

A Limitation and Broader Impact

A.1 Discussion of Limitations

Adapting LVMs to LTSF is an emerging area of active research. This work serves a pioneering effort in investigating LVMs within an MMV framework for LTSF. As an initial exploration, we acknowledge some limitations in this work. First, one limitation of the current best LVM forecaster (e.g., VisionTS) is its sensitivity to segment length used in image construction due to its inductive bias, as discussed in §3 (Fig. 3). By incorporating an additional numerical view for modeling the global trend, the proposed DMMV is expected to alleviate this sensitivity. In our further analysis in Appendix D.2, we observe DMMV-A is less sensitive to the change of segment lengths than VisionTS on some datasets, but cannot consistently enhance the robustness over different datasets, despite the improved overall forecasting performance. This may be caused by the higher weights automatically allocated to $f_{\text{vis}}(\cdot)$ than $f_{\text{num}}(\cdot)$ by the gate fusion mechanism (Fig. 6), which could make the model prone to inherit the behavior of the LVM used in $f_{\text{vis}}(\cdot)$ to some extent, including its sensitivity to segment length, but with a less extent than a sole LVM. As such, a future work to improve DMMV is to reduce such sensitivity to an unnoticeable effect. Second, under our proposed BCMASK strategy, the vision backbone in $f_{\text{vis}}(\cdot)$ is reused three times during training and inference – once for forecasting and twice for reconstructing different masked parts of the look-back window. The triple use of $f_{\text{vis}}(\cdot)$ could lead to a non-trivial computational overhead. In this work, considering the remarkable performance improvement of BCMASK over other masking strategies (Table 2) and its possibly minimum use of $f_{\text{vis}}(\cdot)$, as analyzed in §3.2, we take it as the current solution. However, as a future work, we expect to further reduce the use of $f_{\text{vis}}(\cdot)$ and improve the efficiency, by methods such as joint backcasting and forecasting within a single forward pass or amortizing the use of LVMs across multiple time series samples. Finally, the proposed method shares a similar limitation of the existing LVM forecasters. The imaging process transforms an input time series into a 2D image-like representation to fit pre-trained LVMs, which typically expect a high-resolution input of size such as $224 \times 224 \times 3$. Therefore, upsampling is performed on a smaller image obtained from the patched time series to fit the input requirements of LVMs. While critical to compatibility, resizing may introduce subtle changes that may distort the original temporal structures to some extent. Thus an imaging process that can reflect the temporal patterns more accurately is in demand in future works.

A.2 Broader Impact

LTSF plays a vital role across various domains, including geoscience [1], neuroscience [3], energy [18], healthcare [28], and smart city [27]. This work proposes a novel MMV framework DMMV that integrates LVMs and a numerical forecaster, which could serve as a groundwork in the emerging area of LVM-based time series analysis and shed some lights on broader areas that integrate LLMs, VLMs, and large multi-modal models (LMMs) for future research on multi-modal and agentic time series analysis. This work does not involve sensitive data, legal risks, or ethical concerns. To the best of our knowledge, it does not adversely affect any specific population. The proposed method could serve as a general-purpose time series forecasting technique with a relatively broad applicability and social acceptability.

B Benchmark and Baseline

B.1 Benchmark Datasets

Following [53, 44, 30, 49, 36, 4], our experiments are conducted on 8 widely used LTSF benchmark datasets that cover a wide range of sampling frequencies, number of variates, levels of periodicity, and real-world domains. The four ETT datasets (ETT_h1, ETT_h2, ETT_m1, ETT_m2) record oil temperature from two electric transformers, sampled at 15-minute and hourly intervals. The Weather dataset collects measurements of meteorological indicators in Germany every 10 minutes. The Illness dataset keeps weekly counts of patients and the influenza-like illness ratio from the United States. The Traffic dataset measures hourly road occupancy rates from sensors on San Francisco freeways. The Electricity dataset records hourly electricity consumption of Portuguese clients. Table 3 summarizes the statistics of the datasets.

Table 3: Statistics of the benchmark datasets. “Dataset Size” is organized in (Train, Validation, Test).

Dataset	# Variates	Series Length	Dataset Size	Frequency
ETTh1	7	17420	(8545, 2881, 2881)	Hourly
ETTh2	7	17420	(8545, 2881, 2881)	Hourly
ETTm1	7	69680	(34465, 11521, 11521)	15 mins
ETTm2	7	69680	(34465, 11521, 11521)	15 mins
Weather	321	52696	(36792, 5271, 10540)	10 mins
Illness	7	966	(617, 74, 170)	Weekly
Traffic	862	17544	(12185, 1757, 3509)	Hourly
Electricity	21	26304	(18317, 2633, 5261)	Hourly

B.2 Baselines

In the following, we provide a brief description for each baseline method involved in our experiments.

