Attentional Learn-able Pooling for Human Activity Recognition

Bappaditya Debnath[1] and Mary O'brien[2] and Swagat Kumar[1] and Ardhendu Behera[1]

Abstract— **Human activity/behaviour monitoring and recognition is a key for facilitating humans robot interaction, and allows robots for a better scheduling of future operations. It is challenging and often addressed at different levels, such as human activity classification, future activity prediction and monitoring of the on-going activities. The paper proposes a novel attention-based learn-able pooling mechanism for human activity classification from RGB videos. Recently, most of the best performing human activity recognition approaches are based on 3D skeleton positions. The 3D skeleton positions are not always available in videos captured using RGB cameras, which are widely used in robotics applications. RGB videos contain rich spatio-temporal information and processing them semantically is a difficult task. Moreover, accurately capturing spatial information and long-term temporal dependencies is the key to achieving high recognition accuracy. We use an existing Convolutional Neural Network for image recognition to extract video features which are then processed using our innovative application of attention mechanism to focus the network on features that are more important for discrimination. Afterwards, we use a novel learn-able pooling mechanism to extract activity-aware spatio-temporal cues for efficient activity recognition. The proposed pooling mechanism learns the structural information from hidden states of a bidirectional Long Short-Term Memory network via Fisher Vectors.**

I. INTRODUCTION

There is a significant advancement in robotics research that drives the near future intelligent/smart robots, which will be capable of interacting and collaborating with humans to assist with various complex tasks in our day to day life. Automatic recognition of human activities is a fundamental part of the robot's perception of the situation at a given time and is necessary to facilitate natural interaction between humans and robots. Although this area has been actively researched by the computer vision community, it continues to be a very challenging. This is mainly due to factors like large variations in body pose and object appearance, a wide variety of backgrounds, lighting conditions, occlusion, complex body joint inter-dependencies, and inter- and intra-class variations. Thus, recent activity recognition models have focused on multiple modalities like vision-based 3D human body poses, RGB videos and depth maps. Due to the popularity of depth-based 3D pose estimation devices (e.g. Microsoft Kinect), authors have increasingly relied on pose-based methods [1], [2] for human activity recognition. This is especially true for the large scale datasets that are available now [3]. But, depth based pose-estimation devices often suffer from inherent inaccuracies [4] and requires both RGB and depth information resulting in processing of a high volume of data which is computationally expensive for resource-constrained computational platforms associated with robotics and automobile applications. Moreover, despite the popularity of depth-based pose estimation devices, in situations such as CCTV surveillance, home monitoring and so on, monocular RGB cameras are widely used. Thus, one needs to often rely on RGB videos for various real-world applications. Thus, in this paper, we explore human activity recognition based on RGB data only.

RGB videos present various visual, temporal and contextual cues involving a given human activity. Over the past few years, deep learning models based on CNNs have achieved very promising results [5], [6]. Existing approaches often combine the spatial information extracted using a CNN [7] and temporal dependencies by using recurrent networks such as LSTMs [8]. Approaches such as recurrent CNN has also been explored [9] in video recognition problems. Modern deep CNNs consist of a variety of layer types to capture hierarchical feature representation and the prediction is dominated by the task-specific representation of convolutional layers. These models have shown remarkable success in visual recognition by considering full images with distinctive classes. However, it raises questions about their performance in discriminating small changes in successive frames in a given video. Therefore, there is a need for learning meaningful spatio-temporal structures in videos for discriminating various human activities. In order to address this, we propose a novel learn-able pooling mechanism, which captures the activity-aware spatio-temporal structure in videos by exploring both spatial and temporal information in videos containing human activity. The spatial information is explored using the high-level features from a ImageNet [10] pre-trained CNN model (Inception-ResNet-V2 [11]). The dynamics of these spatial features over a given sequence and their importance for a given activity is captured using a bidirectional LSTM (bi-LSTM) and attention mechanism, which captures sequential attention, as well as spatial attention by focusing on various temporal and spatial locations in the sequence.

Our novel attention mechanism consists of two parts: 1) a sequential self-attention mechanism is used to selectively focus the high-level CNN representations on important temporal points. 2) The output of this sequential self-attention is fed into a bi-LSTM to capture the long-term temporal dependencies. We adapt the bi-LSTM to learn the structural information and similarities contained within its hidden states through Fisher Vectors (FVs). The FVs are

[1]Department of Computer Science, Edge Hill University, L394QP, Ormskirk, United Kingdom {debnathb,kumars,beheraa}@edgehill.ac.uk

