Learning Object Shape: From Drawings to Images

Gal Elidan[1]
Dept. of Computer Science
Stanford University
galel@cs.stanford.edu

Geremy Heitz[1]
Dept. of Electrical Engineering
Stanford University
gaheitz@cs.stanford.edu

Daphne Koller
Dept. of Computer Science
Stanford University
koller@cs.stanford.edu

Abstract

We consider the important challenge of recognizing a variety of deformable object classes in images. Of fundamental importance and particular difficulty in this setting is the problem of "outlining" an object, rather than simply deciding on its presence or absence. A major obstacle in learning a model that will allow us to address this task is the need for hand-segmented training images. In this paper we present a novel landmark-based, piecewise-linear model of the shape of an object class. We then formulate a learning approach that allows us to learn this model with minimal user supervision. We circumvent the need for hand-segmentation by transferring the shape "essence" of an object from drawings to complex images. We show that our method is able to automatically and effectively learn and localize a variety of object classes.

1 Introduction

Recognizing objects in images is a surprisingly difficult task that has challenged the field of computer vision from its inception. Recent works address this problem using a variety of methods with significantly different goals, which broadly fall into three categories. The first is simply to recognize the existence of an object in the image. Such techniques commonly use sophisticated image features that are often unrelated to the object structure or location in the image [22, 14, 2]. A second, more ambitious goal is to roughly localize the different parts of the object toward better recognition [13, 12, 19]. Although these models try to account for shape and localization, evaluation with respect to a localization measure is generally not performed. Finally, some work [7, 15] has been done on precise object localization, but this usually focuses on very specific object classes.

[1] These authors contributed equally to this manuscript

In this paper we address the goal of modeling the shape of a general object class in 2D, and using such a shape model to automatically outline objects in the class in real images. The outlining task is interesting in its own right, as it is important for providing a detailed description of the objects in a scene and their inter-relationships. Moreover, developing a good model of an object's basic shape appears to be an important step in utilizing background knowledge (e.g., the general shape of quadrupeds) for specific recognition tasks.

We describe a flexible, probabilistic object model that captures the contour of the object's deformable shape and its characteristics. In contrast to early work on modeling object shape [3], our model is semi-parametric: it is defined in terms of landmarks, each associated with local contour information; the landmarks are connected with line segments, thereby defining a complete object shape. The probabilistic model is formulated as a *Markov random field (MRF)* [18] over these landmarks, allowing the task of outlining the object to be solved using standard MRF inference algorithms.

A key bottleneck in learning a contour model is the need for training instances of the object outline. One option is to hand outline the object in training images. This approach is laborious, and hard to scale to a large number of classes. We circumvent this problem by training our shape models, in an unsupervised way, using simple (cartoon) drawings of objects in the target class. In drawings, the fundamental contour of the object is clearly visible, avoiding many of the problems (such as clutter and shading variations) associated with complex images. Thus, we learn a model of the "fundamental" shape of the object from cartoon drawings, and transfer it to the task of recognizing objects in real images. We show that our method effectively bootstraps from the simple drawings, and that learning from drawings is comparable to learning from hand-segmented examples. Overall, we demonstrate that our general framework is applicable to very different classes, and that it is able to achieve high performance in outlining them in images.

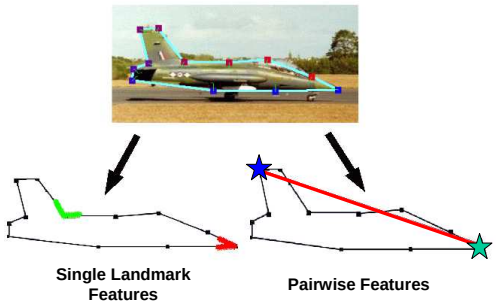

Figure 1: Illustration of a landmark based shape model for the airplane object. Shown are the landmarks outlining the airplane as well as examples of local shape template features and pairwise distance features.

2 Shape Model and Object Outlining

We represent the shape of a class of objects as a probabilistic model $\mathcal{M}$ over a set of landmarks $\mathcal{L}$; Figure 1 illustrates the model for the airplane object. The outline of the object is defined by the piecewise-linear contour that connects adjacent landmark points. In addition to the outline, $\mathcal{M}$ includes different features of individual landmarks, including location, local edge shape, image appearance as well as features of pairs of landmarks, including geometric relationships. All these elements are probabilistically constructed to represent the intraclass variation of the object's shape. In this section, we present the probabilistic model, deferring the specifics of the features used to Section 3.

To outline an object in an image I with a set of pixels $\mathcal{X}_I$, we want to register each of the landmarks $l \in \mathcal{L}$ to a given pixel location $p_l \in \mathcal{X}_I$. We define the localization instance $\mathbf{L}_I$ as a mapping, or *correspondence*, of landmarks to pixels $\mathbf{L}_I : \mathcal{L} \rightarrow \mathcal{X}_I$. Our goal is then to define a probability distribution $P(\mathbf{L}_I \mid I, \mathcal{M})$ which evaluates the extent to which the image characteristics at the landmark locations fit those predicted by the model.

