Gather-Scatter Mamba:
Accelerating Propagation with Efficient State Space Model

Hyun-kyu Ko[1]   Youbin Kim[2]   Jihyeon Park[1]   Dongheok Park[2]   Gyeongjin Kang[1]
Wonjun Cho[3]   Hyung Yi[3]   Eunbyung Park[4*]

[1]Department of Electrical and Computer Engineering, Sungkyunkwan University
[2]Department of Artificial Intelligence, Sungkyunkwan University
[3]Hanwha Systems, Republic of Korea
[4]Department of Artificial Intelligence, Yonsei University

{laniko, ybin108, fairytale, leao8869, ggggjin99}@skku.edu
{wonjun78.cho, hyung.yi}@hanwha.com   epark@yonsei.ac.kr

Abstract

State Space Models (SSMs)—most notably RNNs—have historically played a central role in sequential modeling. Although attention mechanisms such as Transformers have since dominated due to their ability to model global context, their quadratic complexity and limited scalability make them less suited for long sequences. Video super-resolution (VSR) methods have traditionally relied on recurrent architectures to propagate features across frames. However, such approaches suffer from well-known issues including vanishing gradients, lack of parallelism, and slow inference speed. Recent advances in selective SSMs like Mamba [11] offer a compelling alternative: by enabling input-dependent state transitions with linear-time complexity, Mamba mitigates these issues while maintaining strong long-range modeling capabilities. Despite this potential, Mamba alone struggles to capture fine-grained spatial dependencies due to its causal nature and lack of explicit context aggregation. To address this, we propose a hybrid architecture that combines shifted window self-attention for spatial context aggregation with Mamba-based selective scanning for efficient temporal propagation. Furthermore, we introduce Gather-Scatter Mamba (GSM), an alignment-aware mechanism that warps features toward a center anchor frame within the temporal window before Mamba propagation and scatters them back afterward, effectively reducing occlusion artifacts and ensuring effective redistribution of aggregated information across all frames.

The official implementation is provided at: https:// github.com/Ko-Lani/GSMamba.

1. Introduction

Long before the rise of Transformer architectures [37], recurrent neural networks (RNNs)[9] dominated sequence modeling, particularly in early video understanding[34, 45] and video sequence generation [8, 38]. However, the inherent limitations of RNNs—vanishing gradients [1] and limited parallel computation—ultimately led to their decline. Transformers [7, 26, 37] introduced a revolutionary paradigm based on parallel attention mechanisms, now widely adopted across various domains. Despite this success, Transformers' quadratic computational complexity significantly restricts their practicality in video tasks, where long-range temporal modeling is essential.

Temporal modeling remains especially critical in video super-resolution (VSR), which relies on accurately aggregating details from neighboring frames to restore high-frequency visual content. Due to computational constraints, most recent VSR models [4, 5, 22, 33, 41] employ recurrent propagation, maintaining explicit hidden states that propagate forward in time. However, these approaches inherently suffer from two major drawbacks. First, the strictly causal nature of propagation limits the model's ability to leverage future frames, even when bidirectional strategies (forward and backward passes) are employed. Second, supporting frames used for propagation are typically discarded immediately afterward, despite the substantial computational cost of extracting their features.

A recent breakthrough in sequence modeling is Mamba [11], a structured state space model (S4) [12] that employs dynamic, input-dependent transitions. Unlike Transformers, Mamba has linear computational complexity with respect to sequence length, making it highly suitable for long-sequence tasks like video processing. Further-

*Corresponding author

more, Mamba supports parallel computation while maintaining strong long-range dependency modeling capabilities. Although promising, applying Mamba directly to VSR is nontrivial. Previous works in vision [14, 21, 25, 46] have reported that naïve temporal-first scanning can hurt performance, and thus typically adopt spatial-first or hybrid spatial–temporal scanning strategies (e.g., Hilbert curves) instead.

To address these challenges, we propose Gather-Scatter Mamba, a novel VSR framework that leverages alignment-aware temporal propagation and symmetric residual redistribution. At the heart of our method lies a gather-scatter strategy: During the gather phase, neighboring frames are aligned to each anchor frame via warping, and the aligned features are temporally flattened and passed through a State Space Model (Mamba) for efficient long-range temporal modeling. In contrast to previous methods that update only the anchor, our scatter phase performs explicit residual redistribution, warping the output residuals from the anchor frame back to each supporting frame. This allows all frames in the temporal window to be enhanced jointly, enabling joint refinement of all frames in the temporal window and maximizing the use of computed features.

Unlike prior sliding-window approaches that treat past frames merely as support for the current frame, we update all frames within the window jointly by centering the anchor and aligning neighbors toward it. This removes the strict past–present distinction and allows information to flow symmetrically from both directions. The resulting design shortens alignment paths, reduces warping error, and enables balanced feature aggregation across the entire temporal window. Although the window still advances sequentially, we additionally perform a backward pass over the sequence, ensuring bidirectional propagation and maximizing information reuse across all frames.

Our contributions can be summarized as follows:

- We propose Gather-Scatter Mamba, the first VSR framework to integrate Mamba [11] for temporal propagation, enabling long-range temporal modeling with linear complexity.
- Unlike prior Mamba usage in video models, we introduce a gather-and-align step that explicitly aligns neighboring frames before temporal-first scanning, allowing Mamba to robustly capture long-range dependencies while overcoming spatial misalignments between frames.
- We introduce a residual redistribution (scatter) mechanism that updates all frames in a temporal window, maximizing efficiency and improving restoration consistency.
- We replace conventional forward-anchored propagation with a center-anchored propagation scheme that symmetrically leverages past and future frames, enabling robust alignment and global bidirectional information flow across the entire sequence.

