VidTr: Video Transformer Without Convolutions

Yanyi Zhang [1,2] *, Xinyu Li [1]*, Chunhui Liu [1], Bing Shuai [1], Yi Zhu [1],
Biagio Brattoli [1], Hao Chen [1], Ivan Marsic [2] and Joseph Tighe [1]
[1] Amazon Web Service; [2] Rutgers University
{xxnl,chunhliu,bshuai,yzaws,biagib,hxen,tighej}@amazon.com; {yz593,marsic}@rutgers.edu

Abstract

We introduce Video Transformer (VidTr) with separable-attention for video classification. Comparing with commonly used 3D networks, VidTr is able to aggregate spatio-temporal information via stacked attentions and provide better performance with higher efficiency. We first introduce the vanilla video transformer and show that transformer module is able to perform spatio-temporal modeling from raw pixels, but with heavy memory usage. We then present VidTr which reduces the memory cost by 3.3× while keeping the same performance. To further optimize the model, we propose the standard deviation based topK pooling for attention ($pool_{topK\_std}$), which reduces the computation by dropping non-informative features along temporal dimension. VidTr achieves state-of-the-art performance on five commonly used datasets with lower computational requirement, showing both the efficiency and effectiveness of our design. Finally, error analysis and visualization show that VidTr is especially good at predicting actions that require long-term temporal reasoning.

1. Introduction

We introduce Video Transformer (VidTr) with separable-attention, one of the first transformer-based video action classification architecture that performs global spatio-temporal feature aggregation. Convolution-based architectures have dominated the video classification literature in recent years [19, 32, 55], and although successful, the convolution-based approaches have two drawbacks: 1. they have limited receptive field on each layer and 2. information is slowly aggregated through stacked convolution layers, which is inefficient and might be ineffective [31, 55]. Attention is a potential candidate to overcome these limitations as it has a large receptive field which can be leveraged for spatio-temporal modeling. Previous works use attention to modeling long-range spatio-temporal features in videos but still rely on convoluational backbones [31, 55]. Inspired by recent successful applications of transformers on NLP [12, 52] and computer vision [14, 47], we propose a transformer-based video network that directly applies attentions on raw video pixels for video classification, aiming at higher efficiency and better performance.

We first introduce a vanilla video transformer that directly learns spatio-temporal features from raw-pixel inputs via vision transformer [14], showing that it is possible to perform pixel-level spatio-temporal modeling. However, as discussed in [56], the transformer has $\mathbb{O}(n^2)$ complexity with respect to the sequence length. The vanilla video transformer is memory consuming, as training on a 16-frame clip (224×224) with only batch size of 1 requires more than 16GB GPU memory, which makes it infeasible on most commercial devices. Inspired by the R(2+1)D convolution that breaks down 3D convolution kernel to a spatial kernel and a temproal kernel [50], we further introduce our separable-attention, which performs spatial and temporal attention separately. This reduces the memory consumption by 3.3× with no drop in accuracy. We can further reduce the memory and computational requirements of our system by exploiting the fact that a large portion of many videos have redundant information temporally. This notion has been explored in the context of convolutional networks to reduce computation previously [32]. We build on this intuition and propose a standard deviation based topK pooling operation ($topK\_std$ pooling), which reduces the sequence length and encourages the transformer network to focus on representative frames.

We evaluated our VidTr on 6 most commonly used datasets, including Kinetics 400/700, Charades, Something-something V2, UCF-101 and HMDB-51. Our model achieved state-of-the-art (SOTA) or comparable performance on five datasets with lower computational requirements and latency compared to previous SOTA approaches. Our error analysis and ablation experiments show that the VidTr works significantly better than I3D on activities that requires longer temporal reasoning (e.g. making a cake vs. eating a cake), which aligns well with our intuition.

*Equally Contributed.

This also inspires us to ensemble the VidTr with the I3D convolutional network as features from global and local modeling methods should be complementary. We show that simply combining the VidTr with a I3D50 model (8 frames input) via ensemble can lead to roughly a 2% performance improvement on Kinetics 400. We further illustrate how and why the VidTr works by visualizing the separable-attention using attention rollout [1], and show that the spatial-attention is able to focus on informative patches while temporal attention is able to reduce the duplicated/non-informative temporal instances. Our contributions are:

1. **Video transformer:** We propose to efficiently and effectively aggregate spatio-temporal information with stacked attentions as opposed to convolution based approaches. We introduce vanilla video transformer as proof of concept with SOTA comparable performance on video classification.
2. **VidTr:** We introduce VidTr and its permutations, including the VidTr with SOTA performance and the compact-VidTr with significantly reduced computational costs using the proposed standard deviation based pooling method.
3. **Results and model weights:** We provide detailed results and analysis on 6 commonly used datasets which can be used as reference for future research. Our pre-trained model can be used for many down-streaming tasks.

