

VMRNN: Integrating Vision Mamba and LSTM for Efficient and Accurate Spatiotemporal Forecasting

Yujin Tang¹ Peijie Dong² Zhenheng Tang^{2,3} Xiaowen Chu² Junwei Liang^{1,4} *

¹AI Thrust, The Hong Kong University of Science and Technology (Guangzhou)

²DSA Thrust, The Hong Kong University of Science and Technology (Guangzhou)

³Department of Computer Science, Hong Kong Baptist University

⁴Department of Computer Science and Engineering, The Hong Kong University of Science and Technology

tangyujin0275@gmail.com, junweiliang@hkust-gz.edu.cn

Abstract

Combining Convolutional Neural Networks (CNNs) or Vision Transformers (ViTs) with Recurrent Neural Networks (RNNs) for spatiotemporal forecasting has yielded unparalleled results in predicting temporal and spatial dynamics. However, modeling extensive global information remains a formidable challenge; CNNs are limited by their narrow receptive fields, and ViTs struggle with the intensive computational demands of their attention mechanisms. The emergence of recent Mamba-based architectures has been met with enthusiasm for their exceptional long-sequence modeling capabilities, surpassing established vision models in efficiency and accuracy, which motivates us to develop an innovative architecture tailored for spatiotemporal forecasting. In this paper, we propose the VMRNN cell, a new recurrent unit that integrates the strengths of Vision Mamba blocks with LSTM. We construct a network centered on VMRNN cells to tackle spatiotemporal prediction tasks effectively. Our extensive evaluations show that our proposed approach secures competitive results on a variety of tasks while maintaining a smaller model size. Our code is available at <https://github.com/yuyujintang/VMRNN-PyTorch>.

1. Introduction

In recent years, spatiotemporal prediction has experienced a surge in interest due to its potential to enhance a wide range of practical applications. These applications span from precipitation forecasting [41, 42, 49, 54], autonomous driving [1, 25, 28], traffic flow prediction [59, 62], and human motion forecasting [51, 63] to representation learning [20, 38]. The ability to accurately predict spatial and temporal variations holds immense promise for improv-

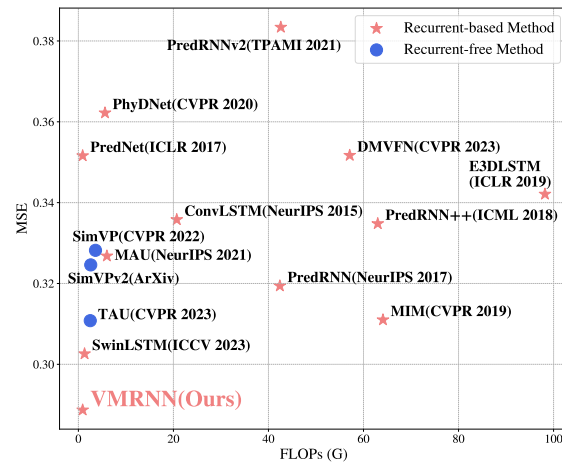


Figure 1. Performance comparison on TaxiBJ over spatial-temporal predictive learning methods. VMRNN outperforms previous methods in terms of Mean-Squared-Error (MSE, lower the better) with a lower computational cost (GFLOPs).

ing decision-making processes and operational efficiencies across diverse sectors, underscoring the importance of continued research and development in spatiotemporal analysis. The complex physical interactions and the unpredictable characteristics of spatiotemporal data present significant obstacles for solely data-driven deep learning approaches to attain accurate predictions. The essence of spatiotemporal predictive learning lies in its capability to delve into the spatial correlations and temporal progressions inherent in the physical realm, highlighting its potential to uncover deep insights into the dynamics of our world.

To address these challenges, a plethora of methodologies have been developed, including recurrent-based methods

*Corresponding author

which combine Convolutional Neural Networks (CNNs) or Vision Transformers (ViTs) with Recurrent Neural Networks (RNNs) [3, 14, 27, 41, 48, 52–54, 56, 61] and recurrent-free methods like SimVP [7, 45], which fully based on CNN. As for recurrent-based methods, lots of innovative RNNs are proposed. Among these, ConvLSTM [41] represents a pivotal advancement by augmenting the fully connected LSTM with convolutional operations to simultaneously capture spatial and temporal dependencies. Building upon ConvLSTM, a variety of innovative approaches have emerged. PredRNN [54] and MIM [56], for instance, refine the LSTM unit’s internal mechanics, while E3D-LSTM [53] introduces 3D convolutions into LSTM structures. PhyDNet utilizes a CNN-based approach to untangle physical dynamics, and MAU [3] introduces a motion-aware unit for enhanced motion capture. Recent recurrent-free models, like SimVP [7], and the Temporal Attention Unit (TAU) [46], which bifurcates temporal attention into static intra-frame and dynamic inter-frame components, offer fresh perspectives on spatiotemporal modeling. Furthermore, advancements such as the Dynamic Multi-scale Voxel Flow Network (DMVFN) [19] and the two-stream MMVP [19] framework underscore the recent innovation in this field, emphasizing the separation of motion and appearance for improved prediction. SwinLSTM [48] successfully integrates Swin Transformer [31] with LSTM which stands out as a strong spatiotemporal prediction baseline.

While these approaches have demonstrated notable success in spatiotemporal forecasting, CNNs are intrinsically constrained by their local receptive fields [33], limiting their capacity to assimilate information from distant image regions. ViTs generally exhibit superior performance compared to CNNs, which could be attributed to global receptive fields and dynamic weights facilitated by the attention mechanism. However, the attention mechanism requires quadratic complexity in terms of image sizes, resulting in expensive computational overhead when addressing downstream dense prediction tasks.

Recently, Structured State Space Models (S4) [12, 13] have emerged as notably efficient and effective in modeling extensive sequences. Mamba [9] positions itself as a potential breakthrough for addressing long-range dependencies in various tasks, which innovatively introduces the selectivity of the input sequence and uses the scan method. Unlike transformers, which often exhibit quadratic scaling for sequence length, Mamba maintains a linear or near-linear scaling, all the while adeptly handling long-range dependencies. This attribute has catapulted them to the forefront of continuous long-sequence data analysis, achieving state-of-the-art results in fields such as natural language processing and genomic analysis. A series of pioneering studies [8, 15, 16, 29, 30, 34, 58, 60, 65] have begun to investigate the utility of Mamba models within the vision

sector, showcasing their versatility across a broad spectrum of vision-based applications. These investigations into Mamba’s application—ranging from basic image recognition to complex segmentation tasks—have yielded encouraging outcomes, underscoring the significant potential that SSMs hold in revolutionizing vision tasks.

