stabilize extremely shaky videos while preserving visual quality and resolution. The proposed model also provides the user with the ability to control the level of stability and quality preservation (up to a certain degree); which is unattainable with currently available regressive solutions for this task.

3. Proposed method

This section begins by presenting the problem setup of pixel-level regressive video stabilization. Next, we discuss the proposed algorithm, outline the meta-training objective functions, and discuss the inference strategy.

3.1. Problem set-up

Consider an unstable video containing n frames as $V = \{I_0, I_1, ..., I_n\}$. The goal of the video stabilization methods is to predict a stable video $\hat{V} = \{\hat{I}_0, \hat{I}_1, ..., \hat{I}_n\}$ using a stabilization network f_θ given the unstable input video V, and the predicted video $\hat{V}$ contains similar content to V with a stabilized camera trajectory. Conventionally, stabilization methods based on pixel synthesis [1, 7] employ a sliding window strategy that considers a local temporal window containing $2k+1$ frames ($\{I_{t-k}, ..., I_t, ..., I_{t+k}\}$) and produce a stabilized frame $\hat{I}_t$ as:

$$\hat{I}_t = f_\theta(S_t), \tag{1}$$

where S_t denotes the local temporal window of $2k+1$ consecutive frames. This temporal window strategy allows the model to regress missing information in synthesized stable frames. For instance, temporal window of 5 consecutive unstable frames (*i.e*. $S_t = \{I_{t-2}, I_{t-1}, I_t, I_{t+1}, I_{t+2}\}$) is used in DMBVS, and a temporal window of 3 consecutive frames with frame recurrence (*i.e*. $S_t = \{\hat{I}_{t-1}, I_t, I_{t+1}\}$) is utilized in DIFRINT. Note that, the initial k and last k frames cannot be stabilized with window-based approaches, but we use $0 \leq t \leq T$ for notational simplicity throughout this paper.

These pixel-synthesis methods are straightforward and allow for end-to-end learning and inference. However, one of the main drawbacks of these works is the limited performance in terms of stability. While the frame recurrence schemes can improve the stability of these methods by propagating synthesized content to regress future frames and can be used with any window-based approach, these approaches can also compromise the quality and introduce wobble (jitter) artifacts.Despite the limited performance in stabilization, pixel-level synthesis solutions are still promising, because they can easily produce full-frame videos after stabilization. Therefore, we formulate our fast adaptation method based on these pixel-level synthesis approaches to improve both stability and image quality.

3.2. Meta-learning for video stabilization

Our key observation highlights the challenge that pixel-level synthesis stabilization models face when dealing with motion in specific scenarios. This challenge arises from biases in conventional training data and the complexities associated with using motion cues from raw pixel values. Therefore, we hypothesize that in real-world videos, the motion profiles can vary significantly even within the same video content, for which models with fixed parameters might be ineffective; thus, to make these models more effective, we propose a fast test-time adaptation strategy that allows these models to explicitly look for and utilize visual cues for specific unique scenarios for better compensation of camera shakes. Specifically, to aid the adaptation process, we use MAML [14], which is known for its ability to effectively adapt to new tasks. The MAML algorithm consists of two components: an inner loop and an outer loop. Within the inner loop, the parameters of the models are adapted through a small number of adaptation steps for each specified task. Following this adaptation, in the outer loop, test sets for the task in the inner loop are sampled to evaluate the generalization of the adapted model. In this work, to define a scene-adaptive video stabilization approach, we consider a

short sequence of frames as a "*task*"; which is then used for fast adaptation to unseen videos through the proposed algorithm. We employ a feed-forward video stabilization network f_θ, which takes a set of *2k + 1* neighboring frames as in Eq. 1 to synthesize its stable counterpart $\hat{I}_t$, and we use the DMBVS and DIFRINT as our baselines. The task in our formulation is defined as the minimization of both of the aforementioned objectives in the MAML framework on *T* consecutive input frame sequences from unstable videos. The overall process of our proposed meta-training process is illustrated in Fig. 1.

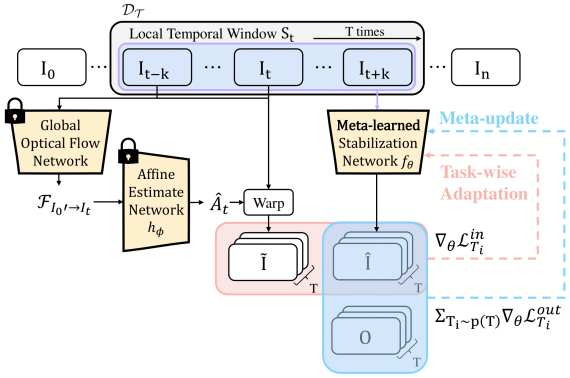

Figure 1. **Overview of the proposed meta-training process.** This figure illustrates the overall pipeline of the training process. The model in the inner loop gets a sequence of local temporal windows ($S_t \in \mathcal{D}_\mathcal{T}$) and synthesizes stable frames. The synthesized frames are penalized according to the aligned frames in the inner loop. For the outer loop, the deviation of synthesized frames is measured with the corresponding DeepStab [39] stable frames. At inference time, only the inner loop optimization is needed.

