Joint Selection using Deep Reinforcement Learning for Skeleton-based Activity Recognition

Bahareh Nikpour[1] and Narges Armanfard[2]

***Abstract*— Skeleton based human activity recognition has attracted lots of attention due to its wide range of applications. Skeleton data includes two or three dimensional coordinates of body joints. All of the body joints are not effective in recognizing different activities, so finding key joints within a video and across different activities has a significant role in improving the performance. In this paper we propose a novel framework that performs joint selection in skeleton video frames for the purpose of human activity recognition. To this end, we formulate the joint selection problem as a Markov Decision Process (MDP) where we employ deep reinforcement learning to find the most informative joints per frame. The proposed joint selection method is a general framework that can be employed to improve human activity classification methods. Experimental results on two benchmark activity recognition data sets using three different classifiers demonstrate effectiveness of the proposed joint selection method.**

***Index Terms*— Joint selection, activity recognition, skeleton data, deep reinforcement learning.**

I. INTRODUCTION

Activity recognition is a challenging, yet very useful task in the field of computer vision. Its applications range from monitoring of indoor and outdoor activities to human-robot interaction [1], [2]. With the prevalence of depth cameras such as Microsoft Kinect, and improvement of human pose estimation methods, skeleton data is easily accessible; therefore, skeleton-based activity recognition has become very popular [3], [4]. Skeleton data, which contains two-dimensional (2D) or three-dimensional (3D) coordinates of human body, is beneficial compared to RGB data since it is robust to variation of environment light, background clutter, view points and body scale.

For capturing skeleton data, often the key body joints are considered; however, for different activities, all the joints are not equally important. Consider two activities kick and throw as examples. For the activity kick, the lower body joints are important while in activity throw, upper body joints play more role. Beside that, in one single activity, the key joints may be different in different temporal frames.

In this paper we propose a novel framework for selecting the key informative joints in video frames for the purpose of human activity recognition. The process of selecting key joints can also be considered as a hard spatial attention learning mechanism to generate frame descriptions for activity classification. The proposed framework, for the first time, formulates the joint selection problem as a Markov Decision Process (MDP) [5] and employs deep reinforcement learning (DRL) to find the optimal solution. Throughout this paper, we refer to the proposed DRL-based joint selection method as JSDRL. In JSDRL, each video frame is associated with its own distinct optimal joint set, which may vary both in membership and size across the video. This allows the joint set to optimally adapt to temporal variations. JSDRL is a general framework that can be employed to improve the recognition performance of human activity classification methods (e.g., decouple GCN-DropGraph (DCGCN) [6], convolutional neural network (CNN) and long-short-term-memory (LSTM) based classifiers) as it only passes the relevant, informative joints to the classifiers. JSDRL reduces the computational complexity when training a classifier as it drops the irrelevant joints. In Reinforcement learning (RL), an agent learns the best policy by interacting with the environment and getting reward or punishment. RL is an effective search tool when the proper searching steps are unknown. In the joint selection scenario, the ground-truth for the key joints is not available, i.e. there is no supervision informing which joints are important. Therefore, it is unclear how to effectively explore spatial information over frames to choose which joints to use. As such, RL is a highly beneficial tool for joint selection.

The rest of the paper is organised as follow: In Section II, the related works to our method are reviewed. Section III explains the proposed method in detail. In Section IV, the experiments we have done are presented. The conclusion is drawn in Section V.

II. RELATED WORKS

A. Activity recognition with skeleton data

There have been a lot of researches for activity recognition in skeleton data some of which focus on extracting hand-crafted features [19], [9], [10], [34], [35], [36], [8], [30]. In [8], a three dimensional relationship between body parts is modeled by translations and rotations, and then the classification is performed in Lie algebra using the obtained representation. Weng el al. partitioned the action sequences into temporal windows and used them as the video descriptors. Then employing these descriptors and an extended version of Naive Bayes Nearest Neighbor algorithm, they performed activity recognition [30].

[1] B. Nikpour is with the Department of Electrical and Computer Engineering, McGill University, Montreal, Quebec, Canada (`Email: bahareh.nikpour@mail.mcgill.ca`).

[2] N. Armanfard is with the Department of Electrical and Computer Engineering, McGill University, Mila-Quebec AI Institute, Montreal, Quebec, Canada (`Email: narges.armanfard@mcgill.ca`).

