

Light Flickering Guided Reflection Removal


Yuchen Hong[†], Yakun Chang[†], Jinxiu Liang, Lei Ma, Tiejun Huang, Boxin Shi 

Received: date / Accepted: date

Abstract When photographing through a piece of glass, reflections usually degrade the quality of captured images or videos. In this paper, by exploiting periodically varying light flickering, we investigate the problem of removing strong reflections from contaminated image sequences or videos with a unified capturing setup. We propose a learning-based method that utilizes short-term and long-term observations of mixture videos to exploit one-side contextual clues in *fluctuant* components and brightness-consistent clues in *consistent* components for achieving layer separation and flickering removal, respectively. A dataset containing synthetic and real mixture videos with light flickering is built for network training and testing. The effectiveness of the proposed method is demonstrated by the comprehensive evaluation on synthetic and real data, the application for video flickering removal, and the exploratory experiment on high-speed scenes.

Keywords Reflection removal · Layer separation · Image restoration · Flickering removal

[†] Equal contribution.

 Boxin Shi (Corresponding author)
E-mail: shiboxin@pku.edu.cn

Yuchen Hong · Jinxiu Liang · Tiejun Huang · Boxin Shi
National Key Laboratory for Multimedia Information Processing and
National Engineering Research Center of Visual Technology, School of
Computer Science, Peking University, China.

Yakun Chang
Institute of Information Science, Beijing Jiaotong University, China.
Beijing Key Laboratory of Advanced Information Science and Network
Technology, China.

Lei Ma
National Biomedical Imaging Center, College of Future Technology,
Peking University, China.
Beijing Academy of Artificial Intelligence, China.

1 Introduction

When photographing through a piece of glass (*e.g.*, a glass window or a showcase), the captured images or videos are often contaminated by reflections. Reflection removal aims at removing undesired reflection layers and recovering clear transmission layers from contaminated mixture images or videos, which is one of the fundamental problems in computer vision and computational photography. By assuming that reflections are out of focus and appear with much weaker edges than transmission layers (Fan et al, 2017; Wan et al, 2018b; Zhang et al, 2018b; Wan et al, 2019), single-image reflection removal methods (*e.g.*, Li et al (2020a) and Dong et al (2021), denoted as IBCLN and DX21¹, respectively) are popular choices due to the convenient capturing setup. However, reflections with clear edges could sometimes dominate image contents, which makes the two layers indistinguishable if depending solely on the edges. Therefore, it is a natural choice to introduce auxiliary information for such challenging cases to tell the reflection and transmission layers apart.

Due to additional constraints involved, reflection removal methods which use multiple images generally provide more stable solutions. A typical category utilizes images captured with active light sources like the flash (Chang et al, 2020; Lei and Chen, 2021) which provide distinctive clues about transmission layers to facilitate reflection removal. However, these methods are primarily designed for static scenes and rely on a stationary camera setup, rendering them unsuitable for more general scenes which contain dynamic contents caused by motions. Moreover, constrained by the illumination power of active light sources, these methods often encounter difficulties when dealing with distant transmission scenes (*e.g.*, capturing outdoor buildings from indoors through windows),

¹ In this paper, when a method is not explicitly named, we adopt the convention of using “the initials of the surnames of the first two authors + year” as synonyms of it.

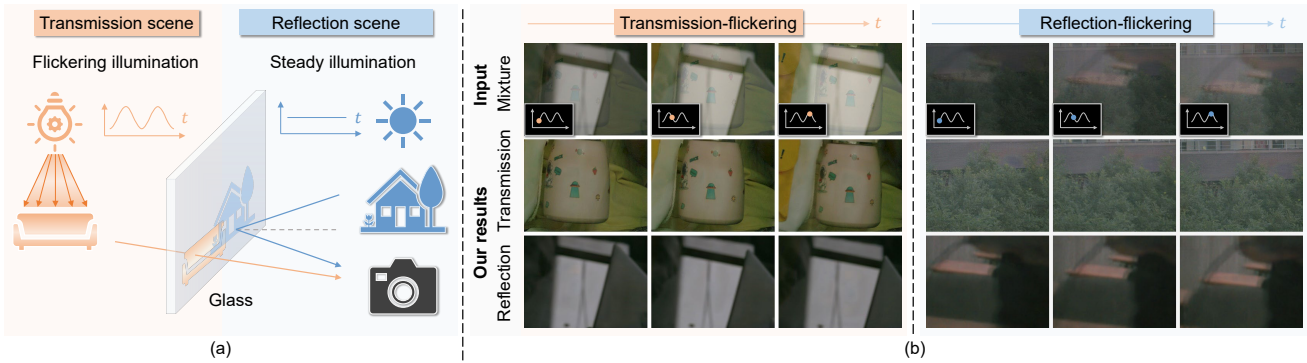


Fig. 1 (a) Illustration of the image formation process when one side of the glass (take the transmission scene as an example) is illuminated by a light source with light flickering. (b) Examples of mixture videos with light flickering at different sides of the glass and our layer separation results. For each group of data, we show three frames with different brightness from the mixture video in sequential order (with a brightness variation curve at the left lower of each frame that qualitatively points out the current brightness) to represent the video, and we show the corresponding layer separation results at the bottom.

which indicates their limitation of applying varying illuminations only for the transmission layer. Consequently, it is imperative to explore an alternative way that can exploit illumination variations at an arbitrary side of the glass to provide contextual clues for reflection removal in more general cases.

If we change the flash, a pulse-like illumination variation at one side of the glass, to a periodically continuous one, reflection removal could potentially be solved for general scenes containing dynamic contents. Fortunately, there exists a commonly overlooked yet widespread continuous illumination variation in daily life that can be leveraged for this purpose, and we denote it as *light flickering* in this paper. Specifically, existing residential light sources, such as incandescent, fluorescent, and LED lamps powered by the alternating current (AC) supply, as well as DC (battery)-powered LED flashlights controlled by pulse width modulation (PWM), are all able to produce high-frequency (e.g., 100 or 120 Hz) light flickering. Though human vision systems are insensitive to such high-frequency light flickering due to the persistence of vision (Wells et al, 2001), digital cameras can record this by capturing discrete frames with short exposures (e.g., < 10 ms). As shown in Fig. 1(a), when one side of the glass is illuminated by a flickering light source, the corresponding layer becomes flickering while the brightness of the other layer (with steady ambient illumination) is temporally consistent. This observation indicates that whether light flickering occurs in transmission or reflection scenes, disparities in the temporal brightness variation between the two layers can serve as an effective clue for reflection removal.

For a mixture video captured in a light flickering environment, it can be decomposed into a *fluctuant* and a *consistent* component. The fluctuant component refers to the video whose intensity is only influenced by the fluctuant part of the light source, thus it provides contextual information about the scene at the side with light flickering. The consistent compo-

nent indicates the stable counterpart, which is identical to the video captured in a steady illumination (removing the fluctuant part). It is intuitive that the fluctuant component which is only correlated to either the transmission or reflection layer can facilitate reflection removal, while directly extracting and exploiting it from a mixture video containing dynamic contents is challenging, since motions and light flickering jointly result in the frame intensity variation in such cases. Besides, recovering the brightness-consistent transmission and reflection layer under the guidance of the rapidly-varying fluctuant component is also an untouched problem that has to be settled.

To tackle the above issues, we analyze the image formation model with the interference of reflection contaminations and light flickering, and investigate the feasibility of leveraging light flickering for reflection removal by considering ideal static scenes. For real scenes containing dynamic contents, we propose a learning-based framework named **L**ight **f**lick**E**ring guide reflection removal **N**etwork (LIKE-Net) which utilizes short-term and long-term observations of mixture videos to exploit the one-side contextual clues in fluctuant components and brightness-consistent clues in consistent components for achieving layer separation and flickering removal, respectively. A dataset containing synthetic and real mixture videos with light flickering is collected for comprehensive validations of the proposed method. Fig. 1(b) shows examples from the dataset and our layer separation results. Besides, we further demonstrate the effectiveness of the proposed method by applying it to video flickering removal and exploring its reflection removal performance on high-speed scenes containing fast motions with a high-speed spiking camera (Huang et al, 2023). This paper contributes in the following aspects:

- introducing light flickering to achieve high-fidelity reflection removal for challenging real dynamic scenes;

- proposing a learning-based framework for light flickering guided reflection removal;
- demonstrating the applicability for video flickering removal and the capability for reflection removal in high-speed scenes with fast motions; and
- building a dataset containing synthetic and real videos with light flickering for validating current methods and inspiring future research.

2 Related work

2.1 Single-image reflection removal

Single-image reflection removal methods mainly rely on the assumption that transmission and reflection layers have different distributions, *e.g.*, edges of reflection layers are more likely to be blurred. Conventional mathematical models use priors of edges in their optimization algorithms, *e.g.*, the gradient sparsity prior (Levin and Weiss, 2007; Wan et al, 2016), relative smoothness (Li and Brown, 2014), ghosting cues (Shih et al, 2015), image content (Wan et al, 2018a), and penalty on the gradient of restored transmission layers (Yang et al, 2019). Fan et al (2017) propose to use deep neural networks for recovering transmission layers in an end-to-end manner. Subsequently, a series of learning-based strategies are proposed, *e.g.*, using concurrent or cooperative network structures (Wan et al, 2018b, 2019), employing generative adversarial network (Goodfellow et al, 2014) based models (Wei et al, 2019; Ma et al, 2019), and training with the perceptual loss (Zhang et al, 2018b). IBCLN (Li et al, 2020a) uses a cascaded refinement strategy to iteratively refine transmission layers. Dong et al (2021) regress reflection confidence maps and achieve reflection removal. Hong et al (2021) and Hong et al (2023b) relieve the content ambiguity by using panoramic images. Zhong et al (2024) introduce language descriptions to provide high-level semantic information for reflection removal. We refer readers to Wan et al (2023) for a comprehensive and up-to-date survey on single-image reflection removal.

2.2 Multi-image reflection removal

Multiple-image reflection removal methods usually leverage the auxiliary information introduced by additional images. Polarization-based methods distinguish reflection and transmission layers by using images captured through different angles of polarizers (Nayar et al, 1997; Schechner et al, 2000; Diamant and Schechner, 2008; Kong et al, 2012; Lyu et al, 2019; Lei et al, 2020; Lyu et al, 2023). However, manually rotating a polarizer or using special polarization cameras (Li et al, 2020b) are needed to obtain polarized images, which narrows the applicability of such methods.

Flash-based methods adopt active light sources (Chang et al, 2020; Lei and Chen, 2021; Hong et al, 2020, 2023a) to illuminate transmission scenes for obtaining reflection-free guidance. Sheinin et al (2017) propose to separate layers by using varying illuminations with a special coded imaging technique. Whereas, requirements on the spatial alignment of captured images prevent these methods from being applied to dynamic scenes. By utilizing multiple images (Li and Brown, 2013; Simon and Kyu Park, 2015; Liu et al, 2020) or videos (Nandoriya et al, 2017) captured from different viewpoints, motion-based methods exploit motion differences of transmission and reflection layers to achieve their separation, while these methods are sensitive to varying illuminations as the motion estimation may fail in such cases. In this paper, we leverage periodically-varying illuminations to achieve reflection removal for general dynamic scenes.

3 Problem formulation

3.1 Image formation model

For periodically varying illuminations such as AC-powered (usually 50 or 60 Hz) light sources or PWM-controlled flashlights, we can measure their characteristics of light flickering. Intensity profiles of two typical types of residential lamps (*i.e.*, fluorescent and LED) and a LED flashlight are illustrated in Fig. 2(a). It can be observed that intensities of light sources fluctuate with a fixed frequency (determined by the frequency of the AC power or PWM controller). As shown in Fig. 2(b), in the condition that camera shutters are asynchronous with light sources and the exposure time is unequal to integer multiples of the light flickering cycle, digital cameras can record light flickering in a video.

We first consider the formation model of a mixture video $\{I_i\}_{i=0}^{N-1}$, where N and i denote the total number of frames in the video and the frame index, respectively. When the transmission scene is captured through a piece of glass, each mixture frame I_i in the video $\{I_i\}_{i=0}^{N-1}$ is the combination of the transmission layer T_i and reflection layer R_i : $I_i = T_i + R_i$. We denote the videos composed of transmission layers and reflection layers as the transmission video $\{T_i\}_{i=0}^{N-1}$ and the reflection video $\{R_i\}_{i=0}^{N-1}$, respectively.