During the training-phase, each task $\mathcal{T}_i$ is sampled from the DeepStab dataset [39] ($\mathcal{D}_{\mathcal{T}_i}$). The inner loop update is governed with the help of an inner loop loss function $\mathcal{L}_{\mathcal{T}_i}^{\text{in}}$ which does not require the ground truth counterpart (as shown in the Fig. 1), whereas, the parameter update at the meta-stage (outer loop) is governed by $\mathcal{L}_{\mathcal{T}_i}^{\text{out}}$ for which we utilize the stable videos from the same dataset. In our formulation, the inner loop loss is focused on input-specific information available at test time which can be used to improve both stability and perceptual quality, whereas, the outer loop loss focuses more on visual quality to instill a sense of mitigating jerk-related degradations such as blur and distortions, hence it requires the stable counterparts of the DeepStab [39] videos; thus, meta-learning is employed to take into consideration both the input specific cues at test time while making the models under consideration stronger in each of the concerned aspect of video stabilization. It is worth noting that despite focusing more on one aspect, both the discussed losses contain parts that penalize deviation from other aspects as well.

3.2.1 Objective functions

Ali *et al*. [1] showed that various motion-related objectives can be abstracted in pixel space, therefore, we implicitly define our motion penalties in both pixel space with the help of a rigid transform estimation module and optical flow space, as there is no ground-truth available for video stabilization, and the videos in the DeepStab dataset [39] contain perspective mismatch [1]. We intentionally opt for rigid transforms in our formulation, as these transforms do not consider scale and shear change, which often causes visual distortions in the transformed images. These unique properties of rigid transforms not only govern the stabilization process but also limit the deviation of visual content from that of actual content as the transformed images are wobble-free. We will now elaborate on the details of our rigid transform regression module and then define the formulation of the proposed losses $\mathcal{L}_{\mathcal{T}_i}^{\text{in}}$ and $\mathcal{L}_{\mathcal{T}_i}^{\text{out}}$, and the proposed algorithm.

First, for rigid transform estimation, we separately trained and froze our affine motion estimation network h_ϕ. This network h_ϕ is pre-trained with the global optical flow $\mathcal{F}_{I \to I'}$ (as presented in [16]) estimated between randomly transformed images I and I' with rigid transforms to regress rotation and translation parameters of the rigid affine transform. We use the global optical flow instead of a conventional optical flow as the input of our h_ϕ network since it masks the flow of dynamic objects from the evaluated flow and is also robust against crops in the input images, which aids the proposed rigid transform estimation network to focus on removing camera shake in a video rather than local motion. To be specific, the proposed network regresses the rigid affine transform parameters as follows:

$$\hat{\mathcal{A}}_{I'} = h_\phi(\mathcal{F}_{I \to I'}), \tag{2}$$

where $\hat{\mathcal{A}}_{I'}$ denotes the estimated rigid transform, and h_ϕ is the proposed affine estimation network which renders rotational and translational parameters of the rigid transformation $\hat{\mathcal{A}}_{I'}$ from the global optical flow ($\mathcal{F}_{I \to I'}$) between the frames I and I'. Then, our h_ϕ network can be used to align short sequences of input frames by estimating transformation parameters w.r.t. the first input frame as follows:

$$\begin{aligned} \hat{\mathcal{A}}_t &= h_\phi(\mathcal{F}_{I_{0'} \to I_t}),\ t \in \{1, ..., T\}, \\ \tilde{I}_t &= \mathcal{W}(I_t, \hat{\mathcal{A}}_t)\ ,\ \tilde{V} = \{\tilde{I}_{0'}, \tilde{I}_1, ..., \tilde{I}_T\}. \end{aligned} \tag{3}$$

Here, $\hat{\mathcal{A}}_t$ denotes the estimated rigid transform that aligns frame I_t to the first frame ($I_{0'}$) of the sequence, T denotes the number of consecutive frames, $\mathcal{W}$ represents the spatial warp operator, and $\tilde{I}_t$ refers to the warped frame. Please note that $I_{0'}$ denotes the first frame of the sampled short sequence instead of the actual first frame of the video. The set ($\tilde{V}$) indicates the aligned frames. Note that($I_{0'}$) is used as the reference frame, so alignment is not required, but $\tilde{I}_{0'}$ is used to keep the notation consistent.

Figure 2. **Affine alignment.** This affine alignment strategy is analogous to the classical stabilization strategies which estimate and smooth transforms to stabilize videos. Please note that these frames are not neighboring frames and were selected to highlight the crops near the image boundaries in aligned frames $\tilde{V}$.

