

## A Additional Dataset

### A.1 Caltech-Pedestrian Dataset

The Caltech-Pedestrian Dollár et al. [2009] dataset presents challenging real-world urban scenarios involving diverse pedestrian movements, occlusions, and complex dynamics. It is evaluated using metrics such as Mean Square Error (MSE), Structural Similarity Index Measure (SSIM), Peak Signal-to-Noise Ratio (PSNR), and Learned Perceptual Image Patch Similarity (LPIPS) Zhang et al. [2018], testing the robustness and accuracy of predictive models. As shown in Table A, VideoTitans demonstrates competitive performance across these metrics, effectively capturing intricate pedestrian trajectories and spatial relationships. This highlights its practical applicability in dynamic environments, combining predictive accuracy with computational efficiency.

Method	MSE ( $\downarrow$ )	SSIM ( $\uparrow$ )	PSNR ( $\uparrow$ )	LPIPS ( $\downarrow$ )	FLOPs(G) ( $\downarrow$ )
ConvLSTM	139.6588	0.9345	27.4644	0.0857	595.0
E3D-LSTM	199.1374	0.9047	25.4612	0.1261	1004.0
MIM	<b>123.9034</b>	0.9410	28.1148	0.0642	1858.0
PhyDNet	310.6844	0.8615	23.2723	0.3218	40.4
PredRNN	129.3306	0.9375	27.8074	0.0745	1216.0
PredRNNv2	143.4366	0.9334	27.1864	0.0895	1223.0
MAU	177.4630	0.9174	26.1504	0.0969	172.0
TAU	128.9193	<b>0.9458</b>	27.8465	0.0551	80.0
SimVP-IncepU	160.2191	0.9338	26.8093	0.0675	60.6
SimVP-gSTA	<u>127.7992</u>	<u>0.9456</u>	27.9191	<u>0.0577</u>	96.3
SimVP-Swin	155.2470	0.9300	27.2542	0.0811	95.2
SimVP-Uniformer	135.9496	0.9393	27.6607	0.0687	104.0
SimVP-ViT	146.3816	0.9380	27.4267	0.0666	155.0
SimVP-Poolformer	153.3675	0.9334	27.3807	0.0700	79.8
VideoTitans	130.4290	0.9448	<b>28.8861</b>	<b>0.0512</b>	<b>9.9</b>

Table A: Performance comparison on Caltech Pedestrian dataset.

### A.2 KTH Dataset

The KTH Schuldt et al. [2004] dataset is characterized by structured human actions and stable motion patterns which test a model’s ability to capture temporal dynamics and spatial coherence in controlled settings. As shown in Table B VideoTitans achieves state-of-the-art performance across all evaluation metrics including MSE, Mean Absolute Error (MAE), PSNR, and SSIM. It combines high predictive accuracy with low computational complexity which confirms its practical effectiveness and shows its strength in modeling regular motion sequences with precision and efficiency.

Method	MSE ( $\downarrow$ )	MAE ( $\downarrow$ )	PSNR ( $\uparrow$ )	SSIM ( $\uparrow$ )	FLOPs(G) ( $\downarrow$ )
ConvLSTM	47.65	445.5	26.99	0.8977	1368.0
E3D-LSTM	136.40	892.7	21.78	0.8153	217.0
MIM	40.73	380.8	27.78	0.9025	1099.0
PhyDNet	91.12	765.6	23.41	0.8322	93.6
PredRNN	41.07	380.6	27.95	0.9097	2800.0
PredRNNv2	<u>39.57</u>	<u>368.8</u>	<u>28.01</u>	<u>0.9099</u>	2815.0
MAU	51.02	471.2	26.73	0.8945	399.0
TAU	45.32	421.7	27.10	0.9086	73.8
SimVP-IncepU	41.11	397.1	27.46	0.9065	<u>62.8</u>
SimVP-gSTA	45.02	417.8	27.04	0.9049	<u>76.8</u>
SimVP-Swin	45.72	405.7	27.01	0.9039	75.9
SimVP-Uniformer	44.71	404.6	27.16	0.9058	78.3
SimVP-ViT	56.57	459.3	26.19	0.8947	112.0
SimVP-Poolformer	45.55	400.9	27.22	0.9065	63.6
VideoTitans	<b>34.27</b>	<b>320.8</b>	<b>29.31</b>	<b>0.9197</b>	<b>50.9</b>

Table B: Performance comparison on KTH dataset.

