Enhancing Interactive Image Segmentation with Automatic Label Set Augmentation

Lei Ding and Alper Yilmaz

Photogrammetric Computer Vision Lab
The Ohio State University
dinglei@cse.ohio-state.edu, yilmaz.15@osu.edu

Abstract. We address the problem of having insufficient labels in an interactive image segmentation framework, for which most current methods would fail without further user interaction. To minimize user interaction, we use the appearance and boundary information synergistically. Specifically, we perform distribution propagation on the image graph constructed with color features to derive an initial estimate of the segment labels. Following that, we include automatically estimated segment distributions at "critical pixels" with uncertain labels to improve the segmentation performance. Such estimation is realized by incorporating boundary information using a non-parametric Dirichlet process for modeling diffusion signatures derived from the salient boundaries. Our main contribution is fusion of image appearance with probabilistic modeling of boundary information to segment the whole-object with a limited number of labeled pixels. Our proposed framework is extensively tested on a standard dataset, and is shown to achieve promising results both quantitatively and qualitatively.

1 Introduction

Image segmentation can be defined as the process of partitioning an image into regions corresponding to potential objects and their backgrounds. Over the course of years, image segmentation techniques without human interaction have not produced satisfactory results. In fact, fully automated segmentation is known to be an ill-posed problem due to the fact that there is (1) no clear definition of a correct segmentation; (2) no agreed-upon objective measure that defines the goodness of a segment, albeit the quality of a segment can be assessed [23] and that of a segmentation can be learned to some extent [21]. In order to do a semantically meaningful segmentation, it is essential to take *a priori* image information into account. This issue has been addressed in the literature as interactive image segmentation, which has been successfully applied in numerous articles [15,19,6,13,12,16,22]. A popular approach for user interaction is through a set of strokes or a trimap [24,5] providing known labels at certain pixels that are called seeds, from which segment labels at other pixels are to be predicted.

Although interactive image segmentation has drawn much attention, little has been done to study the problem of insufficient labels. For instance in Figure 1, in the case there is no label for the duck's beak, a typical interactive segmentation method would fail to segment it as part of the duck. A partial solution to this problem is *active label set augmentation*, which instantiates active learning [25], a framework that allows

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part VI, LNCS 6316, pp. 575–588, 2010.

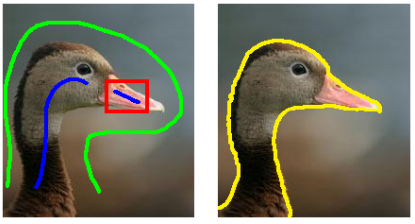

Fig. 1. When labeled pixels are insufficient, interactive image segmentation can perform badly. Our approach automatically introduces labels at critical pixels. Left: labeled object and background pixels are shown as strokes, and the automatically added labels are inside the rectangle. Right: the resulting object contour.

the learner to ask for informative labeled examples at certain costs. However, the luxury of additional labeled information is not always available. Therefore in this paper, we propose *automatic label set augmentation* to address the deficiencies in labeling. In particular, some pixels which we refer to as *critical pixels* have uncertain segment labels. The proposed scheme introduces labels at these critical pixels automatically to help make better segmentation decisions.

Specifically, our work utilizes information available at salient boundaries [18]. In other words, our method does not only merge pixels similar in appearance to the seed pixels in order to form segments; it also introduces unlabeled pixels as seeds in an automatic fashion based on nonlinear modeling of the compatibility between salient boundaries and target segments which leverages a non-parametric classification technique [26]. Our implicit assumption is that pixels within the same object, although may have distinct appearance features (e.g. colors), share similar spatial relations to salient boundaries. Intuitively, such addition of labeled information is vital to whole-object segmentation whose goal is to cut semantic objects from images with little user interaction. While the proposed method does not conclude research on whole-object segmentation, due to lack of high-level semantic knowledge, it provides a reasonable guess for previously unknown segment labels at critical pixels. Finally, the *distribution propagation* framework [8,28], which we adopt to integrate appearance and boundary cues, can be seen as an alternative to previously used graph based label propagation [31,13,12]. Because of the uncertain nature of added segment labels, the way of encoding the label or its strength as a real number at each vertex becomes inadequate, whereas the distribution propagation framework serves as a natural mechanism to integrate additional segment labels probabilistically.

Related Work. Our labeling method is most relevant to the segmentation framework introduced in [13] and recently used in [12] as label propagation, where the segment labels are encoded as a real-valued function at vertices. Label propagation for segmentation is commonly considered in a transductive setting, where known segment labels at some pixels are provided and labels at other pixels are to be estimated by minimizing a certain cost function. Besides, choosing a threshold for determining the segment labels is a difficult problem [31]. Generally, researchers adopt *ad hoc* techniques, e.g. class mass normalization [31] or adaptive window selection [14], which do not necessarily

generalize well. In contrast, rigorous comparison of probabilities is made possible by using the distribution representation.

