JAIST Repository
https://dspace.jaist.ac.jp/

Title	A Study on Learning Interaction Relationships for Indoor Human Activity with RGB-D Video
Author(s)	Troung, Minh Anh
Citation	
Issue Date	2018-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/15247
Rights	
Description	Supervisor:吉高　淳夫，情報科学研究科，修士

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

A Study on Learning Interaction Relationships for Indoor Human Activity with RGB-D Video

TRUONG Minh Anh

School of Information Science
Japan Advanced Institute of Science and Technology
(Degree conferment March, 2018)

Master's Thesis

A Study on Learning Interaction Relationships for Indoor Human Activity with RGB-D Video

1510218 TRUONG Minh Anh

Supervisor:	Atsuo Yoshitaka
Main Examiner:	Masato Akagi
Examiners:	Masato Akagi
	Jianwu Dang
	Atsuo Yoshitaka

A Study on Learning Interaction Relationships for Indoor Human Activity with RGB-D Video

TRUONG Minh Anh (1510218)

School of Information Science,
Japan Advanced Institute of Science and Technology

March 14, 2017

Keywords: Activity Recognition, Human-human interactions, Human-object interactions, Recurrent neural networks.

Understanding human activity has been an important research area in computer vision. Generally, we can model the human interactions as a temporal sequence with the transition in relationships of humans and objects. On the other hand, many studies have proved the effectiveness of Long Short-Term Memory (LSTM) networks in long-term temporal dependency problems. In this study, the author proposed a novel Structured Recurrent Neural Network (S-RNN) to model spatio-temporal relationships between human subjects and objects in daily human interactions. The author represent the evolution of different components as well as the relationships between them over time by several LSTM subnets. Then, the hidden representations of those relations are fused into the later layers to obtain the final hidden representation. The prediction is carried out by the single-layer perceptron. The experimental results of different tasks on the CAD-120, SBU-Kinect-Interaction, and Multi-modal & Multi-view & Interactive (M2I) datasets showed advantages of the proposed method compared with the state-of-art methods.

Copyright © 2018 by TRUONG Minh Anh

Contents

"This dissertation was prepared according to the curriculum for the Collaborative Education Program organized by Japan Advanced Institute of Science and Technology and VNU-HCM University."

List of Figures

List of Tables

Acronyms

BPTT Backpropagation Through Time. 19

BSC Body Surface Context. 10

CNNs Convolutional Neural Networks. ii, 13

CRF Conditional Random Fields. 16

DCNN Deep Convolutional Neural Network. 13

DMM Depth Motion Maps. v, 9, 13, 14

DMM-HOG Depth Motion Maps-Histogram of Oriented Gradients. v, 9

DTW Dynamic Time Warping. 11

FTP Fourier Temporal Pyramid. 11

HOG Histogram of Oriented Gradients. 9

HOJ3D Histograms of 3D Joint Locations. 11

HON4D Histogram of Oriented 4D Normals. 10

HOPC Histogram of Oriented Principal Components. 10

LSTM Long Short-Term Memory. i, iii, 12, 20–22, 24, 25, 32, 34, 39, 53, 54

M2I Multi-modal & Multi-view & Interactive. i, v–vii, 3, 28, 30, 41–48, 54, 57–60, 62

MEMM Maximum Entropy Markov Model. 15

MHI Motion History Images. v, 13, 14

MRF Markov Random Field. 16

RNN Recurrent Neural Network. v, 17, 18, 20

RNNs Recurrent Neural Networks. iii, 2, 7, 12, 17–21, 61, 62

ROP Random Occupancy Pattern. 8

RTRL Real Time Recurrent Learning. 19

S-RNN Structured Recurrent Neural Network. i, iii, v, vi, 6, 7, 22, 24–27, 33–35, 38, 39, 43, 49, 53, 54, 56, 57, 61, 62

SSVM Structural Support Vector Machine. 16

STIPs Spatio-Temporal Interest Points. 10

TPM Temporal Pyramid Matching. 11

Chapter 1

Introduction

1.1 Motivations

During the past decade, human activity recognition draws a lot of attention due to its wide range of potential applications such as human-machine interaction and automatic video surveillance. Thus, there are studies which have been published to recognize human activity from conventional RGB cameras [3]. Recently, the emergence of low-cost range camera has simplified some tasks related to human activity recognition, especially 3D body silhouette estimation and human posture estimation. It allowed recognition methods based on 3D information and human body shape to be developed [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14].

Most of the previous methods recognize a human activity based on low-level hand-crafted features or skeleton-based features without taking the relationships among human subjects and objects into account. Hence, this kind of methods is not able to work well with complex human activity such as human-object interaction or human-human interaction. For instance, human separate stacking object or unstacking object activities based on the way the objects are arranged, not based on any specific human poses. Thus, the relationship among human subjects and objects has an important role for human interaction recognition. Besides, this traditional approach does not consider human activity localization problem. Hence, we could not apply this approach to applications involving human-robot interactions because the robots have to know the location of the activity to react.

To address these issues, most of the existing methods recognize human activities from sophisticated spatio-temporal relationship representation

models [7, 8, 9, 10, 11, 12, 13]. However, these methods are sensitive to the parameter settings or lack of discriminative power due to different assumptions of modeling methods. It makes these methods work well only on a few number of tasks and problems.

Additionally, most of existing human activity recognition methods assume that the input video consists of only one human activity. So, these methods classify an input video into its activity category instead of recognizing human activity. Therefore, in real life application, we have to apply a video segmentation method to obtain the input video for these methods. Thus, the errors and execution time of segmentation method could reduce the effectiveness of these methods.

In this work, the author focus on the problem of modeling the spatio-temporal relationships which could perform well on different tasks based on Recurrent Neural Networks (RNNs).

1.2 Applications

The main goal of human activity recognition is to understand what human subjects did in the observed video. Human activity recognition has numerous types of applications on different domains, such as entertainment, education, security, and surveillance. In this study, the author focus on applications that have videos captured in indoor environments such as human activity surveillance, human-robot interaction, and content-based analysis.

1.3 Problem statement

Normally, human activities are classified depending on the purposes of applications. However, human activities can be philosophically categorized into four levels with regard to their complexity, namely gestures, actions, interactions and group activities [3].

- Gestures are meaningful and elementary movements of a human body part (*e.g.* Raising an arm or a leg).

- Actions are defined as single-person activities, which contains multiple gestures which are associated by the temporal constraints. The typical examples for this kind of human activity are "walking,", "jogging", "running", and "hand waving."
- Interactions are human activities which involve multiple persons and/or objects.
- Group activities are the activities executed by conceptual groups which consist of multiple persons and/or objects.

In this study, the author focus on the problem of recognizing human interactions in indoor environments. The author also consider the problem of modeling spatio-temporal relationships among human subject and objects for recognizing both human-object and human-human interactions in daily life. To simplify the problem of recognizing human-human interaction, the author focus on human-human interactions that consist of two human subjects. For human-object interaction, the author evaluate the proposed method on a challenging human-object interaction dataset consisting of 120 videos with different high-level activities, sub-activities and object affordances. For human-human interaction, the author evaluate the proposed method on two benchmark datasets, namely SBU Kinect Interaction dataset and M2I dataset.

1.4 Challenges

Although a tremendous amount of work has been done on human interaction recognition, this field remains immature due to its challenges. Thus, before discussing this work, the author would like to describe the research challenges as follows.

1.4.1 Semantic gap

As mentioned in Section 1.3, human interaction could involve two or more components (human or object). Therefore, the locations of these components, as well as their states are really important for recognition task. Unfortunately, a computer is only able to recognize the colors of pixels

but could not recognize objects like human or apples. Thus, we have to develop methods to improve the level of understanding of computer for dealing with complex problems like human activity recognition. In other words, we have to apply different preprocessing methods such as object detection, human pose estimation. However, there is no complete solution for this purpose. For example, the performance of the state-of-the-art object detection methods on COCO dataset (COCO is a large-scale object detection benchmark dataset, which consists of more than $300,000$ images) is just around 60% [15].

Unfortunately, the only way to deal with this problem is to improve the performance of existing preprocessing methods. Thus, the performance of human interaction recognition systems is unavoidably affected by the accuracy of these preprocessing methods.

Note that, there are methods that only extract the human-designated features to capture the changes of human subjects and objects during human activity without localizing them. However, it is hard to determine the location and interval of an activity in an input video without the exact locations of subjects and objects. Thus, we can apply these methods to applications that do not require localization task, such as video tagging.

1.4.2 Intra-class and Inter-class variations

For human activity recognition, we cannot discuss the challenges without mentioning the variations of activities.

Intra-class variations are the variations in the same activity class. It happens due to many reasons. It is obvious that different persons could have different ways to execute one human activity. Even each person could execute an activity with different speed, or different manners depending on the situations. Besides, the appearances of the human subjects could also be different due to the camera viewpoints.

Inter-class variations occur due to the similarities between different activity classes. For instance, running is quite similar to walking, except that never both the feet on the ground at the same time.

In order to improve the performance of human activity recognition systems, we have to extract the features that could discriminate inter-class

variations. However, they must have high similarity to intra-class variations.

1.4.3 Environmental parameters

There are several environmental parameters that affect the intra-class variations and preprocessing methods (*e.g.* object detection, object tracking) such as lighting conditions, cluttered backgrounds, occlusions, camera motion, and variations in viewpoint. Thus, they degrade the performance of human activity recognition system.

- Lighting conditions could cause the appearances of observed human subjects and objects to be changed. Especially in low light condition, the color patterns and shapes of objects are not able to be captured by cameras.

- Cluttered backgrounds: as mentioned in Section 1.4.1, the results of preprocessing methods for bridging the semantic gap between high-level abstraction and pixel-level are not perfect. Thus, the accuracy of these methods could be dramatically reduced due to the ambiguous information of input data. For example, object detection methods mark a background region as an interesting object. It definitely affects the final prediction of a human activity recognition methods. It also makes human-designated feature extraction methods extract the features from irrelevant regions due to the ambiguous information.

- Occlusion is an unavoidable problem in human activity recognition which makes the camera could not capture all important information. A subject (an object) could be occluded by itself or another subject (object). This problem could be solved in some applications by installing multiple cameras to observe one area from different viewpoints. However, it is not available for single-view applications.

- Camera motion: due to the movement of the camera, the background is also moved. Thus, human activity methods based on human-designated features could take the motion of irrelevant regions of the background into account. This is the main reason for reducing the effectiveness of

motion features. Camera motion also potentially creates motion blur which degrades the quality of the input data.

- Variations in viewpoint: as mentioned above, the appearances of a subject (an object) are different when we observe it from different viewpoints. Thus, it increases intra-class variations and potentially reduces the effectiveness of recognition methods.

1.5 Contributions

The goal of this study is to model the spatio-temporal relationships of different components in human interactions for human activity recognition. The main contributions are as follows:

- The author propose an end-to-end S-RNN to model the temporal and spatial relationships for recognizing both human-human interaction and human-object interaction. The experiments showed that the proposed model well-performed on different problems and datasets.
- As the accuracy improvement in anticipating and recognizing human-object interaction for CAD-120 dataset, this study also showed the importance of the object states and the relationship between objects in human-object recognition.

1.6 Thesis organization

The thesis is composed of five chapters which are organized as follows:

- **Chapter 1: Introduction**

  In this chapter, the author describe the importance and challenges of understanding human interaction. Besides, the author also present the summary of the main contribution in this study.

- **Chapter 2: Related Work**

  This chapter describes a literature review on related studies with the discussion on their advantages and drawbacks.

- **Chapter 3: Recurrent Neural Networks**

  In chapter 3, the author describe the architectures of standard Recurrent Neural Networks and Long-Short Term Memory Networks. The author also describe the training process to optimize the parameters of RNNs.

- **Chapter 4: S-RNN for Human-Object Interactions**

  In chapter 4, the author describe the architecture of the proposed structured Deep RNN with Long Short-Term Memory units to model the relationships for human-object interactions. Finally, the author evaluate the proposed method on CAD dataset.

- **Chapter 5: S-RNN for Human-Human Interactions**

  This chapter describes the extension of S-RNN for recognizing human-human interactions. The author evaluate the proposed method on two benchmark datasets, namely M2I and SBU.

- **Chapter 6: Dropout for S-RNN**

  Due to the large number of parameters of the RNNs, it is hard to train the proposed networks. Thus, the author applied dropout technique to improve the generalization of the proposed structured Deep Recurrent Neural Networks. The author evaluate the proposed method on all datasets that the author used in previous sections.

- **Chapter 7: Conclusion and Future Work**

  Finally, concluding remarks on the proposed method is given in this chapter. Furthermore, the author discuss the future improvement and extension of the proposed method.

Chapter 2

Related Work

2.1 Human activity recognition using depth maps

Li *et al.* [4] developed a method for human activity recognition based on salient postures. To obtain the shape information of the body, depth maps are projected onto three orthogonal Cartesian planes (xz-, zy-, and xy-plane). The interesting 3D points are extracted along the contours of all three projections. Then, the authors constructed a "bag-of-points" from the sampled points to represent the 3D shape of human body. Li *et al.* used a graphical model called action graph to model the temporal dynamics of human activities. However, because of the noise and occlusions of the projections from both the side view as well as from the top view, the sampled points are unreliable.

To overcome this problem, Vieira *et al.* [16] divided space and time axes into multiple space-time boxes, where each space-time box is split into a 4-dimensional grid. Note that, only the cells located on the silhouettes or moving parts of the human body could represent the temporal dynamics information of human activities. Thus, Vieira *et al.* proposed a saturation scheme to eliminate the redundant cells.

Instead of separating depth sequence into a grid with a fixed size of cells, Wang *et al.* [5] developed a method to randomly sampled 4D subvolumes with different sizes at different locations. In order to deal with noise and occlusion problems, they proposed Random Occupancy Pattern (ROP) features and encoded the most discriminative regions based on sparse coding.

Yang *et al.* [6] proposed a novel method to model motion information from depth maps. First, they projected every depth frame onto three or-

thogonal Cartesian planes. They also chose a region of interest to reduce intra-class variations. Then, for each projected map, the motion energy is computed based on the difference between two consecutive maps. Yang *et al*. stacked the motion energies of all depth frames of each projected view to generate the Depth Motion Maps (DMM). They also applied the Histogram of Oriented Gradients (HOG) [17] to each 2D projected view characterize the local appearance and shape on Depth Motion Maps (DMM). The extracted features are called Depth Motion Maps-Histogram of Oriented Gradients (DMM-HOG). However, this method only works well with a small number of video frames because the DMM becomes more complex when the number of frames is increased. Thus, it is hard to apply this method to long-term and complex human activities.

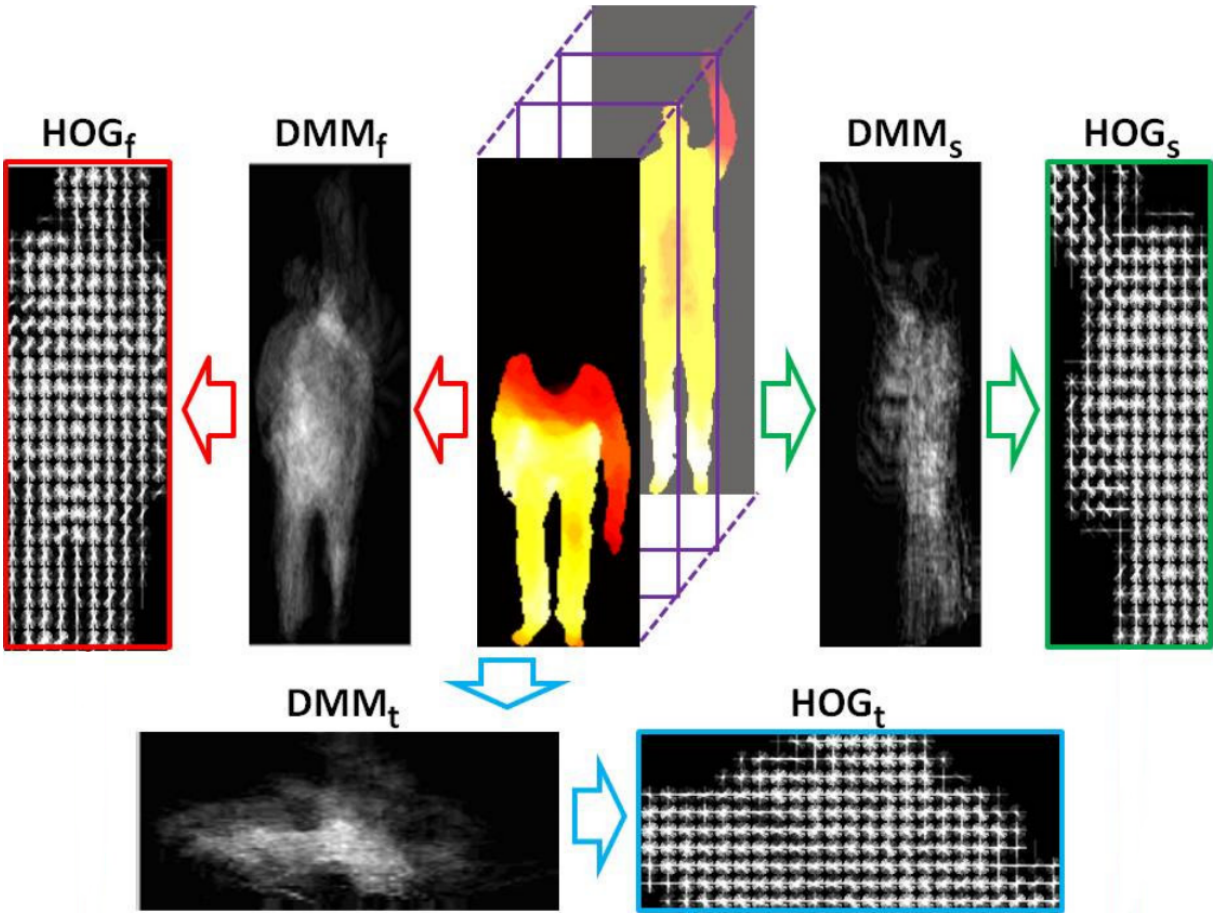


Figure 2.1: **Example Depth Motion Maps-Histogram of Oriented Gradients (DMM-HOG)** [6].