## 18 B Qualitative Results

19 Figure A shows qualitative comparisons between VideoTitans, recurrent (PredRNNv2), and  
 20 transformer-based (ViT) methods on the Moving MNIST dataset. Due to the dataset’s relatively  
 21 simple dynamics, all models perform similarly well, making it challenging to visually distin-  
 22 guish significant differences among predictions. Empirically, we observe that differences primar-  
 23 ily lie in convergence speed rather than final per-  
 24 formance, as extending training epochs tends to  
 25 improve accuracy for all models. Nevertheless,  
 26 VideoTitans consistently provides slightly more  
 27 stable and accurate results. We also present qual-  
 28 itative results for t2m and uv10 variables from  
 29 the WeatherBench dataset. Further qualitative  
 30 results of VideoTitans are also available as GIF  
 31 animations for better visualization.  
 32  
 33  
 34  
 35

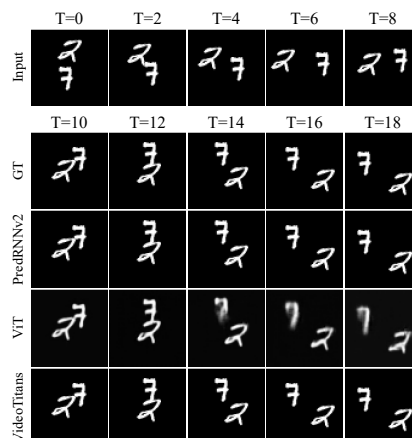


Figure A: Qualitative comparison of predicted frames on the Moving MNIST dataset.

