3D generic object categorization, localization and pose estimation

Silvio Savarese
Beckman Institute
University of Illinois at Urbana-Champaign
Urbana, IL USA
silvio@uiuc.edu

Li Fei-Fei
Computer Science Department
Priceton University
Princeton, NJ USA
feifeili@CS.Princeton.EDU

Abstract

We propose a novel and robust model to represent and learn generic 3D object categories. We aim to solve the problem of true 3D object categorization for handling arbitrary rotations and scale changes. Our approach is to capture a compact model of an object category by linking together diagnostic parts of the objects from different viewing points. We emphasize on the fact that our "parts" are large and discriminative regions of the objects that are composed of many local invariant features. Instead of recovering a full 3D geometry, we connect these parts through their mutual homographic transformation. The resulting model is a compact summarization of both the appearance and geometry information of the object class. We propose a framework in which learning is done via minimal supervision compared to previous works. Our results on categorization show superior performances to state-of-the-art algorithms such as [23]. Furthermore, we have compiled a new 3D object dataset that consists of 10 different object categories. We have tested our algorithm on this dataset and have obtained highly promising results.

1. Introduction

Object categorization has been a central topic of computer vision research in recent years (see literature review in Sec. 1.1). Very little work has been done to address the formidable problem of true 3D object categorization. In this work, we would like to learn a model for each class of the 3D objects so that given a new instance of the class in cluttered scenes, our algorithm can give the correct class label of the object as well as recognizing the pose (i.e. viewing angle) this particular instance comes from w.r.t. the model of the object class.

Our work is inspired by the handful of earlier works for 3D object recognition and categorization (see Sec. 1.1). We believe both geometry and appearance information of objects are critical for a true 3D object recognition. But we hypothesize that one does not need to go all the way to reconstructing a 3D object in order to recognize it. Rather, we believe that the usage of weak geometrical information, such as relative homographic transformations between object components, suffices to provide a robust representation of the 3D object. The main contributions of our paper are:

- We propose a novel 3D model of an object class by encoding both the appearance and 3D geometric shape information (see Fig.1). This model produces a compact yet powerful representation of an object class, differing from most of the previous works which store various image exemplars or model exemplars of different viewing angles. Given a novel testing image containing an object class, our algorithm not only classifies the object, but also infers the pose and scale, and localizes the object in the image.
- Toward the goal of learning a true 3D model of an object class, our algorithm demands less supervision in the learning process compared to previous works (i.e., [23]). Our algorithm is designed to handle either segmented or unsegmented objects in training. Most importantly, we do not need to label any views or scales, sort them in any particular order, or further group images of the same object instances.
- We offer an extensive experimental validation. We first show superior categorization and localization performances of our algorithm in a known dataset (Fig.5). In addition, we have compiled a new and very challenging 3D object dataset of 10 object categories (Fig.7) as a future benchmark for this task. Our algorithm demonstrates highly promising results on this dataset (Fig.6, Fig.8).

1.1. Literature review

Most of the recent advances in object categorization have focused on modeling the appearance and shape variability of objects under limited viewing point changes (*e.g.* [25, 27, 5, 7, 10, 17, 1]). This is reflected by the fact that most of the object category datasets contain images with such limitations (*e.g.* Caltech 101, UIUC car, etc.).

A much smaller number of works have investigated the problem of real 3D object category modeling. One approach is to treat a handful of typical poses as separate classes. The algorithm therefore learns a different model (or a mixture of models) for each of these classes [21, 26, 24].

978-1-4244-1631-8/07/$25.00 ©2007 IEEE

It is, however, theoretically unsatisfying that different 3D poses of the same object category model are completely independent in such methods. More importantly, these separate models are more likely to fire on false alarm cases and are computationally costly to train.

The closest work to our paper is Thomas et al. [23]. In their paper, the authors also incorporate shape and appearance information into a 3D object model. While the two approaches agree in the general spirit, our model provides a more compact and abstract representation of the 3D object. Furthermore, our algorithm is capable of learning a 3D class model under very weakly supervised condition, only requiring a class label (say 'cellphone') for each training image containing a cellphone. Compared to [23], we achieve better performance without sorting the images according to the instances, and aligning the poses of the objects. It is also worth mentioning very recent works by [15, 12] which have presented interesting new ideas toward the goal of representing object categories from multiple views.

Within the context of 3D *single* object modeling and recognition, authors have proposed different frameworks [20, 18, 8] to enforce geometrical and appearance constraints. [3] pioneered unsupervised recognition and reconstruction of object instances from an unlabeled set of images. These contributions, however, can be hardly generalized to the "categorical case". The geometrical relationship used to build the models are no longer suitable to handle shape and appearance variability within a given category. Our work differs from them fundamentally by overcoming these limitations by enforcing "milder" geometrical constraints across different object views. The theory behind aspect graphs (AG) [14, 2, 4] has also offered a powerful and compact representation for 3D objects. However, our model and AG are fundamentally different representations in that: i) AG are built on 'topologically-stable' views and no explicit mapping between these views is provided; our model instead is an ensemble of canonical parts linked together by an explicit homographic relationship; ii) AG only capture the 3D geometry, whereas our method jointly captures appearance and 3D relations; iii) it is unclear how to extend AG to handle generic 3D object categorization, whereas our model is specifically designed to do so.

2. A 3D Object Category Model

We highlight in this section the central ideas of our 3D model for an object category and compare them with relevant modeling schemes proposed in the literature. Fig.1 offers a schematic view of the core components of our model through a hypothetical 3D object category.