Great performance of deeplearning-based techniques in image understanding encouraged researchers to employ deeplearning for activity recognition. Such algorithms can be categorized into methods based on Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), and graph-based networks.

Effectiveness of RNN in modeling sequential data has made it a good choice for video classification [13], [11], [20]. A two-stream RNN-based model is presented in [11] where both temporal dynamics and spatial information are captured by the two-stream network. Shahroudy et al. presented a part-aware LSTM method where each part is considered separately [20]. In [40], body joints are first grouped into five parts, and then each body part is fed into an individual sub-network; output of networks are then fused hierarchically to create one single output at the end, which is then used for recognition. Liu et al. extended LSTM to both spatial and temporal domains and proposed trust gate for dealing with noise [31]. An LSTM-based method is presented in [12] that finds soft spatial and temporal attention in skeleton data.

In CNN-based models, to fulfill the need for image data, the 3D coordinates of joints are usually considered as psedu-image [13]. Li et al. combined the position and velocity information of joints and used a two-stream CNN architecture for activity recognition [14]. In [38], first a transformation is applied, then the transformed data are fed to a CNN for robust feature extractions. Other CNN-based activity recognition methods for skeleton data can be found in [15] and [16].

Human body can be modeled as graph with nodes and vertices intrinsically; therefore, several graph-based methods have been proposed and gained successful results in the skeleton-based activity recognition field. Spatial-temporal graph convolutional network is proposed in [17], which consists of several spatial-temporal graph convolutions to extract body skeleton features. Inspired by [17], other graph-based methods are presented, such as [6], [7] and [18]. Yan et al. suggested a graph CNN which learns both spatial and temporal representations to improve the recognition performance and generalization ability in recognition [37].

All the deeplearning-based methods discussed above focus on developing network(s) that capture skeleton data features in an efficient way in order to realize an accurate activity recognition. None of these methods focus on finding the most informative joints among the set of given joints and discarding the irrelevant ones prior to recognition. This paper presents a novel technique for identification of informative joints across frame/video and employs them for recognition.

B. Reinforcement learning in activity recognition

Inspired by the way humans learn to optimally behave in different environments, reinforcement learning algorithms try to learn how to obtain a complex goal through interaction with the environment and getting reward or punishment. The reward is designed based on the final goal(s) of agent and the agent's objective is to maximize the received reward. There are some researches in the field of computer vision using Reinforcement learning such as [24], [26], and [25] in which RL is used for image recognition, visual tracking and face recognition. However, there are few for activity recognition especially for skeleton-based data. In [22], multi-agent reinforcement learning is used to select key frames in videos where each agent is responsible for selecting a frame. As a result, the number of selected frames is fixed. Dong et al. proposed an RL-based method which finds the most relevant frames using an LSTM agent [23]. Both of the mentioned methods are proposed for RGB data. In [27], authors proposed an RL-based technique called deep progressive reinforcement learning (DPRL) to select key frames in skeleton video. This method uses a graph representation of data, and a graph CNN is used as agent and for reward generation.

III. Proposed Method

The proposed JSDRL method models the joint selection problem as an MDP and solves that with the well-known off policy reinforcement learning algorithm, i.e., Monte Carlo policy gradient (i.e. REINFORCE) [28]. A typical RL algorithm has an agent in its current state of the environment. The agent takes an action that changes its state and receives a reward based on it.

In this paper, we define the k^{th} step of our RL episode as $\mathcal{T}_k = (S_k, A_k, R_k)$, where S_k, A_k and R_k are respectively state, action and reward at the k^{th} step; the full episode of the proposed RL system can be shown as $\mathcal{T} = (S_1, A_1, R_1 ..., S_K, A_K, R_K)$. At each step of episode, the agent goes over all T frames of a given video. Agent, State, Action, and Reward in the proposed joint selection framework are defined as follows:

Agent: Human skeleton can be considered as an ordered sequence of J joints. In this study, we propose to employ Bidirectional LSTM (BiLSTM) network followed by a fully connected (FC) network as the agent. At frame t of $\mathcal{T}_k$, the BiLSTM network takes the state S_k^t (where $S_k = \{S_k^t\}_{t=1}^T$) as input and then feeds its hidden layer, $\{h_j\}_{j=1}^J$, to the FC network. The agent outputs vector $\{p_j^t\}_{j=1}^J$ that is used to define the next action.

