DOI 10.12694/scpe.v20i3.1545
Scalable Computing: Practice and Experience ISSN 1895-1767
Volume 20, Number 3, pp. 485–494. http://www.scpe.org © 2019 SCPE

ENSEMBLE SPATIO-TEMPORAL DISTANCE NET FOR SKELETON BASED ACTION RECOGNITION

NAVEENKUMAR M. AND DOMNIC S.*

Abstract. With the recent developments in sensor technology and pose estimation algorithms, skeleton based action recognition has become popular. This paper proposes a deep learning framework for action recognition task using ensemble learning. We design two subnets to capture spatial and temporal dynamics of the entire video sequence, referred to as $Spatial - distance\ Net$ ($SdNet$) and $Temporal - distance\ Net$ ($TdNet$) respectively. More specifically, $SdNet$ is a Convolutional Neural Network based subnet to capture spatial dynamics of joints within a frame and $TdNet$ is a long short term memory based subnet to exploit temporal dynamics of joints between frames along the sequence. Finally, two subnets are fused as one ensemble network, referred to as $Spatio - Temporal\ distance\ Net$ ($STdNet$) to explore both spatial and temporal information. The efficacy of the proposed method is evaluated on two widely used datasets, UTD MHAD and NTU RGB+D, and the proposed $STdNet$ achieved 91.16% and 82.55% accuracies respectively.

Key words: Human action recognition, Skeleton maps, spatio-temporal distance net, CNN, LSTM

AMS subject classifications. 68M14, 92B20

1. Introduction.

Action recognition is a hot research topic in the field of computer vision. It has various practical applications such as video surveillance, human-computer interaction, elderly care monitoring, smart homes, etc. The earlier studies have been investigated the action recognition task using RGB sensors. When low cost 3D sensors are available in the market, action recognition using depth data has become popular. Depth data is invariant to illumination changes compared with RGB data. In the seminal work, Shatton et al. [24] presented an approach to get skeleton joints from depth data in real time. It has generated a renewed interest in the research community to use of skeleton data for action recognition. Moreover, Skeleton map is invariant to viewpoints or appearances compared with depth map, thus suffering less intra-class variance [36].

In the past decade, multiview learning based methods [26] have achieved state of the art performance in the field of computer vision. In multiview learning, different views (features) are obtained either from multiple sources or from a single source. The traditional methods have been focused on designing hand crafted features for action recognition task [1] [27]. Due to limited representation power of hand crafted features, they often fail on large datasets. Deep multiview based methods [19] [32] have received much attention in the recent years. Most of these methods use a single type of deep network for action recognition [33]. Unlike these methods, this paper proposes a ensemble network using multiple convolutional neural networks (CNN) and multiple long short term memory (LSTM) neural networks.

The contributions of this paper are three fold. First, We design two subnets to capture spatial and temporal dynamics of the entirety sequence, referred to as $Spatial - distance\ Net$ ($SdNet$) and $Temporal - distance\ Net$ ($TdNet$) respectively. Specifically, $SdNet$ is a CNN based network and $TdNet$ is a LSTM based network. Second, the two subnets are fused as one ensemble network ($STdNet$) for action recognition task. Last, the performance of the proposed method is investigated by conducting extensive experiments on two benchmark datasets and detailed analysis is reported. The rest of the paper is organized as follows. Section 2 gives a brief overview of related works in the literature and the proposed method is presented in Sec. 3. The experimental results are described in Sec. 4. The conclusions are drawn in the final section.

2. Related Work.

In this section, we briefly review the literature related to our work i.e. skeleton based approaches for action recognition. Existing literature about the skeleton based action recognition can be classified into two types: Handcrafted feature based methods (Traditional methods), Deep learning based methods.

2.1. Handcrafted feature based methods.

These methods can be classified into three categories: (i) joint based, (ii) mined joint based and (iii) dynamics based methods. The joint based methods capture the correlation between the joints for action recognition task. For example, Mller et al. [20] introduce a class of

*Department of Computer Applications, NIT, Tiruchirappalli, (mnaveenmtech@gmail.com, domnic@nitt.edu)

boolean features expressing geometric relations between body points of a pose. Kerola et al. [14] has employed a graph based approach for action recognition. In this work, an action sequence is represented as spatio-temporal graph and edge weights are calculated based on the pair wise distances. Hussein et al. [13] has used the covariance matrix of skeleton sequence as feature descriptor for action recognition and the multiple covariance matrices are generated to capture the relation between joint movement and time. Mined joint based methods try to learn which body parts are discriminative for action recognition. In paper [4], a genetic algorithm is proposed to identify informative joints for a specific class. Then, K-means algorithm is employed for action recognition task. In work [29], skeleton joints are grouped into five body parts. Then, data-mining techniques are applied to obtain distinctive poses in spatial domain and temporal domain. Support Vector Machine is employed for action classification. In dynamics based methods, the temporal dynamics are captured from the 3D trajectories of the skeleton action sequence for action classification task. Dynamic based approaches use linear dynamical systems(LDS) [2] or hidden Markov models (HMM) or mixed approaches [22] for modeling of actions. Handcrafted features are having limited representation capability and hence they often fail on large datasets.