Appearance information is captured in the diagnostic parts of the objects in one class, denoted as P_i in Fig.1(b,c). Each "part" is a region of an object that tends to appear consistently throughout different instances of the same category (shown in colored regions in Fig.1(a)). It is a collection of a number of smaller image patches (feature points) usually provided by the feature detectors, constrained by some geometric consistency. Readers familiar with the current object recognition literature are reminded that our "part" is not a single detected local patch such as Harris corner or DoG detection, but rather a larger structure that contains many detected local patches. Given P_i, our model also encodes the appearance variations observed in training in the form of distributions of descriptors (Fig.1(b)).

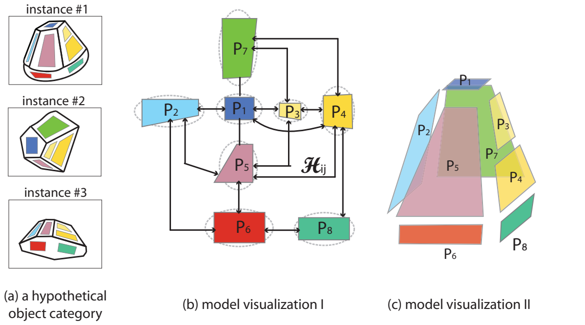

instance #1
instance #2
instance #3
P7
P2
P1
P3
P4
P5
$\mathcal{H}_{ij}$
P6
P8
P1
P2
P3
P7
P5
P4
P8
P6
(a) a hypothetical object category
(b) model visualization I
(c) model visualization II

Figure 1. Schematic illustration of the 3D object category model. **(a)** We illustrate our model by using a hypothetical 3D object category. Three instances of the hypothetical object are shown here as sample training images. The colored regions of each instance are "parts" of the objects that will be put together to form the final model. These "parts" are made of patches usually provided by feature detectors. When parts across different instances share the same color code, it indicates that the model has learnt the correspondences among these parts based on the appearance and geometric consistency. **(b)** The final model of the 3D object category can be viewed as a connected graph of the object parts (colored parts, P_i), part relations ($\mathcal{H}$), and the encoded variabilities of the appearances and geometric structure (visualized by the dashed ellipses surrounding each part). **(c)** A more intuitive visualization of the model puts together the parts in a 3D graph based of the learned geometric relations ($\mathcal{H}$). This figure is best viewed under color.

In our model, we call the diagnostic parts *canonical parts*; that is, they are representative of parts viewed in their most frontal position (Sec.3.3 & 3.4). Canonical parts P_i and P_j are then linked together if and only if P_j (or P_i) is visible when P_i (or P_j) is viewed in its most frontal position (canonical view). This linkage is characterized by the affine transformation $\mathcal{H}_{ij} = \begin{bmatrix} \mathcal{A}_{ij} & \mathbf{t_{ij}} \\ \mathbf{0} & 1 \end{bmatrix}$, where $\mathcal{A}_{ij}$ is a 2×2 affine transformation matrix, $\mathbf{t_{ij}}$ is a 2×1 translation vector, and $\mathbf{0}$ is a 1×2 0-vector. Thus, $\mathcal{H}_{ij}$ denotes the transformation to observe P_j when P_i is viewed in its canonical position. An example is shown in Fig.4. Notice that if two canonical parts P_i and P_j are viewed frontally from the same pose, $\mathcal{A}_{ij}$ is simply $\mathbf{I}$ and the transformation to observe P_j when P_i is viewed in its canonical position is just given by the translation vector $\mathbf{t_{ij}}$. This is the main constraint that previous literature has exploited in the context of the 2D representation of the object structure, e.g. [27]. We can interpret our linkage structure as its generalization to the multi-view case.

The above linkage structure between canonical parts is the key idea behind our model. Fig.1(b) shows these relationships through a connected graph. From Fig.1(b), one can infer the object class by "popping" it out into a 3D space (shown in Fig.1(c)). Similarly to the appearance information, it is also important to encode the intra-class variability of these geometric relationships. Notice that our linkage structure simultaneously captures connections between parts within the same view as well as across multiple views.

The advantage of our proposed linkage structure of canonical parts is that it provides a representation of the object geometry with a high level of abstraction. In fact, under the assumption that the total number of diagnostic part types is fixed, it can be shown that our linkage structure produces an *asymptotically* unique and stable representation of the object as we increase the number of views that are used to build that model. This is not true, for instance, for the model proposed by [23], where its complexity increases as function the number of views. This property is satisfied by the model proposed by [20] because it essentially achieves a full 3D reconstruction of the object. As such, however, the model of [20] fails to provide the necessary flexibility to handle the intra-class variability within each object category. Our proposed linkage structure is able to overcome this limitation.

Section 3 shows how we build such an object category model. Section 4 details how a novel instance of an object category is recognized, localized and its pose inferred. Extensive experimental results are shown in Section 5.

3. Building the model

We detail here our algorithm for building a 3D object class model from a set of training images. We assume that each training image contains one instance of the target object class. We do not, however, have information about the instance membership or pose of the object. The task of learning is to start with this set of raw images, extract features to form parts, obtain a set of *canonical parts* and finally form the 3D object class model by connecting these canonical parts in a 3D space.

