HAL
open science

Scene-Domain Active Part Models for Object Representation

Zhou Ren, Chaohui Wang, Alan Yuille

To cite this version:

Zhou Ren, Chaohui Wang, Alan Yuille. Scene-Domain Active Part Models for Object Representation. IEEE International Conference on Computer Vision (ICCV), Dec 2015, Santiago, Chile. pp.2497 - 2505, 10.1109/ICCV.2015.287 . hal-01578411

HAL Id: hal-01578411

https://hal.science/hal-01578411v1

Submitted on 29 Aug 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Scene-Domain Active Part Models for Object Representation

Zhou Ren[1] Chaohui Wang[2] Alan Yuille[1]
[1]University of California, Los Angeles [2]Université Paris-Est, LIGM - CNRS UMR 8049
zhou.ren@cs.ucla.edu chaohui.wang@u-pem.fr yuille@stat.ucla.edu

Abstract

In this paper, we are interested in enhancing the expressivity and robustness of part-based models for object representation, in the common scenario where the training data are based on 2D images. To this end, we propose scene-domain active part models (SDAPM), which reconstruct and characterize the 3D geometric statistics between object's parts in 3D scene-domain by using 2D training data in the image-domain alone. And on top of this, we explicitly model and handle occlusions in SDAPM. Together with the developed learning and inference algorithms, such a model provides rich object descriptions, including 2D object and parts localization, 3D landmark shape and camera viewpoint, which offers an effective representation to various image understanding tasks, such as object and parts detection, 3D landmark shape and viewpoint estimation from images. Experiments on the above tasks show that SDAPM outperforms previous part-based models, and thus demonstrates the potential of the proposed technique.

1. Introduction

Object representation is a key problem in computer vision. Coarse-grained representation, such as detecting objects by bounding boxes [12], is important for tasks such as object tracking [5] and scene understanding [21]. For example, deep learning approaches [19, 42], based on Convolutional Neural Networks, have validated their ability to extract strong image features and obtain such coarse-grained representation. Moreover, fine-grained representation, such as locating object parts in 2D image-domain and 3D scene-domain, is helpful for further applications such as action analysis [40] and human-computer interaction [31, 32]. For example, part-based models [3, 41] have demonstrated elegant performance in obtaining fine-grained representation.

In this paper, we are interested in enhancing part-based models for fine-grained object representation. Although various part-based object models have been developed in the literature (*e.g.*, [3, 6, 13, 35, 41]) and demonstrated elegant performance in obtaining fine-grained object representation, it is still far from satisfactory to robustly represent generic objects under significant "geometric variations" (this term refers to camera viewpoint changes, non-rigid deformations, intra-class variations, and occlusions in this paper). We observe that one main reason for this arises from the fact that in most of the existing applications of generic objects, the available training data are based on 2D images[1]. In such a scenario, it is natural that one resorts to modeling objects in the 2D image-domain (*e.g.*, by a collection of 2D parts deforming around the corresponding part anchor positions) and the 3D information is discarded.

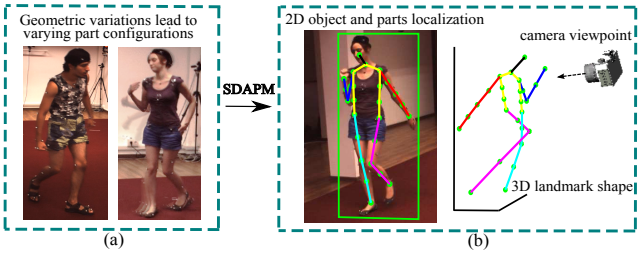

Figure 1. (a) A motivation of SDAPM: geometric variations lead to varying part configurations. Thus, by modeling in the scene-domain, SDAPM can better capture objects' geometric statistics and provides richer object descriptions, including 2D parts localization, 3D landmark shape as well as camera viewpoint estimation, in addition to the 2D object bounding box, as shown in (b).

However, by modeling part deformations in the 2D image-domain, it is actually difficult to well-capture the important statistics on geometric properties of an object, due to the fact that those "geometric variations" can cause very complicated 2D variations (because of the 3D-2D projection) and make such geometric properties highly complex to be described. For instance, human arms can be foreshortened to varying sizes in different viewpoints. Furthermore, a richer description of an object in the 3D scene-domain, is increasingly in demand for further applications such as action detection, scene understanding, *etc*. For instance, it is beneficial to have the 3D part localization and camera viewpoint for scene understanding.

Accordingly, we are particularly interested in proposing such an even finer-grained object representation that

[1]Only for some specific objects such as human body, human hand, car, bed, *etc*., their models have been built in 3D by learning from CAD data or depth data (*e.g.*, [14, 26]), due to the data availability.

can better capture objects' geometric statistics and provides richer object representations in 3D. One important observation is that if we can characterize and learn such statistics directly from the 3D scene-domain, we will be able to remove the viewpoint and non-rigid variations, and also obtain the 3D representation of objects. Moreover, recent progress in non-rigid structure-from-motion techniques [1, 9] provides an effective way to learn such 3D geometric statistics from 2D images. This motivates us to develop a part-based object model that characterizes the geometric variations directly in 3D scene-domain by using 2D training data alone.

Hence, in this paper, we present Scene-Domain Active Parts Models (SDAPM), as shown in Fig. 1. Our approach reconstructs and characterizes the 3D geometric statistics between object parts in the scene-domain by learning from 2D training data in the image-domain. And on top of this, we model such statistics together with the local appearance with occlusions. The main contributions of this paper are two-fold: firstly, we propose a compact and robust part-based representation for objects under geometric variations, *e.g.*, viewpoint changes, non-rigid deformations, and occlusions, by modeling active parts in the 3D scene-domain; and secondly, our method provides a fine-grained representation of object, including 2D object and parts localization, 3D landmark shape and camera viewpoint estimation.

We have conducted experiments on various tasks. In the task of object and parts detection on PASCAL VOC 2010 dataset, our method boosts the performance of previous part-based models, *e.g.*, [3, 4, 12, 15, 28, 41], with three different types of features: HOG feature, Segmentation feature, and Convolutional Neural Networks (CNN) feature. In the task of 2D, 3D landmark shape and camera viewpoint estimation on Human3.6M dataset and PASCAL VOC 2007 car dataset, our method outperforms state-of-the-art methods [2, 20] that are also learned from 2D training data alone.

2. Related work

In the common scenario where the training data are based on 2D images, existing object representation methods usually resort to modeling objects directly in the 2D image-domain. Two main strategies have been adopted.

One line of work aims at providing coarse-grained representation, *i.e.*, detecting the objects with bounding boxes. Some work focus on developing more effective modeling and inference schemes. For example, Felzenszwalb *et al.* [12] proposed the elegant Latent SVM framework. Bourdev *et al.* proposed the Poselets model [4]. Some other work focus on developing stronger image features. For instance, Dalal *et al.* [10] presented HOG descriptors for object representation. Chen *et al.* [8] proposed to combine Bag-of-Features with HOG features in a Latent SVM framework. Image segmentation and region-level cues have also been explored for objects detection (*e.g.*, [6, 15, 28, 36, 39]). Recently, deep learning approaches, based on Convolutional Neural Networks (CNN), have validated their ability to extract strong image features and achieved state-of-the-art performance on the task of image classification [24, 25] and object detection [19, 42]. However, those deep learning approaches do not explicitly model object composition, thus throwing away useful fine-grained high-level part relationships for further applications.

