MUSE: DEPENDENCY-AWARE ADAPTATION OF A FROZEN VISION BACKBONE FOR MULTIVARIATE TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multivariate time-series forecasting is essential to many real-world applications. Recent large vision models (LVMs) offer a promising paradigm by transferring cross-domain visual priors to time-series forecasting. However, existing LVM-based methods face two key challenges: balancing independent visual representation spaces with cross-variable dependency modeling, and adapting vision backbones pretrained on natural images to the distinct temporal semantics of time-series images. To address these challenges, we propose MUSE, a dependency-aware adaptation framework built on a fully frozen pretrained MAE. First, the Variable Context Refinement Module (VCR) aggregates shared temporal information within each variable and models cross-variable contextual dependencies while preserving independent visual spaces. Second, the Temporal–Periodic Refinement Module (TPR) performs lightweight refinement at different encoder depths and explicitly models across-period temporal dependencies and within-period periodic dependencies. The two modules independently produce forecasts, which are fused through a learnable prediction-level gate. Experiments on 10 real-world datasets demonstrate that MUSE achieves state-of-the-art performance.

1 INTRODUCTION

Multivariate time-series forecasting aims to predict the future evolution of multiple correlated variables from historical observations, and plays a vital role in a wide range of real-world applications, including energy systems (Hong et al., 2014; Lago et al., 2021), traffic forecasting (Li et al., 2018; Yu et al., 2018), weather forecasting (Bi et al., 2023; Lam et al., 2023), and environmental monitoring (Zheng et al., 2015; Wu et al., 2026a). In recent years, numerous deep learning methods have been proposed and have substantially advanced time-series forecasting performance (Qiu et al., 2026; Salinas et al., 2020; Zhou et al., 2021; Liu et al., 2022b;a; Zhang & Yan, 2023).

Furthermore, large-scale pretrained models provide time-series forecasting with prior knowledge beyond the numerical modality. Conventional methods primarily learn temporal patterns from limited numerical observations, whereas large-scale pretrained models, such as LLMs and LVMs, acquire rich semantic, structural, and contextual priors from large-scale cross-domain data, complementing information that is difficult to capture from numerical representations alone and thereby enhancing the representation and generalization capabilities of forecasting models (Liu et al., 2025; Sun et al., 2026; Li et al., 2026; Zhang et al., 2026; Liang et al., 2024; Sun et al., 2024; Liu et al., 2024a).

In particular, LVM-based forecasting methods have demonstrated remarkable cross-domain transferability. VisionTS (Chen et al., 2025) reformulates time-series forecasting as an image reconstruction task and directly leverages an MAE pretrained on natural images from ImageNet (Deng et al., 2009; He et al., 2022), achieving competitive zero-shot forecasting performance without further adaptation to the numerical time-series domain. Subsequent works, such as DMMV (Shen et al., 2025a) and VisionTS++ (Shen et al., 2025b), further push forecasting performance to the state of the art. However, these methods still face two key challenges: how to balance independent visual representation spaces for individual variables with cross-variable dependency modeling, and how to better adapt visual representations to the distinctive structures induced by time-series imaging.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

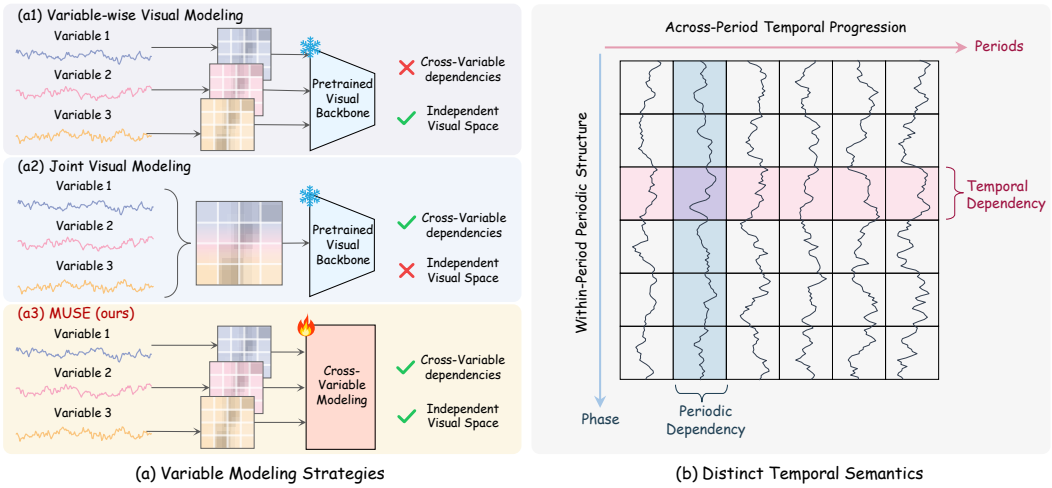

Figure 1: Motivation of MUSE. (a) Variable Modeling Strategies in the visual space. (b) Distinct temporal and periodic semantics of time-series images.

First, there is an inherent trade-off between variable representation capacity and cross-variable dependency modeling. According to how multivariate time series are organized in the visual space, existing LVM-based forecasting methods can be broadly divided into two categories. 1) *Variable-wise Visual Modeling* (Figure 1(a1)): each variable is independently transformed into an image and processed by a shared visual backbone, as adopted by VisionTS (Chen et al., 2025), ViTime (Yang et al., 2025), and DMMV (Shen et al., 2025a). This strategy allows each variable to fully exploit an independent visual representation space, but the isolated processing makes cross-variable dependencies difficult to capture explicitly. 2) *Joint Visual Modeling* (Figure 1(a2)): multiple variables are jointly organized into a single image, enabling their relationships to be modeled through shared visual representations, as in Time-VLM (Zhong et al., 2025) and VisionTS++ (Shen et al., 2025b). However, jointly accommodating multiple variables within a fixed image space requires them to share limited visual representation capacity. Consequently, these two paradigms struggle to simultaneously achieve sufficient variable representation and effective cross-variable dependency modeling.

Second, a substantial structural semantic gap exists between natural images and time-series images. The two axes of natural images primarily encode spatial positions, whereas time-series images carry distinct temporal semantics: one captures temporal progression across periods, while the other characterizes periodic structure within each period, as illustrated in Figure 1(b). An MAE pretrained on natural images is primarily optimized for modeling two-dimensional spatial relations and, when directly transferred to time-series images, lacks additional adaptation to their distinctive temporal–periodic structure, limiting its ability to fully capture temporal and periodic dependencies. Therefore, how to bridge this structural gap while keeping the visual backbone frozen and better adapt pretrained visual representations to time-series imaging constitutes the second key challenge.

Based on the above analysis, we propose **MUSE**, which performs *Dependency-Aware Adaptation of a Frozen Vision Backbone for Multivariate Time Series Forecasting*. With the pretrained MAE fully frozen, MUSE adapts visual representations from two perspectives: *variable-context dependencies and temporal–periodic dependencies*. Specifically, we first introduce the *Variable Context Refinement Module (VCR)*, which aggregates shared temporal information within each variable into variable-level context and further models contextual dependencies across variables, thereby incorporating cross-variable information while preserving independent visual spaces. We then introduce the *Temporal–Periodic Refinement Module (TPR)*, which performs lightweight adaptation at different depths of the frozen MAE Encoder through low-dimensional bottlenecks and models across-period temporal dependencies and within-period periodic dependencies, enabling pretrained visual representations to better accommodate the distinctive temporal–periodic structure of time-series images. Finally, the two modules independently produce forecasts, which are fused through a learnable prediction-level gate. Our main contributions are summarized as follows:

- We propose **MUSE**, a dependency-aware adaptation framework that transfers pretrained visual priors to multivariate time-series forecasting with a fully frozen visual backbone.
- We introduce VCR to model variable-context dependencies while preserving independent visual representation spaces.
- We introduce TPR, a lightweight bottleneck-based adaptation module that captures temporal and periodic dependencies to bridge the structural semantic gap between natural and time-series images.
- We conduct extensive experiments on 10 multivariate time-series datasets. The results show that MUSE consistently outperforms representative state-of-the-art baselines. Additionally, the code is available at `https://anonymous.4open.science/r/MUSE-728D/`.

