Blind Video Deflickering by Neural Filtering with a Flawed Atlas

Chenyang Lei[1,2*] Xuanchi Ren[3,4*] Zhaoxiang Zhang[1,5†] Qifeng Chen[6†]
[1]CAIR, HKISI-CAS [2]Princeton University [3]University of Toronto
[4]Vector Institute [5]CASIA [6]HKUST

Figure 1. **Blind deflickering performance of our approach on *Betty Boop* (1934).** Many videos can have flickering artifacts for various reasons. Our approach takes only the input video and removes the flicker automatically without any extra guidance.

Abstract

Many videos contain flickering artifacts; common causes of flicker include video processing algorithms, video generation algorithms, and capturing videos under specific situations. Prior work usually requires specific guidance such as the flickering frequency, manual annotations, or extra consistent videos to remove the flicker. In this work, we propose a general flicker removal framework that only receives a single flickering video as input without additional guidance. Since it is blind to a specific flickering type or guidance, we name this "blind deflickering." The core of our approach is utilizing the neural atlas in cooperation with a neural filtering strategy. The neural atlas is a unified representation for all frames in a video that provides temporal consistency guidance but is flawed in many cases. To this end, a neural network is trained to mimic a filter to learn the consistent features (e.g., color, brightness) and avoid introducing the artifacts in the atlas. To validate our method, we construct a dataset that contains diverse real-world flickering videos. Extensive experiments show that our method achieves satisfying deflickering performance and even outperforms baselines that use extra guidance on a public benchmark. The source code is publicly available at https://chenyanglei.github.io/deflicker.

1. Introduction

A high-quality video is usually temporally consistent, but many videos suffer from flickering artifacts for various reasons, as shown in Figure 2. For example, the brightness of old movies can be very unstable since some old cameras with low-quality hardware cannot set the exposure time of each frame to be the same [16]. Besides, high-speed cameras with very short exposure time can capture the high-frequency (e.g., 60 Hz) changes of indoor lighting [25]. Effective processing algorithms such as enhancement [38, 45], colorization [29, 60], and style transfer [33] might bring flicker when applied to temporally consistent videos. Videos from video generations approaches [46, 50, 61] also might contain flickering artifacts. Since temporally consistent videos are generally more visually pleasing, removing the flicker from videos is highly desirable in video processing [9, 13, 14, 54, 59] and computational photography.

In this work, we are interested in a general approach for deflickering: (1) it is agnostic to the patterns or levels of flickering (e.g., old movies, high-speed cameras, processing artifacts), (2) it only takes a single flickering video and does not require other guidance (e.g., flickering types, extra consistent videos). That is to say, this model is blind to flickering types and guidance, and we name this task as

[*] Equal contribution.
[†] Corresponding authors.

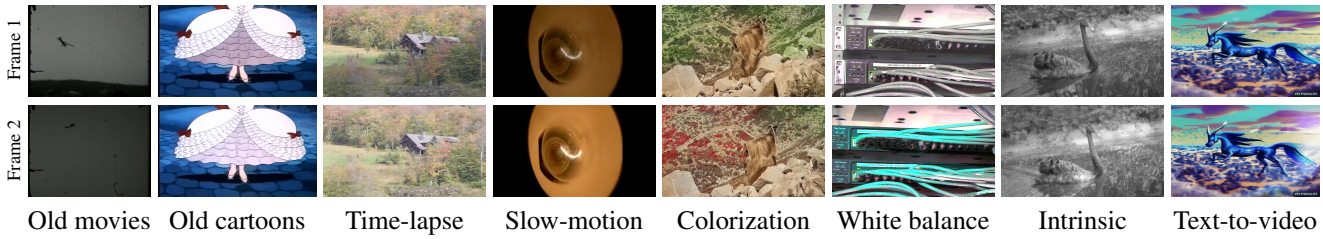

Figure 2. **Videos with flickering artifacts.** Flickering artifacts exist in unprocessed videos, including old movies, old cartoons, time-lapse videos, and slow-motion videos. Besides, some processing algorithms [6, 18, 60] can introduce the flicker to temporally consistent unprocessed videos. Synthesized videos from video generations approaches [46, 50, 61] also might be temporal inconsistent. *Blind deflickering* aims to remove various types of flickering with only an unprocessed video as input.

blind deflickering. Thanks to the blind property, blind deflickering has very wide applications.

Blind deflickering is very challenging since it is hard to enforce temporal consistency across the whole video without any extra guidance. Existing techniques usually design specific strategies for each flickering type with specific knowledge. For example, for slow-motion videos captured by high-speed cameras, prior work [25] can analyze the lighting frequency. For videos processed by image processing algorithms, blind video temporal consistency [31, 32] obtains long-term consistency by training on a temporally consistent unprocessed video. However, the flickering types or unprocessed videos are not always available, and existing flickering-specific algorithms cannot be applied in this case. One intuitive solution is to use the optical flow to track the correspondences. However, the optical flow from the flickering videos is not accurate, and the accumulated errors of optical flow are also increasing with the number of frames due to inaccurate estimation [7].

With two key observations and designs, we successfully propose the first approach for *blind deflickering* that can remove various flickering artifacts without extra guidance or knowledge of flicker. First, we utilize a unified video representation named neural atlas [26] to solve the major challenge of solving long-term inconsistency. This neural atlas tracks all pixels in the video, and correspondences in different frames share the same pixel in this atlas. Hence, a sequence of temporally consistent frames can be obtained by sampling from the shared atlas. Secondly, while the frames from the shared atlas are consistent, the structures of images are flawed: the neural atlas cannot easily model dynamic objects with large motion; the optical flow used to construct the atlas is imperfect. Hence, we propose a neural filtering strategy to take the treasure and throw the trash from the flawed atlas. A neural network is trained to learn the invariant under two types of distortion, which mimics the artifacts in the atlas and the flicker in the video, respectively. At test time, this network works as a filter to preserve the consistency property and block the artifacts from the flawed atlas.