Another line of work proposes to represent objects with fine-grained representation, *e.g.*, the localization of parts. Azizpour and Laptev [3] proposed strongly supervised paradigm for part-based models to locate object parts. Sun and Savarese [35] proposed a coarse-to-fine structure for joint object detection and pose estimation. Yang and Ramanan [41] presented the mixture-of-parts structure for pose estimation. Chen *et al.* [6] proposed to model objects with complete graph structure. And grammar models have also been explored in [17]. Various methods have utilized the CNN feature in part-based models in order to leverage the discriminative power of CNN feature and the fine-grained modeling of part-based models [18, 33, 38].

Despite the potential shown already, these 2D models cannot well-capture the important statistics on the geometric properties of objects under significant viewpoint changes, non-rigid deformations and occlusions, which substantially limits the performance of such models. Since it is useful to model objects in the 3D scene-domain, various 3D part-based models have been developed by modeling the 3D properties of an object directly in the 3D scene-domain using 3D training data. For instance, Fidler *et al.* [14] and Shrivastava *et al.* [34] proposed 3D deformable part models for object detection which used depth data for training. 3D CAD data were utilized in [26, 30] to learn object models for detection and pose estimation. However, most 3D part-based models necessitate 3D data for training, which restrict the use of such models only to a small number of specific objects such as human body, hand, motorcycle, bed, *etc*.

In this paper, we propose to model objects in the scene-domain by using 2D training data alone. In addition to the 2D representation provided by coarse-grained models and previous fine-grained models, our method provides additional finer-grained representation in 3D, including 2D object and parts localization, 3D landmark shape and camera viewpoint. There exist several works which have similar setting as ours, *e.g.*, the methods proposed by Hejrati *et al.* [20] and Nachimson *et al.* [2] can obtain 2D and 3D object representation from 2D images. However, the major difference between [2, 20] and our method is that they first model objects in the image-domain and then reconstruct the 3D landmark shape, via two separate stages, while our method models objects directly in the 3D scene-domain and the 3D landmark shape is recovered via a unified process.

3. Scene-Domain Active Part Models

We start the presentation of our model with a preliminary on 2D part-based models, which have been widely and successfully applied in fine-grained object representation.

3.1. Preliminary: 2D Part-based Object Models

2D part-based models are a category of object models where an object (category) is represented by a set of 2D parts and each part is allowed to deform around its anchor position in the 2D image-domain, which can date back to the original idea of Fischler and Elschlager [16]. For simplicity, we introduce our model at a fixed scale; at test time we handle object of different sizes by searching over an image pyramid. Let $\mathbf{I}$ denote an image, $\mathcal{V}$ be the part set in a part-based model, and p_i denote a candidate location[2] of part i in the image-domain. For a part hypothesis $\mathbf{p} = \{p_i\}_{i\in\mathcal{V}}$ in an image $\mathbf{I}$, the score function of 2D part-based models can be expressed as:

$$S(\mathbf{I}, \mathbf{p}) = \sum_{i\in\mathcal{V}} S_i(\mathbf{I}, p_i) + \sum_{ij\in\mathcal{E}} S_{ij}(p_i, p_j), \tag{1}$$

where $\mathcal{G}{=}(\mathcal{V}, \mathcal{E})$ is the tree-structure relational graph whose node set is the part set $\mathcal{V}$ of the model, and the edge set $\mathcal{E}$ specifies the pairs of parts between which certain geometrical constraints are imposed. $S_i(\mathbf{I}, p_i)$ is the unary term corresponding to the local appearance score for placing the i-th part template at location p_i. And $S_{ij}(p_i, p_j)$ is the pairwise term that penalizes the displacement of the i-th and j-th parts according to some prior model (*e.g.*, the deviation from their anchor position μ_{ij}).

In order to better represent generic objects, various part-based models have been proposed by enhancing the unary term $S_i(\mathbf{I}, p_i)$. For instance, $S_i(\mathbf{I}, p_i)$ is defined as $\boldsymbol{\alpha}_i \cdot \boldsymbol{\phi}(\mathbf{I}, p_i) + b_i$ in [12] for object detection, where $\boldsymbol{\alpha}_i$ is the template parameter of part i, $\boldsymbol{\phi}(\mathbf{I}, p_i)$ is the image feature extracted at location p_i, and b_i is a bias term. In pose estimation, the idea of mixture of parts was adopted in [41], by defining $S_i(\mathbf{I}, p_i, t_i) = \boldsymbol{\alpha}_i^{t_i} \cdot \boldsymbol{\phi}(\mathbf{I}, p_i) + b_i^{t_i}$ where $\boldsymbol{\alpha}_i^{t_i}$ is the template parameter of the i-th part of type t_i, and $b_i^{t_i}$ is the bias term that favors the part type assignment in the relational graph $\mathcal{G}$.

Regarding the pairwise term $S_{ij}(p_i, p_j)$, the following form has been commonly used in previous works[3]: $S_{ij}(p_i, p_j) = \boldsymbol{\beta}_{ij} \cdot \boldsymbol{\psi}(p_i, p_j)$, where $\boldsymbol{\psi}(p_i, p_j)$ is a four-dimensional vector defining the pairwise displacement between part i and part j relative to their anchor position μ_{ij}, *i.e.*, $\boldsymbol{\psi}(p_i, p_j) = (dx_{ij}, dy_{ij}, dx_{ij}^2, dy_{ij}^2)^T$ where $(dx_{ij}, dy_{ij}) = p_i - p_j - \mu_{ij}$, dx_{ij}^2 is a simplified form of $(dx_{ij})^2$, and $\boldsymbol{\beta}_{ij} = (\beta_{ij}^a, \beta_{ij}^b, \beta_{ij}^c, \beta_{ij}^d)$ is the model parameter. Accordingly, the summation of pairwise terms for all edges in $\mathcal{E}$ in Eq. 1 is as follows:

