Estimating the Aspect Layout of Object Categories

Yu Xiang and Silvio Savarese
Department of Computer Science and Electrical Engineering
University of Michigan at Ann Arbor, Ann Arbor, MI 48109, USA
{yuxiang, silvio}@eecs.umich.edu

Abstract

In this work we seek to move away from the traditional paradigm for 2D object recognition whereby objects are identified in the image as 2D bounding boxes. We focus instead on: i) detecting objects; ii) identifying their 3D poses; iii) characterizing the geometrical and topological properties of the objects in terms of their aspect configurations in 3D. We call such characterization an object's aspect layout *(see Fig. 1). We propose a new model for solving these problems in a joint fashion from a single image for object categories. Our model is constructed upon a novel framework based on conditional random fields with maximal margin parameter estimation. Extensive experiments are conducted to evaluate our model's performance in determining object pose and layout from images. We achieve superior viewpoint accuracy results on three public datasets and show extensive quantitative analysis to demonstrate the ability of accurately recovering the aspect layout of objects.*

1. Introduction

In most traditional object recognition methods, object categories are represented as 2D flat entities. The focus lies more on taming the intra-class variability within each category (indeed a very challenging problem) rather than seeking to model the intrinsic 3D nature of the object. Also, most of the methods aim at detecting objects in images and identifying them using a bounding box rather than estimating their geometrical properties such as the object 3D pose or the 3D layout configuration of their parts. While the 2D object detection problem is very useful in many applications such as Internet-based image search (and impressive results have been obtained), it is less so in applications such as robotics, autonomous navigation and manipulation. In such applications it is critical not only to recognize objects in 2D but also to estimate their locations and poses in 3D (Fig. 1). Moreover, the ability to parse the object layout and identify object functional elements such as the back or the seat of a sofa is crucial for enabling an agent to effectively interact with the objects in the scene (Fig. 1).

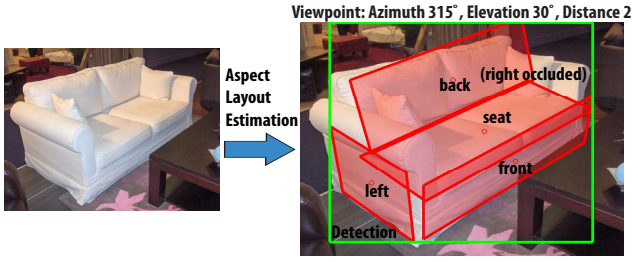

Figure 1. Illustration of aspect layout estimation of a sofa. Left: input image with a sofa. Right: the estimation result given by our method: the sofa is detected by the green bounding box, its viewpoint is estimated and its aspect parts are either located by a red quadrilateral or determined as self-occluded.

In this paper, we address the problem of detecting object categories, determining their 3D poses and estimating the objects' 3D layout from a single image. By object's layout we mean the configuration of object parts in 3D (Fig. 1). Instead of considering an arbitrary definition of *object part*, we seek to identify parts that have geometrical and topological relevance. We call these parts *aspect parts*. An aspect part can be defined as a portion of the object whose entire 3D surface is approximately either entirely visible from the observer or entirely non-visible (i.e., occluded). The seat and the back of a sofa are two examples of approximated aspect parts. The combination of the seat and the back of the sofa is *not* an aspect part as there are certain viewpoints from which either the back is visible and the seat is not, or, conversely, the seat is visible and the back is not. A planar surface is an ideal aspect part. The concept of aspect part is related to that of aspect graph which was introduced in the pioneering work by Koenderink and Doorn [22].

The ability to estimate the pose and the 3D layout of an object is connected to several key computer vision problems. An aspect part can be related to the concept of object affordance or functional part such as the seat or back of a sofa, thus our work is critical in object affordance estimation problems such as these addressed in [30]. Also,

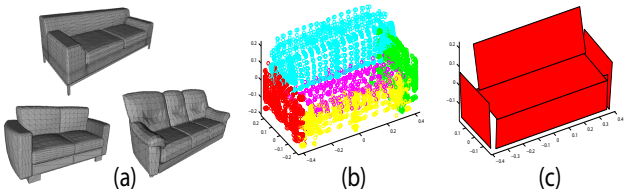

Figure 2. Overview of the training steps to build the 3D object model $O = (o_1, o_2, \ldots, o_n)$. We illustrate an example from the sofa category. i) Collect 3D CAD models of *sofa*, rescale the CAD models to fit into a unit sphere and orient them along their dominant dimension. Fig. 2(a) shows three 3D CAD models of *sofa* we collected from Google 3D Warehouse [1]. ii) Identify aspect parts, segment 3D points in each CAD model according to the aspect parts using manual annotations and aggregate all the 3D points from the CAD models. Fig. 2(b) shows the 3D point cloud after segmentation and aggregation, where different colors represent different aspect parts. iii) Fit a rectangle to the 3D points belonging to each aspect part. First, fit a 2D plane to these 3D points, and then project the 3D points onto the plane. Finally, draw a bounding box of the projected points in the plane to obtain the rectangle for the aspect part. Fig. 2(c) shows the 3D model we built for *sofa*.

it allows us to characterize the object with geometrical attributes such as "it has an horizontal support surface" or "it has a back surface" which are suitable for fine-grained object recognition, zero-shot learning or transfer learning problems [15]. Our work provides tools for effectively modeling object-scene interactions [36] and for scene layout understanding [19, 18, 6, 5]. Finally, it can be useful for automatic 3D object reconstruction or rough 3D shape prototyping from a single image [33, 4, 32].

