A Unified Framework for High-Frame-Rate High-Dynamic-Range Video Synthesis

Thi Hue Nguyen

A THESIS SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Graduate Program in Electrical Engineering and Computer Science
York University
Toronto, Ontario

July 2025

Abstract

Creating high-dynamic-range (HDR) video at high-frame rates is a technically demanding and application-critical problem, particularly in domains such as cinematography and autonomous perception. The challenge arises from the limitations of conventional image sensors in capturing both temporal and radiometric fidelity. This work introduces a unified framework that jointly addresses HDR reconstruction and temporal interpolation from sequences captured with alternating exposures. In contrast to prior methods that focus only on middle-frame interpolation or rely on computationally intensive pipelines, our approach employs a lightweight, end-to-end network capable of generating HDR frames at arbitrary timesteps in real time on mid-range GPUs. To mitigate the need for ground-truth HDR video, we propose a novel self-supervised training paradigm that leverages reconstruction objectives designed to preserve both photometric accuracy and temporal coherence. Experimental results demonstrate that our framework not only maintains competitive visual fidelity but also significantly reduces computational overhead compared to state-of-the-art baselines.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor, Professor Michael S. Brown, for his invaluable guidance, patience, and support throughout this journey. His encouragement, insightful feedback, and unwavering belief in my work have been instrumental in helping me grow as a researcher and as a person. I feel incredibly fortunate to have had the opportunity to learn from him.

I would also like to sincerely thank my thesis committee, Professor Konstantinos G. Derpanis, and Professor Richard Murray, for taking your valuable time to review my work and for offering thoughtful feedback that helped strengthen this thesis.

To my labmates, Saikiran Kumar Tedla and Trevor D. Canham, thank you for the insightful discussions, your guidance, and your genuine support throughout this journey. I am especially grateful to Trevor for generously sharing his dataset; without it, this thesis would not have been possible. Your kindness and collaborative spirit made this experience much less lonely and far more rewarding.

To my friends and family, even though we are separated by thousands of kilometers, your presence has never felt far. Your constant encouragement, love, and belief in me carried me through the hardest days. Whether through a quick call, a message, or just knowing you were there, you reminded me I was never truly alone.

Contents

List of Tables

List of Figures

List of Acronyms

DNN deep neural network.

EV exposure value.

HDR high dynamic range.

HFR high frame rate.

SDR standard dynamic range.

TMO tone mapping operator.

VFI video frame interpolation.

Chapter 1

Introduction

1.1 Motivation and Problem

High-dynamic-range (HDR) video has rapidly emerged as a new standard in visual media, delivering immersive realism through vivid colors, detailed shadows, and striking highlights. Its adoption spans a wide range of platforms and devices, including high-end televisions, smartphones, and mainstream streaming services. At the same time, there is a growing demand for high-frame-rate (HFR) video, which enables smoother motion rendering, high-quality slow-motion effects, and immersive experiences in augmented and virtual reality (AR/VR). Capturing HFR-HDR video has then become essential for various applications, including cinematic production, interactive video editing, and autonomous navigation in complex lighting environments.

Despite the advantages of HFR and HDR videos, combining the two in a single acquisition pipeline remains fundamentally challenging due to the limitations of current sensor technology. Standard image sensors typically have a limited dynamic range, usually between 8 to 14 f-stops, which is inadequate for capturing natural scenes that can exceed 17 f-stops [1]. When sensor capacity is exceeded, overexposed regions clip to white, while underexposed areas are overwhelmed by noise, resulting in irreversible information loss. Furthermore, in low-light conditions, longer exposure times are necessary to preserve shadow details. However, this reduces the number of frames that can be recorded per second, creating a fundamental trade-off between capturing a high dynamic range and

maintaining a high frame rate.

Two main approaches have been explored to overcome this problem. The first relies on custom-built HDR sensors [2, 3], which can capture high dynamic range in a single shot. However, these sensors are expensive and limited to specialized use cases in professional cinematography or automotive applications. The second, more accessible approach is exposure bracketing, which captures multiple standard dynamic range (SDR) frames at varying exposures and fuses them into a single HDR frame. This can be implemented on consumer devices using alternating exposure sequences (*e.g.*, short–long–short–long), but it introduces temporal misalignment and limits the achievable frame rate due to exposure switching latency.

To generate smooth, high-frame-rate HDR video from these sequences, post-processing techniques such as temporal interpolation are required. Recent hybrid methods [4, 5, 6] have augmented exposure-bracketed video with event cameras, which offer fine-grained temporal information by capturing per-pixel brightness changes at microsecond latency. While effective, such systems demand complex calibration and specialized hardware, making them impractical for widespread use.

In this thesis, we present an alternative, software-only solution for generating high-frame-rate HDR video from exposure-alternating sequences, without the need for additional sensors or hardware. While earlier works [7, 8, 9, 10, 11, 12] have addressed HDR video reconstruction from exposure stacks, they are limited to producing output at the original input frame rate. More recent efforts, such as Khan *et al.* [13], attempt high-frame-rate synthesis using pre-trained interpolation networks. However, their method overlooks the complementary nature of alternating exposures and employs a multi-stage pipeline that is computationally expensive and prone to artifacts.

To overcome these limitations, we propose a unified neural framework for joint HDR reconstruction and video frame interpolation. Our approach builds on HDRFlow [12], enhancing it with a lightweight interpolation module capable of synthesizing intermediate frames at arbitrary time steps. To accommodate diverse training scenarios, we have developed both a supervised version that utilizes paired HDR ground truth data and a self-supervised version that relies on photometric and temporal consistency. This makes our method applicable even in unlabeled settings.

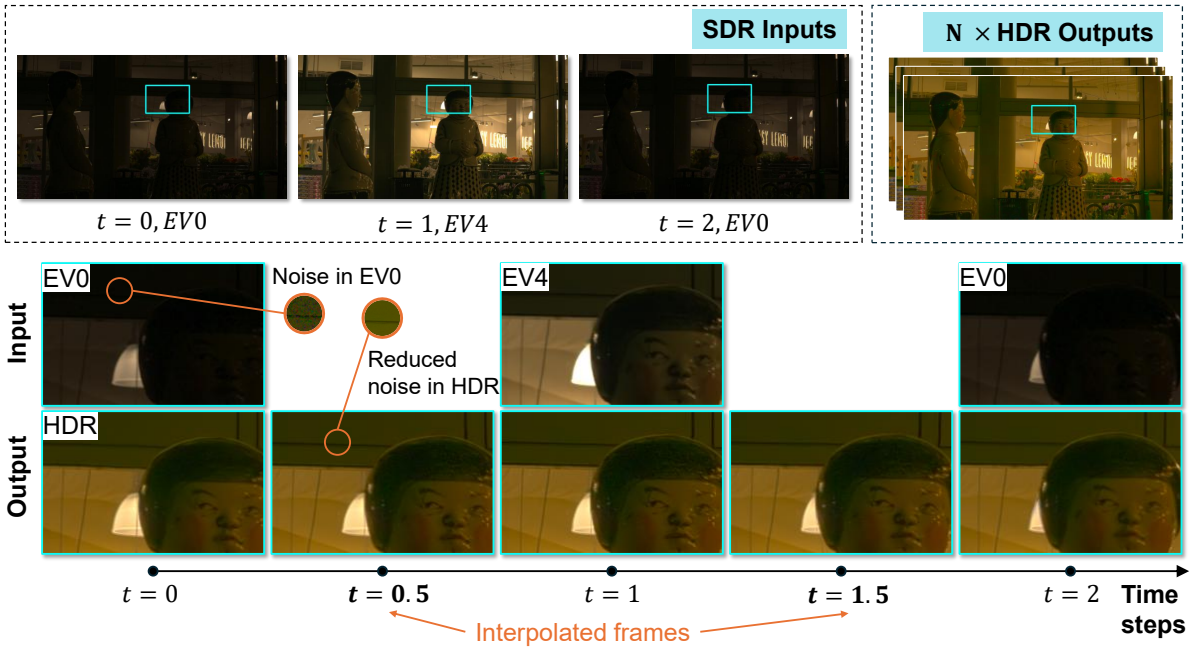


Figure 1.1: Generating high-frame-rate HDR video from alternating exposures. We propose a method for joint high dynamic range video reconstruction and interpolation at **arbitrary timesteps** using a sequence of alternating exposures between a base exposure (EV0) and a higher exposure (*e.g.*, EV4). Input frames often exhibit noise in dark regions (EV0) and loss of detail in under- and over-exposed areas (EV0/EV4). Our method generates **clean shadows and detailed highlights** with temporal consistency while increasing the frame rate through interpolation.

Our method delivers temporally coherent, high-quality HDR video at higher effective frame rates while maintaining low computational and memory overhead. As illustrated in Figure 1.1, the system achieves smooth motion interpolation and rich detail reconstruction. The quantitative comparison in Figure 1.2 demonstrates our method's ability to match or outperform existing techniques while requiring significantly less computational time. This work bridges the gap between HDR video reconstruction and frame interpolation, offering a practical solution for scalable HDR video synthesis.

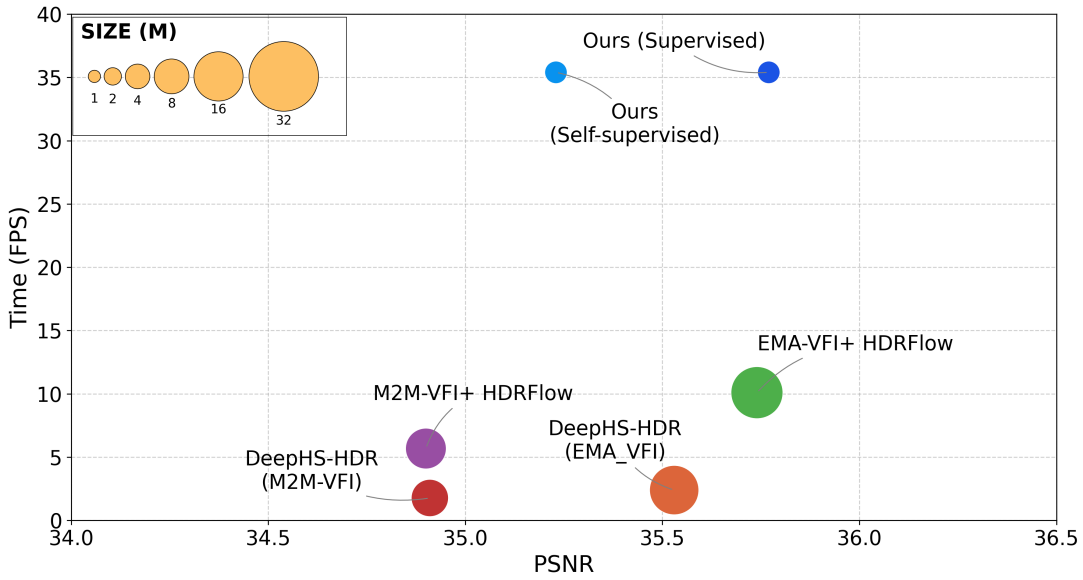


Figure 1.2: Comparison of computational cost and reconstruction quality with prior approaches. Our method is more efficient while maintaining comparable or better quality than state-of-the-art baselines.

1.2 Thesis Contributions

The key contributions of this thesis are:

- We introduce a unified, lightweight neural architecture for joint HDR reconstruction and frame interpolation. This integrated design not only reduces memory usage and computational overhead but also allows seamless integration into existing HDR pipelines to enable interpolation functionality.

- We propose two training strategies: a supervised method using ground-truth HDR data, and a novel self-supervised framework that enables fine-tuning of any pre-trained HDR video reconstruction network for the new interpolation task.

- We conduct extensive experiments to evaluate the performance of our proposed method. Our findings show that our approach achieves comparable performance in both supervised and self-supervised settings while requiring fewer computational resources and less memory.

1.3 Thesis Outline

The remainder of this thesis is organized as follows. Chapter 2 provides background information on HDR imaging and reviews related work on HDR video reconstruction and interpolation. Chapter 3 presents our proposed method in detail. Experimental results and evaluations are discussed in Chapter 4. Finally, Chapter 5 concludes the thesis and outlines potential directions for future research.

Chapter 2

Background and Related Work

High-frame-rate high-dynamic-range video synthesis is an emerging research area at the intersection of two fundamental problems in computational photography: HDR video reconstruction and video frame interpolation. Although significant progress has been made in these areas independently, integrating them into a unified framework introduces unique challenges, particularly in handling exposure variability, motion dynamics, and temporal consistency. This chapter provides the technical background to understand these challenges and situates the proposed work within the broader research landscape.

We begin by reviewing the motivation behind HDR imaging and its principles. Next, we examine exposure bracketing, a widely used technique for capturing HDR content. We then present video frame interpolation, focusing on deep-learning approaches. Finally, we review existing methods that attempt to jointly tackle HDR reconstruction and video frame interpolation (VFI), highlighting their limitations. This comprehensive review highlights key gaps in the literature and motivates the contributions introduced in the remainder of this thesis.

2.1 High Dynamic Range Imaging

HDR imaging aims to overcome the limitations of conventional SDR systems by enabling the capture, representation, and display of the full range of light intensities in natural scenes. This section introduces the motivation behind this technique and the fundamental concepts necessary to understand how HDR images differ from traditional SDR

counterparts regarding acquisition, storage, and display.

Motivation. HDR imaging is motivated by the need to capture and reproduce the full range of light intensities in natural scenes. This range, known as the *dynamic range*, represents the proportion between the brightest and darkest components an imaging system can measure. Dynamic range is typically expressed in units of *f-stops*, calculated as $\log_2\left(\frac{Y_{\max}}{Y_{\min}}\right)$, where $Y_{\max}$ and $Y_{\min}$ denote the maximum and minimum measurable luminance, respectively. It can also be quantified using other metrics such as contrast ratios or signal-to-noise ratios in decibels [1].

While the human visual system can perceive contrasts of over 100,000:1 (approximately 17 f-stops), conventional SDR imaging systems typically capture only 8–14 f-stops under ideal conditions [1]. This limitation stems primarily from the restricted native dynamic range of consumer-grade image sensors, which struggle in scenes with strong contrast. For example, consider a scene viewed through a shaded window into bright daylight. A single exposure setting may either preserve highlight detail outdoors while rendering the indoor region too dark or recover shadows indoors at the expense of blown-out highlights. Figure 2.1 illustrates this dilemma: neither a low nor a high exposure alone can faithfully preserve details across the entire scene.

To overcome this limitation, HDR imaging techniques are used to extend the effective dynamic range beyond what a single exposure can offer. A common approach is exposure bracketing, where multiple images of the same scene are captured at varying exposure values and merged into one HDR image. Another route involves using specialized sensors with a wider native dynamic range. Both strategies aim to preserve structural and color details across bright and dark regions, ensuring a more faithful representation of the real-world scene.

Storing HDR Content. In contrast to SDR images, which are typically stored using 8-bit gamma-compressed formats (*e.g.*, sRGB), HDR images are often stored in a scene-referred representation with a higher bit depth (*e.g.*, using 16- or 32-bit floating point). This means the pixel values directly reflect physical scene radiance or sensor irradiance rather than being adjusted for display. This approach allows HDR images to preserve their full dynamic range and adapt flexibly for various output devices.

