

A Unified Contour-Pixel Model for Figure-Ground Segmentation

Ben Packer, Stephen Gould, and Daphne Koller

Computer Science Department, Stanford University, CA, USA

Abstract. The goal of this paper is to provide an accurate pixel-level segmentation of a deformable foreground object in an image. We combine state-of-the-art local image segmentation techniques with a global object-specific contour model to form a coherent energy function over the outline of the object and the pixels inside it. The energy function includes terms from a variant of the TextonBoost method, which labels each pixel as either foreground or background. It also includes terms over landmark points from a LOOPS model [1], which combines global object shape with landmark-specific detectors. We allow the pixel-level segmentation and object outline to inform each other through energy potentials so that they form a coherent object segmentation with globally consistent shape and appearance. We introduce an inference method to optimize this energy that proposes moves within the complex energy space based on multiple initial oversegmentations of the entire image. We show that this method achieves state-of-the-art results in precisely segmenting articulated objects in cluttered natural scenes.

1 Introduction

The task of figure-ground segmentation is well established in the computer vision literature. There have generally been two types of approaches to this problem: outline-based methods (e.g., [2–5]) that denote the foreground by the interior of an object outline; and pixel-level foreground annotation (e.g., [6–8]) that label each pixel directly as either foreground or background. In this paper we combine these two approaches to achieve a superior and more refined object segmentation. Our method provides both an object contour, which exploits object-level information (such as shape), and a pixel annotation, which exploits pixel-level feature information (such as color and texture). We leverage this complementary relationship to improve the performance of each of these elements over using them in isolation.

We do so through two main contributions: The first, presented in Section 5, is the combination of the elements from two standard models for localization (contour) and segmentation into a unified energy model that can be precisely registered to a foreground object in a scene. Our model combines existing energy terms for each separate task ([1, 4]) with an interaction term that encourages the contour and pixel-level segmentation to agree. Specifically, we introduce landmark-segment masks that capture the local shape of the foreground object in

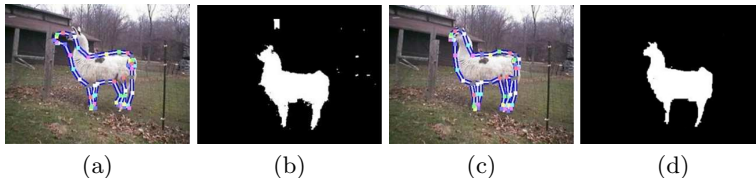


Fig. 1. Contour and Segmentation. (a) Independent LOOPS outline. (b) Independent TextonBoost segmentation. (c) Joint model outline. (d) Joint model segmentation.

the vicinity of a single landmark or pair of landmarks along the object’s outline. Importantly, the masks are oriented and scaled to be consistent with the full object contour. This allows for a refined segmentation based on the articulated contour, which is not possible using a single global mask for the entire object. We also use the contour to construct an image-specific appearance model, which has been used successfully in other settings, further tying the two models. Example output for standard independent contour and segmentation models are shown in Figure 1(a) and (b), respectively. While each task produces reasonable initial results, our unified model leads to much improved figure-ground segmentation results, as shown in Figure 1(c) and (d).

Our second main contribution, presented in Section 6, is a method for optimizing the complex joint energy by proposing sets of moves within the entire search space, which is intractable to navigate in full. We build on the techniques of Gould et al. [9] by iteratively using the novel properties of our model to restrict the search space and efficiently finding a good solution within that subspace. Furthermore, this procedure lends itself to model-aware dynamic updates of the image-specific appearance model, which provides strong boosts in performance. In Section 7, we present experimental results to validate our approach, and show that we achieve both localized outlines and pixel-level segmentations that outperform state-of-the-art methods.

2 Related Work

Among successful object-specific, contour-based methods for object outlining are Ferrari et al. [5] (kAS) and Heitz et al. [1] (LOOPS). Our experimental results outperform both of these methods, and indeed we build on the latter to produce more accurate outlines. Among pixel annotation methods, the OBJ CUT method of Kumar et al. [7] and the method of Levin and Weiss [8] are two examples that, like our method, exploit both high-level shape cues and low-level features. They use these cues, however, in a strictly feed-forward manner to produce a segmentation. Our method propagates information both ways between the shape and pixel models, which results in a superior result for each one. Leibe et al. [6] do include a backprojection step that refines initial hypotheses. They do not, however, utilize a global model of object shape, nor do they produce a

single coherent result — their soft output allows cows to have more or less than four legs, for example.

