TSAM computationally efficient video understanding with audio modality

e-mail: aav.antonov@gmail.com

February 6, 2023

Abstract

For video understanding conventional 2D CNNs are computationally cheap but cannot capture temporal relationships (in difference to 3D CNNs or Transformers). Temporal Shift Module (TSM) was proposed to model temporal axes for 2D CNNs at no computational cost. TSM shifts part of the channels along the temporal dimension. Here, we introduce TSAM (Temporal Shift with Audio Modality) which extends TSM in several ways. First we incorporate audio modality in TSM architecture. Second, we parameterized temporal shift with the number of shifted segments (in the original implementation only neighboring segments were shifted). This improves performance with the increase of the number of sampled frames as facilitates information exchange between temporally distant frames. Finally we implemented an option to initialize weights pretrained from ImageNet21k resulting in further performance improvements. All these modifications improves performance of TSAM on action recognition benchmarks. In addition, availability of audio modality extends applicability of TSAM to video understanding domains where audio impact is important. The code is available at: https://github.com/aav-antonov/TSAM.

1 Introduction

Video understanding is a domain of computer vision aimed to make computers to gain high-level understanding from videos. Important aspect of video understanding is the need for temporal modeling. Order of frames is important and frequently leads to the situations when videos with similar frames but ordered differently to be from the "opposite" classes. This makes ability to learn temporal axes critical for the good performance of the network on video understanding tasks.

One of the most commonly studied tasks in this field consists of *action recognition*, the problem of recognizing what action is being performed in the video , [1] . In the last decade deep learning dominates the field of *action recognition* [2, 1, 3, 4, 5]. Availability of a temporal axis makes it necessary to

learn both spatial and temporal components which makes 3D convolution ideal fit. Good performance of 3D CNN architectures suffer from the large computational cost ([3]). Solutions were proposed to resolve the issue([4],[6]). On the other hand, solutions to adapt 2D CNN backbones that can efficiently learn both spatial and temporal information ([7]) have been developed. Recently transformer based video networks demonstrate superior performance on benchmark data set in comparison to CNN networks([8], [9]). However, transformer based architectures still suffer from the large computational cost [10].

Another active area of research in video understanding is *Affective video content analysis* [11]. In this case the aim is to predict evoked viewer reactions from the watched videos (input represent watched video content, output represent evoked viewer reactions). Models in the area of *action recognition* mostly ignore audio tracks and use only video related modalities [3, 4, 7, 12]. On the opposite, models in *affective video content analysis* are often multimodal [13], given that both visual and auditory features contribute to a video's affective content([14]). It is obvious that for emotion recognition the audio modalities bear much more information in comparison to *action recognition*.

TSM was originally proposed [7] as hardware-efficient 2D CNN video understanding architecture for real-world deployment. At the time of publication perfomance of TSM was comparable to 3DCNN architectures on most action recognition datasets. TSM was frequently used as backbone in multiple variations of 2D CNN architectures in action recognition. However, since introduction of transformers the best performing architectures ([5]) has made significant progress. In addition, original implementation of TSM ignore audio modality. This reduce potential of TSM to be used in *affective video content* applications.

Here we introduce several modifications of TSM which we refer further as TSAM (Temporal Shift with Audio Modality):

- we incorporate audio modality in TSM architecture
- parameterized temporal shift with the number of shifted segments
- weights pretrained from ImageNet21k

All this modifications improved performance of TSM on benchmark *action recognition* datasets (audio modality improves perfomance on kinetics 400). In addition, availability of audio modality makes TSAM attractive choice as backbone for *affective video content* applications.

2 Materials and methods

2.1 Temporal Shift Module

Efficient video understanding requires joint learning of spatial and temporal features. 3D CNN architecture [4, 15] could handle the problem but the computation cost is large. The use of 2D CNN was commonly less efficient [16, 17] as 2D CNN learns spatial features of individual frames but is not efficient to

model temporal information. To overcome this a temporal shift module (TSM) [7] was proposed with significant improvement in performance on action recognition datasets comparable with 3D CNN architectures. TSM processes input frames with Resnet backbone in parallel but, in addition, shifts features between temporally neighboring blocks before convolutional layers (Figure 3). This features exchange between neighboring frames while processing with backbone significantly improves ability to learn temporal features.

