applied sciences

MDPI

Article

Dilated Multi-Temporal Modeling for Action Recognition

Tao Zhang, Yifan Wu and Xiaoqiang Li *

School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China; tod_zhang@shu.edu.cn (T.Z.); victorwu@shu.edu.cn (Y.W.)
* Correspondence: xqli@shu.edu.cn

Abstract: Action recognition involves capturing temporal information from video clips where the duration varies with videos for the same action. Due to the diverse scale of temporal context, uniform size kernels utilized in convolutional neural networks (CNNs) limit the capability of multiple-scale temporal modeling. In this paper, we propose a novel dilated multi-temporal (DMT) module that provides a solution for modeling multi-temporal information in action recognition. By using dilated convolutions with different dilation rates in different feature map channels, the DMT module captures information at multiple scales without the need for costly multi-branch networks, input-level frame pyramids, or feature map stacking that previous works have usually incurred. Therefore, this approach enables the integration of temporal information from multiple scales. In addition, the DMT module can be integrated into existing 2D CNNs, making it a straightforward and intuitive solution for addressing the challenge of multi-temporal modeling. Our proposed method has demonstrated promising results in performance and has achieved about 2% and 1% accuracy improvement on FineGym99 and SthV1. We conducted an empirical analysis that demonstrates how DMT improves the classification accuracy for action classes with varying durations.

Keywords: computer vision; action recognition; multiple temporal modeling; dilated convolution

Citation: Zhang, T.; Wu, Y.; Li, X. Dilated Multi-Temporal Modeling for Action Recognition. *Appl. Sci.* **2023**, *13*, 6934. https://doi.org/10.3390/app13126934

Academic Editor: Steven Davy

Received: 10 May 2023
Revised: 1 June 2023
Accepted: 6 June 2023
Published: 8 June 2023

1. Introduction

Action recognition is a crucial task in the field of computer vision, as it allows for the automatic identification of actions and behaviors in video sequences. In this task, temporal modeling is used to capture the motion information in a video, which is essential for accurately identifying the actions being performed. Several methods have been proposed for temporal modeling, including two-stream networks [1], 3D CNNs [2,3], and 2D + 1D paradigms [4–6]. Such methods based on CNNs have gained widespread use in action recognition due to their ability to effectively extract both spatial and temporal features from videos. However, the use of uniform-size kernels in CNNs limits their ability to capture temporal features, due to the inherent variability in the speed and duration of actions, because uniform-size kernels are fixed in size and do not take into account the varying duration of actions.

Such a phenomenon of inconsistency in duration and speed often observed in action videos can be illustrated by considering a video of a person running. If a person is running at fast pace, the duration of the video will be shorter compared to a video of the same person running at slow pace. This is because the faster the person is running, the less frames that are captured in the same distance, resulting in a shorter frame duration. This inconsistency of duration and speed poses a challenge for action recognition algorithms. Previous efforts have utilized different strategies to model diverse temporal information. Certain approaches [7–9] use a sequence of frames captured at different intervals to model diverse temporal information. SlowFast [7] also samples the frames at two different rates and feeds them into a slow path and a fast path for capturing both the slow and fast tempo. TPN [10] aggregates information from different visual temporal sources into a pyramid

structure by stacking feature maps. These methods, building multi-branch networks, constructing frame pyramids, and stacking feature maps, have employed costly strategies.

In this paper, we propose a dilated multi-temporal (DMT) module offering a novel solution to modeling multi-temporal information without the need for multi-branch networks, input-level frame pyramids, or stacking feature maps, which can be integrated into the existing backbone architecture. As demonstrated in [11], dilated convolution [12] can effectively increase the receptive field size without adding additional parameters or computations. Thus, we argue that time-dilated convolutions can also extend the receptive field in the temporal dimension and the use of dilated CNNs can extract the long-range temporal features. Furthermore, this characteristics of dilation motivates us to adopt CNNs with different dilation rates to model different temporal information—small rates for shorter times and bigger rates for longer times. As depicted in the rightmost part of Figure 1, the DMT module utilizes dilated convolutions with different dilation rates in the different channels of the feature maps in order to capture information at multiple scales. Additionally, we employ a strategy with low computational overhead to efficiently merge temporal information at different scales. This enables our model to integrate information from multiple scales in a manner that preserves relevant details while minimizing computational requirements. This is particularly useful for applications where computational efficiency is a priority, such as real-time video action analysis or large-scale data processing.

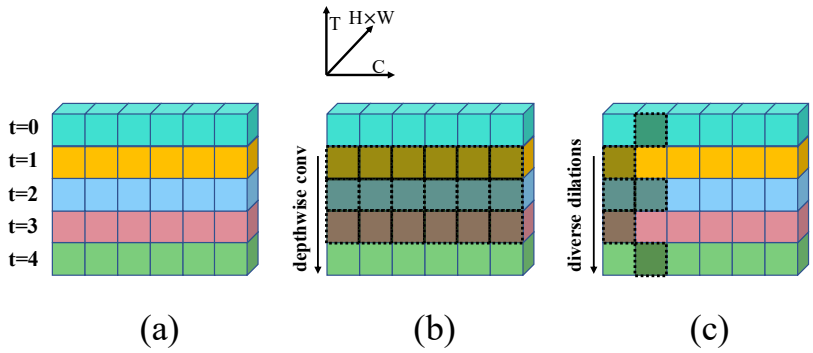

Figure 1. (**a**) The original feature maps without temporal modeling. (**b**) Performing convolution along the temporal dimension using depthwise convolutions with kernels of stationary size. (**c**) Performing convolution along the temporal dimension utilizing different dilation rates across different sections of the feature map channels. This technique allows for an increased receptive field and for the ability to capture multiple temporal information.

We evaluate the performance of the proposed DMT on the task of video-based action classification. The proposed approach leads to enhanced performance on several benchmark datasets, such as FineGym99 [13], Something–Something V1(SthV1) [14], and Kinetics400 [15]. In addition, we conduct ablation experiments on FineGym99 to further validate the effectiveness of our approach. The dataset was chosen because it contains fine-grained action classes with substantial variance in their durations, making it an ideal candidate for evaluating the ability of models for multi-temporal modeling. The experiments demonstrate the ability of the DMT module to perform accurate action recognition. Overall, this work contributes to the field of action recognition by providing a practical solution for modeling multi-temporal information in action recognition.

