

Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems

School of Computing and Information Systems

10-2022

Long-term leap attention, short-term periodic shift for video classification

Hao ZHANG

Lechao CHENG

Yanbin HAO

Chong-wah NGO

Singapore Management University, cwngo@smu.edu.sg

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research



Part of the [Artificial Intelligence and Robotics Commons](#), and the [Graphics and Human Computer Interfaces Commons](#)

Citation

ZHANG, Hao; CHENG, Lechao; HAO, Yanbin; and NGO, Chong-wah. Long-term leap attention, short-term periodic shift for video classification. (2022). *Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 2022 October 10-14*. 5773-5782.

Available at: https://ink.library.smu.edu.sg/sis_research/7507

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylids@smu.edu.sg.

Long-term Leap Attention, Short-term Periodic Shift for Video Classification

Hao Zhang
Singapore Management University
Singapore
hzhang@smu.edu.sg

Yanbin Hao*
University of Science and Technology of China
Hefei, China
haoyanbin@hotmail.com

Lechao Cheng
Zhejiang Lab
Hangzhou, China
chenglc@zhejianglab.com

Chong-wah Ngo
Singapore Management University
Singapore
cwngo@smu.edu.sg

ABSTRACT

Video transformer naturally incurs a heavier computation burden than a static vision transformer, as the former processes T times longer sequence than the latter under the current attention of quadratic complexity (T^2N^2). The existing works treat the temporal axis as a simple extension of spatial axes, focusing on shortening the spatio-temporal sequence by either generic pooling or local windowing without utilizing temporal redundancy.

However, videos naturally contain redundant information between neighboring frames; thereby, we could potentially suppress attention on visually similar frames in a dilated manner. Based on this hypothesis, we propose the LAPS, a long-term “Leap Attention” (LA), short-term “Periodic Shift” (P -Shift) module for video transformers, with $(2TN^2)$ complexity. Specifically, the “LA” groups long-term frames into pairs, then refactors each discrete pair via attention. The “ P -Shift” exchanges features between temporal neighbors to confront the loss of short-term dynamics. By replacing a vanilla 2D attention with the LAPS, we could adapt a static transformer into a video one, with zero extra parameters and neglectable computation overhead ($\sim 2.6\%$). Experiments on the standard Kinetics-400 benchmark demonstrate that our LAPS transformer could achieve competitive performances in terms of accuracy, FLOPs, and Params among CNN and transformer SOTAs. We open-source our project in <https://github.com/VideoNetworks/LAPS-transformer>.

CCS CONCEPTS

• Computing methodologies → Activity recognition and understanding.

KEYWORDS

Video classification; Transformer; Shift; Leap attention

*Yanbin Hao is the corresponding author.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '22, October 10–14, 2022, Lisboa, Portugal

© 2022 Association for Computing Machinery.

ACM ISBN 978-1-4503-9203-7/22/10...\$15.00

<https://doi.org/10.1145/3503161.3547908>

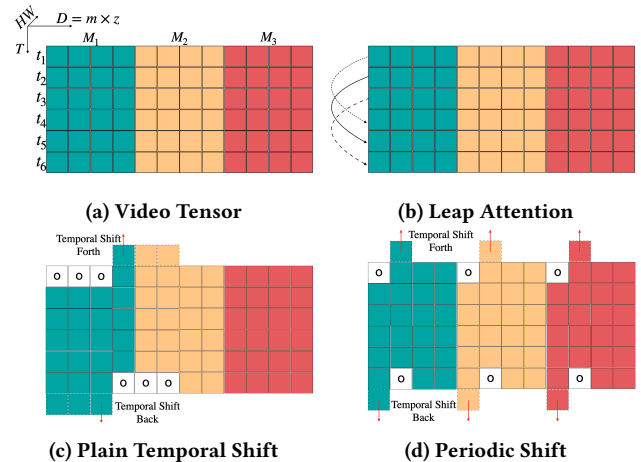


Figure 1: Manipulations on tensor. (a) None; (b) Leap Attention: pairing frames with long-term temporal steps; (c) Plain Temporal Shift; (d) Periodic Shift: shifting features with periodicity. D, m, z, \square denotes total dim, number of heads, dim per head and zeros padding. Features of same hue originates from the same attention head $M_{\{1,2,3\}}$. (The figure is best viewed in color)

ACM Reference Format:

Hao Zhang, Lechao Cheng, Yanbin Hao, and Chong-wah Ngo. 2022. Long-term Leap Attention, Short-term Periodic Shift for Video Classification. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22)*, October 10–14, 2022, Lisboa, Portugal. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3503161.3547908>

1 INTRODUCTION

Transformers recently swept research areas such as NLP [49], vision [10], and video understanding [1] and became the de-facto standards. However, the transformer inherently contains attention mechanism of quadratic complexity, thus innating a disadvantage when dealing with long sequences. This problem is particularly evident with the video input for having T^1 longer sequence than the image (N), boosting the attention complexity to (T^2N^2) .

¹ T refers to the number of frames

Current video transformers mainly treat the temporal axis as a simple extension of spatial axes, resorting to generic operators like pooling [11], local windowing [34], or factorization of spatio-temporal attention [1] to reduce complexity. However, videos are temporally redundant. Temporally neighboring frames are generally similar despite being different in micro details. These details gradually accumulate into a qualitative change with enough time evolution. Inspiringly, we suggest avoiding imposing attention on temporally adjacent frames. Alternatively, we could pair frames with a long-term skipped step in between and then apply attention to each pair. This dilated manner would be computationally economical. We also launch a cost-effective variant of the *Shift* operator to capture micro details.

To connect long-term temporal relations, we propose **Leap Attention**. Similar to the dilated convolutions [60], the LA picks frame pairs with discrete temporal distances for attention inputs. As a result, the temporal reception field is expanded at a low computation cost (see Figure 1b). Besides, to equip the LA with multi-scale temporal reception fields, we loop the skipped step in a multiple pyramid way according to layer’s depth. We experimentally verify that the LA could surpass the performance of 2D attention and approach 3D attention with an economic complexity ($2TN^2$).

To capture short-term micro variations, we complement the LA with **Periodic Shift**. This *P*-Shift is inspired by the TSM [30], but is customized according to the multi-head mechanism. Recalling that channels of CNN feature are generally sourced from an identical network path, while Multi-Head Self-Attention (MSHA) channels are generated with a latent periodicity. As m heads independently yield sub-channels (z -dim) for a total D -dim output, every z out of D channels originates from the same head. Our *P*-Shift complies with this characteristics and periodically shifts each z -dim sub-feature of head (Figure 1d). We compare it with a simple copycat named “plain shift” (Figure 1c) and observe a better experimental performance.

The complete **LAPS** module is a cascade of the LA and *P*-Shift as in Figure 2. It inherits the merit of zero-parameter from both submodules. Thus, it can flexibly convert a static transformer into a video one by replacing the 2D attention. Besides, it could directly load a checkpoint from the 2D pre-training and converge in a few epochs. Extensive experiments on the standard video benchmark verify the efficacy, training/inference efficiency, and flexibility. Our contributions are summarized below:

- **Leap Attention** adjusts generic spatio-temporal attention into a dilated manner, thus achieving a high cost-effective for video transformers. We also adopt a multi-pyramid strategy to equip the LA with multi-scale temporal reception fields.
- **Periodic Shift** is customized according to the multi-head mechanism, which separately imposes a temporal shift on the output of each attention head. It serves for capturing short-term micro variations between temporal neighbors, with a zero-parameter/FLOPs cost.
- **LAPS (their cascade)** is a zero-parameter, lightweight-FLOPs attention alternative. It can flexibly replace a generic 2D attention and convert a static vision transformer into a video one, facilitating the possibility of developing a video transformer with strong 2D backbones.