When light flickering at one side of the glass is introduced into the formation model, the brightness variation between frames needs to be considered. Without losing generality, we assume that the transmission scene is with light flickering and the reflection scene is under a steady illumination, thus $\{T_i\}_{i=0}^{N-1}$ can be formulated as the sum of a brightness-varying *fluctuant* component $\{T_i^{\sim}\}_{i=0}^{N-1}$ and a brightness-invariant *consistent* component $\{T_i^{-}\}_{i=0}^{N-1}$:

$$T_i = \int_{t_e} \Omega_T r_T(t) dt = T_i^{\sim} + T_i^{-}, \quad (1)$$

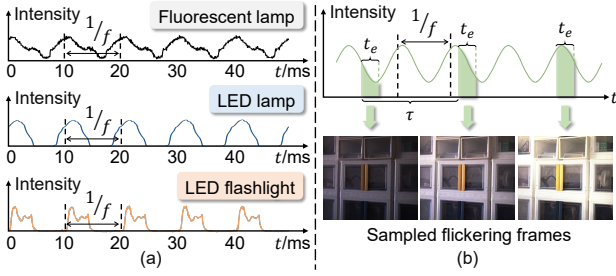


Fig. 2 (a) Intensity profiles of three examples of light flickering. We capture them by using a high-speed camera (Huang et al, 2023), with the frame rate of 20K FPS. (b) Illustration of how a camera records a video with light flickering, where f , t_e , and τ are the frequency of light flickering, the exposure time of the camera, and the interval between two adjacent frames, respectively.

where t_e denotes the exposure time of the camera, Ω_T denotes the refractive coefficient map (Lyu et al, 2019), and $r_T(t)$ denotes the received transient radiance of the transmission scene which varies with time t due to the flickering light sources. Correspondingly, the reflection video $\{R_i\}_{i=0}^{N-1}$ is only composed of a brightness-invariant consistent component $\{R_i^-\}_{i=0}^{N-1}$:

$$R_i = \int_{t_e} \Omega_R r_R dt = R_i^-, \quad (2)$$

where Ω_R denotes the reflective coefficient map (Lyu et al, 2019) and r_R denotes the received consistent radiance of the reflection scene due to the time-invariant illumination. Therefore, the captured mixture video $\{I_i\}_{i=0}^{N-1}$ is also composed of the fluctuant and consistent component:

$$I_i = T_i^{\sim} + T_i^- + R_i^- = I_i^{\sim} + I_i^-. \quad (3)$$

For the case that the transmission scenes is with light flickering, it is apparent that the fluctuant component $\{I_i^{\sim}\}_{i=0}^{N-1}$ is only related to the transmission layer (*i.e.*, $I_i^{\sim} = T_i^{\sim}$), whereas the consistent component $\{I_i^-\}_{i=0}^{N-1}$ is blended (*i.e.*, $I_i^- = T_i^- + R_i^-$). The extraction and exploitation of the fluctuant component in a mixture video plays a critical role for the following reflection removal task, since it can provide one-side contextual clues for telling the two layers apart. Conversely, the fluctuant component will be reflection-dominated if the reflection scene is with light flickering, and the image formation model in Eqn. (3) can be easily modified to fit this case:

$$I_i = R_i^{\sim} + R_i^- + T_i^- = I_i^{\sim} + I_i^-. \quad (4)$$

which also provides reflection-aware clues for reflection removal.

3.2 Feasibility analysis

To analyze the feasibility of employing light flickering for reflection removal and investigate the one-side contextual clues in fluctuant components, we use ideal static scenes for instance, which are free from inter-frame motions and correspond to the case that the camera is fixed on a tripod. When capturing such scenes, there does not exist any other factor except light flickering at one side of the glass that causes temporal variations of captured image intensities. We pick the case that the transmission scene is with light flickering for example. Following Eqn. (3), mixture videos $\{I_i\}_{i=0}^{N-1}$ can be divided into a brightness-varying fluctuant component $\{I_i^{\sim}\}_{i=0}^{N-1}$ and a brightness-invariant blended consistent component $\{I_i^-\}_{i=0}^{N-1}$, in which the fluctuant component is only concerned with transmission layers that containing light flickering.

Since the illuminance of flickering light sources changes periodically, the captured mixture video also varies periodically across video frames. To analyze the temporal periodical property caused by light flickering, we utilize Discrete Fourier Transform (DFT, denoted as \mathcal{F}) (Cooley and Tukey, 1965) to transform the time-domain mixture video $\{I_i\}_{i=0}^{N-1}$ into its frequency-domain counterpart $\{\hat{I}_k\}_{k=0}^{N-1}$:

$$\hat{I}_k = \mathcal{F}(\{I_i\}_{i=0}^{N-1}, k), \quad (5)$$

where k is the index in the frequency domain. According to Eqn. (3), the temporal periodical property of the mixture video is from the fluctuant component, thus in the frequency domain, the fluctuant component owns values at non-zero frequencies. We denote these frequencies as $\{k_f | \hat{I}_{k_f} > \epsilon\}$, where ϵ is a small value to prevent the impact of the noise among video frames. Conversely, the consistent component only owns value at the zero frequency since its brightness is time-invariant.

To extract the consistent and fluctuant component for observing and validating their reflection-correlated properties, we first utilize low-pass or band-pass filters with narrow bandwidths for filtering peaks in the frequency domain. As shown in Fig. 3(b), in the frequency domain, the fluctuant component $\{\hat{I}_k^{\sim}\}_{k=0}^{N-1}$ and the consistent component $\{\hat{I}_k^-\}_{k=0}^{N-1}$ can be extracted by applying band-pass filters centered at the non-zero frequencies $\{k_f\}$ and a low-pass filter at the zero frequency on $\{\hat{I}_k\}_{k=0}^{N-1}$, respectively. Specifically, for each band-pass filter centered at non-zero frequencies $\{k_f\}$, we define its frequency range as $[k_f - 0.5, k_f + 0.5]$, which only allows the frequency within the frequency range to pass through while blocking frequencies outside of this range by setting values to be zero. Similarly, the low-pass filter only allows the frequency lower than a cutoff frequency to pass through, and we set the cutoff frequency at $k = 0.5$, thus in fact filtering the zero frequency. Then through Inverse Discrete Fourier Transform (IDFT) (Cooley and Tukey, 1965),

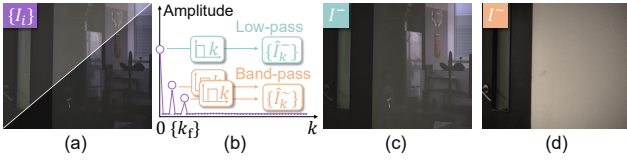


Fig. 3 (a) A mixture video recording a static scene with light flickering. We select two frames with maximum brightness discrepancy in the mixture video (due to light flickering) and show half of each frame at one side of the diagonal line respectively to represent the video. (b) Illustration of filtering the consistent and fluctuant component in the frequency domain. We use a low-pass filter and several band-pass filters to obtain the consistent component $\{\hat{f}_k^-\}_{k=0}^{N-1}$ and fluctuant component $\{\hat{f}_k^-\}_{k=0}^{N-1}$ in the frequency domain. (c) and (d) are the averaged consistent and fluctuant components, respectively, which are obtained by transforming the frequency-domain counterparts in (b) with IDFT and average operations.

the time-domain fluctuant component $\{I_i^\sim\}_{i=0}^{N-1}$ and the consistent component $\{I_i^-\}_{i=0}^{N-1}$ can be obtained.

As frames in the captured video for static scenes are spatially aligned and free from motions, thus for each pixel in the fluctuant component $\{I_i^\sim\}_{i=0}^{N-1}$ and the consistent component $\{I_i^-\}_{i=0}^{N-1}$, we calculate their temporal average and then two frames (denoted as I^\sim and I^-) are generated to represent the two components respectively, which are shown in Fig. 3(c) and (d). It can be observed that the averaged fluctuant frame I^\sim is free from reflection contaminations (since the transmission layer is with light flickering in this case), which validates the capability of the fluctuant component for providing one-side contextual clues, and further indicates the feasibility of introducing light flickering into reflection removal.

Unfortunately, though the above analysis shows the potential of leveraging light flickering for reflection removal, it is based on the ideal static scene assumption. In more general real scenes where mixture videos contain dynamic contents, separating and exploiting the fluctuant component through the simple Fourier transform becomes more challenging since motions and light flickering jointly result in intensity variation across frames. To address this challenge, we propose to leverage the modeling capabilities of data-driven deep learning methods. However, there is currently no existing dataset suitable for training and evaluating methods for light flickering guided reflection removal. Therefore, we create a new dataset that includes both synthetic and real data to fulfill this purpose.

3.3 Data preparation

As shown in Fig. 5, we create a dataset for light flickering guided reflection removal, which consists of both synthetic and real data and considers two distinct conditions as follows: (i) Transmission-flickering scenarios, where transmis-

sion scenes are with light flickering and reflection scenes are with temporally stable illuminations, and a typical example is capturing videos from outdoors to indoors; (ii) Reflection-flickering scenarios, as the conjugated case of (i), involve reflection scenes with light flickering and transmission scenes with temporally steady illuminations. An example of these cases is capturing videos from indoor to outdoor scenes through a piece of glass.

To facilitate the introduction of data preparation, for each mixture frame, we denote the image layer influenced by light flickering as the *flickering layer* (L^\sim) and the image layer displaying the side with temporally stable illuminations as the *constant layer* (L^-). We further denote a *deflickering layer* (L^\simeq) to represent the consistent component of the flickering layer (*i.e.*, removing the fluctuant component from the flickering layer). For example, in scenarios where transmission scenes are with light flickering (Eqn. (3)), the transmission layer T_i corresponds to the flickering layer, the reflection layer R_i corresponds to the constant layer, and the consistent component of the transmission layer T_i^- corresponds to the deflickering layer. Details of our dataset are as follows.

Synthetic data. Since it is challenging to obtain ground truths of mixture videos when light flickering and motions both exist, we opt to use synthetic data for network training and quantitative evaluation. We select a flash dataset (Aksoy et al., 2018) containing flash-only and no-flash image pairs to synthesize sequential flickering layers. The pipeline for our synthetic data is shown in Fig. 4. Since the flash dataset only contains flash-only and no-flash image pairs that are captured in static scenes, to simulate motions in dynamic scenes, at each step, we randomly generate a warping and motion operation for the image pairs:

$$\begin{aligned} I_i^a &= \mathcal{W}_i(I_{i-1}^a), \\ I_i^f &= \mathcal{W}_i(I_{i-1}^f), \end{aligned} \quad (6)$$

where I_i^a and I_i^f are the no-flash and flash-only images at timestamp i , $\mathcal{W}_i(\cdot)$ is the warping and motion operation for the current frame, where the warping function is realized by randomly generating a sequence of flow masks and motions are simulated by image cropping using a randomly generated trajectory. We then synthesize the flickering layer by $L_i^\sim = \alpha_i I_i^f + \beta I_i^a$, where α_i is a fluctuant coefficient applied to I_i^f at each step and β is a constant. For the fluctuant coefficient, we find that utilizing a sine function is efficient to represent the light flickering phenomenon:

$$\alpha_i = \sin(2\tau f i \pi + \epsilon) + 1, \quad (7)$$

where f and τ are the frequency of flickering light and the frame interval, respectively, and ϵ is a random value to simulate the starting time for capturing. In this work, f and τ have two settings: 100 Hz - 1/30 s and 120 Hz - 1/50 s, which will result in the flickering cycle being 3 frames and 5 frames,

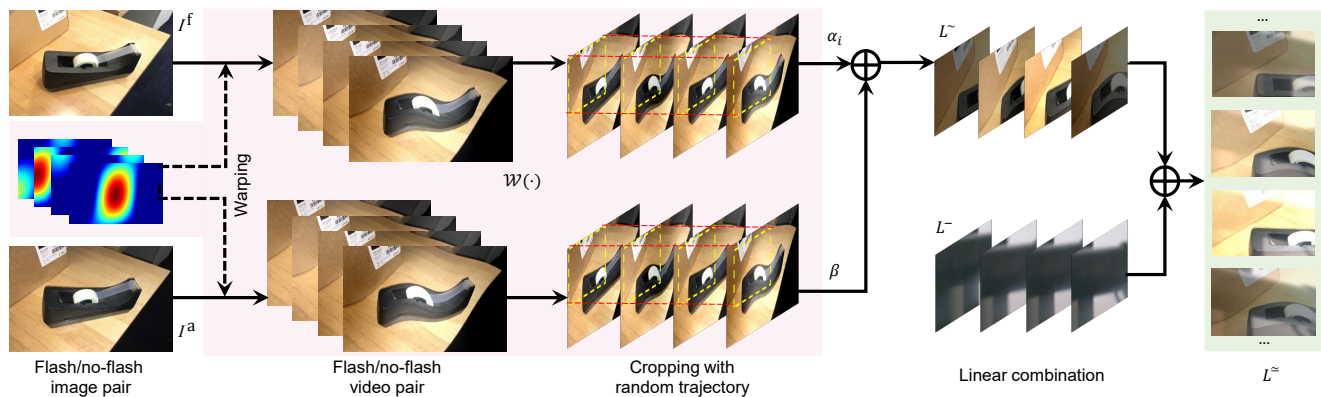


Fig. 4 The pipeline for synthesizing a mixture flickering video from a pair of flash/no-flash images. We first use a sequence of randomly generated masks to warp the flash/no-flash image pair, thereby obtaining a pair of flash/no-flash videos. Subsequently, the videos are cropped by using a randomly generated trajectory. Then, the flickering video is obtained by linearly combining the flash/no-flash videos. Finally, we linearly combine the flickering layer L^{\sim} and consistent layer L^{-} to obtain a mixture flickering video. Note that we randomly apply the defocus blur and ghosting effect to videos selected as reflection layers following previous works (Shi et al, 2015; Li et al, 2020a), which is not displayed in this figure.

respectively². By dropping a proportion of frames, a variety of flickering cycles can be simulated, causing the brightness variation of flickering frames to not strictly adhere to a regular sinusoidal waveform pattern. Besides, we randomly drop a part of the frames to simulate aperiodic flickers. The deflickering layer L_i^{\approx} is a linear combination of I_i^a and I_i^f , with fixed coefficients. Flickering and deflickering layers are shown in the first and second row of Fig. 5, respectively. We synthesize mixture videos with 1300 videos as transmission layers and 1365 videos as reflection layers. 2675 synthetic videos are selected as the training data, and 100 videos are used for testing.

