Joint Object and Part Segmentation using Deep Learned Potentials

Peng Wang[1]   Xiaohui Shen[2]   Zhe Lin[2]   Scott Cohen[2]   Brian Price[2]   Alan Yuille[1]
[1]University of California, Los Angeles   [2]Adobe Research

Abstract

Segmenting semantic objects from images and parsing them into their respective semantic parts are fundamental steps towards detailed object understanding in computer vision. In this paper, we propose a joint solution that tackles semantic object and part segmentation simultaneously, in which higher object-level context is provided to guide part segmentation, and more detailed part-level localization is utilized to refine object segmentation. Specifically, we first introduce the concept of semantic compositional parts (SCP) in which similar semantic parts are grouped and shared among different objects. A two-stream fully convolutional network (FCN) is then trained to provide the SCP and object potentials at each pixel. At the same time, a compact set of segments can also be obtained from the SCP predictions of the network. Given the potentials and the generated segments, in order to explore long-range context, we finally construct an efficient fully connected conditional random field (FCRF) to jointly predict the final object and part labels. Extensive evaluation on three different datasets shows that our approach can mutually enhance the performance of object and part segmentation, and outperforms the current state-of-the-art on both tasks.

1. Introduction

Decomposing an object into semantic parts enables a more detailed understanding of the object, which can provide additional information to benefit many computer vision tasks such as pose estimation [42, 11], detection [3, 8], segmentation [13], and fine-grained recognition [44]. Thus, it has become an attractive research topic to leverage semantic part representation through part detection [5, 16, 8], and human joint estimation [36].

In the literature of semantic segmentation, while object-level segmentation over multiple object categories has been extensively studied along with the growing popularity of standard evaluation benchmarks such as PASCAL VOC [15], object parsing (i.e., segmenting objects into semantic parts) is addressed mostly for a few specific categories provided with accurate localization such as human [45, 13, 11] and cars [13].

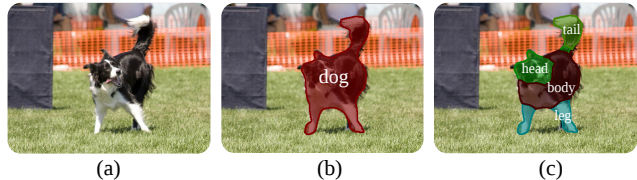With the increasing availability of semantic part anno-

Figure 1. We handle the prediction of semantic object and part segmentation in a wild scene scenario. (a) Original image. (b)&(c) are the object and part segmentation respectively, generated from our algorithm.

tations [8], more recent works have attempted to handle more difficult classes like animals with homogeneous appearance [39], and to perform both object and part segmentation [21], as illustrated in Fig. 1. However, in [21], object and part segmentation are performed sequentially, in which the object mask is first segmented, and then the part labels are assigned to the pixels within the mask. As a result, the errors from the predicted semantic object masks may be propagated to the parts.

In fact, object and part segmentation are complementary and mutually beneficial to each other. Semantic object segmentation requires a larger receptive field in order to correctly recognize the object, while part segmentation focuses on local details to obtain more accurate segmentation boundaries and accommodate large pose and viewpoint variations. If these two tasks are tackled simultaneously, by integrating the object-level guidance with part-level detailed segmentation, we can address two of the most challenging problems at the same time, i.e., discovering the subtle appearance differences between different parts within a single object, and avoiding the ambiguity across similar object categories. Motivated by this observation, we propose a joint solution to object and part segmentation, in which the consistency of the object and parts are enforced through joint training and inference.

Fig. 2 shows the overall framework of our approach. When performing part segmentation over multiple object classes, the appearance of some parts may be very similar, *e.g.*, horse legs and cow legs. Therefore in order to reduce the ambiguity and complexity during training, instead of treating each semantic part type independently [21], we allow some part labels to be shared by related object classes, and group the labelled parts of different classes into a se-
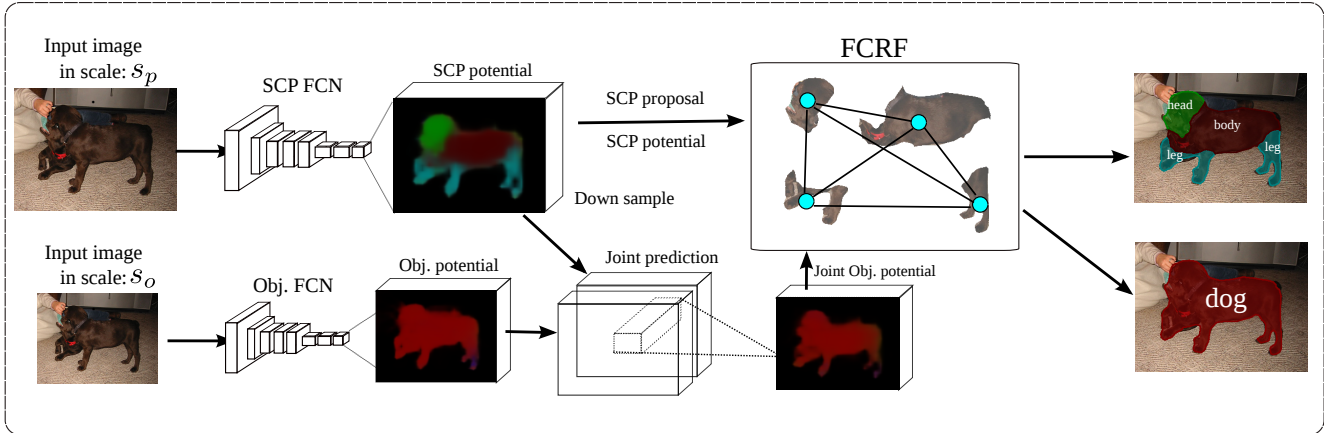
Figure 2. Our framework for joint part and object segmentation. Given an image, a two-stream FCN is performed to predict both the semantic compositional parts (SCP) and object potentials with different input image scales. The two potentials are then concatenated and fed to a new convolutional layer to predict joint object potentials. Finally, from SCP potentials, SCP segments are proposed as nodes for the fully-connected CRF to jointly infer the part and object labels.

mantic compositional part (SCP) representation based on their appearance and shape similarity (*e.g.*, horse legs and cow legs belonging to the same type of leg). Details of part sharing and SCP labels are illustrated in Fig. 3. Since we are performing object and part segmentation jointly, the ambiguity of shared part labels can be solved by the object label.

