A Installation and Data Preparation

A.1 Installation

In our GitHub (github.com/chengtan9907/OpenSTL), we have provided a conda environment setup file for OpenSTL. Users can easily reproduce the environment by executing the following commands:

```
git clone https://github.com/chengtan9907/OpenSTL
cd OpenSTL
conda env create -f environment.yml
conda activate OpenSTL
python setup.py develop  \# or "pip install -e ."
```

By following the instructions above, OpenSTL will be installed in development mode, allowing any local code modifications to take effect. Alternatively, users can install it as a PyPi package using `pip install .`, but remember to reinstall it to apply any local modifications.

A.2 Data Preparation

It is recommended to symlink the dataset root (assuming `$USER_DATA_ROOT`) to `$OpenSTL/data`. If the folder structure of the user is different, the user needs to change the corresponding paths in config files. We provide tools to download and preprocess the datasets in `OpenSTL/tool/prepare_data`.

B Codebase Overview

In this section, we present a comprehensive overview of the codebase structure of OpenSTL. The codebase is organized into three abstracted layers, namely the core layer, algorithm layer, and user interface layer, arranged from the bottom to the top, as illustrated in Figure 4.

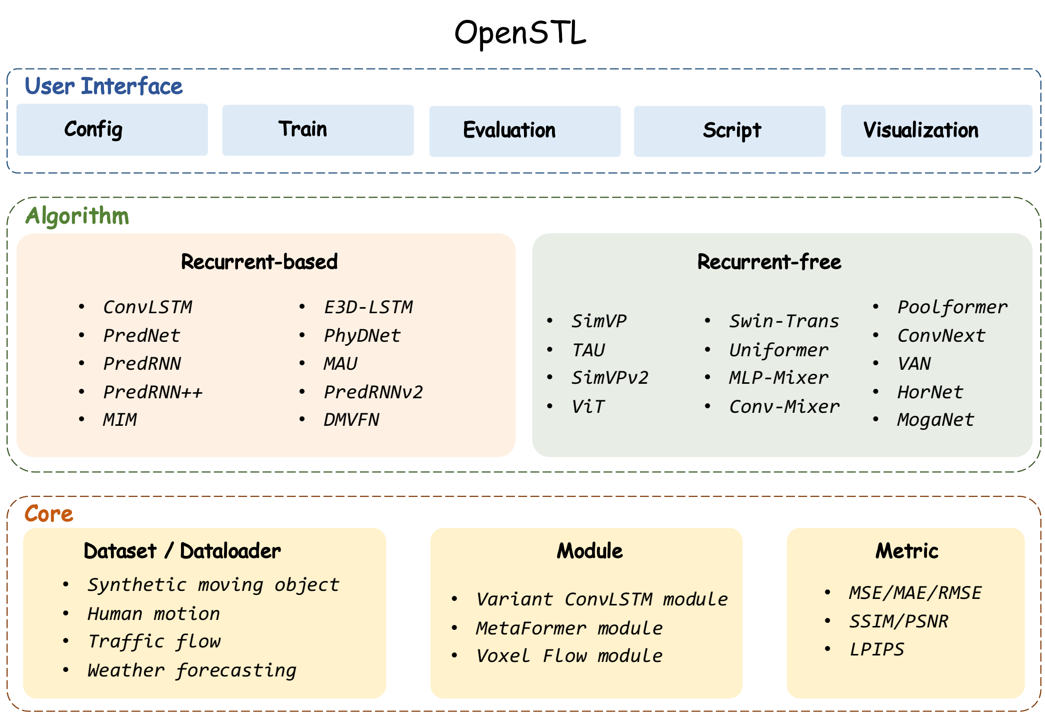

Figure 4: The graphical overview of OpenSTL.

Core Layer The core layer comprises essential components of OpenSTL, such as dataloaders for supported datasets, basic modules for supported models, and metrics for evaluation. The dataloaders offer a unified interface for data loading and preprocessing. The modules consist of foundational unit implementations of supported models. The metrics provide a unified interface for evaluation purposes. The core layer establishes a foundation for the upper layers to ensure flexibility in usage.

Algorithm Layer The algorithm layer encompasses the implementations of the supported models, which are organized into two distinct categories: recurrent-based and recurrent-free models. These implementations are developed using the PyTorch framework and closely adhere to the methodologies described in the original research papers and their official open-sourced code. The algorithm layer ensures the compatibility, reliability, and reproducibility of the supported algorithms by abstracting common components and avoiding code duplication, enabling the easy and flexible implementation of customized algorithms. Moreover, the algorithm layer provides a unified interface that facilitates seamless operations such as model training, evaluation, and testing. By offering a consistent interface, the algorithm layer enhances usability and promotes ease of experimentation with the models.

User Interface Layer The user interface layer comprises configurations, training, evaluation, and scripts that facilitate the basic usage of OpenSTL. We offer convenient tools for generating visualizations. The user interface layer is designed to be user-friendly and intuitive, enabling users to easily train, evaluate, and test the supported algorithms. By offering detailed parameter settings in the configurations, the user interface layer provides a unified interface that enables users to reproduce the results presented in this paper, without requiring any additional efforts.

C Implementation Details

Table 5 describes the hyper-parameters employed in the supported models across multiple datasets, namely MMNIST, KITTI, KTH, Human, TaxiBJ, Weather-S, and Weather-M. For each dataset, the hyperparameters include T, T', hid_S, hid_T, N_S, N_T, epoch, optimizer, drop path, and learning rate. T and T' have the same values for the MMNIST, Human, TaxiBJ, and Weather-M datasets, but differ for KITTI and KTH. The specific values of T and T' are depended on the dataset. The learning rate and drop path are chosen from a set of values, and the best result for each experiment is reported.

The parameters hid_S and hid_T correspond to the size of the hidden layers in the spatial encoder/decoder and the temporal module of the model, respectively. While these parameters exhibit minor variations across datasets, their values largely maintain consistency, underscoring the standardized model structure across diverse datasets. N_S and N_T denote the number of blocks in the spatial encoder/decoder and the temporal module, respectively. These four hyper-parameters are from recurrent-free models, we provide the detailed hyper-parameters of recurrent-based models in GitHub for theirs are various. Please refer to the link OpenSTL/configs for more details.

Table 5: Hyper-parameters of the supported models.

Dataset	MMNIST	KITTI	KTH	Human	TaxiBJ	Weather-S	Weather-M
T	10	10	10	4	4	12	4
T'	10	1	20	4	4	12	4
hid_S	64	64	64	64	32	32	32
hid_T	512	256	256	512	256	256	256
N_S	4	2	2	4	2	2	2
N_T	8	6	6	6	8	8	8
epoch	200	100	100	50	50	50	50
optimizer	Adam						
drop path	$\{0.0, 0.1, 0.2\}$						
learning rate	$\{1e^{-2}, 5e^{-3}, 1e^{-3}, 5e^{-4}, 1e^{-4}\}$						

D Detailed Experimental Results

D.1 Synthetic Moving Object Trajectory Prediction

Moving MNIST In addition to the quantitative results provided in the main text, we also provide a visualization example for qualitative assessment, as shown in Figure 5. For the convenience of formatting, we arrange the frames vertically from bottom to top. It can be observed that the majority of recurrent-based models produce high-quality predicted results, except for PredNet and DMVFN. Recurrent-free models achieve comparable results but exhibit blurriness in the last few frames. This phenomenon suggests that recurrent-based models excel at capturing temporal dependencies.