2.2. Deep learning based methods. There are two types of deep learning models received much attention for action recognition task. They are (i) Recurrent Neural Networks (RNNs) and (ii) Convolutional Neural Networks (CNNs).

Different RNN based structures such as hierarchical RNN [7], spatio-temporal long short-term memory (LSTM) [17], part-aware LSTM [23], spatio-temporal attention based RNN [25] and two stream RNN [30] have been proposed to learn discriminative features from skeleton data for action recognition. The above methods concatenate the coordinates of joints at each time step before applying RNN based methods. Thus, spatial geometrical relations among different joints are lost in this pre-processing stage.

In contrast, CNNs directly extract information from texture images which are encoded from skeleton sequences. Different CNN based methods are proposed for skeleton based action recognition. In [6], the 3D (x,y,z) coordinates of joints in skeleton action sequence are mapped into red, blue and green values respectively. Hence, the skeleton action sequence is converted into the color image, and the action recognition problem is converted to image classification problem, and then the powerful image classifiers such as Convolution Neural Networks (CNN) are employed for action recognition. Due to the small size of the converted color images, it is difficult to fine-tune the existing CNN. In the work [34], the joint trajectories are extracted and encoded into color images by using hue, saturation, and values. The encoded trajectories are used in CNN as inputs for action classification. The joint trajectories capture temporal variations, but fail to extract structural dynamics within a frame in the action sequence. In this context, Li et al. [16] proposed four Joint Distance Maps based on the pairwise distances of skeleton joints within the frame and the CNN was adopted for action recognition. But this method fails to distinguish some actions, which are having similar distance variations i.e. drawcircleclockwise and drawcirclecounter clockwise, due to the loss of local temporal information. To address this issue, this paper proposes an ensemble network, which is formed using CNN and LSTM based networks (SdNet and TdNet), to capture spatial and temporal dynamics of joints along the sequence. It is note that the success of an action recognition task is dependent on how effectively a model captures the spatial and temporal dynamics from the action sequences to achieve higher recognition accuracy.

3. Proposed Method. In this section, we first introduce the some necessary backgrounds. Then, we present two phases, Feature Extraction and Action Representation, of the proposed method. Finally, the action classification phase is discussed.

3.1. Preliminaries. Recurrent Neural Networks (RNN) are designed to model sequential problems. RNN has it's internal memory and can store information about past computations. It allows RNN to exhibit dynamic temporal behaviour. In theory, they can handle arbitrary length sequences, but in practice, RNNs have trouble to model long term sequences due to vanishing/exploding gradients problem. To overcome this problem, Long Short Term Memory (LSTM) [11] is proposed. The basic structure of LSTM unit is shown in Fig. 3.1. From the Fig 3.1, X_t is the input to the LSTM at time step t. I_t, F_t, G_t and O_t are the internal structures of input, forget, cell candidate and output gates of LSTM. I_t, F_t, G_t and O_t are defined at time step t as stated in Equations 3.1 to 3.4 respectively. The cell state (C_t) and hidden state of LSTM (H_t) are updated as stated in

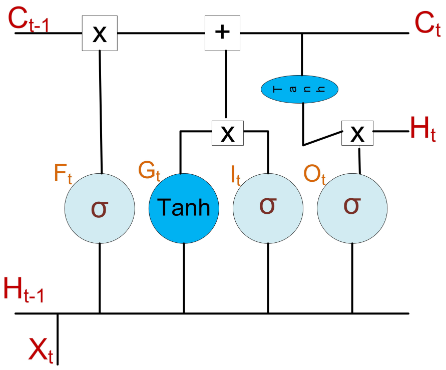

FIG. 3.1. *An LSTM unit.* O_t*,* F_t *and* I_t *are output, forget and input gates of LSTM. ’+’ and ’×’ are the element wise addition and multiplication respectively.*

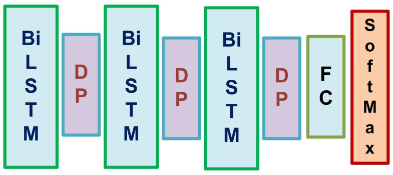

FIG. 3.2. *Block diagram of a BaseNet. BiLSTM, DP, and FC represent the bidirectional LSTM, dropout and fully connected layers.*

Equations 3.5 and 3.6 respectively.

$$I_t = \sigma(W_{IX}X_t + W_{IH}H_{t-1} + b_I) \tag{3.1}$$

$$F_t = \sigma(W_{FX}X_t + W_{FH}H_{t-1} + b_F) \tag{3.2}$$

$$O_t = \sigma(W_{OX}X_t + W_{OH}H_{t-1} + b_O) \tag{3.3}$$

$$G_t = Tanh(W_{GX}X_t + W_{GH}H_{t-1} + b_G) \tag{3.4}$$

$$C_t = F_tC_{t-1} + I_tG_t \tag{3.5}$$

$$H_t = O_tTanh(C_t) \tag{3.6}$$

where W and b represent the weight matrix and bias respectively.