Given an image with certain objects, we train a two-stream fully convolutional network (FCN) with the first channel predicting the SCP potentials while the second predicting the object potentials. We then concatenate the two potentials as the input to an additional convolutional layer to refine the object potentials through joint training. At the same time, a compact set of SCP region proposals are generated from the SCP potentials. Using these region proposals as nodes, we construct a fully connected conditional random field (FCRF) to further incorporate the object and SCP potentials, yielding the jointly predicted object and part segmentations. In our FCRF, the consistency between the object and parts are enforced with long-range constraints.

We did extensive experiments over three derived datasets based on PASCAL VOC segmentation benchmark [14, 8]. Experimental results demonstrate that our joint approach improves both object and part segmentation, and significantly outperforms the state-of-the-art on both tasks.

2. Related work

In the literature of detection, the usefulness of mining semantic part representation in helping object recognition has been long studied. Felzenszwalb et.al [16, 19] proposed the deformable part-based model (DPM) which is an implicit way of discovering hidden parts. Later models use more accurate part-representation through explicit part supervision [34, 46]. Poselets [5] are proposed to model the local human parts through 2D projections from 3D data, which can be used as robust representation for both detection [20] and segmentation [29, 1]. In addition to human body parts,

Azizpour et. al [3] explicitly induce the bounding-box annotations of animal parts, yielding stronger detection results on both object and parts. Chen et. al [8] extend such ideas by providing richer labeling of part segments, which can better capture the appearance features for learning.

In the literature of segmentation, semantic parsing has also been actively investigated. However, due to its increased challenge in getting detailed boundaries, most previous work focused on parsing objects given both the semantic category and a cropped bounding box with no occlusion, such as human parsing [4, 45, 41, 13, 11], car parsing [35, 13, 28] or animal parsing [39]. Such methods are limited in their applications, as objects in real-world images are often occluded with large deformation and appearance variations, which is difficult to be handled by those shape-based [4] or appearance-based [11] models with hand-crafted features or bottom-up segments.

Recently, deep convolutional neural networks [24] (CNN) have achieved great success in many applications such as object detection [17, 22, 47] and end-to-end segmentation [40, 12, 21, 27, 7], with advanced network structures such as the VGG-Net [33]. Some studies tried to understand the implicitly learned filters [43] or comparable structures with the DPM [38, 18] in the network, and discovered some meaningful part representations in deeper layers. However, such representations are still not semantic parts, and using explicit supervision from semantic part labels for segmentation has not been investigated. In our approach, we propose to explicitly model the semantic parts along with the whole object by taking advantage of the recent advance of fully convolutional network (FCN) [27]. It has been very successful in predicting structured output such as semantic object segmentation. Specifically, FCN converts the fully-connected layers in the original CNN to 1×1 convolution layers, thus can efficiently perform

sliding-window-based classification at each pixel with a certain receptive field. However, it starts from local convolutional kernels with limited receptive fields and may not be able to capture all the long-range context, yielding local confusions. In our case, we solve such problem through modelling over object scale context, which is also suggested in prior arts [5, 9, 37, 10].

Perhaps the closest work in our scenario is the hypercolumn approach [21]. However, they perform object and part segmentation sequentially, and train many part classifiers separately for each class, which may suffer from increased training cost and has less scalability when the number of object class is large. In contrast, we use semantic compositional part (SCP) to allow part sharing and reduce training complexity. Moreover, our model leverages the advantage of both object segmentation and semantic part segmentation, yielding strong results in very challenging scenarios. To the best of our knowledge, this is the first work that provides a joint solution to tackle the segmentation of semantic parts and objects, which allows the part and object potentials interact and benefit each other.

3. Joint part and object segmentation

Our framework includes four major parts, i.e. shared semantic compositional parts (SCP) generation, part and object potentials, proposal of SCP regions and fully connected conditional random field (FCRF). In the following sessions, we will describe these techniques in details.

3.1. Semantic compositional parts

When performing semantic object segmentation and parsing over multiple object classes, some parts from different object classes yet with similar semantic meanings may also have very similar shapes and appearances (*e.g.*, horse legs and cow legs). In such scenarios, allowing the parts to be shared among similar object classes [46, 31] could alleviate the difficulties of distinguishing similar parts from different objects, and at the same time reduce the increasing complexity of training and inference as the number of object categories grows. Therefore, before we formally train our framework for these two tasks, we group those similar parts to form the semantic compositional parts (SCP) that are shared among related object classes.

In particular, given a semantic part, it has an object label l_o (*e.g.*, horse) and a particular semantic meaning l_s (*e.g*, leg). The joint label of this part we would like to infer is denoted by l_{op} (*e.g*, horse-leg). We group the original part labels l_{op} to a shared compositional part representation l_{scp} if they have the same semantic meanings l_s and highly similar appearances and shapes. The SCP l_{scp} are then used to compose different objects l_o as illustrated in Fig. 3. For example, in the case of horse and cow, the representation of different labels are, $l_o \in$ {horse, cow}, $l_{op} \in$ {horse-head, horse-body, horse-leg, horse-tail, cow-head, cow-leg, cow-body, cow-tail} and $l_s \in$ {head, body, leg, tail}. If we
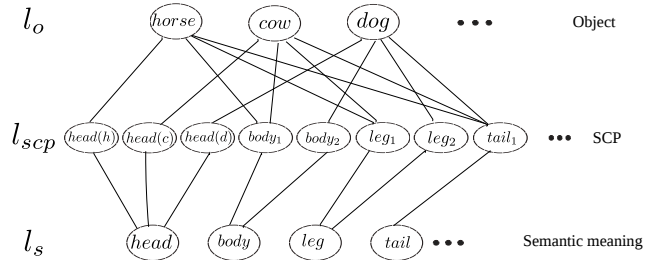

Figure 3. Illustration of our semantic compositional part (SCP) grammer in Sec. 3.1. Each SCP is associated with one semantic meaning, and all the objects are composed of several SCPs.

allow the two objects to share the same type of body, leg and tail, we get the SCP label $l_{scp} \in$ {head(h), head(c), $body_1$, leg_1, $tail_1$ }, as in Fig. 3, which is a much smaller prediction space than l_{op}. The information of l_{op} is kept in the connections between l_o and l_{scp}. During the inference, by enforcing the consistency of l_{scp} and l_o, l_{op} can be directly recovered, *e.g.* known $l_o =$ horse and $l_{scp} = leg_1$, then we get $l_{op} =$ horse-leg. Currently we manually group those part labels. Nonetheless, automatically generating SCP is a very interesting problem especially with increased number of object categories, and will be investigated in the future.

