Joint Video Denoising and Super-Resolution Network for IoT Cameras

Liming Ge, Wei Bao, Dong Yuan, Bing B. Zhou, and Zhiyong Wang

***Abstract*—IoT (Internet of Things) cameras have widely been deployed over the last few years. These cameras are often with limited hardware so that they can only capture noisy videos in low resolution. In this work, we propose the joint video denoising and super-resolution network for IoT cameras, which consists of the noise-robust moving-attention (NRMA) module and the noise-eliminated upsampling (NEU) module. In NRMA, we adopt a coarse-to-fine approach by first extracting the coarse flow and then refining through bi-directional feature propagation among adjacent frames. In NEU, we further utilize inner-frame features for noise-elimination and upsampling. Through this approach, we avoid the negative effects brought by applying denoising and super-resolution in tandem, and enhance the reconstruction of moving objects by the embedded attention layers in NRMA. We conduct comprehensive experiments using existing datasets corrupted by additive white Gaussian noise. We also establish a realistic dataset with real-world noise. Our method achieves drastic performance gain compared with benchmarks in both the existing datasets and the realistic dataset.**

***Index Terms*—IoT cameras, video denoising, video super-resolution.**

I. Introduction

Internet of Things (IoT) cameras have widely been developed and deployed over the last decade. To support a range of anywhere anytime applications (e.g., video surveillance) in a cost-effective way, a massive amount of IoT cameras are geo-distributed, but each IoT camera is with low cost. However, due to the limited hardware (bottlenecked by the cost of photosensor), significant noise is perceived (especially during the night [1]) and the video resolution is low. With these noisy and low-resolution videos, the perceived quality is far from satisfactory. Even if the majority of the footage could be less useful, in the case of an incident (e.g., missing person, traffic accident, etc.), there is a need to recover noise-free and high-quality video to aid the investigation of the incident.

With the significant advancement in deep neural networks, various video enhancement techniques have been proposed. Among them, video denoising and video super-resolution methods have strong potential to enhance the quality of videos captured by IoT cameras. Video denoising methods aim to acquire a clean video from noisy observations. Video super-resolution methods aim to obtain high-resolution videos from low-resolution ones. Through the combination of video denoising and video super-resolution methods, we are expected to enhance the perceived quality. However, the video super-resolution process is very sensitive to noise, while the video denoising process is also sensitive to the artifacts brought by video super-resolution. The resultant quality of direct combination using existing methods is substandard if we apply them in tandem. In particular, the direct combination yields poor results for moving objects (super-resolution may mistakenly assume a noisy pixel as a moving pixel), which are usually the significant part we want to recover from the low-quality footage. Also, the realistic noise, rather than assumed additive white Gaussian noise (AWGN), further complicates the processes in video denoising and super-resolution.

In this work, we propose a joint video denoising and super-resolution network for IoT cameras. It takes a sequence of noisy frames in low resolution (LR). Through a jointly optimized process, it outputs a sequence of noise-free frames in high resolution (HR). It consists of two main modules, namely the noise-robust moving-attention (NRMA) module, and the noise-eliminated upsampling (NEU) module. The motions among consecutive frames are more temporally correlated, while the noise is less dependent among frames, so that we use NRMA and NEU to handle them respectively. In NRMA, we develop a coarse-to-fine approach. It extracts coarse flow without misinterpretation of noise pixels. Then, the coarse flow is refined by bi-directional inter-frame feature propagation. In NEU, the refined flow is further combined with inner-frame features, for eliminating noise and upsampling.

To evaluate the performance of our proposed network, we conduct comprehensive experiments using both existing datasets corrupted by AWGN, and a realistic dataset captured during the night. We obtain paired footage from an IoT camera and a professional camera, which forms the noisy observation and ground-truth observation of the realistic dataset. In terms of performance, our network achieves the highest PSNR on all datasets, and achieves ~ 1 dB gain in PSNR on the realistic dataset (compared with the best benchmark). Our network shows much better noise suppression and detail enhancement especially on the realistic dataset. In terms of delay, our network achieves faster inference than the combinations of the state-of-the-art video denoising and super-resolution methods.

II. Related Work

A. Noise in the Video Shooting Process

IoT cameras can be deployed in large scale, while the hardware of each camera is limited by cost. Due to the

L. Ge, W. Bao, B. B. Zhou, and Z. Wang are with the School of Computer Science, Faculty of Engineering, The University of Sydney, Sydney, NSW 2006, Australia (e-mail: liming.ge@sydney.edu.au; wei.bao@sydney.edu.au; bing.zhou@sydney.edu.au; zhiyong.wang@sydney.edu.au).

D. Yuan is with the School of Electrical and Information Engineering, Faculty of Engineering, The University of Sydney, Sydney, NSW 2006, Australia (e-mail: dong.yuan@sydney.edu.au).

W. Bao is the corresponding author.

characteristics of complementary metal oxide semiconductor (CMOS) photosensors, noise is widely observed especially in these inexpensive devices. There are three types of noise in the entire video shooting process. (1) Due to the quantum nature of light, there is an uncertainty in the number of photons arriving on a pixel area in the CMOS photosensor, which is termed as the shot noise [1] [2]. (2) The photons arriving on the photosensor generate electrons. However, due to the characteristics of the semiconductor device, there exist randomly generated electrons, which bring photon-independent noise, a.k.a. dark current noise [3]. (3) The electrons are read out and amplified in the form of electronic signal. The reading circuit introduces readout noise [4].

Under low light conditions, noise becomes more obvious and severely deteriorates the captured videos. High sensitivity (high ISO) and high shutter speed setting is employed while shooting in low-light [5]. High sensitivity makes objects in the dark more observable [4], while adopting high shutter speed is a prerequisite for video shooting (instead of photo shooting) [6]. However, this setting inevitably brings fewer photons to arrive on the photosensor [7], causing more significant shot noise. It also makes the dark current noise more observable, since the electrons generated by the photons are decreased, while the randomly generated electrons remain unchanged. These undesired electrons are then amplified and cause more readout noise.

Adopting CMOS photosensors with larger sizes allows more photons to arrive and can reduce overall noise. However, using more expensive hardware brings diminishing marginal performance gain. By increasing the CMOS photosensor size from $(1/2.9)''$ (Sony IMX386) to $(1/2.25)''$ (Sony IMX586), a 28% increase, the cost increases from \$111 to \$174, a 56% increase [8]. It is not cost-effective to deploy a large number of IoT cameras with high-performance hardware.

B. Video Denoising

Image [9] and video [10] denoising aims to remove the noise from the captured visual content [11] [12]. While image denoising algorithms use the spatial self-similarity [13] within an image to perform denoising, video denoising algorithms [14] further exploit the self-similarity among frames to benefit the reconstruction.

Deep neural networks (DNNs) yield promising results in exploiting similar pixels (or patches) within the same frame and among adjacent frames [15]. DNN-based video denoising algorithms [16] [17] benefit from this property. Nevertheless, they require paired noisy and ground-truth observations for training [18] [19]. Lacking realistic ground truth data, we can generate random additive white Gaussian noise (AWGN) on top of clean observations to obtain the paired training data. [20] uses generative adversarial networks to better utilize the training data. Blind video denoising algorithms [21] [22] are proposed to bypass this issue by training without clean observations [23] [24]. These algorithms deliver limited performance since they use unpaired noisy data for training. The raw observation from IoT cameras is more complex than randomly generated AWGN with the dynamic streak noise, color channel heterogeneity, and clipping effect [25]. In this paper, we use real-world noisy observations to fine-tune our DNN network.