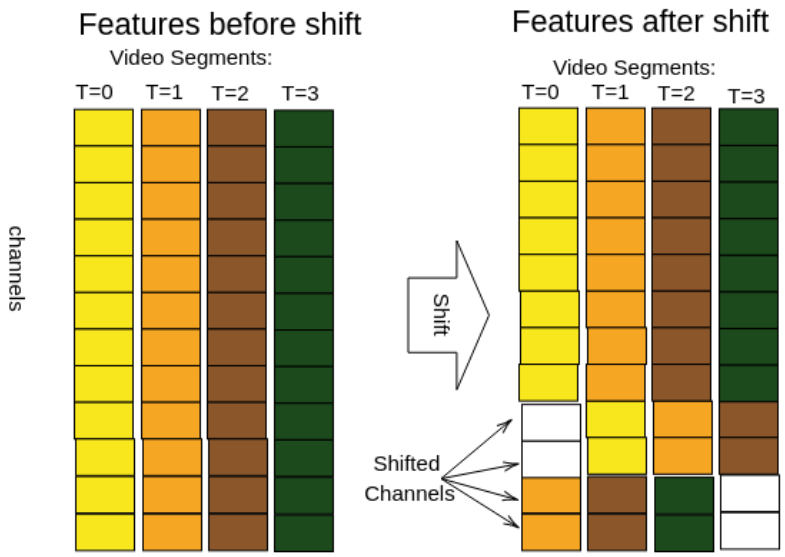

Figure 1: TSM: original implementation. Feature segments originated from frames are shifted by 1 segment both right and left (exchange only between temporally neighboring frames).

2.2 TSAM: parameterizing shift and Audio modality

We extend TSM by parameterizing shift with integer value k. In original implementation k was equal 1 and only temporarily neighbouring segments shift features between each other. In figure 2 we provide example when k = 2, 1/2 of shifted features are shifted by 1 segment and 1/2 of shifted features are shifted by 2 segments. In similar way the shift is defined for k=3,4 and so on. We refer to this parameter as "shift depth". This type of shift increases the speed of information exchange between temporally distant frames and improve performance when the number of sampled frames increases.

In addition, we extend TSM architecture with audio modality. To process the audio content, we initially converted the input audio signal into a 3-channel

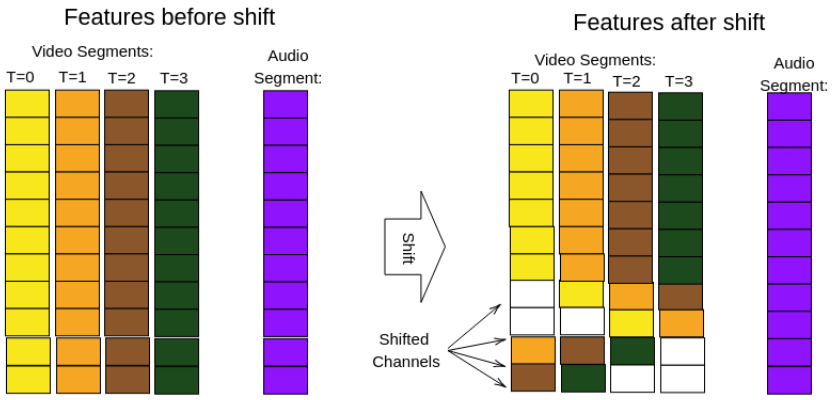

Figure 2: TSAM: shift of feature segments are parameterized by k: equal number of features are shifted by 1, 2 , .., k segments (k=2 in the figure). Audio segment is not shifted but processed by the same backbone (weights of network layers are the same for video and audio segments).

mel spectrogram. A log-mel spectrogram is a temporal sequence of spectra. The window size for computing spectra trades temporal resolution (short windows) against frequential resolution (long windows) [18, 19]. For the conversion of audio input, we used three different window sizes and hop lengths (25ms, 10ms, 50ms, 25ms, and 100ms, 50ms) on each of the channels, respectively, following previous recommendations [20, 21]. The resulting mel spectrograms, represented as images, were resized to match the size of the video frames (224 pixels).

Our implementation of TSAM video classification module uses Resnet50 [22] backbone to process both video and audio segments. Video segments are processed with temporal shift while audio segment (mel spectrogram) is processed by the same backbone without fusing with video features at early stages. Following drop out layer the output of the backbone for audio and video segments is averaged and fused in the last fully-connected layer to predict classification labels. Figure 3 illustrates our multi-modality classification network. We have found experimentally that early fusion, or shifting between audio and video segments, did not achieve the same performance.

