Attributed Grammars for Joint Estimation of Human Attributes, Part and Pose

Seyoung Park and Song-Chun Zhu
Center for Vision, Cognition, Learning and Autonomy,
Department of Computer Science and Statistics, UCLA
{seypark@cs, sczhu@stat}.ucla.edu

Abstract

In this paper, we are interested in developing compositional models to explicit representing pose, parts and attributes and tackling the tasks of attribute recognition, pose estimation and part localization jointly. This is different from the recent trend of using CNN-based approaches for training and testing on these tasks separately with a large amount of data. Conventional attribute models typically use a large number of region-based attribute classifiers on parts of pre-trained pose estimator without explicitly detecting the object or its parts, or considering the correlations between attributes. In contrast, our approach jointly represents both the object parts and their semantic attributes within a unified compositional hierarchy. We apply our attributed grammar model to the task of human parsing by simultaneously performing part localization and attribute recognition. We show our modeling helps performance improvements on pose-estimation task and also outperforms on other existing methods on attribute prediction task.

1. Introduction

In this paper, we design and propose compositional models to explicit representing pose, parts and attributes and tacking the tasks of attribute recognition, pose estimation and part localization jointly. It is different from the recent trend of using Convolutional Neural Networks(CNNs) approaches with large data for training and testing on these tasks separately. Despite the CNNs-based approaches have been significantly improving the performance on many vision tasks, but lack explicit models. In contrast, stochastic grammar models have been successfully used in explicit representing and inferring complex compositional structures for tasks such as object detection[19, 23, 43], hierarchical object representation[7, 14], pose estimation[32, 38], scene parsing [42], and event prediction[30]. However, these models do not incorporate any notions of visual attributes. Current attribute models for objects and scenes are neither compositional nor part-based, and typically employ a large number of independent classifiers to decide which attribute labels to assign to a given bounding box[3, 9, 25, 28]. Such an approach does not explicitly capture the relationship between attribute labels, or localize the part regions described by the attributes. So, our work is now an attempt to propose an unified model that to strengthen the strength and make up for the weakness of both stochastic grammar model and attributed model. Our attribute grammar model is an extension of conventional grammar models: Dependency Grammar(DG), Pharase Structure Grammar(PG), and And-Or Grammar(AOG) [43]. We use PG for part representation, DG for articulated relations, and both can be described in AOG. We, then, extend it by including attribute notating for each part as shown in 1. Since the model output includes representations of object itself and attributes, it allows us to use the model as the application for attribute classification and pose estimation automatically and simultaneously. This is a novel because previous approaches need n attribute classifier for n attribute classifications[2, 25, 41], and rely on pre-trained

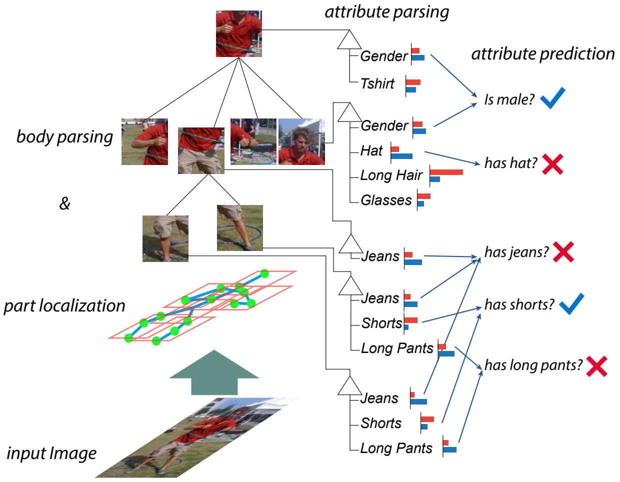


Figure 1. Paring and attribute prediction from an input image using proposed attributed model. The parse tree represents human body part appearance, geometry, and attributes for each part. Then, we aggregate local information for attribute prediction.

