https://doi.org/10.1007/s11704-025-50966-0

RESEARCH ARTICLE

PVDD: a practical benchmark dataset and network for video denoising

Xiaogang XU[1*], Yitong YU[2*], Nianjuan JIANG[3], Jiafei WU[4]✉, Bei YU[1], Jiangbo LU[3], Jiaya JIA[5]✉

1. Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong 999077, China
2. Research and Development Department, Kuaishou Technology, Beijing 100085, China
3. SmartMore Technology, Shenzhen 518000, China
4. Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong 999077, China
5. Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong 999077, China

Received July 5, 2025; accepted November 24, 2025

E-mail: jcjiafeiwu@gmail.com; jia@cse.ust.uk. * These authors contributed equally to this work.

© The Author(s) 2025. This article is published with open access at link.springer.com and journal.hep.com.cn

Abstract

To facilitate video denoising research, we have developed a comprehensive dataset known as the "Practical Video Denoising Dataset" (PVDD), which comprises a total of 200 pairs of noisy-clean dynamic videos, available in both sRGB and RAW formats. In comparison to the existing datasets which are characterized by limited motion information, PVDD stands out by encompassing dynamic scenes that exhibit diverse and natural motion patterns. Different from datasets using primarily Gaussian or Poisson distributions to synthesize noise in the sRGB domain, PVDD synthesizes realistic noise from the RAW domain with a physically meaningful sensor noise model followed by ISP processing. Moreover, we also propose a new video denoising framework, called Recurrent Video Denoising Transformer (RVDT), which can achieve SOTA performance on PVDD and other current video denoising benchmarks. RVDT consists of both spatial and temporal transformer blocks to conduct denoising with long-range operations on the spatial dimension and long-term propagation on the temporal dimension. In particular, RVDT exploits the attention mechanism to implement the bi-directional feature propagation with both implicit and explicit temporal modeling. Extensive experiments demonstrate that 1) models trained on PVDD achieve superior denoising performance on many challenging real-world videos, outperforming models trained on other existing datasets; 2) in instances where models are trained on the same dataset, our proposed RVDT exhibits superior denoising capabilities when juxtaposed with alternative network architectures.

Keywords

video denoising dataset; dynamic scenes; recurrent video denoising transformer

1 Introduction

Signal degradation caused by the sensor noise is a pervasive issue encountered during the acquisition of video data, often leading to a degradation in visual quality. Therefore, video denoising is a task of significant importance. Deep-learning-based denoising techniques have emerged in recent years [1]. These methods learn the mapping from the noisy image/video domain to the clean image/video domain [2], while the corresponding performance heavily depends on the characteristic of the training data.

Although practical image denoising datasets have been introduced to the community, their scope remains relatively confined, particularly with regard to the diversity in motion (one example in Fig. 1). Directly employing them for the video denoising task generally results in the lack of variety in temporal space and thus cannot provide sufficient training features for challenging video processing tasks. In light of the existing deficiency in more comprehensive and proficient video denoising datasets, there arises an imperative need for the development of datasets that can better fulfill these requirements.

There are two kinds of video denoising datasets currently. The first kind collects video pairs. It creates clean video frames by averaging neighboring noisy frames [3]. It requires video frames to be either static or captured at discrete time instances, as shown in Fig. 2(b), and cannot provide training data with complex natural motion. The second group, e.g., DAVIS [4], creates noisy-clean pairs with added synthetic noise. It first collects clean videos and introduces noise (e.g., Gaussian noise) in the sRGB domain. The added noise may not be the same or even similar to that produced by cameras, considering complex distributions of real noise [5].

In addition, as explained in [5], shot noise in RAW format can be modeled by Poisson distributions, while read noise typically adheres to Gaussian distributions. Recent work [6] further shows that the

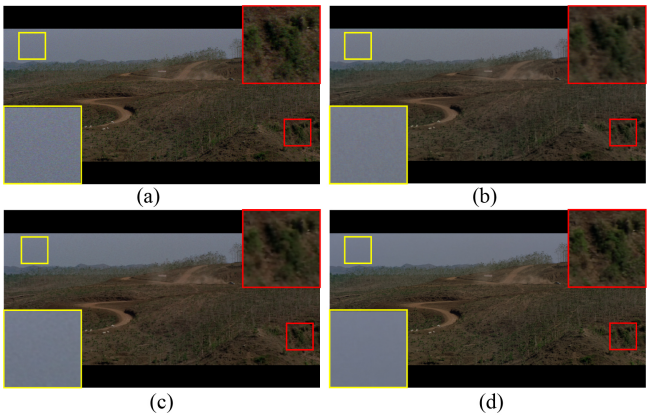


Fig. 1 Models trained on our PVDD yield superior performance on real-world noisy videos. (a) An example frame of a challenging video with the complicated noise pattern that never appears in the training set. (b)–(d) show output of FastDVDnet [7] trained on different datasets, (b) CRVD; (c) DAVIS; (d) PVDD. The model trained on our PVDD achieves the best denoising performance (yellow boxes) and preserves finer details without introducing blur and noise (red boxes). (a) Inupt; (b) FastDVDnet trained on CRVD; (c) FastDVDnet trained on DAVIS; (d) FastDVDnet trained on PVDD

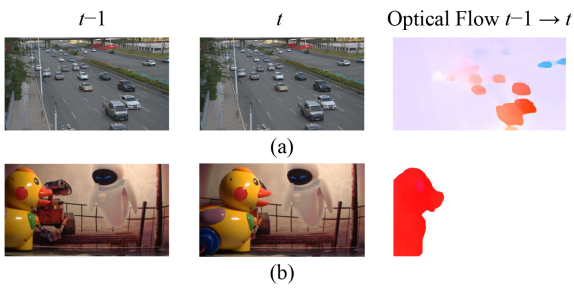


Fig. 2 Example frames with their optical flows from PVDD and CRVD. (a) The frames from PVDD are captured in video mode, containing complex natural motion. (b) The frames from CRVD are captured at discrete time instances and only contain simple and overly large frame-to-frame foreground motion

denoising networks trained with the Poisson-Gaussian noise distribution assumption achieve comparable results with networks trained on data degraded by real noise in the RAW domain. However, no existing datasets take this important fact into consideration yet when collecting data.

In this paper, we construct a new powerful dataset, Practical Video Denoising Dataset (PVDD), by collecting clean videos with complicated natural motion, followed by the corruption of realistic noise in RAW format to form RAW video pairs. To facilitate broader applicability and utility, we further process these RAW pairs through a calibrated ISP processing pipeline, thereby generating the corresponding sRGB video pairs. It contains 200 noisy-clean RAW/sRGB video pairs with rich dynamics, as shown in Fig. 2(a), which greatly diversify temporal local motion when used in the system training.

In addition to the ISP modules utilized in previous work [8], we have introduced two additional modules into our pipeline, namely, a color temperature module and a filmic tone-mapper module to better approximate the camera ISP pipeline. Owing to the calibration of the ISP modules, the synthetic noise generated through this pipeline retains a high degree of realism, ensuring that the added noise in both the RAW domain and sRGB domain adheres closely to a realistic noise model. The practical utility of PVDD in facilitating the video denoising tasks is appropriately demonstrated in Fig. 1.

Besides the PVDD, we propose a novel video denosing framework in this paper, so-called recurrent Video Denoising Transformer (RVDT). RVDT consists of spatial and bi-directional temporal transformer blocks. The spatial transformer blocks are concatenated with convolutional blocks to perform spatial-varying long-range and short-range operations. The bi-directional temporal transformer blocks are built with both forward and backward blocks, where novel attention mechanisms are designed to achieve feature propagation from past and future frames. Further, long-range and short-range dependencies are simultaneously achieved in the transformers' feed-forward modules, enhancing the corresponding representation capacity. Experiments are conducted on different datasets to show the superiority of the proposed RVDT to existing video denoising frameworks in both sRGB and RAW domains.

Our dataset and code have been released for research, and are publicly available. In conclusion, our contribution is threefold.

- We propose an information-rich dataset, named PVDD, for video denoising in RAW and sRGB domains. It incorporates complex natural motion and realistic noise.
- Our PVDD serves as a practical benchmark dataset to evaluate video denoising networks and has the potential to further promote research along this line. We conduct extensive experiments with representative deep learning architectures including both quantitative and qualitative evaluation. They manifest the superiority of the proposed dataset and the new degradation model.
- We propose a novel video denoising framework RVDT with newly designed bi-directional transformer blocks. Extensive experiments are conducted on PVDD and existing video denoising benchmarks, demonstrating the SOTA performance of RVDT.

2 Related work

2.1 Video denoising datasets

Unlike image denoising datasets, e.g., SIDD [8], RENOIR [9], etc., video denoising datasets are far less, but still can be categorized into two kinds. The first kind collects video pairs by utilizing long exposure for clean videos and short exposure for noisy videos, e.g., SMID [10], or creating clean video frames by averaging static neighboring noisy frames, e.g., CRVD [3]. These datasets contain only static or sparse frames captured in the temporal dimension. The second kind creates noisy-clean pairs by noise synthesis, e.g., DAVIS [4]. It adds noise (e.g., Gaussian noise) to clean videos.

2.2 Video denoising methods

Although traditional video denoising methods perform well in some scenes and can be augmented by neural networks [11], they might

fail in complex scenarios. This has led to the rise of data-driven, neural-network-based denoising techniques.

Most video denoising techniques estimate pixel-wise motions between adjacent frames via, for example, patch-wise motion estimation [12], optical flow [13,14], kernel-prediction networks [15], and deformable convolutions [16,17]. Recently, Lindner et al. [18] proposed using a network to estimate inter-frame correspondences, which are then used to predict frame-wise aggregation coefficients to guide the subsequent video denoising process. Li et al. [19] and Zhang et al. [20] designed spatial-temporal shift modules to capture similar correspondences for multi-frame aggregation.

However, because accurate per-pixel motion estimation is challenging in some real-world scenes, methods featuring implicit motion modeling have been proposed. These approaches avoid the need for explicit modeling of point-wise shifts at both the pixel and feature levels. These approaches either take simultaneously multiple frames as input, which are jointly processed by the model [7,21], or use information from previous frames as the additional input [22,23]. Several transformer-based methods have also been developed within this category and have demonstrated strong performance. For instance, Yue et al. [24] employed a transformer architecture with multi-branch spatial and temporal attention modules. However, existing approaches ignore to decouple the denoising process into distinct spatial and temporal dimensions. Furthermore, more efficient attention mechanisms are needed to enhance the propagation of relevant information from adjacent frames.