These aligned frames can be used as a stabilization guide for the proposed algorithm, but these frames include significant cropped regions near the image boundaries as shown in Fig. 2; thus, these frames cannot be used directly as ground-truth stable frames like the ones used in DMBVS. Therefore, we define our inner loop loss for meta-learning as the sum of global camera motion and perceptual distance between these aligned frames and the regressed frames from the feed-forward stabilization networks f_θ as follows:

$$\mathcal{L}^{\text{in}}_{\mathcal{T}_i} = \lambda_s \cdot \mathcal{L}^{\text{in}}{}_{\text{stability}} + \lambda_p \cdot \mathcal{L}^{\text{in}}{}_{\text{quality}}, \tag{4}$$

where λ_s and λ_p are associated weights for stability and quality loss, respectively. The inner loop stability loss ($\mathcal{L}^{\text{in}}{}_{\text{stability}}$) is defined as the absolute mean of global optical flow between the regressed frame $\hat{I}_t$ and the rigid-affine aligned frame $\tilde{I}_t$ as:

$$\mathcal{L}^{\text{in}}{}_{\text{stability}} = \sum_{t=1}^{T} \frac{1}{N} \sum_{N} |\mathcal{F}_{\hat{I}_t \to \tilde{I}_t}|. \tag{5}$$

Here, N represents the total number of pixels in the regressed frame. Please note that the employed global optical flow estimation network is quite robust against augments that resemble the cropped regions in the warped frames $\tilde{I}_t$ and fills these holes by utilizing the visual context from the input images[1]. The intuition behind this loss formulation is to enforce dense alignment between the regressed and aligned sequences, as, ideally, the regressed frames and the aligned frames should align perfectly. However, this loss by itself cannot justify the synthesis of legible content, as there can exist multiple solutions to the optical flow equation [2]; therefore, strong visual penalties should be introduced to ensure content preservation. We introduce these penalties in the form of perceptual loss [21], a contextual loss, and a feature-based gram matrix loss to preserve the visual content and style of the input videos. Please note that throughout our experiments, we fix $T = 5$ due to resource limitations. The proposed loss to secure video quality is defined as:

[1]Please refer to the supplementary material for robustness comparison of the employed and a conventional optical flow estimation network.

$$\begin{aligned}\mathcal{L}^{\text{in}}{}_{\text{quality}} = &\sum_{t=0}^{T} \sum_{l} \left\| \phi_l\left(\hat{I}_t\right) - \phi_l\left(\tilde{I}_t\right) \right\|_2^2 \\ &+ \sum_{t=0}^{T} \sum_{l} \left\| G(\phi_l\left(\hat{I}_t\right)) - G(\phi_l\left(\tilde{I}_t\right)) \right\|_2^2 \\ &- \log(CX(\phi_l(\hat{I}_t), \phi_l(\tilde{I}_t))).\end{aligned} \tag{6}$$

Here $\phi_l(\cdot)$ represents layers of a VGG-16 network till the layer $relu\_4\_3$ (trained on the ImageNet dataset [12]). G represents the gram matrix of features extracted from the corresponding layer l and $CX(\cdot)$ represents contextual loss. We employ the contextual and perceptual losses in our formulation in line with the previous literature [1], which has shown the effectiveness of these losses for video stabilization. In particular, the addition of gram matrix loss further encourages the models to synthesize realistic frames.

The combination of both of these losses is used to carry out the inner loop update of the proposed algorithm to obtain the adapted network parameter θ_i'. Please note that this inner loop update step can be repeated M times.

Next, within the outer loop, our network parameters are updated to minimize the different stability and quality penalties for $f_{\theta_i'}$ w.r.t. θ on different sampled frame sequences along with their stable counterparts from the DeepStab dataset [39]. In the outer loop update, we focus more on the qualitative objectives due to the availability of stable videos which contain roughly the same content with better quality as compared to the unstable videos.

The motion loss for the outer loop update is defined as the deviation between the global camera motion of synthesized frames and their stable counterparts as:

$$\mathcal{L}^{\text{out}}{}_{\text{stability}} = \sum_{t=0}^{T-1} \frac{1}{N} \sum_{N} \left\| \mathcal{F}_{\hat{I}_t \to \hat{I}_{t+1}} - \mathcal{F}_{O_t \to O_{t+1}} \right\|_2^2, \tag{7}$$

where O_t represents the target stable frame in the DeepStab dataset corresponding to the predicted stable frame $\hat{I}_t$. This loss further enforces the learned stability of the model under consideration with smooth real-world trajectories. Similar to the stability loss in the inner loop, this loss alone cannot justify the preservation of legible content; therefore, a qualitative penalty is also added in the outer loop update.

Since both the stable and unstable videos in the DeepStab dataset contain large disjoint perspectives [1], a non-local criterion is needed for a quality guidance. We take inspiration from Ali *et al*. [1] to define our non-local quality penalty using contextual loss [35], which compares unaligned image regions with similar semantics and has been shown to be useful in improving the quality of synthesized stable frames [1]. The outer loop quality loss with the ground-truth target O_t is defined as:

$$\mathcal{L}^{\text{out}}{}_{\text{quality}} = -\log(CX(\phi^l(\hat{I}_t), \phi^l(O_t))), \tag{8}$$

Algorithm 1: Meta-Training.