We define a set of features for each landmark l, based on the image and the pixel p_l to which the landmark is assigned. Intuitively, these features should quantitatively evaluate how well the landmark l fits the candidate pixel location p_l (see Section 3). We use $f_i^l(I, p_l)$ for the i^{th} feature of landmark l, and note that it is a function of the image I, and the pixel location p_l. Pairs of landmarks l, m are also associated with pairwise features $f_j^{l,m}(I, p_l, p_m)$ that evaluate the quality of their jointly assigned pixel locations p_l, p_m. The object model $\mathcal{M}$ defines the full set of feature *functions* for individual and pairs of landmarks. Actual values of these features are computed relative to a specific image I and a landmark correspondence $\mathbf{L}_I$.

We define our probability distribution over landmark correspondences using a *Markov random field (MRF)*, whose variables correspond to landmarks; the domain of landmark variable l is a set of candidate image pixels $p_l \in \mathcal{X}$, as well as an 'absent' value (i.e., the landmark does not appear in the image). Given a model $\mathcal{M}$ of an object class and an image I, the likelihood of a given localization assignment $\mathbf{L}_I$ is:

$$P(\mathbf{L}_I \mid I, \mathcal{M}) \propto \prod_{l \in \mathbf{L}_I} \exp\left\{\sum_i f_i^l(I, p_l)\right\} \times \prod_{l,m \in \mathbf{L}_I} \exp\left\{\sum_j f_j^{l,m}(I, p_l, p_m)\right\}. \quad (1)$$

The probability of the landmark-to-pixel assignment $\mathbf{L}_I$, and thus that of the outline, is high if the aggregation of the multiple cues (features) to its existence is strong. Thus, object outlining is a *registration* task — finding the landmark-to-pixel correspondence that maximizes $P(\mathbf{L}_I|I, \mathcal{M})$. This is simply the most likely (MAP) assignment relative to the distribution defined in the MRF of Eq. (1), which can be found using a variety of algorithms; we use the simple but effective max-product algorithm [18].

However, it is computationally infeasible to allow all pixels in the image as the domain of l: Even for a relatively small image, with 300×200 pixels, the pairwise potentials have size $60,000^2$. Thus, for each landmark l, we select a restricted subset of pixels $\mathcal{X}^l \subset \mathcal{X}$ to serve as candidates for the landmark location. To this end, we first transform the image into an edge map using the Canny edge detector [4] and we prune out all pixels $p \in \mathcal{X}$ which do not lie on a detected edge point. This reduces the space of candidate pixels to a few hundreds or thousands. To further prune our domain, we choose the best 50 edge pixels according to our shape template feature $f_1^l(I, p_l)$ (see Section 3) to serve as the MRF assignment domain $\mathcal{X}^l$ for landmark l.

To summarize, our model $\mathcal{M}$ contains both local and pairwise information about the object. Given an image in which we want to outline an object, we use an MRF to define a probability distribution over possible assignments $\mathbf{L}_I$ of landmarks to pixels. We use a standard MAP inference algorithm to find the (approximately) most likely assignment, which in turn defines the outline of the object.

3 A "Shape Aware" Object Model

As discussed in the previous section, our model defines a set of local feature functions $f_i^l(I, p_l)$ for each landmark l, and a set of pairwise features $f_i^{l,m}(I, p_l, p_m)$ for landmark pairs. The local landmark features in our model encode information about the expected image characteristics in the vicinity of the pixel p_l to which the landmark l is assigned. Importantly, all of the features we consider, be they edge or appearance based, are "shape aware" and contribute in con-

cert toward a well defined yet probabilistically deformable object outline.

Shape Template. The first feature we consider is aimed at capturing the shape directly. We use a template of the edge points surrounding the landmark l. While a common approach is to consider all edge points in a ball or a patch centered at the landmark, we construct our template using regularly sampled points along the contour, up to some "geodesic" distance away from the landmark (measured along the contour itself). Our shape template is defined by two "arms" emanating from the landmark position, where each arm is defined by a set $\mathbf{O}$ of offset means and variances $\mathbf{o}_k = (\mu_k, \sigma_k^2)$, such that each offset mean is relative the previous point. The set of offset means defines an "average" shape which we expect to see at the landmark; Figure 1 illustrates two such average shapes for the nose and body of the airplane.