Oreifej and Liu [18] proposed a Histogram of Oriented 4D Normals (HON4D) for action recognition. The histogram represents the distribution of the surface normal orientation in the 4D volume of time, depth, and spatial coordinates of a depth sequence of human activity. Later on, to improve the robustness of against noise of the histogram of surface normals, Yang and Tian [19] proposed the concept of polynormal to extend the surface normal by grouping the 4D normals in the local neighbourhood.

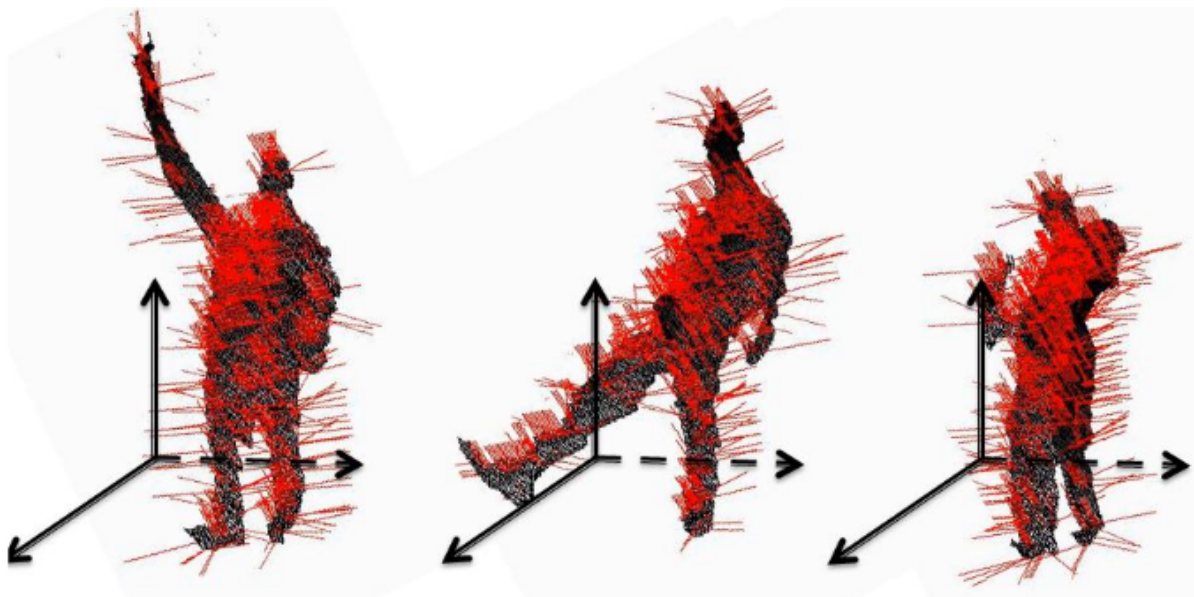


Figure 2.2: **Example surface normals computed from depth images of a human action** [18].

Xia and Aggarwal [20] proposed a method to extract reliable Spatio-Temporal Interest Points (STIPs) from depth videos for human activity recognition. The extracted key points are described by a local 3D depth cuboid based on the self-similarity.

Song *et al*. [21] proposed a Body Surface Context (BSC) feature that is extracted directly from 3D point cloud. These features describe the distribution of relative locations of 3D points which is robust to transformations including translations and rotations.

However, these approaches are sensitive to the human activity speed variations. To deal with this problem, Rahmani *et al*. [22, 23] proposed Histogram of Oriented Principal Components (HOPC) calculating the descriptor from 3D pointclouds.

2.2 Human activity recognition using skeleton joints

Despite lots of advantages, the discriminative power of the local features is limited [24]. Besides, by the introduction of cost-effective depth sensors, the human pose estimation task has been simplified. Many methods have been proposed to leverage the useful information of human poses.

Xia *et al.* [25] partitioned 3D space into n bins using a modified spherical coordinate system. Then, they used Histograms of 3D Joint Locations (HOJ3D) to represent human postures for human activity recognition. In order to make the representation more robust against noise, a joint location is voted to multiple bins by using a Gaussian weight function.

To represent the spatio-temporal relationships between different human body parts, there are methods which have proposed [26, 27, 28, 29, 30, 31, 32, 33]. In [26], Sung *et al.* extracted the body postures, hand and motion features from the skeleton data and applied a hierarchical maximum entropy Markov model to infer human activities.

Wang *et al.* [27] built the spatial and temporal dictionaries based on different human body parts to represent the spatial structure of human pose and its transition over the time.

Vemulapalli *et al.* [34] proposed the 3D geometric relationships of body parts based on rotations and translations in 3D space. To represent the transition of the skeletal representation they applied Dynamic Time Warping (DTW) and Fourier Temporal Pyramid (FTP). To handle the variations in speed and time of human activities, Luo *et al.* [1] proposed a multiple time scale dictionary learning method based on Sparse coding and Temporal Pyramid Matching (TPM).

In spite of the encouraging performance, this approach hardly distinguishes human interactions which have similar human poses (*e.g.* 'drinking' and 'answering the phone') due to lack of information from interacted objects. To overcome this problem, Wang *et al.* [33] proposed a method to extract the local appearance around skeleton joints called Local Occupancy Patterns (LOP). However, this local appearance is not able to represent the long-term spatio-temporal relationships due to the limited information of the local region. To overcome the problem of changing in viewpoint, Taha *et al.* [28] proposed a method to rotate the skeleton to the frontal

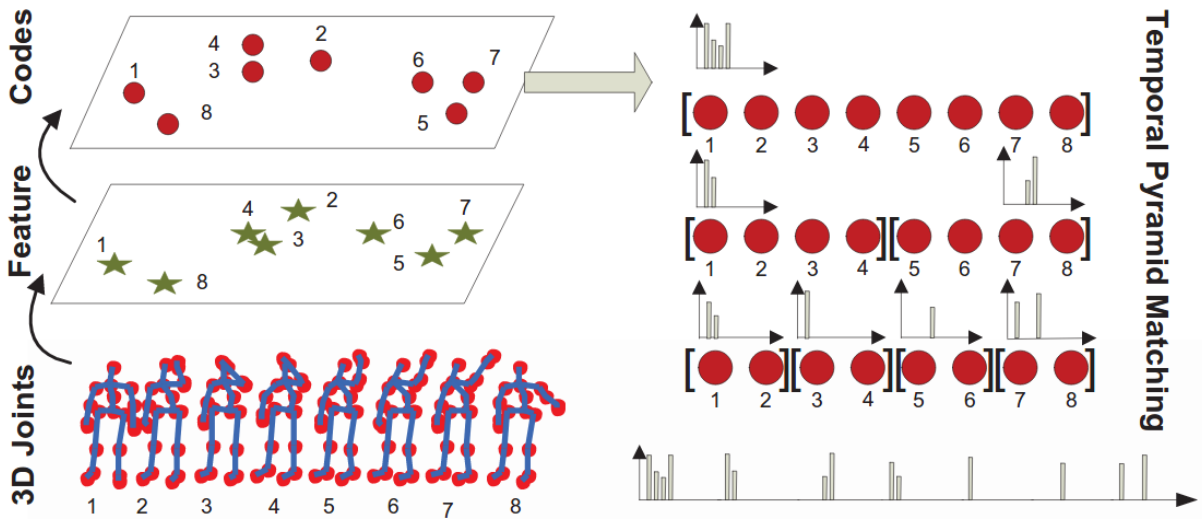


Figure 2.3: **Temporal pyramid matching proposed by Luo *et al.*** [1].

view position.

Based on the success of Deep RNN [35, 36], Du *et al.* [29] proposed a Hierarchical RNN network to explore the capability of Deep RNN for modeling the spatio-temporal relationships of the skeleton sequences. They divided the human skeleton into five groups that correspond to different physical human body parts. Then each group was fed into a separated bidirectional RNN. Then, the movements of the neighboring body parts were modeled by concatenating the representation of the trunk subnet with other subnets. The outputs of the second layer RNNs were concatenated to model the upper body and the lower body hidden representations. Similarly, the global body hidden representation was obtained in the last layer. Finally, the output of the last layer was fed into a softmax classifier for action classification.

In [30], Zhu *et al.* extended the idea of hierarchical Deep RNN for modeling the spatio-temporal relationships of the skeleton sequences by introducing a LSTM network to learn co-occurrence relations among the joints. They also derived a new in-depth dropout for LSTM based on [37] and used it to help the network learn complex motion dynamics.

Shahroudy *et al.* [31] proposed a new part-aware LSTM (P-LSTM) structure to model the long-term motion patterns of human body parts. Each P-LSTM unit consists of many LSTM sub-cells, which correspond to dif-

ferent body parts, with individual input, forget, and modulation gates. The output gate will be shared among the sub-cells. In [32], Liu *et al*. introduced a new LSTM unit with "Trust Gates" to handle the influence of noise in 3D skeleton input data for human activity recognition.

2.3 Human activity recognition using Convolutional Neural Networks (CNNs)

Recently, Convolutional Neural Networks (CNNs) have shown impressive results on image classification [38] and object detection [15, 39]. Some efforts have been made to apply deep neural networks to recognize human activity.

Wang *et al*. [40] considered one complex human activity as a sequence of separated actions. Each separated action is equivalent to a cubic-like video segment with unfixed-length. A spatio-temporal convolutional neural network with multiple subnets was proposed, where each subnet corresponds to a video segment.

In [41], Du *et al*. represented a skeleton sequence as a matrix by concatenating the vector of joint locations of every timestep in chronological order. The matrix is quantified into an interval $[0, 255]$ to make an integral image representation. For classification, the image is fed into a Deep Convolutional Neural Network (DCNN). In [2], Imran *et al*. generated Motion History Images (MHI) from RGB videos and DMM from depth sequences. Then, 4 CNNs are applied for extracting features and classification.

By leveraging a large amount of synthetic data, Rahmani *et al*. [24] proposed a view-invariant hidden representation of human pose based on Deep Convolutional Neural Network (DCNN).

2.4 Human activity recognition using spatio-temporal relationship

In this section, the author review the literature on human interaction recognition that closely relates to our work. In particular, we focus on the

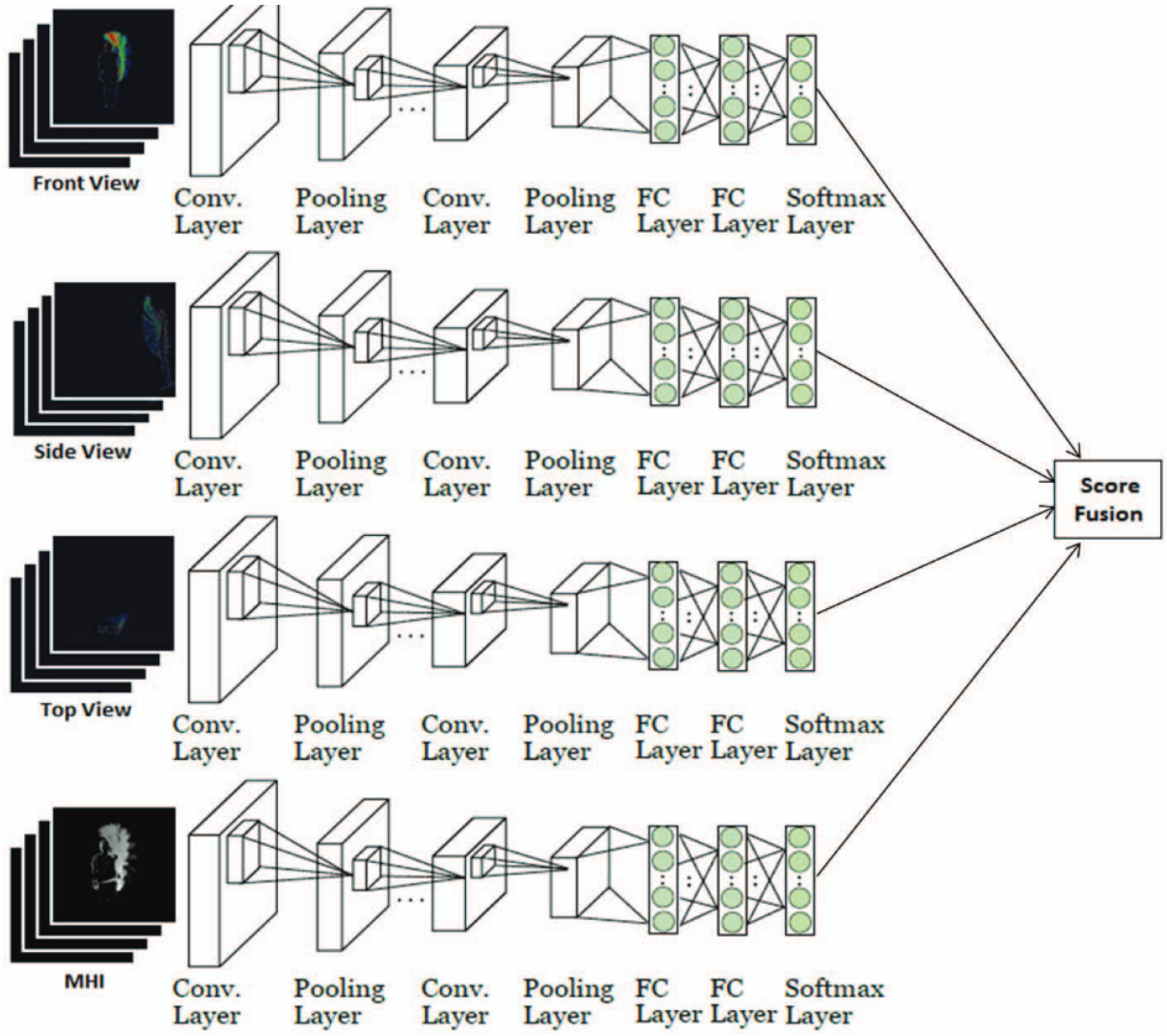


Figure 2.4: **Depth Motion Maps (DMM) and Motion History Images (MHI) based on 4-stream deep CNN architecture proposed by Imran *et al.* [2].**

approaches of representing the spatio-temporal relationship for human interaction.

Those kinds of relationships are commonly modeled based on expert domain knowledge, such as context-free grammar [42], ontology [43], though it is time-consuming to design well. Thus, to cope with this limitation, methods for automatically learning the spatio-temporal relationships in human activity were proposed in recent years [7, 8, 9, 10, 11, 12, 44].

In [7], Kiwon *et al.* proposed different kinds of features to represent both temporal dynamics of human pose and the spatio-temporal relationships

between two persons for human-human interaction recognition. Their experiments showed that joint distance feature and joint motion feature, which are computed based on the Euclidean distances between all pairs of joints of two persons, are the most effective features for human interaction recognition. There is much redundant information which reduces the discriminative power of their model for human interaction recognition task. To overcome this problem, Li *et al.* [13] proposed an Active Joint Interaction Graph to model the spatio-temporal relationships for human-human interaction. To remove redundant joints that do not involve human interaction, they proposed Relative Variance of Joint Relative Distance for choosing the number of spatio-temporal are being used for classification.

In [8], Sung *et al.* presented a two-layered Maximum Entropy Markov Model (MEMM) to model different characteristics of the human activities. They extracted the relation between sub-activities and different types of features from the RGB-D sensor (*e.g.* Microsoft Kinect). They computed those features based on human pose and motion, as well as image and point cloud information. The hierarchical nature and the transition between sub-activities over time are also being used to determine MEMM graph structure based on dynamic programming.

Ni *et al.* [9] introduced a multi-level framework to recognize human activity and localize the position of activity from grayscale and depth images simultaneously. In the first level, they removed false detections by applying different depth-based filters to the detected human and object bounding boxes. In the next level of this framework, they extracted 3-D spatial-temporal relationships between human subjects and objects that were detected in the previous step. They also classified the type of indoor environment where the activity was performed and used it as contextual information to make the final prediction. Finally, a latent SVM model is applied to obtain activity classification.

Gupta *et al.* [10] proposed a method to decide the class of human activities based on available cues which are extracted from depth information. They combined depth and spatial information of the segmented body to generate a human pose description. Then, the unique poses (codewords) are exploited based on a spatial domain clustering method.

Koppula *et al.* [11] presented a graph-based method to model the spatio-

temporal relationship for recognizing human-object interaction. They focused on the problems of recognizing human activities and object affordances from RGB-D videos in personal robots. They consider a long-term human daily activity as a sequence of sub-activities and object affordances. Thus, to label the activities being performed in the RGB-D videos, they jointly model the human activities and object affordances based on Markov Random Field (MRF). They represent the objects and sub-activities by the nodes of the graph. The relationships between object affordances and sub-activities, as well as their transition of those relationships over the time are represented by the edges of the graph. They applied a Structural Support Vector Machine (SSVM) to learn the parameters of this model. To reduce the number of nodes, they applied a temporal segmentation to group similar frames as one node.

More recently, to combine the sequential learning power of Deep RNN and contextual information of spatio-temporal graphs, Jain *et al.* [45] introduced a generic method for converting any spatio-temporal graph into a structured Deep RNN. They formed their structured Deep RNN for human-object interaction problems based on the spatio-temporal graph proposed in [11]. They also achieved a significant improvement compared with the previous work by Koppula *et al.*, because their method did not need to assume Markov property like spatio-temporal Conditional Random Fields (CRF) which is not good for long-term dependency problem such as anticipation [45]. It is worth to note that, the spatio-temporal graph used in their work did not consider object-object relationship, which play important roles in human interactions and reflect the change of objects caused by human activity.

Chapter 3

Recurrent Neural Networks

3.1 Recurrent Neural Networks (RNNs)

RNNs are a special kind of neural network with closed loop connections [46]. They were developed to learn sequential and time-varying patterns in a wide range of problems where input data is a time sequences of events or ordered data (*e.g.* speech recognition, language modeling, translation, image captioning, action recognition) [29, 35, 36, 47, 48, 49, 50].

