Multi-Scale Spatial-Temporal Integration Convolutional Tube for Human Action Recognition

Haoze Wu[1*], **Jiawei Liu**[1*], **Xierong Zhu**[1], **Meng Wang**[2] and **Zheng-Jun Zha**[1†]
[1]University of Science and Technology of China
[2]Hefei University of Technology
wuhaoze@mail.ustc.edu.cn, jwliu6@ustc.edu.cn, zxr8192@mail.ustc.edu.cn,
eric.mengwang@gmail.com, zhazj@ustc.edu.cn

Abstract

Applying multi-scale representations leads to consistent performance improvements on a wide range of image recognition tasks. However, with the addition of the temporal dimension in video domain, directly obtaining layer-wise multi-scale spatial-temporal features will add a lot extra computational cost. In this work, we propose a novel and efficient Multi-Scale Spatial-Temporal Integration Convolutional Tube (MSTI) aiming at achieving accurate recognition of actions with lower computational cost. It firstly extracts multi-scale spatial and temporal features through the multi-scale convolution block. Considering the interaction of different-scales representations and the interaction of spatial appearance and temporal motion, we employ the cross-scale attention weighted blocks to perform feature recalibration by integrating multi-scale spatial and temporal features. An end-to-end deep network, MSTI-Net, is also presented based on the proposed MSTI tube for human action recognition. Extensive experimental results show that our MSTI-Net significantly boosts the performance of existing convolution networks and achieves state-of-the-art accuracy on three challenging benchmarks, *i.e.*, UCF-101, HMDB-51 and Kinetics-400, with much fewer parameters and FLOPs.

1 Introduction

With the rapid development of various video platforms in the social network, video is becoming a popular communication medium among internet users. This has encouraged the development of advanced techniques for a variety of video understanding applications. More specifically, one of the most fundamental tasks that ensures the success of these technological advances is human action recognition. Human action recognition aims to recognize actions by the visual appearance and motion dynamics of the involved humans and objects in video sequences. Recently, the Convolutional Neural Networks (CNNs) demonstrate the high capability of learning visual representation in image domain and the way that extends CNNs into video domain is the main proposal for the action recognition task.

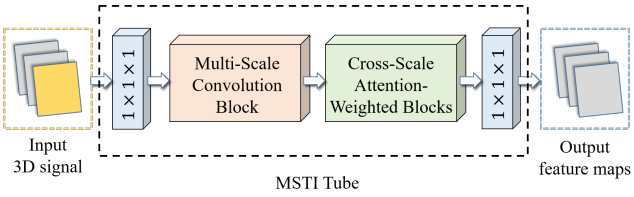


Figure 1: The illustration of our proposed MSTI tube. The MSTI tube mainly consists of two blocks: the multi-scale convolution block and the cross-scale attention weighted blocks.

Video actually can be seen as a sequence of images in the dimension of time. A good model should be able to extract not only the spatial appearance in images, but the dynamic motion change over time. Traditional 2D-CNN base methods [Simonyan and Zisserman, 2014; Donahue *et al.*, 2015] neglected the joint exploration of spatial appearance and temporal motion, which could offer a comprehensive representation of videos and thus enhance the accuracy of action recognition. For most 3D-CNN based methods [Tran *et al.*, 2015; 2018; Carreira and Zisserman, 2017; Wang *et al.*, 2018b], they more or less integrated spatial and temporal information. However, with the addition of a new dimension, the parameters and computational cost of the 3D-CNN based models are always extremely high compared to the 2D-CNN based models. The P3D [Qiu *et al.*, 2017] took the lead in separating the 3D convolution into two separate convolutions, *i.e.*, a 2D spatial convolution plus a 1D temporal convolution, and thus significantly reduced the model size. Nevertheless, this kind of method still ignored the correlation of spatial appearance and temporal motion.

Recently, the multi-scale representations [Szegedy *et al.*, 2017; Gao *et al.*, 2019; Liu *et al.*, 2016] are of critical importance to a number of vision recognition tasks. For human action recognition task, the humans and objects may appear with different spatial sizes in images, and the actions may also last for different lengths of time in videos. However, directly obtaining layer-wise multi-scale spatial-temporal features requires feature extractors to use a large range of receptive

*Equal contribution
†Corresponding author

fields, which will add a large amount of extra computational cost. In addition, the interaction between the feature maps at different scales can guide themselves to pay attention to informative features rather than useless features, since the contextual information of an action may occupy a much larger area than the action itself.

In this paper, we propose a Multi-Scale Spatial-Temporal Integration Convolutional Tube (MSTI) aiming towards robust and accurate human action recognition tasks. The MSTI tube generates multi-scale spatial appearance and temporal motion through multi-group convolution, and then applies feature recalibration by integrating multi-scale spatial and temporal features to obtain effective spatial-temporal features, simultaneously reducing the computational cost. Specifically, the MSTI tube consists of multi-scale convolution block and cross-scale attention weighted blocks. An illustration of our MSTI tube is shown in Fig.1. The multi-scale convolution block divides the input tensor into several groups, and each group has their own convolutional filters. For the first group, it's output feature maps are directly calculated by it's input feature maps and filters. For the other groups, the previous group's output feature maps are sent to the current group's filters along with the current group's input feature maps. The cross-scale attention weighted blocks integrate multi-scale spatial and temporal features, aiming at selectively emphasizing informative spatial-temporal features and suppressing less useful ones. In the spatial branch, the cross-scale attention weighted block takes the previous group's spatial feature maps, the current group's spatial and temporal feature maps as inputs, and then generates optimized spatial feature maps; In the temporal branch, it takes the previous group's temporal feature maps, the current group's temporal and spatial feature maps as inputs, and then generates optimized temporal feature maps.