In this work we propose a new model for jointly solving the object detection, pose classification and layout estimation problem. We call this model the Aspect Layout Model (ALM). ALM is constructed as follows. Aspect parts and their 3D configuration are automatically learnt from a set of 3D CAD models from which the aspect parts are manually identified for each object category (see Fig. 2 for details). The relationship between the 3D configuration of aspect parts and their corresponding projections (observations) in the images are modeled using a discriminative framework based on Conditional Random Fields (CRFs) [23] with maximal margin parameter estimation. The unary potential of the CRF captures appearance and shape properties of each projected aspect part in the image. Projected aspect parts are shared across views and their appearances and shapes are rectified to their most frontal poses in order to guarantee view invariance. As a result, only one 2D part template is trained for each aspect part regardless of the number of viewpoints in the dataset. The pairwise potential is used to enforce spatial constraints to the relative 2D locations of aspect parts.

To summarize, our paper has the following key contributions:

- Object detection, viewpoint classification and aspect layout estimation are jointly solved using a rigorous coherent formulation. Our method allows us to accurately estimate each aspect part's 3D location and orientation in the object reference system as well as reason about which aspect part is visible or occluded from the estimated viewpoint.
- The learnt aspect part templates are made view invariant by injecting a rectification process into inference.
- We significantly outperform state-of-the-art methods in estimating object pose using three public datasets as well as demonstrate the ability of accurately recovering the aspect layout of an object category from a single image.

The rest of the paper is organized as follows: Section 2 reviews related works. Section 3 describes our aspect layout model including parameter estimation and model inference. Section 4 presents the experimental evaluation and Section 5 concludes the paper.

2. Related Work

Part-based object representations have been widely used in computer vision (e.g., [14, 13]). Felzenszwalb et al. [12] utilize a part-based representation for general object detection and achieve remarkable detection results. Gu and Ren [17] extend [12] for viewpoint classification by discriminatively training mixture of templates of object viewpoints. However, both [12] and [17] only train independent models for a small number of discrete viewpoints, and the 3D spatial relationships between parts are not modeled.

Various approaches have been recently proposed that explicitly take into account the 3D nature of object categories [34, 28, 20, 7, 31, 4, 11, 25, 29, 27, 16]. These methods can be roughly classified into two main categories. Methods in the first category represent object as collections of parts or features which are connected across views [34, 28, 31, 4, 11, 27]. Methods in the second category represent objects using an explicit 3D model on top of which features or parts are associated [20, 7, 25, 29, 16]. [20] proposes a CRF built on top of a rough 3D object model. The approach can be used for both object detection and segmentation. Similar to our model, Chui et al. [7] propose a 3D object representation which consists of planar parts. However, [7] mostly uses such 3D representation to generate virtual training examples. Unlike [25, 29], where 2D object detectors and 3D models are independent, our approach is based on the interaction between 3D object representation and 2D part detectors to guide the process of locating aspect parts and estimating object poses. Unlike [11], where object aspects are treated as latent variables, we relate our definition of aspect parts to 3D topological properties of the object category.

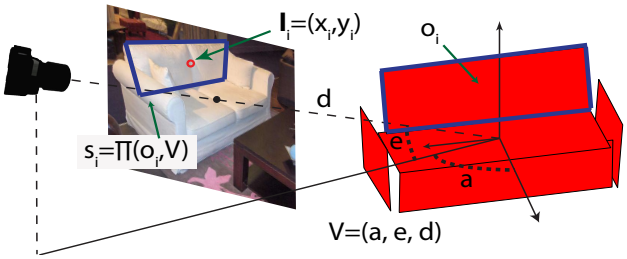

Figure 3. Illustration of viewpoint representation and part shape from 3D. The viewpoint V is represented by azimuth a, elevation e and distance d of the camera pose. 2D part shape s_i is determined by the viewpoint transformation $\Pi(o_i, V)$ with o_i be the ith 3D aspect part (*back* of the sofa in the figure). The part center location $\mathbf{l}_i$ is also shown.

3. Aspect Layout Model

We propose a novel Aspect Layout Model (ALM) for estimating the 3D aspect layout of object categories. Suppose that each object in a category consists of n aspect parts. Let $O = (o_1, o_2, \ldots, o_n)$ denote the object in 3D, where $o_i, i = 1, \ldots, n$ represents the ith aspect part. Fig. 2 illustrates the training steps to construct the 3D object O from a set of 3D CAD models. Given an input image I, ALM predicts the object label $Y \in \{+1, -1\}$ indicating the presence or absence of an object instance in the image, and the part configuration $C = (\mathbf{c}_1, \ldots, \mathbf{c}_n)$ if $Y = +1$. The state of part i is given by $\mathbf{c}_i = (x_i, y_i, s_i)$, x_i and y_i are the part center coordinates in the image coordinate system, and s_i represents the part shape in the image. Based on the observation that a 2D part shape is jointly determined by the 3D geometry of the part and the viewpoint, the part shape s_i is given by the viewpoint transformation $\Pi(o_i, V), i = 1, \ldots, n$, where V denotes the viewpoint. Suppose that the 3D object is positioned at the world coordinate origin and the camera always looks at the origin without in-plane rotation. Then the viewpoint can be parameterized by $V = (a, e, d)$ with a, e, d being azimuth, elevation and distance of the camera pose. Fig. 3 illustrates the viewpoint representation and the 2D part shape generated by the viewpoint transformation. The posterior distribution of object label and part configuration can be written as

