Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition

Jun Liu†, Amir Shahroudy†, Dong Xu‡, and Gang Wang†,⋆

† School of Electrical and Electronic Engineering, Nanyang Technological University
‡ School of Electrical and Information Engineering, University of Sydney
{jliu029,amir3,wanggang}@ntu.edu.sg dong.xu@sydney.edu.au

对时序的语境依赖建模

Abstract. 3D action recognition – analysis of human actions based on 3D skeleton data – becomes popular recently due to its succinctness, robustness, and view-invariant representation. Recent attempts on this problem suggested to develop RNN-based learning methods to model the contextual dependency in the temporal domain. In this paper, we extend this idea to spatio-temporal domains to analyze the hidden sources of action-related information within the input data over both domains concurrently. Inspired by the graphical structure of the human skeleton, we further propose a more powerful tree-structure based traversal method. To handle the noise and occlusion in 3D skeleton data, we introduce new gating mechanism within LSTM to learn the reliability of the sequential input data and accordingly adjust its effect on updating the long-term context information stored in the memory cell. Our method achieves state-of-the-art performance on 4 challenging benchmark datasets for 3D human action analysis.

精简 鲁棒性 视角不变性的表示特征

同时两个领域行为相关信息

Keywords: 3D action recognition, recurrent neural networks, long short-term memory, trust gate, spatio-temporal analysis.

1 Introduction

In recent years, action recognition based on the locations of major joints of the body in 3D space has attracted a lot of attention. Different feature extraction and classifier learning approaches are studied for 3D action recognition [1–3]. For example, Yang and Tian [4] represented the static postures and the dynamics of the motion patterns via eigenjoints and utilized a Naïve-Bayes-Nearest-Neighbor classifier learning. A HMM was applied by [5] for modeling the temporal dynamics of the actions over a histogram-based representation of 3D joint locations. Evangelidis *et al.* [6] learned a GMM over the Fisher kernel representation of a succinct skeletal feature, called skeletal quads. Vemulapalli *et al.* [7] represented the skeleton configurations and actions as points and curves in a Lie group respectively, and utilized a SVM classifier to classify the actions. A skeleton-based dictionary learning utilizing group sparsity and geometry constraint was also

⋆ Corresponding author.

proposed by [8]. An angular skeletal representation over the tree-structured set of joints was introduced in [9], which calculated the similarity of these features over temporal dimension to build the global representation of the action samples and fed them to SVM for final classification.

Recurrent neural networks (RNNs) which are a variant of neural nets for handling sequential data with variable length, have been successfully applied to language modeling [10–12], image captioning [13, 14], video analysis [15–24], human re-identification [25,26], and RGB-based action recognition [27–29]. They also have achieved promising performance in 3D action recognition [30–32].

Existing RNN-based 3D action recognition methods mainly model the long-term contextual information in the temporal domain to represent motion-based dynamics. However, there is also strong dependency between joints in the spatial domain. And the spatial configuration of joints in video frames can be highly discriminative for 3D action recognition task.

In this paper, we propose a spatio-temporal long short-term memory (ST-LSTM) network which extends the traditional LSTM-based learning to two concurrent domains (temporal and spatial domains). Each joint receives contextual information from neighboring joints and also from previous frames to encode the spatio-temporal context. Human body joints are not naturally arranged in a chain, therefore feeding a simple chain of joints to a sequence learner cannot perform well. Instead, a tree-like graph can better represent the adjacency properties between the joints in the skeletal data. Hence, we also propose a tree structure based skeleton traversal method to explore the kinematic relationship between the joints for better spatial dependency modeling.

In addition, since the acquisition of depth sensors is not always accurate, we further improve the design of the ST-LSTM by adding a new gating function, so called "trust gate", to analyze the reliability of the input data at each spatio-temporal step and give better insight to the network about when to update, forget, or remember the contents of the internal memory cell as the representation of long-term context information.

The contributions of this paper are: (1) spatio-temporal design of LSTM networks for 3D action recognition, (2) a skeleton-based tree traversal technique to feed the structure of the skeleton data into a sequential LSTM, (3) improving the design of the ST-LSTM by adding the trust gate, and (4) achieving state-of-the-art performance on all the evaluated datasets.

2 Related Work

Human action recognition using 3D skeleton information is explored in different aspects during recent years [33–50]. In this section, we limit our review to more recent RNN-based and LSTM-based approaches.

HBRNN [30] applied bidirectional RNNs in a novel hierarchical fashion. They divided the entire skeleton to five major groups of joints and each group was fed into a separated bidirectional RNN. The output of these RNNs were concatenated to represent upper-body and lower-body, then each was fed into another

set of RNNs. The global body representation was obtained by concatenating the output of these two RNNs and it was fed to the next layer of RNN. The hidden representation of the final RNN was fed to a softmax classifier layer for action classification.

Zhu *et al.* [51] added a mixed-norm regularization term to a deep LSTM network's cost function in order to push the network towards learning co-occurrence of discriminative joints for action classification. They further introduced an internal dropout [52] technique within the LSTM unit, which was applied on all the gate activations.

Differential LSTM [31] added a new gating inside LSTM to keep track of the derivatives of the memory states in order to discover patterns within salient motion patterns. All the input features for each frame were concatenated and fed to the differential LSTM.

Part-aware LSTM [32] separated the memory cell to part-based sub-cells and pushed the network towards learning the long-term context representations individually for each part. The output of the network was learned over the concatenated part-based memory cells followed by the common output gate.

Unlike the above mentioned works, the framework proposed in this paper does not concatenate the joint-based input features, instead it explicitly models the dependencies between the joints and applies recurrent analysis over spatial and temporal domains concurrently. Besides, a novel trust gate is developed to make LSTM robust to noisy input data.