2. Related Works

2.1. Video Super-Resolution

Video Super-Resolution (VSR) aims to recover high-resolution frames from low-resolution video by leveraging both spatial details and temporal redundancy. Unlike single-image SR, VSR must deal with motion, occlusion, and alignment—making it both more powerful and more challenging. Early models for VSR often adopted recurrent structures to handle the sequential nature of video. Recurrent neural networks (RNNs) [9] provided a natural way to propagate temporal information across frames, enabling temporal consistency and motion-aware enhancement. These models aimed to accumulate context over time, but were fundamentally limited by vanishing gradients and the inefficiency of sequential computation. As a result, they struggled to scale to longer sequences or high-resolution inputs. With the advent of Transformers [37], VSR models began to shift toward architectures that process multiple frames simultaneously. Models such as VSRT [3] and VRT [23] adopt self-attention to model spatiotemporal dependencies across frames, achieving strong performance through global context aggregation. However, the computational cost of attending to all tokens across multiple high-resolution frames is substantial, making such models difficult to scale in practice. To manage the high computational cost of full attention across multiple high-resolution frames, many recent models adopt a more practical alternative: propagating features across frames in a recurrent manner using attention or convolutional modules with limited temporal receptive fields. This strategy has become the dominant paradigm in modern VSR, with representative models including BasicVSR [4], BasicVSR++[5], RVRT[22], and IART [41], all leveraging frame-to-frame propagation to balance efficiency and performance. This paradigm has extended its influence on 3D super-resolution [20, 32], where maintaining multiview consistency across frames is a main challenge.

2.2. State Space Models

State Space Models (SSMs) [15] have long been used in control and signal processing to model temporal dynamics. In deep learning, structured SSMs have emerged as an alternative to RNNs and Transformers, aiming to combine long-range modeling with improved efficiency. The Structured State Space Sequence model (S4) [12] introduced a parameterized kernel derived from linear dynamics, enabling efficient long-range sequence modeling. Follow-up works such as S4D [13] and DSS [28] improved stability and generalization, but remained complex and hardware-unfriendly.

Mamba [11] simplified SSMs through a selective scan mechanism that enables input-dependent transitions, allowing content-aware information routing while maintaining
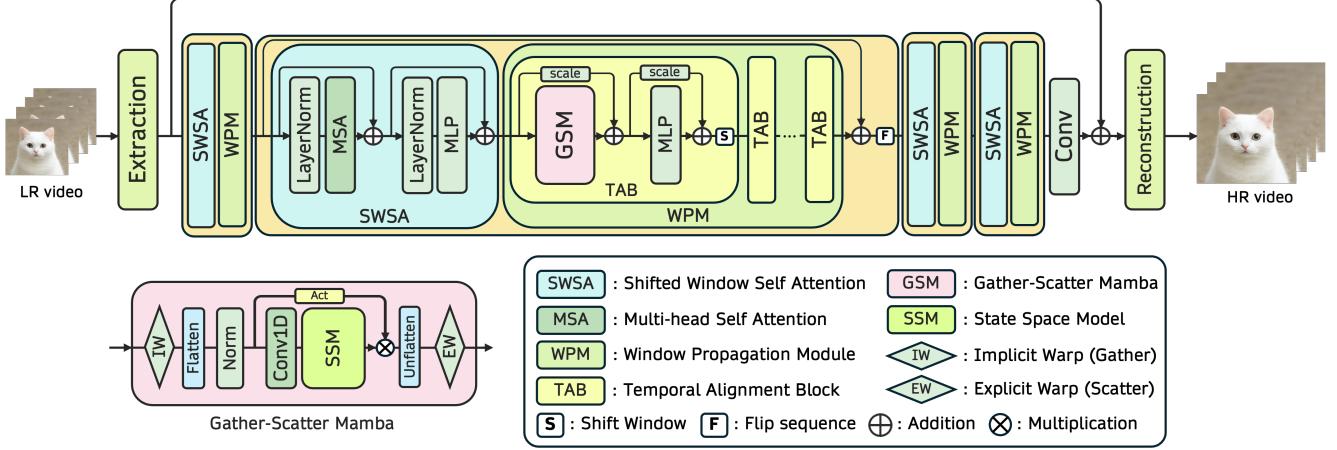
Figure 1. Overall architecture of the proposed Gather-Scatter Mamba (GSM). Given a low-resolution input sequence, local spatial refinement is first performed using shifted window self-attention (SWSA). Temporal propagation is then carried out by the window propagation module (WPM). Within each WPM, the GSM block first *gathers* features by aligning all frames to an anchor frame, processes the aligned features using Mamba's directionally selective scanning, and then *scatters* the updated features back to their original temporal locations.

linear complexity. This design supports key capabilities like associative recall and has inspired extensions in vision domains [10, 14, 16, 18, 21, 25, 36, 43], where adaptive spatiotemporal modeling is crucial.