Most of the aforementioned methods are supervised. The performance of these supervised approaches on real-world noisy videos is highly dependent on the quality of their training data. Consequently, effective video denoising requires a high-quality dataset comprising diverse motions and realistic noise patterns. In contrast, unsupervised methods [25,26] are generally less effective and are not the focus of this work.

3 Proposed PVDD dataset

Our PVDD, which is the main contribution of this work, contains 200 videos in both sRGB and RAW formats (captured with a NIKON Z7 II mirrorless camera), and each video clip spans 8s to 15s.

3.1 Data collection setup

For the RAW video collection, we use a NIKON Z7 II mirrorless camera and an Atomos Ninja V recording monitor. The monitor is attached to the camera by an HDMI interface and is utilized to transport, encode and store the continuous video stream into an Apple ProRes RAW video (Apple ProRes RAW is a well-recognized video codec standard directly applied to the camera sensor's data). All videos are captured in resolution 1,920×1,080@25p and 1,920×1,080@60p format. The captured RAW videos can be further converted into sRGB videos by a calibrated camera ISP.

We capture the video by hand-held shooting to introduce additional camera motion with more complexity than conventional rig-based capture. We control the camera motion slowly to avoid visible motion jittering or motion blur. Camera setting, such as aperture, ISO, and focal length, is adjusted to minimize the noise level of the captured videos. For example, the aperture size is usually set between 4 and 6 to increase the amount of incident light received by the sensor, and the ISO is set below 400 to eliminate random noise. The focal length is adjusted by camera's built-in auto-focus module.

3.2 Realistic noise synthesis

3.2.1 RAW noise synthesis

During collection, we capture one clean video for every scene, as shown in Fig. 3. We obtain a set of readout noise parameters σ_r and shot noise parameters σ_s at different ISO settings, which are pre-calibrated and recorded in the camera by the manufacturer.

To produce noisy RAW videos, we first randomly select (σ_r^K, σ_s^K) corresponding to a particular ISO K and then add Gauss-Poisson noise accordingly to the clean RAW videos. Suppose the clean RAW video is denoted as Y. The process to obtain noisy observation X can be expressed by

$$X_c \sim \mathcal{P}((\sigma_{s,c}^K)^2 Y_c) + \mathcal{N}(0, (\sigma_{r,c}^K)^2), \tag{1}$$

where $c \in \{\mathrm{R}, \mathrm{G}, \mathrm{B}\}$, $\mathcal{P}$ represents the Poisson distribution with mean Y_c and variance $(\sigma_{s,c}^K)^2 Y_c$, and $\mathcal{N}$ denotes the Gaussian

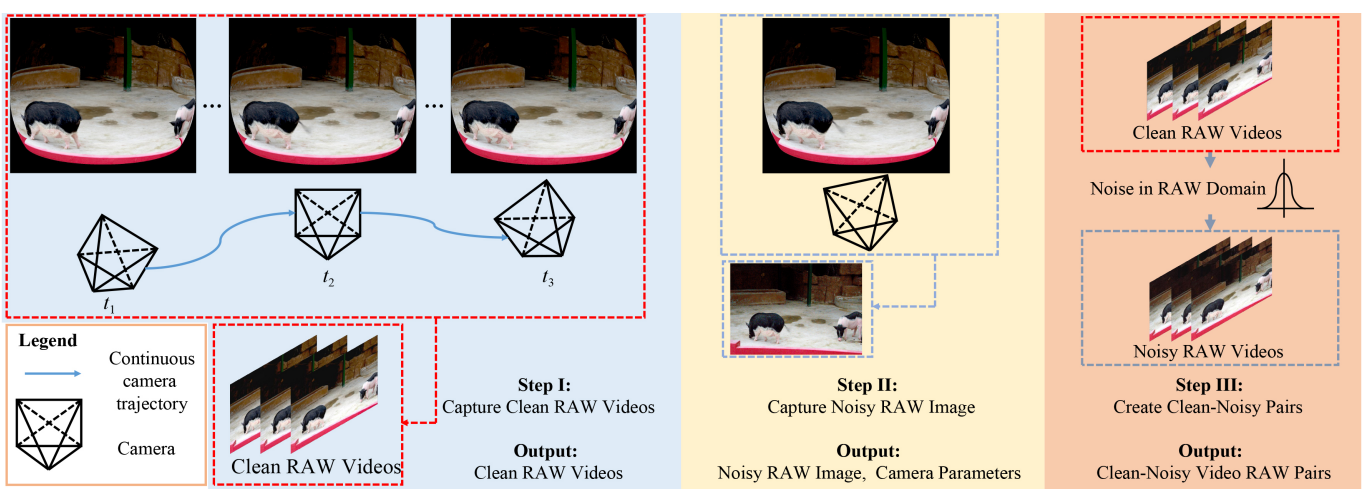


Fig. 3 Overview of the proposed pipeline to create dynamic noisy-clean video pairs in RAW format

distribution with zero mean and variance $(\sigma_{r,c}^K)^2$.

The noise pattern in X is found close to real noises since 1) the noise distribution in the RAW domain can be well approximated by the Gauss-Poisson distribution [5]; 2) noise distribution model parameters are obtained from the actual calibration of the camera.

3.2.2 sRGB video pairs

To obtain noisy-clean video pairs in the sRGB domain, we adopt an integrated camera ISP pipeline as shown in Fig. 4(a) (more examples can be seen in Fig. 5). Camera parameters in the ISP for each clean RAW video are provided by one additional noisy RAW image as shown in Fig. 3. Our ISP is built upon the ISP utilized in [8] and our improvement is the following.

In order to better approximate a real camera ISP, we 1) compute the color temperature and serve it as a hyper-parameter to assist the color space transformation module with an accurate color transformation matrix; 2) choose an ACES tonemapper [27] as our tone mapping function and apply it to ProPhoto RGB color space, which has a wide color gamut. The example in Figs. 4(b) and 4(c) demonstrates the effectiveness of our improvement.

In particular, the white balance module estimates an illumination value in terms of a color temperature, and the color space transformation I module (which maps the sensor-specific RAW RGB colors to a canonical space such as CIE XYZ) uses this estimated color temperature value to interpolate between two factory-calibrated color-correction matrices [28] to obtain an accurate matrix. Moreover, as mentioned in [29], the color transformation I module converts the cameraRGB space into CIE XYZ space, and then into ProPhoto RGB space. The color transformation II module converts ProPhoto RGB space into sRGB space.

3.3 Evaluation dataset

For the quantitative evaluation of our noise synthesis and the superiority over existing video denoising datasets, we also collect an evaluation dataset, namely Static15, consisting of 15 different static scenes. This dataset is produced by capturing static noisy videos by the tripod-mounted NIKON Z7 II camera, whereas the corresponding clean video is derived by averaging neighboring noisy frames [8]. Especially, we average 50 consecutive noisy frames to obtain a clean frame whose SNR value is 50.87dB proving clean frames' quality.

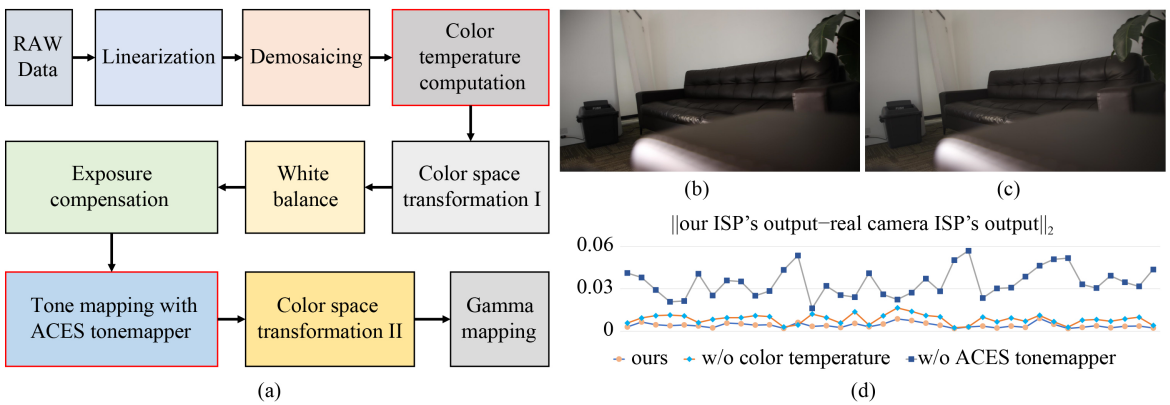


Fig. 4 (a) Our ISP to produce sRGB videos. (b) Comparison between sRGB images produced by our ISP and the real camera ISP. (c) Average normalized per-pixel. (d) Average normalized per-pixel L_2 distance between our ISP output and that of real camera ISP across 38 different scenes. Removing either the color temperature computation module or ACES tonemapper increases this distance

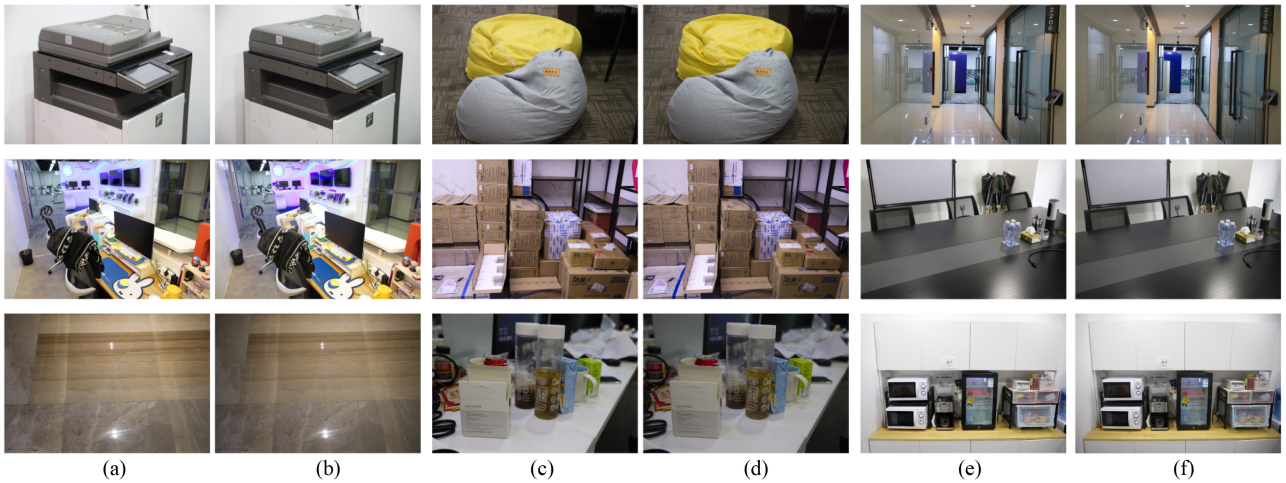


Fig. 5 The visualization to illustrate that our ISP's outputs are highly similar to the real camera ISP's output. (a) Real camera ISP's output; (b) our ISP's output; (c) real camera ISP's output; (d) our ISP's output; (e) real comera ISP's output; (f) out ISP's output