In an ideal image of the object, we expect each point along the average shape to generate an edge pixel in the image. In reality, we have to allow both for local deformation of the shape around the landmark (accounting, among other things, for class variability) and for missing edge pixels and edge detection noise. We therefore select a contour for each arm as follows: Letting p_l^0 be the pixel assigned to the landmark l, we choose p_l^k be the edge pixel (in the Canny edge map) closest to $p_l^{k-1} + \mu_k$. Assuming a Gaussian distribution on the offsets, the divergence (negative log probability) of the shape template offsets for a landmark l assigned to a pixel point p_l is

$$f_1^l(I, p_l) = -\sum_k \log \mathcal{N}\left(p_l^k; \mu_k, \sigma_k^2\right)$$

where $\mathcal{N}(p_l^k; \mu_k, \sigma_k^2)$ is the Gaussian density parameterized by the mean and variance μ_k, σ_k^2 associated with the landmark l and the kth point on the appropriate arm relative to the landmark (several relevant indices have been omitted for clarity). The closer the edge pixels are to the expected offset locations of the shape template, the smaller the divergence and the higher the corresponding potential value. Note that the above formulation of shape template divergence allows the shape to flexibly deform as the expected location of p_l^k follows the "assignment" of p_l^{k-1} to the closest edge point.

Appearance Templates. Our goal is to use appearance templates that also take into account the shape of the object. To do so, we evaluate different appearance characteristics (filter responses) in a square patch centered at each landmark. However, rather than using the common approach of evaluating the patch as a whole (e.g., [13]), we distinguish between characteristics on the inside and outside of the object. More precisely, we use the average shape template of each landmark l as a mask M_l, which separates the region around the assigned landmark position p_l into an "inside" region and an "outside" region. This mask can be applied to other local features, turning standard features into "shape aware" ones.

To reduce the sensitivity to small geometric distortions or transformations, we do not consider the values of individual pixels; rather, we summarize the appearance characteristics in two distinct histograms, one over the set of the inside pixels and one over the outside pixels. Concretely, consider a patch centered around p_l and a particular filter F_n. For each pixel p in the patch, we compute the response of F_n, and quantize the value into discrete bins. We then compute a histogram α_n^l of these values for the inside of the object as defined by the mask M_l and similarly compute β_n^l for the outside of the object. These histograms are then used in the computation of the feature value. We generate these histograms for three types of features. The first histogram pair (α_H, β_H), represents distributions over the hue value of the pixel in HSV space (the H component), and is placed into 8 uniformly spaced bins between 0 and 360. The second pair (α_L, β_L) represents a distribution of intensity value (luminance) of the pixel binned into 8 uniformly spaced intervals. The final pair represents texture; following the work of Malik *et al.* [16], we first compute a response to various Gabor filters [9] at each pixel across 6 orientations, 3 scales, and 3 frequencies. The responses to these filters are vectorized to produce a 54-dimensional vector for each pixel. The responses at regularly sampled pixels across a large set of training images are then clustered using K-means to provide the histogram bin centroids for the descriptors α_T, β_T.

We use the appearance filters to construct a segmentation-like feature that favors different appearance distributions between the inside and outside of the object. We formalize this intuition using the commonly used and effective Earth Movers Distance (EMD) [21] and compute the feature value as a weighted sum:

$$f_2^l(I, p_l) = \sum_{c \in \{H,L,T\}} w_c \, \text{EMD}(A_c^I || A_c^O)$$

where A_c^I denotes the inside appearance histogram values of type c and A_c^O is used for the outside histograms. Consequently, a large difference between the pixels inside and outside of the object mask in the vicinity of the landmark will increase the probability of the landmark assignment. The weights w_c were chosen to emphasize intensity (1) and texture (1) over hue (0.5).

Location Prior. The final local feature we consider is a feature inspired by Fei-Fei *et al.* [11]. As they show, an effective localization prior is important when a model is constructed from few instances. In our setting, we simply use an uninformed localization prior relative to the center of the image, and construct this prior only from the car-

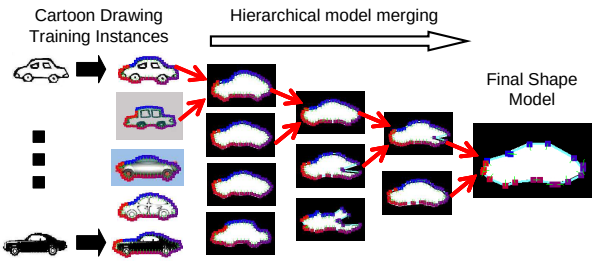

Figure 2: Schematic representation of our learning procedure. We begin with a set of cartoon drawings from which we extract high resolution outline contours. These contours are then hierarchically corresponded to each other to select the set of landmarks that make up the model. Finally, a model $\mathcal{M}$ is constructed from the cartoons and their correspondences to the learned landmarks

toon drawings. For instance, in the airplane model shown in Figure 1, and indeed for all cartoon airplanes as well, the airplane tail is generally in the upper left quadrant of the image center. Specifically, we model the X, Y location of landmark l as a Gaussian with mean $(\mu_{l,X}, \mu_{l,Y})$ and variances $(\sigma^2_{l,X}, \sigma^2_{l,Y})$. With these parameters, we define the location feature value to be:

$$f^l_0(I, p_l) = \log \mathcal{N}(p_{lX}; \mu_{l,X}, \sigma^2_{l,X}) + \log \mathcal{N}(p_{lY}; \mu_{l,Y}, \sigma^2_{l,Y}).$$