[2]Department of Health Social Care and Medicine, Edge Hill University, L394QP Ormskirk, United Kingdom obrienm@edgehill.ac.uk

based on a clustering mechanism that semantically groups information. By exploiting the LSTM hidden states, with learn-able FVs the network is able to take advantage of this information. The output of learn-able FVs is pooled using AAP to represent the number of states equalling the number of activity classes. The novel learn-able FVs with AAP replaces the customary Global Average Pooling (GAP) and Fully-Connected (FC) layers used towards the end of many standard CNN architectures [12], [11]. Statistical pooling methods such as GAP or max pooling does not take into account the temporal and other structural information in recurrent mechanisms such as LSTM. To to pool the most relevant features based on learned representations authors have proposed learn-able pooling approaches [13], [14], [15]. This, inspired us to put forward the FV-based activity aware pooling mechanism that exploits the structural information and long-term temporal dependencies in semantic manner as opposed to simply taking the average or max-values for pooling. Our main contributions are:

- We introduce a novel learn-able FV with activity-aware pooling mechanism that learns structural information from hidden states of an LSTM to give us more effective temporal learning.
- Together with learn-able LSTM FV pooling, we introduce a sequential self-attention-based end-to-end trainable human activity recognition model that gives us state-of-the-art results on two challenging datasets.

II. Related Work

A. Activity Recognition - RGB Models

Human activity recognition models have traditionally relied on RGB video data [16]. Due to the recent advancement of deep learning, CNN models are widely used for learning representations from video data. Usually, there are two ways for processing video data through CNNs: i) First, frames in a video sequence are often encoded through 2D CNN in a time distributed manner [7], [17]. ii) The other method consists of deep 3D CNN models, which directly take input as RGB video [18], [9], [19]. To improve the performance of a given deep network, multiple streams are often combined together. In [7], the authors proposed a three streamed network, where the first stream processes the optical flow while the other two of the streams calculate regions of interest. Deng et al. [17] proposed a CNN-based two stream network for group activity recognition in which the first stream is focused on background scene, which enables the network to capture contextual cues. The second stream is used to recognize the multi-person activities. In [5], the authors use glimpse clouds with ResNet-50 architecture. Glimpse clouds can be interpreted as attention-based interest points. Molchanov et al. used recurrent 3D CNN [9] for online detection of hand gestures. In [6], authors integrate features from different parts of a spatial-temporal LSTM network to make an attention-based activity recognition model. In the proposed architecture, a bi-LSTM with sequential self-attention mechanism is used to capture long term temporal dependencies and meaningful spatial features without exploring multiple streams.

B. Activity Recognition - Attention Mechanisms

Attention mechanism was first proposed by Bahdanau et al. [20] to solve the machine translation problem by selectively focusing on more relevant and discriminatory features. It calculates similarity between queries, and keys and transforms values based on the similarity measure. Typically, keys and values are the same vectors. Zhang et al. [21] introduced self-attention mechanisms where the output is a weighted representation of itself. Recently, Multi-Head attention mechanism [22] has shown encouraging results in natural language processing. Multi-Head attention divides the spectrum into a number of sub-spaces. This allows the model to represent different learned sub-spaces at different positions. Attention mechanisms have also been widely used for video understanding tasks [23], [24], [6], [25]. Sharma et al. [6] introduced the recurrent attention mechanism for activity recognition. The approach tends to recognize important elements in video frames based on the performed action. Similarly, an end-to-end spatial and temporal attention model for human action recognition using skeleton data is proposed in [25]. The approach selectively focuses on discriminative skeleton joints within each frame and pays separate attention to the joints in different frames. The proposed attention mechanism is different from the above-mentioned approaches in the sense that we explore the self-attention and adapt it to sequential self-attention to capture the contextual information. We have also used bidirectional LSTM to capture long-term temporal dependencies. In our ablation study, we also experiment with Multi-Head attention and discuss its performance and complexity in comparison to the sequential self-attention.

C. Activity Recognition - Learn-able Poolings

Often, a pooling layer is very common towards the end of deep CNNs. There exist various pooling mechanisms in literature like Average or Max Pooling [26], [27], Attention Pooling [28], Rank-Pooling [29] and High-Dimensional Feature encoding [30]. The goal of pooling is to select the most important features and reduce the network size so that the model doesn't over-fit. But pooling using statistical methods or high dimensional encoding does not take into account the temporal and other structural information in recurrent mechanisms such as LSTM. Thus, authors have explored learn-able pooling methods to pool the most relevant features based on learned representations. Image features can be well described through descriptors such as Vector of Laterally Aggregated Descriptors (VLAD) [31] and FVs [32] which are an aggregation of unsupervised clustering information. VLAD uses K-means, while FVs use Gaussian mixture model (GMM) for clustering. In [14], authors introduced NetVLAD, where VLAD clusters are learnt in a supervised manner and used as input for learn-able pooling mechanism towards the end of the network. In a similar manner, Girdhar et al. [13] introduced Action VLAD, where VLAD features

