Augmented Self-Mask Attention Transformer for Naturalistic Driving Action Recognition

Tiantian Zhang*, Qingtian Wang*, Xiaodong Dong*, Wenqing Yu*, Hao Sun†, Xuyang Zhou, Aigong Zhen, Shun Cui, Dong Wu, Zhongjiang He
China Telecom Artificial Intelligence Technology (Beijing) Co., Ltd.
{zhangtt13, wangqt2, dongxd1, yuwq, sunh10, zhouxy26, zhenag, cuis2, wud21, hezj}@chinatelecom.cn

Abstract

Nowadays, naturalistic driving action recognition and computer vision techniques provide crucial solutions to identify and eliminate distracting driving behavior. Existing methods often extract features through fixed-size sliding windows and predict an action's start and end time. However, the information about a fixed-size window may be incomplete or redundant and the connections between different windows are insufficient. To alleviate this problem, we propose a novel ***Augmented Self-Mask Attention (AMA)*** *architecture that enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order. We employ an ensemble technique and use a weighted boundaries fusion to combine and refine predictions with high confidence scores action boundaries. On the test dataset of AI City Challenge 2024 Track3, we achieved significant results compared with other teams, the proposed model ranks first on the public leaderboard of the challenge. Codes are available at* https://github.com/wolfworld6/AIcity2024-track3.

1. Introduction

In real-world scenarios, distracted driving poses a significant risk to road safety. While computer vision (CV) focuses on detecting distracted driving incidents on the road, its effectiveness may be hindered by insufficient or low-quality data. To overcome these challenges, Track 3 of the AI City Challenge 2024 [22] has released a dataset and launched a competition focused on naturalistic driving action recognition (DAR). The dataset is collected using three cameras inside a stationary vehicle. There are sixteen distracted driving activities (such as phone call, eating, and reaching back) densely labeled in each video. The objective of the DAR competition is not only to accurately classify but also to localize action segments within an untrimmed video sequence, a problem known as temporal action localization (TAL).

TAL serves as a foundational task in video understanding, to detect all start and end instants from videos. Given its diverse applications spanning security surveillance, home care, video editing, and recommendation systems, among others, TAL has gained substantial attention within the research community in recent years.

State-of-the-art methods for these localization tasks leverage features extracted from video encoders typically pre-trained on large-scale datasets for action classification, such as Kinetics [11] and AVA [9]. However, these approaches utilize a set of sliding windows [7, 8, 16] or anchors sampled from pre-defined sliding windows [12, 13]. But the duration of an action varies greatly in a long video, as illustrated in Fig. 1. We analyze the distribution of different action durations in the dataset, there is a significant difference in the duration of the same action. The action in a fixed window may be incomplete or redundant.

Recently, transformer has shown remarkable performance in TAL [1, 10, 14, 26], which replaces global self-attention with local self-attention to decrease computational complexity. However, most of these methods are based on the local behavior. Namely, they conduct attention operations only in a local window. It is intuitive to leverage the global attention ability of transformer to model the relationship within different windows before prediction. However, only leveraging global attention works ineffective. Inspired by XLNet [25], we introduce AMA as a clip feature, possessing sequence characteristics akin to those found in Natural Language Processing (NLP) tasks.

In AMA, the self-mask method incorporates position information into the window feature and treats it in a way like autoregressive, enhancing the sequential characteristics and enabling models to capture bidirectional contexts. The overview of our pipeline is shown in Fig. 2. Moreover, we employ model ensemble with VideoMAE [17] and VideoMAEv2 [19]. A weighted boundaries fusion method

*These authors contributed equally to this work.
†Corresponding author, sunh10@chinatelecom.cn

is proposed for combining predictions of TAL models. This method significantly improves the quality of the combined predicted boundaries of temporal action.

