PMRMamba: Progressive Multi-Resolution Mamba for Long-Term Time Series Forecasting

Dejiao Niu*, Yuxuan Yang, Tao Cai, Wenyi Xiao, Chengyu Zhang, Liushan Zhang
School of Computer Science and Communication Engineering
Jiangsu University
Zhenjiang, China
Email: djniu@ujs.edu.cn, 2212308041@stmail.ujs.edu.cn

***Abstract*—Long-term time series forecasting remains a challenging task, as existing models struggle to capture long-range dependencies while maintaining computational efficiency. Recently, Mamba has demonstrated the ability to capture long-range dependencies with linear complexity, opening up new perspectives for time series modeling. Although Mamba excels at modeling long-range dependencies with linear complexity, the single-scale state propagation makes it difficult to simultaneously capture long-term trends and short-term variations within time series. In this paper, we propose a novel framework called Progressive Multi-Resolution Mamba (PMRMamba) which models time series through a coarse-to-fine progressive residual paradigm. PMRMamba consists of two tightly coupled modules that operate in conjunction with Selective SSM, progressively modeling time series at multiple resolutions to construct hierarchical temporal dependencies across different time scales. The Progressive Multi-Resolution Residual Decomposition progressively decomposes a time series into a set of complementary multi-resolution residual features through a stage-wise residual refinement mechanism, while the Temporal Resolution-wise Fusion adaptively integrates the multi-resolution residual features by explicitly estimating each resolution's contribution to the overall temporal representation. Extensive experiments on eight real-world benchmark datasets demonstrate that PMRMamba consistently outperforms state-of-the-art models, achieving significant improvements in long-term forecasting accuracy.**

***Index Terms*—Long-Term Time Series Forecasting, Mamba, Multi-Resolution Modeling.**

I. INTRODUCTION

Long-term time series forecasting (LTSF) aims to leverage historical observations to model complex temporal dynamics for accurate future prediction. Compared with short-term forecasting, it captures richer contextual information and long-range dependencies, playing a critical role in applications such as weather forecasting, traffic analysis, electricity demand, and energy scheduling [14].

While Transformer-based models achieve strong performance in LTSF, their quadratic complexity with respect to sequence length leads to high computational cost and low efficiency. To address this limitation, recent studies have explored alternative architectures for modeling long-range dependencies [1]–[7]. Among them, Mamba, as a SOTA State Space Model (SSM) variant, stands out for its ability to capture long-range dependencies with linear complexity [8]–[11].

* Corresponding author

Although Mamba effectively models long-range dependencies via Selective SSM, time series forecasting requires jointly capturing long-term trends and short-term variations, which are coupled within a shared state during recursive propagation. This unified update mechanism tends to emphasize slowly evolving structures while being less sensitive to rapidly changing local patterns, an issue that becomes more pronounced in long sequences as local variations are progressively smoothed.

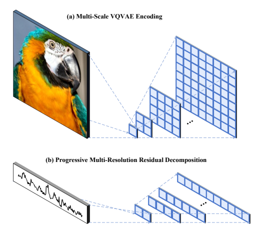

Fig. 1: (a) VAR uses Multi-Scale VQVAE Encoding to decompose image residuals into multi-scale token representations for coarse-to-fine autoregressive generation. (b) Similarly, we propose Progressive Multi-Resolution Residual Decomposition for time series, which splits temporal residuals into subsequences at different resolutions to capture hierarchical dependencies across scales.

Visual Autoregressive Modeling (VAR) [12] introduces a multi-scale residual autoregressive framework for images. Inspired by this, we adapt multi-resolution residual modeling to long-term time series forecasting (Fig. 1) to capture dynamics across scales. However, time series differ from images: cross-resolution dependencies are cumulative and complementary rather than redundant, so naive decomposition can disrupt temporal continuity and attenuate information. Additionally, SSM outputs at different resolutions have inconsistent state evolution and statistics, making simple additive fusion prone to interference and unstable representations.

To address these challenges, we propose Progressive Multi-Resolution Mamba (PMRMamba), a novel Mamba framework for multi-resolution time-series modeling. PMRMamba employs a coarse-to-fine progressive residual architecture to capture hierarchical temporal dependencies. Specifically, Progressive Multi-Resolution Residual Decomposition performs stepwise residual refinement, where the reconstruction residual at each resolution is propagated to the next, explicitly preserving fine-grained variations to jointly capture global structures and local dynamics. We further deploy parallel Selective SSMs, each modeling temporal dependencies at its respective resolution, allowing different scales to focus on characteristic dynamics. To integrate multi-resolution features, Temporal Resolution-wise Fusion adaptively generates fusion weights for each resolution and aligns SSM outputs, mitigating noise accumulation and temporal misalignment while maintaining sequence consistency. Our main contributions are as follows:

- We propose the Progressive Multi-Resolution Mamba (PMRMamba), a multi-resolution framework for long-term time series forecasting, which constructing hierarchical temporal dependencies across resolutions through a coarse-to-fine progressive residual architecture.
- To capture temporal dynamics at multiple granularities, we propose Progressive Multi-Resolution Residual Decomposition, which constructs structured multi-resolution representations via stage-wise residual refinement. To integrate these features, we introduce Temporal Resolution-wise Fusion, a resolution-aware adaptive weighting mechanism that highlights each resolution's contribution while mitigating noise and temporal misalignment.
- PMRMamba is extensively evaluated on eight real-world datasets against a wide range of representative state-of-the-art models, achieving consistent and significant performance improvements across all benchmarks.

II. Related Work

In recent years, researchers have actively explored model architectures for long-term time series forecasting, among which Mamba has emerged as a promising solution due to its ability to capture long-range dependencies with linear computational complexity. Building upon Mamba, several studies have improved its suitability for time series modeling [15]. For instance, S-Mamba [9] enhances multivariate representation via linear tokenization and bidirectional modeling, while Bi-Mamba+ [10] captures local dynamics through patch-based segmentation and introduces a forget gate to balance new and historical information. Timemachine [11] further leverages multiple Mamba modules to jointly model channel-mixed and channel-independent dependencies, enabling comprehensive temporal representation. Overall, these methods improve long-term forecasting through architectural design; however, as all temporal features are still propagated within a unified recursive state, the coupling across different temporal resolutions remains fundamentally unresolved.

III. Overview

Long-term time series forecasting refers to inferring future values $\mathbf{X}_T = \{\mathbf{x}_{L+1}, \mathbf{x}_{L+2}, \ldots, \mathbf{x}_{L+T}\} \in \mathbb{R}^{T\times M}$ from a given historical sequence $\mathbf{X}_L = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_L\} \in \mathbb{R}^{L\times M}$, where L represents the length of the historical sequence, T denotes the length of the forecasting horizon, and M is the number of variables in the time series. The overall architecture of the PMRMamba is illustrated in Figure 2. It consists of the Input Module, the stacking of PMRMamba blocks, and the Forecasting Module.

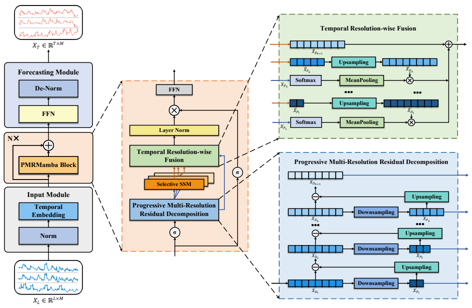

Fig. 2: The overall architecture of PMRMamba consists of an Input Module, a stack of PMRMamba blocks, and a Forecasting Module. The PMRMamba block serves as the core component, modeling temporal dynamics in a coarse-to-fine manner across multiple resolutions. Blue arrows denote the outputs of the Progressive Multi-Resolution Residual Decomposition, while red arrows represent the outputs of the Selective SSMs, which are subsequently fed into the Temporal Resolution-wise Fusion module.

A. Input Module

To mitigate the training instability caused by non-stationarity in time series, inspired by [13], we first apply a normalization operation along the temporal dimension to reduce distributional shifts. Subsequently, value embedding is applied to the normalized series, projecting it into a high-dimensional representation space and providing enriched representations for the subsequent module. This module can be formalized as:

$$\mu_X = \frac{1}{L}\sum_{i=1}^{L} x_i, \quad \sigma_X^2 = \frac{1}{L}\sum_{i=1}^{L}(x_i - \mu_X)^2, \qquad X_{\text{norm}} = \frac{X - \mu_X}{\sigma_X} \tag{1}$$

$$X_D = \text{TemporalEmbedding}(X_{\text{norm}}) \tag{2}$$

where $\mu_X, \sigma_X \in \mathbb{R}^{M\times 1}$, $X_{\text{norm}} \in \mathbb{R}^{L\times M}$ denotes the normalized time series, and $\text{TemporalEmbedding}(\cdot)$ maps the input series to an embedded representation $X_D \in \mathbb{R}^{M\times D}$ via a linear transformation along the temporal dimension, thereby enabling the subsequent multi-resolution decomposition to capture richer temporal dependencies.