The main contribution of this work can be briefly summarized as: 1) we design the multi-scale convolution block to capture multi-scale representations on both spatial and temporal domain; 2) we design the cross-scale attention weighted blocks which integrate multi-scale spatial and temporal features, aiming to perform feature recalibration by selectively emphasizing informative spatial-temporal features and suppressing less useful ones; 3) a deep network with low computational cost, MSTI-Net, is put forward for learning robust and accurate video representation. Experiment results show that our MSTI-Net outperforms other methods on three challenging action recognition benchmarks, Kinetics-400, UCF-101 and HMDB-51 with lower computational cost.

2 Related Work

With the rapid development of convolutional neural networks in the field of image, the video field is becoming a more and more popular field people try to expand into. According to the types of convolutions used in features learning, existing action recognition works can be briefly divided into two categories: 2D CNN and 3D CNN based methods.

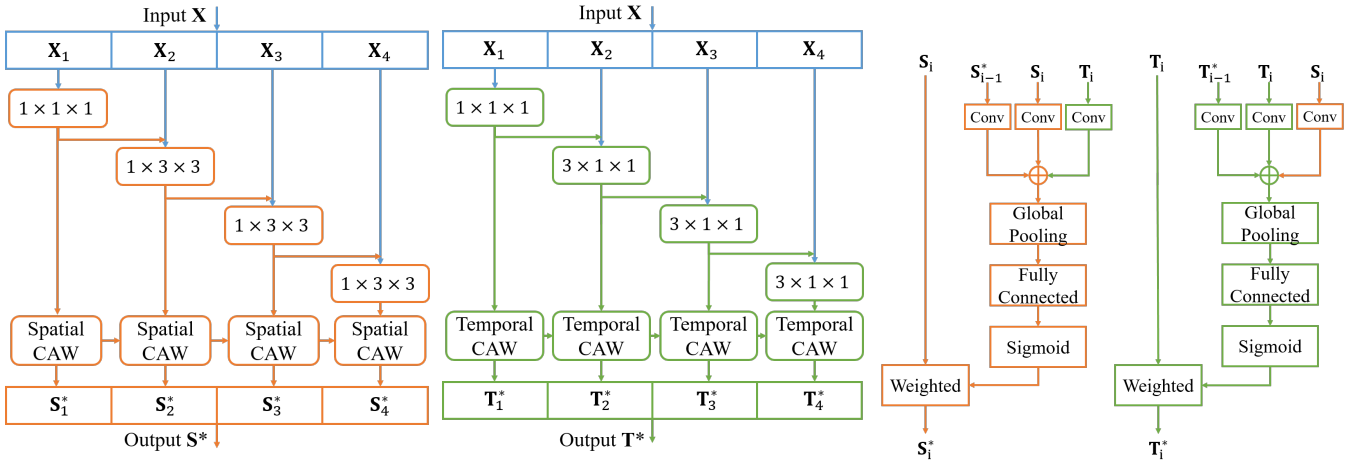
2D CNN based. Karpathy *et al*. [Karpathy *et al.*, 2014] proposed a "slow fusion" model which took the lead in fusing temporal information into 2D CNNs. The model firstly extended temporal connectivity of all convolutional layers and then computed activation through temporal and spatial convolutions. The two-stream structure proposed by [Simonyan and Zisserman, 2014] is one of the influential approaches which directly used two 2D CNNs to capture spatial and temporal information respectively from RGB frames and stacked optical flows, improving video recognition accuracy. Following this idea, several studies have been presented to fuse these two networks over the appearance and motion, *e.g.* the ST-ResNet [Feichtenhofer *et al.*, 2016] and the temporal segment networks [Wang *et al.*, 2016]. LRCN [Donahue *et al.*, 2015] tried to explore the possibility of combining LSTM networks with frame-level features of 2D CNNs to explicitly model spatial-temporal relationships. Recently, the multi-scale representations which are of great importance to various vision tasks have been widely used in a number of networks, such as Inception-Nets [Szegedy *et al.*, 2017] and Res2Net [Gao *et al.*, 2019]. Meanwhile, at the channel level, the SE-Net [Hu *et al.*, 2018] extracted different channels' attention to further improve the quality of representations produced by a 2D network, which achieved impressed results on image classification.