$$\begin{aligned}
&\sum_{ij\in\mathcal{E}} S_{ij}(p_i, p_j) = \sum_{ij\in\mathcal{E}} (\beta_{ij}^a, \beta_{ij}^b, \beta_{ij}^c, \beta_{ij}^d) \cdot (dx_{ij}, dy_{ij}, dx_{ij}^2, dy_{ij}^2)^T \\
&= (\tilde{\beta}_1^a, \tilde{\beta}_2^a, ..., \tilde{\beta}_{|\mathcal{E}|}^a, \tilde{\beta}_1^b, \tilde{\beta}_2^b, ..., \tilde{\beta}_{|\mathcal{E}|}^b) \cdot (d\tilde{x}_1, d\tilde{x}_2, ..., d\tilde{x}_{|\mathcal{E}|}, d\tilde{y}_1, d\tilde{y}_2, ..., d\tilde{y}_{|\mathcal{E}|})^T \\
&+ \begin{pmatrix} d\tilde{x}_1 \\ d\tilde{x}_2 \\ ... \\ d\tilde{x}_{|\mathcal{E}|} \\ d\tilde{y}_1 \\ d\tilde{y}_2 \\ ... \\ d\tilde{y}_{|\mathcal{E}|} \end{pmatrix}^T \cdot \begin{pmatrix} \tilde{\beta}_1^c & & & & & & & \\ & \tilde{\beta}_2^c & & & & & & \\ & & ... & & & & & \\ & & & \tilde{\beta}_{|\mathcal{E}|}^c & & & & \\ & & & & \tilde{\beta}_1^d & & & \\ & & & & & \tilde{\beta}_2^d & & \\ & & & & & & ... & \\ & & & & & & & \tilde{\beta}_{|\mathcal{E}|}^d \end{pmatrix} \cdot \begin{pmatrix} d\tilde{x}_1 \\ d\tilde{x}_2 \\ ... \\ d\tilde{x}_{|\mathcal{E}|} \\ d\tilde{y}_1 \\ d\tilde{y}_2 \\ ... \\ d\tilde{y}_{|\mathcal{E}|} \end{pmatrix},
\end{aligned}$$

where linear indexing is adopted so as to obtain a form based on matrix operations, *e.g.*, $\tilde{\beta}_l^a$ denotes an element in $\{\tilde{\beta}_l^a, l = 1...|\mathcal{E}|\}$ where each element corresponds to the parameter of one edge β_{ij}^a in $\mathcal{E}$ (same concept for all symbols with tilde in this paper).

In order to achieve a compact formulation for later presentation, let $\boldsymbol{\tau}$ denote the parameter vector of size $2|\mathcal{E}| \times 1$: $(\tilde{\beta}_1^a, \tilde{\beta}_2^a, ..., \tilde{\beta}_{|\mathcal{E}|}^a, \tilde{\beta}_1^b, \tilde{\beta}_2^b, ..., \tilde{\beta}_{|\mathcal{E}|}^b)$, and $\boldsymbol{\Lambda}$ denote the diagonal parameter matrix of size $2|\mathcal{E}| \times 2|\mathcal{E}|$: $\text{diag}(\tilde{\beta}_1^c, \tilde{\beta}_2^c, ..., \tilde{\beta}_{|\mathcal{E}|}^c, \tilde{\beta}_1^d, \tilde{\beta}_2^d, ..., \tilde{\beta}_{|\mathcal{E}|}^d)$. Moreover, we let $\boldsymbol{\mu}_{(2|\mathcal{E}|\times 1)} = (\tilde{\mu}_1, \tilde{\mu}_2, ..., \tilde{\mu}_{|\mathcal{E}|})^T$ denote the pairwise anchor position vector of the model, and $\boldsymbol{\Delta}\mathbf{p}_{(2|\mathcal{E}|\times 1)}$ denote the corresponding inter-part distance vector for a specific hypothesis configuration $\mathbf{p}$ of the model. Accordingly, the displacement deviation vector $(d\tilde{x}_1, d\tilde{x}_2, ..., d\tilde{x}_{|\mathcal{E}|}, d\tilde{y}_1, d\tilde{y}_2, ..., d\tilde{y}_{|\mathcal{E}|})^T$ can be denoted as $\boldsymbol{\Delta}\mathbf{p} - \boldsymbol{\mu}$. These lead to the following compact formulation for the summation of pairwise terms:

$$\sum_{ij\in\mathcal{E}} S_{ij}(p_i, p_j) = (\boldsymbol{\Delta}\mathbf{p} - \boldsymbol{\mu})^T \cdot \boldsymbol{\tau} + (\boldsymbol{\Delta}\mathbf{p} - \boldsymbol{\mu})^T \cdot \boldsymbol{\Lambda} \cdot (\boldsymbol{\Delta}\mathbf{p} - \boldsymbol{\mu}) \,. \tag{2}$$

As we see, 2D part-based models represent objects in the image-domain, by allowing the parts deform around the image-domain anchor positions $\boldsymbol{\mu}$.

3.2. Modeling Active Parts in the 3D Scene-Domain

It is difficult to model an object's part configuration in the 2D image-domain, because, for a non-rigid object, the part locations in the image-domain after the 3D-2D projection from different viewpoints can have very different configurations, thus setting anchor positions in a model cannot well-capture the geometric properties of objects. To remove such geometric variations, we introduce the way we model the parts of an object in the 3D scene-domain. To this end, we make the following two assumptions which were often made in the literature: (1) the depth variation of objects are small compared to the distance from the camera, which enables the adoption of the weak-perspective projection model; (2) the 3D configuration of an object's parts can be written as linear combinations of a few basis shapes.

[2] For clarity, here we focus on the case where the parts are parametrized by their 2D locations. However, more complex parametrizations of the geometric configuration of the parts can be considered.

[3] This term can be further extended to enrich the model. For example, a part-type-specific term was adopted in [41] to handle part types.

Under the weak perspective projection model, for an object with $|\mathcal{E}|$ pairs of parts, its inter-part distances in the image-domain, $\mathbf{w}_{(2\times|\mathcal{E}|)}$, is the projection from the part landmark shape in the 3D scene-domain, $\mathbf{S}_{(3\times|\mathcal{E}|)}$, to the image-domain, *i.e.*, $\mathbf{w} = \mathbf{R}\cdot\mathbf{S}+\mathbf{t}$, where $\mathbf{R}_{(2\times 3)}$ is the rotation matrix and $\mathbf{t}_{(2\times|\mathcal{E}|)}$ is the translation matrix [27].

Inspired by the non-rigid structure-from-motion techniques [1, 9], we propose to model an object's part configuration directly in the 3D scene-domain by characterizing the 3D inter-part shape $\mathbf{S}$ as a subspace, which is represented as weighted combinations of K bases $\{\mathbf{B}_k, k=1,...,K\}$, *i.e.*, $\mathbf{S} = \sum_{k=1}^{K} c_k\mathbf{B}_k$ where each base $\mathbf{B}_k$ is a $3\times|\mathcal{E}|$ matrix (note that there are constraints on $\{\mathbf{B}_k\}$ that are imposed to upgrade $\{\mathbf{B}_k\}$ from affine space to Euclidean space. See [1, 9] for details). Thus, the inter-part configuration in the image-domain $\mathbf{w}$ can be formulated as follows:

$$\mathbf{w} = \mathbf{R}\cdot\sum_{k=1}^{K} c_k\mathbf{B}_k + \mathbf{t} = \mathbf{R}\cdot\mathbf{c}^T\cdot\begin{pmatrix}\mathbf{B}_1\\ ...\\ \mathbf{B}_K\end{pmatrix}+\mathbf{t}, \quad (3)$$

where $\mathbf{c} = (c_1, c_2, ..., c_K)^T$ are the weight vector.

