Supplementary Materials for AdamsFormer for Spatial Action Localization in the Future

A. Implementation Details

All experiments are conducted on Nvidia A100 GPU with PyTorch [9] framework. We use a 2D backbone pretrained on PASCAL VOC [8], and a 3D backbone pretrained on Kinetics [6] dataset. We freeze both backbones for all our experiments. For training, we use Adam [7] optimizer with an initial learning rate of 0.0001 and decrease it with a factor of 0.5 at $\{2, 3, 4, 5\}$ epochs for UCF101-24 dataset and $\{3, 4, 5, 6\}$ for JHMDB-21 dataset. Weight decay is set to 0.0005 for the generalization. We set the channel dimension D as 512 for all our experiments. For backbone, we use ResNet 18 [5] for 3D-CNN, and DarkNet [10] for 2D-CNN unless otherwise stated. We stack eight transformer encoder layers for the ODE function ($L = 8$). Multistep order (N) is set to 4 for UCF101-24 and 2 for JHMDB-21 unless specified. For the loss function for training, we set $\gamma = 2$ and $\lambda = 0.1$.

B. Details of the toy example

The toy experiment is conducted based on the example code of Neural ODE [1]. Specifically, we randomly generate a spiral function that follows $r = a + b \cdot \theta$, and sample data by adding random noise to the actual points from the function.

C. Detailed Architecture

We provide the detailed architecture of Adamsformer in Table 1 and descriptions of notations in Table 2. The output size of each module is presented. We use a causal mask for GPT-2 for parallel inference for the observed frames. We further provide detail of the decoder in Fig. 1.

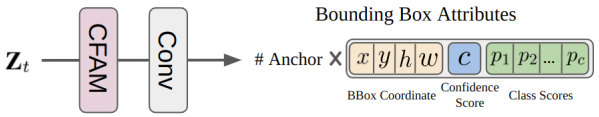

Figure 1. The decoder takes a latent tensor $\mathbf{Z}_t$ and produces a vector with channel #Anchor $\times$ (#Bbox elements + 1 + #Class) as described in Sec 4.3.3. Please refer to [31] for detailed architecture of the CFAM module.

	Blocks	Output Size	Output Size
Video Encoder	2D-CNN	$H \times W \times 3$	$H' \times W' \times D^{2D}$
	3D-CNN	$H \times W \times L \times 3$	$H' \times W' \times D^{3D}$
	Conv	$H' \times W' \times (D^{2D} + D^{3D})$	$H' \times W' \times D$
ODE Function	GPT-2	$T \times H' \times W' \times D$	$H' \times W' \times D$
Temporal Conv	-	$N \times H' \times W' \times D$	$H' \times W' \times D$
Video Encoder	CFAM	$H' \times W' \times D$	$H' \times W' \times D$
	Conv	$H' \times W' \times D$	$H' \times W' \times (5 \times (C+5))$

Table 1. Detailed architecture of the AdamsFormer.

Notations	Descriptions	Notations	Descriptions
H	Height of image	H'	Height of latent tensor
W	Width of image	W'	Width of latent tensor
L	Length of video clip	D	Latent tensor dimension
C	Number of action class	D^{2D}	2D-CNN output dimension
		D^{3D}	3D-CNN output dimension

Table 2. Summary of notations

D. Additional Experimental Results

D.1. Ablation Studies

We present the additional ablation study here. We compare ODE-RNN [1], which also utilizes ODE, with our proposed model by adding each component of AdamsFormer to ODE-RNN in Table 3. We first compare ODE-RNN and AdamsFormer with a stacked convolution layer as an ODE function that is described in Sec 5.4 in our main paper. ODE-RNN uses RNN to capture temporal information from the encoded values from Encoder. Then it uses captured features as input for the ODE function. In contrast, AdamsFormer directly takes the Encoder output as an input for the ODE function. Further, AdamsFormer is supervised to localize action even in observed frames. These differences lead to performance improvements, as shown in the table. Further, as we compared in Sec. 5.4 in our main paper, passing through all historical information as input for the ODE function is more effective for temporal modeling since it can help the model to attend to long-term temporal information.