In addition, we have collected a set of flicker-free videos that exhibit object motions. These videos are captured using a standard industrial camera and serve as the constant layer L_i^{-} . The third row of Fig. 5 displays examples from this dataset. This dataset also contains two subsets: 100 groups of transmission scenes captured without glass and 100 groups of reflection scenes captured with a piece of glass and a black cloth. Furthermore, we randomly cropped the videos to ensure that the number of constant layers is equivalent to that of flickering layers. Finally, the mixture frame I_i is defined as $\gamma L_i^{\sim} + \eta L_i^{-}$. Note that for videos selected as reflection layers, we apply random defocus blur and ghosting effect following previous works (Shi et al, 2015; Li et al, 2020a), as shown in the upper right example of Fig. 5. All videos for training and testing consist of 90 frames. In the fourth row of Fig. 5, we present two examples of synthetic mixture videos.

Real data for static scenes. For quantitative evaluation on real data, we collect a dataset denoted as STAF LIC which contains 20 groups of real data captured by an ordinary in-

dustrial camera³ and a tripod in static scenes with one-side light flickering (*i.e.*, 10 groups of transmission-flickering and 10 groups of reflection-flickering scenes), as shown in Fig. 5. Each group of data contains a mixture video with light flickering, the video of flickering layers with light flickering, and the video of constant layers without light flickering. Each video contains 30 frames. Since there do not exist motions (*i.e.*, spatial misalignments between frames) in the captured static data, we obtain deflickering layers by computing the temporal average of flickering layers. Note that correspondences between transmission and reflection layers to deflickering and constant layers vary across different data groups due to the light flickering at different sides.

Real data for dynamic scenes. To validate the effectiveness on dynamic scenes, we further capture a real dynamic dataset denoted as DYNFLIC which contains 20 groups of mixture videos (*i.e.*, 10 groups of transmission-flickering and 10 groups of reflection-flickering scenes) with one-side light flickering caused by AC light sources or PWM-controlled flashlights, as shown in Fig. 5. The lengths of videos in the DYNFLIC dataset range from 90 to 180 frames. Note that the captured mixture videos do not have corresponding ground truths since it is not feasible to record the same motion after removing the glass. As a result, the DYNFLIC dataset is only utilized for qualitative evaluation.

4 Proposed method

For a mixture video captured in a general case where light flickering and dynamic contents caused by motions both exist, the intensity variation for the same spatial position across

² Since the cycle of $\sin(2\pi i + \epsilon)$ is 1, when $f = 100$ and $\tau = 1/30$, the cycle of $\sin(20/3\pi i + \epsilon)$ becomes 3.

³ <https://www.edmundoptics.com/p/CM3-U3-50S5C-CS-2-3inch-Chameleon3-Color-Camera/37032>

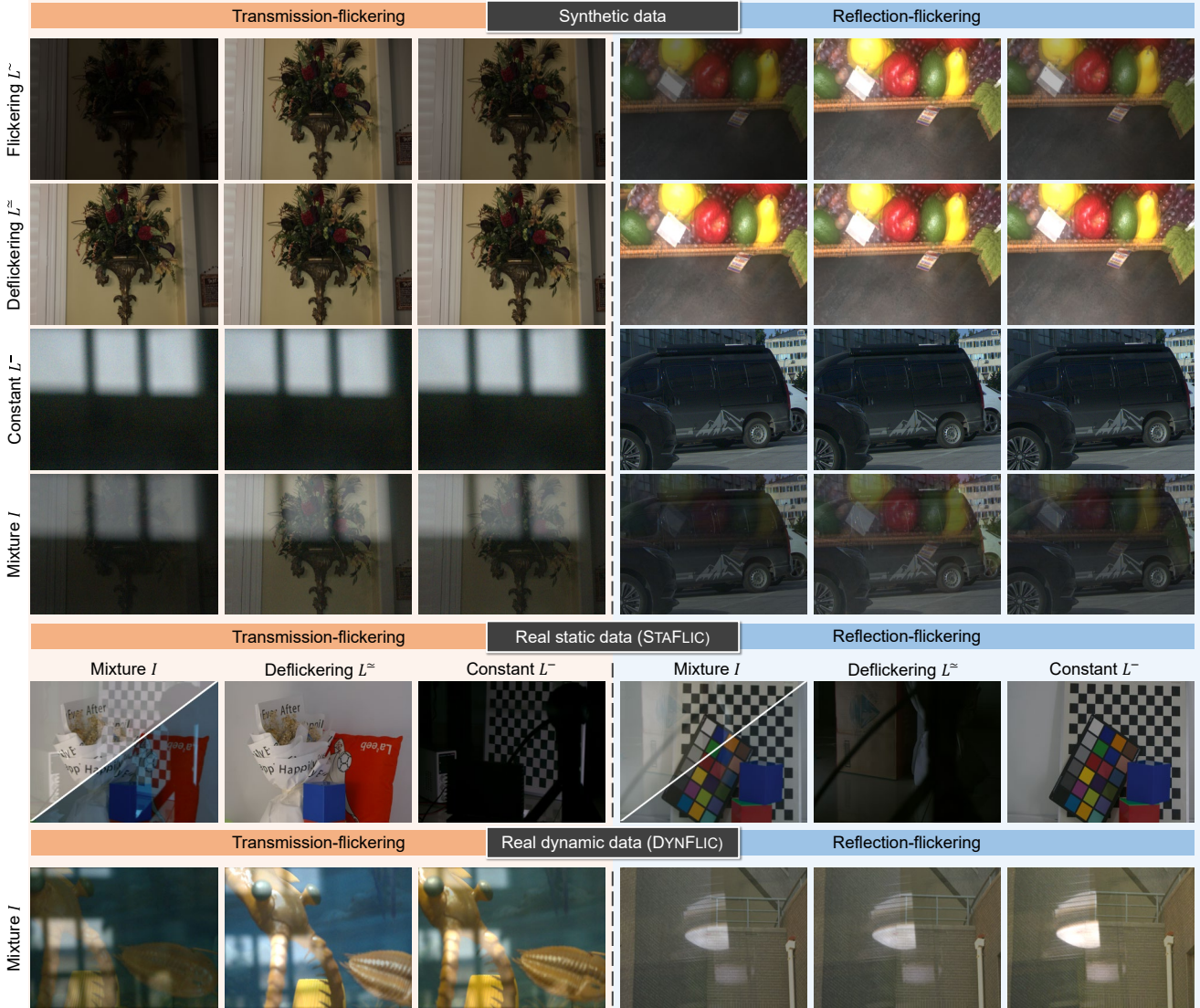


Fig. 5 Examples of our dataset for light flickering guided reflection removal. Top part: Examples from the synthetic dataset which contain flickering layers L^- , constant layers L^- , deflickering layers L^+ , and mixture videos I . Middle part: Examples from the real static dataset STAF LIC which contain constant layers L^- , deflickering layers L^+ , and mixture videos I . Bottom part: Examples from the real dynamic dataset DYNFLIC which only contain mixture videos I since it is infeasible to collect the ground truths. We pick three frames at different moments with brightness variations to show dynamic data and we represent static data as in Fig. 3(a).

video frames is not only related to light flickering but also the global and local motions. A video can be regarded as an observation of the captured scene, and the number of video frames corresponds to the observation time. For the mixture video with dynamic contents, the longer the observation time is, the more obvious the spatial misalignments across video frames are, and the more challenging to leverage the brightness variation caused by one-side light flickering. Since spatial misalignments in a short-term observation (several adjacent frames) is relatively mild, exploring the brightness variation within a short-term observation to provide one-side contextual clues is easier than using a long-term observation with more frames, *i.e.*, using the short-term observation is more

beneficial for exploiting the fluctuant component. However, a short-term observation only involves several frames that may not cover the unknown cycle of the brightness variation caused by light flickering, thus lacking priors for removing the flickering effect across frames. Fortunately, a long-term observation with more frames enables a more comprehensive analysis of the brightness variation, which facilitates robust flickering removal.

In this section, we propose a learning-based framework named **LI**ght **fl**ick**ER**ing guide reflection removal **Net**work (LIKE-Net) which utilizes short-term and long-term observations of mixture videos to exploit the one-side contextual clues in fluctuant components and brightness-consistent clues

in consistent components for achieving layer separation and flickering removal, respectively. The pipeline of the proposed method is shown in Fig. 6. In Sec. 4.1, we will describe how to leverage one-side contextual clues in fluctuant components by using a short-term observation. Sec. 4.2 describes how to exploit brightness-constant clues in consistent components from a long-term observation. With the guidance of the above clues from fluctuant and consistent components, we explore a Layer separation and Flickering removal Module (LFM) described in Sec. 4.3 to jointly achieve layer separation and flickering removal.

4.1 One-side contextual clues from short-term observation

To exploit the one-side contextual clues in fluctuant components, as shown in the upper part of Fig. 6 (with light red background), for each frame I_i in the mixture video $\{I_i\}_{i=0}^{N-1}$, we select its two adjacent frames I_{i-1} and I_{i+1} to form a short-term observation. Though we have selected the observation time as short as possible, the spatial misalignment between frames is still inevitable as long as motions exist. To ensure that the obtained contextual information is solely related to the brightness variation of light, rectifying inter-frame misalignment becomes an essential procedure. For aligning video frames with brightness variation, we take inspiration from high dynamic range (HDR) imaging methods (Yan et al, 2019; Wu et al, 2018) with the setting of exposure bracketing, which unifies the brightness at first and then align pixels in the feature space.

To map the adjacent mixture frames in a short-term observation into the feature space, we use a shared CNN-based encoder to process input frames in parallel, and the encoded mixture features are denoted as $\{F_j\}_{j=i-1}^{i+1}$. We conduct brightness unification (denoted as $\mathcal{U}(\cdot)$) and spatial alignment (denoted as $\mathcal{A}(\cdot)$) for the mixture features, and an average feature is obtained to indicate the average brightness of the current short-term observation:

$$F_i^{\text{avg}} = \frac{1}{N} \sum_{j=i-1}^{i+1} \mathcal{A}[\mathcal{U}(F_j)], \quad (8)$$

where $N = 3$ is the number of frames in a short-term observation. A fluctuant feature (denoted as F_i^{\sim}) which contains one-side contextual clues is then extracted from the combination of the center feature F_i and the average feature F_i^{avg} :

$$F_i^{\sim} = \mathcal{T}(F_i, F_i^{\text{avg}}), \quad (9)$$

where $\mathcal{T}(\cdot)$ is composed of a concatenation operator and several convolutional layers for channel reduction and information refinement. Details of the brightness unification and spatial alignment module will be described as follows.

Brightness unification. Inspired by HDR imaging techniques that aim to harmonize the varying brightness levels in alternating exposure images, we integrate attention blocks with a structure similar to those used in Yan et al (2019) to achieve brightness unification among the frames in the input short-term observation. However, unlike the approach in Yan et al (2019) which selects the middle-exposure image as the reference, our study acknowledges that each frame during the processing of flickering videos may not consistently exhibit a mid-level brightness. Consequently, as shown in Fig. 6, we opt to utilize the accumulated brightness of frames in the short-term observation as our reference to guide the brightness unification. We first accumulate mixture features as the reference feature: $F^{\text{ref}} = \sum_{j=i-1}^{i+1} F_j$. Then F^{ref} is concatenated with each mixture feature and fed into the attention block to learn the corresponding attention map, respectively. Finally, we multiply the attention maps with mixture features to obtain the unified features $\{F_j^{\text{uni}}\}_{j=i-1}^{i+1}$. The above procedures of brightness unification $\mathcal{U}(\cdot)$ can be described as:

$$F_j^{\text{uni}} = \mathcal{U}(F_j) = \mathcal{S}(\mathcal{A}_{tt}(F^{\text{ref}}, F_j)) \odot F_j, \quad (10)$$

where $\mathcal{S}(\cdot)$ denotes the Sigmoid function, $\mathcal{A}_{tt}(\cdot)$ denotes the attention block, and \odot denotes the element-wise multiplication operation.