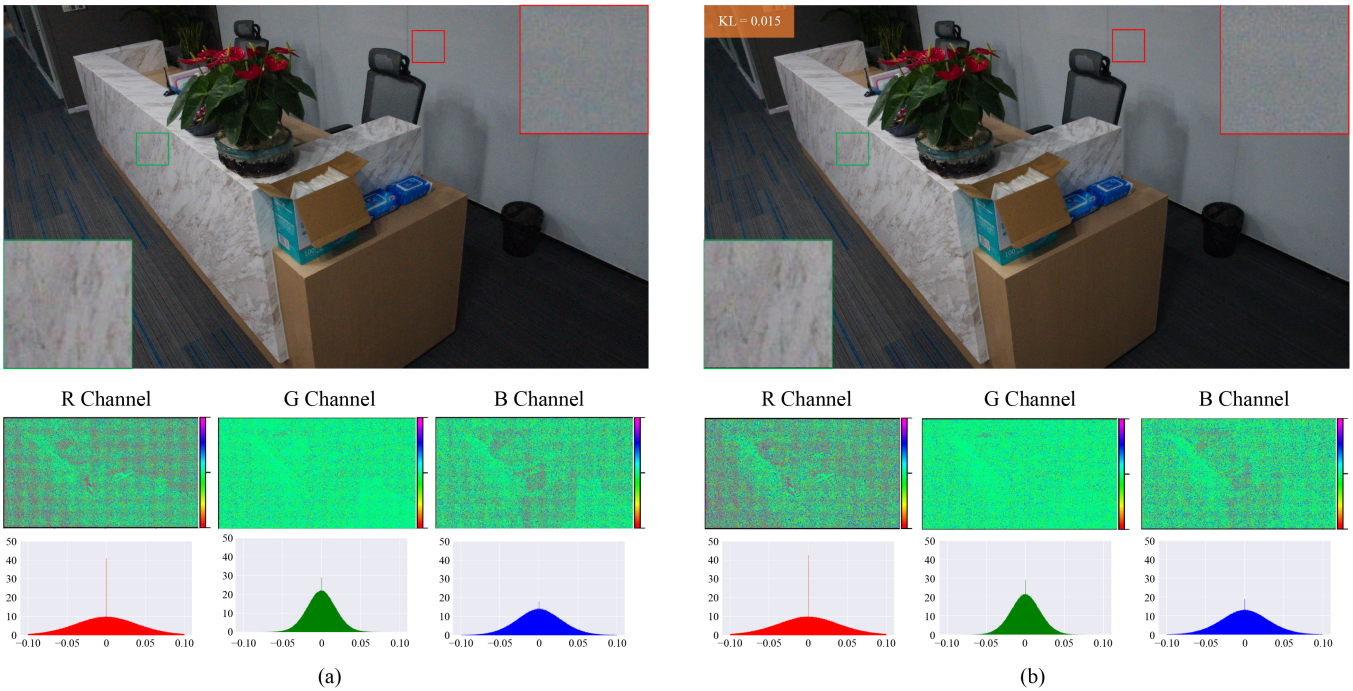


Fig. 6 Illustration of similarity between the real and synthetic noise distributions in sRGB domain. (a) Real noisy image; (b) synthetic noisy image

We already generate noise on clean frames, and thus can compare it with the captured noise. As shown in Fig. 6, the noise distribution of the synthetic noisy frame is surprisingly close to the actual captured ones in the sRGB domain. We compute the KL divergence between synthesized and captured real noise in the sRGB domain across all video clips on Static15 (noise value range [0, 1]). The average normalized per-pixel KL divergence is 0.034 over sRGB and 0.041 over RAW, indicating that the synthesized noise is realistic under the same camera setting (a KL divergence value below 0.05 is acceptable [30]).

3.4 The distribution of data in PVDD

PVDD has large-scale and diverse videos. The proportion of indoor and outdoor videos in our dataset can be viewed in Fig. 7(a). The weather condition includes the sunny day and the overcast day with the distribution as shown in Fig. 7(b). The light condition includes daytime and night time with the distribution as shown in Fig. 7(c). The time length of all videos in our dataset ranges from 8s to 15s.

4 Recurrent video denoising transformer

Besides the newly built video denoising dataset, we further propose a novel video denoising framework with the spatial transformer and bi-directional temporal recurrent video transformer structure. Our framework is called Recurrent Video Denoising Transformer (RVDT). In this section, we first provide an overview of RVDT in Subsection 4.1, and then we describe our newly designed attention blocks and feed-forward modules of RVDT in Subsections 4.2 and 4.3, respectively.

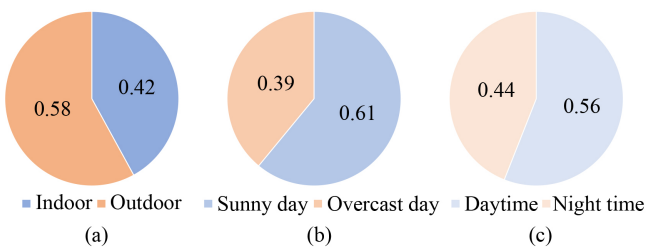


Fig. 7 Distribution: different types of videos in our dataset. (a) Indoor vs. outdoor; (b) sunny day vs. overcast day; (c) day time vs. night time

4.1 Model architecture

Video denoising is a challenging task since it needs to aggregate information from not only the spatial dimension but also the temporal dimension. Existing approaches can be divided into two branches, window-based and recurrent-based. Window-based methods, e.g., FastDVDnet [7] and RViDeNet [3], typically take a video clip as the input and output the centered denoised frame. Although several models have been deliberately designed, recent works [31] have shown that they are computationally inefficient when the window size is large. Due to the limited input frame number, these window-based methods can only handle the propagation of short-term information among video frames.

On the other hand, recurrent-based methods are designed with efficiency in mind and are capable of processing long-range information, benefiting from propagating latent features from one frame to the next sequentially. One of the most important factors for this category is how to merge the adjacent latent features, since merging inaccurately would lead to information error being accumulated. Some approaches deal with temporal merging by optical flow alignment, however, for the video denoising task, estimating motion on noise patch by flow-based methods always suffers from distortions and artifacts and yields an inferior

performance. Others, e.g., EMVD [22], attempt to generate a spatial attention map to guide their temporal merging process, which is efficient for dealing with static scenes, while not good at processing with motion scenarios.

Inspired by the tremendous success of the vision transformer's attention component in building non-local spatial correlations [32], we in this paper propose to employ the transformer for formulating inter-frame temporal correlations in the recurrent-based video denoising framework. As shown in Fig. 8(a), without loss of generality, suppose we aim to complete the denoising process for the i-th frame X_i. The feature of X_i, $\bar{f}_i$, is extracted with the spatial CNN and transformer blocks, and it is merged with features from temporally bi-directional features ($\hat{f}^F_{i-1}$ or $\hat{f}^B_{i+1}$) via designed forward/backward temporal transformer blocks whose details are shown in Fig. 8(b). With the non-local attention in the spatial dimension and the bi-directional attention in the temporal dimension, our recurrent-based video denoising framework is different from current video denoising approaches, can handle various motions efficiently, and achieve the SOTA denoising performance on all datasets (proved by the experiments in Section 5).

The denoising processing in RVDT consists of two stages. In the first stage, as shown in Fig. 8(a), the feature of input frame $X_i \in \mathbb{R}^{3\times H\times W}$ is extracted through a CNN-based encoder C_S as

$$f_i = C_S(X_i), \tag{2}$$

where C_S consists of a shallow UNet and two convolutional layers with down-sampling operations, and $f_i \in \mathbb{R}^{C_{in}\times\frac{H}{4}\times\frac{W}{4}}$. The features extracted with a CNN-based encoder mainly involve local properties, while the denoising task also requires non-local operations for better performance. Thus, we apply spatial 2D transformer blocks for f_i, utilizing the corresponding attention computation module to find suitable non-local areas for the denoising of all image regions. The procedure can be formulated as

$$\bar{f}_i = \mathcal{T}_S(f_i), \tag{3}$$

where $\mathcal{T}_S$ denotes the concatenation of transformer blocks for spatial denoising. The first stage acts as a spatial pre-denoiser, which benefits the temporal denoising in the second stage.

After acquiring spatially denoised result $\bar{f}_i$ from the first stage, it will be merged with bi-directional features ($\hat{f}^F_{i-1}$ or $\hat{f}^B_{i+1}$) which are formulated sequentially with the forward and backward temporal transformer blocks, as

$$\begin{aligned}\hat{f}^F_i &= \mathcal{T}^F_T(\bar{f}_i, \hat{f}^F_{i-1}),\\ \hat{f}^B_i &= \mathcal{T}^B_T(\bar{f}_i, \hat{f}^B_{i+1}),\end{aligned} \tag{4}$$

where $\hat{f}^F_i$ and $\hat{f}^B_i$ represent the forward and backward processed features, respectively. $\mathcal{T}^F_T$ and $\mathcal{T}^B_T$ denote the forward and backward temporal transformer blocks. Thus, the inter-frame correlations are built between $\bar{f}_i$ and its adjacent features $\hat{f}_{i\pm1}$.