2 RELATED WORKS

Our work is relevant to research areas in video and image understandings. Specifically, the prior covers researches in 3D convolutional networks and video transformers; the latter contains recent progress like image transformers. We will review them as below.

Convolutional Neural Networks for video understanding have been extensively studied and widely applied to video-text pre-training [36], cross-modal analysis [29], video detection [19], ecommerce [6–8], adversarial attack [4, 53–55], interactive search [37, 58], retrieval [17, 18, 26, 57, 62, 64, 66], hyperlinking [9, 20, 21, 39], and caption [2, 47, 61], in the CNN era; we select and review representative 3D-CNNs as follows. C3D [48] is a pure 3D-CNN pilot based on a new 3D Conv operator and easily outperforms 2D counterparts on video tasks. Then, the I3D [3] increases the depth of network by migrating 3D convs into inception-net [45], and is pre-trained on a large-scale video benchmark named Kinetics to prevent overfitting. To compress parameters and FLOPs, P3D [40] factorizes 3D convolution kernel into a combination of 2D spatial + 1D temporal kernels. This factorization demonstrates good efficiency and efficacy, thus gaining popularity in later network design. DB-LSTM [24] learns long-range video sequences with Bi-directional LSTM. To further optimize efficiency, TSM [30] adopts the zero-parameter/FLOP Shift operators for temporal modeling. Consequently, it could keep the same complexity² as the 2D backbone. Furthermore, TIN [42] and Gate-Shift [44] extend temporal shift with adaptive steps, which is predicted by sub-networks. RubiksNet [12] further reduces computations by proposing a 3D shift operator, which learns 3D steps to shift channels along both spatial and temporal axis. Several works [22, 27, 33, 35, 46, 50] study the impact of long short-term axial contexts on video tasks. They successfully verified that long short-term axial contexts of videos can be utilized in a squeeze-and-excitation manner and are beneficial for content recognition. Small-Big Network [25] proposes to enlarge the spatial-temporal reception field by paralleling a 3D pooling branch on par with convolutional branches, balancing efficiency and efficacy. Non-Local [51] attempt to implant self-attention into CNNs. Furthermore, CBA-QSA [23] enhances attention with compact bilinear mapping to emphasize local evidence for sport classification. The SlowFast [14] and X3D [13] are SOTAs of 3D-CNN. Specifically, the prior is designed into dual-paths form, one path for semantics and another for motion. The latter relies on searching hyperparameters (e.g., resolution, channels, depth) from a template network, optimizing efficiency and efficacy simultaneously.

Image and Video Transformers. Transformers for images and videos are closely relevant as the latter is often extended from the prior. We elaborate on recent progress for images first, and then turn to videos.

Dosovitskiy et. al [10] develops the ViT out of a vanilla language transformer with a modification on accepting sequential patches of an image as input. Since the ViT is computationally heavy, later efforts are devoted to efficiency optimization. For example, Swin [32] replaces global attention with a local windowed one for reducing the length of tokens. Similar in spirit, the MViT [11] turns to pooling for length shrinking. The rest works develop hybrid

²In terms of model parameters and FLOPs per frame

models, such as CMT [16], Visformer [5], CvT [56] and ResT [65], utilizing efficient convolutions for optimization.

Video transformer often contains an extra temporal module than the image one. [1] introduces factorization on 3D attention, which decomposes spatio-temporal attention into a combination of spatial and temporal attention as P3D. VTN in [38] extracts per-frame feature, using 2D-CNN or vision transformer, and then refines these features with a stack of temporal attention encoders. [63] verify that the Shift operator is still valid for a ViT backbone on video tasks. Moreover, Video Swin [34], and MViT for video are directly extended from corresponding 2D transformers by inflating 2D windowing/pooling to 3D ones. There is also attention-free pilot, such as MLP-3D [41] that optimizes efficiency for video understanding.

3 METHOD

An overview of the **Leap Attention with Periodic Shift** (LAPS) encoder is presented in Figure 2. The LAPS mainly modifies the vanilla 2D Self-Attention from the following two perspectives: (1). *Leap Attention* for imposing attention on frame pairs with a long-term skipped step; (2). *Periodic Shift* for exchanging features between short-term adjacent frames. We illustrate the detailed pipeline of the LAPS video transformer as below.

A video $\mathbf{v} \in \mathbb{R}^{T \times H \times W \times 3}$ is gridly divided into a sequential tensor $\tilde{\mathbf{v}} \in \mathbb{R}^{T \times N \times d}$, where $T, H/W, P$ separately denotes clip-length, height/width and patch-size. $N = \frac{H \cdot W}{P^2}$ and $d = P^2 \cdot 3$ represents amount of patches (also known as tokens) and RGB pixels of a patch.

The sequential tensor $\tilde{\mathbf{v}}$ is then embedded into D -dimensional space as in Equation 1. Hereby, function $f_{em}(\cdot)$ can be achieved by either a single-layer linear projection or a stack of convolutions. Notably, we follow the trend in [5] and [56] whom adopt a stack of convolutions for $f_{em}(\cdot)$ and remove extra [Class] token on vision tasks.

$$\begin{aligned} X_0 &= f_{em}(\tilde{\mathbf{v}}) \\ X_0 &\in \mathbb{R}^{T \times N \times D} \end{aligned} \quad (1)$$

The video tensor X_0 is then fed through several encoders for representation learning.

3.1 LAPS Encoder

Our video transformer consists of L identical LAPS encoders, each encoder (Figure 2) contains a LAPS module for spatial-temporal relation modeling and a Multi-Layer Perceptron (MLP) for feature mapping. The function $f_{laps}(\cdot)$ and $f_{mlp}(\cdot)$ separately represents the LAPS and MLP module.

$$\tilde{X}_{l-1} = f_{laps}(X_{l-1}) + X_{l-1} \quad (2)$$

$$X_l = f_{mlp}(\tilde{X}_{l-1}) + \tilde{X}_{l-1} \quad (3)$$

$$X_{l/l-1}, \tilde{X}_{l/l-1} \in \mathbb{R}^{T \times N \times D}$$

$$D = m \times z, \quad l \in [1, 2, \dots, L]$$

Hereby, D, m, z represent the total dimensions of MSHA, number of heads, and dimensions per head.

The LAPS module $f_{laps}(\cdot)$ contains two sub-modules: Leap Attention (LA) and Periodic Shift (P -Shift). The prior serves to model

long-term temporal relations between paired frames at a distance; the latter servers to capture short-term variation between adjacent frames. We leave the detailed presentation of LA and P -Shift in Section 3.2 & 3.3.