We construct the first dataset containing various types of flickering videos to evaluate the performance of blind deflickering methods faithfully. Extensive experimental results show the effectiveness of our approach in handling different flicker. Our approach also outperforms blind video temporal consistency methods that use an extra input video as guidance on a public benchmark.

Our contributions can be summarized as follows:

- We formulate the problem of blind deflickering and construct a deflickering dataset containing diverse flickering videos for further study.
- We propose the first blind deflickering approach to remove diverse flicker. We introduce the neural atlas to the deflickering problem and design a dedicated strategy to filter the flawed atlas for satisfying deflickering performance.
- Our method outperforms baselines on our dataset and even outperforms methods that use extra input videos on a public benchmark.

2. Related Work

Task-specific deflikcering. Different strategies are designed for specific flickering types. Kanj et al. [25] propose a strategy for high-speed cameras. Delon et al. [16] present a method for local contrast correction, which can be utilized for old movies and biological film sequences. Xu et al. [56] focus on temporal flickering artifacts from GAN-based editing. Videos processed by a specific image-to-image translation algorithm [6, 18, 24, 30, 33, 41, 60, 62] can suffer from flickering artifacts, and blind video temporal consistency [7, 27, 28, 31, 32, 57] is designed to remove the flicker for these processed videos. These approaches are blind to a specific image processing algorithm. However, the temporal consistency of generated frames is guided by a temporal consistent unprocessed video. Bonneel et al. [7] compute the gradient of input frames as guidance. Lai et al. [27] input two consecutive input frames as guidance. Lei et al. [31] directly learn the mapping function between input and processed frames. While these approaches achieve satisfying performance on many tasks, a temporally consistent video is not always available. For example, for many flickering videos such as old movies and synthesized videos

from video generation methods [19, 50, 61], the original videos are temporal inconsistent. A concurrent preprint [2] attempts to eliminate the need for unprocessed videos during inference. Nonetheless, it does not study other types of flickering videos. Our approach has a wider application compared with blind temporal consistency.

Also, some commercial software [3, 47] can be used for deflickering by integrating various task-specific deflickering approaches. However, these approaches require users to have a knowledge background of the flickering types. Our approach aims to remove this requirement so that more videos can be processed efficiently for most users.

Video mosaics and neural atlas. Inheriting from panoramic image stitching [8], video mosaicing is a technique that organizes video data into a compact mosaic representation, especially for dynamic scenes. It supports various applications, including video compression [23], video indexing [22], video texture [1], 2D to 3D conversion [48, 49], and video editing [44]. Building video mosaics based on homography warping often fails to depict motions. To handle the dynamic contents, researchers compose foreground and background regions by spatio-temporal masks [15], or blend the video into a multi-scale tapestry in a patch-based manner [5]. However, these approaches heavily rely on image appearance information and thus are sensitive to lighting changes and flicker.

Recently, aiming for consistent video editing, Kasten et al. [26] propose Neural Layered Atlas (NLA), which decomposes a video into a set of atlases by learning mapping networks between atlases and video frames. Editing on the atlas and then reconstructing frames from the atlas can achieve consistent video editing. Follow-up work validates its power on text-driven video stylization [4, 37] and face video editing [36]. Directly adopting NLA to blind deflickering tasks is not trivial and is mainly limited by two-fold: *(i)* its performance on the complex scene is still not satisfying with notable artifacts. *(ii)* it requires segmentation masks as guidance for decomposing dynamic objects, and each dynamic object requires an additional mapping network. To facilitate automatic deflickering, we use a single-layered atlas without the need for segmentation masks by designing an effective neural filtering strategy for the flawed atlas. Our proposed strategy is also compatible with other atlas generation techniques.

Implicit image/video representations. With the success of using the multi-layer perceptron (MLP) as a continuous implicit representation for 3D geometry [39, 40, 42], such representation has gained popularity for representing images and videos [10, 34, 51, 52]. Follow-up work extends these models to various tasks, such as image super-resolution [12], video decomposition [58], and semantic segmentation [21]. In our work, we follow [26] to employ a coordinate-based MLP to represent the neural atlases.

3. Method

3.1. Overview

Let $\{I_t\}_{t=1}^{T}$ be the input video sequences with flickering artifacts where T is the number of video frames. Our approach aims to remove the flickering artifacts to generate a temporally consistent video $\{O_t\}_{t=1}^{T}$. Flicker denotes a type of temporal inconsistency that correspondences in different frames share different features (e.g., color, brightness). The flicker can be either global or local in the spatial dimension, either long-term or short-term in the temporal dimension.

Figure 3 shows the framework of our approach. We tackle the problem of blind deflickering through the following key designs: (i) We first propose to use a single *neural atlas (Section 3.2)* for the deflickering task. (ii) We design a *neural filtering (Section 3.3)* strategy for the neural atlas as the single atlas is inevitably flawed.

3.2. Flawed Atlas Generation

Motivation. A good blind deflickering model should have the capacity to track correspondences across all the video frames. Most architectures in video processing can only take a small number of frames as input, resulting in a limited receptive field that is insufficient for long-term consistency. To this end, we introduce neural atlases [26] to the deflickering task as we observe it perfectly fits this task. A neural atlas is a unified and concise representation of all the pixels in a video. As shown in Figure 3(a), let $p = (x, y, t) \in \mathbb{R}^3$ be a pixel that locates at (x, y) in frame I_t. Each pixel p is fed into a mapping network $\mathbb{M}$ that predicts a 2D coordinate (u^p, v^p) representing the corresponding position of the pixel in the atlas. Ideally, the correspondences in different frames should share a pixel in the atlas, even if the color of the input pixel is different. That is to say, temporal consistency is ensured.