Recent years have seen much progress on estimating or utilizing salient boundaries [18,17,2,3], which are related to the main contribution of this paper. In two such papers [2,3], the authors study image segmentation from boundaries. However, their goal is either to over-segment images [3], or to extract semantically meaningful objects with sufficient labels [2]. Our work extends these methods by making the object/background inference based on the boundary information through a non-parametric learning framework and missing labels are directly handled. Among the few works addressing whole-object segmentation, an approach based on image matting is presented in [27]. In principle, our diffusion process for generating boundary related features is similar to matting. However, we only use boundary information in constructing feature vectors when appearance fails to provide sufficient evidence for object/background discrimination, whereas in [27], the authors rely on image appearance for generating mattes.

There are several papers on using priors for describing general shape or configuration of an object [29,16,9], which segment out the object in a graph-cut setting. However, the priors used limit their applicability to certain class of images to which the priors are suited. Our work can be considered complementary to theirs in terms of the underlying approach and the overall effects of segmentation, which involve automatically added labels. To better present our work, we briefly introduce the six major steps with more details to follow in their corresponding sections:

1. Distribution propagation using color features (Sec. 2).
2. Let the labeled set be $\mathcal{L}$ and the unlabeled set be $\mathcal{U}$. Identify a subset $\mathcal{A} \subset \mathcal{U}$ where the estimated distribution is ambiguous, i.e., the difference between the probabilities of being object and background is small. Also identify $\mathcal{U}^* \subset \mathcal{U} - \mathcal{A}$, which represents the set of pixels where the estimated distribution is informative, i.e., the probability difference is significant (Sec. 3);
3. Generate boundary probability map I_b indicating the probability of each pixel on a boundary (Sec. 3.1);
4. Compute diffusion signature vectors, which are the feature vectors at each pixel, from I_b (Sec. 3.1);
5. Modeling and classification using Dirichlet process multinomial logistic model, for which the training and test sets consist of feature vectors corresponding to $\mathcal{L} \cup \mathcal{U}^*$ and $\mathcal{A}$ respectively (Sec. 3.2);
6. Automatically generate segment distributions on $\mathcal{A}$. Let the labeled set be $\mathcal{L} \cup \mathcal{A}$, and the unlabeled set be $\mathcal{U} - \mathcal{A}$. Then proceed with the distributions computed by step (1) to derive new segment labels on $\mathcal{U}$, after which the algorithm stops.

2 Distribution Propagation for Labeling

In a graph representing an image, we have a set of n vertices, each of which corresponds to either a labeled or an unlabeled pixel, and a set of edges connecting neighboring pixels. Edges are denoted as $e_1, e_2, \cdots, e_m$, each of which is a binary set of

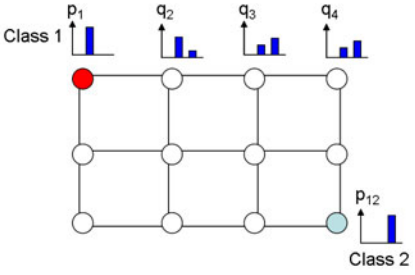

Fig. 2. Illustration of distribution propagation on an image graph. Lines connecting pixels are edges. Distributions, including known p_i's and estimated q_i's, are shown as histograms.

vertices. For representing the appearance, we consider the LUV color space due to its perceptual uniformity [11]. The LUV values of a pixel result in a feature vector $\mathcal{F}_S$ for each pixel S: $\mathcal{F}_S = (L_S, U_S, V_S)$. We define the weight of an edge as: $w(e) = \exp\left(-\sum_{i=1}^{3} \frac{(\mathcal{F}^i(v_1)-\mathcal{F}^i(v_2))^2}{2\delta_i^2}\right)$, where $v_1, v_2 \in e$, and $\mathcal{F}^i(v_j)$ is the i^{th} feature value at the pixel represented by vertex v_j. We pose image segmentation as propagating distributions at labeled vertices to unlabeled ones. Specifically, let us consider a segmentation setting where the set $\mathcal{Y}$ represents l label types which correspond to the objects and their backgrounds, $\mathcal{Y} = \{1, 2, \cdots, l\}$. A labeled vertex $i \in \mathcal{L}$ of class k is assigned a known distribution $p_i(y)$, such that $p_i(k) = 1$ and $p_i(y) = 0$ for $y \neq k$. The estimated distribution at an unlabeled vertex $i \in \mathcal{U}$ is defined as a multinomial distribution $q_i(y)$, which can be described by an l-dimensional vector, $[q_i(y=1), q_i(y=2), \cdots, q_i(y=l)]^T$ of non-negative elements summing to one. Such a distribution can be used for classification of l classes with the maximum likelihood principle, such that, the vertex is labeled as a member of class $y^* = \arg\max_y q_i(y)$.