Require: uniform distribution over sequences $p(\mathcal{T})$, adaptation number M, learning rate α, β

while *not converged* **do**
  Initialize parameters $\theta_i \leftarrow \theta$;
  Sample batch of sequences $\mathcal{T}_i \sim p(\mathcal{T})$;
  foreach *i* **do**
    Sample local temporal windows $\mathcal{D}_{\mathcal{T}_i} = \{S_0, S_1, ..., S_t\}$ from $\mathcal{T}_i$;
    for $m \leftarrow 1$ *to* M **do**
      Compute $\hat{\mathbf{V}}$, $\tilde{\mathbf{V}}$ in Eq. (1), (3);
      Evaluate $\nabla_{\theta_i} \mathcal{L}^{\text{in}}_{\mathcal{T}_i}(f_{\theta_i})$ using $\mathcal{L}_{\mathcal{T}_i}$ in Eq. (4);
      $\theta_i{}' = \theta_i - \alpha \nabla_{\theta_i} \mathcal{L}^{\text{in}}_{\mathcal{T}_i}(f_{\theta_i})$;
    end
  end
  Sample $\mathcal{D}'_{\mathcal{T}_i} = \{(S_0, O_0), (S_1, O_1), ..., (S_t, O_t)\}$ from $\mathcal{T}_i$ for meta-update;
  $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}^{\text{out}}_{\mathcal{T}_i}(f_{\theta_i{}'})$ using each $\mathcal{D}'_{\mathcal{T}_i}$;
end

and the final loss for the outer update is defined as:

$$\mathcal{L}^{\text{out}}_{\mathcal{T}_i} = \mathcal{L}^{\text{out}}{}_{\text{stability}} + \mathcal{L}^{\text{out}}{}_{\text{quality}}. \tag{9}$$

3.2.2 Meta-training and inference

The overall training algorithm is presented in Alg. 1. Please note that at the test-time, only the inner loop loss is needed to update the meta-trained parameters and the updated parameters are used to synthesize the final stabilized results in a feed-forward manner. It is worth mentioning that we experimented with a fixed number of adaptation iterations and a patch size of 320×320 during the inference time to further expedite the adaptation process and empirically found that even with as low as 100 adaptation iterations on randomly sampled sequences from the test videos, the meta-trained models adapt quite well due to the similarity in motion profiles and the content of the videos. This process significantly cuts down the adaptation time as most of the videos from the evaluation dataset [30] contain over 700 frames. Our fast adaptation algorithm is presented in Alg. 2. Please refer to the accompanied supplemental for a detailed description of the implementation details and experiments.

4. Ablation study

To properly evaluate the efficacy of each of the modules and objective functions, we conducted thorough ablation studies and present our findings below. We first present the contribution of each of the losses presented and then present the category-specific hyperparameters in this section.

Objective function contribution. We explore the influence of each loss term presented in Eq. 4 from the main paper ($\mathcal{L}^{\text{in}}{}_{\text{quality}}$ and $\mathcal{L}^{\text{in}}{}_{\text{stability}}$) concerning different weights of each loss term in the adaptation process.

Algorithm 2: Meta-Inference.

Require: meta-trained model f_θ, test sequence $\mathcal{T}$, adaptation number M, learning rate α

Construct local temporal windows $\mathcal{D}_{\mathcal{T}} = \{S_0, S_1, ..., S_t\}$ from $\mathcal{T}$;
for $m \leftarrow 1$ *to* M **do**
  Compute $\hat{\mathbf{V}}$, $\tilde{\mathbf{V}}$ in Eq. (1), (3);
  Evaluate $\nabla_\theta \mathcal{L}^{\text{in}}_{\mathcal{T}}(f_\theta)$ using $\mathcal{L}_{\mathcal{T}}$ in Eq. (4);
  $\theta' = \theta - \alpha \nabla_\theta \mathcal{L}^{\text{in}}_{\mathcal{T}}(f_\theta)$;
end
Stabilize video $\hat{\mathbf{V}} = f_{\theta'}(\mathbf{V})$ with sliding window strategy in Eq. (1);
return stabilized video $\hat{\mathbf{V}}$

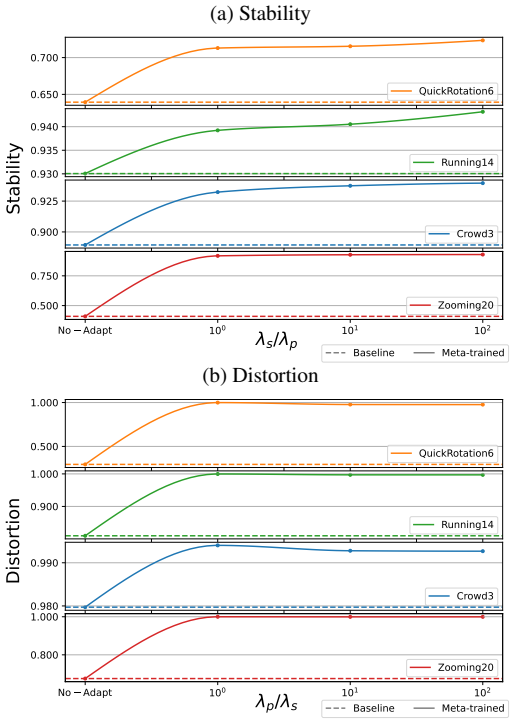

Figure 3. **Contribution of each objective function.** a) The effects of stability loss during the adaptation stage. A higher weight for the proposed stability loss positively affects the stability score. b) The effects of quality loss during the adaptation stage. A higher weight for quality loss positively affects the distortion score.

