Stand-Alone Inter-Frame Attention in Video Models

Fuchen Long[†], Zhaofan Qiu[†], Yingwei Pan[†], Ting Yao[†], Jiebo Luo[§] and Tao Mei[†]
[†]JD Explore Academy, Beijing, China
[§]University of Rochester, Rochester, NY USA
{longfc.ustc, zhaofanqiu, panyw.ustc, tingyao.ustc}@gmail.com
jluo@cs.rochester.edu; tmei@jd.com

Abstract

Motion, as the uniqueness of a video, has been critical to the development of video understanding models. Modern deep learning models leverage motion by either executing spatio-temporal 3D convolutions, factorizing 3D convolutions into spatial and temporal convolutions separately, or computing self-attention along temporal dimension. The implicit assumption behind such successes is that the feature maps across consecutive frames can be nicely aggregated. Nevertheless, the assumption may not always hold especially for the regions with large deformation. In this paper, we present a new recipe of inter-frame attention block, namely Stand-alone Inter-Frame Attention (SIFA), that novelly delves into the deformation across frames to estimate local self-attention on each spatial location. Technically, SIFA remoulds the deformable design via re-scaling the offset predictions by the difference between two frames. Taking each spatial location in the current frame as the query, the locally deformable neighbors in the next frame are regarded as the keys/values. Then, SIFA measures the similarity between query and keys as stand-alone attention to weighted average the values for temporal aggregation. We further plug SIFA block into ConvNets and Vision Transformer, respectively, to devise SIFA-Net and SIFA-Transformer. Extensive experiments conducted on four video datasets demonstrate the superiority of SIFA-Net and SIFA-Transformer as stronger backbones. More remarkably, SIFA-Transformer achieves an accuracy of 83.1% on Kinetics-400 dataset. Source code is available at https://github.com/FuchenUSTC/SIFA.

1. Introduction

Video is an electronic representation of moving visual images and naturally forms the motion, which signifies a continuous change in position of objects or persons with time. Modeling such temporal dynamics is essential to the extension from understanding still images to videos. The recent advances generally suggest to leverage motion along two directions. One involves utilization of temporal convolutions by being integrated into space-time 3D convolutions [18, 50] or explicitly co-working with spatial convolutions [3, 52, 62]. The other measures self-attention of each location over the temporal neighbors at the same spatial position across frames. Figure 1(a) and (b) conceptually depict the implementation of temporal convolution and self-attention along temporal dimension, respectively. The underlying spirit behind these operations originates from the foundation that the feature maps across frames should be well aligned. This assumption nevertheless may not always be valid in practice. Taking the three consecutive frames in Figure 1 as an example, the same positions across frames highlighted in the circles correspond to different objects (person and track in the case) due to the motion of the athlete in pole vault. As such, performing temporal convolution or computing attention over these positions might be suboptimal for temporal feature aggregation.

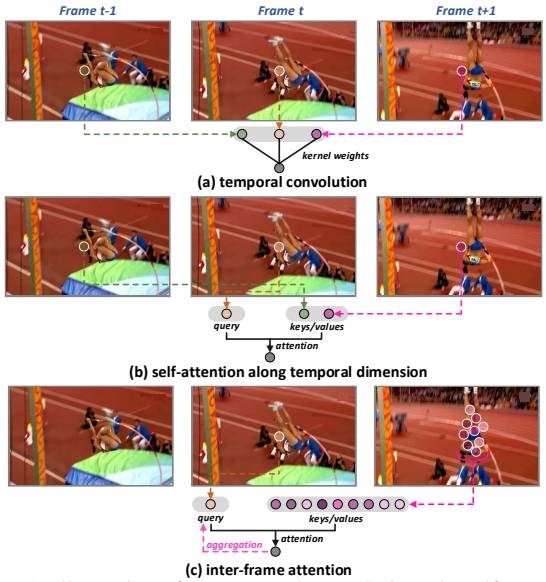

Figure 1. Illustration of (a) temporal convolution, (b) self-attention along temporal dimension, and (c) our inter-frame attention.

To alleviate this issue, we propose to take the changes

in video content caused by motion into account to enhance the alignment of feature maps across frames and eventually improve temporal aggregation. Technically, we develop inter-frame attention as shown in Figure 1(c) to characterize richer inter-frame correlation within a local neighboring region rather than only the same spatial location in consecutive frames. By doing so, inter-frame attention, on one hand, is beneficial more with large receptive fields, and on the other, manifests the emphasis of each location in the region to better achieve feature alignment. In an effort to nicely support the regions with large deformation, we further capitalize on the deformable design and estimate the offset to each spatial location. Moreover, we uniquely exploit the motion cues across frames to act as motion supervisory signal and re-scale the deformable feature re-sampling.

By delving into the deformation across frames to infer temporal attention within locally deformable region for temporal modeling, we present a novel Stand-alone Inter-Frame Attention (SIFA) block in video models. Specifically, we take each spatial location in the current frame as the query, and its temporal neighbors within the local region of the next frame are treated as keys/values accordingly to trigger the inter-frame attention learning. Note that in view of the irregular geometric transformations of objects, we sample the keys/values of temporal neighbors in a spatial deformation, which is learnt with additional guidance of the motion cues across frames. After that, SIFA block regards the estimated inter-frame attention of each temporal neighbor as its temporal correlation against query. Finally, we aggregate all temporal neighbors of nearby frames with inter-frame attention weights to further strengthen the query feature in current frame via temporal aggregation.

The SIFA block can be viewed as a stand-alone attention primitive for temporal modeling, and is readily pluggable to any 2D CNN or Vision Transformer backbones for video representation learning. By directly inserting SIFA block in ResNet [17] and Swin Transformer [32], we construct two new video backbones, named as SIFA-Net and SIFA-Transformer, respectively. Through extensive experiments on a series of action recognition benchmarks, we demonstrate that our SIFA-Net and SIFA-Transformer outperform several state-of-the-art video backbones.

2. Related Work

We categorize existing research for video representation learning into hand-crafted and deep model based methods.

Hand-crafted Representation. The early hand-crafted video feature techniques first detect spatio-temporal interest points and then describe them with local representations, such as STIP [23], Histogram of Gradient and Histogram of Optical Flow [24], 3D Histogram of Gradient [21], and SIFT-3D [45]. Besides, Wang *et al.* design the dense trajectory feature [54] that samples dense local patches from each frame at various scales and tracks them in an optical flow field to convey motion cues in temporal domain. Nevertheless, these hand-crafted features are not optimized, thereby hardly to be generalized across different video tasks.