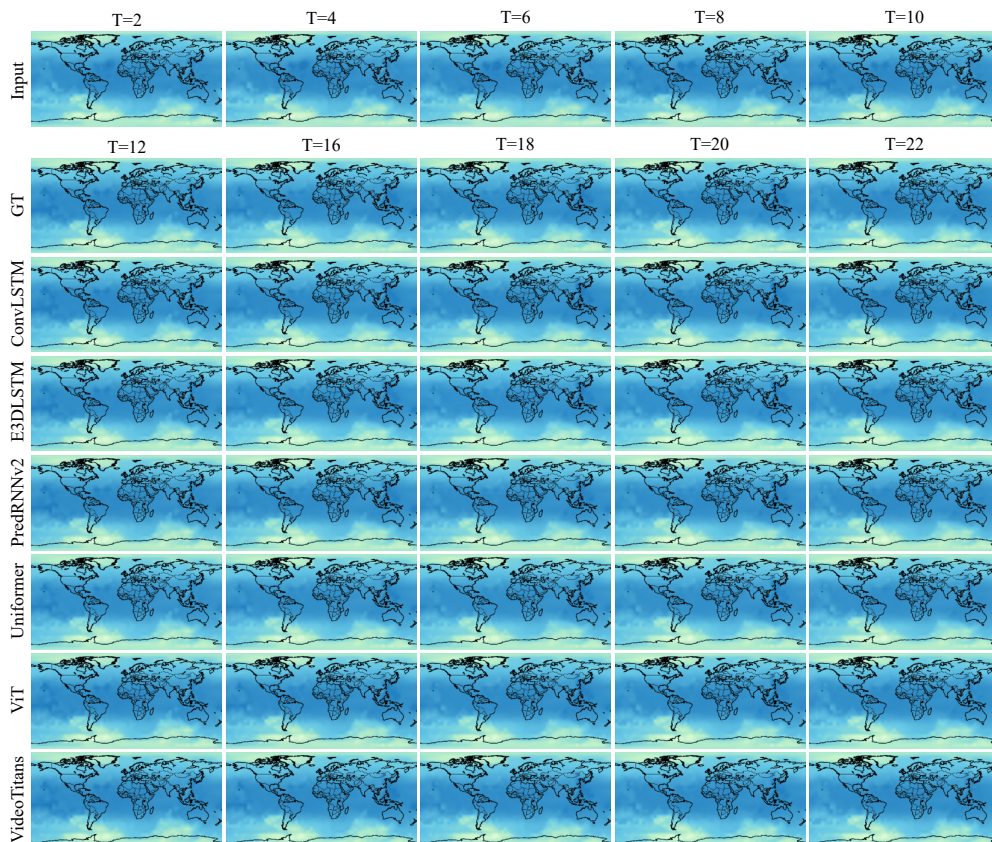


Figure B: A qualitative comparison of predicted 2m temperature (t2m) frames on the WeatherBench dataset, comparing VideoTitans’ predictions with those of other video prediction models.

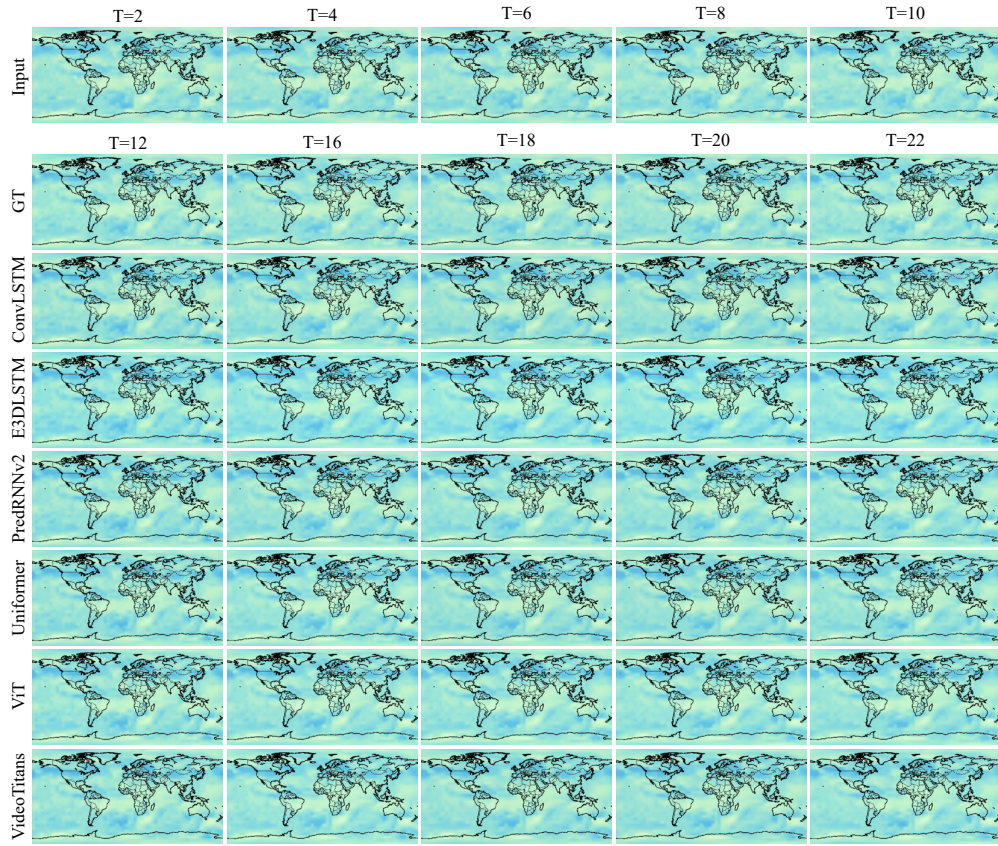


Figure C: A qualitative comparison of predicted wind field (uv10) frames on the WeatherBench dataset, where VideoTitans' predictions are compared with those of other video prediction models.

## 36 C Implementation Details

### 37 C.1 Model Architecture

38 The architecture of VideoTitans consists of three main components: an encoder for spatial feature  
39 extraction, a Titans-based temporal modeling module, and a decoder for frame reconstruction.

