Dense Dilated Network for Video Action Recognition

Baohan Xu, Hao Ye, Yingbin Zheng, Heng Wang, Tianyu Luwang, Yu-Gang Jiang

***Abstract*—The ability to recognize actions throughout a video is essential for surveillance, self-driving, and many other applications. Although many researchers have investigated deep neural networks to get a better result in video action recognition, these networks usually require a large number of well-labeled data to train. In this paper, we introduce a dense dilated network to collect action information from snippet-level to global-level. The dilated dense network is composed of the blocks with densely-connected dilated convolutions layers. Our proposed framework is capable of fusing outputs from each layer to learn high-level representations, and these representations are robust even with only a few training snippets. We study different spatial and temporal modality fusing configurations and introduce a novel temporal guided fusion upon the dense dilated network which can further boost the performance. We conduct extensive experiments on two popular video action datasets: UCF101 and HMDB51. The experiments demonstrate the effectiveness of our proposed framework.**

***Index Terms*—Action recognition; dilated convolution; two-stream fusion; video analysis.**

I. Introduction

Video analysis has drawn a significant amount of attention over the past few years with the prevalence of video capture devices and the surge of the Internet, and recognize the video action is of fundamental importance in many video analysis applications [1]–[6]. Hand-crafted features are proposed to incorporate spatial and temporal information in the early years [7]–[9]. With recent progress by applying convolutional neural networks (ConvNets) to computer vision tasks such as image classification and object detection [10]–[13], deep ConvNets are introduced to video action recognition fields with the capacity of learning discriminative representations. However, state-of-the-art video action recognition methods are still far inferior to human performance.

One major challenge in video action recognition is the lack of a certain amount of well-labeled training data. Due to the complexity and diversity of videos, video action recognition system requires orders of magnitude larger training data, compared with image recognition. And constructing large-scale video datasets requires a considerable amount of efforts and resources for collection and annotation. In addition, although the actions may belong to the same action class, they may have various backgrounds and different point of views; some example frames from the videos in the UCF101 dataset [14] are shown in Fig. 1. It is essential to investigate a deep neural architecture to capture discriminative features of different action categories.

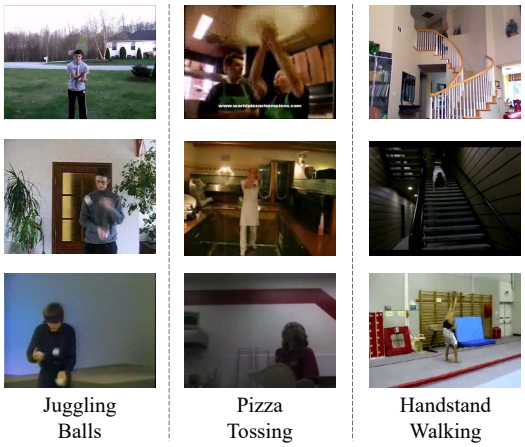


Fig. 1. Some example frames of video action dataset UCF101 [14]. Although the same class of videos contains high-level semantic correlations, different backgrounds and lack of quality control makes action recognition more challenging.

Another challenge of video recognition is that many neural networks focus on explore features from still images. These methods have difficulties in distinguishing some similar actions such as walking and running, due to the ambiguous individual frames. Combining image and optical flow scores after the classification has also not fully take advantage of spatial-temporal features. Thus, how to integrate spatial and temporal cues at a different level is worth to investigate.

To tackle these challenges, we propose a novel dense dilated neural network, which is able to preserve both spatial and temporal information with limited training data. The dense dilated network is consist of pre-trained feature extractors, dense dilated blocks, and classifiers. Two ConvNets are used to extract spatial and temporal representations based on still images and stacked optical flows. The densely connected layers in dense dilated blocks restore all inputs from proceeding layers and send these feature maps to consecutive layers.

We also explore different ways to connect dilated dense blocks and how to fuse spatial and temporal branches. Rather than integrating spatial and temporal features at softmax layer

This work was supported in part by the National Natural Science Foundation of China under Grant 61602459. *(Baohan Xu and Hao Ye contributed equally to this work.) (Corresponding author: Yu-Gang Jiang.)*

B. Xu and H. Wang are with ZhongAn Technology, Shanghai, China. (e-mail: xbh.fdu@gmail.com, wangheng@zhongan.io)

H. Ye and Y. Zheng are with Videt Tech, Shanghai, China. (e-mail: hao.ye@videt.cn; yingbin.zheng@videt.cn)

T. Luwang is with Shanghai Lunion Intelligent Technology Co., Ltd. (LunionData), Shanghai, China. (e-mail: lwty@luniondata.com)

Y.-G. Jiang is with School of Computer Science, Fudan University, and Jilian Technology Group (Video++), Shanghai, China. (e-mail: ygj@fudan.edu.cn)

or input layer, fusing them at a convolution layer can also fully utilize static and motion information. The choices of layers in dense dilated blocks and numbers of dense dilated blocks are also practiced to prevent over-fitting with the limited number of training samples. We report the accuracy of different architectures as well as the comparison with the state-of-the-art approaches on two challenging video action benchmarks, e.g., UCF101 and HMDB51. Our contributions are summarized as follows:

- We propose a novel deep neural network architecture combining dilated temporal convolution and densely layer connection to perform video action recognition. Our framework can generate new class prototypes while only a small number of new action videos are needed.
- The dilated dense blocks in our framework are able to capture spatial-temporal information from both snippet-level and long-term context.
- We explore the fusion of spatial and temporal network which we haven't explored in our preliminary work [15]. We get the insights of the fusion strategies in different stages of the network and demonstrate that adding temporal guided connection gives more complementary features with the spatial network.
- We achieve the state-of-the-art action recognition performance on UCF101 and HMDB51 by only using part of the training videos and then exhibit the effectiveness of dense connection blocks on few shot action recognition.

The rest of our paper is arranged as follows. We begin with a brief review of the literature on action recognition and few shot learning in Section II. In Section III, we give the details of our framework. And Section IV explores the variations of the dense dilated network, including implementation stages of dense connections and two stream fusion configurations. We show the experimental results conducted on UCF-101 and HMDB51 to demonstrate the effectiveness in Section V. Section VI concludes the paper.

II. Related Work

A. Video Action Recognition

Video analysis has drawn more and more attention with the rapid development of smart devices and the Internet. We can roughly classify the methods of video action analysis into hand-crafted feature based methods and deep learning methods.

1) Hand-crafted features: Early researches have explored the interest points detection and representation. The histogram of gradient (HoG) [16], space-time interest points (STIP) [7], and histogram of optical flow (HOF) [8] are introduced to extract both still image and temporal representations. Other video analysis tasks such as video summarization are also benefited from the visual features (e.g., [17]). To collect more motion information, dense trajectory [9] is proposed to densely sample and track each pixel within the dense optical flow. Nevertheless, when facing a significant amount of training data, the hand-crafted features lack flexibility and scalability.