Deep Learning based Representation. This direction first emerges by directly applying 2D CNN over video frames for video representation learning. For instance, Karpathy *et al.* stack frame-level CNN features in a fixed size of window and then leverage spatial convolution to learn video representation [20]. Later in [47], the two-stream model is devised by utilizing two 2D CNN separately on visual frames and stacked optical flows. This technique is further extended by exploring the convolution fusion [13], temporal segment networks [12, 57, 63] and convolutional encoding [6]. To capture the long-term temporal dependency which is commonly ignored in some two-stream networks, LSTM-based methods [40, 48] are designed to model long-range temporal dynamics in videos.

The aforementioned approaches only treat video as a sequence of frames or optical flows, while leaving the pixel-level temporal evolution across consecutive frames unexploited. 3D CNN based video feature [50] is thus proposed to alleviate this issue by employing 3D convolutional kernels over short clips. Furthermore, the subsequent works [3, 41, 43, 62, 64] show that factorizing 3D convolution into 2D spatial convolution and 1D temporal convolution leads to better results and presents good generalization ability on localization task [25, 26, 35–37]. Most recently, inspired by the impressive performances of applying self-attention from NLP field [53] into image feature learning [7, 29, 32], TimeSformer [2] performs self-attention along the temporal dimension and designs five variants for temporal modeling. Nevertheless, these methods equipped with temporal convolution or temporal self-attention still suffer from the robustness problem due to object deformation across frames.

Our work belongs to deep model based techniques that model temporal dynamics through self-attention. Unlike TimeSformer [2] that measures self-attention of each location solely over its temporal neighbors at the same spatial location, SIFA mechanism performs inter-frame attention within a local neighboring region with large receptive fields. Moreover, SIFA block goes beyond the measure of inter-frame self-attention within regular local region, and capitalizes on locally deformable neighbors to tackle the irregular object deformation issue in temporal modeling.

3. Our Approach

We introduce a new Stand-alone Inter-Frame Attention (SIFA) for temporal modeling. SIFA exploits the temporal correlation within local region across consecutive frames, aiming to strengthen per-frame feature by aggregating its local neighbors in nearby frames via attention. Next, a novel stand-alone block in video models, i.e., SIFA block,

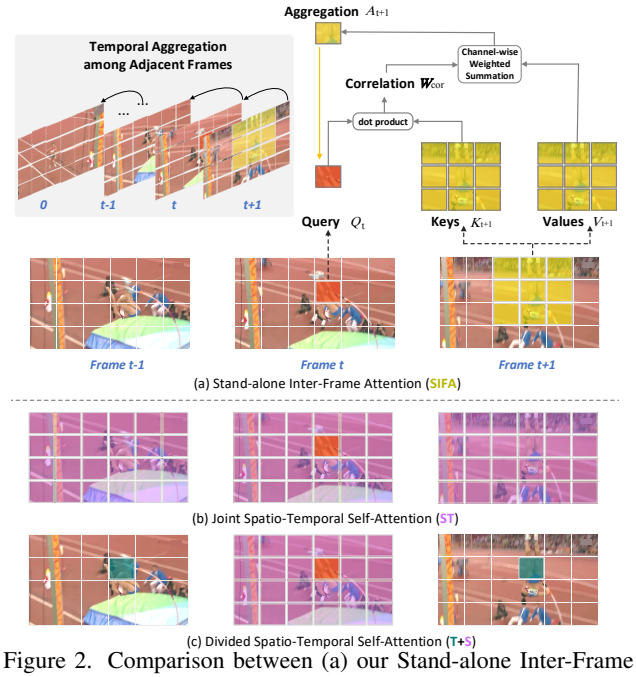

Figure 2. Comparison between (a) our Stand-alone Inter-Frame Attention (**SIFA**) and two kinds of previous spatio-temporal attention, i.e., (b) joint spatio-temporal self-attention (**ST**) and (c) divided spatio-temporal self-attention (**T+S**). By visualizing each video clip as a sequence of frame-level patches, we denote in red the query patch and show its spatio-temporal neighbors in non-red colors for each attention mechanism. The patches without color are excluded for attention learning. Different from ST that employs attention over all frames holistically, T+S separately performs attention along the divided space and time dimensions.

is designed to perform such inter-frame attention over locally deformable region across frames. By plugging our SIFA block into 2D CNN (ResNet [17]) and Vision Transformer (Swin Transformer [32]), we further elaborate two video backbones, i.e., SIFA-Net and SIFA-Transformer.

3.1. Stand-alone Inter-Frame Attention (SIFA)

A natural way for temporal modeling in video representation learning is to use the 1D temporal convolution that conducts pixel-level feature aggregation across frames. However, this way solely captures motion clues among the same spatial locations along temporal dimension, while ignoring the inter-frame correlation at different spatial locations for temporal modeling. Inspired by the modeling of long-range dependencies via attention [53, 58], we devise a new attention mechanism tailored for temporal modeling, i.e., Stand-alone Inter-Frame Attention (SIFA), that exploits the inter-frame correlation within local region for attention learning in an efficient manner. All the temporal neighbors within local region of nearby frames are aggregated with attention to strengthen per-frame feature.

Here we introduce the detailed formulation of our SIFA, as depicted in Figure 2 (a). Technically, let F be the input 3D feature map with the size of $C \times L \times H \times W$, where C, $H \times W$, and L denotes the channel size, spatial size, and temporal length, respectively. We first reshape F into a 2D feature sequence $\{f_t\}_{t=0}^{L-1}$. Next, for t-th frame, we take its feature at the spatial location (x, y) as the query $Q_t \in \mathbb{R}^C$. Meanwhile, the features of (t+1)-th frame within the local region (size: $k \times k$ grid) centered at (x, y) are set as keys $K_{t+1} \in \mathbb{R}^{C \times \{k \times k\}}$ and values $V_{t+1} \in \mathbb{R}^{C \times \{k \times k\}}$. The correlation matrix $\mathbf{W}_{cor}$ between query Q_t and keys K_{t+1} is then calculated via dot production:

$$\mathbf{W}_{cor} = Q_t \odot K_{t+1}, \tag{1}$$

where $\odot$ denotes the matrix multiplication that measures the pairwise temporal correlation between query and its temporal neighbors (i.e., keys) within the local $k \times k$ grid.

