

LC-Mamba: Local and Continuous Mamba with Shifted Windows for Frame Interpolation

Min Wu Jeong Chae Eun Rhee
 Hanyang University

Abstract

In this paper, we propose LC-Mamba, a Mamba-based model that captures fine-grained spatiotemporal information in video frames, addressing limitations in current interpolation methods and enhancing performance. The main contributions are as follows: First, we apply a shifted local window technique to reduce historical decay and enhance local spatial features, allowing multiscale capture of detailed motion between frames. Second, we introduce a Hilbert curve-based selective state scan to maintain continuity across window boundaries, preserving spatial correlations both within and between windows. Third, we extend the Hilbert curve to enable voxel-level scanning to effectively capture spatiotemporal characteristics between frames. The proposed LC-Mamba achieves competitive results, with a PSNR of 36.53 dB on Vimeo-90k, outperforming prior models by +0.03 dB. The code and models are publicly available at <https://github.com/Miinuul/LC-Mamba.git>

1. INTRODUCTION

As deep learning-based image processing technologies reach maturity, there is a rapid increase in interest and research directed toward advancing video processing. Video frame interpolation (VFI) is a technique that generates intermediate frames to convert low frame rate videos into high frame rate ones through interpolation. This approach enhances video quality and immersion, enabling smooth playback on low-performance devices and in low-bandwidth environments. VFI is also critical in applications requiring motion tracking and analysis. Improving VFI performance largely depends on accurately capturing motion between frames, making effective modeling of inter-frame motion information essential.

Research on deep learning-based VFI can be broadly categorized into convolutional neural network (CNN)-based and vision transformer (ViT)-based approaches [2, 5, 9, 11, 14–16, 21, 26, 31–33, 35, 38, 42, 46, 51]. CNN-based methods typically use an encoder-decoder structure to extract op-

tical flow and synthesize intermediate frames. However, the small kernels in CNNs often struggle to capture complex motion accurately. To address this, the 4-dimensional cost volume technique was proposed, but it incurs high computational costs at high resolutions, posing limitations in resource-constrained environments [18, 31, 40]. Recently, ViT-based approaches have aimed to overcome the limitations of CNNs by leveraging attention mechanisms to capture long-range dependencies. For this purpose, local attention has been extended to spatiotemporal dimensions, with Sep-STS blocks introduced to enhance memory efficiency and reduce overhead [37]. However, this approach processes spatial and temporal information sequentially, which can inadequately capture active motion. To address this, a cross-attention approach has been proposed to model motion and structure simultaneously across frames [33, 46].

Recently, state space models like Mamba[10], which combine linear computational efficiency with the ability to handle global interactions, have emerged as promising solutions for high-resolution VFI. However, VFI requires robust spatiotemporal modeling of pixel relationships, posing several challenges. First, achieving locality is essential. Mamba performs attention by converting 2-dimensional (2D) images into 1-dimensional (1D) tokens through scanning, which leads to the loss of spatial characteristics inherent to images. This 1D scan introduces a historical decay effect, where the influence of initially interacted tokens diminishes under the influence of recent tokens, making it difficult to capture neighboring pixel properties. High-resolution frames, with their lower spatial correlation and greater detail, particularly benefit from localized scanning to mitigate historical decay. Second, maintaining spatial continuity during the scanning process is another important aspect. While some studies have attempted localized scanning within windows to achieve locality [13], these approaches often exhibit severe spatial discontinuities between windows. Third, capturing both spatial and temporal features remains a significant focus. For instance, Zhang et al. [47] proposed a selective state 2D scan that interleaves tokens from two frames sequentially to capture motion between frames. However, a global scan alone is insufficient

for accurately modeling the dense details present in high-resolution frames.

In this paper, we propose LC-Mamba, a model capable of locally and precisely capturing the spatiotemporal information of video frames. The proposed model effectively addresses the limitations of prior studies that struggled to capture the spatiotemporal characteristics of video inputs, demonstrating its performance when applied to the VFI task. The contributions of this paper are as follows:

- We implement a shifted window technique that limits the scan area to 8×8 , allowing detailed observation of local spatial characteristics in high-resolution frames and reducing the effects of historical decay. The hierarchical shifted windowing method captures intricate motion between frames at multiple scales.
- A 2D Hilbert curve-based selective state scan is introduced. This approach maintains spatial correlations both within and between windows, preserving spatial continuity.
- We propose a temporal scan using an interleaved selective state scan based on a 2D Hilbert curve, enabling the simultaneous modeling of the spatiotemporal characteristics of two frames.

2. RELATED WORK

2.1. Video Frame Interpolation

Existing VFI approaches are mainly divided into CNN-based and Transformer-based methods. Among them, flow-based models utilizing optical flow have shown strong performance [2, 14, 15, 21, 35, 38, 42]. For example, [21] proposed a CNN encoder-decoder to compute a 3D motion field for frame interpolation, and [14] introduced IFNet, predicting intermediate flow without pre-training. Other techniques, such as leveraging depth information [2], multi-scale features [35], bidirectional optical flow [38], and asymmetric blending [42], have also enhanced performance. Recently, CNN architectures have contributed to motion understanding in video by capturing multi-scale features and learning complex pixel patterns. However, CNNs' small receptive fields limit their ability to capture long-range dependencies. Despite attempts at deeper, more complex architectures, these have led to only modest performance gains relative to increased computational costs.

The ViT [8] enhances performance by addressing CNNs' limited receptive fields with transformers' long-range dependency modeling and attention mechanisms [9, 19, 26, 33, 37, 41, 46, 51]. However, its use in high-resolution, real-time tasks is constrained by the quadratic complexity of self-attention. To mitigate this, [22] introduced the swin transformer with a shifted window approach, effectively capturing local features while reducing computational load. Building on this, [37] extended local attention to spatiotem-

poral domains, and [26] incorporated a U-Net structure for long-range correlations. Further, [33] generated bidirectional motion for high-res interpolation, while [19] developed sparse global matching for effective large-motion handling.

Hybrid models combine CNNs and ViTs to address limitations of existing architectures. For example, [46] developed a model that preserves fine details in high-resolution inputs by using CNNs for low-level features and ViTs for motion and shape at lower resolutions. Likewise, [9] introduced a flow transformer block that merges optical flow with self-attention to handle large motions, enhancing performance with a multi-scale architecture.

2.2. Mamba

Mamba is a continuous state-space model (SSM) designed for efficient handling of complex, data-intensive sequences. An SSM linearly maps an input $x(t) \in \mathbb{R}^M$ at time t to an output $y(t) \in \mathbb{R}^M$ through a hidden state $h(t) \in \mathbb{R}^N$:

$$h'(t) = Ah(t) + Bx(t) \quad (1)$$

$$y(t) = Ch(t) \quad (2)$$

Here, $A \in \mathbb{R}^{N \times N}$, $B \in \mathbb{R}^{N \times 1}$, and $C \in \mathbb{R}^{1 \times N}$ control the state transition, input, and output mappings, respectively. $h'(t)$ represents the time derivative of $h(t)$, predicting the future state based on the current state and input.