40 **Encoder.** The encoder captures spatial and low-level visual features from input frames. Given an  
41 input tensor of shape  $(B, T, C, H, W)$ , each frame is independently processed by convolution-based  
42 patch embedding. Specifically, we employ a convolutional layer with kernel size  $16 \times 16$  and stride  
43 16, converting the input as:

$$(B \times T, C, H, W) \rightarrow (B \times T, \text{embed\_dim}, H/16, W/16).$$

44 Afterward, spatial positional encodings are added to preserve positional information. The tensor is  
45 then reshaped for temporal processing:

$$(B \times T, \text{embed\_dim}, H/16, W/16) \rightarrow (B, H/16 \times W/16, T \times \text{embed\_dim}).$$

46 **Titans-based Temporal Modeling.** The temporal modeling is based on the Titans architecture,  
47 utilizing neural long-term memory that adaptively updates weights via a gradient-based surprise  
48 metric, efficiently capturing essential temporal patterns. Key hyperparameters, such as memory depth,  
49 memory dimension, persistent memory tokens, and maximum gradient norm, are critical for stable  
50 training. In particular, setting the maximum gradient norm to 1.0 prevents training instabilities such  
51 as gradient explosions.

52 The Titans module processes embeddings in segments, employing sliding window attention to model  
53 both local and global temporal dependencies. Persistent memory tokens encode context-independent  
54 knowledge to enhance generalization across datasets.

55 **Decoder.** The decoder reconstructs predicted frames from the temporal features. Mirroring the  
56 encoder structure, it utilizes transpose convolutional layers (kernel size  $16 \times 16$ , stride 16) to restore  
57 spatial dimensions:

$$(B, H/16 \times W/16, T \times \text{embed\_dim}) \rightarrow (B \times T, C, H, W).$$

58 The decoded frames are reshaped to the original dimensions  $(B, T, C, H, W)$  for comparison with  
59 ground-truth.

### 60 C.2 Training Procedure

61 We implement VideoTitans in PyTorch, using the Adam optimizer and Mean Squared Error (MSE)  
62 loss function. Key training parameters are summarized below:

- 63 • **Optimizer:** Adam optimizer ( $\beta_1 = 0.9, \beta_2 = 0.999$ ).
- 64 • **Learning Rate Scheduler:** ReduceLROnPlateau (patience=10 epochs), initial learning rate  
65 chosen from  $\{10^{-2}, 5 \times 10^{-3}, 10^{-3}, 5 \times 10^{-4}, 10^{-4}\}$ .
- 66 • **Batch Size:** 8 for all experiments.
- 67 • **Training Epochs:** MMNIST (200 epochs), Caltech Pedestrian (100 epochs), Human3.6,  
68 TrafficBJ, WeatherBench (50 epochs each).

69 Additionally, we apply the Exponential Moving Average (EMA) with a decay of 0.995 during training  
70 to enhance model stability and generalization.

### 71 C.3 Hyperparameter Sensitivity

72 A key contribution of our study includes identifying sensitive hyperparameters essential for VideoTi-  
73 tans’ stable training. Notably, removing gradient norm constraints (e.g., setting max gradient norm)  
74 caused training instabilities, and overly deep neural memory layers (depth  $> 2$ ) frequently result in  
75 numerical instability. Careful hyperparameter tuning is thus essential for robust training and optimal  
76 performance.

## 77 C.4 Memory Integration Strategies

78 There are three types of memory integration strategies in Titans: Memory as a Gate (MAG), Memory  
79 as a Context (MAC), and Memory as a Layer (MAL). MAG uses a gating mechanism to dynamically  
80 combine short-term attention and long-term memory, allowing the model to integrate previous knowl-  
81 edge adaptively. MAC retrieves past information from memory and appends it to the input sequence  
82 before processing it with attention, enabling selective use of historical data. MAL incorporates mem-  
83 ory as an independent processing layer before the attention, similar to traditional hybrid recurrent  
84 models. Among these approaches, MAG achieves the best performance by effectively balancing  
85 short-term precision with long-term recall, leading to its selection as the baseline model for our work.

## 86 References

- 87 Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark.  
88 In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*,  
89 2009.
- 90 Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm  
91 approach. In *Proceedings of the IEEE International Conference on Pattern Recognition (ICPR)*.  
92 IEEE, 2004.
- 93 Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable  
94 effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference*  
95 *on Computer Vision and Pattern Recognition (CVPR)*, 2018.