$$\begin{aligned} P(Y, C|I) &= P(Y, \mathbf{c}_1, \ldots, \mathbf{c}_n|I) \\ &= P(Y, x_1, y_1, s_1, \ldots, x_n, y_n, s_n|I) \\ &= P(Y, x_1, y_1, \ldots, x_n, y_n, O, V|I) \\ &= P(Y, L, O, V|I), \end{aligned} \tag{1}$$

where $L = (\mathbf{l}_1, \ldots, \mathbf{l}_n)$ and $\mathbf{l}_i = (x_i, y_i), i = 1, \ldots, n$ denotes the 2D part center coordinates. In the third line of Eq. (1), we replace $s_i, i = 1, \ldots, n$ with O and V, since the part shape s_i in the image is completely specified by the viewpoint transformation $\Pi(o_i, V)$. Then, the part configuration is given by L, O and V. Inference is achieved by maximizing the posterior distribution $P(Y, L, O, V|I)$.

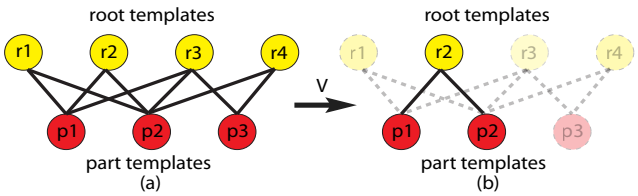

Figure 4. (a) An example of the bipartite graph structure in our model. A root template is connected to all the visible part templates in its view section. (b) Under a specific viewpoint V, the graph reduces to a tree with the root template as the root node and all the visible part templates under the viewpoint as its children.

3.1. Discriminative Modeling

We model ALM discriminatively using a Conditional Random Field (CRF) [23] formulation. The posterior distribution of object label and part configuration is

$$P(Y, L, O, V|I) \propto \exp\big(E(Y, L, O, V, I)\big), \tag{2}$$

where $E(Y, L, O, V, I)$ is the energy function. By imposing a graph structure $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ over parts as described below, the energy function can be decomposed as

$$E(Y, L, O, V, I) = \tag{3}$$
$$\begin{cases} \sum_{i \in \mathcal{V}} V_1(\mathbf{l}_i, O, V, I) + \sum_{(i,j) \in \mathcal{E}} V_2(\mathbf{l}_i, \mathbf{l}_j, O, V), & \text{if } Y = +1 \\ 0, & \text{if } Y = -1, \end{cases}$$

where V_1 and V_2 are the unary potential and pairwise potential respectively. The unary potential captures the visual appearances of parts, while the pairwise potential encodes the spatial relationships between parts. The energy of a negative sample is set to zero.

Graph Structure. In our model, the unary potential is designed as a 2D part template. We use one part template for each aspect part in 3D. Moreover, we introduce root templates which are associated with the whole object from different viewpoints. Specifically, we divide the viewing sphere into a fixed number of view sections (e.g., 8 view sections with each covering 45° azimuth). For each view section, we add one 2D root template into ALM. The root template is activated if the object is viewed inside its view section. All the other root templates are considered to be occluded. Then we impose a bipartite graph structure $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ between the root templates and the part templates. A root template is connected to all the visible part templates in its view section, but there is no link between two root templates or two part templates. An important property of the bipartite graph structure is that, under a specific viewpoint, the graph reduces to a tree formed by all the visible templates. So we can have a local tree structure for

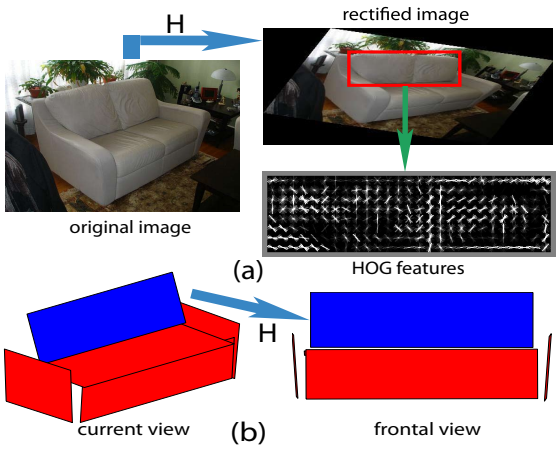

Figure 5. Illustration of rectified HOG features for the *back* of the *sofa* object category. (a) The original image is rectified to the frontal view of the aspect part *back* of the sofa using the homographic transformation H. Rectified HOG features for *back* are extracted from the red bounding box which delimits the transformed image of the *back* part to its frontal view. (b) The homographic transformation H between *back*'s current view and its frontal view is used for rectification.

each viewpoint and solve the inference problem efficiently. Fig. 4 illustrates the graph structure in our model.