The main innovations of our work can be summarized as follows:

- The DMT module allows for modeling of multi-temporal information without the need for multi-branch networks or input-level frame pyramids.
- In different channels, dilated convolutions with different rates enable modeling of different temporal scales.
- Partial channel modeling of temporal information enables the integration of information from multiple scales while minimizing extra computational requirements.

2. Related Work

2.1. Action Recognition

Action recognition is a widely studied problem in the field of computer vision, and numerous approaches have been proposed to tackle it. Traditional approaches [16,17] were mainly based on hand-crafted features, but the recent success of deep learning has led to a shift toward end-to-end learning methods [18,19]. Among them, convolutional neural networks (CNNs) have been widely adopted due to their ability to effectively extract spatial and temporal features from videos. To capture temporal information in videos, several CNN-based architectures have been proposed, including two-stream networks [1,20,21], 3D CNNs [2,3,15], and 2D + 1D paradigms [5,22].

Two-stream networks involve using separate CNNs to process optical flow and RGB frames. The results of these two CNNs are then combined through late fusion. This method has been adopted in numerous action-recognition methods, such as in TSN [23]. TSN extracts short optical flow frames over a fixed number of segments and one random RGB frame from each segment to aggregate spatiotemporal information. Furthermore, 3D-CNN models extend the 2D models used in image recognition by employing 3D convolutions to extract features. I3D [15] inflates 2D filters into 3D to take advantage of the learned parameters of 2D ConvNets trained on the ImageNet [24] dataset. Other 3D-CNN based approaches, such as P3D [4], S3D [6], and R(2 + 1)D [5], separate spatial and temporal convolution to achieve a balance between efficiency and accuracy. For instance, R(2 + 1)D explicitly factorizes 3D convolution into two separate and successive operations: a 2D spatial convolution and a 1D temporal convolution. There are also several techniques that have designed plug-in modules for 2D CNNs to achieve high efficiency, particularly for temporal modeling. TSM [25], for example, performs temporal modeling with zero computation and zero parameters by shifting part of the channels along the temporal dimension to facilitate information exchange among neighboring frames.

2.2. Multi-Temporal Modeling

Accurately recognizing actions from videos can be challenging due to the diverse duration of action instances. This challenge has led to a line of research focused on multiple temporal modeling, which aims to capture different motion information in videos. Temporal modeling techniques can help to overcome the negative impact of variable action durations on recognition accuracy by capturing the temporal dependencies of action instances. By improving the ability to capture different temporal information, multiple temporal modeling has the potential to significantly improve the accuracy of action-recognition systems. Some strategies have been proposed to model diverse temporal information in videos for action recognition. Despite the recent advancements in action recognition, multi-scale temporal modeling is still an underdeveloped area of research. In recent years, there have been some efforts to address this challenge and to develop multi-scale temporal modeling techniques that can capture the temporal relationships across different scales of actions.

For example, the temporal pyramid network (TPN) [10] models the visual tempos of different actions at the feature level, and it can be integrated with mainstream backbones to capture a variety of temporal information of action instances. SlowFast [7] employs two pathways—a slow pathway and a fast pathway—with different temporal speeds to capture spatiotemporal features. These pathways operate at different frame rates and are fused through lateral connections, allowing them to effectively model multi-temporal information. The dynamic temporal pyramid network (DTPN) [9] utilizes a pyramidal representation with varying frame sample rates to address the inherent temporal scale variation in video understanding. While these strategies—such as stacking multiple stages of features, sampling at different rates, and utilizing dual networks—can improve the ability to model multi-scale temporal context, they also incur increased, costly, multi-branch networks. Therefore, in our approach, we model a multi-scale temporal context without relying on stacked pyramids or multi-frequency sampling within a single backbone.

3. Methods

3.1. Overview

The DMT module is a type of modular component that can be seamlessly integrated into 2D CNN for the purpose of modeling multiple temporal information. The initial step of the module involves utilizing an attention block to regulate the importance of feature maps. Then, the feature maps are separated along the channel dimension for modeling different temporal scales. Within each part, depthwise convolutions with varying dilation rates are applied. To fuse the temporal information across different parts, the resulting feature maps are concatenated along the channel dimension. The structure of the network is outlined and its internal formulation is detailed as follows.

3.2. Module Design

Motivated by SENet [26] and TAM [27], the DMT module first learns attention weights for each temporal and channel of feature maps, effectively acting as an attention mechanism that focuses on both the temporal and channel dimensions of the feature maps. This allows the DMT module to attend to relevant features and enhances its ability to capture important temporal and channel information.

We adopt the same strategy and configuration with TAM to generate the modulation weights, as illustrated in Figure 2.