Inspired by these studies, we introduce VMRNN, an innovative recurrent cell that merges Vision Mamba blocks (VSS Block) with an LSTM module to effectively distill spatiotemporal representations. Furthermore, we develop a model centered around VMRNN, specifically engineered to discern both spatial and temporal dynamics crucial for spatiotemporal forecasting. Unlike previous image-level vision tasks, spatiotemporal predictive learning predicts future frames from past frames at the video level. Our model processes each frame at the image level, segments them into patches, and flattens these patches before passing them to the patch embedding layer for preliminary processing. Our method inherits the attribute of the recurrent-based methods. The VMRNN layer utilizes these transformed patches with previous states to capture spatiotemporal representations for the next prediction.

The contributions of this research are threefold:

- We introduce VMRNN, an innovative recurrent cell that fuses an LSTM architecture with Vision Mamba Blocks. To the best of our knowledge, we are the first to introduce Mamba into vision-based spatial-temporal forecasting to harness robust sequence modeling prowess.
- We propose two new architectures based on VMRNN, VMRNN-B, and VMRNN-D, excelling in extracting spatiotemporal representations and providing a new strong baseline for spatiotemporal forecasting.
- Extensive evaluation on Moving MNIST, TaxiBJ, and KTH demonstrates that our VMRNN not only shows a significant reduction in computational complexity and parameters but also matches or surpasses SOTA methods on all three datasets across metrics, validating its efficacy on three pivotal datasets.

2. Related Work

2.1. Convolution-based Architecture

Previous models that merge CNNs with RNNs adopt a range of tactics to more effectively grasp the nuances of spatiotemporal relationships, aiming to enhance predictive precision. ConvLSTM [41] evolves from FC-LSTM [44] by integrating convolutional operations instead of fully connected ones, facilitating the learning of spatiotemporal interdependencies. PredRNN [54] and its Spatiotemporal LSTM (ST-LSTM) unit mark a significant step, enabling the

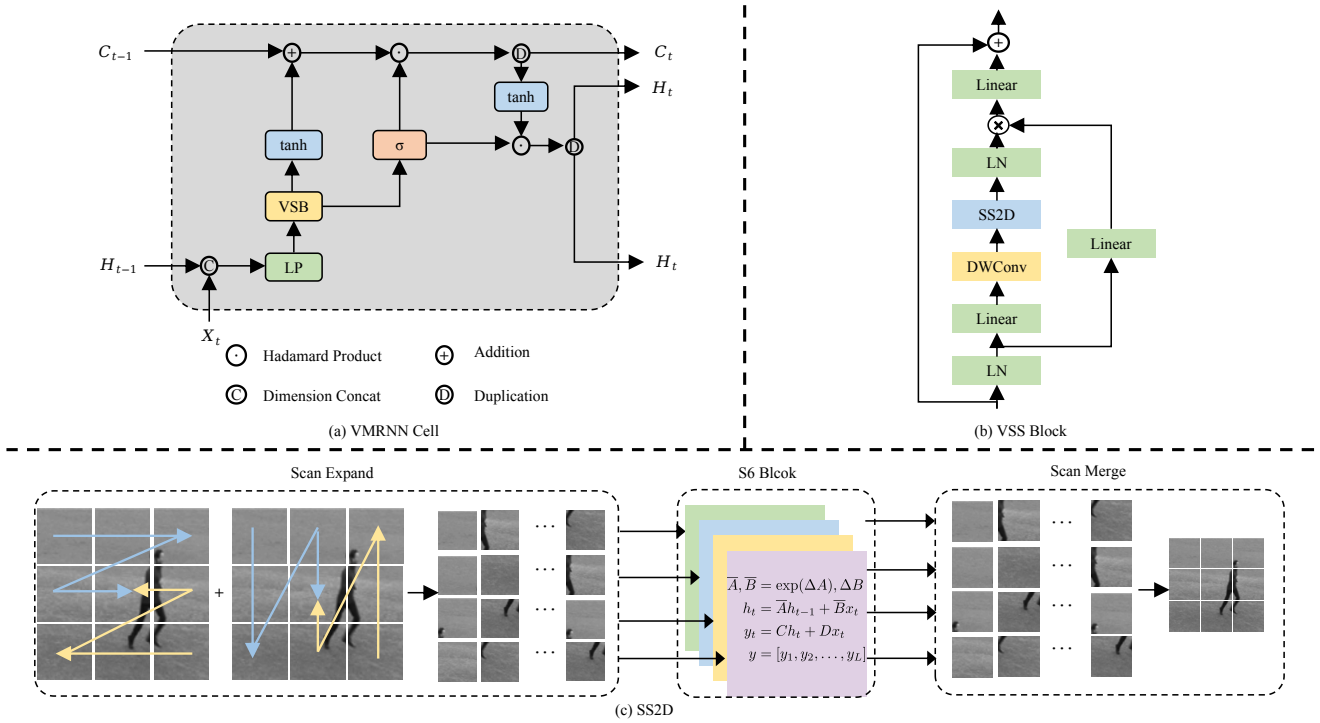


Figure 2. (a): The detailed structure of the proposed recurrent cell: VMRNN. **VSB** and **LP** denote VSS Block and Linear Projection. (b): The architecture of VSS Block. (c): The SS2D process, includes three stages: Scan Expand, S6 Block, and Scan Merge.

concurrent processing of spatiotemporal data by propagating hidden states both horizontally and vertically. Building on this, PredRNN++ [52] contributes the Gradient Highway unit to mitigate the vanishing gradient issue encountered by its predecessor. Meanwhile, E3D-LSTM [53] enhances the ST-LSTM’s memory capacity by implementing 3D convolutions. The MIM model [56] reimagines the ST-LSTM’s forget gate with dual recurrent units to better address the learning of non-stationary information within predictions. CrevNet [61] employs a CNN-based reversible architecture to decode complex spatiotemporal patterns. Additionally, PhyDNet [14] embeds physical principles into CNN frameworks to refine the quality of its predictions. Collectively, these models [14, 41, 52–54, 56, 61] showcase a variety of approaches to enhance the capture of spatiotemporal dependencies and have garnered commendable outcomes. Nonetheless, conventional convolutional methodologies are constrained in their ability to seize spatiotemporal dependencies due to their intrinsic localized operation.

