Attribute And-Or Grammar for Joint Parsing of Human Attributes, Part and Pose
Seyoung Park, Bruce Xiaohan Nie and Song-Chun Zhu

Abstract—This paper presents an attribute and-or grammar (A-AOG) model for jointly inferring human body pose and human attributes in a parse graph with attributes augmented to nodes in the hierarchical representation. In contrast to other popular methods in the current literature that train separate classifiers for poses and individual attributes, our method explicitly represents the decomposition and articulation of body parts, and account for the correlations between poses and attributes. The A-AOG model is an amalgamation of three traditional grammar formulations: (i) Phrase structure grammar representing the hierarchical decomposition of the human body from whole to parts; (ii) Dependency grammar modeling the geometric articulation by a kinematic tree of the body pose; and (iii) Attribute grammar accounting for the compatibility relations between different parts in the hierarchy so that their appearances follow a consistent style. The parse graph outputs human detection, pose estimation, and attribute prediction simultaneously, which are intuitive and interpretable. We conduct experiments on two tasks on two datasets, and experimental results demonstrate the advantage of joint modeling in comparison with computing poses and attributes independently. Furthermore, our model obtains better performance over existing methods for both pose estimation and attribute prediction tasks.

Index Terms—Attribute grammar, And-Or grammar, Attribute prediction, Pose estimation, Part localization, Joint parsing.

1 INTRODUCTION

1.1 Objectives and Motivations

In this paper, we present a probabilistic, compositional and graphical model for explicitly representing human poses, parts, and attributes in an attribute And-Or grammar (A-AOG), which combines three conventional grammar formulations:

- A phrase structure grammar representing the hierarchical decomposition from whole to parts;
- A dependency grammar modeling the geometric articulation by a kinematic tree of the body pose; and
- An Attribute grammar accounting for the compatibility relations between different parts in the hierarchy so that their appearances follow a consistent style.

As Figure 1 illustrates, the A-AOG parses an input image and outputs an attribute parse graph with three components. (i) The phrase structure parse tree is illustrated by vertical blue edges, and defines a valid hierarchical composition of body parts following the phrase structure grammar. (ii) The kinematic tree is illustrated by the horizontal green edges, and describes the articulations of body parts following a dependency grammar. (iii) A number of attributes (triangles) are associated with each node in the parse tree as illustrated with the red edges. Each attribute takes values from a finite set, namely its semantic domain, for example, ‘gender’ ∈ {‘male’, ‘female’}, and ‘hair style’ ∈ {‘long’, ‘short’, ‘bald’} with posterior probabilities shown in the colored bars.

In this representation, some attributes are said to be global, such as gender and age, as they are associated with all nodes in the parse tree. A global attribute can be inferred from individual body parts, for example, one may tell a person’s age or gender from the head, face, or upper body alone with different probabilities, which are illustrated by the colored bars at those nodes. In contrast, the local attributes are confined to nodes in the low levels
of the hierarchy, for example, hair style is an attribute of the head, and can be inferred from the head region alone when the body is occluded. During the inference process, both the global and local attributes pass information top-down and bottom-up in the phrase structure parse tree, and impose constraints to ensure consistency between the parts in a probabilistic way through extra energy terms modeling the correlations between parts. The final output for attribute prediction aggregates information from all the parsed parts.

The following two aspects motivate the proposed A-AOG for jointly modeling and inferring human poses and attributes in an explicit representation.

Firstly, it is desirable to have an explicit and interpretable model for integrating all the attributes in the hierarchical representation. Despite extensive research on attributes for objects [1], humans [2], and scenes [3] in the past decade, most methods train one classifier for each attribute, such as CNN classifiers, using features extracted from the whole image. As human pose has large variations, it is unreasonable to predict the attributes without knowing where the body parts are, especially for those local attributes. The problem will be more prominent when testing data are significantly different from the training data. Such blackbox classifiers are separated from the well-known graphical models for human pose estimation [4], [5], [6] and thus lack explicit understanding of the interactions between attributes and parts. Therefore, an explicit and simple model linking attributes to poses and parts is long desired but missing in the literature.

Secondly, it is desirable to infer pose and attributes jointly in a common representation. Some recent works use a pre-trained pose detection module [4] as a pre-processing stage and then predict attributes sequentially. Such method inevitably propagates the errors in pose estimation to attribute prediction, while a joint inference approach will solve pose and attributes in an iterative closed-loop and thus utilize the mutual information between pose and attribute, such as the co-occurrence of attributes, and the correlation between attributes and the parts.

The proposed A-AOG is aimed at addressing the two aspects above. The A-AOG is a context-sensitive grammar embedded in an And-Or graph [7]. In the and-or graph, the and-nodes represent decomposition or dependency; and the or-nodes represent alternative choices of decomposition or types of parts. The attributes at different levels modulate the choices at the Or-nodes and thus impose the context in a probabilistic way. For example, a female will have higher probability to wear a skirt than a male, and the skirt is represented by some templates in some or-node branch. Near the leaf node, the attribute is directly related to the choices of an Or-node. For example, glasses ∈ {‘yes’, ‘no’} is a local attribute for head, and a head with or without glasses corresponds to different part templates (or detectors) under an or-node.

1.2 Dataset Overview and Scope of Experiments
We evaluate the performance of the A-AOG on multiple benchmarks in various experiment settings.

**Public dataset on attribute prediction.** The attribute of people dataset [2] is a widely contested benchmark. In this dataset, each image is centered and ground truth bounding box of each person is provided with 9 binary attributes. We tested the A-AOG model on attribute classification task in comparison with existing approaches [8], [9], [10], [11], [12], and on pose estimation task in comparison with the state-of-the-art method [13]. We also compared with an approach in conference version [14]. The A-AOG model achieves the state of art performance on both attribute classification and pose estimation tasks.

**Self-collected dataset for joint parsing.** We collected a pedestrian attribute dataset which was introduced in ICCV 2015 [14] for both attribute prediction and pose-estimation tasks. Fig. 2 shows a few example images and this dataset has large variation in margin, size, attribute, pose, appearance, geometry and environment. We list the attribute categories of this dataset in Table 1. We tested the A-AOG model on attribute classification task in comparison with [12] and a method in our conference version [14]. The results demonstrate that human pose and attributes are highly related and that a joint approach is required for better representation and performance.