Viewpoint Invariant Unary Potential. The unary potential is modeled with a linear discriminative model as

$$V_1(\mathbf{l}_i, O, V, I) = \begin{cases} \mathbf{w}_i^T \phi(\mathbf{l}_i, O, V, I), \text{ if unoccluded} \\ \alpha_i, \text{ if occluded,} \end{cases} \quad (4)$$

where $\mathbf{w}_i$ is the weight of the linear model, α_i is the weight for part i if it is occluded under viewpoint V, and $\phi(\mathbf{l}_i, O, V, I)$ represents the feature vector which consists of HOG features [8] in our implementation. Unlike previous methods [12, 17] which train multiple independent object templates for different viewpoints, ALM only trains one template for each part across all viewpoints. Similar to [28], the template corresponds to the frontal view of the part. This is achieved by rectifying the part appearance using an homographic transformation H that transforms a part to its frontal view, where H can be obtained from the 3D model given V. Then HOG features are extracted from the rectified part. A reliable rectification process is also proposed in [18]. Consequently, ALM is able to estimate fine-grained viewpoints and capture the relationships between viewpoints in a compact form. Fig. 5 illustrates an example of rectified HOG features.

Pairwise Potential. The pairwise potential captures the relationship between relative part locations and orientations in the image. In the ideal case, the relative locations given by projecting the 3D object O onto the image according to the viewpoint V and the corresponding observed relative locations should be equal. We design the pairwise potential so as to penalize deviation of the observed relative part locations from the ideal ones. Let (x'_i, y'_i) and (x'_j, y'_j) be the positions of the joints between part i and part j in the image coordinates (see [13] for the definition of joint), $d_{ij,O,V}$ be the learnt distance between part i and part j given by projecting the 3D object O according to the viewpoint V to the image and $\theta_{ij,O,V}$ be the learnt relative orientation between part i and part j. Then the joint coordinates are given by

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \tfrac{1}{2} d_{ij,O,V} \begin{bmatrix} cos(\theta_{ij,O,V}) \\ sin(\theta_{ij,O,V}) \end{bmatrix}, \quad (5)$$

$$\begin{bmatrix} x'_j \\ y'_j \end{bmatrix} = \begin{bmatrix} x_j \\ y_j \end{bmatrix} + \tfrac{1}{2} d_{ji,O,V} \begin{bmatrix} cos(\theta_{ji,O,V}) \\ sin(\theta_{ji,O,V}) \end{bmatrix}, \quad (6)$$

where $d_{ij,O,V}$, $d_{ji,O,V}$, $\theta_{ij,O,V}$, and $\theta_{ji,O,V}$ are computed from the 3D model. The pairwise potential is the negative squared distance between the two joints. Since $d_{ij,O,V} = d_{ji,O,V}$ and $\theta_{ij,O,V} = \theta_{ji,O,V} + \pi$, we have the following pairwise potential

$$V_2(\mathbf{l}_i, \mathbf{l}_j, O, V) = -w_x \big(x_i - x_j + d_{ij,O,V} cos(\theta_{ij,O,V})\big)^2 - w_y \big(y_i - y_j + d_{ij,O,V} sin(\theta_{ij,O,V})\big)^2, \quad (7)$$

where w_x and w_y are the parameters controlling the strength of the pairwise constraints.

Energy Function. Since both the unary and pairwise potentials are linear with respect to its own parameters, we can aggregate all the model parameters into one parameter vector $\theta = (\mathbf{w}_{i,\forall i}, \alpha_{i,\forall i}, w_x, w_y)$, and aggregate all the corresponding energy components into one feature vector $\Psi(Y, L, O, V, I)$. Then the energy function is

$$E(Y, L, O, V, I|\theta) = \theta^T \Psi(Y, L, O, V, I). \quad (8)$$

3.2. Maximal Margin Parameter Estimation

The most widely used technique for parameter estimation in CRFs is maximum likelihood, which requires proper normalization of the probabilities. However, normalization is not necessary in discriminative modeling. Consider the following inference problem:

$$(Y^*, L^*, O^*, V^*) = \arg \max_{Y,L,O,V} E(Y, L, O, V, I|\theta). \quad (9)$$

We note that only the "relative energy" values matter. By relative energy we refer to the difference between two energy values as opposed to the energy values themselves. From the point of view of energy based learning [24], the aim of parameter estimation in our model is to find an energy function which outputs the maximal energy value for the correct label configuration of an object in the image.

To train the model, we are given a set of training samples $\mathcal{T} = \{(I^t, Y^t, L^t, O^t, V^t), t = 1, \ldots, N\}$, where each sample is an image with the object label, 2D part center locations, learnt 3D model and viewpoint. Then a loss function is defined to evaluate the quality of a specific energy

function. Finally, the parameters are estimated by minimizing the loss on the training set $\mathcal{T}$. If hinge loss is used in combination with a quadratic regularizer, the parameter estimation problem is equivalent to the following structural SVM optimization problem [35]:

$$\min_{\theta} \frac{1}{2}\|\theta\|^2 + \lambda \sum_{t=1}^{N} \Big[\max_{Y,L,O,V} \left[\theta^T \Psi_{t,Y,L,O,V} + \Delta_{t,Y,L,O,V}\right] - \theta^T \Psi_{t,Y^t,L^t,O^t,V^t} \Big], \quad (10)$$

where λ is a fixed penalty parameter, $\Psi_{t,Y,L,O,V} = \Psi(Y, L, O, V, I^t)$, $\Psi_{t,Y^t,L^t,O^t,V^t} = \Psi(Y^t, L^t, O^t, V^t, I^t)$ and $\Delta_{t,Y,L,O,V} = \Delta(Y, L, O, V, Y^t, L^t, O^t, V^t)$ is the loss function measuring the difference between two sets of labels. We use the weighted 0-1 loss, i.e., $\Delta_{t,Y,L,O,V} = \beta \mathbf{I}(Y \neq Y^t)$, where β is a predefined constant and $\mathbf{I}$ is the indicator function. The above optimization problem can be solved efficiently using the cutting plane training method [21]. We choose λ and β using a validation procedure.