- Time-VLM [52] integrates time series data with visual views and contextual texts using a pre-trained VLM, ViLT, to enhance forecasting performance.
- VisionTS [4] reformulates time series forecasting as an image reconstruction problem using an LVM, MAE, for zero/few/full-shot forecasting.
- Time-LLM [15] reprograms LLMs by aligning time series patches with text tokens, enabling time series forecasting without re-training LLMs.
- GPT4TS [55] demonstrates that frozen pretrained LLMs, *e.g.*, GPT, can be directly applied to a variety of time series tasks with strong performance.
- CALF [24] adapts LLMs to time series forecasting via cross-modal fine-tuning, bridging the distribution gap between textual and temporal data.
- CycleNet [21] enhances LTSF by explicitly modeling the periodic patterns in time series through a residual cycle forecasting technique.
- PatchTST [30] introduces a patching strategy and a channel-independence strategy for LTSF. It uses patches of time series as the input to a Transformer to capture the temporal dependency of semantically meaningful tokens (*i.e.*, patches).
- TimesNet [43] transforms an input time series into a 2D image-like representation and models temporal variations in the image using inception-like blocks for time series analysis.
- DLinear [49] decomposes an input time series into trend and seasonal components, each of which is modeled by linear layers for time series forecasting.
- FEDformer [54] incorporates frequency-enhanced attention mechanisms by combining Fourier transforms with seasonal-trend decomposition in a Transformer framework.
- Autoformer [44] introduces an auto-correlation mechanism within a Transformer architecture to capture long-term dependencies in time series data.
- Stationary [25] combines series stationarization and de-stationary attention mechanisms to solve the over-stationarization problem in time series forecasting.
- ETSformer [42] decomposes an input time series into interpretable components with exponential smoothing attention and frequency attention for time series forecasting.
- Informer [53] proposes a ProbSparse self-attention mechanism to reduce the computational complexity of LTSF with Transformer models.

C Implementation Details

C.1 Pre-trained LVM Checkpoints

As described in §3.3, $f_{\text{vis}}(\cdot)$ uses pre-trained LVMs. For MAE, we use the checkpoint released by *Meta Research*¹, which was pretrained on $224 \times 224 \times 3$ sized images from *ImageNet-1K* [5] with

¹<https://github.com/facebookresearch/mae>

Algorithm 1: The Training Algorithm of DMMV-A

Input: training dataset $\mathcal{D}_{\text{train}} = \{\mathbf{X}_i, \mathbf{Y}_i\}_{i=1}^n$, where $\mathbf{X}_i \in \mathbb{R}^{D \times T}$ is an MTS, $\mathbf{Y}_i \in \mathbb{R}^{D \times H}$ is the ground truth of forecast

Output: model parameters of DMMV-A

- 1 Load pre-trained $f_{\text{vis}}(\cdot)$ and freeze its weights
- 2 Randomly initialize $f_{\text{num}}(\cdot)$ and the gating parameter g
- /* stage 1: numerical forecaster training */*
- 3 **for** $i \leftarrow 1$ to MaxEpoch **do**
- 4 **for** (\mathbf{X}, \mathbf{Y}) in $\text{Dataloader}(\mathcal{D}_{\text{train}})$ **do**
- 5 */* channel-independence strategy is applied in the following */*
- 6 $\hat{\mathbf{X}}_{\text{season}}, \hat{\mathbf{Y}}_{\text{season}} \leftarrow f_{\text{vis}}(\mathbf{X}, \text{BCMASK})$ *// backcast/forecast seasonal part*
- 7 $\mathbf{X}_{\text{trend}} \leftarrow \mathbf{X} - \hat{\mathbf{X}}_{\text{season}}$ *// extract trend component*
- 8 $\hat{\mathbf{Y}}_{\text{trend}} \leftarrow f_{\text{num}}(\mathbf{X}_{\text{trend}})$ *// forecast trend with $f_{\text{num}}(\cdot)$*
- 9 $\hat{\mathbf{Y}} \leftarrow g \circ \hat{\mathbf{Y}}_{\text{season}} + (1 - g) \circ \hat{\mathbf{Y}}_{\text{trend}}$ *// gate fusion*
- 10 Calculate $\ell_{\text{MSE}}(\hat{\mathbf{Y}}, \mathbf{Y})$ *// calculate MSE loss as specified in §3.3*
- 11 Update model parameters of $f_{\text{num}}(\cdot)$ and g
- 12 **if** *Early stopping condition is TRUE* **then**
- 13 Break
- 14 **end**
- 15 **end**
- /* stage 2: joint training */*
- 16 Unfreeze the norm layers in $f_{\text{vis}}(\cdot)$
- 17 **for** $i \leftarrow 1$ to MaxEpoch **do**
- 18 **for** (\mathbf{X}, \mathbf{Y}) in $\text{Dataloader}(\mathcal{D}_{\text{train}})$ **do**
- 19 */* Repeat lines 5-9 */*
- 20 Update model parameters of $f_{\text{num}}(\cdot)$, norm layers in $f_{\text{vis}}(\cdot)$, and parameter g
- 21 **if** *Early stopping condition is TRUE* **then**
- 22 Break
- 23 **end**
- 24 **end**

ViT-Base Backbone. For SimMIM, we adopt the checkpoint released by *Microsoft*², which has the same pretraining setting as aforementioned for MAE. For these two LVM backbones, the base versions are adopted to balance the performance and computational costs.

C.2 Training Details

For training the proposed DMMV-S and DMMV-A models, we adopt AdamW optimizer throughout the experiments. The batch size is set to 64 for the ETT datasets and Illness dataset, and set to 8 for the other three datasets to balance training stability and memory consumption.