1.3 Contributions
This paper makes the following contributions:

- We propose a novel A-AOG model which combines phrase structure grammar, dependency grammar, and attributed grammar, and is a simple, explicit
and interpretable representation in comparison to the neural network models used in [9], [11], [12].

- We represent pose-estimation, part-localization and attribute prediction problems in a common A-AOG and solve them through joint parsing. This has the following desired properties: (i) We represent object appearance, geometry and attributes in a unified model and solve the three tasks simultaneously; (ii) We use a single trained model for multiple attribute predictions, unlike previous approaches that require \( n \) models or classifiers for \( n \) attributes; (iii) We do not need any pre-processing such as human detection or part localization.

- The experiments on widely used Attributes of people dataset [2] and Pedestrian attribute dataset [14] show that our method outperforms other state of the art methods on both attribute classification and pose estimation tasks, and demonstrate the benefits and strength of our joint modeling of the two tasks.

In comparison to the conference version [14], this paper provides more details in comparing the three types of grammars and their connections to our A-AOG. We also introduce an efficient way to incorporate deep learned features, and provide extended and improved experimental results with deeper analysis.

The rest of the paper is organized as follows: Section 2 summarizes previous works related to our model. Section 3 overviews the three types of grammar models. Section 4 explains the proposed attribute and-or grammar modeling. Section 5 and 6 present the inference and learning algorithms, respectively. Finally, we present various experiments and analysis of experiment results in Section 7.

2 RELATED WORKS

Our approach is related to three streams of research in the literature which we will briefly discuss in the following.

2.1 Research on attribute grammar

Attribute grammars were first developed in computer science by D. Knuth for formalizing the semantics of context-free languages [15], e.g. in compiler writing, and were adopted by natural language processing (NLP) [16]. It augments the productions rules in context-free grammars by providing context sensitive conditions on the attributes of the symbols. In computer vision, attribute grammar was adopted and extended by Han and Zhu [17] for parsing man-made scenes. The buildings and furniture, floor, etc. in such scenes are decomposed into rectangles with geometric attributes, such as center, and orientations, which follow some constraints, such as alignment in 1D (windows), 2D (floor tiling), and 3D (cubic objects). In contrast to the hard deterministic constraints in compiler writing, the attribute grammar in scene parsing uses soft constraints in the form of energy terms. Later, the attribute grammar is adopted for action representations [18], [19], scene attribute tagging [20], and 3D scene construction from a single view [21]. The grammar rules are manually designed in the scene parsing work [17] and 3D reconstruction work [21]. While the scene attribute tagging work [20] learns the attribute grammar in a weakly supervised way, it requires well-aligned images for each scene categories. In contrast, human images have huge pose variations, and thus the pose estimation is integrated into our approach for joint parsing.

2.2 Research on human attribute classification

In the past five years, the study of attribute classification has become a popular topic in computer vision for its practical importance. Early work focused on facial images since face is the most informative and distinct part of the body, and is the most suitable for estimating attributes such as gender [22], [23], [24], age [25], and some local attributes (e.g., hair style and glasses) [26]. Later, as more diverse attributes (e.g., cloth types and actions) were explored, full body parts were used to collect richer and more diverse information. However, input images cannot be used directly without dealing with the variations of geometry and appearance as the full body has large pose variations. Bourdev et al. introduced a method to classify attributes in [2] by detecting important parts of the body using Poselets [27], while Chen et al. proposed a method to explore human clothing styles with a conditional random field (CRF) in [28] using pre-trained pose estimation [29] output. As these methods used the pre-trained part localization method as a pre-processing step for the sequential steps, the attribute recognition performance undoubtedly relies on the localization accuracy. This approach also prohibits modeling any interaction between the locations of parts and their attribute assignments. Joo et al. designed a rich appearance part dictionary to capture large variations of geometry and pose [8], but their model also did not include any part relations and thus cannot cover large variations of human pose. Zhang et al. made a considerable performance improvement in [9] by using the CNN-based approach. They used the pre-trained HOG based poselet approach for part detection and trained classifier with the shallow convolutional neural network for each poselet. This method also relied on part-based approaches and required the ground-truth bounding box at test time, so it may not be suitable to cover large variation of human pose. More recently, Gkioxari et al. made significant performance improvement in [11], [12]. In [11], three body parts are defined, and jointly fine-tuned with CNN features within the whole body bounding box. In [12], the CNN networks are designed to learn context information for attributes and actions. However, this kind of CNN-based approaches still lack the explicit representation.

2.3 Research on part localization

Localization and detection of human and its parts has been a topic of interest for many years. The pictorial structure model is introduced in the early stage for detection [30] and extended in [31], [32], [33], [34] which used a geometry-based tree model to represent the body. Since then, the deformable part model [35] has became one of the most dominant methods in recent years for detecting humans and other objects [36]. In the last few years, hierarchical mixture models [6], [29], [37], [38] made significant progress which is similar to a dependency grammar. Poselets method [27] used a part-based template to interpolate a pose. K-poselets [39] improved performance by using
poselets as part templates in a DPM model and was based on CNN features [40], [13], [41], [42], [43] showed significant improvement compared to previous methods by training keypoint specific part detectors based on a deep convolutional neural network for human body pose estimation. However, these models did not incorporate any notion of visual attributes explicitly. In comparison, we provide a method for finding informative parts for attributes and their geometrical relations in our model.

3 BACKGROUND ON GRAMMAR MODELS

In this section, we overview the three types of grammars in the literature to provide the necessary background information and then derive the A-AOG as a unification in the following section.

3.1 Phrase structure grammar

The phrase structure grammar, also known as constituency grammar, is based on the constituency relation. The constituency relation defines the rule to break down a node (e.g. parent node) into its constituent parts (e.g. child nodes). In other words, each node must geometrically contain all of its constituents. Phrase structure grammars were introduced in syntactic pattern recognition by K.S. Fu in the early 1980s [44], and rejuvenated into compositional models by Geman et al. [45] and stochastic image grammars by Zhu and Mumford [7]. In the past decade, such grammars have been successfully used in parsing objects and scenes in various form [6], [7], [46]. For example the deformable parts models [5] is a simple context free phrase structure grammar.