Let us denote the reconstruction matrix as $\mathbf{m}_{(2\times 3K)} = \mathbf{R}\cdot\mathbf{c}^T$ and the 3D geometric subspace as $\mathbf{B}_{(3K\times|\mathcal{E}|)} = (\mathbf{B}_1;\mathbf{B}_2;...;\mathbf{B}_K)$. By translating to the object hypothesis center such that the centroid $\mathbf{t}$ is cancelled out, we have $\mathbf{w} = \mathbf{mB}$. In this way, we can configure an object's parts in the image-domain from the subspace spanned by $\mathbf{B}$, and allow all parts to deform in the scene-domain rather than fixing them in anchor positions.

More specifically, in previous part-based models, as shown in Eq. 2, the inter-part distance vector $\mathbf{\Delta p}$ is constrained to move around a fixed anchor positions $\boldsymbol{\mu}$, with a penalization on the displacement vector $(\mathbf{\Delta p}-\boldsymbol{\mu})$ in Gaussian fashion. However, in our scene-domain active parts model, we do not associate any of the pairwise parts with an fixed anchor position. Instead, we define our part's anchor configuration in the image-domain as a projection from the 3D scene-domain configuration, which is constructed from the subspace $\mathbf{B}$. Thus, our displacement vector is defined as $\mathbf{\Delta p} - f(\mathbf{w})$, *i.e.*, the difference between the inter-parts distance $\mathbf{\Delta p}$ of the hypothesis and the projected part configuration from the scene-domain.

Here, $f(\mathbf{w})$ is a transformation function from the $2\times|\mathcal{E}|$ image-domain configuration matrix $\mathbf{w} = (x_1,\ x_2, ..., x_{|\mathcal{E}|}; y_1, y_2, ..., y_{|\mathcal{E}|})$ to a $2|\mathcal{E}|\times 1$ vector form $(x_1, x_2, ..., x_{|\mathcal{E}|}, y_1, y_2, ..., y_{|\mathcal{E}|})^T$, namely, $f(\mathbf{w}) = (\mathbf{e}_1\mathbf{wA}+\mathbf{e}_2\mathbf{w}\widehat{\mathbf{A}})^T$ where $\mathbf{e}_1, \mathbf{e}_2, \mathbf{A}, \widehat{\mathbf{A}}$ are constants. $\mathbf{e}_1 = (1,0)$, $\mathbf{e}_2 = (0,1)$. $\mathbf{A}$ and $\widehat{\mathbf{A}}$ are $|\mathcal{E}|\times 2|\mathcal{E}|$ matrices. $\mathbf{A} = (\mathbf{I}_{|\mathcal{E}|}, \mathbf{0})$, and $\widehat{\mathbf{A}} = (\mathbf{0}, \mathbf{I}_{|\mathcal{E}|})$ where $\mathbf{I}_{|\mathcal{E}|}$ is the $|\mathcal{E}|\times|\mathcal{E}|$ identity matrix, and $\mathbf{0}$ is the $|\mathcal{E}|\times|\mathcal{E}|$ zero matrix. Therefore, with $\mathbf{w} = \mathbf{mB}$, the score function of our model is:

$$\begin{aligned} S(\mathbf{I},\mathbf{p},\mathbf{m}) &= \sum_{i\in\mathcal{V}} S_i(\mathbf{I},p_i) + (\mathbf{\Delta p} - f(\mathbf{mB}))^T\cdot\boldsymbol{\tau} \\ &\quad + (\mathbf{\Delta p} - f(\mathbf{mB}))^T\cdot\mathbf{\Lambda}\cdot(\mathbf{\Delta p} - f(\mathbf{mB})), \quad (4)\end{aligned}$$

where $f(\mathbf{mB}) = \left(\mathbf{e}_1\mathbf{mBA}+\mathbf{e}_2\mathbf{mB}\widehat{\mathbf{A}}\right)^T$.

3.3. Modeling Appearance with Occlusions

As discussed in Section 3.1, appearance information is usually encoded within the unary term[4] of part-based models (*e.g.*, [12] models appearance by $\boldsymbol{\alpha}_i\cdot\boldsymbol{\phi}(\mathbf{I},p_i)$). Occlusions frequently occur when a non-rigid object is projected from the 3D scene-domain to the 2D image-domain, either because of self-occlusions or occluded by other objects. In order to handle occlusions, we define a binary occlusion state o_i for each part, and a visible-state template $\boldsymbol{\alpha}_i^v$ when $o_i = 0$ as well as an occluded-state template $\boldsymbol{\alpha}_i^o$ when $o_i = 1$. Accordingly, our unary term in Eq. 4 is defined as:

$$S_i(\mathbf{I},p_i) = \max_{o_i}((1-o_i)(\boldsymbol{\alpha}_i^v\cdot\boldsymbol{\phi}(\mathbf{I},p_i)), o_i\boldsymbol{\alpha}_i^o\cdot\boldsymbol{\phi}(\mathbf{I},p_i)) + b_i. \quad (5)$$

To summarize, we can formally define our model as $(\mathbf{B}, \boldsymbol{\tau}, \mathbf{\Lambda}, \{\boldsymbol{\alpha}_i^v\}, \{\boldsymbol{\alpha}_i^o\}, \{b_i\})$, where $\mathbf{B}$ denotes the scene-domain geometric subspace for 3D part configuration, $\boldsymbol{\tau}$ and $\mathbf{\Lambda}$ are the deformation parameter matrices, $\{\boldsymbol{\alpha}_i^v\}$, $\{\boldsymbol{\alpha}_i^o\}$ are the unary term parameters, and $\{b_i\}$ are the bias terms.

4. Inference

Our method can obtain rich object representation, including 2D object and parts localization $\mathbf{p}$, 3D landmark shape $\mathbf{S}$, and camera viewpoint $\mathbf{R}$. To infer them, we maximize $S(\mathbf{I},\mathbf{p},\mathbf{m})$ in Eq. 4 over the part hypothesis $\mathbf{p}$ and reconstruction matrix $\mathbf{m}$ using a coordinate descent approach:

1). *Optimize over* $\mathbf{p}$: Fix $\mathbf{m}$, maximize $S(\mathbf{I},\mathbf{p},\mathbf{m})$ over $\mathbf{p}$, using Dynamic Programming;

2). *Optimize over* $\mathbf{m}$: Fix $\mathbf{p}$, compute the close-form solution $\mathbf{m}^*$ that maximizes $S(\mathbf{I},\mathbf{p},\mathbf{m})$.

Both steps are executed alternatively until convergence.