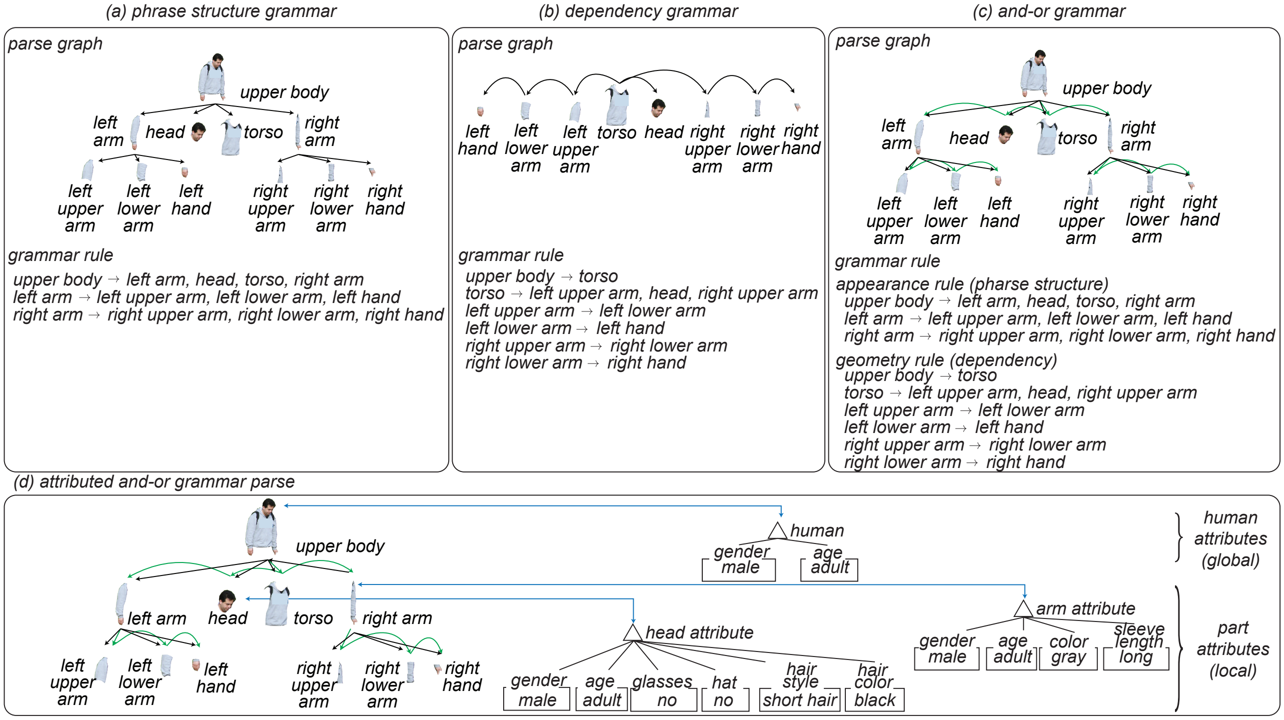

(a) phrase structure grammar
parse graph
upper body
left arm
head
torso
right arm
left upper arm
left lower arm
left hand
right upper arm
right lower arm
right hand
grammar rule
upper body → left arm, head, torso, right arm
left arm → left upper arm, left lower arm, left hand
right arm → right upper arm, right lower arm, right hand
(b) dependency grammar
parse graph
left hand
left lower arm
left upper arm
torso
head
right upper arm
right lower arm
right hand
grammar rule
upper body → torso
torso → left upper arm, head, right upper arm
left upper arm → left lower arm
left lower arm → left hand
right upper arm → right lower arm
right lower arm → right hand
(c) and-or grammar
parse graph
upper body
left arm
head
torso
right arm
left upper arm
left lower arm
left hand
right upper arm
right lower arm
right hand
grammar rule
appearance rule (pharse structure)
upper body → left arm, head, torso, right arm
left arm → left upper arm, left lower arm, left hand
right arm → right upper arm, right lower arm, right hand
geometry rule (dependency)
upper body → torso
torso → left upper arm, head, right upper arm
left upper arm → left lower arm
left lower arm → left hand
right upper arm → right lower arm
right lower arm → right hand
(d) attributed and-or grammar parse
upper body
left arm
head
torso
right arm
left upper arm
left lower arm
left hand
right upper arm
right lower arm
right hand
human
gender male
age adult
human attributes (global)
head attribute
gender male
age adult
glasses no
hat no
hair style short hair
hair color black
arm attribute
gender male
age adult
color gray
sleeve length long
part attributes (local)

Figure 2. (a) Each grammar rule decomposes a part into smaller constituent parts. (b) Each grammar rule defines adjacency relations that connects the geometry of a part to its dependent parts. (c) It provides the framework to represent both dependency grammar (green edges) and phrase structure grammar (black edges). (d) Each node has their own attribute graph node connected by attribute relations.

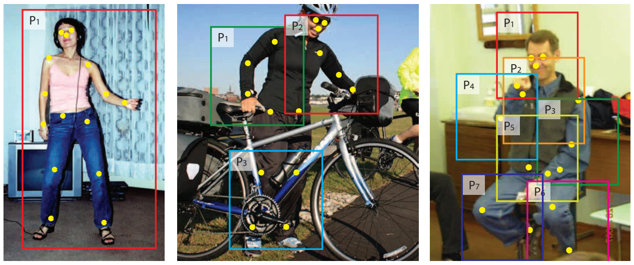
pose-estimator or part detector[4, 2, 41]. We demonstrate our technique in human attribute prediction and pose estimation tasks and indeed observe an improvement in the parsing performance from using these constraints. To see effectiveness of our joint modeling, we use common pretrained CNN-based features with HOG and color feature as our base feature models. Then, we compare the performance with previous approaches and our attributed grammar modeling to see how our modeling helps the performance improvement without fine-tuning of features and having additional data.

2. Related works

Our work is related to 3 research streams in the literature.

Attribute Models. The study of attributes has become a popular topic because of its practical importance. In early work, people focused on facial images because the face is the most informative and distinct part of the body, and suitable for estimating attributes such as gender [6, 21, 29], age[27], and more general attributes (*e.g.* hair style, glasses) [26]. Later, as more diverse attributes (*e.g.* cloth types, actions) are explored, full body parts are used to collect more rich and diverse information. However, as the full body has large pose variations, input images cannot be used directly without dealing with the variation of geometry and appearance. [2] introduced a method to classify by detecting important parts of the body using Poselet[3], [4] proposed a method to explore human clothing with a CRF using pretrained pose estimation[38] output. But, as these methods use the pre-trained localization method as a pre-processing step, performance undoubtedly relies on the localization performance. This approach also prohibits modeling any interaction between the locations of parts and their attribute assignments. [25] designed a rich appearance part dictionary to capture large variations of geometry and pose, but it also does not include any part relations and cannot handle large variation of human pose. Recently, [41] made significant performance improvements. They used pre-trained HOG based Poselet approach for part detection and trained classifier with shallow CNN for each Poselet. But, this method also relies on part-based approaches, and it is not suitable for large variation of human pose. This model improves the performance by having large data with fine-tuning of feature rather than designing better model, but our approach more focuses on modeling and representation.

Part localization Models: Localization and detection of human and its parts has been a topic of interest for many years. The pictorial structure model is introduced in early stage for detection[15] and extended by [1, 8, 13, 33] which uses a geometry-based tree model to represent the body. Then, the deformable part model[10] has become one of the most dominant methods in recent years for detecting humans and other objects[19]. Later, hierarchical mixture



P = 1 P = 3 P = 7

Figure 3. We find part that include all visible keypoints in the examples. If P is small number, the part size is large to include many keypoints. In contrast, if P is large number, the part size is relatively small and include small number of keypoints. We found the $P = 7$ is suitable number to represent human body in our model.

models [32, 38, 39, 31] made significant progress in the last few years. Poselet used part-based template to interpolate a pose, and k-poselets[20] improved performance by using poselets as part templates in a DPM model with the CNN features[18]. [5, 24, 31, 35, 36] show significant improvement compared to previous methods by training keypoint specific part detectors based on the CNN framework for human body pose estimation. However, none of these models do not incorporate any notions of visual attributes.