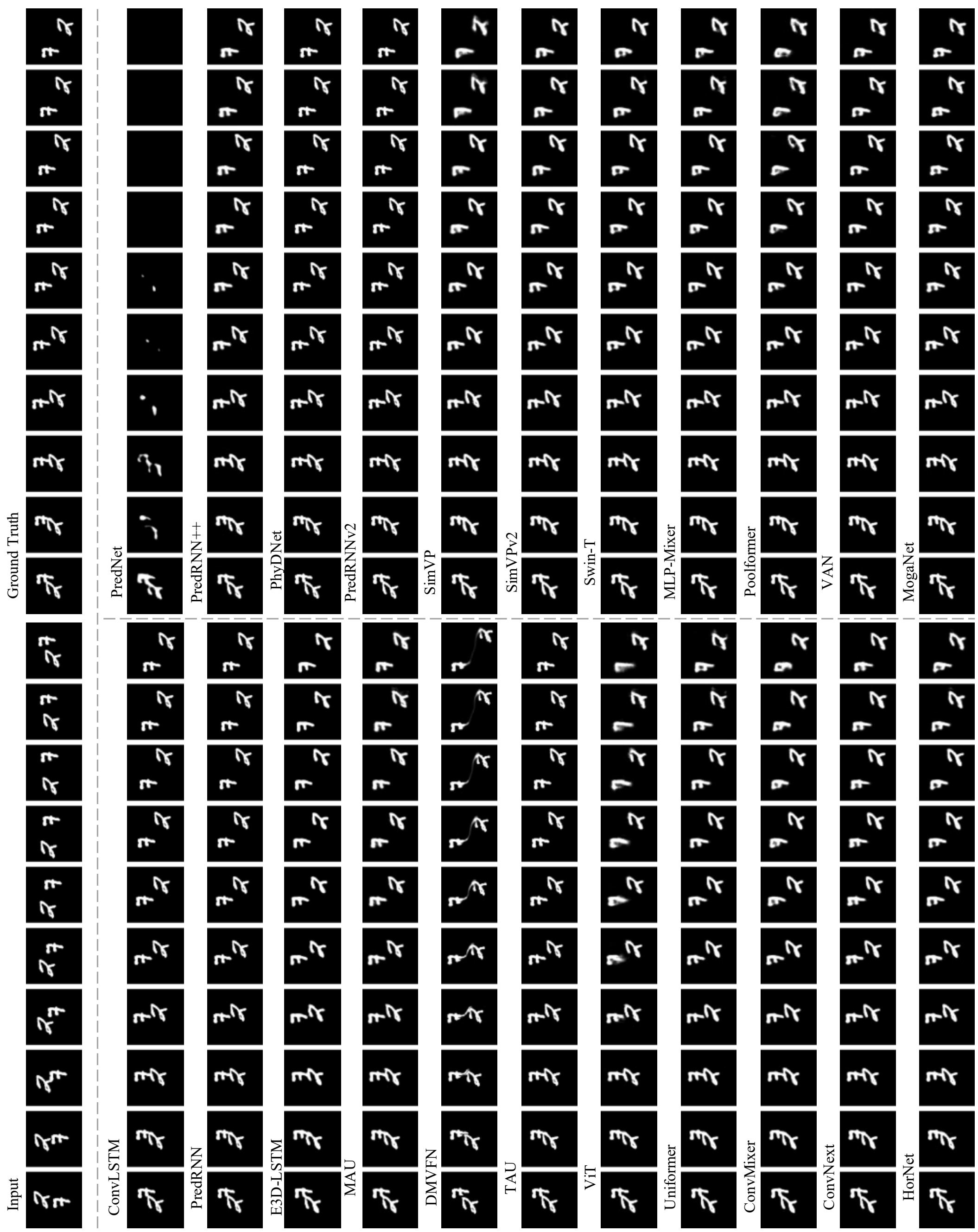

Figure 5: The qualitative visualization on Moving MNIST. For the convenience of formatting, we arrange the frames vertically from bottom to top.

Moving FashionMNIST We show the quantitative results and qualitative visualization examples in Table 6 and Figure 6, respectively. The results are consistent with those of Moving MNIST, where recurrent-based models perform well in long-range temporal modeling.

Table 6: The performance on the Moving FashionMNIST dataset.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	SSIM ↑	PSNR ↑
Recurrent-based	ConvLSTM	15.0	56.8	113	28.87	113.20	0.8793	22.07
	PredNet	12.5	8.4	659	185.94	318.30	0.6713	14.83
	PredRNN	23.8	116.0	54	22.01	**91.74**	0.9091	23.42
	PredRNN++	38.6	171.7	38	**21.71**	91.97	**0.9097**	**23.45**
	MIM	38.0	179.2	37	23.09	96.37	0.9043	23.13
	E3D-LSTM	51.0	298.9	18	35.35	110.09	0.8722	21.27
	PhyDNet	3.1	15.3	182	34.75	125.66	0.8567	22.03
	MAU	4.5	17.8	201	26.56	104.39	0.8916	22.51
	PredRNNv2	23.9	116.6	52	24.13	97.46	0.9004	22.96
	DMVFN	3.5	0.2	1145	118.32	220.02	0.7572	16.76
Recurrent-free	SimVP	58.0	19.4	209	30.77	113.94	0.8740	21.81
	TAU	44.7	16.0	283	24.24	96.72	0.8995	22.87
	SimVPv2	46.8	16.5	282	25.86	101.22	0.8933	22.61
	ViT	46.1	16.9	290	31.05	115.59	0.8712	21.83
	Swin Transformer	46.1	16.4	294	28.66	108.93	0.8815	22.08
	Uniformer	44.8	16.5	296	29.56	111.72	0.8779	21.97
	MLP-Mixer	38.2	14.7	334	28.83	109.51	0.8803	22.01
	ConvMixer	3.9	5.5	658	31.21	115.74	0.8709	21.71
	Poolformer	37.1	14.1	341	30.02	113.07	0.8750	21.95
	ConvNext	37.3	14.1	344	26.41	102.56	0.8908	22.49
	VAN	44.5	16.0	288	31.39	116.28	0.8703	22.82
	HorNet	45.7	16.3	287	29.19	110.17	0.8796	22.03
	MogaNet	46.8	16.5	255	25.14	99.69	0.8960	22.73

Moving MNIST-CIFAR The quantitative results are presented in Table 7, while the qualitative visualizations are depicted in Figure 7. As the task involves more complex backgrounds, the models are required to pay greater attention to spatial modeling. Consequently, the gap between recurrent-based and recurrent-free models is narrowed.

Table 7: The performance on the Moving MNIST-CIFAR dataset.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	SSIM ↑	PSNR ↑
Recurrent-based	ConvLSTM	15.0	56.8	113	73.31	338.56	0.9204	23.09
	PredNet	12.5	8.4	659	286.70	514.14	0.8139	17.49
	PredRNN	23.8	116.0	54	50.09	225.04	0.9499	24.90
	PredRNN++	38.6	171.7	38	**44.19**	198.27	**0.9567**	**25.60**
	MIM	38.0	179.2	37	48.63	213.44	0.9521	25.08
	E3D-LSTM	51.0	298.9	18	80.79	214.86	0.9314	22.89
	PhyDNet	3.1	15.3	182	142.54	700.37	0.8276	19.92
	MAU	4.5	17.8	201	58.84	255.76	0.9408	24.19
	PredRNNv2	23.9	116.6	52	57.27	252.29	0.9419	24.24
	DMVFN	3.5	0.2	1145	298.73	606.92	0.7765	17.07
Recurrent-free	SimVP	58.0	19.4	209	59.83	214.54	0.9414	24.15
	TAU	44.7	16.0	283	48.17	**177.35**	0.9539	25.21
	SimVPv2	46.8	16.5	282	51.13	185.13	0.9512	24.93
	ViT	46.1	16.9	290	64.94	234.01	0.9354	23.90
	Swin Transformer	46.1	16.4	294	57.11	207.45	0.9443	24.34
	Uniformer	44.8	16.5	296	56.96	207.51	0.9442	24.38
	MLP-Mixer	38.2	14.7	334	57.03	206.46	0.9446	24.34
	ConvMixer	3.9	5.5	658	59.29	219.76	0.9403	24.17
	Poolformer	37.1	14.1	341	60.98	219.50	0.9399	24.16
	ConvNext	37.3	14.1	344	51.39	187.17	0.9503	24.89
	VAN	44.5	16.0	288	59.59	221.32	0.9398	25.20
	HorNet	45.7	16.3	287	55.79	202.73	0.9456	24.49
	MogaNet	46.8	16.5	255	49.48	184.11	0.9521	25.07

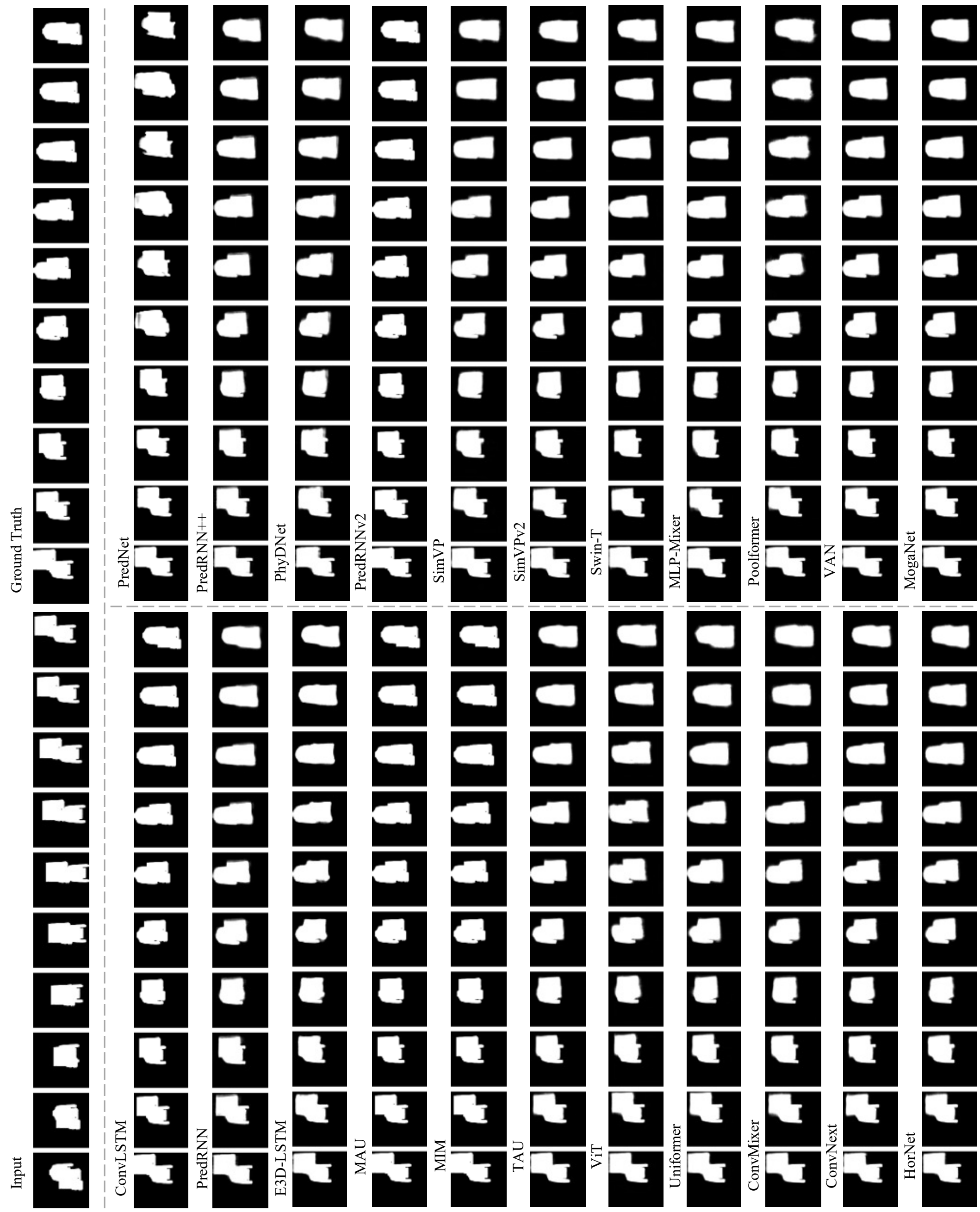

Figure 6: The qualitative visualization on Moving Fashion MNIST. For the convenience of formatting, we arrange the frames vertically from bottom to top.