By fixing $\mathbf{m}$, step 1 is essentially equivalent to the traditional part-based model inference procedure. We follow [12] and use dynamic programming to obtain the optimal $\mathbf{p}^*$. We initialize $\mathbf{m}$ to be the part anchor positions of [3]. And in step 2, we have a closed-form solution $\mathbf{m}^*$ that maximizes $S(\mathbf{I},\mathbf{p},\mathbf{m})$ (see the supplementary materials for the derivation procedure):

$$\begin{aligned}\mathbf{m}_1^* &= \mathbf{H}_1\left(\mathbf{BA}(\mathbf{\Lambda}+\mathbf{\Lambda}^T)\mathbf{A}^T\mathbf{B}^T\right)^{-1}, \\ \mathbf{m}_2^* &= \mathbf{H}_2\left(\mathbf{B}\widehat{\mathbf{A}}(\mathbf{\Lambda}+\mathbf{\Lambda}^T)\widehat{\mathbf{A}}^T\mathbf{B}^T\right)^{-1}, \quad (6)\end{aligned}$$

[4]Here we present our method based on [12] in object detection. However, It is similar to apply our occlusion modeling method to models in pose estimation, *e.g.* [41].

where $\mathbf{m}_1^*, \mathbf{m}_2^*$ are the first and second rows of the optimal solution $\mathbf{m}^*_{(2\times 3K)}$. $\mathbf{B}$ is the scene-domain geometric subspace, $\mathbf{H}_1$ and $\mathbf{H}_2$ are the first and second rows of matrix $\mathbf{H}_{(2\times 3K)}$ which is defined as follows:

$$\mathbf{H}=\mathbf{e}_1^T\Big((\boldsymbol{\Lambda}+\boldsymbol{\Lambda}^T)\boldsymbol{\Delta}\mathbf{p}+\boldsymbol{\tau}\Big)^T\mathbf{A}^T\mathbf{B}^T+\mathbf{e}_2^T\Big((\boldsymbol{\Lambda}+\boldsymbol{\Lambda}^T)\boldsymbol{\Delta}\mathbf{p}+\boldsymbol{\tau}\Big)^T\widehat{\mathbf{A}}^T\mathbf{B}^T,$$

where $\boldsymbol{\tau}$ and $\boldsymbol{\Lambda}$ are the deformation parameter matrices; $\boldsymbol{\Delta}\mathbf{p}$ is the inter-parts distance vector of a hypothesis; $\mathbf{e}_1,\ \mathbf{e}_2,\ \mathbf{A}$, $\widehat{\mathbf{A}}$ are the constant matrices defined in Section 3.2.

4.1. 3D Landmark Shape and Viewpoint Recovery

After we obtain the detected $\mathbf{p}^*$ described above, now we introduce how we recover the 3D landmark shape and viewpoint of the detected object. As discussed in Section 3.2, $\mathbf{w} = \mathbf{R}\cdot\mathbf{S}$ and the 3D landmark shape $\mathbf{S} = \mathbf{c}^T\cdot\mathbf{B}$. Given the detection $\mathbf{p}^*$ in 2D image-domain and the known geometric subspace $\mathbf{B}$, we recover the 3D landmark shape and viewpoint as follows:

$$\mathbf{R}^*, \mathbf{c}^* = \min_{\mathbf{R},\mathbf{c}} ||\boldsymbol{\Delta}\mathbf{p}^* - f(\mathbf{R}\cdot\mathbf{c}^T\cdot\mathbf{B})||^2 \tag{7}$$

where $\boldsymbol{\Delta}\mathbf{p}^*_{(2|\mathcal{E}|\times 1)}$ is the inter-part distance vector obtained from $\mathbf{p}^*$, and $f(\cdot)$ is the transformation function used in Eq. 4. Because the squared error is linear in $\mathbf{R}$ and $\mathbf{c}$, we obtain the optimal $\mathbf{R}^*$, $\mathbf{c}^*$ of Eq. 7 with iterative least-squares algorithm [20]. Then, we obtain the 3D landmark shape $\mathbf{S}^* = \mathbf{c}^{*T}\cdot\mathbf{B}$, and the viewpoint $\mathbf{R}^*$ of the detected object.

5. Learning

Our training data consists of a set of positive training examples $\mathbf{x}_n = \{\mathbf{I}_n, \mathbf{p}_n, \mathbf{o}_n\}$, negative training examples $\mathbf{x}_n = \{\mathbf{I}_n\}$, and corresponding example label y_n, $n \in \{1, ..., N\}$, where N is the total number. $\mathbf{I}_n$ is the image with object bounding boxes, $\mathbf{p}_n$ is the part bounding boxes in $\mathbf{I}_n$, and $\mathbf{o}_n$ is the parts' occlusion states in $\mathbf{I}_n$. We learn the model parameter in two steps: firstly we learn the 2D image-domain parameters $\boldsymbol{\Theta} = (\boldsymbol{\tau}, \boldsymbol{\Lambda}, \{\boldsymbol{\alpha}_i^v\}, \{\boldsymbol{\alpha}_i^o\}, \{b_i\})$, then we learn the 3D scene-domain geometric subspace $\mathbf{B}$.

5.1. Learning the 2D Image-Domain Parameter $\boldsymbol{\Theta}$

We learn the image-domain model parameters in a discriminative way by minimizing the loss function $L(\boldsymbol{\Theta}) = \frac{1}{2}||\boldsymbol{\Theta}||^2 + C\sum_{n=1}^{N}\max(0, 1 - y_n S_{\boldsymbol{\Theta}}(\mathbf{x}_n))$.

We follow the strongly supervised learning paradigm in [3] to learn $\boldsymbol{\Theta}$. In order to reduce the influence of the imprecise part annotation in the training data and the possibly low discriminative power of some annotated parts, we allow our part models to approximately overlap with the training part bounding boxes in positive images. This is achieved by constraining the searching space $\mathbf{p}$ of the score function to be $\mathbf{Z}_{\mathbf{p}}(\mathbf{x}_n)$ that is consistent with the annotation $\mathbf{p}_n$:

$$S_{\boldsymbol{\Theta}}(\mathbf{x}_n) = \max_{\mathbf{p}\in\mathbf{Z}_{\mathbf{p}}(\mathbf{x}_n)} S(\mathbf{I}, \mathbf{p}), \tag{8}$$

where $\mathbf{Z}_{\mathbf{p}}(\mathbf{x}_n) = \begin{cases}\{\mathbf{p}\in\mathbb{P}|O(\mathbf{p},\mathbf{p}_n) > t_{ovp}\} & \text{if } \mathbf{p}_n \text{ available,}\\ \mathbb{P} & \text{otherwise.}\end{cases}$

$\mathbb{P}$ is the set of all possible part bounding boxes, and $O(\cdot,\cdot)$ is the intersection over union (IoU) measure of two bounding boxes, we set $t_{ovp} = 0.5$ in our experiments.