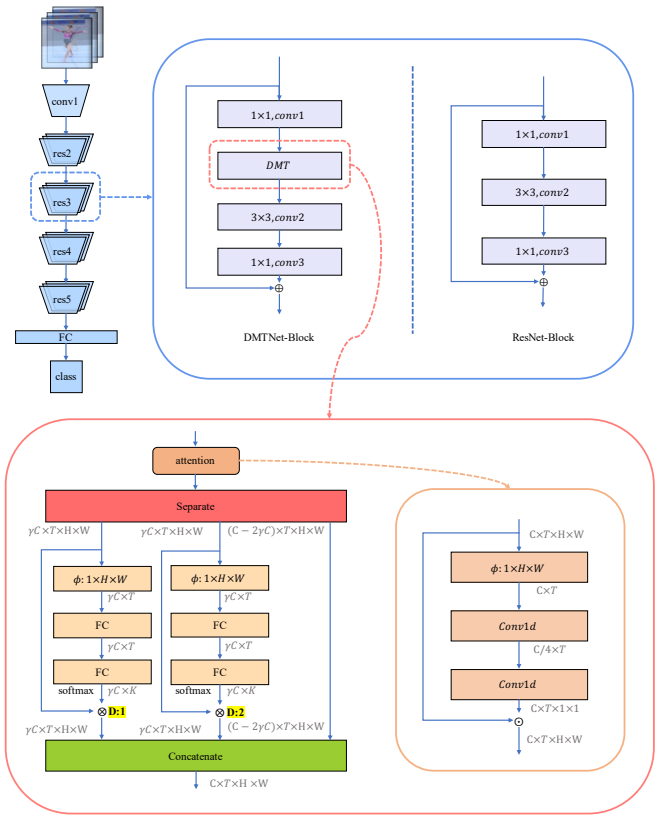

Figure 2. The structure of DMTNet and the DMT module. The top of the figure showcases the structure of the backbone network, including its inner components. Additionally, it displays a comparison between DMTNet-block and ResNet-block. The workflow, as depicted in the bottom of the figure, illustrates the sequence of operations implemented in the module. The element-wise addition operation is denoted by $\oplus$, the element-wise multiplication operation is denoted by $\odot$, and the convolution operation is denoted by $\otimes$. The variable K represents the size of the generated kernel, which is set to 3 in the following implementation. It is worth emphasizing the significant role played by convolution operations with varying dilation rates. In the figure, the dilation is represented by the letter "D" and is highlighted in yellow, with dilation rates set to 1 and 2, respectively. The primary

goal of this module is to partition the feature map into distinct sections along the channel dimension and to subsequently use dilated convolution to model the multiple temporal information within these partitions before finally concatenating them to fuse the captured information.

3.2.1. Multiple Dilations

It is intuitive to utilize different receptive filed blocks to capture the multi-scale temporal context: small ones for short context and large ones for long context. However, directly improving the size of the kernel to capture long context will induce the problem of high computational cost and massive parameters. Based on [11] , dilated convolutions effectively increase the receptive field size without added parameters or extra computation. Thus, we argue that time-dilated convolutions can also extend the receptive field in the temporal dimension and can use dilated CNNs to extract the long-range temporal features. Furthermore, this characteristic of dilation motivates us to adopt CNNs with different dilation rates to model different temporal information—small rates for short times and bigger rates for long times.

As shown in Figure 3, two convolutional operations—depthwise convolutions—with different dilation rates in the temporal dimension aggregate different temporal information. Such operations can be describe as follows:

$$Y = Conv(X, K, D), \tag{1}$$

where $Conv$ denotes the convolutional operation, X is the input, and D means the dilation rate of the convolution. K is a dynamic kernel for each input X generated by the global branch module [27]:

$$K = G(X) = softmax(f(W_2, ReLU(f(W_1, \Phi(X))))), \tag{2}$$

where Φ denotes the function that aggregates the spatial information by pooling, f is the fully connected layers, and G denotes the global module, which generates an adaptive kernel based on the whole temporal information.

Thus, we can describe it as follows:

$$Y = Conv(X, G(X), D). \tag{3}$$

We can set different dilation rates, namely D, to model the different temporal scale information without increasing the size of the kernel. A small dilation rate 1 is used to capture short-range temporal information, while a large dilation rate 2 is used to capture long-range temporal information.

Multiple scale temporal information is extracted using dilated convolutions with varied dilation. This allows the network to capture information at different scales, with each scale focusing on a different rate of motion. For example, a large-scale receptive field will be sensitive to slow movements, while a small one will be more responsive to rapid changes in the scene. By combining the multi-temporal information from different scales, the network can effectively capture the dynamic motion of the object over time.

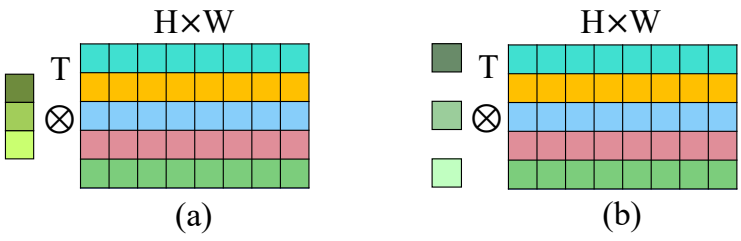

Figure 3. Multiple dilations. The utilization of dilated convolutions allows for an increase in the receptive field size without incurring additional cost. DMT employs different dilation rates for the

dilated convolutions to capture various temporal information. Specifically, (**a**) a small dilation rate of 1 is used to capture fine-grained temporal information, while (**b**) a larger dilation rate of 2 is utilized to capture more coarse-grained temporal information.

3.2.2. Partial Channel Modeling

We employed an partial channel modeling approach to efficiently integrate temporal information at varying scales. This enables our model to combine the information from multiple scales in a way that retains the relevant details while minimizing the amount of additional computation required. This is particularly useful for applications where computational efficiency is a priority, such as real-time video action analysis or large-scale data processing.

The partial channel modeling strategy involves separating the feature map along the channel dimension into different parts for the purpose of modeling different-scale temporal information. Partial channels are utilized to extract multiple-scale temporal information using dilated convolutions with varied dilation rates. The utilization of partial channels in the extraction of multiple-scale temporal information, as opposed to utilizing multi-blocks, is an efficient strategy that does not bring additional computations.

We separate the feature map into three parts in the channel axis:

$$X_1, X_2, X_3 = Separate(X, \gamma), \tag{4}$$

where *Separate* denotes the operation that divides the input by the channel dimension, and γ is a hyperparameter that divides the feature map into three parts: $X_1 = X_{[0:\gamma C)}$, $X_2 = X_{[\gamma C:2\gamma C)}$, $X_3 = X_{[2\gamma C:C)}$.

The first and second parts focus on capturing short and long temporal semantic information, respectively. The third part is used to balance the trade-offs between the other parts and to maintain a good overall representation of the original input. This separation is performed through depth-wise convolutional operations on the first and second parts with different dilation rates as described above, without any operation on the third part, followed by concatenation. These operations can be described as follow:

$$\begin{aligned} Y_1 &= Conv(X_1, G_1(X_1), D_1), \\ Y_2 &= Conv(X_2, G_2(X_2), D_2), \end{aligned} \tag{5}$$

where G_1 and G_2 are two branches with the same structure to generate dynamic kernels for the following convolution; D_1 and D_2 are the dilation rates 1 and 2 for the convolution.

It is simply concatenated with the first, second and third parts in the channel axis to keep the original shape of the feature map:

$$Y = Concatenate(Y_1, Y_2, X_3). \tag{6}$$

The resulting feature maps are then fed into subsequent modules of the network for further processing.