For both DMMV-S and DMMV-A, we propose a two-stage training scheme to facilitate effective integration of numerical and visual features:

- **Stage 1** (Numerical forecaster training). In this stage, we freeze all parameters of $f_{\text{vis}}(\cdot)$ and train $f_{\text{num}}(\cdot)$ only. This warm-up step prevents $f_{\text{vis}}(\cdot)$ from updating with unstable gradients caused by the random representations from the under-trained $f_{\text{num}}(\cdot)$. In this stage, the learning rate is set to 0.01. The training runs up to a maximum of 50 epochs on the training set. Early stopping is applied with a patience of 10 epochs.

²<https://github.com/microsoft/SimMIM>

- **Stage 2** (Joint training). In this stage, we unfreeze the layer normalization parameter in $f_{\text{vis}}(\cdot)$ and jointly train them with $f_{\text{num}}(\cdot)$ to enable deep fusion of visual and numerical views. The learning rate is reduced to 0.005 to preserve learned features and stabilize training. The training at this stage runs up to 5 epochs. Early stopping is applied with a patience of 2 epochs.

The detailed training algorithm of DMMV-A is summarized in Algorithm 1.

C.3 Running Environment

The experiments are conducted on a Linux server (kernel 5.15.0-139) with 8x NVIDIA RTX 6000 Ada GPUs (48 GB each). The environment uses Python 3.12.8, PyTorch 2.5.1 with CUDA 12.4 and cuDNN 9.1. The key libraries include NumPy 2.1.3, Pandas 2.2.3, Matplotlib 3.10.0, SciPy 1.15.1, scikit-learn 1.6.1, and torchvision 0.20.1.

D More Experimental Results

D.1 Comparison with All Baselines

Table 4 provides the full results of comparing DMMV-A and DMMV-S with all of the 14 baseline methods, which complements Table 1 in the paper. In Table 4, Time-VLM’s results on Illness dataset is marked by “–” since its paper doesn’t report the results and its code is not publicly available at the time of this experiment. CycleNet’s paper doesn’t report its results on Illness dataset, so we run its code and reproduce its results on Illness dataset in Table 4.

From Table 4, we can observe that DMMV-A maintains a clear advantage when compared against all of the baseline methods. It achieves 41 first-place results, significantly surpassing the second-best method. Additionally, taking a closer look at all compared methods, MMV-based methods LVM-based methods, and decomposition-based methods demonstrate superiority over other baseline methods. This suggests the synergy of MMV framework, LVMs, and decomposition strategy, which are explored by the proposed DMMV model.

D.2 Further Analysis of The Inductive Bias

A contribution of our work lies in the in-depth analysis of an inductive bias of the current best LVM forecasters. In §3, we have discussed the impact of the alignment of the segment length and the period of time series on model performance. We find that the LVM exhibits a strong *inter-period consistency* when applied to synthetic data. The function of the synthetic time series is $x(t) = A(t) \cdot \sin\left(\frac{2\pi t}{P}\right)$, where the period P is set to 24 and the amplitude function $A(t)$ decreases linearly over time. The forecasts are more accurate when the segment length is a multiple of the period (*e.g.*, 24, 48) than other values. This section provides detailed quantitative results on the synthetic data in Table 5. From Table 5, the fluctuations in MSEs and MAEs across different segment lengths other than 24 and 48 support the findings of the inductive bias toward “forecasting periods”.

In addition, we evaluate the performance of the proposed method DMMV-A and VisionTS *w.r.t.* varying segment lengths to compare their robustness to the change of segment length. Fig. 10 summarizes the results in terms of MSE on four benchmark datasets, where the segment length varies from $\frac{P}{6}$ to $\frac{6P}{6}$ and P is a period of the input time series. From Fig. 10, we have several observations. First, DMMV-A consistently outperforms VisionTS, validating the effectiveness of the proposed MMV framework. Second, in contrast to VisionTS, DMMV-A exhibits a better robustness to the change of segment length on ETTh1 and Weather datasets, but has a similar sensitivity to the change of segment length as VisionTS on ETTm1 and Illness datasets. This implies that by incorporating $f_{\text{num}}(\cdot)$, DMMV-A can alleviate $f_{\text{vis}}(\cdot)$ ’s sensitivity to the inductive bias to some extent. However, the current DMMV-A does not fully mitigate this limitation, suggesting a future work for method development as discussed in Appendix A.

Table 4: Full LTSF performance of the compared methods on the benchmark datasets. Lower MSE and MAE indicate better performance. The best performance is highlighted in **red**. Time-VLM results on the Illness dataset are unavailable in [52]. Its code was not publicly available at the time of this experiment. As such, its results on Illness dataset are marked by “-”.