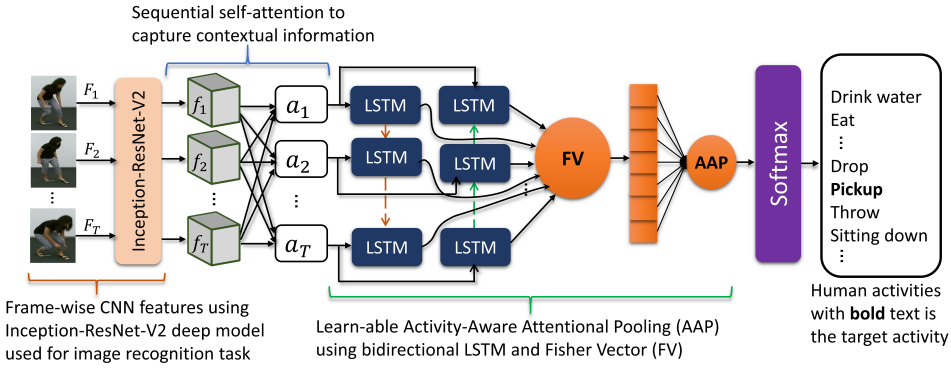

Fig. 1. The proposed deep network consists of: 1) a pre-trained CNN (Inception-ResNet-V2 [11]) model used to extract frame-wise high-level CNN features from a given video consisting of T frames. 2) a sequential self-attention layer to capture the contextual information consisting important spatial and temporal knowledge. 3) Learn-able activity-aware pooling consisting a bidirectional LSTM (bi-LSTM) and Fisher Vectors to learn the structural information and similarities by exploring the hidden states of the bi-LSTM. The AAP feature vector is passed through the Softmax layer to estimate the probabilities of various human activities.

are used as input for learn-able pooling for activity recognition. Instead, K-means, NetVLAD and Action VLAD learn semantic clusters in a supervise manner through a neural network. In [15], authors introduced learn-able FV (NetFV) to semantically cluster and pool audio and video features along with context gating for video classification. Unlike in original FVs, the cluster weights in NetFV are not calculated from GMM but using a differentiable soft-assignment. In the proposed approach, we adapted the NetFV by introducing LSTM FVs that relies on learned FVs from hidden states of a bi-LSTM and uses this representation for activity-aware learn-able pooling.

III. Proposed Approach

The proposed network is based on the widely used inception and residual network Inception ResNet-V2 [11], which is a high performance image classification and object detection model. We use the Inception ResNet-V2 to process the CNN features from each frame Fig.1. The model is used in a time distributed manner in which all the frames from a given video are passed through the same Inception ResNet-V2 model to process the corresponding CNN features. These features are then processed by our sequential self-attention to capture the contextual information consisting of important spatial and temporal knowledge. The sequential self-attention captures the information describing how much to recommend the CNN features at time point t in focus conditioned on all other CNN features from different time points. Afterwards, we introduce a novel bi-LSTM FV pooling that accurately captures the long-term dependencies and semantic structure present in temporal information. Our model is able to exploit the structural information contained in the bi-LSTM cells by semantically grouping its hidden states into learn-able clusters, which are part of the FV representation. This is followed by an AAP mechanism that allows the network to train without the need of an FC layer towards the end. We are able to show this mechanism successfully replaces the widely used customary GAP and FC mechanism and the whole model is trained in an end-to-end manner.

A. Temporal Processing to Capture Contextual Information

In order to capture the contextual information from the sequence of feature map f_t $(t = 1 \dots T)$ as outputs from the Inception ResNet-V2, we use sequential self-attention mechanism that transforms the feature map to a weighted version of itself with conditioned on rest of the feature maps representing rest of the frames. This leads the network to selectively focus on more relevant features to generate a holistic context information for further processing by our learn-able pooling for activity recognition. The goal of the attention mechanism is to assign higher weight-age to more relevant features. Normally, attention mechanism is described as a mapping function, which maps a query and a set of key-value pairs to an output context, where queries $\mathcal{Q}$, keys $\mathcal{K}$, values $\mathcal{V}$, and output context are all vectors. The context vector is deducted from $\mathcal{K}$and $\mathcal{Q}$ which effectively calculates the context compatibility between $\mathcal{Q}$ with $\mathcal{K}$. Thus, the output of the attention mechanism is a mapping of $\mathcal{V}$ weighted by the compatibility of $\mathcal{K}$ with $\mathcal{Q}$. The $\mathcal{Q}$, $\mathcal{K}$ and $\mathcal{V}$ vectors can either come from the same sources (e.g. self-attention) or from different sources (e.g. attention in neural machine translation). In our network, we use self-attention and thus, they come from the same source as a sequence of feature map f_t, where $t = 1 \dots T$. Formally, the attention mechanism can be described as in [22]:

$$Attention(\mathcal{Q}, \mathcal{K}, \mathcal{V}) = \text{softmax}(\mathcal{Q}\mathcal{K}^{T_r})\mathcal{V} \tag{1}$$