3.2. Deep part and object potentials

In this section, we mainly describe the joint prediction of the semantic compositional parts (SCP) potentials and object potentials by a two-stream FCN, which is then used to construct the FCRF. Specifically, as illustrated in the framework (Fig. 2), the first channel of the FCN predicts a $(N_p + 1)$-channel SCP potential map (N_p is the SCP label number, while the additional label represents the background). Similarly, the second channel of the network predicts a $(N_o + 1)$-channel object-class potential map (N_o being the object class number). In addition, we concatenate the SCP potentials and the object potentials as a set of high-level features, and feed them to a new $k \times k$ conv. layer for object potential refinement. This predicted joint object potential has less noise within the object and better boundaries. It is because firstly, the conv. layer accesses larger context for reducing the confusions, and secondly, our SCP potentials keep better edge information that can interact with the object potentials during joint training. As shown in Tab. 2, compared with the original FCN, the "Joint FCN" produce better performance.

One may consider to also generate refined SCP potentials similarly using the joint prediction layer. However, using the SCP potentials in this way does not show much improvement in our experiments. This is because firstly, the SCP potentials have already encoded more detailed boundary information than the object potentials, and the interactions would not help the SCP potentials to refine their boundaries. Secondly, the ambiguity of similar parts from different objects, which is the most challenging problem in

part segmentation, has already been better addressed by using part sharing and the object-scale FCRF. Therefore, the joint prediction for SCP refinement is not adopted in our framework to reduce the system complexity.

Last but not the least, the SCP potentials and object potentials actually need different levels of context. This is also a key factor in DPM [16], where they use a larger image scale for part filters and a smaller scale for root filters. In our case, we adopt a similar strategy with different input resolutions to obtain proper receptive fields, i.e. $s_p \times s_p$ for SCP and $s_o \times s_o$ for object, where we require $s_p > s_o$. We investigated the influence of input scale in our experiments, and chose the optimal s_p and s_o using cross-validation.

3.3. SCP segments proposal

As mentioned earlier, our FCRF is built upon a compact set of SCP segments. In our case, traditional object proposal algorithms such as CPMC [6] or MCG [2] would typically fail due to the subtle difference of appearance between connected parts such as the leg and body. Nevertheless, we can use the SCP FCN network to generate accurate segments that are associated with SCP. Based on the predicted $(N_p + 1)$-channel SCP probability map, we assign the SCP label with the highest probability to each pixel and generate the SCP label map. SCP segments are then generated by grouping the pixels with the same SCP labels.

Fig. 4 shows several examples of the proposed SCP segments from an image. We can see our SCP segments work very well in terms of capturing correct semantic part regions and locating the object boundaries. To practically evaluate the segments, we did an oracle experiments by assigning the proposed SCP segments to be the overlapping ground truth class labels. The best possible object IOU is 85.1% over the quadrupeds animal set in Sec. 4.3, which performs reasonably well for capturing the object boundaries. One might also apply local dense pixel-wise refine strategy [7] to further refine the segments, but it will increase computational cost and need further investigation.

3.4. Joint FCRF

FCN is essentially a sliding-window based approach, where its receptive field is always fixed to a local area given the input image, thus local confusion can hardly be avoided in many cases. Intuitively, the optimal receptive field should be close to the scale of the presented object in the image. Therefore, after generating SCP segments as well as SCP and object potentials from the joint FCN, we further construct a fully-connected CRF (FCRF) to automatically discover the object-scale context, where all the parts of an object can interact with each other, yielding the optimal semantic object and part prediction simultaneously.

Specifically, given the set of SCP segment proposals, we first cluster the proposed SCP segments into several groups according to their spatial distances, in case there are mul-
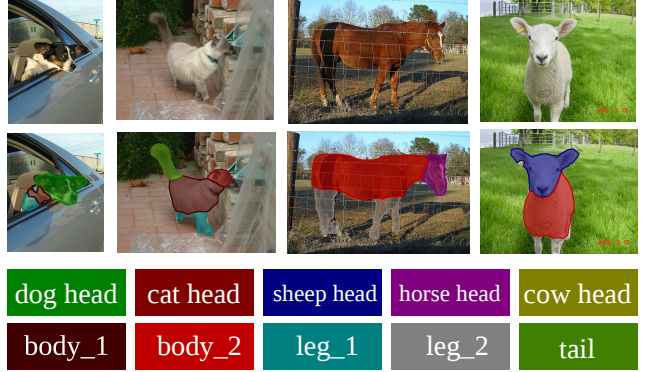

Figure 4. The SCP segments proposals handle various difficulties such as truncation, occlusion, deformation and view point changes. The colors maps shown at the bottom for each kind of SCP. We keep the color consistent in the results and will show more in our supplementary materials.

tiple isolated objects in an image. Two SCP segments are merged in the same group if the minimum distance of the pixels within these two segments is smaller than a threshold $t_s = 10$. We assume each group of SCP segments forms an object or overlapping objects in the image, which provides a good estimation of the object-scale. Then, we build one FCRF for each corresponding group. Formally, the FCRF can be represented as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of SCP segments in the same group and $\mathcal{E}$ is the set of edges connecting every pair of segments. As introduced in our framework (Fig. 2), for each SCP segment $\mathbf{P}$, we want to predict its semantic part label $l_{op}(\mathbf{P})$ (defined in Sec. 3.1), but can be reduced to separately inferring the object label $l_o(\mathbf{P})$ and SCP label $l_{scp}(\mathbf{P})$ by enforcing their label consistency. In the following we use l_{op}^i for $l_{op}(\mathbf{P}_i)$, l_p^i for $l_{scp}(\mathbf{P}_i)$ and l_o^i for $l_o(\mathbf{P}_i)$ for simplicity.