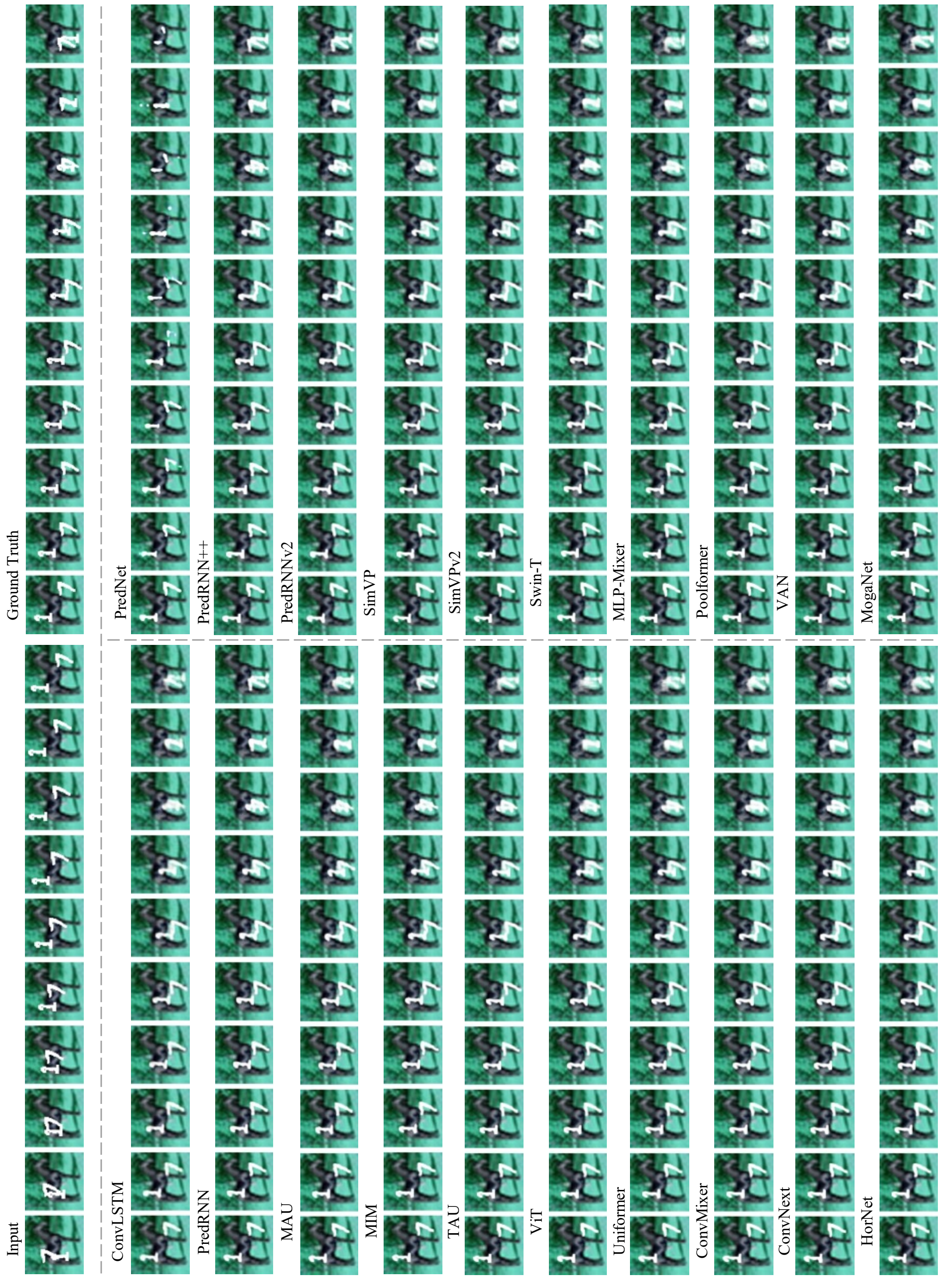

Figure 7: The qualitative visualization on Moving MNIST-CIFAR. For the convenience of formatting, we arrange the frames vertically from bottom to top.

D.2 Real-world Video Prediction

Kitties&Caltech In addition to the quantitative results presented in the main text, we also provide a visualization example for qualitative assessment, as depicted in Figure 8. Interestingly, even though PredNet and DMVFN, which have limited temporal modeling capabilities, can still perform reasonably well in predicting the next frame.

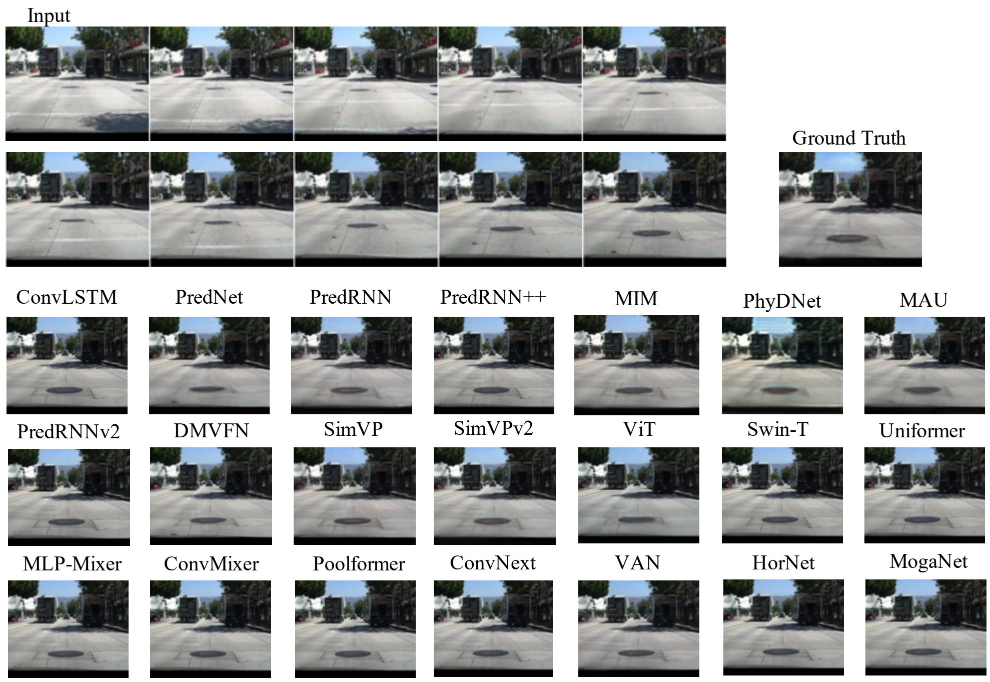

Figure 8: The qualitative visualization on Kitti&Caltech.

KTH We showcase the quantitative results and qualitative visualization on KTH in Table 8 and Figure 9, respectively. Recurrent-free models demonstrate comparable performance while requiring few computational costs, thus striking a favorable balance between performance and efficiency.

Table 8: The performance on the KTH dataset.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	SSIM ↑	PSNR ↑	LPIPS ↓
Recurrent-based	ConvLSTM	14.9	1368.0	16	47.65	445.50	0.8977	26.99	0.26686
	PredNet	12.5	3.4	399	152.11	783.10	0.8094	22.45	0.32159
	PredRNN	23.6	2800.0	7	41.07	380.60	0.9097	27.95	0.21892
	PredRNN++	38.3	4162.0	5	**39.84**	**370.40**	**0.9124**	**28.13**	0.19871
	MIM	39.8	1099.0	17	40.73	380.80	0.9025	27.78	0.18808
	E3D-LSTM	53.5	217.0	17	136.40	892.70	0.8153	21.78	0.48358
	PhyDNet	3.1	93.6	58	91.12	765.60	0.8322	23.41	0.50155
	MAU	20.1	399.0	8	51.02	471.20	0.8945	26.73	0.25442
	PredRNNv2	23.6	2815.0	7	45.84	420.8	0.9039	27.33	0.23236
	DMVFN	3.5	0.9	727	59.61	413.20	0.8976	26.65	**0.12842**
Recurrent-free	SimVP	12.2	62.8	77	41.11	397.10	0.9065	27.46	0.26496
	TAU	15.0	73.8	55	45.32	421.70	0.9086	27.10	0.22856
	SimVPv2	15.6	76.8	53	45.02	417.80	0.9049	27.04	0.25240
	ViT	12.7	112.0	28	56.57	459.30	0.8947	26.19	0.27494
	Swin-T	15.3	75.9	65	45.72	405.70	0.9039	27.01	0.25178
	Uniformer	11.8	78.3	43	44.71	404.60	0.9058	27.16	0.24174
	MLP-Mixer	20.3	66.6	34	59.55	510.4	0.8863	25.59	0.2815
	ConvMixer	1.5	18.3	175	47.31	446.10	0.8993	26.66	0.28149
	Poolformer	12.4	63.6	67	45.44	400.90	0.9065	27.22	0.24763
	ConvNext	12.5	63.9	72	45.48	428.30	0.9037	26.96	0.26253
	VAN	14.9	73.8	55	45.05	409.10	0.9074	27.07	0.23116
	HorNet	15.3	75.3	58	46.84	421.20	0.9005	26.80	0.26921
	MogaNet	15.6	76.7	48	42.98	418.70	0.9065	27.16	0.25146

