Rethinking Video Frame Interpolation from Shutter Mode Induced Degradation

Xiang Ji[1] Zhixiang Wang[1,2] Zhihang Zhong[1,2] Yinqiang Zheng[1†]
[1]The University of Tokyo, Japan [2]National Institute of Informatics, Japan
{jixiang,wangzhixiang}@g.ecc.u-tokyo.ac.jp,
zhong@is.s.u-tokyo.ac.jp, yqzheng@ai.u-tokyo.ac.jp

Abstract

Image restoration from various motion-related degradations, like blurry effects recorded by a global shutter (GS) and jello effects caused by a rolling shutter (RS), has been extensively studied. It has been recently recognized that such degradations encode temporal information, which can be exploited for video frame interpolation (VFI), a more challenging task than pure restoration. However, these VFI researches are mainly grounded on experiments with synthetic data, rather than real data. More fundamentally, under the same imaging condition, it remains unknown which degradation will be more effective toward VFI. In this paper, we present the first real-world dataset for learning and benchmarking degraded video frame interpolation, named RD-VFI, and further explore the performance differences of three types of degradations, including GS blur, RS distortion, and an in-between effect caused by the rolling shutter with global reset (RSGR), thanks to our novel quad-axis imaging system. Moreover, we propose a unified Progressive Mutual Boosting Network (PMBNet) model to interpolate middle frames at arbitrary time for all shutter modes. Its disentanglement strategy and dual-stream correction enable us to adaptively deal with different degradations for VFI. Experimental results demonstrate that our PMBNet is superior to the respective state-of-the-art methods on all shutter modes.

1. Introduction

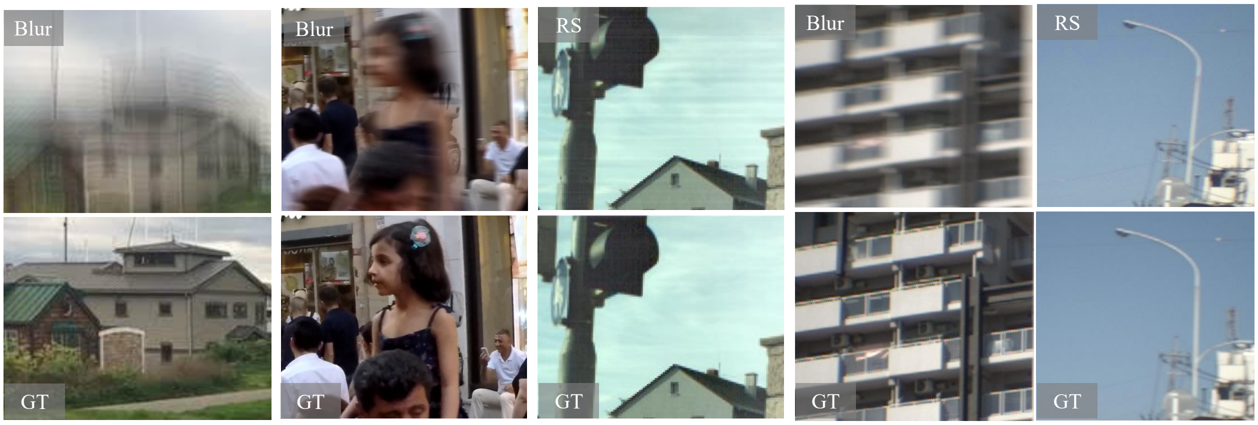
When high-speed cameras are not accessible, the perceiving and modeling of fast motion in video understanding can be challenging. One promising computational alternative is to up-convert low-framerate videos through video frame interpolation (VFI). Specifically, given two consecutive inputs, frame interpolation aims to reconstruct intermediate frames with temporal and spatial coherence, which has been addressed in existing VFI methods [13, 1, 12, 18, 29].

Unfortunately, despite the remarkable success, these approaches with sharp frames as input are less applicable, since image degradations are almost unavoidable in the presence of fast motion, which are closely related to shutter modes as well. Therefore, VFI with degraded inputs are of greater interest in practice, and it is even believed that the degradations encode rich information relating to motion, which might benefit VFI. Recent video restoration works have demonstrated this insight in VFI from single [15, 30] or multiple blurred inputs [14, 33, 3, 41, 28]. Meanwhile, the RS (rolling shutter) counterpart has also been conducted very recently [9, 11].

However, those VFI algorithms with degraded inputs, without exception, are evaluated on synthetic data only, and their performance in the real-world remains unknown. For example, the most prevalent datasets for blur (GoPRO [25], Adobe240fps [36]) or RS (Fastec-RS [20]) VFI cannot exactly mimic real captured degradations due to their oversimplified operations of blending consecutive frames or copying distinct scanlines from sharp frames. Such a synthesis method can easily lead to unnatural artifacts, as shown in Fig. 1, which is also mentioned in [32, 43]. Artifacts caused by unrealistic simulation tend to destroy degradations induced by motion and shutter modes, and the learned model has inferior generalization. Therefore, a real dataset without such synthesis artifacts is in immediate need. More importantly, VFI with GS blur or RS distortion has been studied independently, yet a unified study of these degradations under the same condition can reveal the advantages of different shutter modes for VFI, which is completely missing.

To address the aforementioned issues, we present the first real-world dataset for learning and benchmarking degraded video frame interpolation, which is dubbed RD-VFI. Inspired by the hardware design of [32, 43], we further propose a quad-axis imaging system to capture temporally and geometrically aligned high-speed sharp videos and low-speed degraded videos with three kinds of degradations, including GS blur, RS distortion, and a mixed effect caused by a rolling shutter with global reset (RSGR). RSGR [38] leads to blurry effects with varying magnitude, a special degradation lying between GS blur and RS distortion. Although RSGR video restoration has been explored in [38], its performance for

[†] Corresponding author



(a) Adobe240 (b) GoPRO (c) Fastec-RS (d) RD-VFI

Figure 1: **Samples from the existing synthetic dataset and our RD-VFI dataset.** (a) and (b) are samples from Adobe240fps and GoPRO, respectively. The unnatural hops and steps caused by the discontinuity of averaging process can be easily spotted. (c) are samples from Fastec-RS with horizontal streak artifacts. (d) are from our real-world dataset RD-VFI.

VFI remains unknown. Facilitated by our dataset, systematic comparisons between various degradations for VFI have been made to reveal their correlations or performance gaps.

Furthermore, we propose a unified model, Progressive Mutual Boosting Network (PMBNet), to interpolate middle clear frames at arbitrary time instances for all three exposure modes. PMBNet decouples the VFI task into correction and interpolation parts and reconnects them by latent variables and flow-guided feature alignment module (FFA) among the whole iterative layers with different scales. The dual-stream correction absorbs the merits of two classical paradigms from deblurring and RS correction, which enable our model to adaptively handle three types of degradation. Subsequently, interpolated candidate frame will be refined by a temporal and contextual compensation layer (TCL) to alleviate artifacts at the boundaries of dynamic objects and fill holes caused by occlusion.