3 Spatio-Temporal Recurrent Networks

Human actions can be characterized by the motion of body parts over time. In 3D human action recognition, we have three dimensional locations of the major body joints in each frame. Recently, recurrent neural networks have been successfully employed for skeleton-based 3D action recognition [30, 32, 51].

Long Short-Term Memory (LSTM) networks [53] are very successful extensions of the recurrent neural networks (RNNs). They utilize the gating mechanism over an internal memory cell to learn and represent a better and more complex representation of the long-term dependencies among the input sequential data, thus they are suitable for feature learning over a sequence of temporal data.

In this section, first we will briefly review the standard LSTM networks, then describe the proposed spatio-temporal LSTM model and the skeleton-based tree traversal. Next we will introduce an effective gating scheme for LSTM to deal with the measurement noise in the input data (body joint locations) for the task of 3D human action recognition. 以测量噪声作为输入数据进行三维人体行为识别

3.1 Temporal Modeling with LSTM

A typical LSTM unit contains an input gate i_t, a forget gate f_t, an output gate o_t, and an output state h_t, together with an internal memory cell state c_t. The LSTM transition equations are formulated as:

过度方程

$$\begin{pmatrix} i_t \\ f_t \\ o_t \\ u_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} \left(M \begin{pmatrix} x_t \\ h_{t-1} \end{pmatrix} \right) \tag{1}$$

$$c_t = i_t \odot u_t + f_t \odot c_{t-1} \tag{2}$$

$$h_t = o_t \odot \tanh(c_t) \tag{3}$$

where $\odot$ indicates element-wise product, x_t denotes the input to the network at time step t, and u_t denotes the modulated input. σ is the sigmoid activation function. $M : \Re^{D+d} \rightarrow \Re^{4d}$ is an affine transformation consisting of model parameters, where D is the dimensionality of input x_t and d is the number of LSTM cell state units.

确定了... ...的范围

Intuitively, the input gate i_t determines the extent to which the modulated input information (u_t) is supposed to update the memory cell at time t. The forget gate f_t determines the effectiveness of the previous state of the memory cell (c_{t-1}) on its current state (c_t). Finally, the output gate o_t governs the amount of information output from the memory cell. Readers are referred to [54] for more details about the mechanism of LSTM.

3.2 Spatio-Temporal LSTM

Very recent attempts on applying RNNs for 3D human action recognition [30–32, 51] show outstanding performance and prove the strengths of RNNs in modeling the complex dynamics of the human actions in temporal space.

The main focus of these existing methods was on utilizing RNNs over temporal domain for discovering the discriminative dynamics and body motion patterns for 3D action recognition. However, there is also discriminative information in static postures encoded within the joints' 3D locations in each individual frame and the sequential nature of skeleton data makes it possible to adopt RNN-based learning in spatial domain as well. Unlike other existing methods, which concatenated the joints information, we extend the recurrent analysis towards spatial domain to discover the spatial dependency patterns between different joints at each frame.

In this fashion, we propose a spatio-temporal LSTM (ST-LSTM) model which simultaneously models the spatial dependencies of the joints and the temporal dependencies among the frames. As shown in Fig. 1, every ST-LSTM unit corresponds to one of the skeletal joints. Each of the units receives the hidden representation of the previous joint and also the hidden representation of its own joint from the previous frame. In this section we assume joints are arranged in a chain-like sequence with the order shown in Fig. 2(a). In Section 3.3, we will show a more advanced method to take advantage of the adjacency information of the body joints as a tree structure.

We use $j \in \{1, ..., J\}$ and $t \in \{1, ..., T\}$ to denote the indices of joints and frames respectively. Each ST-LSTM unit is fed with its input ($x_{j,t}$, location of

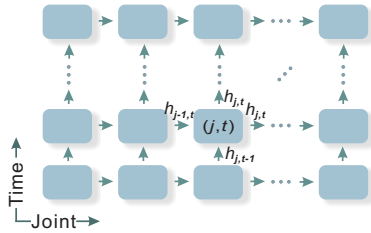

Fig. 1. The illustration of the proposed spatio-temporal LSTM network. In the spatial direction, body joints in a frame are fed in a sequence. In the temporal direction, the locations of the corresponding joints are fed over time. Each unit receives the hidden representation of previous joints and previous frames of the same joint as contextual information.

the corresponding joint at current frame), its own hidden representation at the previous time step ($h_{j,t-1}$), and the hidden representation of the previous joint at current frame ($h_{j-1,t}$). Each unit is also equipped with two different forget gates corresponding to the two incoming channels of context information: $f^S_{j,t}$ for the spatial domain, and $f^T_{j,t}$ for the temporal domain. The proposed ST-LSTM is formulated as:

$$\begin{pmatrix} i_{j,t} \\ f^S_{j,t} \\ f^T_{j,t} \\ o_{j,t} \\ u_{j,t} \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} \left(M \begin{pmatrix} x_{j,t} \\ h_{j-1,t} \\ h_{j,t-1} \end{pmatrix} \right) \tag{4}$$

$$c_{j,t} = i_{j,t} \odot u_{j,t} + f^S_{j,t} \odot c_{j-1,t} + f^T_{j,t} \odot c_{j,t-1} \tag{5}$$

$$h_{j,t} = o_{j,t} \odot \tanh(c_{j,t}) \tag{6}$$

3.3 Tree-Structure based Traversal

Arranging joints in a simple chain ignores the kinematic dependency relations between the joints and adds false connections between body joints which are not strongly related. In human parsing, skeletal joints are popularly modeled as a tree-based pictorial structure [55,56], as illustrated in Fig. 2(b). In our ST-LSTM framework, it is also beneficial to model the spatial dependency of the joints based on their adjacency tree structure. For example, hidden representation of the neck joint (number 2 in Fig. 2(a)) is expected to be more informative for the right hand joints (7,8,9) than the joint number 6.