3.3. Model Inference

Model inference aims to predict the object label and part configuration of an object. The inference problem is already given by Eq. (9). Viewpoints are discretized by sampling the viewing space defined by the azimuth, elevation and distance of the camera pose. Inference is then performed independently for different combinations of O and V.

Given O and V, Belief Propagation (BP) [37] can be utilized to infer the 2D part center locations when $Y = +1$. Since the bipartite graph $\mathcal{G}$ reduces to a tree under a specific view, the inference for part location is optimal. BP works in a message passing fashion. The message that part i sends to its parent j is defined as

$$m_{ij}(\mathbf{l}_j) = \max_{\mathbf{l}_i} \Big(V_1(\mathbf{l}_i) + V_2(\mathbf{l}_i, \mathbf{l}_j) + \sum_{k \in \text{kids}(i)} m_{ki}(\mathbf{l}_i) \Big), \quad (11)$$

where V_1 and V_2 are the unary potential and pairwise potential respectively, and kids(i) denotes the children of part i. Messages are passed in the direction from the leaves to the root. Thus, we can obtain the belief vector at the root

$$b_i(\mathbf{l}_i) = V_1(\mathbf{l}_i) + \sum_{j \in \text{kids}(i)} m_{ji}(\mathbf{l}_i). \quad (12)$$

The location which maximizes the above belief is the optimal location for the root. By keeping track of the argmax indices in Eq. (11), we can backtrace to find all the optimal locations of the other parts. After performing BP for all the combinations of O and V, we can obtain the energy value $E(Y = +1, L^*, O^*, V^*)$. The object label $Y^* = +1$ if and only if $E(Y = +1, L^*, O^*, V^*) > \gamma$, where γ is the detection threshold. To generate multiple detections in image I, we can threshold the belief at the root (Eq. (12)) and apply non-maximum suppression.

4. Experiments

Datasets. We evaluate our method for object aspect layout estimation on three public datasets: the 3DObject dataset [28], the VOC2006 Car dataset [10] and the EPFL Car dataset [26], and a new challenging dataset we extracted from ImageNet [3]. The 3DObject dataset is a standard benchmark for object pose estimation. It consists of 10 categories, each containing 10 different object instances observed from different viewpoints. We exclude the Head and the Monitor categories as they are not evaluated in previous work. The VOC2006 Car dataset consists of 921 car instances with viewpoint labels (Frontal, Rear, Left and Right). The EPFL Car dataset consists of 2,299 images of 20 car instances covering 360° azimuth in $3^\circ - 4^\circ$ steps with nearly the same elevation and distance. The new ImageNet dataset consists of four categories: Bed (400 images), Chair (770 images), Sofa (800 images) and Table (670 images). We manually annotated each object in the four datasets with azimuth, elevation, distance and part center locations following the structure of our 3D models unless the annotations were already available.

For each category in the 3DObject dataset, we use 5 instances for training and the other 5 instances for testing. Negative samples are randomly selected from the VOC2007 dataset [9]. For the VOC2006 Car dataset, we train on the training and validation sets and test on the test set. For the EPFL Car dataset, we use the same training and testing partition as described in [26]. For each category in the ImageNet dataset, we use 50% images for training and test on the other 50% images, where we randomly separate the set of images under the same viewpoint into training images and test images.

Evaluation Measures. Object aspect layout estimation involves object detection, viewpoint estimation and part localization. We use Average Precision (AP) to measure the detection performance. The standard 50% bounding box overlap criteria of PASCAL VOC [10] is used. For viewpoint estimation, we use the average viewpoint accuracy as performance measure, which is the average of the elements on the main diagonal of the viewpoint confusion matrix (see technical report [2] for the confusion matrices in our experiments). As in all previous work, the viewpoint accuracy is computed among the true positives. To see how the viewpoint estimation is related to detection, we report the viewpoint accuracy as a function of the recall (see [2] for details). For part localization, we use the Percentage of Correct Parts (PCP) in true positives as the evaluation measure. A predicted part is considered to be correct if the overlap between the predicted part and ground truth part is larger than 50%. Because part localization is evaluated only when the object is correctly detected, we plot PCP as a function of the recall. Then the area under the PCP-Recall curve is used as the quantitative measure for part localization. In the

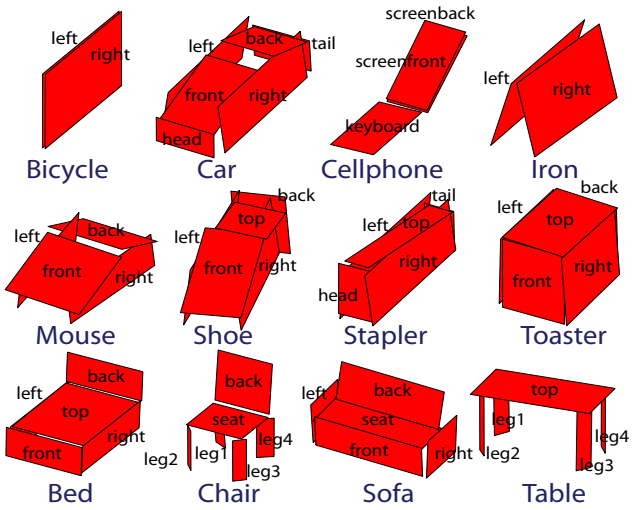

Figure 6. Our 3D object models for the 12 categories in our experiments. Each aspect part is associated to a part label.