Training. Figure 3(a) shows the pipeline to generate the atlas. For each pixel p, we have:

$$(u^p, v^p) = \mathbb{M}(p), \tag{1}$$

$$c^p = \mathbb{A}(\phi(u^p), \phi(v^p)). \tag{2}$$

The 2D coordinate (u^p, v^p) is fed into an atlas network $\mathbb{A}$ with positional encoding function $\phi(\cdot)$ to estimate the RGB color c^p for the pixel. The mapping network $\mathbb{M}$ and the atlas network $\mathbb{A}$ are trained jointly by optimizing the loss between the original RGB color and the predicted color c^p. Besides the reconstruction term, a consistency loss is also employed to encourage the corresponding pixels in different frames to be mapped to the same atlas position. We follow the implementation of loss functions in [26]. After training the networks $\mathbb{M}$ and $\mathbb{A}$, we can reconstruct the videos $\{A_t\}_{t=1}^{T}$ by providing all the coordinates of pixels in the

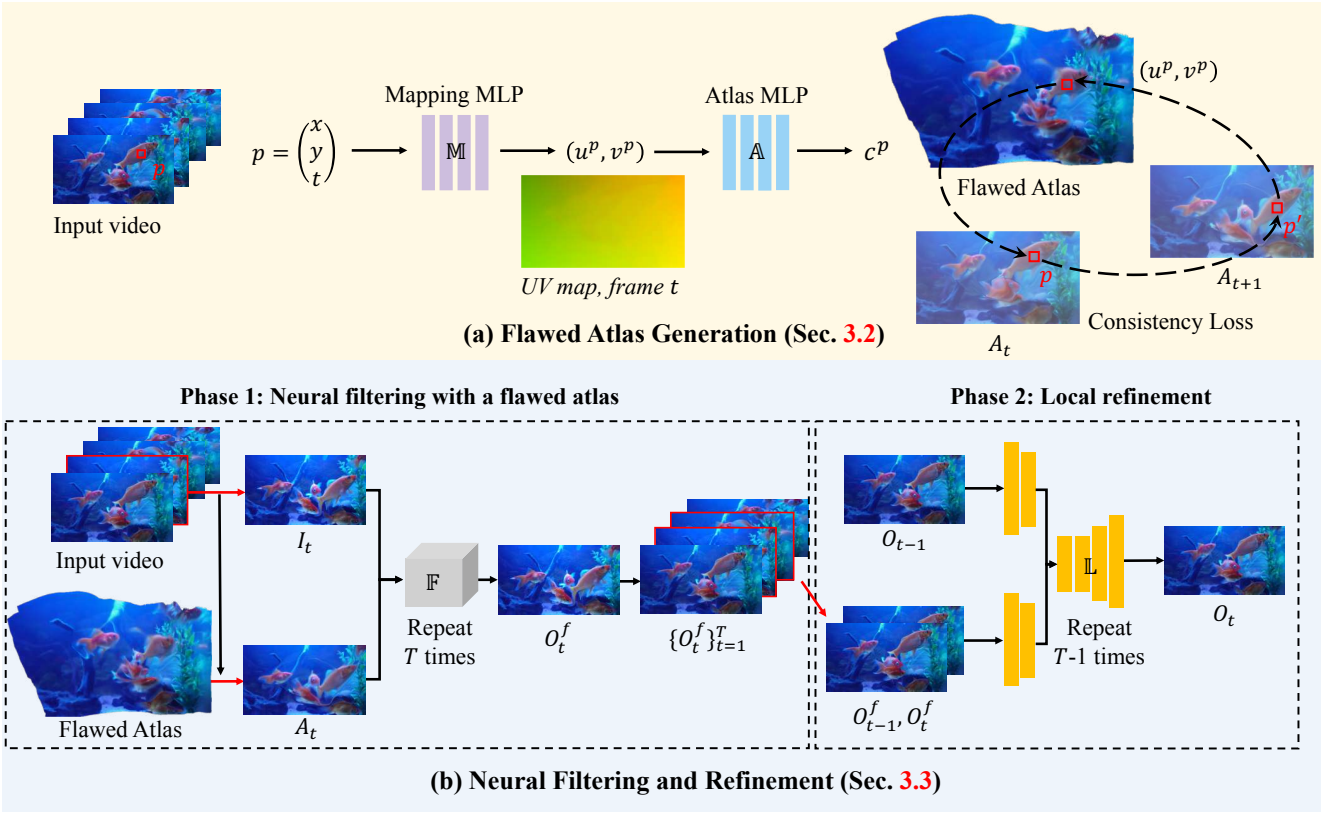

Figure 3. **The framework of our approach**. We first generate an atlas as a unified representation of the whole video, providing consistent guidance for deflickering. Since the atlas is flawed, we then propose a neural filtering strategy to filter the flaws.

whole video. The reconstructed atlas-based video $\{A_t\}_{t=1}^{T}$ is temporally consistent as all the pixels are mapped from a single atlas. In this work, we utilize the temporal consistency property of this video for blind deflickering.

Considering the trade-off between performance and efficiency, we only use a single-layer atlas to represent the whole video, although using two layers (background layer and foreground layer) or multiple layers of atlases might slightly improve the performance. First, in practice, we notice the number of layers is quite different, which varies from a single layer to multiple layers (more than two), making it challenging to apply them to diverse videos automatically. Besides, we notice that artifacts and distortion are inevitable for many scenes, as discussed in [26]. We present how to handle the artifacts in the flawed atlas with the following network designs in Secion 3.3.

3.3. Neural Filtering and Refinement

Motivation. The neural atlas contains not only the treasure but also the trash. In Section 3.2, we argue that an atlas is a strong cue for blind deflickering since it can provide consistent guidance across the whole video. However, the reconstructed frames from the atlas are flawed. First, as analyzed in NLA [26], it cannot perform well when the object is moving rapidly, or multiple layers might be required for multiple objects. For blind deflickering, we need to remove the flicker and avoid introducing new artifacts. Secondly, the optical flow obtained by the flickering video is not accurate, which leads to more flaws in the atlas. At last, there are still some natural changes, such as shadow changes, and we hope to preserve the patterns.