We estimate the distributions q_i for unlabeled vertices based on the distributions p_i of labeled vertices and the neighborhood relations encoded by e_i. Our main assumption is that multinomial distributions q_i should be similar to each other inside each edge. The general setting of distribution propagation is illustrated in Figure 2, where we have two classes (marked with different colors), the labeled distributions are p_1 and p_{12}, and the estimated distributions are q_i. We describe the discrepancy between two probability distributions p and q by the Kullback-Leibler (KL) divergence, $D(p, q) = \sum_{y \in \mathcal{Y}} p(y) \log \frac{p(y)}{q(y)}$. We also define the exponential centroid distribution of an edge e_k to be:

$$q_k^E(y) = \frac{1}{Z_k} \exp\Big(\sum_{i \in e_k} h_{i,k} \log q_i(y)\Big), \tag{1}$$

with Z_k being a normalization constant, and $h_{i,k} = 1/|e_k|$, or 0.5 in our case. Next we can formulate the optimization on the image graph as:

$$\arg\min_{q_i} \sum_{i=1}^{n} r_i D(p_i, q_i) + \sum_{k=1}^{m} w_k \sum_{i \in e_k} h_{i,k} D(q_k^E, q_i), \tag{2}$$

where r_i is 0 for an unlabeled vertex and a positive constant for a labeled vertex. Further, we can relax (2) to the following with the optimal solution q_i^* to be the same:

$$\arg\min_{q_i,\eta_k} \sum_{i=1}^{n} r_i D(p_i, q_i) + \sum_{k=1}^{m} w_k \sum_{i \in e_k} h_{i,k} D(\eta_k, q_i), \tag{3}$$

since $q_k^E(y)$ is the minimizer η_k^* for $J(\eta_k) = \sum_{i \in e_k} h_{i,k} D(\eta_k, q_i)$ [1]. Note that the function in (3) can be decomposed vertex-wise [28]. Based on this observation, and $\sum_{y \in \mathcal{Y}} q_i(y) = 1$, the following decomposed sub-problem for vertex i

$$\arg\min_{q_i} r_i D(p_i, q_i) + \sum_{\{k: i \in e_k\}} h_{i,k} w_k D(\eta_k, q_i), \tag{4}$$

is solved in closed form by:

$$q_i(y) = \frac{1}{r_i + d_i}(r_i p_i(y) + \sum_{\{k: i \in e_k\}} h_{i,k} w_k \eta_k(y)), \tag{5}$$

where $d_i = \sum_{\{k: i \in e_k\}} h_{i,k} w_k$ is the vertex degree.

We take r_i to be a large number, and thus for a labeled vertex, $q_i(y) = p_i(y)$ meaning that our method performs interpolation respecting the user supplied labels. It can be shown that the function to optimize in (2) is convex with respect to all the $q_i(y)$ terms [28], and thus a *unique* solution is guaranteed. This is a nice property for the proposed framework, as after we add in the automatically generated distributions at the critical pixels, it is unnecessary to start over from the initial uniform distributions at other unlabeled vertices. Finally, the algorithm with guaranteed convergence is posed as iterations over the two major steps: (1) compute the centers $q_k^E(y)$ for all k, and set $\eta_k = q_k^E$; (2) solve the optimization sub-problem at each vertex i by updating $q_i(y)$ as in (5).

3 Incorporating Boundary Information

After distribution propagation on a graph corresponding to the image, we obtain reliable estimates of segment labels at pixels where there is minimal ambiguity based on appearance features. For certain regions in the image, prior segment labeling may be missing. We will refer to pixels in such regions as critical pixels. In general, critical pixels are inside segments whose corresponding vertices are disconnected from (or weakly connected to) both kinds of labeled vertices. For single object segmentation, we refer to the object and background as o^1 and o^2 respectively. Intuitively, the pixels with $t_i = |\log \frac{q_i(o^1)}{q_i(o^2)}| < \tau$, where τ is a small positive number, are selected as critical pixels, and thus $\mathcal{A} = \{i : i \in U \wedge t_i < \tau\}$. Such intuition is formalized as a decision problem using the maximum a posteriori (MAP) rule. Specifically, the two classes are likely correct (C^1) and likely wrong (C^0) predictions resulting from the distribution propagation. Class C^0 corresponds to critical pixels, which are equivalent to those pixels with $P(C^0|t) > P(C^1|t)$, where $P(C^j|t) \propto P(t|C^j)P(C^j)$. Such probabilities can be computed from a set of images with ground truth segmentation, and thus the threshold

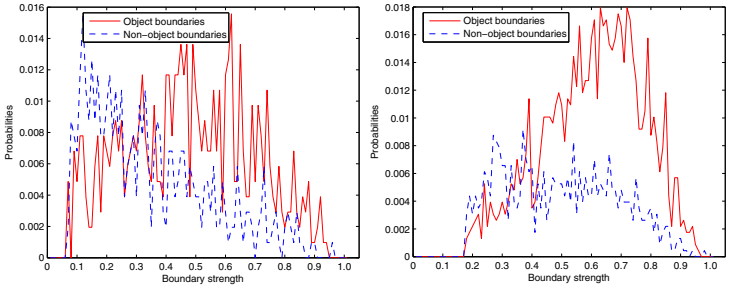

Fig. 3. Joint probabilities $P(strength, O)$ versus $P(strength, N)$, where the two classes of boundaries are object (O) and non-object (N). The relative greatness in them determines which class is more likely. It follows that higher strength implies higher chance of being an object boundary.

is set as $\tau = \max_t\{t : P(C^0|t) > P(C^1|t)\}$. Besides, in training the Dirichlet process based model, we set $\mathcal{U}^* = \{i : i \in \mathcal{U} \wedge t_i > \kappa\}$, such that $P(t > \kappa|C^0) < 5\%$.