Table 1. Results on the 3DObject dataset and the VOC2006 Car dataset.

Dataset	3DObject (8 views)			VOC2006 Car (4 views)		
Method	ALM	[17]	[28]	ALM	[17]	[31]
Viewpoint	**80.7**	74.2	57.2	**85.9**	85.7	73.0
Detection	81.8	n/a	n/a	48.7	**51**	35

Table 4. Average viewpoint accuracy on the 3DObject Car dataset with different training set sizes (number of instances).

Training Set Size	1	2	3	4	5
DPM [12]	69.2	81.9	84.5	84.6	85.0
ALM Root	**80.6**	**88.5**	90.5	91.7	89.2
ALM Full	76.3	85.1	**92.7**	**92.6**	**93.4**

evaluation, we account for occlusion between parts, i.e., an occluded part that is predicted as being visible is considered to be incorrect.

4.1. Results

3DObject Dataset. We first evaluate the performance of ALM for aspect layout estimation using portion of the 3DObject dataset. The first two rows of Fig. 6 show our 3D object models for the 8 categories of the 3DObject dataset. The left portion of Table 1 shows the overall viewpoint estimation and detection results averaged on the 8 categories. Our model achieves 80.7% average viewpoint accuracy over 8 viewpoints, which is higher than 74.2% of the state-of-the-art [17]. [17] and [28] do not report the detection AP. Most of the previous works mainly conducted experiments on the Bicycle and Car categories. We also compare with the state-of-the-art methods on these two categories and present the results in Table 2. Our approach achieves the best performances.

More detailed viewpoint estimation results on the 3DObject dataset are presented in Table 3 (See [2] for detailed detection results on the 3DObject dataset). We compare our full model with our root model and the state-of-the-art object detector DPM [12], where the root model is trained only with root templates. We train and test DPM with the same training and test sets as ALM. Eight root templates with parts are trained for DPM according to the 8 viewpoints. Our full model achieves the best viewpoint estimation among the three models. This demonstrates that adding part templates plays an important role in obtaining high performances. To see more clearly the benefit of employing the relationship between views, we compare the average viewpoint accuracy of our full model, our root model and DPM on the 3DObject Car dataset with different training set sizes. The results are given in Table 4, where the training set size is varied from 1 to 5 instances. The full model and the root model obtain better results than DPM in all the settings. By using more than 3 instances, the full model achieves better performances than the root model.

Table 5. Results on the EPFL Car dataset (16 views).

Method	ALM Full	ALM Root	DPM [12]	[26]
Viewpoint	**64.8**	58.1	56.6	41.6
Detection	96.4	97.5	**98.1**	85.4

We evaluate the ability to localize aspect parts by using the PCP-Recall curves. Fig. 7 reports the PCP-recall curves of parts for the 8 categories. If the area under the curve is close to one, then we have good localization performance for the part (i.e., the *left* and *right* of car). Note that for the toaster category, we only use the *top* aspect part. Since the other parts have nearly no texture, we find that it is almost impossible to locate these parts in a reliable fashion. Some anecdotal aspect layout estimation results for the 8 categories are shown in Fig. 8. Notice that ALM is robust to intra-class variability and viewpoint change.

VOC2006 Car Dataset. We also conducted experiments on the VOC2006 Car dataset. The results for viewpoint estimation and object detection are showed on the right portion of Table 1. We achieve nearly the same results as [17] and better results than [31]. Our method is less effective if the viewpoint distribution in training and testing is too coarse. There are only 4-view labels in the VOC2006 Car dataset.

EPFL Car Dataset. In order to compare the performance of our algorithm with [26], we bin our viewpoint estimation into 16 bins (22.5° azimuth degree). DPM is trained with 16 templates according to the 16 views. The results on this dataset are presented in Table 5. Notice that as the number of viewpoints increases, the full model achieves significant improvement on viewpoint accuracy over the root model and DPM (See [2] for the viewpoint confusion matrix and the histogram of azimuth errors in degree).

ImageNet Dataset. The last row of Fig. 6 shows our 3D models for the 4 categories in the dataset. Most of the objects in the dataset are viewed from their front. So we evaluate the viewpoint estimation on 3 views (front, front-left, front-right) as well as 7 views (azimuth 0°, 15°, 30°, 45°,

Table 2. Results on the Bicycle and Car categories in the 3DObject dataset.

Category	Bicycle			Car						
Method	ALM	[27]	[25]	ALM	[27]	[16]	[29]	[25]	[31]	[4]
Viewpoint	**91.4**	80.8	75.0	**93.4**	85.4	85.3	81	70	67	48.5
Detection	**93.0**	n/a	69.8	98.4	n/a	**99.2**	89.9	76.7	55.3	n/a

Table 3. Average viewpoint accuracy on the 3DObject dataset.

Category	Bicycle	Car	Cellphone	Iron	Mouse	Shoe	Stapler	Toaster	*Mean*
DPM [12]	88.4	85.0	62.1	82.7	40.0	71.7	58.5	55.0	67.9
ALM Root	**92.5**	89.2	83.4	**86.0**	58.7	82.7	69.2	59.6	77.7
ALM Full	91.4	**93.4**	**85.0**	84.6	**66.5**	**87.0**	**72.8**	**65.2**	**80.7**

Table 6. Average viewpoint accuracy on the ImageNet dataset.

