Supplementary: Memory-and-Anticipation Transformer for Online Action Understanding

Jiahao Wang[1*] Guo Chen[1,2*] Yifei Huang[2] Limin Wang[1,2] Tong Lu[1✉]
[1] State Key Laboratory for Novel Software Technology, Nanjing University
[2] Shanghai AI Laboratory

In the supplementary material, we describe the following parts:

- We report more details about the implementation and experimental settings of MAT.
- We give a detailed description and experimental result for MixClip+.
- We give attention visualization of our MAT for an intuitive understanding of the importance of entire temporal structures.
- We give more ablation studies of our MAT on THUMOS'14 [8] and EK100 [3].

1. More Details

1.1. Feature Extraction

In this section, we explain the detail of feature extraction for both action detection and anticipation using different video backbones [12, 5]. To extract features from TSN [12], we take the average of RGB features of 6 consecutive frames at 24 FPS to represent each frame at 4 FPS. Similarly, we stack optical flow maps of 5 frames preceding each frame along channel dimension at 24 FPS to obtain optical flow features for each frame at 4 FPS. Since ViT [5] requires an input of 16 RGB frames, we take 6 consecutive RGB frames at 24 FPS and then use bilinear interpolation to upsample it to align the input size, then we take the average of these features to represent each frame at 4 FPS.

1.2. Experiment Settings

Regarding training settings, we implement our proposed MAT in PyTorch[1]. We train MAT for 20 epochs using Adam [9] for optimization. The batch size is set to 16, and the learning rate was linearly increased from 6×10^{-7} to 7×10^{-5} in the first $\frac{1}{5}$ training iterations and then reduced to zero following a cosine function. When training MAT for action anticipation, we adopt equalization loss [11] to deal with the long tails of actions. Unless otherwise specified, we set m_L, m_S, and T_F to 256, 8, and 12 for TVSeries [4] and THUMOS'14 [8]; 128, 12, and 12 for HDD [10]; and 64, 5, 12 for EK100 [3] respectively.

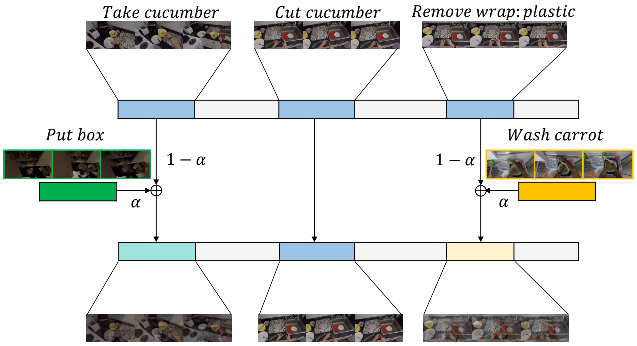

Figure 1: Illustration of Mixclip+. In the example sequence, there are 3 action instances, and 2 of them are fusion with another clip that comes from different videos with a random action category by a fusion coefficient α.

2. MixClip+

In this section, we will introduce MixClip+ and how we adopt it to the MAT model while training for online action anticipation on EK100. Given a short-term memory that is composed of a sequence of action instances $\{(t_i^{(s)}, t_i^{(e)}, a_i)\}$, where $t_i^{(s)}, t_i^{(e)}$ denotes the start and end time for the action instance and a_i denotes the action category. With the probability p_m, we mix each feature token of short-term memory with a random clip $\{(t_i^{(s)'}, t_i^{(e)'}, a_i')\}$ from different videos. To maintain the continuity of short-term memory, we adopt soft fusion with hyper-parameter α to mix features and the corresponding labels, which can be illustrated as:

$$\{f_t\}_{t=t_i^{(s)}}^{t_i^{(e)}} = (1-\alpha)\cdot\{f_t\}_{t=t_i^{(s)}}^{t_i^{(e)}} + \alpha\cdot\{f_{t'}\}_{t=t_i^{(s)'}}^{t_i^{(e)'}} \quad (1)$$

$$a_i = (1-\alpha)\cdot a_i + \alpha\cdot a_i'$$

[*] equal contribution, ✉ corresponding author (lutong@nju.edu.cn)