We will next discuss a non-parametric Bayesian approach to predict segment distributions at critical pixels, which are then incorporated to the known set of labels. The distribution representation gives a principled encoding of the added segment labels with uncertainty. In regards to labeling weights, unlike original labels in which we have total confidence, we set r_i to be a small number for $i \in \mathcal{A}$, such that the effect of occasional erroneous labels can be offset by correct manual labels. After the label set augmentation, we resume, instead of restart, the distribution propagation to derive an updated segmentation result with newly added segment label distributions.

3.1 Boundary Induced Diffusion Signatures

In order to facilitate the incorporation of salient boundaries, we adopt the boundary probability described in [18]. Intuitively, the boundaries internal to semantic objects are weaker in strength (i.e., the mean value of boundary probabilities) than the external ones that outline objects. Such a claim is validated by empirical analysis on GrabCut and LHI datasets [24,30] with object-level segmentation ground truth. In this analysis, we observe that high strength in boundaries is strongly correlated with large probability of being true object boundaries, as plotted in the left (GrabCut dataset) and right (LHI dataset) panels of Figure 3 respectively.

As a result, the spatial location of a pixel with respect to the boundaries provides useful cues for determining its segment label. To get a reliable representation of such information, we first threshold and transform the probability map I_b to a set of boundary fragments $B = \{B_i\}$ by eliminating boundary pixels with small probability values, where each boundary fragment B_i is identified with the set of pixels that it intersects. An undirected boundary characteristic graph G_b is defined as: $G_b = (V, E)$, where the vertex set $V = \{v_i\}$ represents the set of pixels. The edge set $E = E^0 - E^*$, where $E^0 = \{(v_i, v_j) : v_i \text{ and } v_j \text{ are neighbors}\}$ is the full set of initial edges, and

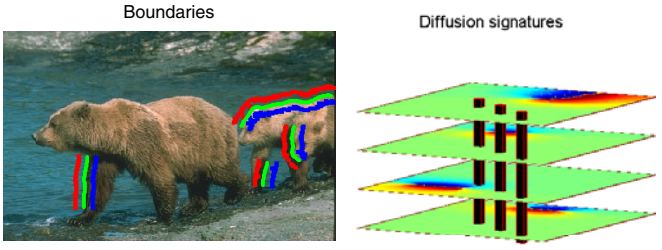

Fig. 4. Left: an image with four displayed boundary fragments together with red (+) and blue (-) heat sources. Right: the corresponding dimensions of diffusion signatures.

$E^* = \{(v_i, v_j) : \exists k \text{ s.t. } v_i \in B_k \vee v_j \in B_k\}$ is the set of edges that must be disconnected to reflect boundary constraints. By doing so, the salient boundaries are transformed to the characteristic graph G_b.

Diffusion signatures are feature vectors at vertices of G_b, and are derived using a diffusion process on the graph. Specifically, the i^{th} dimension of the diffusion signature vector corresponds to the i^{th} boundary fragment, and we place labeling "heat sources" on its both sides (left panel, Figure 4) to generate numerical descriptions. Let $f^i : V \rightarrow [-1, +1]$ be a function, $S = \{S_j\}$ be the set of vertices corresponding to labeling heat sources and U be the set of unlabeled vertices. Let λ be the mean value of boundary probabilities at pixels inside B_i. We proceed by assigning $+\lambda$ and $-\lambda$ as the values of function $f^i(S_j)$ on the two sides of the boundary fragment respectively, such that $f^i(\cdot)$ takes the opposite values on its two sides. In a vector form $\mathbf{f}_S^i = [f^i(S_1), f^i(S_2), \cdots, f^i(S_{|S|})]^T$. The stationary solution vector at unlabeled vertices is computed as: $\mathbf{f}_U^i = -\Delta_{U,U}^{-1}\Delta_{U,S}\mathbf{f}_S^i$, where $\Delta_{\cdot,\cdot}$ is a sub-Laplacian matrix of the graph G_b using block matrix notation [13]. Thus, the i^{th} dimension of diffusion signature at vertex v is defined as $f^i(v)$, which equals $\mathbf{f}_U^i(v)$ if $v \in U$, or $\mathbf{f}_S^i(v)$ if $v \in S$. In a similar vein, the diffusion signature vector at v is $[f^1(v), f^2(v), \cdots, f^{|B|}(v)]^T$. Example diffusion signatures associated with some boundary fragments are displayed on an image in the right panel of Figure 4, where red and blue refer to positive and negative values respectively. Properties of diffusion signatures in a single dimension include (1) the farther the pixels are from the boundary in the diffusion sense, the smaller the absolute values are; (2) the border at which $\{f^i(v) > 0\}$ and $\{f^i(v) < 0\}$ meets corresponds to a natural extension of the original boundary.