3. Method

3.1. Preliminaries

State Space Models (SSMs) are grounded in continuous-time linear time-invariant (LTI) systems, and are traditionally described by the following set of ordinary differential equations (ODEs):

$$\frac{dh(t)}{dt} = Ah(t) + Bx(t), \quad y(t) = Ch(t) \quad (1)$$

where $A \in \mathbb{R}^{N \times N}$, $B \in \mathbb{R}^{N \times 1}$, and $C \in \mathbb{R}^{1 \times N}$. This formulation models the evolution of a hidden state $h(t)$ driven by an input signal $x(t)$, producing output $y(t)$.

To apply this system in discrete settings, Zero-Order Hold (ZOH) discretization is commonly used. This leads to the recurrence formulation:

$$h_k = \bar{A}h_{k-1} + \bar{B}x_k, \quad y_k = Ch_k \quad (2)$$

where the discretized matrices are defined as:

$$\bar{A} = \exp(\Delta A), \quad (3)$$
$$\bar{B} = (\Delta A)^{-1}(\exp(\Delta A) - I)\Delta B \quad (4)$$

Here, Δ denotes the timestep or step size, which controls how much the model forgets the previous hidden state and incorporates the current input.

The output can also be reformulated in terms of a convolutional kernel:

$$y = x * K, \quad \text{where } K = [C\bar{B}, C\bar{A}\bar{B}, \ldots, C\bar{A}^k\bar{B}] \quad (5)$$

Recent work [11] introduces Mamba, which challenges the limitations of SSMs with input-invariant parameters $\bar{A}, \bar{B}, C$. Instead, Mamba proposes a selective scan mechanism that allows these parameters to be dynamically generated from the input sequence, enabling fine-grained control over hidden state updates. Specifically, the model uses:

$$B = S_B(x), \quad C = S_C(x), \quad \Delta = \text{softmax}(\theta + S_\Delta(x)) \quad (6)$$

The discrete recurrence parameters are then recomputed as:

$$\bar{A}, \bar{B} \leftarrow \text{discretize}(\Delta, A, B) \quad (7)$$

By allowing the timestep Δ to be input-dependent, Mamba essentially introduces per-token forget gates, enabling richer and more selective information flow compared to static SSMs.

3.2. Overall Architecture

Given a low-resolution input video sequence $I_t^{\text{LR}} \in \mathbb{R}^{T \times H \times W \times C}$, the objective is to reconstruct the corresponding high-resolution sequence $I_t^{\text{SR}} \in \mathbb{R}^{T \times sH \times sW \times C}$, where T is the temporal length, H and W are the spatial dimensions, C is the number of channels, and s denotes the upsampling factor.

While image and video super-resolution share similar restoration goals, the key distinction lies in the temporal dimension. Propagating information across time is crucial
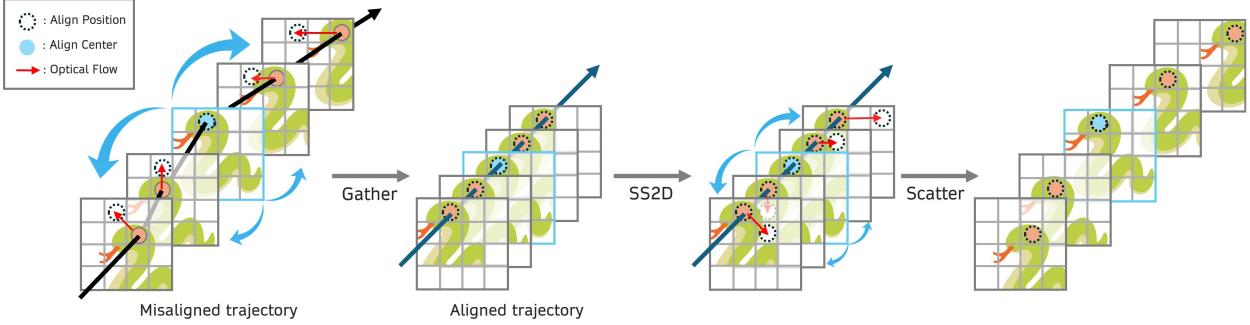
Figure 2. Overview of the proposed Gather-Scatter mechanism. Misaligned trajectories across frames are first temporally aligned toward the anchor frame (align center) using optical flow (Gather). The aligned features are then flattened in temporal-first order and processed by SS2D (Mamba) for long-range temporal modeling. Finally, the output residuals are inversely warped back to their original frame positions (Scatter), updating all frames within the window.

for video super-resolution, but it often incurs high computational cost. As a result, many existing methods rely on recurrent propagation schemes to balance performance and efficiency.

To address this, we propose a two-stage framework that explicitly decouples spatial refinement and temporal propagation. Our architecture first performs local feature refinement within each frame using shifted window self-attention (SWSA), which enables efficient modeling of intra-frame dependencies with limited receptive fields.

Following spatial refinement, we propagate information temporally using a window-based propagation module (WPM). Within each temporal window, we align all frames to a designated anchor frame through a gather-and-scatter mechanism. In the gather stage, each frame in the window is temporally aligned to the anchor, and the aligned tokens are flattened along the temporal axis. These tokens are then updated via Mamba's directionally selective scanning. In the scatter stage, the updated residuals are warped back to their original temporal locations and aggregated with the local features. This process is repeated by shifting the temporal window across the sequence in both forward and backward directions, enabling bidirectional propagation of temporal information throughout the entire video.

