GTA: Global Temporal Attention for Video Action Understanding

SUPPLEMENTARY MATERIAL

Bo He*[1]
bohe@umd.edu
Xitong Yang*[1]
xyang35@cs.umd.edu
Zuxuan Wu[2]
zxwu@fudan.edu.cn
Hao Chen[1]
chenh@umd.umd
Ser-Nam Lim[3]
sernamlim@fb.com
Abhinav Shrivastava[1]
abhinav@cs.umd.edu

[1] University of Maryland, College Park, MD, USA
[2] Fudan University, Shanghai, China
[3] Facebook AI, Sunnyvale, CA, USA

Section 1 reports additional results on the test set of Something-Something v1&v2. Section 2 presents more ablative study results of GTA. Section 3 elaborates on GTA that is designed for better temporal modeling. Section 4 shows more visualization results of global temporal attention weights, transformed regions and swapped attention. Finally, Section 5 provides dataset-specific implementation details on training and testing.

1 Testing Results on Something v1&v2

We compare the performance of our approach on the test set with the state-of-the-art methods on Something-Something v1 & v2 datasets. As is shown in Table 7, our approach based on 2D RestNet-50 with TSM backbone achieves 49.8% and 66.9% on SSv1 and SSv2 at top-1 accuracy, respectively. Although on SSv1 dataset, it is still below the $\text{TSM}_{\text{RGB+Flow}}$, $\text{TSM}_{\text{RGB+Flow}}$ is based on the two-stream network and utilizes additional optical flow information. With only RGB input, our GTA achieves the best performance among the recently proposed STM [7] and bLVNet-TAM [3] on 2D CNN backbone; I3D+NL+GCN [11] and TEA [8] on 3D CNN backbone.

*indicates equal contribution.

Method	Backbone	Frames	SSv1	SSv2
$TRN_{RGB+Flow}$ [13]	BNInc	8+8	40.7	56.2
TSM [8]	2D R50	16	46.0	64.3
$TSM_{RGB+Flow}$ [8]	2D R50	16+16	**50.7**	66.6
STM [6]	2D R50	16	43.1	63.5
bLVNet-TAM [5]	2D R101	64	48.9	-
ECO_{En}*Lite* [14]	BNInc+3D R18	92	42.3	-
I3D+NL+GCN [31]	3D R50	32	45.0	-
TEA [9]	3D R50	16	46.6	63.2
GTA_{En}	2D R50+TSM	16+8	49.8	**66.9**

Table 7: Results on the test set of Something-Something v1 & v2 datasets.

2 More Ablative Studies

Impact of inserting positions and number of blocks Table 8 explores the performance of different inserting positions and the number of blocks inserted. We see that even a single GTA block inserted at res_3 or res_4 can bring significant improvement over the baseline. However, the enhancement on res_5 is relatively minor. We hypothesize that the final residual stage loses too much fine-grained spatial information, which may hinder the learning of temporal attention at the pixel-level and the region-level. Following the common practice [17], our full model inserts five GTA blocks to leverage the complementary information provided by different residual stages and achieves the best result.

Comparison with Temporal Attention with Positional Embedding (TAPE) Our GTA module is more effective in temporal modeling than TAPE because it not only considers the chronological order of video frames but also models the temporal relationships among them. Results in Table 5 of the main paper show that GTA outperforms TAPE by **2.2%** on SSv1. Here, we provide a side-by-side comparison between TAPE and our Pixel GTA (without applying GTA to regions) in Table 9. Our Pixel GTA consistently outperforms TAPE under different settings. Furthermore, TAPE can also benefit from our cross-channel multi-head (CCMH) design, but Pixel GTA still achieves the best performance.

Impact of number of regions. We conduct experiments on the impact of the number of regions used in RegionGTA in Table 10. We can see that when increasing the number of regions from $C/32$ to C (C is the channel dimension of the feature map), the accuracy increase first and reach the peak when $K = C/8$. More importantly, our RegionGTA consistently outperforms the model without RegionGTA under different values of K, which proves the effectiveness of our RegionGTA design.

Comparison on cross-channel multi-head (CCMH) and multi-head. In Table 11, we compare the performance of cross-channel multi-head and multi-head. We can see that the accuracy drops by 0.5% when the cross-channel design is removed from CCHM. It demonstrates that the channel interaction is also critical to help improve the accuracy of the action recognition task.

res_3	res_4	res_5	Top-1
			17.0
+1			46.2
	+1		46.4
		+1	37.4
+1	+1		49.5
+2	+3		50.6

Table 8: Impact of inserting positions and number of blocks.

Model	w/o CCMH	w/ CCMH
+ TAPE	46.5	47.2
+ Pixel GTA	**48.0**	**48.5**
+ SA + TAPE	48.4	48.8
+ SA + Pixel GTA	**49.1**	**49.6**

Table 9: Ablation on positional embedding (TAPE) and cross-channel multi-head (CCMH) design.