3.2 Non-linear Modeling and Classification

Now we present how the distribution at an unlabeled pixel $i \in \mathcal{A}$ is estimated using boundary information. The inputs to this process are the diffusion signatures derived from salient boundaries. The classification method we adopt is akin to Dirichlet process mixture models (DPMMs). Unlike Gaussian mixtures, they allow for automatic determination of the number of clusters. Our method is a semi-supervised extension of a recent DPMM based classification technique termed as Dirichlet process multinomial

logistic model (DPMNL) [26]. We use semi-supervised learning to directly address the problem of insufficient labeling in interactive image segmentation.

In this model, a cluster refers to a group of salient diffusion signatures corresponding to structures in the image. A single cluster might contain both object and background pixels, which is accounted for by a multinomial logistic relation between the input diffusion signatures $\mathbf{x}$ and an output class y. Specifically, each cluster in the mixture model has parameters $\theta = (\mu, \Sigma, \alpha, \beta)$. The distribution of $\mathbf{x}$ within the cluster follows a Gaussian model $\mathcal{N}(\mu, \Sigma)$, where Σ is a diagonal matrix with elements σ_i^2. The distribution of y given $\mathbf{x}$ within the cluster follows a multinomial logistic model $P(y = j|\mathbf{x}, \alpha, \beta) = \frac{\exp(\alpha_j + \mathbf{x}^T \beta_j)}{\sum_{k=1}^{J} \exp(\alpha_k + \mathbf{x}^T \beta_k)}$, where J is the number of segments and equals 2 for single object segmentation. In the left panel of Figure 5, we illustrate the graphical model representation of DPMNL, which reveals the interdependencies among parameters and variables. Specifically, the model parameters θ are drawn from a distribution G that is drawn from a Dirichlet process $\mathcal{D}(G_0, \gamma)$, where G_0 is a base distribution over model parameters and γ is a scale parameter with a gamma prior. G_0's parameters may in turn depend on higher-level hyperparameters. However, we use fixed distributions for simplicity and they led to good performance: $\mu_i \sim \mathcal{N}(0, 1)$, $\log(\sigma_i^2) \sim \mathcal{N}(0, 1)$, $\alpha_j \sim \mathcal{N}(0, 1)$, $\beta_j \sim \mathcal{N}(\mathbf{0}, I)$, where i and j are for a feature dimension and a class respectively, and I is an identity matrix. Without loss of generality, we graphically show the effect of the model on a toy scenario in the right panel of Figure 5, where the two classes are displayed as dots and squares.

We use the Markov chain Monte Carlo (MCMC) algorithm with auxiliary parameters [20] for posterior sampling, which iterates over updating the data to cluster assignment and the cluster parameters. Our problem is relatively small in scale with several thousand training examples per image, and thus computational complexity of MCMC is affordable. In the main iteration of MCMC, we use semi-supervised learning to make use of the unlabeled pixels. Specifically,

$$E\{P(y_i|\mathbf{x}_i)\} = P(y_i = 1|\mathbf{x}_i)q_i(1) + P(y_i = 2|\mathbf{x}_i)q_i(2), \tag{6}$$

which is used in place of the conditional $P(y_i|\mathbf{x}_i)$ of an unlabeled training example (i.e., $i \in \mathcal{U}^*$). Inside this equation, $q_i(\cdot)$ values are obtained by distribution propagation. By doing so, the unlabeled examples are effectively used, instead of being discarded in the training process as in [26].

Once we obtain post-convergence parameters $\theta_i^t = (\mu_i^t, \Sigma_i^t, \alpha_i^t, \beta_i^t)$, for $t = 1, \cdots, T$ and $i = 1, \cdots, |\mathcal{L}| + |\mathcal{U}^*|$, where T is the maximum index of iteration, they are used to estimate the predictive distribution of the class label y_* for a new input diffusion signature vector $\mathbf{x}_*$:

$$P(y_* = j|\mathbf{x}_*) = \frac{\sum_{t=1}^{T} \int P(y_* = j, \mathbf{x}_*|\theta_*^t) P(\theta_*^t|\theta^t, G_0) d\theta_*^t}{\sum_{t=1}^{T} \int P(\mathbf{x}_*|\theta_*^t) P(\theta_*^t|\theta^t, G_0) d\theta_*^t}, \tag{7}$$

where the test example's parameters θ_*^t are drawn from a distribution that is drawn from a Dirichlet process $\mathcal{D}(G_0, \gamma)$: $\theta_*^t \sim \frac{1}{n+\gamma} \sum_{i=1}^{n} \delta(\theta_i^t) + \frac{\gamma}{n+\gamma} G_0$, in which $\delta(\cdot)$ is a distribution concentrated at a single point [4], and $n = |\mathcal{L}| + |\mathcal{U}^*|$. Therefore, equation (7) allows for numerical computation according to the assumed and derived distributions. Note that the predictive distribution is not based on a single parameter estimate,