State: In skeleton based human activity recognition, it has been shown that both joints location and joints motion are informative components. Hence, we define the agent's state at frame t of $\mathcal{T}_k$ as $S_k^t = \{\mathbf{s}_j^t\}_{j=1}^J$ where $\mathbf{s}_j^t = [\mathbf{s}_{j,c}^t, \mathbf{s}_{j,m}^t]$, $\mathbf{s}_{j,c}^t$ is the 3-dim coordinates of the j^{th} joint, and $\mathbf{s}_{j,m}^t$ is the j^{th} joint 3-dim motion vector, i.e. $\mathbf{s}_{j,m}^t = \mathbf{s}_{j,c}^t - \mathbf{s}_{j,c}^{t-1}$.

Action: Consider $T \times J$ matrix $F_k = \{\mathbf{f}_k^1, \dots, \mathbf{f}_k^T\}$, where $\mathbf{f}_k^t$ is a J-dim indicator vector, showing joints that are selected at frame t of $\mathcal{T}_k$. If the j^{th} element of $\mathbf{f}_k^t$ is 1, i.e. $f_{k,j}^t = 1$, then the j^{th} joint is selected for frame t, otherwise it is not. We initialize elements of F_k, $k = 1, ..., K$, with 1. The action taken at frame t of $\mathcal{T}_k$, i.e. $\mathbf{a}_k^t$, is a J-dim vector showing the adjustment needed to be applied to

$\mathbf{f}_{k-1}^t$ to obtain $\mathbf{f}_k^t$. We define two types of actions: 0 and 1, where 0 means no change is needed and 1 means flip the corresponding selection bit. The outputs of the FC network of the agent at the t^{th} frame, $\{p_j^t\}_{j=1}^J$, indicates the probability of changing elements of $\mathbf{f}_{k-1}^t$. Finally, the J elements of action vector at frame t of $\mathcal{T}_k$, $\mathbf{a}_k^t$, are sampled from Bernoulli distributions as follows:

$$\mathbf{a}_k^t = \{a_{k,j}^t \sim \text{Bernoulli}(p_j^t)\}_{j=1}^J \tag{1}$$

$a_{k,j}^t = 1$ indicates flip the j^{th} element of $\mathbf{f}_{k-1}^t$ to obtain the j^{th} element of $\mathbf{f}_k^t$ – i.e. if the j^{th} joint is selected (removed) in the previous step, it will be removed (selected) in the future step, and $a_{k,j}^t = 0$ means no change is needed. In this way, we allow the removed joints to be selected in the next episode if they were erroneously removed from the selected joint set. This changing process is shown below:

$$f_{k,j}^t = \begin{cases} f_{k-1,j}^t & \text{if } a_{k,j} = 0, \\ 1 - f_{k-1,j}^t & \text{if } a_{k,j} = 1. \end{cases} \tag{2}$$

The total action set corresponding to the k^{th} episode is $A_k = \{\mathbf{a}_k^t\}_{t=1}^T$.

Reward: The reward reflects how good the action taken by agent is with regard to the state. We generate the reward with a pre-trained classifier which takes the T frames with selected joints as input, where joints are selected by the agent. If the class label predicted by the classifier turns from the correct label to a wrong one, a strong punishment $-\Omega$ is enforced and a strong reward of Ω is enforced if the turning goes otherwise. Further, if the predicted class label does not change, but the confidence of classifier towards predicting the correct class changes, reward r_0 is given, which is defined as below:

$$r_0 = \text{sgn}(P_l^k - P_l^{k-1}), \tag{3}$$

where P_l^k is the probability of correctly classifying the video as class l in $\mathcal{T}_k$. The Reward at $\mathcal{T}_k$, i.e. R_k can be shown as below:

$$R_k = \begin{cases} \Omega & \text{if reward} \\ -\Omega & \text{if punishment} \\ r_0 & \text{otherwise} \end{cases} \tag{4}$$