3D CNN based. The 3D Convolutional Networks were first presented for learning video representations over 16-frame video clips in the context of large-scale supervised video datasets [Tran *et al.*, 2015]. Compared to 2D kernels which merely model the spatial information, 3D convolution kernels have the capability of modeling more complex relations between appearance and motion. The Res3D [Tran *et al.*, 2017] made one step further by taking the advantage of residual connections to facilitate training. Similarly, I3D [Carreira and Zisserman, 2017] was proposed to use the inception network [Szegedy *et al.*, 2017] as backbone rather than residual networks to learn video representations. Many architectures were also proposed to improve 3D convolution [Tran *et al.*, 2018; Zhou *et al.*, 2018; He *et al.*, 2018; Wang *et al.*, 2018b]. However, all of these methods demanded more than an order of magnitude computational cost than their 2D competitors. This made them difficult to train and apply to practical applications. To overcome the limitation of 3D CNN and decrease the number of parameters, the P3D [Qiu *et al.*, 2017] decomposed a 3D convolution kernel into a 2D spatial kernel and a 1D temporal kernel to reduce the computations of a 3D convolutional layer and achieved better precision at the same time. Wu et al. [Wu *et al.*, 2019a] further optimized this kind of structure, and proposed a mutually reinforced spatio-temporal convolutional tube (MRST) to learn the correlation between spatial and temporal features. Based on the depth-wise separable convolutions idea, the work [Wu *et al.*, 2019b] proposed depth-wise separable 3D convolution networks, which commendably simplified the large inputs tensor.

3 MSTI Tube and Deep MSTI Network

3.1 MSTI Tube

The multi-scale spatial-temporal integration convolutional tube (MSTI) applies the bottleneck structure, as shown in Fig.1, which employs two $1 \times 1 \times 1$ convolutional layers at



(a) MSTI-spatial branch (b) MSTI-temporal branch (c) Spatial and temporal CAW blocks

Figure 2: The detailed architecture of our proposed MSTI tube. (a) MSTI-spatial branch: a stepped structure, generating spatial outputs which have four spatial receptive fields and further optimizing them with four spatial CAW blocks. (b) MSTI-temporal branch: a stepped structure, generating temporal outputs which have four temporal receptive fields and further optimizing them with four temporal CAW blocks. (c) Spatial and temporal cross-scale attention weighted (CAW) blocks: Utilizing the output spatial-temporal feature maps of previous group and current group to selectively emphasize informative spatial and temporal features and suppress useless ones.

both ends of the path to reduce and restore the channel dimensions respectively, decreasing the overall computational cost. In this section, we will first introduce concrete details of the composition of the MSTI tube, *i.e.*, the multi-scale convolution block and the spatial and temporal cross-scale attention weighted blocks. We then present our robust and efficient deep network, MSTI-Net for human action recognition.

Multi-Scale Convolution Block

In the multi-scale convolution block, we first evenly slice the 3D input feature maps $\mathbf{X} \in \mathbb{R}^{L\times H\times W\times C}$ into four groups, denoted by $\mathbf{X}_i \in \mathbb{R}^{L\times H\times W\times \tilde{C}}$, where $i \in \{1,2,3,4\}$, and $L, H, W, \tilde{C}$ refer to the length, height, width and the number of group channels, respectively.

In the MSTI-spatial branch, each group $\mathbf{X}_i$ has a corresponding $1 \times 3 \times 3$ spatial convolution, except that the first group $\mathbf{X}_1$ is followed by a $1 \times 1 \times 1$ spatial convolution. The corresponding spatial convolution of each group are denoted by $\mathbf{K}_i^s$, and the outputs of each spatial convolution are named $\mathbf{S}_i$. The whole multi-scale spatial convolution architecture presents a stepped structure, as shown in Fig.2(a). The output $\mathbf{S}_i$ can be written as:

$$\mathbf{S}_i = \begin{cases} \mathbf{K}_i^s(\mathbf{X}_i) & i = 1 \\ \mathbf{K}_i^s(\mathbf{X}_i + \mathbf{S}_{i-1}) & 2 \le i \le 4 \end{cases} \tag{1}$$

From the above formula, we notice that each time the spatial features split $\mathbf{S}_i$ goes through a $1 \times 3 \times 3$ spatial convolutional kernel, the output result can have a larger spatial receptive field than $\mathbf{S}_i$, increasing by 2 in both the height and width dimensions. In this way, we get four different spatial receptive fields, *i.e.*, $1 \times 1 \times 1$, $1 \times 3 \times 3$, $1 \times 5 \times 5$ and $1 \times 7 \times 7$. With these multi-scale spatial features, we can learn a more discriminative spatial representation.

Similar to the MSTI-spatial branch, in the MSTI-temporal branch, each input group $\mathbf{X}_i$ has a corresponding $3 \times 1 \times 1$ temporal convolution, except that the first group $\mathbf{X}_1$ is followed by a $1 \times 1 \times 1$ temporal convolution. We denote the corresponding temporal convolution of each group by $\mathbf{K}_i^t$, and named each temporal convolution's outputs $\mathbf{T}_i$, as shown in Fig.2(b). The output $\mathbf{T}_i$ can be calculated by:

$$\mathbf{T}_i = \begin{cases} \mathbf{K}_i^t(\mathbf{X}_i) & i = 1 \\ \mathbf{K}_i^t(\mathbf{X}_i + \mathbf{T}_{i-1}) & 2 \le i \le 4 \end{cases} \tag{2}$$

Each time the temporal features split $\mathbf{T}_i$ goes through a $3 \times 1 \times 1$ temporal convolutional kernel, the output's temporal receptive field increases by 2 in the length dimensions. Therefore, we obtain multi-scale temporal representation with four different temporal receptive fields, *i.e.*, $1 \times 1 \times 1$, $3 \times 1 \times 1$, $5 \times 1 \times 1$ and $7 \times 1 \times 1$.