To properly ablate the contribution of each of the proposed losses, we randomly sample 4 videos from the NUS dataset [30] and repeat the adaptation process with various ratios of λ_s and λ_p, and present our findings in Fig. 3. For our ablation studies, we choose the meta-trained DMBVS [1]. Note that similar phenomenons were observed with the meta-trained DIFRINT [7], therefore, we only present the findings from one of the considered models in Fig. 3. It is evident from Fig. 3a, that increasing the weights for the proposed stability loss positively affects the stability of the resultant videos and an increasing trend is observed in terms of stability metric results. As for the quality loss,

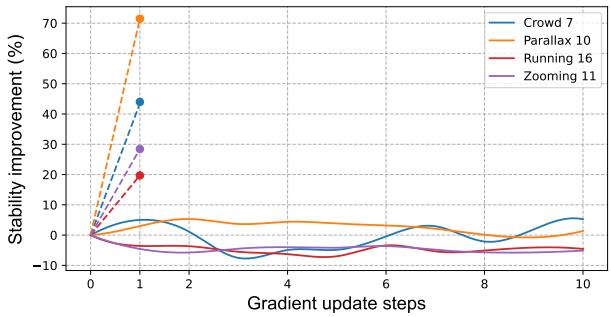

Figure 4. **Finetuning vs meta-inference.** A comparison of the finetuned and the meta-trained models highlights that it takes significant finetuning iterations for a minuscule improvement. Whereas, the proposed algorithm allows for a significant improvement with a single adaptation pass over the video sequence.

a similar increasing trend for distortion score is observed as evident from Fig. 3b. Please note that the presented results in the main manuscript and this supplemental were generated with a *10:1* ratio of λ_s and λ_p.

Category-specific ratios. Each video category within the NUS dataset [30] exhibits distinct characteristics, necessitating tailored weighing configurations to achieve optimal results. This subsection presents the findings of our study for the category-specific hyperparameters on individual video categories. Please note that the presented results (in both the main paper and this supplemental) were evaluated on hyperparameters that demonstrated optimal performance across all the video categories. However, we found the performance on distinct motion profiles can be further improved (by 1∼2%) by selecting specific weights for the stability and quality losses during the adaptation process. We present the category-specific weights in Tab. 1.

Category	Crowd	Parallax	Regular	Running	Quick Rot	Zoom
λ_s	10	1	1	10	10	1
λ_p	1	1	1	1	1	1

Table 1. **Category-specific weights (λ_s and λ_p).** This table highlights the category-specific weights for the proposed loss functions for the adaptation step. The various motion profiles from the NUS dataset [30] can be efficiently stabilized by employing these weights during the adaptation process.

Finteuning VS. meta-training. To highlight the efficacy of the proposed algorithm, we also conducted an ablation study in which we finetuned the baseline DMBVS [1] with the proposed inner-loop losses on its worst-performing videos (with a stability score of 10∼15%) from the evaluation dataset and compared the performance of its meta-trained variant with only 1 adaptation pass (please note that in both of these experiments, we opted the best settings of hyperparameters presented above). We present our findings in Fig. 4. The meta-trained model performs significantly well as compared to the baseline.

5. Experimental results

5.1. Qualitative results

For qualitative comparison, we compare our results with L1 stabilizer [18], bundled, and baselines [1, 7] in Fig. 5. The bounded regions highlight the temporal artifacts present in DIFRINT [7] and the frame recurrent extension of DMBVS [1]. The proposed algorithm mitigates these temporal artifacts successfully and produces sharper results. Due to the space limitation, we only present the qualitative comparison with the longstanding SOTA methods in the main paper and humbly request the readers to refer to the accompanied supplemental for qualitative comparison with other approaches used for quantitative comparison.

Figure 5. **Qualitative Results.** Qualitative comparison of the meta-trained, baseline models and current SOTA methods. The proposed methodology improves the stability of considered models and also mitigates the artifacts present in frame recurrent baseline results. **(Best viewed on a computer screen with zoom).**

5.2. Quantitative results

We compare the quantitative performance of both scene-adaptive models with their baseline variants on the NUS dataset [30] in terms of stability, cropping, and distortion[2] in Tab. 2. This dataset contains videos of 6 distinct categories including different motion profiles. The test-time adapted models perform significantly better than their baseline (non-adaptive) counterparts. We see an average of 5% gain in absolute stability with a single adaptation pass on the test videos for the meta-trained variant of DMBVS [1], and an average gain of 8% for DIFRINT [7]. Please note that

[2]Please refer to the accompanied supplemental for the implementation details of these metrics.