Existing works commonly take the learnt correlation matrix $\mathbf{W}_{cor} \in \mathbb{R}^{1 \times \{k \times k\}}$ as pixel-level displacement information, and directly augment primary feature map with it to subserve flow estimation [14, 61], geometric matching [44] and motion modeling [55]. As an alternative, we capitalize on the correlation matrix as attention weights to dynamically aggregate the corresponding values within local region in nearby frame, targeting for enhancing query feature. In particular, by taking the correlation matrix $\mathbf{W}_{cor}$ as the attention weights, the values V_{t+1} within the local region are aggregated in a channel-wise manner:

$$A_{t+1} = \mathbf{W}_{cor} \odot [V_{t+1}]^T, \tag{2}$$

where A_{t+1} is the aggregated feature derived from the temporal neighbors of query, and the $[\cdot]^T$ denotes the matrix transpose. After that, we integrate the query with the aggregated feature, yielding the enhanced query feature Y_t after temporal feature aggregation:

$$Y_t = Q_t + A_{t+1}. \tag{3}$$

Accordingly, SIFA performs the inter-frame attention over each spatial location in t-th frame to mine its temporal correlation within local region of (t+1)-th frame. The feature map of each frame is thus strengthened by aggregating the features of local neighbors in the next frame via attention. In this way, we operate SIFA between every pair of adjacent frames in the input sequence. Note that for the last frame in the sequence, we conduct the inter-frame attention between this frame and itself, and enhance its feature map by itself through feature aggregation, thereby keeping the temporal length of output frame sequence as L.

Connections with Previous Spatio-temporal Attention. Here we further discuss the detailed relations and differences between our SIFA and the previous spatio-temporal attention mechanisms. [2] introduces two kinds of spatio-temporal attention (i.e., joint or divided spatio-temporal self-attention) that employ self-attention over space and time for video representation learning. Specifically, the joint spatio-temporal self-attention (i.e., ST in

Figure 2 (b)) performs self-attention over the input features/patches of all frames holistically. The divided spatio-temporal self-attention (i.e., T+S in Figure 2 (c)) separately applies the spatial attention within current frame and the temporal attention over the temporal neighbors in the same spatial location of nearby frames. Our SIFA also targets for exploring self-attention along temporal dimension for video modeling. Different from the global temporal attention over the holistic features/patches in ST, SIFA conducts the local temporal attention within local region across frames, which is computationally more efficient. Moreover, compared to S+T that only mines temporal evolution in the same spatial location of consecutive frames, SIFA captures the richer inter-frame correlation within local region for attention learning, thereby facilitating temporal modeling.

3.2. SIFA Block

Recall that our SIFA mechanism is devised to model the temporal evolution of objects within local region across consecutive frames. However, simply employing inter-frame attention over the equally-sized local region ($k \times k$ grid) inevitably ignores the irregular geometric transformations of objects in each frame, resulting in a sub-optimal solution. To alleviate this issue, we devise a SIFA block that applies inter-frame attention over the locally deformable region in nearby frames, which consists of the temporal neighbors sampled in a free-form spatial deformation.

The most typical way to operate deformable feature re-sampling is to augment the spatial sampling locations with additional offsets, that are predicted via a learnable offset estimator as in deformable ConvNets [5]. Nevertheless, this offset estimator learns to infer the 2D offset of each spatial location solely based on the input feature map itself, while leaving the inherent motion clues across consecutive frames unexploited. Instead, we propose to estimate 2D offset of each spatial location within local region based on its motion saliency map (MSM), which acts as motion supervision to guide the deformable feature re-sampling. Figure 3 shows the detailed structure of our SIFA block.

Formally, given each pair of consecutive frames (i.e., t-th frame f_t and (t+1)-th frame f_{t+1}), we first compute the temporal difference (TD) in between:

$$\Delta f = f_{t+1} - f_t. \tag{4}$$

Next, we employ a sigmoid operation over such temporal difference, leading to a normalized attention map. This attention map dynamically pinpoints the spatial locations in (t+1)-th frame that contain highly salient movements of objects. Therefore, the motion saliency map (MSM) f_m is achieved by multiplying the feature map of (t+1)-th frame f_{t+1} with the attention map:

$$f_m = sigmoid(\Delta f) * f_{t+1}. \tag{5}$$

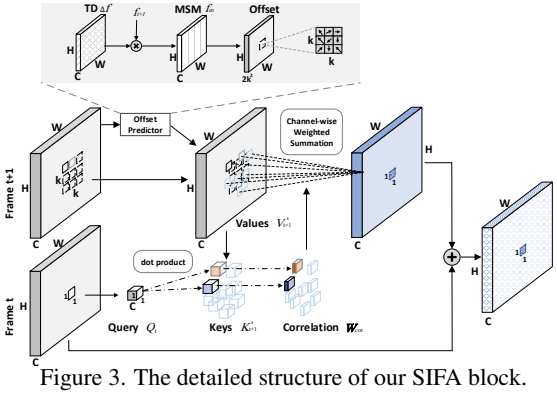

Figure 3. The detailed structure of our SIFA block.

Conditioned on the motion saliency map f_m, we utilize an offset estimator to predict the 2D offset for each spatial location within the local region ($k \times k$ grid) of (t+1)-th frame f_{t+1}. Note that the offset estimator is implemented as a 2D convolutional layer with the output channel size of $2k^2$. More specifically, let $(\Delta a, \Delta b)$ denote the estimated 2D offset of each spatial location $p = (a, b)$ within the $k \times k$ grid centered at the query location (x, y). The corresponding irregular spatial location is thus represented as $p' = (a + \Delta a, b + \Delta b)$. Following [5], we sample the feature $K'_{t+1}(p')$ at each irregular spatial location p' through bilinear interpolation:

$$K'_{t+1}(p') = \sum_{p} G(p, p') \cdot K_{t+1}(p), \tag{6}$$

where p' is the fractional spatial location and p enumerates all integral spatial locations within the local region. $K_{t+1}(p)$ denotes the primary feature at regular spatial location p, and G is bilinear interpolation kernel. After sampling all the k^2 deformable features in (t+1)-th frame f_{t+1}, we take them as the keys $K'_{t+1} \in \mathbb{R}^{C \times \{k \times k\}}$ and values $V'_{t+1} \in \mathbb{R}^{C \times \{k \times k\}}$ with regard to the query $Q_t \in \mathbb{R}^C$ in t-th frame f_t. In this way, we perform SIFA mechanism over the locally deformable region in nearby frame, and further strengthen per-frame feature by aggregating these deformable features via attention:

$$\begin{aligned} \mathbf{W}_{cor} &= Q_t \odot K'_{t+1}, \\ A_{t+1} &= \mathbf{W}_{cor} \odot [V'_{t+1}]^T, \\ Y_t &= Q_t + A_{t+1}. \end{aligned} \tag{7}$$

The enhanced feature Y_t for t-th frame is finally taken as the output of SIFA block.