Grammar Models: The grammar models have become an increasingly popular topic in the literature, and can be categorized into two principal variations: phrase structure grammars(PG) and dependency grammars(DG). In the PG, each node must geometrically contain all of its constituents, and by extension the recursive collection of constituent parts are summarized together into a concise coarse-to-fine abstraction[14, 16, 17]. The PG is well suited to represent compositional structures that do not undergo large deformations. In a DG, constituent parts do not need to be contained within their parents, but instead are constrained by an adjacency relation[22]. The DG is well suited for representing objects that exhibit large articulated deformations. The disadvantage of the DG is that it loses the coarse-to-fine summarization. Stochastic and-or grammar(AOG) provides a general and principled framework for representing compositional structures, and can be formulated to generalize both PG and DG[43]. In contrast to DG and PG, the AOG defines two types of nodes: the and-node and or-node. And-nodes represent a composition of an entity from its constituents, whereas the or-node defines a selection among alternative choices that compete with one another. Our model is an extension of these conventional grammar models to represent human body with attributes in an unified framework and borrow and-or notation for grammar construction. We model both PG and DG structure in the unified grammar.

3. Attributed Grammar Model

In our grammar model, each part is represented by the state variables designating the location and attribute of part. This state representation is common for all parts throughout the hierarchy. Part state is represented by its geometry state (x, y, s) for position and scale. Attribute state is represented by a set of attributes for part. The probability model over the parse tree is formulated as a Gibbs energy model. $P(pt|I) \propto P(I|pt)P(pt) = \frac{1}{Z}\exp\{-E(I|pt) - E(pt)\}$. The likelihood term is used for appearance response, and the prior term is used to describe relations in grammar. Both terms are defined for part and and attribute. Then, we rewrite equation as $P(pt|I) = \frac{1}{Z}\exp\{-E^P_A(I|pt) - E^A_A(I|pt) - E^P_R(pt) - E^A_R(pt)\}$. $E^P_A(I|pt)$ and $E^A_A(I|pt)$ are appearance term for part and attribute respectively. $E^P_R(pt)$ and $E^A_R(pt)$ are relation terms for part and attribute. The energy is expressed as a scoring function. $S(pt, I) = -E^P_A(I|pt) - E^A_A(I|pt) - E^P_R(pt) - E^A_R(pt) = S^P_A(I|pt) + S^A_A(I|pt) + S^P_R(pt) + S^A_R(pt)$. We now describe each component of the scoring function.

3.1. Part Model

Defining Part node Before we design the grammar model, we first need to define parts. And, the part can be defined in several ways: as one part(whole body), three parts(head, upper body, lower body), find-grained parts(eyes, writs, hands, and etc.), and etc. The whole body can be detected easily, and includes large information. But, it requires the huge number of part templates to describe large variation of appearance under different view and pose. In contrast, if the body is constructed with fine-grained parts, each part includes small information and not easy to be detected. But, it more suits for representing objects that exhibit large articulated deformations. Therefore, we find the proper number of parts to describe a large variation of human pose with relatively large size of part for including enough information. To find P parts from keypoints annotations in the dataset[2], we assume the part is defined by a composition of the keypoints, and use 15 keypoints ($N = 15$) as shown in Figure 3 with yellow circles. The key idea for finding P parts is to set P value and find P window regions to cover all observable keypoints. In brief, we find parts by following process: (1) Randomly select the example which has all visible keypoints. (2) Draw P bounding boxes by avoiding overlap with other bounding boxes until each part region includes N/P keypoints approximately. (3) Visit training examples one by one and check whether it can be described with current part design. If we can describe most of examples with current definition, we keep current setting. Otherwise, we go back to step (1) and find another example to find another part design. We repeat this process until we find suitable part definition. We designed each part has same size of window, 196 x 196 in

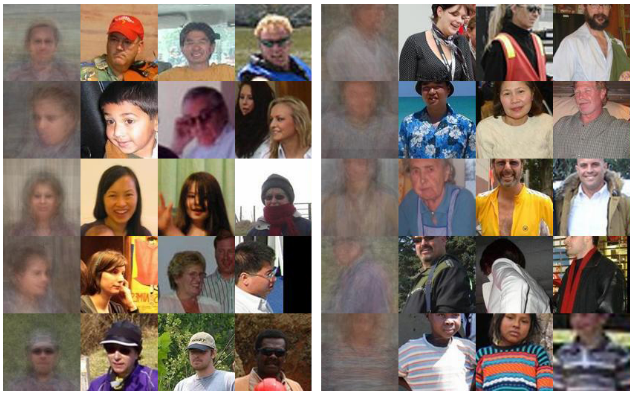


Figure 4. Examples of clustered types for two p_1 and p_2. Each row is a cluster, and left columns show mean images of each cluster.

the dataset[2]. In this process, we found the part is relatively small and have large variation of pose when each part includes 2 or 3 keypoints. So, we define 7 parts ($P = 7$) and show in Figure 3.

Part Relation Model We model the relations between parts now. The root node is defined by finding mostly shared part over the parts. Then, we expand tree structure by adding closest parts to the nodes. We show defined tree structure in Figure 1. The part relation is defined by the and-rule. And-rule is for assembly of constituent parts and enforce geometric constraints between two parts, p_i and p_j. It is described by $S^P_R(p_i, p_j) = \langle\lambda^P_{ij}, [dx^2_{ij}\ \ dy^2_{ij}\ \ ds^2_{ij}\ \ ]\rangle$ where dx^2_{ij}, dy^2_{ij} is relative location, ds^2_{ij} is relative scale. λ^P_{ij} is learned from the geometry relation in training. We also define the articulation constrains between part p_i and its type ω_i. It is defined as $S^P_R(p_i, \omega_i)$ where ω_i is one of types of part p_i and computed in same way from $S^P_R(p_i, p_j)$. Each part in the grammar is defined by the or-rule which means model selects one type over another to explain part. S^P_R over the parse tree is computed as

$$S^P_R(pt) = \sum_{p_i \in pt} \left(S^P_R(p_i, \omega_i) + \sum_{p_j \in \mathrm{C}(p_i)} S^P_R(p_i, p_j) \right)$$

where $C(p_i)$ is the set of child part of p_i.