2. Related Work

2.1. Action Classification

The early research on video based action recognition relies on 2D convolutions [28]. The LSTM [25] was later proposed to model the image feature based on ConvNet features [30, 51, 63]. However, the combination of ConvNet and LSTM did not lead to significantly better performance. Instead of relying on RNNs, the segment based method TSN [54] and its permutations [22, 35, 64] were proposed with good performance.

Although 2D network was proved successful, the spatio-temporal modeling was still separated. Using 3D convolution for spatio-temporal modeling was initially proposed in [26] and further extended to the C3D network [48]. However, training 3D convnet from scratch was hard, initializing the 3D convnet weights by inflate from 2D networks was initially proposed in I3D [7] and soon proved applicable with different type of 2D network [10, 24, 58]. The I3D was used as backbone for many following work including two-stream network [19, 55], the networks with focus on temporal modeling [31, 32, 59], and the 3D networks with refined 3D convolution kernels [27, 33, 39, 44].

The 3D networks are proved effective but often not efficient, the 3D networks with better performance often requires larger kernels or deeper structures. The recent research demonstrates that depth convolution significantly reduce the computation [49], but depth convolution also increase the network inference latency. TSM [37] and TAM [17] proposed a more efficient backbone for temporal modeling, however, such design couldn't achieve SOTA performance on Kinetics dataset. The neural architecture search was proposed for action recognition [18, 43] recently with competitive performance, however, the high latency and limited generalizability remain to be improved.

The previous methods heavily rely on convolution to aggregate features spatio-temporally, which is not efficient. A few previous work tried to perform global spatio-temporal modeling [31, 55] but still limited by the convolution backbone. The proposed VidTr is fundamentally different from previous works based on convolutions, the VidTr doesn't require heavily stacked convolutions [59] for feature aggregation but efficiently learn feature globally via attention from first layer. Besides, the VidTr don't rely on sliding convolutions and depth convolutions, which runs at less FLOPs and lower latency compared with 3D convolutions [18, 59].

2.2. Vision Transformer

The transformers [52] was previously proposed for NLP tasks [13] and recently adopted for computer vision tasks. The transformers were roughly used in three different ways in previous works: 1.To bridge the gap between different modalities, e.g. video captioning [65], video retrieval [20] and dialog system [36]. 2. To aggregate convolutional features for down-streaming tasks, e.g. object detection [5, 11], pose estimation [61], semantic segmentation [15] and action recognition [21]. 3. To perform feature learning on raw pixels, e.g. most recently image classification [14, 47].

Action recognition with self-attention on convolution features [21] is proved successful, however, convolution also generates local feature and gives redundant computations. Different from [21] and inspired by very recent work on applying transformer on raw pixels [14, 47], we pioneer the work on aggregating spatio-temporal feature from raw videos without relying on convolution features. Different from very recent work [41] that extract spatial feature with vision transformer on every video frames and then aggregate feature with attention, our proposed method jointly learns spatio-temporal feature with lower computational cost and higher performance. Our work differs from the concurrent work [4], we present a split attention with better performance without requiring larger video resolution nor extra long clip length. Some more recent work [2, 4, 4, 16, 40, 42] further studied the multi-scale and different attention factorization methods.

3. Video Transformer

We introduce the Video Transformer starting with the vanilla video transformer (section 3.1) which illustrates our idea of video action recognition without convolutions. We then present VidTr by first introducing separable-attention

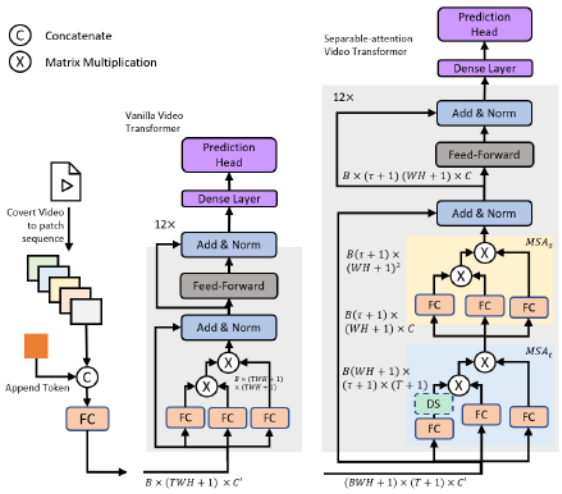


Figure 1: Spatio-temporal separable-attention video transformer (VidTr). The model takes pixels patches as input and learns the spatial temporal feature via proposed separable-attention. The green shaded block denotes the down-sample module which can be inserted into VidTr for higher efficiency. τ denotes the temporal dimension after downsampling.

(section 3.2), and then the attention pooling to drop non-representative information temporally (section 3.2).

