Title
Pose-Guided Human Semantic Part Segmentation

Author
Xia, Fangting

Publication Date
2016

Peer reviewed|Thesis/dissertation

University of California
Los Angeles

Pose-Guided Human Semantic Part Segmentation

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Statistics

by

Fangting Xia

2016

Abstract of the Dissertation

Pose-Guided Human Semantic Part Segmentation

by

Fangting Xia

Doctor of Philosophy in Statistics

University of California, Los Angeles, 2016

Professor Alan Loddon Yuille, Chair

Human semantic part segmentation and human pose estimation are two fundamental and complementary tasks in computer vision. The localization of joints in pose estimation can be much more accurate with the support of part segment consistency while the local confusions in part segmentation can be greatly reduced with the support of top-down pose information. In natural scenes which consist of multiple people, human pose estimation and human part segmentation are still challenging due to multi-instance confusion and large variations in pose, scale, appearance and occlusion. Current state-of-the-art methods for both tasks rely on deep neural networks to extract data-dependent features, and combine them with a carefully designed graphical model. However, these methods have no efficient mechanism to handle multi-person overlapping or to adapt to the scale of human instances, thus are still limited when facing large variability in human pose and scale.

To improve the performance of both tasks over current methods, we propose three models that tackle the difficulty of pose/scale variation in two major directions: (1) introduce top-down pose consistency into semantic part segmentation and introduce part segment consistency into human pose estimation, letting the two tasks benefit each other; (2) handle the scale variation by designing a mechanism to adapt to the size of human instances and their corresponding parts. Our first model incorporates pose cues into a graphical model-based part segmentation framework while our third model combines pose information within a framework made up of fully convolutional networks (FCN). Our second model is a hierar-

chical FCN framework that performs object/part scale estimation and part segmentation jointly, adapting to the size of objects and parts. We show that all our three models achieve state-of-the-art performance on challenging datasets.

The dissertation of Fangting Xia is approved.

Luminita Aura Vese

Hongjing Lu

Yingnian Wu

Alan Loddon Yuille, Committee Chair

University of California, Los Angeles

2016

Table of Contents

List of Figures

List of Tables

Acknowledgments

I am sincerely thanksful to many people for their help and support along the way.

First and foremost, to my advisor, Alan L. Yuille, for leading me into the door of research on Computer Vision, guiding me and encouraging me for the past five years. I remember joining his lab as an exchange undergraduate student when I knew very little about Computer Vision. He kindly explained many concepts to me and showed me how to perform inference on several typical graphical models. I was very thankful for that. Later when I formally joined his lab as a Ph.D, my first project didn't go smoothly. Alan didn't put pressure on me and always discussed with me patiently. He often introduced me to his collaborators when he thinks there's a common research topic we were interested in, from which I learned how to cooperate with others in research. I remember that Alan stayed up late with us many times when there's a deadline for paper submission, helping us revising the paper. I am deeply respectful for his devotion to research and to the lab, and I am very grateful for his guidance and encouragement.

To my thesis defense committee, Yingnian Wu, Hongjing Lu, and Luminita A. Vese, for their feedback and advice of this work, and also for their valuable time taken to participate in my defense.

To my collaborators during the past five years: Jun Zhu, Peng Wang, and Liang-Chieh Chen. I always remember the first time I tried to submit my paper to a top conference, when Jun Zhu didn't sleep for one whole night to help me writing/revising the paper. When I was stuck in my projects, Peng Wang was always willing to help me, sharing with me his opinions and collaborating with me on three projects. Liang-Chieh is very nice to me. He participated in one project of mine, helping me running some baseline methods and giving me valuable feedbacks. When I asked him questions regarding his research area or his code, he always replied me quickly and carefully. I am also thankful to all my other lab mates of Center for Cognition, Vision, and Learning (CCVL): Xianjie Chen, Xiaochen Lian, Chunyu Wang, Jianyu Wang, Xiaobai Liu, Nam-gyu Cho, Zhou Ren, Junhua Mao, Yukun Zhu, Xiaodi Hou, Xingyao Ye, Weichao Qiu, Xin Bo, etc., for their help in research and friendship in life.

To my undergraduate mentors, Enhong Chen, Xiaoping Chen, and Haixun Wang. Thanks for guiding me and encouraging me when I worked as a research assistant in your labs/groups. It's at your labs/groups that I developed strong interest in AI, Computer Science, and Machine Learning.

Last but not least, to my dear family members and good friends. Thanks to my parents and my grandmother for their unconditional love. I always feel safe and encouraged after talking with them. Thanks to Li Du for being a considerate boyfriend, who always supports and encourages me. Thanks to Zhen Li, Hao Wei, Diandian Yu, and Yue Li, who have developed long-time friendship with me. They are always there when I need them.

Vita

2007-2011	Undergraduate in the Department of Computer Science, University of Science & Technology of China (USTC). Research assistant in the home-robot group of Multi-Agent Systems Lab, USTC.
2010-2011	Summer research student in Prof. Alan Yuille's computer vision group, focusing on object detection with graphical models. Research intern in the Web Search & Mining team, Microsoft Research Asia, working on large-scale text classification and understanding.
2011	B.Eng. (Computer Science), USTC, China.
2011-2016	Ph.D Candidate in the Department of Statistics, UCLA. Research Assistant in Center For Cognition, Vision, and Learning, UCLA, focusing on human semantic part segmentation, articulated pose estimation, and deep learning.

Publications

Pose-Guided Human Parsing using AND/OR Graph with Deep-Learned Features. Fangting Xia; Jun Zhu; Peng Wang; and Alan L. Yuille. In AAAI (2016), Oral Presentation.

Zoom Better to See Clearer: Human and Object Parsing with Hierarchical Auto-Zoom Net. Fangting Xia; Peng Wang; Liang-Chieh Chen; and Alan L. Yuille. In ECCV (2016).

CHAPTER 1

Introduction

Human semantic part segmentation [RC12, LSD15] (i.e. decomposing a human body into semantic regions) and human pose estimation [YR11, TGJ15, CY15] (i.e. predicting the position of pose joints) are two crucial and correlated tasks in computer vision. They provide rich descriptions for human-centric image analysis, which is increasingly important for many high-level applications, such as image/video retrieval [DA13, JS13, YKO15], person identification [KZ12], video surveillance [YY11, LBB14], and action recognition [WTL12, WWY13].

Traditional pose estimation approaches [YR11] adopt graphical models, such as And-Or graph(AOG) and conditional random field (CRF), to combine spatial constraints with local observations of joints, based on low-level features like color intensities, HOG [DT05], shape-context [BMP00], and so on. With powerful deep learning architectures and the availability of large-scale annotated data, recent approaches [CY14] rely on deep-learned joint detectors, and use a carefully designed graphical model to select and assemble joints into valid pose configurations. Traditional approaches suffer from limited feature representation power, and can only work in simple datasets with small pose and scale variation. Recent deep-based approaches have much better invariance to pose/scale variation, but their localization of joints is still inaccurate (e.g. joints are sometimes outside the human body) and they still struggle in multi-person overlapping scenes.

For human semantic part segmentation, previous approaches mainly fall into two categories. One category of methods generate part segment proposals ans use graphical models to select and assemble part segment proposals [DCX13, BF11]. These methods are often time-consuming, and have limited power to handle the variability of pose and occlusion in natural images. The other category of methods use fully convolutional nerual networks

(FCN) [LSD15] to directly compute pixel-wise part labels for an image in a simple and fast way [LSX15, XWC16]. These FCN-type methods, however, still make local confusion errors when the person is in an extreme pose, or when there are some other people/objects nearby with similar appearance.

In this work, we aim at improving the performance of human pose estimation and semantic part segmentation in natural images. Each task has some difficult cases if solved individually, e.g. mistakenly missing the knee joints in pose estimation when the person is wearing a long dress, failure to distinguish ambiguous regions in semantic part segmentation when the person is in a non-typical pose, etc. These difficulties, however, can be handled effectively if we consider the correlation of the two tasks and take the advantages of both tasks.

1.1 Problem Statement and Difficulties

Human pose estimation. Given an image as the input, a pose estimation approach needs to output a list of pose configurations in the format of joint locations, for people shown in the image. In this work, we consider 14 types of human joints which are commonly used in previous literature: forehead, neck, left/right shoulder, left/right elbow, left/right wrist, left/right waist, left/right knee, and left/right ankle. Fig. 1.1 demonstrates the input and the output for the human parsing task.

Human semantic part segmentation. Given an image as the input, a semantic part segmentation approach needs to output pixel-wise part label maps, classifying each pixel into one of the semantic part types. In this work, we consider 6 types of semantic parts following previous approaches: head, torso, upper arm, lower arm, upper leg, and lower leg. Fig. 1.2 illustrates the input and the output for the semantic part segmentation task.

Difficulties of both tasks. Human pose estimation and human semantic part segmentation face similar challenges, e.g. large variation in pose, scale, and occlusion, and ap-

Figure 1.1: The human pose estimation task. It takes the original image as the input (Column (a)), and outputs a series of stick figures indicating the location of human pose joints (Column (b)).

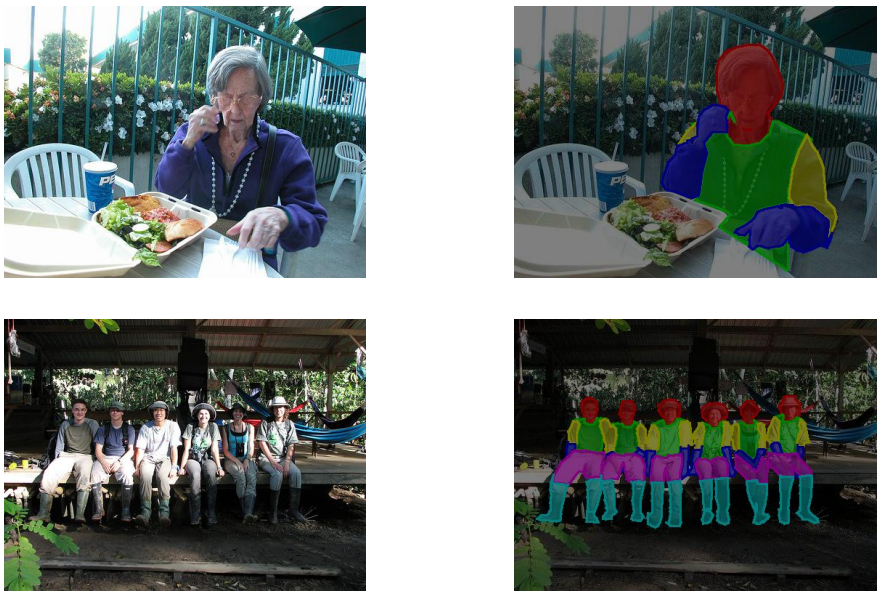


Figure 1.2: The human semantic part segmentation task. It takes the original image as the input (Column (a)), and outputs pixel-wise part label map (Column (b)).

pearance, multi-person overlapping, background clutters, and so on. As shown in Fig. 1.3, current state-of-the-art methods still perform unsatisfactorily on natural images with large pose/scale variation or multi-person overlapping.

1.2 Overview of Dissertation

This dissertation presents three models that target at improving human pose estimation and human semantic part segmentation in natural images with large pose/scale variation. The three models explore three aspects that closely relate to the performance of both tasks: (1) how pose estimation and semantic part segmentation can help each other; (2) how to tackle the large scale variation of human instances and human parts; (3) how to combine deep learning with traditional graphical models, utilizing the advantages of both model types.

In Chapter 1, we briefly introduce the two tasks, and give an overview of this dissertation. In Chapter 2 and Chapter 3, we provide background for the two tasks, and compare our

Figure 1.3: Failure cases of current state-of-the-art methods for human pose estimation (top row) and semantic part segmentation (bottom row).

models with previous ones generally. We give formal descriptions of our three models from Chapter 4 to Chapter 6. This dissertation is concluded in Chapter 7. Here, we give a big picture of our three models.

Model 1. Our first model (Chapter 4) incorporates top-down pose information into an AOG-based part segmentation framework, and defines pose-based features that are able to capture meaningful localization relationship between one part segment and the glocal pose joints. This model performs better than end-to-end deep learning strategies in relatively simple dataset with roughly known scale and typical poses.

Model 2. Our second model (Chapter 5) is a hierarchical FCN model that effectively handles big scale variation in complext natural images. It performs object/part scale estimation and part segmentation jointly, addapting to the size of objects and parts. This model can be easily applied to both tasks, demonstraing excellent results in challenging images.

Model 3. Our third model (Chapter 6) is an iterative framework that combines the two tasks. It formulates the pose estimation problem as a fully-connected CRF, considers the multi-person overlapping issue explicitly, and uses part segment consistency features during inference. This model works quite well on complex multi-person datasets, and thus we learn location & shape priors for semantic parts based on the estimated pose configurations, further improving the part segmentation performance within a FCN framework.

1.3 Overview of Contributions

In our three models, we explored and validated the complementary properties of pose estimation and semantic part segmentation by introducing pose consistency features to help semantic part segmentation (Model 1 & 3) and semantic part consistency features to help pose estimation (Model 3).

We tackle two difficulties that challenge part segmentation and pose estimation in multi-person natural images: large scale variation (Model 2) and large pose variation (Model 3), providing efficient and satisfactory solutions on both tasks.

Detailed contributions of each model are introduced in the speficic chapter of that model.

CHAPTER 2

Preliminaries

In order to be self contained, we provide some preliminary information in this chapter to introduce traditional models relating to our own models. We first introduce three popular graphical models that are popular in human pose estimation and human semantic part segmentation: part-based models, And-Or graph (AOG), and conditional random field (CRF). Then we introduce recent advances in applying deep neural networks to both tasks.

2.1 Part-Based Graphical Models

Objects in 2D images, including humans, can be modeled as collections of interrelated atomic parts. Many part-based models, such as constellation models [FPZ03, WWP00] and pictorial structures [FH05, FE73], have been proposed to explicitly model the appearance of individual parts and the non-rigid geometric deformations between parts. For human pose estimation, "parts" refer to human pose joints like neck and shoulders. For human semantic part segmentation, "parts" refer to semantic part regions of humans, such as head, torso, arms, etc.

Constellation models. In constellation models, a central root node is connected with all the part nodes. The geometric relationship between all the parts and the central node is modeled using a Gaussian distribution. Usually a sparse set of location proposals are first determined by an interest point operator, and then their geometric arrangement is infered by the constellation model. Disadvantages of constellation models include: (1) a joint Gaussian distribution can't capture the articulations between multiple points; (2) the

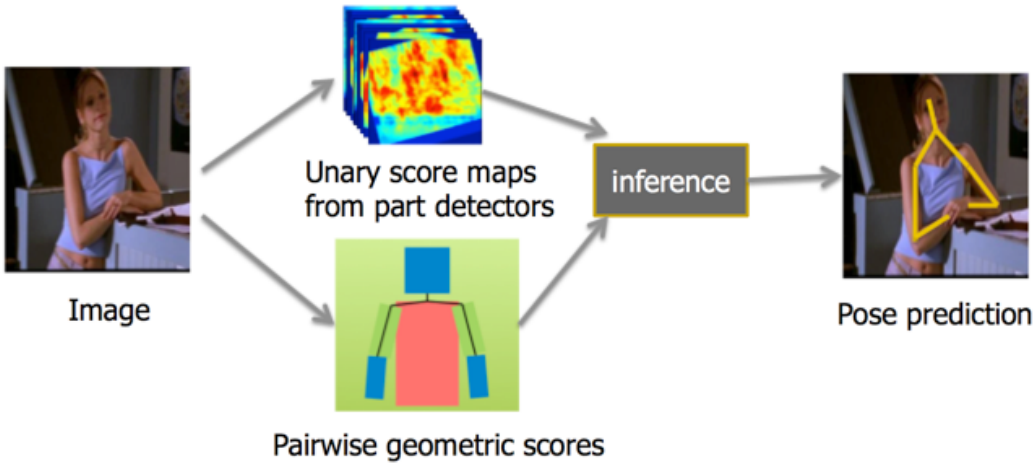


Figure 2.1: Illustration of pictorial structures model on human pose estimation.

inference algorithm normally employs some kind of heuristics, making it difficult to find the optimal configuration of all the parts.

Pictorial structures. Pictorial structures represent objects as a collection of parts, with connections between certain pairs of parts. The model can be described as an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the vertices $\mathcal{V} = \{v_1, v_2, ..., v_n\}$ correspond to the n parts, and each edge (v_i, v_j) corresponds to the connected part pair v_i and v_j. For human pose estimation, $\mathcal{G}$ is usually a tree-structured model resembling the human skeleton, and $\mathcal{E}$ are defined between adjacent joint pairs, such as forehead and neck, neck and left/right shoulder, etc. A human configuration L is defined as $L = \{l_1, l_2, ..., l_n\}$, where l_i denotes the variables we would like to infer for part i, such as part location, orientation, foreshortening, and so on.

The energy function of a particular configuration consists of unary terms that measure how well each part matches the image data, and pairwise terms that measure how well the relative location of each part pair matches the deformation model (see Fig. 2.1). During the inference, we compute the optimal configuration that minimizes Equ. (2.1), where m_i is the unary term for each part i and $d_{i,j}$ is the pairwise deformation term between part i and part j. Generally, the unary term is the match score computed by part filters; the pairwise term is just a function of the relative position between two parts. With the invention of distance

transforms [FH05], pictorial structures can be infered efficiently on images, considering a dense set of part locations.

$$L^* = argmin_L \quad (\sum_{i=1}^{n} m_i(l_i) + \sum_{(v_i,v_j)\in\mathcal{E}} d_{i,j}(l_i, l_j)) \tag{2.1}$$

For part detectors that produce unary scores, pictorial structures use hand-crafted features that have limited representation power and can't handle large variation of pose and appearance. For pairwise term, pictorial structures use relative location only as geometric prior, which is not strong and is not data dependent. Later models have made improvements on pictorial structures by (1) using stronger part detectors and stronger data-dependent pairwise terms; (2) changing the graph structure into cyclic models, mixture of trees, compositional tree models, and so on.

2.2 And-Or Graph (AOG)

And-Or graphs (AOGs) are another type of part-based models, capable of encoding the hierarchical and reconfigurable composition of parts as well as the geometric and compatibility constaints between parts. Some previous methods adopt AOGs for human pose estimation [ZCL08] and human semantic part segmentation[DCX13, XZW16]. Here, we illustrate an AOG for semantic part segmentation (see Fig. 2.2).

We can represent an And-Or graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices, comprising of three node types (i.e. and-nodes, or-nodes, and leaf nodes), and $\mathcal{E}$ is the set of edges, defined according to parent-child relationship. In semantic part segmentation, the leaf nodes ($\mathcal{V}^L$) correspond to human semantic parts at the finest level, e.g. hair and face; the and-nodes ($\mathcal{V}^A$) represent higher-level parts, such as head, which is made up of smaller parts; the or-nodes ($\mathcal{V}^O$) don't have semantic meanings, but are used to specify the topology of the graph, deciding the latent configuration type.

Let z be the state variables of the whole graph. Suppose kids(v) denotes the children of node v, and t denotes switch variables at or-nodes, indicating the set of active nodes $\mathcal{V}(t)$.

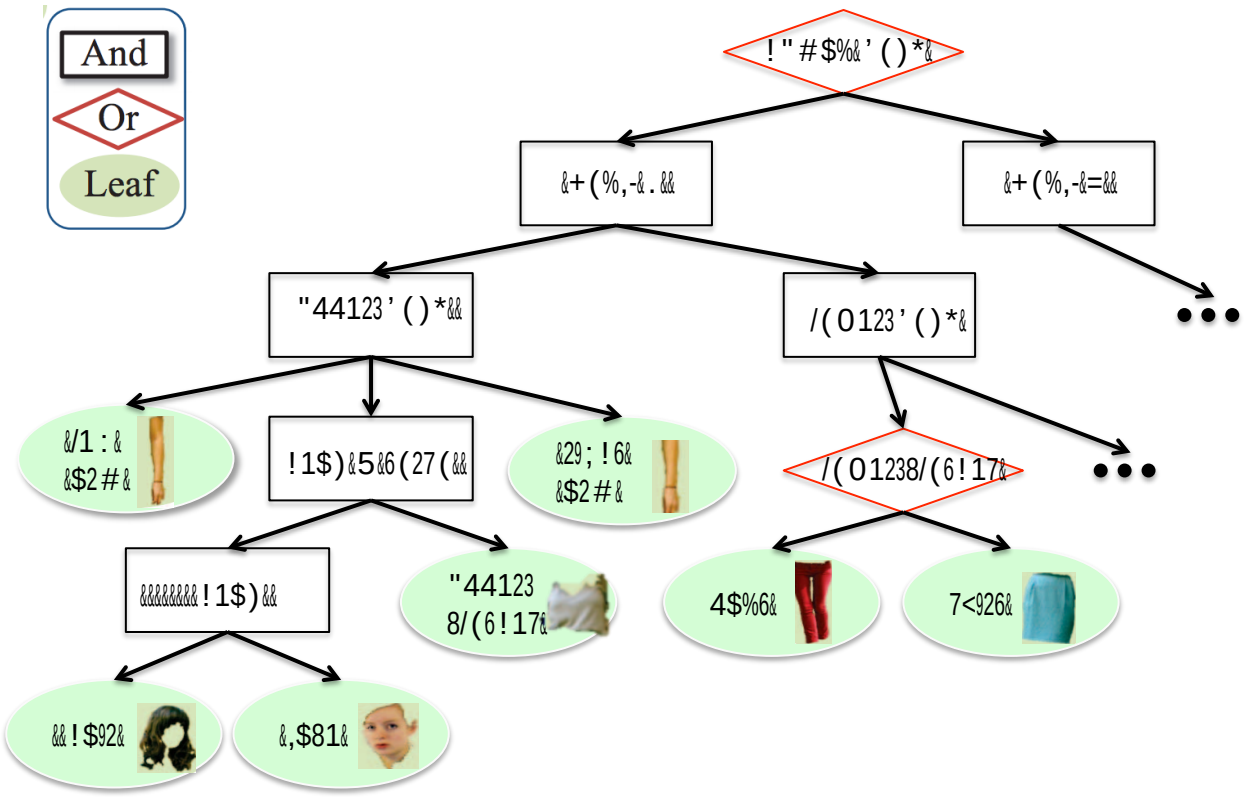


Figure 2.2: Illustration of And-Or graph model on human semantic part segmentation.