Formally, a phrase structure grammar is often formulated as a stochastic context free grammar:

\[ G = \langle s, V_N, V_T, R, P \rangle \]

with five components: a root/initial state \( s \), a set of non-terminal nodes \( V_N \), a set of terminal node \( V_T \), a set of productions rules \( R \), and a probability system \( P \) associated with these rules. Each terminal node in \( V_T \) is represented by an image template or discriminatively trained detector. A typical nonterminal node \( A \in V_N \) is derived by some production rules in the following form,

\[ A \rightarrow \alpha_1 | \alpha_2 | \alpha_3, \quad \text{with } \theta_1|\theta_2|\theta_3, \]

where \( \alpha_i \) is a string of nodes in \( V_N \cup V_T \) and \( \theta_i \) is the branching probability for the three distinct ways for deriving node \( A \). Fig. 3 illustrates a parse tree example. The root node is the upper body and decomposed into torso, head, and hand. The arms are further decomposed into upper arm, lower arm, and hand. It can be described by production rules:

- torso \( \rightarrow \) l. arm, head, torso, r. arm
- l. arm \( \rightarrow \) l. upper arm, l. lower arm, l. hand
- r. arm \( \rightarrow \) r. upper arm, r. lower arm, r. hand

In a general form, the grammar can derive a large number of parse trees for human images depending on the clothing styles, body poses and camera views. Such grammar models have at least two shortcomings in vision tasks.

- It lacks contextual information, such as the correlations between sibling parts or the conditions for expanding an non-terminal node.
- It is often disadvantageous to choose only one of the branches exclusively due to ambiguities in image appearance or because the image templates under different branches are not well-separable. Thus it loses performances to implicit and noisy models like the convolutional neural net.

3.2 Dependency grammar

Dependency grammars have been widely used in natural language processing for syntactic parsing, especially for languages with free word order. It has a root node \( S \) and a set of \( n \) other nodes \( \{ A_1, ..., A_n \} \) with production rules like

\[ S \rightarrow A_1 \mid A_2 \cdots \mid A_n; \quad (1) \]

\[ A_i \rightarrow a_i|A_j \mid A_ja_i; \quad \forall i = 1, 2, ..., n, j \neq i. \quad (2) \]

The root node can transit to any other node once, and then each node \( A_i \) can terminate as \( a_i \) or transit to another node \( A_j \) to the left or right side. Unlike phrase structure grammars, a child node derived from a parent node is not a constituent of the parent but depends on the parent in some semantic relations.

For example, Fig. 4 is a parse tree for the upper body derived from a dependency grammar and is called a kinematic parse tree. The root node is the torso part as it is the center of the body and connected to other parts. The upper arms and head are the child nodes of the torso. It can be described with production rules as:

- torso \( \rightarrow \) l. upper arm, head, r. upper arm
- l. upper arm \( \rightarrow \) l. lower arm
- l. lower arm \( \rightarrow \) l. hand
- r. upper arm \( \rightarrow \) r. lower arm
- r. lower arm \( \rightarrow \) r. hand

L. and R. indicate left and right respectively. The dependency grammar is well suited for representing objects that exhibit large articulated deformations. Body parts at different locations and orientations are treated as different nodes in this grammar. In computer vision, the pictorial model [33] and the flexible mixture of parts model [37] can be viewed as dependency grammars. The advantage of the
In the above rule, the functions \( f \) terms and encodes three types of contextual information.

A non-terminal node \( A \) (or terminal node \( a \)) has attributes denoted by \( A.x \) (or \( a.x \)). These attributes have semantic domains as we shown in Table 1. Such assignments match our intuition in visual perception.

Then for each production rule in a context-free grammar, we can augment a number of conditions or constraints between the attributes of parents and children nodes in the following form,

\[
A \rightarrow aBc \quad \text{s.t.} \quad A.x_1 = f_1(a.x_1, B.x_2, c.x_1) \\
A.x_2 = f_2(a.x_2, B.x_1, c.x_2).
\]

In the above rule, the functions \( f_1 \) and \( f_2 \) impose constraints and conditions for deriving the node \( A \), so that the grammar becomes mildly context-sensitive. Furthermore, they provide means for passing information top-down (for the so-called inherited attributes) and bottom-up (for the so-called synthesized attributes) in the parse tree.

Our A-AOG relaxes the hard constraints to soft energy terms and encodes three types of contextual information:

- Consistency between the same attribute in parent and children nodes, for example, if the root node has gender attribute as female, then its parts should also be female.
- Co-occurrence between attributes, such as a female is more likely to have long hair and wear a skirt.
- Correlations between the assignment of an attribute to a node and its alternative choices, e.g. long hair and short hair will have preferences on the choices of image templates.

### 4 Attribute And-Or Grammar Model

In this section, we present the A-AOG to integrate the three types of grammars into an And-Or graph representation [7].

### 4.1 Attributed And-Or Graph Representation

We construct the And-Or graph in three steps:

1) We use a phrase structure grammar as the backbone of the And-Or graph, which is compositional and reconfigurable, i.e. its parse tree can change structures in contrast to some hierarchical models with fixed structure.

2) We augment the hierarchy with dependency relations, i.e. edges, to encode the articulations between adjacent body parts.

3) We further associate the nodes in the And-Or graph with attributes, which expand the dimensions (or variables) of representation, and introduce additional contextual constraints.

For clarity, we denote the A-AOG by a 4-tuple:

\[
A - AOG = \langle V, E, X, \mathcal{P} \rangle.
\]

1) The vertex set \( V = V_{\text{and}} \cup V_{\text{or}} \cup V_T \) consists of three subsets: (i) a set of and-nodes \( V_{\text{and}} \) for decomposition in the phrase structure grammar; (ii) a set of or-nodes \( V_{\text{or}} \) for branching to alternative decompositions and thus enabling reconfiguration; and (iii) a set of terminal nodes \( V_T \) for grounding on image features. Each node \( v \in V \) represents a body part of different granularities and has state variables designating the location \( (x,y) \). In fact, the state variables are the geometric attributes which are propagated between parent-children nodes in the hierarchy. However, in this paper, we treat them separately from the human attributes.