Hence, we design a *neural filtering* strategy that utilizes the promising temporal consistency property of the atlas-based video and prevents the artifacts from destroying our output videos.

Training Strategy. Figure 3(b) shows the framework to use the atlas. Given an input video $\{I_t\}_{t=1}^{T}$ and an atlas A, in every iteration, we get one frame I_t from the video and input it to the filter network $\mathbb{F}$:

$$O_t^f = \mathbb{F}(I_t, A_t), \tag{3}$$

where A_t is obtained by fetching pixels from the shared atlas A with the coordinates of I_t.

We design a dedicated training strategy for the flawed atlas, as shown in Figure 4. We train the network using only a single image X instead of consecutive frames. In the training time, we apply a transformation $\tau_a(\cdot)$ to distort the appearance, including color, saturation, and brightness of the

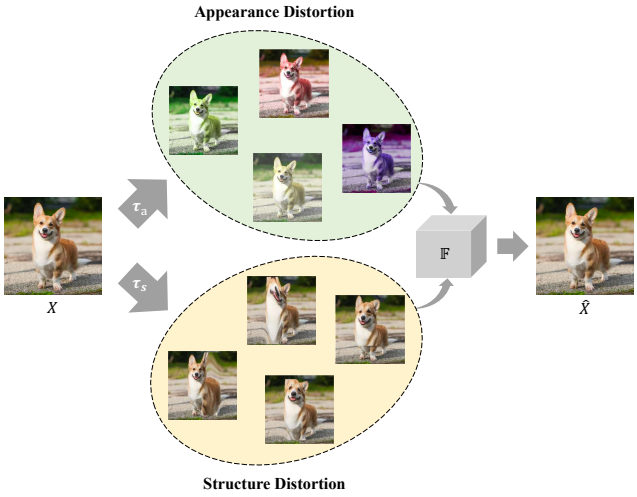

Figure 4. **Training pipeline of our neural filtering strategy.** We apply two transformations to a clean image X for mimicking the flickering input frame and the flawed atlas frame.

image, which mimics the flickering pattern in I_t. We apply another transformation $\tau_s(\cdot)$ to distort the structures of the image, mimicking the distortion of a flawed atlas-based frame in A_t. At last, the network is trained by minimizing the L2 loss function $\mathcal{L}$ between prediction $\mathbb{F}(\tau_a(X), \tau_s(X))$ and the clean ground truth X (i.e., the image before augmentation):

$$\mathcal{L} = ||\mathbb{F}(\tau_a(X), \tau_s(X); \theta_{\mathbb{F}}) - X||_2^2, \tag{4}$$

where $\theta_{\mathbb{F}}$ is the parameters of filtering network $\mathbb{F}$.

The network tends to learn the invariant part from two distorted views respectively. Specifically, $\mathbb{F}$ learns the structures from the input frame I_t and the appearance (e.g., brightness, color) from the atlas frame A_t as they are invariant to the structure distortion τ_s. At the same time, the distortion of $\tau_s(X)$ would not be passed through the network $\mathbb{F}$. With this strategy, we achieve the goal of neural filtering with the flawed atlas.

Note that while this network $\mathbb{F}$ only receives one frame, long-term consistency can be enforced since the temporal information is encoded in the atlas-based frame A_t.

Local refinement. The video frames $\{O_t^f\}_{t=1}^T$ are consistent with each other globally in both the short term and the long term. However, it might contain local flicker due to the misalignment between input and atlas-based frames. Hence, we use an extra local deflickering network to refine the results further. Prior work has shown that local flicker can be well addressed by a flow-based regularization. We thus choose a lightweight pipeline [27] with modification. As shown in Figure 3(b), we predict the output frame O_t by providing two consecutive frames O_t^f, O_{t-1}^f and previous output O_{t-1} to our local refinement network $\mathbb{L}$. Two consecutive frames are firstly followed by a few convolution layers and then fused with the O_{t-1}. The local flickering network is trained with a simple temporal consistency loss $\mathcal{L}_{local}$ to remove local flickering artifacts:

$$\mathcal{L}_{local}(O_t, O_{t-1}) = ||M_{t,t-1} \odot (O_t - \hat{O}_{t-1})||_1, \tag{5}$$

where $\hat{O}_{t-1}$ is obtained by warping the O_{t-1} with the optical flow from frame t to frame $t-1$. $M_{t,t-1}$ is the corresponding occlusion mask. For the frames without local artifacts, the output should O_t be the same as O_t^f. Hence, we also provide a reconstruction loss by minimizing the distance between O_t and O_t^f to regularize the quality.

Implementation details. The network $\mathbb{F}$ is trained on the MS-COCO dataset [35] as we only need images for training. We train it for 20 epochs with a batch size of 8. For the network $\mathbb{L}$, we train it on the processed DAVIS dataset [27, 43] for 50 epochs with a batch size of 8. We adopt the Adam optimizer and set the learning rate to 0.0001.

4. Blind Deflickering Dataset

We construct the first publicly available dataset for blind deflickering.

Real-world data. We first collect real-world videos that contain various types of flickering artifacts. Specifically, we collect five types of real-world flickering videos:

- *Old movies* contain complicated flickering patterns. The flicker is caused by multiple reasons, such as unstable exposure time and aging of film materials. Hence, the flickering can be high-frequency or low-frequency, globally and locally.
- *Old cartoons* are similar to old movies, but the structures differ from natural videos.
- *Time-lapse* videos capture a scene for a long time, and the environment illumination usually changes a lot.
- *Slow-motion* videos can capture high-frequency changes in lighting.
- *Processed* videos denote the videos processed by various processing algorithms. The patterns of flicker are decided by the specific algorithm. We follow the setting in [27].