(a) Image captured with low EV (b) Image captured with high EV

Figure 2.1: Example of a scene whose dynamic range exceeds the capture capabilities of a standard camera. No single exposure can preserve both highlight and shadow detail. (a) A low-exposure setting preserves outdoor detail but loses interior information. (b) A high exposure setting recovers interior details but causes overexposure in the outdoor regions. The images are from HDR+ Burst Photography Dataset [14].

Additionally, this representation benefits a wide range of downstream applications. Tasks like relighting, tone mapping, and intrinsic image decomposition benefit from having physically accurate luminance values. Scene-referred HDR also supports algorithmic reasoning about illumination, exposure, and scene surface geometry, making it essential for high-quality rendering and computational photography workflows.

Displaying HDR Content. While some modern displays—such as OLED panels in high-end smartphones, HDR-enabled televisions, and reference monitors—support the direct rendering of HDR content using standards like HDR10 or Dolby Vision, the vast majority of consumer displays still operate within the narrower Standard Dynamic Range (SDR). As a result, displaying HDR imagery on these devices requires mapping the wide luminance range of HDR data into the limited range supported by SDR displays. One naïve approach is simply clamping or rescaling HDR values into an 8-bit SDR range, often discarding highlight or shadow information (see Figure 2.2c). A more effective alternative is to apply a tone mapping operator, which compresses the dynamic range while aiming

to preserve important perceptual or structural information.

Tone mapping serves different purposes depending on the target application. In artistic workflows, the focus may be on enhancing the image's visual impact or mood, while in technical or scientific contexts, the goal is typically to preserve fine details, luminance relationships, or perceptual contrast [15]. Regardless of the objective, tone mapping is generally performed using either global or local methods or a hybrid of both. Global tone mapping [16, 17] applies a single tone curve uniformly across the image, offering computational simplicity and predictable results, but often at the expense of local contrast. Local tone mapping [18, 19], by contrast, adapts the tone curve based on spatial context—such as local luminance statistics—allowing for better preservation of details in both highlights and shadows across different regions of the scene. For a comprehensive overview of tone mapping algorithms and their trade-offs, readers are referred to the work of Mantiuk *et al.* [20].

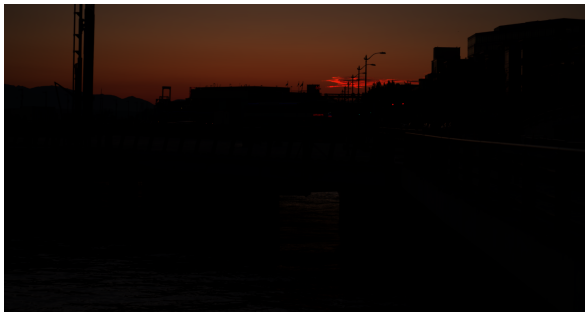
For visualization purposes, all HDR images displayed in this thesis are tone-mapped using Adobe Photoshop's HDR local tone mapping operator. This operator was selected for its strong balance between contrast enhancement and detail preservation, which aligns well with our goals of qualitative comparison. Figure 2.2b shows an example of such tone-mapped output.

2.2 Exposure Bracketing for HDR Imaging

Exposure bracketing is a widely used and effective method for capturing HDR content using standard image sensors. This section discusses basic acquisition strategies and highlights the unique challenges encountered when applying these methods to dynamic scenes and video capture.

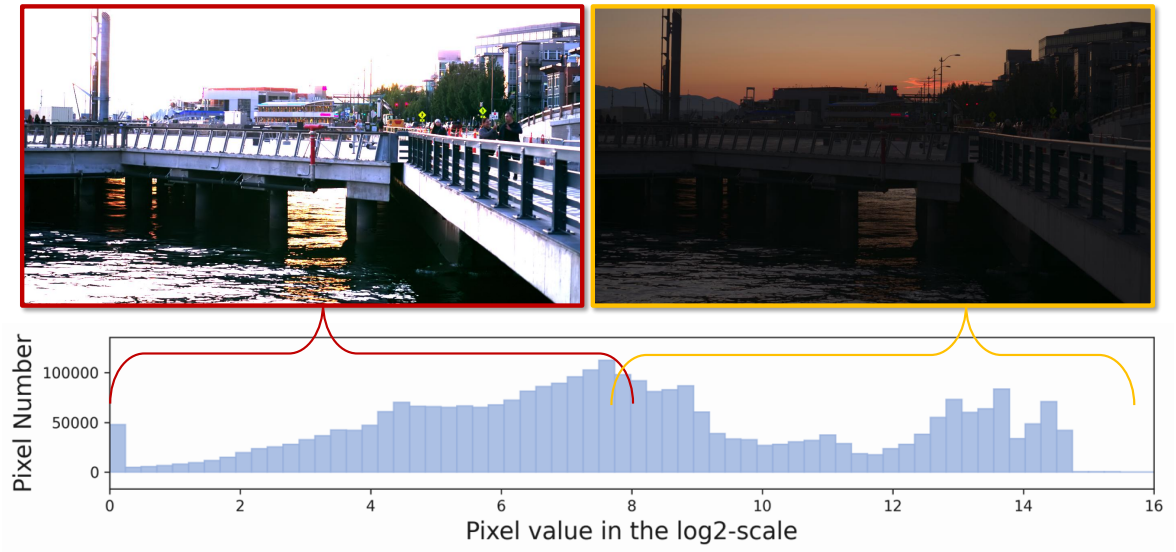
Acquisition Methods. The central concept of exposure bracketing involves capturing multiple SDR images of the same scene, each using a different exposure value (EV). While each covers only a limited portion of the scene's dynamic range, they capture a much wider range of luminance levels together. This collection can then be merged into a single HDR image, allowing for reconstructing a complete scene radiance map when properly fused.

The exposure of an image is determined by three parameters: ISO, shutter speed, and



(a) Linear radiance image.

(b) Photoshop tone-mapped image.



(c) SDR versions with BT.709 gamma.

Figure 2.2: Comparison of different visualizations of an HDR frame. We show (a) the original HDR image in linear radiance domain, (b) the result after applying Photoshop's HDR tone mapping operator, and (c) two standard dynamic range (SDR) visualizations using selected 8-bit value ranges and gamma correction ($\gamma = 2.4$) to simulate BT.709 display characteristics. This comparison highlights how tone mapping preserves perceptual detail better than naïve SDR conversions.

aperture. ISO controls the sensitivity of the camera sensor to light (i.e., the gain applied to boost pixel responses to scene radiance). Higher ISO settings enable proper exposure in low-light conditions but increase the risk of noise, whereas lower ISO values require more light, reducing the chance of sensor saturation. Shutter speed determines the duration for which the sensor is exposed to light. While a longer shutter speed can reduce noise, it may also introduce motion blur or lead to clipping in bright regions due to overexposure. Aperture refers to the size of the lens opening and regulates the amount of light entering the camera. Adjusting the aperture affects the depth of field and can introduce blurriness for objects outside the focal plane. Therefore, in most HDR bracketing workflows, the aperture is kept constant to avoid unwanted optical effects. Instead, exposure is adjusted by varying the ISO and shutter speed. In general, the choice among these parameters depends on the scene dynamics and lighting conditions.

Bracketing often involves capturing two or more scene exposures in quick succession. While this method is effective for still images, but is impractical for video recording. Capturing multiple exposures for each frame leads to processing overhead and reduces the temporal resolution. To address this challenge, recent acquisition pipelines have adopted an *alternating exposure* strategy. In this approach, the camera captures a continuous stream with varying exposure levels in an interleaved manner (for example, alternating between short and long exposures). This method allows the system to gather footage with built-in exposure diversity, creating a dynamic exposure stack that is suitable for HDR video reconstruction.

HDR Image Reconstruction Methods. Reconstructing HDR images from an exposure stack involves combining multiple images of the same scene, each taken at different exposure times, into a single image that accurately represents the scene's radiance. This concept was initially introduced by Mann and Picard [21] and later refined by Debevec and Malik [22], who developed a method to estimate both the camera's response function and the HDR image.

A camera response function f is defined as the function that maps the image irradiance E into pixel values $P = f(E)$. Assuming the camera response function f is monotonic

and hence invertible, we can express the irradiance as:

$$E = f^{-1}(P). \tag{2.1}$$

Since image irradiance E and scene radiance L are related $E = ePL$, where e is the exposure and P is a factor depending on the optics of the system [23], if we capture the same scene at different exposure durations e_i, the irradiance at pixel location j in image i is:

$$E_j^i = E_j \cdot e_i, \tag{2.2}$$

where E_j represents the irradiance at a reference (base) exposure. Substituting Eq. 2.1, we have:

$$f^{-1}(P_j^i) = E_j \cdot e_i, \tag{2.3}$$

where P_j^i is the pixel value at location j in the i^{th} image. Taking the logarithm of both sides and defining $g = \log(f^{-1})$, we obtain:

$$g(P_j^i) = \log(E_j) + \log(e_i). \tag{2.4}$$

This equation forms the basis of the algorithm. By taking multiple pixel values P_j^i at the same location j across different exposures e_i, we can solve a least-squares optimization problem to recover both the response function $g(\cdot)$ and the log irradiance $\log(E_j)$.

$$\sum_{i,j} \left[g(P_j^i) - \log(E_j) - \log(e_i)\right]^2 + \lambda \sum_{z=Z_{\min}+1}^{Z_{\max}-1} \left[g''(z)\right]^2, \tag{2.5}$$

where Z denotes the domain of valid pixel values, and λ is a smoothness weighting parameter. The first term ensures consistency with the observed data, while the second regularizes g by penalizing large second derivatives, encouraging a smooth and physically plausible response curve.

Once the response curve g is recovered, the log irradiance is estimated as:

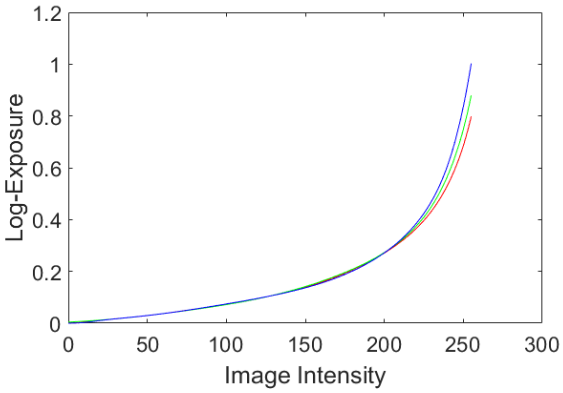
$$\log(E_j) = g(P_j^i) - \log(e_i). \tag{2.6}$$

To improve robustness to sensor noise and saturation, the final HDR image is recovered by computing a weighted average of image irradiance across all exposures:

$$\log(E_j) = \frac{\sum_i w(P_j^i) \cdot \left[g(P_j^i) - \log(e_i)\right]}{\sum_i w(P_j^i)}, \tag{2.7}$$

with $w(\cdot)$ is a weighting function that assigns higher weights to pixel values near the middle of the dynamic range, where the response function is most reliable [22]. An illustration of this entire pipeline—from the exposure stack and inverse CRF estimation to the reconstructed HDR output—is provided in Figure 2.3.

(a) Input exposure stack.



(b) Inverse camera response function.

(c) Reconstructed HDR output.

Figure 2.3: Illustration of the Debevec and Malik HDR reconstruction pipeline [22], using an example from Banterle *et al.* [24]. (a) An exposure stack of images captured at different EVs. (b) The estimated inverse camera response function, which maps pixel intensities back to scene radiance. (c) The reconstructed HDR image, visualized with Photoshop's tone mapping for display.

Although this method works well under static conditions, its effectiveness diminishes significantly in dynamic scenes where objects or the camera move between exposures. Early approaches tackled this issue by globally aligning input images and discarding pixels affected by motion before fusion [25, 26, 27, 28, 29]. However, this strategy often fails when the moving regions contain important HDR content that cannot be discarded.

To address this limitation, later methods introduced motion compensation techniques, such as dense correspondences or optical flow. For example, Hu *et al.* [30] utilized Ha-Cohen *et al.*'s method [31] to map each pixel in the input stack to its counterpart in a

reference image. Similarly, other methods [32, 33] employed optical flow to warp input frames before merging. While these techniques improve alignment, they are still vulnerable to ghosting artifacts in challenging scenarios due to their reliance on precise motion estimation.

With the rise of deep learning, several approaches have been developed to learn HDR reconstruction from captured images directly. Kalantari *et al.* [34] proposed a paired SDR-HDR dataset and trained a convolutional neural network to fuse pre-aligned inputs, where alignment is performed using the optical flow method of Liu [35]. Catley-Chandar *et al.* [36] took a step further by jointly learning optical flow and exposure reliability maps, providing richer guidance for the fusion process. Other methods [37, 38, 39] leverage attention mechanisms to perform alignment in the feature space, achieving strong results in dynamic scenes and effectively recovering fine details. However, most of these models require ground-truth HDR images for supervision, which are expensive and difficult to acquire.

Recent works have explored few-shot and self-supervised HDR reconstruction methods to overcome the need for labeled data. These approaches typically create pseudo-ground-truth data to train the network. For instance, Prabhakar *et al.* [40] initially trained their model on a small labeled dataset using supervised loss while applying a weakly supervised one to unlabeled samples. The trained model is then used to generate pseudo inputs and targets for continued training on the full dataset. In contrast, Nazarczuk *et al.* [41] propose a fully self-supervised framework that extracts pseudo training pairs from static and well-exposed regions within the input stack. Another method, SelfHDR [42], uses a pre-trained optical flow model to align the input stack and applies a simple weighting function to generate pseudo-ground-truth images. During training, the model learns from these pseudo labels and the mid-exposure reference image, providing adequate supervision without manual annotation.

HDR Video Reconstruction Methods. Recent advances have extended the concept of exposure bracketing to video by capturing frames with alternating exposures. This idea was first introduced by Kang *et al.* [7], who used global and local registration techniques to align neighboring frames to a reference frame, enabling the generation of HDR video at the same frame rate as the input sequence. Building on this, Mangiat and Jerry [43]

improved alignment accuracy using block-based motion estimation combined with motion vector refinement.

The rise of deep learning has pushed this field further. Kalantari *et al.* [44] proposed the first end-to-end CNN-based framework for HDR video, incorporating a flow network for alignment and a weighted network for merging the aligned SDR frames. More recently, Xu *et al.* [12] introduced a lightweight optical flow model utilizing multi-size large kernels, allowing for efficient alignment and real-time HDR video synthesis. While these methods demonstrate strong performance and efficiency, they depend on supervised training with paired HDR ground truth. Acquiring such ground truth for video is particularly challenging, requiring high temporal resolution, radiometric consistency, and pixel-accurate spatial alignment.

To address the challenge of collecting HDR video data, Shu *et al.* [11] proposed a stop-motion capture strategy, where static exposure stacks are sequentially recorded to simulate a video sequence. While this technique simplifies alignment across exposures, it does not reflect real-world video conditions—dynamic motion, continuous exposure changes, and real-time scene evolution are absent. As illustrated in Figure 2.4, consecutive frames exhibit unnatural motion patterns due to the lack of genuine temporal continuity. If a model is trained exclusively on such datasets, these fundamental limitations may impair its ability to generalize to real-world video data. Moreover, since accurate timestamp information is missing, these datasets are unsuitable for time-sensitive tasks such as frame interpolation.

