Video frame interpolation (VFI) enables many interesting applications ranging from video compression and framerate up-conversion in TV broadcasting to artistic video effects such as speed ramp in professional cinematography. The performance of VFI methods is largely affected by various factors such as scene lighting conditions, the magnitude and complexity of motion in the scene, the spatial extension of resulting motion blur, the presence of complex occlusions, or thin structures in the scene. Popular VFI methods [Jiang et al.(2018), Bao et al.(2019), Sim et al.(2021)] mostly rely on well-exposed frames in the captured video. Nevertheless, in the case of high dynamic range (HDR) scenes captured using traditional single-exposure sensors, undesired under- and over-exposure effects might appear. The resultant noise and intensity clamping can adversely affect the quality of VFI, as finding the pixel correspondence between the frames becomes more ambiguous. Another major challenge is the large and non-uniform motion in the scene. Although recent methods [Reda et al.(2022), Sim et al.(2021)] have shown progress in handling large motion, they typically heavily rely on the motion linearity assumption that might not hold in practice. Explicit handling of non-linear motion becomes possible by processing more than two subsequent frames [Xu et al.(2019), Park et al.(2021)]; however, temporal sampling might still be too low for reliable motion reconstruction. Motion blur due to low-shutter speed and long-exposure times further leads to spatial and temporal loss of image details. For this reason, handling blurry frames is typically treated as a challenge in the VFI task [Shen et al.(2020), Zhang et al.(2020)], while potentially, motion blur encodes continuous temporal information on the magnitude and direction of motion, particularly for large motion.

Programmable sensors with spatially varying exposures [Sony(1999), Carey et al.(2013)] greatly expand the dynamic range of contrast in captured scenes [Hajsharif et al.(2014), Go et al.(2019), Choi et al.(2017), Heide et al.(2014), Cogalan et al.(2022), Nguyen et al.(2022), Cho et al.(2014)]. In this work, we explore such sensor capabilities toward improving the motion estimation accuracy in VFI. In particular, we consider a dual-exposure sensor that captures short and long exposures for spatially interleaved pixel columns in a single shot [CVM(accessed on Sept. 17, 2021)]. Importantly, while the exposure duration differs, the exposure completion is temporally aligned, which enables recovering two
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors

Figure 1: Video frame interpolation (VFI) in the framerate extension task using the state-of-the-art FILM [Reda et al.(2022)] and our methods. The first column shows our in-between frame reconstruction, while the insets focus on the dark scene details that require long exposure durations and feature significant motion blur due to camera motion (second column). Such blur adversely affects VFI for the FILM method (third column). Our method employs temporally continuous information on the scene motion that is encoded in motion blur to improve the VFI quality (fourth column).

temporal samples of the scene motion that are perfectly spatially registered at the sensor. We show that such an increased temporal sampling rate substantially improves the accuracy of complex motion interpolation, as motion non-linearity can readily be reconstructed for two subsequent frames. Furthermore, the short exposure typically leads to a sharp image, while the long exposure results in substantial motion blur that provides additional insights into the motion direction and magnitude (Fig. 1). This is of particular importance in dark scene regions, where the short exposure might be strongly underexposed and noisy, and the long exposure becomes the only reliable measurement of scene motion. As in other works, we employ a multi-exposure technique to reconstruct high dynamic range video frames, but for the first time, we simultaneously perform VFI that can handle complex, non-linear motion in the scene. We train an end-to-end convolutional network to achieve those goals. We also propose a metric of motion non-linearity that allows us to analyze the existing high-speed videos.

The key contributions of our work are:

• We propose a machine learning solution for VFI that can handle high-dynamic-range content and complex non-uniform motion, enabled by deriving two temporal samples of the scene motion for each frame by joint processing of short and long exposures as captured using a dual-exposure sensor.
• We exploit motion blur information inherent in the long exposure to further improve motion estimation quality.
• We develop a metric of motion complexity that provides interesting insights into existing datasets used in the training of VFI methods.

In the following section, we discuss previous work, and in Sec. 3, we present our VFI method for HDR sequences. In Sec. 4 we introduce our metric of scene motion uniformity that enables a meaningful comparison of existing VFI methods, while Sec. 5 provides implementation details of our network. Sec. 6 confronts our technique with existing works in a performance comparison and reports an outcome of ablation studies. Finally, we conclude this work in Sec. 7.

2 Previous work

In this section, we discuss existing VFI methods dealing with either sharp (Sec. 2.1) or blurry (Sec. 2.2) input video. We focus on the problems of handling non-uniform motion and high-dynamic-range content (Sec. 2.3) that are central to this work. We refer the reader to a recent survey where a more complete treatment of VFI is presented [Parihar et al.(2021)].

