UNIVERSITY OF CALIFORNIA
Los Angeles

Learning Interaction Grammar for Action Recognition

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Computer Science

by

Tengyu Liu

2018

ABSTRACT OF THE THESIS

Learning Interaction Grammar for Action Recognition

by

Tengyu Liu
Master of Science in Computer Science
University of California, Los Angeles, 2018
Professor Song-Chun Zhu, Chair

Video action recognition has been in the center of the stage since its introduction in 2004 [SLC04]. During the past 15 years, countless methods had been proposed to understand what human is doing in a video clip. While some works infer action label directly from pixel information, some other works propose to learn multi-level hierarchical structure that composes actions. In this work, we propose the learning of a two-layer grammar model for action recognition that is based on human object interaction. To evaluate the idea, we proposed an interaction grammar for action recognition on video. The model is evaluated by simulated annealing with MCMC and by deep learning.

The thesis of Tengyu Liu is approved.

Kai-Wei Chang

Ying-Nian Wu

Song-Chun Zhu, Committee Chair

University of California, Los Angeles

2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

Over the recent years video has become more popular than ever due to the increase of internet bandwidth and personal storage size. While state of the art algorithms can already do image classification at human level performance [HZR15], it is still an open problem for computer vision to understand videos. An important aspect of video understanding is to recognize actions being performed by human in video clips. With the exploding volume of videos from both online platforms and surveillance cameras, it is becoming impossible to analyze video streams by human and the ability to recognize actions is becoming important.

State of the art action recognition algorithms generally fall in two categories. Many of them try to analyze video as a whole by applying feature extraction tools such as histogram of gradient (HoG) or convolutional neural networks (CNNs) to each frame, extracting features at each frame and train a classifier on top of the features [SZ14a,FPZ16,TBF15]. More recent approaches, on the other hand, try to exploit the structure of human action by dissecting human action into smaller parts, and try identifying the parts separately [LZR15, LNS16]. The second approach had demonstrated better result than the first approach [LZR15,LNS16], but they require careful design of action composition, detailed annotation of data and require prior knowledge of actions.

In this thesis, we argue that the action grammar is a two-layer grammar that is based on interactions. We designed a stochastic attributed spatial temporal grammar model to illustrate the idea (as shown in Figure 1.1). With the notion of human-object interaction, not only can we improve action recognition, we can also estimate the 3D location of an object that is interacting with human (given by associating interaction label with human pose). In this paper, we implemented and evaluated two methods of parsing video according

our proposed grammar and got satisfactory result from both methods.

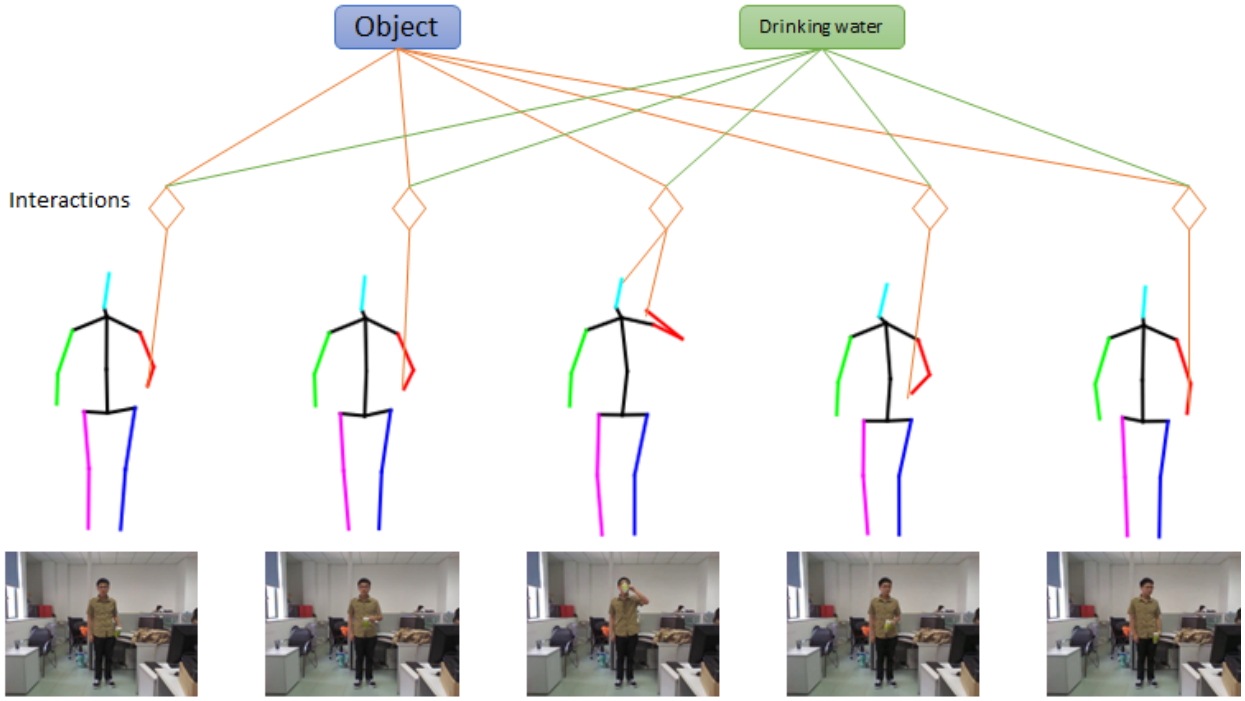

Figure 1.1: Example parse graph parsed from our attributed spatial temporal grammar

In the scope of this work, we will only focus on physical human object interaction, leaving non-physical interaction such as 'look-at' and human human interaction such as 'shake-hand' to my collegues in VCLA.

CHAPTER 2

Literature Review

This chapter is organized in three sections. In section 2.1 we will discuss related work in action recognition in general, and in section 2.2 we will introduce works related to modeling human object interaction. In section 2.3 we will discuss the use of grammar-based models in computer vision tasks.

2.1 Action Recognition

Action recognition on video has been exploited in many ways over the past 15 years since the release of the KTH dataset in 2004 [SLC04]. Over the past 15 years, many research have been devoted to the task of human action recognition. We classify them into three categories according to their structuredness.

The first category tries to understand videos by applying models directly on unstructured input, whether the input is raw image sequence [TBF15] or extracted features such as HoG [McC86], dence trajectory [WKS11] or optic flow [SZ14a]. Before deep learning took over computer vision community, the art of designing quality features had been a heavy part of CV research. An example of such features is the dense trajectory introduced in [WKS11]. Dense trajectory is a method of tracking dense points of interest over time and keep the trajectory as a feature for action recognition. Figure 2.1 illustrates the difference between results from dense trajectory and results from earlier tracking method (KLT [LK81]).

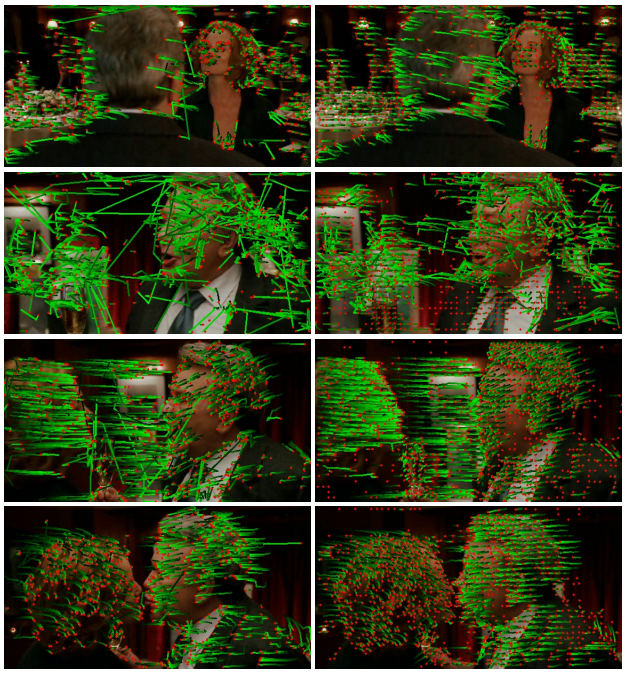

Figure 2.1: Illustration of dense trajectory [WKS11]

In recent years, deep learning and convolutional neural networks (CNNs) have took over the field. Since neural networks can learn feature from large annotated dataset, the research in computer vision had gradually moved towards learning a good feature extractor and designing a good classifier. In [SZ14b], the authors introduced a stacking of convolutional networks to very deep networks (VGG16) to extract features from images. One major contribution of this work is making the point that deeper networks are better at feature extraction but are harder to train. In [HZR16] the authors proposed a novel network architecture (ResNet) that pushed the depth from 16 to 3-digits. With the novel design of residual blocks, ResNet were able to learn even better features from images and remained reasonable training complexity.

Another category for works in action recognition introduces structure in the input data of the model. One very common way of introducing structure is to identify human pose in each frame. The poses are then fed into an optional layer of feature extractor before entering a classifier. In [WW17], the authors employed a two-stream recurrent neural network to classify action purely from human pose. In this work, the authors arranged pose vector of

size p across t frames as a matrix P of size $p \times t$, and fed P and P^T into two streams of recurrent neural networks before it is joined to make final prediciton. Figure 2.2 illustrates the idea. In addition to introducing pose as raw coordinates, some works pushed even further by proposing structured representation of human pose for action recognition. [VAC14] represented human pose as a Lie group which resembles a combination of many special Euclidean groups. The Lie group representation transforms poses in coordinates of joints to parameterizations of rigid body parts, and represents pose as a combination of rotation and translation between each body part. Using the Lie group representation, the authors were able to achieve better result then methods using raw joint coordinates. In addition to pose representations, many research effort had also been put in learning algorithms that are defined specifically for human pose sequences. An example is the [WWY17] where a naive-Bayes nearest neighbor method was proposed for action recognition on pose sequences.