Long	Short	Overall			Unssen			Tail		
		Verb	Noun	Action	Verb	Noun	Action	Verb	Noun	Action
-	-	32.1	36.1	18.1	32.2	28.4	12.5	25.0	29.1	15.4
MixClip	-	32.9	36.9	18.4	29.5	29.1	11.2	25.6	30.3	15.2
MixClip+	-	31.6	36.6	18.2	30.2	26.8	12.1	24.4	30.5	15.5
-	MixClip	31.3	36.2	17.9	31.7	29.9	12.4	23.8	31.0	15.0
-	MixClip+	32.9	37.6	18.8	30.6	29.9	12.3	25.8	31.3	16.5
MixClip	MixClip	31.0	37.4	18.5	31.3	28.6	13.2	23.4	31.1	15.8
MixClip	MixClip+	**35.0**	**38.8**	**19.5**	**32.5**	**30.3**	**13.8**	**28.7**	**33.1**	**16.9**
MixClip+	MixClip	31.4	36.6	18.2	31.0	26.2	12.3	23.8	30.0	14.9
MixClip+	MixClip+	31.6	36.8	18.4	32.0	28.5	12.8	23.8	29.7	15.3

Table 1: Comparison to different augmentation choices on EPIC-Kitchens-100 Action Anticipation [3].

Method	Portion of Action									
	0-10%	10-20%	20-30%	30-40%	40-50%	50-60%	60-70%	70-80%	80-90%	90-100%
CNN [4]	61.0	61.0	61.2	61.1	61.2	61.2	61.3	61.5	61.4	61.5
LSTM[7]	63.3	64.5	64.5	64.3	65.9	64.7	64.4	64.4	64.4	64.3
TRN [14]	78.8	79.6	80.4	81.0	81.6	81.9	82.3	82.7	82.9	83.3
IDN [6]	80.6	81.1	81.9	82.3	82.6	82.8	82.6	82.9	83.0	83.9
OadTR [13]	79.5	83.9	86.4	85.4	86.4	87.9	87.3	87.3	85.9	84.6
Colar[16]	80.2	84.4	87.1	85.8	86.9	88.5	88.1	87.7	86.6	85.1
LSTR [15]	83.6	85.0	86.3	87.0	87.8	88.5	88.6	88.9	89.0	88.9
MAT (Ours)	**83.9**	**85.3**	**86.9**	**87.4**	**88.4**	**88.9**	**89.0**	**89.3**	**89.1**	**89.6**

Table 2: Online action detection results when only portions of videos. The results are considered in mcAP (%) on TVSeries using the TSN-Anet feature. MAT outperforms all existing methods for all portions of action considered.

ratio	mAP
0.25	69.9
0.5	70.2
0.75	**70.4**
1.0	70.0

(a) Keep ratio.

method	mAP
attn dropout	70.2
top-k selection	**70.4**
token dropping	69.5
w/o	70.0

(b) Different strategies.

stride	mAP
1	70.3
2	**70.4**
4	70.0
8	69.8
16	69.7

(c) Temporal stride.

α	V	N	A
0	32.9	36.9	18.4
0.1	32.5	37.1	18.9
0.25	**35.0**	**38.8**	**19.5**
0.5	31.4	37.5	19.0
0.75	30.6	36.8	18.5

(d) Fusion ratio.

Table 3: Ablation studys. We conduct detailed ablation on (a): Keep ratio, (b): Different strategies, (c): Temporal stride, and (d): Fusion ratio α. The gray rows denotes default choices.

Where $\{f_t\}_{t=t_i^{(s)}}^{t_i^{(e)}}$ and $\{f_{t'}\}_{t=t_i^{(s)\prime}}^{t_i^{(e)\prime}}$ denotes the origin action instance feature and a random clip instance feature from different videos, respectively. This input feature sequence is randomly cropped if the new instance's duration is longer. Otherwise, the feature sequence is padded to ensure that the length of the short-term is unchanged for ease of implementation. Fig 1 shows an illustration.