In summary, the main contributions of this paper are summarized as follows:

- Introduce an **Augmented Self-Mask Attention** (**AMA**) which efficiently models relationships between different windows. Specifically, effectively modeling bidirectional temporal context within video sequences.
- Design a weighted boundaries fusion method for combining and refining predictions with high confidence scores action boundaries.
- We show that the proposed framework achieves first place in the AI City Challenge 2024 Track 3 final leaderboard results with a score of 0.8282.

2. Related Works

2.1. Video Recognition.

Video recognition stands as a cornerstone in video understanding, with substantial research endeavors dedicated to its advancement. The primary aim revolves around categorizing a condensed video into distinct action classes. Notably, [20] proposes the Temporal Segment Network (TSN) encoder to capture long-term temporal information. TSN along with other contemporary architectures such as R(2+1)D [18] and I3D [3] have become the de facto feature extractors for TAL. Recent masked autoencoder has shown excellent performance on self-supervised video representation learning such as BEVT [21], VideoMAE [17], VideoMAE V2 [19], MaskedFeat [23], and MAE-ST [6]. VideoMAE [17] is a simple masked video autoencoder with an asymmetric encoder-decoder architecture to handle the input sampled frames. VideoMAE V2 [19] proposes a dual-masked strategy to decrease pre-training overhead, and by expanding both the model size and dataset, it further explores the scalability of VideoMAE [17].

2.2. Temporal Action Localization.

In TAL algorithms, an intuitive idea is to pre-define a set of sliding windows of different time lengths and slide them over the video, such as S-CNN [16], TURN [8] and CBR [7]. Then, the action categories are judged one by one for the temporal intervals within each sliding window. Inspired by the two-stage object detection algorithm, the algorithm first generates some candidate temporal intervals from the video that may contain actions, and then judges the action classes within each candidate temporal interval and corrects the interval boundaries, such as R-C3D [24] and TAL-Net [4]. In addition, the idea of one-stage object detection can also be applied to temporal action localization, such as SSAD [12] and GTAN [13]. Recently, Actionformer [26] and React [15] propose a purely DETR-based design for TAL at multiple scales.

Model	Pretrained Datasets	Fine-tune Datasets	Feature Length
VideoMAE-l	ego-4d	A1	1024
VideoMAEv2-g	hybrid	A1	1408

Table 1. Public models pretrained on different datasets and fine-tuned on A1 dataset, with feature dimensions extracted from the A2 dataset.

3. Method

3.1. Data Preprocess

Follow [5], person detection is performed on each frame of the video, with the frame containing the largest detection area chosen as the reference for cropping. This approach ensures video stability by avoiding background fluctuations due to varying detection sizes. Cropping retains human body-related information while eliminating redundancy, reducing noise interference, and facilitating easier learning of human actions by the model.

3.2. Feature Extraction

Multiple experiments are conducted across various video representation models and three perspectives of A1 videos. For feature extraction, VideoMAE [17] and VideoMAEv2 [19] are chosen due to their superior performance in video recognition. Pre-trained weights from public datasets and fine-tuned specifically for A1 data, as detailed in Table 1. Each model is fine-tuned independently on videos from multiple perspectives, with features extracted from the A2 dataset.

3.3. Augmented Self-Mask Attention

In long videos, the duration of each action varies significantly. Some actions may be brief and occur rapidly, while others may unfold gradually over an extended period. This variability in action duration adds complexity to the task of localization, as the algorithm must accurately identify the start and end points of each action amidst the temporal fluctuations.

Actions in long videos often exhibit contextual dependencies, where the occurrence of one action may influence or be influenced by surrounding actions or events. Understanding these contextual dependencies is essential for accurate action localization, as it allows the algorithm to interpret actions within the broader context of the video sequence.

Despite the variability in action duration, there is often temporal consistency within long videos, where certain actions or patterns may recur or persist over time.