Video storage and delivery requires encoding and decoding. Video codec [26] solutions compress the video and inevitably introduce compression noise (i.e. Gibbs effect [27], blocking artifacts [28], etc.). Please note that compression noise is orthogonal to our research – We focus on eliminating noise in the video shooting process. With the development of network bandwidth, storage, and coding techniques [29], compression noise can be effectively alleviated. In the experiments, we test both video shooting noise only and video shooting noise + compression noise. We show that the compression noise is insignificant compared with video shooting noise and our proposed method works well to handle video shooting noise together with compression noise when mainstream codecs are applied.

C. Video Super-resolution

Video Super-Resolution (VSR) [30] [31] produces high-resolution (HR) pictorial data using low-resolution (LR) observations. It has been used in a variety of applications [32] [33].

Video super-resolution benefits from the details embedded in multiple LR observations of the same scene and produce HR frames with higher quality [34] [35]. Video super-resolution methods apply temporal alignment prior to the fusion [36] to resolve the discrepancy in multiple LR observations [37]. [38] uses optical flow for temporal alignment. [39] and [40] use deformable alignment without estimation of the motion. More recently, following the typical alignment, fusion, and upsampling process, [41] and [34] improve the computational efficiency by condensing DNN designs. Video super-resolution has been applied to many real-world systems, such as multi-camera systems (a.k.a. multi-lens phones) [42] [43] [44] and hand-held cameras with blurry frames [45] [46]. [47] optimizes super-resolution on real-world videos.

The reconstruction of moving objects is crucial in video enhancement. These moving objects are usually key pieces in the video. The boundary pixels of moving objects are so-called "mixed pixels", whose values represent both the foreground object and the varying background [48]. Video enhancement techniques do not work well when dealing with moving objects, particularly for video super-resolution and video denoising methods. Video super-resolution is error-prone at the object boundary since they mistakenly super-resolve these pixels as separate objects. Video denoising methods mistakenly recognize these "mixed pixels" as noise and eliminate them. Efforts have been made to optimize the performance of video super-resolution [49] [50] [51] [52]. The authors of [53] also enhanced the optical flow estimation network for moving objects, while [54] and [55] designed video interpolation network for moving objects. However, their approaches did not consider the video shooting noise. In this paper, we focus on joint video super-resolution and denoising which can handle moving objects more effectively. Our method extracts the moving object reliably and avoids the artifacts at the object boundary.

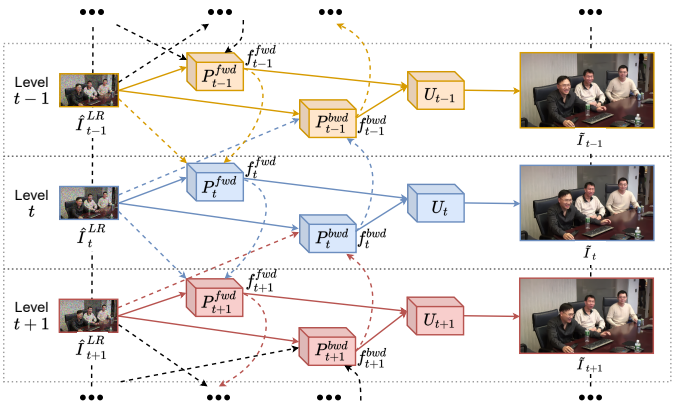


Fig. 1. Detailed pipeline design. Each color represents the reconstruction process of a frame. $P_t^{fwd,bwd}$ denotes the NRMA module (further details in Fig. 2). U_t denotes the NEU module (further details in Fig. 3).

D. Other Video Enhancement Schemes

IoT cameras come in various forms, such as dashcams, wearable cameras, and smartphone cameras. With the number of IoT cameras beginning to surge, efforts have been made on better achieving different specific functions using IoT cameras (e.g., de-raining [56], person identification [57], object detection [58], etc.). However, none of them focus on improving the quality of video through joint video denoising and video super-resolution.

There are other video enhancement schemes including interpolation [59] [60], inpainting [61] [62], and de-fencing [63]. However, they aim to accomplish different tasks, which are drastically different from the scope of this paper.

III. Design

A. Problem Formulation

Our method takes a sequence of n noisy video frames $\hat{I}^{LR}_{\{1,n\}}$ in low resolution (LR), and outputs a sequence of reconstructed noise-free frames $\tilde{I}_{\{1,n\}}$ in high resolution (HR). n is the number of frames in a video chunk. We set $n = 25$ in our experiment but it can be adjusted to other values. Let $\hat{I}^{LR}_t \in \mathbb{R}^{H\times W}$ be the noisy video frame in low resolution at timestamp t, where $t \in \{1, n\}$, and H and W denote the height and width of the frame respectively. We denote its corresponding reconstructed noise-free frame as $\tilde{I}_t \in \mathbb{R}^{sH\times sW}$, where s is the upscaling factor and $s > 1$. We also have the corresponding ground-truth (noise-free) video frame in high resolution, denoted $I_t \in \mathbb{R}^{sH\times sW}$. We aim to minimize the difference between the reconstructed frame $\tilde{I}_t$ and the ground-truth frame I_t.

Please note that we assume the video chuck is available to be processed offline. This is suitable for a range of applications where video chucks are stored and then processed, e.g., the video is stored in the SD card of a dash camera and we process it later after an incident.

B. Pipeline Design

Fig. 1 is the overview of the proposed pipeline. Each frame $\hat{I}^{LR}_t$ is mainly processed by a processing level (level t). There are also cross-level connections, which will be discussed shortly. We illustrate levels $t-1$, t, and $t+1$, and all the other levels follow the same structure. Each frame $\hat{I}^{LR}_t$ in level t passes two noise-robust moving-attention (NRMA) modules with the same structure, one for forward (NRMA-fwd, P^{fwd}_t) and one for backward (NRMA-bwd, P^{bwd}_t) respectively. Then, the outputs pass the noise-eliminated upsampling module (NEU module, denoted as U_t in the figure), to reconstruct the noise-free and high-resolution frame $\tilde{I}_t$. For NRMA-fwd P^{fwd}_t, it takes the current frame $\hat{I}^{LR}_t$ and its previous frame $\hat{I}^{LR}_{t-1}$ as input, and feeds the output feature f^{fwd}_t to the NEU module U_t. For NRMA-bwd, it takes the current frame $\hat{I}^{LR}_t$ and its next frame $\hat{I}^{LR}_{t+1}$ as input, and feeds the output feature f^{bwd}_t to the NEU module U_t. The output of NRMA-fwd f^{fwd}_t will also feed to the NRMA-fwd P^{fwd}_{t+1} in the next level to propagate the inter-frame features. Similarly, the output of NRMA-bwd f^{bwd}_t will also feed to the NRMA-bwd P^{bwd}_{t-1} in the previous level for the same purpose. Please note that in the boundary condition ($t = 1$ or $t = n$) there is no input from $t = 0$ or $t = n + 1$ and we can simply eliminate these connections in levels 1 and n.

Machines, unlike humans, are error-prone when determining whether a pixel belongs to noise or a low-resolution object. In some cases, the "defective pixels" may represent a useful object corrupted by noise. Existing neural networks achieve promising results on sole denoising and super-resolution, but not jointly. We will further elaborate in the next subsection that, with proper design, joint denoising and super-resolution can be achieved both effectively and efficiently. We expand the afferent frames from local neighboring frames (typical case in a multi-input single-output video enhancement neural network) to the whole video. The useful features among consecutive frames yield strong temporal correlation, while the noise is less temporally dependent, so we use the NRMA and NEU modules to handle them respectively. The NRMA-fwd and NRMA-bwd modules propagate temporal information across levels, while the NEU module takes information only from the same level.