Part Appearance Model To have part templates, we first crop the region of part p_i by drawing the bounding box that uses the mean point of K_{p_i} as its center point. K_{p_i} is the set of corresponding keypoints for p_i. In the dataset, however, many images are truncated and many keypoints are invisible because of pose and occlusion, and also scale has large variation. Therefore, cropped part images are noisy and not aligned. To have clear part images, we first do clustering the cropped images for each body part using k-means clustering method. We then train detector for each types of parts by logistic regression, and do detection around initial part bounding box. We apply it onto the entire training set and iteratively and update location and scales to find best part bounding box. We define 10 appearance types for each part and show examples of first five types for p_1 and p_2 in Figure 4. We denote appearance score function of part p and type ω as $S^P_A(I|p)$ and $S^P_A(I|\omega)$ respectively. To describe appearance template, we use four appearance features: CNN-based features, gradient based feature, and two color features. For CNN-based feature extraction, our model generalizes R-CNN method[18] by applying it to part and its type with pre-trained ImageNet weights of CNN which is publicly available. We then extract fully connected layer features for fc$_7$ for part region. For gradient based feature, we used Histogram of gradient(HOG) method. The first color feature is color histogram and second color feature is RGB pixel value. Both color features are in RGB color space. The appearance score for part is now inner product of long vector of all appearance feature vector and feature weight of part p and type ω. It is written as $S^P_A(I|p) = \langle\lambda^{P_a}_p, \phi^{P_a}(I,p)\rangle$, $S^P_A(I|\omega) = \langle\lambda^{P_a}_\omega, \phi^{P_a}(I,\omega)\rangle$. $\phi^{P_a}(I,p)$ and $\phi^{P_a}(I,\omega)$ are the template response vector and, $\lambda^{P_a}_p$ and $\lambda^{P_a}_\omega$ are the trained appearance feature weight of part p and type ω respectively. Then, S^P_A over pt is computed as

$$S^P_A(I|pt) = \sum_{p_i \in pt} S^P_A(I|p_i) + S^P_A(I|\omega_i)$$

Table 1. Attribute prediction accuracy given part. We map the part and attribute if the accuracy is higher than 0.5. For example, Jeans is related with p_5, p_6, and p_7. Figure 3 shows part index.

Part	p_1	p_2	p_3	p_4	p_5	p_6	p_7
Gender	**0.87**	**0.79**	0.43	0.41	0.34	0.27	0.28
Long Hair	**0.82**	0.34	0.24	0.25	0.12	0.10	0.14
Glasses	**0.74**	0.24	0.11	0.12	0.11	0.08	0.09
Hat	**0.83**	0.18	0.14	0.13	0.07	0.08	0.12
T-shirt	0.42	**0.82**	**0.57**	**0.54**	0.28	0.19	0.24
Long Sleeve	0.23	0.47	**0.72**	**0.74**	0.22	0.24	0.27
Shorts	0.31	0.21	0.24	0.22	0.49	**0.78**	**0.77**
Jeans	0.17	0.18	0.21	0.19	**0.69**	**0.90**	**0.89**
Long Pants	0.21	0.19	0.20	0.21	0.38	**0.91**	**0.92**

3.2. Attribute Model

We now combine attribute in the grammar model by defining relations between part and attributes. Previous attribute approaches [2, 4, 25, 40, 41] use all defined parts (or poselets) for attribute classifications, however, we can simply know some parts may not be related with such attributes, and it could hurt attribute prediction if we make attribute relation with unrelated part. For examples, glasses is not related with lower body parts and t-shirt is not related with head and lower body parts. In contrast, long-hair attribute will be highly related with head part. Therefore, we need to learn how attributes and parts are related. We define the set of attributes for each part p and denote by $A(p)$. $A(p)$ includes related attributes for part p. To obtain the $A(p)$ for each part p, (1) we map the attributes with all parts in the structure by assuming each part is related with all attributes. (2) Train binary attribute classifier with logistic regression for each part type and compute prediction accuracy by cross validation. (3) Discard relations between attribute and part, if the average prediction score is lower than threshold ($= 0.5$). We show attribute prediction accuracy in Table 1. In this step, gender is mapped with p_1 and p_2. Long Pants is related with p_6 and p_7. We show part index in Figure 3. $A(p)$ can be treated as a two layered simple tree. The root node $A(p)$ is described by and-rule, it includes corresponding attributes as its child nodes. Then, each attribute follows or-rule. It could be written into production rules as below.

attribute production rule	example
$A(p_2) \to \{A_1, A_2\}$	$A(p_2) \to$ {Gender, T-shirt}
$A_1 \to A_{11} \vert A_{12}$	Gender $\to$ Female|Male
$A_2 \to A_{21} \vert A_{22}$	T-shirt $\to$ Yes|No

Attribute Relation Model Each attribute node $A(p_i)$ is linked with a part p_i through a relation as shown by the blue edges in Figure 2 (d). This relation reflects the co-occurrence frequency of the attribute given the part type. For example, let the specific part type ω of node v, upper body, has an appearance that is blouse-like. This will occur more frequently with female than male, and therefore the model should favor selecting this part type when there is strong evidence for the female attribute. This score is computed as $S_R^A(\omega, A(p)) = \langle \lambda_\omega^{att}, \phi^{att}(A(p)) \rangle$ where ϕ^{att} is a vector to indicate selected attribute types on the $A(p)$. λ_ω^{att} is a learned compatibility weight vector for between part types and attributes. The S_A over pt is computed as

$$S_R^A(pt) = \sum\nolimits_{p_i \in pt} S_R^A(\omega_i, A(p_i))$$

Attribute Appearance Model We define attribute appearance template for each part type and show examples by mean of example in Figure 5. In this figure, we show examples for first 5 types for p_1 with four corresponding attributes from Table 1. We use same features with part template. Appearance score for attribute is written as $S_A^A(I|a) = \langle \lambda_a^{A_a}, \phi^{A_a}(I, a) \rangle$. a is selected attribute type. $\phi^{A_a}(I, a)$ is the template response vector and $\lambda_a^{A_a}$ is the trained appearance feature weight of attribute type a. The S_A^A over pt is computed as

$$S_A^A(I|pt) = \sum_{p_i \in pt} S_A^A(I|A(p_i)) = \sum_{p_i \in pt} \sum_{a \in A(p_i)} S_A^A(I|a)$$