Pairwise Features. Features for landmark pairs allow for interactions that occur more globally in the image. An obvious pairwise property that is closely tied to the object shape is the geometric relationship between the landmarks. Consider landmarks l and m assigned to pixels p_l and p_m, respectively. Let $dx = p_{lX} - p_{mX}$ denote the horizontal offset between the two locations measured in pixels. Modeling the distribution over such an offset using a Gaussian with mean $\mu_{lm,\Delta X}$ and variance $\sigma^2_{lm,\Delta X}$, we compute the corresponding feature value as the log-probability of the offset:

$$f^{l,m}_X(I, p_l, p_m) = \log \mathcal{N}(dx; \mu_{lm,\Delta X}, \sigma^2_{lm,\Delta X})$$

We use a similar feature to model the the Y offset, denoting this feature by $f^{l,m}_Y(I, p_l, p_m)$.

An important choice with respect to pairwise features is which pairs to include in the model. This decision has important ramifications in terms of the complexity of the MRF we use for registration (see Section 2). In particular, using all pairs of landmarks creates a fully connected graph with $O(L^2)$ edges, leading to a difficult inference problem. Therefore, we include in the model only two types of pairs: neighboring landmarks, which account for the local geometry; and pairs of landmarks that lie diametrically "opposite" to each other (farthest apart along the contour), which impose enough global constraints to restrict the degrees of freedom of the overall object shape. The total number of pairwise potentials is therefore linear in the number of landmarks. In early experiments, we also tried using all landmark pairs; this approach did not improve performance, but required an order of magnitude longer running time.

4 Learning the Shape Model

In previous sections we described the elements of our object shape model $\mathcal{M}$ and how this model is registered to an image. We now turn to the problem of learning a model $\mathcal{M}_{class}$ for some particular class. One might consider learning shape models in an unsupervised way, using training images where we know only that the object is in the image. However, for the rich shape models that we have here, this process is highly susceptible to very poor local maxima, and is unlikely to work well without significant prior knowledge.

Conversely, to turn the problem into a fully supervised learning task, we would have to obtain a set of outlines for multiple objects within the class, each tagged with a consistent set of landmarks. The task of generating such data for a large number of object classes is a laborious one. We circumvent this bottleneck using two ideas. The first is the use of simple (cartoon) drawings of an object as training data. In such simple images, we can automatically extract a high-resolution contour using a Gradient Vector Flow snake algorithm [24]. However, even given these contours, we must still generate a consistent set of landmarks for all of the images, and from it learn the model. In this section, we discuss this learning procedure, which is shown schematically in Figure 2.

Hierarchical Contour Model Merging. Let $C_1 \ldots C_N$ denote the N contours obtained for our training images. We generate a single unified model from these contours by using a a hierarchical merging technique reminiscent of model merging techniques in other domains [23, 5]. At a high level, this process successively selects pairs of models and merges them into one. Each model in this construction is thereby a result of some set of registered contours.

We initialize our iterative procedure by creating models $\mathcal{M}^{(1)}_0 \ldots \mathcal{M}^{(N)}_0$, where each $\mathcal{M}^{(i)}_0$ is constructed from the single original training instance C_i (Figure 2, left). In each successive iteration, we begin with K models from the previous step, and perform pairwise merging on consecutive model pairs to produce $K-1$ models. Thus, we combine each pair of models $\mathcal{M}^{(i)}_k$ and $\mathcal{M}^{(i+1)}_k$ into a new model $\mathcal{M}^{(i)}_{k+1}$ as illustrated in the pyramid. In order to