2.1 Sharp Video Frame Interpolation

A vast majority of existing VFI techniques assume that the motion in the input video is uniform, but there are also methods explicitly designed without this assumption.
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors

Uniform motion  SepConv [Niklaus et al.(2017)] merges flow estimation and frame warping into a single convolution step. They predict spatially-varying 1D kernels and convolve with them input frames to interpolate new frames. SuperSlowMo [Jiang et al.(2018)] uses bi-directional flows and an occlusion map to synthesize intermediate frames at any arbitrary time. DAIN [Bao et al.(2019)] utilizes additional interpolation kernels and depth maps for blending the input frames. A cycle consistency loss is introduced to learn frame interpolation with fewer training pairs [Liu et al.(2019)] or without any supervision [Reda et al.(2019)]. BMBC [Park et al.(2020)] warps the input frames with a proposed bilateral motion model and combines them using learned dynamic blending filters. CAIN [Choi et al.(2020)] uses a channel attention module to interpolate video frames without the need for estimation of motion. SoftSplat [Niklaus and Liu(2020)] proposes differentiable forward warping via softmax splatting and shows its benefits for VFI. AdaCoF [Lee et al.(2020)] proposes a warping module in which a target pixel can refer to not only one but many pixels at any location in the reference. XVFI [Sim et al.(2021)] presents a high-speed (1000fps) video dataset and proposes a multi-scale recursive approach to handle large motion in the scene. Recently, FILM [Reda et al.(2022)] has introduced a unified framework that achieves superior results for large and complex motions by balancing the motion range distribution in the training dataset. For all methods discussed here, a combination of large and strongly non-uniform motion might lead to highly objectionable artifacts.

Non-uniform motion  QVI [Xu et al.(2019)] is one of the first video interpolation methods to model curvilinear motion with the quadratic equation using four temporal frames. Chi et al. [Chi et al.(2020)] extend QVI by introducing an additional cubic term that accounts for the change in the acceleration. ABME [Park et al.(2021)] handles the non-uniform motion in the scene by extending the BMBC [Park et al.(2020)] for asymmetric bilateral motion between input frames. In all those methods, more than two consecutive frames are required to capture the motion non-uniformity that for large and complex motions might be challenging both because of temporal sampling deficits as well as overall reduced flow estimation accuracy. In our approach, we capture two exposures in a single frame that increase the sampling rate twice, and we employ motion blur inherent for the longer exposure as an additional cue to the flow estimation. Moreover, by explicitly accounting for motion non-uniformity statistics in our training data, we can improve our network’s ability to handle a wide variation of such motions.

2.2 Joint Video Deblurring and Interpolation

It is inevitable to have motion blur during capturing caused by low shutter speed, long exposure time, or rapid movement in the scene. Recent works demonstrate that a joint deblurring and frame interpolation greatly improves the resulting VFI quality over an independent treatment of these tasks. TNTT [Jin et al.(2019)] adopts a joint optimization scheme to create by averaging neighbor frames of existing high framerate videos with mutually overlapping temporal windows, and without accounting for the camera readout time. Those two factors can significantly hinder the reconstruction of physically correct blur. UTI-VFI [Zhang et al.(2020)] periodically skips selected frames in the input high framerate video to emulate the lost information during the sensor readout time, and it can also generate the intermediate frames at an arbitrary time. It first extracts two sharp keyframes corresponding to the start and the end of a given blurry frame sequence, and then it utilizes an off-the-shelf optical flow network to interpolate in-between frames. Moreover, it adapts quadratic motion formulation in order to properly handle non-uniform motion. Rengarajan et al. [Rengarajan et al.(2020)] take a triplet of short-long-short exposures captured by a programmable machine vision camera and recover a sharp high framerate sequence. In their capturing setup, the short exposures are sharp but noisy and can act as a pivot for deblurring the long exposure. While these methods attempt to remove the motion blur in the video frames, the inherent motion blur can potentially reveal information about the magnitude and direction of the motion, especially in the case of large non-uniform motion. In this work, we explicitly extract the motion flow from the blurry input frames. Moreover, large motions could make the joint deblurring and VFI tasks very challenging. In our approach, we take advantage of the less blurry short exposure in each frame to make the flow estimation more reliable.