3.1. Vanilla Video Transformer

Following previous efforts in NLP [13] and image classification [14], we adopt the transformer [52] encoder structure for action recognition that operates on raw pixels. Given a video clip $V \in \mathbb{R}^{C \times T \times W \times H}$, where T denotes the clip length, W and H denote the video frame width and height, and C denotes the number of channel, we first convert V to a sequence of $s \times s$ spatial patches, and apply a linear embedding to each patch, namely $S \in \mathbb{R}^{T\frac{H}{s}\frac{W}{s} \times C'}$, where C' is the channel dimension after the linear embedding. We add a 1D learnable positional embedding [13, 14] to S and following previous work [13, 14], append a class token as well, whose purpose is to aggregate features from the whole sequence for classification. This results in $S' \in \mathbb{R}^{(\frac{TWH}{s^2}+1) \times C'}$, where $S'_0 \in \mathbb{R}^{1 \times C'}$ is the attached class token. S' is feed into our transformer encoder structure detailed next.

As Figure 1 middle shows, we expand the previous successful ViT transformer architecture for 3D feature learning. Specifically, we stack 12 encoder layers, with each encoder layer consisting of an 8-head self-attention layer and two dense layers with 768 and 3072 hidden units. Different from transformers for 2D images, each attention layer learns a spatio-temporal affinity map $Attn \in \mathbb{R}^{(\frac{TWH}{s^2}+1) \times (\frac{TWH}{s^2}+1)}$.

3.2. VidTr

In Table 2 we show that this simple formulation is capable of learning 3D motion features on a sequence of local patches. However, as explained in [3], the affinity attention matrix $Attn \in \mathbb{R}^{(\frac{TWH}{s^2}+1) \times (\frac{TWH}{s^2}+1)}$ needs to be stored in memory for back propagating, and thus the memory consumption is quadratically related to the sequence length. We can see that the vanilla video transformer increases memory usage for the affinity map from $\mathbb{O}(W^2H^2)$ to $\mathbb{O}(T^2W^2H^2)$, leading to $T^2\times$ memory usage for training, which makes it impractical on most available GPU devices. We now address this inefficiency with a separable attention architecture.

3.2.1 Separable-Attention

To address these memory constraints, we introduce a multi-head separable-attention (MSA) by decoupling the 3D self-attention to a spatial attention MSA_s and a temporal attention MSA_t (Figure 1):

k_t. Following $\mathrm{MSA_s}$, we apply a similar 1D sequential self-attention $\mathrm{MSA_s}$ on spatial dimension:

S model to converge, the training process also scales well with fewer GPUs (e.g. 8 GPUs for 4 days). During **inference** we adopted the commonly used 30-crop evaluation for VidTr and compact VidTr, with 10 uniformly sampled temporal segments and 3 uniformly sampled spatial crop on each temporal segment [55]. It is worth mentioning that we can further boost the inference speed of compact VidTr by adopting a single pass inference mechanise, this is because the attention mechanism captures global information more effectively than 3D convolution. We do this by training a model with frames sampled in TSN [54] style, and uniformly sampling N frames in inference (details in supplemental materials).

4. Experimental Results

4.1. Datasets

We evaluate our method on six of the most widely used datasets. **Kinetics 400 [8] and Kinetics 700 [6]** consists of approximately 240K/650K training videos and 20K/35K validation videos trimmed to 10 seconds from 400/700 human action categories. We report top-1 and top-5 classification accuracy on the validation sets. **Something-Something V2 [23]** dataset consists of 174 actions and contains 168.9K training videos and 24.7K evaluation videos. We report top-1 accuracy following previous works [37] evaluation setup. **Charades [45]** has 9.8k training videos and 1.8k validation videos spanning about 30 seconds on average. Charades contains 157 multi-label classes with longer activities, performance is measured in mean Average Precision (mAP). **UCF-101 [46] and HMDB-51 [29]** are two smaller datasets. UCF-101 contains 13320 videos with an average length of 180 frames per video and 101 action categories. The HMDB-51 has 6,766 videos and 51 action categories. We report the top-1 classification on the validation videos based on split 1 for both dataset.

4.2. Kinetics 400 Results

4.2.1 Comparison To SOTA

We report results on the validation set of Kinetics 400 in Table 2, including the top-1 and top-5 accuracy, GFLOPs (Giga Floating-Point Operations) and latency (ms) required to compute results on one view.

As shown in Table 2, the VidTr achieved the SOTA performance compared to previous I3D based SOTA architectures with lower GFLOPs and latency. The VidTr significantly outperform previous SOTA methods at roughly same computational budget, e.g. at 200 GFLOPs, the VidTr-M outperform I3D50 by 3.6%, NL50 by 2.1%,and TPN50 by 0.9%. At similar accuracy levels, VidTr is significantly more computationally efficient than other works, e.g. at 78% top-1 accuracy, the VidTr-S has 6× fewer FLOPs than NL-101, 2× fewer FLOPs than TPN and 12% fewer FLOPs than Slowfast-101. We also see that our VidTr outperforms I3D based networks at higher sample rate (e.g. $s = 8$, TPN achieved 76.1% top-1 accuracy), this denotes, the global attention learns temporal information more effectively than 3D convolutions. X3D-XXL from architecture search is the only network that outperforms our VidTr. We plan to use architecture search techniques for attention based architecture in future work.