$\mathcal{V}^O(t)$, $\mathcal{V}^A(t)$, and $\mathcal{V}^L(t)$ are active or-nodes, and-nodes, and leaf nodes respectively. For an and-node $v \in \mathcal{V}^A$, we define $z_{kids(v)} = \{z_\mu | \mu \in kids(v)\}$ as the state variables of all the child nodes of v. For an or-node $v \in \mathcal{V}^O$, we use z_{t_v} to denote the state of its selected child node. For a leaf node $v \in \mathcal{V}^L$, z_v denotes the region/segment proposal chosen for part v, linking the model to the image data. Given an image I, the AOG computes the optimal state variable z by minimizing the energy function Equ. (2.2).

$$E(z|I) = \sum_{\mu \in \mathcal{V}^O(t)} E^O(z_\mu) + \sum_{\mu \in \mathcal{V}^A(t)} E^A(z_\mu, z_{kids(\mu)}) + \sum_{\mu \in \mathcal{V}^L(t)} E^L(I|z_\mu) \tag{2.2}$$

The potential $E^O(.)$ encodes the prior distribution for the choices of an or-node. The potential $E^A(.)$ captures the geometry interaction between an and-node and all its child nodes, e.g. it can encode relative location and scale between the and-node and its child nodes. For leaf nodes, $E^L(.)$ matches the selected region/segment part proposal to image, which can be the output of a part score regressor.

We can formulate the optimization of AOG under the structural learning framework, and effectively solve the model by the cutting plane algorithm [JFY09].

2.3 Conditional Random Field (CRF)

Conditional random fields (CRFs) are a type of undirected probabilistic graphical models that are often used for image object segmentation. Give input variables X and output variables $Y = \{y_1, y_2, ..., y_n\}$, a CRF models the conditional distribution $P(Y|X)$, considering not only how each individual output variable y_i matches the input X but also the dependency between each pair of connected output variables. In this section, we regard each human part as a separate class and introduce how we apply a simple CRF to an image to acquire pixel-wise part label map for human semantic part segmentation.

The graph structure for the CRF is $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Given an image I of n pixels, we define each pixel as a node: $\mathcal{V} = \{v_1, v_2, ..., v_n\}$, and define an edge between each pair of neighboring pixels: $\mathcal{E} = \{(v_i, v_j)\}$. The input variables for the CRF are the pixel values of I. The output variables are the pixel-wise part lables $L = \{l_i | i = 1, 2, ..., n\}$. Suppose there are N part categories including background, each l_i takes value from $1, 2, ..., N$. The energy function to minimize is defined as:

$$E(L|I) = \sum_{v_i \in \mathcal{V}} \Phi(l_i|I) + \lambda \sum_{(v_i, v_j) \in \mathcal{E}} \Psi(l_i, l_j|I), \tag{2.3}$$

where $\Phi(.)$ and $\Psi(.)$ are the unary potential and pairwise potential respectively, and λ is the model parameter.

The unary potential $\Phi(l_i|I)$ is defined by the output score of a part classifier. For example, we can learn a logistic regression model for each part type, based on a feature vector $\phi(I_{v_i})$ extracted from the neighboring region of pixel v_i. The logistic regression models output $P(l_i|\phi(I_{v_i}))$, and we set $\Phi(l_i|I)$ as in Equ. (2.4). Traditionally, the feature vector $\phi(I_{v_i})$ consists of a rich set of manually designed features based on color intensities and edges. Recently, the feature vector can also be stronger deep-learned features.

$$\Phi(l_i|I) = ln\ P(l_i|\phi(I_{v_i})) \tag{2.4}$$

The pairwise potential $\Psi(l_i, l_j|I)$ can also be defined using logistic regression models trained for each part type pair. For part type pair l_i and l_j, the logistic regression model predicts $P(l_i, l_j|\psi(I_{v_i}, I_{v_j}))$ and we define $\Psi(l_i, l_j|I)$ as in Equ. (2.5).

$$\Psi(l_i, l_j|I) = ln\ P(l_i, l_j|\psi(I_{v_i}, I_{v_j})) \tag{2.5}$$

Depending on the graph structure $\mathcal{E}$, we use different approaches to perform inference on a CRF. If the graph is a tree, message passing algorithms give exact solutions. If the CRF only contains pair-wise potentials and the energy function is submodular, exact solutions are also possible by running Max-Flow/Min-Cut algorithms. If exact inference is impossible, there are several approximation approaches as well. Recently, fully-connected CRF is very popular in image segmentation, which adds an edge between each pair of pixels and thus considers long-range interactions in the image. [KK11] proposes an efficient approximation inference algorithm on fully-connected CRF based on mean field theory while [IPA16] performs approximation inference using linear programming relaxations.

2.4 Deep Neural Networks for Pose Estimation and Segmentation

With the advent of powerful deep learning techniques [LKF10] and the availability of large-scale annotated data on pose estimation and segmentation (e.g. ImageNet [DDS09], MS-COCO [LMB14] and PASCAL-Part [CML14]), dramatic progress has been made on both human pose estimation and human semantic part segmentation.

For both tasks, there are several popular deep convolutional neural network (DCNN) structures that researchers adopt for different purposes. To name a few, VGG-16 [SZ14], DeepLab [CPK15a], GoogLeNet [SLJ15], Deep Residual Net [HZR15], etc.

On human pose estimation, DCNNs are usually used to compute unary score maps for all joint types and predict pairwise relationship between forehead and neck, neck and left/right

Figure 2.3: Illustration of deep-learned feature maps used in human pose estimation. (a): unary joint score maps; (b) regression prediction (from left shoulder to all the other joints).

shoulder, and other adjacent joint type pairs. These feature maps are later used in a graphical model to infer the optimal pose configuration. As shown in Fig. 2.3, [IPA16] uses deep-learned joint score maps as unary terms for a fully-connected CRF, and defines pairwise terms based on deep-learned neighboring joint prediction maps. The unary score maps learned by DCNNs are very accurate compared with heatmaps produced by traditional part classifiers, therefore a simple assembling model suffice for relatively simple images. However, for natural multi-person scenes, a strong assembling model is necessary and some difficulties haven't been effectively solved, e.g. large pose/scale variation, multi-person confusion, and so on.

For semantic part segmentation, recent approaches adopt fully convolutional networks (FCNs) to directly produce pixel-wise part labels, yielding superior performance to traditional graphical models. However, FCN-type methods give coarse boundary details due to FCN's inherent invariance property, which is undesirable for the fine-grained task of semantic part segmentation. Moreover, these methods still make local confusion errors when the person is in a non-typical pose, or when there are multiple people overlapping each other. An efficient mechanism to handle the large variation of pose/scale and to handle the multi-person issue is needed.

CHAPTER 3

Related Works

As human pose estimation and semantic part segmentation are two fundamental topics in computer vision, many graphical models on the two topics have been proposed during the past thirty years. With the advent of powerful deep learning techniques, some models start to use deep-learned features, or even perform the whole task within a deep neural network. Recently, there are also works that combine pose estimation and segmentation in graphical models. In this chapter, we give an overall review of previous literature in these aspects.

3.1 Human Pose Estimation

Part-based graphical models are popular in human pose estimation due to the fact that human instances are highly articulated. Binford [Bin71] expressed that objects in 2D images can be modeled as collections of interrelated atomic parts and proposed generalized cylinder models. Fischler and Elschlager [FE73] introduced the basic Pictorial Structural model (PS), a tree model with local part scores and kinematic pairwise terms. However, exact inference of PS on dense pixels was too time consuming at that time. Fast inference on PS was made possible by Felzenszwalb and Huttenlocher [FH05], who proposed efficient distance transform algorithms. Later, Andriluka [ARS09] improved Pictorial Structures by learning stronger part filters using Adaboost classifiers built on shape-context features of each part. Traditional models like PS suffer from two disadvantages. They are tree-structured models with traditional hand-crafted features, so they have limited representation power and can't handle large variations in pose and appearance. Besides, the pairwise terms are just geometric priors between part pairs, not strong and not data-dependent.

To overcome these difficulties, some researchers resort to cyclic models and mixture of tree models. [TF10] designed a fully connected graph of limbs and performed approximation inference by local greedy search.Most cyclic models are very time consuming and usually take minites to perform exact inference. [WM08] adopted mixture of tree models to reason about occlusion and spatial constraints while [ZR12] used mixture of tree models for face landmark localization, capturing a broad range of pose. Modes in these models are usually learned from data, and may not have semantic meanings. These models only rely on hand-crafted features and are still limited in handling complex natural images.

To enhance the representation power of previous methods, multimodal compositional models are proposed. [YR11] learned modes only at the part level, considered compositional parts that connect to simple parts and modeled their geometric constraints. [ST] treated the entire human body as a mixture of templates. [ZM07, ZCL08, RPZ13] explored mixtures at the middle level in hierarchical models, and introduced And-Or graph (AOG), a multi-level mixture of MRFs. In an AOG, an and-node represents a decomposition of a mid-level part, while an or-node is a switch that chooses among a set of its child and-nodes.

Recently, graphical models are combined with deep-learned features for even stronger representation power. [CY14, IPA16] used a deep convolutional neural network (DCNN) to predict unary joint score maps and relative positions of neighbouring joints, and combined them as unary and pairwise terms in a graphical model. [CY14] clustered the relative positions of a neighbouring joint into 11 clusters and used the DCNN to predict the cluster id, while [IPA16] directly regress the positions of neighbouring joints from the image. For the graphical model, [CY14] used a tree model while [IPA16] adopted a fully-connected CRF.

There are also some works that give up the pose assembling graphical model by modeling dependencies of joints within DCNNs. [TS14, CAF15] regressed the location for all the joints directly and modified the locations in a recurrent manner. [COL16] added extra convolutional layers in a DCNN to model the dependency of joints in a tree structure, using the score map of one joint to predict the score map of its neighbouring joints. These models perform well on relatively simple datasets, but a good assembling model is really necessary to handle the large pose variation in natural multi-person datasets.

3.2 Human Semantic Part Segmentation

Traditional methods on human semantic part segmentation fall into two major categories. One type of methods first generate region/segment proposals for each semantic part, then select and assemble the region proposals by a graphical model. [BF11] generated region proposals by UCM segmentation [AMF09], ranked the region proposals using shape and appearance features, and assembled the proposals with simple geometric constraints. [DCX13] got region proposals by UCM and CPMC [CS12], extracted a rich set of appearance features for each region proposal encoded by Fisher Kernel and second-order pooling, and assembled the region proposals with a dedicated And-Or graph (AOG). The other type of methods treat semantic part segmentation as scene parsing, adopting pixel-wise conditional random fields (CRFs) to infer the pixel-wise part labels. [YLO12] built a CRF based on super-pixels and tried to classify each super-pixel into one of the semantic part types. They learned unary part classifiers for super-pixels based on traditional features, and used the output scores as unary terms in the CRF. For the pairwise terms, they trained logistic regression models to predict whether two neighboring pixels should have the same label, and also learned a consistency prior between each part type pair from the training data. These traditional methods perform well on simple images, but struggle in natural images with large pose/scale variation.

Over the past few years, DCNNs have pushed the performance to new heights in many computer vision tasks such as image classification, object detection, and fine-grained categorization. For semantic part segmentation, there are two directions in recent methods that deal with DCNNs. Some extract deep-learned features for region proposals, and assemble the region proposals using a graphical model based on those features [XZW16]. Some use fully convolutional networks (FCNs) to output pixel-wise part labels directly [LSD15, CPK15b, XWC16]. Graphical models with DCNN features perform well on relatively simple datasets and give good details of parts, but they struggle in complex natural datasets with large pose/scale variation. FCN-type methods handle the pose variation better, but only give very coarse boundary details of parts, which is not desirable for part segmentation. Besides,

the scale variation in multi-person natural images hasn't been explored.

To improve FCN-type methods on boundary details, [CPK15a] added a post-processing fully-connected CRF to the part segmentation FCN, giving clearer details of big parts such as head and torso. However, the boundary details of small flexible parts, such as arms and legs, are far from satisfactory. [CPK15a] also performed multi-scale score fusion inside FCN, solving the object scale variation to some extent. However, arms and legs of small-scale people are often missed and local confusion errors still occur for large-scale people. Furthermore, the scale variation of human parts hasn't been explored.

3.3 Combining Pose Estimation and Part Segmentation

Recently, some researchers consider the correlation between pose estimation and segmentation. [YLO12] performed human pose estimation and semantic part segmentation sequentially for clothes parsing. [LTZ13] proposed a framework for joint pose estimation and part labeling under a CRF. [DCS14] solved pose estimation and part segmentation within an AOG-based framework. These methods demonstrate the complementary properties of pose estimation and part segmentation, but they can't handle images with large pose variation and multi-person overlapping due to the use of less powerful features in the graphical models and the poor quality of region proposals for parts.

In this dissertation, we propose several approaches that combine DCNNs with dedicated graphical models, greatly boosting the representation power of models to handle large pose variation. We also introduce part segment consistency for pose estimation and pose consistency for part segmentation, further improving the performance for multi-person natural images.

CHAPTER 4

Pose-Guided Human Semantic Part Segmentation in Constrained Scenes

There are some scenarios in our daily life that only require good part segmentation results for images under constrained conditions: the image is well-cropped to mainly contain one person, the background is relatively simple, and people have relatively standard poses such as standing or sitting. In these cases, standard end-to-end deep models like fully convolutional networks (FCNs) [LSD15] don't necessarily outperform dedicated graphical models (see Fig. 4.1) because the inherent invariance of deep models causes ignorance of localization/edge details, which is undesirable for semantic part segmentation. Also, it's worth noting that we can get reliable (roughly right) pose estimation in these images, which may provide helpful cues for part segmentation. Therefore, we propose a part segmentation framework that follows the traditional proposal-assembling pipeline (i.e. segment/region proposals

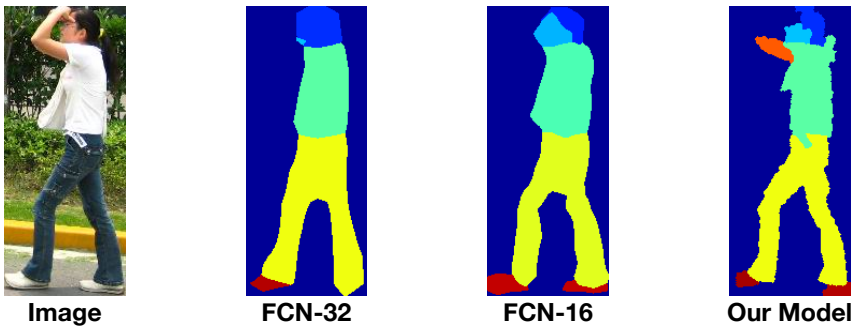


Figure 4.1: Human semantic part segmentation on Penn-Fudan Pedestrian Dataset. Images are roughly cropped bounding boxes for pedestrians. Our model gives better boundary details than standard FCNs [LSD15] of two scales.

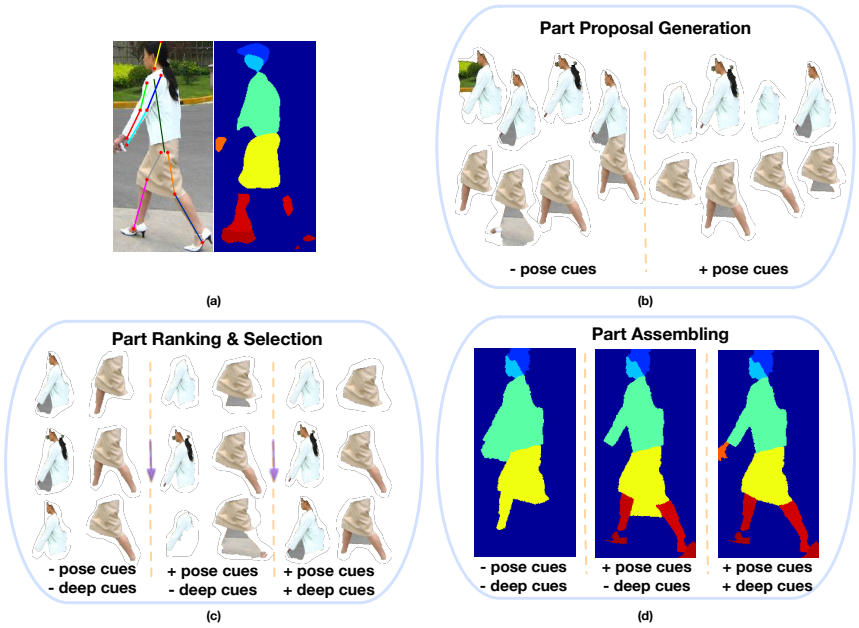


Figure 4.2: Human part segmentation using pose (pose-guided-proposals and pose-context features) and deep-learned part semantic cues.

are first generated for human parts and then a graphical model is used to select and integrates these proposed segments [YLO12, YLL14]), and uses top-down pose information for all stages of the pipeline: segment proposal generation, proposal ranking & selection, and proposal assembling. We visually demonstrate the effectiveness of our proposed pose cues in Fig. 4.2.

Here we elaborate a little on our overall pipeline. As shown in Fig. 4.3, given an input image, we first (bottom left) use a state-of-the-art pose estimation algorithm [CY14] to estimate the locations of the joints and other salient parts of humans. We use the estimates of the joint positions to obtain *pose-guided-proposals* for part segments (top left) based on the intuition that part segments should be correlated to joint positions (e.g., the lower-arm should appear between the wrist and the elbow), which yields a limited set of proposals with high recall. Next we compute rich feature descriptors for each segment proposal, including a novel *pose-context* feature which captures spatial/geometrical relationship between a proposed segment and the estimated human pose joints. We also use standard appearance features

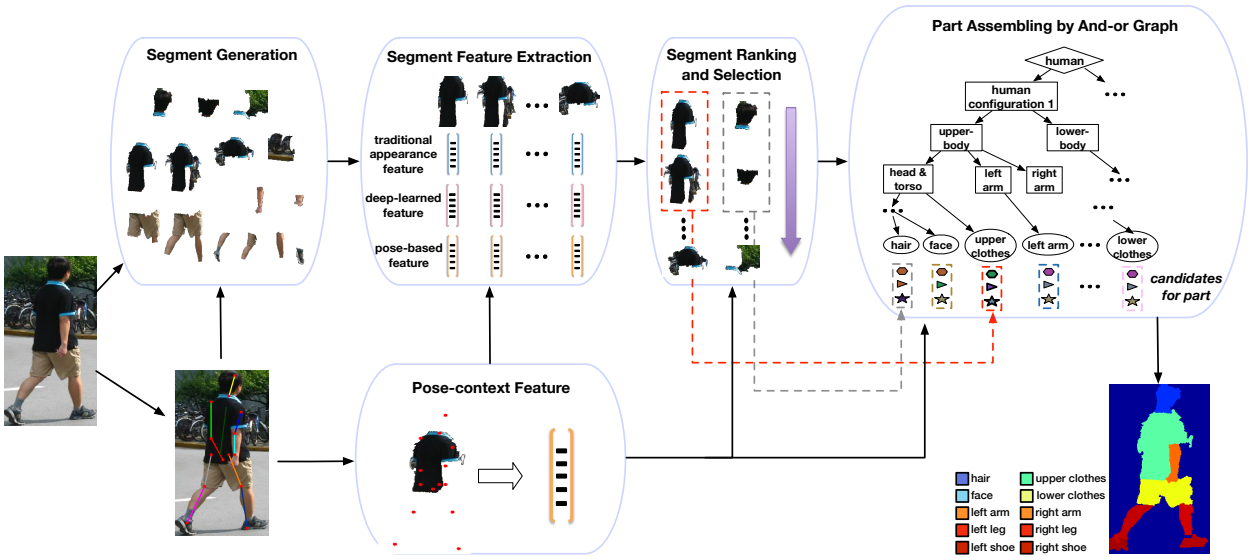


Figure 4.3: Illustration of our human part segmentation pipeline.

and complementary deep-learned part semantic features computed by a fully convolutional network (FCN) [LSD15, HAG15b, CPK15a, TKP15]. Then we rank the segment proposals based on these features and select the top-ranked ones. This leaves a small number of high-quality proposals for each part category which are used as input to the part assembling stage.

For part assembling, we propose an And-Or graph (AOG) [ZM07, ZCL08, ZWZ12, WY15], which is an efficient way to represent the large variability of human appearances. We perform inference over this AOG to select and combine part segment proposals so as to parse the human body. Compared with traditional AOGs, our AOG has more flexible and efficient structure (i.e. each leaf node allows arbitrary number of data-mined part subtypes) and includes an extension of the *pose-context* feature as a pairwise term to measure the compatibility of adjacent parts.

We evaluate our method on a popular pedestrian parsing benchmark dataset, *Penn-Fudan* [WSS07], and show that our approach outperforms other state-of-the-arts by a significant margin.

4.1 Pose-Guided Human Part Segmentation Pipeline

Given a pedestrian image I, we first adopt a state-of-the-art pose estimation approach [CY14] to estimate human pose joints $\mathcal{L} = \{l_1, l_2, \cdots, l_{14}\}$, where l_j denotes the location of the j-th pose joint. As shown in Fig. 4.3, based on the human pose cues, our human part segmentation pipeline has three successive steps: *part segment proposal generation*, *part proposal selection*, and *part assembling*. We will introduce the first two steps below, and elaborate on our AOG-based part assembling method in the next section.

4.1.1 Pose-Guided Part Segment Proposal Generation

To generate part segment proposals, we modify the RIGOR algorithm [HLR14], which can efficiently generate segments aligning with object boundaries given user defined initial seeds and cutting thresholds. In this paper, we propose to generate the seeds based on the estimated pose joint locations. Specifically, given the observation that part segments tend to be surrounding corresponding pose joints, for each joint we sample a set of seeds at the 5×5 grid locations over a 40×40 image patch centered at this joint. We use 8 different cutting thresholds, yielding about 200 segment proposals for each joint. Combining proposals from all the joints, we further prune out duplicate segments (with intersect-over-union (IOU) $\geq$ 0.95 as threshold) and construct a segment pool $\mathcal{S} = \{s_1, s_2, \cdots, s_N\}$ that contains around 800 segment proposals for each image. We use these segments as candidate part segments in the latter two steps.