The goal of agent is learning a policy function by maximizing the expected reward shown below:

$$\mathcal{R}(\theta) = \text{E}_{p_\theta(a_{k,1:J}^{1:T})}[R_k], \tag{5}$$

where $p_\theta(a_{k,1:J}^{1:T})$ is the probability distribution of the possible actions over the frames. In Policy Gradient algorithms, the policy is usually modeled with a function parameterized by θ, and in REINFORCE, which is a policy gradient method [13], the gradient of the expected reward $\mathcal{R}(\theta)$ w.r.t. the parameters θ is calculated as:

$$\nabla_\theta \mathcal{R}(\theta) = \text{E}_{p_\theta(a_{k,1:J}^{1:T})}[R_k \sum_{t=1}^{T} \sum_{j=1}^{J} \nabla_\theta \, ln \, \pi_\theta(a_{k,j}^t | s_{k,j}^t)], \tag{6}$$

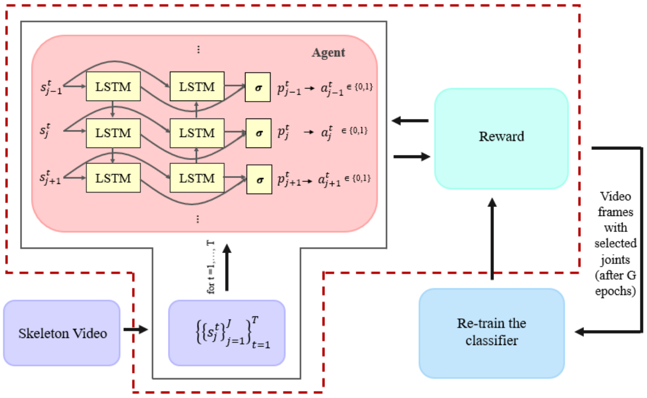
Agent
s_{j-1}^t
s_j^t
s_{j+1}^t
LSTM
σ
$p_{j-1}^t \rightarrow a_{j-1}^t \in \{0,1\}$
$p_j^t \rightarrow a_j^t \in \{0,1\}$
$p_{j+1}^t \rightarrow a_{j+1}^t \in \{0,1\}$
Reward
Video frames with selected joints (after G epochs)
for t =1,..., T
Skeleton Video
$\left\{\{s_j^t\}_{j=1}^J\right\}_{t=1}^T$
Re-train the classifier

Fig. 1: Block diagram of the proposed joint selection method JSDRL.

where π_θ is the policy function, $a_{k,j}$ is the action taken by the agent at $\mathcal{T}_k$ for the joint j and $s_{k,j}$ is the corresponding state.

To simplify Eq. (5), instead of taking the expectation over action sequence, and as we get the reward after observing the whole T frames, we approximate the gradient by taking average of gradients over the total T frames and K steps as follows:

$$\nabla_\theta \mathcal{R}(\theta) \approx \frac{1}{KT} \sum_{k=1}^{K} [R_k \sum_{t=1}^{T} \sum_{j=1}^{J} \nabla_\theta \, ln \, \pi_\theta(a_{k,j}^t | s_{k,j}^t)], \tag{7}$$

where R_k is the reward computed at the k^{th} step of episode. To reduce the variance and guarantee the convergence of the algorithm, a constant baseline b, which is the average rewards of steps, is reduced from the reward as follows:

$$\nabla_\theta \mathcal{R}(\theta) \approx \frac{1}{KT} \sum_{k=1}^{K} [(R_k - b) \sum_{t=1}^{T} \sum_{j=1}^{J} \nabla_\theta \, ln \, \pi_\theta(a_{k,j}^t | s_{k,j}^t)], \tag{8}$$

To make sure the agent selects at least one joint and does not select more than N joints, we propose to add two other terms to the loss function along with the REINFORCE loss as below:

$$\min_\theta -\mathcal{R}(\theta) + \alpha(\mathbf{1}^T \mathbf{p} - N) - \beta(\mathbf{1}^T \mathbf{p}), \tag{9}$$

where $\mathbf{p}$ is the average probability of vector of actions over the T frames and K steps and N is the maximum number of selected joints, and α and β are two hyper-parameters that control the effect of their corresponding terms.

Block diagram of the proposed framework is shown in Fig. 1. Pseudo code of the proposed JSDRL method is shown in Algorithm 1. In summary, first the classifier is pre-trained on the original training data. Then, a video sequence is given to the agent's network (aka policy network), an episode is completed, and the network is updated. This process is repeated for all epochs where the classifier is re-trained every G epochs.