T_r represents the transpose of a given vector/matrix. The proposed sequential self-attention takes a query f_t and maps against a set of keys $f_{t'}$ associated with the candidate feature maps from frames at different time points and return values as context vector $\mathbf{v}_t$ computed by expanding Eqn 1:

$$\mathbf{v}_t = \sum_{t'=1}^{T} a_{t,t'} f_{t'} \text{ and } a_{t,t'} = \text{softmax}(W_a g_{t,t'} + b_a)$$
$$g_{t,t'} = \tanh(\mathcal{Q} + \mathcal{K} + b_g), \mathcal{Q} = \sigma(f_t W_g) \text{ and } \mathcal{K} = f_{t'} W_{g'} \tag{2}$$

The above equation shows the decomposition of Eqn 1 to compute the queries $\mathcal{Q}$, keys $\mathcal{K}$ and the values are nothing

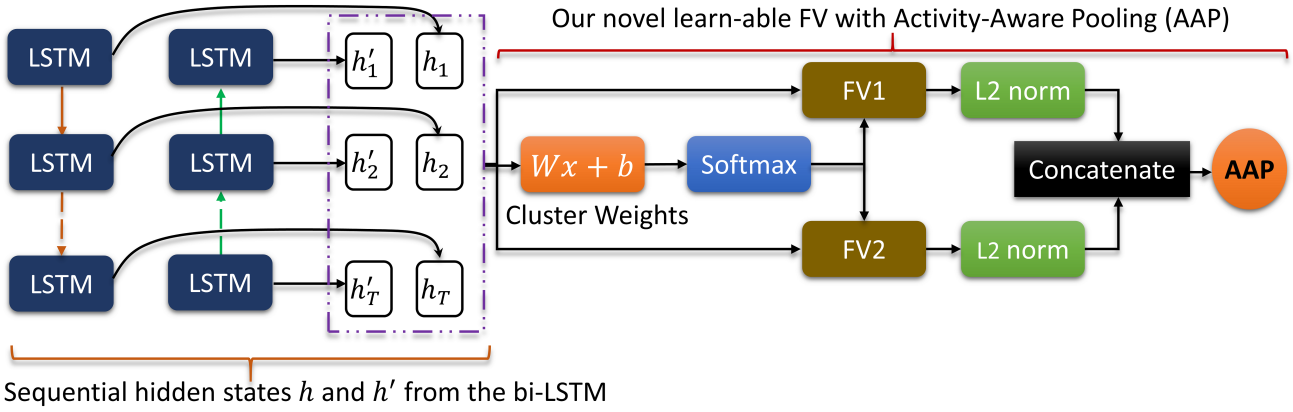

Fig. 2. The proposed learn-able Fisher Vector (FV) pooling using a bidirectional LSTM (bi-LSTM): The structural information in hidden states of the bi-LSTM is learned through FVs. For clarity, the bi-LSTM is unrolled to illustrate the hidden states over the video duration of T. The FV cluster weights are learned through parameters W and b. The weights are then used for deriving first order (FV_1) and second order (FV_2) FVs. The FV_1 and $FV2$ have learned parameters clusters' centers and co-variances as shown in Eq. 4. Towards the end FV_1 and FV_2 are concatenated and pooled with activity-aware weights for human activity classification.

but the output context vector $\mathbf{v}_t \in \mathcal{V}$. σ indicates sigmoid activation function. The weight matrices W_g and W'_g are for the respective feature maps f_t and $f_{t'}$; $W_a$is the weight matrix corresponding to their non-linear combinations. The element $a_{t,t'}$ is computed from $g_{t,t'}$ using the element-wise tanh function; b_a and b_g are the bias vectors. The attention-focused context vector v_t conveys *how much to attend the feature map* f_t in focus *conditioned on its neighbourhood context* representing feature maps of all other frames in a given video (see Fig. 1). The weight matrices W_g and W'_g, and the bias vectors b_a and b_g are learn-able parameters and learned during the training of the model. The output context vector $\mathbf{v}_t$ is now fed into the next stage of our architecture, which is learn-able FV pooling.

B. Learn-able Fisher Vector Pooling

The output of our sequential self-attention is a sequence of context vectors $\mathbf{v} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_T\}$ corresponding to the input frames of the given video $v = \{F_1, F_2, \ldots, F_T\}$. The contextual information captures the neighbourhood context by considering all other surrounding frames. However, it does not capture the sequential structure and dependencies. Our goal is to encode $\mathbf{v}$ using an internal state which summarizes information extracted from the history of past observations. The internal state encodes the sequence knowledge and is responsible for making a decision on how to act. The widely used approach to model this internal state is through hidden units $h_t \in \mathbb{R}^n$ of a recurrent neural network that are updated over time. We achieve this in our next step by using a fully-gated bidirectional LSTM (bi-LSTM). In Fig. 2, we present the unroll bi-LSTM for a better understanding of the temporal dependency, but in reality it is the same bi-LSTM. The bi-LSTM generates output as a sequence of hidden states in forward direction $h = \{h_1, h_2, \ldots, h_T\}$ and backward direction $h^{'} = \{h^{'}_1, h^{'}_2, \ldots, h^{'}_T\}$ corresponding to the input sequence of context vectors $\mathbf{v} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_T\}$. The hidden states in both forward and backward direction are concatenated $\mathbf{h} = [h, h^{'}]$ to produce the final contextual feature vector for further processing.