2) Deep learning methods: The second group of previous works is mainly based on the deep ConvNets since it achieves great success in some computer vision tasks recently. Since the introduction of AlexNet [10] in 2012, the performance of ImageNet Challenge [18] has been dramatically improved. Many ConvNets architectures are introduced to perform image classification and object detection. For example, Simonyan et al. [11] proposed the VGG net by extending the layers in the network and won the 2014 ImageNet Challenge. As the number of layers in the network growing rapidly, the accuracy has not increased only using stacked convolutional layer. Thus He et al. [12] proposed a residual function, which takes the layer inputs into consideration. With a total of 152 layers, the performance still can be improved from the significantly increased depth. Other researchers such as Huang et al. [13] developed a dense connection between layers to take full advantage of features among different layers. However, despite the features from still images, these architectures are difficult to capture temporal information, which is of significant importance in video action analysis task.

To handle these problems, the two-stream approach has recently been employed in several action recognition methods. Simonyan et al. [1] proposed a fusion network, which first decomposes video into spatial and temporal components by using RGB and optical flow frames. These components are fed into separate deep ConvNet architectures, to learn spatial as well as temporal information about the appearance and movement of the objects in a scene. The two-stream approach has recently been incorporated in several action recognition systems [2], [19]–[22], and competitive results were attained on popular video classification benchmarks, such as HMDB51 [23], UCF101 [14], and ActivityNet [24]. To date, this pipeline is the most effective approach of applying deep learning to action recognition, especially with limited training data. The potential problem of these two-stream approaches is, feeding videos into spatial and temporal network separately may omit the complementary of the two streams.

Many studies paid attention to how to use temporal information to achieve multi-modality recognition. Zhao et al. [25] extracted frame and optical flow features then pooled with two pooling strategies. Tran et al. [26] adopted 3D CNN (C3D) for joint learning spatial-temporal representations in video datasets. Later in [27], Qiu constructed 2D spatial convolutions and 1D temporal connections to stimulate 3D convolutions. The deep residual learning framework also designed for efficiently training a deeper neural network. Instead of using traditional convolutions, Lea et al. [28] explored temporal convolutions for action segmentation tasks. Encoder-decoder temporal convolutional network (TCN) and dilated TCN were used to capture long-range temporal patterns. However, these approaches still require a large amount of data and resources to get a satisfactory result.

B. Few Shot Learning

The limit well-labeled data impedes accuracy gained from the deeper neural network. A few years ago, some researchers already studied few shot learning in video action recognition

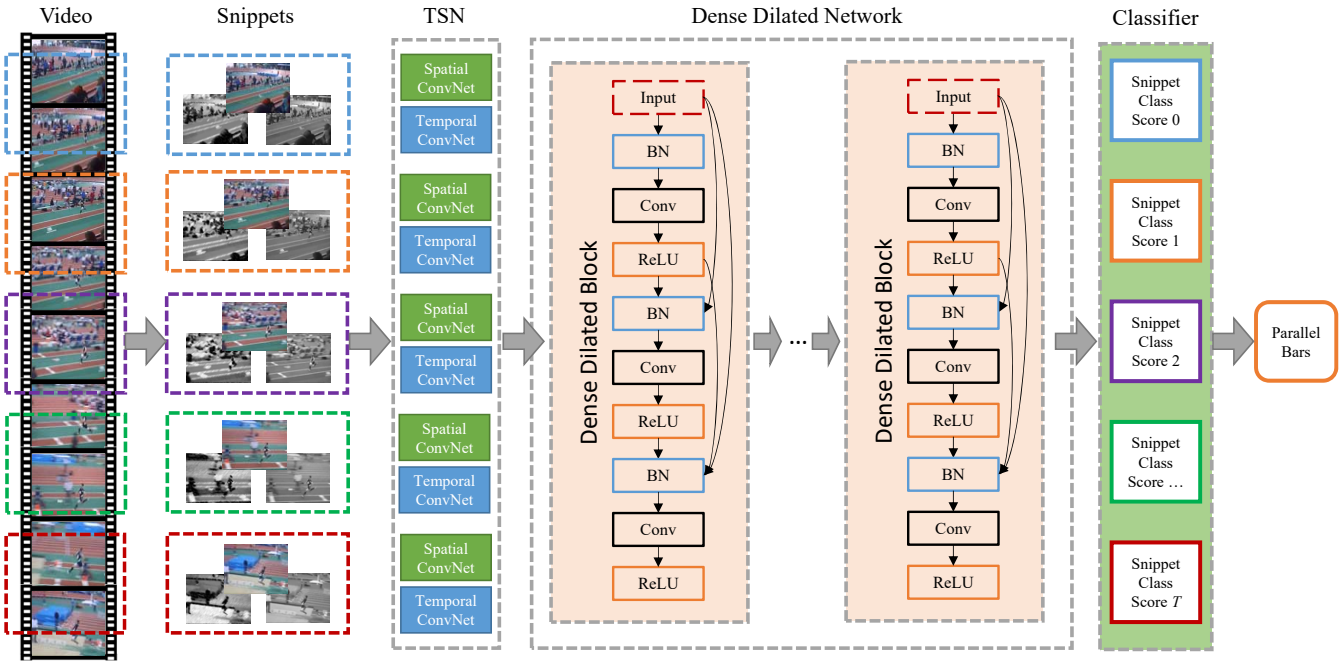


Fig. 2. The proposed framework. Videos are divided into n snippets. Temporal segment network (TSN) [28] is used to extract spatial-temporal features of every snippet. Dense dilated blocks are utilized to densely connect channel wise features. The dilated convolutions help explore temporal relations between different snippets. Prediction scores are made based on all the feature maps in the network, and the scores of all the snippets are leveraged to perform video-level prediction.

task. However, the early studies [29], [30] mainly focused on the dataset such as KTH [31] and Weizmann [32], which contains several fixed actions performed by actors such as walking and jogging. Some approaches explored the Hidden Markov Model (HMM) and encoding scheme and achieve satisfactory results on the small datasets [33]. When facing large-scale real-world data, these methods lack scalability.

More recently, transfer learning methods are used due to well-annotated image data [34] on more larger video benchmarks, such as UCF101 [14] and HMDB51 [23]. For instance, Li et al. [35] proposed a video mapping method to encode videos into a low-dimensional representation with the help of spatial attention map. These frameworks demonstrated some promising attempts on the challenging benchmarks, which have diverse content and lack of quality control. Nevertheless, these methods require data from other related domains, such as image and text information, to help recognition rather than only using video data. And the disadvantage of these methods is that the significant temporal and motion clues are omitted.

Inspired by previous works, we introduce a dense dilated framework to perform video action recognition. Rather than building a deeper or more complicated neural network, we explore how to take full advantage of the dense connection to learn inner-class semantic features. Dilated temporal convolution helps expand the receptive fields. Besides, the dense dilated architecture can converge only using a few layers. Besides, the complementary of spatial and temporal features are investigated by the two stream fusion operation.

A preliminary version of our system was described in [15]. In this work, we explore the fusion of spatial and temporal streams which we haven't explored in our preliminary work, and get the insights of the fusion strategies in different stages of the network. We also extend [15] with state-of-the-art results on one additional dataset, a comprehensive experiment of the fusion and block settings, and extensive analysis for all evaluated datasets.

III. Dense Dilated Network

A. Architecture

Fig. 2 demonstrates the architecture of our framework. Our framework is inspired by the DenseNet [13]. Instead of the traditional feed-forward network, a dense connection on channel level provides high-information flow in the whole network. This simple but novel design helps the improvement on some computer vision tasks even facing small scale of training data. The basic block in this paper is also built upon the dense connection between the layers, in order to take full advantage of spatial-temporal features.