D.2. Qualitative Results

We provide addional qualitative results in this section. For the figures, we use the model's outputs trained with a

Methods	Observation Ratio							
	20%		30%		40%		50%	
	TOTAL	UNSEEN	TOTAL	UNSEEN	TOTAL	UNSEEN	TOTAL	UNSEEN
ODE-RNN	-	34.84	-	35.59	-	37.71	-	39.70
Ours \w ODE function ($\mathbf{Z}_t$)	48.57	40.03	52.51	41.99	56.58	44.10	59.14	45.37
Ours \w ODE function ($\mathbf{Z}_{1:t}$)								
Single-Step	47.00	39.54	51.42	42.75	55.41	44.41	60.13	47.39
Multi-Step	50.04	41.00	52.82	42.92	57.03	45.25	62.21	48.74

Table 3. Comparison of AdamsFormer with ODE-RNN.

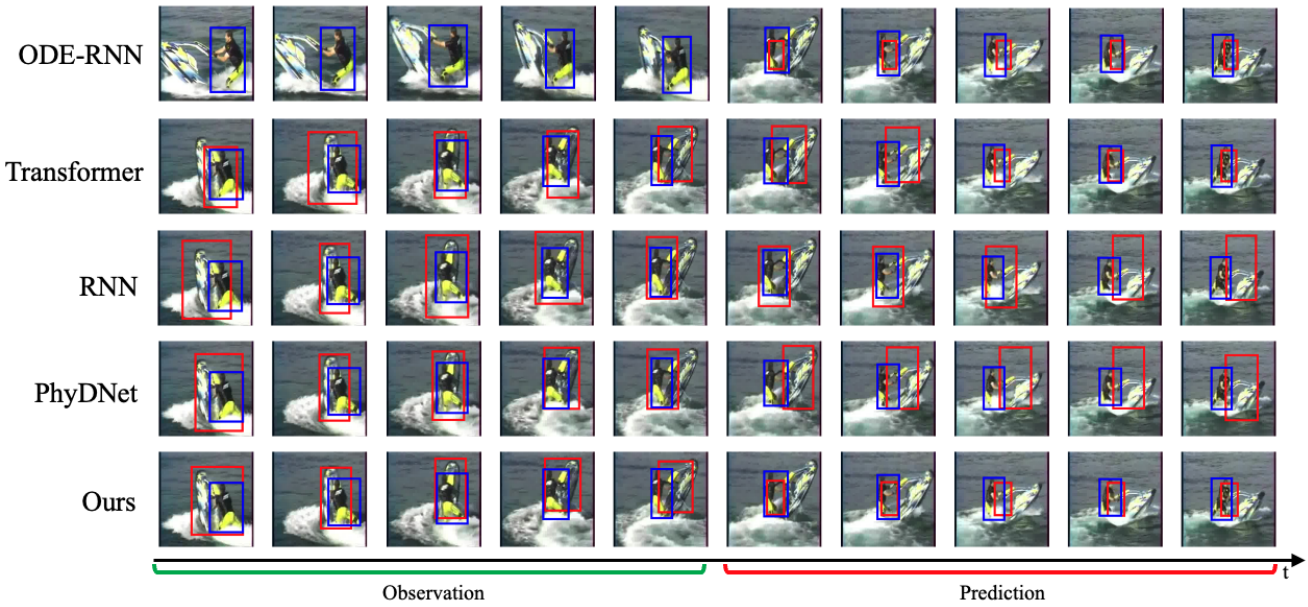

Figure 2. Comparison of qualitative results on UCF101-24 dataset. We visualize action localization results on all our baselines and AdamsFormer.

50 % of observation ratio. The red and blue boxes represent prediction and ground truth, respectively. The first five frames are localization results on observed sequence, whereas the next five frames are those of future frames.