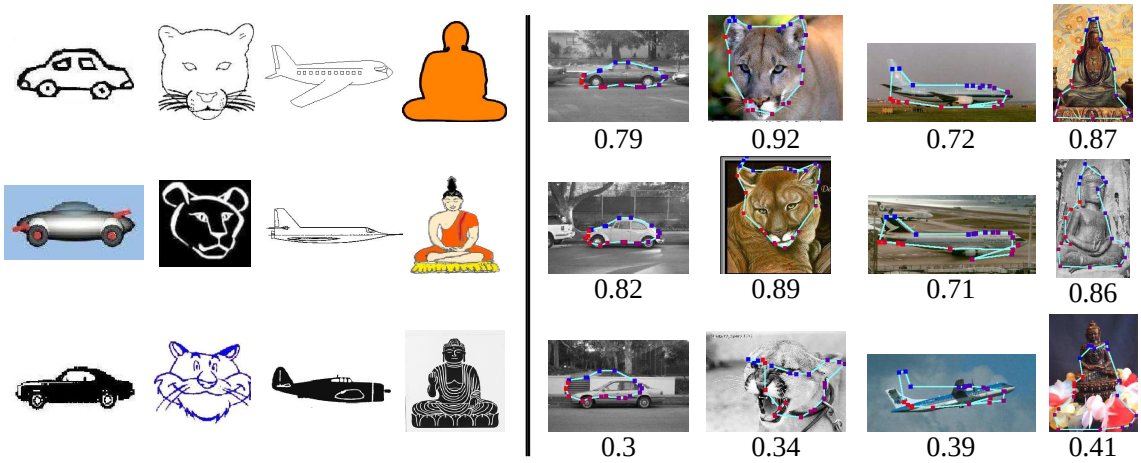

Figure 3: Cartoon training images (left) and sample outlining results on test images (right) along with the overlap scores of these registrations for several object classes. Shown are two good and one bad example for each class.

perform the merge, we first determine the correspondence between landmarks in the two models, using our MRF localization scheme described in Section 2: We use $\mathcal{M}_k^{(i)}$ as the model, and generate an "ideal image", $I_k^{(i+1)}$, from the model $\mathcal{M}_k^{(i+1)}$. In this image, the landmarks lie at their mean locations according to the generating model, and each point along the contour connecting these landmarks serves as an edge pixel in the image. The domain for each landmark l in $\mathcal{M}_k^{(i)}$ is the set of landmark locations in $I_k^{(i+1)}$. We then register $\mathcal{M}_k^{(i)}$ and $I_k^{(i+1)}$ using our MRF algorithm; the resulting correspondence determines a correspondence between landmarks of $\mathcal{M}_k^{(i)}$ and $\mathcal{M}_k^{(i+1)}$. We now construct the merged model $\mathcal{M}_{k+1}^{(i)}$, associated with all of the training contours covered by $\mathcal{M}_k^{(i)}$ and $\mathcal{M}_k^{(i+1)}$. The merged model has the same set of landmarks as $\mathcal{M}_k^{(i)}$, each of which is registered to a landmark in $\mathcal{M}_k^{(i+1)}$. Using the registration between the landmarks of $\mathcal{M}_k^{(i)}, \mathcal{M}_k^{(i+1)}$ and their respective training contours, we obtain a registration between the landmarks of $\mathcal{M}_{k+1}^{(i)}$ and all of its associated contours. This process is continued, as shown in Figure 2, until a single model, learned from all N training instances, remains.

Landmark Pruning. The output model from this procedure may have a fairly large number of landmarks, depending on the number of points in the initial contours. The more landmarks we have, the larger the number of parameters, and the higher the complexity of inference in our MRF. We therefore trade off model complexity and fit to data, to reduce both the risk of overfitting and computational cost.

We prune landmarks using a simple but effective heuristic, inspired by the minimum description length (MDL) framework [20]. We define a probabilistic model over entire contours using a simple Gaussian distribution around the contour of the ideal image induced by a model $\mathcal{M}$. The "fit to data" component of the MDL score is then measured as the log-likelihood of the training contours relative to this likelihood function, which is simply the squared error between the training contours and the mean contour of $\mathcal{M}$. This likelihood is penalized by subtracting a term for model complexity, measured as a constant c times the number of landmarks $|\mathcal{L}|$. We then use a simple greedy algorithm to remove landmarks, so as to maximize this penalized score. This process automatically determines which landmarks to retain for the final model $\mathcal{M}$.

Model Construction. Given the set of landmarks $\mathcal{L}$, and the set of training contours $C_1, \ldots, C_N$ registered to $\mathcal{L}$, we essentially have a fully observed data set. We can therefore construct a probabilistic model $\mathcal{M}_{class}$ using standard Bayesian estimation techniques.

First, to capture the pairwise interactions between landmarks l and m, we simply compute the necessary moments of the distances between the assignments of these landmarks, p_l and p_m, across the training instances, regularized by a large variance Normal-Wishart prior [10] with an imaginary sample size of 1 and a standard deviation equal to a quarter of the average object size.

We estimate the Gaussian posterior distribution for the landmark location feature $f_0^l(I, p_l)$ from the actual locations of the landmark in the contours, regularized using a large variance Normal-Wishart prior [10] with an imaginary sample size of 1 and a standard deviation equal to half the size of the object.