5.2. Learning the 3D Geometric Subspace $\mathbf{B}$

Given training data with labeled 2D part locations $\{\mathbf{p}_n\}$, we can learn the scene-domain geometric subspace $\mathbf{B}$ by casting this as non-rigid structure-from-motion (NRSFM) problem. As shown in Eq. 3, the inter-part distances in the image-domain, $\mathbf{w}_{(2\times|\mathcal{E}|)}$, is related to the 3D inter-part distances in the scene-domain, $\mathbf{S}_{(3\times|\mathcal{E}|)}$, via a weak perspective projection model. Given N positive training examples of a certain object category which shares the same 3D part configuration subspace $\mathbf{B}$, we have $\mathbf{W} = \mathbf{MB} + \mathbf{T}$, where $\mathbf{W}_{(2N\times|\mathcal{E}|)} = (\mathbf{w}_1; \mathbf{w}_2; ...; \mathbf{w}_N)$ is the 2D inter-part distance matrix in N images, $\mathbf{B}_{(3K\times|\mathcal{E}|)}$ is the scene-domain geometric subspace shared among the same object category, $\mathbf{M}_{(2N\times 3K)} = (\mathbf{m}_1; ...; \mathbf{m}_N)$ is the reconstruction coefficient matrix, and $\mathbf{T}_{(2N\times|\mathcal{E}|)}$ is the translation matrix. Given the 2D part locations $\mathbf{W}$ obtained from $\{\mathbf{p}_n\}$, we use the publically-available NRSFM code [1, 9] to learn $\mathbf{B}$.

6. Experiments

Since the proposed method provides fine-grained object representation both in 2D and 3D, we conduct two sets of experiments to evaluate it. The first set of experiments evaluates our method on 2D object and parts detection, and compares with 2D part-based models [3, 4, 12, 18, 29, 39]. The second set tests on 3D landmark shape and viewpoint estimation, and compares our method with 3D part-based models also learned from 2D data alone [2, 20].

6.1. Datasets

The experiments are based on three challenging datasets: the PASCAL VOC 2010 dataset [11] for 2D object and parts detection, the Human3.6M dataset [22] for 2D and 3D landmark shape estimation, and the PASCAL VOC 2007 car dataset [2] for viewpoint classification. For the PASCAL VOC 2010 dataset, following [3, 6], we evaluate on the six animal classes. These animal classes serve as a common testbed for object model evaluation (*e.g.*, in [3, 6]), because of the high difficulty in addressing them caused by highly non-rigid deformations, intra-class variations, and different degrees of occlusions. In addition to the part annotation provided in [3] which is used in our method, we further annotate the locations of occluded parts. We use `trainval` subset of PASCAL VOC 2010 for training and the `test` subset for testing. Meanwhile, the Human3.6M dataset provides both 2D and 3D landmark annotations, and serves as a suitable testbed to evaluate 2D and 3D pose landmark shape

	Bird	Cat	Cow	Dog	Horse	Sheep	mAP
Ours w. HOG	**15.3**	**28.6**	**28.7**	**28.2**	**48.3**	**30.1**	**29.9**
SSDPM [3]	11.3	27.2	25.8	23.7	46.1	28.0	27.0
Poselets [4]	8.5	22.2	20.6	18.5	48.2	28.0	24.3
DPM [12]	11.0	23.6	23.2	20.5	42.5	29.0	25.0
Ours w. Seg & HOG	**26.1**	**51.2**	**35.3**	**41.7**	**52.8**	37.5	**40.8**
Regionlets [39]	25.9	**51.2**	28.9	35.8	40.2	**43.9**	37.65
DefPM [29]	-	45.3	-	36.8	-	-	-
SegDPM [15]	25.3	48.8	30.4	37.7	46.0	35.7	37.3
Ours w. CNN	**38.9**	**48.5**	**38.8**	**47.5**	**55.0**	48.3	**46.2**
DP-DPM [18]	36.5	48.0	35.0	45.7	50.2	**49.1**	44.1

Table 1. Average precision for animal detection on PASCAL VOC 2010. Our method outperforms all baselines of part-based models.

	Head	Fore legs	Hind legs	Torso/Back	Tail
Bird	**50.7** / 28.1	-	**15.2** / 12.5	-	**35.0** / 20.7
Cat	**71.1** / 62.8	**18.9** / 11.4	-	**50.3** / 37.2	**18.1** / 10.1
Cow	**72.8** / 56.2	**85.7** / 60.9	**80.3** / 58.1	**77.2** / 69.3	-
Dog	**59.0** / 48.7	33.6 / **37.5**	-	**34.5** / 21.6	**30.0** / 9.7
Horse	65.9 / **67.1**	**82.2** / 53.1	**80.6** / 55.7	**88.5** / 67.4	**68.2** / 42.9
Sheep	**58.3** / 41.4	**67.4** / 43.8	**65.8** / 39.7	**83.7** / 71.1	**36.1** / 12.7

Table 2. Part localization performance on PASCAL VOC 2010. The numbers are "PCP of our method" / "PCP of SSDPM [3]".

estimation of our method. We use the subject `S1` of walking action for training and `S7` for testing. Lastly, we test viewpoint estimation on the PASCAL VOC 2007 car dataset [2], which consists of 200 cars images marked with 40 discrete viewpoint class labels.

6.2. Implementation details

Our model is modular w.r.t. the appearance feature $\phi(\mathbf{I}, p_i)$ used in the unary term $S_i(\mathbf{I}, p_i)$. Thus in the experiments on 2D object and parts detection, we construct our method based on the DPM structure as in [12], but with various types of features: HOG feature as DPM [12], Segmentation feature as SegDPM [15], and CNN feature as DeepPyramid DPM [18]. And we apply bounding box regression for object detection. While in the experiments on 2D, 3D pose and viewpoint estimation, we construct our model based on the Mixture-of-Parts structure as [41] of 10 part types, based on HOG feature and CNN feature as [7]. We use the Caffe [23] to compute the CNN feature. When learning the scene-domain geometric subspace $\mathbf{B}$, we follow the NRSFM techniques [9] to set the geometric subspace bases number $K = 5$ for object and parts detection and $K = 8$ for pose and viewpoint estimation. The part number in our models is set to be consistent with the part annotation $\{\mathbf{p}_n\}$ of the training data.

6.3. Experiments on Object and Parts Detection

6.3.1 Object detection

In order to achieve a fair comparison, we compare with three groups of part-based model baselines. The first group uses only HOG as local feature in the unary term $S_i(\mathbf{I}, p_i)$, including the DPM [12], SSDPM [3], and Poselets model [4]. The second group of baselines uses both segmentation and HOG features, including the SegDPM [15], DefPM [29], and Regionlets [39]. And the last group of baseline models uses CNN feature, including DeepPyramid DPM (DP-DPM) [18], C-DPM [33] and Conv-DPM [38]. As claimed before, we show the results of our method using three different features (w. HOG, w. Seg & HOG, and w. CNN) as the baselines. The detailed quantitative results[5] are shown in Table 1. Overall, our method improves the mean average precision (mAP) of the DPM baseline by 4.9%, the SegDPM baseline by 3.5%, and the DP-DPM baseline by 2.1%. Our method outperforms all baselines in detecting coarse-grained object representation.

It is important to note that although several deep learning approaches, *e.g.* [19, 42], performs better in the task of object detection, our method models fine-grained spatial relationship between parts, thus providing much richer object representation, such as 2D parts localization, 3D landmark shape and camera viewpoint, which is essential for further fine-grained applications.