3.2. BaseNet. The proposed $BaseNet$ is a multi-layer LSTM network. The backbone of our $BaseNet$ contains three bidirectional LSTM (Bi-LSTM) layers as shown in figure 3.2. The dropout (DP) is employed between two bi-LSTM layers to reduce the overfitting problem in training our $BaseNet$. Finally a fully connected layer (FC) with softmax activation function is used for action classification task.

3.3. Spatial-distance Net (SdNet). We design the $Spatial-distance\ Net$ ($SdNet$) to explore the spatial dynamics of joints of a entirety sequence. The proposed $SdNet$ contains four CNN as shown in Fig. 3.3. With the motivation of the work [16], $SdNet$ employs pair wise distances constructed in 3D space and three 2D

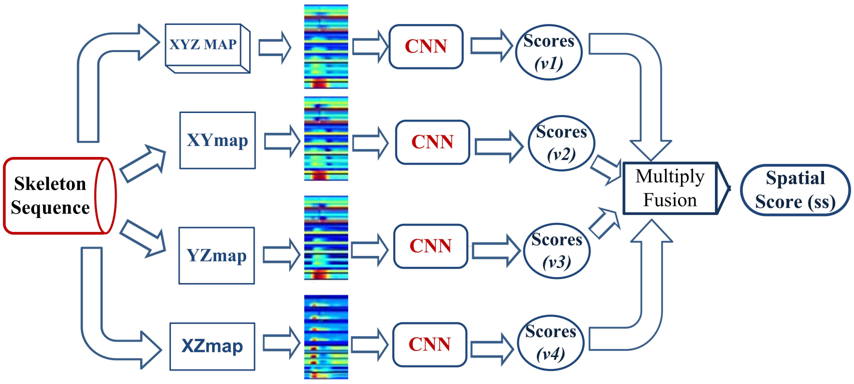

Fig. 3.3. *Spatial-distance Net (SdNet)*

orthogonal spaces as shown in Fig 3.3. Unlike [16], this paper proposes $TdNet$ to explore temporal dynamics between frames along the sequence as explained in Section 3.4

The four features in the spatial domain are referred to $XYZmap, XYmap, YZmap$ and $XZmap$. To deal with actions that contains human to human interaction, every action is assumed to be performed by two subjects (main subject and auxiliary subject). If an action sequence contains only one subject, a shadow subject is copied from main subject [5]. Suppose an action sequence A contains M skeleton frames and each skeleton frame contains $2N$ joints, where N joints are related to main subject and other N joints are for auxiliary subject. A $= \{F_1, ...F_M\}$, where F represents a frame and $F_i = \{J_1^i, ...J_{2N}^i\}$. The J_j^i represents the 3D coordinates (x, y, z) of j^{th} joint of i^{th} frame. The 2D orthogonal projections of 3D Joint J are referred as Jxy, Jyz and Jxz. The four spatial features are constructed as stated in Eqs. (3.7) to (3.10).

$$XYZmap = \{dist_{3D}(J_i^f, J_j^f)|i, j = 1.., 2N; i \neq j; f = 1.., M\} \tag{3.7}$$

$$XYmap = \{dist_{2D}(Jxy_i^f, Jxy_j^f)|i, j = 1.., 2N; i \neq j; f = 1.., M\} \tag{3.8}$$

$$YZmap = \{dist_{2D}(Jyz_i^f, Jyz_j^f)|i, j = 1.., 2N; i \neq j; f = 1.., M\} \tag{3.9}$$

$$XZmap = \{dist_{2D}(Jxz_i^f, Jxz_j^f)|i, j = 1.., 2N; i \neq j; f = 1.., M\} \tag{3.10}$$

where f represents the frame number, J refers (x,y,z) coordinates of joint, Jxy refers (x,y) coordinates of the joint and so on. $dist_{3D}()$ is the Euclidean distance of two points in Euclidean 3-space where as $dist_{2D}()$ is the Euclidean distance of two points in Euclidean 2-space. suppose, $r = (r_1, r_2, ..., r_n)$ and $s = (s_1, s_2...., s_n)$ are two points in Eucledean n-space, the $dist_{nD}()$ is calculated as:

$$dist_{nD}(r, s) = \sqrt{(s_1 - r_1)^2 + (s_2 - r_2)^2 + ...(s_n - r_n)^2} \tag{3.11}$$