4.1.2 Part Proposal Selection

We consider the following image features for each segment proposal $s_i \in \mathcal{S}$: (i) $\phi^{o2p}(s_i)$, a second order pooling (O2P) feature [CCB12] for describing appearance cues; (ii) $\phi^{skin}(s_i)$, an appearance feature [KH10] capturing skin color cues; (iii) $\phi^{pose}(s_i, \mathcal{L})$, a pose-context feature we propose in this paper, which measures the spatial relationship between the segment s_i and the predicted pose joint configuration $\mathcal{L}$; (iv) $\phi^{c-pose}(s_i, \mathcal{L})$, a non-linearly coded version

of $\phi^{pose}(s_i, \mathcal{L})$; (v) $\phi^{fcn}(s_i, \mathcal{H})$, a deep-learned semantic feature using FCN, which measures the compatibilities between the segment image patch and high-level part semantic cues from FCN.

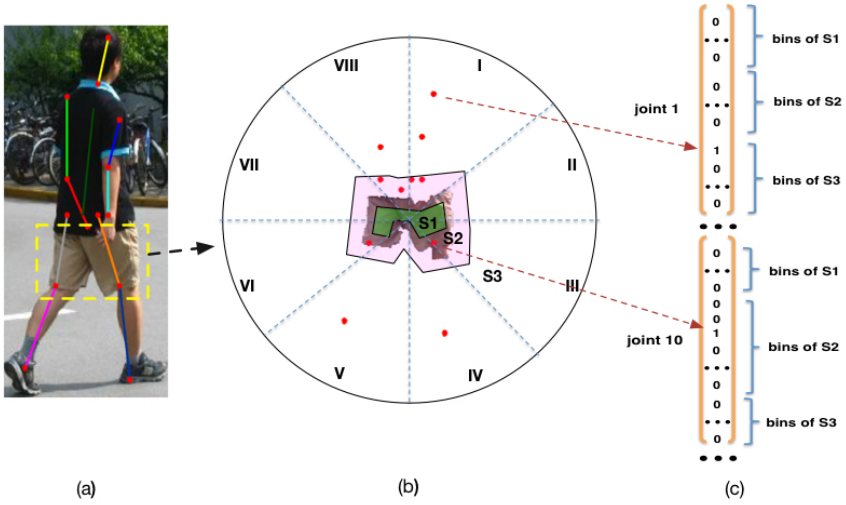


Figure 4.4: Illustration of our proposed pose-context feature.

We now describe the proposed pose-context feature $\phi^{pose}(s_i, \mathcal{L})$. As shown in Fig. 4.4, centered at s_i, the image is equally divided into eight orientations ($\mathrm{I}-\mathrm{VIII}$) and three region scales (S1, S2 and S3), yielding 24 spatial bins in total. Then each joint $l_j \in \mathcal{L}$ falls into one of these spatial bins, producing a binary feature to quantize the spatial relationship of l_j w.r.t. s_i. After that, we concatenate binary features of all the joints, and obtain a $24 \times 14 = 336$ dimensional pose-context feature to describe the spatial relationship of s_i w.r.t. $\mathcal{L}$. Specifically, S1 and S2 are the regions eroded and dilated by 10 pixels from the segment's boundary respectively. S3 is the rest region of image. This segment-dependent definition of region scales depicts semantically meaningful geometric cues from the predicted pose information, e.g. the lower boundary of the short skirt segment should be around the knee joints. The three-scale design (rather than using the segment edge alone) makes the feature robust to pose estimation errors.

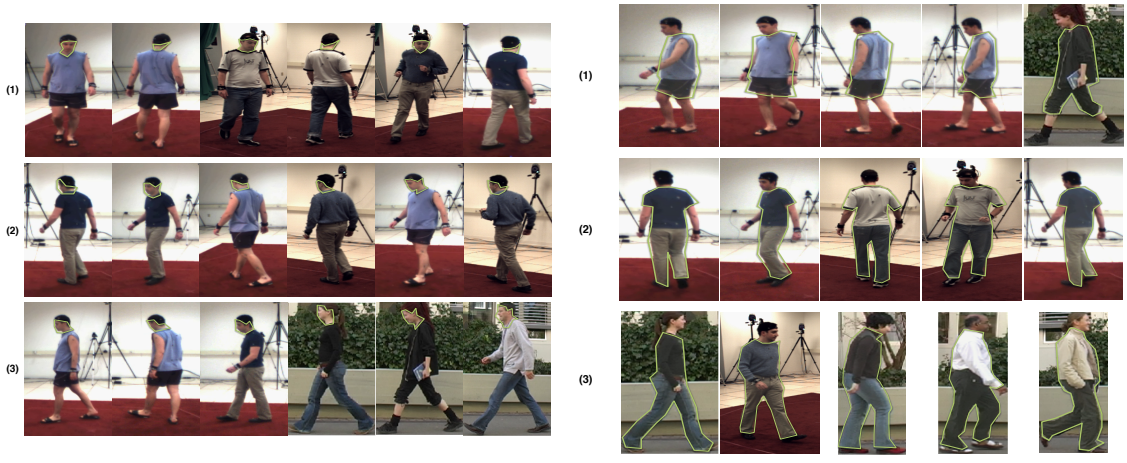
The pose-context feature can be highly non-linear in the feature space, which might be suboptimal for linear classifiers/regressors. This motivates us to apply non-linear coding

technology [YYG09, WYY10] on the pose-context feature to achieve linearity. We adopt soft-assignment quantization (SAQ) coding [LWL11] to encode the pose-context feature into its coded version $\phi^{c-pose}(s_i, \mathcal{L})$, with a dictionary of pose-guided part prototypes $\mathcal{D} = \{\mathbf{b}_m\}_{m=1}^{N_\mathcal{D}}$, learned via K-means clustering algorithm [HW79] on the pose-context feature representation of ground-truth part segment examples. Specifically, to balance K different part categories, we separately perform clustering and obtain $N_p = 6$ prototypes/clusters for each part category, resulting in a dictionary of $N_\mathcal{D} = K \times N_p$ codewords. Given $\mathcal{D}$, we compute the Euclidean distance between original pose-context feature of s_i and each prototype $\mathbf{b}_m$: $d_{i,m} = \parallel \phi^{pose}(s_i, \mathcal{L}) - \mathbf{b}_m \parallel$. Thus $\phi^{c-pose}(s_i, \mathcal{L})$ is formally defined as the concatenation of both the normalized and un-normalized codes w.r.t. $\mathcal{D}$:

$$\phi^{c-pose}(s_i, \mathcal{L} \,|\, \mathcal{D}) = [a_{i,1}, \cdots, a_{i,N_\mathcal{D}}, a'_{i,1}, \cdots, a'_{i,N_\mathcal{D}}]^{\mathrm{T}} \tag{4.1}$$

where $a_{i,m} = \exp^{(-\lambda d_{i,m})}$ and $a'_{i,m} = \frac{a_{i,m}}{\sum_{j=1}^{N_\mathcal{D}} a_{i,j}}$ denote the un-normalized and normalized code values w.r.t. $\mathbf{b}_m$ respectively. λ is a hyper-parameter of our coding method. The coded pose-context feature is adopted in training the support vector regression (SVR) models for part proposal selection.

The learned part prototypes, which generally correspond to different viewpoints of a part or different appearance patterns of a part (e.g. long pants or skirts for the lower-clothes category), are used to define part subtypes in our AOG. As illustrated in Fig. 4.5a, the learned face prototypes generally correspond to different typical views of the face category. We show exemplar images for 3 out of 6 clusters. The other clusters correspond to the symmetric patterns w.r.t. those shown here. Besides, we propose to encode the pairwise pose-context feature (i.e. concatenated pose-context features of a pair of candidate segments), used as a pairwise term in our AOG design. We perform clustering separately for each adjacent part pair and learn a class-specific dictionary for this pairwise pose-context feature. In this paper, the dictionary size is set by $N_{pp} = 8$ for each part pair. As visualized in the right figure of Fig. 4.5b, the learned part-pair prototypes are very meaningful which capture typical viewpoints and part type co-occurrence patterns for adjacent parts. We show 3 out of 8 clusters.



(a) (b)

Figure 4.5: The learned prototypes/clusters for (a) part category *face*; (b) adjacent part pair *upper-clothes* and *lower-clothes*.

For the deep-learned semantic feature, we train a FCN-16s deep neural network [HAG15b, WSL15] with the output to be the part ground truth map, and then produce pixel-wise part potential maps $\mathcal{H}$, from which binary part label masks $\mathcal{B}$ can be obtained via argmax over the potential maps. Thus, for a segment s_i, this deep feature $\phi^{fcn}(s_i, \mathcal{H})$ consists of three components: (1) the mean value inside s_i of $\mathcal{H}$ for each part class; (2) the mean value along the contour of s_i from $\mathcal{H}$ for each part class; (3) The IoU value between s_i and $\mathcal{B}$ for each part class.

Our final feature descriptor of s_i is the concatenation of the aforementioned features, i.e.

$$\begin{aligned}\phi(s_i, \mathcal{L}, \mathcal{H}) = [&\phi^{o2p}(s_i),\ \phi^{skin}(s_i),\ \phi^{fcn}(s_i, \mathcal{H}),\\ &\phi^{pose}(s_i, \mathcal{L}), \phi^{c-pose}(s_i, \mathcal{L})]^{\mathrm{T}}\end{aligned} \tag{4.2}$$

On basis of this hybrid feature representation, we train a linear support vector regressor (SVR) [CCB12] for each part category. Let P denote the total number of part categories and $p \in \{1, 2, \cdots, P\}$ denote the index of a part category. The target variable for training SVR is the IoU value between the segment proposal and ground-truth label map of part p. The output of SVR is given by Equ. (4.3),

$$g^p(s_i|\mathcal{L}, \mathcal{H}) = {\beta_p}^{\mathrm{T}} \phi(s_i, \mathcal{L}, \mathcal{H}), \tag{4.3}$$

where β_p is the model parameter of SVR for the p-th part category. Thus, for any part p, we rank the segment proposals in $\mathcal{S}$ based on their SVR scores $\{g^p(s_i) \,|\, s_i \in \mathcal{S}\}$. Finally, we select the top-n_p scored segments separately for each part category and combine the selected segment proposals from all part categories to form a new segment pool $\tilde{\mathcal{S}} \subseteq \mathcal{S}$.

4.2 Part Assembling with And-Or Graph

There are two different groups of classes (i.e. *parts* and *part compositions*) in our AOG model: the part classes are the finest-level constituents of human body; the part compositions correspond to intermediate concepts in the hierarchy of semantic human body constituents. Specifically, we define them as follows. *Parts*: hair, face, full-body clothes, upper-clothes, left/right arm, lower-clothes, left/right leg skin, left/right shoe. *Part Compositions*: head, head & torso, upper-body, left/right leg, human body.

To assemble the selected part segments, we develop a compositional AOG model as illustrated in Fig. 4.6a, which facilitates flexible composition structure and standard learning/inference routines. Let P and C denote the number of parts and the number of part compositions respectively. Formally, our AOG model is defined as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{T} \cup \mathcal{N}$ denotes a set of vertices and $\mathcal{E}$ refers to the set of edges associated. Meanwhile, $\mathcal{T} = \{1, 2, \cdots, P\}$ and $\mathcal{N} = \{P+1, P+2, \cdots, P+C\}$ denote the set of part indices and the set of part composition indices respectively. In our AOG, each leaf vertex $p \in \mathcal{T}$ represents one human body part and each non-leaf vertex $c \in \mathcal{N}$ represents one part composition. The root vertex corresponds to the whole human body while the vertices below correspond to the part compositions or parts at various semantic levels. Our goal is to parse the human body into a series of part compositions and parts, which is in a hierarchical graph instantiated from the AOG model.

The vertex of our AOG is a nested subgraph as illustrated at the bottom of Fig. 4.6a. For a leaf vertex $p \in \mathcal{T}$, it includes one Or-node followed by a set of terminal nodes as its children. The terminal nodes correspond to different part subtypes learned by clustering the pose-context feature of training part segments, and the Or-node represents a mixture

model indicating the selection of one part subtype from terminal nodes. Formally, we define a state variable $z_p \in \{0, 1, 2, \cdots, K_p\}$ to indicate that the Or-node selects the z_p-th terminal node as the part subtype for leaf vertex p. As an example of a green node in Fig. 4.6a, the lower-clothes part can select one kind of subtype (e.g. long pants or skirt) from its candidate part subtypes. In addition, there is one special terminal node representing the invisibility of the part due to occlusion/self-occlusion, which corresponds to the state $z_p = 0$. For a non-leaf vertex $c \in \mathcal{N}$, it includes one Or-node linked by a set of And-nodes plus one terminal node. The Or-node of non-leaf vertex represents this part composition has several different ways of decompositions into smaller parts and/or part compositions. The And-node corresponds to one possible configuration of the decomposition of c. As shown in Fig. 4.6a, the non-leaf vertex head can be composed by one of several different configurations of two child vertices (i.e., face and hair). Similar to the leaf vertices, we also induce a state variable $z_c \in \{0, 1, 2, \cdots, K_c\}$ to indicate that the Or-node of part composition c selects the z_c-th And-node as the configuration of child vertices for $z_c \neq 0$ or this part composition is invisible when $z_c = 0$.

Further, we define another state variable y to indicate the selection of segment from the candidate pool of a part or part composition. For a leaf vertex $p \in \mathcal{T}$, $y_p \in \{0, 1, 2, \cdots, n_{p,z_p}\}$ represents that the part p selects the y_p-th segment proposal (i.e., $s_{y_p}^{p,z_p}$) from the segment pool $\tilde{\mathcal{S}}_{p,z_p}$, outputted by its segment ranking model on subtype z_p. Meanwhile, $y_p = 0$ is a special state which coincides with the invisibility pattern of part p (i.e., $z_p = 0$). To make the notations consistent, we use s_0^{p,z_p} to represent an "null" segment for part invisibility. For a non-leaf vertex $c \in \mathcal{N}$, $y_c \in \{0, 1, 2, \cdots, n_c\}$ indicates a segment $s_{y_c}^{c,z_c} \in \tilde{\mathcal{S}}_{c,z_c}$ is selected, where $s_{y_c}^{c,z_c}$ is obtained by the union of its child vertices' candidate segments and $\tilde{\mathcal{S}}_{c,z_c}$ denotes the candidate segment pool for the z_c And-node. When $y_c = 0$, likewise, the s_0^{c,z_c} represents a null segment indicating the invisibility pattern of part composition c.

Let $Ch(c, z_c)$ denote the set of child vertices for part composition c and configuration z_c. Formally, $s_{y_c}^{c,z_c}$ is defined by Equ. (4.4), where $\Cup$ represents a pixel-wise union operation of

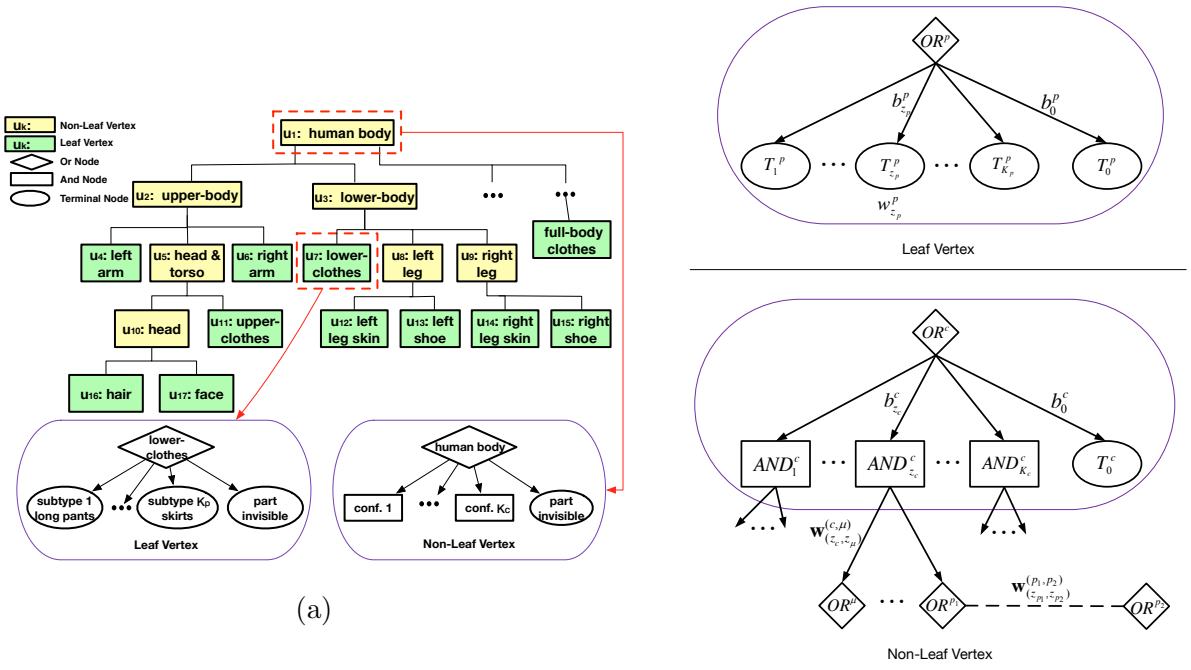


Figure 4.6: Illustration of our AOG design. (a) Architecture of AOG. (b) Structure of vertices in AOG; the symbols of *OR*, *AND* and T represent the Or-node, And-node and terminal node respectively.

combing the child vertices' segment masks to generate a new segment.

$$s_{y_c}^{c,z_c} = \mathbb{U}_{\mu \in Ch(c,z_c)} s_{y_\mu}^{\mu,z_\mu} \tag{4.4}$$

Let $\mathbf{Y} = (y_1, y_2, \cdots, y_P, y_{P+1}, y_{P+2}, \cdots, y_{P+C})$ and $\mathbf{Z} = (z_1, z_2, \cdots, z_P, z_{P+1}, z_{P+2}, \cdots, z_{P+C})$ denote the structural solution of AOG. We define a global score function of AOG $F(\mathbf{Y}, \mathbf{Z} \mid \tilde{\mathcal{S}}, \mathcal{L}, \mathcal{H})$ (here $\tilde{\mathcal{S}} = \bigcup_{p \in \mathcal{T}, z_p \neq 0} \tilde{\mathcal{S}}_{p,z_p}$) to measure the compatibility between $(\mathbf{Y}, \mathbf{Z})$ and $(\tilde{\mathcal{S}}, \mathcal{L}, \mathcal{H})$ for image I, which can be calculated as shown in Equ. (4.5), where $f(y_p, z_p)$ is a local score function of leaf vertex p, which consists of only one unary term, while $f(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\})$ denotes a score function of non-leaf vertex c, which consists of two pairwise terms.

$$F(\mathbf{Y}, \mathbf{Z} | \tilde{\mathcal{S}}, \mathcal{L}, \mathcal{H}) = \sum_{p \in \mathcal{T}} f(y_p, z_p) + \sum_{c \in \mathcal{N}} f(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\}) \tag{4.5}$$

Part Composition (c)	Adjacent Part Pairs ($\mathcal{R}_c$)
human body	(upper-clothes, lower-clothes), (full-body clothes, left leg skin), (full-body clothes, right leg skin)
head	(hair, face)
head & torso	(upper-clothes, hair), (upper-clothes, face), (full-body clothes, hair), (full-body clothes, face)
upper-body	(left arm, upper-clothes), (right arm, upper-clothes), (left arm, full-body clothes), (right arm, full-body clothes)
lower-body	(lower-clothes, left leg skin), (lower-clothes, right leg skin), (lower-clothes, left shoe), (lower-clothes, right shoe)
left leg	(left leg skin, left shoe)
right leg	(right leg skin, right shoe)

Table 4.1: The list of adjacent part pairs in our AOG design.

For each leaf vertex $p \in \mathcal{T}$ (i.e. a part), we compute $f(y_p, z_p)$ by Equ. (4.6), in which $w^p_{z_p}$ and $b^p_{z_p}$ denote the weight and bias parameters of unary term for part p respectively. Particularly, b^p_0 is the bias parameter for the invisibility pattern of p. Besides, $g^p_{z_p}$ is dependent on the part subtype z_p, implying the regression models defined in Equ. (4.3) are trained by different parts and subtypes. Fig. 4.6b illustrates the structure of a leaf vertex and its corresponding model parameters.

$$f(y_p, z_p) = \begin{cases} b^p_{z_p} + w^p_{z_p} \cdot g^p_{z_p}(s^{p,z_p}_{y_p}|\mathcal{L}, \mathcal{H}), & z_p \neq 0 \\ b^p_0, & z_p = 0 \end{cases} \tag{4.6}$$

For each non-leaf vertex $c \in \mathcal{N}$ (i.e. a part composition), we compute $f(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\})$ by Equ. (4.7), where $u(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\})$ is defined in Equ. (4.8).

$$\begin{aligned} &f(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\}) = \\ &\quad \begin{cases} b^c_{z_c} + u(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\}), & z_c \neq 0 \\ b^c_0, & z_c = 0 \end{cases} \end{aligned} \tag{4.7}$$

$$\begin{aligned} u(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\}) = &\sum_{\mu \in Ch(c, z_c)} \mathbf{w}^{(c,\mu)}_{(z_c, z_\mu)}{}^{\mathrm{T}} \varphi(s^{c,z_c}_{y_c}, s^{\mu,z_\mu}_{y_\mu}) + \\ &\sum_{(p_1,p_2) \in \mathcal{R}_c} \mathbf{w}^{(p_1,p_2)}_{(z_{p_1}, z_{p_2})}{}^{\mathrm{T}} \psi(s^{p_1,z_{p_1}}_{y_{p_1}}, s^{p_2,z_{p_2}}_{y_{p_2}}|\mathcal{L}) \end{aligned} \tag{4.8}$$

Concretely, Equ. (4.7) can be divided into three terms:

(1) the bias term of selecting z_c for the Or-node, i.e. $b^c_{z_c}$. b^c_0 is the bias parameter when part composition c is invisible (In this case, all the descendant vertices are also invisible

and thus the latter two terms are zero).

(2) the sum of parent-child pairwise terms (i.e., *vertical edges*) for measuring the spatial compatibility between the segment of part composition c and the segments of its child vertices, i.e. $\sum_{\mu \in Ch(c,z_c)} \mathbf{w}_{(z_c,z_\mu)}^{(c,\mu)\,\mathrm{T}} \varphi(s_{y_c}^{c,z_c}, s_{y_\mu}^{\mu,z_\mu})$, where $\varphi(s_{y_c}^{c,z_c}, s_{y_\mu}^{\mu,z_\mu})$ denotes a spatial compatibility feature of segment pair $(s_{y_c}^{c,z_c}, s_{y_\mu}^{\mu,z_\mu})$ and $\mathbf{w}_{(z_c,z_\mu)}^{(c,\mu)}$ refers to corresponding weight vector. Specifically, φ is defined by $[dx; dx^2; dy; dy^2; ds; ds^2]$, in which dx, dy represent the spatial displacement between the center locations of two segments while ds is the scale ratio of them.