The MLP shares a similar process as it in the vanilla ViT. It contains two layers of linear projection with a GELU activation in between. For simplicity, we implement linear projection in (Conv-1×1) form.

$$f_{mlp}(\tilde{X}_l) = Conv \left(GELU \left(Conv \left(\tilde{X}_l \right) \right) \right) \quad (4)$$

The **Prediction Layer** works on the output $\tilde{X}_L \in \mathbb{R}^{T \times N \times D}$ from the last encoder. We firstly apply spatial average pooling for per-frame prediction, then average scores across timeline for clip-level prediction, hereby, $FC(\cdot)$ denotes linear projection of shape “ $D \times$ Categories”.

$$\mathbf{c} = SpatialAvgPool(\tilde{X}_L) \quad (5)$$

$$\mathbf{y} = \frac{1}{T} \sum_{i=1}^T FC(\mathbf{c}) \quad (6)$$

$$\mathbf{y} \in \mathbb{R}^{Categories}, \quad \mathbf{c} \in \mathbb{R}^{T \times D}$$

3.2 Leap Attention

The Leap Attention module is inspired by the success of *Dilated Convolution* [60] in the CNN era. Common 2/3D transformers impose attention on continuous receptive field; we propose to compute self-attention on discrete temporal frame pairs. With the dilated strategy, we could enlarge the temporal receptive field of the attention with a neglectable computational overhead.

The LA contains two phases: Grouping frames into pairs and Imposing attention on each pair. We firstly illustrate the strategy of skipped pariwise as below.

Pick the i -th sub-tensor $M_i \in \mathbb{R}^{T \times N \times z}$, which would be fed into the i -th attention head, out of input tensor $X \in \mathbb{R}^{T \times N \times (m \times z)}$. Hereby, m denotes the total number of attention heads. The Q, K, V tensors are linear projected from M_i as below:

$$Q, K, V = M_i \times [W_q, W_k, W_v] \quad (7)$$

$$Q, K, V \in \mathbb{R}^{T \times N \times z}, \quad W_{q,k,v} \in \mathbb{R}^{z \times z}$$

We need two temporal index lists A and B to align frames into pairs. Specifically, we separate the temporal indexes $[0, 1, 2, \dots, T-1]$ into two sorted lists A and B , under a condition:

$$B - A = \left[\underbrace{S, S, \dots, S}_{\frac{T}{2}} \right] \quad (8)$$

Frames with temporal index ($A[j], B[j]$) are paired, where $j = [0, 1, \dots, \frac{T}{2}]$. Hereby, S denotes the temporal skipped step. For example in Figure 3a, when $T=8$ and $S=4$, then, $A=[0, 1, 2, 3], B=[4, 5, 6, 7]$, the alignment of pairs will be: (0,4), (1,5), (2,6), (3,7).

We further split Q, K, V into two sub-tensors according to temporal indexes A and B :

$$Q_{\{A \text{ or } B\}}, K_{\{A \text{ or } B\}}, V_{\{A \text{ or } B\}} \in \mathbb{R}^{\frac{T}{2} \times N \times z}$$

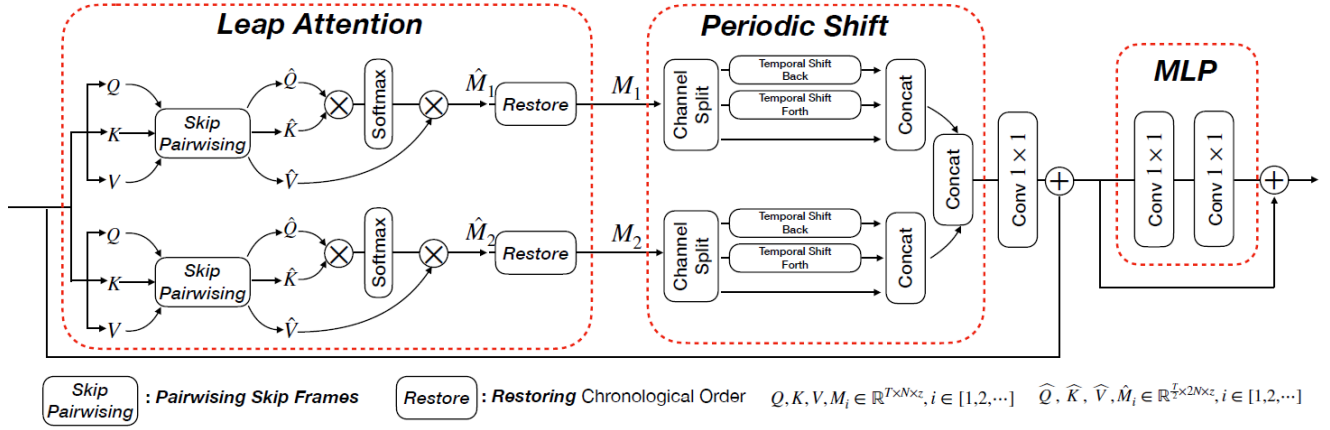


Figure 2: An overview of LAPS Encoder. Compared with vanilla Multi-Head Self-Attention, the LAPS contains a *Leap Attention* and *Periodic Shift* module, separately for modeling long-term interaction between frames at a distance and short-term variation within adjacent frames.

The Q_A, K_A, V_A and Q_B, K_B, V_B are concatenated along the 2-nd axis to align frame t and $t + S$ into pairs.

$$\hat{Q} = \text{Concat}(Q_A, Q_B) \quad (9)$$

$$\hat{K} = \text{Concat}(K_A, K_B) \quad (10)$$

$$\hat{V} = \text{Concat}(V_A, V_B) \quad (11)$$

$$\hat{Q}, \hat{K}, \hat{V} \in \mathbb{R}^{\frac{T}{2} \times (2N) \times z}$$

We then perform attention on each frame pair (along the 2-nd red axis), which contains $2N$ tokens:

$$\hat{M}_i = \text{Softmax}\left(\frac{\hat{Q}\hat{K}}{\sqrt{z}}\right)\hat{V} \quad (12)$$

We reverse the skip pairwsing procedure and restore the \hat{M}_i to original chronological temporal order \tilde{M}_i based on index lists A and B :

$$\hat{M}_i \in \mathbb{R}^{\frac{T}{2} \times (2N) \times z} \rightarrow \tilde{M}_i \in \mathbb{R}^{T \times N \times z} \quad (13)$$

The outputs from m heads are concatenated as \tilde{X} :

$$\tilde{X} = \text{Concat}(\tilde{M}_1, \tilde{M}_2, \dots, \tilde{M}_m) \quad (14)$$

$$\tilde{X} \in \mathbb{R}^{T \times N \times D}, \quad D = m \times z$$

The **Pyramid Skipping** aims to connect frames with various distances into pairs, therefore facilitating the LA to have multi-scale temporal receptive fields. Specifically, we define the skip step S by Equation (15) concerning pyramid level R .

$$S = \frac{T}{2^R} \quad (15)$$

Alignment of temporal pairs with respect to pyramid level $R = 1, 2, 3$ is shown in Figure 3a-3c. Since a transformer contains L amounts of LAs (residing in each encoder), we loop pyramid level R by values

Algorithm 1: Leap Attention

Input: video tensor $X \in \mathbb{R}^{T \times N \times (m \times z)}$, pyramid-level R

Output: video tensor $\tilde{X} \in \mathbb{R}^{T \times N \times (m \times z)}$ refactor from skipped neighbors