Synthetic data. While real-world videos are good for evaluating perceptual performance, they do not have ground truth for quantitative evaluation. Hence, we create a synthetic dataset that provides ground truth for quantitative analysis. Let $\{G_t\}_{t=1}^T$ be the clean video frames, the flickered video $\{G_t\}_{t=1}^T$ can be obtained by adding the flickering artifacts F_t for each frame at time t:

$$I_t = G_t + F_t, \tag{6}$$

where $\{F_t\}_{t=1}^T$ is the synthesized flickering artifacts. For the temporal dimension, we synthesize both short-term and

Video type	$E_{warp} \downarrow$			PSNR ↑	
	Raw video	ConvLSTM	Ours	ConvLSTM	Ours
Synthetic	0.163	0.148	**0.088**	21.84	**26.46**
– $W = 1$	0.199	0.168	**0.091**	19.84	**27.75**
– $W = 3$	0.158	0.151	**0.086**	21.71	**26.40**
– $W = 10$	0.132	0.124	**0.086**	23.98	**25.21**
Processed	0.128	0.118	**0.094**	–	–

(a) Quantitative comparison

Video type	Preference rate	
	ConvLSTM	Ours
Old movies	38.0%	**62.0%**
Old cartoons	33.6%	**66.4%**
Time-lapse	31.9%	**68.1%**
Slow-motion	29.0%	**71.0%**
Average	33.7%	**66.3%**

(b) User study

Table 1. **Comparison to baselines.** We provide the (a) quantitative comparison for processed and synthetic videos since we have the high-quality optical flow for computing the evaluation metric. The warping errors of our approach are much smaller than the baseline, and our results are more similar to the ground truth on synthetic data, according to the PSNR. For the other real-world videos that cannot provide high-quality optical flow, we provide the (b) user study results for comparison. Our results are preferred by most users.

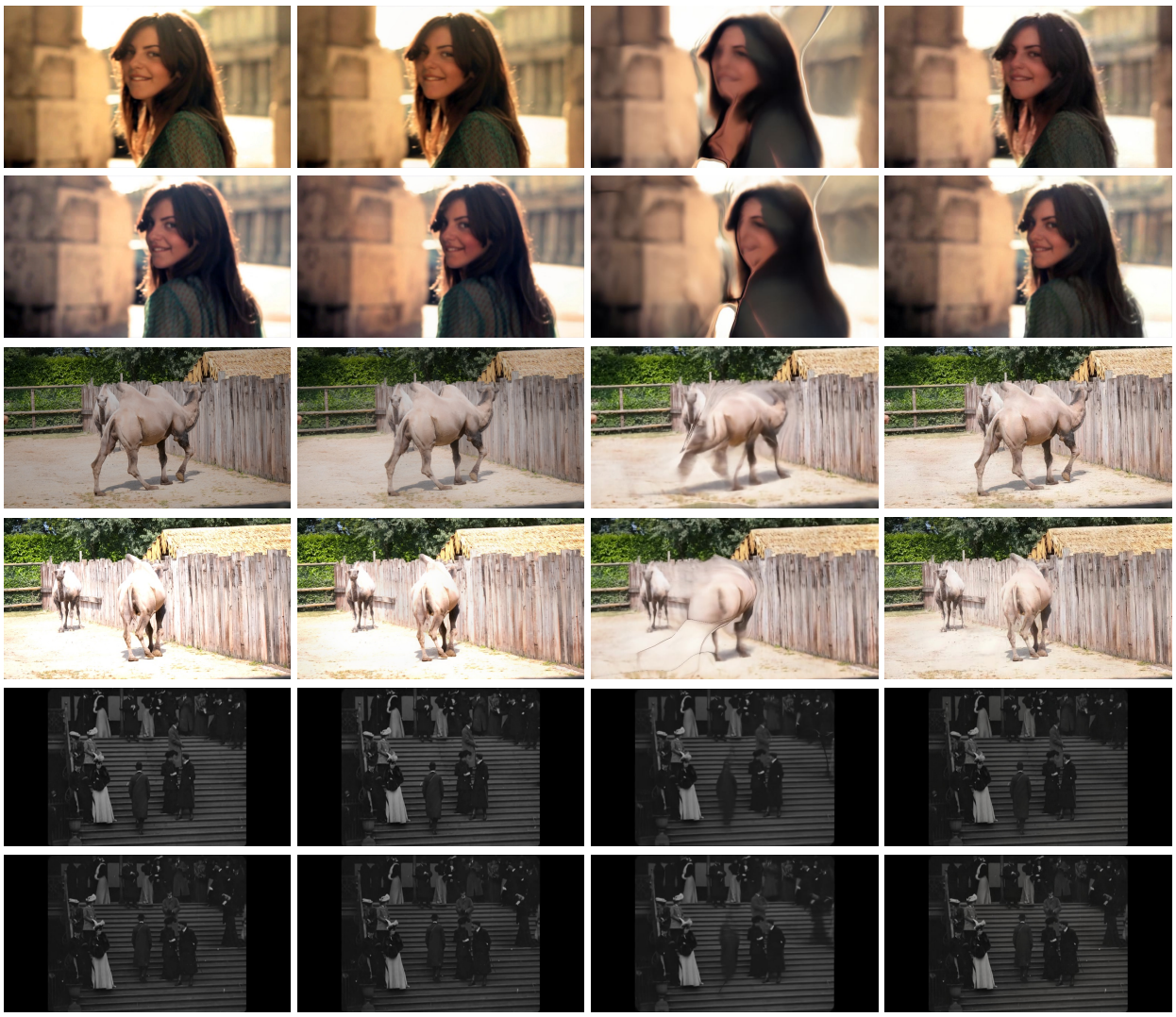

Figure 5. **Qualitative comparisons to baselines.** Our results outperforms the baseline ConvLSTM significantly on various flickering videos. We highly encourage readers to see videos in our project website.