3.3. 2D CNN and Vision Transformer with SIFA

Our SIFA block acts as a stand-alone primitive for temporal modeling, and is pluggable to any 2D CNN or Vision Transformer architectures. Such design naturally upgrades these vision backbones with the capacity of temporal modeling, thereby boosting video representation learning. Here we present how to integrate SIFA block into existing 2D CNN (e.g., ResNet [17]) and Vision Transformer

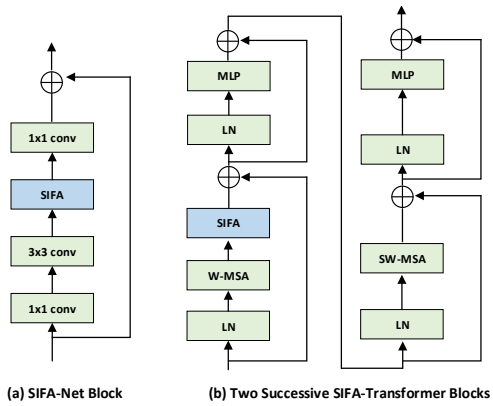

Figure 4. Basic blocks in (a) SIFA-Net and (b) SIFA-Transformer.

(e.g., Swin Transformer [32]). Figure 4 depicts the two different constructions of equipping the basic building block in ResNet/Swin Transformer with our SIFA block, namely SIFA-Net and SIFA-Transformer, respectively.

SIFA-Net. Most of existing video backbones [3, 42, 52, 62] factorize the conventional 3D convolution into 2D spatial convolution and 1D temporal convolution, and the 1D temporal convolution is commonly plugged after the spatial convolutional layers of 2D CNN for temporal modeling across frames. We follow this typical paradigm and construct SIFA-Net by inserting SIFA block after the 3×3 convolution within each residual building block in ResNet [17]. Note that we solely integrate the last three stages (i.e., res_3, res_4 and res_5) in ResNet with our SIFA block, thereby only increasing a small overhead to the computational cost. Finally, the global pooling is employed on the output feature to achieve the clip-level feature for video classification.

SIFA-Transformer. Recently, computer vision field has witnessed the rise of Transformer-style architecture with self-attention [7, 32] in powerful vision backbones. Inspired by this, we further construct the Transformer-style video backbone, named as SIFA-Transformer, by integrating the Swin Transformer [32] with our SIFA block. In particular, for every two successive Swin Transformer blocks in Swin Transformer, we directly insert the SIFA block after the MSA module with regular windowing configuration, leading to the two successive SIFA-Transformer blocks. Note that the output patch sequence of MSA module is reshaped into the sequence of feature map with the normal size ($C \times L \times H \times W$), which acts as the inputs of SIFA block. Based on the output reshaped sequence of feature map for the last block in SIFA-Transformer, we leverage the global pooling to obtain the clip-level feature.

4. Experiments

4.1. Datasets and Implementation Details

Datasets. We empirically evaluate the effectiveness of our SIFA-Net and SIFA-Transformer as video backbones on **Kinetics-400** [3], **Kinetics-600** [15], **Something-Something V1 and V2** [16] datasets. The Kinetics-400 dataset consists of 300K videos derived from 400 action categories. Each video in Kinetics-400 is 10-second short clip cropped from the raw YouTube video. In this dataset, all the 300K videos are divided into 240K, 20K, 40K for training, validation and testing, respectively. The Kinetics-600 is an extended version of Kinetics-400, which includes around 480K videos from 600 action categories. There are 390K, 30K, 60K clips in training, validation and testing sets, respectively. In Something-Something V1 and V2 datasets, there are about 108K and 221K videos from 174 action categories, which are mostly for interaction-related recognition. The training/validation/testing set includes 86K/11.5K/11K and 169K/25K/27K videos, respectively.

Network Training. We implement our proposals on PyTorch framework. The mini-batch Stochastic Gradient Descent (SGD) algorithm with cosine learning rate [38] is employed for model optimization. We fix the resolution of each frame as 224×224, which is randomly cropped from the video clip resized with the short size in $[256, 340]$. The input clip length is set in the range from 16 to 64. We randomly flip each clip along horizontal direction for data augmentation, except for Something-Something V1 and V2 in view of the direction-related classes. The size of local region k in SIFA block is set as 3. We set the base learning rate as 0.04 for SIFA-Net and 0.01 for SIFA-Transformer. The dropout ratio is fixed as 0.5. The maximum training epoch number is 128 in Kinetics datasets and 64 in Something-Something datasets. The mini-batch size is 256 and the weight decay parameter is set as 0.0001.

Inference Strategy. We adopt two kinds of inference strategies to evaluate SIFA-Net and SIFA-Transformer. For SIFA-Net, we follow the **3-crop** strategy as in [11] to crop three 256×256 regions from each clip for evaluation. The video-level prediction score is thus achieved by averaging all scores from **10** uniform sampled clips. For SIFA-Transformer, we directly measure the video-level prediction score based on the **4** uniform sampled clips.

4.2. Ablation Study on SIFA Block

In this section, we perform a series of ablation studies to examine several technical choices of our proposed Stand-alone Inter-Frame Attention (SIFA) block in SIFA-Net. Specifically, the deep architecture of SIFA-Net is constructed based on the backbone of ResNet-50, and we report the top-1 and top-5 accuracy on the validation set of Kinetics-400 for performance comparison.

Stand-alone Inter-Frame Attention. We first investigate how each design in our SIFA block influences the overall performance of SIFA-Net. Table 1a details the performance comparisons among different variants of SIFA block. Note that all ablated runs here are constructed by only plugging the SIFA variants into the building blocks at res_5 stage

Table 1. Ablation study on SIFA block in SIFA-Net with 16-frame inputs on Kinetics-400 dataset. Top-1 and Top-5 accuracy (%), and the computational cost (measured in GFLOPs) for forwarding one clip at inference are reported.

(a) **Stand-alone Inter-Frame Attention.** Comparisons among different variants of SIFA. All runs are constructed by plugging each block into res_5 stage of ResNet-50.

Model	GFLOPs	Top-1	Top-5
2D-ResNet	23	72.0	90.3
SIFA$_C$	23	73.3	90.8
SIFA$_R$	24	74.6	91.5
SIFA	24	75.4	92.9
SIFA*	25	75.5	92.9

(b) **Deformable Offset.** Comparisons across different ways on the measure of deformable offset in SIFA block. All runs are constructed by plugging each block into res_5 stage of ResNet-50.

Offset	GFLOPs	Top-1	Top-5
Regular (SIFA$_R$)	24	74.6	91.5
Conv$_{2D}(f_{t+1})$	24	74.7	91.6
Conv$_{3D}(f)$	27	74.8	91.9
Conv$_{2D}(\Delta f)$	24	75.0	92.1
Conv$_{2D}(f_m)$ (SIFA)	24	**75.4**	**92.9**

(c) **Local Region Size.** Comparisons by using different local region size k. All runs are constructed by plugging each block into res_5 stage of ResNet-50.