(3) the sum of pairwise terms (i.e. *side-way edges*) for measuring the geometric compatibility on all segment pairs specified by an adjacent part-pair set $\mathcal{R}_c$, which defines a couple of adjacent part pairs for c (e.g., for the part composition of lower body, we consider lower-clothes and leg skin to be an adjacent part pair). Tab. 4.1 lists the adjacent part pairs for each non-leaf vertex. To avoid double counting in recursive computation of Equ. (4.8), $\mathcal{R}_c$ only includes the relevant part pairs which have at least one child vertex of c. This side-way pairwise potential corresponds to $\sum_{(p_1,p_2)\in\mathcal{R}_c} \mathbf{w}_{(z_{p_1},z_{p_2})}^{(p_1,p_2)\,\mathrm{T}} \psi(s_{y_{p_1}}^{p_1,z_{p_1}}, s_{y_{p_2}}^{p_2,z_{p_2}}|\mathcal{L})$ in Equ. (4.8), where $\psi(s_{y_{p_1}}^{p_1,z_{p_1}}, s_{y_{p_2}}^{p_2,z_{p_2}}|\mathcal{L})$ represents a geometric compatibility feature of segment pair $(s_{y_{p_1}}^{p_1,z_{p_1}}, s_{y_{p_2}}^{p_2,z_{p_2}})$ and $\mathbf{w}_{(z_{p_1},z_{p_2})}^{(p_1,p_2)}$ is corresponding weight vector. In this paper, we use a coded version of pose-context feature for ψ. Specifically, we adopt the same coding process as in $\phi^{c-pose}(s_i, \mathcal{L})$ but using the concatenated pose-context features for segment pair $(s_{y_{p_1}}^{p_1,z_{p_1}}, s_{y_{p_2}}^{p_2,z_{p_2}})$.

In Fig. 4.6b, we illustrate the structure of a non-leaf vertex and its corresponding model parameters.

4.3 Learning and Inference for And-Or Graph

The score function in Equ. (4.5) is a generalized linear model w.r.t. its parameters. We can concatenate all the model parameters to be a single vector $\mathbf{W}$ and rewrite Equ. (4.5) by $F(\mathbf{Y}, \mathbf{Z}|\mathcal{L}, \tilde{\mathcal{S}}, \mathcal{H}) = \mathbf{W}^{\mathrm{T}}\mathbf{\Phi}(\mathcal{L}, \tilde{\mathcal{S}}, \mathcal{H}, \mathbf{Y}, \mathbf{Z})$. $\mathbf{\Phi}(\mathcal{L}, \tilde{\mathcal{S}}, \mathcal{H}, \mathbf{Y}, \mathbf{Z})$ is a re-organized sparse vector

gathering all the features based on the structural state variable $(\mathbf{Y}, \mathbf{Z})$. In our AOG model, $\mathbf{Z}$ determines the topological structure of a feasible solution (i.e., parse tree), and $\mathbf{Y}$ specifies the segments selected for the vertices of this parse tree. Given a set of labelled examples $\{(\mathbf{Y}_n, \mathbf{Z}_n) \,|\, n = 1, 2, \cdots, J\}$, we formulate a structural max-margin learning problem on $\mathbf{W}$ (Equ. (4.9)), where $\Delta(\mathbf{Y}_n, \mathbf{Z}_n, \mathbf{Y}, \mathbf{Z})$ is a structural loss function to penalize a hypothesized parse tree $(\mathbf{Y}, \mathbf{Z})$ different from ground truth annotation $(\mathbf{Y}_n, \mathbf{Z}_n)$.

$$\begin{aligned} &\min_{\mathbf{W}} \frac{1}{2}\mathbf{W}^{\mathrm{T}}\mathbf{W} + C\sum_{n=1}^{J}\xi_n, \;\; s.t. \; \forall \; \mathbf{Y} \text{ and } \mathbf{Z}: \\ &\mathbf{W}^T\mathbf{\Phi}(\mathcal{L}_n, \tilde{\mathcal{S}}_n, \mathcal{H}_n, \mathbf{Y}_n, \mathbf{Z}_n) - \mathbf{W}^T\mathbf{\Phi}(\mathcal{L}_n, \tilde{\mathcal{S}}_n, \mathcal{H}_n, \mathbf{Y}, \mathbf{Z}) \geq \Delta(\mathbf{Y}_n, \mathbf{Z}_n, \mathbf{Y}, \mathbf{Z}) - \xi_n \end{aligned} \tag{4.9}$$

Similar to [YBS13], we adopt a relative loss as in Equ. (4.10), i.e. the loss of hypothesized parse tree relative to the best one $(\mathbf{Y}^*, \mathbf{Z}^*)$ that could be found from the candidate pool. That is,

$$\Delta(\mathbf{Y}_n, \mathbf{Z}_n, \mathbf{Y}, \mathbf{Z}) = \delta(\mathbf{Y}_n, \mathbf{Z}_n, \mathbf{Y}, \mathbf{Z}) - \delta(\mathbf{Y}_n, \mathbf{Z}_n, \mathbf{Y}^*, \mathbf{Z}^*), \tag{4.10}$$

where $\delta(\mathbf{Y}, \mathbf{Z}, \mathbf{Y}', \mathbf{Z}') = \sum_{p\in\mathcal{T}} IoU(s_{y_p}^{p,z_p}, s_{y_p'}^{p,z_p'})$ is a function of measuring the part segmentation difference between any two parse trees $(\mathbf{Y}, \mathbf{Z})$ and $(\mathbf{Y}', \mathbf{Z}')$. We employ the commonly-used cutting plane algorithm [JFY09] to solve this structural max-margin optimization problem of Equ. (4.9).

For inference on AOG models, dynamic programming (DP) is commonly used in the literature. Our model, however, contains side-way pairwise terms which form closed loops. These closed loops are fairly small so DP is still possible. We combine the DP algorithm with state pruning for model inference, which has a bottom-up scoring step and a top-down backtracking step.

For the bottom-up scoring step, we compute the score of each vertex (to be specific, the score of the subgraph rooted at that vertex) in a bottom-up manner, only retraining the top-k scored candidate state configurations of each vertex for subsequent inference. The score of a subgraph $\mathcal{Q} \subseteq \mathcal{G}$ ($\mathcal{Q} = (\mathcal{V}_Q, \mathcal{E}_Q)$) is defined in Equ. (4.11), which is equivalent to Equ. (4.5) when $\mathcal{Q} = \mathcal{G}$. We set $k = 10$, making the inference procedure tractable with a moderate number of state configurations for each vertex. We show in a diagnostic experiment that this

greedy pruning scarcely affects the quality of the result while reducing the inference time significantly.

$$F_Q(\mathbf{Y}_Q, \mathbf{Z}_Q|\tilde{\mathcal{S}}, \mathcal{L}, \mathcal{H}) = \sum_{p\in\mathcal{T}\cap\mathcal{V}_Q} f(y_p, z_p) + \sum_{c\in\mathcal{N}\cap\mathcal{V}_Q} f(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\}) \quad (4.11)$$

After getting the score of the root vertex (i.e., whole human body), we backtrack the optimum state value from the retained top-k list for each vertex in a top-down manner. Concretely, for each part composition vertex c we select the best scored state configuration value of $(y_c, z_c, \{(y_\mu, z_\mu) : \mu \in Ch(c, z_c)\})$, and recursively infer the optimum state values of the selected child vertices given each $\mu \in Ch(c, z_c)$ as the root vertex of a subgraph. In the end, we can obtain the best parse tree from the pruned solution space of our AOG, and output corresponding state values $(\mathbf{Y}, \mathbf{Z})$ to produce the final part segmentation result.

4.4 Experimental Evaluation

We evaluate our algorithm on the Penn-Fudan benchmark [WSS07], which consists of pedestrians in outdoor scenes with much pose variation. Because this dataset only provides testing data, following previous works, we train our part segmentation models using the HumanEva dataset [SB06], which contains 937 images with pixel-level label maps for parts annotated by [BF11]. The labels of the two datasets are consistent, which include 7 body parts: hair, face, upper-clothes, lower-clothes, arms (arm skin), legs (leg skin), and shoes. For the pose model, we use the model provided by [CY14], trained on the Leeds Sports Pose Dataset [JE10].

4.4.1 Effectiveness of Pose in the Model

Effectiveness of pose for part proposal generation. We first investigate how the pose cues help the part proposal generation. Specifically, we compare our pose-guided segment proposal method with the baseline algorithm, i.e. the standard RIGOR algorithm [HLR14].

For evaluating the proposal algorithms, two standard criteria are used, i.e. average part recall (APR) and average part oracle IOU (AOI). The first measures how much portion of

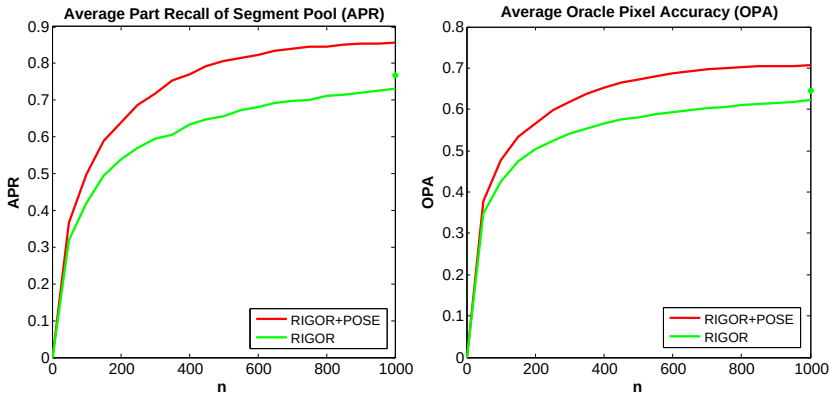


Figure 4.7: Comparison of our part segment proposal method (RIGOR+POSE) to the baseline (RIGOR). The green asterisks on the plots represent the APR/AOI of the RIGOR pool for the pool size $n = 2000$.

the ground-truth segments is covered (i.e., over 50% IOU) by the proposals, and the second measures the best IOU the proposal pool can achieve on average of all ground-truth segments. As shown in Fig. 4.7, our method significantly improves the quality of part segment proposals compared to the baseline by over 10% on average.

Effectiveness of features for part proposal selection. To investigate various features and their complementary properties for part proposal selection, we sequentially add them into our SVR model and test the performance/quality of the selected part segments.

In Tab. 4.2, we report the AOI scores for the top-1 ranked part segment and the top-10 ranked part segments respectively. Firstly, we can see the performance monotonically improves with more features used, which demonstrates the effectiveness of all features we proposed. By comparing (2) and (3), we can see a significant boost of the top-1 accuracy, indicating that the pose information becomes much more effective with the coded pose context feature. Finally, by adding the deep semantic feature in (4), the performance of selected part segment improves further. We set $n_p = 10$ because it strikes a good trade off between the quality and pool size of the selected part segments.

Methods	hair	face	u-cloth	l-cloth	arms	legs	shoes	mean
(1): *o2p* + *skin*	57.1	53.5	70.9	70.9	26.6	20.4	15.6	45.0
	68.8	66.9	80.0	81.4	54.6	55.3	45.3	64.6
(2): (1) + *pose*	61.7	58.6	73.2	72.7	29.9	23.4	17.5	48.1
	69.9	66.4	80.6	82.3	56.4	54.3	45.8	65.1
(3): (2) + *c-pose*	61.8	58.9	73.2	71.9	39.8	44.8	26.5	53.8
	69.9	66.4	80.5	82.4	55.8	59.1	47.4	65.9
(4): (3) + *fcn*	64.4	59.0	77.4	77.1	41.4	43.6	35.1	56.9
	70.7	66.6	82.2	83.4	55.9	59.3	48.8	66.7

Table 4.2: Comparison of four part models by AOI score (%) for top-1 ranked segment (top) and top-10 ranked segments (bottom). Models are numbered as (1) to (4), from top to bottom.

Effectiveness of the AOG. To show the effectiveness of our AOG design, we set up two experimental baselines for comparison: (1) Naive Assembling: considering only the unary terms and basic geometric constraints as defined in the paper [BF11], e.g. upper-clothes and lower-clothes must be adjacent. (2) Basic AOG: considering only the unary terms and the parent-child pairwise terms, without the side-way pairwise terms.

Tab. 4.3 shows that the basic AOG with parent-child spatial relations outperforms the naive assembling model, and by adding the pairwise side-way edges, the performance boosts further, which demonstrates the effectiveness of each component in our AOG model. For comparison, we also test the result of the AOG model without state pruning, which clearly justifies the use of state pruning in AOG inference. We can see that state pruning leads to neglectable decrease in accuracy while it reduces the inference runtime significantly, from 2 min. to 1 sec. per image.

4.4.2 Comparisons to the State of the Art

We compare our approach with four state-of-the-art methods, namely FCN [WSL15], SBP [BF11], P&S [RC12], and DDN [LWT13]. Specially, for FCN, we use the code provided by [WSL15]

Methods	hair	face	u-cloth	arms	l-cloth	legs	Avg
Naive Assembling	62.3	53.5	77.8	36.9	78.3	28.2	56.2
Basic AOG	63.1	52.9	77.1	38.0	78.1	35.9	57.5
Ours	**63.2**	**56.2**	**78.1**	**40.1**	**80.0**	45.5	**60.5**
Ours (w/o pruning)	63.2	56.2	78.1	40.1	80.0	**45.8**	60.5

Table 4.3: Per-pixel accuracy (%) of our AOG and two baselines.

Method	hair	face	u-cloth	arms	l-cloth	legs	shoes	Avg*
FCN [WSL15]	48.7	49.1	70.2	33.9	69.6	29.9	**36.1**	50.2
P&S [RC12]	40.0	42.8	75.2	24.7	73.0	46.6	-	50.4
SBP [BF11]	44.9	**60.8**	74.8	26.2	71.2	42.0	-	53.3
DDN [LWT13]	43.2	57.1	77.5	27.4	75.3	**52.3**	-	56.2
Ours	**63.2**	56.2	**78.1**	**40.1**	**80.0**	45.5	35.0	**60.5**

Table 4.4: Comparison of our approach with other state-of-the-art methods on the Penn-Fudan dataset in terms of per-pixel accuracy (%). The Avg* means the average without shoes class since it was not reported in other methods.

and re-train the networks with our training set.

As shown in Tab. 4.4, our model outperforms the FCN by over 10% and the state-of-the-art DDN method by over 4%, from which we can see most improvement is from small parts such as hair and arms. It implies that by using the pose cues we can produce high-quality segment candidates that align to the boundaries for small parts. In addition, our AOG model together with the pose-context feature can leverage long-range spatial context information, making our model robust in shape variations and appearance ambiguities.

We illustrate our typical part segmentation results in Fig. 4.8, which are generally good. The typical faliure cases of our model is listed in Fig. 4.9, due to color confusion with other objects, multiple instance occlusion, and large variation in lighting respectively, which generally fail most of current human part segmentation systems. For the first and the third failure cases, we got accurate pose estimation but failed to generate satisfactory segment

proposals for lower-clothes, which suggests that we either adopt stronger shape cues in the segment proposal stage or seek richer context information (e.g. handbag in the first case). For the second case, we got a bad pose estimation due to occlusion and thus mixed two people's parts during assembling, which indicates the necessity of handling instance-level pose estimation or segmentation.

4.5 Conclusion

In this chapter, we present an AOG-based human part segmentation approach that performs well on relatively simple images. We integrate top-down pose information into all three stages of our framework (i.e. pose-guided part proposals and pose-context features in part selection & assembling), obtaining state-of-the-art segmentation accuracy on a benchmark human parsing dataset, Penn-Fudan. We propose semantically meaningful *pose-context* features that describe the geometric relationship between segment and pose joints. Our AOG design has flexible composition structure. We show extensive experimental results that validate the

Figure 4.8: Qualitative evaluation of our method on Penn-Fudan.

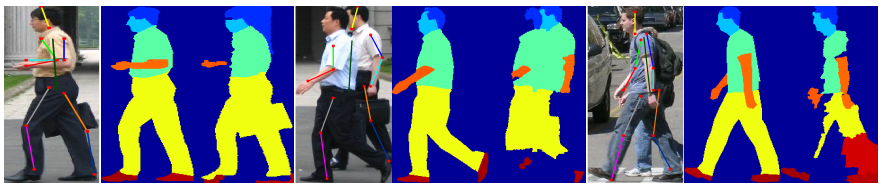


Figure 4.9: Typical failure cases of our method.

effectiveness of each component of our framework.

Observing the failure cases of this model (see Fig. 4.9), we find that: (1) we need a stronger pose estimation model that can handle complex multi-person scenes where people in the image may overlap with each other and have large pose variation; (2) we should try to combine deep neural networks with graphical models in a more efficient way to better utilize their complementary role in the human part segmentation task. These two directions are explored in Chapter 6.

CHAPTER 5

Handling Scale Variation of Objects and Parts in Natural Scenes

Human semantic part segmentation has mainly been studies under constrained conditions which pre-suppose known scale, fairly accurate localization, clear appearances, and/or relatively simple poses [BF11, ZCL11, EW12, YLO12, DCS14, LLL15]. There are few works done on parsing animals, like cows and horses, yet these also face similar restrictions, e.g. roughly known size and location [WY15, WSL15]. Only very recently, people start to address the task of parsing objects, such as humans and animals, in "the wild" where there are large variations in scale, location, occlusion, and pose. One big difficulty of this task is the large variability of scale and location for objects and their corresponding parts. Even limited mistakes in estimating scale and location will degrade the part segmentation output and cause errors in boundary details.

With the emergence of fully convolutional networks (FCNs) [LSD15] and the availability of object part annotations on large-scale datasets, e.g. PASCAL and MS-COCO, some deep-based methods have made big improvements on object parsing [HAG15a, WSL15]. However, these methods can still make mistakes on small or large scale objects and, in particular, they have no mechanism to adapt to the size of the object.

In this chapter, we will introduce a hierarchical method for object part segmentation that performs scale estimation and object parsing jointly and is able to adapt its scale to objects and parts. It is partly motivated by the proposal-free end-to-end detection strategies [HYD15, RHG15], which prove that the scale and location of a target object, and of its corresponding parts, can be estimated accurately from the field-of-view (FOV) window by

applying a deep neural network (Fig. 5.1a). We call our approach "Hierarchical Auto-Zoom Net" (HAZN) which parses the objects at three levels of granularity, namely image-level, object-level, and part-level, gradually giving clearer and better part segmentation results (see Fig. 5.1b). The HAZN sequentially combines two "Auto-Zoom Nets" (AZNs), each of which predicts the locations and scales for objects (the first AZN) or parts (the second AZN), properly zooms (resizes) the predicted image regions, and refines the object parsing results for those image regions (see Fig 5.2). The HAZN uses three FCNs [LSD15] that share the same structure. The first FCN acts directly on the image to estimate a finite set of possible locations and sizes of objects (e.g. bounding boxes) with confidence scores, together with a part score map of the image. The part score map is similar to that proposed by previous deep-learned methods. The object bounding boxes are scaled to a fixed size by zooming in or zooming out (as applicable) and the image and part score maps within the boxes are also scaled by bilinear interpolation for zooming in or downsampling for zooming out. Then the second FCN is applied to the scaled object bounding boxes to make proposals (bounding boxes) for the parts, with confidence values, and to re-estimate the part scores within the object bounding boxes. This yields improved part scores. We then apply the third FCN to the scaled part bounding boxes to produce new estimates of the part scores and to combine all of them (for different object and part bounding boxes) to output final part scores, which are our parse of the object. This strategy is modified slightly so that we scale humans differently depending on whether we have detected a complete human or only the upper part of a human, which can be determined from the part score map.

For dealing with scale, the adaptiveness of our approach and the way it combines scale estimation with parsing give novel computational advantages over traditional multi-scale methods. Previous methods mainly select a fixed set of scales and then perform fusion on the outputs of a deep net at different layers. Computational requirements mean that the number of scales must be small and it is impractical to use very fine scales due to memory limitations. Our approach is considerably more flexible because we adaptively estimate scales at different regions in the image which allows us to search over a large range of scales. In particular, we can use very fine scales because we will probably only need to do this within

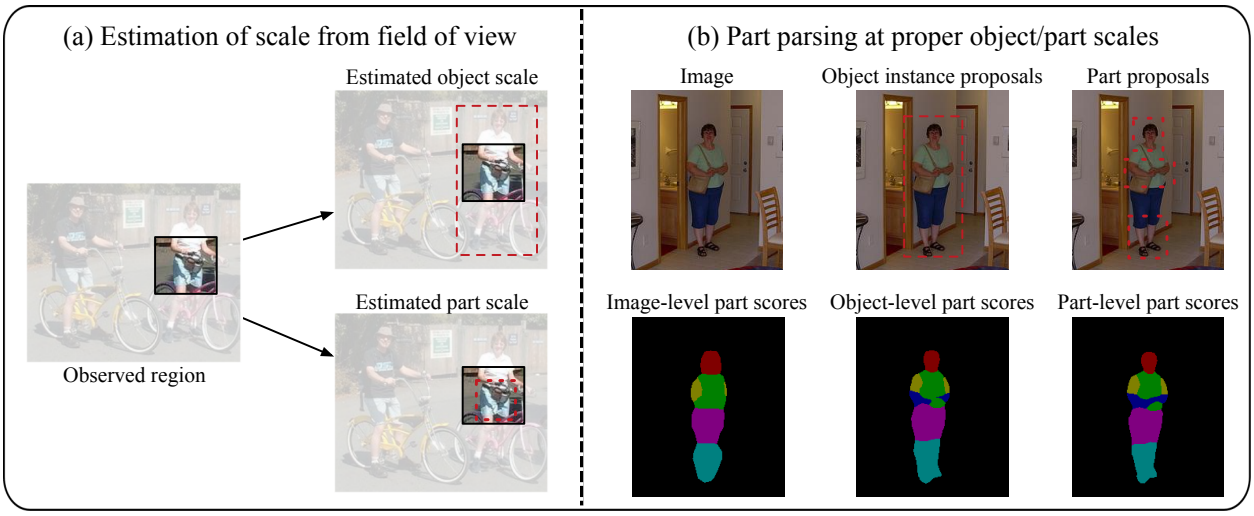


Figure 5.1: Intuition of Hierarchical Auto-Zoom Net (HAZN). (a) The scale and location of an object and its parts (the red dashed boxes) can be estimated from the observed field of view (the black solid box) of a neural network. (b) Part segmentation can be more accurate by using proper object and part scales. At the top row, we show our estimated object and part scales. In the bottom row, our part parsing results gradually become better by increasingly utilizing the estimated object and part scales.

small image regions. For example, our largest zooming ratio is 2.5 (at part level) on PASCAL while that number is 1.5 if we have to zoom the whole image. This is a big advantage when trying to detect small parts, such as the tail of a cow, as is shown by the experiments.