In short, our contributions are:

- We present the first real-world dataset RD-VFI for video frame interpolation with degradations from three different shutter modes.
- Rather than a dual-axis system for single degradation, we develop a quad-axis imaging system that simultaneously captures three degradations and their high-speed ground truth, which allows direct comparison of different shutter modes for VFI.
- We introduce an original VFI task based on RSGR videos and a generic neural network architecture named PMBNet to adaptively handle different degradations, including GS blur, RS distortion, and RSGR effects.

2. Related Work

2.1. Sharp Video Frame Interpolation

The mainstream sharp video frame interpolation could be roughly classified into flow-based [23, 19, 13, 31, 29, 22, 26, 1] and non-flow based [4, 5, 8, 27, 24]. Non-flow based methods usually exploit phase information to learn the motion relationship [24] or formulate VFI as a spatially adaptive convolution [27, 4]. Tremendous efforts have been paid to increase the degree of freedom of convolution kernel [4, 5, 19], or combine with complementary operations [8, 2]. Choi *et al*. [7] attempt to employ a special feature reshaping operation, PixelShuffle, with channel attention to capture motion implicitly. Kim *et al*. [16] propose a joint VFI-SR framework for upscaling the spatio-temporal resolution by imposing temporal regularization.

Recently, significant progress has been made by interpreting motion as optical flow estimation. These methods are usually followed by forward or backward warping to generate intermediate frame candidate, and finally refined by a U-shaped network. SuperSlomo [13] linearly combines bidirectional optical flow to approximate intermediate flow and excludes the influence of occluded pixels to avoid artifacts by estimating visibility maps. Subsequently, Bao *et al*. [1] replace the linear approximation as weight map to estimate the intermediate flow. Niklaus *et al*. [26] propose SoftSplat to solve multiple source pixels mapping to the same target location under forward warping. [21] and [39] conduct quadratic flow prediction to overcome the limitations of linear models. RIFE [12] directly estimates intermediate flow by IFNet and a privileged distillation scheme is designed to stabilize training process. Kong *et al*. [18] jointly perform intermediate flow estimation and feature refinement to achieve efficient interpolation.

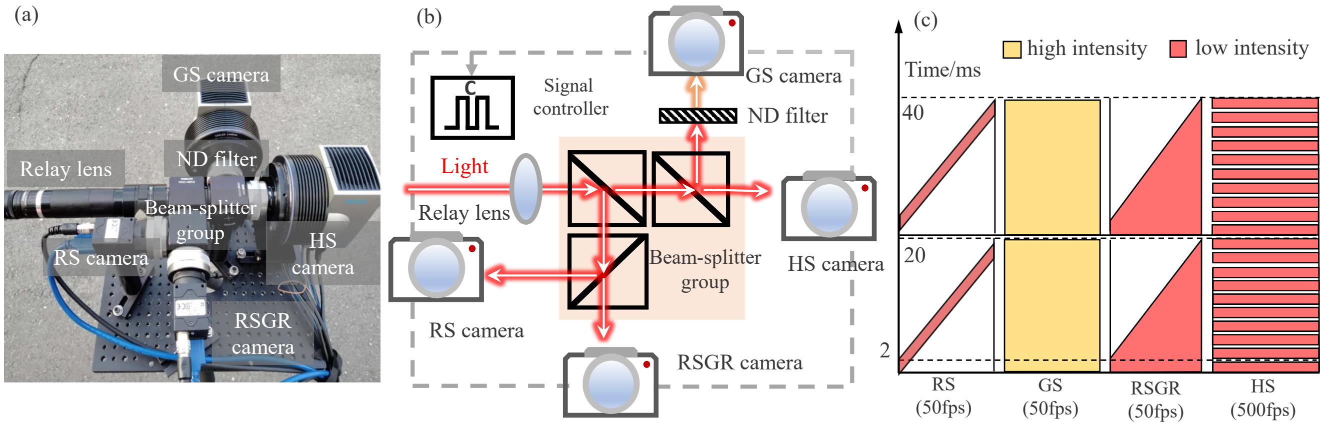


Figure 2: **Our quad-axis imaging system.** (a) the physical camera system exploited to capture our data. (b) abstracted optical diagram. (c) specific exposure comparisons among four modes of camera settings. The horizontal axis denotes image rows of all cameras.

2.2. Degraded Video Frame Interpolation

Current degraded video frame interpolation mainly focuses on inputs with GS blur or RS distortion. A few studies have addressed blur frame interpolation [15, 30, 14, 33, 41, 28]. Theses methods show their superiority over the simple cascading of deblurring and VFI. Jin *et al.* [15] first successfully retrieve latent sharp images from a single blurred image by introducing loss functions invariant to the temporal order. [30] further improves the reconstructed video performance from single input by learning motion representation through a surrogate task of video reconstruction. Jin *et al.* [14] address the temporal smoothness by simultaneously feeding deblurred key frames and blurry inputs to interpolation model. BIN [33] proposes a pyramid module to cyclically synthesize clear intermediate frames, which is later extended with larger size in [34]. All the mentioned methods have an obvious limitation that intermediate frames can only be interpolated either at central time [14, 33] or at a fixed scale factor [15, 30]. DeMFI [28] attempts to generate frames at arbitrary time based on flow-guided feature bolstering and recursive boosting.

In parallel to the evolution of blur frame interpolation, researches have also been conducted on RS counterpart [9, 11]. RSSR [9] recovers a high framerate GS video from consecutive RS images by exploiting the geometric constraint in the RS camera model. CVR [11] further approximates the bilateral motion field and proposes a context-aware video reconstruction architecture to deal with missing regions and motion artifacts.

3. RD-VFI Dataset

3.1. Limitation of Synthetic Dataset

Basically, the synthesis of blur follows the protocol of [33, 34, 28] by averaging consecutive frames in a window with constant size and stride based on GOPRO [25] or Adobe240 [36]. Similarly, Fastec-RS [20] is synthesized by sequentially copying a scanline from global shutter images. As discussed in Sec. 1, synthesized data can barely simulate the actual distribution of real degradation. The hops and steps are apparent as presented in Fig. 1(a-b), and synthesized RS frames also suffer from horizontally repeated streaks shown in Fig. 1(c). Besides, recent studies [32, 43] have demonstrated real-world data can significantly improve models' performance and generalization.

Table 1: **Details of capture setting used in our quad-axis imaging system**. The deadtime between two adjacent high-speed frames is extremely short and thus ignored.