Task	$E_{warp} \downarrow$				
	Processed	Bonneel et al. [7]	Lai et al. [27]	DVP [31]	Ours
w/o Extra Consistent Guidance	–	✗	✗	✗	✓
Dehazing [18]	0.120	0.128	0.136	0.109	**0.106**
Spatial White Balancing [20]	0.087	0.081	0.098	0.073	**0.062**
Colorization [60]	0.109	0.096	0.100	0.097	**0.084**
Enhancement [17]	0.125	0.105	0.115	0.102	**0.093**
CycleGAN [62]	0.124	0.113	0.117	0.103	**0.099**
Intrinsic Decomposition	0.131	0.085	0.108	0.071	**0.069**
Style Transfer	0.202	0.161	0.177	0.143	**0.142**
Average Score	0.128	0.110	0.122	0.100	**0.094**

Table 2. **Qualitative comparison with blind video temporal consistency methods that use input videos as extra guidance.** While our approach does not use input videos as guidance, our method achieves better numerical performance compared with the baselines.

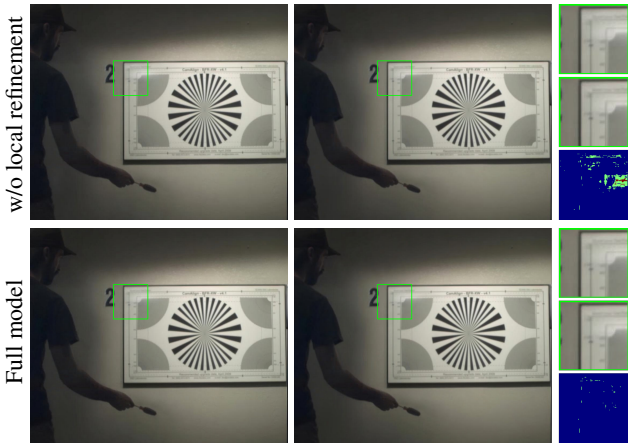

Figure 6. **Ablation study for local refinement module.** The local refinement network is vital to remove the local flickering. Cropped images and their difference maps are placed at the third column.

long-term flicker. Specifically, we set a window size W, which denotes the number of frames that share the same flickering artifacts. We set the window size W as 1, 3, 10 respectively.

Summary. We provide 20, 10, 10, 10, 157, and 90 for old movies, old cartoons, slow-motion videos, time-lapse videos, processed videos, and synthetic videos.

5. Experiments

5.1. Evaluation Setup

Dataset. We mainly use our constructed Blind Deflickering Dataset for evaluation. Details are presented in Section 4.

Evaluation metric. We measure the temporal inconsistency based on the warping error used in DVP [31] that considers both short-term and long-term warping errors for quantitative evaluation. Given a pair of frames O_t and O_s, the warping error E_{pair} can be calculated by:

$$E_{pair}(O_t, O_s) = ||M_{t,s} \odot (O_t - \hat{O}_s)||_1, \quad (7)$$

$$E^t_{warp} = E_{pair}(O_t, O_{t-1}) + E_{pair}(O_t, O_1), \quad (8)$$

where $\hat{O}_s$ is obtained by warping the O_s with the optical flow from frame t to frame s. $M_{t,s}$ is the occlusion mask from frame t and frame s. For each frame t, the warping error E^t_{warp} is computed with the previous and first frames in the video. We compute the warping error for all the frames in a video.

5.2. Comparisons to Baselines

Baseline. As our approach is the first method for blind deflickering, no existing public method can be used for comparison. Thus, we design a baseline inspired by blind video temporal consistency approaches: *ConvLSTM*, modified from Lai et al. [27]. Specifically, we replace the consistent input pair of frames with flickered pair frames, and we retrain the ConvLSTM on their training dataset [27].

Results. Table 1(a) provides quantitative results on two types of videos where we can estimate high-quality optical flow from consistent videos for computing the warping errors. Our results are consistently better than the baseline. In Table 1(b), we conduct a user study on Amazon Mechanical Turk following an A/B test protocol [11] to evaluate the perceptual preference between the main baseline ConvLSTM and our method. Each user needs to choose a video with better perceptual quality from videos processed by our method and the baseline. We use all real-world videos that cannot obtain high-quality optical flow for quantitative evaluation. In total, we have 20 users and 50 pairs of comparisons. Our method outperforms the baseline significantly in all tasks. Figure 5 shows the qualitative comparisons between our approach and the baseline. Our approach removes various types of flicker, and our results are more temporal consistent than baselines.

Comparisons to blind temporal consistency methods.

Method	$E_{warp} \downarrow$
Ours without atlas & neural filtering	0.131
Ours without local refinement	0.090
Ours full model	**0.088**

Table 3. **Quantitative results of ablation study**. The atlas and neural filtering strategy reduce temporal inconsistency significantly. The local refinement makes a slight difference in warping error but does improve perceptual performance.

The comparison between our approach and blind video temporal consistency methods is unfair since our approach *does not* require extra videos and baselines *use* extra input videos as guidance. However, we still provide the comparison results for reference. Specifically, we use three state-of-the-art baselines, including Bonneel et al. [7], Lai et al. [27] and DVP [31]. Table 2 shows the quantitative results between our approach and baselines. The warping error of our approach on various processed videos is lower than all the baselines, which indicates that our approach is more temporal consistent even without the input video guidance.

5.3. Ablation Study

We present the flawed atlas in the third column of Figure 5. While the temporal consistency between frames is perfect, the atlas-based frames contain many artifacts and distortions. Hence, designing dedicated strategies is essential for blind deflickering. In this part, we analyze the importance of our two modules, respectively: (1) *neural filtering* denotes the atlas generation and neural filtering since they cannot be split. (2) *local refinement* denotes the local refinement module.