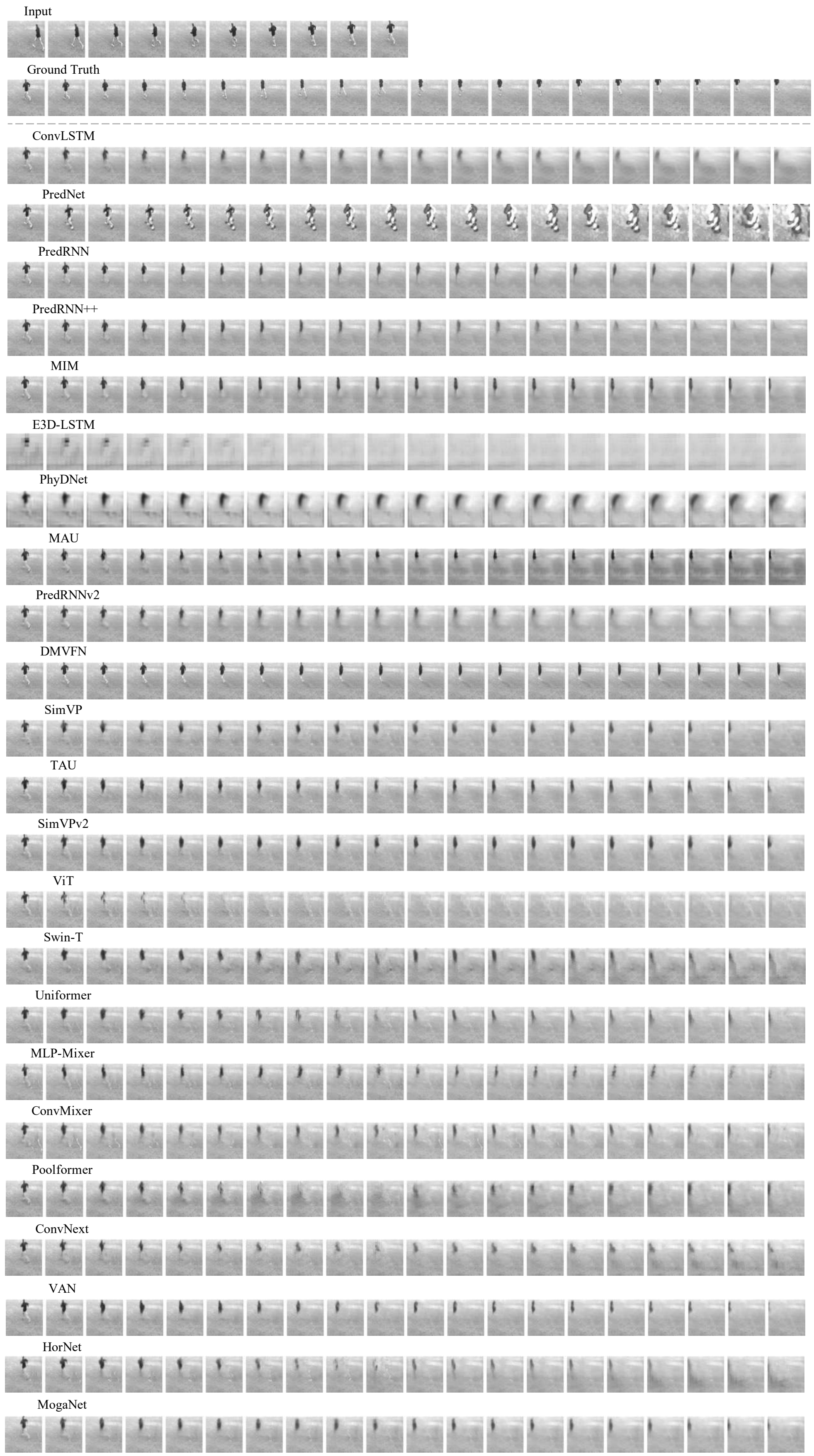

Figure 9: The qualitative visualization on KTH.

Human3.6M The quantitative results are presented in Table 9, and the qualitative visualization is depicted in Figure 10. In this task, human motion exhibits subtle changes between adjacent frames, resulting in a low-frequency signal of overall dynamics. Consequently, recurrent-free models, which excel at spatial learning, can efficiently and accurately predict future frames.

Table 9: The performance on the Human3.6M dataset.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	SSIM ↑	PSNR ↑	LPIPS ↓
Recurrent-based	ConvLSTM	15.5	347.0	8	125.5	1566.7	0.9813	33.40	0.03557
	PredRNN	24.6	704.0	4	113.2	1458.3	0.9831	33.94	0.03245
	PredRNN++	39.3	1033.0	2	110.0	1452.2	0.9832	34.02	0.03196
	MIM	47.6	1051.0	9	112.1	1467.1	0.9829	33.97	0.03338
	E3D-LSTM	60.9	542.0	9	143.3	1442.5	0.9803	32.52	0.04133
	PredRNNv2	24.6	708.0	3	114.9	1484.7	0.9827	33.84	0.03334
	DMVFN	8.6	63.6	321	109.3	1449.3	0.9833	34.05	0.03189
Recurrent-free	SimVP	41.2	197.0	38	115.8	1511.5	0.9822	33.73	0.03467
	TAU	37.6	182.0	30	113.3	**1390.7**	**0.9839**	34.03	**0.02783**
	SimVPv2	11.3	74.6	26	108.4	1441.0	0.9834	**34.08**	0.03224
	Uniformer	27.7	211.0	23	116.3	1497.7	0.9824	33.76	0.03385
	MLP-Mixer	47.0	164.0	18	125.7	1511.9	0.9819	33.49	0.03417
	ConvMixer	3.1	39.4	87	115.8	1527.4	0.9822	33.67	0.03436
	ConvNext	31.4	157.0	36	113.4	1469.7	0.9828	33.86	0.03305
	HorNet	28.1	143.0	29	118.1	1481.1	0.9825	33.73	0.03333
	MogaNet	8.6	163.6	23	109.1	1446.4	0.9834	33.89	0.03243

D.3 Traffic and Weather Forecasting

D.3.1 TaxiBJ

We show the quantitative results in Table 10 and qualitative visualizations in Figure 11. The recurrent-free models have shown promising results in low-frequency traffic flow data than their counterparts.

Table 10: The performance on the TaxiBJ dataset.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	SSIM ↑
Recurrent-based	ConvLSTM	15.0	20.7	815	0.3358	15.32	0.9836
	PredNet	12.5	0.9	5031	0.3516	15.91	0.9828
	PredRNN	23.7	42.4	416	0.3194	15.31	0.9838
	PredRNN++	38.4	63.0	301	0.3348	15.37	0.9834
	MIM	37.9	64.1	275	0.3110	14.96	0.9847
	E3DLSTM	51.0	98.19	60	0.3421	14.98	0.9842
	PhyDNet	3.1	5.6	982	0.3622	15.53	0.9828
	MAU	4.4	6.0	540	0.3268	15.26	0.9834
	PredRNNv2	23.7	42.6	378	0.3834	15.55	0.9826
	DMVFN	3.5	57.1	4772	0.3517	15.72	0.9833
Recurrent-free	SimVP	13.8	3.6	533	0.3282	15.45	0.9835
	TAU	9.6	2.5	1268	0.3108	14.93	**0.9848**
	SimVPv2	10.0	2.6	1217	0.3246	15.03	0.9844
	ViT	9.7	2.8	1301	0.3171	15.15	0.9841
	Swin Transformer	9.7	2.6	1506	0.3128	15.07	0.9847
	Uniformer	9.5	2.7	1333	0.3268	15.16	0.9844
	MLP-Mixer	8.2	2.2	1974	0.3206	15.37	0.9841
	ConvMixer	0.8	0.2	4793	0.3634	15.63	0.9831
	Poolformer	7.6	2.1	1827	0.3273	15.39	0.9840
	ConvNext	7.8	2.1	1918	**0.3106**	**14.90**	0.9845
	VAN	9.5	2.5	1273	0.3125	14.96	**0.9848**
	HorNet	9.7	2.5	1350	0.3186	15.01	0.9843
	MogaNet	10.0	2.6	1005	0.3114	15.06	0.9847

Figure 10: The qualitative visualization on Human3.6M.

(a) InFlow of TaxBJ

(b) OutFlow of TaxBJ

Figure 11: The qualitative visualization on TaxiBJ.

D.3.2 WeatherBench

We strongly recommend readers refer to the GIF animations provided in our GitHub (OpenSTL/docs/en/visualization/), as they provide a clearer visualization of the model's prediction performance.