These challenges highlight the need for a more data-efficient training framework. A straightforward idea is to adapt existing self-supervised frameworks developed for images to video. However, this adaptation is non-trivial, as several assumptions that hold for images break down in the video setting. For instance, SelfHDR [42] assumes that the target frame has a mid-level exposure, an assumption that is reasonable for static image stacks. In HDR video, however, the target frame might be a high-exposure input with highlight clipping, leading to structural detail loss. Consequently, directly applying SelfHDR to video often results in unreliable reconstructions.

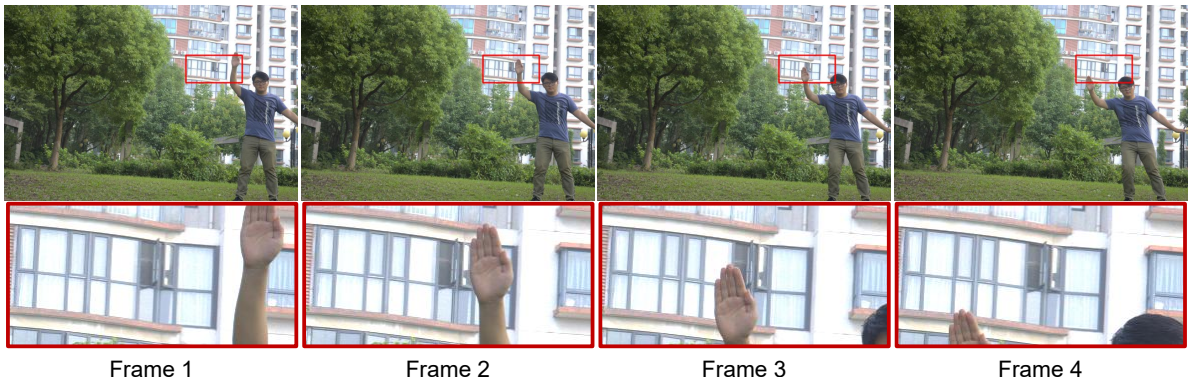


Figure 2.4: Illustration of unnatural motions in a ground truth sequence from the Real-HDRV dataset [11]. The figure shows four consecutive HDR frames captured using a stop-motion setup. Despite being labeled as temporally adjacent, the frames exhibit noticeable discontinuities in human motion due to the lack of true temporal continuity. Such datasets fail to capture realistic scene dynamics and natural motion trajectories, making them unsuitable for evaluating temporal interpolation methods.

2.3 Video Frame Interpolation

Video frame interpolation (VFI) is the task of synthesizing intermediate frames between two given frames to increase the temporal resolution of a video. VFI has widespread applications in frame rate up-conversion, slow-motion video synthesis, video compression, and temporal super-resolution. In this section, we provide a brief historical overview of VFI, followed by a detailed discussion of deep learning-based approaches and their limitations when applied to HDR content.

Traditional methods. Frame interpolation has long been employed in video compression standards such as MPEG [45] and HEVC [46], where intermediate frames are predicted using motion-compensated block matching to reduce temporal redundancy. These methods represent motion using block-based motion vectors and rely on the assumption that neighboring frames contain mostly static or smoothly varying content.

However, traditional block-based approaches often suffer from blocking artifacts and fail in the presence of non-rigid motion, occlusions, or complex lighting variations. Their reliance on simple motion models makes them inadequate for high-quality interpolation,

especially in dynamic scenes.

Beyond block matching, more recent non-learning-based methods have explored analytic models for interpolation. For example, Fiquet *et al.* [47] proposed a Fourier domain approach that leverages the Fourier Shift Theorem to interpolate global motion in the frequency domain. While analytically elegant and computationally efficient, these techniques are limited to small, global displacements and are not well-suited for real-world scenes with local motion and depth variation.

Deep Learning-Based Interpolation Methods. With the advent of deep learning, VFI has seen significant advancements, achieving state-of-the-art performance across many benchmarks. At the heart of modern methods lies the challenge of capturing motion and synthesizing temporally coherent content. Different approaches address this challenge in various ways depending on how they represent and process temporal information.

Some widely used methods involve estimating bi-directional optical flow between the input frames and warping them toward the desired intermediate timestamp, such as Super SloMo [48], M2M-VFI [49], and QVI [50]. M2M-VFI estimates forward flow maps to guide pixel warping, while QVI enhances this by modeling motion trajectories as quadratic functions of time, improving performance in the presence of non-linear motion. Despite their success, flow-based methods often struggle in occluded regions and typically require auxiliary modules for occlusion handling and refinement.

As an alternative to explicit flow estimation, kernel-based methods [51, 52, 53] directly predict spatially adaptive convolution kernels for each pixel. These kernels are used to blend corresponding patches from the input frames to produce the interpolated output. While kernel-based methods are effective for local motion and avoid artifacts from inaccurate warping, they are computationally expensive and typically scale poorly for large or non-rigid motion.

More recently, researchers have begun to move away from explicit motion modeling altogether. Instead, these methods perform interpolation directly in the latent feature space of deep neural networks. For example, FLAVR [54] uses 3D convolutional layers to synthesize intermediate frames at the pixel level. Transformer-based methods [55, 56] take a different approach by modeling long-range spatiotemporal dependencies using attention mechanisms, enabling improved handling of occlusions and complex motion patterns.

Despite the significant advancements achieved by deep learning-based video frame interpolation methods, most of these techniques are specifically trained for SDR content. As such, they are not directly applicable to HDR video, which contains a wider luminance range. To apply these methods to HDR content, a simple workaround can be compressing the HDR frames into SDR through tone mapping or range normalization before feeding them into the model. After interpolation, the resulting frames are re-expanded or remapped to HDR. However, this two-stage conversion process may discard important critical visual cues for accurate motion estimation and frame synthesis. As a result, the interpolation quality may degrade, preventing the general applicability of existing VFI models in HDR settings.

2.4 High-Frame-Rate HDR Video Reconstruction

The goal of high-frame-rate high dynamic range (HFR-HDR) video reconstruction is to synthesize temporally dense video with a wide dynamic range, capturing both fine temporal detail and accurate radiometric information. This task inherently combines the challenges of video frame interpolation and HDR reconstruction, requiring models to reason about motion, exposure variation, and temporal coherence simultaneously.

Hardware-Dependent Approaches. Early efforts to achieve HFR-HDR video synthesis have often relied on custom hardware systems. Chang *et al.* [6] combined a high-speed monochrome spike camera with a low-frame-rate RGB sensor using alternating exposures. This hybrid setup enabled 1000 FPS HDR video capture by fusing complementary radiometric and temporal information from two asynchronous sources. Similarly, Çoğalan *et al.* [57] employed dual-exposure sensors to simultaneously capture low- and high-exposure views, which were then merged into an HDR stream.

While these approaches demonstrate strong performance in controlled settings, they depend on complex, expensive hardware configurations that are not accessible to most consumers or mobile platforms. Additionally, tight synchronization and calibration between sensors make these pipelines difficult to deploy at scale.

Software-Based Alternatives. To overcome the reliance on specialized hardware, recent research has explored software-only pipelines for HFR-HDR video synthesis. These approaches operate on commodity sensors by exploiting exposure-alternating sequences, captured by toggling between short and long exposures across consecutive frames. A representative example is DeepHS-HDRVideo by Khan *et al.* [13], which introduces a recursive multi-stage framework. Their method first applies a pre-trained frame interpolation model (*e.g.*, Super SloMo or QVI) to align same-exposure frame pairs (*e.g.*, short-to-short or long-to-long), and then merges the interpolated results using a supervised HDR fusion network trained to reconstruct scene-referred radiance.

By decoupling interpolation from HDR reconstruction, DeepHS-HDR enables recursive temporal upsampling without modifying camera hardware. However, this separation introduces several significant limitations. First, the multi-stage pipeline incurs considerable computational overhead due to the sequential execution of interpolation and fusion modules, which poses challenges for real-time deployment, particularly on mobile or resource-constrained platforms. Second, the method relies exclusively on same-exposure pairs for interpolation, discarding temporally closer frames that differ in exposure. For instance, in an alternating exposure sequence such as SL-SL, the model interpolates between two distant short-exposure frames while ignoring the intermediate long-exposure frame, which is temporally closer and often shares complementary scene content. This underutilization of available data can result in motion estimates that are less accurate or perceptually inconsistent, especially in fast-moving or high-contrast scenes.

Besides the architecture limitation, the progress in this domain has been constrained by the absence of large-scale benchmark datasets tailored specifically for HFR-HDR reconstruction from alternating exposures. Most existing HDR datasets either lack sufficient temporal resolution or are not designed with exposure-alternating sequences in mind, making them unsuitable for learning-based joint HDR and interpolation tasks. This lack of training and evaluation data poses a significant barrier to advancing supervised methods in this area.

These limitations motivate the need for a unified and data-efficient framework that jointly handles HDR reconstruction and frame interpolation. Such an approach can better exploit exposure redundancy and model motion while minimizing inference costs.

Chapter 3

Methodology

This chapter presents our proposed approach for reconstructing high-frame-rate high-dynamic-range (HFR-HDR) video from low-frame-rate, alternating-exposure inputs. We begin by providing an overview of the pipeline, which is designed to reconstruct HDR frames at arbitrary intermediate timesteps. Next, we describe the core components of our network architecture, including a novel temporally-aware interpolation module, InterpNet. This module can be incorporated into an existing learning-based HDR reconstruction framework to support continuous-time reconstruction. Finally, we present two distinct training strategies: a supervised variant that leverages HDR ground truth, and a self-supervised variant that operates solely on SDR inputs. Collectively, these components enable efficient, temporally consistent HFR-HDR video reconstruction while minimizing computational overhead.

3.1 Pipeline Overview

Given a SDR video sequence $\{\tilde{S}_i \mid i = 1, \ldots, n\}$ captured at a low frame rate (*e.g.*, 30 FPS) using alternating exposure values, our goal is to reconstruct a high-frame-rate (*e.g.*, 60 FPS) HDR video. Specifically, our model takes three consecutive SDR frames $\tilde{S}_0, \tilde{S}_1, \tilde{S}_2$ and a target intermediate time $t \in (0, 2)$ as input, and outputs the corresponding HDR frame in linear domain at time t. Following prior work [58, 59, 9], we assume the camera response function (CRF) [60] of the SDR frames is known. Then, to simplify processing and ensure consistency across different cameras and settings, we replace the original CRF

with a fixed gamma curve using $S_i = \left(\mathcal{F}^{-1}(\tilde{S}_i)\right)^{1/\gamma}$ with $\gamma = 2.2$ [9].

Figure 3.1 illustrates an overview of our pipeline. We extend existing HDR video reconstruction frameworks by introducing a novel interpolation module for synthesizing HDR frames at any intermediate time. The entire pipeline can be trained end-to-end with ground-truth data or via our self-supervised framework, reducing the need for paired supervision

3.2 Network Architecture

As illustrated in Figure 3.1, conventional HDR video reconstruction frameworks [12, 9, 44] typically consist of a flow estimation module followed by a fusion network. We extend this architecture with a novel interpolation module, named *InterpNet*, which synthesizes temporally aligned HDR frames by interpolating optical flow as an implicit quadratic function of time.

Given three consecutive alternating-exposure frames S_0, S_1, and S_2, our pipeline first computes bidirectional optical flows $f_{1\to0}$ and $f_{1\to2}$ to capture motion between the central frame and its adjacent neighbors. These flows are estimated using a frozen FlowNet module from HDRFlow [12], which we adopt as the foundation of our framework. HDRFlow consists of two main components: a FlowNet for motion estimation and a FusionNet for HDR reconstruction. To preserve the pre-trained motion representations and reduce training costs, we keep the FlowNet weights fixed during training. In addition to the flow fields, we extract final-layer feature maps from FlowNet, providing motion-aware contextual information for downstream modules.

Unlike the original HDRFlow, which upsamples flow maps to full resolution via bilinear interpolation, we retain the native 1/4-scale resolution output by FlowNet. This approach helps to avoid artifacts from interpolation and allows for more accurate modeling in subsequent stages. To estimate the optical flow at an arbitrary intermediate time $t \in (0, 2)$, we input the bidirectional flows, motion features, and the scalar t into our proposed InterpNet module. InterpNet uses this temporally-aware input to produce full-resolution flow fields that warp the input frames to the target time t.

The core design of InterpNet is motivated by the observation that pixel trajectories

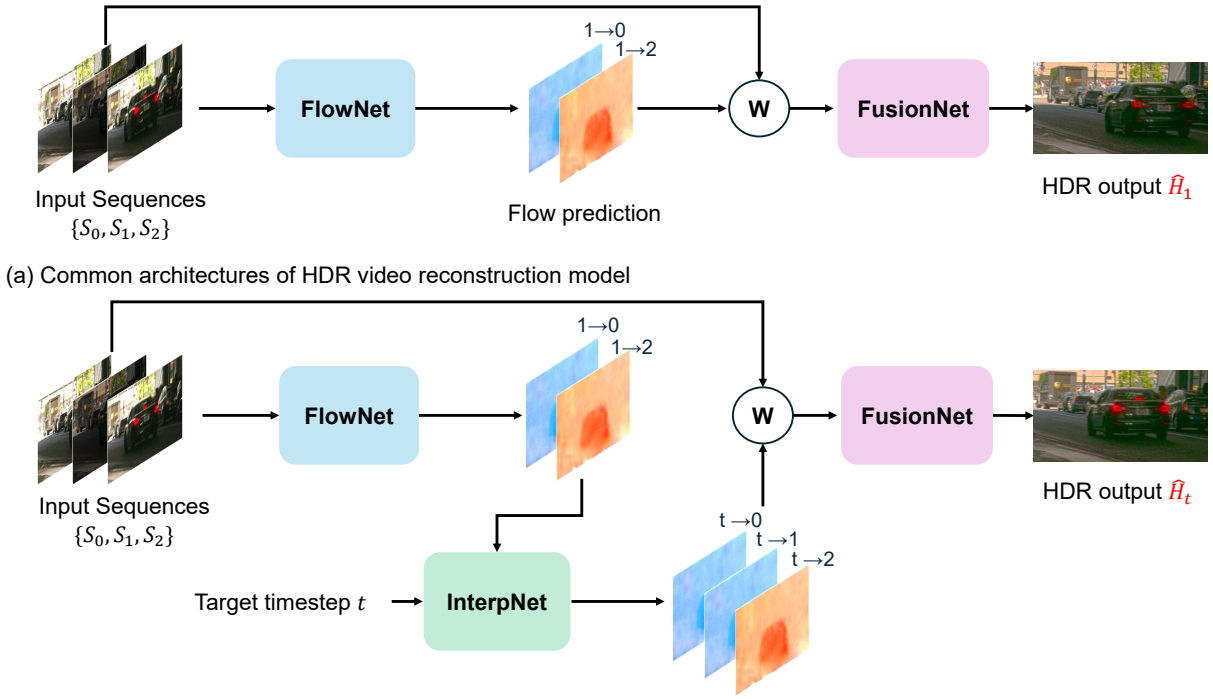


Figure 3.1: Overview of our network architecture. We introduce a novel network designed to recover both high dynamic range and temporal information from a sequence of alternating exposures. (a) Standard HDR video reconstruction pipelines typically fuse a set of SDR frames with varying exposures to produce a single HDR output at a fixed timestamp. (b) Our model builds on this structure by incorporating an interpolation module that enables generation of HDR frames at an **arbitrary** time steps. This design allows for efficient frame-rate upsampling and seamless reuse of pre-trained HDR reconstruction models.

across frames often follow a quadratic path in time [50, 61], due to smooth object or camera motion. Specifically, the flow from a given input frame to an intermediate frame at time t can be modeled as a function of t, t^2, and the known flows between neighboring frames. To exploit this prior, we transform the original flows into a higher-dimensional, temporally-aware latent space using the following quadratic basis:

$$\phi(t; f_{1\to0}, f_{1\to2}) = \left[f_{1\to0},\ f_{1\to0}\cdot t,\ f_{1\to0}\cdot t^2,\ f_{1\to2},\ f_{1\to2}\cdot t,\ f_{1\to2}\cdot t^2\right]. \tag{3.1}$$

This basis-encoded representation captures temporal dependencies and structures the input space for flow interpolation. To enhance expressiveness and robustness in complex scenes, we concatenate this representation with the final-layer feature maps extracted from the frozen FlowNet. These features carry rich motion-aware context, helping the model reason about non-linear motion patterns, occlusions, and ambiguous regions where direct interpolation may fail. To restore full spatial resolution, we apply a *LinearScaling* layer that scales all flow components by a factor of 4, compensating for the 1/4-scale resolution of the original flow fields.