Size k	GFLOPs	Top-1	Top-5
1×1	24	73.4	90.9
3×3	24	75.4	92.9
5×5	25	75.4	93.0
7×7	26	75.4	93.0
9×9	29	75.5	93.1

(d) **Location of SIFA Block in SIFA-Net.** Effect of plugging SIFA block into different stages of ResNet-50.

Stage				GFLOPs	Top-1	Top-5
res_2	res_3	res_4	res_5			
				23	72.0	90.3
			✓	24	75.4	92.9
		✓	✓	24	76.2	93.0
	✓	✓	✓	25	**77.4**	**93.3**
✓	✓	✓	✓	26	77.4	93.2

(e) **Temporal Modeling.** Comparisons with different temporal modeling techniques (backbone: ResNet-50).

Temporal Modeling	GFLOPs	Top-1	Top-5
2D-ResNet	23	72.0	90.3
Temporal Conv [52]	33	74.1	91.4
Temporal Shift [30]	23	74.7	91.4
Correlation [55]	23	75.1	91.6
Temporal Difference [56]	36	76.6	92.8
SIFA	25	**77.4**	**93.3**

of ResNet-50. We start from a base block (**2D-ResNet**), which is a 2D CNN bottleneck block without any temporal modeling. By upgrading the base block with correlation operator [55], **SIFA**$_C$ exhibits better performances, which show the merit of leveraging the pixel-wise movement information for temporal modeling. **SIFA**$_R$ further aggregates its local temporal neighbors through inter-frame attention, leading to a performance boost of 74.6% in top-1 accuracy. The results basically highlight the advantage of leveraging inter-frame attention to model the temporal correlation within local region across frames. **SIFA** is additionally benefited from the deformable feature re-sampling that explores the irregular geometric transformations of objects in the next frame, and the top-1 accuracy of SIFA finally achieves 75.4%. In addition, we include an upgraded version of our SIFA block, i.e., **SIFA***, that aggregates the temporal neighbors within the locally deformable regions derived from both the previous and next frames, rather than solely involving the temporal neighbors from the next frame as in SIFA. Such temporal aggregation along both forward and backward directions in SIFA* only leads to a marginal performance improvement (0.1% in top-1 accuracy), while requiring more GFLOPs.

Deformable Offset. Next, we compare different approaches of predicting the 2D offset of each spatial location in nearby frame for deformable feature re-sampling in SIFA block. As mentioned in previous section, **SIFA**$_R$ denotes the degraded version of SIFA and only employs inter-frame attention over regular local region in nearby frame, without deformable feature re-sampling. We also include three ablated runs of our SIFA, i.e., Conv$_{2D}(f_{t+1})$, Conv$_{3D}(f)$, and Conv$_{2D}(\Delta f)$, that upgrade SIFA$_R$ with deformable feature re-sampling in multiple ways. Concretely, **Conv**$_{2D}(f_{t+1})$ directly predicts the 2D offset solely based on the feature map of the next frame through 2D convolution. **Conv**$_{3D}(f)$ leverages 3D convolution over the whole clip feature (i.e., the sequence of frame feature maps) to achieve the 2D offset of each spatial location within this clip. **Conv**$_{2D}(\Delta f)$ exploits the temporal difference between adjacent frames to infer the 2D offset via 2D convolution. Table 1b summarizes the performances across different ways on the measure of deformable offset. In particular, by additionally exploring the spatial deformation of objects in each frame as in deformable ConvNets, Conv$_{2D}(f_{t+1})$ slightly improves SIFA$_R$. The result basically validates the effectiveness of deformable feature re-sampling. Compared to Conv$_{2D}(f_{t+1})$ that predicts the deformable offsets of each frame independently, Conv$_{3D}(f)$ jointly infers the offset of each spatial location based on the holistic frame sequence, and thus achieves better performances, while requiring more computational cost. Instead of using 3D convolution to capture motion clues for offset prediction in Conv$_{3D}(f)$, Conv$_{2D}(\Delta f)$ explicitly utilizes the temporal difference between consecutive frames to estimate 2D offset via 2D convolution, leading to performance improvements in an efficient way. Furthermore, by integrating the feature map of the next frame with the inter-frame motion saliency map for offset prediction, Conv$_{2D}(f_m)$ (i.e., our SIFA) obtains the highest performances.

Local Region Size. To explore the effect of local region size k for inter-frame attention learning in SIFA block, we evaluate the performance and computational cost by varying k from 1 to 9 with an interval of 2 in Table 1c. In the extreme case of $k = 1$, only a single temporal neighbor at the same spatial location of nearby frame is taken as key to measure inter-frame attention. As such, the SIFA block degenerates to temporal convolution that only explores temporal evolution in the same spatial location across frames. With the use of larger local region size ($k = 3$), the top-1 accuracy is significantly increased from 73.4% to 75.4%. That basically validates the merit of performing inter-frame attention over locally deformable region across consecutive

Table 2. Performance comparisons on Kinetics-400. The input clip length of SIFA-Net is shown inside the bracket.

Approach	Backbone	GFLOPs×views	Top-1	Top-5
Convolutional Networks				
I3D [3]	Inception	108×N/A	72.1	90.3
TSN [57]	Inception	80×10	72.5	90.2
MF-Net [4]	R34	11×50	72.8	90.4
R(2+1)D [52]	R34	152×10	74.3	91.4
S3D [62]	Inception	71×30	74.7	93.4
TSM [30]	R50	33×30	74.1	91.2
TEINet [33]	R50	33×30	74.9	91.8
TEA [28]	R50	33×30	75.0	91.8
SlowFast [11]	R50+R50	36×30	75.6	92.1
NL I3D [58]	R50	282×30	76.5	92.6
SmallBig [27]	R50	57×30	76.3	92.5
CorrNet [55]	R50	115×10	77.2	-
TDN [56]	R50	72×30	77.5	93.2
SIFA-Net (16)	R50	25×30	77.4	93.3
SIFA-Net (32)	R50	51×30	78.5	93.6
SIFA-Net (64)	R50	112×30	**80.1**	**94.4**
ip-CSN [51]	R101	83×30	76.7	92.3
SmallBig [27]	R101	418×12	77.4	93.3
NL I3D [58]	R101	359×30	77.7	93.3
TDN [56]	R101	132×30	78.5	93.9
CorrNet [55]	R101	224×30	79.2	-
SlowFast [11]	R101+R101	234×30	79.8	93.9
SIFA-Net (16)	R101	39×30	78.7	94.0
SIFA-Net (32)	R101	78×30	79.8	94.2
SIFA-Net (64)	R101	157×30	**81.3**	**95.2**
Vision Transformer				
TimeSformer [2]	ViT-B	2,380×3	80.7	94.7
ViViT [1]	ViT-L	3,992×12	81.3	94.7
MViT [8]	MViT-B	455×9	81.2	95.1
Video-Swin [34]	Swin-B	282×12	82.7	95.5
SIFA-Transformer	Swin-B	270×12	**83.1**	**95.7**

frames. When further enlarging the local region size, the performances are less affected and meanwhile the computational cost is generally increased. Therefore, we empirically set the local region size k as 3, which is seemingly to be a good trade-off between performance and computation cost.