In sum, our FCRF is formulated as,

$$\min_{\mathcal{L}} \sum_{i \in \mathcal{V}} \psi_i(l_{op}^i) + \lambda_e \sum_{i,j \in \mathcal{V}, i \neq j} \psi_{i,j}(l_{op}^i, l_{op}^j)$$

where, $\psi_i(l_{op}^i) = \eta(l_o^i, l_p^i)(\psi_i^o(l_o^i) + \lambda_p \psi_i^p(l_p^i));$ (1)

$$\psi_{i,j}(l_{op}^i, l_{op}^j) = \eta(l_o^i, l_p^i)\eta(l_o^j, l_p^j)\psi_{i,j}^{op}(l_o^i, l_o^j, l_p^i, l_p^j);$$

where $\mathcal{L}$ is the label set of the proposed SCP segments, λ_e and λ_p are balancing parameters. $\psi_i^o(l_o^i) = \sum_{\mathbf{x}_j \in \mathbf{P}_i} -\log(P(l_o(\mathbf{x}_j)))$, is the sum of pixel-wise object potentials inside the SCP segment $\mathbf{P}_i$, and $\psi_i^p(l_p^i)$ is similarly defined with the SCP potentials. $\eta(l_o^i, l_p^i)$ is a constraint for part and object combination, which is 1 if l_o^i, l_p^i is a meaningful combination, i.e. a connection exists between l_o^i and l_p^i in the grammar as in Fig. 3, and set to be ∞ otherwise.

In order to learn the pairwise potentials, we train a two-layer fully-connected neural network, which takes the features from a pair of segments i and j as input, and predicts the probabilities of the four labels $l_o^i, l_o^j, l_p^i, l_p^j$. The ground

truth labels for each training segment are the most dominant labels of the pixels within the segment, and we adopt the multinomial logistic loss for training.

For the pairwise features, we consider multiple semantically meaningful cues from the pairwise relations, i.e. $\mathbf{f}_{ij} = [\mathbf{f}_i^T, \mathbf{f}_j^T, \kappa_{ij}, \theta_{j|i}]^T$, where $\mathbf{f}_i = [\mathbf{f}_{oi}^T, \mathbf{f}_{pi}^T, \mathbf{f}_{ai}]^T$ is a segment self-descriptor, $\kappa_{ij} = [d_{ij}^a, d_{ij}^p]^T$ is a spatial metric of the segment pair. $\theta_{j|i}$ is the relative angle. We summarize the features as follows, (1) $\mathbf{f}_{oi}^T$: the mean of pixel-wise object potentials. (2) $\mathbf{f}_{pi}^T$: the mean of pixel-wise SCP potentials. (3) $\mathbf{f}_{ai}$: the segment area, normalized by the discovered object area. (4) d_{ij}^a: the appearance geodesic distance from segment i to j, which accumulates the edge weights along the path from i to j on the image. (5) d_{ij}^p: the Euclidean distance between the center of the two segments normalized by the height and width of the object. (6) $\theta_{j|i}$: the relative angle of the center of the segments j with respect to the center of segment i. Specifically for d_{ij}^a, the edge weight between the two neighbouring segments is the sum of the edgemap [25] values over their overlapping boundary.

After the model is learned, given the pairwise feature $\mathbf{f}_{ij}$, the potentials $\psi_{i,j}^{op}(l_o^i, l_o^j, l_p^i, l_p^j)$ is computed as the negative log-likelihood, i.e. $-\log(P(l_o^i)P(l_o^j)P(l_p^j)P(l_p^j))$, where the probabilities are from the neural network prediction.

For inference, since our graphical model have a very small number of nodes (less than 15 SCP segments in average), using the efficient LBP [30], our algorithm can converge very fast within 5 iterations.

3.5. Relation to the DPM structure

While we are dealing with semantic segmentation and parsing, our approach follows the spirit of DPM [16] in object detection. In [18], Girshick et. al connected the DPM root filter and part filters with the convolutional filters from CNN, and the distance transform can be regarded as an additional pooling and geometry filtering step. Our work aims to solve the segmentation task here but has analogy with DPM, where the object prediction can be considered as the root filter and SCP potentials are our learned part filters. The difference is that rather than firing when the target is at the exact center of the detection window, our model fires at every location inside the target which produces more accurate location estimation. Regarding the part and root geometry, rather than explicitly modeling the geometry of root-part distance transform as in DPM, we implicitly model these spatial relationships using spatial distances and relative angles as spatial features and learn the pair-wise potentials in the fully-connected graphic model, which is more data-driven and generalizes better for handling variations.

For inference, DPM tries to find the probability of the part-root locations $\mathbf{p}$ given the object label l_o through sliding window, i.e. $\max_{\mathbf{p} \in \mathbf{I}} P(\mathbf{p}|l_o)$. In our case, sliding-window for part and object localization is realized by our FCN, based on which we can infer over the smaller label space i.e. $\max_{l_o \in \mathcal{L}} P(l_o|\mathbf{p})$, at the object-scale context.

4. Experiments

In this section, we provide all the experimental details, and evaluate our approach in terms of different experimental settings to demonstrate the advantage of our approach. Specifically, we conduct experiments on the Horse-Cow parsing dataset introduced in [39], the PASCAL Quadrupeds dataset from [8], and our PASCAL Part benchmark, over which extensive comparisons are performed.
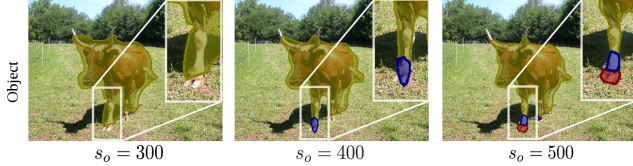
4.1. Model training.

Both of our FCN models for SCP and object prediction are based on the 16-stride (16s) FCN, since there is trivial improvement (less than 1% as shown in our Tab. 2 and Tab.2 of [27]), while significant more time required to train a 8-stride model (8s).