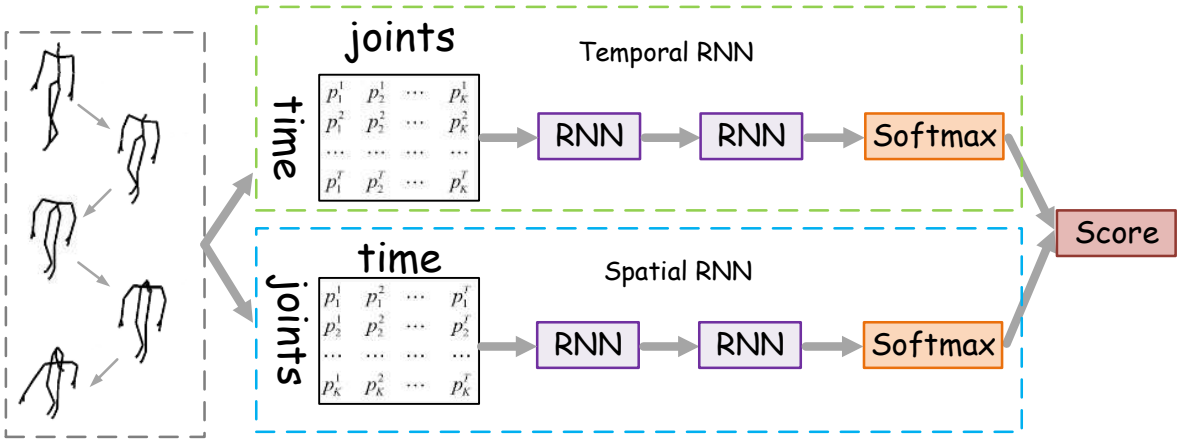

Figure 2.2: Illustration of a two stream recurrent neural network proposed in [WW17]

Some researchers put even more effort in introducing structure to action recognition by decomposing actions into smaller components. People gave them different names such as actionlets [LNS16], motionlet [WYH16] and mid-level elements [LZR15]. In [LNS16], an activity is decomposed into atomic actions, which is then decomposed into actionlets and then into motion poselets. Figure 2.3 illustrates the structure. In the proposed structure, each level in the hierarchy is represented by a combination of its descendent, and contributes

to the recognition of its parent. By introducing structure into actions, the authors were able to do action recognition at a more fine-grained level. Namely, [LNS16] was able to localize human actions in space and time.

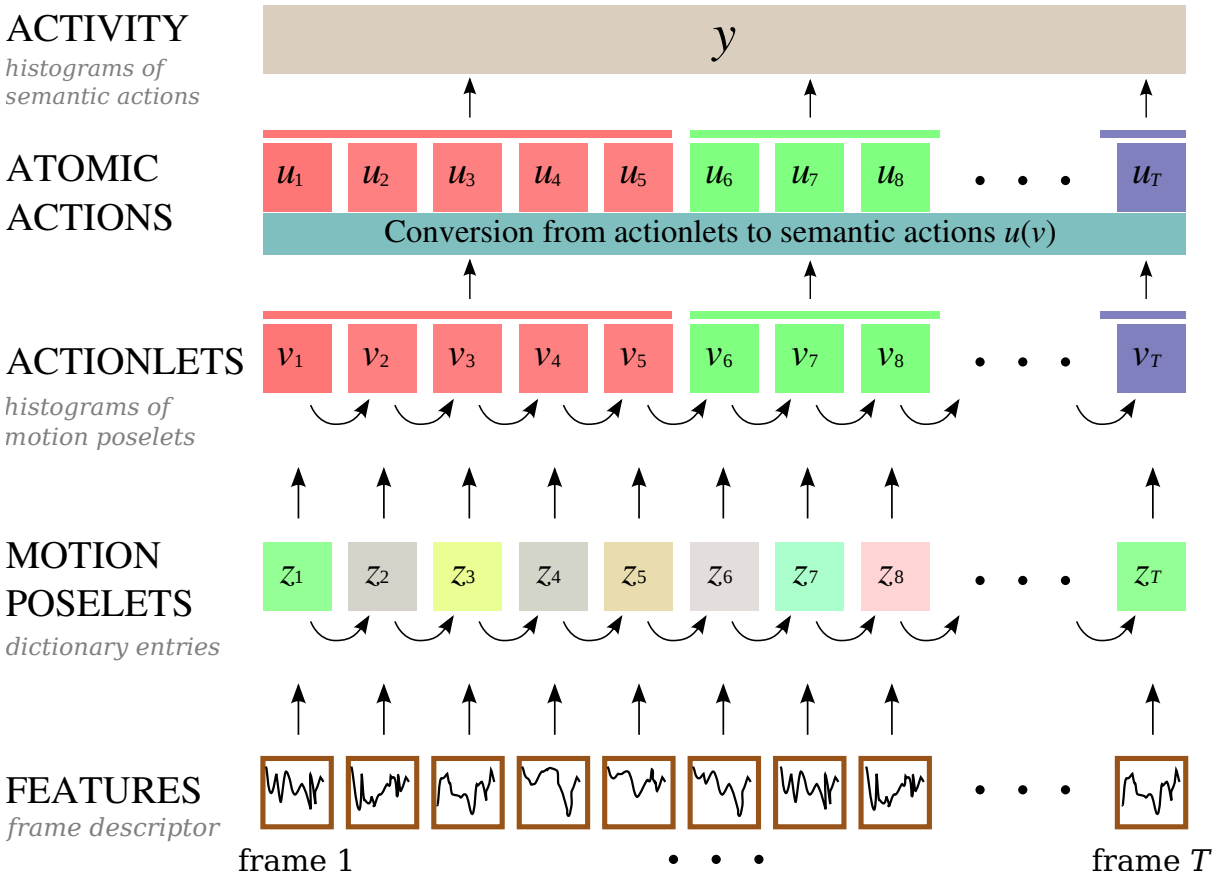

Figure 2.3: Illustration of action hierarchy described in [LNS16]

[WYH16] defined the structure of action as an undirected complete graph where each node is a motionlet. In this paper the structure of video is encoded in the labels of nodes and edges, where label on nodes define the joint and motion direction of detected motion let, and label on edges define the spatial and temporal relationship between motionlets. This paper proposed a subgraph-pattern graph kernel to measure similarity between graphs in order to classify the graph representation of videos.

Many recent approaches also introduces some sense of structure in their model implicitly. In [GM15], the authors proposed a tubed action detection model that implicitly models

the structuredness in the information contained in video. The model first proposes human action detection proposals in each frame in a two stream manner where one stream makes prediction from rgb images and the other stream from motion cues, given by optical flows. The proposed bounding boxes are then connected in a temporal way to see if there is a smooth movement across frames. Only smooth tubes are then fed into a final prediction. This method exploited the fact that most information in video is concentrated in a small region.

Another method that exploited structure in video for action recognition is [JS14]. In this work, the authors proposed a Dual Assignment K-Means clustering algorithm that can perform two co-occuring clustering tasks simultaneously and exploit the relationship between the two spaces. Using the proposed algorithm, the authors were able to cluster videos based on action and also contextual information and hence achieving action recognition.

2.2 Human Object Interaction

Modeling human object interaction (HOI) has been studied by many different scholars for various purposes since it was proposed by [FR08]. In [GKD09], Gupta and Davis stated that using HOI information can help understanding of action. [IS10] implicitly incorporated HOI in their action recognition model by recognizing person, object and scene in turn. Other methods [HBS09, KS03] have also implicitly used HOI formulated as contextual information in their study of action recognition. [DSL11, DRF10, YF10] studied various ways of recognizing HOI in still images.

Most effort put in human object interaction had remained in still images. This is partly because of the difficulty in processing video, but it also has its root in the lack of annotated and unannotated video. It was only until recent years that [WZZ13a, WZZ17] presented a stochastic hierarchical graph to model the human-object spatial relationship in RGB-D videos. In this work Wei et al. also introduced a generalized dynamic programming beam search algorithm to find the optimal understanding of video. [WZZ13b] also explored structure in concurrent human activities. [WZZ13b] explored spatial structure in human actions

where human action are compositions of informative body parts, which is essentially just human object interactions. Being able to recognize action as a combination of human object interactions, [WZZ13b] was able to detect concurrent actions from long video sequences. This work also explored temporal relationship between actions to capture some causal information that are related to objects.

The problem of lacking annotated video dataset is also alleviated in recent years. In 2015 [HZL15] presented an RGBD video dataset that contains 12 annotated human object interactions performed by 40 actors. This dataset was collected in lab environment and therefore does not have large variance in appearance and lighting. In 2016 a large-scale dataset Charades was released [SVW16]. Charades is a large-scale annotated human-object interaction video dataset. It was crowd sourced from the internet and therefore has huge variability in both appearance and lighting. It is very challenging because in addition to the variability, the camera is always moving and in many videos human are heavily occluded. In this work, we will use the dataset presented in [HZL15] to evaluate my model.

Human object interaction is also useful in the graphics community. [JL09] described a 3D computer graphics video editing system that leverages human object interaction information to help coordinating human motion and object motion.

2.3 Grammar Based Models

In [ZM07], Zhu et al. proposed a stochastic grammar of images describing the underlying structure in images. This monumental work had inspired a lot of research in modeling images and videos with structured representation using And-Or Graphs. [HZ09] models scene understanding with attributed grammar that can arrange scene configuration. Such a representation is not only discriminate but also generative, as in the parse graph derived from the grammar and input image can be used to reconstruct the image. This work is followed by [JZQ17] to create configurable photorealistic 3D indoor scenes. [GFM11] proposed a grammar-based object detection model that uses a carefully designed grammar and is robust against occlusion, appearance and geometry variance. [LWP09] proposed a grammar

that supports compositional object modeling. The compositive property of the stochastic grammar allows researchers to model subjects that are inherently hierarchical.

CHAPTER 3

Representation

3.1 Human-Object Interaction

While many previous works have explored decomposing actions into different subactions [LNS16, WYH16, LZR15], we argue that there is only one type of subaction, namely interaction, that is needed. Interaction can be classified as three types, physical human-object interaction, physical human-human interaction, and non-physical interaction. An example of the triplet is a person holding a cup, a person shaking hands with another person, and a person pointing at a cup. Although identifying all types of interactions can help us better understand human activity in videos, we only focus on physical interaction for the scope of this work.

The benefit of having physical interaction as the only subaction instead of having many different subactions is two-fold. Firstly, physical interactions, being one of the most fundamental elements of a human action, is easily generalizable across multiple actions. Other subactions can be very different across different actions. For example, picking up a cup while drinking and picking up a large box from ground and drastically different geometrically. Secondly, physical interaction can be easily modeled as whether the movement of an object part is coherent to the movement of a body part.

3.2 Attributed Spatial-Temporal And-Or Graph

In addition to human-object interaction, we also introduce strong structure into the action recognition model by using an Attributed Spatial-Temporal And-Or Graph (AST-AOG)

to model the joint distribution between 3D human pose, action label, and human-object interaction conditioned on video as an image sequence.