However, trees cannot be directly fed into the ST-LSTM framework. To mitigate this issue, we propose a bidirectional tree traversal method to visit joints in a sequence which maintains the adjacency information of the skeletal tree structure.

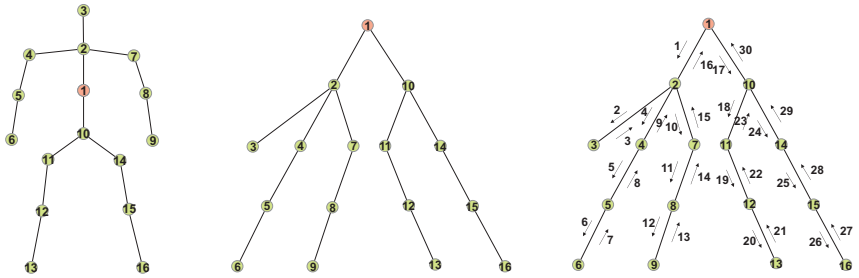

Fig. 2. (a) Skeletal joints of a human body. In the simple joint chain model, the joint visiting order is 1-2-3-...-16. (b) Skeleton is transformed to a tree structure. (c) Tree traversal over the spatial steps. The tree can be unfolded to a chain with the traversal, and the joint visiting order is 1-2-3-2-4-5-6-5-4-2-7-8-9-8-7-2-1-10-11-12-13-12-11-10-14-15-16-15-14-10-1.

As illustrated in Fig. 2(c), at the first spatial step, the root node (central spine joint) is fed to the network, then the network follows a depth-first traversal in the spatial domain. When it reaches a leaf node, it goes back. In this fashion, each connection of the tree structure will be passed twice and the context information is fed along both directions. Upon the end of the traversal, it gets back to the root node.

This traversal strategy guarantees the transmission of the data in both directions (top-down and bottom-up) inside the adjacency tree structure. Therefore each node will have the contextual information from both its descendants and ancestors. Compared to the simple chain model described in section 3.2, this tree traversal technique can discover stronger long-term spatial dependency patterns based on the joints' adjacency structure.

In addition, the input to the ST-LSTM network at each step is limited to a single joint in a specific frame, which is much smaller in size compared to the concatenated input features of other existing methods. As a result, we have much fewer model parameters and this can be considered as a weight sharing regularization inside our learning framework, which leads to better generalization in the scenarios with limited training samples. This is an advantage in 3D action recognition, because most of the current datasets have a small number of training samples.

Similar to other LSTM implementations [57, 58], the representation capacity of our network can be improved by stacking multiple layers of the tree structured ST-LSTMs and constructing a deep yet completely tractable network, as illustrated in Fig. 3. 深层的但是完全可处理的网络。

3.4 Spatio-Temporal LSTM with Trust Gates

The inputs of the proposed tree-structured ST-LSTM are the 3D positions of skeletal joints collected by sensors like Microsoft Kinect, which are not always

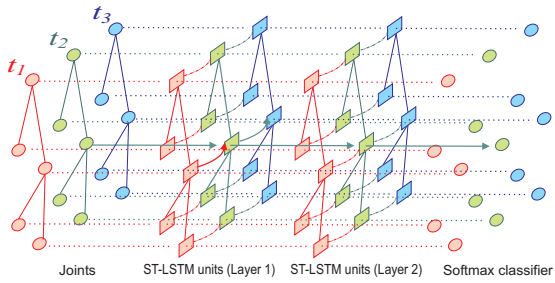

Fig. 3. A graphical model of the deep tree-structured ST-LSTM network. For clarity, some arrows are omitted in the stacked network (better viewed in color). In this figure, the output of the first ST-LSTM layer is fed to the second ST-LSTM layer as its input. The second ST-LSTM layer's output is fed to softmax layer.

reliable due to noise and occlusion. This limits the performance of the network. To address this issue, we propose to add a new gate to the LSTM unit which analyzes the reliability of the input at each spatio-temporal step, based on the estimation of the input from the available contextual information.

Our novel gating method is inspired by the works in natural language processing [58] which predict next word based on LSTM representation of previous words. This idea worked well because of the high dependency among the words in a sentence. Similarly, since the skeletal joints often move together and this articulated motion follows common yet complex patterns at each spatio-temporal step, the input data $x_{j,t}$ is supposed to be predictable from the contextual representations $h_{j,t-1}$ and $h_{j-1,t}$.

This predictability inspired us to add new mechanism to ST-LSTM to predict the input and compare it with the actual incoming input. The amount of the estimation error is used as input to a new "trust gate". The derived trust value provides information to the long-term memory mechanism to learn better decisions about when and how to remember and forget the contents of the memory cell. For example, when the trust gate finds out the current joint has wrong 3D measurements, it can block the input gate and prevent the memory cell from updating based on current unreliable input.

Mathematically, for an input at step (j,t), we develop a function to generate its prediction, based on the available contextual information:

$$p_{j,t} = \tanh\left(M_p \begin{pmatrix} h_{j-1,t} \\ h_{j,t-1} \end{pmatrix}\right) \tag{7}$$

where the affine transformation M_p maps the data from $\Re^{2d}$ to $\Re^d$, so the dimensionality of $p_{j,t}$ is d. It is worth noting that the contextual information at each step is not limited to the hidden states of the previous spatial step but it also includes the previous temporal step, i.e., the long-term memory information of the same joint in previous frames and the contextual information of other visited

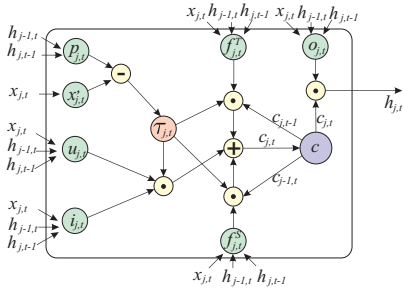

Fig. 4. Schema of the proposed ST-LSTM with trust gate.

joints in the same frame are seamlessly incorporated. Therefore, we can expect this function to be able to produce good predictions.