2 Related Works

2.1 Multivariate Time Series Forecasting

Multivariate time series forecasting has been extensively studied. Early approaches mainly relied on statistical modeling (Sims, 1980), followed by machine learning methods for capturing nonlinear temporal patterns (Sapankevych & Sankar, 2009; Ahmed et al., 2010). In recent years, deep learning has become the dominant paradigm. Representative methods include LSTNet (Lai et al., 2018), Autoformer (Wu et al., 2021), FEDformer (Zhou et al., 2022), DLinear (Zeng et al., 2023), PatchTST (Nie et al., 2023), TimesNet (Wu et al., 2023), iTransformer (Liu et al., 2024b), Pathformer (Chen et al., 2024), TimeMixer (Wang et al., 2024a), and DUET (Qiu et al., 2025b). These methods have continuously improved forecasting performance, but are typically trained end-to-end from scratch on numerical time series, with limited use of the rich prior knowledge provided by large-scale cross-domain pretrained models.

2.2 LVM-Based Time Series Forecasting

With the development of visual models such as ResNet (He et al., 2016), VGG (Simonyan & Zisserman, 2015), Inception (Szegedy et al., 2015), ViT (Dosovitskiy et al., 2021), Swin Transformer (Liu et al., 2021), and MAE (He et al., 2022), transferring visual representations to time-series forecasting has attracted increasing attention. One line of work adopts channel-independent modeling. ViTime (Yang et al., 2025) explores a vision-based foundation model for time-series forecasting. VisionTS (Chen et al., 2025) reformulates forecasting as masked image reconstruction and leverages a pretrained MAE for prediction. DMMV (Shen et al., 2025a) builds upon VisionTS by introducing a numerical branch for complementary forecasting. Aurora (Wu et al., 2026b) develops a cross-domain multimodal time-series foundation model for zero-shot forecasting. Another line of work adopts channel-dependent modeling. Time-VLM (Zhong et al., 2025) and VisionTS++ (Shen et al., 2025b) both encode multiple variables into a single image to capture inter-variable relationships.

Overall, channel-independent methods preserve a complete visual representation space for each variable but lack explicit cross-variable interaction, while channel-dependent methods capture variable relationships at the cost of sharing a limited image space across variables. MUSE addresses both limitations by constructing an individual image for each variable while explicitly modeling inter-variable dependencies, and further refines temporal and periodic structures through lightweight adaptation over a fully frozen visual backbone.

3 Methodology

Problem Statement. Given a multivariate time series with V variables, the historical observations within a look-back window of length L are denoted as $\mathbf{X} = [\mathbf{x}_{t-L+1}, \ldots, \mathbf{x}_t] \in \mathbb{R}^{L\times V}$, where $\mathbf{x}_t \in \mathbb{R}^V$ denotes the observations of all variables at time step t. Multivariate time-series forecasting learns a mapping $f : \mathbb{R}^{L\times V} \to \mathbb{R}^{H\times V}$ to predict the next H steps, yielding the prediction, $\widehat{\mathbf{Y}} = f(\mathbf{X}) \in \mathbb{R}^{H\times V}$, with ground truth $\mathbf{Y} = [\mathbf{x}_{t+1}, \ldots, \mathbf{x}_{t+H}] \in \mathbb{R}^{H\times V}$.

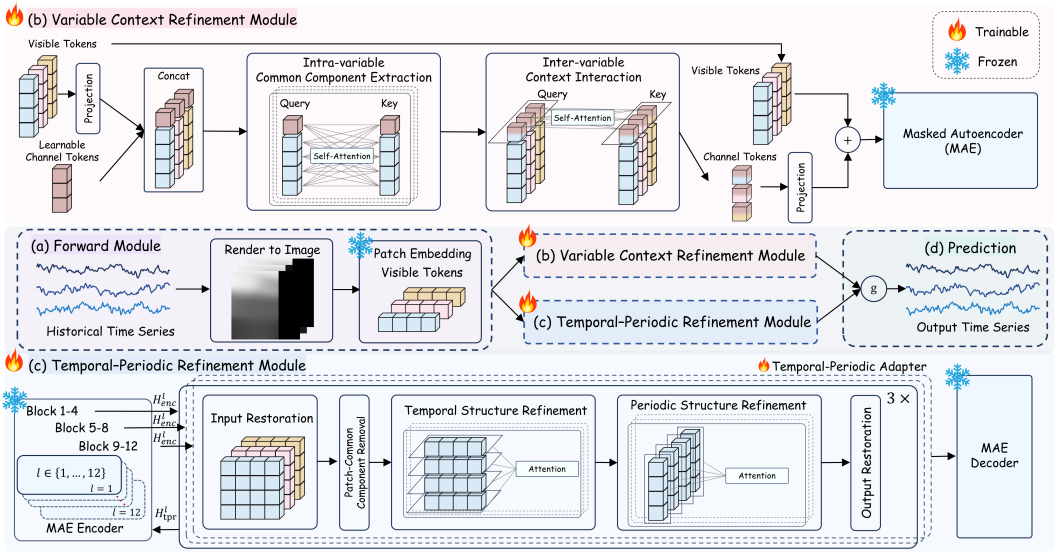

Figure 2: Overall architecture of MUSE. The historical multivariate time series is first processed by the Forward Module to obtain visible historical tokens, which are then fed in parallel into the Variable Context Refinement Module (VCR) and the Temporal–Periodic Refinement Module (TPR). VCR models variable context, while TPR refines temporal–periodic structures. The predictions of the two modules are finally fused through a learnable prediction-level gate.

3.1 Structure Overview

As illustrated in Figure 2, MUSE is built on a fully frozen pretrained MAE (He et al., 2022). First, the historical multivariate time series is processed by the Forward Module to obtain visible historical tokens, which are then fed into two complementary structure-aware modules: the **Variable Context Refinement Module (VCR)** and the **Temporal–Periodic Refinement Module (TPR)**.

VCR aggregates intra-variable information and models cross-variable dependencies to enrich the original visual representations with variable context, while TPR refines temporal and periodic structures at different depths of the frozen MAE Encoder. The two modules independently produce forecasts, which are fused through a learnable prediction-level gate. The details of each module will be introduced in the following sections.

3.2 Forward Module

As in existing LVM-based time-series forecasting methods such as VisionTS and DMMV (Chen et al., 2025; Shen et al., 2025a), MUSE transforms each variable into a 2D visual representation through the Forward Module. A detailed description is provided in Appendix A.

Let G denote the number of patches along each spatial axis and K_{in} the number of historical patch columns, giving $P = GK_{\text{in}}$ visible historical patches per variable. After the frozen Patch Embedding, the visible historical tokens are denoted as $\mathbf{T} \in \mathbb{R}^{V \times P \times E}$, where E is the MAE Encoder embedding dimension. $\mathbf{T}$ serves as the common input to VCR and TPR.

3.3 Variable Context Refinement Module

Historical patch tokens within each variable often contain shared temporal information, while dependencies may also exist across variables. However, period-based imaging processes variables independently, without explicitly aggregating such intra-variable information or modeling cross-variable dependencies, resulting in visual representations that lack variable-level context. Inspired by TimeXer (Wang et al., 2024b), we introduce the Variable Context Refinement Module (VCR),

which aggregates shared temporal information within each variable, models cross-variable dependencies, and uses the resulting variable context to refine the original patch representations.

Given the historical representation $\mathbf{T} \in \mathbb{R}^{V\times P\times E}$ produced by the frozen Patch Embedding, VCR first projects it into a low-dimensional feature space of dimension D:

$$\mathbf{M} = \mathrm{LayerNorm}\left(\mathbf{W}_d\,\mathrm{LayerNorm}(\mathbf{T})\right), \tag{1}$$

where $\mathbf{W}_d : \mathbb{R}^E \to \mathbb{R}^D$ denotes a trainable linear projection. The resulting patch memory $\mathbf{M} \in \mathbb{R}^{V\times P\times D}$ is used for subsequent variable-level information modeling, while the original visual representation $\mathbf{T}$ is retained for the final residual refinement.