Spatial alignment. We use the deformable convolution (Dai et al, 2017) to rectify spatial misalignment across adjacent frames. As shown in Fig. 6, we first select the centered unified feature F_i^{uni} as the reference feature, thus learning two collections of offsets for its neighbors F_{i-1}^{uni} and F_{i+1}^{uni} . Subsequently, the features are aligned by applying the learned offsets to the corresponding deformable convolutional layers. After the spatial alignment, we obtain the average feature F_i^{avg} by calculating the average of the aligned features. Since the brightness of the center feature F_i is not certain in a short-term observation (*i.e.*, it is likely to be either the brightest, the darkest, or in the middle), the brightness discrepancy between F_i and the average feature F_i^{avg} owns the potential to indicate the information about the fluctuant component, which further provides one-side contextual clues. Thus we concatenate F_i and F_i^{avg} and feed them into several convolutional layers to obtain a fluctuant feature F_i^{\sim} as described in Eqn. (9), which is then fed to LFM (details in Sec. 4.3) to facilitate layer separation.

4.2 Brightness-consistent clues from long-term observation

When recording a video with light flickering, the frame rate of the camera and frequency of light flickering both influence the cycle of the flickering effect in the mixture video. However, these two parameters vary with different capturing conditions and devices, resulting in the flickering cycle also

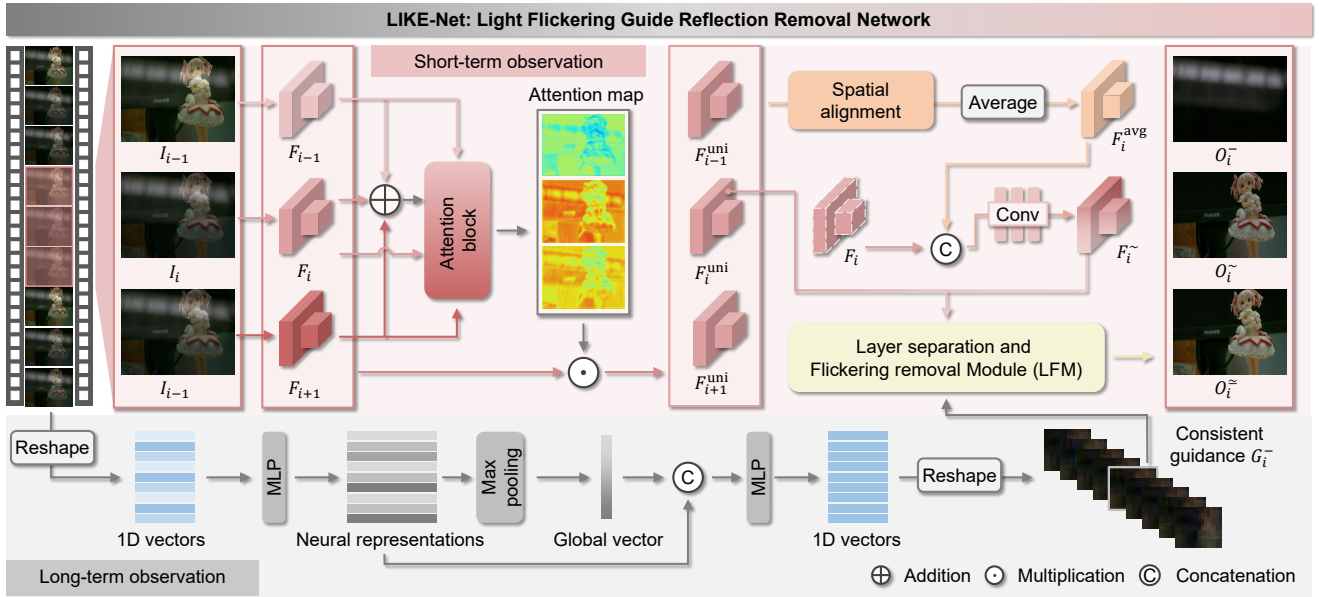


Fig. 6 LIKE-Net is a recurrent CNN-based network that simultaneously separates flickering and constant layers. At each step i , we extract the fluctuant feature F_i^{\sim} from a short-term observation for one-side contextual clues. To achieve this goal, we leverage a shared CNN-based encoder to extract multi-scale features $\{F_{i-1}, F_i, F_{i+1}\}$, utilize an attention block to rectify their uneven brightness, and align unified features by a spatial alignment block. Then the fluctuant feature is generated by applying the operations formulated in Eqn. (8) and Eqn. (9) successively. To obtain flickering-free layers, we force the model to learn the consistent feature from a long-term observation. Finally, we design a layer separation and flickering removal module (LFM) to simultaneously obtain the flickering layer O_i^- , the constant layer O_i^+ , and the deflickering layer O_i^{\sim} . More details of LFM are shown in Fig. 7.

differing among videos. Hence, a robust flickering removal procedure is required for recovering brightness-consistent results from videos with various flickering cycles. To achieve robust flickering removal, it is necessary to have a long-term observation that can reveal as much brightness variation information within the cycle of the flickering effect as possible. Considering the trade-off between the computational cost and the performance, as shown in the lower part of Fig. 6 (with gray background), we set a long-term observation to cover nine frames in total, which is composed of the current frame and its eight adjacent frames.

Low-resolution observation. Though a long-term observation can provide information about the regularity of frame brightness variation, it often exhibits more pronounced spatial misalignments across frames compared with the short-term observation, and rectifying such misalignments in a long-term observation can be a laborious task. Since the goal of using a long-term observation is for flickering removal, it is crucial to ensure that the observation remains sensitive to the flickering effect while being immune to misalignments caused by motions. Fortunately, we find that the observation on blurry or low-resolution frames still retains sensitivity to flickering while being less affected by motions. For instance, when one with high degrees of myopia is observing a distant flickering scene, she (he) is more likely to be responsive to the brightness variation of the scene while might pay less

attention to motions within the scene due to the unclarity. Therefore, as shown in Fig. 6, we downsample the frames in a long-term observation to a significantly low resolution for preserving the information of the brightness variation and diminishing the impact of inter-frame misalignments.

Learn brightness-consistent clues. To facilitate the removal of the flickering effect in captured videos, we introduce a permutation-invariant module to input with a long-term observation on low-resolution frames and learn brightness-consistent clues for the current frame I_i . As shown in Fig. 6, we first flatten each frame in a long-term observation into 1D vectors. Then we employ a set of multi-layer perceptrons (MLPs) with shared parameters to learn high-level neural representations of the long-term observation, and by applying a max pooling layer, we obtain a permutation-invariant vector that captures global information regarding the brightness variation across the entire observation. Subsequently, we concatenate the global vector with neural representations of the long observation and use another set of MLPs with shared parameters to learn the brightness-consistent clues and transform the representations back to the 2D frame space. We finally pick the frame at index i from the restored frames as the consistent guidance (denoted as G_i^-) and feed it to the layer separation and flickering removal module (Fig. 7, described in the next section) to provide auxiliary information on the brightness for flickering removal.

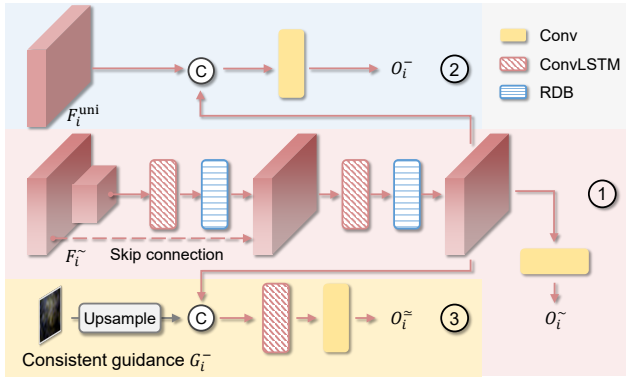


Fig. 7 The detailed architecture of the layer separation and flickering removal module (LFM).

4.3 Joint layer separation and flickering removal

We design a simple yet effective layer separation and flickering removal module (LFM) to jointly separate image layers and remove the flickering effect with the guidance extracted from short-term and long-term observations. As illustrated in Fig. 7, LFM consists of three branches that work together to accomplish our goal. A detailed description of each branch is provided as follows.

Branch ①: This branch serves as the backbone of LFM and outputs a flickering layer O_i^{\sim} . The fluctuant feature F_i^{\sim} extracted from the short-term observation has two scales, *i.e.*, the original scale and the half scale. Initially, we feed the half-scale feature of F_i^{\sim} to a number of ConvLSTM (Shi et al., 2015) layers to learn the relationship between neighboring frames. Afterward, several residual dense blocks (RDBs) are applied to further refine the features. Then, by using PixelShuffle (Shi et al., 2016), we increase the spatial resolution to the original scale while reducing the channel of features. As low-level features retain more texture information, we add a skip connection between the original scale feature of F_i^{\sim} and the upsampled feature. In addition, like the aforementioned structure, we continuously apply a set of ConvLSTM layers and RDBs. Finally, a set of convolutional layers are employed to generate the flickering layer O_i^{\sim} .

Branch ②: This branch decodes the constant layer O_i^{-} from F_i^{uni} and the feature from branch ①. This operation is intuitive since F_i^{uni} contains the information of both the constant and flickering layer, while F_i^{\sim} is only concerned with the flickering layer. Therefore, it is reasonable to learn the constant layer from these two features. We concatenate the two features and use several convolutional layers to obtain O_i^{-} .

Branch ③: This branch outputs a deflickering layer O_i^{\approx} at each step, and it helps to obtain a flicker-free frame sequence corresponding to flickering layers by collecting the outputs step by step. Flickering removal is achieved through two constraints: the consistent guidance learned from the long-

term observation and the guidance of previous output. The consistent guidance is upsampled and then concatenated with the feature from branch ①. By using a set of ConvLSTM layers which have been proven to be effective in removing flickering effects by feeding previous information (Chandran et al., 2022), we generate temporally consistent output, *i.e.*, the deflickering layer O_i^{\approx} .

4.4 Implementation details

Network training and loss functions. Loss functions of three branches are denoted as \mathcal{L}^{\sim} , \mathcal{L}^{-} , and \mathcal{L}^{\approx} , respectively. Each loss function of the corresponding branch is composed of three equal-weighted inner loss functions: the MSE loss, the SSIM loss (Wan et al., 2019), and the calibrated perceptual loss (Zhang et al., 2018a) computed by the VGG model (Simonyan and Zisserman, 2014). The total loss at each step i is formulated as:

$$\mathcal{L}_i^{\text{total}} = \mathcal{L}_i^{\sim} + \omega_1 \mathcal{L}_i^{-} + \omega_2 \mathcal{L}_i^{\approx}, \quad (11)$$

where ω_1 and ω_2 are coefficients that gradually increase from 0 to 1 during the training procedure. During training, we unroll the recurrent units in the model for S steps. The total loss for the unrolled S steps is calculated by $\mathcal{L}^{\text{total}} = \frac{1}{S} \sum_{j=i}^{i+S-1} \mathcal{L}_j^{\text{total}}$. In our experiments, we set $S = 2$ to conduct gradient backpropagation with the average loss of two consecutive steps, which stabilizes the loss and facilitates the convergence of the network. However, during the inference phase, LIKE-Net generates a single output frame at each step without producing intermediate results. The model is implemented using PyTorch (Paszke et al., 2019) and trained with the batch size of 1. The initial learning rate is set as 10^{-4} and is gradually decayed to 10^{-7} after 50 epochs.