Category	Bed	Chair	Sofa	Table	*Mean*
3 views					
DPM [12]	84.1	**88.6**	90.1	75.6	84.6
ALM Root	84.7	60.2	91.0	**80.0**	79.0
ALM Full	**90.0**	87.7	**92.4**	76.0	**86.5**
7 views					
DPM [12]	56.2	41.2	44.0	**56.4**	49.5
ALM Root	37.5	23.4	39.6	35.4	34.0
ALM Full	**62.7**	**73.1**	**65.0**	52.6	**63.4**

315°, 330° and 345°) respectively. The results are shown in Table 6. Our full model achieves significant improvements on viewpoint estimation over the root model and DPM when 7 views are considered. The full model leverages the ability to handle few training samples by sharing part across views. Our full model achieves average detection AP 90.4% on the 4 categories, which is almost on par to 95.5% of DPM (See [2] for detailed detection results on the ImageNet dataset). We show the PCP-Recall curves for part localization of the 4 categories in the last row of Fig. 7. Anecdotal aspect layout estimation results are shown in Fig. 8.

5. Conclusion

We have proposed a new model (called ALM) for jointly detecting objects in a category, estimating the viewpoints of objects and locating the aspect parts of objects. Our approach jointly models object 3D geometry, viewpoint and 2D aspect parts in images. ALM is able to handle a large number of views, locate aspect parts with approximately correct orientations and reason about occlusions among aspect parts. We have conducted extensive experiments to demonstrate the ability of our model to solve the three tasks. We show high precision in detecting aspect parts using the 3DObject dataset and the subset of the ImageNet dataset. These results indicate that our method can be potentially useful in problems where functional parts or affordances are to be estimated.

Acknowledgments

We acknowledge the support of ARO under grant W911NF-09-1-0310 and NSF CAREER under grant #1054127.

References

[1] http://sketchup.google.com/3dwarehouse.
[2] http://www.eecs.umich.edu/vision/papers/xiang_cvpr12_tr.pdf.
[3] http://www.image-net.org.
[4] M. Arie-Nachimson and R. Basri. Constructing implicit 3d shape models for pose estimation. In *ICCV*, 2009.
[5] S. Y. Bao and S. Savarese. Semantic structure from motion. In *CVPR*, 2011.
[6] S. Y. Bao, M. Sun, and S. Savarese. Toward coherent object detection and scene layout understanding. In *CVPR*, 2010.
[7] H. Chiu, L. Kaelbling, and T. Lozano-Pérez. Virtual training for multi-view object class recognition. In *CVPR*, 2007.
[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.
[9] M. Everingham, L. Van Gool, I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 Results.
[10] M. Everingham, A. Zisserman, I. Williams, and L. Van Gool. The PASCAL Visual Object Classes Challenge 2006 Results.
[11] A. Farhadi, M. Tabrizi, I. Endres, and D. Forsyth. A latent model of discriminative aspect. In *ICCV*, 2009.
[12] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 2010.
[13] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 2005.
[14] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR*, 2003.
[15] V. Ferrari and A. Zisserman. Learning visual attributes. In *NIPS*, 2008.
[16] D. Glasner, M. Galun, S. Alpert, R. Basri, and G. Shakhnarovich. Viewpoint-aware object detection and pose estimation. In *ICCV*, 2011.
[17] C. Gu and X. Ren. Discriminative mixture-of-templates for viewpoint classification. In *ECCV*, 2010.
[18] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*, 2010.
[19] D. Hoiem, A. Efros, and M. Hebert. Putting objects in perspective. In *CVPR*, 2006.
[20] D. Hoiem, C. Rother, and J. Winn. 3d layoutcrf for multi-view object class recognition and segmentation. In *CVPR*, 2007.
[21] T. Joachims, T. Finley, and C.-N. Yu. Cutting-plane training of structural svms. *Machine Learning*, 2009.
[22] J. J. Koenderink and A. J. Doorn. The internal representation of solid shape with respect to vision. *Biological Cybernetics*, 1979.
[23] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.
[24] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. J. Huang. A tutorial on energy-based learning. In *Predicting Structured Data*. MIT Press, 2006.