Single-variable Temperature Forecasting The quantitative results and qualitative visualization are presented in Table 11 and Figure 12. The recurrent-free models exhibit a clear superiority over the recurrent-based models in terms of both performance and efficiency, achieving a landslide victory.

Table 11: The performance on the single-variable temperature forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	14.9	136.0	46	1.521	0.7949	1.233
	PredRNN	23.6	278.0	22	1.331	0.7246	1.154
	PredRNN++	38.3	413.0	15	1.634	0.7883	1.278
	MIM	37.8	109.0	126	1.784	0.8716	1.336
	PhyDNet	3.1	36.8	177	285.9	8.7370	16.91
	MAU	5.5	39.6	237	1.251	0.7036	1.119
	PredRNNv2	23.6	279.0	22	1.545	0.7986	1.243
Recurrent-free	SimVP	14.7	8.0	160	1.238	0.7037	1.113
	TAU	12.2	6.7	511	1.162	0.6707	1.078
	SimVPv2	12.8	7.0	504	**1.105**	**0.6567**	**1.051**
	ViT	12.4	8.0	432	1.146	0.6712	1.070
	Swin Transformer	12.4	6.9	581	1.143	0.6735,	1.069
	Uniformer	12.0	7.5	465	1.204	0.6885	1.097
	MLP-Mixer	11.1	5.9	713	1.255	0.7011	1.119
	ConvMixer	1.1	1.0	1705	1.267	0.7073	1.126
	Poolformer	10.0	5.6	722	1.156	0.6715	1.075
	ConvNext	10.1	5.7	689	1.277	0.7220	1.130
	VAN	12.2	6.7	523	1.150	0.6803	1.072
	HorNet	12.4	6.8	517	1.201	0.6906	1.096
	MogaNet	12.8	7.0	416	1.152	0.6665	1.073

Single-variable Humidity Forecasting The quantitative results and qualitative visualization are presented in Table 12 and Figure 13. The results are almost consistent with the temperature forecasting.

Table 12: The performance on the single-variable humidity forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	14.9	136.0	46	35.146	4.012	5.928
	PredRNN	23.6	278.0	22	37.611	4.096	6.133
	PredRNN++	38.3	413.0	15	45.993	4.731	6.782
	MIM	37.8	109.0	126	61.113	5.504	7.817
	PhyDNet	3.1	36.8	177	239.0	8.975	15.46
	MAU	5.5	39.6	237	34.529	4.004	5.876
	PredRNNv2	23.6	279.0	22	36.508	4.087	6.042
Recurrent-free	SimVP	14.7	8.0	160	34.355	3.994	5.861
	TAU	12.2	6.7	511	31.831	3.818	5.642
	SimVPv2	12.8	7.0	504	31.426	**3.765**	5.606
	ViT	12.4	8.0	432	32.616	3.852	5.711
	Swin Transformer	12.4	6.9	581	**31.332**	3.776	**5.597**
	Uniformer	12.0	7.5	465	32.199	3.864	5.674
	MLP-Mixer	11.1	5.9	713	34.467	3.950	5.871
	ConvMixer	1.1	1.0	1705	32.829	3.909	5.730
	Poolformer	10.0	5.6	722	31.989	3.803	5.656
	ConvNext	10.1	5.7	689	33.179	3.928	5.760
	VAN	12.2	6.7	523	31.712	3.812	5.631
	HorNet	12.4	6.8	517	32.081	3.826	5.664
	MogaNet	12.8	7.0	416	31.795	3.816	5.639

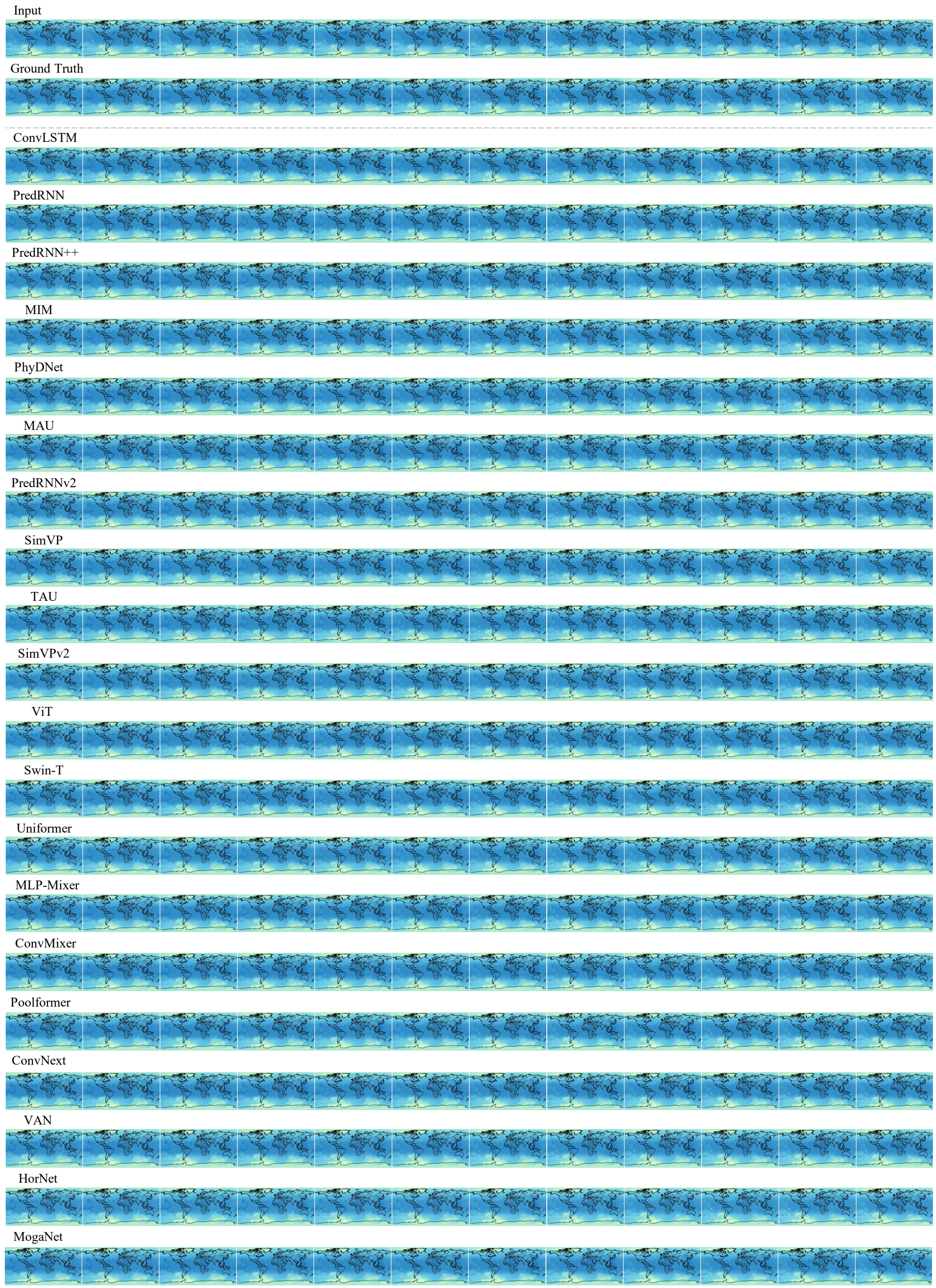

Figure 12: The qualitative visualization on the single-variable temperature forecasting in the WeatherBench dataset.