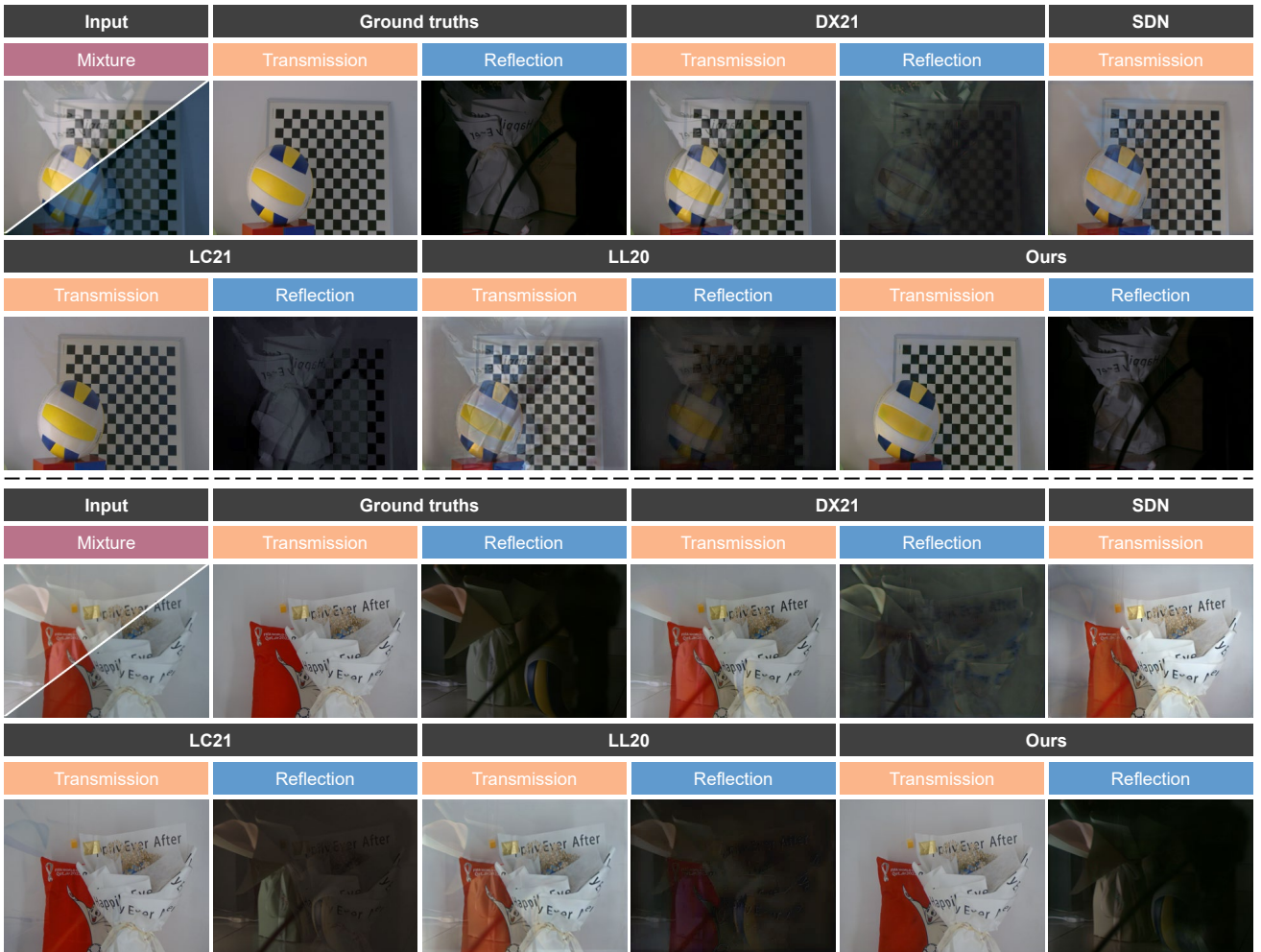
5 Experiments

5.1 Comparison with state-of-the-arts

We compare the proposed method with two single-image reflection removal methods (*i.e.*, IBCLN (Li et al., 2020a) and DX21 (Dong et al., 2021)) and two flash-based methods (*i.e.*, LC21 (Lei and Chen, 2021) and SDN (Chang et al., 2020)) for quantitative and visual quality evaluation. To make flash-based methods (Lei and Chen, 2021; Chang et al., 2020) originally designed for flash/no-flash image pairs applicable in our setting that inputs with mixture videos containing light flickering, for each frame in the input video, we select its adjacent frame containing brightness variation as the other input. Specifically, we select the brighter frame as the flash image and the darker one as the no-flash image. To achieve

Table 1 Quantitative comparisons on the synthetic dataset with light flickering at different sides of the glass. \uparrow (\downarrow) indicates larger (smaller) values are better. Bold numbers indicate the best results. “-” indicates that the compared method cannot provide the required outputs.

Method	Transmission-flickering						Reflection-flickering					
	Transmission layer			Reflection layer			Transmission layer			Reflection layer		
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
IBCLN	22.29	0.815	0.284	16.89	0.634	0.597	22.51	0.822	0.277	17.12	0.621	0.637
DX21	22.56	0.828	0.273	19.15	0.702	0.519	22.67	0.836	0.260	19.41	0.719	0.488
LC21	22.12	0.823	0.268	15.64	0.543	0.682	22.86	0.842	0.259	16.97	0.684	0.554
SDN	21.43	0.812	0.298	-	-	-	21.30	0.813	0.294	-	-	-
LL20	23.28	0.830	0.265	19.79	0.721	0.485	23.61	0.848	0.251	19.47	0.726	0.483
Ours	31.96	0.922	0.154	23.78	0.823	0.277	32.31	0.919	0.145	24.12	0.827	0.271

**Fig. 8** Visual quality comparisons on the STAFlic dataset, compared with several state-of-the-art reflection removal methods, including a single-image method (*i.e.*, DX21 (Dong et al, 2021)), two flash-based methods (*i.e.*, LC21 (Lei and Chen, 2021) and SDN (Chang et al, 2020)), and a motion-based method (*i.e.*, LL20 (Liu et al, 2020)). Note that SDN (Chang et al, 2020) only estimates transmission layers. We show two groups of examples captured with transmission-flickering (the top part) and reflection-flickering (the bottom part) scenarios, respectively, and input mixture videos are shown in the same manner as Fig. 3(a).

comprehensive evaluation, we further compare with a motion-based method (*i.e.*, LL20 (Liu et al, 2020)) by using adjacent frames as inputs.

Evaluation on synthetic data. Quantitative results on the synthetic data are shown in Table 1. Following previous works (Zheng et al, 2021; Lei and Chen, 2021), we adopt PSNR (Huynh-Thu and Ghanbari, 2008), SSIM (Wang et al,

Table 2 Quantitative comparisons on the STAF LIC dataset for static scenes with light flickering at different sides of the glass. \uparrow (\downarrow) indicates larger (smaller) values are better. Bold numbers indicate the best results. “-” indicates that the compared method cannot provide the required outputs.

Method	Transmission-flickering						Reflection-flickering					
	Transmission layer			Reflection layer			Transmission layer			Reflection layer		
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
IBCLN	22.49	0.843	0.254	16.71	0.523	0.598	24.21	0.848	0.249	16.94	0.546	0.662
DX21	24.03	0.850	0.243	17.59	0.562	0.573	23.70	0.840	0.262	17.09	0.538	0.687
LC21	25.13	0.871	0.211	20.01	0.702	0.495	24.78	0.883	0.202	19.57	0.697	0.521
SDN	25.09	0.859	0.228	-	-	-	23.97	0.862	0.234	-	-	-
LL20	23.72	0.845	0.249	17.12	0.538	0.655	24.01	0.843	0.255	17.25	0.604	0.644
Ours	27.27	0.898	0.176	22.78	0.814	0.292	27.85	0.912	0.159	22.41	0.808	0.353

Table 3 Comparisons on the inference time and the model size (with the video resolution equal to 240×320).

Metric	Method					
	IBCLN	DX21	LC21	SDN	LL20	Ours
Time (s/frame)	0.034	0.044	0.154	0.167	1.532	0.311
Params (M)	21.61	10.93	52.49	5.93	42.41	25.38

2003), and LPIPS (Zhang et al, 2018a) as evaluation metrics. Due to the lack of ground truths for real dynamic videos, quantitative comparisons for dynamic scenes are only conducted on synthetic data as the previous method did (Liu et al, 2020). It can be seen that the proposed method achieves the best performance among all metrics, which demonstrates the feasibility of the unified setting that introduces light flickering into reflection removal.

Evaluation on real data. To validate the effectiveness of introducing one-side light flickering into reflection removal, we first conduct quantitative and qualitative experiments on the collected real dataset STAF LIC which is captured with static scenes. As results shown in Table 2 and Fig. 8, single-image methods (Li et al, 2020a; Dong et al, 2021) encounter challenges in distinguishing transmission and reflection layers, while flash-based methods (Chang et al, 2020; Lei and Chen, 2021) especially LC21 (Lei and Chen, 2021) achieves acceptable reflection removal results since in static scenes the brightness differences between frames play a similar role as flash-only images for providing helpful content information. The motion-based method LL20 (Liu et al, 2020) fails in static scenes as it requires observations of the same scene from different viewpoints to exploit distinguishable motions of transmission and reflection layers. The proposed method outperforms other methods in both qualitative metrics and visual quality, demonstrating the efficacy of leveraging one-side contextual clues introduced by light flickering and the effectiveness of our synthetic data for network training.

We conduct visual quality comparisons on the real-word dataset DYNFLIC captured with dynamic scenes to evaluate the performance of the proposed method and other state-of-the-art reflection removal methods. Results are shown

in Fig. 9 and Fig. 10, which display cases when transmission and reflection layers are with light flickering, respectively. It can be observed that the single-image method (Dong et al, 2021) can not effectively remove strong reflections due to the lack of auxiliary contextual information, and flash-based methods (Lei and Chen, 2021; Chang et al, 2020) designed for aligned flash/no-flash image pairs generate results with ghosting artifacts, since spatial misalignment commonly exists for dynamic scenes. The motion-based method LL20 (Liu et al, 2020) also fails to remove reflections, as the brightness variation caused by light flickering interferes with the motion estimation of transmission and reflection layers and further hinders the performance of reflection removal. In general, by exploiting reflection-aware information from one-side light flickering, the proposed method achieves high-fidelity layer separation and flickering removal for general dynamic scenes. Besides, we show an example of real data captured by a mobile phone (*i.e.*, HUAWEI P40 Pro) in Fig. 11, which demonstrates the robustness of the proposed method.

We further compare the inference time and model size (number of parameters) of the proposed LIKE-Net with state-of-the-art reflection removal methods. As shown in Table 3, for a video with the resolution of 240×320 , LIKE-Net spends more time processing the video than single-image (Li et al, 2020a; Dong et al, 2021) and flash-based methods (Chang et al, 2020; Lei and Chen, 2021) since LIKE-Net estimates frames from short-term and long-term observations, but it is still more efficient than the motion-based method (Liu et al, 2020). Besides, the model size of LIKE-Net is comparable with the single-image method IBCLN (Li et al, 2020a), yet smaller than both the flash-based method LC21 (Lei and Chen, 2021) and the motion-based method LL20 (Liu et al, 2020), indicating that the proposed method achieves a trade-off between the model performance and inference efficiency.

5.2 Ablation study

Effectiveness of short/long-term observations. To verify the effectiveness of the short/long-term observations, we compare the proposed method with its two variants: ‘W/o short’