Unlike the traditional neural network, we could apply RNNs on sequences with arbitrary length by adding the feedback (closed loop) connections [51]. It also allows RNNs to leverage the sequential information of the previous elements in a sequence, instead of treating the inputs (outputs) independently like traditional neural networks. For instance, to recognize the action in a video, we traditionally extract the features from an input video. Then, we apply feature representation method such as "Bag-of-Feature" to as a vector to integrate the extracted features into a vector with the fixed number of dimension. Hence, this potentially eliminates the sequential pattern of human actions and degrades the performance of recognition system, while RNNs could preserve the sequential information by considering each video frame as the state of a action.

Given an input sequence $\mathbf{x} = \left(x^1, x^2, ..., x^T\right)$, a standard Recurrent Neural Network (RNN) (Figure 3.1) computes the hidden vector sequence $\mathbf{s} = \left(s^1, s^2, ..., s^T\right)$ and output vector sequence $\mathbf{y} = \left(y^1, y^2, ..., y^T\right)$.

For each timestep t, both new input vector $\mathbf{x}^t$ and old hidden state vector $\mathbf{h}^{t-1}$ of $\mathbf{x}^{t-1}$ were fed into the networks. Then, the output $\mathbf{y}^t$ and the hidden state $\mathbf{h}^t$ can be computed as follows

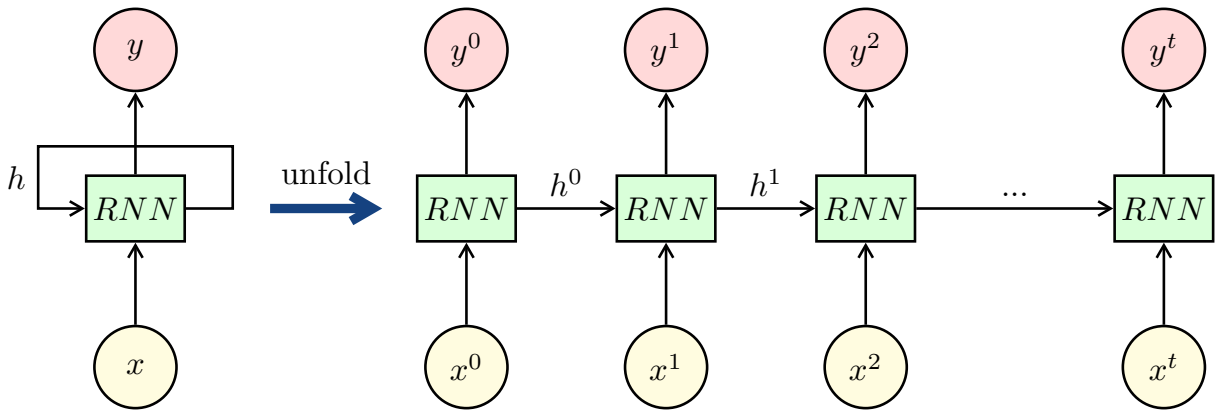


Figure 3.1: **The illustration of the standard RNN.** On the left side of the figure is the standard RNN with loops. The right side of the figure shows the full network after being unrolled (unfolded). Note that the parameters of the RNN are the same at every timestep.

$$\mathbf{s}^t = \mathbf{W}_{xh}\mathbf{x}^t + \mathbf{W}_{hh}\mathbf{h}^{t-1} + \mathbf{b}_h \tag{3.1}$$

$$\mathbf{h}^t = \sigma_h\left(\mathbf{s}^t\right) \tag{3.2}$$

$$\mathbf{y}^t = \sigma_y\left(\mathbf{W}_{hy}\mathbf{h}^t + \mathbf{b}_y\right) \tag{3.3}$$

where σ_h, σ_y denote activation function of hidden layer and output layer, $\mathbf{W}_{xh}$ is the matrix of hidden weights between input layer $\mathbf{x}$ and hidden layer $\mathbf{h}$, $\mathbf{W}_{hh}$ is the matrix of recurrent weights from hidden layer $\mathbf{h}$ to itself, $\mathbf{W}_{hy}$ is the matrix of weights between hidden layer $\mathbf{h}$ and output layer $\mathbf{y}$, $\mathbf{b}_h$ and $\mathbf{b}_y$ are two bias vectors.

Note that we have to choose initial values $\mathbf{h}_0$ for the hidden units, becausse we cannot compute $\mathbf{h}_0$ via equation 3.2 before the networks receive any information from the data sequence. Typically, we choose zero to be the initial values of $\mathbf{h}_0$. However, we could improve the stability and performance of RNNs by using nonzero initial values in some cases [52].

3.2 Recurrent Neural Networks optimization

Normally, the weights of neural networks could be optimized by backpropagation. However, it is a different story for the weights of RNNs. In fact, a hidden layer in RNNs affects not only the output of the networks but

also the hidden layer at the next timestep. Therefore, we could not apply backpropagation on RNNs.

Many methods have been developed to optimize the weights of RNNs such as Real Time Recurrent Learning (RTRL) [53], Backpropagation Through Time (BPTT) [54, 55]. In this work, we applied BPTT to train the proposed networks because of its simplicity and effectiveness in computation time. The main idea of BPTT is to calculate the derivatives of loss function $\mathcal{L}$ with respect to each of the network weights. Then, it repeatedly updates the weights of the networks to decrease the loss.

Like normal backpropagation, BPTT compute the gradient through the repeated application of chain rule for partial derivatives. The derivatives with respect to each of the weight of a standard RNNs are defined as:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}_{xh}} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}}{\partial \mathbf{h}^t} \frac{\partial \mathbf{h}^t}{\partial \mathbf{W}_{xh}} = \sum_{t=1}^{T} \delta^t \mathbf{x}^t \tag{3.4}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}_{hh}} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}}{\partial \mathbf{h}^t} \frac{\partial \mathbf{h}^t}{\partial \mathbf{W}_{hh}} = \sum_{t=1}^{T} \delta^t \mathbf{h}^t \tag{3.5}$$

where

$$\delta^t = \frac{\partial \mathcal{L}}{\partial \mathbf{h}^t} \tag{3.6}$$

If $t < T$, we have

$$\delta^t = \frac{\partial \mathcal{L}}{\partial \mathbf{h}^{t+1}} \frac{\partial \mathbf{h}^{t+1}}{\partial \mathbf{s}^{t+1}} \frac{\partial \mathbf{s}^{t+1}}{\partial \mathbf{h}^t} = \delta^{t+1} {\sigma_h}' \left(\mathbf{s}^{t+1}\right) \mathbf{W}_{hh} \tag{3.7}$$

Otherwise, we could compute δ^T directly without using chain rule. As mentioned above, previous video frames are informative to understand the present frame for activity recognition systems. However, human activities are usually performed over a long period of time. Therefore, an effective learning model for this kind of problem has to be able to deal with the long-term dependencies.

In theory, RNNs is able to handle this type of data without any difficulties. However, because we have to pass the gradient back through a large number of timesteps, the gradient in conventional RNNs tends to vanish

or explode [56, 57, 58]. One of the common solutions is to replace vanilla RNN units by Long Short-Term Memory cells [57].

3.3 Long Short-Term Memory (LSTM)

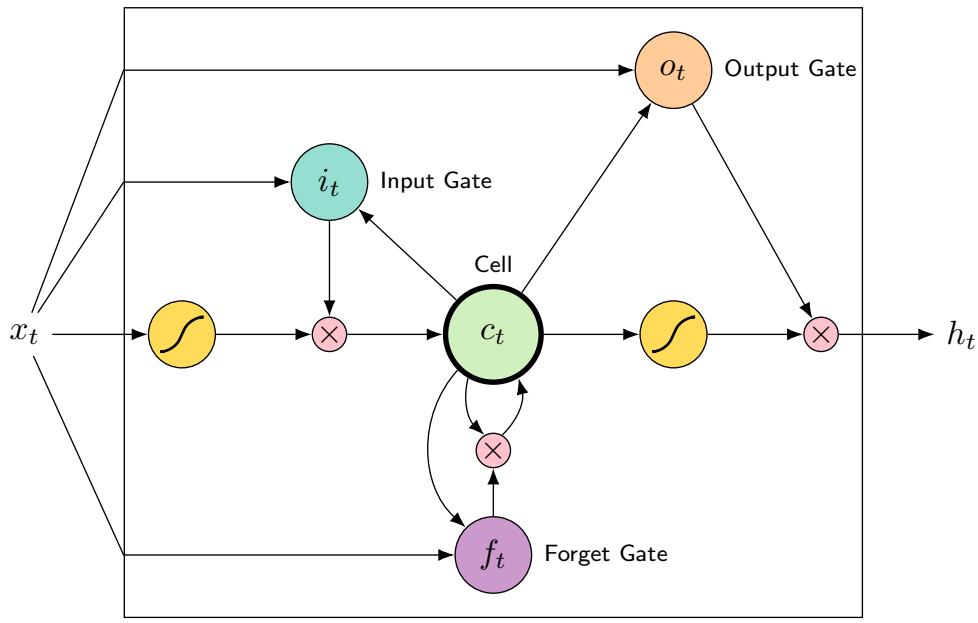


Figure 3.2: The structure of an LSTM neuron.

LSTM is a variant of RNNs proposed by Hochreiter and Schmidhuber [57] to model long-term dependencies better [45]. A single cell of LSTM neuron contains input gate i, forget gate f, memory cell c, output gate o and and hidden state h (Figure [?]). At each timestep of LSTM, the activation of the recurrent unit is given as follows:

$$i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right) \tag{3.8}$$

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \tag{3.9}$$

$$\begin{aligned} c_t = {} & f_t \odot c_{t-1} + \\ & i_t \odot tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \end{aligned} \tag{3.10}$$

$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o\right) \tag{3.11}$$

$$h_t = o_t \odot tanh\left(c_t\right) \tag{3.12}$$

where $\odot$ denotes element-wise product, $\sigma(\cdot)$ is the sigmoid function, W_{mn} is connection weights between m and n. b_i, b_f, b_o and b_c are four bias vectors. This architecture allows the output gate o to learn the way to derive the output of the LSTM unit from the current state of the internal memory cell c_t. Thus, the hidden state h_t in LSTM is not only computed based on previous hidden state h_{t-1} like conventional RNNs. The hidden state h_t also computed based on older hidden states before $t-1$ that is stored in the memory cell. This mechanism ensures the backward gradient can be passed through a long sequence of timesteps and prevents vanishing gradient problem in conventional RNNs.

Chapter 4

S-RNN for Human-Object Interactions

According to [3], we can classify human interaction into two sub-categories, namely human-human interactions and human-object interactions. Due to the differences in the characteristics of these sub-categories, the author separately design two structural Deep RNN for human-human and human-object interaction.

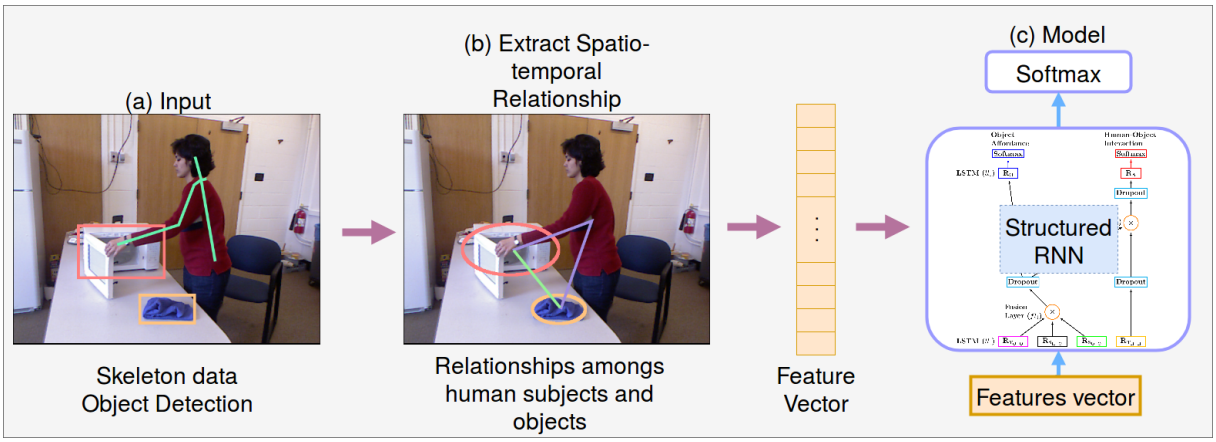


Figure 4.1: **The illustration for the proposed system overview.** (a) The author process the input data including human skeletons and object bounding boxes. (b) The author extract the features for different kinds of spatio-temporal relationships for human interaction. (c) The author feed the feature vector into the S-RNN to obtain the final results. The details of the S-RNN are described in the following sections.

In this section, the author briefly review the architecture of LSTM first. Then, the author describe the proposed Deep RNN architecture with LSTM

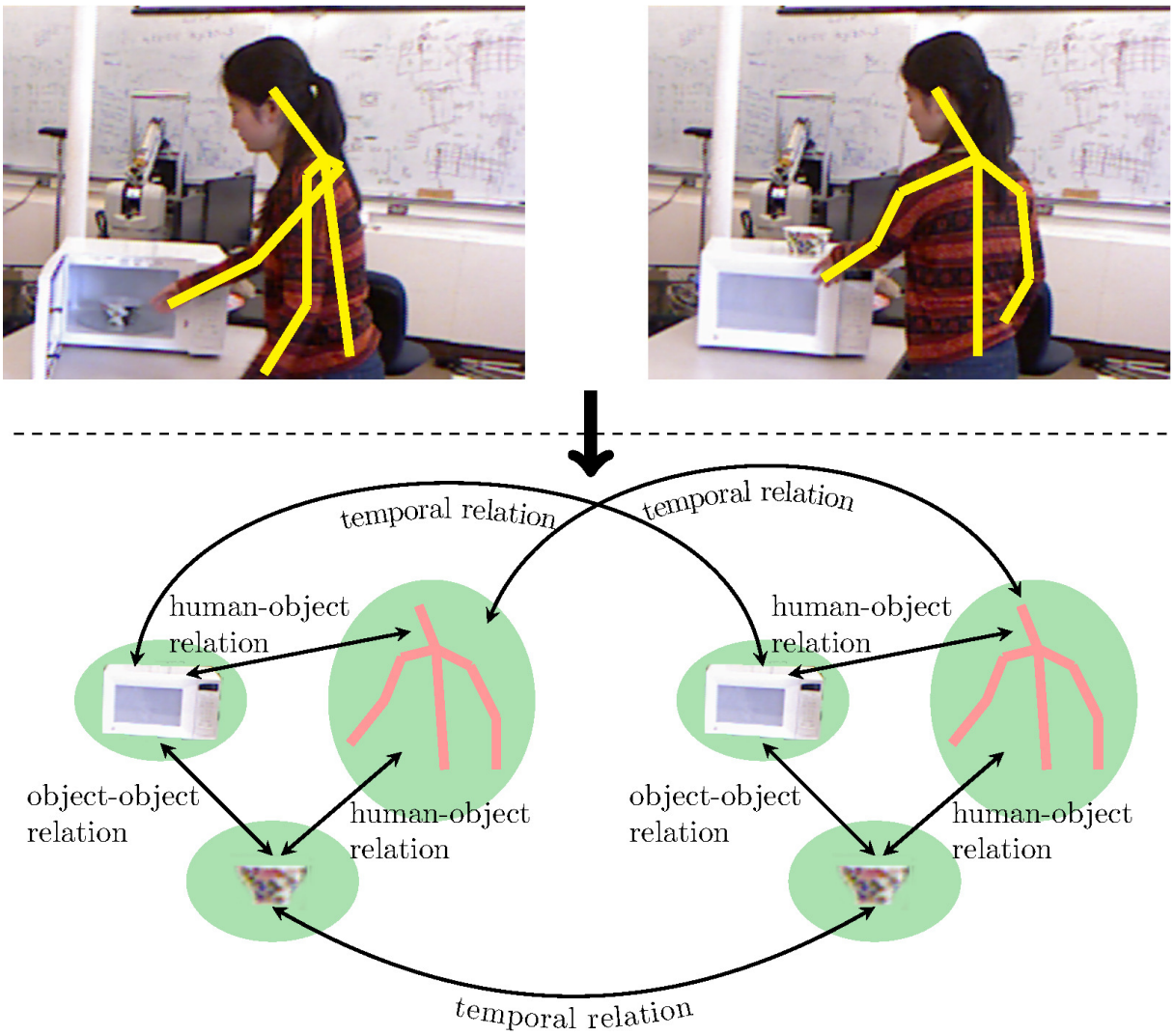


Figure 4.2: **An example of the different types of relationships and components in the activity taking out food from a microwave oven.** At the bottom of the picture, the graph illustrates the spatio-temporal relationship in the interaction between the person with the microwave oven and the bowl of food.

units to model the relationship of human interaction. Finally, we discuss the training process of the proposed model, and the result of the proposed method on CAD-120 dataset.

4.1 Human-object interaction modeling

Generally, human-object interaction consists of two kinds of relations, namely spatial relation and temporal relation. Temporal relation describes

the change of component states during human activity (*e.g.* sum displacement of each joint locations describes the transition of human pose [11]). Spatial relation specifies the property in terms of location from an object to another object [59]. For instance, as shown in Figure 4.2, the illustration of "taking out food from a microwave oven" consists of 3 components, which are the woman, the microwave, and the bowl of food. There are two human-object spatial relations between the woman and other objects. One object-object spatial relation between the microwave and the bowl of food. There is a temporal relation correspond to each component (temporal relations of the woman, the microwave and the bowl of food).