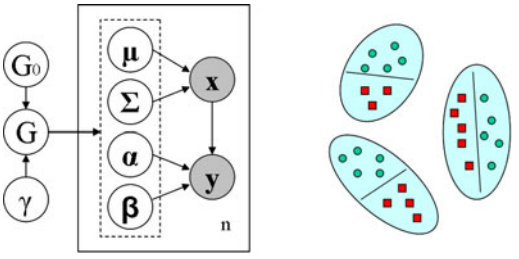

Fig. 5. Left: graphical model representation of DPMNL, where n is the number of training examples. Right: an illustration of its effect on data with 3 clusters. Inside each cluster, a linear decision boundary separates data from the 2 classes. The overall decision boundary is non-linear and not shown here.

but is an average of the predictions using all possible values of the parameters, each of which is weighted by the probability of the parameters having those values. In this way, DPMNL avoids making hard decisions on assigning diffusion signatures to clusters and thus provides flexibility in modeling the boundary information. Finally, each test diffusion signature vector is assigned to a segment class with the highest predictive probability, i.e., $\hat{y}_* = \arg\max_j P(y_* = j|\mathbf{x}_*)$ for the visualization of added labels. However, for the segmentation task, we set $p_i(y = j) = P(y_* = j|\mathbf{x}_*)$ for $i \in \mathcal{A}$ to accommodate the probabilistic nature of added labels.

4 Experiments

In order for quantitative evaluation, we perform experiments on the GrabCut dataset [24], which is one of the few datasets providing both labeling trimaps and segment ground truth. Labeled object, labeled background and unlabeled pixels are reflected in the trimaps as white, dark gray and light gray respectively. In experiments, we stick to original trimaps as provided in the dataset. We will report the performance on *set*-50 (the whole GrabCut dataset) and *set*-25 (a subset where there are a significant number of added labels). While our method helps boost the performance on the whole set, as can be seen, it enhances the performance on *set*-25 by a larger margin, which contains difficult instances due to the lack of labels at numerous critical pixels.

Details of implementation. In implementing our framework, we use superpixels [23] in place of pixels for computational efficiency, such that a superpixel, which roughly contains 15 pixels, is assumed the smallest unit for labeling. In computing diffusion signatures, boundary fragments are generated by thresholding a boundary probability map [18] at its 60^{th} percentile of non-zero values. The boundary fragments with junctions are broken into smaller ones, so that they contain simple curves for generating diffusion signatures. We also project the diffusion signatures onto the subspace learned by principal component analysis (PCA) and retain the top half dimensions to have a compact representation. Thus, the constituent dimensions of small variances resulting

from boundaries with small strength are filtered out. In constructing the set $\mathcal{A}$ for automatic label set augmentation as discussed in Section 3, with an empirical analysis of the segmentation results on the GrabCut dataset using distribution propagation, τ is chosen using an MAP procedure, such that $\tau = \max_t\{t : P(C^0|t) > P(C^1|t)\} \approx 0.004$. Finally, κ is chosen as 0.845, such that $P(t > \kappa|C^0) < 5\%$, i.e., a label in $\mathcal{U}^*$ is very unlikely to be wrong.

Methods for comparison. We have used label propagation (LP) and distribution propagation (DP) as the underlying labeling methods. Together with them, we study several approaches for label set augmentation as detailed next.

- LP-original: An implementation of the random walk approach [13], where the optimal label values at unlabeled vertices are derived by solving linear equations and are thresholded to produce segment labels.
- DP-original: distribution propagation as introduced in Section 2, where in contrast to the label value representation at each vertex, a distribution is used.
- DP-GMM: DP with added labels estimated using Gaussian mixture models learned with EM [10]. A mixture model is trained for each segment class.
- DP-SVM: DP with added labels estimated using support vector machines with radial basis function kernels and probability outputs implemented in [7].
- DP-object: DP with added labels systematically being object, which produces considerably better performance than randomly guessing on the GrabCut dataset.
- DP-DPMNL: DP with added labels estimated with DPMNL, which is the proposed method.
- DP-DPMNL-AT: DP with added labels estimated with DPMNL; however, the threshold for distribution propagation is set according to adaptive thresholding [14].
- LP-DPMNL: LP with the same added labels as above.

Evaluation. For quantitative evaluation of object-background segmentation on the GrabCut dataset, we compute the error rate as the percentage of wrongly labeled pixels in the original unlabeled region. In addition, we demonstrate the error rate of automatic label set augmentation, which is the percentage of wrongly added labels at critical pixels. We note that while the added labels are not 100% accurate, distribution propagation uses these additional labels in a soft and flexible way, so the overall enhancement is not compromised by occasional erroneous labels.