3.3. Mamba for Video Super-resolution

Recurrent architectures [9, 17] have been the dominant paradigm in video super-resolution [4, 5, 33, 35, 41], where each frame is treated as a timestep and features are propagated sequentially. However, such approaches are computationally expensive: the recurrent nature enforces strictly sequential propagation, preventing parallelization across frames, and the widespread use of Transformer-based feature extractors incurs quadratic complexity in spatiotemporal dimensions, making both training and inference prohibitively slow for long video sequences.

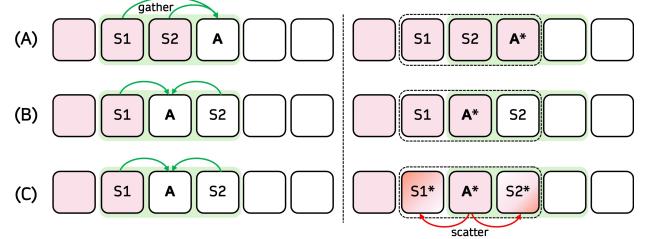To address these limitations, we introduce a Mamba-

Figure 3. Comparison of windowed propagation strategies. (A) *Forward-anchored propagation*: supporting frames **S1**, **S2** are aligned toward the anchor **A** located at the end of the window, and only the anchor is updated. (B) *Center-anchored propagation*: supporting frames are aligned toward the anchor placed at the center of the window, reducing alignment path length and improving feature aggregation. (C) *Center-anchored + Scatter (ours)*: residuals aligned at the center are redistributed back to all supporting frames, enabling joint enhancement of the entire window.

based framework for video super-resolution. Mamba is a structured state space model that enables efficient parallel processing with linear-time complexity, making it well-suited for long-range temporal modeling. To the best of our knowledge, this is the first work to replace temporal propagation with Mamba in the context of video super-resolution.

Whereas existing methods propagate features sequentially across frames, our Mamba-based framework processes the temporal dimension in parallel. Its linear-time design yields a much larger effective temporal receptive field than Transformer-based counterparts, and selective scanning ensures that information from distant frames is preserved and utilized effectively in restoration.

3.4. Gather-Scatter Mamba

Effective temporal propagation in video restoration critically depends on accurate alignment and efficient reuse of features across frames. Despite the recent successes of

Mamba-based architectures [6, 21, 44], directly applying them to video remains challenging due to their inherent sensitivity to spatial misalignment.

Existing Mamba-based video models predominantly adopt spatial-first scanning [21, 31, 40] rather than temporal-first scanning, as experiments consistently report inferior performance with the temporal-first approach. This outcome is somewhat counterintuitive, considering that capturing long-range temporal dependencies should theoretically help mitigate local misalignments. A plausible explanation lies in the structural characteristics of Mamba: since it processes video frames as flattened one-dimensional sequences, even minor spatial misalignments between frames translate into substantial positional distortions in the sequence dimension. Consequently, features that are spatially adjacent or semantically similar in 2D may become distant in the 1D sequence, severely hindering the model's ability to associate relevant information. Unlike Transformer-based architectures—which mitigate misalignment issues through local attention windows and flexible token association [33]—Mamba strictly relies on sequential adjacency [18].

A conventional approach to temporal feature propagation is forward-anchored propagation (fig. 3(A)), which aggregates information from previous frames via optical-flow-based alignment. Formally, given a video frame sequence $\{x_i\}$ and corresponding features f_i^j at propagation step j, the forward-anchored residual for frame i can be computed as:

$$r_i^j = \Phi\Big(f_i^{j-1},\ W(f_{i-1}^j,\ \mathcal{O}_{i\rightarrow i-1}),\ W(f_{i-2}^j,\ \mathcal{O}_{i\rightarrow i-2})\Big),$$
(8)

where $W(\cdot, \cdot)$ denotes backward warping of a supporting feature f_k toward the reference feature f_i using the optical flow $\mathcal{O}_{i\rightarrow k}$ estimated between frame i and frame k. Here, $\Phi(\cdot)$ denotes a feature fusion module, such as a Transformer or Mamba block, that integrates the aligned features and generates a residual update. This strategy, however, neglects useful future context available in subsequent frames.

To leverage bidirectional context, center-anchored propagation incorporates both past and future frames (fig. 3(B)):

$$r_i^j = \Phi\Big(f_i^{j-1},\ W(f_{i-1}^j,\ \mathcal{O}_{i\rightarrow i-1}),\ W(f_{i+1}^{j-1},\ \mathcal{O}_{i\rightarrow i+1})\Big).$$
(9)

Compared to forward-anchored propagation, which relies solely on distant past frames and thus suffers from large temporal gaps and weaker spatial overlap, center-anchored propagation mitigates this issue by symmetrically leveraging nearby past and future frames (Figure 4.) However, it still suffers from another inefficiency: the residuals generated from supporting frames are typically discarded after use, wasting significant computational resources.