3.1. Extract features

Local image patches are the basic building blocks of an object image. Our algorithm, however, works independently of any particular choice of feature detectors or descriptors [18, 19]). In practice, we choose the Saliency detector [13] and the SIFT descriptor [18] to characterize local features. An image i contains hundreds of detected patches, each represented as $f_i = (\mathbf{a_i}, \mathbf{x_i})$, where $\mathbf{a_i}$ is the appearance of the patch, described by a 128-dimension SIFT vector, and $\mathbf{x_i}$ is the location of the patch on the 2D image. Fig.2 shows two examples.

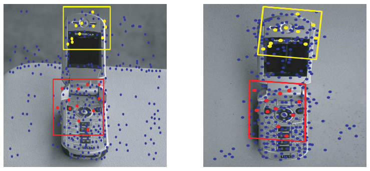


Figure 2. Detected features (patches) using the scaled invariant saliency detector [13]. All interest points are indicated by blue dots. The boxed regions in each image denote the learnt parts for this pair. When two parts across images share the same color (*e.g.* red boxes), they are connected by the algorithm. This figure should be viewed in color.

3.2. Form parts

We represent our 3D object category model in a hierarchical way. Local image features are first grouped into larger regions (called "parts"). A selected subset of these parts (according to appearance and geometric consistency) are then linked together as a full 3D model. This choice stems from the observation that larger regions of objects often carry more discriminative information in appearance and are more stable in their geometric relationships with other parts of the object [16].

The goal of this step is to group local image features into "parts" that are consistent in appearance and geometry across images. A global geometrical constraint is obtained by imposing that feature match candidates (belonging to different views) are related by the fundamental matrix $\mathcal{F}$. A local geometrical constraint is enforced by imposing that features belonging to a neighborhood are related by homographic transformation $\mathcal{H}_F$ induced by $\mathcal{F}$ [11]. We use a scheme based on RANSAC [9] to enforce such constraints while the optimal $\mathcal{F}$ and $\mathcal{H}_F$ are estimated. Below is a brief sketch of our algorithm.

```
1: Obtain a set of $M$ candidate features based on appearance similarity measured by $d(\mathbf{a_i} - \mathbf{a_j})$ across 2 training images.
2: Run RANSAC algorithm on $M$ to obtain a new (and smaller) set of matches $M_F \in M$ based on $\mathbf{x_i}\mathcal{F}\mathbf{x_j} = 0$, where $\mathcal{F}$ denotes the fundamental matrix.
3: Further refine the matches using RANSAC to obtain a set of $M_H$ matches such that $\mathbf{x}_i - \mathcal{H}_F\mathbf{x}_j = 0$, where $M_H \in M_F \in M$.
```

Step 2 and Step 3 can be iterated until the residual error computed on the inliers stops decreasing. Step 3 returns a pair of local neighborhood regions across the 2 training images in which all features $f_i \in M_H^{(i,j)}$ satisfy a vicinity constraint. We call them a matched "part". We follow this procedure for every pair of training images. Fig. 2 shows example parts indicated by boxes on these two cellphone images. Note that there is no presumed shape or size of these parts. For the rest of this model building algorithm, we represent the appearance of each of these parts as a normalized histogram of the occurrences of the codewords, or clustered local patches.

Implementation Details. On average our parts contain $50-200$ features, sufficient to effectively represent the local structure of the object from a particular view. We obtain on average $700-1000$ matched parts within a training set of 48 images. In our current implementation, we use a mask to remove spurious matches coming from the background. This is not a requirement for our algorithm to work. If there is enough variability in the background, [3] shows that spurious matches can be effectively removed by enforcing global constraints across all the views. Notice that, even if matched parts can be obtained from pairs of images belonging to different instances of a given category, the algorithm in 3.2 mostly produces matched parts from images belonging to the *same* object instance. This is due to the inherent lack of flexibility of RANSAC to handle intra-class variability. In fact, this guarantees robustness and stability in the part matching process. Actual matching of corresponding parts belonging to different object instances is achieved in the optimization process detailed in Sec. 3.4.

3.3. Find canonical parts candidates

Our goal is to represent the final object category with "canonical parts" and their mutual geometric relations. To do so, we need to first propose a set of canonical part candidates based on view-point criteria. What we have from our training images is a large set of "parts" that are paired across different images, each part consisting of a number of local features. Many of these parts linked across different images correspond to one *actual* part of the object. Fig. 3 is an illustration of the connected parts estimated from Step 3.2. The most possible front view of an actual object part defines a *canonical part candidate*. This will be by definition the canonical pose attached to the canonical part candidate. A canonical part candidate can be computed from the set of linked parts as follows.

Between every connected pair of parts, we associate them with a *factor of foreshortening* cost $\mathcal{K}_{ij}$. $\mathcal{K}_{ij}$ is a function of $\mathcal{A}_{ij}$ in the homographic relationship $\mathcal{H}_{ij}$ between these two parts. $\mathcal{H}_{ij}$ is provided by the algorithm in section 3.2. $\mathcal{K}_{ij} = \left(\lambda_1^{ij}\lambda_2^{ij} - 1\right)$, where $\lambda_{1,2}^{ij}$ are the two singular values of $\mathcal{A}_{ij}$. $\mathcal{K}_{ij}$ is greater than 0 when P_i is a less slanted version than P_j under affine transformation. Using the sign of $\mathcal{K}_{ij}$, we assign the direction between two parts. The full set of parts and their directed connections weighted by $\mathcal{K}_{ij}$ form a weighted directed graph (Fig. 3). It is easy to show that the path associated to the highest value of the total factor of foreshortening cost $\left(\sum_{(i,j)\in\text{path}} \mathcal{K}_{ij}\right)$ gives rise to a canonical part candidate. This can be identified as the part P attached to the terminal node of such maximum cost path. The intuition here is that the maximum cost path is the one that leads to the part with the smallest foreshortening, thus the canonical one. The maximum cost path can be found with a simple greedy algorithm.