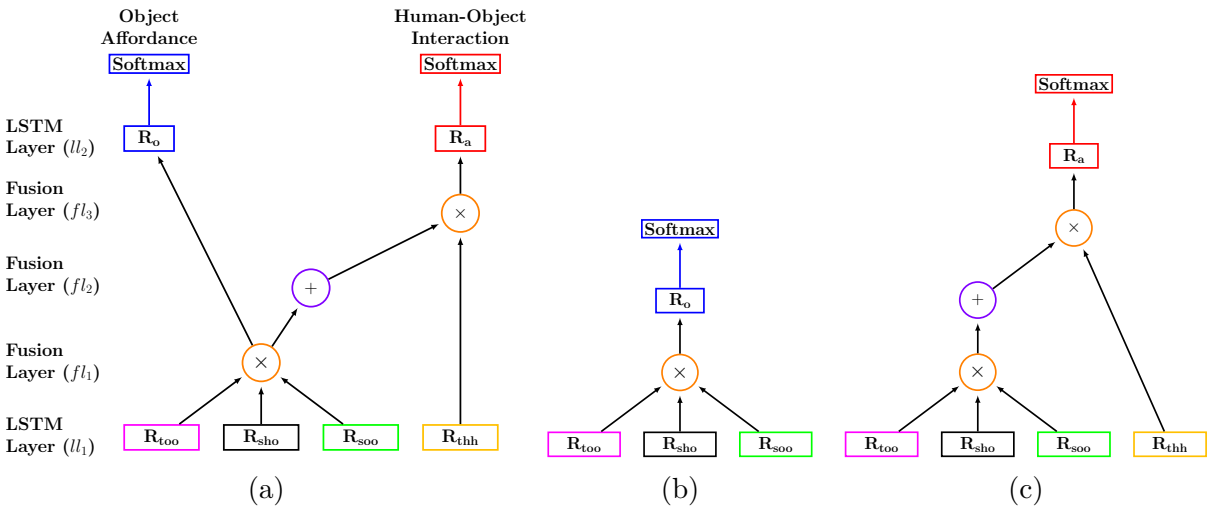


Figure 4.3: **The architecture of the proposed S-RNN for learning human-object interactions and object affordances.** (a) The general model for learning human-object interactions and object affordances. Every rectangular node in this model is a LSTM. (b) The forward pass of object affordance node. (c) The forward pass of human activity node.

Jain *et al.* [45] proposed a structure of Deep RNN based on the spatio-temporal graph from [60] for representing spatio-temporal relationships. However, it did not take the temporal relation of objects and the spatial relations between objects into account as a part of human activity. As mentioned before, these relations depict informative details of the way human activity changes the objects over the time. Therefore, this approach fails to discriminate some human activities that could only be recognized based on the results of the interactions between human subject and objects. Thus, the author cover these relations in the proposed S-RNN to capture

the transition in term of space and time of the relationship among human subjects and objects for recognizing the human-object interaction.

Given a T timesteps video with a human subject h and the set of interesting object $O = \{o_1, o_2, ..., o_k\}$, where k is the number of objects in the observed scene. The author model the spatio-temporal relationships of an activity by a Deep RNN with several subnets that correspond to each spatio-temporal relationship. Figure 4.3a shows the proposed S-RNN architecture for modeling the spatio-temporal structure of human-object interactions. The proposed model is comprised of five layers with different roles: ll_1, fl_1, fl_2, fl_3, and ll_2.

In the first layer ll_1, there are four single-layer LSTMs which are R_{thh}, R_{soo}, R_{sho} and R_{too}. For each timestep, R_{thh} receives a $1 \times d_{thh}$ vector $\mathbf{v}_{thh}$ of human-human temporal relational features of human subject h. The subnet R_{sho} receives a $k \times d_{sho}$ matrix A_{sho} of human-object spatial relational features between human subject h and all interesting objects in the observed scene. R_{soo} receives a $k \times d_{soo}$ matrix of object-object spatial relational features, where i-th row of the matrix represents the spatial relationships between object o_i and other interesting objects. The subnet R_{too} receives a $k \times d_{too}$ matrix of object-object temporal relational features, where each i-th row of the matrix represents the temporal relationship of object o_i.

R_{thh} and R_{too} nodes in this model are LSTM units with $tanh$ activation corresponding to temporal human-human and object-object relations, respectively. R_{sho} and R_{soo} nodes in this model are an LSTM with $tanh$ activation corresponding to spatial human-object relation and object-object relation, respectively.

For each object, in order to model the hidden representation of spatio-temporal relations, we combine the output of hidden layers R_{soo}, R_{sho} and R_{too} at the fusion layer fl_1. At each timestep, the human subject could interact with multiple objects (*e.g.* while the human subject is pouring cereal into a bowl, he/she could interact with a box of cereal and a bowl). Hence, for each timestep t, the hidden representation of spatio-temporal relations of all objects are merged at fl_2. Then, at the fusion layer fl_3, concatenation operator is applied to combine the hidden representation of human with the hidden of objects. Finally, the layer ll_2 and softmax are

applied to get final classification results.

4.2 Training S-RNN architecture

In this section, we discuss the training process of the proposed S-RNN model in detail.

Forward pass: Figure 4.3c and 4.3b show the forward pass through the proposed architecture to label both human activity and object affordance. At timestep $t \in [1, ..., T]$, the sequences of relational features were fed into R_{thh}, R_{too}, R_{sho}, and R_{soo}. Then, layer fl_1 concatenates all the hidden representation $h_{ho}^{spatial}$, $h_{oo}^{spatial}$, and $h_{oo}^{temporal}$ of object q into one vector as follows:

$$\begin{aligned} h^{object}(q,t) = h_{ho}^{spatial}(q,t) \otimes h_{oo}^{spatial}(q,t) \\ \otimes h_{oo}^{temporal}(q,t) \end{aligned} \tag{4.1}$$

Here, $\otimes$ denotes the concatenation operator which is described as follows.

$$\begin{cases} \mathbf{a} = \left(a^1, a^2, ..., a^m\right) \\ \mathbf{b} = \left(b^1, b^2, ..., b^n\right) \\ \mathbf{a} \otimes \mathbf{b} = \left(a^1, a^2, ..., a^m, b^1, b^2, ..., b^n\right) \end{cases} \tag{4.2}$$

Then, h^{object} is fed into R_o to classify object affordance. Hidden representation of all objects are merged to feed to subnet R_a for human activity labeling as follows:

$$h^{object}(t) = \sum_{q \in Q} h^{object}(q,t) \tag{4.3}$$

Then, the concatenation operator is applied once again to combine the hidden representation of human and object.

$$h^{human-object}(t) = h_{hh}^{temporal}(t) \otimes h^{object}(q) \tag{4.4}$$

In the above formula, $h_{hh}^{temporal}(t)$ denotes human-human temporal relation at t-th timestep. For each task, we apply a fully-connected layer to

obtain classification result.

$$\mathbf{z}_{\text{sub-activity}}(t) = \mathbf{W}_{\text{sub-activity}} h^{human-object}(t) + \mathbf{b}_{\text{sub-activity}} \tag{4.5}$$

$$\mathbf{z}_{\text{affordance}}(t) = \mathbf{W}_{\text{affordance}} h^{object}(t) + \mathbf{b}_{\text{affordance}} \tag{4.6}$$

$$\mathbf{z}_{\text{high-level activity}} = \mathbf{W}_{\text{high-level activity}} h^{human-object}(T) + \mathbf{b}_{\text{high-level activity}} \tag{4.7}$$

Finally, the results are normalized by softmax activation to obtain the probability of each class.

$$\begin{gathered} p : \mathbb{R}^K \rightarrow [0,1]^K \\ p_i = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \text{ for } i = 1, ..., K \end{gathered} \tag{4.8}$$

The objective function of S-RNN is to minimize categorical cross-entropy between predictions and ground truth:

$$\mathcal{L} = -log(p_i) \tag{4.9}$$

where i denotes the target class.

Backward pass. At the training phase, the errors in prediction are backpropagated through all nodes in the proposed structured RNN to update the weights. Finally, to ensure the better performance in optimization, the author use ADAGRAD which is an adaptive gradient descent algorithm proposed by Duchi *et al.* [61].

4.3 Feature extraction

For human-object interaction recognition, in the experiment, the author used the well-defined features for spatio-temporal relationships that were used in [11, 45]. The following in this section describes the definitions of these features.

To represent the object-object temporal relation of a given object o, the location of object in the scene and how the object change over the time are used. The (x, y, z) coordinates of the centroid of the object o as well as

the coordinates of its bounding box are used to localize the object o. To represent how the object change over the time, the total displacement and the total distance moved by the centroid of the object are computed. The transformation matrix of SIFT key points [62] of object o between adjacent frames is also used to represent this aspect.

To represent the human-human temporal relation of a given human subject h, the different features based on the human skeleton are extracted from RGB-D video. These features include the location of every joint (8 joints in total), the moved distance of each joint (8 joints in total), displacement of each joint (8 joints). In CAD-120 dataset, the lower body of the human subjects are obstructed and cannot be observed in many frames. Therefore, the author decided to use only the upper-skeleton joints to extract features. Besides, the human activity is irrelevant to the human activities in this dataset. The body pose features and hand position features proposed by Sung *et al*. [8] are also used in this work.

To represent the human-object temporal relation between a given human subject h and object o, the distance between each joint location and object centroid are used. The difference in centroid locations $(\Delta x, \Delta y, \Delta z)$ and distance between centroids are used to represent the spatial relationships between two objects.

4.4 Evaluation datasets

For evaluating the performance of the proposed method on human-human interaction recognition, the author conduct the experiments on two widely used human interaction datasets: SBU Interaction dataset and M2I dataset. In order to evaluate the performance of the proposed method on human-object interaction recognition and anticipation, the author also report the results on the CAD-120 dataset. This dataset contains numerous daily human-object interactions. Table 4.1 shows the characteristics of the evaluation dataset which are used in this study.

CAD-120 dataset [11] is a challenging benchmark for human-object interactions recognition from RGB-D videos because of this dataset consists of many complex human interactions. The dataset was collected by the Cornell University. This dataset contains RGB, depth and skeleton co-

	CAD-120	SBU	M2I
Number of classes	10	8	9
Number of subjects	4	7	20
Number of environments	4	1	1
Number of captured views	1	1	2
Number of video sequences	120	300	720
Resolution	320 × 240	640 × 480	320 × 240
Frame rates	30 fps	15 fps	30 fps
Skeleton data (number of joints)	15	15	20
Year	2013	2012	2015

Table 4.1: **Summary of the statistics of the evaluation datasets.**

ordinates of 120 activity sequences, from ten different high-level activities. Figure 4.4 shows the skeleton joints provided by CAD-120 dataset.

Each of the high-level activities is performed three times by four subjects. The high-level activities include: *making cereal, taking medicine, stacking objects, unstacking objects, microwaving food, picking objects, cleaning objects, taking food, arranging objects* and *having a meal*. Each high-level activity consists of a sequence of atomic sub-activities (*e.g.* reaching, moving, *etc.*) is performed for a long duration. Sub-activities in the dataset include *reaching, moving, pouring, eating, drinking, opening, placing, closing, scrubbing,* and *null*.

This dataset also considers object affordance, which represents the use of an object in specific situations [11, 63]. By understanding object affordances in the surrounding environment, robots could learn the way that a human subject interacts with other objects. Then, robots could plan their actions to interact with objects which is a vital task for robotic applications. The affordances in CAD-120 include: *reachable, movable, pourable, pourto, containable, drinkable, openable, placeable, closable, scrubbable, scrubber,* and *stationary*. Figure 4.5 shows some examples snapshots of all classes of CAD-120 dataset.

Note that a human subject interact with an object in different ways depending on the particular situation during human activity. Hence, the affordance can also be changed in accordance with the way human subject interact with that object. For example, a microwave can be *openable* when the human subject wants to warm up the food in his cup. It is *closable* when

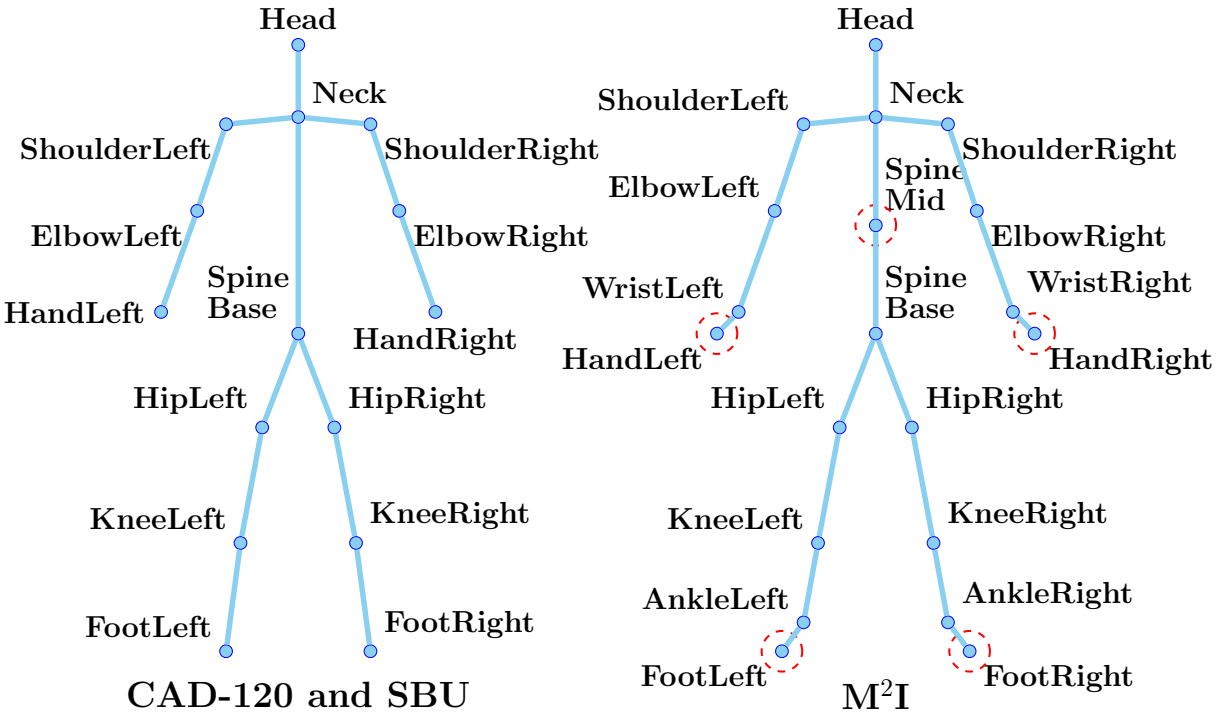


Figure 4.4: **The human skeleton joints provided by CAD-120, SBU Interaction, and M2I datasets.** The dashed circles indicate the joints that were not used to extract the features for the spatio-temporal relationships in the experiments.

the human subject wants to want close the microwave's door. Table 4.2 shows the detailed description of high-level activities with regard to sub-activities. Some activities are composed of same sub-activities. But, they have occurred in a different order.

	reaching	moving	placing	opening	closing	eating	drinking	pouring	scrubbing	null
Making Cereal	✓	✓	✓					✓		✓
Taking Medicine	✓	✓	✓	✓		✓	✓			✓
Stacking Objects	✓	✓	✓							✓
Unstacking Objects	✓	✓	✓							✓
Microwaving Food	✓	✓	✓	✓	✓					✓
Picking Objects	✓	✓								✓
Cleaning Objects	✓	✓		✓	✓				✓	✓
Taking Food	✓		✓	✓	✓					✓
Arranging Objects	✓	✓	✓							✓
Having a Meal	✓	✓				✓	✓			✓

Table 4.2: **The description of high-level activities with regard to sub-activities.** Some activities are composed of same sub-activities. But, they have occurred in a different order.

(a) Arranging objects (b) Cleaning objects

(c) Having a meal (d) Making cereal

(e) Microwaving food (f) Picking objects

(g) Stacking objects (h) Taking food

(i) Taking medicine (j) Unstacking objects

Figure 4.5: **Example snapshots of all classes of CAD-120 dataset** [11].

4.5 Evaluation metrics

In the experiments, the author applied 4-fold cross-validation as the evaluation method, which is the standard evaluation method of this dataset [11, 45]. The author used 93 samples to train the networks and 31 samples to evaluate the performance. To precisely evaluate the performance of the proposed method, the author executed the same training and testing processes 10 times on every fold. Then, the results are averaged over all executions to obtain the final performance.

To evaluate the performance of the proposed method on CAD-120 dataset, the author computed the accuracy for each of the interaction classes. The average class accuracy over all classes are reported to show the performance of the proposed method. The accuracy is defined as:

$$\text{accuracy} = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.10}$$

where P denotes positive instances, N denotes negative instances, TP denotes true positive, FP denotes false positive, TN denotes true negative, and FN denotes false negative. Let C denote the number of classes. The average class accuracy is defined as follows:

$$\text{average accuracy} = \frac{\sum_{c \in 1,\ldots,C} (TP_c + TN_c)}{\sum_{c \in 1,\ldots,C} (TP_c + TN_c + FP_c + FN_c)} \tag{4.11}$$

where TP_c, FP_c, TN_c, and FN_c denote true positive, false positive, true negative, and false negative of class c, respectively.

4.6 Experimental setup

In the experiments, the author implemented the proposed model based on Lasagne library [64] with Theano [65] as the deep learning platform. The author performed the experiments on an Intel Xeon E5-2680v2 (2.8 GHz 10-core central processing unit, 64 GB RAM) and a NVIDIA Tesla K40 GPU to accelerate the training and testing process. The author train the network using ADAGRAD and set learning rate as 0.01 like most implementations of ADAGRAD. To avoid overfitting, the author apply L1 Regularization with $\lambda = 10^{-4}$. The applied probability of dropout that the author configured at each LSTM in the network was 0.5. The author used the same parameter settings mentioned above for all of the experiments and achieved promising results on all the datasets with the same configuration.

CAD-120 dataset. The learning model is trained on the activities of three subjects, then tested on a new (unseen) subject. The final results are averaged with regard to accuracy across the folds. To train the proposed networks for detecting tasks, the author use the labels of the current time step, while the labels of the next time step are used to train the network for anticipation. The author also jointly train the proposed model with both human activity and object affordance labels at the same time.

The author use a single layer LSTM of size 128 considering the size of CAD-120 dataset for $R_{T_{H-H}}, R_{T_{O-O}}, R_{S_{H-O}}$ and $R_{S_{O-O}}$ nodes. The author use single LSTM layer of size 256 for R_A and R_O. The number of classes

in the final softmax layer of S-RNN for high-level activity, sub-activity and object affordance are 10, 10 and 12, respectively.