2.2. Transformer-based Architecture

The adoption of the Transformer model [50], initially celebrated in natural language processing, has prompted its exploration within the realm of computer vision. The Vision Transformer (ViT) [4] broke new ground by directly applying Transformer architecture to image classification,

demonstrating impressive results. Further advancing this domain, the Swin Transformer [31] delivers remarkable achievements across a spectrum of tasks such as image classification, semantic segmentation, and object detection, thanks to its innovative shifted window strategy and hierarchical structure. Building on this, SwinLSTM [48] innovatively merges the Swin Transformer [31] with LSTM, establishing a new robust benchmark for spatiotemporal forecasting. However, ViT and its derivatives exhibit a notable drawback: the attention mechanism’s quadratic complexity in relation to image size, which imposes considerable computational demands.

2.3. State Space Models

State Space Models(SSMs) are recently proposed models that are introduced into deep learning as state space transforming [12, 13, 43]. Inspired by continuous state space models in control systems, combined with HiPPO [10] initialization, LSSL [13] showcases the potential to handle long-range dependency problems. However, due to the prohibitive computation and memory requirements induced by the state representation, LSSL is infeasible to use in practice. To solve this problem, S4 [12] proposes to normalize the parameter into the diagonal structure. Since then, many flavors of structured state space models sprang up with different structures like complex-diagonal structure [11, 17],

multiple-input multiple-output supporting [43], decomposition of diagonal plus low-rank operations [18], selection mechanism [9]. These models are then integrated into large representation models [5, 35, 36]. Among these developments, Mamba [9] proposes the selective scan space state sequential model (S6) Block, which stands out as a promising innovation for tackling long-range dependencies across a spectrum of tasks. It introduces a novel approach by selectively processing the input sequence and employing a scanning method, marking a potential breakthrough in the field.

Several latest studies have preliminarily explored the effectiveness of Mamba in the vision domain. For instance, Vim [65] proposed a generic vision backbones with bidirectional Mamba blocks. In contrast, VMamba [30] builds up a Mamba-based vision backbone with hierarchical representations. Additionally, VMamba introduced a cross-scan module to solve the direction-sensitive problem due to the difference between 1D sequences and 2D images. In this paper, we try to integrate the Vision Mamba blocks proposed in VMamba with the simplified LSTM to form a VMRNN recurrent cell and use it as the core to build a model to capture temporal and spatial dependencies to perform spatiotemporal prediction tasks.

3. Method

3.1. Overall Architecture

The architecture of our predictive model is illustrated in Fig. 3 (a) and (b). Following the framework presented in [48], we introduce a base model and a deeper model centered on the VMRNN Cell, denoted as VMRNN-B and VMRNN-D, respectively. As shown in Fig. 3 (a), at each time step, the image is divided into non-overlapping patches of size P^2 with patch size P . And then these image patches are flattened and sent into the patch embedding layer, which performs a linear transformation of the patches' original features into a specified dimensional space.

For the VMRNN-B model, the VMRNN layer processes the embedded image patches, along with the previous time step's hidden state H_{t-1} and cell state C_{t-1} , to generate the current hidden state H_t and cell state C_t . As illustrated in Fig. 2(a), H_t is replicated, producing two versions: one is directed to the reconstruction layer, and the other, in conjunction with C_t , serves the VMRNN layer for the subsequent time step. For VMRNN-B, the architecture primarily relies on the stacking of VMRNN layers. For the VMRNN-D variant, we incorporate more VMRNN Cells and introduce Patch Merging and Patch Expanding layers, as outlined in [2]. The Patch Merging layer is employed for downsampling, effectively reducing the spatial dimensions of the data, which aids in reducing computational complexity and capturing more abstract, global features. Conversely, the Patch Expanding layer is used for upsam-

pling, which increases the spatial dimensions, facilitating the restoration of detail and enabling precise localization of features in the reconstruction phase. Ultimately, the reconstruction layer takes the hidden state H_t from the VMRNN layer and scales it back to the input size, generating the predicted frame for the next time step.

The integration of downsampling and upsampling processes presents significant advantages in our predictive architecture. Downsampling simplifies the input representation, allowing the model to process higher-level features with reduced computational overhead. This is particularly beneficial for understanding complex patterns and relationships within the data at a more abstract level. Upsampling, on the other hand, ensures that the detailed spatial information is not lost. This balance between abstraction and detail preservation is key to achieving high-quality predictions, especially in tasks requiring fine-grained understanding and visual data generation.

3.2. VMRNN Module

VMRNN Module removes all weights W and biases b in ConvLSTM [41] to obtain Eqn. 3:

$$i_t = f_t = o_t = \sigma(X_t + H_{t-1}) \quad (1)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tanh(X_t + H_{t-1}) \quad (2)$$

$$H_t = o_t \odot \tanh(C_t) \quad (3)$$

We propose the VMRNN Module, detailed in Fig. 2 (a). In VMRNN, the long-term and short-term temporal dependencies are captured by updating the information of cell states C_t , and hidden states H_t are updated from a horizontal perspective. And the VSS Blocks vertically learn spatial dependencies. We show the key equations of VMRNN in Eqn. 6, where VSB means the VSS Blocks in Sec. 3.3 and LP is short for the Linear Projection:

$$F_t = \sigma(\text{VSB}(\text{LP}(X_t; H_{t-1}))) \quad (4)$$

$$C_t = F_t \odot (\tanh(\text{VSB}(\text{LP}(X_t; H_{t-1})))) + C_{t-1} \quad (5)$$

$$H_t = F_t \odot \tanh(C_t) \quad (6)$$

3.3. VSS Block

The structure of VSS block is illustrated in Fig. 2 (b). The process begins with the input being processed through an initial linear embedding layer, which is then split into two distinct information streams. The first stream is channeled through a 3×3 depth-wise convolution layer, enriched with a Silu activation function [40] before it progresses into the SS2D module. After SS2D, its output is refined by a layer normalization process, and subsequently, it merges with the second stream's output, which has been previously