Implementation Details. The graph structure is on average composed of $10-15$ parts but can go as low as 2, if a part is shared by only two views. For that reason, the greedy algorithm finds the optimal solution very quickly. Special care needs to be taken if the graph contains loops. This may occur when the orientation of a part is estimated with low accuracy from the previous step. Typically the number of canonical part candidates is $1/3$ of the initial set of part candidates.

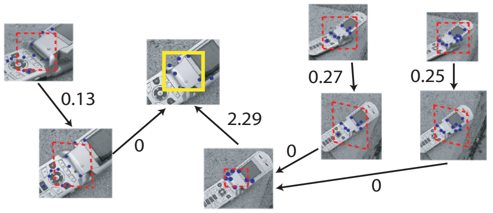


Figure 3. Illustration of linked parts for proposing one canonical part of the cellphone model using directed graph. The boxes indicate parts associated with this canonical part. The blue dots indicate detected local features within the parts. The yellow box is the proposed canonical part obtained by summarizing all *factors of foreshortening* (indicated by the numerical value adjacent to each arrow) given all the connected paths.

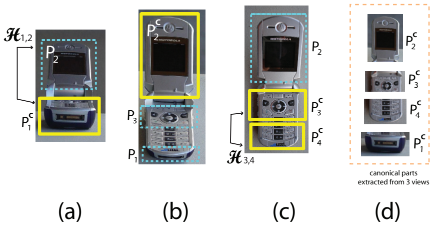


Figure 4. Illustration of the canonical parts and their geometric relations for three views of the same object (a,b,c). Each yellow box indicates the canonical part of interest that is viewed given its canonical pose (i.e. most frontal view by definition). Examples of canonical parts extracted from these three views are shown in (d). The dashed cyan boxes indicate parts that do not share the same canonical pose with the yellow canonical part. The cyan parts have a canonical counter part in a different pose. For instance, there exists a linkage structure between canonical parts P_1^c and P_2^c. The $\mathcal{H}_{12}$ denotes the transformation to observe P_2^c when P_1^c is viewed in its canonical position (thus, generating cyan P_2). In (c), two canonical parts P_3^c and P_4^c share the same canonical pose. In this case, the transformation $\mathcal{H}_{34}$ is just a translation because P_3^c and P_4^c are canonical at the same time.

3.4. Create the model

Sec. 3.3 has proposed a number of canonical part candidates from the training images. So far, we have only utilized local appearance or pair-wise geometry information to find correspondences between parts and find the canonical part candidates. Now we are ready to take all these candidates to obtain a canonical part at the categorical level. This allows to propose a 3D object category model by finding a globally consistent and optimal combination of canonical parts.

We call a canonical part of a given category P_i^c. Given two different canonical part P_i^c and P_j^c, there are two ways that they are placed with respect to each other onto the 3D object model. In the first case, when P_i^c is viewed frontally,

P_j^c is also viewed frontally (Fig. 4(c)). In this case the homographic linkage between these two canonical parts is $\mathcal{H}_{ij} = \begin{bmatrix} \mathbf{I} & \mathbf{t}_{ij} \\ \mathbf{0} & 1 \end{bmatrix}$, where $\mathbf{t}_{ij}$ is the translation vector between P_i^c and P_j^c. In the second case, P_i^c and P_j^c are not viewed frontally simultaneously. They are, therefore, related by a full homography $\mathcal{H}_{ij} = \begin{bmatrix} \mathcal{A}_{ij} & \mathbf{t}_{ij} \\ \mathbf{0} & 1 \end{bmatrix}$. $\mathcal{H}_{ij}$ denotes the transformation to observe P_j when P_i is viewed in its most front view position. Parts P_1^c and P_2^c in Fig. 4(a,b) have this type of linkage. $\mathcal{H}_{ij}$ captures both the 2D relationship (e.g., position) between canonical parts as well as a soft 3D relationship which is provided by the affinity transformation $\mathcal{A}_{ij}$ between parts. Canonical parts that are not connected correspond to sides of the object that can never be seen at the same time.

Given the pool of candidate canonical parts from all the instances of a given category, we wish to calculate the set of canonical parts at the categorical level. This can be done by matching corresponding candidate canonical parts across all the instances. This correspondence problem can be solved by means of an optimization process that jointly minimizes the appearance difference between matching candidates and their corresponding linkage structure $\mathcal{A}_{ij}$. The 1st row of Fig. 8 (3^{rd} and 4^{th} columns) shows an illustration of the learnt cellphone model.

Implementation Details. The optimization is carried out by exploiting similarity of appearance and the estimated linkage structure between canonical part candidates belonging to different object instances. Here, the appearance similarity is computed as a chi-square distance between the histograms representing the canonical region appearances. A matching scheme such as in [1] is also possible. Similarity of linkage structure is computed by comparing $\mathcal{A}_{ij}$ for every pairs of canonical part candidates P_i, P_j. Notice that this optimization step greatly benefits of the fact that parts-to-be-matched are canonical. This means that all the parts are already normalized in term of their viewing angle and scale. Furthermore, the number of canonical part candidates is a small subset of the initial number of parts. All this greatly simplifies the matching process which could have been hardly feasible otherwise.