2.3 INET21K weights initialization

Pretraining from ImageNet [23] weights is commonly improved classification perfomance on action recognition datasets in comparison with random weights initialization [7, 3]. Recently models of popular backbones like Resnet pretrained on much larger dataset ImageNet21K (INET21K) demonstrate superior perfor-

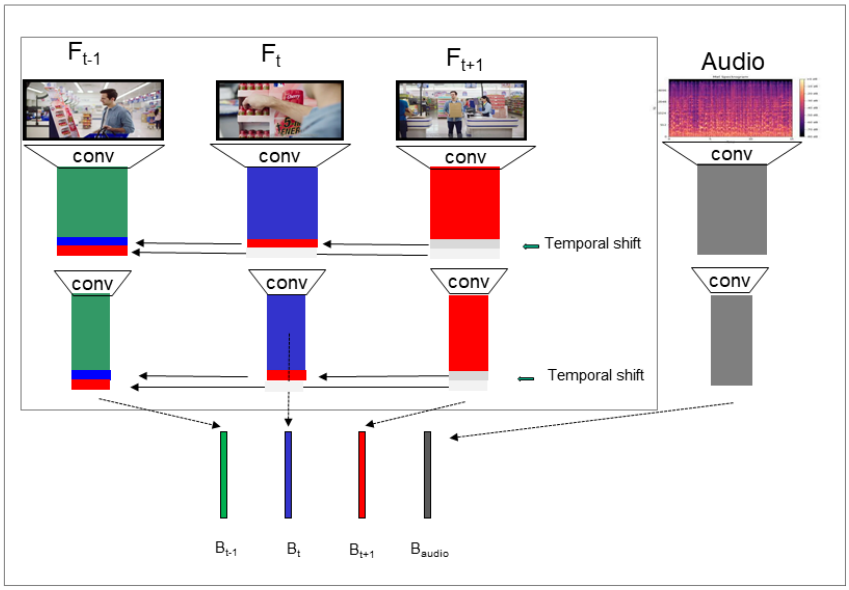

Figure 3: TSAM: video classification module. Resnet50 backbone fuse (shift) features between k-temporally neighboring frames. Audio input (Mel-Spectrogram) is processed by the same resnet50 backbone but without shifting of features. Output (resnet50) features from each frame ($F_t, t \in (1, .., n)$) and audio input (A_{audio}) are fused by averaging and mapped by fully connected layer to the output classes.

mance in comparison to weights initialization from ImageNet on a number of benchmark computer vision tasks [24, 25]. Presented here TSAM implementation can automatically use both weights initialization (ImageNet and INET21K).

3 Experimental results

3.1 Setup: Training and testing procedure

During the training phase, we optimize neural network hyper-parameters (dropout rate, batch size, learning rate schedule, and number of training epochs). In all experiments we follow commonly accepted protocol for sampling frames from the video and preprocessing them. Depending on the number of input frames k, the input video is divided into k equal segments. A random frame is sampled from each segment. For training purposes, each frame is resized to have 256×256 resolution and randomly cropped for the output image (the input for neural

network) to have resolution 224×224 pixels. For validation, we did center crop to get the output image to have 224×224 pixels. We also apply horizontal flip transformation as part of data augmentation procedure.

3.2 Performance assessment on public datasets

We benchmark TSAM on public action recognition datasets. We aim to demonstrate that TSAM achieves SOTA in the class of 2D CNN architectures. In addition, we demonstrate that even on action recognition datasets adding audio modality improves performance.

3.3 Kinetics 400: audio is important

In contrast to most frameworks in action recognition, our implementation along with standard RGB modality incorporates audio modality. We tested TSAM on the kinetics 400 dataset. First, kinetics 400 dataset [6] is one of the most popular video classification benchmark for action recognition and second, this dataset has an audio track in difference to the other popular benchmarks like Something-Something [26] or HMDB51 [27]. We compare our results to the original TSM implementation as well as to the deep learning architectures from other classes (3D CNN and transformers). Results are presented in table 1.