Formally, for each video, the whole video is segmented into 25 snippets equally. The static video frame images and the optical flow images are extracted of each snippet. All the images are feed into Temporal Segment Network (TSN) [28] to generate spatial features (based on the video frame) and temporal features (based on the optical flow) separately. We then use the global pool features before the softmax layer as the clip presentations. Both the spatial and temporal features are fed to the *dense dilated blocks* (which will be described in detail in the next subsection). The output of the last dense dilated block can be regarded as the d-dimension representation of n snippets. We propose majority vote to generate the final video-level prediction. The network structure of the dense dilated network is listed in Table I.

TABLE I
THE STRUCTURE AND SIZE OF OUTPUT IN THE DDN. 25 REPRESENTS THE 25 SEGMENT REPRESENTATIONS WE EXTRACTED FROM TSN. THE 1024 DIMENSION OF SPACIAL FEATURES CONCATENATED WITH 1024 DIMENSION OF TEMPORAL FEATURES TO FORM THE INPUT OF DENSE DILATED NETWORK. THE 101 IN THE SOFTMAX LAYER REPRESENT THE 101 ACTION CLASSES IN UCF101. THE GROWTH RATE IS SET TO BE $k = 12$.

Layers	Output Size
TSN	25×2048
Init Convolution	25×256
Dense Dilated Block 1	25×128
Dense Dilated Block 2	25×64
Dense Dilated Block 3	25×32
Softmax	$101D$ fully-connected

B. Dense Dilated Blocks

The dense dilated network is consist of several dense dilated blocks. Fig. 3 illustrates the detailed structure within dense dilated block, which is composed of three operations: a batch normalization [36], a convolutional layer, and a rectified linear unit (ReLU) [37] function[1]. The use of dense connection can prevent gradient dispersion due to the stack of feature maps. Furthermore, the filters in each dense layer are often very small, which can not only reduce the size of the network but also has a regularizing effect [13], which can reduce the overfitting, especially with the smaller amount of training data.

Besides the dense connection, we also incorporate the dilation during the convolution operation, which is more and more widely used in recent computer vision applications [38]–[42]. Different dilation parameters enable the network to get a different scale of temporal information. More specifically, for a dense dilated block with L subsequent layers, dilated convolution layer l has a growing dilated rate parameter $s_l = 2^l$ $(l = 1, ..., L)$. Adding dilation in traditional convolution layer is able to discover long-term relations in different snippets, which can provide larger receptive fields to enhance the final recognition. Thus, the convolution operations are used on two-time steps, t and $t - s$. Formally, the weights of the filters can be regarded as $W = \{W^{(1)}, W^{(2)}\}$. The dilated convolution can be defined as

$$x_t^l = f(W^{(1)} x_{t-s}^{l-1} + W^{(2)} x_t^{l-1} + b), \tag{1}$$

where x_t^l represents the results of dilated convolution on time t in layer l. The parameter b refer to the bias vector.

The dense connection concatenates feature maps in channel level of all formal layers, which can provide rich information flow to the next layer. Even facing the small scale of training examples, our framework can still get more insight into inner-class similarity. Similar to the growth rate in DenseNet, we set the same number of filters k^i in each dense dilated blocks B^i. The output representation S_t^i of each block B^i on time step t can be referred as

$$S_t^i = [x_t^l, x_t^{(l-1)}, \ldots, x_t^0], \tag{2}$$

where S_t^i concatenate the feature maps of all layers into a single tensor.

[1]The figure is with three convolutional layers, and the effect of different number of layers will be investigated in Section V-A1.

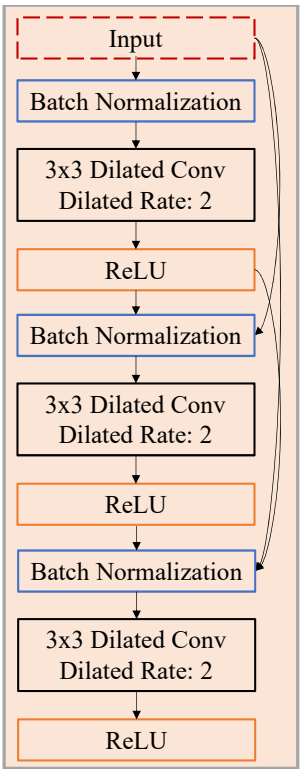


Fig. 3. The framework of temporal dense convolution with 3 layers. Every layer consists of a batch normalization, a dilated convolution, and an activation function. Different layers are concatenated in a dense way.

C. Transition Layers

To reduce the size of the network, the transition layers are also added between dense dilated blocks. The layer is consist of a batch normalization operation, a 1×1 convolution operation in order to downscale the size of the feature maps.

D. Growth Rate

We refer k as growth rate of different blocks. Each layer produce k^i feature-maps in the i-th block, then the filters of different blocks can write down as

$$k^i = k^{i-1} \times (l - 1) + k^0, \tag{3}$$

where k^0 refer to the number of channels in the input layer, l is the number of layers in each block. On limited training data, we show that a relatively small growth rate and a small number of layers and blocks are sufficient to get high accuracy.

IV. VARIATIONS OF NEURAL NETWORK

In this section, we explore variations of our proposed dense dilated network. Two variations of our basic framework are investigated: the first is with different strategies to aggregate dense dilated blocks and leverage the feature of entire video and the snippets, and we also study the fusion of the spatial and temporal streams.

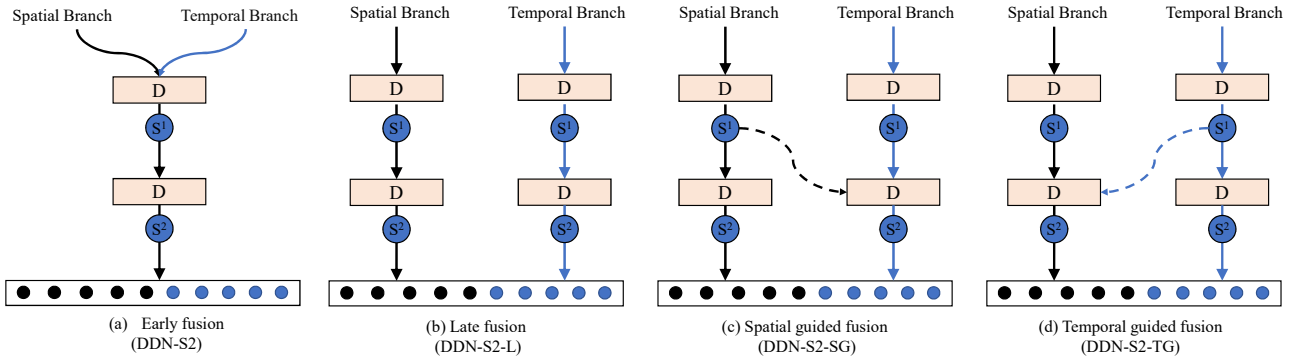


Fig. 5. Different two stream fusion strategies. The black line represents spatial guided network. The blue line represents temporal guided network. And another fusion way is concatenate spatial and temporal features.