3.5. Interpolating views in the 3D model

In this section we study how the linkage structure is modified when the object is not viewed from a canonical view. Consider canonical views V_i and V_j of a given object class. Our goal is to express the relationship between canonical parts P_i and P_j for an intermediate arbitrary view V_s as function of an interpolation parameter s. We can write down the following equations for the viewing constraints:

$$\begin{aligned} P_i^s &= \mathcal{H}_s\left((1-s)\overline{\mathcal{H}}_i P_i^c + s\overline{\mathcal{H}}_j A_{ji} P_i^c\right) \\ P_j^s &= \mathcal{H}_s\left(s\overline{\mathcal{H}}_j P_j^c + (1-s)\overline{\mathcal{H}}_i A_{ij} P_j^c\right) \qquad (1) \end{aligned}$$

where P_i^s and P_j^s are the canonical parts P_i and P_j (respectively) viewed from V_s; here P_i and P_j are expressed in term of the local feature locations composing the part; $\overline{\mathcal{H}}_i$ and $\overline{\mathcal{H}}_j$ are the rectification homographies from view i and j to their corresponding rectified views ($\overline{\mathcal{H}}_i$ and $\overline{\mathcal{H}}_j$ are known given P_i^c and P_j^c); $\mathcal{H}_s^{-1}$ transforms view V_s into its corresponding rectified view. It is interesting to compare these constraints with those introduced in [22].

Similarly, we can write an additional equation for the spatial translation constraint (related to $\mathbf{t}$):

$$\mathbf{t}_{ij}^s = \mathcal{H}_s\left((1-s)\overline{\mathbf{t}}_{ij} + s\overline{\mathbf{t}}_{ji}\right) \qquad (2)$$

where $\overline{\mathbf{t}}_{ij}$ and $\overline{\mathbf{t}}_{ji}$ are the translation vectors in the rectified views. Thus, Eq. 1 and Eq. 2 allow us to estimate P_i^s and P_j^s from canonical regions P_i and P_j such that the interpolation between views V_i and V_j is geometrically consistent. If we approximate $\overline{\mathcal{H}}_i \simeq \overline{\mathcal{H}}_j \simeq I$, then Eq. 1 and Eq. 2 would lead to a simpler linear interpolation scheme. Finally, if the object is viewed from a canonical view (say, V_i), then $s = 0$, $P_i^s = \mathcal{H}_s P_i^c$, $P_j^s = \mathcal{H}_s A_{ij} P_j^c$, and $\mathbf{t}_{ij}^s = \mathcal{H}_s \mathbf{t}_{ij}$.

4. Recognizing a new image

Given a new image, our task is to detect whether a particular type of object exists, localize where the instance is, and estimate the pose orientation. Similarly to the training procedure, we go through the following main steps: extract image features, propose candidate canonical parts, and then finally match an object model to the image via a global optimization procedure using the candidate canonical parts.

4.1. Extract features and get part candidates

We follow the same procedure to find local interest points by using the Saliency detector [13]. Each detected patch is then characterized by a 128-dimension SIFT vector [18]. Given an object model, say the "cellphone" model, we first find a list of canonical part candidates by the following procedure. For each canonical part of the model, we greedily search through the test image by a scanning window across pixel locations, scales and orientations. Canonical parts and test parts are matched by comparing the distributions of features belonging to the relevant regions. The most probably N firings (typically 5) are retained as the N candidates for a canonical part P_i^c.

4.2. Match a model to the image

Let's denote P_h^c and P_k^c as two canonical parts from the object model, and P_i^t and P_j^t as two candidate parts from the test image, corresponding to the learnt canonical parts respectively. The goal in recognition is to find a set of optimal matches of $\{P_i^t, P_j^t, ...\}$ with respect to the object model, defined by the canonical parts $\{P_h^c, P_k^c, ...\}$ and their geometric relations $\{\mathcal{H}_{hk}^c, ...\}$.

In the optimization procedure, just like in the training, two canonical parts are related to each other by a full

$\mathcal{H}_{kh}$ transformation. In other words, when one is viewed frontally, the other is foreshortened, and the amount of foreshortening depends on $\mathcal{H}_{kh}$. The optimization equations can be derived from Eq. 1 and Eq. 2 since the object in the test image can be at an arbitrary pose:

$$\begin{cases} P_i^t - \mathcal{H}\left((1-s)P_h^c + sA_{hk}P_h^c\right) = 0 \\ P_j^t - \mathcal{H}\left((1-s)P_k^c + sA_{kh}P_k^c\right) = 0 \\ \mathbf{t}_{ij}^t - \mathcal{H}\left((1-s)\mathbf{t}_{hk}^c + s\mathbf{t}_{kh}^c\right) = 0 \end{cases} \tag{3}$$

where $\mathcal{H}$ accomodates for changes in scale and rotation between canonical parts and candidate parts, and s is the interpolation parameter. Notice that these general equations contain the special case when the test object is observed from a canonical view (say V_h), then $s = 1$, $P_i^t = \mathcal{H}P_h^c$, $P_j^t = \mathcal{H}A_{hk}P_k^c$ and $\mathbf{t}_{ij}^t = \mathcal{H}\mathbf{t}_{kh}^c$. An interesting case arises when the object is viewed canonically and P_h^t, P_k^t are canonical simultaneously; then: $P_i^t = \mathcal{H}P_h^c$, $P_j^t = \mathcal{H}P_j^c$ and $\mathbf{t}_{ij}^t = \mathcal{H}\mathbf{t}_{kh}^c$. These are the typical 2D matching constraints used in 2D part based recognition.