4.7 Experimental results

CAD-120 dataset. Table 4.3 summarizes the results of detection and anticipation of the proposed method compared with some of the state-of-the-art methods in terms of accuracy averaged over all the classes. The importance of object states and object-object relations to model spatio-temporal relationship for sub-activity detection and anticipation was shown as the improvement by the proposed method in Table 4.3. The proposed method mostly outperforms with regard to both anticipation and detection when compared with previous works based on spatio-temporal relation by Jain *et al.* [45] and by Koppula *et al.* [12]. Especially, the accuracy of sub-activity detection and the accuracy of anticipation are improved as much as 7% and 14%, respectively, compared with the state of the art structured RNN model for modeling spatio-temporal relations proposed by Jain *et al.* [45]. Figure 4.8 and Figure 4.9 shows the confusion matrices for detection and anticipation tasks using the proposed method, respectively. The ground truth activity labels are on rows and the predictions are on columns.

	Detection (%)		Anticipation (%)	
Method	Sub-activity	Object Affordance	Sub-activity	Object Affordance
Hu *et al.* [66]	87.0	N/A	N/A	N/A
Koppula *et al.* [12]	89.3	**93.9**	49.6	67.2
Jain *et al.* [45]	83.2	91.1	65.6	80.9
Taha *et al.* [28]	**91.6**	N/A	N/A	N/A
Structured S-LSTM (Proposed)	90.4	91.8	**80.0**	**85.9**

Table 4.3: **Detection and anticipation results on CAD-120 [11].** Showing average accuracy for affordances, sub-activities.

Note that these methods did not take spatial and temporal object-object relations into account for labeling sub-activities. Thus, these models are

Method	High-level Activity (%)
Koppula *et al.* [12]	93.5
Taha *et al.* [28]	94.4
Structured S-LSTM (Proposed)	**96.4**

Table 4.4: **High-level Activity recognition results on CAD-120 [11].** Showing average accuracy for high-level activity.

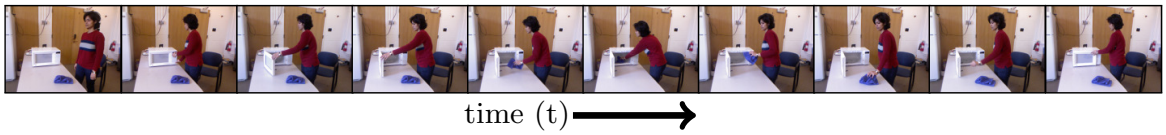
hard to distinguish the activities which have the similar human pose and human-object relations (*e.g.* moving and scrubbing). Taha *et al.* [28] achieved a slightly better performance than the proposed method on sub-activity detection. However, their method is limited to human activity detection. While the proposed method achieved stable performance across various tasks, the method in [28] is not able to recognize and anticipate the object affordances, which are really important for robotic applications as mentioned before.

Table 4.4 illustrates the comparison of the results of high-level activity recognition, where the proposed method achieves better performance than state-of-the-art methods. Note that S-RNN inherits the advantages of modeling long-term contextual information from deep LSTM, and does not need to make any assumption like spatio-temporal CRF [12]. It makes the proposed method work well with long-term and complex human-object interactions.

		a	b	c	d	e	f	g	h	i	k
Arranging objects	a	1.0									
Cleaning objects	b		0.88						0.05	0.07	
Having a meal	c			1.0							
Making cereal	d				1.0						
Microwaving food	e					0.93			0.07		
Picking objects	f						1.0				
Stacking objects	g					0.07		0.93			
Taking food	h		0.03			0.02			0.95		
Taking medicine	i									1.0	
Unstacking objects	k					0.04			0.02		0.94

Figure 4.6: Confusion matrix of high-level activity recognition of the propsed method on CAD-120 dataset.

Moreover, the proposed method outperforms other methods in various



time (t) →

High-level Activity

Ground-truth	Cleaning objects
Detection	Cleaning objects

Sub-activity

Ground-truth	null	reaching	opening	reaching	moving	scrubbing	moving	placing	reaching	closing
Detection	null	reaching	opening	reaching	moving	scrubbing	moving	placing	reaching	closing
Anticipation		reaching	opening	reaching	moving	scrubbing	moving	placing	reaching	closing

Object Affordance - Microwave Oven

Ground-truth	stationary	reachable	openable	stationary	stationary	scrubbable	stationary	stationary	reachable	closeable
Detection	stationary	reachable	openable	stationary	stationary	<u>stationary</u>	stationary	stationary	reachable	closeable
Anticipation		reachable	openable	stationary	stationary	scrubbable	stationary	stationary	reachable	closeable

Object Affordance - Cloth

Ground-truth	stationary	stationary	stationary	reachable	movable	scrubber	movable	placeable	stationary	stationary
Detection	stationary	stationary	stationary	reachable	movable	<u>placeable</u>	movable	placeable	stationary	stationary
Anticipation		stationary	stationary	reachable	<u>stationary</u>	<u>stationary</u>	<u>stationary</u>	placeable	stationary	stationary

Figure 4.7: **An example result on cleaning activity on CAD-120.** This figure shows the success and failure cases (underlined) of detection and anticipation results of S-RNN.

tasks which refer to all information of human pose, human-object relations and the change of objects corresponding to human activity. Figure 4.7 shows an illustrative example of high-level recognition, sub-activity and object affordance of S-RNN of a 'cleaning object' high-level activity video. On the top of the figure, the author illustrate the picture of the sub-activities. The following table shows the comparison of the predictions on high-level activity, sub-activity, and affordances, with the ground-truth. Note that, the labels of sub-activity and affordance at time t are anticipated at time $t-1$. The result is correct if it is identical to the ground-truth from the same column. Because of a large number of "stationary" time steps, the prediction labels are often misclassified as "stationary".

		a	b	c	d	e	f	g	h	i	j
Reaching	a	0.92	0.06							0.02	
Moving	b	0.04	0.94							0.02	
Pouring	c		0.06	0.92				0.02			
Eating	d	0.06	0.08		0.76	0.04		0.01		0.05	
Drinking	e		0.01			0.96				0.03	
Opening	f	0.05	0.10				0.85				
Placing	g	0.01		0.01				0.95		0.03	
Closing	h	0.03	0.03					0.01	0.93		
Null	i	0.05	0.03			0.02		0.03		0.87	
Scrubbing	j					0.03		0.27			0.70

(a) Confusion matrix of sub-activity detection of the propsed method on CAD-120 dataset.

		a	b	c	d	e	f	g	h	i	j	k	l
Movable	a	0.91	0.05	0.02				0.02					
Stationary	b		0.98	0.02									
Reachable	c		0.23	0.76				0.02					
Pourable	d	0.03			0.94					0.03			
Pourto	e		0.27			0.73							
Containable	f		0.32	0.03			0.52		0.04			0.08	
Drinkable	g	0.15	0.02					0.83					
Openable	h	0.07	0.07	0.02					0.84				
Placeable	i	0.02	0.11							0.87			
Closable	j	0.02	0.13	0.04							0.81		
Scrubbable	k	0.02	0.32				0.02					0.62	0.03
Srubber	l		0.15				0.02			0.18		0.07	0.58

(b) Confusion matrix of object affordance detection of the propsed method on CAD-120 dataset.

Figure 4.8: **Confusion matrix of detection task of the propsed method on CAD-120 dataset.** Ground truth activity labels are on rows and the predictions are on columns.

		a	b	c	d	e	f	g	h	i	j
Reaching	a	0.82	0.12		0.01		0.01	0.01		0.02	
Moving	b	0.05	0.89		0.01	0.01	0.01	0.01		0.02	
Pouring	c	0.01	0.05	0.77				0.17			
Eating	d	0.11	0.41	0.01	0.38	0.05				0.04	
Drinking	e		0.11		0.03	0.76		0.07		0.03	
Opening	f	0.05	0.08				0.86				0.01
Placing	g	0.01	0.03			0.01		0.89		0.06	
Closing	h	0.06	0.07			0.01		0.04	0.78	0.03	
Null	i	0.17	0.25	0.01	0.02	0.03		0.09		0.42	
Scrubbing	j					0.05		0.58			0.37

(a) Confusion matrix of sub-activity anticipation of the propsed method on CAD-120 dataset.

		a	b	c	d	e	f	g	h	i	j	k	l
Movable	a	0.76	0.21	0.02						0.01			
Stationary	b	0.02	0.95	0.02						0.01			
Reachable	c	0.02	0.39	0.59									
Pourable	d	0.07	0.01		0.80			0.03		0.09			
Pourto	e		0.70			0.29				0.01			
Containable	f		0.31	0.01			0.68						
Drinkable	g	0.10	0.16					0.56		0.18			
Openable	h	0.04	0.12	0.03					0.80				
Placeable	i	0.02	0.12							0.86			
Closable	j		0.56	0.02							0.42		
Scrubbable	k	0.08	0.42				0.02					0.46	0.02
Srubber	l	0.02	0.25							0.24	0.01	0.02	0.46

(b) Confusion matrix of object affordance anticipation of the propsed method on CAD-120 dataset.

Figure 4.9: **Confusion matrix of anticipation task of the propsed method on CAD-120 dataset.** Ground truth activity labels are on rows and the predictions are on columns.

Chapter 5

S-RNN for Human-Human Interactions

5.1 Human-human interaction modeling

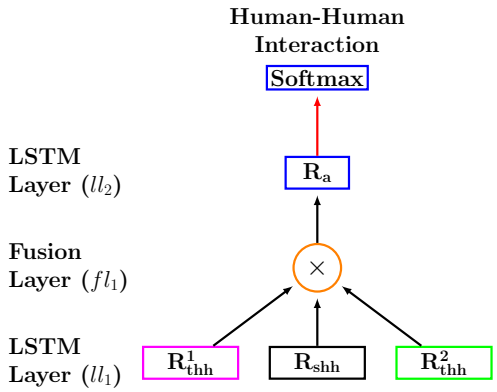


Figure 5.1: **The architecture of the proposed S-RNN for learning human-human interactions.**

Figure 5.1 shows the way that the author model the spatio-temporal relationships of a human-human interaction. The proposed model is comprised of three layers with different roles: ll_1, fl_1, and ll_2 as illustrated.

Given a T timesteps video with a human subject h_1 and human subject h_2. There are three single-layer LSTMs which are R^1_{thh}, R^2_{thh} and R_{shh} in the first layer ll_1. They receive human-human temporal relational features and human-human spatial relational features. Temporal relation describes the transition of human postures during human activity which is similar to human-object interaction. Spatial relation specifies the property in terms

of location between the joints of two human subjects.

R^1_{thh} and R^2_{thh} nodes in this model are LSTM units with $tanh$ activation corresponding to the temporal human-human relation of the human subject h_1 and the human subject h_2, respectively. R_{shh} node in the model is an LSTM with $tanh$ activation corresponding to spatial human-human relation between two human subjects. Then, the author apply concatenation operator to combine the hidden representation of human-human interaction at the fusion layer fl_1. Finally, the layer ll_2 and softmax are applied to get final classification result.

5.2 Training S-RNN architecture

In this section, we discuss the training process of the proposed S-RNN model in detail.

Forward pass. For human-human interaction, the sequences of features $x^{temporal}_{hh}(h_1, t)$, $x^{temporal}_{hh}(h_2, t)$ and $x^{spatial}_{hh}(h_1, h_2, t)$ were fed into R^1_{thh}, R^2_{thh} and R_{shh}, respectively, where $x^{spatial}_{hh}(h_1, h_2, t)$ denotes the spatial relation feature between two human subjects at timestep t. The feature vectors $x^{temporal}_{hh}(h_1, t)$ and $x^{temporal}_{hh}(h_2, t)$ are the human-human temporal relations of the human subject h_1 and the human subject h_2 at timestep t, respectively.

Backward pass. At the training phase, the errors in prediction are backpropagated through all nodes in the proposed structured RNN to update the weights. The author also applied ADAGRAD to ensure the better performance for optimization [61].

5.3 Feature extraction

In this work, the author used joint distance and joint motion as defined in [7], which are the geometric relational features based on the distance between all pairs of joints.

Joint distance: The joint distance is defined as the Euclidean distance between all pairs of joints of two persons at time step t. It captures the

distance between two skeleton joints in a single pose. It is defined as:

$$F^{\text{joint distance}}(i, j; t) = \left\| p^x_{i,t} - p^y_{j,t} \right\| \tag{5.1}$$

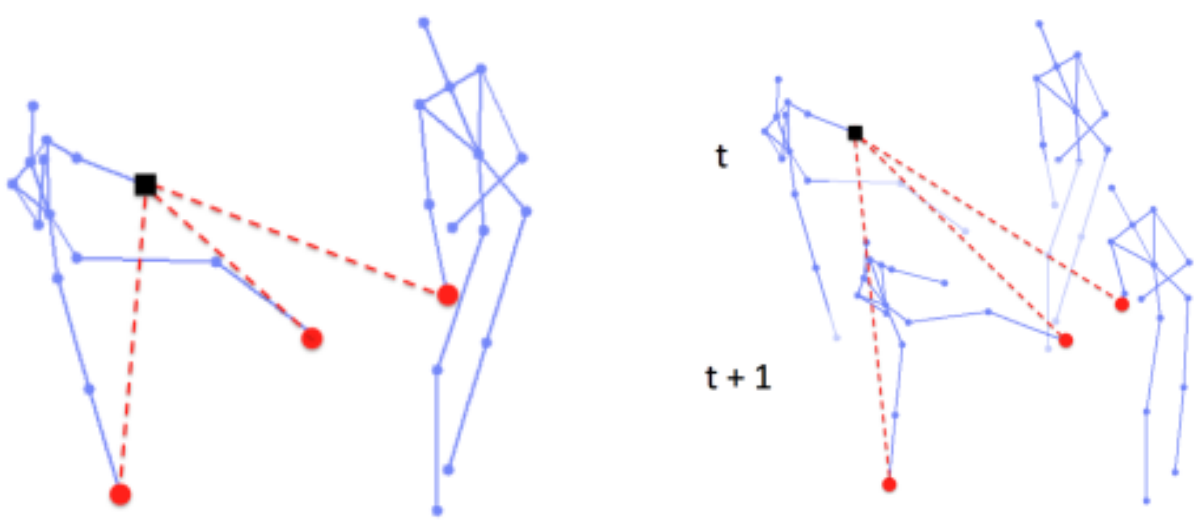
where i and j are any skeleton joints of two human subjects x and y, respectively. This represents human pose for one person in case where $x = y$. It represents spatial relation between two human subjects in case where $x \neq y$.

Joint motion: The joint motion at timestep t is defined as the Euclidean distance between all pairs of joints of two persons between timestep $t - 1$ and t. It is defined as:

$$F^{\text{joint motion}}(i, j; t) = \left\| p_{i,t} - p_{j,t-1} \right\| \tag{5.2}$$

where i and j are skeleton joint indices of human subject. It is used to measure the transition of human posture between two timesteps. In this work, the author

Figure 5.2a shows the illustration of joint distance and joint motion. Black rectangle indicates a reference joint, red circle indicates a target joint. Red line shown the instances that are computed by the definition of features.



(a) Joint distance

(b) Joint motion

Figure 5.2: **Human-human interaction features** [7].

5.4 Evaluation datasets

For evaluating the performance of the proposed method on human-human interaction recognition, the author conduct the experiments on two widely used human interaction datasets: SBU Interaction dataset and M2I dataset. Table 4.1 shows the characteristics of the evaluation datasets which are used in this study. The details of SBU Interaction dataset and M2I dataset are decribed in the following.

SBU Interaction dataset [7] contains 282 interaction sequences was collected with Microsoft Kinect (the resolution of both color image and depth map are 640 × 480 pixels) by Stony Brook University in 2012. It contains 8 classes of two-person interactions, including *approaching, departing, pushing, kicking, punching, exchanging objects, hugging,* and *shaking hands*. Seven participants were involved in performing the activities. The dataset consists of 21 sets of interactions performed once or twice by different pairs of participants. This dataset is quite challenging due to the low accuracy of joint locations in many sequences as well as the similarity of body movements between the activities. Figure 5.3 shows some example snapshots of all classes of CAD-120 dataset.

M2I dataset [67] contains 360 human-person interactions and was captured by two Kinect sensors from both front and side views, which was collected by Tianjin University in 2015. This dataset contains nine classes of human-human interactions, including *walk together, cross, wait, chat, hug, handshake, highfive, bow,* and *box*. Each type of interactions was performed twice by different pairs of participants. The Figure 5.4 shows the camera configurations of this dataset.

The M2I dataset contains: RGB data (image sequence sample: 6.79 Gigabytes; video sample: 19.2 Gigabytes); Depth data (image sequence sample: 49.4 Gigabytes); mask (image sequence sample: 613 Megabytes); 3D Skeleton data (53.9 Megabytes). Figure 4.4 shows the skeleton joints provided by SBU Interaction and M2I datasets. The dashed circles indicate the joints that were not used to extract the features for the spatio-temporal relationships in the experiments.

Figure 5.3: **Example snapshots of all classes of SBU Interaction dataset** [7].

5.5 Evaluation metrics

In the experiments, the author applied 5-fold cross-validation as the evaluation method on both SBU-Interaction and M2I. Note that 5-fold cross-validation is standard evaluation method on SBU dataset, while M2I does not have the standard evaluation method.

The author used 240 samples to train the networks and 60 samples to evaluate the performance for SBU dataset. The author tested the performance of each view of M2I dataset independently. The author used 256 samples to train the networks and 64 samples to evaluate the performance. The author executed the same training and testing processes 10 times on every fold to precisely evaluate the performance of the proposed method. Then, the results are averaged over all executions to obtain the performance of the proposed method.