Estimating the parameters of the shape-template model is slightly more involved. Briefly, for each landmark l, a shape template is first constructed independently for each

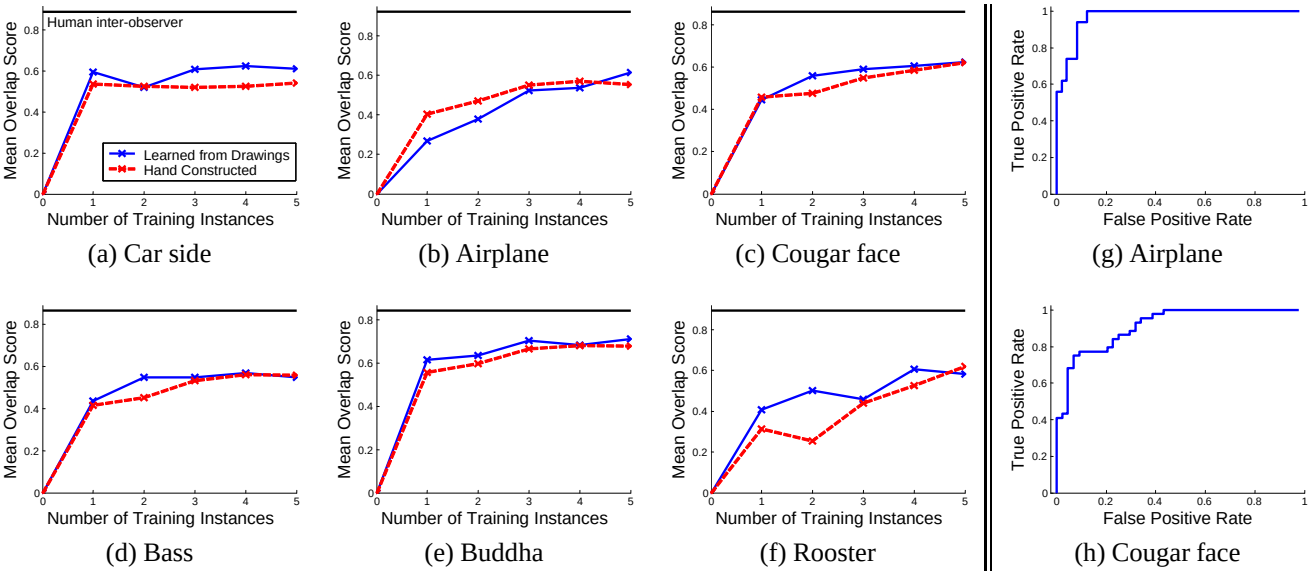

Figure 4: (a)–(f) Average overlap score as a function of the number of training instances for several object classes. Compared are the results for models learned automatically from cartoon drawings (blue,solid) and models learned from hand segmented images (red,dashed). (g),(h) Recognition ROC curves for the 'airplane' and 'cougar face' objects showing sensitivity vs. specificity as the threshold on the likelihood defined in Eq. (1) is varied.

instance by following the outline defined by that instance. The mean offsets of the individual shape templates are then averaged and the variance is estimated with a regularization of a Normal-Wishart prior [10] with a variance of 25 pixels. The final shape template is pruned so that, on average across instances, its length is roughly three quarters of the size (diagonal of bounding box) of the contour model. Estimating the appearance feature is done simply by computing the appearance histograms for each training instance and then averaging across instances. The mask is computed from the above shape template.

5 Experimental Results

We applied our procedure to six diverse object categories from the Caltech 101 dataset [11]. To generate a training set of cartoon drawings, we used simple drawings from 'Google images' that were then scaled to approximately the same dimensions as the Caltech dataset instances. Figure 3(left) shows three of the cartoon drawings used for training in four of the object categories.

We then create a model $\mathcal{M}_{class}$ for each object class as described above in Section 4, and randomly selected a test set of up to 50 images for each class and registered our model to the image using the MRF based procedure desribed in Section 2. Figure 3(right) shows sample outlines for several object classes.

To quantitatively evaluate the ability of our method to localize objects in complex images, we use an overlap metric that measures the extent to which the object is correctly localized relative to a hand-segmented 'ground truth'. Let $\mathcal{X}_L \subset \mathcal{X}_I$ represent the set of pixels in image I contained within the boundary of the contour instance L, and $\mathcal{X}_{GT} \subset \mathcal{X}_I$ represent the set of pixels lying within the hand-labeled ground truth. We define the overlap accuracy between these two sets to be the ratio of the number of pixels in the intersection to the number of pixels in the union: $\text{Overlap}(\mathcal{X}_L, \mathcal{X}_{GT}) = \frac{|\mathcal{X}_L \cap \mathcal{X}_{GT}|}{|\mathcal{X}_L \cup \mathcal{X}_{GT}|} \in [0, 1]$. Intuitively, two good "outlines" of an object will share a vast majority of their interior pixels, and have a high overlap score. To get a sense for this score, Figure 3 shows the overlap score for each example outline.