The results are tabulated in Table 1, where *seg*-50/25 refers to the average error rate of segmentation and *aug*-50/25 refers to the average error rate of label set augmentation. Among all the compared methods, the proposed DP-DPMNL outperforms the others without heuristic thresholding. It can be further seen that the distribution propagation alone performs better than the label propagation, such as an error rate of 5.4% reported for a recent method detailed in [12] which uses sophisticated Laplacians. All label set augmentation methods that we use help to achieve better segmentation accuracy but at different levels; DPMNL provides better results than the other three alternatives. In particular, we attribute the less competitive performance of SVMs to their inability in modeling the cluster structures of diffusion signatures. Besides, DP-DPMNL outperforms

LP-DPMNL, which justifies the choice of distribution propagation as the underlying labeling mechanism.

A comparison with the published average error rates on the entire dataset using several recent approaches also shows that our method performs the best. In particular, our proposed method gives 3.58%, considerably better than error rates achieved in previous works [5,13,12,11]. Together with adaptive thresholding [14], our proposed method produces an error rate 3.08%, which is better than 3.3% that is generated using s-Laplacian together with adaptive thresholding. Despite this fact, we stress that the thresholding technique has heuristic nature. Thus, DP-DPMNL remains the proposed method of this paper and is used to generate all the remaining results, even though DP-DPMNL-AT can give better quantitative results given the GrabCut type of trimaps.

Table 1. Performance in terms of the average error rate in segmentation (*seg*) and that in label set augmentation (*aug*) on the whole set of 50 images and a smaller set of 25 where there are a significant number of missing labels. Compared are variations of our proposed method and recent baselines in the literature. Our proposed method (DP-DPMNL) has shown the best results among all, except for methods combined with adaptive thresholding [14].

Methods	*seg*-50	*aug*-50	*seg*-25	*aug*-25
LP-original	5.92%	–	7.03%	–
DP-original	5.22%	–	6.84%	–
DP-GMM	4.68%	25.25%	5.33%	26.72%
DP-SVM	4.49%	22.08%	5.01%	22.87%
DP-object	5.04%	26.61%	5.66%	30.61%
DP-DPMNL	**3.58%**	**11.08%**	**3.85%**	**15.35%**
DP-DPMNL-AT	**3.08%**	–	**3.14%**	–
LP-DPMNL	4.82%	–	5.63%	–
GM-MRF [5]	7.9%	–	–	–
Random walk [13]	5.4%	–	–	–
s-Laplacian [12]	5.4%	–	–	–
s-Laplacian-AT [12]	3.3%	–	–	–
Hypergraph [11]	5.3%	–	–	–

Sample segmentation results and automatically added labels by using DP-DPMNL are shown in Figure 7 for qualitative evaluation.[1] From top down, the four rows are respectively: original trimaps, results from the baseline method (DP-original), automatically added labels where red is for object and green is for background, and new results using added labels. As can be observed, the proposed method provides smooth object contours while preserving most details. Both missing object labels and background labels can be added at their appropriate places. Besides, for one of the images hard to DP-original due to insufficient labels, the evolution of the computed object contour is visualized in Figure 6, before (iterations 0 to 10) and after (11 to 20) the label set augmentation, where iteration 0 refers to the initial condition.

[1] Additional qualitative segmentation results in the same format are shown in `http://www.cse.ohio-state.edu/~dinglei/autogen.htm`

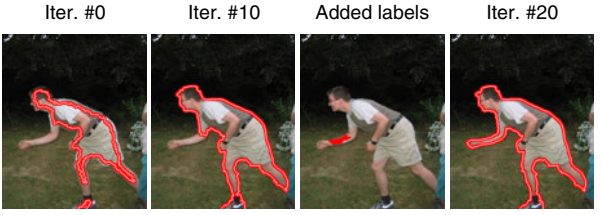

Fig. 6. Contour evolution by our approach. Please refer to Figure 7 for the provided trimap that contains insufficient labels.

Fig. 7. Provided trimaps, baseline results, automatically added labels (red: object, green: background) and new segmentation results

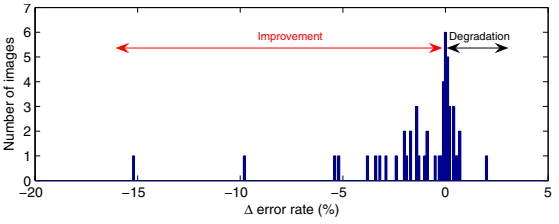

Fig. 8. Histogram of the change in error rates, or the error rate of using DP-DPMNL minus that of using DP-original, on *set*-50. Improved cases are marked red, and the degraded are in black.