Model	Input	GFLOPs	Lat.	top-1	top-5
I3D50 [60]	32 × 2	167	74.4	75.0	92.2
I3D101 [60]	32 × 2	342	118.3	77.4	92.7
NL50 [55]	32 × 2	282	53.3	76.5	92.6
NL101 [55]	32 × 2	544	134.1	77.7	93.3
TEA50 [34]	16 × 2	70	-	76.1	92.5
TEINet [39]	16 × 2	66	49.5	76.2	92.5
CIDC [32]	32 × 2	101	82.3	75.5	92.1
SF50 8×8 [19]	(32+8)×2	66	49.3	77.0	92.6
SF101 8×8 [19]	(32+8)×2	106	71.9	77.5	92.3
SF101 16×8 [19]	(64+16)×2	213	124.3	78.9	93.5
TPN50 [60]	32 × 2	199	89.3	77.7	93.3
TPN101 [60]	32 × 2	374	133.4	78.9	93.9
CorrNet50 [53]	32 × 2	115	-	77.2	N/A
CorrNet101 [53]	32 × 2	187	-	78.5	N/A
X3D-XXL [18]	16 × 5	196	-	80.4	94.6
Vanilla-Tr	8 × 8	89	32.8	77.5	93.2
VidTr-S	8 × 8	89	36.2	77.7	93.3
VidTr-M	16 × 4	179	61.1	78.6	93.5
VidTr-L	32 × 2	351	110.2	79.1	93.9
En-I3D-50-101	32 × 2	509	192.7	77.7	93.2
En-I3D-TPN-101	32 × 2	541	207.8	79.1	94.0
En-VidTr-S	8 × 8	130	73.2	79.4	94.0
En-VidTr-M	16 × 4	220	98.1	79.7	94.2
En-VidTr-L	32 × 2	392	147.2	80.5	94.6

Table 2: Results on Kinetics-400 dataset. We report top-1 accuracy(%) on the validation set. The 'Input' column indicates what frames of the 64 frame clip are actually sent to the network. $n \times s$ input indicates we feed n frames to the network sampled every s frames. Lat. stands for the latency on single crop.

4.2.2 Compact VidTr

We evaluate the effectiveness of our compact VidTr with the proposed temporal down-sampling method (Table 1). The results (Table 3) show that the proposed down-sampling strategy removes roughly 56% of the computation required by VidTr with only 2% performance drop in accuracy. The compact VidTr complete the VidTr family from small models (only 39GFLOPs) to high performance models (up to 79.1% accuracy). Compared with previous SOTA compact models [34, 39], our compact VidTr achieves better or similar performance with lower FLOPs and latency, including: TEA (+0.6% with 16% fewer FLOPs) and TEINet (+0.5% with 11% fewer FLOPs).

Model	Input	Res.	GFLOPs	Latency(ms)	top-1
TSM [37]	$8fTSN$	256	69	29	74.7
TEA [34]	16×4	256	70	-	76.1
3DEffi-B4 [18]	16×5	224	7	-	72.4
TEINet [39]	16×4	256	33	36	74.9
X3D-M [18]	16×5	224	5	40.9	74.6
X3D-L [18]	16×5	312	19	59.4	76.8
C-VidTr-S	8×8	224	39	17.5	75.7
C-VidTr-M	16×4	224	59	26.1	76.7

Table 3: Comparison of VidTr to other fast networks. We present the number of views used for evaluation and FLOPs required for each view. The latency denotes the total time required to get the reported top-1 score.[1]

Top 5 (+)	Acc. gain
making a cake	+26.0%
catching fish	+21.2%
catching baseball	+20.8%
stretching arm	+19.1%
spraying	+ 18.0 %

(a) Top 5 classes that VidTr works better than I3D.

Top 5 (-)	Acc. gain
shaking head	-21.7%
dunking basketball	-20.8%
lunge	-19.9%
playing guitar	-19.9%
tap dancing	-16.3%

(b) Top 5 classes that I3D works better than VidTr.

Table 4: Quantitative analysis on Kinetics-400 dataset. The performance gain is defined as the disparity of the top-1 accuracy between VidTr network and that of I3D.

4.2.3 Error and Ensemble Analysis

We compare the errors made by VidTr-S and the I3D50 network to better understand the local networks' (I3D) and global networks' (VidTr) behavior. We provide the top-5 activities that our VidTr-S gain most significant improvement over the I3D50. We find that our VidTr-S outperformed the I3D on the activities that requires long-term video contexts to be recognized. For example, our VidTr-S outperformed the I3D50 on "making a cake" by 26% in accuracy. The I3D50 overfits to "cakes" and often recognize making a cake as eating a cake. We also analyze the top-5 activities where I3D does better than our VidTr-S (Table 4). Our VidTr-S performs poorly on the activities that need to capture fast and local motions. For example, our VidTr-S performs 21% worse in accuracy on "shaking head".