Image-specific appearance models for object recognition have been used by Winn and Jovic [10], Kumar et al. [7], and Ramanan [11], among others. Our implementation learns this appearance with the help of a LOOPS model. This not only provides a particularly strong cue for using the correct pixels, but also allows us to use the properties of LOOPS to select those pixels carefully. As we describe in Section 4, we use the contour model to rate our uncertainty over different locations in the image, which allows us to learn the appearance only over pixels about which we are confident.

Our work is most similar to Bray et al. [12] and Chen et al. [13], which both combine a CRF-based segmentation model with an object model, as we do. The differences between our approach and theirs highlight our contributions. Bray et al. [12] use a single distance function to relate the object skeleton to the background segmentation. This is roughly equivalent to using masks as we do, but in their case these masks are the same for each part of the object and are restricted to the form of a distance function that does not capture outline detail. Chen et al. [13] use a single mask for the entire object, which is problematic for articulated objects since it cannot account for multiple configurations. Indeed, they report results on classes from Caltech 101 [14] that have rigid shapes and for which segmentation is easier than in cluttered scenes. In contrast, our landmark-specific masks are different for each part of the object, have a general form that can capture outline detail, and are learned from data to capture this detail. This allows us to learn and preserve particular shapes in the segmentation over different object parts such as the outline (and ears) of the head, even in the presence of articulated skeletons such as those found in the Mammals dataset [15], for which we report results. Furthermore, both [12] and [13] alternate between optimizing over the object and segmentation using coordinate ascent. We present an efficient method for joint inference, which can avoid local minima found in each task separately.

3 Localization and Segmentation Models

Our aim is to build a model that encompasses both the localization and the segmentation task, and that incorporates the interactions between the two in order to improve performance on each task. This model is specified by an energy function Ψ that is an aggregation of individual energy terms over various components of the model. In this section, we describe two approaches from the vision literature for solving the two separate tasks, each of which yields individual energy terms. We describe how these tasks can be solved separately as baseline methods, and in later sections we use these energy terms in our joint model. In Section 4, we introduce an interaction from the localization component to the segmentation component through image-specific features. In Section 5, we introduce landmark-segmentation masks that tie the two main model components together in a bidirectional manner.

3.1 Outline Localization

The recent LOOPS model of Heitz et al. [1] treats object localization as a landmark correspondence problem, the solution to which defines a piecewise-linear contour around the object in an image. We describe this model throughout this section. Formally, the task is to assign each landmark L_i to the appropriate pixel on the object’s outline. We denote the full assignment to all landmarks by \mathbf{L} .

Registering the landmarks to a test image requires optimizing an energy function $\Psi^L(\mathbf{L})$ over the landmark assignments. This energy function is composed of two types of terms over the landmark assignments. The first is a singleton feature-based term that predicts the location of a specific landmark from a set of image features. We let $\psi_i^L = \langle \theta_i^L, \phi_i^L \rangle$, where ϕ_i^L is the response vector of a boosted detector [16] for landmark i , and $\langle \cdot, \cdot \rangle$ denotes the dot-product between the model parameters θ_i^L and the landmark features ϕ_i^L .

The second term in Ψ^L is a global shape term that gives preference to the landmarks forming a likely object shape. This term is a multivariate Gaussian over all landmarks, which decomposes into pairwise terms:

$$\delta_{i,j}^L = -\frac{1}{2}(L_i - \mu_i)\Sigma_{ij}^{-1}(L_j - \mu_j), \quad (1)$$

where μ_i is the mean location of landmark i and Σ is the covariance matrix that relates the positions of all landmarks.

Figure 1(a) shows an example result of finding the optimal assignment over the landmark variables of the entire landmark energy:

$$\Psi^L(\mathbf{L}) = w_1 \sum_i \psi_i^L + w_2 \sum_{i,j} \delta_{i,j}^L, \quad (2)$$

where the weights w_1 and w_2 determine the relative influence of each term. The parameters and weights can all be learned from supervised data, and the energy can be optimized approximately in isolation using max-product message passing algorithms (see Section 6).

3.2 Foreground Segmentation

We now turn to a standard technique for foreground-background segmentation. This task amounts to assigning a variable S_k for each image pixel k to be either foreground ($S_k = 1$) or background ($S_k = 0$). The full assignment to all pixels is denoted by \mathbf{S} . We use a variant of the TextonBoost algorithm [17] to perform this task. Since the datasets we consider in Section 7 generally consist of a single foreground object on a background that is comprised of several common categories (such as grass, sky, and trees), we train a separate binary boosted classifier for each of these classes. The outputs of these classifiers are used as features for a logistic classifier that predicts whether each pixel is foreground. We use a pairwise binary conditional Markov random field (CRF) over the pixels in the image, where the singleton potentials are represented by the logistic classifier and the pairwise potentials encourage neighboring pixels with a similar appearance to have the same label.