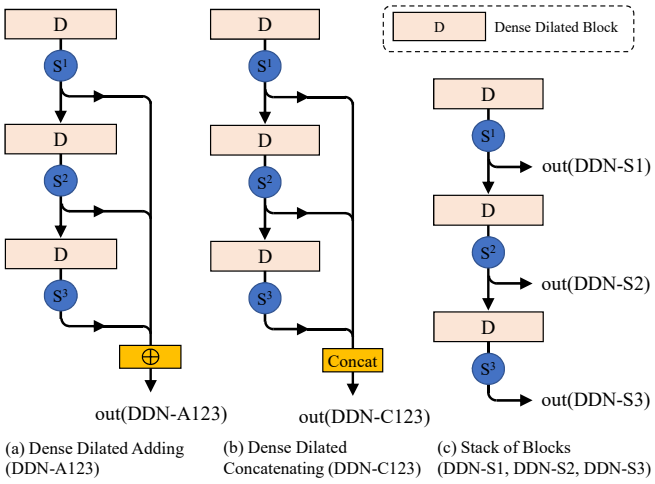


Fig. 4. Different dense dilated block aggregation strategies. (a) is summed using a set of skip connections. (b) is concatenated over the output of different output blocks. (c) are more straightforward to extract every block as snippet-level representation.

A. Dense Dilated Aggregation

We introduce different aggregation ways of dense dilated blocks. The number of basic blocks is empirically set to 3, and the strategies are demonstrated in Fig. 4, which including adding the outputs of the blocks, concatenating the outputs of the blocks and extracting features of different blocks separately. Although the deeper network and more blocks may give more insights into the whole video, the different level of output can also provide different scales of information. The following explains the details of each strategy.

1) Adding layers: In the first place, we explore adding the outputs of the dense dilated blocks. The outputs of different blocks are aligned to the same shapes via 1×1 convolutions. The *dense dilated adding* (DDN-A) strategies is shown more clearly in Fig. 4(a). Thus, the final representation can be regarded as:

$$Z_t = ReLU(V \sum_{i=0}^{B-1} S_t^i + e), \tag{4}$$

where V is the weight matrix and e is the bias.

2) Concatenating layers: We also study concatenating outputs of each block to enlarge the receptive fields in the final representation. Compare to other aggregating strategies, concatenating provide a wider range of channels. It shows in Fig. 4(b) as *dense dilated concatenation* (DDN-C).

$$Z_t = ReLU([S_t^{B-1}, S_t^{B-2}, \ldots, S_t^0] + e). \tag{5}$$

3) Single block feature: We try to extract the output feature of different blocks directly, as illustrated in Fig. 4(c). The strategies are defined as DDN-S1, DDN-S2, and DDN-S3 respectively. This method can help us get more insights into the neural network.

Due to limited training data, the way of adding or concatenating may income overfitting while the feature extraction in the previous layer may provide enough information and makes the network more easy to fit the testing data.

B. Two-stream Fusion

Recent years, researchers found that treating videos as static frames may lose discriminative motion information. Therefore, optical flow is introduced into video analysis. Besides directly fusing image features and optical flow features, the two-stream configuration also becomes popular in video recognition. The two-stream architecture gives separate weights for image branch and optical flow branch. The spatial branch captures significant local information for different classes, while the temporal branch focuses on continuous motion features. Many approaches, such as two-stream CNN and TSN, still train two separate models and only aggregate the spatial and temporal scores for recognition. In this paper, we would like to evaluate the ways of fusing spatial and temporal branches and explore the optimal design to achieves better performance. Traditional early fusion and late fusion methods are first considered. The *early fusion*, which is the baseline in this work as it is with less parameters, combine the spatial and temporal features at the input layer (Fig. 5(a)), while the *late fusion* concatenate the outputs from the two streams (Fig. 5(b)). Here we also propose two novel fusion approaches. The *spatial guided fusion* fuses the first dense block of the spatial branch into the temporal branch, to ensure that the temporal branch receives both temporal features and additional spatial features

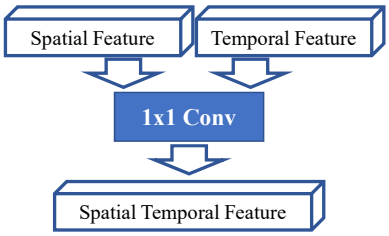


Fig. 6. Two stream fusion diagram. The spatial and temporal streams are concatenated then feed into 1×1 convolution.

(Fig. 5(c)). The *temporal guided fusion* tries to utilize the additional temporal information by fusing the first dense block of the temporal branch into the spatial branch (Fig. 5(d)). Specifically for the spatial guided and temporal guided fusion, the output features are concatenated using 1×1 convolution to align into the input features. We will evaluate the fusion methods in the next section. Figure 6 elaborates the fusion of feature maps.

V. Experiments

We evaluate the framework of dense dilated network on two video recognition benchmarks: UCF101 [14] and HMDB51 [23]. UCF101 [14] consists of 101 action classes, 13,320 clips and 27 hours of unconstrained videos collected from YouTube. The large variations in camera motion, different background, and illumination conditions make it a challenge video action dataset. Fig. 7(a) shows some example frames of different categories in UCF101. HMDB51 [23] is also a popular video action dataset. It collected mostly from movies with 6849 clips divided into 51 action categories, each containing a minimum of 101 clips. It remains challenging to recognize due to low-quality control and a small amount of training set. Samples from HMDB51 are displayed in Fig. 7(b).

We conduct the extensive evaluation, including the impact of settings of the dense dilated block, the fusion strategy, and the few-shot learning, on the UCF101 dataset. And we also compare our methods with the state-of-the-arts on HMDB51. We follow the evaluation protocols of these benchmarks. For each dataset, the experiments are separately done on the three public train/test splits for comparison, and the final results are averaged across three test splits.

Implementation details. For the TSN, we choose the Google Inception network [36] as the basic classification network for both spatial and temporal stream, And the spatial and temporal TSN models are pre-trained on Kinetics dataset [43] with 400 action classes and a total of 300,000 video clips. Each video contains 25 snippets as the default setting in TSN. The growth rate K of all the block is set as 12 to leverage the accuracy and the computation cost. The learning rate of the network is set to be 0.001 and the training epochs are set to be 15. The network is implemented by PyTorch[2].

A. Experiments on UCF101

[2]https://pytorch.org/

1) Ablation Study: In this section, we perform an ablation study on the UCF101 dataset. We start by exploring the hyper-parameters in our frameworks, such as the number of layers and the growth rate, which may affect the performance. Then the variations of the network architecture are evaluated to find the optimal setting of the layer aggregation and fusion strategies. Throughout the experiments, we use **DDN** to represent our dense dilated network, and define our framework as follows for presentation simplicity.