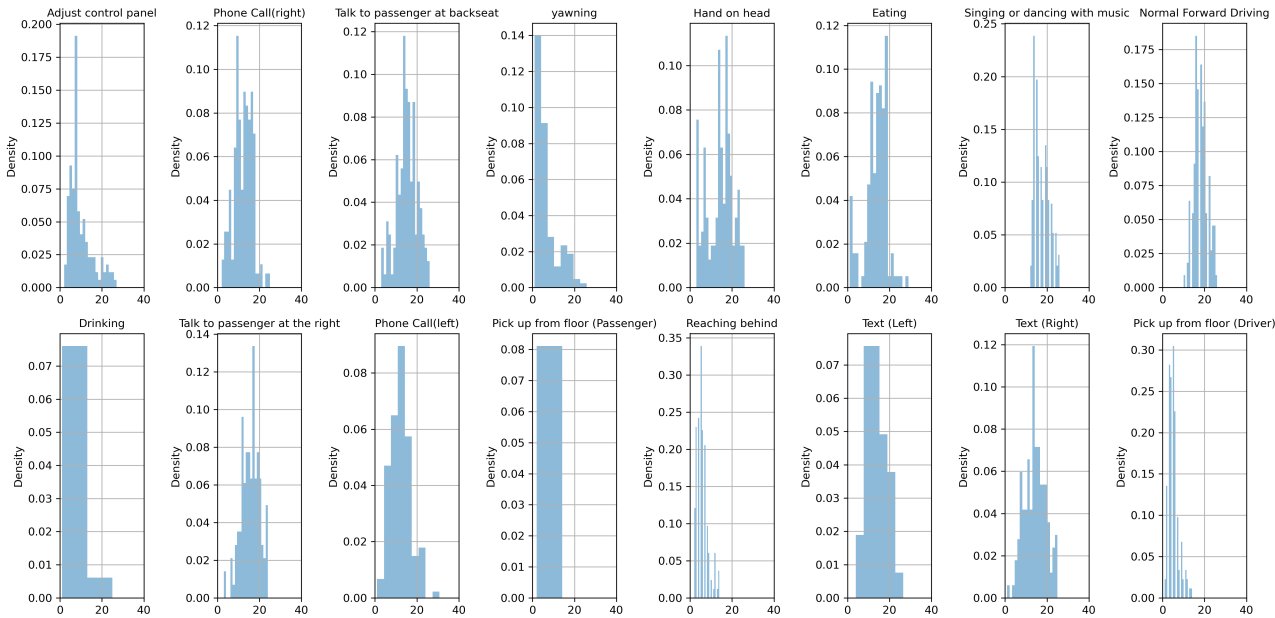

Figure 1. Distribution of Segment Differences for Each Label.The horizontal coordinate represents the duration of the action, and the vertical coordinate is the percentage of the action with different durations.

Detecting and leveraging these temporal consistencies can improve the accuracy and robustness of action localization algorithms. Actionformer [26] combines multi-scale feature representation with local self-attention and uses a lightweight decoder to classify every moment and estimate the corresponding action boundary. The integration of visual models with language models has shown promising performance in downstream visual tasks. Inspired by XL-Net [25], we utilize permutation-based training to capture bidirectional context for video feature frames. By leveraging permutations of the input sequence, we compute the likelihood of a token to all tokens, enabling effective modeling of bidirectional context for video feature, shown in Fig. 2.

In our approach, we employ a transformer encoder along with a pyramid network to encode feature sequences, thereby generating a multi-scale representation. To enrich this representation, we integrate AMA within the transformer encoder simultaneously and subsequently combine the resulting outputs. Moreover, using a novel framework to model the action sensitivity for both classification and localization tasks, taking into account the unique characteristics of each frame within action instances. This approach aims to enhance the performance of our model across various actions with various duration recognition tasks.