Inspired by the findings in our error analysis, we ensembled our VidTr with a light weight I3D50 network by averaging the output values between the two networks. The results (Table 2) show that the the I3D model and transformer model complements each other and the ensemble model roughly lead to 2% performance improvement on Kinetics 400 with limited additional FLOPs (37G). The performance gained by ensembling the VidTr with I3D is significantly better than the improvement by combine two 3D networks (Table 2).

[1]we measure latency of X3D using the authors' code and fast depth convolution patch: https://github.com/facebookresearch/SlowFast/blob/master/projects/x3d/README.md, which only has models for X3D-M and X3D-L and not the XL and XXL variants

4.2.4 Ablations

Model	FP.	top-1
Cubic (4×16^2)	23G	73.1
Cubic (2×16^2)	45G	75.5
Square (1×16^2)	89G	77.7
Square (1×32^2)	21G	71.2

(a) Comparison between different patching strategies.

Model	Mem.	top-1
WH	2.1GB	74.7
WHT	7.6GB	77.5
WH + T	2.3GB	77.7
W + H + T.	1.5GB	72.3

(b) Comparison between different factorization.

Init. from	FP.	top-1
T2T [62]	34G	76.3
ViT-B [14]	89G	77.7
ViT-L [14]	358	77.5

(c) Comparison between different backbones.

Configurations	top-1	top-5
Temp. Avg. Pool.	74.9	91.6
1D Conv. [62]	75.4	92.3
STD Pool.	75.7	92.2

(d) Comparison between different down-sample methods.

Layer	τ	FP.	top-1
[0, 2]	[4, 2]	26G	72.9
[1, 3]	[4, 2]	32G	74.9
[2, 4]	[4, 2]	47G	74.9
[6, 8]	[4, 2]	60G	75.3

(e) Compact VidTr down-sampling twice at layer k and $k+2$.

Layer	τ	FP.	top-1
[1, 2]	[4, 2]	30G	73.9
[1, 3]	[4, 2]	32G	74.9
[1, 4]	[4, 2]	33G	75.0
[1, 5]	[4, 2]	34G	75.2

(f) Compact VidTr down-sampling twice starting from layer 1 and skipping different number of layers.

Table 5: Ablation studies on Kinetics 400 dataset. We use an VidTr-S backbone with 8 frames input for (a,b) and C-VidTr-S for (c,d). The evaluation is performed on 30 views with 8 frame input unless specified. FP. stands for FLOPs.

We perform all ablation experiments with our VidTr-S model on Kinetics 400. We used $8 \times 224 \times 224$ input with a frame sample rate of 8, and 30-view evaluation.

Patching strategies: We first compare the cubic patch (4×16^2), where the video is represented as a sequence of spatio-temporal patches, with the square patch (1×16^2), where the video is represented as a sequence of spatial patches. Our results (Table 5a) show that the model using cubic patches with longer temporal size has fewer FLOPs but results in significant performance drop (73.1 vs. 75.5). The model using square patches significantly outperform all cubic patch based models, likely because the linear embedding is not enough to represent the shot-term temporal association in the cubic. We further compared the performance of using different patch sizes (1×16^2 vs. 1×32^2), using 32^2 patches lead to $4\times$ decreasing of the sequence length, which decreases memory consumption of the affinity matrices by $16\times$, however, using 16^2 patches significantly outperform the model using 32^2 patches (77.7 vs. 71.2). We did not evaluate the model using smaller patching sizes (e.g., 8×8) because of the high memory consumption.

Attention Factorization: We compare different factorization for attention design, including spatial modeling only (WH), jointly spatio-temporal modeling module (WHT,

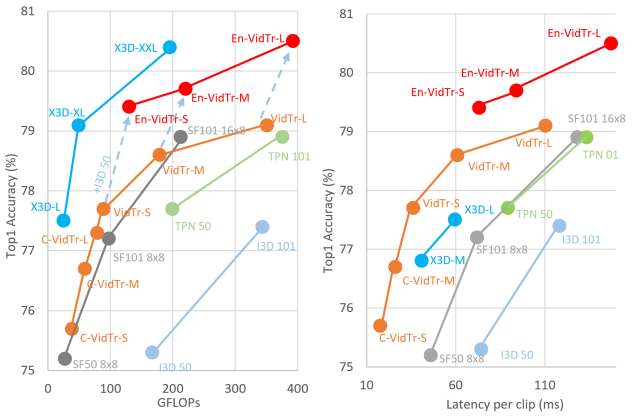


Figure 2: The comparison between different models on accuracy, FLOPs and latency.

vanilla-Tr), spatio-temporal separable-attention (WH + T, VidTr), and axial separable-attention (W + H + T). We first evaluate an spatio-only transformer. We average the class token for each input frame for our final output. Our results (Table 5b) show that the spatio-only transformer requires less memory but has worse performance compare with spatio-temporal attention models. This shows that temporal modeling is critical for attention based architectures. The joint spatio-temporal transformer significantly outperforms the spatio-only transformer but requires a restrictive amount of memory (T^2 times for the affinity matrices). Our VidTr using spatio-temporal separable-attention requires 3.3× less memory with no accuracy drop. We further evaluate the axial separable-attention (W + H + T), which requires the least memory. The results (Table 5b) show that the axial separable-attention has a significant performance drop likely due to breaking the X and Y spatial dimensions.