Intra-variable Common Component Extraction. To capture temporal patterns shared across historical patch tokens within each variable, we introduce a shared learnable channel token $\mathbf{q} \in \mathbb{R}^D$ and replicate it across all variables, yielding $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_V] \in \mathbb{R}^{V\times D}$. For the v-th variable, its channel token $\mathbf{q}_v$ is concatenated with the corresponding patch memory $\mathbf{M}_v \in \mathbb{R}^{P\times D}$ to form $\mathbf{S}_v = [\mathbf{q}_v; \mathbf{M}_v] \in \mathbb{R}^{(1+P)\times D}$.

Multi-head self-attention is then performed independently within each variable:

$$[\widetilde{\mathbf{q}}_v; \widetilde{\mathbf{M}}_v] = \mathrm{LayerNorm}\left(\mathbf{S}_v + \mathrm{MultiHeadSelfAttention}_{\mathrm{intra}}\left(\mathrm{LayerNorm}(\mathbf{S}_v)\right)\right). \tag{2}$$

Through interaction with all historical patch tokens of the corresponding variable, the updated channel token $\widetilde{\mathbf{q}}_v \in \mathbb{R}^D$ aggregates shared temporal patterns into a compact variable-level representation. The same intra-variable self-attention parameters are shared across all variables.

Inter-variable Context Interaction. After obtaining a compact representation for each variable, we further model dependencies across variables. The updated channel tokens are organized as a variable-level sequence $\widetilde{\mathbf{Q}} = [\widetilde{\mathbf{q}}_1, \ldots, \widetilde{\mathbf{q}}_V] \in \mathbb{R}^{V\times D}$, and multi-head self-attention is performed along the variable dimension:

$$\mathbf{Q}' = \mathrm{LayerNorm}\left(\widetilde{\mathbf{Q}} + \mathrm{MultiHeadSelfAttention}_{\mathrm{inter}}\left(\mathrm{LayerNorm}(\widetilde{\mathbf{Q}})\right)\right). \tag{3}$$

This yields $\mathbf{Q}' = [\mathbf{q}'_1, \ldots, \mathbf{q}'_V] \in \mathbb{R}^{V\times D}$, where each $\mathbf{q}'_v$ incorporates contextual information from other variables.

The resulting channel token $\mathbf{q}'_v \in \mathbb{R}^D$ is transformed by a trainable write-back projection and mapped back to the MAE embedding space, producing a variable-specific correction $\mathbf{C}_v \in \mathbb{R}^E$:

$$\mathbf{C}_v = \mathrm{LinearProjection}_{D\to E}\left(\mathrm{WriteBackProjection}(\mathbf{q}'_v)\right). \tag{4}$$

Each $\mathbf{C}_v$ is broadcast along the patch dimension within the corresponding variable:

$$\mathbf{T}^{\mathrm{vcr}}_{v,p} = \mathbf{T}_{v,p} + \mathbf{C}_v, \qquad p = 1, \ldots, P. \tag{5}$$

This yields $\mathbf{T}^{\mathrm{vcr}} \in \mathbb{R}^{V\times P\times E}$, where each variable receives its own shared correction across historical patches. In this way, VCR incorporates intra-variable shared patterns and inter-variable context while preserving the original patch-level local structure.

After prepending the frozen CLS token, $\mathbf{T}^{\mathrm{vcr}}$ is processed by the frozen MAE Encoder and Decoder, yielding the VCR prediction $\widehat{\mathbf{Y}}_{\mathrm{vcr}} \in \mathbb{R}^{H\times V}$.

3.4 Temporal–Periodic Refinement Module

Period-based imaging converts one-dimensional time series into two-dimensional representations, preserving temporal information along both temporal and periodic axes. However, pretrained MAE self-attention is designed for natural images and does not explicitly distinguish these two axes with different temporal semantics. We therefore introduce the Temporal–Periodic Refinement Module (TPR) to model temporal dependencies across historical periods and periodic dependencies within each period. Inspired by parameter-efficient adaptation methods in computer vision such as AdaptFormer (Chen et al., 2022), TPR performs lightweight refinement through a bottleneck architecture in a low-dimensional space.

Unlike VCR, TPR directly operates on the original historical representation $\mathbf{T}$ with a channel-independent design. The frozen CLS token is prepended before the MAE Encoder. For depth-wise adaptation, the 12 frozen Encoder Blocks are divided into three stages: Blocks 1–4, 5–8, and 9–12. Each stage uses an independent Temporal–Periodic Adapter shared across its four Blocks.

Input Restoration. After the l-th Frozen Encoder Block, $l \in \{1, \ldots, 12\}$, let $\mathbf{H}^l_{\text{enc}} \in \mathbb{R}^{V\times(1+P)\times E}$ denote its output. The CLS token is retained unchanged, while the historical patch tokens in $\mathbf{H}^l_{\text{enc}}$ are restored to their original 2D arrangement, yielding $\mathbf{X}^l \in \mathbb{R}^{V\times G\times K_{\text{in}}\times E}$. Each (r, c) indexes a temporal–periodic patch location for subsequent modeling along both axes.

Patch-Common Component Removal. To emphasize temporal–periodic variations, TPR removes the representation component shared across historical patches. For the v-th variable, the mean representation is computed as

$$\overline{\mathbf{X}}^l_v = \frac{1}{GK_{\text{in}}} \sum_{r=1}^{G} \sum_{c=1}^{K_{\text{in}}} \mathbf{X}^l_{v,r,c}, \tag{6}$$

and subtracted from each patch:

$$\mathbf{R}^l_{v,r,c} = \mathbf{X}^l_{v,r,c} - \overline{\mathbf{X}}^l_v. \tag{7}$$

The centered representation $\mathbf{R}^l \in \mathbb{R}^{V\times G\times K_{\text{in}}\times E}$ is used for subsequent structural modeling, while $\mathbf{X}^l$ is retained for residual write-back.

$\mathbf{R}^l$ is then projected into a low-dimensional space of dimension D_{tp}:

$$\mathbf{Z}^l = \mathbf{W}_{\text{down}} \left(\text{LayerNorm} \left(\mathbf{R}^l \right) \right), \tag{8}$$

where $\mathbf{W}_{\text{down}} : \mathbb{R}^E \to \mathbb{R}^{D_{\text{tp}}}$ is the down-projection of the stage-specific Temporal–Periodic Adapter, yielding $\mathbf{Z}^l \in \mathbb{R}^{V\times G\times K_{\text{in}}\times D_{\text{tp}}}$.

Temporal Structure Refinement. Temporal Attention is applied along the historical time axis across periods. For a fixed variable v and periodic position r, the sequence $[\mathbf{Z}^l_{v,r,1}, \ldots, \mathbf{Z}^l_{v,r,K_{\text{in}}}]$ represents the same periodic position across consecutive historical periods, capturing its temporal progression over time. Multi-head self-attention is performed within this sequence.

To encode relative temporal relationships, for query position c_i and key position c_j, we define the directional offset $d_t = c_i - c_j$. The attention logit of the h-th head is

$$\ell^{t,(h)}_{ij} = \frac{\left\langle \mathbf{q}^{t,(h)}_i, \mathbf{k}^{t,(h)}_j \right\rangle}{\sqrt{d_h}} + \mathbf{B}^{(h)}_t[d_t], \tag{9}$$

where $\mathbf{q}^{t,(h)}_i$ and $\mathbf{k}^{t,(h)}_j$ denote the query and key vectors of the h-th Temporal Attention head, d_h is the head dimension, and $\mathbf{B}^{(h)}_t$ is a learnable temporal relative-position bias. The offset d_t encodes both relative distance and direction. The attention output is residually added to $\mathbf{Z}^l$, yielding $\mathbf{Z}^l_t$.

Periodic Structure Refinement. After Temporal Attention, Periodic Attention is applied along the phase axis within each historical period. For a fixed variable v and historical column c, the sequence $[\mathbf{Z}^l_{t,v,1,c}, \ldots, \mathbf{Z}^l_{t,v,G,c}]$ represents different periodic positions within the same historical period segment, capturing their within-period dependencies. Multi-head self-attention is performed within this sequence.