The resulting tensor is then passed through a lightweight convolutional backbone comprising a series of 3×3 standard and deformable convolution layers. The deformable convolutions provide adaptive receptive fields that better capture geometric variations and misalignments across frames. Finally, InterpNet predicts three full-resolution flow fields: $f_{t\to0}$, $f_{t\to1}$, and $f_{t\to2}$. These flows are used to warp the original input frames S_0, S_1, and S_2 toward the target time t, producing temporally aligned inputs for HDR fusion. The overall architecture of InterpNet is visualized in Figure 3.2.

To produce the final HDR output, we adopt FusionNet, a U-Net-style architecture originally proposed in HDRFlow [12], which predicts spatial fusion weights to combine linear irradiance frames into a single HDR image $\hat{H}_t$. In HDRFlow, FusionNet takes five inputs, including the frames at times 0, 1, and 2 (S_0, S_1, S_2), and the warped frames $(S_{0\to1}, S_{2\to1})$. To maintain compatibility with this input structure, we adapt the setup by replacing the original warped frames with the corresponding warps toward time t. Specifically, $S_{0\to1}$, S_1, and $S_{2\to1}$ are substituted by $S_{0\to t}$, $S_{1\to t}$, and $S_{2\to t}$ correspondingly. This input configuration allows us to reuse pre-trained FusionNet weights without modification.

During training, FlowNet remains frozen to preserve its pretrained motion representa-

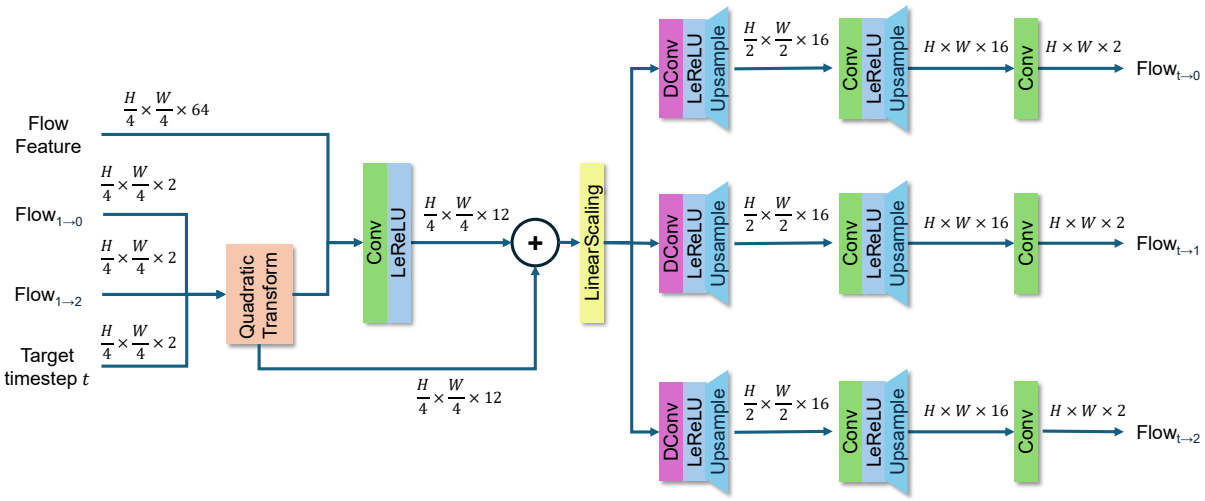


Figure 3.2: InterpNet architecture. The network takes as input low-resolution optical flows, flow features, and a scalar target time t, and outputs full-resolution intermediate flows for warping input frames toward the target timestamp. Each layer is annotated with its spatial resolution and feature depth. "Conv" denotes a 3×3 convolutional layer; "DConv" refers to a 3×3 deformable convolution layer; "LeReLU" indicates a leaky ReLU activation.

tions, while only the parameters of InterpNet and FusionNet are updated. This modular design reduces training cost, accelerates convergence, and facilitates integration into existing HDR reconstruction pipelines with minimal overhead.

3.3 Training Framework

We introduce two training paradigms for our network: a **supervised** approach that leverages high-frame-rate HDR ground truth when available, and a **self-supervised** alternative that relies solely on alternating-exposure SDR videos.

3.3.1 Supervised Training

Given HDR ground truth H_t^{gt} at time t, we apply a reconstruction loss on both the predicted HDR frame $\hat{H}_t$ and the temporally aligned ground truth frames:

$$\mathcal{L}_{sup} = \mathcal{L}_{rec}(\mathcal{T}(\hat{H}_t), \mathcal{T}(H_t^{gt})) + \sum_{i=0}^{2} \mathcal{L}_{rec}(\mathcal{T}(\mathcal{W}(H_i^{gt}, F_{t\to i})), \mathcal{T}(H_t^{gt})). \tag{3.2}$$

Here, $\mathcal{T}(\cdot)$ is the differentiable μ-law tone mapping (Eq. 3.3), and $\mathcal{W}(I, F)$ warps image I using flow F. Following prior work [11, 13, 12], the reconstruction loss $\mathcal{L}_{rec}$ is computed in the tone-mapped domain to better reflect perceptual differences. This can be expressed as:

$$\mathcal{T}(H) = \frac{\log(1+\mu H)}{\log(1+\mu)}, \tag{3.3}$$

with $\mu = 5000$. The reconstruction loss is then defined as a linear combination of the L1 loss and a multi-scale Laplacian pyramid loss with five levels [62]:

$$\mathcal{L}_{\text{rec}}(I_1, I_2) = \|I_1 - I_2\|_1 + \sum_{l=1}^{5} \|L^l(I_1) - L^l(I_2)\|_1, \tag{3.4}$$

where $L^l(\cdot)$ denotes the Laplacian pyramid at level l. The Laplacian pyramid term encourages fidelity in fine-grained textures and edges that are often underemphasized by simple pixel-wise losses [62].

3.3.2 Self-Supervised Training

To eliminate the need for HDR ground truth, we propose a self-supervised framework using only alternating-exposure input. Our approach is inspired by classical HDR fusion, which combines multi-exposure SDR images S^i (with exposures e_i) into an HDR image H^{fusion} using linear irradiance $l^i = \text{CRF}^{-1}(S^i)/e_i$ and pixel-wise exposure-aware weights $w(S^i)$:

$$H^{\text{fusion}} = \frac{\sum_i w(S^i) \cdot l^i}{\sum_i w(S^i)}, \tag{3.5}$$

where $\{S_t^i\}$ are SDR frames of varying exposures available at timestep t. This equation represents how an HDR image would ideally be formed from multiple exposures. If the

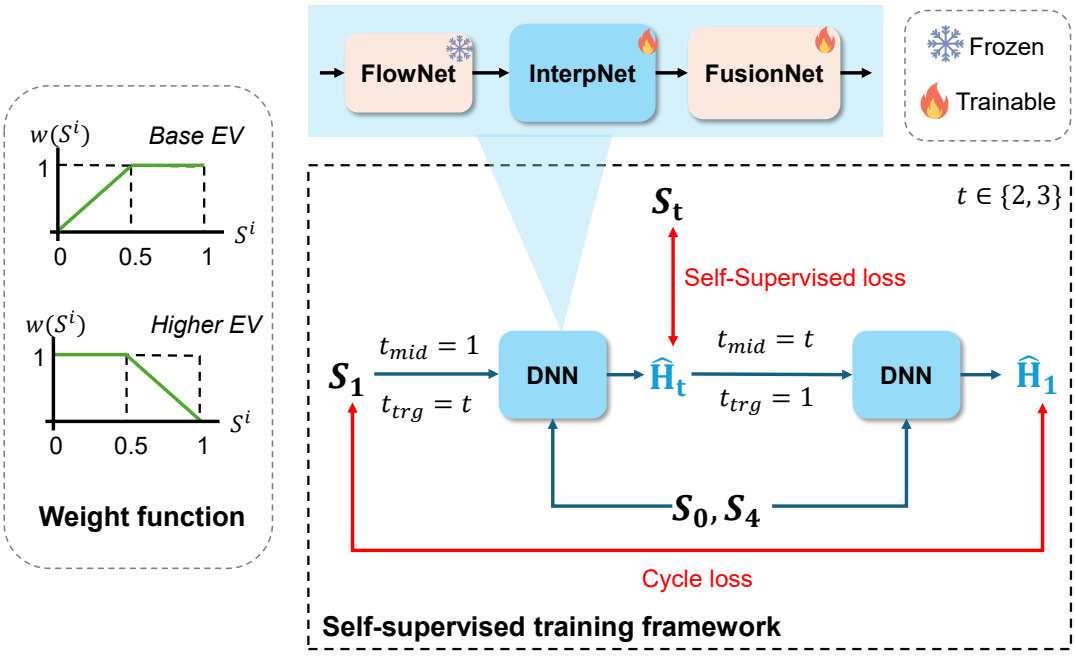


Figure 3.3: Overview of our proposed self-supervised training framework. (Left) The weight function used in the self-supervised reconstruction loss, designed to downweight saturated or unreliable regions and focus learning on well-exposed areas. (Right) The full training pipeline, where the model reconstructs intermediate HDR frames without access to ground-truth supervision. The cycle-consistency loss enforces temporal coherence, while the weighted reconstruction loss guides learning from valid regions of the input.

full exposure set were available at timestep t, we could train a network to minimize the reconstruction error against H_t^{fusion} at t over a training set D as follows:

(3.6)

Using the triangle inequality and the assumption $\sum_i w(S_t^i) \geq 1$, we derive the following upper bound:

$$\begin{aligned}
\left\| \hat{H}_t - H_t^{\text{fusion}} \right\|_1 &= \left\| \hat{H}_t - \sum_i \bar{w}(S_t^i) \cdot l_t^i \right\|_1 \\
&\leq \sum_i \bar{w}(S_t^i) \cdot \left\| \hat{H}_t - l_t^i \right\|_1 \quad \text{(triangle inequality)} \\
&\leq \sum_i w(S_t^i) \cdot \left\| \hat{H}_t - l_t^i \right\|_1 \quad \text{(since } \sum_i w(S_t^i) \geq 1\text{)}, \qquad (3.7)
\end{aligned}$$

where $\bar{w}(S_t^i) = w(S_t^i)/\sum_j w(S_t^j)$ denotes the normalized weight. Minimizing the right-hand side is thus equivalent to minimizing an upper bound of the true HDR fusion loss. In practice, training a network with parameters θ and batch size B over N exposure samples requires computing the full gradient across all available exposures:

$$\nabla_\theta \mathcal{L} = \mathbb{E}_{(L_t^i, e_t^i)} \left[w(S_t^i) \cdot \nabla_\theta \left\| \hat{H}_t - l_t^i \right\|_1 \right]. \tag{3.8}$$

However, in the self-supervised setting, only a single SDR input S_t is available per timestep. We therefore approximate the expectation using this single sample:

$$\nabla_\theta \tilde{\mathcal{L}} = w(L_t) \cdot \nabla_\theta \left\| \hat{H}_t - l_t \right\|_1. \tag{3.9}$$

This can be viewed as an unbiased estimator of the full gradient when exposures are sampled uniformly across the available set. Combined with a stochastic optimizer such as Stochastic Gradient Descent or Adam [63], this approximation enables effective training without requiring access to multiple exposures per timestep.

Approximating the gradient using Eq. 3.9 is equivalent to training our network with the following loss function:

$$w(S_t) \cdot \left\| \hat{H}_t - l_t \right\|_1. \tag{3.10}$$

In our implementation, we adopt the per-pixel exposure-aware weighting function $w(\cdot)$ introduced by Kalantari *et al.* [34], visualized in Figure 3.1

$$w(S^i) = \begin{cases} \min(2 \cdot S^i, 1), & \text{if } S^i \text{ is the frame with base exposure value} \\ \min(2 \cdot (1 - S^i), 1), & \text{if } S^i \text{ is the frame with higher exposure value} \end{cases} \tag{3.11}$$

This function downweights unreliable regions—such as shadows in underexposed frames and highlights in overexposed frames—while assigning higher weights to mid-tone pixels. The function is designed to satisfy the condition:

$$\sum_i w(S^i) \geq 1 \quad \text{for all pixel locations,} \tag{3.12}$$

under the assumption that the exposure stack provides complementary information—namely, each pixel is guaranteed to fall within a well-exposed region in at least one of

the input frames, either due to preserved shadows in the higher exposure or preserved highlights in the lower exposure.

In addition, to better reflect perceptual differences, we apply Eq. 3.10 in the tone-mapped domain. This results in the following self-supervised reconstruction loss:

$$\mathcal{L}_{self\_recon} = w(S_t) \cdot \left\| \mathcal{T}(\hat{H}_t) - \mathcal{T}(l_t) \right\|_1 \tag{3.13}$$

A potential degenerate solution to minimizing Eq. 3.10 is for the network to mimic the input's alternating exposure pattern. Our design prevents this: the fusion network, which generates the final HDR output, has no access to the target time t or exposure metadata. Although t is used in flow estimation, the fusion stage lacks temporal cues and cannot directly reproduce the exposure sequence.

Finally, we introduce a cycle consistency loss to enhance temporal coherence in the absence of supervision. Given a predicted HDR frame $\hat{H}_t$, we use it along with t_0 and t_2 to re-predict the central frame at t_1, enforcing:

$$\mathcal{L}_{self} = \mathcal{L}_{self\_recon} + \lambda_{cycle}\mathcal{L}_{cycle}, \tag{3.14}$$

where $\mathcal{L}_{cycle}$ is the $\mathcal{L}_{self\_recon}$ loss applied to the cyclically reconstructed frame, and λ_{cycle} balances its contribution.