Sequence down-sampling comparison: We compare different down-sampling strategy including temporal average pooling, 1D temporal convolution and the proposed STD-based topK pooling method. The results (Table 5d) show that our proposed STD-based down-sampling method outperformed the temporal average pooling and the convolution-based down-sampling strategies that uniformly aggregate information over time.

Backbone generalization: We evaluate our VidTr initialized with different models, including T2T [62], ViT-B, and ViT-L. The results on Table 5c show that our VidTr achieves reasonable performance across all backbones. The VidTr using T2T as the backbone has the lowest FLOPs but also the lowest accuracy. The Vit-L-based VidTr achieve similar performance with the Vit-B-based VidTr even with 3× FLOPs. As showed in previous work [14], transformer-based network are more likely to over-fit and Kinetics-400 is relatively small for Vit-L-based VidTr.

Where to down-sample: Finally we study where to perform temporal down-sampling. We perform temporal down-sampling at different layers (Table 5e). Our results (Table 5e) show that starting to perform down-sampling after the first encoder layer has the best trade-off between the performance and FLOPs. Starting to perform down-sampling at very beginning leads to the fewest FLOPs but has a significant performance drop (72.9 vs. 74.9). Performing down-sampling later only has slight performance improvement but requires higher FLOPs. We then analyze how many layers to skip between two down-sample layers. Based on the results in Table 5f, skipping one layer between two down-sample operations has the best trade-off. Performing down-sampling on consecutive layers (0 skip layers) has lowest FLOPs but the performance decreases (73.9 vs. 74.9). Skipping more layers did not show significant performance improvement but does have higher FLOPs.

Model	Input	K700	Chad	SS	UCF	HM
I3D [7]	32×2	58.7	32.9	50.0	95.1	74.3
TSM [37]	8(TSN)	-	-	59.3	94.5	70.7
I3D101 [59]	32 × 4	40.3	-	-	-	
CSN152 [49]	32 × 2	70.1	-	-	-	-
TEINet[39]	16 (TSN)	-	-	62.1	96.7	73.3
SF101 [19]	64×2	70.2	-	60.9	-	-
SF101-NL [19]	64×2	70.6	45.2	-	-	-
X3D-XL [18]	16 × 5	-	47.1	-	-	-
VidTr-M	16 × 4	69.5	-	61.9	96.6	74.4
VidTr-L	32 × 2	70.2	43.5	63.0	**96.7**	**74.4**
En-VidTr-L	32 × 2	70.8	**47.3**	-	-	-

Table 6: Results on Kinetics-700 dataset (K700), Charades dataset (Chad), something-something-V2 dataset (SS), UCF-101 and HMDB (HM) dataset. The evaluation metrics are mean average precision (mAP) in percentage for Charades (32×4 input is used), top-1 accuracy for Kinetics 700, something-something-V2 (TSN styled dataloader is used), UCF and HMDB.

4.2.5 Run-time Analysis

We further analyzed the trade-off between latency, FLOPs and accuracy. We note that the VidTr achieved the best balance between these factors (Figure 2). The VidTr-S achieve similar performance but significantly fewer FLOPs compare with I3D101-NL (5× fewer FLOPs), Slowfast101 8 × 8 (12% fewer FLOPs), TPN101 (2× fewer FLOPs), and CorrNet50 (20× fewer FLOPs). Note that the X3D has very low FLOPs but high latency due to the use of depth convolution. Our experiments show that the X3D-L has about 3.6× higher latency comparing with VidTr-S (Figure 2).

4.3. More Results

Kinetics-700 Results: Our experiments show a consistent performance trend on Kinetics 700 (Table 6). The VidTr-S significantly outperformed the baseline I3D model (+9%), the VidTr-M achieved the performance comparable

to Slowfast101 8×8 and the VidTr-L is comparable to previous SOTA slowfast101-nonlocal. There is a small performance gap between our model and Slowfast-NL [19], because Slowfast is pre-trained on both Kinetics 400 and 600 while we only pre-trained on Kinetics 400. Previous findings that VidTr and I3D are being complementary is consistent on Kinetics 700, ensemble VidTr-L with I3D leads to +0.6% performance boost.

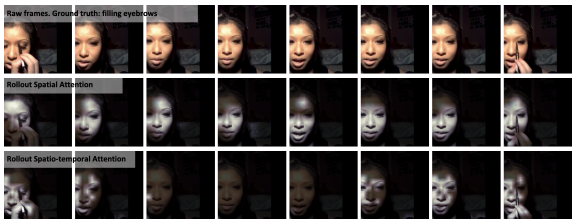
Charades Results: We compare our VidTr with previous SOTA models on Charades. Our VidTr-L outperformed previous SOTA methods LFB and NUTA101, and achieved the performance comparable to Slowfast101-NL (Table 6). The results on Charades demonstrates that our VidTr generalizes well to multi-label activity datasets. Our VidTr performs worse than the current SOTA networks (X3D-XL) on Charades likely due to overfitting. As discussed in previous work [14], the transformer-based networks overfit easier than convolution-based models, and Charades is relatively small. We observed a similar finding with our ensemble, ensembling our VidTr with a I3D network (40.3 mAP) achieved SOTA performance.