For normal temporal attention that is performed in the temporal dimension, input features generate query, key and value tensors $(Q, K, V) \in R^{T\times D}$, where T is the number of frames, D is the embedding dimension, then the output attention $S_a^{'}$ is calculated:

$$S_a^{'} = Softmax(\frac{QK^T}{\sqrt{D}})V, \tag{1}$$

For AMA, We incorporate relative positional embeddings derived from the original sequence. Next, we elaborate on integrating the recurrence mechanism into the proposed permutation framework to facilitate the reutilization of hidden states from preceding segments. For illustrative purposes, let us consider extracting segments from a longer sequence F; i.e., $\tilde{X} = F_{1:T}$ and $X = F_{T+1:2T}$. Let $\tilde{S}$ and S be permutations of $[1 \cdots T]$ and $[T+1 \cdots 2T]$ respectively. Subsequently, employing the permutation $\tilde{S}$, we address the initial segment and retain the resultant content representations $\tilde{H}\,(m)$ for each layer m. Subsequently, when processing the subsequent segment X, the attention update, integrating memory, can be formulated as follows:

$$H_{Z_T}^{(m)} = Softmax\left(Q = H_{Z_T}^{(m-1)}, KV = \left[\tilde{H}^{(m-1)}, H_{Z\leq T}^{(m-1)}\right]\right) \tag{2}$$

where $[\cdot,\cdot]$ denotes concatenation along the sequence dimension of frames. It is worth emphasizing that positional embeddings exclusively derive from the precise positions within the original sequence, devoid of external influences. Consequently, the attention update described above operates autonomously from the variable $\tilde{S}$ once the representations $\tilde{H}\,(m)$ have been acquired.