AST-AOG is an extention to the And-Or Graph model proposed in [ZM07] where it has two sets of inter-connected subgraph S-AOG and T-AOG, capturing the spatial and temporal structure of input video correspondingly. The letter A in AST stands for attributed. And-Or Graph model is a graph model where each node is either an AND node, an OR node, or a terminal node. An AND node decomposition of a node. This is the same as having the following production rule in a grammar:

$$A ::= BCD \tag{3.1}$$

where B, C and D are some other nodes. An OR node states that the current node can choose one of its children as its value when being instantiated. This corresponds to the following rule in a grammar:

$$A ::= B|C|D \text{ with } \theta_B, \theta_C, \theta_D \tag{3.2}$$

where B, C and D are some other nodes and θ_B, θ_C and θ_D are the probabilities associated with the three choices. A terminal node is similar to a terminal t in any grammar, where in the context of computer vision it is usually grounded on images. When an AOG is instantiated, it outputs a parse graph that captures the structure and content of input data. In the case of this work, the input data is a video clip. This is similar to when a linguistic grammar is instantiated on a sentence, it outputs a parse tree that captures the structure and meaning of the sentence.

The AST-AOG is composed of two parts, Spatial And-Or Graph (S-AOG, G^S) and Temporal And-Or Graph (T-AOG, G^T), each with their own hierarchical structures. Given a video, a parse graph pg is instantiated from the AST-AOG to represent the understanding of the video. Figure 3.1 and 3.2 shows part of an example parse graph instantiated from the AST-AOG. Figure 3.1 shows the parse graph from spatial perspective and Figure 3.2 shows the parse graph from temporal perspective. We will explain S-AOG and T-AOG separately in the next two sections. We will then present the probabilistic formulation of the AST-AOG in the next chapter.

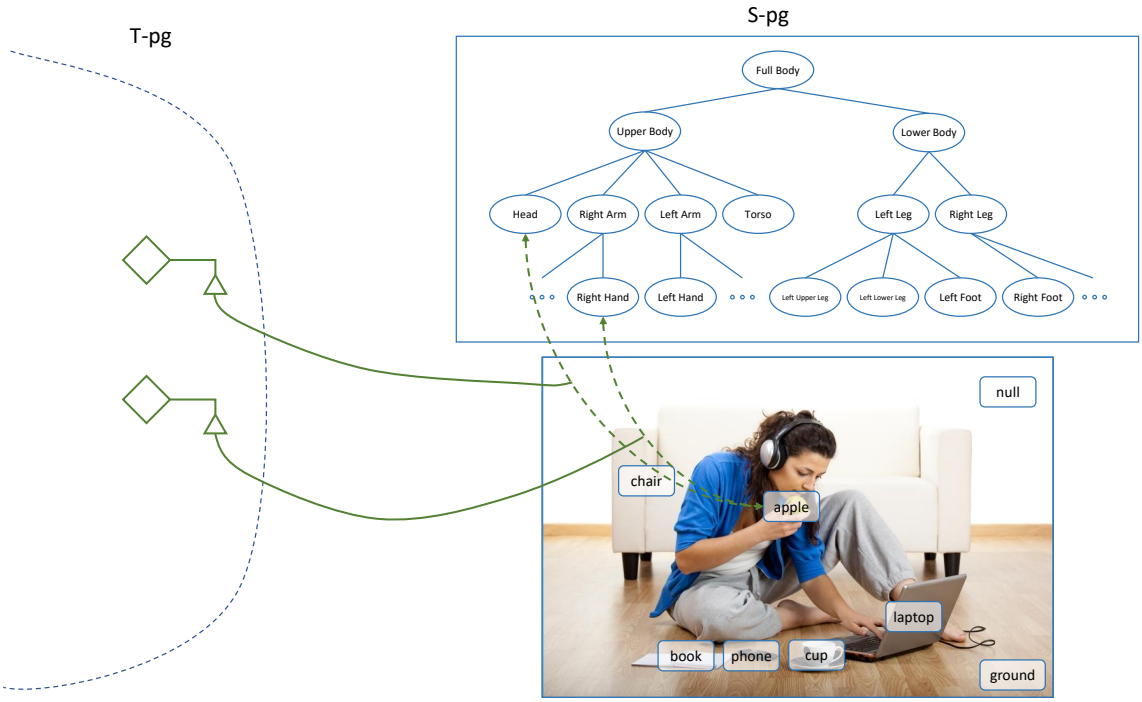

Figure 3.1: Partial Example Parse Graph from AST-AOG from the perspective of S-pg at a single frame. Each green diamond is a node in T-pg. Green triangle represents attributes for nodes in T-pg. Each circle and rounded rectangle is a node in S-pg. Green dashed arrows represents that the geometric relationship between spatial nodes. Green solid curves show that the relationships are imposed by attributes of temporal nodes.

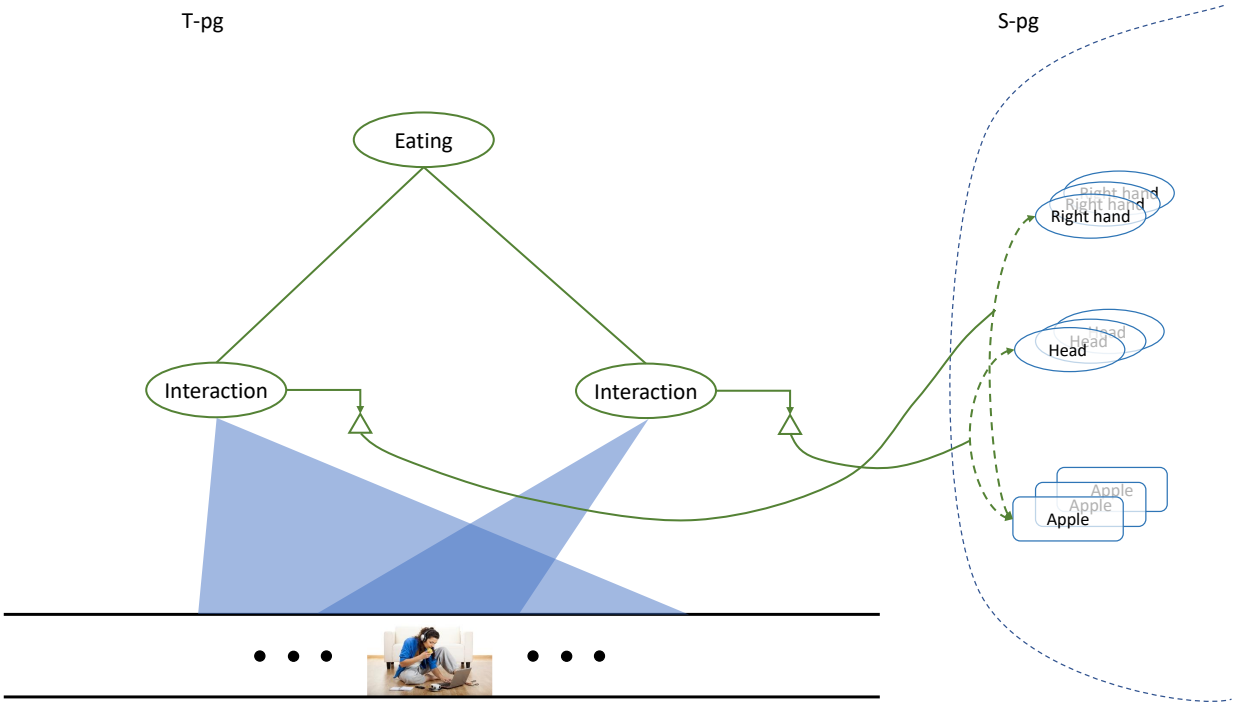

Figure 3.2: Partial Example Parse Graph from AST-AOG from the perspective of T-pg. Each green circle is a node in T-pg. Each green triangle represents the attribute on a T-pg node. Each circle and rounded angle rectangle is a node in S-pg at each frame. The green solid line represents hierarchy in T-pg. The triangular shadows under each interaction node represents where they are grounded in the video. Green dashed arrows represent geometric relationships between spatial nodes. Green solid curves show that the relationships are imposed by the attributes of temporal nodes.

3.2.1 Temporal-AOG

Temporal AOG represents the relationship between actions and interactions. A T-AOG is denoted by $G^T = \langle S^T, V_N^T, V_T^T, R^T, P^T \rangle$ where S^T is the root node of the T-AOG, V_N^T is the set of non-terminal nodes, V_T^T is the set of terminal nodes. R^T represents the production rules and P^T represents the probabilities associated with the production rules. In our model, we define the structure of T-AOG explicitly. The root node is an OR node with an action label, which branches to different versions of a certain action. Each version is an AND node composed by a set of Human Object Interactions (HOIs). Each HOI node again an

OR node whose children are different versions of that HOI. The children are terminal nodes that terminate on images patches. Each HOI terminal node also have a pair of attributes associated with it, denoted by a small triangle. HOI attribute specifies the spatial and temporal constraint on S-AOG nodes. Figure 3.3 illustrates our design of T-AOG.

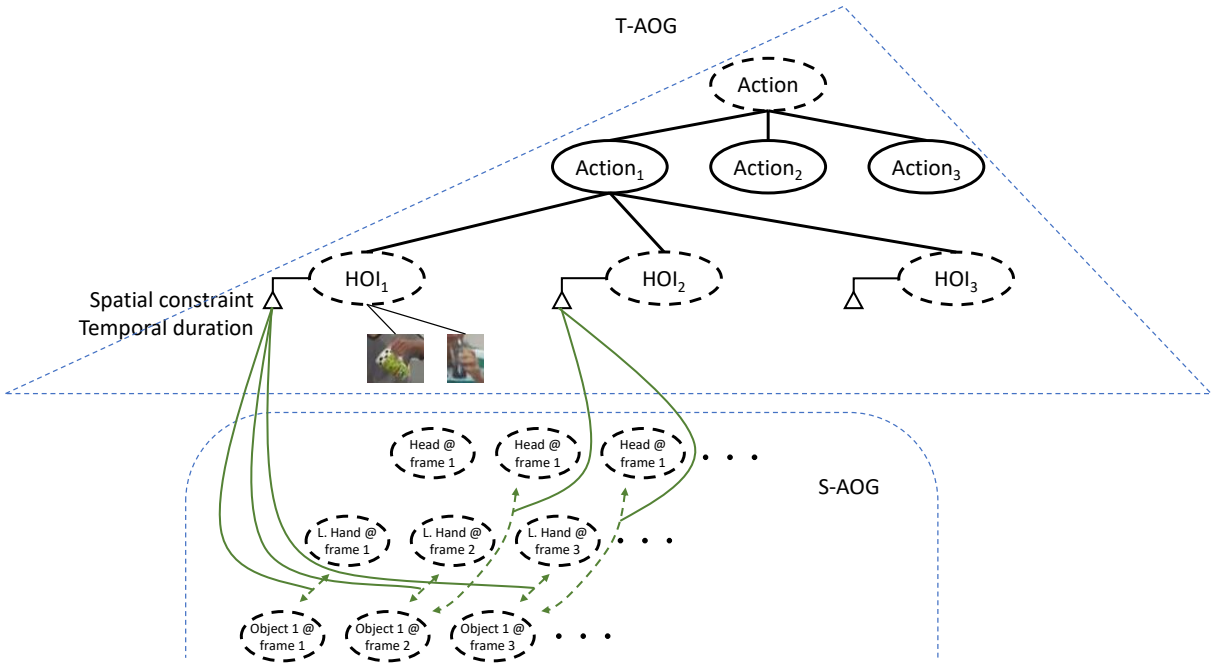

Figure 3.3: AST-AOG from the perspective of T-AOG. Dashed circles denote OR nodes, solid circles denote AND nodes. Image patches show grounding on images. Solid triangles show the attributes associated with each HOI node. Solid and dashed green curves show geometric relationships between spatial nodes imposed by HOI node attributes.