2.3 High dynamic range video

HDR video reconstruction is performed using multi-exposure techniques, where subsequent frames with temporally interleaved different exposures are combined, and their dynamic content is aligned typically using optical flow methods [Kalantari et al.(2013), Kalantari et al.(2017), Kalantari and Ramamoorthi(2019), Yan et al.(2020), Chen et al.(2020)].
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors

Figure 2: Our HDR VFI pipeline takes as the input two subsequent frames \( M_0 \) and \( M_1 \) as captured using our dual-exposure sensor. The input frames are independently processed by a learned \texttt{MakeHDR} network, so that the corresponding sharp HDR images \( \hat{I}_{e0} \) and \( \hat{I}_{e1} \) are aligned with the short exposures, as well as blurry long exposure images \( L_0 \) and \( L_1 \), whose ends are aligned with \( \hat{I}_{s0} \) and \( \hat{I}_{s1} \) obtained. Next, the motion flows \( F_{e0 \rightarrow s0} \) and \( F_{s1 \rightarrow e1} \) are extracted from the blur inherent to the long exposures \( L_0 \) and \( L_1 \) using a learned \texttt{Blur2Flow} network. Also, the motion flows \( F_{e0 \rightarrow s1} \) and \( F_{s1 \rightarrow e0} \) estimated between the sharp HDR frames \( \hat{I}_{e0} \) and \( \hat{I}_{e1} \) in both directions using the state-of-the-art flow estimation method Raft [Teed and Deng(2020)]. Then, the motion flow pairs \( F_{e0 \rightarrow s0} \) and \( F_{e1 \rightarrow s1} \), as well as \( F_{s1 \rightarrow e0} \) and \( F_{s1 \rightarrow e1} \) (reverted from \( F_{e1 \rightarrow s1} \)) are independently fed to a non-learnable \texttt{FitQuad} module to calculate the forward flows \( F_{e0 \rightarrow t} \) and \( F_{s1 \rightarrow t} \) that are parametrized using a quadratic motion model for a position \( t \). Then, we compute the backward flows \( F_{e0 \rightarrow s0} \) and \( F_{s1 \rightarrow e1} \) using a differentiable flow \texttt{Reverse} module [Xu et al.(2019)] in order to warp the frames \( \hat{I}_{e0} \) and \( \hat{I}_{s1} \) to the position \( t \). Finally, the warped images are combined using a trained \texttt{Synthesis} network to produce the output frame \( \hat{I}_t \).

To alleviate the need for such alignment, single-shot HDR techniques are developed that rely on specialized dual-ISO sensors [Hajsharif et al.(2014), Go et al.(2019), Choi et al.(2017), Cogalan and Akyuz(2020)]. Dual-exposure sensors [CVM(accessed on Sept. 17, 2021)] offer two motion samples per frame, which, as we show in this work, is greatly beneficial for VFI, albeit requires long-exposure deblurring in the HDR video reconstruction [Heide et al.(2014), Cogalan et al.(2022), Nguyen et al.(2022), Cho et al.(2014)]. The scope of these methods is mainly limited to HDR video reconstruction, and they do not aim for the VFI task. An exception here is the work of Rebecq et al. [Rebecq et al.(2019)], where high framerate HDR video is reconstructed using a highly specialized event camera that, in a frameless manner, asynchronously responds to per-pixel brightness changes.

3 Method

In this section we propose a method for video frame interpolation that reconstructs high-dynamic range frames in the continuous time domain. Fig. 2 summarizes our processing pipeline and the following paragraphs provide more detailed description of its key components. Our method takes as an input two subsequent video frames \( M_0 \) and \( M_1 \) that are captured using our dual-exposure sensor and produces a sharp HDR frame \( \hat{I}_t \) for any position \( t \) between \( M_0 \) and \( M_1 \). Each captured frame \( M_i \), where with a suffix \( i \) we denote any input frame, contains a pair of spatially interleaved short and long exposures, and it is processed by the \texttt{MakeHDR} network to produce a sharp HDR frame \( \hat{I}_{ei} \) that is aligned with the short exposure, and a blurry long exposure frame \( L_i \), whose end is aligned with \( \hat{I}_{si} \). Both frames are fed to the \texttt{Blur2Flow} network to predict the flow \( F_{ei \rightarrow si} \) that extracts the motion from the blur within the \( L_i \) frame. Next, we compute the flows \( F_{e0 \rightarrow s1} \) and \( F_{s1 \rightarrow e0} \) between the sharp HDR frames \( \hat{I}_{e0} \) and \( \hat{I}_{s1} \) in both directions using an off-the-shelf flow estimation method such as Raft [Teed and Deng(2020)]. Note that the frame \( \hat{I}_{s1} \)
is reconstructed by warping the $\hat{I}_{s1}$ using the backward flow $F_{s1 \rightarrow s1}$, where a Reverse module for differentiable flow reversal [Xu et al.(2019)] is applied to the forward flow $F_{s1 \rightarrow s1}$, as originally derived by Blur2Flow network. Given two estimated flows per each frame ($F_{e0 \rightarrow e0}$, $F_{e0 \rightarrow s1}$ for $M_0$, and $F_{e1 \rightarrow s1}$, $F_{e1 \rightarrow e0}$ for $M_1$), we fit a quadratic motion model [Xu et al.(2019)] using a non-learnable FitQuad module, and compute the forward flows $F_{e0 \rightarrow t}$ and $F_{s1 \rightarrow t}$. Then, we compute the backward flows $F_{t \rightarrow e0}$ and $F_{t \rightarrow s1}$ using the Reverse module. Finally, the frames $\hat{I}_{e0}$ and $\hat{I}_{s1}$ are warped with the backward flows and combined using the Synthesis module to interpolate the frame $\hat{I}_t$.