The CRF for foreground segmentation represents an energy $\Psi^S(\mathbf{S})$ over the pixel assignments that consists of a singleton term ψ_k^S and a pairwise term $\delta_{k,l}^S$. Given the outputs of the various boosted classifiers for each pixel k in feature vectors ϕ_k^S , the first term takes the form $\psi_k^S = \langle \theta_{s_k}^S, \phi_k^S \rangle$, where $\theta_{s_k}^S$ is the set of logistic regression weights (shared between all pixels) associated with the assignment $S_k = s_k$. The pairwise term takes the form

$$\delta_{k,l}^S = \begin{cases} \exp\left(-\frac{\|c_k - c_l\|_2^2}{2\bar{c}}\right), & \text{for } (k, l) \in \mathcal{N}(\mathcal{I}) \text{ and } S_k \neq S_l \\ 0 & \text{otherwise} \end{cases}, \quad (3)$$

where $\mathcal{N}(\mathcal{I})$ is the set of neighboring pixels in image \mathcal{I} (in our implementation we use 4-connected neighbors), c_k is the vector of *Lab* color values at pixel k , $\|\cdot\|_2^2$ is the L_2 distance between such vectors, and \bar{c} is the mean such distance across all neighboring pixels in the image. Note that the pairwise term is only non-zero when neighboring labels that are not equal (i.e., at the boundary between foreground and background), and thus penalizes neighboring pixels when their labels are different and the contrast between them is low. The full segmentation energy over \mathbf{S} is given by

$$\Psi^S(\mathbf{S}) = w_3 \sum_k \psi_k^S + w_4 \sum_{k,l} \delta_{k,l}^S, \quad (4)$$

where w_3 and w_4 weight the two terms. As with the landmark model, the classifiers and weights are learned from the labeled training set. The energy can be optimized exactly in isolation using a graph cut [18] (see Section 6). Figure 1(b) shows an example result for the image in Figure 1(a).

4 Image-Specific Appearance

Building an *image-specific* appearance model helps combat the fact that the variation across images in the appearance of both the object class and background make it difficult or impossible to reliably separate the two. While the segmentation CRF models the fact that an object should have consistent appearance (at least in neighboring pixels) through its pairwise terms, the singleton terms nevertheless adhere to a single appearance model across the entire object class. We therefore use the initial localized outline of the LOOPS model to construct an *image-specific* appearance model to augment the class-level appearance model within the segmentation CRF at test time.

Specifically, we build a naive Bayes classifier based on pixel color values that will distinguish between the object in the image and the background particular to the image. To estimate the parameters of the classifier, we split the image pixels, each of which carries a class label of either foreground or background based on the contour estimate, into three mutually-exclusive sets: **E** (excluded), **C** (certain pixels), and **U** (uncertain pixels). Background pixels that are far away from the border of the localized contour are neither useful for training nor important to consider relabeling, and hence belong in **E**. Certain pixels, **C**,

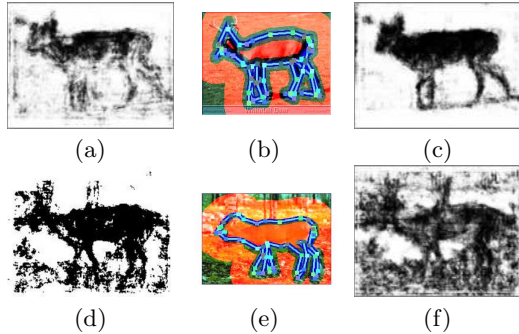


Fig. 2. Local appearance features. (a,d) Response maps of class-level boosted classifiers for deer. (b,e) Initial LOOPS outlines. Highlighted pixels are those chosen as “confident” training examples for the local appearance. (c,f) Response maps of the resulting appearance model.

are non-excluded pixels (either foreground or background) for which the contour model is sufficiently confident about their label (see below). The remaining pixels belong in \mathbf{U} . We train the naive Bayes model over only the pixels in \mathbf{C} and \mathbf{U} as follows: (1) we seed the class labels for the pixels in \mathbf{C} based on whether the pixel is inside or outside the contour, (2) leave the class labels for \mathbf{U} hidden, and (3) use the EM algorithm [19] both to learn an appearance model for the foreground and background, and to reinfer the class labels. The log of the posterior probability of each pixel being the foreground is then used as a feature — alongside the boosted classifier outputs (see Section 3) — for the logistic classifier component of the segmentation CRF, which is retrained.