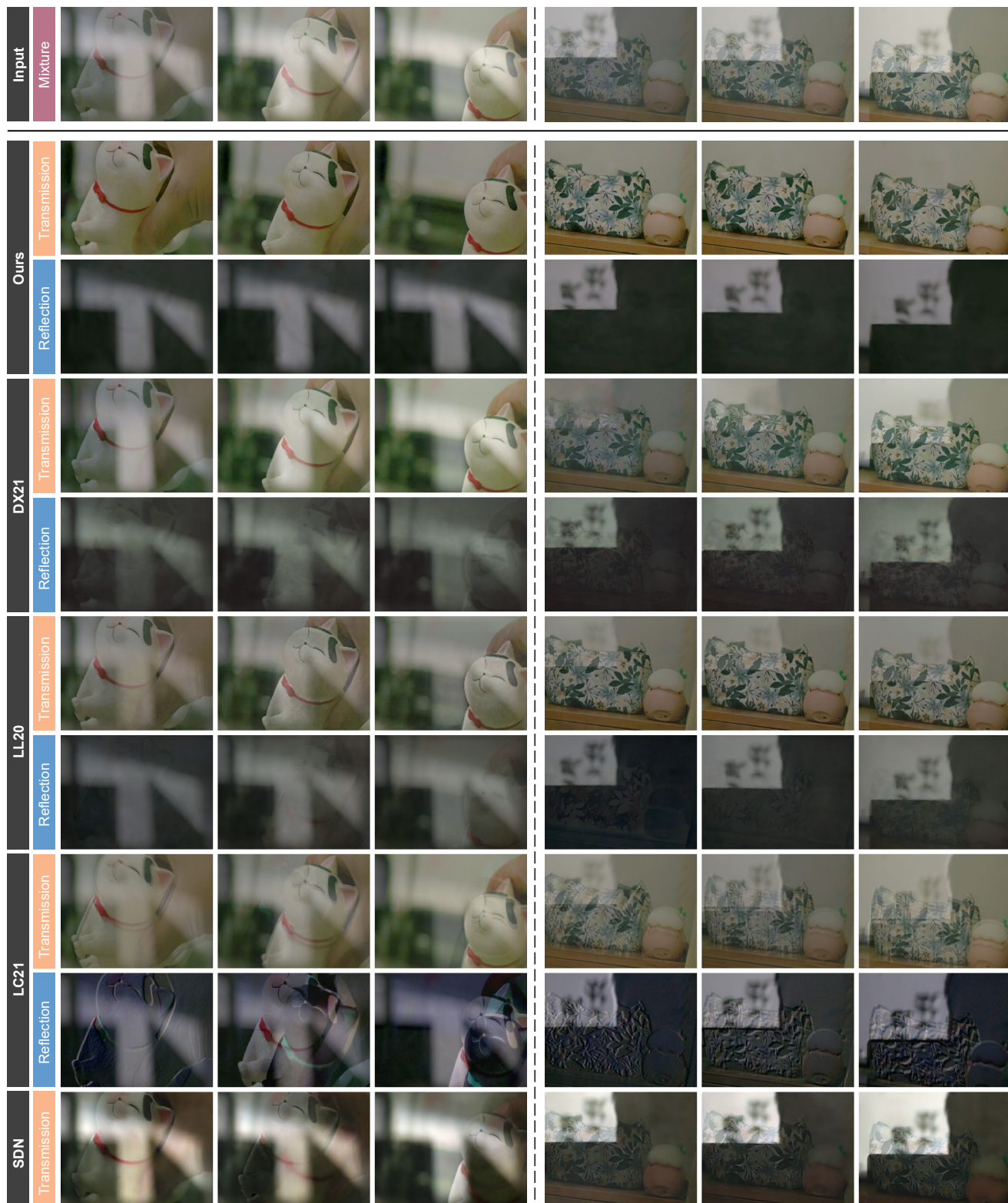


Fig. 9 Visual quality comparisons on real data captured with transmission-flickering scenarios, compared with several state-of-the-art reflection removal methods, including a single-image method (*i.e.*, DX21 (Dong et al, 2021)), a motion-based method (*i.e.*, LL20 (Liu et al, 2020)), and two flash-based methods (*i.e.*, LC21 (Lei and Chen, 2021) and SDN (Chang et al, 2020)). Note that SDN (Chang et al, 2020) only estimates transmission layers. We show two sets of data (captured with real-world indoor light sources by setting the exposure time as $1/200$ s and the frame rate as 30 FPS) and pick three frames at different moments with brightness variations in each set.

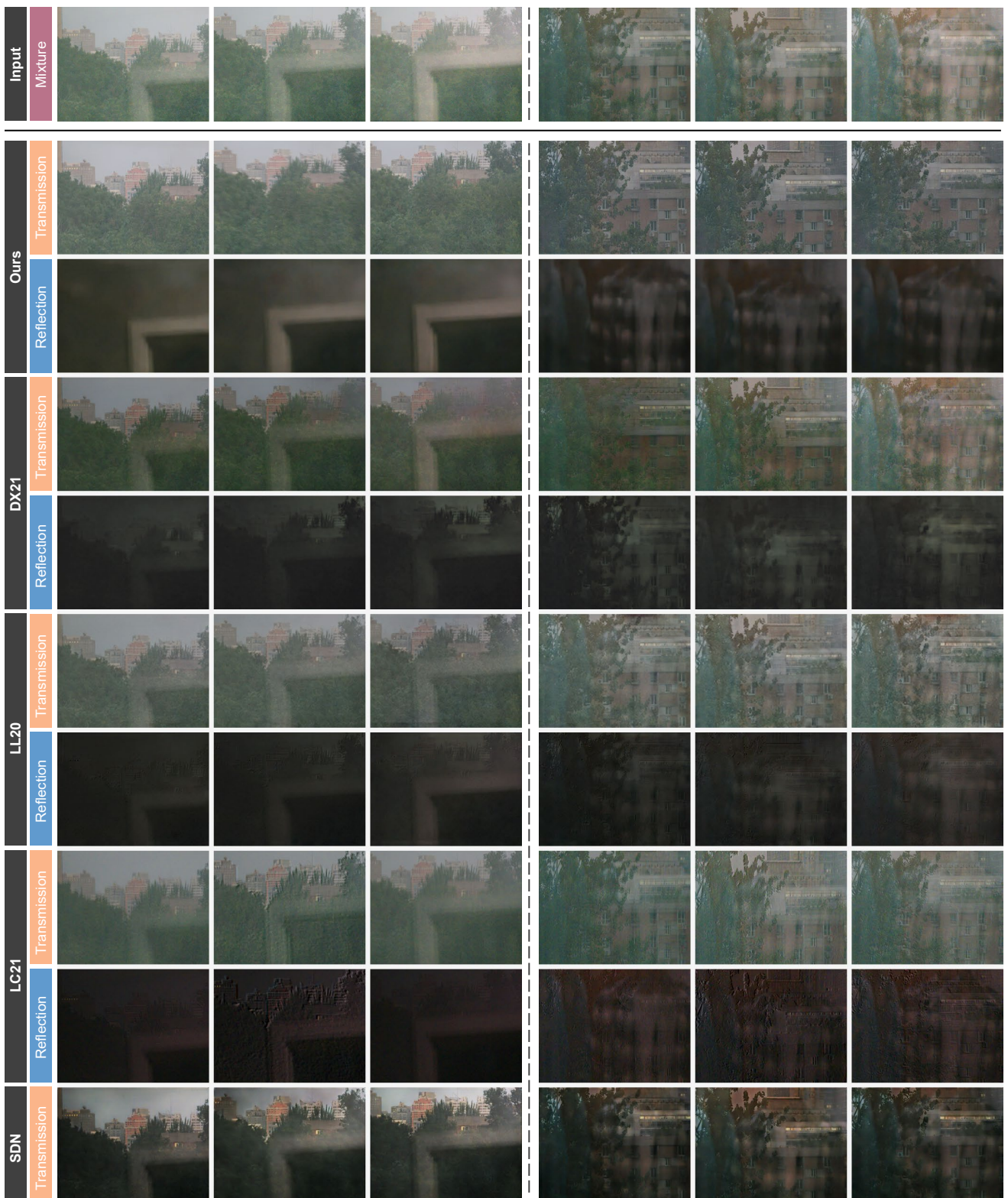
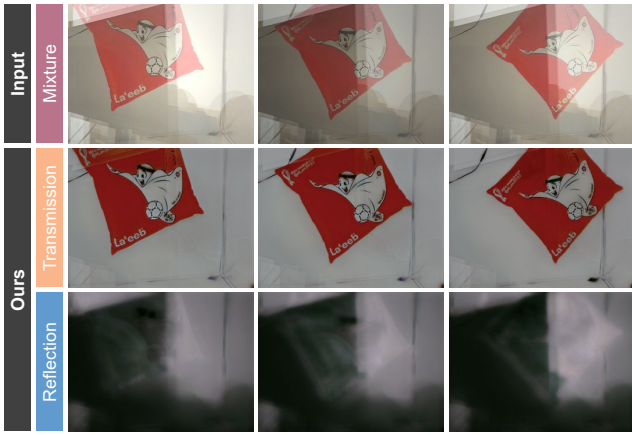


Fig. 10 Visual quality comparisons on real data captured with reflection-flickering scenarios, compared with several state-of-the-art reflection removal methods, including a single-image method (*i.e.*, DX21 (Dong et al, 2021)), a motion-based method (*i.e.*, LL20 (Liu et al, 2020)), and two flash-based methods (*i.e.*, LC21 (Lei and Chen, 2021) and SDN (Chang et al, 2020)). Note that SDN (Chang et al, 2020) only estimates transmission layers. We show two sets of data (captured with real-world indoor light sources by setting the exposure time as $1/200$ s and the frame rate as 30 FPS) and pick three frames at different moments with brightness variations in each set.

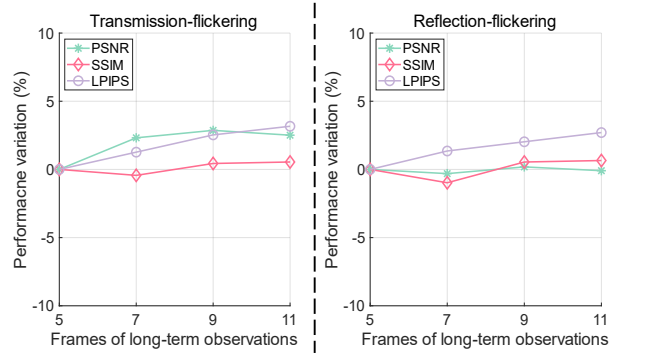
Table 4 Ablation study on the synthetic dataset with light flickering at different sides of the glass. \uparrow (\downarrow) indicates larger (smaller) values are better. Bold numbers indicate the best results.

Method	Transmission-flickering						Reflection-flickering					
	Transmission layer			Reflection layer			Transmission layer			Reflection layer		
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
Ours	31.96	0.922	0.154	23.78	0.823	0.277	32.31	0.919	0.145	24.12	0.827	0.271
W/o short	25.72	0.845	0.256	20.41	0.745	0.448	25.94	0.853	0.239	20.78	0.750	0.442
W/o long	28.31	0.889	0.207	22.23	0.798	0.397	28.65	0.891	0.215	22.84	0.787	0.428
W/o BU	28.97	0.896	0.184	22.56	0.790	0.402	29.65	0.889	0.188	22.03	0.802	0.376
W/o SA	30.15	0.914	0.172	22.95	0.818	0.293	30.28	0.907	0.161	23.34	0.815	0.315
W/o LFM	25.86	0.852	0.245	20.53	0.756	0.424	26.04	0.860	0.237	20.62	0.744	0.457
W/o ①-②	29.32	0.897	0.186	21.42	0.765	0.419	26.51	0.869	0.221	22.96	0.791	0.413

**Fig. 11** Visual quality results of data captured by a mobile phone (*i.e.*, HUAWEI P40 Pro) in a transmission-flickering scene.

removes the extraction of F_i^{\sim} and directly feeds F_i to LFM; ‘W/o long’ removes the long-term observation and directly feeds a low-resolution image of I_i to the LFM. Table 4 shows quantitative evaluation results of the two variants. Due to the lack of guidance from the reflection-aware short-term observation, ‘W/o short’ suffers from severe performance degradation compared with the complete model, indicating the necessity of auxiliary information brought by light flickering. ‘W/o long’ also performs worse than the complete model, since the unstable brightness among recovered frames will lead to the degradation of image quality.

Furthermore, to investigate the influence of the number of frames in long-term observations, we conduct an ablation study by setting long-term observations to be composed of 5, 7, 9, and 11 adjacent frames, respectively. The curves shown in Fig. 12 indicate that the performance of the proposed method improves as long-term observations utilize more frames. Since the performances are similar when long-term observations consist of 9 and 11 frames, we finally use the version of 9 frames to achieve a trade-off of the performance and computational cost.

**Fig. 12** Performance variation curves that illustrate the impact of varying the number of frames in long-term observations. Note that the y axis denotes the variation rate of the performance on recovering transmission layers, which selects the variant that uses 5 frames for long-term observations as the reference. Experiments are conducted on the synthetic dataset with transmission-flickering and reflection-flickering scenes.

Effectiveness of network modules. We conduct ablation studies on the network modules of LIKE-Net with the following variants: ‘W/o BU’ that excludes the brightness unification module, ‘W/o SA’ that removes the spatial alignment module, ‘W/o LFM’ that removes branches ② and ③ in the LFM and directly estimates O_i^{\sim} , O_i^- , and O_i^{\approx} with three convolutional layers, and ‘W/o ①-②’ that removes the connection between branches ① and ②. Note that ‘W/o LFM’ involves ‘W/o long’ since the long-term guidance is not utilized in ‘W/o LFM’. As results shown in Table 4, compared with the complete model, removing either the brightness unification module (‘W/o BU’) or the spatial alignment module (‘W/o SA’) both result in the degradation of performance, and ‘W/o LFM’ suffers from significant performance decline, indicating the effectiveness of the network design of the proposed method to exploit the auxiliary information from the fluctuant components. Besides, the performance of ‘W/o LFM’ degrades more than ‘W/o long’, indicating the effectiveness of the ConvLSTM and convolutional layers in LFM. In addition, removing the connection between branches ①

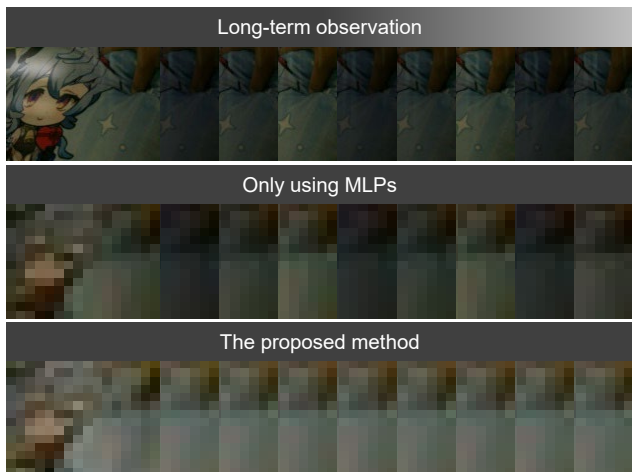


Fig. 13 Verification of the architecture for estimating consistent guidance. Top row: An input long-term observation that shows brightness variation. Middle row: The sequence obtained by replacing the proposed architecture with MLPs. Bottom row: The sequence obtained by the architecture of the proposed method.