Generally, the sequence recognition using the LSTM is carried out by considering the last time step T based on the associated features at T and is based on the previously involved hidden states. This is a fundamental flaw in LSTM since it uses recurrent connections to maintain and communicate temporal information. Thus, researchers have recently explored dynamical temporal pooling [33] as an additional direct pathway for referencing previously seen frames. Inspired by this approach, our method focuses on the hidden states of the bi-LSTM and let the model *learns to attend* the different parts of the hidden states h and h' at each step of the output generation. We achieve this by using learn-able pooling with FVs in which the similar hidden states of the bi-LSTM are grouped together via clustering. The FVs [32] are computed as the aggregation of cluster weights, means and co-variances for each data vector. Instead of calculating the cluster-weights, means and co-variances as in original implementation [32], we use NetFV (FV integrated with neural network) to learns these parameters [15]. The main idea is to assign $\mathbf{h}_t$ to the cluster k as a soft assignment:

$$\alpha_k(\mathbf{h_t}) = \frac{e^{W_k^{T_r}\mathbf{h_t}+b_k}}{\Sigma_{j=1}^{K} e^{W_j^{T_r}\mathbf{h_t}+b_j}} \tag{3}$$

where matrix W_j and bias vector b_j are learn-able parameters. The soft assignment of $\alpha_k(\mathbf{h_t})$ of hidden state $\mathbf{h_t}$ to cluster k measures how close the hidden state $\mathbf{h_t}$ is to cluster k. Here $j \in (1, K)$ where K is the total number of clusters. Using the above soft assignment, we can compute the FVs are using the NetFV representation [15] as:

$$\begin{aligned} FV_1(j,k) &= \sum_{t=1}^{T} \alpha_k(\mathbf{h_t}) \left(\frac{\mathbf{h_t}(j) - c_k(j)}{\sigma_k(j)} \right) \\ FV_2(j,k) &= \sum_{t=1}^{T} \alpha_k(\mathbf{h_t}) \left(\left(\frac{\mathbf{h_t}(j) - c_k(j)}{\sigma_k(j)} \right)^2 - 1 \right) \end{aligned} \tag{4}$$

FVs FV_1 and FV_2 capture the respective first-order and second-order statistics, c_k and σ_k are the learn-able cluster's

center and diagonal co-variance of the k_{th} cluster, where $k \in [1, K]$. This is different from the original FV [32] in the sense that the cluster centers c_k and the co-variance matrices σ_k are not coupled to the cluster weights α_k. Moreover, c_k and σ_k are learned independently from the parameters of the soft assignment α_k as in Eqn. 4. Both FV_1 and FV_2 are normalized and then concatenated to get the final $FV = [FV_1, FV_2]$. Our implementation is different from the approach in [15] since we use the weighted pooling mechanism in an activity-aware manner and is defined as:

$$Pooling(FV) = \text{softmax}(W_p FV + b_p) \tag{5}$$

where matrix $W_p \in \mathbf{R}^{|FV| \times C}$ and bias vector b_p are learn-able parameters and C is number of human activity classes. Our adaptation of FV is different from NetFV in the following ways:

- NetFV doesn't take into account the temporal information contained in the video frames while we learn FV from temporal information contained in hidden LSTM states.
- NetFV uses a FC layer towards the end while we use first order AAP.

In NetFV, authors learn FV directly from CNN features and do not consider any temporal information. The proposed adaptation takes attention weighted CNN features processed through LSTM as input. This helps to exploit the temporal structure contained within the hidden LSTM states and cluster them in a semantic manner. To out knowledge this is the first article that attempts to exploit the hidden LSTM states in this way. In NetFV [15], the pooled size is is a tuneable hyper-parameter which necessitates further layers for classification. Instead we implement AAP where the pooling weight itself acts as the final classifier and thus the pooling output is equal to the number of classes. This obviates the need for a FC layer and thus helps in preventing over-fitting.