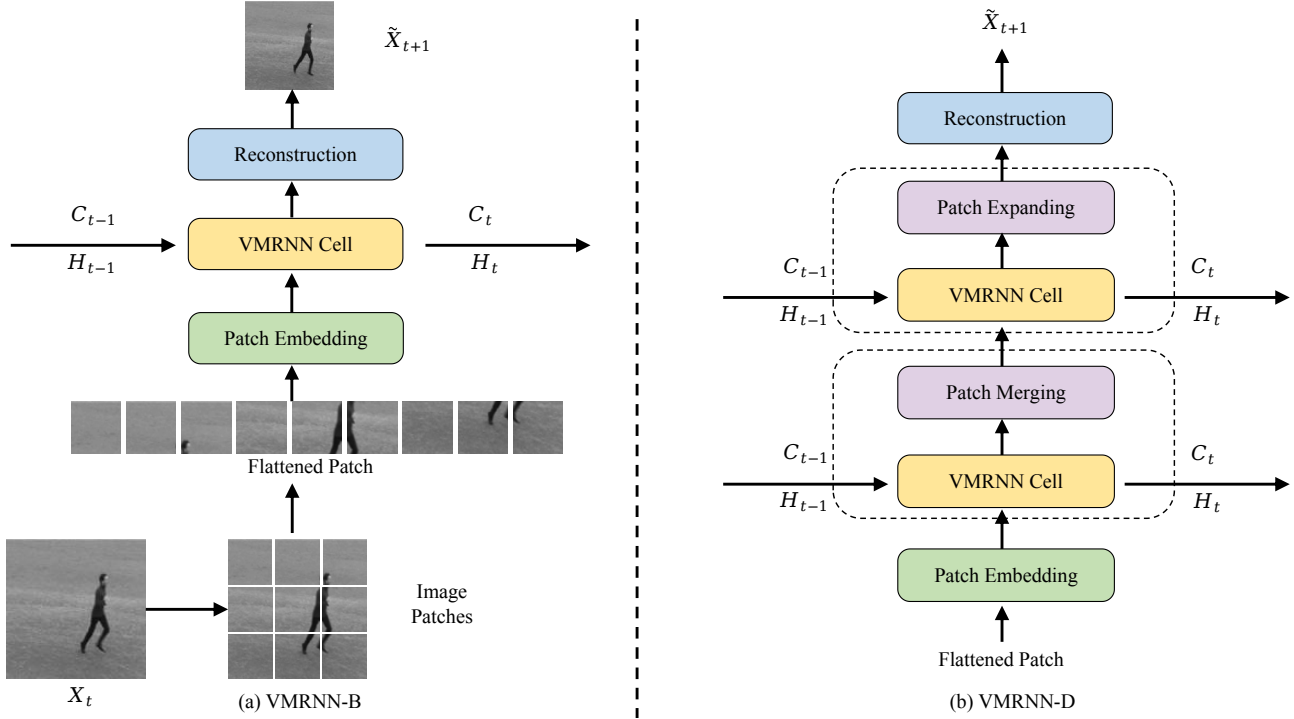


Figure 3. (a): The architecture of the base model with a single VMRNN cell, VMRNN-B. (b): The architecture of the deep model with multiple VMRNN cells, VMRNN-D.

activated by Silu. This combination produces the final output of the VSS block. The architecture takes a novel path compared to vision transformer design, which typically follows a Norm \rightarrow attention \rightarrow Norm \rightarrow MLP sequence within a block, and omits the MLP stage. This deviation renders the VSS block less complex than the ViT block, enabling the incorporation of a greater number of blocks within a comparable total model depth constraint.

VSS block first recovers linear projections to the image shape. (From [B, L, C] to [B, H, W, C]). Then VSS block addresses the challenges associated with 2D image data by employing 2D-selective-scan (SS2D), as illustrated in Fig. 2 (c). This approach unfolds image patches in four distinct directions: from the top-left to the bottom-right, from the bottom-right to the top-left, from the top-right to the bottom-left, and from the bottom-left to the top-right, creating four distinct sequences, as depicted in the Scan Expand Stage. Subsequently, each feature sequence(scan) will be processed through the S6 Block. Finally, these sequences are reconfigured back into individual images, as depicted in the Scan Merge Stage. Given input feature z , the output feature \bar{z} of SS2D can be written as:

$$z_v = \text{expand}(z, v) \quad (7)$$

$$\bar{z}_v = S6(z_v) \quad (8)$$

$$\bar{z} = \text{merge}(\bar{z}_1, \bar{z}_2, \bar{z}_3, \bar{z}_4) \quad (9)$$

where $v \in V = \{1, 2, 3, 4\}$ is four different scanning directions. $\text{expand}(\cdot)$ and $\text{merge}(\cdot)$ corresponding to the *scan expand* and *scan merge* operations. The selective scan space state sequential model (S6) in Eqn. 8 is the core SSM operator of the VSS block. It enables each element in a 1D array to interact with any of the previously scanned samples through a compressed hidden state. We plot the equations of S6 process in Fig. 2 (c). in S6 Block stage.

4. Experiments

4.1. Implementations

We employ the Mean Squared Error (MSE) loss function across all three datasets. For KTH [39] and TaxiBJ [62], our methodology aligns with OpenSTL [47]. For the Moving MNIST [44] dataset, we adhere to the experimental setup detailed in [48]. The precise model parameters, hyper-parameters(including batch size, learning rate, and training epochs), and training machines utilized for each dataset are comprehensively enumerated in Table 1. For TaxiBJ, we train 200 epochs with a learning rate of 4e-4 with a single A6000 GPU, using a batch size of 16. For KTH, we train 100 epochs with a learning rate of 5e-4 and 1e-4 for KTH20 and KTH40, respectively. For the Moving MNIST dataset, we adhere to the experimental setup detailed in [48] and train 2000 epochs with a learning rate of 5e-5 and a batch

size of 8 using a single RTX 3090Ti GPU.

We utilize an extensive array of evaluation metrics, including Mean Squared Error (MSE), Mean Absolute Error (MAE), Peak Signal Noise Ratio (PSNR), and the Structural Similarity Index Measure (SSIM) [57]. These metrics are computed across all predicted frames, where lower MAE and MSE scores, or higher SSIM and PSNR scores, signify superior prediction precision. To assess the models' computational demand, we measure the number of parameters, and floating-point operations (FLOPs) on TaxiBJ, and report the inference speed in frames per second (FPS) on a single NVIDIA A6000 GPU. This multifaceted evaluation provides insight into the efficiency and scalability of different models.