Data augmentation. Effective data augmentation is the key to the success of FCN. Thus in our case, we did sufficient augmentation by first cropping out the connected object masks using a random generated bounding box around it. The size of the crop is specially 1.3 times larger than the object bounding box, which is a rough localization of the object for generalization. Then, each cropped image is resized into 300×300, based on which we further augmented using the ideas from [12], i.e. we perform 4 additional random cropping at the size of 200×200, flipping, changing the color intensity by a random scale in $[0.7, 1.3]$ with a probability of 0.4 and rotation in $[-5, +5]$ degree with a probability of 0.5. In average, each image is augmented up to around 25 training samples.

Optimization and step-wise training. We fine-tune all our models step-by-step from the publicly available VGG-net [27]. For training the SCP FCN, we first train a 32s FCN with the learning rate as 10^{-4} for the final $conv_{fc8}$ layer which is the layer predicting the SCP potentials, and 10^{-5} learning rate for the layers after $pool4$, while we fix the layers before $pool4$. Then for the 16s FCN, we start with training the $conv_{poo4}$ layer which is used for predicting SCP potentials from the $pool4$'s output. Then, we use the trained layers as initialization for the final 16s FCN training. We further fine-tune the $conv_{fc8}$ and $conv_{poo4}$ layer with a learning rate as 10^{-5}. For training the joint object FCN, we fix the SCP FCN and concatenate the SCP potentials with the 16s object potentials, over which another convolutional layer $conv_{jnt}$ (with a kernel size of 5) is used to predict the joint object potentials. For fine-tuning this model, we use 10^{-4} learning rate for the $conv_{jnt}$ layer, while 10^{-6} learning rate for the $conv_{fc8}$ and $conv_{pool4}$ of the object FCN. In all the cases, we keep the batch size as 32. For the other parameters, we refer to the ones given by [27].

For the two-layer neural network in training the pairwise potentials of the fully-connected CRF, we set 32 hidden

Figure 5. Investigation of image scales. Top: an example from our joint object FCN prediction, showing that larger scale leads to finer boundaries, but introduces local ambiguities. Bottom: validate the accuracy of scale combination of s_o and s_p.

	$s_p = 400$	$s_p = 500$	$s_p = 600$
$s_o = 300$	78.58	78.88	**79.18**
$s_o = 400$	77.55	78.62	78.11
$s_o = 500$	77.09	75.81	74.20

nodes, and use the RELU for no-linearity with a dropout rate of 0.2 to regularize the model. We use a batch size of 10000 and set the learning rate to be 10^{-2}. In learning the pair-wise term, other than separately training SCP and object labels, i.e. l_{scp} and l_o, one may also consider output the semantic part label l_{op} which lies in the joint space of object and part. Nevertheless, we found it is harder to train due to that a lot more data are required for prediction in a high dimensional output space. Thus we chose to train separately.

In addition, we found it is important to balance different classes for part segmentation since some part can be very small, e.g. tails. Specifically, for both FCN and FCRF, we set the weight for each class in terms of the portion of samples in the class. This help the network avoid overwhelming of other dominant classes.

All our neural networks are based on the caffe platform [23] and partially from the code provided by [27].

4.2. Parameters and details

In the FCRF, we set $\lambda_e = 2$ and $\lambda_p = 0.3$ which are validated over a validation set from the Quadrupeds dataset. The same set is used for all other validation experiments. For inference over the graphical model, we use the LBP tool provided by Meltzer[1].

Investigation of input image scale. The input image scale for FCN is one of the most important factors for achieving good performance as also shown in recent works [7, 26]. Fig. 5 shows the investigation of changing the input image scale for SCP proposals and object potentials. In the top row, we can see the larger the image scale is, the more accurate the object boundaries could be, while the more local confusion it has, *e.g.* a leg of the cow is starting to be confused with horse. Thus, we validate the scale of $s_p \in \{400, 500, 600\}$ and $s_o \in \{300, 400, 500\}$. At the bottom of Fig. 5, we show the validated results, and the optimal combination, i.e. $s_p = 600, s_o = 300$ is used in all our experiments.

Training and inference time. For training, we found the model would converge after 40k iterations and it takes

---

[1]http://www.cs.huji.ac.il/ talyam/inference.html

around 2 days for SCP FCN, 1 day for object potentials and 8 hours for the graphical model with a platform of 4 core 3.2Hz CPU and a K40 GPU. For inference, in average, one image takes 0.3s for FCN forward propagation with the GPU and 1.3s for the graph model with our CPU.

4.3. Performance comparisons

We compare our algorithm with three state-of-the-art methods on the two tasks. For object segmentation, we compare our method with the FCN [27]. We use the code provided by the author and tune their model based on our dataset, including tuning a fixed optimal image scale for a fair comparison. For semantic part segmentation, we compare with the most recent compositional-based semantic part segmentation (SPS) [39] over the Horse-Cow parsing dataset, and the hypercolumn (HC) [21] in all datasets. For comparing with HC [21], as there is no available code or results from the author, we implemented a similar part parsing framework by first performing figure-ground mask with the trained FCN 8s [27], and then assign part labels inside with a optimally tuned image scale to be our baseline method. This baseline is close to HC and also performs well in the experiments, but is different from [21] in some details, thus we mark it as HC*. For evaluation, we adopt the standard intersection over union (IOU) criteria.

Horse-Cow parsing dataset. The Horse-Cow dataset is a two-animal part segmentation benchmark proposed in [39].
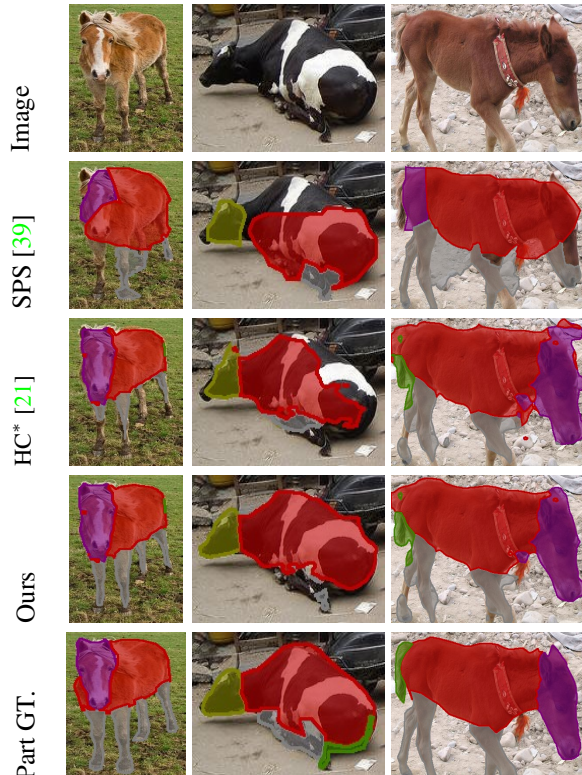

Figure 6. Comparison examples with SPS [39] and HC* [21] in the horse, cow dataset and the color map is in Fig. 4 (Best view in color).