Model		Stability					
		Crowd	Parallax	Regular	Running	Quick Rot	Zoom
DMBVS [1]	Baseline	0.7315	0.7660	0.6938	0.6522	0.8453	0.7811
	Adapt$^{(1)}_{100}$	0.7584	0.7965	0.7133	0.6983	0.8906	0.8368
	Adapt$^{(5)}_{100}$	0.7616	0.8125	0.7290	0.7144	0.9046	0.8412
DIFRINT [7]	Baseline	0.7453	0.8321	0.6371	0.7143	0.9058	0.8258
	Adapt$^{(1)}_{100}$	0.8062	0.8492	0.6501	0.7218	0.9361	0.8501
	Adapt$^{(5)}_{100}$	0.8149	0.8542	0.6617	0.7410	0.9431	0.8611

Table 2. **Quantitative comparison of adapted models against baselines.** The proposed algorithm consistently improves the stability with the increasing number of adaptation iterations for both of the considered models. The subscript shows the number of sequences sampled for adaptation and the superscript denotes the adaptation number. The best stability is highlighted with a green color and the second best is highlighted with a blue color.

this gain does not come at the cost of compromising the full-frame nature of the baseline models and an improvement is also observed in terms of the distortion score as well as evident from Tab. 3. After our baseline comparison, we present a thorough quantitative assessment against well-established SOTA methods known for their stability [18, 30], recent methods [31, 39, 47, 48], and Adobe Premiere Pro 2020's professionally used warp stabilizer in Tab. 3. Despite the classical nature of the methodologies introduced in [18, 30], these approaches still produce state-of-the-art results, in terms of stability [41]. Please note that the proposed method in [48] produces video results across the entire evaluation dataset, however, it is imperative to highlight that videos generated by this method exhibit pronounced shakes in the initial frames, gradually leading to stable videos due to their inherent minimum latency constraints. This instability in the initial segment (spanning over 30 frames per video) impedes the estimation of homography for stability metric calculation. To ensure a fair comparison, we only present average results from their method where the stability metric can be computed for the entire videos.

The proposed algorithm consistently improves the results of both the considered models and equips DIFRINT to achieve SOTA results and also improves the mean stability of DMBVS without compromising the full-frame nature or quality of the stabilized videos.

Please note that the average stability of the adapted method can be further increased by opting for a higher number of adaptation iterations and higher weights for the stability losses during the adaptation process. In Tab. 2, we only present the results generated with up to a single adaptation iteration on each consecutive sequence due to the time complexity and resource limitations. In order to significantly cut down the time required for adaptation, we observe that comparable results can be achieved by adapting on a constant number of randomly sampled sequences with a higher number of adaptation iterations (as evident from Tab. 3). Furthermore, the quality of the results (as indicated by the Distortion metric) also suggests that the proposed algorithm

Method	Stability ↑	Cropping ↑	Distortion ↑
L1 [18]	0.8661	0.7392	0.9215
Bundled [30]	0.8750	0.8215	0.7781
Adobe Premiere Pro 2020*	0.8262	0.7432	0.8230
StabNet* [39]	0.7422	0.6615	0.8878
Yu and Ramamoorthi *et al.* [47]	0.7905	0.8592	0.9105
FuSta [31]	0.8037	0.9992	0.9642
Zhang et al.* [48]	0.7481	0.9592	0.9988
DMBVS [1] (baseline)	0.7372	0.9983	0.9189
DMBVS [1] + Adapt$^{(1)}_{100}$	0.7532	0.9974	0.9112
DMBVS [1] + Adapt$^{(5)}_{100}$	0.7852	0.9973	0.9461
DMBVS [1] + Adapt$^{(1)}_{all}$	0.7760	0.9999	0.9990
DMBVS [1] + Adapt$^{(1)}_{all}$ + recurrent	0.7867	0.9999	0.9818
DIFRINT [7] (baseline)	0.7904	0.9993	0.9438
DIFRINT [7] + Adapt$^{(1)}_{100}$	0.8428	0.9993	0.9587
DIFRINT [7] + Adapt$^{(5)}_{100}$	0.8528	0.9994	0.9596
DIFRINT [7] + Adapt$^{(1)}_{all}$	0.8786	0.9994	0.9569

Table 3. **Quantitative Results.** The proposed algorithm consistently improves the stability with the increasing number of adaptation iterations for both of the considered models. The proposed algorithm enables DIFRINT [7] to achieve SOTA results with a single adaptation iteration over all the frame sequences in videos from the NUS dataset [30]. Please note that the methods proposed in [39] and Adobe Premiere Pro fail to stabilize some videos; therefore, their results are averaged over only the stabilized videos.

not only improves the stability but consistently enhances the quality as well. Please note that employing the iterative strategy proposed in [7] can further enhance the stability of the resultant videos. Please refer to the accompanied supplemental for user studies and other metric results.