B. PMRMamba Block

The unified state update in Mamba emphasizes slow long-term structures while reducing sensitivity to fast local patterns, particularly in long sequences. To address this, we propose Progressive Multi-Resolution Mamba (PMRMamba, Fig. 2), inspired by Visual Autoregressive Modeling (VAR), which captures temporal dynamics across multiple resolutions. PMRMamba adopts a channel-independent strategy, removing convolution and linear projections to enable direct temporal modeling and avoid premature feature coupling. It employs Progressive Multi-Resolution Residual Decomposition for coarse-to-fine separation of dynamics, with each resolution modeled in parallel via channel-independent Selective SSMs.

Multi-resolution features are adaptively fused through the Temporal Resolution-wise Fusion module with time-step–wise weighting and LayerNorm, and a lightweight FFN enhances nonlinear expressiveness before propagating representations to subsequent blocks.

1) Progressive Multi-Resolution Residual Decomposition: As illustrated in the blue part on the right side of Figure 2, we consider a univariate sequence $X_D^i \in \mathbb{R}^{1\times D}$, $i \in \{1, 2, \ldots, M\}$. We define a set of resolutions $\{P_1, P_2, \ldots, P_k\}$ to progressively downsample the sequence at multiple temporal resolutions. The input feature sequence X_D^i is denoted as $X_{D_1} \in \mathbb{R}^{1\times D}$. At the i-th decomposition step, the resolution-specific subsequence is obtained by:

$$X_{P_i} = \mathrm{Downsample}(X_{D_i}, P_i), \quad i \in \{1, 2, \ldots, k\} \tag{3}$$

where $X_{P_i} \in \mathbb{R}^{1\times P_i}$. Then, the subsequence is upsampled back to the temporal feature sequence length.

$$\bar{X}_{P_i} = \mathrm{Upsample}(X_{P_i}, P_i), \quad i \in \{1, 2, \ldots, k\} \tag{4}$$

where $\bar{X}_{P_i} \in \mathbb{R}^{1\times D}$. The residual sequence is subsequently computed in a residual manner, serving as the input for the next decomposition step to progressively capture finer-grained temporal variations.

$$X_{D_{i+1}} = X_{D_i} - \bar{X}_{P_i}, \quad i \in \{1, 2, \ldots, k\} \tag{5}$$

Progressive residual decomposition extracts temporal features across multiple resolutions, with each downsampling stage capturing dominant patterns at its scale. To preserve fine-grained details lost during decomposition, the final residual sequence $X_{D_{k+1}} \in \mathbb{R}^{1\times D}$ is retained, ensuring completeness and consistency of the temporal representation.

In the multi-resolution decomposition, downsampled subsequences $X_{P_1}, \ldots, X_{P_k}$ are processed in parallel by their respective Selective SSMs to capture long-term dependencies at different scales. The retained residual $X_{D_{k+1}}$ is handled by a separate SSM to model fine-grained dynamics not fully captured in the decomposition, formalized as follows:

$$\hat{X}_{P_i} = \mathrm{SSM}_i(X_{P_i}), \quad i \in \{1, 2, \ldots, k\} \tag{6}$$

$$\hat{X}_{D_{k+1}} = \mathrm{SSM}(X_{D_{k+1}}) \tag{7}$$

Consequently, we obtain the output sequence set $\{\hat{X}_{P_1}, \hat{X}_{P_2}, \ldots, \hat{X}_{P_k}, \hat{X}_{D_{k+1}}\}$.

2) Temporal Resolution-wise Fusion: To integrate multi-resolution representations from the Selective SSMs, we propose Temporal Resolution-wise Fusion. Lower resolutions capture global trends, while higher resolutions preserve fine-grained variations. Direct summation would obscure these differences and introduce noise. Instead, each resolution's contribution is adaptively weighted using a Softmax operation followed by mean pooling:

$$W_{P_i} = \mathrm{MeanPooling}\big(\mathrm{Softmax}(X_{P_i})\big), \quad i \in \{1, 2, \ldots, k\} \tag{8}$$

where the weights $\{W_{P_1}, W_{P_2}, \ldots, W_{P_k}\}$ represent the relative contribution of each resolution to the overall temporal representation. To ensure temporal alignment across resolutions, we upsample the output of each Selective SSM to a unified length D.

$$\hat{X}_{D_i} = \mathrm{Upsample}(\hat{X}_{P_i}, D), \quad i \in \{1, 2, \ldots, k\} \tag{9}$$

The aligned sequences are then multiplied element-wise with their corresponding weights:

$$\bar{X}_{D_i} = \hat{X}_{D_i} \odot W_{P_i}, \quad i \in \{1, 2, \ldots, k\} \tag{10}$$

Finally, we fuse the weighted sequences together with the residual sequence from the last decomposition stage to obtain the final temporal representation:

$$\bar{X}_{D_{\text{total}}} = \hat{X}_{D_{k+1}} + \sum_{i=1}^{k} \bar{X}_{D_i} \tag{11}$$