For each animal class, they manually select the mostly ob-servable animal instances in both train-val and test set from the PASCAL VOC 2010 [15]. There are 294 training exam-ples and 277 test images. Since it is not published yet, we asked the author for this dataset and their results. In [39], the task is to segment the part given the object class. Thus for a fair comparison, we also test every instance with the object class known. In such a case, our graphical model in Eqn.(1) is reduced to inferring the part label l_p with a binary object potential l_o.

Tab. 1 provides the results of SPS [39], HC* [21] and ours over semantic part and figure-ground segmentation. Our method outperforms the previous state-of-the-art with a significant margin, averagely 13% better than SPS and 4.5% better than HC*. Several qualitative examples are shown in Fig. 6, where we keep the color map consistent with the SPC labels (Fig. 4). In these cases, we can see that, built on explicit geometry rules, SPS is limited in handling large variance of object parts, which makes it difficult to model highly deformable cases like the tail, or occluded animals. For HC*, due to inaccuracy from the object mask, it may miss some detailed regions like legs such as the 1_{st} column of horse in Fig. 6.

Quadrupeds dataset. We further extend our experiments into the Quadrupeds part dataset which contains five animal classes, i.e., cat, dog, sheep, cow and horse. In this task, we simultaneously predict the object and part masks. We obtain the data given by [8], which include all the part la-bels in PASCAL VOC 2010 training and validation images. We select the images containing the target objects, and treat the validation set as our test images, resulting 3120 train-ing images and 294 testing images. In addition, since we are focusing more on semantic part segmentation, during testing, we roughly localize the object inside by using the strategy in data augmentation (Sec. 4.1). We will provide our code of this localization to others for a fair comparison. It should be noted that such localization is still very coarse with much looser bounding boxes than the ones needed in previous parsing methods [11, 39].

Following HC, the parts of all the quadrupeds are la-belled into head, body, leg and tail, from which we con-struct a shared SCP grammar as introduced in Sec. 3.1. In

our grammar, horse, cow and sheep share the same body and leg type, while cat and dog share others. All the an-imals share a tail label, and each animal has its own head label since the head is highly distinguishable [32]. In total, 10 SCPs are used, while 20 labels are need if all the parts are treated independently.

Tab. 2 shows the compared results on both object and part segmentation. As shown, in terms of object segmenta-tion, comparing with FCN [27], our final results improves 6.2%, which demonstrates the advantage of the joint model. We also compare the variants of our approach with different components. "Joint FCN(16s)" produces the object masks directly from the two-stream FCN without FCRF inference. By including the SCP potentials, the object prediction of "Joint FCN (16s)" already out-performs FCN, which shows that the joint object potentials have less pixel-wise confu-sion. In addition, as shown in the "FCRF+FCN(16s)", us-ing the FCRF with the FCN object potentials without the joint convolutional layer improves over 2% compared to "FCN(16s)", showing FCRF inference can help refine seg-mentation results by exploring long-range context. More-over, with the joint potentials, the performance of our full model has another 4% boost. This shows the joint FCN object potentials provide better evidence for our graphical model and is essential to our system. At last, in "Ours (w/o SCP)", we investigate the usefulness of SCP by not allow-ing the part sharing. The results drops 2% and 4% for ob-ject and part respectively compared to our final results. This is mostly because more confusion happened for parts with similar appearance, yielding worse part segments.

We show several qualitative comparison examples with the FCN at left of Fig. 7, in which our algorithm is able to solve the local ambiguities that FCN usually encounters. For instance, at the 4_{th} row, the legs of a cow are confused with horse using FCN. In contrast, by borrowing the object-scale evidence from the cow body and cow head, our model can correct this local confusion. In addition, as shown in the horse segments at the 3_{rd} row, our SCP segments are able to provide more precise object boundaries in many difficult cases, like the legs of the horse crossing the bar.

	Bkg	head	body	leg	tail	Fg	IOU	Pix. Acc
			Horse					
SPS [39]	79.14	47.64	69.74	38.85	-	68.63	-	81.45
HC* [21]	85.71	57.30	**77.88**	51.93	37.10	78.84	61.98	87.18
Ours	**87.34**	**60.02**	77.52	**58.35**	**51.88**	**80.70**	**65.02**	**88.49**
			Cow					
SPS [39]	78.00	40.55	61.65	36.32	-	71.98	-	78.97
HC* [21]	81.86	55.18	72.75	42.03	11.04	77.07	52.57	84.43
Ours	**85.68**	**58.04**	**76.04**	**51.12**	**15.00**	**82.63**	**57.18**	**87.00**

Table 1. Average precision over the Horse-Cow dataset.