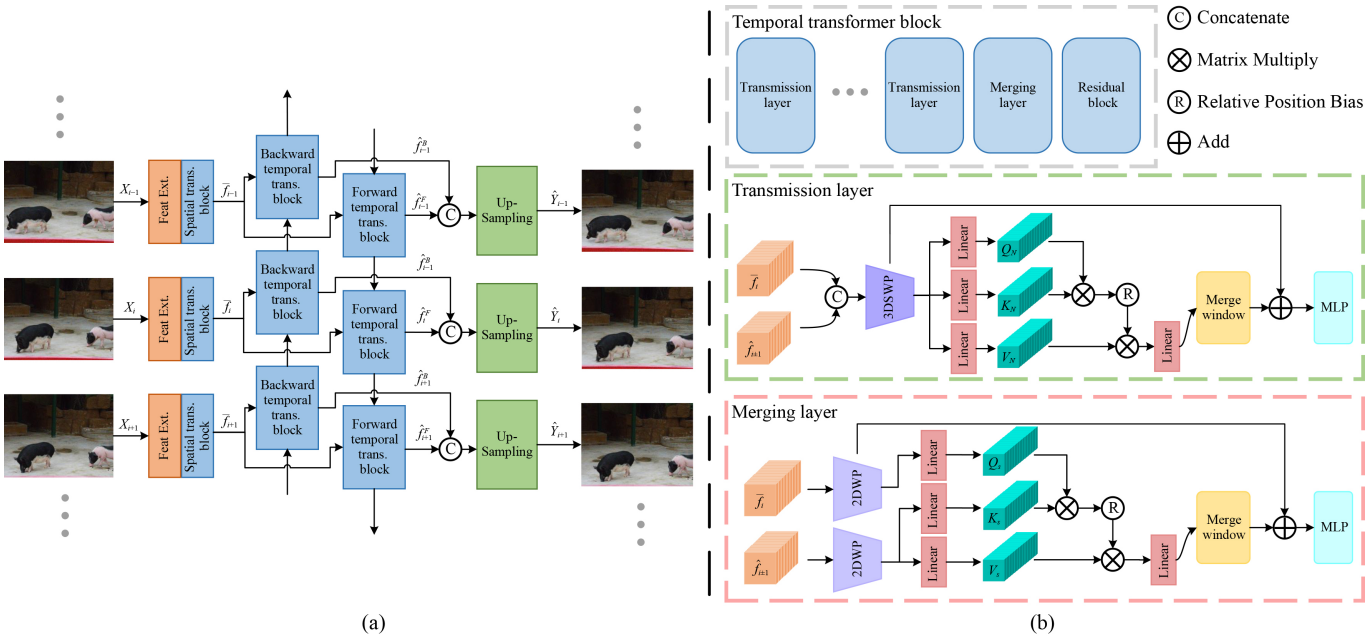
Finally, a decoder with up-sampling layers are employed to obtain the final denoised outputs, as

$$\hat{Y}_i = \mathcal{U}(\hat{f}^F_i, \hat{f}^B_i), \tag{5}$$

where $\mathcal{U}$ is the decoder whose up-sampling layers are implemented with convolutions and pixel-shuffle [33], and $\hat{Y}_i$ is the denoised version of X_i.

4.2 Temporal transformer block

The details of the temporal transformer blocks will be described in this section. There are two kinds of strategies to implement the inter-frame attention via the transformer. The first category employs the 3D transformer structure by conducting attention computations on



(a) (b)

Fig. 8 (a) The overview of the proposed RVDT, which consists of both spatial and bi-directional transformer blocks. (b) The details of the proposed bi-directional transformer blocks, which consist of both "Transmission layer" and "Merging layer"

3D tokens to propagate information in both spatial and temporal dimensions implicitly. Such transformer layers are called "transmission layers". The second approach views the bi-directional features as the references, and utilizes the 2D transformer to explicitly construct the relation between the current frame's feature and the references. Such layers are called "merging layers" that can merge neighboring frames' features. The temporal transformer blocks in RVDT consist of these two kinds of layers. In this way, the mutual feature propagation among neighboring frames is completed before the merge operation, and we find such a strategy can achieve better denoising performance.

Without loss of generality, we describe the details of the forward temporal transformer blocks, and the backward temporal transformer blocks are the same by changing F to B, $i-1$ to $i+1$. Suppose there are N layers in $\mathcal{T}_T^F$, i.e., $\mathcal{T}_T^F=\{\mathcal{T}_{T,1}^F,...,\mathcal{T}_{T,N}^F\}$. Among them, $N-1$ layers are transmission layers, and the last one is the merging layer. Thus, for the input of $\bar{f}_i$ and $\hat{f}_{i-1}^F$, there are $N-1$ intermediate outputs, as $\{\bar{f}_{i,1},...,\bar{f}_{i,N-1}\}$ and $\{\hat{f}_{i-1,1}^F,...,\hat{f}_{i-1,N-1}^F\}$, respectively. Thus, the procedure of the transmission layers is

$$\begin{cases}[\bar{f}_{i,s},\hat{f}_{i-1,s}^F]=\mathcal{T}_{T,s}^F(\bar{f}_i,\hat{f}_{i-1}^F), & s=1,\\ [\bar{f}_{i,s},\hat{f}_{i-1,s}^F]=\mathcal{T}_{T,s}^F(\bar{f}_{i,s-1},\hat{f}_{i-1,s-1}^F), & s\in[2,N-1].\end{cases} \tag{6}$$

The features obtained from the transmission layers are processed with the merging layer to acquire the final output of the temporal transformer blocks, as

$$\hat{f}_i^F=\mathcal{T}_{T,N}(\bar{f}_{i,N-1},\hat{f}_{i-1,N-1}^F). \tag{7}$$

The implementation details of the transmission layers and the merging layer are described in the following.

4.2.1 Transmission layer

The overall architecture of the transmission layer is shown in the middle part of Fig. 8(b). For the input of $[\bar{f}_{i,s},\hat{f}_{i-1,s}^F], s\in[1,N-1]$, they are concatenated and processed with 3D window partition layers (3DWP) to serve as the input of 3D transformer layers.

For the window-based attention computation in the s-th transmission layer, the query Q_s, key K_s, value V_s vectors are obtained via linear projections, as

$$\begin{aligned}h_s&=\text{3DWP}([\bar{f}_{i,s},\hat{f}_{i-1,s}^F]),\\ Q_s&=W_q^s\cdot\text{LN}(h_s),\\ K_s&=W_k^s\cdot\text{LN}(h_s),\\ V_s&=W_v^s\cdot\text{LN}(h_s),\end{aligned} \tag{8}$$

where $[\cdot,\cdot]$ denotes the concatenation operation, LN represents the layer normalization, W_q^s, W_k^s, and W_v^s are matrices of learnable parameters for linear projections. The attention output can be written as

$$\text{Attention}(Q_s,K_s,V_s)=\text{SoftMax}(Q_sK_s^{\text{T}}/\sqrt{d_s}+B_s)V_s, \tag{9}$$

where d_s is for normalization in the s-th layer, and B_s is the relative position bias. The feature tokens are updated as

$$h_s'=h_s+\text{Attention}(Q_s,K_s,V_s). \tag{10}$$

Moreover, in the feed-forward module of the transformer, we adopt a channel-spatial attention MLP (CSA-MLP), which is denoted as $\mathcal{M}$ and will be described in the next section. And the feature tokens after the 3D window reverse layers (3DWR) can formulate the output of the ith transmission layer. The overall process can be formulated as

$$[\bar{f}_{i,s+1},\hat{f}_{i-1,s+1}^F]=\text{3DWR}(\mathcal{M}(\text{LN}(h_s'))+h_s'). \tag{11}$$

4.2.2 Merging layer

Within the merging layer, the input feature $\bar{f}_{i,N-1}$ and $\hat{f}_{i-1,N-1}^F$ are processed with the 2D window partition layer (2DWP) separately to obtain the corresponding feature tokens. The target of the merging layer is to merge the bi-directional features, i.e., $\hat{f}_{i-1,N-1}^F$, to the feature of the current frame, i.e., $\bar{f}_{i,N-1}$. This is achieved by the inter-frame relationship computed in window-based attention module in the merging layer, which can be written as

$$\begin{aligned}m_N&=\text{2DWP}(\bar{f}_{i,N-1}),\\ n_N&=\text{2DWP}(\hat{f}_{i-1,N-1}^F),\\ Q_N&=W_q^N\cdot\text{LN}(m_N),\\ K_N&=W_k^N\cdot\text{LN}(n_N),\\ V_N&=W_v^N\cdot\text{LN}(n_N),\end{aligned} \tag{12}$$

where W_q^N, W_k^N, and W_v^N represent the projection matrices in the merging layer's attention module. In this way, each feature token can be merged with the information from the temporally bi-directional features, as

$$m_N'=m_N+\text{Attention}(Q_N,K_N,V_N). \tag{13}$$

The feature acquired from the attention module is processed with the CSA-MLP and 2D window reverse layer (2DWR) to obtain the final output, as

$$\hat{f}_i^F=\text{2DWR}(\mathcal{M}(\text{LN}(m_N'))+m_N'). \tag{14}$$

4.3 Channel-spatial attention MLP block

Generally, the feed-forward module in the transformer is formulated as an MLP [34], enhancing the representation capacity of feature tokens. Recently, more and more works have noticed the drawbacks of modeling local relations via an MLP in the feed-forward module, and extra modules are introduced to enhance the local property, e.g., the CNN blocks [35]. In this work, we further propose a channel-spatial attention MLP block to formulate the feed-forward module. As shown in Fig. 9, the key points of this design are twofold: 1) we keep the original MLP to capture the long-range dependencies, 2) we incorporate the attention-based convolutional layers to capture short-range dependencies. Different from the existing locally enhanced MLP module [35], channel attention (CA), spatial attention (SA), and jointly spatial-channel attention (SCA) are inserted in the convolutional layers to better model the local dependencies, benefiting the denoising effects.