For an action A, when the distances calculated for a single frame are arranged in a single column, four matrices are generated for four spatial features respectively. Each matrix of size $(2N^2 - N) \times M$. Since number of frames (M) is vary from sequence to sequence, feature matrices do not contain fixed number of columns for all the sequences in the training set. To avoid this problem and produce matrices with fixed number of columns M', bi-linear interpolation is used to resize the spatial feature matrix from $(2N^2 - N) \times M$ to $(2N^2 - N) \times M'$. Then, these feature matrices are encoded into gray images as stated in equation 3.12:

$$grayimage = \frac{FM - min(FM)}{max(FM) - min(FM)} * 255 \tag{3.12}$$

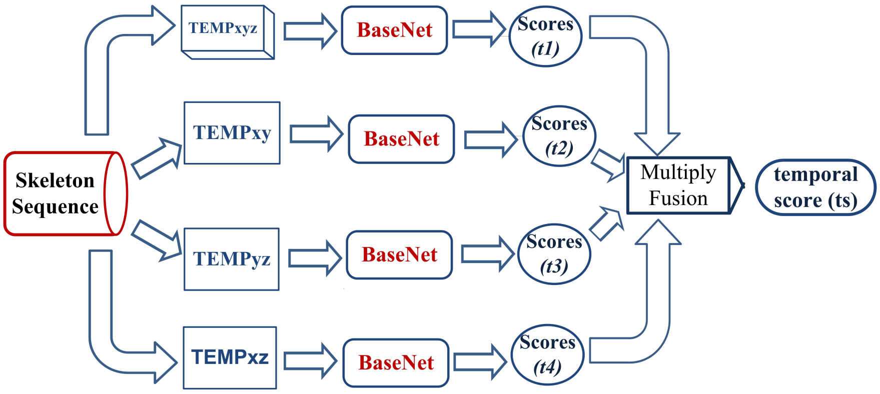

Fig. 3.4. *Temporal-distance Net (TdNet)*

where FM is a feature matrix. $min(FM)$ and $max(FM)$ are the minimum and maximum values of FM. The gray images are encoded into color texture images using Jet colorbar [16] to exploit spatial information using pretrained CNN models. This paper adopts multiplication fusion [16] to calculate the spatial score (ss) of $Sdnet$. Suppose $v1, v2, v3$, and $v4$ are the vectors related to scores of four CNNs, spatial score (ss) for action A is calculated as stated in Eq. (3.13):

$$spatial\ score\ (ss)\ for\ action\ A = (v1\ \diamond\ v2\ \diamond\ v3\ \diamond v4\ ) \tag{3.13}$$

where $\diamond$ represents the element wise multiplication.

3.4. Temporal-distance Net (TdNet). The proposed $Temporal - distance\ Net\ (TdNet)$ contains four $BaseNet$ as shown in Fig. 3.4. Four temporal features, referred to $TEMPxyz$, $TEMPxy$, $TEMPyz$ and $TEMPxz$ are constructed as stated in Eqs (3.14) to (3.17):

$$TEMPxyz = \{dist_{3D}(J_i^f, J_i^{f+1})|i = 1.., 2N; f = 1.., M-1\} \tag{3.14}$$

$$TEMPxy = \{dist_{2D}(Jxy_i^f, Jxy_i^{f+1})|i = 1.., 2N; f = 1.., M-1\} \tag{3.15}$$

$$TEMPyz = \{dist_{2D}(Jyz_i^f, Jyz_i^{f+1})|i = 1.., 2N; f = 1.., M-1\} \tag{3.16}$$

$$TEMPxz = \{dist_{2D}(Jxz_i^f, Jxz_i^{f+1})|i = 1.., 2N; f = 1.., M-1\} \tag{3.17}$$

In the context of the $TEMPmap$ feature, the distances calculated for two consecutive frames are arranged in a single column (i.e. each column as time step (X_t)). As a result, a matrix is formed of size $2N \times (M-1)$ for each $TEMPmap$ feature. Then, bi-linear interpolation is used to resize the $TEMPmap$ from $2N \times (M-1)$ to $2N \times M'$. The output of $TdNet$ for an action A is temporal score (ts). Suppose t1, t2, t3 and t4 are the score vectors of four $BaseNet$, (ts) is defined as stated in Eq. 3.18. It is note that the spatial score (ss) and temporal score (ts) vectors are in same size.

$$temporal\ score\ (ts)\ for\ action\ A = (t1\ \diamond\ t2\ \diamond\ t3\ \diamond t4\ ) \tag{3.18}$$

where $\diamond$ represents the element wise multiplication.