3.2.2 Spatial-AOG

Spatial AOG represents the spatial relationship between human body parts and objects. Figure 3.4 shows the structure of S-AOG.

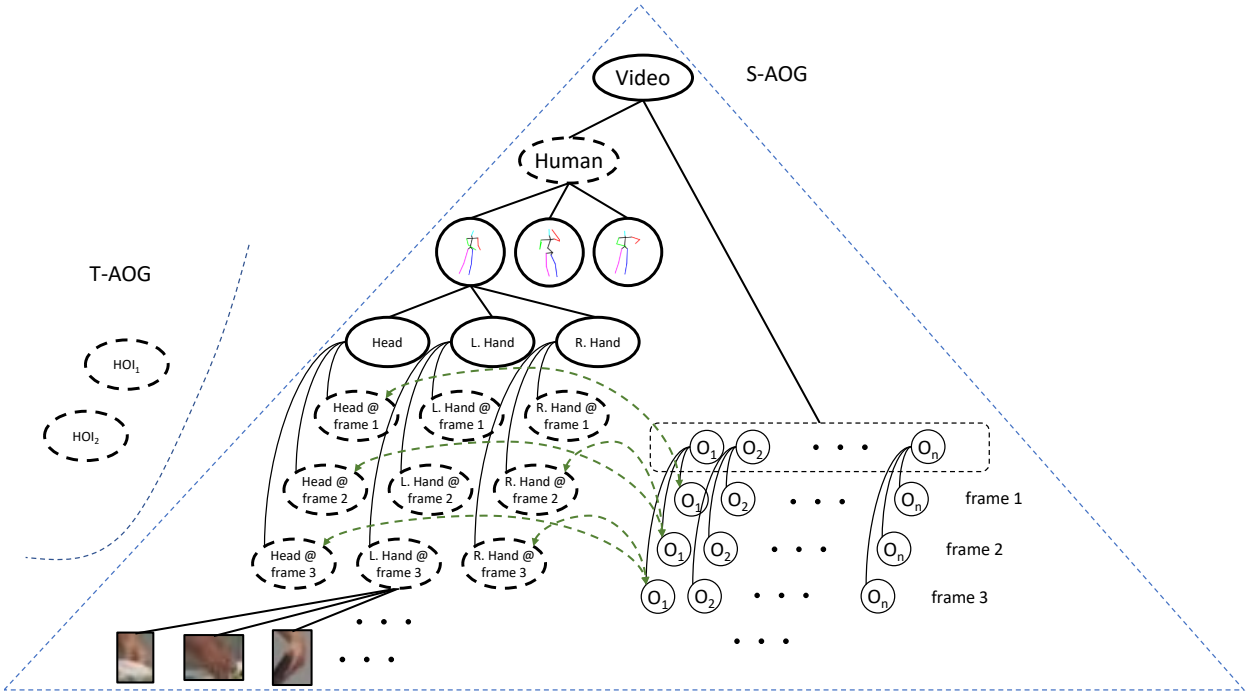

Figure 3.4: AST-AOG from the perspective of S-AOG. Solid circle are AND nodes, dashed circles are OR nodes. Dashed round rectangle is a special SET node for collection of objects. Green dashed double arrows show geometric relationship between spatial nodes. Image patches are possible groundings on image.

An S-AOG is denoted by a 5-tuple: $G^S = \langle S^S, V_N^S, V_T^S, R^S, P^S, \mathcal{C} \rangle$ where S^S is the root node of the grammar, V_N^S is the set of non-terminal nodes, V_T^S is the set of terminal nodes, R is the production rules and P is the probability associated with the rules. $\mathcal{C}$ is the set of relations between nodes. $\mathcal{C} = \{e_{ij} = (v_i, v_j, \theta_{ij})\}$ where v_i, v_j are the two nodes that are related. θ_{ij} is the parameter associated with the relationship representing the geometric relationships between v_i and v_j. The terminal nodes representing body parts in S-AOG are grounded on frames by off-the-shelf pose estimators. Terminal nodes representing objects are not grounded on the image but inferred from it relationship with other spatial nodes. Non-terminal nodes are either an AND node, an OR node or a SET node.

We define our S-AOG explicitly in the way that the root node S represents our video, which is decomposed into a human node and an object node. The object node is a SET node

for accepting multiple objects in a scene. A human node is an OR node that switches between possible human pose sequences, where each option is an AND node that represents each joint. Each joint node is then decomposed into its copies in all frames, which terminates to image patches. Notice that there could be a relationship between a pair of spatial nodes, denoted by dashed green double arrows in Figure 3.4. This relationship is imposed by attributes of temporal nodes. Although we only show part-object relationship, it can be extended to object-object relationship and part-part relationship future works.

3.3 Formulation

In this section we will describe the probabilistic formulation of the AST-AOG. The purpose for the AST-AOG is that given a video $V = \{I_t | t = 0, 1, ...\}$, we can instantiate a parse graph $pg^* = (pg^{S*}, pg^{T*})$ of the AST-AOG that maximizes the posterior probability. The posterior follows the classic Bayesian equation.

$$P(pg|V;\Theta) \propto P(V|pg;\Theta_l)P(pg;\Theta_p) \tag{3.3}$$

$$= \frac{1}{Z}\exp\{-\varepsilon(V|pg;\Theta_l) - \varepsilon(pg;\Theta_p)\} \tag{3.4}$$

3.3.1 Prior

The prior of parse graph pg is decomposed into spatial prior and temporal parse graph.

$$P(pg;\Theta_p) = \frac{1}{Z_p}\exp\{-\varepsilon_p^S(pg^S;\Theta_p^S) - \varepsilon_p^T(pg^T;\Theta_p^T)\} \tag{3.5}$$

3.3.1.1 Spatial Prior

The spatial prior energy $\varepsilon_p^S(pg^S;\Theta_p^S)$ is further decomposed into OR energy, SET energy and Relationship energies. AND nodes produces deterministic decomposition and therefore are not optimized over.

$$\varepsilon(pg^S;\Theta_p^S) = \sum_{v\in V^{OR}(pg^S)} \lambda_{OR}^S(\omega(v)) + \lambda_{SET}(\varphi(v^{SET})) + \sum_{(i,j)\in E(pg^S)} \lambda_{Rel}(v_i, v_j, \theta_{ij}) \tag{3.6}$$

1. **OR energy** λ_{OR}^S defines the energy on branching for OR nodes by assigning different

weights $\lambda^S_{OR}()$ to different branches $\omega(v)$. The weight reflects how frequently or likely a switch is taken. In the context of our model, the OR energy accounts for the a priori quality of the chosen pose sequence. In the context of this work, we approximate this by a sum of two terms:

(a) $\sum_{t\in[0,T]} -\log P(\text{pose}_t;\theta_{pose})$ that constrains each pose to be a valid human pose. The probability is estimated by a discriminative neural network trained on large human pose datasets such as H36M. Here T is the total number of frames of the input video and pose_t is the human pose represented by the spatial parse graph at frame t.

(b) $\sum_{t\in[1,T]} -\log N(\text{pose}_t - \text{pose}_{t-1};\theta_{posed})$ that constrains the difference between poses in two consecutive frames to be small. Since we assume the difference follows normal distribution, θ_{posed} represents μ_{posed} and Σ_{posed} and is estimated from data.

2. **SET energy** λ_{SET} defines the energy on SET node selection. Similarly to OR node, λ_{SET} reflects the preference of a certain set of selection $\varphi(v^{SET})$. Since object nodes do not carry any information more than there is an object somewhere, the only thing that matters in $\varphi(v^{SET})$ is the number of object nodes in parse graph. Therefore, the energy term can be modeled as

$$\lambda_{SET}(\left\|v^{SET}\right\| = i) = -\log \frac{\#\left\|v^{SET}\right\| = i}{\sum_j \#\left\|v^{SET}\right\| = j} \tag{3.7}$$

3. **Relationship energy** λ_{Rel} is captures whether the spatial arrangement of terminal nodes of pg^S is compatible with the geometric relationship imposed by pg^T, which is represented in θ_{ij}. In this work, we use the negative log of a multivariate normal function to approximate the energy of a relationship. The parameter θ_{ij} is therefore μ_{ij} and Σ_{ij}, which renders $\lambda_{Rel}(v_i, v_j, \theta_{ij}) = -\log N(v_i - v_j; \mu_{ij}, \Sigma_{ij})$. Notice that θ_{ij} is not learned, but instead is an attribute of node from temporal parse graph.

Details in how θ_{pose} is obtained is discussed in chapter 4.

3.3.1.2 Temporal Prior

Temporal prior is much simpler since there is no SET node and relationship in the T-AOG. The temporal prior energy $\varepsilon_p^T(pg^T;\Theta_p^T)$ is defined as the sum of OR energy of each OR node. Since there are two types of OR nodes, the OR energy is decomposed into two parts.

$$\varepsilon_p^T(pg^T;\Theta_p^T) = \sum_{v\in V^O R_A(pg^T)} \lambda_A^T(\omega(v)) + \sum_{v\in V^O R_{HOI}(pg^T)} \lambda_{HOI}^T(\omega(v)) \tag{3.8}$$

1. **HOI branching** energy λ_{HOI}^T is estimated by the frequency of the HOI template as a function of its attributes. In the context of this work, we discretize the geometric relationship θ and the HOI duration τ into bins and approximate the frequency according to the bin frequencies.

$$\lambda_{HOI}^T(\theta(v)\in b_i,\tau(v)\in b_j) = -\log\frac{\#\theta(v)\in b_i,\tau(v)\in b_j}{\sum_{m,n}\#\theta(v)\in b_m,\tau(v)\in b_n} \tag{3.9}$$

2. **Action branching** energy λ_A^T models the compatibility between the action node and its children interactions. In this term, we only care about the order of different types of HOIs, which is essentially which child node does the action node pick. Therefore, we can model our energy term as a negative log frequency of switch variable ω.

$$\lambda_A^T(\omega(v)=i) = -\log\frac{\#\omega(v)=i}{\sum_j\#\omega(v)=j} \tag{3.10}$$

3.3.2 Likelihood

The likelihood term $P(V|pg;\Theta_l)$ is also expressed in a Gibbs distribution.

$$P(V|pg;\Theta_l) = \frac{1}{Z_l}\exp\{-\varepsilon_l(V|pg;\Theta_l)\} \tag{3.11}$$

where the energy term ε_l can be re-written as energy terms associated with grounding energies of terminal nodes.

$$\varepsilon_l(V|pg;\Theta_l) = \sum_{t\in V_H^T(pg^S)}\lambda_t^S(V(t),t) + \sum_{t\in V_X^T(pg^T);\Theta_l^S}\lambda_l^T(V(t),t;\Theta_l^T) \tag{3.12}$$

where $V_H^T(pg^S)$ denotes all human body part terminal nodes in spatial parse graph, and $V_X^T(pg^T)$ denotes all interaction terminal nodes in temporal parse graph. $V(t)$ denotes the image patch around the grounding of terminal node t.