C. NRMA Module

To effectively leverage the motion information among frames while avoiding the misinterpretation of noise pixels, the noise-robust moving-attention (NRMA) module is characterized by a coarse-to-fine strategy. As shown in Fig. 2, we design three submodules within NRMA, namely the coarse flow extractor submodule, the spatio warping submodule, and the refinement submodule. It extracts coarse flow without misinterpretation of noise pixels using a coarse flow extractor, then warps and refines the coarse flow using bi-directional inter-frame propagation. The NRMA module has two modes, namely NRMA-fwd P^{fwd}_t and NRMA-bwd P^{bwd}_t. They share the same structure but take different input frames and propagate towards opposite directions.

1) The Coarse Flow Extractor: The coarse flow extractor takes two consecutive frames as input, and extracts the flow. Given a noisy video frame $\hat{I}^{LR}_t$ and a neighboring frame $\hat{I}^{LR}_{t-1}$ or $\hat{I}^{LR}_{t+1}$ in low resolution, the coarse flow extractor of NRMA-

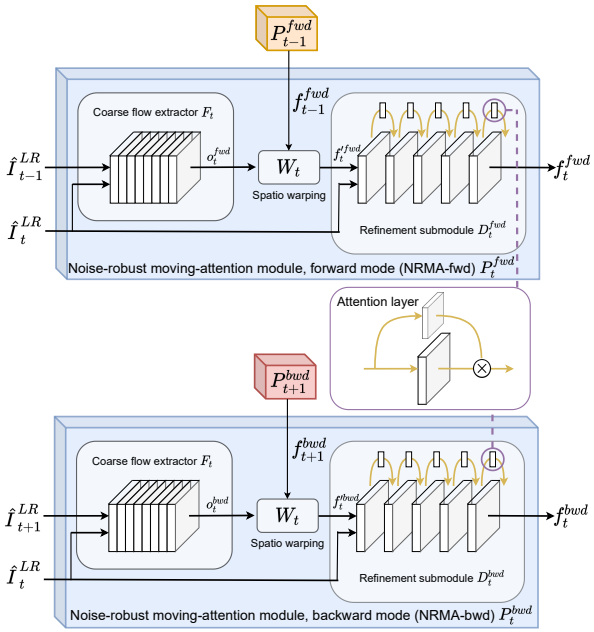


Fig. 2. Noise-robust moving-attention (NRMA) module.

fwd extracts flow o_t^{fwd} using $\hat{I}_t^{LR}$ and $\hat{I}_{t-1}^{LR}$, while the one of NRMA-bwd extracts flow o_t^{bwd} using $\hat{I}_{t+1}^{LR}$ and $\hat{I}_t^{LR}$. Formally, we have

$$o_t^{fwd} = F_t(\hat{I}_t^{LR}, \hat{I}_{t-1}^{LR}), \tag{1}$$
$$o_t^{bwd} = F_t(\hat{I}_{t+1}^{LR}, \hat{I}_t^{LR}), \tag{2}$$

where F_t denotes the coarse flow extractor. The coarse flow extractors of the two modes share the same network parameters (weights). The extracted flows o_t^{fwd} and o_t^{bwd} are coarse but robust to noise, which provide a more accurate view for subsequent reconstruction.

We adopt PWC-Net [64] as the backbone of the coarse flow extraction network. However, we use denoised frames as the training dataset to train the coarse flow extractor individually, and thus it is less likely to be confused by noise. See training details and ablation study in the appendix for more discussions.

2) The Spatio Warping Submodule: In the next step, we employ the spatial warping submodule to warp the propagated features generated in the neighboring levels. In NRMA-fwd, given the propagated feature f_{t-1}^{fwd} and flow o_t^{fwd}, it outputs a warped feature $f_t'^{fwd}$. In NRMA-bwd, given f_{t+1}^{bwd} and o_t^{bwd}, it outputs a warped feature $f_t'^{bwd}$. Formally, we have

$$f_t'^{fwd} = W_t(f_{t-1}^{fwd}, o_t^{fwd}), \tag{3}$$
$$f_t'^{bwd} = W_t(f_{t+1}^{bwd}, o_t^{bwd}), \tag{4}$$

where W_t denotes the spatial warping submodule.

A pixel hindered by noise in one frame is very likely observable in a neighboring frame. Since the features propagated from both ends of the video, the embedded temporal information is more likely to be accurate and the noise is light. By warping the propagated features with the extracted coarse flow, more useful features are preserved while the noise is alleviated.

3) The Refinement Submodule: The refinement submodule refills the warped features with fine details extracted directly from the input frame. In NRMA-fwd, given the input frame $\hat{I}_t^{LR}$ and the warped feature $f_t'^{fwd}$, it outputs a refined feature f_t^{fwd}. In NRMA-bwd, given $\hat{I}_t^{LR}$ and $f_t'^{bwd}$, it outputs a refined feature f_t^{bwd}. Formally, we have

$$f_t^{fwd} = D_t^{fwd}(\hat{I}_t^{LR}, f_t'^{fwd}), \tag{5}$$
$$f_t^{bwd} = D_t^{bwd}(\hat{I}_t^{LR}, f_t'^{bwd}), \tag{6}$$

where D_t^{fwd} and D_t^{bwd} denote the refinement submodule in the forward and backward modes respectively.

The warped features contain little noise but also suffer from the loss of detail due to their coarse nature. The refinement submodule further extracts fine details from the input frame and enhances the features. We obtain two distinct features in two directions to be used by the subsequent noise-eliminated upsampling (NEU) module.

The refinement submodule has five convolutional layers, with attention layers (Fig. 2). This attention layer enables the submodule to learn proper scaling and nonlinear combinations of features, which widely exist in moving objects. The refinement submodule in NRMA-fwd and NRMA-bwd are trained separately to allow propagation in two opposite directions.

By combining the coarse flow extractor, the spatial warping submodule, and the refinement submodule, the NRMA module performs bidirectional propagation and outputs two propagated and refined features, which is robust to noise and accurately extracts moving objects. Formally, we have

$$f_t^{fwd} = P_t^{fwd}(\hat{I}_t^{LR}, \hat{I}_{t-1}^{LR}, f_{t-1}^{fwd}), \tag{7}$$
$$f_t^{bwd} = P_t^{bwd}(\hat{I}_t^{LR}, \hat{I}_{t+1}^{LR}, f_{t+1}^{bwd}). \tag{8}$$

Please note that our design is different from conventional RNNs. RNNs are good at processing noiseless sequence of data or video input [65] [66] [67] [68]. In order to preserve the information from previous frames, RNNs use hidden states in the structure. However, if noise exists, the hidden states yield too strong memory and attempt to find the correlation among noisy pixels, which jeopardize the reconstruction. Therefore, we avoid a design with hidden states. Our design comes with decent memory across frames, enough for finding sufficient temporal redundancy, while reducing the chance of linking irrelevant noise pixels.

D. NEU Module

In this subsection we introduce the noise-eliminated upsampling (NEU) module, which receives the two features from the NRMA-fwd and NRMA-bwd. The received two features are still with noise so we need NEU to further suppress it. Please note that in NRMA, even though the warped and propagated features are robust to noise, the refinement submodule attempts to preserve fine details from the raw input frame in which noise persists. Fortunately, the remained noise is not propagated or enlarged among frames due to the design of NRMA. NEU in the level t is sufficient to effectively eliminate the noise by itself without consulting with NEUs in the neighboring levels, i.e., no connections among NEUs.