Algorithm 1 Progressive Multi-Resolution Residual Decomposition

Inputs: an univariate sequence X_D;
Hyperparameters: steps k, resolutions $\{P_1, P_2, \ldots, P_k\}$;
$R_D = [\,]$;
for $i = 1, \ldots, k$ **do**
    $X_{P_i} = \mathrm{Downsample}(X_D, P_i)$;
    $R_D = \text{queue\_push}(R_D, X_{P_i})$;
    $\bar{X}_{P_i} = \mathrm{Upsample}(X_{P_i}, P_i)$;
    $X_D = X_D - \bar{X}_{P_i}$;
end for
$X_{D_k+1} = X_D$;
$R_D = \text{queue\_push}(R_D, X_{D_k+1})$;
return multi-resolution subsequences R_D;

Algorithm 2 Temporal Resolution-wise Fusion

Inputs: the outputs of the Selective SSMs
    $R_F = \left\{\hat{X}_{P_1}, \hat{X}_{P_2}, \ldots, \hat{X}_{P_k}, \hat{X}_{D_{k+1}}\right\}$;
    multi-resolution subsequences R_D
    $R_D = \{X_{P_1}, X_{P_2}, \ldots, X_{P_k}\}$;
Hyperparameters: steps k, feature dimension D;
$\bar{X}_{D_{total}} = 0$;
for $i = 1, \ldots, k$ **do**
    $\hat{X}_{P_i} = \text{queue\_pop}(R_F)$;
    $X_{P_i} = \text{queue\_pop}(R_D)$;
    $\hat{X}_{D_i} = \mathrm{Upsample}(\hat{X}_{P_i}, D)$;
    $\mathcal{W}_{P_i} = \mathrm{MeanPooling}(\mathrm{Softmax}(X_{P_i}))$
    $\bar{X}_{D_{total}} = \bar{X}_{D_{total}} + \hat{X}_{D_i} \odot \mathcal{W}_{P_i}$;
end for
$\hat{X}_{D_{k+1}} = \text{queue\_pop}(R_D)$;
$\bar{X}_{D_{total}} = \bar{X}_{D_{total}} + \hat{X}_{D_{k+1}}$;
return the output $\bar{X}_{D_{total}}$;

The residual sequence at stage $k+1$ is added directly to the fused representation without a weight, allowing it to compensate for fine-grained details missed by earlier resolutions and preserve temporal completeness and consistency. Experiments confirm the effectiveness of this design. The decomposition and fusion procedures are detailed in Algorithms 1 and 2.

C. Forecasting Module

The output of the final PMRMamba block is fed into the Feed-Forward Network (FFN) that projects the feature dimension to the prediction horizon. The FFN consists of two linear transformations with a GELU activation, introducing nonlinear transformation capabilities to enhance temporal feature representation. Finally, the output sequence $X_T \in \mathbb{R}^{T\times M}$ is obtained via the denormalization operation.

IV. EXPERIMENTS

A. Setup

1) Datasets: To comprehensively evaluate the performance of the proposed PMRMamba in long-term time series forecasting, we conduct extensive experiments on eight widely used real-world benchmark datasets, including four ETT datasets (ETTh1, ETTh2, ETTm1, ETTm2), Traffic, Electricity, Exchange, and Solar. These datasets span diverse domains such as energy, transportation, and climate. The detailed statistics of the datasets are summarized in Table I.

2) Baseline Models: We compare our model with three SSM-based models: Bi-Mamba+ [10], S-Mamba [9], TimeMachine [11];four Transformer-based models: iTransformer [3], PatchTST [5], Crossformer [6], Autoformer [1]; one Linear-based model: DLinear [7]; one TCN-based model: TimesNet [4]. These models achieve strong forecasting performance under their respective architectural designs.

3) Experimental Settings: All experiments are conducted on a single NVIDIA GeForce RTX 3090 GPU with 24GB memory. PMRMamba and baselines are trained using Adam with a learning rate of 1e-4, batch size 32, and up to 10 epochs with early stopping, minimizing Mean Squared Error (MSE). For ETTh1, ETTh2, ETTm1, ETTm2, Electricity, Solar, and Weather, the Mamba hidden state dimension is 8 (16 for Traffic), with an embedding dimension of 256. Temporal resolutions 4,8,16,32,64,128 are defined, and the top four 4,8,16,32 (k=4) are used by default. We further analyze the impact of hyperparameters, evaluating performance with MSE and Mean Absolute Error (MAE), where lower values indicate better accuracy.