To preserve the periodic wrap-around continuity, for periodic positions r_i and r_j, we define the directional circular offset as $d_p = (r_i - r_j) \bmod G$. The attention logit of the h-th head is

$$\ell^{p,(h)}_{ij} = \frac{\left\langle \mathbf{q}^{p,(h)}_i, \mathbf{k}^{p,(h)}_j \right\rangle}{\sqrt{d_h}} + \mathbf{B}^{(h)}_p[d_p], \tag{10}$$

where $\mathbf{q}^{p,(h)}_i$ and $\mathbf{k}^{p,(h)}_j$ denote the query and key vectors of the h-th Periodic Attention head, and $\mathbf{B}^{(h)}_p$ is a learnable periodic relative-position bias. The circular offset preserves the adjacency between the beginning and end of a period while encoding directional relationships among periodic positions. The attention output is residually added to $\mathbf{Z}^l_t$, yielding $\mathbf{Z}^l_{\text{tp}}$.

Output Restoration. For the l-th Encoder Block, the refined low-dimensional representation is projected back to the MAE embedding space:

$$\Delta \mathbf{X}^l_{\text{raw}} = \mathbf{W}_{\text{up}} \left(\text{LayerNorm} \left(\mathbf{Z}^l_{\text{tp}} \right) \right), \tag{11}$$

where $\mathbf{W}_{\text{up}} : \mathbb{R}^{D_{\text{tp}}} \to \mathbb{R}^E$ is the up-projection of the stage-specific Temporal–Periodic Adapter.

The refinement strength is controlled by a learnable parameter s_l, and the resulting correction is injected through a residual connection:

$$\mathbf{X}^l_{\text{out}} = \mathbf{X}^l + \text{Sigmoid}(s_l)\, \Delta \mathbf{X}^l_{\text{raw}}. \tag{12}$$

The refined representation $\mathbf{X}^l_{\text{out}} \in \mathbb{R}^{V \times G \times K_{\text{in}} \times E}$ is flattened to the original patch order and combined with the retained CLS token, forming $\mathbf{H}^l_{\text{tpr}} \in \mathbb{R}^{V \times (1+P) \times E}$ for the next Frozen Encoder Block when $l < 12$.

After the final Encoder Block and TPR refinement, the representation passes through the Frozen Encoder final LayerNorm and the shared Frozen MAE Decoder to reconstruct the masked future region in the image space. The reconstruction is converted back to the time-series domain, yielding the TPR prediction $\widehat{\mathbf{Y}}_{\text{tpr}} \in \mathbb{R}^{H \times V}$.

3.5 Prediction

VCR and TPR produce predictions $\widehat{\mathbf{Y}}_{\text{vcr}}$ and $\widehat{\mathbf{Y}}_{\text{tpr}}$, respectively. We use a learnable prediction-level gate $g = \text{Sigmoid}(a)$, where a is a global learnable parameter. The final prediction is

$$\widehat{\mathbf{Y}} = g\, \widehat{\mathbf{Y}}_{\text{vcr}} + (1 - g) \widehat{\mathbf{Y}}_{\text{tpr}}. \tag{13}$$

We use MSE losses $\mathcal{L}_{\text{vcr}}$, $\mathcal{L}_{\text{tpr}}$, and $\mathcal{L}_{\text{gate}}$ to supervise VCR, TPR, and the gate, respectively, where the gate is optimized using detached branch predictions. The overall objective is

$$\mathcal{L} = \mathcal{L}_{\text{vcr}} + \mathcal{L}_{\text{tpr}} + \mathcal{L}_{\text{gate}}. \tag{14}$$

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate MUSE on 10 multivariate time-series datasets from the Time Series Forecasting Benchmark (TFB) (Qiu et al., 2024), including ETTh1, ETTh2, ETTm1, ETTm2, Weather, Electricity, AQWan, CzeLan, Wind, and ZafNoo, covering electricity, weather, air quality, wind energy, and natural environments. We follow the standard train/validation/test splits, using $60\%/20\%/20\%$ for the ETT datasets and AQWan, and $70\%/10\%/20\%$ for the remaining datasets (Qiu et al., 2024). Further dataset details are provided in Appendix B.1.

Baselines. We compare MUSE with eight representative state-of-the-art forecasting models, including the multimodal model DMMV-A (Shen et al., 2025a), the LVM-based model VisionTS (Chen et al., 2025), the LLM-based models Time-LLM (Jin et al., 2024) and GPT4TS (Zhou et al., 2023), and numerical time-series models DLinear (Zeng et al., 2023), PatchTST (Nie et al., 2023), TimesNet (Wu et al., 2023), and FEDformer (Zhou et al., 2022).

Implementation Details. We use Mean Squared Error (MSE) and Mean Absolute Error (MAE) as the evaluation metrics and uniformly set the forecasting horizon to $H \in \{96, 192, 336, 720\}$ for all datasets. All experiments of MUSE are conducted under the unified evaluation framework of the Time Series Forecasting Benchmark (TFB) (Qiu et al., 2024). All experiments of MUSE are implemented using PyTorch in Python 3.11.15 and executed on NVIDIA H20 GPUs. Further implementation details are provided in Appendix B.2.

Table 1: Average forecasting results on 10 multivariate time-series datasets over four horizons $H \in \{96, 192, 336, 720\}$. **Red**: the best, Blue: the 2nd best. Full results are available in Table 4 in Appendix C.2. Lower is better.

Models	MUSE (ours)		DMMV-A		VisionTS		Time-LLM		GPT4TS		DLinear		PatchTST		TimesNet		FEDformer	
Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	**0.384**	**0.405**	0.395	0.414	0.388	0.414	0.418	0.432	0.418	0.421	0.423	0.437	0.413	0.431	0.458	0.450	0.440	0.460
ETTh2	**0.325**	**0.372**	0.337	0.388	0.339	0.384	0.361	0.396	0.354	0.389	0.431	0.447	0.330	0.379	0.414	0.427	0.437	0.449
ETTm1	**0.330**	**0.362**	0.340	0.371	0.338	0.367	0.356	0.377	0.363	0.378	0.357	0.379	0.351	0.381	0.400	0.406	0.448	0.452
ETTm2	**0.245**	**0.309**	0.256	0.317	0.265	0.324	0.261	0.316	0.254	0.311	0.267	0.334	0.255	0.315	0.291	0.333	0.305	0.349
Electricity	**0.157**	0.251	0.158	**0.248**	0.164	0.257	0.165	0.259	0.170	0.263	0.166	0.264	0.162	0.253	0.193	0.295	0.214	0.327
Wind	**1.018**	**0.735**	1.140	0.751	1.121	0.760	1.173	0.780	1.220	0.768	1.077	**0.735**	1.120	0.765	1.251	0.781	1.254	0.815
Weather	**0.216**	0.261	0.217	0.256	0.262	0.293	0.244	0.270	0.227	**0.255**	0.249	0.300	0.226	0.264	0.259	0.287	0.309	0.360
AQWan	0.805	**0.494**	0.804	0.528	0.826	0.507	**0.800**	0.510	0.819	**0.494**	0.818	0.512	0.812	0.499	0.813	0.501	0.848	0.529
ZafNoo	**0.494**	**0.432**	0.507	0.450	0.538	0.452	0.561	0.483	0.547	0.456	0.496	0.451	0.512	0.465	0.537	0.465	0.578	0.499
CzeLan	**0.211**	**0.264**	0.226	0.294	0.241	0.290	0.229	0.285	0.248	0.271	0.285	0.343	0.227	0.290	0.224	0.285	0.303	0.364
1st Count	**9**	**8**	0	1	0	0	1	0	0	2	0	1	0	0	0	0	0	0

Table 2: Ablation studies for MUSE. TP denotes Temporal–Periodic. Lower is better; the best results are in bold.