We define 14 atomic, i.e. terminal, parts, head, torso, l.shoulder, r.shoulder. l.upper arm, l.lower arm, r.upper arm, r.lower arm, l.hip, r.hip, l.upper leg, l.lower leg, r.upper leg, and r.lower leg. These parts are defined as terminal nodes \( V_T \) in the grammar. We then define non-terminal nodes \( V_T \), upper body and lower body by combining terminal parts. Upper body part includes head, torso, shoulders, and arms. Lower body part includes hips and legs. The root part, full body, is defined by upper body and lower body. We illustrate the defined grammar in Fig. 6.

2) The edge set \( E = E_{\text{psg}} \cup E_{\text{dg}} \) consists of two subsets: (i) a set of edges with phrase structure grammar \( E_{\text{psg}} \), and (ii) a set of edges with dependency grammar \( E_{\text{dg}} \).

3) The attribute set \( X = \{x_1, ..., x_9\} \) are associated with nodes in \( V \).

4) \( \mathcal{P} \) is the probability model on the graphical representation.

Define the parse graph

\[
p_g = (pt, E(pt), X(pt))
\]

where \( E(pt) \) and \( X(pt) \) are the set of edges and attributes respectively in parse tree \( pt \). Fig. 5 shows a parse tree example derived from the A-AOG, which includes a parse tree for human body detection and pose and a parse tree for human attributes.

### 4.2 Formulation of Joint Pose and Attribute Parsing

Attribute state is represented by a set of attributes for each part. Then, the goal is to find the most probable parse tree \( pt \) from the designed grammar model given an image \( I \). The probability model over the parse tree is formulated in a
Bayesian framework, which computes the joint posterior as the product of a likelihood and prior probability, and equivalently represented as the following the Gibbs distribution
\[
P(pt | I; \lambda) \propto P(I | pt; \lambda) P(pt; \lambda) = \frac{1}{Z} \exp \{-E(I | pt; \lambda) - E(pt; \lambda)\}
\]

(4)

The model parameters are denoted as \( \lambda \). The energy functions \( E \) are further decomposed into a set of potential functions. These potentials constrain all aspects of the grammar. The likelihood term describes appearance response and is decomposed into part and attribute appearance.

\[
-E(I | pt; \lambda) = -E^V_{\text{app}}(I | pt; \lambda) - E^A_{\text{app}}(I | pt; \lambda)
\]

(5)

\( E^V_{\text{app}}(I | pt; \lambda) \) and \( E^A_{\text{app}}(I | pt; \lambda) \) are appearance terms for part and attribute respectively. The prior term is used to describe relations in grammar. It is also decomposed into part and attribute relations.

\[
-E(pt; \lambda) = -E^V_{\text{rel}}(pt; \lambda) - E^A_{\text{rel}}(pt; \lambda)
\]

(6)

\( E^V_{\text{rel}}(pt; \lambda) \) and \( E^A_{\text{rel}}(pt; \lambda) \) are relation terms for part and attribute respectively. We rewrite Equation 4 as

\[
P(pt | I; \lambda) = \frac{1}{Z} \exp \{-E^V_{\text{app}}(I | pt; \lambda) - E^A_{\text{app}}(I | pt; \lambda) - E^V_{\text{rel}}(pt; \lambda) - E^A_{\text{rel}}(pt; \lambda)\}
\]

(7)

We, then, can express energy terms as scoring functions.

\[
S(pt, I) = -E^V_{\text{app}}(I | pt) - E^A_{\text{app}}(I | pt) - E^V_{\text{rel}}(pt) - E^A_{\text{rel}}(pt)
= S^V_{\text{app}}(I, pt) + S^A_{\text{app}}(I, pt) + S^V_{\text{rel}}(pt) + S^A_{\text{rel}}(pt)
\]

(8)

We now have four scoring functions.

- \( S^V_{\text{app}}(I, pt) \) part appearance score function
- \( S^A_{\text{app}}(I, pt) \) attribute appearance score function
- \( S^V_{\text{rel}}(pt) \) part relation score function
- \( S^A_{\text{rel}}(pt) \) attribute relation score function

The choice and particular forms for these scoring functions vary on the design and intention of the grammar, which we explore in the following sections.

### 4.3 Part Model

#### 4.3.1 Part Relation Model

The term \( S^V_{\text{rel}}(pt) \) is for the part relation score. We define two relation types: syntactic relation (green edges in Fig. 5) and kinematic relation (red edges in Fig. 5). Syntactic relation follows phrase structure grammar, and controls the part composition. Kinematic relation follows dependency grammar rule, and describes the articulation constraints. The overall relation score sums each of these relations.

Each part \( v_i \) is associated with \((x_i, y_i)\) as its position and \( t_i \) as its part type. The syntactic relation is defined between parts to represent the co-occurrence frequency of two neighboring part types, and captures correlations and compatibilities between the parts. It is described by score function \( S^S_{\text{rel}}(v_i, v_j) \), and the \( S^S_{\text{rel}} \) over parse tree is computed as:

\[
S^S_{\text{rel}}(pt) = \sum_{(i, j) \in E^S_{\text{rel}}(pt)} S^S_{\text{rel}}(v_i, v_j)
\]

(9)

where \( E^S_{\text{rel}}(pt) \) is the set of edges with phrase structure grammar in parse tree \( pt \). Here \( S^S_{\text{rel}}(v_i, v_j) = \log P(t_i, t_j) \) where \( P(t_i, t_j) \) is the probability that part type \( t_i \) occurs with \( t_j \).