Data such as words and tokens are discrete rather than continuous; therefore, Mamba discretizes the continuous SSM to integrate effectively into deep learning frameworks [10]. (3) shows the time update based on the previous state and input. The matrices $\bar{A}$ and $\bar{B}$ represent the influence of the state and input sequence at time $t-1$ on the state at time t .

$$h(t) = \bar{A}h(t-1) + \bar{B}x(t) \quad (3)$$

The output is computed through C in (4):

$$y(t) = Ch(t) \quad (4)$$

To perform discretization, a time scale parameter $\Delta \in \mathbb{R}$ and the zero-order hold (ZOH) method are used to convert the continuous matrices A and B into discrete matrices $\bar{A}$ and $\bar{B}$:

$$\bar{A} = e^{\Delta A} \quad (5)$$

$$\bar{B} = (\Delta A)^{-1}(e^{\Delta A} - I)\Delta B \approx \Delta B. \quad (6)$$

Mamba enhances computational efficiency by calculating the output for the entire sequence simultaneously through a global convolution operation, as shown in (7). Here, $\odot$ denotes convolution, and K represents the convolution kernel.

$$y = x \odot K, \quad K = [CB, CAB, \dots, CA^{L-1}B] \quad (7)$$

2.3. Vision Mamba

Mamba offers linear complexity with global dependency modeling, addressing both CNN and ViT limitations [10]. VMamba [20] and Vision Mamba (Vim) [52] applied Mamba to vision tasks, dividing images into 2D patches for processing with Mamba blocks. VMamba uses cross-scanning on horizontal and vertical axes, while Vim treats 2D images as sequences with bidirectional SSM. Despite this, both face challenges against ViT-based models. The 2D-to-1D scanning process improves efficiency but disrupts pixel structure, complicating information extraction. To counter this, Vision Mamba frameworks are exploring various scanning methods to retain spatial correlations, but current approaches like bidirectional [12, 36], cross [20], continuous 2D [44], and local scanning [13] still struggle to fully capture local relationships. For example, in raster-scanning a $H = 4, W = 4$ matrix X , tokens x_4 and x_5 may have consecutive indices yet remain spatially distant.

From (3), we can derive the matrix hidden state H as shown in (8), with each next state calculated by multiplying matrix A with the prior hidden state, following $h_{ij} = A_i h_{i-1j}$. In (5) and (6), as Δ increases, $\bar{A}$ decreases (due to the negative diagonal in A), while $\bar{B}$ grows. The value of Δ is optimized based on input utility, prioritizing more useful inputs with higher values, allowing selective focus on either the current input or previous hidden state, which manages spatial discontinuity. For instance, in (8), nearby tokens like x_1 and x_2 maintain continuity, leading $\bar{A}_2$ to hold a high value. However, spatially distant tokens like x_4 and x_5 lack continuity, causing Δ to increase, reducing $\bar{A}_5$ and weakening their connection to previous hidden states, particularly for x_5 . This sequential approach can cause information loss across distant tokens, limiting the model’s spatial perception. Resolving this requires architectural modifications beyond adjusting Δ alone.

$$H = \begin{bmatrix} \bar{B}_1 & 0 & \dots & 0 \\ \bar{A}_2 \bar{B}_1 & \bar{B}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{A}_4 \bar{A}_3 \bar{A}_2 \bar{B}_1 & \bar{A}_4 \bar{A}_3 \bar{B}_2 & \dots & 0 \\ \bar{A}_5 \bar{A}_4 \bar{A}_3 \bar{A}_2 \bar{B}_1 & \bar{A}_5 \bar{A}_4 \bar{A}_3 \bar{B}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{A}_{16} \bar{A}_{15} \dots \bar{A}_2 \bar{B}_1 & \bar{A}_{16} \bar{A}_{15} \dots \bar{A}_3 \bar{B}_2 & \dots & \bar{B}_{16} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_4 \\ x_5 \\ \vdots \\ x_{16} \end{bmatrix} \quad (8)$$

Various scanning techniques have been studied to capture the spatial characteristics of 2D images [34, 36, 45, 49, 50, 52], but high computational costs remain an issue. While existing scanning methods are effective for capturing local information and global context individually, they are limited in handling both aspects simultaneously and maintaining computational efficiency. Addressing these limitations is especially crucial for applying Vision Mamba to tasks like video processing, where a temporal dimension is additionally involved.

3. METHOD

3.1. LC-Mamba Overview

3.1.1. LC-Mamba for Frame Interpolation

The proposed LC-Mamba incorporates a shifted-window selective scan to capture local motion and proposes a 2D-selective scan based on the Hilbert curve (H-SS2D) to maintain spatial continuity. H-SS3D, an extension of H-SS2D to 3D, is designed to capture spatiotemporal continuity.

The architecture of LC-Mamba, as shown in Fig. 1, consists of four modules: low-level feature extractor (LFE), spatiotemporal feature extractor (STFE), motion estimator (ME), and residual refiner (RR). LFE is CNN-based and takes I_0 and I_1 as inputs, extracting the low-level features L_0^l and L_1^l for each frame at three pyramid levels, as shown in 9. Here, L_i^l represents the low-level feature of the i -th frame, with each feature map having a size of $\frac{H}{2^l} \times \frac{W}{2^l} \times 2^l C$. The resolution decreases by a factor of 1/4 at each level, while the number of channels doubles. H and W represent the frame height and width, and C denotes the number of channels.

$$L_i^0, L_i^1, L_i^2 = \text{LFE}(I_i) \quad (9)$$

The STFE in (10) takes the final layer outputs of LFE, L_0^2 and L_1^2 , as inputs to extract the spatiotemporal motion characteristics V^3 and V^4 of the frames. It is composed of hierarchical LC-Mamba blocks, which observe motion between the low-level features of each frame to identify the motion feature map.

$$V^3, V^4 = \text{STFE}([L_0^2, L_1^2]) \quad (10)$$

The ME uses the spatio-temporal features V^3 and V^4 to generate the bidirectional flows $F_{t \rightarrow 0}$ and $F_{t \rightarrow 1}$, along with a mask M . These are used to warp the input frames to time t , and the mask blends the two warped frames to produce the preliminary interpolated frame $\tilde{I}_t$, as shown in (11). Here, $\overleftarrow{W}$ denotes backward warping, and $\odot$ represents the Hadamard product.

$$\tilde{I}_t = M \odot \overleftarrow{W}(I_0, F_{t \rightarrow 0}) + (1 - M) \odot \overleftarrow{W}(I_1, F_{t \rightarrow 1}) \quad (11)$$