Location of SIFA Block in SIFA-Net. To show the relationship between performance and the location of SIFA block in SIFA-Net, we progressively plug SIFA blocks into the stages in ResNet-50 backbone, and compare the performances. The results shown in Table 1d indicate that inserting SIFA blocks into more stages can generally improve the performances, while increasing the computation cost. When taking a closer look at the top-1 and top-5 accuracy of different locations of SIFA block, the integration of SIFA blocks in the last three stages (res_3, res_4, and res_5) contributes more to the performance boosts. No significant performance improvement is attained when further plugging SIFA block into res_2 stage. Accordingly, we solely integrate the last three stages in ResNet-50 with SIFA blocks, and seek a good accuracy-computation cost balance.

Temporal Modeling. We also compare our SIFA with other existing temporal modeling techniques. Table 1e summarizes the results by integrating the ResNet-50 backbone with different temporal modeling blocks. Overall, our SIFA exhibits better performances than other temporal modeling approaches with less or similar GFLOPs. The results generally indicate the advantage of exploring the deformation across frames to estimate local self-attention for temporal aggregation. In particular, by explicitly capturing motion displacement across frames, Correlation [55] outperforms Temporal Conv [52]. Temporal Difference [56] further boosts the performances by additionally modeling long-term motion. Nevertheless, the performances of Temporal Difference are still lower than that of our SIFA which exploits inter-frame attention for temporal modeling.

Table 3. Performance comparisons on Kinetics-600. The input clip length of SIFA-Net is shown inside the bracket.

Approach	Backbone	GFLOPs×views	Top-1	Top-5
Convolutional Networks				
I3D [3]	Inception	108×N/A	71.9	90.1
SlowFast [11]	R50+R50	36×30	78.8	94.0
SIFA-Net (16)	R50	25×30	79.6	94.5
SIFA-Net (32)	R50	51×30	80.5	95.2
SIFA-Net (64)	R50	112×30	**82.1**	**95.8**
SlowFast [11]	R101+R101	234×30	81.8	95.1
X3D-XL [10]	custom	48×30	81.9	95.5
SIFA-Net (16)	R101	39×30	80.8	95.2
SIFA-Net (32)	R101	78×30	81.6	95.5
SIFA-Net (64)	R101	157×30	**83.2**	**95.9**
Vision Transformer				
TimeSformer [2]	ViT-B	1,703×3	82.4	96.0
ViViT [1]	ViT-L	3,992×12	83.0	95.7
MViT [8]	ViT-B	236×5	83.8	96.3
Video-Swin [34]	Swin-B	282×12	84.0	96.5
SIFA-Transformer	Swin-B	270×12	**84.5**	**96.9**

4.3. Comparisons with State-of-the-Art Methods

We compare SIFA-Net and SIFA-Transformer with various state-of-the-art techniques on Kinetics-400, Kinetics-600, and Something-Something V1 (SSv1) and V2 (SSv2) datasets. All runs are briefly grouped into two paradigms: Convolutional Networks and Vision Transformer. Note that we implement SIFA-Net in two kinds of backbones, i.e., ResNet-50 (R50) and ResNet-101 (R101), and the input clip length is varied in the range of {16, 32, 64}. The SIFA-Transformer is constructed based on the backbone of Swin Transformer (Swin-B) with the fixed input clip length (64 frames). The computational cost is measured in GFLOPs × views, and the views represent the number of clips sampled from the full video at inference.

Table 2 summarizes the performance comparisons on Kinetics-400. For the group of Convolutional Networks, our SIFA-Net leads to better performances against other baselines. In particular, SIFA-Net (32) in R50 backbone obtains 78.5% top-1 accuracy, and outperforms the best competitor TDN by 1.0% but with ∼30% less computation cost in GFLOPs. By sampling more frames in each clip for temporal modeling, SIFA-Net (64) improves the top-1 accuracy from 78.5% to 80.1%. The superior results of SIFA-Net generally demonstrate the advantage of integrating 2D CNN with inter-frame attention to enable temporal modeling. When further inserting SIFA block into a state-of-the-art 2D Vision Transformer backbone (Swin Transformer), SIFA-Transformer manages to achieve the best performance (83.1% in top-1 accuracy) on Kinetics-400. The performance of SIFA-Transformer is comparable to the superior 3D Vision Transformer (Video-Swin), but requires less computation cost. The performance trends

Table 4. Performances on Something-Something V1 and V2. The input clip length of SIFA-Net is shown inside the bracket.

Approach	Backbone	GFLOPs ×views	SSv1		SSv2	
			Top-1	Top-5	Top-1	Top-5
Convolutional Networks						
NL I3D+GCN [59]	R50	606	46.1	76.8	-	-
CPNet [31]	R34	N/A	-	-	57.7	84.0
TSM [30]	R50	98	47.2	77.1	63.4	88.5
TAM [9]	R50	48	48.4	78.8	61.7	88.1
GST [39]	R50	59	48.6	77.9	62.6	87.9
SmallBig [27]	R50	105	49.3	79.5	62.3	88.5
CorrNet [55]	R50	115×10	49.3	-	-	-
ACTION-Net [60]	R50	69	-	-	64.0	89.3
STM [19]	R50	67×30	50.7	80.4	64.2	89.8
MSNet [22]	R50	67	52.1	82.3	64.7	89.4
TEINet [33]	R50	99	52.5	-	65.5	89.8
MG-TEA [65]	R50	N/A	53.2	-	63.8	-
TDN [56]	R50	72	53.9	82.1	65.3	89.5
SIFA-Net (16)	R50	25×3	52.7	81.9	64.8	89.4
SIFA-Net (32)	R50	51×3	54.0	82.2	66.0	89.6
SIFA-Net (64)	R50	112×3	**55.2**	**83.3**	**66.9**	**90.7**
GSM [49]	Inception	268	55.2	-	-	-
CorrNet [55]	R101	224×30	53.3	-	-	-
MG-TEA [65]	R101	N/A	53.3	-	64.8	-
TDN [56]	R101	132	55.3	83.3	66.9	90.9
SIFA-Net (16)	R101	39×3	53.7	82.0	65.9	89.8
SIFA-Net (32)	R101	78×3	55.4	83.1	67.3	91.1
SIFA-Net (64)	R101	157×3	**56.1**	**84.0**	**68.1**	**92.0**
Vision Transformer						
TimeSformer [2]	ViT-B	1,703×3	-	-	62.5	-
ViViT [1]	ViT-L	903	-	-	65.4	89.8
MViT [8]	ViT-B	455×3	-	-	67.7	90.9
Video-Swin [34]	Swin-B	321×3	-	-	69.6	92.7
SIFA-Transformer	Swin-B	270×3	**57.3**	**85.1**	**69.8**	**93.1**