The above optimization equations form an over-constraint system with three unknowns (s and $\mathcal{H}$). When the optimization process converges, a residual error for all possible matches can be computed from Eq. 3. Thus, for each testing image, we obtain a minimal residual score given by each object model. The object class label is then determined by the model that corresponds to the lowest residual score.

The pose of the object in the testing image can be estimated by projecting back to the combination of canonical parts that give rise to the lowest residual score on the selected object class model. Each canonical part is attached to a canonical view, and the dominant canonical view gives the estimate of the actual object pose. We can use the parameter s to refine the pose estimate between pairs of canonical views. Finally, scale and rotation parameters in $\mathcal{H}$ allow estimating the change in scale and rotation of the test object with the respect to the winning object model.

5. Experiments

5.1. Exp. I: Comparison with Thomas et al. [23]

We first conduct experiments on a known 3D object class dataset (the motorbikes) used by Thomas et al. [23], provided by PASCAL Visual Object Classes (VOC) Challenge [6]. For fair comparison, we use the same testing images in both these classes as in [23]. Specifically, 179 images from the 'motorbikes-test2' testing set are used. The models are learnt by using the provided image set of motorbike instances (each instance has 12-16 poses). We evaluate the categorization and localization performance by using precision-recall curves, under the same conditions as stated by [23]. Fig.5 illustrates that our algorithm significantly outperforms [23].

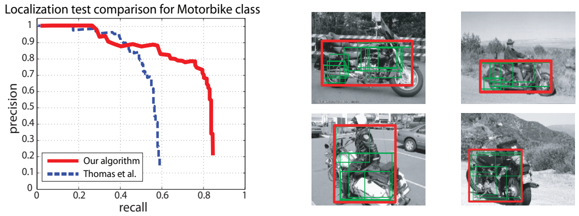
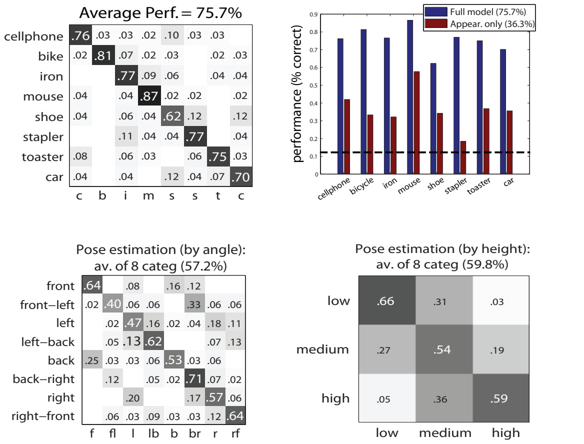


Figure 5. Localization experiment compared with [23]. The precision-recall curves are generated under the PASCAL VOC protocol. Examples of detections are shown on the right. This figure is best viewed in color.

5.2. Exp. II: A New 3D Object Class Dataset

Due to the lack of established works in 3D object categories, it is difficult to obtain a standard dataset to compare our results with. Here we offer a new 3D object category dataset of 10 very different everyday objects: car, stapler, iron, shoe, monitor, computer mouse, head, bicycle, toaster and cellphone (Fig.7).



	c	b	i	m	s	s	t	c
cellphone	.76	.03	.03	.02	.10	.03	.03	
bike	.02	.81	.07	.02	.03		.02	.03
iron			.77	.09	.06		.04	.04
mouse	.04		.04	.87	.02	.02		.02
shoe	.04		.06	.04	.62	.12		.12
stapler			.11	.04	.04	.77		.04
toaster	.08		.06	.03		.06	.75	.03
car	.04		.04		.12	.04	.07	.70

	f	fl	l	lb	b	br	r	rf
front	.64		.08		.16	.12		
front–left	.02	.40	.06	.06		.33	.06	.06
left		.02	.47	.16	.02	.04	.18	.11
left–back		.05	.13	.62			.07	.13
back	.25	.03	.03	.06	.53	.03	.06	
back–right		.12		.05	.02	.71	.07	.02
right			.20			.17	.57	.06
right–front		.06	.03	.09	.03	.03	.12	.64

	low	medium	high
low	.66	.31	.03
medium	.27	.54	.19
high	.05	.36	.59

Figure 6. (Top Left) Confusion table results for 8 categories of objects. (Top Right) Comparison with the bag of words model. The dashed line indicates a random chance performance. (Bottom Left) & (Bottom Right) Pose estimation results. Notice that the most confusion takes place between symmetric poses such as frontal-vs-back, left-vs-right, etc...

Given our dataset, we test our model on a 8-category classification task (excluding heads and monitors). To learn each category, we randomly select 7 object instances ($\sim$ 280 images) to build the model, and 4 novel object instances ($\sim$ 70 images) for testing. The fartherest scale is not considered in the current results. Fig.8 is a detailed summary and explanation of our results.

Fig.6 is a summary of the classification results given 8 object categories. We achieve an average performance of 75.65% on 8 classes. An important observation to make is that our model is far more superior than a simple bag of words model, where only the appearance information is captured. Fig.6 (top-right) compares the average classification result of our model with a bag of words model similarly to [5]. Fig.6 (bottom) are pose estimation results. This is an extremely challenging task. The algorithm not only needs to make a correct overall class decision, but also chooses the correct pose (angle and height) of the object.