Following SwinLSTM [48], we adopt MSE and SSIM as our metrics for evaluating the Moving MNIST dataset, and SSIM and PSNR for the KTH dataset. Following OpenSTL [47], we provide a comprehensive analysis of the TaxiBJ dataset that includes not just MSE, MAE, and SSIM, but also detailed evaluations of model parameters and computational complexity, measured in FLOPs.

We chose three pivot datasets across various domains, including synthetic moving object trajectory, human motion, and traffic flow.

Moving MNIST. The moving MNIST dataset [44] serves as a benchmark synthetic dataset for evaluating sequence prediction models. Our approach to generating Moving MNIST sequences is in line with the methodology described in [44], where each sequence comprises 20 frames. We designate the initial 10 frames for input and the subsequent 10 frames as the prediction target. We adopt 10000 sequences for training and for fair comparisons, we use the pre-generated 10000 sequences [7] for validation.

KTH. The KTH dataset [39] contains 25 individuals performing six types of human actions (walking, jogging, running, boxing, hand waving, and hand clapping) in 4 different scenarios. Following previous works [7, 47, 53], we use persons 1-16 for training and persons 17-25 for validation and resizing each image to 128×128 . The models predict 10 frames from 10 observations at training time and 20 or 40 frames at inference time.

TaxiBJ. TaxiBJ [62] includes GPS data from taxis and meteorological data in Beijing. Each data frame is visualized as a $32 \times 32 \times 2$ heatmap, where the third dimension encapsulates the inflow and outflow of traffic within a designated area. Adhering to the experimental framework proposed in [62], we allocate the final four weeks of data for testing, while the preceding data is utilized for training. Our prediction model is tasked with using four sequential observations to forecast the subsequent four frames.

4.2. Main results

Tables 2, 3, and 4 provide quantitative comparisons between VMRNN and previous state-of-the-art models across three distinct datasets. These comparisons highlight VMRNN's exceptional capability as an efficient and highly generalizable approach for spatiotemporal prediction.

Moving MNIST We present the quantitative outcomes in Table 2, where our VMRNN model demonstrates notably superior performance compared to all other evaluated models. We report the results of previous research directly. On the Moving MNIST dataset, VMRNN not only achieves obviously lower MSE but also secures higher SSIM scores, significantly surpassing earlier methods by a substantial margin and archives 6.8% improvement over SwinLSTM.

TaxiBJ We present the quantitative outcomes in Table 3, where our VMRNN model demonstrates notably superior performance compared to all other evaluated models. For SwinLSTM, which is not reported in OpenSTL, we follow the same hyper-parameters with our VMRNN to ensure a fair comparison. For other methods, we use the results reported in OpenSTL. Obviously, on the TaxiBJ dataset, VMRNN not only registers substantially lower MSE and MAE values but also attains higher SSIM scores, thereby eclipsing previous methodologies to a considerable extent.

In Fig. 1, we provide a comparative analysis of parameters and FLOPs among recent spatiotemporal predictive learning methodologies applied to the TaxiBJ dataset. A position towards the lower left indicates a model that not only requires fewer parameters and computational resources but also delivers superior predictive performance, as evidenced by lower MSE values. Our VMRNN model showcases remarkable efficiency and effectiveness, achieving a clear lead by requiring fewer parameters and FLOPs while maintaining high prediction accuracy. As for Moving MNIST and KTH, the performance is similar, with fewer parameters and FLOPs than other methods.

KTH We present the quantitative results in Table 4. Our VMRNN model shows higher SSIM than all previous methods and comparable PSNR value with SwinLSTM. We report the results from the previous study directly and from our practice in OpenSTL, VMRNN achieves both better results in PSNR and SSIM than SwinLSTM by a large margin, either following the hyper-parameter setting in SwinLSTM or adopting the same setting as VMRNN.

4.3. Ablation Study

In this section, we perform ablation studies on TaxiBJ to analyze the impact of different elements on model performance. We discuss three major elements: the convolution layer, patch size, and the number of VSS Blocks.

The Convolution Layer. The role of the VSS Block convolution layer is to decode the spatiotemporal representations extracted by the VMRNN cell. We conduct experiments

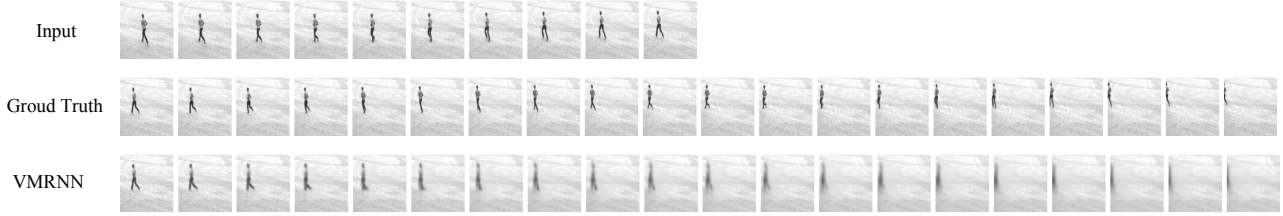


Figure 4. Qualitative results of VMRNN on KTH.

Table 1. Experimental setup. **VMRNNs** denotes the number of the VMRNN Cells in the spatio-temporal forecasting network. **VSB** denotes the number of the VSS Blocks in VMRNN cell. **Patch size** indicates the patch token size.

Dataset	Model	VMRNNs	VSB	Patch Size	Resolution	Train	Test	Epochs	Learning Rate	Batch Size	GPU
Moving MNIST	VMRNN-D	4	(2, 6, 6, 2)	2	(64, 64, 1)	10 → 10	10 → 10	2000	5e-5	8	1 * RTX 3090
KTH	VMRNN-B	1	6	2	(128, 128, 1)	10 → 10	10 → 20/40	100	5e-4/1e-4	2/1	4 * A6000
TaxiBJ	VMRNN-B	1	12	4	(32, 32, 2)	4 → 4	4 → 4	200	4e-4	16	1 * A6000

Table 2. Quantitative comparison of VMRNN and other methods on **Moving MNIST**. Each model observes 10 frames and predicts the subsequent 10 frames. Lower MSE and higher SSIM indicate better predictions.