Dataset	Full Model MSE / MAE	w/o VCR MSE / MAE	w/o TPR MSE / MAE	w/o PCR MSE / MAE	Replace TP Attention MSE / MAE
ETTh2	**0.373 / 0.416**	0.380 / 0.422	0.378 / 0.423	0.374 / 0.418	0.388 / 0.430
Weather	**0.299 / 0.326**	0.309 / 0.330	0.300 / 0.331	0.300 / 0.328	0.301 / 0.327
AQWan	**0.883 / 0.526**	0.899 / 0.535	0.893 / 0.535	0.888 / 0.532	0.887 / 0.529
ZafNoo	**0.564 / 0.468**	0.571 / 0.471	0.575 / 0.473	0.565 / 0.469	0.565 / 0.470
Average	**0.530 / 0.434**	0.540 / 0.440	0.537 / 0.441	0.532 / 0.437	0.535 / 0.439

4.2 Main Results

Comprehensive forecasting results are summarized in Table 1, with full results over all forecasting horizons provided in Appendix C.2. We make the following observations. 1) MUSE shows an overall advantage. Across the averaged results on 10 datasets, MUSE ranks first on 9 MSE and 8 MAE metrics. The remaining results also stay among the leading methods. 2) MUSE substantially improves the adaptation of pretrained visual models to time-series forecasting. Compared with existing visual-pretraining-based methods (e.g., VisionTS and DMMV-A), MUSE performs better on most datasets. Notably, **these gains are achieved with the MAE backbone fully frozen**. Using only lightweight structure-aware modules for temporal adaptation, MUSE effectively adapts frozen visual representations to multivariate forecasting. 3) Across 10 datasets spanning diverse domains and temporal characteristics, MUSE remains competitive against multimodal, LVM-based, LLM-based, and numerical forecasters. The consistent gains across these heterogeneous settings demonstrate stable performance and effective transfer of the cross-domain priors of the frozen visual backbone.

4.3 Ablation Studies

To assess the contribution of each key design in MUSE, we conduct ablation studies on four representative datasets at $H = 720$. We compare the full model with four variants: 1) *w/o VCR*: removes the Variable Context Refinement Module; 2) *w/o TPR*: removes the Temporal–Periodic Refinement Module; 3) *w/o PCR*: removes Patch-Common Component Removal from TPR in Section 3.4; and 4) *Replace TP Attention*: replaces the Temporal–Periodic (TP) Attention, comprising Temporal Attention in Temporal Structure Refinement and Periodic Attention in Periodic Structure Refinement of TPR (Section 3.4), with mean pooling along their corresponding axes.

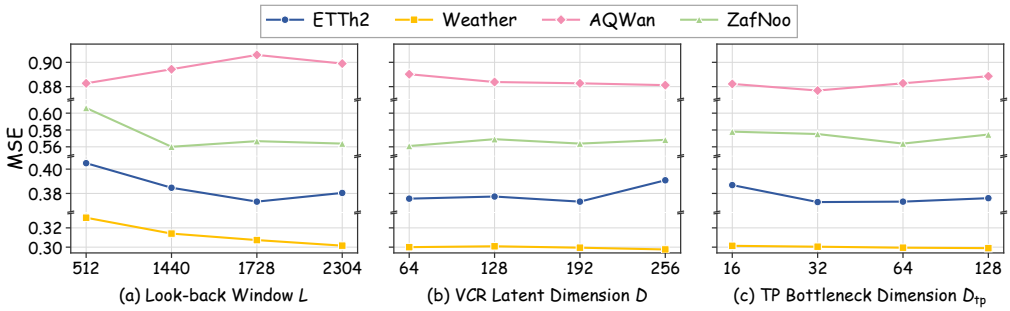

Figure 3: Parameter sensitivity studies of main hyper-parameters in MUSE. The broken vertical axes omit ranges containing no observations; lower is better.

As shown in Table 2, the full model performs best on all datasets and on average. Removing VCR or TPR increases the forecasting error, confirming the effectiveness of both modules. Removing PCR also degrades performance, demonstrating the benefit of eliminating components shared across historical patches. Replacing TP Attention with axis-wise mean pooling degrades performance, showing that adaptive attention captures temporal and periodic dependencies more effectively than uniform aggregation. Overall, MUSE benefits jointly from variable-context modeling, patch-common component removal, and adaptive temporal–periodic refinement, validating the overall design.

4.4 Parameter Sensitivity

We further analyze the sensitivity of MUSE to three key hyper-parameters under the forecasting horizon $H = 720$. The main observations are as follows. 1) Figure 3(a) shows the effect of the look-back window length L. Increasing the historical context generally improves forecasting performance, but the performance is non-monotonic in L, and the preferred range is dataset-specific. A longer look-back window provides richer historical context while compressing a longer temporal span into the fixed visual grid, making the choice of L depend on dataset-specific temporal characteristics. 2) Figure 3(b) studies the low-dimensional feature dimension D in VCR. Performance varies only marginally over $D \in \{64, 128, 192, 256\}$, while the default $D = 192$ achieves the best overall performance, indicating that VCR is robust to this dimension and requires little fine-grained tuning. 3) Figure 3(c) examines the TP bottleneck dimension D_{tp}, i.e., the latent dimension of the low-dimensional bottleneck in the Temporal–Periodic Adapter introduced in Section 3.4. Performance remains stable over $D_{\text{tp}} \in \{16, 32, 64, 128\}$, with the default $D_{\text{tp}} = 64$ achieving the best overall performance. This shows that TPR effectively refines temporal–periodic structures with a compact bottleneck and is robust to D_{tp}. Overall, MUSE is robust to both adaptation dimensions, while the look-back window shows dataset-dependent sensitivity due to interactions between temporal characteristics and the fixed visual representation.

5 Conclusions

In this paper, we propose MUSE, a dependency-aware adaptation framework for multivariate time-series forecasting with a fully frozen pretrained MAE. Specifically, the Variable Context Refinement Module (VCR) aggregates intra-variable temporal information and models cross-variable dependencies while preserving independent visual representation spaces. The Temporal–Periodic Refinement Module (TPR) performs lightweight refinement across encoder depths and explicitly captures across-period temporal and within-period periodic dependencies. The predictions from the two modules are fused through a learnable prediction-level gate. Experiments on 10 real-world datasets demonstrate that MUSE achieves state-of-the-art performance, validating dependency-aware adaptation of frozen visual representations.

AI USE STATEMENT

We used ChatGPT to help polish the manuscript and improve the clarity of the writing. All AI-assisted content was carefully reviewed by the authors. All methodological design and experimental analyses were conducted independently by the authors, and the use of AI was limited to auxiliary functions. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

This work focuses on methodological research for multivariate time-series forecasting and does not involve human subjects or the collection of personal data. All experiments are conducted on existing benchmark datasets following their established evaluation protocols. We do not release any new dataset containing sensitive or personally identifiable information. The proposed method is intended for general-purpose forecasting research, and we are not aware of any direct ethical risks arising from the methodology itself.

REPRODUCIBILITY STATEMENT

The experimental settings are described in Section 4.1, with further dataset and implementation details provided in Appendix B. The accompanying repository contains the complete code for MUSE, together with all hyperparameter settings, scripts, and experiment configurations, and is available at `https://anonymous.4open.science/r/MUSE-728D/`. All experimental results reported in this paper are obtained from actual runs and faithfully reflect the corresponding experimental outcomes.

REFERENCES

Nesreen K. Ahmed, Amir F. Atiya, Neamat El Gayar, and Hisham El-Shishiny. An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*, 29(5–6): 594–621, 2010.

Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Accurate medium-range global weather forecasting with 3d neural networks. *Nature*, 619(7970):533–538, 2023.

Mouxiang Chen, Lefei Shen, Zhuo Li, Xiaoyun Joy Wang, Jianling Sun, and Chenghao Liu. VisionTS: Visual masked autoencoders are free-lunch zero-shot time series forecasters. In *ICML*, pp. 8979–9007, 2025.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *ICLR*, 2024.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. AdaptFormer: Adapting vision transformers for scalable visual recognition. In *NeurIPS*, volume 35, pp. 16664–16678, 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pp. 248–255, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pp. 770–778, 2016.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pp. 16000–16009, 2022.

Tao Hong, Pierre Pinson, and Shu Fan. Global energy forecasting competition 2012. *International Journal of Forecasting*, 30(2):357–363, 2014.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *ICLR*, 2024.

Jesus Lago, Grzegorz Marcjasz, Bart De Schutter, and Rafał Weron. Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. *Applied Energy*, 293:116983, 2021.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *SIGIR*, pp. 95–104, 2018.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Oriol Vinyals, Jacklynn Stott, Alexander Pritzel, Shakir Mohamed, and Peter Battaglia. Learning skillful medium-range global weather forecasting. *Science*, 382(6677):1416–1421, 2023.