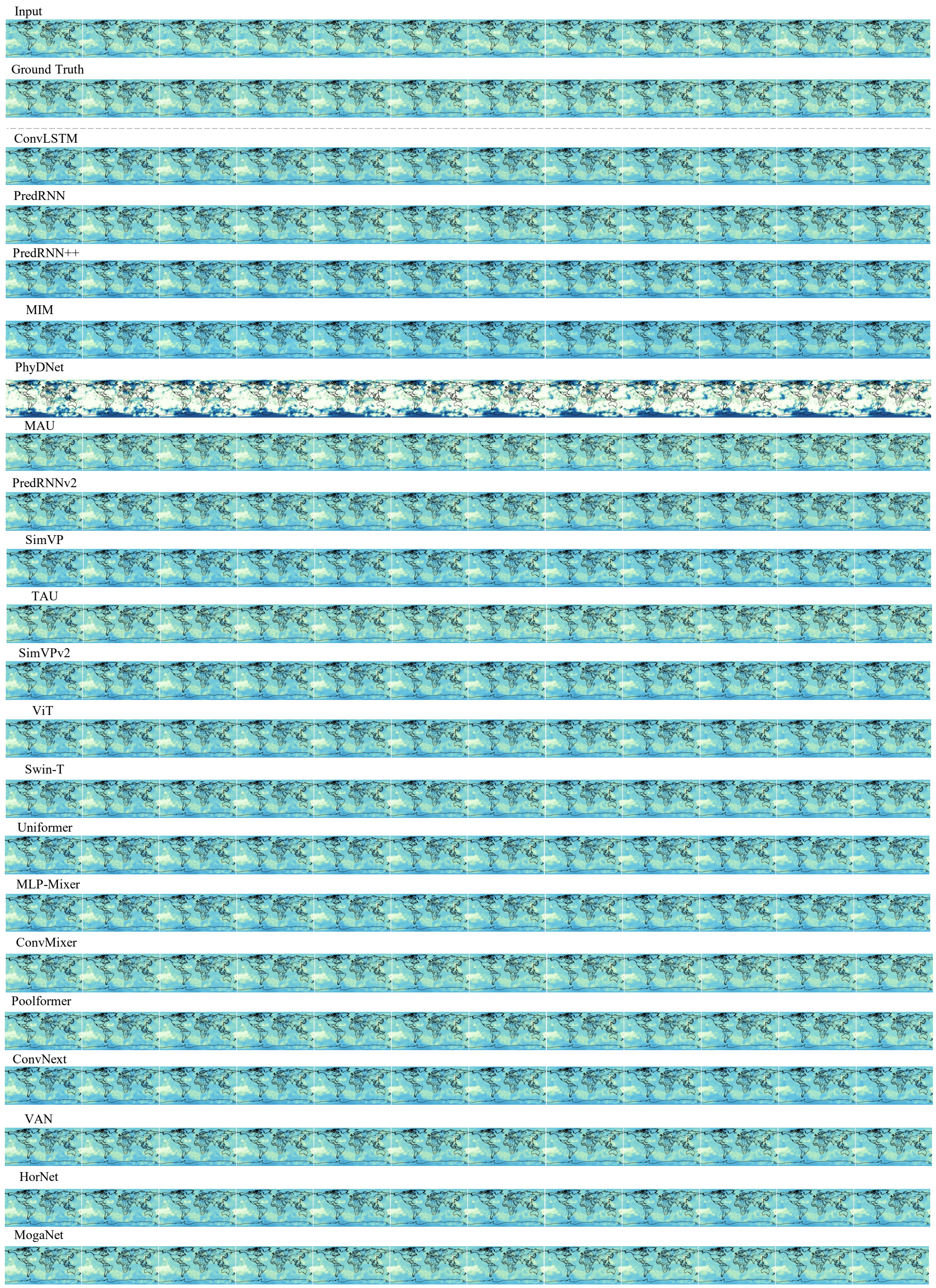

Figure 13: The qualitative visualization on the single-variable humidity forecasting in the WeatherBench dataset.

Single-variable Wind Component Forecasting The quantitative results are presented in Table 13. The qualitative visualizations of latitude and longitude wind are shown in Figure 14 and Figure 15. Most recurrent-free models outperform the recurrent-based models.

Table 13: The performance on the single-variable wind component forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	15.0	136.0	43	1.8976	0.9215	1.3775
	PredRNN	23.7	279.0	21	1.8810	0.9068,	1.3715
	PredRNN++	38.4	414.0	14	1.8727	0.9019	1.3685
	MIM	37.8	109.0	122	3.1399	1.1837	1.7720
	PhyDNet	3.1	36.8	172	16.7983	2.9208	4.0986
	MAU	5.5	39.6	233	1.9001	0.9194	1.3784
	PredRNNv2	23.7	280.0	21	2.0072	0.9413	1.4168
Recurrent-free	SimVP	14.7	8.0	430	1.9993	0.9510	1.4140
	TAU	12.2	6.7	505	1.5925	0.8426	1.2619
	SimVPv2	12.8	7.0	529	1.5069	0.8142	1.2276
	ViT	12.4	8.0	427	1.6262	0.8438	1.2752
	Swin Transformer	12.4	6.9	559	**1.4996**	0.8145	**1.2246**
	Uniformer	12.0	7.5	466	1.4850	**0.8085**	1.2186
	MLP-Mixer	11.1	5.9	687	1.6066	0.8395	1.2675
	ConvMixer	1.1	1.0	1807	1.7067	0.8714	1.3064
	Poolformer	10.0	5.6	746	1.6123	0.8410	1.2698
	ConvNext	10.1	5.7	720	1.6914	0.8698	1.3006
	VAN	12.2	6.7	549	1.5958	0.8371	1.2632
	HorNet	12.4	6.9	539	1.5539	0.8254	1.2466
	MogaNet	12.8	7.0	441	1.6072	0.8451	1.2678

Single-variable Cloud Cover Forecasting The quantitative results and visualization are presented in Table 14 and Figure 16. All the recurrent-free models perform better than their counterparts.

Table 14: The performance on the single-variable cloud cover forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	14.9	136.0	46	0.04944	0.15419	0.222
	PredRNN	23.6	278.0	22	0.05504	0.15877	0.234
	PredRNN++	38.3	413.0	15	0.05479	0.15435	0.234
	MIM	37.75	109.0	126	0.05997	0.17184	0.245
	PhyDNet	3.1	36.8	177	0.09913	0.22614	0.314
	MAU	5.5	39.6	237	0.04955	0.15158	0.222
	PredRNNv2	23.6	279.0	22	0.05051	0.15867	0.224
Recurrent-free	SimVP	14.7	8.0	160	0.04765	0.15029	0.218
	TAU	12.2	6.7	511	0.04723	**0.14604**	0.217
	SimVPv2	12.8	7.0	504	0.04657	0.14688	0.215
	ViT	12.4	8.0	432	0.04778	0.15026	0.218
	Swin Transformer	12.4	6.9	581	**0.04639**	0.14729	**0.215**
	Uniformer	12.0	7.5	465	0.04680	0.14777	0.216
	MLP-Mixer	11.1	5.9	713	0.04925	0.15264	0.221
	ConvMixer	1.1	1.0	1705	0.04717	0.14874	0.217
	Poolformer	10.0	5.6	722	0.04694	0.14884	0.216
	ConvNext	10.1	5.7	689	0.04742	0.14867	0.217
	VAN	12.2	6.7	523	0.04694	0.14725	0.216
	HorNet	12.4	6.8	517	0.04692	0.14751	0.216
	MogaNet	12.8	7.0	416	0.04699	0.14802	0.216

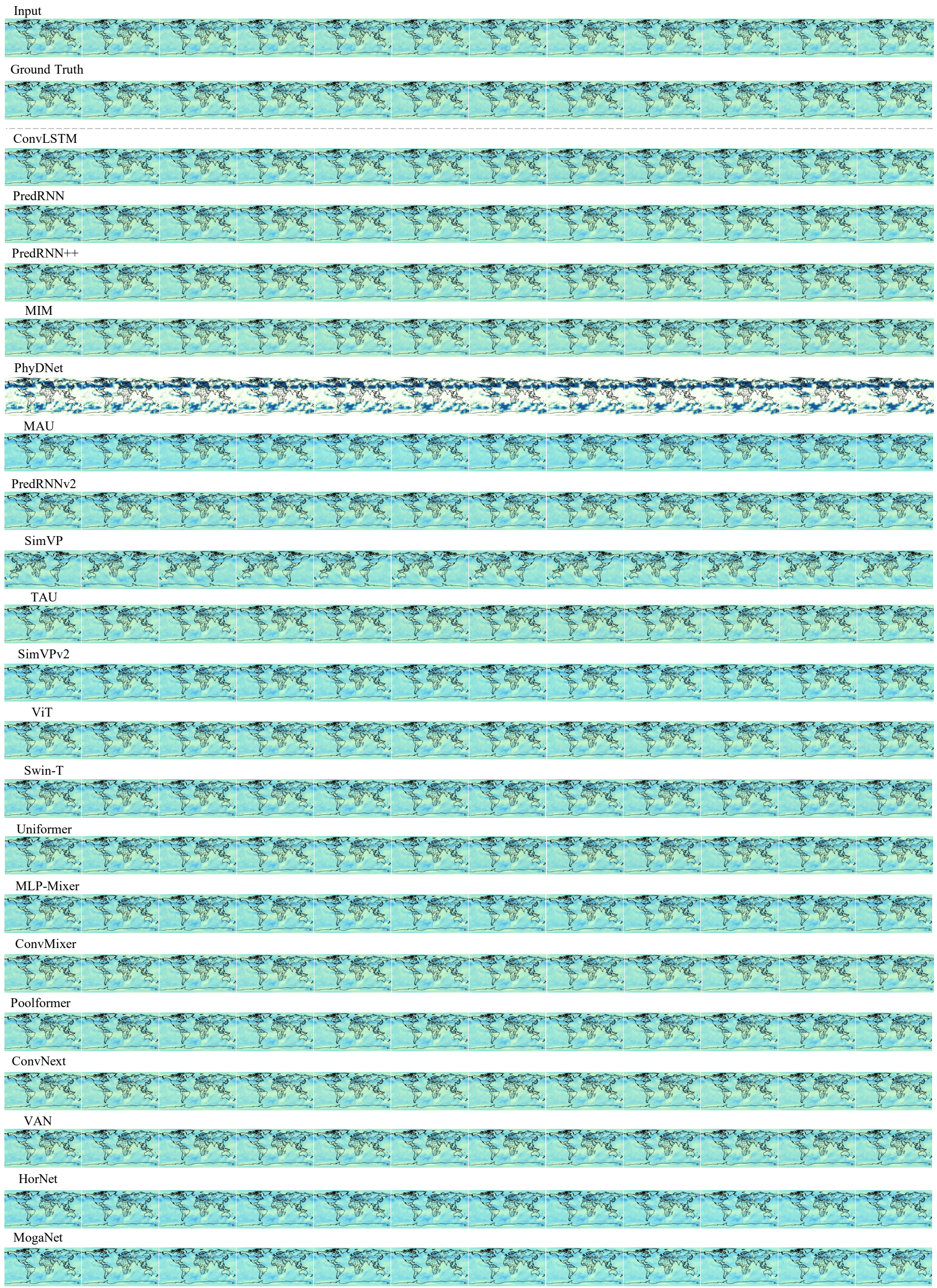

Figure 14: The qualitative visualization on the single-variable latitude wind forecasting in the WeatherBench dataset.