B. Main Results

We evaluate PMRMamba on eight real-world datasets with a fixed input length $L = 96$ and forecasting horizons $T \in 96, 192, 336, 720$. As shown in Table II, PMRMamba achieves consistently strong performance, ranking best in 52 out of 64 settings, demonstrating robust generalization. Compared with Mamba-based baselines, PMRMamba consistently reduces forecasting errors, outperforming Bi-Mamba+ [10] with reductions of 4.53% in MSE and 3.83% in MAE, and S-Mamba [9] with reductions of 9.83% and 7.14%, respectively, highlighting the effectiveness of multi-resolution decomposition and adaptive fusion. Compared with Transformer-based models, iTransformer [3] and PatchTST [5], PMRMamba adopts a coarse-to-fine multi-resolution strategy while leveraging Mamba's long-range modeling capability, achieving reductions of 9.48% in MSE and 6.56% in MAE over iTransformer, and 11.92% and 8.86% over PatchTST. Notably, on high-dimensional datasets such as Electricity, Solar-Energy, and Traffic, PMRMamba consistently attains the lowest errors, demonstrating strong scalability. Overall, PMRMamba effectively captures temporal dependencies under a fixed input window and consistently outperforms SOTA models.

C. Ablation Study

To assess the contribution of each key component in PMRMamba, we conduct ablation studies under consistent settings by varying the input length $L \in 96, 192, 336, 720$ with a fixed prediction horizon $T = 96$. Specifically, we examine four variants: (1) *w/o Fusion*, which replaces Temporal Resolution-wise Fusion with direct summation; (2) *w/o FFN*, which removes the lightweight FFN to evaluate its role in nonlinear modeling; (3) *w/o Residual*, which discards the final residual sequence to assess its contribution as fine-grained detail; and (4) *w/ Linear+Conv*, which retains the original Mamba input projection and convolution layers. The results are summarized in Table III.

Replacing Temporal Resolution-wise Fusion with direct summation leads to notable performance degradation, with MSE and MAE increasing on both ETTm1 and Electricity, indicating that naive fusion fails to differentiate contributions across resolutions and introduces noise. Similarly, removing the residual sequence results in clear performance drops, highlighting its role in supplementing fine-grained information not captured during decomposition. The removal of other components also degrades performance, further confirming the importance of each module in enabling effective multi-resolution time series modeling.

D. Hyperparameter Sensitivity Analysis

We analyze key internal hyperparameters of PMRMamba to assess their effects on performance, focusing on the number of temporal resolutions, embedding dimension of time series, and their joint influence on model behavior.

1) Top-k resolutions k: We analyze the impact of the number of temporal resolutions k on model performance, as illustrated in Fig. 3. Under fixed settings with input length $L \in 96, 192, 336, 720$, prediction horizon $T = 96$, and embedding dimension 256, the performance does not improve monotonically with increasing k. For $L = 96$, the optimal performance is achieved at $k = 4$, indicating that insufficient resolutions fail to capture key dynamics, while excessive resolutions introduce redundancy and noise. As the input length increases, incorporating more resolutions consistently improves performance, suggesting that a richer multi-resolution decomposition is beneficial for modeling temporal dependencies across multiple time scales.

2) Emebedding dimensions D: To examine the effect of embedding dimension, we fix all other hyperparameters and set the number of temporal resolutions to $k = 4$, while varying $D \in 32, 64, 128, 256, 512$. Experiments are conducted on the Solar dataset with different input lengths and a fixed prediction horizon of 96, as shown in Fig. 4.The results show

TABLE I: The statistics of all datasets.

Dataset	ETTh1& ETTh2	ETTm1& ETTm2	Electricity	Solar-Energy	Traffic	Weather
Timesteps	17,420	69,680	26,304	52,560	17,544	52,696
Variables	7	7	321	137	862	21
Frequency	1 hour	15 mins	1 hour	10 mins	1 hour	10 mins
Cyclic Patterns	Daily	Daily	Daily& Weekly	Daily	Daily& Weekly	Daily

TABLE II: Experimental results on the eight real-world datasets. The historical input window is fixed to 96 time steps, and the forecasting horizons are set to 96, 192, 336, and 720 steps into the future. The best results are in bold and the second-best results are underlined.