Finally, in (12), the RR utilizes the low-level features, flow, mask, and input images to generate a residual image, which is then added to $\tilde{I}_t$ to produce the final intermediate frame I_t .

$$I_t = \tilde{I}_t + \text{RR}(I, L, F, M) \quad (12)$$

3.1.2. LC-Mamba Block

The core module of STFE, as shown in Fig. 1, is the LC-Mamba block, which consists of an H-SS3D block for selective scanning and a channel attention block (CAB). Each (S)W H-SS3D and CAB is preceded by layer normalization (LN) and followed by residual connections to enhance

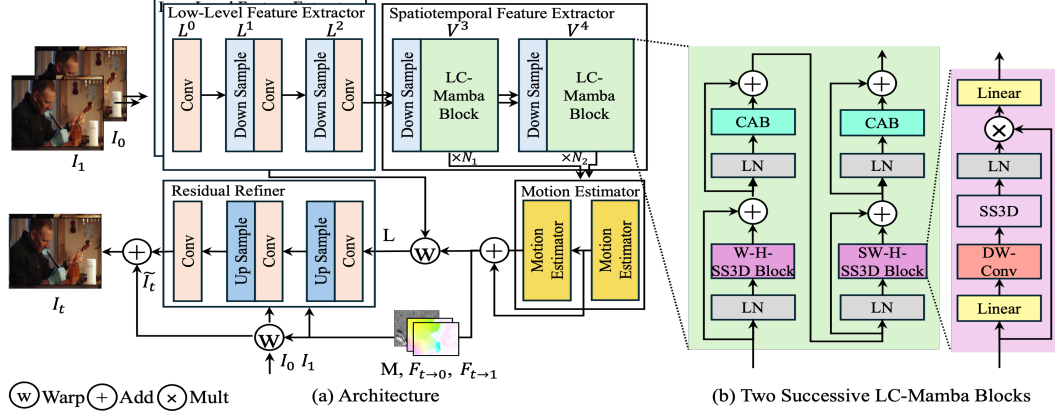


Figure 1. The proposed LC-Mamba model (a) Overall architecture consists of four modules: low-level feature extractor (LFE), spatiotemporal feature extractor (STFE), motion estimator (ME), and residual refiner (RR) (b) Two successive LC-Mamba block observe motion between the low-level features of each frame to identify the motion feature map. W-H-SS3D and SW-H-SS3D represent the window-based and shifted window-based 3D scanning methods, respectively. SS3D denotes the proposed selective scan based on the Hilbert curve.

structural stability. The overall architecture of LC-Mamba is based on the mixed-SSM block (MSB) [47], which adds CAB to the VSS block from Vmamba [20].

The VSS block replaces Mamba’s S6 with a 2D-selective-scan (SS2D) for vision data, scanning horizontally, vertically, and in reverse. While VMamba achieves global reach and efficiency, SS2D faces spatial loss and historical decay issues, especially in high-resolution images. The proposed LC-Mamba block differentiates itself by replacing the SS2D in the VSS Block with an (S)W H-SS3D. By using a Hilbert curve-based selective scan, spatial continuity is preserved, and window-based scanning is introduced to mitigate historical decay, emphasizing detail in high-resolution images. The shifting of windows allows boundary information exchange, capturing local relationships while partially reflecting global context. In video processing tasks like VFI, modeling temporal relationships alongside spatial characteristics is crucial; therefore, H-SS3D expands interleaved 2D scanning to 3D, effectively transforming the spatiotemporal model of two frames into a 1D sequence.

3.2. Hilbert Curve-based Selective Scan Using Shifted Windows

3.2.1. H-SS2D

To ensure spatial continuity in Vision Mamba, we introduce a Hilbert curve-based scan. As shown in Fig. 2 (a), the Hilbert curve is a space-filling curve that traverses 2D space, allowing spatially adjacent pixels in an image to maintain continuity when converted into a 1D sequence. The gradual subdivision of the Hilbert curve effectively captures spatial details in high-resolution images and preserves spatial correlations across various levels by expanding from

fine to coarse scales. Unlike the sequential scan in (8), the Hilbert curve-based scan maintains the spatial continuity of the hidden state H_h as in (13). For example, the continuity among adjacent tokens such as x_1 , x_2 , x_5 , and x_6 is preserved, mitigating the issue of historical decay.

$$H_h = \begin{bmatrix} \bar{B}_1 & 0 & \dots & 0 \\ \bar{A}_2 \bar{B}_1 & \bar{B}_2 & \dots & 0 \\ \bar{A}_6 \bar{A}_2 \bar{B}_1 & \bar{A}_6 \bar{B}_2 & \dots & 0 \\ \bar{A}_5 \bar{A}_6 \bar{A}_2 \bar{B}_1 & \bar{A}_5 \bar{A}_6 \bar{B}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \bar{A}_4 \bar{A}_3 \dots \bar{A}_2 \bar{B}_1 & \bar{A}_4 \bar{A}_3 \dots \bar{A}_6 \bar{B}_2 & \dots & \bar{B}_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_6 \\ x_5 \\ \vdots \\ x_4 \end{bmatrix} \quad (13)$$

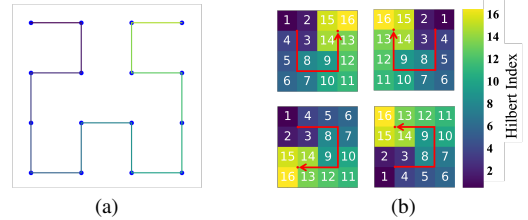


Figure 2. The proposed H-SS2D on 4×4 Grid (a) Hilbert Curve based Scan (b) The four types of scan paths. This multi-path structure help compensate for information loss.

To perform 2D scanning more effectively and minimize attenuation, we propose the H-SS2D model, which rearranges each token by combining four Hilbert curve paths, as shown in Fig. 2(b). Here, the index of each token represents the scanning order, and the color visually indicates the scanning direction. In the 2D scan, information is transmitted along four paths marked in red. In this process, the order and number of times A is multiplied are set differently for each path to facilitate information transfer, and because the value of A lies between 0 and 1, repeated multiplications may lead to attenuation. This multi-path structure is

designed to compensate for information loss through alternate paths, even if attenuation is significant along a specific path.

3.2.2. H-SS2D using Shifted Windows

Local H-SS2D within a window We apply H-SS2D within non-overlapping windows, restricting scanning distance to minimize attenuation and noise between tokens, allowing for dense information extraction. In VFI, important data are often closely located, and many optical flow studies assume that "flow is locally smooth," meaning nearby pixels tend to move similarly [27]. In Mamba, matrix $\bar{A}$ is multiplied with the previous hidden state to compute the next, following $h_{ij} = \bar{A}_i h_{i-1j}$, where values of $\bar{A}$ between 0 and 1 control attenuation, emphasizing recent tokens. For instance, in $h_{81} = \bar{A}_8 \bar{A}_7 \dots \bar{A}_2 \bar{B}_1 x_1$, repeated multiplication attenuates the link to x_1 , weakening spatial modeling between distant tokens, especially in high-resolution images. Unlike language models focusing on recent information, video models must capture consistent motion across both early and recent tokens, making window-based H-SS2D focused on local regions a practical solution.