In summary, the use of different dilated convolutions allows the network to capture information at multiple scales, each one of them focusing on a different rate of motion, while retaining relevant details and minimizing computational cost compared with using multi-block modeling. This strategy is useful for applications where computational efficiency is a priority, such as real-time video analysis or large-scale data processing.

4. Results

We evaluated the proposed method for action recognition on three datasets: FineGym99 [13], Something–Something V1 (SthV1) [14], and Kinetics400 [15]. To further verify the ability of the method to model multi-temporal information, we conducted an ablation study and empirical analysis on FineGym99. The results of our study indicate an improvement in accuracy, demonstrating the efficacy of the DMT module in capturing

multi-temporal information for action recognition tasks. This highlights the potential of this approach to effectively integrate information at varying temporal scales. All experiments were conducted using the MMaction2 [28] framework to ensure a fair comparison.

4.1. Datasets

FineGym99 consists of approximately 20 K training and 8 K validation annotated video clips of various lengths, covering 99 action classes drawn from gymnastic videos. SthV1 is a video dataset annotated with 1 of 174 action classes, which is split into 86,000 training videos and 11,000 validation videos, and the durations of the videos vary from 2 to 6 s. Kinetics400, a large-scale video-action recognition dataset, contains around 240 K training and 19 K validation videos that last for about 10 s, which includes 400 action categories in total. These datasets are valuable in evaluating the generalization and robustness of action-recognition algorithms, providing a comprehensive testbed for performance analysis.

4.2. Training and Inference

4.2.1. Training

In our experiments, we utilized pre-trained weights from ImageNet as the initial weights for our models. This approach is commonly used in deep learning as a means of transferring knowledge from a pre-trained model to a new task. To ensure that the models were well suited for action recognition, we trained them using both 8 and 16 frames as inputs, allowing us to evaluate the impact of input length on performance. For the FineGym99 and SthV1 datasets, we sampled frames at equal intervals from all the video clips. For the Kinetics400 dataset, following the practice in [29], the frames were sampled from 64 consecutive frames in the video. To augment the data, we applied techniques such as resizing the shorter side of the frames to 256, applying multi-scale cropping, and randomly flipping the frames horizontally. The resulting cropped frames were then resized to 224 for network training. For FineGym99, we trained the models for 50 epochs with an initial learning rate of 0.002, which was decreased by a factor of 10 at the 40th epoch. For SthV1, we trained the models for 50 epochs with an initial learning rate of 0.01, which was decreased by a factor of 10 at the 20th and 40th epochs. For Kinetic400, we trained the models for 100 epochs with an initial learning rate of 0.01, which was decreased by a factor of 10 at the 50th, 75th, and 90th epochs. To prevent overfitting, we used SGD [30] with a momentum of 0.9 and a weight decay of 1×10^{-3} for SthV1 (weight decay of 1×10^{-4} for FineGym99 and Kinetics400) during training.

4.2.2. Inference

In order to fairly compare our model with other methods, we used different inference schemes for the FineGym99, SthV1, and Kinetics400 datasets. For the FineGym and SthV1 datasets, we adopted a strategy that balances efficiency and accuracy in the inference process. Specifically, we utilized a center crop of 224×224 in the spatial dimension and performed single sampling in the time dimension to enable efficient inference. Additionally, we employed three crops of 256×256 in the spatial dimension and carried out double sampling in the time dimension to achieve accurate inference. This approach allows us to effectively balance the trade-off between computational efficiency and recognition accuracy. For the Kinetics400 dataset, we uniformly sampled 10 temporal clips, each with three spatial crops of 256×256. We evaluated our model's performance on the validation set of Kinetics400.

4.3. Main Results

4.3.1. Result on FineGym99

In our experiments on FineGym99, as shown in Table 1, DMTNet achieved state-of-the-art performance, outperforming the other methods. Despite only using RGB frames as input, DMTNet outperformed TSM that uses RGB and optical flow frames. To ensure efficiency and fair comparisons with the other models, we used a one-clip and one-crop

evaluation strategy for DMTNet. However, to further demonstrate the high accuracy of our method, we also adopted a two-sample strategy that used two clips and three crops for evaluation. These strategies allowed us to effectively evaluate the performance of DMTNet on FineGym99 and to compare it with other state-of-the-art approaches. The results of our experiments clearly demonstrate the effectiveness and efficiency of MDTNet for action recognition on FineGym99. Our method achieved superior performance without relying on additional optical flow frames, making it a highly competitive and effective approach for action-recognition tasks.

Table 1. FineGym99 result with other methods. These results that were not reported in the original articles have surfaced in a subsequent study FineGym99 [13], and the results of TPN [10] and TAM [27] are reproduced by us in MMactions2 [28], with the same evaluation metrics in eight frames.

Methods	Backbone	Pretrained	Modality	Mean	Top-1
ActionVLAD [31]	VGG-16	ImageNet	RGB	50.1	69.5
TSN [23]	BNInception	ImageNet	RGB	61.4	74.8
TRN [32]	BNInception	ImageNet	RGB	68.7	79.9
TRNms [32]	BNInception	ImageNet	RGB	68.8	79.5
TSM [25]	ResNet-50	ImageNet	RGB	70.6	80.4
I3D [15]	ResNet-50	ImageNet	RGB	63.2	74.8
I3D * [15]	ResNet-50	Kinetics-400	RGB	64.4	75.6
NL I3D [29]	ResNet-50	ImageNet	RGB	62.1	73.0
NL I3D * [29]	ResNet-50	Kinetics-400	RGB	64.3	75.3
TPN [10]	ResNet-50	ImageNet	RGB	53.3	75.0
TANet [27]	ResNet-50	ImageNet	RGB	80.6	85.8
STPG-Net (TSN) [33]	ResNet-50	ImageNet	RGB	83.4	88.6
STPG-Net (I3D) [33]	ResNet-50	ImageNet	RGB	82.6	87.9
TSN [23]	BNInception	ImageNet	Flow	75.6	84.7
TRN [32]	BNInception	ImageNet	Flow	77.2	85.0
TRNms [32]	BNInception	ImageNet	Flow	77.6	85.5
TSM [25]	ResNet-50	ImageNet	Flow	80.3	87.1
TSN [23]	BNInception	ImageNet	2Stream	76.4	86.0
TRN [32]	BNInception	ImageNet	2Stream	79.8	87.4
TRNms [32]	BNInception	ImageNet	2Stream	80.2	87.8
TSM [25]	ResNet-50	ImageNet	2Stream	81.2	88.4
DMTNet	ResNet-50	ImageNet	RGB	83.0	87.5
DMTNet [1]	ResNet-50	ImageNet	RGB	84.0	89.3

* I3D model without the incorporation of temporal downsampling. [1] The double sampling which use two clips and three crops, each with a resolution of 256 × 256, can improve accuracy.