on Kinetics-600 are similar with those on Kinetics-400 as shown in Table 3. The results again verify the impact of SIFA block in both 2D CNN and Vision Transformer backbones for video representation learning. Table 4 lists the performances on both SSv1 and SSv2 datasets. Particularly, we follow the one-clip and 3-crop settings [2,8,34] for testing on Something-Something. Similarly, SIFA-Net (64) in R50 and R101 backbones surpasses the best competitor TDN by 1.3%/1.6% and 0.8%/1.2% in top-1 accuracy on SSv1/SSv2, respectively. Furthermore, by plugging SIFA block into Swin-B backbone, our SIFA-Transformer obtains the best performances on both SSv1 and SSv2 datasets.

4.4. Visualization Analysis of SIFA

To better qualitatively examine SIFA block for video representation learning, we further visualize the inter-frame attention map over the locally deformable region, motion saliency map (MSM) and the class activation map with Grad-CAM [46] of SIFA-Net (R50 backbone) in Figure 5. Note that we take the spatial location in center frame (t-th frame) of each sampled clip as the query, and employ the attention map of SIFA block in res_5 stage for visualization. In addition, for SIFA block with k=3, the deformable feature re-sampling is performed at two levels as in deformable ConvNets, leading to $9^2 = 81$ sampling points in (t+1)-th frame. As shown in the figure, the calculated motion saliency map of (t+1)-th frame matches the class activation map in general, which shows that the learnt MSM is able to capture the meaningful motion cues that benefit action classification. Through re-scaling deformable feature re-sampling with MSM, the sampling points are nicely adjusted according to the objects' scale, irregular shape, and large movements. This again confirms that SIFA block takes the object movement and deformation across frames into account to strengthen inter-frame feature alignment, thereby boosting temporal modeling.

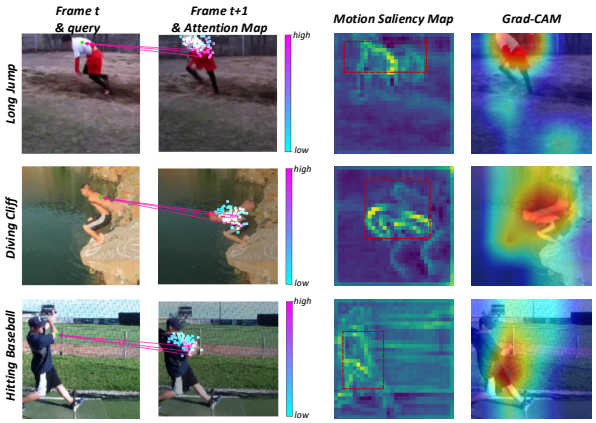

Figure 5. Visualization of the inter-frame attention map, motion saliency map (MSM) and Grad-CAM [46] of SIFA-Net for three videos in Kinetics-400. For the video in each row, the green point in its t-th frame denotes the query location. The correlation between query and sampling points in (t+1)-th frame (i.e., attention weight) is shown in heat map. We link the query and sampling points with top-3 attention weights in purple line. The red box in MSM represents the region with highly salient object movements.

5. Conclusions and Discussions

We have presented Stand-alone Inter-Frame Attention (SIFA) block, which explores the deformation across frames for temporal modeling with local self-attention. Specifically, by taking the spatial location in current frame as query, SIFA performs self-attention over the keys/values in a local neighboring region of next frame. Moreover, to tackle the irregular object deformation in next frame, a deformable design is leveraged to estimate the offset of each spatial location in local region, yielding the keys/values re-sampled in a deformation. Such deformable feature re-sampling is additionally re-scaled by motion cues to facilitate inter-frame attention learning. Finally, all deformable values are aggregated with attention to enhance per-frame feature. By plugging SIFA block into ResNet and Swin Transformer, we construct two new video backbones (SIFA-Net and SIFA-Transformer), and the experiments on four action recognition datasets demonstrate their effectiveness.

Broader Impact. One negative impact of this research in video representation learning is the significant environmental impact associated with training Transformer backbones, which are large and computationally expensive. There is also potential for these action recognition models to be misused, such as for unauthorized surveillance.

Acknowledgments. This work was supported by the National Key R&D Program of China under Grant No. 2020AAA0108600.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lucic, and Cordelia Schmid. ViViT: A Video Vision Transformer. In *ICCV*, 2021. 7, 8

[2] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is Space-Time Attention All You Need for Video Understanding? In *ICML*, 2021. 2, 3, 7, 8

[3] Joao Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *CVPR*, 2017. 1, 2, 5, 7

[4] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. Multi-Fiber Networks for Video Recognition. In *ECCV*, 2018. 7

[5] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable Convolutional Networks. In *ICCV*, 2017. 4

[6] Ali Diba, Vivek Sharma, and Luc Van Gool. Deep Temporal Linear Encoding Networks. In *CVPR*, 2017. 2

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*, 2021. 2, 5

[8] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale Vision Transformers. *arXiv preprint arXiv:2104.11227*, 2021. 7, 8

[9] 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. In *NeurIPS*, 2019. 8

[10] Christoph Feichtenhofer. X3D: Expanding Architectures for Efficient Video Recognition. In *CVPR*, 2020. 7

[11] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. SlowFast Networks for Video Recognition. In *ICCV*, 2019. 5, 7

[12] Christoph Feichtenhofer, Axel Pinz, and Richard P. Wildes. Spatiotemporal Multiplier Networks for Video Action Recognition. In *CVPR*, 2017. 2

[13] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional Two-Stream Network Fusion for Video Action Recognition. In *CVPR*, 2016. 2

[14] Philipp Fischer, Alexey Dosovitskiy, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. FlowNet: Learning Optical Flow with Convolutional Networks. In *ICCV*, 2015. 3

[15] Bernard Ghanem, Juan Carlos Niebles, Cees Snoek, Fabian Caba Heilbron, Humam Alwassel, Victor Escorcia, Ranjay Krishna, Shyamal Buch, and Cuong Duc Dao. The ActivityNet Large-Scale Activity Recognition Challenge 2018 Summary. *arXiv preprint arXiv:1808.03766*, 2018. 5

[16] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The "something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 5