IV. Experiments, Results and Discussion

The model is evaluated on two challenging daily activity recognition datasets. The first one is the MSR Daily activity dataset [34]. which has 320 videos with 10 subjects preforming 16 different daily activities. For evaluation, we follow the standard protocol [34] in which 50% of the subjects (subjects 1 to 5) are used for training and the rest for evaluation. The evaluation protocol is challenging and indicates good generalisation since only half of the data is used for training. The second is the NTU RGBD dataset [3], which is one of the largest human activity recognition dataset. This dataset contains approximately 57K video samples of daily activities containing 60 daily activities preformed by 40 different subjects. We evaluate the model on the cross-subject protocol suggested by the authors which is more difficult than the other cross-view protocol [3]. The model is trained with a mini-batch size of 4 to fit with a GPU memory of 24 GB. To train the model, we use Linux PC (Ubuntu 16.04 LTS) with a Nvidia Quadro P-6000 GPU. The performance of the proposed model and state-of-the-art approaches using the MSR Activity dataset is presented in Table I. It is clear that the proposed approach (91.9%) outperforms the state-of-the-art approaches by a significant margin. For example, using only the RGB video, our approach is 1.9% higher than the approach in [5] (90%) which combines multi-modal information (pose and RGB). Using only RGB information, the accuracy (75.3%) in [5] is significantly inferior to our approach (91.9%). This suggests the benefit of our proposed attentional learn-able pooling for human activity recognition using only RGB information. In Table I, most of the state-of-the-art approaches are based on the body pose represented as a 3D skeleton. The performance of our approach is better than these approaches. This justifies that our approach can be easily applicable to video-based activity recognition without requiring additional information such as depth, which is essential for the computation of 3D skeletons.

TABLE I

Comparison of the proposed model with the state-of-the-art approaches on MSR 3D Daily activity dataset [34]

Methods	**Pose**	**RGB**	**Accuracy** (%)
Ensemble [34]	×	-	68.0
Efficient Pose [35]	×	-	73.1
Moving Pose [36]	×	-	73.8
Poselets [37]	×	-	74.5
MP [38]	×	-	79.4
PDA [8]	-	×	75.3
Actionlet [39]	×	-	88.8
PDA [8]	×	×	90.0
Ours	-	×	**91.9**

TABLE II

Performance of our model in comparison to the state-of-the-art approaches on NTU RGB+D dataset [3]. All the results are in cross subject settings which is more challenging than the cross view settings

Methods	**Pose**	**RGB**	**Accuracy** (%)
Part-aware LSTM [3]	×	-	62.9
C3D [40]	-	×	63.5
DSSCA-SSLM [38]	×	×	74.9
Synthesized CNN [41]	×	-	80.0
ST-GCN [42]	×	-	81.5
DPRL+GCNN [43]	×	-	83.5
PDA [8]	×	×	84.8
3Scale ResNet152 [44]	×	-	85.5
Glimpse Clouds [5]	-	×	86.6
Ours	-	×	**87.2**

Table II presents the performances of the proposed approach and state-of-the-art approaches using the NTU dataset [3]. Similar to the performance in MSR Activity dataset, the proposed approach (87.2%) outperforms the state-of-the-art approaches in which many of them use multi-modal information (RGB + Pose). Using RGB only, our approach is 0.6% better than the best performing approach (Glimpse Clouds [5]) and 23.7% better than the approach in [40]. It is also clear that the proposed approach is significantly better than the 3D skeleton-based approaches. This signifies the proposed attentional learn-able pooling mechanism plays a key role in discriminating human activities in videos.

TABLE III

COMPARISON OF BASE NETWORK ACCURACY ON THE MSR DATASET [34]. 'BASE ACC' IMPLIES THE PERFORMANCE OF THE CORE CNN-LSTM MODELS WITHOUT THE USE OF OUR PROPOSED SEQUENTIAL SELF-ATTENTION AND NOVEL LEARN-ABLE POOLING USING FV. THE ASSOCIATED PARAMETERS ARE PRESENTED AS THE NEAREST MILLIONS

Base CNN Network	Params	Base Acc	Proposed Acc
MobileNets [46]	∼4.2M	75.0%	79.4%
NasNet Mobile [45]	∼2.6M	79.0%	82.5%
Inception V3 [47]	∼23M	79.5%	84.0%
Inception ResNet-V2 [11]	∼54M	**86.9**%	**91.9**%

V. ABLATION STUDY

In this section, we perform three different experiments to justify the suitability of various components: 1) different state-of-the-art deep CNN models to extract CNN features for our network, 2) compare the performance of the proposed learn-able pooling with the traditional GAP and FC combination, and 3) the benefits of the proposed sequential self-attention in comparison to the multi-head attention. First, we analyse the performance using different base CNNs to extract frame-wise CNN features from videos. We use three state-of-the-art CNN models with different characteristics. The performance on MSR dataset [34] is shown in Table III. For base network, the last layer (i.e. classification) is comprised of a GAP layer followed by a FC layer with softmax activation. This is placed on top of the core CNN-LSTM network. The NasNet Mobile [45] outperforms the MobileNets [46]. It also consists of significantly less number of parameters (∼2.6M vs ∼4.2M) in comparison to the MobileNets. Among the three architectures, the Inception-ResNet-V2 [11] achieves the best accuracy and has the largest number of parameters (∼54M). The proposed algorithm also improves accuracy when Inception-V3 [47] as a backbone. Although the proposed model benefits from better backbone (Inception-ResNet-V2), it is able to improve results across 4 different backbones. In Table III, it is evident that the our novel sequential self-attention and lean-able FV pooling enhances the performance of the core CNN-LSTM network. It also demonstrates the applicability of the proposed method across a spectrum of CNNs ranging from lightweight to heavier models.