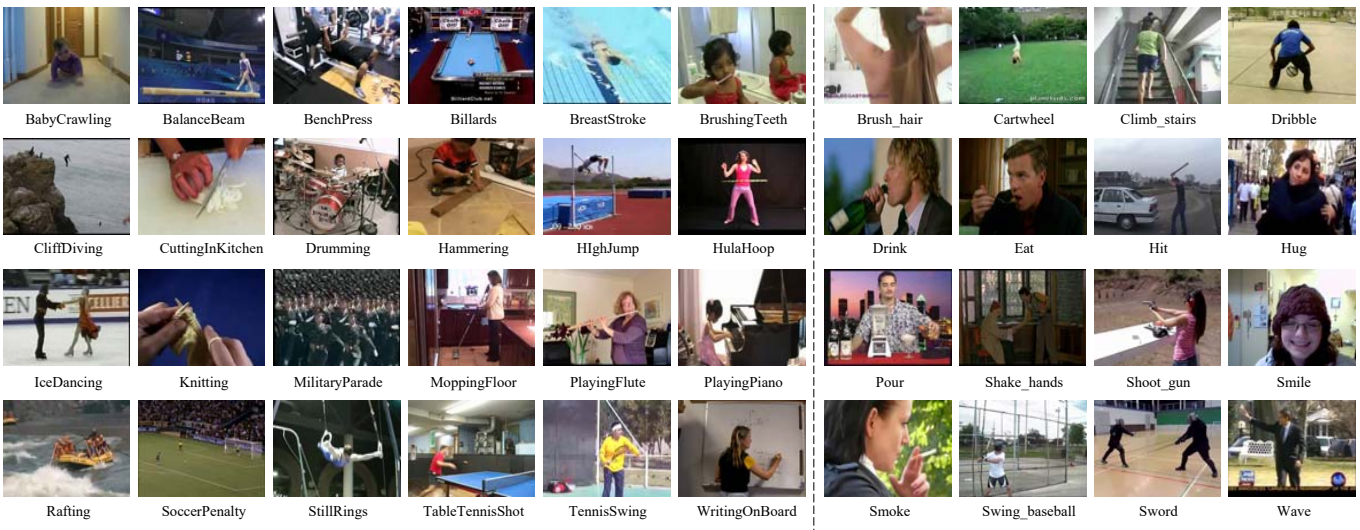
- DDN-A123/A12: We define **A** as adding the output of blocks (see Fig. 4(a)). The numbers represent which blocks are added.
- DDN-C123/C12: Same as above, **C** represents concatenating of blocks (Fig. 4(b)).
- DDN-Sn: As displayed in Fig. 4(c), **S** represent single block feature extraction method, and n denotes the dense dilated block we extract features from.
- DDN-S2-L/SG/TG: Recall that we define four fusion strategies in Section IV-B. The default fusion strategy is the early fusion of spatial and temporal branch (Fig. 5(a)), thus we define the methods without suffix are based on early fusion. The methods with suffix L, SG, and TG are the late fusion, spatial guided fusion, and temporal guided fusion respectively.
- Baseline experiments: We also conducted baseline experiments with only dense connection (Dense-S2) or dilated convolution (Dilated-S2).

Number of layers in each dense dilated block. We compare the performance of our frameworks with a varying number of layers L in the dense dilated block. In Fig. 8(a), lines show accuracy for L from 2 to 4. All the methods perform best at $L = 3$. Compared to $L = 2$, more layers in each block can boost the performance. However, when L gets larger, the performances of both algorithms are decreased. This is because complex neural networks lead to underfitting when the scale of training data is small.

Growth rate. Fig. 8(b) shows how different growth rates affect the accuracy. Due to limited data, our dense dilated architecture can converge with relatively small growth rate. When $K = 12$ and $K = 16$, we can achieve relatively high accuracy without drastically increasing the number of parameters for all the frameworks.

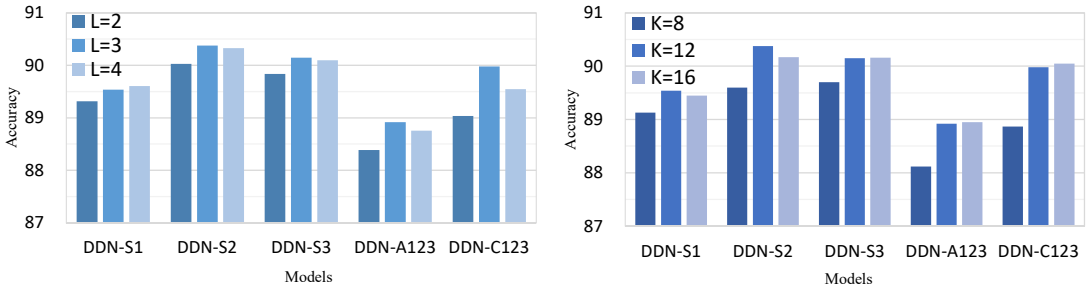
The influence of Kinetics. To further validate whether the performance of our proposed methods is affected by the original training dataset Kinetics [43], we add another simple experiment to recognize the classes in UCF101 but not in Kinetics. According to manual visual inspection, about 45 classes in UCF101 are not the same as the classes in Kinetics. We use DDN-S2 to train 20% of training videos in the 45 classes and make the evaluation. The accuracy is 86.23% of 45 classes. This shows our model can recognize video categories other than those in the pre-trained models.

Block aggregation strategy. Inspired by DenseNet, we first proposed dense dilated network with RGB and optical flow features concatenated as input. The several aggregation ways



(a) UCF101 (b) HMDB51

Fig. 7. Example frames of (a) UCF101 and (b) HMDB51. These two datasets are benchmark datasets in action recognition with various action classes.



(a) Number of layers (L) (b) Growth Rate (K)

Fig. 8. Evaluation of Dense Dilated Network parameters on UCF101 dataset, (a) recognition accuracy w.r.t. the number of layers; (b) recognition accuracy with various growth rate.

TABLE II
THE ACCURACY(%) AND THE AMOUNT OF PARAMETERS OF DIFFERENT DESIGN OF DENSE DILATED NETWORK ON UCF101 DATASET.

	Method	Accuracy	Params
(a)	DDN-S1	95.17	$0.62M$
	DDN-S2	**96.15**	$0.62M$
	DDN-S3	95.48	$0.62M$
(b)	DDN-A12	95.47	$0.36M$
	DDN-C12	**95.76**	$0.38M$
	DDN-A123	94.74	$0.62M$
	DDN-C123	95.30	$0.67M$
(c)	DDN-S2-L	96.26	$1.06M$
	DDN-S2-SG	95.66	$1.06M$
	DDN-S2-TG	**96.85**	$1.06M$
(d)	Dense-S2	95.37	$0.62M$
	Dilated-S2	95.19	$0.22M$

described in Fig. 4 are explored, which is the first variation of the network. With all data in UCF101, (a) and (b) in Table II reports the accuracy of different aggregation framework. Compared among three strategies, we can clearly find that the worst result is from DDN-A, while DDN-C shows better result due to wider reception fields via concatenating channels of feature maps. One interesting finding is that DDN-S achieves the best performance with different numbers of blocks compare to DDN-A and DDN-C.

In the setting of different numbers of blocks, the results show that all three architectures achieve best results with two dense dilated blocks. This finding demonstrates that two dense dilated blocks are enough for video action recognition on a relatively small dataset. Using the third block may not only contribute to overfitting but also lose some essential information during the pooling operation. Besides, we also conduct baseline experiments only using dense connection or dilated convolution in Table II(d). The results show that dense connection would make more contributions to action recognition. It worth noticing that we also try different combinations of aggregation and concatenation operations. We report the results in Table III. The combinations do not show many improvements thus we continue using the DDN-S2 in the following experiments.

Fusion strategy. Furthermore, we try to study the two stream fusion. Fig. 5 shows the detail architectures of different fusion strategies. As reported in previous experiments, DDN-S2 achieves the best performance, therefore we choose two

Fig. 9. Qualitative examples of successful and failure examples on UCF101. The results are based on the DDN-S2-TG model with all the training data.

TABLE III
THE COMBINATION OF CONVOLUTION AND AGGREGATION ON UCF101.