The author compute the accuracy for each of the interaction classes and report the average class accuracy over all classes as defined in Section 4.5 to evaluate the performance of the proposed method on both SBU and M2I

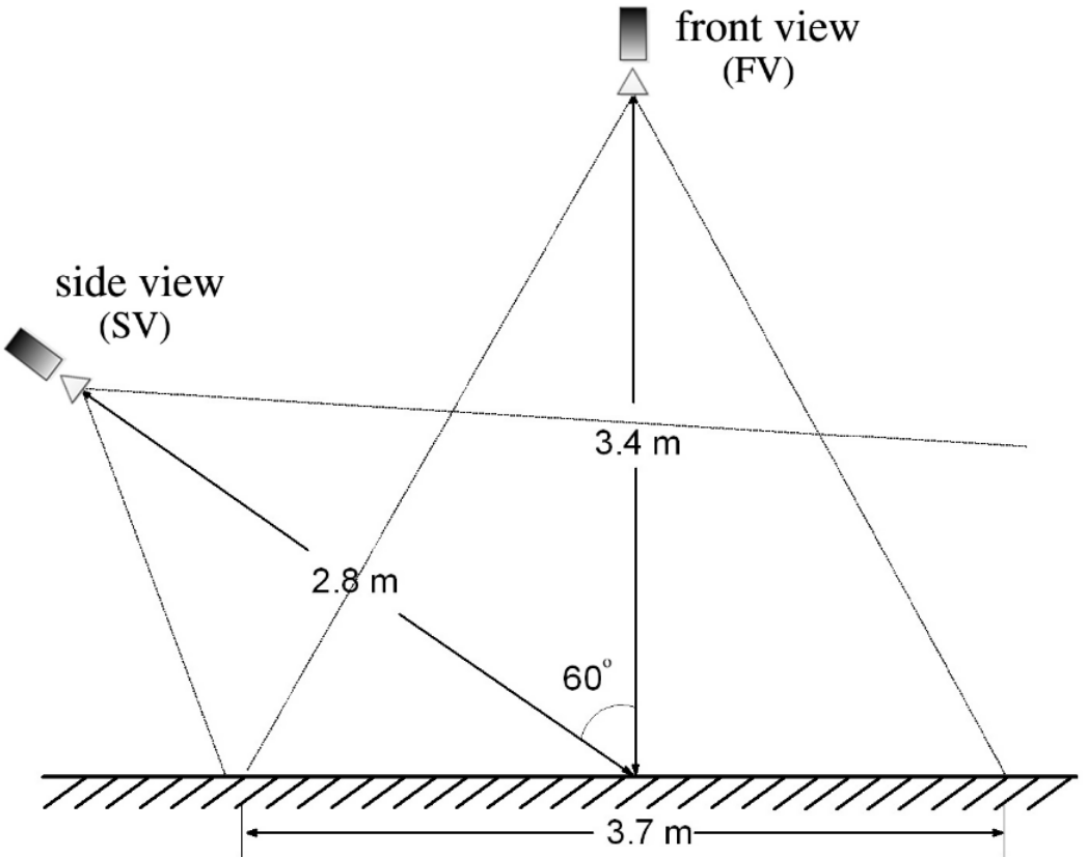


Figure 5.4: **Camera configuration of the M2I dataset [67].**

datasets.

5.6 Experimental setup

SBU Interaction dataset and M2I dataset The final results are averaged with regard to accuracy across the folds. The author extract joint distance and joint motion [7] for the spatial human-human relation and temporal human-human relation in S-RNN, respectively. Note that, the author only use 15 major joint locations (illustrated in Figure 4.4) to extract the relational features in the experiments to avoid the noise of other joints.

5.7 Experimental results

SBU Interaction dataset. The author summarize the results in terms of average classification accuracy in Table 5.1. In this table, only [7] and [13] focus on modeling spatial and temporal relationships in human-human interaction, while the rest of them are skeleton-based methods based on handcrafted features or Deep LSTM.

Method	Acc (%)
Joint Features with SVM classifier [7]	87.60
Joint Features with MILBoost classifier [7]	91.10
Poselet Dictionary [68]	86.90
Hierarchical RNN [41] (as reported in [30])	80.35
Deep LSTM + Co-occurrence + In-depth Dropout [30]	90.41
Spatio-Temporal LSTM with Trust Gates [32]	93.30
Active Joint Interaction Graph [13]	**94.12**
Structured S-LSTM (Proposed)	89.04

Table 5.1: **Human-human interaction recognition results on SBU-Kinect-Interaction dataset [7].** Showing average accuracy for human-human interaction recognition.

Table 5.1 shows that the proposed architecture of Deep LSTM achieves pretty competitive performance. However, these methods consider human-human interaction recognition task as choosing correct interaction label corresponding to two human skeletons. If there are multiple human subjects in the scene, the other methods would recognize two human subjects did some interaction together while they have done nothing. This happens because they focused only on the change human postures over the time without considering the relationship between two human subjects.

In addition, the skeleton data of both SBU and M2I datasets is quite noisy. The skeleton data also contains unrealistic human skeleton poses caused by the noise. All of the other LSTM-based methods applied the Svaitzky-Golay filter in temporal domain to reduce the influence of noise on skeleton data or used trust gate. However, in the experiments, the author compute the relational features from raw skeleton input data without applying any methods to the influence of noise. Therefore, the performance of the proposed method are greatly degraded.

Method	Acc (%)
Front View	
Joint Features with SVM classifier [7]	78.37
Joint Features with MILBoost classifier [7]	81.29
Active Joint Interaction Graph [13]	**88.35**
Structured S-LSTM (Proposed)	83.19
Side View	
Joint Features with SVM classifier [7]	77.16
Joint Features with MILBoost classifier [7]	79.43
Active Joint Interaction Graph [13]	**86.32**
Structured S-LSTM (Proposed)	83.75

Table 5.2: **Human-human interaction recognition results for single task scenario on M2I dataset [67].** Showing average accuracy for human-human interaction recognition.

M2I dataset. The author summarize the comparison in terms of average recognition accuracy in Table 5.2. The proposed method achieves 83.75% for side view and 83.19% for the front view, which is approximately 5% less than Active Joint Interaction Graph. However, this method is based on a complex graph model which is sensitive to the parameter settings. The number of temporal pyramid levels as well as the number of pairs of joints have to be chosen specifically to the problem. Figure 5.5 shows the confusion matrix to label human-human interaction for SBU and M2I datasets. Ground truth interaction labels are on rows and the predictions are on columns.

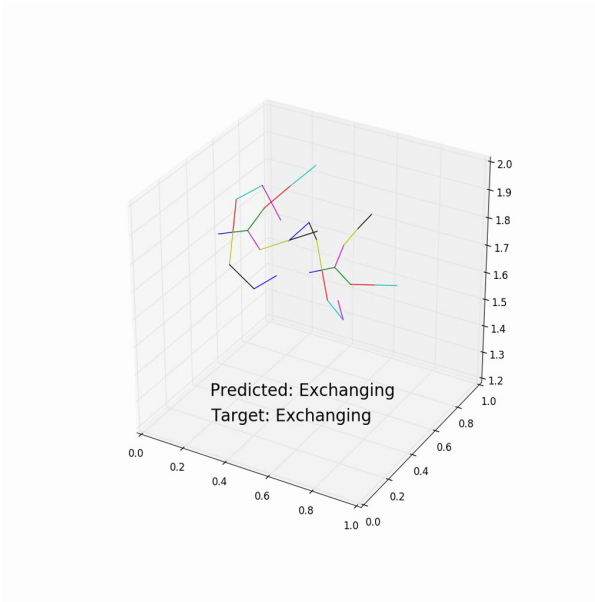
		a	b	c	d	e	f	g	h	i
Bow	a	0.86	0.01	0.02	0.01	0.02			0.06	0.02
Box	b		0.84		0.01	0.01	0.09	0.04		0.01
Chat	c			0.79	0.03		0.13		0.04	0.01
Cross	d		0.02	0.08	0.89					0.01
Handshake	e	0.01	0.01	0.03		0.84	0.04	0.01	0.06	
High Five	f		0.06	0.17		0.04	0.66	0.05	0.02	
Hug	g		0.01			0.04	0.03	0.92		
Wait	h	0.01		0.04	0.04	0.05	0.04	0.01	0.78	0.03
Walk Together	i	0.01	0.01				0.02		0.05	0.91

(a) Confusion matrix of the propsed method on M2I dataset.

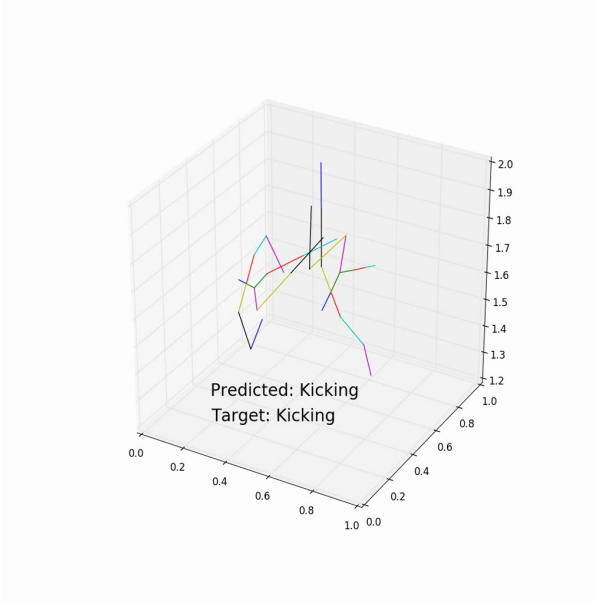
		a	b	c	d	e	f	g	h
Approaching	a	0.98							0.02
Departing	b		0.99						0.01
Kicking	c		0.02	0.87		0.01		0.05	0.05
Pushing	d				0.89		0.02		0.09
ShakingHands	e			0.01		0.80	0.01	0.17	0.01
Hugging	f	0.03			0.09		0.84		0.04
Exchanging	g			0.05		0.01		0.91	0.03
Punching	h	0.01	0.01	0.05	0.07	0.05		0.05	0.78

(b) Confusion matrix of the propsed method on SBU-Kinect-Interaction dataset.

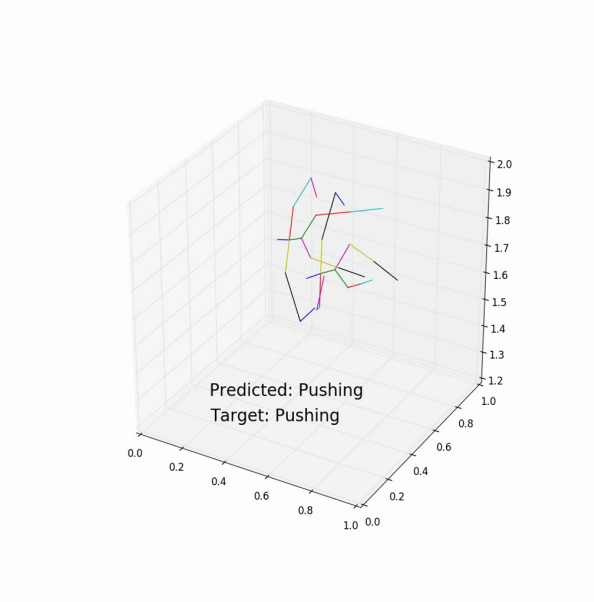
Figure 5.5: **Confusion matrix of the proposed method on Human-Human Interaction datasets.** Ground truth interaction labels are on rows and the predictions are on columns.



(a)



(b)



(c)

Figure 5.6: **Some examples of correct predictions on M2I dataset.**

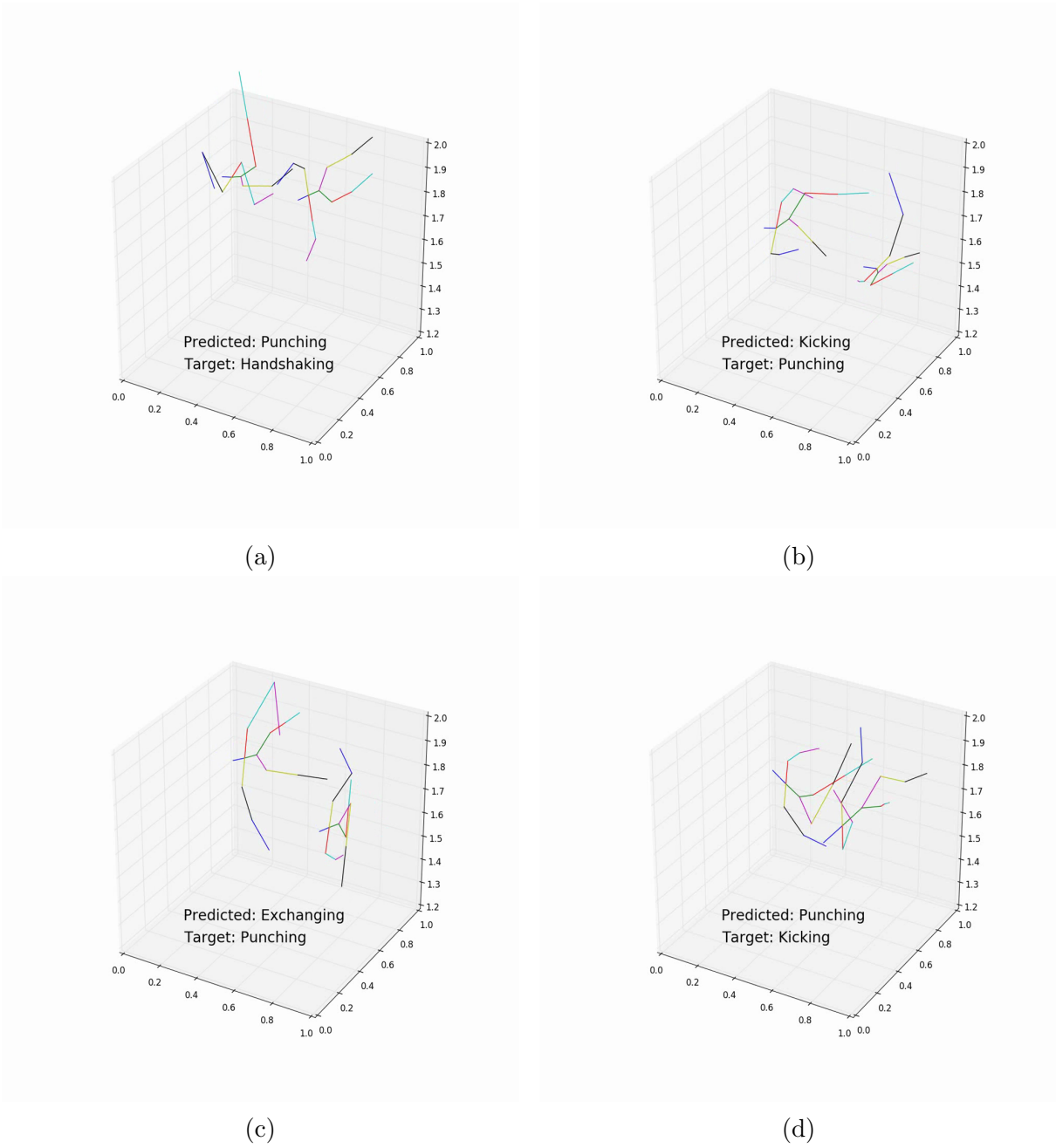


Figure 5.7: **Some examples of incorrect predictions on M2I dataset.**

Chapter 6

Dropout for S-RNN

6.1 Dropout

Deep neural networks contain multiple hidden layers with a large number of parameters. It helps deep neural nets learn complicated relationship between input and output. However, overfitting is a big problem Deep Neural Networks due to the large number of parameters. There are many methods to deal with this problem such as early stopping, regularization.

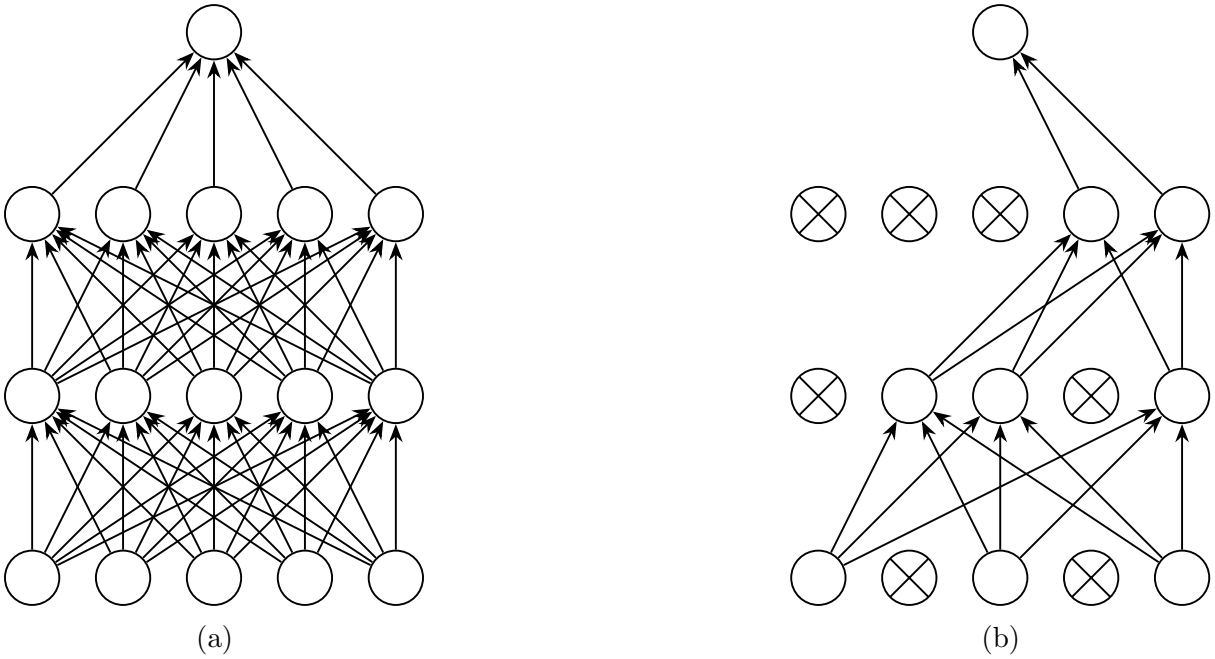
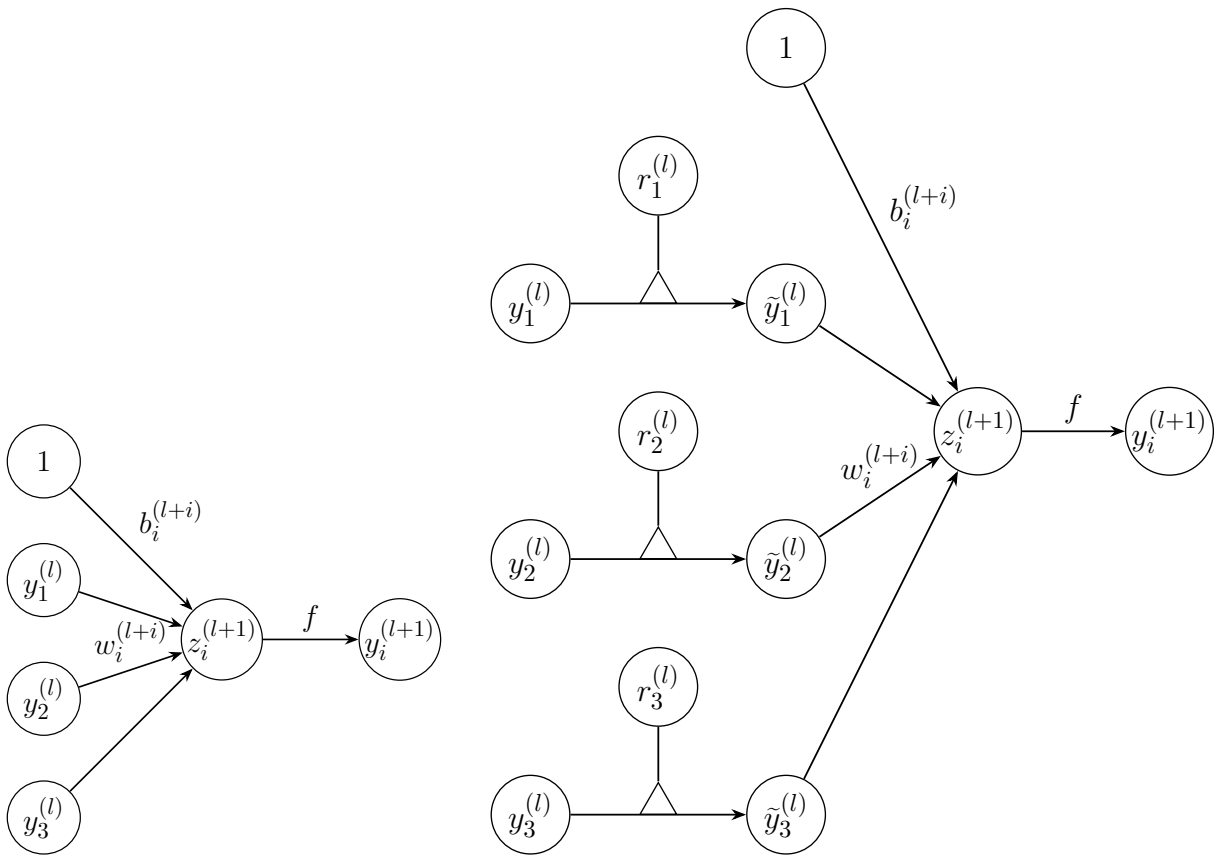


Figure 6.1: **An example of Dropout Neural Net Model** (a) A standard neural net. (b) An example of a thinned net produced by applying dropout to the network on the left. Crossed units have been temporarily removed from the network [37].