To overcome the aforementioned limitations—including mamba's sensitivity to misalignment, suboptimal temporal

Figure 4. Occlusion comparison between forward-anchored and center-anchored approaches. The center-anchored strategy leverages information from adjacent frames, leading to fewer occluded regions. This reduction in occlusion enables more reliable reconstruction of the anchor frame and improves alignment quality.

scanning, and inefficient residual utilization—we propose Gather-Scatter Mamba, composed of two complementary steps:

Prior to Mamba processing, we employ optical-flow-based warping [30] to align features from neighboring frames to the reference frame, which is crucial because Mamba's 1D sequential scanning is highly sensitive to spatial misalignment (Gather phase):

$$\begin{aligned}
\hat{r}_k &= W(\hat{r}_{k\rightarrow i},\ \mathcal{O}_{k\rightarrow i}), \\
f_k^j &= f_k^{j-1} + \hat{r}_k, \\
k &\in \left\{ i - \tfrac{K-1}{2},\ \ldots,\ i + \tfrac{K-1}{2} \right\}.
\end{aligned}$$
(10)

The aligned features are then stacked into a 4D tensor and flattened along the temporal dimension to form a time-major 1D sequence suitable for Mamba processing:

$$\begin{aligned}
\mathbf{G}_i &\in \mathbb{R}^{K \times H \times W \times C}, \\
\tilde{\mathbf{G}}_i &= \mathrm{Flatten}_{\mathrm{temp}}(\mathbf{G}_i) \in \mathbb{R}^{(H \cdot W \cdot K) \times C}, \\
\hat{\mathbf{G}}_i &= \mathcal{M}(\tilde{\mathbf{G}}_i).
\end{aligned}$$
(11)

where $\mathcal{M}$ denotes the Mamba selective scan and K denotes the number of frames considered in the local temporal window.

The output sequence is then reshaped back to the spatio-temporal layout and split into per-frame residuals:

$$\{\hat{r}_{k\rightarrow i}\}_k = \mathrm{Reshape}_{\mathrm{temp}}^{-1}(\hat{\mathbf{G}}_i), \quad \hat{r}_{k\rightarrow i} \in \mathbb{R}^{H \times W \times C}.$$
(12)

Finally, the residuals are inversely warped (Scatter phase) and used to update the supporting frames:

$$\begin{aligned}
\hat{r}_k &= W(\hat{r}_{k\rightarrow i},\ \mathcal{O}_{k\rightarrow i}), \\
f_k^j &= f_k^{j-1} + \hat{r}_k, \\
k &\in \left\{ i - \tfrac{K-1}{2},\ \ldots,\ i + \tfrac{K-1}{2} \right\}.
\end{aligned}$$
(13)

This gather–scatter approach ensures spatial correspondences remain intact, significantly alleviating Mamba's in-

Method	Params (M)	FLOPs (T)	Frames REDS/Vimeo	REDS4		Vimeo-90K-T		Vid4	
				PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
TOFlow	-	-	5/7	27.98	0.7990	33.08	0.9054	25.89	0.7651
EDVR	20.6	2.95	5/7	31.09	0.8800	37.61	0.9489	27.35	0.8264
VSR-T	32.6	-	5/7	31.19	0.8815	37.71	0.9494	27.36	0.8258
BasicVSR	6.3	0.33	15/14	31.42	0.8909	37.18	0.9450	27.24	0.8251
IconVSR	8.7	0.51	15/14	31.67	0.8948	37.47	0.9476	27.39	0.8279
BasicVSR++	7.3	0.39	16/14	32.13	0.8990	37.79	0.9500	27.79	0.8400
VRT	30.7	1.37	16/7	32.19	0.9006	<u>38.20</u>	<u>0.9530</u>	27.93	0.8425
RVRT	10.8	2.21	16/14	32.53	0.9078	38.15	0.9527	27.89	<u>0.8482</u>
IART	13.4	2.51	16/14	**32.90**	**0.9138**	38.14	0.9528	**28.26**	**0.8517**
GSMamba	10.5	1.52	16/14	<u>32.69</u>	<u>0.9105</u>	**38.25**	**0.9534**	<u>28.03</u>	0.8462

Table 1. Quantitative comparison with the state-of-the-art methods on REDS4 [29], Vimeo-90K-T [42], and Vid4 [24] datasets for 4× upsampling. Our GSMamba achieves state-of-the-art performance on Vimeo-90K-T and competitive results on REDS4 and Vid4, while using fewer parameters and FLOPs than existing methods.

herent sensitivity to misalignment. Moreover, residual reuse (scatter) efficiently recycles valuable intermediate computations, enriching the feature representations of supporting frames for subsequent propagation stages. After the scatter phase, the anchor frame index is shifted to $i + 1$ and the process is repeated, allowing information to propagate across all frames in a sliding-window manner.

4. Experiments

4.1. Dataset

Following prior works [4, 5, 22, 41], we use REDS [29] and Vimeo-90K [42] for training, and evaluate on REDS4, Vimeo-90K-T, and Vid4 [24]. REDS provides ×4 bicubic downsampled frames with 1280×720 resolution, which we directly use for training. Vimeo-90K consists of 448×256 resolution 7-frame clips; we generate ×4 bicubic low-resolution inputs using the MATLAB function from BasicVSR [4] for compatibility. For Vid4, we evaluate the Vimeo-90K-trained model without fine-tuning, following standard protocol [2].

4.2. Experiment Settings

For Vimeo-90K [42], we use 14-frame sequences by reversing and concatenating the original 7-frame clips. For REDS [29], we extract 16-frame clips from the original 100-frame sequences. Optical flow is estimated using SpyNet [30], where the network is frozen for the first 5,000 iterations and then jointly trained with a learning rate scaled to $0.125\times$ of the main model. For feature alignment and resampling, we adopt the implicit alignment module from IART [41]. We train our model using the Adam optimizer [19] with β_1=0.9, β_2=0.99 and the Cosine Annealing Scheduler [27]. The initial learning rate is set to $2 \times$

Method	Param. (M)	FLOPs (T)	Runtime (ms)	PSNR (dB)
VRT	35.6	1.37	1394	32.19
RVRT	10.8	2.21	743*	32.53
IART	13.4	2.51	1703	32.90
GSMamba	10.5	1.52	1070	32.69

Table 2. Comparison of model parameters, FLOPs, runtime, and PSNR. Our GSMamba has lower parameter count and FLOPs, achieves shorter inference time, and delivers competitive PSNR compared with other VSR methods. (*) Runtime of RVRT is measured with custom CUDA kernels provided by the authors.