Something-something V2 Results: We observe that the VidTr does not work well on the something-something dataset (Table 6), likely because pure transformer based approaches do not model local motion as well as convolutions. This aligns with our observation in our error analysis. Further improving local motion modeling ability is an area of future work.

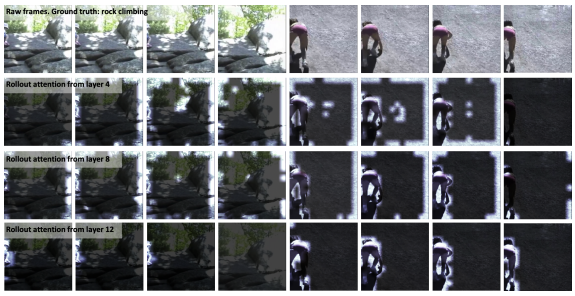
UCF and HMDB Results: Finally we train our VidTr on two small dataset UCF-101 and HMDB-51 to test if VidTr generalizes to smaller datasets. The VidTr achieved SOTA comparable performance with 6 epochs of training (96.6% on UCF and 74.4% on HMDB), showing that the model generalize well on small dataset (Table 6).

5. Visualization and Understanding VidTr

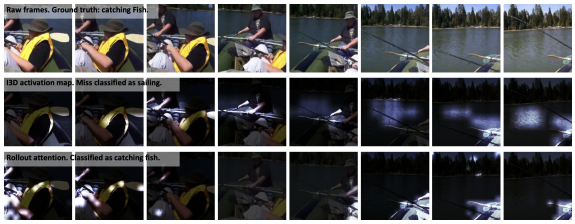
We first visualized the VidTr's separable-attention with attention roll-out method [1] (Figure 3a). We find that the spatial attention is able to focus on informative regions and temporal attention is able to skip the duplicated/non-representative information temporally. We then visualized the attention at 4th, 8th and 12th layer of VidTr (Figure 3b), we found the spatial attention is stronger on deeper layers. The attention does not capture meaningful temporal instances at early stages because the temporal feature relies on the spatial information to determine informative temporal instances. Finally we compared the I3D activation map and rollout attention from VidTr (Figure 3c). The I3D misclassified the catching fish as sailing, as the I3D attention focused on the people sitting behind and water. The VidTr is able to make the correct prediction and the attention showed that the VidTr is able to focus on the action related regions across time.



(a) The spatial and temporal attention in Vidtr. The attention is able to focus on the informative frames and regions.



(b) The rollout attentions from different layers of VidTr.



(c) Comparison of I3D activations and VidTr attentions.

Figure 3: Visualization of spatial and temporal attention of VidTr and comparison with I3D activation.

6. Conclusion

In this paper, we present video transformer with separable-attention, an novel stacked attention based architecture for video action recognition. Our experimental results show that the proposed VidTr achieves state-of-the-art or comparable performance on five public action recognition datasets. The experiments and error analysis show that the VidTr is especially good at modeling the actions that requires long-term reasoning. Further combining the advantage of VidTr and convolution for better local-global action modeling [38, 57] and adopt self-supervised training [9] on large-scaled data will be our future work.

Acknowledgements. We thank NSF grant IIS-1763827 for supporting Yanyi Zhang to study as a Ph.D. student at Rutgers University.

References

[1] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual*

Meeting of the Association for Computational Linguistics, pages 4190–4197, 2020. 2, 8

[2] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691*, 2021. 2

[3] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. 3

[4] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021. 2

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 2

[6] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019. 5

[7] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 2, 7

[8] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733, July 2017. 5

[9] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised visual transformers. *arXiv preprint arXiv:2104.02057*, 2021. 8

[10] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. Multi-fiber networks for video recognition. In *Proceedings of the european conference on computer vision (ECCV)*, pages 352–367, 2018. 2

[11] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. *arXiv preprint arXiv:2011.09094*, 2020. 2

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019. 2, 3

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2, 3, 4, 6, 7, 8

[15] Brendan Duke, Abdalla Ahmed, Christian Wolf, Parham Aarabi, and Graham W Taylor. Sstvos: Sparse spatiotemporal transformers for video object segmentation. *arXiv preprint arXiv:2101.08833*, 2021. 2

[16] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. *arXiv preprint arXiv:2104.11227*, 2021. 2

[17] Quanfu Fan, Chun-Fu Chen, Hilde Kuehne, Marco Pistoia, and David Cox. More is less: Learning efficient video representations by big-little network and depthwise temporal aggregation. *arXiv preprint arXiv:1912.00869*, 2019. 2