The kinematic relation is defined between parts by and-rule from and-or grammar. And-rule can be viewed as the rule for assembly of constituent parts and enforce geometric constraints between two parts \((v_i, v_j)\) with relative geometry of articulation. It is described by score function \( S^K_{\text{rel}}(v_i, v_j) \). \( S^K_{\text{rel}} \) over the parse tree is computed as:

\[
S^K_{\text{rel}}(pt) = \sum_{(i, j) \in E^K_{\text{rel}}(pt)} S^K_{\text{rel}}(v_i, v_j)
\]

(10)

where \( E^K_{\text{rel}}(pt) \) is the set of edges with dependency grammar in parse tree \( pt \). \( S^K_{\text{rel}}(v_i, v_j) = \log P(v_i, v_j) \) where \( P(v_i, v_j) \) is the probability of the mixture of Gaussian on \((x_i - x_j, y_i - y_j)\). We write part relation score function \( S^V_{\text{rel}}(pt) \) as below.

\[
S^V_{\text{rel}}(pt) = S^S_{\text{rel}}(pt) + S^K_{\text{rel}}(pt)
= \sum_{(i, j) \in E^S_{\text{rel}}(pt)} S^S_{\text{rel}}(v_i, v_j) + \sum_{(i, j) \in E^K_{\text{rel}}(pt)} S^K_{\text{rel}}(v_i, v_j)
\]

(11)
4.4.1 Attribute Node

We can treat the set of attributes \( A(v) \) for part \( v \) as a two layered simple tree which follows and-or grammar rules as illustrated on the right side in the Fig. 6. The root node of \( A(v) \) is described by and-rule in and-or grammar. It includes corresponding attributes for part \( v \) as its child nodes. Then, each attribute includes attribute types as child nodes, it follows or-rule. It has two attribute types for binary attribute class, such as ‘gender’ and ‘wearing t-shirt’, or have more than two for multi-class attributes, such as ‘cloth types’ or ‘age’. It can be described in production rule. For example, when part \( v \) has two corresponding attributes, ‘gender’ and ‘wearing t-shirt’, ‘gender’ can have types ‘male’ and ‘female’. ‘wearing t-shirt’ has two child nodes, ‘yes’ and ‘no’. In production rule, it could be written as follow,

\[
\begin{align*}
\text{attribute production rule} & \quad \text{example} \\
A(v) & \rightarrow \{A_1, A_2\} \quad A(v) \rightarrow \{\text{Gender}, \text{T-shirt}\} \\
A_1 & \rightarrow A_{11} | A_{12} \quad \text{Gender} \rightarrow \text{Female} | \text{Male} \\
A_2 & \rightarrow A_{21} | A_{22} \quad \text{T-shirt} \rightarrow \text{Yes} | \text{No}
\end{align*}
\]

4.4.2 Attribute Relation Model

Each attribute node \( A(v_i) \) is linked to part \( v_i \) through a relation as shown by blue edges in Fig. 5 and a dashed line in Fig. 6. This relation describes properties of part node \( v_i \) and reflects the co-occurrence frequency of the attribute given the part type. For example, let the specific part type of node \( v \) (= upper body) have an appearance that is blouse-like. This will occur more frequently with female than male, and therefore the model should favor selecting this part when there is strong evidence for the female attribute. It is described by score function \( S_{rel}^A(v_i, A(v_i)) \), and \( S_{rel}^A \) over \( pt \) is computed as

\[
S_{rel}^A(pt) = \sum_{v_i \in V(pt)} S_{rel}^A(v_i, A(v_i)) \quad (13)
\]

4.4.3 Attribute Appearance Model

Just as we defined appearance model for part, so too we define appearance model for attribute. It corresponds to the scoring function \( S_{app}^A(pt, I) \), and it can be computed from attribute score function \( S_{app}^A(A(v_i), I) \). It indicates the local appearance response of attributes of part \( v_i \) at image path centered at \( v_i = (x_i, y_i) \). The score \( S_{app}^A \) over \( pt \) is computed as

\[
S_{app}^A(pt, I) = \sum_{v_i \in V(pt)} S_{app}^A(A(v_i), I) \quad (14)
\]

4.5 Combine Appearance Models

We defined two appearance models: part appearance \( S_{app}(v_i, I) \) and attribute appearance \( S_{app}(A(v_i), I) \). They...
are now connected through the part-attribute relation, and we can combine those two appearance score functions into single function $S_{\text{app}}(v_i, a_j, I)$ where $a_j \in A(v_i)$. In order to capture the diverse appearance of part and attributes under different viewpoints and poses, we borrow the strength from a deep CNN model. At the last layer of our CNN model, we can directly get $P(v_i, a_j, t_m | I)$ which is the likelihood of the image patch that belongs to part $v_i$ with part type $t_m$ and attribute $a_j$. The score function $S_{\text{app}}(v_i, a_j, I) = \log(P(v_i, a_j, t_m | I))$. The total appearance score over $pt$ is computed as:

$$S_{\text{app}}(pt, I) = \sum_{v_i \in V(pt)} \sum_{a_j \in A(v_i)} S_{\text{app}}(v_i, a_j, I)$$  

(15)

We will explain how we utilize a CNN model to obtain part and attribute scores together in Section 6.

5 PARSING AND INFERENCE

We defined grammar structure for describing human body with attributes. The inference task is now equivalent to finding the most probable parse tree $pt^*$ from the constructed grammar model for given image $I$. We can find $pt^*$ by maximizing the score functions described in previous sections.

$$pt^* = \arg \max_{pt} P(I | pt)P(pt)$$

$$= \arg \max_{pt} [S_{\text{app}}^V(pt, I) + S_{\text{app}}^A(pt, I) + S_{\text{rel}}^V(pt) + S_{\text{rel}}^A(pt)]$$  

(16)

We combine part appearance score $S_{\text{app}}^V(pt, I)$ and attribute score $S_{\text{app}}^A(pt, I)$ into $S_{\text{app}}(pt, I)$. And, the part and attribute relation $S_{\text{rel}}(pt)$ will also be captured by $S_{\text{app}}(pt, I)$ because we model them in the same CNN model. Equation 16 can be rewritten as:

$$pt^* = \arg \max_{pt} [S_{\text{app}}(pt, I) + S_{\text{rel}}(pt)]$$

(17)

We now denote part relation score function $S_{\text{rel}}^V(\cdot)$ by $S_{\text{rel}}(\cdot)$ to simplify the equation. In the conference version [14], we used Equation 16 to find the optimal parse tree $pt^*$, but we maximize the Equation 17 in this paper. We compare two methods quantitatively in Section 7.