Moreover, to ensure balanced supervision, we adopt a sampling strategy that encourages uniform exposure diversity within each mini-batch. For example, consider a 3-frame input consisting of frame 0 (EV 0), frame 1 (EV +3), and frame 4 (EV 0). We include frame 2 (EV 0) and frame 3 (EV +3) as supervision targets within the same mini-batch. This ensures that lower- and higher-exposure frames are evenly represented during training, allowing the model to reconstruct HDR content across the available exposure range.

Chapter 4

Experiments and Custom Dataset

This chapter presents a comprehensive evaluation of our proposed method. We begin by introducing our custom HDR video dataset, detailing its construction process and comparing it against existing benchmarks. We then describe the evaluation metrics used to quantify reconstruction and interpolation quality. Subsequently, we present experimental results, including comparisons with state-of-the-art baselines, qualitative assessments on our dataset and real-world unlabeled data, and runtime analysis. Finally, we conduct an ablation study to examine the impact of key design components and conclude with a discussion of failure cases that highlight current limitations. Together, these experiments validate the effectiveness, efficiency, and generalizability of our approach.

4.1 Dataset

As discussed in Chapter 2, existing datasets are insufficient for training and evaluating HDR video reconstruction and interpolation under alternating exposures. Most available datasets either lack the temporal information for each frame in the sequence [9, 11] or are recorded at low frame rate settings [64, 65]. To address this gap, we utilize a raw HDR video dataset captured by our fellow student, Trevor D. Canham, as part of a research project at York University. This dataset comprises 95 HDR video sequences totaling 32,427 frames, recorded at 60 FPS using a Sony FX6 camera in 12-bit ProRes RAW format (see [66]). It encompasses various indoor and outdoor scenes under varying lighting conditions, providing a rich foundation for both reconstruction and interpolation

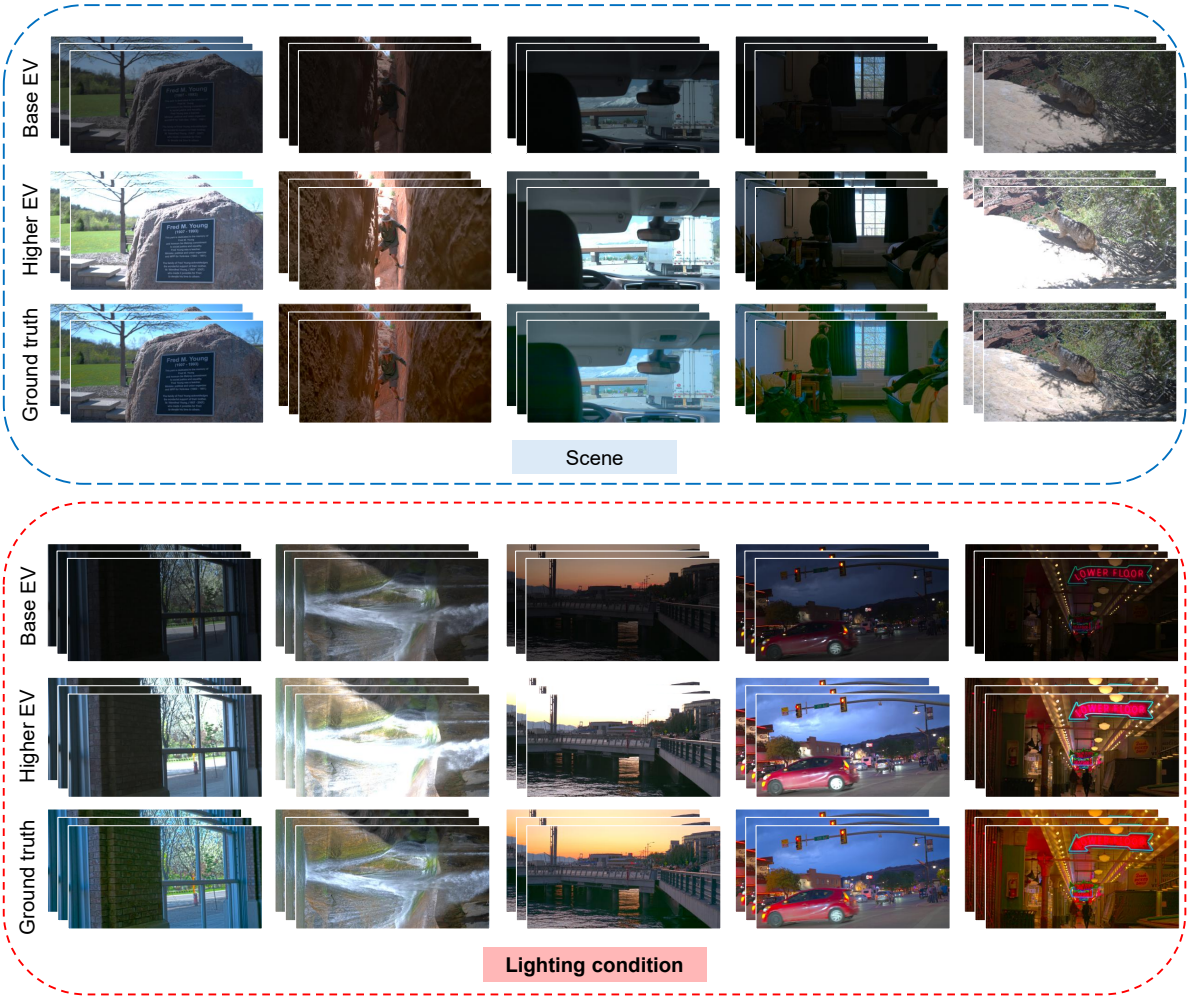


Figure 4.1: Examples from our synthesized dataset showcasing scene and lighting conditions diversity.

tasks. Representative samples from the dataset are shown in Fig. 4.1.

Ground Truth Generation. We first process the RAW video footage into linear HDR radiance maps to make the dataset suitable for supervised learning. This process is done using Adobe Premiere Pro, exporting each frame in the OpenEXR format. These radiance maps serve as the ground truth during supervised training. For scenes with challenging low-light conditions, we apply video denoising using the Neat Video plugin[1], which ef-

[1] `https://www.neatvideo.com/`

fectively reduces sensor noise while preserving high-frequency image details critical for learning fine-grained HDR features.

Input Generation. To generate input sequences simulating alternating exposure settings, we applied synthetic exposure adjustments to each scene. Specifically, two distinct exposure levels were selected per scene—for example, EV0 and EV+3 or EV+4—to ensure a wide dynamic range. These adjustments are made in Adobe Premiere Pro using the *Exposure Control* tool. Notably, only the high-exposure frames are denoised, preserving the realistic noise patterns in the low-exposure frames. After tone mapping and color conversion, the processed frames are exported as 8-bit PNGs in the BT.709 color space. We interleave the exposure streams and apply temporal subsampling to create low-frame-rate SDR sequences suitable for interpolation tasks: 30 FPS sequences for 2× interpolation and 15 FPS sequences for 4× interpolation.

Data Split. The dataset is split into 60 training scenes, seven validation scenes, and 28 test scenes for training and evaluation purposes. All frames are resized to a fixed resolution of 1376×730 pixels to accommodate GPU memory constraints.

Dataset Analysis. To assess the suitability of our dataset for both HDR reconstruction and interpolation, we compare its content diversity and HDR characteristics against two prior HDR datasets: Chen21 [9] and Real-HDRV [11]. Unlike our dataset, both Chen21 and Real-HDRV are captured using stop-motion setups, making them unsuitable for benchmarking temporal interpolation due to the absence of motion continuity. However, they serve as strong references for HDR reconstruction benchmarks.

Following the evaluation protocol in previous works [67, 68, 11], we quantify the diversity of our dataset using seven established metrics, grouped into three categories that assess different aspects of HDR video content.

(1) Extent of HDR:

- *Fraction of Highlight Pixels (FHLP)* measures the proportion of pixels whose normalized luminance exceeds the SDR's peak luminance (i.e., $100nit$), reflecting the prevalence of saturated bright regions in the image [67].

- *Extent of Highlight (EHL)* quantifies how broadly and intensely highlights are distributed in the luminance range. Specifically, it is computed as the average pixel distance between the luminance of HDR and its clip-to-100*nit* version [67].

$$\text{EHL} = \frac{1}{n}\sum_{i=1}^{n}\sqrt{[Y_i - clip(Y_i)]^2} \quad \text{with clip}(x) = clamp(x, 0, 0.01) \tag{4.1}$$

(2) Intra-frame Diversity:

- *Spatial Information (SI)* is the standard deviation of the Sobel-filtered images [69]. Higher SI values indicate a richer high-frequency pattern.

- *Colorfulness (CF)* reflects the perceptual richness of color, using the standard metric proposed by Hasler and Süsstrunk [70]. It is calculated as

$$\text{CF} = \sqrt{\sigma_{rg}^2 + \sigma_{yb}^2} + 0.3 \times \sqrt{\mu_{rg}^2 + \mu_{yb}^2}, \tag{4.2}$$

  where $rg = R - G$, $yb = \frac{R+G}{2} - B$, and μ, σ denote the mean and standard deviation, respectively.

- *Standard Deviation of Luminance (stdL)* quantifies the luminance contrast within a frame by measuring the standard deviation of all pixel luminance values. A higher stdL indicates greater variation in brightness, often associated with more visually dynamic scenes [67].

(3) Overall Scene Style:

- *Average Luminance Level (ALL)* reports the average of normalized luminance $Y = 0.2627R + 0.6780G + 0.0593B$ [67].

- *Dynamic Range (DR)* measures the overall luminance range of the scene by calculating the base-10 logarithmic ratio between the 98th percentile and 2nd percentile luminance values [68]:

$$\text{DR} = \log_{10}\left(\frac{P_{98}}{P_2}\right), \tag{4.3}$$

  where P_{98} and P_2 represent the luminance values at the top and bottom 2% of the pixel distribution, respectively.

Dataset	Extent of HDR		Intra-frame Diversity			Overall-style	
	FHLP	EHL	SI	CF	stdL	ALL	DR
Chen21 [9]	9.89	2.85	10.81	4.31	9.27	4.74	2.28
Real-HDRV [11]	13.41	2.84	**15.06**	5.26	**10.12**	**5.64**	2.37
Ours	**14.48**	**2.98**	13.17	**7.46**	8.55	5.41	**2.44**

Table 4.1: Quantitative analysis of HDR diversity across different benchmark datasets. All values except DR are percentages. Our dataset demonstrates comparable HDR range and intra-frame diversity to prior benchmarks while uniquely supporting video interpolation.

Table 4.1 summarizes the statistics. Overall, the results demonstrate that our dataset achieves comparable—and in some cases superior—diversity characteristics while being uniquely suited for interpolation tasks. More specifically, our dataset achieves the highest FHLP (14.48%) and EHL (2.98%), indicating a broader and more saturated distribution of highlight regions compared to prior benchmarks. Our dataset also records the highest colorfulness (CF = 7.46%), reflecting richer chromatic diversity. While its spatial information (SI) and average luminance (ALL) remain competitive, its dynamic range (DR = 2.44) exceeds that of these two datasets. These results confirm that our dataset offers superior HDR characteristics. Combined with its unique suitability for interpolation under alternating exposures, this makes it well-suited for the experiments that follow.

4.2 Metrics

We evaluate the quality of the reconstructed HDR frames using both traditional pixel-wise metrics and learning-based perceptual metrics. This dual evaluation captures both low-level fidelity and perceptual quality.

Pixel-wise Metrics. We evaluate reconstruction fidelity using four variants of PSNR and SSIM tailored for HDR data: PU-PSNR, PU-SSIM, PSNR-μ, and SSIM-μ. PU-PSNR and PU-SSIM apply standard PSNR and SSIM formulas to HDR frames after perceptual uniform (PU) encoding, which transforms raw radiance values to a perceptually linear space that better models realistic coding artefacts and accounts for glare [71]. In

Model	PU-tone mapping		μ-tone mapping		Q^*_{LPIPS} ↓
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	
HDRFlow [12] (Trivial Copy)	24.60	0.7630	30.62	0.8463	0.1522
M2M-VFI [49]+ HDRFlow	28.90	0.8726	34.90	0.9190	0.1163
EMA-VFI [56] + HDRFlow	29.79	<u>0.8934</u>	<u>35.74</u>	<u>0.9334</u>	0.1164
DeepHS-HDR [13] (EMA-VFI)	<u>29.83</u>	**0.9081**	35.53	**0.9414**	0.1266
DeepHS-HDR [13] (M2M-VFI)	29.16	0.8844	34.91	0.9259	0.1274
Ours	**29.88**	0.8927	**35.77**	0.9326	**0.1010**
Ours (Self-supervise)	29.29	0.8835	35.23	0.9266	<u>0.1104</u>

Table 4.2: Results on our testing set with 2× interpolation. The best score in each column is **bolded**, while the second best is <u>underlined</u>. Our supervised model consistently matches or outperforms existing baselines across all metrics, while our self-supervised variant also demonstrates competitive performance, validating the effectiveness of our approach.

contrast, PSNR-μ and SSIM-μ operate on HDR images tone-mapped using the logarithmic μ-law operator, which compresses high dynamic range luminance values while preserving structural information. Across all these metrics, higher values indicate better agreement between the reconstructed and ground-truth HDR frames.

Learning-based Metric. To assess perceptual quality beyond pixel-wise similarity, we use Q^*_{LPIPS} [72], a deep learning-based HDR image quality assessment metric. This metric extends the original LPIPS [73] by representing HDR images as a stack of SDR images with varying exposures. Unlike traditional metrics, Q^*_{LPIPS} compares high-level feature representations extracted from a neural network to better reflect human visual judgments. Lower values indicate higher perceptual similarity to the ground truth.

4.3 Experiments

Implementation details. We train our network on an NVIDIA L40 GPU using the Adam optimizer for a total of 40 epochs. The learning rate is set to 0.001 with a 3-epoch warm-up, then decreased to 0.0001 and halved at epochs 20 and 30. We use a batch size

Model	PU-tone mapping		μ-tone mapping		$Q^*_{LPIPS}\downarrow$
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	
HDRFlow [12] (Trivial Copy)	23.51	0.7244	29.58	0.8184	0.16714
M2M-VFI [49]+ HDRFlow	25.45	0.7944	31.56	0.8663	0.1591
EMA-VFI [56] + HDRFlow	26.64	0.8334	32.71	0.8938	0.1590
DeepHS-HDR [13] (EMA-VFI)	<u>26.89</u>	**0.8545**	<u>32.80</u>	**0.9069**	0.1702
DeepHS-HDR [13] (M2M-VFI)	25.76	0.8133	31.72	0.8798	0.1734
Ours	**27.18**	<u>0.8385</u>	**33.21**	<u>0.8979</u>	**0.1386**
Ours (Self-supervise)	26.32	0.8176	32.40	0.8839	<u>0.1520</u>

Table 4.3: Results on our testing set with 4× interpolation. The results follow a similar trend to the 2× case: our supervised model achieves state-of-the-art performance in PSNR and Q^*_{LPIPS} scores, while the self-supervised variant remains highly competitive.

of 16 for supervised training and eight for self-supervised training. During training, SDR frame sequences are randomly cropped to 512×512; for self-supervised training, the input crops are augmented with random rotations and flips. In self-supervised training, λ_{cycle} is set to 0.1.