Lehui Li, Yuyao Wang, Jisheng Yan, Wei Zhang, Jinliang Deng, Haoliang Sun, Zhongyi Han, and Yongshun Gong. From text to forecasts: Bridging modality gap with temporal evolution semantic space. In *ICML*, 2026.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *ICLR*, 2018.

Yuxuan Liang, Haomin Wen, Yuqi Nie, Yushan Jiang, Ming Jin, Dongjin Song, Shirui Pan, and Qingsong Wen. Foundation models for time series analysis: A tutorial and survey. In *KDD*, pp. 6555–6565, 2024.

Chenxi Liu, Qianxiong Xu, Hao Miao, Sun Yang, Lingzheng Zhang, Cheng Long, Ziyue Li, and Rui Zhao. TimeCMA: Towards LLM-empowered multivariate time series forecasting via cross-modality alignment. In *AAAI*, volume 39, pp. 18780–18788, 2025.

Minhao Liu, Ailing Zeng, Muxi Chen, Zhijian Xu, Qiuxia Lai, Lingna Ma, and Qiang Xu. SCINet: Time series modeling and forecasting with sample convolution and interaction. In *NeurIPS*, volume 35, pp. 5816–5828, 2022a.

Shizhan Liu, Hang Yu, Cong Liao, Jianguo Li, Weiyao Lin, Alex X. Liu, and Schahram Dustdar. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *ICLR*, 2022b.

Xu Liu, Junfeng Hu, Yuan Li, Shizhe Diao, Yuxuan Liang, Bryan Hooi, and Roger Zimmermann. UniTime: A language-empowered unified model for cross-domain time series forecasting. In *WWW*, pp. 4095–4106, 2024a.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *ICLR*, 2024b.

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 *ICCV*, pp. 10012–10022, 2021.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *ICLR*, 2023.

Xiangfei Qiu, Jilin Hu, Lekui Zhou, Xingjian Wu, Junyang Du, Buang Zhang, Chenjuan Guo, Aoying Zhou, Christian S. Jensen, Zhenli Sheng, and Bin Yang. TFB: Towards comprehensive and fair benchmarking of time series forecasting methods. *Proc. VLDB Endow.*, 17(9):2363–2377, 2024.

Xiangfei Qiu, Zhe Li, Wanghui Qiu, Shiyan Hu, Lekui Zhou, Xingjian Wu, Zhengyu Li, Chenjuan Guo, Aoying Zhou, Zhenli Sheng, Jilin Hu, Christian S. Jensen, and Bin Yang. TAB: Unified benchmarking of time series anomaly detection methods. *Proc. VLDB Endow.*, 18(9):2775–2789, 2025a.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. DUET: Dual clustering enhanced multivariate time series forecasting. In *KDD*, pp. 1185–1196, 2025b.

Xiangfei Qiu, Hanyin Cheng, Xingjian Wu, Junkai Lu, Jilin Hu, Chenjuan Guo, Christian S. Jensen, and Bin Yang. A comprehensive survey of deep learning for multivariate time series forecasting: A channel strategy perspective. In *IJCAI*, 2026.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3): 1181–1191, 2020.

Nicholas I. Sapankevych and Ravi Sankar. Time series prediction using support vector machines: A survey. *IEEE Computational Intelligence Magazine*, 4(2):24–38, 2009.

ChengAo Shen, Wenchao Yu, Ziming Zhao, Dongjin Song, Wei Cheng, Haifeng Chen, and Jingchao Ni. Multi-modal view enhanced large vision models for long-term time series forecasting. In *NeurIPS*, 2025a.

Lefei Shen, Mouxiang Chen, Xu Liu, Han Fu, Xiaoxue Ren, Jianling Sun, Zhuo Li, and Chenghao Liu. VisionTS++: Cross-modal time series foundation model with continual pre-trained visual backbones. *arXiv preprint arXiv:2508.04379*, 2025b.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

Christopher A. Sims. Macroeconomics and reality. *Econometrica*, 48(1):1–48, 1980.

Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. TEST: Text prototype aligned embedding to activate LLM's ability for time series. In *ICLR*, 2024.

Siming Sun, Kai Zhang, Xuejun Jiang, Wenchao Meng, and Qinmin Yang. Markovian linguistic-temporal bridge: Unlocking the potential of LLMs for time series forecasting. In *ACL*, 2026.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pp. 1–9, 2015.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *ICLR*, 2024a.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. TimeXer: Empowering transformers for time series forecasting with exogenous variables. In *NeurIPS*, volume 37, pp. 469–498, 2024b.

Binqing Wu, Zongjiang Shang, Shiyu Liu, Jianlong Huang, Jiahui Xu, and Ling Chen. AirDDE: Multifactor neural delay differential equations for air quality forecasting. In *AAAI*, volume 40, pp. 15941–15949, 2026a.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *NeurIPS*, volume 34, pp. 22419–22430, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *ICLR*, 2023.

Xingjian Wu, Jianxin Jin, Wanghui Qiu, Peng Chen, Yang Shu, Bin Yang, and Chenjuan Guo. Aurora: Towards universal generative multimodal time series forecasting. In *ICLR*, 2026b.

Luoxiao Yang, Yun Wang, Xinqi Fan, Israel Cohen, Jingdong Chen, and Zijun Zhang. ViTime: Foundation model for time series forecasting powered by vision intelligence. *Transactions on Machine Learning Research*, 2025.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *IJCAI*, pp. 3634–3640, 2018.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *AAAI*, volume 37, pp. 11121–11128, 2023.

Fan Zhang, Shiming Fan, and Hua Wang. TimeSAF: Towards LLM-guided semantic asynchronous fusion for time series forecasting. In *ACL*, pp. 26267–26284, 2026.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *ICLR*, 2023.

Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, and Tianrui Li. Forecasting fine-grained air quality based on big data. In *KDD*, pp. 2267–2276, 2015.

Siru Zhong, Weilin Ruan, Ming Jin, Huan Li, Qingsong Wen, and Yuxuan Liang. Time-VLM: Exploring multimodal vision-language models for augmented time series forecasting. In *ICML*, pp. 78478–78497, 2025.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *AAAI*, volume 35, pp. 11106–11115, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *ICML*, pp. 27268–27286, 2022.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One Fits All: Power general time series analysis by pretrained lm. In *NeurIPS*, volume 36, pp. 43322–43355, 2023.

A Details of the Forward Module

MUSE follows the period-based imaging, spatial alignment, and deterministic masking procedure of VisionTS (Chen et al., 2025). Let F denote the periodicity, $I \times I$ the input resolution of the pretrained MAE, and p the patch size, such that $G = I/p$ patches lie along each spatial axis. Following VisionTS, we set $\epsilon = 10^{-5}$, $c_{\text{norm}} = 0.4$, and $c_{\text{align}} = 0.4$, and use bilinear interpolation for image resizing. The complete procedure for constructing the visible historical tokens is summarized in Algorithm 1.