Method	Type	Modality	Frames	PreTrained	Accuracy
TSM (2019, [7])	2D CNN	RGB	8	ImageNet	74.2
TSAM	2D CNN	RGB	8	ImageNet	73.3
TSAM	2D CNN	RGB+audio	8+1	ImageNet	75.8
TSAM	2D CNN	RGB	8	INet21K	75.8
TSAM	2D CNN	RGB+audio	8+1	INet21K	77.8
TSM (2019, [7])	2D CNN	RGB	16	ImageNet	74.7
TSAM	2D CNN	RGB	16	ImageNet	73.8
TSAM	2D CNN	RGB+audio	16+1	ImageNet	76.4
TSAM	2D CNN	RGB	16	INet21K	76.3
TSAM	**2D CNN**	**RGB+audio**	**16+1**	**INet21K**	**78.1**
X3D-M(2020,[28])	3D CNN	RGB	16	-	76.0
X3D-L(2020,[28])	3D CNN	RGB	16	-	78.2
ViViT-L(2021,[29])	Transfomer	RGB	16	INet21K	80.6
UniFormer-S(2022[5])	Transfomer	RGB	16	ImageNet	80.8

Table 1: TSAM performance assessment on Kinetics 400: TSAM achieves state-of-the-art performances in the class of 2D CNN networks (for RGB modality with 16 frames + 1 frame audio).

We have also updated original implementation of the TSM by using weights pretrained on ImageNet21K (referred to as INET21K) ([25], [24]). In summary (Table 1) we demonstrate that on kinetics 400 dataset:

- audio modality improves results on kinetics 400 for about 1.5-2 percent in comparison with the performance of the network without it
- using INET21K weights improves performance for about 2-2.5 percent in comparison with the performance of the network pretrained with ImageNet 1K weights

3.4 Something-Something V1: deep shift improves performance

We also introduce a parameter k, which we refer to as "shift depth". This extends original TSM by shifting not only neghboring segments but also a distant ones defined by parameter k. We demonstrate that using $k = 4$ improves performance on Something-Something V1 dataset [26]. Results could be found in table 2. In summary we demonstrate that increasing k from 1 (original implementation of TSM) to 4 gradually improve performance.

Method	Type	Frames	ShiftDepth	PreTrained	Accuracy
TSM (2019, [7])	2D CNN	8	k=1	ImageNet	45.6
TSM (2019, [7])	2D CNN	16	k=1	ImageNet	47.2
TSAM	2D CNN	16	k=1	INet21K	50.1
TSAM	2D CNN	16	k=2	INet21K	50.4
TSAM	2D CNN	16	k=3	INet21K	50.7
TSAM	2D CNN	16	k=4	INet21K	51.1
TSAM	2D CNN	16	k=5	INet21K	51.0
TSAM	**2D CNN**	**16**	**k=4**	**Kinetics400**	**52.1**
CT-Net(2021, [30])	3D CNN	16	-	ImageNet	52.5
UniFormer-S(2022,[5])	Transfomer	16	-	Kinetics400	53.8

Table 2: Performance on Something-Something V1 (RGB modality only): TSAM achieves state-of-the-art performances in the class of 2D CNN networks (for RGB modality with 16 frames).

Overall, in the class of 2D CNN networks considering that we used only RGB modality (without motion detection like optical flow [31] , [32]) performance of TSAM, up to our knowledge, was one of the best. Increacing shift depth up to $k = 4$ gradually improve performance with 16 frames up to 1 percent. We also presented performance of other heavy architectures (in computational terms or data requirements) to demonstrate that TSAM performance is not far away from the best performance in the field.

4 Conclusions

This short technical report describes TSAM - extensions of Temporal Shift Module which boost performance on public *action recognition* datasets. TSAM remains computationally efficient as TSM with improved performance by 3-5 percent on public benchmark datasets. Implementation of audio modality

extends potential application in video understanding in the areas like *affective video content* analyses.

References

[1] T. Singh and D. K. Vishwakarma, "Video benchmarks of human action datasets: a review," *Artificial Intelligence Review*, vol. 52, no. 2, 2019.

[2] Y. Zhu, X. Li, C. Liu, M. Zolfaghari, Y. Xiong, C. Wu, Z. Zhang, J. Tighe, R. Manmatha, and M. Li, "A comprehensive study of deep video action recognition," *CoRR*, vol. abs/2012.06567, 2020. [Online]. Available: https://arxiv.org/abs/2012.06567

[3] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, 2019.

[4] C. Feichtenhofer, "X3D: Expanding Architectures for Efficient Video Recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2020.

[5] K. Li, Y. Wang, G. Peng, G. Song, Y. Liu, H. Li, and Y. Qiao, "Uniformer: Unified transformer for efficient spatial-temporal representation learning," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nBU_u6DLvoK

[6] J. Carreira and A. Zisserman, "Quo Vadis, action recognition? A new model and the kinetics dataset," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, 2017.