Given the feature z from the attention module, i.e., h_s' in Eq. (11) and m_N' in Eq. (14). The overall procedure in the CSA-MLP can be formulated as

$$\begin{aligned}\bar{z} &= \mathcal{M}_1(\mathrm{LN}(z)),\\ \bar{z} &= \mathrm{Reshape}(\bar{z}),\\ \hat{z} &= \mathrm{SCA}(\mathrm{Conv}([\mathrm{SA}(\bar{z}),\mathrm{CA}(\bar{z})])+\bar{z})+\bar{z},\\ \hat{z} &= \mathrm{Flatten}(\hat{z}),\\ \hat{z} &= \mathcal{M}_2(\hat{z})+z,\end{aligned} \tag{15}$$

where $\mathcal{M}_1(\cdot)$ and $\mathcal{M}_2(\cdot)$ are two MLPs in Fig. 9 to construct long-range relations, $\mathrm{Conv}(\cdot)$ is a convolutional layer, Reshape is the operation to turn feature tokens with the shape as $(B\times T)\times C\times(h\times w)$ into the 2D maps with the shape as $(B\times T)\times C\times h\times w$ (B is the batch size, $T\in\{1,2\}$ is the temporal size, C is the channel number, h and w are height and width, respectively), and Flatten is the reverse process of Reshape.

SA exploits the spatial relationships of features, and CA is designed to build the inter-channel dependencies of convolutional features. For the input with shape as $(B\times T)\times C\times h\times w$, the attention map from SA is $(B\times T)\times 1\times h\times w$, and from CA is $(B\times T)\times C\times 1\times 1$. The attention value is depicted with the sigmoid function. Furthermore, to better extract the content information, we jointly optimize the attention mechanisms in the spatial-channel dimensions via SCA, which is not considered by previous works. The attention map from SCA has the shape as $(B\times T)\times C\times h\times w$, which can refine the features processed with the locally enhanced convolutional layer.

4.4 Implementation details

We implement our RVDT in PyTorch, and train and test it on a PC with an NVIDIA 2080Ti GPU. The proposed RVDT is trained with the reconstruction loss between the output and the noise-free ground truth. For both sRGB and RAW domains, training patches with a size of 256×256 are extracted randomly as inputs, and we adopt common augmentations, e.g., rotation and flip. Especially for the RAW domain, we extract patches after packing one-channel Bayer RAW images into four-channel GBRG images, to avoid sampling improper patterns [36]. We use Adam optimizer with momentum set to 0.9. The batch size and learning rate are set as 1 and 1×10^{-4} for training, respectively. The RVDT model is trained until convergence.

5 Experiments

In this section, we first describe the data, networks, and metric to evaluate the effect of PVDD and RVDT in Subsections 5.1 to 5.3, respectively. We train the same network structure on different datasets to illustrate the superiority of our PVDD dataset over existing video denoising benchmarks in Subsection 5.4; we conduct experiments with different video denoising networks on the same dataset to demonstrate the effectiveness of our proposed RVDT in Subsection 5.5.

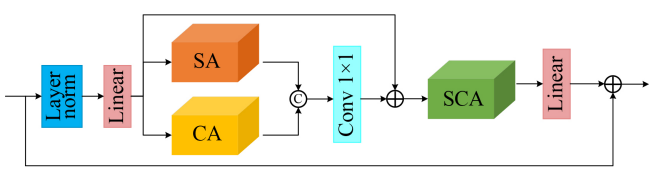


Fig. 9 The architecture of the proposed Channel-Spatial Attention MLP Blocks in transformers' feed forward stages

5.1 Datasets

We first illustrate the training and testing datasets that will be utilized in evaluating PVDD and RVDT.

5.1.1 Training data

We train representative state-of-the-art video denoising networks on both PVDD and popular benchmark datasets and compare the performance under the same network structure or dataset. Previous datasets include CRVD [3] and DAVIS [4]. Note that low-light video datasets, e.g., SMID [10] and SDSD [37] datasets, are not considered since their noisy and clean frames have different illumination values, making training difficult.

We divide all the videos in PVDD into the training collection (90%) and the testing collection (10%). As it is not necessary to employ all the frames (71,714 in total) for training, for each video in the collection, we select two or three clips (each contains 25 continuous frames) randomly to form the actual training set. Note that the random clip sampling strategy is adopted for both storage and computational efficiency. On one hand, each frame in the PVDD dataset is stored in two formats: a 12-bit RAW image for the RAW domain and an 8-bit PNG image for the sRGB domain, requiring approximately 3 MB and 2 MB per 1080 frame, respectively. Consequently, the unpruned PVDD dataset occupies around 300 GB of storage, which makes it impractical for practical deployment. In addition, the full PVDD dataset contains 71,714 frames, making it infeasible to train on every frame for multiple epochs with our limited computational resources. We observe that sampling just two to three clips (each 25 frames long) is sufficient to capture most of a video's content in PVDD, as videos are redundant in spatial-temporal dimensions.

As for CRVD, RAW data is transferred to the sRGB domain using the default ISP implemented by SID [38]. As for DAVIS, we first use the unprocessing technique [39] to turn sRGB videos into the RAW format, and add Gauss-Poisson noise in the RAW domain to form pairs. The noisy RAW videos are finally converted back to sRGB. This is a common setting adopted by [40].

5.1.2 Evaluation data

We consider evaluation settings, where noisy sRGB videos are obtained from ISP without post-processing, which is the common setting for current works [3].

Specifically, the testing dataset of PVDD has two types of data. The first is selected from the testing set of PVDD to contain only dynamic scenes. 20 dynamic clean video clips are picked from the testing set of PVDD to form Dynamic20. For each clip, three noisy versions are generated with noise levels of "Heavy", "Medium", and "Light", corresponding to ISO of 20,000, 8,000, and 2,500. The second type is Static15 (as explained in Subsection 3), which consists of 15 static videos with real camera noise. These noisy videos are captured with 1,920×1,080 resolution at 60 FPS under various ISO settings ranging from 16,000 to 25,600. The corresponding clean version is directly obtained by averaging consecutive noisy frames.

Moreover, we employ the testing dataset of CRVD [3] and DAVIS [4] to evaluate the effectiveness of our proposed video denoising dataset and network.

5.2 Denosing networks

We choose current SOTA video denoising frameworks for experiments, demonstrating the effects of our dataset and RVDT in Subsections 5.4 and 5.5, respectively. These denoising frameworks include the window-based and recurrent-based approaches, and they are released with publicly available training and evaluation codes, including VNLnet [41], FastDVDnet [7], EDVR [16], RViDeNet [3], and EMVD [22] (EMVD is utilized for the evaluation of RAW only that is the target of EMVD's original paper).

We follow the original setup proposed by the authors to train these networks, i.e., 15 consecutive frames for VNLnet, 5 for FastDVDnet and EDVR, and 3 for RViDeNet, respectively. We use the official implementation for training and evaluation, and all training configurations remain as the default ones. We train both blind (without noise level prior) and non-blind (with noise level prior) versions for each dataset-network setup. For the models with noise levels as inputs, we set the noise level map as an additional input channel for each frame. Thus, input channel numbers are different for blind and non-blind models.

5.3 Metrics

For the evaluation on datasets with ground truth (e.g., Dynamic20 and Static15), we compute the PSNR and SSIM between model output and ground truth.

5.4 Evaluation for denoising dataset

In this section, we compare PVDD with existing two video denosing datasets, including CRVD and DAVIS. We aim to prove the superiority of PVDD in training video denoising networks.

5.4.1 Evaluation strategy

To demonstrate the superiority of our framework and avoid the bias of the evaluation datasets, we set a cross-dataset evaluation strategy. For example, to demonstrate the superiority of PVDD over CRVD, we train different networks on PVDD and CRVD, respectively, and evaluate the performance of trained networks on another dataset, i.e., DAVIS. In the similar way, for the comparison between PVDD and DAVIS, we adopt CRVD for evaluation.

5.4.2 Quantitative results

As shown in the left part of Table 1 (we still employ the practical setting where networks are evaluated with the blind version in the sRGB domain), the networks trained on PVDD can be better generalized to the denoising on the DAVIS dataset than the models trained on the CRVD. For example, compared with RVDT trained on the datasets of CRVD, the same model trained on our dataset achieves more than 4 dB gain on DAVIS (in terms of PSNR), which is significant. The advantage of SSIM is also prominent. The advantage stems from the large scale of PVDD that contains natural motion and realistic noise.

Moreover, the generalization superiority is still apparent when the comparison is conducted between PVDD and DAVIS, employing the testing set of CRVD for evaluation. The results are displayed in the right part of Table 1. Models trained on PVDD can reach 32.90/0.842 (PSNR/SSIM) on average on the CRVD testing dataset, while the models trained on DAVIS can only achieve 31.06/0.811 (PSNR/SSIM) on average, verifying the usefulness of diverse motion, realistic noise, and various scenes in our PVDD dataset.

All these improvements are achieved without bias from the training data. Therefore, the model trained on our PVDD dataset yields a better denoising generalization ability.

5.4.3 Qualitative results

We provide the visual comparisons for the results in Table 1. In Fig. 10, we show the visual comparison with the cross-dataset evaluation setting. The two left columns are the results from the networks tested on CRVD, and trained on PVDD and DAVIS, respectively. Obviously, the model trained with PVDD yields more excellent visual quality. For example, in the areas of red rectangles in Fig. 10, the networks trained with PVDD generate the more pleasing result with most of the noise eliminated. Similarly, the two right columns of Fig. 10 provide the visual comparison for networks tested on DAVIS, and trained with PVDD and CRVD, respectively. It can be seen that the models trained with our PVDD give more satisfactory results than the networks trained with CRVD. The superiority can be observed in both smooth areas (most noise removed) and regions with rich textures (details preserved). The excellent generalization ability of the model trained with PVDD comes from the large-scale data and the realistic noise distribution in PVDD. Moreover, as shown in the right two columns of Fig. 10, the VNLnet trained with CRVD could have motion blurry artifacts when tested on DAVIS, since the motion distribution in CRVD is much smaller than the DAVIS with natural moves. On the other hand, our PVDD provides various motion distributions to avoid such blurry artifacts.