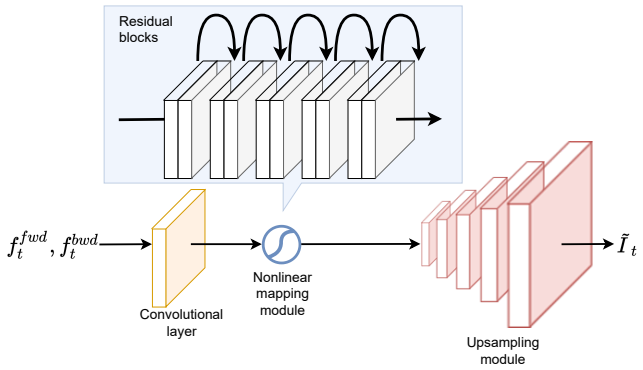


Fig. 3. Noise-eliminated upsampling (NEU) module.

The structure of NEU is shown in Fig. 3. First, since we have obtained two propagated and refined features f_t^{fwd} and f_t^{bwd}, we need to aggregate them for further enhancement. We do not use a simple average on the two features, as it leads to blurry results caused by disputed estimation. Instead, we concatenate the two features and feed them into a convolutional layer to output the fused feature map.

Second, we feed the fused feature map into a nonlinear mapping submodule which is composed of five residual blocks. The nonlinear mapping submodule adopts the nonlinearity to cross-check the two candidate features leveraging inner-frame spatial similarities and redundancies. It enriches the details, further eliminates the noisy pixels, and makes up the corrupted pixels.

Third, we utilize an up-sampling submodule composed of an up-scaling layer [69] to increase the resolution of the feature map, with a sub-pixel convolution [48], and obtain the noise-free frame $\tilde{I}_t$ in high resolution.

In sum, in this NEU module, we have

$$\tilde{I}_t = U_t(f_t^{fwd}, f_t^{bwd}), \tag{9}$$

where U_t denotes the NEU module in level t.

IV. Experiment

In this section, we compare our method with the benchmarks under existing datasets corrupted by AWGN. After that, we collect realistic paired noisy frames in low resolution and ground-truth frames in high resolution during the night and evaluate the performance of our method and the benchmarks. Finally, we discuss the running time and conduct ablation studies.

A. Benchmarks and Evaluation Metrics

Since there is no existing solution jointly handling denoising and super-resolution, we consider adopting state-of-the-art solutions in denoising only and super-resolution only and run them in tandem. For the denoising part, we consider two methods: DVDnet [16] and FastDVDnet [17]. DVDnet is an optical flow based video denoising method while FastDVDnet is a lightweight video denoising method without optical flow. For the super-resolution part, we also consider two methods: IconVSR [38] and RRN [41]. IconVSR utilizes the most essential components to achieve efficient video super-resolution, while RRN utilizes a recurrent residual network to boost the performance. Then, we run denoising and super-resolution in tandem to compare with our method. Please note that we can swap the order of denoising and super-resolution and treat them as different benchmarks. For example, *DVDnet + RRN* means we run DVDnet first and then RRN; *RRN + DVDnet* means we run RRN first and then DVDnet. In sum, we consider 8 benchmark combinations for both existing datasets and the realistic dataset.

We adopt peak signal to noise ratio (PSNR) as the evaluation metric to compare the reconstruction quality quantitatively. Compared with the structural similarity index (SSIM), PSNR is a more representative evaluation metric in the presence of noise [16] [17]. Unless otherwise mentioned, we set $n = 25$ in this section. We employ the RGB color model with each color value in $[0, 255]$.

B. Experiment with Existing Datasets

1) Experiment Settings: The dataset is composed of 2 color sequences from the *Derf's Test Media collection* [70] in 1920×1080 resolution (1080p), and 4 color sequences from the GoPro dataset [17] in 960×540 resolution (540p). We generate the input frames in low resolution by applying $2\times$ bicubic down-sampling, and result in frames in 960×540 resolution for DERF dataset and 480×270 for GoPro dataset. It is then corrupted by AWGN of standard deviation $\sigma \in \{10, 20, 30, 40, 50\}$.

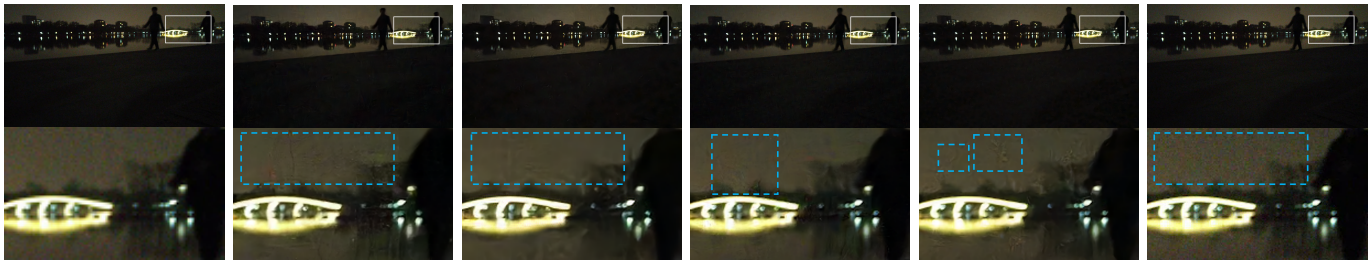
2) Results: The quantitative results are summarized in Table I, and the visual results are demonstrated in the appendix due to space limitation.

Our method substantially outperforms all benchmarks both quantitatively and visually under all scenes and noise levels. In general, our method outputs frame sequences with remarkable temporal coherence, low flickering, and noise is barely observable.

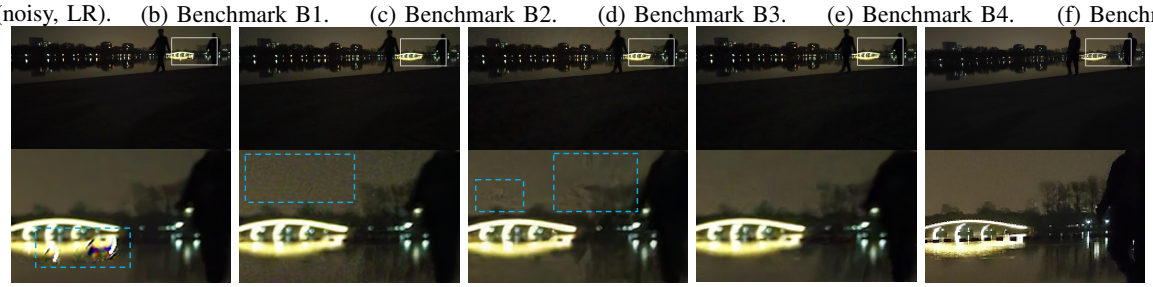
Among benchmark schemes, we observe that running super-resolution first (B1–B4) yields weaker results than running denoising first (B5–B8), as shown in Table I. This is because super-resolution methods are excessively sensitive to abrupt pixels. B1–B4 mistakenly recognize the noise pixels as object boundaries and amplify them. The amplified noise pixels are less likely to be eliminated by a denoising method in the second step, so that the overall performance is low.

As for the comparison between two video denoising benchmark schemes, we observe that DVDnet outperforms FastDVDnet, especially under higher noise levels. This is because DVDnet utilizes optical flows to aid the reconstruction. Under higher levels of noise, optical flow extraction methods find the trajectories of the pixels and still yield satisfactory results. Nevertheless, DVDnet introduces a much larger inference delay.

Both the DERF dataset and the GoPro dataset show the same trend. The scenes in DERF dataset are typical scenes captured by IoT cameras (e.g., video surveillance and video analytics). We expand the experiment by considering more



(a) Input (noisy, LR). (b) Benchmark B1. (c) Benchmark B2. (d) Benchmark B3. (e) Benchmark B4. (f) Benchmark B5.



(g) Benchmark B6. (h) Benchmark B7. (i) Benchmark B8. (j) Ours. (k) Ground-truth.