Models		Ours(PMRMamba)		Bi-Mamba+		S-Mamba		TimeMachine		iTransformer		PatchTST		Crossformer		Autoformer		DLinear		TimesNet	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	**0.374**	0.398	0.382	0.398	0.390	0.409	0.379	**0.392**	0.386	0.405	0.393	0.408	0.423	0.448	0.449	0.459	0.387	0.406	0.384	0.402
	192	0.427	0.430	**0.426**	0.426	0.447	0.442	0.433	**0.422**	0.441	0.436	0.445	0.434	0.471	0.474	0.500	0.482	0.439	0.435	0.436	0.429
	336	**0.468**	**0.449**	0.485	0.456	0.508	0.478	0.487	0.458	0.487	0.458	0.485	0.455	0.570	0.546	0.521	0.496	0.493	0.457	0.491	0.469
	720	**0.474**	**0.441**	0.509	0.482	0.532	0.511	**0.474**	**0.441**	0.503	0.491	0.500	0.488	0.653	0.621	0.514	0.512	0.519	0.516	0.521	0.500
ETTh2	96	**0.292**	**0.342**	0.293	0.343	0.300	0.350	0.297	0.349	0.297	0.349	0.302	0.348	0.745	0.584	0.346	0.388	0.305	0.352	0.340	0.374
	192	**0.378**	**0.398**	0.382	0.400	0.388	0.402	0.380	0.406	0.380	0.400	0.389	0.400	0.877	0.656	0.456	0.452	0.424	0.439	0.402	0.414
	336	**0.424**	**0.430**	0.429	0.430	0.432	0.431	0.425	0.440	0.428	0.432	0.426	0.433	1.043	0.731	0.482	0.486	0.456	0.473	0.452	0.452
	720	**0.422**	**0.425**	0.431	0.432	0.433	0.431	0.429	0.448	0.428	0.432	0.431	0.446	1.104	0.763	0.515	0.511	0.476	0.493	0.462	0.468
ETTm1	96	**0.314**	**0.351**	0.325	0.362	0.336	0.370	0.317	0.355	0.334	0.368	0.329	0.367	0.404	0.426	0.505	0.475	0.353	0.374	0.338	0.375
	192	**0.357**	**0.378**	0.377	0.387	0.387	0.397	0.362	0.379	0.377	0.391	0.367	0.385	0.450	0.451	0.553	0.496	0.389	0.391	0.374	0.387
	336	**0.387**	0.403	0.402	0.411	0.433	0.427	0.392	**0.401**	0.426	0.420	0.399	0.419	0.532	0.515	0.621	0.537	0.421	0.413	0.410	0.411
	720	**0.453**	0.445	0.469	0.453	0.495	0.464	0.460	0.439	0.491	0.459	0.461	**0.438**	0.666	0.589	0.671	0.561	0.484	0.448	0.478	0.450
ETTm2	96	**0.174**	**0.257**	0.176	0.263	0.179	0.263	0.176	**0.257**	0.180	0.264	0.177	0.260	0.287	0.366	0.255	0.339	0.182	0.264	0.187	0.267
	192	**0.241**	**0.301**	0.242	0.304	0.253	0.310	0.248	0.308	0.250	0.309	0.248	0.306	0.414	0.492	0.281	0.340	0.257	0.315	0.249	0.309
	336	**0.304**	**0.342**	**0.304**	0.344	0.312	0.348	0.307	0.346	0.311	0.348	0.305	0.343	0.597	0.542	0.339	0.372	0.318	0.353	0.321	0.351
	720	0.405	0.399	**0.402**	0.402	0.412	0.408	0.407	0.404	0.412	0.407	0.403	**0.397**	1.730	1.042	0.433	0.432	0.426	0.419	0.408	0.403
Weather	96	0.170	**0.213**	0.177	0.220	0.184	0.216	0.172	0.218	0.174	0.214	0.178	0.219	**0.158**	0.230	0.266	0.336	0.196	0.235	0.172	0.220
	192	**0.212**	**0.250**	0.222	0.261	0.232	0.265	0.214	0.254	0.221	0.254	0.224	0.259	0.213	0.284	0.307	0.367	0.241	0.271	0.219	0.261
	336	**0.271**	**0.293**	0.274	0.297	0.286	0.304	**0.271**	0.297	0.278	0.298	0.292	0.306	0.274	0.335	0.359	0.395	0.292	0.306	0.280	0.306
	720	0.350	**0.345**	0.353	0.348	0.362	0.353	**0.349**	0.348	0.358	0.349	0.354	0.348	0.377	0.402	0.419	0.428	0.363	0.353	0.365	0.359
Electricity	96	**0.161**	**0.253**	0.187	0.278	0.189	0.278	0.167	0.260	0.195	0.280	0.221	0.299	0.219	0.314	0.201	0.317	0.206	0.288	0.215	0.312
	192	**0.175**	**0.265**	0.186	0.275	0.200	0.289	0.179	0.270	0.193	0.275	0.209	0.287	0.231	0.322	0.222	0.334	0.206	0.290	0.215	0.311
	336	**0.181**	**0.274**	0.213	0.304	0.221	0.310	0.187	0.279	0.221	0.304	0.239	0.317	0.246	0.337	0.231	0.338	0.220	0.305	0.241	0.335
	720	**0.216**	**0.304**	0.256	0.336	0.266	0.346	0.228	0.312	0.284	0.360	0.296	0.359	0.280	0.363	0.254	0.361	0.252	0.337	0.279	0.363
Traffic	96	**0.438**	**0.301**	0.457	0.359	0.527	0.362	0.463	0.308	0.541	0.369	0.603	0.396	0.522	0.414	0.613	0.388	0.647	0.384	0.593	0.321
	192	**0.456**	0.309	0.484	0.316	0.546	0.374	0.463	**0.308**	0.566	0.383	0.620	0.406	0.530	0.400	0.616	0.382	0.596	0.359	0.617	0.336
	336	**0.457**	**0.301**	0.493	0.360	0.579	0.391	0.481	0.317	0.589	0.398	0.638	0.419	0.558	0.421	0.616	0.382	0.601	0.361	0.629	0.452
	720	**0.491**	**0.322**	0.528	0.362	0.647	0.425	0.516	0.337	0.656	0.427	0.703	0.447	0.598	0.450	0.622	0.459	0.642	0.381	0.640	0.486
Solar	96	**0.209**	**0.251**	0.214	0.254	0.224	0.261	0.230	0.267	0.246	0.275	0.277	0.324	0.310	0.331	0.884	0.711	0.290	0.378	0.250	0.292
	192	**0.238**	**0.275**	0.254	0.283	0.272	0.291	0.274	0.294	0.292	0.300	0.326	0.349	0.734	0.725	0.834	0.692	0.320	0.398	0.296	0.318
	336	**0.251**	**0.277**	0.270	0.292	0.320	0.327	0.292	0.305	0.324	0.320	0.366	0.362	0.750	0.735	0.941	0.723	0.353	0.415	0.319	0.330
	720	**0.252**	**0.282**	0.282	0.304	0.331	0.334	0.295	0.306	0.322	0.322	0.362	0.364	0.769	0.765	0.882	0.717	0.356	0.413	0.338	0.337