In addition, the multi-scale (MS) convolution can also be applied to reduce the impact of the large C_{in} or C_{out} on computational complexity. We present the parameters of the 3D convolution and our MS convolution in terms of formulas, which can be shown as follows:

$$\begin{aligned} P_{(3D)} &= d \times k \times k \times (C_{in} \times C_{out}) \\ P_{(MS)} &= d \times k \times k \times N \times (C_{in}/N \times C_{out}/N) \\ &= d \times k \times k \times (C_{in} \times C_{out})/N = P_{(3D)}/N \end{aligned} \tag{3}$$

where d and $k \times k$ refer to the temporal and spatial kernel size, respectively. We can see that compared to the original 3D convolution, the MS convolution can reduce the number of parameters by a factor of N, and in the meantime keeping the height, width as well as the length of feature maps unchanged. We set $N = 4$ in our multi-scale convolution.

Figure 3: Visualization of the spatial and temporal feature maps at different scales. The left picture is a frame extracted from the baseball pitching video in UCF-101. The middle column of pictures are the small-scale ($1 \times 3 \times 3$) and large-scale ($1 \times 7 \times 7$) spatial feature maps of the left frame; The right column of pictures are the small-scale($3 \times 1 \times 1$) and large-scale ($7 \times 1 \times 1$) temporal feature maps of the left frame and its adjacent frames.

Cross-Scale Attention Weighted Blocks

The spatial and temporal cross-scale attention weighted (CAW) blocks in the MSTI tube are applied after the multi-scale convolution block. The detailed architecture of spatial and temporal CAW blocks are illustrated with Fig.2(c). In the spatial and temporal CAW blocks, we integrate spatial feature maps, temporal feature maps and the previous-scale feature maps to further optimize spatial and temporal features, respectively, selectively emphasizing informative spatial-temporal features and suppressing less useful ones.

We make a visualization of spatial and temporal feature maps at different scales, as shown in Fig.3. we find that the spatial feature maps (middle column) mainly concentrate on the objects with surrounds, such as the human appearance and the backgrounds, and the temporal feature maps (right column) mainly concentrate on the objects which are in continuous motion, such as the human's body and legs. Integrating spatial and temporal feature maps can give a guideline to perform feature recalibration, which can pay more attention to the humans and objects' spatial appearance and temporal motion rather than the backgrounds. In addition, compared to the large-scale feature maps (the second row) which have larger receptive fields and contain more spatial-temporal information, the resolution of the small-scale feature maps (the first row) is much finer, which can guide the attention of large-scale resolution areas. Besides, in terms of the multi-scale convolution structure, the previous group's outputs are a part of the current group's inputs. Thus, the small-scale feature maps is instructive to the large-scale feature maps.

We suppose each group's multi-scale spatial and temporal convolution output tensors have the same size $\mathbf{S}_i, \mathbf{T}_i \in \mathbb{R}^{L \times H \times W \times \tilde{C}}$. Then we can rewrite them as $\mathbf{S}_i = [\mathbf{s}_{i1}, \mathbf{s}_{i2}, \ldots, \mathbf{s}_{i\tilde{C}}]$ and $\mathbf{T}_i = [\mathbf{t}_{i1}, \mathbf{t}_{i2}, \ldots, \mathbf{t}_{i\tilde{C}}]$, where $\mathbf{s}_{ic}, \mathbf{t}_{ic}$ refer to the features of the c-th channel in $\mathbf{S}_i$ and $\mathbf{T}_i$, respectively. We first utilize three $1 \times 1 \times 1$ convolutional layers to integrate the spatial-temporal feature maps of the current group and the optimized feature maps of the previous group, except that in the first group, we only apply two $1 \times 1 \times 1$ convolutional layers to integrate it's spatial and temporal feature maps. In the spatial and temporal CAW blocks, we calculate the i-th group integrated features $\mathbf{U}_i$ as follows:

$$\begin{aligned} \mathbf{U}_{Si} &= \mathbf{W}^*_{SS}\mathbf{S}^*_{i-1} + \mathbf{W}_{SS}\mathbf{S}_i + \mathbf{W}_{TS}\mathbf{T}_i \\ \mathbf{U}_{Ti} &= \mathbf{W}^*_{TT}\mathbf{T}^*_{i-1} + \mathbf{W}_{TT}\mathbf{T}_i + \mathbf{W}_{ST}\mathbf{S}_i \end{aligned} \tag{4}$$

where $\mathbf{U}_{Si}$ and $\mathbf{U}_{Ti}$ refer to the i-th group integrated spatial and temporal features, respectively; $\mathbf{W}^*_{SS}$, $\mathbf{W}_{SS}$, $\mathbf{W}_{TS}$ refer to the parameters of the three $1 \times 1 \times 1$ convolutional layers in spatial CAW block, and $\mathbf{W}^*_{TT}$, $\mathbf{W}_{TT}$, $\mathbf{W}_{ST}$ refer to the parameters of the three $1 \times 1 \times 1$ convolutional layers in temporal CAW block. We can further rewrite $\mathbf{U}_i$ as $\mathbf{U}_i = [\mathbf{u}_{i1}, \mathbf{u}_{i2}, \ldots, \mathbf{u}_{i\tilde{C}}]$.