**HDR reconstruction: MakeHDR** We acquire our input video using a dual-exposure sensor [CVM(accessed on Sept. 17, 2021)] that simultaneously captures a short and long exposure for each frame. In our setup, the exposure time for the long exposure is four times higher than the short exposure. Each exposure is stored at odd and even columns in the sensor. As a result, both exposures are provided as half-resolution images and they need to be up-sampled in the horizontal direction. Moreover, the short exposure exhibits strong noise in dark scene regions and requires denoising. On the other hand, the long exposure is less noisy while it might contain considerable motion blur and requires deblurring. To do so, we employ the network design and the training strategy introduced in [Cogalan et al.(2022)] to jointly deblur, denoise, and upsample our input frames $M_i$ to produce sharp, clean, and full-resolution short and long exposures. Both exposures are then combined using a non-learnable technique, similar to [Debevec and Malik(2008)], to produce a sharp HDR frame $\hat{I}_{s1}$. We also extend the network output to produce an additional full-resolution blurry long exposure $\hat{I}_{e1}$.

**Motion from blur: Blur2Flow** Existing motion blur in the long exposure can potentially reveal information about the motion in the scene [Shen et al.(2020), Rengarajan et al.(2020)]. Using our Blur2Flow network, we aim to recover the motion flow $F_{e1 \rightarrow s1}$ that is responsible for the blur pattern registered in the long exposure $\hat{I}_1$. The sensor design imposes that the short and long exposures are completed precisely at the same time point, and in our HDR reconstruction, the sharp frame $\hat{I}_{s1}$ is aligned with the short exposure. Since we use $\hat{I}_{s1}$ as a reference for the predicted flow $F_{e1 \rightarrow s1}$ alignment and given that its time span is equivalent to the long exposure duration, an additional per-pixel temporal motion sample at the beginning of long exposure is readily available by warping $\hat{I}_{e1}$ to $\hat{I}_{s1}$.

**Flow reversal: Reverse** As the forward warping of the HDR frame $\hat{I}_{e1}$ to $\hat{I}_{s1}$ using the forward flow $F_{e1 \rightarrow s1}$ is not differentiable, we compute the backward flow $F_{s1 \rightarrow e1}$ using a flow reversal module introduced in QVI [Xu et al.(2019)].

**Quadratic motion model: FitQuad** In this step, we first Reverse the forward flow $F_{e1 \rightarrow s1}$, as derived by Blur2Flow, to produce the backward flow $F_{s1 \rightarrow e1}$ that we use to warp $\hat{I}_{s1}$ into $\hat{I}_{e1}$. Then, we compute the bidirectional flows $F_{e0 \rightarrow s1}$ and $F_{e1 \rightarrow e0}$ between the HDR frames $\hat{I}_{e0}$ and $\hat{I}_{s1}$ using a state-of-the-art flow estimation method as proposed in [Teed and Deng(2020)]. We found that in particular, for complex motion, using $\hat{I}_{s1}$, instead of $\hat{I}_{e1}$, eases the bidirectional flow computation and improves its quality, as the time span with respect to $\hat{I}_{e0}$ is reduced. Now, to warp $\hat{I}_{e0}$ to a given time point $t$, we employ the flows $F_{e0 \rightarrow e0}$ and $F_{e0 \rightarrow s1}$ to derive a quadratic motion model as:

$$F_{e0 \rightarrow t} = \frac{1}{2} a_0 \times t^2 + v_0 \times t$$

(1)