Table 1 Quantitative comparison for training on PVDD or CRVD/DAVIS dataset, while evaluating on DAVIS/CRVD dataset

	Eval. DAVIS (PSNR/SSIM)		Eval. CRVD (PSNR/SSIM)	
	Train. PVDD	Train. CRVD	Train. PVDD	Train. DAVIS
VNLnet	32.83/0.912	18.19/0.580	29.61/0.855	26.20/0.850
FastDVDnet	35.96/0.935	21.08/0.726	32.33/0.807	30.12/0.801
RViDeNet	37.06/0.950	33.85/0.891	32.34/0.810	33.63/0.819
EDVR	36.89/0.943	31.84/0.863	34.89/0.870	32.75/0.796
RVDT	37.30/0.950	33.11/0.870	34.99/0.870	32.59/0.789



Fig. 10 Comparisons with varying networks under the cross-dataset evaluation setting. "Noisy" is the noisy input, and "GT" is the ground truth. Models trained with PVDD can be better generalized for denoising videos in the wild that have never appeared during the training, yielding superior visual results on both the smooth area and the region with complex texture

5.4.4 Real-world denoising evaluation

In the real world, a noisy sRGB video from the ISP is very likely to be further processed with a post-processing pipeline in practice, i.e., noise in the sRGB domain is further distorted by various types of degradation, e.g., UV denoiser [42], blur [43], compression [44], resizing, etc. To further evaluate the generalization of models trained on competing datasets for noisy videos undergoing complicated real-world degradations, we select 15 noisy videos from the Internet that undergo various real-world post-processing operations in different applications to form General15. Each video in General15 suffers from noise contamination from the blind post-processing pipeline.

In Fig. 11, we show a visual comparison of test videos in the General15 dataset with practical noise. The model trained with PVDD yields more outstanding visual quality. For example, RVDT in Fig. 11 trained with PVDD generates the most pleasing results in both smooth areas (most noise eliminated) and regions with rich textures (details preserved). Results from training with the CRVD dataset show blurry artifacts, while those with the DAVIS dataset retain much noise. The superiority of PVDD profits from its realistic

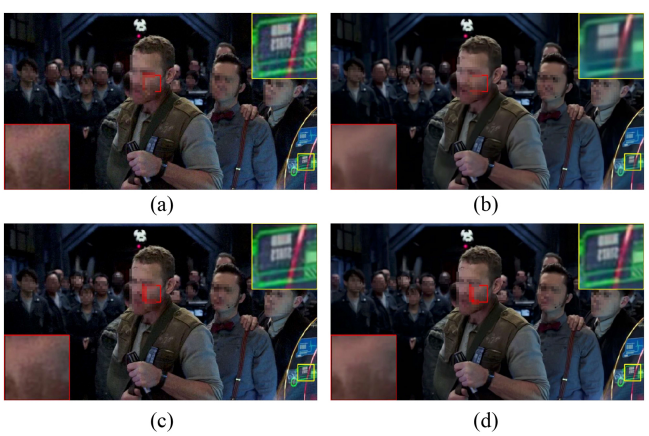


Fig. 11 The denoising results towards real-world noisy videos for RVDT trained on different video denoising datasets and tested on General15. The network trained on PVDD gives the most pleasing results: noise-free smooth area (red rectangle); clean, realistic, and sharp denoised results for regions with complex textures (yellow rectangle). (a) Input; (b) RVDT trained on CRVD; (c) RVDT trained on DAVIS; (d) RVDT trained on PVDD

noise, practical motion patterns, and diverse scenes. General15 will also be publically released to the research community as a real-world video denoising benchmark.

5.4.5 Summary of generalization ability

First, we have conducted a cross-dataset evaluation to measure generalization [45], as detailed in Table 1 and Fig. 10 of the main paper. We adopt two experimental settings: 1) training on PVDD and CRVD, then testing on DAVIS; and 2) training on PVDD and DAVIS, then testing on CRVD. Since the training and evaluation data are distinct in both cases, this setup directly tests generalization ability. The results confirm that models trained on our PVDD exhibit superior generalization ability.

Second, we introduce the General15 dataset, a collection of videos gathered from the Internet that exhibit a range of real-world degradations and post-processing artifacts. As shown in Figs. 1 and 11, the method trained on our PVDD again performs better on these videos, demonstrating the generalization to real-world conditions brought by PVDD.

5.5 Evaluation for denoising network

In this section, we conduct experiments to evaluate the video denoising performance of our designed RVDT. We compare the effects of RVDT against current SOTA video denoising baselines on different datasets for a comprehensive evaluation, including PVDD, CRVD, and DAVIS datasets.

5.5.1 Evaluation strategy

We choose current SOTA video denoising frameworks as the baselines for comparison, as introduced in Subsection 5.2. First, we train RVDT and all baselines on PVDD, and evaluate their performances on Dynamic20 and Static15 that have the corresponding ground truths. Then, RVDT and all baselines are trained on CRVD and DAVIS, and are evaluated on the corresponding testing datasets, respectively.

Figure 12 shows the qualitative comparison among varying non-blind models in the sRGB domain. We can again see the advantage of our designed RVDT compared with current SOTA video denoising approaches. The same visual superiority can also be observed in the RAW domain's comparison, as shown in Fig. 13 (RAW images are processed with demisaicing for visualization).

5.5.2 Quantitative results

The results of the evaluation on the PVDD are shown in Table 2, where all networks are trained with both blind and non-blind versions. Trained on the same dataset, our designed RVDT obviously outperforms the other video-denoising approaches on both PSNR and SSIM, demonstrating the generalization ability of RVDT. For example, even under the challenging blind setting, RVDT achieves more than 1dB gain on the sRGB domain of Dynamic20 in terms of the PSNR, and more than 3% advantage in terms of the SSIM. The effectiveness of RVDT comes from the designed bi-directional spatial-temporal propagation mechanism achieved with the improved transformer blocks.

Moreover, the superiority of RVDT can also be proved when the comparison is conducted on the CRVD and DAVIS datasets. As shown in Table 3, our proposed RVDT can also achieve the SOTA performance, when trained on the training set of CRVD/DAVIS and evaluated on the corresponding testing set. The experiments are conducted in the sRGB domain and blind network version, which is the most challenging situation in practice. Thus, the performance of RVDT is stable and is not dependent on the training data.

Furthermore, the cross-dataset evaluation in Table 1 shows that our proposed RVDT achieves state-of-the-art performance among

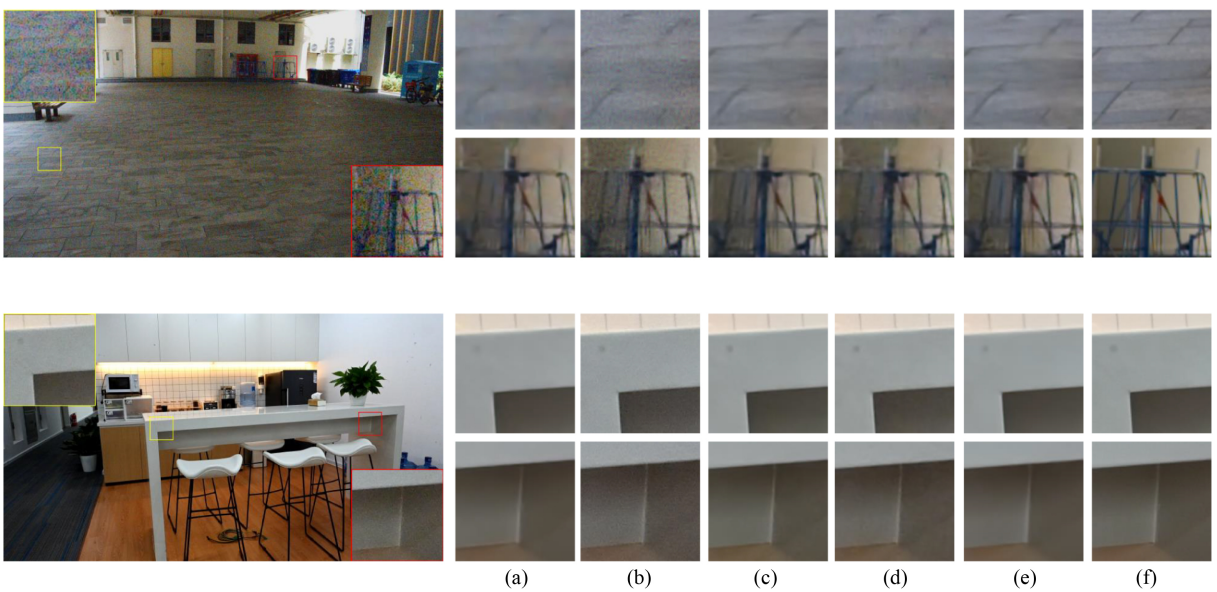


Fig. 12 The visual comparison for different video denoising frameworks trained on PVDD. These networks are trained in the sRGB domain with the non-blind version. Our framework has the best denoising performance in terms of noise removal and detail preservation. The upper part is of Dynamic20 testcase, the lower part is of Static15 testcase. (a) VNLnet; (b) EDVR; (c) FastDVDnet; (d) RViDeNet; (e) RVDT; (f) GT

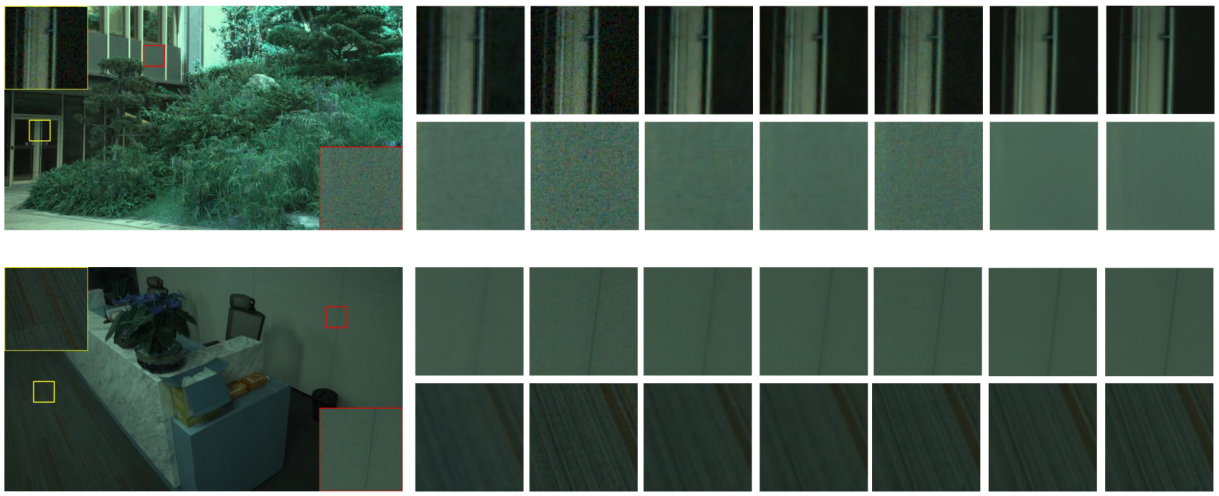


Fig. 13 The qualitative comparison among different video denoising models trained on PVDD, which are trained in the RAW domain with the non-blind version. Our framework achieves the best denoising effects. (a) VNLnet; (b) EDVR; (c) FastDVDnet; (d) RViDeNet; (e) RVDT; (f) GT