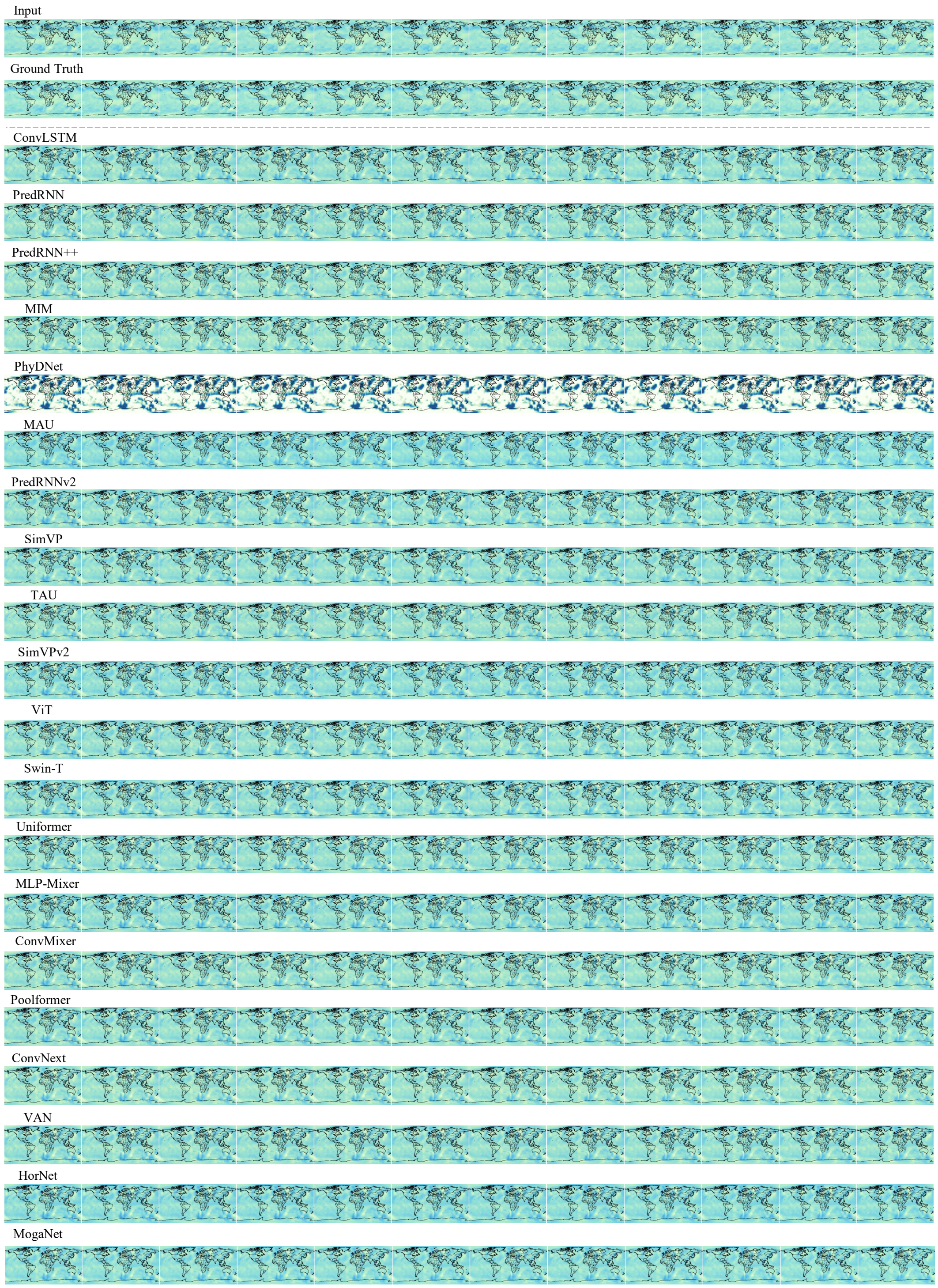

Figure 15: The qualitative visualization on the single-variable longitude wind forecasting in the WeatherBench dataset.

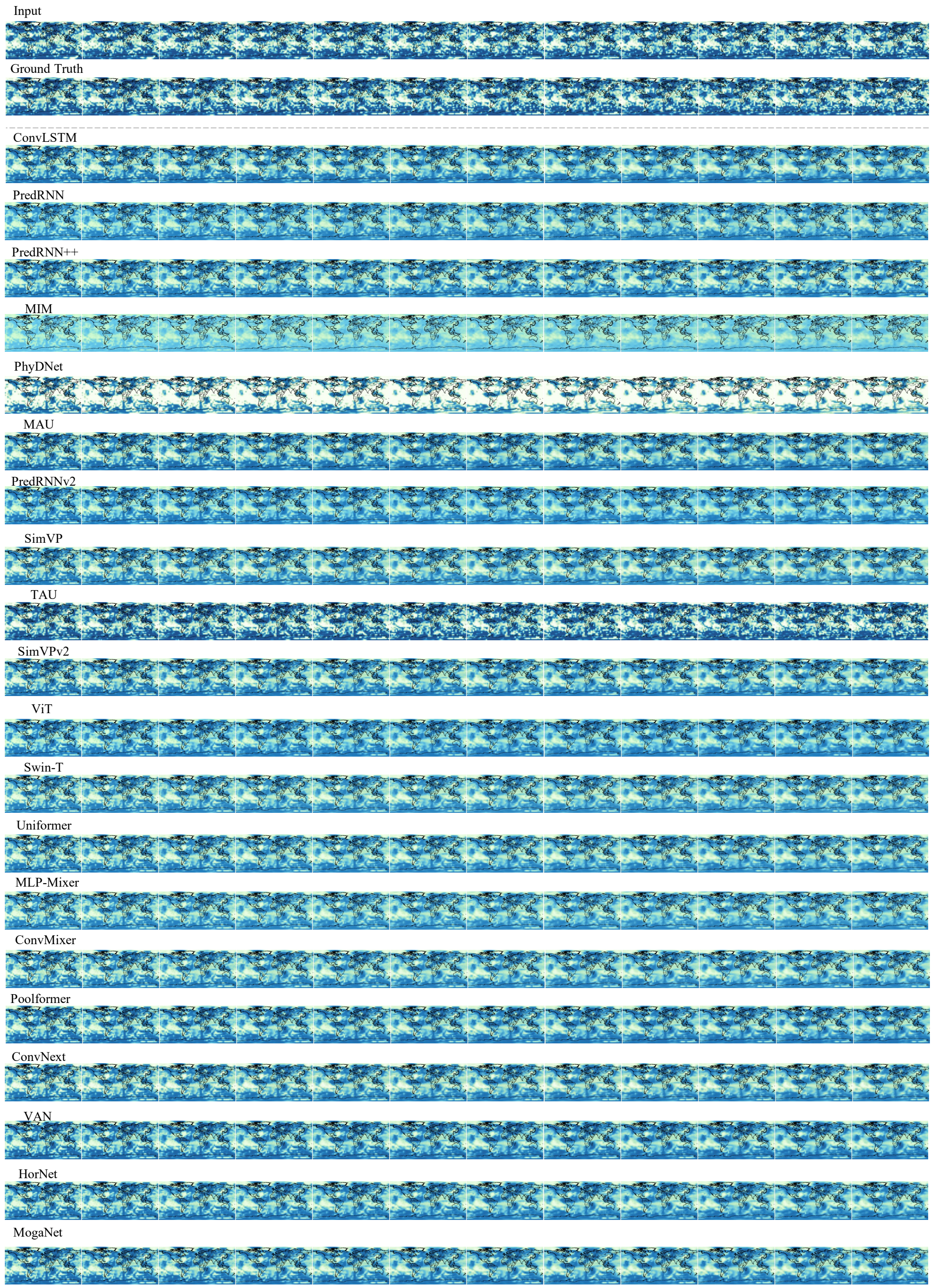

Figure 16: The qualitative visualization on the single-variable cloud cover forecasting in the WeatherBench dataset.

Single-variable Temperature Forecasting with High Resolution We perform experiments on high-resolution (128×256) temperature forecasting. The quantitative results are presented in Table 15. SimVPv2 achieves remarkable performance, surpassing the recurrent-based models by large margins.

Table 15: The performance on the single-variable high-resolution (128×256) temperature forecasting.

Method		Params (M)	FLOPs (G)	FPS	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	15.0	550.0	35	1.0625	0.6517	1.0310
	PredRNN	23.8	1123.0	3	0.8966	0.5869	0.9469
	PredRNN++	38.6	1663.0	2	0.8538	0.5708	0.9240
	MIM	42.2	1739.0	11	1.2138	0.6857	1.1017
	PhyDNet	3.1	148.0	41	297.34	8.9788	17.243
	MAU	11.8	172.0	17	1.0031	0.6316	1.0016
	PredRNNv2	23.9	1129.0	3	1.0451	0.6190	1.0223
Recurrent-free	SimVP	14.7	128.0	27	0.8492	0.5636	0.9215
	TAU	12.3	36.1	94	0.8316	0.5615	0.9119
	SimVPv2	12.8	112.0	33	**0.6499**	**0.4909**	**0.8062**
	ViT	12.5	36.8	50	0.8969	0.5834	0.9470
	Swin Transformer	12.4	110.0	38	0.7606	0.5193	0.8721
	Uniformer	12.1	48.8	57	1.0052	0.6294	1.0026
	MLP-Mixer	27.9	94.7	49	1.1865	0.6593	1.0893
	ConvMixer	1.1	15.1	117	0.8557	0.5669	0.9250
	Poolformer	10.0	89.7	42	0.7983	0.5316	0.8935
	ConvNext	10.1	90.5	47	0.8058	0.5406	0.8976
	VAN	12.2	107.0	34	0.7110	0.5094	0.8432
	HorNet	12.4	109.0	34	0.8250	0.5467	0.9083
	MogaNet	12.8	112.0	27	0.7517	0.5232	0.8670

Multiple-variable Forecasting This task focuses on multi-factor climate prediction. We include temperature, humidity, latitude wind, and longitude factors in the forecasting process. The comprehensive results can be found in Table 16 to Table 19. We also show a comparison in Figure 17. MogaNet achieves significant leading performance across various metrics in predicting climatic factors.