	Model	DMV-A	DMV-S	Time-VLM	VisionTS	Time-LLM	GPT4S	CALF	CycleNet	PatchTST	TimesNet	DLinear	FEDformer	Autoformer	Stationary	ETSformer	Informr
ETTm1	Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
	96	0.354	0.389	0.350	0.386	0.376	0.402	0.370	0.389	0.370	0.399	0.376	0.399	0.376	0.419	0.494	0.479
	192	0.393	0.405	0.399	0.420	0.397	0.415	0.395	0.407	0.413	0.421	0.413	0.429	0.406	0.415	0.534	0.504
	336	0.387	0.413	0.399	0.415	0.420	0.421	0.419	0.421	0.430	0.438	0.448	0.431	0.451	0.440	0.588	0.574
	720	0.445	0.450	0.472	0.441	0.458	0.460	0.457	0.468	0.441	0.449	0.476	0.466	0.521	0.500	0.643	0.616
ETTm2	Avg.	0.395	0.414	0.405	0.426	0.405	0.420	0.407	0.419	0.418	0.432	0.431	0.442	0.413	0.458	0.570	0.537
	96	0.294	0.349	0.286	0.340	0.335	0.288	0.334	0.286	0.336	0.340	0.374	0.358	0.397	0.346	0.388	0.476
	192	0.339	0.395	0.331	0.387	0.326	0.373	0.349	0.380	0.361	0.391	0.383	0.418	0.429	0.439	0.452	0.512
	336	0.322	0.384	0.309	0.378	0.357	0.406	0.364	0.398	0.390	0.414	0.380	0.405	0.361	0.394	0.471	0.493
	720	0.332	0.425	0.430	0.462	0.412	0.449	0.403	0.431	0.405	0.434	0.406	0.436	0.406	0.431	0.562	0.560
ETTm1	Avg.	0.337	0.388	0.339	0.397	0.341	0.391	0.351	0.386	0.361	0.396	0.354	0.389	0.351	0.384	0.516	0.500
	96	0.279	0.329	0.296	0.349	0.304	0.346	0.323	0.350	0.340	0.338	0.375	0.343	0.379	0.345	0.386	0.398
	192	0.317	0.357	0.328	0.370	0.332	0.366	0.327	0.362	0.341	0.369	0.343	0.368	0.341	0.369	0.444	0.408
	336	0.351	0.381	0.369	0.393	0.364	0.383	0.354	0.382	0.359	0.379	0.376	0.386	0.361	0.386	0.464	0.435
	720	0.411	0.415	0.401	0.414	0.402	0.410	0.411	0.415	0.433	0.419	0.431	0.416	0.420	0.478	0.450	0.462
ETTm2	Avg.	0.340	0.371	0.349	0.382	0.351	0.376	0.344	0.373	0.356	0.377	0.363	0.378	0.351	0.381	0.452	0.425
	96	0.172	0.260	0.164	0.254	0.160	0.250	0.174	0.262	0.162	0.248	0.163	0.249	0.177	0.255	0.187	0.267
	192	0.227	0.298	0.217	0.293	0.215	0.291	0.228	0.297	0.235	0.304	0.222	0.291	0.245	0.300	0.224	0.303
	336	0.272	0.327	0.273	0.332	0.270	0.325	0.281	0.337	0.280	0.329	0.273	0.327	0.309	0.341	0.269	0.325
	720	0.351	0.381	0.362	0.393	0.348	0.378	0.384	0.410	0.366	0.382	0.357	0.376	0.402	0.395	0.363	0.403
Illness	Avg.	0.256	0.317	0.254	0.318	0.248	0.311	0.267	0.327	0.261	0.316	0.254	0.311	0.283	0.323	0.267	0.332
	24	1.409	0.754	1.638	0.838	-	1.613	0.834	1.792	0.807	1.869	0.823	1.460	0.788	2.255	1.017	1.319
	36	1.290	0.745	1.323	0.753	-	1.316	0.750	1.833	0.833	1.853	0.854	1.573	0.837	2.121	0.950	1.430
	48	1.499	0.810	1.644	0.851	-	1.548	0.818	2.269	1.012	1.886	0.855	1.784	0.890	2.130	1.024	1.553
	60	1.428	0.773	1.473	0.810	-	1.450	0.783	2.177	0.925	1.877	0.877	1.982	0.962	2.185	0.997	1.470
Electricity	Avg.	1.407	0.771	1.520	0.813	-	1.482	0.796	2.018	0.894	1.871	0.852	1.700	0.869	2.187	0.992	1.443
	96	0.126	0.213	0.165	0.267	0.142	0.245	0.127	0.217	0.137	0.233	0.141	0.239	0.147	0.240	0.128	0.223
	192	0.145	0.237	0.172	0.276	0.157	0.260	0.148	0.237	0.152	0.247	0.158	0.253	0.163	0.254	0.144	0.237
	336	0.162	0.254	0.190	0.296	0.174	0.276	0.163	0.253	0.169	0.267	0.172	0.266	0.178	0.270	0.160	0.254
	720	0.197	0.286	0.242	0.344	0.214	0.308	0.199	0.293	0.200	0.290	0.207	0.293	0.215	0.300	0.198	0.287
Weather	Avg.	0.158	0.248	0.192	0.296	0.172	0.272	0.159	0.250	0.165	0.259	0.170	0.263	0.176	0.266	0.158	0.250
	96	0.143	0.195	0.168	0.218	0.148	0.200	0.146	0.191	0.155	0.199	0.148	0.188	0.168	0.207	0.167	0.221
	192	0.187	0.242	0.220	0.259	0.193	0.240	0.194	0.238	0.223	0.261	0.192	0.230	0.216	0.251	0.212	0.258
	336	0.237	0.273	0.267	0.304	0.243	0.281	0.243	0.275	0.251	0.279	0.246	0.273	0.271	0.292	0.260	0.293
	720	0.302	0.315	0.322	0.343	0.312	0.332	0.318	0.328	0.345	0.342	0.320	0.328	0.350	0.345	0.328	0.339
Traffic	Avg.	0.217	0.256	0.244	0.281	0.224	0.263	0.225	0.258	0.224	0.274	0.242	0.278	0.226	0.264	0.259	0.287
	96	0.344	0.237	0.362	0.353	0.303	0.290	0.346	0.312	0.302	0.267	0.306	0.264	0.416	0.274	0.397	0.278
	192	0.346	0.249	0.385	0.363	0.305	0.296	0.376	0.245	0.240	0.212	0.268	0.243	0.426	0.281	0.431	0.276
	336	0.387	0.286	0.386	0.362	0.305	0.291	0.405	0.245	0.241	0.206	0.324	0.289	0.302	0.293	0.426	0.289
	720	0.433	0.284	0.436	0.292	0.333	0.432	0.293	0.432	0.296	0.440	0.305	0.436	0.315	0.466	0.315	0.463
# Wins	Avg.	0.382	0.257	0.395	0.268	0.419	0.304	0.386	0.256	0.422	0.281	0.421	0.289	0.391	0.264	0.620	0.336