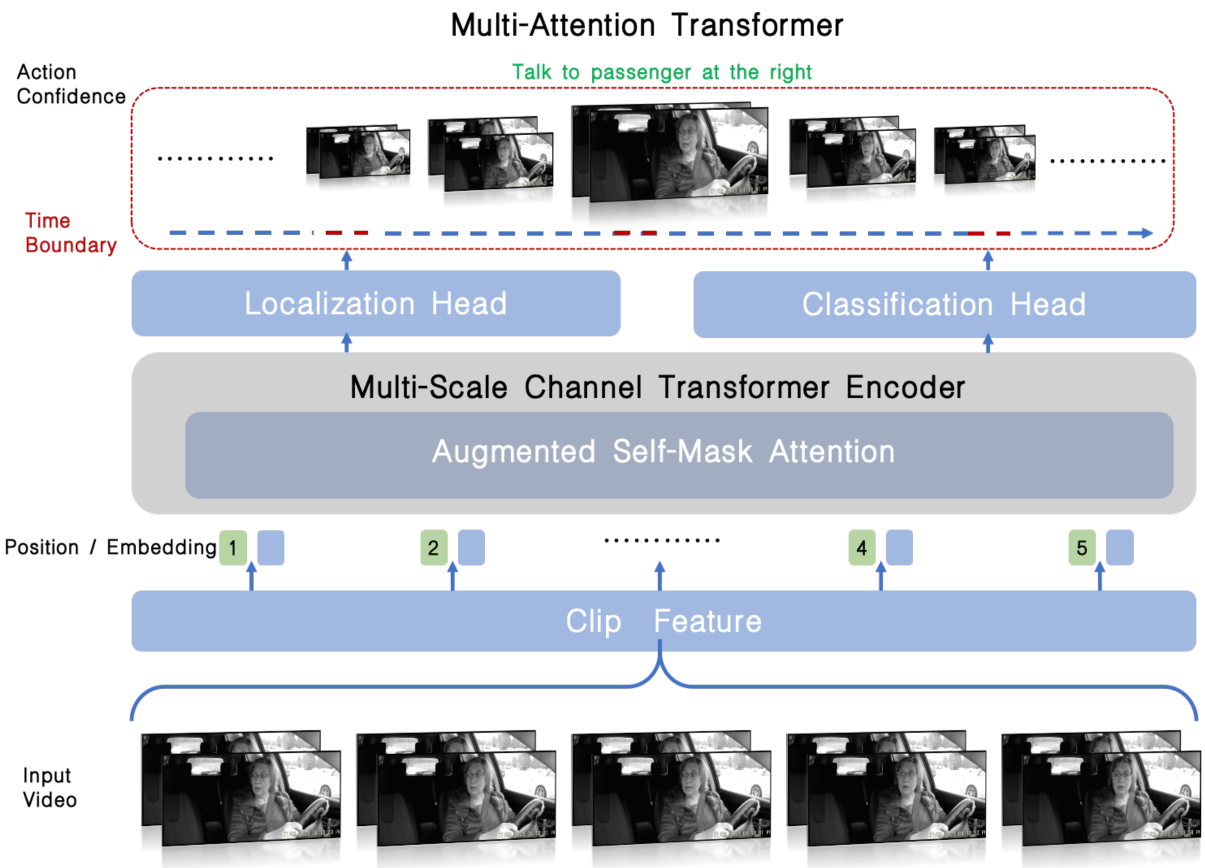

Figure 2. Overview of our model architecture. This method is composed of four parts: video feature extractor, feature encoder, AMA, and two sub-task heads. Given a video clip, we first leverage a trained VideoMAEv2 to extract the video feature and then utilize Transformer encoder to encode features. The weight of each frame during training is adapted according to its sensitivity to actions. In this module, the AMA augments and captures bidirectional context for video feature frames. Then each weight of the frame in training is adjusted based on classification and location head. A candidate action is generated at each time step through using the classification head to predict the action category and the regression head to predict the boundaries of the action time boundaries.

3.4. Ensemble Model

The output from the TAL often yields numerous predictions with varying confidence scores, resulting in a wide range of temporally overlapping regions. To adhere to the scoring criteria, each correct result should be associated with only one prediction, with minimal deviation in the time range. Consequently, it is necessary to filter out predictions and retain only those with high confidence levels. To address this, we employ model ensemble with VideoMAE [17] and VideoMAEv2 [19] to amalgamate and refine predictions with high confidence scores. This enables us to derive final results with enhanced temporal accuracy.

Firstly, each model selects the prediction with the highest score for each unique label among all prediction results for each video ID, discarding any redundant items. Then, we fuse the refined results from the aforementioned step, considering predictions with identical labels and video IDs. This fusion process is guided by the time Intersection over Union (tIoU) and the predictive score.

The start time and end time of the action in the i-th video-id and the j-th label can be calculated by the following formula:

$$\begin{aligned} ts_i{}^j &= \frac{1}{N}\sum_{p=1}^{N} start_p, \\ te_i{}^j &= \frac{1}{N}\sum_{p=1}^{N} end_p, \\ (start_p&, end_p) \in S_i{}^j. \end{aligned} \tag{3}$$

where ${ts_i}^j$ refers to the start time of the action in the i-th video-id and the j-th label, ${te_i}^j$ refers to the end time of the action in the i-th video-id and the j-th label. ${S_i}^j$ denotes the set of predictions where video-id is i and label is j. N is the length of ${S_i}^j$. $start_p$ refers to the start time of the p-th predictions in ${S_i}^j$, and end_p refers to the end time of the p-th predictions in ${S_i}^j$.

when fusing the results of the same video-id and the same label, we weight the fusion of time nodes according to their scores, which is formulated as:

$$\begin{aligned} {ts_i}^j &= \frac{\sum_{p=1}^{N} start_p * score_p}{\sum_{p=1}^{N} score_p}, \\ {te_i}^j &= \frac{\sum_{p=1}^{N} end_p * score_p}{\sum_{p=1}^{N} end_p}, \\ &(start_p, end_p, score_p) \in {S_i}^j. \end{aligned} \tag{4}$$

where $score_p$ refers to the score of the p-th predictions in ${S_i}^j$.