Fig. 4. Visualized comparison of results under the realistic dataset. Quantitative results are shown in Table 2 (RAW Frames). We show the input noisy frame, benchmarks B1 – B8, our method, and the ground-truth frame. The second row are enlarged versions of the regions in the white boxes in the first row. The dashed blue boxes in the second row highlight the artifacts. Best viewed in digital format.

TABLE I

AVERAGE PSNR (IN DB) OF DERF AND GOPRO DATASETS UNDER DIFFERENT NOISE LEVELS (σ), ALONG WITH THE DELAY (IN SECONDS). THE DELAY IS MEASURED USING DERF DATASET WITH 25 FRAMES ($n = 25$). THE RAW SCHEME BICUBIC UPSAMPLES THE FRAME AND DOES NOT ELIMINATE THE NOISE, AND CAUSES NO DELAY. B1 – B8 DENOTE THE BENCHMARK SCHEMES. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD.

Schemes	DERF dataset					GoPro dataset					Delay (s)
	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	
RAW	24.03	22.42	20.28	18.11	16.56	15.57	14.71	13.96	13.09	12.77	None
(B1) RRN + FastDVDnet	23.13	21.71	19.40	17.97	16.75	13.64	13.64	13.63	13.64	13.25	8.37
(B2) RRN + DVDnet	23.41	22.26	20.07	18.42	17.83	14.79	14.78	14.14	13.17	13.11	363.60
(B3) IconVSR + FastDVDnet	22.76	21.37	19.87	18.34	16.88	14.80	14.55	14.18	13.88	13.64	10.20
(B4) IconVSR + DVDnet	23.74	22.22	20.08	18.71	17.30	14.96	14.93	14.72	14.42	14.26	365.65
(B5) FastDVDnet + RRN	24.74	22.97	21.44	20.30	19.49	15.72	15.48	15.14	14.80	14.52	6.56
(B6) DVDnet + RRN	25.34	23.96	22.44	21.17	20.48	15.88	15.79	15.58	15.26	15.03	243.60
(B7) FastDVDnet + IconVSR	26.73	23.37	20.67	18.93	17.81	17.34	16.67	15.99	15.44	15.02	8.43
(B8) DVDnet + IconVSR	25.73	23.36	21.16	19.40	18.37	17.64	17.25	16.72	16.12	15.78	248.36
Ours	**33.47**	**30.45**	**29.39**	**29.39**	**29.64**	**25.76**	**25.37**	**24.87**	**24.34**	**23.88**	**5.43**

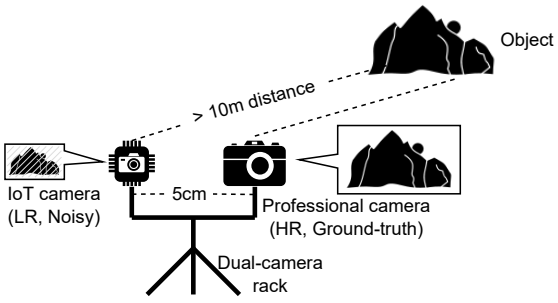


Fig. 5. Illustration of realistic dataset capture.

TABLE II

AVERAGE PSNR (IN DB) OF THE REALISTIC DATASET UNDER DIFFERENT NOISE SETTINGS. FULL NAMES OF B1–B8 ARE SHOWN IN TABLE I. "PRORES" DENOTES APPLE PRORES 422 VIDEO CODEC. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD.

Schemes	AWGN emulated $\sigma = 23$	Real			
		RAW Frames	H.264	HEVC	ProRes
No processing	16.19	16.20	16.18	16.20	16.20
(B1)	16.57	16.07	16.07	16.07	16.07
(B2)	18.24	18.15	18.15	18.14	18.15
(B3)	16.52	16.65	16.65	16.65	16.65
(B4)	18.41	18.82	18.82	18.81	18.82
(B5)	20.22	20.19	20.18	20.19	20.19
(B6)	20.62	20.27	20.27	20.27	20.27
(B7)	21.01	21.00	20.99	20.99	21.00
(B8)	21.24	21.21	21.21	21.21	21.21
Ours	**22.16**	**22.23**	**22.22**	**22.23**	**22.23**

general scenes with moving objects using the GoPro dataset. Our method still outperforms the benchmark schemes. The results demonstrate that our method works well for both static scenes and scenes with moving objects.

C. Experiment with Realistic Dataset

Lacking paired realistic noisy and ground-truth observations, video denoising techniques are conventionally evaluated on clean frames corrupted by synthesized AWGN. However, noise in a realistic environment is more complicated. It does not strictly follow the Gaussian distribution and there are

temporal and/or spacial correlations [3]. Therefore, we are motivated to capture a realistic dataset from IoT cameras and evaluate our method under real-world settings.

1) Experiment Settings: Video Capture. We collect paired realistic data using an IoT camera (for noisy observation) and a professional camera (for clean ground-truth observation). We mount the two cameras on a dual-camera rack, with their lens being 5cm apart from each other, facing toward the same direction, as shown in Fig. 5. We capture scenes > 10 meters away from the camera rack, so that the captured scene difference between the two cameras is negligible. Since the two cameras have different fields of view, the captured video can be zoomed/trimmed according to the results from ORB image matching [71] to obtain identical view.

For calibration purposes, we set the professional camera in low resolution and compare it with the captures of the IoT camera under the daylight condition (when the noise level is low). With the frames captured by the professional camera being ground-truth, the frames captured by the IoT camera achieve more than 38 dB in PSNR under this setting. The high PSNR demonstrates two cameras capture almost identical scenes and we have effectively calibrated the captures of the two cameras.

We use a Sony IMX386 sensor with $(1/2.9)''$ CMOS photosensor as the IoT camera, and a Samsung GN1 sensor with $(1/1.33)''$ CMOS photosensor as the professional camera which shows much better performance under low-light conditions. We capture the video dataset with the following settings: 1/30 second exposure time; ISO 1,600; frame rate 12.5 fps for both cameras. The resolution of the IoT camera is 540p, while the professional camera is 1080p. We capture 50 videos in different scenarios during the night, each with a 30s duration. 40 of them help with training the NEU module and 10 of them are used for evaluation.

Settings of Compression Noise for Comparison. As discussed in Sec. II-B, to understand the effects of compression noise in addition to the video shooting noise, we encode the noisy videos (captured by the IoT camera) by three mainstream video codecs: H.264 [72], HEVC (H.265) [73], and Apple ProRes 422 [74]. We then decode the video and apply our method. Through this way, we can test the joint effect of shooting noise and compression noise.

Settings of AWGN for Comparison. We discuss the difference between the real-world noise and the AWGN. We also add AWGN to the clean observation from the professional camera in the same way as Sec. IV-B1 (AWGN emulated). For a fair comparison, we need to set an appropriate σ value for the AWGN. We compare the noisy observation captured from the IoT camera and the clean observation from the professional camera, and calculate the standard deviation. The resultant standard deviation $\sigma = 23$ reflects the level of the real noise in this dataset and we apply it as the σ value for the AWGN.

2) Results: The quantitative results are summarized in Table II ("RAW Frames" column). We visualize a group of results in Fig. 4. The white boxes are zoomed in.

Our method substantially outperforms all benchmarks both quantitatively and visually. The reconstructed video shows remarkable temporal coherence, low flickering, and noise is barely observable.