The spatial grounding energy term λ_t^S models the likelihood of seeing the image patch $V(t)$ where the spatial parse graph says there is some joint t.

Idealy we want to learn a likelihood energy with a descriptive model that can produce a $P(V'(t)|t)$ where $V'(t)$ is a reduced form of $V(t)$ such as Primal Sketch [GZW07] or HoG feature [McC86]. Due to the limit of time and resource in completion of this work, we approximate $\lambda_l^S(V(t), t)$ by the negative log of posterior probability $-\log P(t|V)$ obtained from heat maps from off-the-shelf pose estimator.

The temporal grounding energy term models the likelihood of seeing each frame conditioned on the temporal parse graph. Since action nodes are not grounded on image, the temporal likelihood is the likelihood of human-object interaction terminal nodes.

Similar to spatial likelihood, ideally we want to learn a descriptive model that can produce likelihood $P(V'(t)|t)$ where $V'(t)$ is a reduced form of $V(t)$. In the scope of this work, we approximate $\lambda_l^T(V(t), t)$ by $-\log P(t|V(t))$, which is obtained by training a binary discriminative model on image patches.

CHAPTER 4

Parse Graph Inference by MCMC

Markov Chain Monte Carlo (MCMC) is a class of algorithms that can sample from complex probability distributions. In this chapter, we will explain MCMC and Simulated Annealing and introduce my approach of sampling parse graph.

4.1 MCMC

Markov Chain Monte Carlo is a general technique that generate samples from complex probability distribution in high dimensions from a random number generator. A MCMC has a Markov chain with a proposal probability distribution $Q(x, y)$. Q defines the probability of proposing state change from state x to state y. An MCMC samples from the equilibrium probability π of its Markov chain. MCMC is widely used in many areas including physics, chemistry, economics and computer vision. In this work, we use Metropolis-Hastings algorithm, which is a member of the MCMC family.

Metropolis-Hastings algorithm was first introduced by Metropolis et al. in [MRR53] and was generalized by Hastings. in [Has70]. The basic idea of Metropolis-Hastings algorithm is that given a proposed state change, we can compute an acceptance rate α according to our target distribution π.

$$\alpha(x, y) = \min(1, \frac{Q(y, x)}{Q(x, y)} \cdot \frac{\pi(y)}{\pi(x)}) \tag{4.1}$$

where Q is the proposal probability. The presence of acceptance probability allows us to use a simple distribution Q to sample from a complex distribution π. In addition, if all state changes are reversible, i.e. $Q(y, x) = Q(x, y)$, the acceptance rate can be further reduced to

the following form.

$$\alpha(x, y) = \min(1, \frac{Q(y, x)}{Q(x, y)} \cdot \frac{\pi(y)}{\pi(x)}) \tag{4.2}$$

$$= \min(1, \frac{\pi(y)}{\pi(x)}) \tag{4.3}$$

If our target distribution follows a Gibb's distribution $\pi(x) = \frac{1}{Z}e^{-E(x)}$, α then becomes

$$\alpha(x, y) = \begin{cases} 1 & \text{if } \Delta E \leq 0 \\ e^{-\Delta E} < 1 & \text{otherwise} \end{cases} \tag{4.4}$$

where $\Delta E = E(x) - E(y)$.

Notice that x and y can be in different spaces as long as the jump is reversible, as proposed in [Gre95]. Markov Chain Monte Carlo has been used in many scenarios in the area of computer vision. [TZ02] and [ZZT00] presented image segmentation and object recognition using data-driven Markov chain Monte Carlo.

4.2 Simulated Annealing

In the previous section we introduced using MCMC to sample from target distribution π. To find the optimal parse graph from a distribution requires locating the state with global maximum probability.

$$pg^* = \arg\max_{pg} P(pg|V) \tag{4.5}$$

In this work, we adopt the simulated annealing algorithm [KGV83, CCP87] to gradually cool down the target distribution π so that the final sampled result is at global optimum. The idea of simulated annealing comes from annealing as a process in metal processing, where a heated metal is let cool gradually to remove internal inconsistency and strengthen it. Simulated annealing takes any probability P and modify it by adding a temperature term T that gradually decreases so that $P' = P^{\frac{1}{T}}$. At initial temperature $T = T_0$ when T_0 is high, $P^{\frac{1}{T}}$ is smooth and therefore the Markov chain can traverse to different states and overcome local barriers. When the temperature slowly cools down, structure in the distribution appears

and allows the Markov chain to focus on optimal regions. When temperature is small enough, we can imagine that the probability is close to 1 at global optimum and is 0 everywhere else. Figure 4.1 shows an illustration of $P^{\frac{1}{T}}$ at different T.

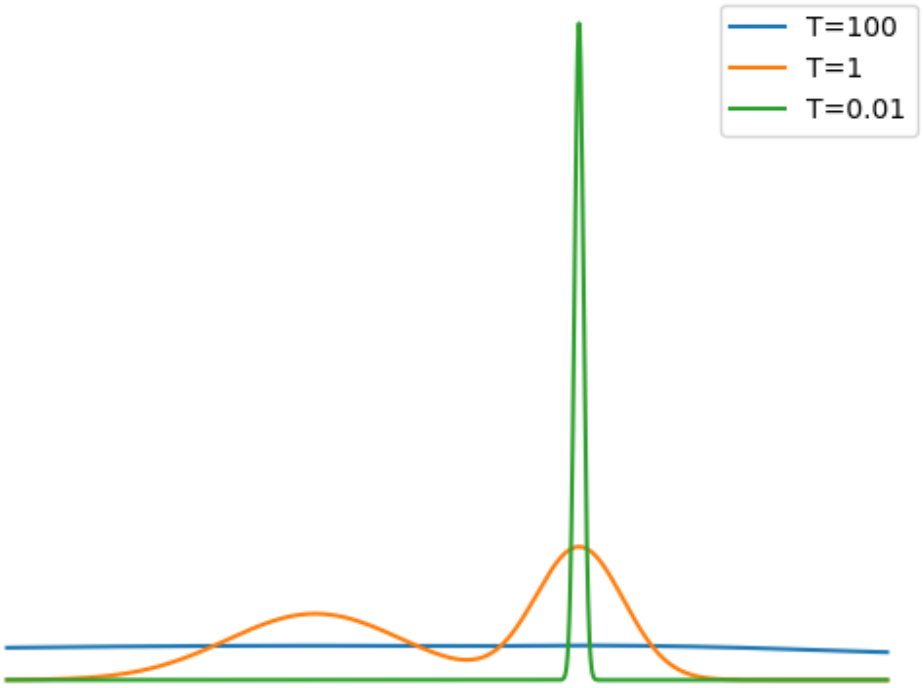

Figure 4.1: Illustration of effect of different temperature.

4.3 Learning

In my formulation, there are 2 sets of parameters that need to be learned. Namely, θ_{pose} for pose prior and Θ_l^T for temporal likelihood.

θ_{pose} is learned by training a valid pose discriminator from large public human pose dataset such as H36M [IPO14]. We use a simple MLP to discriminate between real poses from H36M, and fake poses generated by adding random perturbation at various level to the

real poses.

Θ_l^T is learned from training data. To prepare the data for learning Θ_l^T, we first used off-the-shelf 2D human pose estimator to get the around of each joint, then cropped the images to get the patches near each joint. We trained a small LSTM-CNN model for the binary classification problem of whether a given sequence of image patches have physical interaction in them. The simple model achieved 87% classification accuracy on the SYSU 3DHOI dataset [HZL15].

4.4 Inference by Sampling with Simulated Annealing

Based on the AST-AOG and learned parameters, we sampled video parse graph based on the posterior distribution $P(pg|V;\Theta)$ using a Markov Chain Monte Carlo (MCMC) sampler. To allow the Markov chain to explore more areas and possibly overcome local optimum in order to converge to global optima, we used simulated annealing with log annealing schedule suggested in [CCP87]. The samples are proposed in each step by randomly choosing a Markov chain dynamic described below.

- Dynamic q_1 chooses a human joint in one frame and re-sample the joint position from its posterior distribution $P(t|V)$, which is obtained from off-the-shelf 3D pose estimators.

- Dynamic q_2 creates an interaction node with randomly initialized attributes. Specifically, the created pointer would point to a random body part node and a random object node. Both pointer targets are chosen in uniform distribution. The pointer will sample a geometric relationship bin and a temporal duration bin from frequency

$$\text{freq}(\theta(v) \in b_i, \tau(v) \in b_j) = \frac{\#\theta(v) \in b_i, \tau(v) \in b_j}{\sum_{m,n} \#\theta(v) \in b_m, \tau(v) \in b_n} \tag{4.6}$$

  where $\theta(v)$ is the geometric relationship parameters and $\tau(v)$ is the temporal duration. A specific set of $\theta(v)$ and $\tau(v)$ is then sampled from uniform distribution in the bins. The center of temporal domain of the created node is sampled from the posterior distribution of HOI given by Θ_l^T.

- Dynamic q_3 chooses an HOI node by uniform distribution, and randomly grow the temporal duration on either side by t, where $t \sim N(0,1)$. If $t < 0$ then the temporal duration is shrunk instead of grown. The chance of picking either side to grow is equal. If the picked side cannot grow then this dynamic will do nothing if $t > 0$.

- Dynamic q_4 randomly chooses an HOI node in uniform, randomly choose a pointer from the picked node in uniform, and re-assign the pointer to another body part node or object node chosen uniformly.