4.3.1 Comparison with Baselines

We compare our method against two groups of baselines. The first combines state-of-the-art video interpolation models (M2M-VFI [49] and EMA-VFI [56]) with an HDR video reconstruction method (HDRFlow [12]): the target frame at time t is first interpolated, then used with its neighbors as input to the HDR reconstruction network. The second group follows the DeepHS-HDR pipeline [13], combined with various interpolation models (M2M-VFI [49] and EMA-VFI [56]). As a lower bound, we include a trivial copy baseline that simply uses the nearest HDR output to approximate the frame at time t. All baselines are fine-tuned on our dataset using the authors' recommended settings. For DeepHS-HDR, we re-implemented the method based on the paper, as no official code was available.

Quantitative results. Tables 4.2 and 4.3 report results on our test set for 2× (30→60 fps) and 4× (15→60 fps) video interpolation. Our supervised model consistently matches

or outperforms baselines across all metrics. Notably, our model improves perceptual quality significantly, achieving the lowest Q^*_{LPIPS} scores in both interpolation settings, indicating more perceptually accurate reconstructions.

Our self-supervised variant also demonstrates strong performance. Despite not leveraging paired HDR ground truth, it remarkably surpasses several supervised baselines, including M2M-VFI + HDRFlow and DeepHS-HDR (M2M-VFI), across most metrics in both 2× and 4× interpolation scenarios. For example, in 2× interpolation, our self-supervised model achieves a Q^*_{LPIPS} of 0.1104, outperforming M2M-VFI + HDRFlow (0.1163) and DeepHS-HDR models (0.1266, 0.1274). This strong performance without reliance on paired HDR data highlights the significant practical applicability and generalizability of our self-supervised approach.

Qualitative results. Figure 4.2 and 4.3 present visual comparisons between our method and existing baselines on the test set. Our method consistently produces sharper images with greater detail, particularly in regions with large or fast motion. We attribute this improvement to our single-stage pipeline, which jointly performs HDR reconstruction and interpolation in a single pass. In contrast, two-stage baselines typically interpolate between same-exposure frame pairs (*e.g.*, EV0–EV0 or EV4–EV4), disregarding temporally adjacent but differently exposed frames. This leads to a reduced effective temporal resolution and poor motion estimation in dynamic scenes. As a result, these methods often suffer from motion blur or ghosting artifacts in challenging regions with fast motion.

Evaluation on Unlabeled Real-World Dataset. To assess the generalization capability of our model in real-world settings, we conduct qualitative evaluations on the Chen21 dataset [9], which contains HDR videos captured under diverse lighting conditions but lacks ground-truth intermediate frames for supervision. As shown in Figure 4.4, our method performs better than the baselines even in this unlabeled setting. Consistent with the trends observed on our benchmark dataset, competing methods often exhibit noticeable artifacts—such as motion blur in dynamic regions (*e.g.*, the girl's moving hand) and ghosting near humans or object boundaries—due to inaccurate motion estimation or misalignment between exposures. In contrast, our method produces sharper and more temporally coherent results by jointly modeling motion and exposure variation.

Model	#Params	2× interpolation		4× interpolation	
		FLOPs	RT (ms)	FLOPs	RT (ms)
HDRFlow [12] (Trivial Copy)	3.27M	0.178T	24.14	0.178T	24.14
M2M-VFI [49]+ HDRFlow	10.88M	<u>0.506T</u>	183.08	<u>0.862T</u>	<u>245.23</u>
EMA-VFI [56] + HDRFlow	17.76M	0.592T	<u>99.08</u>	1.551T	246.96
DeepHS-HDR [13] (EMA-VFI)	15.97M	4.047T	417.61	11.70T	1145
DeepHS-HDR [13] (M2M-VFI)	<u>9.09M</u>	3.385T	583.81	8.841T	1142
Ours	**3.30M**	**0.181T**	**28.25**	**0.398T**	**66.14**

Table 4.4: Complexity comparison of HDR interpolation models for 2× and 4× interpolation. We report the number of parameters (#Params), floating point operations (FLOPs), and runtime (RT) at 1280×720 resolution on an RTX 4070 Super. Our model achieves the best balance between efficiency and quality, requiring the fewest parameters and significantly lower FLOPs compared to all other HDR interpolation pipelines.

Complexity Comparisons. Table 4.4 compares model complexity in terms of parameter count, floating point operations (FLOPs), and runtime across all evaluated methods. All runtime measurements are conducted on an RTX 4070 Super GPU at 1280×720 resolution. Our model achieves the best balance between efficiency and quality, requiring only 3.30M parameters and significantly fewer FLOPs than all other HDR interpolation pipelines. Specifically, it operates at 0.181T and 0.398T FLOPs for 2× and 4× interpolation, with runtimes of 28.25ms and 66.14ms, respectively.

Notably, compared to HDRFlow [12], which does not perform interpolation but simply duplicates frames ("Trivial Copy"), our model introduces minimal additional overhead. HDRFlow alone requires 0.178T FLOPs and 24.14ms per frame—virtually identical to our 2× setup—demonstrating that the cost of adding interpolation capability is negligible. Our method adds less than 5% more FLOPs and 4ms runtime with additional capability of interpolation.

In contrast, prior methods such as M2M-VFI + HDRFlow and EMA-VFI + HDRFlow introduce a substantial computational burden, with runtimes exceeding 240ms and FLOPs reaching up to 1.56T. DeepHS-HDR variants are even more demanding, with over 3× the parameters and more than 10× the runtime of our model. These results highlight the

Method	$\text{PSNR}_{\text{PU}} \uparrow$	$\text{SSIM}_{\text{PU}} \uparrow$	$\text{PSNR}_{\mu} \uparrow$	$\text{SSIM}_{\mu} \uparrow$	$Q^*_{LPIPS} \downarrow$
Ours (Self-supervise)	**29.29**	**0.8835**	**35.23**	**0.9266**	**0.1104**
W/o cycle loss	28.36	0.8656	34.35	0.9152	0.1218
W/o FlowNet features	<u>29.05</u>	<u>0.8793</u>	<u>35.00</u>	<u>0.9237</u>	<u>0.1108</u>
Using implicit function of t	28.54	0.8701	34.52	0.9183	0.1206

Table 4.5: Ablation study on the impact of loss design and interpolation architecture choices.

efficiency of our unified design: it delivers high-quality HDR video interpolation at a fraction of the computational cost and with minimal overhead—even compared to non-interpolating baselines—making it suitable for practical applications.

4.3.2 Ablation Study

We conduct a series of ablation experiments to assess the impact of key components in our framework. Quantitative results are summarized in Table 4.5.

Loss Function. We first evaluate the effect of the self-supervised reconstruction loss and the cycle-consistency regularization. When trained solely with the self-supervised reconstruction loss, our model already achieves strong performance across PSNR, SSIM, and perceptual metrics, underscoring the effectiveness of our weighted log-domain formulation. Introducing the cycle-consistency loss further improves performance, particularly in perceptual quality (LPIPS), by encouraging temporal consistency and helping regularize reconstruction in poorly exposed or ambiguous regions. As shown in Figure 4.5, incorporating cycle consistency reduces ghosting artifacts in regions affected by large motion and detail loss due to clipping.

Architecture Design. We also evaluate the influence of two architectural design choices. First, removing the reuse of FlowNet features within InterpNet leads to a noticeable drop in accuracy, suggesting that motion-aware context from FlowNet plays a critical role in improving latent flow interpolation. Second, we test a variant where the target time t is implicitly injected by concatenating it as a scalar feature to the input tensor, replacing

our structured quadratic basis representation. This model performs significantly worse, confirming that our explicit, temporally-structured interpolation mechanism provides a stronger inductive bias, especially important in self-supervised settings where no ground truth annotations are available to guide temporal mapping. As illustrated in Figure 4.6, the implicit encoding model produces visible ghosting around thin structures like tree branches and introduces distortion in static regions, such as the warping of the window frame. These artifacts indicate the model's inability to accurately resolve motion patterns and align images.

4.4 Failure Cases

While our method demonstrates strong performance across both quantitative metrics and visual quality, it still encounters several failure modes in challenging scenarios. Many of these issues are not unique to our approach but are shared by prior work, highlighting persistent limitations in current HDR video reconstruction and interpolation pipelines. We group these challenges into two main categories: (1) general failure cases commonly observed in standard video interpolation tasks, and (2) issues specific to HDR reconstruction from alternating-exposure sequences. For each category, we also discuss potential strategies that could help address these limitations.

4.4.1 Failures Common in Video Interpolation

Inaccurate Flow Estimation at Depth Boundaries. One major challenge is accurate motion estimation around depth discontinuities, such as the boundaries between foreground and background objects. These areas often exhibit complex occlusion relationships, making optical flow estimation unreliable. As a result, the warped features become misaligned, leading to ghosting or geometric artifacts, as shown in the top row of Figure 4.7. This issue is common across all evaluated methods, including ours. Incorporating explicit occlusion handling could improve robustness in such scenarios.

Occlusion and Disocclusion Handling. A second persistent challenge involves handling occluded or newly revealed (disoccluded) regions. Because these areas lack valid

correspondences between input frames, pixel-level fusion alone cannot faithfully recover missing details. As illustrated in Figure 4.7, our method, like other baselines, struggles with hallucinating plausible content in these regions. Post-fusion refinement networks, visibility prediction, or high-level semantic guidance could serve as promising directions to enhance inpainting and ensure temporal consistency.

4.4.2 Failures Specific to Alternating-Exposure HDR Reconstruction

Fusion under Severe Noise in Both Exposures. In extremely low-light conditions, both short- and long-exposure frames may suffer from significant noise, degrading the reliability of fusion. As shown in Figure 4.8, our method can fail to produce an output cleaner than the long-exposure input alone. This is primarily due to the limitations of our pixel-level fusion strategy, which cannot suppress noise. Future work may benefit from a dedicated denoising stage or noise-aware fusion mechanisms.

Information Loss due to Saturated Motion. Another critical limitation arises when fast-moving objects appear saturated in one or both input frames, particularly in the higher-exposure image. In such cases, essential radiance information is irretrievably lost, which not only hampers HDR reconstruction but also degrades motion estimation due to the absence of meaningful texture. This often results in motion blur or visible ghosting artifacts, as demonstrated in Figure 4.9. More robust saturation-aware modeling or predictive reconstruction techniques may help alleviate this issue.

Figure 4.2: Visual comparison of HDR frame interpolation results on 2 × interpolation task. We abbreviate M2M-VFI + HDRFlow as M2M-HDRF and DeepHS-HDR + EMA-VFI as DeepHS-EMA. The "Input Overlay" shows the average of input frames in the linear irradiance domain. All linear outputs are tone-mapped using Photoshop's HDR tone mapping for display purposes.

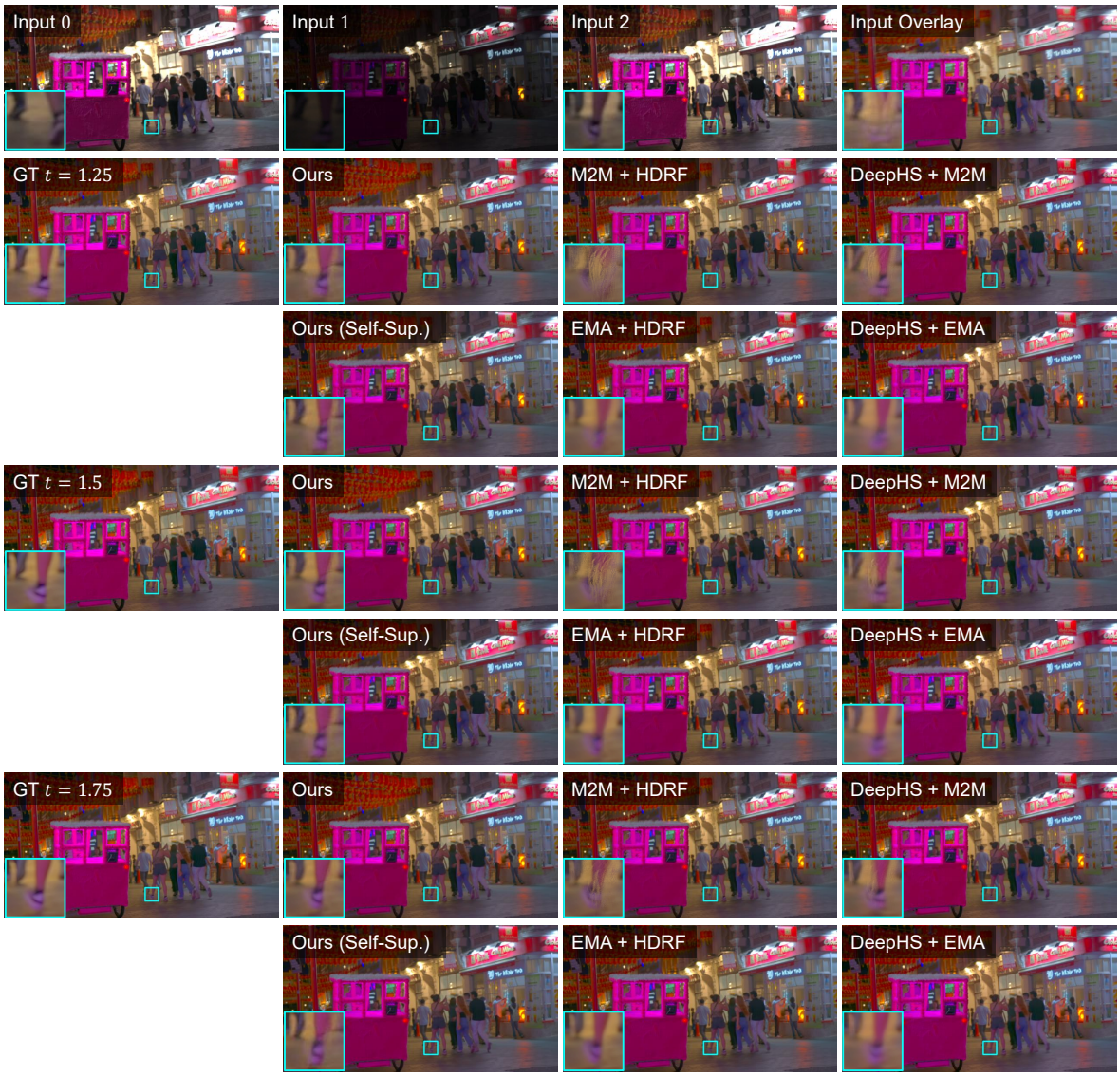


Figure 4.3: Visual comparison of HDR frame interpolation results on 4 × interpolation task.