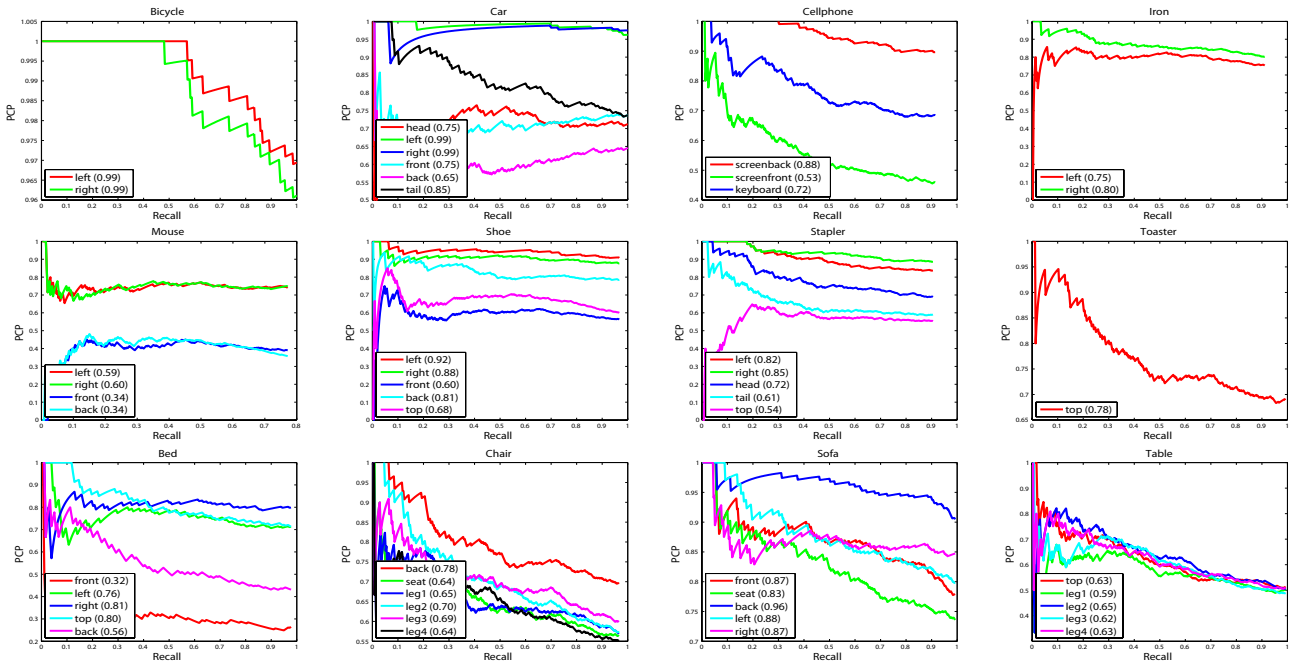

Figure 7. PCP-Recall curves for part localization on the 3DObject dataset (first two rows) and the ImageNet dataset (last row).

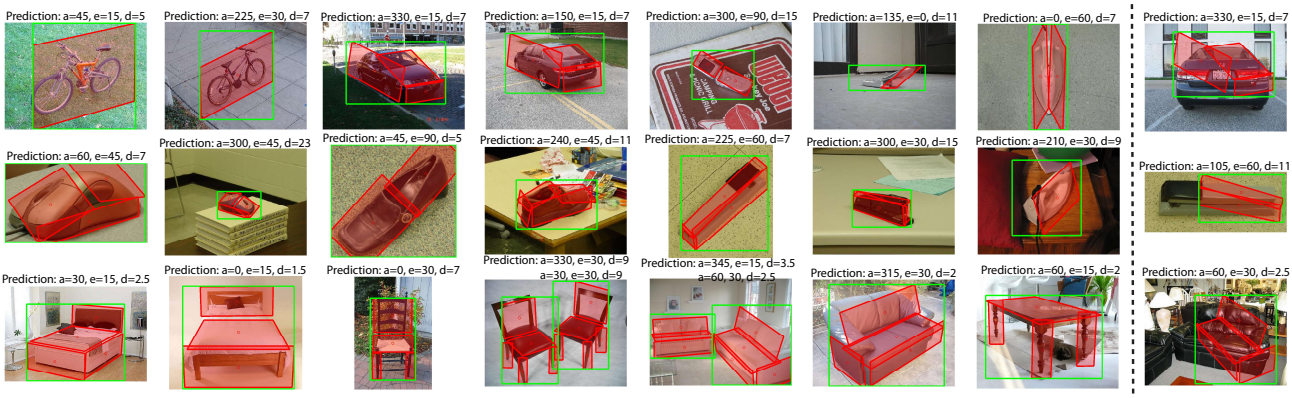

Figure 8. Anecdotal aspect layout estimation results on the 3DObject dataset and the ImageNet dataset. The last column shows some wrong estimations. (Please zoom in to see details, and more results are presented in [2].)

[25] J. Liebelt and C. Schmid. Multi-view object class detection with a 3D geometric model. In *CVPR*, 2010.

[26] M. Ozuysal, V. Lepetit, and P. Fua. Pose estimation for category specific multiview object localization. In *CVPR*, 2009.

[27] N. Payet and S. Todorovic. From contours to 3d object detection and pose estimation. In *ICCV*, 2011.

[28] S. Savarese and L. Fei-Fei. 3d generic object categorization, localization and pose estimation. In *ICCV*, 2007.

[29] M. Stark, M. Goesele, and B. Schiele. Back to the future: Learning shape models from 3d cad data. In *BMVC*, 2010.

[30] M. Stark, P. Lies, M. Zillich, J. Wyatt, and B. Schiele. Functional object class detection based on learned affordance cues. In *International conference on computer vision systems*, 2008.

[31] H. Su, M. Sun, L. Fei-Fei, and S. Savarese. Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In *ICCV*, 2009.

[32] M. Sun, G. Bradski, B.-X. Xu, and S. Savarese. Depth-encoded hough voting for coherent object detection, pose estimation, and shape recovery. In *ECCV*, 2010.

[33] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, and L. V. Gool. Depth-from-recognition: Inferring metadata by cognitive feedback. In *ICCV Workshop on 3D Representations for Recognition*, 2007.

[34] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and L. Van Gool. Towards multi-view object class detection. In *CVPR*, 2006.

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

[36] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010.

[37] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. In *Exploring artificial intelligence in the new millennium*, 2003.