- Dynamic q_5 randomly select a set of HOI nodes and create an action node as a parent of it. We first uniformly pick a number $n \sim Unif([1, N])$ where N is the total number of HOI nodes. We then randomly sample n HOI nodes in uniform distribution. We create an action node v that is a parent of the chosen HOI nodes. We assign action label i to v that minimizes the action branching energy.

$$\lambda_A^T(\omega(v) = i) = -\log \frac{\#\omega(v) = i}{\sum_j \#\omega(v) = j} \tag{4.7}$$

- Dynamic q_6 randomly chooses an HOI node in uniform distribution and destroy it. Remove all edges connected to the node.

- Dynamic q_7 randomly chooses an action node in uniform distribution and destroy it. Remove all edges connected to the node.

where q_1 and q_3 are diffusion dynamics, and the rest are reversible jumps. In each step before we choose a dynamic in random, we first create fresh object node so that HOI nodes can point to new object nodes. After each step when a dynamic performed, we remove all object nodes with no pointers to it and recalculate the position of all object nodes to minimize the Relationship energy in spatial prior.

According to the Metropolis-Hastings algorithm, proposed sample parse graph pg is accepted by the following acceptance probability,

$$\alpha(pg'|pg;\Theta) = min(1, \frac{P(pg'|\Theta)P(pg|pg')}{P(pg|\Theta)P(pg'|pg)}) \tag{4.8}$$

Since all dynamics are reversible with same probability, $P(pg|pg')$ and $P(pg'|pg)$ are the same. Therefore,

$$\alpha(pg'|pg;\Theta) = min(1, \frac{P(pg'|\Theta)}{P(pg|\Theta)}) \tag{4.9}$$

$$= min(1, \exp\{\varepsilon(pg|\Theta) - \varepsilon(pg'|\Theta)\} \tag{4.10}$$

The Markov chain is initialized so that human poses are at the maximum a posteriori state given by human pose estimator.

The complete algorithm is described below.

Require: Video V, iteration number I_{MAX}

1: Initialize $pg = (pg^S, pg^T)$ where human part nodes are created and grounded using off-the-shelf pose estimator. All other nodes are not created.
2: Initialize $C = C_0$, $i = 0$
3: **while** $i < I_{MAX}$ **do**
4: $i = i + 1$
5: $T = \frac{C}{\log(i)}$
6: Randomly pick a Markov chain dynamic q and generate a new pg' from applying q to pg
7: Compute $\alpha(pg'|pg)$ according to Eq. 4.10
8: Sample a random number $a \sim N(0, 1)$
9: $pg = pg'$ if $a > \alpha$
10: **end while**
11: **return** pg

CHAPTER 5

Parse Graph Inference by Deep Learning

In addition to inferring optimal parse graph by sampling with simulated annealing, we also implemented a deep learning model that can be trained in an end-to-end fashion. Since the model incorporates temporal information, we used an long short-term memory (LSTM) [HS97, GSC99] layer to model the temporal relationship between nodes. We will first briefly introduce LSTM before explaining my approach.

5.1 Long Short-Term Memory

LSTM is a specific style of recurrent neural network (RNN) architecture that were designed to model temporal relationships. It can model long-range dependencies better than other styles of RNNs and has become a popular choice of RNN in recent years. Unlike other RNN modules, LSTM contains a memory block in its recurrent module for storing information across time. Each memory block contains an input gate and an output gate. Information is propagated into the memory cell from input gate and back to the model through output gate. There is also a forget gate [GSC99] that adaptively scales the internal state of an LSTM cell. The presence of forget gate allows LSTM cell to segment an input sequence and forget about previous segment when needed to. Figure 5.1 illustrates the architecture of an LSTM module.

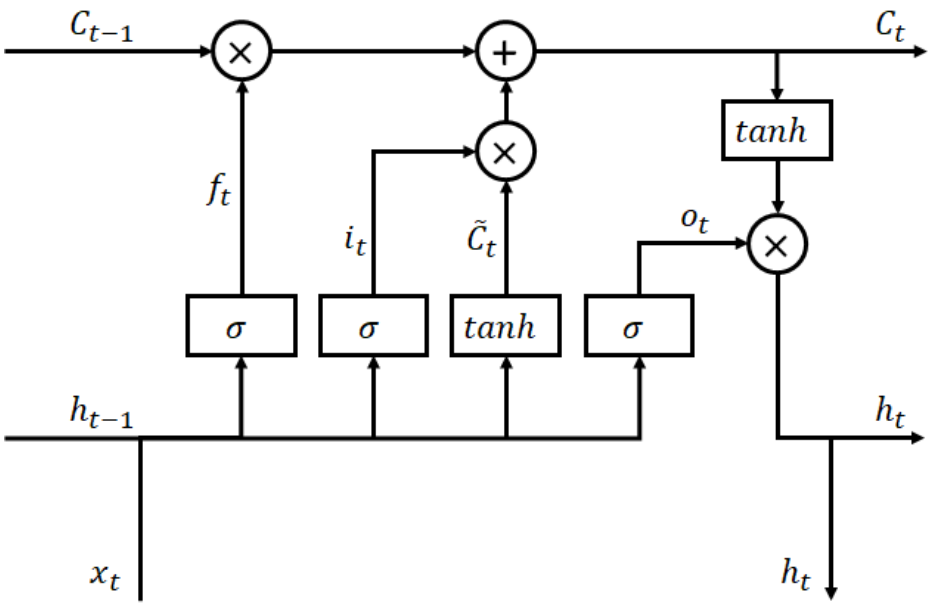

Figure 5.1: Illustration of an LSTM module.

The LSTM module learns a mapping f encoded by network weights W between input sequence $\mathbf{x} = \{x_1, x_2, \ldots, x_T\}$ and output sequence $\mathbf{h} = \{h_1, h_2, \ldots, h_T\}$. Given an input sequence $\mathbf{x}$ and cell states, the output $\mathbf{h}$ can be computed by

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{5.1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5.2}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{5.3}$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{5.4}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5.5}$$

$$h_t = o_t \times \tanh(C_t) \tag{5.6}$$

where each term denotes

Symbol	Meaning
W	weight matrices
b	bias vectors
x	input sequence
h	output sequence
i	input gate
f	forget gate
o	output gate
C	cell activation vector
σ	logistic sigmoid function

5.2 My Approach

In order to make training feasible with neural network, the parse graph pg is encoded into a fixed length vector. For each frame, the vector has length $57 + N$. The first 51 elements encodes the 3D coordinates of 17 human body parts, and then there are 3 bits representing whether each joint of interest has physical interaction, followed by another 3 bits representing whether each pair of joints are interacting with same object. This is then followed by N elements containing a one-hot encoding of action label. In the context of this experiment, $N = 12$. Therefore the resulting vector is of length 69.

The model is designed to do 3 tasks simultaneously: action recognition, interaction detection, and human pose refinement. This is done by a hard parameter sharing mechanism as illustrated by Figure 4.1. [Bax97] showed that hard weight sharing of N tasks can reduce the risk of overfitting by the order of N.

As illustrated by Figure 4.1, the network uses a pretrained VGG-16 module as a feature extractor. Extracted features from each frame are concatenated with estimated 3D human pose at that frame before being fed into a recurrent neural network for parse graph prediction. Network loss $\mathcal{L}$ is established on the entire pg prediction and is a sum of a triplet

$(\mathcal{L}_P, \mathcal{L}_A, \mathcal{L}_X)$.

$$\mathcal{L} = \mathcal{L}_S + \mathcal{L}_A + \mathcal{L}_X \tag{5.7}$$

$$\mathcal{L}_P = \sum_{t\in[0,T]} \sum_{i\in[1,51]} (p_{t,i} - \hat{p}_{t,i})^2 \tag{5.8}$$

$$\mathcal{L}_A = -\sum_{t\in[0,T]} \sum_{i\in[1,12]} a_{t,i} \log \hat{a}_{t,i} \tag{5.9}$$

$$\mathcal{L}_X = -\sum_{t\in[0,T]} \sum_{i\in[1,6]} x_{t,i} \log \hat{x}_{t,i} \tag{5.10}$$

where $p_{t,i}, a_{t,i}, x_{t,i}$ represents the i-th bit in pose, action and interaction representation at frame t. Notice that here we only encode S-pg and T-pg correctness in our loss function. This is because the connection Φ is encoded in and regulated by the LSTM module.

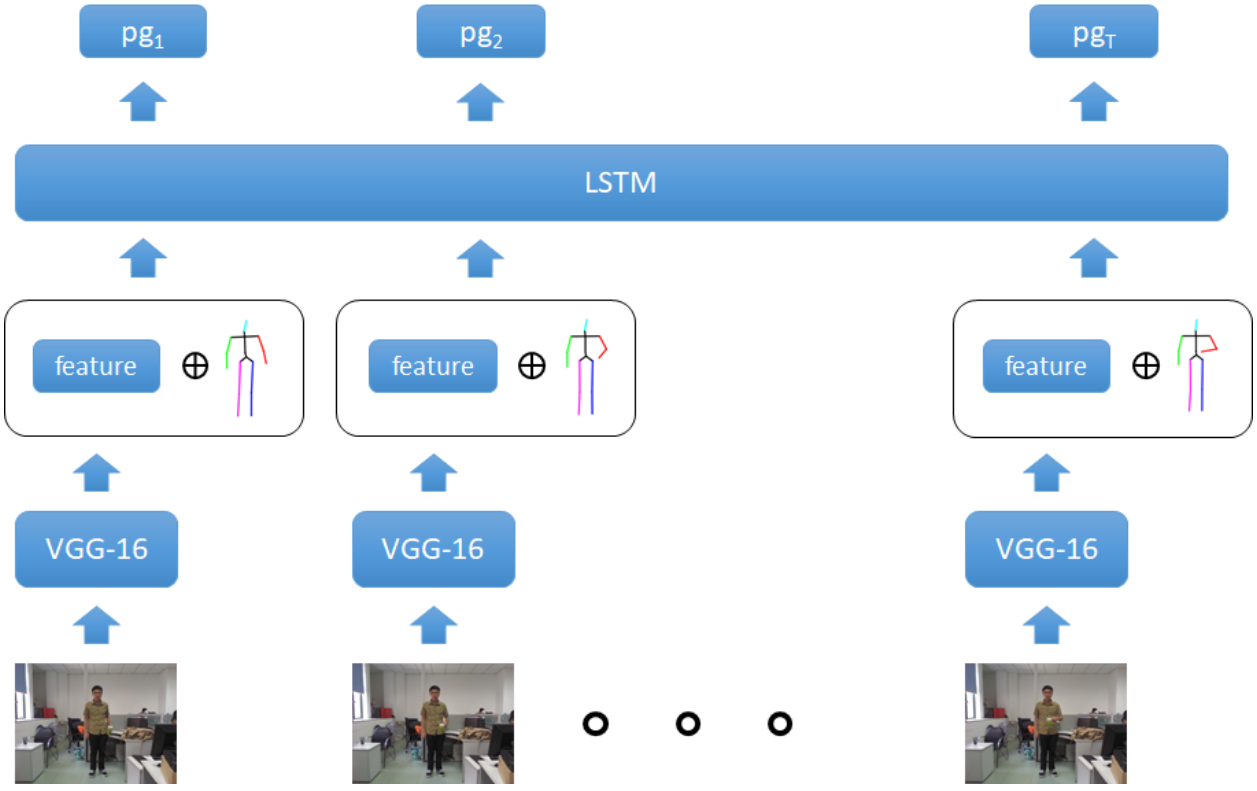

Figure 5.2: Deep Learning Model

5.3 Future Works

In Chapter 3 we presented stochastic grammar and a set of probabilistic formulation in the form of Gibbs distribution

$$P(pg|V) = \frac{1}{Z} \exp\{-\varepsilon(pg|V)\} \tag{5.11}$$

where we proposed a set of energy terms our parse graph should minimize over. In the model presented in this chapter, we only tested the usefulness of having extra supervision in interaction, and did not organize our model so that it is coherent to our representation and formulation. In future works we will extend our model so that it is coherent to our stochastic grammar representation and can minimize the energy terms that we proposed.