To determine which pixels belong in \mathbf{C} , we note that we may be more confident about certain parts of the object than others; for example, the localization method may be certain that it has localized the torso of the deer, but less certain about the particular placement of the legs. We determine the reliability of each landmark separately by measuring how likely the localization method is to have properly assigned that landmark. Let σ_i be the standard deviation of the distance of the localized landmark L_i to the true outline on the training data. We compute a signed distance $Dist(k)$ (also used in Section 5) of each pixel in the test image to the localized outline, where the sign is positive if the pixel is inside the contour and negative otherwise. Pixel k belongs to \mathbf{C} if $|Dist(k)| > \sigma_i$ for the closest landmark i . Note that computing this score, as well as retraining the CRF’s logistic classifier, requires running the localization method on the training data. Figure 2 shows the responses of this naive Bayes classifier on a test image. In the top row, despite the imperfect LOOPS outline, the learned appearance model is still strong. However, as shown in the bottom row, even with a good LOOPS outline, the local appearance is not always a perfect feature. In Section 7, we analyze the results of augmenting the segmentation task in this way, which we refer to as **ImgSpec**.

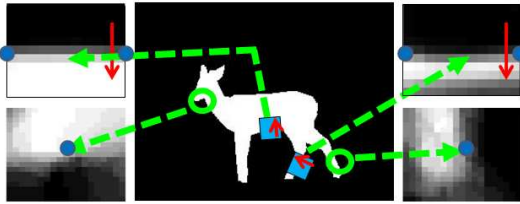


Fig. 3. Landmark-segment masks. The green arrows indicate the mask associated with various landmarks, which are marked as blue dots. The upper two masks are pairwise masks between neighboring landmarks, and are reoriented and rescaled appropriately — the red arrow indicates the “inside” direction of the mask.

5 The Contour-Pixel Model

We now present a unification of the contour and pixel models in which we incorporate more information than pixel appearance. Importantly, this information flows both ways. There is a natural agreement between localization and segmentation in that the pixels inside the contour outlined by the landmarks \mathbf{L} should be labeled as foreground, and those outside should be labeled as background.

A naive way to combine the two signals is simply to merge the segmentation CRF’s probability over each S_k with that pixel’s signed distance to the localized contour. Let $P_0(S_k)$ be the posterior probability over S_k according to the CRF. We define our new probability $P_1(S_k)$ to be the product of $P_0(S_k)$ and the sigmoid of the signed distance $Dist(k)$ (defined in Section 4), normalized to sum to one.

As we show in Section 7, combining the models in this way (which we call **Product**) does not lead to an improvement in performance. To fully exploit these parallel signals, rather than post-processing their outputs, we would prefer to allow each method to reoptimize its own variables in light of information propagated from the other. We now describe a model that unifies the two tasks in a single coherent model.

We introduce a new energy term $\Psi^{L,S}(\mathbf{L}, \mathbf{S})$ that encourages agreement between landmarks \mathbf{L} and segmentation \mathbf{S} . Since a LOOPS landmark is a consistently located element of the object’s shape, the nearby pixel annotations should follow a pattern particular to that part of the object. For example, the pixels above the landmark corresponding to the stomach will generally belong to the foreground, while those below it will generally be part of the background. For each landmark L_i , we build an “annotation mask” M_i of size $N_1 \times N_1$ that is a grid whose (a, b) -th entry indicates the probability that a pixel offset by (a, b) from the location of L_i is a foreground pixel. Each mask is learned from training images by aggregating masks of size N_1 around the groundtruth landmark location in each training image, and the learned mask is simply the average of each of these masks. Examples of landmark masks near the nose and leg of a deer are

shown in Figure 3. The energy term associated with this pairwise mask is

$$\psi_{i,k}^{LS_1} = S_k \log M_i(a, b) + (1 - S_k) \log(1 - M_i(a, b)). \quad (5)$$

If the offset (a, b) between landmark i and pixel k extends beyond the size of the mask ($\frac{N_1}{2}$), then there is no pairwise energy term that relates L_i to S_k . This potential allows information to propagate between the contour model and the segmentation model. A landmark L_i with a high probability of appearing at a given location will encourage the surrounding pixels to be annotated according to the mask. This information can then propagate to the rest of the image pixels via Ψ^S . Conversely, a pattern fitting the mask appearing in the pixel labels encourages the landmark to assign itself in the appropriate nearby location, and this can influence the rest of the landmarks via Ψ^L .