Method	MSE ↓	SSIM ↑
ConvLSTM [41]	103.3	0.707
DFN [21]	89.0	0.726
FRNN [37]	69.7	0.813
VPN [24]	64.1	0.870
PredRNN [54]	56.8	0.867
CausalLSTM [52]	46.5	0.898
MIM [56]	44.2	0.910
E3D-LSTM [53]	41.3	0.910
LMC [27]	41.5	0.924
MAU [3]	27.6	0.937
PhyDNet [14]	24.4	0.947
CrevNet [61]	22.3	0.949
SimVP [7]	23.8	0.958
TAU [46]	19.8	0.957
MMVP [64]	22.2	0.948
SwinLSTM [48]	17.7	0.962
VMRNN	16.5	0.965

on depth-wise convolutions (DW Conv), convolution, and depth-wise convolutions with dilations (DW-D Conv) to analyze the impact of different decoding methods. Following [46], we combine DW Conv-DW-D Conv-1×1 Conv to model the large kernel convolutions, Table 6 shows that DW Conv performs much better than the other two methods.

Patch Size. The choice of image patch size critically influences the length of the input token sequences, where smaller patch sizes yield longer sequences. To evaluate the impact

Table 3. The performance on the TaxiBJ dataset. We provide a comparative analysis of parameters and FLOPs among recent spatiotemporal predictive learning methodologies. Our VMRNN model showcases remarkable efficiency and effectiveness.

Method	Para.(M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	SSIM ↑
ConvLSTM [41]	15.0	20.7	815	0.3358	15.32	0.9836
PredNet [32]	12.5	0.9	5031	0.3516	15.91	0.9828
PredRNN [54]	23.7	42.4	416	0.3194	15.31	0.9838
PredRNN++ [52]	38.4	63.0	301	0.3348	15.37	0.9834
E3DLSTM [53]	51.0	98.19	60	0.3421	14.98	0.9842
PhyDNet [14]	3.1	5.6	982	0.3622	15.53	0.9828
MIM [56]	37.9	64.1	275	0.3110	14.96	0.9847
MAU [3]	4.4	6.0	540	0.3268	15.26	0.9834
PredRNNv2 [55]	23.7	42.6	378	0.3834	15.55	0.9826
SimVP [7]	13.8	3.6	533	0.3282	15.45	0.9835
TAU [46]	9.6	2.5	1268	0.3108	14.93	0.9848
SimVPv2 [45]	10.0	2.6	1217	0.3246	15.03	0.9844
SwinLSTM [48]	2.9	1.3	1425	0.3026	15.00	0.9843
VMRNN	2.6	0.9	526	0.2887	14.69	0.9858

of different patch sizes on performance, we conducted experiments on the TaxiBJ and the KTH dataset using patch sizes 2, 4, and 8. As depicted in Table 5, a patch size of 4 for TaxiBJ and 2 for KTH distinctly outperforms the others.

The Number of VSS Blocks. To investigate the capabilities of different VSS blocks to model global spatial information, we examined the impact of varying the number of VSS blocks from 2 to 18. Fig. 5 illustrates the MSE and SSIM outcomes across different counts of VSS blocks on the TaxiBJ dataset. The results exhibit a trend of improvement followed by deterioration as the number of blocks increases, with an optimal performance observed at 12 VSS blocks, which achieve the best results in terms of both MSE and SSIM metrics.

Table 4. Quantitative evaluation on the **KTH** test set. We present the model observing 10 frames to predict 20 or 40 frames, and all metrics are averaged over the predicted frames. Higher SSIM and PSNR indicate better prediction quality.

Method	KTH (10 → 20)		KTH (10 → 40)	
	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑
ConvLSTM [41]	0.712	23.58	0.639	22.85
SAVP [6]	0.746	25.38	0.701	23.97
FRNN [37]	0.771	26.12	0.678	23.77
DFN [21]	0.794	27.26	0.652	23.01
PredRNN [54]	0.839	27.55	0.703	24.16
VarNet [23]	0.843	28.48	0.739	25.37
SVAP-VAE [26]	0.852	27.77	0.811	26.18
PredRNN++ [52]	0.865	28.47	0.741	25.21
E3d-LSTM [53]	0.879	29.31	0.810	27.24
STMFA Net [22]	0.893	29.85	0.851	27.56
SwinLSTM [48]	0.903	34.34	0.879	33.15
VMRNN	0.907	34.06	0.882	32.69

Table 5. Ablation study on patch size of TaxiBJ and KTH.

Patch size	TaxiBJ			KTH 10 → 20	
	MSE	MAE	SSIM	SSIM	PSNR
2	0.3566	15.46	0.9823	0.907	34.06
4	0.2887	14.69	0.9858	0.887	32.87
8	0.3400	15.32	0.9843	0.874	32.10

Table 6. Ablation study on different convolution methods of VSS Block on TaxiBJ.

Convolution Method	TaxiBJ		
	MSE	MAE	SSIM
DWConv	0.2887	14.69	0.9858
Conv2d	0.3185	16.58	0.9797
DW-DW-D-1x1	0.3050	15.02	0.9853

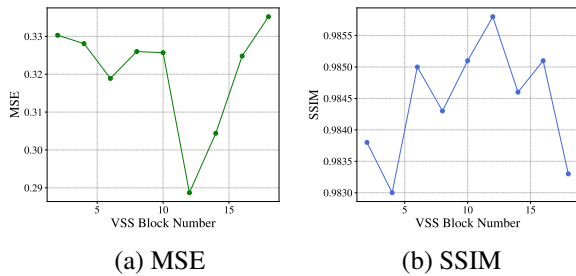


Figure 5. Ablation study on the different numbers of VSS Block with VMRNN on TaxiBJ.

4.4. Visualization

We present qualitative results of VMRNN on Moving MNIST in Fig. 6, TaxiBJ in Fig. 7, and KTH in Fig. 4. For all datasets, the first line is the input, the second line is the

ground truth, and the third line is the prediction of VMRNN. For TaxiBJ, we add the fourth line to show the difference between prediction and target. The visualization results reveal that VMRNN delivers impressive predictive performance, maintaining high accuracy across extended horizons.

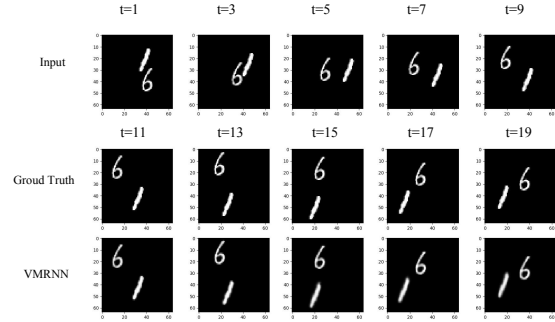


Figure 6. Qualitative results of VMRNN on Moving MNIST.