Algorithm 1 Forward Module of MUSE

Input: Historical series $\mathbf{X} \in \mathbb{R}^{L \times V}$, forecasting horizon H, periodicity F, image size I, patch size p
Output: Visible historical tokens $\mathbf{T} \in \mathbb{R}^{V \times P \times E}$

$G \leftarrow I/p$
Variable-wise normalization
for $v = 1, \ldots, V$ **do**
  $\mu_v \leftarrow \text{Mean}(\mathbf{X}_{:,v})$
  $\sigma_v \leftarrow \sqrt{\text{Var}(\mathbf{X}_{:,v} - \mu_v) + \epsilon}$
  $\widetilde{\mathbf{x}}^{(v)} \leftarrow c_{\text{norm}}(\mathbf{X}_{:,v} - \mu_v)/\sigma_v$
end for
Image geometry and alignment
$\ell_{\text{left}} \leftarrow (F - (L \bmod F)) \bmod F$
$\ell_{\text{right}} \leftarrow (F - (H \bmod F)) \bmod F$
$\rho \leftarrow \dfrac{L + \ell_{\text{left}}}{L + \ell_{\text{left}} + H + \ell_{\text{right}}}$
$K_{\text{in}} \leftarrow \max(1, \lfloor \rho G c_{\text{align}} \rfloor)$
$K_{\text{out}} \leftarrow G - K_{\text{in}}$
$P \leftarrow GK_{\text{in}}$
Period-based imaging
for $v = 1, \ldots, V$ **do**
  $\widetilde{\mathbf{x}}^{(v)}_{\text{pad}} \leftarrow \text{ReplicatePadLeft}\left(\widetilde{\mathbf{x}}^{(v)}, \ell_{\text{left}}\right)$
  $\mathbf{A}^{(v)} \leftarrow \text{Reshape}\left(\widetilde{\mathbf{x}}^{(v)}_{\text{pad}}, F, \frac{L+\ell_{\text{left}}}{F}\right)$
  $\mathbf{A}^{(v)} \leftarrow \text{BilinearResize}\left(\mathbf{A}^{(v)}, I, K_{\text{in}}p\right)$
  $\mathbf{I}^{(v)}_{\text{gray}} \leftarrow \text{Concat}\left(\mathbf{A}^{(v)}, \mathbf{0}_{I \times K_{\text{out}}p}\right)$
  $\mathbf{I}^{(v)}_{\text{rgb}} \leftarrow \text{RepeatChannels}\left(\mathbf{I}^{(v)}_{\text{gray}}, 3\right)$
end for
Deterministic masking and visual embedding
Construct $\mathbf{\Omega} \in \{0,1\}^{G \times G}$ such that $\Omega_{r,c} = 0$ for $c \le K_{\text{in}}$ and $\Omega_{r,c} = 1$ otherwise
for $v = 1, \ldots, V$ **do**
  $\mathbf{U}^{(v)} \leftarrow \text{FrozenPatchEmbed}\left(\mathbf{I}^{(v)}_{\text{rgb}}\right) + \mathbf{E}_{\text{pos}}$
  $\mathbf{T}_v \leftarrow \text{SelectVisible}\left(\mathbf{U}^{(v)}, \mathbf{\Omega}\right) \in \mathbb{R}^{P \times E}$
end for
return $\mathbf{T} = [\mathbf{T}_1, \ldots, \mathbf{T}_V] \in \mathbb{R}^{V \times P \times E}$

The fixed mask retains the first K_{in} patch columns as visible historical observations and masks the remaining K_{out} columns, such that only historical patch tokens are retained as visible inputs.

B Experimental Details

B.1 Datasets

We conduct experiments on 10 commonly used multivariate time-series forecasting datasets, with their statistics summarized in Table 3. Specifically, (1) **ETT** consists of four subsets: ETTh1 and

Table 3: Statistics of multivariate datasets.

Dataset	Domain	Frequency	Length	Dim	Split
ETTh1	Electricity	1 hour	14,400	7	6:2:2
ETTh2	Electricity	1 hour	14,400	7	6:2:2
ETTm1	Electricity	15 mins	57,600	7	6:2:2
ETTm2	Electricity	15 mins	57,600	7	6:2:2
Electricity	Electricity	1 hour	26,304	321	7:1:2
Wind	Energy	15 mins	48,673	7	7:1:2
Weather	Environment	10 mins	52,696	21	7:1:2
AQWan	Environment	1 hour	35,064	11	6:2:2
ZafNoo	Nature	30 mins	19,225	11	7:1:2
CzeLan	Nature	30 mins	19,934	11	7:1:2

ETTh2 contain hourly observations, while ETTm1 and ETTm2 are recorded every 15 minutes. Each subset contains seven variables related to electricity transformers. (2) **Electricity** contains hourly electricity consumption records from 321 clients. (3) **Wind** contains seven wind power variables recorded at 15-minute intervals from 2020 to 2021. (4) **Weather** contains 21 meteorological variables recorded every 10 minutes throughout 2020. (5) **AQWan** contains hourly air-quality measurements from a monitoring station over four years, with 11 variables. (6) **ZafNoo** and **CzeLan** are derived from the Sapflux data project and contain sap-flow measurements and environmental variables recorded every 30 minutes, with 11 variables each. The dataset statistics and train/validation/test splits follow TFB (Qiu et al., 2024).

B.2 Implementation Details

1) We use the Adam optimizer and MSE as the training loss. VCR, TPR, and the prediction-level gate are supervised by separate MSE losses, which are equally weighted in the overall training objective. VCR, TPR, and the layer-wise injection parameters s_l use the main learning rate, while the gate uses an independent parameter group with a learning rate ten times the main learning rate. To ensure a fair comparison, we do not apply the "Drop Last" trick during testing (Qiu et al., 2024; 2025a).

2) At the beginning of training, the VCR write-back mapping is zero-initialized, while W_{up} in TPR is initialized with Xavier uniform initialization and the initial layer-wise injection strength is set to 0.05. The prediction-level gate is initialized with $g = 0.5$.

C Experimental Results

C.1 Limitations and Future Work

Despite the strong forecasting performance, MUSE still has several limitations. First, the architecture requires VCR and TPR to independently process visual representations and generate predictions, leading to higher memory consumption and inference latency. Second, although both modules are designed with explicit structural motivations, the learned dependency patterns and refinement behaviors inside the model have not yet been thoroughly characterized.

Future work could improve computational efficiency by reducing redundant computation or developing more compact mechanisms for combining variable-context and temporal–periodic modeling. It would also be valuable to further investigate how VCR and TPR modify frozen visual representations. Beyond this, extending the framework to capture more flexible multi-scale temporal and dynamic periodic structures may further improve its applicability to diverse time-series characteristics across different domains.

C.2 Full Results

Full forecasting results on the 10 multivariate time-series datasets under all forecasting horizons are reported in Table 4. The corresponding average results are summarized in Table 1 in the main text.

Table 4: Full forecasting results on 10 multivariate time-series datasets under horizons $H \in \{96, 192, 336, 720\}$. **Red**: the best, Blue: the 2nd best. Avg denotes the average over the four forecasting horizons. Lower is better.