Method	Accuracy
A12 + C	94.87
C12 + A	95.03
C + A23	94.78
A + C23	95.12

dense dilated blocks in this evaluation.

As reported in Table II(c), the late fusion (DDN-S2-L) achieves slightly better result than the early fusion (DDN-S2). And the late fusion still shows worse performance than temporal guided network (DDN-S2-TG), which demonstrates that adding backpropogation on temporal branch helps the network capture more discriminative features.

Among these architectures, temporal guided network (DDN-S2-TG) achieves the best performance while spatial guided network (DDN-S2-SG) performs worse. This indicates that keeping a temporal branch can provide significant information for final classification. Compare to spatial feature, temporal information may be even more complementary with spatial-temporal information, and only keeping spatial features may lose motion information and provide redundant features in video analysis.

TABLE IV
COMPARISON WITH STATE-OF-THE-ART METHODS ON UCF101.

Method	RGB	Flow	RGB+Flow
C3D [26]	85.20	-	-
P3D [27]	88.60	-	-
Two-stream fusion [44]	82.60	86.25	93.50
Two-stream MiCT-Net [45]	88.90	-	94.70
Temporal Segment Network [28][3]	85.70	87.90	94.20
Dilated TCN [38]	-	-	92.31
DDN-S2	**89.69**	**92.73**	96.15
DDN-S2-L	-	-	96.26
DDN-S2-SG	-	-	95.66
DDN-S2-TG	-	-	**96.85**

2) Comparison with state-of-the-arts: To evaluate our algorithm, we also compare ours with the state-of-the-art video action recognition methods, which also make use of spatial-temporal features.

- **C3D [26]** proposes a deep 3-dimensional convolution networks and shows effective for joint learning spatial-temporal feature.
- **P3D [27]** proposes a bottleneck block combing spatial-temporal information to simulate 3D convolutions and

[3]It worth noticing the reported results of TSN are from the original TSN paper. For fair comparison, we have finetuned the TSN with Kinetics pretrained model on UCF101 and HMDB51 with the same dataset splits and the center crop strategy. The results is 95.61% for UCF101 and 72.84% for HMDB51.

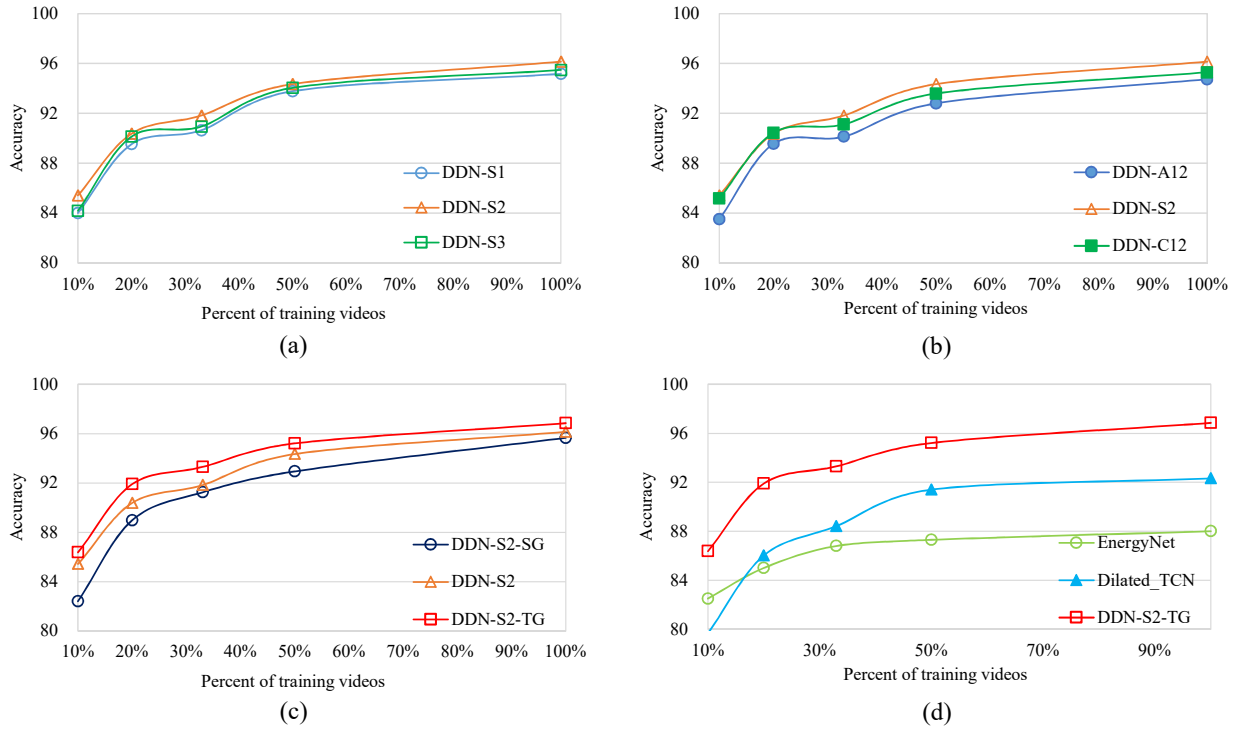


Fig. 10. Few shot video action recognition results on UCF101. (a) recognition accuracy with different blocks using DDN-S; (b) recognition accuracy with differen block aggregation strategies using 2 dense dialted blocks; (c) recognition accuracy of different fusion strategies; (d) recognition accuracy compared to baseline methods.

reduce the scale of parameters at the same time.

- **Two-stream fusion [44]** studies some ways of fusing ConvNet and finds that fuse spatial and temporal network in convolutional layers rather than the softmax layer can boost the performance.
- **Two-stream MiCT-Net [45]** proposes a mixed convolutional tube that integrates 2D CNN with the 3D convolution module. This method can generate deeper and informative feature maps meanwhile reducing training complexity.
- **Temporal segment network [28]** applies a sparse temporal sampling strategy to learn video-level representation. Two-stream is also considered to improve results.
- **Dilated TCN [38]** focuses on dilated temporal convolution. Residual connections and dilated convolutions are used to perform video segmentation and detection.

Table IV shows the results of video action recognition only using RGB frames and the temporal guided network. It clearly shows that our proposed temporal guided network outperform all baseline methods on both datasets when using all data. These results demonstrate that adding dense connection is able to retrieve features from action snippets. The temporal branch reserve temporal and motion features, while spatial-temporal is complementary and provide global information. This also demonstrates that fusion at an early stage can take full advantage of the complementary of temporal and spatial features. And even with only RGB frames, our basic network still outperforms other 3D convolutions and two-stream networks.

Fig. 9 demonstrates some qualitative examples for successful and failure cases. The top examples are successfully recognized examples. With the help of temporal guided network, our methods can successfully classify most of the classes with different backgrounds and qualities. The green frames refer to the snippets that our frameworks are more confident, which also illustrates that the proposed method can identify essential movements for recognition. The last two rows show some failure examples. The similar backgrounds and the unclear movements make the system misclassified these videos into Haircut and Surfing.

3) Few shot recognition: One of the most significant challenges in video action recognition is that training deep ConvNets usually require a large amount of labeled data. With limited training data, our proposed network can fully utilize video information and still performs well. Although our model is pretrained on Kinetics dataset, we have investigated the influence of overlap classes between Kinetics and UCF101 in Section V-A1. The observation shows that our framework can recognize new actions without the help of overlap classes.