In the first step of inference, in order to reduce the searching space of the parse tree, we generate the proposals for each part by the deep CNN model instead of computing response maps on image pyramid. Each part proposal is associated with attributes. After having the proposals for each part: $O_{v_0}, O_{v_1}, ..., O_{v_n}$, we can obtain the final parse tree $pt^*$ with Equation 17. To maximize $S(pt, I) = S_{\text{app}}(pt, I) + S_{\text{rel}}(pt)$, we introduce the total score function $S_{\text{tot}}(v_o, I)$ which is formulated in a recursive way:

$$S_{\text{tot}}(v_i, I) = \sum_{j=1}^{A} S_{\text{app}}(v_i, a_j, I)$$

$$+ \sum_{v_j \in C(v_i)} [S_{\text{rel}}(v_i, v_j) + S_{\text{tot}}(v_j, I)]$$  

(18)

where $A$ is the number of attribute categories and $C(v_i)$ is the set of children of part $v_i$, and $v_i \in O_{v_o}$. The score function $S(pt, I)$ is equivalent to $S_{\text{tot}}(v_o, I)$ where $v_o$ is the root part. We now can infer the parse tree $pt^*$ by maximizing $S_{\text{tot}}(v_o, I)$, and it can be expressed as

$$pt^* = \arg \max_{pt} S_{\text{tot}}(v_o, I)$$  

(19)

We can also infer the parse tree $pt^*$ in a different way by using the attribute as a global constraint which means all parts should have the same attribute. We first find attribute specific parse tree $pt_{a_j}$, $pt_{a_j}$ denotes the parse tree under $j$-th attribute $a_j$. To infer parse tree $pt_{a_j}$, we maximize $S(pt_{a_j}, I) = S_{\text{app}}(pt_{a_j}, I) + S_{\text{rel}}(pt)$ where $S_{\text{app}}(pt_{a_j}, I) = \sum_{v_i \in V(pt_{a_j})} S_{\text{app}}(v_i, a_j, I)$. To maximize $S(pt_{a_j}, I)$, we use score function $S_{\text{tot}}(v_i, a_j, I)$

$$S_{\text{tot}}(v_i, a_j, I) = S_{\text{app}}(v_i, a_j, I)$$

$$+ \sum_{v_j \in C(v_i)} [S_{\text{rel}}(v_i, v_j) + S_{\text{tot}}(v_j, a_j, I)]$$  

(20)

The score function $S(pt_{a_j}, I)$ is now equivalent to $S_{\text{tot}}(v_0, a_j, I)$, and the inference of parse tree can be expressed as $pt^*_{a_j} = \arg \max_{pt_{a_j}} S_{\text{tot}}(v_0, a_j, I)$. Commonly this maximization problem can be computed using dynamic programming (DP), however, the DP cannot be used because of many loopy cliques in our model due to the combination of phrase structure grammar and dependency grammar. Here we applied a greedy algorithm based on the beam search.

We start with the parse tree which only includes one root part $v_0$ and generate $N_o = |O_{v_0}|$ parse tree candidates $pt_1', ..., pt_N'$ from the proposals $O_{v_0}$. The score of each parse tree candidate $S(pt_i')$ is computed as $S_{\text{app}}(v_0, a_j, I)$, then one child part of the root part $v_i \in O_{v_0}$ is added into parse tree and generate $N_0' = N_0 \times |O_{v_i}|$ parse tree candidates. The score $S(pt_i')$ is updated by adding the part relation score $S_{\text{rel}}(v_i, v_j)$. Only top-K high scored parse tree candidates are kept into next step. We continue adding child part of...
generate part proposals from input image

combine with keypoint annotation

find pos and neg part proposals

parts in previous approaches

parts in our approach

Fig. 8. We generate part proposals from input image and find positive and negative proposals using keypoint annotation information.

Fig. 9. In previous approaches, (left) parts are defined by drawing square bounding box around keypoint [13], [37] or by annotating precise bounding box [6]. (right) We define parts based on our part proposal process, and it handles geometric and scale variation of parts efficiently and effectively.

current parse tree, updating the scores and pruning candidates until all parts are added into the parse tree. Finally we pick the parse tree candidate with highest score as the inferred parse tree \(pt^*_a\).

We have \(pt^*_a\) as many as the number of attributes. For example, we have \(9 \times 2\) attribute specific parse trees \(pt^*_a^j\), where \(j = \{1, 2, \ldots, 18\}\) in the experiment for Attribute of People dataset which define 9 binary attributes. The final parse tree \(pt^*\) on image \(I\) is inferred as:

\[
pt^* = \underset{pt^*_a}{\arg\max} S(pt^*_a, I)
\]

(21)

We compare two pose inference methods in Fig. 7. As we can see from the figure, when we do not include attribute constraints by using Equation 18, the model selects parts that maximize the scores locally and the parts could come from different people. However, when we have global attribute constraints with Equation 20, we can control the model to have consistent attributes and can provide better pose estimation.

We compute the score of attribute \(S(a_j)\) from each parse tree \(pt^*_a^j\), and it is the summation of part score on the parse tree \(pt^*_a^j\) under attribute constraints.

\[
S(a_j) = \sum_{v \in V(pt^*_a^j)} S_{app}(v_i, a_j, I) \cdot I(a_j, A(v_j))
\]

(22)

\(A(v_j)\) is the set of attributes corresponding to part \(v_i\) and defined in Section 4.4.2. \(I(\cdot)\) is an indicator function.

6 Learning

Our learning procedure contains two stages: the first stage learns deep CNN models for part proposals generation, and computation of the likelihood of part and attribute, and part positions regression; the second stage learns the geometric relations between parts, and the compatibility relations between parts and attributes.

6.1 Part proposal and appearance model learning

Although Faster R-CNN [47] is used for generic object detection task, it has not been used in the task of fine-grained object recognition and detection, e.g., attribute classification and part localization. [47] trained two networks: region proposal network for object proposal generation and fast-rcnn network [48] for object recognition. Inspired by the success in [47], we design two networks: part proposal network to generate part proposals; the part-attribute network to generate likelihood for all part-attribute combinations using classification layer and predict positions of parts using a regression layer. The two networks share weights of bottom layers. The two networks are initialized with the same pre-trained models in [47].