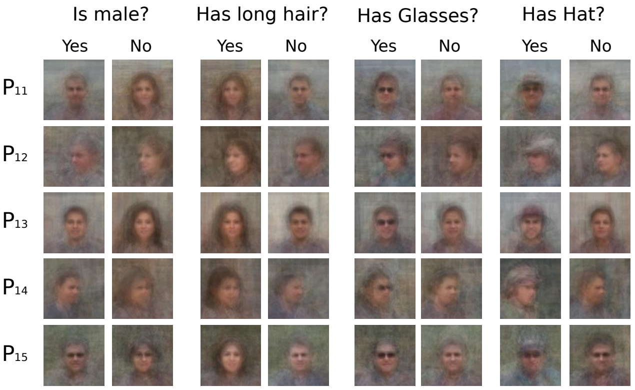


Figure 5. Appearance template of attribute of part 1 (p_1). We show first five types of part 1 with four attributes using mean images of examples.

4. Parsing and Inference

The inference task is equivalent to the finding most probable parse tree pt for given image, and calculated by maximizing the score functions described in previous sections.

$$\begin{aligned} pt^* &= \arg\max_{pt} P(I|pt)P(pt) \\ &= \arg\max_{pt} [S_A^P(I|pt) + S_A^A(I|pt) + S_R^P(pt) + S_R^A(pt)] \end{aligned}$$

The pt also can be formulated recursively given node p_i, and the score function is written as,

$$\begin{aligned} S(p_i, I) = \quad & S_A^P(I|p_i) + S_A^P(I|\omega_i) + S_A^A(I|p_i) \\ & + S_R^P(p_i, \omega_i) + S_R^A(\omega_i, A(p_i)) \\ & + \sum\nolimits_{p_j \in C(p_i)} [S_R^P(p_i, p_j) + S(p_j, I)] \end{aligned} \quad (1)$$

For an image I, parsing is now defined as finding the parse tree that maximizes the score function $pt^* = \arg\max_{pt} S(p_0, I)$ where p_o is the root part. This maximization problem can be computed reasonably efficiently using the dynamic programming. We first compute two score maps, appearance map and relation map, for each part, part type and attributes. As for the appearance score map, we compute the appearance responses for each part, part types, and attributes. As for the relation map, we compute the maximization of their parent compositions, and store the optimal composition scores for all possible geometries and attributes. The child parts are conditionally independent given the parent part, and can be maximized individually. Although the part p_i is fixed, we must maximize over the full state of the p_j, including its attributes. The maximization over positions (x_j, y_j) can be computed very efficiently using distance transforms [12] that have linear complexity in the number of positions. The maximization over scales requires quadratic time to compute. But, the state space is still quite small and the computation is tractable. In equation (1), ω_i is the selected part type for part p_i that maximize the score. However, selecting a part type could be a risky with several reasons. It affects on the result significantly in a negative way when the model selects incorrect type. To minimize the risk we include all part types for each part rather than selecting only one part type. Then, $S(p_i, I)$ can be rewritten as

$$\begin{aligned} S(p_i, I) = \; & S_A^P(I|p_i) + S_A^A(I|p_i) \\ & + \sum_{\omega \in p_i} [S_A^P(I|\omega) + S_R^P(p_i, \omega) + S_R^A(\omega, A(p_i))] \\ & + \textstyle\sum_{p_j \in C(p_i)} [S_R^P(p_i, p_j) + S(p_j, I)] \qquad (2) \end{aligned}$$

The inference algorithm starts from the leaves of the grammar, and works toward the root part by computing relation score maps of the optimal scores. The state in the score map of the root part with maximal score will be the globally optimal solution and can recover the parse tree by using the backtracking method. During the parsing, the model selects the attribute types that maximizes score for each part, and attributes could be different over the hierarchy as the parsing uses dynamic programming method. To have attribute prediction, we can select the attribute which maximize the attribute score over the parse tree. The final attribute prediction is decide by aggregating the all scores over the parse tree.

5. Learning

We aim at finding a prediction function $f : I \rightarrow pt$ during the learning procedure. The parse tree score can be expressed as an inner product of model parameters λ and a corresponding vector of feature responses $\phi(pt, I)$ for the parse tree pt and input image I: $s(pt, I) = \langle \lambda, \phi(pt, I) \rangle$. Unlike we use equation 2 for $s(pt, I)$ for parsing, we use equation 1 during the learning process. The learning task now can be defined as finding model parameters λ that minimizes the empirical risk $R_{emp}(\lambda)$ which is defined as the expected loss over the training examples. The optimal parameters are $\lambda^* = \arg\min_\lambda R_{emp}(\lambda) = E_{(\hat{pt}, I) \sim D}[L(pt, \hat{pt})]$ where $L(pt, \hat{pt})$ is the loss function. We define two loss functions: part localization loss and part type selection loss. As for the localization loss, we use the conventional Intersection Over Union (IOU) metric, which is bounded between 0 and 1. $L_l(pt, \hat{pt}) = \frac{1}{n} \sum_{i=1}^{n} \text{IOU}(t_i, \hat{t}_i)$ where n is the number of parts in the grammar. We use the zero-one loss for the part type selection. $L_c(pt, \hat{pt}) = \frac{1}{n} \sum_{i=1}^{n} I \cdot (t_i \neq \hat{t}_i)$ where I is the indicator notation. We now sum up two loss scores: $L(pt, \hat{pt}_i) = \frac{1}{2}(L_l(pt, \hat{pt}_i) + L_c(pt, \hat{pt}_i))$. We then use the so-called margin-rescaled structural hinge loss [34]. It is written in the following max-margin structural SVM objective function which optimizes a quadratic objective function of the parameters ω given a set of training examples $|D|$.