6. Conclusion

In this study, we aim to improve full-frame pixel-level synthesis video stabilization solutions by leveraging additional information available at test time. We introduce a meta-learning algorithm for this task, enabling rapid adaptation of model parameters for scenes containing unique motion profiles. Our proposed algorithm's versatility is demonstrated through extensive experimentation on publicly available models for this task. The proposed algorithm enables the users to control various aspects of video stabilization (to an extent), which was previously unattainable for such models, and shows consistent improvement in both stability and quality. The proposed algorithm can be seamlessly integrated with upcoming pixel-synthesis solutions for this task without additional parametric or structural changes.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00156, Fundamental research on continual meta-learning for quality enhancement of casual videos and their 3D metaverse transformation), National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2023-00222776), and Samsung Electronics Co., Ltd, and Samsung Research Funding Center of Samsung Electronics under Project Number SRFCIT1901-06.

References

[1] Muhammad Kashif Ali, Sangjoon Yu, and Tae Hyun Kim. Deep motion blind video stabilization. *arXiv preprint arXiv:2011.09697*, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[2] Muhammad Kashif Ali, Dongjin Kim, and Tae Hyun Kim. Learning task agnostic temporal consistency correction. *arXiv preprint arXiv:2206.03753*, 2022. 5

[3] Harkirat Singh Behl, Mohammad Naja, Anurag Arnab, and Philip HS Torr. Meta-learning deep visual words for fast video object segmentation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. 3

[4] Chris Buehler, Michael Bosse, and Leonard McMillan. Non-metric image-based rendering for video stabilization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001. 2

[5] Meng Cheng, Hanli Wang, and Yu Long. Meta-learning-based incremental few-shot object detection. *IEEE Transactions on Circuits and Systems for Video Technology (TCSVT)*, 2021. 3

[6] Hanbyel Cho, Yooshin Cho, Jaemyung Yu, and Junmo Kim. Camera distortion-aware 3d human pose estimation in video with optimization-based meta-learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[7] Jinsoo Choi and In So Kweon. Deep iterative frame interpolation for full-frame video stabilization. *ACM TOG*, 2020. 1, 2, 3, 6, 7, 8

[8] Janghoon Choi, Junseok Kwon, and Kyoung Mu Lee. Deep meta learning for real-time target-aware visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019. 2, 3

[9] Jinsoo Choi, Jaesik Park, and In So Kweon. Self-supervised real-time video stabilization. *arXiv preprint arXiv:2111.05980*, 2021. 1

[10] Myungsub Choi, Janghoon Choi, Sungyong Baik, Tae Hyun Kim, and Kyoung Mu Lee. Scene-adaptive video frame interpolation via meta-learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[11] Myungsub Choi, Janghoon Choi, Sungyong Baik, Tae Hyun Kim, and Kyoung Mu Lee. Test-time adaptation for video frame interpolation via meta-learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2021. 2

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 5

[13] Jiajun Deng, Yingwei Pan, Ting Yao, Wengang Zhou, Houqiang Li, and Tao Mei. Minet: Meta-learning instance identifiers for video object detection. *IEEE Transactions on Image Processing (TIP)*, 2021. 3

[14] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, 2017. 3

[15] Yuqian Fu, Chengrong Wang, Yanwei Fu, Yu-Xiong Wang, Cong Bai, Xiangyang Xue, and Yu-Gang Jiang. Embodied one-shot video recognition: Learning from actions of a virtual embodied agent. In *ACM International Conference on Multimedia (MM)*, 2019. 3

[16] Jerin Geo, Devansh Jain, and Ajit Rajwade. Globalflownet: Video stabilization using deep distilled global motion estimates. In *Winter Conference on Applications of Computer Vision (WACV)*, 2023. 2, 4

[17] Amit Goldstein and Raanan Fattal. Video stabilization using epipolar geometry. *ACM TOG*, 2012. 2

[18] Matthias Grundmann, Vivek Kwatra, and Irfan Essa. Auto-directed video stabilization with robust l1 optimal camera paths. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2, 3, 7, 8

[19] Akash Gupta, Padmaja Jonnalagedda, Bir Bhanu, and Amit K Roy-Chowdhury. Ada-vsr: Adaptive video super-resolution with meta-learning. In *ACM International Conference on Multimedia (MM)*, 2021. 3

[20] Mehrdad Hosseinzadeh and Yang Wang. Few-shot personality-specific image captioning via meta-learning. In *Conference on Robots and Vision*, 2023. 3

[21] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 5

[22] Alexandre Karpenko, David Jacobs, Jongmin Baek, and Marc Levoy. Digital video stabilization and rolling shutter correction using gyroscopes. *CSTR*, 2011. 2

[23] Jessica Lee, Deva Ramanan, and Rohit Girdhar. Metapix: Few-shot video retargeting. *arXiv preprint arXiv:1910.04742*, 2019. 3

[24] Ken-Yi Lee, Yung-Yu Chuang, Bing-Yu Chen, and Ming Ouhyoung. Video stabilization using robust feature trajectories. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2009. 2

[25] Suyoung Lee, Myungsub Choi, and Kyoung Mu Lee. Dynavsr: Dynamic adaptive blind video super-resolution. In *Winter Conference on Applications of Computer Vision (WACV)*, 2021. 2, 3