We then use a global average pooling operation to generate channel-wise statistics. Formally, a statistic $\mathbf{z}_i \in \mathbb{R}^{\tilde{C}}$ is generated by shrinking the i-th group integrated features $\mathbf{U}_i$ through its spatial-temporal dimensions $L \times H \times W$, therefore the c-th element of $\mathbf{z}_i$ is calculated by:

$$z_{ic} = \mathbf{F}_{gp}(\mathbf{u}_{ic}) = \frac{1}{L \times H \times W} \sum_{k=1}^{L} \sum_{m=1}^{H} \sum_{n=1}^{W} u_{ic}(k, m, n) \tag{5}$$

To make a good use of the information aggregated in the global average pooling operation, we follow it with an excitation operation aiming to fully capture channel-wise dependencies. The excitation function must firstly be capable of learning a nonlinear interaction between channels, and secondly it must learn a non-mutually-exclusive relationship since we would like to ensure that multiple channels are allowed to be emphasized. Thus, we apply the fully connected layer and the sigmoid function to make the excitation operation, and the i-th group attention output is denoted by $\mathbf{a}_i$. The function can be shown as follows:

$$\mathbf{a}_i = \mathbf{F}_{ex}(\mathbf{z}_i, \mathbf{W}) = \sigma(\mathbf{W}\mathbf{z}_i) \tag{6}$$

where σ refers to the sigmoid function, and $\mathbf{W} \in \mathbb{R}^{\tilde{C} \times \tilde{C}}$ denotes the weights of the fully connected layer.

The final outputs of the spatial and temporal CAW blocks $\mathbf{S}^*_i$ and $\mathbf{T}^*_i$ are obtained by re-scaling multi-scale convolution outputs $\mathbf{S}_i$ and $\mathbf{T}_i$ with the activation $\mathbf{a}_i$:

$$\begin{aligned} \mathbf{s}^*_{ic} &= \mathbf{F}_{weighted}(\mathbf{s}_{ic}, a_{isc}) = a_{isc} \cdot \mathbf{s}_{ic} \\ \mathbf{t}^*_{ic} &= \mathbf{F}_{weighted}(\mathbf{t}_{ic}, a_{itc}) = a_{itc} \cdot \mathbf{t}_{ic} \end{aligned} \tag{7}$$

where $\mathbf{S}^*_i = [\mathbf{s}^*_{i1}, \mathbf{s}^*_{i2}, \ldots, \mathbf{s}^*_{i\tilde{C}}]$, $\mathbf{T}^*_i = [\mathbf{t}^*_{i1}, \mathbf{t}^*_{i2}, \ldots, \mathbf{t}^*_{i\tilde{C}}]$ and the weighted function refers to channel-wise multiplication between the output attention and the spatial-temporal feature maps.

Finally, we concatenate each group's optimized spatial and temporal features $\mathbf{S}^*_i$ and $\mathbf{T}^*_i$ together and calculate the final outputs of our MSTI tube $\mathbf{X}^*$ as follows:

$$\begin{aligned} \mathbf{S}^* &= concat(\mathbf{S}^*_1, \mathbf{S}^*_2, \mathbf{S}^*_3, \mathbf{S}^*_4) \\ \mathbf{T}^* &= concat(\mathbf{T}^*_1, \mathbf{T}^*_2, \mathbf{T}^*_3, \mathbf{T}^*_4) \\ \mathbf{X}^* &= \mathbf{S}^* + \mathbf{T}^* \end{aligned} \tag{8}$$

MSTI-Net				
layer	Repeat	Kernel	Strides	output size
Input				$16 \times 224 \times 224 \times 3$
conv1	1	$3 \times 7 \times 7$	$(1, 2, 2)$	$16 \times 112 \times 112 \times 64$
MaxPool		$3 \times 3 \times 3$	$(1, 2, 2)$	$16 \times 56 \times 56 \times 64$
conv2_x	1 2	MSTI	$(2, 1, 1)$ $(1, 1, 1)$	$8 \times 56 \times 56 \times 256$
conv3_x	1 3	MSTI	$(1, 2, 2)$ $(1, 1, 1)$	$8 \times 28 \times 28 \times 512$
conv4_x	1 5	MSTI	$(1, 2, 2)$ $(1, 1, 1)$	$8 \times 14 \times 14 \times 1024$
conv5_x	1 2	MSTI	$(1, 2, 2)$ $(1, 1, 1)$	$8 \times 7 \times 7 \times 2048$
global average pooling, fc layer with softmax				$1 \times 1 \times 1 \times N$

Table 1: Architecture of the deep MSTI-Net. The details of each convolutional layer are shown in brackets, in the order of the repeat times, kernel, strides and output size. The dimensions of kernel and strides are given by time, height, and width. The dimensions of output size are given by time, height, width and number of channels.