To enable few shot learning, we randomly sample 10%, 20%, 33%, and 50% videos from the training set. We report the few shot learning results in Fig. 10 when different amounts of training videos are used. (a) demonstrates the effects with different blocks using DDN-S. DDN-S2 consistently performs well. (b) illustrates the different results of block fusion strategies. DDN-C is comparable with DDN-S2 with 10% and 20% training data. This shows that when facing limited training examples, concatenating is also a satisfactory strategy.

TABLE V
THE ACCURACY(%) AND THE AMOUNT OF PARAMETERS OF DIFFERENT DESIGN OF DENSE DILATED NETWORK ON HMDB51 DATASET.

Method	Accuracy	Params
DDN-S1	69.17	$0.60M$
DDN-S2	**73.05**	$0.60M$
DDN-S3	71.85	$0.60M$
DDN-A12	72.30	$0.35M$
DDN-C12	72.68	$0.35M$
DDN-A123	71.34	$0.60M$
DDN-C123	71.91	$0.63M$
DDN-S2-L	73.69	$1.02M$
DDN-S2-SG	71.79	$1.02M$
DDN-S2-TG	**74.51**	$1.02M$
Dense-S2	70.78	$0.60M$
Dilated-S2	70.07	$0.22M$

TABLE VI
COMPARISON WITH STATE-OF-THE-ART METHODS ON HMDB51.

Method	Accuracy
Two-stream fusion [44]	69.20
Two-stream MiCT-Net [45]	70.50
Temporal Segment Network [28]	69.40
Dilated TCN [38]	68.79
DDN-S2	73.05
DDN-S2-L	73.42
DDN-S2-SG	71.79
DDN-S2-TG	**74.51**

(c) shows the different fusion strategies of the two-stream network. The temporal guided network still outperforms all other methods. (d) reports the comparison of our methods with baseline methods which reported few shot learning results. EnergyNet using auxiliary web images to learn attention maps. It suggests that our DDN-S2-TG method outperform our baseline methods by a large margin.

It worth mention that our proposed frameworks can also achieve better performance using fewer videos compared with the baseline methods using all data. Using only 20% training videos, temporal guided network outperforms C3D and P3D which uses 100% training data on UCF101. While with 50% videos, temporal guided network outperforms all the baseline methods with whole training data on both datasets. One interesting finding is that temporal guided network gets more improvements with less training data. Using 10% training videos, temporal guided network outperforms spatial guided network by 3.96%, compared with 2.27% when using 50% training data. This finding demonstrates our proposed method can preserve more intra-class relationship than baseline methods giving little training data.

B. Experiments on HMDB51

To further validate our proposed method, we also perform evaluations on the HMDB51 dataset. Compared to UCF101, the HMDB51 has more complex movie background, which is more challenge for action recognition.

Table V reports the detail results of dense block settings and fusion strategies. Similar to UCF101, DDN-S2 performs better than DDN-S1 and DDN-S3, and DDN-S2-TG outperforms other fusion approaches. This consistently shows the effectiveness of our methods. In addition, we propose two

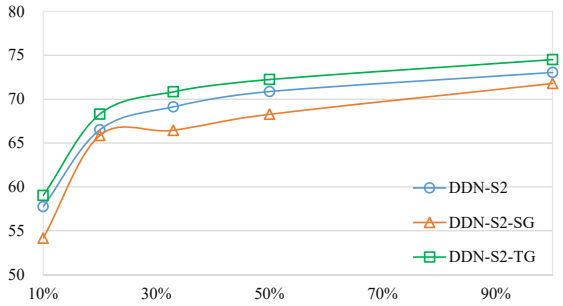


Fig. 11. Few shot video action recognition results on HMDB51.

baseline experiments using only "dense connection" or "dilated convolution". The results also prove that the combination of dense connnection and dilated convolution makes significant contribution to action recognition.

We report the performance compared with state-of-the-art methods in Table VI. The temporal guided methods still achieve the best performance compared to other fusion approaches, which is consistent with UCF101. When facing the limited training data, our proposed network also get promising results in Fig. 11. Using 20% training data, the result of DDN-S2-TG is only about 5% percent lower than using all data. We also report the accuracy of all the classes in HMDB51 in Table VII using 20% and 100% training data. Even with 20% training data, some of the classes can still achieve high accuracy. The comparison results can be viewed more clearly in Figure 12. Our method performs better than fuse TSN features in most classes, especially on long term actions such as Drink, Eat and Shoot_bow. This also demonstrates the ability to capture long term relations of our proposed method.

The qualitative examples on HMDB51 is shown in Fig. 13. We can observe that most of the successful examples share some common clues in actions, while some failure examples are difficult for recognition due to confusion actions and complex background.

VI. CONCLUSIONS

We introduce a novel dense dilated framework to perform video action recognition and few shot learning. The receptive fields include both local and global features via dense connections. We further explore the different strategies for layer aggregation among dense dilated blocks. The concatenating way is better than adding up all the blocks. The features of the second layer of blocks achieve best results under all conditions. We then study the fusing configuration of two stream network. The experiments show that temporal guided network not only preserves temporal branch information but also provides complementary features with the spatial branch. The results show the effectiveness of our proposed framework compared to baseline methods. Our proposed framework also achieves state-of-the-art performance when facing limited data. As future work, we will explore some related topics such as video segmentation and action detection based on the dense dilated framework.

TABLE VII
PRE-CLASS ACCURACY(%) ON HMDB51 (SPLIT 1). 20% AND 100% INDICATE THE RATE OF THE TRAINING DATA USED TO BUILD THE MODELS.

Class	20%	100%	Class	20%	100%	Class	20%	100%	Class	20%	100%
Brush_hair	70.00	76.67	Fencing	66.67	70.00	Pullup	100.00	100.00	Smile	40.00	53.33
Cartwheel	46.67	53.33	Flic_flac	100.00	100.00	Punch	86.67	90.00	Smoke	66.67	66.67
Catch	66.67	90.00	Golf	100.00	96.67	Push	96.67	96.67	Somersault	83.33	86.67
Chew	76.67	76.67	Handstand	93.33	93.33	Pushup	93.33	96.67	Stand	26.67	70.00
Clap	83.33	76.67	Hit	33.33	70.00	Ride_bike	100.00	96.67	Swing_baseball	83.33	80.00
Climb	90.00	96.67	Hug	83.33	90.00	Ride_horse	73.33	90.00	Sword_exercise	73.33	76.67
Climb_stairs	60.00	66.67	Jump	26.67	46.67	Run	53.33	70.00	Sword	33.33	33.33
Dive	56.67	66.67	Kick_ball	76.67	70.00	Shake_hands	83.33	86.67	Talk	56.67	70.00
Draw_sword	53.33	83.33	Kick	56.67	60.00	Shoot_ball	80.00	96.67	Throw	40.00	33.33
Dribble	86.67	90.00	Kiss	90.00	83.33	Shoot_bow	86.67	86.67	Turn	46.67	60.00
Drink	53.33	90.00	Laugh	83.33	66.67	Shoot_gun	63.33	63.33	Walk	26.67	40.00
Eat	63.33	70.00	Pick	43.33	46.67	Sit	56.67	63.33	Wave	23.33	26.67
Fall_floor	36.67	56.67	Pour	76.67	86.67	Situp	100.00	100.00	mean	68.30	74.51