In Figure 4(a)-(f), we compare the overlap performance of our object models learned from cartoon drawings with that of models constructed from hand-labeled training instances. Shown is the average test-image overlap as a function of the number of training instances. As can be clearly seen, the performance of the model learned using our automatic method is similar to that of the model learned from hand-labeled instances, which contains the user's prior knowledge. These graphs show that our method is quite effective at outlining a variety of object classes.

Although recognition is not our primary goal, and our method was not tuned for recognition performance, it is nevertheless instructive to study how our method performs on this task. We registered each object model to 50 randomly selected 'background' images, which served as negative test instances for each class. We used the likelihood of

the correspondence defined in Eq. (1) to evaluate the extent to which our registration is successful on both positive and negative test instances. Figure 4(g),(h) show ROC curves for the 'airplane' and 'cougar face' models. The break-even recognition rates are 92%, 90%, 84%, 82%, 82%, and 69% for airplane, buddha, car, rooster, cougar, and bass, respectively. These recognition rates are surprisingly reasonable when we consider the fact that the model was trained only only five cartoon instances. For example, in the results of FeiFei *et al.* [11], who also trained a generative model from few instances, using 6 training examples, the recognition rate for 'cougar face' was 85%, and comparable performance was achieved only for 11 of the 101 classes. Finally, we note that the recognition rates of the models learned from hand segmented instances was consistently worse. However, we do not believe this is indicative of a significant difference between the cartoon and hand models, as one cannot predict what would happen if we tune our model parameters to improve recognition performance.

Finally, it is insightful to consider specific examples of registration of our model to images. Figure 5(a) shows a registration example where our model fails. Figure 5(b) shows the Canny edge map for that image and demonstrates some of the challenges of the outlining task: in this image, many of the car edge pixels are simply not detected due to the similar intensity between the car and the wall behind it. Clearly, outlining in this case requires a better edge map or alternative low level cues.

Figure 5(c)–(d) shows registrations to the same 'car side' image using different components of our model. In (c) we use only the shape template feature, the location prior, and pairwise distances and are not able to successfully outline the car. In (d) we add our foreground-background appearance feature. This feature, which favors differences in intensity and texture between the inside and outside of the object, is clearly not compatible with the correspondence found in (c) and pushes our outline toward the car, achieving reasonable success. It is also important to note that (c) exemplifies the fact that our location prior is relatively weak: it typically simply helps to prune solutions that are close to the edge of the images while still allowing large deviations from the expected location of the landmarks.

6 Discussion and Related Work

In this work, we introduce a novel method for learning the shape of object classes using a landmark based model. We show how our model can be automatically learned from cartoon drawings and registered to complex images toward the task of precise outlining of the object. We demonstrate the effectiveness of our method for several varied model classes and show that our method achieves similar results to those of a model learned from hand segmented images.

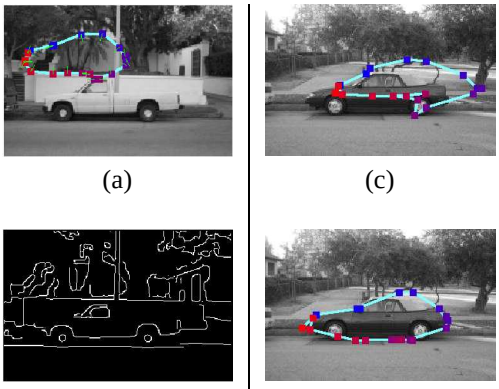

Figure 5: (a) An example where our 'car' model fails; (b) edge map of (a); Registration using (c) only shape template; (d) adding the appearance feature.

Our contribution in this work is twofold. First, we present a model aimed at capturing the "fundamental" shape of the object class. Our model is based on landmarks that define the outline of the objects along with local landmark characteristics and more global geometric constrains. Importantly, all the features we use, including those based on appearance, are "shape aware" and take into account the outline of the object. Second, we introduce a learning approach that effectively bootstraps information in simple cartoon drawings and applies that information to complex images. This allows us to circumvent the need for time-consuming and costly human supervision.

The problem of object recognition has been addressed in many works with different goals. One class is aimed solely at recognition and often uses a discriminative approach (e.g., [14, 22]). As such, these methods are typically ignorant of the shape and often even the location of the object in the image. Importantly, the image features used for discrimination are often not in the object itself, but in the background. Clearly, using the context of an object for recognition is both helpful and legitimate. However, there are many tasks (e.g., distinguishing objects on a kitchen countertop, or recognizing different animals in a grassy field) where the background is similar for the different classes, and so provides no discrimating power. It is therefore useful to investigate methods that focus more directly on the object itself. Moreover, in order to achieve state-of-the-art recognition rates, discriminative methods usually require a significant number of labeled training instances. In fact, Ng and Jordan [17] suggest that this is an inherent limitation of discriminative approaches and that generative methods are more appropriate when the number of instances is small.