Shifted window While window-based local scanning enhances local information, some global modeling is also necessary. In Fig. 3, dashed green boxes indicate intra-level disconnections, while blue boxes show inter-level ones. Due to the nature of Hilbert scanning, tokens in the central region are widely spaced, resulting in intra-level gaps (Fig. 2(b)). To improve connectivity in this region, multi-scale layers are introduced via down-sampling. However, inter-level gaps between non-overlapping windows persist. To balance local scanning efficiency with connectivity, we propose a shifted window strategy in the LC-Mamba block, alternating between two partition configurations. As shown in Fig. 3, the first layer uses a standard partition from the top-left pixel, dividing an 8×8 map into 2×2 windows of 4×4 ($M = 4$), as in the W-H-SS3D module (Fig. 1). The shifted configuration then shifts each window by $\lfloor M/2 \rfloor$ pixels, implemented in the SW-H-SS3D module.

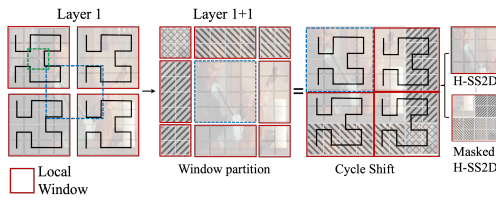


Figure 3. Shifted window based H-SS2D

Masked selective 2D scan A challenge with shifted window partitioning is that the number of windows increases from $\lceil h/M \rceil \times \lceil w/M \rceil$ to $(\lceil h/M \rceil + 1) \times (\lceil w/M \rceil + 1)$, resulting in some windows being smaller than $M \times M$ and non-square, while Hilbert curve-based scans require square

window sizes in powers of two. To address this, we use cyclic shifting inspired by the swin transformer [22], shown in Fig. 3, periodically shifting windows in the top-left direction. This maintains non-overlapping windows and uses a masking mechanism to ensure selective scanning only within each window. During scanning, Δ in (3) acts as a gating mechanism, with masked areas set to zero to control inputs approaching the hidden state as in (14). This cyclic shift preserves the regular partitioning structure, ensuring compatibility with Hilbert curve-based scanning.

$$h(t) = \bar{A}h(t-1) \quad \bar{A} = 1, \bar{B} = 0 \quad \text{if } \Delta = 0 \quad (14)$$

3.3. H-SS3D: Interleaved H-SS2D

To model temporal correlations, we implemented H-SS3D by applying the interleaved scanning method from [17, 25, 47] to H-SS2D. This method interweaves tokens of two frames, enabling information exchange across spatial and temporal dimensions to capture inter-frame characteristics. By scanning spatiotemporally adjacent pixels, H-SS3D preserves local information and progressively captures global patterns, achieving high performance even in complex motion scenes. Fig. 4 visually illustrates the H-SS3D implemented with interleaved H-SS2D. The detailed implementation process is as follows: Feature maps V_0 and V_1 , observed from the 0th and 1st frames, are input into the (S)W H-SS3D block shown in Fig. 1. In the rearrange stage, H-SS2D is applied in four directions to convert spatially neighboring pixels into a 1D token sequence. For example, for input V_0 , four sequences are generated: V_0^h (aligned horizontally), V_0^v (aligned vertically), $\bar{V}_0^h$ (aligned in the opposite direction of V_0^h), and $\bar{V}_1^v$ (aligned in the opposite direction of V_1^v). Each of the four sequences is then interleaved and produce four interleaved sequences: V_{Int_h} , V_{Int_v} , $\bar{V}_{\text{Int}_h}$, and $\bar{V}_{\text{Int}_v}$. Each interleaved sequence undergoes selective scanning using a window-based approach, focusing on spatially adjacent information. After the scanning is complete, each scanned interleaved sequence is deinterleaved and then reverted to its original position, followed by merging to output the scanned 2D features, $V_{\text{SS}0}$ and $V_{\text{SS}1}$. This process preserves the distinct features of each frame while effectively modeling spatiotemporal relationships.

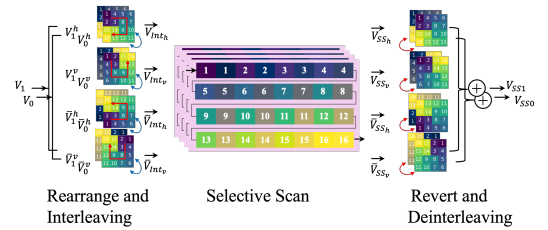


Figure 4. H-SS3D, implemented with interleaved H-SS2D

4. EXPERIENTIAL RESULTS

4.1. Datasets

The datasets used in this study are as follows: Vimeo90K [43] contains 3,782 frame triplets at 448×256 resolution. UCF101 [39] includes a test set of 379 frame triplets at 256×256 , as selected by [21]. Middlebury [1] uses the OTHER set at about 640×480 resolution. SNU-FILM [6] includes 1,240 frame triplets at 1280×720 , categorized by motion difficulty (Easy, Medium, Hard, Extreme). Lastly, Xiph [29] is tested in two versions: Xiph-2K, with images down-sampled to 2K, and Xiph-4K, with centrally cropped 4K images.

4.2. Implementation Details

4.2.1. Model Configuration

To demonstrate scalability, we present three versions of our model: an efficient model (Ours-E) optimized for speed, a balanced model (Ours-B) that provides a trade-off between processing time and performance and a performance model (Ours-P) focused on accuracy. Ours-E consists of 4 Mamba blocks per stage, with 16 channels. In Ours-B, the configuration is adjusted to 2 blocks per stage with 32 channels to achieve balance. Ours-P includes 4 blocks per stage to enhance performance. The window sizes for each model were set to 8, 8 and 16.

4.2.2. Training

The proposed LC-Mamba model was trained from scratch on the Vimeo90K [43] dataset. We used the AdamW [24] optimization algorithm, training on four NVIDIA RTX-4090 GPUs with a batch size of 32 for a total of 300 epochs. During the initial 2000 steps, a warm-up phase gradually increased the learning rate to 2×10^{-4} , followed by a cosine decay schedule [23] that reduced the learning rate to 2×10^{-5} over 300 epochs. Data augmentation included random flipping, rotation, sequence order reversal, and random cropping of 256×256 patches.