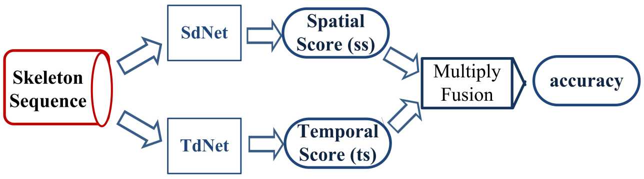

FIG. 3.5. *Spatio Temporal distance Net (STdNet)*

3.5. Spatio Temporal distance Net ($STdNet$). $STdNet$ is a ensemble network, consists of CNN ($SdNet$) and LSTM ($TdNet$) based subnets, to explore spatial and temporal dynamics. Specifically, the proposed two nets ($SdNet$ and $TdNet$) will be trained independently, with cross-entropy as the cost function. Suppose the training set contains K number of classes, the cross entropy is calculated as shown in Eq. 3.19:

$$cross\ entropy\ loss = -\sum_{i=1}^{K} y_i(log(p_i) \tag{3.19}$$

where y_i and p_i are true and predicted probabilities of class i. After training of $SdNet$ and $TdNet$, the ensemble of these two networks, referred to Spatio Temporal distance Net (STdNet), is constructed as shown in Fig. 3.5. The class label for an unknown test instance A is calculated as stated in Eq. 3.20:

$$Label\ of\ test\ instance\ A = Find\_Max\_index(ss\ \diamond\ ts) \tag{3.20}$$

where $\diamond$ represents the element wise multiplication and $Find\_Max\_index(.)$ is the function to find the index (class label) of maximum value.

4. Experiments. The efficacy of the proposed ensemble network ($STdNet$) is evaluated on two benchmark datasets for action recognition. The details are as follows.

4.1. Implementation details. To achieve better results, the popular CNN architecture "ResNet" [10] is fine-tuned for $SdNet$. The matlab implementation of resnet50 (pretrained) model is used for all the experiments. The initial learning rate is 0.001 and batch size is set to 32. Fifteen maximum training cycles are fixed for all the experiments. The network weights are learned using backpropagation with stochastic gradient descent with momentum value set to 0.9. For $BaseNet$, the base learning rate is 0.001 and the drop out rates are set to 0.3, 0.3 and 0.5 for three dropout layers respectively to prevent overfitting. All the experiments are carried out using NVIDIA Titan XP graphics card. The number of epochs are set to 200 and mini-batch size is 128 for all the experiments using $BaseNet$.

4.2. Datasets.

4.2.1. UTD MHAD dataset. UTD MHAD dataset [3] uses a Kinect camera and a wearable inertial sensor to capture depth, skeleton joints, RGB data and inertial sensor data. The skeleton is represented by using 20 joints. It contains 27 actions, performed by 8 subjects with each one performs an action four times. As a result, 864 (27 x8 x 4 = 864) data sequences are generated. After removing 3 corrupted sequences, the total data sequences are 861. For a fair comparison, we follow the cross subject protocol proposed in the paper [3]. For cross subject evaluation, the odd subjects are used for training and even subjects are used for testing. As a result, there are 431 action sequences in the training set and 430 in the testing set. Since this data set contains less number of instances, it is not suitable for training a deep learning model. Hence, we use a transfer learning approach for this dataset. Specifically, the pre-trained models on NTU RGB+D dataset are used for transfer learning.

4.2.2. NTU RGB+D dataset. NTU RGB+D dataset [23] is the largest action recognition dataset and it uses three kinect v2 sensors to capture the depth and skeleton information. The skeleton is represented using 25 joints. There are 56,880 action sequences and more than 4 million frames in this dataset. After removing

Table 4.1
Recognition accuracy of SdNet, TdNet and STdNet on UTD MHAD and NTU RGB+D datasets

Feature	UTD MHAD	NTU RGB+D	
	CS (%)	CS (%)	CV (%)
XYZmap	80.93	75.53	83.23
XYmap	78.14	71.72	74.42
YZmap	76.05	71.46	71.30
XZmap	73.72	68.34	75.93
TEMPxyz	65.58	58.77	64.72
TEMPxy	64.19	54.99	64.06
TEMPyz	68.37	55.38	58.96
TEMPxz	61.40	57.40	61.46
SdNet	87.67	80.61	86.73
TdNet	74.19	66.39	73.06
STdNet	91.16	82.55	88.46

Table 4.2
Comparison results on UTD MHAD Dataset

	Accuracy(%)
ElC-KSVD, 2014 [37]	76.19
Kinect and Inertial, 2015 [3]	79.10
Joint trajectory maps, 2016 [34]	85.81
Joint Distance Maps, 2017 [16]	88.10
Our method (STdNet)	**91.16**

missing skeletons, the dataset contains 56,578 action sequences. This dataset is challenging in two aspects: (i) large intra class variations; (ii) view point variations. Due to large scale of this dataset, it is highly suitable for deep learning. We follow the two experimental protocols, namely cross subject and cross view protocols, proposed in paper [23]. In cross subject test, the actions pertaining to the subjects 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35, 38 are considered for training and rest are for testing. As a result, the training set contains 40,091 samples, whereas testing set consisting of 16,487 action sequences. In cross view evaluation, the samples captured using camera 2 and 3 for training and camera 1 samples for testing.