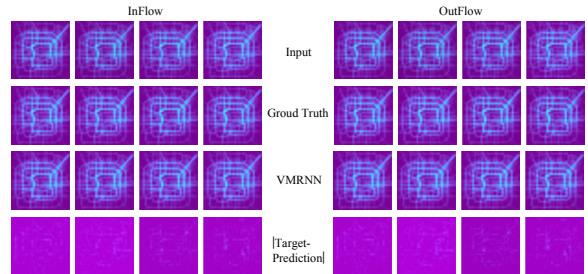


Figure 7. Qualitative results of VMRNN on TaxiBJ.

5. Conclusion

In this work, we introduce VMRNN, a novel approach that integrates LSTM architecture with VSS Blocks to tackle spatiotemporal forecasting challenges. Through rigorous evaluation across diverse datasets, VMRNN has proven its prowess by delivering superior performance while maintaining a smaller model size. This advancement is attributed to the model’s enhanced capability to learn and leverage global spatial dependencies with linear complexity, enabling a more refined understanding of spatiotemporal dynamics. The findings show that VMRNN sets a new strong baseline for future explorations in the field.

6. Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 62306257). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the National Natural Science Foundation

References

- [1] Apratim Bhattacharyya, Mario Fritz, and Bernt Schiele. Long-term on-board prediction of people in traffic scenes under uncertainty. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4194–4202, 2018. 1
- [2] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation. In *Computer Vision—ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 205–218. Springer, 2023. 4
- [3] Zheng Chang, Xinfeng Zhang, Shanshe Wang, Siwei Ma, Yan Ye, Xiang Xinguang, and Wen Gao. Mau: A motion-aware unit for video prediction and beyond. *Advances in Neural Information Processing Systems*, 34:26950–26962, 2021. 2, 7
- [4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3
- [5] Daniel Y Fu, Tri Dao, Khaled Kamal Saab, Armin W Thomas, Atri Rudra, and Christopher Re. Hungry hungry hippos: Towards language modeling with state space models. In *The Eleventh International Conference on Learning Representations*, 2022. 4
- [6] Hang Gao, Huazhe Xu, Qi-Zhi Cai, Ruth Wang, Fisher Yu, and Trevor Darrell. Disentangling propagation and generation for video prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9006–9015, 2019. 8
- [7] Zhangyang Gao, Cheng Tan, Lirong Wu, and Stan Z Li. Simvp: Simpler yet better video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3170–3180, 2022. 2, 6, 7
- [8] Haifan Gong, Luoyao Kang, Yitao Wang, Xiang Wan, and Haofeng Li. nnmamba: 3d biomedical image segmentation, classification and landmark detection with state space model. *arXiv preprint arXiv:2402.03526*, 2024. 2
- [9] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 2, 4
- [10] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. Hippo: Recurrent memory with optimal polynomial projections. *Advances in neural information processing systems*, 33:1474–1487, 2020. 3
- [11] 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. 3
- [12] Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2021. 2, 3
- [13] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021. 2, 3
- [14] Vincent Le Guen and Nicolas Thome. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11474–11484, 2020. 2, 3, 7
- [15] 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. *arXiv preprint arXiv:2402.15648*, 2024. 2
- [16] Tao Guo, Yinuo Wang, and Cai Meng. Mambamorph: a mamba-based backbone with contrastive feature learning for deformable mr-ct registration. *arXiv preprint arXiv:2401.13934*, 2024. 2
- [17] Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. *Advances in Neural Information Processing Systems*, 35:22982–22994, 2022. 3
- [18] Ramin Hasani, Mathias Lechner, Tsun-Hsuan Wang, Makram Chahine, Alexander Amini, and Daniela Rus. Liquid structural state-space models. In *The Eleventh International Conference on Learning Representations*, 2022. 4
- [19] Xiaotao Hu, Zhewei Huang, Ailin Huang, Jun Xu, and Shuchang Zhou. A dynamic multi-scale voxel flow network for video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6121–6131, 2023. 2
- [20] Simon Jenni, Givi Meishvili, and Paolo Favaro. Video representation learning by recognizing temporal transformations. In *European Conference on Computer Vision*, pages 425–442. Springer, 2020. 1
- [21] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. *Advances in neural information processing systems*, 29, 2016. 7, 8
- [22] Beibei Jin, Yu Hu, Qiankun Tang, Jingyu Niu, Zhiping Shi, Yinhe Han, and Xiaowei Li. Exploring spatial-temporal multi-frequency analysis for high-fidelity and temporal-consistency video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4554–4563, 2020. 8
- [23] Beibei Jin, Yu Hu, Yiming Zeng, Qiankun Tang, Shice Liu, and Jing Ye. Varnet: Exploring variations for unsupervised video prediction. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5801–5806. IEEE, 2018. 8
- [24] Nal Kalchbrenner, Aaron Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. Video pixel networks. In *International Conference on Machine Learning*, pages 1771–1779. PMLR, 2017. 7
- [25] Yong-Hoon Kwon and Min-Gyu Park. Predicting future frames using retrospective cycle gan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1811–1820, 2019. 1