We report extensive experimental results for parsing humans on the challenging PASCAL-Person-Part dataset [CML14] and for parsing animals on a horse-cow dataset [WY15]. Our approach outperforms previous state-of-the-arts by a large margin. We are particulary good at detecting small object parts. The reason why we mainly work on PASCAL images [EEG14] is that these images were chosen for studying multiple visual tasks, do not suffer from dataset design bias [LHK14], and include large variations of objects, particularly of scale. Parsing humans in PASCAL is considerably more difficult than in other datasets like Fashionista [YLO12], which were constructed solely to evaluate human part segmentation.

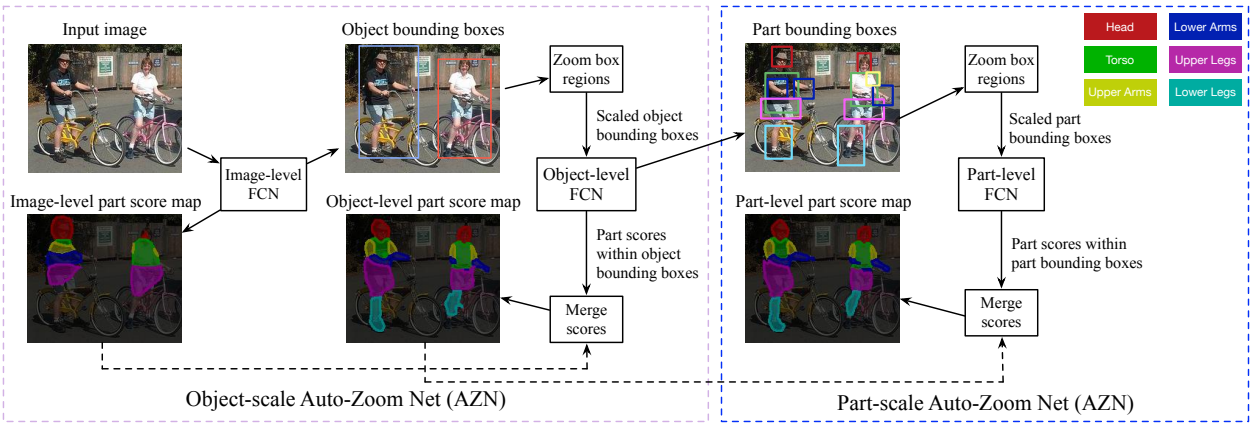


Figure 5.2: Testing framework of HAZN. We address object part segmentation by adapting to the sizes of objects (object-scale AZN) and parts (part-scale AZN). The part scores are predicted and refined by three FCNs, over three levels of granularity, i.e. image-level, object-level, and part-level. At each level, the FCN outputs the part score map for the current level, and estimates the locations and scales for the next level. The details of parts are gradually discovered and improved along the proposed auto-zoom process (i.e. location/scale estimation, region zooming, and part score re-estimation).

5.1 Hierarchical Auto-Zoon Net (HAZN)

As shown in Fig. 5.2, our Hierarchical Auto-Zoom model (HAZN) has three levels of granularity for tackling scale variation in object parsing, i.e. image-level, object-level, and part-level. At each level, a fully convolutional neural network (FCN) is used to perform scale/location estimation and part parsing simultaneously. The three levels of FCNs are all built on the same network structure, a modified FCN called DeepLab-LargeFOV [CPK15b]. This network structure is one of the most effective FCNs in segmentation, so we also treat it as our baseline for final performance comparison.

To handle scale variation in objects and parts, the HAZN concatenates two Auto-Zoom Nets (AZNs), namely object-scale AZN and part-scale AZN, into a unified network. The object-scale AZN refines the image-level part score map with object bounding box proposals while the part-scale AZN further refines the object-level part score map with part bounding

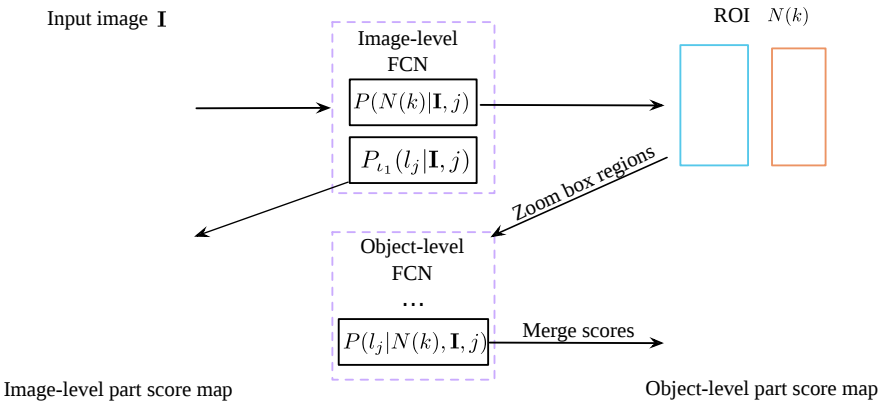


Figure 5.3: Object-scale Auto-Zoom Net from a probabilistic view, which predicts ROI region $N(k)$ at object-scale, and then refines part scores based on the properly zoomed region $N(k)$.

box proposals. Each AZN employs an auto-zoom process: first estimates the region of interest (ROI), then properly resizes the predicted regions, and finally refines the part scores within the resized regions.

5.1.1 Object-Scale Auto-Zoom Net (AZN)

For the task of object part parsing, we are provided with n training examples $\{\mathbf{I}_i, \mathbf{L}_i\}_{i=1}^n$, where $\mathbf{I}$ is the given image and $\mathbf{L}$ is the pixel-wise semantic part labels. Our target is to learn the posterior distribution $P(l_j|\mathbf{I}, j)$ for each pixel j of an image $\mathbf{I}$, which is approximated by our object-scale AZN (see Fig. 5.3).

We first use the image-level FCN (see Fig. 5.2) to produce the image-level part score map $P_{\iota_1}(l_j|\mathbf{I}, j)$, which gives comparable performance to our baseline method (DeepLab-LargeFOV). This is a normal ***part parsing network*** that uses the original image as input and outputs the pixel-wise part score map. Our object-scale AZN aims to refine this part score map with consideration of object instance scales. To do so, we add a second component to the image-level FCN, performing regression to estimate the size and location of an object bounding box (or ROI) for each pixel, together with a confidence map indicating the likelihood that the box is an object. This component is called a ***scale estimation network*** (**SEN**), which shares the first few layers with the part parsing network in the

image-level FCN. In math, the SEN corresponds to a probabilistic model $P(b_j|\mathbf{I}, j)$, where b_j is the estimated bounding box for pixel j, and $P(b_j|...)$ is the confidence score of b_j.

After getting $\{b_j|\forall j \in \mathbf{I}\}$, we threshold the confidence map and perform non-maximum suppresion to yield a finite set of object ROIs (typically 5-10 per image, with some overlap): $\{b_k|k \in \mathbf{I}\}$. Each b_k, the bounding box estimated from pixel k, is associated with a confidence score $P(b_k)$. As shown in Fig. 5.2, a **region zooming** operation is then performed on each b_k, resizing b_k to a standard-sized ROI $N(k)$. Specifically, this zooming operation computes a zooming ratio for bounding box b_k, and then enlarges or shrinks the image within b_k by the zooming ratio. We will discuss how to compute the zooming ratio in Sec. 5.2.

Now we have a set of zoomed ROI proposals $\{N(k)|k \in \mathbf{I}\}$, each $N(k)$ associated with score $P(b_k)$. We learn another probabilistic model $P(l_j|N(k), \mathbf{I}, j)$, which re-estimates the part label for each pixel j within the zoomed ROI $N(k)$. This probabilistic model corresponds to the part parsing network in the object-level FCN (see Fig. 5.2), which takes as input the zoomed object bounding boxes and outputs the part scores within those object bounding boxes.

The new part scores for the zoomed ROIs need to be merged to produce the object-level part score map for the whole image. Since there may be multiple ROIs that cover a pixel j, we define the neighbouring region set for pixel j as $\mathcal{Q}(j) = \{N(k)|j \in N(k), k \in \mathbf{I}\}$. Under this definition of $\mathcal{Q}(j)$, the **score merging** process can be expressed as Equ. (5.1), which essentially computes the weighted sum of part scores for pixel j, from the zoomed ROIs that cover j. For a pixel that is not covered by any zoomed ROI, we simply use its image-level part score as the current part score. Formally, the object-level part score $P_{\iota_2}(l_j|\mathbf{I}, j)$, is computed as,

$$\begin{aligned} P_{\iota_2}(l_j|\mathbf{I}, j) &= \sum\nolimits_{N(k)\in\mathcal{Q}(j)} P(l_j|N(k), \mathbf{I}, j)P(N(k)|\mathbf{I}, j) \\ P(N(k)|\mathbf{I}, j) &= P(b_k)/\textstyle\sum_{k:N(k)\in\mathcal{Q}(j)} P(b_k) \end{aligned} \tag{5.1}$$

5.1.2 Part-Scale Auto-Zoom Net

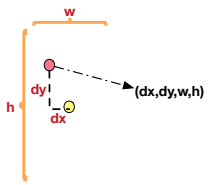
The scale of object parts can also vary considerably even if the scale of the object is fixed. This leads to a hierarchical strategy with multiple stages, called the Hierarchical Auto-Zoom Net (HAZN), which applies AZNs to images to find objects and then on objects to find parts, followed by a part score refinement stage. As shown in Fig. 5.2, we add the part-scale AZN to the end of the object-scale AZN. Specifically, we add a second component (i.e. SEN) to the object-level FCN, to estimate the size and location of part bounding boxes, together with confidence maps for every pixel within a zoomed object ROI. Again the confidence map is thresholded, and non-maximal suppresion is applied, to yield a finite set of part ROIs (typically 5-30 per image, with some overlap). Each part ROI is zoomed to a fixed size. Then, we re-estimate the part scores within each zoomed part ROI using the part parsing network in the part-level FCN. The part parsing network is the only component of the part-level FCN, which takes the zoomed part ROI and the zoomed object-level part scores (within the part ROI) as inputs. After getting the part scores within each zoomed Part ROI, the score merging process is the same as in the object-scale AZN.

It's worth mentioning that we can easily extend our HAZN to include more AZNs at finer scale levels if we focus on smaller object parts such as human eyes.

5.1.3 Training and Testing Phases for Object-Scale AZN

We use **DeepLab-LargeFOV** [CPK15b] as the basic network structure for both the scale estimation network (SEN) and the part parsing network. The two networks, serving as components of a multi-tasking FCN, share the first three layers.

Training the SEN. The SEN aims to regress the region of interest (ROI) for each pixel j in the form of a bounding box, b_j. Here we borrow the idea of DenseBox [HYD15] for scale estimation, since it is simple and performs well enough for our task. In detail, at object level, the ROI of pixel j corresponds to the object instance box that pixel j belongs to. For training the SEN, two output label maps are needed as visualized in Fig. 5.4. The first one



Input image | Object bounding box regression map | Object confidence seeds

Figure 5.4: Ground truth regression target for training the scale estimation network (SEN) in the image-level FCN. Details in Sec. 5.1.3.

is the bounding box regression map $\mathbf{L}_b$, which is a four-channel output for each pixel j to represent its ROI b_j: $\mathbf{l}_{bj} = \{dx_j, dy_j, w_j, h_j\}$. Here (dx_j, dy_j) is the relative position from pixel j to the center of b_j; h_j and w_j are the height and width of b_j. We then re-scale the outputs by dividing them with 400. The other target output map is a binary confidence seed map $\mathbf{L}_c$, in which $\mathbf{l}_{cj} \in \{0, 1\}$ is the ROI selection indicator at pixel j. It indicates the preferred pixels for us to use for ROI prediction, which helps the algorithm prevent many false positives. In practice, we choose the central pixels of each object instance as the confidence seeds, which tend to predict the object bounding boxes more accurately than those pixels at the boundary of an object instance region.

Given the ground-truth label maps of object part parsing, we can easily derive the training examples for the SEN: $\mathcal{H} = \{\mathbf{I}_i, \mathbf{L}_{bi}, \mathbf{L}_{ci}\}_{i=1}^n$, where n is the number of training instances. We minimize the negative log likelihood to learn the weights $\mathbf{W}$ for the SEN, and the loss l_{SEN} is defined in Equ. (5.2).

$$
\begin{aligned}
l_{SEN}(\mathcal{H}|\mathbf{W}) &= \frac{1}{n}\sum\nolimits_i (l_b(\mathbf{I}_i, \mathbf{L}_{bi}|\mathbf{W}) + \lambda l_c(\mathbf{I}_i, \mathbf{L}_{ci}|\mathbf{W}));\\
l_c(\mathbf{I}, \mathbf{L}_c|\mathbf{W}) &= -\beta \sum_{j:l_{cj}=1} \log P(l^*_{cj} = 1|\mathbf{I}, \mathbf{W}) - (1-\beta) \sum_{j:l_{cj}=0} \log P(l^*_{cj} = 0|\mathbf{I}, \mathbf{W})\\
l_b(\mathbf{I}, \mathbf{L}_b|\mathbf{W}) &= \frac{1}{|\mathbf{L}^+_{cj}|}\sum\nolimits_{j:l_{cj}=1} \|\mathbf{l}_{bj} - \mathbf{l}^*_{bj}\|^2
\end{aligned}
\tag{5.2}
$$

For the confidence seeds, we employ the balanced cross entropy loss, where l^*_{cj} and l_{cj} are the predicted value and ground truth value respectively. The probability is from a sigmoid function performing on the activation of the last layer of the CNN at pixel j. β is defined

as the proportion of pixels with $l_{cj} = 0$ in the image, which is used to balance the positive and negative instances. The loss for bounding box regression is the Euclidean distance over the confidence seed points, and $|\mathbf{L}_{cj}^{+}|$ is the number of pixels with $l_{cj} = 1$.

Testing the SEN. The SEN outputs both the confidence score map $P(l_{cj}^{*} = 1|\mathbf{I}, \mathbf{W})$ and a four-dimensional bounding box $\mathbf{l}_{bj}^{*}$ for each pixel j. We regard a pixel j with confidence score higher than 0.5 to be reliable and output its bounding box $b_j = \mathbf{l}_{bj}^{*}$, associated with confidence score $P(b_j) = P(l_{cj}^{*} = 1|\mathbf{I}, \mathbf{W})$. We perform non-maximum suppression (IOU threshold = 0.4) based on the confidence scores, yielding several candidate bounding boxes $\{\mathbf{b}_j|j \in \mathbf{I}\}$ with confidence scores $P(\mathbf{b}_j)$. Each b_j is then properly zoomed, becoming $N(j)$.

Training the part parsing. The training of the part parsing network is standard. For the object-level FCN, the part parsing network is trained based on all the zoomed image regions (ROIs), with the ground-truth part label maps $\mathcal{H}_p = \{\mathbf{L}_{pi}\}_{i=1}^{n}$ within the zoomed ROIs. For the image-level FCN, the part parsing network is trained based on the original training images. We merge the part parsing network with the SEN, yielding the image-level FCN with loss defined in Equ. (5.3). Here, $l_p(\mathbf{I}, \mathbf{L}_p)$ is the commonly used multinomial logistic regression loss for classification.

$$l_{AZN}(\mathcal{H}, \mathcal{H}_p|\mathbf{W}) = \frac{1}{n}\sum\nolimits_i l_p(\mathbf{I}_i, \mathbf{L}_{pi}) + l_{SEN}(\mathcal{H}|\mathbf{W}) \tag{5.3}$$

Testing the part parsing. For testing the object-scale AZN, we first run the image-level FCN, yielding part score maps at the image level and bounding boxes for the object level. Then we zoom onto the bounding boxes and parse these regions based on the object-level FCN, yielding part score maps at the object level. By merging the part score maps from the two levels, we get better parsing results for the whole image.

5.2 Experimental Evaluation

5.2.1 Implementation Details

Selection of confidence seeds. To train the scale estimation network (SEN), we need to select confidence seeds for object instances or parts. For human instances, we use the human instance masks from the PASCAL-Person-Part dataset and select the central 7×7 pixels within each instance mask as the confidence seeds. To get the confidence seeds for human parts, we first compute connected part segments from the groundtruth part label map, and then also select the central 7×7 pixels within each part segment. We present the details of our approach for humans because the extension to horses and cows is straightforward.

Zooming ratio of ROIs. The SEN networks in the FCNs provide a set of human/part bounding boxes (ROIs), $\{b_j | j \in \mathbf{I}\}$, which are then zoomed to a proper human/part scale. The zooming ratio of b_j, $f(b_j, L_p^{b_j})$, is decided based on the size of b_j and the previously computed part label map $L_p^{b_j}$ within b_j. We use slightly different strategies to compute the zooming ratio at the human and part levels. For the part level, we simply resize the bounding box to a fixed size, i.e. $f_p(b_j) = s_t / max(w_j, h_j)$, where $s_t = 255$ is the target size. Here w_j and h_j are the width and height of b_j. For the human level, we need to consider the frequently occurred truncation case when only the upper half of a human instance is visible. In practice, we use the image-level part label map $L_p^{b_j}$ within the box, and check the existence of legs to decide whether the full body is visible. If the full body is visible, we use the same strategy as parts. Otherwise, we change the target size s_t to 140, yielding relative smaller region than the full body visible case. We select the target size based on a validation set. Finally, we limit all zooming ratio $f_p(b_j)$ within the range $[0.4, 2.5]$ for both human and part bounding boxes to avoid artifacts from up or down sampling of images.

5.2.2 Experimental Protocol

Dataset. We conduct experiments on humans part parsing using the PASCAL-Person-Part dataset annotated by [CML14]. The dataset contains detailed part annotations for every person, e.g. head, torso, etc. We merge the annotations into six clases: Head, Torso, Upper/Lower Arms and Upper/Lower Legs (plus one background class). We only use those images containing humans for training (1716 images in the training set) and testing (1817 images in the validation set), the same as [CYW15]. Note that parsing humans in PASCAL is challenging because it has larger variations in scale and pose than other human parsing datasets. In addtion, we also perform parsing experiments on the horse-cow dataset [WY15], which contains animal instances in a rough bounding box. In this dataset, we adopt the same experimental setting as in [WSL15].

Training. We train the FCNs using stochastic gradient descent with mini-batches. Each mini-batch contains 30 images. The initial learning rate is 0.001 (0.01 for the final classifier layer) and is decreased by a factor of 0.1 after every 2000 iterations. We set the momentum to be 0.9 and the weight decay to be 0.0005. The initialization model is a modified VGG-16 network [SZ14] pre-trained on ImageNet [DDS09]. Fine-tuning our network on all the reported experiments takes about 30 hours on a NVIDIA Tesla K40 GPU. After training, the average inference time for one PASCAL image is 1.3 s/image.

Evaluation metric. The object parsing results is evaluated in terms of mean IOU (mIOU). It is computed as the pixel intersection-over-union (IOU) averaged across classes [EEG14], which is also adopted recently to evaluate parts [WSL15, CYW15]. We also evaluate the part parsing performance w.r.t. each object instance in terms of AP^r_{part} as defined in [HAG15a].

Network architecture. We use DeepLab-LargeFOV [CPK15b] as building blocks for the FCNs in our Hierarchical Auto-Zoom Net (HAZN). Recall that our HAZN consists of three FCNs working at different levels of granularity: image level, object level, and part level. At each level, HAZN outputs part parsing scores, and estimats locations and scales for the next

Method	head	torso	u-arms	l-arms	u-legs	l-legs	bg	Avg
DeepLab-LargeFOV [CPK15b]	78.09	54.02	37.29	36.85	33.73	29.61	92.85	51.78
DeepLab-LargeFOV-CRF	80.13	55.56	36.43	38.72	35.50	30.82	93.52	52.95
Multi-Scale Averaging	79.89	57.40	40.57	41.14	37.66	34.31	93.43	54.91
Multi-Scale Attention [CYW15]	**81.47**	59.06	44.15	42.50	38.28	35.62	93.65	56.39
HAZN (no object scale)	80.25	57.20	42.24	42.02	36.40	31.96	93.42	54.78
HAZN (no part scale)	79.83	59.72	43.84	40.84	40.49	37.23	93.55	56.50
HAZN (full model)	80.76	**60.50**	**45.65**	**43.11**	**41.21**	**37.74**	**93.78**	**57.54**

Table 5.1: Part parsing accuracy (%) on PASCAL-Person-Part in terms of mean IOU. We compare our full model (HAZN) with two sub-models and four state-of-the-art baselines.

level of granularity (e.g. objects or parts).

5.2.3 Results on Parsing Humans in the Wild

Comparison with state-of-the-arts. As shown in Tab. 5.1, we compare our full model (HAZN) with four baselines. The first baseline is called DeepLab-LargeFOV [CPK15b]. The second baseline is DeepLab-LargeFOV-CRF, which adds a post-processing step to DeepLab-LargeFOV by means of a fully-connected Conditional Random Field (CRF) [KK11]. CRFs are commonly used as postprocessing for object semantic segmentation to refine boundaries [CPK15b]. The third one is Multi-Scale Averaging, which feeds the DeepLab-LargeFOV model with images resized to three fixed scales (0.5, 1.0 and 1.5) and then takes the average of the three part score maps to produce the final parsing result. The fourth one is Multi-Scale Attention [CYW15], a most recent work which uses a scale attention model to handle the scale variations in object parsing.