The activation of the proposed trust gate τ is a vector in $\Re^d$, which is similar to the activation of the input gate and the forget gate, and it will be calculated as:

$$x'_{j,t} = \tanh\left(M_x\left(x_{j,t}\right)\right) \tag{8}$$

$$\tau_{j,t} = G(x'_{j,t} - p_{j,t}) \tag{9}$$

where $M_x : \Re^D \rightarrow \Re^d$ is an affine transformation, and the new activation function $G(\cdot)$ is an element-wise operation formulated as:

$$G(z) = \exp(-\lambda z^2) \tag{10}$$

In this equation, $\lambda > 0$ is a parameter to control the spread of the Gaussian function. $G(z)$ produces a large response if z is close to origin, and small response when z has a large absolute value.

Utilizing the proposed trust gate, the cell state of the ST-LSTM neuron can be updated as:

$$c_{j,t} = \tau_{j,t} \odot i_{j,t} \odot u_{j,t} + (\mathbf{1} - \tau_{j,t}) \odot f^S_{j,t} \odot c_{j-1,t} + (\mathbf{1} - \tau_{j,t}) \odot f^T_{j,t} \odot c_{j,t-1} \tag{11}$$

If the new input $x_{j,t}$ cannot be trusted (because of noise or occlusion), then we need to take advantage of more history information and try to block the new input. In contrast, if the input is reliable, we can let the learning algorithm update the memory cell by importing input information.

Fig. 4 depicts the scheme of the new ST-LSTM unit empowered with the trust gate. This can be learned similar to other gates by back-propagation. The proposed trust gate technique is theoretically general and can be applied to other applications to deal with unreliable input data.

3.5 Learning the Classifier

Since the action labels are always given at the video level, we feed them as the training outputs of the ST-LSTM at each spatio-temporal step. The network

learns to predict the action class $\hat{y}$ among a discrete set of classes Y using a softmax layer. The overall prediction of a video is computed by averaging the predictions of all the steps. Empirically, this method provides better performance compared to the minimization of the loss at the last step only.

The objective function of our model is formulated as:

$$L = \sum_{j=1}^{J} \sum_{t=1}^{T} l(\hat{y}_{j,t}, y) \tag{12}$$

where $l(\hat{y}_{j,t}, y)$ is the negative log-likelihood loss [54] measuring the difference between the true label y and the predicted result $\hat{y}_{j,t}$ at step (j, t). The objective function can be minimized using back-propagation through time (BPTT) algorithm [54].

4 Experiments

The proposed model is evaluated on four datasets: NTU RGB+D dataset, SBU Interaction dataset, UT-Kinect dataset, and Berkeley MHAD dataset. We conduct extensive experiments with different configurations as follows:

(1) "ST-LSTM (Joint Chain)": In this configuration, the joints are visited one by one in a simple chain order (see Fig. 2(a)).

(2) "ST-LSTM (Tree Traversal)": The proposed tree traversal strategy (Fig. 2(c)) is adopted in this configuration to fully exploit the tree-based spatial structure of human joints.

(3) "ST-LSTM (Tree Traversal) + Trust Gate": This configuration involves the trust gate to deal with noisy input.

4.1 Evaluation datasets

NTU RGB+D Dataset [32]. To the best of our knowledge, this dataset is currently the largest depth-based action recognition dataset. It is collected by Kincet v2 and contains more than 56 thousand sequences and 4 million frames. A total of 60 different action classes including daily actions, pair actions, and medical conditions are performed by 40 subjects aged between 10 and 35. The 3D coordinates of 25 joints are provided in this dataset. The large intra-class and view point variations make this dataset very challenging. Due to the large amount of samples, this dataset is highly suitable for deep learning based action recognition.

SBU Interaction Dataset [59]. This dataset is captured with Kinect and contains 8 classes of two-person interactions. It includes 282 skeleton sequences in 6822 frames. Each skeleton has 15 joints. The challenges of this dataset include: (1) in most interactions, one person is acting and the other one is reacting; and (2) the joint coordinates in many sequences are of low accuracy.

UT-Kinect Dataset [5]. This dataset contains 10 action classes performed by 10 subjects, captured with a stationary Kinect. Each action was performed

twice by every subject. The locations of 20 joints are provided in this dataset. The high intra-class variation and viewpoint diversity makes it challenging.

Berkeley MHAD [60]. The MHAD dataset is captured by a motion capture system. It consists of 659 sequences and about 82 minutes of recording. Eleven different action classes were performed by 7 male and 5 female subjects. The 3D locations of 35 joints are provided in this dataset.

4.2 Implementation details

In our experiments, each video sequence is divided to T sub-sequences with the same length, and one frame was randomly selected from each sub-sequence. Such a method adds randomness into the process of data generation and improves the generalization capability. We observe this strategy achieves better performance in contrast to uniformly sampled frames. We cross-validated the performance based on leave-one-subject-out protocol on NTU RGB+D dataset, and found $T = 20$ as the optimum value.