Among the benchmark schemes that perform video super-resolution first and then denoising, B1 (RRN + FastDVDnet) and B2 (RRN + DVDnet) give more striking creases all over the frame. This is because they both use RRN as the video super-resolution method. RRN is more prone to errors and easily misinterprets the noise as sharp object boundaries, making it harder for video denoising methods to erode and flatten the frame. B3 (IconVSR + FastDVDnet) and B4 (IconVSR + DVDnet) use IconVSR as the video super-resolution method. Compared with B1 and B2, they yield slightly better results. The creases are less significant. However, such creases can still be observed all over the frame, resulting in degraded quality. B1 – B4 performs video super-resolution first, then video denoising. The video super-resolution methods are very sensitive to abrupt pixels (noise). Super-resolution methods mistakenly recognize them as valid pixels and attempt to interpret them as object boundaries. Such fake 'boundaries' are then amplified. Once amplified, these defects are unlikely to be removed by video denoising methods. Video denoising methods attempt to erode these defects, creating noticeable textures, especially in dark areas.

Among the benchmark schemes that perform video denoising first and then super-resolution, B5 (FastDVDnet + RRN) suffers from weak noise suppression. This is because its video denoising method, FastDVDnet, was unable to reduce the strong real world noise. This noise is then amplified by its video super-resolution method, RRN, resulting in poor reconstruction. B6 (DVDnet + RRN) yields strong noise suppression, but suffers from scribbles of the bright spots (e.g., below the illuminated bridge in the dashed blue box in Fig. 4(g)). This is because its denoising network (i.e., DVDnet) shows erroneous estimations of optical flows in the presence of noise. B7 (FastDVDnet + IconVSR) alleviates the aforementioned artifacts since it does use flow for denoising. It reconstructs the objects more accurately, but at the cost of weaker noise suppression: The noise is still observable in the dark region above the illuminated bridge in the dashed box in Fig. 4(h). B8 (DVDnet + IconVSR) shows stronger noise suppression, but suffers from artifacts caused by the interference between the denoising and super-resolution modules (i.e. notable texture is observed in the dark region surrounding the objects in the dashed boxes in Fig. 4(i)). B5 – B8 perform video denoising first, then video super-resolution. Performing video denoising first eliminates the majority of the noise of the frame. However, denoising methods may erode the defective pixels, resulting in corrosion which interferes with super-resolution methods. These corrosions, especially around object boundaries, are then amplified, causing degraded quality. In addition, due to the complexity of real-world noise, existing denoising methods may yield erroneous optical flow estimations, which also harms the reconstruction.

As mentioned in Sec. II-B, here we also validate the compression noise caused by video codecs is negligible compared with the shooting noise. The results are shown in Table II. In the "no processing" row, we can see that codecs will lead to at most 0.02 dB in PSNR degradation, which is almost negligible.

TABLE III
PERFORMANCE OF THE FLOW EXTRACTOR USING DENOISED/NOISY FRAMES FOR TRAINING/VALIDATION. "NOISY" DENOTES RAW FRAMES WITH NOISE, "DENOISED" DENOTES FRAMES DENOISED BY [11]. THE "CLEAN" MODEL IS PRE-TRAINED BY [64].

Training set	Validation set	PSNR (in dB)
Noisy	Noisy	28.60
Noisy	Denoised	28.74
Denoised	Noisy	29.60
Denoised (ours)	**Denoised (ours)**	**30.45**
Clean (pre-trained)		27.13

Also, by applying all benchmarks as well as our method on coded and then decoded videos, the final video outputs show almost identical performance as that when codec is not used (at most 0.01 dB PSNR difference).

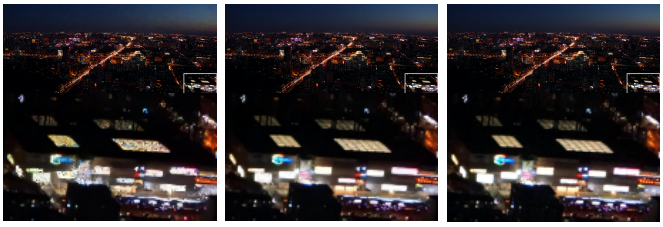
For the AWGN emulated results, the majority of the benchmarks yield slightly better reconstruction compared with the real-world noise cases. This is because they are trained with AWGN. Our method yields promising results on both real-world noise and the AWGN, with the performance under real-world noise being slightly better.

3) Running Time: We measure the running time of all schemes using the DERF dataset, with the same settings in Sec. IV-B1. The results are shown in Table I last column. The running time is averaged on 1,000 rounds of experiment. Our method benefits from reusing the propagated features from adjacent reconstructions. It achieves 5.43s delay to process 25 frames. Among the benchmark schemes, running super-resolution first (B1–B4) is generally slower than running denoising first (B5–B8). This is because the input resolutions of the denoising methods in benchmarks B1–B4 are larger than those in benchmarks B5–B8, with the input and output size of super-resolution methods being fixed. The best PSNR performance among the benchmark schemes is B8, but it yields 248.4 seconds delay (46 times). Our method shows the best running time compared with all benchmarks.

D. Ablation Studies

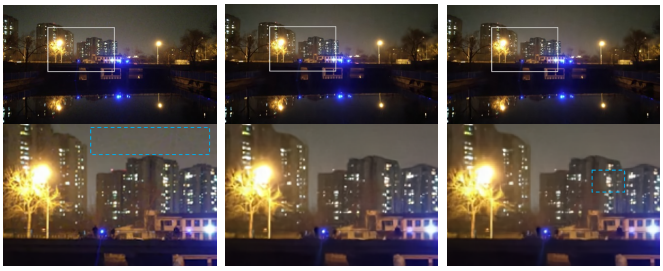
We conduct a series of ablation studies to discuss the key design choices and analyze the effectiveness of each component in the network. Unless otherwise mentioned, the settings are identical as introduced in Sec. IV-B1. We use the DERF dataset, and evaluate under noise level $\sigma = 20$.

1) The Effectiveness of the Attention Layer: To demonstrate the effectiveness of the attention layer proposed in the NRMA module in Sec. III-C3, we also experiment without the attention layer. We re-train the network and evaluate its performance. The network with attention layer achieves 30.45 dB in PSNR while the network without attention layer only achieves 30.32 dB in PSNR. We visually demonstrate a group of results using the realistic dataset in Fig. 6. The attention layer shows significant visual improvement here. The grid-shaped roof is nicely reconstructed with the attention layer, but is misinterpreted into holes without the attention layer. The vehicles at the bottom are also reconstructed more accurately with the attention layer.



(a) W/o attention layer. (b) With attention layer. (c) Ground-truth.

Fig. 6. Ablation study on the attention layer. The second row shows enlarged versions of the regions in the white boxes in the first row. Best viewed in digital format.



(a) 3 blocks. (b) 5 blocks (ours). (c) 7 blocks.

Fig. 7. Ablation study on the number of residual blocks in the NEU module. The second row shows enlarged versions of the regions in the white boxes in the first row. The artifacts are highlighted in the dashed blue boxes. Best viewed in digital format.

2) The Training of the Coarse Flow Extractor: To demonstrate the effectiveness of using denoised frames to train and validate the coarse flow extractor, we enumerate all possible combinations of noisy/denoised training/validation frames and evaluate. We also evaluate the existing pre-trained model [64] and demonstrate its insufficiency. The results are shown in Table III. Using frames denoised by [11] for both training and validation yields the best result. This is because the denoiser reduces but not completely eliminates the noise. By keeping the noise and artifacts at a low level, it minimizes the possible interference with the coarse flow extractor in the training phase. Therefore, we resort to this approach to train the coarse flow extractor.