6. Discussions

We have proposed a novel model to represent 3D object classes. To tackle this formidable task, we emphasize on learning both the diagnostic appearances of the objects as well as the 3D geometric relationships among these parts. We would like to make the following concluding remarks of this paper.

- While we have proposed a model for object categorization, the general framework is useful for any degrees of variability in appearances and/or geometry. It would be interesting to see how our model might offer a unified approach to learn and detect both individual objects and object categories.
- We consider our model representation the most important contribution of this work. There is still much room to improve the learning scheme. We are planning to cast this into a probabilistic framework for training and recognition.
- A number of issues remain unexplored. We would like to explore how feature detectors and descriptors might influence the robustness of the model. Furthermore, much is left to be done in recognition, where one is inevitably searching in a large space of possibilities.

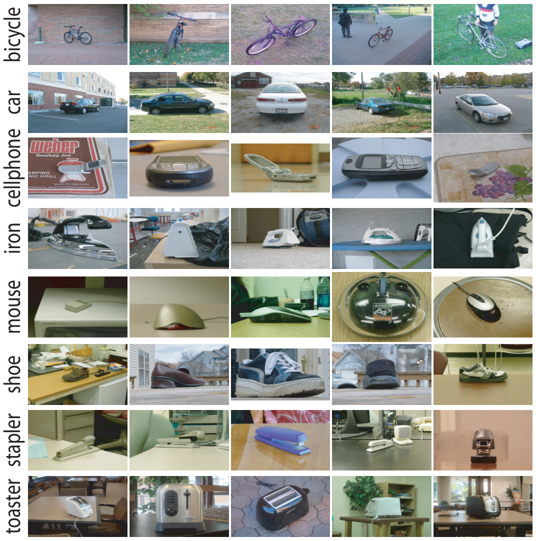


Figure 7. Sample images from our 3D object category dataset. Our dataset consists of 10 different everyday object categories under 8 viewing angles, 3 heights and 3 scales for a total number of $\sim$ 7000 images.

7. Acknowledgements

Dr S.S. and Dr L.F-F research is supported by a Beckman Fellowship and a Microsoft New Faculty Fellowship respectively.

References

[1] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *Proc. Comp. Vis. and Pattern Recogn.*, 2005.

[2] K. Bowyer and R. Dyer. Aspect graphs: An introduction and survey of recent results. *Int. Journal of Imaging Systems and Technology*, 2(4):315–328, 1990.

[3] M. Brown and D. Lowe. Unsupervised 3D object recognition and reconstruction in unordered datasets. In *5th International Conference on 3D Imaging and Modelling (3DIM05)*, Ottawa, Canada, 2005.

[4] C. Cyr and B. Kimia. A similarity-based aspect-graph approach to 3D object recognition. *International Journal of Computer Vision*, 57(1):5–22, 2004.

[5] C. Dance, J. Willamowski, L. Fan, C. Bray, and G. Csurka. Visual categorization with bags of keypoints. In *ECCV International Workshop on Statistical Learning in Computer Vision.*, Prague, 2004.

[6] M. e. Everingham. The 2005 pascal visual object class challenge. In *Selected Proceedings of the 1st PASCAL Challenges Workshop*, to appear.

[7] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. Computer Vision and Pattern Recognition*, pages 264–271, 2003.

[8] V. Ferrari, T. Tuytelaars, and L. Van Gool. Simultaneous object recognition and segmentation from single or multiple model views. *IJCV*, April 2006.

[9] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. In *Comm. of the ACM.*, volume 24, pages 381–395, 1981.

[10] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *In Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Beijing, China, 2005.

[11] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[12] D. Hoeim, C. Rother, and J. Winn. 3D layoutcrf for multi-view object class recognition and segmentation. In *Proc. In IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[13] T. Kadir and M. Brady. Scale, saliency and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[14] J. Koenderink and A. van Doorn. The singularities of the visual mappings. *Biological Cybernetics*, 24(1):51–59, 1976.

[15] A. Kushal, C. Schmid, and J. Ponce. Flexible object models for category-level 3d object recognition. In *Proc. In IEEE Conf. on Comp. Vis. and Patt. Recogn.*, 2007.

[16] S. Lazebnik, C. Schmid, and J. Ponce. Semi-local affine parts for object recognition. In *Proc. of BMVC*, volume 2, pages 959–968, Kingston, UK, 2004.

[17] B. Leibe and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Proc. Workshop on satistical learning in computer vision*, Prague, Czech Republic, 2004.

[18] D. Lowe. Object recognition from local scale-invariant features. In *Proc. International Conference on Computer Vision*, pages 1150–1157, 1999.

[19] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *IJCV*, 60(1):63–86, 2004.

[20] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *IJCV*, 66(3):231–259, March 2006.

[21] H. Schneiderman and T. Kanade. A statistical approach to 3D object detection applied to faces and cars. In *Proc. CVPR*, pages 746–751, 2000.

[22] S. Seitz and C. Dyer. View morphing. In *SIGGRAPH*, pages 21–30, 1996.

[23] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and L. Van Gool. Towards multi-view object class detection. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1589–1596, 2006.

[24] A. Torralba, K. Murphy, and W. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 762–769, 2004.

[25] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. Computer Vision and Pattern Recognition*, volume 1, pages 511–518, 2001.