[18] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 203–213, 2020. 2, 5, 6, 7

[19] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6202–6211, 2019. 1, 2, 5, 7, 8

[20] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *European Conference on Computer Vision (ECCV)*, volume 5. Springer, 2020. 2

[21] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253, 2019. 2

[22] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. ActionVLAD: Learning Spatio-Temporal Aggregation for Action Classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[23] 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*, volume 1, page 3, 2017. 5

[24] 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. 2

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

[26] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012. 2

[27] Boyuan Jiang, MengMeng Wang, Weihao Gan, Wei Wu, and Junjie Yan. STM: SpatioTemporal and Motion Encoding for Action Recognition. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[28] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 2

[29] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563. IEEE, 2011. 5

[30] Qing Li, Zhaofan Qiu, Ting Yao, Tao Mei, Yong Rui, and

Jiebo Luo. Action recognition by learning deep multi-granular spatio-temporal video representation. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, pages 159–166, 2016. 2

[31] Xinyu Li, Chunhui Liu, Bing Shuai, Yi Zhu, Hao Chen, and Joseph Tighe. Nuta: Non-uniform temporal aggregation for action recognition. *arXiv preprint arXiv:2012.08041*, 2020. 1, 2, 4

[32] Xinyu Li, Bing Shuai, and Joseph Tighe. Directional temporal modeling for action recognition. In *European Conference on Computer Vision*, pages 275–291. Springer, 2020. 1, 2, 5

[33] Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. TEA: Temporal Excitation and Aggregation for Action Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[34] Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. Tea: Temporal excitation and aggregation for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 909–918, 2020. 5, 6

[35] Yingwei Li, Weixin Li, Vijay Mahadevan, and Nuno Vasconcelos. VLAD3: Encoding Dynamics of Deep Features for Action Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[36] Zekang Li, Zongjia Li, Jinchao Zhang, Yang Feng, Cheng Niu, and Jie Zhou. Bridging text and video: A universal multimodal transformer for video-audio scene-aware dialog. *arXiv preprint arXiv:2002.00163*, 2020. 2

[37] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7083–7093, 2019. 2, 5, 6, 7

[38] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021. 8

[39] Zhaoyang Liu, Donghao Luo, Yabiao Wang, Limin Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Tong Lu. TEINet: Towards an Efficient Architecture for Video Recognition. In *The Conference on Artificial Intelligence (AAAI)*, 2020. 2, 5, 6, 7

[40] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 2

[41] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselmann. Video transformer network. *arXiv preprint arXiv:2102.00719*, 2021. 2

[42] Mandela Patrick, Dylan Campbell, Yuki M Asano, Ishan Misra Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, Jo Henriques, et al. Keeping your eye on the ball: Trajectory attention in video transformers. *arXiv preprint arXiv:2106.05392*, 2021. 2

[43] AJ Piergiovanni, Anelia Angelova, and Michael S Ryoo. Tiny video networks. *arXiv preprint arXiv:1910.06961*, 2019. 2

[44] Hao Shao, Shengju Qian, and Yu Liu. Temporal interlacing network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11966–11973, 2020. 2

[45] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ivan Laptev, Ali Farhadi, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. *ArXiv e-prints*, 2016. 5

[46] Khurram Soomro, Amir Roshan Zamir, and M Shah. A dataset of 101 human action classes from videos in the wild. *Center for Research in Computer Vision*, 2012. 5

[47] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020. 1, 2

[48] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 2

[49] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5552–5561, 2019. 2, 7

[50] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 1

[51] Amin Ullah, Jamil Ahmad, Khan Muhammad, Muhammad Sajjad, and Sung Wook Baik. Action recognition in video sequences using deep bi-directional lstm with cnn features. *IEEE access*, 6:1155–1166, 2017. 2

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 1, 2, 3

[53] Heng Wang, Du Tran, Lorenzo Torresani, and Matt Feiszli. Video modeling with correlation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 352–361, 2020. 5

[54] 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 *The European Conference on Computer Vision (ECCV)*, 2016. 2, 5

[55] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018. 1, 2, 5

[56] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 284–293, 2019. 1

[57] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. *arXiv preprint arXiv:2103.15808*, 2021. 8

[58] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500,

2017. 2

[59] Ceyuan Yang, Yinghao Xu, Jianping Shi, Bo Dai, and Bolei Zhou. Temporal Pyramid Network for Action Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 7

[60] Ceyuan Yang, Yinghao Xu, Jianping Shi, Bo Dai, and Bolei Zhou. Temporal pyramid network for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 591–600, 2020. 5

[61] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Towards explainable human pose estimation by transformer. *arXiv preprint arXiv:2012.14214*, 2020. 2

[62] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *arXiv preprint arXiv:2101.11986*, 2021. 6, 7

[63] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015. 2

[64] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal Relational Reasoning in Videos. In *The European Conference on Computer Vision (ECCV)*, 2018. 2

[65] Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748, 2018. 2