Dropout is a regularization technique proposed by Srivastava, et al. [37]. For each iteration during the training process, some neurons are randomly dropped out while the networks use the remaining neurons for predicting. In the testing phase, the dropout neural networks make predictions like standard neural network without dropping any neurons.



(a) Standard network (b) Dropout network

Figure 6.2: **Operations of a standard network and dropout network** [37].

Consider a neural network with L layers. Let $l \in \{1, ..., L\}$ denotes the index of the layer, $\mathbf{z}^{(l)}$ denotes the vector of inputs into layer l, and $\mathbf{y}^{(l)}$ denotes the vector of outputs from layer l. Note that $\mathbf{y}^{(0)}$ is the input vector of the network. $\mathbf{W}_i^{(l)}$ and $\mathbf{b}_i^{(l)}$ are the weight matrix and bias vector at unit i of layer l of the network. f is the activation function. The feedforward process of a standard neural network (Figure 6.2a) can be described as:

$$\mathbf{z}_i^{(l+1)} = \mathbf{w}_i^{(l)}\mathbf{y}^{(l)} + \mathbf{b}_i^{(l+1)} \tag{6.1}$$

$$y_i^{(l+1)} = f\left(\mathbf{z}_i^{(l+1)}\right) \tag{6.2}$$

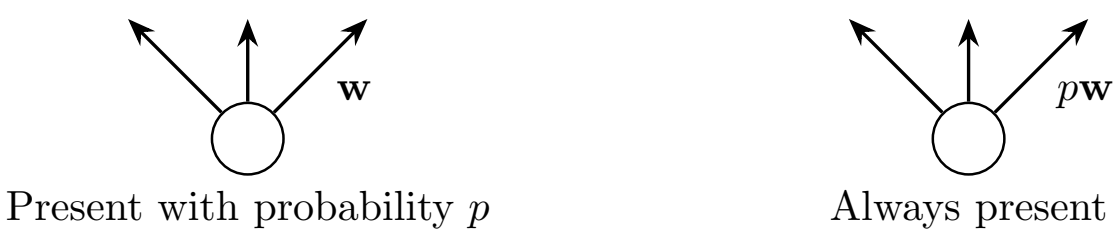
By applying dropout, the feedforward process can be redefined as (Figure 6.2b):

$$\mathbf{r}_i^{(l)} = Bernoulli(p) \tag{6.3}$$

$$\widetilde{\mathbf{y}}_i^{(l)} = \mathbf{r}_i^{(l)} * \mathbf{y}_i^{(l)} \tag{6.4}$$

$$\mathbf{z}_i^{(l+1)} = \mathbf{w}_i^{(l)}\widetilde{\mathbf{y}}^{(l)} + \mathbf{b}_i^{(l+1)} \tag{6.5}$$

where $*$ denotes element-wise (pointwise) product. For each unit i of layer l, $r_i^{(l)}$ is a vector of independent Bernoulli random variables. It takes value 1 with probability p and 0 with probability $1-p$. $r_i^{(l)}$ is sampled and computed the element-wise product with the outputs of corresponding layer $\mathbf{y}_i^{(l)}$ to create dropped output vector $\widetilde{\mathbf{y}}_i^{(l)}$ for each iteration during training. Note that, the weights of dropped neurons are not being updated during training phase. For testing, the weight of every unit in the network is scaled as $\mathbf{w}_{test} = p\mathbf{w}_{train}$ instead of using dropout mechanism (Figure 6.3).



Present with probability p

(a) Training time

Always present

(b) Testing time

Figure 6.3: **Operations of a standard network and dropout network** [37].

Because deep neural networks have a large number of parameters, they could fit the training data quickly. Thus, there could be many "unused" parameters which do not have a significant effect on the final decision of the network. By applying dropout, the contribution of "good" neurons is temporally removed at each iteration during training time. Therefore, the other neurons which have no positive effect on prediction could be improved. As a result of this process, a dropout neural network is the

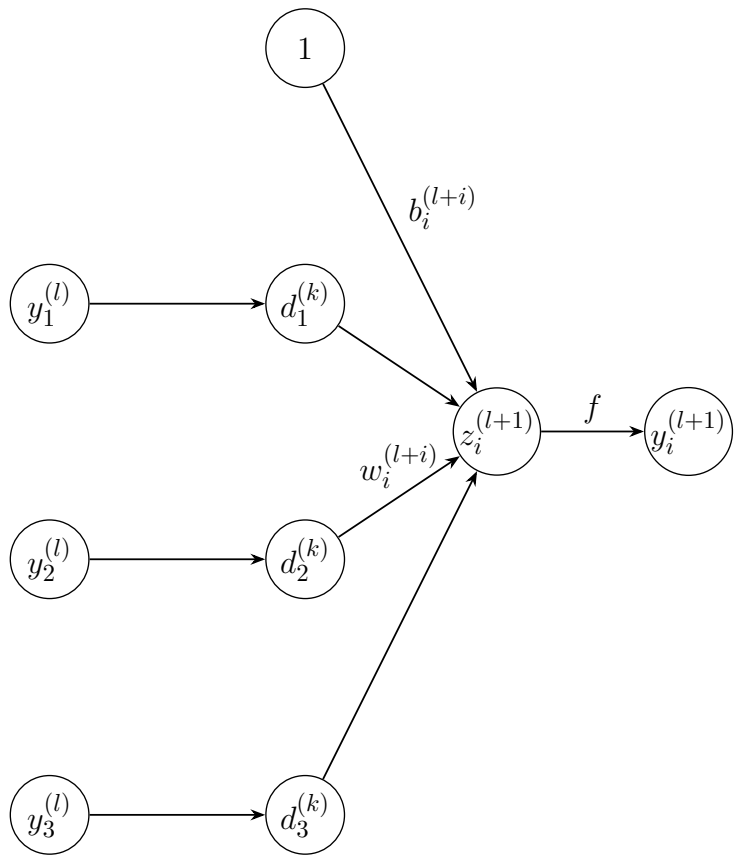


Figure 6.4: **Dropout layer** [37].

combination of multiple "thinned" neural networks. It means that the output of dropout neural networks is the average of the output of multiple thinned neural networks. This makes the neural networks have a better generalization.

In practice, we could add a equivalent dropout layer (Figure 6.4) after normal neural network hidden layers. The dropout unit $d_i^{(k)}$ could be defined as:

$$\begin{cases} d_i^{(k)} = Bernoulli(p) * \mathbf{y}_i^{(l)} \text{ if training} \\ d_i^{(k)} = p\mathbf{y}_i^{(l)} \text{ if testing} \end{cases} \tag{6.6}$$

6.2 Dropout for S-RNN

The number of spatio-temporal relations used to make the predictions in the proposed model is quite large. It is hard to optimize parameters of S-RNN. Therefore, the author utilize the dropout layers to the proposed model to boost the performance. By improving the generalization of the networks, the proposed method could also reduce the "bad" impact due to the noise of the skeleton data.

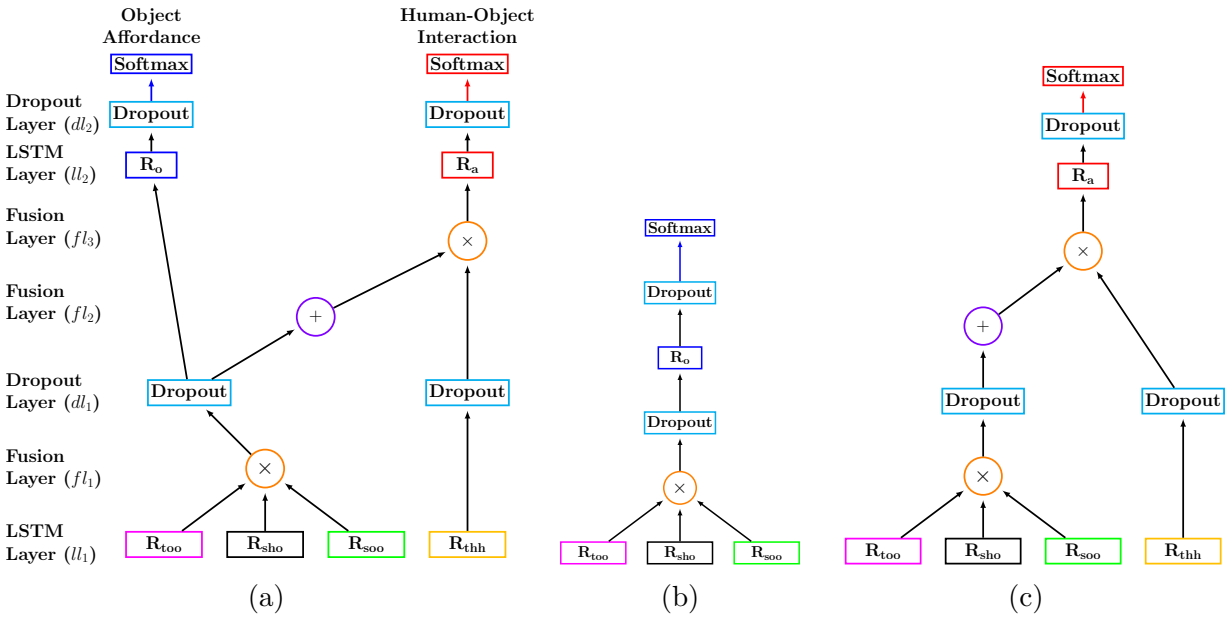


Figure 6.5: **The architecture of the proposed S-RNN with dropout layers for learning human-object interactions and object affordances.** (a) The general model for learning human-object interactions and object affordances. (b) The forward pass of object affordance node. (c) The forward pass of human activity node.

Human-object interaction recognition: the author decided to add two layers of dropout into the model, namely dl_1 and dl_2 (Figure 6.5). Every rectangular node in this model is a LSTM. The layer dl_1 consists of two dropout layers which are used for thinning the hidden representation of spatio-temporal relationship LSTM networks R_{too}, R_{sho}, R_{soo}, and R_{thh}. The layer dl_2 consists of two dropout layers which are used for thinning the activity LSTM networks R_a and object affordance LSTM networks R_o.

Human-human interaction recognition: the author decided to add two dropout layers into the proposed proposed model (Figure 6.6). dl_1 is used for thinning the hidden representation of spatio-temporal relationship

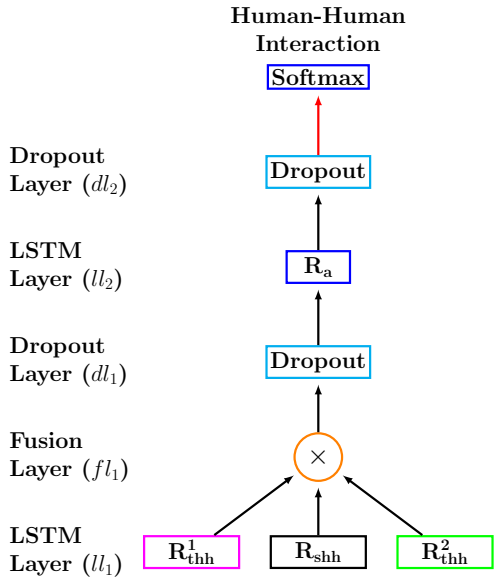


Figure 6.6: **The architecture of the proposed S-RNN with dropout layers for learning human-human interactions.**

LSTM networks R^1_{thh}, R_{shh}, and R^2_{thh}. dl_2 is used for thinning the hidden representation of activity LSTM networks R_a.

6.3 Experimental results

In this section, the author report the experimental results of the proposed method with dropout layer on all the dataset that were used in the previous chapters, namely CAD-120, M2I and SBU datasets.

Experimental results on CAD-120 dataset: Table 6.1 summarizes the results of comparative detection and anticipation of the proposed method compared with the state-of-the-art methods in terms of accurary averaged over all the classes.

The importance of dropout layers was shown in the improvement obtained by the proposed method in Table 6.1. The proposed method mostly outperforms with regard to both anticipation and detection when compared with previous works based on spatio-temporal relation by Jain *et al.* [45] and by Koppula *et al.* [12]. Especially, the accuracy of sub-activity detection and anticipation are improved as much as 10% and 17%, respectively, compared with the state of the art structured RNN for modeling

Method	Detection (%)		Anticipation (%)	
	Sub-activity	Object Affor-dance	Sub-activity	Object Affor-dance
Hu *et al.* [66]	87.0	N/A	N/A	N/A
Koppula *et al.* [12]	89.3	93.9	49.6	67.2
Jain *et al.* [45]	83.2	91.1	65.6	80.9
Taha *et al.* [28]	91.6	N/A	N/A	N/A
Structured S-LSTM (Proposed)	90.4	91.8	80.0	85.9
Structured S-LSTM + Dropout (Proposed)	**93.15**	**94.45**	**85.39**	**88.38**

Table 6.1: **Detection and anticipation results of the proposed method with dropout layer on CAD-120 [11].** Showing average accuracy for affordances, sub-activities.

Method	High-level Activity (%)
Koppula *et al.* [12]	93.5
Taha *et al.* [28]	94.4
Structured S-LSTM (Proposed)	96.4
Structured S-LSTM + Dropout (Proposed)	**97.6**

Table 6.2: **High-level Activity recognition results of the proposed method with dropout layer on CAD-120 [11].** Showing average accuracy for high-level activity.

spatio-temporal relations proposed by Jain *et al.* [45]. Figure 6.7 and Figure 6.8 shows confusion matrix to detect and anticipate sub-activity and object affordance labels for CAD-120 dataset. Ground truth interaction labels are on rows and the predictions are on columns.

Table 6.2 shows the comparison of the results of high-level activity recognition, where the proposed method achieves better performance than state-of-the-art methods. The dropout layers slightly improves the performance of the proposed method as much as 1%. The main reason this is the information of distance between objects and human subjects is not clear enough to conclude the role of objects (object affordance) in some human activity. For example, "scrubbable" and "srubber" are poorly recognized (Figure 6.7). Thus Because the proposed proposed method treats every object in the same way without considering its identity, it is hard to tell the

		a	b	c	d	e	f	g	h	i	j
Reaching	a	0.95	0.03					0.01		0.01	
Moving	b	0.02	0.95				0.01	0.01		0.01	
Pouring	c		0.01	0.99							
Eating	d	0.08	0.10		0.77	0.01				0.04	
Drinking	e		0.01			0.99					
Opening	f	0.03	0.08				0.89				
Placing	g	0.01		0.01				0.97		0.01	
Closing	h	0.03	0.02					0.01	0.94	0.01	
Null	i	0.04	0.03		0.01	0.01		0.02		0.89	
Scrubbing	j							0.03			0.97

(a) Confusion matrix of sub-activity detection of the propsed method with dropout layer on CAD-120 dataset.

		a	b	c	d	e	f	g	h	i	j	k	l
Movable	a	0.96	0.03	0.01									
Stationary	b		0.97	0.03									
Reachable	c	0.01	0.19	0.80									
Pourable	d	0.03			0.97								
Pourto	e		0.20			0.80							
Containable	f		0.14				0.86						
Drinkable	g	0.04	0.03		0.01			0.92					
Openable	h	0.03	0.13	0.02					0.82				
Placeable	i	0.02	0.03							0.95			
Closable	j	0.02	0.09	0.04							0.85		
Scrubbable	k		0.13								0.05	0.75	0.07
Srubber	l		0.05		0.02					0.23		0.10	0.60

(b) Confusion matrix of object affordance detection of the propsed method with dropout layer on CAD-120 dataset.