10^{-4}, and we use a mini-batch size of 8. Our training is conducted in two stages. We first train on REDS for 600,000 iterations using 8 NVIDIA H100 GPUs, and directly use this model for REDS evaluation. The REDS-trained weights are then used to initialize training on Vimeo-90K, where we train for an additional 300,000 iterations using 8 NVIDIA V100 GPUs, and use the resulting model for evaluation on Vimeo-90K-T. Our model configuration is as follows: the embedding dimension is 192 with a propagation depth of four stages. Each stage contains two shifted-window self-attention (SWSA) blocks (with and without shift) followed by two GSM blocks. SWSA uses 8 attention heads with a window size of $(2, 8, 8)$. The window propagation module (WPM) operates on a temporal window of 5 frames ($K = 5$, two past, anchor, two future). For the SS2D component of GSM, the state dimension d_{state} is set to 16.

Alignment	Scanning	Anchor	Scatter	PSNR	SSIM
Align ✗	Spatial-first	–	–	30.70	0.8716
	3D Hilbert	–	–	30.68	0.8710
	Temporal-first	–	–	30.55	0.8666
Align o	Temporal-first	Forward	X	31.74	0.8942
		Center	X	31.83	0.8950
		Center	O	**31.93**	**0.8957**

Table 3. Ablation of scanning and alignment strategies. Scanning defines the 1D token ordering for Mamba, while Anchor indicates temporal alignment (Forward/Center) within each window. The last three rows correspond to the configurations visualized in Fig. 3 (A–C).
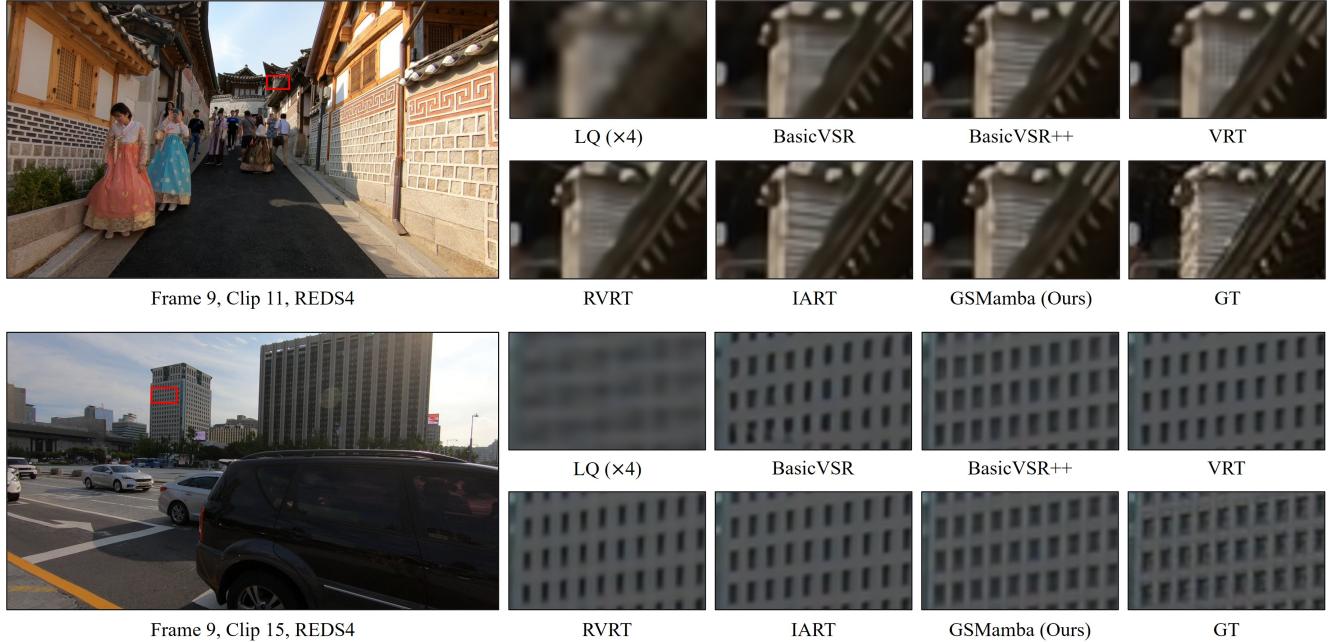

Figure 5. Qualitative results with the state-of-the-art methods on REDS4 [29] dataset

5. Results

5.1. Quantitative Results

We compare PSNR and SSIM metrics on three benchmark datasets against representative VSR baselines, including TOFlow [42], EDVR [39], VSR-T [3], BasicVSR, IconVSR [4], BasicVSR++ [5], VRT [23], RVRT [22], and IART [41]. The results are summarized in Table 1. Our method achieves state-of-the-art performance on the Vimeo-90K-T dataset and competitive performance on the REDS4 and Vid4 datasets. Combined with the results in Table 2, these findings demonstrate that our approach attains strong reconstruction quality while maintaining high efficiency in terms of parameters and FLOPs.