Number of Regions	w/o RegionGTA	C	C/2	C/4	C/8	C/16	C/32
Top-1	49.6	49.7	50	50.3	**50.6**	50.3	50.1

Table 10: Impact of number of regions. *C* denotes the channel dimension of the feature map. Top-1 accuracy on SSv1 validation dataset are reported here.

Model	Top-1
Cross-channel Multi-head	**50.6**
Multi-head	50.1

Table 11: Comparison on cross-channel multi-head and multi-head.

3 Relations to Prior Work

Our proposed decoupled framework and the cross-channel multi-head (CCMH) design are the two key differences between GTA and the prior work (GloRe [2]). Specifically, our Region GTA generates semantic regions within each frame and performs temporal modeling on each region *individually* along the time axis. In contrast, when applied to spatio-temporal data, GloRe projects the whole 3D feature maps into semantic groups and models the interactions among them. We argue that this kind of grouping and modeling is not capable of capturing effective temporal relationships across different time steps. Moreover, GloRe leverages graph convolution to model node-wise interactions, which only considers information diffusion on each channel. Our GTA incorporates channel interactions to further improve temporal modeling, and we show its effectiveness in the experiments.

4 More Visualizations

Visualization of Global Temporal Attention Weights We provide visualization of the global temporal attention weights on two different datasets, Something-Something v1 and Kinetics-400 in Figure 5. Specifically, we average the learned global temporal attention weights across different groups and heads, and visualize the absolute value of attention weights. The darker colors represent larger values of weights. We can see that global attention weights of K400 and SSv1 are visually different. For SSv1, it tends to focus more on the latter part of the frames, while for Kinetics-400, the global temporal attention weights tend to focus more on the middle part of the frames. Our hypothesis is that because there are many action classes "pretending to do something", thus the latter part of the action are of vital importance to distinguish from "pretending to do something" vs "doing something". For example, for "pretending to pick something up" and "picking something up" actions,

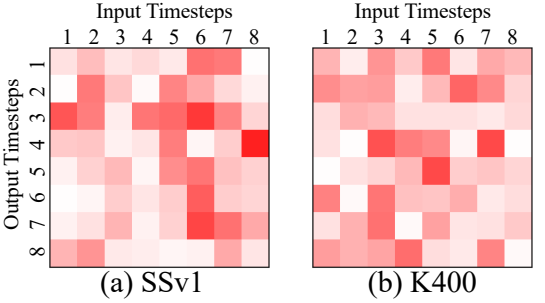

Figure 5: Visualization of global temporal attention weights on Something-Something v1 and Kinetics-400 datasets. Different columns represent timestamps of input from 1 to 8 and different rows represent timestamps of output from 1 to 8. Darker colors represent larger values of weights.

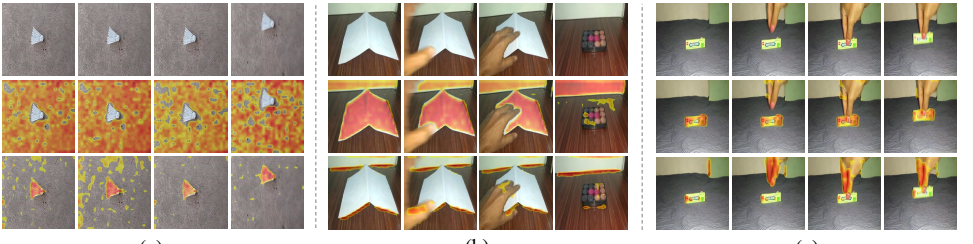

Figure 6: Visualization of the transformed regions of two examples: (a)“Turning the camera downwards while filming something"; (b)“Uncovering something"; (c) “Picking something up". The first row is the frame sequences. The second and third rows are regions obtained by Region GTA.

whether the object has been picked up eventually decides the action type. In addition, the global temporal attention weights are not flat across different timestamps, which verifies the effectiveness of our proposed GTA architecture.

Visualization of Transformed Regions We present visualization of the transformed regions in Figure 6. We can see that Region GTA can discover regions that share similar semantic meanings. For example, in the first video, the “ground” region and the “badminton” region are automatically identified, the “paper" and the “edge" are detected in the second video, and the “green gum" and the “hand" are obtained in the third video.

Visualization of Swapped Attention To further verify that different context information is needed for spatial and temporal attention, we present the visualization of the swapped attention maps in Figure 7. Specifically, we swap the attention functions (i.e., query/key/value projections) of the spatial and temporal attention blocks and visualize the attention maps. We can see that after swapping the spatial and temporal attention functions, the generated temporal attention maps focus more on the frames with similar objects instead of the frames that are useful for recognizing the action.