Fig. 2 shows representative qualitative results on object detection. Fig. 2(a) shows example detection results for each of the six animal classes. As we see, our method provides a richer description for objects, *e.g.* the object parts are effectively localized. Fig. 2(b) shows several cat detections, which demonstrates the robustness of our model under geometric variations. As it shows, we can robustly locate the cat instances with non-rigid deformations, viewpoint changes, and partial occlusions. Fig. 2(c) shows some typical examples that are correctly localized by our model but missed by DPM.

6.3.2 Parts localization:

Our method can localize object parts and provides a richer description of objects. We adopt the widely used measure of PCP (Percentage of Correctly estimated body Parts) [37] to evaluate parts localization by our method. We consider the detection with the highest score that has more than 50% overlap with its bounding box, which factors out the effect of the detection. A part is considered as correctly localized if it has more than 40% overlap with the ground truth annotation. Table 2 shows the PCP result using our SDAPM model based on HOG feature and that of SSDPM [3]. Our model outputs fairly precise locations for parts. As we see, our model offers better part localization than [3], which validates the effectiveness of our method in locating parts.

6.3.3 Diagnostic experiments

Importance of scene-domain modeling: To better justify the contribution of our method by modeling active parts in the scene-domain, we compare our method with its two variants: i) "SDAPM without scene-domain modeling" that

[5] Since C-DPM [33] and Conv-DPM [38] have not reported results in PASCAL VOC 2010 dataset, we do not list results of [33, 38]. Note that they show overall comparable results as DP-DPM.

Figure 2. Our method provides a richer object representation and improves the detection results. The blue bounding boxes correspond to the whole object detection, and boxes of other colors correspond to semantic parts respectively, which may indicate different parts across classes. (a) shows one representative result for each of the six animal classes. (b) shows detection results of the cat class to illustrate the ability of our model to robustly represent objects under non-rigid deformations, viewpoint changes, and occlusions. (c) shows typical examples that are correctly localized by our Scene-Domain Active Part Model (SDAPM) but missed by DPM.

	Bird	Cat	Cow	Dog	Horse	Sheep	mAP
i) Our model without scene-domain modeling	11.2	27.0	26.9	23.1	46.7	28.6	27.3
ii) Our model replaced by image-domain geometric modeling	5.8	13.1	11.9	9.7	29.5	19.2	14.9
iii) Our model without occlusion modeling	11.9	25.7	24.2	22.1	44.8	29.1	26.3
Our full model w. HOG	**15.3**	**28.6**	**28.7**	**28.2**	**48.3**	**30.1**	**29.9**

Table 3. Average detection precision of SDAPM and its three variants on the six animal classes of PASCAL VOC 2010 dataset.

uses DPM's standard pairwise term with fixed anchor positions as in Eq. 2, instead of the one in Eq. 4. ii) "SDAPM replaced by image-domain geometric modeling" that replaces the proposed scene-domain geometric modeling by 2D image-domain geometric modeling, where a 2D image-domain geometric subspace $\overline{\mathbf{B}}$ is learned via PCA on the normalized 2D inter-part distances $\mathbf{w}$, and use it to construct $f(\mathbf{w})$ in Eq. 4, *i.e.*, $f(\mathbf{w}) = \mathbf{c}^T \cdot \overline{\mathbf{B}}$ where $\mathbf{c}$ is the weight vector. The comparison results with HOG feature are shown in Table 3. The 1st, 2nd, and 4th rows validate the contribution of modeling geometric statistics in the 3D scene-domain. In particular, the second row demonstrates the contribution of modeling parts' geometric statistics in 3D scene-domain rather than modeling in 2D image-domain.

Importance of occlusion modeling: To validate the importance of our model that explicitly models occlusions, we create another variant to compare with: iii) "SDAPM without occlusion modeling" that discards the occlusion state $\{o_i\}$ and occluded-state templates $\{\boldsymbol{\alpha}_i^o\}$ of our model, but uses DPM's unary term instead. The last two rows of Table 3 justify the importance of explicitly modeling occlusions.

6.4. Experiment on Pose and Viewpoint Estimation

In addition to 2D object and parts localization, our method provides 3D landmark shape and viewpoint estimation. In this experiments, we evaluate our method on 3D fine-grained representation.

6.4.1 2D pose and 3D landmark shape estimation

We construct our model following the mixture-of-parts structure [41], with HOG and CNN features respectively.

MH-Car [20] is state-of-the-art method that recoveries 3D landmark shape and viewpoint by learning from 2D data alone. It detects the 2D pose, then reconstruct it into 3D via two separate steps. Here, we first compare our model with [20] on 2D pose estimation. As shown[6] in Table 4, our model improves over [20] on average, and especially in the estimation of lower arms. This is mainly because

[6]For fair comparison, we list the result of our model w. HOG feature, the same feature used in [20, 41].

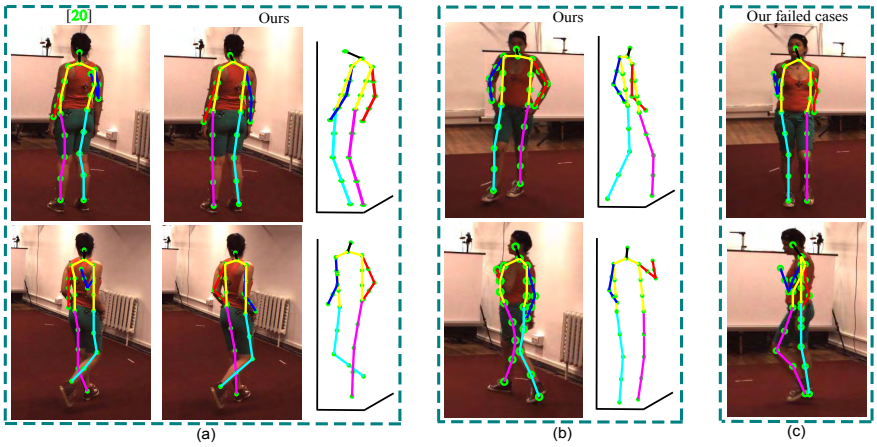

Figure 3. Our method provides a richer object representation including 2D pose, 3D landmark shape, and viewpoint. (a) shows typical poses that are correctly estimated by our method w. HOG but mis-estimated by [20] (*e.g.* the right arm). (b) shows the 2D pose and 3D landmark shape estimations of our method for human in varying viewpoints. (c) gives some failed examples of our method.

	Upper arms	Lower arms	Upper legs	Lower legs	Overall
MoP [41]	60.2	31.1	68.4	62.7	55.6
MH-Car [20]	60.0	31.4	69.0	62.2	55.7
Ours w. HOG	**61.0**	**33.8**	**69.7**	**63.9**	**57.1**

Table 4. 2D pose estimation performance on Human3.6M dataset. The reported numbers are PCP (Probability of Correct Pose).