Figure 6.7: **Confusion matrix of detection task of the propsed method with dropout layer on CAD-120 dataset.** Ground truth activity labels are on rows and the predictions are on columns.

difference between "scrubbable" and "stationary". Besides, the movements of "srubber" are usually too small to differentiate them from noise.

Experimental results on SBU dataset: Table 6.3 shows that the proposed baseline architecture of Deep LSTM without Dropout achieves pretty comparable performance. When the Dropout layers are applied, the recognition accuracy of proposed S-RNN is significantly improved as much as 5% and outperforms all the other methods.

Though the author use two joint motion and distance from [7], the proposed method has better performance on both SBU Interaction dataset

		a	b	c	d	e	f	g	h	i	j
Reaching	a	0.87	0.09		0.01		0.01	0.01		0.01	
Moving	b	0.04	0.92		0.01		0.01	0.01		0.01	
Pouring	c		0.02	0.83				0.15			
Eating	d	0.11	0.43		0.38	0.02		0.01		0.05	
Drinking	e		0.07		0.03	0.79		0.08		0.03	
Opening	f	0.05	0.08				0.87				
Placing	g	0.01	0.03					0.92	0.01	0.03	
Closing	h	0.06	0.07			0.01		0.01	0.78	0.06	
Null	i	0.15	0.25		0.03	0.02		0.09		0.46	
Scrubbing	j	0.02	0.03					0.52		0.03	0.40

(a) Confusion matrix of sub-activity anticipation of the propsed method with dropout layer on CAD-120 dataset.

		a	b	c	d	e	f	g	h	i	j	k	l
Movable	a	0.76	0.21	0.02						0.01			
Stationary	b	0.02	0.95	0.02						0.01			
Reachable	c	0.03	0.34	0.62					0.01				
Pourable	d	0.07			0.81			0.03		0.09			
Pourto	e		0.70			0.29				0.01			
Containable	f		0.30				0.70						
Drinkable	g	0.10	0.16					0.56		0.18			
Openable	h	0.04	0.12	0.03					0.80				
Placeable	i	0.02	0.10							0.88			
Closable	j		0.44	0.03							0.53		
Scrubbable	k		0.42	0.02			0.02					0.51	0.03
Srubber	l	0.02	0.25		0.02					0.23			0.46

(b) Confusion matrix of object affordance anticipation of the propsed method with dropout layer on CAD-120 dataset.

Figure 6.8: **Confusion matrix of detection task of the propsed method with dropout layer on CAD-120 dataset.** Ground truth activity labels are on rows and the predictions are on columns.

and M2I dataset. That is because Deep LSTM is much more effective to handle the sequential problem like human-human interaction.

Experimental results on M2I dataset: The author summarize the comparison in terms of average recognition accuracy in Table 6.4. The performance of S-RNN with dropout layer is improved as much as 4%. the proposed method achieves 87.36% for side view and 87.78% for the front view, respectively, which is competitive to Active Joint Interaction Graph. However, this method is based on a complex graph model which is sensitive to the parameter settings (the number of temporal pyramid levels and the

		a	b	c	d	e	f	g	h	i	k
Arranging objects	a	1.0									
Cleaning objects	b		0.92						0.02	0.07	
Having a meal	c			1.0							
Microwaving food	d				1.0						
Making cereal	e					1.0					
Picking objects	f						1.0				
Stacking objects	g					0.07		0.93			
Taking food	h		0.03			0.02			0.95		
Taking medicine	i									1.0	
Unstacking objects	k					0.04					0.96

Figure 6.9: Confusion matrix of high-level activity recognition of the propsed method with dropout layer on CAD-120 dataset.

Method	Acc (%)
Joint Features with SVM classifier [7]	87.60
Joint Features with MILBoost classifier [7]	91.10
Poselet Dictionary [68]	86.90
Hierarchical RNN [41] (as reported in [30])	80.35
Deep LSTM + Co-occurrence + In-depth Dropout [30]	90.41
Spatio-Temporal LSTM with Trust Gates [32]	93.30
Active Joint Interaction Graph [13]	94.12
Structured S-LSTM (Proposed)	89.04
Structured S-LSTM + Dropout (Proposed)	**94.91**

Table 6.3: **Human-human interaction recognition results on SBU-Kinect-Interaction dataset [7].** Showing average accuracy for human-human interaction recognition.

number of pairs of joints).

Despite this improvements, the proposed method still have trouble to discriminate activities with similar skeleton poses such as "push" and "punch". Moreover, the skeleton data of complex activities are quite noisy. Thus, it is extremely hard to recognize complex activities like "box". Figure 6.10 shows confusion matrix to label human interaction for SBU and M2I datasets. Ground truth interaction labels are on rows and the predictions are on columns.

Method	Acc (%)
Front View	
Joint Features with SVM classifier [7]	78.37
Joint Features with MILBoost classifier [7]	81.29
Active Joint Interaction Graph [13]	**88.35**
Structured S-LSTM (Proposed)	83.19
Structured S-LSTM + Dropout (Proposed)	87.78
Side View	
Joint Features with SVM classifier [7]	77.16
Joint Features with MILBoost classifier [7]	79.43
Active Joint Interaction Graph [13]	86.32
Structured S-LSTM (Proposed)	83.75
Structured S-LSTM + Dropout (Proposed)	**87.36**

Table 6.4: **Human-human interaction recognition results of the propsed method with dropout layer on M2I dataset [67].** Showing average accuracy for human-human interaction recognition.

		a	b	c	d	e	f	g	h	i
Bow	a	0.89		0.03	0.01	0.02			0.03	0.02
Box	b		0.85			0.01	0.08	0.05		0.01
Chat	c		0.01	0.74	0.09		0.12		0.04	
Cross	d		0.01	0.03	0.95	0.01				
Handshake	e	0.01		0.02		0.94	0.01		0.02	
High Five	f		0.04	0.12	0.01	0.01	0.76	0.01	0.05	
Hug	g						0.01	0.99		
Wait	h	0.01		0.03					0.93	0.03
Walk Together	i	0.01	0.01				0.01		0.05	0.92

(a) Confusion matrix of the propsed method with dropout layer on M2I dataset.

		a	b	c	d	e	f	g	h
Approaching	a	0.97		0.01					0.02
Departing	b		1.00						
Kicking	c		0.01	0.96		0.01		0.01	0.02
Pushing	d				0.93		0.02		0.05
ShakingHands	e					0.96		0.04	
Hugging	f	0.02			0.05		0.93		
Exchanging	g			0.01		0.03		0.94	0.02
Punching	h			0.01	0.06	0.02		0.02	0.89

(b) Confusion matrix of the propsed method with dropout layer on SBU-Kinect-Interaction dataset.

Figure 6.10: **Confusion matrix of the proposed method with dropout layer on Human-Human Interaction datasets.** Ground truth interaction labels are on rows and the predictions are on columns.

Chapter 7

Conclusion and Future Work

In this work, the author proposed a novel Structured Recurrent Neural Network to model spatio-temporal relationships among human subjects and objects in daily human interactions. The author represent the evolution of different components and the relationships between them over the time by several subnets. Then, the hidden representations of those relations are fused and fed into the later layers to obtain the final hidden representation. The final prediction is carried out by the single-layer perceptron. The author evaluated the proposed method with the same parameter settings on the various challenging datasets composed of different human-human interactions, human-object interactions and object affordances. The proposed method is well-performed on various tasks and datasets.

The following summarizes the main contributions of this work and the observation derived from the experiments. The author also discuss future research directions to cope with the remaining problems of this work on interaction recognition.

7.1 Human-Object Interaction Recognition and Object Affordance Recognition using Structured Recurrent Neural Network

In this work, the author presented a novel structural RNNs for modeling spatio-temporal relationship among human subject and objects in human-object interaction. The author also investigated the effectiveness of object-object relation for recognizing human-object interactions. By taking this

relationship into account, it enabled the proposed S-RNN to model different characteristic of human poses, human-object relations, the natural hierarchy of human activities, and the way the human activity would affect the state of objects and the relationship between objects.

7.2 Human-Human Interaction Recognition using Structured Recurrent Neural Network

For human-Human interaction recognition, the author proposed a novel structural RNNs for modeling spatio-temporal relationship between human subjects. The proposed method achieved a competitive result to other state-of-the-art methods on M2I and SBU datasets. The skeletal data are not reliable due to noise and occlusion. The performance of the proposed method is greatly degraded. Developing a better way of spatio-temporal relationship feature directly from input data is one of a possible direction to improve the proposed method in the future.

7.3 Dropout Layer

Because the number of spatio-temporal relations used to make the predictions in the model is quite large, it is hard to optimize parameters of S-RNN. The author applied dropout technique to enhance the generalization of the proposed network, as well as to effectively learn important spatio-temporal relationships for human interaction.

7.4 Open issues

Due to the noise of human skeleton, the proposed method could not recognize the human activity in many cases. Besides, the lack of visual appearance also make the proposed method hard to discriminate some human interactions with similar human poses. In order to improve the performance, the author would like to investigate different object detection methods and spatio-temporal relationship feature extraction methods.

Currently, the proposed method is limited to human-object interactions and human-human interactions from RGB-D data, where the depth information is available to estimate 3D human poses. In the future, the author also would like to extend the current structured Deep RNN architecture for more complex scenarios, such as multiple human subjects interact with different objects from RGB data.

Acknowledgement

First of all, I would like to express my sincere gratitude to Associate Professor Atsuo Yoshitaka, my supervisor, for his patience, motivation, enthusiasm, and immense knowledge. His encouragement and guidance helped me in all the time of research and writing of this thesis.

Additionaly, I am greatly indebted to my friends, Sorn Sooksatra, who have helped me a lot in both research and daily life at JAIST. I would also like to thank my fellow labmates in my laboratory for the stimulating discussions.

For my family who always there and believe in everything I do, thank you so much.

February 2018
Japan Advanced Institute of Science and Technology, Japan

References

[1] J. Luo, W. Wang, and H. Qi, "Group sparsity and geometry constrained dictionary learning for action recognition from depth maps," in *2013 IEEE International Conference on Computer Vision*, pp. 1809–1816, 2013.

[2] J. Imran and P. Kumar, "Human action recognition using rgb-d sensor and deep convolutional neural networks," in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 144–148, 2016.

[3] J. Aggarwal and M. Ryoo, "Human activity analysis: A review," *ACM Comput. Surv.*, vol. 43, pp. 16:1–16:43, Apr. 2011.

[4] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3d points," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pp. 9–14, 2010.

[5] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu, "Robust 3d action recognition with random occupancy patterns," in *Computer Vision – ECCV 2012*, pp. 872–885, Springer Berlin Heidelberg, 2012.

[6] X. Yang, C. Zhang, and Y. Tian, "Recognizing actions using depth motion maps-based histograms of oriented gradients," in *Proceedings of the 20th ACM International Conference on Multimedia*, MM '12, pp. 1057–1060, 2012.

[7] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, IEEE, 2012.

[8] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from rgbd images," in *International Conference on Robotics and Automation (ICRA)*, 2012.

[9] B. Ni, Y. Pei, P. Moulin, and S. Yan, "Multilevel depth and image fusion for human activity detection," *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1383–1394, 2013.

[10] R. Gupta, A. Y.-S. Chia, and D. Rajan, "Human activities recognition using depth images," in *Proceedings of the 21st ACM International Conference on Multimedia*, MM '13, pp. 283–292, 2013.

[11] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *Int. J. Rob. Res.*, vol. 32, no. 8, pp. 951–970, 2013.

[12] H. S. Koppula and A. Saxena, "Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation," in *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, pp. 792–800, 2013.

[13] M. Li and H. Leung, "Multiview skeletal interaction recognition using active joint interaction graph," *IEEE Transactions on Multimedia*, vol. 18, no. 11, pp. 2293–2302, 2016.

[14] M. Ye, *A Survey on Human Motion Analysis from Depth Data*, pp. 149–187. Springer Berlin Heidelberg, 2013.

[15] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6517–6525, 2017.

[16] A. W. Vieira, E. R. Nascimento, G. L. Oliveira, Z. Liu, and M. F. M. Campos, "Stop: Space-time occupancy patterns for 3d action recognition from depth map sequences," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pp. 252–259, Springer Berlin Heidelberg, 2012.

[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pp. 886–893 vol. 1, 2005.

[18] O. Oreifej and Z. Liu, "Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 716–723, 2013.

[19] X. Yang and Y. Tian, "Super normal vector for activity recognition using depth sequences," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 804–811, 2014.

[20] L. Xia and J. K. Aggarwal, "Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2834–2841, 2013.

[21] Y. Song, J. Tang, F. Liu, and S. Yan, "Body surface context: A new robust feature for action recognition from depth videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 6, pp. 952–964, 2014.

[22] H. Rahmani, A. Mahmood, D. Q Huynh, and A. Mian, "Hopc: Histogram of oriented principal components of 3d pointclouds for action recognition," in *Computer Vision – ECCV 2014*, (Cham), pp. 742–757, Springer International Publishing, 2014.

[23] H. Rahmani, A. Mahmood, D. Huynh, and A. Mian, "Histogram of oriented principal components for cross-view action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 12, pp. 2430–2443, 2016.

[24] H. Rahmani and A. Mian, "3d action recognition from novel viewpoints," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1506–1515, 2016.

[25] L. Xia, C. C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *2012 IEEE Com-*

puter Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 20–27, 2012.

[26] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Human activity detection from rgbd images," in *Proceedings of the 16th AAAI Conference on Plan, Activity, and Intent Recognition*, pp. 47–55, 2011.

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

[28] A. Taha, H. H. Zayed, M. E. Khalifa, and E.-S. M. El-Horbaty, "Skeleton-based human activity recognition for video surveillance," *International Journal of Scientific & Engineering Research*, vol. 6, no. 1, 2015.

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

[30] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 3697–3703, 2016.

[31] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+d: A large scale dataset for 3d human activity analysis," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[32] 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 (ECCV)*, pp. 816–833, 2016.

[33] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1290–1297, 2012.

[34] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3d skeletons as points in a lie group," in

2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 588–595, 2014.

[35] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pp. II–1764–II–1772, 2014.

[36] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pp. 4346–4354, 2015.

[37] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, (USA), pp. 1097–1105, Curran Associates Inc., 2012.

[39] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[40] K. Wang, X. Wang, L. Lin, M. Wang, and W. Zuo, "3d human activity recognition with reconfigurable convolutional neural networks," in *Proceedings of the 22Nd ACM International Conference on Multimedia*, MM '14, pp. 97–106, 2014.

[41] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1110–1118, 2015.

[42] M. S. Ryoo and J. K. Aggarwal, "Hierarchical recognition of human activities interacting with objects," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

[43] U. Akdemir, P. Turaga, and R. Chellappa, "An ontology based approach for activity recognition from video," in *Proceedings of the 16th ACM International Conference on Multimedia*, pp. 709–712, 2008.

[44] A. Prest, C. Schmid, and V. Ferrari, "Weakly supervised learning of interactions between humans and objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 601–614, 2012.

[45] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5308–5317, 2016.

[46] L. Fausett, *Fundamentals of Neural Networks*. Prentice Hall, Englewood Cliffs, NJ, 1994.

[47] C. L. Giles, S. Lawrence, and A. C. Tsoi, "Rule inference for financial prediction using recurrent neural networks," in *Proceedings of the IEEE/IAFE 1997 Computational Intelligence for Financial Engineering (CIFEr)*, pp. 253–259, 1997.

[48] E. W. Saad, T. P. Caudell, and D. C. Wunsch, "Predictive head tracking for virtual reality," in *Neural Networks, 1999. IJCNN '99. International Joint Conference on*, vol. 6, pp. 3933–3936 vol.6, 1999.

[49] M. Auli, M. Galley, C. Quirk, and G. Zweig, "Joint language and translation modeling with recurrent neural networks," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, (Seattle, Washington, USA), pp. 1044–1054, Association for Computational Linguistics, 2013.

[50] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei,

eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 2048–2057, PMLR, 2015.

[51] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, vol. 385 of *Studies in Computational Intelligence*. Springer, 2012.

[52] H. G. Zimmermann, R. Grothmann, A. M. Schaefer, and Ch, "Identification and forecasting of large dynamical systems by dynamical consistent neural networks," in *New Directions in Statistical Signal Processing: From Systems to Brain* (S. Haykin, J. Principe, T. Sejnowski, and J. Mcwhirter, eds.), pp. 203–242, MIT Press, 2006.

[53] A. J. Robinson and F. Fallside, "The utility driven dynamic error propagation network," tech. rep., Engineering Department, Cambridge University, Cambridge, UK, 1987.

[54] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[55] R. J. Williams and D. Zipser, "Backpropagation," ch. Gradient-based Learning Algorithms for Recurrent Networks and Their Computational Complexity, pp. 433–486, L. Erlbaum Associates Inc., 1995.

[56] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[57] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[58] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International Conference on Machine Learning*, pp. 1310–1318, 2013.

[59] J. Freeman, "The modelling of spatial relations," *Computer Graphics and Image Processing*, vol. 4, no. 2, pp. 156 – 171, 1975.

[60] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 14–29, 2016.

[61] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, 2011.

[62] O. Pele and M. Werman, "A linear time histogram metric for improved sift matching," in *Proceedings of the 10th European Conference on Computer Vision: Part III*, pp. 495–508, 2008.

[63] J. Gibson, *The ecological approach to visual perception.* Houghton Mifflin, 1979.

[64] S. Dieleman, J. Schlüter, C. Raffel, *et al.*, "Lasagne: First release.," 2015.

[65] Theano Development Team, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv e-prints*, vol. abs/1605.02688, 2016.

[66] N. Hu, G. Englebienne, Z. Lou, and B. Krse, "Learning latent structure for activity recognition," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1048–1053, 2014.

[67] N. Xu, A. Liu, W. Nie, Y. Wong, F. Li, and Y. Su, "Multi-modal & multi-view & interactive benchmark dataset for human action recognition," in *Proceedings of the 23th International Conference on Multimedia*, 2015.

[68] Y. Ji, G. Ye, and H. Cheng, "Interactive body part contrast mining for human interaction recognition," in *2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, pp. 1–6, 2014.

Publications

[1] A. M. Truong and A. Yoshitaka, "Structured LSTM for human-object interaction detection and anticipation," 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Lecce, Italy, 2017, pp. 1-6.