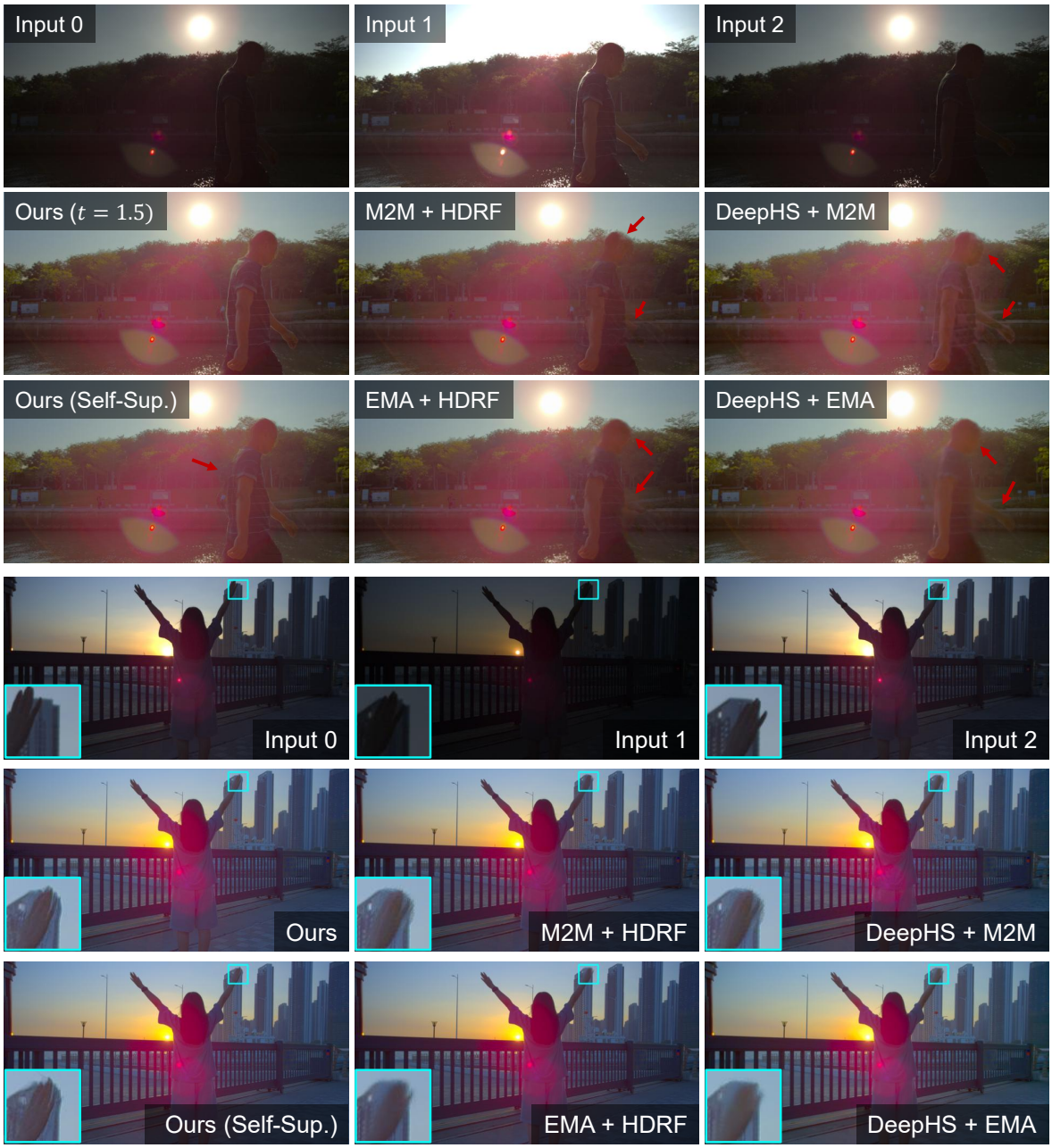


Figure 4.4: Visual comparison of HDR frame interpolation results on an unlabeled real-world dataset with 2 × interpolation task.

Figure 4.5: Ablation study on cycle consistency loss. Using the cycle loss helps reduce ghosting and enhances detail retention in challenging regions, particularly those affected by large motion and exposure clipping. The zoomed-in areas highlight these improvements.

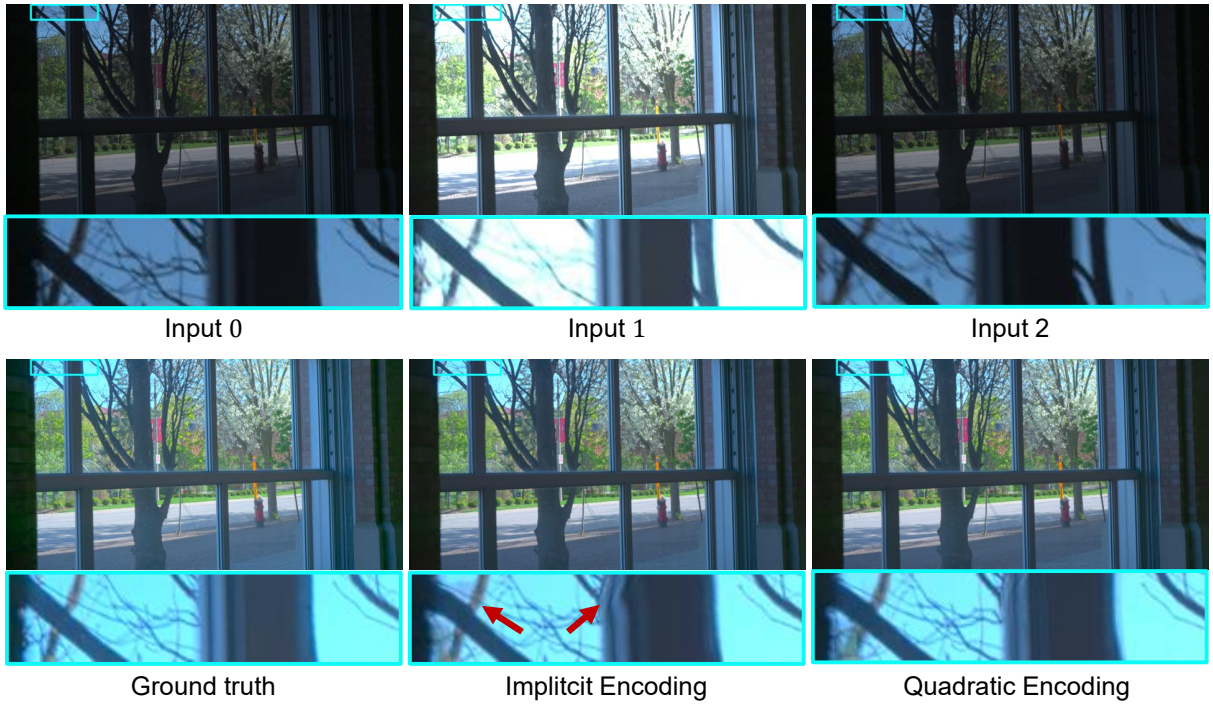


Figure 4.6: Ablation study on the flow encoding function. Using an implicit encoding of t results in noticeable ghosting near fine structures and slight distortions near the edges. These effects point to challenges in resolving motion and maintaining spatial consistency without an explicit temporal representation.

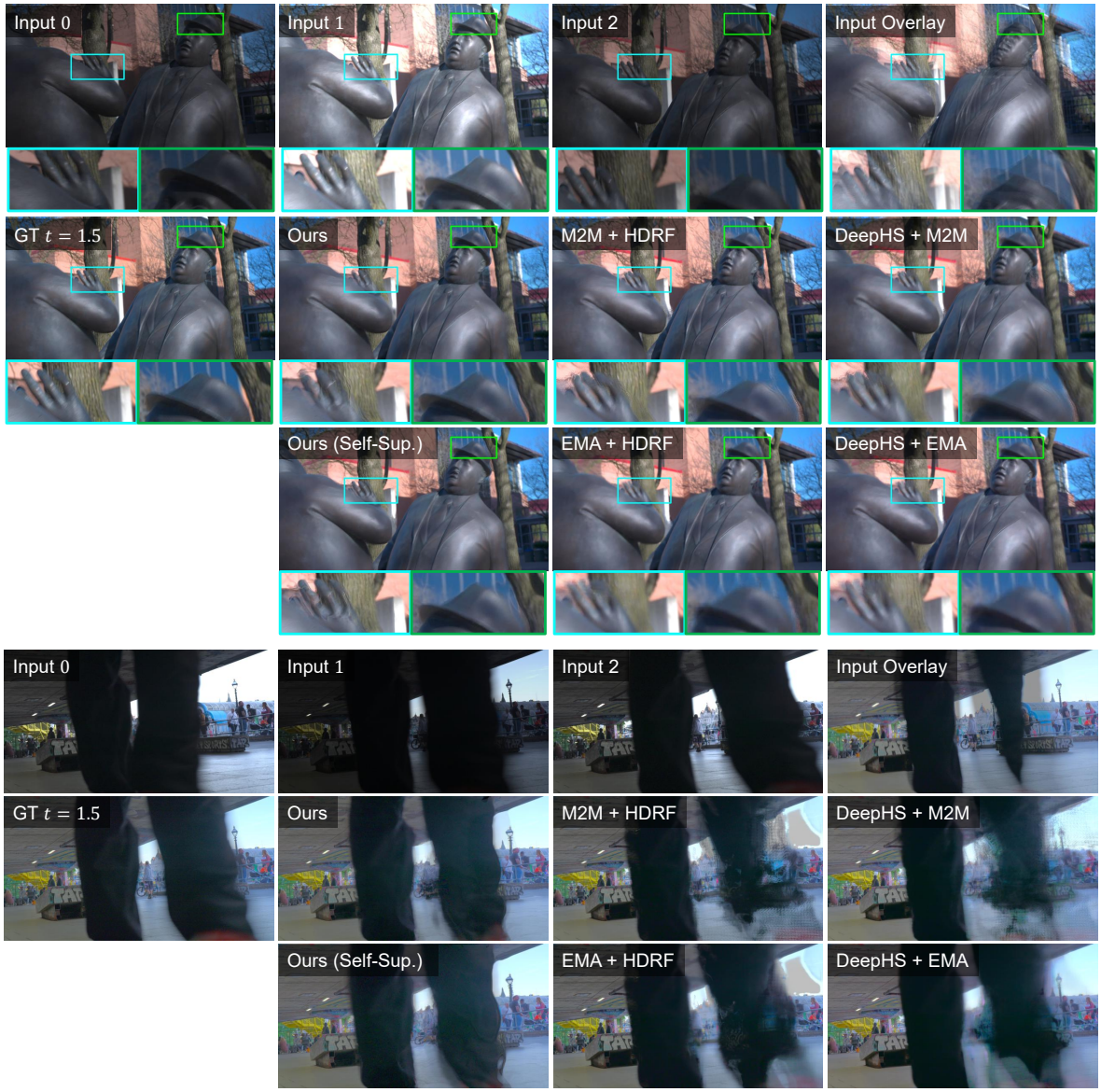


Figure 4.7: Failure cases common in video frame interpolation. We present two representative examples. The first row highlights inaccurate flow estimation at depth boundaries, where ghosting and geometric artifacts become apparent, especially in the zoomed-in regions. The second row illustrates failure in handling occlusions, where disoccluded areas are incorrectly synthesized, resulting in unrealistic content.

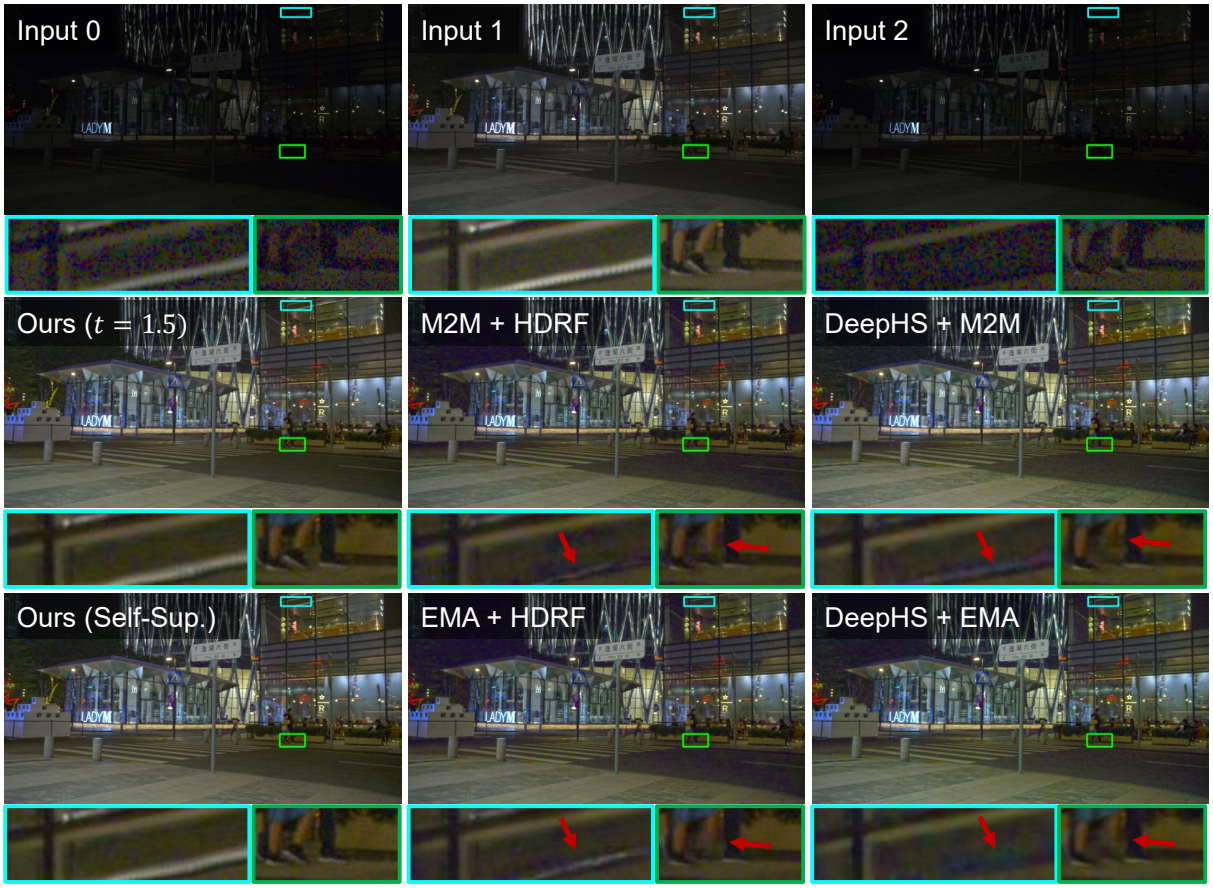


Figure 4.8: Examples of failure cases in real-world data with noise present. For the crop of input images, the brightness has been increased by 40% to better highlight the noise.

Figure 4.9: Examples of failure cases when the moving objects are saturated.

Chapter 5

Summary and Future Work

5.1 Summary

In this thesis, we proposed a novel unified framework for high-frame-rate HDR video synthesis from exposure-bracketed sequences, jointly addressing the tasks of HDR reconstruction and temporal interpolation in a single end-to-end trainable model. Our approach leverages both the temporal redundancy and exposure diversity inherent in alternating exposure video to produce temporally consistent HDR frames at arbitrary time steps.

Key to our design is a lightweight interpolation module based on structured quadratic motion modeling, which enables high temporal fidelity while remaining computationally efficient. To mitigate the need for large-scale ground-truth HDR video data, we further proposed a self-supervised training framework that enables learning from unlabeled sequences.

Through extensive quantitative and qualitative experiments, we demonstrated that our method achieves state-of-the-art performance with significantly reduced computational overhead, making real-time HDR video synthesis feasible even on mid-range GPUs. We hope that our contributions serve as a foundation for further research in high-quality HDR video generation on commodity camera systems.