4. Experiment

4.1. Datasets

The dataset for Track 3 of the AI City Challenge 2024 [22] encompasses 594 video clips, amounting to approximately 90 hours of footage. These clips were recorded from 99 individual drivers. Each driver performs 16 distinct tasks randomly, such as phone calls and eating, with three synchronized cameras capturing different angles. Each driver completes the tasks twice: once without any appearance block and once with an appearance block like sunglasses or a hat. Consequently, there are six videos per driver – three without an appearance block and three with one, resulting in 594 videos. These videos are divided into three datasets: A1, A2, and B, each containing 69, 15, and 15 drivers.

To train the video action classification model, non-repetitive individuals are used in both the training and testing sets. All clip videos are divided into training and testing sets in a ratio of 5722:863 for better recognition of action features.

The main target of the challenge is to identify and localize distracted behaviors in test videos, which requires us to return the action category, starting time, and ending time of the distracted behavior.

4.2. Implementation Details

The implementation is based on the public toolbox Pytorch. All experiments are conducted on a workstation with eight A100 GPU cards of 40GB memory. We exploit VideoMAE-l and VideoMAEv2-g to guide the video encoder to conduct masked token-level reconstruction. We conduct experiments on the A1 dataset, dividing the data into training set and test set with a ratio of 7:3. Both two models are fine-tuned on the training set with training crop size 224. Other hyperparameters are shown in Tab. 2

	VideoMAE-l	**VideoMAEv2-g**
frame numbers	32	16
batch size	16	8
learning rate	5e-4	1e-3
epoch	25	35
feature length	1024	1408

Table 2. Hyperparameters of feature extraction models

In the training process, commencing with experimentation, results are obtained on A1, wherein 1408-D features are extracted using a fixed window size of 32 and a stride of 16 across all views' videos. The VideoMAE [17] and VideoMAEv2 [19] are employed. Hyper-parameters include a kernel size of 9, 8 heads, a mini-batch size of 2, and a maximum segment number set to 1536. The initial learning rate is 1e-4 with cosine decay, and a weight decay of 5e-2 is utilized. Model evaluation is conducted using mAP@[0.1:0.5:5]. The TAL model undergoes training for 20 epochs with a linear warmup phase of 5 epochs. During inference, the initial dense predictions are compressed with SoftNMS [2] and threshold 0.2, then remain 150 final predictions for submission.

4.3. Experiments Results

Ablation Study. The performance of the final Temporal Action Localization results on 50% of the A2 dataset is detailed in Table Tab. 3. Given the system's limited evaluation capacity, exhaustive exploration of all methods for each model combination is unfeasible. Consequently, we adopt the optimal processing method directly, informed by the observed patterns in each comparative experiment, to enhance the performance of superior models. As shown in Table Tab. 3, the model without the AMA module achieves an mAP@tIOU of 71.67 and an average overlap score of 0.80. However, with the introduction of the AMA module, there is a significant improvement in performance, with the mAP@tIOU increasing to 92.40 and the average overlap score rising to 0.8223. This indicates that the inclusion of the AMA module effectively enhances the model's performance. Moreover, the ensemble model with the AMA module achieves the highest average overlap score of 0.8242. This indicates that the ensemble with VideoMAE and VideoMAEv2 leads to further performance improvements.

Comparison with other teams. With the models trained on "A1" split, we infer "A2" split videos and submit our localization results to the evaluation system. Our proposed method ranks 1st with 0.8282 os score. The final leader

feature	model	mAP@tIOU	Average overlap score 50%
VideoMAE	w/o AMA	71.67	0.80
VideoMAE	AMA	92.40	0.8223
VideoMAEv2	**AMA**	**93.06**	**0.8234**
Ensemble	AMA	-	0.8242

Table 3. Comparison of the influence of different modules on the final performance. The proposed AMA leads to the most significant improvement.

Rank	Team name	Average overlap score
1	**TeleAI**	**0.8282**
2	supermonkey	0.8213
3	yptang	0.8149
4	Rockets	0.8045
5	SkkU Automation lab	0.7798
6	Bumblebee AIO	0.7624
7	boat	0.6844
8	MCPRL	0.6080
9	zzl	0.5963
10	USTC-IAT-United	0.2307

Table 4. Top 10 Leaderboard of Track3 in the AI City Challenge 2024.

board result is listed in Tab. 4, which validates the effectiveness and good generalization ability of the proposed approach.