4.3. Results of Action Recognition. Table 4.1 reports the results of proposed $SdNet$, $TdNet$ and $STdNet$. When comparing the individual spatial features, $XYZmap$ outperforms the other features on both the datasets. $STdNet$ is the result of ensemble of two networks $SdNet$and $TdNet$. $STdNet$ significantly achieves good performance than other recent works in the literature. From these results, it is concluded that both $SdNet$ and $TdNet$ have their significance to achieve recognition accuracy. Table 4.2 reports the performance of the proposed method with the state of the art methods on UTD MHAD dataset. It is noted from Table 4.2 that the works [34] [16] achieved 85.81% and 88.10% recognition accuracies respectively, which is not better than that of the proposed method, which achieves the accuracy of 91.16%. The reason is that the proposed method uses both spatial and temporal dynamics whereas the works [34] [16] used either spatial or temporal dynamics for action recognition.

Table 4.3 reports the results on the NTU RGB+D dataset. For this dataset, we compare our results with traditional methods [28] [8] [12], RNN based methods [7] [23] [25] [18] [30] [31] and CNN based methods [34] [16] [21]. The empirical results show that our $STdNet$ achieves 82.55% and 88.46% accuracies for cross subject and cross view settings respectively, which are higher than the recent existing works. Figure 4.1 depicts the individual recognition accuracies of all action classes. Among 60 classes, 36 action classes have achieved >=90% recognition rate in the cross view experimental setting whereas 24 action classes in the cross subject test. Table 4.4 shows the different action classes pertaining to specific recognition range on NTU RGB+D dataset.

Table 4.3
Comparison results on NTU RGB+D Dataset

	cross-subject(%)	cross-view (%)
Lie Group, 2014 [28]	50.10	52.80
Skeletal Quads [8]	38.60	41.40
LieNet, 2017 [12]	61.30	67.00
HBRNN, 2015 [7]	59.10	64.00
Part-aware LSTM, 2016 [23]	62.90	70.30
ST-LSTM + Trust Gate, 2016 [17]	69.20	77.70
JTM, 2016 [34]	73.40	75.20
Ensemble LSTM, 2017 [15]	74.60	81.25
STA-LSTM,2017 [25]	73.40	81.20
GCA-LSTM, 2017 [18]	74.40	82.80
Two-stream RNN, 2017 [30]	71.30	79.50
JDM, 2017 [16]	76.20	82.30
CNN+LSTM, 2018 [21]	67.50	76.21
Multi-task RNN, 2018 [35]	-	82.60
Multiview Re-Observation Fusion, 2018 [9]	73.80	85.90
Beyond Joints, 2018 [31]	79.50	87.60
Our method (STdNet)	**82.55**	**88.03**