1 Let temporal skip step $S = \frac{T}{2^R}$

2 **for** $i = 1$ to m **do**

3 Select the i -th head $M_i \in \mathbb{R}^{T \times N \times z}$ from X .

4 Generate $Q, K, V \in \mathbb{R}^{T \times N \times z}$ from M_i

5 Init index list A and $B = [\emptyset]$

6 **for** $t = 1$ to T **do**

7 **if** $t \notin \{A \cup B\}$ **then**

8 Add t to set A : $A = A \cup \{t\}$

9 Add $t + S$ to set B : $B = B \cup \{t + S\}$

10 **end**

11 **end**

12 Sort A and B in ascending order

13 Select sub-tensor by A : $Q/K/V_A \in \mathbb{R}^{\frac{T}{2} \times N \times z}$.

14 Select sub-tensor by B : $Q/K/V_B \in \mathbb{R}^{\frac{T}{2} \times N \times z}$.

15 Concat over 2-nd axis: $\hat{Q}/\hat{K}/\hat{V} \in \mathbb{R}^{\frac{T}{2} \times (2N) \times z}$

16 Perform attention over 2-nd axis ($2N$ tokens):

$$\hat{M}_i = \text{Softmax}\left(\frac{\hat{Q}\hat{K}}{\sqrt{z}}\right)\hat{V}$$

17 Restore chronological order based on list A, B :

$$\hat{M}_i \in \mathbb{R}^{\frac{T}{2} \times (2N) \times z} \rightarrow \tilde{M}_i \in \mathbb{R}^{T \times N \times z}$$

18 **end**

19 Concatenate m heads outputs \tilde{M}_i into \tilde{X} .

20 **return** $\tilde{X} \in \mathbb{R}^{T \times N \times (m \times z)}$

[1, 2, 3]. We also present the complete pipeline of “leap attention” in Algorithm 1.

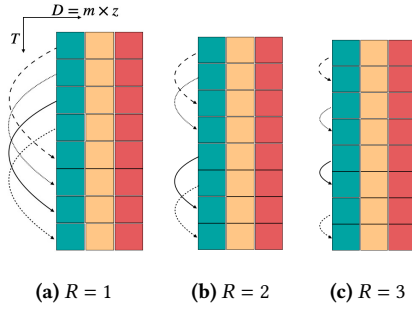


Figure 3: Given a video tensor of T frames, the skip steps of pyramid $R = 1/2/3$ range from: $[\frac{T}{2}, \frac{T}{4}, \frac{T}{8}]$. (The figure is best viewed in color)

Complexity Analysis. We theoretically compare complexities between the LA, 2D, and 3D attentions. Supposing that a clip contains T frames, with N tokens per frame, we then separate frames into g groups and apply attention to each group. The total complexity will be $g \times \left(\frac{T}{g} \times N\right)^2 = \frac{T^2 N^2}{g}$. This complexity will be: (1). TN^2 with 2D attention, when $g=T$; (2). $2TN^2$ with the LA, when $g=\frac{T}{2}$; (3). $T^2 N^2$ with 3D attention, when $g=1$. Overall, the LA introduces a small overhead to 2D attention while being obviously more efficient than 3D attention.

3.3 Periodic Shift

The P -Shift is inspired by the Temporal Shift Module (TSM) [30], originally designed for capturing short-term temporal dynamics in CNNs, but specially adjusted according to the multi-head mechanism of the transformer.

Recalling the output of multi-head attention is $\tilde{X} \in \mathbb{R}^{T \times N \times D}$ (Equation (14)), where D -dimensional feature is concatenated from m amount of z -dimensional sub-feature, namely $D = m \times z$.

Periodic Shift (P -Shift) uniformly processes the sub-feature from each head. Specifically, D channels are firstly decomposed into $m \times z$ shape; then each z -channels are split into three $\mathbf{p}^{a'}$, $\mathbf{p}^{b'}$, $\mathbf{p}^{c'}$ parts:

$$\begin{aligned} \tilde{X} &= \left[\mathbf{p}^{a'}, \mathbf{p}^{b'}, \mathbf{p}^{c'} \right] \\ \mathbf{p}^{\{a', b', c'\}} &\in \mathbb{R}^{T \times N \times m \times \{a', b', c'\}} \\ a' + b' + c' &= z \end{aligned}$$

The shift operation is conducted on of $\mathbf{p}^{a'}$, $\mathbf{p}^{b'}$ part, while keeping $\mathbf{p}^{c'}$ unchanged (Figure 1d). The t, i, u, j denotes indices for temporal, spatial, attention head, and dim (per head) axis. The “ \Leftarrow ” denotes assignment operation.

$$\mathbf{p}_{(t,i,u,j)}^{a'} \Leftarrow \mathbf{p}_{(t-1,i,u,j)}^{a'}, \quad 0 \leq j \leq a' \quad (16)$$

$$\mathbf{p}_{(t,i,u,j)}^{b'} \Leftarrow \mathbf{p}_{(t+1,i,u,j)}^{b'}, \quad 0 \leq j \leq b' \quad (17)$$

$$\begin{aligned} \mathbf{p}_{(t,i,u,j)}^{c'} &\Leftarrow \mathbf{p}_{(t,i,u,j)}^{c'}, \quad 0 \leq j \leq c' \quad (18) \\ 0 \leq t \leq T, \quad 0 \leq i \leq N, \quad 0 \leq u \leq m \end{aligned}$$

Plain Temporal Shift directly inherits the original TSM by splitting D channels into three parts:

$$\begin{aligned} \tilde{X} &= \left[\mathbf{p}^a, \mathbf{p}^b, \mathbf{p}^c \right] \\ \mathbf{p}^{\{a,b,c\}} &\in \mathbb{R}^{T \times N \times \{a,b,c\}} \\ a + b + c &= D \end{aligned}$$

where the split $\mathbf{p}^a/\mathbf{p}^b$ are shift with prior/post time-stamps along the temporal axis. The split \mathbf{p}^c remains unchanged as in Figure 1c.

Figure 1 shows that the P -Shift is less biased than the plain shift in manipulating features from different heads, thus resulting in better performances.

4 EXPERIMENTS

We evaluate the LAPS-Former on the below standard benchmarks in terms of Top-1/5 accuracy, GFLOPs, Params.

Kinetics-400 [3] is a large-scale benchmark for video classification. It defines 400 categories on $\sim 246k/20k$ training/validating clips, with 10 seconds per clip.

UCF-101 [43] serves as a small-scale datasets for classifying “human-object” interactions and sports. It contains three splits on 101 categories. Follow prior works, we report performance on the split 1, with 9,537/3,783 train/validation clips. The average clip-length is 5.8 seconds.

EGTEA Gaze+ [28] is a First-Person-View video dataset captured by wearable-device in daily life. It covers 106 action categories defined on 3.1 seconds long video clips. We adopt the split 1, which contains 8,299/2,022 training/validation clips, for evaluation and comparison.

4.1 Settings

We implant the LAPS module into Visformer [5] which is pre-trained on ImageNet data with 10k categories. All experiments follow the same recipe unless particularly specified. For experiments with variable resolutions and clip-lengths, we linearly scale the lr according to batch-size as in [15].

Training. Each clip contains 8 frames of 224×224 (320 for high resolutions) shape, with a sampling rate of 8. Augmentations, such as random resizing, horizontal flipping, and color perturbation (e.g., brightness, saturation, and hue), are adopted to reduce overfitting. The training schedule is set to 18 epochs, decaying lr decayed by 0.1 at the 10/15-th epoch. The shift ratio is fixed to $\frac{a}{D}, \frac{b}{D}, \frac{a'}{z}, \frac{b'}{z} = \frac{1}{8}$ as in TSM.