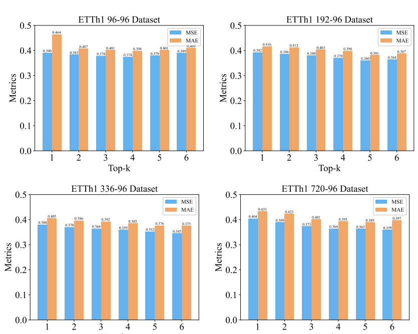

Fig. 3: Sensitivity analysis of the top-k hyperparameter on the ETTh1 dataset.

that increasing D does not consistently improve performance; the best results are achieved at $D = 128$. This suggests that larger embedding dimensions do not necessarily yield better accuracy, which can be attributed to the mismatch between embedding capacity and resolution granularity. Although a larger D enhances representation capability, a limited number of resolutions constrains effective feature disentanglement across temporal scales, thereby limiting the benefits of multi-resolution decomposition. These findings indicate that embedding dimension should be jointly considered with the number of temporal resolutions, which we further explore in the next section.

3) Impact of embedding dimensions D and different resolutions k: We systematically study the joint effect of resolution number k and embedding dimension D on the Weather dataset (Fig. 5). Results show that optimal k depends on D: larger embeddings enable more effective decomposition and benefit from increasing k, while smaller embeddings limit gains. For longer input sequences, small embeddings yield limited improvement, whereas moderate k benefits larger embeddings. Excessive resolution decomposition can degrade performance due to noise, highlighting the trade-off between feature capacity and resolution granularity in PMRMamba.

TABLE III: Ablation study of key components in PMRMamba.

Models		w/o Fusion		w/o FFN		w/o Residual		w/ Linear+Conv		PMRMamba	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	0.325	0.364	0.319	0.366	0.340	0.377	0.333	0.369	**0.314**	**0.351**
	192	0.297	0.347	0.291	0.342	0.310	0.358	0.300	0.353	**0.280**	**0.333**
	336	0.296	0.349	0.288	0.344	0.295	0.350	0.319	0.366	**0.275**	**0.322**
	720	0.306	0.361	0.281	0.342	0.300	0.348	0.307	0.364	**0.277**	**0.324**
Electricity	96	0.189	0.277	0.171	0.263	0.196	0.283	0.172	0.268	**0.161**	**0.253**
	192	0.174	0.264	0.164	0.251	0.182	0.273	0.168	0.259	**0.142**	**0.245**
	336	0.168	0.250	0.155	0.249	0.174	0.267	0.154	0.247	**0.138**	**0.238**
	720	0.166	0.249	0.148	0.234	0.169	0.253	0.146	0.250	**0.139**	**0.240**