Table 5: Forecasting performance of an LVM *w.r.t.* varying segment length on a synthetic dataset. The function of the synthetic time series is $x(t) = A(t) \cdot \sin\left(\frac{2\pi t}{P}\right)$, where the period $P = 24$ and the amplitude function $A(t)$ decreases linearly over time.

Segment Length	16	20	24	28	32	36	40	44	48
MSE	0.043	0.099	0.001	0.147	0.154	0.143	0.221	0.114	0.002
MAE	0.177	0.257	0.024	0.342	0.347	0.315	0.408	0.289	0.045

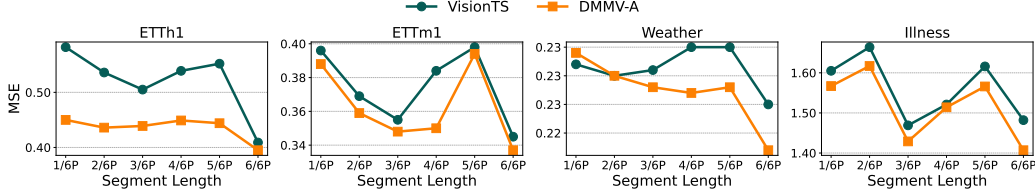


Figure 10: MSE Performance of DMMV-A and VisionTS *w.r.t.* varying segment length that is used in image construction. The x -axis indicates the segment length varies from $\frac{1}{6}$ period to $\frac{6}{6}$ period.

D.3 Ablation Study

In Table 2 (§4.2), we provide ablation analyses for DMMV-A. Table 6 provides the ablation analysis for DMMV-S, where MSE and MAE are averaged over different prediction lengths. In addition, Tables 7 (Table 8) includes the full results for Table 2 (Table 6) with all prediction lengths.

In Table 6, from (a), replacing the linear numerical forecaster with PatchTST can slightly improve the performance of DMMV-S, likely because DMMV-S relies more on the predictions from the numerical view than visual view (Fig. 6). Therefore, in this case, increasing the complexity of the numerical model can improve the ability of $f_{\text{num}}(\cdot)$ and finally improve the overall performance. From (b), replacing MAE with SimMiM reduces the overall performance, this is the same as the findings in Table 2 for DMMV-A. From (c), gate-based fusion outperforms simple summation for DMMV-S, highlighting the effectiveness of gate fusion. From (d), fine-tuning the norm layers of $f_{\text{vis}}(\cdot)$ improves the performance for DMMV-S, suggesting the used fine-tuning strategy.

Table 6: Ablation analysis of DMMV-S. MSE and MAE are averaged over different prediction lengths. Lower MSE and MAE are better. “Improvement” of each ablation is relative to DMMV-S.

Dataset (→)	ETTh1		ETTm1		Illness		Weather	
Method (↓), Metric (→)	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
DMMV-S	0.405	0.426	0.349	0.382	1.520	0.813	0.244	0.281
(a) $f_{\text{num}}(\cdot) \rightarrow \text{Transformer}$	0.402	0.423	0.342	0.376	1.544	0.841	0.229	0.264
Improvement	0.74%	0.47%	2.01%	1.31%	-1.65%	-3.44%	6.15%	6.41%
(b) $f_{\text{vis}}(\cdot) \rightarrow \text{SimMiM}$	0.415	0.423	0.355	0.382	1.810	0.875	0.233	0.272
Improvement	-2.47%	0.47%	-1.72%	0.00%	-19.16%	-7.63%	4.92%	3.20%
(c) Gate \rightarrow Sum	0.419	0.435	0.355	0.379	1.453	0.790	0.256	0.297
Improvement	-3.46%	-2.24%	-2.01%	0.52%	4.34%	2.95%	-4.51%	-5.69%
(d) Freeze $f_{\text{vis}}(\cdot)$	0.436	0.442	0.368	0.386	2.125	0.969	0.251	0.288
Improvement	-7.41%	-3.76%	-5.75%	-1.05%	-39.83%	-19.07%	-2.46%	-2.14%