Neural filtering. We first analyze the importance of neural filtering by removing this part. We direct apply our local refinement module on the flickering input videos. As shown in Table 3, removing the neural filtering module significantly increases temporal inconsistency.

Local refinement. As shown in Table 3, removing the local refinement network degraded the quantitative performance slightly. In Figure 6, we show some local regions in the frames are temporal inconsistent. While these regions are small and the warping errors are only slightly reduced, this local flickering does hurt the perceptual performance, as shown in our supplementary materials.

5.4. Comparisons to Human Experts

We compare our approach to human experts that use commercial software for deflickering. We adopt the RE:Vision DE:Flicker commercial software [47] for comparison, following Bonneel et al. [7]. Specifically, we obtain the official demos processed by experts and compare them. Figure 7 shows the qualitative comparison results. We can see that our approach can obtain competitive results in a fully-automatic manner. Besides, as discussed in Bonneel et al. [7], the videos processed by new users are usually low-quality.

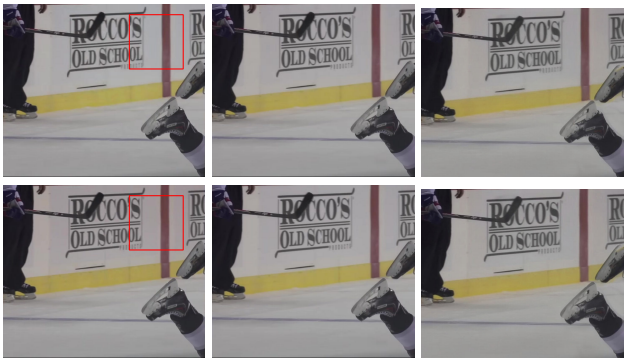

Figure 7. **Comparison to human experts**. Input: the lower frame is redder than the upper one. Our method achieves comparable performance with human experts in this case.

5.5. Discussion and Future Work

Potential applications. Our model can be applied to all evaluated types of flickering videos. Besides, while our approach is designed for videos, it is possible to apply *Blind Deflickering* for other tasks (e.g., novel view synthesis [40, 55]) where flickering artifacts exist.

Temporal consistency beyond our scope. Solving the temporal inconsistency of video content is beyond the scope of deflickering. For example, the contents obtained by video generation algorithms can be very different. Large scratches in old films can destroy the contents and result in unstable videos, which require extra restoration technique [53]. We leave the study for removing these temporally inconsistent artifacts to the future work.

6. Conclusion

In this paper, we define a problem named *blind deflickering* that can remove diverse flickering artifacts without knowing the specific flickering type and extra guidance. We propose the first dedicated approach for this task. The core of our approach is to adopt a neural atlas with a neural filtering strategy. The neural atlas concisely extracts all pixels in the videos and provides strong guidance to enforce long-term consistency, but it is flawed in many regions. We then use a neural network to filter the flaws of the atlas for satisfying performance. We conduct extensive experiments to evaluate the deflickering performance. Results show that our approach outperforms baselines significantly on different datasets, and controlled experiments validate the effectiveness of our key designs.

Acknowledgement

This work was supported by the InnoHK program.

References

[1] Aseem Agarwala, Ke Colin Zheng, Chris Pal, Maneesh Agrawala, Michael F. Cohen, Brian Curless, David Salesin, and Richard Szeliski. Panoramic video textures. *ACM Trans. Graph.*, 2005. 3

[2] Muhammad Kashif Ali, Dongjin Kim, and Tae Hyun Kim. Learning task agnostic temporal consistency correction. *CoRR*, abs/2206.03753, 2022. 3

[3] Digital Anarchy. Flicker free. 3

[4] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *ECCV*, 2022. 3

[5] Connelly Barnes, Dan B. Goldman, Eli Shechtman, and Adam Finkelstein. Video tapestries with continuous temporal zoom. *ACM Trans. Graph.*, 2010. 3

[6] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *ACM Trans. Graph.*, 33(4):159:1–159:12, 2014. 2

[7] Nicolas Bonneel, James Tompkin, Kalyan Sunkavalli, Deqing Sun, Sylvain Paris, and Hanspeter Pfister. Blind video temporal consistency. *ACM Trans. Graph.*, 34(6):196:1–196:9, 2015. 2, 7, 8

[8] Matthew Brown and David G. Lowe. Automatic panoramic image stitching using invariant features. *IJCV*, 2007. 3

[9] Chen Chen, Qifeng Chen, Minh N Do, and Vladlen Koltun. Seeing motion in the dark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3185–3194, 2019. 1

[10] Hao Chen, Bo He, Hanyu Wang, Yixuan Ren, Ser-Nam Lim, and Abhinav Shrivastava. Nerv: Neural representations for videos. In *NeurIPS*, 2021. 3

[11] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017. 7

[12] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *CVPR*, 2021. 3

[13] Mengyu Chu, You Xie, Laura Leal-Taixé, and Nils Thuerey. Temporally coherent gans for video super-resolution (tecogan). *arXiv preprint arXiv:1811.09393*, 1(2):3, 2018. 1

[14] Michele Claus and Jan van Gemert. Videnn: Deep blind video denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 1

[15] Carlos D. Correa and Kwan-Liu Ma. Dynamic video narratives. *ACM Trans. Graph.*, 2010. 3

[16] Julie Delon and Agnes Desolneux. Stabilization of flicker-like effects in image sequences through local contrast correction. *SIAM Journal on Imaging Sciences*, 3(4):703–734, 2010. 1, 2

[17] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Trans. Graph.*, 36(4):118:1–118:12, 2017. 7

[18] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(12):2341–2353, 2011. 2, 7

[19] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 3

[20] Eugene Hsu, Tom Mertens, Sylvain Paris, Shai Avidan, and Frédo Durand. Light mixture estimation for spatially varying white balance. *ACM Trans. Graph.*, 27(3):70, 2008. 7