$$\min_\lambda \tfrac{1}{2}||\omega||^2 + \tfrac{C}{|D|} \textstyle\sum_{i=1}^{|D|} \xi_i$$
$$s.t. \lambda^\top [\phi(\hat{pt}_i, I_i) - \phi(pt, I_i)] \geq L(pt, \hat{pt}_i) - \xi_i, \forall pt \in \Omega_{\mathcal{G}}, \forall i$$

Since the total number of constraints grows exponentially, they cannot be enumerated exhaustively making it intractable to minimize this expression directly. To solve this efficiently, we find and add the most violated constraints at each iteration with the following maximization as the so-called separation oracle,

$$\hat{pt}_i = \arg\max_{pt} \lambda^\top \phi(pt, I_i) + L(pt, \hat{pt}_i)$$

This objective function can be minimized by a multiple will-tuned solvers. The cutting plane solver [37], the stochastic gradient descent solver [11], the conventional QP solver, and the dual coordinate descent solver[38] is also commonly used. For our implementation, we used the dual coordinate descent solver.

6. Experiments

We tested our model on attribute prediction task as defined by [2] on two dataset: Attributes of People Dataset[2] and our new dataset, Pedestrian attribute dataset. We also evaluate on pose-estimation task on PARSE dataset[38].

Attributes of People dataset. In this dataset, each image is centered at each person and manually annotated a visible bounding box of each person. This dataset defines 9 binary attributes, as well as keypoint annotations that can be used for training. The goal is predicting attributes given ground truth bounding box of target person.

Pedestrian attribute dataset. For testing on various datasets for better evaluation, we design new dataset. There are many pedestrian and attribute datasets, however, none provides part and attribute annotations together. In addition, most of the images in other datasets include multiple pedestrians in a single image. Therefore, the evaluation provides the bounding box of each target person, making the problem easier by not requiring the method to detect the human. Our new dataset includes 2257 high resolution images(1257 for training, 1000 for testing), each including a single pedestrian. It consists of many kinds of variation in margin, size, attribute, pose, appearance, geometry and environment. We annotated 10 body parts and 9 attribute classes as listed in table 2. Most of attributes are not binary, and it makes task much harder. For testing on this dataset, we skip the part designing process as dataset includes part bounding box annotation.

Table 2. Attribute list in human attribute dataset

Attribute Class	Attribute Type
gender	male, female
age	youth, adult, elderly
hair-style	long-hair, short-hair, bald
upper cloth type	t-shirt, jumper, suit, no-cloth, swimwear
upper cloth length	long-sleeve, short-sleeve, no-sleeve
lower cloth type	long-pants, short-pants, skirt, jeans
accessory	backpack, glasses, hat

6.1. Experiment variations

During the attribute prediction test, we designed four different experiment cases to illustrate the impact of our model design on performance.

Mapped parts: This is for using mapped part for each attribute as we described in previous sections.

All parts: This setting is for using all parts for each attribute. Most of previous approaches [2, 4, 25, 40, 41] use this setting to extract information from all defined parts or poselet.

Classifier: This is for using classifier for each attribute. By following the method in [2, 25], we train part type detector and attribute classifier using linear SVM with same features that used for our model. We train body model using our model without attributes. It works as pose-estimator. In testing, we first detect parts, and then detect its types around detected part. The final prediction is made by summing up the score from attribute classifiers with the weights given by part type detection scores.

Joint model: This approach is our proposed model structure. We do inference and prediction with part and attribute jointly with our unified model.

To evaluate the pose estimation, we selected the popular PARSE dataset as we can minimize the additional attribute annotation for evaluation because training set includes only 100 images. For joint training, we annotate one attribute, gender. The evaluation protocol for pose estimation uses the standard PCP criteria of [8].

6.2. Results

Table 3 shows comparison of our proposed model with existing approaches [2, 25, 40, 41] on Attributes of People Dataset. Our model leads to state of arts[41] on 5 of 9 attributes and also on mean average precision(mAP). Note that [41] model is trained on an different training dataset with 25k images which is much larger than original set. But, we were not able to train on this dataset because it is not publicly available. If we compare our model to method in [25] which is using original training images from dataset, we outperform on all attributes, and show 10% higher mean average precision. We show the result on our new dataset on Table 4. In the experiment, we use average accuracy for attribute evaluation as it is a multi-class classification problem. We observe similar result from experiment on Attributes of People dataset. From both of results, we can see mapping parts with attribute helps performance improvements on most of attribute predictions, especially attribute that related with small part of body. In addition, our unified joint model assists to have better performance on overall on both dataset. We show output examples in Figure 6 and Figure 7. In Figure 6, we show three most positive and two most negative examples for each attribute. We also predict part as shown in Figure 7. We did not show part localization result in Figure 6 for better visualization. We show the pose-estimation performance on Table 5. Although we have slightly lower performance for four parts than [36], we are showing competitive performance and get best performance for head compared to existing methods. In addition, this output is obtained from our joint model without designing addition pose-estimator. It also can generate attribute prediction output with pose-estimation simultaneously. We evaluate our model with and without attributes. Our model works as pure pose-estimator when we do not combine attribute. It shows including attributes helps the model to obtain better performance on all parts. It means our joint modeling has high potential impact to improve many of other tasks.

7. Conclusion

We have presented a attributed grammar model that can represent fine-grained part, pose, and attribute reasoning. The advantage of our representation is the ability to perform simultaneous attribute reasoning and part detection, unlike previous attribute models that use and rely on large numbers of region-based attribute classifiers without explicitly localizing parts. We also show that mapping part with attribute helps to have better representation of attribute and better performance than exiting methods that make connection of attribute with all defined parts. We demonstrate our model with benchmarks, and achieve better attribute classification performance against recent methods. We also show our joint modeling for part and attribute model helps the pose-estimation task. We believe that evidence of the attributes as well as their consistency constraints can help lead to performance improvements on both tasks.

8. Acknowledgement

This work was gratefully supported by DARPA Award N66001-15-C-4035, MURI grant N000141010933, and NSF IIS 1423305.