CHAPTER 6

Evaluation

In this chapter we will first introduce the dataset used for evaluation, and then discuss the performances of both methods and discuss their implications. Notice that due to the lack of ground truth in 3D object localization, we do not evaluate this part of my model.

6.1 Dataset

The dataset we are using is the SYSU 3DHOI dataset [HZL15]. 3DHOI has 12 action labels, each performed by 40 different subjects, resulting in 480 video clips in total. The actions are carefully designed so that some actions use the same type of object and some actions are similar geometrically. Therefore, models cannot do well on this dataset by simply recognizing the object in the frames or by simply matching human pose with remembered dictionaries. Some example actions are illustrated in Figure 6.1.

Figure 6.1: SYSU 3DHOI Dataset Examples

6.2 MCMC with Simulated Annealing

Due to the random walk nature of the MCMC algorithm, it is very slow for simulated annealing to generate result for a single example. Therefore we do not have a quantitative metric for evaluation. Qualitatively, we have shown that with the AST-AOG formalization we can sample a meaningful representation from an input video sequence. An example representation is shown in Figure 6.2. A failure example is shown in Figure 6.3. When the Markov chains converge, using simulated annealing with MCMC can reduce mean square error in human pose estimation by 2.25% than off-the-shelf 3D human pose estimation. However the error of human pose estimation can be very large if the Markov chain fail to converge. The accuracy for action recognition is not available because there is too few results for this statistic to be meaningful.

One common failure scenario for the MCMC method is that when there is strong enough evidence for a false action label present, the action label will force other parts of the parse

graph to agree with it. Human pose estimation will then be dragged towards the false action label in order to minimize the compatibility energy $\varepsilon^{\Phi}_{comp}$. $\varepsilon^{\Phi}_{comp}$ (pose-action compatibility) and ε^{S}_{l} (pose-image compatibility) will compete against each other resulting in unnatural and unreal human pose estimations. Theoretically, having a stronger Θ^{Φ}_{comp} and Θ^{S}_{v} and put heavier weights on them can solve this problem. Another failure scenario comes from the fail in interaction detection. Tuning down the weight for ε^{T}_{l} alleviated the problem but did not solve it. A more carefully designed interaction detector may be needed in the future.

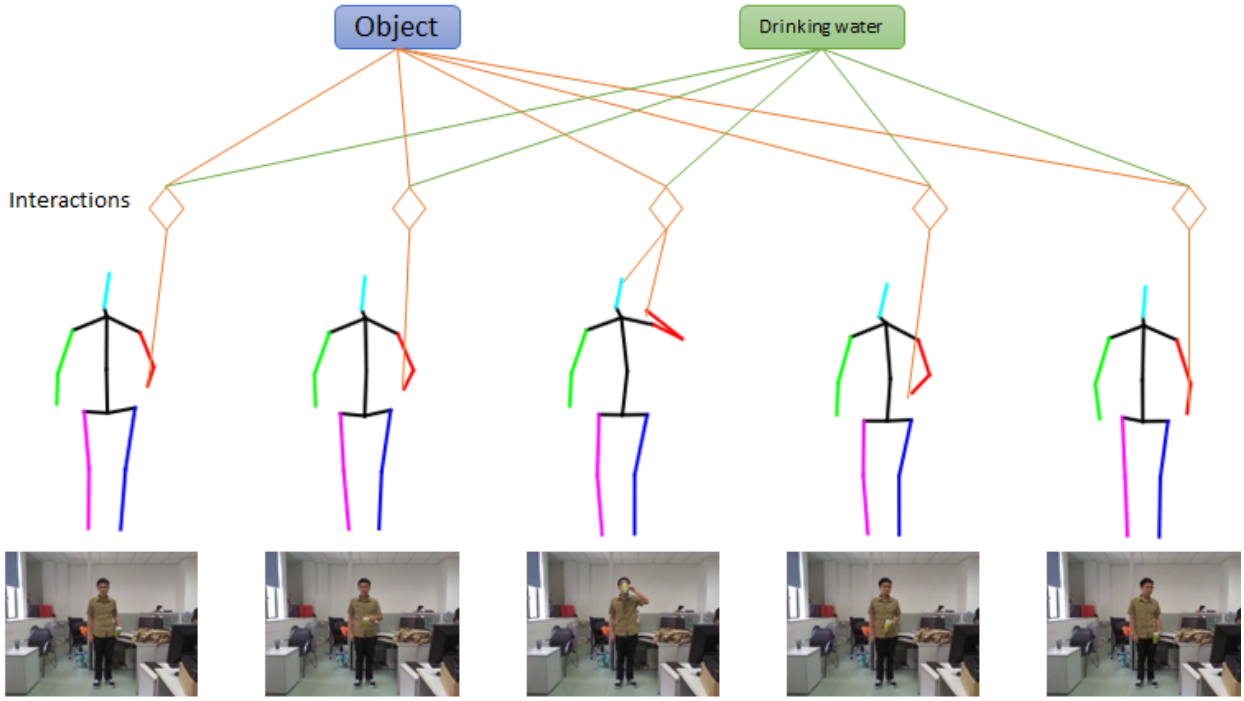

Figure 6.2: Example Video Representation - Drinking Water

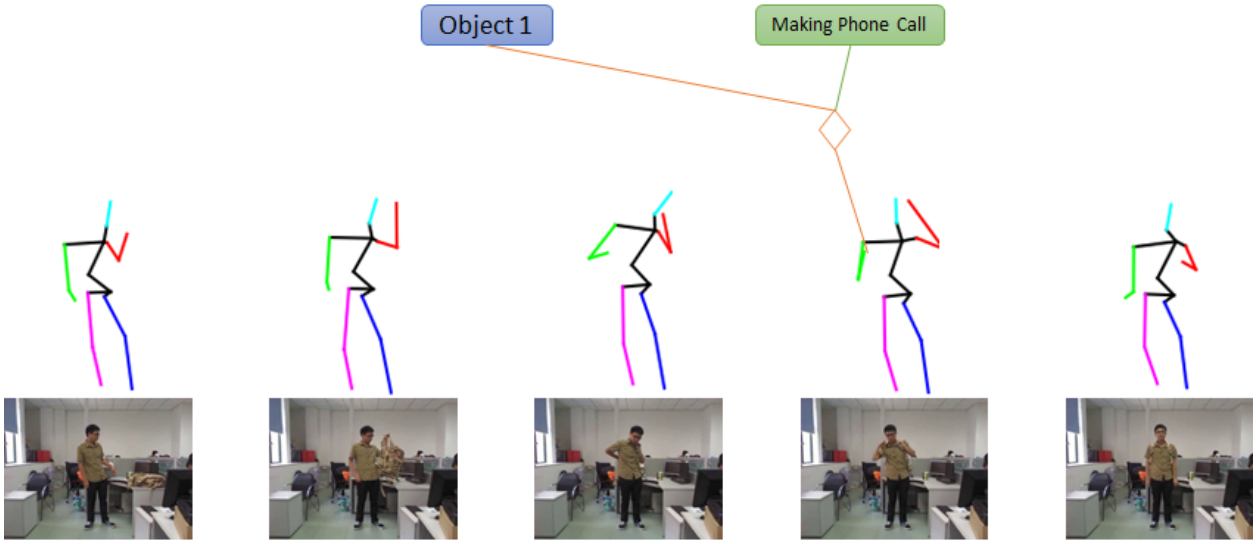

Figure 6.3: Example Video Representation (Failure) - Wear Backpack

6.3 Deep Learning Approach

As illustrated in Table 6.1, the model has outperformed all baseline models that uses only RGB images. To show that having interaction information does improve action recognition performance, we also ran ablation study for two different network structures. As shown in Table 6.2 and Figure 6.4, using interaction information helps improve model performance significantly.

The pose estimation refinement module in the deep learning model is not working well. The output pose is much worse than the input pose estimated by off-the-shelf 3D pose estimator. This is surprising because we assumed that the worst this model can do is to learn an identity mapping that outputs the input directly. Apparently having to predict action and interaction label messes up the network parameters. We believe the issue is that in the deep learning approach it is hard to explicitly learn the mapping between action and its corresponding dictionary of pose sequences. This problem may be solved with a more careful design, such as introducing skip connections or adding an adversarial module to pose estimation.