Device	Resolution	Frame rate	Exposure /row	Delay. /row	Exposure /frame
RS camera	640×480	50 fps	2 ms	37.5 us	20 ms
RSGR camera	640×480	50 fps	2∼20 ms	37.5 us	20 ms
GS camera	640×480	50 fps	20 ms	0 us	20 ms
HS camera	640×480	500 fps	2 ms	0 us	2 ms

3.2. Quad-axis Imaging System

For collecting a realistic dataset of strictly aligned RS, RSGR, GS, and high-speed videos, we extend settings of [32, 43] and construct our quad-axis imaging system. With the assistance of this equipment, we for the first time bridge the gap among three modes of degraded VFI and enable direct comparison of them. In Fig. 2 and Tab. 1, we present the assembly of our proposed system and specific parameter settings. Four cameras, including a GS camera (BITRAN CS-700C), an RS camera (FLIR BFS-U3-63S4C with 4x4 binning), an RSGR camera (FLIR BFS-U3-63S4C with 4x4 binning, an RS camera that also supports RSGR mode) and a high-speed GS camera (BITRAN CS-700C) with forced cooling, are spatially aligned with a group of beamsplitters and temporally synchronized through an external trigger to

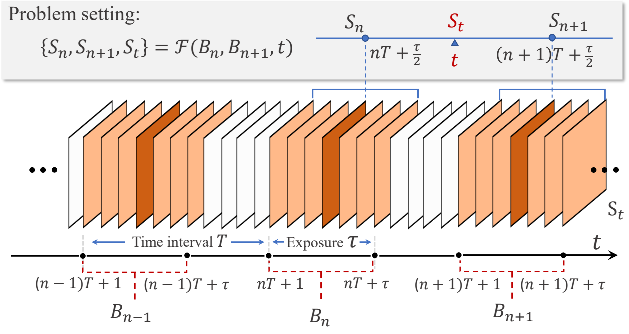


Figure 3: **Degraded imaging process and our problem setting.**

capture consecutive frames simultaneously. We put a neutral density (ND) filter of about 10% transmittance before the GS camera, such that all captured images have almost equalized brightness. More details on used devices, geometrical alignment and temporal synchronization can be found in supplementary materials.

3.3. Collected Data

By holding our customized imaging system in hand, we collected our real-world degraded video frame interpolation (RD-VFI) dataset with four modes: RS, RSGR, GS blur, and high-speed (HS) sharp videos. RD-VFI is diverse, covering dynamic street scenes and complex camera egomotion. Samples are presented in Fig. 1(d) and direct visual comparisons of shutter mode-induced degradations could be found in supplemental martial. We collect 133 video quadruples and each quadruple has $60{\times}3$ degraded frames and 600 sharp HS frames. We further divide them into 85, 13, and 35 sequences as our training, validating, and testing set, respectively. In addition, the corresponding raw format is also available for further research. As illustrated in Fig. 2(c), the total duration per frame is 20 ms for RS, RSGR, and GS cameras. As for the high-speed (HS) camera, it runs at 500fps with an exposure time of 2ms, which means that there are 10 sharp frames corresponding to one degraded frame (RS, RSGR, or GS blur).

4. Unified VFI Method

4.1. Background and Problem Formulation

First, we briefly summarize three types of degraded imaging processes in Fig. 3. T is the total time duration and τ is exposure time. B denotes degraded video frame while S is latent high-speed frame. During the exposure, camera sensors constantly receive light, and each instantaneous sharp stimulation is accumulated, generating blurry images B^{bl}. Most of the literature [6, 37, 28] approximate this process as:

$$B_n^{bl} = \frac{1}{\tau}\sum_{k=1}^{\tau} S_{nT+k}\,. \tag{1}$$

We formulate the RS and RSGR degradation B^{rs}, B^{rg} as:

$$B_n^{rs}[k] = S_{nT+k}[k] \tag{2}$$

$$B_n^{rg}[k] = S_{nT+1}[k] + \delta\sum_{i=2}^{k} S_{nT+i}[k], \tag{3}$$

where $[k]$ denotes extracting k^{th} row of the frame and δ approximates the ratio between readout time and the shortest exposure duration (*i.e.*, first scanline's).

To better compare three types of degraded VFI tasks, we fix the problem setting[1] (Fig. 3) as:

$$\{S_n, S_{n+1}, S_t\} = \mathcal{F}(B_n, B_{n+1}, t)\,, \tag{4}$$

Its superiority has been demonstrated in [28, 11, 10]. Where $\mathcal{F}$ represents an interpolation model takes two consecutive degraded images B_n, B_{n+1} and the interpolation time instance $t \in (nT + \tau/2, n(T+1) + \tau/2)$ as inputs. We assume that the sharp counterparts of two inputs are those latent frames corresponding to their middle scanlines. The model $\mathcal{F}$ will finally correct B_n, B_{n+1} to S_n, S_{n+1} and interpolate S_t at time instance t.

4.2. PMBNet

Considering the complexity of real situation, we propose a generic model that simultaneously handles RS effects, blur, and RSGR distortion. Because, there might exist multiple cameras with different shutter modes, or single camera that can switch between different modes. Without a generic model, different shutter-specific models have to be deployed and maintained. We believe our generic model can help save efforts and costs. The proposed PMBNet incorporates the advantages of both RS correction and motion deblurring building blocks in a progressive mutual boosting manner. As shown in Fig. 4, the entire VFI task is disentangled into correction and interpolation branches with effective way of interaction by flow-guided feature alignment (FFA) and intermediate latent variables. The bidirectional flow prediction module (BFP) first provides flow maps to align features of non-warping deblurring module (NWD). Then corrected inputs from two paradigms are fused through deformable attention (DA) and returned to interpolation branch for next iteration. We gradually update the corrected inputs and bidirectional flow maps from coarse to fine by using multi PMB-blocks with corresponding $scales = [\frac{1}{4}, \frac{1}{2}, 1]$.

Disentanglement and Progressive Mutual Boosting In recent research of VFI from GS blur or RS distortion [14, 42], a cascaded solution that interpolates on outputs of existing methods for deblurring or RS correction is presented. However, a naive cascading of these methods is sub-optimal and unable to take full advantage of cues hidden in the degradation. Thus, Jin *et al.* [14] decouples blur video frame

[1] Note that our dataset supports other evaluation settings.