Fig. 14 Qualitative results on removing the flickering effect from clean flickering videos without reflection contaminations.

and ② (‘W/o ①-②’) leads to a significant degradation to recover layers with consistent illumination, which shows the efficacy of the fluctuant feature. To validate the advantages of the architecture of the sub-network for extracting brightness-consistent clues from long-term observations, we replace the sub-network with a set of MLPs, *i.e.*, the layers after ‘Neural presentations’ in Fig. 6 are removed. Fig. 13 illustrates the comparison of consistent guidance extraction between the proposed architecture and MLPs, which indicates that simply using MLPs fails to extract the consistent guidance.

In addition, we conduct experiments on clean videos (*i.e.*, no reflection contamination) with light flickering to investigate the effectiveness of the employed brightness-consistent clues from long-term observations. As shown in Fig. 14, the brightness of results is more uniform compared with input frames, which demonstrates the effectiveness of exploiting consistent features in flickering videos to guide the brightness unification and further indicates the potential of the proposed method for video flickering removal.

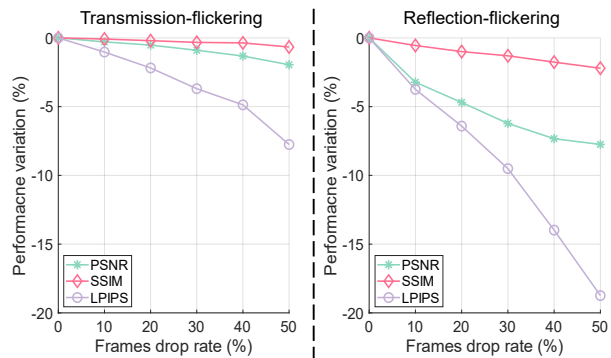


Fig. 15 Performance variation curves that illustrate the impact of unknown flickering cycles by randomly dropping frames in input videos. Note that the y axis denotes the variation rate of the performance on recovering transmission layers, which selects the situation with no dropping frames as the reference. Experiments are conducted on the synthetic dataset with transmission-flickering and reflection-flickering scenes.

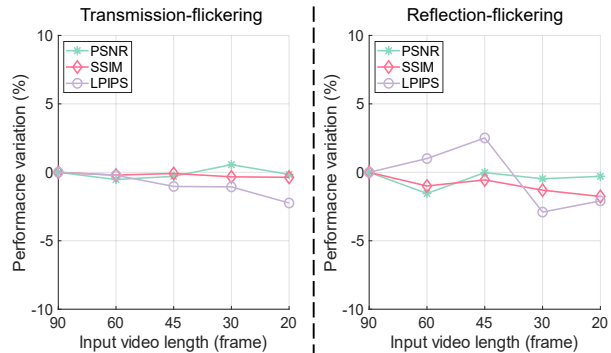


Fig. 16 Performance variation curves that illustrate the influence of input video lengths. Note that the y axis denotes the variation rate of the performance on recovering transmission layers, which selects the situation where input videos contain 90 frames as the reference. Experiments are conducted on the synthetic dataset with transmission-flickering and reflection-flickering scenes.

Results on unknown flickering cycles. Given that the flickering cycle of a mixture video is jointly influenced by two variables, namely the frame rate and the light flickering cycle, it is crucial for a method to effectively handle flickering in unknown cycles. To validate the effectiveness of our method on such flickers in an objective manner, we randomly drop frames to disrupt the periodicity of the synthetic data. We set the frame drop rate from 10% to 50% to observe the degradation trend of quantitative evaluation metrics. We select the metrics when the frame drop rate is 0 as the reference, the degradation rate is defined as $|M_d - M_r|/M_r \times 100\%$, where M_d is the observed metric, M_r is the reference metric. In Fig. 15, it is evident that the performance of the method progressively deteriorates as the frame drop rate increases. Specifically, the degradation of PSNR and SSIM metrics remains insignificant, whereas LPIPS exhibits a slightly higher

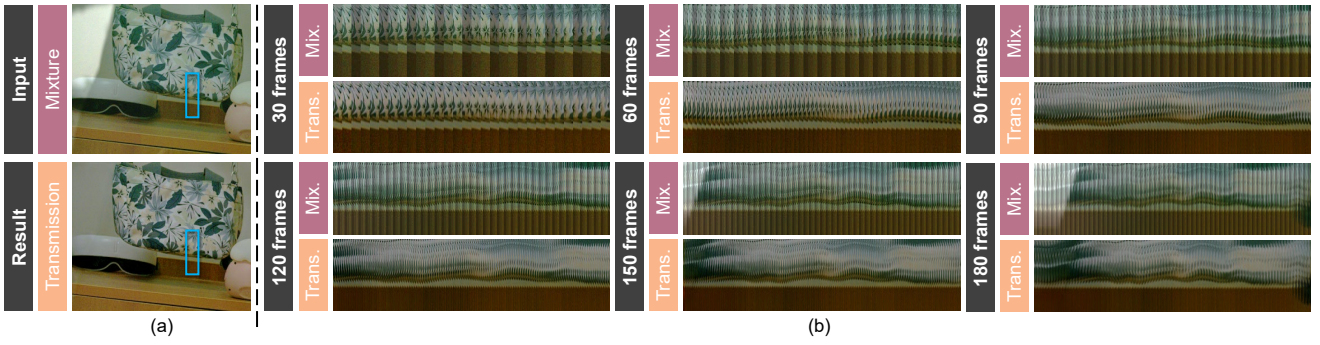


Fig. 17 Illustration of the robustness of the proposed method on flickering and reflection removal with different lengths of input videos. (a) Example frames from the input mixture video and recovered transmission video. (b) Visualization of the temporal brightness variation in the input mixture and recovered transmission video with different lengths. Specifically, for each video, we combine regions at the same location (highlighted with blue boxes in (a)) by the temporal order for better visualization.

degree of degradation. Nevertheless, the overall degradation rate across the metrics remains relatively low, underscoring the effectiveness of our method under unpredictable cycles of light flickering.

Influence of input video lengths. We conduct ablation studies to investigate the influence of input video lengths. As can be observed in Fig. 16, as the frame numbers of input videos decrease, the performance of the proposed method remains essentially unchanged. We further visualize the temporal brightness variation of video frames in Fig. 17. For different input video lengths, frames in input mixture videos suffer from obvious brightness inconsistency while the recovered transmission videos remain consistent in brightness, which demonstrates the capability of the proposed method to achieve robust reflection and flickering removal.

Results on inconspicuous light flickering. To validate the effectiveness of light flickering in removing reflections, it is necessary to compare results estimated from mixture videos with different degrees of light flickering. However, since capturing real data in dynamic scenes with motions for comparative analysis lacks repeatability, we opt for synthetic data to conduct qualitative evaluations. A qualitative experiment is conducted with results shown in Fig. 18, compared with a single-image reflection removal method (Dong et al, 2021). It can be observed that when the degree of light flickering becomes less conspicuous (the middle part), there is a slight decline in the performance of the proposed method, yet a more effective reflection removal is accomplished compared with the single-image method, demonstrating the significance of one-side contextual clues provided by light flickering. Besides, when there is no light flickering (the bottom part), the proposed method still correctly preserves the content of transmission layers, which indicates our superiority.

Ablation study of S . During the training phase, we employ the recurrent structures and set the parameter $S = 2$ to backpropagate the average loss over two consecutive steps, aiming to stabilize the training process. We conduct an ab-

lation study by plotting the loss curves of setting $S = 1$ and $S = 2$ to validate the effectiveness of the recurrent structure. Curves in Fig. 19 reveal that using a backpropagation step of $S = 1$ results in a relatively more unstable descent trend of the total loss function than $S = 2$, which is likely caused by the variations of the illumination brightness and reflection intensity between adjacent frames in the mixture video with light flickering.

5.3 Application to a high-speed camera

When the captured scenes with light flickering contain fast motions, the performance of the proposed method may degrade due to the low frame rate of conventional digital cameras (usually 30 or 60 FPS). Fortunately, we can extend the application scope of the proposed method for such challenging cases by using a spiking camera (Huang et al, 2023; Chang et al, 2023), which has the attractive high-speed characteristic (perceiving scene radiance changes at 20K FPS in the forms of spikes). As shown in Fig. 20, reflection removal results using data captured by a conventional RGB camera are severely degraded by blurry artifacts, whereas the ones captured by the high-speed spiking camera are free from reflections and show clear textures of the objects, which demonstrates the effectiveness of the proposed method in high-speed scenes.

6 Conclusion

By exploiting the widely observed light flickering, we achieve reflection removal for general dynamic scenes with a unified setting. We model the image formation process when one side of the glass is under a periodically varying illumination, and demonstrate that the fluctuant component can provide one-side contextual clues. A learning-based framework is proposed to tackle misalignment issues and accomplish layer

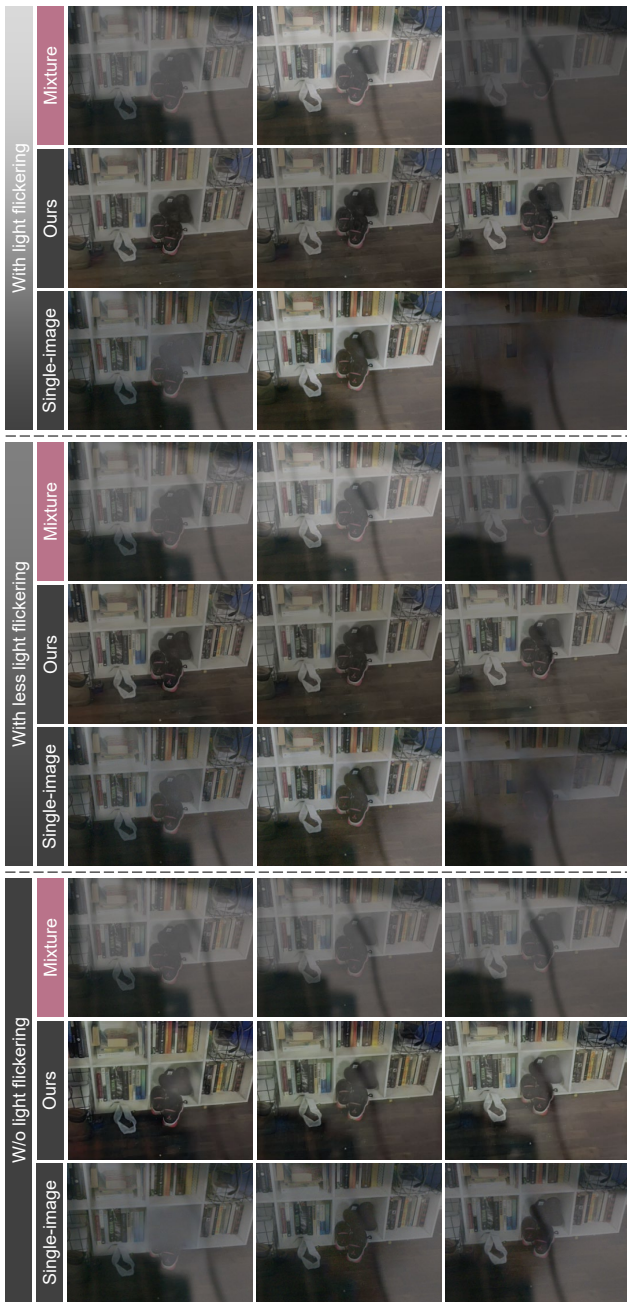


Fig. 18 Qualitative results of recovered transmission layers from mixture videos with different degrees of light flickering, compared with a single-image reflection removal method (Dong et al, 2021). Top part: The mixture video with light flickering and the reflection removal results. Middle part: The mixture video with less light flickering (the intensity of flickering is set at 40% of that in the top part) and results. Bottom part: The mixture video with no light flickering and results.

separation and flickering removal for general dynamic scenes. Quantitative and qualitative results show the effectiveness of the proposed method. Besides, additional experiments also indicate the applicability of video flickering removal and the capability for reflection removal in scenes with fast motions (by using a high-speed camera).