Second, we demonstrate the effectiveness of the proposed sequential self-attention in comparison with the multi-head attention mechanism [22]. The multi-head attention mechanism focuses on more important parts of the feature map in discriminating various activities. The results are shown in Table IV, using both the MSR Activity [34] and NTU-RGBD [3] datasets. The performance of both attention mechanisms significantly improves the recognition accuracy in comparison to the base accuracy. In case of multi-head attention mechanism [22], the keys $\mathcal{K}$, queries $\mathcal{Q}$ and values $\mathcal{V}$ vectors are transformed through a number of trainable weights. Each transformation produces a different mapping of the same input vectors, where each mappings are called heads and hence the name multi-head attention. The optimum number of heads for multi-head attention is 4 and is found experimentally. The performance of the proposed sequential self-attention is better than the multi-head attention. Moreover, the associated number of learn-able parameters with sequential self-attention (∼98K) is significantly less than the multi-head attention (∼9.4M). This justifies the benefit of the proposed sequential self-attention, which not only gives higher accuracy but also computationally more efficient.

TABLE IV

COMPARISON OF THE PERFORMANCE OF THE PROPOSED SEQUENTIAL SELF-ATTENTION (SSA) WITH THE MULTI-HEAD ATTENTION (MHA). THE CLASSIFICATION LAYER CONSISTS OF THE COMBINATION OF GAP AND FC

Dataset	Base Acc	MHA Params	SSA Params	MHA Acc	SSA Acc
MSR [34]	86.9%	∼9.4M	∼98K	90.6%	**91.3**%
NTU [3]	82.2%	∼9.4M	∼98K	86.3%	**86.6**%

TABLE V

IMPACT OF SEQUENTIAL SELF-ATTENTION AND OUR NOVEL FV POOLING. THE BASE NETWORK IS INCPETION-RESNET-V2 + LSTM + GAP/FC

Dataset	Base	SSA & GAP/FC	SSA & FV pooling
MSR [34]	86.9%	91.3%	**91.9**%
NTU [3]	82.2%	86.6%	**87.2**%

In the third, we study the impact of our novel learn-able activity-aware pooling (AAP) using FVs on model's recognition performance Table V. In this experiment, we compare the recognition accuracy of our model using the proposed AAP with the customary combination of the GAP and FC layer. It is evident that the recognition accuracy is significantly better when our learn-able pooling mechanism is used. This is because the proposed learn-able pooling learns semantic clusters to pool more effective hidden states of a bi-LSTM to represent high-level encoding of the spatio-temporal structure in videos and thus, achieves better performance. The number of clusters is a tune-able hyper-parameter, and we have experimentally found the optimal number of clusters to be 32 and 64 for the MSR Daily Activity [34] and NTU-RGBD dataset [3], respectively.

VI. CONCLUSION

We have proposed a simple yet effective approach to recognize human activities using only monocular RGB videos for robotics applications. The novel attentional learn-able pooling mechanism can be easily integrated to any of the existing deep CNN models used for image/object recognition. A sequential self-attention mechanism is used to capture the contextual information which conveys how much to attend a feature map in focus conditioned on its neighbourhood feature maps. We further present an alternative to the customary GAP/MAP and FC layer with a learn-able pooling mechanism in the form of lear-nable FVs. The FVs semantically cluster temporal structures and dependencies present in hidden LSTM states to further enhance the performance. The end-to-end trained model is evaluated using two challenging datasets and preforms better than state-of-the-art.

REFERENCES

[1] T. S. Kim and A. Reiter, "Interpretable 3d human action analysis with temporal convolutional networks," in *CVPR workshops*. IEEE, 2017, pp. 1623–1631.

[2] Y. Xu, J. Cheng, L. Wang, H. Xia, F. Liu, and D. Tao, "Ensemble one-dimensional convolution neural networks for skeleton-based action recognition," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 1044–1048, 2018.

[3] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," in *CVPR*, 2016, pp. 1010–1019.

[4] B. Galna, G. Barry, D. Jackson, D. Mhiripiri, P. Olivier, and L. Rochester, "Accuracy of the microsoft kinect sensor for measuring movement in people with parkinson's disease," *Gait & posture*, vol. 39, no. 4, pp. 1062–1068, 2014.

[5] F. Baradel, C. Wolf, J. Mille, and G. W. Taylor, "Glimpse clouds: Human activity recognition from unstructured feature points," in *CVPR*, 2018.

[6] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," *ICLR workshop*, 2016.

[7] M. Ma, H. Fan, and K. M. Kitani, "Going deeper into first-person activity recognition," in *Proc. of the IEEE CVPR*, 2016, pp. 1894–1903.