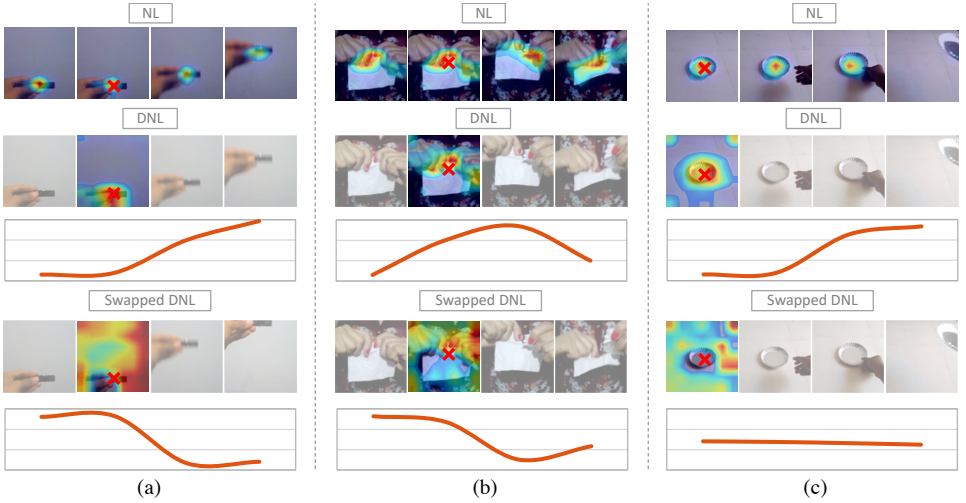

Figure 7: Visualization of the attention maps of three examples: (a)“Moving something up"; (b)“Tearing something into two pieces"; (c)“Picking something up". The first row is the spatio-temporal attention map generated by the non-local module. The second and third row is the spatial and temporal attention map obtained by our decoupled non-local module. The fourth and fifth row is the spatial and temporal attention map generated by swapping the attention functions of the spatial and temporal attention block. The red cross mark denotes the query position.

For example, in Figure 7(a), the temporal attention weights are larger in the first two frames which share a similar appearance with the same query position (i.e., the pen). Moreover, the spatial attention maps generated by the temporal attention functions also show substantially different patterns than the original ones. The visualization results further verify that different types of context information needed in spatial and temporal attention are captured in the decoupled non-local module.

5 Experiment Details

Something-Something v1&v2 [6] For the experiments based on the 2D CNN backbone, we follow the same sampling strategy as TSN [10] to sample 8 frames from the whole video. The same data augmentation is applied as TSN, which first resizes the input frames to 240×320, followed by the multi-scale cropping and random horizontal flipping. Note that we do not flip the clips which include the words “left" or “right" in their class labels (e.g., “pushing something from right to left"). We train the model for 50 epochs and start with a base learning rate of 0.01 with a batch size of 32. The first 2 epochs are used for linear warm-up [5] and the learning rate is reduced by a factor of 10 at 30, 40, 45 epochs. The backbone network is initialized with ImageNet pre-trained weights. For testing, we resize the input images to 240×320 pixels and center crop 224×224 pixels region. We sample 1 clip from each video for the experiments using 2D backbones.

For the experiments based on the 3D CNN backbone, we employ the same training and testing strategy as SlowFast-16×8-R50 [4]. We sample 16 and 64 frames for the slow and

fast pathways, respectively.

Kinetics-400 [1] For the experiments using 2D CNN backbones, we adopt R2D-50 as the backbone and use 8 frames as input. The model is initialized with ImageNet pre-trained weights and trained with step-wise learning schedule following the PySLowFast codebase [4]. For the experiments using 3D CNN backbones, we use SlowFast-8×8-R101 that samples 8 and 32 frames for the slow and fast pathway, respectively. We first train the backbone model on Kinetics-400 and then fine-tune it with GTA, following the same practice for training the non-local blocks [2]. We sample 10 clips temporally and 3 crops spatially from each video for testing.

References

[1] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[2] Yunpeng Chen, Marcus Rohrbach, Zhicheng Yan, Yan Shuicheng, Jiashi Feng, and Yannis Kalantidis. Graph-based global reasoning networks. In *CVPR*, 2019.

[3] Quanfu Fan, Chun-Fu Richard Chen, Hilde Kuehne, Marco Pistoia, and David Cox. More is less: Learning efficient video representations by big-little network and depth-wise temporal aggregation. In *NeurIPS*, 2019.

[4] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019.

[5] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[6] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017.

[7] Boyuan Jiang, MengMeng Wang, Weihao Gan, Wei Wu, and Junjie Yan. Stm: Spatiotemporal and motion encoding for action recognition. In *ICCV*, 2019.

[8] Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. Tea: Temporal excitation and aggregation for action recognition. In *CVPR*, 2020.

[9] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019.

[10] 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*, 2016.

[11] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *ECCV*, 2018.

[12] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2017.

[13] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *ECCV*, 2018.

[14] Mohammadreza Zolfaghari, Kamaljeet Singh, and Thomas Brox. Eco: Efficient convolutional network for online video understanding. In *ECCV*, 2018.