Algorithm 1 The proposed JSDRL method

Input: The video sequences with labels

Output: Trained agent (i.e. policy network)

```
Pre-train the classifier and Initialize the agent (policy) network
Count = 0
for epochs do
    for videos do
        Count += 1
        for $K$ steps of episode do
            run the policy network
            find the action using Eq. (1)
            take the action and update the state (select joints)
            compute reward using Eq. (3) and Eq. (4)
        end for
        compute the average reward
        compute the loss (Eq. 9)
        update the policy network parameters
        if Count$\leq G$ then
            retrain the classifier
        end if
    end for
end for
```

IV. Experiments

To evaluate the performance of the proposed JSDRL method, we conducted experiments on two benchmark activity recognition datasets. To demonstrate the effectiveness of joint selection in activity recognition, we show recognition results with and without joint selection using three classifiers: CNN-based, BiLSTM-based and Graph-based.

A. Data sets

NTU+RGBD Dataset (NTU) [20]: NTU is currently the largest activity recognition data with 56,880 sequences and 4 million frames. The video samples belong to 60 classes, and there are two settings for train/test sample partitioning: Cross-Subject (CS) and Cross-View (CV). In the CS setting, samples of 20 subjects are used as train and the remaining ones are used for testing. In the CV setting, samples of camera views 2 and 3 are selected as the train set and samples captured by camera 1, are used as the test set. The number of skeleton joints captured for this data set is 25 and there are either one or two subjects in each video.

UT-Kinect Dataset (UT) [29]: UT includes 200 sequences belonging to 10 classes. Each activity is performed by 10 subjects twice and there is no interactive activity in the data which means there is only one subject in each video sample. There are 20 joints in each frame and Leave-one-out cross-validation protocol is used to evaluate the proposed method on this data.

B. Implementation Details

We use BiLSTM with 3 layers as the agent's network (i.e. policy network) and the optimizer is Adam with initial learning rate 1e-4. The number of epochs, values of K, Ω α, and β are respectively set to 20, 5, 10, 0.1 and 0.1. We divide the number of video samples to 5 and use that as the value of G. The value of N is set to half of the number of available joints. The proposed method was implemented with Pytorch.

Effectiveness of the proposed JSDRL method is demonstrated using three different classifiers including the two basic classifiers BiLSTM and CNN and a state-of-the-art graph-based classifier which is specifically designed for skeletonbased human activity recognition, i.e. decoupling graph Convolutional neural networks with dropGraph module (DCGCN) [6]. The DCGCN parameters are set to their default value suggested in the original paper. The BiLSTM classifier has 3 layers with hidden layer size 256, where it is trained using Adam optimization method. The CNN classifier has 2 convolution layers followed by one fully connected layer, and the optimizer is Adam.

C. Recognition Accuracy

The classification accuracy with and without joint selection, i.e. applying JSDRL, for the two datasets are reported in Table I, where the best performance is shown in bold.

As can be seen, the proposed JSDRL method improves the classifiers performance for both the two datasets. The average performance of each Method over the three sets CS, CV and UT are shown at the last column. The average values confirm the improved performance of the proposed method compare to without-joint-selection cases. That is while on average, almost 60 % of joints are eliminated leading to a decline in classification cost in both training and testing phases.

In Tables II and III, performance of JSDRL (with the DCGCN classifier) is compared with several state-of-the-art activity recognition methods. Table II shows superior performance of the proposed method, on both CS and CV settings of the NTU dataset, to its eight competitors. Table III shows that the proposed method outperforms ten state-of-the-art skeleton-based activity recognition classifiers, on the UT dataset.

To visualize the performance of the proposed method, the resultant selected joints for the two activities kick and phone call are depicted in Fig. 2. The intensity of red color at each joint indicates the frequency of selecting that joint over the whole video frames; e.g. in the activity phone call, hand,

TABLE I: Recognition accuracy (in percent) of three different classifiers with and without JSDRL. CS and CV respectively indicate the cross subject and cross view settings of the NTU dataset.