Table 2 Quantitative comparison in both the sRGB and RAW domain on PVDD dataset, -RAW means the RAW models

	Dynamic20		Static15	
	Blind	Non-blind	Blind	Non-blind
VNLnet	33.30/0.912	33.69/0.918	35.33/0.933	35.89/0.926
FastDVDnet	34.46/0.916	34.78/0.923	35.25/0.908	36.09/0.929
RViDeNet	33.90/0.911	34.15/0.915	35.31/0.919	35.98/0.917
EDVR	31.34/0.789	31.78/0.810	33.98/0.866	34.46/0.877
RVDT	**35.71/0.941**	**35.85/0.942**	**35.76/0.924**	**36.61/0.941**
VNLnet-RAW	37.05/0.941	35.75/0.933	42.21/0.980	41.01/0.975
FastDVDnet-RAW	43.95/0.966	42.17/0.965	46.42/0.983	47.94/0.989
RViDeNet-RAW	43.92/0.977	43.76/0.975	47.14/0.985	47.95/0.984
EDVR-RAW	36.60/0.873	36.66/0.875	42.57/0.954	42.64/0.955
EMVD-RAW	39.15/0.934	38.44/0.928	46.16/0.983	44.82/0.978
RVDT-RAW	**44.30/0.979**	**44.86/0.981**	**48.70/0.992**	**48.56/0.991**

Table 3 Quantitative comparison in the sRGB domain on both DAVIS and CRVD dataset

	CRVD	DAVIS
VNLnet	22.62/0.873	28.40/0.907
FastDVDnet	15.77/0.717	32.13/0.919
RViDeNet	37.98/0.957	37.40/0.957
EDVR	38.48/0.957	32.78/0.866
RVDT	**39.01/0.964**	**37.77/0.962**

baseline methods, when trained on the same dataset and tested on a completely different one. This highlights the strong generalization capability of RVDT, which is beneficial for real-world scenarios.

5.5.3 Efficiency comparison

Besides the performance comparison, we conduct efficiency analysis, comparing computational costs and inference time. The analysis includes representative metrics: parameter number, FLOPs, and running time. The comparison results are presented in Table 4, demonstrating that our method does not impose a heavy burden in terms of computational cost or inference time. In other words, the effectiveness and generalization ability of our method are not achieved at the expense of efficiency.

6 Conclusion

Our contributions are summarized as follows.

1) PVDD Dataset

We capture clean RAW videos exhibiting diverse dynamic motions.

Table 4 Quantitative comparison in the sRGB domain on both DAVIS and CRVD datasets. FLOPs are measured using 720p input frames, and runtime is evaluated on 720p video sequences with an NVIDIA 4090 GPU. Our method attains noticeable performance improvements without imposing excessive parameter or computational costs

Method	Params (M)	FLOPs (G)	Runtime (ms)	CRVD	DAVIS
VNLnet	3.5	257.84	126	22.62/0.873	28.40/0.907
FastDVDnet	2.9	664.99	48	15.77/0.717	32.13/0.919
RViDeNet	12.0	1965.04	972	37.98/0.957	37.40/0.957
EDVR	20.6	3088.98	1512	38.48/0.957	32.78/0.866
RVDT (Ours)	14.2	2294.63	732	**39.01/0.964**	**37.77/0.962**

Then, we synthesize realistic noise using our calibrated noise model with accurate camera parameters (Fig. 3). Finally, an ISP is designed with tailored components and parameters to approximate its real-world counterpart (Figs. 4 and 5).

This leads to our paper's first contribution: PVDD, a large-scale, information-rich video denoising dataset in both the RAW and RGB domains. Unlike the existing datasets, ours features complex natural motion and realistic noise.

The experimental results in Subsection 5.4 demonstrate the superiority of our PVDD dataset. This is evidenced by its strong cross-dataset performance (Table 1, Fig. 10) and superior results on the General15 real-world benchmark (Fig. 11), highlighting its exceptional generalization capability.

2) RVDT Denoising Network

We design spatial-varying operations to capture both long- and short-range dependencies for spatial denoising. Furthermore, we propose a novel attention mechanism for effective temporal feature propagation. This mechanism comprises two key components: a temporal transformer block with transmission and merging layers, and a channel-spatial attention MLP block.

This constitutes our second contribution: a novel video denoising network based on directional transformer blocks. Its performance is validated via extensive evaluations (Subsection 5.5) on the Dynamic20 and Static15 (Table 2, Figs. 12 and 13), as well as on the CRVD and DAVIS datasets (Table 3).

7 Future work

Although our dataset provides rich and diverse motion information and realistic noise patterns, it has only been captured with a single camera. In the following, we discuss the limitations of relying on a single camera, as well as future work to extend our dataset and explore better frameworks.

Limitation of relying on a single camera. We believe the limitations stem from two primary factors. First, data collected from one camera cannot be easily utilized to train models for another, as different cameras have unique properties, such as their noise characteristics. On the other hand, collecting a unique dataset for each individual camera leads to inefficient resource usage and redundant work.

Second, designing a unique network for each individual camera is impractical for real-world applications, as each model requires its own set of parameters. This leads to significant computational and implementation costs when an application must support multiple devices.

In summary, developing a multi-camera data collection pipeline is essential. This will not only advance research into efficient denoising networks for diverse cameras but also significantly improve data collection efficiency and reduce costs.

The roadmap for expanding the dataset. First, we will survey current representative camera devices, including DSLRs (e.g., Sony, Canon, Fujifilm) and smartphones (e.g., iPhone, Samsung, Huawei). We select these devices for their high likelihood of providing detailed official documentation and support, which is crucial for collecting RAW data.

Second, we will establish a schedule for the various data collection within different scenes. We will then use the selected devices to sequentially capture RAW data from each scene and perform all necessary calibration procedures. This process will enable us to synthesize paired RAW and RGB datasets using the strategy outlined in our work.

Finally, we will use these datasets to investigate methods for handling multi-camera scenarios. For instance, we will explore the use of multimodal unified models or Mixtures of Experts (MoE) to address video denoising from different devices within a single framework. These models can learn to handle varying noise characteristics, and combining data from multiple sources may yield significant performance gains through scaling laws.

Acknowledgements

This work was supported by the Key R&D Program of Zhejiang Province (2024C01036), the Key Project of the National Natural Science Foundation of China (Grant No. 62536007), the Zhejiang Province Science Foundation (LD24F020002), the Zhejiang Province's 2025 "Leading Goose + X" Science and Technology Plan (2025C02034), Shenzhen Science and Technology Program (KQTD20210811090149095), and the Pearl River Talent Recruitment Program (2019QN01X226).

Competing interests

The authors declare that they have no competing interests or financial conflicts to disclose.

Open Access

This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made.

The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Appendixes

A diagram detailing the conversion of single-channel Bayer RAW images to 4-channel RGBG format is shown in Fig. A1. This is a standard processing pipeline, which is also employed in prior works such as CRVD [3].

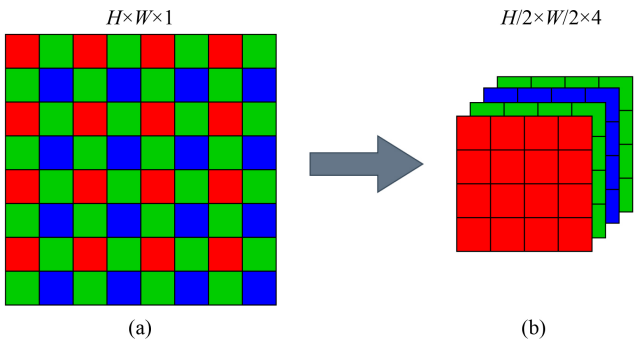


Fig. A1 Illustration of the conversion from single-channel Bayer RAW images to the 4-channel RGBG format. (a) Bayer RAW; (b) RGBG

References

[1] Quan Y, Chen Y, Shao Y, Teng H, Xu Y, Ji H. Image denoising using complex-valued deep CNN. Pattern Recognition, 2021, 111: 107639

[2] Tian C, Zheng M, Zuo W, Zhang B, Zhang Y, Zhang D. Multi-stage image denoising with the wavelet transform. Pattern Recognition, 2023, 134: 109050

[3] Yue H, Cao C, Liao L, Chu R, Yang J. Supervised Raw video denoising with a benchmark dataset on dynamic scenes. In: Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020, 2298−2307

[4] Perazzi F, Pont-Tuset J, McWilliams B, Van Gool L, Gross M, Sorkine-Hornung A. A benchmark dataset and evaluation methodology for video object segmentation. In: Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. 2016, 724−732

[5] Zhou Y, Jiao J, Huang H, Wang Y, Wang J, Shi H, Huang T. When AWGN-based denoiser meets real noises. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence. 2020, 13074−13081

[6] Zhang Y, Qin H, Wang X, Li H. Rethinking noise synthesis and modeling in Raw denoising. In: Proceedings of 2021 IEEE/CVF International Conference on Computer Vision. 2021, 4573−4581

[7] Tassano M, Delon J, Veit T. FastDVDnet: towards real-time deep video denoising without flow estimation. In: Proceedings of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020, 1351−1360

[8] Abdelhamed A, Lin S, Brown M S. A high-quality denoising dataset for smartphone cameras. In: Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018, 1692−1700

[9] Anaya J, Barbu A. RENOIR–a dataset for real low-light image noise reduction. Journal of Visual Communication and Image Representation, 2018, 51: 144−154

[10] Chen C, Chen Q, Do M, Koltun V. Seeing motion in the dark. In: Proceedings of 2019 IEEE/CVF International Conference on Computer Vision. 2019, 3184−3193

[11] Jin X, Niklaus S, Zhang Z, Xia Z, Guo C, Yang Y, Chen J, Li C. Classic video denoising in a machine learning world: robust, fast, and controllable. In: Proceedings of 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2025, 2084−2093

[12] Vaksman G, Elad M, Milanfar P. Patch craft: video denoising by deep modeling and patch matching. In: Proceedings of 2021 IEEE/CVF International Conference on Computer Vision. 2021, 2137−2146