[8] F. Baradel, C. Wolf, and J. Mille, "Human activity recognition with pose-driven attention to rgb," in *BMVC*, 2018.

[9] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network," in *CVPR*, 2016.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, no. 3, 2015.

[11] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *AAAI*, 2017.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[13] R. Girdhar, D. Ramanan, A. Gupta, J. Sivic, and B. Russell, "Actionvlad: Learning spatio-temporal aggregation for action classification," in *CVPR*, 2017, pp. 971–980.

[14] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *CVPR*, 2016, pp. 5297–5307.

[15] A. Miech, I. Laptev, and J. Sivic, "Learnable pooling with context gating for video classification," *arXiv arXiv:1706.06905*, 2017.

[16] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *IVC*, vol. 60, pp. 4–21, 2017.

[17] Z. Deng, A. Vahdat, H. Hu, and G. Mori, "Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition," in *Proc. of the IEEE CVPR*, 2016, pp. 4772–4781.

[18] D. Wu, L. Pigou, P.-J. Kindermans, N. D.-H. Le, L. Shao, J. Dambre, and J.-M. Odobez, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE Trans. on PAMI*, vol. 38, no. 8, pp. 1583–1597, 2016.

[19] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Trans on PAMI*, vol. 35, no. 1, pp. 221–231, 2012.

[20] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv arXiv:1409.0473*, 2014.

[21] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," *arXiv arXiv:1805.08318*, 2018.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[23] K. Cho, A. Courville, and Y. Bengio, "Describing multimedia content using attention-based encoder-decoder networks," *IEEE Trans on Multimedia*, vol. 17, no. 11, pp. 1875–1886, 2015.

[24] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *ICML*, 2015, pp. 2048–2057.

[25] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *AAAI*, 2017.

[26] A. Habibian, T. Mensink, and C. G. Snoek, "Video2vec embeddings recognize events when examples are scarce," *PAMI*, vol. 39, no. 10, pp. 2089–2103, 2016.

[27] N. Hussein, E. Gavves, and A. W. Smeulders, "Unified embedding and metric learning for zero-exemplar event detection," in *CVPR*, 2017, pp. 1096–1105.

[28] R. Girdhar and D. Ramanan, "Attentional pooling for action recognition," in *NIPS*, 2017, pp. 34–45.

[29] B. Fernando, E. Gavves, J. Oramas, A. Ghodrati, and T. Tuytelaars, "Rank pooling for action recognition," *IEEE Trans. on PAMI*, vol. 39, no. 4, pp. 773–787, 2016.

[30] Z. Xu, Y. Yang, and A. G. Hauptmann, "A discriminative cnn video representation for event detection," in *CVPR*, 2015, pp. 1798–1807.

[31] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *CVPR*. IEEE, 2010, pp. 3304–3311.

[32] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *CVPR*. IEEE, 2007, pp. 1–8.

[33] S. Yeung, O. Russakovsky, N. Jin, M. Andriluka, G. Mori, and L. Fei-Fei, "Every moment counts: Dense detailed labeling of actions in complex videos," *IJCV*, vol. 126, no. 2-4, pp. 375–389, 2018.

[34] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *CVPR*. IEEE, 2012, pp. 1290–1297.

[35] A. Eweiwi, M. S. Cheema, C. Bauckhage, and J. Gall, "Efficient pose-based action recognition," in *ACCV*. Springer, 2014, pp. 428–443.

[36] M. Zanfir, M. Leordeanu, and C. Sminchisescu, "The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection," in *CVPR*, 2013, pp. 2752–2759.

[37] L. Tao and R. Vidal, "Moving poselets: A discriminative and interpretable skeletal motion representation for action recognition," in *CVPR*, 2015, pp. 61–69.

[38] A. Shahroudy, T.-T. Ng, Y. Gong, and G. Wang, "Deep multimodal feature analysis for action recognition in rgb+ d videos," *PAMI*, vol. 40, no. 5, pp. 1045–1058, 2017.

[39] J. Wang and Y. Wu, "Learning maximum margin temporal warping for action recognition," in *CVPR*, 2013, pp. 2688–2695.

[40] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *CVPR*, 2015, pp. 4489–4497.

[41] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognition*, vol. 68, pp. 346–362, 2017.

[42] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI*, 2018.

[43] Y. Tang, Y. Tian, J. Lu, P. Li, and J. Zhou, "Deep progressive reinforcement learning for skeleton-based action recognition," in *CVPR*, 2018, pp. 5323–5332.

[44] B. Li, Y. Dai, X. Cheng, H. Chen, Y. Lin, and M. He, "Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep cnn," in *ICMEW workshop*. IEEE, 2017, pp. 601–604.

[45] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *CVPR*, 2018.

[46] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv arXiv:1704.04861*, 2017.

[47] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *CVPR*, 2016, pp. 2818–2826.