Method	CS	CV	UT	average
BiLSTM	65	69.01	94.5	76.17
CNN	70.2	71.3	91.9	77.8
DCGCN [6]	88.1	95.2	98.2	93.83
BiLSTM+JSDRL	**67**	**71.5**	**96.9**	**78.46**
CNN+JSDRL	**75.6**	**80.5**	**95.8**	**83.96**
DCGCN+JSDRL	**89.3**	**96**	**99.1**	**94.8**

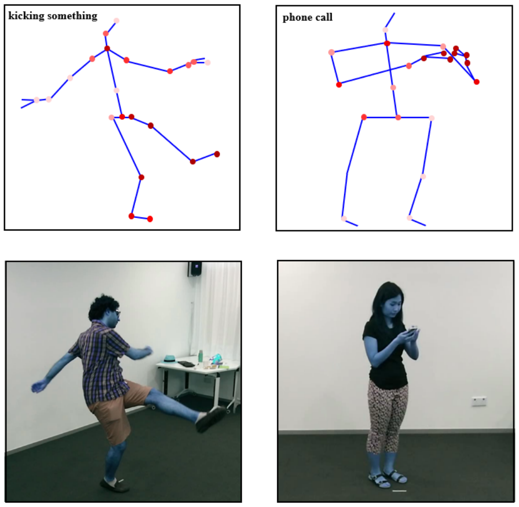

Fig. 2: Visualizing the frequency of selected joints by JSDRL; a lighter spot means the corresponding joint is selected less frequently over the frames of a video and a darker one means a higher selection frequency.

TABLE II: Recognition accuracy (in percent) of different methods on NTU dataset.

Method	CS	CV	year
Lie Group [8]	50.1	52.8	2014
HBRNN[40]	59.1	64.0	2015
Part-aware LSTM [20]	62.9	70.3	2016
LieNet-3Blocks [39]	61.4	67.0	2017
Mengyuan et al. [38]	76	82.56	2017
ST-GCN [37]	81.5	88.3	2018
DPRL [27]	83.5	89.8	2018
DCGCN [6]	88.1	95.2	2020
JSDRL	**89.3**	**96**	

thumb and fingers tip are correctly selected in all frames and the irrelevant foot and head joints are not selected in any frame. This figure demonstrates the effectiveness of JSDRL method in selecting relevant joints.

D. Sensitivity to hyperparameter N

To investigate the sensitivity of the JSDRL method to the hyperparameter N, introduced in (9), we apply JSDRL+BiLSTM to the UT dataset for different N values, i.e. $N \in \{3, 6, 10, 12, 15, 20\}$. Note that J is equal to 20 in the UT dataset. The role of N in the loss function is to set an upper bound on the number of selected joints. Accuracy of activity recognition versus N is shown in Figure 3. The figure shows that the JSDRL method retains high accuracy for a wide range of N, demonstrating that JSDRL method is not too sensitive to N, which is a desirable behaviour.

V. Conclusion

In this paper, we proposed a deep RL-based joint selection method, JSDRL, that models the joint selection problem as a Markov Decision Process and finds the most informative joints in each frame of skeleton data using the popular policy gradient algorithm, REINFORCE. In JSDRL, each video frame is associated with its own distinct optimal joint set, which may vary both in membership and size across the video. This allows the joint set to optimally adapt to temporal variations. Employing reinforcement learning in the JSDRL method allows to find relevant joints, per frame, without requiring any extra labels. The JSDRL can be used as a filtering block, to identify and filter out irrelevant joints, prior to any sophisticated activity classification algorithm; this enhances the classifier performance and reduces the training time. We evaluated the JSDRL method on two benchmark skeleton-based activity recognition datasets employing three different classifiers. The experimental results demonstrated the effectiveness of JSDRL. Furthermore, the proposed JSDRL method outperforms, in terms of recognition accuracy, several state-of-the-art skeleton-based activity recognition methods.

TABLE III: Recognition accuracy (in percent) of different methods on UT-Kinect dataset.

Method	UT	year
Histogram of 3D Joints [36]	90.9	2012
Riemannian Manifold [35]	91.5	2015
Grassmann Manifold [34]	88.5	2015
GMSM [33]	97.4	2016
SCK+DCK [32]	98.2	2016
ST-LSTM+Trust Gate [31]	97.0	2017
ST-NBNN [30]	98.0	2017
DPRL+GCNN [27]	98.5	2018
DCGCN [6]	98.2	2020
JSDRL	**99.1**	

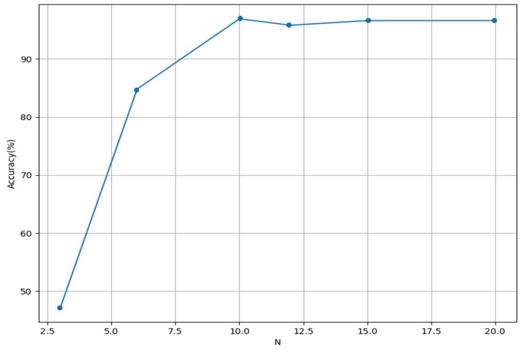

Fig. 3: Accuracy of JSDRL-BiLSTM vs. hyperparameter N for the UT-Kinect dataset.