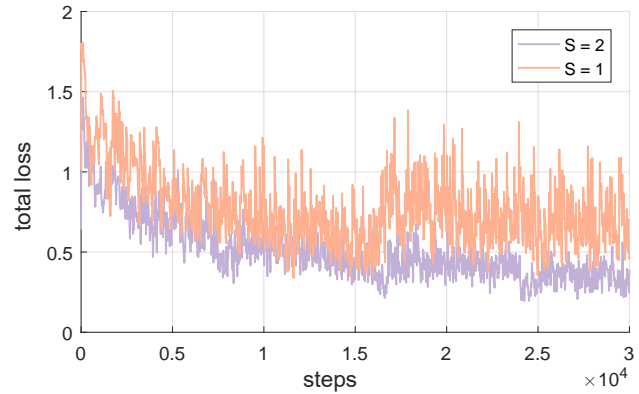


Fig. 19 Visualization of the total loss when training the proposed network with $S = 1$ and $S = 2$.

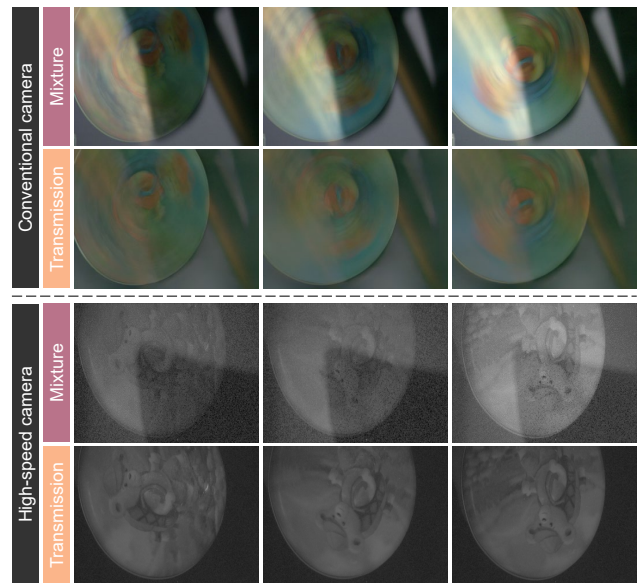


Fig. 20 We extend the proposed method by using a high-speed spiking camera (Chang et al, 2023), which accomplishes clearer recovery of transmission layers with less blurring compared with the situation that uses a conventional camera.

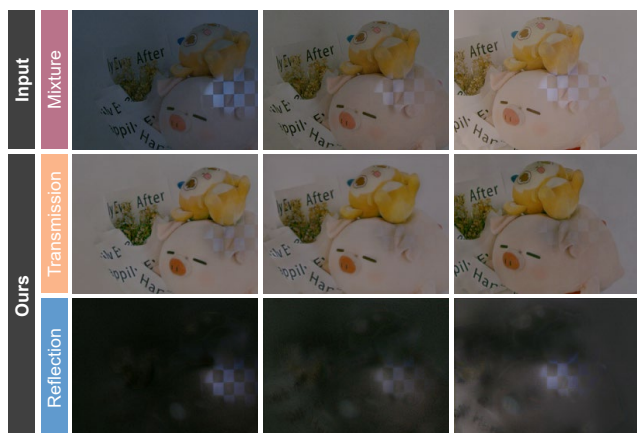


Fig. 21 A failure case of the proposed method, where both transmission and reflection scenes exhibit light flickering.

Limitations. Capturing videos with light flickering typically requires short exposures, resulting in low-light imagery with higher noise levels. Though we can enhance the brightness through tone mapping techniques, the mitigation of noise is not achieved. Nonetheless, in pursuing the goal of reflection removal, our primary focus lies in minimizing the impact of reflections. The decision to employ short exposures, despite the associated increase in noise, represents a deliberate trade-off aimed at achieving superior performance in reflection removal. Besides, the proposed method leverages auxiliary contextual clues from one-side light flickering, however, as shown in Fig. 21, when both transmission and reflection scenes exhibit light flickering, a decline in performance can be observed. It is noteworthy that if the light flickering in the two scenes is asynchronous, indicating that the two layers reach their maximum brightness at different moments, this inherently contains clues for layer separation. By further analyzing phases of the light flickering, the above issue may be tackled, which is left as our future work.

Acknowledgement

This work was supported by National Science and Technology Major Project (Grant No. 2021ZD0109803), National Natural Science Foundation of China under Grant No. 62301009, 62088102, and 62136001.

Data availability. Datasets used in this study are available from co-first authors on reasonable request.

References

- Aksoy Y, Kim C, Kellnhofer P, Paris S, Elgharib M, Pollefeys M, Matusik W (2018) A dataset of flash and ambient illumination pairs from the crowd. In: Proc. of European Conference on Computer Vision
- Chandran S, Hold-Geoffroy Y, Sunkavalli K, Shu Z, Jayasuriya S (2022) Temporally consistent relighting for portrait videos. In: Proc. of Winter Conference on Applications of Computer Vision
- Chang Y, Jung C, Sun J, Wang F (2020) Siamese dense network for reflection removal with flash and no-flash image pairs. *International Journal of Computer Vision* 128(6):1673–1698
- Chang Y, Zhou C, Hong Y, Hu L, Xu C, Huang T, Shi B (2023) 1000 fps hdr video with a spike-rgb hybrid camera. In: Proc. of Computer Vision and Pattern Recognition
- Cooley JW, Tukey JW (1965) An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation* 19(90):297–301
- Dai J, Qi H, Xiong Y, Li Y, Zhang G, Hu H, Wei Y (2017) Deformable convolutional networks. In: Proc. of Computer Vision and Pattern Recognition
- Diamant Y, Schechner YY (2008) Overcoming visual reverberations. In: Proc. of Computer Vision and Pattern Recognition
- Dong Z, Xu K, Yang Y, Bao H, Xu W, Lau RW (2021) Location-aware single image reflection removal. In: Proc. of International Conference on Computer Vision
- Fan Q, Yang J, Hua G, Chen B, Wipf D (2017) A generic deep architecture for single image reflection removal and image smoothing. In: Proc. of International Conference on Computer Vision
- Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. Proc of Advances in Neural Information Processing Systems
- Hong Y, Lyu Y, Li S, Shi B (2020) Near-infrared image guided reflection removal. In: Proc. of International Conference on Multimedia and Expo
- Hong Y, Zheng Q, Zhao L, Jiang X, Kot AC, Shi B (2021) Panoramic image reflection removal. In: Proc. of Computer Vision and Pattern Recognition
- Hong Y, Lyu Y, Li S, Cao G, Shi B (2023a) Reflection removal with NIR and RGB image feature fusion. *IEEE Transactions on Multimedia* 25:7101–7112
- Hong Y, Zheng Q, Zhao L, Jiang X, Kot AC, Shi B (2023b) PAR²Net: End-to-end panoramic image reflection removal. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45(10):12192–12205
- Huang T, Zheng Y, Yu Z, Chen R, Li Y, Xiong R, Ma L, Zhao J, Dong S, Zhu L, et al (2023) 1000× faster camera and machine vision with ordinary devices. *Engineering* 25:110–119
- Huynh-Thu Q, Ghanbari M (2008) Scope of validity of psnr in image/video quality assessment. *Electronics Letters* 44(13):800–801
- Kong N, Tai YW, Shin SY (2012) A physically-based approach to reflection separation. In: Proc. of Computer Vision and Pattern Recognition
- Lei C, Chen Q (2021) Robust reflection removal with reflection-free flash-only cues. In: Proc. of Computer Vision and Pattern Recognition
- Lei C, Huang X, Zhang M, Yan Q, Sun W, Chen Q (2020) Polarized reflection removal with perfect alignment in the wild. In: Proc. of Computer Vision and Pattern Recognition
- Levin A, Weiss Y (2007) User assisted separation of reflections from a single image using a sparsity prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(9):1647–1654
- Li C, Yang Y, He K, Lin S, Hopcroft JE (2020a) Single image reflection removal through cascaded refinement. In: Proc. of Computer Vision and Pattern Recognition
- Li R, Qiu S, Zang G, Heidrich W (2020b) Reflection separation via multi-bounce polarization state tracing. In: Proc. of European Conference on Computer Vision
- Li Y, Brown MS (2013) Exploiting reflection change for automatic reflection removal. In: Proc. of International Conference on Computer Vision
- Li Y, Brown MS (2014) Single image layer separation using relative smoothness. In: Proc. of Computer Vision and Pattern Recognition
- Liu YL, Lai WS, Yang MH, Chuang YY, Huang JB (2020) Learning to see through obstructions. In: Proc. of Computer Vision and Pattern Recognition
- Lyu Y, Cui Z, Li S, Pollefeys M, Shi B (2019) Reflection separation using a pair of unpolarized and polarized images. In: Proc. of Advances in Neural Information Processing Systems
- Lyu Y, Cui Z, Li S, Pollefeys M, Shi B (2023) Physics-guided reflection separation from a pair of unpolarized and polarized images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45(2):2151–2165
- Ma D, Wan R, Shi B, Kot AC, Duan LY (2019) Learning to jointly generate and separate reflections. In: Proc. of International Conference on Computer Vision
- Nandoriya A, Elgharib M, Kim C, Hefeeda M, Matusik W (2017) Video reflection removal through spatio-temporal optimization. In: Proc. of International Conference on Computer Vision
- Nayar SK, Fang XS, Boulton T (1997) Separation of reflection components using color and polarization. *International Journal of Computer Vision* 21(3):163–186
- Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, et al (2019) Pytorch: An imperative

- style, high-performance deep learning library. In: Proc. of Advances in Neural Information Processing Systems
- Schechner YY, Kiryati N, Basri R (2000) Separation of transparent layers using focus. *International Journal of Computer Vision* 39(1):25–39
- Sheinin M, Schechner YY, Kutulakos KN (2017) Computational imaging on the electric grid. In: Proc. of Computer Vision and Pattern Recognition
- Shi W, Caballero J, Huszár F, Totz J, Aitken AP, Bishop R, Rueckert D, Wang Z (2016) Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proc. of Computer Vision and Pattern Recognition
- Shi X, Chen Z, Wang H, Yeung DY, Wong WK, Woo Wc (2015) Convolutional lstm network: A machine learning approach for precipitation nowcasting. Proc of Advances in Neural Information Processing Systems
- Shih Y, Krishnan D, Durand F, Freeman WT (2015) Reflection removal using ghosting cues. In: Proc. of Computer Vision and Pattern Recognition
- Simon C, Kyu Park I (2015) Reflection removal for in-vehicle black box videos. In: Proc. of Computer Vision and Pattern Recognition
- Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556
- Wan R, Shi B, Hwee TA, Kot AC (2016) Depth of field guided reflection removal. In: Proc. of International Conference on Image Processing
- Wan R, Shi B, Duan LY, Tan AH, Gao W, Kot AC (2018a) Region-aware reflection removal with unified content and gradient priors. *IEEE Transactions on Image Processing* 27(6):2927–2941
- Wan R, Shi B, Duan LY, Tan AH, Kot AC (2018b) CRRN: Multi-scale guided concurrent reflection removal network. In: Proc. of Computer Vision and Pattern Recognition
- Wan R, Shi B, Li H, Duan LY, Tan AH, Kot AC (2019) CoRRN: Cooperative reflection removal network. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42(12):2969–2982
- Wan R, Shi B, Li H, Hong Y, Duan LY, Kot AC (2023) Benchmarking single-image reflection removal algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45(2):1424–1441
- Wang Z, Simoncelli EP, Bovik AC (2003) Multiscale structural similarity for image quality assessment. In: Proc. of Asilomar Conference on Signals, Systems & Computers
- Wei K, Yang J, Fu Y, Wipf D, Huang H (2019) Single image reflection removal exploiting misaligned training data and network enhancements. In: Proc. of Computer Vision and Pattern Recognition
- Wells EF, Bernstein GM, Scott BW, Bennett PJ, Mendelson JR (2001) Critical flicker frequency responses in visual cortex. *Experimental Brain Research* 139(1):106–110
- Wu S, Xu J, Tai YW, Tang CK (2018) Deep high dynamic range imaging with large foreground motions. In: Proc. of European Conference on Computer Vision
- Yan Q, Gong D, Shi Q, Hengel Avd, Shen C, Reid I, Zhang Y (2019) Attention-guided network for ghost-free high dynamic range imaging. In: Proc. of Computer Vision and Pattern Recognition
- Yang Y, Ma W, Zheng Y, Cai JF, Xu W (2019) Fast single image reflection suppression via convex optimization. In: Proc. of Computer Vision and Pattern Recognition
- Zhang R, Isola P, Efros AA, Shechtman E, Wang O (2018a) The unreasonable effectiveness of deep features as a perceptual metric. In: Proc. of Computer Vision and Pattern Recognition
- Zhang X, Ng R, Chen Q (2018b) Single image reflection separation with perceptual losses. In: Proc. of Computer Vision and Pattern Recognition
- Zheng Q, Shi B, Chen J, Jiang X, Duan LY, Kot AC (2021) Single image reflection removal with absorption effect. In: Proc. of Computer Vision and Pattern Recognition
- Zhong H, Hong Y, Weng S, Liang J, Shi B (2024) Language-guided image reflection separation. arXiv preprint arXiv:240211874