4.2.3. Loss

In the reconstruction loss L_{rec} , both locality and globality are considered. For intermediate frame generation, we use image reconstruction loss between the network output I_t and the ground truth frame I_t^{gt} . This loss consists of the following two main terms:

$$L_{\text{rec}} = \rho(I_t - I_t^{\text{gt}}) + L_{\text{cen}}(I_t, I_t^{\text{gt}}) + L_{\text{lap}}(I_t, I_t^{\text{gt}}) \quad (15)$$

For globality, we use the Charbonnier loss [3], $\rho(x) = (x^2 + \epsilon^2)^\alpha$, with $\alpha = 0.5$ and $\epsilon = 10^{-3}$, as an alternative to L_1 loss. To enhance locality, we use L_{cen} , which is calculated with the census loss [28], computing soft Hamming distance between census-transformed image patches of size 7×7 . Additionally, L_{lap} represents the L_1 loss between the Laplacian pyramids of the generated and ground

truth frames [30]. The warp loss L_{warp} is the sum of Laplacian pyramid losses between the warped frame and ground truth frame at each l -th motion estimation stage. λ serves as the weight for warp loss, set to 0.5 to balance the loss components.

$$L_{\text{warp}} = \lambda \sum_l L_{\text{lap}}(\tilde{I}_t^l, I_t^{\text{gt}}) \quad (16)$$

The final loss function L is defined as the sum of the reconstruction loss L_{rec} and the weighted warp loss L_{warp} , aiming to minimize distortion while maintaining reconstruction accuracy across local and global scales:

$$L = L_{\text{rec}} + L_{\text{warp}} \quad (17)$$

4.3. Comparison with the State-of-the-Art Methods

We compared the proposed LC-Mamba model with recent state-of-the-art video frame interpolation methods. The comparison included CNN-based models [1, 2, 5, 6, 11, 14–16, 30–32], transformer-based models [19, 26, 46], and Mamba-based models [47], evaluating performance across various datasets.

4.3.1. Quantitative Evaluation

To validate our model's strength in capturing both fine details and global context with competitive performance using fewer parameters, we evaluated its performance across several datasets. Vimeo90K and UCF101 were used for interpolation evaluation at low resolutions, while Xiph 2K and Xiph 4K assessed performance at higher resolutions. The SNU-FILM dataset, divided into four categories based on motion difficulty (easy, medium, hard, and extreme), provided a comprehensive assessment across varying levels of complexity. For fair comparison with previous studies, we grouped algorithms based on similar parameter counts.

Table 1 presents the quality of intermediate frames, showing interpolation error (IE) for the Middlebury dataset and peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) for the other datasets. The Ours-E model demonstrates balanced performance across low-resolution (Vimeo90K, UCF101) and high-resolution datasets (Xiph 2K, 4K), as well as SNU-FILM with complex motion. This success is due to the window-based Hilbert scan, which prioritizes local detail while preserving global context through shifted windowing, resulting in balanced interpolation. For the Ours-B model, we increased the channels to 32 and reduced the number of Mamba blocks, N_1 and N_2 , from 4 to 2, to achieve a balance between efficiency and performance. While Ours-B performs well across both low- and high-resolution datasets, a slight performance drop was observed on SNU-FILM, which provides a rigorous evaluation of interpolation capabilities. In Ours-P, we set the number of Mamba blocks

Table 1. Quantitative comparison across benchmarks (IE on Middlebury, PSNR/SSIM on other datasets). The best and second-best results are **bolded** and underlined, respectively. "Out of Memory" is marked as "OOM," and "+" denotes results obtained by us; other results are cited from [11, 14, 15, 26, 35, 46]. Testing was performed following the procedures in [14] for Vimeo90K/UCF101/Middlebury, [30] for Xiph, and [15] for SNU-FILM, with Test-Time Augmentation (TTA) disabled to ensure a fair comparison.

Method	Vimeo90K	UCF101	Xiph		M.B.	SNU-FILM				Params (M)
			2K	4K		Easy	Medium	Hard	Extreme	
ToFlow [1]	33.73/0.9682	34.58/0.9667	33.93/0.922	30.74/0.856	2.15	39.08/0.9890	34.39/0.9740	28.44/0.9180	23.39/0.8310	1.4
IFRNet [15]	35.80/0.9794	35.29/0.9693	36.00/0.936	33.99/0.893	1.95	40.03/0.9905	35.94/0.9793	30.41/0.9358	25.05/0.8587	5
M2M [11]	35.47/0.9778	35.28/0.9694	36.44/0.943	33.92/0.899	2.09	39.66/0.9904	35.74/0.9794	30.30/0.9360	25.08/0.8604	7.6
SoftSplat [30]	36.10/0.9802	35.39/0.9697	36.62/0.944	33.60/0.901	1.81	39.88/0.9897	35.68/0.9772	30.19/0.9312	24.83/0.8500	7.7
RIFE [14]	35.61/0.9779	35.28/0.9690	36.19/0.938	33.76/0.894	1.96	39.80/0.9903	35.76/0.9787	30.36/0.9351	25.27/0.8601	9.8
BM3C [31]	35.01/0.9764	35.15/0.9689	32.82/0.928	31.19/0.880	2.04	39.90/0.9902	35.31/0.9774	29.33/0.9270	23.92/0.8432	11.1
Ours-E	36.19/0.9803	35.33/0.9695	36.67/0.943	34.26/0.903	1.98	39.82/0.9907	35.87/0.9797	30.54/0.9373	25.33/0.8626	6.7
EMA-S [46]	36.07/0.9794+	35.34/0.9696+	36.54/0.942+	34.24/0.902+	1.94+	39.81/0.9903+	35.88/0.9792+	30.68/0.9371+	25.47/0.8627+	14.5
VFI-Mamba-S [47]	36.09/0.9800+	35.35/0.9696+	36.71/0.942+	34.26/0.902+	1.97+	40.21/0.9912+	36.17/0.9802+	30.80/0.9382+	25.59/0.8655+	16.8
VFIFormer-S [26]	36.37/0.9810+	35.36/0.9698+	36.55/0.943+	33.37/0.899+	1.89+	40.02/0.9906+	35.91/0.9793+	30.22/0.9348+	24.80/0.8568+	17.1
ABME [32]	36.18/0.9805	35.38/0.9698	36.53/0.944	33.73/0.901	2.01	39.59/0.9901	35.77/0.9789	30.58/0.9364	25.42/0.8639	18.1
Ours-B	36.43/0.9813	35.39/0.9698	36.90/0.945	34.26/0.904	1.89	40.07/0.9909	36.08/0.9801	30.59/0.9375	25.35/0.8630	16.2
SGM-VFI-S-1/2 [19]	35.81/0.9785+	35.33/0.9692+	36.06/0.940+	33.26/0.897+	1.87+	40.36/0.9900+	36.12/0.9787+	30.62/0.9351+	25.38/0.8615+	20.8
SepConv [5]	33.79/0.9702	34.78/0.9669	34.77/0.929	32.06/0.880	2.27	39.41/0.9900	34.97/0.9762	29.36/0.9253	24.31/0.8448	21.7
AdaCoF [16]	34.47/0.9730	34.90/0.9680	34.86/0.928	31.68/0.870	2.24	39.80/0.9900	35.05/0.9754	29.46/0.9244	24.31/0.8439	21.8
DAIN [2]	34.71/0.9756	34.99/0.9683	35.95/0.940	33.49/0.895	2.04	39.73/0.9902	35.46/0.9780	30.17/0.9335	25.09/0.8584	24.0
VFIFormer [26]	36.50/0.9815+	35.42/0.9699+	OOM+	OOM+	1.82+	40.12/0.9907+	36.09/0.9798+	30.67/0.9378+	25.43/0.8643+	24.1
Ours-P	36.53/0.9816	35.42/0.9699	36.99/0.946	34.49/0.906	1.92	40.16/0.9909	36.17/0.9802	30.72/0.9382	25.48/0.8645	25.4
CAIN [6]	34.65/0.9730	34.91/0.9690	35.21/0.937	32.56/0.901	2.28	39.89/0.9900	35.61/0.9776	29.90/0.9292	24.78/0.8507	42.8
EMA [46]	36.50/0.9814+	35.38/0.9697+	36.74/0.944+	34.54/0.905+	1.84+	39.57/0.9905+	35.85/0.9797+	30.80/0.9389+	25.59/0.8650+	65.6
VFI-Mamba [47]	36.45/0.9807+	35.37/0.9699+	37.02/0.944+	34.39/0.904+	1.89+	40.41/0.9903+	36.30/0.9794+	30.89/0.9387+	25.68/0.8661+	66.1
Ours-P	36.53/0.9816	35.42/0.9699	36.99/0.946	34.49/0.906	1.92	40.16/0.9909	36.17/0.9802	30.72/0.9382	25.48/0.8645	25.4