Table 3. The attribute classification performance (average precision) on the **Poselet Attributes of People Dataset**[2].

Method	Male	Long hair	Glasses	Hat	T-shirt	Long sleeve	Shorts	Jeans	Long pants	mAP
Poselet [2]	82.4	72.5	55.6	60.1	51.2	74.2	45.5	54.7	90.3	65.18
Joo *et al*.[25]	88.0	80.1	56.0	75.4	53.5	75.2	47.6	69.3	91.1	70.7
DPD [40]	83.7	70.0	38.1	73.4	49.8	78.1	64.1	78.1	93.5	69.88
PANDA* [41]	91.7	82.7	**70.0**	74.2	68.8	**86.0**	**79.1**	81.0	**96.4**	78.98
Ours (All parts + Classifier)	90.3	80.0	61.7	73.0	63.2	76.6	62.9	79.4	90.7	75.41
Ours (Mapped parts + Classifier)	89.8	82.2	62.4	74.3	64.4	78.4	64.2	80.4	91.7	76.42
Ours (All parts + Joint model)	91.8	82.2	65.6	75.2	68.1	82.8	63.4	81.1	92.7	77.20
Ours (Mapped parts + Joint model)	**92.1**	**85.2**	69.4	**76.2**	**69.1**	84.4	68.2	**82.4**	94.9	**80.20**

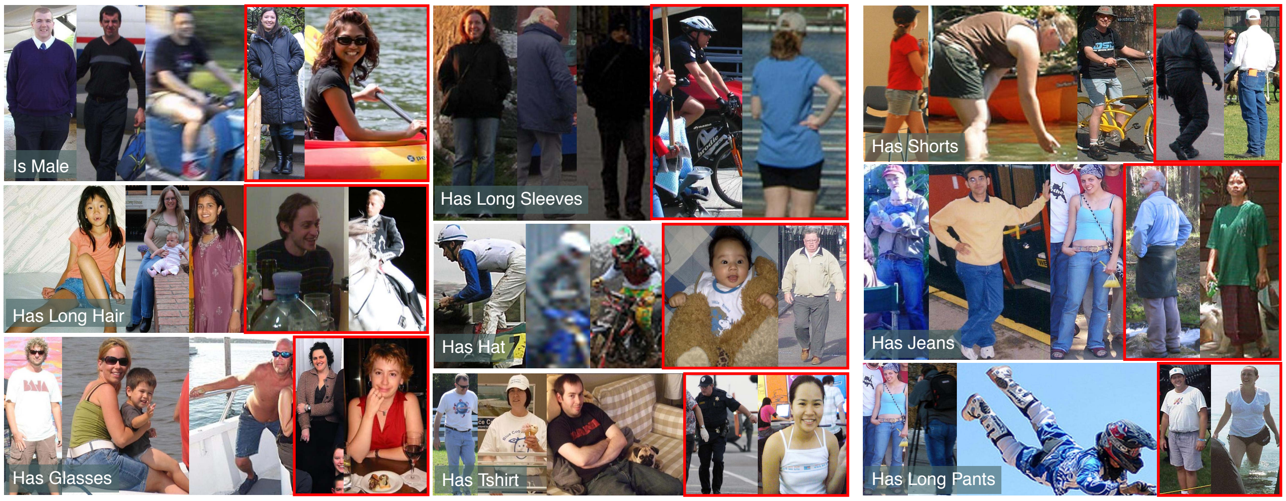


Figure 6. Attribute prediction on Attributes of People Dataset[2]. First three examples for most positive examples, and last two examples show most negative examples for each attribute. We cropped the image around the ground truth bounding box for display purpose.

Table 4. Results for attribute classification on the proposed **Human Attribute Dataset**. We use average accuracy for evaluation.

Method	Gender	Age	Hair-style	Upper cloth type	Upper cloth length	Lower cloth type	Backpack	Glasses	Hat
Ours (All parts + Classifier)	76.11	84.31	65.78	72.11	73.14	67.58	62.65	56.78	69.20
Ours (Mapped parts + Classifier)	76.54	84.08	66.99	74.32	75.89	68.80	69.55	58.11	75.22
Ours (All parts + Joint model)	78.92	86.41	70.71	74.27	73.11	67.89	62.78	57.89	74.11
Ours (Mapped parts + Joint model)	**79.77**	**88.17**	**71.71**	**74.97**	**77.19**	**69.89**	**70.78**	**61.11**	**78.11**

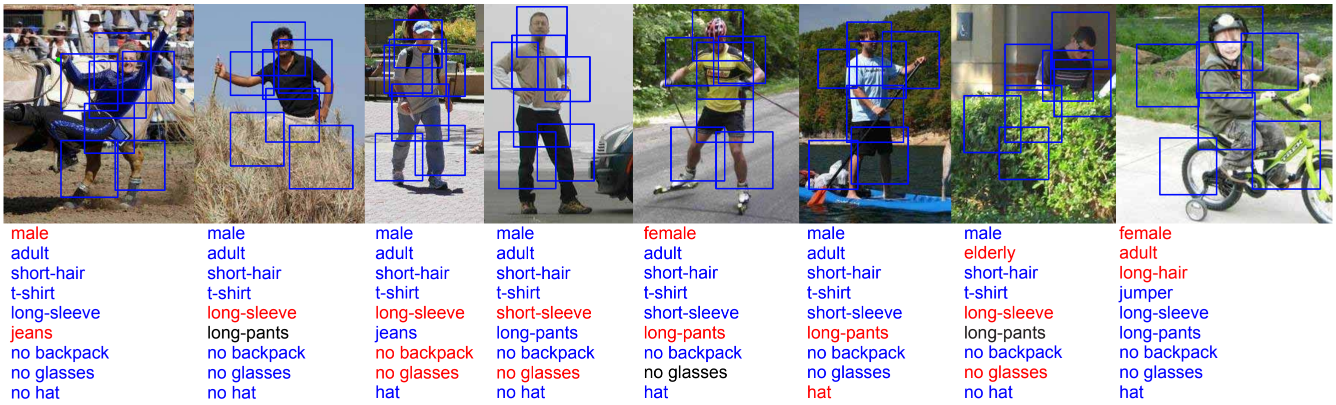


Figure 7. Output examples on **Human Attribute dataset**. Attribute predicted correctly are shown in blue, and incorrectly in red. Unknown ground truths are in Black. We also show detected parts with blue rectangles.