Comparison with other methods We now compare qualitative results of different long-term temporal modeling methods [1, 3, 4, 11] on UCF101-24 dataset in Fig. 2. We can see that AdamsFormer localizes action more accurately than other methods by capturing actor dynamics in the observed frames.

Results on other datasets We present qualitative results for the JHMDB-21 and Collective Activity datasets. The JHMDB-21 results are depicted in Fig. 3. Despite the actions in the JHMDB-21 dataset being less dynamic compared to other datasets, AdamsFormer effectively localizes actions in future frames and outperforms other methods, as demonstrated in Table 1 of the main paper.

We further evaluate AdamsFormer's action localization performance in a multi-agent scenario using the Collective Activity dataset [2]. This dataset comprises 44 short video sequences featuring five activities: crossing, walking, waiting, talking, and queueing. Unlike UCF101-24 and JHMDB-21, which contain up to two agents per video, the Collective Activity dataset includes multiple agents in a single video. We apply AdamsFormer, trained on the UCF101-24 dataset with a 50% observation ratio, to localize actors within the Collective Activity dataset. The qualitative results are visualized in Fig. 4. Our findings indicate that AdamsFormer can successfully detect the locations of multiple actors.

E. Discussion & Limitations

An ODE function of AdamsFormer takes all previous latent features $\mathbf{Z}_{i:t}$ as input. However, this cannot be extended to a longer sequence because of the quadratically increasing memory requirement of the Transformer. One possible approach for this problem is leveraging the concept of long short-term memories [12] that were recently introduced for online action detection. We leave this for our future work. Also, the validity of this work on multiple subsequent actions remains to be tested. This warrants further investigation to demonstrate our model's usability.