We use Torch toolbox as the deep learning platform and an NVIDIA Tesla K40 GPU to run our experiments. We train the network using stochastic gradient descent, and set learning rate, momentum and decay rate as 2×10^{-3}, 0.9 and 0.95, respectively. For our network, we set the neuron size d to 128, and the parameter λ used in $G(\cdot)$ to 0.5. We use two ST-LSTM layers in the stacked network, and the applied probability of dropout is 0.5. Though there are variations in terms of sequence length, joint number, and data acquisition equipment for different datasets, we use the same parameter settings mentioned above. This indicates the insensitiveness of our method to the parameter settings, as it achieves promising results on all the datasets with the same configuration.

4.3 Experimental results

NTU RGB+D Dataset. This dataset has two standard evaluation protocols [32]. One is cross-subject evaluation, for which half of the subjects are used for training and the remaining are for testing. The second is cross-view evaluation, for which two viewpoints are used for training and one is left out for testing.

Table 1. Experimental results (accuracies) on NTU RGB+D Dataset

Method	Cross subject	Cross view
Lie Group [7]	50.1%	52.8%
Skeletal Quads [6]	38.6%	41.4%
Dynamic Skeletons [61]	60.2%	65.2%
HBRNN [30]	59.1%	64.0%
Part-aware LSTM [32]	62.9%	70.3%
Deep RNN [32]	56.3%	64.1%
Deep LSTM [32]	60.7%	67.3%
ST-LSTM (Joint Chain)	61.7%	75.5%
ST-LSTM (Tree Traversal)	65.2%	76.1%
ST-LSTM (Tree Traversal) + Trust Gate	**69.2%**	**77.7%**

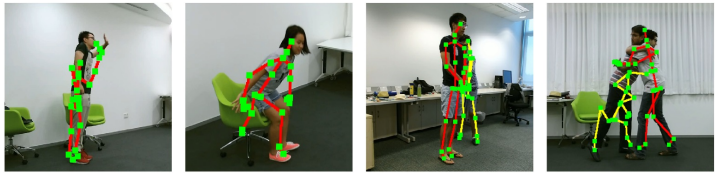

Fig. 5. Example images with noisy skeletons from NTU RGB+D dataset.

The results are shown in Table 1. Deep RNN and deep LSTM models concatenate the joints features at each frame and then feed them to the network to model the temporal dynamics and ignore the spatial dynamics. As can be seen, both "ST-LSTM (Joint Chain)" and "ST-LSTM (Tree Traversal)" models outperform these methods by a notable margin.

It can also be observed that the trust gate brings significant performance improvement, because the data acquired by Kinect is noisy and some joints are frequently occluded in this dataset.

A notable portion of samples of this dataset are captured from side view, and based on the design of Kinect's body tracking mechanism, side view skeletal data is less accurate than the front view. To further show the effectiveness of trust gate, we analyze the performance using only the samples in side views. When using "ST-LSTM (Tree Traversal)", the accuracy is 76.5%, while "ST-LSTM (Tree Traversal) + Trust Gate" achieves 81.6%. This indicates the proposed trust gate can effectively handle severely noisy data.

To verify the effectiveness of layer stacking, we decrease the network size by using only one ST-LSTM layer, and the accuracies drop to 65.5% (cross-subject) and 77.0% (cross-view). It indicates our two-layer stacked model has better representation strengths than a single-layer model.

The sensitivity of the proposed model to neural unit sizes and λ values are also evaluated and the results are depicted in Fig. 6. When trust gate is used, our model achieves better performance for all the λ values tested compared to the model without trust gate.

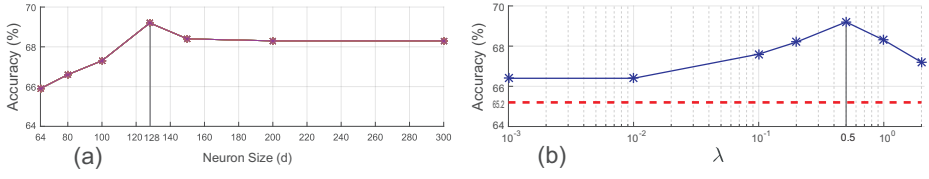

Fig. 6. (a) Comparison of the performance for different neuron size (d) values on NTU RGB+D dataset (cross-subject). (b) Comparison of different λ values on NTU RGB+D dataset (cross-subject). The blue line indicates the results when different λ values are used for trust gate, and the red dashed line indicates the performance when trust gate is not added.

Table 2. Experimental results on SBU Interaction Dataset

Method	Accuracy
Yun et al., [59]	80.3%
Ji et al., [63]	86.9%
CHARM [64]	83.9%
HBRNN [30] (reported by [51])	80.4%
Co-occurrence LSTM [51]	90.4%
Deep LSTM (reported by [51])	86.0%
ST-LSTM (Joint Chain)	84.7%
ST-LSTM (Tree)	88.6%
ST-LSTM (Tree) + Trust Gate	**93.3%**

Table 3. Results on UT-Kinect Dataset (leave-one-out-cross-validation protocol [5])

Method	Accuracy
Histogram of 3D Joints [5]	90.9%
Grassmann Manifold [65]	88.5%
Riemannian Manifold [66]	91.5%
ST-LSTM (Joint Chain)	91.0%
ST-LSTM (Tree)	92.4%
ST-LSTM (Tree) + Trust Gate	**97.0%**

Finally, we evaluate the classification performance on early stopping conditions by feeding the first p $(0 < p < 1)$ portion of the testing video to the trained network on cross-subject protocol. When setting p as $0.1, 0.2, ..., 1.0$, the corresponding accuracies are 13.4%, 21.6%, 33.9%, 46.6%, 55.5%, 61.1%, 64.6%, 66.7%, 68.2%, 69.2%, respectively. We can find that the results improve when a larger portion of video is fed.