to 4 per stage, a window size of 16, and omitted shifting. This configuration enhanced interpolation capabilities across resolutions, especially in complex tasks, demonstrating LC-Mamba’s effectiveness at capturing intricate local details. Notably, it achieved competitive performance even compared to models with 2–3 times more parameters. In summary, the proposed LC-Mamba model’s robust performance can be attributed to the window-based Hilbert selective scan, which effectively captures information from fine details to broader context. This highlights LC-Mamba’s potential as a parameter-efficient, high-performing solution for video frame interpolation across a range of applications. Further experimental and analytical results can be found in the supplementary material.

4.3.2. Qualitative Evaluation

Table 2 shows that the proposed Ours-E model achieved competitive learning-based perceptual image patch similarity (LPIPS) [48] and flow-based LPIPS (FloLPIPS) [7] scores across various datasets. Here, lower values of LPIPS and FloLPIPS indicate better visual quality.

Interpolating frames by focusing too much on either global or local aspects alone can adversely affect subjective image quality. The window-based Hilbert selective scanning method effectively provides balanced image quality performance by leveraging both local and global features. To verify this, we visually compared the intermediate frames generated by various VFI models, including the proposed model. As shown in Fig. 5, in the areas marked by orange squares, the results are generally similar, but some contain sharper details. Despite being an efficient model, it effectively captures precise spatial details in scenes with complex motions, maintaining high quality and demonstrating superior visual performance. In conclu-

sion, the LC-Mamba model demonstrates high performance in both quantitative and qualitative evaluations, proving excellence in parameter efficiency and visual quality.

Table 2. Evaluation of LPIPS and FloLPIPS. The top-performing results are highlighted in **bold**, with the second-best results underlined. All reported results are derived from our own experiments.

Method	Vimeo90K	Xiph-2K	Xiph-4K	SNU-FILM (avg.)	Params (M)
VFIFormer-S [26]	0.0219 / 0.0415	0.1074 / 0.1506	0.2266 / 0.2581	0.0678 / 0.1142	17.1
SGM-VFI-S-1/2 [46]	0.0227 / 0.0405	0.1011 / 0.1430	0.2205 / 0.2534	0.0612 / 0.0984	20.8
EMA-S [19]	0.0219 / 0.0400	0.0970 / 0.1271	0.2178 / 0.2415	0.0610 / 0.0984	14.5
VFI-Mamba-S [47]	0.0214 / 0.0390	0.1007 / 0.1304	0.2182 / 0.2414	0.0592 / 0.0957	16.8
Ours-E	<u>0.0215</u> / <u>0.0398</u>	<u>0.0995</u> / 0.1310	0.2144 / 0.2428	<u>0.0601</u> / 0.0989	6.7

4.4. Ablation Study

In Table 3, the proposed scanning method is compared with cross [20, 25], continuous [44], local [13], bidirectional [4, 52], and Z-order scanning methods across the Vimeo90K, Xiph, and SNU-FILM datasets. To clearly observe the impact of scanning, comparisons were made using the Ours-E model. The proposed SW-H-SS3D uses window-based Hilbert selective scanning, while comparison methods employ interleaving (ILV) without windows. Differences were minimal in low-resolution results but emerged significantly in high-resolution, complex-motion datasets. Bidirectional scanning, limited to the horizontal direction, fails to fully capture 2D spatial locality. Cross scanning, covering both horizontal and vertical directions, achieves good average performance, while continuous scanning, using a snake pattern, performs well in high resolutions but struggles with complex motion. Local and Z-order scanning, which prioritize inter-window continuity, maintain spatial locality and generate high-quality frames but face challenges with global feature capture at higher resolutions. In contrast, the proposed window-based Hilbert

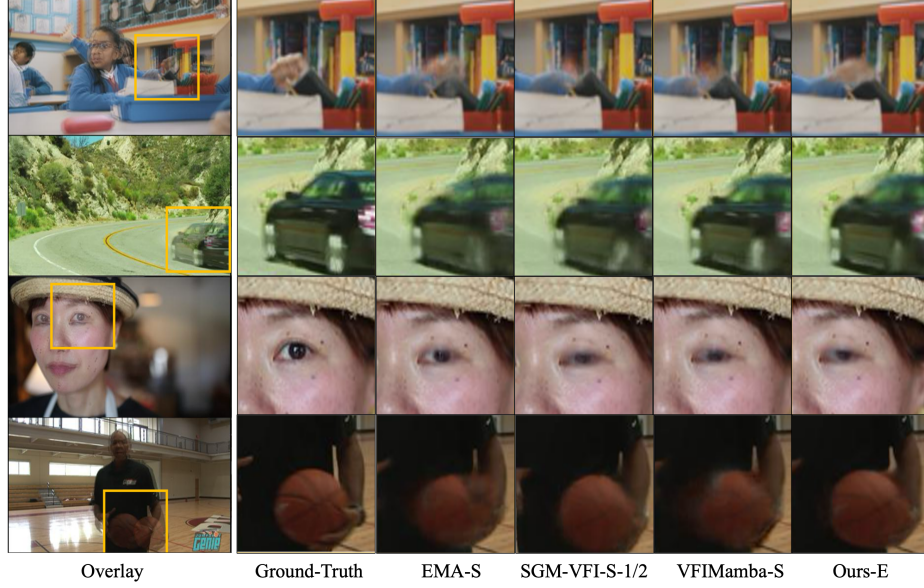


Figure 5. Visual comparison on Vimeo90K [43] triplet set. The position pointed by the box indicates where our model performs better.

scan gradually extends locality to a global scale, providing balanced performance. It consistently outperformed other methods across settings, reducing information loss in high-resolution datasets and maintaining strong performance in scenes with complex motion.