Inference. We uniformly sample 5 clips from a video, resizing their short-sides to 224. We then crop three squared sub-clips from each clip’s “left, center, right” parts. The prediction of a video is averaged over those of 15 sub-clips.

4.2 Ablation Study

We experimentally study impacts of various temporal operations (e.g. Plain Shift, P -Shift and Leap Attention), pyramid combinations, backbones, clip lengths, and resolutions on the LAPS module.

Leap Attention vs 2/3D Attention. We implement different attentions, such as generic 2/3D and Leap Attentions, in the MSHA. As in Table 1, the LA easily surpasses 2D attention (75.84% vs 74.00%), with only 2.6% computational overhead. Though the 3D attention

Model	MSHA	GFLOPs	Params (M)	Top-1 (%)
Base2D	2D Atten	39.1	39.8	74.00
Base3D	3D Atten	46.5 (18.9% ↑)	39.8	76.31
LAPS	Plain Shift	39.1	39.8	74.86
	<i>P</i> -Shift	39.1	39.8	75.19
	LA	40.1 (2.6% ↑)	39.8	75.84
	<i>P</i> -Shift + LA	40.1 (2.6% ↑)	39.8	76.04

Table 1: Impacts of temporal modules on LAPS-Former and Comparisons with 2/3D Attention baselines.

Model	Pyramids	Skipped Steps	Top-1
LAPS	R=[3, 3, 3]	$S=[\frac{1}{8}, \frac{1}{8}, \frac{1}{8}] \cdot T$	75.55
	R=[2, 2, 2]	$S=[\frac{1}{4}, \frac{1}{4}, \frac{1}{4}] \cdot T$	75.82
	R=[1, 1, 1]	$S=[\frac{1}{2}, \frac{1}{2}, \frac{1}{2}] \cdot T$	75.86
	R=[1, 2, 3]	$S=[\frac{1}{2}, \frac{1}{4}, \frac{1}{8}] \cdot T$	76.04

Table 2: Comparisons of LAPS-Former under different combinations of skipped steps (pyramids). (Top-1 Accuracy).

(76.31%) performs slightly better than LA, it suffers from an increase of 18.9% computations. Moreover, combining the LA with *P*-Shift (76.04%) approaches the performance of 3D attention, with much less costs.

Periodic Shift vs plain Shift. To eliminate the influence of attention, both models adopt simple 2D attention. As is shown in Table 1, both Shifts (74.86%, 75.19%) outperform 2D Baseline, verifying the efficacy of implanting the Shift into the attention. Besides, the *P*-Shift performs better than the plain counterpart (75.19% vs 74.86%), showing that output from each head requires to be equally shifted.

Notably, our LAPS-Former shares the same amount of parameters as the 2D Baseline, verifying that the LAPS module alone is a zero-parameter operator.

Pyramid combinations. We test LAPS with various combinations of skipped steps. Since our LAPS transformer contains eight LAs, we can set an identical or different *S* for them. For multi-pyramid settings, we loop *R* by values [1, 2, 3] with respect to the depth of LA. Additionally, a smaller *R* value indicates a larger skipped step *S* between paired frames (Equation (15)).

Table 2 compares different combinations. We observe that: (1). The LAs with multiple pyramids (76.04%) outperform single-level ones. A potential reason is that multi-scale temporal information is complementary; (2). Among all single levels, precision is positively correlated with *S*, (when $\frac{T}{8} \rightarrow \frac{T}{4} \rightarrow \frac{T}{2}$, then 75.55% \rightarrow 75.82% \rightarrow 75.86%), showing that LA prefers intaking distinct features than similar ones.

Length of tokens. We study the impact of tokens' length on LAPS-Former. Factors like clip length and spatial resolution determine this length.

Table 3 shows that LAPS benefits from both large temporal/spatial resolution. Specifically, we observe that: (1) LAPS (B) intakes a larger spatial resolution (224 \rightarrow 320) over origin LAPS, resulting in a boost of 1.52% top-1 accuracy (76.04% \rightarrow 77.56%); (2) From LAPS

Model	Base	#F×Res <i>T</i> × <i>HW</i>	GFLOPs	Paras (M)	Top-1 (%)
Base2D	Visf	8×224 ²	39.1	39.8	74.00
LAPS	Visf	8×224 ²	40.1	39.8	76.04
LAPS (B)	Visf	8×320 ²	86.4	40.0	77.56
LAPS (L)	Visf	16×320 ²	173.0	40.0	78.71
LAPS (H)	Visf	32×320 ²	346.0	40.0	79.72
LAPS (E)	Visf	32×360 ²	434.0	40.2	80.03
Base2D	RstS	8×224 ²	16.9	13.4	68.13
LAPS (RS)	RstS	8×224 ²	17.6	13.4	69.72
LAPS (RB)	RstB	8×224 ²	39.3	29.9	71.89
LAPS (RL)	RstL	32×320 ²	595.0	51.2	76.71
Base2D	ViT	8×224 ²	141.0	86.1	76.41
LAPS (VT)	ViT	8×224 ²	146.0	86.1	78.15

Table 3: Impacts of backbone, clip-length and spatial resolution on LAPS-Former (Top-1 Accuracy).

(B)-(H), precision increases (77.56% \rightarrow 78.71% \rightarrow 79.72%) with the temporal resolution (8f \rightarrow 16f \rightarrow 32f).

On various backbones. Our LAPS could be flexibly and directly plugged into backbones with a *standard* multi-head self-attention mechanism. We verify this by implanting it into 2D transformers, such as Visformer [5], ResT [65] and ViT[10]. The ResT and ViT backbone is separately pre-trained on ImageNet-1k and 22k.

Table 3 also lists performance of LAPS on 2D backbones. Our LAPS consistently boosts Baseline-2D (68.13% \rightarrow 69.72% for ResT, 76.41% \rightarrow 78.15% for ViT) while introducing small/zero extra GFLOPs or parameters. Besides, the performance is further improved with more encoders, longer clip-length, and larger resolution.

4.3 Comparison with the State-of-the-Art

In Table 4, we compare LAPS-Former with the current 3D-CNN and video transformer SOTAs in terms of training/inference efficiency (#Epochs, GFLOPs), model size (Params), precision (Top-1/5 accuracy).

LAPS vs 3D-CNNs. Firstly, like pilot transformers, the LAPS-Former shows a higher training efficiency (10 times fewer epochs) than all 3D-CNNs. An extra factor is that LAPS is built from temporal operators of zero-parameter. Thus, we could directly load model and resume training from pre-trained weights of 2D tasks. Secondly, the LAPS achieves better performance than 3D-CNNs, such as Non-Local (79.72% vs 77.7%, 346G vs 359G) and TSM (76.04% vs 74.7%, 40.1G vs 65G), with less computations. Finally, our LAPS-Former achieves better performance (80.03%) compared with strong SOTAs in the CNN era, such as SlowFast (79.8%), X3D (79.1%), with slightly more computations. The reason is that optimizations for 3D convolutions have been intensively studied and applied in designing. For instance, the SlowFast adopts 3D convolution decomposition, and the X3D is automatically searched from a base net.