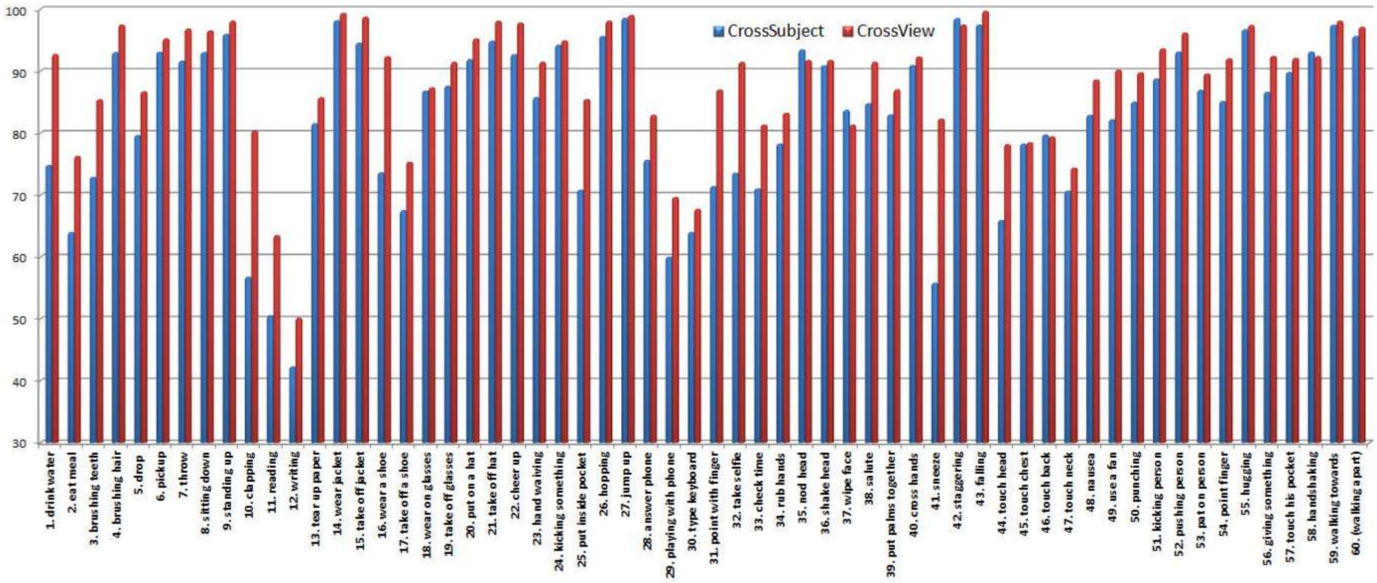

Fig. 4.1. *Recognition accuracies of each action class of NTU RGB+D dataset. There are 60 action classes in this dataset.*

5. Conclusion. This paper proposed an ensemble network consists of convolutional neural networks (CNN) and long short term memory (LSTM) neural networks. A CNN based subnet (*SdNet*) is designed to capture spatial information, where as LSTM based subnet (*TdNet*) exploits temporal dynamics along the video sequence. With the motivation of ensemble learning, these two subnets are fused as one ensemble network (*STdNet*). The efficacy of the proposed method is evaluated on two widely used datasets: UTD MHAD and NTU RGB+D datasets. Our *STdNet* achieved competitive results with the recent works for action recognition task.

Table 4.4
Action classes pertaining to specific recognition range on NTU RGB+D dataset

Recognition range	cross-subject	cross-view
>=95%	**11** classes (9, 14, 15, 21, 26, 27, 42, 43, 55, 59, 60)	**19** classes (4, 6, 7, 8, 9, 14, 15, 20, 21, 22, 24, 26, 27, 42, 43, 52, 55, 59, 60)
90% - 94%	**13** classes (4, 6, 7, 8, 20, 22, 24, 35, 36, 40, 52, 57, 58)	**17** classes (1, 16, 19, 23, 32, 35, 36, 38, 40, 49, 50, 51, 53, 54, 56, 57, 58)
85% - 89%	**9** classes (18, 19, 23, 38, 50, 51, 53, 54, 56)	**8** classes (3, 5, 13, 18, 25, 31, 39, 48)
80% - 84%	**7** classes (5, 13, 37, 39, 46, 48, 49)	**6** classes (10, 28, 33, 34, 37, 41)
75% - 79%	**4** classes (1, 28, 34, 45)	**5** classes (2, 17, 44, 45, 46)
70% - 74%	**7** classes (3, 16, 25, 31, 32, 33, 47)	- **2** classes (47, 29)
65% - 69%	**2** classes (17, 44)	**2** classes (11, 30)
60% - 64%	**3** classes (2, 29, 30)	-
<=59%	**4** classes (10, 11, 12, 41)	**1** classes(12)

6. Acknowledgments. The authors would like to acknowledge the NVIDIA Corporation for the donation of Titan XP GPU under NVIDIA GPU Grant Program.

REFERENCES

[1] Z. Cai, L. Wang, X. Peng, and Y. Qiao, *Multi-view super vector for action recognition*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2014, pp. 596–603.

[2] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, and R. Vidal, *Bio-inspired dynamic 3d discriminative skeletal features for human action recognition*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2013, pp. 471–478.

[3] C. Chen, R. Jafari, and N. Kehtarnavaz, *Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor*, in Image Processing (ICIP), 2015 IEEE International Conference on, IEEE, 2015, pp. 168–172.

[4] P. Climent-Pérez, A. A. Chaaraoui, J. R. Padilla-López, and F. Flórez-Revuelta, *Optimal joint selection for skeletal data from rgb-d devices using a genetic algorithm*, in Mexican International Conference on Artificial Intelligence, Springer, 2012, pp. 163–174.

[5] Z. Ding, P. Wang, P. O. Ogunbona, and W. Li, *Investigation of different skeleton features for cnn-based 3d action recognition*, in Multimedia & Expo Workshops (ICMEW), 2017 IEEE International Conference on, IEEE, 2017, pp. 617–622.

[6] Y. Du, Y. Fu, and L. Wang, *Skeleton based action recognition with convolutional neural network*, in Pattern Recognition (ACPR), 2015 3rd IAPR Asian Conference on, IEEE, 2015, pp. 579–583.

[7] Y. Du, W. Wang, and L. Wang, *Hierarchical recurrent neural network for skeleton based action recognition*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1110–1118.

[8] G. Evangelidis, G. Singh, and R. Horaud, *Skeletal quads: Human action recognition using joint quadruples*, in Pattern Recognition (ICPR), 2014 22nd International Conference on, IEEE, 2014, pp. 4513–4518.