5. Conclusion

In this paper, we have presented a solution for the Track 3 of the AI City Challenge 2024. We propose a novel Augmented Self-Mask Attention (AMA) architecture that enables learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order. AMA alleviates the problem that fixed-size sliding windows may be incomplete or redundant and the connections among different windows are insufficient. We also employ an ensemble and a weighted boundaries fusion to combine and refine predictions with high confidence scores action boundaries. Moreover, extensive experimentation is conducted, encompassing a wide array of video recognition models, feature extraction networks with varying lengths, and pre-trained datasets. Our method demonstrates significant potential to enhance TAL accuracy and robustness in real-world scenarios.

References

[1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. 1

[2] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis. Soft-nms – improving object detection with one line of code, 2017. 5

[3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. 2

[4] Y. W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar. Rethinking the faster r-cnn architecture for temporal action localization. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018. 2

[5] Xiaodong Dong, Ruijie Zhao, Hao Sun, Dong Wu, Jin Wang, Xuyang Zhou, Jiang Liu, Shun Cui, and Zhongjiang He. Multi-attention transformer for naturalistic driving action recognition. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 5435–5441, 2023. 2

[6] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. 2

[7] J. Gao, Z. Yang, and R. Nevatia. Cascaded boundary regression for temporal action detection. 2017. 1, 2

[8] J. Gao, Z. Yang, C. Sun, K. Chen, and R. Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. *IEEE*, 2017. 1, 2

[9] Chunhui Gu, Chen Sun, David A. Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. 1

[10] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. In *NeurIPS*, 2021. 1

[11] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017. 1

[12] T. Lin, X. Zhao, and Z. Shou. Single shot temporal action detection. *ACM*, pages 988–996, 2017. 1, 2

[13] F. Long, T. Yao, Z. Qiu, X. Tian, and T. Mei. Gaussian temporal awareness networks for action localization. *IEEE*, 2019. 1, 2

[14] Dingfeng Shi, Yujie Zhong, Qiong Cao, Lin Ma, Jia Li, and Dacheng Tao. Tridet: Temporal action detection with relative boundary modeling. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 18857–18866. IEEE, 2023. 1

[15] Dingfeng Shi, Yujie Zhong, Qiong Cao, Jing Zhang, Lin Ma, Jia Li, and Dacheng Tao. React: Temporal action detection with relational queries. Jul 2022. 2

[16] Z. Shou, D. Wang, and S. F. Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2

[17] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *arXiv preprint arXiv:2203.12602*, 2022. 1, 2, 4, 5

[18] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. 2

[19] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking, 2023. 1, 2, 4, 5

[20] 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*, page 20–36. Jan 2016. 2

[21] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Yu-Gang Jiang, Luowei Zhou, and Lu Yuan. Bevt: Bert pretraining of video transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14733–14743, June 2022. 2

[22] Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Meenakshi S. Arya, Anuj Sharma, Pranamesh Chakraborty, Sanjita Prajapati, Quan Kong, Norimasa Kobori, Munkhjargal Gochoo, Munkh-Erdene Otgonbold, Ganzorig Batnasan, Fady Alnajjar, Ping-Yang Chen, Jun-Wei Hsieh, Xunlei Wu, Sameer Satish Pusegaonkar, Yizhou Wang, Sujit Biswas, and Rama Chellappa. The 8th AI City Challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2024. 1, 5

[23] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14668–14678, June 2022. 2

[24] H. Xu, A. Das, and K. Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. *IEEE Computer Society*, 2017. 2

[25] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764, 2019. 1, 3

[26] Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part IV*, volume 13664 of *Lecture Notes in Computer Science*, pages 492–510. Springer, 2022. 1, 2, 3