5.2 Future Research Directions

While our method achieves strong results, several opportunities remain to further improve its robustness, generalization, and real-world applicability. We outline three suggested directions for future research: enhancing model resilience to challenging cases, expanding evaluation on real-world data, and enabling broader deployment in practical scenarios.

Improving Network Robustness and Training Strategies. As discussed in Section 4.4, our method still struggles in complex scenarios such as occluded regions, saturated motion, and scenes with significant noise in both exposures. These limitations suggest the need for more robust handling of ambiguous or degraded inputs. Future efforts could focus on introducing refinement modules to enhance reconstruction in these challenging areas or implementing strategies to explicitly address occlusions. Additionally, adjusting training methods—such as reweighting losses, using confidence-aware objectives, or enforcing temporal consistency—may improve the model's ability to manage edge cases and to generalize better to unseen conditions.

Enhancing Real-World Dataset Evaluation. Another key limitation lies in the scarcity of real-world HDR datasets with ground-truth intermediate frames. Our current evaluation primarily relies on synthetic data, which may not fully capture the complexity of natural scenes. Building or leveraging real-world datasets with accurate HDR annotations and dense temporal labeling would support more rigorous benchmarking and guide the development of models that perform reliably in diverse, uncontrolled environments. This step is essential for transitioning from academic evaluation to practical use.

Toward Practical Deployment. Finally, to enable real-world deployment, especially on resource-constrained devices, further efficiency improvements are crucial. While our method is already more lightweight than previous HDR pipelines, additional optimizations—such as model compression, quantization-aware training, or hardware-specific acceleration—could enhance its suitability for edge devices like smartphones and embedded cameras.

References

[1] Karol Myszkowski, Rafal Mantiuk, and Grzegorz Krawczyk, *High dynamic range video*, vol. 5, Morgan & Claypool Publishers, 2008.

[2] OMNIVISION OG05C product guide, "ovt.com," `https://www.ovt.com/wp-content/uploads/2024/10/OG05C-PB-v1.0-WEB.pdf`, [Accessed 07-06-2025].

[3] "Sony's cmos image sensor for automotive," `https://www.sony-semicon.com/files/62/pdf/p-15_IMX490.pdf`, [Accessed 07-06-2025].

[4] Ales Leonardis Richard Shaw, Sibi Catley-Chandar and Eduardo Pérez-Pellitero, "HDR reconstruction from bracketed exposures and events," *BMVC*, 2022.

[5] Yixin Yang, Jin Han, Jinxiu Liang, Imari Sato, and Boxin Shi, "Learning event guided high dynamic range video reconstruction," *CVPR*, 2023.

[6] Yakun Chang, Chu Zhou, Yuchen Hong, Liwen Hu, Chao Xu, Tiejun Huang, and Boxin Shi, "1000 FPS HDR video with a spike-rgb hybrid camera," in *CVPR*, 2023.

[7] Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski, "High dynamic range video," *ACM TOG*, 2003.

[8] Jun Hu, Orazio Gallo, Kari Pulli, and Xiaobai Sun, "HDR deghosting: How to deal with saturation?," in *CVPR*, 2013.

[9] Guanying Chen, Chaofeng Chen, Shi Guo, Zhetong Liang, Kwan-Yee K. Wong, and Lei Zhang, "HDR video reconstruction: A coarse-to-fine network and A real-world benchmark dataset," in *ICCV*, 2021.

[10] Haesoo Chung and Nam Ik Cho, "LAN-HDR: Luminance-based alignment network for high dynamic range video reconstruction," in *ICCV*, 2023.

[11] Yong Shu, Liquan Shen, Xiangyu Hu, Mengyao Li, and Zihao Zhou, "Towards real-world HDR video reconstruction: A large-scale benchmark dataset and A two-stage alignment network," in *CVPR*, 2024.

[12] Gangwei Xu, Yujin Wang, Jinwei Gu, Tianfan Xue, and Xin Yang, "HDRFlow: Real-time HDR video reconstruction with large motions," in *CVPR*, 2024.

[13] Zeeshan Khan, Parth Shettiwar, Mukul Khanna, and Shanmuganathan Raman, "DeepHS-HDRVideo: Deep high speed high dynamic range video reconstruction," in *ICPR*, 2022.

[14] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy, "Burst photography for high dynamic range and low-light imaging on mobile cameras," *ACM TOG*, 2016.

[15] Yafei Ou, Prasoon Ambalathankandy, Shinya Takamaeda, Masato Motomura, Tetsuya Asai, and Masayuki Ikebe, "Real-time tone mapping: A survey and cross-implementation hardware benchmark," *IEEE TCSVT*, 2022.

[16] Erik Reinhard, Michael Stark, Peter Shirley, and James Ferwerda, "Photographic tone reproduction for digital images," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 2023.

[17] Piti Irawan, James A Ferwerda, and Stephen R Marschner, "Perceptually based tone mapping of high dynamic range image streams.," in *Rendering Techniques*, 2005.

[18] Xian-Shi Zhang and Yong-Jie Li, "A retina inspired model for high dynamic range image rendering," in *Advances in Brain Inspired Cognitive Systems*, 2016.

[19] Raanan Fattal, Dani Lischinski, and Michael Werman, "Gradient domain high dynamic range compression," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 2023.

[20] Rafał K. Mantiuk, Karol Myszkowski, and Hans-Peter Seidel, *High Dynamic Range Imaging*, John Wiley Sons, Ltd, 2015.

[21] S Mann and R Picard, "On being "undigital" with digital cameras: Extending dynamic range by combining differently exposed pictures," 1994.

[22] Paul E. Debevec and Jitendra Malik, "Recovering high dynamic range radiance maps from photographs," in *SIGGRAPH*, 1997.

[23] Michael D Grossberg and Shree K Nayar, "Determining the camera response from images: What is knowable?," *Transactions on Pattern Analysis and Machine Intelligence*, 2003.

[24] Francesco Banterle, Alessandro Artusi, Kurt Debattista, and Alan Chalmers, *Advanced High Dynamic Range Imaging (2nd Edition)*, AK Peters (CRC Press), Natick, MA, USA, July 2017.

[25] Erum Arif Khan, Ahmet Oguz Akyuz, and Erik Reinhard, "Ghost removal in high dynamic range images," in *ICIP*, 2006.

[26] Katrien Jacobs, Celine Loscos, and Greg Ward, "Automatic high-dynamic range image generation for dynamic scenes," *IEEE Computer Graphics and Applications*, 2008.

[27] Fabrizio Pece and Jan Kautz, "Bitmap movement detection: Hdr for dynamic scenes," in *CVMP*. IEEE, 2010.

[28] Tae-Hyun Oh, Joon-Young Lee, Yu-Wing Tai, and In So Kweon, "Robust high dynamic range imaging by rank minimization," *Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[29] Chul Lee, Yuelong Li, and Vishal Monga, "Ghost-free high dynamic range imaging via rank minimization," *IEEE Signal Processing Letters*, 2014.

[30] Jun Hu, Orazio Gallo, and Kari Pulli, "Exposure stacks of live scenes with hand-held cameras," in *ECCV*, 2012.

[31] Yoav HaCohen, Eli Shechtman, Dan B Goldman, and Dani Lischinski, "Non-rigid dense correspondence with applications for image enhancement," *ACM TOG*, 2011.

[32] Luca Bogoni, "Extending dynamic range of monochrome and color images through fusion," in *ICPR*. IEEE, 2000.

[33] Henning Zimmer, Andrés Bruhn, and Joachim Weickert, "Freehand HDR imaging of moving scenes with simultaneous resolution enhancement," in *Computer Graphics Forum*. Wiley Online Library, 2011, vol. 30, pp. 405–414.

[34] Nima Khademi Kalantari and Ravi Ramamoorthi, "Deep high dynamic range imaging of dynamic scenes," *ACM TOG*, 2017.

[35] Ce Liu, *Beyond pixels: exploring new representations and applications for motion analysis*, Ph.D. thesis, Massachusetts Institute of Technology, 2009.

[36] Sibi Catley-Chandar, Thomas Tanay, Lucas Vandroux, Ales Leonardis, Gregory Slabaugh, and Eduardo Pérez-Pellitero, "FlexHDR: Modeling alignment and exposure uncertainties for flexible hdr imaging," *IEEE TIP*, 2022.

[37] Zhen Liu, Wenjie Lin, Xinpeng Li, Qing Rao, Ting Jiang, Mingyan Han, Haoqiang Fan, Jian Sun, and Shuaicheng Liu, "ADNet: Attention-guided deformable convolutional network for high dynamic range imaging," *CVPRW*, 2021.

[38] Qingsen Yan, Weiye Chen, Song Zhang, Yu Zhu, Jinqiu Sun, and Yanning Zhang, "A unified hdr imaging method with pixel and patch level," in *CVPR*, 2023.

[39] Qingsen Yan, Dong Gong, Qinfeng Shi, Anton van den Hengel, Chunhua Shen, Ian Reid, and Yanning Zhang, "Attention-guided network for ghost-free high dynamic range imaging," in *CVPR*, 2019.

[40] K Ram Prabhakar, Gowtham Senthil, Susmit Agrawal, R Venkatesh Babu, and Rama Krishna Sai S Gorthi, "Labeled from unlabeled: Exploiting unlabeled data for few-shot deep HDR deghosting," in *CVPR*, 2021.

[41] Michal Nazarczuk, Sibi Catley-Chandar, Ales Leonardis, and Eduardo Pérez-Pellitero, "Self-supervised HDR imaging from motion and exposure cues," in *ECCV*. Springer, 2025.

[42] Zhilu Zhang, Haoyu Wang, Shuai Liu, Xiaotao Wang, Lei Lei, and Wangmeng Zuo, "Self-supervised high dynamic range imaging with multi-exposure images in dynamic scenes," in *ICLR*, 2024.

[43] Stephen Mangiat and Jerry Gibson, "High dynamic range video with ghost removal," in *Applications of digital image processing XXXIII*. SPIE, 2010, vol. 7798, pp. 307–314.

[44] Nima Khademi Kalantari and Ravi Ramamoorthi, "Deep HDR video from sequences with alternating exposures," in *Computer Graphics Forum*. Wiley Online Library, 2019.

[45] Didier Le Gall, "Mpeg: A video compression standard for multimedia applications," *Digital Multimedia Systems*, vol. 34, no. 4, 1991.

[46] Masha T. Pourazad, Colin Doutre, Maryam Azimi, and Panos Nasiopoulos, "HEVC: The new gold standard for video compression," *IEEE Consumer Electronics Magazine*, 2012.

[47] Pierre-Étienne H. Fiquet and Eero P Simoncelli, "A polar prediction model for learning to represent visual transforms," in *NeurIPS*, 2023.

[48] 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 *CVPR*, 2018.

[49] Ping Hu, Simon Niklaus, Stan Sclaroff, and Kate Saenko, "Many-to-many splatting for efficient video frame interpolation," in *CVPR*, 2022.

[50] Xiangyu Xu, Li Siyao, Wenxiu Sun, Qian Yin, and Ming-Hsuan Yang, "Quadratic video interpolation," *NeurIPS*, vol. 32, 2019.

[51] Simon Niklaus, Long Mai, and Feng Liu, "Video frame interpolation via adaptive convolution," in *CVPR*, 2017.

[52] Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee, "AdaCoF: Adaptive collaboration of flows for video frame interpolation," in *CVPR*, 2020.

[53] Simon Niklaus, Long Mai, and Oliver Wang, "Revisiting adaptive convolutions for video frame interpolation," in *WACV*, 2021.

[54] Tarun Kalluri, Deepak Pathak, Manmohan Chandraker, and Du Tran, "FLAVR: Flow-agnostic video representations for fast frame interpolation," in *WACV*, 2023.

[55] Zhihao Shi, Xiangyu Xu, Xiaohong Liu, Jun Chen, and Ming-Hsuan Yang, "Video frame interpolation transformer," in *CVPR*, 2022.

[56] Guozhen Zhang, Yuhan Zhu, Haonan Wang, Youxin Chen, Gangshan Wu, and Limin Wang, "Extracting motion and appearance via inter-frame attention for efficient video frame interpolation," in *CVPR*, 2023.

[57] Ugur Çogalan, Mojtaba Bemana, Hans-Peter Seidel, and Karol Myszkowski, "Video frame interpolation for high dynamic range sequences captured with dual-exposure sensors," *Computer Graphics Forum*, 2023.

[58] Nima Khademi Kalantari, Eli Shechtman, Connelly Barnes, Soheil Darabi, Dan B Goldman, and Pradeep Sen, "Patch-based high dynamic range video.," *ACM TOG*, 2013.

[59] Yuelong Li, Chul Lee, and Vishal Monga, "A maximum a posteriori estimation framework for robust high dynamic range video synthesis," *IEEE TIP*, 2016.

[60] Michael D Grossberg and Shree K Nayar, "What is the space of camera response functions?," in *CVPR*, 2003.

[61] Mengshun Hu, Kui Jiang, Zhihang Zhong, Zheng Wang, and Yinqiang Zheng, "IQ-VFI: implicit quadratic motion estimation for video frame interpolation," in *CVPR*, 2024.

[62] Simon Niklaus and Feng Liu, "Context-aware synthesis for video frame interpolation," in *CVPR*, 2018.

[63] Diederik P Kingma, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[64] Joel Kronander, Stefan Gustavson, Gerhard Bonnet, and Jonas Unger, "Unified hdr reconstruction from raw cfa data," in *ICCP*. IEEE, 2013.

[65] Jan Froehlich, Stefan Grandinetti, Bernd Eberhardt, Simon Walter, Andreas Schilling, and Harald Brendel, "Creating cinematic wide gamut hdr-video for the evaluation of tone mapping operators and hdr-displays," in *SPIE Digital Photography X*, 2014.

[66] Trevor Canham, Michael Murdoch, Andre Sevigny, David Long, and Michael Brown, "2HDRVD: The handheld high dynamic range video dataset," in *Color Impact*, 2025.

[67] Cheng Guo, Leidong Fan, Ziyu Xue, and Xiuhua Jiang, "Learning a practical SDR-to-HDRTV up-conversion using new dataset and degradation models," in *CVPR*, 2023.

[68] Xiangyu Hu, Liquan Shen, Mingxing Jiang, Ran Ma, and Ping An, "LA-HDR: Light adaptive HDR reconstruction framework for single LDR image considering varied light conditions," *IEEE Transactions on Multimedia*, vol. 25, pp. 4814–4829, 2022.

[69] "ITU, Geneva, Switzerland, Recommendation ITU-R BT.500-14: Methodologies for the subjective assessment of the quality of television images," 2019.

[70] David Hasler and Sabine E Suesstrunk, "Measuring colorfulness in natural images," in *Human vision and electronic imaging VIII*. SPIE, 2003.

[71] Maryam Azimi and Rafal K. Mantiuk, "PU21: A novel perceptually uniform encoding for adapting existing quality metrics for HDR," in *IEEE Picture Coding Symposium*, 2021.

[72] Peibei Cao, Rafal K. Mantiuk, and Kede Ma, "Perceptual assessment and optimization of HDR image rendering," in *CVPR*, 2024.

[73] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018.