In addition to masks that capture the relationship between single landmarks and their surrounding pixels, we introduce masks $M'_{i,j}$ that tie neighboring pairs of landmarks L_i and L_j jointly to their surrounding pixels. These masks are similar to M , but account for different orientations of consecutive landmarks. Each adjacent pair of landmarks L_i, L_j is associated with an oriented and scaled mask $M'_{i,j}$ whose (a, b) -th entry is the foreground probability of the pixel offset by (c, d) from the midpoint between L_i and L_j , where (c, d) is found by rotating the vector (a, b) by the angle of the segment $L_i - L_j$ and dividing by the length of that segment. We learn these pairwise masks from training data similar to the singleton masks above. The energy term associated with this mask is

$$\psi_{i,j,k}^{LS_2} = S_k \log M'_{i,j}(a, b) + (1 - S_k) \log(1 - M'_{i,j}(a, b)). \quad (6)$$

Figure 3 shows an example of a such a mask for consecutive landmarks along one of the deer’s hind legs. It clearly indicates that, regardless of the orientation of the leg, pixels that are on the “inside” of the line segment on the neck are more likely to be foreground.

Now that we have created the energy terms that tie together the variables of our model, we define the energy of a full variable assignment (\mathbf{S}, \mathbf{L}) given the image as

$$\Psi = \sum_t w_t \cdot \Psi^t(\mathbf{S}, \mathbf{L}), \quad (7)$$

where t ranges over the types of energy terms. While weight ratios learned for each model are kept fixed, the relative weights for all terms are learned using cross-validation on the training set. Note that Ψ is composed of at most triple-wise terms between the variables \mathbf{S} and \mathbf{L} . Having defined this CRF over \mathbf{S} , \mathbf{L} , and input image \mathcal{I} , we seek the single joint assignment to \mathbf{S} and \mathbf{L} that minimizes the energy. That is, the MAP solution is $(\mathbf{S}^*, \mathbf{L}^*) = \operatorname{argmin}_{\mathbf{S}, \mathbf{L}} \sum_t w_t \cdot \Psi^t(\mathbf{S}, \mathbf{L})$.

6 Superpixel-based Inference

6.1 Inference challenges

We now consider the properties of our coherent energy function in deciding how to optimize it. The pixel annotation terms (Ψ^S) can be optimized exactly using

a graph cut [18] if considered independently, since there are regular pairwise terms between binary-valued variables. However, the landmark location terms (Ψ^L) cannot be optimized exactly even if considered independently, and in fact performing inference with these terms proves to be a challenge. To complicate matters, we have pairwise terms between pixels and landmarks (which can take many values) and triplewise terms between pixels and pairs of landmarks. A model with 50 landmarks in a 300×200 pixel image, for example, would have 3 million pairwise terms and 150 million triplewise terms. Thus, there is a great deal of interconnectivity between the variables, and even constructing a graph to represent the full joint energy may be intractable.

Coordinate Descent Baseline

One straightforward approach to inference would be to simply perform coordinate descent on the full energy. This can be done by first optimizing Ψ^L over \mathbf{L} , then folding the potentials in $\Psi^{L,S}$ evaluated at the fixed \mathbf{L} into the singleton terms ψ_k^S , then optimizing Ψ^S separately over \mathbf{S} (which, again, may be done exactly and efficiently), then folding $\Psi^{L,S}$ evaluated at the fixed \mathbf{S} into the singleton potentials of Ψ^L , and iterating back and forth in this manner. As we show in Section 7, this approach (which we call **Coord**) succeeds in sharing the signals between the two energies, but is susceptible to local minima and does not allow the exploration of the full variable space.

6.2 Joint Inference

To overcome these inference issues, we develop a search strategy for dealing with MAP inference in the face of such a complex and large search space by exploring dynamically constructed discrete subspaces. We then use a final refined stage, initialized from the result of the discrete stage, that uses the full search space.

Our joint inference algorithm proceeds as follows. We begin with an initial assignment to all of the variables, and then find a naturally defined and much smaller subspace through which we can explore the energy function. This subspace is defined by a set of proposal moves from the current assignment to new assignments to the variables. After performing inference within the simpler subspace, if the new assignment achieves an improved energy (note that since inference is not exact, we cannot guarantee that we have found the optimal assignment within the subspace), we keep the new assignment, and otherwise revert to the previous assignment. We then construct a new subspace and repeat.