	Bkg	Dog	Cat	Cow	Horse	Sheep	IOU	Pix. Acc
			Object segmentation accuracy					
FCN 16s [27]	93.25	74.30	78.62	61.88	56.56	67.63	72.04	93.00
FCN 8s [27]	93.55	74.39	78.52	60.81	58.39	69.15	72.47	93.17
Joint FCN(16s)	94.04	75.13	80.52	66.76	63.04	**71.54**	75.17	93.77
FCRF+FCN(16s)	93.88	77.10	80.92	68.76	63.40	64.54	74.57	93.87
Ours (w/o SCP)	93.29	77.50	77.81	70.24	68.94	71.31	76.51	93.89
Ours final	**94.40**	**79.03**	**83.04**	**74.82**	**69.94**	70.59	**78.64**	**94.71**
			Semantic part segmentation accuracy					
HC* [21]	92.83	42.07	43.99	35.49	38.59	33.80	41.36	89.54
Ours (w/o SCP)	93.31	40.71	43.37	36.65	43.65	32.81	42.01	90.68
Ours final	**94.46**	**45.63**	**47.81**	**42.7**	**49.60**	**35.74**	**46.69**	**91.74**

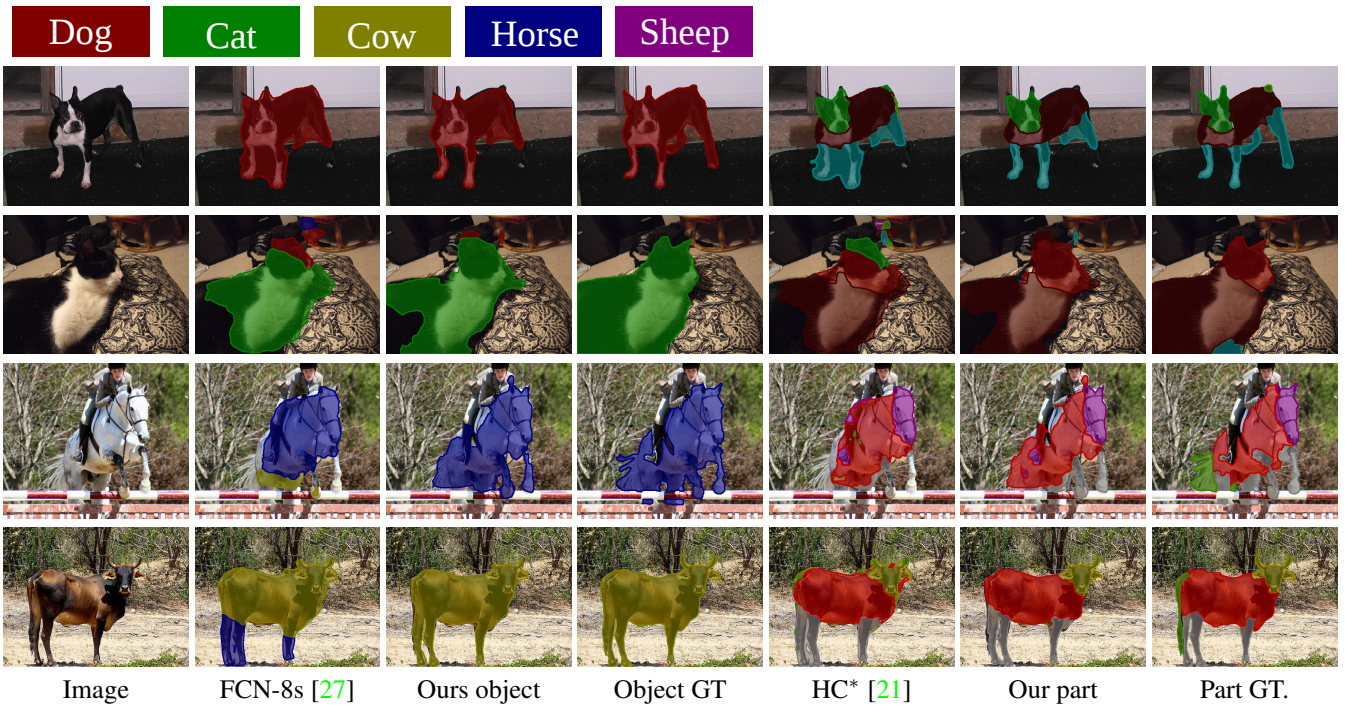Table 2. Average precision over the Quadrupeds data.

Figure 7. Comparison examples with FCN [27] for object segmentation and HC* [21] for part segmentation in the Quadrupeds dataset. The object label color map is shown at above and part label color map is shown in Fig. 4 (Best view in color).

For semantic part segmentation, we summarized the mean IOU of all parts for each object in Tab. 2 due to space limitation. We will include more detailed results in our supplementary materials. Our results are also significantly higher than the results of HC* [21] (over 5%). As HC performs object and part segmentation sequentially, the errors in object predictions, including local confusion and inaccurate boundaries, will propagate to the parts. For example, at the 2_{nd} row of Fig. 7, the cat and dog are confused in the object FCN prediction, which makes part of the cat head errorly labelled as dog head in HC. We solve such problems through the FCRF by considering object-scale context. In addition, thanks to our optimized image scale for both object and part, our method can capture part boundaries that are sometimes missed by the FCN and HC. We provide more results in our supplementary materials.

PASCAL part segmentation benchmark. In addition to the labels from the train-validation set of [8], following the same hierarchical part labelling system, we additionally labelled semantic parts over the PASCAL VOC 2010 test set [14] of the object segmentation task, which includes 994 images. With respect to the PASCAL VOC test benchmark, we are not going to release the labels and will instead launch an evaluation server for researchers to fairly compare their part segmentation results.

We test our algorithm over images with the five quadrupeds, which include 281 images. As shown in Tab. 3, the results are consistent with that from the validation set, and our method achieves the best overall IOU outperforming the state-of-the-art with a large margin.

5. Conclusion and discussion

In this paper, we proposed the framework for jointly solving object and part segmentation. Our approach follows the spirit of DPM, and leverages the advantages of both sides, yielding the state-of-the-art results. Recently, for object segmentation, there are other methods that provides additional improvement over the FCN such as adding pixel-wise Dense CRF [7, 26]. Our framework is complementary to them in terms of modelling over parts and adaptive to object scale for solving local ambiguity. In addition, we tackles the part segmentation beyond the object. Possible failure cases for us would be strong appearance confusion, strong occlusion, where object scale or SCP segments can be misled, yielding inaccurate results (*e.g.* the back leg of the dog in the first example of Fig. 7).

In the future, we will try to automatically learn the SCP, and jointly model the instance segmentation to incorporate detection, which provides better object scale and solve the remaining localization issue.