Our HAZN obtains the performance of 57.5%, which is 5.8% better than DeepLab-LargeFOV, and 4.5% better than DeepLab-LargeFOV-CRF. Our model significantly improves the segmentation accuracy in all parts. Note we do not use any CRF for post processing. The CRF, though proven effective in refining boundaries in object segmentation, is

not strong enough at recovering details of human parts as well as correcting the errors made by the DeepLab-LargeFOV.

The third baseline (Multi-Scale Averaging) enumerates multi-scale features which is commonly used to handle the scale variations, yet its performance is poorer than ours, indicating the effectiveness of our Auto-Zoom framework.

Our overall mIOU is 1.15% better than the fourth baseline (Multi-Scale Attention), but we are much better in terms of detailed parts like upper legs (around 3% improvement). In addition, we further analyze the scale-invariant ability in Tab. 5.2, which both methods aim to improve. We can see that our model surpasses Multi-Scale Attention in all instance sizes especially at size XS (9.5%) and size S (5.5%).

Importance of object and part scale. As shown in Tab. 5.1, we study the effect of the two scales in our HAZN. In practice, we remove either the object-scale AZN or the part-scale AZN from the full HAZN model, yielding two sub-models: (1) **HAZN (no object scale)**, which only handles the scale variation at part level. (2)**HAZN (no part scale)**, which only handles the scale variation at object instance level.

Compared with our full model, removing the object-scale AZN causes 2.8% mIOU degradation while removing the part-scale AZN results in 1% mIOU degradation. We can see that the object-scale AZN, which handles the scale variation at object instance level, contributes a lot to our final parsing performance. For the part-scale AZN, it further improves the parsing by refining the detailed part predictions, e.g. around 3% improvement of lower arms as shown in Tab. 5.1, yielding visually more satisfactory results. This demonstrates the effectiveness of the two scales in our HAZN model.

Part parsing accuracy w.r.t. size of human instance. Since we handle human with various sizes, it is important to check how our model performs with respect to the change of human size in images. In our experiments, we categorize all the ground truth human instances into four different sizes according to the bounding box area of each instance s_b (the square root of the bounding box area). Then we compute the mean IOU (within the

bounding box) for each of these four scales.

The four sizes are defined as follows: (1) Size XS: $s_b \in [0, 80]$, where the human instance is extremely small in the image; (2) Size S: $s_b \in [80, 140]$; (3) Size M: $s_b \in [140, 220]$; (4) Size L: $s_b \in [220, 520]$, which usually corresponds to truncated human instances where the human's head or torso covers the majority of the image.

The results are given in Tab. 5.2. The baseline DeepLab-LargeFOV performs badly at size XS or S (usually only the head or the torso can be detected by the baseline), while our HAZN (full model) improves over it significantly by 14.6% for size XS and 10.8% for size S. This shows that HAZN is particularly good for small objects, where the parsing is difficult to obtain. For instances in size M and L, our model also significantly improve the baselines by around 5%. In general, by using HAZN, we achieve much better scale invariant property to object size than a generally used FCN type of model. We also list the results for the other three baselines for reference.

In addition, it is also important to jointly perform the two scale AZNs in a sequence. To show this, we additionally list the results from our model without object/part scale AZN in the 5_{th} and the 6_{th} row respectively. By jumping over object scale (HAZN no object scale), the performance becomes significantly worse at size XS, since the model can barely detect the object parts at the image-level when the object is too small. However, if we remove part scale (HAZN no part scale), the performance also dropped in all sizes. This is because using part scale AZN can recover the part details much better than only using object scale. Our HAZN (full model), which sequentially leverage the benefits from both the object scale and part scale, yielding the best performance overall.

Instance-wise part parsing accuracy. We evaluate our part segmentation results w.r.t. each human instance in terms of AP^r_{part} as defined in [HAG15a]. The segment IOU threshold is set to 0.5. A human instance segment is correct only when it overlaps enough with a groundtruth instance segment. To compute the intersection of two segments, we only consider the pixels whose part labels are also right.

To generate instance segmentation (which is not our major task), we follow a similar

Method	Size XS	Size S	Size M	Size L
DeepLab-LargeFOV [CPK15b]	32.5	44.5	50.7	50.9
DeepLab-LargeFOV-CRF	31.5	44.6	51.5	52.5
Multi-Scale Averaging	33.7	45.9	52.5	54.7
Multi-Scale Attention [CYW15]	37.6	49.8	55.1	55.5
HAZN (no object scale)	38.2	51.0	55.1	53.4
HAZN (no part scale)	45.1	53.1	55.0	55.0
HAZN (full model)	**47.1**	**55.3**	**56.8**	**56.0**

Table 5.2: Part parsing accuracy w.r.t. size of human instance (%) on PASCAL-Person-Part in terms of mean IOU.

strategy to [HAG15a] by first generating object detection box and then doing instance segmentation. Specifically, we use faster R-CNN [RHG15] to produce a set of object bounding box proposals, and each box is associated with a confidence score. Then within each bounding box, we use FCN to predict a coarse object instance mask, and use the coarse instance mask to retrieve corresponding part segments from our final HAZN part label map. Last, we use the retrieved part segments to compose a new instance mask where we keep the boundary of part segments. In the instance overlapping cases, we follow the boundary from the predicted instance mask.

We first directly compare with the number reported by [HAG15a], on the whole validation set of PASCAL 2010. Our full HAZN achieves **43.08%** in AP^r_{part}, **14%** higher than [HAG15a]. We also compare with two state-of-the-art baselines on the PASCAL-Person-Part dataset: DeepLab-LargeFOV [CPK15b] and Multi-Scale Attention [CYW15]. For both baselines, we applied the same strategy to generate instances but used different part parsing strategies. As shown in Tab. 5.3, our model is 12% points higher than DeepLab-LargeFOV and 6% points higher than Multi-Scale Attention, in terms of AP^r_{part}.

Qualitative results. We visually show several example results from the PASCAL-Person-Part dataset in Fig. 5.5. The baseline DeepLab-LargeFOV-CRF produces several errors due to lack of object and part scale information, e.g. background confusion (1_{st} row), human part confusion (3_{rd} row), or important part missing (4_{th} row), etc., yielding non-satisfactory part segmentation results. Our HAZN (no part scale), which only contains object-scale AZN, already successfully relieves the confusions for large scale human instances while recovers the parts for small scale human instances. By further introducing part scale, the part details and boundaries are recovered even better, which are more visually satisfactory.

More visual examples are provided in Fig. 5.7, comparing with more baselines. It can be seen that our full model (HAZN) gives much more satisfied part parsing results than the state-of-the-art baselines. Specifically, for small-scale human instances (e.g. the 1, 2, 5 rows of the figure), our HAZN recovers human parts like lower arms and lower legs and gives more accurate part boundaries; for medium-scale or large-scale human instances (e.g. the 3, 4, 9, 10 rows of the figure), our model relieves the local confusion with other parts or with the background.

Failure cases. Our typical failure modes are shown in Fig. 5.6. Compared with the baseline DeepLab-LargeFOV-CRF, our models give more reasonable parsing results with less local confusion, but they still suffer from heavy occlusion and unusual poses.

5.2.4 Results on Parsing Animals

To show the generality of our method to instance-wise object part parsing, we also applied our method to horse instances and cow instances presented in [WY15]. All the testing procedures are the same as those described above for humans.

	DeepLab-LargeFOV [CPK15b]	Multi-Scale Attention [CYW15]	HAZN(full model)
AP^r_{part}	31.32	37.53	43.72

Table 5.3: Instance-wise part parsing accuracy on PASCAL-Person-Part in terms of AP^r_{part}.

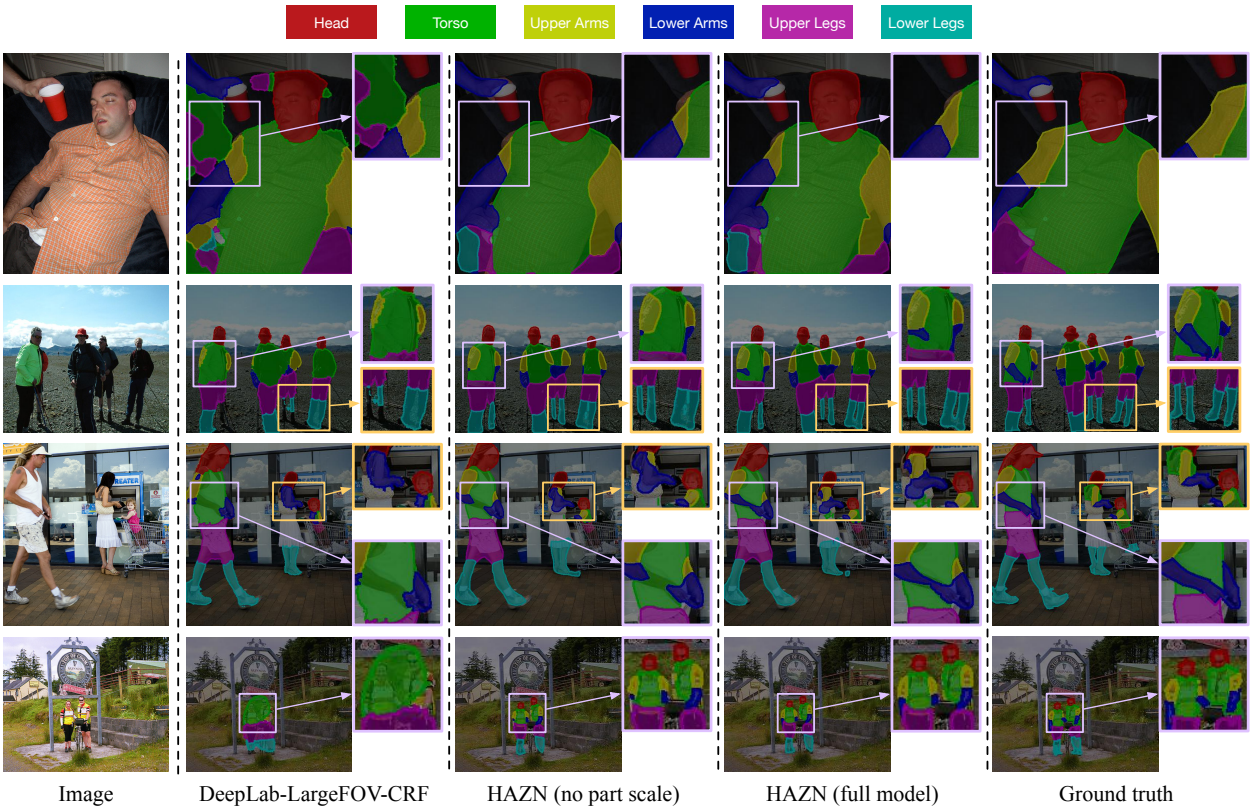


Figure 5.5: Qualitative comparison on the PASCAL-Person-Part dataset. We compare with DeepLab-LargeFOV-CRF [CPK15b] and HAZN (no part scale). Our proposed HAZN models (the 3_{rd} and 4_{th} columns) attain better visual parsing results, especially for small scale human instances and small parts such as legs and arms.

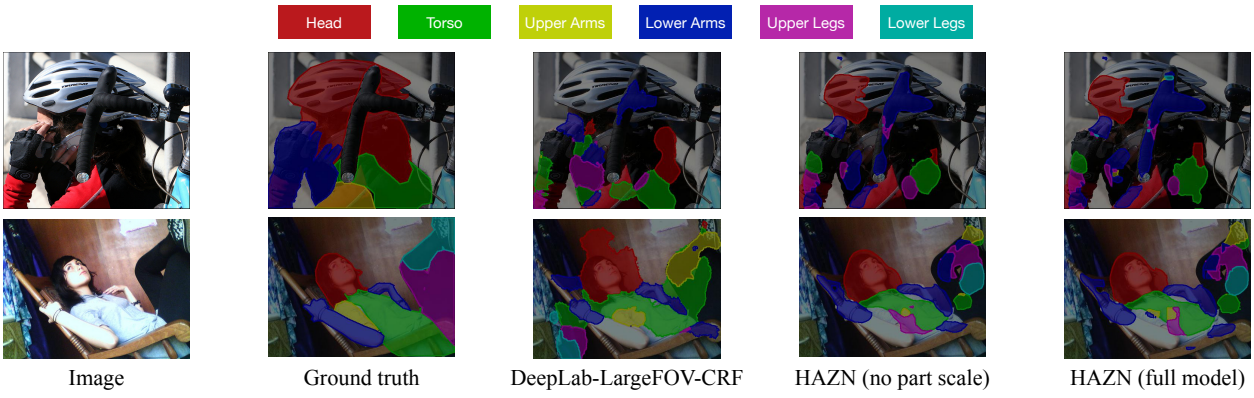


Figure 5.6: Failure cases for both the baseline and our models.

We copy the baseline numbers from [WSL15], and give the evaluation results in Tab. 5.4. It shows that our baseline models from the DeepLab-LargeFOV [CPK15b] already achieve

competative results with the state-of-the-arts, while our HAZN provides a big improvement for horses and cows. The improvement over the state-of-the-art method [WSL15] is roughly 5% mIOU. It is most noticeable for small parts, e.g. the improvement for detecting horse/cow head and cow tails is more than 10%. This shows that our auto-zoom strategy can be effectively generalized to other objects for part parsing.

We also provide qualitative evaluations in Fig. 5.8, comparing our full model with three state-of-the-art baselines. The three baselines are explained in Sec. 5.2.3. We can observe that using our model, small parts such as legs and tails have been effectively recovered, and the boundary accuracy of all parts has been improved.

	Horse						Cow					
Method	Bkg	head	body	leg	tail	Avg.	Bkg	head	body	leg	tail	Avg.
SPS [WY15]	79.14	47.64	69.74	38.85	-	-	78.00	40.55	61.65	36.32	-	-
HC* [HAG15a]	85.71	57.30	77.88	51.93	37.10	61.98	81.86	55.18	72.75	42.03	11.04	52.57
JPO [WSL15]	87.34	60.02	77.52	58.35	**51.88**	67.02	85.68	58.04	76.04	51.12	15.00	57.18
LargeFOV	87.44	64.45	80.70	54.61	44.03	66.25	86.56	62.76	78.42	48.83	19.97	59.31
HAZN	**90.94**	**70.75**	**84.49**	**63.91**	51.73	**72.36**	**90.71**	**75.18**	**83.33**	**57.42**	**29.37**	**67.20**

Table 5.4: Mean IOU (mIOU) over the Horse-Cow dataset. We compare with the semantic part segmentation (SPS) [WY15], the Hypercolumn (HC*) [HAG15a] and the joint part and object (JPO) results [WSL15]. We also list the performance of DeepLab-LargeFOV (LargeFOV) [CPK15b].

5.3 Conclusion

To handle the big scale variation in natural images, we explain in this chapter an object part segmentation model called "Hierarachical Auto-Zoom Net" (HAZN), yielding per-pixel segmentation of the object parts. It adaptably estimates the scales of objects, and their parts, by a two-stage process of Auto-Zoom Nets. We show that on the challenging PASCAL dataset, HAZN performs significantly better (by 5% mIOU), compared to state of the art

methods, when applied to humans, horses, and cows. Unlike standard methods which process the image at a fixed range of scales, HAZN's strategy of searching for objects and then for parts enables it, for example, to zoom in to small image regions and enlarge them to scales which would be prohibitively expensive (in terms of memory) if applied to the entire image (as fixed scale methods would require).

The "auto-zoom" idea of HAZN can be easily applied to other tasks, e.g. human pose estimation, fine-grained part localization, and so on. In Chapter 6, we use the "auto-zoom" idea to significantly improve the performance of multi-person human pose estimation in natural images.

Looking at the failure cases of HAZN (see Fig. 5.6), we notice that HAZN can still make errors about the overall pose configuration (e.g. regarding arms as legs, regarding a second person's legs as the first person's, etc.) when the person is in an extreme pose, or the appearance of different parts are very similar, or there are other people overlapping with this person. This motivates us to combine useful top-down pose information with the deep-based segmentation model HAZN, which we also explore in Chapter 6.

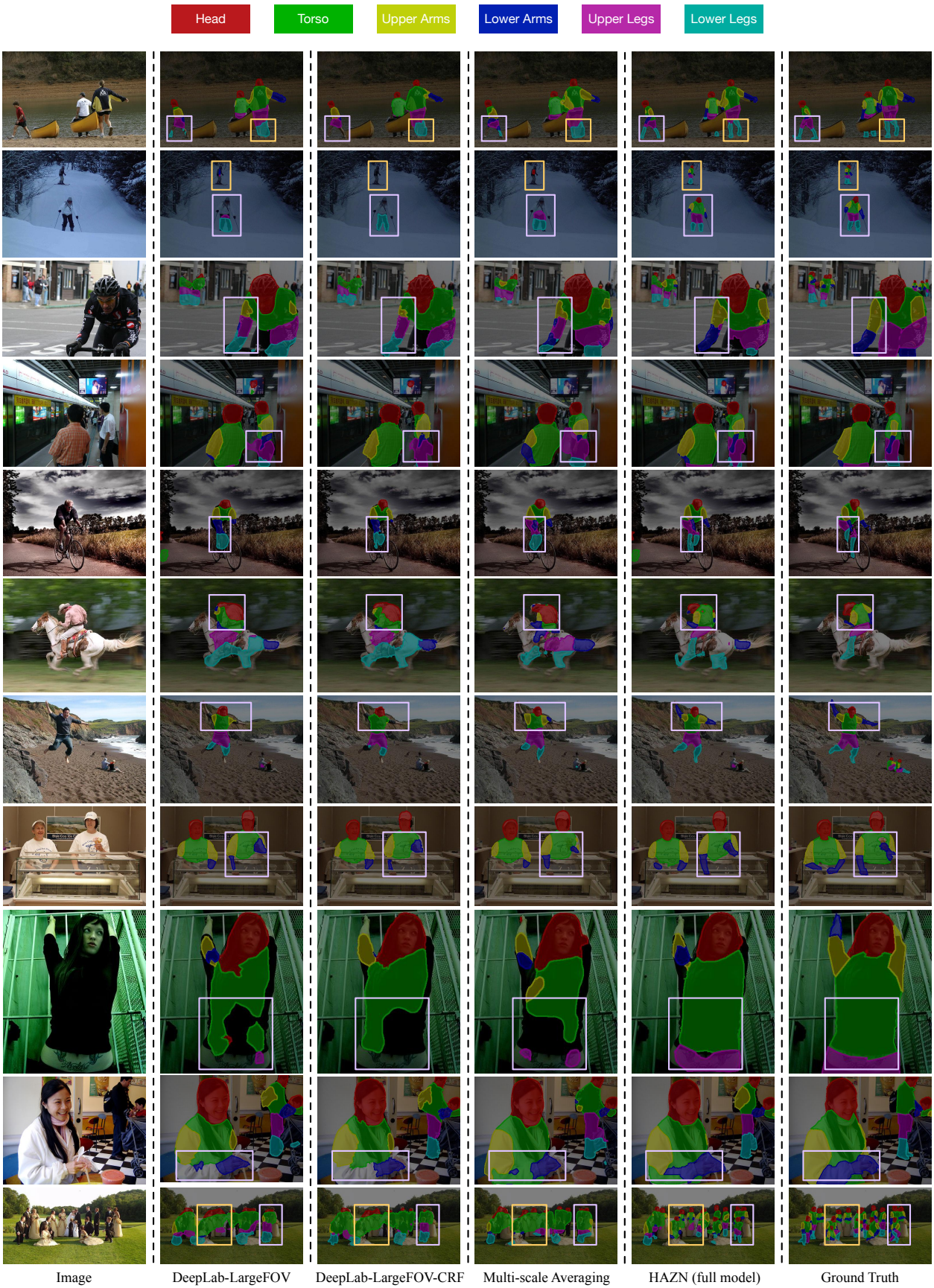


Figure 5.7: More qualitative comparison on PASCAL-Person-Part. The baselines are explained in Sec. 5.2.3.

Figure 5.8: Qualitative comparison on the Horse-Cow Dataset. The baselines are explained in Sec. 5.2.3.

CHAPTER 6

Combining Human Pose Estimation and Semantic Part Segmentation in Multi-Person Natural Scenes

Human pose estimation [YR11, BM09, CY15] and human semantic part segmentation [YLO12, DCX13, CPK15b, XZW16] are two crucial and correlated tasks in human-centric analysis. They are helpful in many high-level applications, spanning from fine-grained recognition and tracking, to video surveillance and human activity recognition. Dramatic progress has been made on both tasks with the advent of powerful deep convolutional neural networks (DCNNs) and the availability of pose/segment annotations on large-scale datasets. However, these methods are still limited in natural scenes with multiple people, where multi-person overlapping/occlusion, extreme human poses and large object scale variation exist. Each task faces some challenging cases if solved individually, e.g. inaccurate localization or erroneous visibility prediction of the knee joints in pose estimation when the person is wearing a long dress, failure to distinguish ambiguous regions in semantic part segmentation when the person is in non-typical pose, etc. These difficult cases can still be handled effectively if we consider the correlation of the two tasks and let the two tasks benefit each other (see Fig. 6.1).

In Chapter 4, we have already proved the effectiveness of pose information in semantic part segmentation by incorporating top-down pose cues into an AOG-based framework. We only use deep learning to extract features for part region proposals, and rely on SVM and AOG to select and assemble part proposals. This works in relatively simple images like Penn-Fudan Pedestrian [WSS07], but would fail in multi-person natural images like PASCAL [EEG14]. In Chapter 5, we rely mainly on deep-based models (i.e. variants of FCN [CPK15b] with the powerful architecture VGG-16 [SZ14]) and find that they perform

quite well on the challenging dataset PASCAL. However, some severe pose errors cannot be avoided since the deep models doesn't model the dependency between the output variables, and also the boundary details of arms and legs are far from satisfactory. This motivates us to find a way to incorporate useful pose cues into the deep-based segmentation framework, giving guidance of top-down configurations.

With the popularity of deep learning techniques, current prevailing approaches for human pose estimation rely on strong joint detectors trained by DCNNs [CY14, TGJ15], and use a simple graphical model to select and assemble joints into a valid pose configuration. These methods perform much better than traditional ones, but the localization of joints is still inaccurate (e.g. sometimes outside the human body) and these methods still struggle in cases where there are multiple people overlapping each other. A very recent work, Deeper-Cut [IPA16], explicitly considers the multi-person issue, using a fully-connected CRF to simultaneously cluster joint candiates into human instances and label each joint candidate its joint type. Deeper-Cut handles multi-person overlapping well, but it's very time-consuming (4 minutes per image) and its performance on datasets with large scale variation are not satisfactory. Treating Deeper-Cut as a good starting point, we propose to: (1) model consistency with semantic part segmentation results by incorporating simple segment-based unary and pairwise terms into the fully connected CRF, penalizing joint candidates and joint candidate pairs whose locations doesn't agree with their corresponding semantic parts; (2) employ the idea of "auto-zoom" to first detect human bounding boxes and then perform multi-person pose estimation within each properly resized bounding boxes. With these two strategies, our model gives better results than Deeper-Cut in multi-person scenes, and reduces the running time to only 8 seconds per image (including the time of computing all necessary features).