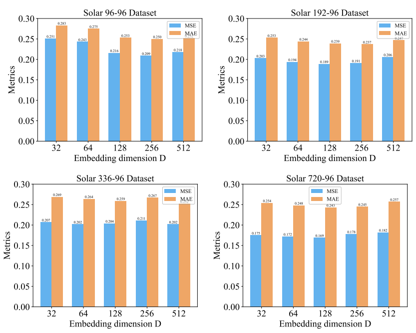

Fig. 4: Sensitivity analysis of the embedding dimension D on the Solar dataset.

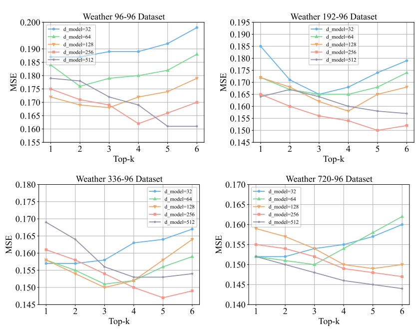

Fig. 5: Sensitivity analysis of embedding dimensions and different resolutions on PMRMamba performance across different historical input lengths on the Weather dataset.

V. CONCLUSION

We propose PMRMamba, a multi-resolution Mamba framework for long-term time series forecasting that captures long-range dependencies across multiple temporal resolutions. The Progressive Multi-Resolution Residual Decomposition module extracts hierarchical dependencies via coarse-to-fine residual modeling, while the Temporal Resolution-wise Fusion module adaptively integrates features according to each resolution's contribution. Extensive experiments show that PMRMamba achieves competitive performance across benchmarks, outperforming state-of-the-art methods in most cases. Ablation studies and sensitivity analyses further confirm the effectiveness and robustness of its components. Moreover, PMRMamba maintains stable performance with increasing input lengths, demonstrating strong long-term dependency modeling. Future work will explore finer-grained and frequency-domain resolutions to more comprehensively characterize multi-resolution dynamics.

REFERENCES

[1] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in neural information processing systems*, vol. 34, pp. 22 419–22 430, 2021.

[2] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *International conference on machine learning*. PMLR, 2022, pp. 27 268–27 286.

[3] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," *arXiv preprint arXiv:2310.06625*, 2023.

[4] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," *arXiv preprint arXiv:2210.02186*, 2022.

[5] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers. arxiv 2022," *arXiv preprint arXiv:2211.14730*, 2022.

[6] Y. Zhang and J. Yan, "Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting," in *The eleventh international conference on learning representations*, 2023.

[7] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 9, 2023, pp. 11 121–11 128.

[8] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," in *First conference on language modeling*, 2024.

[9] Z. Wang, F. Kong, S. Feng, M. Wang, X. Yang, H. Zhao, D. Wang, and Y. Zhang, "Is mamba effective for time series forecasting?" *Neurocomputing*, vol. 619, p. 129178, 2025.

[10] A. Liang, X. Jiang, Y. Sun, X. Shi, and K. Li, "Bi-mamba+: Bidirectional mamba for time series forecasting," *arXiv preprint arXiv:2404.15772*, 2024.

[11] M. A. Ahamed and Q. Cheng, "Timemachine: A time series is worth 4 mambas for long-term forecasting," in *ECAI 2024: 27th European Conference on Artificial Intelligence, 19-24 October 2024, Santiago de Compostela, Spain-Including 13th Conference on Prestigious Applications of Intelligent Systems. European Conference on Artificial Intelli*, vol. 392, 2024, p. 1688.

[12] K. Tian, Y. Jiang, Z. Yuan, B. Peng, and L. Wang, "Visual autoregressive modeling: Scalable image generation via next-scale prediction," *Advances in neural information processing systems*, vol. 37, pp. 84 839–84 865, 2024.

[13] Y. Liu, H. Wu, J. Wang, and M. Long, "Non-stationary transformers: Exploring the stationarity in time series forecasting," *Advances in neural information processing systems*, vol. 35, pp. 9881–9893, 2022.

[14] Y. Wang, H. Wu, J. Dong, Y. Liu, M. Long, and J. Wang, "Deep time series models: A comprehensive survey and benchmark.(2024)," *URL https://arxiv. org/abs/2407.13278*, vol. 18, 2024.

[15] H. Qu, L. Ning, R. An, W. Fan, T. Derr, H. Liu, X. Xu, and Q. Li, "A survey of mamba," *arXiv preprint arXiv:2408.01129*, 2024.