4.3.2. Results on Something–Something V1

As Shown in Table 2, our method achieves competitive accuracy compared with the other models on SthV1. For fair comparison, we show the results by taking a single clip with a center crop as input. In order to gain further insight into the capabilities of DMTNet, we conducted additional evaluations by sampling two temporal clips with three spatial crops. These two strategies adopted in the evaluation can both demonstrate the efficiency and accuracy of DMTNet. DMTNet achieved a 1% improvement in performance with 8 frames and a 1.8% improvement with 16 frames compared to TANet, which is our baseline that also uses the same attention strategy and adoptive kernel generating for depth-wise convolution. TEFE [34] showed competitiveness at the 16-frame sampling, but DMTNet exhibited significant results for real-time applications such as for 8-frame rapid inference. The results were obtained by testing the validation set.

Table 2. Something–Something V1 results with other methods. In the evaluation, we instantiated our DMT with a ResNet50 backbone. To ensure a fair comparison, we employed an evaluation strategy using 8 and 16 frames, and we compared our method to others that use similar-scale backbone networks. The GFLOPs calculation is based on spatial resolutions of 224 × 224 and 256 × 256.

Methods	Backbones	Frames	FLOPs	Top-1	Top-5
TSN-RGB [23]	BNInception	8 f	16 G	19.5	-
TRN-Multiscale [32]	BNInception	8 f	33 G	34.4	-
S3D-G [6]	Inception	64 f	71 G	48.2	78.7
ECO [35]	BNIncep + Res18	16 f	64 G	41.6	-
ECO_{En}Lite [35]	BNIncep + Res18	92 f	267 G	46.4	-
TSN [23]	ResNet50	8 f	33 G	19.7	46.6
I3D [15]	ResNet50	32 f × 2	306 G	41.6	72.2
NL I3D [29]	ResNet50	32 f × 2	334 G	44.4	76.0
NL I3D+GCN [29]	ResNet50+GCN	32 f × 2	606 G	46.1	76.8
TSM [25]	ResNet50	8 f	33 G	45.6	74.2
TSM [25]	ResNet50	16 f	65 G	47.2	77.1
TSM_{En} [25]	ResNet50	8 f + 16 f	98 G	49.7	78.5
bLvNet-TAM [36]	bLResNet-50	16 f × 2	48 G	48.4	78.8
GST [37]	ResNet50	8 f	30 G	47.0	76.1
GST [37]	ResNet50	16 f	59 G	48.6	77.9
TEINet [38]	ResNet50	8 f	33 G	47.4	-
TEINet [38]	ResNet50	16 f	66 G	49.9	-
TPN [10]	ResNet50	8 f	-	49.0	-
TANet [27]	ResNet50	8 f	33 G	47.3	75.8
TANet [27]	ResNet50	16 f	66 G	47.6	77.7
STPG-Net (TSM) [33]	ResNet-50	8 f × 2	35.9 G	49.3	77.8
STPG-Net (TSM) [33]	ResNet-50	16 f × 2	69.4 G	50.7	78.8
TEFE [34]	ResNet-50	8 f	90 G	46.7	75.3
TEFE [34]	ResNet-50	16 f	181 G	50.4	78.9
DMTNet	ResNet50	8 f	33 G	48.3	77.6
DMTNet	ResNet50	16 f	66 G	49.4	78.4
DMTNet [1]	ResNet50	16 f × 2	86 G × 2	51.0	79.0
DMTNet [2]	ResNet50	8 f × 2 × 3	43 G × 6	49.9	78.4
DMTNet [2]	ResNet50	16 f × 2 × 3	86 G × 6	51.2	79.0

[1] Double sampling that uses two clips and one crop, each with a resolution of 256 × 256, can improve accuracy.
[2] Double sampling that uses two clips and three crops, each with a resolution of 256 × 256, can improve accuracy.

4.3.3. Results8960/432 on Kinetics400

Table 3 shows the comparison with the state-of-the-art results for our DMTNet using ResNet-50 as a backbone with 8- and 16-frame input samplings and testing on the validation set. In comparison to the baseline TANet [27], we argue that we obtain a competitive accuracy without higher improvement because the activities in Kinetics400 are more easily inferred from a single frame, unlike the FingGym99 and SthV1 datasets that have a stronger dependency on temporal modeling. Additionally, we used ResNet-50 as the backbone for our DMTNet, which may have limited the performance compared to deeper network architectures such as ResNet-101 and ResNet-152. However, on the FineGym99 and SthV1 datasets, which have a stronger dependency on temporal modeling, ResNet-50 as the backbone for our DMTNet is able to achieve significant performance improvements as well.

Table 3. Kinetics400 results with other methods. In the evaluation, we instantiated our DMT with a ResNet50 backbone. To ensure a fair comparison, we employed an evaluation strategy using 8 and 16 frames, and we compared our method to others that use similar-scale backbone networks and the same evaluation metrics.