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016. 2, 3, 4, 5

[18] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Trans. on PAMI*, 2013. 1

[19] Boyuan Jiang, MengMeng Wang, Weihao Gan, Wei Wu, and Junjie Yan. STM: SpatioTemporal and Motion Encoding for Action Recognition. In *ICCV*, 2019. 8

[20] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale Video Classification with Convolutional Neural Networks. In *CVPR*, 2014. 2

[21] Alexander Klaser, Marcin Marszalek, and Cordelia Schmid. A Spatio-Temporal Descriptor based on 3D-Gradients. In *BMVC*, 2008. 2

[22] Heeseung Kwon, Manjin Kim, Suha Kwak, and Minsu Cho. MotionSqueeze: Neural Motion Feature Learning for Video Understanding. In *ECCV*, 2020. 8

[23] Ivan Laptev. On Space-Time Interest Points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005. 2

[24] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning Realistic Human Actions from Movies. In *CVPR*, 2008. 2

[25] Dong Li, Zhaofan Qiu, Yingwei Pan, Ting Yao, Houqiang Li, and Tao Mei. Representing Videos as Discriminative Sub-graphs for Action Recognition. In *CVPR*, 2021. 2

[26] Dong Li, Ting Yao, Zhaofan Qiu, Houqiang Li, and Tao Mei. Long Short-Term Relation Networks for Video Action Detection. In *ACM MM*, 2019. 2

[27] Xianhang Li, Yali Wang, Zhipeng Zhou, and Yu Qiao. SmallBigNet: Integrating Core and Contextual Views for Video Classification. In *CVPR*, 2020. 7, 8

[28] Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. TEA: Temporal Excitation and Aggregation for Action Recognition. In *CVPR*, 2020. 7

[29] Yehao Li, Ting Yao, Yingwei Pan, and Tao Mei. Contextual Transformer Networks for Visual Recognition. *IEEE Trans. on PAMI*, 2022. 2

[30] Ji Lin, Chuang Gan, and Song Han. TSM: Temporal Shift Module for Efficient Video Understanding. In *ICCV*, 2019. 6, 7, 8

[31] Xingyu Liu, Joon-Young Lee, and Hailin Jin. Learning Video Representations from Correspondence Proposals. In *CVPR*, 2019. 8

[32] 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. In *ICCV*, 2021. 2, 3, 5

[33] 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 *AAAI*, 2020. 7, 8

[34] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video Swin Transformer. *arXiv preprint arXiv:2106.13230*, 2021. 7, 8

[35] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Gaussian Temporal Awareness Networks for Action Localization. In *CVPR*, 2019. 2

[36] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Learning to Localize Actions from Moments. In *ECCV*, 2020. 2

[37] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Tao Mei, and Jiebo Luo. Coarse-to-Fine Localization of Temporal Action Proposals. *IEEE Trans. on Multimedia*, 22(6):1577 – 1590, 2020. 2

[38] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic Gradient Descent with Warm Restarts. In *ICLR*, 2017. 5

[39] Chenxu Luo and Alan Yuille. Grouped Spatial-Temporal Aggregation for Efficient Action Recognition. In *ICCV*, 2019. 8

[40] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond Short Snippets: Deep Networks for Video Classification. In *CVPR*, 2015. 2

[41] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks. In *ICCV*, 2017. 2

[42] Zhaofan Qiu, Ting Yao, Chong-Wah Ngo, and Tao Mei. Optimization Planning for 3D ConvNets. In *ICML*, 2021. 5

[43] Zhaofan Qiu, Ting Yao, Chong-Wah Ngo, Xinmei Tian, and Tao Mei. Learning Spatio-Temporal Representation with Local and Global Diffusion. In *CVPR*, 2019. 2

[44] Ignacio Rocco, Relja Arandjelovic, and Josef Sivic. Convolutional Neural Network Architecture for Geometric Matching. In *CVPR*, 2017. 3

[45] Paul Scovanner, Saad Ali, and Mubarak Shah. A 3-Dimensional SIFT Descriptor and Its Application to Action Recognition. In *ACM MM*, 2007. 2

[46] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations From Deep Networks via Gradient-Based Localization. In *ICCV*, 2017. 8

[47] Karen Simonyan and Andrew Zisserman. Two-stream Convolutional Networks for Action Recognition in Videos. In *NIPS*, 2014. 2

[48] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised Learning of Video Representations using LSTMs. In *ICML*, 2015. 2

[49] Swathikiran Sudhakaran, Sergio Escalera, and Oswald Lanz. Gate-Shift Networks for Video Action Recognition. In *CVPR*, 2020. 8

[50] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning Spatiotemporal Features with 3D Convolutional Networks. In *ICCV*, 2015. 1, 2

[51] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video Classification with Channel-Separated Convolutional Networks. In *ICCV*, 2019. 7

[52] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A Closer Look at Spatiotemporal Convolutions for Action Recognition. In *CVPR*, 2018. 1, 5, 6, 7

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *NIPS*, 2017. 2, 3

[54] Heng Wang, Alexander Klaser, Cordelia Schmid, and Cheng-Lin Liu. Action Recognition by Dense Trajectories. In *CVPR*, 2011. 2

[55] Heng Wang, Du Tran, Lorenzo Torresani, and Matt Feiszli. Video Modeling with Correlation Networks. In *CVPR*, 2020. 3, 6, 7, 8

[56] Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. TDN: Temporal Difference Networks for Efficient Action Recognition. In *CVPR*, 2021. 6, 7, 8

[57] 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. 2, 7

[58] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local Neural Networks. In *CVPR*, 2018. 3, 7

[59] Xiaolong Wang and Abhinav Gupta. Videos as Space-Time Region Graphs. In *ECCV*, 2018. 8

[60] Zhengwei Wang, Qi She, and Aljosa Smolic. ACTION-Net: Multipath Excitation for Action Recognition. In *CVPR*, 2021. 8

[61] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. DeepFlow: Large Displacement Optical Flow with Deep Matching. In *ICCV*, 2013. 3

[62] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-offs in Video Classification. In *ECCV*, 2018. 1, 2, 5, 7

[63] Ting Yao, Yiheng Zhang, Zhaofan Qiu, Yingwei Pan, and Tao Mei. SeCo: Exploring Sequence Supervision for Unsupervised Representation Learning. In *AAAI*, 2021. 2

[64] Yue Zhao, Yuanjun Xiong, and Dahua Lin. Trajectory Convolution for Action Recognition. In *NeurIPS*, 2018. 2

[65] Yuan Zhi, Zhan Tong, Limin Wang, and Gangshan Wu. MGSampler: An Explainable Sampling Strategy for Video Action Recognition. In *ICCV*, 2021. 8