We use strong supervision which includes the 14 joints of human (see Figure 14), the bounding box of the target person and the attribute labels. To train the part proposal network, we first generate 9 bounding boxes (= 3 scales \(\times 3\) aspect ratios) which correspond to 9 part types at each location, and then compute the overlap between 9 bounding boxes and the ground-truth bounding box for each location. The locations are predicting the part proposals if the overlap is bigger than 0.5. From the part proposal network, we can generate proposal set \(O_i\) for each part \(v_j\). The part type \(t_p \in \{1, 2, \ldots, 9\}\) for each proposal is the index of the predicted bounding box with highest probability. To train part-attribute network, we decide the training labels for each proposal by the following process: (1) compute the overlap between the proposed and the ground-truth human bounding box; (2) label the proposals with overlap lower than 0.5 for negatives. (3) select the proposals with overlap higher than 0.7 and compute the minimum distance \(d_k\) between each proposal and all parts as \(\min_{p, k} (d_k)\). \(d_k\) is the center of a proposal \(k\), \([x_k, y_k]\) is the keypoint of part \(i\), and \([w_k, h_k]\) are the width and height of the proposal, and \(n\) is the total number of parts. The keypoint of atomic part is defined as its joint and the keypoint of a non-terminal part (upper-body, lower-body, full-body) is the center of joints included in this part. We also record the part index \(I_p\) which gives the minimum distance. \(I_p\) is the index of a non-terminal part if the proposal includes all the joints of this part, or the index of an atomic part otherwise. (4) keeps the proposals whose \(d_k\) are smaller than 0.5. Each proposal is labeled as the part index \(I_p\) and part type \(t_p\).

We illustrate the part proposal process in Fig. 8 and compare our part design with previous approaches in Fig. 9. From our approach for part design, we can handle large variation of part scale and aspect ratio.

In part-attribute network, the output of the classification layer is the likelihood of all part-attribute combinations for each proposal region. The length of the output is \(17 \times 9 \times k + 1\)
in which 17 is the number of parts (14 atomic parts + 2 mid-level parts + root part) and 9 is the number of part types and \( k \) is the number of attribute categories, and +1 is for negatives (i.e., background). The output of the regression layer are the keypoint positions of parts.

### 6.2 Part relation and part-attribute relation learning

The part relation includes the syntactic relation between parent and children parts, and kinematic relation between atomic parts. The syntactic relation is defined as

\[
S_{rel}^S(v_i, v_j) = \log P(t_i, t_j) \quad \text{and} \quad P(t_i, t_j) \text{ is from the normalized histogram of co-occurrence of part types.}
\]

We use the mixture of Gaussian to measure the part relation \( P(v_i, v_j) \) and the score \( S_{rel}^S(v_i, v_j) = \log P(v_i, v_j) \), and it penalizes the displacement of the \( i \)-th part and \( j \)-th part. The number of mixtures is set to 10.

The part-attribute relation describes the compatibility relationship between part and attribute. In the Attributes of People dataset [2], about one third of the images are annotated unknown for each attribute, and many joints are also annotated invisible due to the occlusion or truncation of human body. In most cases, the attributes are annotated unknown because of the invisible specific parts. For example, the attribute ‘long pants’ of most images are unknown because only the upper body is visible. Thus, these annotations actually provide strong evidence about which parts play a role in distinguishing the attribute. To find associated part for each attribute, we compute mutual information between attributes with label ‘known’ and ‘unknown’ and 14 atomic parts with label ‘visible’ and ‘invisible’. We show the computed mutual information in Fig. 10 for four attributes.

For each attribute \( a_j \), we pick the parts of which the values are above the mean value as the associated part set. If a part is picked then its parent part will also be picked. For example, ‘leg’ is the related part for ‘jeans’, and ‘lower body’ and ‘full body’ are considered as related parts for ‘jeans’ as well. We denote corresponding attributes for each part by \( A(v) \).

### 7 EXPERIMENTS

We conduct three sets of experiments on joint inference of human attribute and pose. The first set of experiments evaluates our method on attribute classification, and compares against the previous approaches in [8], [9], [11], [12], [14]. The second set tests pose estimation, and compares against the state-of-the-art method in [15]. The last set is diagnostic analysis and compares our joint model with its different variants and settings. In particular, we show that pose estimation improves attribute classification and vice versa.

#### 7.1 Benchmarks

**Attributes of People dataset.** Introduced by [2], this dataset consists of 4013 training images and 4022 testing images. Each image is centered at the target person which is annotated as a bounding box. This dataset defines 9 binary attributes, and keypoint annotations can be used for training. This dataset is challenging for attribute classification and pose estimation because the person is always occluded and truncated, and the pose variation is very large.

**Pedestrian Attribute dataset.** We collected and annotated a Pedestrian attribute dataset. It provides part and attribute annotations as illustrated in Figure 2. It is designed to have one person in an image, and does not provide a bounding box of the target human at test time. It includes 2257 high resolution images of which 1257 images are for training and 1000 images for testing. It consists of many types of variations in attribute, pose, appearance, geometry and environment. The 16 joint positions and labels of 9 attributes are provided.

#### 7.2 Evaluation on Attribute Classification

Table 2 compares the average precision of attribute classification between our method and other methods [8], [9], [11], [12], [14] on Attributes of People dataset. All the methods from the 1st row to the 10th row use the ground-truth bounding box of the target person at test time. To compute the log likelihood of part and attribute in our structure, we train our 8-layer networks and 16-layer networks based from the 1st row to the 10th row use the ground-truth bounding box of the target person at test time. To compute the log likelihood of part and attribute in our structure, we train our 8-layer networks and 16-layer networks based on the same pre-trained networks used in [47]. For attribute classification evaluation, we compute the score using Equation 22. Note that [8] did not use joint annotations during training, and [9] trained model on a different training dataset (25K images) which is much larger than the original set. Our method with pose (i.e., joint model) outperforms the state-of-the-art methods [11] using 8-layer and 16-layer network architectures respectively.