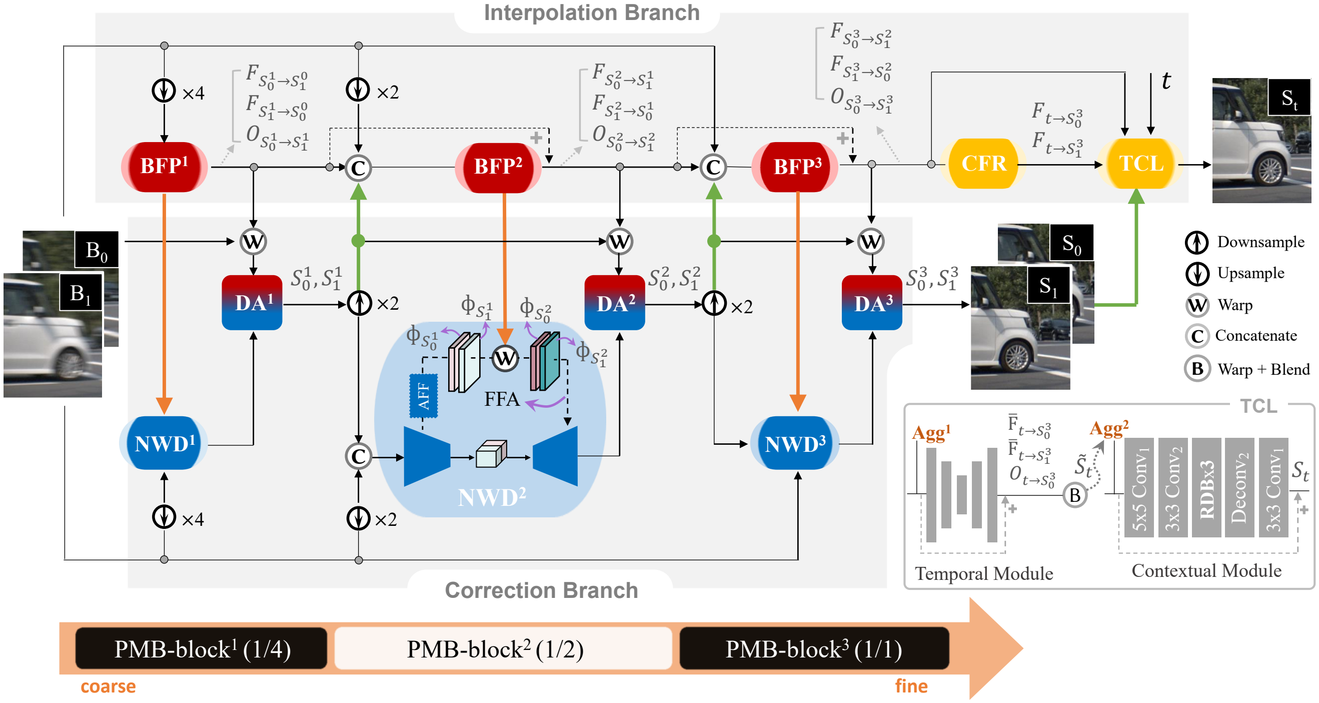


Figure 4: **Network architecture of PMB**. Our proposed model takes two consecutive degraded video frames as input to reconstruct corresponding sharp ones and interpolates the middle frame at arbitrary times. The bidirectional flow prediction module (BFP) first provides flow maps to align features of non-warping deblurring module (NWD). Then corrected inputs from two paradigms are fused through deformable attention (DA) and returned to interpolation branch for next iteration. Finally, corrected inputs will be exploited to boost the interpolation performance through the temporal and contextual layer (TCL). Agg^1= $[F_{S_0^3\to S_1^2}, F_{S_1^3\to S_0^2}, O_{S_0^3\to S_1^3}, F_{t\to S_0^3}, F_{t\to S_1^3}, t, S_0^3, S_1^3]$ and Agg^2= $[F_{S_0^3\to S_1^2}, F_{S_1^3\to S_0^2}, O_{t\to S_0^3}, \bar{F}_{t\to S_0^3}, \bar{F}_{t\to S_1^3}, t, S_0^3, S_1^3, S_{0t}, S_{1t}, \tilde{S}_t]$.

interpolation into two *fully independent* modules: DeblurNet and InterpNet. DeblurNet generates sharp keyframes, and InterpNet interpolates middle frames based on keyframes and blurry inputs. Different from them, we disentangle the VFI problem as two *mutually boosted* branches: correction and interpolation through the FFA module and latent variables. **1)** The interpolation branch first estimates the bidirectional flow by BFP. **2)** Then, the flow will be used for aligning features from the encoder and decoder of the non-warping deblurring module (NWD). **3)** Finally, the corrected frames obtained by merging outputs of NWD and flow-warping will be returned to the interpolation branch for further enhancement.

Existing flow-based correction or degraded VFI methods [28, 11, 18, 42] normally formulate the motion part as an iterative flow estimation:

$$\begin{aligned}\{F_{S_0^i\to S_1^i}, F_{S_1^i\to S_0^i}\} = \\ \text{FlowNet}(B_0, B_1, F_{S_0^{i-1}\to S_1^{i-1}}, F_{S_1^{i-1}\to S_0^{i-1}}),\end{aligned} \tag{5}$$

with iteration index $i = 2, 3, ..., K$. Modeling bidirectional flow of latent sharp frames solely based on degraded frames is nontrivial. Therefore, we propose a progressive manner to predict flow using refined inputs:

$$\begin{aligned}\{F_{S_0^i\to S_1^{i-1}}, F_{S_1^i\to S_0^{i-1}}, S_0^i, S_1^i\} = \text{PMB-block}(B_0, B_1, \\ S_0^{i-1}, S_1^{i-1}, F_{S_0^{i-1}\to S_1^{i-2}}, F_{S_1^{i-1}\to S_0^{i-2}}),\end{aligned} \tag{6}$$

with $i = 2, 3, ..., K$, $S_0^0 = B_0$, and $S_1^0 = B_1$. The PMB-block begins with initial degraded frames. It first estimates coarser flow maps and rectified inputs, based on which the block could improve the outcomes further. By repeating these steps, we will finally get satisfied bidirectional flows and corrected frames.

Dual-stream Correction The dual correction module takes in the merits of two classical paradigms: bidirectional flow warping (BFP) estimates displacements to correct RS distortion while non-warping deblurring (NDW) structure directly reconstructs sharp outputs in an encoder-decoder fashion. Recovered frames from two streams will be then fused through deformable attention (DA) to make a trade-off between deblurring and RS correction, which equips our model with the ability of adaptively handling GS blur, RS distortion or coexistence of them (i.e. RSGR effects). We follow RIFE [12] and MIMO-UNet [6] to construct building blocks of our BFP and NDW modules respectively. For the i^{th} iteration, bidirectional flows are calculated by:

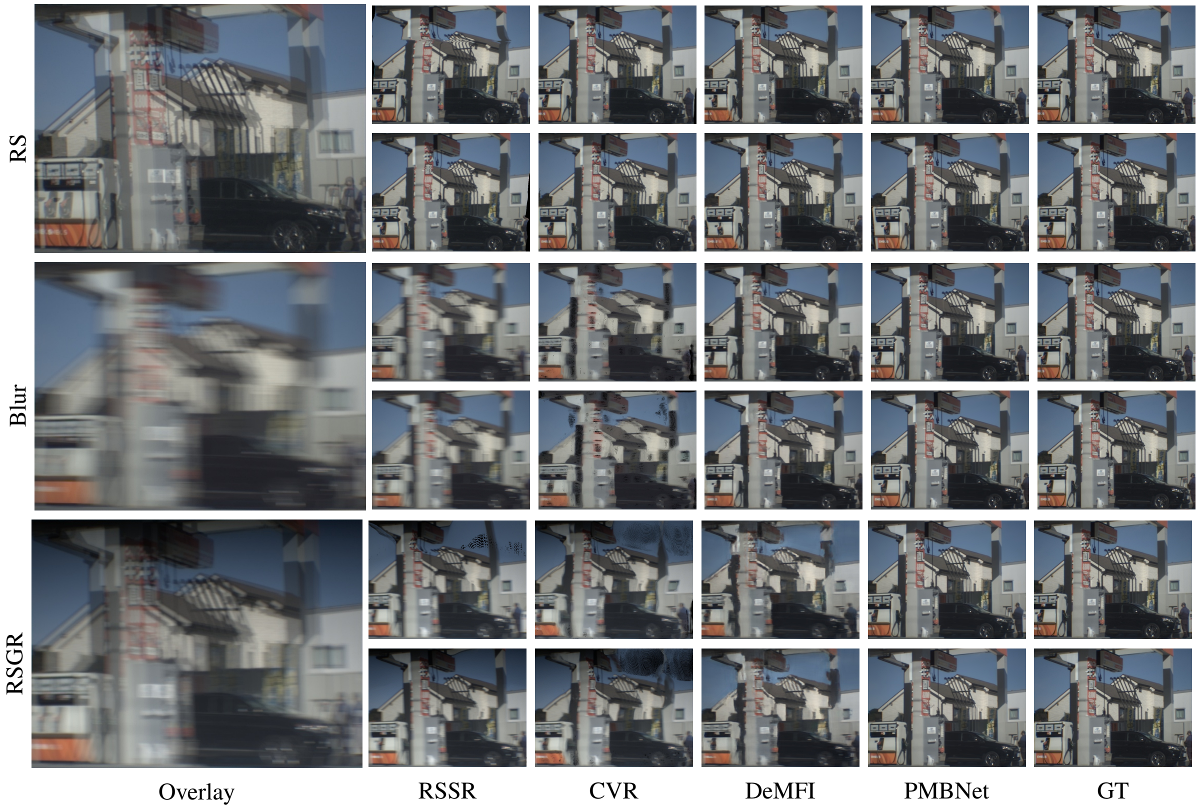


Figure 5: **Visual comparison.** We compare VFI results by different methods with RS, RSGR and GS blur degradations, respectively. In each mode, we present the results of $S_{1/10}$ (first row) and $S_{9/10}$ (second row).

$$\{F_{S_0^i \to S_1^{i-1}}, F_{S_1^i \to S_0^{i-1}}, O_{S_0^i \to S_1^i}\} = \text{BFP}(B_0, B_1, S_0^{i-1}, S_1^{i-1}, F_{S_0^{i-1} \to S_1^{i-2}}, F_{S_1^{i-1} \to S_0^{i-2}}, O_{S_0^{i-1} \to S_1^{i-1}})\,. \tag{7}$$

O denotes occlusion map. Then the results from RS correction stream are generated by backward-warping W_b:

$$\begin{aligned} \tilde{S}_0^i &= W_b(F_{S_0^i \to S_1^{i-1}}, S_1^{i-1}) \\ \tilde{S}_1^i &= W_b(F_{S_1^i \to S_0^{i-1}}, S_0^{i-1})\,. \end{aligned} \tag{8}$$

We omit the refinement process after warping in the above equations for simplicity. On the other hand, corrected results of the deblurring stream are computed as:

$$\{\bar{S}_0^i, \bar{S}_1^i\} = \text{NWD}(B_0, B_1, S_0^{i-1}, S_1^{i-1}, F_{S_0^i \to S_1^{i-1}}, F_{S_1^i \to S_0^{i-1}})\,, \tag{9}$$

where the flow maps from BFP are used to align features between encoder and decoder through FFA. Similar to Eq. 8, the feature $\Phi_{S_0^{i-1}}$,$\Phi_{S_1^{i-1}}$ integrated by asymmetric feature fusion (AFF) [6] from scale-variant encoders are aligned to $\Phi_{S_0^i}$,$\Phi_{S_1^i}$, then fused with corresponding decoder output Φ_{S^i} by:

$$\bar{\Phi}_{S^i} = \text{Conv}([\Phi_{S_0^i}, \Phi_{S_1^i}] \otimes \Phi_{S^i}) + \Phi_{S^i}\,, \tag{10}$$

where $[\cdot]$ means concatenation and $\otimes$ denotes element-wise multiplication. Finally, results $\{\tilde{S}_0^i, \tilde{S}_1^i\}$ and $\{\bar{S}_0^i, \bar{S}_1^i\}$ are merged by a deformable attention (DA) to generate $\{S_0^i, S_1^i\}$.

Temporal and Contextual Compensation As discussed in [11], flow-based warping methods tend to cause holes or misalignment due to heavy occlusions or partially moving objects. Therefore, we propose a temporal and contextual compensation layer (TCL) to alleviate the artifacts and enhance temporal accuracy of interpolation. We first linearly approximate $F_{S_0^3 \to t}, F_{S_1^3 \to t}$ from $F_{S_0^3 \to S_1^2}, F_{S_1^3 \to S_0^2}$ then reverse them to $F_{t \to S_0^3}, F_{t \to S_1^3}$ by using complementary flow reversal (CFR) [35]. As shown in Fig. 4, the temporal module further refines intermediate flows and occlusion map based on collected variable Agg^1 to get $\bar{F}_{t \to S_0^3}, \bar{F}_{t \to S_1^3}$ and $O_{t \to S_0^3}$. Next, we calculate the candidate intermediate frame $\tilde{S}_t$ by:

$$\tilde{S}_t = \frac{(1-t) \cdot O_{t \to S_0^3} \cdot S_{0t} + t \cdot (1 - O_{t \to S_0^3}) \cdot S_{1t}}{(1-t) \cdot O_{t \to S_0^3} + t \cdot (1 - O_{t \to S_0^3})} \tag{11}$$

$$S_{0t} = W_b(\bar{F}_{t \to S_0^3}, S_0^3)\,, \quad S_{1t} = W_b(\bar{F}_{t \to S_1^3}, S_1^3)\,. \tag{12}$$

Last, the contextual module integrates contextual, motion, and temporal information (Agg^2) to interpolate the final middle frame S_t.

Objective The total loss function is given by:

$$
\begin{aligned}
\mathcal{L}_{total} &= \lambda \mathcal{L}_{corr} + \beta \mathcal{L}_{intr} \\
\mathcal{L}_{corr} &= \frac{1}{K} \sum\nolimits_{i=1}^{K} \sum\nolimits_{j\in(0,1)} \|S_j^i - G_j^i\|_1 \\
\mathcal{L}_{intr} &= \|S_t - G_t\|_1,
\end{aligned}
\tag{13}
$$

where λ, β are weights to balance $\mathcal{L}_{corr}$ and $\mathcal{L}_{intr}$. We set $\lambda = \beta = 1$ and $K = 3$ in our implementation.