	Bkg	Dog	Cat	Cow	Horse	Sheep	IOU	Pix. Acc
Object segmentation accuracy								
FCN 8s [27]	94.45	70.14	75.45	64.06	64.75	**69.06**	72.99	93.90
Ours final	**95.31**	**77.44**	**80.47**	**72.13**	**76.18**	67.96	**78.25**	**95.26**
Semantic part segmentation accuracy								
HC* [21]	94.36	41.24	42.42	35.22	45.00	**38.86**	43.11	90.64
Ours final	**95.14**	**46.52**	**48.06**	**41.80**	**56.67**	36.02	**48.16**	**92.47**

Table 3. Average precision over the part segmentation benchmark.

References

[1] P. Arbelaez, B. Hariharan, C. Gu, S. Gupta, L. D. Bourdev, and J. Malik. Semantic segmentation using regions and parts. In *CVPR*, pages 3378–3385, 2012. 2

[2] P. A. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marqués, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, pages 328–335, 2014. 4

[3] H. Azizpour and I. Laptev. Object detection using strongly-supervised deformable part models. In *ECCV*, pages 836–849, 2012. 1, 2

[4] Y. Bo and C. C. Fowlkes. Shape-based pedestrian parsing. In *CVPR*, pages 2265–2272, 2011. 2

[5] L. D. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *ICCV*, pages 1365–1372, 2009. 1, 2, 3

[6] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV (7)*, pages 430–443, 2012. 4

[7] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 2, 4, 6, 8

[8] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. L. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*, pages 1979–1986, 2014. 1, 2, 5, 7, 8

[9] S. K. Divvala, D. Hoiem, J. Hays, A. A. Efros, and M. Hebert. An empirical study of context in object detection. In *CVPR*, pages 1271–1278, 2009. 3

[10] C. Doersch, A. Gupta, and A. A. Efros. Context as supervisory signal: Discovering objects with predictable context. In *ECCV*, pages 362–377, 2014. 3

[11] J. Dong, Q. Chen, X. Shen, J. Yang, and S. Yan. Towards unified human parsing and pose estimation. In *CVPR*, pages 843–850, 2014. 1, 2, 7

[12] D. Eigen and R. Fergus. Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture, 2014. 2, 5

[13] S. M. A. Eslami and C. K. I. Williams. A generative model for parts-based object segmentation. In *NIPS*, pages 100–107, 2012. 1, 2

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html. 2, 8

[15] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 1, 7

[16] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, 2010. 1, 2, 4, 5

[17] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. June 2014. 2

[18] R. Girshick, F. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. In *CVPR*, 2015. 2, 5

[19] R. B. Girshick, P. F. Felzenszwalb, and D. A. McAllester. Object detection with grammar models. In *NIPS*, pages 442–450, 2011. 2

[20] C. Gu, P. A. Arbeláez, Y. Lin, K. Yu, and J. Malik. Multi-component models for object detection. In *ECCV*, pages 445–458, 2012. 2

[21] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015. 1, 2, 3, 6, 7, 8

[22] B. Hariharan, P. A. Arbeláez, R. B. Girshick, and J. Malik. Simultaneous detection and segmentation. In *ECCV*, pages 297–312, 2014. 2

[23] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 6

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012. 2

[25] M. Leordeanu, R. Sukthankar, and C. Sminchisescu. Efficient closed-form solution to generalized boundary detection. In *ECCV (4)*, pages 516–529, 2012. 5

[26] G. Lin, C. Shen, I. Reid, and A. van dan Hengel. Efficient piecewise training of deep structured models for semantic segmentation, 2015. 6, 8

[27] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CVPR*, Nov. 2015. 2, 5, 6, 7, 8

[28] W. Lu, X. Lian, and A. L. Yuille. Parsing semantic parts of cars using graphical models and segment appearance consistency. In *BMVC*, 2014. 2

[29] M. Maire, S. X. Yu, and P. Perona. Object detection and segmentation from joint embedding of parts and pixels. In *ICCV*, pages 2142–2149, 2011. 2

[30] K. P. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *CoRR*, 2013. 5

[31] P. Ott and M. Everingham. Shared parts for deformable part-based models. In *CVPR*, pages 1513–1520, 2011. 3

[32] O. M. Parkhi, A. Vedaldi, C. V. Jawahar, and A. Zisserman. The truth about cats and dogs. In *ICCV*, pages 1427–1434, 2011. 7

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 2

[34] M. Sun and S. Savarese. Articulated part-based model for joint object detection and pose estimation. In *ICCV*, pages 723–730, 2011. 2

[35] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, and L. J. V. Gool. Using recognition to guide a robot's attention. In *RSS*, 2008. 2

[36] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *CVPR*, pages 1653–1660, 2014. 1

[37] Z. Tu and X. Bai. Auto-context and its application to high-level vision tasks and 3d brain image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(10):1744–1757, 2010. 3

[38] L. Wan, D. Eigen, and R. Fergus. End-to-end integration of a convolutional network, deformable parts model and non-maximum suppression. *arXiv preprint arXiv:1411.5309*, 2014. 2

[39] J. Wang and A. Yuille. Semantic part segmentation using compositional model combining shape and appearance. In *CVPR*, 2015. 1, 2, 5, 6, 7

[40] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. Yuille. Towards unified depth and semantic prediction from a single image. In *CVPR*, 2015. 2

[41] K. Yamaguchi, M. H. Kiapour, L. E. Ortiz, and T. L. Berg. Parsing clothing in fashion photographs. In *CVPR*, pages 3570–3577, 2012. 2

[42] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, pages 1385–1392, 2011. 1

[43] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833, 2014. 2

[44] N. Zhang, J. Donahue, R. B. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*, pages 834–849, 2014. 1

[45] L. Zhu, Y. Chen, C. Lin, and A. L. Yuille. Max margin learning of hierarchical configural deformable templates (hcdts) for efficient object parsing and pose estimation. *IJCV*, 93(1):1–21, 2011. 1, 2

[46] L. Zhu, Y. Chen, A. Torralba, W. T. Freeman, and A. L. Yuille. Part and appearance sharing: Recursive compositional models for multi-view. In *CVPR*, pages 1919–1926, 2010. 2, 3

[47] Y. Zhu, R. Urtasun, R. Salakhutdinov, and S. Fidler. segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *CVPR*, 2015. 2