Another line of work tries to model geometric elements or parts of the object without precisely outlining it in the

image. Recent years have seen several works that address this task using a part based model that is either purely generative such as the constellation model [13, 11], or semi-discriminative such as the work of Quattoni *et al.* [19], which uses conditional random fields to model the object. Many of the above works are able to achieve impressive recognition performance. However, their ability to localize the object precisely has not been thoroughly evaluated.

On the surface, the most similar to our work in this class is that of Berg *et al.* [2], which uses a landmark based model that is registered to images. Two important differences between their approach and ours are worth noting. First, they use a larger set of images for training (3 times as many) and store all of them as exemplars instead of creating a unified probabilistic model. Thus, they never attempt to explicitly model the shape of an object but rather to match templates to images. Second and more importantly, they use a larger number of landmarks with features that cover a large portion of the image. As a result, their work leverages general image statistics more than object-specific statistics, allowing them to exploit information about the background, similarly to the discriminative methods discussed above.

Finally, several works also considered the problem of precise outlining of objects. Typical examples include the active shape models of Cootes *et al* [6], the stick figures model of Coughlan and Ferreira [7], or the object-based segmentation of Kumar *et al.* [15]. These methods are generally applied to relatively limited scenarios or use simple models with few degrees of freedom, and it is not clear whether and how they can be applied to general object classes. Our work addresses the same challenging task of precisely outlining objects in images, but defines a general and flexible "shape aware" model that can be registered to varied and complex object classes.

Our work can be extended in several directions. Most obviously, our model currently assumes that the object is in a fixed pose and scale. We hope to avoid these assumptions by introducing a search over object pose and scale. We also hope to incorporate a more effective appearance model by learning first on cartoon images, followed by an appearance learning stage using real images. More broadly, inspired by the exemplar-based approach of Berg *et al.* [2], we would like to consider a mixture-model for the shape template at each landmark, allowing greater flexiblity in shape modeling. Even more generally, we note that our notion of object shape can also be viewed as a natural 2D projection of a 3D mesh model. We believe that object-class variability can be very naturally modeled in 3D (e.g., as in Anguelov *et al.* [1]), and hope to extend our approch to allow registration of 2D images to flexible 3D shape models.

Acknowledgements. This work was supported by DARPA under the Transfer Learning program. We thank Dragomir Anguelov for useful discussions.

References

[1] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: Shape completion and animation of people. SIGGRAPH, 2005.

[2] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondence. CVPR, 2005.

[3] T. Binford. Visual perception by computer. CSC, 1971.

[4] J. Canny. A computational approach to edge detection. PAMI, 1986.

[5] P. Cheeseman, J. Kelly, M. Self, J. Stutz, W. Taylor, and D. Freeman. Autoclass: a Bayesian classifi cation system. ML, 1988.

[6] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models: their training and application. CVIU, 1995.

[7] J. Coughlan and S. Ferreira. Finding deformable shapes using loopy belief propagation. ECCV, 2002.

[8] T. Cover and J. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, 1991.

[9] J. Daugman. Uncertainty relations for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical fi lters. JOSA, 1985.

[10] M. DeGroot. *Probability and Statistics*. Addison Wesley, Reading, MA, 1989.

[11] L. Fei-Fei, R. Fergus, and P. Perona. A Bayesian approach to unsupervised one-shot learning of object categories. ICCV, 2003.

[12] P. Felzenszwalb and D. Huttenlocher. Effi cient matching of pictorial structures. CVPR, 2000.

[13] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. CVPR, 2003.

[14] K. Grauman and T. Darrell. Pyramid match kernels: Discriminative classifi cation with sets of image features. ICCV, 2005.

[15] M. Kumar, P. Torr and A. Zisserman. OBJ CUT. CVPR, 2005.

[16] J. Malik, S. Belongie, T. dLeung, and J. Shi. Contour and texture analysis for image segmentation. IJCV, 2001.

[17] A. Ng and M. Jordan. On discriminative vs. generative classifi ers: A comparison of logistic regression and naive Bayes. NIPS, 2002.

[18] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, 1988.

[19] A. Quattoni, M. Collins, and T. Darrell. Conditional random fi elds for object recognition. NIPS, 2004.

[20] J. Rissanen. Modeling by shortest data description. Automatica, 1978.

[21] Y. Rubner, C. Tomasi, and L. Guibas. A metric for distributions with applications to image databases. ICCV, 1998.

[22] T. Serre, L. Wolf, and T. Poggio. Object recognition with features inspired by visual cortex. CVPR, 2005.

[23] A. Stolcke and S. Omohundro. Best-fi rst model merging for hidden Markov model induction. Technical Report TR-94-003, ICSI, 1994.

[24] C. Xu and J. L. Prince. Gradient vector fbw: A new external force for snakes. CVPR, 1997.