5. Experimental Results

Implementation details As described in Sec. 3, each degraded frame corresponds to 10 high-speed sharp frames. So, the interpolation time instances could be multiples of $1/10$ with $0 < t < 1$. Every training sample consists of two consecutive degraded inputs (B_0, B_1), a time instance t, and corresponding ground-truth frames (G_0, G_1, G_t), where t is randomly selected from all candidates when training and validating. We train our PMBNet using Adam optimizer [17] with an initial learning rate of 10^{-4}, which reduces to 10^{-6} by a cosine annealing scheduler. The total training epoch is empirically set as 800. We adopt 480×480 random crop and horizontal flip to augment training data. Experiments are performed on two NVIDIA GeForce RTX 3090 GPUs with a batch size of 8. Standard metrics PSNR, SSIM, and LPIPS [40] are applied.

Comparison with SOTA Methods We focus on the problem setting illustrated in Fig. 3 so as to better reveal the characteristics of different shutter modes for VFI. Therefore, we compare our generic model with existing SOTA methods that could be directly adapted to the aforementioned setting: DeMFI [28], CVR [11] and RSSR [9]. We also note that other close works like UTI-VFI [41], TNTT [14], and BIN [33] are devised for central interpolation based on blurry videos, which limits their capability to interpolation at a fixed time and cannot distinguish deadtime from exposure duration. So, these methods can not be directly compared with models that only present intra-frame interpolation (like CVR and RSSR). Although DeMFI is designed for blurry video frame interpolation and the RSSR and CVR are for RS VFI, to better demonstrate performance and adaptability to all three degradations, we retrain them on all three modes (*i.e*. RS, RSGR, and GS blur) of RD-VFI dataset. In addition, the reported experimental results are obtained by setting $t = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]$ for each pair of consecutive inputs.

Tab. 2 shows quantitative comparisons of all methods on RS, RSGR, and GS blur modes in terms of correction and interpolation. Overall, all methods achieve the highest performance on GS blur mode but obtain the largest relative improvements on RSGR mode from the perspective of correction. Note that CVR and RSSR are constructed on RS-aware geometry but blur apparently violates this assumption. So, their performance sharply degrades on RSGR and blur mode, even worse than the initial performance on the correction part. Furthermore, the significant performance gaps among initial inputs of RS, RSGR, and blur (26.97 *vs*. 17.39 *vs*. 26.46) could have an impact on the final metrics. If the initial differences are relatively small, RSGR would be the best choice for the VFI task due to its prominent performance gains. Also, the performances of RSSR, CVR, and DeMFI on their corresponding modes of RD-VFI are far inferior to those reported in original papers using synthetic data, which reminds us of the nontrivial gap between simulation and real data.

Benefiting from the progressive mutual boosting structure, our PMBNet is capable of dealing with different degradations and outperforms SOTA methods on three modes of RD-VFI. Interpolated frames at time of $1/10$ and $9/10$ are presented in Fig. 5. Existing methods present mitigated distortions, yet local details and structures are still not restored sufficiently, while our results are much clearer. We also computed the complexity of all algorithms (Tab. 3). FLOPs and running time are measured by interpolating one frame (256×256) on an NVIDIA Geforce RTX 3090. We could see that the complexity of our model is at a medium level.

Additionally, we also provide a synthetic dataset named GOPRO-VFI as supplementary to our real data. The detailed synthesizing process and experimental results are presented in the supplemental material.

Video Reconstruction Results We apply our model to generate multiple in-between frames at arbitrary times on all shutter modes. The visual results are shown in Fig. 6. Besides correcting distortion, our PMBNet can reconstruct temporally and spatially consistent frames. Video demos are also presented in the supplemental material for comparison.

Thirdparty Evaluation To demonstrate the need for a real dataset, we additionally collected third-party data with different settings in resolution, framerate, exposure time, and deadtime. As shown in Fig. 8, The model trained on real data outperforms that trained on synthetic one in generalization and performance.

Ablation Study To analyze the effectiveness of each component in our model, we perform ablation studies. Tab. 4 and Fig. 7 show the experimental results of: v_1 (PMBnet without TCL), v_2 (PMBNet without FFA), v_3 (single-stream correction with flow warping structure) and v_4 (single-stream correction with nonwarping deblurring structure). It is noticed that the combination of two correction paradigms will contribute to a performance gain of at least 0.33 dB. TCL and FFA modules could also make significant performance gains compared to the baseline model.

Table 2: **Quantitative comparisons on three modes of RD-VFI dataset**. We compare our model with SoTA methods for degraded video frame interpolation. We report the mean PSNR (↑) / SSIM (↑) / LPIPS (↓) scores. Bold numbers represent the best performance. We provide the evaluation metrics of correction, VFI (10 times interpolation), and average. $B0$, $B1$ denote the initial performance of inputs. Correction metrics are computed from B_0, B_1 and S_0, S_1 while the interpolation part is obtained by averaging all intermediate frames S_t.

Methods	**RS Mode**			**RSGR Mode**			**Blur Mode**		
	Correction	VFI (×10)	Average	Correction	VFI (×10)	Average	Correction	VFI (×10)	Average
B_0, B_1	26.96 / 0.858 / 0.047	– / – / –	– / – / –	17.39 / 0.771 / 0.150	– / – / –	– / – / –	26.46 / 0.867 / 0.182	– / – / –	– / – / –
RSSR [9]	27.61 / 0.890 / 0.059	21.06 / 0.722 / 0.181	21.71 / 0.739 / 0.169	17.62 / 0.751 / 0.200	16.21 / 0.694 / 0.241	16.35 / 0.700 / 0.237	25.68 / 0.851 / 0.206	22.95 / 0.790 / 0.246	23.22 / 0.797 / 0.242
CVR [11]	27.22 / 0.873 / 0.057	28.20 / 0.889 / 0.057	28.11 / 0.887 / 0.057	17.91 / 0.737 / 0.227	17.79 / 0.733 / 0.230	17.81 / 0.733 / 0.230	25.79 / 0.845 / 0.179	25.64 / 0.841 / 0.182	25.66 / 0.841 / 0.181
DeMFI [28]	27.95 / 0.888 / 0.055	27.84 / 0.887 / 0.070	27.85 / 0.887 / 0.068	24.27 / 0.845 / 0.138	24.21 / 0.844 / 0.152	24.21 / 0.844 / 0.151	30.37 / 0.915 / 0.093	30.53 / 0.918 / 0.088	30.51 / 0.918 / 0.088
PMBNet	**29.08 / 0.905 / 0.041**	**29.09 / 0.904 / 0.044**	**29.09 / 0.904 / 0.043**	**26.10 / 0.874 / 0.070**	**26.03 / 0.874 / 0.069**	**26.037 / 0.870 / 0.069**	**31.23 / 0.929 / 0.051**	**31.27 / 0.930 / 0.049**	**31.27 / 0.930 / 0.050**