Acknowledgment

The authors wish to acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC). This research was enabled in part by support provided by Calcul Québec and Compute Canada.

References

[1] Foggia P, Percannella G, Saggese A, Vento M. Recognizing human actions by a bag of visual words. In2013 IEEE International Conference on Systems, Man, and Cybernetics 2013 Oct 13 (pp. 2910-2915). IEEE.

[2] Chen S, Liu J, Wang H, Augusto JC. A hierarchical human activity recognition framework based on automated reasoning. In2013 IEEE International Conference on Systems, Man, and Cybernetics 2013 Oct 13 (pp. 3495-3499). IEEE.

[3] Liao LC, Yang YH, Fu LC. Joint-oriented Features for Skeleton-based Action Recognition. In2019 IEEE International Conference on Systems, Man and Cybernetics (SMC) 2019 Oct 6 (pp. 1154-1159). IEEE.

[4] Liu B, Yu H, Zhou X, Tang D, Liu H. Combining 3D joints moving trend and geometry property for human action recognition. In2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC) 2016 Oct 9 (pp. 000332-000337). IEEE.

[5] Bellman R. A Markovian decision process. Journal of mathematics and mechanics. 1957 Jan 1;6(5):679-84.

[6] Cheng K, Zhang Y, Cao C, Shi L, Cheng J, Lu H. Decoupling GCN with DropGraph Module for Skeleton-Based Action Recognition.

[7] Shi L, Zhang Y, Cheng J, Lu H. Skeleton-based action recognition with directed graph neural networks. InProceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2019 (pp. 7912-7921).

[8] Vemulapalli R, Arrate F, Chellappa R. Human action recognition by representing 3D skeletons as points in a lie group. InProceedings of the IEEE conference on computer vision and pattern recognition 2014 (pp. 588-595).

[9] Weng J, Weng C, Yuan J. Spatio-temporal naive-bayes nearest-neighbor (st-nbnn) for skeleton-based action recognition. InProceedings of the IEEE Conference on computer vision and pattern recognition 2017 (pp. 4171-4180).

[10] Wang P, Yuan C, Hu W, Li B, Zhang Y. Graph based skeleton motion representation and similarity measurement for action recognition. InEuropean conference on computer vision 2016 Oct 8 (pp. 370-385). Springer, Cham.

[11] Wang H, Wang L. Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks. InProceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017 (pp. 499-508).

[12] Song S, Lan C, Xing J, Zeng W, Liu J. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. InProceedings of the AAAI conference on artificial intelligence 2017 Feb 12 (Vol. 31, No. 1).

[13] Ren B, Liu M, Ding R, Liu H. A survey on 3d skeleton-based action recognition using learning method. arXiv preprint arXiv:2002.05907. 2020 Feb 14.

[14] Li C, Zhong Q, Xie D, Pu S. Skeleton-based action recognition with convolutional neural networks. In2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW) 2017 Jul 10 (pp. 597-600). IEEE.

[15] Ke Q, Bennamoun M, An S, Sohel F, Boussaid F. A new representation of skeleton sequences for 3D action recognition. InProceedings of the IEEE conference on computer vision and pattern recognition 2017 (pp. 3288-3297).

[16] Liu M, Liu H, Chen C. Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition. 2017 Aug 1;68:346-62.

[17] Yan S, Xiong Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition. InProceedings of the AAAI conference on artificial intelligence 2018 Apr 27 (Vol. 32, No. 1).

[18] Shi L, Zhang Y, Cheng J, Lu H. Skeleton-based action recognition with multi-stream adaptive graph convolutional networks. IEEE Transactions on Image Processing. 2020 Oct 9;29:9532-45.