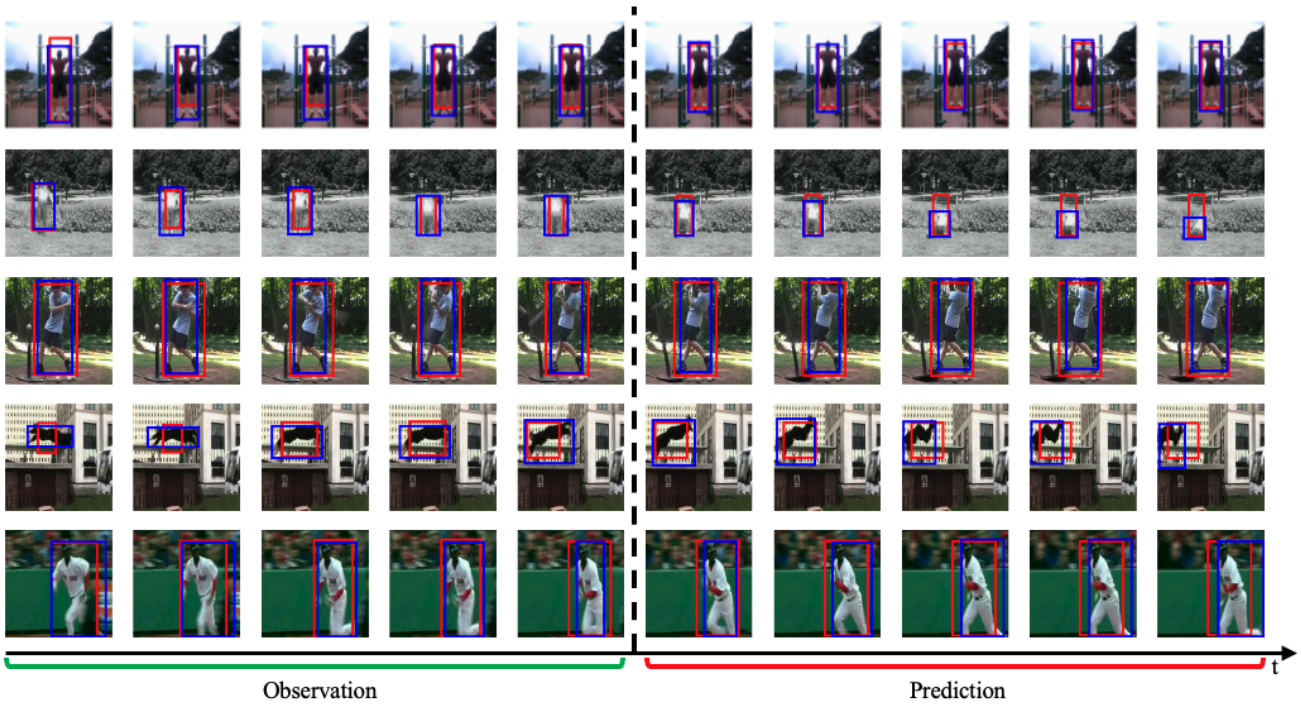

Figure 3. Qualitative results of AdamsFormer on JHMDB-21.

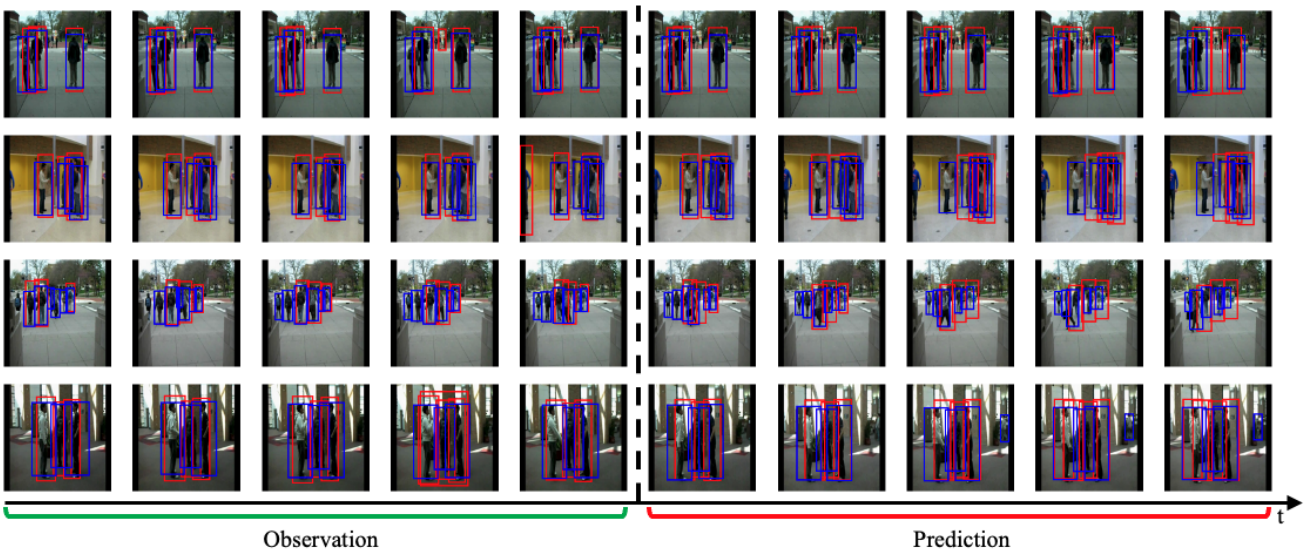

Figure 4. Qualitative results of AdamsFormer on Collective activity.

References

[1] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018. 1, 2

[2] Wongun Choi, Khuram Shahid, and Silvio Savarese. What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops*, pages 1282–1289. IEEE, 2009. 2

[3] Rohit Girdhar and Kristen Grauman. Anticipative video transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13505–13515, 2021. 2

[4] Vincent Le Guen and Nicolas Thome. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11474–11484, 2020. 2

[5] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018. 1

[6] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 1

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1

[9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 1

[10] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017. 1

[11] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 2015. 2

[12] Mingze Xu, Yuanjun Xiong, Hao Chen, Xinyu Li, Wei Xia, Zhuowen Tu, and Stefano Soatto. Long short-term transformer for online action detection. *Advances in Neural Information Processing Systems*, 34:1086–1099, 2021. 2