3.2 Deep MSTI Network

We propose an efficient and effective MSTI-Net based on the ResNet-50 structure. The proposed MSTI-Net has 50 layers, which contains an initial convolutional layer (conv1), a max-pooling layer, four convolutional residual blocks (conv2-conv5) and a fully connected layer. The main idea of the residual block is to learn the additive residual function with reference to the unit inputs which is realized through a shortcut connection, instead of directly learning unreferenced non-linear functions [He *et al.*, 2016]. The repeat times, kernel size, strides and output size of each convolutional block are all shown in Table 1. In the conv1 layer which is mainly proposed to learn rough spatial-temporal features, we apply $3 \times 7 \times 7$ as kernel. And in the other convolutional blocks, we apply the MSTI tube as kernel. We don't apply the temporal pooling in the last three convolutional blocks in order to ensure the effectiveness of multi-scale convolution and multi-scale spatial-temporal information interaction.

4 Experiments

4.1 Datasets and Implementation Details

Datasets. We use three widely-used and challenging benchmarks, *i.e.* Kinetics-400 [Kay *et al.*, 2017], UCF-101 [Soomro *et al.*, 2012], and HMDB-51 [Kuehne *et al.*, 2013] in the experiments. The large-scale Kinetics-400 dataset consists of about 300, 000 videos from 400 action categories. The UCF-101 dataset is composed by 101 action categories and 13,320 manually labeled video clips in total. The HMDB-51 dataset is collected from various sources, *e.g.* movies and web videos. It is composed by 51 categories and 6,849 labeled video clips in total. Both UCF-101 and HMDB-51 consists of three training/test splits provided by the datasets organizers. We report the accuracy by averaging over all 3 splits.

Implementation details. Our data augmentation includes random clipping on both spatial dimension (by firstly resizing the smaller video side to 256 pixels, then randomly cropping a 224×224 patch) and temporal dimension (by randomly picking the starting frame among those early enough to guarantee a desired number of frames). To obtain the video predictions, we average clip predictions uniformly sampled from the long video sequence. We apply batch normalization and ReLU nonlinearities [Nair and Hinton, 2010] to all convolutional layers in our proposed network. We use the Adam Gradient Descent optimizer with an initial learning rate of $1e^{-4}$ to train the MSTI-related networks from scratch. The drop out ratio is set to 0.5 and the weight decay rate is set to $5e^{-5}$. The gradient descent optimizer has the $1e^{-5}$ initial learning rate, and it is adopted with a momentum of 0.9 to train our MSTI-Net initialized with the Kinetics-400 and ImageNet-1k pre-trained model. To prevent over-fitting, we further employ a higher drop out ratio of 0.9 and a weight decay rate of $5e^{-4}$.

method	**UCF-101**	**HMDB-51**
P3D-B[Qiu *et al.*, 2017]	86.9%	60.8%
Multi-Scale Convolution(T)	88.4%	64.2%
Multi-Scale Convolution(S)	89.6%	64.7%
Multi-Scale Convolution(ST)	90.7%	67.3%
MSTI	**92.8**%	**70.4**%

Table 2: Ablation study. Performance of our proposed MSTI tube compared with P3D-B baseline and multi-scale convolution on UCF-101 and HMDB-51. They use the same network backbone and they are all pre-trained on Kinetics-400.

4.2 Ablation Study

To demonstrate the effectiveness of each component of our proposed MSTI tube, we conduct a series of ablation experiments on UCF-101 and HMDB-51 datasets. We choose the P3D-B [Qiu *et al.*, 2017] architecture as our baseline, which is the only parallel structure in all three P3Ds. All architectures use the same network backbone (with 8 convolutional layers, 5 max-pooling layers, and 2 fully connected layers) and the same 3D input size for a fair comparison. All architectures were pre-trained on the Kinetics-400 dataset.

Table 2 provides the comparison results in terms of the Top-1 classification accuracy on both UCF-101 and HMDB-51 datasets. We notice that compared to the P3D-B baseline the multi-scale temporal convolution improves accuracy by 1.5% on UCF-101 and 3.4% on HMDB-51, and the multi-scale spatial convolution can improve the performance by 2.7% and 3.9% on UCF-101 and HMDB-51, respectively. Applying multi-scale convolution on both spatial and temporal dimensions can further improve the performance compared to the multi-scale temporal convolution and multi-scale spatial convolution, achieving 90.7% accuracy on UCF-101 and 67.3% accuracy on HMDB-51, which demonstrates that the multi-scale convolution is effective and can greatly improve performance. We also observe our final MSTI tube which contains the spatial and temporal CAW blocks gets the best performance, obtaining 92.8% and 70.4% accuracy on UCF-101 and HMDB-51, respectively. This can demonstrate the importance of integrating multi-scale spatial and temporal features, which aims to perform feature recalibration by selectively emphasizing informative spatial-temporal features and suppressing less useful ones. Overall, we verify the effectiveness of two proposed blocks (multi-scale convolution block and cross-scale attention weighted blocks) in our proposed MSTI tube and we can see that the MSTI tube gets a tremendous increase compared to the baseline P3D-B.