5.2. Qualitative Results

We further provide qualitative comparisons on the REDS4 and Vimeo-90K-T datasets to visually assess the reconstruction quality of our method. We compare against strong baselines, including BasicVSR, BasicVSR++, VRT, RVRT, and IART. Representative visual results are shown in Figure 6. Our method produces sharper textures and fewer artifacts, demonstrating superior perceptual quality compared with existing approaches.

5.3. Model Efficiency

Table 2 compares the number of parameters, FLOPs, and runtime of our method with other VSR baselines. Our model has a lower parameter count, lower FLOPs, and faster runtime compared with existing methods. The experiments are conducted on an RTX 3090 GPU using low-
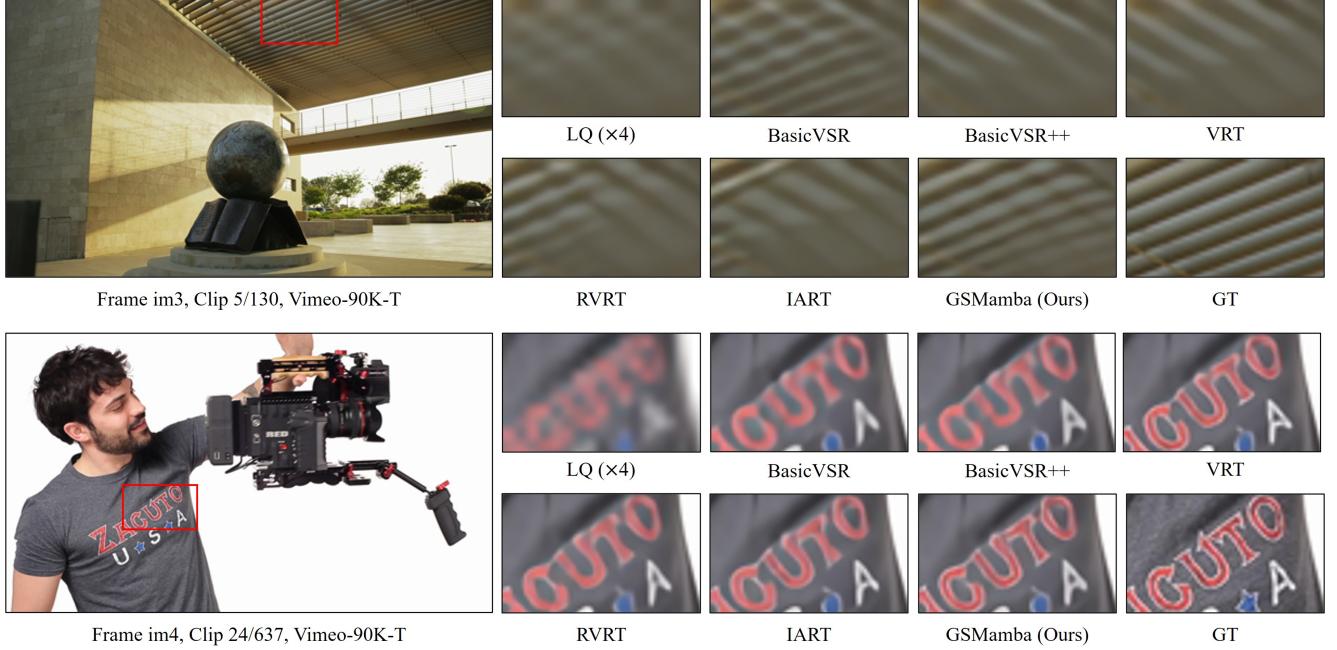
Figure 6. Qualitative results with the state-of-the-art methods on Vimeo-90K-T [42] dataset

quality inputs of $16 \times 320 \times 180$ from the REDS dataset.

6. Ablation Study

We conduct ablation studies on the REDS dataset to evaluate the impact of scanning order, the scatter mechanism, and anchor selection in our Gather-Scatter Mamba (GSM). For all ablation training, we extract 6 consecutive frames and use explicit bilinear resampling for alignment during the gather stage (instead of the implicit warping module [41] used in the main experiments.) The results are summarized in Table 3.

Scanning Strategy. Temporal-first scanning yields the best performance among different token orderings. This result highlights that when using Mamba for temporal propagation, explicit alignment becomes necessary and arranging tokens along the temporal axis enables Mamba to more effectively capture long-range dependencies.

Scatter Mechanism. Finally, enabling the scatter phase—which redistributes the Mamba outputs back to their original temporal locations—yields additional PSNR and SSIM improvements. This confirms that jointly updating all frames within a window leads to more consistent temporal propagation compared to updating only the anchor frame.

Anchor Selection. Center-anchored alignment consistently outperforms forward-anchored alignment. Center anchoring provides two advantages: (i) it reduces occlusions, allowing more nearby frames to contribute to reconstruction, and (ii) it minimizes warping errors due to shorter motion paths, resulting in improved alignment and overall reconstruction quality.

7. Conclusion

In this work, we propose Gather-Scatter Mamba (GSM), a novel video super-resolution framework that integrates Mamba-based temporal propagation with alignment-aware residual redistribution. Our gather-scatter design ensures that all frames within a temporal window are enhanced jointly, improving both efficiency and consistency. Extensive experiments on REDS, Vimeo-90K, and Vid4 demonstrate that GSMamba achieves competitive or superior performance to state-of-the-art methods while requiring fewer parameters and FLOPs. Moreover, our ablation studies highlight the importance of center-anchored alignment and residual scattering, both of which significantly contribute to the final performance. Our results suggest that structured state-space models such as Mamba are a promising alternative to recurrent or attention-based propagation for video restoration tasks, combining scalability, efficiency, and strong temporal modeling ability.