Methods	Backbones	Training Input	GFLOPs [1]	Top-1	Top-5
TSN [23]	InceptionV3	3 × 224 × 224	3 × 250	72.5	90.2
ARTNet [39]	ResNet18	16 × 112 × 112	24 × 250	70.7	89.3
I3D [15]	InceptionV3	64 × 224 × 224	108 × N/A	72.1	90.3
R(2+1)D [5]	ResNet34	32 × 112 × 112	142 × 10	74.3	91.4
NL I3D [29]	ResNet50	128 × 224 × 224	282 × 30	76.5	92.6
ip-CSN [40]	ResNet50	8 × 224 × 224	1.2 × 10	70.8	-
TSM [25]	ResNet50	16 × 224 × 224	65 × 30	74.7	91.4
TEINet [38]	ResNet50	16 × 224 × 224	86 × 30	76.2	92.5
bLVNet-TAM [36]	bLResNet50	48 × 224 × 224	93 × 9	73.5	91.2
SlowOnly [7]	ResNet50	8 × 224 × 224	42 × 30	74.8	91.6
SlowFast$_{4\times16}$ [7]	ResNet50	(4+32) × 224 × 224	36 × 30	75.6	92.1
SlowFast$_{8\times8}$ [7]	ResNet50	(8+32) × 224 × 224	66 × 30	77.0	92.6
I3D * [15]	ResNet50	32 × 224 × 224	335 × 30	76.6	-
TPN [10]	ResNet50	8 × 224 × 224	-	75.5	92.1
TANet-50 [27]	ResNet50	8 × 224 × 224	43 × 30	76.3	92.6
TANet-50 [27]	ResNet50	16 × 224 × 224	86 × 12	76.9	92.9
STM [41]	ResNet50	8 × 224 × 224	33 × 30	75.5	92.0
STM [41]	ResNet50	16 × 224 × 224	67 × 30	76.9	92.7
DMTNet	ResNet50	8 × 224 × 224	43 × 30	75.9	92.6
DMTNet	ResNet50	16 × 224 × 224	86 × 30	77.1	93.0

* I3D model without the incorporation of temporal downsampling. [1] The complexity is expressed as GFLOPs per view × number of views with spatial crops with 256 × 256 resolution.

4.4. Ablation Studies

4.4.1. Parameter Choices

We performed experiments to optimize the hyperparameters of our model by testing different values of γ and varying numbers of dilated convolutions with different dilation rates. Starting with a dilation rate of 1 for the first convolution, we increased the dilation rate by 1 for each subsequent convolution. For example, when the number of dilated convolutions was 3, the dilation rates were 1, 2, and 3, respectively. Our model and instances are illustrated in Figure 2, and Table 4 displays the results of our hyperparameter selection experiments. We found that using two dilated convolutions with a γ value of 1/8 yielded optimal results for FineGym99.

Table 4. Ablation study on hyperparameters.

Kernels [1]	Proportion [2]	Frames	Top-1	Top-5	Mean
2	1/2	8	85.2	98.7	80.0
2	1/4	8	86.9	98.8	82.1
2	1/8	8	87.5	98.9	83.0
2	1/16	8	86.7	98.8	82.2
1	1/8	8	86.9	98.7	82.0
2	1/8	8	87.5	98.9	83.0
3	1/8	8	86.5	98.6	81.9
4	1/8	8	86.4	98.8	81.7

[1] The number of dilated convolutions and kernels generated. [2] The proportion of channels allocated for each dilated convolution.

4.4.2. Different Components

The DMT module incorporates two components to model temporal information: an attention mechanism and the multiple-dilation (MD) operation. The attention mechanism

learns weights that focus on the importance of both temporal and channel dimensions of the feature map, while the MD operation models multiscale temporal information. The effectiveness of the MD operation was evaluated and reported in Table 5. The results demonstrate that the model that utilizes both an attention mechanism and MD operation performed the best. Furthermore, the results indicate that the contribution of the MD operation to the improvement in accuracy is significant. This suggests that while the attention mechanism only weights the feature map, MD operation engages in both temporal interactions and multi-timescale modeling.

Table 5. Ablation study on components.

Attention	MD [1]	Frames	Top-1	Top-5	Mean
✓		8	85.0	98.6	79.5
	✓	8	87.1	98.8	82.7
✓	✓	8	87.5	98.9	83.0

[1] Multiple-dilation operation modeling multi-temporal information.

4.4.3. Accuracy and Loss

As shown in Figure 4, we examined the validation accuracy and training loss during the training process. DMT without MD (marked with w/o MD) and TAM have similar attention and adaptive convolution strategies. Therefore, the performance curves of TANet and DMTNet without the MD operation tend to converge during the training process. Meanwhile, it can be observed that the validation accuracy of DMTNet containing the MD operation experiences a more rapid increase during the training process and reaches a superior accuracy at the 50th epoch in comparison to both DMTNet without the MD operation and TANet. Additionally, it is evident that the loss curve of DMTNet with the MD operation exhibits a faster decline and reaches a more favorable minimum value compared to DMTNet without the MD operation during the training process. These results indicate the effectiveness of the MD operation in improving the performance of the DMT module, as demonstrated by its faster increase in validation accuracy and ability to reach a higher accuracy compared to models without this operation.

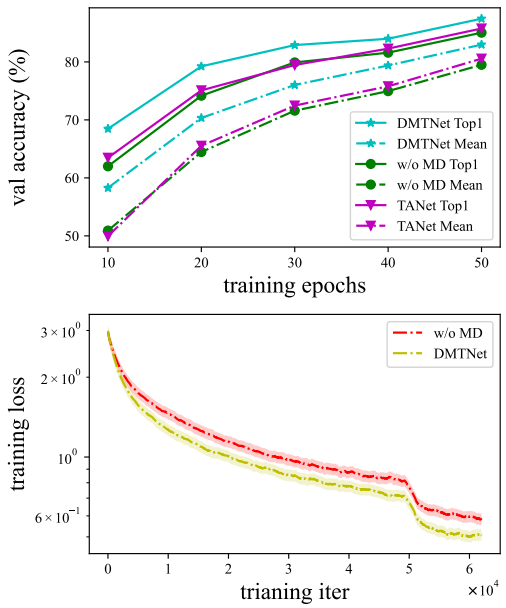

Figure 4. Validation accuracy and training loss during the training process.

4.4.4. Why FineGym99?

The selection of FineGym99 as the dataset for our experiments was based on several key factors. First and foremost, the dataset demonstrates significant temporal variance in the same category of actions across different videos, providing an ideal platform for evaluating the competence of models in addressing temporal differences. With 99 fine-grained action classes, FineGym99 is characterized by subtle differences in spatial appearance but distinct movements [13]. In other words, the dataset presents a challenge in distinguishing actions based on a few frames alone, emphasizing the need for a better understanding of the action processes within the videos. This characteristic poses a challenge to accurately classify actions based solely on their spatial appearance, highlighting the need for models to effectively model the temporal information present in video data. As such, FineGym99 serves as an ideal platform to assess the capability of models in this regard. Our ablation studies on FineGym99 provide valuable insights into the effectiveness of the DMT module in capturing the diverse temporal information present in videos.