where the acceleration $a_0 = \frac{2(F_{e0 \rightarrow s1} + \lambda F_{e0 \rightarrow e0})}{\lambda^2 + \lambda}$ and the velocity $v_0 = \frac{F_{e0 \rightarrow s1} - \lambda^2 F_{e0 \rightarrow e0}}{\lambda^2 + \lambda}$ directly depend on the ratio $\lambda$ between the camera readout time and its complete duty cycle that is required for capturing each frame. Similarly, to warp $\hat{I}_{s1}$ we compute the flow $F_{s1 \rightarrow t}$:

$$F_{s1 \rightarrow t} = \frac{1}{2} a_1 \times t^2 + v_1 \times t$$

(2)

where $a_1 = \frac{2(F_{s1 \rightarrow e0} + \lambda F_{s1 \rightarrow s1})}{\lambda^2 + \lambda}$ and $v_1 = \frac{\lambda^2 F_{s1 \rightarrow e0} - F_{s1 \rightarrow s1}}{\lambda^2 + \lambda}$.

**Intermediate frame synthesis: Synthesis** As the forward warping of HDR frames $\hat{I}_{e0}$ and $\hat{I}_{s1}$ to a novel position $t$ using the flows $F_{e0 \rightarrow e0}$ and $F_{s1 \rightarrow e1}$ is not differentiable, we use the Reverse module to compute the backward flows $F_{e0 \rightarrow e0}$ and $F_{s1 \rightarrow e1}$. Then, we employ a differentiable warping operator warp [Jaderberg et al.(2015)] as:

$$\hat{I}_{e0 \rightarrow t} = \text{warp}(\hat{I}_{e0}, F_{e0 \rightarrow e0}) \text{ and } \hat{I}_{s1 \rightarrow t} = \text{warp}(\hat{I}_{s1}, F_{s1 \rightarrow e1}).$$

(3)

Afterwards, the warped frames along with the backward flows are fed to the Synthesis network:

$$\alpha = \text{Synthesis}(\hat{I}_{e0 \rightarrow t}, \hat{I}_{s1 \rightarrow t}, F_{e0 \rightarrow e0}, F_{s1 \rightarrow e1})$$

(4)

which predicts the soft occlusion weight $\alpha$ that controls the contribution of input warped images $\hat{I}_{e0 \rightarrow t}$ and $\hat{I}_{s1 \rightarrow t}$ for each pixel. Finally, we synthesize the interpolated frame $\hat{I}_t$ as:

$$\hat{I}_t = \alpha \circ \hat{I}_{e0 \rightarrow t} + (1 - \alpha) \circ \hat{I}_{s1 \rightarrow t}$$

(5)
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors


<table>
<thead>
<tr>
<th></th>
<th>Adobe 240</th>
<th>GoPro</th>
<th>X4K1000FPS</th>
<th>SlowFlow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform</td>
<td>26 (20%)</td>
<td>2 (6%)</td>
<td>108 (98%)</td>
<td>14 (35%)</td>
</tr>
<tr>
<td>Non-uniform</td>
<td>107 (80%)</td>
<td>31 (94%)</td>
<td>2 (2%)</td>
<td>25 (65%)</td>
</tr>
</tbody>
</table>

where ⊙ stands for per pixel multiplication.

**Loss function** Our loss function is composed of three components that are targeted to train MakeHDR, Blur2Flow, and Synthesis networks. First, the output of the MakeHDR network is supervised with the ground truth $I_{e1}$ and $L_i$ (refer to Sec. 6.1 on details how we acquire the ground truth frames from high-framerate video datasets) using the reconstruction loss:

$$L_{hdr} = \sum_{i=0,1} \| I_{e1} - \hat{I}_{e1} \|_1 + \| L_i - \hat{L}_i \|_1$$  \hspace{1cm} (6)

As the ground truth flow is not available, we supervise the Blur2Flow network using the re-projection loss:

$$L_{flow} = \sum_{i=0,1} \| I_{e1} - \text{warp}(I_{s1}, F_{e1 \rightarrow s1}) \|_1$$  \hspace{1cm} (7)

where the ground truth frame $I_{s1}$ is warped using the flow $F_{e1 \rightarrow s1}$ to align it with the ground truth frame $I_{e1}$. Finally, we supervise the Synthesis network using the reconstruction loss:

$$L_{synth} = \| I_t - \hat{I}_t \|_1$$  \hspace{1cm} (8)

where the interpolated frame $\hat{I}_t$ and the corresponding ground truth $I_t$ are compared. The final loss $L_{total}$ is then computed as:

$$L_{total} = L_{hdr} + L_{flow} + L_{synth}$$  \hspace{1cm} (9)

4 Motion non-uniformity analysis

In order to properly validate our proposed method, we need to assure that our dataset contains diverse examples of scene motion non-uniformity. To this end, we perform an analysis of motion non-uniformity in some of popular
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors

Figure 4: The histogram of measured non-uniform motions for different datasets, where the horizontal axis shows the actual motion error ($\times 10^{-2}$) with respect to the linear motion fit, and the vertical axis denotes the probability of the observed frames given an error value.

high-framerate video datasets including Adobe240 [Su et al.(2017)], GoPro [Nah et al.(2017)], X4K1000FPS [Sim et al.(2021)], and SlowFlow [Janai et al.(2017)]. Our procedure is as follows: For each pixel in a given frame we use Raft [Teed and Deng(2020)] to track the corresponding pixels for $N = 15$ consecutive frames. Note that in some cases such tracking might fail due to occlusions and textureless regions. We find the occlusion regions by applying a forward-backward flow consistency check [Jonschkowski et al.(2020)] between the first and last frames, and exclude them in our measurements. Likewise, as the estimated flow in the textureless regions is usually erroneous, we clip the flow to zero if its value is less than one pixel. In the next step, we find a linear model that in the least square sense fits to the motion trajectory for each pixel. We then consider the mean square error with respect to such a linear fit, where higher errors indicate more motion non-uniformity. Fig. 3 shows the trajectories of pixels for a sample scene containing regions with non-uniform motion in each dataset. Note that for each pixel, the error value is normalized by the aggregated pixel displacement across the consecutive frames. Since the error is calculated for individual pixels, we measure the amount of motion non-uniformity in a frame by taking the 90th percentile of the calculated error over all pixels. We then repeat this procedure for non-overlapping sets of $N$ consecutive frames in each scene in each dataset. Fig. 4 shows the histogram of measured non-uniform motions for each dataset, where the horizontal axis denotes the error of the linear fit ($\times 10^{-2}$), and the vertical axis is the probability of observing scene for a given error value. In our experiments, we identify a scene as containing the non-uniform motion, when the average error is over $6.0 \times 10^{-2}$ (the threshold value is chosen empirically based on our visual inspection of the pixel trajectories as illustrated in Fig. 3). Based on this threshold, we provide statistics of scenes with uniform and non-uniform motions for each dataset in Tbl. 1. The Adobe240 and GoPro datasets feature significant percentages of non-uniform motion as they are captured with a handheld camera. Although large motions are present in the X4K1000FPS dataset, the camera moves along mostly linear trajectories.

5 Implementation

Our MakeHDR network architecture follows [Cogalan et al.(2022)]. The network output is given in the Bayer domain, and we apply demosaicing using OpenCV [Bradski(2000)], followed by a gamma correction to create the final short and long exposures in the sRGB format. The Blur2Flow network employs a network architecture similar to the PWCNet [Sun et al.(2017)]. Our Synthesis network is implemented as a 12-layer conventional neural network with dilated convolutions and skip connections. During training we use the patch size of $768 \times 768$, nevertheless, at the inference time our convolutional network as well as all non-learnable component scale with resolution.

6 Results

In this section, we first introduce the training and evaluation datasets. Then we show quantitative and qualitative comparisons of our method with existing VFI methods. Finally, we provide ablation to justify our training set and different components of our method.
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors

Figure 5: Visual comparisons of our method with the state-of-the-art VFI methods using the synthetic dataset described in Sec. 6.1. For each of three scenes, the first row of insets shows the performance of respective VFI methods, while the second row presents the corresponding per-pixel error maps between the interpolated results and the ground truth. The PSNR/SSIM values written below each error map are computed for each inset rather than the entire image. In the upper scene taken from the X4K1000FPS test set, the wheel moves in a non-linear trajectory and the existing VFI methods struggle to position the wheel correctly for the interpolated frames, while our method leads to a good alignment with the ground truth. In the middle scene taken from the GoPro dataset the camera is moving with an extremely non-uniform motion as shown in Fig. 3. While the existing VFI methods produce visually plausible results, they are not correctly aligned with the ground truth as the error map reveals. The bottom scene, taken again from the X4K1000FPS test set, contains a combination of camera and object movements. In this case, the existing VFI methods fail to properly handle the occlusion boundaries.