Constructing Search Subspaces

We choose a subspace for each iteration in two ways. The first stems from the observation that groups of nearby pixels tend to have the same label, and the relationship between landmarks and nearby pixels tends to be the same for entire groups of pixels. We therefore divide the image into superpixels (using the mean-shift segmentation algorithm [20]) and define our proposal moves over superpixel regions. Specifically, given a starting assignment, the proposal moves assign all pixels within a superpixel to either background, foreground, or their current assignment. This approach is similar to the search strategy proposed by Gould

et al. [9]. The inference problem can thus be recast in terms of region variables \mathbf{R} that can take on one of three values rather than individual pixel variables \mathbf{S} that can take on two values (foreground or background).

To avoid committing to any single oversegmentation of image, we use a different oversegmentation (by varying the parameters of the mean-shift algorithm) in each iteration. Each oversegmentation proposes a different set of moves within the space of pixel assignments. For example, an oversegmentation with a small number of large superpixels might propose assigning every pixel in the torso of the deer to the foreground, while a finer-grained oversegmentation might propose refining the pixel assignments around the edge of the torso.

The second simplification of the search space is a restriction on the values of the landmarks, and corresponds to the simplification presented in Heitz et al. [1]: Rather than consider all pixels as possible assignments for the contour landmarks, we choose a small subset (of size $K = 25$) of likely pixels as candidates in each round. Performing multiple rounds of inference, however, allows for the flexibility of choosing candidates for each round in a dynamic and more sophisticated way. In each round, we choose the landmark candidates to be the most likely pixels according to the singleton feature energy terms, subject to two restrictions that vary by round. First, we require that the candidates lie on a superpixel border. This allows us to use the signal from the segmentation differently in each round, tailoring the choices in tandem with the proposed moves that the pixel labels may take. We also restrict each landmark to fall within two standard deviations of its mean location *given* the location of all other landmarks from the previous round of inference. Since the joint model over all landmarks is a Gaussian, computing the conditional Gaussian is straightforward. This restriction allows us to take advantage of the *global* shape information as well as cues from previous rounds. By restricting the search space in these two ways, for a 50-landmark model in a 300×200 image that is split into 300 superpixels, the landmark search space is reduced from $50^{60,000}$ to 50^K and the segmentation problem is reduced from a binary one over 60,000 variables to a ternary problem over 300.

Inference Over Multiple Subspaces

Note that, although we construct a different inference model in each round, the algorithm always optimizes a single, consistent energy function. What differs in each round is the way in which the energy terms are combined and the set of moves that may be taken.

Once we have constructed the simplified inference model over the search subspace, we use residual belief propagation (RBP) [21] to perform MAP inference. The ability to do so efficiently depends on the important property of Ψ that it is composed of at most triple-wise terms between the component variables (the regions \mathbf{R} and the landmarks \mathbf{L}). Specifically, the decomposition of Ψ^L presented in Section 3 uses only singleton and pairwise terms between the landmarks \mathbf{L} , and similarly the decomposition of Ψ^S uses only singleton and pairwise terms between the pixel labels \mathbf{S} , which translates into the same property over the smaller set of region labels \mathbf{R} . Finally, the landmark-pixel masks M result in

pairwise terms between a single landmark L_i and a single region R_k , and the oriented masks M' result in triplewise terms between a pair of neighboring landmarks and a single region. Consequently, RBP is able to converge quickly to a joint solution over all variables \mathbf{L} and \mathbf{R} . We experimented with other inference algorithms, such as dual decomposition [22], which generally achieved the same energy solutions as RBP.

Final Refined Stage

Once this iterative process has converged, we reintroduce the full landmark domain and perform a final refined inference step as in LOOPS, allowing the contour landmarks to lie anywhere in the image. As a post-processing step, since our model defines a closed contour over the foreground object, we set all pixels outside the contour (with a buffer of size $\delta = 5$ pixels) to be background. Though this post-processing step operates outside of the framework of the unified energy, it is not a deficiency of the energy construction itself. It is necessary to set pixels that are beyond the reach of the landmark masks to be part of the background. In principle, if the mask sizes were large enough, this step would not be necessary. However, the mask sizes must be kept reasonably small to avoid an overly dense connectivity among the variables. As a result, there is no term in the energy to discourage these faraway pixels from being set to the foreground.

7 Experimental Results

To validate our approach, we ran our method on several classes from the Mammals [15] and Caltech [14] datasets. For each class, we average over five random folds of the data with 20 images for training and the remaining (20-50) for testing. We obtained groundtruth segmentation labels using Amazon’s Mechanical Turk to augment existing contour labels for these datasets.