3) The Number of Residual Blocks in the NEU Module: In the NEU module in Section III-D, we proposed a non-linear mapping module with five residual blocks. To examine the effectiveness of the non-linear mapping module, we compare it with various numbers of residual blocks. The module with 3 blocks yields 28.18 dB in PSNR, 5 blocks yield 30.45 dB in PSNR, and 7 blocks yield 30.25 dB in PSNR. It is demonstrated that the module with five residual blocks outperforms others, while others also yield acceptable performance. We also visually demonstrate a group of results using the realistic dataset in Fig. 7 for analysis. Since the non-linear mapping submodule is designated to cross-check the two candidate features, smaller numbers of residual blocks are insufficient to handle the features: some noises are still present after the cross-check process, and the visually reconstructed frame is rough, as shown in Fig. 7 (a). Larger numbers of residual blocks are over-complicated to cross-check only two candidate features, which preserve less sharpness (i.e., the light spot

in the blue boxes in Fig. 7 (c)), and yield slightly worse performance.

4) Other Details: The training details and more groups of visualized results are shown in the appendix [75].

V. Conclusion

In this work, we propose the joint video denoising and super-resolution network to enhance the quality of perceived video from IoT cameras. It uses a reformed pipeline to reduce the artifacts brought by the tandem of conventional denoising and super-resolution modules. An attention layer is added in the NRMA module to accurately reconstruct moving objects. We compare the proposed method with state-of-the-art benchmarks, and conduct comprehensive experiments under both existing datasets corrupted by AWGN and a realistic dataset (collected paired realistic noisy and ground-truth observations using real devices). Results demonstrate that the proposed method significantly outperforms the benchmarks, with ~ 7 dB PSNR gain in the existing datasets and ~ 1 dB gain in the realistic dataset (compared with the best benchmark).

References

[1] J. Wang, Y. Yu, S. Wu, C. Lei, and K. Xu, "Rethinking noise modeling in extreme low-light environments," in *IEEE International Conference on Multimedia and Expo*. IEEE, 2021, pp. 1–6.

[2] C. Chen, Q. Chen, J. Xu, and V. Koltun, "Learning to see in the dark," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3291–3300.

[3] R. D. Gow, D. Renshaw, K. Findlater, L. Grant, S. J. McLeod, J. Hart, and R. L. Nicol, "A comprehensive tool for modeling CMOS image-sensor-noise performance," vol. 54, no. 6. IEEE, 2007, pp. 1321–1329.

[4] W. Wang, X. Chen, C. Yang, X. Li, X. Hu, and T. Yue, "Enhancing low light videos by exploring high sensitivity camera noise," in *IEEE International Conference on Computer Vision*, 2019, pp. 4111–4119.

[5] Y. P. Loh and C. S. Chan, "Getting to know low-light images with the exclusively dark dataset," vol. 178. Elsevier, 2019, pp. 30–42.

[6] J. R. Mullen, R. C. Srinivasan, D. A. Tuckman, and W. C. Hammert, "How to shoot and edit high-quality surgical videos for hand and upper extremity surgery." Elsevier, 2021.

[7] P. Maharjan, L. Li, Z. Li, N. Xu, C. Ma, and Y. Li, "Improving extreme low-light image denoising via residual learning," in *IEEE International Conference on Multimedia and Expo*. IEEE, 2019, pp. 916–921.

[8] D. Ranks, "Camera sensors ranking," https://www.deviceranks.com/en/camera-sensor, 2022.

[9] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 6, pp. 1256–1272, 2016.

[10] A. Buades, J.-L. Lisani, and M. Miladinović, "Patch-based video denoising with optical flow estimation," vol. 25, no. 6. IEEE, 2016, pp. 2573–2586.

[11] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," vol. 26, no. 7. IEEE, 2017, pp. 3142–3155.

[12] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a fast and flexible solution for cnn-based image denoising," vol. 27, no. 9. IEEE, 2018, pp. 4608–4622.

[13] M. Zontak and M. Irani, "Internal statistics of a single natural image," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 977–984.

[14] M. Maggioni, G. Boracchi, A. Foi, and K. Egiazarian, "Video denoising, deblocking, and enhancement through separable 4-D nonlocal spatiotemporal transforms," vol. 21, no. 9. IEEE, 2012, pp. 3952–3966.

[15] H. Chen, Y. Jin, K. Xu, Y. Chen, and C. Zhu, "Multiframe-to-multiframe network for video denoising." IEEE, 2021.

[16] M. Tassano, J. Delon, and T. Veit, "DVDnet: A fast network for deep video denoising," in *IEEE International Conference on Image Processing*. IEEE, 2019, pp. 1805–1809.

[17] M. Tassano, J. Delon, and T. Veit, "FastDVDnet: Towards real-time deep video denoising without flow estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1354–1363.

[18] H. Chen, J. Wang, M. Duan, Y. Jin, Y. Kan, and C. Zhu, "Video denoising for scenes with challenging motion: A comprehensive analysis and a new framework," *IEEE Transactions on Multimedia*, 2022.

[19] P. K. Ostrowski, E. Katsaros, D. Węsierski, and A. Jezierska, "BP-EVD: Forward block-output propagation for efficient video denoising," *IEEE Transactions on Image Processing*, vol. 31, pp. 3809–3824, 2022.

[20] K. Monakhova, S. R. Richter, L. Waller, and V. Koltun, "Dancing under the stars: video denoising in starlight," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 241–16 251.

[21] M. Claus and J. van Gemert, "Videnn: Deep blind video denoising," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[22] S. Yu, B. Park, J. Park, and J. Jeong, "Joint learning of blind video denoising and optical flow estimation," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 500–501.

[23] V. Dewil, J. Anger, A. Davy, T. Ehret, G. Facciolo, and P. Arias, "Self-supervised training for blind multi-frame video denoising," in *IEEE Winter Conference on Applications of Computer Vision*, January 2021, pp. 2724–2734.

[24] D. Y. Sheth, S. Mohan, J. L. Vincent, R. Manzorro, P. A. Crozier, M. M. Khapra, E. P. Simoncelli, and C. Fernandez-Granda, "Unsupervised deep video denoising," in *IEEE International Conference on Computer Vision*, 2021, pp. 1759–1768.

[25] K. Wei, Y. Fu, J. Yang, and H. Huang, "A physics-based noise formation model for extreme low-light raw denoising," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2758–2767.

[26] C. Fogg, D. J. LeGall, J. L. Mitchell, and W. B. Pennebaker, *MPEG video compression standard*. Springer Science & Business Media, 2007.

[27] C. P. Fenimore, J. M. Libert, and P. Roitman, "Mosquito noise in MPEG-compressed video: test patterns and metrics," in *Human Vision and Electronic Imaging.*, vol. 3959. SPIE, 2000, pp. 604–612.

[28] F. Pan, X. Lin, S. Rahardja, W. Lin, E. Ong, S. Yao, Z. Lu, and X. Yang, "A locally-adaptive algorithm for measuring blocking artifacts in images and videos," in *IEEE International Symposium on Circuits and Systems.*, vol. 3. IEEE, 2004, pp. III–925.

[29] P. Hu, J. Im, Z. Asgar, and S. Katti, "Starfish: Resilient image compression for aiot cameras," in *Conference on Embedded Networked Sensor Systems*, 2020, pp. 395–408.

[30] T. Isobe, X. Jia, X. Tao, C. Li, R. Li, Y. Shi, J. Mu, H. Lu, and Y.-W. Tai, "Look back and forth: Video super-resolution with explicit temporal difference modeling," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 411–17 420.

[31] C. Zhou, Z. Lu, L. Li, Q. Yan, and J.-H. Xue, "How video super-resolution and frame interpolation mutually benefit," in *ACM International Conference on Multimedia*, 2021, pp. 5445–5453.

[32] W. Dong, F. Fu, G. Shi, X. Cao, J. Wu, G. Li, and X. Li, "Hyperspectral image super-resolution via non-negative structured sparse representation," vol. 25, no. 5. IEEE, 2016, pp. 2337–2352.

[33] L. Zhang, H. Zhang, H. Shen, and P. Li, "A super-resolution reconstruction algorithm for surveillance images," vol. 90, no. 3. Elsevier, 2010, pp. 848–859.