[21] Hanzhe Hu, Yinbo Chen, Jiarui Xu, Shubhankar Borse, Hong Cai, Fatih Porikli, and Xiaolong Wang. Learning implicit feature alignment function for semantic segmentation. In *ECCV*, 2022. 3

[22] Michal Irani and P. Anandan. Video indexing based on mosaic representations. *Proc. IEEE*, 1998. 3

[23] Michal Irani, P. Anandan, and Steven C. Hsu. Mosaic based representations of video sequences and their applications. In *ICCV*, 1995. 3

[24] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 2

[25] Ali Kanj, Hugues Talbot, and Raoul Rodriguez Luparello. Flicker removal and superpixel-based motion tracking for high speed videos. In *2017 IEEE international conference on image processing (ICIP)*, pages 245–249. IEEE, 2017. 1, 2

[26] Yoni Kasten, Dolev Ofri, Oliver Wang, and Tali Dekel. Layered neural atlases for consistent video editing. *ACM Transactions on Graphics (TOG)*, 40(6):1–12, 2021. 2, 3, 4

[27] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *ECCV*, 2018. 2, 5, 7, 8

[28] Manuel Lang, Oliver Wang, Tunc Aydin, Aljoscha Smolic, and Markus Gross. Practical temporal consistency for image-based graphics applications. *ACM Trans. Graph.*, 31(4):34:1–34:8, 2012. 2

[29] Chenyang Lei and Qifeng Chen. Fully automatic video colorization with self-regularization and diversity. In *CVPR*, 2019. 1

[30] Chenyang Lei, Xuhua Huang, Mengdi Zhang, Qiong Yan, Wenxiu Sun, and Qifeng Chen. Polarized reflection removal with perfect alignment in the wild. In *CVPR*, 2020. 2

[31] Chenyang Lei, Yazhou Xing, and Qifeng Chen. Blind video temporal consistency via deep video prior. In *NeurIPS*, 2020. 2, 7, 8

[32] Chenyang Lei, Yazhou Xing, Hao Ouyang, and Qifeng Chen. Deep video prior for video consistency and propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[33] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. In *NeurIPS*, 2017. 1, 2

[34] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *CVPR*, 2021. 3

[35] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014. 5

[36] Feng-Lin Liu, Shu-Yu Chen, Yu-Kun Lai, Chunpeng Li, Yue-Ren Jiang, Hongbo Fu, and Lin Gao. Deepfacevideoediting: sketch-based deep editing of face videos. *ACM Trans. Graph.*, 2022. 3

[37] Sebastian Loeschcke, Serge J. Belongie, and Sagie Benaim. Text-driven stylization of video objects. *CoRR*, abs/2206.12396, 2022. 3

[38] Nicolas Märki, Federico Perazzi, Oliver Wang, and Alexander Sorkine-Hornung. Bilateral space video segmentation. In *CVPR*, 2016. 1

[39] Lars M. Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *CVPR*, 2019. 3

[40] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 3, 8

[41] Hao Ouyang, Zifan Shi, Chenyang Lei, Ka Lung Law, and Qifeng Chen. Neural camera simulators. In *CVPR*, 2021. 2

[42] Jeong Joon Park, Peter Florence, Julian Straub, Richard A. Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019. 3

[43] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus H. Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016. 5

[44] Alex Rav-Acha, Pushmeet Kohli, Carsten Rother, and Andrew W. Fitzgibbon. Unwrap mosaics: a new representation for video editing. *ACM Trans. Graph.*, 2008. 3

[45] Xuanchi Ren, Zian Qian, and Qifeng Chen. Video deblurring by fitting to test data. *CoRR*, abs/2012.05228, 2020. 1

[46] Xuanchi Ren and Xiaolong Wang. Look outside the room: Synthesizing A consistent long-term 3d scene video from A single image. In *CVPR*, 2022. 1, 2

[47] RE:VISION. De:flicker. 3, 8

[48] Roger Blanco Ribera, Sungwoo Choi, Younghui Kim, Jungjin Lee, and Junyong Noh. Video panorama for 2d to 3d conversion. *Comput. Graph. Forum*, 2012. 3

[49] Lars Schnyder, Oliver Wang, and Aljoscha Smolic. 2d to 3d conversion of sports content using panoramas. In *ICIP*, 2011. 3

[50] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 1, 2, 3

[51] Vincent Sitzmann, Julien N. P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *NeurIPS*, 2020. 3

[52] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *NeurIPS*, 2020. 3

[53] Ziyu Wan, Bo Zhang, Dongdong Chen, and Jing Liao. Bringing old films back to life. In *CVPR*, 2022. 8

[54] Jiaxin Xie, Chenyang Lei, Zhuwen Li, Li Erran Li, and Qifeng Chen. Video depth estimation by fusing flow-to-depth proposals. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10100–10107. IEEE, 2020. 1

[55] Jiaxin Xie, Hao Ouyang, Jingtan Piao, Chenyang Lei, and Qifeng Chen. High-fidelity 3d gan inversion by pseudo-multi-view optimization. *arXiv preprint arXiv:2211.15662*, 2022. 8

[56] Yiran Xu, Badour AlBahar, and Jia-Bin Huang. Temporally consistent semantic video editing. In *ECCV*, 2022. 2

[57] Chun-Han Yao, Chia-Yang Chang, and Shao-Yi Chien. Occlusion-aware video temporal consistency. In *ACM Multimedia*, 2017. 2

[58] Vickie Ye, Zhengqi Li, Richard Tucker, Angjoo Kanazawa, and Noah Snavely. Deformable sprites for unsupervised video decomposition. In *CVPR*, 2022. 3

[59] Bo Zhang, Mingming He, Jing Liao, Pedro V Sander, Lu Yuan, Amine Bermak, and Dong Chen. Deep exemplar-based video colorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10100–10107. IEEE, 2019. 1

[60] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016. 1, 2, 7

[61] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 1, 2, 3

[62] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *ICCV*, 2017. 2, 7