5 Discussion and Conclusions

Although user interaction has been widely used for segmentation, the insufficiency in provided labeling seeds has drawn little attention. In this paper, we addressed this by exploiting the boundary information. Our experiments on the GrabCut dataset have shown that automatically added labels helped the segmentation process at different success levels. The histogram of change in segmentation error rates with label set augmentation is shown in Figure 8. We note that the proposed method has dramatically reduced the error rates in many cases. There is only one case where the added labels reduced the accuracy rate by more than 1% (which is 2.0%). This reduction in quantitative performance was due to similarity between the object and its shadow which resulted in labeling the shadow as the object. However, the result does not deteriorate qualitatively.[2] Besides, the automatically added labels can be prompted to the user to accept or reject in an interactive environment, which is a functionality not previously offered.

To summarize, we have presented a framework using distribution propagation to address interactive image segmentation. A key component of our framework is the automatic augmentation of labels at critical pixels via a Dirichlet process based non-linear model. Extensive experiments have shown that the proposed framework performs competitively in predicting critical missing labels and enhancing the overall segmentation results. In particular, using a limited number of supplied labels, we have achieved both qualitatively and quantitatively excellent results on the GrabCut dataset.

References

1. Akaho, S.: The e-PCA and m-PCA: Dimension reduction of parameters by information geometry. In: IJCNN (2004)
2. Arbelaez, P., Cohen, L.: Constrained image segmentation from hierarchical boundaries. In: CVPR (2008)
3. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: From contours to regions: An empirical evaluation. In: CVPR (2009)
4. Blackwell, D., MacQueen, J.B.: Ferguson distributions via polya urn scheme. Annals of Statistics 1, 353–355 (1973)
5. Blake, A., Rother, C., Brown, M., Perez, P., Torr, P.: Interactive image segmentation using an adaptive GMMRF model. In: ECCV (2006)

[2] For qualitative comparison, please look at the first column, bottom group of Figure 7.

6. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In: ICCV (2001)
7. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001) (software available online)
8. Corduneanu, A., Jaakkola, T.: Distributed information regularization on graphs. In: NIPS (2005)
9. Das, P., Veksler, O., Zavadsky, V., Boykov, Y.: Semiautomatic segmentation with compact shape prior. Image and Vision Computing 27(1-2), 206–219 (2009)
10. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, Series B 39(1), 1–38 (1977)
11. Ding, L., Yilmaz, A.: Interactive image segmentation using probabilistic hypergraphs. Pattern Recognition 43(5) (2010)
12. Duchenne, O., Audibert, J.Y., Keriven, R., Ponce, J., Segonne, F.: Segmentation by transduction. In: CVPR (2008)
13. Grady, L.: Random walks for image segmentation. IEEE PAMI 28(11), 1768–1783 (2006)
14. Guan, J., Qiu, G.: Interactive image segmentation using optimization with statistical priors. In: ECCV Workshops (2006)
15. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. IJCV 1(4), 321–331 (1987)
16. Lempitsky, V., Kohli, P., Rother, C., Sharp, T.: Image segmentation with a bounding box prior. In: ICCV (2009)
17. Maire, M., Arbelaez, P., Fowlkes, C., Malik, J.: Using contours to detect and localize junctions in natural images. In: CVPR (2008)
18. Martin, D., Fowlkes, C., Malik, J.: Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE PAMI 26(5), 530–549 (2004)
19. Mortensen, E.N., Barrett, W.A.: Intelligent scissors for image composition. In: SIGGRAPH (1995)
20. Neal, R.: Markov chain sampling methods for Dirichlet process mixture models. Journal of Computational and Graphical Statistics 9(2), 249–265 (2000)
21. Peng, B., Veksler, O.: Parameter selection for graph cut based image segmentation. In: BMVC (2008)
22. Price, B.L., Morse, B., Cohen, S.: Geodesic graph cut for interactive image segmentation. In: CVPR (2010)
23. Ren, X., Malik, J.: Learning a classification model for segmentation. In: ICCV (2003)
24. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. In: SIGGRAPH (2004)
25. Settles, B.: Active learning literature survey. Tech. rep., University of Wisconsin-Madison Computer Sciences Technical Report 1648 (2009)
26. Shahbaba, B., Neal, R.: Nonlinear models using Dirichlet process mixtures. JMLR 10, 1755–1776 (2009)
27. Stein, A.N., Stepleton, T.S., Hebert, M.: Towards unsupervised whole-object segmentation: Combining automated matting with boundary detection. In: CVPR (2008)
28. Tsuda, K.: Propagating distributions on a hypergraph by dual information regularization. In: ICML (2005)
29. Veksler, O.: Star shape prior for graph-cut image segmentation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 454–467. Springer, Heidelberg (2008)
30. Yao, Z., Yang, X., Zhu, S.C.: Introduction to a large scale general purpose ground truth dataset: Methodology, annotation tool, and benchmarks. In: Yuille, A.L., Zhu, S.-C., Cremers, D., Wang, Y. (eds.) EMMCVPR 2007. LNCS, vol. 4679, pp. 169–183. Springer, Heidelberg (2007)
31. Zhu, X., Ghahramani, Z., Lafferty, J.: Semi-supervised learning using Gaussian fields and harmonic functions. In: ICML (2003)