[26] Yuanze Lin, Xun Guo, and Yan Lu. Self-supervised video representation learning with meta-contrastive network. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 3

[27] Feng Liu, Michael Gleicher, Hailin Jin, and Aseem Agarwala. Content-preserving warps for 3d video stabilization. *ACM Transactions on Graphics (SIGGRAPH)*, 2009. 2

[28] Feng Liu, Michael Gleicher, Jue Wang, Hailin Jin, and Aseem Agarwala. Subspace video stabilization. *ACM TOG*, 2011. 2

[29] Shuaicheng Liu, Yinting Wang, Lu Yuan, Jiajun Bu, Ping Tan, and Jian Sun. Video stabilization with a depth camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 2

[30] Shuaicheng Liu, Lu Yuan, Ping Tan, and Jian Sun. Bundled camera paths for video stabilization. *ACM TOG*, 2013. 2, 3, 6, 7, 8

[31] Yu-Lun Liu, Wei-Sheng Lai, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Hybrid neural fusion for full-frame video stabilization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 8

[32] Yihong Lu, Jianyong Cai, Hua Zheng, and Yuanqiang Zeng. A deep meta-learning neural network for single image rain removal. In *International Congress on Image and Signal Processing, BioMedical Engineering and Informatics*, 2020. 3

[33] Long Ma, Dian Jin, Nan An, Jinyuan Liu, Xin Fan, and Risheng Liu. Bilevel fast scene adaptation for low-light image enhancement. *arXiv preprint arXiv:2306.01343*, 2023. 3

[34] Yasuyuki Matsushita, Eyal Ofek, Weina Ge, Xiaoou Tang, and Heung-Yeung Shum. Full-frame video stabilization with motion inpainting. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2006. 2

[35] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 5

[36] Xuanchi Ren, Zian Qian, and Qifeng Chen. Video deblurring by fitting to test data. *arXiv preprint arXiv:2012.05228*, 2020. 3

[37] Brandon M Smith, Li Zhang, Hailin Jin, and Aseem Agarwala. Light field video stabilization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2009. 2

[38] Guangting Wang, Chong Luo, Xiaoyan Sun, Zhiwei Xiong, and Wenjun Zeng. Tracking by instance detection: A meta-learning approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[39] Miao Wang, Guo-Ye Yang, Jin-Kun Lin, Song-Hai Zhang, Ariel Shamir, Shao-Ping Lu, and Shi-Min Hu. Deep online video stabilization with multi-grid warping transformation learning. *IEEE Transactions on Image Processing (TIP)*, 2018. 4, 5, 8

[40] Rui Wang, Bin Kang, and Wei-Ping Zhu. Meta-learning based siamese network with channel-wise self-attention for visual tracking. In *International Conference on Image, Video and Signal Processing*, 2021. 3

[41] Yiming Wang, Qian Huang, Chuanxu Jiang, Jiwen Liu, Mingzhou Shang, and Zhuang Miao. Video stabilization: A comprehensive survey. *Neurocomputing*, 2022. 8

[42] Yu-Shuen Wang, Feng Liu, Pu-Sheng Hsu, and Tong-Yee Lee. Spatially and temporally optimized video stabilization. *IEEE transactions on visualization and computer graphics*, 2013. 2

[43] Yufei Xu, Jing Zhang, and Dacheng Tao. Out-of-boundary view synthesis towards full-frame video stabilization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. 1, 2

[44] Yufei Xu, Jing Zhang, Stephen J Maybank, and Dacheng Tao. Dut: Learning video stabilization by simply watching unstable videos. *IEEE Transactions on Image Processing (TIP)*, 2022. 2, 3

[45] Tao Yang, Fan Wang, Junfan Lin, Zhongang Qi, Yang Wu, Jing Xu, Ying Shan, and Changwen Chen. Toward human perception-centric video thumbnail generation. In *ACM International Conference on Multimedia (MM)*, 2023. 3

[46] Jiyang Yu and Ravi Ramamoorthi. Robust video stabilization by optimization in cnn weight space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[47] Jiyang Yu and Ravi Ramamoorthi. Learning video stabilization using optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 8

[48] Zhuofan Zhang, Zhen Liu, Ping Tan, Bing Zeng, and Shuaicheng Liu. Minimum latency deep online video stabilization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2023. 8

[49] Zixu Zhao, Yueming Jin, Bo Lu, Chi-Fai Ng, Qi Dou, Yun-Hui Liu, and Pheng-Ann Heng. One to many: Adaptive instrument segmentation via meta learning and dynamic online adaptation in robotic surgical video. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021. 3

[50] Zihan Zhou, Hailin Jin, and Yi Ma. Plane-based content preserving warps for video stabilization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[51] Nannan Zou, Honglei Zhang, Francesco Cricri, Hamed R Tavakoli, Jani Lainema, Miska Hannuksela, Emre Aksu, and Esa Rahtu. L 2 c–learning to learn to compress. In *IEEE 22nd International Workshop on Multimedia Signal Processing*, 2020. 3