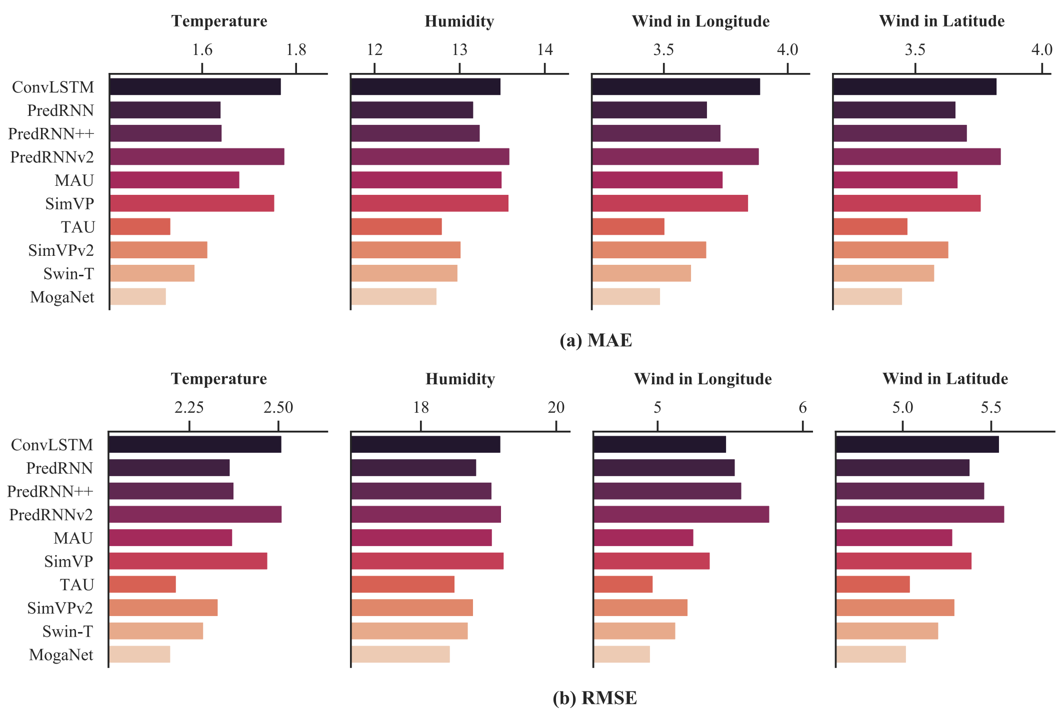

Figure 17: The (a) MAE and (b) RMSE metrics of the representative approaches on the four weather forecasting tasks in WeatherBench (Muti-variable setting).

Table 16: The performance on the multiple-variable temperature forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	15.5	43.3	6.3034	1.7695	2.5107
	PredRNN	24.6	88.0	5.5966	1.6411	2.3657
	PredRNN++	39.3	129.0	5.6471	1.6433	2.3763
	MIM	5.5	12.1	7.5152	1.9650	2.7414
	PhyDNet	3.1	11.3	95.113	6.4749	9.7526
	MAU	5.5	12.1	5.6287	1.6810	2.3725
	PredRNNv2	24.6	88.5	6.3078	1.7770	2.5115
Recurrent-free	SimVP	13.8	7.3	6.1068	1.7554	2.4712
	TAU	9.6	5.0	4.9042	1.5341	2.2145
	SimVPv2	10.0	5.3	5.4382	1.6129	2.3319
	ViT	9.7	6.1	5.2722	1.6005	2.2961
	Swin Transformer	9.7	5.2	5.2486	1.5856	2.2910
	Uniformer	9.5	5.9	5.1174	1.5758	2.2622
	MLP-Mixer	8.7	4.4	5.8546	1.6948	2.4196
	ConvMixer	0.9	0.5	6.5838	1.8228	2.5659
	Poolformer	7.8	4.1	7.1077	1.8791	2.6660
	ConvNext	7.9	4.2	6.1749	1.7448	2.4849
	VAN	9.5	5.0	4.9396	1.5390	2.2225
	HorNet	9.7	5.1	5.5856	1.6198	2.3634
	MogaNet	10.0	5.3	**4.8335**	**1.5246**	**2.1985**

Table 17: The performance on the multiple-variable humidity forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	15.5	43.3	368.15	13.490	19.187
	PredRNN	24.6	88.0	354.57	13.169	18.830
	PredRNN++	39.3	129.0	363.15	13.246	19.056
	MIM	41.7	35.8	408.24	14.658	20.205
	PhyDNet	3.1	11.3	668.40	21.398	25.853
	MAU	5.5	12.1	363.36	13.503	19.062
	PredRNNv2	24.6	88.5	368.52	13.594	19.197
Recurrent-free	SimVP	13.8	7.3	370.03	13.584	19.236
	TAU	9.6	5.0	342.63	12.801	**18.510**
	SimVPv2	10.0	5.3	352.79	13.021	18.783
	ViT	9.7	6.1	352.36	13.056	18.771
	Swin Transformer	9.7	5.2	349.92	**12.984**	18.706
	Uniformer	9.5	5.9	351.66	12.994	18.753
	MLP-Mixer	8.7	4.4	365.48	13.408	19.118
	ConvMixer	0.9	0.5	381.85	13.917	19.541
	Poolformer	7.8	4.1	380.18	13.908	19.498
	ConvNext	7.9	4.2	367.39	13.516	19.168
	VAN	9.5	5.0	343.61	12.790	18.537
	HorNet	9.7	5.1	353.02	13.024	18.789
	MogaNet	10.0	5.3	**340.06**	12.738	18.441

Table 18: The performance on the multiple-variable latitude wind forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	15.5	43.3	30.789	3.8238	5.5488
	PredRNN	24.6	88.0	28.973	3.6617	5.3827
	PredRNN++	39.3	129.0	29.872	3.7067	5.4655
	MIM	41.7	35.8	36.464	4.2066	6.0386
	PhyDNet	3.1	11.3	54.389	5.1996	7.3749
	MAU	5.5	12.1	27.929	3.6700	5.2848
	PredRNNv2	24.6	88.5	31.120	3.8406	5.5785
Recurrent-free	SimVP	13.8	7.3	29.094	3.7614	5.3939
	TAU	9.6	5.0	25.456	3.4723	5.0454
	SimVPv2	10.0	5.3	28.058	3.6335	5.2970
	ViT	9.66	6.12	27.381	3.6068	5.2327
	Swin Transformer	9.7	5.2	27.097	3.5777	5.2055
	Uniformer	9.5	5.9	26.799	3.5676	5.1768
	MLP-Mixer	8.7	4.4	30.014	3.7840	5.4785
	ConvMixer	0.9	0.5	31.609	3.9104	5.6222
	Poolformer	7.8	4.1	35.161	4.0764	5.9296
	ConvNext	7.9	4.2	31.326	3.8435	5.5969
	VAN	9.5	5.0	25.720	3.4858	5.0715
	HorNet	9.7	5.1	30.028	3.7148	5.4798
	MogaNet	10.0	5.3	**25.232**	**3.4509**	**5.0231**

Table 19: The performance on the multiple-variable longitude wind forecasting in WeatherBench.

Method		Params (M)	FLOPs (G)	MSE ↓	MAE ↓	RMSE ↓
Recurrent-based	ConvLSTM	15.5	43.3	30.002	3.8923	5.4774
	PredRNN	24.6	88.0	27.484	3.6776	5.2425
	PredRNN++	39.3	129.0	28.396	3.7322	5.3288
	MIM	41.7	35.8	35.586	4.2842	5.9654
	PhyDNet	3.1	11.3	97.424	7.3637	9.8704
	MAU	5.5	12.1	27.582	3.7409	5.2519
	PredRNNv2	24.6	88.5	29.833	3.8870	5.4620
Recurrent-free	SimVP	13.8	7.3	28.782	3.8435	5.3649
	TAU	9.6	5.0	24.719	3.5060	4.9719
	SimVPv2	10.0	5.3	27.166	3.6747	5.2121
	ViT	9.7	6.1	26.595	3.6472	5.1570
	Swin Transformer	9.7	5.2	26.292	3.6133	5.1276
	Uniformer	9.5	5.9	25.994	3.6069	5.0985
	MLP-Mixer	8.7	4.4	29.242	3.8407	5.4076
	ConvMixer	0.9	0.5	30.983	3.9949	5.5662
	Poolformer	7.8	4.1	33.757	4.1280	5.8101
	ConvNext	7.9	4.2	29.764	3.8688	5.4556
	VAN	9.5	5.0	24.991	3.5254	4.9991
	HorNet	9.7	5.1	28.192	3.7142	5.3096
	MogaNet	10.0	5.3	**24.535**	**3.4882**	**4.9533**