Because our task involves both locating the object landmark points and the annotated foreground-background segmentation, we present several metrics to evaluate the success of our method. The first is the simple pixel accuracy of the segmentation (percent of total pixels accurately labeled as foreground or background compared to the groundtruth segmentation). The second measures the accuracy of the precise contour implied by the annotated segmentation. We take the gradient of both the assigned segmentation and the groundtruth segmentation, dilate each by 5 pixels, and then compute the Jaccard similarity (intersection divided by union) between the two. The third metric is the symmetric outline-to-outline root-mean-squared (RMS) distance between the outline created by the assigned landmarks and the groundtruth outline.

The first baseline for comparison with our model is the **Independent** model that separately considers the landmark points and the annotated segmentation. That is, this baseline uses the implementations of TextonBoost and LOOPS in isolation as described in Section 3, utilizing neither the image-specific appearance features nor the landmark-segmentation masks in $\Psi^{L,S}$. We are thus comparing to a standard method for segmentation as well as a state-of-the-art method for landmark localization. For the **ImgSpec** baseline, specified in Section 5, the

	Pixel Accuracy			Jaccard Similarity			RMS Distance	
	Indep	ImgSpec	Joint	Indep	ImgSpec	Joint	Indep	Joint
bison	96.6	96.3	96.4	78.6	79.0	81.2	4.0	3.9
elephant	90.5	92.2	93.3	71.7	70.8	76.1	4.7	4.7
llama	89.7	89.4	93.0	61.8	64.1	73.4	6.4	5.3
rhino	91.0	94.0	95.1	64.5	73.3	75.7	4.7	4.4
deer	88.7	89.5	92.1	56.9	54.8	61.6	8.9	7.0
giraffe	89.9	92.0	92.6	62.0	64.9	65.8	6.4	6.7
airplane	92.3	96.3	96.6	60.8	74.7	74.6	4.2	4.0
bass	92.5	92.5	93.5	58.4	60.1	60.5	10.7	9.5
buddha	84.4	86.0	91.9	42.2	44.7	56.8	10.8	10.6
rooster	91.3	92.1	95.5	57.9	61.1	63.6	10.8	9.4

Table 1. Outlining and Segmentation Results. Best performance is in bold; multiple results are in bold when the differences are statistically insignificant.

contour is used to learn the image-specific appearance, and the probability over \mathbf{S} according to the segmentation model is simply multiplied by a similar probability according to the landmark contour. We refer to our method of optimizing the full energy Ψ jointly over all variables, as well as using the image-specific appearance, as **Joint** in the results that follow. Though we do not show the results here, the **Product** baseline from Section 5 did not outperform **Independent**.

The results for the classes considered are presented in Table 1. Our **Joint** method achieves a marked improvement over the **Independent** methods. It achieves higher pixel accuracy than the baseline segmentation on all classes except “bison,” for which the accuracy is statistically the same. All other differences are statistically significant according to a paired t-test: the least significant difference was the “bass” class with a mean difference of 1.0% and p-value of 0.003. For the outline similarity metric, our method was better on all classes, with the least significant difference being the “bison” class with a mean difference of 2.6% and a p-value of 10^{-6} . For the landmark-based RMS distance, our model is statistically similar to the independent LOOPS on the “elephant” class, worse on “giraffe,” and better on all other classes despite small differences for some of them. The mean difference for the “bison” class is 0.1 pixels, but the paired t-test yields a p-value of 0.028. All other classes had significant differences, with the least significant p-value being 10^{-5} .

Note that simply using the image-specific features (**ImgSpec**) gives a boost in segmentation over the baseline, but does not achieve the same level of results as using our full energy and inference. The full model’s pixel accuracy is superior on 7 out of the 10 classes, with all differences being statistically significant, while there is no statistical difference between the other 3 classes. For the outline similarity metric, the full model is superior on all classes except for “airplane,” for which the instances have relatively uniform appearance so that our outline-aided image-specific features account for all of the improvement in our method.

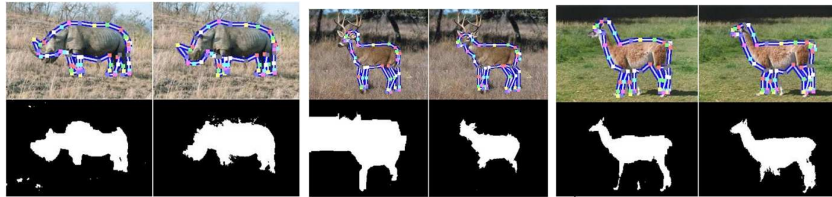


Fig. 4. Three representative model segmentations. Each panel in the left column is produced separately by the **Independent** methods and the right column is produced by our **Joint** method.