[26] M. Weber, W. Einhaeuser, M. Welling, and P. Perona. Viewpoint-invariant learning and detection of human heads. In *Proc. 4th Int. Conf. Autom. Face and Gesture Rec.*, pages 20–27, 2000.

[27] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *Proc. European Conference on Computer Vision*, volume 2, pages 101–108, 2000.

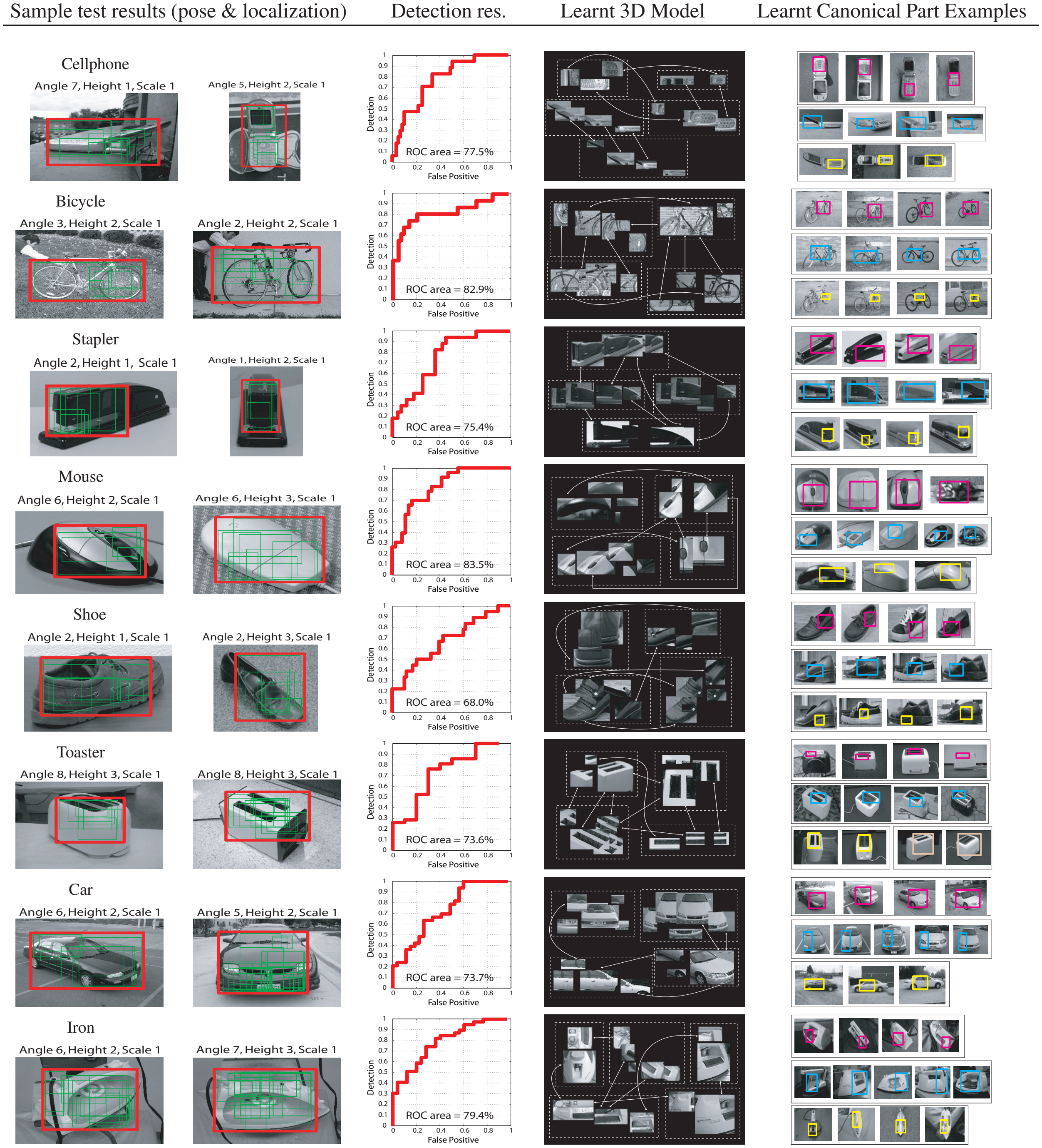


Figure 8. Summary of the learnt 3D object category models, sample test images and binary detection results (ROC). This figure should be viewed in colors. **Column 1** presents two correctly identified sample testing images. The red bounding box on each image indicates the best combination of canonical parts (i.e., that of the smallest error function), whereas the thin green boxes inside the red box correspond to the canonical parts detected on the object. Using the pose estimation scheme, we are able to predict which pose this particular instance of the model comes from. **Column 2** shows the binary detection result in ROC curves. **Column 3** visualizes the learnt model of each object category. We show here a single object instance from the training images. Each dashed box indicates a particular view of the object instance. A subset of the learnt canonical parts is presented for each view. Across from different views, the canonical parts relationships are denoted by the arrows. Note that for clarity, we only visualize a small number of canonical parts as well as their $\mathcal{H}$. **Column 4** illustrates the appearance variability of a given canonical part. For each object model, 3 or 4 canonical parts are shown, indicated by the boxes. For each canonical part (i.e. within each box), we show a number of examples that belong to the same part. Note that these parts not only share a similar appearance, but also similar locations with respect to the object. This figure is best viewed in color and with magnification.