	MA-N [2]	MH-Car [20]	Ours w. HOG	Ours w. CNN
Average RMS error (mm)	-	298.6	217.2	146.7
Average viewpoint error (°)	27	16	14	8

Table 5. 3D landmark shape estimation on Human3.6M dataset and camera viewpoint estimation on PASCAL Car 2007 dataset.

the scene-domain modeling in our model helps excluding those incorrect configurations in the lower arms. Two examples are shown in Fig. 3(a) to illustrate this, where the right arms are correctly estimated by SDAPM but mis-estimated by [20]. Our method outperforms [41] on average[6], which demonstrates the contribution of modeling parts' geometric statistics in 3D scene-domain instead of 2D image-domain.

Moreover, 3D landmark shapes are recovered in addition to the 2D poses. As shown in Fig. 3(a) (b), the estimated 3D landmark shapes are shown beside the corresponding 2D poses. Although there is inaccuracy in the recovered 3D landmark shapes, *e.g.*, the 3D head position in the upper image of the third column in Fig. 3(a) is not correct, the results are fairly good.

In order to quantitatively evaluate on 3D landmark shape estimation, we compare with [20] using the root mean square error (RMS) metric, which measures the difference of the estimated 3D landmark shape comparing to the 3D landmark ground truth. As shown in the first row of Table 5, SDAPM outperforms [20], especially with CNN feature, which validates the effectiveness of modeling in the scene-domain via a unified process other than two separated steps.

6.4.2 Camera viewpoint estimation

Together with the 3D landmark shape, our SDAPM model estimates projection viewpoint as well. We evaluate viewpoint classification on our car SDAPM model learned from the PASCAL VOC 2007 car dataset [2]. Given a test instance, we run our car model to estimate the camera projection matrix $\mathbf{R}^*$ as well as 3D landmark shape $\mathbf{S}^*$ as discussed in Section 4.1. Then we produce a quantized viewpoint label by matching the reconstructed 2D landmarks generated using the estimated $\mathbf{R}^*$ and $\mathbf{S}^*$ to the landmark locations of the reference images (provided in the dataset). As shown in the last row of Table 5, our method produces an average viewpoint classification error of 8°, which outperforms state-of-the-art viewpoint estimation method [20] with a mean error of 16° and [2] with a mean error of 27°. This suggests that our model can accurately recover the projection viewpoints.

7. Conclusion

In this paper, we have proposed a novel part-based modeling method in the scenario where the training data are based on 2D images. Our method models object parts in the 3D scene-domain and explicitly models occlusions, and accordingly provides finer-grained object representation, including 2D object, parts localization, 3D landmark shape and camera viewpoint estimation. Our method differs from previous part-based object models in that we explore and model the 3D geometric statistics of object parts. Experimental results on two challenging tasks, *i.e.*, object and parts detection, 3D pose and viewpoint estimation, have demonstrated that the proposed method shows superior performance over existing methods with both better robustness to geometric variations and richer object descriptions.

Acknowledgement This work is partially supported by NSF award CCF-1317376, ARO 62250-CS, and CNRS INS2I-JCJC-INVISANA. We also acknowledge the support of NVIDIA Corporation with the donation of GPUs.

References

[1] I. Akhter, Y. A. Sheikh, S. Khan, and T. Kanade. Nonrigid structure from motion in trajectory space. In *NIPS*, 2008.

[2] M. Arie-Nachimson and R. Basri. Constructing implicit 3d shape models for pose estimation. In *ICCV*, 2009.

[3] H. Azizpour and I. Laptev. Object detection using strongly-supervised deformable part models. In *ECCV*, 2012.

[4] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. In *ECCV*, 2010.

[5] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Robust tracking-by-detection using a detector confidence particle filter. In *ICCV*, 2009.

[6] X. Chen, R. Mottaghi, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. Detect what you can: Detecting and representing objects using holistic models and body parts. In *CVPR*, 2014.

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

[8] Y. Chen, L. Zhu, and A. Yuille. Active mask hierarchies for object detection. In *ECCV*, 2010.

[9] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure-from-motion factorization. In *CVPR*, 2012.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010.

[12] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *TPAMI*, 32:1627–1645, 2010.

[13] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structure for object recognition. *IJCV*, 61(1):55–79, 2005.

[14] S. Fidler, S. Dickinson, and R. Urtasun. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In *NIPS*, 2012.

[15] S. Fidler, R. Mottaghi, A. Yuille, and R. Urtasun. Bottom-up segmentation for top-down detection. In *CVPR*, 2013.

[16] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. on Computers*, C-22(1):67 – 92, 1973.

[17] R. Girshick, P. Felzenszwalb, and D. Mcallester. Object detection with grammar models. In *NIPS*, 2011.

[18] R. Girshick, F. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. *CoRR*, abs/1409.5403, 2014.

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

[20] M. Hejrati and D. Ramanan. Analyzing 3d objects in cluttered images. In *NIPS*, 2012.

[21] D. Hoiem, A. A. Efros, and M. Hebert. Closing the loop in scene interpretation. In *CVPR*, 2008.

[22] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 36(7):1325–1339, 2014.

[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:1408.5093*, 2014.

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

[25] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1989.

[26] J. J. Lim, A. Khosla, and A. Torralba. Fpm: Fine pose parts-based model with 3d cad models. In *ECCV*, 2014.

[27] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. *An invitation to 3-d vision: from images to geometric models*. Springer Verlag, 2004.

[28] O. Parkhi, A. Vedaldi, C. Jawahar, and A. Zisserman. The truth about cats and dogs. In *ICCV*, 2011.

[29] O. M. Parkhi, A. Vedaldi, C. V. Jawahar, and A. Zisserman. The truth about cats and dogs. In *ICCV*, 2011.

[30] B. Pepik, P. Gehler, M. Stark, and B. Schiele. $3d^2$pm - 3d deformable part models. In *ECCV*, 2012.

[31] Z. Ren, J. Yuan, J. Meng, and Z. Zhang. Robust part-based hand gesture recognition using kinect sensor. *IEEE Trans. on Multimedia*, 15(5):1110–1120, 2013.

[32] Z. Ren, J. Yuan, and Z. Zhang. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In *ACM Multimedia*, 2011.

[33] P.-A. Savalle, S. Tsogkas, G. Papandreou, and I. Kokkinos. Deformable part models with cnn features. In *3rd Parts and Attributes Workshop, ECCV*, 2013.

[34] A. Shrivastava and A. Gupta. Building part-based object detectors via 3d geometry. In *ICCV*, 2013.

[35] M. Sun and S. Savarese. Articulated part-based model for joint object detection and pose estimation. In *ICCV*, 2011.

[36] E. Trulls, S. Tsogkas, I. Kokkinos, A. Sanfeliu, and F. Moreno. Segmentation-aware deformable part models. In *CVPR*, 2014.

[37] V.Ferrari, M.Marin-Jimenez, and A.Zisserman. Progressive search space reduction for human pose estimation. In *CVPR*, 2008.

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

[39] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *ICCV*, 2013.

[40] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *CVPR*, 2010.

[41] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *TPAMI*, 35:2878 – 2890, 2012.

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