6.1 Dataset
As it is impossible to capture ground truth high-framerate HDR videos using our dual exposure sensor, and third-party high-framerate HDR videos are unavailable, we synthesize our training and evaluation datasets using existing LDR high-framerate videos. In our experiments, we take the scenes from X4K1000FPS [Sim et al.(2021)] and SlowFlow [Jai et al.(2017)] as our training dataset, and we consider Adobe240 [Su et al.(2017)] and GoPro [Nah et al.(2017)] as our evaluation dataset. Our training and testing video sequences are defined as follows: We take 16 consecutive frames in a high framerate video, where the 1st and 4th frames are our sharp beginning and ending frames ($I_{s0}$ and $I_{e0}$). We sum up the four neighboring frames starting from 1 to 4 to simulate the long exposure $I_0$. We then skip 9 frames to simulate the camera readout gap. Similarly, we take the 13th and 16th frames as the $I_{s1}$ and $I_{e1}$ and sum the frames from 13 to 16 to create the long exposure $I_1$. We consider frame 7 and 10 as the target frames for the reconstructions. Note that in our simulation of long exposures, we clip the aggregated pixel intensity if it exceeds the value of 255. In our simulation, we ignore each patch if more than 20% of its content is already saturated in the original high framerate video. For our test set, we are interested in evaluating our method for a different dataset containing the scenes with solely uniform and non-uniform motion; hence we split all the scenes in the Adobe240 [Su et al.(2017)] and GoPro [Nah et al.(2017)] datasets into uniform and non-uniform sets based on our metric discussed in Sec. 4.

6.2 Quantitative comparison
We compare our proposed method with state-of-the-art sharp VFI methods (refer to Sec. 2.1): FILM [Reda et al.(2022)] and XVFI [Sim et al.(2021)] that rely on a uniform motion assumption, and QVI [Xu et al.(2019)] and ABME [Park...
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors

Table 2: Quantitative comparison of our method with state-of-the-art video frame interpolation methods. The ABME and QVI methods are designed to handle non-uniform motions, while the XVFI and FILM methods rely on a linear motion assumption but can handle large motions.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Uniform PSNR</th>
<th>Uniform SSIM</th>
<th>Non-uniform PSNR</th>
<th>Non-uniform SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABME [Park et al.(2021)]</td>
<td>33.16</td>
<td>0.88</td>
<td>31.04</td>
<td>0.83</td>
</tr>
<tr>
<td>QVI [Xu et al.(2019)]</td>
<td>32.10</td>
<td>0.89</td>
<td>31.32</td>
<td>0.85</td>
</tr>
<tr>
<td>XVFI [Sim et al.(2021)]</td>
<td>32.56</td>
<td>0.87</td>
<td>30.88</td>
<td>0.82</td>
</tr>
<tr>
<td>FILM [Reda et al.(2022)]</td>
<td>32.89</td>
<td>0.87</td>
<td>30.90</td>
<td>0.82</td>
</tr>
<tr>
<td>Ours</td>
<td>35.97</td>
<td>0.94</td>
<td>35.57</td>
<td>0.93</td>
</tr>
</tbody>
</table>

Figure 6: The visual comparisons of our interpolation results for three scenes captured using our camera with a dual-exposure sensor. Our method is able to correctly interpolate the frames in the scenes with the challenging cases of a rolling round box (the upper scene), a rotary camera motion (the middle scene) and a moving object behind a refractive object (the bottom scene).

et al.(2021)] that explicitly support non-uniform motion. FILM, XVFI, and AMBE require just two neighboring frames as an input, while the method QVI needs to process four consecutive frames. Since these VFI methods expect post-processed sRGB images as an input, and moreover, we are interested in evaluating our interpolation part, we feed the sharp HDR output to the sharp frame interpolation methods. Note that we are unable to compare with the blurry VFI methods (refer to Sec. 2.2), as they require well-exposed blurry input frames (effectively, blurry HDR frames) while our long exposure typically contains a considerable amount of saturation that is poorly handled by these methods. Table 1 summarizes our comparisons with the VFI methods for the aggregated Adobe240 and GoPro datasets that we split into two categories with respect to the motion uniformity (Sec. 6.1). For the dataset with uniform motion, the competing VFI methods perform similarly, while for the non-uniform motion dataset, the methods with a non-uniform motion assumption (AMBE and QVI) perform, as expected, better compared to ones with a uniform motion assumption (FILM and XVFI). In any case, our method outperforms the existing VFI methods by a large margin.
6.3 Qualitative comparison