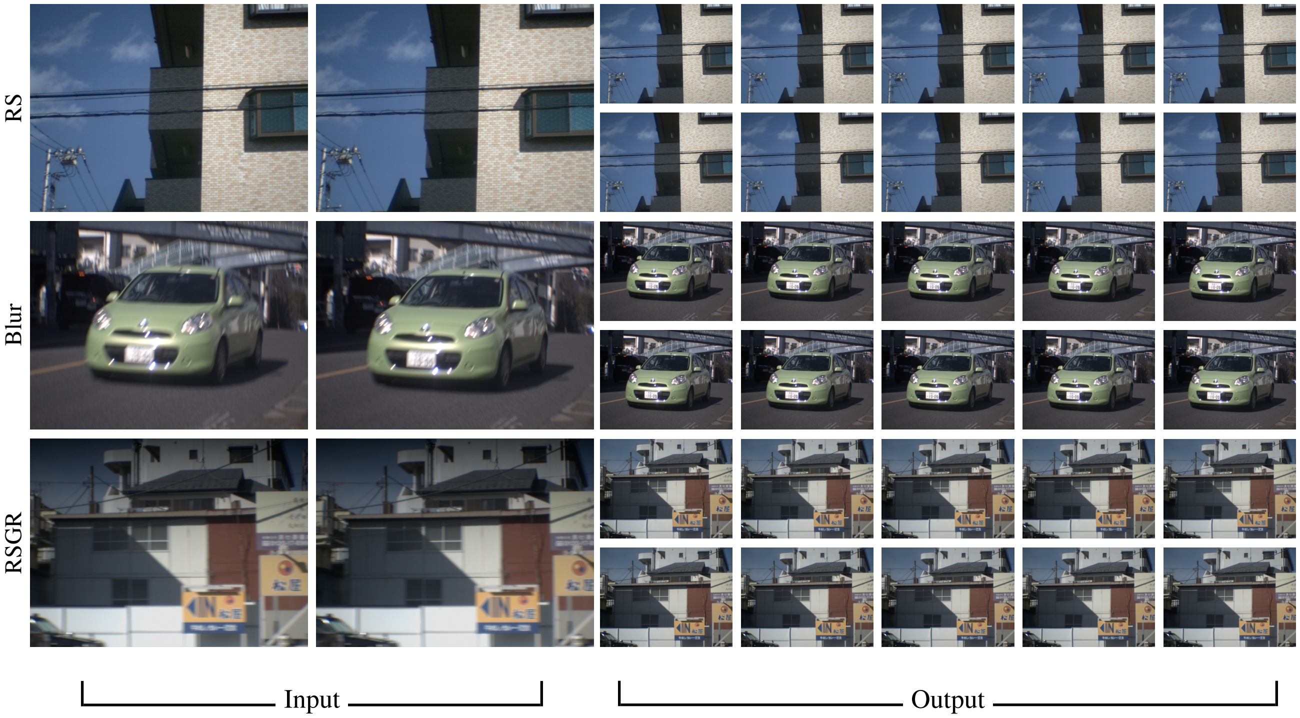


Figure 6: **Reconstructed consecutive frames from two degraded inputs by using our PMBNet.** We present the multiple intermediate frames at different times generated by three types of shutter-induced degradations. They are temporally located at $t = [0, 1)$ with stride of 0.1 and arranged in two rows from left to right. *Best viewed in zoom.*

Table 3: **Computational complexity comparison.** We also compute the complexity of all models in term of running time, number of parameters and FLOPs to make a better comparison.

Methods	Time (s)	Params (M)	FLOPs (G)
RSSR [9]	0.47	26.03	42.67
CVR [11]	0.48	42.70	101.05
DeMFI [28]	0.53	7.41	420.02
PMBNet	0.49	30.10	121.21

Table 4: **Model ablation** on GS blur mode of RD-VFI dataset. †NWDB is a non-warping deblur method.

Variants	**PSNR** (↑)	**SSIM** (↑)	**LPIPS** (↓)
PMBNet w/o TCL (v1)	30.63	0.9170	0.0893
PMBNet w/o FFA (v2)	31.03	0.9269	0.0525
SSC-flow warping (v3)	30.36	0.9137	0.0808
SSC-NWDB† (v4)	30.94	0.9267	0.0538
PMBNet (full)	31.27	0.9299	0.0496

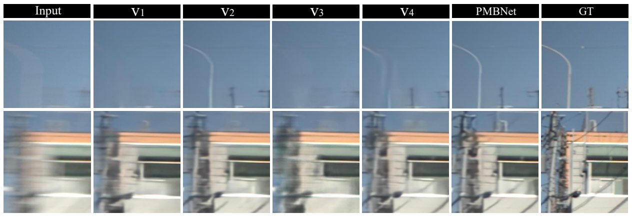


Figure 7: **Qualitative results for model ablation.** Obviously, the full model is capable of reconstructing sharper details than other variants.

6. Conclusion

In this paper, we highlighted the effects of shutter mode-induced degradations on the VFI task. By developing a novel quad-axis imaging system, we were able to present the first real word dataset with strictly aligned RS, RSGR, and GS

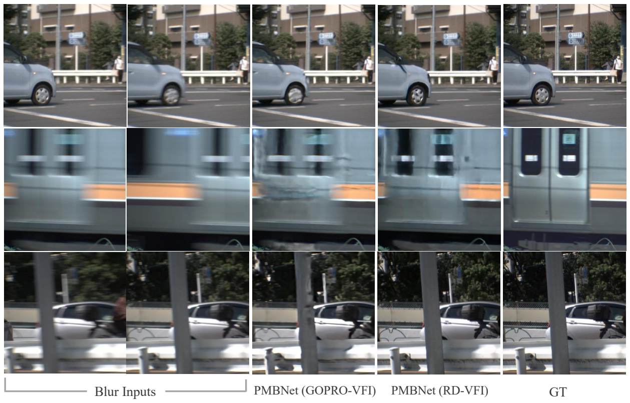


Figure 8: **Thirdpart evaluation.** We present comparisons on another collected real data with different settings by using PMBNet trained on RD-VFI and GOPRO-VFI, respectively.

blur videos and high-speed reference videos, which allows direct comparison of different algorithms and shutter modes. We also proposed a generic model to interpolate middle clear frames at arbitrary times by disentangling the task into correction and interpolation parts with mutual boosting. The experimental results validated our PMBNet is capable of adaptively handling different degradations and revealed new observations regarding different shutter modes for VFI.

Acknowledgement. This research was supported in part by JSPS KAKENHI Grant Numbers 22H00529, 20H05951, the Value Exchange Engineering, a joint research project between Mercari, Inc. and RIISE.