4.5. Empirical Analysis

To evaluate the performance of our DMT module, we conducted experiments on the FineGym99 dataset, focusing on analyzing its accuracy on different variances of action duration. We introduced the concept of class variance to describe the variation between the durations of videos in the class. We then conducted an empirical analysis to study the accuracy improvement achieved by our module on selected actions. Our analysis showed that the DMT module effectively improves accuracy across various action classes. These results demonstrate the potential of our module for enhancing the performance of action recognition.

4.5.1. Class Variance

In the FineGym99 dataset, we conducted an investigation into the variance of the video duration within each class and class variance. The duration of the videos varied among the classes; for example, one class may have two videos containing 100 and 200 frames, while another class may have two videos containing 50 and 100 frames. Despite that the first difference in frame count of such two videos ($200 - 100 = 100$) is double compared with the second one ($100 - 50 = 50$), the ratios are the same ($200/100 = 100/50$). To align the ratios of videos within a class, normalization was applied by dividing the median of the videos and then multiplying by 25 (the amount of frames per second) in order to establish a uniform standard of speed rate across all classes. We counted the variance of the resulting normalized duration as the variance of the class, providing an understanding of the variability of video duration within each class. Class variance is defined as follows:

$$variance = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{f_i - \bar{f}}{\widetilde{f}} \times 25\right)^2, \tag{7}$$

where n represents the number of videos in a class, $\bar{f}$ represents the mean number of frames per video, and $\widetilde{f}$ represents the median number of frames per video, which is used for normalization.

4.5.2. Performance Gain

As illustrated in Figure 5, an analysis was conducted to investigate the correlation between the improved accuracy and class variance. The improved accuracy refers to the difference between the prediction accuracy of using the MD strategy (described in Section 4.4.2) and the prediction accuracy without using it for that class. Because of the variability between classes, the performance gain was not consistently linear across all classes. To further investigate this trend, the classes demonstrated were grouped into four bins and sorted by the absolute gain in each bin. The results revealed that the classes in the leftmost bin demonstrated a significant improvement in accuracy, whereas the classes in the rightmost

bin exhibited a relatively small improvement. Furthermore, it was observed that there is a positive correlation between the absolute gains and class variance in each bin.

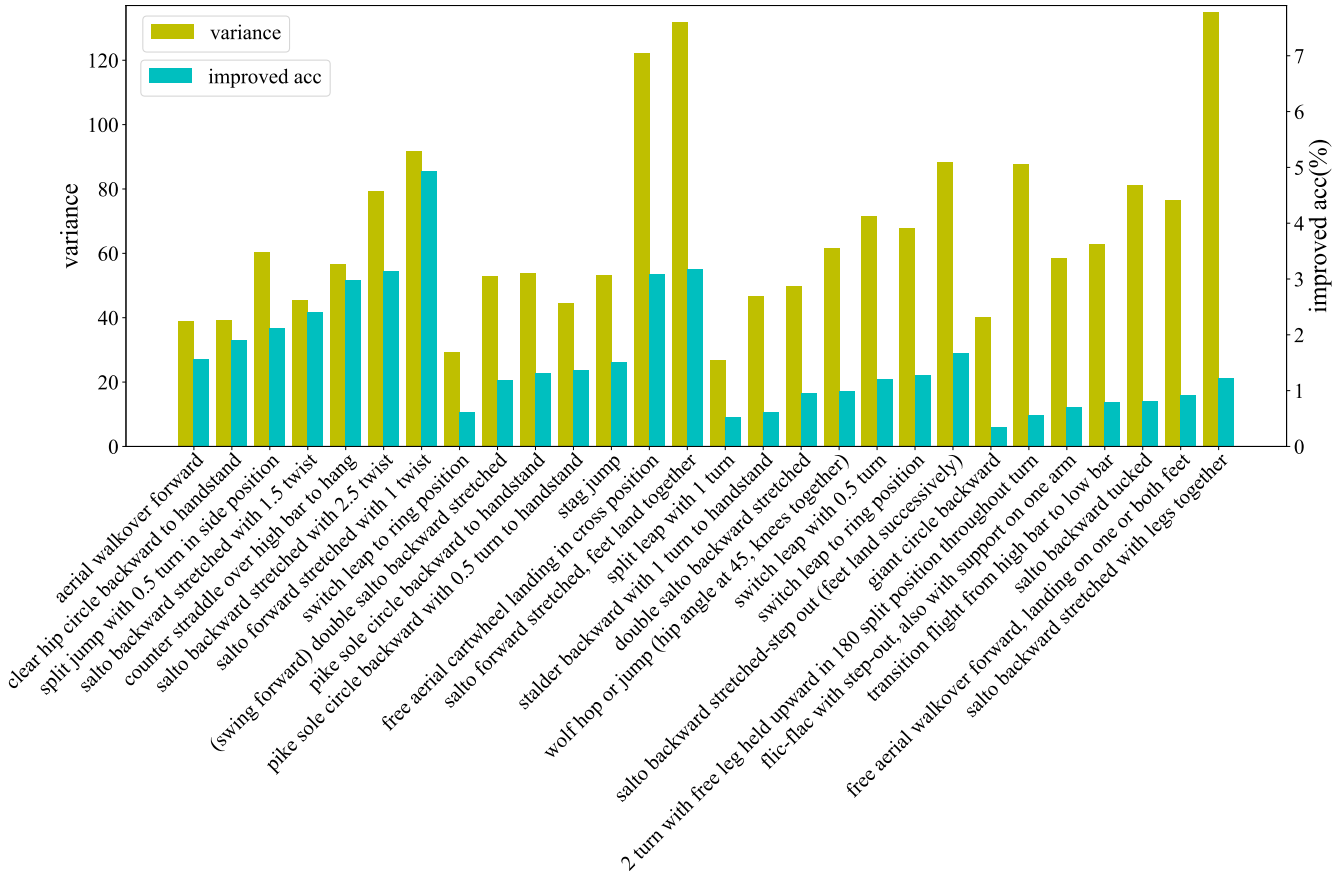

Figure 5. Class variance and performance gain. Each yellow bar represents the variance of the class, while the corresponding cyan bar demonstrates the performance improvement achieved in that class.