D.4 Additional Visualizations on Decomposition

Fig. 11 and Fig. 12 several more examples the decomposed time series of DMMV-S and DMMV-A. Fig. 11 illustrates a case where the series has a localized periodic anomaly at time step around 192, which poses a challenge for detecting periodic patterns. In this case, DMMV-A effectively suppresses the influence of the anomaly and extracts a clear periodic pattern from the time series. In contrast, DMMV-S is affected by the anomaly and fails to capture a smooth periodic pattern. Fig. 12 is an example with weak periodicity, where the periodic signal is either faint or overwhelmed by trend. In this case, DMMV-A is able to extract and utilize the underlying periodicity to produce reasonable forecasts, which is better than DMMV-S, suggesting the importance of the

Table 7: Full results of the ablation analysis of DMMV-A. Lower MSE and MAE are better. The Illness dataset uses prediction lengths of $\{24, 36, 48, 60\}$ due to its short time series (in total 966 time steps), which is different from the prediction lengths of other datasets.

Dataset(\rightarrow) Method(\downarrow), Metric(\rightarrow)	Length	ETTh1		ETTm1		Illness		Weather	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
DMMV-A	96	0.354	0.389	0.279	0.329	1.409	0.754	0.143	0.195
	192	0.393	0.405	0.317	0.357	1.290	0.745	0.187	0.242
	336	0.387	0.413	0.351	0.381	1.499	0.810	0.237	0.273
	720	0.445	0.450	0.411	0.415	1.428	0.773	0.302	0.315
	Avg.	0.395	0.414	0.340	0.371	1.407	0.771	0.217	0.256
(a) $f_{\text{num}}(\cdot) \rightarrow \text{Transformer}$	96	0.357	0.389	0.279	0.329	1.604	0.823	0.145	0.193
	192	0.407	0.420	0.318	0.359	1.250	0.742	0.187	0.239
	336	0.389	0.411	0.352	0.382	1.555	0.803	0.241	0.283
	720	0.474	0.462	0.407	0.416	1.359	0.774	0.301	0.326
	Avg.	0.407	0.421	0.339	0.372	1.442	0.786	0.219	0.260
(b) $f_{\text{vis}}(\cdot) \rightarrow \text{SimMiM}$	96	0.358	0.383	0.301	0.348	1.729	0.832	0.145	0.194
	192	0.405	0.41	0.325	0.363	1.643	0.734	0.192	0.242
	336	0.412	0.414	0.354	0.383	1.689	0.845	0.241	0.275
	720	0.453	0.452	0.398	0.412	1.534	0.845	0.328	0.332
	Avg.	0.407	0.415	0.345	0.377	1.649	0.814	0.227	0.261
(c) Gate \rightarrow Sum	96	0.373	0.400	0.286	0.339	1.728	0.845	0.156	0.214
	192	0.414	0.424	0.329	0.369	1.423	0.795	0.204	0.261
	336	0.411	0.422	0.364	0.392	1.693	0.920	0.258	0.302
	720	0.457	0.461	0.427	0.430	1.580	0.890	0.315	0.335
	Avg.	0.414	0.427	0.352	0.383	1.606	0.863	0.233	0.278
(d) BCMASK \rightarrow No mask	96	0.384	0.402	0.288	0.342	1.628	0.840	0.145	0.198
	192	0.413	0.440	0.325	0.363	1.325	0.796	0.191	0.244
	336	0.434	0.448	0.361	0.384	1.606	0.865	0.241	0.285
	720	0.474	0.473	0.421	0.419	1.414	0.811	0.308	0.340
	Avg.	0.426	0.441	0.349	0.377	1.493	0.828	0.221	0.267
(e) BCMASK \rightarrow Random mask	96	0.348	0.384	0.279	0.329	1.618	0.859	0.146	0.197
	192	0.388	0.405	0.318	0.360	1.318	0.798	0.189	0.240
	336	0.383	0.404	0.350	0.381	1.560	0.858	0.243	0.282
	720	0.458	0.462	0.414	0.418	1.392	0.800	0.312	0.328
	Avg.	0.394	0.414	0.340	0.372	1.472	0.829	0.223	0.262
(f) Freeze $f_{\text{vis}}(\cdot)$	96	0.389	0.402	0.293	0.342	1.482	0.761	0.161	0.224
	192	0.434	0.425	0.335	0.367	1.218	0.694	0.203	0.287
	336	0.431	0.428	0.372	0.389	1.58	0.82	0.285	0.302
	720	0.468	0.457	0.431	0.422	1.489	0.815	0.335	0.338
	Avg.	0.431	0.428	0.358	0.380	1.442	0.773	0.246	0.288
(g) W/o decomposition	96	0.352	0.387	0.274	0.329	1.728	0.938	0.143	0.195
	192	0.402	0.414	0.315	0.358	1.841	0.940	0.187	0.242
	336	0.391	0.410	0.347	0.382	1.672	0.886	0.237	0.284
	720	0.487	0.486	0.417	0.422	1.606	0.846	0.309	0.350
	Avg.	0.408	0.424	0.338	0.373	1.712	0.903	0.219	0.268

proposed adaptive decomposition method. In summary, the results demonstrate that DMMV-A has a strong modeling ability of temporal structures and robustness to fluctuations even when dealing with anomalous or weakly periodic time series, validating its reliability and applicability across a broad range of scenarios.