[13] Liang J, Fan Y, Xiang X, Ranjan R, Ilg E, Green S, Cao J, Zhang K, Timofte R, Van Gool L. Recurrent video restoration transformer with guided deformable attention. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. 2022, 28

[14] Liang J, Cao J, Fan Y, Zhang K, Ranjan R, Li Y, Timofte R, Van Gool L. VRT: a video restoration transformer. IEEE Transactions on Image Processing, 2024, 33: 2171−2182

[15] Xu X, Li M, Sun W, Yang M H. Learning spatial and spatio-temporal pixel aggregations for image and video denoising. IEEE Transactions on Image Processing, 2020, 29: 7153−7165

[16] Wang X, Chan K C K, Yu K, Dong C, Change Loy C. EDVR: video restoration with enhanced deformable convolutional networks. In: Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. 2019, 1954−1963

[17] Im Y, Pak J, Na S, Park J, Ryu J, Moon S, Koo B, Kang S J. Supervised denoising for extreme low-light raw videos. IEEE Transactions on Circuits and Systems for Video Technology, 2025, 35(11): 10693−10704

[18] Lindner L, Effland A, Ilic F, Pock T, Kobler E. Lightweight video denoising using aggregated shifted window attention. In: Proceedings of 2023 IEEE/CVF Winter Conference on Applications of Computer Vision. 2023, 351−360

[19] Li D, Shi X, Zhang Y, Cheung K C, See S, Wang X, Qin H, Li H. A simple baseline for video restoration with grouped spatial-temporal shift. In: Proceedings of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 9822−9832

[20] Zhang G, Zhang Y, Yuan X, Fu Y. Binarized low-light raw video

enhancement. In: Proceedings of 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024, 25753−25762

[21] Zhang Z, Jiang Y, Shao W, Wang X, Luo P, Lin K, Gu J. Real-time controllable denoising for image and video. In: Proceedings of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023, 14028−14038

[22] Maggioni M, Huang Y, Li C, Xiao S, Fu Z, Song F. Efficient multi-stage video denoising with recurrent spatio-temporal fusion. In: Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 3465−3474

[23] Zhao H, Tian L, Xiao X, Hu P, Gou Y, Peng X. AverNet: all-in-one video restoration for time-varying unknown degradations. In: Proceedings of the 38th International Conference on Neural Information Processing Systems. 2024, 4043

[24] Yue H, Cao C, Liao L, Yang J. RViDeformer: efficient raw video denoising transformer with a larger benchmark dataset. IEEE Transactions on Circuits and Systems for Video Technology, 2025, 35(9): 8929−8944

[25] Fu Z, Guo L, Wang C, Wang Y, Li Z, Wen B. Temporal as a plugin: unsupervised video denoising with pre-trained image denoisers. In: Proceedings of the 18th European Conference on Computer Vision. 2024, 349−367

[26] Zheng H, Pang T, Ji H. Unsupervised deep video denoising with untrained network. In: Proceedings of the 37th AAAI Conference on Artificial Intelligence. 2023, 3651−3659

[27] Reinhard E, Stark M, Shirley P, Ferwerda J. Photographic tone reproduction for digital images. In: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques. 2002, 661−670

[28] Rowlands D A. Color conversion matrices in digital cameras: a tutorial. Optical Engineering, 2020, 59(11): 110801

[29] Robertson A R. Computation of correlated color temperature and distribution temperature. Journal of the optical society of America, 1968, 58(11): 1528−1535

[30] Abdelhamed A, Brubaker M, Brown M. Noise flow: noise modeling with conditional normalizing flows. In: Proceedings of 2019 IEEE/CVF International Conference on Computer Vision. 2019, 3165−3173

[31] Chan K C K, Wang X, Yu K, Dong C, Loy C C. BasicVSR: the search for essential components in video super-resolution and beyond. In: Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 4945−4954

[32] Mazzia V, Angarano S, Salvetti F, Angelini F, Chiaberge M. Action transformer: a self-attention model for short-time pose-based human action recognition. Pattern Recognition, 2022, 124: 108487

[33] Shi W, Caballero J, Huszár F, Totz J, Aitken A P, Bishop R, Rueckert D, Wang Z. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. 2016, 1874−1883

[34] Yuan F, Zhang Z, Fang Z. An effective CNN and transformer complementary network for medical image segmentation. Pattern Recognition, 2023, 136: 109228

[35] Liang J, Cao J, Sun G, Zhang K, Van Gool L, Timofte R. SwinIR: image restoration using swin transformer. In: Proceedings of 2021 IEEE/CVF International Conference on Computer Vision. 2021, 1833−1844

[36] Liu J, Wu C H, Wang Y, Xu Q, Zhou Y, Huang H, Wang C, Cai S, Ding Y, Fan H, Wang J. Learning raw image denoising with Bayer pattern unification and Bayer preserving augmentation. In: Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019, 2070−2077

[37] Xu X, Wang R, Fu C W, Jia J. SNR-aware low-light image enhancement. In: Proceedings of 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, 17693−17703

[38] Chen C, Chen Q, Xu J, Koltun V. Learning to see in the dark. In: Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018, 3291−3300

[39] Brooks T, Mildenhall B, Xue T, Chen J, Sharlet D, Barron J T. Unprocessing images for learned raw denoising. In: Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019, 11028−11037

[40] Lee S, Cho D, Kim J, Kim T H. Restore from restored: video restoration with pseudo clean video. In: Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021, 3536−3545

[41] Davy A, Ehret T, Morel J M, Arias P, Facciolo G. A non-local CNN for video denoising. In: Proceedings of 2019 IEEE International Conference on Image Processing. 2019, 2409−2413

[42] Hasinoff S W, Sharlet D, Geiss R, Adams A, Barron J T, Kainz F, Chen J, Levoy M. Burst photography for high dynamic range and low-light imaging on mobile cameras. ACM Transactions on Graphics, 2016, 35(6): 192

[43] Liu Y Q, Du X, Shen H L, Chen S J. Estimating generalized Gaussian blur kernels for out-of-focus image deblurring. IEEE Transactions on Circuits and Systems for Video Technology, 2021, 31(3): 829−843

[44] Cheng K L, Xie Y, Chen Q. Optimizing image compression via joint learning with denoising. In: Proceedings of the 17th European Conference on Computer Vision. 2022, 56−73

[45] Jin Y, Ma X, Zhang R, Chen H, Gu Y, Ling P, Chen E. Masked video pretraining advances real-world video denoising. IEEE Transactions on Multimedia, 2025, 27: 622−636



Xiaogang XU is a research fellow in the Chinese University of Hong Kong (CUHK), China. He obtained his PhD degree from CUHK, China in 2022. He received his bachelor degree from Zhejiang University, China. He serves as a reviewer for CVPR, ICCV, ECCV, AAAI, ICLR, NeurIPS, IJCV, and TPAMI. His research interest includes deep learning, large generative models, AIGC, etc.



Yitong YU is currently a computer vision researcher in Kuaishou Technology, China. Before that, he received his bachelor degree from Central University of Finance and Economics, China. His research interests include deep learning, image processing, knowledge distillation, etc.



Nianjuan JIANG received her BS and PhD degree in electrical and computer engineering from National University of Singapore, Singapore in 2007 and 2012, respectively. From 2012 to 2016, she was with the Advanced Digital Sciences Center (ADSC), Singapore, which is a joint research center between the University of Illinois at Urbana-Champaign (UIUC), USA, and the Agency for Science, Technology and Research (A*STAR), Singapore. From 2017 to 2020, she was with Shenzhen Cloudream Technology Co., Ltd. as the R&D director. Since 2020, she has been with SmartMore Co., Ltd. as a R&D director. Her research interests include computer vision, computer graphics, 3D vision, and computational photography.



Jiafei WU received his BS degree from JXUFE, China in 2010, his MS degree and PhD degree from the University of Hong Kong (HKU), China in 2012 and 2017, respectively. He has been a senior engineer, manager, and deputy director from 2018 to 2023 in SenseTime. He is currently with HKU, China. His research interests include deep learning, trustworthy AI, embedded system, and computational intelligence.



Bei YU received his PhD degree from The University of Texas at Austin in 2014. He is currently a Professor in the Department of CSE, the Chinese University of Hong Kong, China. He has served as TPC Chair of ACM/IEEE Workshop on Machine Learning for CAD, and in many journal editorial boards and conference committees. He received eleven Best Paper Awards from ICCAD 2024 & 2021 & 2013, IEEE TSM 2022, DATE 2022, ASPDAC 2021 & 2012, ICTAI 2019, Integration, the VLSI Journal in 2018, ISPD 2017, SPIE Advanced Lithography Conference 2016, six ICCAD/ISPD contest awards, IEEE CEDA Ernest S. Kuh Early Career Award in 2021, DAC Under-40 Innovator Award in 2024, and Hong Kong RGC Research Fellowship Scheme (RFS) Award in 2024.



Jiangbo LU (Senior Member, IEEE) received his PhD degree from Katholieke Universiteit Leuven, Belgium in 2009. From 2009 to 2016, he was a Senior Research Scientist with the Advanced Digital Sciences Center (ADSC), a Singapore-based research center of University of Illinois at Urbana–Champaign (UIUC). From 2017 to 2020, he was with Shenzhen Cloudream Technology Co., Ltd. as the Chief Technology Officer (CTO) and as a Chief Scientist in 2019. Since 2020, he has been with SmartMore Co., Ltd. as the Co-founder and a CTO. Since 2021, he has also been an Adjunct Professor of the South China University of Technology, China. His research interests include computer vision, 3D vision, and image processing.



Jiaya JIA (IEEE Fellow) received his PhD degree in Computer Science from Hong Kong University of Science and Technology (HKUST), China in 2004 and is currently a chair professor in HKUST, China. Previously, he was a full professor in Department of CSE at the Chinese University of Hong Kong (CUHK), China. He is an Associate Editor-in-Chief of IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) and is also in the editorial board of International Journal of Computer Vision (IJCV). He continuously served as area chairs for ICCV, CVPR, AAAI, ECCV, and several other conferences for organization. He was on program committees of major conferences in graphics and computational imaging, including ICCP, SIGGRAPH, and SIGGRAPH Asia. He received the Young Researcher Award 2008 and Research Excellence Award 2009 from CUHK, China.