SBU Interaction Dataset. We follow the standard experimental protocol of [59] and perform 5-fold cross validation on SBU Interaction Dataset. In this dataset, two human skeletons are provided in each frame, so our traversal visits the joints throughout the two skeletons over the spatial steps. We summarize the results in terms of average classification accuracy in Table 2. In the table, [51] and [30] are both LSTM-based methods, which are more relevant to our model.

As can be seen, the proposed "ST-LSTM (Tree Traversal) + Trust Gate" model outperforms all other skeleton-based methods. "ST-LSTM (Tree Traversal)" yields higher accuracy than "ST-LSTM (Joint Chain)", as the latter adds some unreasonable links between the less related joints.

It is worth noting that deep LSTM [51], Co-occurrence LSTM [51], and HBRNN [30] all use the Svaitzky-Golay filter in temporal domain to smooth the skeleton joint positions to reduce the influence of the noise in the data captured by Kinect. However, even without trust gate (which aims at handling noisy input), the "ST-LSTM (Tree Traversal)" model outperforms HBRNN and deep LSTM, and achieves comparable result (88.6%) to Co-occurrence LSTM. Once the trust gate is utilized, the accuracy jumps to 93.3%. We do not adopt any skeleton normalization operation, such as translation or rotation of the skeleton [7], and achieve state-of-the-art performance. We notice that [62] obtained very similar result (93.4%) on SBU dataset. However, their method utilized both RGB and depth images, while our method just uses the skeleton data.

UT-Kinect Dataset. There are two popular protocols on UT-Kinect dataset. First is the leave-one-out-cross-validation protocol [5]. Second is proposed in [67], for which half of the subjects are used for training and the remaining are used for testing. We use both protocols to evaluate the proposed method more extensively.

On the first protocol, our model achieves superior performance over other skeleton-based methods by a large margin, as shown in Table 3. On the second

evaluation protocol (Table 4), our model achieves competitive result (95.0%) to Elastic functional coding [68] (94.9%), which is an extension of the Lie Group model [7].

Berkeley MHAD. We follow the protocol in [30] on MHAD dataset, in which 384 sequences corresponding to the first 7 subjects are used for training and the 275 sequences of the remaining 5 subjects are used for testing. The results are shown in Table 5. Our method achieves the accuracy of 100% without preliminary smoothing operations, which are adopted in [30].

Besides, we have tested our model on **MSR Action3D dataset** [69] following the protocol in [30], and achieved an accuracy of 94.8%, which is slightly superior to 94.5% achieved by HBRNN [30].

4.4 Effectiveness of Trust Gate

To better study the effectiveness of the trust gate in the proposed network model, we specifically evaluate noisy samples from MSR Action3D dataset. We manually rectify some noisy joints of these samples by referring to the corresponding depth maps, and compared the activations of the trust gates on noisy and rectified inputs. As shown in Fig. 7(a), the activation of the trust gate is smaller when a noisy joint is fed, compared to the corresponding rectified joint. This shows how the network reduces the impact of the noisy input data.

To comprehensively evaluate the trust gate, we also manually add noise to one joint for all testing samples on MHAD dataset. Note that MHAD dataset was captured with motion capture system, thus the skeleton joints are much more accurate than those collected by Kinect. We add noise to the right foot joint by moving the joint away from the original position. The direction of the translation vector is randomly chosen and the norm is also a random value around 30cm (this is a significant noise in the scale of human bodies). For each video, we add noise to the same joint at the same time step, and then analyze the effect in average.

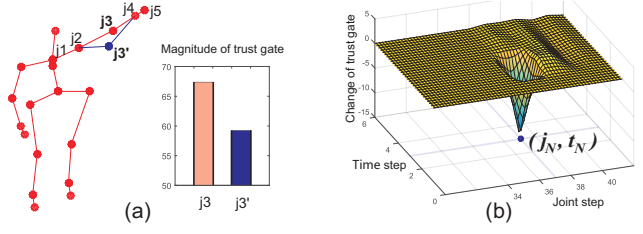

Fig. 7. Behavior of trust gate when inputting noisy data. (a) $j_{3'}$ is a noisy joint location, and j_3 is the corresponding rectified joint position. In the histogram, the blue bar is the magnitude of the trust gate when inputting the noisy joint $j_{3'}$. The red bar is the magnitude of the corresponding trust gate when $j_{3'}$ is rectified to j_3. (b) The difference between the trust gate calculated when inputting the original data and that calculated when the noise is imposed at the j_N-th spatial step and t_N-th time step.

Table 4. Experimental results on UT-Kinect Dataset (half-vs-half protocol [67])

Method	Accuracy
Skeleton Joint Features [67]	87.9%
Lie Group [7] (reported by [68])	93.6%
Elastic functional coding [68]	94.9%
ST-LSTM (Tree) + Trust Gate	**95.0%**

Table 5. Experimental results on MHAD Dataset

Method	Accuracy
Vantigodi et al. [70]	96.1%
Ofli et al. [71]	95.4%
Vantigodi et al. [72]	97.6%
Kapsouras et al. [73]	98.2%
HBRNN [30]	**100%**
ST-LSTM (Tree) + Trust Gate	**100%**

We measure the difference in the magnitude of the trust gate activations between the original data and the noisy ones. For all the testing samples, we perform the same procedure, then calculate the average difference. The result is depicted in Fig. 7(b). We can see when the noisy data is fed to the network, the magnitude of the trust gate is reduced. This shows how the network ignores the noisy input, and tries to prevent it from affecting the network. In this experiment, we observe the overall accuracy does not drop after adding the noise.