References

[1] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3703–3712, 2019. 1, 2

[2] Wenbo Bao, Wei-Sheng Lai, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Memc-net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement. *IEEE transactions on pattern analysis and machine intelligence*, 43(3):933–948, 2019. 2

[3] Jiezhang Cao, Jingyun Liang, Kai Zhang, Wenguan Wang, Qin Wang, Yulun Zhang, Hao Tang, and Luc Van Gool. Towards interpretable video super-resolution via alternating optimization. *arXiv preprint arXiv:2207.10765*, 2022. 1

[4] Xianhang Cheng and Zhenzhong Chen. Video frame interpolation via deformable separable convolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10607–10614, 2020. 2

[5] Xianhang Cheng and Zhenzhong Chen. Multiple video frame interpolation via enhanced deformable separable convolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[6] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4641–4650, 2021. 4, 5, 6

[7] Myungsub Choi, Heewon Kim, Bohyung Han, Ning Xu, and Kyoung Mu Lee. Channel attention is all you need for video frame interpolation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10663–10671, 2020. 2

[8] Tianyu Ding, Luming Liang, Zhihui Zhu, and Ilya Zharkov. Cdfi: Compression-driven network design for frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8001–8011, 2021. 2

[9] Bin Fan and Yuchao Dai. Inverting a rolling shutter camera: bring rolling shutter images to high framerate global shutter video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4228–4237, 2021. 1, 3, 7, 8

[10] Bin Fan, Yuchao Dai, and Mingyi He. Sunet: symmetric undistortion network for rolling shutter correction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4541–4550, 2021. 4

[11] Bin Fan, Yuchao Dai, Zhiyuan Zhang, Qi Liu, and Mingyi He. Context-aware video reconstruction for rolling shutter cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17572–17582, 2022. 1, 3, 4, 5, 6, 7, 8

[12] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 5

[13] Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9000–9008, 2018. 1, 2

[14] Meiguang Jin, Zhe Hu, and Paolo Favaro. Learning to extract flawless slow motion from blurry videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8112–8121, 2019. 1, 3, 4, 7

[15] Meiguang Jin, Givi Meishvili, and Paolo Favaro. Learning to extract a video sequence from a single motion-blurred image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6334–6342, 2018. 1, 3

[16] Soo Ye Kim, Jihyong Oh, and Munchurl Kim. Fisr: Deep joint frame interpolation and super-resolution with a multi-scale temporal loss. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11278–11286, 2020. 2

[17] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[18] Lingtong Kong, Boyuan Jiang, Donghao Luo, Wenqing Chu, Xiaoming Huang, Ying Tai, Chengjie Wang, and Jie Yang. Ifrnet: Intermediate feature refine network for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on*

Computer Vision and Pattern Recognition, pages 1969–1978, 2022. 1, 2, 5

[19] Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. Adacof: Adaptive collaboration of flows for video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5316–5325, 2020. 2

[20] Peidong Liu, Zhaopeng Cui, Viktor Larsson, and Marc Pollefeys. Deep shutter unrolling network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5941–5949, 2020. 1, 3

[21] Yihao Liu, Liangbin Xie, Li Siyao, Wenxiu Sun, Yu Qiao, and Chao Dong. Enhanced quadratic video interpolation. In *European Conference on Computer Vision*, pages 41–56. Springer, 2020. 2

[22] Yu-Lun Liu, Yi-Tung Liao, Yen-Yu Lin, and Yung-Yu Chuang. Deep video frame interpolation using cyclic frame generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8794–8802, 2019. 2

[23] Ziwei Liu, Raymond A Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video frame synthesis using deep voxel flow. In *Proceedings of the IEEE international conference on computer vision*, pages 4463–4471, 2017. 2

[24] Simone Meyer, Oliver Wang, Henning Zimmer, Max Grosse, and Alexander Sorkine-Hornung. Phase-based frame interpolation for video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1410–1418, 2015. 2

[25] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3883–3891, 2017. 1, 3

[26] Simon Niklaus and Feng Liu. Softmax splatting for video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5437–5446, 2020. 2

[27] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive convolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 670–679, 2017. 2

[28] Jihyong Oh and Munchurl Kim. Demfi: deep joint deblurring and multi-frame interpolation with flow-guided attentive correlation and recursive boosting. In *European Conference on Computer Vision*, pages 198–215. Springer, 2022. 1, 3, 4, 5, 7, 8

[29] Junheum Park, Keunsoo Ko, Chul Lee, and Chang-Su Kim. Bmbc: Bilateral motion estimation with bilateral cost volume for video interpolation. In *European Conference on Computer Vision*, pages 109–125. Springer, 2020. 1, 2

[30] Kuldeep Purohit, Anshul Shah, and AN Rajagopalan. Bringing alive blurred moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6830–6839, 2019. 1, 3

[31] Fitsum A Reda, Deqing Sun, Aysegul Dundar, Mohammad Shoeybi, Guilin Liu, Kevin J Shih, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Unsupervised video interpolation using cycle consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 892–900, 2019. 2

[32] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *European Conference on Computer Vision*, pages 184–201. Springer, 2020. 1, 3

[33] Wang Shen, Wenbo Bao, Guangtao Zhai, Li Chen, Xiongkuo Min, and Zhiyong Gao. Blurry video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5114–5123, 2020. 1, 3, 7

[34] Wang Shen, Wenbo Bao, Guangtao Zhai, Li Chen, Xiongkuo Min, and Zhiyong Gao. Video frame interpolation and enhancement via pyramid recurrent framework. *IEEE Transactions on Image Processing*, 30:277–292, 2020. 3

[35] Hyeonjun Sim, Jihyong Oh, and Munchurl Kim. Xvfi: Extreme video frame interpolation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14489–14498, 2021. 6

[36] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1279–1288, 2017. 1, 3

[37] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8174–8182, 2018. 4

[38] Zhixiang Wang, Xiang Ji, Jia-Bin Huang, Shin'ichi Satoh, Xiao Zhou, and Yinqiang Zheng. Neural global shutter: Learn to restore video from a rolling shutter camera with global reset feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17794–17803, 2022. 1

[39] Xiangyu Xu, Li Siyao, Wenxiu Sun, Qian Yin, and Ming-Hsuan Yang. Quadratic video interpolation. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[40] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7

[41] Youjian Zhang, Chaoyue Wang, and Dacheng Tao. Video frame interpolation without temporal priors. *Advances in Neural Information Processing Systems*, 33:13308–13318, 2020. 1, 3, 7

[42] Zhihang Zhong, Mingdeng Cao, Xiao Sun, Zhirong Wu, Zhongyi Zhou, Yinqiang Zheng, Stephen Lin, and Imari Sato. Bringing rolling shutter images alive with dual reversed distortion. *arXiv preprint arXiv:2203.06451*, 2022. 4, 5

[43] Zhihang Zhong, Yinqiang Zheng, and Imari Sato. Towards rolling shutter correction and deblurring in dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9219–9228, 2021. 1, 3