Table 3. Performance comparison of different scanning methods

Scanning	Vimeo90K	Xiph-2K	Xiph-4K	SNU-FILM(avg.)
Bidirection w/ ILV	35.41/0.9799	36.00/0.9381	33.13/0.8937	32.32/0.9405
Cross w/ ILV	36.07/0.9799	35.73/0.9362	33.80/0.8947	32.53/0.9413
Continuous w/ ILV	36.09/0.9800	36.57/0.9428	33.99/0.9010	24.59/0.8335
Local w/ ILV	36.11/0.9801	36.38/0.9415	34.01/0.9008	32.62/0.9411
Z-order w/ ILV	36.13/0.9800	35.91/0.9371	33.30/0.8932	32.36/0.9417
SW-H-SS3D	36.19/0.9803	36.67/0.9437	34.26/0.9036	32.89/0.9426

Table 4 shows that window size impacts the performance of Hilbert curve-based scanning in Ours-B model. Larger window sizes capture a wider area, improving spatiotemporal information integration, but excessively large windows can weaken fine motion detail capture. While omitting shifted windows enhanced performance in lower resolutions, using them in 4K resolution improved global information extraction. At a window size of 16, the model captured global features effectively without shifted windows, maintaining performance levels.

Table 4. Ablation studies for window settings. The “Settings” column shows window size and whether shifting is used, while the other columns display performance (PSNR/SSIM).

Settings	Vimeo90K	Xiph-2K	Xiph-4K	SNU-FILM(avg.)
8 w/ shift	36.43/0.9813	36.90/0.9452	34.26/0.9046	33.02/0.9429
8 w/o shift	36.45/0.9813	36.78/0.9448	34.15/0.9042	32.95/0.9428
16 w/ shift	36.44/0.9813	36.88/0.9454	34.15/0.9047	33.02/0.9429
16 w/o shift	36.46/0.9813	36.88/0.9449	34.23/0.9045	33.05/0.9429

Table 5 demonstrates the scalability of LC-Mamba blocks. Adjusting the number and complexity of these blocks allows control over the model’s performance. Doubling either the number of blocks or the channel size significantly boosts performance. The increase in channel size raises model complexity, yielding notable performance gains.

Table 5. Ablation study on the scalable capability of LC-Mamba blocks. This table presents the performance (PSNR/SSIM) for each model configuration.

$N_1 / N_2 / C$	Vimeo90K	Xiph-2K	Xiph-4K	SNU-FILM(avg.)
4 / 4 / 16	36.19/0.9803	36.67/0.9437	34.26/0.9036	32.89/0.9426
2 / 2 / 32	36.43/0.9813	36.90/0.9452	34.26/0.9046	33.02/0.9429
4 / 4 / 32	36.53/0.9816	36.99/0.9459	34.49/0.9061	33.13/0.9435

5. Conclusion and Future Work

This study aimed to enhance Mamba’s spatiotemporal modeling for VFI tasks by leveraging the Hilbert curve to capture spatiotemporal information and strengthen spatial continuity, minimizing information loss. The masked selective scan method, using window-based and shifted scanning, balanced local and global information effectively. Experimental results showed improved efficiency in both low- and high-resolution settings, capturing diverse motion patterns. Despite Mamba’s linear computational benefits, it still has limitations in global modeling, suggesting a need for further work on long-range dependency modeling.

Acknowledgements

This work was supported by the Technology Innovation Program (or Industrial Strategic Technology Development Program) (RS-2023-00235718, 23040-15FC) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea) (1415187505), and by the Institute of Information & Communications Technology Planning & Evaluation (IITP) under the artificial intelligence semiconductor support program to nurture the best talents (IITP-2025-RS-2023-00253914) grant funded by the Korea government (MSIT). It was also supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2025-00562356).

References

- [1] Simon Baker, Daniel Scharstein, James P Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International journal of computer vision*, 92:1–31, 2011. [1](#), [2](#), [6](#), [7](#)
- [2] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3703–3712, 2019. [1](#), [2](#), [6](#), [7](#)
- [3] Pierre Charbonnier, Laure Blanc-Feraud, Gilles Aubert, and Michel Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *Proceedings of 1st international conference on image processing*, pages 168–172. IEEE, 1994. [6](#)
- [4] Guo Chen, Yifei Huang, Jilan Xu, Baoqi Pei, Zhe Chen, Zhiqi Li, Jiahao Wang, Kunchang Li, Tong Lu, and Limin Wang. Video mamba suite: State space model as a versatile alternative for video understanding. *arXiv preprint arXiv:2403.09626*, 2024. [7](#)
- [5] Xianhang Cheng and Zhenzhong Chen. Video frame interpolation via deformable separable convolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10607–10614, 2020. [1](#), [6](#), [7](#)
- [6] Myungsub Choi, Heewon Kim, Bohyung Han, Ning Xu, and Kyoung Mu Lee. Channel attention is all you need for video frame interpolation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10663–10671, 2020. [6](#), [7](#)
- [7] Duolikun Danier, Fan Zhang, and David Bull. Flolpips: A bespoke video quality metric for frame interpolation. In *2022 Picture Coding Symposium (PCS)*, pages 283–287. IEEE, 2022. [7](#)
- [8] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. [2](#)
- [9] Pan Gao, Haoyue Tian, and Jie Qin. Video frame interpolation with flow transformer. *arXiv preprint arXiv:2307.16144*, 2023. [1](#), [2](#)
- [10] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. [1](#), [2](#), [3](#)
- [11] Ping Hu, Simon Niklaus, Stan Sclaroff, and Kate Saenko. Many-to-many splatting for efficient video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3553–3562, 2022. [1](#), [6](#), [7](#)
- [12] Vincent Tao Hu, Stefan Andreas Baumann, Ming Gui, Olga Grebenkova, Pingchuan Ma, Johannes S Fischer, and Björn Ommer. Zigma: A dit-style zigzag mamba diffusion model. *arXiv preprint arXiv:2403.13802*, 2024. [3](#)
- [13] Tao Huang, Xiaohuan Pei, Shan You, Fei Wang, Chen Qian, and Chang Xu. Localmamba: Visual state space model with windowed selective scan. *arXiv preprint arXiv:2403.09338*, 2024. [1](#), [3](#), [7](#)
- [14] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *European Conference on Computer Vision*, pages 624–642. Springer, 2022. [1](#), [2](#), [6](#), [7](#)
- [15] Lingtong Kong, Boyuan Jiang, Donghao Luo, Wenqing Chu, Xiaoming Huang, Ying Tai, Chengjie Wang, and Jie Yang. Ifrnet: Intermediate feature refine network for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1969–1978, 2022. [2](#), [7](#)
- [16] Hyeonmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. Adacof: Adaptive collaboration of flows for video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5316–5325, 2020. [1](#), [6](#), [7](#)
- [17] Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. In *European Conference on Computer Vision*, pages 237–255. Springer, 2025. [5](#)
- [18] Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9801–9810, 2023. [1](#)
- [19] Chunxu Liu, Guozhen Zhang, Rui Zhao, and Limin Wang. Sparse global matching for video frame interpolation with large motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19125–19134, 2024. [2](#), [6](#), [7](#)
- [20] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. VMamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024. [3](#), [4](#), [7](#)
- [21] Ziwei Liu, Raymond A Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video frame synthesis using deep voxel flow. In *Proceedings of the IEEE international conference on computer vision*, pages 4463–4471, 2017. [1](#), [2](#), [6](#)
- [22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. [2](#), [5](#)
- [23] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic

- gradient descent with warm restarts. arXiv preprint arXiv:1608.03983, 2016. 6
- [24] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. arXiv preprint arXiv:1711.05101, 5, 2017. 6
- [25] Hui Lu, Albert Ali Salah, and Ronald Poppe. Videomamba: A leap forward for mamba in video understanding. arXiv preprint arXiv:2406.19006, 2024. 5, 7
- [26] Liying Lu, Ruizheng Wu, Huaijia Lin, Jiangbo Lu, and Jiaya Jia. Video frame interpolation with transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3532–3542, 2022. 1, 2, 6, 7
- [27] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In IJCAI’81: 7th international joint conference on Artificial intelligence, pages 674–679, 1981. 5
- [28] Simon Meister, Junhwa Hur, and Stefan Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In Proceedings of the AAAI conference on artificial intelligence, 2018. 6
- [29] Christopher Montgomery and H Lars. Xiph. org video test media (derf’s collection). Online, <https://media.xiph.org/video/derf>, 6, 1994. 6
- [30] Simon Niklaus and Feng Liu. Softmax splatting for video frame interpolation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 5437–5446, 2020. 6, 7
- [31] Junheum Park, Keunsoo Ko, Chul Lee, and Chang-Su Kim. Bmbc: Bilateral motion estimation with bilateral cost volume for video interpolation. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16, pages 109–125. Springer, 2020. 1, 7
- [32] Junheum Park, Chul Lee, and Chang-Su Kim. Asymmetric bilateral motion estimation for video frame interpolation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 14539–14548, 2021. 6, 7
- [33] Junheum Park, Jintae Kim, and Chang-Su Kim. Biformer: Learning bilateral motion estimation via bilateral transformer for 4k video frame interpolation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1568–1577, 2023. 1, 2
- [34] Yanyuan Qiao, Zheng Yu, Longteng Guo, Sihan Chen, Zijia Zhao, Mingzhen Sun, Qi Wu, and Jing Liu. VI-mamba: Exploring state space models for multimodal learning. arXiv preprint arXiv:2403.13600, 2024. 3
- [35] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In European Conference on Computer Vision, pages 250–266. Springer, 2022. 1, 2, 7
- [36] Yuan Shi, Bin Xia, Xiaoyu Jin, Xing Wang, Tianyu Zhao, Xin Xia, Xuefeng Xiao, and Wenming Yang. Vmambair: Visual state space model for image restoration. arXiv preprint arXiv:2403.11423, 2024. 3
- [37] Zhihao Shi, Xiangyu Xu, Xiaohong Liu, Jun Chen, and Ming-Hsuan Yang. Video frame interpolation transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 17482–17491, 2022. 1, 2
- [38] Hyeonjun Sim, Jihyong Oh, and Munchurl Kim. Xvfi: extreme video frame interpolation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 14489–14498, 2021. 1, 2
- [39] K Soomro. Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402, 2012. 6
- [40] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16, pages 402–419. Springer, 2020. 1
- [41] A Vaswani. Attention is all you need. Advances in Neural Information Processing Systems, 2017. 2
- [42] Guangyang Wu, Xin Tao, Changlin Li, Wenyi Wang, Xiaohong Liu, and Qingqing Zheng. Perception-oriented video frame interpolation via asymmetric blending. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2753–2762, 2024. 1, 2
- [43] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. International Journal of Computer Vision, 127:1106–1125, 2019. 6, 8
- [44] Chenhongyi Yang, Zehui Chen, Miguel Espinosa, Linus Ericsson, Zhenyu Wang, Jiaming Liu, and Elliot J Crowley. Plainmamba: Improving non-hierarchical mamba in visual recognition. arXiv preprint arXiv:2403.17695, 2024. 3, 7
- [45] Yijun Yang, Zhaohu Xing, and Lei Zhu. Vivim: a video vision mamba for medical video object segmentation. arXiv preprint arXiv:2401.14168, 2024. 3
- [46] Guozhen Zhang, Yuhao Zhu, Haonan Wang, Youxin Chen, Gangshan Wu, and Limin Wang. Extracting motion and appearance via inter-frame attention for efficient video frame interpolation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5682–5692, 2023. 1, 2, 6, 7
- [47] Guozhen Zhang, Chunxu Liu, Yutao Cui, Xiaotong Zhao, Kai Ma, and Limin Wang. Vfimamba: Video frame interpolation with state space models. arXiv preprint arXiv:2407.02315, 2024. 1, 4, 5, 6, 7
- [48] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 586–595, 2018. 7
- [49] Zeyu Zhang, Akide Liu, Ian Reid, Richard Hartley, Bohan Zhuang, and Hao Tang. Motion mamba: Efficient and long sequence motion generation. In European Conference on Computer Vision, pages 265–282. Springer, 2025. 3
- [50] Sijie Zhao, Hao Chen, Xueliang Zhang, Pengfeng Xiao, Lei Bai, and Wanli Ouyang. Rs-mamba for large remote sensing image dense prediction. arXiv preprint arXiv:2404.02668, 2024. 3
- [51] Kun Zhou, Wenbo Li, Xiaoguang Han, and Jiangbo Lu. Exploring motion ambiguity and alignment for high-quality

video frame interpolation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 22169–22179, 2023. [1](#), [2](#)

- [52] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. arXiv preprint arXiv:2401.09417, 2024. [3](#), [7](#)