- [26] Alex X Lee, Richard Zhang, Frederik Ebert, Pieter Abbeel, Chelsea Finn, and Sergey Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018. [8](#)
- [27] Sangmin Lee, Hak Gu Kim, Dae Hwi Choi, Hyung-Il Kim, and Yong Man Ro. Video prediction recalling long-term motion context via memory alignment learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3054–3063, 2021. [2](#), [7](#)
- [28] Rong Li, ShiJie Li, Xieyuanli Chen, Teli Ma, Wang Hao, Juergen Gall, and Junwei Liang. Tfnet: Exploiting temporal cues for fast and accurate lidar semantic segmentation. *arXiv preprint arXiv:2309.07849*, 2023. [1](#)
- [29] Jiarun Liu, Hao Yang, Hong-Yu Zhou, Yan Xi, Lequan Yu, Yizhou Yu, Yong Liang, Guangming Shi, Shaoting Zhang, Hairong Zheng, et al. Swin-umamba: Mamba-based unet with imagenet-based pretraining. *arXiv preprint arXiv:2402.03302*, 2024. [2](#)
- [30] 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. [2](#), [4](#)
- [31] 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](#), [3](#)
- [32] William Lotter, Gabriel Kreiman, and David Cox. Deep predictive coding networks for video prediction and unsupervised learning. In *International Conference on Learning Representations*, 2017. [7](#)
- [33] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29, 2016. [2](#)
- [34] Jun Ma, Feifei Li, and Bo Wang. U-mamba: Enhancing long-range dependency for biomedical image segmentation. *arXiv preprint arXiv:2401.04722*, 2024. [2](#)
- [35] Xuezhe Ma, Chunting Zhou, Xiang Kong, Junxian He, Liangke Gui, Graham Neubig, Jonathan May, and Luke Zettlemoyer. Mega: Moving average equipped gated attention. In *The Eleventh International Conference on Learning Representations*, 2022. [4](#)
- [36] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. In *International Conference on Learning Representations*, 2023. [4](#)
- [37] Marc Oliu, Javier Selva, and Sergio Escalera. Folded recurrent neural networks for future video prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 716–731, 2018. [7](#), [8](#)
- [38] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6964–6974, 2021. [1](#)
- [39] Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 32–36. IEEE, 2004. [5](#), [6](#)
- [40] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. [4](#)
- [41] Xingjian Shi, Zhouong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015. [1](#), [2](#), [3](#), [4](#), [7](#), [8](#)
- [42] Xingjian Shi, Zhihan Gao, Leonard Lausen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. Deep learning for precipitation nowcasting: A benchmark and a new model. *Advances in neural information processing systems*, 30, 2017. [1](#)
- [43] Jimmy TH Smith, Andrew Warrington, and Scott Linderman. Simplified state space layers for sequence modeling. In *The Eleventh International Conference on Learning Representations*, 2022. [3](#), [4](#)
- [44] 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. [2](#), [5](#), [6](#)
- [45] Cheng Tan, Zhangyang Gao, Siyuan Li, and Stan Z Li. Simvp: Towards simple yet powerful spatiotemporal predictive learning. *arXiv preprint arXiv:2211.12509*, 2022. [2](#), [7](#)
- [46] Cheng Tan, Zhangyang Gao, Lirong Wu, Yongjie Xu, Jun Xia, Siyuan Li, and Stan Z Li. Temporal attention unit: Towards efficient spatiotemporal predictive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18770–18782, 2023. [2](#), [7](#)
- [47] Cheng Tan, Siyuan Li, Zhangyang Gao, Wenfei Guan, Zedong Wang, Zicheng Liu, Lirong Wu, and Stan Z Li. Opentst: A comprehensive benchmark of spatio-temporal predictive learning. *Advances in Neural Information Processing Systems*, 36, 2024. [5](#), [6](#)
- [48] Song Tang, Chuang Li, Pu Zhang, and RongNian Tang. Swinlstm: Improving spatiotemporal prediction accuracy using swin transformer and lstm. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13470–13479, October 2023. [2](#), [3](#), [4](#), [5](#), [6](#), [7](#), [8](#)
- [49] Yujin Tang, Jiaming Zhou, Xiang Pan, Zeying Gong, and Junwei Liang. Postrainbench: A comprehensive benchmark and a new model for precipitation forecasting. *arXiv preprint arXiv:2310.02676*, 2023. [1](#)
- [50] 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. [3](#)
- [51] Pichao Wang, Wanqing Li, Philip Ogunbona, Jun Wan, and Sergio Escalera. Rgb-d-based human motion recognition with deep learning: A survey. *Computer Vision and Image Understanding*, 171:118–139, 2018. [1](#)
- [52] Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and S Yu Philip. Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *International Conference on Machine Learning*, pages 5123–5132. PMLR, 2018. [2](#), [3](#), [7](#), [8](#)

- [53] Yunbo Wang, Lu Jiang, Ming-Hsuan Yang, Li-Jia Li, Mingsheng Long, and Li Fei-Fei. Eidetic 3d lstm: A model for video prediction and beyond. In *International conference on learning representations*, 2018. [2](#), [3](#), [6](#), [7](#), [8](#)
- [54] Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and Philip S Yu. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. *Advances in neural information processing systems*, 30, 2017. [1](#), [2](#), [3](#), [7](#), [8](#)
- [55] Yunbo Wang, Haixu Wu, Jianjin Zhang, Zhifeng Gao, Jianmin Wang, Philip S Yu, and Mingsheng Long. Predrnn: A recurrent neural network for spatiotemporal predictive learning. *arXiv preprint arXiv:2103.09504*, 2021. [7](#)
- [56] Yunbo Wang, Jianjin Zhang, Hongyu Zhu, Mingsheng Long, Jianmin Wang, and Philip S Yu. Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9154–9162, 2019. [2](#), [3](#), [7](#)
- [57] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. [6](#)
- [58] Zhaohu Xing, Tian Ye, Yijun Yang, Guang Liu, and Lei Zhu. Segmamba: Long-range sequential modeling mamba for 3d medical image segmentation. *arXiv preprint arXiv:2401.13560*, 2024. [2](#)
- [59] Ziru Xu, Yunbo Wang, Mingsheng Long, Jianmin Wang, and M KLiss. Predcnn: Predictive learning with cascade convolutions. In *IJCAI*, pages 2940–2947, 2018. [1](#)
- [60] Yijun Yang, Zhaohu Xing, and Lei Zhu. Vivim: a video vision mamba for medical video object segmentation. *arXiv preprint arXiv:2401.14168*, 2024. [2](#)
- [61] Wei Yu, Yichao Lu, Steve Easterbrook, and Sanja Fidler. Efficient and information-preserving future frame prediction and beyond. 2020. [2](#), [3](#), [7](#)
- [62] Junbo Zhang, Yu Zheng, and Dekang Qi. Deep spatiotemporal residual networks for citywide crowd flows prediction. In *Thirty-first AAAI conference on artificial intelligence*, 2017. [1](#), [5](#), [6](#)
- [63] Liang Zhang, Guangming Zhu, Peiyi Shen, Juan Song, Syed Afaq Shah, and Mohammed Bennamoun. Learning spatiotemporal features using 3dcnn and convolutional lstm for gesture recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 3120–3128, 2017. [1](#)
- [64] Yiqi Zhong, Luming Liang, Ilya Zharkov, and Ulrich Neumann. Mmvp: Motion-matrix-based video prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4273–4283, October 2023. [7](#)
- [65] 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. [2](#), [4](#)