LAPS vs SOTAs transformer. Our LAPS-Former (78.71%, 173G, 40M) surpasses pilot video transformers, such as TokShift (78.2%, 269.5G, 85.9M), VTN (78.6%, 4.2×10³G, 114M) and TimeFormer (78.0%, 590G, 121.4M) under all metrics. Besides, our model (80.03%) is slightly lower than strong transformers like Video Swin (80.6%)

Model	Base	Pretrain	#F×Res ($T \times HW$)	GFLOPs×Views	Params (M)	Training Epochs	Top-1 (%)	Top-5 (%)
I3D [3] from [13]	InceptionV1	IN-1K	250×224^2	$108 \times NA$	12.0	-	71.10	90.30
Two-Stream I3D [3] from [13]	InceptionV1	IN-1K	500×224^2	$216 \times NA$	25.0	-	75.70	92.00
TSM [30]	ResNet50	IN-1K	16×256^2	65.0×10	24.3	100	74.70	-
S3D-G [59]	InceptionV1	IN-1K	250×224^2	$71.3 \times NA$	11.5	112	74.70	93.40
TEA [27]	ResNet50	IN-1K	16×256^2	70.0×30	35.3	50	76.10	92.50
TEINet [33]	ResNet50	IN-1K	16×256^2	66.0×30	30.8	100	76.20	92.50
TANNet [35]	ResNet50	IN-1K	16×256^2	86.0×12	25.6	100	76.90	92.90
Non-Local R50 [51] from [13]	ResNet50	IN-1K	128×256^2	282.0×30	35.3	118	76.50	92.60
Non-Local R101 [51] from [13]	ResNet101	IN-1K	128×256^2	359.0×30	54.3	196	77.70	93.30
Small-Big [25]	ResNet101	IN-1K	32×224^2	418.0×12	-	110	77.40	93.30
TDN-R50 [50]	ResNet50	IN-1K	24×256^2	108.0×30	26.6	100	78.40	93.60
TDN-R101 [50]	ResNet101	IN-1K	24×256^2	198.0×30	43.9	100	79.40	94.40
GC-TDN-R50 [22]	ResNet50	IN-1K	24×256^2	110.1×30	27.4	100	79.60	94.10
SlowFast 8×8 [14]	ResNet50	None	32×256^2	65.7×30	-	196	77.00	92.60
SlowFast 16×8 [14]	ResNet101+NL	None	32×256^2	234.0×30	59.9	196	79.80	93.90
X3D-L [13]	X2D	None	16×356^2	24.8×30	6.1	256	77.50	92.90
X3D-XL [13]	X2D	None	16×356^2	48.4×30	11.0	256	79.10	93.90
ViT (Video) [63]	ViT-B	IN-22K	8×224^2	134.7×30	85.9	18	76.00	92.50
TokShift [63]	ViT-B	IN-22K	16×224^2	269.5×30	85.9	18	78.20	93.80
TokShift (MR) [63]	ViT-B	IN-22K	8×256^2	175.8×30	85.9	18	77.68	93.55
VTN [38]	ViT-B	IN-22K	250×224^2	4218.0×1	114.0	25	78.60	93.70
TimeSformer [1]	ViT-B	IN-22K	8×224^2	590.0×3	121.4	15	78.00	93.70
Video Swin [34]	Swin-B	IN-1K	32×224^2	281.6×12	88.0	30	80.60	94.60
MViT [11]	MViT-B	None	64×224^2	455.0×9	36.64	200	81.20	95.10
LAPS	Visformer	IN-10K	8×224^2	40.1×15	39.8	18	76.04	92.56
LAPS (L)	Visformer	IN-10K	16×320^2	173.0×15	40.0	18	78.71	93.77
LAPS (H)	Visformer	IN-10K	32×320^2	346.0×15	40.0	18	79.72	94.08
LAPS (E)	Visformer	IN-15K	32×360^2	434.0×15	40.2	18	80.03	94.48

Table 4: Comparison to state-of-the-arts on Kinetics-400 Val.

and MViT (81.2%). A possible reason might be they directly learn of cube’ embeddings in generating video sequences, whereas our LAPS-Former relies on patch embeddings. Besides, our 2D backbone (i.e., Visformer) is weaker than Swin and static MViT. However, the LAPS module still has merits, such as fewer training epochs and being more flexible than Video Swin and MViT. As our LAPS-Former can adapt to various backbones, it could potentially take advantage of newly emerging 2D transformers for 3D modeling.

4.4 Fine-tune on small-scale datasets

We further verify the effectiveness of LAPS on downstream small-scale video datasets, such as UCF-101 and EGTEA Gaze+ datasets. We initialize the training phase with pre-trained weight on the Kinetics-400 datasets to save training time and avoid overfitting. Their training/inference recipe is almost identical to the Kinetics-400 datasets, except that the training schedule is separately optimized for both datasets. Specifically, we train 18 epochs, decaying LR by 0.1 at 10/15-th epoch for UCF-101; whereas for EGTEA-Gaze+, the total epoch is set to 36, decaying LR by 0.1 at 20/30-th epoch. Experimental results for UCF-101 and EGTEA-Gaze+ is separately present in Table 5 and Table 6.

Model	Pretrain	Res ($H \times W$)	# Frames T	UCF101 Acc1 (%)
I3D [3]	K-400	224×224	250	84.50
P3D [40]	K-400	224×224	16	84.20
Two-Stream I3D [3]	K-400	224×224	500	93.40
TSM [31]	K-400	256×256	8	95.90
ViT (Video) [63]	IN-22K	256×256	8	91.46
TokShift [63]	K-400	256×256	8	95.35
TokShift-L (HR) [63]	K-400	384×384	8	96.80
LAPS	K-400	224×224	8	95.43
LAPS (H)	K-400	320×320	32	96.85

Table 5: Fine-tune on UCF-101 Split 1 set.

For the UCF-101 dataset, we observe that under a condition of similar spatial-temporal resolution ($T \times HW=8 \times 224^2$), our LAPS achieves competitive accuracy (95.43%) among 3D-CNN and Transformer pilots. With more frames and larger spatial resolutions (i.e., 32×320^2 , LAPS (H) further boosts accuracy to (96.85%), reaching SOTA performance.

We test the optimal settings for the EGTEA Gaze++ dataset, where spatial-temporal resolution is 32×320^2 . We observe the LAPS