3. More Ablation Studies

Memory Dropping Strategy. We investigate the effectiveness of memory-dropping strategies and find that using top-k selection (top-75%) is 0.6% higher than not using any strategy, as shown in Table 3a and Table 3b. Adopting a simple dropout (ratio=0.2) for attention layers can also improve performance slightly. In contrast, token dropping (ratio=0.15), which randomly drops tokens during training and uses all tokens during testing, reduces accuracy. These findings suggest that row-wise random dropping (top-k selection and dropout) for attention weights is more effective in capturing long-term memory than dropping entire rows randomly (token dropping). Moreover, discarding weights with low relevance also helps improve performance.

Effect of α for MixClip+. In MixClip+, we define a hyperparameter α to achieve soft fusion. Table 3d indicates the effect of α on anticipation result. When no MixClip+ is applied, the baseline drops to 18.4% action recall. The performance consistently improves with MixClip+ and achieves the best (19.3%) at $\alpha = 0.25$.

Entire Ablation on MixClip+. Table 1 shows the results of using different data augmentations for long and short-term memory. When not using any data augmentation, MAT can still achieve great performance (18.1% action recall). Interestingly, using MixClip augmentation for short-term memory degrades model performance to a certain extent, indicating that MixClip may damage the continuity of actions in the short-term memory and cause the model to fail to learn short-term key motion information correctly. In contrast, MixClip+ can generate complex unseen samples while maintaining short-term continuity, enhancing the robustness of the model.

Effect of Future Stride. We implement MAT with m_S as 8 seconds, m_L as 256 seconds, and N_f as 24 seconds, test the effect of downsampling unseen future. Table 3c shows the results that too long or too short stride both affect the model's performance. It's interesting that when the number of future frames is held constant, the efficacy of future prediction over a 12-second interval with a stride of 1 is comparable to that of future prediction over a 24-second interval with a stride of 2.

Performance under different portions of actions. Following prior art [13, 15, 2], we also evaluate the accuracy of online action detection on TVSeries when only a certain portion of the action occurrences is considered. This evaluation aims to assess how well a method performs at different stages of an ongoing action. Following prior art, we divide each action occurrence into ten equal parts. We then compute a separate mcAP for each portion of action overall action occurrences. We tabulate the results across all action portions in Table 2. The table shows that our method outperforms all existing methods for all the different portions of action considered.

References

[1] https://pytorch.org/. 1

[2] Junwen Chen, Gaurav Mittal, Ye Yu, Yu Kong, and Mei Chen. Gatehub: Gated history unit with background suppression for online action detection. In *CVPR*, pages 19925–19934, 2022. 3

[3] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision (IJCV)*, 130:33–55, 2022. 1, 2

[4] Roeland De Geest, Efstratios Gavves, Amir Ghodrati, Zhenyang Li, Cees Snoek, and Tinne Tuytelaars. Online action detection. In *ECCV*, pages 269–284, 2016. 1, 2

[5] 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 *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 1

[6] Hyunjun Eun, Jinyoung Moon, Jongyoul Park, Chanho Jung, and Changick Kim. Learning to discriminate information for online action detection. In *CVPR*, pages 806–815, 2020. 2

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997. 2

[8] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/THUMOS14/, 2014. 1

[9] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 1

[10] Vasili Ramanishka, Yi-Ting Chen, Teruhisa Misu, and Kate Saenko. Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning. In *CVPR*, 2018. 1

[11] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 11659–11668. Computer Vision Foundation / IEEE, 2020. 1

[12] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, pages 20–36, 2016. 1

[13] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Yuanjie Shao, Zhengrong Zuo, Changxin Gao, and Nong Sang. Long short-term transformer for online action detection. In *ICCV*, 2021. 2, 3

[14] Mingze Xu, Mingfei Gao, Yi-Ting Chen, Larry S Davis, and David J Crandall. Temporal recurrent networks for online action detection. In *ICCV*, pages 5532–5541, 2019. 2

[15] Mingze Xu, Yuanjun Xiong, Hao Chen, Xinyu Li, Wei Xia, Zhuowen Tu, and Stefano Soatto. Long short-term transformer for online action detection. In *NeurIPS*, 2021. 2, 3

[16] Le Yang, Junwei Han, and Dingwen Zhang. Colar: Effective and efficient online action detection by consulting exemplars. In *CVPR*, 2022. 2