[19] Koniusz P, Cherian A, Porikli F. Tensor representations via kernel linearization for action recognition from 3D skeletons. InEuropean conference on computer vision 2016 Oct 8 (pp. 37-53). Springer, Cham.

[20] Shahroudy A, Liu J, Ng TT, Wang G. NTU RGB+D: A large scale dataset for 3D human activity analysis. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 1010-1019).

[21] Yun K, Honorio J, Chattopadhyay D, Berg TL, Samaras D. Two-person interaction detection using body-pose features and multiple instance learning. In2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops 2012 Jun 16 (pp. 28-35). IEEE.

[22] Wu W, He D, Tan X, Chen S, Wen S. Multi-agent reinforcement learning based frame sampling for effective untrimmed video recognition. InProceedings of the IEEE/CVF International Conference on Computer Vision 2019 (pp. 6222-6231).

[23] Dong W, Zhang Z, Tan T. Attention-aware sampling via deep reinforcement learning for action recognition. InProceedings of the AAAI Conference on Artificial Intelligence 2019 Jul 17 (Vol. 33, No. 01, pp. 8247-8254).

[24] Chen T, Wang Z, Li G, Lin L. Recurrent attentional reinforcement learning for multi-label image recognition. InProceedings of the AAAI Conference on Artificial Intelligence 2018 Apr 27 (Vol. 32, No. 1).

[25] Rao Y, Lu J, Zhou J. Attention-aware deep reinforcement learning for video face recognition. InProceedings of the IEEE international conference on computer vision 2017 (pp. 3931-3940).

[26] Yun S, Choi J, Yoo Y, Yun K, Young Choi J. Action-decision networks for visual tracking with deep reinforcement learning. InProceedings of the IEEE conference on computer vision and pattern recognition 2017 (pp. 2711-2720).

[27] Tang Y, Tian Y, Lu J, Li P, Zhou J. Deep progressive reinforcement learning for skeleton-based action recognition. InProceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018 (pp. 5323-5332).

[28] Williams RJ. Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning. 1992 May 1;8(3-4):229-56.

[29] Xia L, Chen CC, Aggarwal JK. View invariant human action recognition using histograms of 3D joints. In2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops 2012 Jun 16 (pp. 20-27). IEEE.

[30] Weng J, Weng C, Yuan J. Spatio-temporal naive-bayes nearest-neighbor (st-nbnn) for skeleton-based action recognition. InProceedings of the IEEE Conference on computer vision and pattern recognition 2017 (pp. 4171-4180).

[31] Liu J, Shahroudy A, Xu D, Kot AC, Wang G. Skeleton-based action recognition using spatio-temporal LSTM network with trust gates. IEEE transactions on pattern analysis and machine intelligence. 2017 Nov 9;40(12):3007-21.

[32] Koniusz P, Cherian A, Porikli F. Tensor representations via kernel linearization for action recognition from 3D skeletons. InEuropean conference on computer vision 2016 Oct 8 (pp. 37-53). Springer, Cham.

[33] Wang P, Yuan C, Hu W, Li B, Zhang Y. Graph based skeleton motion representation and similarity measurement for action recognition. InEuropean conference on computer vision 2016 Oct 8 (pp. 370-385). Springer, Cham.

[34] Slama R, Wannous H, Daoudi M, Srivastava A. Accurate 3D action recognition using learning on the Grassmann manifold. Pattern Recognition. 2015 Feb 1;48(2):556-67.

[35] Devanne M, Wannous H, Berretti S, Pala P, Daoudi M, Del Bimbo A. 3D human action recognition by shape analysis of motion trajectories on riemannian manifold. IEEE transactions on cybernetics. 2014 Sep 9;45(7):1340-52.

[36] Xia L, Chen CC, Aggarwal JK. View invariant human action recognition using histograms of 3D joints. In 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops 2012 Jun 16 (pp. 20-27). IEEE.

[37] Yan S, Xiong Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In Proceedings of the AAAI conference on artificial intelligence 2018 Apr 27 (Vol. 32, No. 1).

[38] Liu M, Liu H, Chen C. Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition. 2017 Aug 1;68:346-62.

[39] Huang Z, Wan C, Probst T, Van Gool L. Deep learning on lie groups for skeleton-based action recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition 2017 (pp. 6099-6108).

[40] Du Y, Wang W, Wang L. 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).