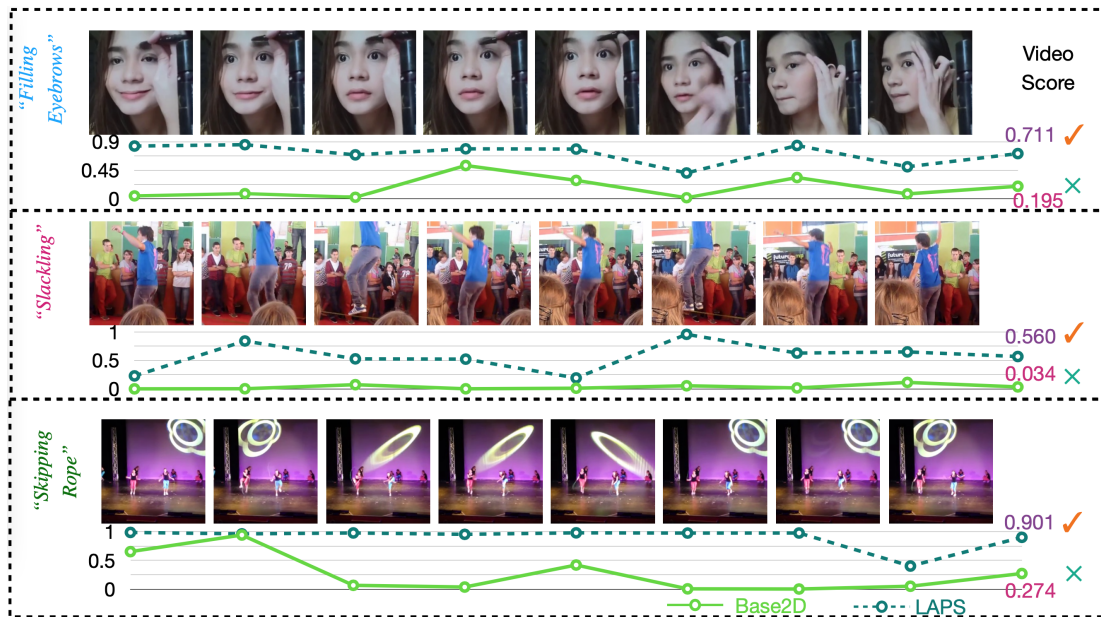


Figure 4: Visualization of Video Exemplars. We test Base2D and LAPS models on the validation set of the Kinetics-400 dataset and pick video examples with significant improvement. To highlight the impact of temporal relations, we present per-frame prediction scores and their trend along with clip frames. The solid and dashed line separately denotes predictions from Base2D and LAPS models. (The figure is best viewed in color)

Model	Pretrain	Res ($H \times W$)	# Frames T	EGAZ+ Acc1 (%)
TSM [31]	K-400	224×224	8	63.45
SAP [52]	K-400	256×256	64	64.10
ViT (Video) [10]	IN-22K	224×224	8	62.59
TokShift [63]	K-400	224×224	8	64.82
TokShift-En* [63]	K-400	224×224	8	65.08
TokShift (HR) [63]	K-400	384×384	8	65.77
LAPS (H)	K-400	320×320	32	66.07

Table 6: Fine-tune on EGTEA-GAZE++ Split-1 dataset. (* indicates using RGB+Optical Flow modalities)

(H) could achieve SOTA performance (66.07%) among transformers and 3D-CNNs, verifying the robustness and generalization capacity of the LAPS model.

4.5 Visualization of Exemplars with Significant Improvement

To verify the impact of the LAPS temporal process, we compare the Base2D and LAPS model predictions on the validation set of the Kinetics-400 dataset. As video prediction is averaged over frame-level predictions (e.g. Eq. 6), we present per-frame prediction of corresponding category to highlight the impact of the temporal process. We pick and compare video clips that are poorly predicted when frames are individually predicted (Base2D), but are corrected when connecting temporal relations (LAPS). Clip exemplars are shown in Figure 4.

We observe that: (1) By connecting long, short-term temporal relations, frame-level predictions are smoother than treating each frame individually (e.g., clips with “Filling Eyebrows” and “Skipping Rope”), which is consistent with what we envisioned; (2) The LAPS module is more robust to failure on particular frames (e.g. 1st/5th frame of clip “Slackling”), and yield correct video-level prediction.

5 CONCLUSION

We propose a zero-parameter LAPS module for video transformers. Specifically, the LAPS module is constructed from the Leap Attention and Periodic Shift sub-modules. It can flexibly replace an image transformer’s generic 2D attention part with neglectable computational overhead and convert it into a video transformer. Like existing video transformers, our LAPS could not intake a very long video sequence (> 200 frames) in an end-to-end training manner, even though it is pretty efficient. But the dilated manner in temporal attention could be extended to the spatial domain to compress computations further, which will step toward long-video processing in the future.

ACKNOWLEDGMENTS

The work present in this paper is partially supported by the National Natural Science Foundation of China (Grant No. 62101524), by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier-1 grant, by the Zhejiang Provincial Natural Science Foundation of China (LQ21F020003), by the Exploratory Research Project (2022PG0AN01).