We first provide visual comparisons with the state-of-the-art VFI methods for three scenes synthesized data with ground truth in Fig. 5. Moreover, in Fig. 1 and Fig. 6 we visualize the results for four captured scenes using our Axiom-beta camera with a CMOSIS CMV12000 sensor. In both setups, we use the exposure ratio of 4 between the short and long exposures. Since the captured frames using our camera cannot be fed directly to the other VFI methods, we provide them with the reconstructed sharp HDR images $\hat{I}_{e0}$ and $\hat{I}_{e1}$ using our MakeHDR network. In Fig. 1, we have an example a scene with high dynamic range content where there is a moving camera with acceleration during capturing; the magnitude of the motion blur in the long exposure can show the amount of camera motion. In this case, we can see a method such as FILM drastically fails, while our method employs the encoded blur information to improve the interpolation quality. The upper scene in the Fig. 6 shows an example of a rolling box in which the existing VFI methods even the ones designed to deal with non-uniform motion such as AMBE and QVI fail to properly interpolate an intermediate frame due to non-uniform motion caused by rotatory motion of the round box. In the next examples, we captured a crystal ball while the camera is rapidly rotating (the middle scene) or an object is moving behind the crystal ball (the bottom scene). We can observe that in these challenging examples where even a uniform motion in the scene might appear non-uniform in the refracted image, other methods struggle to correctly reconstruct an in-between frame. In all cases, we can see our method faithfully reconstruct the in-between frames even in difficult conditions where there are reflections on the crystal ball (the middle and bottom scenes). Please refer to our supplementary video for the temporal consistency of our method.

![Image](image_url)

Figure 7: Our interpolation failure example in case the moving content is largely saturated in the long exposure.

Table 3: The effect of including the scenes with non-uniform motion in our training dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Uniform</th>
<th>Non-uniform</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PSNR</td>
<td>SSIM</td>
</tr>
<tr>
<td>Ours-uniform</td>
<td>34.86</td>
<td>0.93</td>
</tr>
<tr>
<td>Ours-non-uniform</td>
<td>34.42</td>
<td>0.93</td>
</tr>
</tbody>
</table>

6.4 Ablation study

We perform an experiment to show the effect of including the scenes with non-uniform motion in our training dataset. We summarize the results in table Tbl. 3. We first train our method using only the scenes with uniform motion (first row in the Tbl. 3) and train using scenes with non-uniform motion (in the second row). Our results show no significant difference in the performance of our method when trained using the scenes with non-uniform motion. This is due to the fact that the quadratic motion fitting part in our method does not have any learnable parameters that could benefit from non-uniform motion samples in the training set.

Moreover, we analyze the contribution of the Blur2Flow network where we attempt to reconstruct the intermediate frames using only the backward and forward flows between the sharp HDR frames $\hat{I}_{e0}$ and $\hat{I}_{e1}$ using Raft [Teed and Deng(2020)]. In this case, we linearly split each flow to reach to any position $t$ between the frames. The results are
Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors

shown in the Tbl. 4. The higher number for our full method shows the effectiveness of including the Blur2Flow network in our pipeline.

Table 4: The effect of Blur2Flow network in our pipeline.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Uniform</th>
<th>Non-uniform</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PSNR</td>
<td>SSIM</td>
</tr>
<tr>
<td>Ours-w/o blur2flow</td>
<td>31.51</td>
<td>0.86</td>
</tr>
<tr>
<td>Ours-full</td>
<td>35.97</td>
<td>0.94</td>
</tr>
</tbody>
</table>

6.5 Limitation

Saturation is inevitable in the long exposure for bright scene regions. If the moving content is present in such regions, our flow prediction using the Blur2Flow network becomes less accurate. Fig. 7 shows an example of this case where we synthetically increase the saturation in the long exposure for the wheel example shown in Fig. 5 and our method fails to correctly reconstruct the intermediate frame.

The dynamic range that we can reconstruct is limited by the exposure ratio of four that we assume in this work. For larger ratios, the accuracy of HDR frame reconstruction by the MakeHDR network might be reduced [Cogalan et al.(2022)], which could adversely affect the accuracy of HDR video interpolation.

7 Conclusion

In this work, we presented a method for high-dynamic-range video frame interpolation using the dual-exposure sensors. Our method outperforms the existing VFI methods both in terms of quantitative metrics as well as visual results for the challenging scenes containing non-uniform motions. In particular, we achieve high precision alignment of scene motion with the ground truth, where other methods clearly fail, although they may produce visually plausible results. Our method can handle complex motion with consistently high performance as it depends little on explicitly training this reconstruction aspect. Instead, we capitalize on the increased temporal sampling rate due to motion reconstruction from blur information. Also, our method is less dependent on scene lighting conditions, whereas other methods designed for single exposure sensors may suffer from image saturation in bright regions or excessive noise in dark conditions.

References


Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors


[Hajsharif et al.(2014)] Saggi Hajsharif, Joel Kronander, and Jonas Unger. Hdr reconstruction for alternating gain (iso) sensor readout. In Eurographics, Strasbourg, France, April 7-11, 2014. 1, 4


Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors


[Sony(1999)] Sony. Quad bayer sensors, 1999. 1


13