D.5 Additional Visualizations on Masking Strategies

Fig. 13 and Fig. 14 present additional examples of BCMASK in DMMV-A. Similar to §4.3, Fig. 13 and Fig. 14 compare different masking methods. From both figures, we observe that BCMASK produces smooth patterns along the temporal (x -axis) dimension, effectively capturing periodic structures. Notably, when the input time series contains an anomaly (e.g., Fig. 13, time steps 288-336), BCMASK can effectively extract the periodic patterns.

D.6 Impact of Look-Back Window

Fig. 15 provides the MSE results that compare DMMV-A with the other three models. Fig. 15 demonstrate a similar trend as that of the MAE results in Fig. 9.

Table 8: Full results of the ablation analysis of DMMV-S. Lower MSE and MAE are better. The Illness dataset uses prediction lengths of $\{24, 36, 48, 60\}$ due to its short time series (in total 966 time steps), which is different from the prediction lengths of other datasets.

Dataset(\rightarrow) Method(\downarrow), Metric(\rightarrow)	Length	ETTh1		ETTm1		Illness		Weather	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
DMMV-S	96	0.350	0.388	0.296	0.349	1.638	0.838	0.168	0.218
	192	0.399	0.420	0.328	0.370	1.323	0.753	0.220	0.259
	336	0.399	0.415	0.369	0.393	1.644	0.851	0.267	0.304
	720	0.472	0.479	0.401	0.414	1.473	0.810	0.322	0.343
	Avg.	0.405	0.426	0.349	0.382	1.520	0.813	0.244	0.281
(a) $f_{\text{num}}(\cdot) \rightarrow \text{Transformer}$	96	0.352	0.387	0.286	0.339	1.613	0.829	0.148	0.194
	192	0.401	0.420	0.325	0.364	1.417	0.825	0.193	0.240
	336	0.395	0.415	0.354	0.387	1.610	0.853	0.246	0.280
	720	0.460	0.471	0.401	0.414	1.536	0.858	0.330	0.341
	Avg.	0.402	0.423	0.342	0.376	1.544	0.841	0.229	0.264
(b) $f_{\text{vis}}(\cdot) \rightarrow \text{SimMiM}$	96	0.366	0.391	0.323	0.360	1.923	0.901	0.153	0.210
	192	0.412	0.420	0.331	0.364	1.812	0.863	0.194	0.248
	336	0.419	0.420	0.361	0.386	1.793	0.854	0.245	0.279
	720	0.464	0.461	0.404	0.416	1.712	0.883	0.339	0.352
	Avg.	0.415	0.423	0.355	0.382	1.810	0.875	0.233	0.272
(c) Gate \rightarrow Sum	96	0.356	0.389	0.300	0.346	1.503	0.763	0.183	0.234
	192	0.403	0.417	0.334	0.365	1.350	0.746	0.236	0.277
	336	0.414	0.426	0.362	0.385	1.530	0.820	0.271	0.308
	720	0.504	0.506	0.424	0.420	1.429	0.830	0.333	0.369
	Avg.	0.419	0.435	0.355	0.379	1.453	0.790	0.256	0.297
(d) Freeze $f_{\text{vis}}(\cdot)$	96	0.386	0.404	0.306	0.352	1.966	0.921	0.156	0.225
	192	0.436	0.434	0.347	0.375	2.050	0.945	0.240	0.261
	336	0.436	0.440	0.377	0.392	2.223	0.999	0.271	0.312
	720	0.484	0.488	0.443	0.424	2.259	1.009	0.335	0.353
	Avg.	0.436	0.442	0.368	0.386	2.125	0.969	0.251	0.288

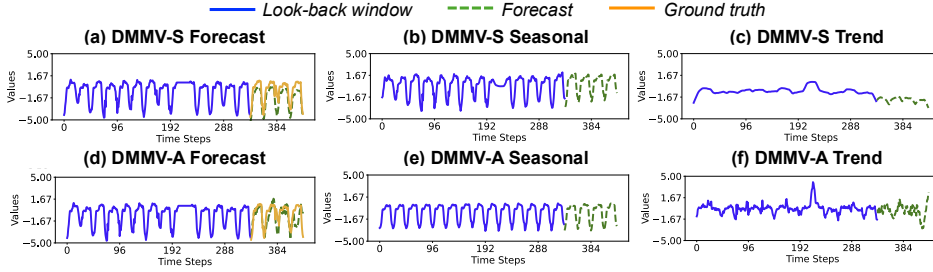


Figure 11: The decompositions of DMMV-S and DMMV-A on the same example in ETTh1: (a)(d) input time series and forecasts, (b)(e) seasonal component, and (c)(f) trend component.

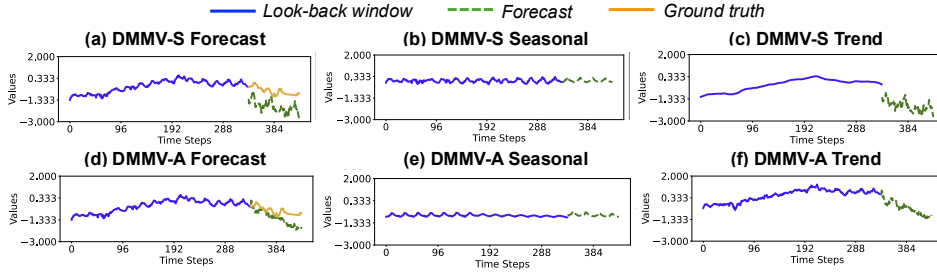


Figure 12: The decompositions of DMMV-S and DMMV-A on the same example in ETTh2: (a)(d) input time series and forecasts, (b)(e) seasonal component, and (c)(f) trend component.