References

[1] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994. 1

[2] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4778–4787, 2017. 6

[3] Jiezhang Cao, Yawei Li, Kai Zhang, and Luc Van Gool. Video super-resolution transformer. *arXiv preprint arXiv:2106.06847*, 2021. 2, 7

[4] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential components in video super-resolution and beyond. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4947–4956, 2021. 1, 2, 4, 6, 7

[5] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5972–5981, 2022. 1, 2, 4, 6, 7

[6] 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. 5

[7] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359, 2022. 1

[8] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015. 1

[9] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990. 1, 2, 4

[10] Yu Gao, Jiancheng Huang, Xiaopeng Sun, Zequn Jie, Yujie Zhong, and Lin Ma. Matten: Video generation with mamba-attention. *arXiv preprint arXiv:2405.03025*, 2024. 3

[11] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 1, 2, 3

[12] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 1, 2

[13] Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022. 2

[14] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. In *European conference on computer vision*, pages 222–241. Springer, 2024. 2, 3

[15] James D Hamilton. State-space models. *Handbook of econometrics*, 4:3039–3080, 1994. 2

[16] Ali Hatamizadeh and Jan Kautz. Mambavision: A hybrid mamba-transformer vision backbone. *arXiv preprint arXiv:2407.08083*, 2024. 3

[17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 4

[18] 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. 3, 5

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[20] Hyun-kyu Ko, Dongheok Park, Youngin Park, Byeonghyeon Lee, Juhee Han, and Eunbyung Park. Sequence matters: Harnessing video models in 3d super-resolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4356–4364, 2025. 2

[21] 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, 2024. 2, 3, 5

[22] Jingyun Liang, Yuchen Fan, Xiaoyu Xiang, Rakesh Ranjan, Eddy Ilg, Simon Green, Jiezhang Cao, Kai Zhang, Radu Timofte, and Luc V Gool. Recurrent video restoration transformer with guided deformable attention. *Advances in Neural Information Processing Systems*, 35:378–393, 2022. 1, 2, 6, 7

[23] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. Vrt: A video restoration transformer. *IEEE Transactions on Image Processing*, 2024. 2, 7

[24] Ce Liu and Deqing Sun. On bayesian adaptive video super resolution. *IEEE transactions on pattern analysis and machine intelligence*, 36(2):346–360, 2013. 6

[25] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Advances in neural information processing systems*, 37:103031–103063, 2024. 2, 3

[26] 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. 1

[27] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[28] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. *arXiv preprint arXiv:2206.13947*, 2022. 2

[29] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee.

Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019. 6, 7

[30] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017. 5, 6

[31] Weiming Ren, Wentao Ma, Huan Yang, Cong Wei, Ge Zhang, and Wenhu Chen. Vamba: Understanding hour-long videos with hybrid mamba-transformers. *arXiv preprint arXiv:2503.11579*, 2025. 5

[32] Yuan Shen, Duygu Ceylan, Paul Guerrero, Zexiang Xu, Niloy J Mitra, Shenlong Wang, and Anna Frühstück. Supergaussian: Repurposing video models for 3d super resolution. In *European Conference on Computer Vision*, pages 215–233. Springer, 2024. 2

[33] Shuwei Shi, Jinjin Gu, Liangbin Xie, Xintao Wang, Yujiu Yang, and Chao Dong. Rethinking alignment in video super-resolution transformers. *Advances in Neural Information Processing Systems*, 35:36081–36093, 2022. 1, 4, 5

[34] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852. PMLR, 2015. 1

[35] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *Proceedings of the IEEE international conference on computer vision*, pages 4472–4480, 2017. 4

[36] Yao Teng, Yue Wu, Han Shi, Xuefei Ning, Guohao Dai, Yu Wang, Zhenguo Li, and Xihui Liu. Dim: Diffusion mamba for efficient high-resolution image synthesis. *arXiv preprint arXiv:2405.14224*, 2024. 3

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1, 2

[38] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015. 1

[39] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 0–0, 2019. 7

[40] Hongtao Wu, Yijun Yang, Huihui Xu, Weiming Wang, Jinni Zhou, and Lei Zhu. Rainmamba: Enhanced locality learning with state space models for video deraining. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7881–7890, 2024. 5

[41] Kai Xu, Ziwei Yu, Xin Wang, Michael Bi Mi, and Angela Yao. Enhancing video super-resolution via implicit resampling-based alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2546–2555, 2024. 1, 2, 4, 6, 7, 8

[42] 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, 7, 8

[43] 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

[44] Yijun Yang, Zhaohu Xing, Lequan Yu, Chunwang Huang, Huazhu Fu, and Lei Zhu. Vivim: A video vision mamba for medical video segmentation. *arXiv preprint arXiv:2401.14168*, 2024. 5

[45] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015. 1

[46] Jingwei Zhang, Anh Tien Nguyen, Xi Han, Vincent Quoc-Huy Trinh, Hong Qin, Dimitris Samaras, and Mahdi S Hosseini. 2dmamba: Efficient state space model for image representation with applications on giga-pixel whole slide image classification. *arXiv preprint arXiv:2412.00678*, 2024. 2