Method	Backbone	Input×clips number	Kinetics-400	#Params	FLOPs
C3D [Tran *et al.*, 2015]	-	$[16 \times 3 \times 112 \times 112] \times 1$	56.1%	79.0M	296.7G
LRCN [Donahue *et al.*, 2015]	-	$[25 \times 3 \times 224 \times 224] \times 1$	63.3%	9.0M	41.5G
ARTNet [Wang *et al.*, 2018a]	ResNet18	$[16 \times 3 \times 112 \times 112] \times 25$	69.2%	35.2M	25.7G
I3D-RGB [Carreira and Zisserman, 2017]	BN-Inception	$[All \times 3 \times 256 \times 256] \times 1$	71.1%	12.7M	544.4G
StNet [He *et al.*, 2018]	ResNet101	$[25 \times 15 \times 256 \times 256] \times 1$	71.4%	52.2M	310.5G
R(2+1)D-RGB [Tran *et al.*, 2018]	ResNet34	$[32 \times 3 \times 112 \times 112] \times 10$	72.0%	63.8M	152.4G
S3D [Xie *et al.*, 2018]	BN-Inception	$[All \times 3 \times 224 \times 224] \times 1$	72.2%	8.8M	518.6G
MRST-Net [Wu *et al.*, 2019a]	ResNet101	$[16 \times 3 \times 224 \times 224] \times 20$	74.1%	31.7M	99.6G
CFST [Wu *et al.*, 2019b]	ResNet34	$[16 \times 3 \times 224 \times 224] \times 50$	74.6%	12.9M	14.1G
Nonlocal-I3D [Wang *et al.*, 2018b]	ResNet50	$[32 \times 3 \times 224 \times 224] \times 10$	74.9%	35.3M	163.3G
SlowFast [Feichtenhofer *et al.*, 2019]	ResNet50	$[64 \times 3 \times 224 \times 224] \times 10$	75.6%	32.9M	36.1G
MSTI(ours)	ResNet50	$[16 \times 3 \times 224 \times 224] \times 20$	**76.1%**	16.4M	46.3G

Table 3: Performance comparison with the state-of-the-art results on Kinetics-400 with only RGB frames as inputs. The dimensions of input are given by the number of frames in a clip, the number of channels, the frame height and width size. Here, "All" means using all frames in a video. Our detailed MSTI-Net architecture is shown in Table 1. #Params means the total number of model parameters and FLOPs means floating point operations which both are the significant indicators to measure the computational cost.

4.3 Comparison to the State-of-the-Art Methods

We further demonstrate the advances of our proposed MSTI-Net in comparison with state-of-the-art methods for human action recognition. For fair comparison, all methods use only RGB inputs. The detailed structure of our MSTI-Net is shown in Table 1. We uniformly sample 20 clips per video and average these clip predictions to obtain the final video prediction. The results on Kinetics-400, UCF-101 and HMDB-51 are shown in Tables 3 and 4, respectively.

Results on Kinetics-400. Table 3 shows the performance comparison of our proposed MSTI-Net (pre-trained on ImageNet-1k) against ten state-of-the-art methods in terms of Top-1 classification accuracy on Kinetics-400. The proposed MSTI-Net achieves 76.1% Top-1 classification accuracy, and the total number of parameters is 16.4M and the FLOPs is 46.3G. We can see that our MSTI-Net surpasses existing methods, improving the baseline C3D network[Tran *et al.*, 2015] by 20.0% at Top-1 classification accuracy. Moreover, our MSTI-Net improves the second best compared method SlowFast[Feichtenhofer *et al.*, 2019] by 0.5% in terms of Top-1 classification accuracy. In addition to this, the total number of parameters and FLOPs of our MSTI-Net are much fewer than those of most methods in the table. Compared to the C3D baseline, the MSTI-Net has only $1/5$ of the parameters and $1/6$ of the FLOPs. In short, the comparison indicates that our MSTI-Net can learn more effective spatial-temporal features much more efficiently.

Results on UCF-101 and HMDB-51. We also evaluate the fine-tuning MSTI-Net (pre-trained on ImageNet-1k and Kinetics-400) on UCF-101 and HMDB-51 datasets to investigate the generality and robustness. From Table 4, we can observe that our proposed MSTI-Net outperforms all the existing state-of-the-art methods with only RGB inputs on both UCF-101 and HMDB-51, which obtains 97.1% Top-1 classification accuracy on UCF-101 and 76.8% Top-1 classification accuracy on HMDB-51. Compared to the second best method R(2+1)D-RGB[Tran *et al.*, 2018] on UCF-101, our MSTI-Net improves 0.3% Top-1 classification accuracy. Compared to the second best method MRST-Net[Wu *et al.*, 2019a] on HMDB-51, our MSTI-Net gets 1.4% Top-1 classification accuracy improvement.