Table 5. Results for Pose estimation on the **PARSE dataset** [38]. We define gender for this model for training joint model. Therefore, this output is generated with attribute (gender) prediction together.

Method	Torso	Upper Leg	Lower Leg	Upper Arm	Lower Arm	Head	Avg
YR [39]	85.9	74.9	68.3	63.4	42.7	86.8	67.1
Johnson et al., [24]	87.6	74.7	67.1	67.3	45.8	76.8	67.4
Rothrock *et al*. [32]	87.5	77.4	69.6	65.1	46.5	87.1	69.1
Pishchulin et al., [31]	**93.2**	76.4	68.0	63.4	48.8	86.3	**69.4**
DeepPose [36]	-	**80.0**	**75.0**	**71.0**	**50.0**	-	-
Ours (No Attr)	86.8	76.5	68.0	64.9	44.8	86.8	68.2
Ours (Mapped parts + Joint model)	88.4	77.0	68.9	66.3	46.5	**88.4**	**69.4**

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, 2009. 2

[2] L. D. Bourdev, S. Maji, and J. Malik. Describing people: A poselet-based approach to attribute classification. In *ICCV*, 2011. 1, 2, 3, 4, 6, 7, 8

[3] L. D. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, 2009. 1, 2

[4] H. Chen, A. Gallagher, and B. Girod. Describing clothing by semantic attributes. In *ECCV*, 2012. 2, 4, 7

[5] X. Chen and A. Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *NIPS*, 2014. 3

[6] G. W. Cottrell and J. Metcalfe. Empath: Face, emotion, and gender recognition using holons. In *NIPS*, 1990. 2

[7] J. Dai, Y. Hong, W. Hu, S.-C. Zhu, and Y. N. Wu. Unsupervised learning of dictionaries of hierarchical compositional models. In *CVPR*, 2014. 1

[8] M. Eichner and V. Ferrari. Better appearance models for pictorial structures. In *BMVC*, 2009. 2, 7

[9] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *CVPR*, 2009. 1

[10] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. Cascade object detection with deformable part models. In *CVPR*, 2010. 2

[11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. In *PAMI*, volume 32, pages 1627–1645. IEEE, 2010. 6

[12] P. F. Felzenszwalb and D. P. Huttenlocher. Distance transforms of sampled functions. 2004. 5

[13] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. In *IJCV*, 2005. 2

[14] S. Fidler, G. Berginc, and A. Leonardis. Hierarchical statistical learning of generic parts of object structure. In *CVPR*, 2006. 1, 3

[15] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. In *IEEE Trans. on Computer*, 1973. 2

[16] H. Gaifman. Dependency systems and phrase-structure systems. In *Information and Control*, 1965. 3

[17] G. Gazdar, E. Klein, G. Pullum, and I. Sag. Generalized phrase structure grammar. In *Basil Blackwell*, 1985. 3

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 3, 4

[19] R. B. Girshick, P. F. Felzenszwalb, and D. McAllester. Object detection with grammar models. In *NIPS*, 2011. 1, 2

[20] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik. Using k-poselets for detecting people and localizing their keypoints. In *CVPR*, 2014. 3

[21] B. A. Golomb, D. Lawrence, and T. Sejnowski. Sexnet: A neural network identifies sex from human faces. In *NIPS*, 1991. 2

[22] D. G. Hays. Dependency theory: A formalism and some observations. In *Language*, 1964. 3

[23] Y. Jin and S. Geman. Context and hierarchy in a probabilistic image model. In *CVPR*, 2006. 1

[24] S. Johnson and M. Everingham. Learning effective human pose estimation from inaccurate annotation. In *CVPR*, 2011. 3, 8

[25] J. Joo, S. Wang, and S. C. Zhu. Human attribute recognition by rich appearance dictionary. In *ICCV*, 2013. 1, 2, 4, 7, 8

[26] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Describable visual attributes for face verification and image search. In *PAMI*, volume 33, 2011. 2

[27] Y. H. Kwon and N. da Vitoria Lobo. Age classification from facial images. In *CVIU*, 1999. 2

[28] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009. 1

[29] B. Moghaddam and M.-H. Yang. Learning gender with support faces. In *PAMI*, 2002. 2

[30] M. Pei, Y. Jia, and S.-C. Zhu. Parsing video events with goal inference and intent prediction. In *ICCV*, 2011. 1

[31] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele. Strong appearance and expressive spatial models for human pose estimation. In *ICCV*, 2013. 3, 8

[32] B. Rothrock, S. Park, and S. C. Zhu. Integrating grammar and segmentation for human pose estimation. In *CVPR*, 2013. 1, 3, 8

[33] B. Sapp, C. Jordan, and B. Taskar. Adaptive pose priors for pictorial structures. In *CVPR*, 2010. 2

[34] B. Taskar, C. Guestrin, and D. Koller. Max-margin markov networks. In *NIPS*, 2003. 6

[35] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *NIPS*. 2014. 3

[36] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, 2014. 3, 7, 8

[37] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *ICML*, 2004. 6

[38] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011. 1, 2, 3, 6, 8

[39] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. In *PAMI*, volume 35, 2013. 3, 8

[40] N. Zhang, R. Farrell, F. Iandola, and T. Darrell. Deformable part descriptors for fine-grained recognition and attribute prediction. In *ICCV*, 2013. 4, 7, 8

[41] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. Panda: Pose aligned networks for deep attribute modeling. In *CVPR*, 2014. 1, 2, 4, 7, 8

[42] Y. Zhao and S. C. Zhu. Image parsing via stochastic scene grammar. In *NIPS*, 2011. 1

[43] S. C. Zhu and D. Mumford. A stochastic grammar of images. In *Foundations and Trends in Computer Graphics and Vision*, volume 2, 2006. 1, 3