[9] Z. Fan, X. Zhao, T. Lin, and H. Su, *Attention-based multiview re-observation fusion network for skeletal action recognition*, IEEE Transactions on Multimedia, 21 (2018), pp. 363–374.

[10] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[11] S. Hochreiter and J. Schmidhuber, *Long short-term memory*, Neural computation, 9 (1997), pp. 1735–1780.

[12] Z. Huang, C. Wan, T. Probst, and L. Van Gool, *Deep learning on lie groups for skeleton-based action recognition*, in Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE computer Society, 2017, pp. 1243–1252.

[13] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, *Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations.*, in IJCAI, vol. 13, 2013, pp. 2466–2472.

[14] T. Kerola, N. Inoue, and K. Shinoda, *Spectral graph skeletons for 3d action recognition*, in Asian Conference on Computer Vision, Springer, 2014, pp. 417–432.

[15] I. Lee, D. Kim, S. Kang, and S. Lee, *Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks*, in 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, 2017, pp. 1012–1020.

[16] C. Li, Y. Hou, P. Wang, and W. Li, *Joint distance maps based action recognition with convolutional neural networks*, IEEE Signal Processing Letters, 24 (2017), pp. 624–628.

[17] J. Liu, A. Shahroudy, D. Xu, and G. Wang, *Spatio-temporal lstm with trust gates for 3d human action recognition*, in

European Conference on Computer Vision, Springer, 2016, pp. 816–833.

[18] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, *Global context-aware attention lstm networks for 3d action recognition*, in The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 7, 2017, p. 43.

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

[20] M. Müller, T. Röder, and M. Clausen, *Efficient content-based retrieval of motion capture data*, in ACM Transactions on Graphics (ToG), vol. 24, ACM, 2005, pp. 677–685.

[21] J. C. Núñez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Vélez, *Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition*, Pattern Recognition, 76 (2018), pp. 80–94.

[22] L. L. Presti, M. La Cascia, S. Sclaroff, and O. Camps, *Gesture modeling by hanklet-based hidden markov model*, in Asian Conference on Computer Vision, Springer, 2014, pp. 529–546.

[23] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, *Ntu rgb+ d: A large scale dataset for 3d human activity analysis*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1010–1019.

[24] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, *Real-time human pose recognition in parts from single depth images*, in Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, Ieee, 2011, pp. 1297–1304.

[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, vol. 1, 2017, pp. 4263–4270.

[26] S. Sun, *A survey of multi-view machine learning*, Neural Computing and Applications, 23 (2013), pp. 2031–2038.

[27] A. Taalimi, A. Rahimpour, C. Capdevila, Z. Zhang, and H. Qi, *Robust coupling in space of sparse codes for multi-view recognition*, in 2016 IEEE International Conference on Image Processing (ICIP), IEEE, 2016, pp. 3897–3901.

[28] R. Vemulapalli, F. Arrate, and R. Chellappa, *Human action recognition by representing 3d skeletons as points in a lie group*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 588–595.

[29] C. Wang, Y. Wang, and A. L. Yuille, *An approach to pose-based action recognition*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 915–922.

[30] H. Wang and L. Wang, *Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks*, in e Conference on Computer Vision and Pa ern Recognition (CVPR), 2017.

[31] H. Wang and L. Wang, *Beyond joints: Learning representations from primitive geometries for skeleton-based action recognition and detection*, IEEE Transactions on Image Processing, 27 (2018), pp. 4382–4394.

[32] P. Wang, W. Li, Z. Gao, Y. Zhang, C. Tang, and P. Ogunbona, *Scene flow to action map: A new representation for rgb-d based action recognition with convolutional neural networks*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 595–604.

[33] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera, *Rgb-d-based human motion recognition with deep learning: A survey*, Computer Vision and Image Understanding, 171 (2018), pp. 118–139.

[34] P. Wang, Z. Li, Y. Hou, and W. Li, *Action recognition based on joint trajectory maps using convolutional neural networks*, in Proceedings of the 2016 ACM on Multimedia Conference, ACM, 2016, pp. 102–106.

[35] H. Wangg and L. Wang, *Learning content and style: Joint action recognition and person identification from human skeletons*, Pattern Recognition, 81 (2018), pp. 23–35.

[36] S. Zhang, X. Liu, and J. Xiao, *On geometric features for skeleton-based action recognition using multilayer lstm networks*, in 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2017, pp. 148–157.

[37] L. Zhou, W. Li, Y. Zhang, P. Ogunbona, D. T. Nguyen, and H. Zhang, *Discriminative key pose extraction using extended lc-ksvd for action recognition*, in Digital lmage Computing: Techniques and Applications (DlCTA), 2014 International Conference on, IEEE, 2014, pp. 1–8.

Edited by: Shajulin Benedict
Received: March 31, 2019
Accepted: August 30, 2019