For semantic part segmentation, previous approaches either use graphical models to select and assemble region proposals [DCS14, XZW16], or use fully convolutional neural networks [LSD15] (FCNs) to directly produce pixel-wise part labels. Traditional graphical models are often complicated and time-consuming, and can't handle the large variability of pose and occlusion in natural images. FCN-type approaches, though simple and fast, still make local confusion errors when the person is in a non-typical pose, or when there

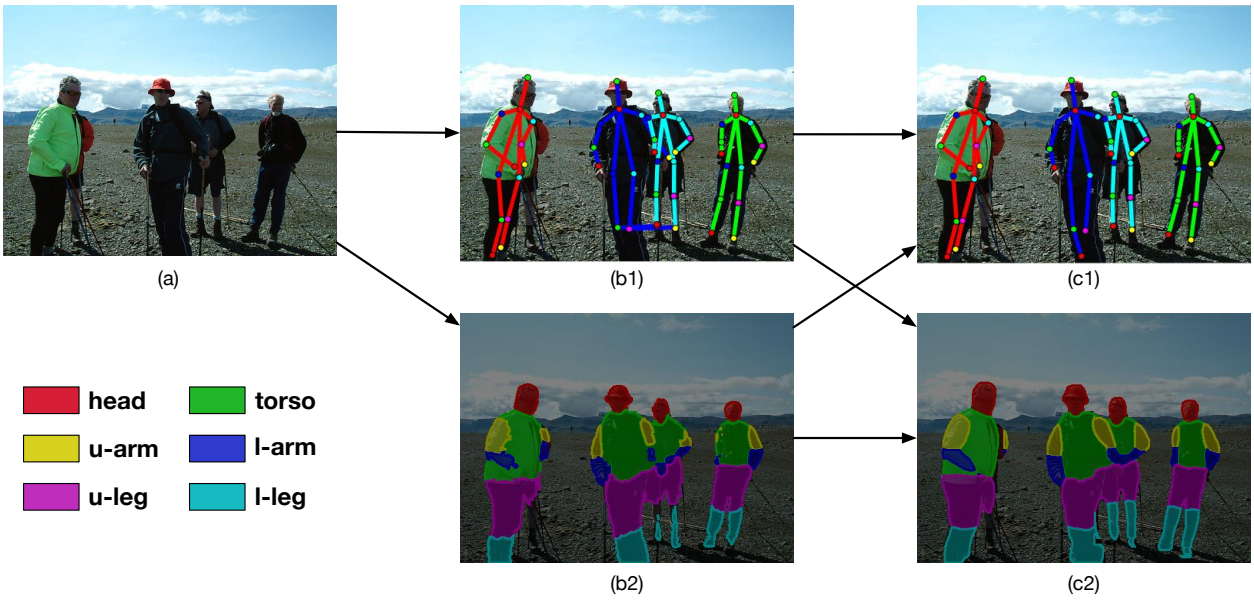


Figure 6.1: Human pose estimation and semantic part segmentation are two complementary tasks and can benefit each other. Column (a): the original image. Column (b): original pose estimation and semantic part segmentation. Column (c): our final pose estimation result using semantic part information, and our final part segmentation result using pose information. (c1) corrects the location error of ankle joints in (b1) for the person in the middle; (c2) gives much clearer details of lower arms and legs than (b2) for the two people in the middle.

are some other object/person nearby with similar appearance. In this chapter, we propose to use top-down pose cues so as to relieve local confusion in semantic part segmentation. Specifically, we infer feature maps that capture joints and skeleton information from a good pose estimate, and feed these feature maps together with the original image to a dedicated FCN to refine the part segmentation results.

We perform experiments on Pascal-Person-Part [CML14], a challenging multi-person dataset with detailed annotations of pose joints and semantic parts. We compare with other state-of-the-arts on both tasks, demonstrating the effectiveness of our approach.

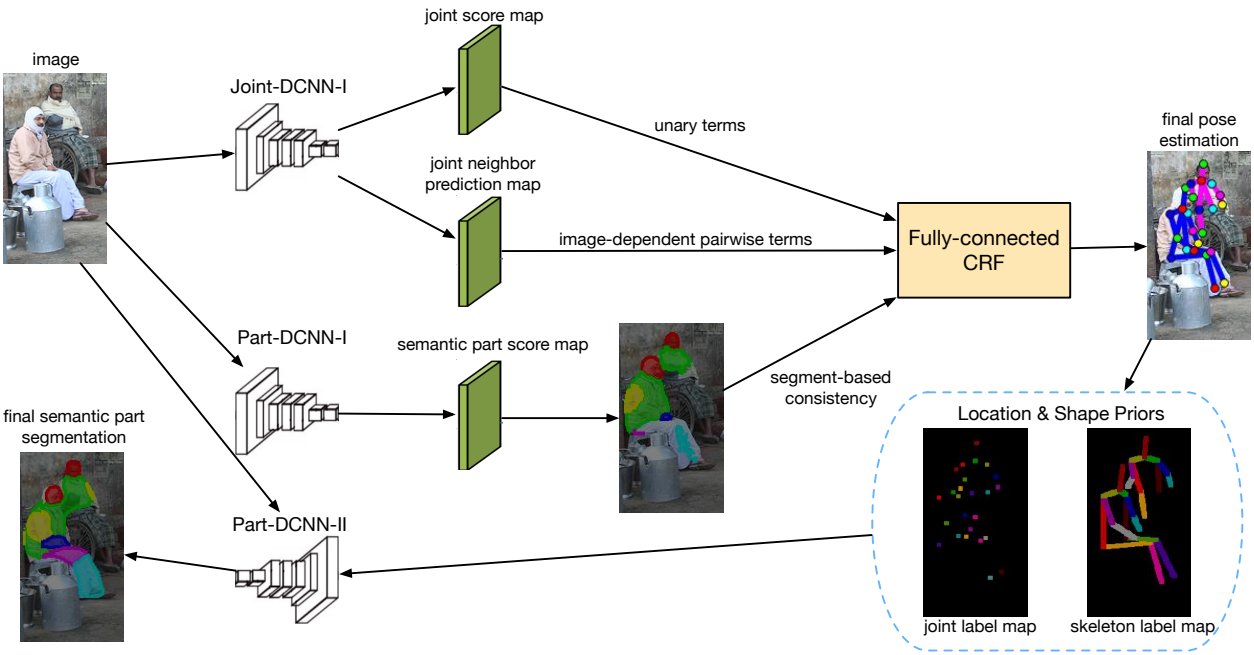


Figure 6.2: Overall model pipeline.

6.1 The Model

Given an image I, our task is to output a pixel-wise part label map L^{part} which provides the semantic part segmentation, and also a list of scored pose configurations $C^{pose} = \{(c_i, s_i)|i = 1, 2, \dots\}$, where c_i is the location of pose joints for a single person and s_i is the score of this pose configuration.

Fig. 6.2 illustrates the overall pipeline of our model, in which the main components are three DCNNs (i.e. Joint-DCNN, Part-DCNN-I and Part-DCNN-II) as well as a fully-connected conditional random field (CRF). The three DCNNs employ roughly the same network structure (i.e. ResNet-101, one type of Deep Residual Net [HZR15]), but for different purposes. Given an image I, we first pass I to Joint-DCNN to get two feature maps: pixel-wise joint score map and pixel-wise joint neighbor prediction map. Proposals for each joint type are generated from the joint score map, and traditional unary terms (based on joint score map) and pairwise terms (based on joint neighbor prediction map) are computed for the fully-connnected CRF. We also pass I to Part-DCNN-I to get a semantic part score map

in relatively good quality. Based on the semantic part score map, we compute segment-based consistency terms for joint proposals, which are also used in the CRF. The CRF assembles joint proposals into a series of pose configurations, partitioning and labeling joint proposals at the same time. These pose configurations are our final result for the human pose estimation task. We further use these pose configurations to refine semantic part segmentation. Specifically, we generate intuitive joint label map and skeleton label map from these pose configurations, and pass them to Part-DCNN-II, which utilize them as location and shape priors to refine the semantic part score map. The output of Part-DCNN-II is the final result for the semantic part segmentation task.

In our experiments on PASCAL-Person-Part, we inherit some ideas from the Auto-Zoom Net [XWC16] to deal with the large object scale variation in PASCAL images and to lessen the inference time of CRF. For an image I, human detection boxes of I are produced by Faster R-CNN [RHG15]. Each detection box is properly zoomed (e.g. small person will be enlarged and extra large person will be shrinked) and serves as the input to our pipeline in Fig. 5.2. Results of human pose estimation and semantic part segmentation for all detection boxes are then merged as the final results for the whole image I.

6.1.1 Human Pose Estimation Model

For pose estimation, we follow one line of works that first generate joint location proposals for each joint type and then select and assemble those joint proposals into valid pose configurations. Thanks to the recent powerful DCNN structures such as Deep Residual Net [HZR15] and DeepLab [CPK15b], using only a few proposals per joint type still guarantees state-of-the-art performance. Specifically, for each detection box of an image, we generate only 6 proposals per joint type by performing thresholding (score threshold = 0.2) and non-maximum suppresion (NMS, nms distance threshold = 16) on the joint score map outputted by Joint-DCNN-I.

Suppose we generate n proposals in total, with image location $g_1, g_2, \ldots, g_n$. To select and assemble these joint proposals, we use a fully-connected CRF that aims to infer: (1)

the joint type l_{g_i} for each joint proposal g_i (background is also a type); (2) a binary variable l_{g_i,g_j} for each proposal pair (g_i, g_j) indicating whether g_i and g_j belong to the same person. The CRF treats each joint proposal as a node ($\mathcal{V} = \{g_1, g_2, \ldots, g_n\}$), and defines an edge between each pair of the joint proposals ($\mathcal{E} = \{(g_i, g_j)|i = 1, 2, \ldots, n, j = 1, 2, \ldots, n, i < j\}$). Let $L = \{l_{g_i}|g_i \in \mathcal{V}\} \cup \{l_{g_i,g_j}|(g_i, g_j) \in \mathcal{E}\}$. Given image I and its semantic part label map L^{part}, the energy function of the CRF is defined as Equ. (6.1), where $A(I|l_{g_i})$ and $B(I, L^{part}|l_{g_i}, l_{g_j}, l_{g_i,g_j})$ are the unary potential and pairwise potential respectively.

$$\phi(L|I, L^{part}) = \sum_{g_i \in \mathcal{V}} A(I|l_{g_i}) + \sum_{(g_i,g_j) \in \mathcal{E}} B(I, L^{part}|l_{g_i}, l_{g_j}, l_{g_i,g_j}) \tag{6.1}$$

Now we explain how we define and compute the potentials and how the part segment consistency is incorporated. As shown in Equ. (6.2), the unary potential $A(I|l_{g_i})$ relies solely on $P(l_{g_i}|I)$, the probability of pixel location g_i being of the joint type l_{g_i}, which can be directly acquired from the pixel-wise joint score map outputted by Joint-DCNN-I.

$$A(I|l_{g_i}) = log\frac{1 - P(l_{g_i}|I)}{P(l_{g_i}|I)} \tag{6.2}$$

The pairwise potential $B(I, L^{part}|l_{g_i}, l_{g_j}, l_{g_i,g_j})$, as shown in Equ. (6.3), is based on $P(l_{g_i}, l_{g_j}|I, L^{part})$, which is the output of a logistic regression model for joint type pair l_{g_i} and l_{g_j}. For each joint type pair l_{g_i} and l_{g_j}, the logistic regression model (see Equ. (6.4)) is separately trained on the training set, treating ground-truth joint pair of type l_{g_i} and l_{g_j} (of the same person) as positive examples while treating all other pixel pairs as negative examples, using feature vector $\boldsymbol{f}(g_i, g_j, l_{g_i}, l_{g_j}|I, L^{part})$.

$$B(I, L^{part}|l_{g_i}, l_{g_j}, l_{g_i,g_j}) = \begin{cases} log\frac{1-P(l_{g_i},l_{g_j}|I,L^{part})}{P(l_{g_i},l_{g_j}|I,L^{part})} & (l_{g_i,g_j} = 1) \\ 0 & (l_{g_i,g_j} = 0) \end{cases} \tag{6.3}$$

$$P(l_{g_i}, l_{g_j}|I, L^{part}) = \frac{1}{1 + exp(-\boldsymbol{\omega} \cdot \boldsymbol{f}(g_i, g_j, l_{g_i}, l_{g_j}|I, L^{part}))} \tag{6.4}$$

Given two joint proposals g_i and g_j as well as their joint type labels l_{g_i} and l_{g_j}, the feature vector $\boldsymbol{f}$ encodes information to help decide whether the two proposals belong to the same person. $\boldsymbol{f}$ includes pairwise image-dependent features $\boldsymbol{f_{data}}$ proposed by [IPA16],

which are computed from the joint neighbor prediction map output by the Joint-DCNN. More specifically, given that g_i belongs to type l_{g_i} and g_j belongs to type l_{g_j}, the joint neighbor prediction map predicts the location of g_j (denoted as $\tilde{g_j}$) from the location of g_i and predicts the location of g_i (denoted as $\tilde{g_i}$) from the location of g_j. $\boldsymbol{f_{data}}$ describes the distance and angle between two pairs of vectors ($\boldsymbol{g_i g_j}$ vs. $\boldsymbol{g_i \tilde{g_j}}$, and $\boldsymbol{g_j g_i}$ vs. $\boldsymbol{g_j \tilde{g_i}}$): $\boldsymbol{f_{data}} = [$ $|\boldsymbol{g_i g_j} - \boldsymbol{g_i \tilde{g_j}}|$, $|\boldsymbol{g_j g_i} - \boldsymbol{g_j \tilde{g_i}}|$, $< \boldsymbol{g_i g_j}, \boldsymbol{g_i \tilde{g_j}} >$, $< \boldsymbol{g_j g_i}, \boldsymbol{g_j \tilde{g_i}} >]$, in which $|. - .|$ is the euclidean distance between two vectors and $< ., . >$ is the angle between two vectors.

In addition to $\boldsymbol{f_{data}}$, $\boldsymbol{f}$ also includes our novel consistency features $\boldsymbol{f_{seg}}$ based on the semantic part segmentation L^{part}. $\boldsymbol{f_{seg}}$ describes the location consistency between $\boldsymbol{g_i g_j}$ and their corresponding semantic parts. In our design, each joint type is associated with one or two semantic part types and each neighbouring joint type pair is associated with one semantic part type. Given L^{part}, suppose l_{g_i}= forehead and l_{g_j}= neck, then g_i is related to the head region, g_j is related to the head and the torso regions, and $\boldsymbol{g_i g_j}$ is related to the head region. In this case, the segment-based consistency features $\boldsymbol{f_{seg}}$ are made up of: (1) a 2-d binary feature indicating the location of g_i w.r.t. the head region: inside the region or not, around the boundary of the region or not; (2) a 4-d binary feature indicating the location of g_j w.r.t. the head region and the torso region respectively; (3) the proportion of pixels on the line $\boldsymbol{g_i g_j}$ that fall inside the head region; (4) the IOU overlap between an oriented rectangle computed from $\boldsymbol{g_i g_j}$ (aspect ratio = 2.5:1) and the head region.

Based on the unary and pairwise potentials explained above, the CRF infers the best labels L for the generated joint proposals $g_1, g_2, \ldots, g_n$, selecting and assembling them into a list of pose configurations. We adopt the inference algorithm introduced in [IPA16], transforming the CRF into an integer linear programming (ILP) problem with additional constraints on L: $l_{g_i,g_j} + l_{g_j,g_k} \leqslant l_{g_i,g_k} + 1$, $\forall i, j = 1, 2, \ldots, n$.

For each detection box of I, the inference algorithm gives the labels L for joint proposals within 1 sec. and we can acquire a list of pose configurations based on L, with pose score equal to the sum of unary scores for all visible joints. For each detection box, we choose only one pose configuration whose center is closest to the detection box center, and add that pose configuration to our final pose estimation result C^{pose}.

6.1.2 Semantic Part Segmentation Model

We train a part segmentation model Part-DCNN-II to segment an image into semantic parts with estimated high-quality pose configurations C^{pose}. We define two pose feature maps from C^{pose}, a joint label map and a skeleton label map, and use them as inputs to Part-DCNN-II besides the original image. For the joint label map, we draw a circle with radius 3 at each joint location in C^{pose}. For the skeleton label map, we draw a stick with width 7 between neighbouring joints in C^{pose}. Fig. 6.2 illustrates the two simple and intuitive feature maps.

Part-DCNN-II is different from Part-DCNN-I in that it has a new stream that uses two pose feature maps to estimate semantic part score map and this part score map is later fused with the original part score map estimated from only the image. We use extra supervision to train the part score map estimated from pose feature maps. To handle the big object scale variation, we use the idea of "auto-zoom" [XWC16]. For each detection box of an image I, we crop the box out of I and the two feature maps, properly resize them, and feed them to Part-DCNN-II, getting the part score map for the detection box. The part score maps for all detection boxes of I are then resized to their original size and merged according to the detection box score.

6.2 Experimental Evaluation

We perform extensive experiments on PASCAL-Person-Part [CML14], which are PASCAL person images with large variation in pose and scale. There are 14 annotated joint types (i.e. forehead, neck, l/r shoulder, l/r elbow, l/r wrist, l/r waist, l/r knee and l/r ankle) and we combine part labels into 6 semantic part types (i.e. head, torso, upper arm, lower arm, upper leg and lower leg). We only use those images containing persons for training (1716 images) and validation (1817 images).

6.2.1 Human Pose Estimation

Comparison to state-ot-the-arts. Some previous evaluation metrics like PCK and PCP, don't penalize false positives that are not part of the groundtruth. So following [IPA16], we compare our model with other state-of-the-arts in terms of Mean Average Precision (mAP). The computation is the same as described in [IPA16]. Briefly speaking, pose configurations in C^{pose} are first matched to groundtruth pose configurations according to overlap, and then AP for each joint type is computed and reported. Each groundtruth can only be matched to one estimated pose configuration. Unassigned pose configurations in C^{pose} are all treated as false positives.

Our model is compared with two other state-of-the-arts: (1) Chen & Yuille [CY15], a tree-structured model designed specifically for single-person estimation with occlusion, using unary scores and image-dependent pairwise terms conditioned on DCNN features; (2) Deeper-Cut [IPA16], an integer linear programming model that jointly performs multi-person detection and multi-person pose estimation. These two methods both use strong assembling model.

We also build two other baselines, which use simple assembling model. One is AOG-Simple, which only uses geometric connectivity between neighbouring joints. The other one is AOG-Seg, which adds part segment consistency features to AOG-Simple. The part segment consistency features are the same as those used in our fully-connected CRF. To test the effectiveness of our proposed part segment consistency, we also list the result of our model without the consistency features ("Our Model (w/o seg)").

The results are shown in Tab. 6.1. We can see that our model outperforms all the other methods, improving over arm joints and leg joints significantly, and that a good assembling model is indispensable in challenging multi-person scenes like PASCAL.

Detailed joint localization accuracy. Our proposed part segment consistency features not only help in the overall pose estimation results, but help improve the accuracy of detailed joint localization. Previous evaluation metrics PCP, PCK and mAP treat any joint

Method	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	U-Body	Total (mAP)
Chen & Yuille [CY15]	45.3	34.6	24.8	21.7	9.8	8.6	7.7	31.6	21.8
Deeper-Cut [IPA16]	41.5	39.3	34.0	27.5	16.3	21.3	20.6	35.5	28.6
AOG-Simple	56.8	29.6	14.9	11.9	6.6	7.3	8.6	28.3	19.4
AOG-Seg	**58.5**	33.7	17.6	13.4	7.3	8.3	9.2	30.8	21.2
Our Model (w/o seg)	56.8	52.1	42.7	36.7	21.9	30.5	30.4	47.1	38.7
Our Model (final)	58.0	**52.1**	**43.1**	**37.2**	**22.1**	**30.8**	**31.1**	**47.6**	**39.2**

Table 6.1: Mean Average Precision (mAP) of Human Pose Estimation on PASCAL-Person-Part.

estimate within a certain distance of the groundtruth to be correct, but they don't encourage joint estimates to be as close as possible to the groundtruth. Therefore, we design a new evaluation metric called Average Distance of Keypoints (ADK). For each groundtruth pose configuration, we compute its reference scale as half of the distance between forehead and neck, then find only one pose configuration estimate among the generated pose configuration proposals that has the highest overlap with the groundtruth configuration. For each joint that is visible in both the groundtruth configuration and the estimated configuration, the relative distance (w.r.t. the reference scale) between the estimated location and the groundtruth location is computed. Finally, we compute the average distance for each joint type across all the testing images.

The result is shown in Tab. 6.2. It can be seen that our model reduces the average distance of keypoints significantly for wrists and lower-body joints by employing consistency with semantic part segmentation.

Qualitative evaluation. In Fig. 6.3, we visually demonstrate our pose estimation results on PASCAL-Person-Part, comparing with the recent state-of-the-art (Deeper-Cut [IPA16]) and also a sub-model of ours (Our Model (w/o seg)) which doesn't consider part segment consistency. It can be easily seen that our model gives more accurate prediction of joints on heads, arms and legs, and is especially better at handling people of small scale (see the 6_{th}

Method	Forehead	Neck	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Ave.
Chen & Yuille [CY15]	37.5	29.7	51.6	65.9	72.0	70.5	79.9	78.6	60.7
Deeper-Cut [IPA16]	32.1	30.9	37.5	44.6	53.5	53.9	65.8	67.8	48.3
AOG-Simple	33.0	33.2	66.7	82.3	90.5	89.7	101.3	101.1	74.7
AOG-Seg	32.2	31.6	59.8	72.4	85.1	85.7	97.1	92.7	69.6
Our Model (w/o seg)	27.7	26.9	33.1	40.2	47.3	51.8	54.6	53.4	41.9
Our Model (final)	**26.9**	**26.1**	**32.7**	**39.5**	**45.3**	**50.9**	**52.3**	**51.8**	**40.7**

Table 6.2: Average Distance of Keypoints (ADK) (%) of Human Pose Estimation on PASCAL-Person-Part.

and 7_{th} row of Fig. 6.3) and extra large scale (see the first two rows of Fig. 6.3). We attribute this to our "auto-zoom" strategy and the incorporation of part segment consistency.