Table 6.1: Action Recognition Accuracy

Model	Action Recognition Accuracy
LSTM-VGG16	54.40%
TwoStream-VGG16	45.83%
TwoStream Fusion	79.17%
Ours	**85.00%**

Table 6.2: Ablation Study

Model	LSTM-VGG16	Ours
Without interaction	54.40%	79.19%
With interaction	**64.08%**	**85.00%**

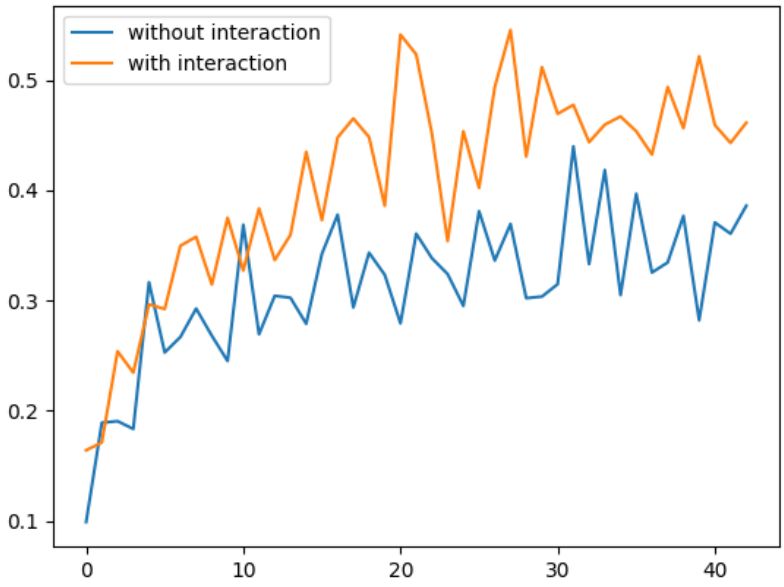

Figure 6.4: Classification Accuracy of LSTM-VGG16

CHAPTER 7

Conclusion

In this thesis we have proposed a two-layer attributed spatial temporal grammar of action recognition that leverages human object interaction. We used simulated annealing to demonstrate our grammar by generating meaningful parse graph from example videos and designed a deep learning model to verify that having a set of interaction node in the representation does improve action recognition performance.

There are a few things that can be improved or explored in the future. Number one is the joint training of And-Or Graph model with deep learning. Some preliminary works have been done inside VCLA and I will be very excited to work on combining the power of stochastic grammar with end-to-end training. Another improvement point is that in the current version we do not use object detector of any kind. In future we could consider incorporating object detection. Although it may not work for small objects that are often completely occluded by human, it may reduce the search size of possible parse graph by a large factor for larger objects. With the recent breakthrough in object detection in RGB images [HGD17], I believe grounding some object node on image can improve the performance of our model by a large margin. Another very exciting potential for improving our model lies in our modeling of likelihood energy. In Section 3.3.2 we used the negative log of posterior probability $P(t|V(t))$ to approximate likelihood energy. Should we have access to a descriptive model to produce likelihood $P(V(t)|t)$, or $P(V'(t)|t)$ where $V'(t)$ is a reduced version of $V(t)$, we would have a more accurate modeling of the conditional probability distribution of parse graphs.

REFERENCES

[Bax97] Jonathan Baxter. "A Bayesian/Information Theoretic Model of Learning to Learn via Multiple Task Sampling." *Machine Learning*, **28**(1):7–39, Jul 1997.

[CCP87] Paolo Carnevali, Lattanzio Coletti, and Stefano Patarnello. "Image processing by simulated annealing." In *Readings in Computer Vision*, pp. 551–561. Elsevier, 1987.

[DRF10] Chaitanya Desai, Deva Ramanan, and Charless Fowlkes. "Discriminative models for static human-object interactions." In *Computer vision and pattern recognition workshops (CVPRW), 2010 IEEE computer society conference on*, pp. 9–16. IEEE, 2010.

[DSL11] Vincent Delaitre, Josef Sivic, and Ivan Laptev. "Learning person-object interactions for action recognition in still images." In *Advances in neural information processing systems*, pp. 1503–1511, 2011.

[FPZ16] Christoph Feichtenhofer, Axel Pinz, and AP Zisserman. "Convolutional two-stream network fusion for video action recognition." 2016.

[FR08] Roman Filipovych and Eraldo Ribeiro. "Recognizing primitive interactions by exploring actor-object states." In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–7. IEEE, 2008.

[GFM11] Ross B Girshick, Pedro F Felzenszwalb, and David A Mcallester. "Object detection with grammar models." In *Advances in Neural Information Processing Systems*, pp. 442–450, 2011.

[GKD09] Abhinav Gupta, Aniruddha Kembhavi, and Larry S Davis. "Observing human-object interactions: Using spatial and functional compatibility for recognition." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **31**(10):1775–1789, 2009.

[GM15] Georgia Gkioxari and Jitendra Malik. "Finding action tubes." In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pp. 759–768. IEEE, 2015.

[Gre95] Peter J Green. "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination." *Biometrika*, **82**(4):711–732, 1995.

[GSC99] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. "Learning to forget: Continual prediction with LSTM." 1999.

[GZW07] Cheng-en Guo, Song-Chun Zhu, and Ying Nian Wu. "Primal sketch: Integrating structure and texture." *Computer Vision and Image Understanding*, **106**(1):5–19, 2007.

[Has70] W Keith Hastings. "Monte Carlo sampling methods using Markov chains and their applications." *Biometrika*, **57**(1):97–109, 1970.

[HBS09] Dong Han, Liefeng Bo, and Cristian Sminchisescu. "Selection and context for action recognition." In *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1933–1940. IEEE, 2009.

[HGD17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. "Mask r-cnn." In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 2980–2988. IEEE, 2017.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory." *Neural computation*, **9**(8):1735–1780, 1997.

[HZ09] Feng Han and Song-Chun Zhu. "Bottom-up/top-down image parsing with attribute grammar." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **31**(1):59–73, 2009.

[HZL15] Jian-Fang Hu, Wei-Shi Zheng, Jianhuang Lai, and Jianguo Zhang. "Jointly learning heterogeneous features for RGB-D activity recognition." In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pp. 5344–5352. IEEE, 2015.

[HZR15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification." In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

[HZR16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[IPO14] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. "Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **36**(7):1325–1339, jul 2014.

[IS10] Nazli Ikizler-Cinbis and Stan Sclaroff. "Object, scene and actions: Combining multiple features for human action recognition." In *European conference on computer vision*, pp. 494–507. Springer, 2010.

[JL09] Sumit Jain and C Karen Liu. "Interactive synthesis of human-object interaction." In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 47–53. ACM, 2009.

[JS14] Simon Jones and Ling Shao. "Unsupervised spectral dual assignment clustering of human actions in context." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 604–611, 2014.

[JZQ17] Chenfanfu Jiang, Yixin Zhu, Siyuan Qi, Siyuan Huang, Jenny Lin, Xiongwen Guo, Lap-Fai Yu, Demetri Terzopoulos, and Song-Chun Zhu. "Configurable, Photorealistic Image Rendering and Ground Truth Synthesis by Sampling Stochastic Grammars Representing Indoor Scenes." *arXiv preprint arXiv:1704.00112*, 2017.

[KGV83] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. "Optimization by simulated annealing." *science*, **220**(4598):671–680, 1983.

[KS03] Yasuo Kuniyoshi and Moriaki Shimozaki. "A self-organizing neural model for context-based action recognition." In *Neural Engineering, 2003. Conference Proceedings. First International IEEE EMBS Conference on*, pp. 442–445. IEEE, 2003.

[LK81] Bruce D Lucas, Takeo Kanade, et al. "An iterative image registration technique with an application to stereo vision." 1981.

[LNS16] Ivan Lillo, Juan Carlos Niebles, and Alvaro Soto. "A hierarchical pose-based approach to complex action understanding using dictionaries of actionlets and motion poselets." *arXiv preprint arXiv:1606.04992*, 2016.

[LWP09] Liang Lin, Tianfu Wu, Jake Porway, and Zijian Xu. "A stochastic graph grammar for compositional object representation and recognition." *Pattern Recognition*, **42**(7):1297–1307, 2009.

[LZR15] Tian Lan, Yuke Zhu, Amir Roshan Zamir, and Silvio Savarese. "Action recognition by hierarchical mid-level action elements." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4552–4560, 2015.

[McC86] Robert K McConnell. "Method of and apparatus for pattern recognition.", January 28 1986. US Patent 4,567,610.

[MRR53] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. "Equation of state calculations by fast computing machines." *The journal of chemical physics*, **21**(6):1087–1092, 1953.

[SLC04] Christian Schuldt, Ivan Laptev, and Barbara Caputo. "Recognizing human actions: a local SVM approach." In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pp. 32–36. IEEE, 2004.

[SVW16] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. "Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding." In *European Conference on Computer Vision*, 2016.

[SZ14a] Karen Simonyan and Andrew Zisserman. "Two-stream convolutional networks for action recognition in videos." In *Advances in neural information processing systems*, pp. 568–576, 2014.

[SZ14b] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556*, 2014.

[TBF15] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. "Learning spatiotemporal features with 3d convolutional networks." In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pp. 4489–4497. IEEE, 2015.

[TZ02] Zhuowen Tu and Song-Chun Zhu. "Image segmentation by data-driven Markov chain Monte Carlo." *IEEE Transactions on pattern analysis and machine intelligence*, **24**(5):657–673, 2002.

[VAC14] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. "Human action recognition by representing 3d skeletons as points in a lie group." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 588–595, 2014.

[WKS11] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. "Action recognition by dense trajectories." In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 3169–3176. IEEE, 2011.

[WW17] Hongsong Wang and Liang Wang. "Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks." In *e Conference on Computer Vision and Pa ern Recognition (CVPR)*, 2017.

[WWY17] Junwu Weng, Chaoqun Weng, and Junsong Yuan. "Spatio-Temporal Naive-Bayes Nearest-Neighbor (ST-NBNN) for Skeleton-Based Action Recognition." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4171–4180, 2017.

[WYH16] Pei Wang, Chunfeng Yuan, Weiming Hu, Bing Li, and Yanning Zhang. "Graph based skeleton motion representation and similarity measurement for action recognition." In *European Conference on Computer Vision*, pp. 370–385. Springer, 2016.

[WZZ13a] Ping Wei, Yibiao Zhao, Nanning Zheng, and Song-Chun Zhu. "Modeling 4d human-object interactions for event and object recognition." In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 3272–3279. IEEE, 2013.

[WZZ13b] Ping Wei, Nanning Zheng, Yibiao Zhao, and Song-Chun Zhu. "Concurrent action detection with structural prediction." In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 3136–3143. IEEE, 2013.

[WZZ17] Ping Wei, Yibiao Zhao, Nanning Zheng, and Song-Chun Zhu. "Modeling 4d human-object interactions for joint event segmentation, recognition, and object localization." *IEEE transactions on pattern analysis and machine intelligence*, **39**(6):1165–1179, 2017.

[YF10] Bangpeng Yao and Li Fei-Fei. "Grouplet: A structured image representation for recognizing human and object interactions." In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 9–16. IEEE, 2010.

[ZM07] Song-Chun Zhu, David Mumford, et al. "A stochastic grammar of images." *Foundations and Trends® in Computer Graphics and Vision*, **2**(4):259–362, 2007.

[ZZT00] Song-Chun Zhu, Rong Zhang, and Zhuowen Tu. "Integrating bottom-up/top-down for object recognition by data driven Markov chain Monte Carlo." In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pp. 738–745. IEEE, 2000.