5. Conclusions

In this paper, we presented a novel approach to action recognition in videos by introducing a dilated multi-temporal module. The proposed module is compatible with existing backbones and has been shown to improve performance through a comprehensive set of experiments and ablation studies. Additionally, the empirical analysis demonstrates the effectiveness of the module for multiple temporal modeling, with a positive correlation between the absolute gains and class variance. The proposed method can be a valuable addition to the current methods in action recognition, and further research can be conducted to explore its potential in other video-related tasks.

Author Contributions: Conceptualization, T.Z.; software, T.Z.; validation, T.Z. and Y.W.; formal analysis, T.Z. and X.L.; resources, X.L.; writing—review and editing, T.Z. and Y.W.; supervision, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: In this manuscript, the employed datasets have been taken with license agreements from the corresponding institutions with proper channels.

Acknowledgments: We appreciate the HighPerformance Computing Center of Shanghai University, and Shanghai Engineering Research Center of Intelligent Computing Systems for providing the computing resources and technical support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 568–576.
2. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 221–231. [CrossRef] [PubMed]
3. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
4. Qiu, Z.; Yao, T.; Mei, T. Learning spatio-temporal representation with pseudo-3d residual networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5533–5541.
5. Tran, D.; Wang, H.; Torresani, L.; Ray, J.; LeCun, Y.; Paluri, M. A closer look at spatiotemporal convolutions for action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6450–6459.
6. Xie, S.; Sun, C.; Huang, J.; Tu, Z.; Murphy, K. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 305–321.
7. Feichtenhofer, C.; Fan, H.; Malik, J.; He, K. Slowfast networks for video recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6202–6211.
8. Wang, Y.; Long, M.; Wang, J.; Yu, P.S. Spatiotemporal pyramid network for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1529–1538.
9. Zhang, D.; Dai, X.; Wang, Y.F. Dynamic temporal pyramid network: A closer look at multi-scale modeling for activity detection. In Proceedings of the Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 712–728.
10. Yang, C.; Xu, Y.; Shi, J.; Dai, B.; Zhou, B. Temporal pyramid network for action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 591–600.
11. Luo, W.; Li, Y.; Urtasun, R.; Zemel, R. Understanding the effective receptive field in deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 4905–4913.
12. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
13. Shao, D.; Zhao, Y.; Dai, B.; Lin, D. Finegym: A hierarchical video dataset for fine-grained action understanding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2616–2625.
14. Goyal, R.; Ebrahimi Kahou, S.; Michalski, V.; Materzynska, J.; Westphal, S.; Kim, H.; Haenel, V.; Fruend, I.; Yianilos, P.; Mueller-Freitag, M.; et al. The "something something" video database for learning and evaluating visual common sense. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5842–5850.
15. Carreira, J.; Zisserman, A. Quo vadis, action recognition? A new model and the kinetics dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
16. Wang, H.; Schmid, C. Action recognition with improved trajectories. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 3551–3558.
17. Kerdvibulvech, C.; Yamauchi, K. Structural human shape analysis for modeling and recognition. In Proceedings of the Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshop, S+SSPR 2014, Joensuu, Finland, 20–22 August 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 282–290.
18. Cui, M.; Wang, W.; Zhang, K.; Sun, Z.; Wang, L. Pose-Appearance Relational Modeling for Video Action Recognition. *IEEE Trans. Image Process.* **2022**, *32*, 295–308. [CrossRef] [PubMed]
19. Thatipelli, A.; Narayan, S.; Khan, S.; Anwer, R.M.; Khan, F.S.; Ghanem, B. Spatio-temporal relation modeling for few-shot action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 19958–19967.
20. Feichtenhofer, C.; Pinz, A.; Wildes, R.P. Spatiotemporal multiplier networks for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4768–4777.
21. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1725–1732.
22. Wang, X.; Gupta, A. Videos as space-time region graphs. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 399–417.

23. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal segment networks: Towards good practices for deep action recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 20–36.
24. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
25. Lin, J.; Gan, C.; Han, S. Tsm: Temporal shift module for efficient video understanding. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7083–7093.
26. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
27. Liu, Z.; Wang, L.; Wu, W.; Qian, C.; Lu, T. Tam: Temporal adaptive module for video recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 13708–13718.
28. Contributors, M. OpenMMLab's Next Generation Video Understanding Toolbox and Benchmark. 2020. Available online: https://github.com/open-mmlab/mmaction2 (accessed on 10 May 2022).
29. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
30. Bottou, L. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 421–436.
31. Girdhar, R.; Ramanan, D.; Gupta, A.; Sivic, J.; Russell, B. Actionvlad: Learning spatio-temporal aggregation for action classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 971–980.
32. Zhou, B.; Andonian, A.; Oliva, A.; Torralba, A. Temporal relational reasoning in videos. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 803–818.
33. Geng, T.; Zheng, F.; Hou, X.; Lu, K.; Qi, G.J.; Shao, L. Spatial-Temporal Pyramid Graph Reasoning for Action Recognition. *IEEE Trans. Image Process.* **2022**, *31*, 5484–5497. [CrossRef] [PubMed]
34. Jiang, J.; Zhang, Y. An improved action recognition network with temporal extraction and feature enhancement. *IEEE Access* **2022**, *10*, 13926–13935. [CrossRef]
35. Zolfaghari, M.; Singh, K.; Brox, T. Eco: Efficient convolutional network for online video understanding. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 695–712.
36. Fan, Q.; Chen, C.F.R.; Kuehne, H.; Pistoia, M.; Cox, D. More is less: Learning efficient video representations by big-little network and depthwise temporal aggregation. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 2264–2273.
37. Luo, C.; Yuille, A.L. Grouped spatial-temporal aggregation for efficient action recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5512–5521.
38. Liu, Z.; Luo, D.; Wang, Y.; Wang, L.; Tai, Y.; Wang, C.; Li, J.; Huang, F.; Lu, T. Teinet: Towards an efficient architecture for video recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11669–11676.
39. Wang, L.; Li, W.; Li, W.; Van Gool, L. Appearance-and-relation networks for video classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1430–1439.
40. Tran, D.; Wang, H.; Torresani, L.; Feiszli, M. Video classification with channel-separated convolutional networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 5552–5561.
41. Wang, M.; Xing, J.; Su, J.; Chen, J.; Liu, Y. Learning spatiotemporal and motion features in a unified 2d network for action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 3347–3362. [CrossRef] [PubMed]