6.2.2 Human Semantic Part Segmentation

Comparison to the state-of-the-arts. We evaluate part segmentation in terms of mean pixel IOU (mIOU) following previous works [CPK15b, XWC16]. In Tab. 6.3, we compare our model with three other state-of-the-art methods as well as one inferior baseline (i.e. the output part label map of Part-DCNN-I,without the help of pose information) of our own model. It can be seen that our model surpasses previous methods and the added pose information is effective in improving the segmentation results. Our advantages mainly lie in small flexible parts: arms and legs, and that's exactly where pose consistency can help.

Qualitative evaluation. Fig. 6.4 visually illustrates the advantages of our model over two other recent methods, Attention [CYW15] and HAZN [XWC16]. Our model estimates the overall part configuration more accurately. For example, in the 2_{rd} row of Fig. 6.4, we correctly labels the right arm of the person while the other two baseline methods label it as upper-leg and lower-leg. Furthermore, our model gives clearer details of arms and legs (see the last three rows of Fig. 6.4), especially for small-scale people.

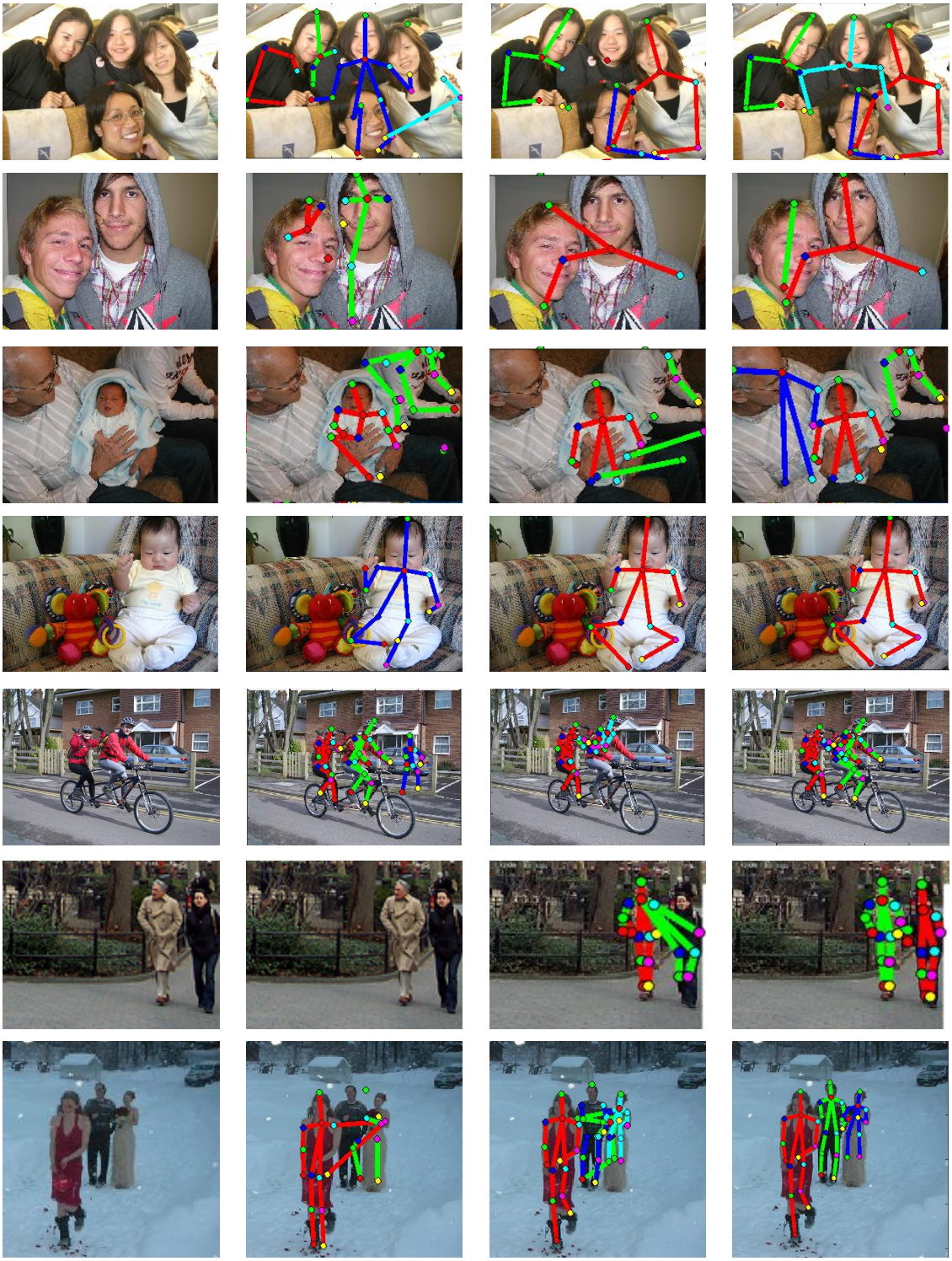


Figure 6.3: Visual comparison of human pose estimation on PASCAL-Person-Part.

6.3 Conclusion

In this chapter, we explore and prove the complementary properties of human pose estimation and human semantic part segmentation in complex multi-person images. We present an efficient framework that perform the two tasks iteratively, improving the results gradually. For human pose estimation, we adpot a fully-connected CRF that jointly performs human instance clustering and joint labeling, using deep-learned features and part segment based consistency features. This model gives better localization of joints, especially for arms and legs. For human semantic segmentation, we train a FCN that uses estimated pose configurations as shape and location priors, successfully relieving local confusions of people and giving clearer details of arms and legs. We also adopt an effective "auto-zoom" strategy that deals with object scale variation for both tasks and helps reduces the inference time of the CRF by a factor of 30. We test our approach on the challenging PASCAL-Person-Parts dataset and show that it outperforms state-of-the-art methods for both tasks.

There's still much space for improvement. For example, we can construct simple graphical models within a DCNN structure, making end-to-end training of human pose estimation possible. We can also perform pose estimation and part segmentation simultaneously, to further improve the results.

Method	Head	Torso	U-arms	L-arms	U-legs	L-legs	Background	Ave.
Attention [CYW15]	**81.47**	59.06	44.15	42.50	38.28	35.62	**93.65**	56.39
HAZN [XWC16]	80.76	60.50	45.65	43.11	41.21	37.74	93.78	57.54
LG-LSTM [LSX15]	82.72	60.99	45.40	47.76	42.33	37.96	88.63	57.97
Our model (w/o pose)	79.83	59.72	43.84	40.84	40.49	37.23	93.55	56.50
Our model (final)	80.21	**61.36**	**47.53**	**43.94**	**41.77**	**38.00**	93.64	**58.06**

Table 6.3: Mean Pixel IOU (mIOU) of Human Semantic Part Segmentation on PASCAL-Person-Part.

Figure 6.4: Visual comparison of human semantic part segmentation on PASCAL-Person-Part. Compared with Attention [CYW15] and HAZN [XWC16], our model is better at estimating the overall configuration, recovering small instances, and giving accurate details of arms and legs.

CHAPTER 7

Discussion and Conclusion

This dissertation proposes advancements in 2D human pose estimation and 2D semantic part segmentation to improve upon state-of-the-art performance. Current prevailing methods for the two tasks rely on deep neural networks or dedicated graphical models, achieving good performance on simple images with well-cropped person instances in standard poses. These methods, nevertheless, don't provide efficient mechanism to handle the following difficulties: (1) large variability of human pose; (2) large variability of human instances and human semantic parts; (3) multi-person overlapping. Therefore, they still struggle in multi-person natural images. Our proposed models, instead, directly tackle these difficulties, improving the performance of both tasks in multi-person images with large pose/scale variation.

We have explored and proved the complementary properties of pose estimation and semantic part segmentation, yielding models that are more robust to large pose variation. In our first model, we incorporate top-down pose cues as well as deep-learned features into an AOG-based semantic part assembling model, producing state-of-the-art results for images with constrained layouts. In our third model, we introduce part segment consistency terms into a fully-connected pose-assembling CRF, which outputs pose configurations with more accurate joint localization and less ambiguation in multi-person scenes. At the same time, we incorporate pose consistency terms into a part segmentation FCN, effectively correcting configuration errors in semantic part segmentation.

We handle the large scale variability in natural images by designing a mechanism to adapt to the size of human instancces adn their corresponding parts. In our second model, we adopt a hierarchical FCN framework that performs object/part scale estimation and part segmentation jointly, greatly recovering missing details of arms and legs for small-scale

human instances and effectively relieving local confusions for large-scale human instances.

There's still space for improvement on our proposed models. We can design layers that have the function of an assembling model within a DCNN architecture, making an end-to-end human pose estimation system. We can also design models that perform pose estimation and semantic part segmentation simultaneously within DCNNs, in order to make the two tasks benefit each other in a better way.

References

[AMF09] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. "From contours to regions: An empirical evaluation." In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 2294–2301. IEEE, 2009.

[ARS09] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. "Pictorial structures revisited: People detection and articulated pose estimation." In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 1014–1021. IEEE, 2009.

[BF11] Yihang Bo and Charless C. Fowlkes. "Shape-based pedestrian parsing." In *CVPR*, 2011.

[Bin71] Thomas O Binford. "Visual perception by computer." In *IEEE conference on Systems and Control*, volume 261, p. 262, 1971.

[BM09] Lubomir Bourdev and Jitendra Malik. "Poselets: Body part detectors trained using 3d human pose annotations." In *2009 IEEE 12th International Conference on Computer Vision*, pp. 1365–1372. IEEE, 2009.

[BMP00] Serge Belongie, Jitendra Malik, and Jan Puzicha. "Shape context: A new descriptor for shape matching and object recognition." In *NIPS*, volume 2, p. 3, 2000.

[CAF15] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. "Human pose estimation with iterative error feedback." *arXiv preprint arXiv:1507.06550*, 2015.

[CCB12] João Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu. "Semantic segmentation with second-order pooling." In *ECCV*, 2012.

[CML14] Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan L. Yuille. "Detect what you can: Detecting and representing objects using holistic models and body parts." In *CVPR*, 2014.

[COL16] Xiao Chu, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. "Structured feature learning for pose estimation." *arXiv preprint arXiv:1603.09065*, 2016.

[CPK15a] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan Yuille. "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs." In *ICLR*, 2015.

[CPK15b] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs." In *ICLR*, 2015.

[CS12] Joao Carreira and Cristian Sminchisescu. "Cpmc: Automatic object segmentation using constrained parametric min-cuts." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**(7):1312–1328, 2012.

[CY14] Xianjie Chen and Alan Yuille. "Articulated Pose Estimation by a Graphical Model with Image Dependent Pairwise Relations." In *NIPS*, 2014.

[CY15] Xianjie Chen and Alan Yuille. "Parsing Occluded People by Flexible Compositions." In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[CYW15] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L. Yuille. "Attention to Scale: Scale-aware Semantic Image Segmentation." *arXiv:1511.03339*, 2015.

[DA13] T Dharani and I Laurence Aroquiaraj. "A survey on content based image retrieval." In *Pattern Recognition, Informatics and Mobile Engineering (PRIME), 2013 International Conference on*, pp. 485–490. IEEE, 2013.

[DCS14] Jian Dong, Qiang Chen, Xiaohui Shen, Jianchao Yang, and Shuicheng Yan. "Towards unified human parsing and pose estimation." In *CVPR*, 2014.

[DCX13] Jian Dong, Qiang Chen, Wei Xia, Zhongyang Huang, and Shuicheng Yan. "A deformable mixture parsing model with parselets." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3408–3415, 2013.

[DDS09] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. "ImageNet: A Large-Scale Hierarchical Image Database." In *CVPR09*, 2009.

[DT05] Navneet Dalal and Bill Triggs. "Histograms of oriented gradients for human detection." In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pp. 886–893. IEEE, 2005.

[EEG14] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. "The Pascal Visual Object Classes Challenge: A Retrospective." *IJCV*, **111**(1):98–136, 2014.

[EW12] S. M. Ali Eslami and Christopher K. I. Williams. "A generative model for parts-based object segmentation." In *NIPS*, 2012.

[FE73] Martin A Fischler and Robert A Elschlager. "The representation and matching of pictorial structures." *IEEE Transactions on computers*, **22**(1):67–92, 1973.

[FH05] Pedro F Felzenszwalb and Daniel P Huttenlocher. "Pictorial structures for object recognition." *International Journal of Computer Vision*, **61**(1):55–79, 2005.

[FPZ03] Robert Fergus, Pietro Perona, and Andrew Zisserman. "Object class recognition by unsupervised scale-invariant learning." In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pp. II–264. IEEE, 2003.

[HAG15a] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. "Hypercolumns for Object Segmentation and Fine-grained Localization." In *CVPR*, 2015.

[HAG15b] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. "Hypercolumns for Object Segmentation and Fine-grained Localization." In *CVPR*, 2015.

[HLR14] Ahmad Humayun, Fuxin Li, and James M. Rehg. "RIGOR: Reusing Inference in Graph Cuts for generating Object Regions." In *Computer Vision and Pattern Recognition (CVPR), Proceedings of IEEE Conference on*, 2014.

[HW79] John A Hartigan and Manchek A Wong. "Algorithm AS 136: A k-means clustering algorithm." *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **28**(1):100–108, 1979.

[HYD15] Lichao Huang, Yi Yang, Yafeng Deng, and Yinan Yu. "DenseBox: Unifying Landmark Localization with End to End Object Detection." *arXiv:1509.04874*, 2015.

[HZR15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." *arXiv preprint arXiv:1512.03385*, 2015.

[IPA16] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. "DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model." *arXiv preprint arXiv:1605.03170*, 2016.

[JE10] Sam Johnson and Mark Everingham. "Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation." In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12.

[JFY09] Thorsten Joachims, Thomas Finley, and Chun-Nam John Yu. "Cutting-plane training of structural SVMs." *Machine Learning*, **77**(1):27–59, 2009.

[JS13] Simon Jones and Ling Shao. "Content-based retrieval of human actions from realistic video databases." *Information Sciences*, **236**:56–65, 2013.

[KH10] Rehanullah Khan, Allan Hanbury, , and Julian Stottinger. "Skin detection: A random forest approach." In *ICIP*, 2010.

[KK11] Philipp Krähenbühl and Vladlen Koltun. "Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials." In *NIPS*, 2011.

[KZ12] Ajay Kumar and Yingbo Zhou. "Human identification using finger images." *IEEE Transactions on Image Processing*, **21**(4):2228–2244, 2012.

[LBB14] Yanyun Lu, Khaled Boukharouba, Jacques Boonært, Anthony Fleury, and Stéphane Lecoeuche. "Application of an incremental SVM algorithm for online human recognition from video surveillance using texture and color features." *Neurocomputing*, **126**:132–140, 2014.

[LHK14] Yin Li, Xiaodi Hou, Christof Koch, James Rehg, and Alan Yuille. "The secrets of salient object segmentation." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 280–287, 2014.

[LKF10] Yann LeCun, Koray Kavukcuoglu, Clément Farabet, et al. "Convolutional networks and applications in vision." In *ISCAS*, pp. 253–256, 2010.

[LLL15] Si Liu, Xiaodan Liang, Luoqi Liu, Xiaohui Shen, Jianchao Yang, Changsheng Xu, Liang Lin, Xiaochun Cao, and Shuicheng Yan. "Matching-CNN Meets KNN: Quasi-Parametric Human Parsing." In *CVPR*, 2015.

[LMB14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. "Microsoft coco: Common objects in context." In *European Conference on Computer Vision*, pp. 740–755. Springer, 2014.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully Convolutional Networks for Semantic Segmentation." In *CVPR*, 2015.

[LSX15] Xiaodan Liang, Xiaohui Shen, Donglai Xiang, Jiashi Feng, Liang Lin, and Shuicheng Yan. "Semantic object parsing with local-global long short-term memory." *arXiv preprint arXiv:1511.04510*, 2015.

[LTZ13] Lubor Ladicky, Philip HS Torr, and Andrew Zisserman. "Human pose estimation using a joint pixel-wise and part-wise formulation." In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3578–3585, 2013.

[LWL11] Lingqiao Liu, Lei Wang, and Xinwang Liu. "In defense of soft-assignment coding." In *2011 International Conference on Computer Vision*, pp. 2486–2493. IEEE, 2011.

[LWT13] Ping Luo, Xiaogang Wang, and Xiaoou Tang. "Pedestrian parsing via deep decompositional network." In *ICCV*, 2013.

[RC12] Ingmar Rauschert and Robert T. Collins. "A generative model for simultaneous estimation of human body shape and pixel-level segmentation." In *ECCV*, 2012.

[RHG15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." *arXiv:1506.01497*, 2015.

[RPZ13] Brandon Rothrock, Seyoung Park, and Song-Chun Zhu. "Integrating grammar and segmentation for human pose estimation." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3214–3221, 2013.

[SB06] L. Sigal and M. J. Black. "Synchronized video and motion capture dataset for evaluation of articulated human motion." *Technical Report CS-06-08, Brown University*, 2006.

[SLJ15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.

[ST] B Sapp and B Taskar. "Multimodal decomposable models for human pose estimation." In *CVPR*, volume 13, p. 3.

[SZ14] K. Simonyan and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." *CoRR*, **abs/1409.1556**, 2014.

[TF10] Duan Tran and David Forsyth. "Improved human parsing with a full relational model." In *European Conference on Computer Vision*, pp. 227–240. Springer, 2010.

[TGJ15] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. "Efficient object localization using convolutional networks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 648–656, 2015.

[TKP15] Stavros Tsogkas, Iasonas Kokkinos, George Papandreou, and Andrea Vedaldi. "Semantic Part Segmentation with Deep Learning." *arXiv preprint arXiv:1505.02438*, 2015.

[TS14] Alexander Toshev and Christian Szegedy. "Deeppose: Human pose estimation via deep neural networks." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1653–1660, 2014.

[WM08] Yang Wang and Greg Mori. "Multiple tree models for occlusion and spatial constraints in human pose estimation." In *European Conference on Computer Vision*, pp. 710–724. Springer, 2008.

[WSL15] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan Yuille. "Joint Object and Part Segmentation using Deep Learned Potentials." In *ICCV*, 2015.

[WSS07] L. Wang, J. Shi, G. Song, and I. Fan Shen. "Object detection combining recognition and segmentation." In *ACCV*, 2007.

[WTL12] Yang Wang, Duan Tran, Zicheng Liao, and David Forsyth. "Discriminative hierarchical part-based models for human parsing and action recognition." *Journal of Machine Learning Research*, **13**(Oct):3075–3102, 2012.

[WWP00] Markus Weber, Max Welling, and Pietro Perona. "Towards automatic discovery of object categories." In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pp. 101–108. IEEE, 2000.

[WWY13] Chunyu Wang, Yizhou Wang, and Alan L Yuille. "An approach to pose-based action recognition." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 915–922, 2013.

[WY15] Jianyu Wang and Alan Yuille. "Semantic Part Segmentation Using Compositional Model Combining Shape and Appearance." In *CVPR*, 2015.

[WYY10] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong. "Locality-constrained linear coding for image classification." In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 3360–3367. IEEE, 2010.

[XWC16] Fangting Xia, Peng Wang, Liang-Chieh Chen, and Alan L Yuille. "Zoom Better to See Clearer: Human and Object Parsing with Hierarchical Auto-Zoom Net." In *ECCV*, 2016.

[XZW16] Fangting Xia, Jun Zhu, Peng Wang, and A Yuille. "Pose-Guided Human Parsing by an AND/OR Graph Using Pose-Context Features." In *AAAI Conference on Artificial Intelligence*, 2016.

[YBS13] Payman Yadollahpour, Dhruv Batra, and Gregory Shakhnarovich. "Discriminative Re-ranking of Diverse Segmentations." In *CVPR*, 2013.

[YKO15] Kota Yamaguchi, M Hadi Kiapour, Luis E Ortiz, and Tamara L Berg. "Retrieving similar styles to parse clothing." *IEEE transactions on pattern analysis and machine intelligence*, **37**(5):1028–1040, 2015.

[YLL14] Wei Yang, Ping Luo, and Liang Lin. "Clothing co-parsing by joint image segmentation and labeling." In *CVPR*, 2014.

[YLO12] Kota Yamaguchi, M. Hadi Kiapour Luis, E. Ortiz, and Tamara L. Berg. "Parsing clothing in fashion photographs." In *CVPR*, 2012.

[YR11] Yi Yang and Deva Ramanan. "Articulated pose estimation with flexible mixtures-of-parts." In *CVPR*, 2011.

[YY11] Ming Yang and Kai Yu. "Real-time clothing recognition in surveillance videos." In *2011 18th IEEE International Conference on Image Processing*, pp. 2937–2940. IEEE, 2011.

[YYG09] Jianchao Yang, Kai Yu, Yihong Gong, and Tingwen Huang. "Linear spatial pyramid matching using sparse coding for image classification." In *CVPR*, pp. 1794–1801. IEEE, 2009.

[ZCL08] Long Zhu, Yuanhao Chen, Yifei Lu, Chenxi Lin, and Alan Yuille. "Max Margin AND/OR Graph Learning for Parsing the Human Body." In *CVPR*, 2008.

[ZCL11] Long Leo Zhu, Yuanhao Chen, Chenxi Lin, and Alan Yuille. "Max margin learning of hierarchical configural deformable templates (hcdts) for efficient object parsing and pose estimation." *IJCV*, **93**(1):1–21, 2011.

[ZM07] Song-Chun Zhu and David Mumford. "A Stochastic Grammar of Images." *Foundations and Trends in Computer Graphics and Vision*, **2**(4):259–362, 2007.

[ZR12] Xiangxin Zhu and Deva Ramanan. "Face detection, pose estimation, and landmark localization in the wild." In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2879–2886. IEEE, 2012.

[ZWZ12] Jun Zhu, Tianfu Wu, Song-Chun Zhu, Xiaokang Yang, and Wenjun Zhang. "Learning Reconfigurable Scene Representation by Tangram Model." In *WACV*, 2012.