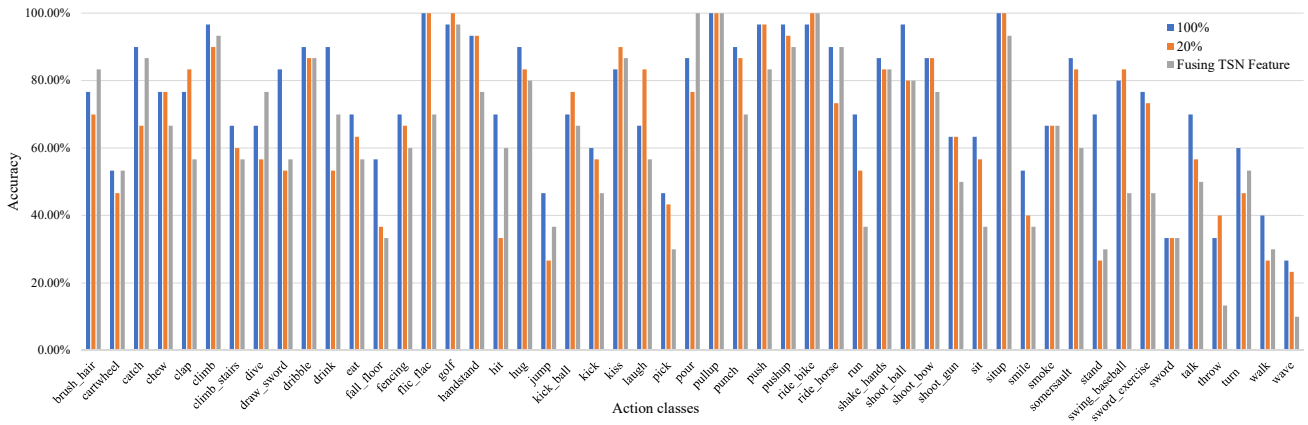


Fig. 12. Per-class accuracy of HMDB51 dataset using 100% traning examples, 20% traning examples with DDN-S2-TG, and only simply fuse the results of TSN features.

REFERENCES

[1] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Neural Information Processing Systems (NIPS)*, 2014, pp. 568–576.

[2] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[3] Y.-G. Jiang, Y. Wang, R. Feng, X. Xue, Y. Zheng, and H. Yang, "Understanding and predicting interestingness of videos," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2013.

[4] B. Xu, Y. Fu, Y.-G. Jiang, B. Li, and L. Sigal, "Video emotion recognition with transferred deep feature encodings," in *ACM International Conference on Multimedia Retrieval (ICMR)*, 2016, pp. 15–22.

[5] B. Su, J. Zhou, X. Ding, and Y. Wu, "Unsupervised hierarchical dynamic parsing and encoding for action recognition," *IEEE Transactions on Image Processing*, vol. 26, no. 12, pp. 5784–5799, 2017.

[6] W. Du, Y. Wang, and Y. Qiao, "Recurrent spatial-temporal attention network for action recognition in videos," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1347–1360, 2018.

[7] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[8] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[9] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 3169–3176.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Neural Information Processing Systems (NIPS)*, 2012.

[11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations (ICLR)*, 2015.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks." in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 2, 2017, p. 3.

[14] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv:1212.0402*, 2012.

[15] B. Xu, H. Ye, Y. Zheng, H. Wang, T. Luwang, and Y.-G. Jiang, "Dense dilated network for few shot action recognition," in *ACM International Conference on Multimedia Retrieval (ICMR)*, 2018, pp. 379–387.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2005, pp. 886–893.

[17] R. Hong, J. Tang, H.-K. Tan, C.-W. Ngo, S. Yan, and T.-S. Chua, "Beyond search: Event-driven summarization for web videos," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 7, no. 4, p. 35, 2011.

[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[19] H. Ye, Z. Wu, R.-W. Zhao, X. Wang, Y.-G. Jiang, and X. Xue, "Evaluating two-stream cnn for video classification," in *ACM International Conference on Multimedia Retrieval (ICMR)*, 2015, pp. 435–442.

[20] G. Gkioxari and J. Malik, "Finding action tubes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015, pp. 759–768.

[21] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *International Conference on Machine Learning (ICML)*, 2015, pp. 843–852.

[22] G. Chéron, I. Laptev, and C. Schmid, "P-cnn: Pose-based cnn features for action recognition," in *International Conference on Computer Vision (ICCV)*, 2015, pp. 3218–3226.

Fig. 13. Qualitative examples of successful and failure examples on HMDB51. The results are based on the DDN-S2-TG model with all the training data.

[23] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: a large video database for human motion recognition," in *International Conference on Computer Vision (ICCV)*, 2011, pp. 2556–2563.

[24] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 961–970.

[25] S. Zhao, Y. Liu, Y. Han, R. Hong, Q. Hu, and Q. Tian, "Pooling the convolutional layers in deep convnets for video action recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 8, pp. 1839–1849, 2018.

[26] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *International Conference on Computer Vision (ICCV)*, 2015, pp. 4489–4497.

[27] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *International Conference on Computer Vision (ICCV)*, 2017, pp. 5534–5542.

[28] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *European Conference on Computer Vision (ECCV)*, 2016, pp. 20–36.

[29] Y. Yang, I. Saleemi, and M. Shah, "Discovering motion primitives for unsupervised grouping and one-shot learning of human actions, gestures, and expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1635–1648, 2013.

[30] C. Orrite, M. Rodríguez, and M. Montañés, "One-sequence learning of human actions," in *International Workshop on Human Behavior Understanding*. Springer, 2011, pp. 40–51.

[31] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local svm approach," in *ICPR*, vol. 3, 2004, pp. 32–36.

[32] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *International Conference on Computer Vision (ICCV)*, vol. 2, 2005, pp. 1395–1402.

[33] M. Rodriguez, C. Orrite, C. Medrano, and D. Makris, "Fast simplex-hmm for one-shot learning activity recognition," in *CVPR Workshops*, 2017, pp. 41–48.

[34] C. Gan, C. Sun, L. Duan, and B. Gong, "Webly-supervised video recognition by mutually voting for relevant web images and web video frames," in *European Conference on Computer Vision (ECCV)*, 2016, pp. 849–866.

[35] J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli, "Attention transfer from web images for video recognition," in *ACM International Conference on Multimedia (MM)*, 2017.

[36] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning (ICML)*, 2015, pp. 448–456.

[37] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011, pp. 315–323.

[38] C. Lea, M. Flynn, R. Vidal, A. Reiter, and G. Hager, "Temporal convolutional networks for action segmentation and detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[39] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[40] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[41] Y. Zheng, H. Ye, L. Wang, and J. Pu, "Learning multiviewpoint context-aware representation for rgb-d scene classification," *IEEE Signal Processing Letters*, vol. 25, no. 1, pp. 30–34, 2018.

[42] A. Gupta and A. M. Rush, "Dilated convolutions for modeling long-distance genomic dependencies," *arXiv:1710.01278*, 2017.

[43] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," *arXiv:1705.06950*, 2017.

[44] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[45] Y. Zhou, X. Sun, Z.-J. Zha, and W. Zeng, "Mict: Mixed 3d/2d convolutional tube for human action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 449–458.