5 Conclusion

In this paper we propose to extend the RNN-based 3D action recognition to spatio-temporal domain. A new ST-LSTM network is introduced which analyses the 3D location of each individual joint in each video frame, at each processing step. For better representation of the structured input to the network, a skeleton tree traversal algorithm is proposed which takes the adjacency graph of body joints into account and improves the performance of the network by arranging the most related joints together in the input sequence. Due to the unreliability of the 3D input data, a new gating mechanism is also proposed to improve the robustness of the network against noise and occlusion. The provided experimental results validate the proposed contributions and prove the effectiveness of our method by achieving superior performance over the existing state-of-the-art methods on four evaluated datasets.

Acknowledgement. The research is supported by Singapore Ministry of Education (MOE) Tier 2 ARC28/14, and Singapore A*STAR Science and Engineering Research Council PSF1321202099. This research was carried out at the Rapid-Rich Object Search (ROSE) Lab at Nanyang Technological University. The ROSE Lab is supported by the National Research Foundation, Singapore, under its Interactive Digital Media (IDM) Strategic Research Programme. We also would like to thank NVIDIA for the GPU donation.

References

1. Presti, L.L., La Cascia, M.: 3d skeleton-based human action classification: A survey. PR (2016)

2. Han, F., Reily, B., Hoff, W., Zhang, H.: Space-time representation of people based on 3d skeletal data: a review. arXiv (2016)
3. Zhu, F., Shao, L., Xie, J., Fang, Y.: From handcrafted to learned representations for human action recognition: a survey. IVC (2016)
4. Yang, X., Tian, Y.: Effective 3d action recognition using eigenjoints. JVCIR (2014)
5. Xia, L., Chen, C., Aggarwal, J.: View invariant human action recognition using histograms of 3d joints. In: CVPRW. (2012)
6. Evangelidis, G., Singh, G., Horaud, R.: Skeletal quads: Human action recognition using joint quadruples. In: ICPR. (2014)
7. Vemulapalli, R., Arrate, F., Chellappa, R.: Human action recognition by representing 3d skeletons as points in a lie group. In: CVPR. (2014)
8. Luo, J., Wang, W., Qi, H.: Group sparsity and geometry constrained dictionary learning for action recognition from depth maps. In: ICCV. (2013)
9. Ohn-Bar, E., Trivedi, M.: Joint angles similarities and hog^2 for action recognition. In: CVPRW. (2013)
10. Mikolov, T., Kombrink, S., Burget, L., Černockỳ, J.H., Khudanpur, S.: Extensions of recurrent neural network language model. In: ICASSP. (2011)
11. Sundermeyer, M., Schlüter, R., Ney, H.: Lstm neural networks for language modeling. In: INTERSPEECH. (2012)
12. Mesnil, G., He, X., Deng, L., Bengio, Y.: Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In: INTERSPEECH. (2013)
13. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: CVPR. (2015)
14. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: ICML. (2015)
15. Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: Deep networks for video classification. In: CVPR. (2015)
16. Srivastava, N., Mansimov, E., Salakhudinov, R.: Unsupervised learning of video representations using lstms. In: ICML. (2015)
17. Singh, B., Marks, T.K., Jones, M., Tuzel, O., Shao, M.: A multi-stream bi-directional recurrent neural network for fine-grained action detection. In: CVPR. (2016)
18. Jain, A., Zamir, A.R., Savarese, S., Saxena, A.: Structural-rnn: Deep learning on spatio-temporal graphs. In: CVPR. (2016)
19. Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L., Savarese, S.: Social lstm: Human trajectory prediction in crowded spaces. In: CVPR. (2016)
20. Deng, Z., Vahdat, A., Hu, H., Mori, G.: Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition. In: CVPR. (2016)
21. Ibrahim, M.S., Muralidharan, S., Deng, Z., Vahdat, A., Mori, G.: A hierarchical deep temporal model for group activity recognition. In: CVPR. (2016)
22. Ma, S., Sigal, L., Sclaroff, S.: Learning activity progression in lstms for activity detection and early detection. In: CVPR. (2016)
23. Ni, B., Yang, X., Gao, S.: Progressively parsing interactional objects for fine grained action detection. In: CVPR. (2016)
24. Li, Y., Lan, C., Xing, J., Zeng, W., Yuan, C., Liu, J.: Online human action detection using joint classification-regression recurrent neural networks. arXiv (2016)