[7] J. Lin, C. Gan, and S. Han, "TSM: Temporal shift module for efficient video understanding," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, 2019.

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *CoRR*, vol. abs/2010.11929, 2020. [Online]. Available: https://arxiv.org/abs/2010.11929

[9] K. Li, Y. Wang, P. Gao, G. Song, Y. Liu, H. Li, and Y. Qiao, "Uniformer: Unified transformer for efficient spatiotemporal representation learning," *CoRR*, vol. abs/2201.04676, 2022. [Online]. Available: https://arxiv.org/abs/2201.04676

[10] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, and D. Tao, "A survey on vision transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2022. [Online]. Available: https://doi.org/10.1109%2Ftpami.2022.3152247

[11] A. Hanjalic and L. Q. Xu, "Affective video content representation and modeling," *IEEE Transactions on Multimedia*, vol. 7, no. 1, 2005.

[12] C. Zhang, Y. Zou, G. Chen, and L. Gan, "PAN: Persistent appearance network with an efficient motion cue for fast action recognition," in *MM 2019 - Proceedings of the 27th ACM International Conference on Multimedia*, 2019.

[13] Y. Baveye, C. Chamaret, E. Dellandrea, and L. Chen, "Affective video content analysis: A multidisciplinary insight," *IEEE Transactions on Affective Computing*, vol. 9, no. 4, 2018.

[14] J. J. Sun, T. Liu, A. S. Cowen, F. Schroff, H. Adam, and G. Prasad, "EEV Dataset: Predicting Expressions Evoked by Diverse Videos," *arXiv*, 2020.

[15] H. Rahmani and A. Mian, "3D action recognition from novel viewpoints," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, 2016.

[16] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9912 LNCS, 2016.

[17] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal Segment Networks for Action Recognition in Videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 11, 2019.

[18] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S. Y. Chang, and T. Sainath, "Deep Learning for Audio Signal Processing," *IEEE Journal on Selected Topics in Signal Processing*, vol. 13, no. 2, 2019.

[19] K. W. Cheuk, K. Agres, and D. Herremans, "The Impact of Audio Input Representations on Neural Network based Music Transcription," in *Proceedings of the International Joint Conference on Neural Networks*, 2020.

[20] K. Palanisamy, D. Singhania, and A. Yao, "Rethinking CNN models for audio classification," *CoRR*, vol. abs/2007.11154, 2020. [Online]. Available: https://arxiv.org/abs/2007.11154

[21] A. Recasens, P. Luc, J. Alayrac, L. Wang, F. Strub, C. Tallec, M. Malinowski, V. Patraucean, F. Altché, M. Valko, J. Grill, A. van den Oord, and A. Zisserman, "Broaden your views for self-supervised video learning," *CoRR*, vol. abs/2103.16559, 2021. [Online]. Available: https://arxiv.org/abs/2103.16559

[22] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, 2017.

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2010.

[24] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, "Imagenet-21k pretraining for the masses," 2021.

[25] R. Wightman, "Pytorch image models," https://github.com/rwightman/pytorch-image-models, 2019.

[26] R. Goyal, S. E. Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic, "The 'Something Something' Video Database for Learning and Evaluating Visual Common Sense," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-October, 2017.

[27] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: a large video database for human motion recognition," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2556–2563.

[28] C. Feichtenhofer, "X3D: expanding architectures for efficient video recognition," *CoRR*, vol. abs/2004.04730, 2020. [Online]. Available: https://arxiv.org/abs/2004.04730

[29] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lucic, and C. Schmid, "Vivit: A video vision transformer," *CoRR*, vol. abs/2103.15691, 2021. [Online]. Available: https://arxiv.org/abs/2103.15691

[30] K. Li, X. Li, Y. Wang, J. Wang, and Y. Qiao, "Ct-net: Channel tensorization network for video classification," *CoRR*, vol. abs/2106.01603, 2021. [Online]. Available: https://arxiv.org/abs/2106.01603

[31] C. Zach, T. Pock, and H. Bischof, "A Duality Based Approach for Realtime TV-{\itshape L}¡Superscript¿1¡/Superscript¿ Optical Flow," in *Pattern Recognition*, vol. 4713, 2007.

[32] L. Fan, W. Huang, C. Gan, S. Ermon, B. Gong, and J. Huang, "End-to-End Learning of Motion Representation for Video Understanding," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018.