[34] K. C. Chan, S. Zhou, X. Xu, and C. C. Loy, "BasicVSR++: Improving video super-resolution with enhanced propagation and alignment," in *IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5972–5981.

[35] Z. Xiao, Z. Xiong, X. Fu, D. Liu, and Z.-J. Zha, "Space-time video super-resolution using temporal profiles," in *ACM International Conference on Multimedia*, 2020, pp. 664–672.

[36] R. Xu, Z. Xiao, M. Yao, Y. Zhang, and Z. Xiong, "Stereo video super-resolution via exploiting view-temporal correlations," in *ACM International Conference on Multimedia*, 2021, pp. 460–468.

[37] W. Wen, W. Ren, Y. Shi, Y. Nie, J. Zhang, and X. Cao, "Video super-resolution via a spatio-temporal alignment network," *IEEE Transactions on Image Processing*, vol. 31, pp. 1761–1773, 2022.

[38] K. C. Chan, X. Wang, K. Yu, C. Dong, and C. C. Loy, "BasicVSR: The search for essential components in video super-resolution and beyond," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4947–4956.

[39] Y. Tian, Y. Zhang, Y. Fu, and C. Xu, "TDAN: Temporally-deformable alignment network for video super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3360–3369.

[40] X. Wang, K. C. Chan, K. Yu, C. Dong, and C. Change Loy, "EDVR: Video restoration with enhanced deformable convolutional networks,"

in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[41] T. Isobe, F. Zhu, X. Jia, and S. Wang, "Revisiting temporal modeling for video super-resolution," 2020.

[42] J. Lee, M. Lee, S. Cho, and S. Lee, "Reference-based video super-resolution using multi-camera video triplets," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 824–17 833.

[43] T. Wang, J. Xie, W. Sun, Q. Yan, and Q. Chen, "Dual-camera super-resolution with aligned attention modules," in *IEEE International Conference on Computer Vision*, 2021, pp. 2001–2010.

[44] X. Yang, W. Xiang, H. Zeng, and L. Zhang, "Real-world video super-resolution: A benchmark dataset and a decomposition based learning scheme," in *IEEE International Conference on Computer Vision*, 2021, pp. 4781–4790.

[45] G. Bhat, M. Danelljan, L. Van Gool, and R. Timofte, "Deep burst super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9209–9218.

[46] M. Emad, M. Peemen, and H. Corporaal, "DualSR: Zero-shot dual learning for real-world super-resolution," in *IEEE Winter Conference on Applications of Computer Vision*, January 2021, pp. 1630–1639.

[47] K. C. Chan, S. Zhou, X. Xu, and C. C. Loy, "Investigating tradeoffs in real-world video super-resolution," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5962–5971.

[48] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883.

[49] A. ElTantawy and M. S. Shehata, "MARO: Matrix rank optimization for the detection of small-size moving objects from aerial camera platforms," vol. 12, no. 4. Springer, 2018, pp. 641–649.

[50] A. W. van Eekeren, K. Schutte, and L. J. van Vliet, "Super-resolution on small moving objects," in *IEEE International Conference on Image Processing*. IEEE, 2008, pp. 1248–1251.

[51] A. W. Van Eekeren, K. Schutte, and L. J. Van Vliet, "Multiframe super-resolution reconstruction of small moving objects," vol. 19, no. 11. IEEE, 2010, pp. 2901–2912.

[52] A. Xuehui, Z. Li, L. Zuguang, W. Chengzhi, L. Pengfei, and L. Zhiwei, "Dataset and benchmark for detecting moving objects in construction sites," vol. 122. Elsevier, 2021, p. 103482.

[53] S. Yu, Y. Zhang, C. Wang, X. Bai, L. Zhang, and E. R. Hancock, "HMFlow: Hybrid matching optical flow network for small and fast-moving objects," in *International Conference on Pattern Recognition*. IEEE, 2021, pp. 1197–1204.

[54] J. Choi, J. Park, and I. S. Kweon, "High-quality frame interpolation via tridirectional inference," in *IEEE Winter Conference on Applications of Computer Vision*, 2021, pp. 596–604.

[55] M. Hu, J. Xiao, L. Liao, Z. Wang, C.-W. Lin, M. Wang, and S. Satoh, "Capturing small, fast-moving objects: Frame interpolation via recurrent motion enhancement." IEEE, 2021.

[56] C.-H. Lu and B.-E. Shao, "Environment-aware multiscene image enhancement for internet of things enabled edge cameras," vol. 15, no. 3. IEEE, 2020, pp. 3439–3449.

[57] J. Yi, S. Choi, and Y. Lee, "EagleEye: Wearable camera-based person identification in crowded urban spaces," in *Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–14.

[58] A. Matsuda, T. Matsui, Y. Matsuda, H. Suwa, and K. Yasumoto, "A method for detecting street parking using dashboard camera videos," vol. 33, no. 1, 2021, pp. 17–34.

[59] J. Park, K. Ko, C. Lee, and C.-S. Kim, "BMBC: Bilateral motion estimation with bilateral cost volume for video interpolation," in *European Conference on Computer Vision*, 2020.

[60] H. Sim, J. Oh, and M. Kim, "XVFI: extreme video frame interpolation," in *IEEE/CVF international conference on computer vision*, 2021, pp. 14 489–14 498.

[61] H. V. Vo, N. Q. Duong, and P. Pérez, "Structural inpainting," in *ACM International Conference on Multimedia*, 2018, pp. 1948–1956.

[62] Z. Li, C.-Z. Lu, J. Qin, C.-L. Guo, and M.-M. Cheng, "Towards an end-to-end framework for flow-guided video inpainting," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 562–17 571.

[63] C. Du, B. Kang, Z. Xu, J. Dai, and T. Nguyen, "Accurate and efficient video de-fencing using convolutional neural networks and temporal information," in *IEEE International Conference on Multimedia and Expo*. IEEE, 2018, pp. 1–6.

[64] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-net: CNNs for optical flow using pyramid, warping, and cost volume," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8934–8943.

[65] Y. Huang, W. Wang, and L. Wang, "Video super-resolution via bidirectional recurrent convolutional networks," vol. 40, no. 4. IEEE, 2017, pp. 1015–1028.

[66] Z. Wang, P. Yi, K. Jiang, J. Jiang, Z. Han, T. Lu, and J. Ma, "Multi-memory convolutional neural network for video super-resolution," vol. 28, no. 5. IEEE, 2018, pp. 2530–2544.

[67] T. Isobe, X. Jia, S. Gu, S. Li, S. Wang, and Q. Tian, "Video super-resolution with recurrent structure-detail network," in *European Conference on Computer Vision*. Springer, 2020, pp. 645–660.

[68] S. Dutta, N. A. Shah, and A. Mittal, "Efficient space-time video super resolution using low-resolution flow and mask upsampling," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 314–323.

[69] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition workshops*, 2017, pp. 136–144.

[70] Xiph.org, "Derf's test media collection," https://media.xiph.org/video/derf, 2022.

[71] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *International conference on computer vision*. Ieee, 2011, pp. 2564–2571.

[72] *Information Technology–Coding of Audio-Visual Objects–Part 10: Advanced Video Coding*. Standard ISO/IEC 14496-10, 2003.

[73] *Information Technology–High Efficiency Coding and Media Delivery in Heterogeneous Environments–Part 2: High Efficiency Video Coding*. Standard ISO/IEC 23008-2, 2015.

[74] "Apple ProRes 422," https://support.apple.com/en-us/HT202410, 2023.

[75] https://www.dropbox.com/s/jthz0uf3hn55uxo/, 2023.