REFERENCES

- [1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. 2021. Is Space-Time Attention All You Need for Video Understanding?. In *Proceedings of the International Conference on Machine Learning (ICML)*.
- [2] Yi Bin, Xindi Shang, Bo Peng, Yujian Ding, and Tat-Seng Chua. 2021. Multi-Perspective Video Captioning. In *Proceedings of the 29th ACM International Conference on Multimedia*. 5110–5118.
- [3] Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6299–6308.
- [4] Kai Chen, Zhipeng Wei, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. 2022. Attacking Video Recognition Models with Bullet-Screen Comments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 312–320.
- [5] Zhengsu Chen, Lingxi Xie, Jianwei Niu, Xuefeng Liu, Longhui Wei, and Qi Tian. 2021. Visformer: The vision-friendly transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 589–598.
- [6] Zhi-Qi Cheng, Yang Liu, Xiao Wu, and Xian-Sheng Hua. 2016. Video ecommerce: Towards online video advertising. In *Proceedings of the 24th ACM international conference on Multimedia*. 1365–1374.
- [7] Zhi-Qi Cheng, Xiao Wu, Yang Liu, and Xian-Sheng Hua. 2017. Video ecommerce++: Toward large scale online video advertising. *IEEE transactions on multimedia* 19, 6 (2017), 1170–1183.
- [8] Zhi-Qi Cheng, Xiao Wu, Yang Liu, and Xian-Sheng Hua. 2017. Video2shop: Exact matching clothes in videos to online shopping images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4048–4056.
- [9] Zhi-Qi Cheng, Hao Zhang, Xiao Wu, and Chong-Wah Ngo. 2017. On the selection of anchors and targets for video hyperlinking. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*. 287–293.
- [10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiuhua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of International Conference on Learning Representations (ICLR)* (2021).
- [11] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. 2021. Multiscale vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 6824–6835.
- [12] Linxi Fan*, Shyamal Buch*, Guanzhi Wang, Ryan Cao, Yuke Zhu, Juan Carlos Niebles, and Li Fei-Fei. 2020. RubiksNet: Learnable 3D-Shift for Efficient Video Action Recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
- [13] Christoph Feichtenhofer. 2020. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 203–213.
- [14] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. 2019. Slow-fast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*. 6202–6211.
- [15] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677* (2017).
- [16] Jianyuan Guo, Kai Han, Han Wu, Yehui Tang, Xinghao Chen, Yunhe Wang, and Chang Xu. 2022. Cmt: Convolutional neural networks meet vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12175–12185.
- [17] Ning Han, Jingjing Chen, Guangyi Xiao, Hao Zhang, Yawen Zeng, and Hao Chen. 2021. Fine-grained cross-modal alignment network for text-video retrieval. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3826–3834.
- [18] Ning Han, Jingjing Chen, Hao Zhang, Huanwen Wang, and Hao Chen. 2022. Adversarial Multi-Grained Embedding Network for Cross-Modal Text-Video Retrieval. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 18, 2 (2022), 1–23.
- [19] Yanbin Hao, Zi-Niu Liu, Hao Zhang, Bin Zhu, Jingjing Chen, Yu-Gang Jiang, and Chong-Wah Ngo. 2020. Person-level Action Recognition in Complex Events via TSD-TSM Networks. In *Proceedings of the 28th ACM International Conference on Multimedia*. 4699–4702.
- [20] Yanbin Hao, Chong-Wah Ngo, and Benoit Huet. 2019. Neighbourhood structure preserving cross-modal embedding for video hyperlinking. *IEEE Transactions on Multimedia* 22, 1 (2019), 188–200.
- [21] Yanbin Hao, Chong-Wah Ngo, and Bin Zhu. 2021. Learning to match anchor-target video pairs with dual attentional holographic networks. *IEEE Transactions on Image Processing* 30 (2021), 8130–8143.
- [22] Yanbin Hao, Hao Zhang, Chong-Wah Ngo, and Xiangnan He. 2022. Group Contextualization for Video Recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- [23] Yanbin Hao, Hao Zhang, Chong-Wah Ngo, Qiang Liu, and Xiaojun Hu. 2020. Compact Bilinear Augmented Query Structured Attention for Sport Highlights Classification. In *Proceedings of the 28th ACM International Conference on Multimedia*. 628–636.
- [24] Jun-Yan He, Xiao Wu, Zhi-Qi Cheng, Zhaoquan Yuan, and Yu-Gang Jiang. 2021. DB-LSTM: Densely-connected Bi-directional LSTM for human action recognition. *Neurocomputing* 444 (2021), 319–331.
- [25] Xianhang Li, Yali Wang, Zhipeng Zhou, and Yu Qiao. 2020. Smallbignet: Integrating core and contextual views for video classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1092–1101.
- [26] Xirong Li, Chaoyi Xu, Gang Yang, Zhineng Chen, and Jianfeng Dong. 2019. W2vv++ fully deep learning for ad-hoc video search. In *Proceedings of the 27th ACM international conference on multimedia*. 1786–1794.
- [27] Yan Li, Bin Ji, Xintian Shi, Jianguo Zhang, Bin Kang, and Limin Wang. 2020. Tea: Temporal excitation and aggregation for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 909–918.
- [28] Yin Li, Miao Liu, Rehg, and James M. 2018. In the eye of beholder: Joint learning of gaze and actions in first person video. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 619–635.
- [29] Yehao Li, Yingwei Pan, Jingwen Chen, Ting Yao, and Tao Mei. 2021. X-modaler: A Versatile and High-performance Codebase for Cross-modal Analytics. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3799–3802.
- [30] Ji Lin, Chuang Gan, and Song Han. 2019. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7083–7093.
- [31] Ji Lin, Chuang Gan, and Song Han. 2019. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7083–7093.
- [32] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10012–10022.
- [33] Zhaoyang Liu, Donghao Luo, Yabiao Wang, Limin Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Tong Lu. 2020. Teinet: Towards an efficient architecture for video recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 11669–11676.
- [34] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. 2022. Video swin transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3202–3211.
- [35] Zhaoyang Liu, Limin Wang, Wayne Wu, Chen Qian, and Tong Lu. 2021. Tam: Temporal adaptive module for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13708–13718.
- [36] Jianjie Luo, Yehao Li, Yingwei Pan, Ting Yao, Hongyang Chao, and Tao Mei. 2021. CoCo-BERT: Improving video-language pre-training with contrastive cross-modal matching and denoising. In *Proceedings of the 29th ACM International Conference on Multimedia*. 5600–5608.
- [37] Zhixin Ma, Jiaxin Wu, Zhijian Hou, and Chong-Wah Ngo. 2022. Reinforcement Learning-Based Interactive Video Search. In *International Conference on Multimedia Modeling*. Springer, 549–555.
- [38] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselmann. 2021. Video transformer network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3163–3172.
- [39] Phuong Anh Nguyen, Qing Li, Zhi-Qi Cheng, Yi-Jie Lu, Hao Zhang, Xiao Wu, and Chong-Wah Ngo. 2017. Vireo@ trecvid 2017: Video-to-text, ad-hoc video search and video hyperlinking. (2017).
- [40] Zhaofan Qiu, Ting Yao, and Tao Mei. 2017. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*. 5533–5541.
- [41] Zhaofan Qiu, Ting Yao, Chong-Wah Ngo, and Tao Mei. 2022. MLP-3D: A MLP-Like 3D Architecture With Grouped Time Mixing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3062–3072.
- [42] Hao Shao, Shengju Qian, and Yu Liu. 2020. Temporal interlacing network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 11966–11973.
- [43] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012).
- [44] Swathikiran Sudhakaran, Sergio Escalera, and Oswald Lanz. 2020. Gate-shift networks for video action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1102–1111.
- [45] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.
- [46] Yi Tan, Yanbin Hao, Xiangnan He, Yinwei Wei, and Xun Yang. 2021. Selective dependency aggregation for action classification. In *Proceedings of the 29th ACM International Conference on Multimedia*. 592–601.
- [47] Mingkang Tang, Zhanyu Wang, Zhenhua Liu, Fengyuan Rao, Dian Li, and Xiu Li. 2021. Clip4caption: Clip for video caption. In *Proceedings of the 29th ACM International Conference on Multimedia*. 4858–4862.

- [48] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 4489–4497.
- [49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.
- [50] Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. 2021. Tdn: Temporal difference networks for efficient action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1895–1904.
- [51] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7794–7803.
- [52] Xiaohan Wang, Yu Wu, Linchao Zhu, and Yi Yang. 2020. Symbiotic attention with privileged information for egocentric action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 12249–12256.
- [53] Zhipeng Wei, Jingjing Chen, Micah Goldblum, Zuxuan Wu, Tom Goldstein, and Yu-Gang Jiang. 2022. Towards transferable adversarial attacks on vision transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 2668–2676.
- [54] Zhipeng Wei, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. 2022. Boosting the Transferability of Video Adversarial Examples via Temporal Translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 2659–2667.
- [55] Zhipeng Wei, Jingjing Chen, Zuxuan Wu, and Yu-Gang Jiang. 2022. Cross-Modal Transferable Adversarial Attacks from Images to Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15064–15073.
- [56] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. 2021. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 22–31.
- [57] Jiaxin Wu and Chong-Wah Ngo. 2020. Interpretable embedding for ad-hoc video search. In *Proceedings of the 28th ACM International Conference on Multimedia*. 3357–3366.
- [58] Jiaxin Wu, Phuong Anh Nguyen, Zhixin Ma, and Chong-Wah Ngo. 2021. SQL-like interpretable interactive video search. In *International Conference on Multimedia Modeling*. Springer, 391–397.
- [59] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. 2018. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 305–321.
- [60] Fisher Yu and Vladlen Koltun. 2016. Multi-scale context aggregation by dilated convolutions. In *Proceedings of International Conference on Learning Representations (ICLR) (2016)*.
- [61] Yitian Yuan, Lin Ma, Jingwen Wang, and Wenwu Zhu. 2020. Controllable video captioning with an exemplar sentence. In *Proceedings of the 28th ACM International Conference on Multimedia*. 1085–1093.
- [62] Yawen Zeng. 2022. Point Prompt Tuning for Temporally Language Grounding. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2003–2007.
- [63] Hao Zhang, Yanbin Hao, and Chong-Wah Ngo. 2021. Token shift transformer for video classification. In *Proceedings of the 29th ACM International Conference on Multimedia*. 917–925.
- [64] Hao Zhang and Chong-Wah Ngo. 2018. A fine granularity object-level representation for event detection and recounting. *IEEE Transactions on Multimedia* 21, 6 (2018), 1450–1463.
- [65] Qinglong Zhang and Yu-Bin Yang. 2021. ResT: An efficient transformer for visual recognition. *Advances in Neural Information Processing Systems* 34 (2021), 15475–15485.
- [66] Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. 2022. CenterCLIP: Token Clustering for Efficient Text-Video Retrieval. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, July 11–15, 2022, Madrid, Spain*.