We also compared to the **OBJ CUT** method of Kumar et al. [7] and the **kAS Detector** method of Ferrari et al. [5], using downloaded code to run on these datasets. On the pixel accuracy, outline Jaccard similarity, and outline RMS scores, our **Joint** model outperforms the **kAS Detector** by macro-averages over all classes of 3.2%, 1.5%, and 2.3 pixels, respectively. It outperforms **OBJ CUT** by macro-averages of 4.7%, 3.2%, and 4.2, respectively. In addition, we ran our **Joint** model on a single random fold of the Weizmann horses dataset [6] and achieved 95% pixel accuracy (compared to 89% for **Independent**). This is consistent with the performance of Levin and Weiss [8] and likely near the limit of what methods of this type can achieve.

The results of the **Coordinate** approach described in Section 6 isolate the contribution of the joint inference method that we introduced. This approach was worse than **Joint** by macro-averages of 1%, 1%, and 0.2, demonstrating that the inference routine does in fact contribute to the performance.

8 Discussion

This paper presented a new model that fuses methods for object localization and segmentation into a coherent energy model in order to produce more accurate foreground segmentations. The utility of the combined model lies in the use of its outline model in learning the image-specific appearance for the segmentation model, and the terms that encourage agreement between the two while still allowing each the flexibility to reoptimize its own variables. We demonstrated that this model is able to achieve both outlines and segmentations that are superior to several state-of-the-art methods. One promising direction for future work is integration with more sophisticated segmentation algorithms. For example, the use of a robust multi-class segmentation method would allow for class-aware landmark-segment masks that could capture that the giraffe head is often surrounded by sky or trees, while the legs are often found in the grass. Our modular energy function and novel optimization procedure would facilitate such an extension while keeping inference tractable.

Acknowledgements This material is based upon work supported by the National Science Foundation under Grant No. RI-0917151.

References

1. Heitz, G., Elidan, G., Packer, B., Koller, D.: Shape-based object localization for descriptive classification. In: NIPS. (2008)
2. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. In: ECCV. (1998)
3. Thayananthan, A., Stenger, B., Torr, P., Cipolla, R.: Shape context and chamfer matching in cluttered scenes. In: CVPR. (2003)
4. Shotton, J., Blake, A., Cipolla, R.: Contour-based learning for object detection. In: ICCV. (2005)
5. Ferrari, V., Jurie, F., Schmid, C.: Accurate object detection with deformable shape models learnt from images. In: CVPR. (2007)
6. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: ECCV Workshop on Statistical Learning in Computer Vision. (2004)
7. Kumar, M.P., Torr, P., Zisserman, A.: OBJ CUT. In: CVPR. (2005)
8. Levin, A., Weiss, Y.: Learning to combine bottom-up and top-down segmentation. In: ECCV. (2006)
9. Gould, S., Fulton, R., Koller, D.: Decomposing a scene into geometric and semantically consistent regions. In: ICCV. (2009)
10. Winn, J., Jojic, N.: Locus: Learning object classes with unsupervised segmentation. In: ICCV. (2005)
11. Ramanan, D.: Learning to parse images of articulated objects. In: NIPS. (2006)
12. Bray, M., Kohli, P., Torr, P.H.S.: Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In: ECCV. (2006)
13. Chen, Y., Zhu, L., Yuille, A.L., Zhang, H.: Unsupervised learning of probabilistic object models (poms) for object classification, segmentation and recognition using knowledge propagation. PAMI (2009)
14. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In: CVPR. (2004)
15. Fink, M., Ullman, S.: From aardvark to zorro: A benchmark of mammal images. In: IJCV. (2008)
16. Torralba, A., Murphy, K., Freeman, W.: Contextual models for object detection using boosted random fields. In: NIPS. (2005)
17. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV. (2006)
18. Greig, D.M., Porteous, B.T., Seheult, A.H.: Exact maximum a posteriori estimation for binary images. In: Royal Stats. Society. (1989)
19. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. In: Royal Stats. Society. (1977)
20. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. PAMI (2002)
21. Elidan, G., McGraw, I., Koller, D.: Residual belief propagation: Informed scheduling for async. message passing. In: UAI. (2006)
22. Komodakis, N., Paragios, N., Tziritas, G.: Mrf optimization via dual decomposition: Message-passing revisited. In: ICCV. (2007)