25. Varior, R.R., Shuai, B., Lu, J., Xu, D., Wang, G.: A siamese long short-term memory architecture for human re-identification. In: ECCV. (2016)
26. Varior, R.R., Haloi, M., Wang, G.: Gated siamese convolutional neural network architecture for human re-identification. In: ECCV. (2016)
27. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: CVPR. (2015)
28. Li, Q., Qiu, Z., Yao, T., Mei, T., Rui, Y., Luo, J.: Action recognition by learning deep multi-granular spatio-temporal video representation. In: ICMR. (2016)
29. Wu, Z., Wang, X., Jiang, Y.G., Ye, H., Xue, X.: Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In: ACM MM. (2015)
30. Du, Y., Wang, W., Wang, L.: Hierarchical recurrent neural network for skeleton based action recognition. In: CVPR. (2015)
31. Veeriah, V., Zhuang, N., Qi, G.J.: Differential recurrent neural networks for action recognition. In: ICCV. (2015)
32. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: Ntu rgb+d: A large scale dataset for 3d human activity analysis. In: CVPR. (2016)
33. Wang, J., Liu, Z., Wu, Y., Yuan, J.: Learning actionlet ensemble for 3d human action recognition. TPAMI (2014)
34. Meng, M., Drira, H., Daoudi, M., Boonaert, J.: Human-object interaction recognition by learning the distances between the object and the skeleton joints. In: FG. (2015)
35. Shahroudy, A., Ng, T.T., Yang, Q., Wang, G.: Multimodal multipart learning for action recognition in depth videos. TPAMI (2016)
36. Wang, J., Wu, Y.: Learning maximum margin temporal warping for action recognition. In: ICCV. (2013)
37. Rahmani, H., Mahmood, A., Huynh, D.Q., Mian, A.: Real time action recognition using histograms of depth gradients and random decision forests. In: WACV. (2014)
38. Shahroudy, A., Wang, G., Ng, T.T.: Multi-modal feature fusion for action recognition in rgb-d sequences. In: ISCCSP. (2014)
39. Wang, C., Wang, Y., Yuille, A.L.: Mining 3d key-pose-motifs for action recognition. In: CVPR. (2016)
40. Rahmani, H., Mian, A.: Learning a non-linear knowledge transfer model for cross-view action recognition. In: CVPR. (2015)
41. Lillo, I., Carlos Niebles, J., Soto, A.: A hierarchical pose-based approach to complex action understanding using dictionaries of actionlets and motion poselets. In: CVPR. (2016)
42. Hu, J.F., Zheng, W.S., Ma, L., Wang, G., Lai, J.: Real-time rgb-d activity prediction by soft regression. In: ECCV. (2016)
43. Chen, C., Jafari, R., Kehtarnavaz, N.: Fusion of depth, skeleton, and inertial data for human action recognition. In: ICASSP. (2016)
44. Rahmani, H., Mian, A.: 3d action recognition from novel viewpoints. In: CVPR. (2016)
45. Liu, Z., Zhang, C., Tian, Y.: 3d-based deep convolutional neural network for action recognition with depth sequences. IVC (2016)
46. Cai, X., Zhou, W., Wu, L., Luo, J., Li, H.: Effective active skeleton representation for low latency human action recognition. TMM (2016)
47. Al Alwani, A.S., Chahir, Y.: Spatiotemporal representation of 3d skeleton joints-based action recognition using modified spherical harmonics. PR Letters (2016)

48. Tao, L., Vidal, R.: Moving poselets: A discriminative and interpretable skeletal motion representation for action recognition. In: ICCVW. (2015)
49. Shahroudy, A., Ng, T.T., Gong, Y., Wang, G.: Deep multimodal feature analysis for action recognition in rgb+d videos. arXiv (2016)
50. Du, Y., Fu, Y., Wang, L.: Representation learning of temporal dynamics for skeleton-based action recognition. TIP (2016)
51. Zhu, W., Lan, C., Xing, J., Zeng, W., Li, Y., Shen, L., Xie, X.: Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. In: AAAI. (2016)
52. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. JMLR (2014)
53. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation (1997)
54. Graves, A. In: Supervised Sequence Labelling with Recurrent Neural Networks. Springer (2012)
55. Zou, B., Chen, S., Shi, C., Providence, U.M.: Automatic reconstruction of 3d human motion pose from uncalibrated monocular video sequences based on markerless human motion tracking. PR (2009)
56. Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: CVPR. (2011)
57. Graves, A., Mohamed, A.r., Hinton, G.: Speech recognition with deep recurrent neural networks. In: ICASSP. (2013)
58. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: NIPS. (2014)
59. Yun, K., Honorio, J., Chattopadhyay, D., Berg, T.L., Samaras, D.: Two-person interaction detection using body-pose features and multiple instance learning. In: CVPRW. (2012)
60. Ofli, F., Chaudhry, R., Kurillo, G., Vidal, R., Bajcsy, R.: Berkeley mhad: A comprehensive multimodal human action database. In: WACV. (2013)
61. Hu, J.F., Zheng, W.S., Lai, J., Zhang, J.: Jointly learning heterogeneous features for rgb-d activity recognition. In: CVPR. (2015)
62. Lin, L., Wang, K., Zuo, W., Wang, M., Luo, J., Zhang, L.: A deep structured model with radius–margin bound for 3d human activity recognition. IJCV (2015)
63. Ji, Y., Ye, G., Cheng, H.: Interactive body part contrast mining for human interaction recognition. In: ICMEW. (2014)
64. Li, W., Wen, L., Choo Chuah, M., Lyu, S.: Category-blind human action recognition: a practical recognition system. In: ICCV. (2015)
65. Slama, R., Wannous, H., Daoudi, M., Srivastava, A.: Accurate 3d action recognition using learning on the grassmann manifold. PR (2015)
66. Devanne, M., Wannous, H., Berretti, S., Pala, P., Daoudi, M., Del Bimbo, A.: 3-d human action recognition by shape analysis of motion trajectories on riemannian manifold. IEEE Transactions on Cybernetics (2015)
67. Zhu, Y., Chen, W., Guo, G.: Fusing spatiotemporal features and joints for 3d action recognition. In: CVPRW. (2013)
68. Anirudh, R., Turaga, P., Su, J., Srivastava, A.: Elastic functional coding of human actions: from vector-fields to latent variables. In: CVPR. (2015)
69. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3d points. In: CVPRW. (2010)
70. Vantigodi, S., Babu, R.V.: Real-time human action recognition from motion capture data. In: NCVPRIPG. (2013)

71. Ofli, F., Chaudhry, R., Kurillo, G., Vidal, R., Bajcsy, R.: Sequence of the most informative joints (smij): A new representation for human skeletal action recognition. JVCIR (2014)
72. Vantigodi, S., Radhakrishnan, V.B.: Action recognition from motion capture data using meta-cognitive rbf network classifier. In: ISSNIP. (2014)
73. Kapsouras, I., Nikolaidis, N.: Action recognition on motion capture data using a dynemes and forward differences representation. JVCIR (2014)