Models		MUSE (ours)		DMMV-A		VisionTS		Time-LLM		GPT4TS		DLinear		PatchTST		TimesNet		FEDformer	
Metrics		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	**0.338**	**0.371**	0.354	0.389	0.357	0.388	0.376	0.402	0.370	0.389	0.375	0.399	0.370	0.399	0.384	0.402	0.376	0.419
	192	0.388	**0.403**	0.393	0.405	**0.387**	0.409	0.407	0.421	0.412	0.413	0.405	0.416	0.413	0.421	0.436	0.429	0.420	0.448
	336	0.405	0.417	**0.387**	**0.413**	0.392	0.414	0.430	0.438	0.448	0.431	0.439	0.443	0.422	0.436	0.491	0.469	0.459	0.465
	720	**0.407**	**0.431**	0.445	0.450	0.413	0.443	0.457	0.468	0.441	0.449	0.472	0.490	0.447	0.466	0.521	0.500	0.506	0.507
	Avg	**0.384**	**0.405**	0.395	0.414	0.388	0.414	0.418	0.432	0.418	0.421	0.423	0.437	0.413	0.431	0.458	0.450	0.440	0.460
ETTh2	96	**0.264**	**0.325**	0.294	0.349	**0.264**	**0.325**	0.286	0.346	0.280	0.335	0.289	0.353	0.274	0.336	0.340	0.374	0.358	0.397
	192	**0.320**	**0.363**	0.339	0.395	0.321	0.368	0.361	0.391	0.348	0.380	0.383	0.418	0.339	0.379	0.402	0.414	0.429	0.439
	336	0.342	0.385	**0.322**	0.384	0.386	0.417	0.390	0.414	0.380	0.405	0.448	0.465	0.329	**0.380**	0.452	0.452	0.496	0.487
	720	**0.373**	**0.416**	0.392	0.425	0.387	0.426	0.405	0.434	0.406	0.436	0.605	0.551	0.379	0.422	0.462	0.468	0.463	0.474
	Avg	**0.325**	**0.372**	0.337	0.388	0.339	0.384	0.361	0.396	0.354	0.389	0.431	0.447	0.330	0.379	0.414	0.427	0.437	0.449
ETTm1	96	0.285	0.330	**0.279**	0.329	0.289	**0.326**	0.291	0.341	0.300	0.340	0.299	0.343	0.290	0.342	0.338	0.375	0.379	0.419
	192	**0.317**	**0.353**	**0.317**	0.357	0.325	**0.353**	0.341	0.369	0.343	0.368	0.335	0.365	0.332	0.369	0.374	0.387	0.426	0.441
	336	**0.338**	**0.369**	0.351	0.381	0.349	0.377	0.359	0.379	0.376	0.386	0.369	0.386	0.366	0.392	0.410	0.411	0.445	0.459
	720	**0.379**	**0.397**	0.411	0.415	0.388	0.412	0.433	0.419	0.431	0.416	0.425	0.421	0.416	0.420	0.478	0.450	0.543	0.490
	Avg	**0.330**	**0.362**	0.340	0.371	0.338	0.367	0.356	0.377	0.363	0.378	0.357	0.379	0.351	0.381	0.400	0.406	0.448	0.452
ETTm2	96	**0.162**	0.252	0.172	0.260	0.163	0.251	**0.162**	**0.248**	0.163	0.249	0.167	0.269	0.165	0.255	0.187	0.267	0.203	0.287
	192	0.222	0.293	0.227	0.298	0.233	0.304	0.235	0.304	0.222	**0.291**	0.224	0.303	**0.220**	0.292	0.249	0.309	0.269	0.328
	336	**0.270**	**0.321**	0.272	0.327	0.304	0.349	0.280	0.329	0.273	0.327	0.281	0.342	0.274	0.329	0.321	0.351	0.325	0.366
	720	**0.327**	**0.371**	0.351	0.381	0.361	0.391	0.366	0.382	0.357	0.376	0.397	0.421	0.362	0.385	0.408	0.403	0.421	0.415
	Avg	**0.245**	**0.309**	0.256	0.317	0.265	0.324	0.261	0.316	0.254	0.311	0.267	0.334	0.255	0.315	0.291	0.333	0.305	0.349
Electricity	96	0.129	0.221	**0.126**	**0.213**	0.133	0.225	0.137	0.233	0.141	0.239	0.140	0.237	0.129	0.222	0.168	0.272	0.193	0.308
	192	0.146	0.240	**0.145**	**0.237**	0.151	0.245	0.152	0.247	0.158	0.253	0.153	0.249	0.157	0.240	0.184	0.289	0.201	0.315
	336	0.164	0.260	**0.162**	**0.254**	0.171	0.266	0.169	0.267	0.172	0.266	0.169	0.267	0.163	0.259	0.198	0.300	0.214	0.329
	720	**0.190**	**0.283**	0.197	0.286	0.200	0.293	0.200	0.290	0.207	0.293	0.203	0.301	0.197	0.290	0.220	0.320	0.246	0.355
	Avg	**0.157**	0.251	0.158	**0.248**	0.164	0.257	0.165	0.259	0.170	0.263	0.166	0.264	0.162	0.253	0.193	0.295	0.214	0.327
Wind	96	**0.864**	0.643	0.977	0.672	0.984	0.675	0.955	0.674	0.952	0.638	0.881	**0.632**	0.889	0.652	0.998	0.658	1.030	0.697
	192	**0.998**	0.727	1.125	0.747	1.091	0.741	1.178	0.771	1.157	0.742	1.034	**0.715**	1.076	0.747	1.214	0.752	1.275	0.807
	336	**1.074**	**0.769**	1.218	0.780	1.202	0.807	1.236	0.818	1.328	0.819	1.159	0.779	1.209	0.809	1.318	0.828	1.307	0.864
	720	**1.135**	**0.800**	1.240	0.805	1.207	0.817	1.322	0.855	1.441	0.873	1.233	0.815	1.304	0.851	1.474	0.887	1.402	0.890
	Avg	**1.018**	**0.735**	1.140	0.751	1.121	0.760	1.173	0.780	1.220	0.768	1.077	**0.735**	1.120	0.765	1.251	0.781	1.254	0.815
Weather	96	**0.142**	0.197	0.143	0.195	0.156	0.210	0.155	0.199	0.148	**0.188**	0.176	0.237	0.149	0.198	0.172	0.220	0.217	0.296
	192	0.188	0.242	**0.187**	0.242	0.229	0.275	0.223	0.261	0.192	**0.230**	0.220	0.282	0.194	0.241	0.219	0.261	0.276	0.336
	336	**0.233**	0.279	0.237	**0.273**	0.297	0.319	0.251	0.279	0.246	**0.273**	0.265	0.319	0.245	0.282	0.280	0.306	0.339	0.380
	720	**0.299**	0.326	0.302	**0.315**	0.366	0.368	0.345	0.342	0.320	0.328	0.333	0.362	0.314	0.334	0.365	0.359	0.403	0.428
	Avg	**0.216**	0.261	0.217	0.256	0.262	0.293	0.244	0.270	0.227	**0.255**	0.249	0.300	0.226	0.264	0.259	0.287	0.309	0.360
AQWan	96	0.739	0.465	0.762	0.506	0.760	0.481	**0.712**	0.468	0.745	**0.464**	0.756	0.481	0.745	0.470	0.791	0.488	0.796	0.508
	192	0.791	**0.487**	0.789	0.520	0.858	0.516	0.838	0.533	0.813	0.490	0.800	0.502	0.792	0.491	**0.779**	0.490	0.825	0.517
	336	0.809	**0.497**	0.805	0.529	0.820	0.507	**0.788**	0.504	0.818	**0.497**	0.823	0.516	0.819	0.503	0.814	0.505	0.863	0.537
	720	0.883	0.526	**0.859**	0.557	0.865	0.524	0.863	0.536	0.898	0.526	0.891	0.548	0.890	0.533	0.869	**0.519**	0.907	0.552
	Avg	0.805	**0.494**	0.804	0.528	0.826	0.507	**0.800**	0.510	0.819	**0.494**	0.818	0.512	0.812	0.499	0.813	0.501	0.848	0.529
ZaiNoo	96	**0.423**	**0.392**	0.436	0.410	0.465	0.411	0.470	0.436	0.452	0.399	0.434	0.411	0.444	0.426	0.479	0.424	0.475	0.441
	192	**0.475**	**0.422**	0.488	0.440	0.537	0.451	0.610	0.502	0.526	0.444	0.484	0.444	0.498	0.456	0.491	0.446	0.544	0.479
	336	**0.513**	**0.446**	0.537	0.466	0.574	0.470	0.556	0.482	0.576	0.472	0.518	0.464	0.530	0.480	0.551	0.479	0.595	0.517
	720	0.564	**0.468**	0.566	0.483	0.575	0.477	0.608	0.512	0.634	0.507	**0.548**	0.486	0.574	0.499	0.627	0.511	0.697	0.560
	Avg	**0.494**	**0.432**	0.507	0.450	0.538	0.452	0.561	0.483	0.547	0.456	0.496	0.451	0.512	0.465	0.537	0.465	0.578	0.499
CzeLan	96	**0.170**	**0.224**	0.184	0.255	0.180	0.242	0.191	0.249	0.195	0.233	0.211	0.289	0.183	0.251	0.176	0.237	0.231	0.310
	192	**0.203**	**0.255**	0.207	0.272	0.237	0.281	0.213	0.269	0.233	0.259	0.252	0.323	0.208	0.271	0.215	0.279	0.268	0.339
	336	0.234	**0.283**	0.236	0.304	0.265	0.308	0.239	0.299	0.264	**0.283**	0.317	0.366	0.243	0.302	**0.224**	0.288	0.298	0.361
	720	**0.238**	**0.295**	0.275	0.345	0.284	0.330	0.272	0.321	0.300	0.311	0.358	0.392	0.273	0.335	0.282	0.337	0.416	0.446
	Avg	**0.211**	**0.264**	0.226	0.294	0.241	0.290	0.229	0.285	0.248	0.271	0.285	0.343	0.227	0.290	0.224	0.285	0.303	0.364
1st Count		**34**	**32**	9	7	2	3	4	1	0	9	1	3	1	1	2	1	0	0