Method	UCF-101	HMDB-51
Two-stream[Simonyan and Zisserman, 2014]	73.0%	40.5%
C3D[Tran *et al.*, 2015]	82.3%	51.6%
ST-ResNet-50[Feichtenhofer *et al.*, 2016]	82.3%	48.9%
ST-ResNet-152[Feichtenhofer *et al.*, 2016]	83.4%	46.7%
TSN[Wang *et al.*, 2016]	85.7%	54.6%
Res3D[Tran *et al.*, 2017]	85.8%	54.9%
P3D ResNet[Qiu *et al.*, 2017]	88.6%	-
MiCT-Net[Zhou *et al.*, 2018]	88.9%	63.8%
ARTNet[Wang *et al.*, 2018a]	94.3%	70.9%
I3D-RGB[Carreira and Zisserman, 2017]	95.6%	74.8%
R(2+1)D-34-RGB[Tran *et al.*, 2018]	96.8%	74.5%
MRST-Met[Wu *et al.*, 2019a]	96.5%	75.4%
MSTI(ours)	**97.1%**	**76.8%**

Table 4: Action recognition accuracy on UCF-101 and HMDB-51, averaged over three splits. The top part of the table refers to related methods with the Sports-1M pre-trained, the lower part refers to related methods with the Kinetics-400 pre-trained.

5 Conclusion

In this work, we address the problem of building highly efficient deep neural networks for human action recognition from the perspectives of generating multi-scale representations and integrating multi-scale spatial-temporal features. We propose a novel Multi-Scale Spatial-Temporal Integration Convolutional Tube in which the multi-scale convolution block generates multi-scale spatial appearance and temporal motion, and the cross-scale attention weighted blocks perform feature recalibration by integrating multi-scale spatial and temporal features. Benefiting from the two blocks, our MSTI-Net requires significantly less computational resources yet achieving the state-of-the-art action recognition accuracy.

Acknowledgments

This work was supported by the National Key R&D Program of China under Grant 2017YFB1300201, the National Natural Science Foundation of China (NSFC) under Grants U19B2038, 61620106009 and 61725203 as well as the Fundamental Research Funds for the Central Universities under Grant WK2100100030.

References

[Carreira and Zisserman, 2017] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017.

[Donahue *et al.*, 2015] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2625–2634, 2015.

[Feichtenhofer *et al.*, 2016] Christoph Feichtenhofer, Axel Pinz, and Richard Wildes. Spatiotemporal residual networks for video action recognition. In *NIPS*, pages 3468–3476, 2016.

[Feichtenhofer *et al.*, 2019] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, 2019.

[Gao *et al.*, 2019] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *arXiv preprint arXiv:1904.01169*, 2019.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[He *et al.*, 2018] Dongliang He, Zhichao Zhou, Chuang Gan, Fu Li, Xiao Liu, Yandong Li, Liming Wang, and Shilei Wen. Stnet: Local and global spatial-temporal modeling for action recognition. *arXiv preprint arXiv:1811.01549*, 2018.

[Hu *et al.*, 2018] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018.

[Karpathy *et al.*, 2014] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, pages 1725–1732, 2014.

[Kay *et al.*, 2017] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[Kuehne *et al.*, 2013] Hilde Kuehne, Hueihan Jhuang, Rainer Stiefelhagen, and Thomas Serre. Hmdb51: A large video database for human motion recognition. In *HPCSE*, pages 571–582. Springer, 2013.

[Liu *et al.*, 2016] Jiawei Liu, Zheng-Jun Zha, QI Tian, Dong Liu, Ting Yao, Qiang Ling, and Tao Mei. Multi-scale triplet cnn for person re-identification. In *ACM Multimedia*, pages 192–196, 2016.

[Nair and Hinton, 2010] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, pages 807–814, 2010.

[Qiu *et al.*, 2017] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, pages 5533–5541, 2017.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014.

[Soomro *et al.*, 2012] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[Szegedy *et al.*, 2017] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.

[Tran *et al.*, 2015] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015.

[Tran *et al.*, 2017] Du Tran, Jamie Ray, Zheng Shou, Shih-Fu Chang, and Manohar Paluri. Convnet architecture search for spatiotemporal feature learning. *arXiv preprint arXiv:1708.05038*, 2017.

[Tran *et al.*, 2018] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, pages 6450–6459, 2018.

[Wang *et al.*, 2016] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, pages 20–36. Springer, 2016.

[Wang *et al.*, 2018a] Limin Wang, Wei Li, Wen Li, and Luc Van Gool. Appearance-and-relation networks for video classification. In *CVPR*, pages 1430–1439, 2018.

[Wang *et al.*, 2018b] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018.

[Wu *et al.*, 2019a] Haoze Wu, Jiawei Liu, Zheng-Jun Zha, Zhenzhong Chen, and Xiaoyan Sun. Mutually reinforced spatio-temporal convolutional tube for human action recognition. In *IJCAI*, pages 968–974, 2019.

[Wu *et al.*, 2019b] Haoze Wu, Zheng-Jun Zha, Xin Wen, Zhenzhong Chen, Dong Liu, and Xuejin Chen. Cross-fiber spatial-temporal co-enhanced networks for video action recognition. In *ACM Multimedia*, pages 620–628, 2019.

[Xie *et al.*, 2018] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*, pages 305–321, 2018.

[Zhou *et al.*, 2018] Yizhou Zhou, Xiaoyan Sun, Zheng-Jun Zha, and Wenjun Zeng. Mict: Mixed 3d/2d convolutional tube for human action recognition. In *CVPR*, pages 449–458, 2018.