We also compare our approach when ground truth bounding boxes of the target persons are not given at test time. The results are shown from the 11th row in Table...
In [11], they use R-CNN [40] person detection as the preprocessing step to detect the target person. However, our approach detects people, classifies attributes, and estimates human pose simultaneously in a unified framework. Overall, our method with 8-layer network improves mAP of same network architecture by 6.4% point, and even achieves 2.0% point better mAP than 16-layer network in [11]. In addition, we achieve 5.8% point better performance than 16-layer network architecture in [11] using the same network. We also show a tremendous improvement from our conference version [14]. It is important to note that we perform better with the same network architecture, and it demonstrates the advantage of the joint modeling in which pose-estimation and attribute prediction help each other during training and inference.

We compare the attribute classification performance on self-collected Pedestrian attribute dataset on Table 3. We outperform our conference version [14] substantially on all attributes without the ground truth boxes at test time. To have a better comparison, we train the model of R* CNN [12] on this dataset. This dataset is designed without providing ground truth bounding box at test time, but R* CNN requires the ground truth bounding box of target human. In order to have a fair comparison, we use the person detection method in [47] to generate bounding boxes for R* CNN instead of using ground truth bounding boxes. We only train and evaluate binary attributes, because R* CNN is designed for binary attribute classification. We show better performance on 3 binary attributes out of 4. We show some examples of our results in Figure 13.

### 7.3 Evaluation on Pose Estimation

We show the pose estimation results on the Attributes of People dataset in Table 4. A widely used evaluation method, strict Percentage of Correct Part (PCP) is used as the evaluation metric to compare with the state-of-the-art method in [13]. Unlike traditional pose estimation methods which are designed to test on the image with one person and small margin, our approach does not require a bounding box because we detect human and estimate pose. For a better and fair comparison we conduct experiments under two different settings. In the first setting, ground truth bounding boxes are provided at test time, while the second setting does not have such. Overall, our method shows a substantial improvement on both settings. We show part indexes in Fig. 14 (b). We believe it is because previous approaches conduct experiments on images with similar scales for human detection, but images in this dataset has
large scale and appearance variations, and heavy occlusion and truncation. For instance, the traditional pose estimation dataset in [4] is scaled to contain people of roughly 150 pixels in height; however, the smallest height of humans present in this dataset is 72 pixels and the largest is 1118 pixels in this attribute dataset.

We show examples of pose-estimation in Fig. 14 (a). The top row shows successful examples, and the bottom row shows failures. As we can see from the examples, our method handles large variation of human pose, occlusion and truncation very well. Although we improve the pose-estimation in multiple people in the scene using the attribute constraints as shown in Fig. 7, most of our failures come from the situation with multiple people because it is hard to distinguish part identities when people have the same attributes.

7.4 Diagnostic experiments
To better justify the contribution of joint modeling of attribute and pose, we have two diagnostic comparisons: 1) attribute prediction without pose; 2) pose estimation without attributes.

Attribute prediction without pose. Instead of inferring the best pose for each attribute and computing the corresponding attribute score, we pick the highest score among all part proposals for each attribute and use that score as the attribute classification score. The mean average precision shown in Table 2 is 2.6% (8-layer) and 2.7% (16-layer) point lower than the result from our joint model without or-nodes when a ground truth bounding box is not provided. It demonstrates that the attributes can be improved when the pose is inferred together. Impressively, it is still better than [11] with the same 8-layers network architecture because we detect parts with a large number of different scales and aspect ratios instead of 200 poselets used by [11]. We also see similar result when a ground truth bounding box is provided.

Pose estimation without attribute. We infer the pose using the part proposals from all attributes instead of one pose for each attribute. In other words, we use Equation 18 instead of Equation 20. In this approach, \( p^* \) provides the pose by allowing parts to have inconsistent attributes because we maximize the attribute score for each part. This approach can cause problems at pose estimation by choosing parts from different people who are close to each other. This kind of mistake happens frequently in existing pose estimation approaches, and that’s why they require the bounding box of target person at test. In Table 4, the performance of pose estimation is decreased by 4.6% point from our joint modeling with attribute and without or-nodes. From Fig. 7, we can clearly see that the parts of the estimated pose are from different person in the image if no attribute information is used. However, this problem is solved to some extent when we infer the best pose for each attribute because different people from the same image may have different attributes.

8 Conclusion
This paper presents an attributed and-or grammar to describe compositionality, reconfigurability, articulation, and attributes of human in a hierarchical joint representation. Our approach parses an input image by inferring the human, body pose, parts, and attributes. The advantage of our approach is the ability to perform simultaneous attribute reasoning and part detection, unlike previous attribute models that use large numbers of region-based attribute classifiers without explicitly localizing parts.

Our method currently requires large keypoint annotations, unlike recent approaches did not. The keypoint is required to parse pose and attribute jointly. If we use the small number of parts, e.g. head, upper body, lower body, we can avoid large keypoint annotations but we still can learn relations between part and attribute.

We demonstrate our model on benchmarks, and achieve the state-of-the-art attribute classification and pose estimation performance against recent methods. We also show our joint modeling for part and attribute model has better results on the pose-estimation task than previous pose-estimation approaches. We believe that evidence of the attributes as well as their consistency constraints can lead to performance improvements on both tasks.

Acknowledgments
This work was supported by a DARPA SIMPLEX project N66001-15-C-4035, ONR MURI project N00014-16-1-2007, and NSF IIS 1423305. The authors’d like to thank Dr. Brandon Rothrock for discussions.

References
Fig. 11. Most positive attribute prediction on attributes of people dataset [2]. We cropped the image around the ground truth bounding box for display purpose.

Fig. 12. Most negative attribute prediction on attributes of people dataset [2]. We cropped the image around the ground truth bounding box for display purpose.

Fig. 13. Output examples on Pedestrian attribute dataset. First six examples show successful examples with high scores. Rest of examples show failure examples with high scores. Actual testing images have large margin, but we crop the image around the ground truth bounding box for visualization.

Fig. 14. (a) Examples of pose-estimation result on Attributes of People dataset. First row shows success examples and second row shows failure examples. Most of failures are caused by multiple people in the image. (b) We define 14 keypoints and 13 sticks.