D.7 Standard Deviations

To assess the uncertainty and stability of the forecasting performance, we report the standard deviations of DMMV-S and DMMV-A on the four benchmark datasets used in §4.2 and §4.3 in Table 9. From Table 9, the relative standard deviations of the proposed models, which are calculated as

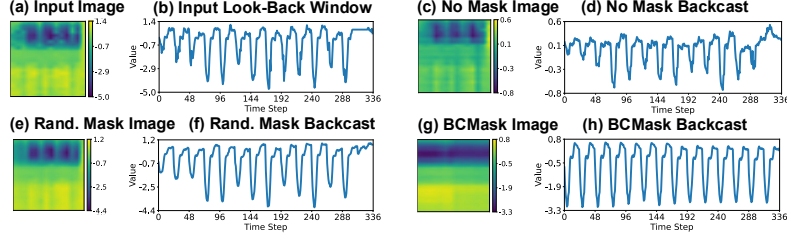


Figure 13: Comparison of different masking methods on the same example in ETTh1. (a) image of input look-back window; (c)(e)(g) are images of backcast output by DMMV-A: (c) uses “No mask”; (e) uses “Random mask”; (g) uses BCMASK. (b)(d)(f)(h) are their recovered time series, respectively.

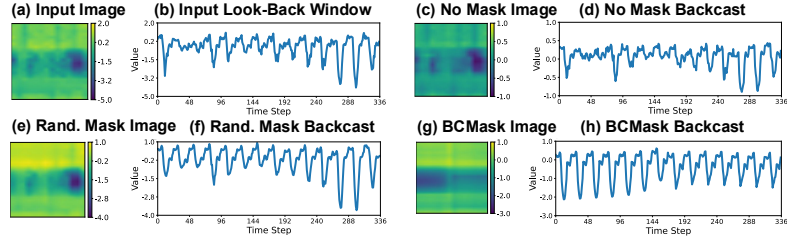


Figure 14: Comparison of different masking methods on the same example in ETTh2. (a) image of input look-back window; (c)(e)(g) are images of backcast output by DMMV-A: (c) uses “No mask”; (e) uses “Random mask”; (g) uses BCMASK. (b)(d)(f)(h) are their recovered time series, respectively.

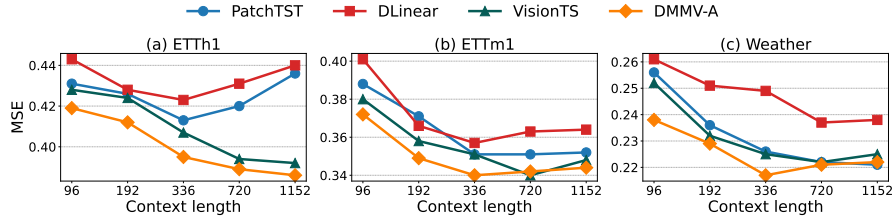


Figure 15: Average MSE comparison with varying look-back window (or context) lengths.

Table 9: Standard Deviations of DMMV-S and DMMV-A in terms of MSE and MAE on four LTSF benchmark datasets.

Model		DMMV-A		DMMV-S	
Metric		MSE	MAE	MSE	MAE
ETTh1	96	0.354±0.001	0.390±0.001	0.350±0.001	0.388±0.002
	192	0.393±0.001	0.405±0.001	0.399±0.002	0.420±0.001
	336	0.387±0.001	0.413±0.001	0.401±0.002	0.415±0.001
	720	0.447±0.002	0.451±0.001	0.472±0.001	0.480±0.002
ETTh2	96	0.278±0.001	0.329±0.000	0.296±0.001	0.348±0.002
	192	0.317±0.001	0.358±0.001	0.328±0.001	0.368±0.002
	336	0.351±0.001	0.381±0.000	0.367±0.002	0.393±0.002
	720	0.411±0.000	0.415±0.000	0.401±0.002	0.415±0.003
Illness	24	1.409±0.001	0.754±0.001	1.638±0.003	0.842±0.005
	36	1.291±0.002	0.742±0.003	1.329±0.012	0.751±0.002
	48	1.499±0.002	0.810±0.011	1.643±0.002	0.853±0.005
	60	1.430±0.003	0.774±0.001	1.473±0.002	0.810±0.002
Weather	96	0.143±0.001	0.196±0.002	0.168±0.001	0.218±0.002
	192	0.187±0.001	0.245±0.003	0.221±0.002	0.259±0.002
	336	0.237±0.001	0.272±0.003	0.267±0.002	0.305±0.001
	720	0.300±0.002	0.318±0.003	0.323±0.001	0.341±0.003

the ratio between standard deviation and mean, are all below 1.30% across different datasets and evaluation metrics, demonstrating their stability and robustness over different runs.